HAL
open science

Navigation multimodale dans une vue bifocale sur dispositifs mobiles

Sébastien Pelurson

To cite this version:

Sébastien Pelurson. Navigation multimodale dans une vue bifocale sur dispositifs mobiles. Ordinateur et société [cs.CY]. Université Grenoble Alpes, 2016. Français. NNT : 2016GREAM035 . tel-01689409

HAL Id: tel-01689409

https://theses.hal.science/tel-01689409v1

Submitted on 22 Jan 2018

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.

Communauté UNIVERSITÉ Grenoble Alpes

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Informatique**

Arrêté ministériel : 7 août 2006

Présentée par

Sébastien PELURSON

Thèse dirigée par **Laurence NIGAY**

préparée au sein du **Laboratoire d'Informatique de Grenoble, équipe Ingénierie de l'Interaction Homme-Machine**
dans **l'École Doctorale Mathématiques, Sciences et Technologies de l'Information, Informatique**

Navigation multimodale dans une vue bifocale sur dispositifs mobiles

Thèse soutenue publiquement le **2 Septembre 2016,**
devant le jury composé de :

M. Emmanuel DUBOIS
Professeur à l'Université de Toulouse (Président)
M. Géry CASIEZ
Professeur à l'Université de Lille 1 (Rapporteur)
M. Martin HACHET
Chargé de recherche à l'INRIA Bordeaux (Rapporteur)
M. Julien CASTET
Directeur de recherche à Immersion (Examinateur)
M. Eric LECOLINET
Maître de conférences à Télécom ParisTech (Examinateur)
Mme Laurence NIGAY
Professeur à l'Université Grenoble Alpes (Directrice de thèse)

REMERCIEMENTS

Je remercie tout d'abord les membres du jury d'avoir accepté d'évaluer mes travaux de thèse, et de s'être déplacés pour ma soutenance.

Ensuite, je remercie l'équipe IIHM et la société Streamdata.io de m'avoir accueilli durant cette période. Cette double expérience m'a permis de découvrir à la fois le milieu académique et le milieu industriel, et a donc été très enrichissante.

Je veux également remercier Laurence, sans qui rien de tout cela ne serait arrivé., puisque c'est elle qui m'a encouragé à me lancer dans cette aventure. Son soutien et son implication ont été un énorme atout dans l'accomplissement de ces travaux.

Enfin, je voudrais remercier mes amis et ma famille. Tous ont été un réel soutien durant ces travaux, me permettant de m'évader des préoccupations professionnelles lorsque c'était nécessaire. Je remercie plus particulièrement ma femme et mon fils. Ma femme pour m'avoir supporté et soutenu durant ces années de travail, malgrè les soirées (ou nuits) passées sur l'ordinateur à travailler. Et mon fils, arrivé durant les derniers mois de l'aventure, pour m'avoir sans cesse donné le sourire malgrè les courtes nuits qu'il a pu me faire passer.

TABLE DES MATIERES

Remerciements ii

Table des matières III

Liste des tables VI

Listes des figures VII

1 Introduction 1

1.1 Constat 1
1.1.1 Evolution des dispositifs mobiles 1
1.1.2 Evolution de l'usage 2
1.1.3 Sujet 2
1.1.4 Exemple d'application 3

1.2 Démarche de recherche 4

1.3 Contributions 5
1.3.1 Espaces conceptuels 5
1.3.2 Conception et évaluation de techniques de visualisation et de navigation 6

1.4 Organisation du manuscrit 7

2 Visualisation sur dispositifs mobiles 9

1.1 Introduction 9

2.1 Types de données visualisées sur dispositifs mobiles 10
2.1.1 Données à une dimension 10
2.1.2 Données temporelles 11
2.1.3 Données à deux dimensions 11
2.1.4 Données à trois dimensions 12
2.1.5 Données multidimensionnelles 13
2.1.6 Arbres 14
2.1.7 Réseaux 15
2.1.8 Synthèse 15

2.2 Techniques de visualisation sur dispositifs mobiles 16
2.2.1 Notion de multiplexage 17
2.2.2 Techniques reposant sur le multiplexage temporel 18
2.2.2.1 Barres de défilement 18
2.2.2.2 Déplacement & Zoom 19
2.2.2.3 Filtrage 21
2.2.3 Techniques utilisant le multiplexage spatial 22
2.2.3.1 Indices contextuels 22
2.2.3.2 Vue Globale+Détails 23
2.2.3.3 Focus+Contexte 24
2.2.4 Techniques utilisant le multiplexage en profondeur 26
2.2.5 Synthèse 27

2.3 Vue Focus+Contexte bifocale conçue sur dispositifs mobiles 30
2.3.1 Vue bifocale conçue 30
2.3.2 Expérience 32
2.3.2.1 Vues comparées 32
2.3.2.2 Participants et tâches 34
2.3.2.3 Procédure 36
2.3.2.4 Résultats et discussions 36

2.4 Conclusion **38**

3 Interaction avec une vue bifocale sur dispositifs mobiles : modèle conceptuel de navigation multimodale **40**

3.1 Introduction **40**

3.2 Interaction sur dispositifs mobiles : contraintes **41**
3.2.1 Taille d'écran réduite 41
3.2.1.1 Occultation 41
3.2.1.2 Accessibilité 42
3.2.2 Modalités d'interaction en entrée limitées 44
3.2.3 Modalités d'interaction dépendantes du contexte d'utilisation 44
3.2.4 Synthèse 45

3.3 Modalités de navigation sur dispositifs mobiles **45**
3.3.1 Grille d'analyse des modalités de navigation 46
3.3.2 Modalités tactiles 48
3.3.3 Modalités reposant sur les capteurs du dispositif 50
3.3.3.1 Modalités utilisant les déplacements du dispositif 50
3.3.3.2 Modalités exploitant l'inclinaison du dispositif 51
3.3.3.3 Modalités utilisant des capteurs de pression 56
3.3.3.4 Modalités autour du dispositif 58
3.3.4 Synthèse 60

3.4 Navigation avec une vue bifocale **61**

3.5 Modèle conceptuel de navigation dans une vue bifocale **62**

3.6 Conclusion **65**

4 Navigation multimodale dans une vue bifocale **67**

4.1 Introduction **67**

4.2 Etude pilote et Expérience 1 : bien-fondé du modèle conceptuel en trois phases **68**
4.2.1 Tâche de navigation 68
4.2.2 Etude pilote 69
4.2.2.1 Tilt 70
4.2.2.2 Interaction physique 72
4.2.2.3 Procédure et résultats 74
4.2.3 Expérience 1 75
4.2.3.1 Participants, tâche et procédure 76
4.2.3.2 Résultats 77
4.2.4 Conclusion 80

4.3 Expérience 2 ..... **80**
4.3.1 Contexte 81
4.3.2 Modalités comparées 81
4.3.2.1 FastDrag et DirectTouch 82
4.3.2.2 Tilt 83
4.3.2.3 ContinuousPress et DiscretePress 84
4.3.2.4 Peephole 85
4.3.3 Tâche de navigation 86
4.3.4 Procédure 87
4.3.5 Hypothèses 88
4.3.6 Résultats 88
4.3.6.1 Performances : temps d'exécution 88
4.3.6.2 Préférences utilisateur 94

4.4 Conclusion **97**

5 Navigation avec des capteurs de pression dans une vue bifocale 99

5.1 Introduction 99

5.2 Interaction bimanuelle et interaction par pression 100

5.3 Matériel 102

5.4 Options de conception des modalités basées sur la pression 103

5.5 Modalités basées sur le capteur de pression implémentées 104

5.5.1 Modalités *Continuous* 106
5.5.2 Modalité ContinuousInertia 107
5.5.3 Modalité *Discrete* 107
5.5.4 Modalité DiscreteInertia 107

5.6 Expérience 107

5.6.1 Modalités tactiles utilisées 108
5.6.2 Tâche expérimentale 109
5.6.3 Participants et procédure 110
5.6.4 Analyse du temps d'exécution par phase 111
5.6.5 Résultats 112
5.6.5.1 Performances : temps d'exécution 112
5.6.5.2 Préférences des utilisateurs 114
5.6.6 Discussion 115
5.6.6.1 Pression vs Tactile 115
5.6.6.2 Actions Continues vs Discrètes 116
5.6.6.3 Inertie vs Sans inertie 117
5.6.6.4 Combinaison des facteurs d'action et d'inertie 118

5.7 Conclusion 118

6 Conclusion 120

6.1 Démarche scientifique et contributions 120

6.1.1 Espaces conceptuels 121
6.1.2 Conception et évaluation de techniques de visualisation et d'interaction 122

6.2 Perspectives 122

6.2.1 Perspectives à court et moyen terme 122
6.2.1.1 Capteurs de pression avec des actions discrètes 123
6.2.1.2 Evaluation en contexte 123
6.2.1.3 Compléter l'espace conceptuel de navigation multimodale : étude de l'inertie 123
6.2.1.4 Unifier les deux espaces de conception 124
6.2.2 Perspectives à plus long terme 124
6.2.2.1 Incitation au changement de modalités 124
6.2.2.2 Données dynamiques 125

7 Publications personnelles 127

8 Bibliographie 128

9 Annexes 138

9.1 Annexe 1 – Tâches réalisées durant l'expérience de visualisation 138

9.2 Annexe 2 – Questionnaire SUS 139

9.3 Annexe 3 – Linéarisation des capteurs de pression avec un amplificateur opérationnel 140

LISTE DES TABLES

TABLE 2.1 SYNTHESE : AVANTAGES ET INCONVENIENTS DES DIFFERENTES TECHNIQUES DE VISUALISATION SUR DISPOSITIFS MOBILES.....29
TABLE 4.1 – 6 MODALITES POUR LA PHASE DE NAVIGATION RAPIDE. 81
TABLE 5.1 – 6 MODALITES POUR LA PHASE DE NAVIGATION RAPIDE. 108

LISTES DES FIGURES

FIGURE 1.1 EVOLUTION DES DISPOSITIFS MOBILES DEPUIS 1990. 2
FIGURE 1.2 PERIPHERIQUES D'ENTREE ET DE SORTIE DIFFERENTS. 3
FIGURE 1.3 APPLICATION BOURSE D'APPLE : (A) VALEURS DE LA JOURNEE EN COURS. (B) HISTORIQUE DE VALEURS SUR 10 ANS. 4
FIGURE 1.4 ORGANISATION DU MANUSCRIT. 7
FIGURE 2.1 (A) VISUALISATIONS DE DONNEES A UNE DIMENSION : (A) DOCUMENT PDF, (B) SPIRALIST, ISSUE DE [57] 10
FIGURE 2.2 VISUALISATIONS DE DONNEES TEMPORELLES : (A) DOSSIER MEDICAL, (B) DONNEES BOURSIERES. RESPECTIVEMENT ISSUES DE [28] ET [47]. 11
FIGURE 2.3 APPLICATION GOOGLE MAP SUR ANDROID. 12
FIGURE 2.4 VISUALISATIONS DE DONNEES A TROIS DIMENSIONS : (A) REALITE AUGMENTEE, (B) MODELE 3D D'UNE VILLE. RESPECTIVEMENT ISSUES DE [113] ET [18]. 13
FIGURE 2.5 APPLICATION SCATTERDICE MOBILE. ISSUE DE [125]. 13
FIGURE 2.6 (A) APPLICATION GENEANET SUR ANDROID, (B) TECHNIQUE RELT, ISSUE DE [53] 14
FIGURE 2.7 RECTANGULAR VIEW. ISSUE DE [65]. 15
FIGURE 2.8 - DIFFERENTS TYPES DE MULTIPLEXAGE : (A) TEMPOREL (ZONEZOOM), (B) SPATIAL (VUE GLOBALE+DETAILS), (C) EN PROFONDEUR (MINIMAP). RESPECTIVEMENT ISSUES DE [105], [19] ET [107]. 17
FIGURE 2.9 – PINCH (A) POUR ZOOMER, DRAG (B) POUR SE DEPLACER, FLICK (C) POUR SE DEPLACER RAPIDEMENT 19
FIGURE 2.10 – ZOOM GEOMETRIQUE 20
FIGURE 2.11 – ZOOM SEMANTIQUE : (A) CARTES. (B) TREEMAP, ISSUE DE [35] 21
FIGURE 2.12 – FILTRAGE (APPLICATION AGENDA SUR ANDROID) : (A) UN JOUR, (B) UNE SEMAINE, (C) UN MOIS 22
FIGURE 2.13 – INDICES CONTEXTUELS : (A) HALO, (B) WEDGE. RESPECTIVEMENT ISSUES DE [10] ET [50]. 22
FIGURE 2.14 – VUE GLOBALE+DETAILS : (A) VUE GLOBALE DE PETITE TAILLE RECOUVRANT LA VUE DETAILLEE, (B) VUE GLOBALE NE RECOUVRANT PAS LA VUE DETAILLEE, (C) VUE GLOBALE REDIMENSIONNABLE. ISSUES DE [48] 23
FIGURE 2.15 – VUES FOCUS+CONTEXTE : (A) DATELENS, (B) NAVIGATEUR WEST. RESPECTIVEMENT ISSUES DE [13] ET [15] 25
FIGURE 2.16 – METRO MAPS. ISSUE DE [130]. 26
FIGURE 2.17 – MULTIPLEXAGE EN PROFONDEUR (SCATTERDICE MOBILE). ISSUE DE [125]. 26
FIGURE 2.18 – APPLICATION WIKITUDE. 27
FIGURE 2.19 – (A) VUE BIFOCALE, (B) FONCTION DE ZOOM DE LA VUE BIFOCALE. 30
FIGURE 2.20 – VUE BIFOCALE IMPLEMENTEE. 30
FIGURE 2.21 – DEUX METAPHORES D'INTERACTION : (A) DEPLACER L'ESPACE D'INFORMATION (METAPHORE D'INTERACTION DU RUBAN) (B) DEPLACER LA ZONE DE FOCUS (METAPHORE D'INTERACTION DE LA LOUPE). 31
FIGURE 2.22 – QUATRE VUES COMPAREES. (A) ENTIRE = VUE GLOBALE AVEC UN CURSEUR (TRAIT MIS EN EVIDENCE EN BLEU) POUR ACCEDER A LA VALEUR D'UNE DONNEE AFFICHEE EN HAUT DE L'ECRAN (B) FOCUS = VUE DETAILLEE AFFICHANT LES VALEURS DE 20 DONNEES (C) BIFOCAL-SMALL = VUE BIFOCALE AVEC UN FOCUS DE 500 PIXELS DE LARGEUR (D) BIFOCAL-LARGE = VUE BIFOCALE AVEC UN FOCUS DE 800 PIXELS DE LARGEUR. 33
FIGURE 2.23 – (A) LES SCORES SUS MOYENS AVEC UN INTERVALLE DE CONFIANCE CORRESPONDANT DE 95 % (B) CLASSEMENT PAR PREFERENCES ET PAR TYPE DE TACHES DES VUES. 36
FIGURE 2.24 – REDIMENSIONNEMENT DU FOCUS AVEC UN GESTE DE *PINCH*. 37
FIGURE 3.1 – (A) OCCULTATION DE L'ECRAN DU DISPOSITIF MOBILE PAR LE POUCE, (B) « FAT FINGER » [118] : OCCULTATION DES TOUCHES DU CLAVIER NUMERIQUE. 42
FIGURE 3.2 – ERREUR DE PERCEPTION DE LA POSITION DU POINTEUR SOUS LE POUCE : (A) POSITION PERÇUE PAR L'UTILISATEUR, (B) POSITION CAPTEE PAR LE SYSTEME. ISSUES DE [108]. 42
FIGURE 3.3 – ZONES DIFFICILEMENT ACCESSIBLES (ZONES SOMBRES) AVEC LE POUCE DE LA MAIN QUI TIENT LE DISPOSITIF MOBILE DANS LE CAS D'UN DROITIER. ISSUE DE [66]. 43
FIGURE 3.4 – DIFFERENTES AMPLITUDES POSSIBLES DES MOUVEMENTS DU POIGNET : (A) FLEXION/EXTENSION (B) PRONATION/SUPINATION (C) INCLINAISON ULNAIRE/ INCLINAISON RADIALE. ISSUES DE [102] 44
FIGURE 3.5 – DEUX MODES DE NAVIGATION DIFFERENTS (A) RELATIF. (B) ABSOLU. 47
FIGURE 3.6 – (A) GESTE DE *PINCH*. (B) MACHINE A ETAT DES GESTES DE *DRAG* ET *FLICK*. 49
FIGURE 3.7 – CLUTCHING. ISSUE DE [101]. 49
FIGURE 3.8 – NAVIGATION UTILISANT LES DEPLACEMENTS DU DISPOSITIF : (A) CHAMELEON. (B) BOOM CHAMELEON. RESPECTIVEMENT ISSUES DE [42] ET [126]. 50
FIGURE 3.9 – AXES DE ROTATION DU MOBILE. 51

FIGURE 3.10 – RECONNAISSANCE DE GESTES A PARTIR DE L'INCLINAISON DU DISPOSITIF POUR LE CONTROLE D'UN LECTEUR MUSICAL. ISSUE DE [6]. 52
FIGURE 3.11 – NAVIGATION PAR INCLINAISON DU DISPOSITIF SELON L'AXE X : (A) PAR POSITION. (B) PAR VITESSE. 53
FIGURE 3.12 – SDAZ : (A) VUE STATIQUE, (B) DEFILEMENT LENT, (C) DEFILEMENT RAPIDE, (D) DEFILEMENT TRES RAPIDE. ISSUES DE [59]. 54
FIGURE 3.13 – NAVIGATION PAR INCLINAISON DU DISPOSITIF : LA VITESSE DE NAVIGATION EST CALCULEE PAR RAPPORT A L'ANGLE D'INCLINAISON ET LE NIVEAU DE ZOOM EST CONTROLE A L'ECRAN PAR L'INTERACTION TACTILE. ISSUE DE [72]. 55
FIGURE 3.14 – (A) RESISTANCE A DETECTION DE FORCE (B) PROTOTYPE DE CLAVIER AVEC CAPTEURS DE PRESSION, ISSUE DE [31]. 56
FIGURE 3.15 – SIDEPRESS : TECHNIQUE UTILISANT DEUX CAPTEURS DE PRESSION POUR NAVIGUER DANS UN ESPACE A UNE DIMENSION. ISSUES DE [119]. 57
FIGURE 3.16 – HOVERFLOW : INTERACTION AUTOUR DU DISPOSITIF GRACE A DES CAPTEURS INFRAROUGES. ISSUES DE [73]. 58
FIGURE 3.17 – INTERACTION AUTOUR DU DISPOSITIF GRACE A UN SYSTEME DE SUIVI PAR CAMERA. ISSUES DE [60]. 58
FIGURE 3.18 – INTERACTION AUTOUR DU DISPOSITIF GRACE A UN CAPTEUR MAGNETIQUE SOUS FORME DE BAGUE. ISSUES DE [70]. 59
FIGURE 3.19 – (A) FONCTION DE ZOOM D'UNE VUE EN ŒIL DE POISSON (B) FONCTION DE ZOOM DE LA VUE BIFOCALE. 61
FIGURE 3.20 – MODALITE DE NAVIGATION DANS UNE VUE BIFOCALE : DES EVENEMENTS CAPTES AUX DEPLACEMENTS DANS LA VUE BIFOCALE. 63
FIGURE 3.21 – MODELE CONCEPTUEL DE NAVIGATION MULTIMODALE EN TROIS PHASES. 64
FIGURE 3.22 – RELATIONS TEMPORELLES ENTRE LES DEUX MODALITES DE NAVIGATION CARACTERISANT LA PHASE DE TRANSITION DE LA FIGURE 3.21. 65
FIGURE 3.23 – ETUDE DE L'INTERACTION AVEC UNE VUE BIFOCALE SUR DISPOSITIFS MOBILES : DEMARCHE D'ANALYSE RELATEE PAR LA STRUCTURE DE CE CHAPITRE. 66
FIGURE 4.1 – (A) DEBUT DE LA TACHE : LA CIBLE EST INDIQUEE DANS LE CONTEXTE PAR UN TRAIT BLEU. LA FLECHE INDIQUE LA DIRECTION DE LA POSITION DE LA CIBLE. (B) FIN DE LA TACHE : LA CIBLE EST POSITIONNEE DANS LA ZONE DESIGNEE DU FOCUS. LA ZONE EST ALORS AFFICHEE EN VERT. 69
FIGURE 4.2 – (A) MODALITE *TILT*. (B) AXES DE ROTATION DU DISPOSITIF. 70
FIGURE 4.3 – MODALITE *INTERACTION PHYSIQUE*. 72
FIGURE 4.4 – MODALITE PAR DEPLACEMENTS DU DISPOSITIF : FONCTIONS D'ASSOCIATION : (A) ASSOCIATION LINEAIRE (B) ASSOCIATION SPHERIQUE (C) ASSOCIATION DYNAMIQUE. ISSUES DE [97]. 74
FIGURE 4.5 – TEMPS MOYENS D'EXECUTION PAR MODALITE ET ID AVEC INTERVALLE DE CONFIANCE DE 95%....77
FIGURE 4.6 – (A) LES SCORES SUS MOYENS AVEC UN INTERVALLE DE CONFIANCE CORRESPONDANT DE 95% (B) CLASSEMENT PAR PREFERENCE DES TROIS MODALITES. 78
FIGURE 4.7 – EFFET D'AGRANDISSEMENT : QUAND LA LOUPE SE DEPLACE VERS LE HAUT, L'IMAGE AGRANDIE SE DEPLACE VERS LE BAS. ISSUE DE [51]. 82
FIGURE 4.8 – PROTOTYPE BASE SUR DES CAPTEURS DE PRESSION....84
FIGURE 4.9 – MODALITE *PEEPHOLE*. 85
FIGURE 4.10 – TACHE: (A) DEBUT DE LA TACHE: LA FLECHE INDIQUE LA DIRECTION DE LA CIBLE. (B) TACHE EN COURS : LA CIBLE EST DANS LE FOCUS. (C) FIN DE LA TACHE: LE CURSEUR SE TROUVE SUR LA CIBLE. 87
FIGURE 4.11 – (A) TEMPS MOYENS D'EXECUTION POUR LA PHASE DE NAVIGATION RAPIDE (B) TEMPS MOYENS D'EXECUTION POUR LA PHASE DE NAVIGATION PRECICE (C) TEMPS MOYENS D'EXECUTION POUR LA PHASE DE TRANSITION (D) TEMPS MOYENS D'EXECUTION POUR LA TACHE GLOBALE AVEC UN INTERVALLE DE CONFIANCE CORRESPONDANT DE 95%. 91
FIGURE 4.12 – (A) SCORE SUS MOYEN AVEC UN INTERVALLE DE CONFIANCE DE 95% (B) CLASSEMENT PAR PREFERENCE AVEC LE NOMBRE FOIS OU CHAQUE MODALITE COMBINEE A ETE CLASSEE 1ERE, 2E OU 3E. 94
FIGURE 4.13 – FASTDRAG (A) A PARTIR DU BAS DE L'ECRAN : OCCULTATION LIMITEE (B) A PARTIR DU COTE DE L'ECRAN : OCCULTATION IMPORTANTE....95
FIGURE 4.14 – DISCRETEPRESS (A) POUCES SOUS LE DISPOSITIF: APPUIS LEGERS ET FORTS DIFFICILES A DIFFERENCIER (B) POUCES SUR L'ECRAN: APPUIS FORTS DIFFICILES A EFFECTUER....95
FIGURE 4.15 – AMBIGUÏTE SUR LE SENS DE DEFILEMENT LORSQUE LA POSITION DU FOCUS N'EST PAS FIXE : (A) LE FOCUS SE DEPLACE VERS LA GAUCHE, LA CIBLE SEMBLE SE DEPLACER VERS LA DROITE. (B) LE GESTE DE *DRAG/FLICK* EST EFFECTUEE VERS LA DROITE, LA CIBLE SE DEPLACER VERS LA DROITE....96
FIGURE 5.1 – PROTOTYPE CONÇU : CAPTEUR DE PRESSION SUR LE PANNEAU AVANT DU DISPOSITIF. LA MAIN NON DOMINANTE QUI TIENT LE DISPOSITIF EST DEDIEE A LA MODALITE BASEE SUR LA PRESSION AVEC LE POUCE. LA MAIN DOMINANTE EST DEDIEE A L'INTERACTION TACTILE. 100

FIGURE 5.2 – PROTOTYPE CONÇU : LE CAPTEUR DE PRESSION UTILISE EST CELUI PLACE SUR LE PANNEAU AVANT DU DISPOSITIF. 102
FIGURE 5.3 – EXEMPLES DE CHEMINS D'INTERACTION. 105
FIGURE 5.4 – LES QUATRES MODALITES BASEES SUR LA PRESSION ET LEUR COMBINAISON AVEC LA MODALITE *FLICK*. 106
FIGURE 5.5 – TACHE EXPERIMENTALE: LES FLECHES INDIQUENT LE SENS DE DEFILEMENT. (A) LA CIBLE EST DANS LE COTE GAUCHE DE LA ZONE DE CONTEXTE. (B) LA CIBLE EST DANS LA ZONE DE FOCUS, A DROITE DU CURSEUR AFFICHE AU CENTRE (LIGNE NOIRE). 109
FIGURE 5.6 – (A) TEMPS MOYENS POUR LA PHASE DE NAVIGATION RAPIDE. (B) TEMPS MOYENS POUR LA PHASE DE NAVIGATION PRECISE. (C) TEMPS MOYENS POUR LA PHASE DE TRANSITION. (D) TEMPS MOYENS POUR LA TACHE GOBALE AVEC UN INTERVALLE DE CONFIANCE CORRESPONDANT DE 95%. 112
FIGURE 5.7 – (A) SCORES MOYENS SUS AVEC DES INTERVALLES DE CONFIANCE A 95% (B) CLASSEMENT PAR ORDRE DE PREFERENCE ET NOMBRE DE FOIS OU LES MODALITES ONT ETE CLASSEES 1ER, 2EME OU 3EME. 115
FIGURE 5.8 – MODALITE *FLICK*: LES POSTURES DE LA MAIN DE L'UTILISATEUR (A) AVEC UN DOIGT ET (B) AVEC DEUX DOIGTS. 116
FIGURE 5.9 – TOUPIE A POMPE ET FRICTION. 119
FIGURE 6.1 – DEMARCHE SCIENTIFIQUE ET CONTRIBUTIONS. 121
FIGURE 9.1 – SCHEMA DU CONVERTISSEUR COURANT/TENSION UTILISE POUR LINEARISER LES VALEURS RETOURNEES PAR LES CAPTEURS DE PRESSION. 140

1 INTRODUCTION

1.1 Constat

1.1.1 Evolution des dispositifs mobiles

Depuis 2014, le nombre d'utilisateurs de dispositifs mobiles (smartphone, tablette, montre connectée, etc.) a dépassé le nombre d'utilisateurs d'ordinateurs, et cette différence continue à s'accentuer. En janvier 2016, le monde entier compte plus de 50% d'utilisateurs de dispositifs mobiles, et ce chiffre est en constante augmentation (+4% par rapport à janvier 2015).
Ces dispositifs prennent donc une part très importante dans l'utilisation des services numériques et tendent à remplacer les ordinateurs de bureaux et les ordinateurs portables. En effet, tout comme les capacités de connexion maintenant proches de celles disponibles sur les ordinateurs de bureau (3G, 4G), les évolutions technologiques des dispositifs mobiles font qu'ils ont aujourd'hui des performances comparables à celles des ordinateurs standards (résolution d'écran, processeur, mémoire vive, etc.). De plus, l'apparition dans les dispositifs mobiles de nouveaux capteurs (gyroscope, accéléromètre, ou plus récemment capteur de pression) offrent de nouvelles possibilités d'interaction à exploiter.
Malgré ces évolutions technologiques, la taille des dispositifs mobiles n'a que peu évolué au cours des années, et celle-ci tend à se stabiliser depuis peu (Figure 1.1).

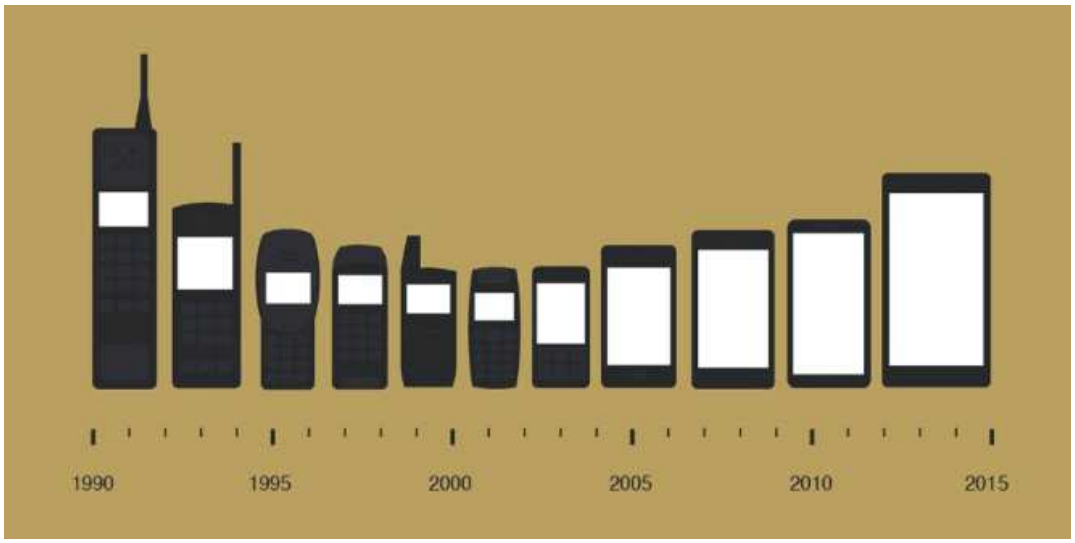


Figure 1.1 Evolution des dispositifs mobiles depuis 1990[1].

1.1.2 Evolution de l'usage

Couplée à ces évolutions technologiques, l'apparition du Cloud Computing et du Big Data a complètement modifié les habitudes d'utilisation des dispositifs mobiles. Alors qu'autrefois ils servaient uniquement à téléphoner, les dispositifs mobiles sont aujourd'hui de véritables ordinateurs connectés en permanence à Internet. Les utilisateurs ont donc la possibilité d'accéder à leurs données n'importe où et n'importe quand. Ainsi des tâches effectuées autrefois uniquement sur les ordinateurs de bureau ou les ordinateurs portables le sont aujourd'hui majoritairement sur dispositifs mobiles : consultation de mails, visualisation de photos, visite d'une boutique en ligne, jeux vidéo, etc.
La société Cisco prévoit ainsi un trafic mondial de données sur dispositifs mobiles de 291.6 exaoctets par an en 2019 contre 50.4 exaoctets pour l'année 2015. Face à une diffusion instantanée et ubiquitaire d'une très grande quantité d'information l'enjeu est de pouvoir la gérer sur les dispositifs mobiles.

1.1.3 Sujet

Aussi le sujet des travaux de cette thèse est la visualisation interactive de grandes quantités de données sur un dispositif mobile pour aider les utilisateurs à appréhender une grande quantité d'information.
Malgré les évolutions technologiques décrites ci-dessus, les dispositifs mobiles ne s'utilisent pas de la même manière que les ordinateurs de bureau ou les ordinateurs portables. En particulier, pour l'interaction homme-machine, les périphériques d'entrée et de sortie sont différents. L'écran, par exemple, bien que de qualité comparable, reste beaucoup plus petit sur un dispositif mobile contraint par le contexte de mobilité à respecter. La surface d'affichage des informations est donc fortement restreinte sur ce type de dispositif (Figure 1.2).

---

[1] http://www.dailyinfographic.com/mobile-phone-size-evolution
[2] http://streamdata.io

Figure 1.2 Périphériques d'entrée et de sortie différents.

De plus, alors que l'interaction graphique avec des ordinateurs de bureau ou portables repose principalement sur l'interaction indirecte avec l'utilisation d'un clavier et d'une souris, les dispositifs mobiles peuvent utiliser soit l'interaction directe à l'aide d'un écran tactile, soit l'interaction indirecte à l'aide d'un ensemble de capteurs embarqués (gyroscope, accéléromètre, caméra, etc.). Ceci implique d'utiliser des modalités d'interaction en entrée et des degrés d'indirection différents pour une même tâche, selon qu'elle est effectuée sur dispositifs mobiles ou sur un ordinateur de bureau ou portable. En particulier, les écrans tactiles sur dispositifs mobiles étant utilisés à la fois comme périphérique d'entrée et de sortie, ceci entraîne des limitations comme l'occultation d'une partie de l'écran lors de l'interaction tactile.
Enfin, le contexte de mobilité implique une utilisation particulière des dispositifs mobiles. Par exemple marcher dans la rue en portant un sac imposera une utilisation à une main ou même une utilisation sans regarder l'écran (interaction *eyes-free*), permettant ainsi aux utilisateurs de décharger ou libérer le canal de perception visuelle pour rester concentré sur l'environnement qui l'entoure.

Ces spécificités inhérentes aux capacités interactionnelles et au contexte d'usage mobile des dispositifs mobiles imposent donc de concevoir des techniques de visualisation et d'interaction adaptées. C'est l'enjeu de ces travaux de thèse.

1.1.4 Exemple d'application

Nous illustrons le sujet de nos travaux avec une application : la visualisation de graphes d'indices boursiers sur dispositifs mobiles (Figure 1.3). Ce type d'applications est en lien direct avec la société Streamdata.io[2] qui fût le contexte de réalisation de ces travaux de thèse. En effet, la société traitait initialement avec de nombreux clients bancaires dont l'un des besoins récurrents de leurs applications sur dispositifs mobiles était la visualisation de graphes d'indices boursiers.

[2] http://streamdata.io

(a) **(b)**

Figure 1.3 Application Bourse d'Apple : (a) Valeurs de la journée en cours. (b) Historique de valeurs sur 10 ans.

Ce besoin nécessite de visualiser des historiques de valeurs pouvant aller jusqu'à 15 ans en arrière, ce qui représente une quantité d'information considérable. Alors que sur ordinateur de bureau les tailles d'écran permettent de visualiser relativement aisément des quantités d'information de cette ampleur, la tâche est beaucoup plus complexe sur un dispositif mobile disposant d'une taille d'écran réduite (Figure 1.2). De plus, l'interaction à la souris utilisée sur les ordinateurs de bureau permet une interaction très précise permettant d'accéder à des valeurs en les pointant ou les sélectionnant avec le curseur. L'interaction tactile majoritairement utilisée sur dispositifs mobiles est beaucoup moins précise et rend une fois de plus la tâche plus complexe que sur un ordinateur de bureau.

1.2 Démarche de recherche

Le sujet est d'importance et concerne principalement deux grands axes de recherche en Interaction Homme-Machine : la visualisation interactive d'information et l'interaction sur dispositifs mobiles.

La visualisation interactive d'information définit un axe de recherche dont l'objet d'étude est la représentation visuelle et interactive de données. Cet axe correspond à une communauté très active disposant d'une plateforme communautaire[3]. Le lieu principal de publications des contributions est la conférence IEEE Information Visualization (InfoVis).
L'interaction sur dispositifs mobiles, un axe plus récent, comprend également une communauté très active. Cet axe de recherche s'est établi au sein de l'Interaction Homme-Machine et a donné lieu en 1998 à la création d'une conférence ACM dédiée : MobileHCI.

A la croisée de ces deux axes de recherche très actifs, notre démarche de recherche a consisté à étudier :
les techniques de visualisation présentant à la fois une vue globale de l'espace d'information et une vue détaillée d'une sous-partie de celui-ci : Comme illustré à la Figure 1.2, visualiser uniquement une sous-partie de l'espace d'information rend

[3] http://www.infovis-wiki.net

sa compréhension difficile du fait du manque de contexte. A l'inverse, visualiser l'espace d'information complet à l'écran le rend illisible. Afin d'être efficace, une technique de visualisation conçue sur dispositifs mobiles doit donc permettre de visualiser à la fois une partie détaillée de l'espace d'information, ainsi qu'une vue globale de celui-ci permettant de connaître le contexte de la partie détaillée.
les techniques d'interaction pour naviguer dans l'espace d'information : La variété des capteurs disponibles aujourd'hui sur les dispositifs mobiles définissent un espace de possibilités important en termes de modalités d'interaction. De plus, les techniques de visualisation à plusieurs niveaux de détail incitent à considérer plusieurs modalités pour naviguer, chaque modalité étant adaptée à un niveau de détail. Notre démarche a donc été d'étudier les modalités de navigation et leurs combinaisons. Cette combinaison de modalités enrichit un espace des possibilités déjà très vaste.

Pour étudier les techniques de visualisation présentant une vue globale de l'espace d'information ainsi qu'une sous-partie détaillée de celui-ci, la démarche de travail a consisté à considérer les techniques de multiplexage. Le concept de multiplexage, bien établi en visualisation d'information, fait référence à la manière dont plusieurs vues sont liées : de façon spatiale, temporelle ou en profondeur.
Une approche expérimentale a permis ensuite de mettre au point une technique de visualisation à deux niveaux de détail sur dispositifs mobiles.

Pour étudier les techniques de navigation dans la visualisation conçue, notre démarche a été de prendre en compte d'une part les contraintes liées aux dispositifs mobiles et d'autre part les besoins de navigation multi-échelle dans la visualisation. A partir de ces contraintes et besoins interactionnels, nous avons adopté une approche reposant sur la combinaison de modalités : la multimodalité.

1.3 Contributions

Nous présentons ci-après les contributions de cette thèse qui a été réalisée au sein du laboratoire LIG (Laboratoire d'Informatique de Grenoble) dans l'équipe IIHM (Ingénierie de l'Interaction Homme-Machine) et la société Streamdata.io. Cette société propose principalement une plateforme logicielle permettant de récupérer efficacement des données en temps réel dans un contexte de mobilité contraint (perte de réseau, quantité limitée de données à transmettre). Particulièrement sollicitée par des établissements bancaires, la société Streamdata.io définit un contexte applicatif aux travaux : données et besoins utilisateurs.

1.3.1 Espaces conceptuels

Un premier espace de conception des modalités de navigation sur dispositifs mobiles permet de classer les modalités selon trois dimensions :

- le dispositif d'interaction utilisé (écran tactile, gyroscope, capteur de pression, etc.) ;
- le type d'action (discret ou continu), relatif au langage d'interaction utilisé ;
- le mode de navigation (relatif ou absolu).

Cet espace de conception trouve sa structure dans la définition d'une modalité d'interaction comme un dispositif d'interaction couplé à un langage d'interaction [95].

Nous proposons un deuxième modèle conceptuel de navigation qui est multimodal pour gérer l'aspect multi-échelle de la navigation. En effet les différents niveaux de zoom des espaces d'information multi-échelle requièrent deux besoins : un accès rapide aux données dans la zone non détaillée et une interaction précise avec les données dans la zone détaillée. Pour cela il est possible de soit :
Naviguer à l'échelle détaillée et ainsi avoir une navigation précise mais lente.
Naviguer à l'échelle non détaillée et ainsi avoir une navigation rapide mais imprécise.
Ces deux possibilités nous ont conduits à définir un modèle conceptuel de navigation basé sur deux modalités (une rapide et une précise), considérant également leur transition dans le but de les combiner efficacement.

1.3.2 Conception et évaluation de techniques de visualisation et de navigation

Nous avons conçu et évalué une technique de visualisation bifocale sur dispositifs mobiles. La technique conçue considère deux niveaux de détail : un pour la vue globale de l'espace d'information, et un pour la sous-partie détaillée de celui-ci. Cette technique utilise le mode de présentation paysage des dispositifs mobiles afin de profiter d'une largeur d'écran plus importante. Nous avons appliqué un effet de perspective à la vue globale de l'espace d'information afin que celle-ci puisse être visualisée de manière plus confortable sur un écran de petite taille de dispositifs mobiles. Ainsi, la technique de visualisation comporte deux espaces de travail distincts : un espace détaillé permettant d'effectuer des tâches précises, et un espace beaucoup moins détaillé permettant de présenter le contexte global de l'espace d'information.

De plus, basé sur le modèle conceptuel de navigation multimodale proposé, nous avons conçu et évalué plusieurs techniques de navigation multi-échelle multimodale :
Tilt+Flick utilise l'inclinaison du dispositif pour naviguer rapidement dans l'espace d'information et l'interaction tactile pour naviguer de manière précise.
Interaction physique+Flick utilise les déplacements du dispositif dans l'espace (interaction physique) pour naviguer rapidement dans l'espace d'information et l'interaction tactile pour naviguer de manière précise.
Pression+Flick utilise l'interaction avec des capteurs de pression pour naviguer rapidement dans l'espace d'information et l'interaction tactile pour naviguer de manière précise.
Pour chacune de ces techniques, nous avons exploré plusieurs solutions de conception en nous référant à notre espace de conception des modalités de navigation sur dispositifs mobiles. Nous avons ensuite évalué expérimentalement ces solutions de conception.

Les deux classes de contribution, espaces de conception et techniques de navigation, sont représentées dans le schéma de la Figure 1.4, en les positionnant par rapport à la structure du manuscrit.

1.4 Organisation du manuscrit

Au delà de l'introduction et conclusion, le manuscrit s'articule donc selon quatre chapitres représentés à la Figure 1.4.

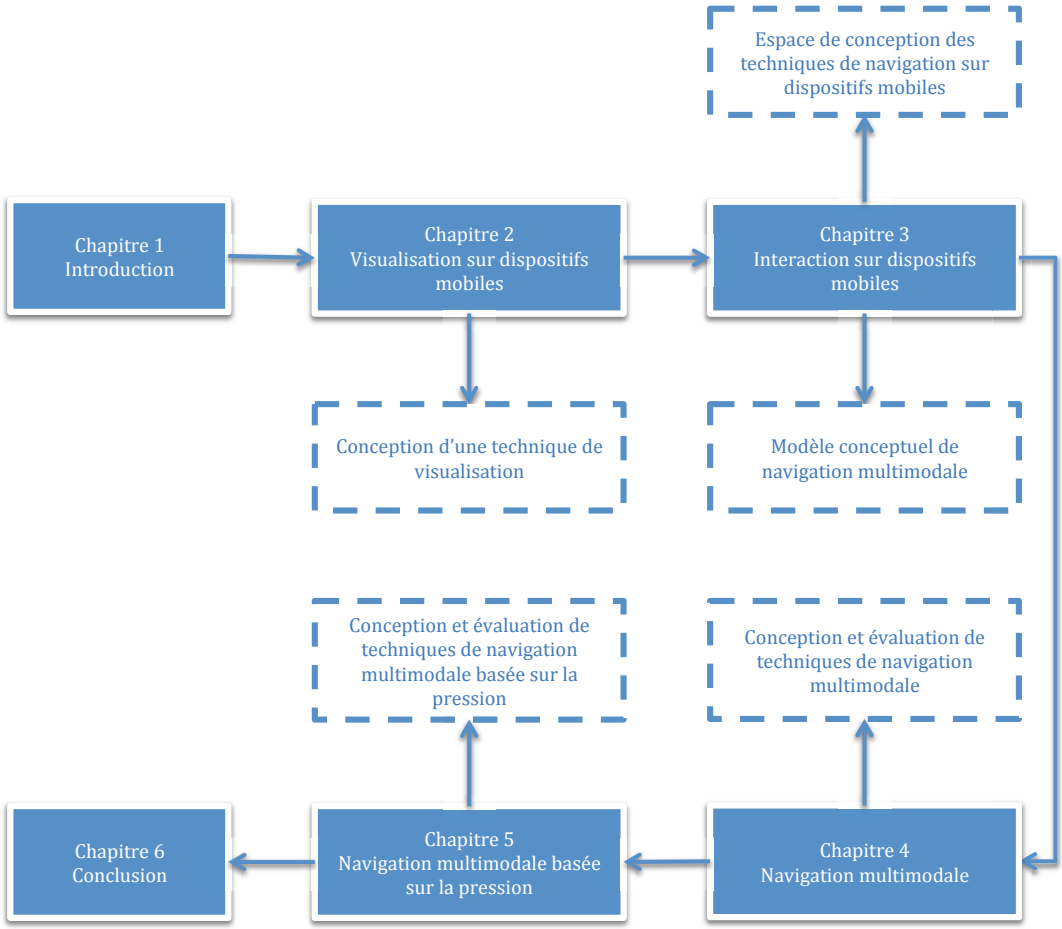


Figure 1.4 Organisation du manuscrit.

Dans le chapitre 2 nous présentons et classons les techniques de visualisation existantes sur dispositifs mobiles (section 2.1 et 2.2). Basée sur cet état de l'art nous présentons la technique de visualisation conçue ainsi que son évaluation (section 2.3). Reposant sur deux niveaux de détail (vue bifocale), elle permet de visualiser simultanément l'ensemble de l'espace d'information et une sous-partie détaillée de cet espace.

Dans le chapitre 3 nous présentons les contraintes d'interaction inhérentes aux dispositifs mobiles (section 3.2) et classons les techniques de navigation existantes sur dispositifs mobiles (section 3.3). Pour naviguer au sein de la vue bifocale conçue, nous définissons un modèle conceptuel en trois phases basé sur une approche multimodale (section 3.5).

Les chapitres 4 et 5 décrivent notre exploration, certes partielle, du vaste espace de possibilités en termes de modalités d'interaction et combinaisons de modalités que définit notre modèle conceptuel de navigation multimodale.

Aussi dans le chapitre 4, le rationnel de conception des 6 techniques conçues repose sur notre modèle de navigation et la taxonomie des modalités de navigation du chapitre 3. Nous présentons les résultats de deux évaluations expérimentales. La première compare *Tilt+Flick* et *Interaction physique+Flick* à une modalité n'utilisant qu'une seule modalité pure (au sens non combinée) tactile (section 4.2). La deuxième expérience compare *Tilt+Flick*, *Interaction physique+Flick* et *Pression+Flick* à deux techniques de navigation multimodale combinant deux modalités tactiles (section 4.3). Les deux modalités tactiles diffèrent donc par leurs langages d'interaction et partagent le même dispositif d'interaction (l'écran tactile).

Dans le chapitre 5, toujours en utilisant le modèle conceptuel de navigation du chapitre 3, nous étudions l'interaction bimanuelle et multimodale avec la vue bifocale conçue. Nous combinons pour cela un capteur de pression, associé à la main non dominante des utilisateurs et tenant le dispositif mobile, à l'interaction tactile avec la main dominante. Nous étudions l'influence de deux facteurs sur l'utilisation de la modalité basée sur la pression : le type d'action (discret ou continu) identifié dans notre espace de conception et l'inertie. Ceci a donné lieu à quatre techniques de navigation multimodale utilisant toutes un capteur de pression. Nous présentons les résultats d'une évaluation expérimentale comparant ces quatre techniques à deux techniques de navigation multimodale utilisant l'interaction tactile uniquement (section 5.6).

Enfin, nous concluons ce mémoire en rappelant nos contributions, et en proposant des pistes de recherche qui étendent les travaux présentés.

2 VISUALISATION SUR DISPOSITIFS MOBILES

1.1 Introduction

Ce chapitre est dédié à la visualisation d'information sur dispositifs mobiles. Cet axe de recherche a émergé suite à l'apparition des PDA (« Personal Digital Assistant ») [15], et a connu un essor considérable plus récemment avec les *smartphones*.

Les dispositifs mobiles actuels embarquent de nombreux capteurs (accéléromètre, gyroscope, GPS, etc.) et possèdent des écrans de très bonne qualité (par exemple l'écran du Samsung Galaxy S6 ayant une résolution de 2560x1440 pixels). Malgré ces progrès technologiques, le contexte de mobilité (comme l'interaction avec le pouce de la main qui tient le dispositif) et les limitations interactionnelles des dispositifs mobiles (comme la taille de l'écran) font que la réutilisation des techniques de visualisation conçues pour des ordinateurs de bureau n'est pas immédiate.
Au delà des caractéristiques techniques des dispositifs, les habitudes des utilisateurs ont également évolué. Ils manipulent aujourd'hui sur leurs dispositifs mobiles au moins autant de données que sur leurs ordinateurs de bureau. Ainsi des tâches effectuées autrefois uniquement sur les ordinateurs de bureau ou portables le sont aujourd'hui majoritairement sur dispositifs mobiles comme la consultation de mails, la visualisation de photos ou encore les jeux vidéo. Les utilisateurs ont donc besoin de manipuler tout type d'information.

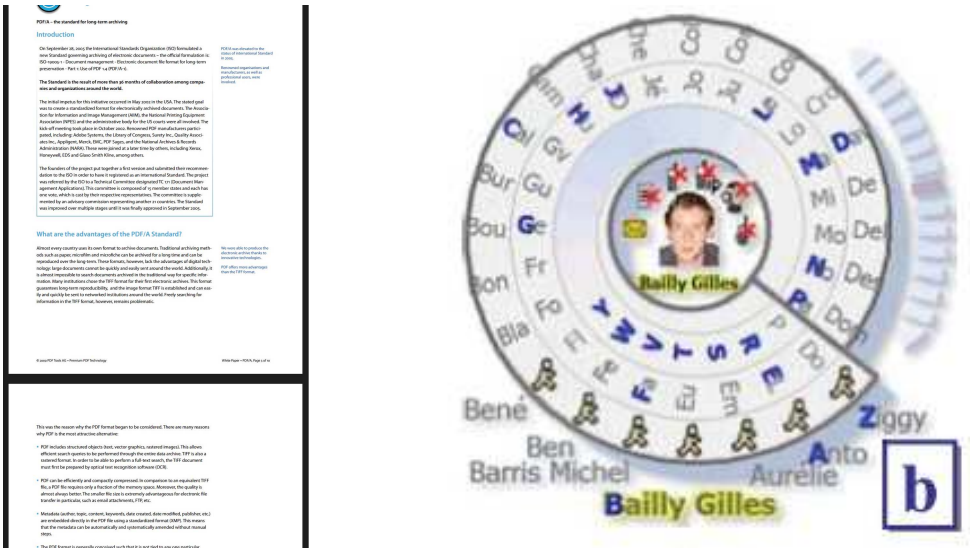
La première partie de ce chapitre présente les différents types de données qu'il est possible de visualiser sur dispositifs mobiles. Cette partie souligne en particulier la variété des techniques de visualisation selon les données manipulées. Elle motive aussi l'importance du sujet aujourd'hui en mettant en évidence la nécessité de

manipuler sur dispositifs mobiles des espaces d'information de types très différents.
La seconde partie dresse un état de l'art et classe les techniques de visualisation existantes sur dispositifs mobiles. Pour cela nous nous reposons sur la notion de multiplexage qui nous permet de traiter l'enjeu central de la visualisation d'un grand espace d'information sur un écran de taille réduite.
Ces deux premières parties permettent de motiver le type de données et de visualisation que nous avons choisi de traiter dans nos travaux. Aussi la dernière partie du chapitre présente la technique que nous avons implémentée et évaluée : une vue bifocale pour visualiser des données à une dimension et temporelles.

2.1 Types de données visualisées sur dispositifs mobiles

Comme c'est le cas sur les ordinateurs de bureau, les utilisateurs peuvent maintenant faire tout type de tâches sur leurs dispositifs mobiles et ainsi manipuler tout type de données (édition de texte, retouche photo, etc.). Shneiderman [117] présente une taxonomie en sept types de données relative à la visualisation d'information. Cette taxonomie étant dédiée aux données à visualiser, elle s'applique à la visualisation sur dispositifs mobiles. Nous l'illustrons dans la section en présentant les techniques existantes relatives à chacun des sept types.

2.1.1 Données à une dimension



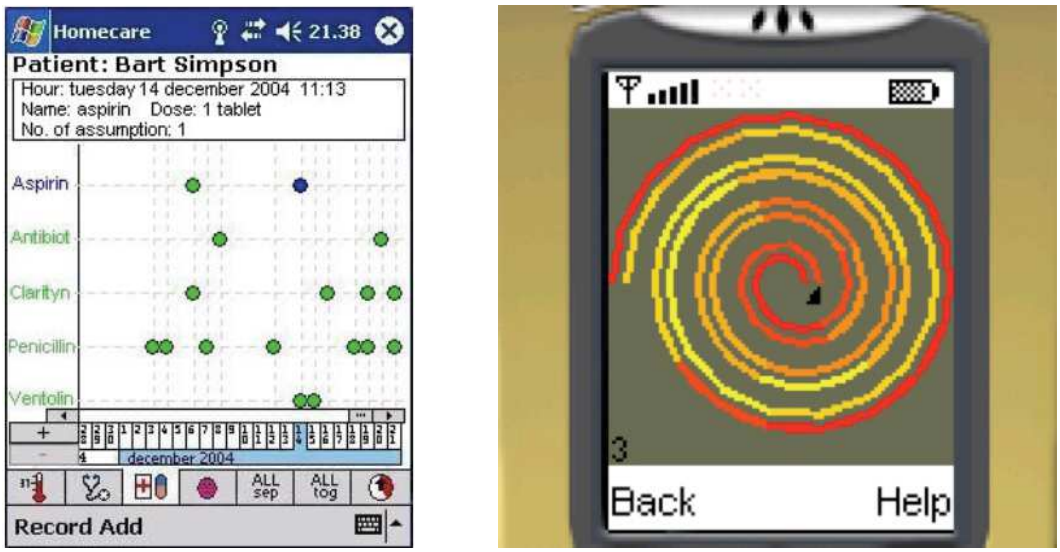
(a) (b)

Figure 2.1 (a) Visualisations de données à une dimension : (a) Document PDF, (b) SpiraList, issue de [57].

Les données à une dimension peuvent être un document texte (Figure 2.1.a), une liste de contacts ou encore une galerie de photos. Ces données sont organisées de manière séquentielle (selon un ordre alphabétique par exemple) et selon un axe unique de valeurs. Lire un livre électronique ou parcourir une liste de mails restent des tâches compliquées à accomplir sur dispositifs mobiles du fait de la taille de l'écran réduite. Plusieurs travaux tentent de répondre à ce problème, comme ceux

de Huot et Lecolinet [57] permettant de représenter une liste ordonnée alphabétiquement sous forme de spirale (Figure 2.1.b). Cette visualisation permet d'afficher entièrement la liste à l'écran.

2.1.2 Données temporelles



(a) (b)

Figure 2.2 Visualisations de données temporelles : (a) Dossier médical, (b) Données boursières. Respectivement issues de [28] et [47].

Les données temporelles représentent un cas particulier des données à une dimension, mais sont assez nombreuses pour être classées dans une catégorie à part entière. Elles sont organisées selon un axe temporel. Sur dispositifs mobiles, on les retrouve par exemple dans des applications permettant la visualisation de graphiques (historiques boursiers, nombre de pas effectués dans la journée, etc.) ou encore une liste de mails classés par date de réception. Un exemple d'application utilisant ce type de données est présenté par Chittaro [28] permettant de visualiser le dossier médical d'un patient sur dispositifs mobiles (Figure 2.2.a). Grégory & al. [47] ont proposé un affichage en spirale pour visualiser simultanément les valeurs de deux données boursières (Figure 2.2.b).

2.1.3 Données à deux dimensions

Les données à deux dimensions peuvent être positionnées dans un plan. Sur dispositifs mobiles, les applications les plus utilisées représentant des données à deux dimensions sont les cartes. Les applications cartographiques (Figure 2.3) sont devenues courantes sur dispositifs mobiles depuis que ceux-ci embarquent des capteurs GPS. Les données à représenter peuvent alors atteindre des tailles très importantes (carte du monde par exemple) et les visualiser constitue donc un véritable défi. De nombreuses études ont été menées sur la visualisation de cartes sur dispositifs mobiles [10, 20]. Au delà de l'affichage d'une carte à différents niveaux de détail, la visualisation de Baudisch & Rosenholtz [10] consiste à représenter des indications sur les points d'intérêts apparaissant en dehors de

l'écran. Dans l'étude de Burigat & Chittaro [20] différentes techniques de visualisation pour trois types de données dont les cartes (cartes, diagrammes et pages web) sont expérimentalement comparées. Ils ont notamment montré qu'une tâche de recherche dans une carte avec une interface zoomable présentant une vue globale de l'espace d'information était effectuée plus efficacement qu'avec une interface zoomable sans vue globale, alors qu'il n'y a pas de différence dans le cas d'un diagramme ou d'une page web.

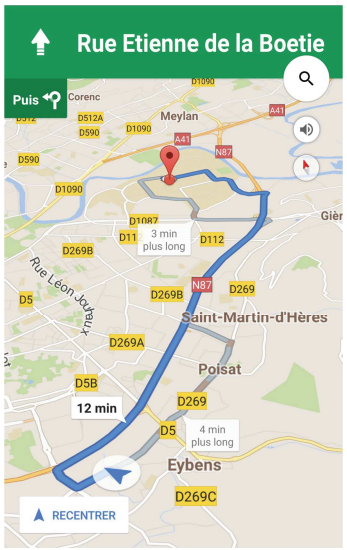


Figure 2.3 Application Google Map sur Android[4].

Les pages web font elles aussi partie de ce type de données à deux dimensions et ont donné lieu à plusieurs travaux comme la technique de Baudisch & al. [11] qui présente un navigateur web permettant de supprimer du contenu jugé non pertinent. Elles permet pour cela aux utilisateurs de faire disparaître des zones de l'interface jugées non pertinentes en utilisant des gestes avec un stylet sur l'écran tactile. Les utilisateurs peuvent alors également utiliser des gestes pour agrandir certaines zones pour profiter de l'espace ainsi libéré.

2.1.4 Données à trois dimensions

Les données à trois dimensions sont représentées selon les trois axes x, y et z. Elles intègrent donc une notion de profondeur. Les progrès techniques réalisés sur les dispositifs mobiles font qu'elles sont de plus en plus présentes : les performances actuels des dispositifs mobiles (processeur, mémoire vive, etc.) permettent de représenter ces types de données. Ce type de données est souvent utilisé pour représenter des objets de la vie réelle, comme notamment les jeux vidéo, les applications de construction [82], ou encore les applications de réalité augmentée [113] (Figure 2.4.a). Burigat & Chittaro [18] proposent une application mobile permettant d'afficher et manipuler en temps réel le modèle en trois dimensions d'une ville (Figure 2.4.b).

[4] https://play.google.com/store/apps/details?id=com.google.android.apps.maps&hl=fr

(a) (b)

Figure 2.4 Visualisations de données à trois dimensions : (a) Réalité augmentée, (b) Modèle 3D d'une ville. Respectivement issues de [113] et [18].

2.1.5 Données multidimensionnelles

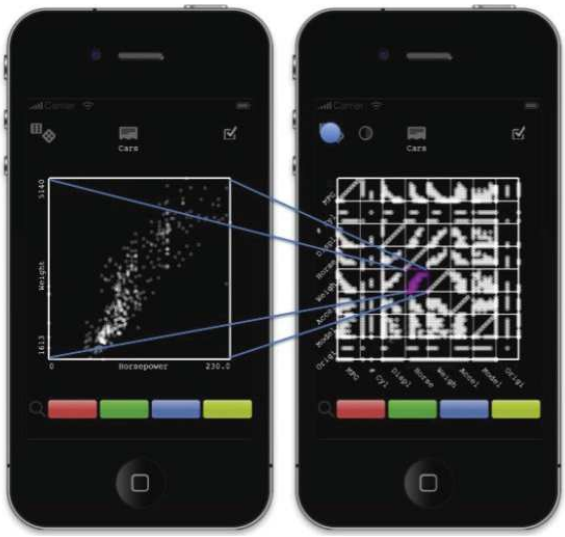
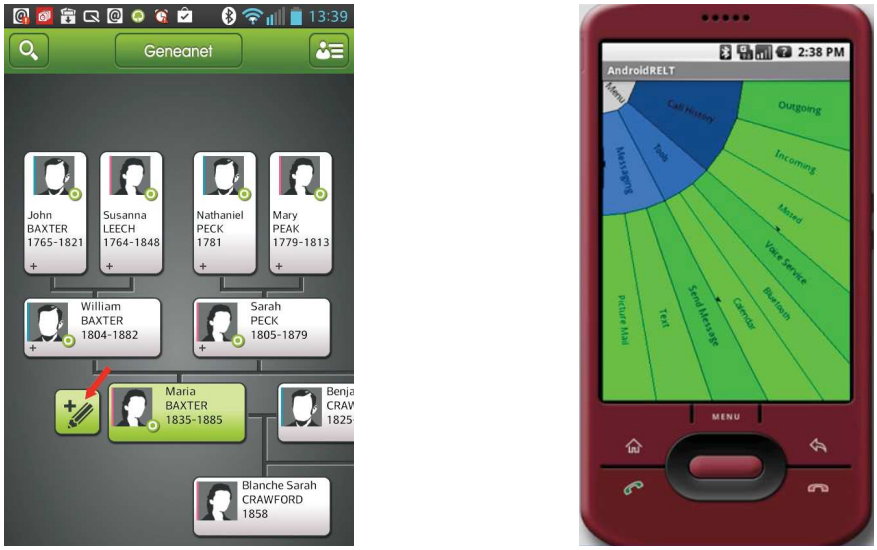


Figure 2.5 Application ScatterDice Mobile. Issue de [125].

Pour ce type de données, à chaque point de l'espace correspond au moins trois attributs. Il s'agit par exemple de visualiser les tuples issus d'une requête de base de données, ou encore une collection de musiques (où chaque élément contient un titre, un artiste, une année de sortie, une durée, etc.). Thomason & Wang [125] ont par exemple développé *ScatterDice Mobile* (Figure 2.5) : cette technique permet de visualiser des données multidimensionnelles : un jeu de données comportant 12 dimensions issu d'une capture vidéo, et un deuxième comportant 8 dimensions issu de données automobiles. ScatterDice Mobile utilise le gyroscope des dispositifs mobiles pour explorer une matrice de graphiques. La Figure 2.5 présente la matrice de graphiques (droite) sur laquelle apparaît en violet la région actuellement sélectionnée. L'utilisateur a alors la possibilité de zoomer sur celle-ci

pour la visualiser plus en détail comme illustré sur l'image de gauche de la Figure 2.5.

2.1.6 Arbres



(a) (b)

Figure 2.6 (a) Application Geneanet sur Android[5], (b) Technique RELT, issue de [53].

Les arbres sont des structures hiérarchisées où chaque élément possède un lien vers son élément parent. On retrouve une illustration de ce type de données avec l'application *Geneanet* disponible sur les différentes plateformes mobiles et permettant de visualiser un arbre généalogique (Figure 2.6.a). De même, Hao et al. [53] décrivent *RELT*, une technique adaptable et paramétrable permettant de visualiser ce type de données pour différents domaines applicatifs (Figure 2.6.b). Son adaptabilité permet de l'utiliser pour différents domaines applicatifs, la seule contrainte étant que les données soient hiérarchisées. Les utilisateurs ont la possibilité de paramétrer la technique en spécifiant par exemple la profondeur de l'arbre à afficher à l'écran (2 sur la Figure 2.6.b), sa racine étant représentée dans le coin supérieur gauche. Khella & Bederson [71] présentent *Pocket PhotoMesa*, une application permettant de visualiser une collection de photos hierarchisées par dossier. Engdahl & al. [35] utilisent un *treemap* pour visualiser les fils de discussion d'un forum.

[5] http://www.geneanet.org

2.1.7 Réseaux

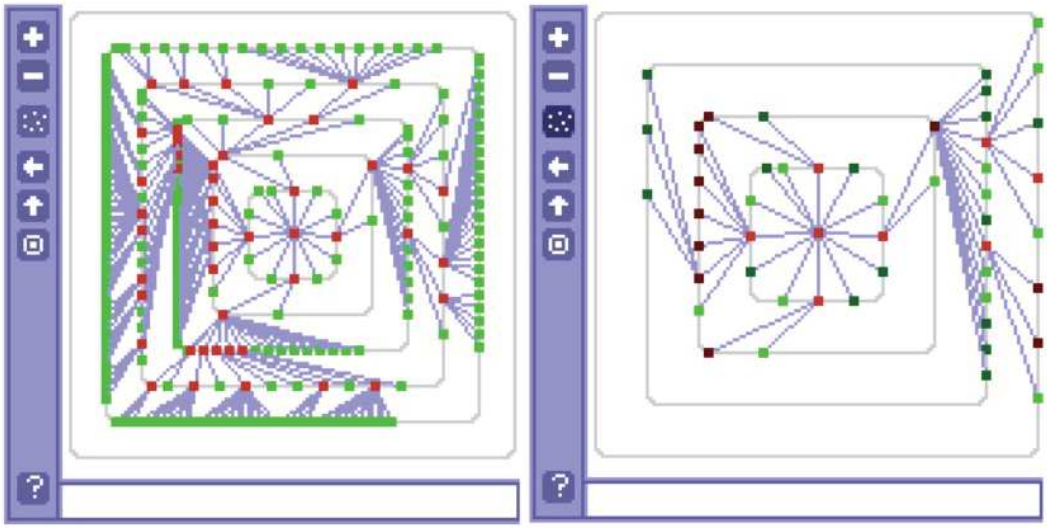


Figure 2.7 Rectangular View. Issue de [65].

Les réseaux sont eux aussi des structures hiérarchisées, mais contrairement aux arbres les liens vers un élément ne proviennent par forcément d'un élément parent, mais d'un ou plusieurs éléments quelconque de l'espace d'information. Un grand nombre d'espaces d'information peuvent être représentés sous cette forme, comme les liens hypertextes ou encore les réseaux sociaux. Karstens et al. [65] présentent *Rectangular View*, une technique permettant de visualiser et parcourir des réseaux de nœuds génériques à différents niveaux de détail (Figure 2.7). Trois couleurs sont utilisées pour représenter les types de nœuds (vert pour les feuilles, rouge pour les nœuds ayant des fils, et brunâtre pour les sous-réseaux). Les sous-réseaux peuvent ainsi être dépliés ou non pour modifier le niveau de détail. Par exemple l'image de gauche de la Figure 2.7 présente une vue complètement dépliée de l'espace d'information, alors que celle de droite est beaucoup moins détaillée.

2.1.8 Synthèse

Nous avons montré par des exemples que les sept types de données de la taxonomie de Schneiderman [117] sont aujourd'hui visualisés sur dispositifs mobiles. Ces types de données sont des types de base dont il existe des variantes (données à 4 dimensions, arbres multiples, etc.). De plus, de nombreuses applications ont besoin de manipuler plusieurs de ces types : l'application Geneanet gère par exemple un arbre de données multidimensionnelles (informations sur les personnes).

Les arbres et les réseaux, que l'on peut regrouper dans une catégorie de données hiérarchisées, ont déjà été largement étudiés sur dispositifs mobiles [38, 52, 53, 65], de même que les données à deux dimensions (et notamment les cartes) [72, 97, 105, 130].
Les données à trois dimensions, bien que manipulées sur dispositifs mobiles, ne nécessitent pas les mêmes besoins interactionnels que sur un ordinateur de

bureau. En effet, les tâches effectuées sur les deux types de dispositifs sont souvent différentes, et le besoin de détail est moins important sur dispositifs mobiles. De plus, les objets en trois dimensions de grande taille nécessitent une puissance de calcul pour être manipulés, qui n'est pas encore disponible sur les dispositifs mobiles actuels. Ces types de données majoritairement utilisés sur les ordinateurs de bureau aujourd'hui pourront donc très certainement également être visualisés sur dispositifs mobiles demain.

Ce constat sur les types de données visualisées nous a incité à concentrer nos travaux sur le type de données à une dimension et temporelles. De plus, ce type de données est présent dans de très nombreuses applications mobiles aujourd'hui : liste de mails, galerie d'images, graphes, etc.

2.2 Techniques de visualisation sur dispositifs mobiles

Plusieurs taxonomies ont été proposées pour classer les techniques de visualisation.

Une première façon immédiate de classer les techniques de visualisation est de considérer les types de données visualisées en se référant à la taxonomie des types de données de Schneiderman [117]. C'est l'approche adoptée dans la section précédente.

La taxonomie des techniques de visualisation de Chittaro [29] sépare les techniques adaptées de celles conçues pour des ordinateurs de bureau, des techniques conçues spécifiquement pour dispositifs mobiles. Il propose ainsi 4 catégories de techniques :

- Vue Globale+Détails (adaptées de celles conçues pour des ordinateurs de bureau)
- Focus+Contexte (adaptées de celles conçues pour des ordinateurs de bureau)
- Références visuelles à des zones hors écran (spécifiques aux dispositifs mobiles)
- Techniques de navigation intuitives (spécifiques aux dispositifs mobiles)

Si les trois premières catégories semblent pertinentes, la dernière fait plutôt référence aux moyens de naviguer dans un type de visualisation en utilisant des modalités propres aux dispositifs mobiles. Or, il nous semble plus pertinent de distinguer l'aspect présentation de l'aspect interaction. En effet, même si une technique de visualisation est inexistante sans technique d'interaction associée, nous nous intéressons dans cette section uniquement à l'aspect présentation des données. Nous adoptons donc une autre approche de classification décrite ci-dessous.

Orthogonalement aux types de données visualisées, un enjeu central à la visualisation sur un dispositif mobile est la taille limitée de l'écran : le défi est de rendre accessible l'ensemble de l'espace d'information alors qu'il est plus large que l'espace disponible à l'écran. Ce problème n'est pas spécifique aux dispositifs mobiles et est classiquement traité en visualisation d'information sur un

ordinateur [128]. Néanmoins il est amplifié de par la taille de l'écran du dispositif mobile. Aussi, pour dresser un état de l'art des techniques de visualisation existantes de façon complémentaire à la section précédente, nous considérons ce problème de visualisation d'une grande quantité de données sur un écran de taille limitée en nous référant à la notion de multiplexage.

2.2.1 Notion de multiplexage

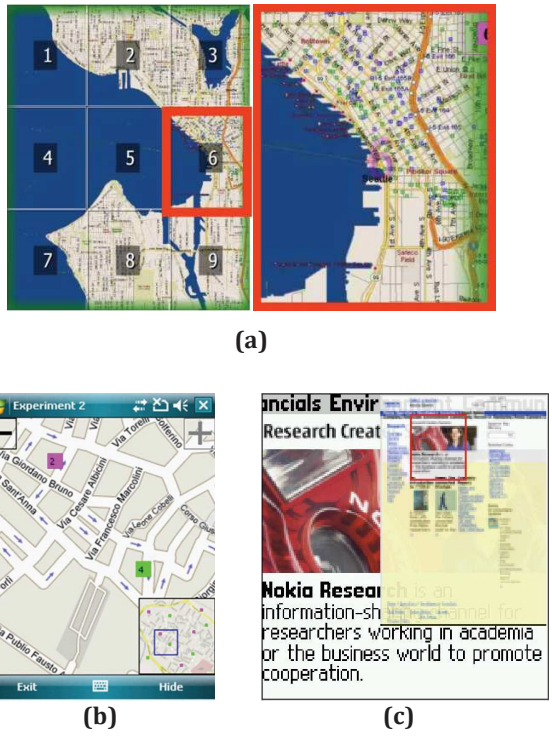


Figure 2.8 - Différents types de multiplexage : (a) Temporel (ZoneZoom), (b) Spatial (Vue Globale+Détails), (c) En profondeur (Minimap). Respectivement issues de [105], [19] et [107].

Lorsqu'une technique de visualisation affiche simultanément ou non plusieurs parties d'un espace d'information, elle utilise le multiplexage temporel, spatial, ou en profondeur pour les lier entre elles (Figure 2.8).

- Le multiplexage temporel : différentes parties de l'espace d'information sont présentées séquentiellement dans le temps. Les données affichées à un instant t peuvent être contrôlées par le système ou directement par l'utilisateur. Dans la Figure 2.8.a, la vue du deuxième écran correspond à un agrandissement de la zone encadrée en rouge dans le premier écran. Le principal inconvénient de ce type de multiplexage est que l'utilisateur doit se souvenir de ce qu'il a vu précédemment pour pouvoir faire le lien avec ce qui sera affiché ensuite.
- Le multiplexage spatial : différentes parties de l'espace d'information sont présentées simultanément. Dans la Figure 2.8.b, la vue globale de l'espace d'information est affichée dans le coin inférieur droit, sur le même écran que la vue zoomée. Le principal inconvénient de ce type de multiplexage est l'espace occupé à l'écran par les deux vues.

- Le multiplexage en profondeur : différentes parties de l'espace d'information sont superposées à l'écran. Dans la Figure 2.8.c, la vue globale est affichée au dessus de la vue zoomée en utilisant un effet de transparence. Le principal inconvénient de ce type de multiplexage est qu'il peut engendrer une surcharge d'information, rendant la visualisation difficilement perceptible à l'écran.

En considérant une visualisation comme une modalité d'interaction en sortie, les types de multiplexage définissent des relations spatiales et temporelles entre modalités dans l'espace de caractérisation de Vernier & Nigay [127]. Comme dans la taxonomie de Pook [100] pour le cas de la visualisation sur un ordinateur de bureau, nous utilisons la notion de multiplexage pour classer les techniques de visualisation sur dispositifs mobiles.

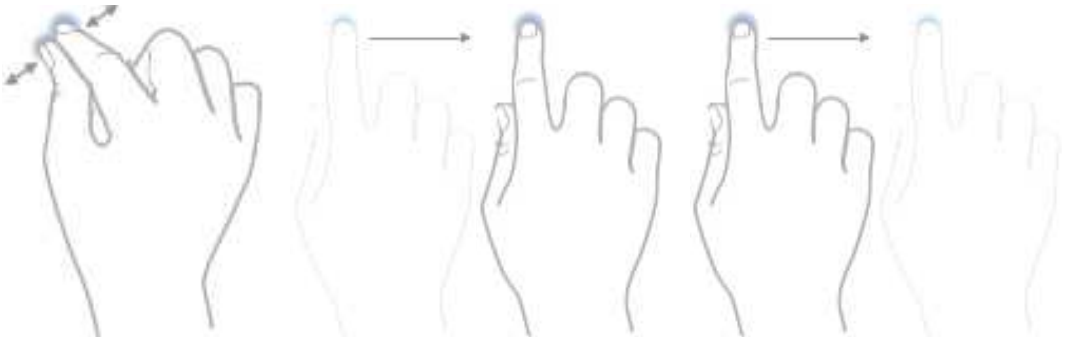
2.2.2 Techniques reposant sur le multiplexage temporel

Les techniques utilisant le multiplexage temporel nécessitent toutes un moyen de navigation pour parcourir l'espace d'information qui n'est pas entièrement affiché à l'écran et pour définir la partie de l'espace d'information qui a besoin d'être détaillée pour l'accomplissement de la tâche. Nous présentons dans la partie suivante ces moyens de navigation dans le cas de multiplexage temporel : les barres de défilement, les techniques de Déplacement & Zoom et les techniques de filtrage. Ces deux dernières actions de navigation sont mises en évidence dans le mantra énoncé par Shneiderman [117] : *"Overview first, zoom and filter, then details-on-demand ".*

2.2.2.1 Barres de défilement

Une première solution consiste, comme sur les ordinateurs de bureau, à utiliser des barres de défilement. Cela permet, alors que les utilisateurs ne peuvent visualiser qu'une petite partie des données à la fois, de parcourir l'ensemble de l'espace d'information. Or, Jones et al. [61] ont montré que l'utilisation d'une barre de défilement sur des écrans de petite taille peut réduire de 50% les performances par rapport à leur utilisation sur un écran d'ordinateur de bureau. De même, Laarni [74] a comparé la performance des barres de défilement pour différentes tailles d'écran. Il a observé que les performances augmentent avec la taille de l'écran. Ces deux résultats s'expliquent par le fait que plus l'écran est petit, plus les utilisateurs doivent effectuer de gestes de défilement. Aussi, ces gestes répétés peuvent être difficiles à effectuer surtout en condition de mobilité, et dégradent les performances en augmentant la longueur de la trajectoire d'interaction (nombre d'actions effectuées par l'utilisateur).
Enfin, le fait de pouvoir visualiser uniquement qu'une sous-partie de l'espace d'information à un instant donné réduit grandement la compréhension globale de celui-ci. En effet, pour une exploration efficace d'un espace d'information, les utilisateurs ont besoin d'accéder à la fois à des données détaillées, et à des données plus grossières mais globales, comme rappelé par Chen [26].

2.2.2.2 Déplacement & Zoom



(a) (b) (c)

Figure 2.9 – Pinch (a) pour zoomer, Drag (b) pour se déplacer, Flick (c) pour se déplacer rapidement

Les techniques de Déplacement & Zoom consistent à ajouter une troisième dimension aux barres de défilement. En plus de pouvoir se déplacer dans l'espace d'information, les utilisateurs ont également la possibilité de zoomer, et ainsi de modifier le niveau de détail des données représentées à l'écran. Spence [120] définit le fait de zoomer comme « augmenter/diminuer la taille d'une partie d'une image à deux dimensions tout en ayant une fenêtre d'affichage de taille constante ». Il définit le déplacement comme « un mouvement fluide de la fenêtre d'affichage sur une image à deux dimensions de plus grande taille ».
Par exemple, Robbins et al. [105] ont utilisé cette stratégie pour développer *ZoneZoom*, une technique permettant de naviguer à travers un espace d'information à deux dimensions en utilisant le clavier du dispositif mobile. L'espace est divisé en 9 sous-espaces, chacun associé à une touche du clavier du dispositif mobile (Figure 2.8.a). Ainsi, l'état initial présente une vue globale de l'espace d'information, et l'utilisateur peut presser une des touches du clavier pour zoomer et se déplacer sur le sous-espace associé qui est alors à son tour découpé en 9 sous-espaces. Il peut à tout moment revenir à la vue précédente en pressant la touche #. Ce même principe a aussi été appliqué pour le cas de pages web par Baudisch & al. [11].
Devenu un standard avec les écrans tactiles, la technique du *Pinch-Drag-Flick* (Figure 2.9) permet de se déplacer et zoomer dans un espace d'information. Le geste de *Pinch* (Figure 2.9.a) permet de zoomer et dézoomer en écartant ou rapprochant deux doigts à l'écran. Le geste de *Drag* (Figure 2.9.b) permet de se déplacer lentement en maintenant et faisant glisser un doigt à l'écran. Le geste de *Flick* (Figure 2.9.c) permet de se déplacer rapidement en effectuant un balayage rapide du doigt à l'écran. Cette combinaison de gestes permet donc de gérer le niveau de zoom tout en se déplaçant dans l'espace d'information. Contrairement à *ZoneZoom* [105], cette technique permet de gérer le zoom et le déplacement de manière plus fluide car ils sont effectués au fur et à mesure que le geste associé est effectué à l'écran : par exemple, un déplacement de un pixel à l'écran déplace un pixel dans l'espace d'information, alors que dans *ZoneZoom*, la selection d'une tôuche entraîne un déplacement beaucoup plus important dans l'espace d'information. De plus, le déplacement peut être fait manuellement par l'utilisateur (alors qu'il est fait automatiquement par le système dans *ZoneZoom*).

Orthogonalement aux actions effectuées lors de Déplacement & Zoom (actions de pression sur des boutons ou gestes sur la surface tactile), nous distinguons deux types de zoom : le zoom géométrique et le zoom sémantique.

2.2.2.2.1 Zoom géométrique

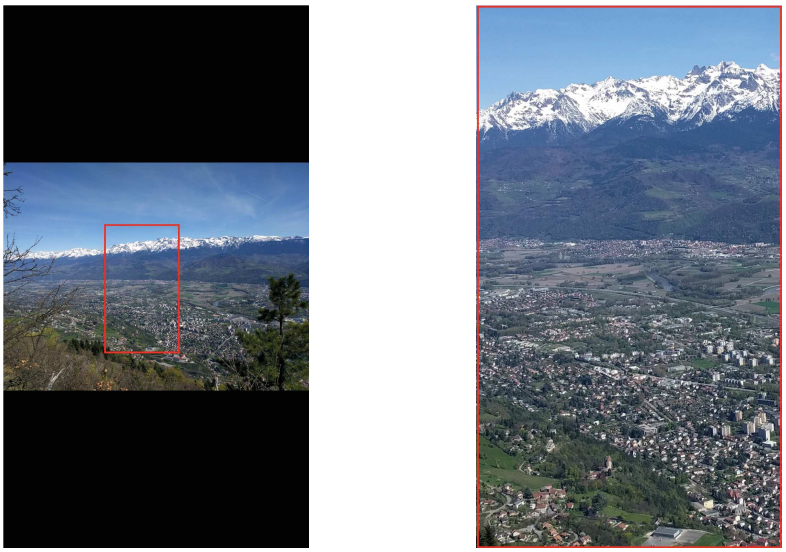
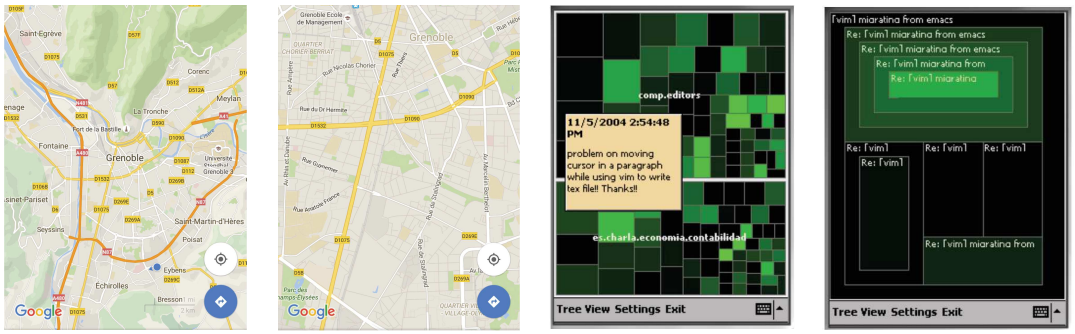


Figure 2.10 – Zoom géométrique

Le zoom géométrique n'agit que sur la taille de l'objet à représenter. Comme illustré à la Figure 2.10, il agrandit les objets lorsque l'utilisateur zoome, et les rétrécit lorsqu'il dézoome. L'avantage est la cohérence des données, mais le principal inconvénient est le manque de lisibilité quand trop d'informations sont affichées à l'écran. On le retrouve souvent dans les lecteurs de PDF ou les galeries photos par exemple (Figure 2.10). Autrefois utilisé par les applications de cartographie, il a depuis été remplacé par le zoom sémantique par soucis de lisibilité des données. Ce principe a été utilisé par Khella & Bederson [71] pour développer *PhotoMesa*, une application facilitant la visualisation d'une galerie d'images classées par dossier sur PDA. Les utilisateurs peuvent naviguer à travers la hiérarchie d'images : un zoom géométrique est appliqué modifiant la résolution des images miniatures affichées pour chaque niveau de la hiérarchie.

2.2.2.2.2 Zoom sémantique

Le zoom sémantique agit sur la représentation de l'objet à afficher. Ainsi, un même objet pourra être représenté de différentes manières, voire même être supprimé, selon le niveau de zoom (Figure 2.11). Les différentes représentations permettent de maintenir la lisibilité des données, mais il est important que celles-ci soient cohérentes entre elles. En effet, les différentes représentations des objets doivent conserver une même sémantique : par exemple sur une carte, une autoroute devra toujours apparaître plus importante qu'un chemin.



(a) (b)

Figure 2.11 – Zoom sémantique : (a) Cartes. (b) Treemap, issue de [35].

Cette technique de zoom sémantique peut s'appliquer à tout type de données. Ainsi au delà des cartes, cette technique a par exemple été utilisée par Lam & Baudisch [75] dans un navigateur web pour PDA qui affiche les sites web comme un ensemble de vues miniatures. Pour afficher l'ensemble de la page à l'écran, les différents champs textes sont abrégés, ce qui permet de conserver leur lisibilité tout en occupant moins de place à l'écran. Lorsque l'utilisateur zoome, la version originale (non abrégée) du texte est représentée. Le même principe a été appliqué par Karlson & al. [67] pour la visualisation d'une matrice de 36 applications. Des boutons circulaires sont disposés aux intersections de quatre applications, permettant de zoomer sur celles-ci. Les quatre applications sont alors représentées seules à l'écran et peuvent à nouveau être sélectionnées individuellement. L'apparence des applications change en fonction du niveau de zoom : icônes ou applications directement manipulables. Ce principe a également été appliqué par Engdhal & al. [35] pour la visualisation d'un *treemap* représentant les discussions d'un forum. Les utilisateurs peuvent naviguer dans les différentes catégories, présentant plus ou moins d'informations selon le niveau de zoom.

2.2.2.3 Filtrage

Les approches par filtrage consistent à réduire la quantité de données à afficher en appliquant des filtres sur un ou plusieurs attributs de l'espace d'information. Le niveau de détail varie donc dans le temps. Dans le cas de données temporelles, l'attribut à filtrer peut être le temps, afin de modifier la période à visualiser, comme montré à la Figure 2.12 avec une application de calendrier sur dispositifs mobiles. Les utilisateurs ont la possibilité d'afficher un jour, une semaine, un mois ou une année de données.
Karlson & al. [68] ont par exemple utilisé ce mécanisme pour développer Fathumb, une application permettant de naviguer dans une base de données de type pages jaunes. Les touches du dispositif mobile sont associées à des critères (lieu, distance, horaires, etc). En choisissant un critère l'utilisateur peut alors lui attribuer une valeur (par exemple Lieu = Grenoble). Dachselt & Frisch [33] ont également appliqué ce principe pour visualiser une collection musicale pouvant être filtrée selon différents critères comme la durée, le genre, l'artiste, etc.

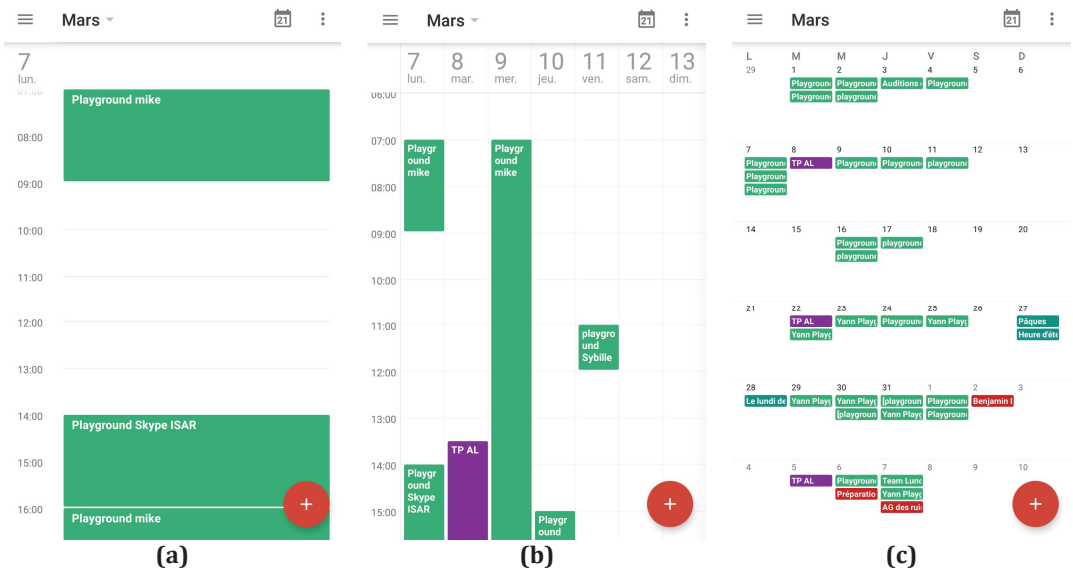


Figure 2.12 – Filtrage (Application Agenda sur Android) : (a) un jour, (b) une semaine, (c) un mois.

2.2.3 Techniques utilisant le multiplexage spatial

Les techniques utilisant le multiplexage spatial ont été classées en trois catégories par Roudaut [108] : Indices contextuels, Vue Globale+Détails et Focus+Contexte.

2.2.3.1 Indices contextuels

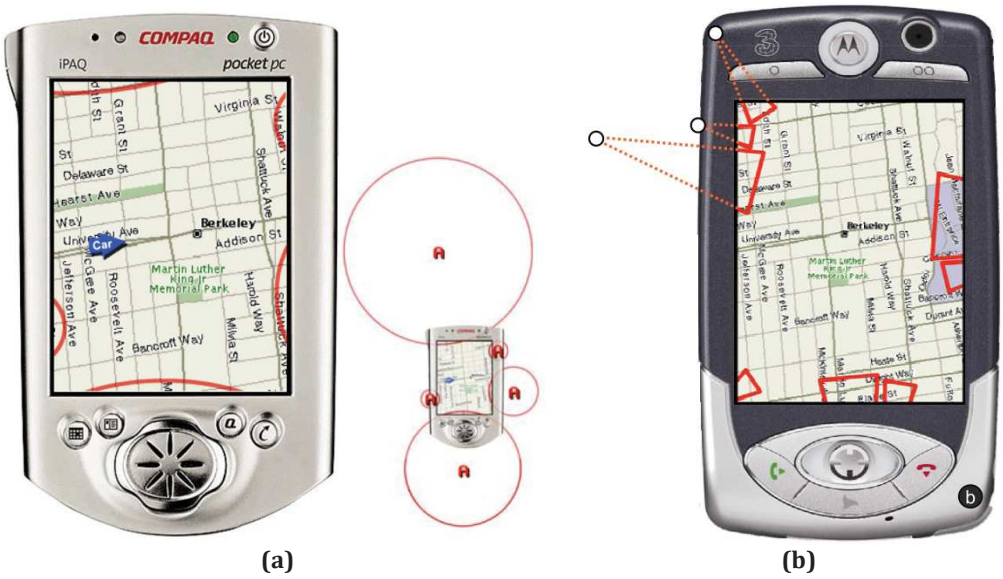
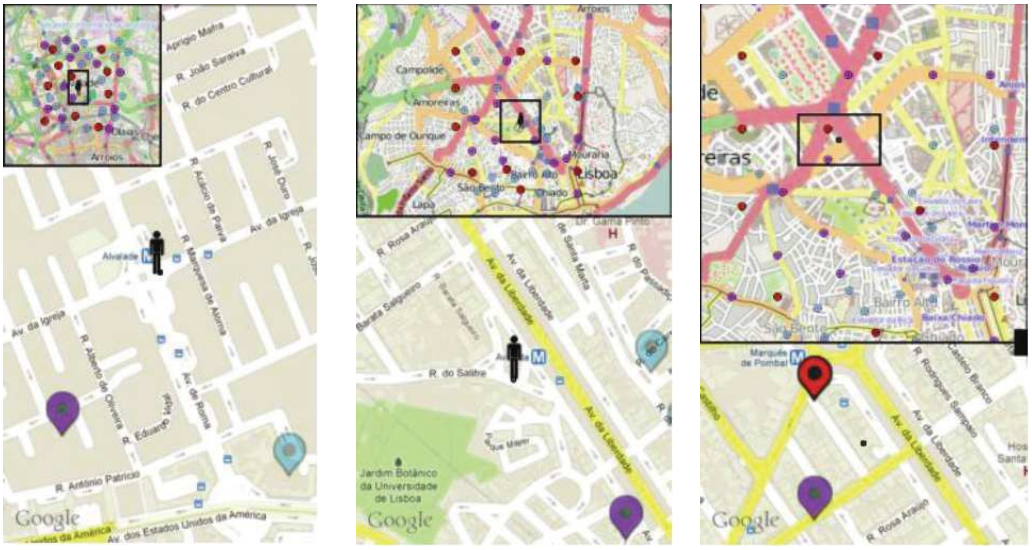


Figure 2.13 – Indices contextuels : (a) Halo, (b) Wedge. Respectivement issues de [10] et [50].

Cette première catégorie de techniques consiste à donner des informations sur des points d'intérêt situés en dehors de l'écran. Par exemple, Baudisch & Rosenholtz [10] utilisent des cercles (Halo) centrés sur les points d'intérêt. Grâce à la position

du cercle et à son diamètre, les utilisateurs peuvent deviner dans quelle direction et à quelle distance se trouve le point d'intérêt (Figure 2.13.a). Avec *Wedge*, Guftafson & Baudisch [50] étendent cette technique pour afficher des triangles à la place des cercles (Figure 2.13.b). Les triangles permettent d'éviter les chevauchements potentiels créés par les cercles, en offrant deux degrés de liberté supplémentaires : rotation et angle d'ouverture.

2.2.3.2 Vue Globale+Détails



(a) (b) (c)

Figure 2.14 – Vue Globale+Détails : (a) Vue globale de petite taille recouvrant la vue détaillée, (b) Vue globale ne recouvrant pas la vue détaillée, (c) Vue globale redimensionnable. Issues de [48]

Les techniques Vue Globale+Détails utilisent deux vues distinctes avec une vue globale de l'espace d'information affichée par dessus ou à côté d'une vue détaillée plus large. Généralement, la vue globale est représentée par une miniature de l'espace d'information complet dans lequel un cadre permet de situer la vue actuellement affichée dans la vue détaillée (Figure 2.14). Cette technique permet de visualiser en détail une partie de l'espace d'information tout en en conservant son contexte grâce à la vue globale. Cela évite à l'utilisateur d'effectuer des opérations de zoom arrière pour se repérer comme il est nécessaire de le faire pour s'orienter avec les techniques de Déplacement & Zoom par exemple.
Les techniques Vue Globale+Détails ont été montrées efficaces sur ordinateurs de bureau par Hornbaek & Forkjaer [56], mais ces résultats ne sont pas facilement transposables sur dispositifs mobiles. En effet, ces techniques nécessitent l'affichage de deux vues distinctes à l'écran alors même que la taille de l'écran est limitée sur dispositifs mobiles. Par exemple, dans l'étude de Pietriga & al. [98] la vue globale occupe 4.5% de la taille d'écran, ce qui n'est pas possible sur dispositifs mobiles avec un écran de petite taille : cela rendrait la vue globale difficilement lisible et manipulable, comme souligné par Chittaro [29]. L'enjeu de conception est alors de définir un bon compromis pour la taille des deux vues : si on choisit une vue globale trop petite elle devient illisible, mais l'espace réservé à la vue détaillée est plus important. Or c'est en général le principal point d'intérêt des utilisateurs puisque c'est le lieu de réalisation de leurs tâches.

Il convient de noter que contrairement aux Vues Globale+Détails conçues pour des ordinateurs de bureau, où la vue globale se situe généralement dans un coin de l'écran, il est possible sur dispositifs mobiles de visualiser les deux vues d'un seul coup d'œil : la largeur d'écran d'un ordinateur de bureau ne le permet pas. Les mouvements des yeux et donc les changements d'attention sont par conséquent beaucoup plus courts sur dispositifs mobiles.

Burigat & Chittaro [19] ont étudié l'impact de deux caractéristiques des techniques Vue Globale+Détails : mettre en évidence les points d'intérêt dans la vue globale, et permettre de déplacer la fenêtre d'affichage représentant la vue détaillée dans la vue globale pour se déplacer dans l'espace d'information. Leurs résultats montrent que chacune des caractéristiques est bénéfique aux techniques Vue Globale+Détails sur dispositifs mobiles en terme de performance, en soulignant que la manipulation de la vue détaillée dans la vue globale permet l'amélioration des performance la plus importante. De plus, Gonçalves & al. [48] ont comparé trois manières différentes de représenter la vue globale :

- Une « standard », de petite taille et recouvrant la vue détaillée (Figure 2.14.a)
- Une plus grande ne recouvrant pas la vue détaillée (Figure 2.14.b)
- Une redimensionnable (Figure 2.14.c)

Leurs résultats révèlent une préférence des utilisateurs pour une vue globale redimensionnable, ce qui confirme les recommandations de Plaisant & al. [99] sur l'adaptation de la taille de la vue globale et de la vue détaillée en fonction de la tâche à accomplir.

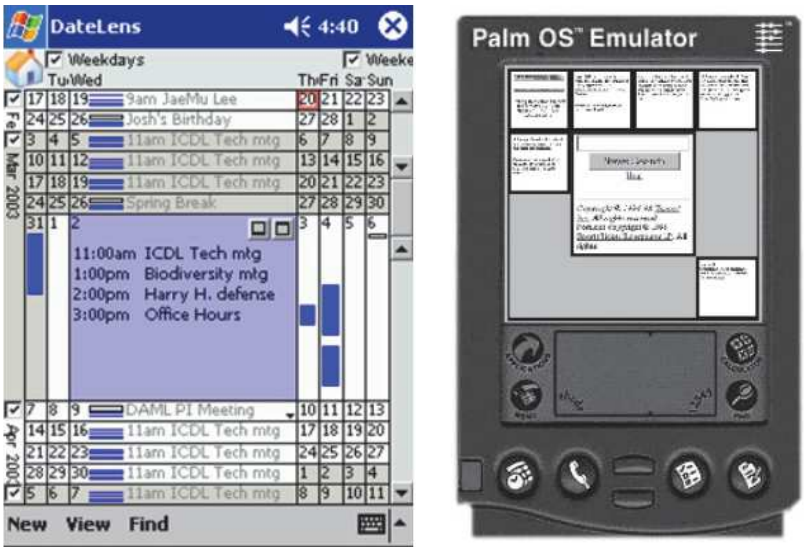
Néanmoins l'efficacité de ce type de technique reste mitigée sur dispositifs mobiles. Alors que Burigat & al. [20] ont montré que l'ajout d'une vue globale pouvait être comparable, voire être meilleure que les interfaces zoomables traditionnelles, d'autres études comme celle de Büring & al. [21] ont montré que les utilisateurs effectuaient des tâches de recherche plus facilement sans la vue globale.

2.2.3.3 Focus+Contexte

La classe de techniques précédente sépare les vues globales et détaillées dans deux espaces distincts, laissant à l'utilisateur l'effort de comprendre leur relation. Les techniques Focus+Contexte intègrent la vue globale (Contexte) et la vue détaillée (Focus), au sein d'une seule et même vue (Figure 2.15). Ceci réduit donc la charge cognitive nécessaire à la mise en relation des deux vues.

Cette classe de techniques de visualisation a été créée pour des ordinateurs de bureau avec les vues dites *bifocales* introduites par Apperley & al. [4]. Leur principe est de distordre l'espace d'information pour que les objets placés dans le focus soient agrandis alors que ceux en périphérie sont rétrécis. Les vues bifocales ont été généralisées et renommées vues en *œil de poisson* par Furnas [45]. Pour cela, Furnas propose la notion de degré d'intérêt permettant de calculer la pertinence de chaque objet dans l'espace d'information.
L'efficacité de ces techniques a été prouvée pour des ordinateurs de bureau avec différents types de données comme les menus [12], les listes [79] ou encore les arbres [76]. Comme pour la classe de visualisation précédente, les résultats

obtenus sur des ordinateurs de bureau ne sont pas facilement transposables sur dispositifs mobiles, notamment à cause de la taille d'écran réduite ne permettant pas d'allouer autant d'espace au focus et au contexte comme il est possible de le faire sur un écran d'ordinateur.



(a) (b)

Figure 2.15 – Vues Focus+Contexte : (a) DateLens, (b) Navigateur WEST. Respectivement issues de [13] et [15].

Le mécanisme de Focus+Contexte a été appliqué à différents types de données sur dispositifs mobiles. DateLens [13] (Figure 2.15.a) permet la visualisation d'un calendrier sur PDA en utilisant ce principe : le jour sélectionné est affiché de manière détaillée au centre de l'écran alors que le reste de l'espace est consacré aux autres jours pour lesquels ne sont affichées uniquement que les informations indispensables. Björk & al. [15] utilisent le même principe pour leur navigateur web *WEST*. Ainsi, chaque page web est découpée en un ensemble d'objets représentés sous forme de cartes (Figure 2.15.b). La carte sélectionnée est agrandie au centre de l'écran alors que les autres sont affichées autour de manière moins détaillée.

Toujours en proposant une vue Focus+Contexte sur dispositifs mobiles, Athmos [46] permet de visualiser une galerie d'images, Wang & Chi [130] propose de visualiser des cartes à deux dimensions (Figure 2.16), l'application de Lank & Phan [77] permet de dessiner sur PDA, AppLens [67] permet de visualiser et naviguer dans une matrice d'applications et SpiraList [57] permet de visualiser une liste de contacts sous forme de spirale (Figure 2.1.b).

Figure 2.16 – Metro maps. Issue de [130].

2.2.4 Techniques utilisant le multiplexage en profondeur

Cette dernière classe de techniques se base sur le principe de superposition des couches. Plusieurs vues peuvent alors être affichées en même temps et dans un même espace.

Figure 2.17 – Multiplexage en profondeur (ScatterDice Mobile). Issue de [125].

Roto & al. [107] utilisent ce principe avec le navigateur web *MiniMap* (Figure 2.8.c). Une vue globale de la page peut être affichée en semi-transparence au dessus de la partie détaillée. A l'inverse, ScatterDice Mobile [125] (Figure 2.5 et Figure 2.17) utilise la transparence pour afficher une sous-partie détaillée d'une matrice de graphiques en trois dimensions au dessus de la matrice globale. Le multiplexage en profondeur est également exploité dans les applications de réalité augmentée mobile, qui consistent à superposer des objets virtuels au dessus d'une vue réelle

retournée par la caméra du dispositif mobile. Aujourd'hui de nombreuses applications commerciales utilisent ce mécanisme comme par exemple Wikitude qui permet d'afficher des informations liées à ce que l'utilisateur est en train de voir (Figure 2.18).

Figure 2.18 – Application Wikitude[6].

Le multiplexage en profondeur a pour inconvénient de surcharger l'affichage avec différentes couches représentées: cet inconvénient identifié sur les ordinateurs de bureau [100], est exacerbé sur dispositifs mobiles avec la taille réduite de l'écran.

2.2.5 Synthèse

Nous avons présenté et classé différentes techniques de visualisation existantes sur dispositifs mobiles en nous référant à la notion de multiplexage. Nous avons présenté les trois catégories de multiplexage (temporel, spatial et en profondeur) de façon distincte, mais de nombreuses techniques utilisent une combinaison de celles-ci. SpiraList [57] par exemple, utilise le multiplexage spatial comme technique principale, mais emploie également le multiplexage temporel lorsque la liste est trop longue pour être affichée entièrement sur la spirale : les contacts sont alors regroupés selon leur première lettre, formant ainsi des groupes pouvant être élargis lorsque l'utilisateur les sélectionne. De manière générale, les techniques de multiplexage spatial utilisent le multiplexage temporel afin d'amener les points d'intérêt dans la vue détaillée (Vue Globale+Détails et Focus+Contexte). La classification se fait donc en fonction du type de multiplexage principal de la technique.

Les avantages et inconvénients de chacune des classes de techniques décrites sont résumés dans la Table 2.1. Bien que ces avantages et inconvénients sont semblables dans le cas d'un ordinateur de bureau, certains sont accentués dans le cas du dispositif mobile. Nous les notons en gras dans la Table 2.1.

Les techniques utilisant le multiplexage temporel sont couteuses en temps : ne permettant pas de visualiser simultanément une partie détaillée de l'espace d'information et son contexte, l'utilisateur est contraint d'effectuer de très

[6] http://www.wikitude.com

nombreuses opérations de zoom avant et arrière pour accomplir ses tâches sur un dispositif mobile.

Les deux autres types de multiplexage semblent donc offrir de meilleures options dans le contexte d'un écran de taille réduite. Le multiplexage en profondeur ne semble, lui, pas adapté à une quantité d'information trop importante, qui viendrait alors surchargé l'écran du dispositif mobile et rendrait son contenu illisible, surtout en situation de mobilité où l'attention visuelle est partagée entre l'écran et le monde physique.

Le multiplexage spatial semble donc être la meilleure solution. Parmi les catégories qu'elle contient, nous constatons que les techniques de Vue Globale+Détails n'optimisent pas l'espace à l'écran en utilisant deux vues distinctes : or il s'agit d'un critère central sur dispositifs mobiles. De plus, un effort mental est nécessaire par l'utilisateur afin de faire le lien entre les deux vues. Les indices contextuels, eux, ne font que donner des informations sur le contexte, sans le représenter. Si beaucoup d'informations sont à afficher, alors les bords de l'écran deviennent vite surchargés par les représentations des points d'intérêt.

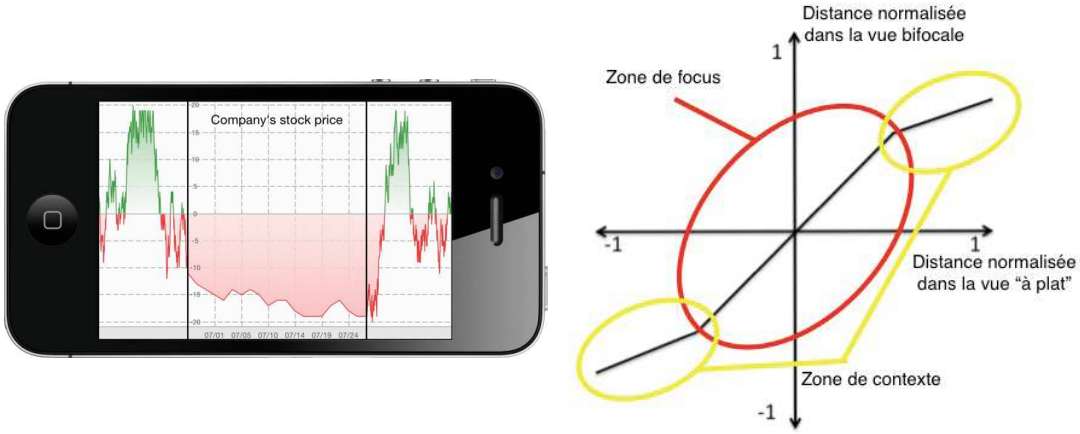
Enfin, les techniques Focus+Contexte résolvent les problèmes des deux catégories précédentes en fusionnant les deux vues et en permettant ainsi de visualiser simultanément tout l'espace d'information.

Par contre, tout comme le cas des Vue Globale+Détails qui ont déjà été étudiées [48], il est nécessaire de trouver un bon compromis entre la taille du focus et celle du contexte afin que le focus reste manipulable pour réaliser la tâche, tout en permettant au contexte d'apporter assez d'information. Nous décrivons donc dans la partie suivante la vue Focus+Contexte que nous avons conçue et implémentée sur dispositifs mobiles, ainsi que son évaluation expérimentale.

Technique	**Avantages**	**Inconvénients**
Barres de défilement	➢ Simple à comprendre (connues des utilisateurs d'ordinateurs de bureau) ➢ Ne déforment pas la vue initiale	➢ Pas de vision du contexte ➢ **Nécessite beaucoup de gestes** pour naviguer dans un grand espace d'information
Déplacement & Zoom	➢ Exploite les capacités d'interaction des dispositifs mobiles ➢ Permet d'avoir une vue globale ou une vue détaillée de l'espace d'information	➢ Pas de vision simultanée du contexte et des détails ➢ Nécessite de zoomer régulièrement en avant et en arrière pour basculer entre une vue globale et une vue détaillée
Filtrage	➢ Personnalisation possible des données à visualiser	➢ Pas de vision simultanée du contexte et des détails
Indices contextuels	➢ Espace de données détaillées maximisé	➢ Informations sur le contexte limitées ➢ Bords de l'écran surchargé si beaucoup d'informations dans le contexte
Vue Globale+Détails	➢ Vision simultanée du contexte et des détails	➢ Compromis sur les tailles des deux vues difficile ➢ Effort mental pour faire le lien entre les deux vues
Focus+Contexte	➢ Vision simultanée du contexte et des détails ➢ Contexte et détails dans une même vue	➢ Compromis sur les tailles des deux espaces difficile ➢ Transition entre focus et contexte difficile à appréhender si la déformation est trop importante
Multiplexage en profondeur	➢ Espaces pour la vue globale et la vue détaillée maximisés	➢ Ecran surchargé ➢ Vue du dessous peu lisible

Table 2.1 Synthèse : Avantages et inconvénients des différentes techniques de visualisation sur dispositifs mobiles.

2.3 Vue Focus+Contexte bifocale conçue sur dispositifs mobiles



(a) **(b)**

Figure 2.19 – (a) Vue bifocale, (b) Fonction de zoom de la vue bifocale.

Motivés par l'état de l'art dressé dans les sections précédentes, nos travaux concernent :

1) les types de données à une dimension et temporelles présentes dans de très nombreuses applications mobiles aujourd'hui : liste de mails, galerie d'images, graphes, etc.
2) les vues Focus+Contexte fusionnant deux vues permettant ainsi de visualiser simultanément tout l'espace d'information.

Nous avons donc conçu et implémenté une technique de type Focus+Contexte sur iPhone 4 pour des données à une dimension et temporelles.

2.3.1 Vue bifocale conçue

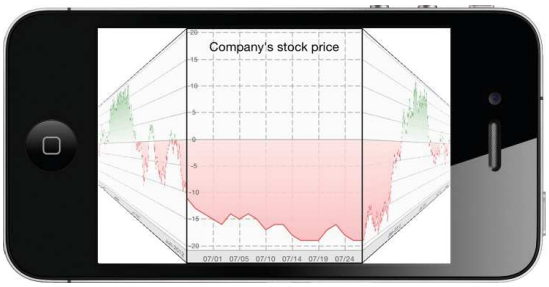
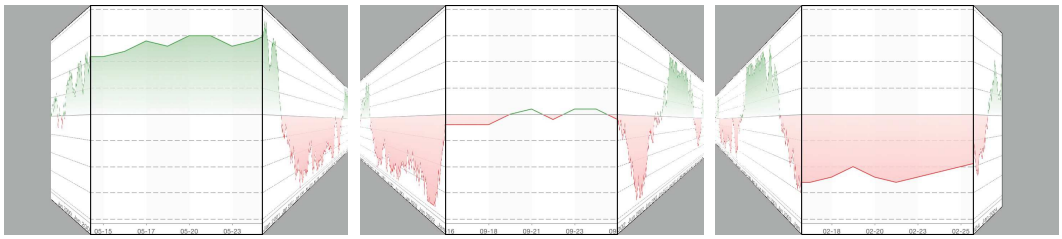


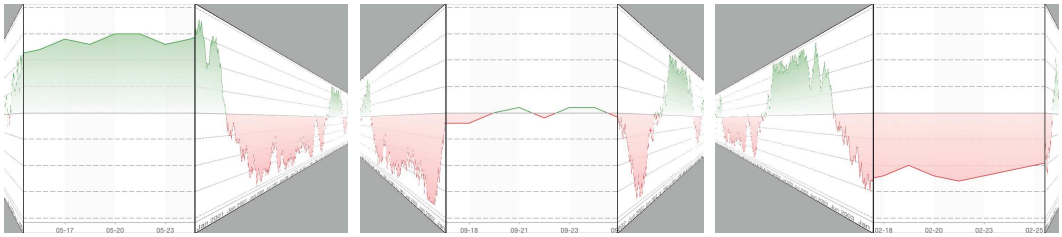
Figure 2.20 – Vue bifocale implémentée.

La technique de visualisation conçue est une vue bifocale [4]. Celle-ci comporte donc deux niveaux de détail : un pour le focus, et un pour le contexte. Le focus, avec un niveau de détail important, permet d'avoir un espace de travail pour les tâches précises comme les lectures de valeurs. Le contexte, avec un niveau de détail moins

important, permet d'obtenir des informations de plus haut niveau d'abstraction comme des tendances. Pour obtenir une vue détaillée au centre (le focus) et deux vues non détaillées de chaque côté (le contexte), nous avons appliqué une fonction de transformation à la dimension horizontale de l'espace d'information à une dimension. La fonction appliquée est celle décrite par Leung & Apperley [81] et illustrée à la Figure 2.19.b. Ainsi, les données affichées dans la zone de contexte sont représentées de manière plus condensée que celles affichées dans la zone de focus. La vue bifocale obtenue (Figure 2.19.a) est particulièrement adaptée pour optimiser la taille occupée à l'écran en tenant le dispositif en mode paysage. Cette orientation de l'écran est très utilisée pour le type de données considérées et notamment les données temporelles souvent représentées sous forme de courbes (graphes d'indices boursiers, applications de santé présentant des graphes sur le nombre de pas ou de calories brulées, etc.).



(a)



(b)

Figure 2.21 – Deux métaphores d'interaction : (a) Déplacer l'espace d'information (métaphore d'interaction du ruban) (b) Déplacer la zone de focus (métaphore d'interaction de la loupe).

Tout en fournissant une vue globale et détaillée intégrées, l'inconvénient est la discontinuité perceptible (Table 2.1) aux frontières entre la vue détaillée du focus et les vues déformées du contexte [81]. Le mur en perspective décrit par Mackinlay & al. [84] est une généralisation de l'affichage bifocal et applique une fonction de transformation non constante pour fournir une continuité perceptuelle entre le focus et le contexte. La solution que nous avons implémentée ne considère que deux niveaux de détail. Cependant, du mur en perspective [84], nous réutilisons l'effet de perspective 3D intuitif qui permet d'afficher plus d'informations dans l'espace du contexte [84] comme l'illustre la Figure 2.20.
La taille des différents espaces (Focus et Contexte) est un choix de conception difficile et nous avons mené une évaluation expérimentale comparant deux tailles différentes.

Enfin, pour naviguer dans l'espace d'information de cette vue bifocale, deux métaphores d'interaction sont possibles :

- Métaphore d'interaction du ruban : déplacer l'espace d'information complet, à l'image d'un ruban que l'on déroule (Figure 2.21.a). La zone de focus reste fixe.
- Métaphore d'interaction de la loupe : déplacer la zone de focus sur l'espace d'information, à l'image d'une loupe que l'on déplace au dessus de l'espace d'information (Figure 2.21.b). L'espace d'information reste fixe.

2.3.2 Expérience

Nous avons mené une étude qualitative pour comparer la vue bifocale mise en œuvre avec des vues plus couramment utilisées sur les dispositifs mobiles. Le principal objectif de cette expérimentation était de montrer par une évaluation qualitative (préférence et satisfaction de l'utilisateur) que la vue bifocale offre un bon compromis par rapport à deux autres visualisations plus couramment utilisées sur dispositifs mobiles et décrites dans la section suivante.
Cette évaluation nous a également permis de comparer deux tailles de focus différentes sur un écran d'iPhone 4 (960 pixels de largeur en mode paysage) pour la visualisation bifocale, afin de déterminer laquelle serait la plus appropriée à son utilisation sur dispositifs mobiles. En effet, comme décrit dans notre synthèse de la Table 2.1, un des inconvénients des vues Focus+Contexte est le compromis à adopter sur la taille de chacun des espaces. Dans cette étude, nous allons comparer deux tailles différentes sur différents types de tâches.
Nous avons décidé de ne pas évaluer les temps de réalisation des tâches car les résultats auraient été dépendant des techniques de navigation utilisées pour chacune des vues comparées. Notre objectif était ici d'évaluer uniquement les vues indépendamment de la technique de navigation utilisée, nous avons donc choisi de mener une étude purement qualitative.

2.3.2.1 Vues comparées

Nous avons comparé deux versions de la vue bifocale (taille de focus différente) à deux autres vues dans le cas d'un ensemble de 960 valeurs sur un écran de 960 pixels de large (iPhone 4) :

- Une vue globale affichant tous l'espace d'information à l'écran (vue ENTIRE), et dont les différentes valeurs sont accessibles grâce à un curseur manipulable avec le doigt (Figure 2.22-a). La taille de l'espace d'information a été sélectionnée pour être entièrement affichable sur l'écran de l'iPhone 4, en considérant un pixel par valeur. Comme annoncé, l'espace d'information résultant contient donc 960 données. L'utilisateur peut à tout moment déplacer son doigt à l'écran pour faire apparaître un curseur lui permettant de lire la valeur d'une donnée (Figure 2.22.a). Nous avons décidé de ne pas considérer ici les approches qui gèrent plus d'une valeur par pixel, comme l'interaction *subpixel* de Roussel & al. [106]. Notre objectif était de ne pas favoriser la vue bifocale en la comparant à une vue qui permet d'accéder à toutes les valeurs de l'espace d'information en déplaçant un curseur à l'écran.
- Une vue détaillée qui affiche les valeurs suffisamment large à l'écran pour être lisibles sans curseur (vue FOCUS), mais qui nécessite de naviguer pour parcourir l'ensemble de l'espace d'information (Figure 2.22-b). Celle-ci affiche donc seulement *n* données à l'écran (*n* inférieur au nombre de

données total). Il est donc nécessaire de naviguer à travers l'espace d'information pour parcourir les 960 valeurs qu'il contient. Pour cela, l'utilisateur effectue un geste de *Swipe* (glissement du doigt à l'écran à gauche ou à droite) pour se déplacer de *n* données dans l'espace d'information.

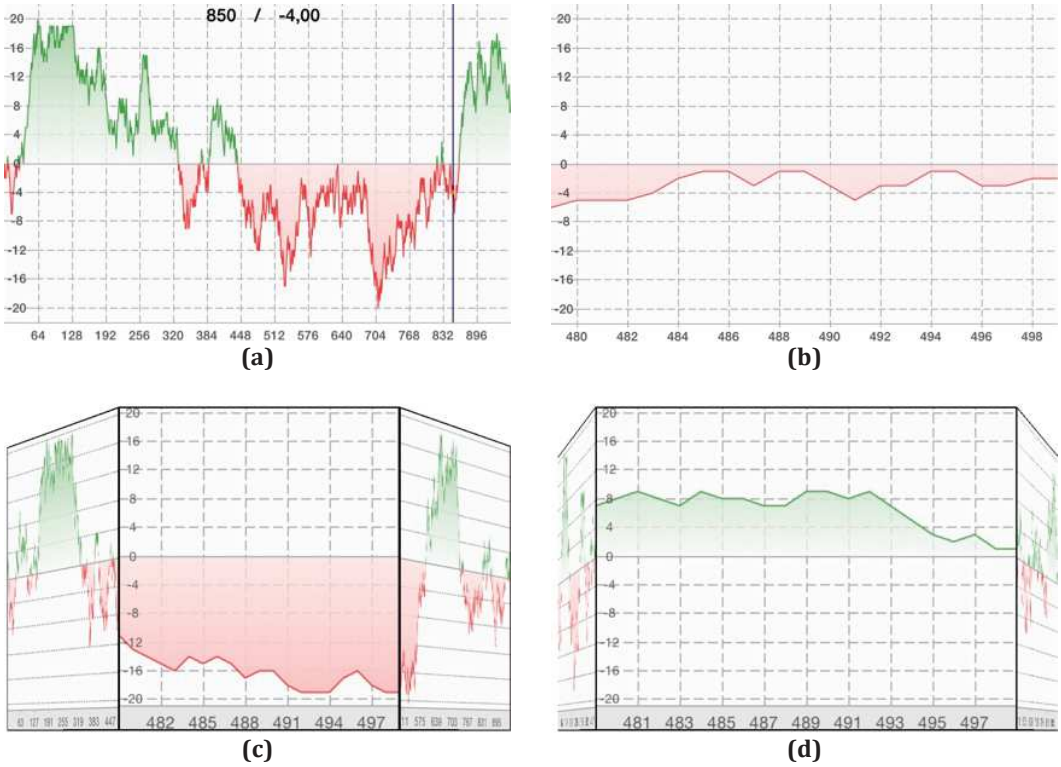


Figure 2.22 – Quatre vues comparées. (a) ENTIRE = Vue globale avec un curseur (trait mis en évidence en bleu) pour accéder à la valeur d'une donnée affichée en haut de l'écran (b) FOCUS = Vue détaillée affichant les valeurs de 20 données (c) BIFOCAL-Small = Vue bifocale avec un focus de 500 pixels de largeur (d) BIFOCAL-Large = Vue bifocale avec un focus de 800 pixels de largeur.

La vue bifocale affiche également *n* données dans le focus (*n* inférieur au nombre total de données), comme la vue détaillée décrite ci-dessus. De la même manière, le déplacement dans l'espace d'information se fait grâce au geste de *Swipe*, qui permet un déplacement de *n* données. Nous considérons deux tailles de focus différentes :

- Un focus de 500 pixels de largeur (vue BIFOCAL-Small), soit environ la moitié de l'espace disponible à l'écran. Ceci implique que le contexte affiche une donnée sur deux (Figure 2.22-c) pour la taille de l'espace d'information considéré.
- Un focus de 800 pixels de largeur (vue BIFOCAL-Large), soit environ 80% de l'espace disponible à l'écran. Ceci implique que le contexte affiche une donnée sur six (Figure 2.22-d) pour la taille de l'espace d'information considéré.

Pour les deux versions de la vue bifocale, l'utilisateur peut accéder aux valeurs des données du contexte (si disponibles) en déplaçant un curseur avec le doigt dans la vue du contexte. Les deux vues bifocales mettent en œuvre deux niveaux de détail, et diffèrent donc dans le niveau de détail du contexte en ayant des tailles de focus distinctes.

En référence aux deux métaphores d'interaction décrites dans la section précédente, la navigation dans la vue FOCUS s'effectue en utilisant la métaphore d'interaction du ruban. Cette métaphore d'interaction est celle couramment utilisée sur dispositifs mobiles pour ce type de tâche (navigation dans une liste de mails, une galerie d'images, etc.). Nous avons donc voulu la conserver. Ainsi, afin de rendre les différentes vues comparables nous avons également adopté cette métaphore d'interaction pour naviguer dans les deux vues bifocales.

Une étude pilote avec cinq utilisateurs a montré que les utilisateurs étaient capables de lire 20 valeurs maximum à l'œil nu (sans zoom, curseur ou outil supplémentaire) dans une zone de focus de 500 pixels de large. Nous avons donc défini *n* à 20 pour les techniques FOCUS, BIFOCAL-Small et BIFOCAL-Large.

2.3.2.2 Participants et tâches

Nous avons recruté 16 bénévoles, des ingénieurs professionnels en informatique à l'exception de 4 chercheurs de notre laboratoire. Les participants étaient tous utilisateurs réguliers de dispositifs mobiles tactiles et familiers des graphes comme ceux présentés à la Figure 2.22.

Pour définir les tâches à réaliser, nous nous sommes basés sur la taxonomie de Andrienko [2]. Celle-ci décrit 6 types de tâches et distingue des tâches élémentaires de celles dites complexes. Les tâches élémentaires agissent sur une donnée de l'espace d'information alors que les tâches complexes agissent sur un ensemble de données. De plus, certains types de tâche proposent deux versions : une directe et une indirecte. Une tâche directe consiste à trouver une caractérisque en fonction d'une référence (par exemple trouver une valeur pour un temps donné). Une tâche indirecte consiste à trouver une référence en fonction d'une caractéristique (par exemple trouver le temps pour lequel la valeur équivaut à une valeur donnée). Ainsi, ils distinguent au total 10 types de tâches décrits ci-dessous par des exemples :

- Tâches élémentaires
  - Recherche
    - Recherche directe : « Pour une date donnée, quelle est la valeur de l'indice boursier ? »
    - Recherche indirecte : « Pour une valeur donnée, trouver à quelle date l'indice boursier a atteint cette valeur. »
  - Comparaison :
    - Comparaison directe : « Comparer la valeur de l'indice boursier au premier et au dernier jour de la semaine. »
    - Comparaison indirecte : « Comparer les dates pour lesquelles l'indice boursier a atteint une valeur donnée. »
  - Recherche de relation : « A quelle date l'indice boursier a-t-il dépassé sa valeur à une date donnée ? »

- Tâches complexes
  - Identification de tendances
    - Identification de tendances directe : « Durant un intervalle donné, quelle a été la tendance de l'indice boursier ? »
    - Identification de tendances indirecte : « Trouver les intervalles pour lesquels la valeur de l'indice boursier a augmenté. »
  - Comparaison de tendances
    - Comparaison de tendances directe: « Comparer les comportements de l'indice boursier durant la première et la dernière semaine du mois. »
    - Comparaison de tendances indirecte: « Comment est la période de croissance de l'indice boursier par rapport à sa période de décroissance ?
  - Recherche de relation : « Durant quelle période la tendance de l'indice boursier était l'opposée de celle d'une période donnée ? »

De cette taxonomie [2], nous avons retenu uniquement les tâches élémentaires. En effet, les tâches plus complexes résultent de la combinaison de ces tâches élémentaires. Les cinq types de tâches sélectionnés sont :

- T1 : tâche de recherche directe d'une valeur proche du focus (nécessite au plus un geste de *Swipe* pour y parvenir). Un exemple de tâche concrète serait l'accès à la valeur d'un indice boursier où d'un taux d'intérêt qui soit proche de la date courante visualisée.
- T2 : tâche de recherche directe d'une valeur éloignée du focus (nécessite au moins 5 gestes de *Swipe* pour y parvenir). Un exemple de tâche concrète serait l'accès à la valeur d'un indice boursier le même mois que celui courant mais de l'année précédente.
- T3 : tâche de type synoptique impliquant d'explorer l'ensemble de l'espace d'information. Trois tâches ont été considérées : trouver la valeur maximale et minimale d'un indice boursier, et trouver la date la plus ancienne où la valeur d'un taux d'intérêt était nulle.
- T4 : tâche de comparaison entre deux valeurs proches (maximum deux gestes de *Swipe* pour naviguer entre les deux valeurs). Un exemple de tâche concrète serait de comparer la valeur d'un indice boursier sur deux jours consécutifs.
- T5 : tâches de comparaison entre deux valeurs éloignées (minimum dix gestes de *Swipe* pour naviguer entre les deux valeurs). Un exemple de tâche concrète serait de comparer la valeur d'un indice boursier le même jour de deux années consécutives.

Cet ensemble de tâches permet de montrer l'apport de chacune des vues comparées. L'exécution d'une tâche de type T5 semble par exemple particulièrement bien adaptée à la vue ENTIRE du fait que tout l'espace d'information est affiché à l'écran. Alors que la vue FOCUS semble plus adaptée aux tâches de type T1 ou T4 qui ne nécessitent pas un point de vue globale sur l'espace d'information. En choisissant cet ensemble de tâches, nous ne défavorisons donc pas l'une ou l'autre des vues comparées.

2.3.2.3 Procédure

Chaque participant a effectué les cinq types de tâches trois fois pour chaque vue comparée en utilisant un iPhone 4 en mode paysage (liste des tâches en Annexe 1). L'ordre de présentation des tâches a été décidé au hasard, mais l'ordre était le même pour les quatre vues. L'ordre des quatre vues a été contrebalancé par un carré latin. Les participants ont utilisé chaque vue avec un espace d'information différent. L'ordre d'attribution de ces espaces d'information aux différentes vues a également été contrebalancé par un carré latin.
Avant chaque vue, une description de celle-ci était faite aux utilisateurs. La consigne donnée aux 16 participants était d'être aussi rapide et précis que possible dans la réalisation des tâches. Après avoir effectué les 15 tâches pour une vue, les participants ont répondu à un questionnaire SUS [17] (Annexe 2): l'objectif était de recueillir le point de vue subjectif du participant sur la vue utilisée. Enfin, à la fin de la session, il était demandé aux participants de classer les quatre vues utilisées par ordre de préférence pour les cinq types de tâches (les ex-aequo étant autorisés). La session pour un participant durait en moyenne 45 minutes.

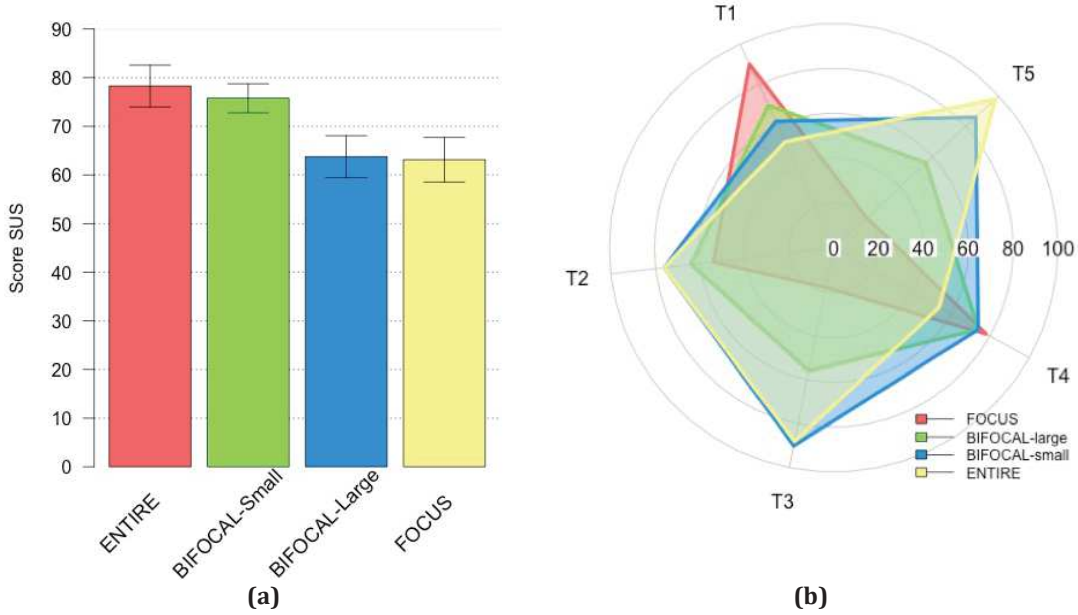


Figure 2.23 – (a) Les scores SUS moyens avec un intervalle de confiance correspondant de 95 % (b) Classement par préférences et par type de tâches des vues.

2.3.2.4 Résultats et discussions

La Figure 2.23.a illustre les quatre vues ordonnées de gauche à droite par ordre décroissant de score moyen SUS. La vue globale (ENTIRE) et celle bifocale avec un focus de 500 pixels de largeur (BIFOCAL-Small) obtiennent les scores SUS les plus élevés, en moyenne supérieurs à 75 (ENTIRE : 78.28, BIFOCAL-Small : 75.78). Les deux autres vues (BIFOCAL-Large et FOCUS) sont perçues comme moins utilisables (BIFOCAL-Large: 63.75, FOCUS : 63.12). Ainsi, sur l'ensemble des tâche proposées, le fait d'avoir un aperçu global de l'espace d'information semble important pour les participants puisque les deux techniques les moins bien notées affichent soit

aucune représentation du contexte (FOCUS), soit une zone de contexte de petite taille (BIFOCAL-Large). La remarque principale faite sur la technique FOCUS est que le manque de contexte oblige l'utilisateur à naviguer prudemment notamment sur les longues distances. En effet, s'il ne connaît pas parfaitement l'espace d'information, il ne connaît pas forcément les données présentes en périphérie du focus. Il ne peut donc pas naviguer rapidement car doit vérifier à chaque étape de la navigation si la donnée qu'il recherche ne vient pas d'apparaître dans le focus. Ce manque de contexte est aussi très gênant pour les tâches de comparaison. En effet, pour comparer deux valeurs, l'utilisateur est obligé de naviguer vers la première valeur, la retenir (car une fois qu'il aura naviguer ailleurs il n'aura plus d'aperçu de celle-ci), puis naviguer vers la seconde valeur. BIFOCAL-Large, malgré l'apport d'un contexte, n'a pas été plus appréciée. En effet, les utilisateurs ont rapporté que la taille allouée au contexte était trop restreinte pour permettre un réel apport.

La Figure 2.23.b présente les classements par préférences des vues pour les cinq types de tâches sous forme de diagrammes en radar. Pour calculer le score global de préférence, noté S, pour chaque vue et pour chaque type de tâches, nous avons appliqué la formule suivante : S = 7x1^{er} +5x2^{e} +3x3^{e} +4^{e} où 1er, 2e, 3e et 4e correspondent au nombre de fois où la vue a été classée au rang correspondant. Nous avons vérifié que des coefficients inférieurs ({4, 3, 2, 1} au lieu de {7, 5, 3, 1} comme ci-dessus), donnent des résultats comparables.
Les résultats confirment que la préférence pour une vue bifocale dépend du type de tâche, comme souligné par Bederson & al. [13] et Ganhör [46] pour d'autres types de données et dispositifs mobiles (dont un dispositif non tactile). En effet, la présence d'une vue globale de l'espace d'information (ENTIRE et BIFOCAL-Small) est par exemple très importante pour des tâches de comparaison et de navigation sur de longues distances. Au contraire, pour des tâches de navigation sur de courtes distances, le fait d'avoir une zone de lecture plus importante est un atout (BIFOCAL-Large et FOCUS). Aussi une vue bifocale n'est pas forcément meilleure dans tous les cas, mais n'est pas mauvaise peu importe la tâche réalisée. La vue bifocale BIFOCAL-Small est par exemple préférée pour la tâche de type synoptique impliquant une exploration de tout l'espace d'information. Pour les autres types de tâches, une des deux vues bifocales est toujours classée en deuxième position de préférence. Elle définit donc un bon compromis pour tous les types de tâches considérés.

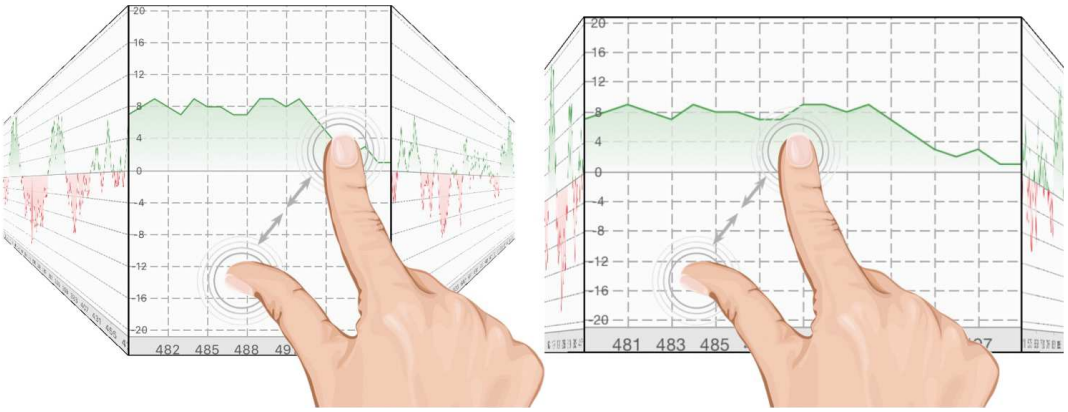


Figure 2.24 – Redimensionnement du focus avec un geste de *Pinch*.

Les résultats ne sont pas surprenants car connus sur ordinateur de bureau, mais sont utiles pour confirmer empiriquement l'intérêt d'une vue bifocale sur le petit écran d'un dispositif mobile quand différents types de tâches sont requis. En effet, la possibilité de visualiser simultanément un espace détaillé (le focus) et le reste de l'espace d'information (le contexte) est un apport considérable sur un écran de petite taille : ce point a été souligné par quasiment tous les utilisateurs. Dans le cas où une seule tâche est requise (par exemple une tâche de type T1), une autre technique pourra être plus appropriée (FOCUS dans le cas de T1), mais une vue bifocale (BIFOCAL-Large dans le cas de T1), est toujours une bonne option de conception.

Ainsi, dans le cas de l'exploration d'un espace d'information nécessitant l'emploi de différentes tâches, utiliser une vue bifocale est une bonne solution.

De plus, le classement différent par type de tâches des deux vues bifocales suggère de permettre à l'utilisateur de modifier la taille du focus comme le propose Mackinlay et al. [84] pour le Perspective Wall sur un ordinateur de bureau. Ceci confirme également les recommandations de Plaisant & al. [99] quant à la nécessité de pouvoir redimensionner les différents espaces de notre technique de visualisation en fonction de la tâche à réaliser. Sur dispositifs mobiles, nous avons implémenté cette possibilité de changer la taille du focus par un geste de *Pinch* effectué dans la zone de focus (Figure 2.24).

2.4 Conclusion

Dans ce chapitre consacré à la visualisation d'information sur dispositifs mobiles, nous identifions deux contributions : un état de l'art et une nouvelle technique de visualisation sur dispositifs mobiles.

Nous avons dressé un état de l'art structuré des techniques de visualisation sur dispositifs mobiles selon deux dimensions orthogonales :

1) Tout d'abord nous avons considéré les techniques de visualisation sur dispositifs mobiles selon les types de données. Cette analyse a motivé nos travaux sur la visualisation de données de types une dimension et temporelles.
2) Ensuite et de façon complémentaire, nous avons classé les techniques de visualisation sur dispositifs mobiles selon le type de multiplexage offert par la technique. A partir de cette analyse nous motivons nos travaux sur les vues de type Focus+Contexte.

Nous avons proposé une vue bifocale de type Focus+Contexte pour des données à une dimension et temporelles. Celle-ci fusionne deux niveaux de détail dans une seule vue en optimisant l'espace du contexte par une vue en perspective empruntée à la technique du Perspective Wall [84]. L'évaluation expérimentale menée avec 16 participants a montré l'intérêt d'utiliser une vue bifocale sur dispositifs mobiles, pour visualiser des données à une dimension et temporelles. Cette technique définit un bon compromis car adaptée à tous les types de tâches proposés dans notre évaluation. De plus la comparaison de deux tailles de focus différentes pour la vue bifocale nous a montré que le focus plus petit était préféré et que chaque taille était plus ou moins adaptée à la tâche à accomplir. Nous avons

ainsi modifié notre technique en proposant la possibilité de redimensionner la taille du focus pour l'adapter à la tâche à réaliser avec un geste de *Pinch*.

La principale tâche élémentaire pour une technique de visualisation Focus+Contexte comme notre vue bifocale est de déplacer les données du contexte dans le focus afin d'obtenir une niveau de détail adapté à la réalisation de la tâche [5]. Après avoir conçu, implémenté et évalué notre technique de visualisation, nous allons donc dans les prochains chapitres étudier différentes modalités pour naviguer dans celle-ci : déplacer un point d'intérêt du contexte vers le focus.
Notre approche de recherche a donc été d'étudier les visualisations graphiques (interface graphique en sortie) d'abord puis à focaliser sur les techniques de navigation (interface en entrée) ensuite. Ainsi ce chapitre a focalisé sur l'interface graphique en sortie et les prochains chapitres sont consacrés aux interfaces en entrée en étudiant les modalités d'interaction en entrée pour naviguer dans une vue bifocale.

3 INTERACTION AVEC UNE VUE BIFOCALE SUR DISPOSITIFS MOBILES : MODELE CONCEPTUEL DE NAVIGATION MULTIMODALE

3.1 Introduction

Au chapitre précédent, nous avons présenté la vue bifocale conçue et évaluée. Dans ce chapitre nous nous intéressons à l'interaction en entrée, et nous considérons les différentes modalités pour interagir avec la vue bifocale.

La tâche interactive élémentaire pour ce type de visualisation est de déplacer des données du contexte pour les placer à l'intérieur du focus et ainsi avoir un niveau de détail des données adapté à la réalisation de la tâche de haut niveau à réaliser. Il s'agit donc d'une tâche de navigation puisqu'elle consiste à déplacer un ensemble de données jusqu'à un point d'intérêt recherché.

Pour concevoir des modalités ou techniques d'interaction permettant de réaliser cette tâche de navigation, il convient de prendre en compte d'une part les contraintes liées au type de dispositif utilisé et au contexte de mobilité, et d'autre part les particularités inhérentes à une vue bifocale. Aussi la première partie de ce chapitre présente les contraintes liées à l'interaction sur dispositifs mobiles. La deuxième partie dresse un état de l'art des modalités de navigation sur dispositifs mobiles. Dans une troisième partie nous considérons les particularités liées à l'interaction avec une vue bifocale, pour enfin présenter un modèle de navigation

multimodale en trois phases. Ce modèle de navigation est conceptuel car il ne dépend pas d'une modalité particulière. Il permet de répondre aux besoins identifiés liés à la navigation dans ce type de visualisation.

3.2 Interaction sur dispositifs mobiles : contraintes

La conception de modalités d'interaction pour dispositifs mobiles doit faire face à plusieurs restrictions inhérentes à ce type de dispositifs.

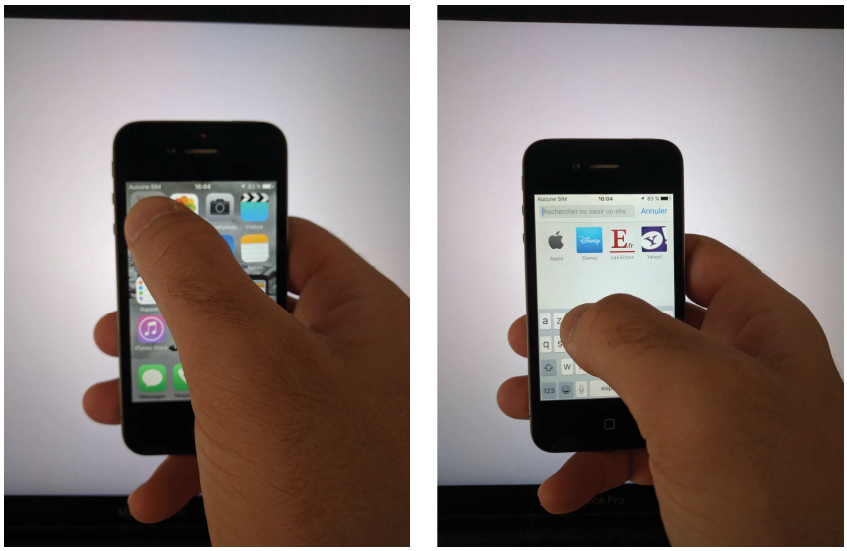
3.2.1 Taille d'écran réduite

Malgré des résolutions d'écran qui ont plus que doublé ces dernières années, la taille des écrans reste tout de même très limitée, dans le but de conserver la portabilité des dispositifs. Nous avons vu dans le chapitre précédent que cela définit un défi majeur pour la visualisation, mais c'est également une limitation pour l'interaction en entrée. En effet, aujourd'hui la majorité des dispositifs mobiles embarquent un écran tactile, et l'interaction au doigt est donc la plus couramment utilisée sur ce type de dispositif. Celle-ci entraîne différents problèmes décrits ci-après.

3.2.1.1 Occultation

Les dispositifs mobiles tactiles, à la différence des ordinateurs portables et de bureau, ont la particularité d'utiliser le même espace (l'écran) pour l'interaction en entrée et en sortie. Ceci entraîne donc des problèmes d'occultation comme illustré à la Figure 3.1.a.

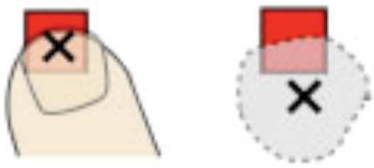
L'interaction à une main entraîne une occultation importante de l'écran par le pouce. Après un calcul approximatif de la surface du pouce à l'écran, Roudaut [108] a montré qu'environ 34% de l'écran était occulté par le pouce lorsque celui-ci couvre la diagonale de l'écran. Aujourd'hui les tailles d'écran des dispositifs mobiles ont augmenté (5 pouces en moyenne) : cette estimation du taux d'occultation a donc diminué légèrement mais l'occultation reste tout de même importante. De plus, les écrans tactiles actuels étant capables de détecter plusieurs points d'appui simultanément, ce problème d'occultation est amplifié lors d'une utilisation avec plusieurs doigts : utiliser deux doigts à l'écran multiplie le taux d'occultation par deux [58].



(a) (b)

Figure 3.1 – (a) Occultation de l'écran du dispositif mobile par le pouce, (b) « Fat finger » [118] : occultation des touches du clavier numérique.

L'interaction tactile provoque également une occultation de la cible à sélectionner lorsque que celle-ci est plus petite que la taille du doigt. Nous illustrons ce problème dans le cas du clavier numérique à la Figure 3.1.b. Ce problème a été identifié par Siek & al. [118] et désigné sous le nom de « Fat finger ». Il est accentué lors de la sélection de cibles de très petite taille car les utilisateurs ne perçoivent pas toujours le point de contact pris en compte par le système. Benko & al. [14] ont en effet montré que les utilisateurs, lorsqu'ils interagissent avec le pouce, perçoivent ce point au bout de leur pouce alors qu'il se trouve en réalité au centre (Figure 3.2). Ce phénomène rend ainsi difficile l'accès aux bords de l'écran car les utilisateurs peuvent avoir l'impression de toucher l'écran alors que le point d'appui est en réalité sur le bezel (contour) du dispositif mobile.



(a) (b)

Figure 3.2 – Erreur de perception de la position du pointeur sous le pouce : (a) Position perçue par l'utilisateur, (b) Position captée par le système. Issues de [108].

3.2.1.2 Accessibilité

En mode portrait, les dispositifs mobiles sont majoritairement utilisés à une main avec le pouce comme doigt d'interaction. De ce fait, certaines parties de l'écran sont plus facilement atteignables que d'autres avec ce seul doigt, comme montré par Karlson & Bederson [66] et illustré à la Figure 3.3. 50% de l'écran est difficile à atteindre en manipulant le dispositif mobile à une main. Là encore, ce problème est amplifié avec la taille grandissante des écrans de dispositifs mobiles. Ceci

s'explique d'une part par la taille des écrans (paradoxalement à ce qui a été dit précédemment, les écrans sont trop grands pour être manipulés à une main) et d'autre part par les limitations mécaniques des articulations de la main : Karlson & al. [67] ont par exemple montré que l'articulation à la base du pouce rend difficile les gestes diagonaux dans l'axe du pouce (gestes Nord-Ouest vers Sud-Est pour les droitiers).

Figure 3.3 – Zones difficilement accessibles (zones sombres) avec le pouce de la main qui tient le dispositif mobile dans le cas d'un droitier. Issue de [66].

Les capteurs embarqués dans les dispositifs mobiles actuels (comme les gyroscopes et les accéléromètres,) permettent d'autres modalités d'interaction que la modalité tactile (section 3.3.2). Lorsque la modalité d'interaction implique des mouvements de la main entière, il convient de considérer comme pour la modalité tactile les limitations mécaniques des mouvements de la main. Rahman & al. [87] ont étudié les amplitudes sur lesquelles peuvent s'étendre les mouvements de la main tenant le dispositif. Ils ont ainsi souligné les limitations des mouvements dans les différentes directions présentées à la Figure 3.4.

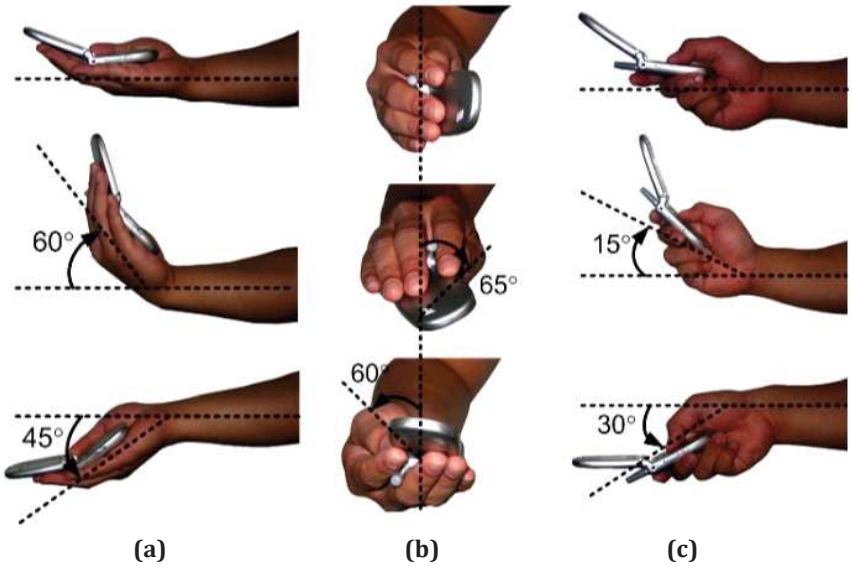


Figure 3.4 – Différentes amplitudes possibles des mouvements du poignet : (a) Flexion/extension (b) Pronation/supination (c) Inclinaison ulnaire/ inclinaison radiale. Issues de [102].

3.2.2 Modalités d'interaction en entrée limitées

Bien que les dispositifs mobiles actuels embarquent des capteurs permettant de considérer plusieurs modalités d'interaction (interaction gestuelle, interaction tactile, reconnaissance vidéo, capture de pression, interaction vocale, etc.), ces dernières doivent reposer sur les technologies que les dispositifs mobiles embarquent, contrairement aux ordinateurs de bureau ou portables sur lesquels il est possible de brancher différents types de dispositifs d'interaction (clavier, souris, joystick, etc.). Même s'il est techniquement possible de créer des adaptateurs matériels et logiciels pour brancher d'autres dispositifs d'interaction sur un dispositif mobile, la contrainte de mobilité reste souvent la limitation majeure.

3.2.3 Modalités d'interaction dépendantes du contexte d'utilisation

Comme présenté ci-dessus, les dispositifs mobiles actuels embarquent plusieurs capteurs offrant différentes possibilités d'interaction, ne limitant plus l'interaction en entrée à l'interaction tactile ou au clavier physique comme c'était le cas pour les premiers dispositifs mobiles. Par contre, les conditions de mobilité entraînent différentes contraintes qu'il faut prendre en compte lors de la conception d'une modalité d'interaction pour dispositif mobile (en entrée comme en sortie).
Ainsi, même si la majorité des dispositifs mobiles proposent aujourd'hui une interaction vocale, celle-ci est peu utilisée. En effet, en tant que modalité en entrée, celle-ci peut engendrer une gêne sociale dans un lieu public par exemple. De même, les capteurs tels que les gyroscopes ou les accéléromètres permettent aujourd'hui d'interagir de manière gestuelle avec un dispositif mobile. Ce type d'interaction n'étant pas très répandu, son utilisation peut aussi créer une gêne dans un lieu public par exemple.

De plus, la modalité auditive en sortie est peu utilisée : en plus de la difficulté à entendre dans un lieu bruyant, les utilisateurs peuvent également vouloir préserver la confidentialité de leurs informations.
Enfin, l'interaction tactile couplée à un retour visuel concentre l'attention de l'utilisateur sur l'écran de son dispositif mobile et peut le couper de l'environnement extérieur. Ceci peut être dangereux dans le cas d'une utilisation dans la rue par exemple, où l'utilisateur doit faire attention aux véhicules et passants l'entourant.

3.2.4 Synthèse

Les contraintes d'interaction inhérentes aux dispositifs mobiles sont d'une part liées à la taille de l'écran et aux capacités humaines de mouvements (du doigt ou de la main) et d'autre part aux limites technologiques des dispositifs mobiles et au contexte de mobilité. Ces contraintes sont identifiées et ont donné lieu à de nombreux travaux dans la communauté en Interaction Homme-Machine : une conférence ACM, MobileHCI, créée en 1998, est dédiée à l'interaction sur dispositifs mobiles. Ainsi pour pallier ces contraintes, de nombreuses techniques d'interaction ont été proposées. Pour ne citer que quelques exemples :
Pour pallier la difficulté d'accès au pouce à certaines zones de l'écran, plusieurs techniques d'interaction graphique ont été proposées. Par exemple Roudaut et al. [109] ont présenté *MagStick* une technique à curseur décalé. Le curseur est contrôlé par un bâtonnet télescopique qui est manipulé par le pouce et qui permet d'atteindre facilement des cibles en dehors des zones faciles d'accès avec le pouce (Figure 3.3).
Plusieurs techniques visent à améliorer la précision dans le pointage avec le pouce. Par exemple Vogel & Baudish [129] ont proposé *Shift* une technique de pointage précis en deux temps : un premier appui sur l'écran provoque l'affichage de la zone occultée par le pouce et l'utilisateur peut alors déplacer un curseur dans cette partie répliquée. Le relâchement du pouce provoque la disparition de la zone répliquée.
Pour offrir une interaction sans regarder (« eyes-free interaction »), l'adaptation des marking menus aux dispositifs mobiles, notée *Wavelet Menu*, autorise la sélection d'un élément du menu en mode expert sans regarder l'écran [44], par un geste appris en mode novice.

Face à la multitude des modalités d'interaction / techniques d'interaction traitant chacune d'une contrainte liée aux dispositifs mobiles, nous ne dressons pas un état de l'art général mais nous étudions dans la section suivante uniquement les modalités pour naviguer dans un espace d'information.

3.3 Modalités de navigation sur dispositifs mobiles

La modalité de navigation la plus couramment utilisée sur dispositifs mobiles avec écran tactile est le *Pinch/Drag/Flick* que nous décrivons dans la section suivante (3.3.2). Avec l'ajout de plusieurs capteurs embarqués dans les dispositifs mobiles actuels, d'autres techniques de navigation ont vu le jour, n'utilisant plus l'écran et évitant notamment les problèmes d'occultation décrits précédemment. Nous les présentons dans la section 3.3.3.

3.3.1 Grille d'analyse des modalités de navigation

En nous référant à la définition de Nigay & al. [95] d'une modalité comme un couple dispositif d'interaction *d* et langage d'interaction *l*, nous définissons le dispositif d'interaction *d* comme l'un des capteurs embarqués dans les dispositifs mobiles et utilisé pour l'interaction, et *l* comme le langage en entrée associé à ce dispositif.
Le langage *l* peut être caractérisé par deux types d'action :

- Continue
- Discrète

Les notions d'action continue ou d'action discrète caractérisent les actions effectuées par l'utilisateur au regard des commandes effectuées par le système. Continu/Discret décrit donc comment le langage d'interaction de la modalité interprète les événements reçus suite à des actions physiques sur un dispositif pour obtenir des commandes. Aussi nous caractérisons une action de discrète (respectivement continue) si elle déclenche une commande ou action-système élémentaire (respectivement une suite d'actions-système à une fréquence donnée).

Du point de vue de l'utilisateur : Une action continue ou discrète peut correspondre à une ou plusieurs actions physiques de sa part. Par exemple l'action <appui-relâchement> d'un bouton implique un ensemble d'actions physiques de la part de l'utilisateur. Au niveau du système, ces actions physiques captées par un dispositif comme une souris sont interprétées par le langage d'interaction pour ne définir qu'une seule action système. Il s'agit donc d'une action discrète. Cette définition est en accord avec le concept de « phrasing » de Buxton [23].

Du point de vue système : De façon orthogonale, le pilote d'un dispositif peut effectivement envoyer des événements de façon discrète ou continue. Continu/Discret peut donc aussi caractériser le fonctionnement interne d'un dispositif : par exemple le pilote d'une Wiimote fournit des événements de façon continue mais ce dispositif peut être utilisé pour effectuer des actions qualifiées de discrètes comme une commande de navigation dans des transparents de présentation avec l'application *SlideShow* [64].

Ces notions de continu/discret sont étudiées dans les travaux sur l'ingénierie de l'interaction multimodale. Ces notions sont par exemple explicites dans le modèle à composants de la plateforme logicielle OpenInterface pour l'interaction multimodale [115]. Dans notre étude, Continu/Discret caractérise comment les actions physiques effectuées par l'utilisateur sont interprétées au regard des commandes ou tâches élémentaires de navigation déclenchées.

Les actions effectuées par l'utilisateur peuvent définir deux tâches élémentaires de navigation dont la sémantique diffère. Nous positionnons ces tâches élémentaires au même niveau d'abstraction que celles de la taxonomie des tâches génériques de Foley & al. [43] :

- *Selection* : Sélectionner un objet
- *Position* : Positionner un objet
- *Orient* : Orienter un objet
- *Ink* : Dessiner un trait

- *Text* : Saisir un texte
- *Value* : Spécifier une valeur scalaire

Nous définissons ainsi deux tâches génériques de navigation :

- *Déplacer de* : se déplacer d'une distance *D* dans l'espace d'information. Les actions captées et interprétées définissent un déplacement dans l'espace d'information. Par exemple, lors d'une navigation avec une barre de défilement, l'utilisateur effectue des actions discrètes sur les flèches de navigation pour naviguer. Une action provoque par exemple un déplacement de 200 pixels dans l'espace d'information.
- *Aller à* : aller à la position *P* dans l'espace d'information. Les actions définissent une position dans l'espace d'information. Par exemple, lors d'une navigation avec une barre de défilement l'utilisateur appuie n'importe où sur la barre pour se déplacer à la position correspondante dans l'espace d'information.

Ces deux tâches élémentaires illustrées à la Figure 3.5 sont désignées dans la suite du manuscrit comme des modes de navigation :

- Relatif (Figure 3.5.a) : Ce mode correspond à la tâche élémentaire générique « Déplacer de ».
- Absolu (Figure 3.5.b) : Ce mode correspond à la tâche élémentaire générique « Aller à ».

Le mode de navigation, le type d'action (langage d'interaction) et le capteur (dispositif d'interaction) définissent trois dimensions orthogonales pour caractériser les différentes modalités de navigation utilisées sur dispositifs mobiles. Nous organisons la revue des modalités de navigation existantes de la section suivante selon le type de capteur, en d'autres termes selon le type de dispositif d'interaction de la modalité.

(a) **(b)**

Figure 3.5 – Deux modes de navigation différents (a) Relatif. (b) Absolu.

3.3.2 Modalités tactiles

Aujourd'hui la majorité des dispositifs mobiles possèdent un écran tactile. Tous proposent les gestes de *Pinch/Drag/Flick* pour interagir avec ce dernier. Le geste de *Pinch* est associé à la fonction de zoom : rapprocher deux doigts à l'écran diminue le niveau de zoom alors que les éloigner l'augmente (Figure 3.6.a). Les gestes de *Drag/Flick* sont associés aux fonctions de déplacement et sont couramment utilisés en mode de navigation relatif (Figure 3.5.a). Le geste de *Drag* consiste en un déplacement du doigt à l'écran (Figure 3.6.b) : un déplacement du doigt de un pixel à l'écran provoque un déplacement de un pixel dans l'espace d'information (Figure 3.6.b). Le geste de *Flick* est un geste de *Drag* auquel on a donné une vélocité supérieure à un certain seuil lorsqu'on relâche le doigt de l'écran (Figure 3.6.b). Cette combinaison de gestes est la plus utilisée sur dispositifs mobiles, avec des fonctions de transfert du déplacement du doigt en vitesse de navigation différentes selon les systèmes, comme étudié par Quinn & al. [101].

Or, en plus des problèmes d'occultation (décrits ci-dessus) causés par ce type de modalités, leur utilisation peut également s'avérer fastidieuse dans le cas de larges espaces d'information. La surface d'interaction étant réduite, l'amplitude des gestes l'est également. Ainsi, pour naviguer de longues distances, les utilisateurs doivent effectuer du « clutching ». Le clutching consiste à relâcher temporairement le doigt de l'écran, puis à le repositionner en l'air pour retoucher l'écran près du point de contact initial, comme illustré à la Figure 3.7. Le clutching est inévitable dans le cas où un seul geste de *Drag/Flick* ne permet pas de parcourir toute la distance. Pendant que la main est repositionnée pour effectuer le prochain geste, aucune action en entrée n'est effectuée. Quinn & al. [101] ont montré que le temps moyen de clutching est de 160ms dans le cas d'une navigation très rapide, et augmente à 536ms en moyenne dans le cas d'une navigation exploratoire, plus lente. Dans le cas de distances longues impliquant de nombreux gestes, le temps de « clutching » n'est donc pas négligeable.

Nous caractérisons les gestes de *Drag* comme des actions continues : en effet ils déclenchent une série de déplacements à l'écran entraînant un déplacement équivalent dans l'espace d'information. En revanche, les gestes de *Flick* sont un peu particuliers. En effet, ce sont des gestes de *Drag* auxquels a été ajouté un effet d'inertie lors du relâchement du doigt. Ainsi, la première partie du geste (*Drag*) est une action continue, alors que le relâchement du doigt déclenchant l'inertie définit une action discrète.

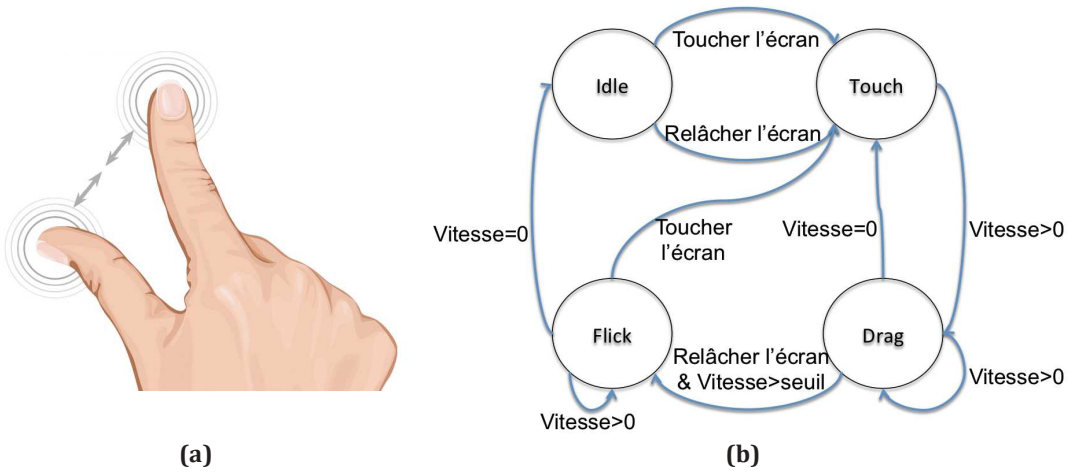


Figure 3.6 – (a) Geste de *Pinch*. (b) Machine à état des gestes de *Drag* et *Flick*.

Flick&Brake, une technique définie par Baglioni & al. [7], permet de n'utiliser qu'un seul geste de *Flick* peu importe la distance à naviguer. La navigation est initiée à l'aide d'un geste de *Flick*, qui entraîne une navigation continue à vitesse constante tant qu'aucun doigt ne touche l'écran. Un fois l'écran touché, la pression appliquée définit la vitesse à laquelle le défilement ralentit, comme si l'utilisateur freinait le défilement à l'écran avec son doigt. L'utilisateur peut ainsi contrôler la vitesse de décélération du *Flick*. En dessous d'une certaine vitesse, la navigation s'arrête.
Comme décrit précédemment, le seul geste de *Flick* utilisé pour lancer la navigation utilise une action continue (*Drag*), et une action discrète (relâchement du doigt) que le système interprète comme le début de la navigation .

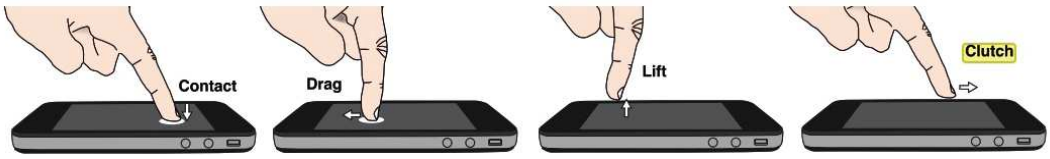


Figure 3.7 – Clutching. Issue de [101].

Fitchett & Cockburn [40] ont présenté *MultiScroll*, une technique de navigation basée sur les gestes de *Drag/Flick* permettant de limiter le clutching. Pour cela ils utilisent deux modes : un rapide et un lent. Le mode lent est basé sur les gestes de *Drag/Flick* classiques décrits ci-dessus et est efficace sur les distances courtes et moyennes. Le mode rapide permet de parcourir les longues distances. Ce mode rapide est basé sur un geste de *Drag* permettant de parcourir l'ensemble de l'espace d'information d'un seul geste. Ainsi, parcourir X% de l'écran avec ce mode fait défiler X% de l'espace d'information. Les deux modes de navigation utilisant tous les deux un geste de *Drag*, la désambigüisation se fait grâce au nombre de doigts à l'écran : un geste de *Drag* avec un doigt active le mode lent tandis qu'un geste de *Drag* avec deux doigts active le mode rapide.

Toutes les modalités et techniques décrites ci-dessus autorisent un mode de navigation relatif.

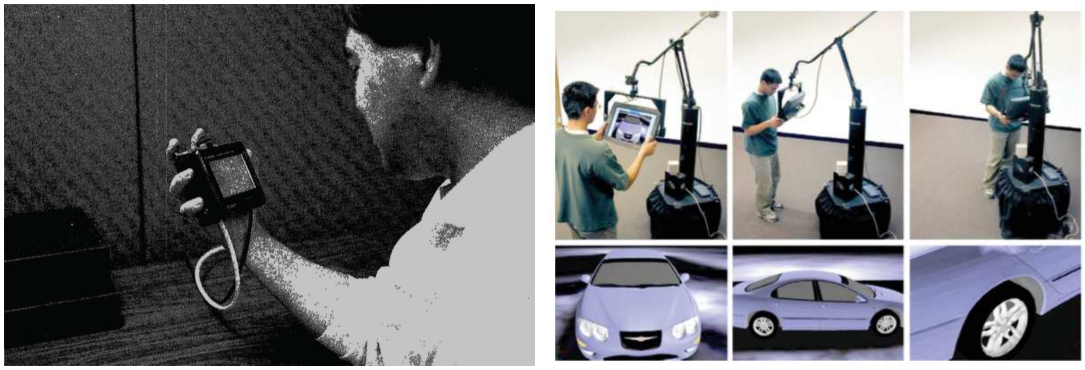
3.3.3 Modalités reposant sur les capteurs du dispositif

La présence de capteurs dans les dispositifs mobiles actuels a permis l'étude de plusieurs modalités de navigation qui ne reposent pas sur l'interaction tactile. La plupart de ces modalités ont pour avantage de limiter voire éliminer le problème d'occultation, puisque l'interaction ne se fait plus à l'écran. De plus, ces modalités permettent d'assigner les gestes de *Pinch/Drag/Flick* habituellement utilisés pour la navigation à d'autres tâches.

Nous classons ces modalités de navigation en trois catégories : les modalités impliquant des déplacements du dispositif, les modalités reposant sur l'inclinaison du dispositif et les modalités basées sur des capteurs de pression.

3.3.3.1 Modalités utilisant les déplacements du dispositif

Les capteurs embarqués dans les dispositifs mobiles permettent de détecter ses déplacements. Ces modalités consistent à positionner et déplacer le dispositif mobile au dessus de l'espace d'information (Figure 3.5.b).



(a) **(b)**

Figure 3.8 – Navigation utilisant les déplacements du dispositif : (a) Chameleon. (b) Boom Chameleon. Respectivement issues de [42] et [126].

La technique Chameleon présentée par Fitzmaurice & al. [42] fût l'une des premières à utiliser l'interaction spatiale par déplacements dans l'espace du dispositif mobile: le dispositif sert alors de fenêtre donnant accès à un espace « virtuel » que les utilisateurs peuvent explorer en déplaçant le dispositif dans l'espace (Figure 3.8.a). Avec Chameleon, les utilisateurs peuvent se déplacer dans une scène en trois dimensions en utilisant les six directions possibles (haut, bas, gauche, droite, avant, arrière). Ce principe a ensuite été repris dans plusieurs études. Yee [134] a par exemple combiné cette interaction spatiale avec une interaction au stylet sur le dispositif mobile et l'a évaluée sur différents types d'application : navigation dans une liste, dans une image de grande taille, ou application de dessin. Tsang & al. [126] ont couplé l'interaction spatiale avec une interaction tactile pour visualiser et annoter un modèle à trois dimensions (Figure 3.8.b). Deux autres études [97, 122] ont comparé l'interaction spatiale avec les gestes de *Pinch/Drag/Flick* pour naviguer dans un espace d'information à deux dimensions. Les deux techniques étaient conçues de la même manière : les déplacements horizontaux et verticaux du dispositif mobile permettaient de se déplacer de gauche à droite et de haut en bas, alors que les déplacements d'avant

en arrière permettaient de zoomer et dézoomer. Les deux études ont montré que la navigation basée sur l'interaction spatiale offre des performances meilleures que celles du *Pinch/Drag/Flick*, et particulièrement lorsque l'utilisateur navigue vers une cible qu'il a déjà visitée.
Ces modalités sont utilisées en mode de navigation relatif avec des actions continues. En effet, une fois la navigation activée, les utilisateurs peuvent effectuer des mouvements fluides dans l'espace pour se déplacer dans l'espace d'information. Dans ce cas, les utilisateurs effectuent des actions continues (correspondant aux mouvements entre l'activation et la désactivation du déplacement), dans un mode de navigation relatif.

3.3.3.2 Modalités exploitant l'inclinaison du dispositif

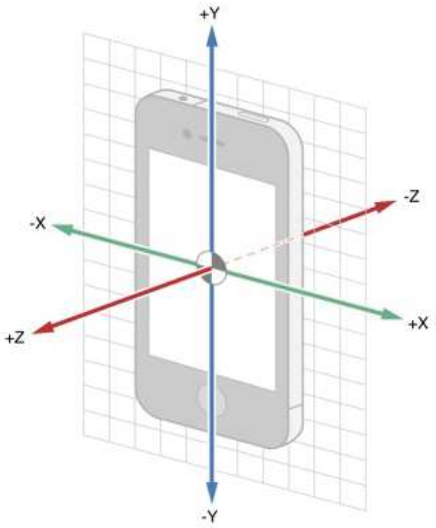


Figure 3.9 – Axes de rotation du mobile[7].

Les gyroscopes et accéléromètres embarqués dans les dispositifs mobiles permettent d'obtenir relativement précisément les angles de rotation du dispositif selon trois axes (Figure 3.9). Ils sont par exemple utilisés nativement sur la plupart des dispositifs mobiles pour adapter l'orientation d'une image ou d'une application à celle de l'écran du dispositif. Ces capteurs sont aussi couramment utilisés pour contrôler des jeux vidéo : contrôler la trajectoire d'un véhicule dans un jeu de course par exemple.

Il convient de noter que les modalités en entrée exploitant ces capteurs ont été étudiées bien avant que les capteurs ne soient embarqués dans les dispositifs mobiles. Le fait d'incliner le dispositif pour interagir est connu sous le geste de *Tilt*. Inspiré de la taxonomie de Scoditti & al. [114] nous distinguons trois manières d'exploiter ce geste pour naviguer dans un espace d'information :

[7] http://developer.apple.com

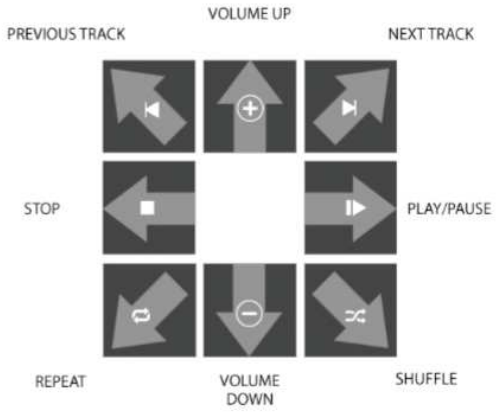


Figure 3.10 – Reconnaissance de gestes à partir de l'inclinaison du dispositif pour le contrôle d'un lecteur musical. Issue de [6].

- **Par reconnaissance de gestes :** l'inclinaison du dispositif mobile est interprétée comme un geste (par exemple, inclinaison du dispositif sur l'axe Y vers la gauche). Les gestes reconnus sont associés à des commandes (Figure 3.10), ce sont donc des actions qualifiées de discrètes. Par exemple, Barlett [8] a construit un vocabulaire gestuel basé sur l'orientation du dispositif mobile pour naviguer dans une galerie d'images. Chacune des orientations est associée à une seule commande : afficher une image en plein écran, revenir au dossier des miniatures, passer à l'image suivante ou encore passer à l'image précédente. Baglioni & al. [6] utilisent également ce principe avec *JerkTilts*. Les auteurs utilisent huit orientations différentes pour les associer à huit commandes (Figure 3.10). Pour activer une commande, les utilisateurs doivent donc incliner le dispositif mobile dans une direction et immédiatement revenir à la position initiale : c'est l'ensemble de ces actions physiques qui déclencheront une action-système ou commande.

- **Par position :** l'inclinaison du dispositif mobile permet de contrôler la position d'un curseur (Figure 3.11.a). Ainsi, plus le dispositif mobile est incliné, plus le curseur est placé loin dans l'espace d'information. Cette catégorie de modalités utilise donc un mode navigation absolu : l'inclinaison du dispositif définit la position du curseur dans l'espace d'information. Oakley & O'Modhrain [96] utilisent ce principe pour associer chaque élément d'un menu à une inclinaison du dispositif. Sad & Poirier [111] ont également utilisé le Tilt basé sur la position pour des tâches de déplacement. Le Tilt basé sur la position implique d'allouer une plage angulaire à un élément de l'espace d'information, comme illustré à la Figure 3.11.a. Ainsi, le nombre d'éléments pouvant être différenciés avec un geste de *Tilt* est limité. Rahman & al. [102] ont étudié le niveau de contrôle possible avec le poignet. Leurs recommandations sont d'utiliser principalement les axes de pronation/supination et flexion/extension pour interagir par inclinaison du dispositif (Figure 3.4). De plus, ils recommandent de limiter à 8 le nombre d'éléments sur l'axe flexion/extension, et à 12 sur l'axe pronation/supination. Au delà de ces

seuils, ils ont observé une dégradation des performances et du taux d'erreur.

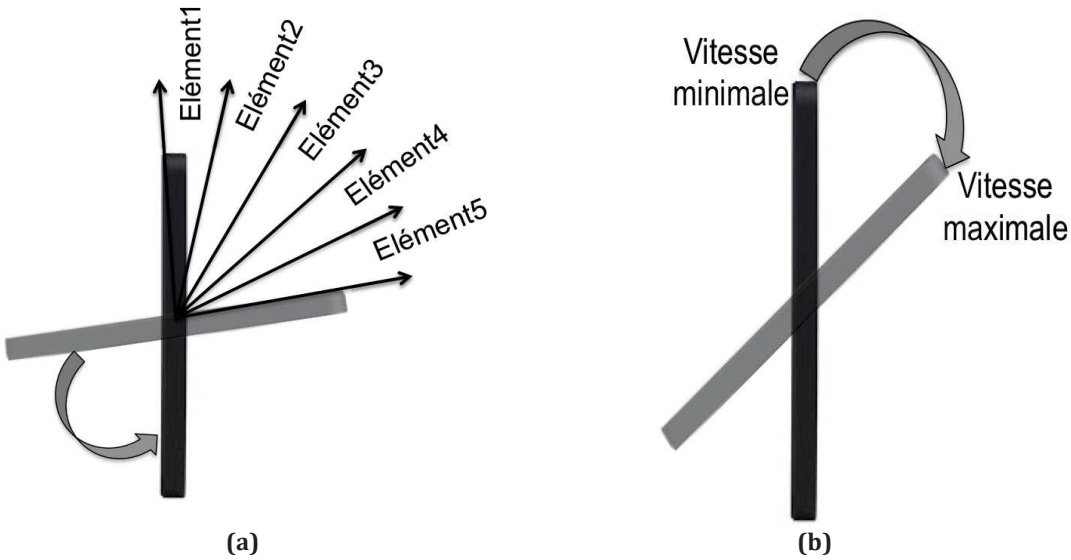


Figure 3.11 – Navigation par inclinaison du dispositif selon l'axe X : (a) Par position. (b) Par vitesse.

- **Par vitesse :** l'inclinaison du dispositif mobile est associée à une vitesse de défilement (Figure 3.11.b). Ainsi, plus le dispositif est incliné, plus la vitesse de navigation est grande. Cette catégorie de modalités utilise donc un mode de navigation relatif : l'inclinaison permet de spécifier un déplacement dans l'espace d'information. Cho & al. [30] utilisent ce principe pour naviguer dans une galerie d'images : incliner le dispositif mobile à gauche (respectivement à droite) fait défiler les images vers la gauche (respectivement vers la droite). Avec le *Tilt* basé sur la vitesse, le défilement est actif tant que le dispositif mobile reste incliné. Rekimoto [103] (qui fût le premier à étudier cette modalité), utilise également le geste de *Tilt* pour définir la vitesse de défilement pour naviguer dans des menus et des cartes. Fitchett & Cockburn [39] ont comparé les gestes de *Drag/Flick* avec une modalité utilisant le geste de *Tilt* basé sur la vitesse. Ils ont montré que dans le cas de tâches de lecture et d'analyse de données (qui impliquent principalement des mouvements lents), le geste de *Tilt* obtient des performances au moins équivalentes à celles des gestes de *Drag/Flick*.

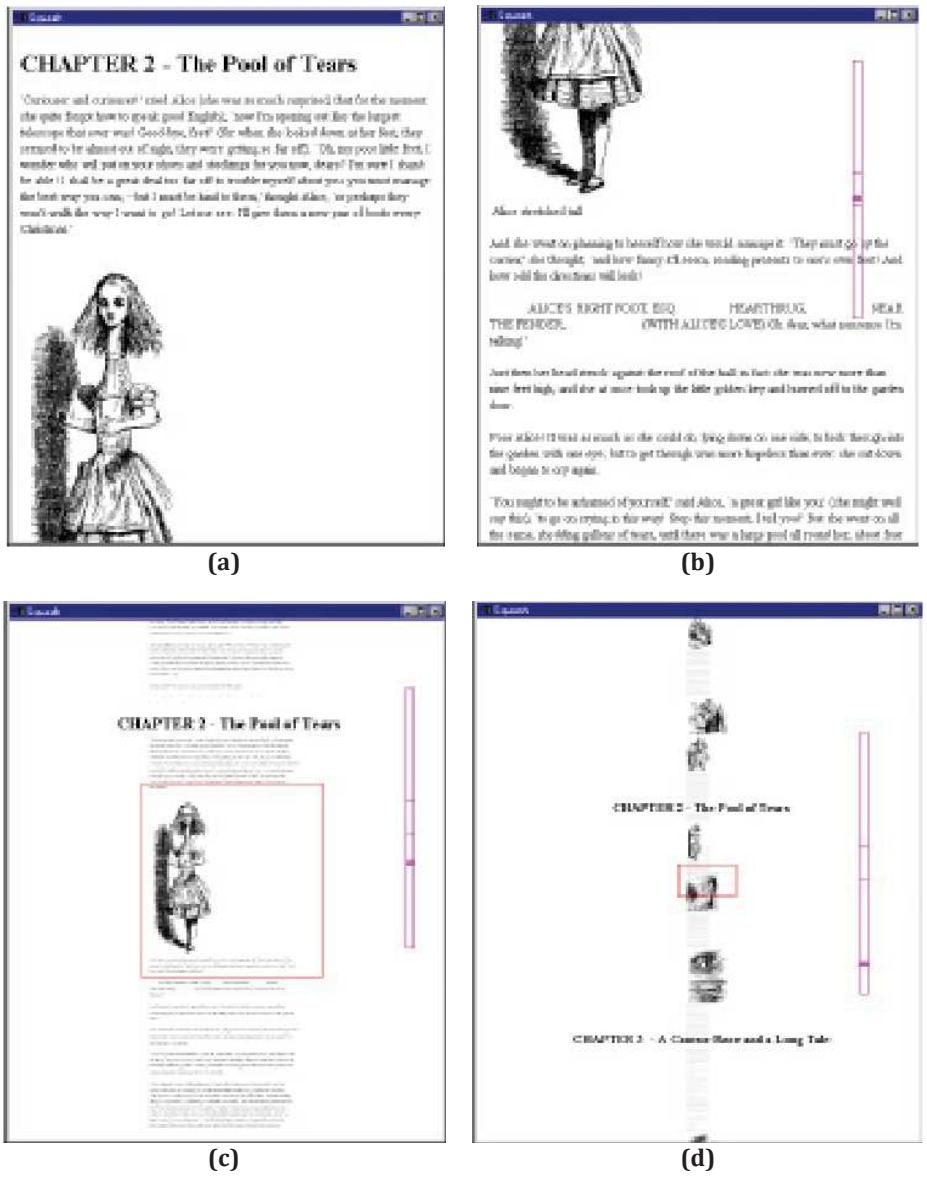


Figure 3.12 – SDAZ : (a) Vue statique, (b) Défilement lent, (c) Défilement rapide, (d) Défilement très rapide. Issues de [59].

Inspirées du travail de Igarashi & Hinckley [59] sur ordinateur de bureau, des études [37, 72] ont proposé sur dispositifs mobiles un zoom automatique en fonction de la vitesse (SDAZ pour Speed-Automatic Dependant Zooming). Cette technique permet d'adapter le niveau de zoom de l'espace d'information en fonction de la vitesse de défilement : plus celle-ci est rapide, plus le niveau de zoom est faible, et vice versa (Figure 3.12). Le geste de *Tilt* est alors utilisé pour naviguer dans une carte à la place des gestes tactiles standards. Si cette technique a obtenu de bons résultats sur ordinateur de bureau, les résultats sur dispositifs mobiles restent mitigés. En effet, l'avantage de ce type de technique est d'utiliser des indices contextuels pour se repérer dans l'espace d'information lorsque la navigation est rapide et le niveau de zoom faible. Or sur dispositifs mobiles, quand le système applique un niveau de zoom plus faible alors que

l'utilisateur navigue rapidement, la capacité à lire les indices contextuels est réduite par rapport aux ordinateurs de bureau du fait de la taille d'écra réduite et de l'inclinaison de l'écran. Kratz & al. [72] ont alors choisi de rendre le niveau de zoom modifiable par l'utilisateur par interaction tactile sur l'écran, tout en naviguant grâce au geste de *Tilt* (Figure 3.13). Les performances de l'interaction multimodale résultante sont meilleures qu'avec SDAZ, mais toujours moins bonnes que celles avec les gestes tactiles (section 3.3.2 Modalités tactiles).

Figure 3.13 – Navigation par inclinaison du dispositif : la vitesse de navigation est calculée par rapport à l'angle d'inclinaison et le niveau de zoom est contrôlé à l'écran par l'interaction tactile. Issue de [72].

Le geste de *Tilt* basé sur la reconnaissance de gestes utilise des actions discrètes, à la différence du geste de *Tilt* basée sur la position ou la vitesse qui utilisent des actions continues. Imaginons que pour faire défiler une page dans un document, il faille incliner le dispositif mobile vers le haut ou vers le bas, il faudrait alors répéter le geste 10 fois pour faire défiler 10 pages. Le geste de *Tilt* basé sur la position ou la vitesse semble donc plus efficace pour naviguer dans de grands espaces d'information. Leur différence réside dans le mode de navigation utilisé : absolu pour le geste de *Tilt* basé sur la position et relatif pour le geste de *Tilt* basé sur la vitesse. Teaser & Mackenzie [105] ont comparé ces deux approches et ont montré que le geste de *Tilt* basé sur la position offre de meilleures performances et est préférée par les utilisateurs que le geste de *Tilt* basé sur la vitesse. Ces résultats sont valides dans le contexte de l'étude qui comporte uniquement 12 cibles, ce qui reste un nombre manipulable par un geste de *Tilt* [87]. Cependant, le geste de *Tilt* basé sur la position n'est pas adapté à de très grands espaces d'information, comme précisé par les auteurs de l'étude.
Toutes les techniques basées sur l'inclinaison du dispositif souffrent de la même limitation : être capable de distinguer :

- les gestes de *Tilt* non intentionnels qui sont fréquents sur dispositifs mobiles comme en marchant,
- des gestes de *Tilt* pour naviguer.

Une solution proposée par Ruiz & Li [94] consiste à utiliser un geste particulier qui agit comme délimiteur pour les mouvements non volontaires. Ce geste permet à l'utilisateur de contrôler l'activation de la détection de l'inclinaison. Une autre solution consiste à considéré une zone neutre dans laquelle aucun défilement n'est déclenché. C'est ce qu'ont fait Fitchett & Cockburn [39] en utilisant une plage

d'angles de 13° (6.5° de chaque côté) : tous les mouvements réalisés dans cette zone n'ont aucun impact sur la navigation.

3.3.3.3 Modalités utilisant des capteurs de pression

(a) (b)

Figure 3.14 – (a) Résistance à détection de force (b) Prototype de clavier avec capteurs de pression, issue de [31].

Les derniers *smartphones* comme ceux d'Apple et Huawei embarquent désormais des écrans tactiles sensibles à la pression. Mais il est également facile d'ajouter un capteur de pression à un dispositif mobile existant sans en modifier le facteur de forme. En effet, des capteurs de petites tailles sont disponibles, comme les résistances à détection de force (Figure 3.14.a). Tout comme les modalités décrites ci-dessus, utiliser un capteur de pression a l'avantage de limiter voire éliminer le problème d'occultation si le ou les capteurs ne se trouvent pas sur l'écran du dispositif mobile. Les modalités utilisant des capteurs de pression se partagent en deux catégories similaires à celles des modalités utilisant l'inclinaison du dispositif :

- **Contrôle par position :** la position dans l'espace d'information est relative à la pression appliquée sur le capteur. Ainsi, la pression maximale correspond au dernier élément de l'espace d'information alors qu'une pression minimale correspond au premier élément. Cette catégorie de modalités utilise donc un mode de navigation absolu : la pression appliquée sur le capteur définit la position du curseur dans l'espace d'information. Plusieurs études [91, 132, 133] ont montré que l'utilisation de capteurs de pression pouvait être précise jusqu'à 8-10 éléments. Ce type de contrôle est donc majoritairement utilisé pour des tâches de saisie de texte [16, 87] ou de sélection dans un menu [133]. De plus, Wilson & al. [131] ont comparé l'utilisation de la pression basée sur la position avec celle basée sur la vitesse (décrite ci-dessous) pour naviguer dans un menu. Ils ont montré que la navigation dans le menu basée sur la vitesse était plus performante que celle basée sur la position, à la fois en condition de mobilité ou statique.

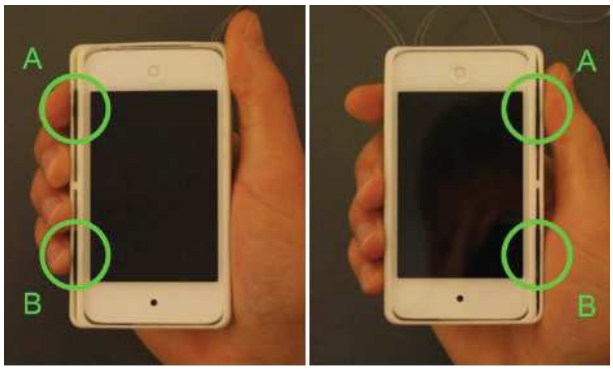


Figure 3.15 – SidePress : technique utilisant deux capteurs de pression pour naviguer dans un espace à une dimension. Issues de [119].

- **Contrôle par vitesse :** la vitesse de défilement est contrôlée par la pression exercée sur le capteur. Cette catégorie de modalités offre donc un mode de navigation relatif : la pression appliquée sur le capteur permet de spécifier un déplacement dans l'espace d'information. Clarkson & al. [31] ont utilisé ce principe en ajoutant un capteur de pression sur chacune des touches du clavier physique d'un dispositif mobile (Figure 3.14.b). Ils ont utilisé ce prototype pour naviguer dans une carte. Ainsi, chacune des touches (à l'exception de 5, 0, # et *) est assignée à une direction : par exemple la touche 1 déplace la vue vers le coin supérieur gauche. La vitesse de déplacement est alors proportionnelle à la pression exercée sur la touche. Force Gestures [55] augmente les gestes tactiles en couvrant l'écran d'un film relié à plusieurs capteurs de pression. Il est ainsi possible de détecter la pression exercée par un doigt à l'écran. Ceci permet par exemple de transformer un simple geste de *Drag* en une action plus riche dépendant de la pression appliquée sur l'écran durant l'exécution du geste. Ainsi, dans le cas d'une application de lecture de livre électronique, un même geste de *Drag* pourra faire défiler une ou plusieurs pages selon la pression appliquée. *GraspZoom* [90] utilise un capteur placé sur le dos du dispositif mobile combiné à l'interaction tactile pour naviguer et zoomer dans un espace d'information à deux dimensions. Les utilisateurs doivent par exemple toucher la zone sur laquelle ils veulent zoomer ou vers laquelle ils veulent naviguer, puis presser le capteur de pression pour effectuer l'action plus ou moins rapidement en fonction de la pression appliquée. Le capteur de pression ne permettant pas d'action bi-directionnelle, l'inversement du sens de zoom ou le changement de direction pour le défilement s'effectue en faisant un geste à l'écran avant de presser le capteur. Pour éviter d'interagir avec l'écran lors du changement de direction Spelmezan & al. [119] ont conçu *SidePress*, une technique utilisant deux capteurs de pression placés sur la tranche du dispositif mobile pour naviguer dans un espace d'information à une dimension (Figure 3.15). Chacun des capteurs est donc dédié à un sens de navigation. Ils ont associé à ces capteurs un vocabulaire de navigation basé sur les actions effectuées sur les capteurs. Il est ainsi possible d'utiliser les deux types d'actions :
    - continue : plus l'utilisateur presse fort, plus la navigation est rapide ;
    - discrète : l'utilisateur peut effectuer des clics <appui-relâchement> pour déclencher des sauts dans l'espace d'information. Un appui

léger provoque un petit saut tandis qu'un appui fort provoque un saut plus grand.

3.3.3.4 Modalités autour du dispositif

Figure 3.16 – HoverFlow : interaction autour du dispositif grâce à des capteurs infrarouges. Issues de [73].

L'interaction autour du dispositif utilise la reconnaissance de gestes effectués dans l'espace, autour ou au dessus du dispositif mobile. Elle permet de limiter le problème d'occultation, et permet également d'élargir la surface d'interaction limitée à la taille de l'écran dans le cas de l'interaction tactile. Différentes technologies sont utilisées pour cela :

- **Capteurs infrarouges :** Les premiers prototypes utilisaient des capteurs infrarouges fixés sur le dispositif mobile [22, 73]. *SideSight* [22] utilise des capteurs fixés sur chaque tranche du dispositif et orientés vers l'extérieur, détectant ainsi les doigst approchant le dispositif mobile par les côtés. Ceci a l'inconvénient de limiter les gestes à une surface plane autour du dispositif.
  HoverFlow [73] améliore la solution précédente en orientant les capteurs vers le haut. Les mouvemens de doigts peuvent donc être captés lorsqu'ils sont effectués au desssus du dispositif mobile, permettant ainsi de reconnaître des gestes dans une zone à trois dimensions au dessus du dispositif (Figure 3.16).

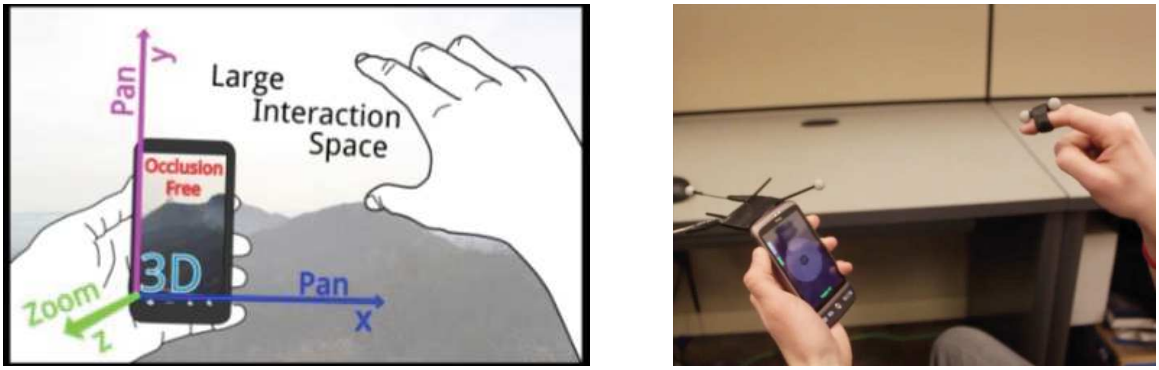


Figure 3.17 – Interaction autour du dispositif grâce à un système de suivi par caméra. Issues de [60].

- **Suivi par caméras :** Jones & al. [60] ont utilisé un système de suivi par caméras. Les auteurs ont comparé différentes techniques d'interaction autour du dispositif mobile pour naviguer dans un espace d'information multi-échelle. Ils ont pour cela utilisé des marqueurs placés sur les doigts dédiés à l'interaction, ainsi que sur le dispositif (Figure 3.17). Les mouvements de chacun des doigts, ainsi que leur distance par rapport à l'écran du dispositif sont captés. Dans [60] la distance du doigt à l'écran était utilisée pour définir la vitesse de navigation : plus les gestes étaient effectués proche de l'écran, plus la navigation était rapide. Ils ont montré que ce type d'interaction pouvait obtenir des résultats similaires aux gestes de *Pinch/Drag/Flick*, tout en étant préféré par les utilisateurs.

Figure 3.18 – Interaction autour du dispositif grâce à un capteur magnétique sous forme de bague. Issues de [70].

- **Capteurs magnétiques :** Couplées au magnétomètre déjà embarqué dans la plupart des dispositifs mobiles, ces techniques reposent sur un ou plusieurs capteurs magnétiques, sous forme de bagues par exemple, pour détecter les mouvements des doigts à proximité du dispositif (Figure 3.18). Cette solution a été adoptée par différentes études [27, 54, 69, 70].

L'utilisation de capteurs infrarouges nécessite d'ajouter du matériel non embarqué aux dispositifs mobiles, modifiant ainsi leur facteur de forme. Les solutions de suivi par caméras souffrent de limitations dues à l'occultation des marqueurs utilisés pour détecter les doigts et le dispositif. Les solutions utilisant les capteurs magnétiques ne souffrent pas des deux limitations citées. Elles permettent une interaction n'importe où autour du dispositif mobile, autorisant ainsi une interaction au dos de celui-ci (Figure 3.18).

Pour toutes les techniques décrites dans cette section, les actions effectuées peuvent être discrètes ou continues. En effet, alors que les gestes reconnus par la caméra dans [60] reproduisent les gestes de *Drag/Pinch* dans l'espace (actions continues), *HoverFlow* [73] repose sur la reconnaissance de gestes, un geste étant

associé à une action de défilement dans l'espace d'information (action discrète), le défilement se faisant élément par élément par une suite de gestes (Figure 3.16).

3.3.4 Synthèse

De cette revue des modalités de navigation sur dispositifs mobiles, nous notons tout d'abord que la modalité tactile avec les gestes de *Pinch/Drag/Flick* est la plus utilisée aujourd'hui. Les capteurs embarqués dans les dispositifs mobiles permettent néanmoins d'envisager d'autres modalités que tactiles : aussi de nombreuses autres modalités ont été proposées avec des performances qui ne sont pas toujours meilleures que la modalité tactile. Néanmoins la modalité tactile souffre d'une limitation importante qui est l'occultation de l'écran alors que la tâche consiste à naviguer dans l'espace d'information affiché. C'est donc une limitation importante de l'interaction tactile pour la tâche que nous considérons dans nos travaux. Ceci a motivé nos travaux visant à explorer d'autres modalités que tactiles pour naviguer dans la vue bifocale.

En nous référant à la définition de Nigay & al. [95] d'une modalité comme un couple dispositif d'interaction *d* et langage d'interaction *l*, nous avons présenté plusieurs modalités dont le dispositif d'interaction *d* est l'un des capteurs embarqués dans les dispositifs mobiles. Pour le langage d'interaction *l*, nous retenons la distinction entre les actions continues et les actions discrètes. En effet et indépendamment du capteur, l'utilisateur peut effectuer des actions qui seront interprétées comme continues ayant une épaisseur temporelle et déclenchant plusieurs actions-système, ou comme discrètes déclenchant qu'une seule action-système. Par exemple, l'utilisateur peut naviguer dans l'espace par un appui plus ou moins fort sur un capteur de pression ou par une suite d'actions <appui-relâchement> du capteur de pression [119]. Bien que l'action <appui-relâchement> sur un capteur de pression peut être modélisée à un niveau fin comme une suite d'actions élémentaires discrètes et continues, dans le cadre de la navigation par sauts dans l'espace d'information, nous la considérons comme une action dite discrète.

Enfin les deux modes de navigation que sont l'accès absolu et l'accès relatif concernent le niveau sémantique de l'interaction en définissant comment les actions captées (*d*) et segmentées (*l*) selon une modalité sont traduites en changement de positions dans l'espace d'information.

Nous concluons que les modalités de navigation sur dispositifs mobiles peuvent s'étudier selon trois dimensions orthogonales que nous comparons au niveau lexical, syntaxique et sémantique en linguistique, que sont :
le capteur embarqué dans le dispositif mobile qui définit le dispositif d'interaction (*d*) :
le type d'action définissant la segmentation (*l*) des actions comme continues ou discrètes,
le mode de navigation, absolu ou relatif, définissant l'action-système déclenchée à partir des évènements physiques captés et traités.

Ces dimensions sont particulièrement structurantes des travaux menés sur l'interaction sur dispositifs mobiles. Notons néanmoins que les deux premières

dimensions sont générales à toute modalité d'interaction mais dans cet état de l'art nous avons considéré les différents dispositifs (d) d'interaction en entrée disponibles sur dispositifs mobiles. La troisième dimension est elle spécifique aux modalités de navigation : pour cela nous avons défini deux tâches élémentaires et génériques de navigation que nous désignons par le terme « mode de navigation ».

3.4 Navigation avec une vue bifocale

Ayant présenté un panorama des modalités de navigation sur dispositifs mobiles, nous détaillons dans cette section les besoins liés à la tâche de navigation dans une vue bifocale.

Comme décrit dans le chapitre 2, la vue bifocale, de type Focus+Contexte, permet de visualiser à la fois une partie de l'espace d'information en détail, et son contexte global au sein d'une même vue. Rendre interactive ce type de vue consiste à fournir à l'utilisateur un moyen de spécifier quelle partie de l'espace d'information doit être affichée en détail.

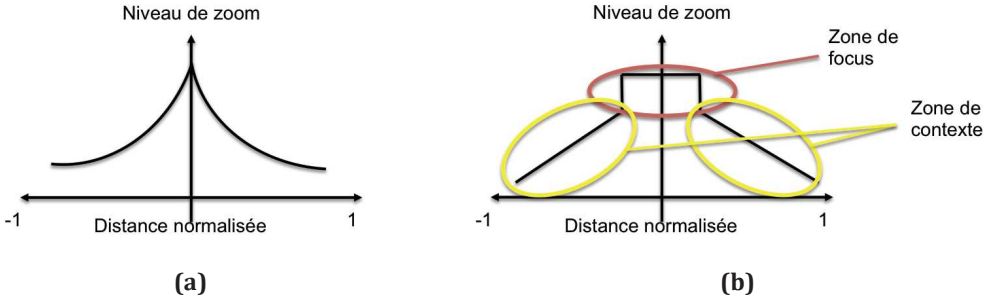


Figure 3.19 – (a) Fonction de zoom d'une vue en œil de poisson (b) Fonction de zoom de la vue bifocale.

Le problème majeur dans la réalisation de cette tâche est la déformation existante entre la zone de focus et la zone de contexte qui rend la tâche difficile, notamment lorsqu'il y a une déformation graduelle entre les deux zones (cas des vues en œil de poisson décrites dans le chapitre 2). Dans ce cas, le déplacement d'une cible est de plus en plus rapide au fur et à mesure qu'elle s'approche de la zone de focus, comme expliqué par Gutwin [51]. En effet, la déformation géométrique non linéaire (Figure 3.19.a) fait que pour un même nombre de pixels déplacés à l'écran, la distance parcourue dans l'espace d'information sera moins importante lorsqu'on se trouve loin du focus. La vitesse la plus rapide est donc atteinte au moment où la cible est au plus proche de la zone de focus, car c'est la zone où le niveau de zoom est le plus important.

Ce n'est pas le cas de la vue bifocale considérée et présentée à la section 2.3.1 du chapitre 2 qui possède une fonction de zoom linéaire (Figure 3.19.b). Ainsi, à vitesse constante, la distance parcourue dans la zone de contexte est la même. Par contre, à cette même vitesse, la distance parcourue dans la zone de focus est multipliée par *fz*, où est *fz* est le facteur de zoom entre le focus et le contexte. Ainsi, la difficulté de la tâche réside dans le passage de la cible de la zone de contexte à la zone de focus puisque deux échelles de vitesse sont utilisées. Le modèle conceptuel

de navigation de la section suivante vise à réduire cette difficulté de réalisation de la tâche de navigation liée au changement d'échelle.

3.5 Modèle conceptuel de navigation dans une vue bifocale

Comme décrit dans le chapitre 2, pour naviguer dans une vue bifocale deux métaphores d'interaction différentes sont applicables :

- Faire défiler l'espace d'information à l'intérieur de la zone de focus qui reste fixe, comme c'est le cas avec la technique Perspective Wall [84] (métaphore d'interaction du ruban);
- Faire défiler la zone de focus au dessus de l'espace d'information à la manière d'une loupe, comme c'est le cas avec la technique Document Lens [104] (métaphore d'interaction de la loupe).

Dans les évaluations menées et présentées dans les chapitres 4 et 5, nous avons utilisé alternativement les deux métaphores. Aucune d'elles ne semble meilleure que l'autre, et nous avons ainsi pu évaluer la perception et la compréhension que les utilisateurs ont ressenti sur chacune d'elles.

La métaphore d'interaction définit comment les tâches de navigation seront interprétées dans la vue bifocale. En effet les deux tâches génériques de navigation « Déplacer de » et « Aller à » définies dans la section précédente comme des modes de navigation (absolu ou relatif) sont traduites différemment dans la vue bifocale : mouvement des données ou mouvement de la zone de focus. La métaphore d'interaction appliquée pourrait alors s'apparenter au niveau pragmatique au sein du processus d'abstraction des événements captés aux mouvements dans la vue bifocale. La Figure 3.20 combine les trois niveaux considérés pour une modalité de navigation de la section précédente avec le niveau correspondant à la métaphore d'interaction.

Quelque soit la métaphore d'interaction appliquée, et comme expliqué par Appert & al. [5], pour naviguer dans une vue multi-échelle (deux échelles dans le cas d'une vue bifocale), deux besoins sont requis : un accès rapide aux données dans la zone de contexte et une interaction précise avec l'espace d'information dans la zone de détail. Pour cela il est possible soit de :

- Naviguer à l'échelle du focus, ce qui permet la navigation précise dans le focus mais ne permet pas l'accès rapide aux données du contexte ;
- Naviguer à l'échelle du contexte, ce qui permet l'accès rapide aux données du contexte mais ne permet pas la navigation précise dans le focus.

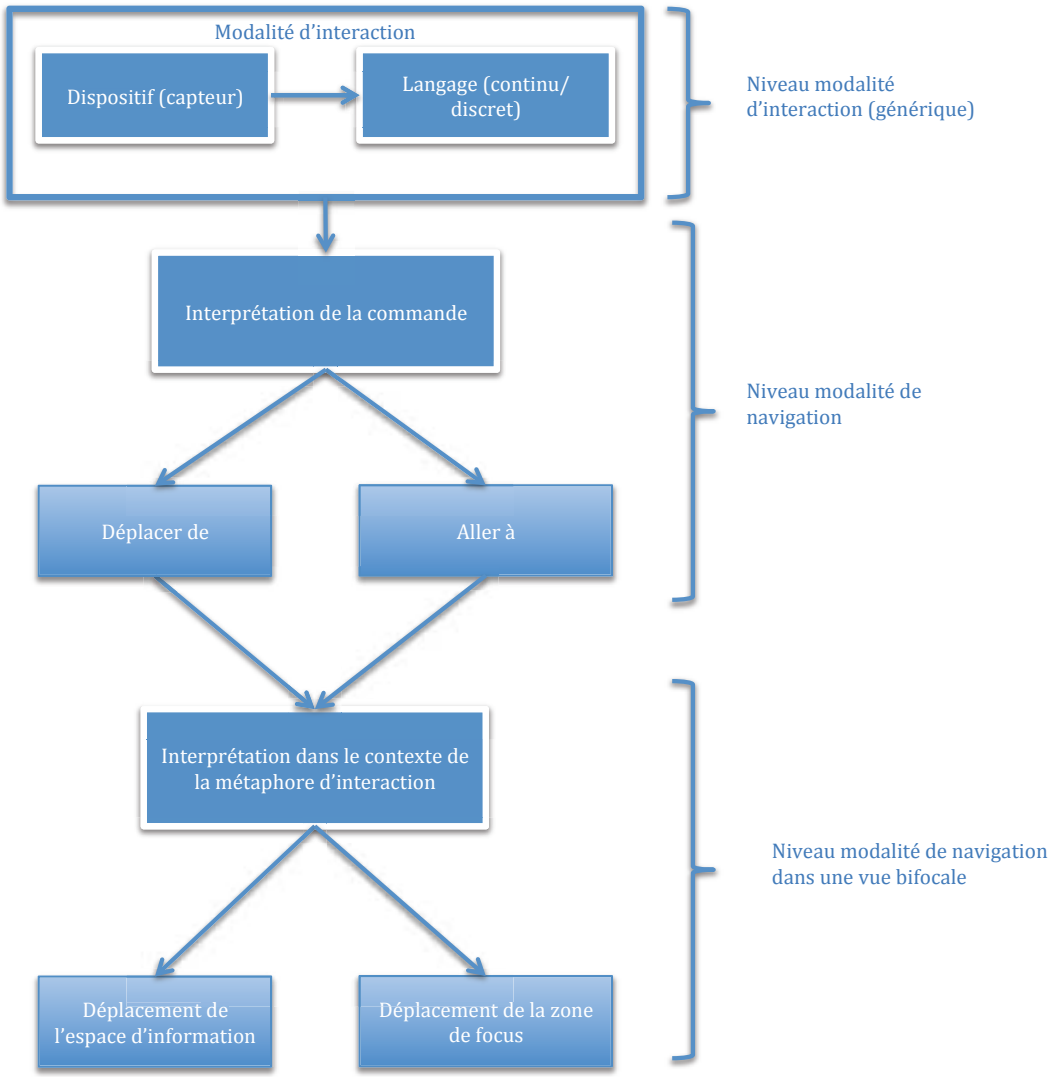


Figure 3.20 – Modalité de navigation dans une vue bifocale : des événements captés aux déplacements dans la vue bifocale.

Pour les deux métaphores d'interaction, un déplacement de un pixel à l'échelle du contexte entraînera un déplacement de l'espace d'information ou de la zone de focus de *fz* pixels (*fz* étant le facteur de zoom entre le focus et le contexte). Il sera alors très difficile d'effectuer des déplacements précis, mais parcourir de longues distances sera aisé. Au contraire, un déplacement de un pixel à l'échelle du focus entraînera un déplacement de l'espace d'information de un pixel pour la métaphore d'interaction du ruban, alors que pour la métaphore d'interaction de la loupe la zone de focus ne se déplacera pas. En effet, avec la métaphore d'interaction de la loupe, la zone de focus peut se déplacer sur une distance correspondant à la taille de l'écran. Ainsi, un déplacement de un pixel de la zone de focus correspond à un déplacement de *x* pixels à l'échelle du focus, où *x* est le rapport entre la taille de l'espace d'information et la taille de l'écran. Par exemple, si ce rapport vaut 100, la zone de focus se déplacera tous les 100 pixels déplacés à l'échelle du focus. Pour les deux métaphores d'interaction, naviguer à l'échelle du focus rendra la navigation précise aisée, mais parcourir de longues distances sera fastidieux.

Ce besoin est également présent pour le pointage à distance sur une cible numérique ou physique, où les utilisateurs doivent d'abord spécifier grossièrement une zone dans laquelle se trouve la cible, puis effectivement la pointer ensuite. Des études ont donc proposé des techniques « bi-mode » [34, 93] ou multimodales, combinant deux modalités de pointage : l'une absolue pour effectuer la phase de pointage grossière, et l'autre relative pour effectuer la phase de pointage précise.

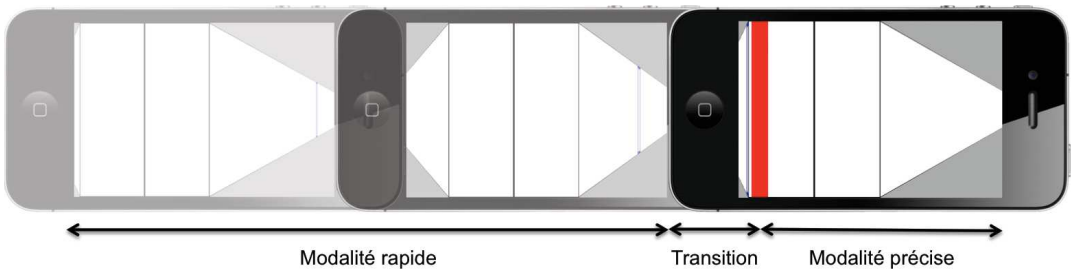


Figure 3.21 – Modèle conceptuel de navigation multimodale en trois phases.

Comme expliqué dans [34, 93], l'enjeu de conception au-delà du choix de deux modalités (une modalité - Figure 3.21 - pour chaque phase) réside alors dans la transition entre les phases de navigation rapide et précise. Ainsi, associer une modalité de navigation rapide à la phase de navigation rapide, et une modalité de navigation précise à la phase de navigation précise ne suffit pas. La transition entre ces deux modalités doit être fluide, rapide et facile à effectuer, autant d'un point de vue physique que cognitif. En se basant sur ce besoin et l'approche utilisée par les techniques bi-mode, nous avons décomposé la tâche de navigation dans une vue bifocale d'un point de vue multimodal en combinant deux modalités, et en étudiant leur complémentarité. Ainsi, la tâche peut être décomposée en trois phases (Figure 3.21) :

1. Navigation rapide (appelée *Focus-targeting* par Appert & al. [5], ou *Motion* par Gutwin [51]) : utilisation d'une première modalité permettant d'approcher la cible au plus près de la zone du focus.
2. Navigation précise (appelée *Cursor-pointing* par Appert & al. [5], ou *Acquisition* par Gutwin [51]) : utilisation d'une seconde modalité permettant d'effectivement mettre la cible à l'intérieur de la zone du focus.
3. Transition : Séquencement ou entrelacement des deux modalités.

En se référant aux propriétés CARE (Complémentarité, Assignation, Redondance et Equivalence) [32] définissant des relations entre les modalités mises en jeu dans une interaction multimodale, les deux modalités de navigation utilisées sont complémentaires.

Nous pouvons étudier au delà de la relation sémantique entre les deux modalités, leurs relations temporelles comme dans l'espace de conception de la multimodalité par Vernier & Nigay [127]. Comme le montre la Figure 3.22, les modalités peuvent être utilisées de manière anachronique, séquentielle ou concomitante (entrelacement des modalités).

Relation temporelle	Anachronisme	Séquencement	Concomitance

Figure 3.22 – Relations temporelles entre les deux modalités de navigation caractérisant la phase de transition de la Figure 3.21.

Ce modèle de navigation en trois phases est conceptuel car il est indépendant de toute modalité de navigation (du paragraphe 3.3). Ce modèle structure notre démarche de recherche : aussi dans la suite de nos travaux, nous nous intéressons donc à différentes combinaisons de modalités offrant de bonnes performances pour chacune des trois phases du modèle de navigation conceptuel présenté.

3.6 Conclusion

Dans ce chapitre consacré à l'interaction avec la vue bifocale décrite dans le chapitre précédent, nous avons d'abord décrit un panorama structuré des modalités de navigation sur dispositifs mobiles, après avoir listé les contraintes d'interaction inhérentes à ce type de dispositif.

Pour la conception de modalités de navigation avec la vue bifocale, nous avons ensuite considéré les besoins interactionnels qui sont propres à la navigation multi-échelle induite par la vue bifocale.

Nous avons enfin proposé un modèle simple de navigation multimodale en trois phases. Celui-ci est composé d'une première phase permettant de rapprocher la cible de la zone de focus en utilisant une première modalité dite rapide. Une deuxième phase permet de mettre la cible à l'intérieur du focus en utilisant une seconde modalité dite précise. La troisième phase est celle de transition entre les deux premières, permettant d'étudier de quelle manière se combinent les deux modalités utilisées : facilité et rapidité de changement de modalités.

La Figure 3.23 résume la démarche d'analyse adoptée qui a induit la structure de ce chapitre.

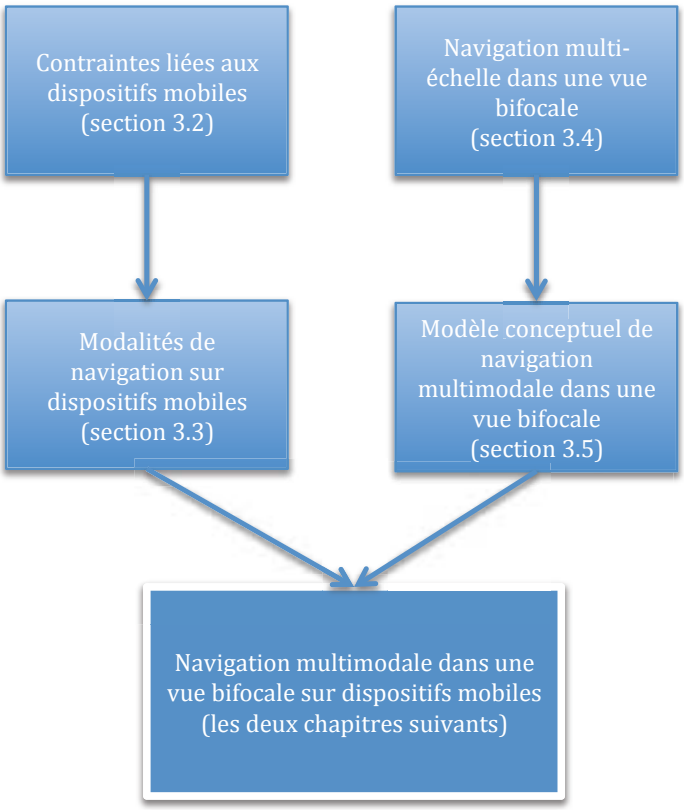


Figure 3.23 – Etude de l'interaction avec une vue bifocale sur dispositifs mobiles : démarche d'analyse relatée par la structure de ce chapitre.

Le modèle conceptuel de navigation multimodale définit notre démarche de recherche : en effet dans les deux chapitres suivants, nous allons instancier ce modèle avec différents couples de modalités (comme schématisé à la Figure 3.21), et les évaluer sur notre vue bifocale afin d'étudier les performances globales mais aussi les performances pour chaque phase du modèle.

4 NAVIGATION MULTIMODALE DANS UNE VUE BIFOCALE

4.1 Introduction

En nous basant sur le modèle en trois phases décrit dans le chapitre précédent, nous avons conçu et évalué un ensemble de modalités combinées pour naviguer dans une vue bifocale : une modalité pour amener rapidement la cible à proximité de la zone de focus, et une modalité pour mettre la cible à l'intérieur de la zone de focus.

L'espace des possibilités est très vaste si nous nous référons à l'ensemble des modalités de navigation décrites dans le chapitre précédent (section 3.3 du chapitre 3), et d'autant plus si nous considérons l'ensemble de leurs combinaisons deux à deux. L'exploration systématique d'un tel espace de possibilités est donc ambitieuse. Face à ce défi, notre approche a été expérimentale et incrémentale. Ce chapitre et le suivant décrivent notre exploration, certes partielle, de ce vaste espace de possibilités.

Dans ce chapitre structuré en deux parties, nous décrivons une étude pilote et deux expériences. Pour chaque expérience conduite, nous avons considéré comme modalité de référence la modalité purement tactile.

1. Nous avons d'abord vérifié le bien-fondé du modèle conceptuel en trois phases aux besoins distincts en menant une étude pilote. Nous avons considéré trois modalités pures (au sens non combinées) : l'inclinaison du dispositif (*Tilt,* section 3.3.3.2 du chapitre 3), le déplacement du dispositif (*Interaction physique*, section 3.3.3.1 du chapitre 3) et les gestes tactiles (*Flick*, section 3.3.2 du chapitre 3). La comparaison des performances a permis de conclure que ces trois modalités ne permettent pas de fournir à la fois la rapidité et la précision requises pour chacune des deux phases de navigation (section 4.2.2).

2. Nous avons alors considéré la combinaison de modalités selon notre modèle de navigation multimodale : nous avons repris les deux modalités *Tilt* et *Interaction physique* de l'étude pilote que nous avons combinées avec la modalité tactile *Flick* pour la phase précise. Les résultats expérimentaux sont encourageants (section 4.2.3) car les modalités combinées réduisent l'occultation de l'écran lors de la phase où l'utilisateur parcourt une grande partie de l'espace de l'information. Néanmoins les performances avec les modalités combinées ne sont pas meilleures qu'avec la modalité purement tactile (section 4.2.3).
3. Nous avons enfin mené une deuxième expérience en analysant en détail les performances pour chaque phase de notre modèle. Dans cette deuxième expérience, nous avons ajouté au panel des modalités, la modalité reposant sur des capteurs de pression que nous avons là encore combinée avec la modalité tactile (section 4.3).

4.2 Etude pilote et Expérience 1 : bien-fondé du modèle conceptuel en trois phases

Pour mener les expériences, nous avons utilisé le prototype de notre vue bifocale (section 2.3.1 du chapitre 2) développé sur un iPhone 4 avec une taille d'écran de 960x640 pixels en mode paysage. Le programme était exclusivement codé sur la plateforme iOS.

La taille de la zone de focus était de 500 pixels de largeur, soit un peu plus de la moitié de l'écran. La taille de l'espace d'information était de 24 000 pixels à l'échelle du focus. L'espace alloué au contexte était de 2 400 pixels de largeur au total. Ainsi, le facteur de zoom utilisé entre le focus et le contexte était de 10.

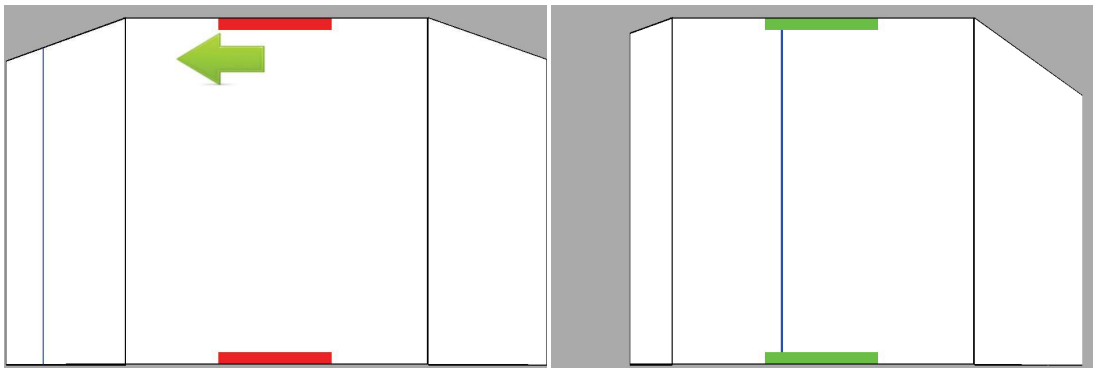
Une zone de contexte (une des deux vues en perspective) permettait de représenter entre 0 et 23 500 pixels (selon qu'elle comportait aucune valeur ou bien le maximum de valeurs). L'espace alloué pour cette zone de contexte étant de 2400 pixels, en allouant une valeur par pixel, toutes les valeurs de l'espace d'information n'étaient donc pas représentées à l'écran lorsque leur nombre dépassait ce seuil. Le contexte donnait donc une vue condensée de l'espace d'information.

Dans ce contexte, nous avons considéré la tâche de navigation décrite dans le paragraphe suivant.

4.2.1 Tâche de navigation

Pour l'étude pilote et l'expérience 1 décrites ci-dessous, nous avons utilisé une tâche de navigation identique avec la métaphore d'interaction du ruban (déplacement de l'espace d'information). Nous avons réutilisé le protocole expérimental et la tâche décrite par Appert & al. [3] pour la navigation dans un large espace d'information. Nous avons utilisé la métaphore d'interaction du ruban. L'objectif pour les participants était de déplacer une cible affichée dans la zone de contexte, vers une zone désignée par deux traits rouges dans la zone du focus (Figure 4.1.a). Une fois la cible dans cette zone, les deux traits de la zone étaient affichés en vert (Figure 4.1.b). La cible devait alors y être maintenue

pendant au moins une seconde afin que la tâche soit validée. Une fois la validation effectuée, une nouvelle cible était proposée.



(a) **(b)**

Figure 4.1 – (a) Début de la tâche : la cible est indiquée dans le contexte par un trait bleu. La flèche indique la direction de la position de la cible. (b) Fin de la tâche : la cible est positionnée dans la zone désignée du focus. La zone est alors affichée en vert.

L'effet de perspective rendant la visualisation de la cible difficile lorsque celle-ci se trouve loin de la zone de focus, un facteur de zoom de 3 a été appliqué afin qu'elle soit toujours visible peu importe sa position. Ainsi, les cibles étaient représentées par des traits bleus de 4 pixels de largeur dans la zone de focus, et par des traits bleus de 12 pixels de largeur dans la zone du contexte (Figure 4.1). Toujours dans le but d'aider l'utilisateur à repérer la cible au début et pendant toute la tâche, une flèche verte indiquait la direction vers laquelle se trouve la cible (Figure 4.1.a). Cette flèche évite que l'utilisateur se retrouve perdu dans l'espace d'information et navigue ainsi dans la mauvaise direction (phénomène de « desert fog » décrit par Jul & Furnas [63]).

La zone dans laquelle devait être déposée la cible faisait 200 pixels de largeur. Cette zone désignée par deux traits dans le focus était donc relativement grande. En effet dans le cas d'une tâche concrète, il convient d'amener la donnée qui intéresse l'utilisateur n'importe où dans le focus pour obtenir sa valeur détaillée. La tâche considérée n'était donc pas une tâche de pointage précis. Nous ne voulions pas que la difficulté de la tâche réside dans la fin de la tâche consistant à mettre la cible dans la zone de validation.

Nous avons considéré quatre distances différentes pour positionner les cibles, réparties équitablement dans la zone du contexte à gauche et à droite du focus. Ces distances correspondent à quatre indices de difficulté (ID) différents (2 / 5.3 / 6.3 / 7) données par la loi de Fitts [41] : ID = log(1+D/W), où D est la distance jusqu'à la cible, et W est sa taille (correspondant à la taille de la zone dans laquelle déposer la cible dans notre cas donc 200 pixels de largeur).

4.2.2 Etude pilote

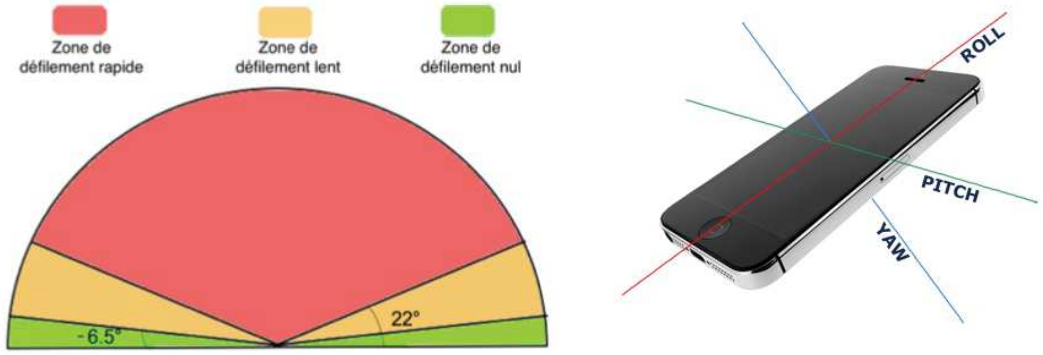
Nous avons réalisé une étude pilote avec 12 participants pour vérifier les deux phases de navigation de notre modèle conceptuel pour naviguer dans notre vue

bifocale (une phase de navigation rapide et une phase de navigation précise). Notre hypothèse est que l'usage d'une modalité pure (non combinée) rendrait difficile l'une des deux phases de la tâche.
Pour cela et comme indiqué dans l'introduction, nous avons choisi de comparer une modalité tactile basée sur les gestes de *Drag/Flick* (appelée *Flick* par la suite) connus de tous les utilisateurs de dispositifs mobiles avec deux autres modalités : une basée sur le geste de *Tilt* (nommée *Tilt* par la suite), l'autre utilisant les déplacements du dispositif (nommée *Interaction physique* par la suite). Comme énoncé dans le chapitre précédent (section 3.3.3) :
Ces deux modalités permettent d'éviter les problèmes d'occultation de l'écran créés par la modalité *Flick*.
Ces deux modalités ont déjà été comparées à la modalité tactile (par Fitchett & Cockburn [39] pour *Tilt*, et par Spindler & al. [122] pour *Interaction physique*). Même si le contexte d'application n'était pas le même, elles ont obtenu de bonnes performances qui nous ont incités à les sélectionner pour cette étude.

De plus ces deux modalités n'utilisent pas le même mode de navigation (section 3.3 du chapitre 3). En effet, alors que *Tilt* utilise un mode de navigation relatif basé sur la vitesse (Figure 3.5.a), notre implémentation de *Interaction physique* permet un mode de navigation absolu associant une position du dispositif mobile à une position dans l'espace d'information (Figure 3.5.b). Ainsi avec *Tilt*, les utilisateurs déroulent le ruban de l'espace d'information pour le parcourir, alors qu'avec *Interaction physique*, ils spécifient une position dans l'espace d'information en positionnant le dispositif dans l'espace.

Nous détaillons l'implémentation de ces deux modalités ci-dessous.

4.2.2.1 Tilt



(a) **(b)**

Figure 4.2 – (a) Modalité *Tilt*. (b) Axes de rotation du dispositif.

Cette modalité permet d'effectuer les deux phases de navigation (rapide et précise) avec des gestes de *Tilt* uniquement. La modalité repose sur un mode de navigation relatif et utilise donc une navigation par vitesse (section 3.3.3.2 du chapitre 3). Nous avons choisi cette modalité car elle a été comparée à la modalité *Flick* habituellement utilisée pour naviguer sur un dispositif mobile [39], et a obtenu de bons résultats. Néanmoins dans l'expérience décrite dans [39], les distances évaluées étaient moins longues, donc le besoin de rapidité moins important. Le

contexte d'utilisation (distances plus longues) de notre étude peut ainsi conduire à des résultats différents.

Pour développer la modalité *Tilt*, nous avons utilisé l'accéléromètre et le gyroscope intégrés au dispositif. En effet, le SDK iOS utilise des algorithmes de fusion permettant d'affiner les valeurs relatives aux mouvements du dispostif en utilisant les différents capteurs. Nous avons pour cela utilisé les angles de position du dispositif avec la formule suivante :
tiltAngle = asin(2(quat.y*quat.z - quat.w*quat.x))* où *quat* est le quaternion représentant la position du dispositif mobile selon ses trois axes de rotation (Figure 4.2.b).
tiltAngle représente ainsi l'angle selon l'axe Pitch (Figure 4.2.b). Nous avons appliqué une conversion linéaire à l'angle d'inclinaison pour obtenir une plage de vitesse comprise entre 0 et 3 000 pixels par seconde en considérant un angle d'inclinaison maximum de 60°. Ainsi, incliner le dispositif vers la gauche (respectivement vers la droite) déplaçait l'espace d'information vers la gauche (respectivement vers la droite) à l'image d'un ruban qui se déroule du côté où le dispostif est incliné.

Pour augmenter la précision lors de la navigation, nous avons réduit la vitesse de défilement par 3 lorsque l'angle d'inclinaison est compris entre -22° et 22° par rapport à la position de référence (Figure 4.2.a). La position de référence est celle du dispositif au lancement de l'application. Cette solution permet ainsi des vitesses comprises entre 0 et 800 pixels par seconde jusqu'à 22°. Ce seuil a été défini de façon expérimentale. Nous avons demandé à 4 utilisateurs d'utiliser la modalité *Tilt* de la même manière que pendant l'étude, et de nous dire à haute voix à partir de quel seuil ils aimeraient voir la vitesse diminuer pour gagner en précision. L'angle d'inclinaison du dispositif affiché en direct sur un écran nous permettait de connaître la valeur désirée par l'utilisateur. La valeur de l'angle +/-22° est la moyenne des valeurs observées. Ceci permet d'avoir deux niveaux de vitesse, et améliore ainsi la précision lorsque l'inclinaison du dispositif n'est pas très importante.

Plutôt que de proposer une activation sur demande de la modalité *Tilt*, qui implique une action supplémentaire de l'utilisateur, nous avons défini comme dans [39] une zone dans laquelle les mouvements n'entraînent pas de défilement (Figure 4.2.a). Cette zone est comprise entre -6.5° et + 6.5° (Figure 4.2.a). Ceci a également l'avantage d'augmenter la stabilité en filtrant toutes les inclinaisons non volontaires du dispositif.

4.2.2.2 Interaction physique

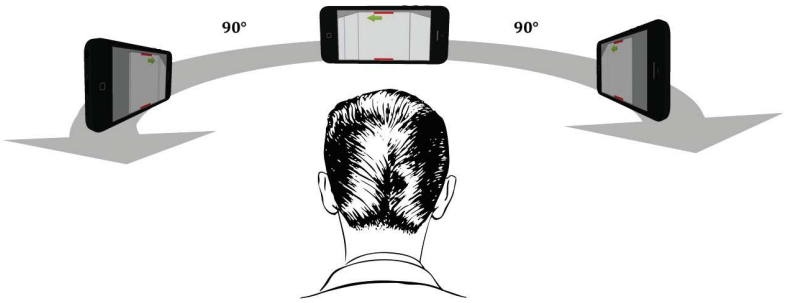


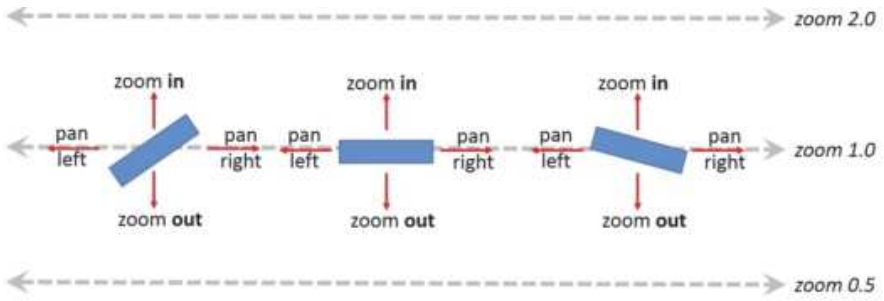
Figure 4.3 – Modalité *Interaction physique*.

A l'inverse de *Tilt*, cette modalité considère le dispositif comme une fenêtre d'affichage que l'on déplace sur un ruban de données. L'utilisateur a donc la possibilité de déplacer physiquement le dispositif dans l'espace le positionner dans l'espace d'information (Figure 4.3). Cette modalité permet alors un mode de navigation absolu à des données en positionnant le dispositif lui-même sur un ruban virtuel placé devant l'utilisateur. Pour éviter des déplacements non voulus lors de l'interaction ou en situation de mobilité, la modalité *Interaction physique* n'est activée que lorsque l'utilisateur maintient un doigt appuyé (quasi-mode comme dans [122]) dans la zone du contexte de la vue bifocale. Ainsi l'activation sur demande de la modalité *Interaction physique* permet à l'utilisateur d'avoir une position stable dans l'espace d'information lorsque la modalité n'est pas activée. Du fait qu'il y a une correspondance entre la position physique du dispositif mobile et la position virtuelle dans l'espace d'information, nous n'avons pas utilisé une zone dans laquelle aucun défilement n'est déclenché, comme fait pour la modalité *Tilt* décrite ci-dessus. Cela aurait créé un décalage dans la correspondance entre la position physique du dispositif et la position virtuelle dans les données lors de chaque arrêt : la modalité aurait alors été moins précise et intuitive.

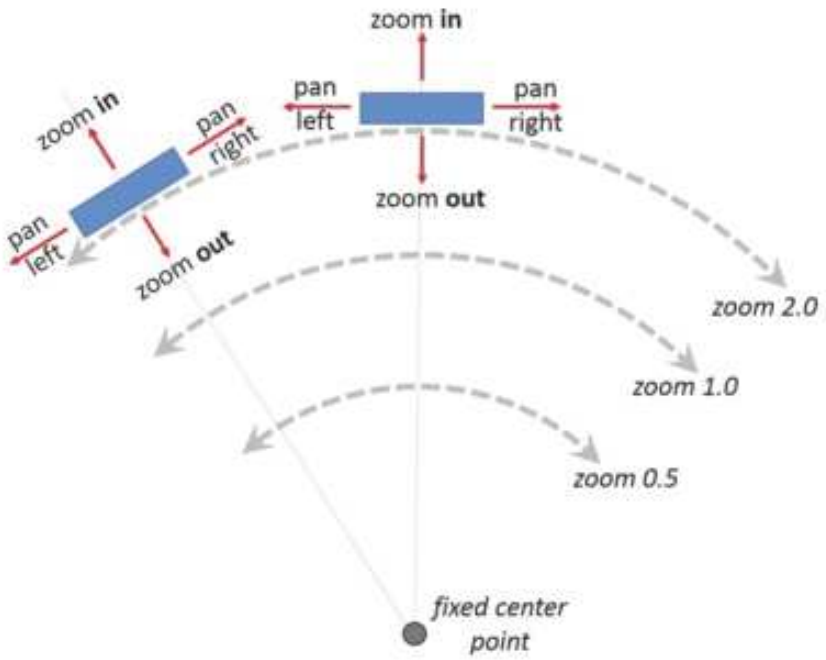
Pour la mise en œuvre de la modalité *Interaction physique* nous avons utilisé le gyroscope intégré aux dispositifs mobiles, bien que des systèmes de positionnement plus précis existent (par exemple, OptiTrack[8]). En effet nous voulions utiliser la technologie actuellement disponible dans les dispositifs mobiles. Les accéléromètres permettent de détecter des translations grâce à l'accélération du dispositif en effectuant une double intégration de celle-ci sur le temps. Or, cette méthode est peu fiable car très imprécise. En effet, la double intégration accentue la moindre erreur de mesure retournée par les capteurs, comme reporté par Yun & al. [135]. C'est encore plus vrai dans notre cas où les mouvements sont de faible amplitude (moins d'un mètre).
Pahud et al. [97] ont présenté trois fonctions pour associer une position physique à une position dans l'espace d'information : une association plane fixe, une association sphérique fixe et une association dynamique (Figure 4.4). Nous avons choisi une association sphérique (Figure 4.3). Elle est décrite par Pahud et al. [97] comme difficile à utiliser lorsque la manipulation spatiale permet aux utilisateurs

[8] https://www.optitrack.com

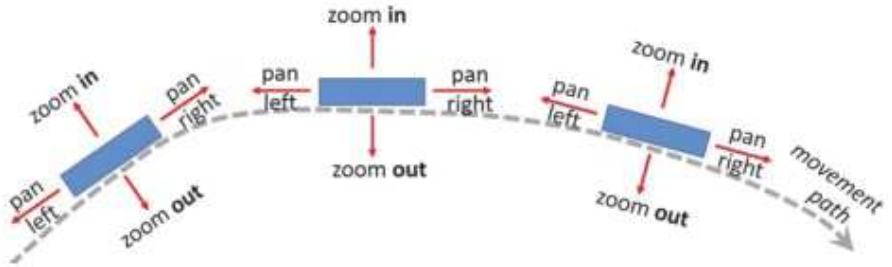
de zoomer en déplaçant le dispositif mobile plus ou moins proche du corps : ce n'est pas notre cas car nous n'avons pas de fonction de zoom. De plus, une association sphérique nous permet de mettre en œuvre la modalité sans un système externe de positionnement précis (par exemple, Optitrack) : en effet, nous avons utilisé le gyroscope du dispositif pour détecter sa rotation. Pour cela, nous avons explicitement demandé aux utilisateurs de décrire un arc de cercle devant eux pour parcourir l'ensemble de l'espace d'information (Figure 4.3). Or, comme rapporté par Pahud & al. [97], les utilisateurs ont naturellement tendance à déplacer les bras de gauche à droite de manière circulaire, ce qui n'impose donc pas un mouvement inconfortable ou non spontané aux utilisateurs.



(a)



(b)



(c)

Figure 4.4 – Modalité par déplacements du dispositif : fonctions d'association : (a) Association linéaire (b) Association sphérique (c) Association dynamique. Issues de [97].

La mise en œuvre de la modalité *Interaction physique* utilise donc l'angle de rotation autour de l'axe Pitch (Figure 4.2.b) et permet de parcourir entièrement l'espace d'information avec un angle de 180°, 90° vers la gauche, 90° vers la droite (Figure 4.3). Comme pour la modalité *Tilt*, la position initiale a été fixée au lancement de l'application. Cette position est associée au milieu de l'espace d'information.

4.2.2.3 Procédure et résultats

Nous avons demandé à 12 utilisateurs d'effectuer la tâche de navigation décrite précédemment en utilisant trois modalités différentes : les modalités *Tilt* et *Interaction physique* décrites ci-dessus, ainsi que la modalité *Flick*. Cette dernière modalité agit sur l'espace focus, cela signifie qu'un déplacement du doigt de un pixel à l'écran entraîne un déplacement de un pixel dans l'espace d'information à l'échelle du focus. La mise en œuvre de cette modalité était celle existante du SDK iOS. Les gestes de *Drag/Flick* devaient avoir lieu dans la zone de focus. L'objectif était ici de renforcer la métaphore d'interaction du ruban. Cela s'apparente donc à faire défiler le ruban de l'espace d'information avec le doigt qui est la métaphore d'interaction la plus couramment utilisée sur dispositifs mobiles.

Les participants devaient tous réaliser trois blocs de tâches, soit un bloc par modalité. Un bloc comprenait 24 cibles avec les quatre indices de difficulté décrits à la section 4.2.1, soit 6 cibles par indice de difficulté. La consigne donnée était d'être aussi rapide que possible.

Comme attendu, cette étude a montré les limitations en terme de précision pour les deux modalités *Tilt* et *Interaction physique*.

- En effet, les participants ont eu beaucoup de mal à effectuer la phase précise de la tâche, même avec la modalité *Tilt* qui comporte deux niveaux de vitesse.
- Quant à la modalité *Interaction physique*, les tremblements des mains provoquaient de nombreux mouvements non désirés par les utilisateurs, rendant très compliquée la phase de navigation précise. L'utilisation d'un filtre comme celui conçu par Casiez & al. [24] aurait amélioré ce problème, mais une difficulté aurait persisté due à la correspondance déplacement

physique / déplacement virtuel. En effet, un déplacement du dispositif de un pixel dans l'espace déplaçant plus de un pixel dans l'espace d'information, la précision restait donc limitée.

En terme de performance, le temps moyen d'exécution de la tâche était deux à trois plus élevés avec les modalités *Tilt* et *Interaction physique*, qu'avec la modalité *Flick*.

Les utilisateurs ont cependant commenté le fait que les deux modalités *Tilt* et *Interaction physique* semblaient efficaces durant la phase de navigation rapide.

Concernant la modalité *Flick*, le constat fût l'inverse :

- Pour la navigation s'effectuant dans l'espace focus, les participants ont trouvé la phase de navigation rapide laborieuse car nécessitant beaucoup de gestes (clutching).
- Au contraire la modalité tactile leur a permis d'être assez précis pour effectuer la seconde phase efficacement.

Les résultats de cette étude pilote ont confirmé le bien-fondé de notre modèle conceptuel de navigation en termes de phases aux besoins distincts, respectivement de la rapidité et de la précision. Pour les trois modalités pures testées, ces besoins ont été soulignés par les participants.

Dans l'expérience suivante nous avons donc considéré la combinaison de deux modalités : une pour la phase de navigation rapide, une pour la phase de navigation précise. Pour cela nous avons repris les deux modalités *Tilt* et *Interaction physique* développées pour cette étude pilote pour les combiner avec la modalité *Flick*.

4.2.3 Expérience 1

Comme confirmé par l'étude pilote, la navigation dans une vue bifocale nécessite à la fois de la rapidité pour atteindre les cibles les plus éloignées, et de la précision pour positionner les cibles à l'intérieur de la zone de focus. L'enjeu de conception selon notre modèle conceptuel est donc :

- d'assigner à chacune des deux phases une modalité adéquate aux besoins (rapidité et précision), mais aussi
- de permettre un enchaînement rapide entre l'usage séquentiel des deux modalités, voire permettre un usage entrelacé [127] des deux modalités. L'objectif est ici de ne pas perdre le bénéfice des deux modalités en coût de changement.

Nous avons donc considéré deux modalités combinées que sont les modalités *Tilt* et *Interaction physique* couplées à la modalité tactile *Flick :* (*Tilt+Flick*) et (*Interaction physique+Flick*). Ainsi pour les deux modalités combinées, nous avons associé une modalité rapide avec une modalité précise, selon notre modèle de navigation multimodale. Un autre point de vue centré sur la modalité tactile permet de conclure que ces deux modalités combinées sont des extensions de la modalité tactile pour autoriser une navigation rapide.

Chacune des trois modalités Flick, (Tilt+Flick) et (Interaction physique+Flick)

agit sur l'espace focus. Cela signifie qu'un déplacement d'un pixel (peu importe la modalité utilisée), entraîne un déplacement d'un pixel à l'échelle du focus.

Comme lors de l'étude pilote, la mise en œuvre de la modalité *Flick*, qu'elle soit utilisée seule ou en combinaison avec une autre modalité, est celle existante du SDK iOS. L'utilisateur doit débuter son geste de *Drag/Flick* dans le focus. Nous avons fait ce choix pour renforcer la métaphore d'interaction du ruban : lors de l'interaction dans la zone de focus, les utilisateurs déplacent l'ensemble de l'espace d'information (glissement du ruban) plutôt que la zone de focus elle-même. Par conséquent, les données restent sous le doigt d'utilisateur.

Pour la modalité combinée (*Tilt+Flick*), les deux modalités reposent sur un mode de navigation similaire par accès relatif. Les deux modalités *Tilt* et *Flick* sont actives à tout instant et peuvent être utilisées en parallèle autorisant ainsi un usage entrelacé des deux modalités. L'implémentation de la modalité *Tilt* est la même que celle de l'étude pilote.

Au contraire, pour la modalité combinée (*Interaction physique+Flick*), les deux modalités qui sont combinées offrent des modes de navigation différents : accès absolu pour *Interaction physique* et accès relatif pour *Flick.* Ces différences de mode de navigation peuvent rendre la phase de transition plus difficile pour l'utilisateur. Il convient néanmoins de noter que contrairement à la modalité (*Tilt +Flick*), l'utilisateur ne peut pas utiliser les deux modalités *Interaction physique* et *Flick* en parallèle. En effet *Interaction physique* nécessite d'être activée en touchant l'écran. L'implémentation de la modalité *Interaction physique* est la même que celle de l'étude pilote.

Nous avons mené une évaluation contrôlée pour comparer les performances des trois modalités : *Flick*, (*Tilt+Flick*) et (*Interaction physique+Flick*) pour naviguer dans une vue bifocale. Nous avons formulé deux hypothèses :

- H1 : Les deux modalités combinées (*Tilt+Flick*) et (*Interaction physique+Flick*) sont plus performantes que la modalité *Flick*. En effet, ces modalités bénéficient de la précision de la modalité *Flick* pour la phase de navigation précise, et d'une plus grande rapidité pour la phase de navigation rapide.
- H2 : Les trois modalités sont équivalentes pour les cibles peu éloignées. En effet, nous nous attendons à ce que pour ces cibles proches du focus, les utilisateurs n'utilisent que la modalité *Flick* puisque le besoin de précision est plus important que le besoin rapidité.

4.2.3.1 Participants, tâche et procédure

Nous avons mené l'expérimentation avec 12 bénévoles (en dehors de notre laboratoire), tous ingénieurs professionnels en informatique, âgés de 30 ans en moyenne. Tous étaient des utilisateurs réguliers de dispositifs mobiles tactiles. Les participants étaient les mêmes que ceux de l'étude pilote et nous rappelons qu'ils avaient souligné les limites de précision des deux modalités pures *Tilt* et *Interaction physique*. En sollicitant à nouveau les mêmes participants, nous avons ainsi minimisé la différence d'expertise entre la modalité *Flick* couramment utilisée et les deux modalités *Tilt* et *Interaction physique* nouvelles pour les participants.

La tâche à réaliser (section 4.2.1) était la même que celle de l'étude pilote. L'ordre d'utilisation des modalités a été contrebalancé par un carré latin.

En début de session, après une brève démonstration des trois modalités, les participants se sont entrainés jusqu'à ce qu'ils se sentent à l'aise avec chaque modalité (3-4 minutes en moyenne). Ensuite, pour chaque modalité, les participants ont effectué 9 tâches d'entraînement avant de commencer l'évaluation. Parce que tous les participants sont experts avec la modalité *Flick*, trois fois moins de tâches d'entraînement étaient demandées avec cette modalité (3 tâches seulement).
La consigne donnée aux 12 participants était d'être aussi rapide que possible. Pour chaque tâche (cible), nous avons mesuré le temps de réalisation. Après avoir effectué les 24 tâches de pointage (24 cibles) pour une modalité, les participants ont répondu à un questionnaire SUS [17] (Annexe 2) pour recueillir leur point de vue subjectif sur la modalité utilisée. Enfin il était demandé aux participants de classer les trois modalités par ordre de préférence et d'expliquer leurs choix à la fin de la session. Une session a duré en moyenne 20 minutes par participant.

4.2.3.2 Résultats

4.2.3.2.1 Temps d'exécution

La Figure 4.5 présente les temps moyens d'exécution pour chaque modalité et chaque indice de difficulté. Nous avons vérifié la normalité des données avec le test de Shapiro-Wilk. Le test a révélé un petit écart par rapport à la distribution normale. Après normalisation par une transformation logarithmique, le test a révélé la normalité des données ($W = 0.9875$, $p = 0.2188$). Le test de Levene n'a pas signalé une violation de l'homogénéité des variances entre les conditions ($F_{11,132} = 0.4278$, $p = 0.9414$).

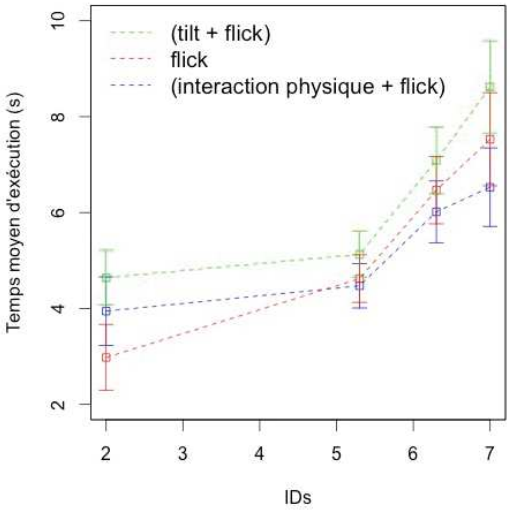


Figure 4.5 – Temps moyens d'exécution par modalité et ID avec intervalle de confiance de 95%.

Par conséquent, nous avons effectué une analyse de variance (ANOVA) à deux facteurs (Modalité et ID) sur le temps d'exécution. L'analyse a montré un effet

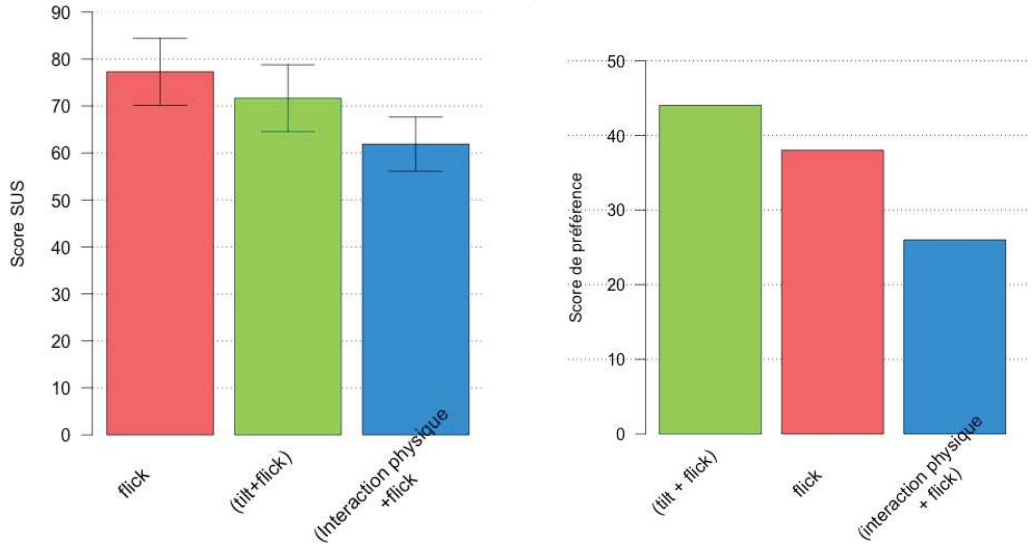
significatif des facteurs Modalité ($F_{2,132}$ = 13.59, p < 0.0001) et ID ($F_{3,132}$ = 86.32, p < 0.0001).
L'interaction entre Modalité x ID est également significative ($F_{6,132}$ = 3,06, p < 0.01). Un test post-hoc de Tukey a révélé une différence significative entre (*Tilt+Flick*) et *Flick* (p < 0.001) et entre (*Tilt+Flick*) et (*Interaction physique+Flick*) (p < 0.001).

Nous avons également effectué une analyse de variance (ANOVA) à un seul facteur Modalité en effectuant des tests post-hoc de Tukey pour les IDs :

- Pour ID = 7, les résultats ont montré que (*Interaction physique+Flick*) est plus rapide que (*Tilt+Flick*) (p < 0.01). Par contre, il n'y a pas de différence significative entre *Flick* et (*Interaction physique+Flick*) ainsi qu'entre *Flick* et (*Tilt+Flick*).
- Pour ID = 5.3 et ID = 6.3, nous n'avons observé aucun effet significatif (respectivement $F_{2,33}$ = 2.23, p = 0.124 et $F_{2,33}$ = 1.65, p = 0.207).
- Pour ID = 2, nous avons trouvé un effet significatif principal ($F_{2,33}$ = 11.37, p < 0.001). Le test post-hoc de Tukey a révélé que *Flick* est plus rapide que (*Tilt+Flick*) et que (*Interaction physique+Flick*) (respectivement p < 0.001 et p < 0.01).

4.2.3.2.2 Utilisabilité subjective et préférences

L'utilisabilité subjective et les préférences des participants diffèrent des résultats de performance. À la Figure 4.6.a, les modalités sont ordonnées de gauche à droite par ordre décroissant de score moyen SUS. Les participants ont classé la modalité *Flick* comme la plus utilisable (Figure 4.6.a).



(a) **(b)**

Figure 4.6 – (a) Les scores SUS moyens avec un intervalle de confiance correspondant de 95% (b) Classement par préférence des trois modalités.

Néanmoins les scores SUS (Figure 4.6.a) sont en contradiction avec les préférences (Figure 4.6.b). Pour calculer le score global de préférences à la Figure 4.6.b pour chaque modalité, nous avons appliqué la formule suivante : 5x1^{er} +3x2^{e} + 3^{e} où 1er, 2e, et 3e correspondent au nombre de fois où la modalité à été classée au rang

correspondant. Nous avons vérifié que des coefficients inférieurs ({3, 2, 1} au lieu de {5, 3, 1} ci-dessus), donnent des résultats comparables. Les participants préfèrent (*Tilt+Flick*) alors que cette modalité est observée moins rapide que (*Interaction physique+Flick*) et *Flick*. De plus malgré les bonnes performances de (*Interaction physique+Flick*), cette modalité est classée en dernier en termes de scores SUS et de préférences.

4.2.3.2.3 Discussion

Les différences de performance n'étant pas significatives pour les IDs 5.3, 6.3 et 7, nous ne pouvons pas valider l'hypothèse H1. Nous n'observons donc pas de gain de performance significatif pour les cibles les plus éloignées. Il est possible qu'en considérant des distances encore plus importantes l'écart entre la modalité *Flick* et la modalité combinée (*Interaction physique+Flick*) continue à se creuser, comme le montre la tendance de la Figure 4.5. Pour vérifier cette assertion, une autre expérience serait nécessaire.

Nous notons que la modalité *Flick* étant très connue des utilisateurs il est intéressant d'obtenir des résultats comparables avec les deux modalités combinées. Cela est donc prometteur : avec des performances semblables, ces modalités combinées permettent de résoudre en partie les problèmes liés à l'interaction tactile sur mobile : occultation et réduction du clutching.

Il est également pertinent de noter que la modalité combinée (*Interaction physique+Flick*) n'a pas été utilisée en mode de navigation absolu comme nous le pensions, mais en mode de navigation relatif. En effet les utilisateurs faisaient défiler l'espace d'information à l'écran en bougeant le dispositif tout en maintenant leur doigt appuyé dans la zone du contexte. Or nous avions conçu cette modalité pour un mode de navigation absolu : les utilisateurs positionnent alors le dispositif directement à l'endroit souhaité dans l'espace puis appuient dans la zone de contexte pour mettre à jour l'affichage (comme pour prendre une photographie : cadrage puis prise de photo). Ce mode de navigation absolu aurait permis un gain de performance, surtout pour les distances longues. Il est possible qu'avec plus d'expérience et une meilleure connaissance de l'espace d'information, les utilisateurs profitent de cette possibilité, qui n'est pas actuellement possible sur les dispositifs mobiles. Cette remarque nécessiterait une expérience plus longue focalisant sur cette modalité *Interaction physique*, qui n'était pas l'objectif principal de cette expérience.

De plus pour les cibles proches et, contrairement à ce que nous attendions, les trois modalités ne sont pas équivalentes. H2 n'est pas vérifiée. En effet nous pensions que les utilisateurs allaient utiliser uniquement la modalité *Flick* même dans le cas des deux modalités combinées (*Tilt+Flick*) et (*Interaction physique+Flick*). Ce ne fut pas le comportement observé. Nous pensons que le fait que les tâches soient chronométrées a incité les participants à utiliser la modalité la plus rapide même si cela n'est pas le plus efficace. Cet effet peut disparaître après un temps d'apprentissage plus long. Utiliser uniquement la modalité *Flick* pour les cibles proches montrerait alors que les utilisateurs exploitent pleinement les capacités des deux modalités combinées (*Tilt+Flick*) et (*Interaction physique+Flick*).

Finalement, les temps d'exécution sont en contradiction avec les préférences des participants qui ont choisi (*Tilt+Flick*) en premier dans les préférences, en

indiquant qu'ils estimaient avoir été plus rapides avec cette modalité. Cette fausse perception de rapidité traduit sans doute une préférence dans l'utilisation de la modalité. De plus, malgré les bons résultats en temps d'exécution de la modalité (*Interaction physique+Flick*), celle-ci a obtenu un score SUS le plus faible et est la modalité la moins appréciée des participants. Les mouvements du dispositif dans l'espace ont été qualifiés de contraignants par les participants. L'aspect répétitif de l'expérience explique peut-être cette remarque, et une application réelle en contexte pourrait changer l'opinion des utilisateurs. Néanmoins, certains participants ont mentionné explicitement la question de l'acceptabilité sociale de cette modalité qui implique des mouvements du dispositif dans l'espace.

4.2.4 Conclusion

Pour naviguer efficacement dans la vue bifocale, nous avons étudié expérimentalement deux modalités combinées qui visent à améliorer la modalité classique tactile par des gestes *Flick* : (*Tilt+Flick*) et (*Interaction physique+Flick*). Une étude pilote nous a montré l'intérêt d'utiliser ce type de modalité pour bénéficier à la fois de la rapidité et de la précision de chacune des modalités combinées dans le but d'accomplir les deux phases de navigation de notre modèle de façon efficace.
L'expérience comparant les performances des trois modalités Flick, (*Tilt+Flick*) et (*Interaction physique+Flick*) a montré des résultats prometteurs de performance ou de préférence pour ces deux modalités combinées : incliner ou déplacer le dispositif dans l'espace pour une navigation rapide mais imprécise, et *Flick* pour une navigation précise mais lente. En particulier les résultats de performance de (*Interaction physique+Flick*) confirment et complètent ceux de Spindler et al. [24]. Néanmoins, malgré les bonnes performances de cette modalité, les résultats ont montré que les participants ne sont pas prêts à effectuer des mouvements physiques dans l'espace pour interagir avec leurs dispositifs mobiles : cette barrière d'acceptabilité sociale peut changer dans un futur proche si la modalité est largement disponible et utilisée partout. Ainsi, en plus de fournir des résultats équivalents à ceux de la modalité *Flick*, ces deux modalités combinées permettent d'éviter les problèmes d'occultation dus à l'interaction tactile.
Ces résultats nous ont donc motivé à approfondir l'étude de modalités combinées pour naviguer dans une vue bifocale. En particulier l'expérience décrite dans la section suivante mesure les performances pour chaque phase de notre modèle. Nous avons de plus étendu l'éventail des possibilités pour les modalités : en plus des trois modalités *Flick*, *Tilt* et *Interaction physique*, nous considérons une modalité reposant sur des capteurs de pression.

4.3 Expérience 2

Lors de cette deuxième expérience sur la navigation multimodale dans une vue bifocale, nous étudions les performances de chacune des deux phases de navigation du modèle de navigation ainsi que la phase de transition entre ces deux phases. De plus nous élargissons le panel des modalités en considérant une modalité basée sur des capteurs de pression, évaluée efficace pour naviguer dans de grands espaces d'information [119].

4.3.1 Contexte

Nous avons mené l'expérience sur un iPhone 4 (960x640 pixels en mode paysage). Le programme a été entièrement codé sur iOS, à l'exception d'un programme en C qui permettait de récupérer les valeurs des capteurs de pression et de les renvoyer au dispositif mobile en utilisant le WiFi.

Dans la vue bifocale, nous distinguons l'espace focus de l'espace contexte qui sont représentés à l'écran à deux échelles différentes. Pour afficher les deux zones correspondantes, zone de focus et zone de contexte, nous avons divisé l'écran en deux zones égales pour chacun des deux espaces. Ainsi, la zone de focus et la zone de contexte (divisée elle aussi en deux zones à l'écran) étaient de 480 pixels de large sur un iPhone 4.

La taille choisie pour l'espace d'information était de 57 600 pixels. Nous avons donc considéré un espace d'information plus grand que dans l'expérience 1 de la section 4.2 afin de tester des distances plus longues et vérifier la tendance observée dans l'expérience 1.

Comme précédemment, nous avons alloué à la zone de contexte un espace de 4 800 pixels de largeur, soit dix fois plus grand que celui du focus. Une fois l'effet de perspective appliqué, cette zone n'occupe plus que 480 pixels à l'écran. Nous obtenons donc un facteur de zoom (fz) entre l'échelle du focus et l'échelle du contexte égal à 12 (57 600 / 4 800). Nous notons enfin que ces paramètres correspondent à des cas d'application concrets : par exemple la visualisation d'un graphe d'indice boursier sur 10 ans avec une période d'un mois affiché dans la zone de focus : 480 pixels (taille du focus) * 10 ans * 12 mois = 57 600 pixels.

4.3.2 Modalités comparées

Comme pour l'expérience 1, nous fixons la modalité utilisée pour la phase de navigation précise. Elle repose sur la modalité *Flick* fournie par le SDK iOS. Elle est intégrée dans les dispositifs iOS et est un standard bien établi pour la navigation sur mobile. Cette modalité est active uniquement dans la zone de focus.

Type d'action / Mode de navigation	Discrète	Continue
Relatif	DiscretePress	ContinuousPress, Tilt, FastDrag
Absolu	DirectTouch, Peephole	

Table 4.1 – 6 modalités pour la phase de navigation rapide.

Cette modalité est utilisée comme modalité précise lors de la phase de navigation précise pour toutes les modalités combinées. La modalité *Flick* est prioritaire par rapport à toutes les modalités permettant ainsi une transition rapide entre les modalités. Cela signifie qu'une interaction tactile à l'écran stoppe l'action en cours d'une autre modalité. Nous avons fait ce choix pour permettre une transition plus rapide des deux phases de navigation. Par exemple, pour passer de la modalité *Tilt* à la modalité *Flick*, il suffira de poser un doigt à l'écran alors même que le dispositif est encore incliné. Sans cela, il aurait d'abord fallu ramener le dispositif mobile à sa position de référence pour stopper la navigation causée par le *Tilt* pour ensuite utiliser la modalité *Flick*.

Nous avons exploré la combinaison de cette modalité précise et bien connue des utilisateurs avec 6 modalités pour la phase de navigation rapide. Ces 6 modalités sont répertoriées dans la Table 4.1 selon les dimensions identifiées dans l'état de l'art du chapitre 3. Nous les décrivons dans les sections suivantes.

Figure 4.7 – Effet d'agrandissement : quand la loupe se déplace vers le haut, l'image agrandie se déplace vers le bas. Issue de [51].

Contrairement à l'expérience 1 où chacune des modalités utilisait la métaphore d'interaction du ruban, nous avons décidé ici d'utiliser la métaphore d'interaction de la loupe. Ainsi, naviguer vers la droite (respectivement vers la gauche) déplacera la zone de focus vers la droite (respectivement vers la gauche). Nous avons fait ce choix dans le but d'optimiser un peu plus l'espace à l'écran. En effet, en fixant l'espace d'information et en déplaçant la zone de focus, l'espace à l'écran est tout le temps optimisé. Avec la métaphore d'interaction du ruban, une partie de l'écran se retrouve vide lorsque le focus est sur l'une ou l'autre des extrémités de l'espace d'information (Figure 2.21).

La métaphore d'interaction de la loupe pose par contre un problème lors de l'utilisation de la modalité précise *Flick*. En effet, en déplaçant son doigt à l'écran vers la gauche (respectivement vers la droite), la zone de focus se déplace vers la gauche (respectivement vers la droite). Or, la faible amplitude des déplacements réalisés lors de cette phase rend les mouvements de la zone de focus presque imperceptibles, alors que ceux de l'espace d'information le sont. L'utilisateur a alors l'impression de manipuler l'espace d'information plutôt que la zone de focus. Or, en déplaçant son doigt à l'écran vers la gauche (respectivement vers la droite), l'utilisateur voit l'espace d'information se déplacer dans le sens opposé, ce qui n'est pas le sens de défilement couramment utilisé sur dispositifs mobiles. Ce phénomène a été décrit par Gutwin [51] et est illustré à la Figure 4.7.
Pour cette raison, nous avons décidé, d'inverser le sens des gestes de *Drag*/*Flick* de la modalité précise *Flick*, pour qu'un déplacement du doigt à l'écran vers la gauche (respectivement vers la droite) déplace la zone de focus dans le sens opposé, et donne ainsi l'impression d'utiliser la métaphore d'interaction du ruban.

4.3.2.1 FastDrag et DirectTouch

FastDrag et *DirectTouch* sont deux modalités basées sur l'interaction tactile.

FastDrag est basée sur des gestes de *Drag* dans l'espace contexte. Ainsi, un déplacement de un pixel à l'écran provoque un déplacement de la zone de focus de un pixel dans l'espace contexte, et par conséquent un déplacement de *fz* pixels dans l'espace focus (fz étant le facteur de zoom). Ceci permet de minimiser le

clutching par rapport à une navigation à l'échelle du focus, dans le cas d'un grand espace d'information.
Ainsi les deux modalités *FastDrag* et *Flick* sont basées sur des gestes de *Drag* :
Nous distinguons la modalité *FastDrag* de celle *Flick* par le nombre de doigts à l'écran : deux doigts se déplaçant n'importe où à l'écran définissent la navigation dans l'espace de contexte (*FastDrag* : phase de navigation rapide) et un doigt se déplaçant dans la zone de focus définit la navigation dans l'espace de focus (*Flick* : phase de navigation précise).
Avec la combinaison (*FastDrag*+*Flick*), les utilisateurs peuvent facilement passer d'une modalité à l'autre en retirant ou posant un doigt à l'écran.

DirectTouch permet des sauts directs vers une nouvelle position. L'utilisateur sélectionne un point (par un appui à l'écran) n'importe où dans la zone de contexte pour le déplacer dans la zone de focus. Cela permet un mode de navigation absolu très rapide et évite le clutching. Lors de la sélection d'un point, une animation est déclenchée, aidant ainsi l'utilisateur à limiter les effets de désorientation. En raison du facteur de zoom entre l'espace de focus et l'espace de contexte, ainsi que l'effet de perspective de la vue bifocale, la modalité n'est pas précise. L'utilisateur rencontre les problèmes de pointage sur surface tactile décrits dans la section 3.2.1 chapitre 3 (« fat finger » et perception de la zone touchée). En effet, la zone tactile à l'écran est de 480 pixels alors que la zone de contexte représentée en perspective est de 4 800 pixels : un pixel de l'écran est donc associé à 10 points dans l'espace contexte, soit 120 points dans l'espace focus. La modalité *DirectTouch* est basée sur une interaction tactile. Lors de la combinaison (*DirectTouch*+*Flick*) la transition vers la modalité *Flick* est donc très rapide puisqu'un même doigt peut servir aux deux modalités.

4.3.2.2 Tilt

Les résultats prometteurs de *Tilt* présentés par Fitchett & Cockburn [39] ainsi que les commentaires positifs de notre précédente étude (préférence utilisateur de la Figure 4.6) nous ont motivés à réutiliser cette modalité dans cette nouvelle étude. Comme dans l'expérience 1, les utilisateurs inclinent le dispositif autour de l'axe vertical du plan défini par le dispositif. Incliner le dispositif vers la gauche (respectivement vers la droite) déplace la zone de focus vers la gauche (respectivement vers la droite). Plus l'appareil est incliné, plus la vitesse de défilement est importante dans l'espace de contexte.

Notre implémentation de *Tilt* est légèrement différente de celle de l'expérience 1 : nous avons supprimé les deux paliers de vitesse et augmenté les vitesses de navigation possibles. En effet, l'espace d'information étant plus grand, nous avons voulu autoriser des vitesses plus importantes. La perte de précision engendrée est minime du fait de la présence de la modalité précise *Flick*. MacKenzie [83] révèle qu'un gain (facteur utilisé pour la conversion linéaire de l'angle en vitesse) compris entre 50 et 100 est optimal pour contrôler le *Tilt*, nous avons donc utilisé un facteur égal à 75, autorisant ainsi une vitesse de défilement de 4 500 pixels par seconde pour un angle d'inclinaison de 60° autour de la position de référence. Cette position de référence est définie par la position de l'appareil lors du lancement de l'application. Puisque les utilisateurs peuvent changer de position au cours de l'utilisation, nous leur avons permis de redéfinir cette position de référence en effectuant un double tap sur l'écran.

Comme pour l'implémentation de *Tilt* dans l'expérience 1 (Figure 4.2) et celle de Fitchett & Cockburn [39], nous avons défini une zone morte dans laquelle aucun défilement n'est déclenché : [-6.5°, 6.5°] autour de la position de référence. Cette région stable, nous le rappelons, permet d'éviter des défilements indésirables dus à des tremblement ou au changement de modalité. Les utilisateurs doivent donc replacer le dispositif dans cette zone stable pour arrêter le défilement.

4.3.2.3 ContinuousPress et DiscretePress

Figure 4.8 – Prototype basé sur des capteurs de pression.

ContinuousPress et *DiscretePress* sont deux modalités basées sur des capteurs de pression. Nous pensons que ce type de capteurs sera intégré dans les dispositifs mobiles dans un avenir très proche. Les derniers *smartphones* Apple et Huawei intègrent déjà des écrans sensibles à la pression, mais cela implique une interaction à l'écran et donc n'empêche pas les problèmes d'occultation. De plus de bons résultats obtenus par Spelmezan et al. [119] avec ce type de capteurs nous ont incités à considérer cette modalité. Nous avons construit un prototype utilisant une carte Arduino Micro et deux capteurs de pression Interlink Elektroniks FSR 400 comme le montre la Figure 4.8. Ces capteurs sont capables de détecter une force comprise entre 0.2N et 20N. Leur petite taille (0.38mm d'épaisseur et 5mm de diamètre), permet de les intégrer facilement sur un dispositif mobile sans en modifier le facteur de forme. Les capteurs ont été fixés sous une coque de dispositif mobile en plastique. Nous avons ajouté de petits morceaux de tissu sur la coque de dispositif mobile pour permettre aux utilisateurs de savoir rapidement où poser leurs doigts sans avoir à regarder le dispositif. La carte Arduino est fixée sur le dos du téléphone, ne gênant pas la tenue de celui-ci en mode paysage. Le capteur gauche (respectivement droit) déplace la zone de focus vers la gauche (respectivement vers la droite). Comme les autres modalités dédiées à la phase de navigation rapide, le défilement se produit à l'échelle du contexte.

Avec *DiscretePress*, les capteurs utilisent des actions discrètes. Ainsi, les utilisateurs naviguent en cliquant (action <appui-relâchement>) sur les capteurs. Comme dans l'étude de Spelmezan et al. [119], deux niveaux de pression déclenchent deux événements différents : *light-click* et *strong-click*.
light-click est déclenché après un clic avec une pression minimale de 0.5N et une pression maximale de 2N. Cet événement génère alors un déplacement de 80 pixels.
strong-click est déclenché après un clic avec une pression minimale de 2N. Cet événement est associé à un déplacement de 800 pixels.

strong-click permet aux utilisateurs de se déplacer rapidement vers la cible alors que *light-click* leur permet d'être un peu plus précis en arrivant à proximité de la cible.

ContinuousPress utilise les capteurs de pression avec des actions continues. Plus un capteur est pressé fort, plus la vitesse de défilement est importante. La navigation reste active jusqu'à ce que le capteur soit relâché. Comme nous ne disposions pas d'un amplificateur opérationnel pour linéariser les valeurs retournées par les capteurs, nous avons défini trois plages de pressions associées à trois gains de vitesse différents pour simuler une linéarisation :
3 quand la pression est inférieure à 0.5N,
4 quand la pression est comprise entre 0.5 N et 2.5N, et
6 quand la pression est supérieure 2.5N.
Ces facteurs sont appliqués aux valeurs renvoyées par la carte Arduino qui convertit la force détectée en entier compris entre 0 et 1023 (0 = 0N, 1023 = 20N). Cette solution permet d'avoir une vitesse de défilement similaire à celle de la modalité *Tilt* (0 à 4 500 pixels par seconde).

4.3.2.4 Peephole

Comme la modalité *Interaction physique* de l'expérience 1, cette modalité utilise la manipulation spatiale afin d'associer une position physique du dispositif à une position virtuelle dans l'espace d'information. Là encore, nous avons choisi une association sphérique entre les deux positions (physique et virtuelle) comme illustré sur la Figure 4.4. Pour implémenter la modalité *Peephole* nous avons cette fois-ci utilisé le magnétomètre intégré au dispositif pour détecter la rotation autour de l'utilisateur (en utilisant l'angle par rapport au Nord). Nous voulions vérifié si une utilisation du magnétomètre à la place du gyroscope permettait de rendre l'utilisation de cette modalité plus stable. En effet, la position un peu moins confortable qu'avec la modalité *Tilt* (mouvements des bras et/ou du buste) peut entraîner des tremblements déclenchant une navigation non désirée. Nous n'avons pas observé d'amélioration. Nous n'avons pas envisagé de fusionner les deux capteurs car c'est finalement la sensisiblité des mouvements perçues qui peut déranger les utilisateurs en position d'instabilité, et l'améliorer ne ferait qu'agraver le problème.

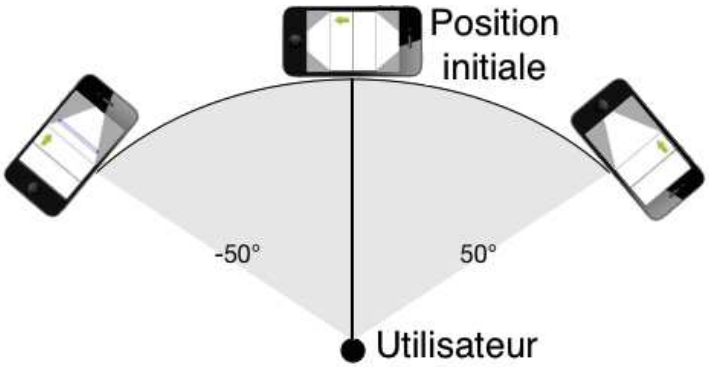


Figure 4.9 – Modalité *Peephole*.

Lors du lancement de l'application, l'angle courant est capté et mis en correspondance avec le milieu de l'espace d'information car c'est la position initiale de la zone de focus. Nous utilisons ensuite la différence entre cet angle de

référence et l'angle courant pour définir la position correspondante dans l'espace d'information. Nous avons restreint l'amplitude du mouvement dans la plage de [-50°, 50°] autour de la position initiale (Figure 4.9). Cela permet aux utilisateurs de naviguer dans tout l'espace d'information sans avoir à effectuer une rotation importante du buste ou des bras. Un point de l'espace physique correspond par conséquent à plusieurs points de l'espace virtuel : la modalité n'est donc pas précise. Néanmoins cette modalité est dédiée à la phase de navigation rapide, la modalité *Flick* apportant la précision pour la phase de navigation suivante.

Cette modalité est activée sur demande, comme décrit par Spindler et al. [122]. L'activation repose sur le capteur de pression droit afin d'éviter le problème d'occultation : en effet lors de l'expérience 1 l'activation était faite par un appui sur l'écran provoquant une occultation partielle de la vue bifocale. La modalité est active tant que le capteur est pressé ; elle est désactivée lorsque le capteur est relâché. Cette activation à la demande permet aux utilisateurs de passer d'un mode de navigation absolu à relatif et permet une visualisation stable lorsque la navigation spatiale est désactivée. Pour utiliser la manipulation spatiale, les utilisateurs doivent décrire une rotation avec leurs bras ou leur buste vers la position désirée, puis appuyer sur le capteur pour déplacer la zone de focus sur la position correspondante.

4.3.3 Tâche de navigation

Nous avons repris la tâche de navigation de l'expérience 1, tout en apportant des modifications graphiques.

Comme pour cette expérience la zone de focus était mobile (métaphore d'interaction de la loupe), le curseur était représenté au centre de celle-ci. Par conséquent, les participants déplaçaient la zone de focus pour amener le curseur sur la cible et le maintenir pendant 1 seconde. Après cela, une nouvelle cible apparaissait. Le curseur était représenté par une ligne noire verticale de 6 pixels de large (Figure 4.10) afin d'être clairement visible à l'écran. La cible était représentée par une zone rouge de 60 pixels de large (10 fois plus grande que le curseur) dans la zone de focus (Figure 4.10.b). Donc, sa taille dans la zone de contexte était $60/fz$ pixels, où fz est le facteur de zoom entre l'espace focus et l'espace contexte.

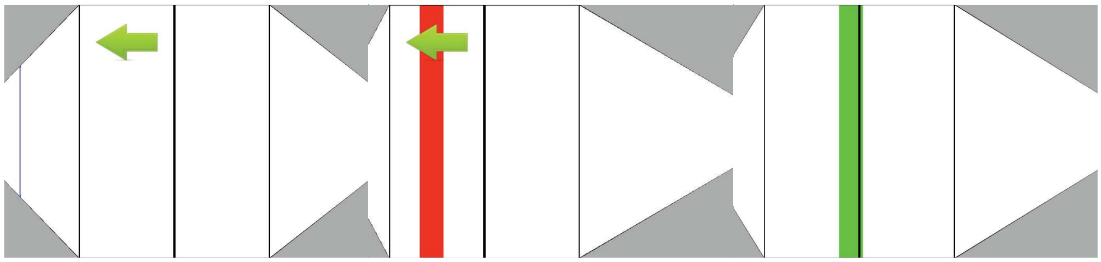
La taille de la cible a été choisie pour être facilement sélectionnée. En effet comme pour l'expérience 1, la tâche concrète est de déplacer des points d'intérêt dans la zone de focus afin d'obtenir leurs détails : la tâche concrète n'est donc pas une tâche de pointage précis. Nous ne voulions donc pas que la difficulté de la tâche soit due à la phase de pointage précis lorsqu'on met le curseur sur la cible. Enfin la cible devient verte lorsque le curseur est sur elle (Figure 4.10.c).

En raison de l'effet de perspective et du facteur de zoom, la cible n'était pas clairement visible dans la zone de contexte lorsqu'elle se trouvait éloignée de la zone de focus. Ainsi, comme dans l'expérience 1, une flèche verte affichée en haut de la zone de focus indiquait la direction vers laquelle se trouve la cible (Figure 4.10). Deuxièmement, la cible était entourée par un rectangle bleu dans la zone de contexte pour être toujours perceptible, peu importe la distance. Ceci permettait

d'éviter l'étape de recherche de la cible, qui peut être trouvée par hasard. Nous nous concentrons donc que sur les performances de navigation.

L'ordre d'apparition des cibles contraint les participants à effectuer la tâche à la fois à gauche et à droite, de manière équitable.

Dans notre évaluation, alors que le facteur de zoom est défini par la vue bifocale, nous avons fixé la taille de la cible et fait varier la difficulté des tâches en tenant compte de 4 distances pour la cible: D1 = 4800 pixels, D2 = 9600 pixels, D3 = 19200 pixels et D4 = 38400 pixels. Ces distances sont exprimées à l'échelle du focus (respectivement 400, 800, 1600 et 3200 pixels à l'échelle du contexte). Ces distances définissent 4 indices de difficulté pour la tâche, respectivement ID1 = 6.33, ID2 = 7.33, ID3 = 8.32 et ID4 = 9.32.



(a) **(b)** **(c)**

Figure 4.10 – Tâche: (a) Début de la tâche: La flèche indique la direction de la cible. (b) Tâche en cours : la cible est dans le focus. (c) Fin de la tâche: Le curseur se trouve sur la cible.

4.3.4 Procédure

Nous avons regroupé les tâches en six blocs, un par modalité combinée. Après une brève démonstration de la modalité combinée à utiliser, les participants ont effectué six tâches d'entraînement avant de commencer un bloc. Ils pouvaient poser toutes les questions souhaitées au cours de cette étape.

L'ordre de présentation des modalités combinées a été contrebalancé en utilisant un carré latin. Pour chaque bloc (chaque modalité combinée), les participants devaient effectuer 6 tâches par indice de difficulté. Par conséquent, ils devaient effectuer 24 (6 x 4 IDs) tâches par modalité combinée soit 144 tâches de navigation au total.

Les participants ont également répondu à un questionnaire d'utilisabilité (SUS) [17] (Annexe 2) sur la modalité combinée utilisée avant de commencer la suivante. Ceci permettait de recueillir le point de vue subjectif du participant sur la modalité qu'il venait d'utiliser. Les participants avaient pour consigne d'être aussi rapide que possible. Enfin, les participants ont été invités à classer les trois modalités combinées qu'ils préféraient par ordre de préférence et d'expliquer leurs choix. L'évaluation a duré environ 35 minutes par participant.

4.3.5 Hypothèses

En nous basant sur la logique de conception de notre modèle de navigation multimodale et sur les résultats de l'expérience 1, nous formulons deux hypothèses :

- H1 : Les modalités combinées offrant un mode de navigation absolu (*DirectTouch* et *Peephole*) seront plus rapides pour les indices de difficulté les plus grands lors de la phase de navigation rapide que les modalités au mode de navigation relatif. Les deux modalités *DirectTouch* et *Peephole* sont très imprécises en raison du facteur de zoom ; par conséquent, nous nous attendons également à ce que la phase de navigation précise soit plus longue en raison de la grande distance restant à parcourir pour atteindre la cible.
- H2 : Les modalités combinées utilisant des gestes sur l'écran pour la phase de navigation rapide et la phase de navigation précise (*FastDrag+Flick)* , (*DirectTouch+Flick*) auront un temps de transition plus court en réduisant le temps de « homing ». Le temps de « homing » est le temps nécessaire au changement de position physique pour passer d'une modalité à l'autre.

4.3.6 Résultats

4.3.6.1 Performances : temps d'exécution

Pour chaque tâche, nous avons enregistré les différents événements (écran tactile, capteurs de pression et capteurs embarqués) déclenchés par les participants. Une analyse à postériori des fichiers générés nous a permis de mesurer avec précision pour chaque cible, le temps des trois phases du modèle de navigation : phase de navigation rapide – phase de transition – phase de navigation précise.

Pour chaque modalité, nous définissons la phase de navigation rapide comme le temps passé à utiliser la modalité dédiée à la navigation rapide à l'échelle du contexte.
Pour les modalités basées sur la pression cela commence lorsque les utilisateurs pressent un des capteurs de pression. Elle se termine lorsque plus aucun capteur de pression n'est pressé.
Pour la modalité *Tilt*, elle débute lorsque l'inclinaison sort de la zone stable, et se termine au retour dans la zone stable.
Pour la modalité *Peephole*, elle débute lorsque l'utilisateur active la modalité en touchant le capteur de pression, et termine lorsqu'il le relâche.
Pour les modalités basées sur l'interaction tactile, la phase de navigation rapide débute lorsque l'utilisateur touche l'écran, et termine lorsqu'il relâche l'écran.
Dans le cas des modalités combinées autorisant l'entrelacement des deux modalités (*DiscretePress*, *ContinuousPress*, *Tilt*), la phase de navigation rapide peut également terminer lorsque l'utilisateur utilise la modalité précise alors qu'il est en train d'utiliser la modalité rapide. En effet, comme détaillé dans la description des modalités, la modalité précise est prioritaire par rapport à la modalité rapide.
Enfin, si plusieurs actions successives sont effectuées avec la modalité rapide (par exemple plusieurs actions <appui-relâchement> du capteur pour les modalités basées sur les capteurs de pression), la phase de navigation rapide comprend le temps passé pour chaque action ainsi que le temps entre chacune d'elles.

La phase de navigation précise est définie comme le temps passé en utilisant la modalité précise (à l'échelle du focus). Elle commence lorsque les utilisateurs touchent l'écran et se termine quand ils enlèvent leur doigt de l'écran (geste de *Drag*) ou lorsque l'effet d'inertie s'arrête (geste de *Flick*). Si plusieurs gestes de *Drag/Flick* sont effectués, la phase de navigation précise comprend le temps passé pour chacun des gestes ainsi que le temps entre chacun d'eux.

La phase de transition est définie comme étant le temps écoulé entre ces deux phases. Ainsi, dans le cas des capteurs de pression par exemple, si les utilisateurs ont mis le doigt sur l'écran tandis que le capteur de pression était pressé, le temps de transition est nul. Dans le cas contraire, il est défini comme le temps entre le moment où la navigation rapide cesse et le moment où les utilisateurs touchent l'écran pour initier la phase de navigation précise.

Nous avons vérifié la normalité de nos données en utilisant le test de Shapiro-Wilk. Il a révélé un écart par rapport à la distribution normale, même avec une transformation (log, racine carrée). Nous avons donc utilisé le test non paramétrique de Kruskal-Wallis afin de tester l'effet significatif de chaque facteur (Modalité et ID) sur chaque variable dépendante (temps d'exécution pour chaque phase).

4.3.6.1.1 Phase de navigation rapide

Le test de Kruskal-Wallis a révélé un effet significatif pour les modalités combinées sur le temps d'exécution de la phase de navigation rapide ($p < 0.001$). Un test post-hoc de Dunn avec une correction de Bonferroni a montré qu'il existait des différences significatives entre toutes les paires de modalités combinées, sauf entre *ContinuousPress* et *DiscretePress* ($p = 0.1779$), entre *ContinuousPress* et *FastDrag* ($p = 0{,}8912$) et entre *DiscretePress* et *FastDrag* ($p = 0.2867$).

Comme prévu (H1), *DirectTouch* est clairement la modalité la plus rapide (Figure 4.11.a), car elle fournit un mode de navigation absolu. Toutefois, pour la modalité *Peephole*, qui permet également d'accéder directement à une zone du contexte en déplaçant l'appareil, le résultat est différent : *Peephole* est plus lent que *FastDrag* et *ContinuousPress*. Donc H1 n'est pas vérifiée pour *Peephole*. Cela s'explique par le fait que les participants ont utilisé cette modalité dans un mode de navigation relatif, comme lors de l'expérience précédente. Par conséquent, les participants ont passé plus de temps que prévu sur cette phase. Cette utilisation observée de la modalité *Peephole* est en contradiction avec notre rationnel de conception qui consiste à fournir un mode de navigation absolu par le positionnement du dispositif sur un espace virtuel à une dimension situé en face de l'utilisateur. Deux raisons peuvent expliquer cette utilisation observée de la modalité. D'abord la métaphore d'interaction du déplacement du dispositif qui fournit une fenêtre sur un espace virtuel d'information (appelée métaphore Lens-in-Hand par Pahud et al. [97]) est peut-être plus forte avec un espace virtuel 2D comme dans [97, 122]. De plus, cette métaphore Lens-in-Hand est peut-être perçue en opposition avec la métaphore graphique consistant à afficher l'espace d'information complet (focus et contexte) sur l'écran du dispositif mobile. Pour le reste de l'analyse, nous considérons que la modalité *Peephole* est une modalité dont le mode de navigation est relatif, comme la modalité *Tilt* par exemple.

Le temps d'exécution est beaucoup plus important pour la modalité *Tilt*. Tout d'abord, nous avons observé pour cette modalité seulement, que les participants revenaient accidentellement à la phase de navigation rapide (*Tilt*) après la phase de navigation précise (*Flick*). Bien que nous ayons défini une zone stable pour l'inclinaison ([-6.5°, 6.5°] autour de la position initiale) dans laquelle aucun défilement n'est déclenché, l'interaction tactile a provoqué une inclinaison non désirée du dispositif entraînant ainsi un défilement non voulu. Cette observation explique que le temps d'exécution de la phase de navigation rapide pour la modalité *Tilt* soit très long. Une autre explication est la façon dont les participants utilisaient les modalités combinées. En effet, nous avons observé une stratégie de navigation inattendue. Plutôt que de déplacer rapidement la zone de focus pas trop loin de la cible pour passer ensuite à la modalité précise, les participants ont passé du temps à essayer de mettre la cible dans la zone de focus directement avec la modalité rapide. Cette stratégie a peut-être été encouragée par le fait que la modalité précise implique des gestes tactiles dans la zone de focus seulement. La phase de navigation précise était alors très courte puisqu'il y avait très peu de distance à parcourir pour mettre le curseur sur la cible, mais la phase de navigation rapide était elle beaucoup plus longue du fait du manque de précision des modalités utilisées : le temps gagné sur la phase de navigation précise ne compensait pas celui perdu sur la phase de navigation rapide.

Ainsi, le temps d'exécution au cours de cette phase est lié à la précision de la modalité. Ceci est confirmé par la bonne performance de la modalité *FastDrag*, qui est une modalité relativement précise du fait de l'interaction tactile qu'elle utilise. De plus, avec *FastDrag* le clutching a été réduit au minimum, ce qui rend la modalité encore plus rapide. Avec *ContinuousPress*, une pression brève et légère sur un capteur de pression effectuait une phase de navigation très courte et permettait donc aux participants d'être précis. En ce qui concerne *Peephole*, l'activation à la demande avec une pression sur le capteur de pression améliore la précision. Les participants devaient simplement relâcher le capteur pour arrêter la navigation. Alors qu'avec *Tilt*, ils devaient revenir à la position statique, puis anticiper le mouvement ou naviguer plus lentement. Ceci explique que les performances sont meilleures avec *Peephole* qu'avec *Tilt*.

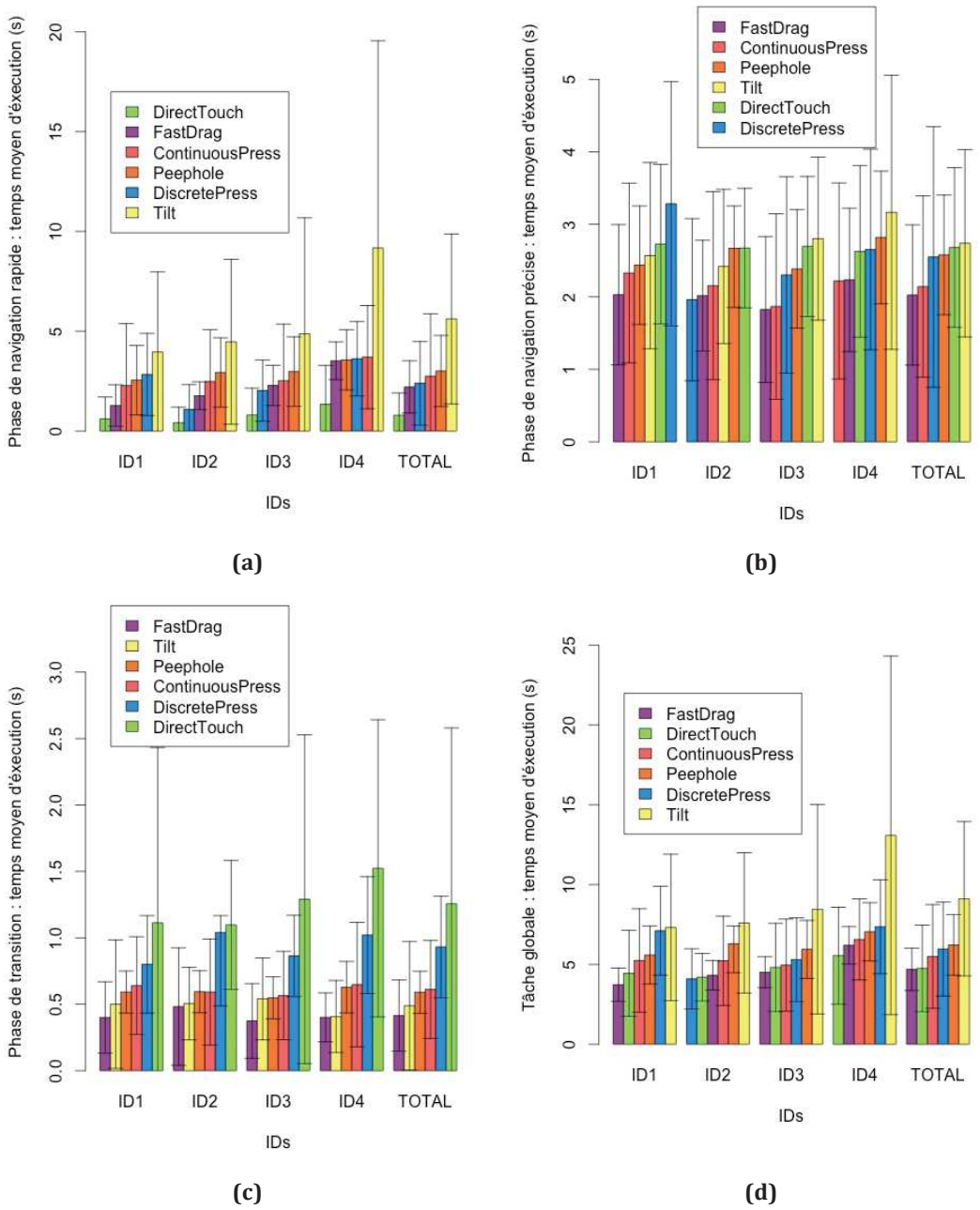


Figure 4.11 – (a) Temps moyens d'exécution pour la phase de navigation rapide (b) Temps moyens d'exécution pour la phase de navigation précice (c) Temps moyens d'exécution pour la phase de transition (d) Temps moyens d'exécution pour la tâche globale avec un intervalle de confiance correspondant de 95%.

Enfin, comme le montre la Figure 4.11.a, le temps d'exécution augmente au fur et à mesure que l'indice de difficulté augmente pour toutes les modalités combinées, sauf pour *DiscretePress* et *DirectTouch*.
Premièrement, *DiscretePress* est plus lente pour l'indice de difficulté le plus faible. Nous expliquons ce résultat par la difficulté pour certains participants à distinguer les deux niveaux de pression. Cette difficulté est confirmée par les préférences des utilisateurs. Ils ont par conséquent effectué de longs mouvements non désirés suite à un appui un peu trop fort sur les capteurs de pression.

Deuxièmement, malgré que *DirectTouch* fournit un accès direct, la modalité est néanmoins dépendante de la distance en raison de l'effet de perspective. En effet plus la cible est éloignée de la zone de focus, moins elle est visible. La cible est donc plus difficile à sélectionner (problèmes de « fat finger » et perception de la zone touchée décrits à la section 3.2.1 du chapitre 3). Par conséquent, la tâche pouvait nécessiter plusieurs clics avant de placer la cible à une distance convenable de la zone de focus pour ensuite basculer en mode de navigation précis. Cette observation explique pourquoi le temps d'exécution de DirectTouch est le plus long pour ID4.

4.3.6.1.2 Phase de navigation précise

Le test de Kruskal-Wallis a révélé un effet significatif pour les modalités combinées sur le temps d'exécution de la phase de navigation précise ($p < 0.001$). Le test post-hoc a montré qu'il n'y avait pas de différence significative entre *ContinuousPress* et *DiscretePress* ($p = 0.07909$), entre *ContinuousPress* et *FastDrag* ($p = 0.4561$), entre *DirectTouch* et *Peephole* ($p = 0.1267$), entre *DirectTouch* et *Tilt* ($p = 0.5413$), et entre *Peephole* et *Tilt* ($p = 0.5516$).

Comme prévu, cette phase ne dépend pas du facteur ID (Figure 4.11.b). En fait, les performances de cette phase dépendent de la précision lors la phase précédente (navigation rapide), comme souligné aussi par Appert et al. [5]. *FastDrag* est la modalité la plus rapide pour cette phase. Ceci est dû au fait que cette modalité combinée fournit un meilleur contrôle pendant la phase de navigation rapide, de sorte que les participants parviennent à positionner la cible très proche du curseur. De la même manière et comme expliqué ci-dessus, avec *ContinuousPress*, une pression brève sur le capteur de pression effectue une phase de navigation très courte, ce qui permet aux participants d'être précis. Augmenter la précision lors de la phase de navigation rapide diminue donc le temps de la phase de navigation précise.

Néanmoins, étant donné que les participants ont utilisé la même stratégie de navigation pour toutes les modalités combinées en essayant d'être précis lors de la phase de navigation rapide, il est surprenant de constater de faibles variations du temps d'exécution entre les modalités au cours de cette phase.

4.3.6.1.3 Phase de transition

Le test de Kruskal-Wallis a révélé un effet significatif pour les modalités combinées sur le temps de transition ($p < 0.001$). Le test post-hoc a montré qu'il existait des différences significatives entre toutes les paires de modalités combinées, sauf entre *ContinuousPress* et *Peephole* ($p = 0.6315$).
Comme prévu, la durée de transition ne dépend pas de la distance de la cible.

FastDrag a la phase de transition la plus courte (Figure 4.11.c), confirmant H2 pour cette modalité (pas de temps « homing »).
Puisque *Tilt* n'a pas besoin d'être activée, la transition peut être anticipée, ce qui la rend très rapide : les utilisateurs inclinent le dispositif pour la navigation rapide et touchent l'écran pour la navigation précise sans avoir à ramener le dispositif dans sa position initiale.

ContinuousPress et *Peephole* ont une phase de transition plus longue avec des temps de transition similaires. Pour les deux modalités, nous avons observé que

plusieurs participants ne se sentaient pas à l'aise pour interagir sur l'écran avec le pouce tout en tenant les capteurs avec les index. Par conséquent, ils devaient retirer leurs index des capteurs de pression pour effectuer la phase de navigation précise, ce qui augmentait le temps de transition.

Enfin, la Figure 4.11.c montre que *DirectTouch* et *DiscretePress* ont une phase de transition beaucoup plus longue. H2 n'est donc pas vérifiée pour *DirectTouch*. Ces deux modalités impliquent des actions discrètes. Cela entraîne une désorientation due à la perte de contrôle des mouvements à l'écran, même si une animation était fournie. La durée de la transition est principalement due à cet effet de désorientation.
Ce résultat montre que les changements de positions des doigts ou de la main (présents pour *Tilt*, *Peephole*, *ContinuousPress* et *DiscretePress*) sont moins importants que la désorientation causée par la perte de contrôle des mouvements à l'écran avec les deux modalités *DirectTouch* et *DiscretePress*.

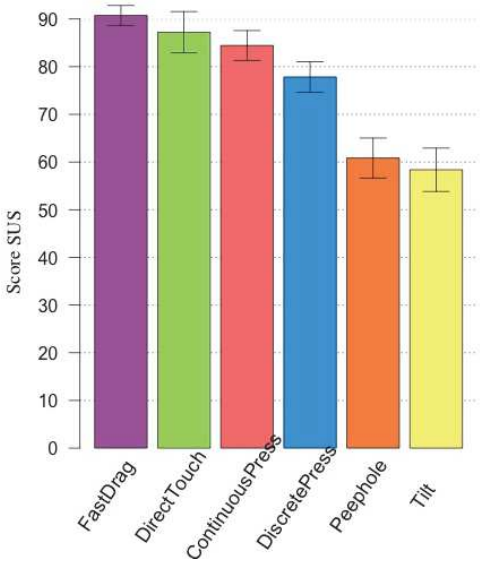
4.3.6.1.4 Temps d'exécution global

Le test de Kruskal-Wallis a révélé un effet significatif pour les modalités combinées sur le temps total d'exécution ($p < 0.001$). Le test post-hoc a montré qu'il existait des différences significatives entre toutes les paires de modalités combinées, sauf entre *FastDrag* et *ContinuousPress* ($p = 0.0734$) et entre *FastDrag* et *DirectTouch* ($p = 0.2193$).

Le total des temps d'exécution confirme la modélisation de la tâche en trois phases. En effet :
La Figure 4.11.d montre que les trois modalités combinées les plus rapides sont *DirectTouch*, *FastDrag*, et *ContinuousPress*. Malgré que *DirectTouch* soit très rapide lors de la phase de navigation rapide, la différence avec les autres modalités est sérieusement réduite sur le temps total d'exécution, notamment en raison des durées de transition.
Les temps quasiment constants constatés durant la phase de navigation précise montrent que cette phase a peu d'influence sur le temps total d'exécution. Nous l'expliquons par la stratégie de navigation adoptée par les participants, en essayant de mettre la cible dans la zone de focus avant de passer à la phase de navigation précise. Cette stratégie faisait que l'amplitude des déplacements pour la phase de navigation précise était très similaire pour toutes les modalités. Ainsi, avec cette stratégie, les deux principaux facteurs sont donc la phase de navigation rapide et la phase de transition. Pour le vérfier, nous avons calculé l'écart-type pour chaque phase et chacune des 6 modalités. Nous avons obtenu 4.2 pour la phase de navigation rapide, 0.7 pour la phase de transition et 1.7 pour la phase de navigation précise. En calculant le rapport entre le temps moyen de chaque phase (2.45 secondes pour la phase de navigation rapide, 0.71 secondes pour la phase de transition et 2.82 secondes pour la phase de navigation précise), la phase de navigation précise a le rapport le plus faible (1.71 pour la phase de navigation rapide, 0.98 pour la phase de transition et 0.49 ciblage pour la phase de navigation précise). Cela signifie que la variabilité de la phase de navigation précise est moins importante que pour les deux autres phases, et que la durée totale d'exécution en utilisant cette stratégie de navigation est influencée par le temps passé sur la phase de navigation rapide et la phase de transition.
En ce qui concerne la dépendance entre le temps d'exécution et l'indice de difficulté, les tendances sont assez semblables à celles de la phase de navigation

rapide. Ce constat renforce le fait que le temps total d'exécution dépend fortement du temps passé sur la phase de navigation rapide et du temps de transition.

4.3.6.2 Préférences utilisateur



Technique	#1	#2	#3	Score
DirectTouch	13	3	0	93
FastDrag	1	9	7	79
ContinuousPress	4	5	4	65
DiscretePress	0	0	4	16
Tilt	0	1	2	13
Peephole	0	0	1	4

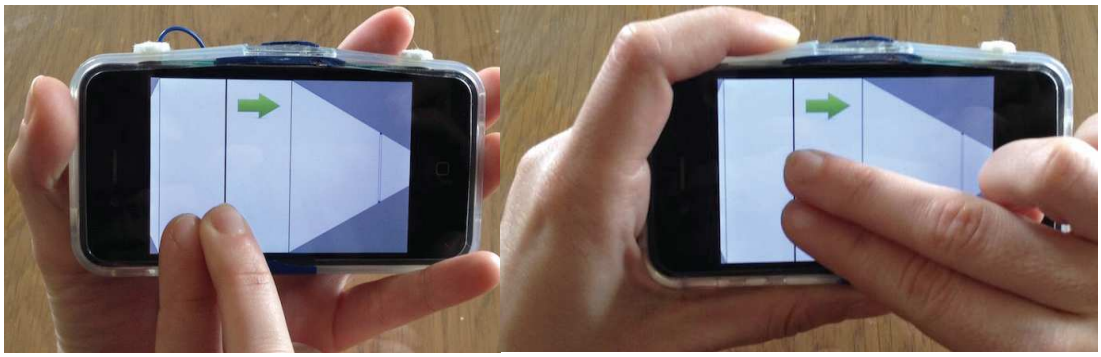
(a) **(b)**

Figure 4.12 – (a) Score SUS moyen avec un intervalle de confiance de 95% (b) Classement par préférence avec le nombre fois où chaque modalité combinée a été classée 1ère, 2è ou 3è.

Contrairement à l'expérience 1, les préférences utilisateur reflètent les résultats de performance. Dans la Figure 4.12.a, les modalités sont ordonnées de gauche à droite par ordre décroissant de score moyen SUS. Le questionnaire SUS montre que les participants ont classé la modalité combinée *FastDrag* comme la plus utilisable.

La préférence de classement (Figure 4.12.b) est similaire au score SUS moyen. Les trois modalités combinées préférées ont également été décrites comme étant les plus utilisables dans le questionnaire SUS. Nous avons calculé le score de classement (S) d'une modalité combinée en utilisant la formule $S = 6 * 1^{er} + 5 * 2^{e} + 4 * 3^{e}$, où le 1er, 2e et 3e désignent le nombre de fois où la modalité a été classée à la position correspondante. Nous avons vérifié qu'avec des coefficients inférieurs (respectivement 3, 2, 1), les résultats restent similaires.

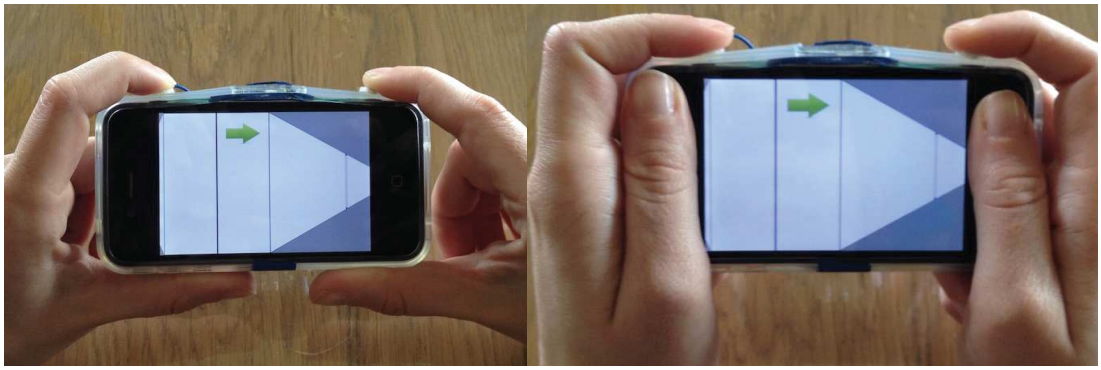
Les participants ont trouvé la modalité *DirectTouch* très facile à utiliser: « *Après 1 ou 2 clics, je sais que je suis assez proche pour passer à la modalité précise* ». Seuls quelques participants ont signalé le problème de désorientation noté dans l'analyse de performances. Nous pensons que l'impression de facilité et d'efficacité fournie par cette modalité connue compensait ce problème.



(a) (b)

Figure 4.13 – FastDrag (a) A partir du bas de l'écran : occultation limitée (b) A partir du côté de l'écran : occultation importante.

FastDrag a été appréciée parce qu'elle est une modalité bien connue. Même si le clutching a été réduit au minimum (mouvements dans l'espace de contexte impliquant 4 ou 5 gestes de Drag pour atteindre les cibles très éloignées), plusieurs participants ont signalé ce problème. De plus, certains participants ont également mentionné le problème de l'occultation. Selon la façon dont les participants tenaient le dispositif, les doigts pouvaient obstruer une grande partie de l'écran (Figure 4.13).



(a) (b)

Figure 4.14 – DiscretePress (a) Pouces sous le dispositif: appuis légers et forts difficiles à différencier (b) Pouces sur l'écran: appuis forts difficiles à effectuer.

Les participants ont expliqué les mauvais scores pour *Tilt*, *DiscretePress* et *Peephole*. Plusieurs participants ont jugé ces modalités trop compliquées à utiliser pour la tâche perçue comme simple à effectuer. Certains d'entre eux nous ont dit que dans une utilisation en contexte réel, cette difficulté perçue pourrait les empêcher de rester concentrés sur l'espace d'information.

- *Tilt* a été perçue comme trop sensible. Tout d'abord, dans notre mise en œuvre, nous avons voulu permettre une navigation rapide sans avoir à fournir un grand angle d'inclinaison et donc sans trop réduire la visibilité de l'écran. Toutefois, certains participants nous ont dit que le problème de visibilité de l'écran lors de la navigation persistait tout de même. Deuxièmement, comme expliqué ci-dessus, les participants ont essayé

d'être aussi précis avec cette modalité qu'avec d'autres modalités, comme *FastDrag* ou *DiscretePress,* qui permettent des mouvements plus précis. Enfin, nous avons parfois observé des navigations non désirées suite à des inclinaisons du dispositif non voulues provoquées par un appui sur l'écran.

- La modalité *DiscretePress* a deux problèmes selon les participants.
  Tout d'abord, contrairement à *DirectTouch*, les participants ont clairement exprimé qu'il y avait un problème de désorientation. Ils n'étaient pas en mesure d'anticiper la position du curseur après avoir appuyé sur un capteur de pression. Cette modalité est nouvelle pour les participants et ce problème pourrait être réduit au minimum et même disparaître après une période d'apprentissage plus longue. De plus la présence d'un retour d'information prédictif (feedforward) pour l'interaction avec des capteurs de pression a été soulignée comme important [119, 133]. Nous revenons sur ces aspects dans le chapitre 5 car il nécessite une autre étude.
  Deuxièmement, selon la manière dont les participants tenaient l'appareil, ils ont constaté qu'il était difficile de différencier les deux niveaux de pression. En effet, les participants tenant le dispositif comme un appareil photo (Figure 4.14.a) avaient du mal à effectuer des pressions légères, et donc effectuaient de grands déplacements à la place des petits. Ceux tenant le dispositif comme une manette de jeu (Figure 4.14.b) avaient au contraire beaucoup de mal à atteindre les pressions élevées et donc à effectuer de grands déplacements. Certains participants ont suggéré d'avoir une étape d'étalonnage pour leur permettre de choisir leurs niveaux de pression préférés.
- Pour la modalité *Peephole*, presque tous les participants ont indiqué que les mouvements physiques étaient trop contraignants et physiquement fatigants. De plus, comme lors de l'étude précédente, certains participants ont mentionné explicitement la question de l'acceptabilité sociale de cette modalité. Malgré le fait que la modalité *Peephole* soit plus couramment utilisée avec des visualisations concaves, alors que la vue bifocale est une visualisation convexe, les participants n'ont pas fait de remarques sur ce point.

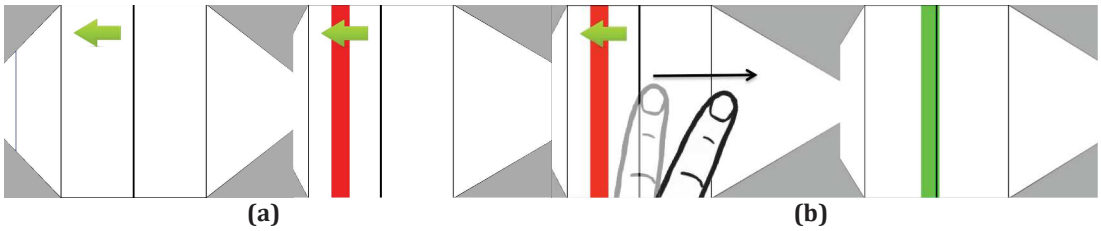


Figure 4.15 – Ambiguïté sur le sens de défilement lorsque la position du focus n'est pas fixe : (a) le focus se déplace vers la gauche, la cible semble se déplacer vers la droite. (b) le geste de *Drag/Flick* est effectuée vers la droite, la cible se déplacer vers la droite.

Enfin, plusieurs utilisateurs nous on fait par d'un problème d'ambiguïté sur le sens de défilement à utiliser pour les gestes de *Drag/Flick*. En effet, en inversant le sens de défilement des gestes de *Drag/Flick* lors de la phase de navigation précise (voir section 4.3.2) les sens de défilement des modalités tactiles lors des phases de navigation rapide et précise sont inversées : durant la phase de navigation rapide, déplacer les doigts vers la gauche (respectivement vers la droite) déplace la zone

de focus vers la gauche (respectivement vers la droite), alors que durant la phase de navigation précise déplacer les doigts vers la gauche (respectivement vers la droite) déplace la zone de focus vers la droite (respectivement vers la gauche). Cette inversion du sens de défilement a perturbé certains utilisateurs qui nous ont fait part qu'ils préfèreraient un sens de défilement unique, mais que la phase de navigation précise devait conserver le sens de défilement habituellement utilisé sur dispositifs mobile car ils y sont habitués. Pour cette raison, l'expérience décrite au chapitre 5 utilise à nouveau la métaphore d'interaction du ruban qui permet de répondre à ce besoin.

4.4 Conclusion

En nous basant sur le modèle de navigation à trois phases décrit dans le chapitre précédent, les deux expériences menées nous ont permis de :

1) Confirmer l'intérêt d'utiliser des modalités combinées pour naviguer dans une vue bifocale sur mobile (expérience 1).
2) Comparer différentes combinaisons de modalités (expériences 1 et 2)
3) Expliquer les performances obtenues à partir de notre modèle (expérience 2).

Nous avons ainsi observé que les modalités combinées basées exclusivement sur l'interaction tactile (*DirectTouch*, *FastDrag*) ont obtenu les meilleurs résultats en terme de performance et de préférence des utilisateurs. Toutefois, malgré ces résultats, ces deux modalités souffrent toujours des problèmes liés à l'interaction tactile : principalement occultation mais aussi le problème du « fat finger ». Les autres modalités comparées permettent de résoudre ces problèmes. Nous soulignons donc le compromis de conception entre les performances de réalisation de la tâche et principalement la non occultation de l'écran, un critère particulièrement crucial lors d'une tâche de navigation.

Pour les modalités autres que tactiles, l'occultation de l'écran est réduite mais les performances de réalisation sont moindres. De plus, ces modalités présentent d'autres difficultés d'interaction. Par exemple, *Tilt* provoque une difficulté de lecture à l'écran, lorsque le dispositif est trop incliné, alors que *Peephole* provoque une fatigue physique due aux mouvements dans l'espace. Les deux autres modalités basées sur les capteurs de pression ont ainsi obtenu de meilleurs résultats (performance et préférence des utilisateurs). En effet, ces derniers ne souffrent pas des problèmes cités ci-dessus, et sont facilement intégrables aux dispositifs actuels. Ce constat fait de ces deux modalités basées sur les capteurs de pression des solutions potentielles au problème de navigation dans une vue bifocale sur mobile. Bien que les performances obtenues ne dépassent pas encore celles des modalités tactiles, deux pistes d'amélioration sont envisageables :

- Positionnement des capteurs : la tenue du téléphone à deux mains force les utilisateurs à utiliser une des mains interagissant sur les capteurs de pression pour interagir à l'écran, rallongeant ainsi la phase de transition, puisque dépendante des changements de position de la main et des doigts. Permettre aux utilisateurs d'avoir une main dédiée à l'interaction tactile permettrait ainsi de réduire la durée de la phase de transition. Ce point reste à vérifier car le temps de changement entre les modalités (phase de

transition) implique au-delà du changement de positions de la main et des doigts un coût cognitif.

- Contrôle durant la phase de navigation rapide : les utilisateurs cherchent à se rapprocher au plus prés de la cible durant cette phase, le temps perdu n'étant pas compensé durant la phase de navigation précise. Offrir plus de contrôle (et donc plus de précision) aux utilisateurs durant cette phase permettrait ainsi de réduire la durée totale de réalisation de la tâche sans toutefois remettre en cause notre approche basée sur la multimodalité.

Le chapitre suivant (chapitre 5) continue notre exploration de l'espace des possibilités en termes de modalités combinées pour naviguer dans une vue bifocale. Les résultats décrits dans ce chapitre nous ont incités à explorer plus en détail les modalités basées sur les capteurs de pression, objet du chapitre suivant.

5 NAVIGATION AVEC DES CAPTEURS DE PRESSION DANS UNE VUE BIFOCALE

5.1 Introduction

L'expérience décrite dans le chapitre précédent a montré l'intérêt d'utiliser des capteurs de pression pour naviguer rapidement dans une vue bifocale sur dispositifs mobiles. Les modalités dont le dispositif d'interaction est un ou des capteurs de pression permettent de résoudre les limitations inhérentes à l'interaction purement tactile pour cette tâche de navigation multi-échelle :

- L'occultation provoquée par l'interaction sur la surface de visualisation elle-même (section 3.2.1 du chapitre 3).
- La nécessité d'effectuer un changement de mode pour distinguer la navigation à l'échelle du contexte (rapide) de la navigation à l'échelle du focus (précise), puisque ces deux types de navigation reposent sur l'interaction tactile. Pour ce changement de mode, plusieurs solutions ont été conçues : utiliser différentes zones à l'écran (navigation rapide en agissant dans la zone de contexte et navigation précise en agissant dans la zone de focus), différents gestes (navigation rapide à deux doigts et navigation précise à un doigt), ou encore utiliser un bouton (physique ou numérique) pour sélectionner le mode à utiliser. Casiez & al. [25] ont par exemple utilisé deux zones distinctes sur un pavé tactile pour distinguer deux modes de navigation : l'un utilisant la navigation basée sur la vitesse permettant une navigation précise, l'autre utilisant la navigation basée sur la position permettant une navigation rapide.

En utilisant les capteurs de pression et en appliquant le modèle en trois phases pour la navigation multi-échelle dans une vue bifocale, ce chapitre présente une approche basée sur une utilisation bimanuelle des modalités combinées. Ceci a pour avantage de limiter le temps passé sur la phase de transition en assignant une

main à la modalité précise (en l'occurrence la modalité tactile comme au chapitre 4) qui est par conséquent tout le temps disponible.

Ainsi, la première section présente et justifie l'approche bimanuelle choisie. Les sections suivantes présentent les différentes modalités conçues qui sont basées sur la capture de pression, ainsi que l'évaluation expérimentale menée.

5.2 Interaction bimanuelle et interaction par pression

Contrairement à l'approche adoptée dans le chapitre précédent où l'utilisateur utilisait ses deux mains durant la phase de navigation rapide (un doigt sur chaque capteur de pression), nous explorons l'usage d'une seule main avec un seul capteur. En plaçant celui-ci sur la face avant du dispositif mobile (Figure 5.1), cela permet :

- d'utiliser la main non dominante tenant le dispositif en mode paysage pour interagir avec le capteur de pression ;
- d'éviter toute occultation de l'écran durant la phase de navigation rapide, qui requiert de parcourir une grande partie de l'espace d'information ;
- libérer la main dominante pour la phase de navigation précise.

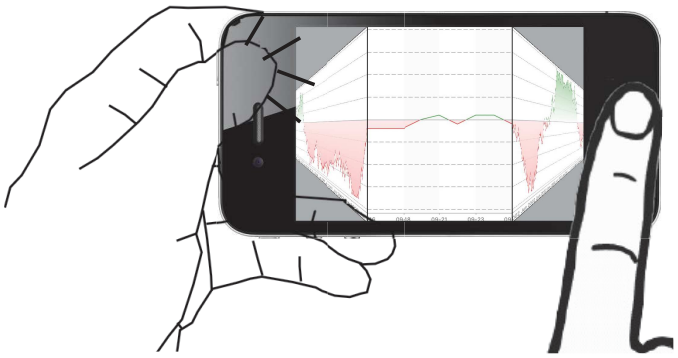


Figure 5.1 – Prototype conçu : capteur de pression sur le panneau avant du dispositif. La main non dominante qui tient le dispositif est dédiée à la modalité basée sur la pression avec le pouce. La main dominante est dédiée à l'interaction tactile.

Utiliser un seul capteur permet donc une utilisation à une main de la modalité basée sur la pression. Nous pouvons ainsi associer une main à chacune des phases de navigation (rapide et précise). Ainsi, selon les propriétés CARE (Complémentarité, Assignation, Redondance et Equivalence) de Serrano & Nigay [116] permettant de caractériser des interfaces multimodales, chaque main est assignée à une modalité, les deux modalités étant complémentaires pour la navigation dans la vue bifocale. Nous attribuons la modalité basée sur la pression à la main non dominante tenant le dispositif pour naviguer dans l'espace contexte, et la modalité tactile (*Flick*) à la main dominante pour la navigation dans l'espace focus. De plus, cette approche bimanuelle est également compatible avec le modèle de la chaîne cinématique présenté par Guiard [49]. En effet, la modalité basée sur la pression est utilisée avec la main non dominante :

- avant la main dominante qui termine la tâche,

- pour définir un espace de référence à la navigation par interaction tactile avec la main dominante.

Néanmoins utiliser un seul capteur de pression ne suffit pas car nous avons besoin de naviguer dans deux directions. Il convient donc d'étudier comment spécifier le sens de défilement désiré.

McLahlan et Brewster [88] ont étudié, sur une tablette, la combinaison bimanuelle de plusieurs modalités. Les auteurs ont décomposé la tâche de navigation en deux sous-tâches : définition du sens de défilement et contrôle de la vitesse de défilement. Ils ont ensuite étudié l'association des deux mains aux deux sous-tâches. Ils ont notamment exploré la combinaison :

- de la modalité basée sur la pression utilisée avec la main non dominante pour contrôler la vitesse
- avec l'interaction tactile avec la main dominante pour spécifier le sens de défilement.

Bien que les résultats expérimentaux aient montré qu'une barre de défilement affichée à l'écran et manipulée par la main non dominante était plus efficace que le capteur de pression pour contrôler la vitesse, les auteurs ont conclu que le fractionnement du contrôle de la vitesse de défilement (pression, main non dominante) et de la définition du sens de défilement (interaction tactile, main dominante) entre les deux mains était une bonne solution de conception en terme de performances pour naviguer sur une tablette.

MacLahlan et al. [89] ont étendu ce travail en étudiant un autre type de combinaison des deux modalités : la pression utilisée en tant que modalité transitoire avec la main non dominante, alors que l'interaction tactile est utilisée pour contrôler la sélection avec la main dominante. Contrairement à l'étude précédente [88], la pression est utilisée ici comme une modalité d'entrée auxiliaire, complétant mais ne remplaçant pas l'interaction tactile.

Pour la conception de la modalité basée sur la pression pour la navigation, nous résumons ici les résultats clefs d'autres études. D'abord, les utilisateurs peuvent utiliser la modalité basée sur la pression avec suffisamment de précision pour naviguer en utilisant leur main dominante sur un smartphone [119] comme avec leur main non dominante sur une tablette [89]. Deuxièmement, la pression peut être contrôlée par la main non dominante tenant le dispositif, ce qui permet une interaction tactile avec la main dominante [88, 89].

Dans ce chapitre, nous étendons ces travaux en considérant la navigation sur un smartphone. Par opposition à [89], nous considérons la pression en tant que modalité principale avec la main non dominante pour une navigation rapide (dans l'espace contexte de la vue bifocale) et l'interaction tactile comme modalité principale avec la main dominante pour la navigation précise (dans l'espace focus de la vue bifocale). Plus étroitement liée à l'étude de l'interaction bimanuelle sur tablette [88], nous nous concentrons sur l'interaction avec un smartphone, pour lequel la posture de la main de l'utilisateur et les muscles impliqués pour contrôler la pression peuvent être différents. En effet, la taille du dispositif, son poids ou encore le contexte d'utilisation sont autant de facteurs pouvant conduire à des performances différentes sur les deux types de dispositifs (tablette et smartphone)

pour une même modalité. Ainsi selon les caractéristiques d'une modalité de navigation sur dispositifs mobiles (section 3.3.1 du chapitre 3) nous fixons le dispositif utilisé comme étant le capteur de pression, et le mode de navigation comme étant relatif. Nous explorons l'effet du type d'action (continu/discret) ainsi que celui d'un nouveau paramètre de conception : l'inertie.

5.3 Matériel

Nous avons développé un prototype utilisant une carte Arduino micro et un capteur de pression Interlink Elektroniks FSR 400. Nous avons fixé le capteur de pression sur le panneau avant du dispositif (Figure 5.2). Contrairement au prototype de l'espérience 2 décrit au chapitre 4, nous n'avons pas fixé la carte Arduino derrière le dispositif pour ne pas gêner sa tenue à une main. Celle-ci était donc déportée sur un bureau. Le dispositif mobile utilisé est un iPhone 4 avec une résolution d'écran de 960x640 pixels en mode paysage. Le programme a été entièrement codé en utilisant le SDK iOS, à l'exception d'un programme en C chargé de récupérer les valeurs retournées par le capteur de pression pour les renvoyer en WiFi au dispositif.

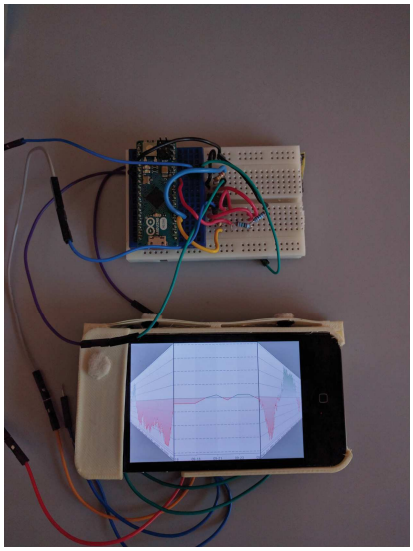


Figure 5.2 – Prototype conçu : le capteur de pression utilisé est celui placé sur le panneau avant du dispositif.

Les valeurs retournées par le capteur de pression n'étant pas linéaires, nous avons branché un amplificateur opérationnel sur le circuit électronique pour les linéariser en fonction de la force appliquée sur le capteur (Annexe 3). L'association d'un amplificateur opérationnel à une fonction de transfert linéaire a été montrée efficace par Stewart & al. [123] et a été utilisée notamment par Spelmezan & a. [119] qui ont obtenu ainsi de bons résultats avec leur technique de navigation basée sur des capteurs de pression.

Le capteur de pression utilise un mode de navigation relatif (contrôle basé sur la vitesse -section 3.3.3.3 du chapitre 3-). Ceci permet d'être plus précis, plus rapide et moins exigeant mentalement que le contrôle basé sur la position [119, 131]. De plus, un mode de navigation absolu avec des capteurs de pression pour naviguer dans un grand espace d'information à une dimension impliquerait de distinguer

une très large plage de pressions différentes. Or, il a été démontré qu'au dessus de 8 ou 10 niveaux de pression, la précision commence à décliner [131].
Enfin même si cette modalité est utilisée pour la phase de navigation rapide, nous avons observé lors de l'étude précédente (section 4.3.6 du chapitre 4) que la stratégie de navigation employée par les utilisateurs implique un minimum de précision, ce que le mode de navigation relatif permet de fournir.

5.4 Options de conception des modalités basées sur la pression

Nous décrivons tout d'abord le contexte d'interaction. Les modalités basées sur les capteurs de pression sont dédiées à la navigation rapide à l'échelle du contexte de la vue bifocale, en utilisant la main non dominante qui tient également le dispositif. Ces modalités sont combinées avec une modalité tactile (*Flick*) pour une interaction précise à l'échelle du focus de la vue bifocale. En outre, comme dans [88] dans le cas d'une tablette, le sens de défilement est défini par un geste de *Swipe* (geste de *Drag* court et rapide) à l'écran avec la main dominante (Figure 5.3 lignes 1 et 2) et la vitesse de défilement est définie par la pression appliquée avec la main non dominante.

Toutes les modalités basées sur la pression conçues implémentent un mode de navigation relatif. Ce mode de navigation relatif nous a incité à faire le parallèle entre la modalité basée sur la pression et la modalité tactile *Flick* bien connue des utilisateurs. Nous considérons ainsi deux options pour la conception des modalités basées sur la pression :

- Types d'action : continue / discrète (section 3.3.1 du chapitre 3)
- Inertie

Inspirée par le geste *Drag*, une première solution de conception est d'associer la pression appliquée à la vitesse de défilement à l'aide d'actions continues (Figure 5.3 ligne 1). Plus les utilisateurs pressent le capteur fort en utilisant la main non dominante, plus la vitesse de défilement est importante. Le défilement associé en temps réel permet un retour visuel continu de la pression appliquée, comme dans les études de Spelmezan & al. [119] ou encore de Wilson & al. [133]. Tout contact à l'écran avec la main dominante arrête la navigation initiée par les capteurs de pression : cela donne la priorité à la modalité tactile et permet aux utilisateurs de mieux contrôler la navigation en leur fournissant un moyen de la stopper avec la main dominante disponible durant cette phase.

En plus d'actions continues sur le capteur, nous explorons également les actions discrètes. Cela implique des actions discrètes sur le capteur de pression et évite ainsi aux utilisateurs de maintenir une force sur le capteur. Plus le capteur est pressé fort, la pression étant calculée au moment du relâchement, plus le déplacement dans l'espace d'information est important. Pour naviguer, les utilisateurs doivent ainsi effectuer une série d'actions discrètes <appui-relâchement> sur le capteur en ajustant la pression en fonction de la distance à parcourir jusqu'à la cible (Figure 5.3 ligne 2). La navigation démarre lorsque l'utilisateur relâche le capteur. Ceci permet aux utilisateurs de mieux contrôler la force appliquée sur le capteur puisqu'ils peuvent la modifier avant de relâcher complètement le capteur. La série d'actions effectuées sur le capteur est similaire à la série de gestes tactiles lors de l'exécution des gestes de *Flick* utilisant le

clutching. Comme l'ont expliqué Nancel et al. [94], les gestes n'utilisant pas le clutching sont plus difficiles à réaliser. Une explication possible est que le clutching permet de décomposer un mouvement difficile en mouvements plus simples et contrôlable. Nous pensons donc que ce principe peut aussi s'appliquer à l'interaction basée sur des capteurs de pression.

Ce parallèle entre la pression et les gestes de *Flick* associés à du clutching nous a conduit à envisager une deuxième option de conception : ajouter de l'inertie aux modalités basées sur la pression. Tout comme les gestes de *Flick* pour la navigation tactile, la navigation basée sur la pression peut être augmentée grâce à un défilement cinétique. Le système simule alors de l'inertie. Le mouvement de défilement par inertie est calculé lors du relâchement du capteur de pression et selon la moyenne des cinq dernières valeurs de pression avant que celle-ci soit nulle. Toute action à l'écran avec la main dominante arrête le défilement par inertie.

Les deux options de conception, à savoir le type d'action (continu/discret) et l'inertie, nous ont permis de définir quatre modalités basées sur la pression. La section suivante décrit leur mise en œuvre.

5.5 Modalités basées sur le capteur de pression implémentées

Comme lors de l'expérience 2 du chapitre 4, nous avons utilisé un espace d'information de 57600 pixels de large. Cette taille correspond à des cas d'application concrets : par exemple la visualisation d'un graphe d'indice boursier de 10 ans avec une période d'un mois affichée dans le focus : 480 pixels (taille de focus) * 10 ans * 12 mois = 57600 pixels.

Les gestes de *Drag/Flick* ont priorité sur le capteur de pression tel que décrit par la machine à état de la Figure 5.4. Les utilisateurs peuvent arrêter la navigation déclenchée par le capteur de pression en touchant l'écran. Ainsi, les événements du capteur de pression sont ignorés s'il y a un doigt sur l'écran (états « Pressure ignored » et « Pressure ignored & Inertia » à la Figure 5.4).

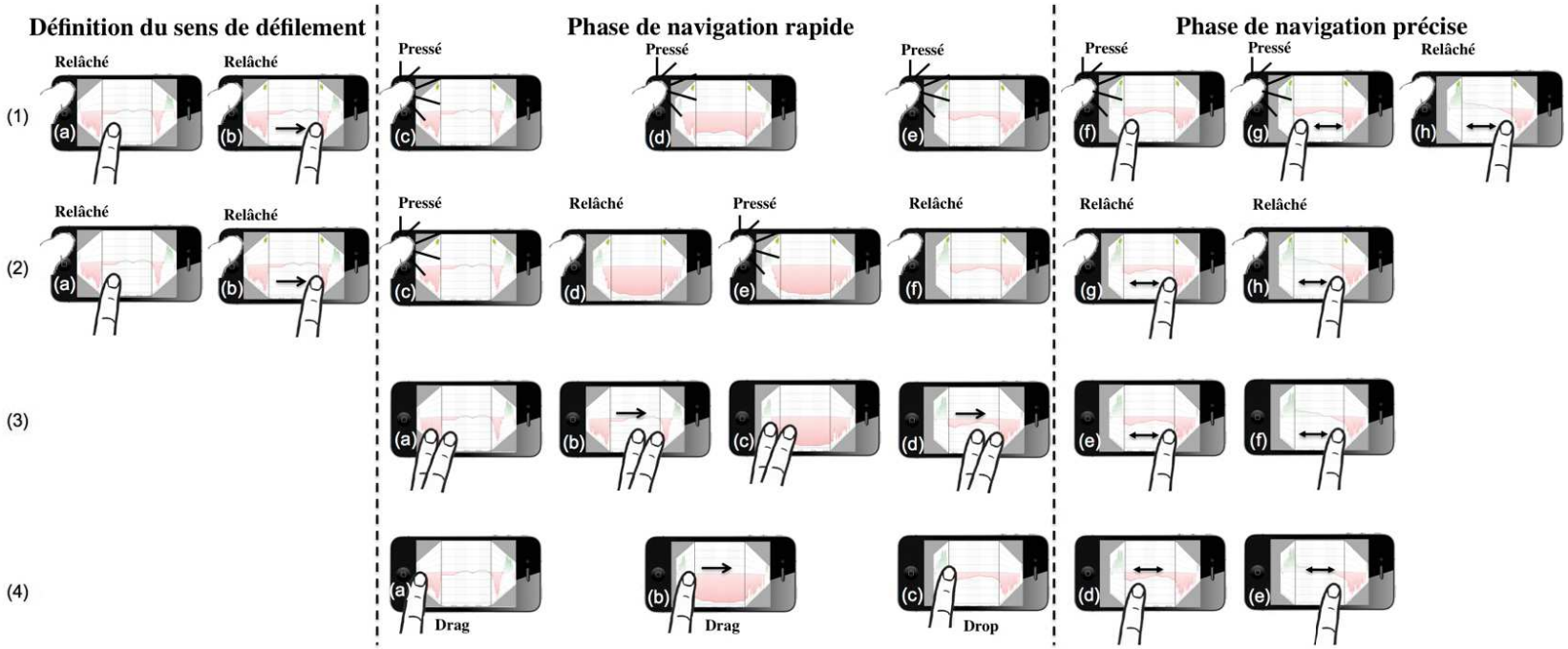


Figure 5.3 – Exemples de chemins d'interaction.

Ligne (1): C*ontinuous* - Action continue sur le capteur de pression. (a-b) geste de *Swipe* pour définir le sens de défilement (c-e) capteur pressé et le défilement continu (mouvement à l'échelle du contexte) (f) capteur non relâché, mais toucher l'écran arrête le défilement (g-h) gestes *Drag/Flick* précis effectués partout à l'écran (mouvements à l'échelle du focus).

Ligne (2): *Discrete* - Action discrète sur le capteur de pression. (a-b) geste de *Swipe* pour définir le sens de défilement (c-d) [pression et relâchement] fort provoquant un long déplacement dans l'espace d'information (mouvement à l'échelle du contexte) (e-f) [pression et relâchement] léger provoquant un court déplacement dans l'espace d'information (g-h) gestes *Drag/Flick* précis effectués partout à l'écran (mouvements à l'échelle du focus).

Ligne (3): *Flick*. La main non dominante (non représenté sur la figure) tient le dispositif. Les actions tactiles sont effectuées avec la main dominante. (a-d) Deux gestes *Flick* avec deux doigts nécessitant du clutching (mouvement à l'échelle du contexte). (e-f) gestes *Drag/Flick* précis avec un doigt effectués partout à l'écran (mouvements à l'échelle du focus).

Ligne (4): DirectTouch. (a-b-c) *Drag* et *Drop* (mouvement à l'échelle du contexte). (d-e) gestes *Drag/Flick* précis avec un doigt effectués partout à l'écran (mouvements à l'échelle du focus).

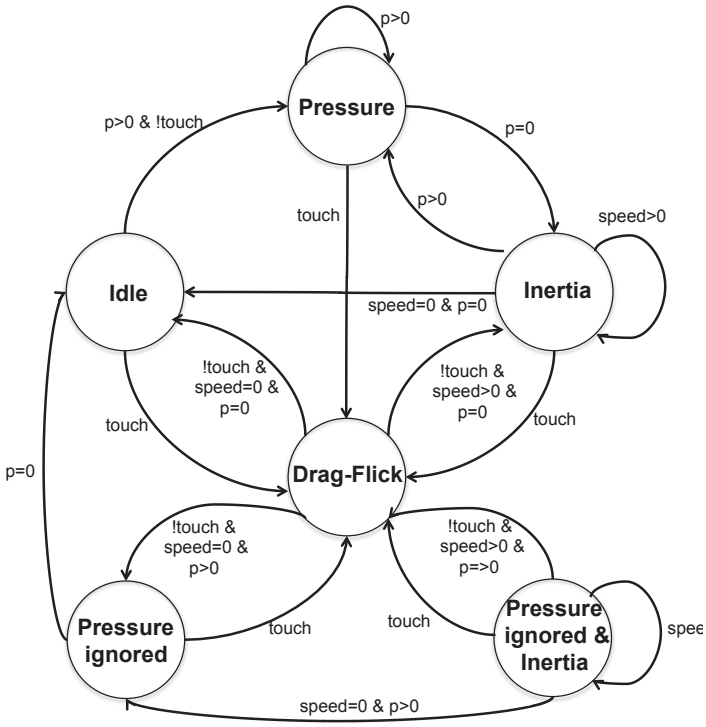


Figure 5.4 – Les quatres modalités basées sur la pression et leu avec la modalité *Flick*.

5.5.1 Modalités *Continuous*

Dès que le capteur est pressé, le mouvement de défilement com plus le capteur est pressé fort, plus la navigation est rapide. Lorsqu relâché, le mouvement de défilement s'arrête. Le type d'actions ut continu.

La conversion pression / vitesse est linéaire, ce qui permet d navigation entre 0 et 2800 pixels par seconde dans l'espac (respectivement 0 et 33600 pixels par seconde dans l'espace d vitesse maximale a été déterminée après une étude pilote compa facteurs de gain pour la fonction de transfert linéaire. Cette étude avec 10 participants et comparait 4 facteurs différents (permettant les vitesses maximales de 700, 1400, 2800 et 5600 pixels par se participant a réalisé 8 tâches pour chaque facteur. Nous avons den participant de choisir le facteur jugé le plus adéquat. 7 participant choisi celui permettant une vitesse maximale de 2800 pixels par avons donc sélectionné ce facteur avec la fonction de transfert suiva $V = 10 * p$, où 10 correspond au facteur retenu dans l'étude pilote, la pression retournée par la capteur de pression sous la forme d'un entre 0 et 280 (0 = 0N, 280 = 20N).

5.5.2 Modalité ContinuousInertia

Avec la modalité *ContinuousInertia*, lorsque le capteur est relâché, un effet d'inertie est ajouté. Cet effet provoque une décélération de la vitesse de défilement au fil du temps. Le type d'actions utilisées est donc continu.
La vitesse initiale utilisée pour l'effet d'inertie est basée sur la moyenne des cinq dernières valeurs de pression détectées avant qu'elle soit nulle. La vitesse de défilement diminue ensuite jusqu'à ce qu'elle atteigne 0 en utilisant l'équation suivante : $V(t+1) = V(t) \times 0{,}997^{dt}$, où dt est l'intervalle de temps entre deux calculs (30ms). En dessous de 1 pixel par seconde, le défilement s'arrête. Ainsi, plus la vitesse de défilement est élevée lorsque le capteur est relâché, plus la phase de décélération est longue et la distance parcourue importante. A tout moment, le défilement créé par l'effet d'inertie peut être arrêté avec une interaction tactile (Figure 5.4).

5.5.3 Modalité *Discrete*

Avec la modalité *Discrete*, rien ne se passe lorsque le capteur est enfoncé, et ce jusqu'à ce qu'il soit relâché. Le type d'actions utilisées est donc discret.
Lorsque le capteur est relâché, la moyenne des cinq dernières valeurs de pression définit la longueur du déplacement dans l'espace d'information. La conversion pression / déplacement est linéaire et permet un déplacement de 0 à 11000 pixels dans l'espace focus. Cela permet aux utilisateurs de parcourir les 57600 pixels de l'espace d'information avec environ 5 gestes de <appui-relâchement>.

5.5.4 Modalité DiscreteInertia

Comme la modalité précédente, la modalité *DiscreteInertia* ne déclenche pas de défilement jusqu'à ce que le capteur de pression soit relâché. Plutôt que de déclencher un déplacement dans l'espace d'information comme c'est le cas pour *Discrete*, *DiscreteInertia* provoque un effet d'inertie. Le type d'actions utilisées est donc discret.
Lorsque le capteur est relâché, la moyenne des cinq dernières valeurs de pression est utilisée en tant que vitesse initiale du défilement par inertie. L'équation de décélération est la même que pour *ContinuousInertia*. La plage de vitesse est de 0 à 2800 pixels par seconde dans l'espace du contexte (respectivement 0 et 33600 pixels par seconde dans l'espace du focus). Si les utilisateurs appuient et relâchent le capteur à nouveau au cours de la phase de décélération inertielle, les vitesses sont cumulées après le quatrième geste de <appui-relâchement> comme c'est le cas pour la modalité *Flick* sous iOS [101]. Nous avons pour cela utilisé la formule suivante pour calculer la vitesse de référence à utiliser pour l'effet d'inertie :
$V(t+1) = V(t) + V(p)$, où $V(p)$ est la vitesse de référence de l'effet d'inertie du nouveau geste de <appui-relâchement> et $V(t)$ la vitesse de défilement actuelle due à l'effet d'inertie en cours (au moment où le capteur à été pressé). Le défilement par inertie peut être arrêté avec une interaction tactile à tout moment (Figure 5.4).

5.6 Expérience

Nous avons conduit une évaluation contrôlée pour comparer les quatre modalités basées sur la pression décrites ci-dessus (*Continuous*, *ContinuousInertia*, *Discrete*,

DiscreteInertia) avec deux modalités tactiles qui utilisent respectivement les gestes *Drag/Flick* et *Drag/Drop* (nommée respectivement par la suite *Flick* et *DirectTouch*). Nous résumons les caractéristiques des modalités considérées pour la phase de navigation rapide dans la Table 5.1.

La métaphore d'interaction utilisée est la métaphore d'interaction du ruban. En effet, l'expérience 2 du chapitre 4 nous a révélé que la métaphore d'interaction de la loupe était moins appropriée à l'utilisation de modalités combinées : comme expliqué au chapitre 4 (section 4.3.2), cette métaphore n'est pas adaptée à une navigation précise avec les gestes tactiles. Cela nous a obligé à modifier le sens de défilement à utiliser pour les gestes de *Drag/Flick* durant la phase de navigation précise, Ainsi, durant la phase de navigation rapide, les gestes de *Drag/Flick* vers la gauche (respectivement vers la droite) déplaçaient la zone de focus vers la gauche (respectivement vers la droite), alors que durant la phase de navigation précise, les gestes de *Drag/Flick* vers la gauche (respectivement vers la droite) déplaçaient la zone de focus vers la droite (respectivement vers la gauche) comme illustré sur la Figure 4.15. Cette ambiguïté à perturbé certains utilisateurs, et nous a donc incité à utiliser à nouveau la métaphore de l'interaction du ruban, ne créant pas d'ambiguïté entre les deux phases.

Type d'action / Inertie	Discrète	Continue
Oui	DiscreteInertia	ContinuousInertia, Flick
Non	Discrete	Continuous, DirectTouch

Table 5.1 – 6 modalités pour la phase de navigation rapide.

L'évaluation nous a permis d'étudier les questions de recherche suivantes liées au rationnel de conception des modalités basées sur la pression :

- Q1 : Les modalités basées sur la pression peuvent-elles obtenir de meilleures performances que les modalités tactiles ?
- Q2 : Y a t-il un avantage à l'utilisation de modalités basées sur des actions continues au lieu de discrètes sur le capteur de pression ?
- Q3 : Quels sont les effets de l'inertie sur la phase de navigation rapide et celle de transition ?

5.6.1 Modalités tactiles utilisées

La première modalité, notée *Flick*, utilise des gestes de *Drag/Flick* dans l'espace du contexte avec deux doigts, pour les distinguer des gestes de *Drag/Flick* dans l'espace du focus effectués avec un doigt pendant la phase de navigation précise (Figure 2, ligne 3).
Ainsi, les utilisateurs peuvent facilement basculer entre les deux modalités et effectuer des gestes de *Drag/Flick* partout à l'écran. Un déplacement de un pixel à l'écran avec deux doigts provoque un déplacement de un pixel dans l'espace du contexte, et donc un déplacement de *fz* pixels dans l'espace du focus (*fz* étant le facteur de zoom entre le focus et le contexte de la vue bifocale). Les directions des gestes sont celles habituellement utilisés sur les dispositifs mobiles avec écrans tactiles : un geste de *Drag/Flick* vers la gauche (respectivement vers la droite) déplace les données vers la gauche (respectivement vers la droite). Cette modalité utilise un mode de navigation relatif avec des actions continues (Table 5.1).

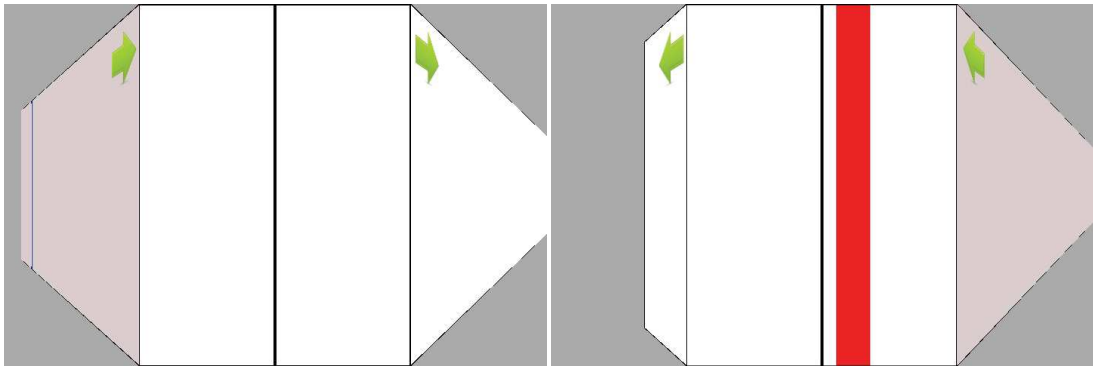
La seconde modalité, à savoir *DirectTouch*, utilise des gestes de *Drag/Drop*. Les utilisateurs sélectionnent directement un point dans la zone de contexte pour le relâcher dans la zone de focus. Ainsi la modalité exploite pleinement les possibilités d'une vue Focus + Contexte.
Comme les gestes de *Drag* dans la zone de contexte définissent les actions de *Drag/Drop*, les utilisateurs ne peuvent effectuer des gestes de *Drag/Flick* de la phase de navigation précise uniquement dans la zone de focus (Figure 2, ligne 4). Cette modalité utilise un mode de navigation relatif avec des actions continues (Table 5.1).

Les deux modalités tactiles définissent deux façons différentes pour distinguer les deux phases de navigation : *Flick* est basée sur des gestes différents (un / deux doigts) alors que *DirectTouch* implique des gestes similaires dans différentes zones de l'écran. Au contraire les modalités basées sur la pression avec la main non dominante enrichissent le vocabulaire des événements et nous permettent d'attribuer une modalité distincte par phase de navigation.

5.6.2 Tâche expérimentale

Nous avons réutilisé la tâche de l'expérience 2 du chapitre 4, comme illustré à la Figure 5.5, avec quelques modifications.

Les participants devaient sélectionner des cibles dans la zone de contexte. Ils devaient mettre la cible sur un curseur affiché au centre de la zone de focus et la maintenir pendant 1 seconde : la cible rouge devenait alors verte lorsqu'elle était positionnée sur le curseur. Après cela, une nouvelle cible apparaissait.



(a) **(b)**

Figure 5.5 – Tâche expérimentale: Les flèches indiquent le sens de défilement. (a) La cible est dans le côté gauche de la zone de contexte. (b) La cible est dans la zone de focus, à droite du curseur affiché au centre (ligne noire).

Les cibles étaient générées de manière équitable à gauche et à droite. Elles étaient représentées par des zones rouges de 60 pixels de largeur. Le curseur était représenté au centre de la zone de focus par une grande ligne noire de 6 pixels de largeur (10 fois plus petite que les cibles). Ainsi, les cibles pouvaient être facilement sélectionnées. En effet la tâche concrète est de déplacer des points d'intérêt dans la zone de focus afin d'obtenir des détails et non de pointer

précisément. Notre objectif n'est donc pas d'évaluer la précision de *Drag/Flick* dans la zone de focus (phase de navigation précise). Nous rendons donc cette phase facile à réaliser en termes de précision.

En raison de l'effet de perspective, et du facteur de zoom, la cible dans la vue de contexte n'était pas toujours clairement visible (notamment lorsqu'elle était très éloignée de la zone de focus). Pour résoudre ce problème, la couleur de fond du côté du contexte contenant la cible était modifiée (Figure 5.5.a) afin d'indiquer à l'utilisateur de quel côté la cible se trouvait et ainsi minimiser l'effet de « desert fog » [63] : les participants savaient donc toujours dans quelle direction ils devaient naviguer. De plus, la cible était entourée par un rectangle bleu dans la zone de contexte afin d'être toujours perceptible peu importe sa distance à la zone de focus (Figure 5.5.a).

Enfin des flèches dans les deux zones de contexte (Figure 5.5) indiquaient le sens de défilement de l'espace d'information.

Nous avons utilisé des cibles de largeur fixe (W = 60 pixels) et considéré 4 distances différentes : D1 = 4800 pixels, D2 = 9600 pixels, D3 = 19200 pixels et D4 = 38400 pixels. Ces distances dans l'espace du focus correspondent respectivement à 400, 800, 1600, et 3200 pixels dans l'espace du contexte. D1 est accessible avec un seul geste de *Flick* dans l'espace du contexte. Les autres distances sont tout simplement D1 multipliée par 2, 4 et 8 (D2 = 2 * D1, D3 = 2 * D2, D4 = 2 * D3). Ainsi, ces distances définissent quatre niveaux de difficulté pour la tâche (IDi = log2 (Di / 60 + 1), respectivement ID1 = 6.33, ID2 = 7.33, ID3 = 8.32 et ID4 = 9.32.

5.6.3 Participants et procédure

Dix-huit bénévoles (dont huit femmes), âgés de 15 à 38 ans (moyenne 25, médiane 27), ont participé à l'expérience. Ils étaient tous experts en interaction tactile sur dispositif mobile, car ils étaient utilisateurs réguliers d'un smartphone avec un écran tactile.

Les participants étaient assis et tenaient le dispositif mobile comme le montre la Figure 5.1. La condition de mobilité n'a pas été considérée. Tout d'abord Wilson et al. [132] ont montré que l'interaction basée sur la pression reste utilisable pendant la marche. Deuxièmement, la navigation et l'exploration d'une longue liste de données correspond à une tâche que les utilisateurs effectuent plutôt debout ou assis, mais de manière immobile (par exemple, une liste des mails, de musique ou les produits d'un site Web de commerce en position debout ou assise dans le tramway ou à la maison).

Nous avons d'abord expliqué le principe de la vue bifocale en présentant aux participants un exemple concret avec des données sur le marché boursier. Ensuite, l'évaluation a été divisée en 6 blocs, un par modalité. L'ordre des blocs a été contrebalancé par un carré latin.
Pour chaque bloc, nous avons d'abord décrit la modalité, et les participants ont du réaliser 6 tâches d'entraînement. Ils pouvaient alors poser toutes les questions qu'ils désiraient au cours de cette étape. Ils ont ensuite effectué 32 tâches de navigation (8 tâches par indices de difficulté). Ils ont finalement rempli un

questionnaire SUS [17] (Annexe 2) sur la modalité qu'ils venaient d'utiliser afin de recueillir leur point de vue subjectif sur sa facilité d'utilisation. Enfin, après les 6 blocs, les participants ont été invités à classer les 6 modalités par ordre de préférence et à expliquer leurs choix. L'expérience entière a duré environ 45 minutes pour chaque participant.

5.6.4 Analyse du temps d'exécution par phase

Pour chaque modalité, nous définissons la phase de navigation rapide comme le temps passé à utiliser la modalité dédiée à la navigation rapide (à l'échelle du contexte).

- Pour les modalités basées sur la pression, elle commence lorsque les utilisateurs touchent l'écran pour définir le sens de défilement.
- Pour les modalités avec un effet d'inertie, elle se termine lorsque l'animation d'inertie se termine ou lorsque les utilisateurs touchent l'écran pour utiliser la modalité précise. Pour les modalités sans effet d'inertie, la phase de navigation rapide se termine lorsque les utilisateurs enlèvent leur doigt du capteur de pression (pour les modalités basées sur la pression) ou de l'écran (pour les modalités tactiles).
- Pour toutes ces modalités, si des actions successives sont effectuées (plusieurs gestes de <appui-relâchement> pour la pression, plusieurs gestes de *Drag*/*Flick* pour l'interaction tactile), cette phase comprend le temps de chacune des actions ainsi que le temps passé entre chacune d'elles.

La phase de navigation précise est définie par le temps passé à utiliser la modalité précise (à l'échelle du focus). Elle commence lorsque les utilisateurs touchent l'écran et se termine quand ils enlèvent leur doigt de l'écran (geste de *Drag*) ou lorsque l'effet d'inertie s'arrête (geste de *Flick*). Comme pour la phase précédente, si des actions successives sont effectuées à l'écran (clutching), cette phase comprend le temps passé à effectuer chacun des gestes, ainsi que le temps passé entre chacun d'eux.

La phase de transition est définie comme le temps écoulé entre ces deux phases. Ainsi, si les utilisateurs mettent le doigt sur l'écran tandis que le capteur de pression est enfoncé, la phase de transition est nulle. Dans le cas contraire, il est défini comme le temps écoulé entre le moment où la navigation déclenchée par le capteur de pression cesse et le moment où les utilisateurs mettent le doigt sur l'écran.

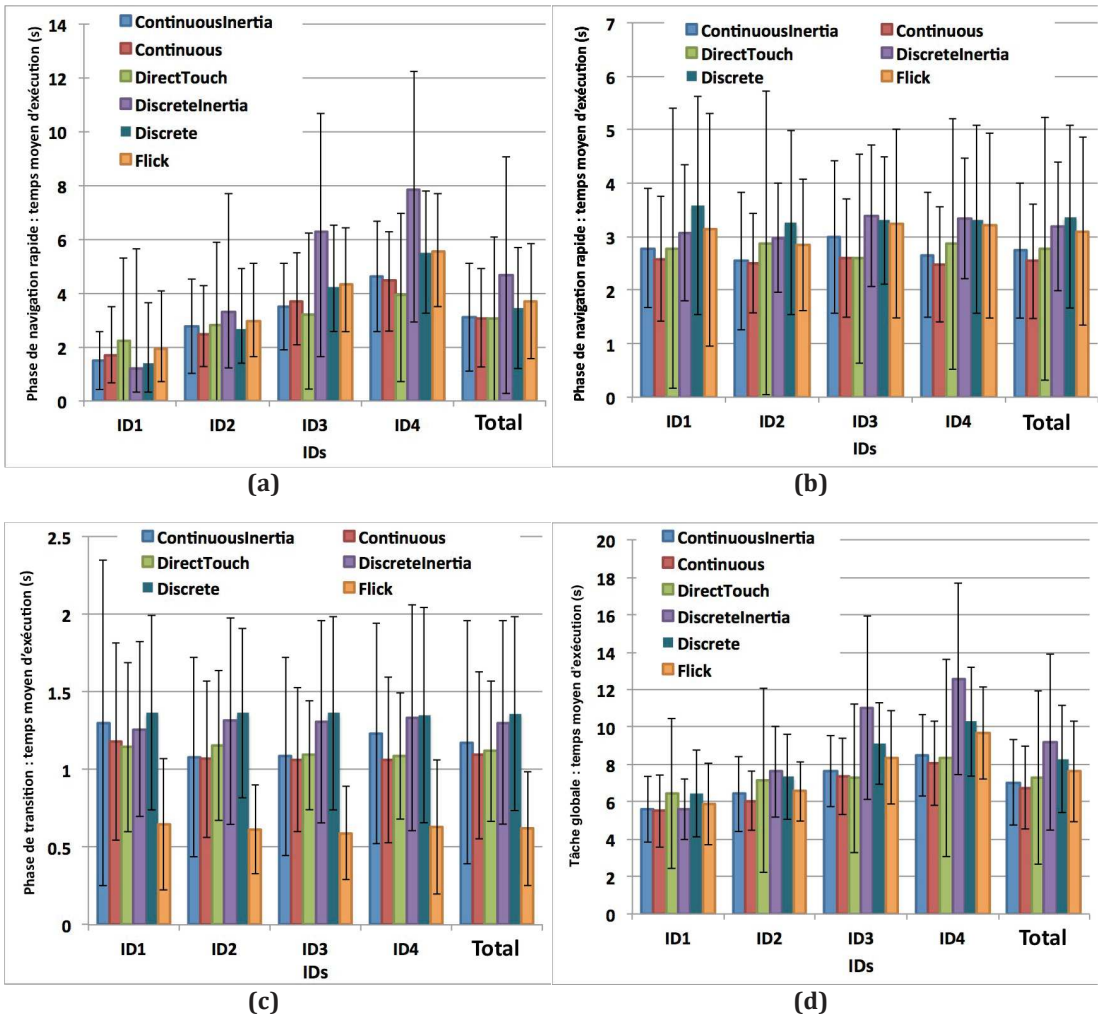


Figure 5.6 – (a) Temps moyens pour la phase de navigation rapide. (b) Temps moyens pour la phase de navigation précise. (c) Temps moyens pour la phase de transition. (d) Temps moyens pour la tâche gobale avec un intervalle de confiance correspondant de 95%.

5.6.5 Résultats

5.6.5.1 Performances : temps d'exécution

Pour chaque tâche, nous avons enregistré tous les événements captés par un des dispositifs des modalités considérées (événements tactiles et pression). Une analyse à postériori des fichiers générés nous a permis de savoir pour chaque tâche, quelle modalité était utilisée et pendant combien de temps. Nous avons vérifié la normalité de nos données en utilisant le test de Shapiro-Wilk. Elle a révélé un biais dans la distribution des données et aucune des transformations (logaritmique ou racine carrée) généralement utilisées avec ce type d'écart n'a normalisé les données. Nous avons donc effectué le test non paramértique de Friedman afin de tester l'effet de chaque facteur (Modalité et Indice de Difficulté-ID) et de la combinaison de facteurs (Modalité x Indice de difficulté -ID) sur chaque variable dépendante (temps d'exécution pour chaque phase). Nous avons également utilisé le test de Wilcoxon T avec une correction de Bonferroni pour les comparaisons par paires.

5.6.5.1.1 Temps d'exécution pour la phase de navigation rapide

Le test de Friedman a révélé un effet significatif du facteur Modalité sur le temps de cette phase. ($X2$ (5) = 185, $p < 0.0001$). Une comparaison des paires avec le test de Wilcoxon et une correction de Bonferroni a montré que *Flick* (moyenne = 3.70s) et *DiscreteInertia* (moyenne = 4.66s) sont significativement plus lentes que toutes les autres modalités ($p < 0.05$). *Discrete* (moyenne = 3.45s) est significativement plus lente ($p < 0.0001$) que *DirectTouch* (moyenne = 3.04s), *Continuous* (moyenne = 3.08s) et *ContinuousInertia* (moyenne = 3.09s). *DirectTouch* est significativement plus rapide ($p < 0.005$) que *Continuous* et *ContinuousInertia*.

En ce qui concerne l'indice de difficulté (Figure 5.6.a), le test de Wilcoxon a montré que *DiscreteInertia* est significativement plus lente que les autres modalités ($p < 0.005$) pour ID3 et ID4 tandis que *DirectTouch* est significativement plus rapide que toutes les autres modalités ($p < 0.05$) pour les mêmes ID. *Flick* et *Discrete* sont significativement plus lentes que *ContinuousInertia* pour ID3 et ID4 ($p <0.05$) alors qu'elles sont significativement plus lentes que *Continuous* seulement pour ID4 ($p <0.0001$). Pour ID2, DiscreteInertia est significativement plus lente que toutes les autres modalités ($p < 0.05$), sauf *Flick*. Enfin, pour ID1, *Discrete* et *DiscreteInertia* sont significativement plus rapides que *Continuous*, *Flick* et *DirectTouch* ($p < 0.05$).

5.6.5.1.2 Temps d'exécution pour la phase de navigation précise

Le test de Friedman a révélé un effet significatif du facteur Modalité sur le temps de cette phase ($X2$ (5) = 150, $p < 0.0001$). Une comparaison des paires avec le test de Wilcoxon et une correction de Bonferroni a révélé que *Flick* (moyenne = 3.10s), *Discrete* (moyenne = 3.36s) et *DiscreteInertia* (moyenne = 3.19s) sont significativement plus lentes ($p < 0.05$) que *DirectTouch* (moyenne = 2.79s), *Continuous* (moyenne = 2.74s) et *ContinuousInertia* (moyenne = 2.54s). *ContinuousInertia* est significativement plus rapide que *Continuous* ($p < 0.005$).

En ce qui concerne l'indice de difficulté (Figure 5.6.b), le test de Wilcoxon a montré que *Discrete* est significativement plus lente que *DirectTouch*, *Continuous*, *ContinuousInertia* ($p < 0.01$) pour ID1 tandis que *DiscreteInertia* est significativement plus lente que *DirectTouch* et *Continuous* ($p < 0.01$) pour la même distance. Pour ID2, *Discrete* est significativement plus lente que *DirectTouch*, *Continuous*, *ContinuousInertia* ($p < 0.01$), tandis que *DiscreteInertia* est significativement plus lente que *ContinuousInertia et* Continuous ($p < 0.01$). Pour ID3, *DirectTouch* est significativement plus rapide que *ContinuousInertia*, *Discrete*, *DiscreteInertia* et *Flick* ($p < 0.05$), tandis que *Continuous* est significativement plus rapide que *Discrete*, *DiscreteInertia* et *Flick* ($p < 0.05$). Enfin, pour ID4, *Discrete* et *DiscreteInertia* sont significativement plus lentes que *DirectTouch*, *Continuous* et *ContinuousInertia* ($p < 0.05$). *Continuous* est significativement plus rapide que *Flick* ($p < 0.05$).

5.6.5.1.3 Temps d'exécution pour la phase de transition

Le test de Friedman a révélé un effet significatif du facteur Modalité sur le temps de transition ($X2$ (5) = 679, $p < 0.0001$). Une comparaison des paires avec le test de Wilcoxon et une correction de Bonferroni a montré que *Flick* (moyenne = 0.62s) est plus rapide que toutes les autres modalités ($p < 0.0001$). *DirectTouch* (moyenne = 0.83s), *Continuous* (moyenne = 1.09s) et *ContinuousInertia* (moyenne = 1.17s) sont significativement plus rapides ($p < 0.0001$) que *Discrete* (moyenne = 1.37s) et *DiscreteInertia* (moyenne = 1.30s).

En ce qui concerne le l'indice de difficulté (Figure 5.6.c) le test de Wilcoxon a montré que *Flick* est plus rapide que toutes les modalités pour tous les ID ($p < 0.0001$). *Discrete* est significativement plus lente que *Continuous* et *DirectTouch* pour ID1. Pour ID2 et ID3, *Discrete* est significativement plus lente que *DirectTouch* ($p < 0.005$), *Continuous* et *ContinuousInertia* tandis que *DiscreteInertia* est significativement plus lente que *Continuous* et *ContinuousInertia* ($p < 0.05$). Pour ID4, *Discrete* et *DiscreteInertia* sont significativement plus lentes que *DirectTouch* et *Continuous* ($p < 0.005$).

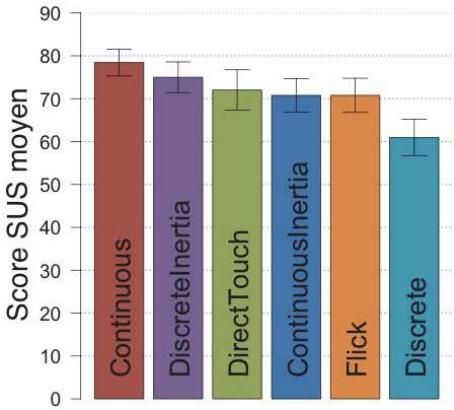
5.6.5.1.4 Temps d'exécution global

Le test de Friedman a révélé un effet significatif du facteur Modalité sur le temps total d'exécution ($X2$ (5) = 312, $p < 0.0001$). Une comparaison des paires avec le test de Wilcoxon et une correction de Bonferroni a montré que *DirectTouch* (moyenne = 7.29s), *Continuous* (moyenne = 7.02s) et *ContinuousInertia* (moyenne = 6.74s) sont significativement plus rapides ($p < 0.001$) que *Discrete* (moyenne = 8.29s), *DiscreteInertia* (moyenne = 9.20s) et *Flick* (moyenne = 7.61s). *DiscreteInertia* est significativement la modalité la plus lente ($p < 0.0001$), tandis que *Flick* est significativement plus rapide que *Discrete* ($p < 0.0001$).

En ce qui concerne l'indice de difficulté (Figure 5.6.d), le test de Wilcoxon a montré que *Discrete* est significativement plus lente que *Continuous*, *ContinuousInertia* et *DiscreteInertia* pour ID1 ($p < 0.05$). Pour ID2, *DiscreteInertia* est significativement plus lente que *Flick*, *Continuous*, *ContinuousInertia* et *DirectTouch* ($p < 0.01$), tandis que *Discrete* est significativement plus lente que *Continuous* et *ContinuousInertia* ($p < 0.001$). Pour ID3, *Discrete* et *DiscreteInertia* sont significativement plus lentes que toutes les autres modalités ($p < 0.05$). *Flick* est nettement plus lente que *Continuous* et *DirectTouch* ($p < 0.05$), tandis que *ContinuousInertia* est significativement plus lente que *DirectTouch* ($p < 0.05$). Enfin, pour ID4, *Discrete* est sensiblement la modalité la plus lente ($p < 0.0001$), tandis que *DiscreteInertia* et *Flick* sont significativement plus lentes que *DirectTouch*, *Continuous* et *ContinuousInertia* ($p < 0.01$).

5.6.5.2 Préférences des utilisateurs

Les préférences des utilisateurs ne reflètent pas entièrement les résultats de performance. Dans la Figure 5.7.a, les modalités sont ordonnées de gauche à droite par ordre décroissant de score moyen SUS. Un test de Friedman a révélé un effet significatif du facteur Modalité sur le score SUS ($X2$ (5) = 15, $p < 0.01$). Un test de Wilcoxon avec une correction de Bonferroni a montré que *Discrete* a été notée significativement moins bien que *Continuous* et *DiscreteInertia* ($p < 0.05$). Toutes les autres modalités ne sont pas significativement différentes.



(a)

Technique	#1	#2	#3	Score
Continuous	7	6	3	36
DirectTouch	7	2	3	28
ContinuousInertia	1	3	7	16
DiscreteInertia	1	5	2	15
Flick	2	2	3	13
Discrete	0	0	0	0

(b)

Figure 5.7 – (a) Scores moyens SUS avec des intervalles de confiance à 95% (b) Classement par ordre de préférence et nombre de fois où les modalités ont été classées 1er, 2ème ou 3ème.

Le classement de préférence de la Figure 5.7.b n'est pas tout à fait similaire aux scores SUS moyens. Les modalités préférées sont les plus rapides, mais sont différentes des trois modalités jugées les plus utilisables. Nous avons calculé le score de classement (S) d'une modalité en utilisant la formule S = 3x1^{er} +2x2^{e} + 3^{e}, où 1er, 2e et 3e sont le nombre de fois où la modalité a été classée à la position correspondante. Nous avons vérifié qu'avec de plus grands coefficients (respectivement 5, 3, 1), les résultats restaient similaires.

5.6.6 Discussion

5.6.6.1 Pression vs Tactile

Continuous et *DirectTouch* ont été les modalités les plus rapides et ont été les mieux classées par les utilisateurs. Dans un contexte différent d'utilisation et de tâche, ceci confirme les résultats d'études antérieures [89, 119] montrant que les modalités basées sur la pression peuvent être aussi efficaces voire surpasser les modalités tactiles (respectivement *DirectTouch* et *Flick*) tout en minimisant l'occultation de l'écran.

Flick a été décrite comme inconfortable. En effet, les utilisateurs de dispositifs mobiles avec écran tactile sont habitués à interagir avec l'écran en utilisant leurs pouces. De cette façon, le défilement vers la gauche ou vers la droite implique simplement des mouvements de l'articulation du pouce (Figure 5.8.a). La modalité Flick consistait à utiliser deux doigts pour effectuer une navigation rapide à l'échelle du contexte. Pour ce faire, tous les participants ont utilisé l'index et le majeur pour interagir. La plupart d'entre eux orientaient leurs doigts dans le sens de défilement : cela les contraignait à effectuer une légère rotation du poignet (Figure 5.8.b) et créait ainsi de l'inconfort rapporté par plusieurs participants. De plus, nous avons observé certains participants ayant des problèmes avec la reconnaissance tactile en raison de leurs ongles, un problème déjà observé par Spindler et al. [122]. Les très mauvais résultats obtenus par la modalité *Flick* peuvent donc être expliqués par cette position inhabituelle. Ainsi, nous pouvons conclure que l'approche de désambiguïsation en utilisant un ou deux doigts pour effectuer des gestes de *Flick* dérange les habitudes des utilisateurs de dispositifs

mobiles avec écran tactile et conduit à de mauvaises performances. L'approche consistant à associer chacun des modes (phases de navigation rapide et précise) à une zone de l'écran semble donc une meilleure solution, comme nous montre les résultats de *DirectTouch*. En effet, en tenant le dispositif mobile à deux mains (Figure 5.8.a), l'interaction avec le pouce est possible avec chacune des mains. Ainsi, le pouce de la main gauche (respectivement droite) peut interagir dans la zone de contexte à gauche (respectivement à droite) de la zone de focus pour la phase de navigation rapide, et chacun d'eux peut également interagir dans la zone de focus pour la phase de navigation précise.

Pour la phase de transition, la modalité *Flick* était nettement plus rapide que les autres modalités. Cependant, les participants n'ont pas perçu une différence assez significative pour la commenter. Néanmoins, les modalités basées sur des actions discrètes sur le capteur de pression ont montré une plus longue phase de transition. Ceci peut être expliqué par l'effet de désorientation due à la perte de contrôle des mouvements comme nous l'avions observé lors l'expérience 2 du chapitre 4 pour les modalités *DiscretePress* et *DirectTouch*.

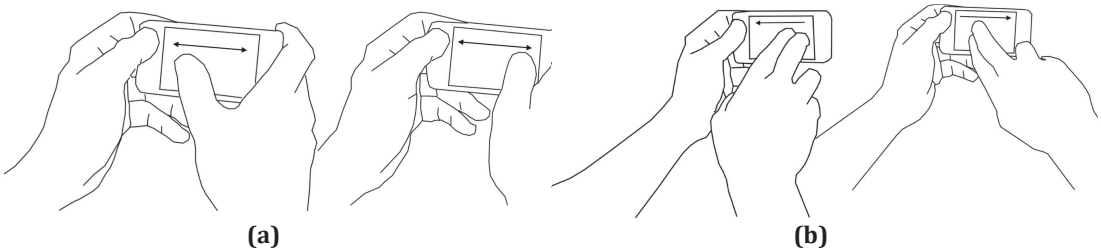


Figure 5.8 – Modalité *Flick*: les postures de la main de l'utilisateur (a) avec un doigt et (b) avec deux doigts.

5.6.6.2 Actions Continues vs Discrètes

Les modalités basées sur des actions continues sur le capteur de pression ont fourni de meilleurs résultats que celles nécessitant des actions discrètes. Les participants ont exprimé que le contrôle de la plage de pression a été difficile avec les actions discrètes. Certains d'entre eux nous ont dit qu'ils avaient utilisé ces modalités « au hasard » parce qu'ils ne savaient pas quelle pression ils appliquaient sur le capteur. Nous pensons que ce problème peut être réduit après une période d'apprentissage plus longue. Le contrôle de la pression était plus facile avec les actions continues en raison du retour visuel en temps réel. Les mécanismes de retour continu ont été mis en évidence et définis comme importants dans [119, 133]. Une solution pourrait donc être d'ajouter un retour visuel pour indiquer la force appliquée sur le capteur ou la longueur du déplacement, mais cela perturberait l'attention de l'utilisateur comme indiqué dans [55]. Nous suggérons donc plutôt l'ajout d'un retour haptique au capteur de pression (améliorer le matériel pour une meilleure sensation de la pression appliquée). Cela permettrait aux utilisateurs d'être conscients de la pression qu'ils appliquent, tout en restant concentrés sur l'espace d'information.

Enfin, certains participants ont trouvé la modalité basée sur les actions discrètes fastidieuse en raison des gestes de pression répétitifs qu'ils devaient réaliser. Mais certains participants ont également signalé que les gestes de pression successifs semblent plus « naturels » que des actions continues sur le capteur. Ceci peut être

expliqué par le fait que les participants sont habitués à utiliser des séries de *Flick* (avec clutching) lorsqu'ils naviguent sur leurs dispositifs mobiles. Les modalités basées sur des actions discrètes sur le capteur de pression impliquent également une série d'actions (séries de <appui-relâchement> sur le capteur de pression). Ainsi, certains participants, en raison de la similitude avec le geste de *Flick*, ont apprécié utiliser le capteur de pression en effectuant des actions discrètes. *DiscreteInertia* a donné lieu à de bons résultats pour la distance la plus courte. Cela nous permet de penser que les utilisateurs sont plus à l'aise avec les niveaux de pression les plus faibles, ce qui contredit les résultats subjectifs de Stewart & al. [123]. Ainsi, essayer d'améliorer les modalités basées sur des actions discrètes pourrait conduire à de bonnes performances et de bonnes évaluations des utilisateurs.

5.6.6.3 Inertie vs Sans inertie

Contrairement à ce à quoi nous nous attendions, l'inertie n'a pas été très utile. Elle n'a pas eu d'effet sur la phase de transition. On aurait pu penser que l'effet d'inertie réduirait cette phase de transition en permettant aux utilisateurs d'anticiper l'action suivante selon la modalité tactile. Mais les résultats de performance ne le confirment pas.

Les utilisateurs ont été partagés quant à son utilité. D'une part, la plupart d'entre eux ont trouvé qu'elle augmentait la charge cognitive : ils ont trouvé qu'il était difficile d'anticiper quand relâcher le capteur afin de mieux utiliser le défilement par inertie. Ils devaient donc se concentrer sur ce problème. D'autre part, plusieurs participants ont indiqué que le défilement par inertie leur a permis de passer plus facilement à la modalité précise (phase de navigation rapide), car ils ne sont pas en train d'interagir avec l'appareil lorsque le défilement par inertie est actif. En outre, la plupart des participants nous ont dit qu'ils seraient probablement plus efficaces avec plus d'expérience. Cela est surprenant, car l'inertie est bien connue des utilisateurs de dispositifs mobiles avec écran tactile. Nous avons appliqué une loi de puissance de la pratique (« power law of practice ») sur nos données afin d'étudier la courbe d'apprentissage des modalités en utilisant le défilement par inertie : nous n'avons pas trouvé d'effet d'apprentissage.

De plus, l'inertie a eu plus d'effet sur les modalités basées sur des actions discrètes que sur celles basées sur des actions continues concernant la phase de navigation rapide. En effet, *Continuous* et *ContinuousInertia* ont montré des performances similaires alors que *Discrete* a montré des performances significativement meilleures que *DiscreteInertia*.
En fait, la plupart des utilisateurs effectuaient de nombreuses actions de pression à faible intensité, en particulier pour les longues distances (12.89 actions de <appui-relâchement> en moyenne pour *DiscreteInertia* et 9.78 pour *Discrete*), plutôt que d'effectuer moins d'actions à plus haute intensité. Comme pour la modalité *Flick*, cela entraîne un défilement à vitesse presque constante et assez lente, donnant l'impression d'un défilement continu. Ce comportement n'a pas été observé avec la modalité *Flick* parce que les utilisateurs sont habitués à l'utiliser et donc expérimentés. Cette utilisation de *DiscreteInertia* explique ses mauvaises performances, étant plus lente que *Discrete*.
Néanmoins cette utilisation diminue également le nombre de dépassements de la cible (overshoot) en raison de la difficulté à contrôler les modalités basées sur la

pression par actions discrètes (voir section ci-dessus). En effet, il y avait plus du double de dépassements avec *Discrete* (0.45 en moyenne) qu'avec *DiscreteInertia* (0.21 en moyenne) pour les longues distances. Par conséquent, nous pensons qu'une étude plus longue devrait se traduire par des performances différentes pour les modalités utilisant le défilement par inertie, en particulier avec *DiscreteInertia*.

5.6.6.4 Combinaison des facteurs d'action et d'inertie

Continuous a obtenu de meilleures performances et a été préférée à *ContinuousInertia*. *Discrete* a été moins bien classée que DiscreteInertia en dépit de meilleures performances. Nous pouvons alors anticiper qu'une étude plus longue permettant aux utilisateurs de mieux utiliser l'effet d'inertie peut conduire à des résultats différents, en particulier dans le cas d'actions discrètes. En effet, *DiscreteInertia* correspond à une version basée sur la pression de la modalité tactile *Drag/Flick* bien connue des utilisateurs. Les participants ont d'ailleurs fait des commentaires positifs à ce sujet (voir la section Continu vs Discret).

5.7 Conclusion

En nous basant sur le modèle de navigation en trois phases décrit dans le chapitre 3, nous avons exploré dans ce chapitre les modalités basées sur un capteur de pression pour la navigation rapide, couplé à une modalité tactile pour la phase précise. Nous avons donc étendu l'exploration de l'espace des possibilités de la navigation multimodale dans une vue bifocale que nous avions entamée dans le chapitre 4. Contrairement au chapitres 4, nous considérons dans ce chapitre une interaction bi-manuelle, assignant une main à chaque modalité et donc à chaque phase de navigation.

Pour les modalités basées sur le capteur de pression, nous avons considéré le type d'action (continu / discret) qui caractérise le langage de la modalité (section 3.3.1 du chapitre 3) mais aussi l'effet d'inertie. La combinaison de ces deux caractéristiques nous a permis de concevoir quatre modalités basées sur le capteur de pression.

Nous avons montré grâce à une étude expérimentale qu'une interaction basée sur la pression était efficace pour naviguer dans une vue bifocale sur mobile. En particulier la modalité impliquant des actions continues sur le capteur avec ou sans inertie :

- surpasse la modalité tactile basée sur les gestes de *Drag/Flick* qui est bien connue des utilisateurs ;
- montre des performances de temps équivalentes à la modalité tactile basée sur les gestes de *Drag/Drop ;*
- réduit les problèmes d'occultation à l'écran créés par les modalités purement tactiles.

Alors que la modalité impliquant des actions continues sur le capteur de pression offre de bonnes performances, la modalité impliquant des actions discrètes sur le capteur définit une modalité qui mérite d'être approfondie. En effet, cette modalité implique une série d'actions courtes <appui-relâchement> et simule, en utilisant un capteur de pression, les gestes de *Drag/Flick* bien connus des utilisateurs.

Cette modalité peut être améliorée en étudiant un retour d'information prédictif (feedforward) pour aider l'utilisateur à mesurer la pression appliquée. Le retour d'information prédictif continu a été souligné comme important pour l'utilisation de modalités basées sur la pression dans plusieurs études [119, 133]. Nous confirmons ce constat : l'absence de retour d'information a été soulignée à la fois pour la modalité impliquant des actions discrètes mais aussi pour anticiper l'inertie basée sur la dernière pression appliquée. Nous identifions deux pistes pour améliorer la modalité par actions discrètes sur le capteur de pression :

- Ajouter un retour haptique au capteur de pression afin de mieux percevoir la pression appliquée avant validation et début de déplacement ;
- Commencer le défilement des données lors de l'appui du capteur et permettre de répéter les actions comme une pompe. La métaphore d'interaction du monde réel utilisée serait alors celle d'une toupie à pompe et friction (Figure 5.9).

Figure 5.9 – Toupie à pompe et friction[9].

[9] http://ludopetit.com/exposition/expo-toupie/

6 CONCLUSION

Nos travaux de thèse en Interaction Homme-Machine ont concerné la visualisation interactive de données sur dispositifs mobiles. A la croisée de deux axes de recherche, nos travaux sont dédiés :

- aux techniques de visualisation sur dispositifs mobiles permettant de présenter à la fois une vue globale de l'espace d'information et une sous-partie détaillée de celui-ci ;
- aux techniques de navigation permettant de naviguer dans cet espace d'information.

Pour conclure ce manuscrit, nous présentons dans les deux sections suivantes les contributions apportées par nos travaux, ainsi que les pistes de recherche pouvant étendre ces travaux.

6.1 Démarche scientifique et contributions

Nos travaux sont dédiés à la visualisation interactive de données sur dispositifs mobiles. Dans ce contexte d'étude, la démarche scientifique s'est inscrite dans une approche itérative qui allie induction et déduction (Figure 6.1) :

- Au sein d'une approche par déduction partant de modèles conceptuels pour concevoir et évaluer de nouvelles formes d'interaction, il convient de prendre en compte les facteurs humains (et leurs modélisations) mais aussi les facteurs liés aux contraintes des dispositifs mobiles actuels tant du point de vue de l'action que de la perception.
- Au sein d'une approche par induction partant de l'expérimentation, de nouvelles techniques d'interaction pour enrichir des modèles conceptuels et formes d'interaction doivent être conçues et développées.

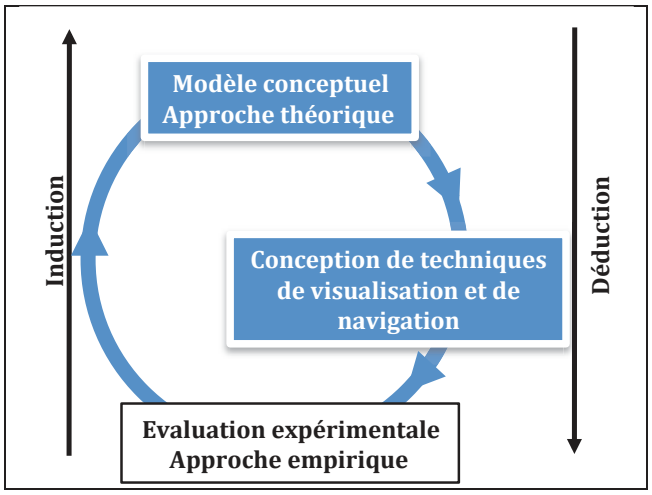


Figure 6.1 – Démarche scientifique et contributions.

Selon cette démarche, nos contributions sont donc de deux ordres : conceptuelles mais aussi pratiques sous la forme de techniques d'interaction. Ces deux types de contributions sont positionnés au sein de notre démarche à la Figure 6.1. La section suivante présente les deux espaces conceptuels conçus. Nous résumons ensuite les techniques de visualisation et d'interaction conçues et évaluées.

6.1.1 Espaces conceptuels

Nous avons tout d'abord conçu un espace de conception pour les modalités de navigation sur dispositifs mobiles. Cet espace comprend trois dimensions, correspondant au niveau lexical (dispositif), syntaxique (langage) et sémantique (actions-système déclenchées) d'une modalité d'interaction :

- le dispositif d'interaction utilisé (écran tactile, gyroscope, capteur de pression, etc.) ;
- le type d'action que définit le langage utilisé (discret ou continu) ;
- le mode de navigation (relatif ou absolu).

Cet espace de conception permet de classer/comparer les modalités de navigation existantes et de concevoir de nouvelles modalités (comme la modalité basée sur un capteur de pression et des actions discrètes, décrite dans le chapitre 5).

Nous avons ensuite conçu un modèle conceptuel de navigation multimodale pour la navigation multi-échelle. Ce modèle repose sur une approche multimodale pour naviguer à différents niveaux de zoom d'un espace d'information multi-échelle. Pour cela, nous considérons :

- une phase de navigation rapide utilisant une première modalité ;
- une phase de navigation précise utilisant une seconde modalité ;
- une phase de transition entre les deux premières phases.

En considérant ces trois phases, ce modèle permet de concevoir des techniques de navigation multimodale alliant à la fois rapidité et précision, tout en combinant efficacement les modalités (phase de transition).

Les contributions suivantes instancient ces deux espaces conceptuels pour créer de nouvelles techniques de navigation multimodale sur dispositifs mobiles.

6.1.2 Conception et évaluation de techniques de visualisation et d'interaction

Vue bifocale : Nous avons tout d'abord conçu et évalué une technique de visualisation bifocale sur dispositifs mobiles. Dans le but de répondre au besoin de visualiser à la fois une vue globale de l'espace d'information et une sous-partie détaillée de celui-ci, la technique conçue considère deux niveaux de détail : un pour la vue globale de l'espace d'information, et un pour la sous-partie détaillée. Cette technique utilise un multiplexage spatial pour combiner les deux espaces : à tout instant, les deux espaces sont visibles simultanément. Dans le but de visualiser plus d'information dans la vue globale, nous avons appliqué un effet de perspective. La technique permet la visualisation d'une grande quantité de données sur dispositifs mobiles.

Navigation multimodale : Les techniques de navigation multimodale conçues reposent sur notre modèle conceptuel de navigation. Ainsi, chacune d'elles repose sur la combinaison de deux modalités (une rapide et une précise). Face au vaste espace des possibilités, nous avons réduit notre exploration aux cas où la modalité précise est tactile. Nous avons donc fait varier la modalité rapide et évalué son efficacité pour la phase de navigation rapide et pour la phase de transition vers la modalité tactile. Nous avons pour cela utilisé l'espace de conception des modalités de navigation pour faire varier différentes dimensions :

- *Tilt+Flick* : actions continues d'inclinaison du dispositif pour un mode de navigation relatif.
- *Interaction physique+Flick* : déplacements du dispositif dans l'espace interprétés comme des actions discrètes pour un mode de navigation absolu, et des actions continues pour un mode de navigation relatif.
- *Pression+Flick* : actions discrètes ou continues de pression sur un ou des capteurs de pression pour un mode de navigation relatif.

Nous avons montré par expérimentations que la modalité combinée *Pression+Flick* impliquant des actions continues est aussi rapide que des modalités tactiles bien connues des utilisateurs de dispositifs mobiles. Elle a de plus l'avantage de limiter l'effet d'occultation engendré par les modalités tactiles.

6.2 Perspectives

Face au vaste espace de possibilités, les perspectives sont nombreuses. Nous les divisons en deux parties : les perspectives à court et moyen terme et celles à plus long terme. Nous retrouvons dans nos perspectives la dualité entre travaux conceptuels et pratiques de notre démarche scientifique.

6.2.1 Perspectives à court et moyen terme

Pour continuer les travaux entrepris selon notre approche multimodale, nous identifions trois axes, l'un concernant les techniques de navigation, un autre la méthode d'évaluation, et les deux autres l'extension de notre modèle conceptuel.

6.2.1.1 Capteurs de pression avec des actions discrètes

Nous avons montré que les techniques qui combinaient une modalité basée sur les capteurs de pression et des actions continues avec une modalité tactile définissait une bonne alternative aux modalités purement tactile. Cependant l'utilisation des capteurs de pression avec des actions discrètes n'est pas à exclure. En effet, l'évaluation expérimentale menée a montré que tenter de reproduire les gestes de *Drag/Flick* avec un capteur de pression avait été apprécié par les utilisateurs habitués à effectuer ce type de gestes. La limitation était due au manque de retour quant à la pression appliquée au capteur, et à la difficulté à maîtriser l'effet d'inertie ajouté. Ainsi, nous pensons que deux points pourraient permettre d'obtenir de meilleurs résultats avec cette technique :

- Améliorer le retour haptique du capteur en utilisant un matériau plus déformable pour fournir un retour d'information sur la pression appliquée. Ce retour d'information continue sur la pression appliquée est un élément identifié comme crucial par plusieurs travaux [119, 133]. Nous souhaitons étudier le retour d'information haptique. En effet nous ne pensons pas qu'un retour d'information visuelle est adéquate afin de ne pas surcharger l'écran qui constitue une ressource critique sur dispositifs mobiles.
- Réaliser une étude plus longue permettant la montée en compétence des utilisateurs sur l'effet d'inertie et ainsi permettre d'anticiper l'action suivante. Cette étude permettrait par exemple d'observer une phase de transition minime vers la modalité précise.

6.2.1.2 Evaluation en contexte

Nous regrettons de ne pas avoir eu le temps de mener des expériences avec des vrais jeux de données et des tâches concrètes en nous référant à la taxonomie de tâches de Andrienko [2]. En effet, un espace d'information contenant de vraies données devrait nécessiter une compréhension et une appréhension de celui-ci lors de la navigation pour réaliser des tâches plus concrètes comme celles utilisées dans l'expérience du chapitre 2. Le facteur d'inertie étudié dans le cas des modalités de pression considérées au chapitre 5 pourrait alors montrer son utilité pour mieux contrôller la phase de navigation rapide aux abords d'un point d'interêt. De manière générale, les modalités n'utilisant pas l'interaction tactile limitent l'occultation à l'écran et améliore ainsi la compréhension et l'appréhension de l'espace d'information.

6.2.1.3 Compléter l'espace conceptuel de navigation multimodale : étude de l'inertie

Notre espace de conception pourrait être complété pour prendre en compte l'aspect d'inertie étudié expérimentalement. Nous conceptualisons l'inertie comme une forme de migration de tâche de l'utilisateur vers le système. L'étude conceptuelle de l'inertie nous conduira à étudier une forme de navigation à initiative mixte. L'objet d'étude sera alors les différentes formes d'initiative mixte à la granularité de la navigation et non de la tâche complète : par exemple le système exécute la navigation en mode de navigation relatif et l'utilisateur en modifie la vitesse (comme la technique *Flick&Brake* proposée par Baglioni et al. [7]). Dans ce contexte il convient d'étudier les modalités d'interaction pour initier la tâche de navigation mais aussi celles pour contrôler la navigation ainsi que le partage d'autorité entre le système et l'utilisateur [78]. Par exemple la définition de la vitesse de navigation peut être à tout instant entièrement définie par l'utilisateur ou alors définie en collaboration entre le système et l'utilisateur comme cela est

fait dans la technique *Flick&Brake* où l'utilisateur modifie la vitesse courante définie par le système. Ce partage d'autorité entre l'utilisateur et le système est illustré avec l'extension que nous envisageons à la modalité basée sur le capteur de pression du chapitre 5 : la métaphore de la toupie à pompe et friction (Figure 5.9).

6.2.1.4 Unifier les deux espaces de conception

Un axe d'étude ambitieux consiste à unifier nos deux espaces de conception afin de prendre en compte les caractéristiques des modalités dans notre espace de navigation multimodale. Cette unification des deux espaces permettra alors d'étudier en particulier la phase de transition en s'appuyant sur les caractéristiques de chaque modalité combinée. Ce travail implique une exploration systématique de la combinaison de modalités à l'échelle de leurs caractéristiques. Il permettra d'identifier des règles de combinaison de modalités qui seront alors prescriptives pour la conception.
Dans nos travaux nous avons commencé à étudier ces aspects de façon non systématique en considérant par exemple les changements de tenue du dispositif mobile et les changements de position des doigts lors de la transition d'une modalité à l'autre. Il s'agit ici que des aspects sensori-moteurs humains et d'autres aspects comme le coût cognitif sont à prendre en compte. Par exemple tandis que l'interaction bi-manuelle permet de minimiser les mouvements de l'utilisateur en assignant une main à chaque modalité (voir chapitre 5), qu'en est-il du coût cognitif [80] de changement de modalités ? Nous souhaitons étudier ces aspects au regard des caractéristiques de chaque modalité.

6.2.2 Perspectives à plus long terme

Les deux perspectives à plus long terme étendent le sujet de notre étude et concernent à la fois la navigation (interface en entrée) mais aussi la visualisation (interface en sortie).

6.2.2.1 Incitation au changement de modalités

Comme relaté par Nancel & al. [93] dans le cas du pointage à distance, nous avons observé que les utilisateurs hésitaient à passer à la modalité précise. Ainsi, beaucoup de temps était perdu durant la phase de navigation rapide en tentant d'être précis, temps qui n'était pas compensé durant la phase de navigation précise. Nancel & al. [93] ont visé à minimiser le coût cognitif associé à la décision du changement de mode. Ils ont pour cela forcé les utilisateurs à changer de modalité en découpant l'espace d'information en zones grossières. La navigation rapide consistait alors uniquement à naviguer entre ces zones, la navigation précise à l'intérieur d'une zone pouvant uniquement être effectuée avec la modalité précise. Ils ont ainsi observé un gain de temps durant la phase de navigation rapide, mais pas assez important pour observer un gain de temps sur la tâche globale.

Sur la base des observations dans nos travaux expérimentaux, nous pensons qu'inciter l'utilisateur à changer de modalité peut améliorer les performances des modalités combinées, d'autant plus pour des utilisateurs novices.

Pour cette étude, nous identifions quatre questions de recherche et des pistes pour commencer à les traiter :

- Modéliser les performances de ce changement de modalités : tandis que le modèle de performance de transition de modalités du mode novice au mode expert décrit par Scarr et al. [112] combine deux lois de puissance et souligne une perte de performance lors du passage d'une modalité à l'autre, nous visons la modélisation d'un gain de performance. Cette modélisation de la performance ne sera plus basée sur le temps (montée en compétence comme dans [112]) mais sur la distance restante à parcourir pour atteindre la cible.
- Concevoir un moyen d'inciter les utilisateurs à effectuer ce changement : Les travaux sur le changement de modalités entre le mode novice et le mode expert peuvent définir un point de départ à cette étude comme les travaux de Malacria et al. [85, 86].
- Déterminer le meilleur moment pour l'effectuer : ce calcul peut dans le cadre d'expérimentations contrôlées être immédiat en connaissant la cible mais dans le cas général devra reposer sur les actions effectuées par l'utilisateur sans connaître la cible.
- Faire varier cette aide selon le niveau d'expertise de l'utilisateur.

6.2.2.2 Données dynamiques

Tout au long de nos travaux, nous avons considéré un espace d'information disponible et statique. Une exetnsion de notre travail est alors de considérer un espace de données dynamiques, mis à jour continuellement.

L'aspect dynamique des données à visualiser est fréquent dans de nombreuses applications. Par exemple dans le domaine boursier, il peut être nécessaire de visualiser des historiques de valeurs qui continuent d'être mis à jour. De plus nous retrouvons cet aspect dynamique des données lors de la visualisation d'une liste de courriers électroniques : un nouveau courrier peut être délivré à tout moment, alors que l'utilisateur est en train de visualiser un courrier reçu.

Ce sujet est donc d'importance. Dans le contexte de la société Streamdata.io dans laquelle ont été effectués ces travaux de thèse, ce sujet est particulièrement important et complémentaire à la plateforme logicielle proposée. En effet, la plateforme de la société Streamdata.io consiste justement à envoyer aux dispositifs clients uniquement les nouvelles valeurs de l'espace d'information, lorsqu'il y en a.

Nous définissons deux axes d'étude complémentaires pour prendre en compte la dynamicité de l'espace d'information visualisé : l'observabilité des nouvelles données et l'interaction dans un espace dynamique. Dans ce vaste domaine que constitue la visualisation interactive de données dynamiques [36, 92], une approche de travail est de focaliser sur la visualisation bifocale.

- Observabilité : Ainsi, tout en visualisant un grand espace d'information, il convient de notifier l'utilisateur qu'une ou plusieurs nouvelles valeurs sont disponibles. Dans le cas d'une vue bifocale, si la dernière valeur à visualiser se situe dans la partie détaillée de la technique de visualisation, l'utilisateur la remarquera facilement en utilisant une animation par exemple. Par contre si la donnée à modifier se trouve dans la partie non détaillée de l'espace d'information, il devient alors plus compliqué de rendre observable à l'utilisateur où se situe cette nouvelle donnée et quelle est sa valeur.

- Interaction : La mise à jour dynamique de l'espace de données peut entraîner des difficultés lorsque l'utilisateur est en cours de navigation car la visualisation devient instable. Une solution est par exemple de figer la visualisation le temps de la navigation puis de mettre à jour l'espace d'information. Cette solution est inspirée des solutions d'interaction en réalité augmentée mobile où l'espace est momentanément figé (« freeze » de la vidéo) pour plus de stabilité lors de l'interaction [128].
  Ceci définit un vaste sujet d'étude qui doit, au delà des aspects interactionnels du niveau des actions physiques, prendre en compte la tâche courante de l'utilisateur. Par exemple en nous référant à la taxonomie des tâches décrite par Andrienko [2], si la tâche de l'utilisateur est de type synoptique impliquant d'explorer l'ensemble de l'espace d'information alors les nouvelles données sont d'importance et ne peuvent être masquées.

7 PUBLICATIONS PERSONNELLES

- Vincent, T., Pelurson, S., Regazzoni, V., Kurata, T., & Nigay, L. (2013). Relations spatiales en Réalité Augmentée sur dispositifs mobiles. In *Actes des 9èmes journées francophones Mobilité et Ubiquité (UbiMob 2013)*.
- Pelurson, S., & Nigay, L. (2015, October). Bifocal display on mobile devices: empirical investigation. In *Proceedings of the 27th Conference on l'Interaction Homme-Machine* (p. 20). ACM.
- Bertolo, D., Nigay, L., Pelurson, S., & Vivian, R. (2015, October). Exploratory experimentation of three techniques for rotating a 3D scene by primary school students. In *Proceedings of the 27th Conference on l'Interaction Homme-Machine* (p. 43). ACM.
- Pelurson, S., & Nigay, L. (2015, November). Multimodal Interaction with a Bifocal View on Mobile Devices. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction* (pp. 191-198). ACM.
- Pelurson, S., & Nigay, L. Bimanual Input for Multiscale Navigation with Pressure and Touch Gestures. In *Proceedings of the 2016 ACM on International Conference on Multimodal Interaction*. ACM.

8 BIBLIOGRAPHIE

1. Allen, J. F. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM*, *26*(11), 832-843.
2. Andrienko, N., & Andrienko, G. (2006). *Exploratory analysis of spatial and temporal data: a systematic approach*. Springer Science & Business Media.
3. Appert, C., & Fekete, J. D. (2006, April). OrthoZoom scroller: 1D multi-scale navigation. In *Proceedings of the SIGCHI conference on Human Factors in computing systems* (pp. 21-30). ACM.
4. Apperley, M. D., Tzavaras, I., & Spence, R. (1982, September). A bifocal display technique for data presentation. In *Proceedings of Eurographics* (Vol. 82, pp. 27-43).
5. Appert, C., Chapuis, O., & Pietriga, E. (2010, April). High-precision magnification lenses. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 273-282). ACM.
6. Baglioni, M., Lecolinet, E., & Guiard, Y. (2011, November). JerkTilts: using accelerometers for eight-choice selection on mobile devices. In *Proceedings of the 13th international conference on multimodal interfaces* (pp. 121-128). ACM.
7. Baglioni, M., Malacria, S., Lecolinet, E., & Guiard, Y. (2011, May). Flick-and-brake: finger control over inertial/sustained scroll motion. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems* (pp. 2281-2286). ACM.
8. Bartlett, J. F. (2000). Rock'n'Scroll is here to stay [user interface]. *Computer Graphics and Applications, IEEE*, *20*(3), 40-45.
9. Bartram, L., Ho, A., Dill, J., & Henigman, F. (1995, December). The continuous zoom: A constrained fisheye technique for viewing and navigating large information spaces. In *Proceedings of the 8th annual ACM symposium on User interface and software technology* (pp. 207-215). ACM.
10. Baudisch, P., & Rosenholtz, R. (1993). Halo: a technique for visualizing offscreen location. In *In Proceedings of the Conference on Human Factors in Computing Systems CHI'03*.
11. Baudisch, P., Xie, X., Wang, C., & Ma, W. Y. (2004, October). Collapse-to-zoom: viewing web pages on small screen devices by interactively removing irrelevant content. In *Proceedings of the 17th annual ACM symposium on User interface software and technology* (pp. 91-94). ACM.

12. Bederson, B. B. (2000, November). Fisheye menus. In Proceedings of the 13th annual ACM symposium on User interface software and technology (pp. 217-225). ACM.
13. Bederson, B. B., Clamage, A., Czerwinski, M. P., & Robertson, G. G. (2004). DateLens: A fisheye calendar interface for PDAs. *ACM Transactions on Computer-Human Interaction (TOCHI)*, *11*(1), 90-119.
14. Benko, H., Wilson, A. D., & Baudisch, P. (2006, April). Precise selection techniques for multi-touch screens. In *Proceedings of the SIGCHI conference on Human Factors in computing systems* (pp. 1263-1272). ACM.
15. Björk, S., Holmquist, L. E., Redström, J., Bretan, I., Danielsson, R., Karlgren, J., & Franzén, K. (1999, November). WEST: a Web browser for small terminals. In *Proceedings of the 12th annual ACM symposium on User interface software and technology* (pp. 187-196). ACM.
16. Brewster, S. A., & Hughes, M. (2009, September). Pressure-based text entry for mobile devices. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services* (p. 9). ACM.
17. Brooke, J. (1996). SUS: A 'Quick and Dirty'Usability Scale, Usability Evaluation in Industry, Jordan, PW, Thomas, B., Weerdmeester, BA and McClelland, AL.
18. Burigat, S., & Chittaro, L. (2005, March). Location-aware visualization of VRML models in GPS-based mobile guides. In *Proceedings of the tenth international conference on 3D Web technology* (pp. 57-64). ACM.
19. Burigat, S., & Chittaro, L. (2013). On the effectiveness of Overview+ Detail visualization on mobile devices. *Personal and ubiquitous computing*, *17*(2), 371-385.
20. Burigat, S., Chittaro, L., & Parlato, E. (2008, September). Map, diagram, and web page navigation on mobile devices: the effectiveness of zoomable user interfaces with overviews. In *Proceedings of the 10th international conference on Human computer interaction with mobile devices and services* (pp. 147-156). ACM.
21. Büring, T., Gerken, J., & Reiterer, H. (2006, May). Usability of overview-supported zooming on small screens with regard to individual differences in spatial ability. In *Proceedings of the working conference on Advanced visual interfaces* (pp. 233-240). ACM.
22. Butler, A., Izadi, S., & Hodges, S. (2008, October). SideSight: multi-touch interaction around small devices. In *Proceedings of the 21st annual ACM symposium on User interface software and technology* (pp. 201-204). ACM.
23. Buxton, W. (1986, September). Chunking and phrasing and the design of human-computer dialogues. In *IFIP Congress* (pp. 475-480).
24. Casiez, G., Roussel, N., & Vogel, D. (2012, May). 1€ filter: a simple speed-based low-pass filter for noisy input in interactive systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 2527-2530). ACM.
25. Casiez, G., Vogel, D., Pan, Q., & Chaillou, C. (2007, October). 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* (pp. 129-138). ACM.
26. Chen, C. (2006). *Information visualization: Beyond the horizon*. Springer Science & Business Media.
27. Chen, K. Y., Lyons, K., White, S., & Patel, S. (2013, October). uTrack: 3D input using two magnetic sensors. In *Proceedings of the 26th annual ACM symposium on User interface software and technology* (pp. 237-244). ACM.

28. Chittaro, L. (2006, May). Visualization of patient data at different temporal granularities on mobile devices. In *Proceedings of the working conference on Advanced visual interfaces* (pp. 484-487). ACM.
29. Chittaro, L. (2006). Visualizing information on mobile devices. *Computer*, *39*(3), 40-45.
30. Cho, S. J., Choi, C., Sung, Y., Lee, K., Kim, Y. B., & Murray-Smith, R. (2007, April). Dynamics of tilt-based browsing on mobile devices. In *CHI'07 Extended Abstracts on Human Factors in Computing Systems* (pp. 1947-1952). ACM.
31. Clarkson, E. C., Patel, S. N., Pierce, J. S., & Abowd, G. D. (2006). Exploring continuous pressure input for mobile phones. GVU Techreport; GIT-GVU-06-20, Georgia Inst. of Tech.
32. Coutaz, J., Nigay, L., Salber, D., Blandford, A., May, J., & Young, R. M. (1995). Four easy pieces for assessing the usability of multimodal interaction: the CARE properties. In *Human—Computer Interaction* (pp. 115-120). Springer US.
33. Dachselt, R., & Frisch, M. (2007, December). Mambo: a facet-based zoomable music browser. In *Proceedings of the 6th international conference on Mobile and ubiquitous multimedia* (pp. 110-117). ACM.
34. Delamare, W., Coutrix, C., & Nigay, L. (2013, August). Mobile pointing task in the physical world: balancing focus and performance while disambiguating. In *Proceedings of the 15th international conference on Human-computer interaction with mobile devices and services* (pp. 89-98). ACM.
35. Engdahl, B., Köksal, M., & Marsden, G. (2005, April). Using treemaps to visualize threaded discussion forums on PDAs. In *CHI'05 extended abstracts on Human factors in computing systems* (pp. 1355-1358). ACM.
36. Erbacher, R. F., Walker, K. L., & Frincke, D. A. (2002). Intrusion and misuse detection in large-scale systems. *Computer Graphics and Applications, IEEE*, *22*(1), 38-47.
37. Eslambolchilar, P., & Murray-Smith, R. (2004). Tilt-based automatic zooming and scaling in mobile devices–a state-space implementation. In *Mobile Human-Computer Interaction-MobileHCI 2004* (pp. 120-131). Springer Berlin Heidelberg.
38. Ferreira, L. A. A. (2013). Hyperbolic tree visualization on mobile devices.
39. Fitchett, S., & Cockburn, A. (2009, November). Evaluating reading and analysis tasks on mobile devices: a case study of tilt and flick scrolling. In *Proceedings of the 21st Annual Conference of the Australian Computer-Human Interaction Special Interest Group: Design: Open 24/7* (pp. 225-232). ACM.
40. Fitchett, S., & Cockburn, A. (2010, September). MultiScroll: using multitouch input to disambiguate relative and absolute mobile scroll modes. In *Proceedings of the 24th BCS Interaction Specialist Group Conference* (pp. 393-402). British Computer Society.
41. Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology*, *47*(6), 381.
42. Fitzmaurice, G. W., Zhai, S., & Chignell, M. H. (1993). Virtual reality for palmtop computers. *ACM Transactions on Information Systems (TOIS)*, *11*(3), 197-218.
43. Foley, J. D., Wallace, V. L., & Chan, P. (1984). The human factors of computer graphics interaction techniques. *Computer Graphics and Applications, IEEE*, *4*(11), 13-48.
44. Francone, J., Bailly, G., Lecolinet, E., Mandran, N., & Nigay, L. (2010, May). Wavelet menus on handheld devices: stacking metaphor for novice mode and

eyes-free selection for expert mode. In *Proceedings of the International Conference on Advanced Visual Interfaces* (pp. 173-180). ACM.
45. Furnas, G. W. (1986). *Generalized fisheye views* (Vol. 17, No. 4, pp. 16-23). ACM.
46. Ganhör, R. (2014, April). Athmos: Focus+ Context for Browsing in Mobile Thumbnail Collections. In *Proceedings of International Conference on Multimedia Retrieval* (p. 65). ACM.
47. Grégory, C., Noirhomme-Fraiture, M., & Luca, C. (2004, August). Visualisation d'informations boursières sur téléphones mobiles. In *Proceedings of the 16th conference on Association Francophone d'Interaction Homme-Machine* (pp. 215-218). ACM.
48. Gonçalves, T., Afonso, A. P., Carmo, M. B., & Pombinho, P. (2012, September). Overview vs detail on mobile devices: a struggle for screen space. In *Proceedings of the 26th Annual BCS Interaction Specialist Group Conference on People and Computers* (pp. 233-238). British Computer Society.
49. Guiard, Y. (1987). Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. *Journal of motor behavior*, *19*(4), 486-517.
50. Gustafson, S., Baudisch, P., Gutwin, C., & Irani, P. (2008, April). Wedge: clutter-free visualization of off-screen locations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 787-796). ACM.
51. Gutwin, C. (2002, April). Improving focus targeting in interactive fisheye views. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 267-274). ACM.
52. Hao, J., & Zhang, K. (2007, June). A mobile interface for hierarchical information visualization and navigation. In *Consumer Electronics, 2007. ISCE 2007. IEEE International Symposium on* (pp. 1-7). IEEE.
53. Hao, J., Gabrysch, C. A., Zhao, C., Zhu, Q., & Zhang, K. (2010). Visualizing and navigating hierarchical information on mobile user interfaces. *International Journal of Advanced Intelligence*, *2*(1), 81-103.
54. Harrison, C., & Hudson, S. E. (2009, October). 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* (pp. 121-124). ACM.
55. Heo, S., & Lee, G. (2011, October). Force gestures: augmenting touch screen gestures with normal and tangential forces. In *Proceedings of the 24th annual ACM symposium on User interface software and technology* (pp. 621-626). ACM.
56. Hornbæk, K., & Frøkjær, E. (2001, March). Reading of electronic documents: the usability of linear, fisheye, and overview+ detail interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 293-300). ACM.
57. Huot, S., & Lecolinet, E. (2006, October). SpiraList: a compact visualization technique for one-handed interaction with large lists on mobile devices. In *Proceedings of the 4th Nordic conference on Human-computer interaction: changing roles* (pp. 445-448). ACM.
58. Hwang, S., Lee, K., & Lim, C. VirtualThumb: One-Handed Multi-Touch Emulation Technique on Small Devices.
59. Igarashi, T., & Hinckley, K. (2000, November). Speed-dependent automatic zooming for browsing large documents. In *Proceedings of the 13th annual ACM symposium on User interface software and technology* (pp. 139-148). ACM.
60. Jones, B., Sodhi, R., Forsyth, D., Bailey, B., & Maciocci, G. (2012, September). Around device interaction for multiscale navigation. In *Proceedings of the 14th*

international conference on Human-computer interaction with mobile devices and services (pp. 83-92). ACM.

61. Jones, M., Marsden, G., Mohd-Nasir, N., Boone, K., & Buchanan, G. (1999). Improving Web interaction on small displays. *Computer Networks*, *31*(11), 1129-1137.
62. Jones, S., Jones, M., Marsden, G., Patel, D., & Cockburn, A. (2005). An evaluation of integrated zooming and scrolling on small screens. *International Journal of Human-Computer Studies*, *63*(3), 271-303.
63. Jul, S., & Furnas, G. W. (1998, November). Critical zones in desert fog: aids to multiscale navigation. In *Proceedings of the 11th annual ACM symposium on User interface software and technology* (pp. 97-106). ACM.
64. Juras, D., Nigay, L., Ortega, M., & Serrano, M. (2008, October). Multimodal slideshow: demonstration of the openinterface interaction development environment. In *Proceedings of the 10th international conference on Multimodal interfaces* (pp. 193-194). ACM.
65. Karstens, B., Kreuseler, M., & Schumann, H. (2003, June). Visualization of complex structures on mobile handhelds. In *Proc. International Workshop on Mobile Computing*.
66. Karlson, A. K., & Bederson, B. B. (2008, April). One-handed touchscreen input for legacy applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 1399-1408). ACM.
67. Karlson, A. K., Bederson, B. B., & SanGiovanni, J. (2005, April). AppLens and launchTile: two designs for one-handed thumb use on small devices. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 201-210). ACM.
68. Karlson, A. K., Robertson, G. G., Robbins, D. C., Czerwinski, M. P., & Smith, G. R. (2006, April). FaThumb: a facet-based interface for mobile search. In *Proceedings of the SIGCHI conference on Human Factors in computing systems* (pp. 711-720). ACM.
69. Ketabdar, H., Yüksel, K. A., & Roshandel, M. (2010, February). MagiTact: interaction with mobile devices based on compass (magnetic) sensor. In *Proceedings of the 15th international conference on Intelligent user interfaces* (pp. 413-414). ACM.
70. Ketabdar, H., Roshandel, M., & Yüksel, K. A. (2010, September). Towards using embedded magnetic field sensor for around mobile device 3D interaction. In *Proceedings of the 12th international conference on Human computer interaction with mobile devices and services* (pp. 153-156). ACM.
71. Khella, A., & Bederson, B. B. (2004, October). Pocket PhotoMesa: a Zoomable image browser for PDAs. In *Proceedings of the 3rd international conference on Mobile and ubiquitous multimedia* (pp. 19-24). ACM.
72. Kratz, S., Brodien, I., & Rohs, M. (2010, September). Semi-automatic zooming for mobile map navigation. In *Proceedings of the 12th international conference on Human computer interaction with mobile devices and services* (pp. 63-72). ACM.
73. Kratz, S., & Rohs, M. (2009, September). HoverFlow: expanding the design space of around-device interaction. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services* (p. 4). ACM.
74. Laarni, J. (2002, October). Searching for optimal methods of presenting dynamic text on different types of screens. In *Proceedings of the second Nordic conference on Human-computer interaction* (pp. 219-222). ACM.

75. Lam, H., & Baudisch, P. (2005, April). Summary thumbnails: readable overviews for small screen web browsers. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 681-690). ACM.
76. Lamping, J., & Rao, R. (1996). The hyperbolic browser: A focus+ context technique for visualizing large hierarchies. *Journal of Visual Languages & Computing*, *7*(1), 33-55.
77. Lank, E., & Phan, S. (2004, April). Focus+ Context sketching on a pocket PC. In *CHI'04 Extended Abstracts on Human Factors in Computing Systems* (pp. 1275-1278). ACM.
78. Leal, A., Bouchet, J., Langlois, G., & Jourde, F. (2009, October). Conception de l'interaction homme-machine et partage d'autorité: application aux systèmes de drones. In *Proceedings of the 21st International Conference on Association Francophone d'Interaction Homme-Machine* (pp. 283-290). ACM.
79. Lecolinet, E., & Nguyen, D. (2006, April). Représentation focus+ contexte de listes hiérarchiques zoomables. In *Proceedings of the 18th International Conferenceof the Association Francophone d'Interaction Homme-Machine* (pp. 195-198). ACM.
80. Leganchuk, A., Zhai, S., & Buxton, W. (1998). Manual and cognitive benefits of two-handed input: an experimental study. *ACM Transactions on Computer-Human Interaction (TOCHI)*, *5*(4), 326-359.
81. Leung, Y. K., & Apperley, M. D. (1994). A review and taxonomy of distortion-oriented presentation techniques. *ACM Transactions on Computer-Human Interaction (TOCHI)*, *1*(2), 126-160.
82. Lipman, R. R. (2004). Mobile 3D visualization for steel structures. *Automation in Construction*, *13*(1), 119-125.
83. MacKenzie, I. S., & Teather, R. J. (2012, October). FittsTilt: the application of Fitts' law to tilt-based interaction. In *Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making Sense Through Design* (pp. 568-577). ACM.
84. Mackinlay, J. D., Robertson, G. G., & Card, S. K. (1991, April). The perspective wall: Detail and context smoothly integrated. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 173-176). ACM.
85. Malacria, S., Bailly, G., Harrison, J., Cockburn, A., & Gutwin, C. (2013, April). Promoting hotkey use through rehearsal with exposehk. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 573-582). ACM.
86. Malacria, S., Scarr, J., Cockburn, A., Gutwin, C., & Grossman, T. (2013, October). Skillometers: Reflective widgets that motivate and help users to improve performance. In *Proceedings of the 26th annual ACM symposium on User interface software and technology* (pp. 321-330). ACM.
87. McCallum, D. C., Mak, E., Irani, P., & Subramanian, S. (2009, April). PressureText: pressure input for mobile phone text entry. In *CHI'09 Extended Abstracts on Human Factors in Computing Systems* (pp. 4519-4524). ACM.
88. McLachlan, R., & Brewster, S. A. (2013). Novel Modalities for Bimanual Scrolling on Tablet Devices. In *Human-Computer Interaction–INTERACT 2013* (pp. 229-246). Springer Berlin Heidelberg.
89. McLachlan, R., Boland, D., & Brewster, S. (2014, April). Transient and transitional states: pressure as an auxiliary input modality for bimanual interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 401-410). ACM.
90. Miyaki, T., & Rekimoto, J. (2009, September). 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 (p. 11). ACM.

91. Mizobuchi, S., Terasaki, S., Keski-Jaskari, T., Nousiainen, J., Ryynanen, M., & Silfverberg, M. (2005, April). Making an impression: force-controlled pen input for handheld devices. In *CHI'05 extended abstracts on Human factors in computing systems* (pp. 1661-1664). ACM.
92. Moere, A. V. (2004, October). Time-varying data visualization using information flocking boids. In *Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on* (pp. 97-104). IEEE.
93. Nancel, M., Chapuis, O., Pietriga, E., Yang, X. D., Irani, P. P., & Beaudouin-Lafon, M. (2013, April). High-precision pointing on large wall displays using small handheld devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 831-840). ACM.
94. Nancel, M., Vogel, D., & Lank, E. (2015, April). Clutching is not (necessarily) the enemy. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (pp. 4199-4202). ACM.
95. Nigay, J. C. L., Jambon, F., & Coutaz, J. (1995). Formal specification of multimodality. In *CHI'95 Workshop*.
96. Oakley, I., & O'Modhrain, S. (2005, March). Tilt to scroll: evaluating a motion based vibrotactile mobile interface. In Eurohaptics Conference, 2005 and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2005. World Haptics 2005. First Joint (pp. 40-49). IEEE.
97. Pahud, M., Hinckley, K., Iqbal, S., Sellen, A., & Buxton, B. (2013, August). Toward compound navigation tasks on mobiles via spatial manipulation. In *Proceedings of the 15th international conference on Human-computer interaction with mobile devices and services* (pp. 113-122). ACM.
98. Pietriga, E., Appert, C., & Beaudouin-Lafon, M. (2007, April). Pointing and beyond: an operationalization and preliminary evaluation of multi-scale searching. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 1215-1224). ACM.
99. Plaisant, C., Carr, D., & Shneiderman, B. (1995). Image-browser taxonomy and guidelines for designers. *Software, IEEE*, *12*(2), 21-32.
100. Pook, S. (2001). *Interaction et contexte dans les interfaces zoomables* (Doctoral dissertation, Télécom ParisTech).
101. Quinn, P., Malacria, S., & Cockburn, A. (2013, October). Touch scrolling transfer functions. In *Proceedings of the 26th annual ACM symposium on User interface software and technology* (pp. 61-70). ACM.
102. Rahman, M., Gustafson, S., Irani, P., & Subramanian, S. (2009, April). Tilt techniques: investigating the dexterity of wrist-based input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 1943-1952). ACM.
103. Rekimoto, J. (1996, November). Tilting operations for small screen interfaces. In *Proceedings of the 9th annual ACM symposium on User interface software and technology* (pp. 167-168). ACM.
104. Robertson, G. G., & Mackinlay, J. D. (1993, December). The document lens. In *Proceedings of the 6th annual ACM symposium on User interface software and technology* (pp. 101-108). ACM.
105. Robbins, D. C., Cutrell, E., Sarin, R., & Horvitz, E. (2004, May). ZoneZoom: map navigation for smartphones with recursive view segmentation. In *Proceedings of the working conference on Advanced visual interfaces* (pp. 231-234). ACM.

106. Roussel, N., Casiez, G., Aceituno, J., & Vogel, D. (2012, October). Giving a hand to the eyes: leveraging input accuracy for subpixel interaction. In *Proceedings of the 25th annual ACM symposium on User interface software and technology* (pp. 351-358). ACM.
107. Roto, V., Popescu, A., Koivisto, A., & Vartiainen, E. (2006, April). Minimap: a web page visualization method for mobile phones. In *Proceedings of the SIGCHI conference on Human Factors in computing systems* (pp. 35-44). ACM.
108. Roudaut, A. (2010). Conception et Evaluation de Technique d'Interaction pour Dispositifs Mobiles (Doctoral dissertation, Télécom ParisTech).
109. Roudaut, A., Huot, S., & Lecolinet, E. (2008, May). TapTap and MagStick: improving one-handed target acquisition on small touch-screens. In *Proceedings of the working conference on Advanced visual interfaces* (pp. 146-153). ACM.
110. Ruiz, J., & Li, Y. (2011, May). DoubleFlip: a motion gesture delimiter for mobile interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 2717-2720). ACM.
111. Sad, H. H., & Poirier, F. (2009, February). Evaluation and modeling of user performance for pointing and scrolling tasks on handheld devices using tilt sensor. In *Advances in Computer-Human Interactions, 2009. ACHI'09. Second International Conferences on* (pp. 295-300). IEEE.
112. Scarr, J., Cockburn, A., Gutwin, C., & Quinn, P. (2011, May). Dips and ceilings: understanding and supporting transitions to expertise in user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 2741-2750). ACM.
113. Schall, G., Mendez, E., Kruijff, E., Veas, E., Junghanns, S., Reitinger, B., & Schmalstieg, D. (2009). Handheld augmented reality for underground infrastructure visualization. *Personal and ubiquitous computing*, *13*(4), 281-291.
114. Scoditti, A., Blanch, R., & Coutaz, J. (2011, February). A novel taxonomy for gestural interaction techniques based on accelerometers. In *Proceedings of the 16th international conference on Intelligent user interfaces* (pp. 63-72). ACM.
115. Serrano, M., Juras, D., & Nigay, L. (2008, October). A three-dimensional characterization space of software components for rapidly developing multimodal interfaces. In *Proceedings of the 10th international conference on Multimodal interfaces* (pp. 149-156). ACM.
116. Serrano, M., & Nigay, L. (2009, November). Temporal aspects of CARE-based multimodal fusion: from a fusion mechanism to composition components and WoZ components. In *Proceedings of the 2009 international conference on Multimodal interfaces* (pp. 177-184). ACM.
117. Shneiderman, B. (1996, September). The eyes have it: A task by data type taxonomy for information visualizations. In *Visual Languages, 1996. Proceedings., IEEE Symposium on* (pp. 336-343). IEEE.
118. Siek, K. A., Rogers, Y., & Connelly, K. H. (2005). Fat finger worries: how older and younger users physically interact with PDAs. In *Human-Computer Interaction-INTERACT 2005* (pp. 267-280). Springer Berlin Heidelberg.
119. Spelmezan, D., Appert, C., Chapuis, O., & Pietriga, E. (2013, August). Side pressure for bidirectional navigation on small devices. In *Proceedings of the 15th international conference on Human-computer interaction with mobile devices and services* (pp. 11-20). ACM.
120. Spence, R. (2001). *Information visualization* (Vol. 1). New York: Addison-Wesley.

121. Spence, R., & Apperley, M. (1982). Data base navigation: an office environment for the professional. *Behaviour & Information Technology*, *1*(1), 43-54.
122. Spindler, M., Schuessler, M., Martsch, M., & Dachselt, R. (2014, April). Pinch-drag-flick vs. spatial input: rethinking zoom & pan on mobile displays. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems* (pp. 1113-1122). ACM.
123. Stewart, C., Rohs, M., Kratz, S., & Essl, G. (2010, April). Characteristics of pressure-based input for mobile devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 801-810). ACM.
124. Teather, R. J., & MacKenzie, I. S. (2014, May). Position vs. velocity control for tilt-based interaction. In *Proceedings of the 2014 Graphics Interface Conference* (pp. 51-58). Canadian Information Processing Society.
125. Thomason, J., & Wang, J. (2012, September). Exploring multi-dimensional data on mobile devices with single hand motion and orientation gestures. In *Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services companion* (pp. 173-176). ACM.
126. Tsang, M., Fitzmaurice, G. W., Kurtenbach, G., Khan, A., & Buxton, B. (2002, October). Boom chameleon: simultaneous capture of 3D viewpoint, voice and gesture annotations on a spatially-aware display. In *Proceedings of the 15th annual ACM symposium on User interface software and technology* (pp. 111-120). ACM.
127. Vernier, F., & Nigay, L. (2000). A framework for the combination and characterization of output modalities. In Interactive Systems Design, Specification, and Verification (pp. 35-50). Springer Berlin Heidelberg.
128. Vincent, T. (2014). Interaction en réalité augmentée sur dispositif mobile: relations spatiales.
129. Vogel, D., & Baudisch, P. (2007, April). Shift: a technique for operating pen-based interfaces using touch. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 657-666). ACM.
130. Wang, Y. S., & Chi, M. T. (2011). Focus+ context metro maps. *Visualization and Computer Graphics, IEEE Transactions on*, *17*(12), 2528-2535.
131. Wilson, G., Brewster, S. A., & Halvey, M. (2011, May). The effects of walking and control method on pressure-based interaction. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems* (pp. 2275-2280). ACM.
132. Wilson, G., Brewster, S. A., Halvey, M., Crossan, A., & Stewart, C. (2011, August). The effects of walking, feedback and control method on pressure-based interaction. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services* (pp. 147-156). ACM.
133. Wilson, G., Stewart, C., & Brewster, S. A. (2010, September). Pressure-based menu selection for mobile devices. In *Proceedings of the 12th international conference on Human computer interaction with mobile devices and services* (pp. 181-190). ACM.
134. Yee, K. P. (2003, April). Peephole displays: pen interaction on spatially aware handheld computers. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 1-8). ACM.
135. Yun, S., Chen, Y. C., & Qiu, L. (2015, May). Turning a mobile device into a mouse in the air. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services* (pp. 15-29). ACM.

9 Annexes

Ces annexes visent à fournir des informations complémentaires au manuscrit de thèse dans le but de permettre à un lecteur de reproduire les expériences menées et/ou de re-développer les modalités conçues.

9.1 Annexe 1 – Tâches réalisées durant l'expérience de visualisation

Cette annexe présente la liste des tâches utilisée pour l'expérience présenté dans le chapitre 2 et inspiré de la taxonomie de tâches de Andrienko [2]. Les espaces d'information utilisés simulaient des données réalistes nous permettant de formuler des tâches concrêtes.

1. Quelle est la valeur de l'indice le 28/08/14 ?
2. Quelles sont les valeurs de l'indice aux dates suivantes ?
    a. 20/08/14
    b. 17/09/09
    c. 16/10/04
3. A quelle date l'indice a-t-il atteint sa valeur maximale ?
4. La valeur minimale a-t-elle été atteinte avant ou après la valeur maximale ?
5. Durant le mois de mars 2008, quelle a été la tendance de l'indice ?
6. A quelles dates l'indice a-t-il dépassé sa valeur à la date 03/04/2008 ?
7. Quelles sont les valeurs de l'indice aux dates suivantes ?
    a. 08/08/14
    b. 09/08/12
    c. 12/08/10
    d. 21/08/08
    e. 10/08/06
    f. 28/10/04
8. Laquelle des deux dates suivantes a la valeur la plus élevée ?
    a. 25/07/14
    b. 04/08/14
9. Quel est le plus grand intervalle durant lequel l'indice n'a pas décru ?

10. Le mois d'avril 2010 a-t-il subi une hausse ?
11. Quelles sont les valeurs de l'indice aux dates suivantes ?
    a. 18/08/14
    b. 13/04/14
    c. 08/08/13
    d. 12/04/12
    e. 20/08/09
    f. 21/10/04
12. Laquelle des deux dates suivantes a la valeur la plus élevée ?
    a. 13/01/11
    b. 27/08/14
13. Quelles sont les valeurs de l'indice aux dates suivantes ?
    a. 04/08/14
    b. 12/09/13
    c. 18/10/12
    d. 10/11/11
    e. 23/12/10
    f. 21/01/10
    g. 12/02/09
    h. 27/03/08
    i. 05/04/07
    j. 25/05/06
    k. 02/06/05
    l. 04/11/04
14. Quelle année a subi la plus grosse différence de valeur ?
15. Quelles sont les valeurs de l'indice aux dates suivantes ?
    a. 18/08/14
    b. 19/12/13
    c. 09/08/12
    d. 05/08/10
    e. 20/12/07
    f. 18/11/04
16. Donner deux mois ayant des tendances opposées.
17. Laquelle des deux dates suivantes a la valeur la plus faible ?
    a. 21/08/14
    b. 06/05/14

9.2 Annexe 2 – Questionnaire SUS

Cette annexe contient la traduction française du questionnaire System Usability Scale (SUS) [17] utilisé durant les différentes évaluations présentées dans ce manuscrit. Ce questionnaire comprend 10 affirmations nécessitant une évaluation sur une échelle de Likert de 5. La note de 1 signifie « Pas du tout d'accord » avec l'affirmation, la note de 5 signifie « Tout à fait d'accord » avec l'affirmation.

18. Je pense que j'aimerais utiliser ce type de système fréquemment.
19. Je trouve que ce système est inutilement complexe.
20. J'ai trouvé que le système était facile à utiliser.
21. Je pense que j'aurais besoin de l'aide d'une personne expérimentée pour utiliser ce système.
22. J'ai trouvé que les différentes fonctions du système étaient bien intégrées.

23. J'ai trouvé qu'il y avait trop d'incohérence dans ce système.
24. J'imagine que plusieurs personnes apprendront vite à utiliser ce système.
25. J'ai trouvé que le système était pesant, lourd, lors de son utilisation.
26. Je me suis senti en confiance lors de l'utilisation du système.
27. J'ai eu besoin d'apprendre beaucoup de choses avant d'utiliser le système.

Le calcul du score pour ce questionnaire s'effectue de la manière suivante :

- Pour les affirmations 1, 3, 5, 7 et 9, le score est le résultat sur l'échelle moins 1. (tout à fait d'accord : 5-1 = 4)
- Pour les affirmations 2, 4, 6, 8 et 10, le score est 5 moins le résultat sur l'échelle . (tout à fait d'accord : 5-5 = 0)
- Faire le total des scores et multiplier par 2.5 pour obtenir le score SUS qui varie de 0 à 100.

9.3 Annexe 3 – Linéarisation des capteurs de pression avec un amplificateur opérationnel

Les capteurs de pression que nous avons utilisé dans nos prototypes (Interlink Elektroniks FSR 400) ne retournent pas des valeurs linéarisées, mais logarithmiques. Ceci rend l'accès aux pressions basses et hautes compliqué. Comme [135, 136], nous avons utilisé un amplificateur opérationnel pour linéariser ces données. Nous fournissons ci-dessous le schéma électrique permettant de reproduire ce montage et tiré de la documentatin du capteur[10].

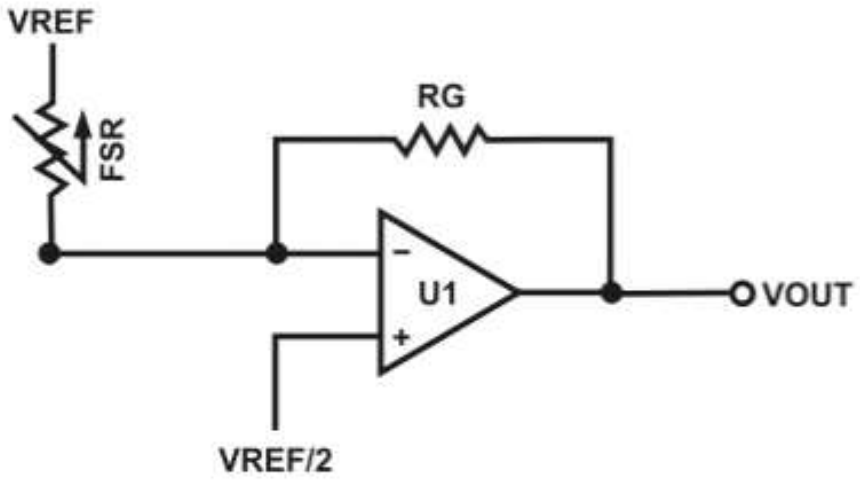


Figure 9.1 – Schéma du convertisseur courant/tension utilisé pour linéariser les valeurs retournées par les capteurs de pression.

[10] http://www.interlinkelectronics.com/integration_guides/FSR400Series_IG.zip