



**HAL**  
open science

# Conception, développement et évaluation de techniques d'interactions fluides pour des environnements multidimensionnels : application aux logiciels du service public

Gary Perelman

## ► To cite this version:

Gary Perelman. Conception, développement et évaluation de techniques d'interactions fluides pour des environnements multidimensionnels : application aux logiciels du service public. Interface homme-machine [cs.HC]. Université Paul Sabatier - Toulouse III, 2018. Français. NNT : 2018TOU30255 . tel-02333703

**HAL Id: tel-02333703**

**<https://theses.hal.science/tel-02333703>**

Submitted on 25 Oct 2019

**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.

Université Fédérale



Toulouse Midi-Pyrénées

# THÈSE

En vue de l'obtention du  
**DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE**

Délivré par l'Université Toulouse 3 - Paul Sabatier

---

Présentée et soutenue par

**Gary PERELMAN**

Le 2 octobre 2018

**Conception, développement et évaluation de techniques  
d'interactions fluides pour des environnements  
multidimensionnels : application aux logiciels du service  
public**

---

Ecole doctorale : **EDMITT - Ecole Doctorale Mathématiques, Informatique et  
Télécommunications de Toulouse**

Spécialité : **Informatique et Télécommunications**

Unité de recherche :

**IRIT : Institut de Recherche en Informatique de Toulouse**

Thèse dirigée par

**Emmanuel DUBOIS et Marcos SERRANO**

Jury

M. Gilles Bailly, Rapporteur

M. Géry Casiez, Rapporteur

Mme Laurence Nigay, Examineur

M. Emmanuel DUBOIS, Directeur de thèse

# REMERCIEMENTS

---

Avant toute chose, je souhaiterais remercier Géry Casiez et Gilles Bailly pour avoir accepté de relire ce manuscrit. Je souhaite aussi remercier Laurence Nigay pour avoir accepté d'être membre et présidente de mon jury de thèse.

Il est évident que ces travaux de thèse n'auraient pas pu voir le jour sans le soutien de mes encadrants de thèse au sein de l'équipe ELIPSE et de Berger-Levrault. Je remercie donc particulièrement Emmanuel Dubois et Marcos Serrano d'avoir accepté de guider mes travaux de thèse et de m'avoir appris, pendant 3 ans, tout ce que je sais de la recherche. Je remercie aussi Berger-Levrault, et l'ensemble de mes encadrants et collègues qui m'ont toujours soutenu, défendu et guidé tout au long de ces années. Merci à Célia Picard, Christophe Bortolaso, Mustapha Derras et Valérie Reiner.

Au-delà de mes encadrants, nombreux ont été ceux qui m'ont soutenu et aidé durant ces dernières années, que ce soit en supportant mes plaintes, en discutant longuement de tout et de rien ou en échangeant nos connaissances pour mieux progresser. Merci à tous les doctorants, stagiaires, post-doc, docteurs, ingénieurs et autres joyeusetés de l'équipe ELIPSE et de Berger-Levrault pour tous ces moments passés avec vous.

Dans un registre plus professionnel, je souhaite aussi tout particulièrement remercier les élus du club MAGEL qui m'ont ouvert leurs portes et m'ont permis de découvrir un univers que je ne connaissais pas. Merci pour votre accueil toujours chaleureux et vos retours toujours pertinents.

Enfin, je souhaite remercier ma famille et ma femme pour m'avoir soutenu durant ces quelques années et pour m'avoir supporté, même dans les moments difficiles.



# Table des matières

INTRODUCTION.....	13
1. CONTEXTE.....	14
2. OBJECTIFS.....	17
3. STRUCTURE DU MANUSCRIT.....	19
CONTEXTE INDUSTRIEL DES TRAVAUX.....	21
1. INTRODUCTION.....	22
1.1. Contexte.....	22
1.2. Structure du chapitre.....	22
2. DESCRIPTION DE L'ENTREPRISE.....	23
2.1. Introduction.....	23
2.2. Description des logiciels Berger-Levrault.....	25
2.2.1. Ville et Territoire.....	25
2.2.2. Domaine de la Santé et du Social.....	29
2.2.3. Conclusion.....	30
3. ÉTUDE DE CAS : LES ÉLUS.....	31
3.1. Description de l'étude.....	31
3.2. Résultats.....	33
3.2.1. Période de travail.....	33
3.2.2. <i>Environnement et outils de travail</i> .....	33
3.2.3. Description des tâches.....	34
3.2.3.1. Relation citoyen.....	34
3.2.3.2. Veille d'actualité et communication.....	36
3.2.3.3. Gestion des finances.....	38
3.3. Conclusion.....	39
4. CONCLUSION DU CHAPITRE 2.....	40

ÉTAT DE L'ART : INTERACTION AVEC DES DONNÉES	
MULTIDIMENSIONNELLES .....	41
1. INTRODUCTION .....	42
2. DISPOSITIFS À MULTIPLES DEGRÉS DE LIBERTÉS ADAPTÉS AUX ENVIRONNEMENTS FIXES .....	44
2.1. Dispositifs isotoniques .....	45
2.2. Dispositifs isométriques.....	47
2.3. Dispositifs élastiques .....	48
2.4. Dispositifs hybrides .....	49
2.5. Autres approches .....	49
3. DISPOSITIFS ET TECHNIQUES D'INTERACTION MOBILES À MULTIPLES DEGRÉS DE LIBERTÉS.....	51
3.1. Multi-touch .....	51
3.2. Stylet.....	53
3.3. Interaction en l'air.....	55
3.4. Interaction tangible avec un dispositif mobile.....	57
3.5. Dispositif mobile augmenté.....	58
4. CONCEPTION DE DISPOSITIF À MULTIPLES DEGRÉS DE LIBERTÉS PAR COMPOSITION.....	60
4.1. Outils de composition en IHM.....	60
4.2. Approches analytiques.....	61
5. SYNTHÈSE.....	66
LA ROLY-POLY MOUSE : CONCEPTION D'UN DISPOSITIF A MULTIPLES	
DEGRES DE LIBERTE.....	68
1. INTRODUCTION.....	70
1.1. Motivation.....	70
1.2. Structure du chapitre.....	71
2. CONCEPTION ET PROTOTYPAGE DE LA ROLY-POLY MOUSE.....	72

2.1.	La Roly-Poly Mouse .....	72
2.1.1.	Facteurs de conception .....	73
2.1.2.	Prototypes développés .....	76
3.	ÉTUDE PRELIMINAIRE : FORME DU DISPOSITIF ET PRISE EN MAIN .	79
3.1.	Description de l'étude .....	79
3.2.	Résultats .....	79
4.	ÉTUDE 1 : AMPLITUDE ET STABILITE DES GESTES .....	81
4.1.	Description de l'étude .....	81
4.2.	Résultats .....	83
5.	ÉTUDE 2 : POINTAGE 2D .....	85
5.1.	Algorithme correctif .....	85
5.2.	Description de l'étude .....	86
5.3.	Résultats .....	88
6.	ÉTUDE 3 : MANIPULATION 3D .....	90
6.1.	Techniques d'interaction pour la manipulation 3D .....	90
6.2.	Description de l'étude : translation et rotation 3D .....	92
6.3.	Résultats .....	95
7.	TRANSFERT DANS LE CONTEXTE DE L'ENTREPRISE .....	97
7.1.	Organis'Or .....	97
7.2.	Utilisation envisagée de la RPM .....	98
7.3.	Intégration .....	98
8.	DISCUSSION ET PERSPECTIVES .....	99
8.1.	Limitations de nos études .....	99
8.2.	Conseils de conception pour dispositifs inclinables .....	99
8.3.	Instabilité du dispositif .....	100
8.4.	Axe Z .....	100
8.5.	Défis techniques .....	101

8.6.	Autres scénarios d'utilisation.....	103
9.	CONCLUSION DU CHAPITRE 4.....	105
	COMBINAISON DE DISPOSITIFS : ESPACE DE CONCEPTION .....	106
1.	INTRODUCTION.....	108
1.1.	Motivation.....	108
1.2.	Structure du chapitre.....	109
2.	APPROCHE DE COMPOSITION : DÉFINITIONS.....	110
3.	DECO : UN ESPACE DE CONCEPTION POUR LA COMPOSITION DE DISPOSITIFS.....	111
3.1.	Axe 1 : arrangement physique .....	111
3.1.1.	Topologie .....	111
3.1.2.	Type de fusion.....	112
3.1.3.	Dynamicité .....	114
3.2.	Axe 2 : manipulation physique .....	114
3.2.1.	Effecteurs corporels.....	115
3.2.2.	Actions physiques .....	116
3.2.3.	Utilisation temporelle.....	116
4.	CLASSIFICATION DES DISPOSITIFS COMPOSÉS.....	118
4.1.	Classification physique et temporelle.....	118
4.2.	Comparaison des dispositifs composés avec DECO .....	118
5.	APPLICATION : CONCEPTION D'UN DISPOSITIF COMPOSÉ .....	122
5.1.	Motivation.....	122
5.2.	Première itération.....	123
5.3.	Deuxième itération : changement du type de fusion.....	124
5.4.	Troisième itération : usage temporel parallèle.....	128
5.5.	Quatrième itération : dynamicité .....	129
5.6.	Conclusion .....	130

6. DISCUSSION ET PERSPECTIVES .....	131
6.1. DECO pour décrire, comparer et générer des solutions .....	131
6.2. Recommandations pour la conception de dispositifs composés .....	131
6.3. Limites et travaux futurs .....	133
7. CONCLUSION DU CHAPITRE 5 .....	134
COMBINAISON DE DISPOSITIFS : APPLICATION À L'INTERACTION AVEC DES TABLEURS EN MOBILITÉ .....	136
1. INTRODUCTION .....	138
1.1. Motivation.....	138
1.2. Terminologie.....	139
1.3. Structure du chapitre .....	139
1.4. Analyse statistique des résultats .....	140
2. ÉTUDE PRÉLIMINAIRE : EXPLORATION DES GESTES.....	141
2.1. Dimension du stacking.....	141
2.2. Étude préliminaire sur le confort des gestes .....	142
2.3. Résultats.....	144
3. PROTOTYPE ET IMPLÉMENTATION .....	145
3.1. Coque conductrice .....	145
3.2. Algorithme de détection.....	147
3.3. Faisabilité industrielle.....	147
4. ÉTUDE 1 : SÉLECTION D'UNE PLAGE DE CELLULES .....	149
4.1. Techniques d'interaction pour la sélection d'une plage de cellules .....	149
4.1.1. Interaction tactile.....	149
4.1.2. Intersection .....	150
4.1.3. Stack-and-Drag.....	151
4.1.4. Finger Stack.....	152
4.2. Description de l'étude .....	154

4.3.	Résultats.....	157
5.	ÉTUDE 2 : TABLEAU CROISÉ DYNAMIQUE .....	159
5.1.	Présentation des tableaux croisés dynamiques .....	159
5.2.	Techniques d'interaction pour la création et la manipulation de tableaux croisés dynamiques.....	160
5.2.1.	Interaction tactile.....	160
5.2.2.	Dual-Screen Touch.....	161
5.2.3.	Stack-and-Touch .....	161
5.2.4.	Stack-and-Translate.....	162
5.3.	Description de l'étude.....	162
5.4.	Résultats.....	165
6.	SCÉNARIO D'UTILISATION .....	167
7.	DISCUSSION ET PERSPECTIVES .....	169
8.	CONCLUSION DU CHAPITRE 6 .....	173
	APPLICATION INDUSTRIELLE .....	174
1.	CONTEXTE .....	176
1.1.	Motivation.....	176
1.2.	Structure du chapitre .....	176
2.	PROJET TREETECH .....	177
2.1.	Contexte.....	177
2.2.	Fonctionnalités de l'application.....	177
2.3.	Scénario d'utilisation .....	179
2.4.	Identification de la problématique .....	179
2.5.	Solutions envisagées .....	180
2.5.1.	Localiser un arbre.....	181
2.5.2.	S'orienter par rapport à l'arbre.....	182
2.5.3.	Télémétrie.....	183

2.5.4.	Synthèse .....	185
2.6.	Description du prototype attendu.....	185
2.7.	Développement du prototype et limitations.....	187
2.8.	Perspectives .....	187
3.	PROJET ATTAL MOBILE .....	189
3.1.	Contexte .....	189
3.2.	Fonctionnalités de l'application.....	189
3.3.	Scénarii d'utilisation .....	190
3.3.1.	Observation d'un arbre.....	190
3.3.2.	Veille d'information sur un ensemble d'arbres.....	191
3.4.	Identification de la problématique .....	191
3.5.	Solutions envisagées .....	191
3.5.1.	Barre de menu .....	192
3.5.2.	Interaction bimanuelle.....	192
3.5.3.	Représentation schématique.....	193
3.5.4.	Accords tactiles .....	194
3.6.	Prototype développé .....	195
3.6.1.	Barre de menu zoomable.....	195
3.6.2.	Barre de menu navigable.....	196
3.6.3.	Triple barre de menu .....	197
3.6.4.	Barre de menu hiérarchique .....	198
3.6.5.	Synthèse .....	199
3.7.	Limitations .....	199
3.8.	Perspectives .....	200
4.	CONCLUSION DU CHAPITRE 7 .....	202
	CONCLUSION .....	204
1.	RÉSUMÉ DES CONTRIBUTIONS .....	206

1.1.	Contribution théorique.....	206
1.2.	Contributions pratiques.....	207
1.3.	Synthèse.....	209
2.	PERSPECTIVES.....	210
2.1.	Perspectives à moyens termes.....	210
2.2.	Perspectives à longs termes.....	211
	ANNEXES.....	216
1.	RÉSULTATS DÉTAILLÉS : LA ROLY-POLY MOUSE.....	218
1.1.	Étude 1 : Amplitude et stabilité des gestes.....	218
1.1.1.	Amplitude des gestes.....	218
1.1.2.	Stabilité des gestes.....	219
1.1.3.	Fatigue.....	220
1.2.	Étude 2 : Pointage 2D.....	220
1.2.1.	Temps de réalisation.....	220
1.2.2.	Taux d'erreur.....	221
1.2.3.	Débit et modélisation.....	221
1.2.4.	Préférence utilisateur et fatigue.....	221
1.3.	Étude 3 : Manipulation 3D.....	222
1.3.1.	Pointage 3D.....	222
1.3.2.	Rotation 3D.....	222
1.3.3.	Préférence utilisateur et utilisabilité.....	222
1.3.4.	Fatigue.....	223
2.	RESULTATS DETAILLES : LE STACKING.....	224
2.1.	ÉTUDE PRÉLIMINAIRE : EXPLORATION DE GESTES.....	224
2.2.	ÉTUDE 1 : SÉLECTION D'UNE PLAGE DE CELLULES.....	224
2.2.1.	Temps de sélection.....	224
2.2.2.	Utilisabilité et préférence utilisateur.....	226

2.3. ÉTUDE 2 : TABLEAUX CROISÉS DYNAMIQUES .....	228
2.3.1. Temps de sélection (étape 1).....	228
2.3.2. Temps d'affectation (étape 2 et 3).....	229
2.3.3. Erreurs .....	230
2.3.4. Utilisabilité et préférence utilisateur .....	231
BIBLIOGRAPHIE .....	234
LISTE DES FIGURES .....	260
LISTE DES TABLES.....	266



# CHAPITRE 1

## INTRODUCTION

---

### Structure du chapitre

---

1. CONTEXTE .....	14
2. OBJECTIFS.....	17
3. STRUCTURE DU MANUSCRIT .....	19

# 1. CONTEXTE

Les travaux présentés ci-après s'inscrivent dans le domaine de l'Interaction Homme-Machine (IHM). Ce domaine a pour but de concevoir, évaluer et mettre en œuvre des systèmes informatiques interactifs pour l'usage humain (<http://www.sigchi.org/>). Cette discipline place l'utilisateur au centre de ses études en visant la meilleure utilisabilité et satisfaction possible de l'utilisateur au moyen de techniques d'interaction. Les solutions proposées par l'IHM peuvent, en autres, se décomposer en deux catégories : les visualisations interactives et les techniques d'interaction. Les visualisations interactives visent à améliorer les visualisations de données afin d'en tirer des informations pertinentes. Les techniques d'interaction permettent de faciliter ou de rendre accessible la manipulation de données. Nos travaux s'inscrivent dans cette seconde catégorie. Les travaux de cette thèse s'inscrivent dans une collaboration entre l'équipe ELIPSE de l'IRIT et l'entreprise Berger-Levrault, acteur majeur du développement de logiciels de gestion pour les services publics.

Ces travaux s'appuient sur trois constats liés à ce contexte.

Dans un premier temps, la politique de numérisation des services publics induit la nécessité de disposer de logiciels adaptés aux métiers et procédures de l'ensemble des établissements publics. Cette politique de digitalisation des procédures administratives publiques, débutée il y a plus de 15 ans sous l'impulsion du gouvernement français et de l'union européenne, vise à simplifier la réalisation de démarches auprès des services de l'état (préfecture, mairies, établissement de santé et scolaires, sécurité sociales, ...) et à faciliter leur gestion administrative (services RH, paie, gestion du patrimoine, ...). Depuis peu, la numérisation du service public est même devenue obligatoire en France. Ainsi, au fil du temps, les administrations publiques sont passées progressivement d'une gestion papier à une gestion informatisée. Or, plusieurs problématiques nuisent à la réalisation de cette transition. D'une part, les législations françaises et européennes en matière de protection des données informatiques, de télépaiement ou de sécurité informatiques évoluent au cours du temps ; or l'impression d'un nouveau document administratif papier respectant ces nouvelles normes est souvent plus rapide que la mise à jour d'un logiciel informatique développé par une entreprise tierce. D'autre part, la densité et la subtilité de ces législations induit une complexité élevée des logiciels administratifs. Concrètement, la numérisation des services publics apporte un nombre conséquent d'avantages. En

premier lieu, l'utilisation d'outils informatiques permet aux citoyens des petites collectivités, ayant donc peu de moyens, de bénéficier de l'accès à des services publics sans avoir à se déplacer dans une ville plus importante (accès numérique aux services). De plus, les nouvelles possibilités de gestion offertes par la numérisation des données ajoutent un ensemble complémentaire de fonctionnalités aux logiciels administratifs. À titre d'exemple, des tâches comme la gestion et le suivi de la santé des arbres d'une grande collectivité, impossibles à faire à l'aide d'un support papier à cause du trop grand nombre d'arbres, sont rendues possibles par l'existence d'outils informatiques facilitant le recueil et le stockage de ces informations. Cependant, la simplicité de recueil et de stockage des données induit une augmentation de leur complexité : il devient simple de stocker beaucoup plus d'informations sur un citoyen ou un élément de patrimoine que ce que permettaient les formulaires papier. Ainsi, les logiciels administratifs contiennent un ensemble de données de plus en plus riche, c.-à-d. contenant un nombre important de champs (ou dimensions). Ces données sont appelées données multidimensionnelles. À titre d'exemple, les données associées à un employé municipal peuvent inclure son emploi du temps sur plusieurs mois, ses compétences, ses chantiers en cours, ses congés, sa paie, son adresse, etc. Ainsi, naviguer au sein de l'espace de données associé à un seul employé et manipuler les valeurs de leurs champs devient une tâche particulièrement complexe. Le passage à l'échelle, c.-à-d. gérer plusieurs employés dans les multiples services d'un établissement public ne fait qu'accroître la complexité.

Le second constat à l'origine de ces travaux de thèse est basé sur le fait qu'en parallèle de l'évolution de la complexité des données contenues dans ces logiciels, les dispositifs permettant d'interagir avec ces données n'ont que très peu évolué. Depuis plusieurs décennies, la souris et le clavier restent la norme dans un environnement fixe. Bien que la souris ait connu quelques améliorations comme l'apparition de la molette ou quelques modifications de forme comme les souris ergonomiques, les possibilités de manipulation offertes par ce dispositif sont restées limitées. Plusieurs travaux scientifiques [22, 26, 86, 87] ont démontré que la manipulation de données multidimensionnelles pouvait être facilitée à l'aide d'un dispositif possédant de nombreuses dimensions manipulables. On appelle ces dimensions des degrés de liberté en entrée. La souris et clavier ne possèdent que peu de degrés de liberté en entrée : Ces dispositifs sont initialement conçus pour une interaction en deux dimensions et demandent des adaptations (métaphore logicielle ou combinaison de touches) pour pouvoir manipuler des données multidimensionnelles. Un

plus grand nombre d'étapes est nécessaire pour réaliser une tâche multidimensionnelle, ce qui rallonge le chemin d'interaction.

Le troisième constat à l'origine de ces travaux est lié à la part de plus en plus importante de l'interaction hors du contexte du bureau dans les établissements publics. À titre d'exemple, les employés municipaux utilisent des dispositifs mobiles pour le recueil de données terrain ou la remontée d'alertes ; les élus les utilisent pour la veille d'actualité ou la communication entre services et le personnel médical pour le suivi des patients. Bien que ces dispositifs soient possédés par une majorité d'utilisateurs et bien qu'ils contiennent un grand ensemble de capteurs (accéléromètres, gyroscopes, capteurs de pression, écran capacitif), en général, seuls quelques gestes tactiles et l'orientation du dispositif mobile (portrait / paysage) sont utilisés : l'utilisation des degrés de libertés supplémentaires induit des problématiques de découvrabilité des gestes possibles et de difficulté d'apprentissage. En conséquence, les degrés de libertés de ces dispositifs ne sont pas toujours suffisants pour la manipulation des données multidimensionnelles contenues dans les logiciels administratifs.

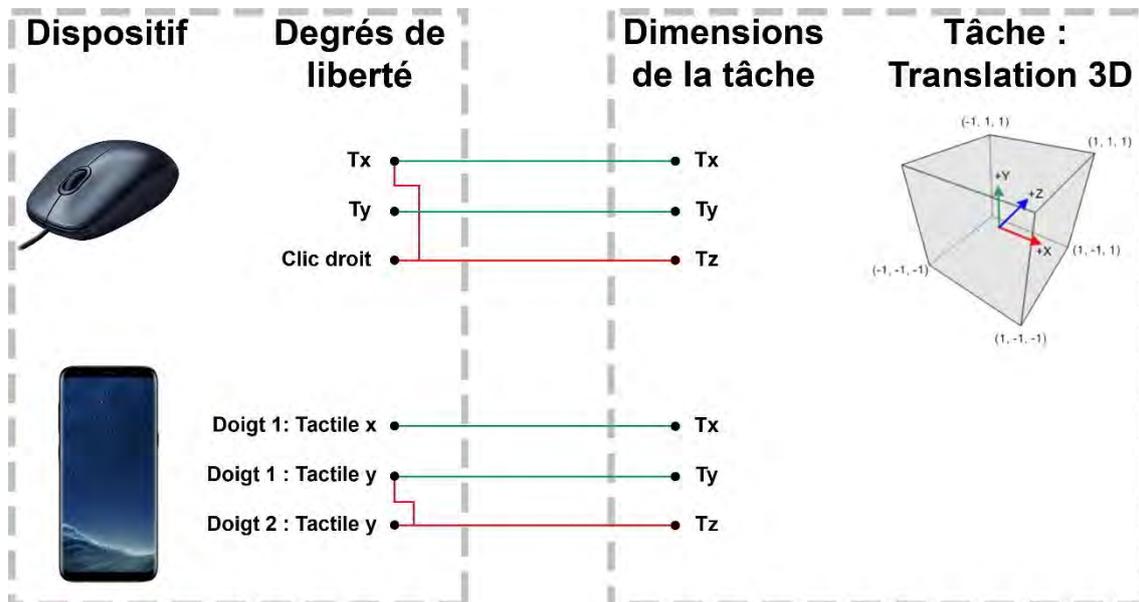


Figure 1. Schéma récapitulatif: les degrés de liberté des dispositifs n'étant pas suffisant pour interagir avec des données multidimensionnelles, une combinaison de ceux-ci est utilisée, complexifiant l'interaction.

## 2. OBJECTIFS

L'objectif de ces travaux de thèse est de contribuer à la fluidification de l'interaction avec des données multidimensionnelles, contenues dans les logiciels du service public, au travers de l'augmentation des degrés de libertés en entrée proposés par les dispositifs fixes et mobiles (Figure 1). La fluidité de l'interaction a été caractérisée dans la littérature par plusieurs critères [13]. Dans notre contexte, nous nous intéressons plus particulièrement à deux de ces critères : la réduction des modes d'interaction (c.-à-d. utilisation d'une touche clavier ou d'un clic souris pour configurer l'interaction) et la limitation des interruptions dues au changement de dispositifs (c.-à-d. alterner entre la souris et un dispositif dédié).

Pour optimiser l'interaction selon ces deux critères, nous explorons plus particulièrement comment accroître le nombre de degrés de libertés du dispositif d'interaction en entrée, pour ainsi réduire l'utilisation de modes et limiter l'utilisation de plusieurs dispositifs dédiés, fluidifiant par conséquent l'interaction.

Nos contributions s'articulent autour de trois axes :

1. La conception, le développement et l'évaluation d'un dispositif à multiples degrés de liberté, la Roly-Poly Mouse ;
2. Le développement d'un cadre de conception de nouveaux dispositifs à partir de la composition physique de dispositifs préexistants: DECO ;
3. La conception, le développement et l'évaluation d'un ensemble de techniques d'interaction pour dispositifs mobiles basées sur le paradigme du stacking.

Tout d'abord, nous proposons un nouveau dispositif à multiples degrés de liberté : la Roly-Poly Mouse (RPM). Ce dispositif, dont la base est arrondie, vise à remplacer la souris traditionnelle. Il possède 6 degrés de libertés (3 translations dont 2 exploitées et 3 rotations). Son facteur de forme permet une utilisation parallèle ou séparée de ses différents degrés de libertés, conservant ainsi les propriétés de la souris traditionnelle, tout en y intégrant des degrés de libertés supplémentaires. Nous avons évalué ses performances et l'avons comparé à d'autres dispositifs pour une tâche nécessitant 6 degrés de libertés (manipulation d'objet 3D).

Cependant, chaque nouveau dispositif reste souvent une solution ad-hoc à un problème donné et nécessite un temps de conception et test élevé. Nous avons donc proposé une

alternative facilitant la conception de dispositifs à multiples degrés de libertés : la composition physique de dispositifs préexistants. La seconde contribution de nos travaux consiste donc en la définition d'un espace de conception se focalisant sur l'aspect physique de la composition de dispositifs : DECO. DECO s'appuie sur deux axes : l'arrangement physique et la manipulation physique. À partir de DECO, nous décrivons les dispositifs composés existants et mettons en évidence les différences entre ceux-ci en termes d'arrangement physique et les conséquences que cet arrangement a sur les manipulations physiques possibles.

Enfin, la troisième contribution de ces travaux consiste en l'utilisation du principe de composition de dispositifs pour proposer une solution fluidifiant l'interaction avec des données multidimensionnelles dans un environnement mobile : le stacking. Cette approche consiste en l'utilisation d'un dispositif complémentaire (dans notre cas, un smartphone) pour étendre les degrés de libertés en entrée et en sortie d'un dispositif support (dans notre cas, une tablette). Pour étendre les degrés de libertés en entrée, l'écran du smartphone peut être utilisé comme surface tactile et le smartphone peut être utilisé comme interacteur tangible sur l'écran de la tablette. Nous avons conçu et évalué plusieurs techniques d'interaction visant à fluidifier l'interaction avec un tableur sur tablette. Au travers d'un ensemble d'études, nous montrons que le stacking permet de sélectionner des plages de cellules et de créer des tableaux croisés dynamiques plus rapidement qu'avec l'interaction tactile.

### 3. STRUCTURE DU MANUSCRIT

La structure de ce manuscrit reflète la démarche scientifique adoptée qui nous a conduits à élaborer plusieurs contributions, entrelaçant théorie et application.

Dans le chapitre 2, nous présentons l'entreprise Berger-Levrault, acteur majeur du développement de logiciel pour administration publique en France. Ensuite, nous décrivons les logiciels de l'entreprise ayant servi de cadre à nos réflexions. Enfin, nous présentons les résultats d'une série d'interviews visant à découvrir les besoins en termes d'interaction d'un groupe d'utilisateurs de l'entreprise : les élus des petites moyennes collectivités.

Dans le chapitre 3, nous passons en revue les travaux existant visant à augmenter les degrés de liberté en entrée des dispositifs en environnement fixe et mobile. Dans un premier temps, nous passons en revue les dispositifs et techniques d'interaction étendant les degrés de libertés en entrée en contexte fixe. Ensuite, nous décrivons les solutions permettant d'étendre les degrés de liberté dans un contexte mobile. Enfin, nous présentons le principe de composition ainsi que les dispositifs existants basés sur ce principe.

Dans le chapitre 4, nous présentons la Roly-Poly Mouse (RPM), un dispositif à multiples degrés de libertés permettant des gestes de translation, rotation et inclinaison. Dans un premier temps, nous décrivons les différents prototypes que nous avons conçus. Dans une pré-étude, nous explorons les différentes prises en main du dispositif et explorons plusieurs variantes de forme. Dans une première étude, nous mesurons la séparabilité des degrés de libertés du dispositif, l'amplitude des mouvements ainsi que la précision des inclinaisons et rotations. Dans une seconde étude, nous explorons les capacités de pointage 2D de la RPM. Dans une troisième étude, nous évaluons une technique d'interaction pour la manipulation d'objet 3D en translation et en rotation, et comparons la RPM à un dispositif commercial, la SpaceMouse.

Le chapitre 5 décrit notre espace de conception DECO. Nous présentons ses deux axes (arrangement physique et manipulation physique) ainsi que ses six propriétés : topologie, type de fusion et dynamique pour l'axe d'arrangement physique ; effecteurs corporels, actions physiques et utilisation temporelle pour l'axe manipulation physique. Ensuite, nous décrivons et comparons plusieurs dispositifs composés existants à l'aide de DECO. Puis, nous présentons la conception d'un nouveau dispositif guidée par notre espace de conception, la RPM2. Enfin, nous évaluons les performances des différentes versions de

notre dispositif pour une tâche de RST (Rotate-Scale-Translate) et comparons ces performances à celles de la souris traditionnelle.

Dans le chapitre 6, nous présentons dans un premier temps le paradigme du stacking d'un smartphone sur une tablette et mettons en lumière ses avantages. Dans un second temps, nous décrivons une pré-étude visant à explorer les différents gestes offerts par le stacking. Puis, nous présentons notre prototype permettant la détection du stacking d'un smartphone sur l'écran d'une tablette. Ce prototype est basé sur l'utilisation d'une coque conductrice et d'un algorithme de reconnaissance de motifs tactiles. Dans une première étude, nous explorons un ensemble de techniques d'interaction basées sur le stacking pour la sélection d'une plage de données dans un tableur sur tablette. Nous comparons ces solutions à l'interaction tactile traditionnelle. Dans une seconde étude, nous comparons un ensemble de techniques d'interaction pour la création et la manipulation d'un tableau croisé dynamique dans un tableur sur tablette.

Dans le chapitre 7, nous présentons deux cas d'application de ces travaux de thèse dans des logiciels existants commercialisés par l'entreprise Berger-Levrault. Dans les deux cas, les propositions élaborées s'inspirent des différentes contributions de cette thèse. Dans un premier temps, nous décrivons le logiciel TreeTech visant à faciliter le diagnostic des arbres dans une collectivité. Puis, nous présentons une technique d'interaction basées sur la combinaison de l'ensemble des capteurs du dispositif afin d'étendre les degrés de libertés de l'interaction tactile traditionnelle. Dans un second temps, nous présentons un second cas d'application de nos travaux : le logiciel Atal mobile. Suite à une présentation de ce logiciel, nous décrivons nos différentes techniques d'interaction basées sur l'utilisation de l'interaction multi-touch, de l'interaction mid-air et du stacking pour faciliter la manipulation (filtrage) d'un grand ensemble de données.

Enfin, nous concluons ce manuscrit en dressant un bilan de nos contributions théoriques et expérimentales autour de la conception de dispositifs et techniques d'interaction à multiples degrés de libertés pour fluidifier l'interaction avec des données multidimensionnelles. Nous développons ensuite un ensemble de perspectives à donner à ces travaux.

# CHAPITRE 2

## CONTEXTE INDUSTRIEL DES TRAVAUX

---

### Structure du chapitre

---

1. INTRODUCTION.....	22
1.1. Contexte.....	22
1.2. Structure du chapitre.....	22
2. DESCRIPTION DE L'ENTREPRISE.....	23
2.1. Introduction.....	23
2.2. Description des logiciels Berger-Levrault.....	25
2.2.1. Ville et Territoire.....	25
2.2.2. Domaine de la Santé et du Social.....	29
2.2.3. Conclusion.....	30
3. ÉTUDE DE CAS : LES ÉLUS.....	31
3.1. Description de l'étude.....	31
3.2. Résultats.....	33
3.2.1. Période de travail.....	33
3.2.2. <i>Environnement et outils de travail</i> .....	33
3.2.3. Description des tâches.....	34
3.2.3.1. Relation citoyen.....	34
3.2.3.2. Veille d'actualité et communication.....	36
3.2.3.3. Gestion des finances.....	38
3.3. Conclusion.....	39
4. CONCLUSION DU CHAPITRE 2.....	40

# 1. INTRODUCTION

## 1.1. CONTEXTE

Comme mentionné dans l'introduction, l'objectif principal de nos travaux vise à proposer de nouvelles techniques d'interaction afin de faciliter l'accès aux données multidimensionnelles dans le contexte particulier des logiciels du service public. Ce contexte découle de l'implication et du co-encadrement de ces travaux par l'entreprise Berger-Levrault, leader national de la production de logiciels et de contenus réglementaires à destination des services publics. Les logiciels commercialisés sont destinés à répondre aux besoins variés des différents services de la fonction publique : la gestion des élections, du patrimoine des collectivités, de la relation élu/citoyen, des ressources humaines, de la facturation, etc. Ces logiciels riches et denses permettent la manipulation de données multidimensionnelles et représentent un formidable contexte d'étude pour nos travaux.

À titre d'exemple, le logiciel ATAL II, qui permet la gestion du patrimoine d'une collectivité, regroupe plusieurs dizaine de milliers d'éléments (c.à.d. espaces verts, mobilier urbain, etc.). Chacun de ces éléments est représenté à l'aide d'un ensemble de caractéristiques qui dépendent des pratiques propres à chaque collectivité. Ces variables peuvent être de type différents tel que des booléens, des valeurs numériques, des champs de texte, des médias divers, etc. De cette grande quantité et hétérogénéité de données multidimensionnelles découle de nombreuses problématiques d'interaction telles que la navigation dans cet espace de données, la recherche d'un élément ou encore la modification d'un champ.

L'ensemble des travaux de recherche présentés dans cette thèse visent à répondre à des problématiques d'interaction rencontrées par les utilisateurs des logiciels proposés par Berger-Levrault.

## 1.2. STRUCTURE DU CHAPITRE

Dans la section 2, nous présentons l'entreprise Berger-Levrault au travers de son histoire et d'une partie des logiciels qu'elle édite. L'entreprise éditant un très grand nombre de logiciels, nous ne présenterons que ceux étant pertinents pour le contexte des travaux de cette thèse. La section 3 décrit une étude de cas que nous avons réalisé auprès de clients de l'entreprise : les élus.

## 2. DESCRIPTION DE L'ENTREPRISE

### 2.1. INTRODUCTION

Berger-Levrault est historiquement une société d'impression de documents administratifs créée entre 1676 et 1681 à Strasbourg (Figure 2). Jusqu'à la fin du XXème siècle, elle sera successivement imprimeur du roi, de l'annuaire militaire de l'armée française, du code Napoléon, de l'Almanach ainsi que des 1ers ouvrages de Charles de Gaulle. Au fil du temps, l'entreprise s'étend et s'installe à Nancy. Aujourd'hui encore, Berger-Levrault continue d'éditer des documents réglementaires et administratifs pour l'état. Cependant, elle a commencé à éditer des logiciels pour l'administration en 1981 et continue d'étendre ce marché à l'aide de croissances externes sur le marché des logiciels administratifs. Avec la cession de l'édition graphique au début du XXIème siècle, l'entreprise sort partiellement des métiers de l'imprimerie et accentue sa présence sur le marché du logiciel de gestion pour les collectivités.

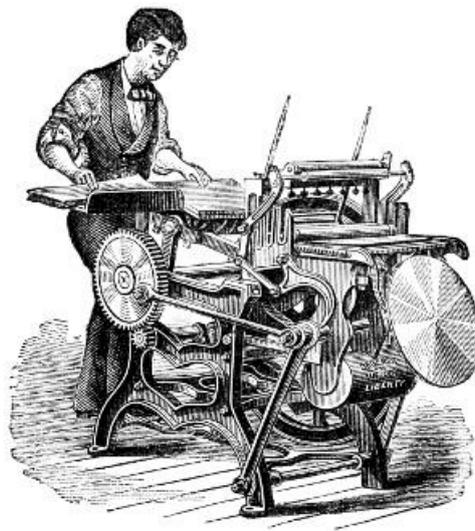


Figure 2. Illustration d'un imprimeur de Berger-Levrault.

En 2002, le groupe Accueil devient actionnaire majoritaire de l'entreprise. Depuis, l'entreprise fait l'acquisition de nombreux acteurs français de l'édition de logiciels tels que Magnus, Sofiac, Sedit-Marianne, DIS, Convergence, Ecolesoft, Progor, Segilog, Aductis, Intuitive, Sigems et InfoPark. À partir de 2013, Berger-Levrault s'installe à l'international avec l'acquisition de sociétés telles que Coba, Exagon et Infosilem au Canada ainsi qu'Aytos, Absis et Tecnogeo en Espagne. En 2016, Berger-Levrault est classée 12ème éditeur de logiciels en France avec un chiffre d'affaire de 110 M€ et plus de 45 000 clients [32]. Aujourd'hui l'entreprise est dirigée par Pierre-Marie Lehucher et

possède des locaux dans de nombreuses villes de France (Paris, Toulouse, Marseille, Nantes, Nancy, Nîmes, Colmar, Strasbourg, Montpellier et Lyon), au Canada (Québec, Laval, Blainville, Montréal) et en Espagne (Barcelone, Madrid, Séville et Valence) représentant une masse salariale de plus de 1600 collaborateurs (Figure 3).



Figure 3. Carte du monde représentant les pays dans lesquels Berger-Levrault possède des agences, des filiales et des clients.

Berger-Levrault édite et commercialise un ensemble de logiciels destinés à trois types de clientèle : l'administration publique d'état, l'administration publique locale et le domaine de la Santé / Social. Concernant l'administration publique d'état, Berger-Levrault édite des produits spécifiques pour la gestion des ministères, des administrations centrales de l'Etat et des services déconcentrés (le sénat, les préfetures, les rectorats, les directions départementales, etc.). Pour l'administration publique locale, Berger-Levrault édite des produits de gestion spécifique pour les collectivités locales (communes, départements, régions), les établissements publics administratifs rattachés aux collectivités locales (offices HLM, etc.) et les établissements publics intercommunaux. Enfin, pour le domaine de la Santé et du Social, Berger-Levrault édite un ensemble de logiciels offrant une solution globale et intégrée dédiée à tous les acteurs de santé publics et privés tel que les CHU, les EHPAD, les services de HAD (Hospitalisation A Domicile), les SSIAD (Services de Soins Infirmiers A Domicile), les SPASAD (Services Polyvalents d'Aide et de Soins à Domicile), et les SAAD (services d'aide à domicile non médicalisés). Une description brève des logiciels édités et commercialisés par l'entreprise est fournie ci-après.

## 2.2. DESCRIPTION DES LOGICIELS BERGER-LEVRAULT

Les gammes de logiciels de l'entreprise (pertinents pour ces travaux de thèse) s'axent autour de deux catégories : « Ville et Territoire » et « Sanitaire et Social ». Ces catégories représentent une partie des besoins logiciels de l'administration publique d'état, locale ou des établissements de santé.

### 2.2.1. VILLE ET TERRITOIRE

Cette catégorie de logiciels regroupe l'ensemble des produits visant à répondre aux besoins métier d'une collectivité. Bien que le client de ces logiciels soit toujours la collectivité, les utilisateurs dépendent du logiciel et peuvent aller de l' élu au citoyen en passant par le personnel administratif ou les agents techniques.

#### Gestion administrative

Berger-Levrault propose un panel large de solutions permettant la gestion administrative d'une collectivité. Pour la gestion des finances, des logiciels tels qu'e.magnus Gestion Financière (GF) ou e.sedit GF permettent de gérer la comptabilité et le budget d'une collectivité. Ces logiciels contiennent l'ensemble des dépenses financières de la collectivité (employés, location de matériel, travaux, etc.), proposant en conséquence une vaste quantité de données à explorer. De plus, chaque entrée dans le logiciel est liée à de nombreux facteurs comme le montant de la dépense, le service, les éventuels devis et documents administratifs auxquels celle-ci est liée, les dates de facturations, etc.

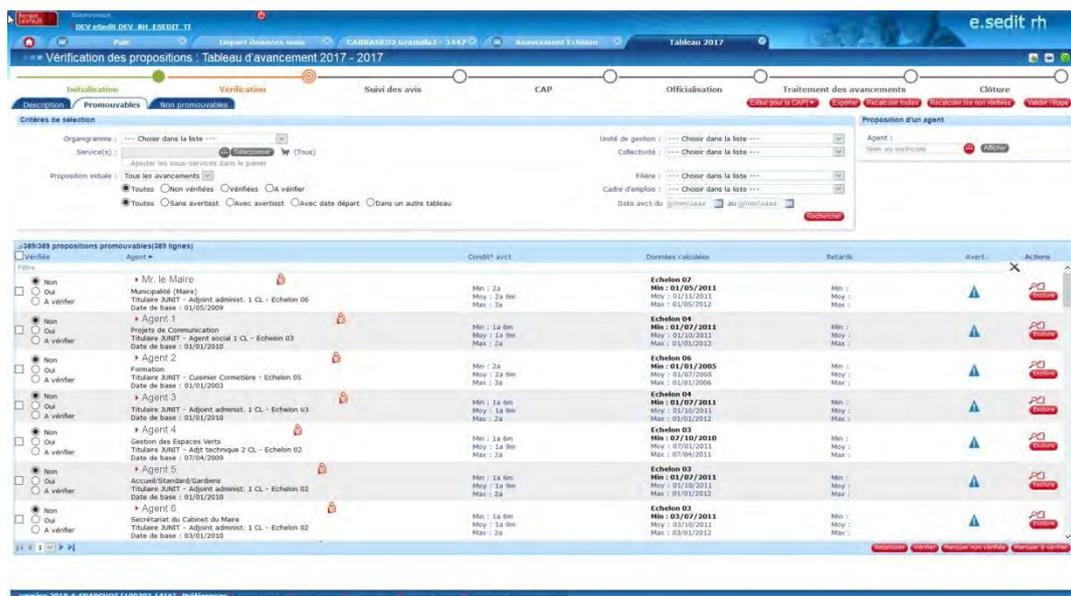


Figure 4. Capture d'écran du logiciel e.sedit RH.

Concernant la gestion des ressources humaines, Berger-Levrault propose les solutions e.magnus RH et e.sedit RH (Figure 4). Ces logiciels permettent une gestion complète de la paie et des ressources humaines associées aux collectivités. Ils regroupent l'ensemble des informations relatives à la situation de chaque agent (nom, prénom, adresse, qualifications, etc.) pour faciliter la saisie et la consultation. Une gestion multi-onglets permet également de travailler en simultané sur plusieurs dossiers agents. De nombreuses éditions et extractions de fichier du personnel sont réalisables. Le module Paie gère tous les cycles de paie : salaires, primes, cotisations, congés et RTT, mutuelle, etc. Il produit les déclarations au format électronique ou papier et intègre la dématérialisation vers les organismes extérieurs. Le module Carrière gère le déroulement de la carrière des agents (avancements d'échelon, avancements de grade, promotion interne, notation, titularisation, médaille du travail).

E.magnus Gestion Relation Citoyen (GRC) permet la gestion de l'administration d'une collectivité et la gestion de la relation citoyen. Cette solution est composée de plusieurs modules permettant la gestion d'actes d'état civil, d'élections, du recensement militaire et l'édition de formulaires administratifs. Elle regroupe les informations des citoyens de la collectivité (nom, adresse, liens de parenté, etc.) ainsi que leurs demandes diverses en faisant le lien avec d'autres produits tels que Post-Office ou e.magnus cimetièrre (pour la gestion des concessions). Cette solution permet aussi de conserver, d'éditer et de transmettre l'ensemble des pièces justificatives liées à la demande d'un citoyen.

Le cabinet numérique est un outil visant à faciliter la communication entre les différents membres du conseil municipal (Figure 5). Il permet la transmission de documents numériques et la communication entre élus et facilite la gestion de l'arbitrage sur l'ensemble des dossiers de la collectivité. Cet outil, disponible au format web, existe aussi sur dispositif mobile au travers d'une application dédiée. Il contient l'ensemble des dossiers en cours et archivés de la collectivité ainsi que tous les documents numériques qui y sont associés.

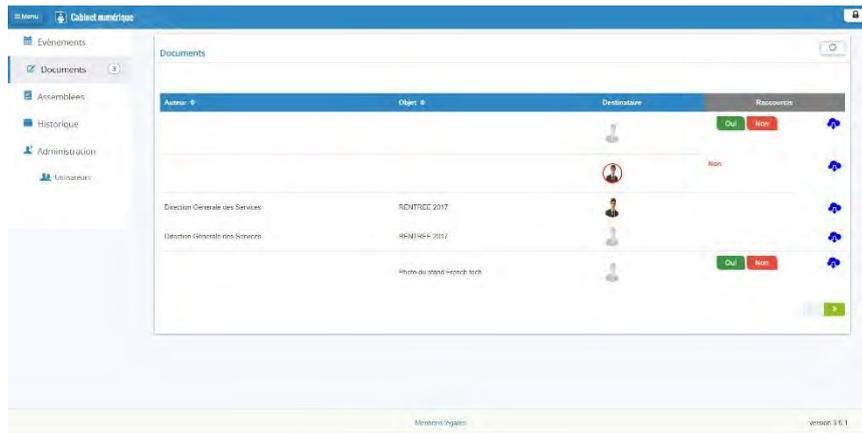


Figure 5. Capture d'écran du logiciel Cabinet Numérique.

## Gestion du territoire

Atal II est une solution Berger-Levrault qui propose de nombreuses fonctionnalités permettant d'assurer en toute facilité la gestion du patrimoine, des activités techniques et services liés (Figure 6). Autour d'un tronc commun, une grande diversité de modules métiers est proposée selon le domaine d'activité concerné : du référencement du patrimoine, aux bâtiments, espaces verts, à la gestion du parc automobiles et carburants, des fluides, etc. L'application permet la gestion des interventions, de la planification, du management, de l'état des stocks et des commandes et des transferts en comptabilité. Le logiciel regroupe par catégorie les différents éléments de patrimoine. Au sein d'une catégorie, les éléments sont représentés par un ensemble de dimensions dont la nature et le nombre sont modifiables. À titre d'exemple, les arbres référencés dans Atal II pour la ville de Lyon, au nombre de 40 000, possèdent chacun 47 dimensions qui peuvent être des nombres, des identifiants, du texte, des booléens ou des valeurs parmi une liste donnée (exemple : type d'arbre, espèce, variété, etc.).

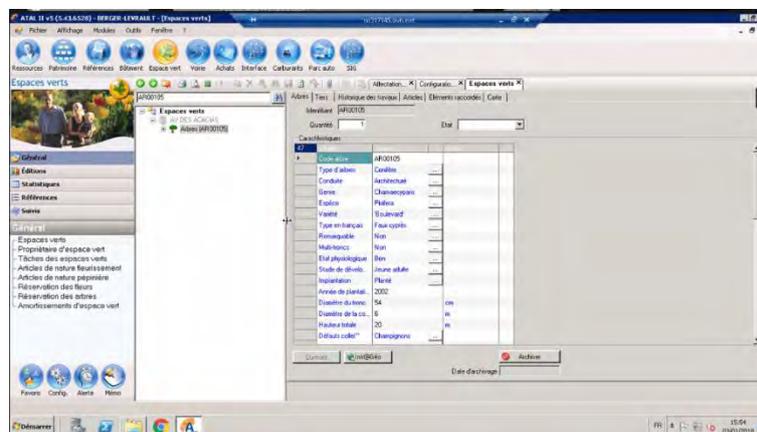


Figure 6. Capture d'écran du logiciel Atal II.

## Gestion des services citoyens

BL.citoyens est une solution permettant un accès aux différents logiciels administratifs de Berger-Levrault. Elle offre un portail web et une application mobile, le portail citoyen, proposant aux citoyens une interface de communication avec la municipalité. Les citoyens sont ainsi informés en temps réel pour faciliter l'accès aux services de la vie quotidienne, participer à la vie citoyenne et réaliser des demandes administratives. Ils possèdent aussi un compte dédié regroupant leurs données personnelles et des documents administratifs (justificatif d'adresse, livret de famille, copie de la carte d'identité, etc.). Le portail web intègre plusieurs modules et notamment un espace de gestion de l'enfance (scolarité, restauration, activités périscolaires, etc.) et un espace de télépaiement. En complément, il permet la réalisation de démarches administratives auprès de la mairie et facilite le suivi des demandes des citoyens pour le personnel administratif.

Commune.IT est un logiciel permettant de faciliter la transmission d'information entre les élus et les citoyens (Figure 7). Disponible en version Web ou via des applications mobiles, il permet à la collectivité d'informer les citoyens à propos d'évènements se déroulant dans la collectivité, d'informations pratiques telles que des zones en travaux ou encore des informations sur les commerces (horaires, promotions) de la collectivité. En complément de ces fonctionnalités, il permet au citoyen d'accéder aux démarches administratives numériques proposées par sa collectivité au travers d'un accès au portail citoyen. Enfin, il permet au citoyen de faire remonter une alerte auprès de la mairie (dégât sur le matériel municipal ou litiges).



Figure 7. Capture d'écran du logiciel Commune.IT.

### 2.2.2. DOMAINE DE LA SANTE ET DU SOCIAL

Cette catégorie de logiciels regroupe l'ensemble des produits visant à répondre aux besoins métier d'un établissement de santé public ou privé. Les clients de ces produits sont les différents services hospitaliers, les cliniques et les centres à vocation sociale (aide à domicile, portage de repas, etc.).

#### Gestion des établissements de santé

BL.Planning (anciennement Organis'Or) permet d'élaborer le planning de l'ensemble du personnel d'un établissement de santé. Il permet de gérer le planning prévisionnel des agents de l'établissement, leurs horaires de roulements et absences selon plusieurs critères (services, compétences, type de roulement, corps de métier, etc.). Ce progiciel intègre l'ensemble des réglementations légales afin de faciliter la gestion du temps de travail des salariés. La visualisation de ces informations et leur modification (p. ex. le remplacement d'un agent absent) se fait à l'aide d'une grille de type tableur représentant les horaires de chaque agent selon son des catégories, code couleurs et codes alphanumériques. Chaque cellule représente une plage horaire d'un agent et peut contenir plusieurs encoches représentant une modification récente, une absence, etc. Pour faciliter le suivi des patients, BL.Résident permet le suivi administratif et la facturation des résidents. Il facilite le partage d'information entre les différents agents et services de santé. Pour le cas spécifique des urgences, le logiciel ResUrgences permet le suivi des patients à travers un dossier patient intégré dans une architecture web. Il inclut de nombreuses fonctionnalités comme une visualisation géographique du parcours du patient dans les services de santé, la planification des soins et gestion des prescriptions ou encore la gestion des justificatifs médicaux.

#### Gestion de la prise en charge à domicile

BL.domicile intègre l'ensemble des processus clés de la prise en charge et des soins à domicile. Il permet la gestion des dossiers des bénéficiaires, la planification des interventions, le suivi du temps de travail, la facturation des prestations et la transmission des informations au travers d'un portail web et d'une application mobile. Il contient l'ensemble des informations de chaque bénéficiaire au travers d'un dossier de liaison (fiche signalétique, adresses, gestion de l'entourage, plan médico-social de prise en charge, etc.). Ce logiciel peut être utilisé en complément de BL.Social, une solution visant à faciliter la gestion du parcours social de l'utilisateur (Figure 8). Disponible en offre SaaS

ou via une application mobile, il permet la gestion des démarches administratives liées aux différents services d'assistance à l'utilisateur (accueil et accompagnement des usagers, inscription et facturation de l'hébergement des personnes âgées en établissement spécialisé, etc.).

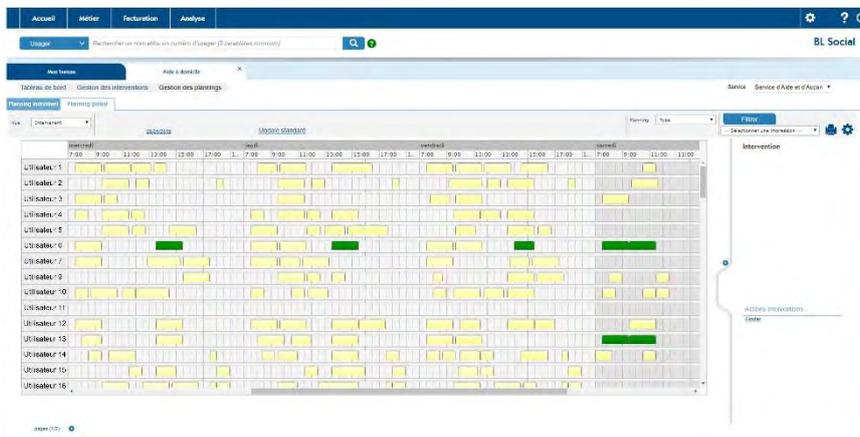


Figure 8. Capture d'écran du logiciel BL.Social.

### 2.2.3. CONCLUSION

Dans la majorité de ces logiciels, la combinaison de la grande quantité de données et de la multidimensionnalité de la donnée (due à la richesse de cette multitude de logiciels) induit une complexité élevée en termes d'interaction. Ces deux facteurs rendent l'exploration et la modification des données fastidieuse pour l'utilisateur : un temps d'apprentissage plus long est nécessaire et la charge cognitive des utilisateurs peut être élevée. Pour améliorer la fluidité de l'interaction, il semble pertinent de proposer de nouveaux dispositifs d'interaction adaptés aux données contenues dans ces logiciels et par conséquent, à multiples degrés de liberté. Dans ce manuscrit, nous présentons deux nouveaux dispositifs à multiples degrés de liberté : la Roly-Poly Mouse (cf. chapitre 4) et la Roly-Poly Mouse 2 (cf. chapitre 5).

Afin de mieux cerner les besoins en termes d'interaction des usagers des logiciels de Berger-Levrault, nous avons choisi de réaliser une étude de cas auprès de clients importants pour l'entreprise : les élus.

### 3. ÉTUDE DE CAS : LES ÉLUS

Les élus sont au contact d'une large gamme de produits (logiciels de gestion administrative, logiciels spécialisés - action sociale, enfance, marchés publics, métiers techniques) dont plusieurs sont particulièrement complexes dû à la quantité importante de réglementations (qui de plus évoluent régulièrement) et de services proposés. L'étude de leur contexte de travail et de leurs besoins permet de mieux comprendre les problématiques sous-jacentes en termes d'interaction et de pouvoir proposer des solutions pertinentes et adaptées.

Nous avons donc réalisé une série d'interviews auprès d'élus de plusieurs collectivités. Ces interviews ont été effectuées au cours de l'année 2016. Les participants à ces interviews ont été recrutés grâce aux salariés de l'entreprise et à l'aide du club MAGEL. Ce club (association loi 1901) regroupe les collectivités utilisatrices des logiciels Berger-Levrault dont l'objectif est de représenter et défendre les intérêts des collectivités publiques auprès de l'entreprise.

#### 3.1. DESCRIPTION DE L'ETUDE

Le but de cette étude est de recenser les outils informatiques manipulés par les élus, leurs habitudes d'interaction et l'environnement dans lesquels ceux-ci travaillent.

##### Tâche et instructions

Les participants ont été invités à remplir un formulaire de consentement éclairé et un questionnaire démographique (tranche d'âge, familiarisation avec les outils informatiques, etc.). Enfin, l'interview elle-même était guidée par un document regroupant un ensemble de questions. Cependant, afin de conserver la forme d'une discussion libre, le document ne servait de support qu'en cas de nécessité ou afin de recentrer la discussion. Les questions posées abordaient trois thèmes distincts : 1) les périodes de travail, 2) les outils de travail et 3) la description des tâches principales (les plus importantes) réalisée par l' élu.

##### Participants et démographie

Ces entretiens ont été réalisés auprès de 8 élus (tous hommes). Leurs collectivités respectives sont composées de 563 à 466 297 habitants. Les conseils municipaux sont composés de 14 à 69 élus (Figure 8).

Les élus interviewés étaient familiarisés avec les ordinateurs de bureau ainsi que les dispositifs mobiles classiques (7.5 sur une échelle de 1 à 10) (smartphone, tablette). Tous les élus utilisent des suites bureautiques (traitement de texte, tableaux et présentations tels que la suite Microsoft Office) et des navigateurs internet (ex : Chrome). Presque tous (6/8) utilisent des logiciels de messagerie (ex : Gmail). Certains utilisent des logiciels administratifs dédiés fournis par Berger-Levrault (4/8, notamment E.GRC).

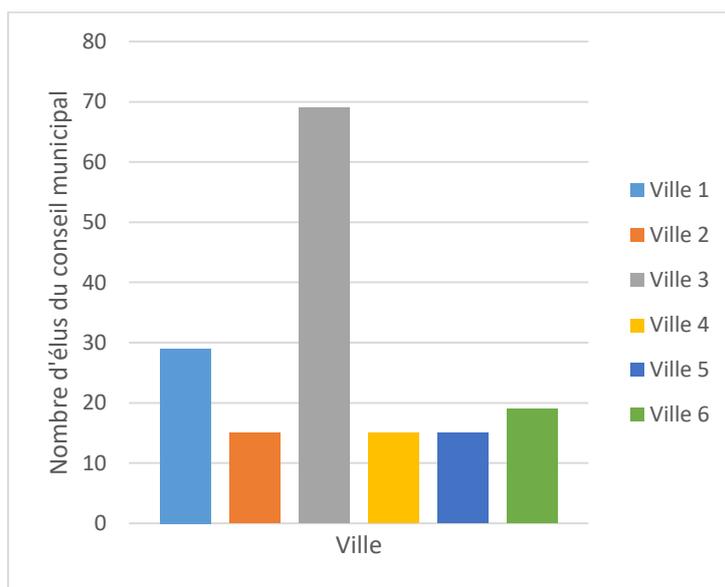


Figure 9. Nombre d'élus du conseil municipal par collectivité.

Les élus interviewés exercent des fonctions différentes au sein de leurs conseils municipaux. Certains sont maires de leur collectivité, d'autres sont adjoints au maire et d'autres sont conseillers municipaux. Un seul sujet interviewé n'exerçait plus la fonction d'élus au moment de l'entretien. Les participants étaient attachés à différents domaines d'activités tels que la gestion des finances et du budget, l'agriculture, la gestion des transports et de l'urbanisation, l'écologie, la relation aux citoyens ou la gestion de la police municipale.

#### Données collectées

En complément du formulaire démographique, un enregistrement audio complet de l'entretien a été collecté. De plus, des remarques informelles manuscrites ont été notées par l'interviewer. L'interview durait en moyenne 66 minutes par participant.

Matériel

Afin d'enregistrer la bande audio de l'interview, nous avons utilisé un smartphone Motorola G3. Les participants étaient assis avec l'interviewer autour d'une table.

## 3.2. RESULTATS

### 3.2.1. PERIODE DE TRAVAIL

La majorité des élus interviewés (7 sur 8) ont un métier à temps plein en parallèle de leur mandat d' élu. En conséquent, ils effectuent les tâches liées à leur fonction d' élu en matinée ou en fin de journée, avant ou après les horaires de travail de leur métier. Dans ces horaires, ils réalisent plusieurs tâches : consultation des e-mails, suivi de l'actualité, mise-à-jour d'un site-web ou encore communication et collecte d'informations via les réseaux sociaux.

Dans ce contexte, plusieurs élus ont reporté réaliser ces tâches sur dispositifs mobiles (smartphone, tablette) dans des environnements variés : transports en communs, domicile, bureau.

Certains élus travaillent aussi plusieurs jours par semaine directement en mairie. Ces jours peuvent être groupés ou séparés dans la semaine et peuvent inclure le week-end. Les tâches réalisées dépendent alors de la fonction de l' élu.

### 3.2.2. ENVIRONNEMENT ET OUTILS DE TRAVAIL

Lors de déplacements, une majorité d' élus utilisent leur smartphone ou tablette. Dans ce cas, les logiciels utilisés sont les navigateurs web (Chrome, Safari), les logiciels de messagerie (Gmail, Yahoo) et les outils de communication téléphoniques (SMS, appels). Aucun élu n'a reporté utiliser plusieurs dispositifs mobiles à la fois.

Lors d'un travail en lieu fixe, qu'il soit réalisé au domicile ou en mairie, c'est un ordinateur fixe ou portable qui est utilisé. Dans ce cas, les logiciels de bureautique sont utilisés (Microsoft Office), les logiciels de messagerie (Gmail, Yahoo), d'accès à internet (Chrome, Safari) et des logiciels dédiés à certaines tâches métier (e.GRC, e.Compta, e.Cimetière, etc.). Seuls trois élus ont signalé travailler avec plusieurs écrans en même temps ou avec plusieurs ordinateurs. D'une manière générale, les élus travaillent avec plusieurs logiciels actifs simultanément, mais avec un seul en plein écran à la fois.

Certains confirment passer régulièrement d'une fenêtre à l'autre afin de comparer des données ou de lire un document et prendre des notes.

### 3.2.3. DESCRIPTION DES TACHES

Dans le cadre de ces interviews, il était demandé aux élus de décrire plus en profondeur les deux ou trois tâches qui leur semblaient les plus pertinentes ou chronophages. Les tâches décrites sont très variées et nous avons choisi de présenter ci-dessous celles qui nous semblent les plus pertinentes pour nos travaux et propices au développement d'une nouvelle technique d'interaction. Ces tâches sont données dans un ordre aléatoire.

#### 3.2.3.1. RELATION CITOYEN

Cette tâche consiste en l'écoute, l'enregistrement et la réponse aux différentes demandes des citoyens. Elle peut être réalisée lors de permanences organisées par le conseil municipal et de manière imprévue lors de déplacements.

Description de la tâche : Permanences

La tâche consiste en l'accueil des citoyens au sein de la mairie ou d'un local dédié afin d'écouter, de conseiller et d'enregistrer les différentes demandes des citoyens. Ces permanences peuvent durer de quelques heures à une journée complète. Généralement, cette tâche incombe à l'ensemble des élus du conseil municipal. Ils peuvent être jusqu'à trois élus à la fois et réalisent un roulement afin que chacun puisse être présent au moins une fois face aux citoyens. Dans ce contexte, l' élu est dans un environnement fixe qui, dans certains cas, peut être équipé d'un ordinateur de bureau. Dans les cas où l' élu peut formuler une réponse directe au citoyen, la demande est traitée directement. Dans le cas contraire, une demande est enregistrée au format papier ou informatique. Cette demande sera transmise à l' élu en charge du secteur d'activité à l'aide d'une boîte aux lettres dédiée (papier) ou d'une boîte mail (informatique) afin qu'il puisse fournir une réponse dans les meilleurs délais. Dans ce cas, la réponse est généralement formulée par courrier papier ou informatique. Dans les petites collectivités, l' élu peut aussi directement se déplacer pour répondre de vive voix au citoyen concerné. La durée de traitement varie selon le domaine d'activité de l' élu recueillant la demande (cette durée peut être plus courte si l' élu est en charge du secteur d'activité concerné par la demande) et selon le type de demande. Dans la majorité des cas, ce traitement peut prendre jusqu'à un mois entre l'enregistrement de la demande et la réception d'une réponse par le citoyen.

Description de la tâche : Rencontre imprévue

L' élu peut également être interpellé, à n'importe quel moment de la journée ou de la semaine, sur un sujet relatif à la commune lors de ses déplacements personnels ou professionnels. Au même titre que pour les permanences, la tâche réalisée par l' élu consiste à écouter, conseiller et enregistrer la demande du citoyen. L'ensemble des élus du conseil municipal participe à cette tâche. Dans ce contexte, l' élu est dans un environnement mobile et ne possède donc pas de dispositif fixe tel qu' un ordinateur de bureau afin de traiter la demande du citoyen. Si la réponse ne peut pas être formulée de manière directe, les élus utilisent leur téléphone portable pour contacter un autre élu ou un personnel administratif (généralement par appel téléphonique) à même de fournir une réponse au citoyen ou pour enregistrer la demande qui sera traitée ultérieurement. Dans ce second cas, la demande est enregistrée à l' aide d' un mémo informatique ou d' un message électronique envoyé à sa propre boîte de messagerie.

Problèmes rencontrés et besoins

Les outils informatiques classiques utilisés par la mairie étant disponibles, aucun problème majeur n' a été soulevé pour le premier scénario relatif aux permanences.

En revanche, lors de rencontres imprévues avec les citoyens, plusieurs problèmes se posent. Premièrement, l'interaction avec les dispositifs mobiles est plus limitée qu' avec un ordinateur fixe du fait de la petite taille de l' écran et du vocabulaire tactile réduit. En conséquence, l' élu peut avoir des difficultés à récupérer les informations nécessaires au traitement de la demande du citoyen à l' aide d' un dispositif mobile. Deuxièmement, l' élu n' ayant accès qu' à son téléphone, les données stockées en mairie ne sont pas accessibles. Dans ce cas, il est généralement plus complexe de répondre au citoyen de manière directe et précise, ce qui peut induire une perte de crédibilité pour l' élu se trouvant dans l' incapacité de répondre à la demande d' un citoyen. Enfin, les élus souhaitant traiter la demande ultérieurement se voient contraints d' envoyer un message électronique à leur propre boîte de messagerie afin d' enregistrer la question du citoyen, ce qui peut induire des problématiques de traçabilité de la demande.

Afin de faciliter la réalisation de cette tâche par les élus, l' accès aux logiciels de mairie en mobilité et la possibilité de faire une remontée d' alerte ou de demande directement à

travers ces logiciels semble être une piste d'amélioration pertinente. Certains logiciels de Berger-Levrault, tels que Commune.IT ou le Cabinet Numérique permettent de répondre partiellement à ce besoin.

#### 3.2.3.2. VEILLE D'ACTUALITE ET COMMUNICATION

Cette tâche, réalisée en majorité par les élus de grandes collectivités, est composée de trois volets. Le premier volet consiste en la découverte et la recherche d'informations d'actualité et à leur retransmission via les réseaux sociaux et site-web. Le second volet consiste en la publication d'articles rédigés par l' élu sur son site-web personnel ou sur celui de la mairie. Le troisième volet consiste en la réflexion collective avec les citoyens sur des sujets de société à l'aide d'une plateforme numérique de communication (de type forum de discussion) ou lors de rencontres citoyennes.

Description de la tâche : Veille d'actualité

Cette tâche consiste en la lecture de l'actualité à l'aide de documents papiers et/ou informatiques. L' élu réalise cette veille afin d'être au plus proche des préoccupations des citoyens et de se tenir informé des événements nationaux et/ou locaux importants. Ces actualités peuvent être relayées par l' élu à l'aide de réseaux sociaux ou d'un site-web. Cette veille est généralement réalisée dans les déplacements (transports en commun) ou au domicile et peut prendre plusieurs heures par jour. Pour réaliser cette tâche, les élus possèdent un smartphone ou une tablette personnelle sur laquelle sont utilisés des logiciels d'accès à internet. Certaines informations, jugées pertinentes par l' élu, peuvent être relayées via les réseaux sociaux de l' élu ou via un site internet personnel.

Description de la tâche : Publication d'articles

Cette tâche consiste en la rédaction suivie de la publication sur internet d'articles de fond sur des sujets d'actualité spécifiques. Les articles sont rédigés sur le temps personnel de l' élu, généralement le soir ou le week-end. Ils peuvent être rédigés sur plusieurs jours voire plusieurs semaines. Les élus utilisent des dispositifs fixes et mobiles afin de réaliser leurs recherches mais utilisent uniquement un ordinateur de bureau afin de rédiger l'article. Dans ce contexte, seul un élu a signalé travailler avec plusieurs dispositifs à la fois et par conséquent, avec plusieurs applications utilisées simultanément. L'objectif de ces articles est double : permettre aux citoyens de mieux s'informer sur les événements locaux et/ou nationaux et permettre aux citoyens de mieux comprendre les positions de l' élu sur certains sujets.

## Description de la tâche : Réflexion collective

L' élu peut être amené à échanger avec les citoyens autour de l'actualité ou de sujets liés à l'activité de la commune. Dans ce contexte, les discussions peuvent se faire à l'aide des réseaux sociaux, sur des forums de discussion ou via des rencontres citoyennes au sein de la collectivité. Lorsqu'elle est réalisée de manière numérique, cette tâche est généralement faite le soir ou le week-end. Les élus ont alors à leur disposition des dispositifs fixes et mobiles pour effectuer des recherches et échanger avec les citoyens. En revanche, lors de rencontres citoyennes, cette tâche est réalisée à des créneaux spécifiques en journée ou en soirée. Dans ce contexte, les élus n'ont à leur disposition que des dispositifs mobiles (smartphone, tablette) utilisés uniquement pour vérifier un point de discussion spécifique ou publier un message sur les réseaux sociaux. Cette tâche a pour objectif de faire avancer la réflexion des citoyens et de l' élu sur des sujets d'actualité. Elle permet aussi de rapprocher l' élu des citoyens et de tisser un lien avec les associations, organisations et entreprises locales.

## Problèmes rencontrés et besoins

Les élus n'ont mentionné aucun problème particulier concernant la veille d'actualité et la rédaction d'articles. Les outils informatiques classiques utilisés par les élus sont suffisamment simples d'utilisation et performants pour réaliser ces tâches.

Cependant, l'utilisation des réseaux sociaux, et plus précisément la centralisation de ceux-ci, semble être un problème majeur. En effet, il existe de nombreux réseaux sociaux disponibles (facebook, twitter, etc.) mais ceux-ci ne sont pas regroupés afin de faciliter leur utilisation. De plus, les nombreux contacts de l' élu au sein de la société ne sont pas directement associés à leurs réseaux sociaux. Il est donc nécessaire pour l' élu de faire un lien entre toutes ces applications et la personne physique associée. Faire le lien entre les différents réseaux sociaux de l' élu, mais aussi avec ceux d'autres acteurs du monde politique, citoyen ou associatif reste une tâche chronophage et particulièrement complexe.

Afin de faciliter la réalisation de cette tâche par les élus, l'accès à un logiciel sur dispositif mobile centralisant l'ensemble des contacts et leurs réseaux sociaux associés semblerait pertinent. Des logiciels tels que FellowUp [110] ou NetVibes [111] permettent d'effectuer cette tâche. Au regard des solutions existantes et du cadre de recherche posé par ces travaux de thèse, nous n'avons pas donné suite à la résolution de cette problématique.

### 3.2.3.3. GESTION DES FINANCES

Cette tâche est définie par la gestion du budget de la collectivité. Elle est liée au calcul des recettes et dépenses de la commune au cours de son exercice annuel. Ces dépenses peuvent être liées au personnel embauché, aux travaux sur la collectivité, aux associations, etc.

Description de la tâche : Gestion des finances

Afin d'administrer les finances de la commune, l' élu est en contact avec tous les services de la collectivité. Il centralise les recettes et les dépenses des différents services et collecte l'ensemble des demandes citoyennes ou administratives. Ce travail est généralement réalisé en semaine au cours de périodes de travail dédiées. Ces sessions peuvent être étalées sur plusieurs créneaux tout au long de l'année, généralement d'une demi-journée chacune. Afin de traiter les demandes de financement, l' élu peut être amené à réaliser un dossier qu'il présentera lors d'un conseil municipal. La gestion des finances est généralement réalisée au sein de la mairie à l'aide d'ordinateurs fixes. Les logiciels utilisés sont très variés : messagerie électronique pour communiquer avec les autres services (Gmail, Yahoo, messagerie dédiée), accès à internet pour la recherche d'informations (Chrome, Internet Explorer), logiciels dédiés à la comptabilité (e.compta) et/ou logiciels de type tableur (QlikView, Excel). Ces outils sont généralement utilisés sur un seul dispositif avec une seule application plein écran à la fois. L'utilisation des logiciels de type tableur permet de générer des représentations graphiques utilisées comme support à la discussion lors de conseils municipaux ou réunions avec des acteurs locaux ou externes.

Problèmes rencontrés et besoins

Les élus signalent la complexité importante des logiciels de comptabilité et de tableurs qui réduisent leur capacité à manipuler correctement les données. En effet, les élus pointent le besoin de pouvoir manipuler des représentations graphiques en temps réel afin d'alimenter les discussions en conseil municipal et d'avoir une meilleure compréhension de celles-ci. De plus, la gestion du budget d'une collectivité est une tâche très importante, en particulier pour les petites collectivités disposant de peu de moyens financiers. Avoir une vision plus précise et complète de la gestion du budget et des dépenses d'une collectivité semble donc être d'une importance élevée. En conséquence, il semble important de simplifier l'utilisation de ces logiciels.

Dans un second temps, il semble pertinent de permettre l'accès aux représentations graphiques générées en mobilité (par exemple à l'aide d'un cloud) afin de simplifier leur utilisation en tant que support à la discussion. En effet, plusieurs élus (7/8) nous ont signalé avoir déjà été interpellés en mobilité sur les données de la collectivité. Or, dans ce contexte, l' élu ne dispose pas des logiciels présents en mairie et contenant l'ensemble des données de la collectivité. En conséquence, il semble pertinent de proposer des outils afin de rendre accessibles les données des logiciels en mairie dans un contexte de mobilité.

### 3.3. CONCLUSION

Cette série d'interview a permis une meilleure compréhension des conditions de travail des élus et de leurs besoins. Cette étape importante de nos travaux a permis de poser un cadre à la réflexion que nous avons menée afin d'améliorer l'utilisation des logiciels manipulés par les élus.

Au regard de ce rapport, il apparaît clairement qu'une des préoccupations des élus réside dans l'accès simplifié, en mobilité, aux données stockées en mairie dans différents logiciels d'administration. Plusieurs logiciels sont concernés : les logiciels de comptabilité, de gestion administrative, de gestion des travaux ou encore des tableurs tels qu'Excel. Cette problématique nous semble particulièrement importante et prometteuse. En conséquence, cette série d'interviews a servi de base à la proposition, la conception et l'évaluation de nouvelles techniques d'interaction pour dispositifs mobiles simplifiant l'accès aux logiciels utilisés par les élus en mairie. Ces solutions sont basées sur l'utilisation combinée de plusieurs dispositifs mobiles : le stacking (cf. chapitre 6).

## 4. CONCLUSION DU CHAPITRE 2

Le contexte de travail, permis par le co-encadrement de l'entreprise Berger-Levrault, nous a offert un vaste ensemble de logiciels manipulant des données multidimensionnelles et en grande quantité. Ces logiciels ont 1) servi de base pour améliorer notre compréhension des problématiques d'interaction liées à de tels environnements et 2) servi de base pour le développement de techniques d'interaction fluides adaptées à un contexte d'utilisation réel.

Dans un premier temps, nous avons présenté une partie des logiciels de l'entreprise. Nous avons mis en évidence la grande quantité de données contenue dans ces logiciels, leur hétérogénéité et leur multidimensionnalité. Cette première phase exploratoire nous a permis de mettre en évidence le besoin d'un nouveau dispositif d'interaction adapté à ces données : la Roly-Poly Mouse (cf. chapitre 4). Afin de faciliter le déploiement de ce dispositif et de réduire son coût de fabrication, nous avons proposé une version alternative de ce dispositif, la Roly-Poly Mouse 2 basée sur la combinaison d'une souris traditionnelle et de la Roly-Poly Mouse (cf. chapitre 5).

Puis, afin de mieux comprendre les besoins d'un groupe d'utilisateurs de ces logiciels, nous avons mené une série d'interview auprès d'élus (pour certains, clients de l'entreprise). Ces interviews nous ont permis de mettre en lumière l'environnement de travail des élus au travers des dispositifs et logiciels qu'ils utilisent ainsi que les tâches les plus fréquemment réalisés par ceux-ci. Au sein de ces différentes tâches, nous avons identifié des problématiques d'interaction liées à la multidimensionnalité des données manipulées ce qui a structuré nos idées et nos propositions de solutions interactives avancées pour dispositif mobile. Dans ce contexte, nous avons proposé un ensemble de techniques d'interaction pour dispositifs mobiles basées sur le paradigme du stacking (cf. chapitre 6).

# CHAPITRE 3

## ÉTAT DE L'ART : INTERACTION AVEC DES DONNÉES MULTIDIMENSIONNELLES

---

### Structure du chapitre

---

1. INTRODUCTION .....	42
2. DISPOSITIFS À MULTIPLES DEGRÉS DE LIBERTÉS ADAPTÉS AUX ENVIRONNEMENTS FIXES .....	44
2.1. Dispositifs isotoniques .....	45
2.2. Dispositifs isométriques .....	47
2.3. Dispositifs élastiques .....	48
2.4. Dispositifs hybrides .....	49
2.5. Autres approches .....	49
3. DISPOSITIFS ET TECHNIQUES D'INTERACTION MOBILES À MULTIPLES DEGRÉS DE LIBERTÉS .....	51
3.1. Multi-touch .....	51
3.2. Stylet .....	53
3.3. Interaction en l'air .....	55
3.4. Interaction tangible avec un dispositif mobile .....	57
3.5. Dispositif mobile augmenté .....	58
4. CONCEPTION DE DISPOSITIF À MULTIPLES DEGRÉS DE LIBERTÉS PAR COMPOSITION .....	60
4.1. Outils de composition en IHM .....	60
4.2. Approches analytiques .....	61
5. SYNTHÈSE .....	66

# 1. INTRODUCTION

Comme nous l'avons souligné dans l'introduction de ce manuscrit, nous proposons d'étudier l'apport des dispositifs à multiples degrés de liberté pour manipuler des données multidimensionnelles. En effet, les dispositifs d'entrée classiques (clavier et souris en environnement fixe et écran tactile en environnement mobile) offrent un nombre de dimensions suffisant pour interagir efficacement avec la majorité des logiciels. Ces modes d'interaction ont l'avantage d'être utilisés très fréquemment, ce qui nous permet d'en avoir une grande maîtrise. Cependant, leurs degrés de liberté ne sont pas toujours adaptés aux données sous-jacentes. À titre d'exemple, la manipulation d'un objet 3D nécessite 6 dimensions (3 rotations et 3 translations) alors que la souris ou un pavé tactile n'en offrent que deux. Pour compenser cette limitation, des modes peuvent être utilisés : zone de l'interface dédiée à un mouvement spécifique, maintien du bouton droit de la souris, etc. De plus, la granularité et la précision de ces degrés de liberté ne sont pas toujours adaptées aux données que l'on souhaite manipuler. Par exemple, la manipulation d'une échelle de temps à granularité multiple (jours, mois, années, etc.) nécessite plusieurs éléments d'interface graphique (tels que des boutons de défilement ou définissant la granularité voulue) pour compenser les degrés de libertés non adaptés de la souris ou de l'interaction tactile.

Le problème de ces solutions est qu'elles rallongent une tâche qui serait idéalement réalisée en une seule manipulation. Par exemple, au lieu de manipuler les trois dimensions en translation d'un objet 3D, il est nécessaire de manipuler successivement plusieurs plans 2D sélectionnés à l'aide d'un clic (p. ex. mouvement selon le plan xy, puis xz). Bien que la tâche soit toujours réalisable, elle nécessite plus d'étapes intermédiaires, ce qui entraîne un coût temporel supplémentaire.

Plusieurs travaux [22, 26, 86, 87] ont souligné le besoin de dispositifs en entrée adaptés aux dimensions de la donnée manipulée. En effet, Jacob et al. [86, 87] ont montré qu'un dispositif est plus performant si le nombre de degrés de libertés qu'il possède est équivalent au nombre de dimensions des données à manipuler pour réaliser une tâche donnée (p. ex. un espace en 3D, les différents niveaux d'une échelle de temps).

De nombreuses approches proposent de répondre à cette problématique afin de fluidifier l'interaction avec des données multidimensionnelles. Ces approches dépendent principalement du contexte dans lequel elles peuvent être utilisées. Ainsi, dans un

environnement fixe, l'utilisation de dispositifs augmentés peut être envisagée. Dans un contexte mobile, l'utilisation de techniques d'interaction tactile ou de gestes en l'air pourra être privilégiée. Cette distinction entre les solutions adaptées à des environnements fixes et mobiles nous semble être un critère particulièrement important. De plus, ces deux environnements correspondent aux contextes d'application industriels identifiés :

- de nombreux logiciels de l'entreprise tels que BL.planning, e.sedit RH ou Atal II sont des logiciels développés pour environnements fixes
- nous avons identifié un besoin d'interaction en environnement mobile suite à une série d'entretiens avec des clients de l'entreprise Berger-Levrault.

Nous avons donc considéré séparément ces deux types de solutions dans la présentation des dispositifs et techniques d'interaction à multiples degrés de liberté.

Indépendamment de ces deux axes, la mise au point d'un nouveau dispositif peut paraître simple mais reste couteuse, notamment en temps afin d'explorer les multiples aspects du dispositif généré. Ce processus ne peut en plus s'appuyer que sur une approche ad hoc. Différentes approches existent à l'inverse pour assister la création d'un nouveau dispositif à partir de la combinaison (physique ou temporelle) de dispositifs et/ou de briques existantes. Nous abordons donc également ce troisième axe en vue de dresser une synthèse des solutions de la littérature pour concevoir un dispositif à multiple degrés de liberté.

En résumé, dans un premier temps nous présenterons les dispositifs et techniques d'interaction à multiples degrés de libertés dédiés aux environnements fixes (section 2). Puis, nous décrivons les dispositifs et techniques d'interaction basés sur l'utilisation de dispositifs mobiles (section 3). Pour finir, nous présenterons les supports à une approche de composition pour concevoir un dispositif à multiples degrés de libertés (section 4).

## 2. DISPOSITIFS À MULTIPLES DEGRÉS DE LIBERTÉS ADAPTÉS AUX ENVIRONNEMENTS FIXES

Dans cette section nous décrivons les dispositifs conçus pour répondre aux besoins d'interaction avec des données multidimensionnelles dans des environnements fixes. Ces dispositifs peuvent être classés selon de nombreux critères (forme, contexte d'usage, nombre de degrés de liberté, etc.). Nous avons choisi de les classer selon une caractéristique bien établie dans la littérature : le degré de résistance qu'ils offrent à l'utilisateur. Cette propriété est particulièrement importante car elle permet de distinguer les utilisations futures d'un dispositif et ainsi d'anticiper les tâches pour lesquelles il sera probablement le plus pertinent [206]. En conséquence, nous séparons les dispositifs existants selon quatre catégories : les dispositifs isotoniques, isométriques, élastiques et hybrides.

Un dispositif isotonique, aussi appelé dispositif de déplacement, détecte le mouvement. Il offre une résistance au mouvement nulle ou constante (frottements). Un exemple très connu de ce type de dispositif est la souris.

Un dispositif isométrique, aussi appelé dispositif de pression ou de force, offre une résistance infinie à une force qui lui est appliquée. Ce type de dispositif ne peut généralement pas être déplacé. La DesktopBat [175] est un exemple de ce type de dispositif.

Un dispositif élastique est un cas particulier parmi les dispositifs isométriques : c'est un dispositif dont la résistance à une force appliquée augmente avec le déplacement ou la distance à la position de stabilité. La SpaceMouse [173] est un exemple de dispositif élastique.

Les dispositifs hybrides combinent les propriétés des dispositifs isotoniques, isométriques et/ou élastiques. La résistance appliquée correspond à l'une de ces trois catégories de manière distincte (CAT [65]) ou dans le prolongement l'une de l'autre (RubberEdge [28]).

Nous présentons les dispositifs existants selon ces quatre catégories dans les sections suivantes. Nous proposons également une cinquième section décrivant les approches n'étant pas basées sur un dispositif (objets tangibles, interaction en l'air).

## 2.1. DISPOSITIFS ISOTONIQUES

A l'instar de la souris traditionnelle, de nombreux travaux ont conçu des dispositifs isotoniques [7, 14, 49, 78]. Une partie de ces solutions proposent de modifier la forme de la souris afin de permettre la réalisation de rotations et/ou d'inclinaisons. La Rockin'Mouse [7] est une souris augmentée ayant une base arrondie (Figure 10 - Gauche). Cette particularité lui permet d'être inclinée dans deux directions (Gauche-Droite et Avant-Arrière) ajoutant ainsi deux degrés de libertés. Afin de détecter les mouvements (translations et inclinaisons) de la Rockin'Mouse, une tablette Wacom [189] est utilisée. La VideoMouse [78] est une souris augmentée basée sur le même principe : arrondir la base de la souris pour permettre des inclinaisons (Figure 10 - Droite). Basée sur l'utilisation d'une caméra embarquée et d'un tapis de souris spécifique, elle ajoute quatre degrés de liberté supplémentaires à la souris traditionnelle : inclinaison Gauche-Droite, inclinaison Avant-Arrière, Rotation Gauche-Droite et translation selon l'axe Z. Bien que la base de ces deux dispositifs soit arrondie, une partie plate est conservée au niveau du pointeur laser/caméra afin de préserver la stabilité du dispositif. Ce facteur de forme possède deux limitations principales : premièrement, la partie arrondie de la base est restreinte, ce qui limite l'amplitude des inclinaisons disponibles (+/- 20° pour la VideoMouse) et deuxièmement, les différents degrés de libertés ajoutés par ces dispositifs ne peuvent pas être utilisés en simultané [7, 78].



Figure 10. La Rockin'Mouse (Gauche, tiré de [7]) et la VideoMouse (Droite, tiré de [78]).

D'autres solutions basées sur l'ajout de capteurs à la souris ont été proposées. Cechanowicz et al. ont ajouté des capteurs de pression à une souris traditionnelle. Ces capteurs permettent la sélection de commandes ou la configuration d'une valeur de paramètre (p. ex. la valeur d'un zoom). Plusieurs approches [112] proposent l'ajout d'un second capteur de position (laser ou boule de souris) afin de détecter les rotations Gauche-Droite du dispositif. Almeida et al. utilisent ce degré de liberté supplémentaire pour faciliter la manipulation de fenêtres dans un environnement 3D.

Le PHANToM [134] est un dispositif à 6 degrés de libertés. Il est composé d'un stylet placé au bout d'un bras articulé relié à un socle (Figure 11). La position et l'orientation du stylet par rapport au socle sont calculées et transmises afin de manipuler un objet en 3D. De plus, à l'aide de moteurs situés sur le bras articulé, il est possible d'obtenir un retour de force au niveau du stylet. Cette propriété a été très utilisée afin de simuler le contact avec un objet virtuel [47], des collisions [89], ou encore les caractéristiques d'un objet virtuel telles que la rigidité [81] ou le poids [161].



*Figure 11. Exemple de dispositif à multiple degrés de liberté: le PHANToM.*

Afin d'augmenter les degrés de libertés en entrée, plusieurs souris augmentées sont basées sur la combinaison d'un clavier et d'une souris [136, 144]. Pietrzak et al proposent d'intégrer des touches du clavier (les modifieurs de type CTRL ou CMD) sur le bord de la souris traditionnelle. L'objectif est de faciliter l'accès à ces boutons normalement disposés sur le clavier afin de fluidifier la tâche de pointage. Une autre approche consiste à combiner un pavé ou une surface tactile avec une souris. C'est notamment le cas de la Magic Mouse d'Apple ou encore de la PadMouse [8]. La PadMouse intègre une surface tactile sur sa partie supérieure. En plus de permettre le remplacement des clics gauche et droits et de la molette, cette surface tactile peut être utilisée pour activer jusqu'à 32 commandes à l'aide d'une technique proche des Marking Menus [99, 100] et de la main non-dominante.

En complément de l'augmentation des degrés de libertés en entrée, certains dispositifs permettent aussi d'étendre les degrés de libertés en sortie. C'est notamment le cas de la LensMouse (Figure 12 - Gauche). Ce dispositif est composé d'un écran tactile incliné placé sur la partie supérieure d'une souris. De manière similaire à la Padmouse, l'écran tactile présent sur la LensMouse permet de réaliser des gestes de clics gauche et droit. En

compléments de ces gestes, l'écran tactile est utilisé pour afficher des palettes ou menus, réduisant ainsi les trajets du curseur à l'écran et libérant de l'espace d'affichage pour la zone d'interaction principale (p. ex. zone de dessin, de traitement de texte, ...). Une autre approche consiste à combiner un ballon et une souris traditionnelle [94]. L'Inflatable Mouse (Figure 12 - Droite) est équipée d'un capteur de pression dans son ballon interne. Une pression sur celui-ci peut être détectée et servir à l'interaction (p. ex. action de zoom/dézoom). En complément le niveau de gonflage du ballon peut être utilisé comme feedback pour transmettre de l'information à un utilisateur. Kim et al. [94] proposent d'étendre son usage de souris traditionnelle pour en faire un vecteur de transmission des émotions (en entrée à l'aide de la pression et en sortie en percevant le gonflement/dégonflement de la souris).

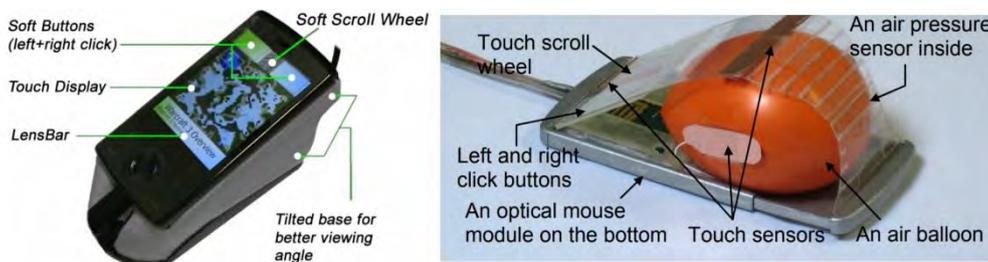


Figure 12. La LensMouse (Gauche, tiré de [204]) et l'Inflatable Mouse (Droite, tiré de [94]).

## 2.2. DISPOSITIFS ISOMETRIQUES

La SpaceBall est un dispositif à 6 DDL [193] permettant la manipulation en translation et rotation d'un objet 3D. Ce dispositif est composé d'une base fixe et lestée sur laquelle sont disposés des boutons (clic gauche, droit, etc.) et d'une sphère résistante au mouvement. L'application d'une force sur cette sphère permet de déplacer ou tourner un objet 3D. Les caractéristiques inhérentes à ce dispositif le rendent particulièrement intéressant pour la manipulation 3D mais difficilement utilisable pour l'interaction avec des interfaces 2D, très présentes dans la majorité des logiciels (bureautique, CAO, etc.). Il convient donc d'utiliser ce dispositif en complément d'une souris traditionnelle, ce qui nuit à la fluidité de l'interaction.

La Desktop Bat [175], une souris à 5 DDL, est basée sur l'approche opposée : attacher un dôme sur la partie supérieure d'une souris. Ce dôme permet la réalisation de rotations en 3D et possède 5 boutons. Bien que des résultats expérimentaux montrent que ce dispositif soit particulièrement performant pour réaliser des tâches en 3D (navigation, sélection,

manipulation), ce dispositif n'a jamais été comparé à d'autres dispositifs d'interaction à multiples degrés de libertés. Il est donc complexe d'en évaluer précisément les bénéfices.



Figure 13. La SpaceBall 4000 FLX

### 2.3. DISPOSITIFS ELASTIQUES

La SpaceMouse est un dispositif à 6 DDL [173] permettant la manipulation en translation et rotation d'un objet 3D. Ce dispositif possède une base lestée, lui permettant de rester stable, sur laquelle est disposé un joystick attaché à un ensemble de ressorts. Celui-ci peut être translaté (3D) et/ou incliné (3D) et se remet automatiquement en position de repos lorsqu'il est relâché. Cependant, la faible amplitude de mouvement et la petite taille du joystick induisent une difficulté à séparer les degrés de libertés lors de la manipulation [59] : utilisé en main non-dominante, la difficulté de manipulation induit une baisse de performance, ce qui la rend moins efficace qu'une souris traditionnelle pour manipuler un objet 3D [141]. En revanche, utilisée en main dominante, une étude montre que ses performances dépassent celles de la souris [59].

L'EGG (Elastic General purpose Grip) est un dispositif élastique à 6 degrés de libertés [206]. Il est composé d'une quasi-sphère en forme d'œuf placée au centre d'un cadre cubique. L'œuf central est maintenu en suspension au centre du cube à l'aide de 8 élastiques accrochés à chaque coin du cube. Les manipulations (translations et rotations) de l'œuf dans l'espace sont associées à la manipulation d'un curseur virtuel. Bien que ce dispositif offre une manipulation très différente des dispositifs isométriques, il souffre de la même limitation : il est nécessaire de l'utiliser en complément d'une souris traditionnelle car il n'est pas adapté à la manipulation d'interfaces 2D, ce qui nuit à la fluidité de l'interaction.

## 2.4. DISPOSITIFS HYBRIDES

En parallèle du développement de dispositifs intégralement isométriques, des dispositifs hybrides ont été proposés [53, 65, 66]. La CAT [65] est un dispositif à 6 degrés de liberté qui combine des propriétés isotoniques et isométriques. Ainsi, les rotations 3D peuvent être réalisées de manière isotonique et les translations 3D de manière isométrique.

La GlobeFish [53] est basée sur l'ajout d'une trackball dans un cadre élastique. Cette configuration permet de réaliser des rotations isotoniques et des translations isométriques ou élastiques. La GlobeMouse, basée sur un principe similaire, est composée d'une trackball placée sur une coque entourant une SpaceMouse. Dans ce contexte, les translations sont isométriques et les rotations isotoniques. Cependant, dans ce second dispositif, un changement de prise en main est nécessaire pour changer de mode (rotations / translations).

Plus proche d'une souris traditionnelle, la Lexip 3D [106] ou la Wing [195] sont des souris augmentées d'un ensemble de joysticks permettant, notamment, la réalisation d'inclinaisons et de rotations élastiques. Cependant, afin de conserver les propriétés de la souris traditionnelle (isotonie, amplitude de gestes, prise en main), les amplitudes d'inclinaison et de rotation sont faibles et la combinaison de ces manipulations avec les translations 2D restent difficiles à réaliser.

RubberEdge [28] est un dispositif élastique composé d'un pavé tactile entouré d'un cercle en plastique. La manipulation du cercle plastique permet de limiter le « clutching » en permettant de contrôler la vitesse de défilement du curseur plutôt que sa position (associée au pavé tactile placé au centre du plastique). Cependant, si ce dispositif étend les capacités de la fonction de transfert reliant les manipulations du dispositif et son impact sur un objet virtuel (en l'occurrence, un curseur), il n'offre pas de degrés de libertés supplémentaires.

## 2.5. AUTRES APPROCHES

Plusieurs approches proposées dans la littérature ne sont pas basées sur l'utilisation d'un dispositif isotonique, isométrique, élastique ou hybride. Ces approches peuvent être basées, par exemple, sur l'interaction en l'air [95, 102], l'utilisation d'objets tangibles [188] ou sur des techniques d'interaction spécifiques [16, 55]. Ces approches offrent un nombre important de degrés de liberté, ce qui fluidifie l'interaction avec des données

multidimensionnelles. Cependant elles souffrent de limitations qui les rendent peu attrayantes pour un contexte d'interaction en environnement fixe. L'interaction en l'air peut induire une fatigue importante lors d'un usage prolongé. L'utilisation d'un objet tangible, bien que très proche de l'interaction avec un dispositif traditionnel, induit des problématiques de "découvrabilité" des gestes. Afin de réduire l'impact de cette limitation sur l'interaction, une forme suggérant les manipulations possibles peut être utilisée. Enfin, les approches basées sur le découpage d'un pavé tactile afin d'étendre l'interaction ajoutent des degrés de libertés en réduisant l'amplitude des mouvements dans les degrés de libertés préexistants, ce qui peut nuire à la précision de l'utilisateur.

En résumé, de nombreux dispositifs à multiples degrés de liberté existent pour un environnement fixe. Ils permettent de faciliter l'interaction en entrée en proposant des dimensions adaptées à celles des données qu'ils permettent de manipuler. Bien que ces approches soient principalement ad hoc, elles mettent en lumière l'intérêt de tels dispositifs pour fluidifier l'interaction avec des données multidimensionnelles dans un environnement fixe. Cependant, les dispositifs à multiples degrés de liberté résultants se heurtent à plusieurs problématiques : dans le cas où le dispositif vise à **remplacer la souris**, l'amplitude et le nombre de degrés de liberté supplémentaires reste faible ; dans le cas où le dispositif vise à être utilisé en **complément de la souris**, il induit un changement de dispositif nuisant à la fluidité de l'interaction. Une solution consiste en la conception d'un dispositif tout-en-un offrant de multiples degrés de liberté et permettant d'éviter le surcoût occasionné par le changement de dispositif.

Au-delà des solutions adaptées aux environnements fixes, le contexte industriel de ces travaux nous a permis d'identifier un second environnement d'interaction : l'environnement mobile. En effet, une partie des clients de l'entreprise, les élus de petites et moyenne collectivités, effectue des déplacements hors de leur environnement de travail traditionnel. Dans ce contexte, l'utilisation de dispositifs mobiles pour accéder aux données multidimensionnelles semble adaptée. Nous présentons donc dans la section suivante les dispositifs et techniques d'interaction à multiple degrés de liberté adaptées aux dispositifs mobiles.

### 3. DISPOSITIFS ET TECHNIQUES D'INTERACTION MOBILES À MULTIPLES DEGRÉS DE LIBERTÉS

Dans cette section nous décrivons les techniques d'interaction adaptées aux dispositifs mobiles visant à étendre leurs degrés de liberté en entrée. Nous présentons dans un premier temps les techniques d'interaction multi-touch ou basées sur un stylet. Puis, nous décrivons des travaux proposant l'interaction autour d'un dispositif mobile (interaction en l'air) ou avec celui-ci (interaction tangible). Enfin, nous présentons l'approche du « back-of-device » pour étendre les degrés de liberté en entrée des dispositifs mobiles.

#### 3.1. MULTI-TOUCH

L'interaction multi-touch est une solution très commune pour étendre l'interaction tactile sur dispositif mobile. Cette solution, étudiée depuis plusieurs décennies [23], a été démontrée comme particulièrement intéressante par de nombreuses études [24, 118]. Elle est notamment utilisée pour la réalisation de gestes de zoom ou de défilement dans la plupart des applications mobiles. Wu et al. (Figure 14) ont mis en lumière l'importance de la création de gestes multi-touch et ont proposé un ensemble de conseils afin de faciliter leur conception [198]. Wobbrock et al. ont aussi proposé une solution pour la création de gestes multi-touch au travers de l'élicitation par des utilisateurs [196]. Ces approches, basées sur l'utilisation de gestes multi-touch, ont pour objectif principal de permettre ou faciliter l'accès à de multiples commandes.

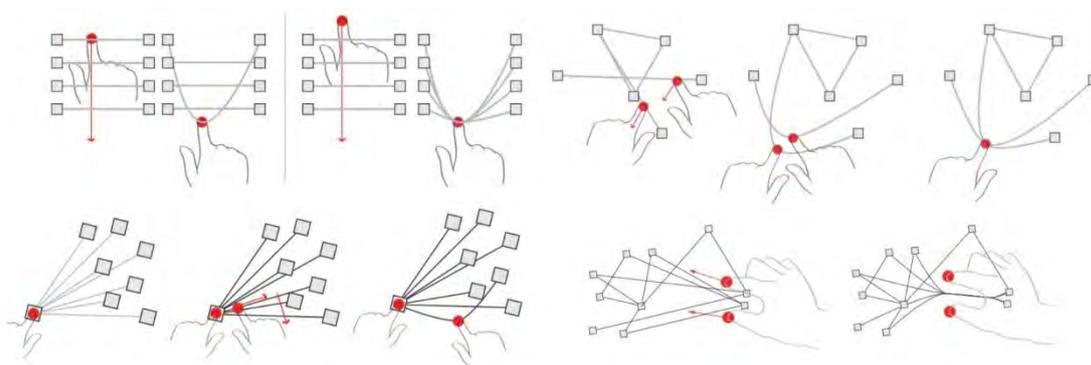


Figure 14. Ensemble de gestes tactiles pour la manipulation d'un graphe (tiré de [162]).

Plusieurs propriétés de l'interaction multi-touch ont été étudiées : la combinaison de doigts pour former des accords [9, 182, 190], l'utilisation de touchers consécutifs formant une mélodie [76, 77], l'interaction bimanuelle tactile [60, 191], l'utilisation de la force d'appui [34, 117, 145], l'orientation du doigt lors d'un toucher [150] ou l'utilisation de

gestes spécifiques [67,87,115,153]. Dans les paragraphes suivants, nous présentons un exemple de chacune de ces catégories.

Banovic et al. (Figure 15 - Gauche) proposent d'utiliser l'interaction multi-touch au travers d'un pie menu afin de faciliter la sélection de commandes [9]. Les commandes disponibles peuvent être activées de manière unimanuelle via deux touchers successifs ou simultanément via un accord.

Heo et al. (Figure 15 - Droite) identifient un nouveau geste tactile : les touchers consécutifs [76]. Ce geste correspond à un double-tap dont le second point de contact est éloigné du premier. Cette technique d'interaction permet notamment d'activer un pie menu ou de réaliser des gestes de zoom et de rotation sur une carte à l'aide d'une seule main, évitant ainsi les difficultés de manipulations liées au multi-touch unimanuel sur dispositif mobile. Ils proposent aussi l'utilisation de multiples touchers consécutifs représentant une forme géométrique afin d'activer des commandes. Cette approche permet d'étendre les degrés de liberté offerts par l'interaction multi-touch en ajoutant la notion de séquence de touchers. Cependant, cette approche nécessite une phase de mémorisation et de découverte des gestes importante.

Wagner et al. envisagent l'utilisation du pouce de la main non dominante comme outil de configuration pour l'interaction tactile [191]. Cette interaction bimanuelle permet de faciliter la navigation dans un document, d'activer des commandes lors de l'édition de texte (p. ex. touche majuscule) ou lors de la navigation sur une carte (p. ex. activer un zoom). Cette solution permet d'étendre les degrés de liberté de l'interaction tactile en ajoutant l'utilisation de la main non-dominante. Cependant, l'espace d'affichage et la disponibilité des deux mains nécessaires pour réaliser cette interaction la rend difficilement applicable à tous les dispositifs mobiles : l'écran d'un smartphone semble être trop petit et la position d'une smartwatch sur le poignet empêche l'utilisation de cette solution.

Corsten et al. proposent d'utiliser la pression du toucher afin de configurer une valeur numérique dans un formulaire [34]. Cette solution permet d'utiliser un degré de liberté supplémentaire, libérant ainsi l'espace d'affichage normalement utilisé pour afficher une liste déroulante.

Roudaut et al. utilisent des micro-gestes de rotation et inclinaison du pouce afin d'activer des commandes [150]. Cette solution permet d'ajouter à l'interaction tactile une

granularité supplémentaire. Leurs travaux montrent que 16 micro-gestes tels que des swipes, des rotations ou des drags peuvent être automatiquement reconnus. Cependant, ces solutions reposent sur des modalités d'interaction complémentaires de l'interaction tactile qui ne sont pas directement découvrables par l'utilisateur. Une phase d'apprentissage importante peut donc être nécessaire et l'affichage d'un feedforward adapté pour guider l'utilisateur semble primordial.

Enfin, Huang et al. proposent de décomposer l'action de toucher selon la zone du doigt en contact avec l'écran tactile [82]. Cette solution permet d'étendre le nombre de commandes accessible à l'utilisateur et ne nécessite que peu de temps d'apprentissage mais elle ne permet que de subdiviser les degrés de liberté existants sans en apporter de supplémentaires.

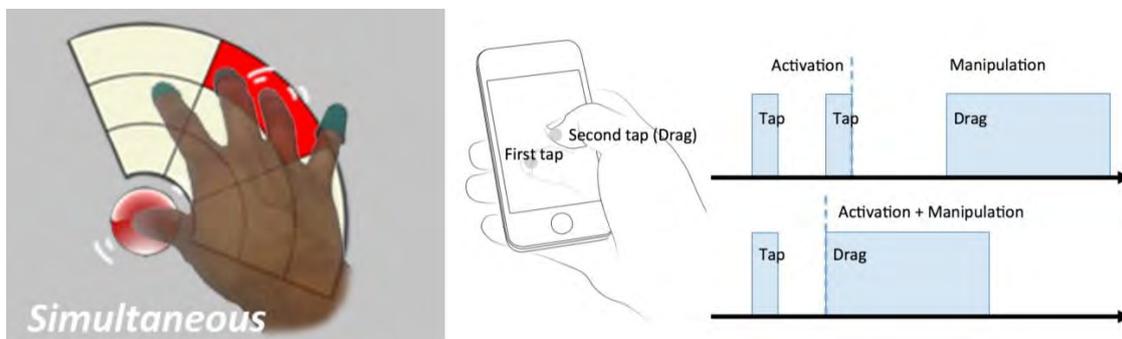


Figure 15. Sélection d'un élément de pie-menu en multi-touch (Gauche, tiré de [9]) et principe de fonctionnement des touches consécutifs (Droite, tiré de [76]).

### 3.2. STYLET

Une solution alternative à l'utilisation de l'interaction multi-touch consiste à tirer parti du stylet. Hinckley et al. [79] ont exploré les avantages de l'interaction multimodale combinant un stylet avec l'interaction tactile dans la conception de nouvelles interfaces. L'interaction combinant stylet et toucher (nommée Pen & Touch) a été proposée dans de nombreux travaux visant à fluidifier l'analyse de données [20, 37, 103, 133, 142, 192, 205] ou l'exploration et la manipulation de représentations graphiques [11, 45, 52, 90, 154, 155, 162, 186]. Bien que cette approche nécessite la possession d'un stylet tactile différenciable d'un toucher classique par le dispositif mobile, elle permet d'étendre les degrés de l'interaction tactile en apportant un nouveau type de toucher : l'appui du stylet.

Frisch et al. étudient l'apport de l'interaction Stylet & Touch pour l'édition de diagrammes [52]. Ils proposent d'utiliser le stylet pour réaliser des commandes de copier-coller, de dessin libre, pour tracer des liens entre éléments du diagramme ou pour

sélectionner des éléments à l'aide d'un geste de lasso. Dans ce cas, le stylet est utilisé comme un doigt supplémentaire ayant une précision plus élevée. Cette solution d'interaction permet de résoudre le problème du fat-finger, fluidifiant ainsi l'interaction avec les données. Cependant, le stylet n'était pas différencié d'un toucher classique, cette solution n'apporte pas de degré de liberté supplémentaire.

Pfeuffer et al. [133] ont proposé plusieurs solutions basées sur l'utilisation du pouce et d'un stylet afin de fluidifier l'interaction avec un tableur sur tablette (Figure 16). Cette solution permet d'éditer une cellule, de copier-coller des données, de naviguer dans une feuille de calcul ou de changer le format/la couleur d'une cellule. À titre d'exemple, il est possible d'ajuster la valeur numérique contenue dans une cellule à l'aide d'un geste de glisser réalisé avec le stylet. TouchPivot [90] propose l'utilisation d'un stylet en combinaison avec l'interaction tactile afin de faciliter la création et la manipulation de représentations graphiques pour les utilisateurs novices au travers de l'usage d'un tableau croisé dynamique. Leur solution permet la création de visualisations de données pertinentes plus rapidement et simplement. Ces deux solutions permettent d'étendre les degrés de liberté de l'interaction tactile en y ajoutant des gestes basés sur le stylet, ce qui fluidifie l'interaction avec les données. Cependant, ces approches nécessitent un dispositif dédié, le stylet, qui n'est pas fourni avec tous les dispositifs mobiles (peu de stylets sont fournis avec les smartphones et le smartwatches). De plus, l'interaction bimanuelle que ces solutions nécessitent n'est pas toujours possible en contexte mobile et notamment lors de l'utilisation d'une smartwatch.

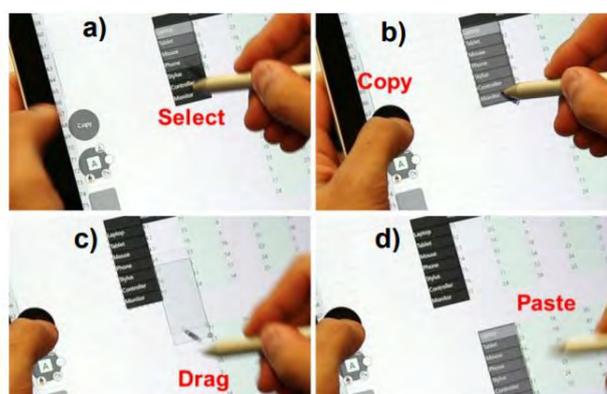


Figure 16. Techniques d'interaction combinant l'utilisation du tactile et du stylet pour manipuler un tableur sur tablette (tiré de [133]).

Une autre ligne de recherche consiste à utiliser d'autres paramètres du stylet comme son inclinaison (Figure 17 - Gauche) et sa rotation [17, 178] ou encore la force d'appui [143]. À titre d'exemple, le « Tilt Menu » (Figure 17 - Droite) est une technique d'interaction

permettant de choisir un élément de pie menu, parmi huit éléments, à l'aide de l'inclinaison du stylet. Ramos et al. présentent des éléments d'interfaces à multiples niveaux sélectionnables grâce à la pression appliquée sur une surface tactile à l'aide d'un stylet [143]. Cependant, ces solutions basées sur l'utilisation de modalités d'interaction supplémentaires peuvent être difficilement découvrables par l'utilisateur et une phase de familiarisation avec celles-ci peut être nécessaire. De plus, l'utilisation de l'inclinaison du stylet peut être inconfortable dans le cas où le dispositif mobile ne repose pas sur une surface plane (tablette prise en main ou inclinée à l'aide d'un support).

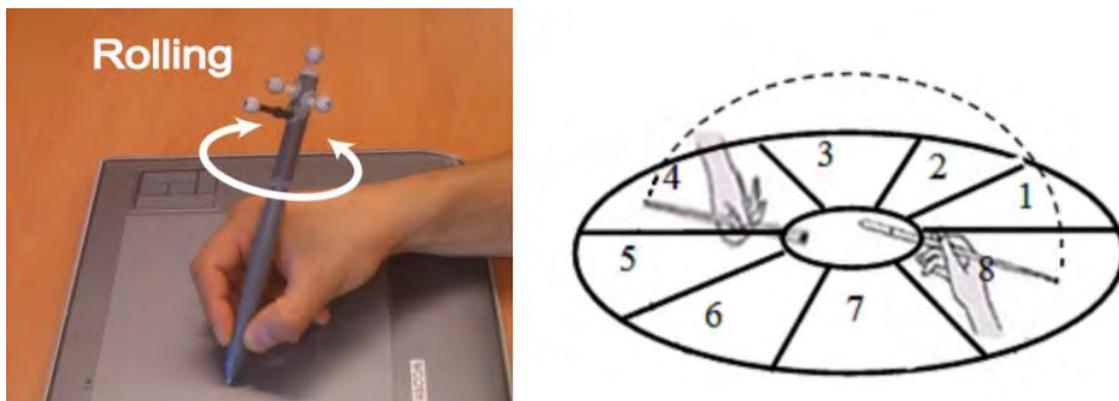


Figure 17. Utilisation de la rotation (Gauche, tiré de [17]) et de l'inclinaison du stylet (Droite, tiré de [178]).

### 3.3. INTERACTION EN L'AIR

Cette approche permet d'étendre les degrés de libertés en entrée des dispositifs mobiles, limités par la dimension réduite de l'écran. De nombreux travaux ont proposé d'utiliser l'interaction en l'air à côté [21, 56, 73, 121], au-dessus [51, 74, 97], derrière [40, 194] ou autour [15, 21, 64, 69, 91, 93, 98, 200, 208] d'un dispositif mobile. Cette solution permet d'étendre les degrés de liberté de l'interaction en tirant profit de l'espace environnant le dispositif mobile. Cependant, l'utilisation de gestes en l'air peut être fatigante et difficile à découvrir pour un utilisateur novice. Enfin, la détection de gestes en l'air environnant un dispositif mobile reste complexe et nécessite des capteurs coûteux ou non disponibles sur les dispositifs mobiles actuels et donc difficilement applicables dans des environnements écologiques. Dans les paragraphes suivants, nous présentons des exemples illustrant l'utilisation des multiples zones d'interaction possibles autour du dispositif mobile.

Dans Ad-binning (Figure 18 - Gauche), Hasan et al. proposent d'interagir autour d'un smartphone pour décharger l'espace d'affichage et permettre la sélection d'éléments hors de l'écran du smartphone [73]. De plus, ils identifient un ensemble de propriétés de

l'interaction à côté du smartphone telles que le nombre d'éléments discriminables et la taille des différentes zones autour du smartphone. Cependant, cette solution est basée sur l'utilisation d'un système de caméras infrarouges complexe et nécessite d'équiper le doigt de l'utilisateur afin de le détecter. Bien que cette solution permette de fluidifier l'interaction, elle est donc difficilement utilisable dans un environnement mobile.



Figure 18. Utilisation de l'espace autour (Gauche, tiré de [73]) ou au-dessus (Droite, tiré de [74]) d'un dispositif mobile.

Thumbs-up (Figure 18 - Droite) est une interaction au-dessus de l'écran du smartphone visant à étendre les degrés de libertés de l'interaction tactile [74]. L'objectif est de rendre accessibles les zones de l'écran inatteignables avec le pouce en les déplaçant dans l'espace au-dessus du smartphone. Cette technique d'interaction permet aussi de mettre en surbrillance les éléments sous le pouce avant leur sélection, réduisant ainsi les problématiques d'occultation dues au fat finger. Cependant, cette solution peut nécessiter un temps d'adaptation important pour l'utilisateur. En effet, le pouce étant souvent utilisé pour l'interaction tactile avec un dispositif mobile, des interactions involontaires avec les données peuvent survenir suite à la détection du pouce au-dessus du dispositif mobile.

Bergé et al. envisagent l'utilisation de l'espace autour du smartphone pour faciliter la manipulation d'objets 3D virtuels [15]. Le smartphone, tenu en main non-dominante, représente l'objet virtuel à manipuler. Les rotations de la main dominante autour du smartphone permettent d'effectuer des rotations identiques dans l'environnement 3D : l'utilisation de l'espace autour du smartphone permet une manipulation directe de l'objet 3D. Cette solution offre une manipulation directe de la donnée ce qui permet de fluidifier l'interaction avec l'objet 3D. Cependant, la réalisation de ces gestes en l'air sur une longue période peut être fatigante pour l'utilisateur.

Afin de détecter les gestes autour du dispositif mobile, plusieurs approches existent. La première approche consiste en l'utilisation d'un système de caméras infrarouges [15, 73, 74] ou optiques [56, 64]. À titre d'exemple, Grubert et al. (Figure 19 - Gauche) combinent

l'utilisation d'une caméra de smartphone avec les reflets présents dans les lunettes de l'utilisateur afin de détecter la position des doigts autour du smartphone. Cette solution, bien que moins précise qu'un système de caméras infrarouges, permet de réduire le coût et la complexité du système de détection des interactions autour du smartphone. Une seconde solution repose dans l'utilisation de capteurs infrarouges embarqués dans un dispositif mobile [21, 97]. Butler et al. (Figure 19 - Droite) envisagent l'utilisation d'une rangée de capteurs infrarouges placés le long des bords gauche et droit d'un smartphone. Ces capteurs permettent la détection de l'interaction tactile autour du smartphone à faible distance.

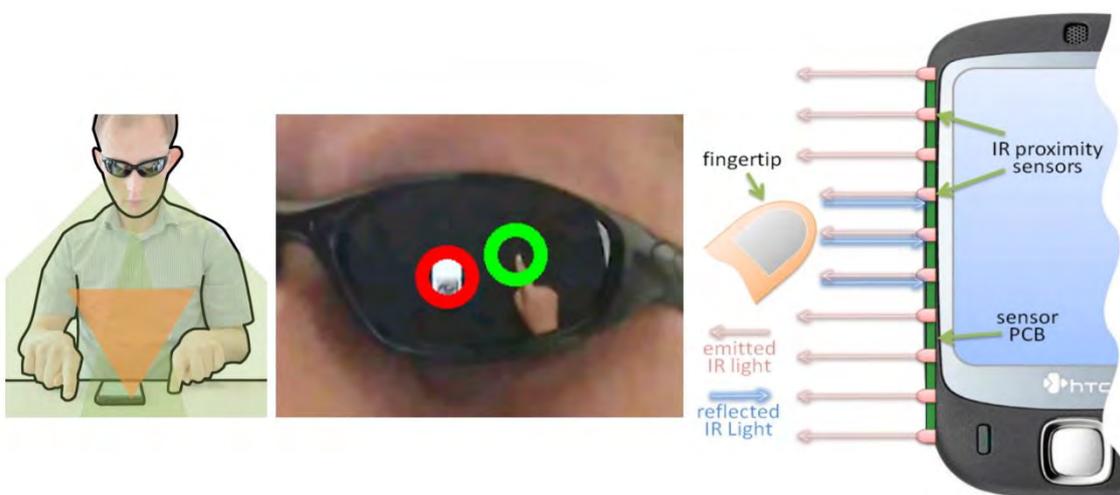


Figure 19. Utilisation des reflets (Gauche, tiré de [64]) ou de capteurs infra-rouges (Droite, tiré de [21]) pour la détection de l'interaction autour d'un dispositif mobile.

### 3.4. INTERACTION TANGIBLE AVEC UN DISPOSITIF MOBILE

L'interaction tangible avec un dispositif mobile consiste en la manipulation tangible directe de celui-ci [4, 29, 39, 61, 70, 84, 108, 122, 125, 146, 151, 152, 159, 187]. Dans ce contexte, aucun interacteur tangible supplémentaire n'est utilisé. Les travaux proposant ce type de manipulation envisagent son utilisation pour manipuler des objets 3D [15], la navigation dans des menus [5, 70, 122, 146], dans les applications [151]. À titre d'exemple, Bergé et al. (Figure 20 - Gauche) proposent la manipulation directe d'un objet virtuel à travers la manipulation physique d'un smartphone. Ainsi, les translations et rotations physiques du smartphone s'appliquent directement sur l'objet virtuel sélectionné. Afin d'éviter la détection de gestes involontaires comme faisant parti de l'interaction, un mode de manipulation activable via un toucher sur l'écran du smartphone permet de délimiter le début et la fin de la manipulation 3D. Baglioni et al. (Figure 20 - Droite) envisagent l'utilisation de l'inclinaison du dispositif afin d'étendre les degrés de

libertés de l'interaction tactile. Ces inclinaisons permettent d'activer des commandes (p. ex. démarrer une piste sonore, copier-coller des données) ou de naviguer dans un ensemble d'écrans ou de fenêtres ouvertes. D'après les auteurs, l'utilisation des inclinaisons peut être combinée à l'interaction tactile sans interférer avec la performance de ces interactions.

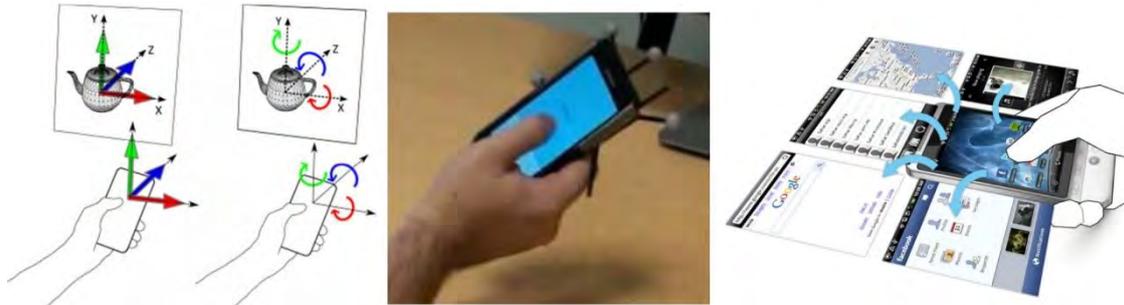


Figure 20. Manipulation tangible d'un dispositif mobile pour la manipulation 3D (Gauche, tiré de [15]) et pour la navigation dans un ensemble d'applications (Droite, tiré de [5]).

### 3.5. DISPOSITIF MOBILE AUGMENTE

Dans le but d'augmenter les degrés de libertés en entrée, certains travaux envisagent d'ajouter des capteurs sur un dispositif mobile existant. Dans ce contexte, une majorité de solution proposent l'utilisation de la surface arrière du dispositif [40, 75, 109, 165, 170, 176, 203]. En complément des degrés de libertés supplémentaires ajoutés, l'utilisation de la surface arrière d'un dispositif mobile permet de résoudre le problème du « fat finger » [201] : les doigts de l'utilisateur ne cachent aucune information visible par l'utilisateur. De Luca et al. (Figure 21 - Gauche) proposent d'utiliser la surface arrière d'un smartphone afin d'authentifier un utilisateur. Pour ce faire, une succession de formes, basées sur le tracé de lignes droites à l'aide de l'index, doivent être tracées sur la surface arrière du smartphone. Afin de détecter ces gestes, De Luca et al. ont conçu un prototype basé sur la combinaison de deux smartphones placés dos à dos. Dans un autre registre, Scott et al. (Figure 21 - Droite) proposent d'intégrer les touches d'un clavier physique à l'arrière d'une tablette afin de faciliter la tâche de saisie de texte. Cette approche permet d'éviter le problème du « fat finger » sur l'écran et permet de libérer l'espace d'affichage normalement utilisé pour le clavier virtuel.



Figure 21. Utilisation de l'arrière du dispositif pour l'authentification (Gauche, tiré de [40]) ou la saisie de texte (Droite, tiré de [165]).

En résumé, de nombreux travaux ont proposé d'étendre les degrés de liberté des dispositifs mobiles afin de fluidifier l'interaction avec les données qu'ils contiennent. Ces solutions peuvent être basées sur l'interaction multi-touch, l'utilisation d'un stylet, l'utilisation de l'espace environnant ou l'ajout de capteurs sur un dispositif mobile. Cependant, ces approches souffrent de plusieurs limitations : elles peuvent être difficilement découvrables, nécessitent un système de détection coûteux et complexe ou impliquent une altération physique du dispositif mobile. Une solution alternative permettant de résoudre ces limitations consiste en l'utilisation combinée de plusieurs dispositifs mobiles : aucun nouveau degré de liberté n'est ajouté, ce qui réduit le temps de découvrabilité et d'apprentissage des gestes, aucun système de détection complexe n'est nécessaire et aucune altération des dispositifs mobiles n'est requise. Nous présentons les approches de conception pour ce type d'interaction ci-après.

## 4. CONCEPTION DE DISPOSITIF À MULTIPLES DEGRÉS DE LIBERTÉS PAR COMPOSITION

La conception d'un nouveau dispositif de manière ad-hoc peut être couteuse en temps du fait de l'exploration et de l'évaluation des nombreux paramètres du nouveau dispositif. En revanche, l'approche de composition vise à augmenter les degrés de libertés en entrée ou en sortie des dispositifs existants au travers de leur utilisation conjointe. Cette composition peut être physique (fusion de plusieurs dispositifs et/ou capteurs) ou temporelle (utilisation simultanée de plusieurs dispositifs et/ou capteurs). Dans cette section nous décrivons en premier lieu les outils de composition existants dans la littérature. En second lieu, nous présentons les approches analytiques décrivant le processus de composition. Enfin, nous présentons plusieurs dispositifs composés mobiles existants.

### 4.1. OUTILS DE COMPOSITION EN IHM

L'idée de composition peut être trouvée dans plusieurs boîtes à outils en IHM, depuis des plates-formes matérielles telles que Arduino ou Phidgets [172] jusqu'à des IDE tels que OpenInterface ou PureData [10, 33, 42, 167].

Les boîtes à outils axées sur l'approche matérielle permettent d'assembler et de combiner différents capteurs, tels que des capteurs de pression, des boutons poussoirs, des détecteurs de lumière, etc., produisant un nouveau dispositif basé sur une fusion de capteurs. Phidgets [172] par exemple, proposent plusieurs capteurs qui peuvent être facilement branchés pour créer un dispositif. Arduino [171] (Figure 22) propose également un ensemble d'actionneurs et de capteurs qui peuvent être combinés pour produire des dispositifs interactifs. Les boîtes à outils de composition axées sur une approche matérielle prennent en charge la composition à un niveau bas, c'est-à-dire à la granularité du capteur, mais n'offrent aucune aide pour guider le processus de conception.



Figure 22. Ensemble de capteurs arduino (pression, infra-rouge, lumière, etc.).

D'autres approches ont développé le concept de composition au niveau du logiciel. La plateforme OpenInterface [167] (Figure 23) permet d'assembler des composants abstraits et de lier leurs entrées et sorties pour développer une interaction multimodale. ContextToolkit [33] est un autre exemple utilisant des composants logiciels qui encapsulent plusieurs types de capteurs. Il est alors possible de combiner ces composants pour développer une technique d'interaction. La plupart de ces boîtes à outils logicielles s'appuient sur des modèles analytiques pour faciliter l'identification et la structuration des composants logiciels. Cependant, ils n'aident pas à déterminer comment concevoir la combinaison physique des dispositifs.

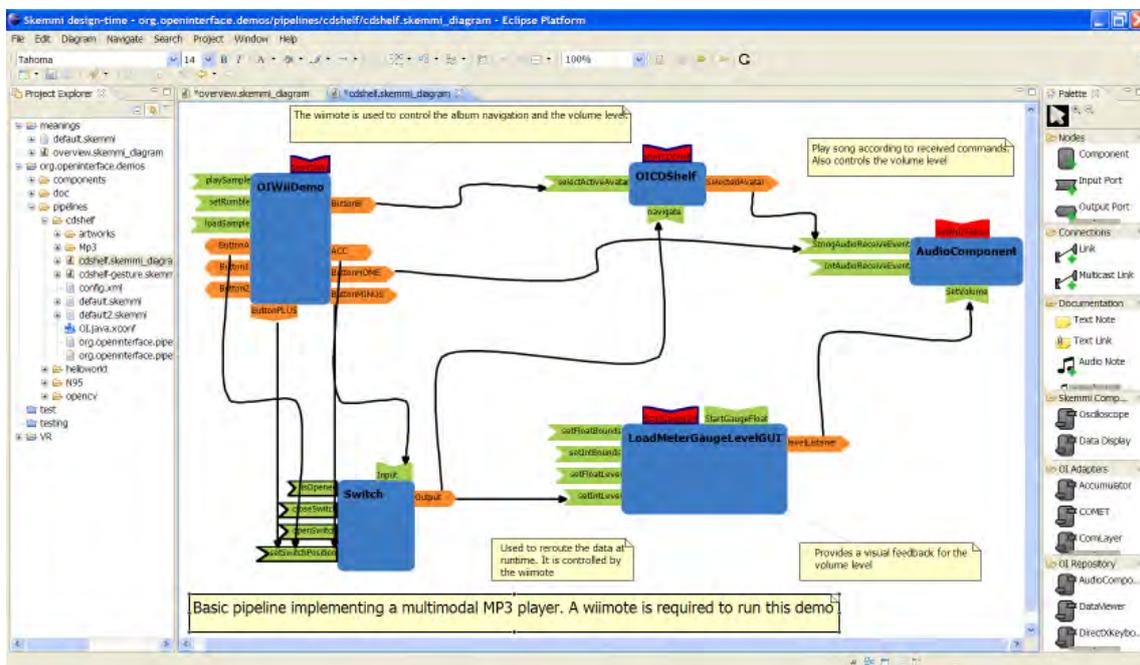


Figure 23. Interface du logiciel OpenInterface.

## 4.2. APPROCHES ANALYTIQUES

En IHM, plusieurs travaux ont proposé de classer et de décrire les dispositifs d'entrée et leurs caractéristiques sur la base des capteurs utilisés. Parmi eux, l'espace de conception de Card [26] définit trois opérateurs de composition qui peuvent être utilisés pour combiner des capteurs: Merger, Layout et Connection. Ces trois opérateurs se réfèrent à la composition des données (Merger), à la composition spatiale (Layout) et à la possibilité de brancher la sortie d'un capteur sur l'entrée d'un autre capteur (Connection).

À un niveau d'abstraction plus élevé, les propriétés multimodales ont été conçues pour mieux décrire la composition des modalités d'interaction. Les propriétés temporelles d'Allen [3] (Figure 24) définissent la composition temporelle entre deux modalités

d'interaction dans le temps. Les propriétés CARE [35] définissent les relations entre plusieurs modalités définies par un couple <dispositif, langage>. La fusion des modalités est effectuée au niveau du langage.

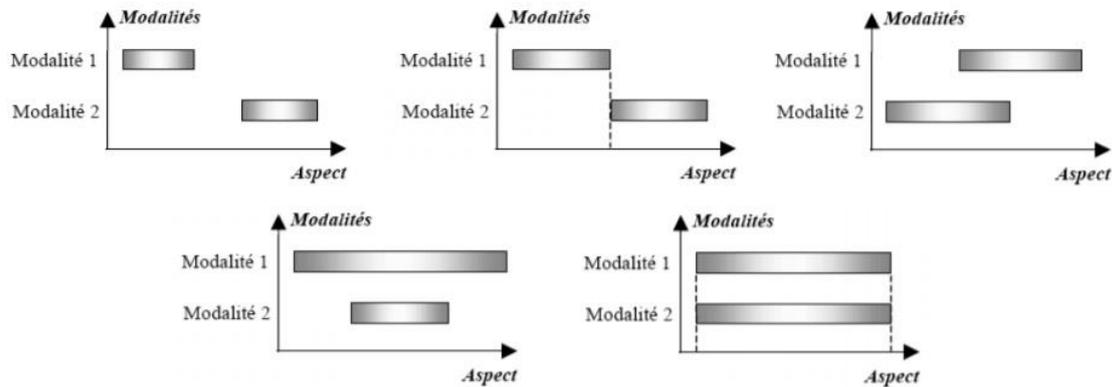


Figure 24. Cinq relations d'Allen décrivant la composition temporelle de modalités d'interaction (tiré de [168]).

Hartmann et al. [71] ont étudié la combinaison de capteurs avec des dispositifs réels et ont mis en lumière l'importance de l'aspect physique d'une telle composition. Selon ces auteurs, deux dispositifs ou capteurs conçus pour supporter explicitement leur composition et qui sont conscients les uns des autres conduisent à un meilleur dispositif résultant. Ce type de composition est appelé "joint à queue d'aronde". Même si les auteurs mettent en garde les futurs concepteurs de dispositifs contre les risques d'une mauvaise composition, il manque un modèle de conception qui décrirait mieux les propriétés de la composition d'un point de vue physique.

#### 4.3. Combinaison de dispositifs mobiles

De nombreux travaux proposent de combiner plusieurs dispositifs mobiles afin d'augmenter leur capacités d'interaction [80, 96, 116, 166]. Cette combinaison peut être faite en attachant plusieurs dispositifs entre eux [71, 80, 96, 113, 147, 169] ou en les conservant séparés [30, 31, 41, 101, 119, 156, 163, 166]. Lorsque les dispositifs sont séparés, il est important que ceux-ci puissent se reconnaître entre eux [139, 140, 199] afin de pouvoir transmettre des informations pertinentes pour l'interaction [68, 116, 124, 149, 164]. Rädle et al. mettent en avant l'importance de la notion de spatialisation des dispositifs pour l'interaction : la prise en compte de la spatialisation des dispositifs dans la conception des techniques d'interaction permet de réduire la charge cognitive ressentie par les utilisateurs. À titre d'exemple, Siftables est basé sur la combinaison spatiale de plusieurs petits écrans [116]. Ceux-ci peuvent être rassemblés ou séparés afin de réaliser

des tâches telles que l'application de filtres ou le tri des éléments selon un critère spécifique. Rooke et al. (Figure 25 - Gauche) proposent l'utilisation d'une combinaison de plusieurs petits écrans hexagonaux sans bords. En complément de l'interaction sur ces écrans (permettant la sélection d'éléments dans un menu ou activant une fonctionnalité), ceux-ci peuvent être manipulés physiquement (regroupement, rotation, inclinaison) afin de déclencher des commandes.

Une solution alternative consiste en la liaison physique entre plusieurs dispositifs mobiles [71, 80, 96 113, 147, 169]. Codex est un dispositif mobile possédant deux écrans [80]. Il résulte de la combinaison entre deux tablettes au travers d'une coque commune. Les deux dispositifs peuvent être placés dans plusieurs configurations spatiales l'un par rapport à l'autre, étendant ainsi les usages possibles de ce dispositif composite. À titre d'exemple, les deux écrans peuvent être associés à une tâche différente (lecture et annotation) ou chacun à un utilisateur dans un contexte collaboratif. Dualscreen (Figure 25 - Droite) combine deux écrans sur un ordinateur pour partager du contenu personnel avec d'autres utilisateurs [80]. Dans ce contexte, le second écran n'est donc pas destiné à l'utilisateur principal mais permet de transmettre de l'information dans le cas d'un travail collaboratif.

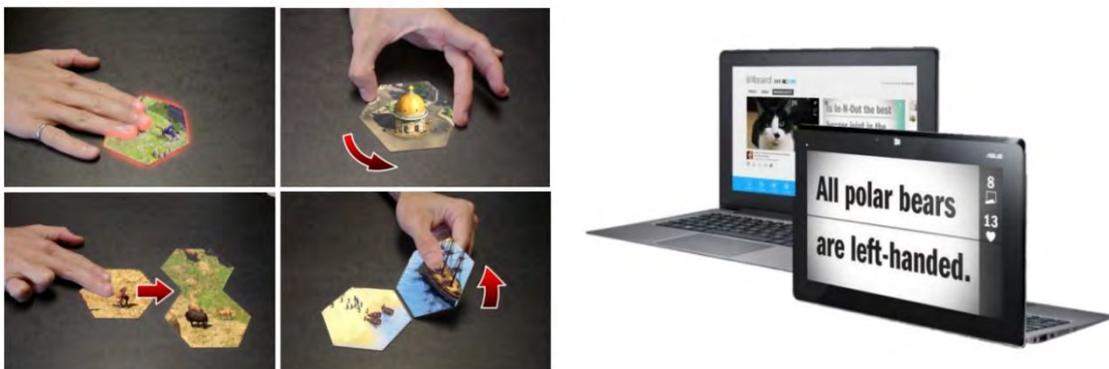


Figure 25. Interaction spatiale avec plusieurs dispositifs mobiles (Gauche, tiré de [149]) et combinaison physique de dispositifs mobiles (Droite, tiré de [96]).

D'autres travaux proposent l'utilisation de multiples dispositifs mobiles disjoints [30, 31, 41, 119, 166]. Au travers de Duet (Figure 26 - Gauche), Chen et al. [31] proposent l'utilisation de la smartwatch comme dispositif complémentaire afin d'étendre les degrés de libertés de l'interaction tactile avec un smartphone. La smartwatch peut être utilisée comme menu contenant un ensemble de commandes, libérant ainsi de l'espace d'affichage sur le smartphone. De plus, les capteurs embarqués de la smartwatch permettent d'étendre le vocabulaire d'interaction tactile sur l'écran du smartphone en ajoutant des gestes de « knuckle » (toucher avec l'articulation du doigt) ou de « flip and

touch » (toucher suite à une rotation du poignet). HoverPad envisage l'utilisation d'un écran au-dessus d'une table tactile [166]. Grâce à l'intégration d'une centrale inertielle sur l'écran, celui-ci peut être localisé au-dessus de la table tactile. Attaché à un ensemble de moteurs, l'écran peut se déplacer automatiquement au-dessus des données affichées sur la table tactile, servant ainsi de « magic lense » et permettant d'afficher des informations complémentaires (i.e. réalité augmentée) à celles affichées sur l'écran principal.

Basés sur l'utilisation d'un smartphone comme dispositif complémentaire, de nombreux travaux se sont focalisés sur les intérêts que peut apporter leur combinaison avec une tablette. Dans Thaddeus [197], une représentation graphique de données peut être explorée à l'aide d'un smartphone en manipulant sa position relativement à la tablette (Figure 26 - Droite). Dans BATMAN [135], la position du smartphone est utilisée pour déclencher des actions spécifiques dans une application comme la sélection d'un élément dans un menu circulaire ou la configuration de la taille du pinceau dans une application de dessin. Cependant, ces approches reposent sur l'utilisation de capteurs externes afin de détecter la position du smartphone autour de la tablette, ce qui peut contraindre l'adoption de telles techniques d'interaction au-delà d'un contexte de recherche (c.-à-d. équipé).



Figure 26. Combinaison de dispositifs mobiles: un smartphone avec une smartwatch (Gauche, tiré de [31]) ainsi qu'un smartphone avec une tablette (Droite, tiré de [197]).

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 [105], l'écran d'un ordinateur de bureau affiche un marqueur spécifique (motif coloré) qui est détecté par la caméra d'un smartphone (Figure 27 – Haut). Ce marqueur permet au smartphone de calculer sa position par rapport à l'écran. L'utilisation d'une centrale inertielle embarquée est utilisée pour détecter son orientation. Ainsi, le smartphone peut être utilisé comme lentille magique ou directement comme objet tangible dans l'application (p. ex. une plateforme 2D sur lequel le personnage du joueur peut sauter). Dans ACTUI [107], 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 3ème dimension d'un modèle 2D affiché sur l'écran principal. Enfin, dans PhoneTouch [163], Schmidt et al. proposent l'utilisation des capteurs du smartphone (accéléromètre) en combinaison avec les capacités de détection d'une table tactile afin de reconnaître le stacking d'un smartphone (Figure 27 – Bas). Dans ce contexte, le stacking est utilisé comme interacteur tangible afin de transférer du contenu entre le smartphone et la tablette (p. ex. transférer un ensemble de photos, sélection d'éléments).

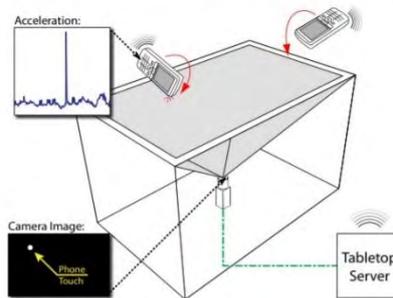


Figure 27. Interaction basée sur le principe du stacking avec un écran d'ordinateur (Haut, tiré de [105]) ou une table tactile (Bas, tiré de [163]).

## 5. SYNTHÈSE

Dans la littérature, il apparaît que les dispositifs isotoniques sont plus adaptés au contrôle d'un mouvement alors que les dispositifs isométriques sont plus adaptés au contrôle d'une vitesse [206]. Afin de couvrir l'ensemble des tâches réalisables lors de l'interaction avec des données multidimensionnelles, ces deux catégories de dispositifs sont utiles. L'utilisation séquentielle de plusieurs dispositifs (certains isométriques, d'autres isotoniques) permettrait de fluidifier l'interaction avec les données multidimensionnelles. Cependant, le temps de changement de dispositif induit par leur nombre, pourrait nuire à la fluidité de l'interaction [59]. Ainsi, il apparaît que la conception d'un dispositif tout-en-un, possédant de nombreux degrés de libertés et tirant profit des catégories isotoniques et isométriques soit une solution pertinente. Nous proposons un tel prototype, appelé la Roly-Poly Mouse dans le chapitre 4.

Concernant les techniques d'interaction adaptées aux dispositifs mobiles, l'utilisation du multi-touch induit des problématiques de découvrabilité, de mémorisation et d'occultation des données. Le stylet, alternative pertinente au multi-touch, n'est pas disponible pour tous les dispositifs mobiles. De plus, celui-ci n'est pas toujours différenciable d'un doigt par la surface tactile. En outre, l'interaction en l'air nécessite un équipement coûteux et potentiellement encombrant. Ces solutions sont donc difficilement envisageables. Or, l'une des limitations de l'interaction sur dispositif mobile réside dans la taille réduite de l'espace d'affichage et d'interaction. En conséquence, ce type d'interaction ne nous semble pas être une solution adaptée à notre problématique.

En revanche, le paradigme de composition de dispositifs semble être une solution pertinente dans un contexte de mobilité : il permet d'étendre les degrés de libertés des dispositifs mobiles sans être limité par les problématiques liées à l'utilisation de l'interaction multi-touch ou du stylet. Cependant, les nombreuses approches de composition existantes ne se focalisent pas sur l'aspect physique de la composition, pourtant primordial afin de concevoir une technique d'interaction adaptée à un contexte donné. Ainsi, nous proposons un espace de conception visant à caractériser de manière systématique la composition de dispositifs d'un point de vue physique. Nous avons appelé cet espace de conception : DECO (DEvice COMposition). Nous décrivons cet espace de conception dans le chapitre 5.

Différents types de dispositifs composés peuvent alors être envisagés. Cependant, le contexte industriel de nos travaux limite les solutions pouvant être envisagées. En conséquence, les solutions proposant l'altération physique de dispositifs mobiles [40, 75, 109, 165, 170, 176, 203] ne semblent pas pertinentes dans notre contexte : les clients de l'entreprise ne possèdent pas tous des dispositifs mobiles dédiés à leur fonction et il n'est pas envisageable d'altérer leurs dispositifs personnels. Ainsi, une utilisation disjointe de ces dispositifs semble plus adaptée. Dans ce contexte, l'utilisation de capteurs coûteux et/ou nécessitant un environnement contrôlé (p. ex. système de caméras OptiTrack) n'est pas envisageable. En effet, les utilisateurs des logiciels Berger-Levrault peuvent être des élus de petites collectivités, ne disposant pas de budgets élevés pour l'achat de telles technologies. Par conséquent, l'approche du stacking, ne modifiant pas les dispositifs mobiles existants et ayant un coup de production faible, semble la plus adaptée à notre contexte. Ainsi, nous proposons une solution basée sur le stacking pour l'interaction avec des données multidimensionnelles en mobilité. Nous décrivons cette solution dans le chapitre 6.

# CHAPITRE 4

## LA ROLY-POLY MOUSE : CONCEPTION D'UN DISPOSITIF A MULTIPLES DEGRES DE LIBERTE

---

### Structure du chapitre

---

1. INTRODUCTION.....	70
1.1. Motivation.....	70
1.2. Structure du chapitre.....	71
2. CONCEPTION ET PROTOTYPAGE DE LA ROLY-POLY MOUSE.....	72
2.1. La Roly-Poly Mouse.....	72
2.1.1. Facteurs de conception.....	73
2.1.2. Prototypes développés.....	76
3. ÉTUDE PRELIMINAIRE : FORME DU DISPOSITIF ET PRISE EN MAIN .	79
3.1. Description de l'étude.....	79
3.2. Résultats.....	79
4. ÉTUDE 1 : AMPLITUDE ET STABILITE DES GESTES .....	81
4.1. Description de l'étude.....	81
4.2. Résultats.....	83
5. ÉTUDE 2 : POINTAGE 2D .....	85
5.1. Algorithme correctif .....	85
5.2. Description de l'étude.....	86
5.3. Résultats.....	88
6. ÉTUDE 3 : MANIPULATION 3D .....	90
6.1. Techniques d'interaction pour la manipulation 3D .....	90
6.2. Description de l'étude : translation et rotation 3D.....	92
6.3. Résultats.....	95

7.	TRANSFERT DANS LE CONTEXTE DE L'ENTREPRISE .....	97
7.1.	Organis'Or .....	97
7.2.	Utilisation envisagée de la RPM.....	98
7.3.	Intégration.....	98
8.	DISCUSSION ET PERSPECTIVES .....	99
8.1.	Limitations de nos études .....	99
8.2.	Conseils de conception pour dispositifs inclinables .....	99
8.3.	Instabilité du dispositif.....	100
8.4.	Axe Z .....	100
8.5.	Défis techniques.....	101
8.6.	Autres scénarios d'utilisation.....	103
9.	CONCLUSION DU CHAPITRE 4 .....	105

---

### Publications associées

---

G. Perelman, M. Serrano, M. Raynal, C. Picard, M. Derras and E. Dubois. Conception d'un Dispositif pour Interagir avec des Données Multidimensionnelles : Disco. IHM'14, 91-100. Article Long

G. Perelman, M. Serrano, M. Raynal, C. Picard, M. Derras and E. Dubois. Disco(s) : Dispositifs à Multiples Degrés de Liberté pour Interagir avec des Données Multidimensionnelles. IHM'15. Démonstration

G. Perelman, M. Serrano, M. Raynal, C. Picard, M. Derras and E. Dubois. The Roly-Poly Mouse: Designing a Rolling Input Device Unifying 2D and 3D Interaction. CHI'15, 327-336. Article Long

# 1. INTRODUCTION

## 1.1. MOTIVATION

Dans le chapitre précédent, nous avons décrit plusieurs dispositifs visant à fournir un plus grand nombre de degrés de libertés (DDL). L'objectif de tels dispositifs est de faciliter l'utilisation et l'accessibilité à des données multidimensionnelles complexes. Cependant, ces dispositifs sont généralement proposés en complément de la souris classique. Il devient donc nécessaire, selon les données manipulées, de changer de dispositif régulièrement. Or, ce changement peut être coûteux en termes de temps : il est, par exemple, nécessaire de déplacer la souris de la zone d'interaction (zone dans laquelle le dispositif en main dominante est manipulé de manière confortable), la relâcher, prendre en main le nouveau dispositif et le déplacer dans la zone d'interaction [59].

À titre d'exemple, dans un logiciel de CAO (p. ex. Blender, unreal engine 4), une souris classique est particulièrement adaptée pour interagir avec l'interface graphique et les menus : elle est utilisée pour configurer une action, activer des options, ou encore enregistrer son travail. En revanche, les deux degrés de libertés qu'elle offre ne permettent que difficilement de manipuler un objet 3D : il devient nécessaire d'intégrer un mode de manipulation (maintien du clic droit, touche spécifique du clavier, ...), complexifiant l'interaction. Pour des experts en interaction 3D, cette difficulté peut être rapidement surmontée car de nombreux raccourcis clavier existent afin de fluidifier l'interaction. Cependant, pour des utilisateurs novices ou occasionnels, l'apprentissage de ces raccourcis représente un frein important à la maîtrise du logiciel. Pour faciliter la courbe d'apprentissage, des dispositifs dédiés, tels que la souris 3D, permettent une manipulation plus efficace de modèles 3D en proposant une plus grande quantité de degrés de libertés et en permettant une manipulation directe de l'objet 3D sélectionné (p. ex. une rotation sur la souris appliquera la même rotation sur l'objet 3D sélectionné). Néanmoins, à l'inverse de la souris (position control), ce type de dispositif contrôle la vitesse du curseur (rate control) : ce dispositif n'est donc pas adapté à l'interaction avec une interface graphique traditionnelle, rendant cette tâche fastidieuse [207].

Le premier travail préliminaire à cette thèse a consisté en la conception d'un dispositif tout-en-un qui permet d'éviter le surcoût occasionné par le changement de dispositif ainsi qu'en l'étude de ses apports pour une activité mêlant environnement 3D et interface graphique (GUI) 2D. Ce nouveau dispositif conserve les avantages de la souris

(interaction simple avec l'interface graphique) tout en facilitant la manipulation des objets 3D (manipulation directe, degrés de liberté suffisants). Nous l'avons appelé la Roly-Poly Mouse (RPM), en référence à sa forme et son comportement physique proches de ceux d'un culbuto (roly-poly toy en anglais). Ce jouet (Figure 28 - Gauche) possède une base arrondie et lestée qui lui permet de toujours revenir à la verticale, après une période d'oscillation, lorsqu'il est relâché. Nous avons réalisé et publié ces travaux au cours de l'année qui précédait le démarrage de cette thèse. Ces travaux ont été menés dans le cadre d'une collaboration entre l'équipe ELIPSE (Gary Perelman, Emmanuel Dubois, Marcos Serrano, Mathieu Raynal) et l'entreprise Berger-Levrault (Célia Picard, Mustapha Derras).

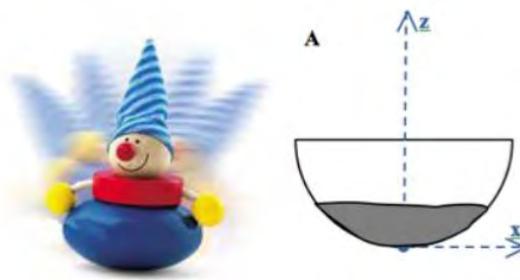


Figure 28. La Roly-Poly Mouse, basée sur le culbuto (gauche) possède un centre de gravité bas (droite), ce qui lui permet de revenir à la verticale après avoir été relâché.

## 1.2. STRUCTURE DU CHAPITRE

Dans la section 2, nous décrivons les étapes de conception des différentes versions et prototypes de la Roly-Poly Mouse ainsi que les technologies sur lesquelles ces versions sont basées. La section 3 présente une étude préliminaire visant à explorer différentes formes pour le dispositif ainsi que les prises en main qui y sont associées. La section 4 décrit l'amplitude des gestes réalisables avec notre dispositif ainsi que sa stabilité au travers d'une première étude expérimentale. Nous comparons ensuite notre dispositif avec une souris traditionnelle dans une tâche de pointage (section 5), puis à une souris 3D dans deux tâches (translation et rotation) de manipulation d'objet 3D (section 6). Puis, nous décrivons l'intégration de notre prototype dans plusieurs logiciels existants et nous discutons les avantages que celui-ci apporte (section 7). Enfin, dans la section 8, nous discutons les avantages et limitations de notre dispositif (dans ses différentes implémentations matérielles) et suggérons des pistes d'amélioration.

## 2. CONCEPTION ET PROTOTYPAGE DE LA ROLY-POLY MOUSE

### 2.1. LA ROLY-POLY MOUSE

La RPM est basée sur le principe du culbuto (Figure 28). Sa partie basse est hémisphérique et lestée, ce qui induit un centre de gravité bas. Lorsque le dispositif est relâché, il retourne à sa position initiale d'équilibre après un mouvement de roulis inertiel.

Le dispositif offre jusqu'à 6 degrés de liberté (DDL) permettant trois types de manipulations distinctes : les translations (3 DDL), les inclinaisons (2 DDL : roulis, tangage) et la rotation (1 DDL : lacet). Dans le contexte d'une utilisation dans un environnement de bureau classique, nous n'avons pas considéré les translations sur l'axe Z (perpendiculaire à la surface sur laquelle est posé la RPM). En effet, ces translations nécessitent de maintenir le dispositif en l'air et peuvent ainsi causer de la fatigue. Nous avons donc choisi d'écarter ce degré de liberté de nos études. Ainsi, dans nos travaux, nous considérons que la Roly-Poly Mouse ne possède que 5 degrés de libertés : 2 en translation, 2 en inclinaison et un en rotation.

La forme de la partie basse, hémisphérique, que nous avons choisi d'explorer présente de nombreux avantages qui rendent ce dispositif particulièrement intéressant. Premièrement, cette forme rend possible les translations de la RPM sur le plan x-y (représenté par une surface plane sur lequel il est posé, p. ex. une table). La RPM peut donc être utilisée comme n'importe quel dispositif isotonique<sup>1</sup> [206] (position control) tel que la souris traditionnelle [25]. De plus, la surface de contact entre le dispositif et son plan support est minimale, ce qui réduit considérablement la friction avec la surface d'appui et rend ainsi les translations du dispositif plus fluides. En parallèle, les inclinaisons du dispositif et l'auto-repositionnement qui en découle lorsqu'il est relâché sont des caractéristiques proches des dispositifs isométriques<sup>2</sup> [206] (rate control) ou élastiques [25]. Deuxièmement, la partie basse hémisphérique du dispositif offre une large surface courbe, ce qui permet une amplitude d'inclinaison particulièrement élevée, contrairement

---

<sup>1</sup> Un dispositif isotonique, aussi appelé dispositif de déplacement ou de mouvement libre, est un dispositif qui détecte un déplacement. Ce dispositif possède une force nulle ou constante à une force qui lui est appliqué.

<sup>2</sup> Un dispositif isométrique, aussi appelé dispositif de pression ou de force, est un dispositif qui détecte une force qui lui est appliqué mais qui ne détecte pas le mouvement. Ce dispositif possède une résistance élevée à la force qui lui est appliqué.

aux approches existantes dans la littérature ; dont notamment la VidéoMouse [78] et la Rockin' Mouse [7], deux souris à multiples DDL bien connues. Troisièmement, la RPM est symétrique sur son axe de rotation (lacet), ce qui permet de l'utiliser dans n'importe quelle orientation. Enfin, les trois gestes simples (translation, rotation et inclinaison) peuvent être combinés pour créer des gestes composés.

Pour résumer, le facteur de forme retenu pour la RPM combine les caractéristiques d'un dispositif isotonique et isométrique, ce qui en fait un candidat de choix pour un **dispositif tout-en-un** (Table 1).

Degrés de liberté	Souris traditionnelle		Rockin' Mouse [7]		Video Mouse [78]		Space Mouse [173]		Roly-Poly Mouse	
	Déecté	Vitesse/Position	Déecté	Vitesse/Position	Déecté	Vitesse/Position	Déecté	Vitesse/Position	Déecté	Vitesse/Position
	Tx	✓	p	✓	p	✓	p	✓	v	✓
Ty	✓	p	✓	p	✓	p	✓	v	✓	p
Tz	x		x		✓	p	✓	v	✓	p*
Rx	x		✓	v,p	✓	v,p	✓	v	✓	v,p
Ry	x		✓	v,p	✓	v,p	✓	v	✓	v,p
Rz	x		x		✓	v	✓	v	✓	v,p
<b>T+R</b>	x		x		x		✓	v	✓	v,p

Table 1. Comparatif des degrés de libertés détectés par plusieurs souris augmentées (\*Tz est détectée mais n'es pas utilisée). V signifie un contrôle de vitesse et P un contrôle de position.

### 2.1.1. FACTEURS DE CONCEPTION

La première étape de conception de la RPM a été d'identifier et d'analyser les nombreux facteurs qui peuvent avoir un effet sur son usage. Ces facteurs sont les suivants : taille du dispositif ; mécanisme de sélection ; forme de la partie haute ; prise en main ; amplitude des gestes ; intégration et séparabilité des degrés de libertés.

#### Taille du dispositif

La taille de la RPM est basée sur la taille moyenne d'une souris traditionnelle (approximativement 12x6cm). Nous avons réalisé des tests informels auprès d'un panel de 6 utilisateurs pendant lesquels nous avons testé trois tailles différentes pour le dispositif : 6, 8 et 10 cm de diamètre. Nos tests ont révélé que la version de 8cm de diamètre était

la plus confortable à prendre en main et simple à manipuler. Par conséquent, tous nos prototypes de la RPM sont basés sur cette taille.

#### Mécanisme de sélection

La RPM ne peut pas contenir de boutons sur l'avant du dispositif (comme la souris traditionnelle) car ceux-ci pourraient être difficiles d'accès lorsqu'une rotation du dispositif est réalisée : le dispositif possédant une symétrie centrale, il ne possède pas de partie avant. Il est donc nécessaire que les boutons soient toujours disponibles, quelles que soient les manipulations de la RPM. Nous avons identifié deux solutions : utiliser un bouton en forme d'anneau disposé tout autour du dispositif ou utiliser une surface capacitive comme couvercle de la RPM. La première solution a été implémentée dans un des prototypes que nous décrivons dans la section 2.1.2. La seconde solution a été proposée dans des travaux ultérieurs [157] dans lesquels un écran tactile a été intégré sur le couvercle de la RPM.

#### Forme de la partie haute

Nous avons considéré différentes formes pour la partie haute de l'hémisphère composant la RPM. Dans la suite de ce chapitre, nous nous y référerons en tant que « couvercle ». Souhaitant proposer une forme symétrique (sphère complète) et confortable à manipuler, nous avons opté pour une forme hémisphérique. Nous avons fait varier son degré de courbure ainsi que la direction de la courbure (vers l'intérieur ou l'extérieur du dispositif). Dans notre étude préliminaire, nous avons considéré trois couvercles possibles : Hémisphérique, Convexe (courbée vers l'extérieur) ou Concave (courbée vers l'intérieur) (Figure 29). Pour la version Concave, aucun couvercle n'était ajouté. Pour la version Hémisphérique, une seconde demi-sphère en plastique était placée sur le dispositif. Enfin, pour la version Convexe, nous avons réalisé une coupe dans une sphère de 12cm de diamètre afin d'en conserver uniquement un cercle de 8cm de diamètre. Ce procédé a permis d'avoir un couvercle bien plus aplati : cette courbure est proche de celle d'une souris traditionnelle.

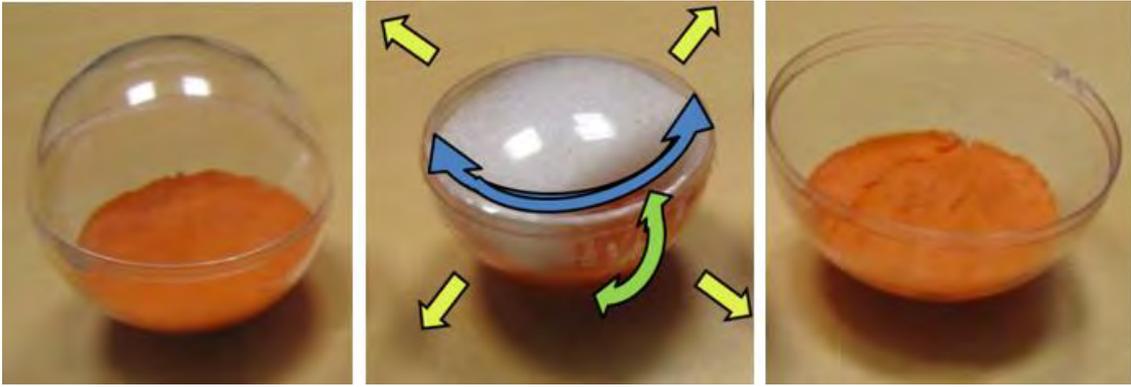


Figure 29. Couverts de la RPM : Hémisphérique (gauche), Convexe (centre) et Concave (droite).

#### Prise en main

Des travaux en psychologie et physiologie traitant de la prise en main d'objets tangibles ont montrés que le type de prise en main (nombre et position des doigts) dépendent de la forme de l'objet manipulé autant que de l'objectif de la manipulation [123]. De plus, la prise en main a déjà été utilisée comme modalité d'interaction dans de précédents travaux [177]. En conséquence, nous avons conduit une étude préliminaire visant à identifier les différentes prises en main les plus fréquentes lors de l'utilisation de la RPM. Nous explorons ensuite les avantages et inconvénients de celles-ci dans nos études principales et évaluons leur impact sur l'amplitude des gestes réalisables avec notre dispositif.

#### Amplitude des gestes

Chacun des trois gestes (translation, rotation et inclinaison) permis par notre dispositif possède des amplitudes distinctes. La RPM étant un dispositif sans-fil, le mouvement de translation sur les axes x-y n'est limité que par le plan sur lequel elle est posée. Concernant la rotation, l'amplitude est en théorie illimitée. En effet, le dispositif est parfaitement symétrique selon son axe de rotation : une rotation de  $360^\circ$  replace le dispositif en position initiale, il est donc possible de continuer la rotation à l'infini en théorie. L'amplitude d'inclinaison est théoriquement de  $180^\circ$  (de  $+90^\circ$  à  $-90^\circ$ ). Cependant, la prise en main et les limitations biomécaniques des articulations impliquées dans ces deux gestes, tels que le poignet, restreignent ces amplitudes de manipulation théoriques [114, 141]. En prenant en compte les prises en mains définies dans notre étude préliminaire, nous avons exploré l'amplitude des trois gestes possibles à travers notre première étude principale.

#### Intégration et séparabilité des degrés de liberté

Les différents degrés de libertés de la RPM sont physiquement intégrés : lorsqu'une translation est effectuée, la base du dispositif étant arrondie, de petites inclinaisons/rotations sont involontairement effectuées ; lors d'une rotation/inclinaison, le centre de masse du dispositif sera inévitablement déplacé. Afin de permettre une séparabilité des degrés de liberté efficace [87], il est nécessaire de connaître l'impact de chaque degré de liberté sur les autres. Cette question est abordée dans la première étude de ce chapitre.

Dans les sections suivantes, nous décrivons les deux prototypes de la Roly-Poly Mouse que nous avons conçus, les technologies sur lesquelles ils sont basés ainsi que les avantages et inconvénients de chacune des approches.

### 2.1.2. PROTOTYPES DEVELOPPES

#### Prototype expérimental

Nous avons développé un premier prototype destiné à être utilisé lors de nos expérimentations, nous avons donc choisi une technologie offrant la meilleure précision la plus importante possible : les caméras infrarouges OptiTrack. La base hémisphérique de la RPM est composée d'une demi-sphère en plastique de 8cm de diamètre et de 3mm d'épaisseur. Afin que son poids soit similaire à celui d'une souris classique (de 70 à 100g selon le modèle) et ainsi éviter un temps d'adaptation à l'utilisateur, nous avons lesté cette base à l'aide de pâte à modeler : la RPM pèse 80g. Puis, selon la partie haute du dispositif, trois versions étaient possibles.

Pour la détection des manipulations, nous nous sommes basés sur un système de 12 caméras OptiTrack (Figure 30 – Droite). Ce système permet de détecter les translations avec une précision d'un millimètre et les rotations et inclinaisons avec une précision de  $0.1^\circ$ . Les informations de position et d'orientation sont collectées par le système à une fréquence de 100Hz. Nous avons équipé notre dispositif d'un ensemble de quatre billes réfléchissant la lumière infrarouge que nous avons disposées sur quatre pics en métal, sortant de quatre directions du dispositif (Nord-Ouest, Nord-Est, Sud-Est et Sud-Ouest, Figure 30 - Gauche). Ces billes permettaient aux caméras de détecter le modèle de la RPM et ainsi d'en déduire sa position et son orientation en temps réel. Le placement de ces pics en métal a fait l'objet de tests informels afin de vérifier que leur présence et leur disposition ne gênent pas les manipulations et l'amplitude des mouvements réalisables. Sur ce prototype, aucun mécanisme de sélection (clic gauche ou droit) n'était intégré.

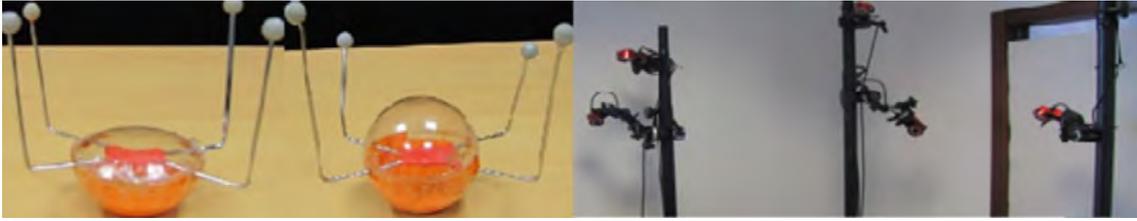


Figure 30. Prototypes de la RPM basés sur la technologie OptiTrack (gauche). Dispositif des caméras utilisées pour localiser le dispositif (droite).

### Version intégrée

Nous souhaitons concevoir un second prototype plus embarqué de la RPM, initialement basée sur un système de 12 caméras. Cette technologie permet une localisation 6D (position et orientation 3D) avec une grande précision mais induit un coût et un encombrement élevé. Nous avons donc conçu un prototype moins performant (en termes de précision), mais totalement intégré et avec un coût de fabrication plus faible (Figure 31). Ce prototype est basé sur la même boule en plastique hémisphérique que le prototype précédent lesté à l'aide de pâte à modeler. En revanche, le système de détection des manipulations est basé sur l'utilisation d'un capteur de type « flock of birds » sans fil (Polhemus Patriot [137]). Ce capteur permet la détection des rotations et inclinaisons du dispositif avec une précision d'un degré. Les translations de la RPM sont détectées avec une précision de 6mm. Les données de ce type de capteur étant relativement bruitées, nous avons utilisé un 1€ filter [1]. De plus, nous avons intégré un mécanisme de sélection en nous basant sur une approche similaire à celle du bracelet WatchIt [132] : le bouton consiste en un potentiomètre résistif (81x7.5x0.5mm), enroulé autour de la RPM, qui fournit jusqu'à 1024 valeurs de pressions selon la position à laquelle le point de contact est détecté. Ainsi, une pression sur une section de la bande peut être détectée et située sur le contour de la RPM. En combinant la position de cette pression avec l'orientation actuelle du dispositif, nous pouvons proposer des boutons tout autour de la RPM et qui conservent leur position relative à l'utilisateur tout au long des différentes manipulations. Ainsi, un bouton situé à l'avant du dispositif restera toujours à cette position, peu importe l'orientation de RPM. Pour interfacer cette bande résistive, nous avons utilisé un Arduino Fio équipé d'une batterie externe et d'un module Bluetooth.

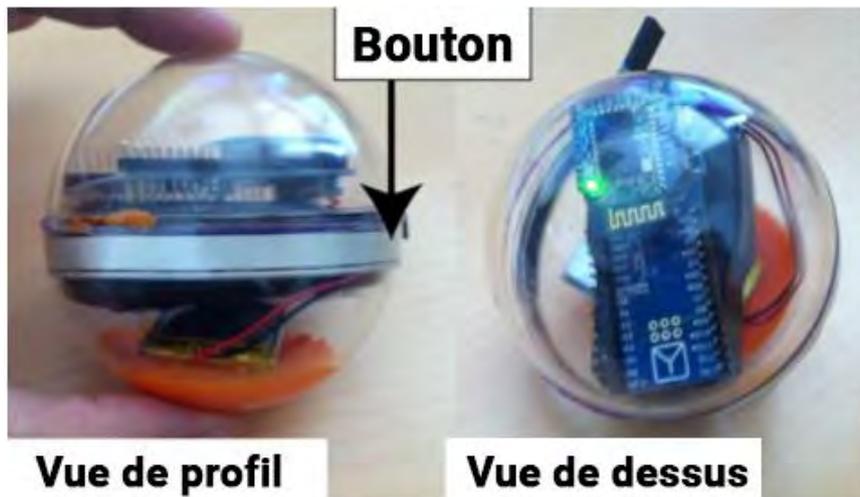


Figure 31. Prototype intégré de la RPM.

Afin d'explorer différentes formes et tailles de la RPM et d'en conserver les plus utilisables, nous avons utilisé un prototype dénué de tout système de localisation. Puis, nous avons utilisé le prototype expérimental afin d'explorer les capacités d'interaction du dispositif et d'en mesurer la précision : une précision maximale était nécessaire pour mesurer les amplitudes de mouvements, nous ne pouvions donc pas utiliser le prototype intégré. Enfin, nous avons utilisé la version intégrée de la RPM afin d'illustrer plus efficacement le transfert de ces travaux au travers de plusieurs logiciels existants.

## 3. ÉTUDE PRELIMINAIRE : FORME DU DISPOSITIF ET PRISE EN MAIN

L'objectif de cette étude préliminaire était d'identifier les prises en main les plus fréquentes pour chaque geste réalisable avec la RPM (translation, inclinaison, rotation et gestes composés) ainsi que la préférence des utilisateurs vis-à-vis des trois versions (Concave, Convexe et Hémisphérique) de notre dispositif.

### 3.1. DESCRIPTION DE L'ÉTUDE

Nous avons étudié les trois formes (Hémisphérique, Convexe, Concave) du couvercle du dispositif et demandé aux utilisateurs, pour chaque forme, de tenir le dispositif en main dominante pour réaliser chacun des quatre gestes possibles : translations, rotations, inclinaisons et gestes composés. Pour cette étude, aucun système de détection des gestes n'a été utilisé. Ainsi, les prototypes de la RPM étaient uniquement composés de leur partie basse lestée et du couvercle associé à chaque version du dispositif. De plus, aucun feedback informatique n'était fourni aux utilisateurs. Ceux-ci étaient assis, face à une table sur laquelle était posée la version de la RPM associée à la tâche.

Pour cette étude, nous avons recruté 12 participants (tous membre d'un laboratoire universitaire), dont 2 femmes, avec un âge moyen de 24.8 ans (écart-type = 1.8). Parmi ces utilisateurs, deux avaient une expérience préalable avec des dispositifs à multiples degrés de liberté (souris augmentées ou souris 3D). Pour chaque geste réalisé, nous avons photographié la prise en main de l'utilisateur, ce qui nous a permis d'obtenir un total de 144 photographies (3 formes de RPM x 4 gestes x 12 participants). À la fin de l'expérimentation, nous avons demandé à chaque utilisateur d'ordonner les dispositifs selon leur forme préférée. Chaque session durait en moyenne 30 minutes.

### 3.2. RESULTATS

Sur les 12 participants, 7 ont préféré la forme Convexe, 4 ont préféré la forme Hémisphérique et seulement 1 a préféré la forme Concave. La version Concave de la RPM a été perçue comme difficile à prendre en main et à déplacer. En conséquence, nous avons choisi de la supprimer de la suite de nos études. Concernant les prises en main, nous avons annoté chaque photographie prise pendant l'expérience ce qui nous a permis d'identifier plusieurs motifs récurrents dont principalement trois qui étaient les plus

fréquents : Squeeze (Figure 32 - a), Lay (Figure 32 - b) et Touch (Figure 32 - c). Ces trois prises en main ont été utilisées pour les études suivantes.



*Figure 32. Catégories de prises en main du dispositif : Squeeze (a), Lay (b) et Touch (c).*

Une fois les formes et prises en main de la RPM identifiées, nous nous sommes attachés à déterminer les amplitudes et la stabilité des gestes réalisables avec notre dispositif, i.e. le comportement du dispositif au cours de son utilisation.

## 4. ÉTUDE 1 : AMPLITUDE ET STABILITE DES GESTES

Nous avons donc exploré la stabilité des gestes de RPM et l'amplitude maximale de confort pour chaque couvercle, considérant diverses prises en main. La stabilité des gestes correspond aux inclinaisons, rotations et translations involontaires réalisées lors d'une manipulation de la RPM.

### 4.1. DESCRIPTION DE L'ETUDE

#### Tâche et instructions

Sur la base de l'étude préliminaire, nous avons décidé d'étudier la stabilité et l'amplitude des gestes avec deux couvercles (Convexe et Hémisphérique) et trois prises en main (Squeeze, Lay et Touch). Nous avons choisi de contrôler les prises en main pour éviter tout effet confondant sur les résultats (la prise en main de l'utilisateur peut avoir un impact important sur l'amplitude des gestes).

La tâche consistait à effectuer des translations et inclinaisons dans 8 directions différentes (les 8 directions cardinales, Figure 33 – Gauche) ainsi que des rotations dans les deux directions possibles (Gauche et Droite). Nous avons inclus deux distances différentes (12 et 24 cm) pour la tâche de translation afin d'étudier son effet sur la stabilité de la RPM. Toutes les distances et directions ont été tracées sur la surface d'expérimentation avec respectivement des cercles et des lignes (Figure 33 - Gauche). Nous avons demandé aux participants d'exécuter les translations d'une manière confortable du centre de la zone de manipulation jusqu'au bord du cercle correspondant, puis de retourner au centre. En ce qui concerne les inclinaisons et les rotations, nous leur avons demandé d'effectuer le geste ayant la plus grande amplitude confortable. Afin de mesurer l'amplitude et la stabilité des gestes, nous avons dû équiper la RPM de capteurs pour mesurer sa position et son orientation de manière très précise (Figure 33 - Droite).

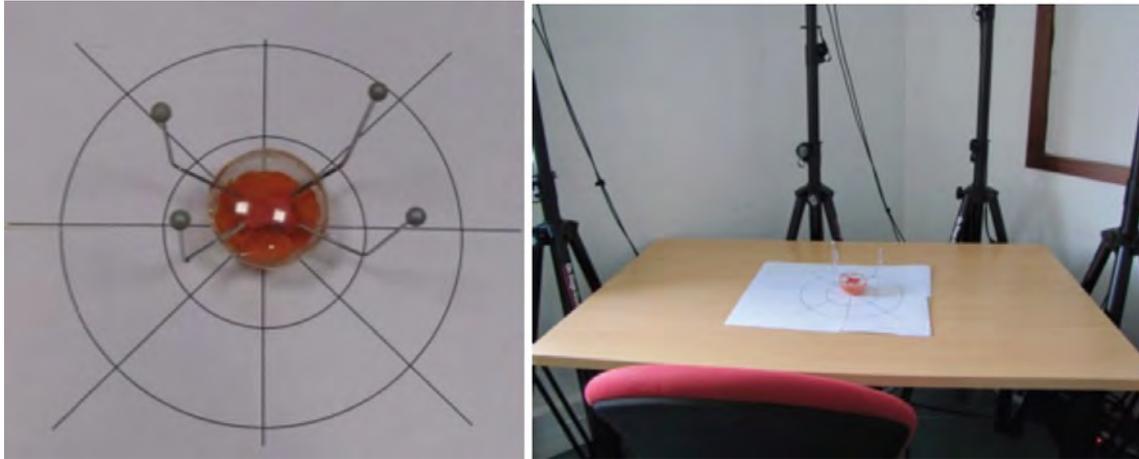


Figure 33. Configuration spatiale de l'expérimentation : lignes et cercles de manipulation (gauche) et disposition de la RPM dans son environnement de localisation (droite).

### Participants

Nous avons recruté 12 participants (tous membre d'un laboratoire universitaire), dont une femme, âgés en moyenne de 25,7 ans (écart-type = 3,8). L'un d'entre eux avait une expérience préalable avec des dispositifs à multiples degrés de liberté (souris augmentées ou souris 3D) et onze d'entre eux ont pris part à l'étude préliminaire.

### Protocole expérimental

L'étude suit un protocole 2x3x4 intra-sujet avec Couvercle (Convexe ou Hémisphérique), Prise en main (Squeeze, Touch et Lay) et Geste (Translation courte, Translation longue, Rotation et Inclinaison) comme facteurs. Nous avons contrebalancé les facteurs Couvercle et Prise en main à l'aide d'un carré latin. L'étude est composée de 6 blocs (chaque bloc est une combinaison d'un Couvercle et d'une Prise en main). Chaque bloc se compose de 8 translations courtes, 8 translations longues, 8 inclinaisons et 2 rotations ordonnées au hasard. Après chaque essai, la RPM est replacée dans sa position initiale, au centre de la table.

### Données collectées et analyse statistique

Nous avons collecté 2 dispositifs x 3 prises en main x (16 translations + 8 inclinaisons + 2 rotations) x 12 sujets = 1872 gestes. Après chaque bloc, nous avons mesuré la fatigue avec une échelle de Borg [18] allant de 6 à 20. Nous avons utilisé un test de Shapiro-Wilk pour déterminer la normalité des données recueillies. Si les données étaient normales ou pouvaient être normalisées, nous avons utilisé un test d'ANOVA uni-varié. Dans le cas contraire, nous avons utilisé un test de Friedman pour comparer plus de 2 conditions et

de Wilcoxon pour comparer 2 conditions. Quand nous avons exécuté plus d'un test statistique sur un ensemble particulier de données, nous avons utilisé la méthode de correction de Bonferroni. Seuls les résultats principaux sont présentés ci-après. Une analyse statistique plus poussée peut être trouvée dans les annexes section 1.1.

## 4.2. RESULTATS

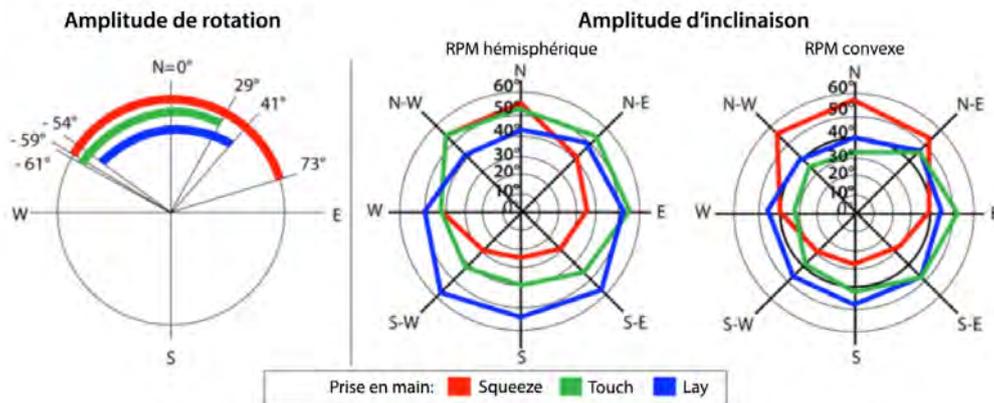


Figure 34. Amplitude de rotation (gauche) par prise en main. Amplitude d'inclinaison par prise en main pour la RPM hémisphérique (centre) et convexe (droite).

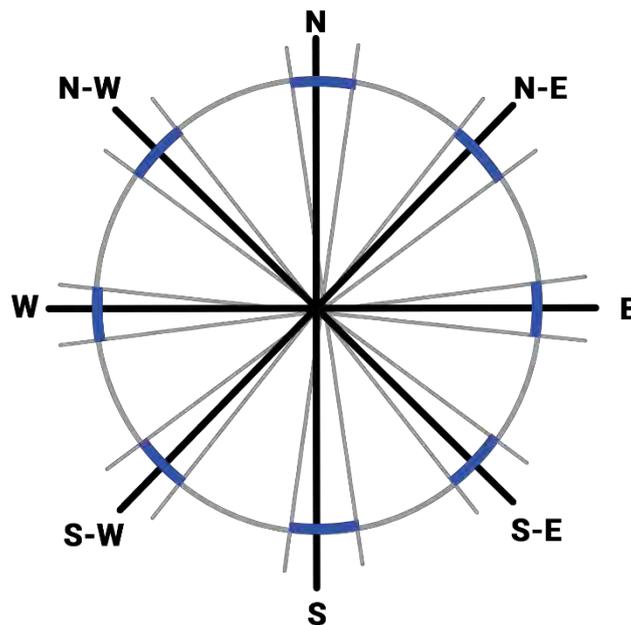


Figure 35. Déviation entre la direction de l'inclinaison demandée et celle réalisée.

Notre étude sur l'amplitude du geste révèle que le couvercle Hémisphérique permet une plus grande amplitude d'inclinaison (Figure 34). De plus, les amplitudes de rotation varient selon la direction (Gauche et Droite) et sont plus grandes avec la posture Squeeze. En ce qui concerne la précision des gestes, l'inclinaison permet 8 directions précises (Figure 35) et les translations induisent des rotations et inclinaisons involontaires du

dispositif. Étant donné les résultats de fatigue pour Squeeze, nous avons éliminé cette prise en main des études suivantes. Sur la base de ces résultats, nous avons défini un seuil minimal d'inclinaison de 12° afin d'éviter tout risque d'activation involontaire des inclinaisons. Afin de tirer pleinement parti des propriétés de l'inclinaison, nous souhaitons avoir une amplitude d'inclinaison utilisable de 30°. Les amplitudes d'inclinaison des deux versions de la RPM étant de 39° et 37° en moyenne, nous les avons légèrement étendues pour atteindre le seuil de 42° : les utilisateurs inclineront confortablement la RPM de 26° en moyenne et pourront poursuivre l'inclinaison de 4° supplémentaires avec un confort moindre.

L'objectif premier de la RPM étant de remplacer la souris traditionnelle tout en conservant ses capacités, nous avons choisi d'étudier les capacités de pointage de notre dispositif. Ainsi, l'étude suivante vise à comparer les performances de pointage de notre dispositif avec celle de la souris traditionnelle.

## 5. ÉTUDE 2 : POINTAGE 2D

Dans cette deuxième étude, nous analysons l'utilisation de la RPM comme dispositif de pointage 2D. Notre dispositif ayant une base hémisphérique, et par conséquent non stable, nous proposons en premier lieu un algorithme correctif permettant d'améliorer la précision du pointage. Puis, nous comparons les capacités de pointage de la RPM avec un prototype de la souris, basée sur la même technologie de localisation que la RPM pour éviter tout biais technologique. Bien que les résultats de cette comparaison ne puissent pas être généralisés à une souris traditionnelle, ils nous permettent de cerner l'impact de la forme de la RPM sur ses capacités de pointage 2D. L'utilisation d'une souris traditionnelle était impossible car les résultats obtenus auraient pu être liés à la différence de technologie entre la RPM et le capteur laser d'une souris, rendant ainsi toute analyse statistique des résultats erronée.

### 5.1. ALGORITHME CORRECTIF

La première étude a révélé l'existence d'inclinaisons involontaires lorsque le dispositif est déplacé. Pour la tâche de pointage 2D, cela peut poser un problème de précision : lorsque la RPM est inclinée, le centre du dispositif (localisé par le système de caméra) se déplace et ainsi le pointeur, visible sur l'écran, est translaté, Figure 36). Il est important de noter que ce problème est inhérent à la morphologie du dispositif et non à la technique d'interaction utilisée ou au système de localisation : que l'on considère le centre de l'appareil ou le point de contact avec la surface, le problème existe.

Pour rendre le pointage 2D plus précis, nous avons implémenté un algorithme de « correction d'inclinaison ». Cet algorithme consiste à calculer la position  $P_0$  (représentée par le point de contact au repos du dispositif avec une surface sous-jacente), lorsque le dispositif est incliné dans une position  $P_1$  (Figure 36). L'objectif est de retrancher la longueur de l'arc qui sépare  $P_1$  de  $P_0$ . Pour calculer  $P_0$ , nous utilisons la position actuelle du dispositif  $P_1$ , les angles de roulis et de tangage  $a$  et  $b$  (représentés par l'inclinaison), et le rayon du dispositif  $R$  comme suit:

$$P_0.x = P_1.x - \pi R * (\sin(a) + \cos(b))$$

$$P_0.y = P_1.y - \pi R * (\cos(a) + \sin(b))$$

Dans notre implémentation, lorsque nous utilisons la RPM comme une souris, nous appliquons l'algorithme pour chaque donnée reçue afin de corriger l'écart induit par

l'inclinaison. Nous calculons ensuite le déplacement et appliquons la fonction de transfert du système d'exploitation utilisé [27] pour définir la position du pointeur.

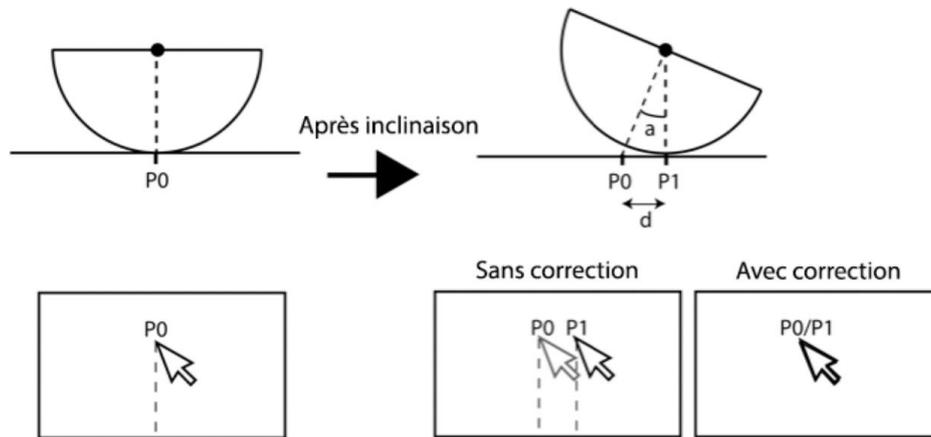


Figure 36. Problème du pointage 2D lié à l'utilisation d'un dispositif inclinable.

## 5.2. DESCRIPTION DE L'ETUDE

La RPM étant une souris augmentée, il était important d'évaluer l'impact de la forme du dispositif sur sa capacité de pointage. Nous avons donc conduit une étude dans laquelle nous comparons la RPM avec notre prototype de souris dans une tâche de pointage 2D classique.

### Tâche et instructions

Nous avons proposé une tâche de pointage inspirée de la norme ISO 9241-9: huit cercles étaient répartis équitablement sur un cercle support (Figure 37). La taille des cercles et la taille du cercle support définissent un indice de difficulté (ID). Au démarrage, une cible parmi les huit était affichée d'une couleur différente. Ce cercle représentait la cible de démarrage de l'essai. La tâche de pointage débutait lorsque le participant sélectionnait le cercle de départ (en rouge). À partir de cette première sélection, la cible de départ n'était plus colorée et l'essai débutait. Une seconde cible, diamétralement opposée à la première et colorée en jaune signalait le cercle de destination. Le participant devait déplacer son curseur dans ce cercle et le sélectionner le plus rapidement et précisément possible. Lorsque le participant faisait une erreur de pointage, la sélection n'était pas acceptée et le participant devait de nouveau sélectionner la même cible. Les participants ont été invités à exécuter la tâche avec leur main dominante aussi rapidement et précisément que possible. Afin de s'assurer que tous les participants prenaient en main le dispositif de manière identique, nous avons imposé la prise en main Lay.

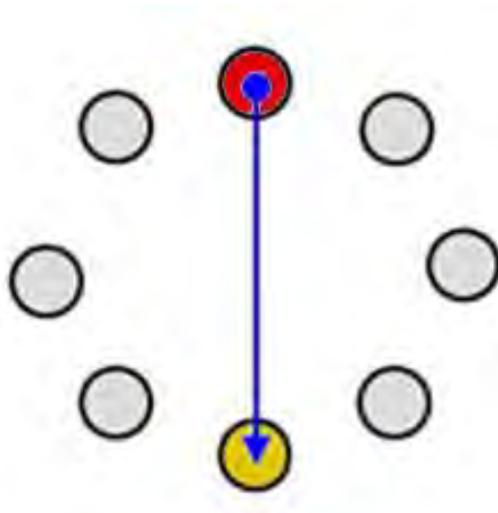


Figure 37. Représentation schématique du cercle support et des huit cercles cibles.

### Matériel

Nous avons utilisé le même système de localisation que dans l'étude précédente (12 caméras OptiTrack) pour suivre les mouvements de la RPM. Pour éviter tout facteur de confusion de la technologie de localisation, nous avons décidé d'intégrer cette technologie dans une souris optique : nous avons équipé la souris de marqueurs infrarouges. Notre réglage n'a pas modifié le poids initial de la souris (81 gr), similaire aux deux RPM (80 gr). Nous avons utilisé le même gain pour tous les périphériques : le gain par défaut de Windows [27]. Le mécanisme de sélection utilisé avec la RPM et la souris est basé sur l'appui de la barre d'espace d'un clavier. Le participant était assis à 50cm d'un écran 15 '(1280x1024px) sur une chaise fixe. Le logiciel de l'expérimentation a été développé en C++ en utilisant la bibliothèque SFML.

### Participants

Nous avons recruté 15 participants (tous membre d'un laboratoire universitaire), dont 3 femmes, âgés de 24,5 ans en moyenne (écart-type = 3.9). Tous étaient droitiers.

### Protocole expérimental

Cette étude suit un protocole 5x5 intra-sujets avec la technique d'interaction (TI) et l'indice de difficulté (ID) comme facteurs. Cinq techniques d'interaction ont été considérées: la souris et 4 utilisations différentes de la RPM résultant des combinaisons de 2 Versions (Convexe, Hémisphérique) et de 2 Prises en main (Lay, Touch). Les valeurs d'ID sont 2.32, 2.74, 3.63, 4.08 et 5.35. Ces valeurs ont été choisies afin de représenter

un ensemble de difficultés réparties entre les indices de difficulté 2 et 6 définis comme bornes. Les cinq techniques en entrée ont été contrebalancées entre les participants au moyen d'un carré latin de  $5 \times 5$ . L'étude était composée de 5 blocs, chaque bloc correspondant à une technique d'interaction. Chaque bloc était constitué de 8 tâches de pointage pour chaque ID. La tâche de pointage commençait à partir de l'un des 8 cercles et se terminait sur le cercle opposé. Les indices de difficulté étaient ordonnés de manière aléatoire à l'intérieur de chaque bloc. Chacun des 15 participants a complété un total de 5 techniques d'interaction x 5 indices de difficulté x 8 tâches de pointage = 200 essais, soit un total de 3000 essais enregistrés. Chaque session a duré en moyenne 48 minutes.

Les participants ont, en premier lieu, réalisé un bloc d'entraînement avec chaque technique d'interaction avant de commencer. Après chaque bloc, i.e. après avoir terminé toutes les tâches de pointage avec une technique d'interaction, les participants ont rempli une échelle de Borg pour évaluer la fatigue ressentie. À la fin de l'expérimentation, les participants ont rempli un questionnaire SUS pour chaque version de la RPM et un pour la souris, puis, ont ordonné les cinq techniques d'interaction par ordre de préférence.

Données collectées et analyse statistique

Les données enregistrées comprenaient le temps de réalisation de la tâche (ms) et le taux d'erreur (pourcentage de cibles manquées). Nous avons utilisé la même approche d'analyse statistique que dans l'étude précédente. Nous présentons les principaux résultats ci-après, l'analyse statistique détaillée se trouve dans les annexes section 1.2.

### 5.3. RESULTATS

Cette étude démontre qu'il n'y a globalement pas de différence significative en termes de performance (temps, erreur, débit) dans une tâche de pointage 2D entre notre souris et la RPM hémisphérique (Figure 38). La RPM Convexe était la technique préférée des utilisateurs. Cependant, le mécanisme de sélection et la technologie de localisation utilisée n'étant pas celles d'une souris traditionnelle (précision, fréquence de rafraichissement), ces résultats ne permettent pas d'affirmer que la RPM possède une performance de pointage analogue à celle d'une souris traditionnelle. En revanche, le modèle prédictif obtenu, proche de celui d'une souris traditionnelle, permet de supposer que la RPM permet un pointage performant.

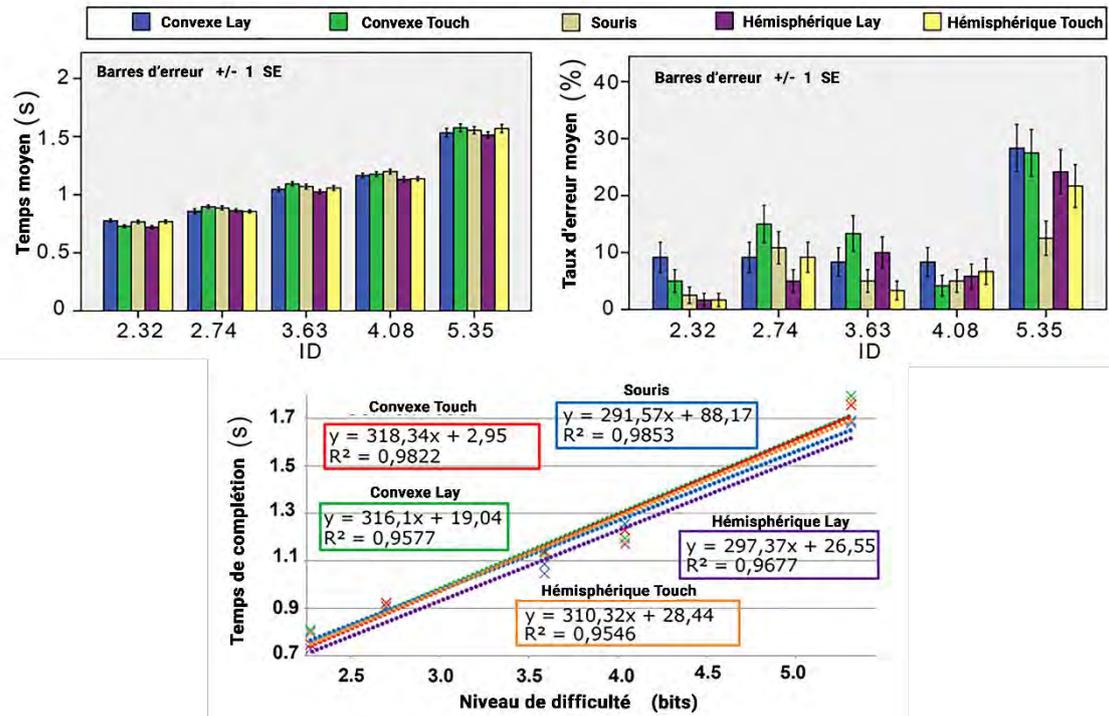


Figure 38. Temps de réalisation moyen (gauche), taux d'erreur (centre) pour chaque technique d'interaction et indice de difficulté. Modèle de Fitts pour chaque technique d'interaction (droite).

## 6. ÉTUDE 3 : MANIPULATION 3D

Profitant des avantages clés de la RPM pour la manipulation 3D, nous proposons une technique d'interaction combinant l'inclinaison et la translation afin effectuer des translations d'objets 3D. Nous proposons aussi d'utiliser l'inclinaison et la rotation de la RPM afin d'effectuer des rotations d'objets 3D. Nous évaluons notre solution en la comparant à un dispositif commercial dédié à la manipulation 3D, le SpaceMouse (ou Souris 3D, Figure 39).



*Figure 39. La SpaceMouse.*

### 6.1. TECHNIQUES D'INTERACTION POUR LA MANIPULATION 3D

La tâche de manipulation 3D nécessite 6 DDL, or nous n'utilisons que 5 DDL de la RPM, notre technique d'interaction repose donc sur une utilisation modale : la translation et la rotation sont effectuées séparément. Comme RPM offre 2 DDL pour la translation, 2 DDL pour l'inclinaison et 1 DDL pour la rotation, toute tâche nécessitant 3 DDL sera effectuée en utilisant une combinaison d'au moins deux de ces trois gestes (p. ex. une translation pour les axes X et Y et une inclinaison pour l'axe Z). De plus, des travaux précédents ont montré que dans des conditions d'utilisation où les deux tâches (translation et rotation 3D) peuvent être réalisées simultanément, les utilisateurs réalisent très fréquemment les tâches de manières indépendantes [115]. Nous avons donc choisi de séparer ces deux tâches dans notre étude.

## Translation 3D : Inclinaison et translation

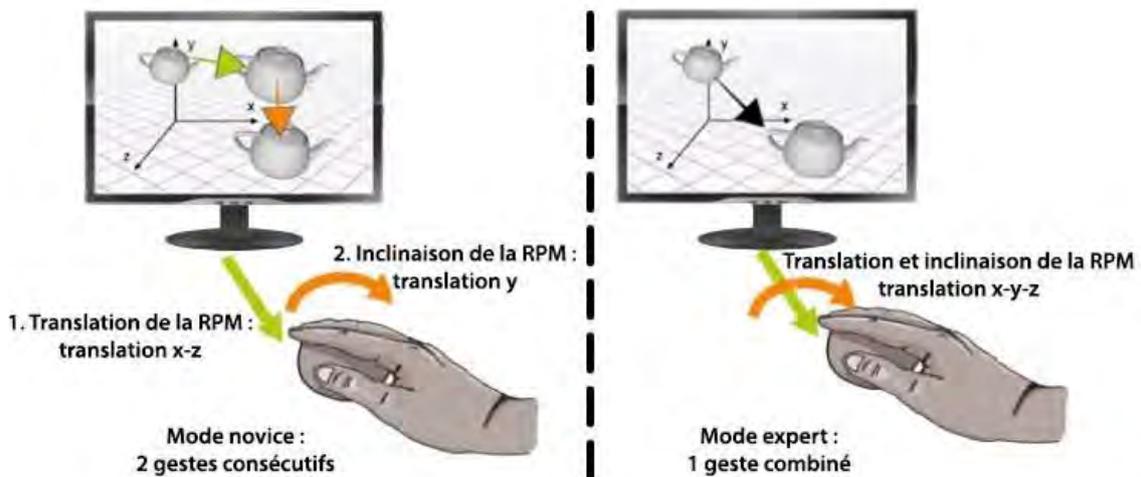


Figure 40. Technique d'interaction pour la translation en 3D : mode novice en deux gestes (gauche) et mode expert en 1 geste combiné (droite).

Pour effectuer une translation 3D d'un objet, nous utilisons une technique d'interaction basée sur le principe du contrôle de la position: le déplacement physique 2D de la RPM est associé à la translation 2D du curseur sur le plan x-z de la scène 3D (Figure 40). L'inclinaison avant / arrière du dispositif contrôle l'élévation du plan x-z, c'est-à-dire la position y du plan (hauteur). Cette technique offre l'avantage d'être facile à utiliser par les débutants et permet une transition fluide vers une utilisation par des experts. En effet, les débutants peuvent décomposer la tâche de translation 3D en effectuant d'abord un mouvement 2D sur le plan x-z, puis une inclinaison pour définir la hauteur (axe y). Les utilisateurs experts peuvent combiner simultanément déplacement et inclinaison dans le même geste et effectuer directement la translation 3D.

## Rotation 3D : Inclinaison et rotation

Pour effectuer des rotations 3D, deux modes sont possibles : contrôle de position ou contrôle de vitesse. En contrôle de position, la rotation et l'inclinaison de la RPM sont directement associées à l'orientation 3D de l'objet (Figure 41). Ce mode nécessite un mécanisme de « clutching » car des tests informels ont montré que pour être confortablement et efficacement exploitée, l'inclinaison de la RPM dans sa zone de confort ( $12^\circ$  à  $42^\circ$ ) ne peut être associée qu'à une rotation maximale de  $\pm 90^\circ$  dans l'environnement virtuel. Ainsi, toute rotation 3D supérieure à cette valeur doit être réalisée en plusieurs étapes : activation du mouvement, inclinaison du dispositif, désactivation puis remise en place du dispositif dans son état initial, activation du

mouvement, inclinaison du dispositif, etc. En contrôle de vitesse, les angles d'inclinaison et de rotation de la RPM sont associés aux vitesses de rotation de l'objet 3D dans ces directions. Ainsi, incliner le dispositif légèrement dans une direction induit une rotation lente de l'objet dans cette même direction et une inclinaison plus élevée induit une rotation plus rapide. Des tests informels, réalisés auprès de 4 utilisateurs, ont montré que le contrôle de position était plus performant que le contrôle de vitesse car il offre une manipulation directe de l'objet 3D, ce qui semble préféré par les utilisateurs.

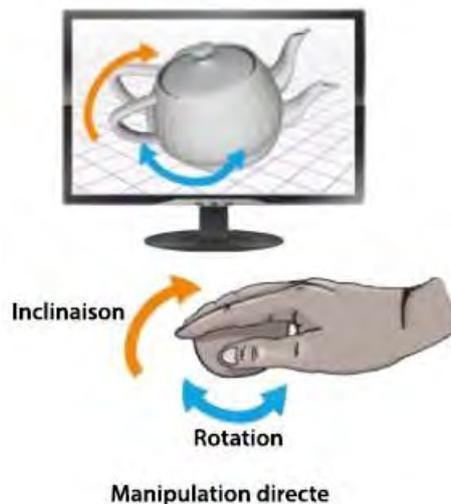


Figure 41. Technique d'interaction pour la rotation 3D.

## 6.2. DESCRIPTION DE L'ETUDE : TRANSLATION ET ROTATION 3D

Dans cette étude, nous comparons les performances de la RPM avec un dispositif dédié à l'interaction 3D, la SpaceMouse [173], en deux tâches 3D séparées : pointage (translation) et rotation [115]. Nous ne comparons pas notre technique d'interaction avec la souris traditionnelle car il a été démontré qu'elle était moins performante que la SpaceMouse pour l'interaction 3D [49].

### Tâche et instructions

Nous avons décidé de mener deux sessions, chacune consacrée à une tâche 3D : la première session consistait à effectuer une translation 3D (tâche de pointage 3D), et la seconde une rotation 3D. La raison pour laquelle nous avons divisé notre expérimentation en deux sous-tâches, au lieu d'une seule tâche d'ancrage 6D [115, 206], est que le nombre de conditions pour chacune des tâches était très grand et une combinaison des deux aurait rendu l'expérimentation excessivement longue, induisant des effets de fatigue et d'apprentissage conséquents. D'autre part, nous avons voulu étudier séparément les

performances de notre appareil pour la translation et la rotation 3D car ces deux tâches ne sont pas toujours nécessairement combinées. Ainsi l'analyse des performances de chaque sous-tâche aurait été complexifiée.

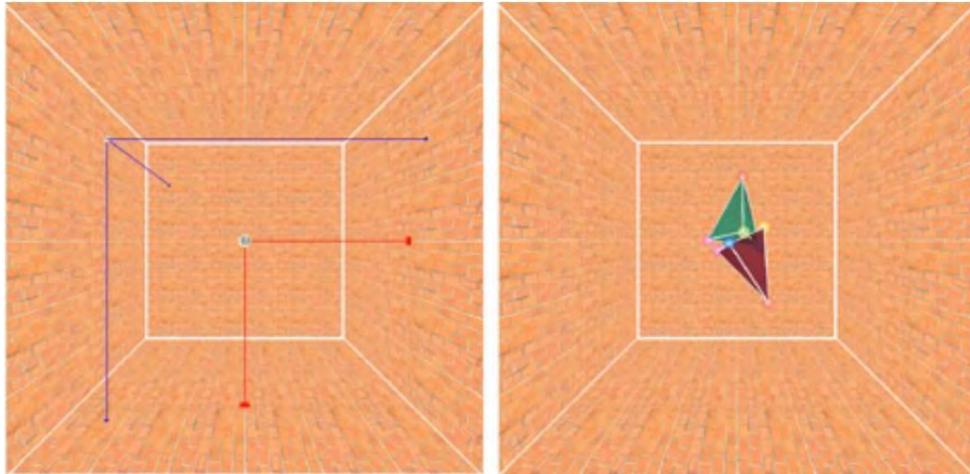


Figure 42. Environnement 3D pour la tâche de pointage (gauche) et pour la tâche de rotation (droite).

Dans la tâche de pointage 3D, le participant doit atteindre une cible sphérique 3D dans une scène 3D en faisant une translation d'un curseur (Figure 42 - Gauche). La cible est affichée au centre d'un cube qui délimite l'environnement 3D. La position initiale du curseur est située dans 8 emplacements différents (combinant des translations en x, y et z) à deux distances différentes de la cible (12.99 et 8.66, ces distances n'ont pas d'unité spécifique. Elles représentent une distance dans la scène 3D). Nous avons mis en place deux cibles de différentes tailles (0.5 et 1.7) pour produire 4 Indices de difficulté (2.6, 3,11, 4,19 et 4,75 bits). Dans la tâche de rotation 3D, il était demandé à l'utilisateur de faire pivoter un curseur tétraédrique jusqu'à ce qu'il s'adapte à l'orientation de la cible tétraédrique (Figure 42 - Droite). L'orientation initiale est une combinaison non-triviale de deux ou trois axes de rotations donnant 8 orientations différentes. Dans les deux tâches, l'utilisateur démarre et valide un essai en appuyant sur la barre d'espace sur le clavier (le dispositif ne possédant pas de mécanisme de sélection). Pour activer la manipulation (clutching) avec la RPM l'utilisateur devait appuyer sur la barre d'espace. En effet, nous avons pensé qu'ajouter une seconde touche du clavier pour activer la manipulation pourrait surcharger les capacités cognitives de l'utilisateur et nuire à l'expérimentation. En conséquence, nous n'avons pas mesuré les erreurs de sélection pour la rotation 3D puisque l'activation des manipulations et la validation de l'essai utilisaient la même touche du clavier. Il était demandé aux participants de réaliser la tâche en utilisant la RPM avec leur main dominante, aussi rapidement et précisément que possible.

## Matériel

Nous avons utilisé le même système de localisation de la RPM que dans l'étude précédente (système de caméras OptiTrack). Pour définir la valeur de gain de la technique d'interaction basée sur la SpaceMouse, nous avons effectué un certain nombre de tests auprès d'utilisateurs. Certains ont trouvé le gain classique trop élevé pour manipuler précisément l'objet 3D. Nous avons donc décidé de tester deux valeurs de gain dans notre étude : un gain similaire au comportement par défaut de la SpaceMouse (idéal pour la plupart des utilisateurs après un peu d'entraînement) et un plus petit gain favorisant la précision sur la vitesse (mieux pour la plupart des débutants). Avec la RPM, nous avons utilisé un facteur de gain de 1:1 pour les translations et de 1:3 pour les rotations et inclinaisons (i.e. une inclinaison de 30° induisait une rotation de 90° dans l'environnement 3D). Le logiciel de l'expérimentation a été développé en C++ et utilisait le Moteur 3D Irrlicht. Un écran affichant l'application et relié au système de caméra était situé à 50 cm de l'utilisateur. L'utilisateur était assis sur une chaise, le dispositif d'interaction placé entre lui et l'écran.

## Participants

Nous avons recruté 12 participants pour cette étude (tous membres d'un laboratoire universitaire), dont 6 femmes, âgés en moyenne de 28.5 ans (écart-type = 6) et tous droitiers. Parmi ceux-ci, 5 étaient familiarisés à l'interaction 3D et 3 avaient déjà participé aux études précédentes.

## Protocole expérimental

L'étude suit un protocole 4x4 intra-sujet avec la Technique d'interaction (RPM Hémisphérique, RPM Convexe, SpaceMouse à petit gain et SpaceMouse gain par défaut) et l'Indice de difficulté (2.60, 3.11, 4.19 et 4.75) comme facteurs pour chacune des deux tâches (translation et rotation 3D). Ces tâches ont été réalisées dans deux sessions consécutives. Chaque session était divisée en 4 blocs, chacun correspondant à une technique d'interaction. L'ordre des blocs est contrebalancé à travers les participants au moyen d'un carré latin 4x4. La position initiale du curseur et l'indice de difficulté (translation 3D) ainsi que l'orientation initiale (rotation 3D) ont été ordonnés aléatoirement dans chaque bloc.

En premier lieu, les utilisateurs ont participé à une séance d'entraînement pour les deux tâches : les utilisateurs devaient effectuer 30 essais par tâche et par technique d'interaction avant l'expérimentation. Au cours de l'expérimentation, chaque bloc était précédé de 8 essais d'entraînement avec la technique concernée afin que les participants se remémorent le fonctionnement de la technique d'interaction. Aucune contrainte n'a été donnée en ce qui concerne la prise en main de la RPM car nous ne voulions pas contraindre la prise en main de la SpaceMouse.

#### Données collectées et analyse statistique

Nous avons enregistré toutes les données de localisation de la RPM (position et orientation) ainsi que le temps d'exécution de la tâche. Nous avons également recueilli les préférences des utilisateurs et l'utilisabilité de chaque technique en utilisant deux échelles de Likert à 5 points ainsi que la fatigue en utilisant une échelle de Borg allant de 6 à 20. Pour la tâche de pointage 3D, nous avons collecté 384 essais par utilisateur (4 techniques d'interaction x 8 directions x 4 indices de difficulté x 3 répétitions) x 12 utilisateurs = 4608 essais au total. Cette 1ère session durait en moyenne 73 minutes. Pour la tâche de rotation 3D, nous avons collecté 192 essais par utilisateur (4 techniques d'interaction x 8 orientations x 6 répétitions) x 12 utilisateurs = 2304 essais au total. Cette 2ème session durait en moyenne 55 minutes. Nous avons utilisé la même approche d'analyse statistique que dans les études précédentes. Seuls les résultats principaux sont présentés ci-après. Une analyse statistique détaillée se trouve dans les annexes section 1.3.

### 6.3. RESULTATS

Cette étude démontre qu'il y a une différence significative en termes de temps de pointage (translation 3D) entre les deux versions de la RPM et les deux versions de la SpaceMouse (Figure 43 - Gauche). RPM est 31% plus rapide que la SpaceMouse. Concernant la tâche de rotation 3D, il n'y a pas de différence significative entre les différents dispositifs et gains de la SpaceMouse en termes de temps de réalisation. Concernant la préférence utilisateur, les deux versions de la RPM sont considérées comme plus utilisables et sont évaluées plus positivement que les deux versions de la SpaceMouse (Figure 43 – Droite). Dans l'ensemble, cette étude démontre que la RPM peut être utilisée efficacement comme dispositif d'entrée pour la translation et la rotation 3D et que les utilisateurs préfèrent (appréciation générale et utilisabilité) la RPM à la SpaceMouse.

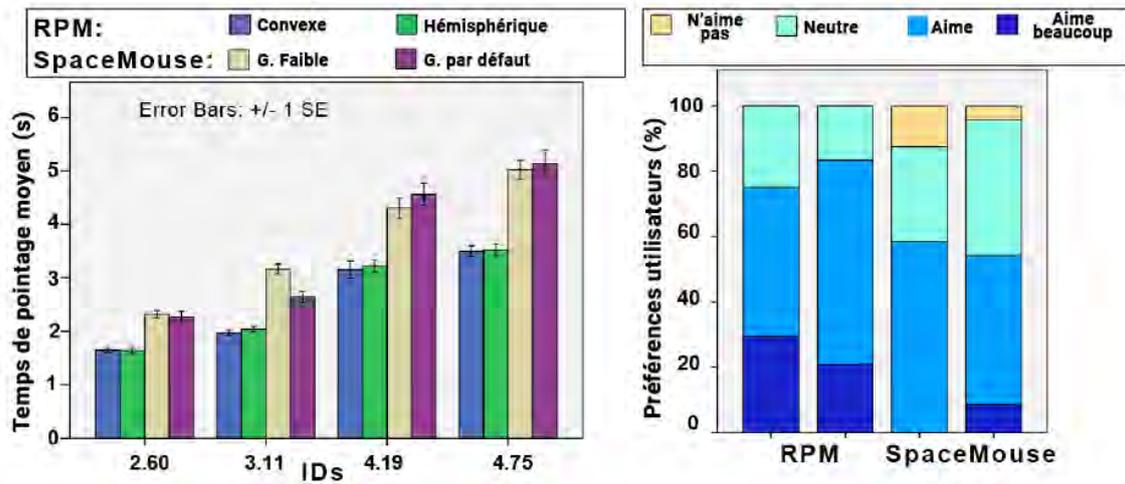


Figure 43. Temps de réalisation de la tâche de pointage 3D (gauche) pour chaque technique d'interaction et indice de difficulté. Préférence utilisateur pour chaque technique d'interaction (droite).

## 7. TRANSFERT DANS LE CONTEXTE DE L'ENTREPRISE

Bien que les logiciels de Berger-Levrault ne contiennent pas d'environnements 3D, ils sont particulièrement complexes et nécessitent également la manipulation de plusieurs dimensions en parallèle. Il est ainsi envisageable d'utiliser la RPM pour faciliter l'accès et la manipulation de ces dimensions. Dans ce but, nous avons choisi d'illustrer le transfert possible de notre dispositif au contexte de l'entreprise à travers un logiciel actuellement commercialisé : Organis'Or

### 7.1. ORGANIS'OR

Ce logiciel, récemment modernisé et renommé BL.Planning, permet la gestion des plannings prévisionnels et effectifs du personnel dans un établissement médico-social. Il intègre l'ensemble des contraintes légales du code du travail et permet une gestion efficace du personnel en fonction des compétences des employés et des services auxquels ils sont affectés. La visualisation de ces informations se fait sous la forme d'un tableur.

L'axe des abscisses est utilisé pour gérer l'échelle de temps et l'axe des ordonnées pour gérer les différents employés et services de l'établissement. Une cellule (représentant une plage horaire du planning d'un salarié) est composée d'un code alphanumérique coloré représentant le type d'horaire de roulement que le salarié effectue au jour donné ainsi que quatre encoches de couleur (bleu, jaune, vert et rouge) disposées aux quatre coins de la cellule. Ces encoches peuvent être présentes ou non sur chaque cellule et permettent d'alerter l'utilisateur du logiciel (modification récente du planning sur cette tranche horaire, absence d'un membre du personnel, etc.). Le cumul de l'ensemble de ces dimensions dans chaque cellule du tableur rend l'usage du logiciel complexe et fastidieux : la visualisation des informations pertinentes est difficile car de trop nombreuses informations sont affichées simultanément.

En complément, il est possible de sélectionner une case afin d'ouvrir une fenêtre permettant de visualiser les informations de l'employé concerné sur cette plage horaire. L'usage de cette fonctionnalité est fréquent car il permet, entre autres, d'effectuer des remplacements si cela est nécessaire en comparant plusieurs fiches de salariés (vérification des compétences, contraintes légales, horaires, etc.). Ainsi, les utilisateurs sélectionnent plusieurs cellules afin de comparer les paramètres de plusieurs employés avant d'effectuer le remplacement. Ces multiples sélections rallongent le chemin

d'interaction car les fenêtres de visualisation doivent à chaque fois être ouvertes puis fermées afin de pouvoir sélectionner une nouvelle cellule.

## 7.2. UTILISATION ENVISAGEE DE LA RPM

L'utilisation de la RPM dans ce contexte permettrait d'apporter deux bénéfices majeurs. D'une part, les inclinaisons de la RPM permettraient d'activer ou de désactiver la visualisation d'un certain type d'encoche, ce qui réduirait la surcharge visuelle des utilisateurs : une inclinaison vers le Nord-Est permettrait d'afficher sur le tableur toutes les encoches situées dans le coin haut-droit des cellules. Ainsi, la surcharge d'affichage en serait amoindrie.

D'autre part, une fois les salariés pertinents sélectionnés, il serait possible d'itérer parmi l'ensemble de leurs fiches à l'aide des rotations de la RPM. Une inclinaison vers l'arrière permettrait de supprimer un employé de la liste de sélection et une inclinaison vers l'avant (lorsque le curseur est sur une cellule donnée), permettrait d'ajouter l'employé à la liste. L'utilisation de la RPM permettrait une navigation simplifiée et plus rapide dans les fiches des employés afin de réaliser le remplacement souhaité, réduisant ainsi la longueur du chemin d'interaction.

## 7.3. INTEGRATION

Afin d'intégrer la RPM au logiciel Organis'Or, de nombreuses étapes sont encore nécessaires. D'abord, une version plus intégrée et précise de la RPM serait souhaitable afin de pouvoir remplacer efficacement la souris traditionnelle sans un coût trop important pour les clients de l'entreprise. Ensuite, l'utilisation de la RPM dans Organis'Or nécessite d'intégrer nos algorithmes de détection et correction au cœur de l'application. Ce logiciel étant particulièrement dense, riche et complexe, cette intégration nécessite un investissement humain élevé, non adapté à ce contexte de travail de thèse. Cependant, avec la refonte progressive des logiciels de l'entreprise visant à tirer parti des nouvelles technologies, les travaux que nous présentons dans ce chapitre pourront faire partie d'un processus d'intégration dans les nouvelles versions des logiciels Berger-Levrault (suite à des tests d'acceptabilité qui devront être réalisés auprès des utilisateurs).

## 8. DISCUSSION ET PERSPECTIVES

### 8.1. LIMITATIONS DE NOS ETUDES

Notre exploration a été limitée par l'utilisation de marqueurs sur la RPM, par la technologie de suivi, par le manque de mécanisme de sélection et de « clutching ». L'utilisation des caméras OptiTrack nous a permis de mesurer avec précision les amplitudes maximales de chaque geste. En contrepartie, nous devons soigneusement placer les marqueurs infra-rouges sur le dispositif. Même si ces marqueurs ont pu avoir un effet mineur sur les gestes de l'utilisateur, il serait nécessaire de confirmer nos résultats en reproduisant nos expérimentations avec une technologie de localisation précise totalement embarquée. Cependant, il n'existe pas à notre connaissance de technologie embarquée permettant une localisation précise en position et en orientation de la RPM. En outre, nous n'avons pas évalué différents mécanismes de sélection ou de « clutching » : ceux-ci pourraient améliorer ou nuire à nos techniques d'interaction et une étude de ces mécanismes serait pertinente.

Concernant la taille de la RPM, elle est actuellement basée sur la taille d'une souris traditionnelle. Il pourrait être intéressant d'explorer un prototype plus petit de notre dispositif avec lequel les manipulations seraient différentes : l'utilisateur pourrait tenir le dispositif dans sa main plutôt que de le poser sur la table.

Concernant le pointage 2D, une étude comparant notre dispositif avec une souris traditionnelle serait nécessaire afin de valider nos résultats expérimentaux. D'autre part une étude à long terme sur le confort du dispositif pourrait être intéressante (il existe des souris ergonomiques réduisant la fatigue des articulations). Enfin, une étude sur l'acceptabilité de notre dispositif pourrait être pertinente. En effet, la souris est un dispositif traditionnel, connu, utilisé et maîtrisé par une majorité d'utilisateurs. La remplacer par la RPM pourrait nécessiter un temps d'adaptation et engendrer des réticences.

### 8.2. CONSEILS DE CONCEPTION POUR DISPOSITIFS INCLINABLES

Nos études nous ont permis d'avoir un regard critique sur la conception d'un dispositif inclinable et ainsi de proposer des directives de conception pour de tels dispositifs que nous résumons ci-après. Le couvercle hémisphérique permet une amplitude d'inclinaison plus élevée. Lors de l'association entre les actions logicielles et les gestes d'inclinaison

et de rotation, il est important de prendre en compte l'orientation du geste. À titre d'exemple, les rotations à droite et à gauche n'ont pas la même amplitude. Les gestes d'inclinaison peuvent être réalisés dans 8 directions distinctes avec un seuil d'activation minimal de  $12^\circ$  afin d'éviter les faux positifs et permettre une amplitude maximale de  $42^\circ$ . Pour le pointage 3D, l'utilisation de gestes composés semble être la plus efficace. Pour la rotation 3D, le contrôle de position pourrait être couplé avec un contrôle de vitesse comme dans les travaux de Tsandilas et al. [180].

### 8.3. INSTABILITE DU DISPOSITIF

Contrairement à la souris traditionnelle ou aux dispositifs arrondis précédents tels que la Rockin'Mouse, qui ont une partie basse plate, celle de la RPM est hémisphérique : le dispositif n'est donc pas stable. Ceci se manifeste particulièrement dans deux cas : lorsque le dispositif est relâché, il oscille ; lorsqu'il est déplacé, il induit des inclinaisons involontaires. Dans le premier cas, une solution simple est de détecter le moment où l'utilisateur prend le dispositif en main. L'appareil peut ainsi être éteint lorsqu'aucun contact avec la main de l'utilisateur n'est détecté (à l'aide d'une surface tactile, par exemple). Dans le deuxième cas, lorsque l'utilisateur déplace le dispositif comme une souris sur une surface plane, un seuil approprié (p. ex.  $12^\circ$ ) doit être utilisé pour éviter les faux positifs dus à une inclinaison involontaire. Enfin, pour le pointage 2D, un algorithme correctif tel que celui que nous proposons dans la section 5.1 peut être utilisé pour éviter les déplacements involontaires du curseur.

### 8.4. AXE Z

Nous n'avons pas exploré cette dimension dans nos travaux parce que nous pensons que celle-ci induit une fatigue élevée. Le dispositif ayant pour but de remplacer une souris traditionnelle, il ne nous semblait pas envisageable d'utiliser cette dimension pour un usage à longue durée. Cependant une utilisation sporadique est possible, par exemple en maintenant la RPM sur la paume pendant une présentation. Dans ce contexte, la RPM pourrait être utilisée pour effectuer des gestes en l'air afin de contrôler des diapositives. L'usage de cette dimension a été partiellement exploré dans des travaux réalisés suite aux nôtres [157].

## 8.5. DEFIS TECHNIQUES

Construire un prototype final utilisable en contexte réel de la RPM présente plusieurs défis techniques.

*Position* : pour localiser la position de la RPM, plusieurs solutions embarquées ou externes peuvent être envisagées. Les solutions embarquées incluent l'utilisation de capteurs magnétiques (bien que peu précis) ou l'utilisation d'une mini-caméra embarquée comme dans la VideoMouse [78] (nécessite l'utilisation d'une surface d'interaction avec un motif spécifique). Les solutions externes comprennent l'utilisation de caméras infrarouges ou l'utilisation d'une surface conductrice telle qu'une tablette Wacom. Cette seconde solution a été explorée dans des travaux conduits par la suite [46]. Certaines de ces solutions permettent, en complément, de détecter la position sur l'axe Z de notre dispositif.

*Orientation* : pour suivre l'orientation de l'appareil, nous pourrions utiliser une unité de mesure inertielle intégrée (IMU). Cette solution a été utilisée dans plusieurs travaux postérieurs [46, 157].

*Contact avec la surface support* : pour permettre le mécanisme de « clutching », les solutions embarquées comprennent l'utilisation d'une surface basse résistive ou conductrice, d'un capteur de distance ou encore d'une caméra comme dans VideoMouse [78]. Ce contact peut être facilement détecté à l'aide d'une surface capacitive sous-jacente.

*Contact avec la main* : Pour détecter la prise en main du dispositif, une surface capacitive pourrait être utilisée. Celle-ci permettrait d'éviter les problèmes d'instabilité liés à l'oscillation du dispositif une fois celui-ci relâché.

*Mécanisme de sélection* : il existe deux options possibles pour intégrer un mécanisme de sélection toujours disponible indépendamment de la rotation du dispositif : un bouton 'anneau' tout autour et un bouton capacitif sur la partie haute de la RPM. L'utilisation d'un bouton en forme d'anneau, relié via un algorithme à la rotation du dispositif, pourrait être envisagé. Ce type de bouton, intégré dans notre prototype embarqué de la RPM, permet de réaliser des clics autour du dispositif. Nous l'avons utilisé pour proposer un clic gauche et droit, mais une utilisation plus complexe avec de plus nombreux boutons est possible. Une seconde solution, plus élégante, serait d'utiliser une surface multi-touch.

Cependant, cette solution manquerait de retour haptique, comme sur un bouton régulier. Cette solution a déjà été utilisée pour étendre les capacités d'interaction d'une souris traditionnelle [14, 185, 204] tels que dans la Magic Mouse d'Apple dans lequel il est combiné avec un interrupteur mécanique. Le principal défi de cette solution serait de distinguer un toucher du doigt d'un contact avec la paume ou avec un geste de préhension / compression.

Stabilité de la sélection : le dispositif ayant une base arrondie, l'appui sur un bouton pourrait engendrer des mouvements involontaires (inclinaiions). Pour assurer la stabilité de RPM, un mécanisme de sélection autour du dispositif plutôt que sur le dessus (tel que le bouton en anneau) permettrait de limiter les inclinaiions involontaires : la pression du doigt s'exerce parallèlement à la surface de contact de la RPM, réduisant ainsi les potentiels mouvements involontaires.

Repositionnement de l'appareil : même si le centre de gravité bas du dispositif permet le repositionnement automatique, l'appareil oscille lorsqu'il est relâché. Pour limiter cette oscillation, notamment lorsque l'utilisateur passe de la souris au clavier, une solution basée sur un mécanisme d'auto-équilibrage couplé avec une unité de mesure inertielle pourrait être intégrée. Ainsi, des moteurs internes pourraient éviter une oscillation trop importante de la RPM avant de retrouver une position stable. Le principal défi de cette solution serait de réussir à intégrer l'ensemble du matériel nécessaire à l'intérieur du volume de la RPM. Ce défi a été relevé, dans un contexte d'utilisation différent, avec Sphero [174], une balle connectée. Celle-ci est composée de moteurs internes qui permettent à la balle de se déplacer dans l'espace. Elle est équipée de capteurs d'orientation ainsi que d'un module Bluetooth qui permettent la communication avec un smartphone pour piloter la balle ou contrôler un mini-jeu lancé sur le smartphone. De plus, une version alternative de Sphero, basée sur le modèle du robot BB8 de Star Wars possède une demi-sphère complémentaire placée sur le couvercle du robot représentant sa tête. Lorsque le robot se déplace, la tête reste en position au même endroit de la sphère support. En s'inspirant de ces technologies, il est ainsi possible de réduire les oscillations de la RPM en compensant ses mouvements/rotations afin de la conserver dans une orientation spécifique (orientation au moment du relâchement du dispositif ou orientation verticale de repos).

## 8.6. AUTRES SCENARIOS D'UTILISATION

Alors que notre motivation principale pour la conception de la RPM était de créer un dispositif tout-en-un pour l'interaction 2D et 3D, nous avons exploré d'autres scénarios d'utilisation qui pourraient bénéficier des propriétés de notre dispositif. Nous avons réalisé une séance de design d'une heure avec 15 participants pour collecter des scénarios d'application. Beaucoup de participants ont proposé des cas d'utilisation liés à l'interaction 3D, allant de jeux 3D à l'édition 3D. De nombreuses idées sur une grande variété d'applications ont également été proposées : en montage vidéo ; composition musicale ; dessin graphique ; navigation web ; navigation cartographique ; marking menus ; ou l'exploration de grands ensembles de données. Pour chaque idée, les participants ont associé différentes fonctions aux degrés de liberté RPM. Par exemple, dans une application cartographique, la translation est utilisée pour déplacer la vue et l'inclinaison pour effectuer des gestes des rotations (rotation gauche-droite) ou pour gérer le niveau de zoom (inclinaison avant-arrière). Dans une application de dessin, les translations sont associées au dessin, les rotations à la sélection des couleurs et l'inclinaison à l'épaisseur et au type de trait. Cette session de design illustre la gamme possible d'applications d'utilisation de la RPM.

Dans un second temps, nous avons intégré la RPM à certaines de ces applications. Pour ce faire, nous avons simulé des clics et déplacements de la souris à l'aide d'un algorithme développé en C++ et relié à la RPM. De plus, afin d'activer des fonctionnalités de l'application, nous avons simulé l'utilisation de raccourcis clavier lors des mouvements de la RPM. À titre d'exemple, l'inclinaison vers l'arrière de la RPM produisait le raccourci CTRL+Z dans l'application de dessin. Nous avons intégré la RPM dans trois applications : Google Street View [62], GIMP [58] et un logiciel de gestion de cartes moléculaires d'interaction [126]. Ces applications sont illustrées dans une vidéo associée à l'une de nos publications<sup>3</sup>.

Variantes alternatives et consécutives de la RPM

L'intérêt de notre dispositif pour faciliter l'interaction avec des données à nombreuses dimensions ne s'est pas arrêté à nos études. D'autres travaux se sont basés sur la RPM pour proposer de nouvelles techniques d'interaction. Ainsi, TDome [157] est un dispositif

---

<sup>3</sup> <https://www.youtube.com/watch?v=yu0BNsnA9Gg>

d'interaction adapté aux environnements multi-écrans (Figure 44). Ce dispositif est basé sur la RPM et y ajoute une surface tactile sur le couvercle à l'aide d'une smartwatch ou d'un smartphone. Dans cette variante, la position et la rotation sont calculées à partir d'une centrale inertielle (X-IMU [202]) placée à l'intérieur. À l'aide d'une combinaison de gestes physiques et tactiles, TDome permet de configurer un environnement multi-écrans en sélectionnant et spatialisant les différents écrans, redirige les entrées utilisateurs sur un écran spécifique, permet de transférer du contenu entre plusieurs écrans et peut être utilisé pour entrer des données confidentielles telles qu'un mot de passe.

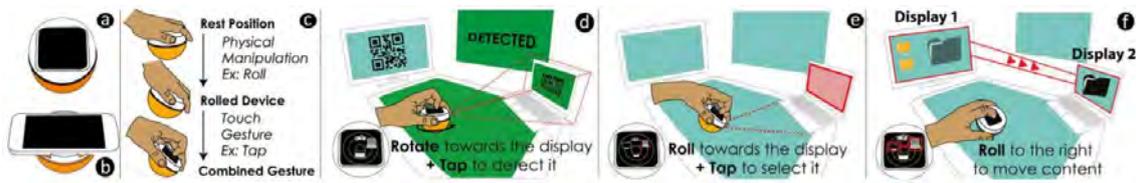


Figure 44. TDome: un dispositif combinant les avantages des modes d'interaction isotoniques et isométriques.

Par ailleurs, une seconde variante de la RPM a été proposée pour fluidifier l'interaction avec des menus linéaires dans une application de bureau [46]. Dans ces travaux, la position de la RPM est détectée à l'aide d'une tablette sous-jacente et la rotation est calculée à partir d'une centrale inertielle intégrée dans le dispositif. Afin de détecter le contact avec la tablette, la partie basse de la RPM a été enduite avec un spray conducteur. D'autre part, la tablette permet aussi d'intégrer un mécanisme de sélection puisque les touchers sur celle-ci permettent la sélection d'un menu ou sous-menu.

Ces deux dispositifs permettent d'illustrer le potentiel prometteur de la RPM dans de nombreux domaines applicatifs différents.

## 9. CONCLUSION DU CHAPITRE 4

Dans ce chapitre, nous avons présenté la conception d'un nouveau dispositif d'interaction, la Roly-Poly Mouse (RPM) combinant les avantages de la souris et des dispositifs 3D en permettant trois types de gestes : translations, inclinaisons et rotations. Pour compenser la base arrondie de la RPM qui cause de l'instabilité lors de la tâche de pointage 2D, nous avons proposé un algorithme de correction de l'inclinaison. Nous avons exploré plusieurs formes, tailles et prises en main fréquemment adoptées du dispositif et identifié celles préférées par les utilisateurs. Sur la base de ces résultats, nous avons exploré l'amplitude des gestes ainsi que les performances du dispositif en pointage 2D et manipulation 3D. Nous en avons tiré des directives de conception et d'utilisation d'un dispositif à base arrondie que nous présentons dans la section 8.2.

Cependant, nous avons pu constater au travers de ce travail que la conception d'un nouveau dispositif adapté à un certain type de données (en l'occurrence, pointage 2D et manipulation 3D) reste une tâche complexe et difficilement reproductible pour fluidifier l'interaction dans tous les contextes. De plus, la conception d'un nouveau dispositif, disposant de degrés de libertés spécifiques, soulève de nombreux challenges technologiques. En conséquence, une approche alternative nous semble pertinente afin de concevoir de nouveaux dispositifs à multiple degrés de libertés : la composition de dispositifs existants. Nous présentons cette approche dans le chapitre suivant au travers d'un espace de conception pour la composition de dispositifs : DECO.

# CHAPITRE 5

## COMBINAISON DE DISPOSITIFS : ESPACE DE CONCEPTION

---

### Structure du chapitre

---

1. INTRODUCTION .....	108
1.1. Motivation.....	108
1.2. Structure du chapitre .....	109
2. APPROCHE DE COMPOSITION : DÉFINITIONS.....	110
3. DECO : UN ESPACE DE CONCEPTION POUR LA COMPOSITION DE DISPOSITIFS.....	111
3.1. Axe 1 : arrangement physique .....	111
3.1.1. Topologie .....	111
3.1.2. Type de fusion.....	112
3.1.3. Dynamicité .....	114
3.2. Axe 2 : manipulation physique .....	114
3.2.1. Effecteurs corporels.....	115
3.2.2. Actions physiques .....	116
3.2.3. Utilisation temporelle.....	116
4. CLASSIFICATION DES DISPOSITIFS COMPOSÉS.....	118
4.1. Classification physique et temporelle .....	118
4.2. Comparaison des dispositifs composés avec DECO .....	118
5. APPLICATION : CONCEPTION D'UN DISPOSITIF COMPOSÉ .....	122
5.1. Motivation.....	122
5.2. Première itération.....	123
5.3. Deuxième itération : changement du type de fusion.....	124
5.4. Troisième itération : usage temporel parallèle.....	128
5.5. Quatrième itération : dynamicité .....	129

5.6. Conclusion .....	130
6. DISCUSSION ET PERSPECTIVES .....	131
6.1. DECO pour décrire, comparer et générer des solutions .....	131
6.2. Recommandations pour la conception de dispositifs composés .....	131
6.3. Limites et travaux futurs .....	133
7. CONCLUSION DU CHAPITRE 5 .....	134

---

### Publications associées

---

G. Perelman, M. Serrano, M. Raynal, C. Picard, M. Derras and E. Dubois. DECO: A Design Space for Device Composition. DIS'16, 435-446. Article Long

# 1. INTRODUCTION

## 1.1. MOTIVATION

Au cours des dernières années, de nombreux projets de recherche ont conçu et mis en œuvre de nouveaux dispositifs d'interaction résultant de la combinaison ou de l'agrégation de dispositifs existants [50, 80, 94, 106, 144, 179, 204]. Parmi ceux-ci, nous pouvons citer l'agrégation d'un écran tactile sur une souris comme dans LensMouse [204], l'ajout d'un ballon gonflable dans une souris comme dans Inflatable Mouse [94], deux claviers fixés ensemble comme dans FlipKeyboard [50], ou deux tablettes collées pour former un dispositif à double face comme dans Codex [80]. Ces projets ont suivi une approche de conception empirique aboutissant au développement de solutions ad-hoc combinant des dispositifs existants, sans mettre en œuvre un processus systématique ou structuré. Bien que ces combinaisons se soient révélées utiles dans plusieurs contextes, les concepteurs qui veulent produire ce type d'appareils sont livrés à eux-mêmes pour explorer les multiples alternatives de conception.

Dans ce chapitre, nous étudions et affinons le concept de composition de dispositifs afin de mieux exploiter les possibilités offertes par la combinaison de dispositifs existants dans le but d'en générer de nouveaux, ainsi que de tirer parti de cette combinaison de manière plus systématique. La composition de dispositif consiste en l'assemblage physique de plusieurs dispositifs existants pour en créer un nouveau, ci-après appelé dispositif composé. Le concept de composition de dispositifs peut être relié aux concepts d'interaction multimodale et de dispositif composite [10, 71]. L'interaction multimodale est basée sur la combinaison de plusieurs modalités d'interaction, chacune étant définie comme un couple de composants : <dispositif, langage> [35]. La notion de dispositif composite, introduite par Bardram [10], définit la combinaison de services interactifs. Mais dans nos travaux, le concept de composition de dispositifs se concentre plus spécifiquement sur la combinaison physique de deux dispositifs et non de deux modalités ou services interactifs.

Dans ce chapitre, nous développons donc le concept de la composition de dispositifs à travers la définition, l'illustration et l'évaluation [12] d'un espace de conception pour la combinaison physique des dispositifs d'interaction. Notre espace de conception, DECO (DEvice Composition), est structuré selon deux axes : 1) l'arrangement physique, c'est-à-dire la composition physique des différents dispositifs et 2) la manipulation physique,

c'est-à-dire la façon dont un utilisateur manipule chaque élément du dispositif composé. À l'aide de DECO, nous classifions et comparons les dispositifs existants pour illustrer comment notre espace de conception aide à décrire et à comparer différentes solutions.

En plus de l'illustration de DECO sur les dispositifs existants, nous utilisons cet espace de conception pour élaborer un nouveau dispositif composé supportant l'interaction multidimensionnelle. Ce dispositif composé vise à ajouter des degrés de liberté à une souris traditionnelle. Nous décrivons en détail le processus de conception itératif centré utilisateur, en utilisant notre espace de conception, le long de quatre itérations complètes impliquant des évaluations d'utilisateurs. Ces itérations prouvent que DECO contribue à la conception de dispositifs composés utilisables et performants.

## 1.2. STRUCTURE DU CHAPITRE

Dans la section 2, nous définissons les termes liés à la composition de dispositifs et utilisés au cours de ce chapitre. La section 3 présente notre espace de conception pour la composition de dispositifs, DECO, et décrit chacun de ses axes et propriétés. La section 4 illustre cet espace par des dispositifs composés existants et compare leurs propriétés physiques. La section 5 décrit l'utilisation de notre espace de conception pour créer un nouveau dispositif composé : la Roly-Poly Mouse 2 (RPM2). Enfin, dans la section 6, nous discutons les avantages et limitations de notre approche de conception et du prototype généré et suggérons des pistes d'amélioration.

## 2. APPROCHE DE COMPOSITION : DÉFINITIONS

Les travaux existants traitant du paradigme de la composition, décrits dans notre état de l'art, ont porté peu d'attention à l'identification et à l'analyse des propriétés physiques de la composition de dispositifs. Une telle considération est cruciale car elle affecte les capacités d'interaction du dispositif composé. Par exemple, en fonction de leur position relative, deux appareils peuvent être utilisés simultanément (par exemple un ballon et une souris, comme dans Inflatable Mouse [94]) ou séquentiellement (un clavier et un touchpad collés à l'arrière du dispositif, comme dans FlipKeyboard [50]). La modification de la position relative des composants peut également altérer fortement le dispositif résultant : la LensMouse [204] offrirait des fonctionnalités complètement différentes si l'écran tactile était placé sur le côté de la souris au lieu de sa partie haute.

L'objectif de notre concept de composition de dispositif est donc de prendre en compte cet aspect physique. Nous définissons ce concept comme suit : une **composition de dispositifs** est l'assemblage (ou composition / combinaison) physique de plusieurs **éléments** existants, selon une configuration physique bien identifiée, aboutissant à un **dispositif composé**. Les éléments possibles impliqués dans une composition de dispositifs comprennent des objets physiques (p. ex. une balle) et des dispositifs interactifs (p. ex. une souris).

Cette définition soulève deux problématiques de conception de la composition de dispositifs :

- *Arrangement physique* : l'intégration des éléments physiques doit tenir compte de leur position relative et de l'altération de leur forme initiale.
- *Manipulation physique* : différentes actions et parties du corps de l'utilisateur peuvent être impliquées pour interagir avec le dispositif composé.

Ces deux considérations sont au cœur de notre espace de conception, DECO.

## 3. DECO : UN ESPACE DE CONCEPTION POUR LA COMPOSITION DE DISPOSITIFS

L'objectif de DECO est d'aider à la conception de nouveaux dispositifs d'interaction résultant de la composition de dispositifs telle que définie précédemment. Pour atteindre cet objectif, DECO est construit autour de deux axes : l'arrangement physique des éléments (Axe 1) et la manipulation physique du dispositif composé (Axe 2). Nous retenons ces deux axes car ils décrivent les deux aspects les plus importants de tout dispositif : sa forme (Axe 1) et son utilisation (Axe 2).

### 3.1. AXE 1 : ARRANGEMENT PHYSIQUE

L'axe d'arrangement physique (Axe 1, Figure 45) décrit la configuration physique et/ou spatiale des différents éléments pour créer un dispositif composé. Ces configurations physiques ou spatiales ont un impact direct sur la forme finale du dispositif et sur ses manipulations physiques, c'est-à-dire sa facilité d'utilisation et les degrés de liberté (DDL) disponibles. Cet axe décrit la position relative de chaque élément (Topologie), la manière dont les éléments sont liés physiquement (Type de fusion) ainsi que la variation de cette configuration au cours du temps (Dynamicité).

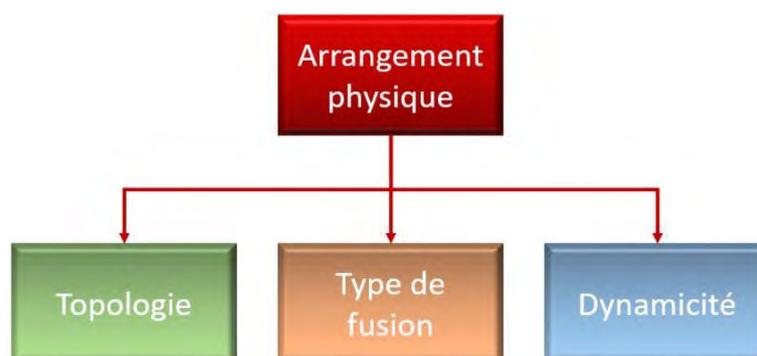


Figure 45. Axe de l'arrangement physique et ses propriétés.

#### 3.1.1. TOPOLOGIE

Cette propriété de DECO décrit l'organisation spatiale et physique de chaque élément les uns par rapport aux autres. Nous avons identifié trois valeurs possibles pour cet axe : Imbriqué, Attaché et Séparé (Figure 46).

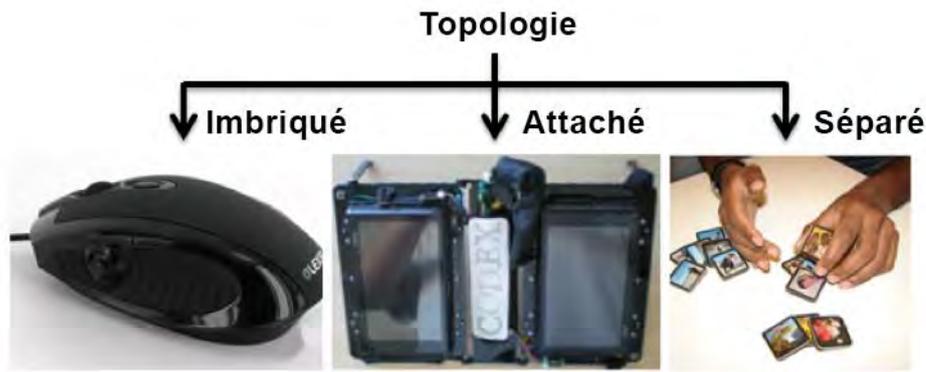


Figure 46. Illustration des trois catégories de topologie : Imbriqué comme dans Lexip 3D [106], Attaché comme dans Codex [80] et Séparé comme dans Siftables [116].

*Imbriqué* : La topologie de composition est imbriquée lorsqu'un élément est confiné à l'intérieur d'un autre. Par exemple, GlobeMouse [53] résulte de l'insertion d'une SpaceMouse dans une trackball 3DDL. L'Inflatable Mouse résulte de l'insertion d'un ballon dans une souris [94].

*Attaché* : La topologie de composition est attachée lorsque les éléments sont agrégés physiquement. C'est la forme la plus courante d'arrangement physique parmi les dispositifs composés. Par exemple, la LensMouse [204] est un dispositif composé résultant de l'assemblage d'une souris et d'un écran tactile (les éléments sont attachés ensemble, Figure 47 - centre). Par exemple, le Codex [80] est un dispositif créé en fixant deux écrans tactiles ensemble (Figure 46 - centre).

*Séparé* : la topologie de composition est séparée lorsque plusieurs éléments sont spatialement composés, c'est-à-dire que leur relation spatiale est détectée, sans être physiquement composés. La paire d'appareils (souris, clavier) par exemple ne peut pas être considérée comme un appareil composé car ils ne sont pas spatialement composés (leur disposition spatiale n'a aucun impact sur l'interaction). Un exemple de dispositif composé séparé est Siftables [116] : ces petits écrans ne sont pas collés ensemble mais peuvent détecter leur position relative et leur orientation (Figure 46 - droite). Une instance particulière d'une topologie séparée existe lorsque le dispositif est fait de deux éléments séparés non liés spatialement, mais qui peuvent être collés si nécessaire (propriété de dynamicité, cf. Section 3.1.3).

### 3.1.2. TYPE DE FUSION

Si dans la topologie Imbriquée l'un des éléments est à l'intérieur de l'autre et dans la topologie Séparée il n'y a pas de contact physique identifié, dans le cas d'une topologie de type Attaché, une question subséquente est de savoir si la forme physique des éléments

est modifiée pour supporter la composition. Les concepteurs de tels dispositifs doivent anticiper si les périphériques existants peuvent être utilisés directement ou si de nouveaux dispositifs doivent spécifiquement être conçus. La propriété Type de fusion décrit quels éléments (le cas échéant) sont modifiés au cours d'une composition. Nous avons identifié trois valeurs de Fusion : (A-B), (A←B) et (A↔B).

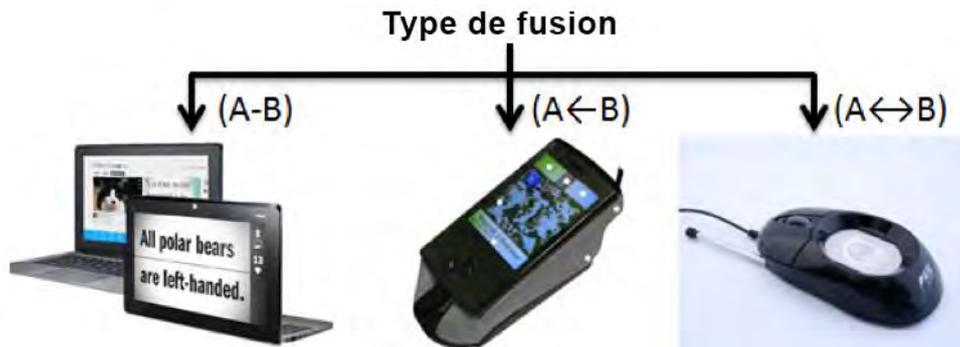


Figure 47. Illustration des trois types de fusion : (A-B) comme dans Billboard [96], (A←B) comme dans LensMouse [204] et (A↔B) comme dans Ink222 [85].

(A-B) : Cette forme de fusion représente la situation où les éléments sont inchangés pendant la composition : tous les éléments sont attachés ensemble sans altérer leur forme originale. Par exemple, un écran d'ordinateur peut simplement être fixé à un autre sans autre modification physique, comme dans l'ordinateur portable à double écran Billboard [96] (Figure 47 - à gauche). Une approche similaire est utilisée dans le FlipKeyboard [50], qui est composé d'un touchpad attaché au dos d'un clavier.

(A ← B) : Cette forme de fusion représente la situation où la forme physique de l'élément A est modifiée (forcée, enlevée, remodelée, etc.) pour accueillir ou y attacher l'élément B. Par exemple dans le cas de LensMouse [204] (Figure 47 - centre), la souris sert de réceptacle pour un écran tactile : pour placer l'écran tactile à un angle approprié, le couvercle de la souris a été modifié et les boutons de la souris ont également été retirés. Dans le Trackball Keyboard [179], un clavier a été sculpté pour intégrer une trackball sur son dessus.

(A↔B) : Cette forme de fusion représente la situation où la forme des deux éléments a été modifiée pour permettre leur composition. La souris Ink222 [85] (Figure 47 - droite) est un dispositif composé d'une souris classique et une tablette tactile. Dans ce cas, la souris a été sculptée et la tablette tactile a été remodelée pour s'adapter à un espace circulaire. Dans la CapMouse [185], les boutons de la souris ont été supprimés et un pavé tactile a été adapté sur la partie supérieure de la souris.

### 3.1.3. DYNAMICITE

Cette propriété de DECO caractérise le changement de l'arrangement physique du dispositif au cours du temps. L'arrangement physique est soit Statique, c'est-à-dire défini une fois pour toutes, soit Dynamique, c'est-à-dire qu'il peut changer en fonction de la tâche, de l'utilisateur ou du contexte (Figure 48). Habituellement, cette propriété s'applique aux valeurs de topologie Attaché et Séparé, lorsqu'un périphérique peut passer de l'une à l'autre. Dans le cas d'une topologie Imbriquée, il serait nécessaire d'ouvrir l'un des éléments pour en sortir le second. À notre connaissance, cette combinaison de propriété n'a jamais été proposée. Par exemple, Rooke et al. [149] propose de multiples arrangements spatiaux entre des écrans hexagonaux sans bords. En utilisant des écrans séparés, un utilisateur peut effectuer des sélections et des rotations tandis qu'en utilisant des écrans attachés, un utilisateur peut effectuer des « swipes » ou des inclinaisons (Figure 48).

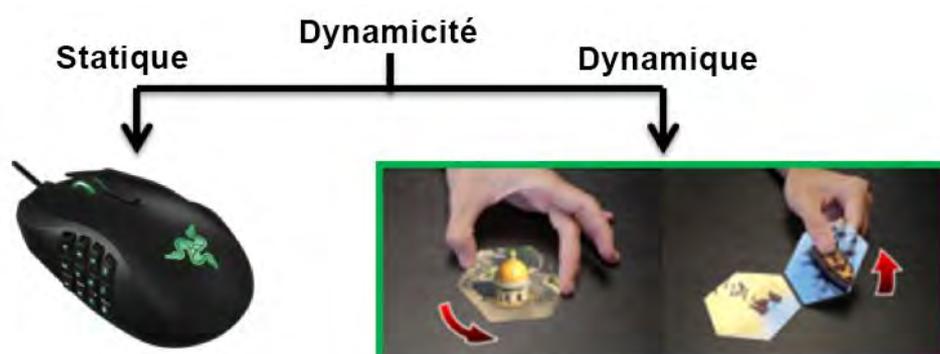


Figure 48. Illustration de la propriété de dynamicité : statique comme dans la Razer Naga [144] et dynamique comme dans les Hexagonal bezel-less screens [149].

## 3.2. AXE 2 : MANIPULATION PHYSIQUE

L'axe de manipulation physique de DECO (Figure 49) décrit les manipulations des éléments du dispositif composé et vise à caractériser son utilisation. D'un point de vue physique, se concentrer sur l'utilisation du dispositif signifie se concentrer sur sa prise en main (effecteurs corporels) et sur les manipulations détectables par le dispositif (actions physique et utilisation temporelle). L'axe de manipulation physique est donc décrit par ces 3 propriétés.

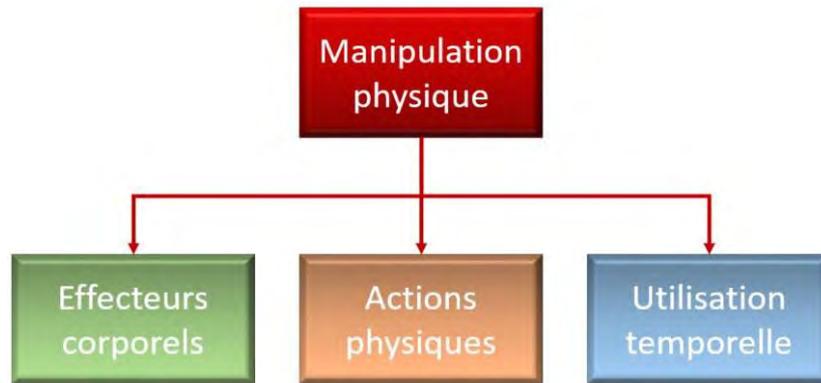


Figure 49. Axe de la manipulation physique et ses propriétés.

### 3.2.1. EFFECTEURS CORPORELS

Cette propriété de DECO identifie les parties du corps agissant sur chaque élément du dispositif composé. Les dépendances anatomiques entre effecteurs corporels peuvent largement influencer leur utilisation dans un dispositif composé [104]. Un effecteur corporel peut être par exemple un doigt, une main, un pied ou la combinaison de plusieurs d'entre eux. Cette propriété peut donc prendre comme valeur la ou les parties du corps impliquées dans l'interaction avec chaque élément. Par exemple, le Trackball Keyboard [179] est un dispositif conçu pour un usage bimanuel, avec une main sur chaque élément : il peut être décrit par le couple <keyboard : main non dom.; trackball : main dom.> (Figure 50 - gauche). L'Inflatable Mouse [94] est conçue pour être utilisée avec la main dominante : elle peut être décrite par le couple <mouse : main dom.; ballon : main dom.> (Figure 50 - droite).

Cet axe permet d'identifier si plusieurs éléments du dispositif composé impliquent la même partie du corps, comme avec l'Inflatable Mouse. Dans ce cas, les actions physiques appliquées à chaque élément doivent être compatibles, ce qui peut impliquer d'adapter ou de modifier l'un des éléments.



Figure 50. Illustration de la propriété effecteurs corporels. Trackball keyboard [179]: <clavier : main non dom.; trackball : main dom.> (gauche). La Inflatable Mouse [94]: <souris : main dom.; ballon : main dom.> (droite).

### 3.2.2. ACTIONS PHYSIQUES

Cette propriété de DECO décrit les actions physiques qui peuvent être détectées sur chaque élément. Des exemples d'actions incluent des translations, des rotations, des « swipes », etc. Les notations existantes telles que UAN [72] peuvent également être employées pour décrire des actions plus complexes. Le rôle de cette propriété est donc d'aider à anticiper la capacité de l'utilisateur à produire les actions physiques attendues sur chaque élément, à anticiper les conflits et à envisager différentes alternatives de conception.

Cette propriété est affinée par les degrés de libertés (DDL) impliqués par chaque action physique. La CubicMouse [54], par exemple, est un dispositif composé combinant un cube et trois tiges. Le cube possède un capteur de suivi 6DDL qui permet de consigner la position et l'orientation de l'appareil. Il peut être décrit comme suit : <Cube : translations (3DDL), rotations (3DDL); StickX : translation (1DDL); StickY : translation (1DDL); StickZ : translation (1DDL)>.

### 3.2.3. UTILISATION TEMPORELLE

Cette propriété de DECO décrit la composante temporelle des actions physiques réalisées avec le dispositif composé. L'utilisation temporelle d'un système interactif a déjà été décrite dans la littérature. Les propriétés d'Allen [3] décrivent la composition temporelle des modalités. Cette approche a, par la suite, été simplifiée avec deux opérateurs dans le modèle CASE [120] : Parallèle et Séquentiel.

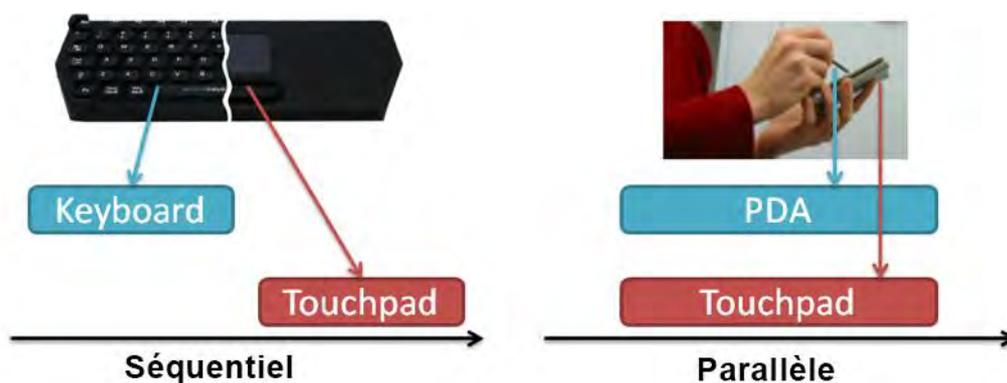


Figure 51. Illustration de la propriété d'usage temporel : séquentiel comme dans FlipKeyboard (gauche) et parallèle comme dans HybridTouch (droite).

Dans notre espace de conception, nous utilisons ces opérateurs pour définir la possibilité d'utilisation temporelle simultanée des éléments dans un dispositif composé. Évidemment, cette propriété est influencée par l'arrangement physique et les actions

physiques, ce qui peut empêcher [85] ou encourager [185] l'utilisation simultanée d'éléments. Par exemple, le FlipKeyboard [50] et HybridTouch [176] ont le même arrangement physique (attaché sur l'arrière). Cependant, dans le FlipKeyboard, seul un de ses éléments peut être utilisé à la fois : l'utilisation temporelle est séquentielle. HybridTouch, basé sur la composition d'un PDA et d'un touchpad, facilite l'utilisation simultanée de ses deux éléments avec une interaction bimanuelle : l'usage temporel est parallèle (c.-à-d. qu'il est possible d'utiliser les deux éléments en parallèle, Figure 51).

Pour valider notre espace de conception, nous utilisons l'approche proposée par Beaudoin-Lafon [12] afin d'évaluer les modèles de conception. Cette approche est basée sur trois propriétés caractérisant la capacité d'un modèle à décrire des solutions existantes (pouvoir descriptif), à comparer des solutions existantes (pouvoir d'évaluation) et à générer des solutions novatrices (puissance générative). La pertinence d'une telle validation a déjà été établie dans des travaux antérieurs visant à évaluer les approches de conception d'interaction [36, 92].

## 4. CLASSIFICATION DES DISPOSITIFS COMPOSÉS

Dans cette section, nous illustrons l'utilisation de DECO pour décrire un ensemble de dispositifs composés existants. Nous donnons d'abord un aperçu de la façon dont ces appareils s'intègrent à notre espace. Ensuite, nous décrivons et comparons en détail quatre appareils existants pour illustrer la manière dont notre espace de conception dévoile leurs différences.

### 4.1. CLASSIFICATION PHYSIQUE ET TEMPORELLE

Parmi les propriétés identifiées pour les deux axes de DECO, nous avons adopté une représentation graphique basée sur la topologie physique, le type de fusion, la dynamique et l'utilisation temporelle. Les deux propriétés restantes (effecteurs corporels et actions physiques) correspondent à des n-uplets moins appropriés pour une représentation graphique. Dans le tableau ci-dessous, les dispositifs composés de la littérature sont positionnés par rapport à ces propriétés sélectionnées de DECO. Une étoile est utilisée pour mettre en évidence les dispositifs composés dynamiques (Table 2).

		Arrangement physique					
		Topologie	Imbriqué	Attaché			Séparé
		Type de fusion		A-B	A<-B	A<->B	
Utilisation temporelle	Séquentielle	GlobeMouse	Hexagonal bezel-less ★ FlipKeyboard Codex	Trackball Keyboard LensMouse	Ink222	Hexagonal bezel-less ★	
	Parallèle	Lexip 3D Inflatable Mouse	Razer naga Dualscreen HybridTouch	PadMouse CubicMouse RPM2 ★	Mouse 2.0	Siftable HoverPad RPM2 ★	

★ = Dispositif composé dynamique

Table 2. Dispositifs composés de la littérature classés selon les axes de DECO.

Cette représentation graphique met en évidence la diversité des dispositifs composés existants évoqués précédemment dans ce chapitre. Il révèle également que l'espace de conception est pertinent pour classer le travail existant car chaque combinaison possible de propriétés (cellule) est illustrée avec un dispositif composé existant.

### 4.2. COMPARAISON DES DISPOSITIFS COMPOSÉS AVEC DECO

Pour illustrer plus précisément DECO, nous avons sélectionné quatre dispositifs composés de la littérature : la GlobeMouse [53], la CubicMouse [54], la PadMouse [8] et

la CapMouse [185] (Figure 52). Bien que ces dispositifs semblent similaires sur certains aspects, DECO permet de les comparer et d'exposer leurs similitudes et leurs différences (Table 2).



Figure 52. De gauche à droite, la Cubic Mouse [54], la GlobeMouse [53], la PadMouse [8] et la CapMouse [185].

### GlobeMouse

Ce dispositif est basé sur la composition d'une SpaceMouse et d'un trackball. La SpaceMouse est imbriquée dans la coque du trackball. L'arrangement physique de ce dispositif est statique. Une seule main de l'utilisateur est nécessaire pour manipuler le périphérique composé : <SpaceMouse : main dom.; trackball : main dom.>. Les actions physiques attendues peuvent être formalisées comme suit : <SpaceMouse : translation (3DDL); Trackball : rotation (3DDL)>. Ces actions ne peuvent être effectuées que de manière séquentielle.

### CubicMouse

Ce dispositif est basé sur quatre éléments principaux : un cube physique localisé en 6D (position et orientation dans un espace 3D) et trois barres traversant le centre du cube. Le cube et les baguettes sont attachés ensemble. Le cube a été modifié pour permettre le positionnement des tiges (Type de fusion : A ← B). L'arrangement physique de la CubicMouse est statique. Les deux mains de l'utilisateur sont nécessaires pour manipuler le dispositif composé : un pour tenir le cube et l'autre pour régler les tiges <cube : main non dom.; tiges : main dom.>. Comme indiqué précédemment, les actions physiques peuvent être décrites comme suit : <Cube : translations (3DDL), rotations (3DDL); StickX : translations (1DDL); StickY : translations (1DDL); StickZ : translations (1DDL)>. Les actions physiques sur les tiges et sur le cube peuvent être effectuées en parallèle.

Nous utilisons DECO pour comparer CubicMouse et GlobeMouse. Bien qu'il s'agisse de deux souris augmentées, l'arrangement physique de ces dispositifs diffère, car leur topologie respective est Imbriqué et Attaché. La manipulation physique diffère aussi : la

CubicMouse est un dispositif bimanuel, qui offre un total de 9 DDL, tandis que la GlobeMouse peut être utilisée d'une seule main et offre 6 DDL. De plus, la CubicMouse fournit un usage parallèle et la GlobeMouse un usage séquentiel. Notre espace de conception met en évidence avec précision les différences entre ces deux dispositifs composés qui diffèrent à la fois en termes d'arrangement physique et de manipulation physique.

#### PadMouse

Ce dispositif est basé sur la composition de deux éléments : une souris ordinaire et un pavé tactile. Les deux éléments sont attachés ensemble. Seule la souris a été modifiée pour produire le PadMouse : les boutons de la souris ont été supprimés pour permettre la composition avec le touchpad (Type de fusion :  $A \leftarrow B$ ). Un bouton numérique sur le pavé tactile remplace les boutons de la souris. L'arrangement physique de PadMouse est statique. Pour être manipulé physiquement, le dispositif composé ne nécessite qu'une seule main. Il peut être décrit comme <mouse : main dom.; pavé tactile : main dom.>. Les actions physiques incluent des translations 2D du dispositif composé et un l'interaction tactile 2D sur le pavé tactile, représentés comme suit : <Souris : translation (2DDL); Pavé tactile : entrée tactile (2DDL)>. Enfin, il est possible d'utiliser les éléments en parallèle.

À la lumière de DECO, la CubicMouse et la PadMouse sont similaires en termes d'arrangement physique (attaché, type de fusion  $A \leftarrow B$  et statique), mais ils diffèrent fortement en termes de manipulation physique. Tout d'abord, en ce qui concerne les effecteurs corporels, la CubicMouse nécessite deux mains alors que la PadMouse n'en nécessite qu'une seule. De plus, la composition des dispositifs a des effets différents en termes de DDL disponibles : les boutons du PadMouse ont été retirés et remplacés par le touchpad, ce qui induit un changement de DDL; dans la CubicMouse, les DDL de chaque élément ont été conservés. La comparaison entre ces deux dispositifs met en évidence la capacité de DECO à différencier les dispositifs du point de vue de leur manipulation physique, même si l'arrangement physique est similaire. Cela souligne la pertinence de considérer la manipulation physique comme un axe de notre espace de conception.

#### CapMouse

Ce dispositif est basé sur la composition de deux éléments : une souris régulière et une surface capacitive. Ces deux éléments sont attachés ensemble, mais les deux éléments ont été modifiés pour créer la CapMouse (Fusion Type  $A \leftrightarrow B$ ) : les boutons de la souris ont

été modifiés et la surface capacitive a été courbée pour s'adapter à la forme de la souris. L'arrangement physique de la CapMouse est statique. Sous la surface capacitive, un seul bouton permet des clics de souris. Les effecteurs corporels, les actions physiques et l'utilisation temporelle de ce dispositif composé sont les mêmes que pour la PadMouse.

Bien que les manipulations physiques de la CapMouse et de la PadMouse soit les mêmes, leur arrangement physique met en évidence une différence fondamentale. La CapMouse nécessite des modifications physiques des deux éléments (type de fusion  $A \leftrightarrow B$ ), alors que la PadMouse n'altère qu'un seul de ses éléments (type de fusion  $A \leftarrow B$ ). En conséquence, les concepteurs pourraient facilement prototyper de nouvelles versions de la PadMouse en remplaçant le pavé tactile par un autre dispositif tactile rectangulaire, tel qu'un smartphone ou une smartwatch par exemple. En d'autres termes, dans la fusion  $A \leftarrow B$ , la conception de multiples variantes du dispositif peut être envisagée en ne modifiant que l'élément B, tandis que dans la fusion  $A \leftrightarrow B$ , cette approche serait plus complexe car les deux éléments doivent être modifiés.

#### Conclusion

Cette description de quatre dispositifs composés et leur comparaison à l'aide de DECO met en lumière les capacités de notre espace de conception pour décrire et distinguer avec précision les solutions existantes en termes d'arrangement physique et/ou de manipulation physique (Table 3).

			GlobeMouse	CubicMouse	PadMouse	CapMouse
Arrangement physique	Topologie	Imbriqué	x			
		Attaché		x	x	x
		Séparé				
	Type de fusion	A-B		x		
		A<-B			x	
		A<->B				x
	Dynamicité	Statique	x	x	x	x
Dynamique						
Manipulation physique	Effecteurs humain	Main dom.	x	x	x	x
		Main non dom.		x		
	Actions physiques	Rotation	3 DoFs	3 DoFs		
		Translation	3 DoFs	6 DoFs	2 DoFs	2 DoFs
		Entrée tactile			2 DoFs	2 DoFs
	Utilisation temporelle	Séquentiel	x			
		Parallèle		x	x	x

Table 3. Description des GlobeMouse, CubicMouse, PadMouse et CapMouse à l'aide de DECO.

## 5. APPLICATION : CONCEPTION D'UN DISPOSITIF COMPOSÉ

Pour évaluer l'utilité de DECO pour la génération de nouvelles solutions, nous rapportons la conception et l'évaluation d'un nouveau périphérique composé, la RPM2, basée sur la combinaison d'une souris traditionnelle avec la Roly-Poly Mouse (RPM) (cf. chapitre 4). La RPM est un dispositif inspiré du culbuto, avec une base circulaire qui permet jusqu'à 5 DDL (translation 2D, inclinaison 2D et rotation 1D). Ce dispositif est décrit dans le chapitre précédent.

### 5.1. MOTIVATION

Dans cette section, nous détaillons la motivation qui justifie notre conception d'un nouveau dispositif composé combinant la RPM et une souris, à savoir : doter la RPM d'un système de localisation autonome. Pour ce faire, nous avons choisi de la rendre compatible avec le dispositif d'entrée le plus répandu, la souris, afin de faciliter son adoption.

Le prototype de la RPM initial repose sur un système de localisation infrarouge (basé sur un ensemble de caméras OptiTrack) coûteux pour enregistrer la position et l'orientation de l'appareil. Une solution embarquée, utilisant un petit capteur de 6 DDL, a été proposée mais n'a pas la précision d'une souris laser pour détecter sa position 2D. Une manière de surmonter cette limitation technique est de combiner la RPM avec une souris pour bénéficier du capteur laser 2D d'origine de cette dernière. Cette solution présente l'avantage supplémentaire de réduire considérablement le coût du dispositif résultant.

Pour faciliter l'adoption de la RPM, nous avons décidé d'envisager son utilisation comme une extension d'une souris classique : la RPM pourrait alors être utilisée en conjonction avec une souris mais aussi de manière indépendante. Du point de vue de DECO, cette configuration induit un arrangement physique dynamique : un dispositif dans lequel les éléments peuvent être attachés ou séparés en fonction du contexte d'utilisation. Par exemple, lorsqu'il est attaché, l'utilisateur peut manipuler les deux éléments avec la main dominante, tout en utilisant le clavier avec son autre main. Lorsque le clavier n'est plus nécessaire, les deux dispositifs peuvent être séparés et utilisés en mode bimanuel, permettant ainsi d'étendre l'amplitude des mouvements réalisables avec la RPM.

Par conséquent, nous avons décidé d'utiliser DECO pour guider notre processus de conception d'un nouveau dispositif composé combinant la souris et la RPM. Dans la section suivante, nous illustrons les différentes versions de la nouvelle RPM (simplement appelé RPM 2) grâce à un processus de conception itératif centré utilisateur basé sur DECO. Les participants aux différentes évaluations que nous avons conduit tout au long de ce processus (10 hommes et 2 femmes) avaient en moyenne 24.8 ans (écart-type = 3.6). Tous étaient droitiers.

## 5.2. PREMIERE ITERATION

### Premiers prototypes

Nous avons commencé par concevoir la version attachée du dispositif composé : cette version pouvait potentiellement conduire à modifier physiquement l'un des deux éléments (RPM ou souris) et devait donc être considérée avant d'explorer la version séparée. En premier lieu, nous nous sommes attachés à choisir un type de fusion. Nous voulions prototyper une version simple du dispositif et l'améliorer, par la suite, en utilisant le second axe de DECO, la manipulation physique.



Figure 53. Prototypes en carton de la RPM 2 : la souris est placée sur le côté droit (gauche), à l'arrière (centre) ou sous (droite) la RPM.

Nous avons construit un prototype en carton en utilisant le type de fusion A-B (aucun appareil n'est physiquement modifié) car c'est la forme de fusion la plus simple (Figure 53). Nous avons utilisé un support de carton intermédiaire afin de maintenir les deux éléments liés entre eux. Nous avons réalisé plusieurs prototypes en composant une souris USB avec un support en carton sur lequel était placée la RPM, sans altérer aucun des éléments servant à la composition. Nous avons ensuite fait varier la position de la souris par rapport au carton pour tester différentes compositions physiques. Ces itérations nous a conduit à 5 versions différentes du dispositif composé en fonction de la position de la souris : à gauche, à droite, à l'avant, à l'arrière ou sous la RPM. La RPM a été placée sur

le support en carton mais pas attaché à celui-ci pour assurer la rotation libre du dispositif et ainsi conserver son amplitude de manipulation.

Nous avons ensuite effectué une pré-étude afin d'évaluer l'impact de la topologie spatiale des éléments sur l'utilisabilité et sur les manipulations physiques. Nous avons demandé à quatre utilisateurs de manipuler toutes les versions en carton de la RPM 2. Les utilisateurs devaient manipuler le dispositif composé en effectuant des translations, des rotations, des inclinaisons et des gestes composés (translation + inclinaison). À la fin de l'expérience, les utilisateurs ont évalué la facilité d'utilisation du dispositif au moyen d'une entrevue informelle.

Nous avons constaté que la distance entre la paume de la main et la souris, résultant du support en carton, avait un impact sur l'utilisabilité du dispositif. Cela est dû au fait que lorsque la RPM 2 est translaturée, le carton dérive lentement autour de la RPM et la souris n'est plus alignée avec le mouvement de la main.

En outre, les utilisateurs ont déclaré que les gestes composés étaient complexes à effectuer sur la RPM 2. Cela était également dû à la distance entre les deux éléments. Comme nous voulions conserver les propriétés d'interaction de la RPM d'origine, nous avons considéré ces limitations comme un problème majeur.

### 5.3. DEUXIEME ITERATION : CHANGEMENT DU TYPE DE FUSION

Pour résoudre les problèmes du premier prototype, nous devions permettre à l'utilisateur de tenir à la fois la souris et la RPM en même temps, c'est-à-dire réduire la distance physique entre les deux éléments. En utilisant DECO, nous avons cherché un autre type de fusion qui rapprocherait les deux éléments sans altérer les actions physiques. Dans notre contexte, une modification physique sur la RPM était impossible car elle aurait un impact sur les DDL de rotation 3D de l'appareil. Le type de fusion ( $A \leftrightarrow B$ ) n'était donc pas approprié. Nous avons donc décidé de concevoir une fusion ( $A \leftarrow B$ , avec la souris en A et la RPM en B) en considérant les modifications de la souris visant à y intégrer la RPM. Une contrainte supplémentaire était que les deux éléments devaient être collés sans les fixer de manière permanente pour préserver la propriété dynamique du dispositif.

Second prototype

La deuxième version du dispositif composé était globalement plus petite que la précédente car nous avons enlevé le support en carton. La partie supérieure de la coque

de la souris a été remplacée par une version imprimée en 3D (prototype v2, Figure 54). Cette couverture a été sculptée pour permettre l'intégration de la RPM. Une conséquence était que nous devions déplacer les boutons sur le côté de la souris, leur place habituelle étant utilisée par la RPM. En parallèle, nous avons voulu déterminer la taille optimale de la RPM car sa taille pouvait avoir un impact important sur la manipulation physique du dispositif et sur le confort de l'utilisateur. Pour répondre à ces questions de conception, nous avons effectué une série de tests utilisateur.



Figure 54. Prototype v2 de la RPM 2. Ce prototype est basé sur le type de fusion (A ← B).

Dans le premier test, nous avons fait varier la taille de la RPM pour mesurer son impact sur les actions physiques. Douze utilisateurs devaient effectuer des translations, des rotations, des inclinaisons et des gestes composés avec le dispositif. À la fin de l'expérience, les participants ont été invités à évaluer la taille de la RPM par rapport à la taille de la souris et ils devaient évaluer la facilité d'utilisation du dispositif. Les diamètres des différentes versions de RPM étaient de 4, 5, 6, 8 et 10 cm (Figure 55).



Figure 55. Différentes tailles de RPM, de gauche à droite : 4, 5, 6, 8 et 10cm.

Selon les résultats, la taille optimale pour la RPM était de 5 cm. Avec cette taille, les deux éléments avaient presque la même largeur et permettaient une posture plus confortable. De plus, les utilisateurs ont préféré placer les boutons sur le côté gauche de la souris plutôt que sur la droite car appuyer sur les boutons avec le pouce (à gauche) est moins fatigant qu'avec l'annulaire ou le petit doigt (à droite). Cela confirme l'importance de choisir correctement les effecteurs corporels pour chaque élément.

Pour évaluer l'efficacité du dispositif dans une tâche concrète, nous avons effectué un second test où nous avons demandé aux utilisateurs d'effectuer une tâche de pointage 2D en suivant le protocole d'expérimentation de la loi de Fitts [172]. Quatre utilisateurs devaient effectuer des sélections parmi 25 cibles disposées circulairement en utilisant notre dispositif et une souris traditionnelle. Nous avons évalué six indices de difficulté (définis comme un ratio de la taille cible sur la taille du cercle support) : 3.17, 3.70, 4.08, 5.93 et 6.33. Les résultats ont montré que le pointage avec la souris était plus rapide qu'avec la RPM 2. Les erreurs de pointage, c'est-à-dire le fait de manquer la cible, étaient significativement plus élevées avec notre prototype. Les commentaires des participants ont révélé que le support 3D imprimé était difficile à prendre en main : «Le support n'est pas stable, il est compliqué de le maintenir correctement pendant que je clique. » ou « La forme du support n'est pas bien adaptée. C'était inconfortable à saisir. ».

### Troisième prototype

Sur la base de ces résultats, nous avons construit une troisième version du dispositif composé. Nous avons retiré le support imprimé en 3D et directement sculpté la partie supérieure de la coque d'une souris pour accueillir la RPM de 5cm de diamètre. Une fois la taille optimale de la RPM définie, nous avons exploré l'emplacement des boutons via un nouveau test utilisateur. La tâche consistait à effectuer des translations, des rotations, des inclinaisons et des gestes composés avec la RPM 2. Les utilisateurs devaient manipuler deux versions du dispositif composé : une sur laquelle la souris était ouverte au centre et les boutons placés sur le côté gauche (prototype v3a, Figure 56 - Gauche); et une sur laquelle la souris était ouverte sur la droite, laissant le bouton gauche de la souris d'origine disponible (prototype v3b, Figure 56 - Droite). Ces deux solutions nécessitent l'utilisation de différents effecteurs corporels pour appuyer sur le bouton (pouce ou index) mais aussi pour la manipulation (rotation ou inclinaison) de la RPM.

Le premier résultat obtenu indique que les utilisateurs préfèrent la version avec la RPM sur le centre (prototype v3a, Figure 56 - Gauche). Ceci est dû au fait que, lorsque la RPM est placée sur la droite, la posture de la main est particulièrement inconfortable lors des rotations et des inclinaisons : l'index repose sur le clic gauche alors que tous les autres doigts reposent sur la RPM. Non seulement la manipulation de la RPM est plus complexe mais la posture est fatigante. En outre, les utilisateurs ont trouvé que les gestes composés étaient faciles à effectuer sur la version creusée au centre. Dans cette version, la manipulation physique de la RPM est effectuée en utilisant les trois doigts du milieu.



Figure 56. Prototype v3a (Gauche) et v3b (Droite) de la RPM 2. Ces prototypes diffèrent dans la position de la RPM par rapport à la souris ainsi que dans la disposition des boutons.

Sur la base de ces résultats, nous avons réévalué le dispositif composé avec quatre utilisateurs dans une tâche de pointage 2D suivant le même protocole qu'auparavant. Les résultats ont montré une amélioration significative de la performance du dispositif composé (10% de mieux que la version précédente). Cependant, le prototype v3 avait encore une performance inférieure et un taux d'erreur plus élevé que la souris traditionnelle. Les retours des participants ont montré que cette perte de performance était due à une forte résistance des boutons, difficiles et fatigants à contrôler avec le pouce pendant une longue période de temps. Par conséquent, nous avons amélioré le prototype en intégrant des boutons plus doux (Figure 57).



Figure 57. Prototype v3 de la RPM 2 avec des boutons plus doux. Pour détecter les manipulations de la RPM, celle-ci a été équipée de billes infrarouges (Droite).

## 5.4. TROISIEME ITERATION : USAGE TEMPOREL PARALLELE

Après nous être concentrés sur l'arrangement physique du dispositif composé, nous nous sommes attachés à l'utilisation temporelle de ses manipulations physiques. Notre objectif était de prendre en compte les gestes composés, c'est-à-dire l'utilisation temporelle parallèle des éléments de l'appareil. Bien que ces gestes soient potentiellement réalisables avec notre prototype, nous n'avons pas encore pris en compte ceux-ci dans la conception du dispositif.

Pour évaluer l'utilisation parallèle de la RPM 2, nous avons conçu une expérience basée sur une tâche graphique multidimensionnelle : rotation, échelle et translation (RST, Figure 58) [181]. Quatre utilisateurs ont dû manipuler des rectangles virtuels dans un environnement 2D et les ancrer sur un rectangle cible. Pour déplacer le rectangle, les utilisateurs appuyaient sur le clic gauche, puis réalisaient une translation du dispositif. Pour faire pivoter et mettre à l'échelle le rectangle, les utilisateurs pouvaient respectivement tourner (gauche-droite) ou incliner (avant-arrière) la RPM. Les trois tâches RST pourraient être exécutées en parallèle avec le prototype de la RPM2 v3a (Figure 57 - Droite). À la fin de l'expérience, les utilisateurs ont été invités à évaluer la facilité d'utilisation du dispositif. Nous avons comparé la RPM 2 à la souris classique, répliquant l'interaction présente dans le logiciel PowerPoint [138] (des ancrs sélectionnables autour du rectangle).

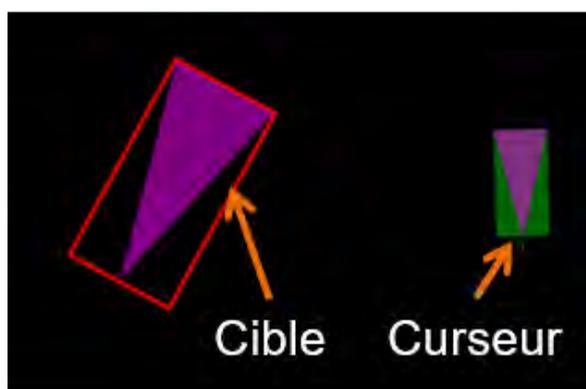


Figure 58. Exemple de tâche de RST.

Les résultats révèlent que notre dispositif n'est que 9% plus lent que la souris ordinaire, ce qui signifie que celui-ci est prometteur mais que l'association entre les gestes et les tâches doit être améliorée. Nous avons observé que la plupart des utilisateurs effectuaient

une rotation, une mise à l'échelle et une translation en parallèle. Les utilisateurs ont trouvé que la RPM 2 était « un bon dispositif parce que les manipulations sont intuitives ».

Ces résultats montrent clairement que nous avons réussi à concevoir un nouveau dispositif composé en utilisant notre espace de conception, DECO. Le prototype v3a (Figure 57) suit une topologie Attaché et les résultats de l'expérience ont mis en lumière une utilisabilité et une performance proche de celle d'une souris traditionnelle. Notre objectif principal étant de proposer un dispositif dynamique, nous avons effectué une dernière itération pour valider son utilisation dans une topologie séparée.

## 5.5. QUATRIEME ITERATION : DYNAMICITE

Dans notre dispositif composé, la RPM et la souris ne sont pas physiquement fixés l'une à l'autre. Par conséquent, pour obtenir une topologie séparée, chaque élément peut être manipulé avec une main différente : ils sont physiquement découplés (Figure 59). Cependant, le couvercle de la souris reste ouvert, ce qui diffère d'une souris traditionnelle : une solution peut être d'utiliser une plaque coulissante pour fermer le trou lorsque la RPM est retirée. Dans cette itération, la solution proposée ci-dessus n'a pas été mise en pratique. Nous avons utilisé une souris ordinaire sans trou car nous nous sommes concentrés sur la performance de l'interaction bimanuelle et non sur la conception du prototype.



*Figure 59. Version séparée de la RPM 2.*

La technique d'interaction bimanuelle est basée sur l'utilisation d'une souris traditionnelle avec la main dominante et de la RPM avec l'autre main (effecteurs corporels : <souris : main dom., RPM : main non dom.>). Ce périphérique composé offre toujours 5 DDL et prend en charge les actions physiques suivantes : <souris : translation (2DDL), RPM :

rotation (3DDL)>. Les deux éléments peuvent être utilisés en parallèle (utilisation temporelle).

Nous avons évalué les performances de cette nouvelle version de la RPM 2 (RPM 2 - Separated) pour une tâche de rotation, d'échelle et de translation (RST). Nous avons comparé la technique bimanuelle avec une souris régulière dans les mêmes conditions que dans l'itération précédente. Douze sujets ont effectué l'expérience. Les résultats ont révélé une différence significative entre les deux techniques d'interaction : le dispositif composé est plus rapide que la souris (22% plus rapide). Ces résultats montrent que la version Séparée de la RPM 2 fonctionne mieux que son homologue attaché et illustre l'intérêt de développer un dispositif composé dynamique : selon la tâche à effectuer, l'utilisateur pourra facilement adopter la version attachée ou séparée du dispositif composé.

## 5.6. CONCLUSION

Nous avons proposé un nouveau dispositif composé dynamique, la RPM 2. Elle peut être utilisée de manière unimanuelle ou bimanuelle. Une série de tests montre que la RPM 2 est un dispositif efficace dans les deux arrangements physiques. Pour concevoir ces deux versions de la RPM 2, nous avons utilisé DECO dans un processus de conception itératif centré utilisateur : nous avons guidé notre processus de conception avec les deux axes de DECO et avons utilisé ses propriétés pour proposer itérativement plusieurs prototypes (Tableau 3). En résumé, cet exemple illustratif montre que DECO est un outil utile pour la conception de nouveaux dispositifs composés.

		Arrangement physique				
		Topologie	Imbriqué	Attaché		Séparé
		Type de fusion		A-B	A<-B	A<->B
Utilisation temporelle	Séquentiel		Iteration 1	Iteration 2		
	Parallèle			Iteration 3 ★	Iteration 4 ★	

★ = Dispositif composé dynamique

Table 4. Itérations de conception dans DECO.

## 6. DISCUSSION ET PERSPECTIVES

### 6.1. DECO POUR DECRIRE, COMPARER ET GENERER DES SOLUTIONS

Comme illustré sur l'étude de cas (RPM 2), en utilisant DECO, un concepteur peut explorer les implications d'un design donné en changeant certains paramètres parmi les axes et propriétés de DECO. Les axes de DECO constituent également des leviers pour suggérer de nouvelles possibilités de conception, basées sur les propriétés observées dans différentes situations de composition de dispositif.

Pour valider notre espace de conception, nous utilisons l'approche proposée par Beaudoin-Lafon [12] afin d'évaluer les modèles de conception. Concrètement, pour valider le pouvoir descriptif de DECO, c'est-à-dire sa capacité à décrire un large éventail de solutions différentes, nous avons décrit et classé les dispositifs composés existants en utilisant l'axe d'arrangement physique et la propriété d'utilisation temporelle de notre espace de conception (Figure 45 et Figure 49). Pour valider le pouvoir d'évaluation de DECO, c'est-à-dire sa capacité à différencier plusieurs solutions, nous avons comparé quatre de ces dispositifs, à priori similaires, et montré comment DECO permet d'identifier leurs similitudes et leurs différences en termes d'arrangement physique et de manipulation. Enfin, pour valider le pouvoir génératif de DECO, c'est-à-dire sa capacité à créer de nouvelles solutions, nous avons généré un nouveau dispositif composé basé sur la combinaison de la souris traditionnelle et de la souris Roly-Poly Mouse (cf. chapitre 4) : nous avons utilisé DECO pour explorer différentes alternatives dans un processus de conception itératif centré utilisateur.

Grâce à cette exploration en trois parties, nous avons démontré que DECO est un espace de conception pertinent qui peut être utilisé pour décrire, comparer et générer de nouveaux dispositifs composés.

### 6.2. RECOMMANDATIONS POUR LA CONCEPTION DE DISPOSITIFS COMPOSES

La génération et la conception de nouveaux dispositifs reposent souvent sur une approche empirique qui allie créativité et connaissances pratiques, comme décrit par Vallgård et al. [183]. Cependant, le manque d'inspiration peut affecter la phase d'idéation du

processus de conception. Dans le contexte particulier de la composition de dispositifs, DECO est une approche analytique structurée qui cherche à offrir aux concepteurs des leviers pour stimuler la génération d'idées et produire des alternatives de conception. La combinaison d'une telle approche formelle avec des ressources de conception plus informelles comme un brainstorming est possible et a déjà été proposée en IHM [19].

Pour aider les concepteurs à tirer profit de DECO dans un processus de conception, nous proposons un ensemble de recommandations tirées de la conception de la RPM 2 avec DECO.

#### Arrangement physique

L'une des premières étapes, lors de la conception d'un dispositif interactif, consiste à analyser les besoins, les tâches et le contexte d'usage des utilisateurs. Ces aspects peuvent notamment impacter les possibilités en termes d'arrangement physique des éléments. À ce stade de conception, l'utilisation de DECO devrait conduire à choisir une topologie spécifique. Les deux choix possibles sont une topologie imbriquée / attachée, qui conduit à un seul objet physique, ou une topologie séparée, qui conduit à plusieurs objets.

Dans le cas d'une topologie attachée, l'avantage de commencer par la fusion (A-B) est qu'elle n'induit aucune modification sur les dispositifs initiaux. Cela permet d'évaluer les premiers prototypes et d'obtenir des commentaires avant d'envisager des modifications physiques sur l'un des éléments impliqués dans la composition du dispositif.

#### Manipulation physique

La conception des attributs de manipulation physique doit être reconsidérée après chaque itération de conception sur l'arrangement physique. Même de petits changements physiques dans le prototype peuvent induire de grandes différences en termes de manipulation : par exemple, dans notre expérience avec la RPM 2, la modification de la taille de la RPM a considérablement affecté sa facilité d'utilisation.

Enfin, lorsque l'objectif est de produire un usage parallèle du dispositif, une bonne approche consiste, a priori, à commencer à concevoir et évaluer un usage séquentiel dans les premières itérations : l'idée est de valider que chaque élément peut être manipulé correctement et d'identifier leur potentiel de manipulation (amplitude maximale, stabilité, précision, etc.), avant d'envisager la manipulation combinée, plus complexe à découvrir et à manipuler pour un utilisateur.

### 6.3. LIMITES ET TRAVAUX FUTURS

Nous avons identifié quelques limitations de notre espace de conception en termes de généralisation, granularité et définition de propriétés que nous détaillons ci-dessous.

Dans notre travail, nous avons principalement considéré les dispositifs composés de seulement deux éléments. Bien que le travail relatif aux dispositifs combinant plus de deux éléments soit limité, nous pensons qu'une généralisation de l'utilisation de notre espace de conception est pertinente. L'évaluation de sa capacité à décrire, évaluer ou aider à générer des dispositifs composés basés sur plus de 2 éléments devra donc aussi être étudiée.

La définition de DECO a été largement déterminée par une analyse critique des dispositifs composés décrits dans la littérature. Ces solutions se concentrent clairement sur la combinaison de dispositifs existants tels qu'une souris et un pavé tactile, c'est-à-dire la combinaison de deux ensembles de capteurs et d'objets physiques. Bien que nous ayons adopté la même approche pour définir le concept de la composition de dispositifs, DECO pourrait également constituer un bon support pour analyser des combinaisons à un grain de détail plus fin, c'est-à-dire au niveau d'un capteur. Ainsi, une suite à ces travaux pourra porter sur l'extension de notre espace de conception afin d'envisager ce type de composition. L'objectif de cette extension sera de permettre de mieux caractériser les dispositifs de type souris augmentée où le second élément n'est pas à proprement parlé un dispositif (capteur de pression, moteur générant des vibrations, etc.).

À l'avenir, il serait aussi nécessaire d'analyser et d'affiner les caractéristiques de la topologie séparée. Ce cas est particulièrement intéressant pour les systèmes multi-surfaces, c'est-à-dire composant différentes surfaces interactives. Nous pouvons par exemple explorer les relations géométriques et spatiales entre les surfaces. Décrire les manipulations physiques dans ce cas nécessitera également une extension de DECO pour décrire des interactions telles que des gestes en l'air ou autour de l'appareil [15], qui ont également des propriétés spatiales. Cette piste sera partiellement explorée dans le chapitre 6.

## 7. CONCLUSION DU CHAPITRE 5

Dans ce chapitre, nous avons présenté DECO, un espace de conception pour la composition de dispositifs basé sur leurs dimensions physiques. DECO est structuré selon deux axes : l'arrangement physique des dispositifs et leur manipulation physique par l'utilisateur. L'axe d'arrangement physique décrit la topologie de la composition (imbriquée, attachée ou séparée), le type de fusion (quel dispositif est physiquement modifié pour produire la composition) et la dynamique de l'arrangement physique. L'axe de manipulation physique décrit les effecteurs corporels, c'est-à-dire les parties du corps, utilisées pour manipuler le dispositif, les actions physiques et l'utilisation temporelle des éléments. Nous avons validé DECO en : 1) l'utilisant pour classer les dispositifs composés existants 2) en comparant ces dispositifs entre eux et 3) en illustrant son pouvoir génératif : en mettant en œuvre DECO, nous avons conçu un nouveau dispositif composé en combinant une souris avec la Roly-Poly Mouse dans un processus de conception centré utilisateur. Enfin, nous avons proposé un ensemble de lignes directrices pour aider les concepteurs à appliquer notre espace de conception à la création de nouveaux dispositifs composés.



# CHAPITRE 6

## COMBINAISON DE DISPOSITIFS : APPLICATION À L'INTERACTION AVEC DES TABLEURS EN MOBILITÉ

---

### Structure du chapitre

---

1. INTRODUCTION.....	138
1.1. Motivation.....	138
1.2. Terminologie.....	139
1.3. Structure du chapitre.....	139
1.4. Analyse statistique des résultats.....	140
2. ÉTUDE PRÉLIMINAIRE : EXPLORATION DES GESTES.....	141
2.1. Dimension du stacking.....	141
2.2. Étude préliminaire sur le confort des gestes.....	142
2.3. Résultats.....	144
3. PROTOTYPE ET IMPLÉMENTATION.....	145
3.1. Coque conductrice.....	145
3.2. Algorithme de détection.....	147
3.3. Faisabilité industrielle.....	147
4. ÉTUDE 1 : SÉLECTION D'UNE PLAGE DE CELLULES.....	149
4.1. Techniques d'interaction pour la sélection d'une plage de cellules.....	149
4.1.1. Interaction tactile.....	149
4.1.2. Intersection.....	150
4.1.3. Stack-and-Drag.....	151
4.1.4. Finger Stack.....	152
4.2. Description de l'étude.....	154
4.3. Résultats.....	157

5. ÉTUDE 2 : TABLEAU CROISÉ DYNAMIQUE .....	159
5.1. Présentation des tableaux croisés dynamiques .....	159
5.2. Techniques d'interaction pour la création et la manipulation de tableaux croisés dynamiques.....	160
5.2.1. Interaction tactile.....	160
5.2.2. Dual-Screen Touch.....	161
5.2.3. Stack-and-Touch .....	161
5.2.4. Stack-and-Translate.....	162
5.3. Description de l'étude.....	162
5.4. Résultats.....	165
6. SCÉNARIO D'UTILISATION.....	167
7. DISCUSSION ET PERSPECTIVES .....	169
8. CONCLUSION DU CHAPITRE 6 .....	173

---

### Publications associées

---

Gary Perelman, Marcos Serrano, Célia Picard, Mustapha Derras, Emmanuel Dubois. Interaction multi-dispositifs pour tableurs sur tablette. AFIHM. 29ème conférence francophone sur l'Interaction Homme-Machine, Aug 2017, Poitiers, France. ACM, IHM-2017, 12 p., 2017, Article long.

Gary Perelman, Marcos Serrano, Celia Picard, Christophe Bortolaso, Mustapha Derras, and Emmanuel Dubois. 2018. Cell Selection for Spreadsheets on Tablets: Stacking-Based Interaction. In Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems (CHI EA '18). ACM, New York, NY, USA, Paper D505, 4 pages. DOI: <https://doi.org/10.1145/3170427.3186546>

# 1. INTRODUCTION

## 1.1. MOTIVATION

Les tableurs sont des logiciels très utilisés [2, 57], comme en témoignent leurs 55 millions d'utilisateurs aux États-Unis en 2012 [160]. Ces applications sont également largement utilisées sur dispositifs mobiles [48, 63], avec plus de 100 millions de téléchargements d'Excel sur Android. Cependant, la taille des écrans de dispositifs mobiles étant réduite, il est nécessaire de réaliser fréquemment des défilements et des changements de vues, ce qui nuit à la fluidité de l'interaction. De plus, le vocabulaire d'interaction tactile contient peu de gestes comparativement à la complexité qu'offrent un clavier et une souris. Enfin, les gestes disponibles sur tablette sont, pour la plupart, déjà associés à la navigation dans les données (défilement, zoom). Ainsi, il est nécessaire d'utiliser une combinaison de plusieurs gestes pour manipuler les données, ce qui rallonge l'interaction. En résumé, la taille limitée de l'affichage et surtout les limitations de l'interaction tactile ne représentent pas une solution pertinente pour interagir avec les tableurs sur tablettes.

Pour résoudre ces problèmes, plusieurs approches ont été envisagées. Une solution consiste à utiliser des techniques d'interaction multi-touch [60, 76, 77, 182, 190] pour étendre les degrés de liberté limités de l'interaction tactile. Cependant, ces approches ont certaines limitations (par exemple la mémorisation ou le manque de visibilité) et n'augmentent pas la taille d'affichage. En conséquence, elles ne parviennent pas à répondre entièrement aux problèmes soulevés par l'interaction avec un tableur sur dispositif mobile.

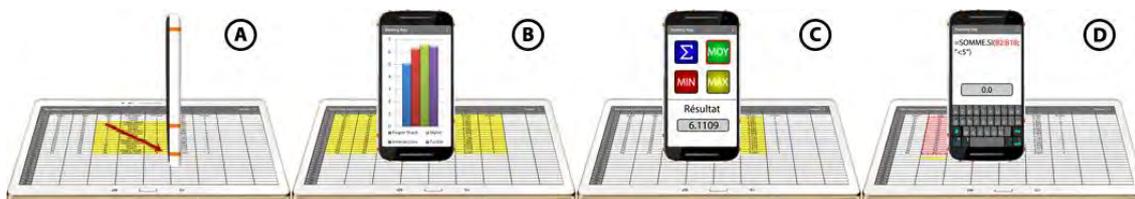


Figure 60. Illustration du stacking d'un smartphone sur l'écran d'une tablette pour l'utilisation d'un tableur : sélection d'une plage de cellules (A), visualisation d'un graphique (B), calcul d'une moyenne (C) et édition d'une cellule (D).

Pour répondre aux besoins soulevés par ces applications, nous proposons d'utiliser un dispositif toujours disponible, le smartphone [158], et le paradigme du stacking (ou empilement), qui consiste à mettre en contact un bord du smartphone avec l'écran de la tablette qui exécute l'application du tableur (Figure 60). L'intérêt est triple :

1) la zone d'affichage initiale est étendue avec un deuxième écran, ce qui permet de séparer les visualisations multiples des données du tableur (telles que des données brutes, des graphiques ou des tableaux croisés dynamiques);

2) le vocabulaire tactile est étendu par des gestes de stacking (déplacement du smartphone sur l'écran de la tablette ou interaction tactile sur l'écran du smartphone) qui peuvent être utilisés pour raccourcir le chemin d'interaction (le toucher peut être utilisé pour naviguer dans les données et le stacking pour leur manipulation);

3) les techniques de stacking n'interfèrent pas avec l'interaction tactile.

Dans ce travail, nous explorons d'abord les différents gestes physiques et tactiles pouvant être effectués avec un smartphone positionné sur une tablette. Ensuite, en fonction des gestes identifiés, nous concevons et évaluons plusieurs techniques d'interaction basées sur le stacking pour deux tâches habituelles dans les tableurs : la sélection d'une plage de cellules (requis pour la plupart des tâches d'un tableur) et la création de tableaux croisés dynamiques. Les tableaux croisés dynamiques sont largement utilisés pour créer des visualisations de données et soutenir l'analyse des données [88].

## 1.2. TERMINOLOGIE

Dans ce chapitre, nous utiliserons le terme de stacking ou d'empilement pour décrire le paradigme d'interaction utilisé et basé sur la mise en contact d'une arête, d'une face ou d'un coin du smartphone avec l'écran de la tablette. Les arêtes, faces et coins du smartphone sont appelés surfaces empilées. Le geste consistant à mettre en contact une partie du smartphone avec l'écran de la tablette sera appelé geste de stack. Le geste consistant à retirer le smartphone du contact avec la tablette sera appelé geste d'unstack.

## 1.3. STRUCTURE DU CHAPITRE

Dans la section 2, nous présentons une étude préliminaire visant à explorer l'espace des gestes possibles apportés par le stacking. La section 3 décrit le matériel nécessaire à la conception de notre prototype ainsi que son processus de fabrication. Nous décrivons aussi les algorithmes logiciels implémentés permettant la détection du smartphone sur l'écran de la tablette ainsi que la détection des gestes définis dans la section précédente. Les sections 4 et 5 décrivent des techniques d'interaction basées sur le stacking ayant pour but de faciliter respectivement la sélection d'une plage de cellules et la création de tableaux croisés dynamiques. Ces techniques ont été évaluées et comparées à l'interaction

tactile traditionnelle. Dans la section 6, nous décrivons l'usage du stacking dans un contexte concret. Enfin, dans la section 7, nous discutons les avantages et limitations de notre approche et de notre prototype et suggérons des pistes d'amélioration.

## 1.4. ANALYSE STATISTIQUE DES RESULTATS

Les sections 2, 4 et 5 ne présentent que les conclusions principales obtenues à partir d'une analyse statistique rigoureuse des données collectées afin de faciliter la lecture. L'analyse statistique complète des résultats de ces études peut être retrouvée en annexe section 2.

## 2. ÉTUDE PRÉLIMINAIRE : EXPLORATION DES GESTES

Comme première étape de notre travail sur l'utilisation du stacking pour faciliter l'interaction avec un tableur sur tablette, nous avons en premier lieu identifié les dimensions du stacking et exploré les différents gestes afin d'écartier les plus inconfortables.

### 2.1. DIMENSION DU STACKING

Dans notre contexte, le stacking consiste à mettre en contact un smartphone avec un écran de tablette qui le détecte. Dans le contexte de notre espace de conception DECO, ce paradigme d'interaction peut être décrit ainsi : les techniques d'interaction basées sur le stacking ont une topologie attachée (A-B) et séparée (arrangement physique dynamique). Les effecteurs corporels impliqués sont les suivants : <Tablette : main non dom. ; Smartphone : main dom.>. Les actions physiques sont les suivantes : <Tablette : Interaction tactile (2 DDL) ; Smartphone : Interaction tactile (2 DDL), Translations (2 DDL), Rotations (3 DDL)>. Enfin, dans le cadre de ces travaux, nous n'explorons que l'utilisation séquentielle des deux éléments.

Nous avons identifié trois dimensions permettant de caractériser le geste du stacking : les surfaces empilées, la prise en main et le type de gestes.

*Surfaces empilées* : nous avons d'abord envisagé le stacking des quatre arêtes et des quatre coins du smartphone sur l'écran de la tablette. Les faces avant et arrière du smartphone peuvent également être utilisées, mais nous avons choisi de les écarter car le smartphone aurait occulté une partie importante de l'écran de la tablette.

*Prise en main* : l'utilisateur tient le smartphone avec la main dominante, tandis que la tablette peut être tenue avec la main non dominante ou être placée sur une surface plane (par exemple une table). Dans le second cas, l'utilisateur peut également effectuer des gestes tactiles sur les écrans de la tablette ou du smartphone en utilisant sa main non dominante. Nous limitons notre exploration au second cas : le smartphone est utilisé comme interacteur tangible et tactile sur l'écran d'une tablette posée sur une surface plane.

*Type de gestes* : les deux gestes de délimitation du stacking sont le stack (c'est-à-dire, poser une surface du smartphone sur l'écran de la tablette) et l'unstack (c'est-à-dire soulever le smartphone afin de supprimer le contact entre le smartphone et la tablette).

Une fois le geste de stack réalisé, l'utilisateur peut effectuer un ensemble de gestes tactiles, manipulations physiques ou gestes composés (c'est-à-dire physiques puis tactiles, p. ex. une translation du smartphone suivi d'un glisser sur son écran). Des gestes tactiles (toucher, glisser ou pincer / étirer) peuvent être réalisés sur les deux dispositifs. Nous identifions quatre manipulations physiques qui peuvent être réalisées avec le smartphone mis en contact avec l'écran de la tablette (Figure 61) : la translation du smartphone sur l'écran de la tablette (A) ; la rotation à plat du smartphone sur l'écran de la tablette (B) ; l'inclinaison du smartphone autour de la surface empilée (C); et la rotation autour du coin du smartphone (D).

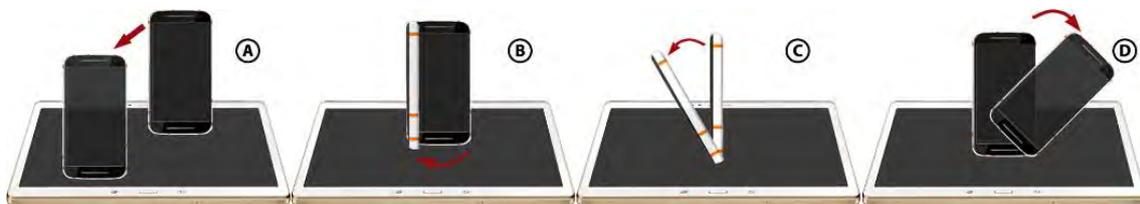


Figure 61. Gestes physiques du "stacking": Translation (A), rotation à plat (B), inclinaison (C) et rotation de coin (D).

## 2.2. ÉTUDE PRELIMINAIRE SUR LE CONFORT DES GESTES

Le nombre de gestes possibles 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 de réduire ce vocabulaire en écartant les gestes inconfortables.

### Tâche et instructions

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 orientations à plat pour les deux configurations ; Figure 62). Ensuite, les participants devaient réaliser (1) une des quatre manipulations physiques du smartphone (translation, rotation à plat, inclinaison, rotation de coin – Figure 61), (2) un geste tactile sur l'écran du smartphone (toucher, glisser, pincer/étirer) 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 de coin revient à réaliser ce geste tactile suite à un

stack initial dans une autre position/orientation (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 notre protocole. 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).

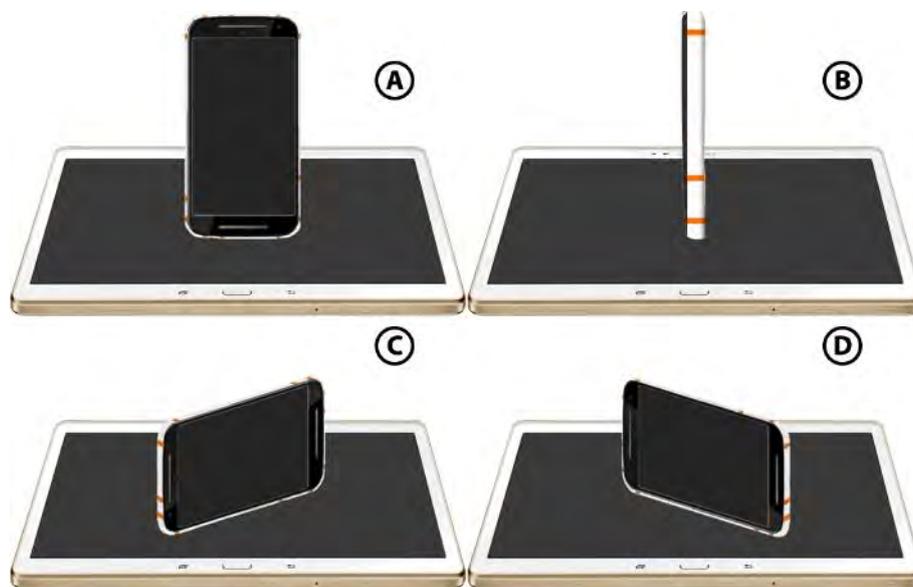


Figure 62. Exemples d'orientation initiale :  $0^\circ$  et  $90^\circ$  en mode portrait (A et B) puis  $-45^\circ$  et  $45^\circ$  en mode paysage (C et D).

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

#### 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). Le smartphone était équipé d'une coque qui permet sa détection par la tablette (c.f. Section 3.1), afin d'évaluer notre dispositif

dans des conditions d'utilisation réelles. Les deux dispositifs étaient éteints tout au long de l'étude.

#### Participants

Six sujets (âge moyen = 26.4, écart-type = 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.

#### Données collectées et analyse statistique

Nous avons filmé l'ensemble de l'expérimentation. 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 participants. Seuls les résultats principaux sont présentés ci-après. Une analyse statistique plus détaillée peut être trouvée dans l'annexe section 2.1.

## 2.3. RESULTATS

Les gestes tactiles du toucher et du glisser ont été considérés comme confortables. Il en va de même pour l'ensemble des gestes physiques. Les résultats obtenus nous ont conduit à ne pas considérer l'utilisation de gestes composés dans la suite de nos travaux et à éliminer le geste du pincer/étirer : ces gestes ont été considérés inconfortables par les utilisateurs et dans certains cas, les gestes étaient impossibles à réaliser (pincer/étirer suite à une inclinaison). 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. En mode 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. 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.

### 3. PROTOTYPE ET IMPLÉMENTATION

Afin de permettre l'utilisation du stacking du smartphone sur l'écran de la tablette, il est nécessaire de détecter la position et l'orientation d'une des surfaces empilées du smartphone sur la surface capacitive de la tablette. Dans ce but nous avons conçu une coque conductrice spécifique pour le smartphone.

#### 3.1. COQUE CONDUCTRICE

Différentes approches ont été proposées pour combiner interaction tangible et surface capacitive. Elles s'appuient soit sur la détection des doigts autour d'un objet non conducteur contraignant la position des doigts [61], soit sur l'utilisation d'effecteurs [187] ou de capteurs [84] spécifiques. Pour éviter l'ajout de contraintes de prise ou main ou de technologies trop complexes et potentiellement coûteuses dans une coque de smartphone, notre approche s'apparente plus à celles de CapStones, ZebraWidgets et CapTUI [29] : nous utilisons un motif de points conducteur assimilables à des touches et reconnaissable par l'écran de la tablette.



Figure 63. Coque conductrice en cuivre (gauche). stacking de notre prototype sur l'écran de la tablette (droite).

Pour obtenir une coque conductrice adaptée au stacking, nous avons conçu plusieurs prototypes basés sur le même schéma : sur chaque arête du smartphone sont disposées trois fines bandes conductrices, des contacteurs. Lorsque le smartphone est placé en contact de l'écran de la tablette, ces contacteurs sont détectés par la surface capacitive de la tablette (Figure 63 – Droite). Afin d'identifier et de différencier chaque arête du smartphone, une répartition spécifique des trois contacteurs (en termes de distance relative entre eux) a été utilisée.

Le premier prototype conçu est basé sur une coque de smartphone classique en plastique que nous avons augmenté à l'aide d'un ruban adhésif épais (en mousse) et d'un ruban

adhésif cuivré. Le ruban adhésif en mousse permet d'aplanir la surface de contact de chaque arête, évitant ainsi toute fausse détection due aux bords arrondis de la coque en plastique. Sur celui-ci sont disposées trois fines bandes de cuivre. Ces bandes sont reliées à l'arrière du smartphone à l'aide d'une feuille de cuivre d'1mm d'épaisseur. Cette feuille a pour but de permettre à chaque contacteur d'être lié à une surface cuivrée suffisante pour déclencher sa détection sur la surface capacitive.

Le second prototype conçu est basé sur une coque de smartphone imprimée en 3D (Figure 63 – Gauche). Sur celle-ci les zones associées aux contacteurs sont bombées et de taille identique afin de faciliter la détection par la surface capacitive de la tablette. Des ouvertures dans la coque permettent de conserver l'utilisation des boutons du smartphone et permettent de le brancher (micro USB, jack, etc.). Quatre vis disposées sur chaque coin de la coque permettent de maintenir le smartphone à l'intérieur de la coque. Du ruban adhésif cuivré a été utilisé sur chaque contacteur de la coque de manière similaire au prototype précédent. Ceux-ci sont reliés à l'aide de ruban adhésif cuivré disposé tout autour de la face arrière de la coque du smartphone afin d'avoir une quantité de cuivre suffisante pour déclencher la détection.

Le troisième prototype conçu est basé sur une coque de smartphone usinée en aluminium (Figure 64). Cette coque possède une forme identique à celle du second prototype : les contacteurs sont légèrement bombés afin de faciliter leur détection. Dans ce prototype, aucun ajout de matériau conducteur n'est nécessaire puisque l'aluminium est directement utilisé comme contacteur. Cette coque a été réalisée au travers d'une collaboration avec le lycée professionnel Renée Bonnet, situé à Toulouse.



*Figure 64. Prototype usiné en aluminium*

## 3.2. ALGORITHME DE DETECTION

Afin de détecter la position et l'orientation du smartphone lorsque celui-ci est placé sur l'écran de la tablette, nous avons développé un algorithme de détection. Tout d'abord, il est important de noter que nous avons choisi d'utiliser trois contacteurs par arête du smartphone afin d'éviter toute interférence avec les gestes classiques de l'interaction tactile (nécessitant un ou deux points de contact). D'autre part, l'utilisation de trois contacteurs permet d'obtenir l'orientation du smartphone sur l'écran de la tablette. Ainsi, si trois points de contact alignés sont détectés par la tablette, celle-ci considère qu'un geste de stack vient d'être effectué. La distance relative entre les points de contact permet de déterminer la face du smartphone mise en contact avec l'écran de la tablette. Enfin, l'orientation de la droite formée par les points de contact permet de déterminer l'orientation du smartphone. Lorsque ces points de contacts sont temporairement perdus, notamment dû à une détection erronée, notre algorithme simule la présence des points manquants jusqu'à leur prochaine détection à l'aide de leur distance relative connue. Lorsque le smartphone est soulevé, les trois points de contact disparaissent et la tablette peut ainsi reconnaître le geste d'unstack.

Une fois le geste de stack détecté, notre algorithme suit les points de contact et recalcule la nouvelle position/orientation du smartphone au cours du temps. Concernant les rotations d'arête/de coin et les inclinaisons, les données de la centrale inertielle du smartphone (accéléromètre, gyromètre, magnétomètre) 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 sont réalisées à l'aide d'un réseau Wi-Fi local ou du Bluetooth.

## 3.3. FAISABILITE INDUSTRIELLE

Ces travaux étant destinés aux utilisateurs des logiciels Berger-Levrault, la faisabilité industrielle de notre prototype représente un paramètre important. Nous avons donc recherché des solutions qui puissent être peu coûteuses et facilement reproductibles à grande échelle.

Nos trois prototypes possèdent des coûts relativement faibles (inférieur à 2 €). Leur construction ne nécessitant aucun matériel rare ou complexe à assembler, la production

de ces coques à grande échelle est donc envisageable. Les second et troisième prototypes de coque (imprimée en 3D et usinée en aluminium) ont été fabriqués en collaboration avec le lycée professionnel Renée Bonnet [148].

La conception de ces coques permet la détection de l'ensemble des gestes confortables identifiés lors de l'étude préliminaire. Ainsi, la prochaine étape de notre travail consiste en l'évaluation de l'utilisabilité et des performances du stacking pour l'interaction avec des tableurs sur tablette.

## 4. ÉTUDE 1 : SÉLECTION D'UNE PLAGE DE CELLULES

La sélection de cellules est une tâche fondamentale dans un tableur car elle est nécessaire pour utiliser la plupart des fonctionnalités du logiciel. En effet, elle définit la plage de cellules sur laquelle une commande ou une fonction s'applique. Nous avons conçu trois techniques d'interaction basées sur le stacking pour la sélection de cellules. Ces techniques sont basées sur trois approches différentes : 1) l'intersection des lignes et des colonnes du tableau (Intersection), 2) le pointer-glisser (Stack-and-Drag), et 3) la combinaison de l'interaction tactile et du stacking (Finger Stack). Aucune de ces techniques n'interfère avec l'interaction tactile traditionnelle. Pour chacune d'entre elles, nous décrivons le mécanisme de sélection, la modification d'une plage ainsi que le mécanisme de défilement (lorsque la plage à sélectionner n'est pas intégralement contenue dans l'écran). Nous comparons ces différentes techniques à l'interaction tactile traditionnelle pour sélectionner une plage de cellules dans un tableur sur tablette.

### 4.1. TECHNIQUES D'INTERACTION POUR LA SÉLECTION D'UNE PLAGE DE CELLULES

#### 4.1.1. INTERACTION TACTILE

Afin de sélectionner une plage de données à l'aide de l'interaction tactile traditionnelle, trois étapes sont nécessaires (Figure 65) : 1) un premier appui (toucher) permet de sélectionner une cellule, 2) un deuxième appui sur l'ancre de la cellule sélectionnée permet d'activer la modification de la plage, et 3) un geste de glisser permet d'étendre la plage sélectionnée. Cet exemple illustre à quel point l'interaction tactile est fastidieuse pour les tâches de feuille de calcul courantes, telles que la sélection de cellules.

Afin de modifier une plage sélectionnée, il est nécessaire de toucher une ancre, puis de réaliser un geste de glisser. Lorsque la plage désirée n'est pas intégralement contenue dans l'écran, il est possible, au cours du geste de glisser, d'approcher le doigt du bord de l'écran : un défilement automatique, dont la vitesse est liée à la proximité du doigt au bord de l'écran, s'enclenche et se désactive lorsque l'utilisateur éloigne son doigt du bord ou relâche son appui.

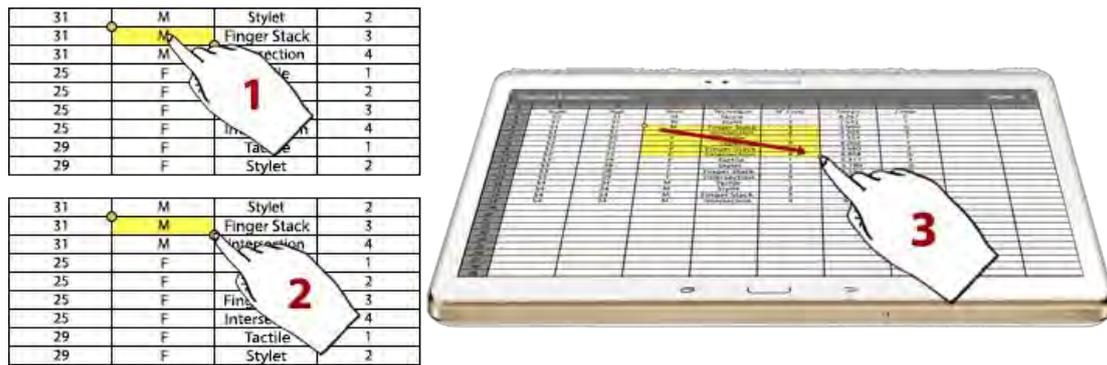


Figure 65. Sélection d'une plage de cellules : Appui sur une cellule (1), sélection de l'ancre (2), glisser pour étendre la sélection (3).

#### 4.1.2. INTERSECTION

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

Afin de réaliser la sélection d'une plage avec cette 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 66 - 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 66 - 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, la sélection est validée.



Figure 66. Technique Intersection : stack (1) et translation (2) pour sélectionner des colonnes et stack (3) et translation (4) pour sélectionner des lignes.

Afin de modifier une sélection déjà réalisée, l'utilisateur peut réaliser 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 (Figure 67). Dans le cas d'un défilement latéral gauche ou droit, le smartphone peut sortir légèrement de la surface tactile de la tablette : seuls deux points de contacts sont nécessaires pour continuer l'action de défilement. La fonction de transfert contrôlant la vitesse de défilement est identique à celle de la technique d'interaction tactile présentée ci-dessus.



Figure 67. Défilement vertical (gauche) et horizontal (droite). La zone représentée en vert désigne la zone dans laquelle le mécanisme de défilement peut être activé.

#### 4.1.3. STACK-AND-DRAG

Cette seconde technique basée sur le stacking s'inspire de l'utilisation classique d'un stylet pour tracer une sélection rectangulaire : le premier contact définit le coin haut-gauche de la sélection, puis le déplacement du stylet permet de définir le coin bas-droit. À l'aide de l'interaction tactile traditionnelle, ce geste de glisser est associé au défilement de l'écran : il n'est donc pas utilisable pour sélectionner une plage de cellules. Cependant, la détection du smartphone étant différenciée de celle d'un geste tactile traditionnel, il est possible de réaliser ce geste de glisser sans déclencher le défilement de l'écran.

Pour utiliser cette technique, un stack initial du smartphone permet de sélectionner 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, 1 cm devant la face avant du smartphone (moitié 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 68 - 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 68 - Droite].



Figure 68. Technique Stack-and-Drag : 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 traditionnelle, des ancres 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 ancres. 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.

#### 4.1.4. FINGER STACK

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



Figure 69. Technique Finger Stack : toucher pour sélectionner le coin haut-gauche (1) et stack pour sélectionner le coin bas-droit (2) de la plage.

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 69 – Gauche). Puis il doit réaliser un stack du smartphone avec sa main droite pour positionner le coin bas-droit (Figure 69 – Droite). 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 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 Stack-and-Drag permet de connaître l'emplacement du « curseur » en évitant l'occultation 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 toucher puis stack permet de sélectionner l'intégralité de la plage demandée (Figure 69).

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.

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.

Avantages des techniques basées sur le stacking pour la sélection de cellules

Chacune des trois techniques précédentes offre des avantages différents pour la sélection des cellules. L'intersection est basée sur deux gestes simples qui ne nécessitent pas de pointage précis sur une seule cellule, généralement petite. Stack-and-Drag est similaire à une sélection basée sur la souris. Finger Stack offre l'avantage d'une sélection directe (simple toucher et stack) sans aucune translation du smartphone ou du doigt (Figure 69). Nous avons décidé de confronter ces techniques dans une étude contrôlée et de les comparer à l'interaction tactile traditionnelle.

## 4.2. DESCRIPTION DE L'ETUDE

Le but de cette étude est de comparer les performances des trois techniques basées sur le stacking entre elles et avec l'interaction tactile traditionnelle pour une tâche de sélection de plage de cellules.

### Tâche et instructions

Les participants ont été invités à sélectionner différentes plages de cellules aussi rapidement que possible avec chacune des quatre techniques d'interaction proposées. Pour limiter les biais expérimentaux, nous avons désactivé les actions de défilement et de zoom : ces deux actions, possiblement utilisées selon la stratégie de sélection de l'utilisateur, auraient pu affecter le temps de réalisation de la tâche de sélection de plage de cellules. Cependant, nous avons gardé l'action de défilement disponible pendant la sélection : pour sélectionner des plages de cellules au-delà de la limite d'écran, l'utilisateur doit effectuer un défilement en utilisant la technique d'interaction tactile traditionnelle (ci-après dénommée Tactile) ou l'une des techniques d'interactions basée sur le stacking (Intersection, Stack-and-Drag ou Finger Stack) et décrites ci-dessus.

Pour représenter la diversité des plages de cellules, nous avons considéré sept catégories distinctes : les lignes complètes, les colonnes complètes,  $1 \times M$  (une seule colonne),  $N \times 1$  (une seule ligne),  $N \times M$  (carré visuel, rectangle horizontal et vertical, voir Figure 70). Les valeurs pour  $N$  sont de 1, 2, 4 ou 9 colonnes et 1, 7, 13 ou 33 lignes pour  $M$ . Ces valeurs ont été choisies en fonction de la différence entre la hauteur et la largeur des cellules : la hauteur de la cellule ( $M$ ) est inférieur à sa hauteur ( $N$ ). La taille des cellules est basée sur celle d'un tableur sur tablette existant [63]. L'origine de la plage de cellules était toujours la cellule D6.

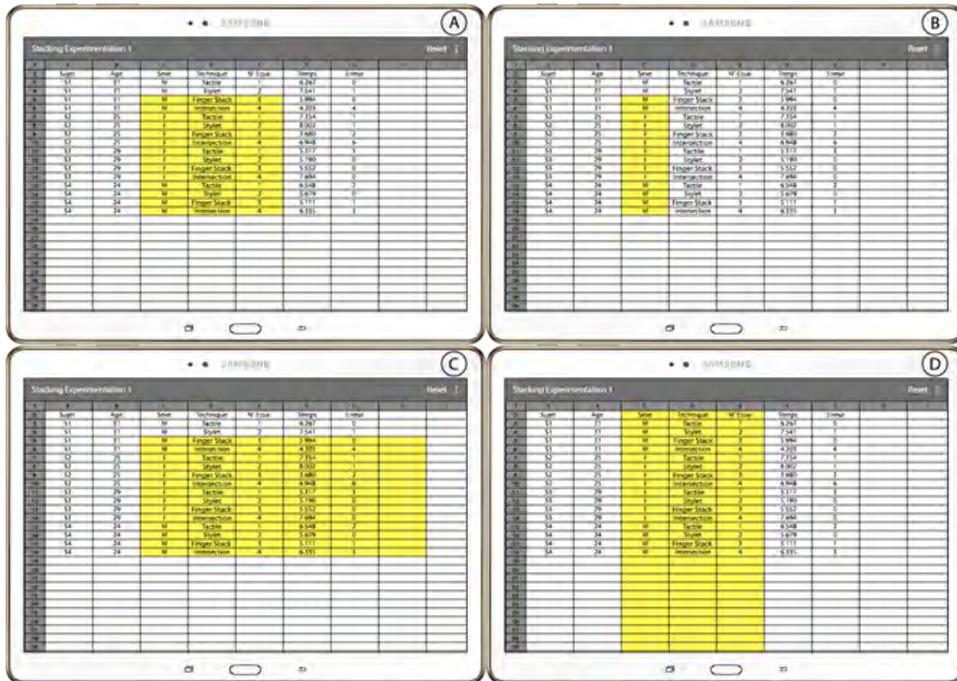


Figure 70. Différentes plages de cellules (de gauche à droite et de haut en bas) : carré, une colonne, rectangle horizontal et colonnes complètes.

Nous avons évalué deux conditions pour chaque plage : une où la plage était totalement à l'intérieur de l'écran (aucun défilement n'était requis) et une où la plage dépassait les limites d'écran (un défilement était requis). Dans ce cas, l'utilisateur doit faire défiler l'écran pour sélectionner la plage dans son intégralité.

Chaque essai correspond à une sélection de plage de cellules et est réalisé comme suit. Une première vue, affichée sur l'écran de la tablette, présente une description de la plage à sélectionner de trois manières : dans un format textuel (ex : NxN), en utilisant les entêtes des cellules (ex : A6 à L35) et en utilisant une image schématique (Figure 71 - Gauche). Une fois prêt, l'utilisateur peut appuyer sur un bouton affiché sur l'écran de la tablette pour lancer l'essai : ceci déclenche l'affichage de la feuille de calcul. Les entêtes associés à la plage de cellules ciblée sont affichés en rouge. Une fois les cellules sélectionnées, si elles étaient incluses dans la plage ciblée, les entêtes correspondants s'affichaient en vert. Sinon, ils étaient affichés en jaune (Figure 71 - Milieu), mettant ainsi en évidence une erreur. En conséquence, l'utilisateur peut toujours savoir quelles cellules doivent encore être sélectionnées (entêtes rouges) et lesquelles doivent être supprimées de la sélection en cours (entêtes jaunes). Une fois la plage cible correctement sélectionnée, une validation automatique de l'essai est déclenchée après 500ms (Figure 71 - Droite) afin d'éviter les faux-positifs.



Figure 71. Première vue de l'application (gauche), exemple de sélection (centre) et validation automatique (droite). La couleur des en-têtes de cellule sont colorés selon l'état de la sélection (vert pour sélectionnés et inclus dans la plage cible, jaune pour sélectionnés mais non-inclus dans la plage cible et rouge pour non sélectionnés mais inclus dans la plage cible).

## Matériel

Nous avons utilisé une tablette Samsung Galaxy S (Android 5.1, octa-core 1.9Ghz et 1.3Ghz) avec un écran 10.5" (2560x1600px), et un smartphone Samsung Galaxy S4 (Android 5.1, quad-core 1.9Ghz) avec un écran 5" (1920x1080px). Le smartphone était équipé de notre coque en cuivre (1er prototype) et la tablette était équipée d'une protection d'écran en plastique. Le smartphone était éteint et une application de tableur développée pour l'étude était lancée sur la tablette. Cette application nous permettait de contrôler les nombreux paramètres et options disponibles pour l'utilisateur. Nous avons reproduit l'interface, la disposition des éléments, l'interaction tactile et la taille des cellules (284x53px) de Google Spreadsheet [63]. L'utilisateur était assis, tenant le smartphone avec sa main dominante, et la tablette posée sur la table en face de lui.

## Participants

Nous avons recruté 12 participants (5 femmes), âgés en moyenne de 27.7 ans (écart-type = 4.7). Six d'entre eux ont participé à l'étude préliminaire. Ils étaient tous droitiers et familiers des dispositifs mobiles (7.5 sur une échelle de 0 à 9) et des tableurs en environnement fixe (7.0). Un seul sujet était familier des tableurs sur tablette (7.0, contre 2.1 en moyenne pour les autres).

## Protocole expérimental

Notre étude a suivi un design intra-sujets 4x7x2 avec Technique d'Interaction (TI) (Tactile, Intersection, Stack-and-Drag et Finger Stack), Type de plage (les sept types décrits précédemment) et Défilement (avec ou sans) comme facteurs.

L'étude a été divisée en quatre blocs, chaque bloc correspondant à une technique d'interaction. L'ordre des blocs a été contrebalancé entre les participants au moyen d'un

carré latin 4x4. Pour chaque bloc, les participants devaient effectuer 14 essais d'entraînement (la moitié d'entre eux avec défilement, l'autre moitié sans), puis 42 sélections de plage de cellules (14 conditions x 3 répétitions). L'ordre des essais dans un bloc était aléatoire. Les participants pouvaient faire une pause entre chaque essai et chaque bloc. Nous avons recueilli un total de 4 techniques d'interaction x 42 essais (7 types de plage x 2 conditions de défilement x 3 répétitions) x 12 participants = 2016 sélections.

#### Données collectées et analyse statistique

Nous avons mesuré le temps nécessaire pour effectuer chaque essai, de la pression du bouton sur la tablette à la validation automatique. À la fin de chaque bloc, nous avons demandé aux participants de remplir un questionnaire SUS et nous avons recueilli leurs retours informels. Les participants ont également été invités à exprimer une appréciation générale (j'aime / je n'aime pas) de la technique d'interaction sur une échelle de Likert à 7 points. L'expérience a duré une heure en moyenne. Seuls les résultats principaux sont présentés ci-après. Une analyse statistique plus détaillée des résultats peut être trouvée dans la section 2.2.

### 4.3. RESULTATS

Les résultats obtenus permettent de conclure clairement que les techniques Finger Stack et Intersection nécessitent, en moyenne, moins de temps pour sélectionner correctement une plage de données que l'interaction Tactile (Figure 72 - Gauche). L'interaction Tactile est respectivement 19.9% et 16.1% plus lente que les techniques Finger Stack et Intersection (Figure 72 - Droite). Ces résultats restent valables quel que soit le type de plage à sélectionner (Figure 73) et sont également maintenus lorsque le défilement est requis. En termes d'utilisabilité, les techniques Intersection et Tactile obtiennent des scores SUS similaires et il apparaît clairement que leurs scores sont supérieurs à ceux obtenus par les autres techniques (Figure 74). Bien que des améliorations puissent être envisagées pour les autres techniques, nous ne considérerons que les techniques Intersection et Tactile (comme base de référence) pour l'étude suivante.

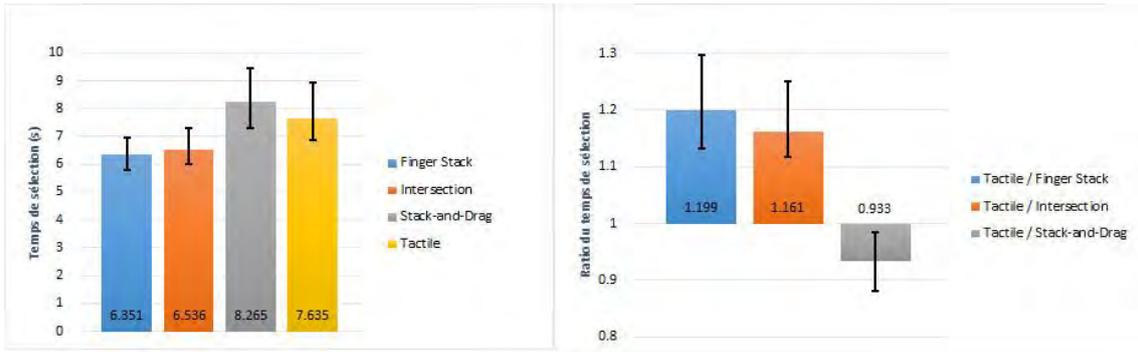


Figure 72. Temps de sélection en s (Gauche) et ratio (Droite) par rapport à Tactile (95% IC).

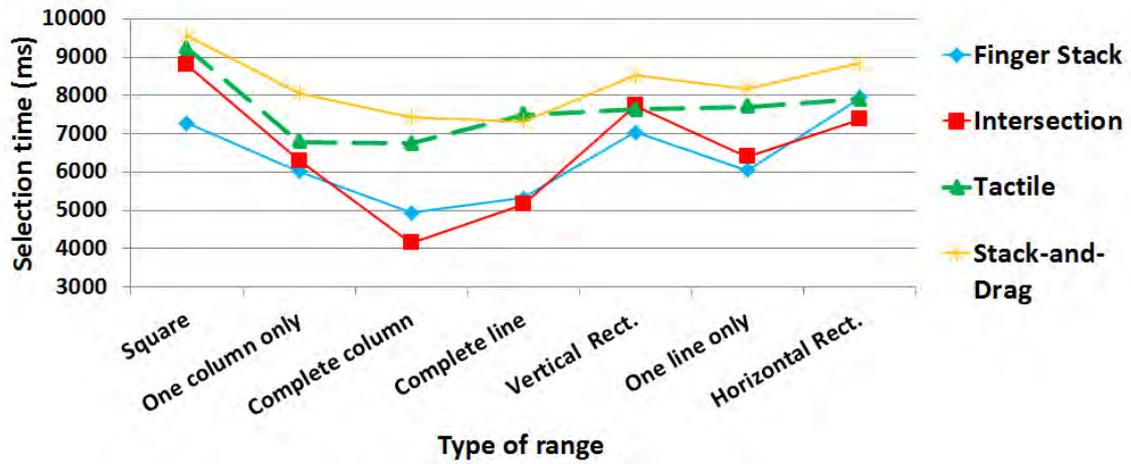


Figure 73. Temps de sélection (en ms) par type de plage et par technique.

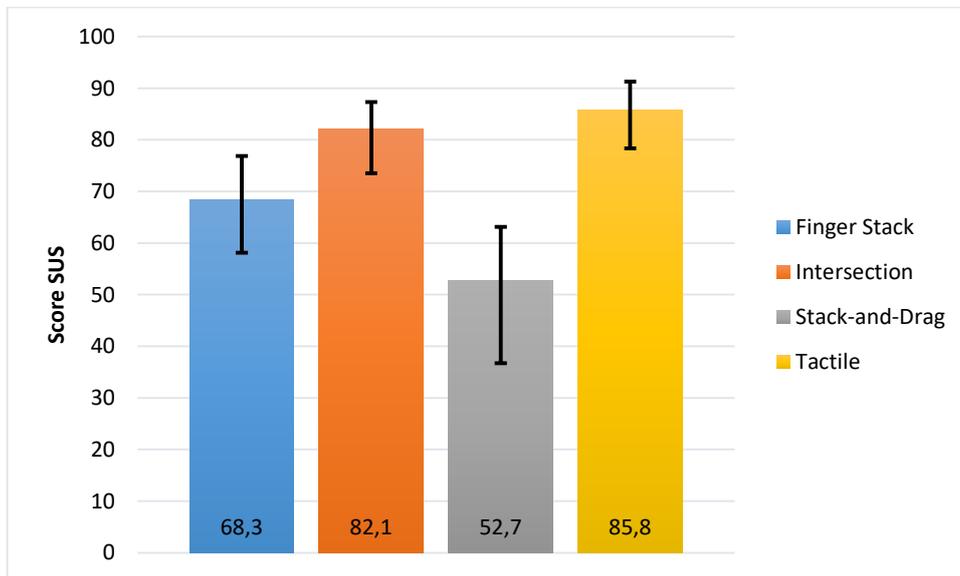


Figure 74. Score SUS par technique d'interaction (95% IC).

## 5. ÉTUDE 2 : TABLEAU CROISÉ DYNAMIQUE

Les tableaux croisés dynamiques sont largement utilisés pour faciliter l'analyse de grands ensembles de données et pour effectuer des visualisations de données selon un regroupement pertinent dans un tableur. Cependant, ils s'appuient sur une manipulation complexe de plusieurs paramètres et filtres. Nous avons conçu trois techniques d'interaction basées sur le stacking pour prendre en charge la création et la configuration de tableaux croisés dynamiques sur des tablettes.

### 5.1. PRESENTATION DES TABLEAUX CROISES DYNAMIQUES

Un tableau croisé dynamique est un outil qui résume rapidement les données d'un tableur à travers différentes fonctions (telles que le nombre, le total ou la moyenne). Après avoir sélectionné la plage de données brutes concernées, l'utilisateur configure le tableau croisé dynamique en sélectionnant et en faisant glisser des en-têtes de données sur un assistant. L'assistant est composé de quatre zones (Filtre, Colonne, Ligne et Valeur). Chaque zone peut être associée à un ou plusieurs en-têtes des données brutes. Ces tableaux croisés dynamiques peuvent être rapidement modifiés en changeant les en-têtes affectés à chaque zone de l'assistant. De plus, la fonction d'agrégation de la zone Valeur (par exemple, moyenne, nombre, etc.) peut être modifiée pour calculer différentes valeurs.

La création d'un tableau croisé dynamique implique trois étapes : 1) la sélection d'une plage de cellules, 2) la sélection d'un en-tête de colonne dans cette plage et 3) l'affectation de cet en-tête à l'une des zones de l'assistant du tableau croisé dynamique. Les étapes 2 et 3 sont répétées pour chaque colonne insérée dans l'assistant.

Nous avons développé trois techniques d'interaction basées sur le stacking pour créer des tableaux croisés dynamiques. Dans ces techniques, le smartphone affiche l'assistant du tableau croisé dynamique tandis que la tablette affiche les données brutes (cellules). Pour toutes les techniques, la sélection initiale de la plage de cellules (étape 1) est effectuée en utilisant la technique Intersection avec le smartphone dans la main dominante (meilleure technique dans l'étude précédente). Cependant, chacune utilise un mécanisme différent pour sélectionner un en-tête de colonne (étape 2) et remplir l'assistant du tableau croisé dynamique (étape 3). Nous avons comparé nos trois techniques d'interaction à l'interaction tactile traditionnelle présentée ci-après.

## 5.2. TECHNIQUES D'INTERACTION POUR LA CREATION ET LA MANIPULATION DE TABLEAUX CROISES DYNAMIQUES

### 5.2.1. INTERACTION TACTILE

Nous n'avons trouvé aucune application Android sur la tablette prenant en charge les tableaux croisés dynamiques. Cependant, Excel est disponible sur les tablettes tactiles Surface (tablettes Windows). Nous avons donc utilisé cette solution comme référence pour développer une application de tableur sur Android intégrant les tableaux croisés dynamiques et la décrivons dans cette section.

Dans Excel sur Surface, l'assistant de tableau croisé dynamique peut être affiché sur un onglet secondaire ou sur l'onglet contenant les données. Après avoir sélectionné les données brutes souhaitées (étape 1) et avoir appuyé sur le menu / sous-menu de droite pour insérer le tableau croisé dynamique, l'application active l'assistant. L'assistant est affiché sur le bord droit de l'écran. Une représentation textuelle du tableau croisé dynamique est affichée dans la feuille de calcul de l'onglet courant. Les en-têtes de colonne obtenus à partir de la sélection précédente peuvent ensuite être glissés (étape 2) et affectés (étape 3) à l'une des quatre zones de l'assistant correspondant à Filtre, Colonne, Ligne et Valeur (Figure 75). Un toucher sur les éléments dans la zone Valeur permet de modifier la fonction d'agrégation à l'aide d'une fenêtre contextuelle. Pour filtrer les éléments, un appui sur l'en-tête dans la représentation textuelle du tableau croisé dynamique ouvre une fenêtre contextuelle spécifique.

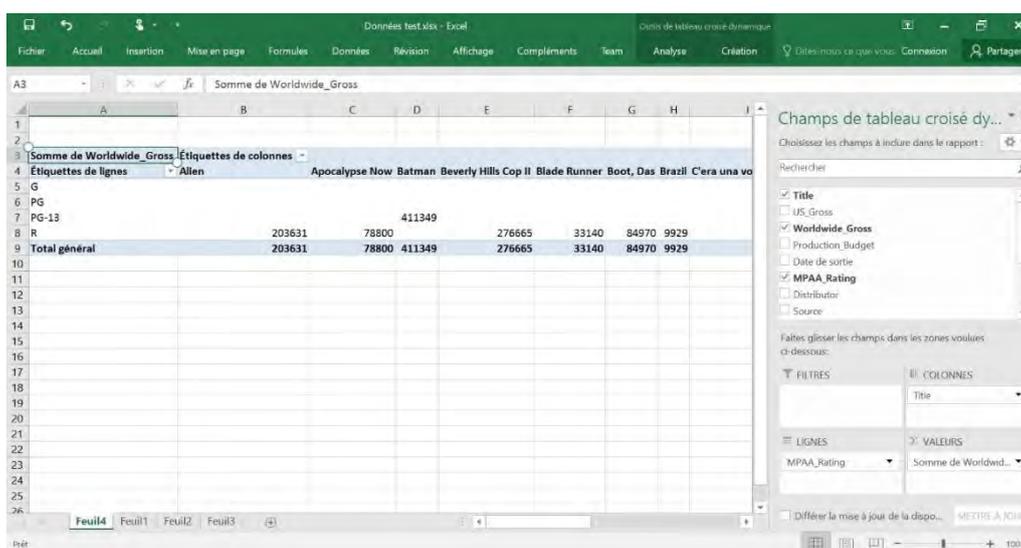


Figure 75. Tableau croisé dynamique dans Excel sur Microsoft Surface.

Au-delà de cette solution tactile, un cycle de conception itératif nous a amené à concevoir 3 techniques d'interaction basées sur le stacking pour la création de tableaux croisés dynamiques.

### 5.2.2. DUAL-SCREEN TOUCH

La première technique que nous avons conçue est basée uniquement sur l'interaction tactile. Notre objectif était d'évaluer l'apport de l'utilisation de l'écran supplémentaire apporté par le smartphone, indépendamment de l'utilisation du stacking. Tout d'abord, un appui sur la tablette (sur un en-tête de colonne) sélectionne la colonne de données (étape 2). Ensuite, l'utilisateur peut insérer ces données dans l'assistant en touchant la zone souhaitée sur le smartphone (étape 3) (Figure 76 - Top). Les données peuvent ensuite être glissées et déposées entre les différentes zones de l'assistant. Un appui sur les éléments permet de filtrer et de changer la fonction d'agrégation. Un appui long sur un élément le supprime de la zone.



Figure 76. Techniques Dual-Screen touch et Stack-and-Touch.

### 5.2.3. STACK-AND-TOUCH

Cette seconde technique est basée sur le stacking du smartphone sur la colonne désirée pour la sélectionner (étape 2, comme dans la technique Intersection). Dans ce cas, nous souhaitons évaluer l'apport d'une technique mêlant l'interaction tactile et le stacking. Après avoir sélectionné la colonne, l'interaction est la même qu'avec la technique Dual-Screen Touch, c'est-à-dire, appuyer sur l'une des zones de l'assistant y affecter l'en-tête (étape 3, Figure 76 - Bas).

#### 5.2.4. STACK-AND-TRANSLATE

Cette troisième technique repose également sur le stacking du smartphone sur la colonne souhaitée (étape 2). Dans ce cas, nous souhaitons évaluer l'apport d'une technique intégralement basée sur le stacking. Pour cela, l'assistant s'affiche temporairement sur l'écran de la tablette (en plus de l'écran du smartphone) sous la position du stack (Figure 77). Pour insérer une colonne de données dans l'assistant, l'utilisateur déplace le smartphone dans la direction de la zone de l'assistant souhaitée (étape 3). Après la translation, l'assistant sur l'écran de la tablette disparaît pour éviter l'occultation des données.

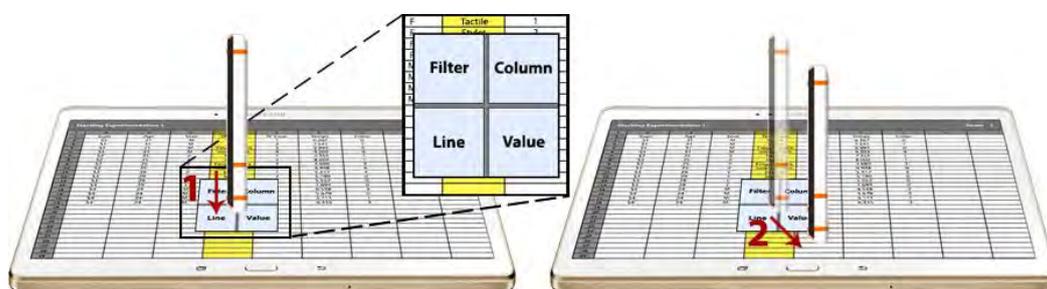


Figure 77. Technique Stack-and-Translate.

Avantages des techniques basées sur le stacking pour les tableaux croisés dynamiques

L'interaction basée sur le stacking pour les tableaux croisés dynamiques présente deux avantages. Tout d'abord, l'écran du smartphone peut être utilisé pour afficher l'assistant et les tableaux de données du tableau croisé dynamique, limitant l'occultation des données de la feuille de calcul. Deuxièmement, le smartphone peut être utilisé comme un interacteur tactile et tangible pour sélectionner les données et configurer l'assistant du tableau croisé dynamique. Cela permet une manipulation directe des données.

### 5.3. DESCRIPTION DE L'ETUDE

Le but de cette étude est de comparer les performances des précédentes techniques basées sur le stacking et l'interaction tactile pour une tâche de création de tableau croisé dynamique.

Tâche et instructions

La tâche impliquait deux étapes consécutives : sélectionner une plage de cellules puis créer un tableau croisé dynamique en remplissant correctement les zones de l'assistant.

Pour chaque essai, les participants devaient d'abord sélectionner une plage de cellules de 50 lignes x 12 colonnes.

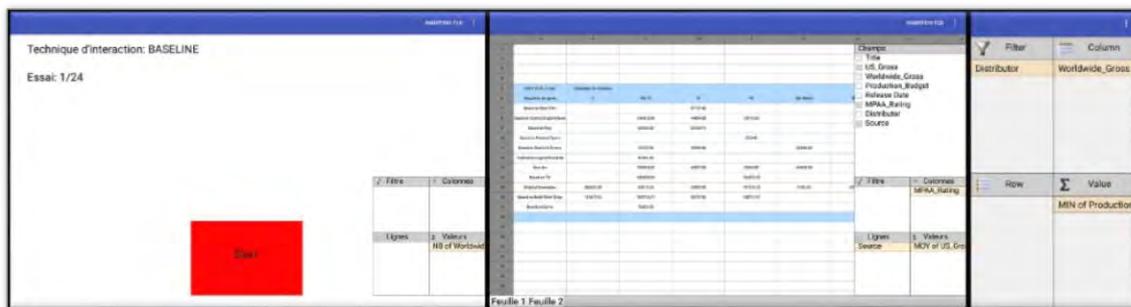


Figure 78. Écran principal de l'application (Gauche), Tableau croisé dynamique avec la technique Tactile (Milieu) et Tableau croisé dynamique sur le smartphone (Droite).

Après la sélection, les participants devaient ouvrir l'assistant du tableau croisé dynamique en utilisant un bouton de la barre de menu, dans le coin supérieur droit de la tablette pour la technique Tactile, et sur le smartphone pour les techniques basées sur le stacking. Ensuite, ils devaient remplir les zones de l'assistant du tableau croisé dynamique selon les instructions données (Figure 78 - Gauche). Chaque essai consistait à remplir une zone (Valeur), deux zones (Valeur et Ligne), trois zones (Valeur, Ligne et Colonne) ou quatre zones (Valeur, Ligne, Colonne et Filtre) de l'assistant avec un élément (Figure 78 – Milieu et Droite selon la technique d'interaction). Dans certains essais, ils ont également dû modifier la fonction d'agrégation (Somme, Nombre, Moyenne). Ce protocole est inspiré de celui proposé par Jo et al. [90].

Les participants ont été invités à terminer la tâche aussi rapidement et précisément que possible. Les instructions de la tâche étaient affichées sur l'écran de la tablette avant chaque essai et sur un écran d'ordinateur secondaire pendant l'essai. Les instructions représentaient un assistant de tableau croisé dynamique pré-rempli avec les éléments requis et configuré avec une fonction d'agrégation spécifique. Un bouton tactile permettait à l'utilisateur de démarrer l'essai. Une fois l'assistant correctement rempli, une validation automatique était déclenchée après 500ms. L'objectif de ce délai était de prévenir la validation lorsque l'utilisateur affecte une valeur à un champ par erreur.

Nous avons utilisé le même ensemble de données que dans [90], composé de différents attributs de films (genre, notation MPAA, etc.), leur budget, le total brut des profits générés et une note d'appréciation. Nous avons réduit l'ensemble de données à 200 lignes et 12 colonnes comme dans [90].

## Matériel

Nous avons utilisé la même tablette et le même smartphone que dans notre étude précédente. Nous avons mis à jour les deux appareils avec Android 7.1 pour exécuter notre code de tableau croisé dynamique (qui nécessitait Java 8). Nous avons également utilisé un écran d'ordinateur (16", 1366x768 pixels) pour afficher en permanence les instructions lors d'un essai. Le smartphone était équipé de notre coque en cuivre (1er prototype) et la tablette était équipée d'une protection d'écran en plastique. Sur le smartphone était lancée une application d'assistant de tableau croisé dynamique et une application de tableur était lancée sur la tablette. Les deux applications ont été développées spécifiquement pour l'étude pour Android. Ces applications nous permettaient de contrôler les nombreux paramètres et options disponibles pour l'utilisateur. Nous avons reproduit l'interface, la disposition des éléments, l'interaction tactile et la taille des cellules (284x53px) de Google Spreadsheet comme fait dans [63] ainsi que l'interaction tactile avec les tableaux croisés dynamiques d'Excel sur Surface. L'utilisateur était assis, tenant le smartphone avec sa main dominante lorsque cela était nécessaire, et la tablette posée sur la table en face de lui.

## Participants

Nous avons recruté 12 participants (7 femmes), âgés de 31.7 ans en moyenne (écart-type = 6.8). Deux d'entre eux ont participé à nos études précédentes. Ils étaient tous droitiers et familiers avec les appareils mobiles (7.8 sur une échelle de 0-9), les tableurs en environnement fixe (6.5) et les tableaux croisés dynamiques (6.3).

## Protocole expérimental

Notre étude a suivi un design intra-sujets 4x4x2 avec la Technique d'Interaction (Tactile, Dual-Screen Touch, Stack-and-Touch et Stack-and-Translate), le Nombre de zones de l'assistant à remplir (un, deux, trois ou quatre) et la Fonction d'Agrégation à modifier (oui ou non) en tant que facteurs. L'étude a été divisée en 4 blocs, chaque bloc correspondant à une technique. L'ordre des blocs a été contrebalancé entre les participants au moyen d'un carré latin 4x4. Pour chaque bloc, les participants pouvaient effectuer des essais d'entraînement jusqu'à ce qu'ils se sentent prêts. Ensuite, ils ont réalisé 24 créations de tableaux croisés dynamiques (8 conditions effectuées 3 fois). L'ordre des essais au sein d'un bloc était aléatoire. Les participants pouvaient faire une pause entre chaque essai et

chaque bloc. Au total, nous avons collecté 4 techniques d'interaction x 8 configurations x 3 répétitions x 12 participants = 1152 essais.

Données collectées et analyse statistique

Pour chaque essai, nous avons mesuré le temps total de réalisation de la tâche, ainsi que les temps de sélection et de remplissage de l'assistant (ou temps d'affectation). Nous avons enregistré les erreurs de sélection (c'est-à-dire, lorsqu'un utilisateur sélectionne une plage de cellules incorrecte) et les erreurs d'affectation (c'est-à-dire, affecter un en-tête à la mauvaise zone de l'assistant ou affecter la mauvaise fonction d'agrégation). À la fin de chaque bloc, nous avons demandé aux participants de remplir un questionnaire SUS et nous avons recueilli des retours informels. Les participants ont également été invités à évaluer leur appréciation générale (j'aime / je n'aime pas) de la technique d'interaction sur une échelle de Likert à 7 points. L'expérience a duré en moyenne 78 minutes. Seuls les résultats principaux sont présentés ci-après. Une analyse statistique plus détaillée peut être trouvée dans l'annexe section 2.3.

## 5.4. RESULTATS

Concernant l'étape de sélection, les temps mesurés confirment nos résultats obtenus lors de la première expérimentation. La sélection d'une plage de cellules avec l'interaction Tactile est plus lente que la sélection basée sur le stacking (Table 5) : en moyenne, la sélection Tactile nécessitait 14,4% plus de temps qu'avec nos techniques d'interaction.

	Moyenne (s)	IC.Min	IC.Max
<b>Tactile</b>	10.6	10.2	10.9
<b>Dual-Screen Touch</b>	9.2	8.4	9.7
<b>Stack-and-Touch</b>	9.2	9.0	9.8
<b>Stack-and-Translate</b>	9.2	8.5	9.6

Table 5. Temps de sélection (en secondes) et intervalle de confiance par technique d'interaction.

Les résultats de cette étude montrent que la technique Stack-and-Translate permet de remplir un assistant de tableau croisé dynamique (étapes 2 et 3) plus rapidement que les autres techniques (7.3s contre 8.3s pour le Dual-Screen Touch, 8.7s pour Stack-and-Touch et 9.5s pour Tactile, Figure 79). En termes d'utilisabilité, Stack-and-Translate et Dual-Screen Touch obtiennent de meilleurs scores SUS que les autres techniques (Figure

80). Par conséquent, Stack-and-Translate semble être la meilleure technique d'interaction pour remplir un assistant de tableau croisé dynamique.

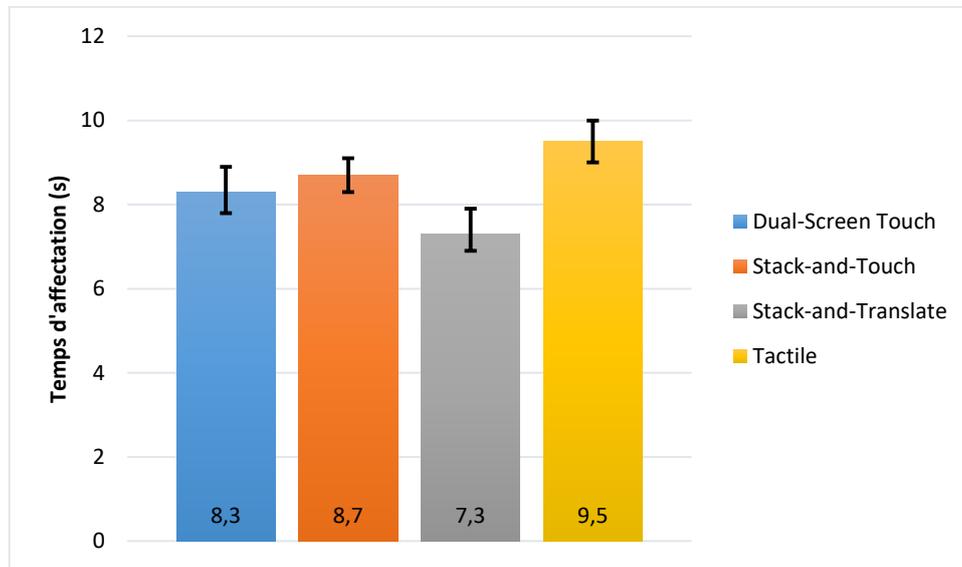


Figure 79. Temps d'affectation par technique d'interaction (en ms, 95% CI).

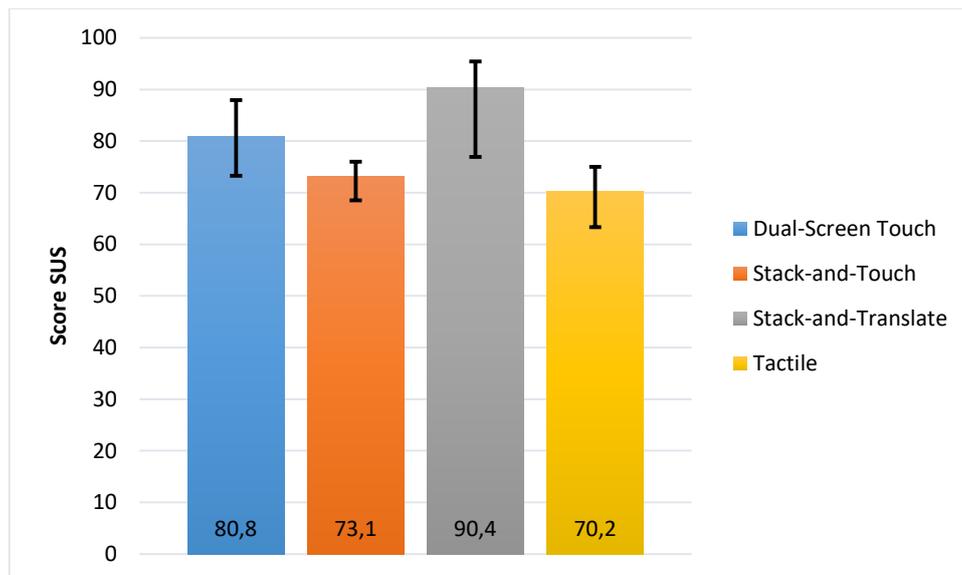


Figure 80. Score SUS par technique d'interaction (95% CI).

## 6. SCÉNARIO D'UTILISATION

L'utilisation du smartphone comme dispositif complémentaire pourrait facilement être insérée dans une application existante pour répondre à des difficultés d'interaction courantes. Par exemple, considérons un analyste qui a effectué une analyse de données avant une réunion. La présentation de son analyse, au cours d'une réunion, génère d'autres hypothèses : il a ensuite besoin d'explorer davantage les données au cours de la réunion, ce qui est fastidieux sur les tablettes. Les smartphones, omniprésents dans les milieux personnels et professionnels, peuvent alors être utilisés pour faciliter l'interaction. Cette situation d'interaction, commune à de nombreux métiers, sert de cadre pour notre travail. Les smartphones peuvent servir d'interacteurs principaux avec la tablette, étendant à la fois les degrés de libertés d'entrée et l'espace d'affichage, en faisant un candidat unique pour générer une analyse plus approfondie des données.

Dans ce contexte, en plus de la création de tableaux croisés dynamiques, le smartphone permettrait de créer et afficher des représentations graphiques des données, libérant ainsi de l'espace d'affichage sur la tablette. L'utilisation du stacking permettrait alors de transférer les graphiques générés d'un dispositif à l'autre, afin que chaque utilisateur puisse les explorer.

Afin de confirmer ce scénario d'utilisation, nous avons réalisé un second entretien avec les élus que nous avons interviewés. Au cours de cette session, nous avons présenté les différentes techniques d'interaction développées. Nous avons recueilli leurs remarques ainsi que leur appréciation de ces techniques et nous leur avons demandé si notre scénario semblait pertinent pour leur contexte de travail.

Les résultats de cet entretien montrent que notre scénario s'intègre parfaitement dans leur fonction d'élu : il représente les conseils municipaux et intercommunaux dans lesquels les données des collectivités / intercommunalités sont analysées et discutées. Enfin, après avoir manipulé le smartphone et généré quelques tableaux croisés dynamiques, les élus ont trouvé nos techniques pertinentes et faciles à prendre en main. Il a cependant été noté que l'ajout d'une représentation graphique permettant de visualiser les données, après avoir créé le tableau croisé dynamique, était nécessaire et devait être la prochaine étape de ce travail.

Dans le contexte de l'entreprise Berger-Levrault, l'utilisation du stacking pourrait donc être envisagée dans de nombreuses applications. La version mobile du logiciel Atal (outil

de gestion du patrimoine de la collectivité) représente un cas d'application particulièrement pertinent pour ces travaux. Ce logiciel affiche une liste déroulante de l'ensemble des éléments de patrimoine de la collectivité regroupés par catégories (arbres, bancs publics, etc.). Chaque élément est représenté par de nombreuses caractéristiques (p. ex. 47 pour les arbres de la ville de Lyon). L'utilisation du stacking permettrait de faciliter la manipulation de l'ensemble de ces critères tout en permettant leur visualisation sur l'écran du smartphone, libérant ainsi de l'espace d'affichage pour afficher les données sur l'écran de la tablette. De plus, l'utilisation du smartphone permettrait l'affichage de représentations graphiques issues de l'analyse de ces données (p. ex. l'état de santé des arbres d'un quartier). D'autre part, le smartphone pourrait aussi être utilisé comme une Magic Lens et ainsi permettre de filtrer les données de la tablette plus efficacement.

## 7. DISCUSSION ET PERSPECTIVES

### Pistes d'amélioration

La technique d'interaction Stack-and-Drag 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.

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 Stack-and-Drag sont également valables ici. Une autre critique majeure de cette technique d'interaction concerne 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.

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. 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.

### Avantages de l'interaction basée sur le stacking pour tableurs sur tablette

Nos deux études révèlent que l'utilisation de l'interaction basée sur le stacking pour manipuler les données d'un tableur sur tablette est plus efficace que l'interaction tactile

traditionnelle. La manipulation physique du smartphone permet une manipulation directe des données, sans interférer avec les gestes tactiles traditionnels, et raccourcit le chemin d'interaction. De plus, l'utilisation de l'écran du smartphone pour afficher l'assistant du tableau croisé dynamique a été considéré pratique et utile (cf. commentaires informels des utilisateurs). Dans l'ensemble, l'interaction basée sur le stacking semble particulièrement prometteuse pour faciliter l'utilisation des tableurs sur tablette.

#### Extension des possibilités de sélection de cellules actuelles

Dans ces travaux, nous avons seulement exploré des tâches de sélection de cellules qui peuvent être effectuées avec une interaction tactile traditionnelle. Cependant, l'interaction basée sur le stacking pourrait ouvrir de nouvelles possibilités en termes de sélection de cellules. Par exemple, la sélection de plusieurs plages est possible sur les feuilles de calcul dans un environnement de bureautique fixe à l'aide d'une touche de modification (CTRL, MAJ). Cette fonctionnalité pourrait être intégrée sur une tablette en utilisant les différents bords du smartphone, chaque bord correspondant à une sélection de cellule différente (ainsi, quatre types de sélections peuvent être effectués). Une autre solution consisterait à utiliser un geste tactile sur l'écran du smartphone ou sur celui de la tablette comme modificateur pour sélectionner différentes plages de cellules.

#### Limitations de nos études

En premier lieu, nos études sont limitées par notre prototype. En effet, la fabrication du premier prototype (utilisé dans nos études) 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 deux nouvelles coques (l'une imprimée en 3D, l'autre usinée en aluminium) afin de limiter les défauts de la 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. Ces outils permettent de localiser 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 exprimé ressentir une charge cognitive plus faible lors de l'utilisation de la technique Intersection par rapport à l'interaction tactile traditionnelle. Or, le vocabulaire d'interaction offert par le stacking est bien plus important que la portion que nous avons explorée à travers les techniques pour la sélection de cellules et celles pour la création de tableaux croisés dynamiques. Il serait ainsi nécessaire d'évaluer l'impact de gestes complexes (notamment les gestes composés, associant par exemple une rotation à plat du smartphone suivi d'un geste de glisser) 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 défilement et de zoom. En effet, nous avons souhaité limiter nos travaux à la sélection de plage de cellules 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. À titre d'exemple, lors d'une sélection sortant du cadre de l'écran, un utilisateur choisissant de dé-zoomer pourrait avoir des problèmes de précisions plus importants qu'un utilisateur choisissant le défilement. Nous sommes cependant conscients que de telles stratégies pourraient impacter les techniques d'interaction que nous proposons.

#### Perspectives des travaux

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 de cellules, 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. Une suite à ce travail pourrait être de concevoir de nouvelles techniques d'interaction impliquant ces mécanismes complémentaires afin d'en évaluer les performances.

Dans notre étude sur la création de tableaux croisés dynamiques, nous n'avons pas considéré l'ordre ou le filtrage des éléments sur l'assistant. Une suite à ce travail pourrait être d'intégrer ces fonctionnalités ainsi que l'intégration des visualisations qui en découlent, telles que les représentations graphiques et tabulaires des résultats. Ces deux visualisations pourraient être proposées sur l'écran du smartphone en utilisant un geste de balayage gauche / droite. Il serait intéressant d'intégrer ces fonctionnalités à notre application et de les évaluer dans une étude longitudinale avec des utilisateurs experts en tableaux croisés dynamiques (par exemple des responsables de ressources humaines).

Par ailleurs, nos études se focalisent sur les tâches de sélection de plage de cellules et de création de tableaux croisés dynamiques dans un tableur sur tablette. Ces tâches, bien que très utilisées dans un tableur, ne représentent qu'une petite partie de l'ensemble des fonctionnalités disponibles. Une perspective pertinente serait d'étendre l'interaction basée sur le stacking à d'autres tâches réalisées dans un tableur telles que la création et manipulation de graphiques, la mise en forme des cellules, le calcul mathématique ou l'activation de filtres.

En dernier lieu, il serait intéressant d'étudier les capacités de personnalisation apportées par le smartphone. En effet, dans ces travaux, 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. Le smartphone servirait alors d'interacteur universel configurable pouvant être adapté à chacune des applications utilisées sur la tablette ou toute autre surface tactile.

## 8. CONCLUSION DU CHAPITRE 6

Dans ce chapitre, nous avons présenté différentes techniques d'interaction multi-dispositifs pour faciliter l'interaction avec un tableur sur tablette. Nos techniques sont basées sur le principe du stacking d'un smartphone sur l'écran d'une tablette, qui le détecte. Nous avons exploré les gestes basés sur le stacking disponibles pour rejeter les plus inconfortables. Sur la base de ces résultats, nous avons conçu plusieurs coques conductrices qui permettent la détection du smartphone sur l'écran de la tablette. Nous avons ensuite proposé des techniques basées sur le stacking pour deux tâches courantes sur un tableur : la sélection de plage de cellules et la création de tableaux croisés dynamiques. Nous avons montré que les techniques basées sur le stacking sont plus rapides que l'interaction tactile traditionnelle pour la sélection de plage de cellules. De plus, nous avons montré que l'utilisation du smartphone comme interacteur tangible avec un écran tactile complémentaire accélère la création et la manipulation de l'assistant d'un tableau croisé dynamique tout en étant plus utilisable que l'interaction tactile traditionnelle.

# CHAPITRE 7

## APPLICATION INDUSTRIELLE

---

### Structure du chapitre

---

1. CONTEXTE .....	176
1.1. Motivation.....	176
1.2. Structure du chapitre .....	176
2. PROJET TREETECH .....	177
2.1. Contexte .....	177
2.2. Fonctionnalités de l'application.....	177
2.3. Scénario d'utilisation .....	179
2.4. Identification de la problématique .....	179
2.5. Solutions envisagées .....	180
2.5.1. Localiser un arbre.....	181
2.5.2. S'orienter par rapport à l'arbre.....	182
2.5.3. Télémétrie.....	183
2.5.4. Synthèse .....	185
2.6. Description du prototype attendu.....	185
2.7. Développement du prototype et limitations.....	187
2.8. Perspectives .....	187
3. PROJET ATTAL MOBILE .....	189
3.1. Contexte.....	189
3.2. Fonctionnalités de l'application.....	189
3.3. Scénarii d'utilisation .....	190
3.3.1. Observation d'un arbre.....	190
3.3.2. Veille d'information sur un ensemble d'arbres.....	191
3.4. Identification de la problématique .....	191

3.5.	Solutions envisagées .....	191
3.5.1.	Barre de menu .....	192
3.5.2.	Interaction bimanuelle.....	192
3.5.3.	Représentation schématique.....	193
3.5.4.	Accords tactiles .....	194
3.6.	Prototype développé .....	195
3.6.1.	Barre de menu zoomable.....	195
3.6.2.	Barre de menu navigable.....	196
3.6.3.	Triple barre de menu .....	197
3.6.4.	Barre de menu hiérarchique .....	198
3.6.5.	Synthèse .....	199
3.7.	Limitations .....	199
3.8.	Perspectives .....	200
4.	CONCLUSION DU CHAPITRE 7 .....	202

# 1. CONTEXTE

## 1.1. MOTIVATION

Au-delà des contributions scientifiques que nous proposons, le contexte industriel dans lequel ces travaux ont été conduits nous offre l'opportunité de pouvoir proposer des solutions d'interaction adaptées à un contexte d'utilisation réel. L'entreprise Berger-Levrault possède un grand nombre de clients utilisant des logiciels administratifs complexes et contenant de larges volumes de données multidimensionnelles. Ce contexte applicatif nous permet d'étudier les besoins d'utilisateurs réels et de proposer des solutions d'interaction avancées répondant aux limitations des produits actuellement commercialisés par l'entreprise. Dans la continuité de nos travaux, nous avons travaillé sur deux logiciels de l'entreprise actuellement en développement afin d'y intégrer des techniques d'interaction avancées pour en fluidifier l'interaction.

## 1.2. STRUCTURE DU CHAPITRE

Dans la section 2, nous présentons le premier projet sur lequel nous avons proposé des solutions d'interaction avancées : TreeTech. Cette application pour dispositif mobile vise à informatiser une méthode botanique de diagnostic des arbres. Dans la section 3, nous décrivons l'application ATTAL Mobile permettant la gestion du patrimoine d'une collectivité sur dispositif mobile. Puis, nous présentons plusieurs techniques d'interactions et prototypes que nous avons développés pour cette application.

## 2. PROJET TREETECH

### 2.1. CONTEXTE

L'application Attal II permet de gérer le patrimoine d'une collectivité. Il regroupe l'ensemble des éléments de patrimoine tels que les arbres, les bancs publics ou encore les locaux possédés par la collectivité. Le module TreeTech vise à compléter la gestion des arbres en facilitant la collecte d'informations sur leur état de santé. Cette application sur dispositif mobile permet d'exécuter pas à pas le diagnostic d'un arbre. Le diagnostic est réalisé en saisissant des informations textuelles, en réalisant des mesures, en annotant des photos et en réalisant des graphiques représentant l'état de l'arbre. Elle est destinée à plusieurs types d'utilisateurs allant de l'expert en botanique jusqu'à l'employé municipal. Dans le contexte de ces travaux, nous nous focalisons sur la partie de l'application dédiée aux employés municipaux réalisant une tournée d'inspection des arbres de la collectivité afin d'en faire leur diagnostic.

### 2.2. FONCTIONNALITES DE L'APPLICATION

L'application TreeTech est composée de quatre sections. Trois sections sont destinés à des usages et utilisateurs spécifiques : la section Routine est dédié aux employés municipaux et permet d'effectuer un diagnostic général de l'arbre ; la section Expert est dédié aux experts en botanique et permet d'effectuer un diagnostic plus poussé ; la section Patrimoine est dédié aux experts en gestion du patrimoine et d'espace verts et permet d'effectuer le suivi des besoins spécifiques d'arbres patrimoniaux. Enfin, la dernière section permet de consulter les fiches d'observations réalisées à partir de l'application. Dans notre contexte, nous nous focalisons sur la section Routine de l'application.



Figure 81. Onglets de l'application TreeTech.

Cette section est composée de sept onglets. Six onglets doivent être renseignés pour effectuer le diagnostic complet de l'arbre (Figure 81). Le septième onglet permet

d'afficher le récapitulatif des informations entrées dans l'application avant de les valider et de les enregistrer.

Le premier onglet, appelé Dendrologie, permet de définir le nom de l'espèce de l'arbre, ses coordonnées géo-localisées ainsi que des informations générales comme le diamètre de son tronc ou de sa couronne. Un numéro d'identification de l'arbre, choisi par la collectivité, peut aussi être rempli.

Le second onglet, appelé Boussole, permet d'enregistrer l'orientation de l'observateur par rapport à l'arbre. Cette information est particulièrement importante car les problèmes de santé d'un arbre peuvent ne pas être visibles dans toutes les orientations (Figure 82). En complément, cet onglet met en surbrillance les orientations utilisées pour les observations précédentes, ce qui permet de se replacer dans les mêmes conditions qu'une observation donnée (p. ex. pour noter l'évolution d'une maladie).



*Figure 82. Photo d'un arbre malade vue de face et de profil.*

Le troisième onglet, appelé Télémétrie, contient la hauteur de l'arbre et de sa couronne. Actuellement, le remplissage des informations de cet onglet se fait manuellement. Cependant, il est envisagé que cette tâche soit automatisée à l'aide d'une photographie de laquelle serait tirée les informations de télémétrie de l'arbre.

Les quatrième, cinquième et sixième onglets, permettent respectivement de renseigner des informations sur la silhouette de l'arbre, sa densité de feuillage ainsi que de noter des observations spécifiques.

Après avoir décrit les fonctionnalités présentes dans cette application, nous illustrons son usage au travers d'un scénario d'utilisation.

## 2.3. SCENARIO D'UTILISATION

Un employé municipal souhaite effectuer sa tournée de diagnostic des arbres dans une collectivité. Il récupère l'ensemble des identifiants ainsi que des coordonnées GPS des arbres dont un diagnostic doit être effectué en mairie ou via un portail web. Ces informations sont transmises à une application sur son téléphone portable ou sa tablette. Il utilise un GPS afin de s'orienter dans la collectivité. Une fois localisé proche du premier arbre de la liste, il effectue son diagnostic en l'observant et en remplissant les champs de l'application TreeTech. Une fois l'observation terminée, l'employé municipal se déplace vers le prochain arbre de la liste. Cette tâche est répétée pour chaque arbre présent dans la liste. En fin de journée, l'employé municipal transfère les données de la journée dans une base de données globale des observations.

## 2.4. IDENTIFICATION DE LA PROBLEMATIQUE

À partir de ce scénario d'utilisation et de tests de l'application, nous avons mis en exergue plusieurs problématiques. La première problématique est liée à la localisation des arbres sur lesquels un diagnostic doit être effectué. Actuellement, le GPS d'un dispositif mobile (e.g. tablette) est utilisé afin de localiser l'arbre. Bien que les coordonnées de l'arbre puissent être très précisément mesurées, la précision du GPS de l'utilisateur de l'application reste faible : de l'ordre de 10m. Pour se rapprocher de l'arbre, cette précision est suffisante mais pour le localiser, elle peut être trop faible. Dans le cas d'un arbre isolé ou suffisamment éloigné des autres, il est simple de retrouver l'arbre à diagnostiquer. Cependant, de nombreux arbres sont proches les uns des autres (dans un parc, Figure 83), sont alignés et se ressemblent (bord de nationale) ou ne sont pas forcément accessibles à moins de 10m (bord d'autoroute, végétation dense). Ainsi, l'identification précise et rapide d'un arbre en particulier reste une tâche délicate et sujette à erreur.

Une fois l'arbre trouvé, il est nécessaire d'adopter le bon point de vue afin de réaliser une observation comparable à l'une des observations précédentes. En effet, certaines caractéristiques telles que des pertes dans la densité de feuillage ne sont visibles que d'un certain point de vue. Cette seconde tâche pose problème car, bien que l'orientation par rapport à l'arbre soit connue, la distance de l'utilisateur à l'arbre lors de l'observation n'est pas renseignée. Il est donc difficile de se placer à la même position que celle d'une observation précédente, ce qui peut nuire à la capacité de l'employé municipal à diagnostiquer correctement l'arbre.

Une troisième problématique réside dans la difficulté de mesurer la taille de l'arbre à partir d'une photographie : les capteurs intégrés dans un dispositif mobile bon marché ne permettent pas de réaliser cette tâche précisément.



Figure 83. Ensemble d'arbres d'un parking d'une collectivité (tiré de Google Street View).

Ces trois problématiques découlent du manque de capteurs embarqués précis dans le dispositif mobile utilisé pour réaliser le diagnostic d'un arbre. Une solution couteuse mais possédant des capteurs précis permettrait de répondre aux trois problématiques soulevées. Cependant, compte tenu du contexte industriel et économique de ces travaux (les collectivités locales n'ont pas les moyens de fournir des dispositifs mobiles très haut de gamme à leurs employés municipaux), cette solution n'est pas envisageable. En conséquence, notre problématique principale est la suivante : **est-il possible de compenser le manque de précision des capteurs d'un dispositif mobile à l'aide d'une technique d'interaction permettant de guider l'utilisateur pas à pas ?**

## 2.5. SOLUTIONS ENVISAGEES

Pour répondre à ces trois problématiques, nous avons proposé un ensemble de solutions logicielles exploitant l'ensemble des capteurs d'un dispositif mobile. En nous inspirant des travaux autour de la Roly-Poly Mouse, notre objectif était de proposer une solution d'interaction basée sur la rotation et l'inclinaison du dispositif mobile dans l'espace. Dans

un premier temps nous présentons les solutions permettant de trouver un arbre plus rapidement. Puis, nous décrivons les solutions permettant de se placer correctement par rapport à l'arbre pour effectuer une observation. Enfin, nous présentons plusieurs solutions permettant d'améliorer la précision de la télémétrie servant à mesurer la taille de l'arbre. L'objectif de nos solutions est de guider l'utilisateur vers un arbre et les positions des observations précédentes de cet arbre en compensant l'imprécision du GPS.

### 2.5.1. LOCALISER UN ARBRE

Afin de compenser les limitations du GPS, plusieurs approches ont été envisagées.

Une première approche consiste à utiliser une vue 3D (de type Google street-view) dans laquelle l'arbre à diagnostiquer aurait été mis en surbrillance (Figure 84 – Gauche). L'utilisateur pourrait donc se déplacer jusqu'à l'arbre avec un suivi 3D en temps réel pour confirmer son trajet. Une fois sur place, l'arbre ayant été mis en surbrillance dans l'application, celui-ci est plus facilement identifiable. Cependant, cette approche est limitée par deux facteurs. D'une part, pour que cette solution fonctionne, il est nécessaire que chaque arbre sur lequel un diagnostic peut être effectué soit préalablement identifié et mis en surbrillance dans l'application. Cette approche nécessite donc une grande quantité d'annotations. D'autre part, l'aspect visuel d'un arbre peut grandement changer entre deux observations : les observations peuvent être espacées de quelques jours à plusieurs mois, ce qui peut impliquer un changement de saison et donc de l'aspect de l'arbre. Il n'est donc pas toujours simple d'identifier un arbre spécifique à partir de cette approche.



Figure 84. Solutions de localisation d'un arbre, de gauche à droite: par surbrillance, à l'aide de flèches indiquant la direction ou par triangulation.

Une seconde approche consiste à utiliser une flèche de guidage affichée sur l'écran d'un dispositif mobile (Figure 84 – Milieu). L'utilisation du gyroscope, bien plus précis que le GPS, permet de connaître son orientation actuelle dans l'espace. Cette orientation à elle seule ne suffit pas, mais elle permet de se repositionner dans l'orientation de la précédente observation, réduisant ainsi l'espace de recherche de l'arbre à une ligne droite partant de l'utilisateur et passant par l'arbre. Cette solution ne permet pas d'avoir une identification de l'arbre à 100%, mais elle permet de raffiner la précision du GPS afin de réduire le nombre de candidats potentiels.

Une troisième solution se base sur la triangulation afin d'obtenir une position GPS beaucoup plus précise (Figure 84 – Droite). Cette solution consiste à utiliser plusieurs objets urbains spécifiques (boîte aux lettres, panneau de circulation, feu rouge, etc.) gérés par la collectivité afin de trianguler la position de l'utilisateur. Ces objets faisant parti du matériel urbain de la collectivité, il est envisageable pour celle-ci de les avoir précisément localisés préalablement à l'aide d'un GPS à haute résolution. Une fois proche de l'arbre à diagnostiquer, l'utilisateur est notifié des objets urbains proches permettant sa triangulation. Il peut choisir ceux qu'il souhaite utiliser puis utiliser le gyroscope de son dispositif mobile afin de s'orienter par rapport à ces objets. Une fois ceux-ci correctement alignés, il est possible de trianguler la position de l'utilisateur par rapport à ces points et ainsi connaître précisément sa position GPS. Ensuite, une estimation de distance et l'orientation par rapport à l'arbre peuvent être données à l'utilisateur, lui permettant d'identifier l'arbre à diagnostiquer efficacement. Cette solution souffre de trois limitations : d'une part, de nombreux objets urbains doivent être précisément localisés, ce qui implique un travail d'annotation préalable conséquent. D'autre part, les objets urbains ne sont pas toujours présents autour d'un arbre (bord d'autoroute, bord de canal), ce qui rend cette solution impossible à mettre en pratique dans certaines situations. Enfin, cette solution ne permet que d'obtenir une distance et une orientation à suivre pour localiser un arbre. Si l'utilisateur était trop loin de l'arbre ou que plusieurs arbres correspondent aux informations données (p. ex. arbres alignés), la triangulation doit être de nouveau effectuée, rallongeant le temps d'identification de l'arbre.

### 2.5.2. S'ORIENTER PAR RAPPORT A L'ARBRE

Cette tâche est aussi limitée par la précision du GPS qui empêche d'obtenir une distance précise de l'utilisateur à l'arbre.

Une première solution consiste à utiliser la photographie d'une observation précédente afin d'estimer visuellement (en comparant la réalité et la photo) la distance de l'utilisateur à l'arbre ainsi que l'orientation de l'observation (Figure 85 – Gauche). Cette solution permet d'affiner la position GPS de l'utilisateur à une précision de quelques mètres au lieu de 10m. Bien que triviale à implémenter, cette solution possède de nombreuses limitations : l'aspect visuel de l'arbre et de son entourage peut avoir évolué depuis la dernière observation, rendant ainsi le repositionnement complexe et plus la distance à l'arbre de l'observation est élevée (p. ex. arbre inaccessible), plus la précision de cette solution est faible.

Une seconde solution se base sur l'utilisation de la réalité augmentée pour afficher les différents points de vue des observations précédentes (Figure 85 – Droite). Cette technique d'interaction se base sur une position GPS précise obtenue à l'aide de la triangulation décrite précédemment où en se plaçant au pied de l'arbre si celui-ci est accessible (la position de l'arbre étant précisément géo-localisée). Une fois correctement placé, les position et orientations des observations précédentes peuvent être représentées dans l'environnement virtuel à l'aide d'objets 3D décrivant la distance à l'utilisateur ainsi que l'orientation de l'observation.



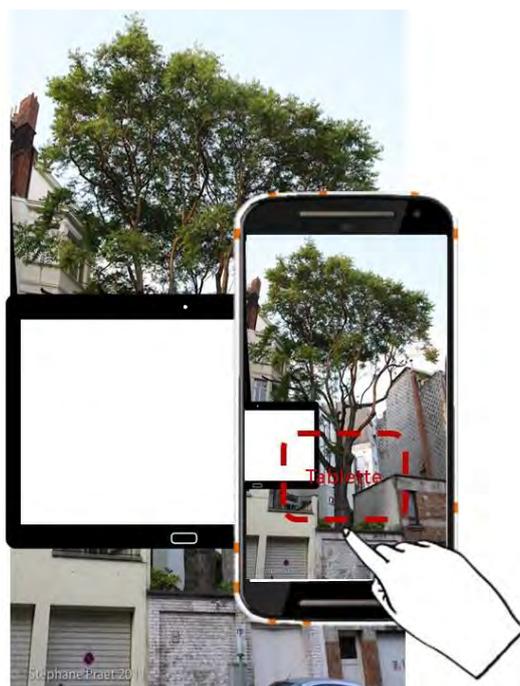
Figure 85. Localisation de la position d'une observation en le comparant à une photographie (Gauche) et indications des précédentes observations (Droite).

### 2.5.3. TELEMETRIE

Cette tâche n'est pas directement réalisable à l'aide des capteurs embarqués dans un dispositif mobile traditionnel. En effet, les dispositifs mobiles bon marché ne sont pas munis de capteurs de profondeur et ne peuvent donc pas identifier la distance à l'arbre, nécessaire pour le calcul de ses dimensions.

Afin de compenser cette limitation, une première solution consiste à utiliser un dispositif mobile disposant de ce type de capteur, tel qu'un casque de réalité augmentée. Cependant, ce type de dispositif mobile ne possède pas une autonomie élevée, peut être coûteuse et être complexe à transporter pour les employés municipaux (risque de casse, acceptabilité en lieu public). Cette solution ne semble donc pas adaptée.

Une seconde solution consiste à utiliser un second dispositif mobile ou objet comme étalon (Figure 86). Cet objet, dont les dimensions sont connues, pourrait être placé dans le champ de vision de la caméra du dispositif mobile contenant l'application. Afin de guider le placement de l'objet, un repaire visuel pourrait être affiché dans l'application. Une fois placé correctement, l'application pourrait utiliser les dimensions et distances de l'objet à partir desquelles calculer celle d'un arbre. Cette solution nécessite néanmoins l'utilisation d'un objet externe complémentaire.



*Figure 86. Calcul de la télémétrie de l'arbre à l'aide d'un objet étalon: une tablette.*

Une troisième solution consiste à utiliser un podomètre afin de calculer la distance entre l'arbre et la position de mesure. Après une phase de calibration pour déterminer la longueur d'un pas moyen de l'utilisateur, celui peut effectuer le trajet du pied de l'arbre au point de mesure désiré. À l'aide de la caméra du dispositif mobile et de la distance parcourue, une mesure approximative de la taille de l'arbre peut être calculée. Ce calcul ne peut cependant pas être précis car la mesure de la distance à l'aide d'un podomètre n'est qu'approximative, le pas de l'utilisateur n'étant pas toujours de même distance,

particulièrement si des obstacles comme une route ou un objet urbain doivent être franchis. De plus, cette approche nécessite que l'utilisateur puisse accéder physiquement à l'arbre et à une distance suffisante de celui-ci pour en mesurer la taille : tous les arbres ne sont pas accessibles (bord d'autoroute), cette approche peut donc être impossible à mettre en œuvre.

#### 2.5.4. SYNTHÈSE

Pour les trois tâches que nous avons défini (localisation d'un arbre, orientation par rapport à l'arbre et télémétrie), nos solutions présentent un certain nombre d'avantages et d'inconvénients. Nous résumons ces informations dans le tableau ci-après (Table 6).

Tâche	Solutions	Avantages	Inconvénients
Localisation d'un arbre	Vue 3D	Aide visuelle à la recherche, familiarité des utilisateurs avec Google Street View	Nécessite une base de données annotée pour chaque arbre
	Flèches de guidage	Donne l'orientation de l'arbre	Peu précis
	Triangulation	Précision élevée	Nécessite la présence d'objets urbains identifiés
S'orienter par rapport à l'arbre	Photographies	Aide visuelle à la recherche	Difficulté de comparaison (saisons, travaux, etc.), précision faible
	Réalité augmentée	Précision élevée	Nécessite une position GPS précise (triangulation)
Télémétrie	Caméra de profondeur	Précision élevée	Utilisable uniquement sur les dispositifs mobiles haut de gamme
	Objet étalon	Précision élevée	Nécessite un objet étalon spécifique
	Podomètre	Peu coûteux, capteur embarqué dans la majorité des dispositifs mobiles	Peu précis, inutilisable si l'objet urbain est inaccessible

Table 6. Tableau comparatif des avantages et inconvénients des solutions envisagées pour chaque tâche.

## 2.6. DESCRIPTION DU PROTOTYPE ATTENDU

À partir de l'ensemble de ces solutions alternatives, nous proposons une technique d'interaction permettant de faciliter la localisation d'un arbre ainsi que des observations précédentes de celui-ci. Celle-ci se base sur la combinaison de l'ensemble des capteurs du smartphone afin de compenser les limitations du GPS embarqué (Figure 87). Nous proposons aussi une solution permettant de mesurer les dimensions d'un arbre.



*Figure 87. Scénario complet d'utilisation de notre technique d'interaction, de gauche à droite : déplacement jusqu'à 10m de l'arbre à l'aide du GPS embarqué, triangulation à l'aide d'objets urbains, affichage de la direction de l'arbre par rapport à la position actuelle (flèche rouge) et position des précédentes observations et mise en évidence de l'arbre à l'aide de photos des précédentes observations.*

Une fois l'application lancée, une carte 2D représente la position de l'utilisateur ainsi que de l'arbre à diagnostiquer. Nous faisons le postulat initial que des coordonnées GPS précises sont disponibles pour fournir la position de l'arbre sur la carte. La position de l'utilisateur est récupérée par les capteurs embarqués du dispositif (GPS).

Une fois approché à moins de 10m de l'arbre (niveau de précision du GPS), un second mode s'active. Nous utilisons la troisième solution présentée ci-dessus : la triangulation à l'aide d'objets urbains. L'utilisateur voit apparaître un ensemble de coordonnées GPS et d'orientation d'objets urbains environnants. Il peut ensuite en choisir plusieurs et se placer correctement par rapport à ceux-ci à l'aide des capteurs d'orientation du dispositif mobile. Une fois la position GPS affinée à l'aide de la triangulation, le dispositif mobile est utilisé comme appareil de réalité augmentée. Sa caméra s'active et s'affiche sur l'écran en complément d'informations sur la distance, et la direction de l'arbre à diagnostiquer. Cette étape peut être répétée jusqu'à trouver l'arbre. Une fois celui-ci trouvé, l'utilisateur peut choisir de visualiser les précédentes positions et orientations des précédentes observations à l'aide d'une vue en réalité augmentée. Les autres positions sont représentées selon leur distance et leur orientation par rapport à la position de l'utilisateur. Un ensemble de photos de l'arbre à partir des points des précédentes observations permettent de faciliter leur localisation.

Concernant la tâche de télémétrie, nous proposons d'utiliser la seconde solution présentée ci-dessus. Cette solution permet le calcul des dimensions de l'arbre à l'aide d'un objet étalon placé à une distance spécifique.

## 2.7. DEVELOPPEMENT DU PROTOTYPE ET LIMITATIONS

Notre prototype a été développé à l'aide d'Android Studio pour des dispositifs Android en version 5.0 et supérieure. Le prototype a été développé pour fonctionner aussi bien sur smartphone que sur tablette.

En nous basant sur un Samsung Galaxy S4, nous avons donc développé une application de réalité augmentée dans laquelle était placée une cible dans l'espace autour du dispositif. L'objectif était d'orienter le smartphone afin de retrouver l'objet virtuel. Cependant, la précision des capteurs embarqués dans le smartphone était trop faible : l'orientation du smartphone déviait peu à peu lors de son usage, rendant ainsi la localisation précise d'un objet dans l'espace impossible.

En parallèle, la grande déviation de l'orientation du dispositif au cours du temps ne permettait pas d'utiliser la technique de localisation de l'utilisateur par triangulation. En effet, il était impossible d'obtenir une orientation précise du dispositif dans l'espace, et donc de s'orienter par rapport à un ensemble d'objets urbains pour raffiner la précision de localisation. Bien que cette solution n'ait pas pu être testée, nous avons développé un algorithme de raffinement de la position de l'utilisateur une fois correctement orienté par rapport à trois objets virtuels géo-localisés.

Ces étapes étant nécessaires pour la reconnaissance de l'arbre à diagnostiquer et ne fonctionnant pas, nous n'avons pas développé de solution pour la télémétrie : celle-ci nécessite que l'arbre à diagnostiquer soit préalablement identifié.

## 2.8. PERSPECTIVES

Nos solutions d'interaction ne sont pas fonctionnelles car les capteurs embarqués dans les dispositifs mobiles ciblés ne suffisent pas. Ces capteurs sont généralement proposés sur les smartphones haut de gamme actuels (Samsung Galaxy S7 et S8). Ces dispositifs, actuellement très coûteux, verront leur prix baisser dans les prochaines années. Il pourra donc être pertinent de réutiliser notre prototype sur l'un de ces dispositifs afin d'évaluer la performance.

Une solution alternative à ces travaux, mais plus incertaine, réside dans l'attente d'un système de localisation spatial plus performant. En effet, nos techniques d'interaction visent à faciliter la localisation d'un arbre étant donné que le GPS de l'utilisateur possède une précision de l'ordre de 10m. Des systèmes alternatifs tels que le GNSS, ou

prochainement Galileo proposent une précision de l'ordre du mètre (service commercial) ou de moins de 4m (service gratuit). L'utilisation d'un tel système permettrait de grandement faciliter la localisation d'un arbre donné. Cependant, ce système n'est actuellement pas disponible et son déploiement ne doit s'achever qu'aux alentours de 2020.

Enfin, concernant l'étape de télémétrie, il est envisageable que des capteurs de distance tels que des caméras infrarouges soient intégrées dans les dispositifs mobiles. Ces caméras permettant un calcul précis de la profondeur, ils pourraient être utilisés pour calculer la distance de l'utilisateur à l'arbre, permettant ainsi le calcul de sa hauteur à partir d'une photographie. Certains smartphones et tablettes intégrant ce type de capteurs ont été annoncés en vue d'une commercialisation pour l'année 2018. Cette solution semble donc pertinente pour la mesure de la taille d'un arbre par télémétrie.

## 3. PROJET ATTAL MOBILE

### 3.1. CONTEXTE

L'application Attal Mobile est une extension de l'application Attal II. Elle permet la gestion du patrimoine d'une collectivité sur dispositif mobile. Bien que compatible avec un smartphone, cette application est principalement dédiée pour tablette. Cette application mobile contient l'ensemble des éléments de patrimoine (bâtiments, bancs, arbres, etc.) de la collectivité. Elle permet de naviguer dans l'arborescence d'éléments et d'observer ou manipuler leurs caractéristiques. À titre d'exemple, dans le cas d'un arbre, les caractéristiques peuvent être les dimensions de celui-ci, la présence de champignons ou encore la présence de nids d'oiseaux. Elle est destinée à plusieurs types d'utilisateurs regroupant la totalité du personnel en mairie (de l'employé municipal à l' élu). Dans le contexte de ces travaux, nous nous focalisons sur la partie de l'application dédiée aux employés municipaux.

### 3.2. FONCTIONNALITES DE L'APPLICATION

L'application se décompose en trois écrans.



Figure 88. Trois écrans de l'application Attal Mobile.

Le premier écran permet de configurer l'application (Figure 88 – Gauche). Il est composé de trois sections. La première permet d'appeler un web-service afin de télécharger la base de données contenant les éléments de patrimoine de la collectivité ciblée. De plus, il est possible de ne charger qu'une partie des tables de la base de données de l'application. Pour tester différents volumes et caractéristiques du jeu de données, nous avons eu mis en œuvre un simulateur de base de données. Il est ainsi possible de générer une base de test, de définir la taille de cette nouvelle base et de configurer certains aspects de la génération (compression, nombre de lignes dans la base de données, etc.). La troisième

section de l'écran correspond à un terminal dans lequel s'affichent les étapes de chargement des données. Une fois les données chargées ou générées, il est possible de les visualiser à l'aide d'un bouton, faisant ainsi la transition vers le second écran.

Le second écran contient l'ensemble des éléments du patrimoine (Figure 88 – Milieu). Contrairement à l'application en environnement fixe, Attal Mobile ne contient aucune arborescence : les éléments sont affichés les uns à la suite des autres dans une grande liste. Chaque élément est représenté par son code ainsi que son libellé ce qui permet de les identifier. La quantité de données étant particulièrement importante, cette liste utilise un « RecyclerView », module Android permettant de ne charger en mémoire que la partie visible des données, évitant ainsi la surcharge de la mémoire du dispositif. Outre la navigation par défilement, une barre de recherche est disponible en haut de l'écran. Celle-ci permet de rechercher n'importe quelle chaîne de texte contenue dans l'écran, qu'elle fasse parti du code de l'élément ou de son libellé. Pour visualiser les caractéristiques d'un élément et faire la transition avec le troisième écran, il est nécessaire d'effectuer un toucher sur l'élément désiré.

Le troisième écran affiche l'ensemble des caractéristiques de l'élément choisi (Figure 88 – Droite). Ces caractéristiques sont représentées selon trois valeurs : leur nom, leur type et leur valeur. Cet écran affiche aussi le code de l'élément et son libellé. La modification d'une caractéristique peut se faire à l'aide d'un toucher ouvrant un outil de configuration adapté (champ de texte, slider, case à cocher, etc.). Pour retourner à l'écran précédent, un appui sur le bouton retour (triangle orienté vers la gauche) est possible.

### 3.3. SCENARII D'UTILISATION

Nous avons envisagé deux scénarii d'utilisation possibles de cette application.

#### 3.3.1. OBSERVATION D'UN ARBRE

Un employé de mairie souhaite effectuer l'observation d'un arbre. Après avoir trouvé l'arbre, il ouvre l'application pour renseigner les informations qu'il a collectées. On suppose la base de données de la collectivité pré-chargée. Après une recherche de l'arbre dans la liste des éléments de patrimoine, il recherche les caractéristiques concernées et les modifie. En effectuant l'observation de l'arbre, il se rend compte qu'une maladie s'est déclarée. Il se souvient avoir déjà vu cette maladie sur un autre arbre dont il a effectué l'observation. Il recherche donc celui-ci dans l'application et compare les caractéristiques

de celui-ci avec celles de l'arbre actuel. Après avoir trouvé les informations pertinentes, il les reporte dans les champs de l'arbre observé.

### 3.3.2. VEILLE D'INFORMATION SUR UN ENSEMBLE D'ARBRES

Une employée municipale revient de sa tournée d'observation d'arbres. Après être arrivée en mairie, elle souhaite avoir une vue d'ensemble des arbres de la collectivité afin de planifier sa prochaine tournée. Elle effectue donc une recherche dans la base de données à l'aide de plusieurs critères pertinents (coordonnées GPS, caractéristiques physiques, présence de champignons ou de maladie, lien avec une zone de travaux). La liste des arbres validant ces critères s'affiche. Elle parcourt leurs caractéristiques pour obtenir de plus amples informations sur ceux-ci. Puis, elle note les arbres qui lui paraissent pertinent pour sa prochaine tournée en dehors de l'application.

## 3.4. IDENTIFICATION DE LA PROBLEMATIQUE

La problématique principale de cette application réside dans la grande quantité d'informations affichable sur un dispositif mobile. En effet, dans un environnement fixe, il est possible de proposer un filtrage multicritère (ex. le bon coin, Amazon, etc.) afin de faciliter la recherche d'éléments dans une grande quantité d'informations. Cependant, sur dispositif mobile, la saisie de texte nécessaire au remplissage des champs de filtre est une tâche fastidieuse : elle nécessite l'ouverture puis la fermeture d'une fenêtre contenant le clavier virtuel cachant une partie des données. De plus, l'ensemble de ces champs de filtre nécessite un espace d'affichage conséquent, laissant peu de place pour la donnée. L'exploration des données nécessite par conséquent un aller-retour itératif entre les mécanismes de filtrage et l'affichage des résultats. Il semble donc pertinent de proposer une solution permettant d'effectuer une recherche multicritère efficace adaptée à un dispositif mobile.

## 3.5. SOLUTIONS ENVISAGEES

Pour répondre au besoin d'interaction de cette application, nous avons proposé quatre techniques d'interaction basées sur l'interaction tactile traditionnelle. Ces solutions d'interaction visent à fluidifier la recherche multicritère de l'application. Dans notre contexte, nous nous sommes focalisés sur un certain type d'éléments de patrimoine : les arbres. Pour le développement de nos prototypes, nous utilisons une base de données de

test collectées à partir des données réelles d'une ville française. Celle-ci contient environ 40 000 arbres, chacun représenté par 47 caractéristiques distinctes.

### 3.5.1. BARRE DE MENU

Notre première solution consiste en l'utilisation d'une barre de menu regroupant l'ensemble des caractéristiques d'un arbre (Figure 89). Cette barre est composée d'une icône pour chaque caractéristique. Lors du passage du doigt au niveau d'une icône, un texte décrivant le nom de la caractéristique est affiché. Lorsqu'un filtre est sélectionné, une fenêtre pop-up de configuration est affichée. Celle-ci contient un élément d'interface permettant de configurer le filtre (case à cocher, slider, champ de texte) ainsi qu'une case à cocher pour activer/désactiver le filtre. Lors d'un toucher à l'extérieur de la fenêtre, celle-ci se ferme. Il est donc possible de configurer plusieurs filtres afin d'affiner une recherche. Cette technique d'interaction est très classique, ce qui permettra de faciliter son adoption par les utilisateurs de l'application. Cependant, plusieurs types de barres de menu peuvent être utilisés. Nous en décrivons quatre dans la section prototype développé.

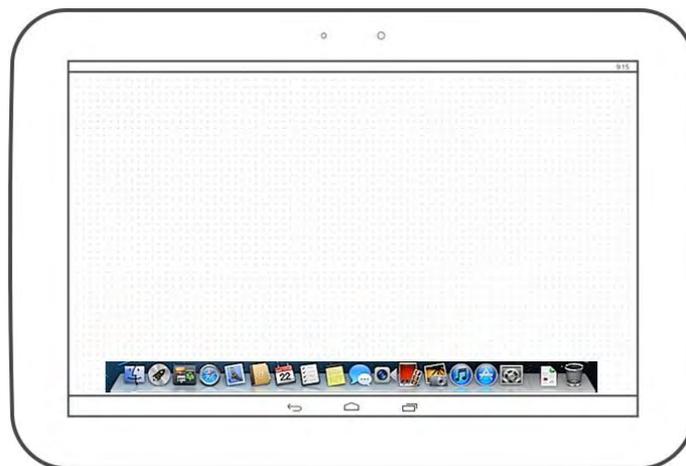


Figure 89. Barre de menu pour les caractéristiques de l'arbre.

### 3.5.2. INTERACTION BIMANUELLE

Notre seconde solution est basée sur l'interaction bimanuelle (Figure 90). Cette solution s'inspire de nos travaux autour de DECO et du stacking : nous envisageons la combinaison des deux mains de l'utilisateur (au lieu de deux dispositifs) afin de faciliter l'interaction avec les données. La main non-dominante permet d'accéder à un menu circulaire servant à sélectionner un sous-ensemble de filtres. Le menu circulaire est composé de trois sections, chacune découpée en trois quartiers concentriques. Ce menu permet donc d'accéder à 9 ensembles de filtres distincts. La main dominante, placée de

l'autre côté de la tablette, permet de choisir un filtre dans le sous-ensemble activé par le menu circulaire. Une fois choisi, une fenêtre pop-up identique à celle décrite précédemment est utilisée pour configurer le filtre. Cette technique d'interaction avancée permet de libérer de l'espace d'affichage en découpant l'ensemble des filtres en catégories. Cependant, son usage semble moins compatible avec un smartphone : l'écran étant plus petit, le menu circulaire contiendra moins d'éléments, réduisant ainsi la pertinence de cette technique d'interaction.

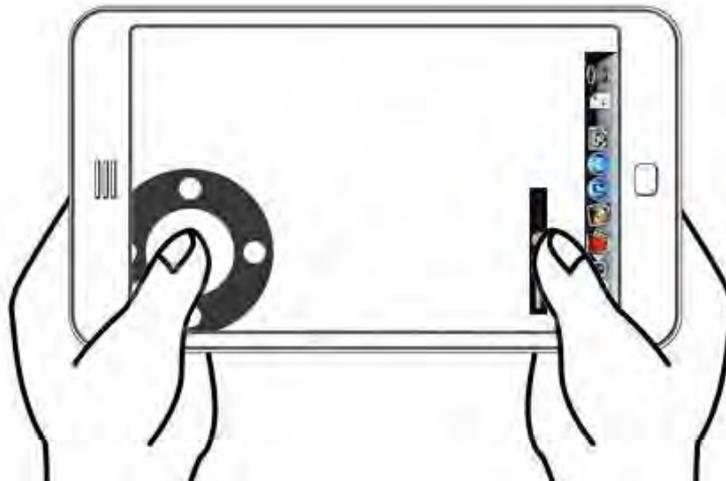


Figure 90. Interaction bimanuelle: la main gauche permet de sélectionner un sous-menu et la main droite permet de choisir une caractéristique et la manipuler.

### 3.5.3. REPRESENTATION SCHEMATIQUE

Notre troisième solution consiste en l'utilisation de plusieurs représentations schématiques (Figure 91). Ces représentations ont pour objectif d'offrir une visualisation graphique des caractéristiques de l'arbre. Cette solution s'inspire des LociPad proposés par Berthelley et al. [16]. La manipulation d'éléments visuels de cette représentation graphique permet d'activer des filtres sur certaines caractéristiques. À titre d'exemple, une image d'arbre pourrait être manipulée graphiquement afin de représenter ses dimensions (taille, largeur du tronc, hauteur et largeur de la couronne). Des gestes de touchers, de zoom ou de défilement peuvent être utilisés pour effectuer ces manipulations. Afin de couvrir l'ensemble des caractéristiques d'un arbre, il est nécessaire d'offrir plusieurs représentations graphiques. Ainsi, un mécanisme de navigation dans ces visualisations est disponible à l'aide d'un geste de swipe gauche ou droit. Cette technique d'interaction permet de n'utiliser qu'une surface faible de l'écran : seule une représentation graphique schématique est visible à un instant donné. L'intérêt principal de cette technique d'interaction est de permettre une manipulation plus directe des

caractéristiques de l'arbre, ce qui peut faciliter le choix des valeurs à appliquer à un filtre : un employé municipal pourrait, par exemple, se servir d'un arbre de référence face à lui pour configurer un filtre et trouver les arbres ayant un profil similaire.

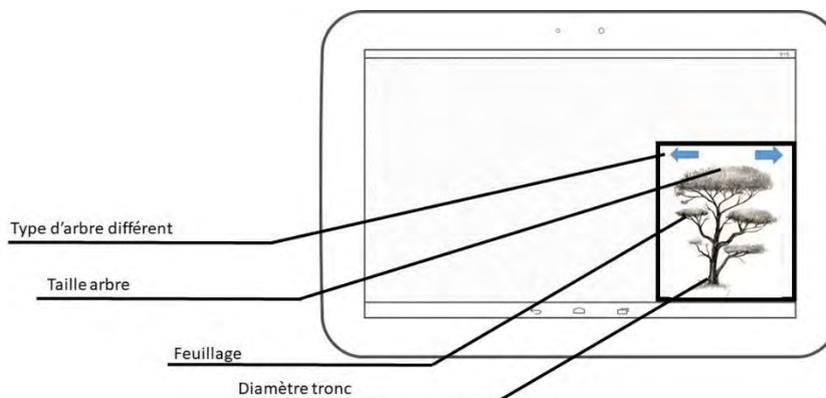


Figure 91. Représentation schématique des caractéristiques d'un arbre.

#### 3.5.4. ACCORDS TACTILES

La quatrième solution d'interaction est basée sur l'utilisation d'accords tactiles (Figure 92). En plaçant le pouce sur l'écran, un menu circulaire apparaît. Un déplacement du pouce permet de sélectionner l'une des seize catégories (8 directions et 2 niveaux d'écartement à la position initiale du pouce). Tout en maintenant le pouce sur l'écran, l'utilisation d'autres doigts permet de sélectionner une caractéristique particulière. À titre d'exemple, une sélection de filtre pourrait être faite comme suit : placement du pouce sur l'écran, déplacement du pouce vers le haut pour sélectionner une catégorie, placement du majeur et de l'auriculaire pour sélectionner une caractéristique donnée. Une fois une caractéristique sélectionnée, une fenêtre pop-up, identique à celle décrite précédemment, s'ouvre et permet de manipuler le filtre. Afin de faciliter l'apprentissage des accords, une représentation visuelle du menu circulaire et des accords disponibles est utilisée. L'avantage principal de cette technique d'interaction réside dans sa capacité à ne pas nécessiter d'espace d'affichage une fois l'utilisateur devenu expert. En effet, une fois les accords connus, il n'est plus nécessaire d'afficher des indices visuels pour permettre à l'utilisateur de faire son choix parmi l'ensemble des menus et accords. Cependant, cette technique peut nécessiter un temps d'apprentissage conséquent avant de maîtriser les différentes combinaisons de doigts.

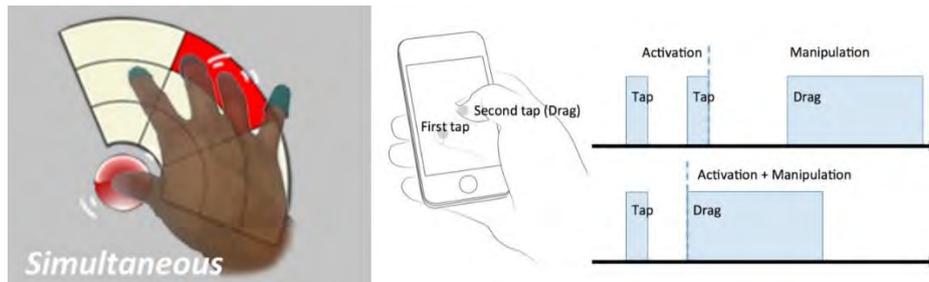


Figure 92. Exemples d'accords tactiles existants dans la littérature (tirés de [9, 76]).

### 3.6. PROTOTYPE DEVELOPPE

Parmi les quatre solutions d'interaction que nous proposons, nous avons choisi de nous focaliser dans un premier temps sur la plus classique : l'utilisation d'un menu tactile. Cette technique d'interaction étant la plus commune, elle permet d'avoir un accès aux possibilités de filtrage multicritère intégré dans l'application à court-termes. Cependant, le nombre de caractéristiques total étant élevé (au nombre de 47), il semble complexe de pouvoir afficher l'ensemble de ces filtres sans nuire à l'espace d'affichage des données. Nous avons donc exploré et développé quatre solutions visant à rendre accessibles l'ensemble de ces filtres.

L'ensemble de ces solutions ont été développées avec Android Studio pour des versions d'Android 5.0 ou supérieure. En complément, nous avons utilisé un module graphique intitulé « HorizontalListView » développé sous licence MIT<sup>4</sup>. Nous décrivons les quatre prototypes développés ci-après.

#### 3.6.1. BARRE DE MENU ZOOMABLE

Notre première solution consiste en une barre de menu placée en bas de l'écran. Cette barre contient l'ensemble des 47 critères représentatif d'un arbre. Cependant, le nombre de critère étant trop important, la taille des icônes représentant ces critères est trop petite pour permettre une interaction. Afin de compenser cette limitation, nous avons développé un mécanisme traditionnel de zoom des icônes de la barre lorsque le doigt de l'utilisateur est proche (Figure 93). Ce mécanisme, de fish-eye, similaire à celui retrouvé dans la barre de menu disponible sur Mac OS, permet d'agrandir les éléments d'interface proche du doigt de l'utilisateur. Dans ce contexte, les icones sont d'une taille suffisante pour permettre l'interaction. Une fois un des éléments sélectionné, une fenêtre pop-up, centrée

<sup>4</sup> Cette licence permet la distribution, l'utilisation, la vente, la reproduction et la modification du contenu sans aucune restriction.

sur l'icône, est affichée. Celle-ci contient un élément d'interface (slider, case à cocher, etc.) permettant de manipuler le filtre de la caractéristique donnée. En complément, un bouton permet d'activer le filtre.

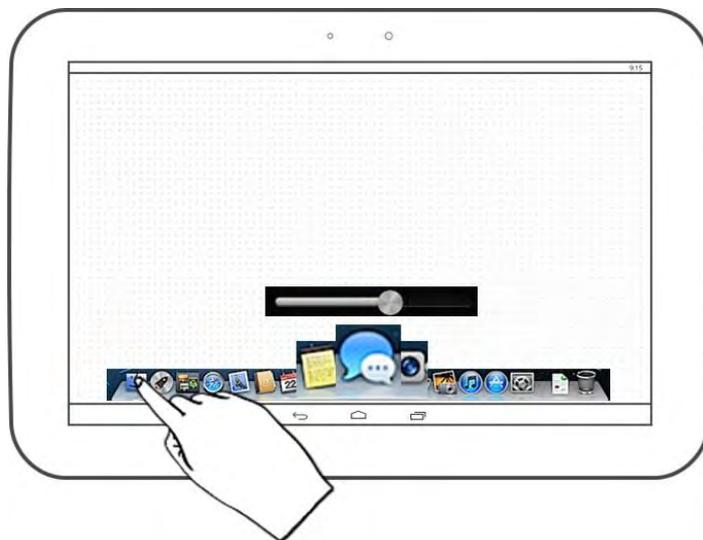


Figure 93. Barre de menu avec zoom selon la position du doigt.

Cette solution possède l'avantage de n'utiliser qu'une bande sur la partie basse de l'écran pour contenir les 47 caractéristiques. Cependant, la sélection d'un élément directement n'est pas possible due à la petite taille des icônes : il est nécessaire de passer son doigt proche de l'icône désiré afin de l'agrandir et ainsi pouvoir le sélectionner.

### 3.6.2. BARRE DE MENU NAVIGABLE

Cette seconde solution est basée sur l'utilisation d'une barre de menu contenant les 47 caractéristiques et dépassant les dimensions de l'écran en largeur. Afin de sélectionner les éléments hors-écran, il est possible de naviguer dans la barre à l'aide d'un geste de swipe gauche ou droit (Figure 94). Les éléments possèdent par défaut une taille qui permet l'interaction (cette taille correspond au zoom maximal d'un icône de la version précédente). Une fois un des éléments sélectionné, une fenêtre pop-up, identique à celle décrite ci-dessus, apparaît afin de configurer le filtre.

Cette solution possède l'avantage de permettre l'interaction directe avec l'une des caractéristiques, sans nécessiter d'approcher au préalable le doigt d'un icône. Cependant, tous les filtres ne sont pas directement accessibles. Il est donc parfois nécessaire d'effectuer en premier lieu un geste de swipe avant d'activer un filtre. De plus, aucune représentation visuelle ne permet de connaître les filtres actuellement hors écran. Lorsqu'un filtre n'est pas contenu dans l'écran, plusieurs gestes de swipe peuvent être

nécessaires avant de trouver l'emplacement de celui-ci dans la barre. Cette limitation peut être réduite si une fois l'utilisateur familier de la disposition des filtres dans la barre : cette technique nécessite un temps d'apprentissage important avant de permettre un accès aux filtres optimal.

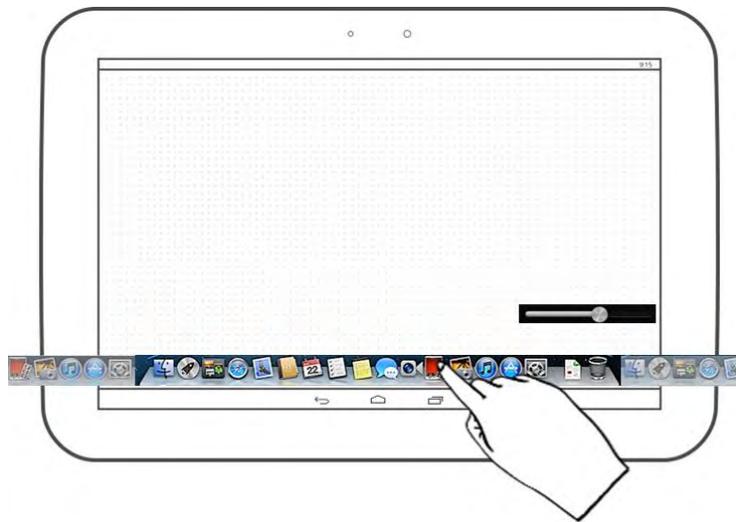


Figure 94. Barre de menu navigable.

### 3.6.3. TRIPLE BARRE DE MENU

La troisième solution consiste à multiplier les barres de menus à l'écran afin d'afficher l'ensemble des caractéristiques (Figure 95). Trois barres sont utilisées : la première est placée sur la gauche de l'écran, la seconde sur le bas et la troisième sur la droite. Lorsqu'un filtre est sélectionné, la fenêtre pop-up s'affiche à proximité. Les icônes sont suffisamment grands pour permettre une interaction directe.

Cette solution permet d'avoir l'ensemble des caractéristiques affichées à l'écran, ce qui facilite la recherche d'un filtre en particulier. Cependant, cette solution nécessite un espace d'affichage conséquent : trois barres de menus sont affichées en simultané, ce qui peut nuire à la visualisation des données et des résultats obtenus suite à l'application d'un filtre.

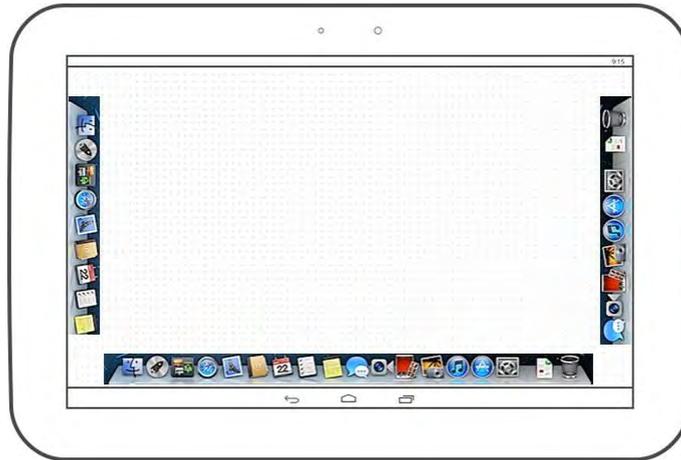


Figure 95. Barre de menu triple.

### 3.6.4. BARRE DE MENU HIERARCHIQUE

La quatrième solution que nous avons développée est basée sur l'utilisation d'un menu hiérarchique (Figure 96). Une barre de menu contenant un nombre restreint d'éléments est affichée sur le bas de l'écran. Les icônes disposent d'une taille suffisante pour permettre l'interaction directe. Une fois l'un de ces éléments sélectionné, un second niveau de menu s'affiche au-dessus de l'icône. Pour sélectionner un icône, deux solutions ont été envisagées : l'appui sur celui-ci ou un geste de défilement vers le haut après avoir placé le doigt sur l'icône. Le sous-menu affiché contient un nombre spécifique d'icônes dépendant de la catégorie de caractéristique choisie. La sélection d'un des éléments du sous-menu déclenche l'apparition de la fenêtre pop-up.

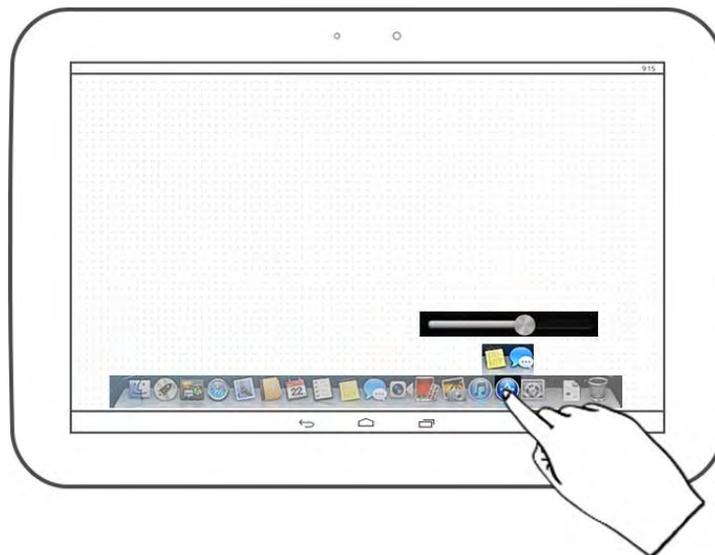


Figure 96. Barre de menu hiérarchique.

L'avantage principal de cette solution est de permettre l'utilisation de l'ensemble des filtres dans un espace d'affichage restreint. Cependant, cette solution nécessite une interaction en deux temps (sélection d'un menu puis d'un élément du sous-menu), ce qui peut allonger l'interaction. Enfin, cette solution repose sur la possibilité de pouvoir classer en catégories équilibrées l'ensemble des caractéristiques de l'arbre. Or, bien que cette solution puisse fonctionner dans ce contexte, il n'est pas obligatoire que l'ensemble des éléments du patrimoine puissent être catégorisés de la sorte.

### 3.6.5. SYNTHÈSE

Nous résumons les avantages et inconvénients de nos quatre techniques d'interaction dans le tableau ci-après (Table 2).

Technique d'interaction	Nombre de caractéristiques gérées	Inconvénients
Barre de menu zoomable	≈30	Limitation du nombre de caractéristiques total et petite taille des icônes non sélectionnés
Barre de menu navigable	∞	Interaction allongée (swipe + sélection) et recherche des caractéristiques nécessaire
Triple barre de menu	≈40	Espace d'affichage utilisé par le menu important
Barre de menu hiérarchique	∞	Nécessite une hiérarchie équilibrée dans les caractéristiques

Table 7. Tableau comparatif du nombre de caractéristiques gérées et des inconvénients de chacune des barres de menu envisagées.

## 3.7. LIMITATIONS

L'une des limitations principale de nos techniques d'interaction réside dans l'espace d'affichage qu'elles utilisent. Bien que certaines de nos techniques ne nécessitent que l'utilisation d'une faible partie de l'écran, la fenêtre de pop-up ainsi que les éléments d'interface permettant la saisie (clavier virtuel, slider, case à cocher, etc.) prennent une part importante de l'écran. Nous proposons en perspective de ces travaux une solution permettant de limiter la quantité de ces éléments d'interface.

D'autre part, le nombre de caractéristiques utilisées dans notre prototype reste limité à un seul contexte d'application : les arbres d'une ville française spécifique. Dans l'application Attal II, chaque élément de patrimoine peut être représenté par le nombre de caractéristique paramétrable. Cette généralité de l'application est indispensable dans un contexte où les différentes collectivités clientes ne gèrent pas nécessairement de la même manière leurs objets du patrimoine. Il est donc possible que d'autres objets urbains

possèdent un nombre plus élevé de caractéristiques. Or, certaines de nos techniques d'interaction ne permettent que peu d'augmentation de ce nombre, notamment la triple barre de menu. Nos solutions sont donc pour l'instant difficilement généralisables à tous les contextes d'utilisation de l'application.

### 3.8. PERSPECTIVES

Une première perspective de ces travaux vise à réduire l'espace d'affichage utilisé par la fenêtre pop-up et les éléments de manipulation du filtre associé à une caractéristique. Pour ce faire, nous envisageons la manipulation directe d'une représentation graphique de données. Après la sélection d'une caractéristique, la répartition des données selon cette caractéristique pourrait être représentée graphiquement (graphique en barre, en camembert, histogramme de distribution, etc.). Une interaction directe avec la représentation graphique pourrait permettre de configurer le filtre à appliquer. Par exemple, après avoir choisi une caractéristique de type booléen, une représentation graphique pourrait représenter le nombre d'éléments de patrimoine en deux catégories. La sélection d'une de ces catégories permettrait d'activer le filtrage selon ce critère. Pour les valeurs numériques (p. ex. taille de l'arbre), l'utilisation d'un intervalle pourrait être envisagée.

Dans un second temps, il pourrait être pertinent d'intégrer une possibilité de prévisualisation du résultat de l'application d'un filtre. Une telle prévisualisation permettrait de faciliter l'application d'un filtre en proposant un rendu graphique du résultat attendu. Ainsi, certaines étapes du filtrage pourraient être évitées. Par exemple, savoir qu'aucun arbre de la base de données ne dépasse les 15m permettrait d'éviter l'activation puis l'annulation de ce filtre.

D'autre part, la gestion d'un historique d'application des filtres pourrait être pertinente. Cet historique permettrait de connaître l'ensemble des filtres actuellement appliqués afin de les désactiver mais elle permettrait aussi de modifier certains critères des filtres déjà actifs. Enfin, l'affichage de cet historique permettrait de mieux prendre en compte l'impact de l'ordre d'application des filtres sur le résultat final obtenu : en réduisant le nombre d'éléments du résultat, certains filtres peuvent impacter le nombre de catégories restantes des autres caractéristiques.

Une quatrième perspective à ces travaux consiste à étendre les techniques d'interaction que nous avons proposées en utilisant des modalités d'interaction avancées. Les solutions

que nous avons proposées sont basées sur l'interaction tactile traditionnelle, ce qui les rend facilement développable et déployable dans l'application Attal Mobile. Cependant, l'interaction tactile avec des menus WIMP sur dispositif mobile est limitée par de nombreux paramètres : petite taille de l'écran, occultations induites par l'interaction tactile et vocabulaire d'interaction limité. Afin de palier à ces limitations, nous envisageons deux catégories de techniques d'interaction basées sur la combinaison de dispositifs : l'interaction en l'air autour du dispositif ou l'utilisation du paradigme du stacking<sup>5</sup>. Dans les deux cas, l'utilisation d'un dispositif complémentaire permettrait d'étendre le vocabulaire d'interaction et d'étendre la surface d'affichage en y ajoutant un écran supplémentaire.

Nous envisageons l'utilisation de l'interaction en l'air afin de configurer les filtres appliquées aux caractéristiques. Dans ce contexte, différentes zones autour de la tablette pourraient être associées à des caractéristiques différentes. Le placement du smartphone dans ces zones permettrait d'afficher sur son écran la caractéristique à filtrer ainsi que ses différentes valeurs. L'utilisation de l'écran du smartphone permettrait de configurer la valeur à appliquer au filtre.

Concernant l'utilisation du stacking, celui-ci pourrait permettre de sélectionner une caractéristique spécifique selon la position ou l'orientation du smartphone sur l'écran de la tablette. Puis, une manipulation physique du smartphone, par exemple à l'aide d'une translation, permettrait de configurer le filtre. Enfin, le déplacement du smartphone sur la tablette permettrait de visualiser l'arborescence de filtres.

---

<sup>5</sup> Le stacking consiste à la mise en contact d'un dispositif complémentaire (un smartphone) avec l'écran d'un dispositif support (la tablette) qui le détecte.

## 4. CONCLUSION DU CHAPITRE 7

Dans ce chapitre, nous avons présenté deux projets de l'entreprise Berger-Levrault dans lesquelles nous avons pu proposer des solutions d'interaction classiques et avancées afin de fluidifier l'interaction. Dans le premier projet, TreeTech, nous avons proposé une technique basée sur l'utilisation de la réalité augmentée et de divers capteurs afin de faciliter la localiser d'un objet dans un environnement 3D. Bien que les limitations techniques des dispositifs mobiles utilisés ne permettent pas d'aboutir à un prototype fonctionnel, celles-ci ont été identifiées. L'évolution des capteurs embarqués dans ces dispositifs permettra le développement de notre prototype dans les prochaines années.

Dans le second projet, Attal Mobile, nous avons proposé un ensemble de solutions d'interaction afin de faciliter la tâche de filtrage multicritère sur dispositif mobile. Nous avons développé quatre variantes d'une barre de menu contenant les filtres à appliquer sur les données. Ces prototypes, basés sur l'interaction tactile traditionnelle, représentent une première étape permettant de manipuler la grande quantité de données contenues dans l'application. Dans un second temps, nous identifions deux solutions d'interaction avancées, basées sur l'interaction en l'air ou sur le paradigme du stacking afin d'étendre les capacités d'interaction et de visualisation actuellement proposées par l'application Attal Mobile.



# CHAPITRE 8

## CONCLUSION

---

Structure du chapitre	
1. RÉSUMÉ DES CONTRIBUTIONS .....	206
1.1. Contribution théorique .....	206
1.2. Contributions pratiques .....	207
1.3. Synthèse .....	209
2. PERSPECTIVES .....	210
2.1. Perspectives à moyens termes .....	210
2.2. Perspectives à longs termes .....	211
1. RÉSULTATS DÉTAILLÉS : LA ROLY-POLY MOUSE .....	218
1.1. Étude 1 : Amplitude et stabilité des gestes .....	218
1.1.1. Amplitude des gestes .....	218
1.1.2. Stabilité des gestes .....	219
1.1.3. Fatigue .....	220
1.2. Étude 2 : Pointage 2D .....	220
1.2.1. Temps de réalisation .....	220
1.2.2. Taux d'erreur .....	221
1.2.3. Débit et modélisation .....	221
1.2.4. Préférence utilisateur et fatigue .....	221
1.3. Étude 3 : Manipulation 3D .....	222
1.3.1. Pointage 3D .....	222
1.3.2. Rotation 3D .....	222
1.3.3. Préférence utilisateur et utilisabilité .....	222
1.3.4. Fatigue .....	223
2. RESULTATS DETAILLES : LE STACKING .....	224

2.1.	ÉTUDE PRÉLIMINAIRE : EXPLORATION DE GESTES .....	224
2.2.	ÉTUDE 1 : SÉLECTION D'UNE PLAGE DE CELLULES.....	224
2.2.1.	Temps de sélection .....	224
2.2.2.	Utilisabilité et préférence utilisateur .....	226
2.3.	ÉTUDE 2 : TABLEAUX CROISÉS DYNAMIQUES .....	228
2.3.1.	Temps de sélection (étape 1).....	228
2.3.2.	Temps d'affectation (étape 2 et 3).....	229
2.3.3.	Erreurs .....	230
2.3.4.	Utilisabilité et préférence utilisateur .....	231

Motivés par la complexité des logiciels et par l'évolution limitée des dispositifs d'interaction, les travaux présentés dans ce manuscrit contribuent à la fluidification de l'interaction avec des données multidimensionnelles par le biais de l'augmentation du nombre de degrés de libertés en entrée des dispositifs. Ces travaux sont appliqués au contexte des logiciels du service public, et plus précisément auprès des utilisateurs des logiciels de l'entreprise Berger-Levrault, acteur majeur du développement de logiciels de gestion administrative pour les services publics qui a permis d'identifier des cas d'étude concrets tout au long de nos avancées et a permis un contact avec des utilisateurs finaux.

Pour conclure ce manuscrit, nous rappelons ici les contributions à la fois théoriques et pratiques incluant des développements logiciels, un cadre théorique et des études expérimentales. Puis, nous présentons plusieurs perspectives à moyen et long termes de nos travaux.

## 1. RÉSUMÉ DES CONTRIBUTIONS

### 1.1. CONTRIBUTION THEORIQUE

Les dispositifs à multiples degrés de libertés proposés dans la littérature représentent des solutions ad-hoc, spécifiquement conçues pour des problèmes donnés. La conception de tels prototype nécessite de nombreuses études expérimentales tout au long de leur conception afin d'évaluer leur potentiel et corriger les éventuelles limitations de ces dispositifs (évaluation du facteur de forme, de la séparabilité des degrés de libertés ou encore de la prise en main). De plus, la proposition d'un nouveau dispositif peut engendrer une réticence en termes d'acceptabilité de la part des utilisateurs : ceux-ci doivent acheter un dispositif supplémentaire (surcôt matériel) s'ajoutant aux nombreux dispositifs existants. Du point de vue des concepteurs du dispositif, il est nécessaire de pouvoir proposer une méthode de fabrication à plus grande échelle afin de pouvoir commercialiser leur prototype. Ce processus de fabrication n'est pas toujours possible et les capteurs utilisés pour les prototypes expérimentaux ne sont pas forcément transposables à une utilisation en environnement écologique (p. ex. caméras OptiTrack).

Nous avons exploré dans nos travaux une approche de composition de dispositifs qui permet d'éviter ces barrières de production et d'acceptabilité. Cette approche consiste en la composition de plusieurs dispositifs existants afin d'augmenter leur degrés de libertés en entrée et en sortie. Bien que plusieurs approches se soient focalisées sur le principe de

composition, aucune ne s'attarde particulièrement sur l'aspect physique de cette composition. Or, cet aspect est primordial pour la composition car il caractérise les manipulations physiques offertes par le dispositif résultant.

Notre contribution théorique consiste en la proposition d'un espace de conception visant à caractériser la composition de dispositifs d'un point de vue physique : DECO. DECO s'appuie sur deux axes : l'arrangement physique et la manipulation physique. L'arrangement physique caractérise la disposition physique des deux dispositifs initiaux (appelés éléments) au travers de trois propriétés : la topologie, le type de fusion et la dynamicité. L'axe des manipulations physiques décrit l'ensemble des manipulations détectables par les capteurs du dispositif composé (dispositif résultant). Cet axe est défini par trois propriétés : les effecteurs humains, les actions physiques et l'utilisation temporelle. À partir de DECO, nous avons décrit les dispositifs composés existants et avons mis en évidence les différences entre ceux-ci en termes d'arrangement physique et les conséquences que cet arrangement a sur les manipulations physiques possibles.

## 1.2. CONTRIBUTIONS PRATIQUES

Nous avons proposé trois contributions pratiques visant à ajouter des degrés de liberté à un dispositif pour la manipulation de données multidimensionnelles.

La première contribution consiste en la conception et l'évaluation d'un nouveau dispositif à multiple degrés de libertés : la Roly-Poly Mouse (RPM). Ce dispositif vise à remplacer la souris traditionnelle pour l'interaction avec des données multidimensionnelles. Basé sur une forme arrondie, il offre 6 degrés de libertés (3 translations dont 2 exploitées et 3 rotations). Ces degrés de libertés peuvent être utilisés de manière parallèle ou séparée, ce qui permet de conserver les propriétés de la souris traditionnelle tout en y intégrant des degrés de libertés supplémentaires utilisables lorsque cela s'avère nécessaire. Ce dispositif étant nouveau, nous avons proposé et étudié différents facteurs de forme (Convexe, Concave et Hémisphérique) et avons identifié plusieurs prises en main (Squeeze, Lay et Touch). Puis, nous avons exploré l'amplitude des mouvements disponibles pour ces facteurs de forme et prises en main au travers d'une première étude expérimentale. Nos résultats mettent en exergue les différences entre prises en main et forme supérieure du dispositif : la forme Hémisphérique permet une plus grande amplitude d'inclinaison alors que la forme Convexe possède une forme plus proche de la souris traditionnelle, la rendant plus appréciée des utilisateurs. Ce dispositif ayant pour

but de remplacer la souris traditionnelle, nous avons évalué ses capacités de pointage. Nos résultats montrent que la RPM, bien que légèrement moins précise et efficace que la souris, est un dispositif adapté au pointage 2D. Enfin, nous avons exploré l'utilisation de la RPM dans le cadre d'une tâche nécessitant plus de 2 degrés de libertés : la manipulation d'objet 3D (en translation et en rotation). Notre étude expérimentale établit que la RPM permet d'effectuer des translations d'objets 3D plus rapidement qu'avec une souris spécialisée pour l'interaction 3D : la SpaceMouse.

Notre seconde contribution consiste en la conception d'un dispositif composé, la RPM2, basé sur la combinaison de la souris traditionnelle avec la RPM. L'utilisation de DECO, notre espace de conception, nous a permis de guider la conception de ce dispositif. Dans un premier temps, nous avons exploré différents arrangements physiques pour les deux éléments (souris et RPM) : nous avons fait varier la topologie et le type de fusion liant ceux-ci. Puis, nous avons exploré l'utilisation parallèle des deux éléments afin de fluidifier l'interaction pour une tâche nécessitant plus de deux dimensions : la tâche de RST (Rotate-Scale-Translate). Les résultats d'une étude expérimentale montrent que cette version du dispositif possède des performances équivalentes à celles de la souris traditionnelle. Afin de réduire le problème d'acceptabilité des utilisateurs vis-à-vis d'un nouveau dispositif remplaçant la souris traditionnelle, nous nous sommes tournés vers une version séparée de la RPM2. Dans cette version, la RPM et la souris peuvent être physiquement détachées et ainsi utilisées de manière bimanuelle. Notre étude expérimentale montre que notre prototype permet de réaliser la tâche de RST plus rapidement qu'avec la souris traditionnelle, prouvant ainsi la capacité de DECO à générer un dispositif composé pertinent et efficace.

Notre troisième contribution consiste en la conception d'un ensemble de techniques d'interaction visant à fluidifier l'interaction avec des données multidimensionnelles sur dispositifs mobiles. Notre approche se base sur l'utilisation du paradigme de stacking. Elle consiste en la détection d'une surface interactive complémentaire (dans notre cas un smartphone) sur une surface support (dans notre cas une tablette) à l'aide de son écran capacitif. Pour ce faire, nous avons conçu une coque conductrice résultant de l'impression 3D de la coque d'un smartphone à laquelle des bandes de cuivre adhésives ont été ajoutées. Nos travaux proposent l'utilisation du stacking pour interagir avec un ensemble de données contenues dans un tableur sur tablette. Nous proposons et évaluons trois techniques d'interaction pour la tâche de sélection d'une plage de données dans un tableur

sur tablette. Parmi ces techniques, nous montrons que deux sur trois permettent de réaliser cette tâche plus rapidement qu'avec l'interaction tactile traditionnelle. Enfin, nous proposons et évaluons trois techniques d'interaction pour la tâche de création et manipulation d'un tableau croisé dynamique dans un tableur sur tablette. Nous montrons qu'en plus de permettre d'effectuer cette tâche plus rapidement, l'une de nos techniques est aussi beaucoup plus appréciée par les utilisateurs que l'interaction tactile traditionnelle.

### 1.3. SYNTHÈSE

Afin de fluidifier l'interaction avec des données multidimensionnelles, nous avons proposé un nouveau dispositif à multiples degrés de liberté, la Roly-Poly Mouse. Nous avons montré que ce dispositif permet de réaliser des tâches de pointages 2D efficacement et permet la manipulation de données multidimensionnelles plus efficacement qu'à l'aide d'un dispositif dédié, la SpaceMouse. Ainsi, ce dispositif semble être un candidat pertinent comme remplaçant de la souris traditionnelle pour fluidifier l'interaction avec des données multidimensionnelles. Cependant, cette approche de conception étant fastidieuse et adaptée à un contexte interactif spécifique, nous proposons l'approche de composition de dispositif afin de faciliter la conception de nouveaux dispositifs à multiples degrés de libertés. L'aspect physique de cette composition étant particulièrement important et celui-ci n'ayant été que peu pris en compte par les travaux précédents, nous proposons un cadre de conception, DECO, afin de caractériser cette composition. À l'aide de cet espace de conception, nous avons conçu un nouveau dispositif à multiple degrés de libertés basé sur la combinaison de la souris traditionnelle et de la RPM. Nos études expérimentales montrent que ce dispositif est plus performant que la souris pour la réalisation d'une tâche de RST. De plus, afin de réduire les réticences de futurs utilisateurs vis-à-vis de ce dispositif, nous proposons un prototype dans lequel les deux éléments sont physiquement séparés. Enfin, nous proposons un ensemble de techniques d'interaction basées sur le stacking pour fluidifier l'interaction avec les données multidimensionnelles sur dispositifs mobiles dans un contexte spécifique : l'utilisation d'un tableur sur tablette. Nos techniques d'interaction permettent de sélectionner une plage de données plus rapidement qu'avec l'interaction tactile traditionnelle. De plus, nos techniques de création et manipulation de tableaux croisés dynamiques sont plus rapides et plus appréciées que l'interaction tactile traditionnelle.

## 2. PERSPECTIVES

Dans la dernière partie de ce mémoire, nous présentons tout d'abord des perspectives à moyen terme avant de présenter nos perspectives à plus long terme.

### 2.1. PERSPECTIVES A MOYENS TERMES

Une première perspective à ces travaux évoquée à la fin du chapitre 4 est de développer un prototype embarqué à faible coût pour la Roly-Poly Mouse. Comme nous l'avons explicité durant la conception et l'évaluation de notre dispositif, celui-ci repose sur l'utilisation d'un capteur externe coûteux et lourde à mettre en place (difficilement déplaçable et avec une calibration indispensable pour la motion capture). Bien que nous ayons proposé plusieurs variantes de ce dispositif plus embarquées, celles-ci se basent toujours sur des capteurs externes (Polhemus Patriot) ou sont limitées dans leurs capacités de détection (la centrale inertielle intégrée ne détecte pas la position). Maintenant que nous avons prouvé la pertinence de ce dispositif à multiples degrés de libertés pour remplacer la souris traditionnelle, l'utilisation de capteurs embarqués et peu coûteux doit être étudiée afin de rendre la Roly-Poly Mouse plus robuste et plus facilement déployable et intégrable dans un contexte d'utilisation réel. Afin d'accompagner le déploiement du dispositif, le développement d'un pilote permettant de traiter les différentes entrées sera nécessaire. En complément, la réalisation d'une étude d'acceptabilité auprès d'utilisateurs sera nécessaire afin d'évaluer le nombre d'utilisateurs potentiels de ce dispositif. Enfin, une étude longitudinale de l'utilisation de la RPM semble nécessaire afin d'évaluer son impact sur les troubles musculo squelettiques. Une version ergonomique de la RPM pourrait être envisagée dans ce contexte.

Une seconde perspective à ces travaux évoquée à la fin du chapitre 5 consiste à étendre notre espace de conception, DECO, en affinant nos propriétés et en y intégrant de nouvelles. La topologie séparée pourrait, en effet, être affinée en y intégrant la notion de spatialisation entre les éléments du dispositif composé. Des travaux en ce sens vont être initiés dans une thèse CIFRE menée dans un contexte d'interaction avec un environnement contenant de multiples écrans spatialisés. Cette modification de DECO permettrait de mieux décrire des environnements multi-surfaces. Cette propriété devra aussi être intégrée dans l'axe manipulation physique afin de permettre la description des interactions en l'air ou autour d'un dispositif. Puis, l'ajout du nombre d'éléments servants à la composition pourrait être intégré. Cette propriété pourrait aussi être utilisée pour

caractériser les environnements multi-surfaces. Enfin, l'utilisation temporelle des éléments peut être raffinée afin de mieux décrire l'utilisation parallèle des dimensions du dispositif composé. Actuellement, cette propriété décrit la possibilité d'utilisation parallèle des éléments composant le dispositif résultant. Or, il est envisageable qu'une partie seulement des dimensions d'un élément puissent être utilisées parallèlement avec celles du second élément. Une granularité plus fine de cette propriété permettrait de mieux différencier les dimensions utilisables en parallèle de celles utilisables de manière séquentielle.

Une troisième perspective à ces travaux évoquée à la fin du chapitre 6 est de réaliser une étude complète de l'utilisation du stacking pour l'interaction avec un tableur. Nos études expérimentales évaluent l'apport du stacking pour fluidifier l'interaction avec un tableur sur tablette pour des tâches de sélection de plage de cellules et de création de tableau croisé dynamique. Or, ces tâches ne représentent qu'un sous ensemble des fonctionnalités disponibles dans un tableur. Une étude des besoins auprès d'utilisateurs de tableurs sur tablette permettrait de mettre en lumière les fonctionnalités utilisées les plus fréquemment. À partir de ces résultats, le développement et l'évaluation de techniques d'interaction basées sur le stacking adaptées à ces tâches serait nécessaire. Puis, la réalisation d'un prototype intégrant l'ensemble de ces techniques d'interaction permettrait d'évaluer l'apport du stacking dans le cadre d'un usage réel d'un tableur sur tablette. Un premier travail en ce sens a été réalisé au cours d'un stage encadré par Berger-Levrault. L'objectif était de trouver une solution technologique permettant d'intégrer la détection du stacking dans un tableur existant afin d'activer des fonctionnalités (mise en page, édition, etc.). Plusieurs solutions, basées sur l'utilisation de tableurs open-source, ont été envisagées mais aucune n'a été développée. De plus, le développement de ces solutions permettrait de faciliter le déploiement du stacking.

Pour finir sur les perspectives à court terme, nous pensons que le développement d'une API intégrant la détection du stacking sur Android permettrait de faciliter la conception de telles techniques. En complément, la conception d'un ensemble de modèles virtuels de coques conductrices adaptées à différents modèles de smartphone pourrait être pertinente.

## 2.2. PERSPECTIVES A LONGS TERMES

Dans le cadre de ces travaux, nous avons proposé la Roly-Poly Mouse comme dispositif à multiples degrés de libertés pour interagir dans un environnement fixe classique. Des

travaux récents ont proposé d'étendre ce cas d'usage aux environnements multi-écrans [157]. Dans ce type d'environnement, l'utilisation d'une souris peut être problématique car ce dispositif ne prend pas en compte la disposition spatiale des surfaces interactives. Or, la prise en compte de cette spatialisation pour concevoir des techniques d'interaction est importante [140]. Dans leurs travaux, Houssein et al. proposent une version améliorée de la RPM appelée TDome. Ce dispositif sert d'interacteur tangible facilitant la transition et le transfert d'information entre différentes surfaces en utilisant leur position spatiale. Cependant, ces deux environnements ne couvrent pas l'ensemble des contextes d'interaction. Par exemple, dans un environnement immersif (p. ex. basé sur un casque de réalité augmentée) ou en mobilité, l'utilisation d'un dispositif tel que la souris ne semble pas adaptée : l'utilisateur peut se déplacer dans l'environnement ce qui rend ces dispositifs inaccessibles. Une utilisation de la RPM sur le corps ou en l'air pourrait être envisagée.

Bien que son facteur de forme offre à la RPM de multiples degrés de libertés et une amplitude de mouvement élevée, l'un des défauts de notre prototype réside dans son incapacité à rester dans un état précis une fois relâché. En effet, le dispositif ayant une base ronde et étant lesté, il oscille après avoir été relâché jusqu'à revenir en position de repos. Une version de la RPM motorisée permettrait d'éviter ces oscillations et de maintenir le dispositif en place. D'autre part, cette version motorisée permettrait d'ajouter de nouvelles propriétés au dispositif telles que son déplacement automatique sur une surface ou l'ajout de vibrations pour offrir un feedback à l'utilisateur.

Dans un second temps, il pourrait être pertinent d'envisager l'extension de notre espace de conception, DECO, en y intégrant la notion de degrés de libertés en sortie. En effet, notre espace de conception se focalise sur la composition de dispositifs visant à augmenter les degrés de libertés en entrée. Dans ce contexte, l'aspect physique est particulièrement important puisqu'il définit l'ensemble des manipulations possibles du dispositif composé. Cependant, nous n'avons pas mis l'accent sur la présence de feedback tangibles (vibrations, retour de force), audio ou visuels (écrans). À titre d'exemple, la prise en compte de la composition physique de dispositifs n'est pas forcément pertinente dans le cadre d'un feedback audio. Dans ce contexte, il semble que plusieurs facteurs tels que la spatialisation de dispositifs (topologie séparée), le type de feedback ainsi que les modalités sensorielles utilisées permettent de mieux caractériser la composition. Enfin, la composition de degrés de libertés en sortie peut induire une perte des degrés de libertés

initiaux sans pour autant nuire à l'utilisabilité ou à la performance du dispositif résultant. Ainsi, la combinaison d'un élément possédant un feedback audio avec un élément possédant un feedback haptique, tel que la LensMouse, n'induit pas forcément la conservation de ces deux feedbacks [204].

Une troisième perspective à longs termes de nos travaux consiste à étudier l'usage du smartphone comme interacteur universel. Dans le chapitre 6, nous mettons en avant son intérêt pour fluidifier l'interaction avec un tableur sur tablette. Il permet d'étendre les degrés de libertés de l'interaction tactile traditionnelle au travers de sa manipulation physique sur l'écran de la tablette et ajoute une surface complémentaire d'affichage permettant de limiter les occultations. Cependant, dans notre contexte, nous n'envisageons l'utilisation du smartphone comme interacteur tangible qu'avec la tablette. Or, celui-ci pourrait être utilisé pour interagir avec de multiples surfaces. Il servirait alors de « pont » permettant de lier les contenus des différentes surfaces d'interaction et de transférer leur contenu. L'avantage de cette solution est triple. D'une part, le smartphone est un dispositif très commun, possédé par un grand nombre d'utilisateurs, ce qui évite le surcout induit par l'achat d'un dispositif dédié. D'autre part, le smartphone est un dispositif personnel et configurable. Ainsi, chaque utilisateur pourrait configurer son smartphone avec un ensemble de fonctionnalités ou de favoris permettant de faciliter l'interaction avec différentes applications ou surfaces. Enfin, le smartphone est équipé d'un grand nombre de capteur, offrant de multiples degrés de libertés et permettant de détecter des gestes en l'air ou l'interaction tactile sur son écran.

Une dernière perspective à long terme réside dans l'intégration des techniques d'interaction proposées dans ces travaux dans les logiciels de l'entreprise Berger-Levrault. En effet, un des objectifs de cette thèse était d'une part, d'identifier les problématiques rencontrées par les utilisateurs des logiciels du service public et d'autre part de proposer des solutions d'interaction adaptées pour répondre à ces problématiques. Dans ces travaux, nous avons identifié qu'une des explications de la difficulté d'interaction avec les logiciels du service public réside dans la trop faible quantité de degrés de libertés offerte par les dispositifs conventionnels. Nous avons donc proposé un ensemble de dispositifs et techniques d'interaction, en environnement fixe et mobile, afin de palier à cette problématique. Cependant, bien que nos études expérimentales restent limitées à des tâches bien spécifiques conduites en environnement contrôlé (laboratoire), elles ont mené à des prototypes illustratifs d'un transfert possible en contexte plus

écologique (cf. chapitre 7). Il pourrait être pertinent d'intégrer l'ensemble de nos dispositifs et techniques d'interaction dans les logiciels commercialisés par l'entreprise. Une étude à long terme visant à évaluer l'apport de ces dispositifs/techniques d'interaction permettrait de vérifier si celles-ci sont suffisantes pour répondre aux problématiques rencontrées par les clients de l'entreprise.



# ANNEXES

---

## Structure du chapitre

---

1. RÉSULTATS DÉTAILLÉS : LA ROLY-POLY MOUSE.....	218
1.1. Étude 1 : Amplitude et stabilité des gestes .....	218
1.1.1. Amplitude des gestes.....	218
1.1.2. Stabilité des gestes .....	219
1.1.3. Fatigue.....	220
1.2. Étude 2 : Pointage 2D .....	220
1.2.1. Temps de réalisation.....	220
1.2.2. Taux d'erreur .....	221
1.2.3. Débit et modélisation .....	221
1.2.4. Préférence utilisateur et fatigue.....	221
1.3. Étude 3 : Manipulation 3D.....	222
1.3.1. Pointage 3D .....	222
1.3.2. Rotation 3D .....	222
1.3.3. Préférence utilisateur et utilisabilité.....	222
1.3.4. Fatigue.....	223
2. RESULTATS DETAILLES : LE STACKING.....	224
2.1. ÉTUDE PRÉLIMINAIRE : EXPLORATION DE GESTES .....	224
2.2. ÉTUDE 1 : SÉLECTION D'UNE PLAGE DE CELLULES.....	224
2.2.1. Temps de sélection .....	224
2.2.2. Utilisabilité et préférence utilisateur .....	226
2.3. ÉTUDE 2 : TABLEAUX CROISÉS DYNAMIQUES .....	228
2.3.1. Temps de sélection (étape 1).....	228
2.3.2. Temps d'affectation (étape 2 et 3).....	229

2.3.3.	Erreurs .....	230
2.3.4.	Utilisabilité et préférence utilisateur .....	231

# 1. RESULTATS DETAILLES : LA ROLY-POLY MOUSE

## 1.1. ÉTUDE 1 : AMPLITUDE ET STABILITE DES GESTES

### 1.1.1. AMPLITUDE DES GESTES

**Rotation** : Un test de Wilcoxon ne révèle aucun effet significatif de la forme du dispositif sur l'amplitude de rotation ( $Z=-0,98$ ,  $p=0,33$ ), nous décrivons donc les valeurs moyennes en considérant à la fois les deux formes supérieures. Un test de Wilcoxon confirme la différence entre les rotations à gauche et celles à droite ( $Z = -3.13$ ,  $p = 0.0017$ ): les rotations à gauche sont, en moyenne, supérieure à celle de droite ( $57^\circ$  contre  $48^\circ$ ) (Figure 97 - gauche). Les résultats sur la prise en main indiquent que Squeeze permet une rotation moyenne plus grande que Touch or Lay ( $66^\circ$  vs.  $45^\circ$ ) et  $48^\circ$ ;  $\chi^2(2) = 24$ ,  $p < .001$ ).

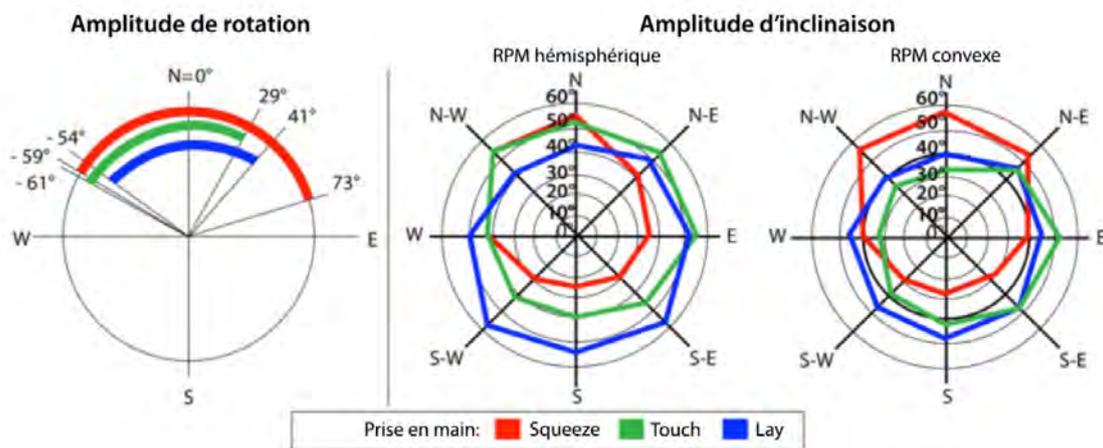


Figure 97. Amplitude de rotation (gauche) par prise en main. Amplitude d'inclinaison par prise en main pour la RPM hémisphérique (centre) et convexe (droite).

**Inclinaison** : Les résultats présentés ci-après décrivent l'amplitude d'inclinaison, c.-à-d. l'inclinaison maximale, en degrés, dans une direction donnée. Un test ANOVA établit une différence significative entre les deux couvercles de la RPM ( $F(1,528) = 34.319$ ,  $p < .001$ ). L'amplitude moyenne maximale d'inclinaison est significativement plus élevée avec la RPM hémisphérique ( $39^\circ$ ) qu'avec la RPM convexe ( $37^\circ$ ). Les amplitudes d'inclinaison sont comprises entre  $27^\circ$  et presque  $60^\circ$  selon la direction d'inclinaison et la prise en main (Figure 97 – centre, droite).

**Translation** : Aucune différence significative n'a été trouvée pour les translations et ce, quelle que soit la forme supérieure, la prise en main ou la direction.

### 1.1.2. STABILITE DES GESTES

**Rotation** : Nous n'avons trouvé aucun effet de la rotation sur la stabilité du dispositif, c'est-à-dire qu'une rotation de la RPM ne produisait aucune inclinaison ou translation involontaires mesurables par nos capteurs. Ainsi, il n'est pas exclu que la rotation du dispositif affecte les inclinaisons ou translations à une échelle plus faible que la précision minimale de notre système de détection.

**Inclinaison** : Les résultats sur la précision de l'inclinaison, c'est-à-dire les déviations entre la direction d'inclinaison effectuée par l'utilisateur et celle demandée par la consigne, ne révèlent pas d'incidence de la forme supérieure de la RPM ( $F(7,528) = 2,68$   $p = 0,09$ ). Globalement, l'écart variait de  $8^\circ$  à  $17^\circ$  pour chaque direction (Figure 98). Ces résultats confirment que 8 directions distinctes peuvent être atteintes sans erreur. En conséquence, il est possible d'envisager d'utiliser l'inclinaison de la RPM pour interagir avec un Marking Menu [ref 17 RPM].

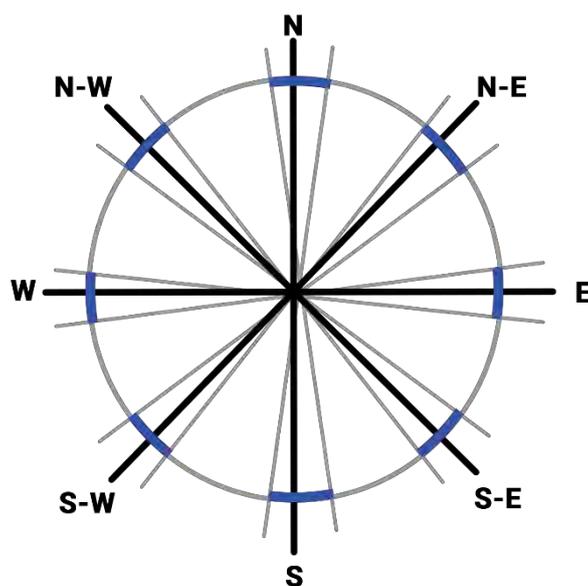


Figure 98. Déviation entre la direction de l'inclinaison demandée et celle réalisée.

**Translation** : Les résultats sur la stabilité de la translation, c'est-à-dire l'amplitude de rotation et d'inclinaison involontaires lors des translations, révèlent qu'il y a en moyenne  $12^\circ$  d'inclinaison et  $15^\circ$  de rotation pendant le déplacement de la RPM. Il n'y a pas de différence significative selon la forme du couvercle du dispositif ( $Z = -4.29$   $p = 0.08$ ). Alors qu'un test de Wilcoxon établit une différence significative entre les translations longues et courtes ( $Z = -2.34$ ,  $p = 0.02$ ), cette différence se limite à  $1^\circ$  pour les

inclinaisons. Enfin, aucune différence significative n'a été constatée entre les différentes prises en main ( $Z = -0.78$ ,  $p = 0.44$ ).

### 1.1.3. FATIGUE

La fatigue moyenne obtenue, mesurée avec l'échelle de Borg [18], est de 11.72 (sur une échelle de 6-20). La forme du couvercle n'affecte pas la fatigue de façon significative. Cependant, la fatigue mesurée lors de l'utilisation de la posture Squeeze est significativement plus élevée qu'avec d'autres postures (Friedman  $\chi^2(2) = 8.46$ ,  $p = 0.01$ ).

## 1.2. ÉTUDE 2 : POINTAGE 2D

Nous présentons d'abord les mesures collectées, c'est-à-dire le temps de réalisation de la tâche ainsi que le taux d'erreur. Puis, nous calculons le débit ou « throughput » (en bits / s).

### 1.2.1. TEMPS DE REALISATION

Un test de Friedman révèle qu'il n'y a pas de différence significative ( $\chi^2(4) = 8,16$ ,  $p = 0.085$ ) entre les cinq techniques d'entrée en termes de temps de réalisation (Figure 99 – Haut-Gauche). Il n'y a pas non plus de différence significative entre les techniques d'interaction pour chaque indice de difficulté ( $p > 0,15$ ).

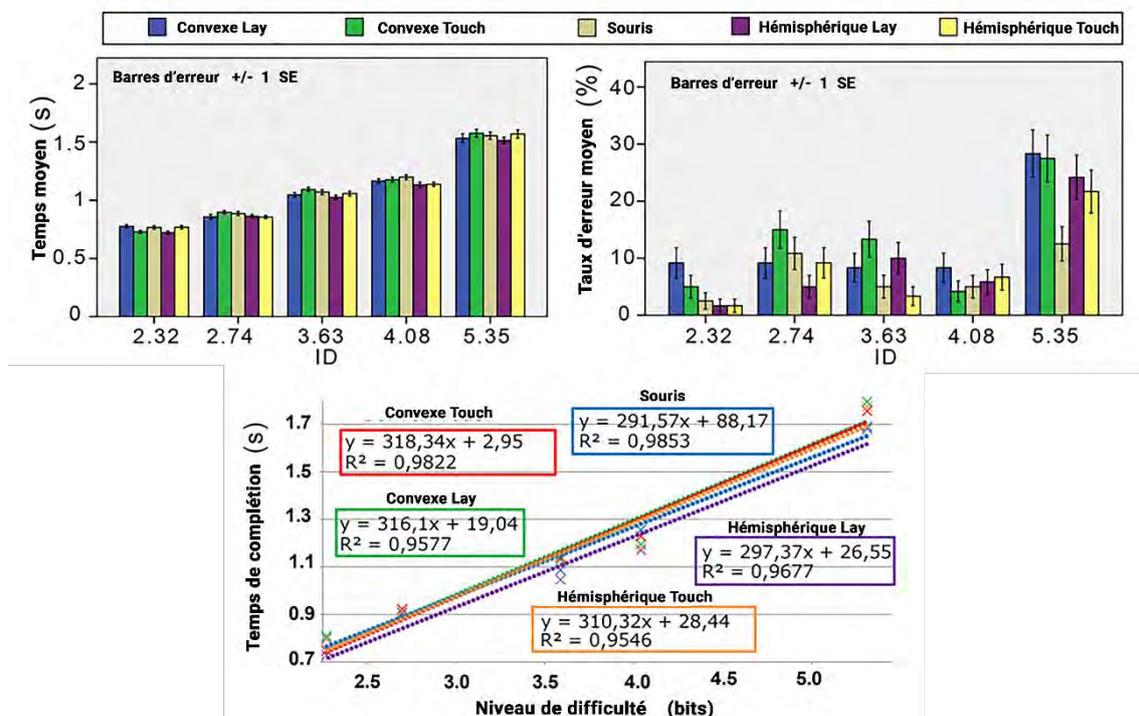


Figure 99. Temps de réalisation moyen (gauche), taux d'erreur (centre) pour chaque technique d'interaction et indice de difficulté. Modèle de Fitts pour chaque technique d'interaction (droite).

### 1.2.2. TAUX D'ERREUR

Nous étudions le taux d'erreur en pourcentage d'essais pour lesquels la première tentative de sélection a été faite en dehors de la cible (Figure 99 – Haut-Droite). Un test de Friedman révèle qu'il y a un effet significatif du facteur technique d'interaction ( $\chi^2(4) = 16.1164, p = 0.0028$ ). Un test de Wilcoxon par paire montre que les techniques basées sur la RPM convexes sont significativement plus sujettes aux erreurs (13,1% d'erreurs) que la souris ( $Z = -20.44, p < 0.001$ ) et la RPM hémisphérique ( $Z = -5.96, p < 0.001$ ). Il n'y a pas de différence entre la souris (7,1%) et les deux techniques basées sur la RPM hémisphérique (8,9%) ( $Z = -2.52, p = 0.63$ ). Nous n'avons pas trouvé de différence significative entre les techniques d'interaction pour chaque indice de difficulté ( $p > 0,08$ ). Le taux d'erreur de toutes nos techniques d'interaction est relativement élevé par rapport à d'autres études. Nous pensons que cela peut être dû au mécanisme de sélection (le fait d'appuyer sur la barre d'espace peut être gênant) et à la précision du système de localisation (moins précis qu'un laser de souris).

### 1.2.3. DEBIT ET MODELISATION

Nous complétons l'analyse des performances en considérant le débit de chaque périphérique d'entrée (Figure 99 - Bas). Le débit obtenu pour la souris ( $\mu = 4.15$  bits / s,  $\sigma = 0.74$  bit / s) est similaire à celui obtenu dans la littérature [ref 10 RPM pointage 2D]. Un test de Friedman révèle qu'il n'y a pas de différence significative entre les techniques d'interaction en termes de débit. Nous avons effectué une analyse via la loi de Fitts pour chaque technique d'entrée (Figure 99 - Bas). La qualité prédictive du modèle est très élevée: les valeurs  $R^2$  sont supérieures à 0.95 pour chacune des techniques d'interaction.

### 1.2.4. PREFERENCE UTILISATEUR ET FATIGUE

Les résultats sur le score SUS pour la RPM hémisphérique, la souris et la RPM convexe sont respectivement de 78.5 (SD = 6.3), 80.33 (SD = 3.8) et 85 (SD = 7.5). L'ordre de ces scores est cohérent avec les préférences des utilisateurs: les participants ont placé la RPM Convexe 11 fois en première position, la RPM hémisphérique 9 fois en seconde et la souris 9 fois en dernière position.

## 1.3. ÉTUDE 3 : MANIPULATION 3D

### 1.3.1. POINTAGE 3D

Concernant le temps de réalisation Figure 100 - Gauche), un test de Friedman a révélé une différence significative entre les techniques d'interaction ( $\chi^2 (3) = 25.2$   $p < 0.001$ ). Les deux versions de la RPM (2.28 s) sont plus rapides que les deux versions de la SpaceMouse (3.68 s) ( $Z = -13.34$ ,  $p < 0.001$ ). Ce résultat est vrai pour toutes les valeurs des indices de difficulté et la différence augmente avec la difficulté (29% pour les indices les plus simples contre 33% pour les plus difficiles). Il n'y a pas de différence significative entre les techniques d'interaction concernant le taux d'erreur, qui était en moyenne de 5% ( $\chi^2 (3) = 0.28$   $p = 0.96$ ).

### 1.3.2. ROTATION 3D

Nous n'avons trouvé aucune différence significative entre les deux versions de la RPM et les deux versions de la SpaceMouse concernant le temps de réalisation ( $\chi^2 (3) = 2.3$   $p = 0.51$ ).

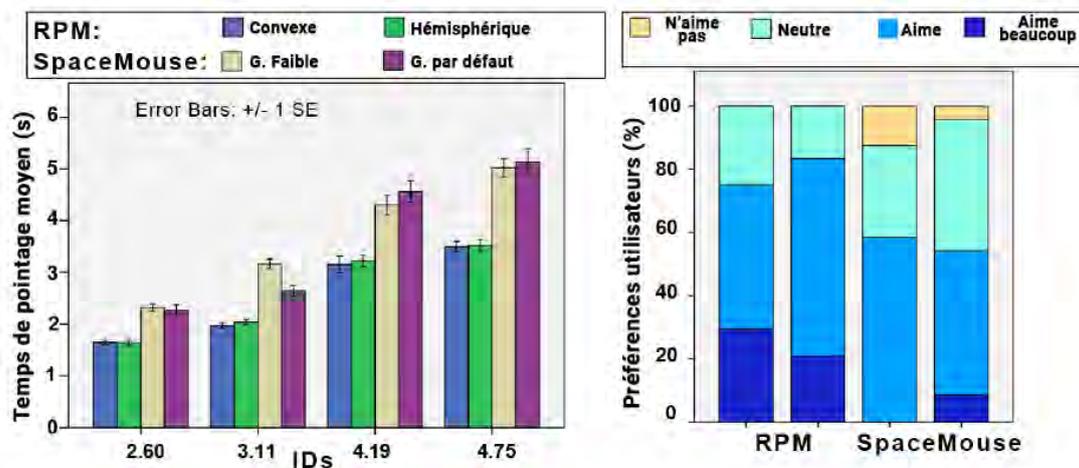


Figure 100. Temps de réalisation de la tâche de pointage 3D (gauche) pour chaque technique d'interaction et indice de difficulté. Préférence utilisateur pour chaque technique d'interaction (droite).

### 1.3.3. PREFERENCE UTILISATEUR ET UTILISABILITE

Il n'y avait pas de différence significative en termes de préférence utilisateur ou d'utilisabilité entre les deux tâches. Par conséquent, nous présentons les résultats de ces deux tâches fusionnés. En termes de préférence (Figure 100 - Droite), 75% des utilisateurs ont évalué positivement la RPM convexe (avec un score de 4 ou plus); 80% hémisphérique; 54% la SpaceMouse avec le gain par défaut et 58% avec le gain plus faible. En terme d'utilisabilité, 75% des participants ont évalué la RPM convexe

positivement (avec un score de 4 ou plus); 87% la RPM hémisphérique; 45% la SpaceMouse avec gain par défaut et 54% avec gain plus faible. Les utilisateurs ont notamment trouvé que "la manipulation de la SpaceMouse nécessite plus de concentration que pour la RPM "(p4, p9) et que "RPM est plus agréable parce que je peux effectuer le mouvement en contrôlant la position du curseur, comme avec une souris classique"(p6, p8, p12).

#### 1.3.4. FATIGUE

Il n'y avait pas de différence significative en termes de fatigue entre les deux tâches. Par conséquent, nous présentons les résultats des deux tâches fusionnés. En termes de fatigue, il n'y avait pas de différence significative entre les techniques d'interaction ( $\chi^2 (3) = 3.50$ ,  $p = 0.32$ ), avec une valeur moyenne de 11 (échelle de Borg 6-20).

## 2. RESULTATS DETAILLES : LE STACKING

Alors que les données expérimentales en IHM sont traditionnellement analysées en appliquant le test de signification d'hypothèse nulle (NHST), cette forme d'analyse des données expérimentales est de plus en plus critiquée dans les communautés statistiques [6, 38] et de l'IHM [43, 44]. Nous présentons donc nos résultats en utilisant des techniques d'estimation avec des tailles d'effet et des intervalles de confiance (0,95), au lieu des statistiques de p-value, cohérentes avec les recommandations récentes de l'APA [184].

### 2.1. ÉTUDE PRÉLIMINAIRE : EXPLORATION DE GESTES

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).

### 2.2. ÉTUDE 1 : SÉLECTION D'UNE PLAGE DE CELLULES

#### 2.2.1. TEMPS DE SELECTION

Les résultats montrent une nette différence entre les quatre techniques d'interaction. L'interaction Tactile est respectivement 19.9% IC [11.3%, 29.7%] et 16.1% IC [11.7%, 25%] plus lente que les technique Finger Stack et Intersection (Table 9). Cependant,

Tactile était 6,7%, IC [1,5%, 13,0%] plus rapide que Stack-and-Drag (Figure 101 - Droite).

	Moyenne (s)	IC.Min	IC.Max
Tactile	7.635	6.872	8.942
Intersection	6.536	6.018	7.299
Finger Stack	6.351	5.794	6.933
Stack-and-Drag	8.265	7.300	9.458

Table 8. Temps de sélection (en secondes) et intervalle de confiance par technique d'interaction.

	Moyenne	IC.Min	IC.Max
Tactile / Intersection	1.161	1.117	1.25
Tactile / Finger Stack	1.199	1.133	1.297
Tactile / Stack-and-Drag	0.933	0.870	0.985

Table 9. Ratio du temps de sélection et intervalle de confiance par technique d'interaction.

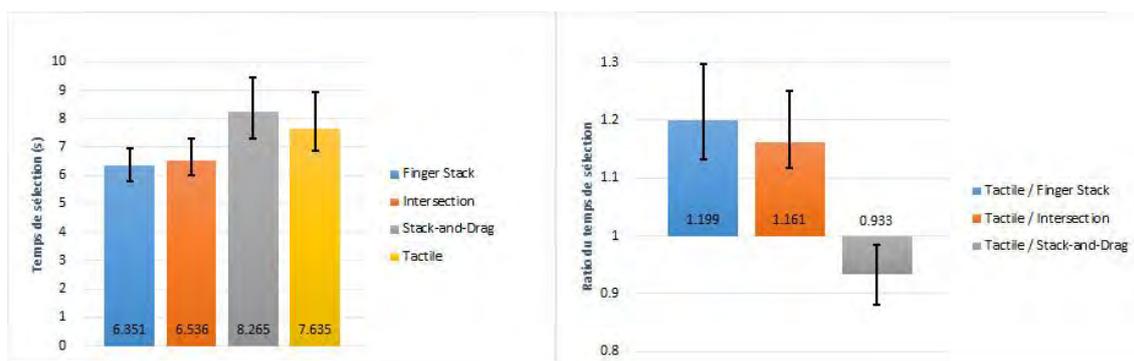


Figure 101. Temps de sélection en s (Gauche) et ratio (Droite) par rapport à Tactile (95% IC).

Une analyse par type de plage (Figure 102) révèle que cette amélioration des performances existe pour chacune des plages testées, avec la différence la plus importante dans les colonnes complètes (Intersection et Finger Stack requièrent respectivement 45,6% et 40,9% moins de temps que Tactile) et lignes complètes (Intersection et Finger Stack requièrent respectivement 62,7% et 36,9% moins de temps que Tactile).

Une analyse par type de défilement (avec ou sans) révèle que lorsqu'un défilement est requis, le temps d'exécution d'une sélection augmente du même pourcentage (IC de 64,1% [58,8, 69,5]) pour chaque technique. Ainsi, notre mécanisme de défilement n'induit pas une augmentation de temps plus importante que celle du Tactile.

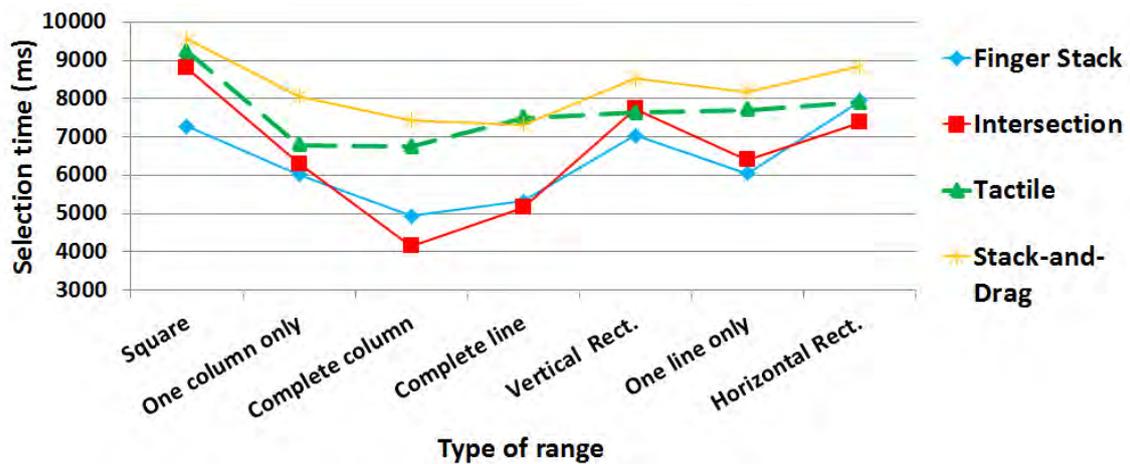


Figure 102. Temps de sélection (en ms) par type de plage et par technique.

### 2.2.2. UTILISABILITE ET PREFERENCE UTILISATEUR

L'analyse des questionnaires SUS (Table 10, Figure 103) établit clairement un meilleur score d'utilisabilité pour les techniques Tactile et Intersection, respectivement de 85.8, IC [78.3, 91.3] et de 82.1, IC [73.5, 87.3], que pour Finger Stack (SUS = 68.3, IC [58.1, 76,9]) et Stack-and-Drag (SUS = 52,7, IC [36,7, 63,1]). L'appréciation générale des techniques, confirme ces résultats avec 11 participants sur 12 qui ont aimé (5 et plus) Tactile, 10 qui ont aimé Intersection, 8 Finger Stack et 5 Stack-and-Drag.

	Moyenne	IC.Min	IC.Max
<b>Tactile</b>	85.8	78.33	91.25
<b>Intersection</b>	82.1	73.54	87.29
<b>Finger Stack</b>	68.3	58.13	76.88
<b>Stack-and-Drag</b>	52.7	36.74	63.13

Table 10. Score SUS par technique d'interaction.

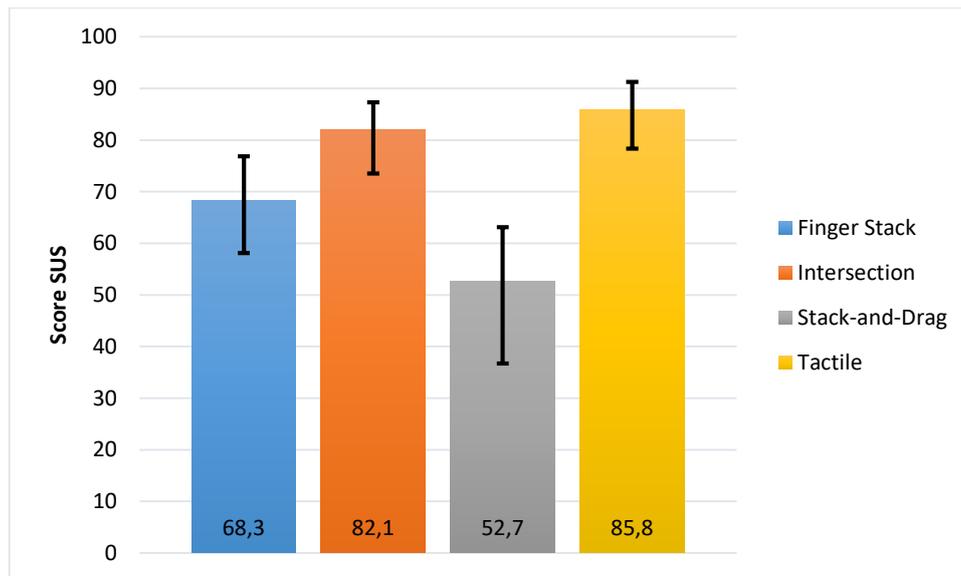


Figure 103. Score SUS par technique d'interaction (95% IC).

Les commentaires informels rapportés par les participants concordent avec cette analyse. En effet, l'interaction Tactile a été décrite comme : « Intuitive, cohérente avec les gestes tactiles classiques » (8/12). Cependant, les utilisateurs ont trouvé que « la sélection avec le doigt [est] complexe » (7/12), et qu'il est « difficile de viser la bonne cellule » (10/12). La technique Intersection a été qualifiée de « Simple à utiliser car les dimensions sont clairement séparées, ce qui réduit la charge cognitive » (9/12). D'un autre côté, certains participants ont trouvé « Difficile dans certains cas de voir les en-têtes de colonne, le smartphone les cache » (5/12). La technique Finger Stack a été décrite comme « Pratique car la sélection est presque immédiate si la gamme est complètement contenue à l'intérieur de l'écran » (12/12). Cependant, il a été remarqué que « l'usage des deux mains est restrictif » (3/12). La technique Stack-and-Drag est considérée : « Agréable à utiliser une fois la sélection initiale faite » (6/12) mais « Difficile à manipuler car j'ai du mal à viser la bonne cellule avec le point rouge [pointeur smartphone] » (9/12).

## 2.3. ÉTUDE 2 : TABLEAUX CROISÉS DYNAMIQUES

La tâche étant divisée en deux sous-tâches (sélection et assignation), nous rapportons les mesures quantitatives (temps, erreur) pour chaque étape séparément.

### 2.3.1. TEMPS DE SELECTION (ETAPE 1)

Les résultats confirment que la sélection d'une plage de cellules avec l'interaction Tactile est plus lente que la sélection basée sur le stacking (Table 11, Figure 104) : en moyenne, la sélection Tactile nécessitait 14,4% (IC [10,3, 22,2]) plus de temps qu'avec nos techniques d'interaction. Le temps de sélection était de 10.6s (IC [10.2, 10.9]) pour Tactile, de 9.2s (IC [8.4, 9.7]) pour Dual-Screen Touch, de 9.4s (IC [9.0, 9.8]) pour Stack-and-Touch et de 9.2 s (IC [8.5, 9.6]) pour Stack-and-Translate.

	Moyenne (s)	IC.Min	IC.Max
Tactile	10.6	10.2	10.9
Dual-Screen Touch	9.2	8.4	9.7
Stack-and-Touch	9.2	9.0	9.8
Stack-and-Translate	9.2	8.5	9.6

Table 11. Temps de sélection (en secondes) et intervalle de confiance par technique

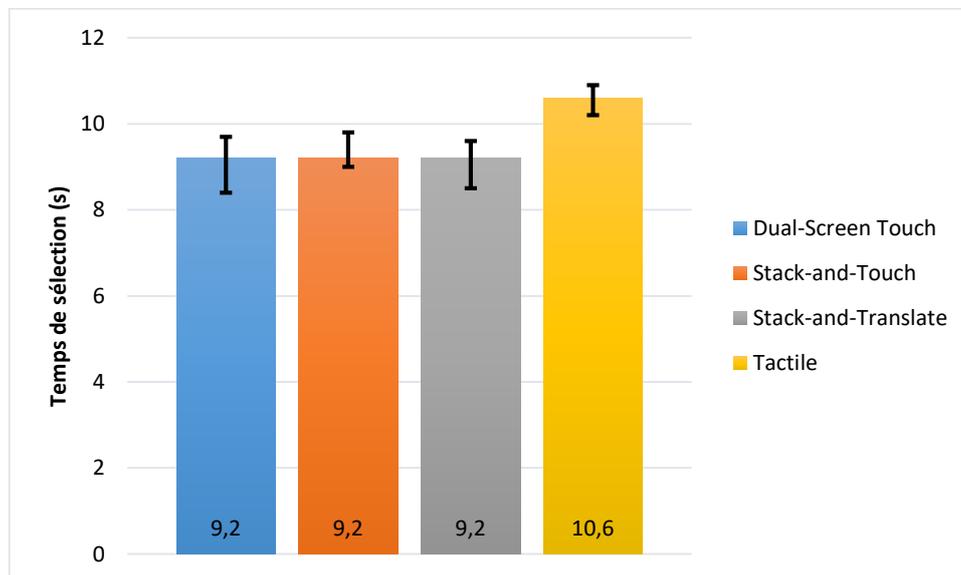


Figure 104. Temps de sélection par technique d'interaction (en ms, 95% CI).

### 2.3.2. TEMPS D’AFFECTATION (ETAPE 2 ET 3)

Les résultats montrent que l'assignation (c'est-à-dire le remplissage des zones de l'assistant du tableau croisé dynamique avec les éléments correspondants, Table 12, Figure 105) avec Stack-and-Translate nécessite moins de temps (7.3s, CI [6.9, 7.9]) qu'avec les techniques Dual-Screen Touch (8.3s, CI [7.8, 8.9]), Stack-and-Touch (8.7s, CI [8.3, 9.1]) et Tactile (9.5s, CI [9.0, 10.0]). Ces résultats restent valables pour chaque nombre de zones testées (c'est-à-dire que les performances relatives des techniques d'interaction sont équivalentes pour chaque condition de nombre de zones).

	Moyenne (s)	IC.Min	IC.Max
Tactile	9.5	9.0	10.0
Dual-Screen Touch	8.3	7.8	8.9
Stack-and-Touch	8.7	8.3	9.1
Stack-and-Translate	7.3	6.9	7.9

Table 12. Temps d'affectation (en secondes) et intervalle de confiance par technique d'interaction.

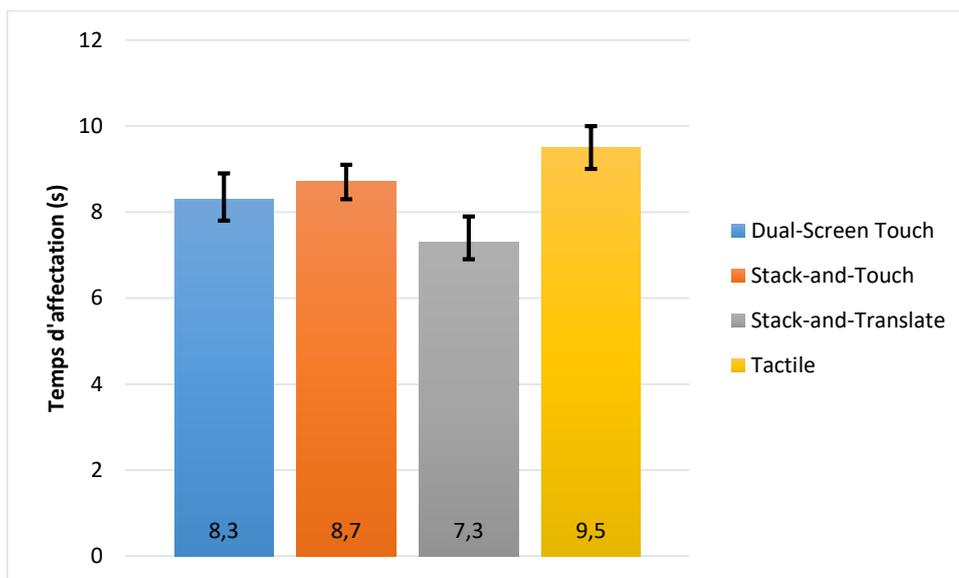


Figure 105. Temps d'affectation par technique d'interaction (en ms, 95% CI).

Lors de la modification de la fonction d'agrégation de la zone Valeur (par exemple Somme au lieu de Moyenne), le temps d'affectation de la technique Tactile augmente de 42.6% CI [32.7, 54.4] alors qu'il n'augmente que de 15.5% CI [9.3, 21.4] pour Stack-and-Translate, de 14,2% CI [8.8, 20.6] pour Stack-and-Touch et de 13.0% CI [7.6, 22.0] pour Dual-Screen Touch (Figure 106). Nous pouvons expliquer ces résultats par le fait que les boutons de configuration sont plus difficiles à cibler en utilisant Tactile qu'en utilisant les techniques basées sur le stacking.

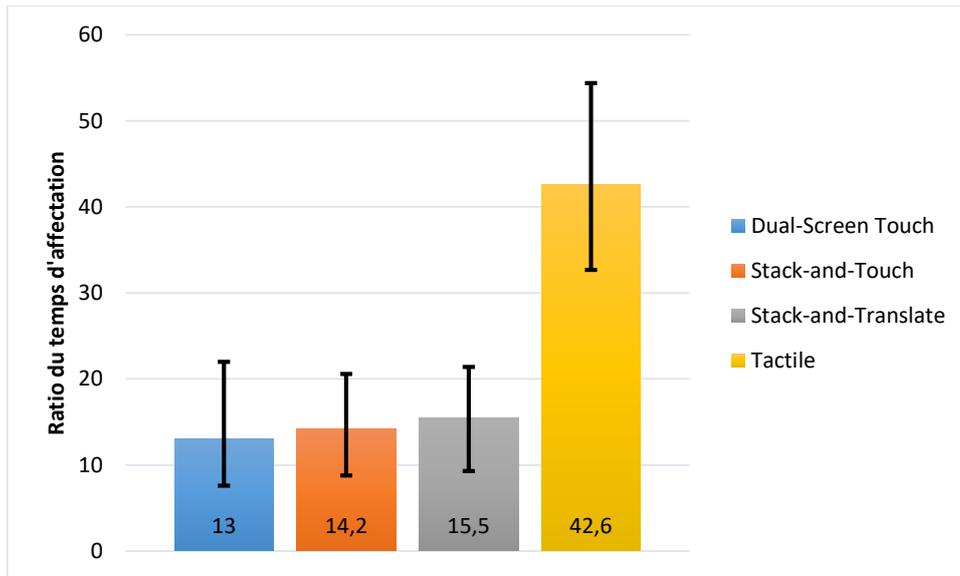


Figure 106. Ratio de l'augmentation du temps d'affectation (en pourcentage, 95% CI, i.e. modification nécessaire sur aucune modification).

### 2.3.3. ERREURS

Nous n'avons mesuré aucune erreur de sélection. En ce qui concerne les erreurs d'affectation (Figure 107), il semble que Stack-and-Translate (8,3% des essais contenaient au moins une erreur CI [5.6, 11.5]) et Stack-and-Touch (9,4% des essais CI [6.6, 12.5]) ont produit moins d'erreurs que Tactile (14,9% des essais CI [11,1, 18,1]) et Dual-Screen Touch (13,5% des essais CI [10,4, 19,1]).

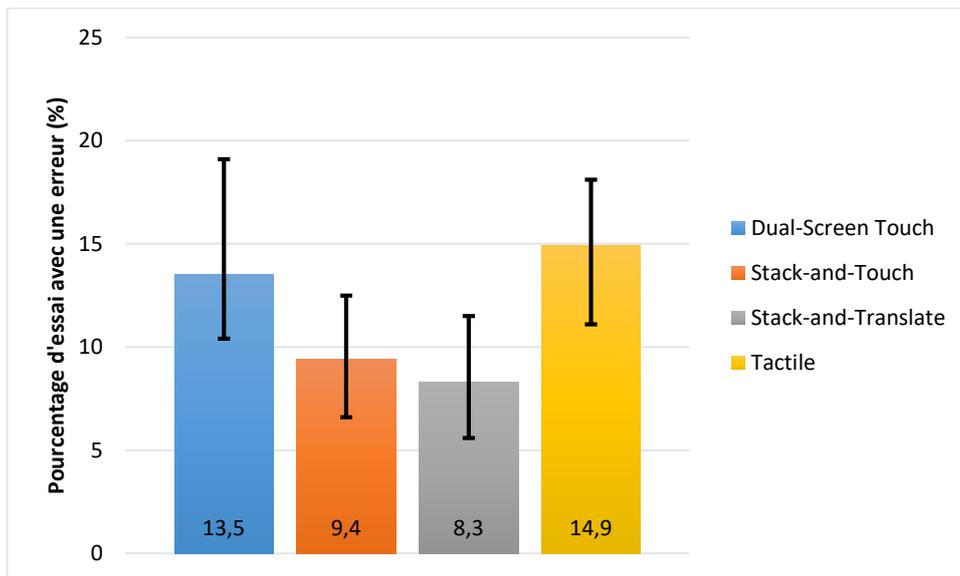


Figure 107. Pourcentage des essais dans lesquels au moins une erreur d'affectation est survenue (95% CI).

### 2.3.4. UTILISABILITE ET PREFERENCE UTILISATEUR

Les questionnaires SUS (Table 13, Figure 108) établissent un meilleur score d'utilisabilité pour Stack-and-Translate (90.4, IC [76.9, 95.4]) et Dual-Screen Touch (80.8, IC [73.3, 87.9]), que pour Tactile (70.2, CI [63.3, 75.0]) et Stack-and-Touch (73.1, CI [68.5, 76.0]). L'appréciation générale des techniques, recueillies à l'aide d'une échelle de Likert à 7 points, confirme ces résultats avec 11 participants sur 12 qui ont aimé (5 et plus) Stack-and-Translate, 10 qui ont aimé Dual-Screen touch, 9 Stack-and-Touch et 7 Tactile.

	Moyenne (s)	IC.Min	IC.Max
Tactile	70.2	63.3	75.0
Dual-Screen Touch	80.8	73.3	87.9
Stack-and-Touch	73.1	68.5	76.0
Stack-and-Translate	90.4	76.9	95.4

Table 13. Score SUS par technique d'interaction.

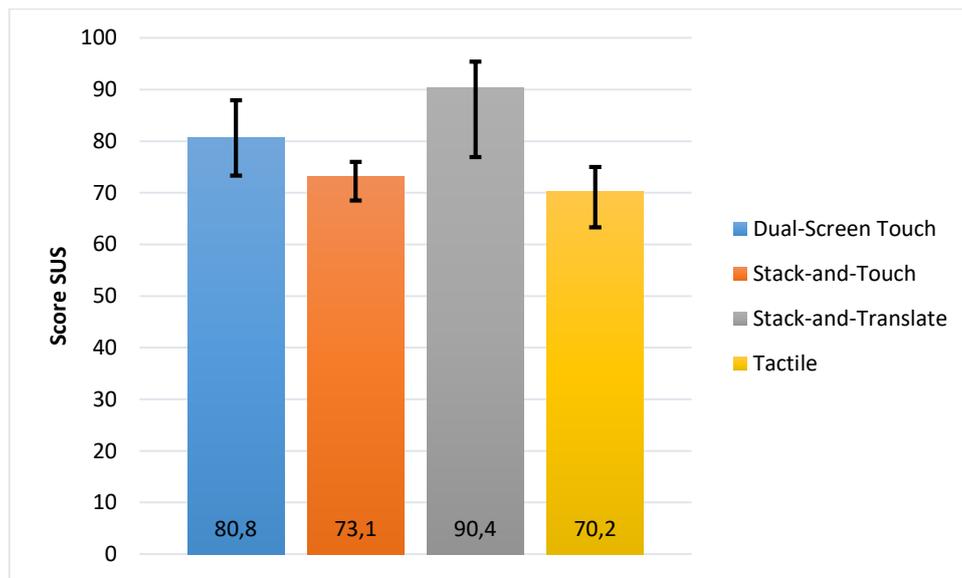


Figure 108. Score SUS par technique d'interaction (95% CI).

Les commentaires informels rapportés par les participants concordent avec cette analyse. Stack-and-Translate a été décrit comme « très simple et très rapide » (9/12). Cependant, deux utilisateurs ont trouvé « fastidieux de changer la posture de la main quand je veux changer la fonction d'agrégation » (2/12). La technique Dual-Screen Touch a été qualifiée de « simple et facile à apprendre » (11/12). D'un autre côté, la plupart des participants ont déclaré que « cliquer sur l'en-tête de la colonne n'est pas pratique; Je voudrais cliquer sur l'une des cellules de la colonne » (9/12). La technique Stack-and-Touch a été décrite comme « plus simple car le menu sur le smartphone est plus clair que sur la tablette »

(7/12). Cependant, certains ont souligné que « le besoin de stack plusieurs fois est fatigant » (4/12). La technique Tactile a été considérée comme « facile à utiliser car elle est la même que sur la version bureau classique mais tactile » (6/12) mais « vraiment dérangeante : je ne peux pas contrôler dans quelle cellule l'en-tête ira si je coche la case et je peux aussi le faire glisser mais je n'arrive pas souvent à le placer correctement » (8/12).



# BIBLIOGRAPHIE

---

1. 1€ filter : <http://cristal.univ-lille.fr/~casiez/1euro/> (dernière visite le 05/07/2018)
2. Brian Albright. 2009. *Mathematical Modeling with Excel* (1st ed.). Jones and Bartlett Publishers, Inc., , USA.
3. James F. Allen. Maintaining knowledge about temporal intervals. *Commun. ACM*, 26(11):832–843, 1983.
4. Daniel Avrahami, Jacob O. Wobbrock, and Shahram Izadi. 2011. Portico: tangible interaction on and around a tablet. In *Proceedings of the 24th annual ACM symposium on User interface software and technology (UIST '11)*. ACM, New York, NY, USA, 347-356. DOI: <https://doi.org/10.1145/2047196.2047241>
5. Mathias Baglioni, Eric Lecolinet, and Yves Guiard. 2011. JerkTilts: using accelerometers for eight-choice selection on mobile devices. In *Proceedings of the 13th international conference on multimodal interfaces (ICMI '11)*. ACM, New York, NY, USA, 121-128. DOI=<http://dx.doi.org/10.1145/2070481.2070503>
6. 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>
7. Ravin Balakrishnan, Thomas Baudel, Gordon Kurtenbach, and George Fitzmaurice. 1997. The Rockin'Mouse: integral 3D manipulation on a plane. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems (CHI '97)*. ACM, New York, NY, USA, 311-318. DOI=<http://dx.doi.org/10.1145/258549.258778>
8. Ravin Balakrishnan and Pranay Patel. 1998. The PadMouse: facilitating selection and spatial positioning for the non-dominant hand. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '98)*, Clare-Marie Karat, Arnold Lund, Joëlle Coutaz, and John Karat (Eds.). ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 9-16. DOI=<http://dx.doi.org/10.1145/274644.274646>
9. Nikola Banovic, Frank Chun Yat Li, David Dearman, Koji Yatani, and Khai N. Truong. 2011. Design of unimanual multi-finger pie menu interaction. In *Proceedings of the ACM International Conference on Interactive Tabletops and*

- Surfaces (ITS '11). ACM, New York, NY, USA, 120-129. DOI: <https://doi.org/10.1145/2076354.2076378>
10. Jakob E. Bardram, Christina Fuglsang, and Simon C. Pedersen. 2010. CompUTE: a runtime infrastructure for device composition. In *Procs of AVI '10*. ACM, New York, NY, USA, 111-118.
  11. Dominikus Baur, Bongshin Lee, and Sheelagh Carpendale. 2012. TouchWave: kinetic multi-touch manipulation for hierarchical stacked graphs. In *Proceedings of the 2012 ACM international conference on Interactive tabletops and surfaces (ITS '12)*. ACM, New York, NY, USA, 255-264. DOI: <https://doi.org/10.1145/2396636.2396675>
  12. Michel Beaudouin-Lafon. 2004. Designing interaction, not interfaces. In *Procs of AVI '04*. ACM, New York, NY, USA, 15-22.
  13. Benjamin B. Bederson. 2004. Interfaces for staying in the flow. *Ubiquity 2004*, September (September 2004), 1-1.
  14. Hrvoje Benko, Shahram Izadi, Andrew D. Wilson, Xiang Cao, Dan Rosenfeld, and Ken Hinckley. 2010. Design and evaluation of interaction models for multi-touch mice. In *Proceedings of Graphics Interface 2010 (GI '10)*. Canadian Information Processing Society, Toronto, Ont., Canada, Canada, 253-260
  15. 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: <https://doi.org/10.1145/2788940.2788941>
  16. Mathieu Berthelley, Elodie Cayez, Marwan Ajem, Gilles Bailly, Sylvain Malacria, and Eric Lecolinet. 2015. SpotPad, LociPad, ChordPad and InOutPad: investigating gesture-based input on touchpad. In *Proceedings of the 27th Conference on l'Interaction Homme-Machine (IHM '15)*. ACM, New York, NY, USA, Article 4, 8 pages. DOI: <https://doi.org/10.1145/2820619.2820623>
  17. Bi, X., T. Moscovich, G. Ramos, R. Balakrishnan, and K. Hinckley. An exploration of pen rolling for pen-based interaction. In *Proc. UIST'08*. ACM Press. p. 191-200
  18. Borg, G. Borg's Perceived Exertion and Pain Scales. *Human Kinetics* (1998), viii 104pp.

19. Christophe Bortolaso, Emmanuel Dubois. 2013. Model Assisted Creativity Sessions for the Design of Mixed Interactive Systems: a Protocol Analysis. In INTERACT'13, Springer, LNCS 8119, p. 126-143, 2013.
20. Jeffrey Browne, Bongshin Lee, Sheelagh Carpendale, Nathalie Riche, and Timothy Sherwood. 2011. Data analysis on interactive whiteboards through sketch-based interaction. In Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '11). ACM, New York, NY, USA, 154-157. DOI: <https://doi.org/10.1145/2076354.2076383>
21. Alex Butler, Shahram Izadi, and Steve Hodges. 2008. SideSight: multi-"touch" interaction around small devices. In Proceedings of the 21st annual ACM symposium on User interface software and technology (UIST '08). ACM, New York, NY, USA, 201-204. DOI: <https://doi.org/10.1145/1449715.1449746>
22. Buxton, W. (1986). There's more to interaction than meets the eye: Some issues in manual input. In Norman, D. A. and Draper, S. W. (Eds.), User Centered System Design." New Perspectives on Human-Computer Interaction, 319-337. Lawrence Erlbaum: New Jersey.
23. W. Buxton. Multi-touch systems that i have known and loved. <http://billbuxton.com/multitouchOverview.html>.
24. W. Buxton and B. Myers. 1986. A study in two-handed input. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems(CHI '86), Marilyn Mantei and Peter Orbeton (Eds.). ACM, New York, NY, USA, 321-326. DOI=<http://dx.doi.org/10.1145/22627.22390>
25. Buxton, W. Lexical and pragmatic considerations of input structures. Comput. Graph., 17(1) :31-37, 1983.
26. Stuart K. Card , Jock D. Mackinlay , George G. Robertson, A morphological analysis of the design space of input devices, ACM Transactions on Information Systems (TOIS), v.9 n.2, p.99-122, April 1991 [doi>10.1145/123078.128726]
27. Géry Casiez and Nicolas Roussel. 2011. No more bricolage!: methods and tools to characterize, replicate and compare pointing transfer functions. In Proceedings of the 24th annual ACM symposium on User interface software and technology (UIST '11). ACM, New York, NY, USA, 603-614. DOI: <https://doi.org/10.1145/2047196.2047276>
28. Géry Casiez, Daniel Vogel, Qing Pan, and Christophe Chaillou. 2007. RubberEdge: reducing clutching by combining position and rate control with

- elastic feedback. In Proceedings of the 20th annual ACM symposium on User interface software and technology (UIST '07). ACM, New York, NY, USA, 129-138. DOI: <https://doi.org/10.1145/1294211.1294234>
29. 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>
  30. Nicholas Chen, François Guimbretière, and Abigail Sellen. 2013. Graduate student use of a multi-slate reading system. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13). ACM, New York, NY, USA, 1799-1808. DOI: <https://doi.org/10.1145/2470654.2466237>
  31. Xiang 'Anthony' Chen, Tovi Grossman, Daniel J. Wigdor, and George Fitzmaurice. 2014. Duet: exploring joint interactions on a smart phone and a smart watch. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14). ACM, New York, NY, USA, 159-168. DOI: <https://doi.org/10.1145/2556288.2556955>
  32. Classement Truffle100 : <https://www.truffle100.fr/2017.html> (dernière visite le 05/07/2018)
  33. ContextToolkit <http://contexttoolkit.sourceforge.net/>
  34. Christian Corsten, Simon Voelker, Andreas Link, and Jan Borchers. 2018. Use the Force Picker, Luke: Space-Efficient Value Input on Force-Sensitive Mobile Touchscreens. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18). ACM, New York, NY, USA, Paper 661, 12 pages. DOI: <https://doi.org/10.1145/3173574.3174235>
  35. Joëlle Coutaz and Laurence Nigay. Les propriétés CARE dans les interfaces multimodales. In Actes de la conférence IHM'94, Lille, pages 7–14, 1994.
  36. Coutrix, C., Nigay, L.: Mixed reality: a model of mixed interaction. In: Proc. of the Working Conf. on Advanced visual interfaces, AVI 2006, pp. 43–50. ACM Press (2006).
  37. Andrew Crotty, Alex Galakatos, Emanuel Zraggen, Carsten Binnig, and Tim Kraska. 2015. Vizdom: interactive analytics through pen and touch. Proc. VLDB Endow. 8, 12 (August 2015), 2024-2027. DOI=<http://dx.doi.org/10.14778/2824032.2824127>

38. 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>
39. Florian Daiber, Lianchao Li, and Antonio Krüger. 2012. Designing gestures for mobile 3D gaming. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia (MUM '12)*. ACM, New York, NY, USA, , Article 3 , 4 pages. DOI=<http://dx.doi.org/10.1145/2406367.2406371>
40. Alexander De Luca, Emanuel von Zezschwitz, Ngo Dieu Huong Nguyen, Max-Emanuel Maurer, Elisa Rubegni, Marcello Paolo Scipioni, and Marc Langheinrich. 2013. Back-of-device authentication on smartphones. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 2389-2398. DOI: <https://doi.org/10.1145/2470654.2481330>
41. Tilman Dingler, Rufat Rzayev, Alireza Sahami Shirazi, and Niels Henze. 2018. Designing Consistent Gestures Across Device Types: Eliciting RSVP Controls for Phone, Watch, and Glasses. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Paper 419, 12 pages. DOI: <https://doi.org/10.1145/3173574.3173993>
42. Pierre Dragicevic, Jean-Daniel Fekete. Input Device Selection and Interaction Configuration with ICON. A Blandford and J Vanderdonckt and P Gray. *Proceedings of the International Conference IHM-HCI 2001*, Sep 2001, Lille, France. Springer Verlag, pp.543-448, 2001.
43. 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)
44. 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>
45. Steven M. Drucker, Danyel Fisher, Ramik Sadana, Jessica Herron, and m.c. schraefel. 2013. TouchViz: a case study comparing two interfaces for data analytics on tablets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 2301-2310. DOI: <https://doi.org/10.1145/2470654.2481318>

46. Emmanuel Dubois, Marcos Serrano, and Mathieu Raynal. 2018. Rolling-Menu: Rapid Command Selection in Toolbars Using Roll Gestures with a Multi-DoF Mouse. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18). ACM, New York, NY, USA, Paper 367, 12 pages. DOI: <https://doi.org/10.1145/3173574.3173941>
47. Ulrich Eck and Christian Sandor. 2013. HARP: A framework for visuo-haptic augmented reality. In IEEE Virtual Reality
48. Excel sur Android: <https://play.google.com/store/apps/details?id=com.microsoft.office.excel&hl=fr> (dernière visite le 05/07/2018)
49. Ribel Fares, Shaomin Fang, and Oleg Komogortsev. 2013. Can we beat the mouse with MAGIC?. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13). ACM, New York, NY, USA, 1387-1390. DOI: <https://doi.org/10.1145/2470654.2466183>
50. Le FlipKeyboard. <http://research.microsoft.com/enus/um/people/bibuxton/buxtoncollection/detail.aspx?id=76>
51. Euan Freeman, Stephen Brewster, and Vuokko Lantz. 2014. Towards usable and acceptable above-device interactions. In Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services (MobileHCI '14). ACM, New York, NY, USA, 459-464. DOI: <https://doi.org/10.1145/2628363.2634215>
52. Mathias Frisch, Jens Heydekorn, and Raimund Dachzelt. 2009. Investigating multi-touch and pen gestures for diagram editing on interactive surfaces. In Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '09). ACM, New York, NY, USA, 149-156. DOI: <https://doi.org/10.1145/1731903.1731933>
53. Bernd Froehlich, Jan Hochstrate, Verena Skuk, and Anke Huckauf. 2006. The GlobeFish and the GlobeMouse: two new six degree of freedom input devices for graphics applications. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06), Rebecca Grinter, Thomas Rodden, Paul Aoki, Ed Cutrell, Robin Jeffries, and Gary Olson (Eds.). ACM, New York, NY, USA, 191-199. DOI=<http://dx.doi.org/10.1145/1124772.1124802>

54. Bernd Fröhlich and John Plate. 2000. The cubic mouse: a new device for three-dimensional input. In Procs of CHI '00. ACM, New York, NY, USA, 526-531.
55. Bruno Fruchard, Eric Lecolinet, Olivier Chapuis. Démonstration de MarkPad : Augmentation du pavé tactile pour la sélection de commandes. AFIHM. 29ème conférence francophone sur l'Interaction Homme-Machine, Aug 2017, Poitiers, France. IHM-2017, 2 p., 2017.
56. Fernando Garcia-Sanjuan, Javier Jaen, Geraldine Fitzpatrick, and Alejandro Catala. 2016. MarkAirs: Around-Device Interactions with Tablets Using Fiducial Markers -- An Evaluation of Precision Tasks. In Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '16). ACM, New York, NY, USA, 2474-2481. DOI: <https://doi.org/10.1145/2851581.2892486>
57. Mark Gardener. 2015. Managing Data Using Excel. Pelagic Publishing, , United Kingdom.
58. GIMP : <https://www.gimp.org/fr/> (dernière visite le 05/07/2018)
59. David Glessner, François Bérard, and Jeremy R. Cooperstock. 2013. Overcoming limitations of the trackpad for 3d docking operations. In CHI '13 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13). ACM, New York, NY, USA, 1239-1244. DOI: <https://doi.org/10.1145/2468356.2468578>
60. Alix Goguy, Géry Casiez, Thomas Pietrzak, Daniel Vogel, and Nicolas Roussel. 2014. Adoiraccourcix: multi-touch command selection using finger identification. In Proceedings of the 26th Conference on l'Interaction Homme-Machine (IHM '14). ACM, New York, NY, USA, 28-37. DOI: <https://doi.org/10.1145/2670444.2670446>
61. Rafael Morales González, Caroline Appert, Gilles Bailly, and Emmanuel Pietriga. 2017. Passive yet Expressive TouchTokens. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17). ACM, New York, NY, USA, 3741-3745. DOI: <https://doi.org/10.1145/3025453.3025894>
62. Google Street View : <https://www.google.com/maps> (dernière visite le 05/07/2018)
63. Google Spreadsheet sur Android: <https://play.google.com/store/apps/details?id=com.google.android.apps.docs.editors.sheets&hl=fr> (dernière visite le 05/07/2018)

64. Jens Grubert, Eyal Ofek, Michel Pahud, Matthias Kranz, and Dieter Schmalstieg. 2016. GlassHands: Interaction Around Unmodified Mobile Devices Using Sunglasses. In Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces (ISS '16). ACM, New York, NY, USA, 215-224. DOI: <https://doi.org/10.1145/2992154.2992162>
65. M. Hachet , P. Guitton, The interaction table: a new input device designed for interaction in immersive large display environments, Proceedings of the workshop on Virtual environments 2002, May 30-31, 2002, Barcelona, Spain
66. Martin Hachet , Pascal Guitton , Patrick Reuter, The CAT for efficient 2D and 3D interaction as an alternative to mouse adaptations, Proceedings of the ACM symposium on Virtual reality software and technology, October 01-03, 2003, Osaka, Japan
67. Martin Hachet, Pascal Guitton, and Patrick Reuter. 2003. The CAT for efficient 2D and 3D interaction as an alternative to mouse adaptations. In Proceedings of the ACM symposium on Virtual reality software and technology (VRST '03). ACM, New York, NY, USA, 225-112. DOI=<http://dx.doi.org/10.1145/1008653.1008689>
68. Peter Hamilton and Daniel J. Wigdor. 2014. Conductor: enabling and understanding cross-device interaction. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14). ACM, New York, NY, USA, 2773-2782. DOI: <https://doi.org/10.1145/2556288.2557170>
69. Chris Harrison and Scott E. Hudson. 2009. Abracadabra: wireless, high-precision, and unpowered finger input for very small mobile devices. In Proceedings of the 22nd annual ACM symposium on User interface software and technology (UIST '09). ACM, New York, NY, USA, 121-124. DOI: <https://doi.org/10.1145/1622176.1622199>
70. Beverly L. Harrison, Kenneth P. Fishkin, Anuj Gujar, Carlos Mochon, and Roy Want. 1998. Squeeze me, hold me, tilt me! An exploration of manipulative user interfaces. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '98), Clare-Marie Karat, Arnold Lund, Joëlle Coutaz, and John Karat (Eds.). ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 17-24. DOI=<http://dx.doi.org/10.1145/274644.274647>

71. Björn Hartmann, Scott Doorley, and Scott R. Klemmer. 2008. Hacking, Mashing, Gluing: Understanding Opportunistic Design. *IEEE Pervasive Computing* 7, 3 (July 2008), 46-54. DOI=<http://dx.doi.org/10.1109/MPRV.2008.54>
72. H. Rex Hartson, Antonio C. Siochi, and D. Hix. 1990. The UAN: a user-oriented representation for direct manipulation interface designs. *ACM Trans. Inf. Syst.* 8, 3 (July 1990), 181-203.
73. 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>
74. Khalad Hasan, Junhyeok Kim, David Ahlström, and Pourang Irani. 2016. Thumbs-Up: 3D Spatial Thumb-Reachable Space for One-Handed Thumb Interaction on Smartphones. In *Proceedings of the 2016 Symposium on Spatial User Interaction (SUI '16)*. ACM, New York, NY, USA, 103-106. DOI: <https://doi.org/10.1145/2983310.2985755>
75. Khalad Hasan, Xing-Dong Yang, Hai-Ning Liang, and Pourang Irani. 2012. How to position the cursor?: an exploration of absolute and relative cursor positioning for back-of-device input. In *Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services (MobileHCI '12)*. ACM, New York, NY, USA, 103-112. DOI: <https://doi.org/10.1145/2371574.2371591>
76. Seongkook Heo, Jiseong Gu, and Geehyuk Lee. 2014. Expanding touch input vocabulary by using consecutive distant taps. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 2597-2606. DOI: <https://doi.org/10.1145/2556288.2557234>
77. Seongkook Heo, Jingun Jung, and Geehyuk Lee. 2016. MelodicTap: fingering hotkey for touch tablets. In *Proceedings of the 28th Australian Conference on Computer-Human Interaction (OzCHI '16)*. ACM, New York, NY, USA, 396-400. DOI: <https://doi.org/10.1145/3010915.3010993>
78. Ken Hinckley, Mike Sinclair, Erik Hanson, Richard Szeliski, and Matt Conway. 1999. The VideoMouse: a camera-based multi-degree-of-freedom input device. In *Proceedings of the 12th annual ACM symposium on User interface software*

- and technology (UIST '99). ACM, New York, NY, USA, 103-112. DOI=<http://dx.doi.org/10.1145/320719.322591>
79. Ken Hinckley, Koji Yatani, Michel Pahud, Nicole Coddington, Jenny Rodenhouse, Andy Wilson, Hrvoje Benko, and Bill Buxton. 2010. Pen + touch = new tools. In Proceedings of the 23rd annual ACM symposium on User interface software and technology (UIST '10). ACM, New York, NY, USA, 27-36. DOI: <https://doi.org/10.1145/1866029.1866036>
  80. Ken Hinckley, Morgan Dixon, Raman Sarin, Francois Guimbretiere, and Ravin Balakrishnan. 2009. Codex: a dual screen tablet computer. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09). ACM, New York, NY, USA, 1933-1942. DOI: <https://doi.org/10.1145/1518701.1518996>
  81. John N. Howell, Robert L. Williams, Robert R. Conatser, Janet M. Burns, and David C. Eland. 2005. The Virtual Haptic Back (VHB): a virtual reality simulation of the human back for palpatory diagnostic training. SAE Technical Paper, No. 2005-01--2679
  82. Da-Yuan Huang, Ming-Chang Tsai, Ying-Chao Tung, Min-Lun Tsai, Yen-Ting Yeh, Liwei Chan, Yi-Ping Hung, and Mike Y. Chen. 2014. TouchSense: expanding touchscreen input vocabulary using different areas of users' finger pads. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14). ACM, New York, NY, USA, 189-192. DOI: <https://doi.org/10.1145/2556288.2557258>
  83. Dandan Huang, Melanie Tory, Bon Adriel Aseniero, Lyn Bartram, Scott Bateman, Sheelagh Carpendale, Anthony Tang, and Robert Woodbury. 2015. Personal visualization and personal visual analytics. IEEE Transactions on Visualization and Computer Graphics (TVCG) 21, 3: 420-433.
  84. Sungjae Hwang, Myungwook Ahn, and Kwang-yun Wohn. 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: <https://doi.org/10.1145/2501988.2501991>
  85. Ink222 T&Mouse  
<http://research.microsoft.com/enus/um/people/bibuxton/buxtoncollection/detail.aspx?id=118>

86. Robert J. K. Jacob , Linda E. Sibert, The perceptual structure of multidimensional input device selection, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, p.211-218, May 03-07, 1992, Monterey, California, USA
87. Robert J. K. Jacob , Linda E. Sibert , Daniel C. McFarlane , M. Preston Mullen, Jr., Integrality and separability of input devices, ACM Transactions on Computer-Human Interaction (TOCHI), v.1 n.1, p.3-26, March 1994 [doi>10.1145/174630.174631]
88. Bill Jelen and Michael Alexander. 2010. Pivot Table Data Crunching: Microsoft Excel 2010 (1st ed.). Que Publishing Company.
89. Seungzoo Jeong, Naoki Hashimoto, and Sato Makoto. 2004. A novel interaction system with force feedback between real-and virtual human: an entertainment system: virtual catch ball. In Proceedings of the 2004 ACM SIGCHI International Conference on Advances in computer entertainment technology, 61--66
90. Jaemin Jo, Sehi L'Yi, Bongshin Lee, and Jinwook Seo. 2017. TouchPivot: Blending WIMP & Post-WIMP Interfaces for Data Exploration on Tablet Devices. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17). ACM, New York, NY, USA, 2660-2671. DOI: <https://doi.org/10.1145/3025453.3025752>
91. Brett Jones, Rajinder Sodhi, David Forsyth, Brian Bailey, and Giuliano Maciocchi. 2012. Around device interaction for multiscale navigation. In Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services (MobileHCI '12). ACM, New York, NY, USA, 83-92. DOI: <https://doi.org/10.1145/2371574.2371589>
92. Jourde, F., Laurillau, Y., Morán, A.L., Nigay, L.: Towards specifying multimodal collaborative user interfaces: A comparison of collaboration notations. In: Interactive Systems. Design, Specification, and Verification, 15th Int. Workshop, DSV-IS, Kingston, Canada, pp. 281–286 (2008)
93. Hamed Ketabdar, Kamer Ali Yüksel, and Mehran Roshandel. 2010. MagiTact: interaction with mobile devices based on compass (magnetic) sensor. In Proceedings of the 15th international conference on Intelligent user interfaces (IUI '10). ACM, New York, NY, USA, 413-414. DOI: <https://doi.org/10.1145/1719970.1720048>
94. Seoktae Kim, Hyunjung Kim, Boram Lee, Tek-Jin Nam, and Woohun Lee. 2008. Inflatable mouse: volume-adjustable mouse with air-pressure-sensitive input and

- haptic feedback. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08). ACM, New York, NY, USA, 211-224. DOI: <https://doi.org/10.1145/1357054.1357090>
95. Microsoft Kinect : <https://developer.microsoft.com/fr-fr/windows/kinect> (Dernière visite 28/10/2018).
  96. Lisa Kleinman, Tad Hirsch, and Matt Yurdana. 2015. Exploring Mobile Devices as Personal Public Displays. In Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '15). ACM, New York, NY, USA, 233-243. DOI: <https://doi.org/10.1145/2785830.2785833>
  97. Sven Kratz and Michael Rohs. 2009. Hoverflow: exploring around-device interaction with IR distance sensors. In Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '09). ACM, New York, NY, USA, , Article 42 , 4 pages. DOI=<http://dx.doi.org/10.1145/1613858.1613912>
  98. Sven Kratz, Michael Rohs, Dennis Guse, Jörg Müller, Gilles Bailly, and Michael Nischt. 2012. PalmSpace: continuous around-device gestures vs. multitouch for 3D rotation tasks on mobile devices. In Proceedings of the International Working Conference on Advanced Visual Interfaces (AVI '12), Genny Tortora, Stefano Levialdi, and Maurizio Tucci (Eds.). ACM, New York, NY, USA, 181-188. DOI=<http://dx.doi.org/10.1145/2254556.2254590>
  99. Gordon Kurtenbach and William Buxton. 1994. User learning and performance with marking menus. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '94), Beth Adelson, Susan Dumais, and Judith Olson (Eds.). ACM, New York, NY, USA, 258-264. DOI=<http://dx.doi.org/10.1145/191666.191759>
  100. Gordon Paul Kurtenbach. 1993. The Design and Evaluation of Marking Menus. Ph.D. Dissertation. University of Toronto, Toronto, Ont., Canada, Canada. UMI Order No. GAXNN-82896
  101. Ricardo Langner, Tom Horak, and Raimund Dachsel. 2018. Demonstrating vistiles: visual data exploration using mobile devices. In Proceedings of the 2018 International Conference on Advanced Visual Interfaces (AVI '18). ACM, New York, NY, USA, Article 69, 3 pages. DOI: <https://doi.org/10.1145/3206505.3206583>

102. Leap Motion : <https://www.leapmotion.com/> (Dernière visite le 28/10/2018)
103. Bongshin Lee, Greg Smith, Nathalie Henry Riche, Amy Karlson, and Sheelagh Cappendale. 2015. SketchInsight: natural data exploration on interactive whiteboards leveraging pen and touch interaction. In Proceedings of 2015 IEEE Pacific Visualization Symposium (PacificVis '15), 199--206. [doi>10.1109/PACIFICVIS.2015.7156378]
104. Lehman, S.L. and B.M. Calhoun, An identified model for human wrist movements. *Experimental Brain Research*, 1990. 81(1): p. 199-208.
105. Sang-won Leigh, Philipp Schoessler, Felix Heibeck, Pattie Maes, and Hiroshi Ishii. 2014. THAW: tangible interaction with see-through augmentation for smartphones on computer screens. In Proceedings of the adjunct publication of the 27th annual ACM symposium on User interface software and technology (UIST'14 Adjunct). ACM, New York, NY, USA, 55-56. DOI: <https://doi.org/10.1145/2658779.2659111>
106. Lexip 3D <http://www.amazon.fr/Lexip-3DM-PROSouris-Filaire-Noir/dp/B00IZU1NE4>
107. 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>
108. Hai-Ning Liang, Cary Williams, Myron Semegen, Wolfgang Stuerzlinger, and Pourang Irani. 2012. User-defined surface+motion gestures for 3d manipulation of objects at a distance through a mobile device. In Proceedings of the 10th asia pacific conference on Computer human interaction(APCHI '12). ACM, New York, NY, USA, 299-308. DOI: <https://doi.org/10.1145/2350046.2350098>
109. Markus Löchtefeld, Christoph Hirtz, and Sven Gehring. 2013. Evaluation of hybrid front- and back-of-device interaction on mobile devices. In Proceedings of the 12th International Conference on Mobile and Ubiquitous Multimedia (MUM '13). ACM, New York, NY, USA, Article 17, 4 pages. DOI: <https://doi.org/10.1145/2541831.2541865>
110. Logiciel FellowUp : <https://www.crunchbase.com/organization/fellowup> (dernière visite le 05/07/2018)
111. Logiciel NetVibes : <https://www.netvibes.com/fr> (dernière visite le 05/07/2018)

112. I. Scott MacKenzie, R. William Soukoreff, and Chris Pal. 1997. A two-ball mouse affords three degrees of freedom. In CHI '97 Extended Abstracts on Human Factors in Computing Systems (CHI EA '97). ACM, New York, NY, USA, 303-304. DOI: <https://doi.org/10.1145/1120212.1120405>
113. Nicolai Marquardt, Frederik Brudy, Can Liu, Ben Bengler, and Christian Holz. 2018. SurfaceConstellations: A Modular Hardware Platform for Ad-Hoc Reconfigurable Cross-Device Workspaces. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18). ACM, New York, NY, USA, Paper 354, 14 pages. DOI: <https://doi.org/10.1145/3173574.3173928>
114. Marteniuk, R. G., MacKenzie, C., Jeannerod, M., Athenes, S., and Dugas, C. Constraints on human arm movement trajectories. *Canadian Journal of Psychology* 41, 3 (1987), 365--378
115. Anthony Martinet, Gery Casiez, and Laurent Grisoni. 2012. Integrality and Separability of Multitouch Interaction Techniques in 3D Manipulation Tasks. *IEEE Transactions on Visualization and Computer Graphics* 18, 3 (March 2012), 369-380. DOI=<http://dx.doi.org/10.1109/TVCG.2011.129>
116. David Merrill, Jeevan Kalanithi, and Pattie Maes. 2007. Siftables: towards sensor network user interfaces. In Proceedings of the 1st international conference on Tangible and embedded interaction. ACM, 75--78
117. Takashi Miyaki and Jun Rekimoto. 2009. GraspZoom: Zooming and Scrolling Control Model for Single-Handed Mobile Interaction. In Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '09). ACM, New York, NY, USA, Article 11, 4 pages. DOI: <http://dx.doi.org/10.1145/1613858.1613872>
118. Tomer Moscovich and John F. Hughes. 2008. Indirect mappings of multi-touch input using one and two hands. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08). ACM, New York, NY, USA, 1275-1284. DOI: <https://doi.org/10.1145/1357054.1357254>
119. Mathieu Nancel, Olivier Chapuis, Emmanuel Pietriga, Xing-Dong Yang, Pourang P. Irani, and Michel Beaudouin-Lafon. 2013. High-precision pointing on large wall displays using small handheld devices. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13). ACM, New York, NY, USA, 831-840. DOI: <https://doi.org/10.1145/2470654.2470773>

120. Laurence Nigay and Joëlle Coutaz. 1993. A design space for multimodal systems: concurrent processing and data fusion. In Procs of INTERACT'93 and CHI '93. ACM, New York, NY, USA, 172-178.
121. Ian Oakley and Doyoung Lee. 2014. Interaction on the edge: offset sensing for small devices. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14). ACM, New York, NY, USA, 169-178. DOI: <https://doi.org/10.1145/2556288.2557138>
122. Ian Oakley and Junseok Park. 2007. A motion-based marking menu system. In CHI '07 Extended Abstracts on Human Factors in Computing Systems (CHI EA '07). ACM, New York, NY, USA, 2597-2602. DOI: <https://doi.org/10.1145/1240866.1241048>
123. Olafsdottir, H., Tsandilas, T. and Appert, C. 2014. Prospective motor control on tabletops: planning grasp for multitouch interaction. CHI '14. ACM, 2893-2902.
124. Seonwook Park, Christoph Gebhardt, Roman Rädle, Anna Maria Feit, Hana Vrzakova, Niraj Ramesh Dayama, Hui-Shyong Yeo, Clemens N. Klokmoose, Aaron Quigley, Antti Oulasvirta, and Otmar Hilliges. 2018. AdaM: Adapting Multi-User Interfaces for Collaborative Environments in Real-Time. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18). ACM, New York, NY, USA, Paper 184, 14 pages. DOI: <https://doi.org/10.1145/3173574.3173758>
125. Kurt Partridge, Saurav Chatterjee, Vibha Sazawal, Gaetano Borriello, and Roy Want. 2002. TiltType: accelerometer-supported text entry for very small devices. In Proceedings of the 15th annual ACM symposium on User interface software and technology (UIST '02). ACM, New York, NY, USA, 201-204. DOI: <https://doi.org/10.1145/571985.572013>
126. Pathvisio : <https://www.pathvisio.org/> (dernière visite le 05/07/2018)
127. Gary Perelman, Marcos Serrano, Mathieu Raynal, Emmanuel Dubois, Célia Picard, and Mustapha Derras. 2014. Designing an input device to interact with multidimensional data : disco. In Proceedings of the 26th Conference on l'Interaction Homme-Machine (IHM '14). ACM, New York, NY, USA, 91-100. DOI: <https://doi.org/10.1145/2670444.2670453>
128. Gary Perelman, Marcos Serrano, Mathieu Raynal, Celia Picard, Mustapha Derras, and Emmanuel Dubois. 2015. The Roly-Poly Mouse: Designing a Rolling Input Device Unifying 2D and 3D Interaction. In Proceedings of the 33rd Annual

- ACM Conference on Human Factors in Computing Systems (CHI '15). ACM, New York, NY, USA, 327-336. DOI: <https://doi.org/10.1145/2702123.2702244>
129. 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>
  130. Gary Perelman, Marcos Serrano, Célia Picard, Mustapha Derras, and Emmanuel Dubois. 2017. Multi-device interaction for spreadsheet on tablets. In Proceedings of the 29th Conference on l'Interaction Homme-Machine (IHM '17). ACM, New York, NY, USA, 105-115. DOI: <https://doi.org/10.1145/3132129.3132132>
  131. Gary Perelman, Marcos Serrano, Celia Picard, Christophe Bortolaso, Mustapha Derras, and Emmanuel Dubois. 2018. Cell Selection for Spreadsheets on Tablets: Stacking-Based Interaction. In Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems (CHI EA '18). ACM, New York, NY, USA, Paper D505, 4 pages. DOI: <https://doi.org/10.1145/3170427.3186546>
  132. Perrault, S., Lecolinet, E., Eagan, J., and Guiard, Y. 2013. Watchit: simple gestures and eyes-free interaction for wristwatches and bracelets. CHI '13. ACM, 1451-60
  133. Ken Pfeuffer, Ken Hinckley, Michel Pahud, and Bill Buxton. 2017. Thumb + Pen Interaction on Tablets. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems(CHI '17). ACM, New York, NY, USA, 3254-3266. DOI: <https://doi.org/10.1145/3025453.3025567>
  134. Phantom: <https://fr.3dsystems.com/haptics-devices/3d-systems-phantom-premium> (dernière visite le 05/07/2018)
  135. 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: <https://doi.org/10.1145/2525194.2525205>
  136. Thomas Pietrzak, Sylvain Malacria, and Gilles Bailly. 2014. CtrlMouse et TouchCtrl: duplicating mode delimiters on the mouse. In Proceedings of the 26th Conference on l'Interaction Homme-Machine (IHM '14). ACM, New York, NY, USA, 38-47. DOI: <https://doi.org/10.1145/2670444.2670447>

137. Polhemus Patriot : <https://polhemus.com/motion-tracking/all-trackers/patriot/> (dernière visite le 05/07/2018)
138. Logiciel Powerpoint : <https://products.office.com/fr-fr/powerpoint> (dernière visite le 05/07/2018)
139. Roman Rädle, Hans-Christian Jetter, Nicolai Marquardt, Harald Reiterer, and Yvonne Rogers. 2014. HuddleLamp: Spatially-Aware Mobile Displays for Ad-hoc Around-the-Table Collaboration. In Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces (ITS '14). ACM, New York, NY, USA, 45-54. DOI: <https://doi.org/10.1145/2669485.2669500>
140. Roman Rädle, Hans-Christian Jetter, Mario Schreiner, Zhihao Lu, Harald Reiterer, and Yvonne Rogers. 2015. Spatially-aware or Spatially-agnostic?: Elicitation and Evaluation of User-Defined Cross-Device Interactions. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15). ACM, New York, NY, USA, 3913-3922. DOI: <https://doi.org/10.1145/2702123.2702287>
141. Mahfuz Rahman, Sean Gustafson, Pourang Irani, and Sriram Subramanian. 2009. Tilt techniques: investigating the dexterity of wrist-based input. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09). ACM, New York, NY, USA, 1943-1952. DOI: <https://doi.org/10.1145/1518701.1518997>
142. Gonzalo Ramos , Ravin Balakrishnan, Zliding: fluid zooming and sliding for high precision parameter manipulation, Proceedings of the 18th annual ACM symposium on User interface software and technology, October 23-26, 2005, Seattle, WA, USA [[doi>10.1145/1095034.1095059](https://doi.org/10.1145/1095034.1095059)]
143. Ramos, G., M. Boulos, and R. Balakrishnan. Pressure Widgets. In Proc. CHI'04. ACM Press. p. 527 - 534
144. Razer naga mouse. <http://www.razerzone.com/gamingmice/razer-naga> (dernière visite 05/07/2018)
145. Jun Rekimoto and Carsten Schwesig. 2006. PreSenseII: Bi-Directional Touch and Pressure Sensing Interactions with Tactile Feedback. In CHI '06 Extended Abstracts on Human Factors in Computing Systems (CHI EA '06). ACM, New York, NY, USA, 1253–1258. DOI: <http://dx.doi.org/10.1145/1125451.1125685>
146. Jun Rekimoto. 1996. Tilting operations for small screen interfaces. In Proceedings of the 9th annual ACM symposium on User interface software and

- technology (UIST '96). ACM, New York, NY, USA, 167-168. DOI=<http://dx.doi.org/10.1145/237091.237115>
147. Christian Rendl, David Kim, Patrick Parzer, Sean Fanello, Martin Zirkl, Gregor Scheipl, Michael Haller, and Shahram Izadi. 2016. FlexCase: Enhancing Mobile Interaction with a Flexible Sensing and Display Cover. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems(CHI '16). ACM, New York, NY, USA, 5138-5150. DOI: <https://doi.org/10.1145/2858036.2858314>
148. Lycée professionnel Renée Bonnet : <http://renee-bonnet.entmip.fr/> (dernière visite le 05/07/2018)
149. Mike Rooke and Roel Vertegaal. 2010. Physics on display: tangible graphics on hexagonal bezel-less screens. In Proceedings of the fourth international conference on Tangible, embedded, and embodied interaction (TEI '10). ACM, New York, NY, USA, 233-236. DOI=<http://dx.doi.org/10.1145/1709886.1709932>
150. Anne Roudaut, Eric Lecolinet, and Yves Guiard. 2009. MicroRolls: expanding touch-screen input vocabulary by distinguishing rolls vs. slides of the thumb. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09). ACM, New York, NY, USA, 927-936. DOI: <https://doi.org/10.1145/1518701.1518843>
151. Anne Roudaut, Mathias Baglioni, and Eric Lecolinet. 2009. TimeTilt: Using Sensor-Based Gestures to Travel through Multiple Applications on a Mobile Device. In Proceedings of the 12th IFIP TC 13 International Conference on Human-Computer Interaction: Part I (INTERACT '09), Tom Gross, Jan Gulliksen, Paula Kotzé, Lars Oestreicher, Philippe Palanque, Raquel Oliveira Prates, and Marco Winckler (Eds.). Springer-Verlag, Berlin, Heidelberg, 830-834. DOI: [https://doi.org/10.1007/978-3-642-03655-2\\_90](https://doi.org/10.1007/978-3-642-03655-2_90)
152. Jaime Ruiz and Yang Li. 2011. DoubleFlip: a motion gesture delimiter for mobile interaction. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11). ACM, New York, NY, USA, 2717-2720. DOI: <https://doi.org/10.1145/1978942.1979341>
153. Jeffrey M. Rzeszotarski and Aniket Kittur. 2014. Kinetica: naturalistic multi-touch data visualization. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14). ACM, New York, NY, USA, 897-906. DOI: <https://doi.org/10.1145/2556288.2557231>

154. R. Sadana and J. Stasko. 2016. Designing Multiple Coordinated Visualizations for Tablets. *Comput. Graph. Forum* 35, 3 (June 2016), 261-270. DOI: <https://doi.org/10.1111/cgf.12902>
155. Ramik Sadana and John Stasko. 2016. Expanding Selection for Information Visualization Systems on Tablet Devices. In *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces (ISS '16)*. ACM, New York, NY, USA, 149-158. DOI: <https://doi.org/10.1145/2992154.2992157>
156. Houssein Saidi, Marcos Serrano, and Emmanuel Dubois. 2016. Investigating the effects of splitting detailed views in Overview+Detail interfaces. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '16)*. ACM, New York, NY, USA, 180-184. DOI: <https://doi.org/10.1145/2935334.2935341>
157. Houssein Saidi, Marcos Serrano, Pourang Irani, and Emmanuel Dubois. 2017. TDome: A Touch-Enabled 6DOF Interactive Device for Multi-Display Environments. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 5892-5904. DOI: <https://doi.org/10.1145/3025453.3025661>
158. Stephanie Santosa , Daniel Wigdor, A field study of multi-device workflows in distributed workspaces, *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, September 08-12, 2013, Zurich, Switzerland doi>10.1145/2493432.2493476
159. Vibha Sazawal, Roy Want, and Gaetano Borriello. 2002. The Unigesture Approach. In *Proceedings of the 4th International Symposium on Mobile Human-Computer Interaction (Mobile HCI '02)*, Fabio Paternò (Ed.). Springer-Verlag, London, UK, UK, 256-270
160. 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.
161. Stefano Scheggi, Gionata Salvietti, and Domenico Prattichizzo. 2010. Shape and weight rendering for haptic augmented reality. In *Proceedings of the 19th International Symposium in Robot and Human Interactive Communication*, 44--49

162. Sebastian Schmidt, Miguel A. Nacenta, Raimund Dachsel, and Sheelagh Carpendale. 2010. A set of multi-touch graph interaction techniques. In ACM International Conference on Interactive Tabletops and Surfaces (ITS '10). ACM, New York, NY, USA, 113-116. DOI: <https://doi.org/10.1145/1936652.1936673>
163. Dominik Schmidt, Fadi Chehimi, Enrico Rukzio, and Hans Gellersen. 2010. PhoneTouch: a technique for direct phone interaction on surfaces. In Proceedings of the 23rd annual ACM symposium on User interface software and technology (UIST '10). ACM, New York, NY, USA, 13-16. DOI: <https://doi.org/10.1145/1866029.1866034>
164. Mario Schreiner, Roman Rädle, Hans-Christian Jetter, and Harald Reiterer. 2015. Connichiwa: A Framework for Cross-Device Web Applications. In Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '15). ACM, New York, NY, USA, 2163-2168. DOI: <https://doi.org/10.1145/2702613.2732909>
165. James Scott, Shahram Izadi, Leila Sadat Rezai, Dominika Ruskowski, Xiaojun Bi, and Ravin Balakrishnan. 2010. RearType: text entry using keys on the back of a device. In Proceedings of the 12th international conference on Human computer interaction with mobile devices and services (MobileHCI '10). ACM, New York, NY, USA, 171-180. DOI: <https://doi.org/10.1145/1851600.1851630>
166. Julian Seifert, Sebastian Boring, Christian Winkler, Florian Schaub, Fabian Schwab, Steffen Herrdum, Fabian Maier, Daniel Mayer, and Enrico Rukzio. 2014. Hover Pad: interacting with autonomous and self-actuated displays in space. In Proceedings of the 27th annual ACM symposium on User interface software and technology (UIST '14). ACM, New York, NY, USA, 139-147. DOI: <https://doi.org/10.1145/2642918.2647385>
167. Marcos Serrano, Laurence Nigay, Jean-Yves L. Lawson, Andrew Ramsay, Roderick Murray-Smith, and Sebastian Deneff. 2008. The openinterface framework: a tool for multimodal interaction. In CHI EA '08. ACM, New York, NY, USA, 3501-3506.
168. Marcos Serrano. Interaction multimodale en entrée : Conception et Prototypage. Interface hommemachine [cs.HC]. Université de Grenoble, 2010. Français.
169. Teddy Seyed, Xing-Dong Yang, and Daniel Vogel. 2016. Doppio: A Reconfigurable Dual-Face Smartwatch for Tangible Interaction. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16).

- ACM, New York, NY, USA, 4675-4686. DOI: <https://doi.org/10.1145/2858036.2858256>
170. Erh-li Early Shen, Sung-sheng Daniel Tsai, Hao-hua Chu, Yung-jen Jane Hsu, and Chi-wen Euro Chen. 2009. Double-side multi-touch input for mobile devices. In CHI '09 Extended Abstracts on Human Factors in Computing Systems (CHI EA '09). ACM, New York, NY, USA, 4339-4344. DOI: <https://doi.org/10.1145/1520340.1520663>
  171. David Sirkin and Wendy Ju. 2014. Make this!: introduction to electronics prototyping using arduino. In CHI EA '14.
  172. Soukoreff, R. W., & MacKenzie, I. S. (2004). Towards a standard for pointing device evaluation: Perspectives on 27 years of Fitts' law research in HCI. *International Journal of Human-Computer Studies*, 61, 751-789.
  173. SpaceMouse. Press Note March 31, 2011. <http://www.3dconnexion.fr/nc/company/press-room/> (dernière visite 05/07/2018)
  174. Sphero : <https://www.sphero.com/> (dernière visite le 05/07/2018)
  175. Steed, A. and Slater, M. 1995. 3D Interaction with the Desktop Bat. In *Computer Graphics Forum*. Blackwell Publishers, vol 14, 97-104.
  176. Masanori Sugimoto and Keiichi Hiroki. 2006. HybridTouch: an intuitive manipulation technique for PDAs using their front and rear surfaces. In *Proceedings of the 8th conference on Human-computer interaction with mobile devices and services (MobileHCI '06)*. ACM, New York, NY, USA, 137-140. DOI=<http://dx.doi.org/10.1145/1152215.1152243>
  177. Taylor, B. and Bove, M. 2009. Graspables: grasprecognition as a user interface. CHI '09. ACM, 917-926
  178. Tian, F., L. Xu, H. Wang, X. Zhang, Y. Liu, V. Setlur, and G. Dai. Tilt menu: using the 3D orientation information of pen devices to extend the selection capability of pen-based user interfaces. In *Proc. CHI'08*. ACM Press. p. 1371-1380
  179. The trackball keyboard. <http://www.ione-usa.com/ionescorpius-35-trackball-keyboard.html>
  180. Tsandilas, T., Dubois, E., Raynal, M. 2013. Modeless Pointing with Low-Precision Wrist Movements. *INTERACT 2013*. Springer, p. 494-511.

181. Jayson Turner, Jason Alexander, Andreas Bulling, and Hans Gellersen. 2015. Gaze+RST: Integrating Gaze and Multitouch for Remote Rotate-Scale-Translate Tasks. In *Procs of CHI '15*. ACM, New York, NY, USA, 4179-4188
182. Md. Sami Uddin and Carl Gutwin. 2016. Rapid Command Selection on Multi-Touch Tablets with Single-Handed HandMark Menus. In *Proceedings of the 2016 ACM on Interactive Surfaces and Spaces (ISS '16)*. ACM, New York, NY, USA, 205-214. DOI: <https://doi.org/10.1145/2992154.2992172>
183. Anna Vallgård and Ylva Fernaeus. 2015. Interaction Design as a Bricolage Practice. In *Procs of TEI '15*. ACM, New York, NY, USA, 173-180.
184. Gary R. VandenBos (Ed.). 2009. *Publication Manual of the American Psychological Association (6th ed.)*. American Psychological Association, Washington, DC. <http://www.apastyle.org/manual/>
185. Villar, N., Izadi, S., Rosenfeld, D., Benko, H., Helmes, J., Westhues, J., Hodges, S., Ofek, E., Butler, A., Cao, X., and Chen, B. 2009. Mouse 2.0: multi-touch meets the mouse. *UIST '09*. ACM, 33-42.
186. Vizable by Tableau. Retrieved September 21, 2016 from <https://vizable.tableau.com/>
187. Simon Voelker, Kosuke Nakajima, Christian Thoresen, Yuichi Itoh, Kjell Ivar Øvergård, and Jan Borchers. 2013. PUCs: detecting transparent, passive untouched capacitive widgets on unmodified multi-touch displays. In *Proceedings of the 2013 ACM international conference on Interactive tabletops and surfaces (ITS '13)*. ACM, New York, NY, USA, 101-104. DOI: <https://doi.org/10.1145/2512349.2512791>
188. Daniel Vogel and Géry Casiez. 2011. Conté: multimodal input inspired by an artist's crayon. In *Proceedings of the 24th annual ACM symposium on User interface software and technology (UIST '11)*. ACM, New York, NY, USA, 357-366. DOI: <https://doi.org/10.1145/2047196.2047242>
189. Wacom Technology Corporation. Product Information. (<http://wacom.com/>).
190. Julie Wagner, Eric Lecolinet, and Ted Selker. 2014. Multi-finger chords for hand-held tablets: recognizable and memorable. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 2883-2892. DOI: <https://doi.org/10.1145/2556288.2556958>
191. Julie Wagner, Stéphane Huot, and Wendy Mackay. 2012. BiTouch and BiPad: designing bimanual interaction for hand-held tablets. In *Proceedings of the*

- SIGCHI Conference on Human Factors in Computing Systems (CHI '12). ACM, New York, NY, USA, 2317-2326. DOI: <http://dx.doi.org/10.1145/2207676.2208391>
192. Jagoda Walny, Bongshin Lee, Paul Johns, Nathalie Henry Riche, and Sheelagh Carpendale. 2012. Understanding Pen and Touch Interaction for Data Exploration on Interactive Whiteboards. *IEEE Transactions on Visualization and Computer Graphics* 18, 12 (December 2012), 2779-2788. DOI: <https://doi.org/10.1109/TVCG.2012.275>
193. Ware, C., & Slipp, L. (1991). Using velocity control to navigate 3D graphical environments: a comparison of three interfaces. In *Proceedings of Human Factors Society 35th Annual Meeting.*, San Francisco, CA.
194. Daniel Wigdor, Clifton Forlines, Patrick Baudisch, John Barnwell, and Chia Shen. 2007. Lucid touch: a see-through mobile device. In *Proceedings of the 20th annual ACM symposium on User interface software and technology (UIST '07)*. ACM, New York, NY, USA, 269-278. DOI: <https://doi.org/10.1145/1294211.1294259>
195. Wing mouse : <https://www.worthingtonsharpe.com/> (dernière visite le 12/12/2018)
196. Jacob O. Wobbrock, Meredith Ringel Morris, and Andrew D. Wilson. 2009. User-defined gestures for surface computing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. ACM, New York, NY, USA, 1083-1092. DOI: <https://doi.org/10.1145/1518701.1518866>
197. 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: <https://doi.org/10.1145/2639189.2639237>
198. Mike Wu, Chia Shen, Kathy Ryall, Clifton Forlines, and Ravin Balakrishnan. 2006. Gesture Registration, Relaxation, and Reuse for Multi-Point Direct-Touch Surfaces. In *Proceedings of the First IEEE International Workshop on Horizontal Interactive Human-Computer Systems (TABLETOP '06)*. IEEE Computer Society, Washington, DC, USA, 185-192. DOI: <https://doi.org/10.1109/TABLETOP.2006.19>

199. Chi-Jui Wu, Steven Houben, and Nicolai Marquardt. 2017. EagleSense: Tracking People and Devices in Interactive Spaces using Real-Time Top-View Depth-Sensing. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17). ACM, New York, NY, USA, 3929-3942. DOI: <https://doi.org/10.1145/3025453.3025562>
200. Robert Xiao, Greg Lew, James Marsanico, Divya Hariharan, Scott Hudson, and Chris Harrison. 2014. Toffee: enabling ad hoc, around-device interaction with acoustic time-of-arrival correlation. In Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services (MobileHCI '14). ACM, New York, NY, USA, 67-76. DOI: <https://doi.org/10.1145/2628363.2628383>
201. Xiang Xiao, Teng Han, and Jingtao Wang. 2013. LensGesture: augmenting mobile interactions with back-of-device finger gestures. In Proceedings of the 15th ACM on International conference on multimodal interaction (ICMI '13). ACM, New York, NY, USA, 287-294. DOI: <https://doi.org/10.1145/2522848.2522850>
202. X-IMU : [x-io.co.uk/x-imu/](http://x-io.co.uk/x-imu/) (dernière visite le 05/07/2018)
203. Xing-Dong Yang, Edward Mak, Pourang Irani, and Walter F. Bischof. 2009. Dual-Surface input: augmenting one-handed interaction with coordinated front and behind-the-screen input. In Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '09). ACM, New York, NY, USA, , Article 5 , 10 pages. DOI=<http://dx.doi.org/10.1145/1613858.1613865>
204. Yang, X-D., Mak, E., McCallum, D., Irani, P., Cao, X., and Izadi, S. 2010. LensMouse: augmenting the mouse with an interactive touch display. CHI '10. ACM, 2431-40.
205. Emanuel Zraggen, Robert Zeleznik, and Steven M. Drucker. 2014. PanoramicData: data analysis through pen & touch. IEEE Transactions on Visualization and Computer Graphics (TVCG) 20, 12: 2112--2121. [doi>[10.1109/TVCG.2014.2346293](https://doi.org/10.1109/TVCG.2014.2346293)]
206. Zhai, S. Human Performance in Six Degree of Freedom Input Control. PhD thesis, University of Toronto, 1995.
207. Zhai, S. 1998. User performance in relation to 3D input device design. SIGGRAPH Comput. Graph. 32,4,50-54.

208. Chen Zhao, Ke-Yu Chen, Md Tanvir Islam Aumi, Shwetak Patel, and Matthew S. Reynolds. 2014. SideSwipe: detecting in-air gestures around mobile devices using actual GSM signal. In Proceedings of the 27th annual ACM symposium on User interface software and technology (UIST '14). ACM, New York, NY, USA, 527-534. DOI: <https://doi.org/10.1145/2642918.2647380>



# LISTE DES FIGURES

---

Figure 1. Schéma récapitulatif: les degrés de liberté des dispositifs n'étant pas suffisant pour interagir avec des données multidimensionnelles, une combinaison de ceux-ci est utilisée, complexifiant l'interaction. ....	16
Figure 2. Illustration d'un imprimeur de Berger-Levrault. ....	23
Figure 3. Carte du monde représentant les pays dans lesquels Berger-Levrault possède des agences, des filiales et des clients. ....	24
Figure 4. Capture d'écran du logiciel e.sedit RH. ....	25
Figure 5. Capture d'écran du logiciel Cabinet Numérique. ....	27
Figure 6. Capture d'écran du logiciel Attal II. ....	27
Figure 7. Capture d'écran du logiciel Commune.IT. ....	28
Figure 8. Capture d'écran du logiciel BL.Social. ....	30
Figure 9. Nombre d'élus du conseil municipal par collectivité. ....	32
Figure 10. La Rockin'Mouse (Gauche, tiré de [7]) et la VideoMouse (Droite, tiré de [78]). .....	45
Figure 11. Exemple de dispositif à multiple degrés de liberté: le PHANToM. ....	46
Figure 12. La LensMouse (Gauche, tiré de [198]) et l'Inflatable Mouse (Droite, tiré de [94]). ....	47
Figure 13. La SpaceBall 4000 FLX. ....	48
Figure 14. Ensemble de gestes tactiles pour la manipulation d'un graphe (tiré de [157]). .....	51
Figure 15. Sélection d'un élément de pie-menu en multi-touch (Gauche, tiré de [9]) et principe de fonctionnement des touchers consécutifs (Droite, tiré de [76]). ....	53
Figure 16. Techniques d'interaction combinant l'utilisation du tactile et du stylet pour manipuler un tableur sur tablette (tiré de [128]). ....	54
Figure 17. Utilisation de la rotation (Gauche, tiré de [17]) et de l'inclinaison du stylet (Droite, tiré de [173]). ....	55
Figure 18. Utilisation de l'espace autour (Gauche, tiré de [73]) ou au-dessus (Droite, tiré de [74]) d'un dispositif mobile. ....	56
Figure 19. Utilisation des reflets (Gauche, tiré de [64]) ou de capteurs infra-rouges (Droite, tiré de [21]) pour la détection de l'interaction autour d'un dispositif mobile. ...	57

Figure 20. Manipulation tangible d'un dispositif mobile pour la manipulation 3D (Gauche, tiré de [15]) et pour la navigation dans un ensemble d'applications (Droite, tiré de [5]). .....	58
Figure 21. Utilisation de l'arrière du dispositif pour l'authentification (Gauche, tiré de [40]) ou la saisie de texte (Droite, tiré de [160]). .....	59
Figure 22. Ensemble de capteurs arduino (pression, infra-rouge, lumière, etc.) .....	60
Figure 23. Interface du logiciel OpenInterface. .....	61
Figure 24. Cinq relations d'Allen décrivant la composition temporelle de modalités d'interaction (tiré de [163]). .....	62
Figure 25. Interaction spatiale avec plusieurs dispositifs mobiles (Gauche, tiré de [144]) et combinaison physique de dispositifs mobiles (Droite, tiré de [96]). .....	63
Figure 26. Combinaison de dispositifs mobiles: un smartphone avec une smartwatch (Gauche, tiré de [31]) ainsi qu'un smartphone avec une tablette (Droite, tiré de [191]).	64
Figure 27. Interaction basée sur le principe du stacking avec un écran d'ordinateur (Haut, tiré de [105]) ou une table tactile (Bas, tiré de [158]). .....	65
Figure 28. La Roly-Poly Mouse, basée sur le culbuto (gauche) possède un centre de gravité bas (droite), ce qui lui permet de revenir à la verticale après avoir été relâché.	71
Figure 29. Couvertures de la RPM : Hémisphérique (gauche), Convexe (centre) et Concave (droite). .....	75
Figure 30. Prototypes de la RPM basés sur la technologie OptiTrack (gauche). Dispositif des caméras utilisées pour localiser le dispositif (droite). .....	77
Figure 31. Prototype intégré de la RPM. .....	78
Figure 32. Catégories de prises en main du dispositif : Squeeze (a), Lay (b) et Touch (c). .....	80
Figure 33. Configuration spatiale de l'expérimentation : lignes et cercles de manipulation (gauche) et disposition de la RPM dans son environnement de localisation (droite). ...	82
Figure 34. Amplitude de rotation (gauche) par prise en main. Amplitude d'inclinaison par prise en main pour la RPM hémisphérique (centre) et convexe (droite). .....	83
Figure 35. Déviation entre la direction de l'inclinaison demandée et celle réalisée. ....	83
Figure 36. Problème du pointage 2D lié à l'utilisation d'un dispositif inclinable. .....	86
Figure 37. Représentation schématique du cercle support et des huit cercles cibles. ....	87
Figure 38. Temps de réalisation moyen (gauche), taux d'erreur (centre) pour chaque technique d'interaction et indice de difficulté. Modèle de Fitts pour chaque technique d'interaction (droite). .....	89

Figure 39. La SpaceMouse.....	90
Figure 40. Technique d'interaction pour la translation en 3D : mode novice en deux gestes (gauche) et mode expert en 1 geste combiné (droite).....	91
Figure 41. Technique d'interaction pour la rotation 3D.....	92
Figure 42. Environnement 3D pour la tâche de pointage (gauche) et pour la tâche de rotation (droite).....	93
Figure 43. Temps de réalisation de la tâche de pointage 3D (gauche) pour chaque technique d'interaction et indice de difficulté. Préférence utilisateur pour chaque technique d'interaction (droite).....	96
Figure 44. TDome: un dispositif combinant les avantages des modes d'interaction isotoniques et isométriques.....	104
Figure 45. Axe de l'arrangement physique et ses propriétés.....	111
Figure 46. Illustration des trois catégories de topologie : Imbriqué comme dans Lexip 3D [106], Attaché comme dans Codex [80] et Séparé comme dans Siftables [116]. .....	112
Figure 47. Illustration des trois types de fusion : (A-B) comme dans Billboard [96], (A←B) comme dans LensMouse [198] et (A↔B) comme dans Ink222 [85]. .....	113
Figure 48. Illustration de la propriété de dynamicité : statique comme dans la Razer Naga [139] et dynamique comme dans les Hexagonal bezel-less screens [144].....	114
Figure 49. Axe de la manipulation physique et ses propriétés.....	115
Figure 50. Illustration de la propriété effecteurs corporels. Trackball keyboard [174]: <clavier : main non dom.; trackball : main dom.> (gauche). La Inflatable Mouse [94]: <souris : main dom.; ballon : main dom.> (droite).....	115
Figure 51. Illustration de la propriété d'usage temporel : séquentiel comme dans FlipKeyboard (gauche) et parallèle comme dans HybridTouch (droite).....	116
Figure 52. De gauche à droite, la Cubic Mouse [54], la GlobeMouse [53], la PadMouse [8] et la CapMouse [180].....	119
Figure 53. Prototypes en carton de la RPM 2 : la souris est placée sur le côté droit (gauche), à l'arrière (centre) ou sous (droite) la RPM.....	123
Figure 54. Prototype v2 de la RPM 2. Ce prototype est basé sur le type de fusion (A ← B). .....	125
Figure 55. Différentes tailles de RPM, de gauche à droite : 4, 5, 6, 8 et 10cm.....	125
Figure 56. Prototype v3a (Gauche) et v3b (Droite) de la RPM 2. Ces prototypes diffèrent dans la position de la RPM par rapport à la souris ainsi que dans la disposition des boutons. ....	127

Figure 57. Prototype v3 de la RPM 2 avec des boutons plus doux. Pour détecter les manipulations de la RPM, celle-ci a été équipée de billes infrarouges (Droite). .....	127
Figure 58. Exemple de tâche de RST. ....	128
Figure 59. Version séparée de la RPM 2. ....	129
Figure 60. Illustration du stacking d'un smartphone sur l'écran d'une tablette pour l'utilisation d'un tableur : sélection d'une plage de cellules (A), visualisation d'un graphique (B), calcul d'une moyenne (C) et édition d'une cellule (D). ....	138
Figure 61. Gestes physiques du "stacking": Translation (A), rotation à plat (B), inclinaison (C) et rotation de coin (D). ....	142
Figure 62. Exemples d'orientation initiale : 0° et 90° en mode portrait (A et B) puis -45° et 45° en mode paysage (C et D). ....	143
Figure 63. Coque conductrice en cuivre (gauche). stacking de notre prototype sur l'écran de la tablette (droite). ....	145
Figure 64. Prototype usiné en aluminium. ....	146
Figure 65. Sélection d'une plage de cellules : Appui sur une cellule (1), sélection de l'ancre (2), glisser pour étendre la sélection (3). ....	150
Figure 66. Technique Intersection : stack (1) et translation (2) pour sélectionner des colonnes et stack (3) et translation (4) pour sélectionner des lignes. ....	150
Figure 67. Défilement vertical (gauche) et horizontal (droite). La zone représentée en vert désigne la zone dans laquelle le mécanisme de défilement peut être activé. ....	151
Figure 68. Technique Stack-and-Drag : stack du smartphone (1) et translation (2) pour étendre la sélection. ....	152
Figure 69. Technique Finger Stack : toucher pour sélectionner le coin haut-gauche (1) et stack pour sélectionner le coin bas-droit (2) de la plage. ....	152
Figure 70. Différentes plages de cellules (de gauche à droite et de haut en bas) : carré, une colonne, rectangle horizontal et colonnes complètes. ....	155
Figure 71. Première vue de l'application (gauche), exemple de sélection (centre) et validation automatique (droite). La couleur des en-têtes de cellule sont colorés selon l'état de la sélection (vert pour sélectionnés et inclus dans la plage cible, jaune pour sélectionnés mais non-inclus dans la plage cible et rouge pour non sélectionnés mais inclus dans la plage cible). ....	156
Figure 72. Temps de sélection en s (Gauche) et ratio (Droite) par rapport à Tactile (95% IC). ....	158
Figure 73. Temps de sélection (en ms) par type de plage et par technique. ....	158

Figure 74. Score SUS par technique d'interaction (95% IC).....	158
Figure 75. Tableau croisé dynamique dans Excel sur Microsoft Surface. ....	160
Figure 76. Techniques Dual-Screen touch et Stack-and-Touch. ....	161
Figure 77. Technique Stack-and-Translate.....	162
Figure 78. Écran principal de l'application (Gauche), Tableau croisé dynamique avec la technique Tactile (Milieu) et Tableau croisé dynamique sur le smartphone (Droite)..	163
Figure 79. Temps d'affectation par technique d'interaction (en ms, 95% CI).....	166
Figure 80. Score SUS par technique d'interaction (95% CI).....	166
Figure 81. Onglets de l'application TreeTech.....	177
Figure 82. Photo d'un arbre malade vue de face et de profil. ....	178
Figure 83. Ensemble d'arbres d'un parking d'une collectivité (tiré de Google Street View). .....	180
Figure 84. Solutions de localisation d'un arbre, de gauche à droite: par surbrillance, à l'aide de flèches indiquant la direction ou par triangulation.....	181
Figure 85. Localisation de la position d'une observation en le comparant à une photographie (Gauche) et indications des précédentes observations (Droite). ....	183
Figure 86. Calcul de la télémétrie de l'arbre à l'aide d'un objet étalon: une tablette. ...	184
Figure 87. Scénario complet d'utilisation de notre technique d'interaction, de gauche à droite : déplacement jusqu'à 10m de l'arbre à l'aide du GPS embarqué, triangulation à l'aide d'objets urbains, affichage de la direction de l'arbre par rapport à la position actuelle (flèche rouge) et position des précédentes observations et mise en évidence de l'arbre à l'aide de photos des précédentes observations. ....	186
Figure 88. Trois écrans de l'application Attal Mobile. ....	189
Figure 89. Barre de menu pour les caractéristiques de l'arbre.....	192
Figure 90. Interaction bimanuelle: la main gauche permet de sélectionner un sous-menu et la main droite permet de choisir une caractéristique et la manipuler. ....	193
Figure 91. Représentation schématique des caractéristiques d'un arbre. ....	194
Figure 92. Exemples d'accords tactiles existants dans la littérature (tirés de [9, 76])..	195
Figure 93. Barre de menu avec zoom selon la position du doigt. ....	196
Figure 94. Barre de menu navigable.....	197
Figure 95. Barre de menu triple.....	198
Figure 96. Barre de menu hiérarchique. ....	198
Figure 97. Amplitude de rotation (gauche) par prise en main. Amplitude d'inclinaison par prise en main pour la RPM hémisphérique (centre) et convexe (droite).....	218

Figure 98. Déviation entre la direction de l'inclinaison demandée et celle réalisée. ...	219
Figure 99. Temps de réalisation moyen (gauche), taux d'erreur (centre) pour chaque technique d'interaction et indice de difficulté. Modèle de Fitts pour chaque technique d'interaction (droite).....	220
Figure 100. Temps de réalisation de la tâche de pointage 3D (gauche) pour chaque technique d'interaction et indice de difficulté. Préférence utilisateur pour chaque technique d'interaction (droite). ....	222
Figure 101. Temps de sélection en s (Gauche) et ratio (Droite) par rapport à Tactile (95% IC).....	225
Figure 102. Temps de sélection (en ms) par type de plage et par technique.....	226
Figure 103. Score SUS par technique d'interaction (95% IC).....	227
Figure 104. Temps de sélection par technique d'interaction (en ms, 95% CI). ....	228
Figure 105. Temps d'affectation par technique d'interaction (en ms, 95% CI).....	229
Figure 106. Ratio de l'augmentation du temps d'affectation (en pourcentage, 95% CI, i.e. modification nécessaire sur aucune modification). ....	230
Figure 107. Pourcentage des essais dans lesquels au moins une erreur d'affectation est survenue (95% CI).....	230
Figure 108. Score SUS par technique d'interaction (95% CI).....	231

# LISTE DES TABLES

---

Table 1. Comparatif des degrés de libertés détectés par plusieurs souris augmentées (*Tz est détectée mais n'es pas utilisée). V signifie un contrôle de vitesse et P un contrôle de position. ....	73
Table 2. Dispositifs composés de la littérature classés selon les axes de DECO. ....	118
Table 3. Description des GlobeMouse, CubicMouse, PadMouse et CapMouse à l'aide de DECO. ....	121
Table 4. Itérations de conception dans DECO. ....	130
Table 5. Temps de sélection (en secondes) et intervalle de confiance par technique d'interaction. ....	165
Table 6. Tableau comparatif des avantages et inconvénients des solutions envisagées pour chaque tâche. ....	185
Table 7. Tableau comparatif du nombre de caractéristiques gérées et des inconvénients de chacune des barres de menu envisagées. ....	199
Table 8. Temps de sélection (en secondes) et intervalle de confiance par technique d'interaction. ....	225
Table 9. Ratio du temps de sélection et intervalle de confiance par technique d'interaction. ....	225
Table 10. Score SUS par technique d'interaction. ....	226
Table 11. Temps de sélection (en secondes) et intervalle de confiance par technique	228
Table 12. Temps d'affectation (en secondes) et intervalle de confiance par technique d'interaction. ....	229
Table 13. Score SUS par technique d'interaction. ....	231



Gary PERELMAN

## **Conception, development and evaluation of fluid interaction techniques for multidimensional environments: application on public services' software**

**Thesis Supervisors:**

Emmanuel Dubois

Marcos Serrano

### **Abstract:**

The work of this thesis is part of a collaboration with the company Berger-Levrault, a major actor in the development of administrative management software for public services. This work is based on two observations. On the first hand, the policy of digitization of public services induces the need for software adapted to the professions of all public institutions. These software are complex and particularly rich compared to classically used office software (Office, mailbox, etc.). On the other hand, we observe that the devices used to interact with these software did not evolve. Since several decades, the mouse and the keyboard remain the norm in a fixed environment. However, these devices have only few input degrees of freedom. The manipulation of multidimensional data with these devices induces a greater number of steps to perform a task, thus lengthening the interaction path.

In this context, the objective of these thesis work is to improve the interaction flow with multidimensional data contained in the software of the public service through the increase of the input degrees of freedom proposed by the devices. Indeed, a larger amount of input degrees of freedom would reduce the number of steps necessary to the accomplishment of a given task, thus improving the interaction flow. We propose three major contributions: a device with multiple degrees of freedom, the Roly-Poly Mouse; a design space, DECO; as well as a set of interaction techniques with mobile devices based on the principle of stacking.

A first contribution of our work is the design of a new device with multiple degrees of freedom: the Roly-Poly Mouse (RPM). This device, whose base is rounded, aims to replace the traditional mouse. It has 6 degrees of freedom (3 translations of which 2 exploited and 3 rotations). We evaluated its performance and compared it to other devices for a task requiring 6 degrees of freedom (3D object manipulation).

A second contribution of our work is the definition of a design space focusing on the physical aspect of the composition of devices: DECO. DECO relies on two axes: physical arrangement and physical manipulation. From this design space, we designed a compound device: the Roly-Poly Mouse 2, a device consisting of the combination of a Roly-Poly Mouse and a traditional mouse. We evaluated its performance and compared it to other devices through a RST task (Rotate-Scale-Translate, 5D task).

Finally, a third contribution of this work consists in the design of a solution that improve the interaction flow with multidimensional data in a mobile environment: the stacking. This approach consists in using a complementary device (a smartphone) to extend the input and output degrees of freedom of a support device (a tablet). We designed and evaluated several interaction techniques to improve the interaction with a spreadsheet on tablet. Through a set of studies, we show that stacking allows cell ranges to be selected and Pivot Tables to be created faster than with touch interaction.

### **Keywords:**

Human Computer Interaction, degrees of freedom, augmented mouse, design space, mobile devices

Discipline: Computer Science

Institut de recherche en informatique de Toulouse – UMR 5505

Université Paul Sabatier, 118 Route de Narbonne, 31 062 Toulouse Cedex 4, France

Gary PERELMAN

## **Conception, développement et évaluation de techniques d'interaction fluides pour des environnements multidimensionnels : application aux logiciels du service public**

**Directeurs de thèse :**

Emmanuel Dubois

Marcos Serrano

### **Résumé :**

Les travaux de cette thèse s'inscrivent dans une collaboration avec l'entreprise Berger-Levrault, acteur majeur du développement de logiciels de gestion administrative pour les services publics. Ces travaux s'appuient sur deux constats. D'une part, la politique de numérisation des services publics induit la nécessité de disposer de logiciels adaptés aux métiers de l'ensemble des établissements publics. Ces logiciels sont complexes et particulièrement riches comparativement aux logiciels de bureautique classiquement utilisés. D'autre part, on observe que les dispositifs utilisés pour interagir avec ces logiciels n'évoluent que très peu. Depuis plusieurs décennies, la souris et le clavier restent la norme dans un environnement fixe. Or, ces dispositifs ne possèdent que peu de degrés de libertés en entrée. La manipulation de données multidimensionnelles avec ceux-ci induit un plus grand nombre d'étapes pour réaliser une tâche, rallongeant ainsi le chemin d'interaction.

Dans ce contexte, l'objectif de ces travaux de thèse est de contribuer à la fluidification de l'interaction avec des données multidimensionnelles, contenues dans les logiciels du service public, au travers de l'augmentation des degrés de libertés en entrée proposés par les dispositifs. En effet, une plus grande quantité de degrés de libertés en entrée réduirait le nombre d'étapes nécessaires à la réalisation d'une tâche donnée, fluidifiant ainsi l'interaction. Nous proposons 3 contributions majeures : un dispositif à multiples degrés de libertés, la Roly-Poly Mouse ; un espace de conception, DECO ; ainsi qu'un ensemble de techniques d'interaction avec dispositifs mobiles basées sur le principe du stacking.

Une première contribution de nos travaux consiste en la conception d'un nouveau dispositif à multiples degrés de liberté : la Roly-Poly Mouse (RPM). Ce dispositif, dont la base est arrondie, vise à remplacer la souris traditionnelle. Il possède 6 degrés de libertés (3 translations dont 2 exploitées et 3 rotations). Nous avons évalué ses performances et l'avons comparé à d'autres dispositifs pour une tâche nécessitant 6 degrés de libertés (manipulation d'objet 3D).

Une seconde contribution de nos travaux consiste en la définition d'un espace de conception se focalisant sur l'aspect physique de la composition de dispositifs : DECO. DECO s'appuie sur deux axes : l'arrangement physique et la manipulation physique. À partir de cet espace de conception, nous avons conçu un dispositif composé : la Roly-Poly Mouse 2, un dispositif composé d'une Roly-Poly Mouse et d'une souris traditionnelle. Nous avons évalué ses performances et l'avons comparé à d'autres dispositifs au travers d'une tâche de RST (Rotate-Scale-Translate, tâche 5D).

Enfin, une troisième contribution de ces travaux consiste en la conception d'une solution fluidifiant l'interaction avec des données multidimensionnelles dans un environnement mobile : le stacking. Cette approche consiste en l'utilisation d'un dispositif complémentaire (un smartphone) pour étendre les degrés de libertés en entrée et en sortie d'un dispositif support (une tablette). Nous avons conçu et évalué plusieurs techniques d'interaction visant à fluidifier l'interaction avec un tableur sur tablette. Au travers d'un ensemble d'études, nous montrons que le stacking permet de sélectionner des plages de cellules et de créer des tableaux croisés dynamiques plus rapidement qu'avec l'interaction tactile.

### **Mots-clés :**

Interaction Homme-Machine, degrés de libertés, souris augmentées, espace de conception, dispositifs mobiles

Discipline : Informatique

Institut de recherche en informatique de Toulouse – UMR 5505

Université Paul Sabatier, 118 Route de Narbonne, 31 062 Toulouse Cedex 4, France