



**HAL**  
open science

# Contrôle orbital pour le tracé de trajectoires 3D à l'aide des mouvements de la tête

Thibaut Jacob

► **To cite this version:**

Thibaut Jacob. Contrôle orbital pour le tracé de trajectoires 3D à l'aide des mouvements de la tête. Interface homme-machine [cs.HC]. Télécom ParisTech, 2017. Français. NNT : 2017ENST0044 . tel-03419931

**HAL Id: tel-03419931**

**<https://pastel.hal.science/tel-03419931>**

Submitted on 8 Nov 2021

**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.



EDITE - ED 130

## Doctorat ParisTech

# THÈSE

pour obtenir le grade de docteur délivré par

**TELECOM ParisTech**

**Spécialité « Informatique et Réseaux »**

*présentée et soutenue publiquement par*

**Thibaut JACOB**

le 22 septembre 2017

## **Contrôle orbital pour le tracé de trajectoires 3D à l'aide des mouvements de la tête**

Directeur de thèse : **Éric LECOLINET**  
Co-encadrement de la thèse : **Gilles BAILLY**

### Jury

**Mme Nadine ROUILLON COUTURE**, Professeur HDR, LaBRI, ESTIA  
**M. Laurent GRISONI**, Professeur HDR, CRISTAL, Université Lille 1  
**M. Thierry DUVAL**, Professeur HDR, LabSTICC, IMT Atlantique  
**M. Mehdi AMMI**, HDR, LIMSI, Université Paris-Saclay

Rapporteur  
Rapporteur  
Examineur  
Examineur

**TELECOM ParisTech**

école de l'Institut Mines-Télécom - membre de ParisTech



Je suis extrêmement reconnaissant aux membres de mon jury de thèse, Mme Nadine Couture, M. Laurent Grisoni, M. Thierry Duval et M. Medhi Ammi d'avoir accepté de relire mon manuscrit de thèse et pour leurs remarques très constructives sur celui-ci.

Un grand merci également à mes deux encadrants, Éric Lecolinet et Gilles Bailly pour m'avoir guidé tout au long de cette thèse, pour leur indéfectible soutien au milieu des deadlines, pour leurs conseils avisés en tout temps et pour tout l'enseignement que j'en ai tiré.

Je remercie aussi tous les membres du consortium EDISON3D et plus particulièrement nos interlocuteurs principaux pour nous avoir permis de réaliser des études de terrain : Frédéric Changenet, Hervé Déjardin, Étienne Corteel et Raphaël Foulon.

Merci à tous les membres présents et passés de l'équipe VIA ainsi que certains doctorants des équipes ILDA et Ex-situ qui ont su me conseiller aux différents stades de cette thèse : James Eagan, Yves Guiard, Simon Perrault, Quentin Roy, Hind Gacem, Aurélie Cohé, Bastien Liutkus, Nadia Boukhelifa, Can Liu, Marc-Emmanuel Perrin, Arnaud Prouzeau, Hugo Romat, Jessalyn Alvina, Carla Griggio, Germán Leiva, Evanthia Dimara, Emmanouil Giannidakis, Marc Teyssier, Bruno Fruchard, Emeline Brulé, Wanyu Liu et Rafael Morales.

Je remercie tout particulièrement Marc-Emmanuel Perrin d'avoir suggéré en premier lieu l'utilisation d'une centrale inertielle comme dispositif d'entrée pour mesurer les rotations de la tête, pour l'implémentation du logiciel embarqué sur Arduino et l'interface logicielle permettant la récupération des données de rotation sur l'ordinateur. Un grand merci également pour son enthousiasme, ses nombreuses contributions sur nos projets communs (*AMEL*, *Sunset PI*, *ART2Websocket*, etc.) et ses qualités de bricoleur hors-pair.

Je remercie également Marc Teyssier pour son expertise dans l'utilisation du moteur de jeu *Unity* et du logiciel de modélisation *Blender*, ses conseils en tant que designer web et pour l'implémentation de ma technique de contrôle du point de vue dans plusieurs prototypes *Unity*.

Je voudrais également remercier Anastasia Bezerianos et Theophanis Tsandilas qui m'ont donné goût à la recherche en me permettant d'effectuer un stage de recherche au sein du *Laboratoire de Recherche en Informatique* et en soutenant ma candidature pour l'obtention d'une bourse de thèse.

Je remercie également Michel Beaudouin-Lafon et Wendy Mackay pour leurs enseignements très enrichissants au sein du *Master 2 Interaction* qui m'ont motivé à réaliser cette thèse.

Merci aussi à Olivier Chapuis et Tamy Boubekeur d'avoir accepté de présider à ma soutenance de mi-parcours et de m'avoir apporté leurs conseils éclairés.

Enfin, ce travail n'aurait pas été possible sans le soutien de ma famille au complet tant russe que française et je tiens à remercier tout particulièrement Daria de m'avoir encouragé et d'avoir été à mes côtés tout au long du déroulement de celle-ci.



---

# TABLE DES MATIÈRES

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Contributions de la thèse . . . . .	3
1.2	Structure/organisation . . . . .	3
<b>2</b>	<b>Espace de conception pour le tracé et l'édition de courbes 3D</b>	<b>5</b>
2.1	Étude de terrain . . . . .	5
2.1.1	Méthode . . . . .	5
2.1.2	Interviews . . . . .	5
2.1.3	Conclusion . . . . .	8
2.2	Espace de conception : définition et couverture . . . . .	9
2.2.1	Définition et couverture . . . . .	9
2.2.2	Espace de conception . . . . .	9
2.3	Dispositifs d'entrée . . . . .	10
2.3.1	Souris + clavier . . . . .	10
2.3.2	Souris augmentées . . . . .	10
2.3.3	Surfaces tactiles . . . . .	12
2.3.4	Interaction en l'air . . . . .	13
2.3.5	Interaction tangible . . . . .	14
2.4	Dispositifs de sortie et perception . . . . .	16
2.4.1	Écrans traditionnels . . . . .	16
2.4.2	Écrans stéréoscopiques . . . . .	16
2.4.3	Retour haptique . . . . .	17
2.5	Langage d'interaction . . . . .	17
2.5.1	Tracé de courbes 3D . . . . .	17
2.5.2	Édition de courbes 3D . . . . .	20
2.6	Conclusion . . . . .	22
<b>3</b>	<b>Contrôle du point de vue</b>	<b>25</b>
3.1	Indices visuels . . . . .	26
3.1.1	Indices monoculaires . . . . .	26
3.1.2	Indices binoculaires . . . . .	28
3.2	Contrôles de caméra . . . . .	28
3.3	Contrôles de caméra utilisés dans les logiciels 3D . . . . .	29

3.3.1	Translation . . . . .	30
3.3.2	Zoom . . . . .	31
3.3.3	Rotation égocentrique . . . . .	31
3.3.4	Rotation exocentrique . . . . .	32
3.3.5	Récapitulatif . . . . .	32
3.4	Techniques d'interaction pour le contrôle du point de vue . . . . .	33
3.4.1	Zoom . . . . .	33
3.4.2	Translation . . . . .	34
3.4.3	Rotation égocentrique . . . . .	39
3.4.4	Translation + rotation égocentrique . . . . .	41
3.4.5	Rotation exocentrique . . . . .	48
3.5	Conclusion . . . . .	56
<b>4</b>	<b>Contrôle du point de vue : précision utilisateur et matériel</b>	<b>59</b>
4.1	Approche . . . . .	59
4.2	Mesure de bruit du dispositif ART et résolution effective de la tête . . . . .	60
4.2.1	Précision du dispositif ART . . . . .	61
4.2.2	Résolution effective de la tête . . . . .	61
4.3	Élicitation : contrôle de caméra à partir de la tête et des épaules . . . . .	62
4.3.1	Espace de conception : tête + épaules . . . . .	63
4.3.2	Étude utilisateur . . . . .	63
4.3.3	Résultats & analyse . . . . .	65
4.3.4	Conclusion . . . . .	70
4.4	Amplitude confortable . . . . .	70
4.4.1	Étude utilisateur . . . . .	70
4.4.2	Résultats et discussion . . . . .	72
4.5	Résolution utile . . . . .	74
4.5.1	Étude utilisateur . . . . .	74
4.5.2	Résultats et discussion . . . . .	77
4.6	Conclusion . . . . .	77
<b>5</b>	<b>Contrôle du point de vue orbital : fonctions de transfert et techniques d'interaction</b>	<b>79</b>
5.1	Définitions . . . . .	79
5.1.1	Fonction de transfert . . . . .	79
5.1.2	Gain Moteur Écran (Control Display gain) . . . . .	80
5.1.3	Débrayage . . . . .	81
5.1.4	Contrôle en position et en vitesse . . . . .	81
5.1.5	Type de dispositifs . . . . .	81
5.1.6	Correspondance (mapping) . . . . .	82
5.1.7	Conformité au point zéro (nulling compliance) . . . . .	83
5.2	Fonctions de transfert & études . . . . .	84
5.2.1	$Gain_{CD}$ constant . . . . .	84
5.2.2	$Gain_{CD}$ variable . . . . .	86
5.3	Création des techniques d'interaction . . . . .	90
5.3.1	Considérations et réalisation . . . . .	90
5.4	Comparaison de techniques . . . . .	92

5.4.1	Étude utilisateur . . . . .	92
5.4.2	Résultats et discussion . . . . .	94
5.4.3	Modèle . . . . .	96
5.4.4	Conclusion . . . . .	96
5.5	Validité externe . . . . .	97
5.5.1	Étude utilisateur . . . . .	97
5.5.2	Résultats et discussion . . . . .	100
5.5.3	Conclusion . . . . .	101
5.6	Applications . . . . .	102
5.6.1	Modalités d'entrée . . . . .	102
5.6.2	Room planner . . . . .	102
5.6.3	Logiciels de modélisation 3D . . . . .	103
5.6.4	Application au son 3D . . . . .	104
5.7	Conclusion . . . . .	104
<b>6</b>	<b>Conclusion</b>	<b>107</b>
6.1	Contributions de la thèse . . . . .	107
6.1.1	Contribution conceptuelle . . . . .	107
6.1.2	Contribution technique . . . . .	108
6.1.3	Contribution empirique . . . . .	108
6.2	Perspectives . . . . .	109
6.2.1	Perspectives à moyen terme . . . . .	109
6.2.2	Perspectives à long terme . . . . .	110
	<b>Bibliographie</b>	<b>113</b>
	<b>Publications</b>	<b>125</b>



---

## TABLE DES FIGURES

1.1	Visualisation des trajectoires sonores dans deux logiciels de son 3D existants (à gauche <i>Holoedit</i> , à droite <i>Iannix</i> ). Les trajectoires sonores se superposent et s'occulent les unes les autres. . . . .	2
2.1	Outils utilisés dans le studio 105 de la <i>Maison de la Radio</i> pour la spatialisation du son en <i>live</i> . À gauche, le logiciel <i>Performer</i> s'interface avec le matériel de rendu sonore ( <i>Performer Wave 1</i> ) et permet de contrôler la position de restitution du son récupéré au niveau de la console. Avec cette interface, l'ingénieur visualise une représentation 3D du studio de dessus et peut déplacer les sources sonores (boules rouges). Tout déplacement d'une source via cette interface entraîne immédiatement une différence de rendu sonore dans le studio. À droite, la console de mixage de l'ingénieur du son permet de récupérer les sources sonores à partir des différents micros de la scène et de les dispatcher vers l'outil de spatialisation. . . . .	6
2.2	<i>HyperSpace</i> permet de définir une trajectoire 3D en deux étapes : d'abord un tracé en vue de haut, puis un tracé en vue du sol de l'enveloppe d'élévation. . . . .	8
2.3	Exemples d'interfaces dans les logiciels d'édition de son 3D existants. À gauche, <i>IanniX</i> permet de générer des trajectoires de manière algorithmique. À droite, visualisation des trajectoires de plusieurs sources sonores dans <i>Holo-Edit</i> à travers une visualisation 3D perspective. . . . .	8
2.4	Notre champ d'études est à la croisée de plusieurs domaines de recherches tels que la modélisation 3D par le tracé, la visualisation de l'information, la navigation 3D, la vision par ordinateur ou bien encore la création d'animations. . . . .	9
2.5	Les six degrés de liberté : 3 degrés de rotation (lacet, tangage, roulis) et 3 degrés de translation). 10	10
2.6	La <i>SpaceMouse</i> (figure 2.6a) et la <i>SpaceBall</i> (figure 2.6b) permettent de naviguer en trois dimensions grâce à 6 degrés de liberté. À gauche, la <i>SpaceMouse</i> dispose d'un capteur de type élastique mesurant le déplacement ou le couple selon un axe donné en y opposant une force opposée proportionnelle à la distance. À droite, la <i>SpaceBall</i> dispose d'un capteur isométrique mesurant la force exercée sans déplacement. . . . .	11
2.7	À gauche, <i>Rockin' Mouse</i> [Balakrishnan et al., 1997] est une souris arrondie qui peut pivoter vers l'avant et sur les côtés et permet, en plus des deux degrés de liberté en position, de manipuler dans une certaine mesure deux degrés de rotation. À droite, <i>The Two-ball Mouse</i> [MacKenzie et al., 1997] est une souris avec une boule additionnelle qui permet de mesurer sa rotation autour de l'axe vertical. . . . .	11

2.8	À gauche, <i>3D Tractus</i> [Lapides et al., 2006] est une tablette disposée sur un support ajustable en hauteur, permettant de tracer des courbes 3D. Au centre, le <i>CAT</i> [Hachet et al., 2003] est un système permettant à la fois d'interagir en 2D et en 3D. À droite, <i>Mockup Builder</i> [De Araùjo et al., 2012] est un système basé sur une table interactive et des capteurs <i>GameTraks</i> permettant de faire de la modélisation 3D en interagissant à la fois sur la table et au-dessus de celle-ci. . . . .	12
2.9	À gauche, <i>Beyond</i> [Lee et al., 2010] permet d'interagir à la fois sur et "à l'intérieur" d'une table interactive. À droite, dans <i>Pressure-Based 3D Curve Drawing</i> , Lai et Zakaria [2010] utilisent la pression pour exprimer la profondeur de tracé de la courbe. . . . .	13
2.10	À gauche, <i>Tilt Brush</i> permet de dessiner dans l'espace grâce à un contrôleur HTC Vive et de visualiser son dessin avec un casque de réalité virtuelle HTC Vive. Au centre, <i>CavePainting</i> [Keefe et al., 2001] permet de dessiner des objets 3D dans un environnement CAVE avec des outils physiques. À droite, <i>Digital Tape Drawing</i> [Balakrishnan et al., 1999a] est un système simulant le tracé de courbes au ruban adhésif (tape drawing) utilisé dans l'industrie du design automobile. . . . .	13
2.11	À gauche, <i>ShapeTape</i> [Balakrishnan et al., 1999b] permet de tracer des courbes à l'aide d'un ruban magnétique. Au centre, <i>surface Drawing</i> [Schkolne et al., 2001] permet de dessiner des objets 3D et de les manipuler avec des outils tangibles. À droite, <i>ZeroN</i> [Lee et al., 2011] est un système basé sur un aimant lévitant au sein d'un champ magnétique. . . . .	15
2.12	À gauche, les trois projections orthogonales et la vue en perspective dans <i>Blender</i> . À droite, <i>Tangible NURBS</i> [Bae et al., 2004] permet le tracé de courbes sur grand écran à l'aide de poignées tangibles. . . . .	16
2.13	À gauche, <i>Toucheo</i> [Hachet et al., 2011] permet de manipuler des objets 3D en offrant une meilleure perception de la profondeur. À droite, <i>Drawing on Air</i> [Keefe et al., 2007] permet de tracer des courbes en 3D à l'aide d'un bras à retour haptique. . . . .	17
2.14	À gauche, <i>Blender</i> offre à l'utilisateur la possibilité de créer plusieurs types de courbes à travers un menu. À droite, après sélection dans le menu, la courbe est ajoutée à la scène 3D.	18
2.15	À gauche, le tracé de la courbe 3D se fait en plusieurs étapes. L'utilisateur commence par dessiner celle-ci dans le plan $zx$ , puis il dessine son ombre portée sur le sol (plan $xy$ ). Au centre, le système infère la courbe finale. À droite, dans <i>Mockup Builder</i> [De Araùjo et al., 2012], l'utilisateur définit une contrainte de taille sur la forme en cours de création. . . . .	18
2.16	À gauche, <i>Longcat</i> représente la position des sources sonores sous forme de sphères que l'utilisateur peut déplacer avec la souris. L'utilisation d'une ombre portée sur un plan permet de donner une meilleure appréciation de la hauteur de la source audio par rapport au sol. À droite, <i>Spat-scene</i> [Garcia et al., 2017] permet de tracer des trajectoires 3D à partir de plusieurs plans 2D (vue de face et de dessus). . . . .	20
2.17	À gauche, sélection d'un point de contrôle d'une courbe de Bézier dans <i>Blender</i> . Une poignée représentant les trois axes de translation permet de déplacer le point dans l'espace. En déplaçant le point, l'apparence de la courbe est modifiée en temps réel. À gauche, <i>Blender</i> permet d'effacer l'un des points de contrôle de la courbe ou un segment de celle-ci. . . . .	21
2.18	En dessinant par dessus une courbe existante, l'utilisateur remodèle celle-ci ( <i>oversketching</i> [Baudel, 1994]). Cette technique a été utilisée avec succès dans <i>I Love Sketch</i> [Bae et al., 2008].	21
2.19	L'approche pour remodeler une courbe utilisée dans <i>3D Eraser Pen</i> est de revenir en arrière sur le segment de courbe tracé précédemment et de repartir dans la direction souhaitée. La portion de courbe sur laquelle l'utilisateur est revenu est ainsi retirée. . . . .	21

2.20	Il est possible de supprimer une section de courbe en rayant celle-ci. Cette approche est ici utilisée dans <i>Terrain Sketching</i> [Gain et al., 2009] et <i>I Love Sketch</i> [Bae et al., 2008]. . . . .	22
3.1	D'un unique point de vue, la position de la courbe est ambiguë. La courbe passe-t-elle au travers du plan horizontal ? Devant ou derrière l'arête du cube ? . . . . .	25
3.2	À la figure 3.2a, la sphère blanche est occultée par le cube noir au premier plan. Dans la figure 3.2b, deux parallélépipèdes positionnés parallèlement semblent converger l'un vers l'autre dans la scène 3D. . . . .	26
3.3	En connaissant la taille des deux cubes dans la première figure, il est possible de déduire la profondeur à laquelle se situe le cube de droite dans la seconde figure. . . . .	27
3.4	À gauche, le cube rouge masque la boule grise. À droite, la caméra s'est déplacée sur la droite, révélant la boule grise. Tous les éléments de la scène se sont déplacés vers la gauche : les éléments au premier plan semblent plus loin du centre que le cube rouge, qui n'a pratiquement pas bougé. . . . .	27
3.5	À gauche, la vue de l'oeil directeur, à droite celle de l'autre oeil. Cette disparité binoculaire est due à la distance interoculaire. . . . .	28
3.6	Quatre contrôles de caméra différents . . . . .	29
3.7	À gauche, la barre d'outils de <i>Unity</i> . Le bouton avec la main représente la manipulation de la caméra et permet de réaliser de la translation dans la scène. Au centre les différents contrôles de caméra de Cinema 4D, incluant translation dans le plan et rotation. À droite les contrôles de caméra utilisés dans Maya : rotation exocentrique, translation haut/bas et droite/gauche, translation dans l'axe de la profondeur/zoom . . . . .	29
3.8	À gauche, l'interface permettant de contrôler les contrôles de caméra dans <i>Sketchup</i> . Au centre, les contrôles de zoom et de rotation exocentrique sont supportés dans les <i>Sims 2</i> . À droite, les contrôles de caméra utilisés dans 3ds Max : translation de la caméra dans le plan de la profondeur, zoom, roulis de la caméra, translation de la caméra dans le plan de la caméra, rotation exocentrique . . . . .	30
3.9	À gauche, l'interface de contrôle de <i>Google Earth</i> . Le slider, permet de contrôler le niveau de zoom en déplaçant le curseur. Le zoom peut également être contrôlé avec les boutons +/- . Le cercle central avec les flèches directionnelles permet de contrôler la rotation égocentrique de la caméra sur les axes de lacet et de tangage. À droite, le <i>Room Planner</i> d'Ikea supporte les mouvements de zoom et de rotation exocentrique sur les axes de lacet et de tangage. . . . .	30
3.10	La technique <i>Lean and Zoom</i> [Harrison et Dey, 2008] permet à l'utilisateur de contrôler le niveau de zoom sur son écran en se penchant en avant. . . . .	34
3.11	À gauche, l'utilisateur navigue dans la scène en traçant sur un dispositif 2D le chemin que la caméra doit prendre. À gauche, Navidget permet de contrôler le déplacement de la caméra vers des points d'intérêts. . . . .	35
3.12	Cette technique pour tables interactives permet de manipuler la position de la caméra de manière directe à partir d'une vue 2D. . . . .	35
3.13	Un utilisateur dessine le chemin que la caméra devra suivre dans la scène 3D. Le dessin d'un cercle à la fin du tracé signifie que la caméra doit réaliser une rotation complète sur elle-même afin d'obtenir un aperçu de la scène ( <i>overview</i> ) lorsque celle-ci arrive au bout de son chemin. . . . .	36
3.14	Vocabulaire de gestes basés sur le tracé utilisateur . . . . .	36
3.15	Mouvements du corps proposés pour la translation avec la technique <i>LazyNav</i> [Guy et al., 2015]. . . . .	38

3.16 Yamaguchi et al. [2009] utilisent une caméra Pan Tilt Zoom (PTZ) dans un contexte de vidéoconférence pour étendre le champ de vision des participants. . . . .	40
3.17 Approches de navigation 3D décrites par Ware et Osborne [1990]. <i>Eyeball in hand</i> utilise la métaphore d'un œil dont la position serait contrôlée par l'utilisateur. La scène est observée du point de vue de cet œil. Avec <i>Scene in hand</i> , l'utilisateur manipule le dispositif d'entrée comme s'il manipulait directement la scène 3D. Enfin, avec <i>Flying vehicle</i> , l'utilisateur déplace un véhicule dans la scène 3D et la scène est observée du point de vue de celui-ci. Toutes ces approches utilisent un dispositif d'entrée à 6 degrés de liberté. . . . .	42
3.18 Le <i>World in Miniature</i> est un outil pour la navigation et la manipulation d'objets dans des scènes 3D immersives. Celui-ci se présente sous la forme d'une maquette et offre à l'utilisateur une vision d'ensemble de la scène. De plus, celle-ci étant interactive et directement liée à la scène 3D, elle permet à l'utilisateur de déplacer des objets lointains sans changer de point de vue. . . . .	42
3.19 La perspective couplée à la tête renforce la perception de l'utilisateur en adaptant le rendu de la scène 3D par rapport à la position de la tête de l'utilisateur. Ainsi, si celui se penche à gauche (image de droite), la scène est translatée sur la gauche et la caméra s'oriente verticalement pour conserver le point central de la scène au centre de son champ de vision. . . . .	43
3.20 Le fonctionnement de la perspective couplée à la tête est le suivant : pour une translation donnée de la tête de l'utilisateur, le système opère une translation de la caméra dans la scène 3D et une rotation de la caméra sur elle-même de sorte que celle-ci pointe toujours vers le même point. . . . .	44
3.21 Sur les deux figures, un objet 3D est situé à une profondeur plus faible que le centre de rotation de la perspective couplée à la tête (parallaxe négative). L'objet paraît donc sortir de l'écran. À gauche, l'objet paraît tronqué du fait de la limite de l'écran, ce qui rend le rendu inconstant. À droite, en ajoutant un cadre virtuel entre la projection de l'objet et les yeux de l'utilisateur, l'objet paraît tronqué par le cadre virtuel et non plus par les limites de l'écran. . . . .	45
3.22 La perspective couplée à la tête est utilisée par Harrison et Hudson [2008] afin d'augmenter la présence dans le contexte de la vidéoconférence en adaptant la perspective par rapport à la position de la tête. . . . .	45
3.23 À gauche, la technique <i>Zooming</i> permet à l'utilisateur de modifier le niveau de zoom en se penchant en avant. Deux techniques sont proposées. La première permet de contrôler le niveau de zoom de manière continue, tandis que la seconde donne un niveau de zoom maximal passé un certain seuil. À droite, la technique <i>Peering</i> permet de jeter un coup d'oeil derrière un obstacle. Pour réaliser ce mouvement, l'utilisateur penche la tête à droite ou à gauche. . . . .	46
3.24 À gauche, la technique <i>Iron Sight</i> permet à l'utilisateur d'aligner le point de vue avec la ligne de mire de son arme virtuelle. Ce mouvement est réalisé en penchant la tête sur le côté et s'active passé un certain seuil. À droite, la technique d'immersion <i>Handy-cam</i> ajoute du bruit au niveau du point de vue pour donner l'impression que la caméra n'est pas fixée sur un support, mais tenue par un être humain. . . . .	46
3.25 À gauche, la perspective couplée à la tête permet de facilement obtenir une perception augmentée d'immersion, à travers la parallaxe de mouvement. À droite, un exemple de contrôle du point de vue dans une scène 3D avec la perspective couplée à la tête. . . . .	47
3.26 À gauche, <i>The Interaction Slippers</i> permet de faire apparaître ou disparaître le WIM et de se déplacer au sein de celui-ci. À droite, l'utilisateur se penche pour naviguer dans la scène. . . . .	47
3.27 Sliders . . . . .	48

3.28	La technique <i>Speed-coupled Flying with Orbit</i> couple la vitesse de déplacement dans la scène à la hauteur de la caméra. Ainsi, si l'utilisateur se déplace lentement, il se déplace au raz du sol (figure 3.28a). En revanche, avec une vitesse rapide, la caméra offre un aperçu de la scène (figure 3.28b). . . . .	49
3.29	À gauche, l'utilisateur contrôle la position de la caméra par rapport au cube rouge grâce à un contrôle orbital opéré avec la main non dominante. Afin de valider l'essai, il doit trouver la cible jaune qui est positionnée dans une des cases du cube. La sélection de la cible se fait avec la main dominante. À droite, l'utilisateur doit placer la sphère bleue dans la cible violette positionnée dans le coin de la scène diamétralement opposé. . . . .	50
3.30	À gauche, représentation des 26 points de vue du <i>ViewCube</i> [Khan et al., 2008], un élément d'interface pour le contrôle orbital et la visualisation de l'orientation de la scène dans l'espace. À droite, sélection d'une arête, d'une face ou d'un coin. Après sélection, la scène transitionne vers l'orientation sélectionnée. . . . .	50
3.31	À gauche, <i>HoverCam</i> [Khan et al., 2005] est une technique pour le contrôle orbital qui se base sur la topologie de l'objet afin de maintenir une distance constante au cours du déplacement et ainsi empêcher les collisions dans le cas d'objets convexes. À droite, les différentes zones de la technique <i>HoverCam</i> . . . . .	51
3.32	À gauche, la technique <i>HoverCam</i> [Khan et al., 2005] souffre de problèmes de stabilité dans certains cas où le modèle exploré possède des coins ou des creux (modèle concave). À gauche, la technique <i>SHOCam</i> [Ortega et al., 2015] résout ces problèmes. . . . .	52
3.33	Automated Camera Control for Drawing permet de dessiner sur des objets 3D en adaptant le point de vue de manière automatique afin d'éviter toute occlusion. . . . .	52
3.34	La tâche de l'utilisateur consiste à envoyer un rayon vers la sphère centrale (bleue et rouge) représentant une tumeur tout en évitant le plus possible d'entrer en intersection avec les sphères voisines représentant les organes du patient . . . . .	53
3.35	À gauche, l'utilisateur doit réaliser une tâche d'alignement d'objets 3D. Le participant visualise la position de sa main dans la scène grâce au capteur magnétique <i>Polhemus FASTRAK</i> dans un environnement immersif grâce à son casque de réalité virtuelle. Au centre, représentation du contrôle orbital. À droite, représentation du contrôle égocentrique. . . . .	54
3.36	À gauche, déplacement de la main dans le référentiel du point de vue utilisateur si aucun objet n'a été saisi par celui-ci. À droite, si l'utilisateur a saisi un objet dans la scène pour le déplacer, la main reste fixe, tandis que le point de vue tourne autour du point central de la scène. . . . .	55
3.37	Contrôle du point de vue hybride permettant de combiner du contrôle orbital au contrôle égocentrique avec $VC = 0.5$ . $eye_{ego}$ représente la position et l'orientation du point de vue égocentrique tandis que $eye_{orb}$ représente la position et l'orientation du point de vue orbital. Avec $VC = 0.5$ , la position du point de vue se trouve à mi-chemin entre les points de vue égocentrique et orbital. . . . .	55
3.38	Contrôle de la rotation exocentrique à l'aide des pieds. Le déplacement en profondeur du pied contrôle la rotation sur le lacet tandis que le déplacement du pied à l'horizontal contrôle la rotation sur le tangage. . . . .	56

4.1	Afin de pouvoir manipuler le contrôle orbital (6) en utilisant les mouvements de la tête, plusieurs étapes sont nécessaires. Pour opérer une rotation de 360° dans la scène 3D, il faut passer par une fonction de transfert (5). Afin de déterminer cette fonction de transfert, il faut déterminer l'axe de la tête à utiliser (2), le plus grand mouvement confortable réalisable avec la tête (3) et la résolution minimale de l'utilisateur (4). Dans un premier temps, il faut tout d'abord déterminer la résolution du système de suivi (1).	60
4.2	Nous utilisons le système de tracking infrarouge ART. À gauche, une cible à marqueurs passifs de type Tree Target que nous utilisons tout au long de nos expériences. À droite, une des neuf caméras infrarouges de notre système.	61
4.3	Bruit global sur une heure d'enregistrement pour la rotation sur l'axe y.	62
4.4	À gauche, la jauge permet au participant de visualiser la rotation de sa tête et de se recentrer le cas échéant. À droite, nous utilisons un casque de chantier sur lequel nous avons fixé la cible ART afin d'avoir un support rigide pour nos expériences.	62
4.5	Espace de conception des mouvements de la tête et des épaules pour le contrôle de caméra	64
4.6	Quatre contrôles de caméra différents	65
4.7	Présentation de la vidéo du mouvement de rotation exocentrique par rapport à l'axe horizontal aux participants. À gauche, le début du mouvement. À droite, la fin du mouvement après que la caméra ait opéré une rotation sur l'axe horizontal en se déplaçant vers le haut.	66
4.8	À gauche, mouvement de zoom arrière proposé par l'un des participants en vue de face. Les points rouges représentent la position de la tête dans l'espace pendant le mouvement et les points verts la position des épaules. À droite, mouvement de zoom arrière proposé par l'un des participants en vue de haut.	66
4.9	Équation de Wobbrock pour la mesure du consensus entre utilisateurs.	66
4.10	Consensus entre utilisateurs pour les mouvements de la tête et des épaules proposés. Les mouvements sur un axe identique, mais de directions opposées ont été regroupés au sein de la même catégorie.	67
4.11	Mouvements de la tête et des épaules proposés pour chaque contrôle de caméra. Les trois représentations face (F), haut (H) et côté (C) représentent les mouvements de la tête (en rouge) et des épaules (en vert et bleu) enregistrés par le système de suivi infrarouge.	69
4.12	Le dispositif expérimental était constitué d'un écran externe de 17 pouces relié à un MacBook Pro, d'un clavier et d'une souris.	71
4.13	À gauche, avant chaque condition, l'utilisateur doit se placer en position de repos. Les deux jauges représentent l'axe y (lacet) et l'axe z (roulis). Le participant doit aligner le curseur vert dans la zone verte. À droite, l'utilisateur doit trouver le bon mot parmi une liste de distracteurs pour valider l'essai.	72
4.14	À gauche, les amplitudes d'angles avant le début de l'essai pour les 4 conditions (roulis gauche, roulis droit, lacet gauche, lacet droit). À droite, les amplitudes d'angles au début de l'essai pour les 4 conditions.	74
4.15	(a) le participant commence par orienter sa tête selon l'axe indiqué dans la bonne direction afin de placer le curseur blanc au sein de l'intervalle jaune. (b) Lorsque le participant est bien placé, l'intervalle devient vert et celui-ci peut appuyer sur la barre espace pour passer à la suite de l'essai.	75
4.16	La valeur d'un <i>count</i> correspond à la valeur maximale entre la résolution du dispositif et la valeur effective de la tête	75

4.17	(a) Le participant est en train de réaliser le déplacement afin de valider l'essai. (b) Le participant vient de valider l'essai. (c) Le participant a échoué à l'essai en dépassant l'amplitude maximale autorisée. (d) L'essai est annulé, car le participant est revenu en arrière sur son mouvement. . . . .	76
4.18	Lorsque l'orientation initiale est nulle, le nombre de tentatives pour réussir l'essai est à peu près constant, mis à part pour une valeur de 1 <i>count</i> où les participants sont plus vigilants. En revanche, lorsque l'orientation initiale est différente de 0, le nombre de tentatives semble augmenter avec la précision requise. . . . .	77
4.19	Le taux de succès sur l'ensemble des participants en pourcentages sur les deux axes avec les deux orientations initiales et les amplitudes affichées en <i>count</i> . . . . .	78
5.1	À gauche, le $gain_{CD}$ est inférieur à 1. Tout mouvement dans l'espace moteur produit un mouvement d'amplitude plus faible en sortie (ici le mouvement du curseur). L'utilisateur perd en amplitude, mais gagne en précision. À droite, le $gain_{CD}$ est supérieur à 1. Tout mouvement dans l'espace moteur est donc, à l'inverse, plus grand en sortie. L'utilisateur gagne en amplitude, mais perd en précision. . . . .	80
5.2	Exemple de $gain_{CD}$ variable en fonction de la vitesse . . . . .	81
5.3	Correspondance absolue . . . . .	82
5.4	Correspondance relative . . . . .	83
5.5	Conformité en zéro . . . . .	83
5.6	À gauche, un <i>Wanda</i> est utilisé au cours de l'expérience dans la main non dominante de l'utilisateur pour valider l'orientation du modèle. À droite, un capteur magnétique <i>Polhemus FASTRAK</i> à 6 degrés de liberté permet d'orienter le modèle dans l'espace. . . . .	85
5.7	À gauche, courbe représentant l'évolution du gain en fonction de la distance angulaire pour $\omega_0 = 5$ et $c = \frac{1}{20}$ . À droite, fonction de transfert couplant l'utilisation d'un $gain_{CD}$ fixe à un $gain_{CD}$ variable. La valeur de $gain_{CD}$ $g$ est égale à 1 si la distance angulaire de l'utilisateur $\omega$ est inférieure à une distance angulaire seuil $\omega_0$ . Autrement, la valeur du $gain_{CD}$ augmente de manière avec la distance et un facteur $c$ . . . . .	86
5.8	Fonction de transfert définissant la correspondance entre la distance du périphérique suivi par rapport à l'origine dans le monde réel et sa distance dans le monde virtuel. La variable $d$ représente la distance dans le monde réel, $d_0$ le rayon de la sphère définissant le LVS et $D_0$ le rayon de la sphère définissant le DOI . . . . .	87
5.9	Fonction de transfert définissant la valeur du $gain_{CD}$ pour la technique <i>PRISM</i> . Si la vitesse des mouvements de l'utilisateur est inférieure à <i>MinS</i> , ceux-ci ne sont pas pris en compte. Si leur vitesse est comprise entre <i>MinS</i> et <i>SC</i> , le $gain_{CD}$ est le ratio entre la vitesse et <i>SC</i> . Si la vitesse est comprise, entre <i>SC</i> et <i>MaxS</i> le $gain_{CD}$ est de 1. Enfin, si la vitesse est supérieure à <i>MaxS</i> , le $gain_{CD}$ augmente brièvement afin de rattraper le décalage accumulé. . . . .	88
5.10	Normalisation de la vitesse $v_x(t)$ et du décalage $d_x(t)$ avec la technique <i>Adaptive Pointing</i> . La valeur du décalage $d_x(t)$ est la différence entre le déplacement effectif de l'utilisateur à un instant $t$ , $x_{mot}(t)$ et la valeur de déplacement du système au temps $t - 1$ , $x_{disp}(t - 1)$ . . . . .	88
5.11	Fonction de transfert définissant la valeur du $gain_{CD}$ pour la technique <i>Adaptive pointing</i> [König et al., 2009]. La fonction périodique permet une transition plus souple lors du passage de correspondance relative à correspondance absolue. . . . .	89
5.12	Fonction de transfert définissant la valeur du $gain_{CD}$ . $\alpha$ représente la valeur minimale du $gain_{CD}$ , $\beta$ le facteur d'amplification du $gain_{CD}$ , $b$ la valeur seuil au-delà de laquelle le $gain_{CD}$ est constant et $p$ l'Inclinaison de la tête . . . . .	89

5.13	Fonction de transfert définissant la valeur du $gain_{CD}$ . $\dot{p}$ représente la dérivée de l'inclinaison de la tête par rapport au temps, $\delta$ la valeur de $gain_{CD}$ maximale et $\epsilon$ un facteur d'amplification du $gain_{CD}$ . . . . .	90
5.14	Fonction de transfert définissant la valeur du $gain_{CD}$ . Celui-ci est constant et le contrôle se fait en position tant que l'inclinaison de la tête reste sous un seuil $c$ . Passé ce seuil, le contrôle se fait en vitesse. Les variables $\alpha_2$ et $\gamma$ représentent respectivement la valeur de $gain_{CD}$ minimale et un facteur d'amplification . . . . .	90
5.15	Le facteur de $gain_{CD}$ permettant d'effectuer $180^\circ$ de rotation est le ratio entre l'amplitude de rotation maximale de la caméra dans la scène 3D et l'amplitude maximale de la tête sur un axe donné. . . . .	91
5.16	Fonctions de transfert pour le contrôle de la rotation exocentrique sur le lacet et le roulis . .	91
5.17	À gauche, la scène vue de dessus montrant les quatre cadrant dans lesquels la cible peut être positionnée ainsi que les distances angulaires. À droite, une vue prise de la caméra de point de vue montrant la flèche verte indiquant la direction du mouvement à effectuer et la cible à atteindre. . . . .	93
5.18	L'utilisateur doit 1) ramasser une balle positionnée sur le sol de la scène 3D en utilisant la souris ; 2) orienter la vue (en utilisant soit le lacet, soit le roulis de la tête en fonction de la condition) dans la direction de la cible (qui est indiquée par une flèche verte au sol) ; 3) lancer la balle sur la cible en cliquant le bouton de la souris une fois que la vue est alignée avec la cible (cible avec un halo vert). . . . .	94
5.19	À gauche, le taux de succès pour chaque technique en fonction de la direction. À droite, le temps de sélection pour chaque technique en fonction de la direction. . . . .	95
5.20	Équations linéaires modélisant les résultats obtenus pour les deux techniques. . . . .	96
5.21	Régression linéaire modélisant les rotations sur le roulis et le lacet. . . . .	97
5.22	Le participant pouvait déplacer l'objet (ball) dans la cible (target) de trois manières différentes : 1) en utilisant les axes ( $x$ , $y$ et $z$ ) ; 2) En utilisant les plans ( $xy$ , $yz$ et $xz$ ) ; 3) ou en manipulant la boule directement dans le plan parallèle à celui de la caméra de point de vue. . . . .	99
5.23	À gauche, le temps total pour chaque technique en fonction de la distance angulaire. À droite, le temps de retour. . . . .	100
5.24	À gauche, la webcam utilisée pour nos applications. À droite, la centrale inertielle ainsi que l'Arduino Nano utilisés pour mesurer les déplacements de l'utilisateur. . . . .	102
5.25	Contrôle du point de vue avec la tête dans le contexte d'un <i>Room Planner</i> . L'utilisateur garde les mains libres pour réaliser d'autres tâches, par exemple dessiner. . . . .	103
5.26	Contrôle du point de vue dans le moteur de jeux vidéo <i>Unity</i> avec le roulis de la tête. . . . .	103
5.27	À gauche, contrôle du point de vue dans le logiciel <i>Blender</i> en utilisant le roulis de la tête. À droite, la centrale inertielle installée sur un casque audio. . . . .	104
5.28	À gauche, vue principale du logiciel <i>Performer</i> utilisé par les ingénieurs du son de <i>Radio France</i> pour contrôler le positionnement des sources sonores dans l'espace. À droite, utilisation des mouvements de la tête pour contrôler le point de vue orbital laissant les mains libres. . . . .	105
6.1	À gauche, la centrale inertielle ainsi que l'Arduino Nano utilisés pour mesurer les déplacements de l'utilisateur. À droite, la centrale inertielle installée sur un casque audio . . . . .	111

---

## LISTE DES TABLEAUX

3.1	Contrôles de caméra couramment utilisés dans les logiciels 3D . . . . .	32
4.1	Valeur moyenne et intervalles de confiances à 95% (CI) pour l' <i>amplitude maximale confortable</i> en fonction de la rotation et de la direction. . . . .	74
5.1	Correspondance et type de contrôle utilisés en fonction du type de dispositif et de l'application	83
5.2	Temps de sélection et temps de retour pour les techniques Roulis et Lacet dans chaque direction, pour chaque largeur de cible et distance. . . . .	96
5.3	Taux de succès moyen pour chaque technique en fonction de la direction et de la largeur de cible . . . . .	97
5.4	Temps total, temps d'alignement et temps de retour pour les techniques Roulis et Clavier/Souris dans chaque direction. . . . .	101



---

## INTRODUCTION

Les techniques de son 3D visent à restituer l'aspect tridimensionnel du son, c'est-à-dire permettre à l'auditeur de se situer et de localiser les éléments sonores dans l'espace à trois dimensions. Ce domaine est aujourd'hui en pleine effervescence du fait d'une combinaison de facteurs : la normalisation de nouveaux formats (ADM, Dolby ATMOS, DTS MDA, Auro-3D), l'engouement de grands acteurs (Radio France, BBC, Dolby, Sennheiser...), l'équipement d'un nombre croissant de salles de cinéma en Dolby Atmos, et enfin l'intérêt du public pour les événements *live* et les contenus mis à disposition sur le Web<sup>1</sup>.

Les travaux effectués dans ce domaine se sont pour l'instant principalement focalisés sur le codage et le traitement du son 3D. Or la création interactive de contenus sonores 3D n'est pas un problème trivial, car elle nécessite de tracer ou d'éditer des trajectoires en trois dimensions pour animer des sources sonores dans l'espace. Compte tenu de la dimension temporelle du son, il est donc nécessaire de permettre à l'utilisateur de contrôler ces quatre dimensions. Le sujet principal de cette thèse, effectuée dans le cadre du projet ANR EDISON3D<sup>2</sup>, se plaçait dans cette perspective, son objectif étant de faciliter l'édition et la manipulation du son 3D par les compositeurs et les ingénieurs du son.

L'approche utilisée dans les logiciels de son 3D existants est généralement limitée (les sources sonores ne peuvent être positionnées qu'en deux dimensions) ou peu intuitive. Dans le second cas, le tracé se fait typiquement en deux étapes, d'abord en positionnant les sources sur un plan puis en ajustant leur élévation. Compte tenu du nombre de sources sonores, ce processus est fastidieux, et particulièrement inadapté à des événements "live" où un DJ doit pouvoir contrôler les sources en temps réel. De plus, la représentation des trajectoires des sources sonores est vite problématique, ces trajectoires se superposant et s'occultant mutuellement (voir figure 1.1).

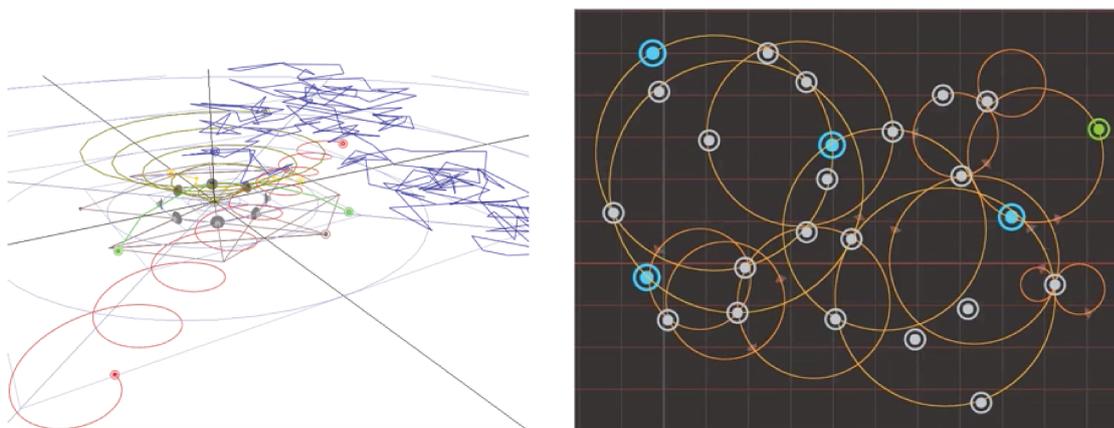
Dans cette thèse, nous avons considéré l'édition de trajectoires de sources sonores en 3D comme un cas particulier de tâche de modélisation 3D. Ce sujet a déjà été largement étudié au sein des communautés IHM et graphique, tant sur le plan logiciel que matériel, ce qui nous a tout d'abord conduits à réaliser un état de l'art visant à classer ces différents travaux [Jacob et al., 2014].

---

1. par exemple les contenus au format binaural sur le site *Hyperradio de Radio France* : <http://hyperradio.radiofrance.fr/>

2. Cette thèse, financée par le projet ANR 13 CORD 008, a été effectuée au sein du consortium EDISON3D visant à promouvoir le développement d'outils pour la création de contenus liés au son 3D.

Malgré la pertinence des solutions proposées, cette problématique est loin d'être totalement résolue, en particulier lorsque l'édition doit être rapide, raisonnablement intuitive et ne pas requérir l'emploi de matériels trop exotiques. C'est typiquement le cas des compositeurs et des ingénieurs du son. Ils sont en effet soumis à de fortes contraintes de production (en particulier de temps, mais aussi de matériel, dans le cas des événements "live"), ne sont pas des experts de l'édition 3D, et doivent réaliser simultanément bien d'autres tâches, généralement plus prioritaires que la spatialisation. Nous avons donc placé notre étude dans ce cadre, en prenant pour hypothèse l'utilisation de matériels standards comme un écran non stéréoscopique et une souris (alternativement un trackpad ou une tablette tactile) accompagnés de dispositifs courants comme une webcam ou une centrale inertielle.



**FIGURE 1.1** – Visualisation des trajectoires sonores dans deux logiciels de son 3D existants (à gauche *Holoedit*, à droite *Iannix*). Les trajectoires sonores se superposent et s'occultent les unes les autres.

Ce cas de figure n'est pas limité aux besoins de l'édition du son 3D. Bien que l'utilisation de la 3D soit devenue courante (modélisation 3D, jeux vidéo, etc.), la majorité des logiciels permettant de créer et visualiser des contenus 3D, ou d'interagir avec eux, repose toujours sur l'utilisation d'un écran standard et d'une souris (ou équivalent). L'utilisation d'un écran standard pose pourtant un problème majeur dans ce cas de figure : l'occultation des données et le manque de perception de la profondeur.

Une approche courante pour pallier ce problème consiste à permettre à l'utilisateur de contrôler le point de vue dans la scène 3D à l'aide de la souris ou du clavier. Cette solution conduit à ce que les mains soient à la fois utilisées pour les tâches d'édition en tant que telles et pour le contrôle du point de vue. Ainsi, dans des outils de modélisation comme *Blender* ou *Maya* les utilisateurs doivent en permanence alterner entre des tâches de contrôle du point de vue (tâches secondaires) et des tâches de création ou manipulation d'objets (tâches primaires). Cette utilisation d'une même modalité, alors que des modalités différentes sont employées dans le monde physique pour chacune de ces tâches, est susceptible de rendre l'interaction plus laborieuse et d'en diminuer l'efficacité. Face à cette problématique, nous proposons une solution innovante permettant de contrôler le point de vue tout en respectant les contraintes des ingénieurs du son.

## 1.1 CONTRIBUTIONS DE LA THÈSE

Dans le cadre de ce travail, nous nous sommes focalisés sur le contrôle du point de vue pour la création et l'édition de trajectoires en trois dimensions. Les contributions de cette thèse sont multiples.

Sur le plan conceptuel, nous avons proposé un espace de conception (chapitre 2) pour le tracé de trajectoires en trois dimensions reposant sur la littérature académique existante. Cet espace de conception présente deux axes : *dispositif*, le dispositif d'entrée ou de sortie utilisé pour tracer ou visualiser la trajectoire ; *langage*, le langage d'interaction utilisé pour créer ou éditer la trajectoire. Ce travail a donné lieu à une publication (voir *Publications*) présentée à la conférence *IHM'14*.

Sur le plan conceptuel également, nous présentons au chapitre 3 une classification des contrôles de caméra existants dans la littérature académique en fonction du type de contrôle et de la modalité utilisée pour contrôler celui-ci.

Sur le plan empirique, nous avons conduit 5 études utilisateurs dans le but de créer une technique d'interaction pour le contrôle du point de vue orbital permettant d'effectuer une rotation de 360° en utilisant le roulis de la tête. Ces études visaient à :

- comprendre comment les utilisateurs associent de manière spontanée un mouvement du corps à un contrôle de caméra (étude par élicitation) (chapitre 4) ;
- déterminer la plus petite rotation de tête réalisable de manière contrôlée (*résolution utile de la tête*) (chapitre 4) ;
- déterminer l'amplitude maximale confortable de rotation de la tête (chapitre 4) ;
- comparer deux fonctions de transfert pour le contrôle du point de vue orbital sur une tâche de pointage (chapitre 5) ;
- vérifier la validité externe de notre technique d'interaction par rapport à la baseline clavier/souris sur une tâche de docking (chapitre 5).

Ces travaux (voir *Publications*) ont été présentés aux conférences *IHM'15* (Travaux en cours) et *SUI'16*.

Enfin, sur le plan technique, nous présentons la création d'une technique d'interaction pour le contrôle orbital à 360° avec le roulis de la tête et son intégration dans le logiciel de modélisation *Blender* ainsi que dans dans *Performer*, l'outil de positionnement des sources sonores utilisé par *Radio France*.

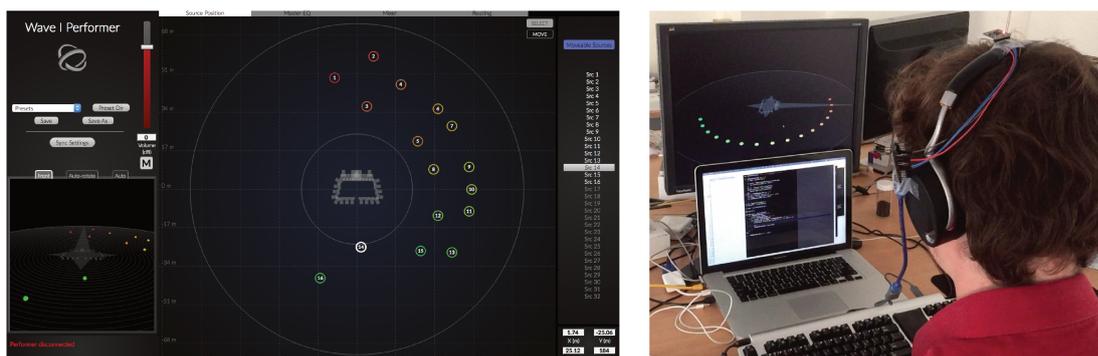
## 1.2 STRUCTURE/ORGANISATION

Dans le chapitre 2, nous décrivons les besoins de nos deux principaux interlocuteurs, *Radio France* et *Sonic Emotion*, suite à la réalisation d'interviews. Celles-ci ont mis en évidence les problématiques liées au tracé de trajectoire en trois dimensions et au contrôle du point de vue. Nous présentons ensuite un espace de conception (chapitre 2, section 2.2.2) pour la création et l'édition de trajectoires en trois dimensions qui s'appuie sur deux axes : *dispositifs* et *langages d'interaction*. Cet espace de conception nous a permis de mettre en évidence l'importance de pouvoir percevoir la profondeur de la scène et l'absence de solutions satisfaisantes pour les ingénieurs du son.

Dans le chapitre 3, nous décrivons les indices visuels permettant à l'être humain de percevoir la profondeur dans son environnement, comme la parallaxe de mouvement et la disparité binoculaire (vision stéréo). Nous présentons dans la suite de ce chapitre les mouvements de caméra existants (*zoom*, *translation*, *rotation égocentrique*, *rotation exocentrique* ou *orbitale*) dans le monde virtuel et dans quels contextes ceux-ci sont utilisés dans les logiciels 3D. Dans la section suivante (section 3.2), nous présentons un ensemble de techniques utilisées pour contrôler le point de vue dans le domaine académique selon deux axes : *contrôle de caméra* et *modalité*. Ce chapitre met en évidence l'importance du contrôle orbital qui permet d'obtenir un aperçu global de la scène 3D en tournant autour de celle-ci.

Dans le chapitre 4, nous présentons notre approche globale pour la création d'une nouvelle technique d'interaction pour le contrôle du point de vue ainsi que les trois premières études utilisateurs. Avant toute chose, nous vérifions la précision de notre système de tracking pour nous assurer que celui-ci est suffisamment précis pour réaliser les mesures. La première étude, une étude par élicitation, permet de comprendre comment les utilisateurs associent de manière spontanée un mouvement du corps à un contrôle de caméra. La seconde étude nous permet de déterminer le plus petit mouvement réalisable de manière contrôlée par l'utilisateur. Enfin, dans la troisième étude nous déterminons l'amplitude maximale confortable de la tête.

Dans le chapitre 5, nous présentons un rapide état de l'art des fonctions de transfert existantes dans la littérature académique (5.2). Nous détaillons ensuite, le raisonnement suivi pour la création de notre technique d'interaction et présentons deux fonctions de transfert. Dans une première étude (5.4), nous comparons ces deux fonctions de transfert sur une tâche de pointage. Dans un second temps, nous comparons la fonction de transfert la plus performante au couple souris/clavier sur une tâche de docking qui nécessite de contrôler le point de vue tout en positionnant un objet (voir section 5.5). Finalement, nous présentons plusieurs applications (voir section 5.6) de notre technique pour contrôler le point de vue à l'aide d'une centrale inertielle, notamment dans le logiciel de modélisation *Blender* et dans le logiciel utilisé par *Radio France* pour placer les sources sonores dans l'espace (figure 5.28).



**FIGURE 1.2** – À gauche, vue principale du logiciel *Performer* utilisé par les ingénieurs du son de *Radio France* pour contrôler le positionnement des sources sonores dans l'espace. À droite, utilisation des mouvements de la tête pour contrôler le point de vue orbital laissant les mains libres.

Enfin, dans la conclusion nous rappelons les contributions de cette thèse et présentons les perspectives à court, moyen et long termes.

---

## ESPACE DE CONCEPTION POUR LE TRACÉ ET L'ÉDITION DE COURBES 3D

Dans ce chapitre, nous présentons dans un premier temps le contexte dans lequel s'est effectuée la thèse, notamment les deux acteurs (*Radio France* et *Sonic Emotion*) avec lesquels nous avons principalement travaillé ainsi que leurs besoins, que nous avons évalués au travers d'études de terrain. Nous avons en particulier conduit des interviews qui ont mis en avant le besoin de tracer et d'éditer des courbes dans l'espace. Ceci nous a conduit à réaliser un état de l'art sur le tracé et l'édition de courbes, un travail permettant d'offrir une vue globale des travaux existants tant dans le monde académique qu'industriel.

### 2.1 ÉTUDE DE TERRAIN

#### 2.1.1 Méthode

Nous avons réalisé des interviews semi-structurées dans les locaux de *Radio France* (studio 105) et de *Sonic Emotion*. Celles-ci ont été enregistrées afin d'être transcrites ultérieurement. Ces transcriptions ont à leur tour permis d'extraire certaines problématiques de *Sonic Emotion* et *Radio France*. Nous en présentons un résumé dans la section suivante.

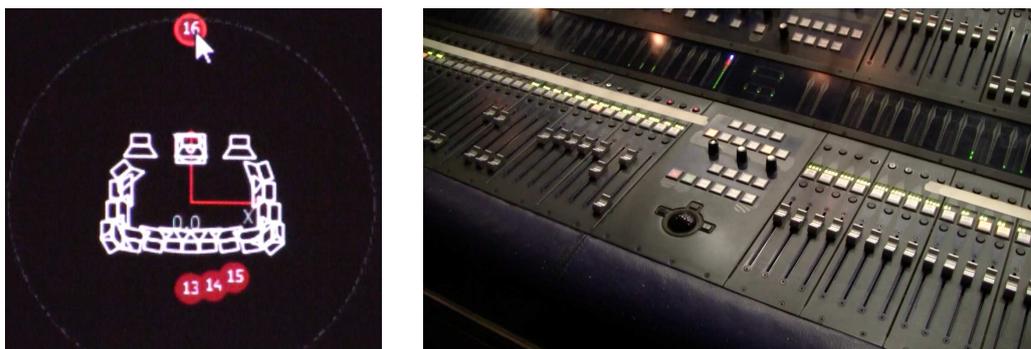
#### 2.1.2 Interviews

##### *Interviews avec les ingénieurs du son de Radio France*

Le son 3D est utilisé dans deux contextes différents par *Radio France*. Dans le contexte *live*, le son est enregistré à partir d'une performance avec des artistes et un éventuel public. Dans le contexte *studio*, le son enregistré dans le contexte *live* est traité pour une diffusion ultérieure ou une mise à disposition du public via la plateforme *Hyperradio*<sup>1</sup> de *Radio France*. Afin d'en apprendre plus sur ces deux contextes, nous avons donc conduit des interviews avec les ingénieurs du son de *Radio France*.

---

1. <http://hyperradio.radiofrance.fr>



(a) Interface du Performer Wave 1

(b) Console de mixage de l'ingénieur du son

**FIGURE 2.1** – Outils utilisés dans le studio 105 de la *Maison de la Radio* pour la spatialisation du son en *live*. À gauche, le logiciel *Performer* s'interface avec le matériel de rendu sonore (*Performer Wave 1*) et permet de contrôler la position de restitution du son récupéré au niveau de la console. Avec cette interface, l'ingénieur visualise une représentation 3D du studio de dessus et peut déplacer les sources sonores (boules rouges). Tout déplacement d'une source via cette interface entraîne immédiatement une différence de rendu sonore dans le studio. À droite, la console de mixage de l'ingénieur du son permet de récupérer les sources sonores à partir des différents micros de la scène et de les dispatcher vers l'outil de spatialisation.

### **Contexte live**

Un ingénieur du son de *Radio France* (Frédéric Changenet) nous a présenté les étapes suivies afin de procéder à la *phase de balance* de plusieurs groupes musicaux en vue d'un concert studio *live*. La *phase de balance* est l'étape pendant laquelle l'ingénieur du son procède aux réglages sonores, à savoir proportionner les niveaux sonores des instruments de la scène afin d'avoir un mixage cohérent. Pendant cette phase, l'ingénieur du son commence dans un premier temps par attribuer les entrées sonores des micros de chaque instrument sur scène aux entrées de sa console (figure 2.1b). Chaque instrument peut avoir un ou plusieurs micros associés. La phase de balance permet, en utilisant les outils de la console, de contrôler le niveau en entrée de chaque source sonore, retirer des fréquences causant du Larsen ou ajouter des effets (réverbération, écho, etc.).

Le studio 105 de *Radio France* est équipé pour le son 3D grâce à la technologie de spatialisation sonore WFS<sup>1</sup>. Celui-ci dispose d'un équipement spécialisé dans la spatialisation du son, *le Performer Wave 1* qui est produit par *Sonic Emotion*. Avec l'interface de contrôle du *Performer Wave 1* l'ingénieur du son peut, via une représentation du studio (figure 2.1a), positionner une source sonore dans l'espace physique du studio. Par exemple, le son de la voix d'un chanteur peut sembler provenir du milieu du public alors que celui-ci est sur scène. Il suffit à l'ingénieur du son de déplacer la position de la source sonore via l'interface de contrôle (boule rouge) de la figure 2.1a sur la visualisation pour que le système adapte le rendu sonore en temps réel. En revanche, cette interface de contrôle ne permet pas de créer de trajectoires pour animer une source sonore dans l'espace. De plus, il est impossible de positionner une source sonore en hauteur avec l'outil actuel, la visualisation étant exclusivement en vue de dessus.

1. Wave Field Synthesis, une technique de spatialisation basée sur la création de fronts d'ondes artificiels

### Contexte studio

Afin d'en apprendre davantage sur le contexte *studio*, nous avons interviewé Hervé Déjardin également ingénieur du son à *Radio France* qui est chargé de la diffusion du son spatialisé sur internet. Cette interview a mis en évidence que les contraintes du contexte *studio* diffèrent de celles du contexte *live* sur plusieurs points. En effet, là où le *live* laisse peu de place à l'improvisation pour le traitement audio, le contexte *studio* permet d'être plus créatif en laissant l'utilisateur tracer des trajectoires, par exemple avec la souris (figure 2.2) ou même des dispositifs d'interaction "en l'air" comme le Leap Motion<sup>1</sup>. L'interview a également mis en évidence les différences de perception existant entre la forme d'une trajectoire spatiale et la restitution sonore à partir d'un système audio. Par exemple, une trajectoire circulaire sur un système sonore 5.1 sera perçue comme carrée par l'auditeur. Enfin, l'interview a mis en lumière que la plupart des logiciels de montage audio utilisés en production ne supportent pas le positionnement des sources sonores en trois dimensions (par exemple, pas de contrôle de la hauteur).

### Interview avec Sonic Emotion

Notre interview avec *Sonic Emotion* s'est concentrée sur les outils permettant de contrôler la position des sources sonores dans un espace en trois dimensions. Ces outils ont de nombreuses fonctionnalités, mais leur utilisation est souvent perçue comme compliquée, la puissance ayant tendance à être mise en avant au détriment de la simplicité. Afin de gagner en simplicité, il fallait identifier précisément les tâches les plus courantes et en simplifier leur exécution. Un des points identifiés pendant ces interviews a été le manque d'outils permettant de créer des trajectoires sonores facilement dans l'espace. Le son 3D étant un domaine récent en pleine expansion, et encore assez peu connu du grand public, il existe très peu d'outils dédiés aux contenus audio spatialisés. De plus, nous avons appris que la qualité de la perception humaine varie avec l'axe d'écoute (meilleure perception de la position d'un objet lorsque celui-ci tourne autour de l'axe vertical de la tête plutôt que l'axe transversal). Tous les axes ne sont donc pas équivalents.

L'interview avec *Sonic Emotion* a permis de mettre en évidence deux problématiques principales :

- Le tracé et l'édition de courbes 3D :
  - l'approche suivie pour tracer une trajectoire en 3D dans les logiciels de son 3D existants est généralement peu intuitive (lorsqu'elle existe). La plupart permettent de tracer une courbe sur un plan 2D et ensuite de contrôler l'élévation de manière séparée. Ces logiciels permettent certes de définir une trajectoire 3D mais ils ne permettent pas de visualiser celle-ci de manière adaptée ;
  - l'acquisition du tracé de l'utilisateur est imparfaite du fait des tremblements involontaires de l'utilisateur (bruit). La figure 2.2 représentant un tracé de courbe à main levée illustre ce problème : le tracé de l'utilisateur n'est pas fluide ;
- La visualisation de ces courbes et sources/sonores :

Dans des conditions d'utilisations réelles, un enregistrement possède un grand nombre de sources sonores (plusieurs microphones par instrument) possédant chacune une position propre dans l'espace 3D. Il y a donc généralement une grande quantité d'objets à manipuler (figure 2.3a). Ceci pose plusieurs problèmes :

---

1. <https://www.leapmotion.com/>

- comment doit-on afficher les informations liées à ces sources sonores (ex : nom de l'instrument, volume sonore, orientation, etc.) ;
- quelle est la visualisation la plus adaptée pour le contrôle d'objets 3D et leur animation (évolution au cours du temps). Comme on peut le voir sur la figure 2.3b, la trajectoire d'une source sonore peut devenir difficile à suivre ;
- un instrument de musique peut avoir plusieurs microphones qui lui sont associés, donc plusieurs sources sonores du côté de l'ingénieur du son. Afin de pouvoir contrôler celles-ci, l'ingénieur du son doit être en mesure de regrouper ces sources pour pouvoir les déplacer de manière simultanée en manipulant directement ces groupes ou en définissant des comportements (ex : une source sonore suit une autre source, ou un chemin symétrique).

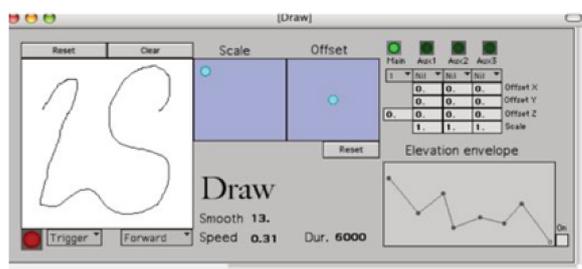
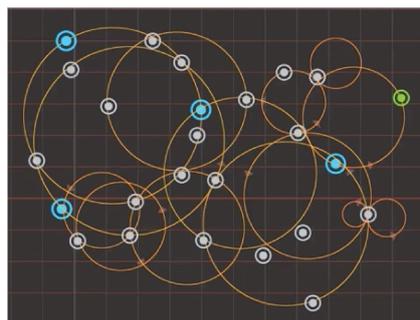
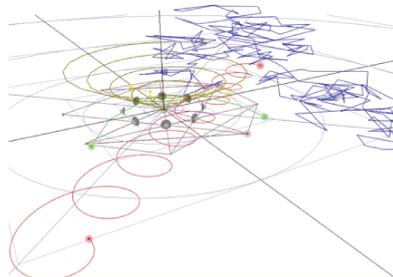


FIGURE 2.2 – *HyperSpace* permet de définir une trajectoire 3D en deux étapes : d'abord un tracé en vue de haut, puis un tracé en vue du sol de l'enveloppe d'élévation.



(a) IanniX



(b) Holo-Edit

FIGURE 2.3 – Exemples d'interfaces dans les logiciels d'édition de son 3D existants. À gauche, *IanniX* permet de générer des trajectoires de manière algorithmique. À droite, visualisation des trajectoires de plusieurs sources sonores dans *Holo-Edit* à travers une visualisation 3D perspective.

### 2.1.3 Conclusion

En conclusion, les interviews et études de terrain ont permis de mettre en avant deux grandes problématiques : d'une part comment tracer et éditer des courbes 3D simplement et efficacement et d'autre part comment visualiser une grande quantité d'informations (groupes, trajectoires).

En conséquence, nous nous sommes intéressé au tracé et à l'édition de courbes 3D et tout d'abord aux travaux académiques effectués dans ce domaine. Notamment nous avons étudié quelles modalités ont été utilisées pour le tracé, l'édition et la visualisation des courbes. Nous en proposons ci-après un espace de conception qui a initialement été publié à IHM'14 ([Jacob et al., 2014]). Nous avons également étudié les approches utilisées pour tracer des trajectoires dans les logiciels de son 3D que nous décrivons dans la section suivante.

## 2.2 ESPACE DE CONCEPTION : DÉFINITION ET COUVERTURE

Nous proposons un espace de conception sur le tracé et l'édition de courbes en trois dimensions, avec le double objectif de mieux comprendre l'étendue des travaux effectués sur ce sujet et proposer de nouvelles techniques d'interaction pour le tracé et l'édition de telles courbes. Nous présentons cet espace de conception dans la suite de ce chapitre.

### 2.2.1 Définition et couverture

#### Définitions

Nous définissons une *courbe en trois dimensions* comme étant une courbe dont les points qui la constituent ont des coordonnées dans un espace en trois dimensions. Le *tracé de courbe 3D* est l'opération permettant à un utilisateur de créer une courbe 3D. Enfin, nous appelons *édition de courbe 3D*, l'opération permettant à un utilisateur de modifier une courbe 3D existante afin d'en changer ses caractéristiques.

#### Couverture

Le tracé et l'édition de courbes sont fréquemment utilisés dans une grande variété d'applications (figure 2.4) comme la visualisation de l'information ([Tsandilas et al., 2015], [Wattenberg, 2001], [Yu et al., 2012]), la navigation 3D ([Igarashi et al., 1998]), la vision par ordinateur, la création d'animations ([Fei et al., 2008], [Santosa et al., 2013], [Wilches et al., 2012]) ou la modélisation (Blender, Maya, [Igarashi et al., 1999], etc.).

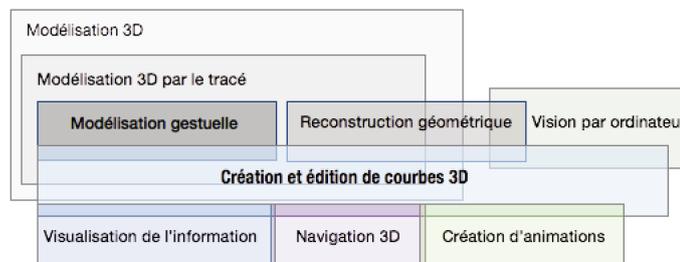


FIGURE 2.4 – Notre champ d'études est à la croisée de plusieurs domaines de recherches tels que la modélisation 3D par le tracé, la visualisation de l'information, la navigation 3D, la vision par ordinateur ou bien encore la création d'animations.

Dans cet espace de conception, nous nous intéressons exclusivement au tracé et à l'édition de courbes 3D.

### 2.2.2 Espace de conception

Notre espace de conception est basé sur la notion de *modalité* telle que définie par Nigay et Coutaz [1996] : une modalité correspond à un couple  $\langle D, L \rangle$ , où  $D$  est un *dispositif physique* et  $L$  un *langage d'interaction*. Un dispositif physique est une interface utilisateur qui permet d'acquérir (dispositif

d'entrée) ou de fournir des informations (dispositif de sortie) à l'utilisateur. Dans ce cadre, un langage d'interaction est défini comme étant un ensemble d'actions réalisables sur le système par l'utilisateur à l'aide de ces dispositifs. Dans cet espace de conception, nous distinguons d'abord les dispositifs d'entrée et les dispositifs de sortie. Nous présentons ensuite les langages d'interaction pour le tracé et l'édition de courbes.

## 2.3 DISPOSITIFS D'ENTRÉE

### 2.3.1 Souris + clavier

Le couple souris + clavier reste le dispositif d'entrée le plus utilisé pour le tracé et l'édition de contenu 3D. Ceci est sans doute dû à son faible prix et à son confort d'utilisation. Cependant, ce dispositif n'est pas le plus adapté pour effectuer des tâches de manipulation dans un espace en trois dimensions. En effet, pour interagir de manière complète dans un espace en trois dimensions, l'utilisateur doit avoir accès à six degrés de liberté (3 degrés de rotation et 3 degrés de translation, voir figure 2.5). Si ce n'est pas le cas, l'utilisateur doit passer par des changements de modes (en sélectionnant un mode dans un menu ou via la barre d'outils) ou utiliser un clavier dont certaines touches joueront le rôle de "modificateurs" qui serviront à changer de mode. Ceci a pour effet de rendre la tâche plus longue et plus ardue pour l'utilisateur.

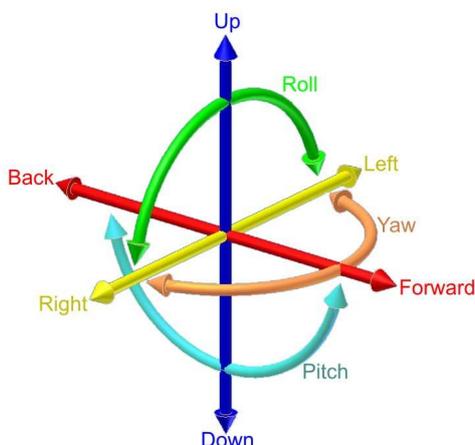


FIGURE 2.5 – Les six degrés de liberté : 3 degrés de rotation (lacet, tangage, roulis) et 3 degrés de translation).

Nous présentons ci-après divers dispositifs permettant de pallier les limitations du couple souris + clavier.

### 2.3.2 Souris augmentées

Il existe une grande variété de dispositifs de souris augmentant la souris traditionnelle pour naviguer dans une scène 3D avec six degrés de liberté. Ainsi, 3Dconnexion<sup>1</sup>, a créé plusieurs modèles de souris augmentées comme la *SpaceMouse* (figure 2.6a) et la *SpaceBall* (qui n'est plus commercialisée, figure

1. Une compagnie spécialisée dans les dispositifs d'interaction pour la navigation 3D



(a) SpaceMouse



(b) SpaceBall

**FIGURE 2.6** – La *SpaceMouse* (figure 2.6a) et la *SpaceBall* (figure 2.6b) permettent de naviguer en trois dimensions grâce à 6 degrés de liberté. À gauche, la *SpaceMouse* dispose d'un capteur de type élastique mesurant le déplacement ou le couple selon un axe donné en y opposant une force opposée proportionnelle à la distance. À droite, la *SpaceBall* dispose d'un capteur isométrique mesurant la force exercée sans déplacement.

2.6b). Ces deux modèles de souris fournissent un contrôle en vitesse, à la manière d'un joystick, dans le sens où elles sont fixes et renvoient une valeur en fonction de la pression appliquée par l'utilisateur. La principale différence entre ces deux modèles est relative au type de capteur utilisé. En effet, la *SpaceMouse* dispose d'un capteur de type élastique [Zhai, 1995] mesurant le déplacement ou le couple selon un axe en y opposant une force contraire proportionnelle à la distance, tandis que la *SpaceBall* dispose d'un capteur isométrique, qui mesure la force exercée sans déplacement. [Zhai, 1995] a montré que les dispositifs élastiques étaient mieux adaptés au contrôle en vitesse que les dispositifs isométriques. Ces deux modèles nécessitent de plus une souris classique pour toute tâche de pointage (sélection de commandes, etc.).



(a) *Rockin' Mouse* [Balakrishnan et al., 1997]



(b) *The Two-ball Mouse* [MacKenzie et al., 1997]

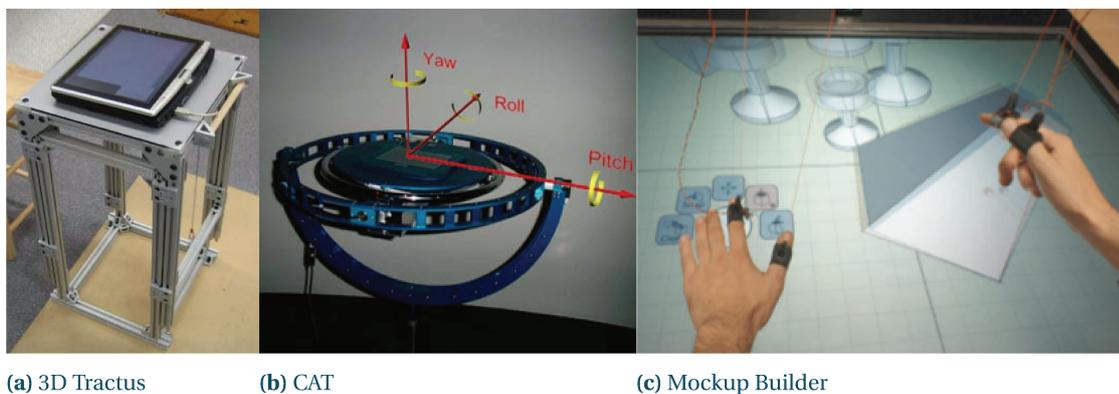
**FIGURE 2.7** – À gauche, *Rockin' Mouse* [Balakrishnan et al., 1997] est une souris arrondie qui peut pivoter vers l'avant et sur les côtés et permet, en plus des deux degrés de liberté en position, de manipuler dans une certaine mesure deux degrés de rotation. À droite, *The Two-ball Mouse* [MacKenzie et al., 1997] est une souris avec une boule additionnelle qui permet de mesurer sa rotation autour de l'axe vertical.

Certains prototypes académiques ont cherché un compromis entre souris classique et souris augmentée en permettant de contrôler davantage de dimensions qu'avec une souris traditionnelle, tout en gardant la possibilité de pointer avec celle-ci. *Rockin' Mouse* [Balakrishnan et al., 1997] offre par exemple 4 degrés de liberté : 2 degrés de translation et 2 de rotation grâce à sa forme arrondie qui lui permet

d'être inclinée vers l'avant ou sur les côtés (voir figure 2.7a). Dans *The Two-ball Mouse* [MacKenzie et al., 1997], les auteurs présentent une souris capable de détecter 2 degrés de translation et 1 de rotation en l'augmentant d'une seconde boule (figure 2.7b). Grâce à cet ajout, la souris peut mesurer sa rotation autour de l'axe vertical. En se basant sur le prototype précédent, Hinckley et al. ont créé la *VideoMouse* [Hinckley et al., 1999], une souris permettant de contrôler jusqu'à 6 degrés de liberté. L'approche suivie est une combinaison des deux travaux précédemment cités. Ainsi la rotation selon l'axe vertical suit l'approche décrite dans *Two-ball Mouse* tandis que les deux autres rotations suivent celle de *Rockin' Mouse*. Le pointage sur l'axe vertical se fait en levant la souris jusqu'à 3 cm de hauteur.

En conclusion, les souris augmentées remplissent des fonctions particulières mais certaines (*SpaceBall*, *SpaceMouse*) ne peuvent être utilisées sans souris classiques, car leurs modalités ne sont pas adaptées aux tâches de pointage. Dans la section suivante, nous étudions les dispositifs d'entrée tactiles.

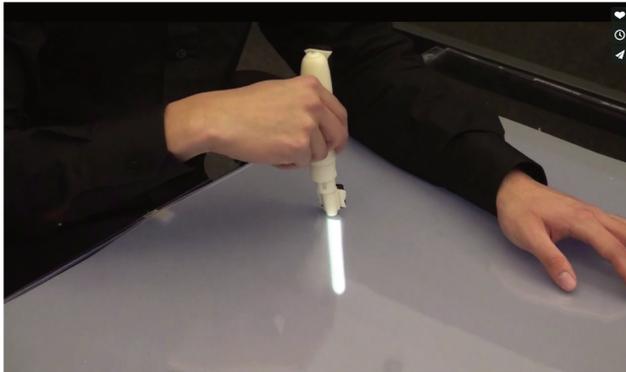
### 2.3.3 Surfaces tactiles



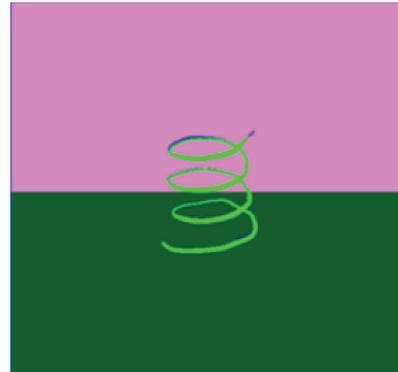
**FIGURE 2.8** – À gauche, *3D Tractus* [Lapides et al., 2006] est une tablette disposée sur un support ajustable en hauteur, permettant de tracer des courbes 3D. Au centre, le *CAT* [Hachet et al., 2003] est un système permettant à la fois d'interagir en 2D et en 3D. À droite, *Mockup Builder* [De Araújo et al., 2012] est un système basé sur une table interactive et des capteurs *GameTraks* permettant de faire de la modélisation 3D en interagissant à la fois sur la table et au-dessus de celle-ci.

Un certain nombre de travaux sur l'interaction 3D s'appuient sur des tablettes ou tables interactives. On trouve par exemple *3D Tractus* [Lapides et al., 2006] (figure 2.8a), un système basé sur l'utilisation d'une tablette montée sur un support contrôlable en élévation. L'utilisateur peut ainsi dessiner en 2D d'une main et contrôler la hauteur du support avec l'autre. L'approche est celle d'un volume virtuel dans lequel la tablette se déplace verticalement, ce qui permet à l'utilisateur de tracer en 3D. La tablette représente le plan sur lequel l'utilisateur trace sa courbe à un moment donné. Ce dispositif permet donc de contrôler les trois degrés de liberté en translation (deux en dessinant sur la tablette et 1 en manipulant l'élévation avec le support). D'autres travaux [Lai et Zakaria, 2010, 2014] (figure 2.9b) utilisent la pression comme entrée supplémentaire. Ainsi, plus la pression exercée sur la tablette est importante, plus le tracé est épais et par métaphore, proche de l'utilisateur. Le *CAT* [Hachet et al., 2003] est un dispositif, dont la surface tactile est montée sur plusieurs cardans afin de pouvoir orienter celle-ci sans contrainte (3 degrés de liberté en rotation). Des capteurs de pression isométriques permettent d'obtenir la translation de la surface selon chaque axe. La surface, placée au centre du dispositif, permet d'interagir en 2D. Il est donc possible d'interagir à la fois en 2D avec la tablette et en 3D avec le reste du dispositif. Avec *Beyond* (figure 2.9a), Lee et al. [2010] proposent un dispositif permettant d'interagir "à l'intérieur" d'une table

interactive. Grâce à un stylet rétractable, et à la visualisation d'un stylet virtuel sur la table, l'utilisateur a la sensation de pouvoir interagir à la fois sur la surface de la table interactive et à l'intérieur de celle-ci. Il peut ainsi contrôler la position de son tracé en trois dimensions.



(a) *Beyond*

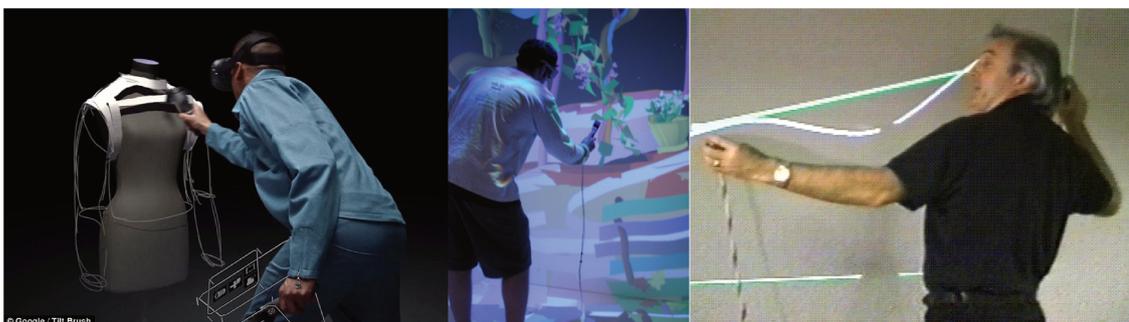


(b) *Pressure-based 3D Curve Drawing*

**FIGURE 2.9** – À gauche, *Beyond* [Lee et al., 2010] permet d'interagir à la fois sur et "à l'intérieur" d'une table interactive. À droite, dans *Pressure-Based 3D Curve Drawing*, Lai et Zakaria [2010] utilisent la pression pour exprimer la profondeur de tracé de la courbe.

D'autres travaux augmentent les tables interactives en permettant d'interagir au-dessus de celles-ci. C'est le cas de *Mockup Builder* [De Araùjo et al., 2012], un système pour la modélisation 3D. L'interaction sur la table interactive permet de créer des formes 2D et de les manipuler. À l'aide de capteurs *GameTraks*, le système permet également d'interagir au-dessus de la table interactive pour réaliser des actions supplémentaires, telles que l'extrusion de formes 2D. Le système *GameTraks* utilise des câbles pour déterminer la position des mains en 3D et limite la fatigue de l'utilisateur en soutenant la main, un problème courant de l'interaction "en l'air".

### 2.3.4 Interaction en l'air



(a) *Tilt Brush*

(b) *CavePainting*

(c) *Digital Tape Drawing*

**FIGURE 2.10** – À gauche, *Tilt Brush* permet de dessiner dans l'espace grâce à un contrôleur HTC Vive et de visualiser son dessin avec un casque de réalité virtuelle HTC Vive. Au centre, *CavePainting* [Keefe et al., 2001] permet de dessiner des objets 3D dans un environnement CAVE avec des outils physiques. À droite, *Digital Tape Drawing* [Balakrishnan et al., 1999a] est un système simulant le tracé de courbes au ruban adhésif (tape drawing) utilisé dans l'industrie du design automobile.

L'interaction "en l'air" peut sembler en adéquation avec l'édition de courbes, car la main a 6 degrés de liberté. Nous distinguons deux classes de dispositifs : ceux laissant libres les mains de l'utilisateur comme le *Kinect*<sup>1</sup> ou le *Leap Motion*<sup>2</sup> et ceux nécessitant de tenir un dispositif.

Une difficulté récurrente de l'interaction dans les airs et du tracé à main levée est de déterminer le début et la fin du geste, ce qui impose d'introduire des gestes supplémentaires servant de délimiteur. On retrouve assez souvent l'utilisation du pouce en tant que délimiteur [Clay et al., 2013, Schkolne et al., 2001] pour le tracé de courbe : le tracé débute quand le pouce est le long de la paume et s'arrête quand le pouce n'est plus en contact avec la paume. Il est intéressant de remarquer que plusieurs approches ont été étudiées pour la sélection de mode (tracé de courbe, remodelage de la courbe, redimensionnement, effaçage de la courbe, manipulation du point de vue). Une approche courante est d'utiliser des gestes spécifiques de la main pour changer de mode [De Araùjo et al., 2012, Wang et al., 2011].

Dans certains cas le système propose en revanche une panoplie d'outils physiques dédiés à chaque mode [Keefe et al., 2001, Schkolne et al., 2001]. Certains systèmes ont été conçus pour l'interaction bimanuelle en utilisant des gants [Schkolne et al., 2001] ou des poignées [Bae et al., 2004, Balakrishnan et al., 1999a, Grossman et al., 2001, 2002]. Par exemple, des travaux effectués chez Alias|Wavefront (figure 2.10c) utilisent des poignées (une dans chaque main), symbolisant les deux extrémités d'un ruban de scotch. Cette technique, basée sur les techniques de conception pour le design automobile s'appuie sur une métaphore, celle d'un utilisateur dessinant avec une bande de ruban adhésif, une main collant le scotch progressivement au support tandis que l'autre tire la bobine et guide le tracé.

En revanche, ces dispositifs peuvent rapidement devenir fatigants pour l'utilisateur si son bras ne repose pas la plupart du temps sur une surface (syndrome du "Gorilla arm") [Hincapié-Ramos et al., 2014].

Une approche proposée par Wang et al. [2011] est d'utiliser deux caméras afin de suivre la position des mains. Ce système est utilisable en position assise, par exemple face à un bureau et permet à l'utilisateur de reposer ses mains. Il peut de plus poser ses coudes sur la table pour réduire la fatigue. Plus récemment, le Leap Motion<sup>3</sup>, un nouveau dispositif doté de plusieurs caméras infrarouges permet de suivre précisément la position et l'orientation des mains. Ce dispositif commence à être utilisé dans des contextes différents allant du tracé de trajectoires 3D à la navigation [Nabiyouni et al., 2014] dans des environnements 3D.

Avec l'augmentation de la puissance des ordinateurs et le développement des jeux vidéo, ces dernières années ont vu l'apparition de casques de réalité virtuelle grand public permettant d'augmenter la sensation des joueurs comme le HTC Vive<sup>4</sup> ou l'Oculus Rift<sup>5</sup>. Ces casques de réalité virtuelle sont la plupart du temps couplés à des contrôleurs sans fils, qui sont visibles par l'utilisateur dans la scène immersive et lui permettent ainsi d'interagir avec la scène 3D. On citera par exemple *Tilt Brush* (voir figure 2.10a), une application de dessin immersive dans laquelle l'utilisateur trace des courbes en trois dimensions.

---

1. <https://developer.microsoft.com/en-us/windows/kinect>

2. <https://www.leapmotion.com/>

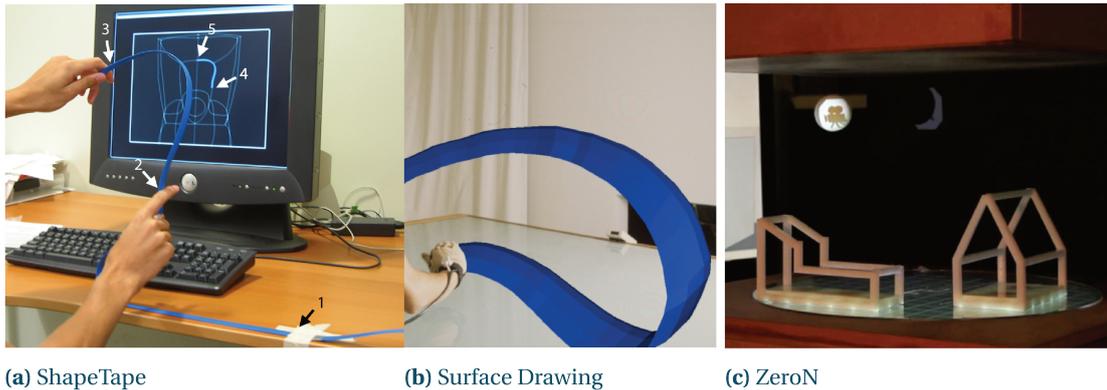
3. <https://www.leapmotion.com/>

4. <https://www.vive.com/fr/>

5. <https://www.oculus.com/rift/>

### 2.3.5 Interaction tangible

L'interaction tangible [Ishii et Ullmer, 1997] consiste à interagir avec un système numérique grâce à des objets physiques fortement couplés aux données numériques manipulées. Il existe relativement peu de systèmes pour le tracé de courbes utilisant des dispositifs tangibles. Nous présentons *ShapeTape* ou *Surface Drawing* qui ont été spécifiquement conçus pour le tracé de courbes ainsi que d'autres outils non dédiés à cet usage mais qui, bien qu'ils ne soient pas utilisés pour le tracé de courbe, nous semblent intéressants dans ce contexte.



**FIGURE 2.11** – À gauche, *ShapeTape* [Balakrishnan et al., 1999b] permet de tracer des courbes à l'aide d'un ruban magnétique. Au centre, *surface Drawing* [Schkolne et al., 2001] permet de dessiner des objets 3D et de les manipuler avec des outils tangibles. À droite, *ZeroN* [Lee et al., 2011] est un système basé sur un aimant lévitant au sein d'un champ magnétique.

*ShapeTape* [Balakrishnan et al., 1999b, Grossman et al., 2003], illustré figure 2.11a, est un ruban physique déformable permettant la modélisation de courbes 3D, la construction de surfaces, l'extrusion et la construction d'objets de révolution. Celui-ci est composé de deux capteurs de torsion pour fibres optiques fixés sur le ruban. Ces capteurs permettent d'obtenir à chaque position, le degré de courbure du ruban et sa torsion, ce qui permet de reconstruire la forme exacte du ruban de manière virtuelle. Plusieurs itérations de ce ruban ont été réalisées. L'une d'elles est accompagnée d'une double pédale et d'un palet disposés au sol pour entrer dans un mode de contrôle du point de vue.

*Surface Drawing* [Schkolne et al., 2001] (figure 2.11b) est un système permettant de tracer des objets 3D en faisant des gestes en l'air. L'utilisateur peut tracer des surfaces 3D en utilisant la main entière ou tracer des courbes 3D en utilisant l'index. Le mouvement de la main est suivi grâce à un gant de type *CyberGlove<sup>TM</sup>*<sup>1</sup>. L'utilisateur dispose d'un certain nombre d'outils physiques. Un aimant permet de déformer la courbe ou la surface tracée de manière ponctuelle. Une pince permet de déplacer l'objet 3D dessiné ou changer sa taille. Enfin, une gomme permet d'effacer des portions de courbes ou de surfaces.

Finalement, *ZeroN* [Lee et al., 2011] est un système avec un aimant lévitant dans un champ magnétique. La position de l'aimant est capturée par un algorithme de vision par ordinateur et le système peut déplacer l'aimant en modifiant le champ magnétique. Le dispositif possède également un projecteur permettant d'afficher de l'information sur ou autour de l'aimant. Bien que ce dispositif ne soit pas utilisé pour le tracé de courbe, une des applications présentées est le tracé de trajectoire dans une vue 3D.

1. <http://www.cyberglovesystems.com/>

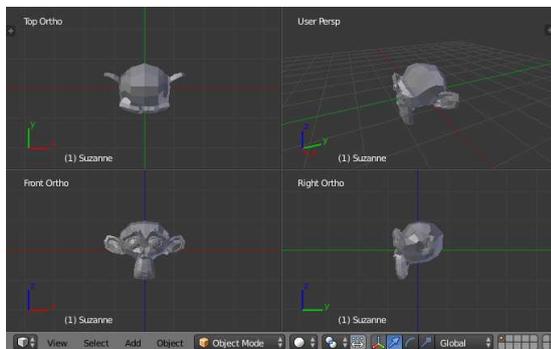
Nous avons présenté un ensemble de dispositifs d'entrée pour le tracé de courbes 3D. Il existe également un certain nombre de dispositifs de sortie permettant au système de fournir un retour sensoriel à l'utilisateur au cours de son interaction. Ces travaux sont présentés dans la section suivante.

## 2.4 DISPOSITIFS DE SORTIE ET PERCEPTION

Il est capital pour l'utilisateur d'obtenir un retour sensoriel lorsqu'il effectue une action avec un dispositif d'entrée, afin de comprendre l'effet de celle-ci sur le système. Dans cette section nous présentons les dispositifs de sortie existants. Nous distinguons les écrans traditionnels, les écrans 3D et les dispositifs offrant du retour haptique.

### 2.4.1 Écrans traditionnels

Les logiciels de modélisation 3D ainsi que beaucoup de systèmes décrits dans le domaine académique s'appuient sur des écrans de bureau traditionnels. Cependant, ces écrans sont assez mal adaptés à la visualisation de contenus 3D. En effet, une visualisation en deux dimensions ne permet pas aux utilisateurs de comprendre facilement certaines relations entre les objets d'intérêt (tailles, distances, etc.) d'une scène 3D. De plus certains objets peuvent s'occulter. Il est courant, dans les logiciels de modélisation, d'utiliser 4 vues distinctes (figure 2.12a) représentant les projections orthogonales de la scène selon les 3 axes, ainsi qu'une vue en perspective de la scène. D'autres systèmes [Bae et al., 2004, Grossman et al., 2001] utilisent de larges écrans n'affichant qu'une unique vue pour favoriser l'immersion.



(a) Projections dans *Blender*



(b) *Tangible NURBS*

**FIGURE 2.12** – À gauche, les trois projections orthogonales et la vue en perspective dans *Blender*. À droite, *Tangible NURBS* [Bae et al., 2004] permet le tracé de courbes sur grand écran à l'aide de poignées tangibles.

### 2.4.2 Écrans stéréoscopiques

Les écrans stéréoscopiques permettent d'avoir une meilleure perception de la profondeur dans la scène 3D, ce qui facilite la manipulation des objets. L'utilisateur aura ainsi l'impression qu'un objet virtuel se trouve hors de l'écran, à une certaine distance de lui-même. *Toucheo* [Hachet et al., 2011] (figure 2.13a) par exemple, utilise un écran stéréoscopique placé en haut du dispositif permettant de générer une image qui est ensuite réfléchiée sur une surface intermédiaire, via un miroir à mi-chemin de la surface

d'interaction. L'image est formée au-dessus de la surface d'interaction et donne une hauteur à l'objet dans l'espace physique. Afin de fournir une meilleure sensation d'immersion dans la scène 3D, il est également possible de coupler la perspective de la scène 3D avec la position de la tête de l'utilisateur. Ainsi, si l'utilisateur se penche sur la gauche ou sur la droite, le système recalculera la perspective de la scène 3D en conséquence. Cet indice visuel est intuitif pour l'utilisateur puisque qu'il simule un comportement déjà connu dans le monde physique. Plusieurs systèmes (*Mockup Builder*, *Toucheo*, etc.) s'appuient sur ce principe.

Il existe aussi d'autres systèmes comme *Eraser Pen* [De Amicis et al., 2002] ou *Responsive Workbench* [Krueger et Froehlich, 1994] qui emploient également des lunettes. Celles-ci ont pour inconvénient de forcer l'utilisateur à porter un équipement additionnel.



(a) Toucheo

(b) Drawing on Air

**FIGURE 2.13** – À gauche, *Toucheo* [Hachet et al., 2011] permet de manipuler des objets 3D en offrant une meilleure perception de la profondeur. À droite, *Drawing on Air* [Keefe et al., 2007] permet de tracer des courbes en 3D à l'aide d'un bras à retour haptique.

### 2.4.3 Retour haptique

Finalement, quelques systèmes renforcent la perception de l'utilisateur grâce à une modalité sensorielle complémentaire. C'est le cas de *Drawing on Air* [Keefe et al., 2007] (figure 2.13b) qui fournit du retour haptique à l'utilisateur via un bras articulé à retour de force *Phantom*, qui sert aussi de dispositif d'entrée. Avec *SATIN* Bordegoni et al. [2011] présentent un système permettant de simuler les caractéristiques d'une courbe 3D en couplant une bande déformable haptique déformée par plusieurs servomoteurs à un dispositif de vision stéréo afin de visualiser celle-ci.

Après avoir présenté les principaux types de dispositifs de sortie existants, et leur apport à la perception utilisateur, nous présentons, dans la section suivante, le langage d'interaction associé au tracé de courbes.

## 2.5 LANGAGE D'INTERACTION

Dans cette section, nous présentons le second axe de notre espace de conception, intitulé langage d'interaction. Celui-ci vise à présenter les techniques d'interaction existantes pour le tracé et l'édition

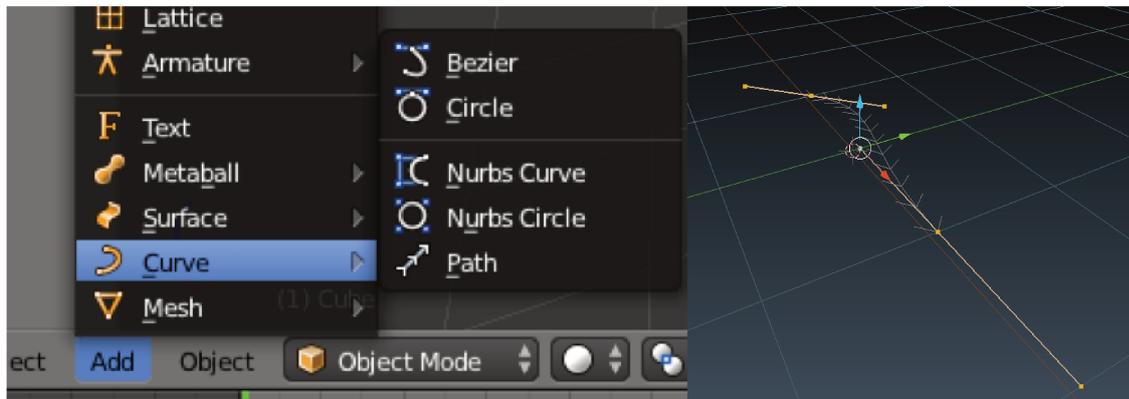
de courbes avec les dispositifs présentés précédemment. Cet axe est organisé en deux sous-sections, tracé et édition, reflétant les deux types de tâches de l'édition de courbe.

### 2.5.1 Tracé de courbes 3D

Dans les logiciels 3D, la stratégie la plus courante consiste à proposer un ensemble de courbes types (figure 2.14a) que l'utilisateur peut choisir à l'aide d'un menu. D'autres stratégies ont également été proposées. La première consiste à créer la courbe en deux temps, de manière séquentielle (figure 2.15a). La seconde nécessite que l'utilisateur définisse des contraintes (figure 2.15c). Enfin, la dernière stratégie consiste à laisser le système inférer les contraintes automatiquement. Nous présentons maintenant ces stratégies de manière plus détaillée.

#### Approche typique dans les logiciels 3D

Dans les logiciels de modélisation 3D, le tracé de courbes se fait à travers une tâche de sélection dans une liste ou un menu. Cette stratégie est efficace si le nombre de choix possibles est limité. Par exemple, dans *Blender* (figure 2.14a), l'utilisateur clique sur le menu permettant d'ajouter des formes 3D dans la scène. Il sélectionne ensuite le sous-menu "Courbes" et choisit le type de courbe voulu dans le menu. En cliquant, la courbe est automatiquement ajoutée à la scène 3D.



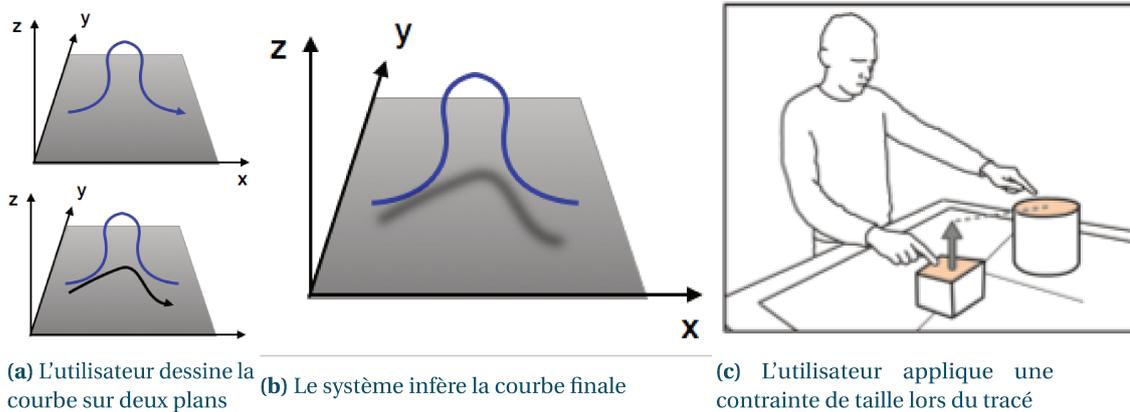
(a) Blender : choix de courbes

(b) La courbe est ajoutée à la scène

**FIGURE 2.14** – À gauche, *Blender* offre à l'utilisateur la possibilité de créer plusieurs types de courbes à travers un menu. À droite, après sélection dans le menu, la courbe est ajoutée à la scène 3D.

#### Travaux académiques

**Approche séquentielle.** Avec l'approche séquentielle, l'utilisateur crée une courbe en deux temps (figure 2.15a). Dans un premier temps, il trace la courbe en 2D sur l'un des plans de projection. Il fait de même sur un second plan, ce qui permet d'inférer la position des points de la courbe. Certains systèmes ont proposé d'utiliser l'ombre portée [Cohen et al., 1999, Zeleznik et al., 1996] de la courbe lors du second tracé. L'utilisateur dessine d'abord la courbe voulue sur le plan X-Y. Il dessine ensuite l'ombre portée de la courbe sur le plan X-Z. Le système infère la profondeur de la courbe à partir de ces deux tracés et crée une courbe 3D.



**FIGURE 2.15** – À gauche, le tracé de la courbe 3D se fait en plusieurs étapes. L'utilisateur commence par dessiner celle-ci dans le plan  $zx$ , puis il dessine son ombre portée sur le sol (plan  $xy$ ). Au centre, le système infère la courbe finale. À droite, dans *Mockup Builder* [De Araùjo et al., 2012], l'utilisateur définit une contrainte de taille sur la forme en cours de création.

**Définition de contraintes.** L'utilisateur peut définir des contraintes (taille, hauteur, bornes, voir figure 2.15c) afin de réaliser une courbe 3D à partir d'un dispositif 2D. Les contraintes peuvent être définies avant le tracé de la courbe. Par exemple, dans *3-Draw* [Sachs et al., 1991], l'utilisateur peut préciser à l'avance le point de départ et le point d'arrivée de la future courbe. Dans *Terrain Sketching* [Gain et al., 2009], l'utilisateur définit d'abord un volume afin de borner la partie du terrain à remodeler. La définition de contraintes peut également être réalisée en même temps que le tracé grâce à de l'interaction bimanuelle ou tangible : par exemple, l'utilisateur peut définir un plan de symétrie avec la main gauche dans *I Love Sketch* ([Bae et al., 2008]) ou fixer une taille maximale relativement à un autre objet de la scène (figure 2.15c) dans *Mockup Builder* [De Araùjo et al., 2012] en pointant vers celui-ci. Dans *ShapeTape* [Balakrishnan et al., 1999b, Grossman et al., 2003], l'utilisateur contraint le tracé de la courbe à l'aide du ruban physique.

Ces approches sont utiles pour pallier les limitations des dispositifs 2D. Elles peuvent également servir à augmenter la précision du tracé avec des dispositifs 3D.

### Approche dans les logiciels de son 3D

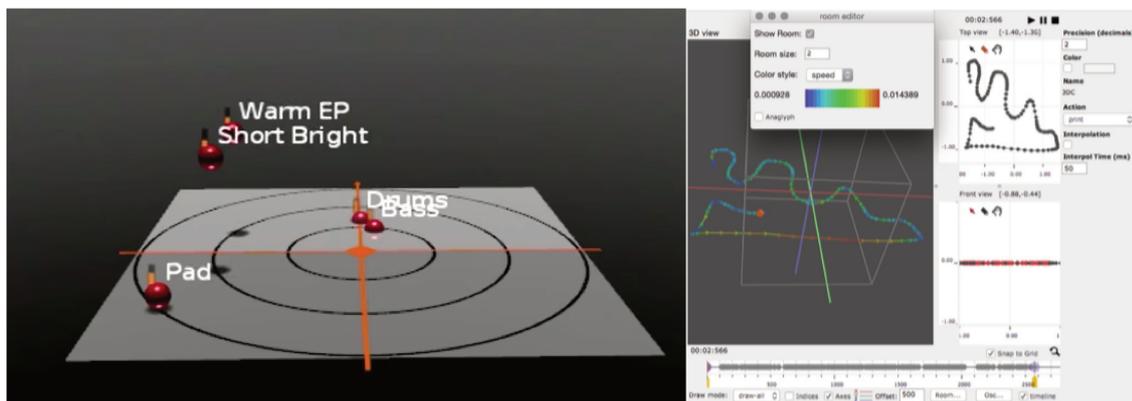
Plusieurs stratégies sont adoptées pour la création de trajectoires dans les logiciels de son 3D.

**Script.** Une première stratégie passe par l'implémentation d'un algorithme ou d'un script comme dans *OM-Prisma* [Schumacher et Bresson, 2010], *Holo-Edit* [Pottier, 1998] et *IanniX* [Jacquemin et al., 2012]

**Trajectoires prédéfinies.** Une stratégie assez courante est d'utiliser des trajectoires pré-définies (primitives) telles que des cercles, spirales ou LFO (Low-Frequency Oscillation) pouvant être couplées à des fonctions de transformation telles que des opérations de symétrie ou de translation/rotation. On retrouve cette approche dans *Holo-Edit* [Pottier, 1998] ou *Hyperspace* [Parmerud, 2010].

**Positionnement interactif.** Une autre stratégie consiste à positionner interactivement les sources sonores dans l'espace à l'aide d'une visualisation 3D de la scène couplée à une vue en plongée comme dans *Holo-Edit* [Pottier, 1998]. On retrouve cette stratégie dans le logiciel *Longcat H3D*<sup>1</sup> (figure 2.16a) qui représente l'ensemble des sources audio sous forme de sphères et permet à l'utilisateur de placer celles-ci dans l'espace. Un ombrage est utilisé afin de représenter la hauteur de la source par rapport au plan de la scène représentant le sol.

**Tracé à la souris.** Hyperspace [Parmerud, 2010], *OM-Prisma* [Schumacher et Bresson, 2010] et *IanniX* [Jacquemin et al., 2012] permettent de tracer des fonctions continues à la souris représentant les trajectoires dans un espace bidimensionnel. Cette approche est également utilisée dans *Spat-scene* [Garcia et al., 2017], un système de composition pour le son 3D qui permet de tracer des trajectoires 3D à partir de plusieurs vues 2D : une vue de face et une vue de dessus. Il est également possible d'associer une fonction définissant l'élévation dans le temps afin d'obtenir une trajectoire tridimensionnelle.



(a) *Longcat H3D*

(b) *Spat-scene* [Garcia et al., 2017]

**FIGURE 2.16** – À gauche, *Longcat* représente la position des sources sonores sous forme de sphères que l'utilisateur peut déplacer avec la souris. L'utilisation d'une ombre portée sur un plan permet de donner une meilleure appréciation de la hauteur de la source audio par rapport au sol. À droite, *Spat-scene* [Garcia et al., 2017] permet de tracer des trajectoires 3D à partir de plusieurs plans 2D (vue de face et de dessus).

### 2.5.2 Édition de courbes 3D

Nous distinguons trois sous-tâches constituant l'édition de courbes 3D : la sélection, le remodelage et la suppression. Dans les sections suivantes, nous présentons les approches existant dans les logiciels 3D et dans le domaine académique.

#### Approche dans les logiciels 3D

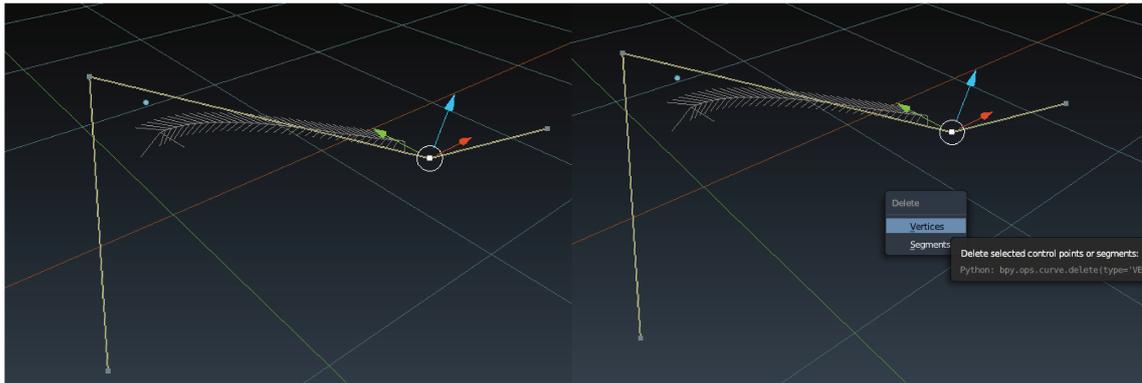
**Sélection.** La sélection se fait généralement à l'aide de la souris. La phase de sélection permet à l'utilisateur de préciser l'ensemble des objets à manipuler lors de la prochaine action.

**Remodelage.** Le remodelage permet à l'utilisateur de retravailler une section de courbe afin de l'affiner. Pour les courbes de type polyligne qui sont représentées par un ensemble de segments, l'approche

1. <http://www.longcat.fr/>

classique consiste à sélectionner un ensemble de points de la courbe et à les déplacer (figure 2.17a). L'approche est différente pour les courbes de type spline (NURBS, Bézier, etc.). Le remodelage se fait alors à travers de points de contrôle qui ne sont pas directement positionnés sur la courbe.

**Suppression.** L'approche classique pour supprimer une section de courbes est de sélectionner un ensemble de points et de les supprimer, ou de supprimer directement l'objet 3D représentant la courbe (figure 2.17b).



(a) Édition d'une courbe de Bézier dans *Blender*

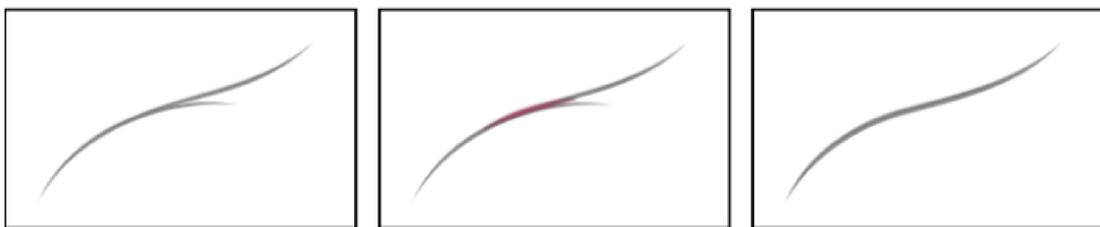
(b) Suppression d'un point de contrôle

**FIGURE 2.17** – À gauche, sélection d'un point de contrôle d'une courbe de Bézier dans *Blender*. Une poignée représentant les trois axes de translation permet de déplacer le point dans l'espace. En déplaçant le point, l'apparence de la courbe est modifiée en temps réel. À gauche, *Blender* permet d'effacer l'un des points de contrôle de la courbe ou un segment de celle-ci.

### Travaux académiques

**Sélection.** Certains systèmes permettent une sélection implicite pour certaines tâches. Comme nous allons le voir dans la section suivante, le système peut alors inférer l'intention de l'utilisateur pour le remodelage de courbes, par exemple avec l'*oversketching*.

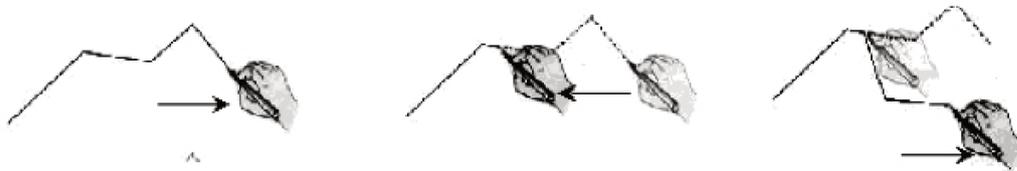
**Remodelage.** L'approche classique est bien adaptée à la CAO, mais pose souvent problème aux graphistes, car leur représentation mentale ne correspond pas forcément au modèle conceptuel de cette technique.



**FIGURE 2.18** – En dessinant par dessus une courbe existante, l'utilisateur remodèle celle-ci (*oversketching* [Baudel, 1994]). Cette technique a été utilisée avec succès dans *I Love Sketch* [Bae et al., 2008].

En s'appuyant sur les habitudes des dessinateurs professionnels, Baudel [1994] a proposé la notion d'*oversketching* (figure 2.18). L'utilisateur repasse sur son trait plusieurs fois créant ainsi plusieurs tracés. Ceux-ci sont alors fusionnés pour produire une nouvelle courbe qui "moyenne" les précédentes.

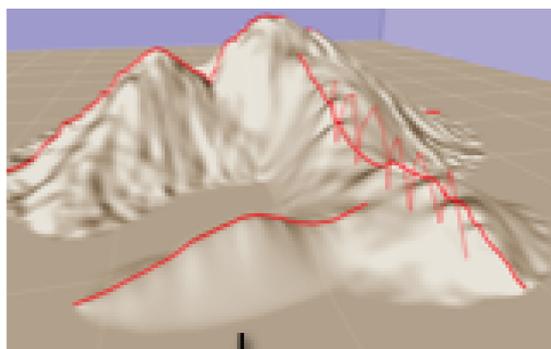
Cette technique a ensuite été reprise dans plusieurs travaux [Bae et al., 2008, Gain et al., 2009, Igarashi et al., 1998, 1999, Zeleznik et al., 1996]. Enfin, une dernière approche [De Amicis et al., 2002] permet à l'utilisateur de remodeler une section de courbe précédemment tracée en repassant sur son tracé dans le sens opposé (figure 2.19). La section de courbe sur laquelle l'utilisateur est revenu est alors effacée.



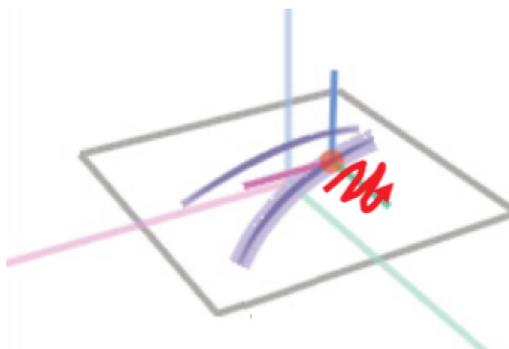
**FIGURE 2.19** – L'approche pour remodeler une courbe utilisée dans *3D Eraser Pen* est de revenir en arrière sur le segment de courbe tracé précédemment et de repartir dans la direction souhaitée. La portion de courbe sur laquelle l'utilisateur est revenu est ainsi retirée.

On notera également le cas du lissage, un remodelage passif consistant à supprimer le bruit du dispositif d'entrée ou celui lié à l'inexpérience de l'utilisateur. De nombreux outils de dessin [Bae et al., 2008, De Araùjo et al., 2012, Thiel et al., 2011] utilisent cette fonctionnalité afin de rendre les courbes plus harmonieuses.

**Suppression.** Il existe plusieurs stratégies permettant d'effacer une section de courbe. L'une d'elles consiste à rayer la partie de courbe à retirer en dessinant par dessus celle-ci (figure 2.20). On peut retrouver cette technique dans *3-Draw* [Sachs et al., 1991], *I Love Sketch* [Bae et al., 2008], *Terrain Sketching* [Gain et al., 2009] et *Mockup Builder* [De Araùjo et al., 2012]. Comme présenté précédemment, une seconde stratégie consiste à sélectionner un sous-ensemble de points contigus de la courbe et à supprimer ceux-ci.



(a) *Terrain Sketching* [Gain et al., 2009]



(b) *I Love Sketch* [Bae et al., 2008]

**FIGURE 2.20** – Il est possible de supprimer une section de courbe en rayant celle-ci. Cette approche est ici utilisée dans *Terrain Sketching* [Gain et al., 2009] et *I Love Sketch* [Bae et al., 2008].

### Approche dans les logiciels de son 3D

La plupart des logiciels de son 3D offrent des méthodes de lissage. *Holo-Edit* [Pottier, 1998] propose également un ensemble de transformations spatiales (expansion, rotation, symétrie, translation, etc.) et temporelles (étirement, ré-échantillonnage, inversion, etc.), pour modifier une trajectoire dans son intégralité, ou partiellement. Il est possible d'exécuter un algorithme ou un script préalablement défini, de changer le sens de déplacement de la source sonore, ainsi que d'autres effets. Enfin, *Hyperspace*

[Parmerud, 2010] propose un mode de contrôle supplémentaire, qui permet de naviguer spatialement entre deux positions tridimensionnelles, préalablement définies via un curseur de défilement.

## 2.6 CONCLUSION

Dans ce chapitre nous avons présenté dans un premier temps les besoins des acteurs du projet, à savoir comment visualiser une grande quantité d'informations (groupes et trajectoires) et tracer et éditer des courbes 3D. Ce dernier point a motivé la création d'un espace de conception qui nous a permis de structurer les travaux académiques et industriels existants dans le domaine du tracé et de l'édition de courbe. Nous avons présenté un espace de conception basé sur le couple <dispositif, langage d'interaction>. Nous avons vu qu'il existait plusieurs types de dispositifs de sortie, notamment des écrans stéréoscopiques permettant d'améliorer la perception de profondeur de l'utilisateur face à une scène 3D.

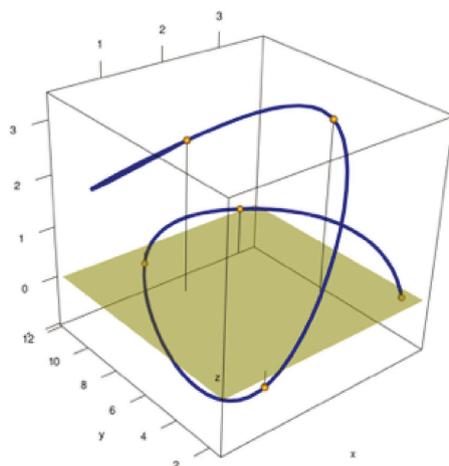
Cependant, les ingénieurs du son utilisent quasi exclusivement des écrans classiques pour éditer leurs enregistrements et éditer les trajectoires des courbes sonores. Tracer ou éditer une courbe dans l'espace n'est pas trivial sur ce type d'écran et nécessite de pouvoir contrôler le point de vue afin d'examiner le tracé de la courbe sous plusieurs angles. Ceci nous a amené à concevoir et développer une technique permettant à l'utilisateur de contrôler le point de vue de manière plus confortable et intuitive. Dans cette optique, nous donnons dans un premier temps, dans le chapitre suivant, un aperçu des indices visuels permettant à l'être humain de mieux appréhender son environnement, puis nous nous intéressons aux techniques existantes de contrôle de point de vue.



## CONTRÔLE DU POINT DE VUE

La visualisation de contenu tridimensionnel sur des écrans classiques (2D) souffre à la fois de la perte de la dimension de profondeur due à la projection de contenu tridimensionnel vers un dispositif d'affichage bidimensionnel et de l'absence de parallaxe de mouvement (voir section 3.1.1), à savoir le déplacement apparent des objets lors du déplacement de la tête, qui joue un rôle prépondérant dans la compréhension de notre environnement. En effet, les interfaces graphiques ainsi que les dispositifs de pointage que nous utilisons au quotidien sont adaptés à nos écrans, traditionnellement en deux dimensions. Il est par conséquent difficile de correctement appréhender une scène 3D avec les dispositifs classiques en l'observant à partir d'un unique point de vue. Il est nécessaire pour l'utilisateur, à l'instar du monde réel, de pouvoir se déplacer dans la scène afin de comprendre les relations entre les objets qu'elle contient. Prenons l'exemple de la figure 3.1, en l'observant sur un écran 2D on ne peut pas décrire de manière sûre la topologie de la courbe 3D. La courbe passe-t-elle au travers du plan horizontal ? Devant l'arête du cube ? À partir de ce point de vue, les courbes semblent se croiser, mais il est impossible de dire quelle section de courbe passe devant l'autre. L'utilisation d'un écran stéréoscopique pourrait contribuer à désambiguïser la scène, mais l'approche la plus efficace reste d'exploiter la parallaxe de mouvement ([Ware et al., 1993]).

Afin de comprendre comment améliorer la perception de profondeur dans le monde virtuel et contrôler le point de vue dans une scène 3D, il est nécessaire d'expliquer comment celle-ci est perçue dans le monde réel. Dans ce chapitre, nous commençons par donner un aperçu des indices visuels permettant



**FIGURE 3.1** – D'un unique point de vue, la position de la courbe est ambiguë. La courbe passe-t-elle au travers du plan horizontal ? Devant ou derrière l'arête du cube ?

de percevoir la profondeur. Nous présentons ensuite les contrôles de caméra élémentaires (contrôle du point de vue), puis nous décrivons comment ceux-ci sont utilisés dans les logiciels 3D. Le terme *logiciels 3D* est ici utilisé pour représenter tout logiciel utilisant une visualisation en trois dimensions (logiciels de modélisation, jeux en 3D, mappemonde en 3D, etc.). Par la suite, nous décrivons les travaux de recherche liés au contrôle du point de vue et les modalités d'interaction permettant de les contrôler.

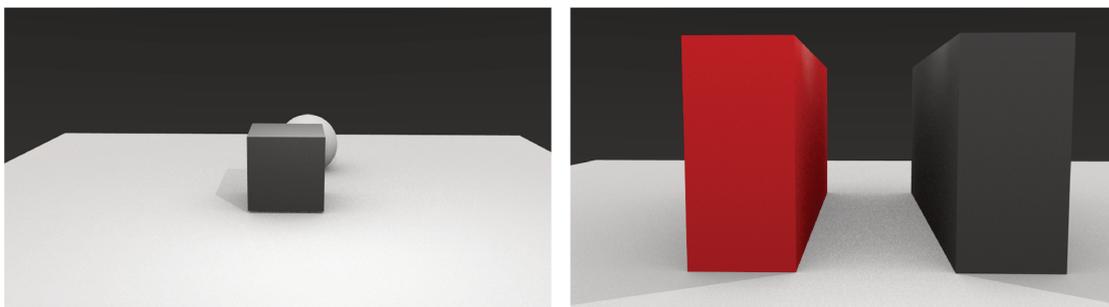
### 3.1 INDICES VISUELS

Dans le monde physique, l'utilisateur dispose de plusieurs indices visuels lui permettant de se repérer dans son environnement et d'inférer la position des objets physiques autour de lui. Selon Ware [2012], il existe deux catégories d'indices visuels, les indices monoculaires et les indices binoculaires.

#### 3.1.1 Indices monoculaires

Les indices monoculaires permettent de percevoir la profondeur à partir d'un seul oeil.

**Occultation.** L'occultation est l'un des indices visuels monoculaires les plus importants. Elle intervient quand plusieurs objets se recouvrent dans un environnement en trois dimensions. Si un objet en recouvre un autre, c'est donc que celui-ci est au premier plan. Dans la figure 3.2a, la sphère blanche est occultée par le cube noir au premier plan. La sphère blanche se trouve donc derrière le cube noir.



(a) Occultation

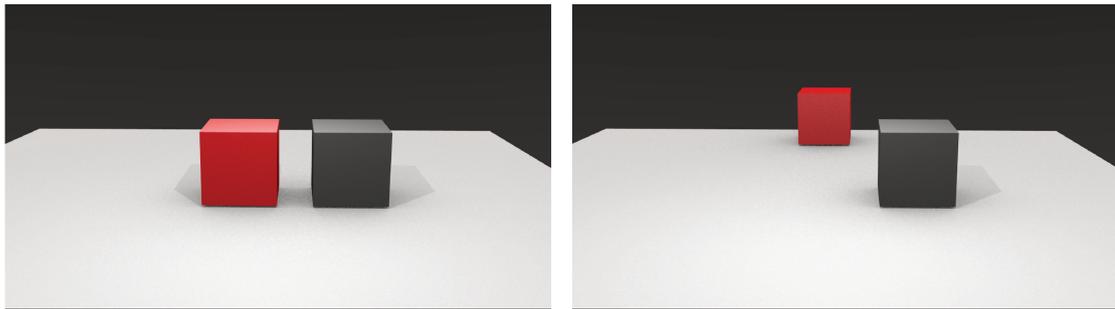
(b) Perspective linéaire

**FIGURE 3.2** – À la figure 3.2a, la sphère blanche est occultée par le cube noir au premier plan. Dans la figure 3.2b, deux parallélépipèdes positionnés parallèlement semblent converger l'un vers l'autre dans la scène 3D.

**Perspective linéaire.** Deux objets positionnés de manière parallèle dans la scène 3D vont sembler converger avec la profondeur (voir figure 3.2b) et réduire de taille.

**Taille relative.** La taille relative d'un objet par rapport à un autre permet, en connaissant la taille des deux objets, d'estimer la position de ces deux objets dans un espace 3D. En effet, si les deux objets sont de tailles identiques (figure 3.3a) et que le premier objet apparaît plus petit que le second (figure 3.3b), cela signifie qu'il est plus éloigné du point de vue que le second objet.

**Hauteur relative.** Selon un principe identique, si l'on connaît la taille de deux objets, on peut à partir de la hauteur de ces objets déterminer la distance qui les sépare. Prenons l'exemple de la figure 3.3b.



(a) Deux cubes de tailles identiques sont placés à la même hauteur. (b) En éloignant un des cubes, sa taille semble diminuer avec la distance.

**FIGURE 3.3** – En connaissant la taille des deux cubes dans la première figure, il est possible de déduire la profondeur à laquelle se situe le cube de droite dans la seconde figure.

Dans cette figure, les deux cubes ont une taille identique, cependant le cube rouge paraît plus haut dans la scène. Il est donc situé plus loin du point de vue que le cube noir.

**Parallaxe de mouvement.** La parallaxe de mouvement est le déplacement apparent d'objets fixes lors du déplacement d'un observateur (voir figure 3.4). Plus un objet est proche de l'observateur (cubes rouges ou noirs), plus le mouvement de celui-ci sera large. Parallèlement, plus cet objet est éloigné de l'observateur et plus son déplacement sera faible (sphère blanche) [Ware, 2012]. Cet indice visuel est très utile pour estimer la distance entre un objet et l'utilisateur, mais aussi pour estimer la distance entre plusieurs objets proches.



(a) Vue centrale

(b) Vue en se déplaçant sur la gauche

**FIGURE 3.4** – À gauche, le cube rouge masque la boule grise. À droite, la caméra s'est déplacée sur la droite, révélant la boule grise. Tous les éléments de la scène se sont déplacés vers la gauche : les éléments au premier plan semblent plus loin du centre que le cube rouge, qui n'a pratiquement pas bougé.

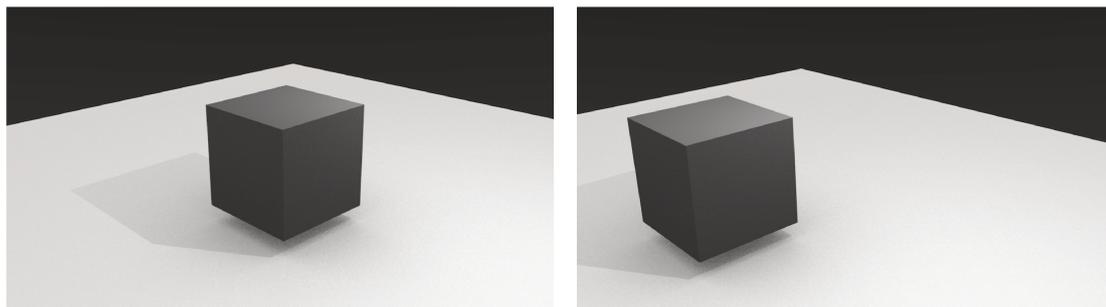
**Gradient de texture.** Le gradient de texture est un indice visuel qui intervient sur des textures uniformes, par exemple un maillage. Au fur et à mesure que la texture s'étend dans l'espace, le maillage rétrécit et finit par sembler avoir une couleur unie.

**Ombres portées.** Les ombres portées permettent d'obtenir, si l'on connaît la position de la source de lumière, une information sur la hauteur d'un objet par rapport à une surface. En fonction de la taille de l'ombre sur la surface, l'utilisateur peut inférer, à partir de la taille initiale de l'objet, la hauteur de celui-ci par rapport au plan sur lequel l'ombre est projetée.

### 3.1.2 Indices binoculaires

Les indices binoculaires sont les indices visuels nécessitant les deux yeux.

**Disparité binoculaire.** La disparité binoculaire est le décalage du point de vue entre les deux yeux. Celle-ci est due à la distance interoculaire, qui est en moyenne de 6.4 cm et qui permet d'obtenir deux vues différentes pour un même point (voir figure 3.5).



(a) Vue de l'oeil directeur

(b) Vue de l'oeil non directeur

**FIGURE 3.5** – À gauche, la vue de l'oeil directeur, à droite celle de l'autre oeil. Cette disparité binoculaire est due à la distance interoculaire.

Lorsque l'utilisateur utilise un logiciel 3D, il est face à un écran (dispositif de sortie 2D). La scène 3D est projetée sur le plan de la caméra de point de vue et apparaît donc en 2D sur l'écran de l'utilisateur, que l'on peut assimiler à la rétine de celui-ci dans le monde physique. Cependant, dans le cas de l'écran, l'utilisateur ne dispose pas de perspective couplée à la tête (parallaxe de mouvement), à savoir l'adaptation de la perspective de la scène 3D par rapport à la position de la tête de l'utilisateur, un des indices monoscopiques visuels les plus importants [van Schooten et al., 2010]. Dans la suite de ce chapitre, nous présentons les différents contrôles de point de vue permettant de désambiguïser la scène. D'abord, nous nous sommes intéressés aux contrôles de caméra existants dans le but de mieux comprendre comment ceux-ci peuvent être utilisés afin d'augmenter la perception de profondeur de l'utilisateur et lui permettre de contrôler son point de vue de manière efficace dans une scène 3D.

## 3.2 CONTRÔLES DE CAMÉRA

Le contrôle du point de vue se fait habituellement à l'aide de sept paramètres [Marchal et al., 2013] : 3 degrés de rotation, 3 degrés de translation et la distance focale de la caméra. À partir de ces paramètres, plusieurs contrôles de caméra [Christie et al., 2008] sont communément utilisés pour contrôler le point de vue dans un espace en trois dimensions.

**Translation.** Le mouvement de translation (figure 3.6a) permet de déplacer la caméra dans la scène 3D selon un axe, un plan ou bien un vecteur 3D donné.

**Zoom.** Le zoom (figure 3.6b) correspond à une manipulation de la distance focale de la caméra de point de vue. En augmentant la distance focale, la projection semble augmenter d'échelle, tandis qu'à l'inverse si la distance focale diminue, la projection rétrécit.

**Rotation.** On distingue deux types de rotations en fonction de la position du centre de rotation :

- La rotation égo-centrique (figure 3.6c), le centre de rotation de la caméra est la caméra elle-même. Cette vue est adaptée pour représenter par exemple, le point de vue d'un humain, dont la tête peut tourner de droite à gauche pour changer de point de vue ;
- la rotation exocentrique (figure 3.6d), la caméra orbite autour d'un centre de rotation qui n'est pas celui de la caméra. Si la caméra reste toujours pointée sur ce point, alors la vue orbite autour de ce point.

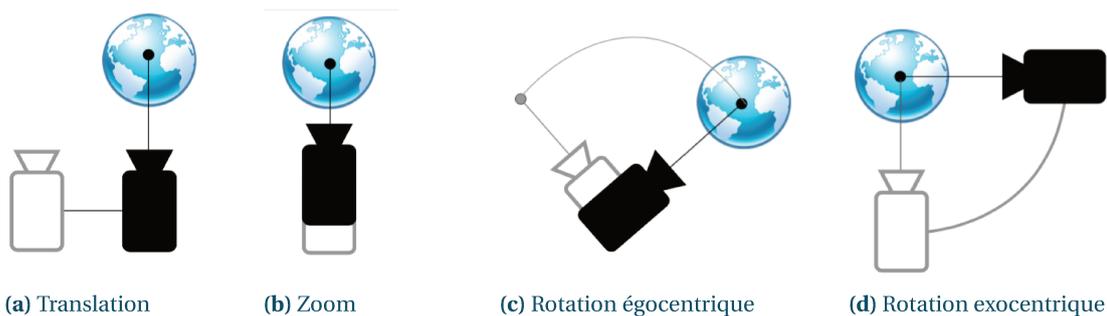


FIGURE 3.6 – Quatre contrôles de caméra différents

Nous étudions maintenant comment les logiciels 3D tels que des logiciels de modélisation (*Blender*, *Maya*), de visualisation de contenu 3D (*Meshlab*), d'exploration (*Google Earth*) ou bien même les jeux (*Age of Empire III*, *Portal*) donnent accès à ces contrôles de caméra.

### 3.3 CONTRÔLES DE CAMÉRA UTILISÉS DANS LES LOGICIELS 3D

La plupart des contrôles de caméra sont effectués à l'aide du clavier et de la souris et nécessitent des modes spécifiques, ce qui ne permet pas de faire une utilisation concurrentielle de plusieurs contrôles de caméra.

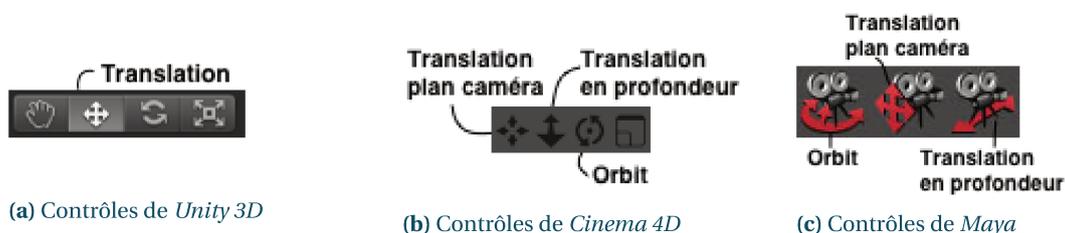
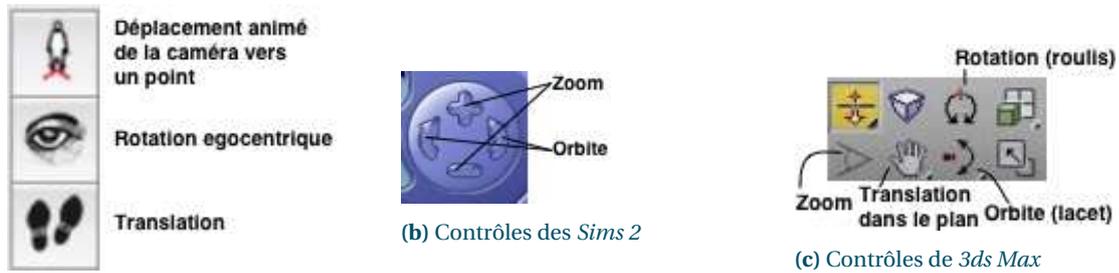
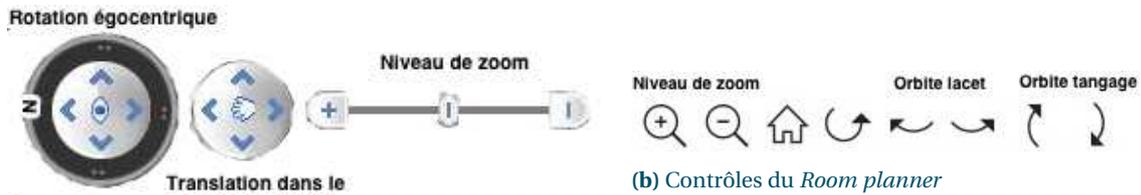


FIGURE 3.7 – À gauche, la barre d'outils de *Unity*. Le bouton avec la main représente la manipulation de la caméra et permet de réaliser de la translation dans la scène. Au centre les différents contrôles de caméra de *Cinema 4D*, incluant translation dans le plan et rotation. À droite les contrôles de caméra utilisés dans *Maya* : rotation exocentrique, translation haut/bas et droite/gauche, translation dans l'axe de la profondeur/zoom



(a) Contrôles de Sketchup

**FIGURE 3.8** – À gauche, l’interface permettant de contrôler les contrôles de caméra dans Sketchup. Au centre, les contrôles de zoom et de rotation exocentrique sont supportés dans les Sims 2. À droite, les contrôles de caméra utilisés dans 3ds Max : translation de la caméra dans le plan de la profondeur, zoom, roulis de la caméra, translation de la caméra dans le plan de la caméra, rotation exocentrique



(a) Contrôles de Google Earth

**FIGURE 3.9** – À gauche, l’interface de contrôle de Google Earth. Le slider, permet de contrôler le niveau de zoom en déplaçant le curseur. Le zoom peut également être contrôlé avec les boutons +/- . Le cercle central avec les flèches directionnelles permet de contrôler la rotation égocentrique de la caméra sur les axes de lacet et de tangage. À droite, le Room Planner d’Ikea supporte les mouvements de zoom et de rotation exocentrique sur les axes de lacet et de tangage.

### 3.3.1 Translation

#### Contexte d’utilisation

Dans les logiciels de modélisation 3D, la translation de la caméra se fait essentiellement dans le plan de l’écran sur les axes horizontal et vertical. Ce type de mouvement est généralement utilisé en alternance avec le contrôle du zoom afin de recentrer la vue 3D avant ou après une manipulation.

#### Modalités d’interaction

Le contrôle se fait en sélectionnant l’outil de translation dans la barre d’outils puis en effectuant un déplacement horizontal de la souris dans la vue tout en maintenant le bouton appuyé pour déplacer le point de vue horizontalement ou dans la profondeur pour déplacer verticalement. Dans Unity 3D (figure 3.7a) les touches **A** et **D** sont également utilisées pour le déplacement horizontal et les touches **Q** et **E** pour le déplacement vertical. L’utilisateur peut également contrôler la translation en appuyant sur un bouton dédié dans la barre d’outils (figure 3.7b) et en déplaçant la souris tout en maintenant son bouton appuyé (pseudo-mode). Dans les jeux de stratégie, la translation se fait généralement au clavier à l’aide des touches directionnelles. Enfin, certains jeux de stratégies (*Age of Empire, 0 A.D.*) effectuent une translation dans le plan transverse lorsque la souris atteint les bords de l’écran.

### 3.3.2 Zoom

#### Contexte d'utilisation

On trouve deux approches différentes pour effectuer un mouvement de zoom dans une scène 3D. La plus courante consiste à déplacer la caméra sur l'axe de la profondeur. La seconde consiste à faire varier la distance focale de la caméra. Cette dernière approche est davantage utilisée pour le rendu 3D, car son fonctionnement se rapproche de celui d'un appareil photo. En revanche, cette approche n'est pas utilisée pour contrôler le point de vue, car elle fait varier le champ de vision de la caméra. Une approche mixte consistant à faire varier à la fois la focale et la distance de la caméra donne un effet très utilisé dans le monde du cinéma : le Dolly zoom.

#### Modalités d'interaction

Le zoom est généralement contrôlé avec la molette de la souris en particulier dans les logiciels de modélisation 3D (*Blender*, *Maya*, etc.) et dans les jeux de stratégies (*Age of Empire*, etc.). Certains logiciels (*Cinema 4D*, *3ds Max*, etc.) offrent également un mode spécifique, typiquement activé par un bouton de la barre d'outils. Dans ce dernier cas, le contrôle du niveau de zoom ne se fait alors généralement pas par un mouvement de glisser de la souris. Une autre approche que l'on retrouve dans les *Sims 2* (figure 3.8b) ou le *Room Planner* (figure 3.9b) d'Ikea, consiste à utiliser deux boutons, pour incrémenter ou décrémenter le niveau de zoom. *Google Earth* (figure 3.9a) utilise plusieurs approches encore différentes pour contrôler le niveau de zoom. Ainsi, l'utilisateur peut contrôler directement celui-ci à l'aide d'un slider. Il est aussi possible de double cliquer sur un point du globe, et la caméra se déplace en avant dans sa direction. Enfin, dans *Unity*, le déplacement de la caméra le long de l'axe de la profondeur se fait comme dans un jeu à la première personne (*Mirror's Edge*, *Flight Simulator*, etc.), i.e. l'utilisateur se déplace dans la scène en appuyant sur les touches directionnelles.

### 3.3.3 Rotation égocentrique

#### Contexte d'utilisation

Le contrôle de rotation égocentrique est très utilisé dans les jeux vidéo à la première personne (point de vue égocentrique), car il donne un aspect immersif au mouvement. En revanche, il est peu utilisé pour la modélisation 3D, car ce mouvement a tendance à faire sortir la scène du champ de vision et ne permet pas d'en avoir un aperçu global. On trouve aussi ce contrôle de caméra dans *Unity* et également dans *Google Earth* lorsque l'utilisateur a zoomé suffisamment sur la carte pour être en mode *StreetView*.

#### Modalités d'interaction

La rotation égocentrique est généralement contrôlée à la souris. Dans *Unity*, l'utilisateur peut tourner la caméra en maintenant le bouton de la souris appuyé tout en déplaçant celle-ci dans le sens voulu. L'approche est identique dans les jeux à la première personne, la sensation d'immersion étant recherchée. La rotation de la caméra est alors en permanence liée à la position de la souris et représente la direction du regard de l'utilisateur.

### 3.3.4 Rotation exocentrique

#### Contexte d'utilisation

La rotation exocentrique verticale ou horizontale est probablement le contrôle de caméra le plus utilisé dans les logiciels de modélisation 3D car il permet de visualiser rapidement une scène sous tous ses angles [Balakrishnan et Kurtenbach, 1999]. De ce fait, c'est le mouvement de rotation par défaut dans la plupart des logiciels de modélisation 3D. Ces logiciels permettent aussi généralement à l'utilisateur de choisir le centre de la rotation. Par exemple, dans *Cinema 4D*, il est possible de choisir le centre d'une face d'un objet comme centre de rotation. Le contrôle du diamètre de l'orbite se fait alors avec le zoom.

#### Modalités d'interaction

Dans les logiciels de modélisation 3D, le contrôle de la rotation exocentrique passe par un changement de mode à l'aide de la barre d'outils. Le contrôle se fait ensuite à la souris. Dans Blender, la rotation exocentrique se fait soit au clavier à l'aide du pavé numérique, soit avec le bouton central de la souris sans passer par un mode particulier. Dans Google Earth, ce contrôle de caméra se présente sous la forme d'un cercle avec quatre flèches directionnelles permettant de contrôler les rotations de type tangage et lacet.

### 3.3.5 Récapitulatif

La table 3.1 montre l'ensemble des contrôles élémentaires de caméra réalisables dans différents logiciels 3D, ainsi que les axes sur lesquels ces contrôles de caméra peuvent être réalisés. Nous avons regroupé dans la catégorie *Modélisation*, les logiciels de modélisation 3D tels que *Blender*, *Maya*, etc.

Logiciel	Translation	Rot. exocentrique	Rot. égocentrique	Modalités
Modélisation	tous	tous	tous	souris, clavier
SketchUp	y, z	lacet, tangage	lacet, tangage	souris, clavier
Google Earth	tous	-	tous	souris
MeshLab	tous	tous	-	souris, clavier
IKEA	x, y	lacet, tangage	-	souris
Unity	tous	-	lacet, tangage	souris, clavier
Jeux stratégie	x, z	-	lacet	souris, clavier
Jeux première personne	x, z	lacet, tangage	-	souris, clavier

TABLE 3.1 – Contrôles de caméra couramment utilisés dans les logiciels 3D

Cette section a permis de présenter les contrôles de caméra utilisés dans les logiciels 3D, leurs modalités ainsi que leurs contextes d'utilisation. Nous présentons maintenant les techniques d'interaction pour le contrôle du point de vue 3D proposées dans les papiers de recherche. Dans la section suivante, nous séparons les techniques de contrôle du point de vue en fonction de leurs modalités d'entrée (mouvements du corps, clavier/souris ...) et du type de contrôle de caméra.

### 3.4 TECHNIQUES D'INTERACTION POUR LE CONTRÔLE DU POINT DE VUE

Le contrôle du point de vue est réalisé via des techniques d'interaction permettant d'effectuer un type de contrôle de caméra particulier ou d'en combiner plusieurs. Nous avons donc organisé les techniques d'interaction en fonction des contrôles de caméra effectués : translation, zoom, rotation égocentrique, rotation exocentrique et combinaisons de plusieurs contrôles de caméra. Pour chaque contrôle, nous avons distingué les modalités utilisées : dispositif "traditionnels" (souris + clavier, surface tactile) ou dispositifs non "traditionnels". Cette catégorie regroupe les dispositifs augmentés (centrales inertielles, capteur magnétique et dispositifs sans contact) et ceux exploitant les mouvements du corps (buste, tête, pieds). Nous avons choisi de partager les dispositifs non "traditionnels" en deux sous-catégories car nous nous intéressons plus particulièrement aux mouvements du corps et de la tête. Certains travaux proposent des techniques de navigation 3D permettant à la fois de s'orienter dans la scène (rotation égocentrique) et de s'y déplacer (translation), c'est pourquoi nous avons regroupé celles-ci au sein d'une section *Translation + rotation égocentrique* (section 3.4.4).

#### 3.4.1 Zoom

##### *Dispositifs traditionnels et non traditionnels*

Bien que la molette de la souris soit très utilisée pour contrôler le niveau de zoom dans la plupart des logiciels 3D existants, peu de travaux académiques se sont intéressés à l'utilisation du couple clavier/souris pour ce type de contrôle. En revanche, certaines approches plus originales ont été proposées à l'aide de dispositifs non traditionnels.

##### *Dispositifs augmentés*

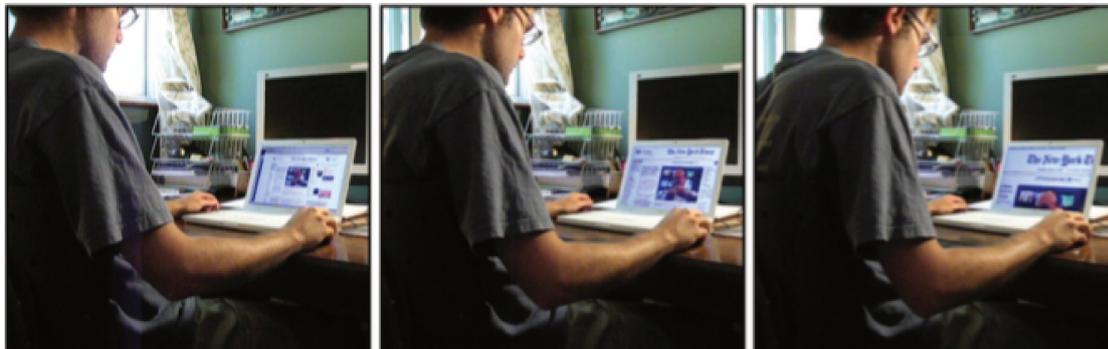
Adhikarla et al. [2014] (technique décrite à la section 3.4.2) proposent de contrôler le niveau de zoom à l'aide des deux mains en utilisant un *Leap Motion*. L'utilisateur dézoome en approchant les mains et réalise l'opération inverse en les écartant.

##### *Mouvements du corps*

**Tête.** La tête est couramment utilisée pour contrôler le niveau de zoom dans un certain nombre de travaux et semble être le seul mouvement du corps utilisé pour cette fonction.

Par exemple Harrison et Dey [2008] présentent une technique pour le contrôle du niveau de zoom dans un contexte de bureautique standard basée sur un couplage avec la tête (figure 3.10). Habituellement, lorsqu'un utilisateur tente de lire un texte trop petit sur son écran, son premier réflexe est de se pencher en avant, afin de voir celui-ci de plus près. La technique présentée par Harrison et al. utilise le même geste pour contrôler le niveau de zoom de l'écran : le zoom augmente si l'utilisateur se penche en avant et diminue s'il recule. Le système utilise une caméra comme dispositif d'entrée. La correspondance entre la distance de la tête de l'utilisateur par rapport à l'écran et le niveau de zoom se fait de manière discrète afin de gagner en stabilité et d'éviter d'avoir un niveau de zoom oscillant en permanence. Un système d'hystérésis permet de plus d'éviter une oscillation du niveau de zoom près des valeurs seuils. Une étude utilisateur a été menée afin d'évaluer la technique sur deux tâches distinctes nécessitant

de contrôler le niveau de zoom. Les participants ont trouvé cette technique intuitive, l'ont jugé plus confortable et plus efficace qu'une condition où la technique n'était pas employée.



**FIGURE 3.10** – La technique *Lean and Zoom* [Harrison et Dey, 2008] permet à l'utilisateur de contrôler le niveau de zoom sur son écran en se penchant en avant.

Sko et Gardner [2009] proposent également trois autres approches pour le zoom dans les jeux de tir à la première personne. *Continuous Zooming* et *Threshold Zooming* (figure 3.23a) permettent également à l'utilisateur de zoomer à l'intérieur de la scène 3D en se penchant en avant. La différence avec *Lean and Zoom* est que le contrôle est ici continu (*Continuous Zoom*)<sup>1</sup> ou avec un seuil (*Threshold Zoom*)<sup>2</sup>. Une autre technique permet, en inclinant la tête sur le roulis, d'aligner le point de vue de l'utilisateur avec la ligne de mire de son arme virtuelle.

La plupart des techniques utilisées pour simuler la parallaxe de mouvement (voir section 3.4.4) couplent également le niveau de zoom à la distance entre la position de la tête de l'utilisateur et le dispositif de suivi. C'est par exemple le cas de la technique de Fishtank ([Ware et al., 1993]). On retrouve notamment cette approche dans les travaux de Sko et Gardner [2009] et Yamaguchi et al. [2009].

### 3.4.2 Translation

Plusieurs approches qui exploitent différentes modalités ont été proposées dans la littérature académique pour le contrôle de la translation.

#### *Dispositifs traditionnels*

Une approche communément utilisée pour naviguer dans une scène 3D à l'aide d'un dispositif 2D est d'exploiter le tracé de l'utilisateur. Dans *Harold* ([Cohen et al., 2000]), un système pour la création de scènes 3D à la volée, l'utilisateur peut déplacer la caméra en traçant le chemin à suivre sur le plan représentant le sol de la scène 3D. Le déplacement de la caméra se fait ensuite automatiquement en suivant ce chemin.

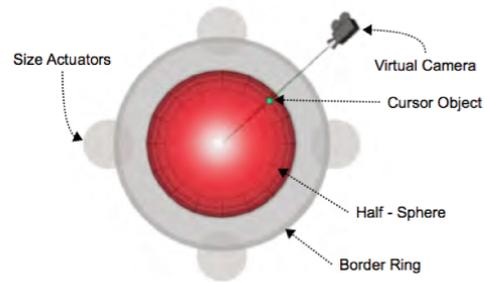
Avec une approche similaire Igarashi et al. [1998] (figure 3.11a) présentent une technique d'interaction pour l'exploration d'environnements 3D sur tablettes. L'utilisateur dessine en 2D le chemin de la caméra dans la scène. Le système projette le tracé de l'utilisateur dans la scène 3D sur le plan du "sol" et déplace

1. Le niveau de zoom (distance focale) est lié à la distance entre la tête et l'écran.

2. Le niveau de zoom est maximal quand l'utilisateur dépasse un seuil de distance par rapport à l'écran



(a) [Igarashi et al., 1998]

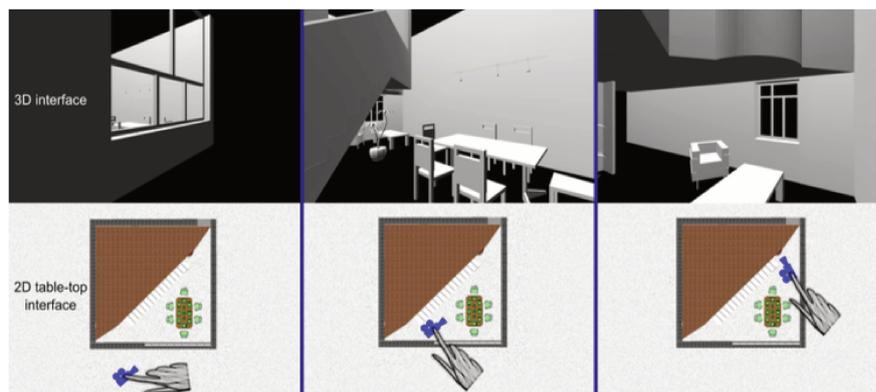


(b) Navidget [Hachet et al., 2008]

**FIGURE 3.11** – À gauche, l'utilisateur navigue dans la scène en traçant sur un dispositif 2D le chemin que la caméra doit prendre. À gauche, Navidget permet de contrôler le déplacement de la caméra vers des points d'intérêts.

la caméra selon le tracé. L'utilisateur peut également remanier le tracé pendant le déplacement de la caméra. Bien que cette technique effectue également des mouvements de rotation pour diriger le déplacement, ceux-ci sont automatiques, l'utilisateur contrôlant uniquement le point d'arrivée et le chemin à suivre dans la scène 3D.

Ajaj et al. [2009] présentent deux techniques (*Camera Based* et *View Based*, figure 3.12) permettant de se déplacer dans des environnements 3D virtuels via une interaction sur surface tactile. Le système couple à un écran sur lequel est représenté la scène observée du point de vue de la caméra avec une table interactive représentant la scène vue de haut et permettant à l'utilisateur de contrôler la position de la caméra de point de vue. L'utilisateur peut directement interagir avec la caméra en la déplaçant sur la table interactive et ainsi changer le point de vue. Dans l'approche *Camera Based*, l'utilisateur déplace la caméra en la faisant glisser du doigt. Avec l'approche, *View Based*, l'utilisateur déplace la scène en la faisant glisser avec le doigt tandis que la caméra reste en position fixe.



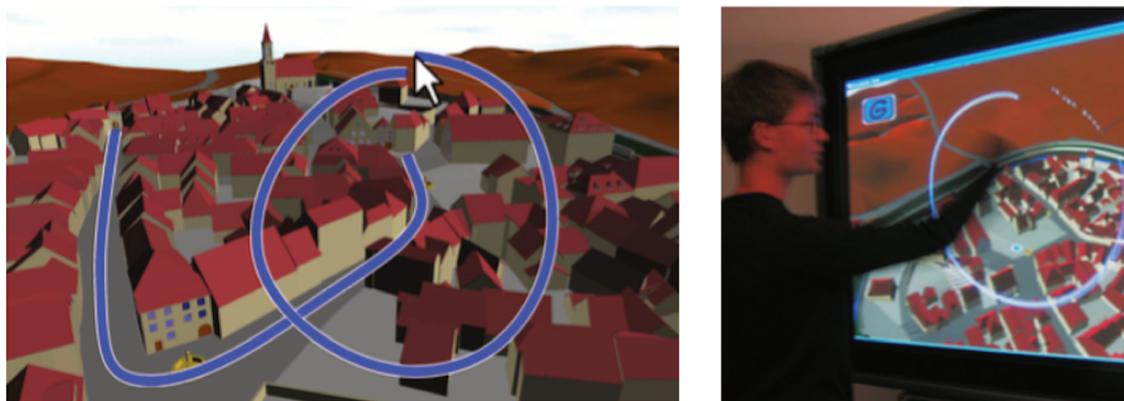
**FIGURE 3.12** – Cette technique pour tables interactives permet de manipuler la position de la caméra de manière directe à partir d'une vue 2D.

Une autre approche, *Point of Interest*, permet à l'utilisateur de préciser un point d'intérêt (POI), la caméra étant alors déplacée de manière automatique afin d'atteindre le point ciblé. Elle a été créée par Mackinlay et al. [1990] et permet de couvrir de larges distances dans une scène 3D et d'atteindre un point d'intérêt rapidement et précisément. Le déplacement du point de vue jusqu'au POI se fait en minimisant le temps de déplacement à travers une fonction décrivant l'approche du POI de manière

asymptotique. L'animation du déplacement permet à l'utilisateur de ne pas se perdre dans la scène 3D, ce qui pourrait se produire si la technique sautait directement d'une position à une autre.

Hachet et al. [2008] ont proposé une technique similaire, *Navidget* pour le positionnement de caméra en 3D à partir du tracé de l'utilisateur (via une tablette, souris, etc.). L'utilisateur entoure la zone vers laquelle il veut se déplacer. La caméra se déplace ensuite, lorsque l'utilisateur termine le tracé, en direction du centre de la zone entourée. Si l'utilisateur termine le tracé, mais maintient le bouton appuyé (dwell) pendant un intervalle de temps précis, le *Navidget* (figure 3.11b) apparaît. Le cercle rouge du *Navidget* représente la partie de l'objet visible à la fin de l'interaction, lorsque la caméra se sera déplacée. La position de la caméra et son orientation sont symbolisées par un symbole de caméra et une ligne partant du centre du *Navidget*. Dans le cas où la position de la caméra ne permettrait pas de visualiser l'objet ciblé, par exemple en cas d'occultation, le symbole de la caméra devient rouge et permet ainsi à l'utilisateur d'éviter toute interaction inutile. L'utilisateur peut également visualiser l'arrière des objets en sortant le curseur du *Navidget* et en y rentrant.

D'autres techniques proposent de combiner à la fois une translation par sélection du point d'intérêt (POI) et tracé du chemin à suivre. Hagedorn et Döllner [2008] ont proposé d'utiliser ce type d'interaction pour manipuler la position de la caméra de point de vue (figure 3.13) dans des environnements virtuels. Un vocabulaire de gestes de tracé (figure 3.14) permet à l'utilisateur de tracer les commandes directement sur la projection 3D. L'action effectuée peut varier en fonction de l'objet sur lequel le tracé est effectué. Ainsi, pour déplacer la caméra le long d'un chemin (ici dans le contexte d'une ville virtuelle), l'utilisateur effectue un tracé sur le plan de la route dans la scène 3D. L'utilisateur peut aussi examiner certains bâtiments en cliquant dessus (POI), la caméra se déplaçant alors automatiquement vers le point d'intérêt.



**FIGURE 3.13** – Un utilisateur dessine le chemin que la caméra devra suivre dans la scène 3D. Le dessin d'un cercle à la fin du tracé signifie que la caméra doit réaliser une rotation complète sur elle-même afin d'obtenir un aperçu de la scène (*overview*) lorsque celle-ci arrive au bout de son chemin.

Dans les travaux de Tan et al. [2001] la translation dans la scène se fait à l'aide de la souris en maintenant le bouton de celle-ci appuyé. La direction du mouvement dépend du sens de déplacement de la souris. Par exemple, si l'utilisateur déplace la souris vers l'avant, la caméra se déplace vers l'avant dans la scène 3D. Le contrôle se fait en vitesse, à la manière d'un joystick : plus le vecteur de déplacement de la souris par rapport à sa position initiale est important, plus la vitesse du déplacement dans la scène 3D est élevée.

Sketch	Navigation Command	Sketch	Navigation Command	Sketch	Navigation Command
	Tilt up the camera.		Rotate the camera to right.		Take an overview position.
	Tilt down the camera.		Rotate the camera to left.		Undo last navigation.

FIGURE 3.14 – Vocabulaire de gestes basés sur le tracé utilisateur

*UniCam* [Zeleznik et Forsberg, 1999], offre un ensemble de techniques permettant d'effectuer plusieurs contrôles de caméra à l'aide de gestes effectués avec la souris. La sélection du contrôle de caméra s'effectue en fonction du point de départ du tracé et de sa direction. Parmi les contrôles proposés, les auteurs proposent d'utiliser le glisser-déposer pour contrôler la translation sur l'axe vertical (y) et horizontal (x). Ainsi, la caméra sera par exemple translatée sur l'axe horizontal si l'utilisateur maintient la souris et la déplace horizontalement.

#### *Au-delà des dispositifs traditionnels*

La translation a également été étudiée avec des dispositifs non conventionnels, par exemple des dispositifs Polhemus à 6 degrés de liberté (seulement 3 degrés de libertés utilisés pour la translation), ou en exploitant les mouvements du corps.

#### *Dispositifs augmentés*

Stoakley et al. [1995] et Pausch et al. [1995] ont proposé une technique d'interaction pour les scènes 3D immersives appelée *World in Miniature* (WIM, voir figure 3.18a). Cette technique permet à l'utilisateur d'interagir avec la scène 3D à l'aide d'un modèle réduit de la scène qui permet d'avoir une vue d'ensemble de celle-ci. Toute modification impactant la scène originale est reflétée sur le modèle réduit. Ainsi, si l'utilisateur déplace un objet dans le monde miniature, la position de l'objet dans la scène grande nature est modifiée à l'identique. La position de l'utilisateur est également représentée dans le WIM, ainsi que son champ de vision (figure 3.18b). Cette technique permet donc à l'utilisateur de se déplacer en déplaçant sa représentation dans le monde miniature. Pausch et al. ont ainsi étudié la translation en utilisant un WIM comme outil de locomotion dans les scènes 3D.

Brooks [1992] présentent une approche différente pour se déplacer dans une scène 3D qui repose sur l'utilisation de deux joysticks à trois degrés de liberté pour effectuer des translations horizontales et transversales et des rotations verticales. Le joystick placé dans la main dominante contrôle la translation sur l'axe de la profondeur à l'aide de l'axe transversal et la translation horizontale à l'aide de l'axe horizontal tandis que la rotation verticale de l'autre joystick contrôle la translation sur l'axe vertical. Les autres actions réalisables sont décrites dans la suite de ce chapitre à la section 3.4.3.

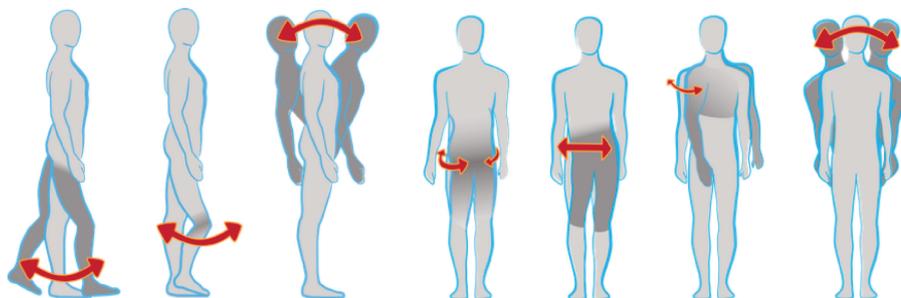
Adhikarla et al. [2014] proposent une approche pour la navigation dans des cartes 3D qui utilise un *Leap Motion*<sup>1</sup>. En déplaçant la main sur l'axe horizontal ou sur l'axe vertical, la caméra est translatée sur les mêmes axes.

1. <https://www.leapmotion.com/>

### Mouvements du corps

**Buste.** Roupé et al. [2014] proposent un ensemble de mouvement du corps pour contrôler la translation dans les scènes 3D. Les auteurs proposent notamment d'utiliser l'inclinaison du buste vers l'avant ou vers l'arrière pour, respectivement, avancer ou reculer dans la scène 3D. D'autres contrôles de caméra sont également supportés (voir 3.4.3).

Dans *LazyNav* Guy et al. [2015] présentent une méthodologie pour la création de techniques de translation basées sur des mouvements du corps (figure 3.15) autres que les mouvements de la tête et des mains, ce qui laisse les mains et la tête libres pour effectuer d'autres tâches. Trois mouvements différents sont considérés pour la translation : genou fléchi en avant, buste en avant/arrière, mettre un pied en avant/arrière.



**FIGURE 3.15** – Mouvements du corps proposés pour la translation avec la technique *LazyNav* [Guy et al., 2015].

**Pieds.** Beaucoup de techniques proposées pour la translation reposent sur la simulation de la marche (*Walk in Place* ou *WIP*). *The Virtual treadmill*, une des premières techniques de ce type a été proposé par Slater et al. [1995]. Cette technique utilise un réseau de neurones pour reconnaître les mouvements de marche de l'utilisateur. Une expérience comparant la technique *WIP* par rapport à une technique de base utilisant une souris 3D a montré qu'elle s'avérait plus naturelle à utiliser en raison de sa ressemblance aux mouvements dans le monde réel et qu'elle offrait une meilleure sensation d'immersion et de présence dans la scène 3D.

Dans une étude conduite ultérieurement, Usoh et al. [1999] ont comparé trois modalités de déplacement dans les scènes 3D immersives :

- *Flying*, la technique standard de déplacement utilisée de nos jours dans les logiciels de modélisation 3D (la caméra de point de vue est déplacée dans la scène à la manière d'un objet volant, voir [Ware et Osborne, 1990]) ;
- la technique *Walk in Place* ;
- la marche réelle.

Leur étude confirme les résultats obtenus plus tôt par Slater et al. [1995] et montre que la marche réelle pour le déplacement dans des scènes 3D offre une meilleure sensation de présence que les techniques *Flying* et *WIP*.

En 2001, LaViola et al. [2001] ont présenté un ensemble de techniques pour la translation dans des environnements immersifs laissant les mains libres grâce à l'utilisation des pieds pour se déplacer dans

la scène 3D. La première technique présentée dans ce travail est le *Step WIM*. Le *Step WIM* est une carte réduite de la scène virtuelle projetée sous les pieds de l'utilisateur. Afin de pouvoir interagir avec celle-ci, l'utilisateur doit porter les *Interaction Splippers* (figure 3.26a). En se déplaçant sur cette carte, l'utilisateur peut contrôler sa position dans la scène virtuelle. Trois opérations sont définies dans le *Step WIM* : l'invocation ou ouverture du *Step WIM* lorsque l'utilisateur tape ses pieds l'un sur l'autre, le redimensionnement du WIM et la fermeture du *Step WIM*.

Le déplacement dans le WIM se fait en baissant le regard vers le WIM et en claquant ses pieds l'un contre l'autre. Pour fermer le WIM, l'utilisateur doit claquer ses pieds en détournant le regard du WIM. Une autre approche consiste à se hisser sur la pointe des pieds et redescendre. Le délimiteur est différent, mais la technique reste la même.

### 3.4.3 Rotation égocentrique

#### *Dispositifs traditionnels*

Dans [Ajaj et al., 2009] (vu à la section 3.4.2), l'utilisateur peut contrôler la rotation de la caméra sur son axe vertical. En fonction de l'approche choisie (*Camera based* ou *View based*) la rotation se fait différemment. Pour la première approche, les auteurs utilisent la technique *Rotate'N Translate* [Kruger et al., 2005] qui permet de combiner la translation avec la rotation en fonction du point de contact choisi par l'utilisateur pour le déplacement. Dans la seconde approche (*View based*), les utilisateurs passent par un mode pour contrôler la rotation de la scène autour de la caméra. La rotation s'effectue alors avec un glisser-déposer.

Dans la technique présentée par Tan et al. [2001], déjà vue à la section 3.4.2, l'utilisateur peut faire une rotation de la caméra sur l'axe vertical afin pour regarder à gauche ou à droite. Ce contrôle est réalisé avec la souris en faisant un glisser-déposer vers la droite ou vers la gauche.

#### *Au-delà des dispositifs traditionnels*

#### *Dispositifs augmentés*

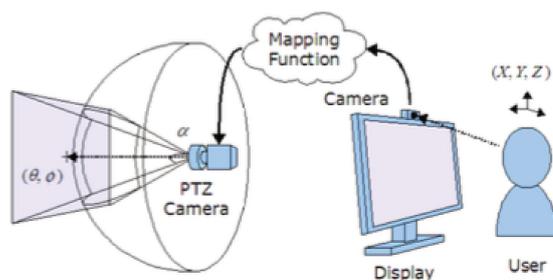
Plusieurs travaux ont proposé l'utilisation de dispositifs avancés pour le contrôle de la rotation exocentrique. Mulder et Liere [2000] ont proposé d'utiliser une baguette munie d'un capteur Polhemus à 6 degrés de liberté. En faisant une rotation de la baguette sur le lacet et le tangage, la caméra effectue un mouvement correspondant. [Brooks, 1987] (présenté en section 3.4.2) permet de contrôler la rotation égocentrique du point de vue avec deux joysticks. Le premier (dans la main dominante) contrôle la rotation du point de vue sur le lacet en effectuant une torsion du joystick. Le second joystick contrôle la rotation sur le roulis et le tangage en effectuant une translation du joystick horizontale ou transversale. Enfin, le *Leap Motion* a été utilisé par Adhikarla et al. [2014] pour contrôler la rotation sur des cartes 3D. En effectuant une rotation de la main face à l'écran, une rotation sur le roulis est effectuée.

#### *Mouvements du corps*

**Tête.** Dans [Mulder et Liere, 2000] les rotations de la tête contrôlent la rotation du point de vue. Cette technique amplifie les mouvements de rotation de la tête sur un axe (lacet ou tangage), ce qui permet

d'augmenter l'amplitude. Lorsque l'utilisateur tourne la tête dans un sens de  $\alpha$  degrés, la scène tourne dans le sens opposé de  $\alpha$  degrés, comme si la scène 3D se trouvait autour de l'utilisateur. Cette approche a été utilisée à la fois avec le lacet et le tangage.

Dans [Yamaguchi et al., 2009] la tête sert également à contrôler la rotation du point de vue mais dans le contexte de la vidéoconférence (figure 3.16) en utilisant plusieurs caméras : une caméra standard et une caméra Pan Tilt Zoom (PTZ). La caméra standard permet d'obtenir la position de la tête d'un premier participant. La caméra PTZ, positionnée chez son interlocuteur peut être manipulée par le premier de manière intuitive comme s'il se trouvait directement en face de son interlocuteur en bougeant la tête. Plusieurs paramètres sont ajustables au niveau de la caméra PTZ, parmi lesquels la rotation de la caméra sur le lacet  $\theta$  et la rotation de la caméra sur le tangage  $\phi$ . Lorsque l'utilisateur se déplace, les paramètres de la caméra PTZ sont modifiés afin de prendre en compte sa nouvelle position. Ainsi, si l'utilisateur se penche sur la gauche, la caméra PTZ tourne vers la gauche sur le lacet pour révéler à l'utilisateur ce qui se situe à la gauche de son interlocuteur. Si l'utilisateur se penche en avant vers l'écran, la distance focale de la caméra PTZ augmente, ce qui produit un effet de zoom.



**FIGURE 3.16** – Yamaguchi et al. [2009] utilisent une caméra Pan Tilt Zoom (PTZ) dans un contexte de vidéoconférence pour étendre le champ de vision des participants.

Deux autres techniques de contrôle de rotation égocentrique à l'aide de la tête ont été proposés dans [Sko et Gardner, 2009]. La première technique, *Spinning*, permet à l'utilisateur de faire une rotation du point de vue de  $180^\circ$  sur le lacet en faisant un mouvement rapide de la tête au-delà d'un certain seuil sur le lacet. La seconde, *Peering* (figure 3.23b) permet d'effectuer un mouvement de roulis pour jeter un coup d'oeil derrière un obstacle en tournant la tête sur le roulis.

La technique *Walk-in-Place* proposée par Slater et al. [1995] (section 3.4.2) permet également à l'utilisateur de choisir la direction du déplacement, qui est contrôlée à l'aide de l'angle de lacet de la tête. Une approche identique est utilisée par Usoh et al. [1999].

Fuhrmann et al. [1998] présentent une technique d'interaction pour la navigation 3D utilisant la rotation de la tête. Cette technique utilise le tangage de la tête afin de permettre à l'utilisateur d'avancer ou de reculer dans la scène 3D. Deux modes d'interactions sont définis. Le premier, qui permet de contrôler le déplacement dans la scène, est décrit à la section 3.4.4. Le second permet à l'utilisateur de tourner la tête sur le lacet afin d'observer la scène. Le changement de mode s'effectue en tournant la tête sur l'axe du roulis : l'utilisateur se penche à droite pour passer en mode navigation et à gauche pour passer en mode observation.

**Buste.** LaViola et al. [2001] (voir section 3.4.2) ont proposé d'utiliser l'orientation du buste pour contrôler la direction du mouvement lors du déplacement. Une approche similaire ([Roupé et al., 2014]) exploite la rotation des épaules sur le lacet pour contrôler l'orientation du déplacement dans la scène 3D. La rotation des épaules fait également partie des mouvements proposés dans [Guy et al., 2015], ainsi que d'autres mouvements du corps tels que la rotation des hanches et l'inclinaison du buste à l'horizontale (droite/gauche).

#### 3.4.4 Translation + rotation égocentrique

Certaines techniques permettent de manipuler à la fois la translation et la rotation et ainsi de réaliser de la navigation 3D.

##### *Dispositifs traditionnels*

[Ware et Osborne, 1990] est une technique de POI avancée permettant, après sélection d'un point d'intérêt à l'aide de la souris, de déplacer la caméra vers ce point et de l'orienter afin que celle-ci soit face à l'objet. La caméra réalise donc un mélange de translation et de rotation sur elle-même.

Dans *Navidget* [Hachet et al., 2008], en plus de préciser le POI, l'utilisateur peut également préciser la direction vers laquelle la caméra doit être orientée à la fin du déplacement en faisant un dwell avec la souris lors de la sélection du POI et en réalisant un glisser-déposer de la souris pour choisir l'angle.

Enfin, le système de Hagedorn et Döllner [2008] présenté à la section 3.4.2 combine des mouvements de translation et de rotation. Après sélection du POI, la caméra se déplace automatiquement vers le point d'intérêt en suivant la topologie de la scène 3D (chemin, routes).

##### *Au-delà des dispositifs traditionnels*

##### *Dispositifs augmentés*

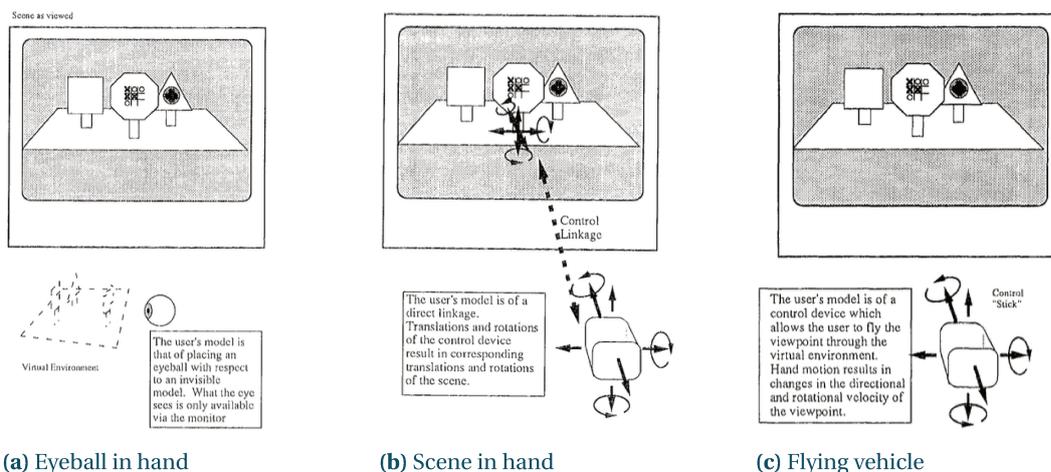
Ware et Osborne [1990] décrivent trois approches pour la navigation 3D dans des scènes immersives (figure 3.17) : *Eyeball in hand*, *Scene in hand* et *Flying vehicle* décrites ci-dessous. Toutes ces approches reposent sur un dispositif d'entrée à 6 degrés de liberté (ici un *Polhemus 3Space*) avec une correspondance absolue.

**Eyeball in hand** (figure 3.17a) utilise la métaphore d'un œil par lequel la scène est observée. En orientant l'œil, l'utilisateur contrôle l'orientation de la vue et en déplaçant l'œil la position du point de vue dans la scène 3D. Cette approche est notamment utilisée par Badler et al. [1987] pour contrôler le positionnement d'objets dans une scène 3D à l'aide d'un dispositif Polhemus à 6 degrés de liberté.

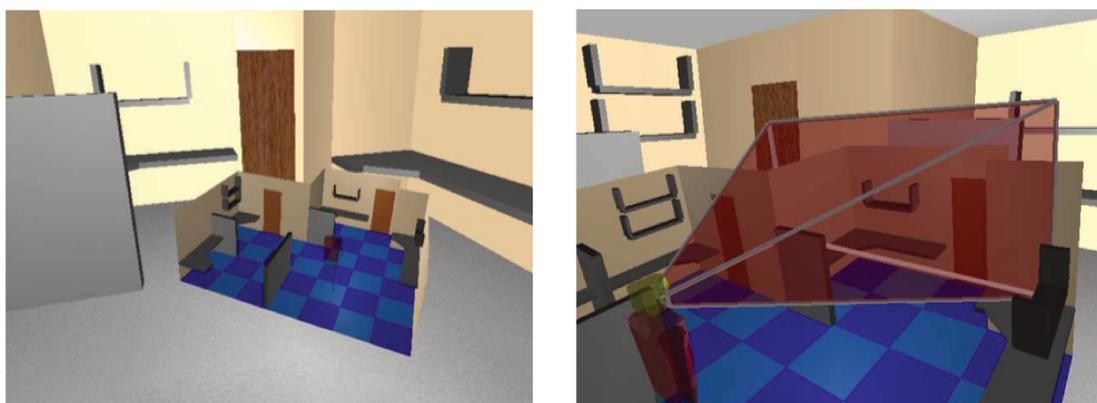
**Scene in Hand ou World in miniature** (figure 3.17b) utilise une métaphore selon laquelle la totalité de la scène serait incluse dans le dispositif d'interaction. Ainsi, si l'utilisateur tourne le dispositif dans le sens antihoraire, la scène tourne de manière identique. Cette approche est utilisée par Ware et Jessome [1988], avec une souris augmentée à six degrés de liberté permettant de manipuler la scène 3D comme si celle-ci se trouvait dans la main de l'utilisateur.

### 3. Contrôle du point de vue

#### 3.4. Techniques d'interaction pour le contrôle du point de vue



**FIGURE 3.17** – Approches de navigation 3D décrites par Ware et Osborne [1990]. *Eyeball in hand* utilise la métaphore d'un œil dont la position serait contrôlée par l'utilisateur. La scène est observée du point de vue de cet œil. Avec *Scene in hand*, l'utilisateur manipule le dispositif d'entrée comme s'il manipulait directement la scène 3D. Enfin, avec *Flying vehicle*, l'utilisateur déplace un véhicule dans la scène 3D et la scène est observée du point de vue de celui-ci. Toutes ces approches utilisent un dispositif d'entrée à 6 degrés de liberté.



**(a) World in Miniature** **(b) L'utilisateur et son champ de vision dans le WIM**

**FIGURE 3.18** – Le *World in Miniature* est un outil pour la navigation et la manipulation d'objets dans des scènes 3D immersives. Celui-ci se présente sous la forme d'une maquette et offre à l'utilisateur une vision d'ensemble de la scène. De plus, celle-ci étant interactive et directement liée à la scène 3D, elle permet à l'utilisateur de déplacer des objets lointains sans changer de point de vue.

**Flying vehicle** (figure 3.17c) permet à l'utilisateur de contrôler son point de vue dans la scène comme s'il se trouvait dans un véhicule. Cette approche est très utilisée dans le contexte des scènes 3D immersives puisque l'utilisateur se trouve dans la scène virtuelle. On citera notamment [Brooks, 1992] pour la visite de monuments virtuels dans le contexte de l'architecture. L'approche proposée repose sur un capteur Polhemus à 6 degrés de liberté utilisant seulement trois des six degrés de liberté : translation horizontale, translation transversale et rotation sur le lacet. Pour chaque mouvement du capteur, le contrôle équivalent est effectué par la caméra de point de vue. Fleury et al. [2010] ont également proposé l'*Immersive Interactive Virtual Cabin*, un concept permettant à l'utilisateur de se déplacer dans la scène 3D tout en restant au sein du référentiel d'une cabine virtuelle embarquant les outils nécessaires à l'interaction avec la scène 3D.

Cependant, la plupart des dispositifs à 6 degrés de liberté sont incompatibles avec des tâches de pointage en deux dimensions. Avec la *Roly Poly Mouse*, Perelman et al. [2015] présentent un nouveau dispositif d'entrée permettant de contrôler jusqu'à 6 degrés de liberté (3 degrés de rotation et 3 degrés de translation) tout en permettant de réaliser des tâches de pointage 2D grâce à la stabilité et à la forme du dispositif.

### **Mouvements du corps**

**Tête.** De nombreuses techniques couplent la position de la tête de l'utilisateur à la position de la caméra. Ce couplage appelé "perspective couplée à la tête" (Head Coupled Perspective, HCP), permet de contrôler la perspective d'une scène 3D à partir des mouvements de la tête. Grâce à un dispositif de capture détectant la position de la tête de l'utilisateur, le système adapte le rendu de la scène 3D afin de reproduire la sensation de parallaxe de mouvement vécue dans un environnement réel. Ainsi, lorsque l'utilisateur se penche sur un côté, par exemple pour voir derrière un objet, la caméra effectue un mouvement combinant translation et rotation égocentrique (voir figure 3.19).



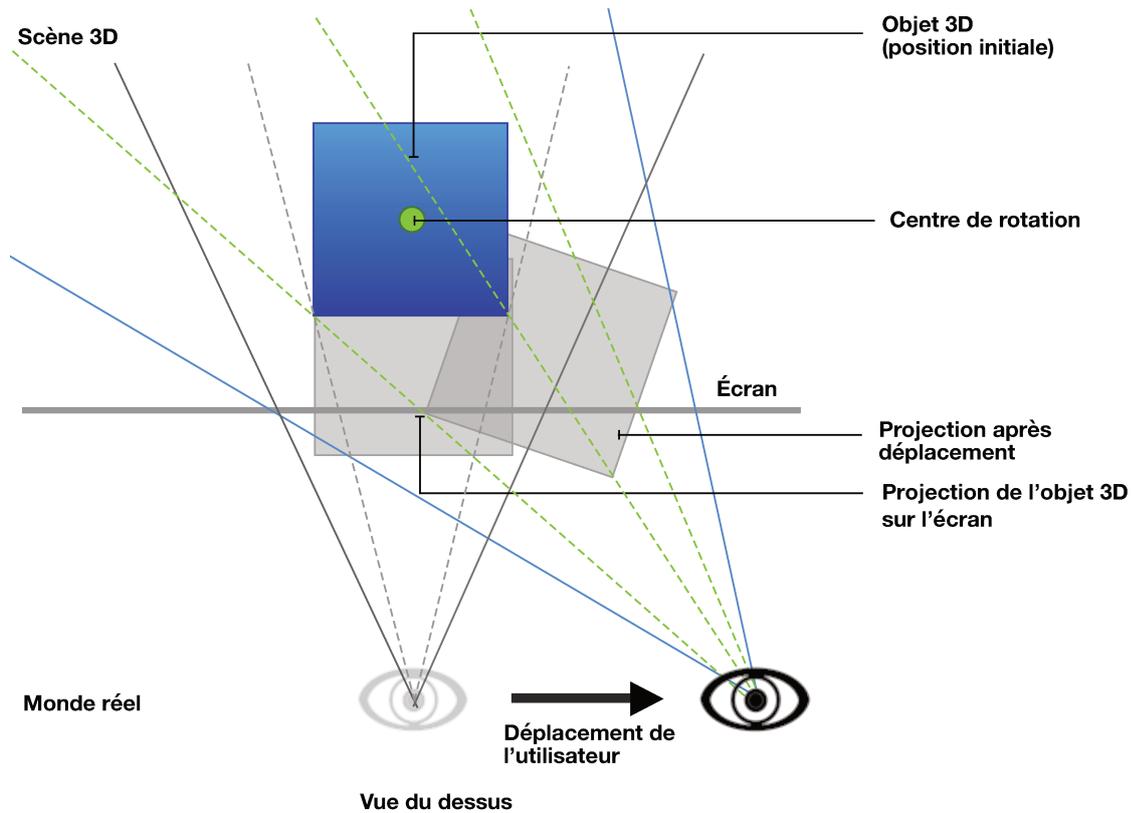
**FIGURE 3.19** – La perspective couplée à la tête renforce la perception de l'utilisateur en adaptant le rendu de la scène 3D par rapport à la position de la tête de l'utilisateur. Ainsi, si celui-ci se penche à gauche (image de droite), la scène est traduite sur la gauche et la caméra s'oriente verticalement pour conserver le point central de la scène au centre de son champ de vision.

L'HCP fonctionne de la manière suivante. Un centre de rotation est défini par l'utilisateur (en vert sur la figure 3.20). Un système de tracking permet d'obtenir le déplacement de la tête par rapport à une position de départ dans l'espace physique. Une correspondance est faite entre la position de la tête dans l'espace physique et la position de la caméra dans la scène 3D. La contrainte principale de l'HCP est que la caméra de point de vue doit toujours rester pointée vers le centre de rotation en effectuant une rotation sur elle-même (rotation égocentrique). Ainsi, pour chaque déplacement de l'utilisateur, la caméra est traduite dans la même direction que celui-ci et une rotation sur les axes vertical et horizontal est effectuée afin que la caméra reste pointée vers le centre de rotation.

L'HCP permet de renforcer l'immersion dans une scène 3D. Ce contrôle de caméra est à la base du *Fishtank VR* [Ware et al., 1993][Arthur et al., 1993], une technique de contrôle du point de vue mêlant de la perspective couplée à la tête avec une vision stéréoscopique. Dans leur travail, Ware et al. mènent plusieurs études sur l'impact de la vision stéréoscopique et de l'HCP sur la perception 3D. Une première étude révèle que l'HCP offre une meilleure perception de la profondeur que la vision stéréo. Une seconde étude nécessitant de tracer un chemin dans une scène 3D semble également révéler que l'HCP joue un

### 3. Contrôle du point de vue

#### 3.4. Techniques d'interaction pour le contrôle du point de vue



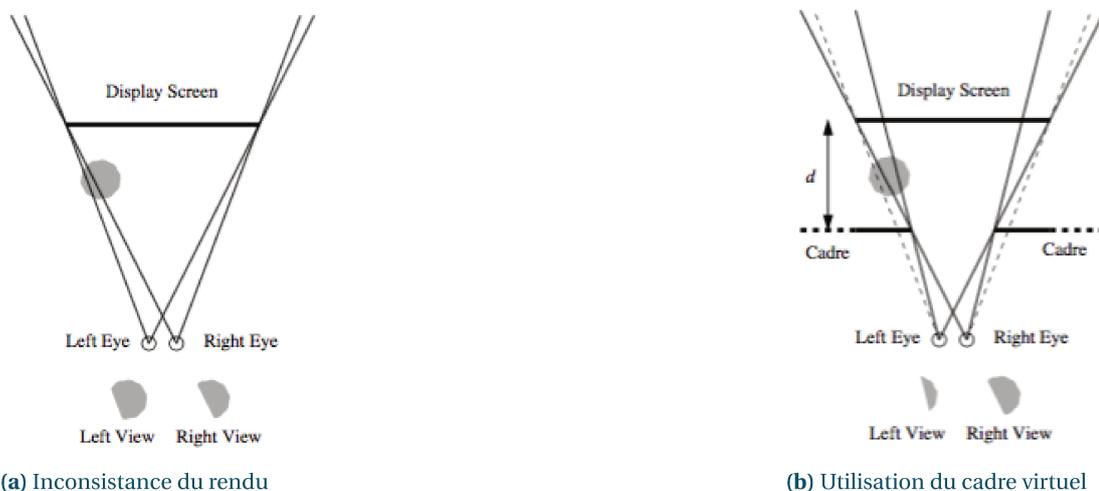
**FIGURE 3.20** – Le fonctionnement de la perspective couplée à la tête est le suivant : pour une translation donnée de la tête de l'utilisateur, le système opère une translation de la caméra dans la scène 3D et une rotation de la caméra sur elle-même de sorte que celle-ci pointe toujours vers le même point.

rôle plus important que la vision stéréo. Ware et Franck [1996] ont également étudié l'impact de la vision stéréo et de l'HCP sur le tracé dans des graphes en trois dimensions. Cette étude montre que l'HCP permet de tracer un chemin dans des graphes 1.375 fois plus larges qu'avec la vision stéréo. Arsenault et Ware [2000] ont étudié les performances obtenues avec ou sans HCP sur une tâche de type Fitts en trois dimensions. L'étude montre une diminution de temps de réalisation de la tâche dans la condition utilisant l'HCP.

Cependant, d'autres travaux ont au contraire montré que la vision stéréoscopique offrait de meilleures performances que l'HCP. C'est notamment le cas de Boritz et Booth [1997] qui ont comparé l'impact d'interactions avec ou sans HCP et avec ou sans vision stéréo sur le temps et la précision dans une tâche de recherche dans un environnement en trois dimensions. Les résultats montrent un net avantage de la vision stéréoscopique par rapport à la vision monoscopique tant au niveau de la précision qu'au niveau du temps de réalisation de la tâche. Cependant, aucun effet significatif de l'HCP n'a été observé sur le temps de réalisation de la tâche. Dans une étude postérieure Teather et Stuerzlinger [2007] ont également comparé l'impact de l'HCP et de la vision stéréoscopique sur le placement d'objets dans une scène 3D. Comme dans l'étude précédente, les résultats montrent un effet significatif de la vision stéréoscopique sur la précision de la tâche, mais aucun effet significatif de l'HCP. Une explication avancée par Arsenault et Ware [2000] concernant les résultats mitigés avec l'HCP dans les travaux de Boritz et al. est que la tâche utilisée dans [Boritz et Booth, 1997] ne nécessite pas de déplacer la tête.

Tous ces travaux s'accordent en revanche sur le fait que la combinaison de l'HCP et de la vision stéréo offre de meilleurs performances que l'un ou l'autre des indices visuels pris séparément.

Dans certains cas (voir ci-dessous), le rendu obtenu avec l'HCP peut s'avérer inconsistant. Mulder et Liere [2000] ont proposé d'améliorer la perception de l'utilisateur avec l'HCP dans le cas où un objet de la scène 3D, au premier plan par rapport au centre de la scène, paraît tronqué, alors qu'il devrait sortir de l'écran (figure 3.21a). Afin de pallier ce problème, un cadre virtuel d'une certaine profondeur est placé entre la projection de la caméra et la scène 3D (figure 3.21b). L'utilisateur a ainsi l'impression de regarder la scène 3D à travers le cadre d'une fenêtre. Ainsi, l'objet qui auparavant semblait tronqué sans raison apparente est maintenant masqué partiellement par le cadre virtuel.



**FIGURE 3.21** – Sur les deux figures, un objet 3D est situé à une profondeur plus faible que le centre de la rotation de la perspective couplée à la tête (parallaxe négative). L'objet paraît donc sortir de l'écran. À gauche, l'objet paraît tronqué du fait de la limite de l'écran, ce qui rend le rendu inconsistant. À droite, en ajoutant un cadre virtuel entre la projection de l'objet et les yeux de l'utilisateur, l'objet paraît tronqué par le cadre virtuel et non plus par les limites de l'écran.

L'HCP a également été utilisée dans le contexte de la vidéoconférence. Harrison et Hudson [2008] ont travaillé sur un système utilisant la perspective couplée à la tête (figure 3.22) afin d'offrir davantage d'immersion dans ce contexte. Un algorithme de soustraction d'image permet de créer deux calques, l'un comprenant l'utilisateur seulement et le second l'arrière-plan sans l'utilisateur. Grâce à ces deux calques, l'impression de profondeur de la scène est accentuée. Les deux calques étant séparés et le système adaptant la perspective en fonction de la tête de l'utilisateur, chaque interlocuteur peut en bougeant la tête jeter un coup d'oeil autour et derrière son interlocuteur.

L'HCP est également utilisée dans les jeux vidéo, principalement pour apporter davantage d'immersion dans les jeux à la première personne ([Sko et Gardner, 2009], voir figure 3.25a). Kulshreshth et LaViola [2013] ont conduit une étude pour évaluer les performances de deux catégories de joueurs : expérimentés et débutants. Quatre jeux sont utilisés pour cette étude : *Arma II*<sup>1</sup>, *Dirt 2*<sup>2</sup>, *Microsoft Flight*<sup>3</sup> et *Wings of Prey*<sup>4</sup>. Tous ces jeux sont représentés à la première personne et permettent à l'utilisateur d'interagir de

1. un jeu de tir à la première personne  
2. un jeu de course  
3. un simulateur de vol  
4. un simulateur de combat aérien

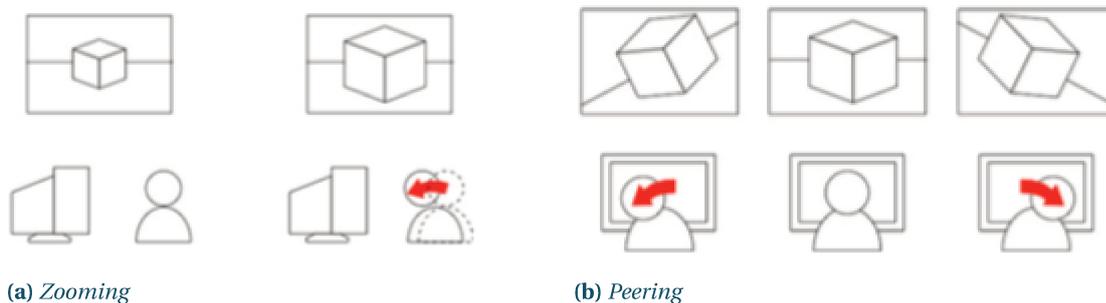
### 3. Contrôle du point de vue

#### 3.4. Techniques d'interaction pour le contrôle du point de vue



**FIGURE 3.22** – La perspective couplée à la tête est utilisée par Harrison et Hudson [2008] afin d'augmenter la présence dans le contexte de la vidéoconférence en adaptant la perspective par rapport à la position de la tête.

manière intuitive avec l'environnement du jeu. L'utilisateur peut ainsi jeter un coup d'oeil à droite ou à gauche pour mieux prendre conscience de son environnement comme il le ferait dans la réalité. Chaque participant devait tester les quatre jeux avec et sans suivi de la tête. Les résultats de l'étude montrent que le suivi de la tête améliore les performances dans les jeux de tirs à la première personne, mais que celui-ci est moins bien adapté pour les jeux de course. Au final, en fonction du niveau d'expérience des joueurs, les performances avec le suivi de la tête varient. Les utilisateurs débutants bénéficient peu de celui-ci, tandis que les plus expérimentés en profitent davantage.



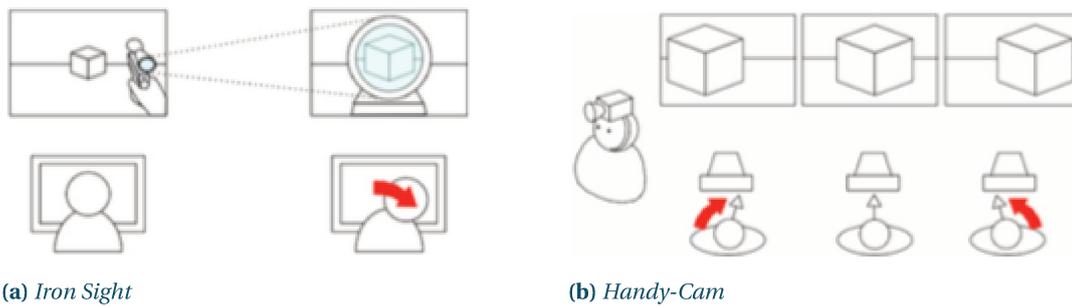
**FIGURE 3.23** – À gauche, la technique *Zooming* permet à l'utilisateur de modifier le niveau de zoom en se penchant en avant. Deux techniques sont proposées. La première permet de contrôler le niveau de zoom de manière continue, tandis que la seconde donne un niveau de zoom maximal passé un certain seuil. À droite, la technique *Peering* permet de jeter un coup d'oeil derrière un obstacle. Pour réaliser ce mouvement, l'utilisateur penche la tête à droite ou à gauche.

**Buste.** Dans *Heaven and Earth* [Fairchild et al., 1993], en plus de la navigation avec les mains, il est possible de naviguer en penchant le buste. *Lean-Based* permet à l'utilisateur de se déplacer dans la scène en se penchant sur un côté. Cette technique utilisant la position de la tête pour évaluer le vecteur vitesse, les mains restent libres pour mener à bien d'autres tâches en parallèle. Deux versions de cette technique ont été créées utilisant soit une correspondance absolue avec un contrôle en position, soit une correspondance relative avec un contrôle en vitesse. La première version amplifie<sup>1</sup> le vecteur de déplacement de la tête par rapport à un point d'origine fixe. La seconde version utilise le vecteur de

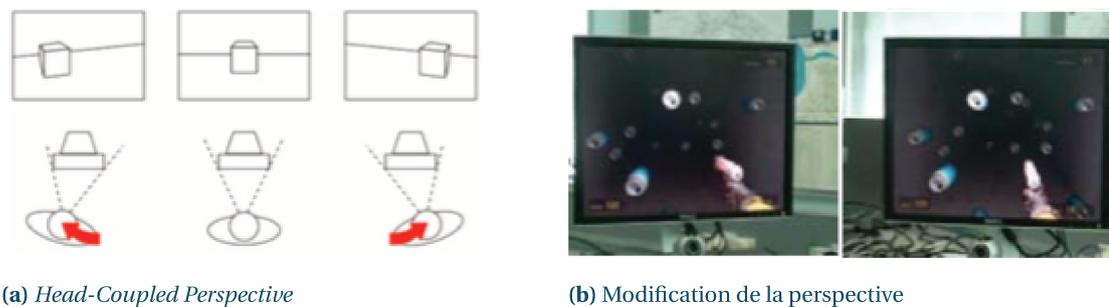
1. La fonction utilisée est la suivante :

$$f(p) = \frac{1}{k} \cdot (e^{k \cdot p} - 1)$$

Un faible déplacement de la tête aura donc peu d'incidence sur le déplacement dans la scène 3D, tandis qu'un déplacement plus large permet à l'utilisateur d'atteindre des amplitudes de mouvement bien plus élevées.



**FIGURE 3.24** – À gauche, la technique *Iron Sight* permet à l'utilisateur d'aligner le point de vue avec la ligne de mire de son arme virtuelle. Ce mouvement est réalisé en penchant la tête sur le côté et s'active passé un certain seuil. À droite, la technique d'immersion *Handy-cam* ajoute du bruit au niveau du point de vue pour donner l'impression que la caméra n'est pas fixée sur un support, mais tenue par un être humain.



**FIGURE 3.25** – À gauche, la perspective couplée à la tête permet de facilement obtenir une perception augmentée d'immersion, à travers la parallaxe de mouvement. À droite, un exemple de contrôle du point de vue dans une scène 3D avec la perspective couplée à la tête.

déplacement de la tête pour calculer la vitesse. Tant que l'utilisateur reste penché, celui-ci se déplace dans la scène dans la direction souhaitée.

Une autre technique présentée dans cette étude consiste à utiliser l'inclinaison du buste (figure 3.26b) pour naviguer dans la scène. L'angle d'inclinaison du corps est calculé à l'aide d'un suivi de la position de la tête et des hanches de l'utilisateur. Le contrôle se fait en vitesse : plus l'inclinaison du buste est importante et plus l'utilisateur se déplace rapidement dans la scène.



**FIGURE 3.26** – À gauche, *The Interaction Slippers* permet de faire apparaître ou disparaître le WIM et de se déplacer au sein de celui-ci. À droite, l'utilisateur se penche pour naviguer dans la scène.

**Mains.** Nabiyouni et al. [2014] ont utilisé des gestes de la main capturés par un *Leap Motion* afin de contrôler la translation. Une étude a comparé l'efficacité de plusieurs techniques (la première basée sur *EyeBall in Hand* et la seconde sur *Flying Vehicle*) pour la navigation dans des environnements 3D selon diverses tâches. La première tâche consistait à se déplacer dans la scène 3D d'un point A à un point B. La deuxième nécessitait de trouver un objet dans la scène 3D. Enfin, la troisième nécessitait de suivre un chemin dans la scène. Avec *EyeBall in Hand*, la main joue le rôle de la caméra et la correspondance est absolue. Avec *Flying Vehicle*, le contrôle de la position de la caméra se fait en vitesse.

*Heaven and Earth* [Fairchild et al., 1993] (déjà vu à la section 3.4.4) sont trois méthodes de navigation différentes pour des utilisateurs novices dans des scènes immersives avec des casques de réalité virtuelle. La première technique, *Flying Hand* permet de naviguer au sein de la scène 3D à l'aide de sa main. La distance de la main par rapport à la tête permet de définir la vitesse du mouvement. Une seconde technique basée sur la première, *Floating Guide*, fournit à l'utilisateur un guide visuel sous la forme d'une sphère virtuelle projetée sur l'écran de son casque. En manipulant cette sphère, l'utilisateur réalise les mêmes mouvements que pour la première technique mais avec un retour visuel supplémentaire.

**Pieds.** Brooks [1992] a étudié l'usage d'un tapis roulant directionnel pour contrôler à la fois la rotation et la translation de l'utilisateur. Une approche similaire a été utilisée par Darken et al. [1997] avec l'*Omni-directional Treadmill*.

#### 3.4.5 Rotation exocentrique

La rotation exocentrique ou contrôle orbital est la rotation de la caméra dans une scène 3D par rapport à un point donné de la scène de manière à ce que la caméra pointe toujours vers le point central de la scène. Elle a été étudiée de manière extensive dans divers cas d'applications tels que les jeux à la première personne ou la modélisation 3D.

##### *Dispositifs traditionnels*

Dans [Chen et al., 1988] une étude évalue différentes interfaces permettant d'ajuster l'orientation 3D d'un objet. Bien que ces techniques ne permettent pas de faire du contrôle de point de vue (car la caméra ne se déplace pas dans la scène), ce travail est souvent cité dans les travaux traitant du contrôle orbital. En effet, au lieu de permettre à l'utilisateur de tourner autour d'un objet, l'utilisateur fait tourner l'objet sur lui-même tandis que la caméra reste fixe. Toutes les techniques considérées dans cette étude pour contrôler l'orientation d'un objet passent par l'utilisation d'un ou plusieurs sliders.

Les auteurs proposent *Virtual Sphere*, une technique qui permet à l'utilisateur de manipuler l'orientation verticale (lacet) et horizontale (tangage) du modèle lorsqu'il interagit à l'intérieur du cercle et l'orientation sur l'axe transversal (roulis) en faisant un mouvement de glissement de la souris le long des extrémités du cercle. Une expérience comparant cette technique à l'utilisation de plusieurs sliders montrent des temps moyens nettement plus faibles pour des rotations complexes.

Des travaux similaires ont été conduits par Shoemake [1992] avec *Arcball*. Cette technique est semblable à *Virtual Sphere* pour le contrôle de la rotation d'objets mais offre cependant un avantage supplémentaire. En effet, avec l'approche classique, en fonction du chemin suivi par la souris pour manipuler l'orientation en partant d'un point A jusqu'à un point B, l'orientation finale de l'objet peut varier, ce

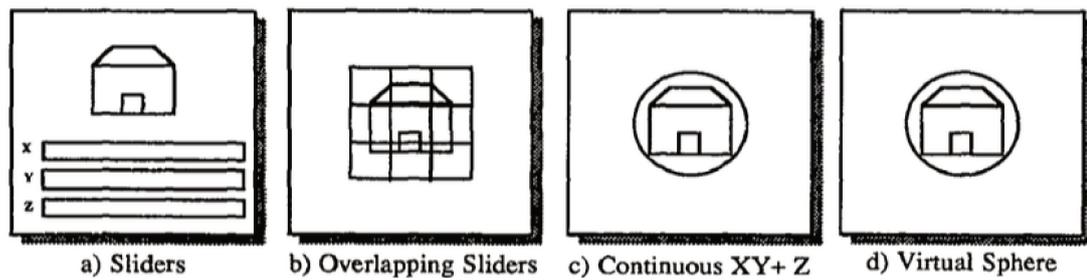


FIGURE 3.27 – Sliders

que Shoemake qualifie d'*hystérésis*. *Arball* (figure 3.34b) résout ce problème grâce à l'utilisation des quaternions.

Tan et al. [2001] ont proposé de coupler la navigation de type *Flying* avec un contrôle orbital afin de permettre à la fois de naviguer dans la scène, d'obtenir un aperçu de celle-ci et de pouvoir examiner de manière précise des objets ciblés dans un contexte bureautique. En plus de permettre de faire de la translation (3.4.2) et de la rotation égocentrique (3.4.3), ce système possède donc un mode de contrôle orbital. Le choix du mode se fait en fonction de l'objet sous le curseur lors d'un clic souris. Si l'utilisateur fait un mouvement de *glisser* sur un espace vide, par exemple le ciel ou le sol, la technique passe en mode navigation. En revanche, elle passe en mode orbital si l'utilisateur fait un mouvement de *glisser* sur un objet. Dans ce mode, l'utilisateur contrôle la distance par rapport au modèle autour duquel il orbite en déplaçant la souris vers l'avant ou l'arrière. En la déplaçant vers la droite ou vers la gauche, l'utilisateur tourne autour du modèle.



(a) Vitesse lente

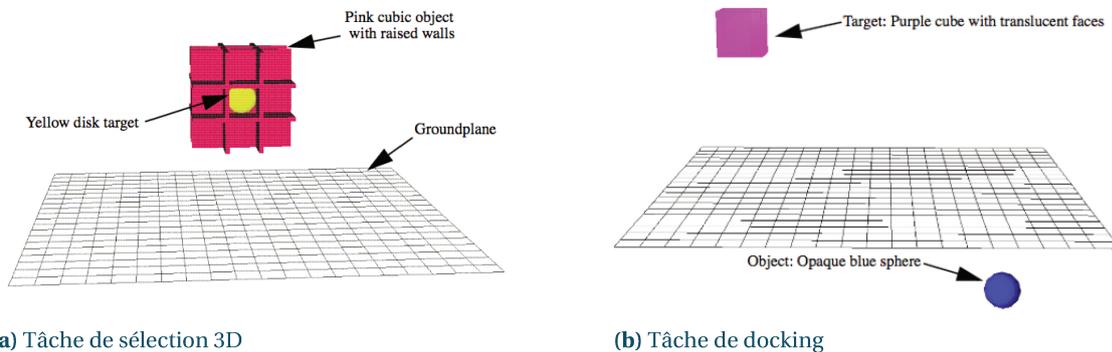
(b) Vitesse élevée

FIGURE 3.28 – La technique *Speed-coupled Flying with Orbit* couple la vitesse de déplacement dans la scène à la hauteur de la caméra. Ainsi, si l'utilisateur se déplace lentement, il se déplace au ras du sol (figure 3.28a). En revanche, avec une vitesse rapide, la caméra offre un aperçu de la scène (figure 3.28b).

Dans *UniCam*[Zeleznik et Forsberg, 1999], des gestes effectués avec une souris permettent de réaliser une grande variété de contrôles de caméra. Parmi ceux-ci, deux mouvements de contrôles orbitaux sont proposés. Le premier permet de faire tourner la caméra par rapport au centre de la vue. Le premier objet entrant en intersection avec le centre de la vue devient le centre de rotation de ce mouvement orbital. Le second contrôle orbital permet de faire tourner la caméra autour d'un point arbitraire situé sur un objet de la scène. Ces contrôles sont effectués en faisant un glisser-déposer de la souris.

3. Contrôle du point de vue  
 3.4. Techniques d'interaction pour le contrôle du point de vue

Balakrishnan et Kurtenbach [1999] ont comparé deux techniques pour effectuer du contrôle de caméra orbital et déplacer des objets dans un contexte bureautique (souris + clavier). La première, qui nécessite une seule main, permet à la fois de manipuler la caméra avec un *glisser* de la souris et la position de la sphère bleue (figure 3.29) dans le plan de la caméra. La seconde technique repose sur une approche bimanuelle. Elle permet de manipuler le point de vue avec la main non dominante, tandis que la main dominante déplace les objets 3D. Ces deux techniques sont comparées sur une tâche de docking. Les résultats ne montrent pas d'effet significatif de la technique sur le temps de réalisation de la tâche. Une explication avancée par les auteurs est que les participants cherchent à contrôler à la fois la rotation de la caméra et le déplacement de la sphère de manière simultanée avec la seconde méthode, ce qui complexifie la tâche.

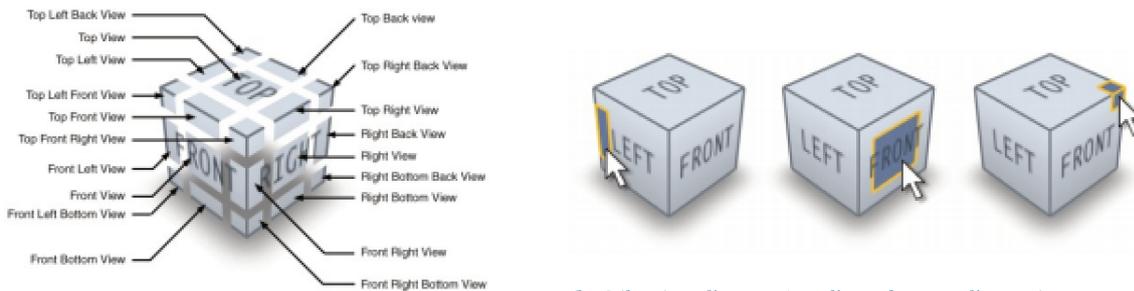


(a) Tâche de sélection 3D

(b) Tâche de docking

**FIGURE 3.29** – À gauche, l'utilisateur contrôle la position de la caméra par rapport au cube rouge grâce à un contrôle orbital opéré avec la main non dominante. Afin de valider l'essai, il doit trouver la cible jaune qui est positionnée dans une des cases du cube. La sélection de la cible se fait avec la main dominante. À droite, l'utilisateur doit placer la sphère bleue dans la cible violette positionnée dans le coin de la scène diamétralement opposé.

Certains travaux proposent un composant graphique spécial pour le contrôle de la rotation orbitale.



(a) Les 26 points de vue du *ViewCube*

(b) Sélection d'une arête, d'une face ou d'un coin

**FIGURE 3.30** – À gauche, représentation des 26 points de vue du *ViewCube* [Khan et al., 2008], un élément d'interface pour le contrôle orbital et la visualisation de l'orientation de la scène dans l'espace. À droite, sélection d'une arête, d'une face ou d'un coin. Après sélection, la scène transitionne vers l'orientation sélectionnée.

Par exemple, le *ViewCube* [Khan et al., 2008] est un composant graphique servant à la fois de guide et d'interface de contrôle permettant à l'utilisateur de visualiser et de manipuler l'orientation de la scène 3D avec un contrôle orbital. Ainsi, lorsque l'orientation de la scène change, l'orientation du cube change également de la même façon. Ce composant graphique a la forme d'un cube découpé en 26

sous parties (figure 3.30a), dont 8 coins, 12 arêtes et 6 faces. Chacune de ces parties est sélectionnable et actionnable par l'utilisateur. L'utilisateur peut soit utiliser le cube en utilisant un mouvement de *glisser* de la souris, soit cliquer directement sur l'une des vues du cube (figure 3.30b). Lorsque l'utilisateur manipule le cube sans utiliser l'un des points de vue définis, la sous-partie du cube correspondant à la vue courante change de couleur pour lui indiquer comment atteindre cette vue à partir du *ViewCube*. Une expérience comparant les performances des utilisateurs sur une tâche de rotation avec trois techniques de manipulation du point de vue a été menée. Ces trois techniques étaient les suivantes :

- utilisation du *ViewCube* en sélectionnant les points de vue ;
- contrôle de la rotation de la scène en manipulant directement le *ViewCube* à l'aide du *glisser* de la souris (mouvement de type *ArcBall*) ;
- une approche plus classique avec une barre d'outils, représentative des logiciels de modélisation.

Les résultats montrent un net avantage pour la technique *Arcball* avec le *ViewCube* pour un temps moyen de réalisation de la tâche deux fois plus faible par rapport aux deux autres techniques.

Certains travaux utilisent une approche de la rotation orbitale avec laquelle la position de la caméra est contrainte par rapport à la topologie de l'objet.

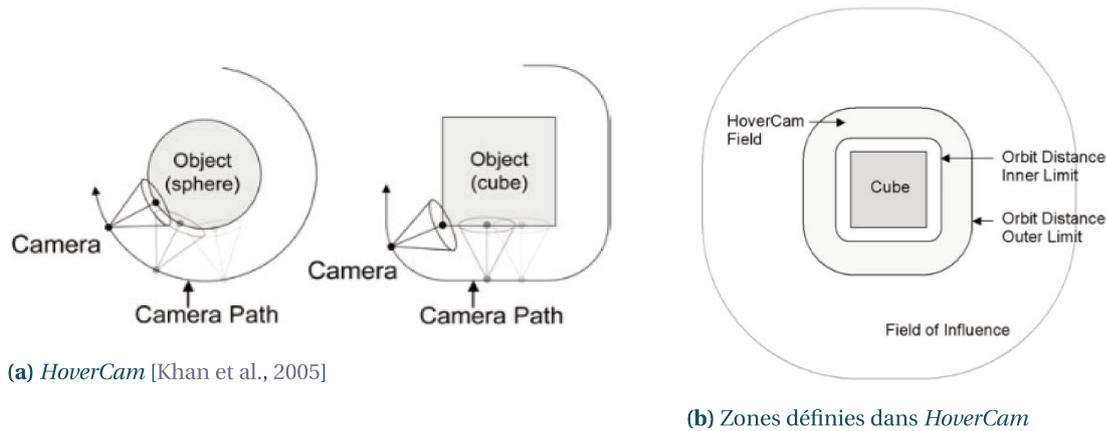
Par exemple, Khan et al. [2005] ont proposé *HoverCam* (figure 3.31) une technique basée sur le contrôle orbital pour l'inspection d'objet 3D. Contrairement à une technique classique de contrôle orbital avec laquelle la rotation du point de vue se fait par rapport au centre d'un objet, la rotation se fait ici par rapport à la topologie de l'objet. Ainsi, la caméra donne l'impression de flotter au-dessus de l'objet (figure 3.31a), sans jamais rentrer en collision avec celui-ci, contrairement au contrôle orbital classique, en particulier avec des objets convexes. Une des particularités de cette technique est d'être activable à tout moment. Les auteurs se sont donc intéressés à la transition entre contrôles standards de caméra et *HoverCam*. Une zone appelée *zone d'influence* (figure 3.31b) est définie autour de l'objet pour gérer plusieurs cas de figure. Tant que la caméra est hors de cette zone d'influence, il n'y a pas de transition vers la technique *HoverCam*. En revanche, si la caméra de point de vue entre dans la zone d'influence, celle-ci est peu à peu attirée vers l'objet et entre dans la *zone HoverCam* où elle devient assujettie à l'algorithme. L'utilisateur peut décider de sortir de ce mode de contrôle à tout moment. Une étude utilisateur a été menée avec des designers afin d'obtenir leurs impressions sur la technique. Bien qu'appréciée dans son ensemble, un inconvénient a été mis en lumière durant cette étude : lorsque la caméra est trop proche des faces de l'objet 3D, celle-ci devient instable et très sensible aux changements d'orientation des faces de l'objet examiné.

Un problème rencontré avec la technique précédente est le manque de stabilité de la caméra lors de l'exploration de modèles 3D ayant une densité très importante de polygones. Ortega et al. [2015] ont résolu les problèmes de stabilité de *HoverCam* (figure 3.32a) en proposant *SHOCam* (figure 3.32b). Cette technique permet également de faire orbiter la caméra autour de groupes d'objets. *ShellCam* proposé par Boubekeur [2014] permet de résoudre les problèmes de stabilité et fonctionne également sur des scènes avec un grand nombre d'objets 3D dynamiques.

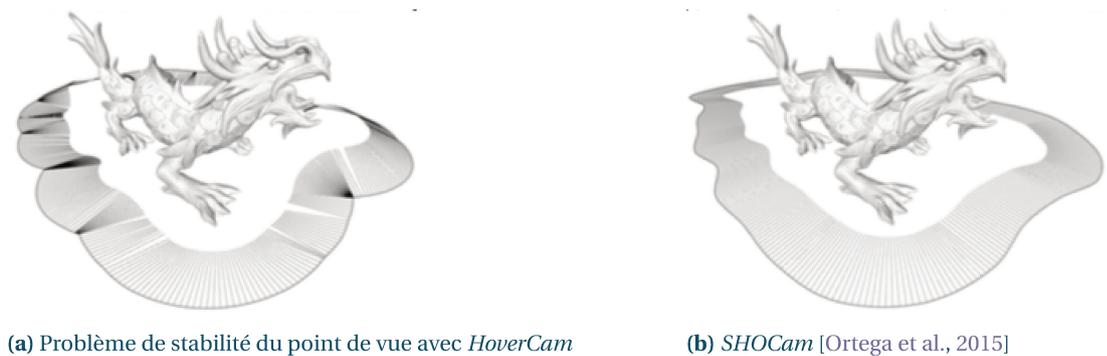
Avec *ScrutiCam*, Declé et al. [2009] présentent une technique pour l'inspection d'objets 3D à partir d'un dispositif d'entrée à deux degrés de liberté. L'approche est différente de celle utilisée dans *HoverCam*, car l'utilisateur précise lui-même le centre de rotation de la technique en ciblant une zone à la surface de l'objet 3D. De plus, cette technique nécessite beaucoup moins de puissance de calcul et peut être

### 3. Contrôle du point de vue

#### 3.4. Techniques d'interaction pour le contrôle du point de vue



**FIGURE 3.31** – À gauche, *HoverCam* [Khan et al., 2005] est une technique pour le contrôle orbital qui se base sur la topologie de l'objet afin de maintenir une distance constante au cours du déplacement et ainsi empêcher les collisions dans le cas d'objets convexes. À droite, les différentes zones de la technique *HoverCam*.



**FIGURE 3.32** – À gauche, la technique *HoverCam* [Khan et al., 2005] souffre de problèmes de stabilité dans certains cas où le modèle exploré possède des coins ou des creux (modèle concave). À gauche, la technique *SHOCam* [Ortega et al., 2015] résout ces problèmes.

utilisée sur des dispositifs aux ressources plus modestes tels que des tablettes ou des smartphones.

La technique *Automated Camera Control for Drawing* (ACCD) [Ortega et Vincent, 2014] permet de dessiner sur des objets 3D en adaptant le point de vue de manière automatique afin d'éviter les occlusions. Cette technique laisse l'utilisateur se concentrer sur la tâche de dessin plutôt que de devoir alterner entre des tâches d'édition et de contrôle du point de vue. La technique a été évaluée par rapport à deux autres techniques :

- une technique de base à une main utilisant la méthode *Arball* pour le contrôle du point de vue qui nécessite donc d'alterner entre contrôle du point de vue et dessin ;
- une technique bimanuelle utilisant le contrôle du point de vue sur la main non dominante.

La tâche de cette expérience était de dessiner par dessus un tracé existant sur la surface d'un objet 3D. Les résultats ont montré que *ACCD* réduit le nombre de tracés nécessaires pour dessiner la courbe complète par rapport aux deux autres techniques et améliore la précision des participants puisqu'ils n'ont qu'une seule tâche à faire.



**FIGURE 3.33** – Automated Camera Control for Drawing permet de dessiner sur des objets 3D en adaptant le point de vue de manière automatique afin d'éviter toute occlusion.

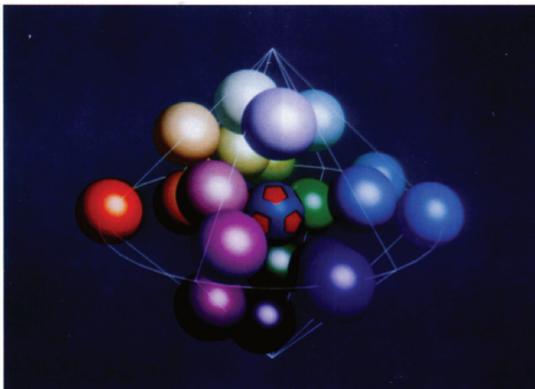
### *Au-delà des dispositifs traditionnels*

#### *Dispositifs augmentés*

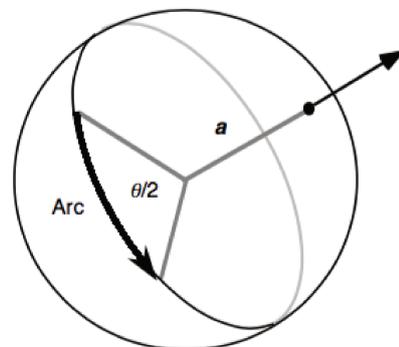
Les premiers travaux mentionnant le contrôle orbital remontent à ceux de Chung [1992] en 1992, réalisés dans le contexte de tâches de ciblage pour la radiothérapie. Plusieurs techniques de contrôle du point de vue utilisant un casque de réalité virtuelle sont alors comparées sur une tâche de pointage. Parmi sept techniques, trois utilisent des dispositifs à 6 degrés de liberté :

- *Joystick*, permet de faire une rotation sur le modèle selon les trois axes de rotation ;
- *Spaceball*, un dispositif de contrôle isométrique à 6 degrés de liberté permettant ici de faire une rotation sur le modèle selon les trois axes de rotation ;
- *6-D Mouse*, un dispositif d'entrée à 6 degrés de liberté intégré dans une boule de billard que l'utilisateur peut manipuler directement grâce à une métaphore de type *Scene in Hand*.

Une expérience comparant ces différentes techniques a été menée. Le but de la tâche était de cibler à l'aide d'un rayon une sphère donnée parmi un ensemble de sphères (figure 3.34a). Les résultats montrent que le joystick obtient les meilleures performances pour les techniques sans suivi de tête.



(a) Exemple de tâche



(b) La technique *Arcball*

**FIGURE 3.34** – La tâche de l'utilisateur consiste à envoyer un rayon vers la sphère centrale (bleue et rouge) représentant une tumeur tout en évitant le plus possible d'entrer en intersection avec les sphères voisines représentant les organes du patient

### **Mouvements du corps**

**Tête.** Chung [1992] a également comparé des techniques pour le contrôle de la rotation exocentrique utilisant le suivi de la tête. Ces techniques sont les suivantes :

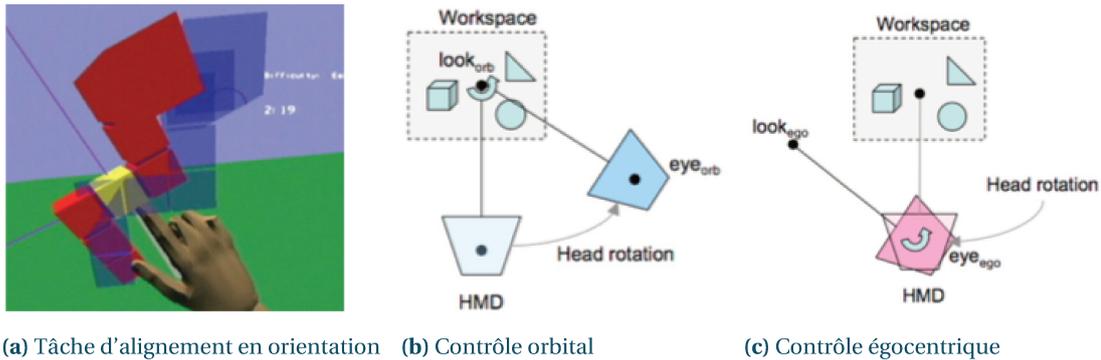
- *Walkaround* est basée sur le déplacement physique de l'utilisateur. La direction du rayon est directement liée à celle du regard de l'utilisateur. L'utilisateur a également la possibilité d'utiliser un dispositif d'entrée pour la translation verticale du modèle afin de pouvoir cibler celui-ci par dessus ou par en dessous ;
- *Walkaround/Rotation* est une technique semblable à la précédente dans laquelle l'utilisateur peut faire tourner le modèle selon les trois axes de rotation en utilisant un dispositif d'entrée ;
- *Orbital*, le regard de l'utilisateur est en permanence centré sur la position centrale du modèle. La rotation de la tête de l'utilisateur est utilisée pour tourner autour de l'objet ;
- *Immersion*, l'utilisateur est placé au centre du modèle et visualise celui en regardant vers l'extérieure afin de trouver la meilleure ouverture pour faire passer le rayon.

Les résultats montrent que la technique *Orbit* offre les meilleures performances par rapport aux autres techniques utilisant le suivi de la tête.

En 1996, Koller et al. [1996] ont décrit une technique d'interaction permettant de visualiser une scène 3D sous différentes perspectives. Cette technique basée sur le suivi de la tête permet à l'utilisateur de manipuler la caméra de point de vue, de sorte que celle-ci "orbite" autour d'un point central de la scène. L'utilisateur peut contrôler la position du centre de rotation ainsi que le rayon de l'orbite (distance entre le centre de rotation et la caméra de point de vue). Une correspondance parfaite (1 :1) entre les mouvements de tête de l'utilisateur et les contrôles de caméra a été utilisée.

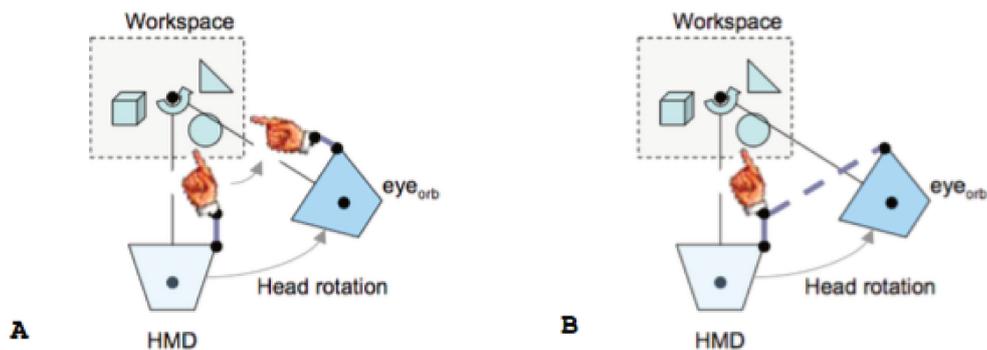
Frees et Lancellotti [2011] se sont intéressés au déplacement d'objets 3D dans un contexte où l'utilisateur contrôle le point de vue orbital (figure 3.35b) avec la rotation de la tête et la position des objets dans la scène 3D avec ses mains. L'orientation de la tête est obtenue avec un casque de réalité virtuelle et la position de la main avec un capteur magnétique *Polhemus FASTRAK*. Une étude utilisateur a d'abord comparé les performances sur une tâche d'alignement (orientation seulement, figure 3.35a) entre un contrôle du point de vue égocentrique et un contrôle du point de vue orbital. Aucun résultat significatif n'a été obtenu au cours de cette étude. Cependant qualitativement, les participants semblent avoir préféré le contrôle du point de vue orbital pour son confort d'utilisation.

La suite du travail s'intéresse à l'inconsistance entre l'utilisation d'un point de vue orbital avec le déplacement d'objets 3D. En effet, contrairement au contrôle égocentrique, le contrôle du point de vue orbital provoque un déplacement dans la scène (translation et orientation par rapport au point central de la scène, voir figure 3.35b). Les mains restent fixes dans le monde réel alors que leur position devrait rester dans le même référentiel que la tête de l'utilisateur dans le monde virtuel. Ceci entraîne donc une inconsistance entre le monde virtuel et le monde réel. L'approche proposée par les auteurs (figure 3.36) est de déplacer la représentation virtuelle de la main dans le repère de la tête seulement dans le cas où aucun objet n'est en cours de déplacement par l'utilisateur. Dans le cas contraire, la position des mains reste inchangée pendant que l'utilisateur déplace la tête. Si l'utilisateur relâche l'objet, la main virtuelle revient dans le référentiel de la tête de l'utilisateur. Dans une dernière partie, les auteurs présentent une technique offrant un contrôle mixte combinant rotation orbitale et égocentrique selon une variable *VC*. Pour  $VC = 1$ , l'utilisateur opère un contrôle orbital pur sur la scène, tandis qu'il opère un contrôle



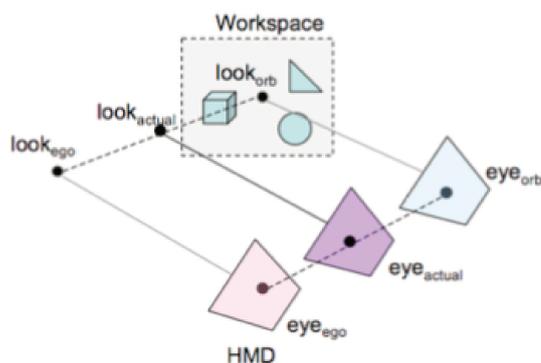
**FIGURE 3.35** – À gauche, l'utilisateur doit réaliser une tâche d'alignement d'objets 3D. Le participant visualise la position de sa main dans la scène grâce au capteur magnétique *Polhemus FASTRAK* dans un environnement immersif grâce à son casque de réalité virtuelle. Au centre, représentation du contrôle orbital. À droite, représentation du contrôle égocentrique.

égocentrique pour  $VC = 0$ . En pratique, une interpolation linéaire est réalisée entre les positions des points de vue égocentrique et orbital afin de déterminer la position du point de vue pour la valeur  $VC$  (voir figure 3.37). Ainsi, pour un taux de 0.5, le point de vue se trouve équidistant entre les points de vue égocentrique et orbital. L'orientation du point de vue reste la même (figure 3.37). La transition du contrôle égocentrique vers le contrôle orbital ou inversement se fait graduellement à la demande de l'utilisateur en appuyant sur un bouton.

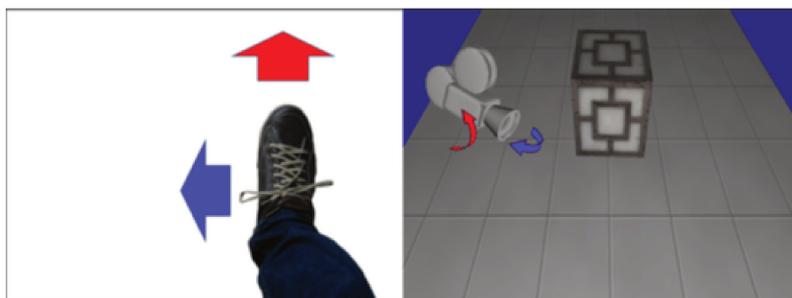


**FIGURE 3.36** – À gauche, déplacement de la main dans le référentiel du point de vue utilisateur si aucun objet n'a été saisi par celui-ci. À droite, si l'utilisateur a saisi un objet dans la scène pour le déplacer, la main reste fixe, tandis que le point de vue tourne autour du point central de la scène.

**Pieds.** *Simeone et al. [2014]* ont proposé d'effectuer des rotations exocentriques à l'aide du mouvement des pieds afin de garder les mains libres. Le pied dominant contrôle la rotation sur le lacet en faisant une translation transversale ( $z$ ), tandis que la rotation sur le tangage est contrôlée en déplaçant le pied à l'horizontale.



**FIGURE 3.37** – Contrôle du point de vue hybride permettant de combiner du contrôle orbital au contrôle égocentrique avec  $VC = 0.5$ .  $eye_{ego}$  représente la position et l'orientation du point de vue égocentrique tandis que  $eye_{orb}$  représente la position et l'orientation du point de vue orbital. Avec  $VC = 0.5$ , la position du point de vue se trouve à mi-chemin entre les points de vue égocentrique et orbital.



**FIGURE 3.38** – Contrôle de la rotation exocentrique à l'aide des pieds. Le déplacement en profondeur du pied contrôle la rotation sur le lacet tandis que le déplacement du pied à l'horizontal contrôle la rotation sur le tangage.

### 3.5 CONCLUSION

Dans ce chapitre nous avons tout d'abord cherché à comprendre comment la perception de profondeur est perçue chez l'être humain et pourquoi celle-ci est nettement réduite sur un écran à deux dimensions. Nous avons vu qu'il existait un ensemble d'indices visuels que nous utilisons dans la vie de tous les jours afin de nous permettre de mieux appréhender notre environnement, en particulier la parallaxe de mouvement et la disparité binoculaire (vision stéréoscopique). Nous avons vu que la parallaxe de mouvement permettait de désambiguïser les relations entre objets et d'améliorer la perception de la profondeur sur des écrans 2D. Ceci montre l'importance de pouvoir contrôler le point de vue facilement et confortablement. Bien que les opinions soient mitigées sur la question, une majorité de travaux s'accordent sur le fait que la parallaxe de mouvement offre de meilleures performances que la vision stéréo. C'est pourquoi nous nous intéressons plus particulièrement à la parallaxe de mouvement dans la suite de ce travail.

Nous avons ensuite étudié les contrôles de caméra existants, défini leur modalités de contrôle et le contexte dans lequel ceux-ci sont utilisés dans les logiciels 3D. Nous avons également vu les travaux

existant dans la littérature académique, que nous avons classifiés en fonction des contrôles de caméra effectués et des modalités utilisées.

Un grand nombre de travaux académiques [Balakrishnan et Kurtenbach, 1999][Koller et al., 1996] soulignent l'importance du contrôle de la rotation exocentrique afin de pouvoir obtenir un aperçu global de la scène. De plus, nous avons vu que la tête était souvent utilisée pour interagir intuitivement avec la scène 3D en répliquant la parallaxe de mouvement. Certains travaux ([Frees et Lancellotti, 2011], [Koller et al., 1996], [Poupyrev et al., 1999]) ont par ailleurs utilisé la tête comme modalité pour le contrôle de la rotation exocentrique.

Dans le chapitre suivant, nous nous appuyons donc sur le couplage de la tête pour le contrôle de la rotation exocentrique et présentons un ensemble d'études préliminaires à l'élaboration d'une technique de contrôle du point de vue.



---

## CONTRÔLE DU POINT DE VUE : PRÉCISION UTILISATEUR ET MATÉRIEL

### 4.1 APPROCHE

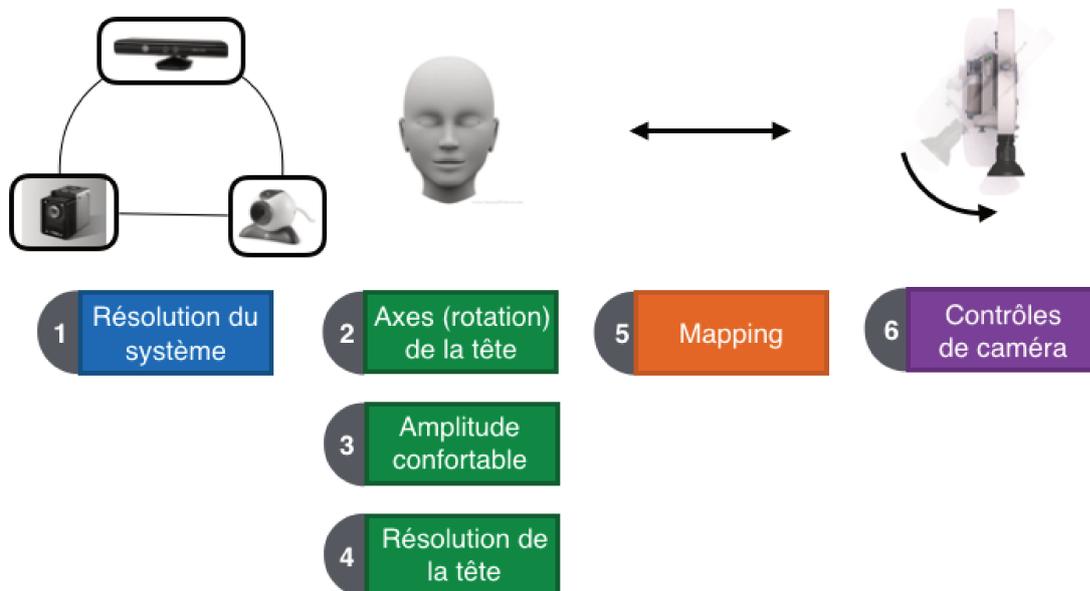
Comme nous l'avons vu précédemment, interagir avec une scène 3D sur un écran 2D nécessite de contrôler le point de vue en permanence. En effet, dans les logiciels de modélisation, l'utilisateur doit régulièrement alterner entre des tâches de contrôle du point de vue et des tâches d'édition. Dans la suite, nous différencions les tâches primaires, liées à la création de contenu 3D, des tâches secondaires telles que le contrôle du point de vue. Prenons l'exemple d'un objet 3D nécessitant d'être déplacé hors du champ de vision de la caméra, par exemple derrière un autre objet de la scène. L'utilisateur doit d'abord contrôler le point de vue de la caméra, puis déplacer l'objet dans la bonne direction et recommencer jusqu'à ce que celui-ci soit correctement positionné. Notre but est de réduire le nombre d'étapes pour réaliser ces opérations et ainsi permettre aux utilisateurs de rester concentrés sur leurs tâches primaires. À cette fin, nous proposons d'utiliser les mouvements de la tête pour contrôler le point de vue, ce qui permettra à l'utilisateur de garder les mains libres pour réaliser les tâches primaires et de ce fait, d'augmenter la bande passante interactionnelle.

L'étude des travaux existants et des logiciels 3D en ce qui concerne le contrôle du point de vue, montre l'importance du contrôle orbital pour permettre aux utilisateurs d'observer une scène 3D et de comprendre les relations entre les objets la constituant. C'est d'ailleurs ce type de contrôle de rotation qui est utilisé par défaut dans la majorité des logiciels de modélisation 3D (chapitre 3). Le tracé de courbes en 3D pouvant être assimilé à une tâche de modélisation, le contrôle orbital nous semble donc particulièrement adapté. Nous souhaitons donc à la fois permettre à l'utilisateur de réaliser des mouvements de rotation rapides et précis afin de jeter un coup d'oeil rapide, mais aussi d'effectuer des mouvements de rotation complets de 360° afin de visualiser l'arrière des objets 3D. Se pose alors la question de savoir quel est l'axe de la tête à utiliser et avec quelle amplitude de mouvement, et quelle fonction de transfert est la plus adaptée pour réaliser ce type de contrôle. Notre approche (voir figure 4.1), qui est constituée de 5 étapes, est la suivante :

1. L'amplitude de rotation de la tête étant limitée du fait des contraintes biomécaniques du corps humain, nous avons besoin de mettre au point une fonction de transfert (que nous détaillons au chapitre suivant) permettant de faire une correspondance appropriée entre les mouvements de

- la tête et le contrôle de la caméra. Pour pouvoir mettre au point cette fonction de transfert, nous avons besoin de connaître plusieurs paramètres explicités ci-après ;
2. Nous devons déterminer quel mouvement de la tête effectuer pour contrôler le point de vue. Nous avons réalisé une étude par élicitation que nous décrivons dans la section 4.3 ;
  3. Il nous faut également mesurer le plus grand mouvement de tête réalisable (section 4.4) afin d'obtenir le facteur d'amplification nécessaire pour atteindre une rotation de 360° ;
  4. Nous devons aussi déterminer le plus petit mouvement réalisable par l'utilisateur de manière contrôlée (section 4.5) ;
  5. Enfin, pour réaliser ces mesures, nous avons besoin d'un système ayant une bonne résolution théorique ce qui nécessite de vérifier sa précision par rapport à la résolution effective de la tête (section 4.2).

La figure 4.1 illustre l'approche que nous avons suivie.



**FIGURE 4.1** – Afin de pouvoir manipuler le contrôle orbital (6) en utilisant les mouvements de la tête, plusieurs étapes sont nécessaires. Pour opérer une rotation de 360° dans la scène 3D, il faut passer par une fonction de transfert (5). Afin de déterminer cette fonction de transfert, il faut déterminer l'axe de la tête à utiliser (2), le plus grand mouvement confortable réalisable avec la tête (3) et la résolution minimale de l'utilisateur (4). Dans un premier temps, il faut tout d'abord déterminer la résolution du système de suivi (1).

#### 4.2 MESURE DE BRUIT DU DISPOSITIF ART ET RÉOLUTION EFFECTIVE DE LA TÊTE

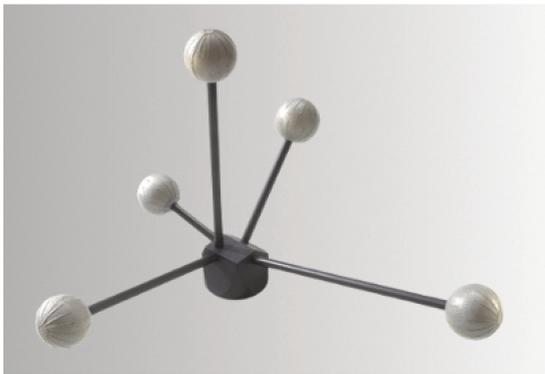
Afin d'avoir suffisamment de précision pour nos études, nous avons choisi un système de tracking ART<sup>1</sup> (Advanced Realtime Tracking) constitué de huit caméras infrarouges (voir figure 4.2b). Il fallait cependant vérifier que celui-ci était suffisamment précis pour mesurer les mouvements de l'utilisateur.

1. <http://www.ar-tracking.com/home/>

Nous avons donc évalué la précision du système (la plus petite rotation mesurable par le système), puis nous avons comparé la précision du système à la résolution effective de la tête de l'utilisateur, à partir du bruit de celle-ci au repos. Il fallait en effet s'assurer que la précision du système de tracking était supérieure à celle de la tête pour pouvoir l'utiliser dans la suite de notre étude.

#### 4.2.1 Précision du dispositif ART

Afin de mesurer la précision de notre système ART, nous avons utilisé la méthode décrite dans [Bérard et al., 2011] consistant à enregistrer pendant un intervalle de temps les données d'un dispositif et prendre comme mesure de la résolution, 4 écarts types par rapport à la moyenne dans la distribution des valeurs ( $4\sigma$ , soit un degré de confiance de 99.99%). Nous avons donc mesuré pendant une heure, la position et la rotation d'une cible ART de type *Tree* (ART TT3, figure 4.2a) positionnée de manière stable sur une table. La figure 4.3b montre l'évolution du bruit pour la rotation sur le lacet (rotation verticale). Nous avons observé un bruit global correspondant à un décalage sinusoïdal ( $\approx 0.015^\circ$ ) au cours du temps sur les deux axes lacets ( $y$ ) et roulis ( $x$ ) (figure 4.3a).



(a) Une cible ART de type Tree Target



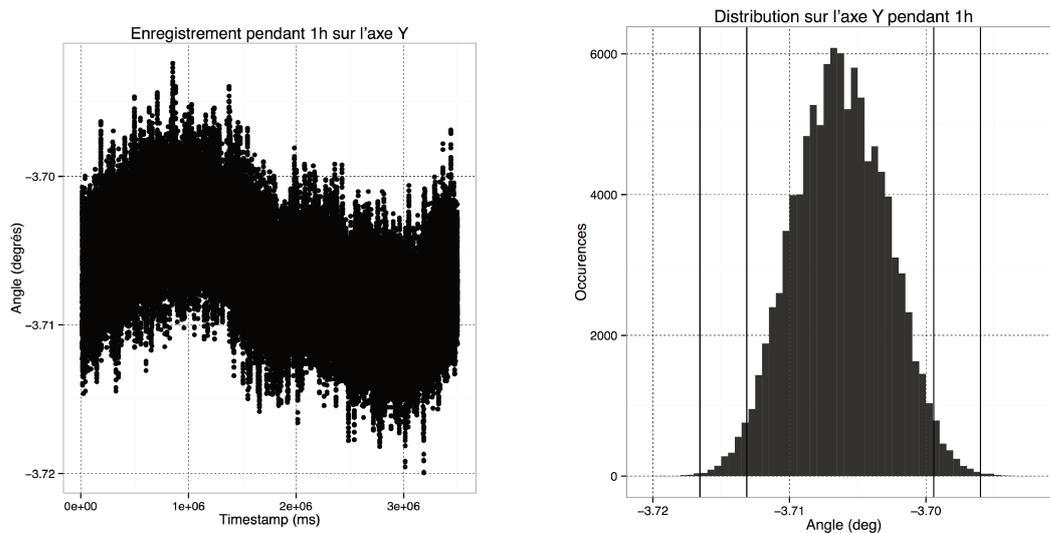
(b) Une des neuf caméras infrarouges du système ART

**FIGURE 4.2** – Nous utilisons le système de tracking infrarouge ART. À gauche, une cible à marqueurs passifs de type Tree Target que nous utilisons tout au long de nos expériences. À droite, une des neuf caméras infrarouges de notre système.

À partir de ces données, selon le principe exposé précédemment, nous avons défini un intervalle de confiance correspondant à un écart par rapport à la moyenne de 4 écarts types ( $4\sigma$ ). Cet intervalle correspond à  $0.05^\circ$  sur les axes de rotation du dispositif. Nous appelons cette résolution, la *résolution du dispositif* ou  $R_{dispositif}$ . Il fallait maintenant vérifier si la *résolution du dispositif* était effectivement supérieure à celle de la tête.

#### 4.2.2 Résolution effective de la tête

Afin de mesurer le bruit de mouvement de la tête au repos, nous avons réalisé une étude utilisateur avec cinq participants issus de notre entourage académique. Celle-ci consistait à placer une cible (identique à celle utilisée pour mesurer la résolution du dispositif) sur la tête de l'utilisateur à l'aide d'un casque (figure 4.4b) en lui donnant comme instruction de rester aussi immobile que possible pendant une minute pendant que nous enregistrions les variations de rotation de sa tête. Un retour visuel sous forme de jauge (voir figure 4.4a) informait le participant afin qu'il garde la tête centrée par rapport à sa position de repos. Nous appelons le bruit de la tête au repos la *résolution effective* de la tête  $R_{eff}^{tete}$ , à savoir



(a) Évolution du bruit global, montrant l'oscillation des valeurs enregistrées au cours du temps de manière périodique. (b) Distribution du bruit global. Les barres verticales correspondent respectivement à  $2\sigma$  et  $3\sigma$  d'écart par rapport à la moyenne.

**FIGURE 4.3** – Bruit global sur une heure d'enregistrement pour la rotation sur l'axe y

l'amplitude de mouvement réalisée par un utilisateur au repos du fait des tremblements involontaires. En prenant l'amplitude de mouvement réalisée par intervalle de 1 seconde, nous trouvons  $R_{eff}^{tete} = 0.2^\circ$ , une valeur qui est 4 fois plus large que la résolution de notre dispositif ART ( $R_{dispositif} = 0.05^\circ$ ). Le système est donc suffisamment précis pour l'utiliser dans la suite de nos travaux.



(a) Jauge de calibration utilisée pour permettre à l'utilisateur de rester centré par rapport à sa position de repos initiale. (b) Cible ART montée sur un casque.

**FIGURE 4.4** – À gauche, la jauge permet au participant de visualiser la rotation de sa tête et de se recentrer le cas échéant. À droite, nous utilisons un casque de chantier sur lequel nous avons fixé la cible ART afin d'avoir un support rigide pour nos expériences.

### 4.3 ÉLICITATION : CONTRÔLE DE CAMÉRA À PARTIR DE LA TÊTE ET DES ÉPAULES

Nous avons vu que les mouvements de la tête ont été déjà utilisés pour le contrôle du point de vue, notamment pour la navigation 3D ([Fuhrmann et al., 1998, Terziman et al., 2010]) et la rotation exocentrique ([Koller et al., 1996, Ortega et Vincent, 2014]). En nous appuyant sur le contrôle de la rotation

exocentrique à l'aide de la tête, nous avons étudié comment associer le contrôle orbital de caméra aux mouvements de la tête. Nous en avons également profité pour étudier les contrôles de caméra mis en évidence dans le chapitre précédent et comprendre si tous ces contrôles peuvent être effectués avec cette modalité en utilisant des gestes différents. Nous nous intéressons ici aux quatre mouvements de caméra les plus communément utilisés qui ont été définis dans le chapitre précédent : la translation, le zoom, la rotation égocentrique et la rotation exocentrique. Afin de contrôler ces 4 mouvements de caméra, nous avons besoin d'un certain nombre de degrés de liberté :

- Translation : 3 degrés de liberté ;
- Rotation exocentrique : 3 degrés de liberté ;
- Rotation égocentrique : 3 degrés de liberté ;
- Zoom : 1 degré de liberté.

On voit rapidement que la tête à elle seule ne peut pas être utilisée pour tous ces contrôles de manière simultanée. L'utilisateur étant assis face à son écran, nous avons proposé d'utiliser en plus les épaules afin d'étendre la bande passante interactionnelle. Cette partie du corps étant proche de la tête, cette approche permet de rester compatible avec des dispositifs de suivi de type webcam. Nous avons conduit une étude afin de mieux comprendre quel vocabulaire de gestes utiliser.

#### 4.3.1 Espace de conception : tête + épaules

Nous avons défini un espace de conception basé sur les mouvements de la tête et des épaules dans lequel nous avons identifié six mouvements pour la tête et six pour les épaules. Ces mouvements sont illustrés à la figure 4.5 :

- Tête :
  - lacet, la rotation de la tête sur l'axe vertical ;
  - tangage, la rotation de la tête sur l'axe horizontal ;
  - roulis, la rotation de la tête sur l'axe transversal ;
  - translation verticale ;
  - translation horizontale ;
  - translation transversale.
- Épaules :
  - lacet, la rotation du buste sur l'axe vertical ;
  - tangage, la rotation du buste sur l'axe horizontal ;
  - roulis, la rotation du buste sur l'axe transversal ;
  - élévation de l'épaule droite ;
  - élévation de l'épaule gauche ;
  - élévation des deux épaules simultanément.

#### 4.3.2 Étude utilisateur

Nous avons réalisé une étude utilisateur afin de déterminer les mouvements de la tête et des épaules les plus appropriés pour contrôler le point de vue dans le contexte du tracé et de l'édition de courbes 3D. Nous avons utilisé pour cela une approche du type *user-defined gestures* [Wobbrock et al., 2009] qui est basée sur la présentation des effets d'une action aux participants et l'élicitation par ceux-ci des causes ayant menées à cette action. Dans notre étude, le participant devait visionner une vidéo représentant le

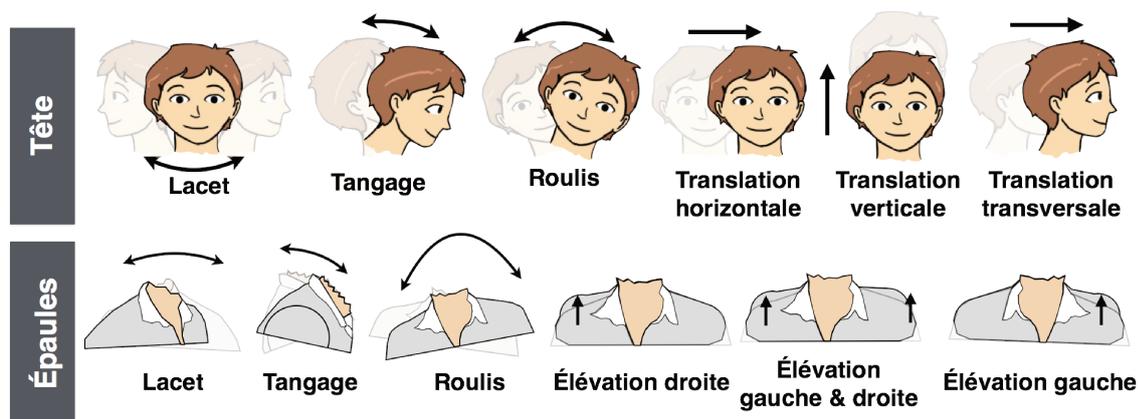


FIGURE 4.5 – Espace de conception des mouvements de la tête et des épaules pour le contrôle de caméra

contrôle de caméra que nous souhaitons opérer et proposer le mouvement de la tête et des épaules qui lui aurait permis de réaliser ce contrôle parmi les mouvements proposés dans notre espace de conception.

### Participants

Nous avons recruté dix participants au sein de notre laboratoire (6 hommes et 4 femmes) âgés de 24 à 33 ans ( $x = 26.4$  ;  $\sigma = 2.8$ ).

### Appareillage

Durant l'expérience, les participants étaient assis sur une chaise de bureau standard, afin de leur permettre de tourner facilement le buste. Les utilisateurs disposaient d'un écran de 15 pouces, d'un clavier et d'une souris. Nous avons mené cette étude dans une pièce munie du système de capture infrarouge "A.R.T."<sup>1</sup> (section 4.2). Des marqueurs passifs étaient positionnés sur les épaules et la tête des participants afin de pouvoir obtenir en temps réel la position de celles-ci dans l'espace de la salle. Une caméra permettait également de filmer les participants afin de garder une trace des mouvements effectués pour une analyse ultérieure.

### Procédure

À chaque essai, les participants commençaient par regarder la vidéo représentant le contrôle de caméra (voir la figure 4.7 pour un exemple de vidéo du mouvement exocentrique). Cette vidéo était répétée en boucle jusqu'à ce que le participant propose son mouvement. À tout moment celui-ci pouvait appuyer sur une touche du clavier pour démarrer l'enregistrement et appuyer de nouveau pour l'arrêter. Pendant l'étude, les participants étaient encouragés à réfléchir à haute voix afin d'expliquer leur choix.

1. [www.ar-tracking.com](http://www.ar-tracking.com)

### Design expérimental

Quatre contrôles de caméra (figure 4.6) ont été présentés aux participants sous forme de vidéo :

- *translation droite/gauche et haut/bas* ;
- *zoom avant/arrière* ;
- *rotation égocentrique* sur le lacet et le tangage ;
- *rotation exocentrique* sur le lacet et le tangage (orbite).

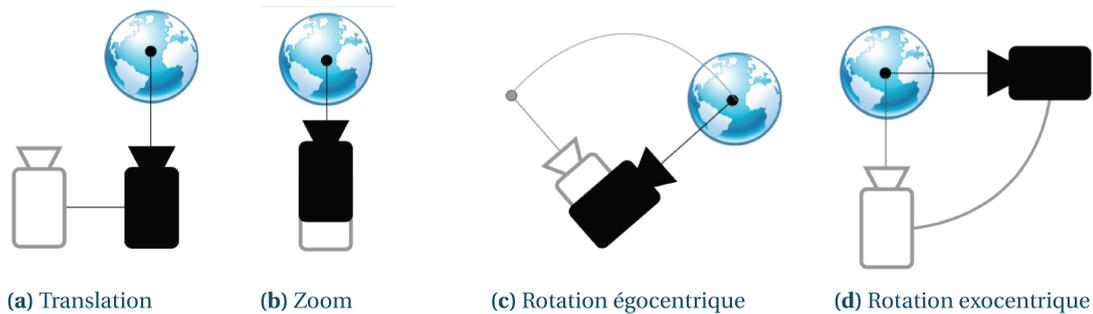


FIGURE 4.6 – Quatre contrôles de caméra différents

L'ordre des contrôles de caméra était contrebalancé à l'aide d'un carré latin afin que chaque participant visualise les contrôles de caméra dans un ordre différent. Chaque essai était répété 5 fois. Au total, 14 contrôles différents (zoom avant/arrière, translations droite/gauche/haut/bas, rotations exocentriques droite/gauche/haut/bas, rotations égocentriques droite/gauche/haut/bas) devaient être effectués par chaque participant, soit 70 essais en prenant en compte les répétitions.

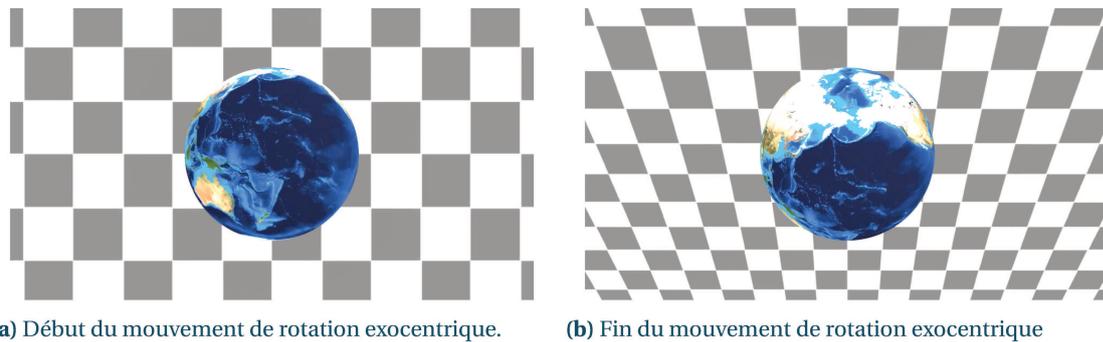
#### 4.3.3 Résultats & analyse

Nous avons tracé l'évolution des mouvements de la tête et des épaules dans l'espace et la distribution des positions de la tête et des épaules. Pour chaque mouvement proposé, nous disposons donc d'une vue de face, de haut et de côté (figure 4.8). À l'aide de celles-ci et des enregistrements vidéo, nous avons classifié les propositions des participants pour chaque contrôle de caméra par similarité.

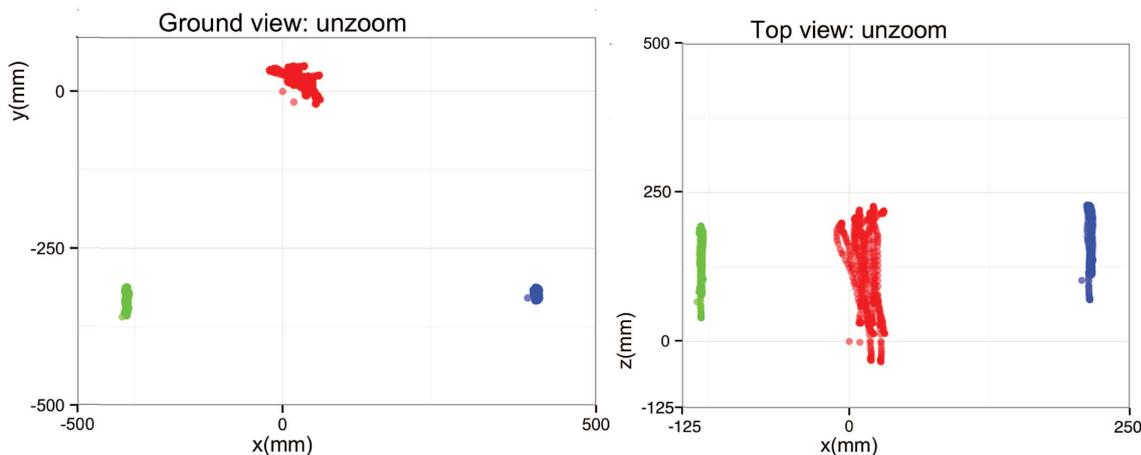
Nous avons ensuite utilisé la formule<sup>1</sup> de Wobbrock et al. [2009] (figure 4.9) pour mesurer le taux de consensus de chaque groupe de proposition. Cette formule permet de calculer le niveau de consensus pour chaque contrôle  $c$  en fonction du nombre de gestes proposés pour le contrôle  $c$  ( $|P_c|$ ) et du nombre de gestes dans le groupe  $g$  ( $|P_g|$ ). La figure 4.10 représente le niveau de consensus obtenu pour chaque contrôle.

Nous avons classé les contrôles de caméra en fonction du niveau de consensus obtenu. Les stratégies majoritaires observées sont les suivantes :

1. Cependant, certains travaux [Tsandilas et Dragicevic, 2016] ont montré que cette formule peut comporter un biais dans le cas où les participants n'arrivent pas à se décider (dû à des instructions peu claires, aucun geste intuitif apparent, etc.). Faute de pouvoir faire un choix précis parmi les options proposées, le participant peut aller jusqu'à faire un choix arbitraire.



**FIGURE 4.7** – Présentation de la vidéo du mouvement de rotation exocentrique par rapport à l’axe horizontal aux participants. À gauche, le début du mouvement. À droite, la fin du mouvement après que la caméra ait opéré une rotation sur l’axe horizontal en se déplaçant vers le haut.



**FIGURE 4.8** – À gauche, mouvement de zoom arrière proposé par l’un des participants en vue de face. Les points rouges représentent la position de la tête dans l’espace pendant le mouvement et les points verts la position des épaules. À droite, mouvement de zoom arrière proposé par l’un des participants en vue de haut.

$$A_c = \sum \left( \frac{|P_g|}{|P_c|} \right)^2$$

**FIGURE 4.9** – Équation de Wobbrock pour la mesure du consensus entre utilisateurs.

### Zoom

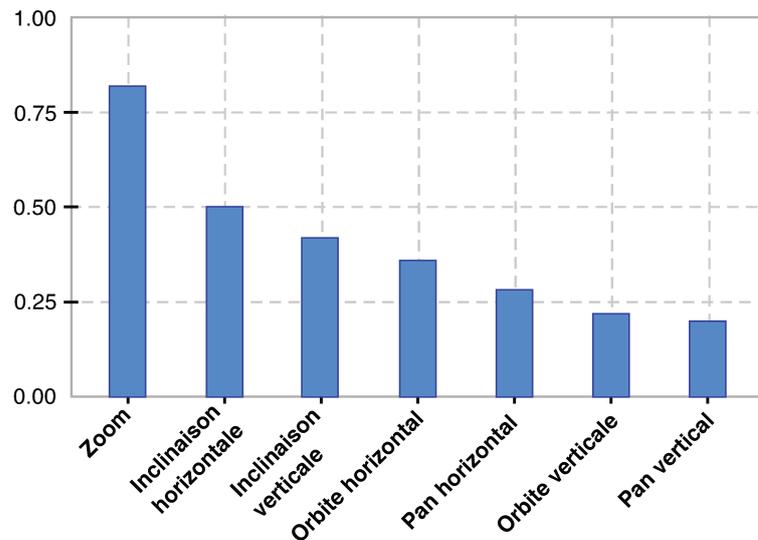
On observe un fort niveau de consensus pour le *zoom*, 9 utilisateurs sur 10 se penchant en avant ou en arrière pour respectivement réaliser un zoom avant et un zoom arrière.



Neuf participants ont proposé d’incliner le buste en avant ou en arrière pour respectivement réaliser un zoom avant et un zoom arrière ;



Un participant a proposé de faire une translation du buste vers l’avant ou vers l’arrière pour



**FIGURE 4.10** – Consensus entre utilisateurs pour les mouvements de la tête et des épaules proposés. Les mouvements sur un axe identique, mais de directions opposées ont été regroupés au sein de la même catégorie.

respectivement réaliser un zoom avant et un zoom arrière.

#### *Rotation égocentrique horizontale*

La *rotation égocentrique horizontale* a obtenu un fort niveau de consensus, la moitié des utilisateurs faisant une rotation du buste et l'autre moitié une rotation de la tête en gardant les épaules fixes.



Cinq participants ont proposé de faire une rotation du buste sur l'axe vertical dans le sens horaire ou antihoraire afin de respectivement effectuer une rotation égocentrique vers la droite ou vers la gauche ;



Cinq participants ont proposé de faire une rotation de la tête uniquement (épaules fixes) sur le lacet dans le sens horaire ou antihoraire afin de contrôler respectivement la rotation égocentrique vers la droite ou vers la gauche.

#### *Rotation égocentrique verticale*

La *rotation égocentrique verticale* a obtenu un niveau de consensus plus faible, une majorité de participants proposant d'utiliser le tangage de la tête et le reste proposant des stratégies différentes (translation d'une épaule, translation du buste, ...).



Quatre participants ont proposé un mouvement de rotation de la tête sur le tangage vers le haut pour contrôler la rotation vers le haut. Cependant, six participants ont proposé un mouvement symétrique utilisant la rotation de la tête sur le tangage vers le bas pour effectuer une rotation vers le bas.

### Rotation exocentrique horizontale

Le contrôle de *rotation exocentrique horizontale* a obtenu un niveau de consensus faible. Cependant, les participants ont principalement proposés deux stratégies utilisant soit la rotation des épaules, soit l'inclinaison du buste.



Cinq participants ont proposé d'utiliser la rotation des épaules sur le lacet dans le sens horaire ou antihoraire pour contrôler respectivement la rotation vers la gauche ou vers la droite ;



Trois participants ont proposé d'utiliser l'inclinaison du buste vers la gauche ou vers la droite pour contrôler respectivement la rotation vers la gauche ou vers la droite.

### Translation horizontale

La *translation horizontale* a obtenu un niveau de consensus faible, les participants étant partagés entre utiliser l'inclinaison ou la translation horizontale du buste.



Quatre participants ont proposé d'incliner le buste sur la gauche ou sur la droite pour respectivement faire une translation vers la gauche ou vers la droite ;



Trois participants ont proposé de translater le buste vers la gauche ou vers la droite pour respectivement faire une translation vers la gauche ou vers la droite.

### Rotation exocentrique verticale

La *rotation exocentrique verticale* a obtenu un niveau de consensus plus faible que la rotation exocentrique horizontale du fait d'un grand nombre de stratégies proposées. Les deux stratégies principales utilisaient soit le tangage, soit une translation du buste vers le haut/bas couplée à une translation du buste vers l'avant.



Trois participants ont proposé d'utiliser une inclinaison du buste vers l'avant ou vers l'arrière afin de contrôler respectivement la rotation vers le haut ou vers le bas ;



Trois participants ont proposé d'utiliser une translation du buste vers le haut couplée à une translation du buste vers l'avant ou une translation du buste vers le bas couplée à une translation vers l'avant, afin de contrôler respectivement la rotation vers le haut ou la rotation vers le bas.

### Translation verticale

La *translation verticale* a obtenu un faible niveau de consensus, les participants étant partagés entre l'utilisation de la translation verticale du buste, le tangage et la translation verticale des épaules.



Quatre participants ont proposé de translater le buste vers le haut ou vers le bas pour respectivement faire une translation vers le haut ou vers le bas ;

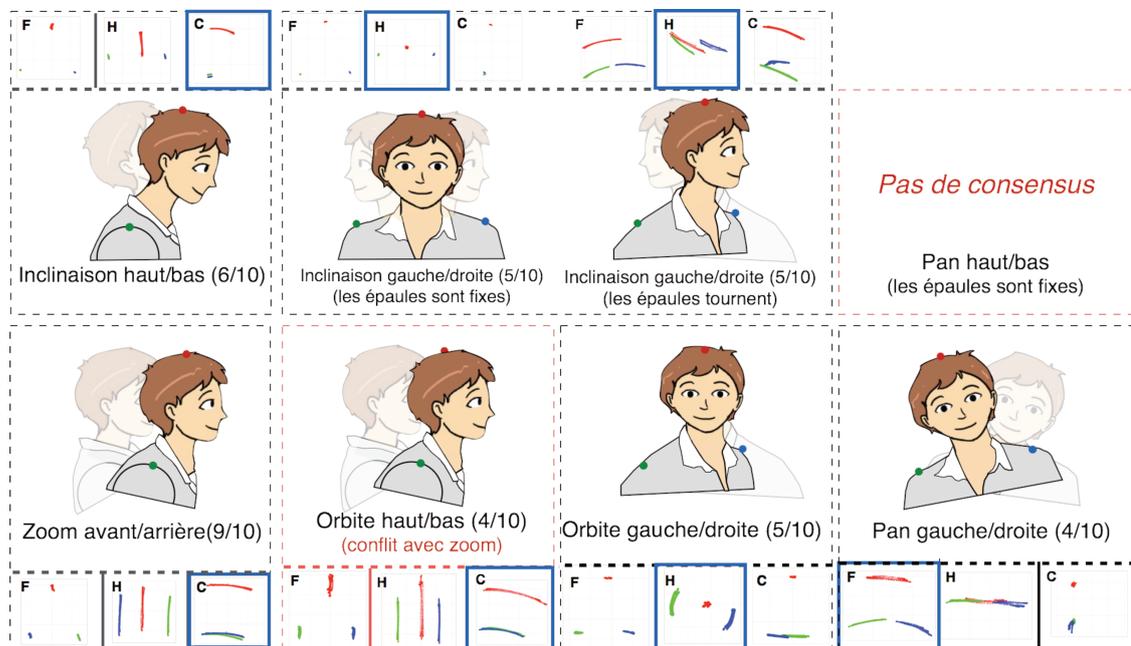


Deux participants ont proposé d'incliner le buste vers l'avant ou vers l'arrière pour respectivement faire une translation vers le haut ou vers le bas. Une variation de cette proposition consistait à ajouter une translation du buste vers le haut/bas à la translation transversale ;

Deux participants ont proposé d'utiliser la translation des épaules vers le haut pour contrôler la translation verticale.



Un récapitulatif de ces résultats est présenté à la figure 4.11.



**FIGURE 4.11** – Mouvements de la tête et des épaules proposés pour chaque contrôle de caméra. Les trois représentations face (F), haut (H) et côté (C) représentent les mouvements de la tête (en rouge) et des épaules (en vert et bleu) enregistrés par le système de suivi infrarouge.

#### 4.3.4 Conclusion

Cette étude nous a permis de mieux comprendre quels mouvements (et amplitudes) de la tête et des épaules étaient utilisés de manière spontanée par les participants pour contrôler la caméra dans une scène 3D. Hormis le contrôle de zoom qui a obtenu un niveau de consensus élevé, les autres contrôles obtiennent un niveau de consensus inférieur ou égal à 50% (voir figure 4.10). En particulier, la rotation exocentrique (rotation orbitale), que nous avons décidé d'étudier plus particulièrement (voir 3.5), obtient un niveau de consensus faible (0.35). Ceci offre davantage de flexibilité pour définir un nouveau type de contrôle dans la mesure où les utilisateurs ne s'attendent pas un comportement particulier.

À la suite de plusieurs sessions de brainstorming pour le contrôle orbital, nous nous sommes arrêtés sur deux modalités : la rotation de la tête sur le lacet et la rotation de la tête sur le roulis. La première semblait pertinente car elle s'appuyait sur le même axe de rotation (l'axe vertical y). La seconde s'est progressivement avérée intéressante, car l'écran reste dans le champ de vision de l'utilisateur tout au long du contrôle. Nous nous intéressons donc dans la suite de ce travail à la rotation orbitale à l'aide de la rotation de la tête sur le lacet et le roulis. En particulier, nous allons mesurer l'amplitude maximale confortable sur chacun de ces axes afin de déterminer le facteur d'amplification à utiliser dans notre fonction de transfert.

Dans les perspectives de ce travail (Chapitre 6), nous reviendrons sur l'utilisation des épaules pour augmenter la bande passante interactionnelle.

## 4.4 AMPLITUDE CONFORTABLE

### 4.4.1 Étude utilisateur

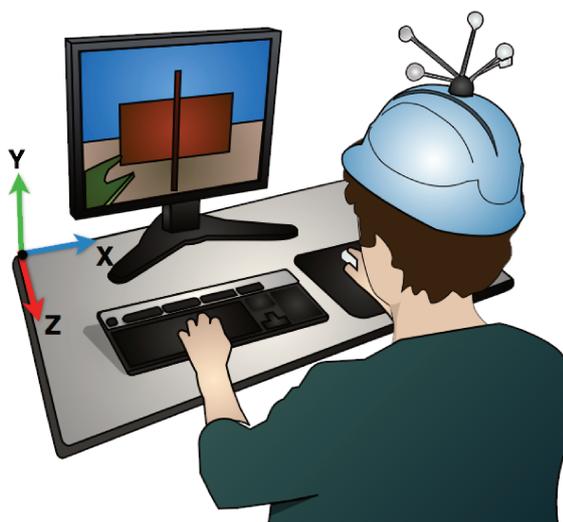
Le but de cette étude était de déterminer l'amplitude maximale de la tête pouvant être opérée par l'utilisateur sur les axes de lacet (y) et de roulis (x) tout en restant dans une position à la fois confortable physiquement et visuellement. Les études en biomécaniques [Thibodeau et Patton, 1996] ont montré que l'être humain est capable de tourner la tête de plus de 70° sur l'axe y (lacet) et de plus de 40° sur l'axe z (roulis). Ces études se sont concentrées sur les capacités biomécaniques mais ne prennent pas en compte les niveaux de confort physique et visuel de l'être humain face à un écran dans un contexte de bureautique. En effet, en fonction de la distance de l'utilisateur par rapport à l'écran et de la taille de celui-ci, certaines parties de l'écran peuvent être hors du champ de vision de l'utilisateur. Nous proposons de mesurer l'amplitude *maximale confortable*, à savoir l'amplitude maximale à laquelle l'utilisateur peut tourner la tête tout en étant dans une position confortable physiquement et visuellement, au travers de deux critères. Le premier, l'*amplitude physique confortable*, impose à l'utilisateur de tourner sa tête à l'amplitude maximale possible à laquelle il ne ressent pas de fatigue. Le second critère, l'*amplitude visuelle confortable*, impose à l'utilisateur de tourner sa tête à l'amplitude maximale possible tout en étant capable de voir l'intégralité de son écran. Dans le but d'évaluer cette amplitude, nous avons conduit une expérience utilisateur simulant une recherche visuelle sur l'intégralité de la surface de l'écran.

### Participants

Nous avons conduit une expérience avec 12 participants (3 femmes), âgés de 23 à 33 ans ( $\mu = 28, \sigma = 2.66$ ) recrutés au sein de notre institution par courrier électronique. À la fin de l'expérience, les participants étaient récompensés avec des friandises.

### Appareillage

Le dispositif expérimental était constitué d'un ordinateur portable de type MacBook Pro branché à un écran externe de 17 pouces, d'un clavier et d'une souris (figure 4.12). Les participants pouvaient utiliser le clavier à l'aide de leur main non dominante et de la souris avec l'autre. L'écran était espacé de l'utilisateur d'une distance de 50 cm afin de suivre les recommandations en matière d'ergonomie [ISO, 2010]. En revanche, le siège du participant ne pouvait pas tourner sur lui-même afin de ne pas biaiser l'expérience.



**FIGURE 4.12** – Le dispositif expérimental était constitué d'un écran externe de 17 pouces relié à un MacBook Pro, d'un clavier et d'une souris.

### Procédure

La tâche consistait à réaliser une recherche visuelle dans une grille de mots. Un stimulus constitué d'un mot de trois lettres était affiché au centre de l'écran. Les participants devaient mémoriser ce mot et par la suite retrouver celui-ci dans une grille de 7 x 7 (figure 4.13b) constituée de mots de 3 lettres jouant le rôle de distracteur, aussi rapidement et précisément que possible. Afin de constituer ces mots, nous avons utilisé les lettres recommandées par le standard *BS 4274-1 :2003* [BSi, 2003]. Ces lettres, facilement discernables les unes des autres, sont utilisées dans les tableaux de Snellen pour mesurer l'acuité visuelle. Nous avons choisi cette tâche car nous voulions être sûr que le participant soit en mesure de voir la totalité de l'écran et nous voulions que cette tâche soit réaliste (validité externe).

Nous avons contrôlé deux facteurs dans cette expérience :

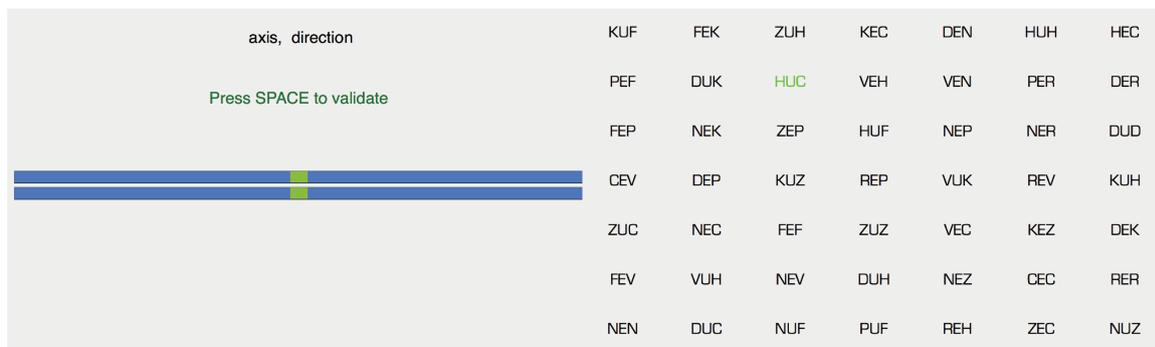
- l'AXE de rotation, qui pouvait être le lacet ou le roulis ;
- la DIRECTION de l'orientation, droite ou gauche.

Comme expliqué auparavant, nous ne prenons pas en compte le tangage dans nos expériences puisqu'aucun des participants ne l'a utilisé de manière spontanée dans notre étude précédente.

La procédure consistait dans un premier temps à demander au participant de placer sa tête dans une position de repos (figure 4.13a) pour calibrer le système de tracking et définir la position d'origine (0°). Le système indiquait ensuite au participant l'axe de rotation sur lequel tourner la tête (lacet ou roulis), ainsi que le sens de la rotation (droite ou gauche). Un mot à mémoriser apparaissait alors au centre de l'écran. Le participant devait mémoriser ce mot, puis tourner la tête selon le bon axe et dans la bonne direction afin d'atteindre l'amplitude maximale tout en prenant en compte son confort visuel (l'utilisateur devait être en mesure de lire n'importe quelle partie de l'écran) et son confort physique (pas de tension musculaire des yeux ni du cou). Lorsque celui-ci considérait qu'il était dans une bonne posture, il pouvait appuyer sur une touche du clavier afin de réaliser la tâche de recherche. L'essai se terminait lorsque le participant sélectionnait l'un des mots.

### Design expérimental

Nous avons utilisé un plan intra-sujets et une analyse à mesures répétées. L'ordre des conditions était contrebalancé entre les participants et chaque condition était répétée 7 fois. Le mot à mémoriser ainsi que sa position étaient déterminés aléatoirement parmi un ensemble de mots et de positions prédéfinis. Pour chaque essai nous avons mesuré l'angle de rotation de la tête lorsque les participants cliquaient sur un mot pour valider l'essai. Au total, le design expérimental comprenait 12 participants x 2 AXES de rotation (lacet ou roulis) x 2 DIRECTIONS (droite ou gauche) x 7 répétitions, soit 336 essais.



(a) Jauge de calibration

(b) Grille de mots

**FIGURE 4.13** – À gauche, avant chaque condition, l'utilisateur doit se placer en position de repos. Les deux jauges représentent l'axe y (lacet) et l'axe z (roulis). Le participant doit aligner le curseur vert dans la zone verte. À droite, l'utilisateur doit trouver le bon mot parmi une liste de distracteurs pour valider l'essai.

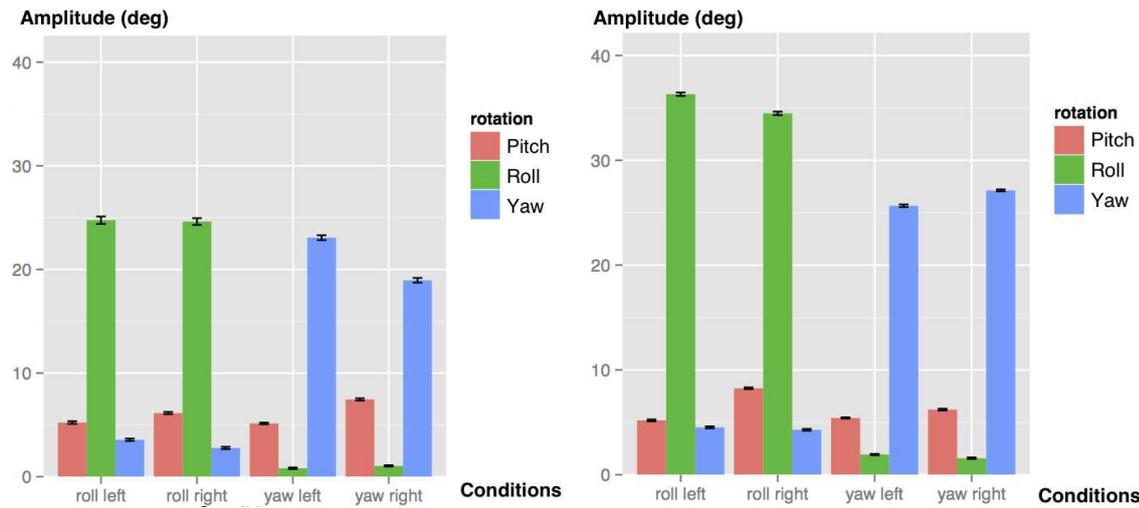
### 4.4.2 Résultats et discussion

Les résultats de l'expérience sont résumés dans la table 4.1. Une ANOVA a révélé un effet significatif de l'AXE sur l'angle ( $F_{1,33} = 26.1, p < 0.0001$ ). Un test post-hoc de Tukey a révélé que les utilisateurs effectuaient des rotations plus larges sur l'axe z (roulis) (35.3°) que sur l'axe y (lacet) (25.8°). Aucun effet de la direction ou d'une interaction DIRECTION x AXE n'a été observé sur les angles. L'ANOVA a confirmé que les conditions n'avaient aucun effet sur la tâche de recherche visuelle (précision et vitesse).

Nous nous attendions à obtenir des valeurs d'amplitudes confortables conformes aux études de biomécaniques [Thibodeau et Patton, 1996], à savoir des angles plus larges sur l'axe y (lacet) par rapport à ceux effectués sur l'axe z (roulis). Cependant, nos résultats montrent un effet opposé dû aux deux critères imposés lors de cette expérience :

- Les participants doivent effectuer une amplitude confortable ;
- Les participants doivent toujours garder l'écran dans leur champ de vision.

Les participants nous ont expliqué qu'avec l'axe y (lacet), l'écran avait tendance à sortir de leur champ de vision avant que ceux-ci puissent atteindre une amplitude physique maximale sur cet axe. En revanche, les participants ne rencontraient pas ce problème sur l'axe z (roulis), l'écran étant en permanence en face de l'utilisateur.



(a) Amplitudes d'angles avant le démarrage de la tâche de recherche (b) Amplitudes d'angles au moment de la tâche de recherche

FIGURE 4.14 – À gauche, les amplitudes d'angles avant le début de l'essai pour les 4 conditions (roulis gauche, roulis droit, lacet gauche, lacet droit). À droite, les amplitudes d'angles au début de l'essai pour les 4 conditions.

Rotation	Amplitude biomécanique (°)	Direction (°)	Angle (°)	CI
Lacet	70	Gauche	25.0	3.7
		Droite	26.5	3.2
Roulis	40	Gauche	36.8	6.9
		Droite	33.9	6.9

TABLE 4.1 – Valeur moyenne et intervalles de confiance à 95% (CI) pour l'amplitude maximale confortable en fonction de la rotation et de la direction.

Nous avons déterminé le plus grand mouvement réalisable par l'utilisateur face à son écran de manière à respecter à la fois le confort physique et le confort visuel. Il nous a fallu ensuite déterminer le plus petit mouvement réalisable par l'utilisateur, ou *résolution utile*.

## 4.5 RÉOLUTION UTILE

### 4.5.1 Étude utilisateur

L'objectif de cette expérience était de déterminer la *résolution utile* de la tête, à savoir, la plus petite amplitude de rotation réalisable de manière contrôlée par l'utilisateur. Cette amplitude se différencie

de la *résolution effective* de la tête ( $R_{eff}^{tête}$ ) que nous avons déterminée auparavant (section 4.2) et qui représente l'amplitude de mouvement réalisée par l'utilisateur au repos à cause des tremblements involontaires. En déterminant la *résolution utile* de la tête, nous cherchons à savoir avec quelle précision l'utilisateur est capable de contrôler l'orientation de sa tête.

### Participants et appareillage

Nous avons recruté 12 participants dans notre environnement académique afin de réaliser cette expérience. Ceux-ci ont reçu une poignée de friandises pour leur participation. Nous avons utilisé un dispositif expérimental identique à celui de l'expérience précédente (voir figure 4.12).

### Procédure

Dans cette expérience, nous avons suivi une méthodologie utilisée par Aceituno et al. [2013] pour déterminer la *résolution utile* de la souris. Cette méthode a été utilisée avec succès auparavant pour déterminer le plus petit mouvement réalisable avec une souris. Nous appliquons ici la même méthode pour déterminer le plus petit mouvement de tête que l'utilisateur peut faire volontairement sur les axes de lacet et de roulis. La méthode utilisée est la suivante :

1. choisir une amplitude maximale  $A$  afin de tester la résolution  $R_A$  ;
2. demander aux participants d'effectuer un mouvement de rotation de la tête inférieur à  $A$ . Répéter cette condition  $k$  fois et calculer le taux de succès  $S_A$  ;
3. choisir une amplitude maximale plus faible  $A$  et retourner au point 2 ;
4. la *résolution utile*  $\hat{R}$  est la résolution  $R_A$  pour laquelle le taux de succès  $S_A$  est supérieur à 95%.



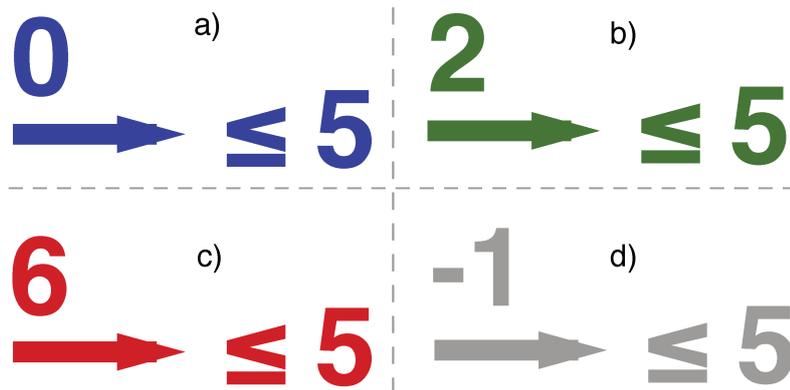
**FIGURE 4.15** – (a) le participant commence par orienter sa tête selon l'axe indiqué dans la bonne direction afin de placer le curseur blanc au sein de l'intervalle jaune. (b) Lorsque le participant est bien placé, l'intervalle devient vert et celui-ci peut appuyer sur la barre espace pour passer à la suite de l'essai.

Comme dans [Aceituno et al., 2013], nous définissons une unité arbitraire, le *count*. La valeur d'un *count* est basée sur celle de la résolution effective de la tête que nous avons déterminée antérieurement. Cette valeur (figure 4.16) correspond à la plus grande valeur entre la résolution du dispositif  $R_{dispositif}$  et la résolution effective de la tête  $R_{eff}^{tête}$ .

Nous définissons donc le *count* grâce à la formule suivante :

$$Count = \max(R_{dispositif}, R_{eff}^{tête}) = \max(0.05, 0.2) = 0.2^\circ$$

**FIGURE 4.16** – La valeur d'un *count* correspond à la valeur maximale entre la résolution du dispositif et la valeur effective de la tête



**FIGURE 4.17** – (a) Le participant est en train de réaliser le déplacement afin de valider l’essai. (b) Le participant vient de valider l’essai. (c) Le participant a échoué à l’essai en dépassant l’amplitude maximale autorisée. (d) L’essai est annulé, car le participant est revenu en arrière sur son mouvement.

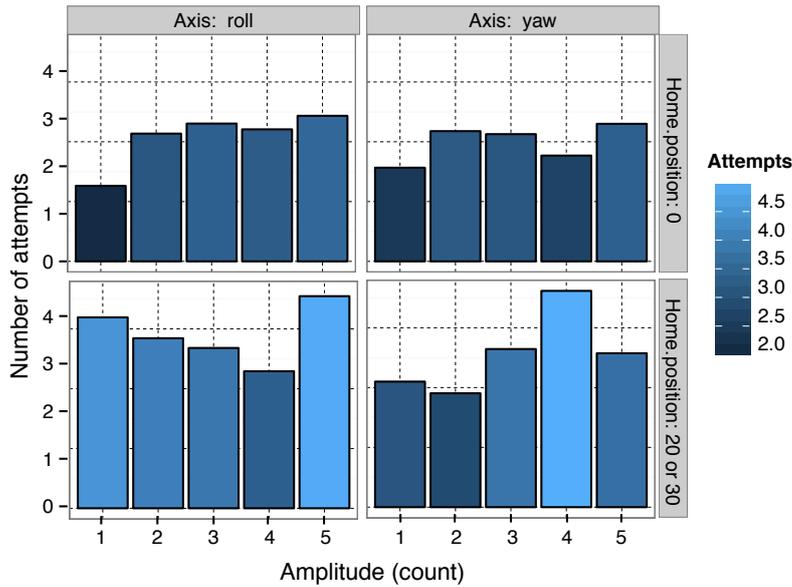
Les participants avaient pour instruction de faire de leur mieux pour rester en dessous de l’amplitude maximale. L’essai démarrait à partir du premier mouvement de tête de l’utilisateur et se terminait si celui-ci dépassait l’amplitude maximale (cas d’échec) (figure 4.17c) ou si l’utilisateur arrivait à rester en position au sein de cet intervalle pendant 1 seconde (essai validé) (figure 4.17b). Tout mouvement dans la direction opposée annulait l’essai et forçait le participant à recommencer. Un changement de direction était détecté lorsque l’utilisateur réalisait un mouvement dans la direction opposée de plus d’une unité (*count*).

### Design expérimental

Nous avons utilisé un plan intra-sujets et une analyse à mesures répétées. L’ordre des conditions était contrebalancé entre les participants et chaque condition était répétée 10 fois. La RÉSOLUTION était présentée en ordre décroissant. En résumé, le design expérimental comprenait 12 participants x 2 AXES de rotation (lacet ou roulis) x 2 ORIENTATIONS INITIALES x 5 RÉSOLUTIONS x 10 répétitions, soit 2400 essais.

**Facteurs.** Dans cette expérience nous avons contrôlé quatre facteurs :

- l’AXE, qui peut être soit le *roulis*, soit le *lacet* ;
- la DIRECTION, qui peut être soit *gauche*, soit *droite* ;
- l’ORIENTATION INITIALE de la tête, car nous pensions que celle-ci pouvait avoir un impact sur la *résolution utile* de la tête à cause des contraintes biomécaniques. Nous avons ainsi défini deux ensembles de valeurs en fonction de l’AXE : 0°, 10° et 20° pour le lacet, et 0°, 20° et 30° pour le roulis. Ces valeurs sont en concordance avec notre expérience précédente : (1) elles sont incluses dans l’intervalle de confort pour chaque axe ; (2) Les rotations sont plus larges sur l’axe z (roulis) que sur le lacet ;
- enfin, nous avons contrôlé également la RÉSOLUTION, à savoir l’amplitude maximale que les utilisateurs ne doivent pas dépasser. Nous avons pris les valeurs suivantes en menant plusieurs pilotes : 1°, 0.8°, 0.6° et 0.4°.



**FIGURE 4.18** – Lorsque l’orientation initiale est nulle, le nombre de tentatives pour réussir l’essai est à peu près constant, mis à part pour une valeur de 1 *count* où les participants sont plus vigilants. En revanche, lorsque l’orientation initiale est différente de 0, le nombre de tentatives semble augmenter avec la précision requise.

**Tâche.** Les utilisateurs devaient effectuer une rotation de la tête à partir d’une orientation initiale dans une direction donnée inférieure à une valeur seuil donnée. Un essai était divisé en trois parties illustrées au niveau des figures 4.15 et 4.17 :

1. le participant commençait par placer sa tête en position initiale à une certaine orientation à l’aide du feedback visuel affiché à l’écran. Pour ce faire il devait déplacer la tête sur le bon axe, de manière à ce que le curseur blanc soit compris dans l’intervalle jaune (figure 4.15) ;
2. après s’être placé en position initiale, l’intervalle sur la jauge devenait vert et le participant pouvait démarrer l’essai en appuyant sur la barre espace du clavier ;
3. le système indiquait ensuite à l’utilisateur la direction dans laquelle se déplacer ainsi que l’amplitude maximale à laquelle il pouvait orienter la tête (figure 4.17).

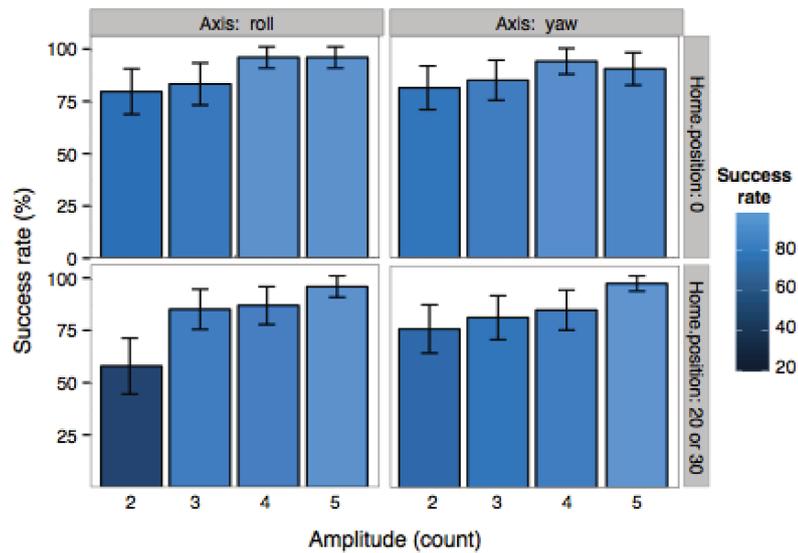
#### 4.5.2 Résultats et discussion

Les résultats sont représentés aux figures 4.18 et 4.19. Une ANOVA a permis de confirmer l’effet significatif de la RESOLUTION sur la précision ( $F_{4,190} = 62.5, p < 0.0001$ ). Cependant, aucun autre effet significatif n’a pu être trouvé. Les résultats montrent que les participants atteignent un taux de succès supérieur à 95% à partir d’une amplitude de 5 *counts*. À partir de ces résultats, nous estimons que la valeur de la *résolution utile* de la tête  $\hat{R}$ , à savoir la plus petite rotation réalisable avec la tête de manière contrôlée pour les axes de lacet et de roulis est :  $\hat{R} = 5 \times 0.2 = 1^\circ$ .

## 4.6 CONCLUSION

Nous avons dans un premier temps mené une étude utilisateur pour vérifier que le système de tracking était suffisamment précis pour mesurer de manière fiable les mouvements de l’utilisateur. Nous avons déterminé d’une part le bruit du système de tracking "A.R.T." ( $R_{dispositif} = 0.05^\circ$ ) et le bruit de l’utilisateur au repos ( $R_{eff}^{tete} = 0.2^\circ$ ) ce qui a permis de montrer que le système de tracking était suffisamment précis pour mesurer les mouvements de l’utilisateur.

**FIGURE 4.19** – Le taux de succès sur l'ensemble des participants en pourcentages sur les deux axes avec les deux orientations initiales et les amplitudes affichées en *count*.



Dans un second temps, nous avons cherché à comprendre comment les utilisateurs associent de manière spontanée un contrôle de caméra à un mouvement de la tête et des épaules, plus particulièrement pour le contrôle de la rotation exocentrique. Nous avons observé que les utilisateurs ne s'attendaient pas à l'utilisation d'une modalité particulière pour ce contrôle. À la suite de plusieurs brainstorming, nous avons décidé de nous concentrer sur deux modalités : la rotation de la tête sur le lacet et sur le roulis. Cette étude a également souligné l'intérêt de l'utilisation des épaules. Nous reviendrons sur ce point dans les perspectives de ce travail (chapitre 6).

Enfin, nous avons mené plusieurs études préliminaires afin de créer une technique d'interaction basée sur les mouvements de la tête (lacet et roulis) pour le contrôle orbital. Nous avons ainsi déterminé l'amplitude de mouvement visuellement et physiquement *confortable* de la tête (en moyenne 25° sur le lacet et 35° sur le roulis) ainsi que la *résolution utile* de la tête (1°), à savoir le plus petit mouvement de tête contrôlable par l'utilisateur face à son écran. Le fait d'imposer aux participants un confort visuel comme critère, réduit nettement l'amplitude de mouvement sur le lacet, sans réduire celle du roulis. Contrairement aux résultats des études biomécaniques [Thibodeau et Patton, 1996], on obtient donc une amplitude confortable plus grande sur le roulis que sur le lacet.

Dans le chapitre suivant, nous présentons un état de l'art sur les fonctions de transfert existantes et les stratégies utilisées en fonction du contexte afin d'introduire les bases nécessaires pour la création de notre propre fonction de transfert. Cette fonction utilisera l'amplitude confortable et la *résolution utile* de la tête que nous venons de déterminer.

---

## CONTRÔLE DU POINT DE VUE ORBITAL : FONCTIONS DE TRANSFERT ET TECHNIQUES D'INTERACTION

Dans ce chapitre, nous définissons dans un premier temps certaines notions fondamentales telles que les fonctions de transfert, le  $gain_{CD}$  ou le débrayage. Nous donnons ensuite un aperçu des fonctions de transfert existant dans la littérature, que nous classifions en deux catégories : fonctions à  $gain_{CD}$  constant et  $gain_{CD}$  variable. Nous présentons ensuite deux techniques de contrôle du point de vue orbital. Pour cela, nous nous appuyons sur les résultats du chapitre 4 qui nous avaient permis de déterminer deux paramètres : l'*amplitude maximale confortable* et la *résolution utile* de la tête. Nous comparons ensuite ces techniques sur une tâche de pointage de type Fitts en 3D. Une dernière étude vise à montrer la validité externe de notre technique. Finalement, nous présentons trois applications de notre technique d'interaction.

### 5.1 DÉFINITIONS

#### 5.1.1 Fonction de transfert

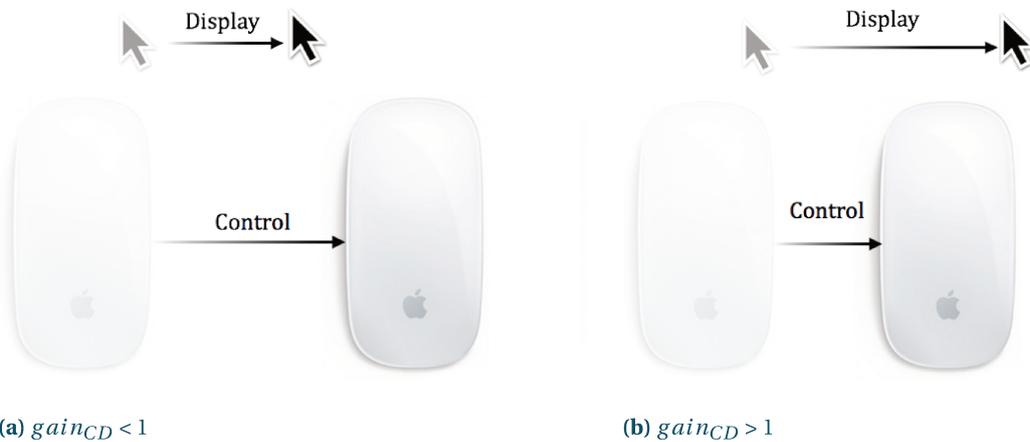
Au sens général, une fonction de transfert établit la correspondance entre la valeur d'entrée d'un système et sa valeur en sortie. Dans cette thèse nous nous intéressons à un sous ensemble de fonctions de transfert définissant la correspondance entre une valeur d'entrée  $valeur_{entrée}$  produite au sein d'un espace *Moteur* (control) par un utilisateur à travers une modalité d'interaction donnée et la valeur en sortie  $valeur_{sortie}$  correspondante dans un espace *Écran* (display). Dans notre cas particulier, cette correspondance est régie par le  $gain_{CD}$  qui peut être éventuellement fonction d'un ensemble de paramètres  $X$  (vitesse, distance par rapport à un objet, etc.).

$$valeur_{sortie} = f(valeur_{entrée}, X) = gain_{CD}(X) \times valeur_{entrée}$$

### 5.1.2 Gain Moteur Écran (Control Display gain)

De manière générale, le gain représente le ratio entre la valeur en sortie d'un système et sa valeur en entrée. Dans notre cas, le gain Moteur Écran ( $gain_{CD}$ ) [Gibbs, 1962] est couramment défini comme étant le ratio sans unité entre la valeur obtenue sur le dispositif de sortie représentant l'espace Écran et la valeur fournie par l'utilisateur au niveau du dispositif d'entrée dans l'espace Moteur. Il est aussi représenté dans certains travaux utilisant un  $gain_{CD}$  variable comme une fonction de plusieurs paramètres. Un  $gain_{CD}$  de 1 signifie qu'il y a une correspondance parfaite entre les valeurs du dispositif d'entrée et celles du dispositif de sortie. On parle alors de  $gain_{CD}$  isomorphe ([Poupyrev et al., 2000]) ou à l'inverse de  $gain_{CD}$  non-isomorphe pour un  $gain_{CD} \neq 1$ .

$$gain_{CD} = \frac{Display}{Control} = \frac{valeur_{sortie}}{valeur_{entrée}}$$



**FIGURE 5.1** – À gauche, le  $gain_{CD}$  est inférieur à 1. Tout mouvement dans l'espace moteur produit un mouvement d'amplitude plus faible en sortie (ici le mouvement du curseur). L'utilisateur perd en amplitude, mais gagne en précision. À droite, le  $gain_{CD}$  est supérieur à 1. Tout mouvement dans l'espace moteur est donc, à l'inverse, plus grand en sortie. L'utilisateur gagne en amplitude, mais perd en précision.

#### *Gain<sub>CD</sub> < 1*

Une valeur de  $gain_{CD} < 1$  signifie que la valeur du dispositif de sortie est plus faible que celle au niveau du dispositif d'entrée (voir figure 5.1a). Cela a deux conséquences sur l'interaction : le mouvement généré est moindre, en revanche la précision avec laquelle celui-ci peut être contrôlé augmente.

#### *Gain<sub>CD</sub> > 1*

En revanche, une valeur de  $gain_{CD} > 1$  signifie que la valeur du dispositif de sortie est plus importante que celle au niveau du dispositif d'entrée (voir figure 5.1b). Cela a pour conséquence une augmentation de l'amplitude de l'utilisateur, mais aussi une baisse de sa précision.

Certaines techniques d'interaction peuvent faire varier la valeur du gain en fonction de divers paramètres (temps, orientation, distance, vitesse, etc.). Nous appelons gain *variable* tout  $gain_{CD}$  dont la valeur

n'est pas fixe au cours du temps et peut varier en fonction de diverses conditions (figure 5.2). À l'opposé, nous appelons gain *constant*, un  $gain_{CD}$  dont la valeur est immuable. On distingue donc deux types de  $gain_{CD}$  en fonction de l'évolution de celui-ci.

$$gain_{CD}(vitesse) = \begin{cases} 1 & \text{si vitesse} < 2 \\ 1 + \frac{vitesse}{2} & \text{autrement} \end{cases}$$

FIGURE 5.2 – Exemple de  $gain_{CD}$  variable en fonction de la vitesse

### 5.1.3 Débrayage

Dans le cas où l'espace moteur est insuffisant pour réaliser un mouvement donné dans l'espace écran, plusieurs stratégies sont possibles : augmenter la valeur du  $gain_{CD}$  ou réaliser le mouvement en plusieurs étapes avec le débrayage.

Le débrayage souvent appelé *clutching* ([Hand, 1997, Stoakley et al., 1995]) ou *ratcheting* ([Hinckley et al., 1997, Ware et Jessome, 1988]) permet à l'utilisateur de repositionner le dispositif d'entrée sans changer la valeur en sortie afin de gagner en amplitude de mouvement. Un exemple fréquent est la souris qui, si l'espace moteur est trop faible pour placer le curseur dans une position donnée, peut être soulevée et placée à une nouvelle position afin de gagner en espace moteur.

### 5.1.4 Contrôle en position et en vitesse

Deux types de contrôles sont fréquemment utilisés ([Poupyrev et al., 2000, Zhai, 1995]) dans les fonctions de transfert : le *contrôle en position* et le *contrôle en vitesse*.

**Contrôle en position.** Avec le *contrôle en position* (position control), aussi appelé correspondance d'ordre 0 ([Poupyrev et al., 2000]), le dispositif d'entrée contrôle la position de l'objet à manipuler (i.e. curseur). Ce type de contrôle est souvent utilisé avec les dispositifs de pointage tels que la souris pour contrôler la position du curseur.

**Contrôle en vitesse.** Dans le cas du *contrôle en vitesse* (rate control) ou une correspondance d'ordre 1 ([Poupyrev et al., 2000]), le dispositif d'entrée contrôle le vecteur vitesse (vélocité) de l'objet à manipuler. Ce type de contrôle est particulièrement bien adapté pour les joysticks. On le trouve également avec les dispositifs de type trackpoint.

Des correspondances d'ordre supérieur à 1 (par exemple basée sur l'accélération) ont été explorées, mais celles-ci sont généralement moins efficaces que le contrôle en position ou en vitesse ([Massimino et al., 1989, Zhai, 1995]).

### 5.1.5 Type de dispositifs

En fonction du type de contrôle utilisé (position ou vitesse), il est nécessaire d'utiliser le type de dispositif le mieux adapté pour interagir. Dans ce but, Zhai [1995] distingue et décrit trois types de dispositifs : *isotoniques*, *isométriques* et *élastiques*.

**Dispositifs isotoniques.** Les dispositifs *isotoniques* (force identique) sont des dispositifs pouvant être déplacés librement, sans résistance et renvoyant une position en sortie. La souris ou les stylets pour tables tactiles sont des exemples de dispositifs isotoniques.

**Dispositifs isométriques.** Les dispositifs *isométriques* (position identique) sont des dispositifs fixes sur lesquels l'utilisateur doit appliquer une force pour interagir. Les dispositifs isotoniques renvoient donc une force en sortie. C'est par exemple, le cas du trackpoint sur les ordinateurs portables.

**Dispositifs élastiques.** Les dispositifs *élastiques* sont des dispositifs opposants une résistance au déplacement en fonction de l'amplitude de mouvement et possédant également un retour en position nulle. Ces dispositifs peuvent renvoyer soit une force, soit une position en sortie. Le joystick est un exemple de dispositif élastique.

Zhai [1995] a montré que les dispositifs *isotoniques* sont plus adaptés au contrôle en position, tandis que les dispositifs *élastiques* ou *isométriques* sont plus adaptés pour le contrôle en vitesse. En effet, les dispositifs *élastiques* ou *isométriques* reviennent en position d'origine dès lors que l'utilisateur cesse d'appliquer une force contraire, ce qui est particulièrement adapté pour le contrôle en vitesse, en permettant d'interrompre le déplacement/rotation. Avec un dispositif *isotonique*, le retour en position d'origine n'est pas une tâche aisée, ainsi ce type de dispositif n'est pas adapté pour effectuer du contrôle en vitesse. À l'inverse, les dispositifs *isotoniques* pouvant être déplacés en y appliquant une force constante, ceux-ci sont particulièrement adaptés pour réaliser du contrôle en position, ce qui n'est pas le cas des dispositifs *élastiques* ou *isométriques*.

Dans nos travaux, nous utilisons la tête. La tête n'est évidemment pas un *dispositif d'entrée* classique. D'après les définitions précédentes, celle-ci n'est pas assimilable à un dispositif *isométrique*, mais plutôt probablement à mi-chemin entre un dispositif *isotonique* et un dispositif *élastique*. Plusieurs types de contrôles ont d'ailleurs été envisagés avec la tête comme modalité principale. Guy et al. [2015] ont par exemple utilisé du contrôle en vitesse, tandis que Poupyrev et al. [1999] ont utilisé du contrôle en position.

#### 5.1.6 Correspondance (mapping)

En plus du type de contrôle utilisé et du dispositif, il existe différentes manières d'établir la correspondance entre l'entrée et la sortie du système.

**Correspondance absolue.** Une correspondance *absolue* entre une valeur d'entrée et une valeur de sortie signifie qu'à tout moment, pour une valeur d'entrée donnée correspond une valeur unique en sortie, quel que soit l'état du système et les manipulations précédemment effectuées sur celui-ci (voir figure 5.3). Avec un  $gain_{CD} = 1$  et en utilisant un contrôle en position, il est possible de connaître à tout moment la valeur du dispositif de sortie, on a donc une correspondance absolue. Ce type de correspondance est utilisée avec les dispositifs tactiles sur lesquels la position du doigt permet d'interagir directement avec le contenu sous-jacent.

$$\forall t_1 \neq t_2, v_{entree}^{t_1} = v_{entree}^{t_2} \implies f(v_{entree}^{t_1}) = f(v_{entree}^{t_2})$$

FIGURE 5.3 – Correspondance absolue

**Correspondance relative.** Avec une correspondance *relative* en revanche, la valeur en sortie est dépendante des manipulations précédemment effectuées sur le système (voir figure 5.4). Ce type de correspondance est par exemple utilisée entre la souris et le curseur [Casiez et Roussel, 2011, Casiez et Vogel, 2008]. Tout déplacement du curseur s'effectue par rapport à sa dernière position en y ajoutant un vecteur déplacement calculé par rapport au déplacement de la souris par l'utilisateur.

$$\exists t_1 \neq t_2, v_{entree}^{t_1} = v_{entree}^{t_2} \text{ tel que } f(v_{entree}^{t_1}) \neq f(v_{entree}^{t_2})$$

FIGURE 5.4 – Correspondance relative

Type de dispositif	Correspondance	Contrôle	Application
Isotonique (mains)	Absolue	Position	Pointage sur tablette
Isotonique (stylet)	Absolue	Position	Dessin sur tablette graphique
Isotonique (souris)	Relative	Position	Pointage sur ordinateur
Isométrique (trackpoint)	Relative	Vitesse	Pointage sur ordinateur
Elastique (joystick)	Relative	Vitesse	Simulateur de vol

TABLE 5.1 – Correspondance et type de contrôle utilisés en fonction du type de dispositif et de l'application

Le type de correspondance à utiliser dépend de la taille de l'espace moteur par rapport à l'espace écran et du type de dispositif utilisé. En règle générale, on utilise la correspondance absolue avec un dispositif de contrôle en position quand l'espace moteur est suffisamment large pour atteindre n'importe quelle partie de l'espace écran sans avoir recours à un mécanisme de débrayage. En revanche, on utilise la correspondance relative avec un mécanisme de débrayage sur les dispositifs de contrôle en position quand l'espace moteur est inférieur à l'espace écran. On utilise également la correspondance relative avec les dispositifs de contrôle en vitesse. Voir le tableau 5.1 pour un aperçu.

### 5.1.7 Conformité au point zéro (nulling compliance)

Une technique d'interaction est *conforme au point zéro* si à la valeur d'entrée qui représente la position d'origine du dispositif correspond la valeur d'origine en sortie (voir figure 5.5). Prenons l'exemple d'un dispositif de type joystick permettant de contrôler la rotation d'un objet dans une scène 3D. Si à tout moment dans le temps et indépendamment des manipulations effectuées précédemment avec le joystick, à la position d'origine du joystick correspond une orientation nulle de l'objet, alors il y a conformité au point zéro. Toute technique utilisant une correspondance absolue est par conséquent conforme au point zéro, tandis qu'une technique utilisant une correspondance relative ne l'est pas ([Dominjon et al., 2006, Poupyrev et al., 2000]).

$$valeur_{entree} = 0 \implies valeur_{sortie} = 0$$

FIGURE 5.5 – Conformité en zéro

Avec le débrayage ou l'utilisation d'un  $gain_{CD}$  dynamique, un décalage s'accumule progressivement en sortie. Dans le cas du contrôle de la rotation avec la tête, il est important qu'il y ait une correspondance en zéro entre la tête et le dispositif de sortie afin de préserver le modèle mental de l'utilisateur, c'est pourquoi nous définissons le critère de la conformité au point zéro.

## 5.2 FONCTIONS DE TRANSFERT & ÉTUDES

Il existe un grand nombre de travaux sur les fonctions de transfert pour une large variété d'applications (pointage [Beaudouin-Lafon et al., 2014, König et al., 2009], manipulation d'objets distants dans une scène immersive [Poupyrev et al., 1996], etc.). Dans cette section nous nous intéressons uniquement aux fonctions de transfert ou aux études existantes sur le contrôle du point de vue orbital (ou la rotation d'objets qui peut y être assimilée). Cependant, nous ne nous limitons pas uniquement aux travaux utilisant la tête comme modalité d'entrée principale, car nous considérons que ceux-ci sont pertinents, indépendamment de la modalité utilisée. Nous avons classifié les travaux existants sur deux axes : *type de gain<sub>CD</sub>* (constant ou variable) et *valeur de gain<sub>CD</sub>* ( $Gain_{CD} > 1$  ou  $Gain_{CD} < 1$ ). Nous commençons par présenter les travaux utilisant un *gain<sub>CD</sub>* constant.

### 5.2.1 *Gain<sub>CD</sub>* constant

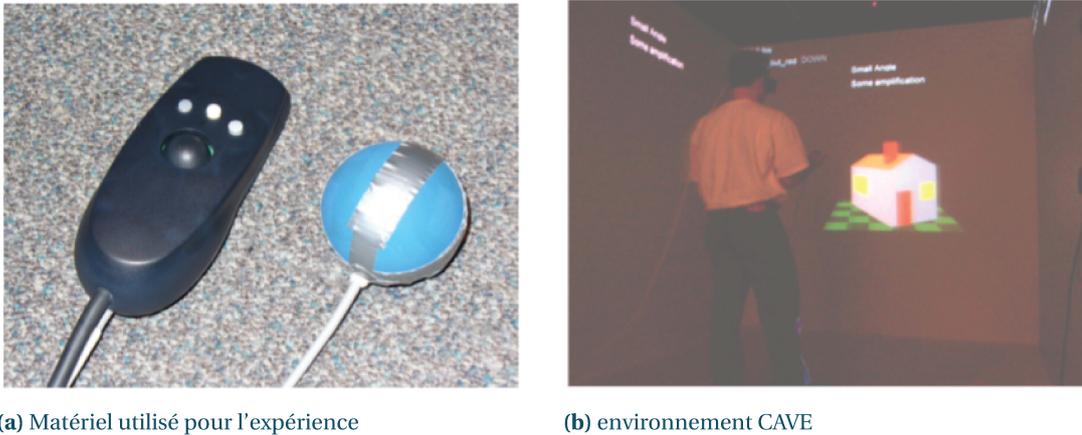
On trouve dans la littérature plusieurs études ainsi que des travaux utilisant des valeurs de *gain<sub>CD</sub>* constantes.

#### *Gain<sub>CD</sub> > 1*

Beaucoup d'études ont notamment étudié l'impact d'une valeur de *gain<sub>CD</sub> > 1*. Par exemple, Poupyrev et al. [2000] présentent une méthodologie montrant l'avantage des techniques de rotation 3D basées sur l'utilisation d'un *gain<sub>CD</sub>* non-isomorphe constant. Les auteurs mènent une étude afin de comparer les performances obtenues en utilisant un *gain<sub>CD</sub>* isomorphe ( $gain_{CD} = 1$ ) et un *gain<sub>CD</sub>* de 1.8. Dans cette étude, 20 sujets sont amenés à réaliser une tâche nécessitant de placer un objet 3D - à l'aide d'un capteur magnétique à 6 degrés de liberté -, originalement placé dans une orientation arbitraire, dans une orientation cible. Dans leur expérience, deux facteurs ont été étudiés : d'une part la technique d'interaction (isomorphe ou non-isomorphe avec amplification) et d'autre part l'amplitude de rotation nécessaire (petite pour une valeur allant de 20° à 60° ou grande pour une valeur allant de 70° à 180°). Les résultats de leur étude montrent une amélioration de 13% sur le temps de réalisation de la tâche avec la technique d'interaction non isomorphe, quelle que soit l'amplitude (petite ou grande) de la rotation. D'autre part, l'amplification n'a aucun effet sur le taux d'erreurs d'orientation. En conclusion, 18 des 20 participants préfèrent l'utilisation de la technique non-isomorphe, car celle-ci leur permet d'interagir plus rapidement tout en nécessitant moins d'effort physique.

Laviola et Katzourin [2007] présentent une étude qui se base sur la précédente étude de Poupyrev et al. [2000] dans le but de vérifier si les conclusions de leur travail sont applicables dans un environnement de type CAVE. Une expérience est menée pour comparer l'effet d'un *gain<sub>CD</sub>* isomorphe sur le temps de réalisation et la précision d'une tâche d'orientation d'objets 3D. Quatre valeurs de *gain<sub>CD</sub>* différentes ont été étudiées : 1 :1, 1 :2, 1 :3 et 1 :4. Afin de réaliser la tâche, le participant partait d'un modèle incliné avec une orientation soit petite (entre 20 et 60°), soit large (entre 70 et 180°) et devait incliner celui-ci avec une orientation cible à une précision soit de 6°, soit de 18°. Les résultats montrent que la tâche peut être réalisée 15% plus rapidement en utilisant un *gain<sub>CD</sub>* de 3 sans perte de précision.

Dans une étude similaire à la précédente, Teather et Stuerzlinger [2008] ont étudié l'effet du *gain<sub>CD</sub>* avec la perspective couplée à la tête dans le contexte des jeux vidéo. Les auteurs comparent quatre



**FIGURE 5.6** – À gauche, un *Wanda* est utilisé au cours de l'expérience dans la main non dominante de l'utilisateur pour valider l'orientation du modèle. À droite, un capteur magnétique *Polhemus FASTRAK* à 6 degrés de liberté permet d'orienter le modèle dans l'espace.

valeurs de  $gain_{CD}$  différentes (1 :1, 1 :2, 1 :3, 1 :5) sur une tâche de placement d'objets dans une scène 3D. Cependant, dans leur cas, aucun effet significatif n'a pu être observé sur le temps de réalisation de la tâche ni sur le taux de succès, quelle que soit la valeur du  $gain_{CD}$  utilisée.

Des travaux ont été conduits par Jay et Hubbard [2003] afin d'élargir le champ de vision des utilisateurs avec les casques de réalité virtuelle (HMD), grâce à un  $gain_{CD} > 1$ . En effet, selon les auteurs, le champ de vision de l'être humain est de  $200^\circ$  horizontalement et  $150^\circ$  verticalement. Or, les HMD ne permettent pas d'obtenir un champ de vision aussi étendu. Jay et al. proposent donc d'amplifier la valeur de rotation de la tête dans le monde physique afin d'augmenter le champ de vision de l'utilisateur lors du port du casque artificiellement. Une valeur d'amplification de rotation de la tête de 2 a été choisie. Une étude a été conduite afin de vérifier le gain de performance en temps de localisation d'objets dans une scène virtuelle. Trois conditions d'amplification sont étudiées : sans amplification, amplification de la tête, amplification de la tête et des mains (celles-ci tournent avec la tête).

Les résultats de l'étude montrent une amélioration sur le temps de réalisation de la tâche de 21% lorsque les mouvements de la tête et des mains sont amplifiés. Une amplification de la tête seulement ne présente pas de résultats significatifs sur le temps de réalisation de la tâche. Une explication avancée par les auteurs est l'inconsistance de la perception. Dans la réalité les mains et la tête tournent avec l'orientation du buste. Or, si les mouvements de la tête sont amplifiés uniquement dans la simulation, les mains ou la tête ne semblent plus être attachées au buste ce qui devient incohérent, gêne et entraîne une perte de performances.

#### $Gain_{CD} < 1$

Contrairement au cas où le  $Gain_{CD}$  est  $> 1$ , on ne trouve pas dans la littérature de travaux traitant de l'utilisation d'un  $Gain_{CD} < 1$  constant pour le contrôle du point de vue orbital. La plupart des travaux existants cherchent au contraire à permettre à l'utilisateur de gagner en rapidité en augmentant l'amplitude de mouvement sans perdre en précision.

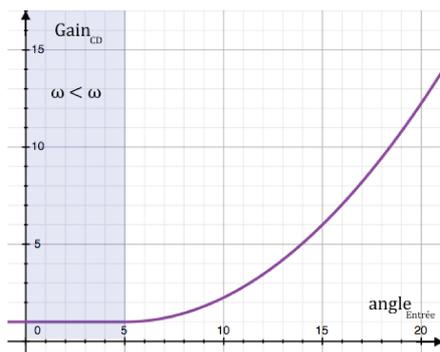
En revanche, Lin et al. [1992] ont étudié l'utilisation de la tête comme modalité pour réaliser des tâches de pointage avec des  $Gain_{CD} < 1$  (0.15, 0.3 et 0.6). Cependant, la nature de la tâche à réaliser (déplacer un curseur avec la tête) et la manière dont la valeur de gain est calculée (ratio entre l'angle sous-tendu par le déplacement du curseur et l'angle de la tête) sont différentes de ce que nous étudions ici.

### 5.2.2 $Gain_{CD}$ variable

Un certain nombre de travaux se sont intéressés à des  $Gain_{CD}$  dont la valeur peut varier en fonction de différents critères (distances, vitesse de l'utilisateur), voire en fonction des intentions de l'utilisateur.

#### $Gain_{CD} > 1$

Poupyrev et al. [1999] présentent une technique d'interaction (figure 5.7b) pour la rotation d'objets 3D ou le contrôle du point de vue utilisant un  $gain_{CD}$  non linéaire. Tant que la distance angulaire de l'utilisateur  $\omega$  est sous un seuil prédéfini  $\omega_0$ , la correspondance entre ses mouvements et le dispositif de sortie est 1 :1 (voir figure 5.7a). Lorsque celui-ci dépasse le seuil  $\omega_0$ , la correspondance devient non linéaire. Plus la distance angulaire de l'utilisateur s'éloigne de l'angle seuil, plus la valeur du  $gain_{CD}$  est importante.



(a) Évolution du gain en fonction de la distance angulaire

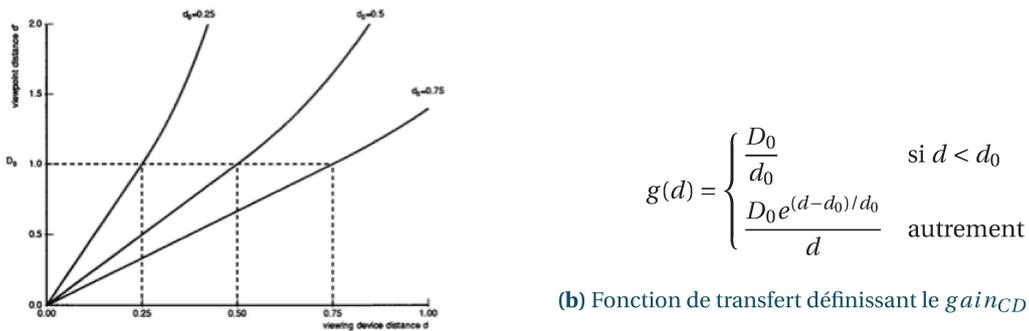
$$g(\omega) = \begin{cases} 1 & \text{si } \omega < \omega_0 \\ 1 + c(\omega - \omega_0)^2 & \text{autrement} \end{cases}$$

(b) Fonction de transfert définissant la valeur du gain

**FIGURE 5.7** – À gauche, courbe représentant l'évolution du gain en fonction de la distance angulaire pour  $\omega_0 = 5$  et  $c = \frac{1}{20}$ . À droite, fonction de transfert couplant l'utilisation d'un  $gain_{CD}$  fixe à un  $gain_{CD}$  variable. La valeur de  $gain_{CD}$   $g$  est égale à 1 si la distance angulaire de l'utilisateur  $\omega$  est inférieure à une distance angulaire seuil  $\omega_0$ . Autrement, la valeur du  $gain_{CD}$  augmente de manière avec la distance et un facteur  $c$ .

Song et Norman [1993] proposent une approche similaire à celle de Poupyrev et al. pour le contrôle du point de vue dans une scène 3D. La fonction de transfert définie dans ce travail permet de contrôler à la fois la position du point de vue dans la scène de manière précise et de réaliser de larges amplitudes. La fonction (figure 5.8) prend en paramètre la distance du dispositif d'entrée par rapport à l'origine du repère et renvoie une valeur de  $gain_{CD}$  qui est soit constante pour des distances faibles par rapport à l'origine, soit variable et amplifiée pour des distances plus grandes. Cette fonction découpe l'espace écran en deux parties, séparées par une sphère de diamètre  $D_0$ . L'espace écran inclus dans cette sphère est appelé *DOI* (Domain Of Interest). À ce sous-espace écran correspond un sous-espace physique défini par une sphère de diamètre  $d_0$  et intitulé *LVS* (Linear Viewing Space). Tout mouvement dans le *LVS* entraîne un mouvement d'amplitude identique dans le *DOI*. En revanche, si la distance dans le monde physique devient supérieure à  $d_0$ , à savoir le déplacement se fait hors du *LVS*, le mouvement est

amplifié par un  $gain_{CD}$  non linéaire augmentant de manière exponentielle par rapport à la distance.



(a) Correspondance entre la distance du périphérique d'entrée par rapport à l'origine et la distance en sortie de fonction

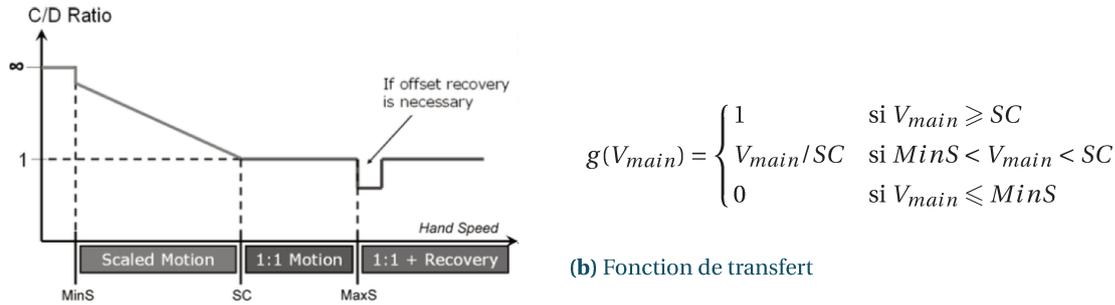
**FIGURE 5.8** – Fonction de transfert définissant la correspondance entre la distance du périphérique suivi par rapport à l'origine dans le monde réel et sa distance dans le monde virtuel. La variable  $d$  représente la distance dans le monde réel,  $d_0$  le rayon de la sphère définissant le LVS et  $D_0$  le rayon de la sphère définissant le DOI

$Gain_{CD} < 1$

À l'inverse, d'autres travaux proposent non pas d'amplifier le  $gain_{CD}$ , mais de diminuer celui-ci afin de gagner en précision.

Ainsi, avec *PRISM*, Frees et al. [2007] proposent une technique d'interaction (figure 5.9) à  $gain_{CD}$  variable se basant sur la vitesse pour interpréter les intentions de l'utilisateur. Si l'utilisateur se déplace lentement, c'est que celui-ci a besoin de précision. Au contraire, si celui-ci se déplace rapidement, c'est qu'il cherche à couvrir une large amplitude. Cette technique utilise trois seuils de vitesse afin de déterminer la valeur de  $gain_{CD}$  :  $MinS$ ,  $SC$  et  $MaxS$ . La constante  $MinS$ , pour *Minimum speed*, représente la vitesse minimum à laquelle l'utilisateur doit effectuer un mouvement, pour que celui-ci soit pris en compte par le système. Cela permet d'identifier les mouvements volontaires de l'utilisateur parmi le bruit. Un mouvement dont la vitesse est inférieure à ce niveau est considéré comme involontaire. La constante  $SC$ , pour *Scaling constant*, représente la vitesse seuil à atteindre pour avoir un  $gain_{CD}$  de 1. Tout mouvement dont la vitesse est comprise entre  $MinS$  et  $SC$  a donc une valeur de  $gain_{CD}$  comprise entre 0 et 1. Le facteur de  $gain_{CD}$  n'étant pas constant au cours du temps, la correspondance entre les mouvements de l'utilisateur et la valeur du système est relative. Or, comme souligné par Pouppeyev et al. [2000], une correspondance relative ne permet pas de maintenir la conformité en zéro. L'utilisateur amasse donc au cours du temps un décalage entre ses valeurs et les valeurs du système. *PRISM* permet, à l'aide de deux méthodes différentes, de rattraper ce décalage. La première permet à l'utilisateur de revenir sur son déplacement, c'est-à-dire se déplacer en sens inverse pour rattraper le décalage. La seconde augmente brièvement la valeur du  $gain_{CD}$  au-delà de 1 si la vitesse de l'utilisateur est supérieure à la constante  $MaxS$  (*Maximum Speed*). L'inconvénient est que le rattrapage du décalage entraîne un saut visuel, ce qui est déstabilisant pour l'utilisateur.

Un autre exemple de ce genre d'approche est *Adaptive Pointing*. König et al. [2009] présentent une fonction de transfert (figure 5.10) pour le pointage distant sur les périphériques d'entrées absolus



**FIGURE 5.9** – Fonction de transfert définissant la valeur du  $gain_{CD}$  pour la technique *PRISM*. Si la vitesse des mouvements de l'utilisateur est inférieure à  $MinS$ , ceux-ci ne sont pas pris en compte. Si leur vitesse est comprise entre  $MinS$  et  $SC$ , le  $gain_{CD}$  est le ratio entre la vitesse et  $SC$ . Si la vitesse est comprise, entre  $SC$  et  $MaxS$  le  $gain_{CD}$  est de 1. Enfin, si la vitesse est supérieure à  $MaxS$ , le  $gain_{CD}$  augmente brièvement afin de rattraper le décalage accumulé.

permettant d'ajuster la valeur du  $gain_{CD}$  pour obtenir davantage de précision. Bien que ce travail porte sur du pointage, l'approche rappelle celle utilisée dans *PRISM* c'est pourquoi nous présentons celle-ci. Cette approche est similaire dans le sens où plusieurs seuils de vitesse sont utilisés afin de définir l'évolution du  $gain_{CD}$ . Ainsi, si la vitesse du dispositif d'entrée tombe en dessous de  $v_{min} = 0.0028$  m/s, le  $gain_{CD}$  prend une valeur constante  $g_{min} = 0.32$ . Si la vitesse est comprise entre  $v_{min}$  et  $v_{max}$ , le  $gain_{CD}$  est compris entre  $g_{min}$  et  $g_{max} = 1.055$ . L'approche varie cependant, car la valeur du décalage accumulé pendant les déplacements où le  $gain_{CD}$  est différent de 1 est utilisée dans le calcul du  $gain_{CD}$ . Dans un premier temps, la vitesse (figure 5.10a) et le décalage (figure 5.10b) sont normalisés. La valeur de  $gain_{CD}$  est ensuite calculée à l'aide de la fonction 5.11b. Une étude a été menée sur une tâche de pointage afin de comparer les performances de la technique *Adaptive Pointing* par rapport à une autre technique de pointage absolu basée sur un filtre de Kalman. Les résultats montrent une diminution du taux d'erreur de 63% en faveur de la technique *Adaptive Pointing* ainsi qu'une diminution du temps de réalisation de la tâche de 19%.

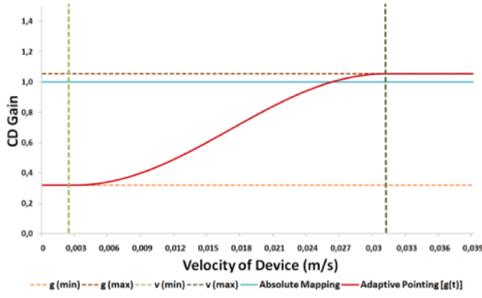
$$v_x(t) = \begin{cases} 1 & \text{si } v_x(t) > v_{max} \\ 0 & \text{si } v_x(t) < v_{min} \\ \frac{v_x(t) - v_{min}}{v_{max} - v_{min}} & \text{autrement} \end{cases} \quad d_x(t) = \begin{cases} 1 & \text{si } |d_x(t)| > d_{max} \\ 0 & \text{si } |d_x(t)| < d_{min} \\ \frac{|d_x(t)| - d_{min}}{d_{max} - d_{min}} & \text{autrement} \end{cases}$$

(a) Normalisation de la vitesse

(b) Normalisation du décalage

**FIGURE 5.10** – Normalisation de la vitesse  $v_x(t)$  et du décalage  $d_x(t)$  avec la technique *Adaptive Pointing*. La valeur du décalage  $d_x(t)$  est la différence entre le déplacement effectif de l'utilisateur à un instant  $t$ ,  $x_{mot}(t)$  et la valeur de déplacement du système au temps  $t - 1$ ,  $x_{disp}(t - 1)$ .

Reilink et al. [2010] proposent trois techniques d'interaction pour le contrôle d'un endoscope à partir des mouvements d'inclinaison verticale et horizontale de la tête. La première technique (figure 5.12) est basée sur un contrôle en position. Si la valeur d'inclinaison de la tête est inférieure à une variable seuil  $b$ , le  $gain_{CD}$  augmente linéairement en fonction de l'inclinaison de la tête. En revanche, passé ce seuil, le  $gain_{CD}$  est fixé à 1. La seconde technique (figure 5.13) est basée sur un contrôle en vitesse. Ainsi, la valeur du  $gain_{CD}$  est paramétrée par la vitesse de rotation de la tête. Des mouvements rapides de la tête permettent d'atteindre une amplitude plus large avec moins de précision, tandis que des mouvements



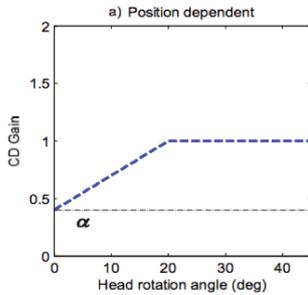
$$g_x(t) = g_{min} + \frac{1}{2} \left[ \sin\left(m_x(t) \cdot \pi - \frac{\pi}{2}\right) + 1 \right] (g_{max} - g_{min})$$

(b) Définition du  $gain_{CD}$ ,  $m_x(t)$  étant la valeur maximale entre le décalage et la vitesse normalisés.

(a) Évolution du  $gain_{CD}$  en fonction de la vitesse

**FIGURE 5.11** – Fonction de transfert définissant la valeur du  $gain_{CD}$  pour la technique Adaptive pointing [König et al., 2009]. La fonction périodique permet une transition plus souple lors du passage de correspondance relative à correspondance absolue.

lents donnent plus de précision sur une petite amplitude. Enfin, la dernière technique (figure 5.14) utilise un  $gain_{CD}$  constant tant que l'inclinaison de la tête reste sous un seuil  $c$ . Passé ce seuil, la valeur du  $gain_{CD}$  est contrôlée non plus en position, mais en vitesse et se fait par rapport à la vitesse d'inclinaison de la tête. Ces trois techniques ont pu être comparées sur une tâche d'orientation, l'inclinaison de la tête permettant de manipuler la position du curseur verticalement et horizontalement. Le temps moyen de réalisation de la tâche a été mesuré pour chaque technique. Globalement, la technique hybride obtient le temps moyen le plus long tandis que les techniques basées sur le contrôle en position et le contrôle en vitesse obtiennent des temps moyens équivalents.



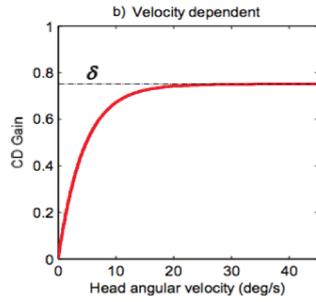
$$g(p) = \begin{cases} \alpha + \beta \|p\|_2 & \text{si } 0 < \|p\|_2 \leq b \\ \alpha + b\beta & \text{si } \|p\|_2 > b \end{cases}$$

(b) Première fonction de transfert pour le contrôle en position

(a) Contrôle en position

**FIGURE 5.12** – Fonction de transfert définissant la valeur du  $gain_{CD}$ .  $\alpha$  représente la valeur minimale du  $gain_{CD}$ ,  $\beta$  le facteur d'amplification du  $gain_{CD}$ ,  $b$  la valeur seuil au-delà de laquelle le  $gain_{CD}$  est constant et  $p$  l'inclinaison de la tête

Nous avons vu plusieurs exemples de fonctions de transfert dans cette section utilisant différentes approches pour déterminer une valeur de  $gain_{CD}$ . Certaines fonctions utilisent un  $gain_{CD}$  constant tandis que d'autres utilisent un  $gain_{CD}$  variable, généralement basé sur la vitesse de l'utilisateur. Dans la section suivante, nous présentons la création de deux techniques d'interaction pour le contrôle du point de vue basées sur les fonctions de transfert vues dans ce chapitre et les résultats obtenus dans le chapitre précédent (chapitre 4, *Contrôle du point de vue : précision utilisateur et matériel*).

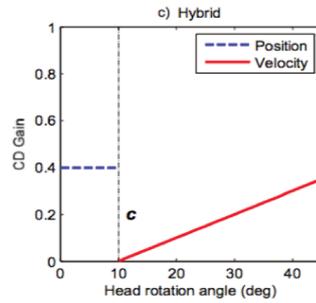


$$g(\dot{p}) = \delta \left( 1 - e^{-\epsilon \|\dot{p}\|_2} \right)$$

(b) Seconde fonction de transfert pour le contrôle en vitesse

(a) Contrôle en vitesse

FIGURE 5.13 – Fonction de transfert définissant la valeur du  $gain_{CD}$ .  $\dot{p}$  représente la dérivée de l'inclinaison de la tête par rapport au temps,  $\delta$  la valeur de  $gain_{CD}$  maximale et  $\epsilon$  un facteur d'amplification du  $gain_{CD}$



$$g(p) = \begin{cases} g_p = \alpha_2 & \text{si } 0 < \|p\|_2 \leq c \\ g_v = \gamma(\|p\|_2 - c) & \text{si } \|p\|_2 \geq c \end{cases}$$

(b) Troisième fonction de transfert pour un contrôle hybride soit en position, soit en vitesse

(a) Contrôle hybride

FIGURE 5.14 – Fonction de transfert définissant la valeur du  $gain_{CD}$ . Celui-ci est constant et le contrôle se fait en position tant que l'inclinaison de la tête reste sous un seuil  $c$ . Passé ce seuil, le contrôle se fait en vitesse. Les variables  $\alpha_2$  et  $\gamma$  représentent respectivement la valeur de  $gain_{CD}$  minimale et un facteur d'amplification

### 5.3 CRÉATION DES TECHNIQUES D'INTERACTION

Dans le chapitre précédent, nous avons déterminé deux paramètres nécessaires à la création de techniques d'interaction pour le contrôle du point de vue orbital : *l'amplitude maximale confortable* de la tête (en moyenne  $25^\circ$  sur le lacet et  $35^\circ$  sur le roulis) et le plus petit mouvement contrôlable de la tête ( $1^\circ$ ) que nous avons appelé *résolution utile de la tête*. Dans cette section nous détaillons la création de deux techniques d'interaction pour le contrôle du point de vue et leurs fonctions de transfert.

#### 5.3.1 Considérations et réalisation

Comme nous l'avons vu au chapitre 3 (Contrôle du point de vue), le mouvement de rotation orbital est l'un des mouvements les plus utilisés dans les logiciels de modélisation 3D car il permet d'obtenir un aperçu de la scène sous différents angles et ainsi de compenser la perte de perception de profondeur observée avec un écran 2D. De notre côté, nous nous intéressons au tracé de trajectoire 3D, une tâche qui peut être assimilée à de la modélisation 3D, c'est pourquoi ce type de contrôle est particulièrement adapté. À la manière des logiciels de type *Floor Planner*, dans lesquels l'utilisateur peut tourner autour

de la scène afin de placer divers objets 3D (meubles, appareils, etc.), nous voulions être en mesure de réaliser un mouvement de rotation exocentrique complet dans la scène 3D (360°) afin d'avoir un aperçu global et une meilleure perception de celle-ci. Il fallait donc pouvoir tourner de 180° dans chaque direction. Nous voulions également être en mesure de jeter un coup d'oeil rapide dans la scène en facilitant le retour en position d'origine sans avoir à utiliser un débrayage qui aurait pour effet de ralentir l'interaction ([Casiez et al., 2007]).

Comme nous avons pu le voir dans la section précédente (5.2), il existe deux types de  $gain_{CD}$  : constant (5.2.1) et variable (5.2.2). Avec un  $gain_{CD}$  variable, un décalage s'accumule au cours du temps entre l'orientation de la tête et l'orientation de la scène, ce qui est gênant pour l'utilisateur en l'absence de mécanisme de débrayage permettant de rattraper ce décalage. Nous avons envisagé dans un premier temps d'utiliser ce type de  $gain_{CD}$ , en utilisant une approche similaire à celle décrite dans PRISM [Frees et al., 2007], qui permet d'ajuster le  $gain_{CD}$  en fonction de la vitesse de l'utilisateur et de rattraper le décalage accumulé au cours du temps par un mécanisme de récupération basé sur la vitesse et le sens de déplacement de l'utilisateur. Cependant, bien que cette technique fonctionne bien avec des valeurs de  $gain_{CD} < 1$ , ce n'est pas le cas avec des valeurs de  $gain_{CD}$  élevées. Nous avons réalisé plusieurs pilotes et il est apparu que les participants étaient rapidement perdus dans la scène 3D du fait des variations de gains et du mécanisme de rattrapage du décalage, car il provoquait un saut brutal d'une orientation à une autre en raison du  $gain_{CD}$  élevé. C'est pour ces raisons que nous avons opté pour l'utilisation d'un  $gain_{CD}$  constant, n'entraînant donc pas de décalage et permettant de plus d'avoir une conformité en zéro.

En fonction des contraintes définies précédemment, il fallait appliquer un facteur de  $gain_{CD}$  permettant d'amplifier suffisamment l'amplitude de l'utilisateur. Nous avons calculé une valeur de  $gain_{CD}$  pour le lacet et pour le roulis répondant à ces critères : lorsque la tête de l'utilisateur est en position de repos, la caméra est à la position d'origine ; lorsque la tête de l'utilisateur est tournée à amplitude maximale sur l'un des axes, la caméra réalise une rotation de 180° sur un axe donné.

$$Gain_{CD}^{lacet} = \frac{A_{max}^{Cam}}{A_{max}^{lacet}} = \frac{180}{26} = 6.9$$

$$Gain_{CD}^{roulis} = \frac{A_{max}^{Cam}}{A_{max}^{roulis}} = \frac{180}{35} = 5.1$$

(a) Facteur de  $gain_{CD}$  sur le lacet

(b) Facteur de  $gain_{CD}$  sur le roulis

**FIGURE 5.15** – Le facteur de  $gain_{CD}$  permettant d'effectuer 180° de rotation est le ratio entre l'amplitude de rotation maximale de la caméra dans la scène 3D et l'amplitude maximale de la tête sur un axe donné.

À l'aide de ces valeurs de  $gain_{CD}$  nous avons conçu deux techniques d'interaction pour le contrôle orbital autour de l'axe vertical. La première utilise le lacet de la tête en entrée, tandis que la seconde utilise le roulis.

$$\begin{aligned} F_{lacet}(A^{lacet}) &= A^{lacet} \times Gain_{CD}^{lacet} \\ &= A^{lacet} \times 6.9 \end{aligned}$$

$$\begin{aligned} F_{roulis}(A^{roulis}) &= A^{roulis} \times Gain_{CD}^{roulis} \\ &= A^{roulis} \times 5.1 \end{aligned}$$

(a) Fonction de transfert sur le lacet

(b) Fonction de transfert sur le roulis

**FIGURE 5.16** – Fonctions de transfert pour le contrôle de la rotation exocentrique sur le lacet et le roulis

## 5.4 COMPARAISON DE TECHNIQUES

### 5.4.1 Étude utilisateur

Nous avons mené une étude afin d'évaluer les deux techniques précédemment définies sur une tâche de type Fitts. Dans cette étude nous comparons la précision et le temps de réalisation de la tâche.

#### Participants

12 participants (4 femmes) âgés de 23 à 36 ans ( $\mu = 27.4, \sigma = 3.8$ ) ont été recrutés via des listes de diffusion ou via le bouche-à-oreille. À la fin de l'expérience, les participants étaient récompensés avec des friandises. 5 participants sur les 12 ont également participé à l'une des deux études précédentes.

#### Appareillage

Le dispositif expérimental était constitué d'un ordinateur portable de type MacBook Pro branché à un écran externe de 17 pouces, d'un clavier et d'une souris (figure 4.12), comme dans les études précédentes (voir chapitre 4).

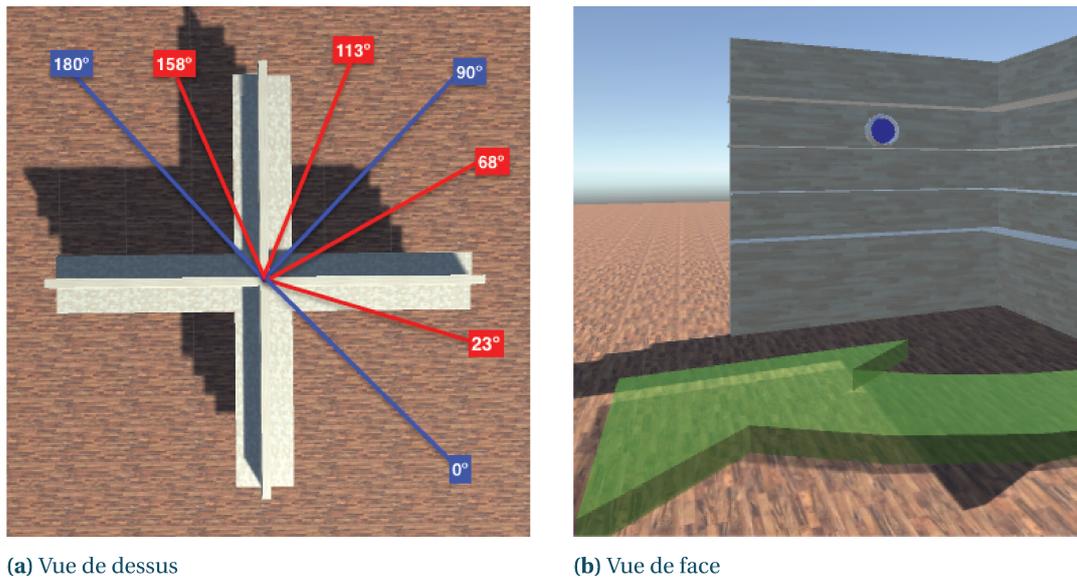
#### Procédure

La tâche à réaliser dans cette expérience s'inspire des tâches que l'on retrouve dans les logiciels de type *Floor planner*, dans lesquels l'utilisateur planifie l'aménagement de sa maison virtuelle en y déplaçant divers objets 3D (meubles, appareils électriques, etc.). De manière plus abstraite, l'utilisateur veut déplacer un objet 3D existant dans une autre position. Afin de simuler ce type de tâche, nous avons mis en place une scène 3D constituée de 4 murs munis d'étagères (figure 5.17b) positionnés orthogonalement les uns par rapport aux autres (figure 5.17a). La disposition des murs dans la scène visait à masquer la position exacte de la cible.

La procédure suivante nécessite de contrôler à la fois le point de vue et la sélection (voir figure 5.18) :

- Les participants commencent par réaliser une tâche de pointage en sélectionnant une balle positionnée sur le sol (voir figure 5.18) ;
- Ils orientent la vue (en utilisant soit le lacet, soit le roulis en fonction des conditions expérimentales) en direction de la cible qui est indiquée par une flèche verte au sol (voir figure 5.17b). Cette flèche permet d'indiquer aux participants la distance à parcourir ainsi que la direction dans laquelle opérer le mouvement ;
- Lorsque la tête des participants est alignée avec la cible, un halo vert apparaît autour de celle-ci ;
- Les participants réalisent une deuxième tâche de pointage en visant la cible à l'aide du curseur de la souris et en cliquant pour envoyer la balle sur la cible ;
- Enfin, les utilisateurs orientent la vue pour retourner en position d'origine et commencer l'essai suivant.

Un retour visuel indique à l'utilisateur après le lancer de la balle s'il a réussi à toucher la cible ou non. Un tir raté peut être dû soit à un non-alignement du point de vue avec la cible, soit à une erreur de visée sur la cible.



**FIGURE 5.17** – À gauche, la scène vue de dessus montrant les quatre quadrant dans lesquels la cible peut être positionnée ainsi que les distances angulaires. À droite, une vue prise de la caméra de point de vue montrant la flèche verte indiquant la direction du mouvement à effectuer et la cible à atteindre.

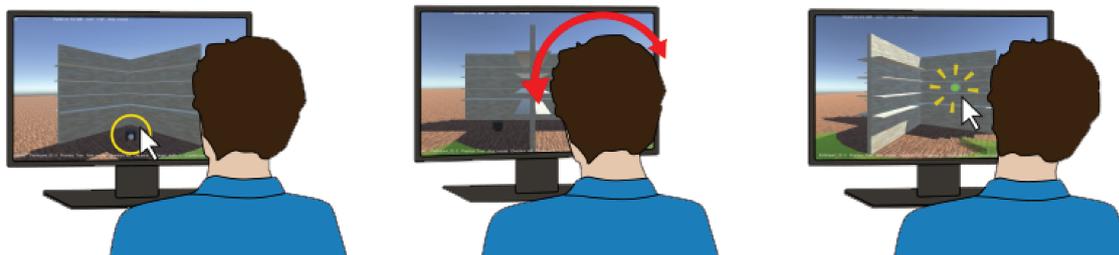
Nous contrôlons trois facteurs dans cette expérience :

- La **LARGEUR** de cible, qui définit le seuil angulaire à partir duquel l'utilisateur est considéré comme étant aligné avec la cible. Plus ce seuil est grand, moins la précision requise est importante et inversement. Nous avons choisi quatre valeurs proches des valeurs de  $gain_{CD}$  de nos deux techniques :  $4^\circ$ ,  $5.1^\circ$ ,  $6^\circ$  et  $6.9^\circ$  ;
- La **DISTANCE** angulaire de la cible par rapport à la position d'origine de l'utilisateur. Comme nous voulons réaliser des mouvements de caméra d'une amplitude maximale de  $180^\circ$ , nous avons choisi des distances proches et lointaines (voir figure 5.17a) :  $23^\circ$ ,  $68^\circ$ ,  $113^\circ$  et  $158^\circ$  ;
- La **DIRECTION** *droite* ou *gauche*, la cible pouvant être positionnée des deux côtés.

Nous avons dans un premier temps expliqué aux participants la tâche à réaliser. Les participants pouvaient ensuite s'entraîner sur les deux techniques avant de commencer l'expérience proprement dite. Pendant cette phase d'entraînement, les utilisateurs étaient libres de changer la sensibilité de la souris et la direction de la rotation par rapport aux mouvements de la tête (un mouvement de la tête à gauche pouvait selon leur choix orienter la caméra vers l'une des deux directions). Cependant, la plupart des participants étaient satisfaits des réglages d'origine. Sur les 12 participants, 11 ont choisi de conserver les réglages d'origine pour le roulis, tandis que 8 ont choisi ces réglages pour le lacet. L'expérience durait 1 heure et les participants étaient libres de prendre une pause quand ils le souhaitaient et nous les invitons à faire un break tous les 5 essais. À la fin de l'expérience, les participants devaient répondre à un questionnaire de type NASA TLX afin de mesurer la charge cognitive, la fatigue, les stratégies d'utilisation et facilité d'utilisation associées aux deux techniques étudiées.

### *Design expérimental*

Nous avons utilisé un plan intra-sujets et une analyse à mesures répétées. L'ordre des conditions était contrebalancé entre les participants et chaque condition était répétée 5 fois. Les variables indépendantes



**FIGURE 5.18** – L'utilisateur doit 1) ramasser une balle positionnée sur le sol de la scène 3D en utilisant la souris ; 2) orienter la vue (en utilisant soit le lacet, soit le roulis de la tête en fonction de la condition) dans la direction de la cible (qui est indiquée par une flèche verte au sol) ; 3) lancer la balle sur la cible en cliquant le bouton de la souris une fois que la vue est alignée avec la cible (cible avec un halo vert).

LARGEUR, DISTANCE et DIRECTION étaient contrebalancées entre les deux techniques. Pour chaque condition, les participants ont effectué 5 répétitions. Nous avons mesuré le temps de réalisation de la tâche (que nous séparons en deux : temps de sélection et temps de retour) et le taux de succès. Au total, le design expérimental comprenait 12 participants x 2 TECHNIQUES x 2 DIRECTIONS x 4 LARGEURS de cibles x 4 DISTANCES angulaires x 5 répétitions = 3840 essais.

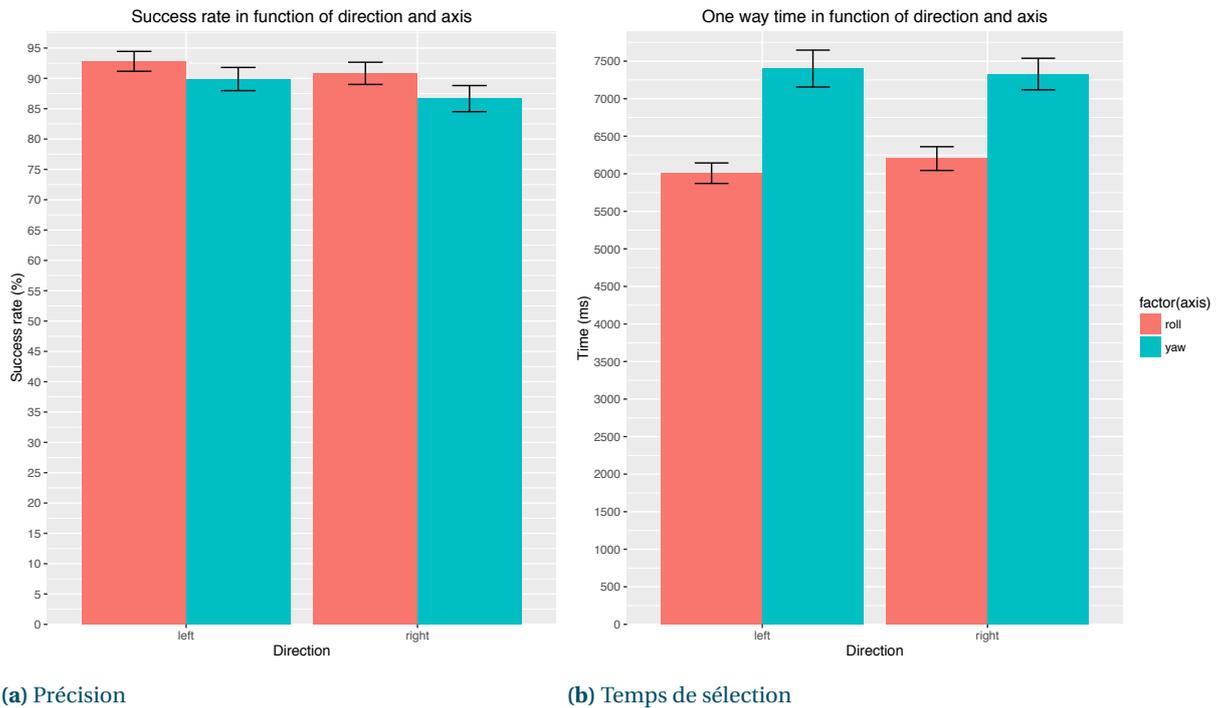
#### 5.4.2 Résultats et discussion

Nous avons considéré trois mesures pour évaluer les techniques : (1) le taux de précision, à savoir le nombre d'essais dans lesquels le participant a réussi à toucher la cible par rapport au nombre total d'essais ; (2) le temps de sélection, qui correspond au temps pris par le participant pour saisir la balle et la lancer sur la cible à partir de la position de départ ; (3) le temps de retour, qui correspond au temps pris par le participant pour retourner à sa position initiale après avoir lancé la balle sur la cible. En résumé, nous observons que la technique utilisant le Roulis (92%) est 4% plus précise que celle utilisant le Lacet (88%) (voir figure 5.19a). De plus, la technique utilisant le Roulis (4.0s) est également plus rapide en moyenne de 13% que la technique Lacet (4.6s) sur le temps de sélection (voir figure 5.19b). En revanche, aucune différence significative sur le temps de retour n'est observée entre le Lacet (1.5s) et le Roulis (1.6s).

Nous donnons ci-dessous le détail des mesures.

##### Précision

Une ANOVA a révélé un effet significatif de la TECHNIQUE sur la Précision ( $F_{1,11} = 13.2, p < 0.001$ ). Un test post-hoc de Tukey montre que les participants sont plus précis sur le Roulis (92%) que sur le Lacet (88%). ANOVA a aussi révélé un effet significatif de la LARGEUR de cible sur la Précision ( $F_{3,33} = 10.3, p < 0.0001$ ). Un test post-hoc de Tukey montre que les participants sont significativement moins précis sur la plus petite cible ( $4^\circ$  : 84% de précision) que sur les autres cibles ( $5.1^\circ$  : 92%,  $6^\circ$  : 91% et  $6.9^\circ$  : 92%). Enfin, un effet significatif de la DIRECTION sur la précision a été observé ( $F_{1,11} = 7.2, p < 0.01$ ). Un test post-hoc de Tukey montre que les participants sont plus précis sur la Gauche (91%) que sur la Droite (88%). Ce phénomène peut être dû à l'oeil directeur qui est la préférence du cerveau pour l'un des deux yeux. Un test de Miles Miles [1930] a révélé que 7 participants sur 12 avait un oeil directeur gauche et une analyse post-hoc suggère que les utilisateurs sont plus précis dans la direction correspondant à



**FIGURE 5.19** – À gauche, le taux de succès pour chaque technique en fonction de la direction. À droite, le temps de sélection pour chaque technique en fonction de la direction.

leur oeil directeur (91.6%,  $\sigma = 1.2$ ) par rapport à la direction opposée (88.6%,  $\sigma = 1.8$ ). Aucun effet de la DISTANCE n'a été observé sur le taux de précision.

### Temps de sélection

Une ANOVA a révélé un effet significatif de la TECHNIQUE sur le temps de sélection ( $F_{1,11} = 61.8$ ,  $p < 0.0001$ ). Un test post-hoc de Tukey montre que les participants sont plus rapides sur le Roulis (4.0s) que sur le Lacet (4.6s). ANOVA révèle aussi un effet significatif de la DISTANCE ( $F_{3,33} = 59.7$ ,  $p < 0.0001$ ) et de la LARGEUR ( $F_{3,33} = 24.1$ ,  $p < 0.0001$ ) sur le temps de sélection. Un test post-hoc de Tukey confirme que le temps de sélection augmente avec la distance et diminue avec la largeur de cible. Aucun effet de la DIRECTION sur le temps de sélection n'a pu être observé.

### Temps de retour

ANOVA révèle un effet significatif de la DISTANCE sur le temps de retour ( $F_{3,33} = 78.3$ ,  $p < 0.0001$ ). Un test post-hoc de Tukey confirme que le temps de retour augmente avec la distance. ANOVA révèle aussi un effet significatif de la DIRECTION ( $F_{3,33} = 7.8$ ,  $p < 0.01$ ). Un test post-hoc de Tukey montre que les participants sont plus rapides à gauche (1.5s) qu'à droite (1.6s) ce qui peut être dû à l'oeil directeur, comme expliqué précédemment. Aucun effet de la TECHNIQUE ou de la LARGEUR n'a été observé sur le temps de retour.

Mesures qualitatives

Un test de Kruskal-Wallis ne révèle aucun effet de la TECHNIQUE sur la demande physique, temporelle ou mentale de la tâche. Cependant, 8 participants sur 12 ont dit préférer la technique utilisant le roulis de la tête et 10 participants se sentaient plus rapides en utilisant le roulis qu'en utilisant le lacet. Par ailleurs, 10 participants ont trouvé la technique basée sur le lacet trop sensible.

Technique / Largeur	4				5.1				6				6.9			
Distance (°)	23	68	113	158	23	68	113	158	23	68	113	158	23	68	113	158
Temps de sélection																
Roulis Droite	3.3	4.0	4.9	5.2	3.4	4.2	4.4	5.1	3.2	4.0	4.0	4.1	2.9	3.8	4.2	4.4
Roulis Gauche	3.8	4.5	4.5	4.8	3.3	3.9	4.4	4.5	3.0	3.5	4.4	4.5	3.2	3.6	4.0	4.5
Roulis Total	3.5	4.2	4.7	5.0	3.3	4.0	4.4	4.8	3.1	3.7	4.2	4.3	3.0	3.7	4.1	4.4
Lacet Droite	4.1	5.4	5.6	5.8	3.7	4.9	4.7	5.1	3.8	4.3	4.5	4.4	3.4	4.4	4.7	4.3
Lacet Gauche	4.9	4.7	5.4	5.5	4.2	4.5	5.7	5.0	3.7	4.0	4.4	5.1	3.3	4.0	4.7	4.8
Lacet Total	4.5	5.1	5.5	5.6	4.0	4.7	5.2	5.0	3.7	4.2	4.4	4.8	3.4	4.2	4.7	4.6
Temps de retour																
Roulis Droite	1.2	1.7	1.8	1.9	1.3	1.6	1.7	1.9	1.0	1.9	1.6	1.7	1.1	2.0	1.7	1.8
Roulis Gauche	1.2	1.6	1.6	1.8	1.3	1.6	1.7	1.9	1.1	1.5	1.5	1.8	1.3	1.4	1.7	1.8
Roulis Total	1.2	1.7	1.7	1.9	1.3	1.6	1.7	1.9	1.1	1.7	1.6	1.8	1.2	1.7	1.7	1.8
Lacet Droite	1.2	2.1	1.7	1.8	1.2	1.7	1.6	1.9	1.1	1.7	1.8	1.7	1.2	1.8	1.8	1.6
Lacet Gauche	1.3	1.5	1.6	1.6	1.1	1.5	1.5	1.7	1.1	1.4	1.5	1.7	1.1	1.5	1.5	1.7
Lacet Total	1.2	1.8	1.7	1.7	1.2	1.6	1.6	1.8	1.1	1.6	1.7	1.7	1.1	1.7	1.7	1.6

TABLE 5.2 – Temps de sélection et temps de retour pour les techniques Roulis et Lacet dans chaque direction, pour chaque largeur de cible et distance.

5.4.3 Modèle

Nous présentons un modèle basé sur la loi de Fitts afin de modéliser les résultats obtenus. Nous définissons l'index de difficulté de la tâche  $ID$  comme étant  $\log(1 + \frac{D}{W})$ , où  $D$  est la distance angulaire et  $W$  la largeur de cible. Une régression linéaire produit les deux équations linéaires (figure 5.21) et montre l'évolution du temps de réalisation de la tâche par rapport à l'index de difficulté  $ID$ . Bien que le modèle semble consistant avec la loi de Fitts pour le Roulis, cela ne semble pas être le cas pour le Lacet. Une explication possible est que certains index de difficulté sont trop élevés. Une autre raison peut être que dans le cas du Lacet, l'écran sort du champ de vision du participant alors que ce n'est pas le cas avec le Roulis.

$$T(lacet) = 2.5 + 0.5 \times ID, R^2 = 0.78$$

$$T(roulis) = 1.7 + 0.6 \times ID, R^2 = 0.96$$

FIGURE 5.20 – Équations linéaires modélisant les résultats obtenus pour les deux techniques.

5.4.4 Conclusion

Dans la section *Création des techniques d'interaction* (section 5.3), nous avons créé deux techniques utilisant les mouvements de rotation de la tête (la première avec le lacet, la seconde avec le roulis) pour le contrôle du point de vue orbital en utilisant un  $gain_{CD}$  constant afin d'avoir une conformité en zéro. Nous avons comparé les performances (précision, temps) de ces deux techniques sur un

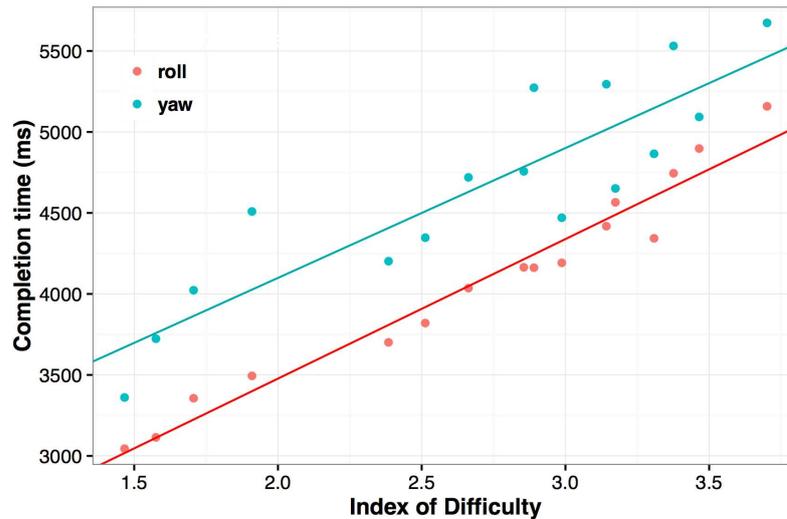


FIGURE 5.21 – Régression linéaire modélisant les rotations sur le roulis et le lacet.

Technique / largeur de cible	4	5.1	6	6.9
Roulis Gauche	88	92	95	95
Roulis Droite	85	92	93	95
Roulis Total	86	92	94	95
Lacet Gauche	88	93	89	90
Lacet Droite	81	90	87	88
Lacet Total	84	91	88	89

TABLE 5.3 – Taux de succès moyen pour chaque technique en fonction de la direction et de la largeur de cible

ensemble de tâches de sélection. L'expérience a montré une précision accrue et un temps de réalisation par essai moindre avec la technique utilisant le roulis. Ces résultats sont très intéressants, car peu intuitifs. En effet, bien que le lacet semble être le mouvement le plus adapté pour réaliser un contrôle de caméra orbital autour de l'axe vertical (mouvement de tête couramment effectué, bonne conformité entre le mouvement et le contrôle de caméra), la technique basée sur le roulis permet d'obtenir des performances supérieures. Un facteur permettant d'expliquer ces résultats est la différence de  $gain_{CD}$  entre les deux techniques. Avec un  $gain_{CD}$  supérieur (6.9) sur le lacet par rapport au roulis (5.1), l'amplitude de mouvement de l'utilisateur est accrue, mais sa précision est réduite. Avec une précision réduite, l'utilisateur a plus de mal à réaliser la tâche ce qui se traduit par une augmentation du temps de réalisation.

## 5.5 VALIDITÉ EXTERNE

### 5.5.1 Étude utilisateur

L'étude précédente a montré la supériorité de la technique basée sur le roulis tant au niveau de la précision (4% de précision en plus) que sur le temps de réalisation (13% de temps en moins sur la

partie sélection) des tâches. Afin de vérifier la validité externe de notre technique basée sur le roulis, nous avons décidé de comparer celle-ci au couple clavier + souris. Nous avons choisi une tâche de manipulation 3D courante pour contrôler le point de vue dans le but de déplacer un objet 3D dans la scène.

### Participants & appareillage

10 participants (2 femmes) âgés de 24 à 37 ans ( $\mu = 28.2, \sigma = 4.9$ ) ont été recrutés. Nous avons utilisé le même appareillage que celui utilisé dans l'étude précédente.

### Procédure

Les participants devaient réaliser une tâche de docking classique en déplaçant un objet 3D d'une position initiale à une position cible (voir figure 5.22). Nous avons utilisé la même scène 3D que dans l'expérience précédente. La tâche nécessitait à la fois de déplacer le point de vue et de contrôler la position de l'objet.

Les participants contrôlaient le point de vue orbital en utilisant soit le roulis de la tête, soit la souris ou le clavier. Les mouvements de tête sur le roulis permettaient de réaliser une rotation orbitale de  $180^\circ$  avec une valeur de  $gain_{CD}$  de 5.1 dans chaque direction, comme dans l'expérience précédente. Dans cette expérience, nous avons appliqué une hystérésis sur les valeurs d'entrée du roulis de la tête afin de réduire le bruit utilisateur et accroître la stabilité de l'entrée utilisateur. Ainsi, tant que la valeur d'entrée restait comprise dans un intervalle de 2 unités<sup>1</sup>, la valeur restait inchangée. Afin de nous placer dans des conditions réelles, nous avons tenté de reproduire (de manière empirique) au maximum le comportement du clavier et de la souris observé dans le logiciel de modélisation 3D *Blender*.

Ainsi, pour réaliser un mouvement orbital avec la souris, le participant doit cliquer sur la molette de la souris et exécuter un mouvement de glisser-déposer. Cependant, à la différence de *Blender*, les mouvements verticaux de la souris qui dans *Blender* permettent de contrôler la rotation orbitale sur le tangage (voir chapitre 2), étaient ignorés.

Pour le clavier, l'appui sur la touche 4 ou la touche 6 du pavé numérique entraînait une rotation de  $15^\circ$  par itération. En maintenant une touche du clavier, l'utilisateur pouvait ainsi réaliser un mouvement de  $360^\circ$  en 1 seconde.

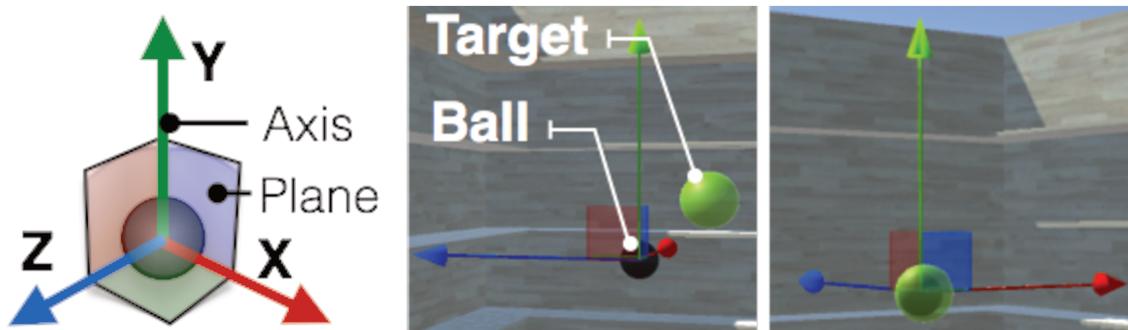
Enfin, avec le roulis de la tête, l'orientation du point de vue se faisait avec la fonction de transfert définie dans le chapitre précédent (chapitre 4).

Le déplacement de l'objet se faisait de la même manière que dans les logiciels de modélisation 3D (voir figure 5.22). L'objet est muni de trois axes, permettant au participant de contrôler la translation sur l'un des axes en tirant sur celui-ci. L'objet est également entouré de trois plans, permettant au participant de déplacer celui-ci sur deux dimensions en simultané. Enfin, en cliquant sur l'objet directement et en réalisant un glisser-déposer de la souris, le participant pouvait déplacer l'objet dans le plan parallèle à

---

1.  $[valeur_{hysteresis} - 0.5; valeur_{hysteresis} + 1.5]$  pour un déplacement vers la droite et  $[valeur_{hysteresis} - 1.5; valeur_{hysteresis} + 0.5]$  pour un déplacement vers la gauche,  $valeur_{hysteresis}$  étant un arrondi de la valeur en entrée

celui de la caméra de point de vue.



**FIGURE 5.22** – Le participant pouvait déplacer l'objet (ball) dans la cible (target) de trois manières différentes : 1) en utilisant les axes ( $x$ ,  $y$  et  $z$ ) ; 2) En utilisant les plans ( $xy$ ,  $yz$  et  $xz$ ) ; 3) ou en manipulant la boule directement dans le plan parallèle à celui de la caméra de point de vue.

Dans cette expérience, nous contrôlions trois facteurs :

- La **TECHNIQUE** : roulis de la tête, Souris/Clavier ;
- La **DISTANCE** angulaire des cibles par rapport à l'origine :  $68^\circ$ ,  $113^\circ$  et  $158^\circ$  ;
- La **DIRECTION** : droite ou gauche.

Les participants étaient libres de réaliser plusieurs essais d'entraînement avant chaque technique et pouvaient, comme dans l'expérience précédente, inverser le sens de rotation du point de vue. Des pauses étaient prévues entre chaque bloc afin de permettre aux participants de se reposer et éventuellement donner du feedback sur leurs impressions. À la fin de l'expérience, les participants devaient remplir un questionnaire de type NASA TLX pour chacune des trois techniques. L'expérience durait 45 min.

### *Design expérimental*

Nous avons utilisé un plan intra-sujets et une analyse à mesures répétées. L'expérience était composée de deux sessions.

Durant la première session, les participants devaient réaliser quatre blocs consécutifs avec chaque technique. L'ordre des techniques était contrebalancé entre les participants. Chaque bloc contenait 6 essais correspondant à 3 DISTANCES x 2 DIRECTIONS. L'ordre des essais variait avec les blocs et les participants.

Durant la seconde session, les participants étaient libres d'utiliser soit la tête, soit le couple souris/clavier, soit les deux en simultanément pour réaliser la tâche. Nous avons conduit cette seconde session afin d'observer les préférences des utilisateurs et de voir si ceux-ci combinaient les deux techniques. Nous avons choisi de conduire cette seconde session (appelée condition "libre") dans un bloc séparé afin d'éviter tout effet d'ordre.

Au total, le design expérimental comprenait 10 participants x 2 + 1 TECHNIQUES x 4 blocs x 2 DIRECTIONS x 3 DISTANCES angulaires de cibles = 720 essais.

### 5.5.2 Résultats et discussion

Dans cette expérience, nous avons utilisé trois mesures de temps afin d'analyser les résultats : (1) le temps d'alignement qui correspond au temps pris par le participant pour aligner l'objet avec la cible (en incluant le temps nécessaire au contrôle du point de vue) ; (2) le temps de retour correspondant au délai nécessaire à l'utilisateur pour retourner dans sa position originale après avoir aligné l'objet et la cible ; (3) le temps total qui correspond au temps nécessaire pour réaliser la tâche dans son intégralité.

En résumé, nous observons que le Roulis de la tête est plus rapide que Souris/Clavier sur le temps total (14.5%) (voir figure 5.23a), le temps d'alignement (14.7%) et le temps de retour (13.4%) (voir figure 5.23b). Nous n'avons cependant observé aucun effet significatif du BLOCK ou de la DIRECTION sur les mesures de temps. Avec la condition "libre", les utilisateurs ont été plus rapide (7s) qu'avec le Roulis de la tête (8.3s) et la condition Souris/Clavier (9.8s). Cependant, comme cette session est conduite après la première, il n'est pas possible de la comparer statistiquement à la précédente.

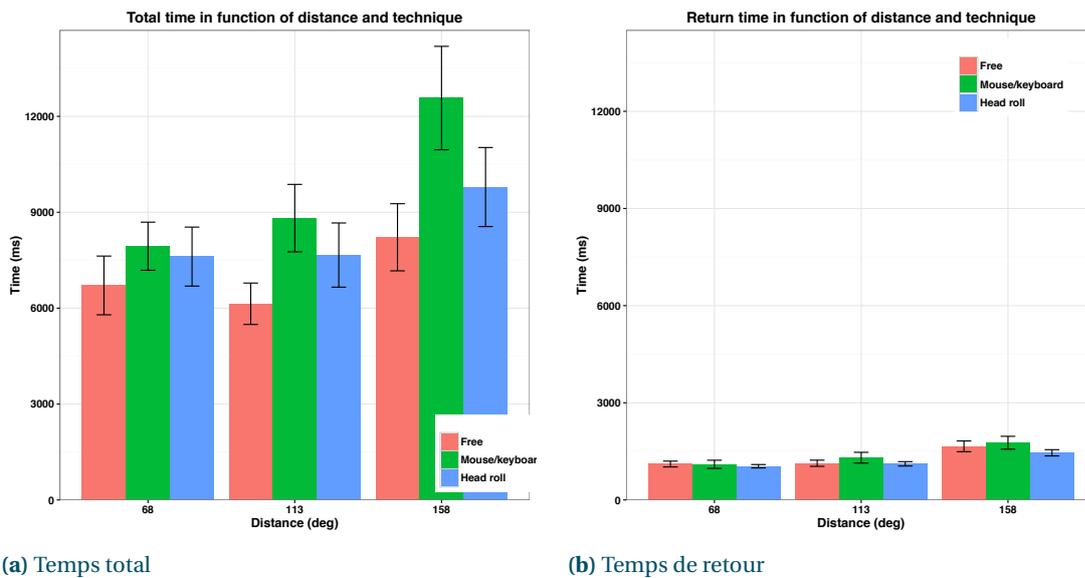


FIGURE 5.23 – À gauche, le temps total pour chaque technique en fonction de la distance angulaire. À droite, le temps de retour.

Nous donnons ci-dessous le détail des mesures.

#### Temps total

Une ANOVA révèle un effet significatif de la DISTANCE ( $F_{2,18} = 20.2, p < 0.001$ ) et de la TECHNIQUE ( $F_{1,9} = 9, p < 0.01$ ) sur le temps total. Un test post-hoc de Tukey montre que la plus grande distance angulaire ( $158^\circ : 11.2s$ ) prend significativement plus de temps que les deux autres distances ( $68^\circ : 7.8s; 113^\circ : 8.2s$ ). Un test post-hoc de Tukey révèle aussi que le Roulis de la tête (8.3s) est plus rapide que Souris/Clavier (9.8s).

### Temps d'alignement

Une ANOVA révèle un effet significatif de la DISTANCE ( $F_{2,18} = 15.3$ ,  $p < 0.001$ ) et de la TECHNIQUE ( $F_{1,9} = 7.1$ ,  $p < 0.01$ ) sur le temps d'alignement. Un test post-hoc de Tukey montre que la plus grande distance angulaire (158° : 9.6s) prend significativement plus de temps que les deux autres distances (68° : 6.7s; 113° : 7). Un test post-hoc de Tukey révèle aussi que le Roulis de la tête (7.1s) est plus rapide que Souris/Clavier (8.4s).

### Temps de retour

Une ANOVA révèle un effet significatif de la DISTANCE ( $F_{2,18} = 36.2$ ,  $p < 0.001$ ) et de la TECHNIQUE ( $F_{1,9} = 12$ ,  $p < 0.001$ ) sur le temps de retour. Un test post-hoc de Tukey montre que la plus grande distance angulaire (158° : 1.6s) prend significativement plus de temps que les deux autres distances (68° : 1.1s; 113° : 1.2s). Un test post-hoc de Tukey révèle aussi que le Roulis de la tête (1.2s) est plus rapide que Souris/Clavier (1.4s).

### Mesures qualitatives

Un test de Kruskal-Wallis ne révèle aucun effet de la TECHNIQUE sur la charge cognitive, la fatigue et la facilité d'utilisation.

Technique / Distance	68°			113°			158°		
	Total	Align.	Retour	Total	Align.	Retour	Total	Align.	Retour
Roulis Droite	7.3	6.3	1.0	7.4	6.2	1.2	8.9	7.4	1.5
Roulis Gauche	7.9	6.8	1.0	7.9	6.9	1.0	10.7	9.3	1.4
Roulis Total	7.6	6.6	1.0	7.7	6.5	1.1	9.8	8.3	1.4
Souris/Clavier Droite	8.1	7.0	1.1	8.7	7.4	1.3	12.9	11.0	1.9
Souris/Clavier Gauche	7.8	6.7	1.1	8.9	7.7	1.3	12.2	10.6	1.6
Souris/Clavier Total	7.9	6.8	1.1	8.8	7.5	1.3	12.6	10.8	1.8

TABLE 5.4 – Temps total, temps d'alignement et temps de retour pour les techniques Roulis et Clavier/Souris dans chaque direction.

### Post-expérience

Au cours de la deuxième session, dite session "libre", 8 participants ont utilisé le Roulis de la tête pour contrôler le point de vue : 6 parmi les 8 ont utilisé à la fois le Roulis de la tête et le Clavier tandis que 2 ont utilisé le Roulis de la tête et la Souris. Ces participants ont justifié leur choix en expliquant que la combinaison Roulis avec le Clavier ou la Souris permettait d'être plus rapide et augmentait le confort en réduisant le déplacement à effectuer sur le Roulis. Parmi les participants n'ayant pas utilisé le Roulis, l'un d'eux a utilisé uniquement la Souris tandis que l'autre n'a utilisé que le Clavier.

### 5.5.3 Conclusion

À la suite de la première expérience (section 5.4), nous voulions déterminer si la technique basée sur le roulis était plus efficace que le couple souris/clavier traditionnellement utilisé dans les applications 3D (validité externe). Nous avons donc réalisé une expérience afin de comparer les performances sur une

tâche de docking entre notre technique roulis et le couple Souris/clavier. À l'issue de la première phase de cette expérience, les résultats montrent que le roulis permet de réaliser la tâche plus rapidement que le couple Souris/clavier. Dans une seconde phase, nous avons laissé les participants choisir la modalité souhaitée pour le contrôle du point de vue. Une majorité de participants ont continué à utiliser les mouvements de la tête pour le contrôle du point de vue et certains y ont couplé le clavier afin de gagner en rapidité et en confort en réduisant l'amplitude de rotation à effectuer avec la tête. Globalement, les participants ont été plus rapides avec la technique libre, bien que celle-ci ne puisse être statistiquement comparée avec les deux autres puisque les mesures ont été effectuées au cours de deux expériences différentes.

## 5.6 APPLICATIONS

Afin d'illustrer et montrer la polyvalence de notre technique d'interaction, nous avons sélectionné et intégré celle-ci au sein de plusieurs applications allant de la modélisation au son 3D dans lesquelles notre technique d'interaction pourrait permettre à l'utilisateur de gagner en performances.

### 5.6.1 Modalités d'entrée

Afin d'illustrer notre technique sur des applications de bureautique dans un environnement de travail standard, nous avons utilisé des technologies plus abordables que le système de suivi infrarouge utilisé pendant les expériences utilisateurs. Ainsi, pour les deux premières applications, une webcam a été utilisée comme modalité d'entrée et les six degrés de liberté de la tête ont été obtenus avec une technologie de suivi du visage (Visage|SDK FaceTrack<sup>1</sup>). Pour les deux dernières applications, nous avons utilisé une centrale inertielle installée sur un casque audio servant de support afin de mesurer les rotations de la tête.



FIGURE 5.24 – À gauche, la webcam utilisée pour nos applications. À droite, la centrale inertielle ainsi que l'Arduino Nano utilisés pour mesurer les déplacements de l'utilisateur.

### 5.6.2 Room planner

Un *Room planner* est un type d'application permettant de prévoir l'aménagement d'une pièce en permettant à l'utilisateur de préciser la géométrie de celle-ci et de tester différents positionnements d'objets (meubles, étagères, etc.). Dans les *Room Planner* classiques, l'utilisateur doit souvent alterner entre des tâches de contrôle du point de vue et la manipulation d'objets 3D. En revanche, dans notre

1. <http://visagetechnologies.com>

application, le point de vue est couplé avec l'orientation de la tête, ce qui laisse les mains libres pour manipuler les objets de la scène ou réaliser d'autres tâches, par exemple dessiner (voir figure 5.25). Nous avons utilisé deux types de contrôle de caméra : une rotation orbitale permettant d'avoir un aperçu global de l'agencement des pièces et une rotation égocentrique pour visualiser les pièces à la première personne, comme si l'utilisateur se trouvait dans celles-ci. La transition entre les deux contrôles de caméra se fait en zoomant/dézoomant à l'aide de la molette de la souris. Nous avons choisi d'implémenter cette application car la tâche réalisée dans celle-ci est très proche du type de tâches utilisées au cours de nos deux dernières études et nous souhaitions avoir un aperçu de notre technique dans un contexte plus réaliste. L'implémentation a été effectuée sous *Unity*, un moteur de jeux vidéo multi-plateforme.

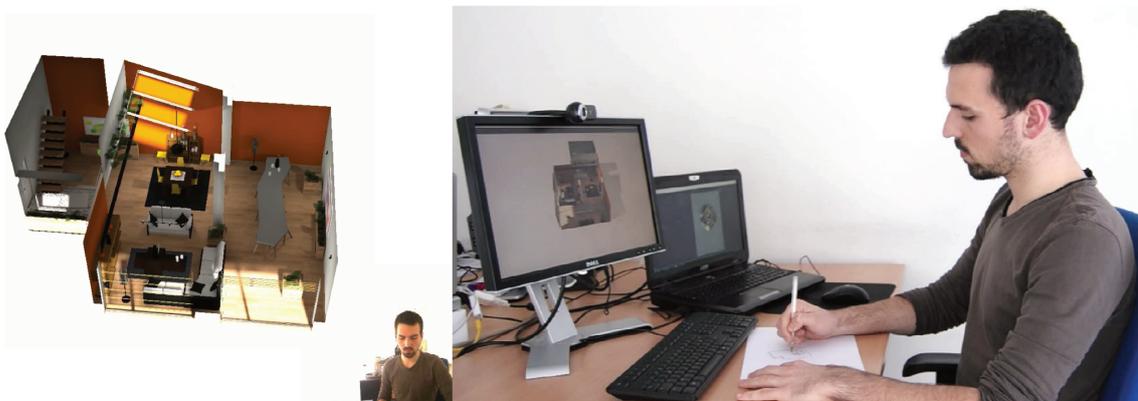


FIGURE 5.25 – Contrôle du point de vue avec la tête dans le contexte d'un *Room Planner*. L'utilisateur garde les mains libres pour réaliser d'autres tâches, par exemple dessiner.

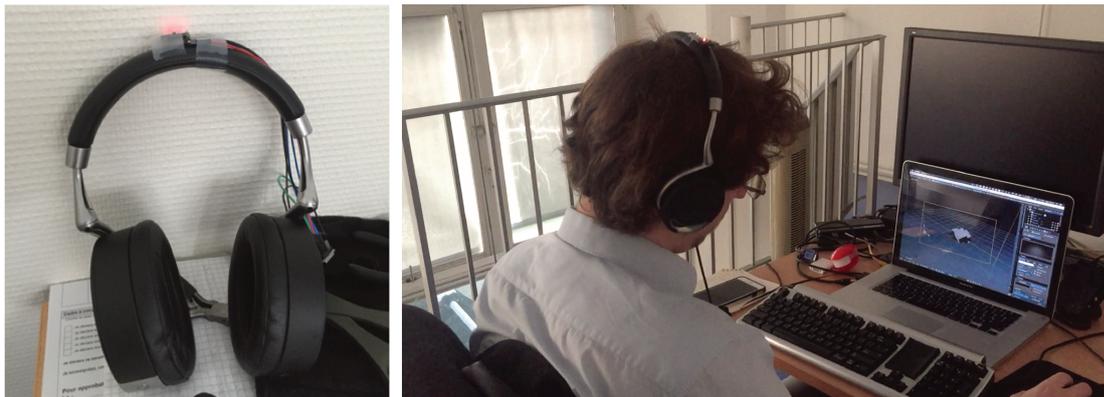
### 5.6.3 Logiciels de modélisation 3D

Depuis le début de ces travaux, nous avons assimilé la création de trajectoires 3D à une tâche de modélisation, c'est pourquoi nous avons voulu intégrer notre technique dans un logiciel de modélisation 3D. Nous avons donc réalisé une implémentation permettant de contrôler le point de vue orbital dans le moteur de jeux vidéo *Unity* (voir figure 5.26), utilisé dans nos deux dernières études et dans la création du *Room planner*. Nous avons pour cela utilisé une webcam ainsi que le plugin pour *Unity* fourni avec le SDK de *FaceTrack*, la technologie de suivi de visage décrite à la section 5.6.1.



FIGURE 5.26 – Contrôle du point de vue dans le moteur de jeux vidéo *Unity* avec le roulis de la tête.

Le suivi du visage avec la webcam s'avérant relativement peu précis, nous avons essayé une autre approche consistant à utiliser une centrale inertielle pour mesurer l'orientation de la tête (section 5.6.1). Avec celle-ci, nous avons implémenté un plugin (voir figure 5.27) pour le logiciel de modélisation 3D *Blender* permettant de contrôler le point de vue orbital à l'aide du roulis de la tête. Nous avons pour ce faire utilisé l'API de *Blender* en Python. Du côté de *Blender*, un serveur OSC écoute sur un port donné les valeurs entrantes. Les valeurs de l'IMU sont récupérées au niveau de l'Arduino puis renvoyées vers l'ordinateur ou un client OSC envoie les valeurs vers le serveur OSC de *Blender*. Les valeurs reçues dans *Blender* sont ensuite traitées dans un script python pour réaliser du contrôle orbital dans la scène 3D.



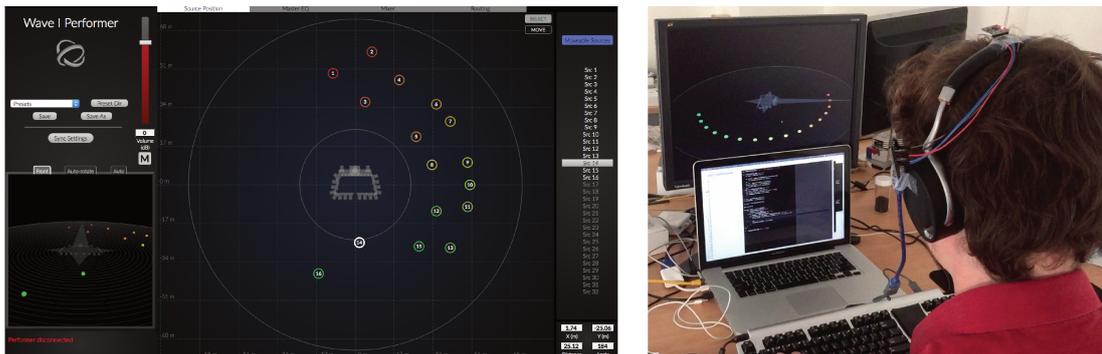
**FIGURE 5.27** – À gauche, contrôle du point de vue dans le logiciel *Blender* en utilisant le roulis de la tête. À droite, la centrale inertielle installée sur un casque audio.

#### 5.6.4 Application au son 3D

Afin d'augmenter la perception de la profondeur des ingénieurs du son dans le contexte du positionnement de sources sonores dans l'espace et du tracé de trajectoires 3D, nous avons intégré notre technique de contrôle du point de vue dans un logiciel de son 3D existant. Nous avons suivi une approche identique à celle du plugin *Blender* (utilisation d'une centrale inertielle, protocole OSC) pour intégrer le contrôle du point de vue orbital dans le logiciel *Performer* développé par *Sonic Emotion* et utilisé par *Radio France* afin de contrôler le positionnement des sources sonores dans l'espace (figure 5.28). Ce logiciel constitué de plusieurs vues (une vue de haut et une vue en perspective représentant le studio et la position des sources sonores au sein de celui-ci) permet à l'ingénieur du son de déplacer les sources dans l'espace du studio à l'aide d'un glisser-déposer de la souris. Cependant, le placement des sources dans un espace en trois dimensions se fait nécessairement en plusieurs phases de contrôle du point de vue et de déplacement d'objet. Avec notre technique, l'ingénieur du son sélectionne une source et peut déplacer celle-ci de manière simultanée avec la souris. Nous avons présenté ce prototype aux ingénieurs du son de *Radio France* qui l'ont trouvé très prometteur. Il faudra cependant mener des études avec eux pour éventuellement paramétrer notre technique en fonction de leurs contraintes (activation, valeur de gain, etc.).

## 5.7 CONCLUSION

Dans le chapitre 4, nous avons déterminé plusieurs paramètres nécessaires à la création de techniques d'interaction pour le contrôle du point de vue. Dans un premier temps, nous avons conduit une étude



**FIGURE 5.28** – À gauche, vue principale du logiciel *Performer* utilisé par les ingénieurs du son de *Radio France* pour contrôler le positionnement des sources sonores dans l'espace. À droite, utilisation des mouvements de la tête pour contrôler le point de vue orbital laissant les mains libres.

(section 4.3) afin de comprendre comment les utilisateurs associent de manière spontanée un contrôle de caméra à un mouvement de la tête et des épaules. Nous avons observé que ceux-ci proposaient de contrôler la rotation orbitale à l'aide du lacet ou du roulis de la tête. En nous basant sur ce résultat, nous avons mené deux études préliminaires préalables à l'élaboration de nos techniques d'interaction. Nous avons ensuite conduit une deuxième étude (section 4.4) afin de déterminer l'amplitude de mouvement confortable sur le lacet et le roulis de la tête. Enfin, dans une troisième étude nous avons déterminé le plus petit mouvement de tête contrôlable par l'utilisateur (section 4.5) sur le lacet et le roulis.

Dans ce chapitre nous avons commencé par présenter un certain nombre de définitions nécessaires à la compréhension des travaux académiques existants sur les fonctions de transfert (section 5.1). Dans la section suivante (5.2), nous avons présenté un ensemble de fonctions de transfert pour le contrôle du point de vue classées en fonction du type de  $gain_{CD}$  utilisé : constant ou variable. Ensuite, nous avons motivé et présenté (section 5.3) la conception de deux techniques d'interaction (une sur le lacet, l'autre sur le roulis) utilisant un  $gain_{CD}$  constant en nous basant sur les résultats obtenus dans le chapitre précédent.

Ces deux techniques ont été comparées à la section 5.4 sur une tâche de sélection de cibles afin de déterminer laquelle des deux était la plus précise et la plus rapide. Les résultats de l'expérience ont montré que la technique basée sur le Roulis était à la fois plus précise (4%) et plus rapide (13%) que la technique basée sur le Lacet. Une explication avancée pour expliquer ces différences de performance est que les utilisateurs disposent d'une amplitude de mouvement plus large sur le Roulis que sur le Lacet (voir chapitre 4). Ainsi, pour atteindre une amplitude de  $180^\circ$ , la technique basée sur le Lacet nécessite un  $gain_{CD}$  de 6.9, supérieur à celui de la technique basée sur le Roulis (5.1), ce qui entraîne une perte de précision et par la même occasion des temps de réalisation accrus.

Dans une seconde expérience (section 5.5), nous avons comparé la technique basée sur le Roulis avec la baseline la plus utilisée dans l'industrie, à savoir le couple Souris/Clavier, sur une tâche de docking courante en modélisation 3D. Le but de cette seconde expérience était de vérifier la validité externe de notre technique et son appropriation par des utilisateurs novices. Les résultats de l'expérience montrent que la technique Roulis est en moyenne 14% plus rapide que la technique Souris/Clavier sur le temps total d'un essai, bien que les utilisateurs soient particulièrement habitués à utiliser ce type de dispositif. Durant une seconde session dite session "libre", nous avons proposé aux participants de combiner leurs

modalités préférées parmi les modalités (Roulis, Souris, Clavier) précédemment utilisées. Une majorité des participants (8/12) a préféré l'utilisation du Roulis pour contrôler le point de vue en couplant celui-ci à la Souris ou au Clavier afin d'augmenter la vitesse de rotation du point de vue et réduire le déplacement de la tête.

Enfin, plusieurs implémentations de notre technique de contrôle du point de vue ont été réalisées (*Room planner* et logiciels de modélisation 3D) afin d'illustrer de possibles applications. Nous avons également pu interfacier notre technique avec le logiciel *Performer* utilisé par les ingénieurs du son de *Radio France* pour contrôler la position des sources sonores dans le studio 105 (voir chapitre 2) de la *Maison de la Radio*. L'intégration de notre technique de contrôle du point de vue avec l'outil existant devrait permettre aux ingénieurs du son de gagner en efficacité lors du placement des sources tout en minimisant la phase d'apprentissage. Nous avons pour l'instant intégré notre technique de contrôle du point de vue avec l'outil utilisé durant les concerts "live" car c'était l'usage pour lequel notre logiciel était susceptible d'apporter la plus grande valeur ajoutée du point de vue des ingénieurs du son. Dans une seconde étape, il sera donc nécessaire d'entreprendre de nouvelles discussions avec les ingénieurs du son pour intégrer notre technique aux autres logiciels audio utilisés pour le montage du son 3D, en particulier ceux utilisés pour la diffusion sur la toute nouvelle plateforme internet *Hyperradio*<sup>1</sup>.

---

1. <http://hyperradio.radiofrance.fr>

---

## CONCLUSION

En conclusion de cette thèse, nous présentons d'abord nos contributions, qui concernent la définition d'un espace de conception des techniques et dispositifs pour le tracé de trajectoires 3D et la création d'une technique d'interaction pour le contrôle du point de vue orbital à partir des mouvements de la tête. Nous présentons ensuite les perspectives à moyen et long termes.

### 6.1 CONTRIBUTIONS DE LA THÈSE

Dans ce chapitre nous résumons nos contributions. Ces contributions sont à la fois conceptuelles (espace de conception pour le tracé de trajectoires), techniques (création d'une technique d'interaction et intégration dans *Blender*) et empiriques (études utilisateurs). Nous détaillons leur réalisation ci-après.

#### 6.1.1 Contribution conceptuelle

Notre première contribution est la création d'un espace de conception (chapitre 2) pour la création et l'édition de trajectoires en trois dimensions. Face à la quantité importante de travaux existant dans le domaine du tracé de trajectoires et à leur dissémination au sein de plusieurs communautés scientifiques (notamment les communautés IHM et graphique), cet espace a pour objectif d'une part de rendre compte de la diversité des travaux existants et d'autre part de faciliter leur classification.

Cet espace de conception s'appuie sur la définition d'une modalité comme étant un couple *<dispositif, langage>* [Nigay et Coutaz, 1996]. Le premier axe *Dispositif*, permet de distinguer les dispositifs d'entrée (souris + clavier, souris augmentées, surfaces tactiles, interaction en l'air et tangible) des dispositifs de sortie (écrans traditionnels, écrans stéréo/HMD, retour haptique). Le second axe, *Langage*, permet de distinguer trois approches pour la création de trajectoires :

- l'utilisation de modèles de courbe, qui est actuellement utilisée dans les outils de modélisation 3D. L'utilisateur sélectionne un modèle de courbe dans un menu et l'ajoute à la scène 3D ;
- l'approche séquentielle, qui permet de tracer une trajectoire en deux temps, en traçant celle-ci sur deux plans 2D différents ;
- la définition de contraintes, qui impose certains critères au tracé de la trajectoire.

Puisque l'une des problématiques de cette thèse est de proposer de nouvelles techniques pour créer du

contenu audio 3D, nous présentons également les approches utilisées dans les logiciels de son 3D et classifions celles-ci au sein de notre espace de conception.

### 6.1.2 Contribution technique

Notre seconde contribution concerne la conception d'une technique d'interaction pour le contrôle du point de vue orbital. Celle-ci permet de renforcer la perception de profondeur dans la scène 3D en tenant compte des contraintes matérielles des ingénieurs du son (écran 2D, pas de casque de réalité virtuelle) et vise à simplifier le contrôle du point de vue. Elle consiste à contrôler le point de vue orbital d'une scène 3D en utilisant les mouvements de la tête, et plus précisément en utilisant le roulis. La fonction de transfert proposée utilise un  $gain_{CD}$  constant, offre une amplitude de rotation de 360 degrés et maintient la conformité en zéro.

Cette technique d'interaction a plusieurs avantages. Elle facilite l'inspection rapide d'objets 3D et offre une meilleure perception de la profondeur. De plus, elle laisse les mains libres pour d'autres contrôles (manipulation, sélection d'outils, etc.) à la souris et au clavier. Enfin, elle est adaptée à l'environnement des ingénieurs du son, car elle fonctionne avec un écran 2D et ne nécessite pas d'équipement coûteux.

Nous avons réalisé cette technique en utilisant une centrale inertielle et d'un Arduino comme dispositif d'entrée et l'avons intégrée dans le logiciel *Blender* à l'aide de son API en python.

### 6.1.3 Contribution empirique

Sur le plan empirique, nous avons conduit cinq études afin de guider et évaluer notre technique d'interaction pour le contrôle du point de vue orbital :

1. une étude par élicitation afin de comprendre comment les utilisateurs associent de manière spontanée un mouvement du corps (tête ou épaules) à un contrôle de caméra, montrant que le lacet et le roulis sont les mouvements de tête les plus adaptés pour contrôler la rotation orbitale;
2. une étude qui a permis d'évaluer la plus petite rotation réalisable sur le roulis et le lacet de la tête à une amplitude de  $1^\circ$  ;
3. une étude pour déterminer le plus grand mouvement confortable réalisable de manière contrôlée sur le lacet ( $25^\circ$ ) et le roulis ( $35^\circ$ ) de la tête.
4. une étude pour comparer les performances de deux fonctions de transfert et qui montre que la fonction basée sur le roulis permet de réaliser la tâche 13% plus rapidement que la fonction basée sur le lacet et est également 4% plus précise ;
5. une étude pour évaluer la validité externe de notre technique face au couple souris/clavier sur une tâche de docking. Les résultats révèlent que notre technique d'interaction permet de réaliser la tâche 14.5% plus rapidement qu'avec le couple clavier/souris.

## 6.2 PERSPECTIVES

Les perspectives liées à cette thèse sont réparties en deux catégories : les perspectives à moyen terme et les perspectives à long terme.

### 6.2.1 Perspectives à moyen terme

#### *Combinaison de modalités et utilisation des épaules*

Nous envisageons d'étudier comment combiner les rotations de la tête (lacet et roulis) afin de gagner en amplitude de mouvement puisque nous avons deux modalités d'entrée pour une unique rotation. Deux approches ont été envisagées :

- avec l'approche *additive*, les espaces moteurs du lacet et du roulis de la tête seraient additionnés, chacun avec leur  $gain_{CD}$  propre, afin de gagner en espace moteur (voir équation ci-dessous). Ainsi, l'utilisateur pourrait commencer par réaliser une rotation sur le roulis et terminer celle-ci sur le lacet et gagnerait ainsi en amplitude de mouvement ;

$$f(\text{lacet}, \text{roulis}) = gain_{\text{lacet}} \cdot valeur_{\text{lacet}} + gain_{\text{roulis}} \cdot valeur_{\text{roulis}}$$

- avec l'approche *multiplicative*, l'un des axes de rotation de la tête, typiquement celui ayant la plus faible amplitude de mouvement, pourrait servir à moduler le  $gain_{CD}$  de notre fonction de transfert et ainsi permettre de gagner en amplitude de mouvement ou en précision, en fonction de la valeur de  $gain_{CD}$  utilisée (voir équation ci-dessous).

$$f(\text{lacet}, \text{roulis}) = gain(\text{lacet})_{CD} \cdot valeur_{\text{roulis}}$$

Combiner deux rotations de la tête nécessiterait d'être en mesure de séparer le roulis du lacet. Cependant, toute rotation de la tête, quel que soit l'axe de rotation principale sur laquelle est effectuée la rotation, n'est jamais une rotation pure sur un seul axe. Au contraire, celle-ci contient à la fois de la rotation sur le lacet, le roulis et le tangage. Il s'agira donc de différencier les rotations volontaires de l'utilisateur de celles liées aux contraintes physiologiques. Un autre inconvénient potentiel est la réduction de l'amplitude de mouvement sur un axe, après une rotation sur un autre axe. Nous avons évalué l'amplitude confortable maximale sur le roulis à 35° et 25° sur le lacet. En combinant les deux axes, obtient-on 60° ou l'amplitude de mouvement est-elle réduite ? Il faudrait mener une étude pour comparer l'amplitude de mouvement sur le lacet et le roulis pris indépendamment et l'amplitude de mouvement en combinant les deux.

Une alternative à la combinaison des rotations de la tête est d'utiliser les épaules.

Dans notre étude par élicitation, nous avons proposé d'utiliser les épaules afin d'étendre la bande passante interactionnelle. Plusieurs participants ont utilisé cette dimension pour effectuer des rotations. Nous pensons que cette modalité gagnerait à être étudiée plus en détail. En effet, l'utilisation des épaules est intéressante, car peu utilisée aujourd'hui. Elle permettrait de disposer d'un espace moteur additionnel. Ainsi, en combinant les mouvements des épaules avec les mouvements de la tête, il serait possible en théorie de gagner en amplitude et par conséquent en précision. En effet, l'utilisateur pourrait par exemple utiliser les épaules pour effectuer une rotation de large amplitude et utiliser la tête pour

ajuster celle-ci de manière précise. Une autre utilisation pourrait être de contrôler la rotation orbitale horizontale à l'aide des épaules et d'assigner un autre contrôle aux rotations de la tête, ce qui permettrait de combiner plusieurs mouvements de caméra de manière simultanée. Enfin, l'utilisation des épaules offre plus de facilité aux utilisateurs en leur permettant de conserver l'écran dans leur champ de vision.

En revanche, l'inconvénient de ces deux approches (combinaison lacet + roulis et combinaison tête + épaule) est leur non-conformité en zéro. Il faudrait donc prévoir un mécanisme de débrayage, permettant à l'utilisateur de se replacer dans sa position initiale.

### *Débrayage du contrôle du point de vue et activation*

L'utilisation d'un mécanisme de débrayage est généralement la conséquence d'un espace moteur insuffisant. Elle a typiquement pour effet une perte de performance, par exemple sur des tâches de pointage [Casiez et al., 2007]. Notre technique de contrôle du point de vue a pour avantage de ne pas nécessiter de mécanisme de débrayage pour étendre l'espace moteur puisque la valeur du gain tient compte de l'amplitude de mouvement maximale de l'utilisateur. En revanche, une application du débrayage, suggérée par plusieurs participants, serait de permettre à l'utilisateur de repositionner sa tête face à l'écran. Ceci peut être utile si l'utilisateur doit réaliser une tâche prenant un temps significatif en plaçant sa tête à une orientation éloignée de celle d'origine. Avec un mécanisme de débrayage, l'utilisateur serait en mesure de changer de position d'origine et gagner en confort d'utilisation. L'activation du débrayage pourrait se faire en appuyant sur une touche du clavier.

Une autre problématique qui rejoint celle du débrayage que nous n'abordons pas dans le chapitre précédent est la question de l'activation. Dans un contexte réel, l'activation permanente de la technique peut parfois s'avérer gênante pour l'utilisateur. Plusieurs approches pour activer le contrôle du point de vue seront envisagées : l'utilisation d'une zone de repos ou l'utilisation d'un délimiteur.

Dans le cas de la zone de repos, une zone spécifique serait établie au sein de laquelle tout mouvement de tête de l'utilisateur serait ignoré. En revanche, tout mouvement hors de cette zone serait pris en compte pour le contrôle du point de vue. L'inconvénient de cette dernière approche est qu'elle nécessite de prendre en compte l'amplitude de la zone de repos définie dans le calcul de la fonction de transfert. En réduisant l'espace moteur de l'utilisateur et par conséquent l'amplitude de mouvement disponible, on diminue potentiellement la précision. Il faudrait conduire une étude utilisateur afin de déterminer la largeur de cette zone de repos.

Dans le cas du délimiteur, l'utilisateur aurait par exemple, à appuyer sur l'une des touches du clavier pour activer la technique, à l'inverse d'un débrayage.

### *6.2.2 Perspectives à long terme*

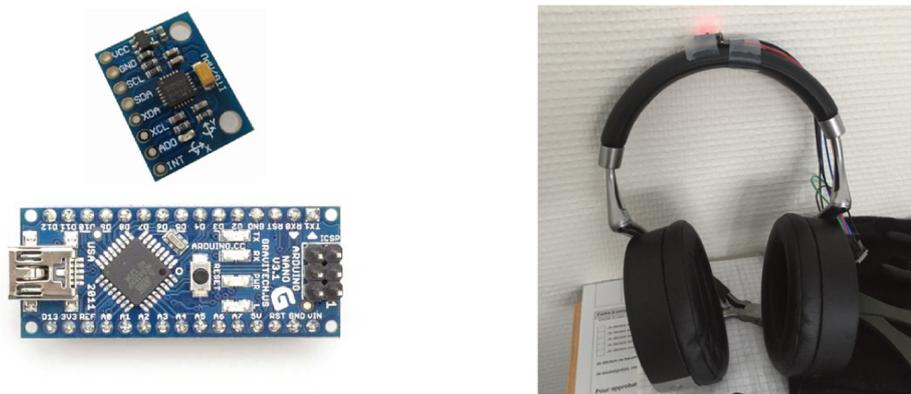
Bien que notre dernière étude pour évaluer la validité externe de notre technique se soit révélée concluante, il serait utile de mener une étude avec des experts dans le domaine de la modélisation 3D ou des compositeurs habitués aux créations musicales utilisant du son 3D. Ceci nous permettrait d'évaluer notre technique de contrôle du point de vue sur une plus longue durée et sur une véritable tâche, et ainsi d'évaluer la robustesse de la technique ainsi que son acceptabilité. En effet, bien que la mise au point de cette technique ait été réalisée avec plus d'une trentaine de participants, une grande

majorité de ceux-ci sont issus de notre établissement et ne sont donc pas nécessairement représentatifs des utilisateurs finaux (habitudes, âge). Ainsi, les paramètres (amplitude confortable, résolution utile de la tête) que nous avons déterminés dans le chapitre 4 pourraient varier.

Il faudrait également réfléchir au type de dispositif d'entrée à utiliser pour contrôler le point de vue. Un dispositif de suivi externe tel que la Kinect est souvent sujet à des erreurs de détection dès lors que plusieurs personnes entrent dans son champ de vision et nécessite par conséquent un large espace pour interagir. L'utilisation d'une webcam, bien que permettant d'illustrer le concept (comme montré à la section *Applications* du chapitre 5), n'est pas suffisamment robuste et comporte l'inconvénient d'être particulièrement sensible au bruit.

Une autre approche, dont la réalisation a été présentée au chapitre 5, consiste à utiliser une centrale inertielle portée par l'utilisateur, en l'intégrant par exemple à un casque audio comme illustré à la figure 6.1. L'intérêt de cette approche vient du fait qu'elle fournit une assez bonne précision tout en étant robuste, en particulier lorsque les conditions lumineuses sont variables, comme par exemple dans un studio. Elle est de plus très avantageuse au niveau coût ( $\approx 10$  euros pour une carte de type Arduino et une centrale inertielle).

Cependant, cette approche impose à l'utilisateur de porter un dispositif, ce qui n'est pas forcément adéquat dans tous les cas de figure. Il faudrait en particulier déterminer si ce type de dispositif est compatible avec l'environnement de travail des ingénieurs du son (concert live ou montage studio) et le cas échéant interfacé celui-ci dans leurs outils quotidiens. Une fois le dispositif intégré, il faudrait conduire une étude centrée utilisateur afin de recueillir des retours d'expérience sur les atouts et les inconvénients de cette approche et proposer plusieurs itérations du dispositif et de son intégration logicielle.



**FIGURE 6.1** – À gauche, la centrale inertielle ainsi que l'Arduino Nano utilisés pour mesurer les déplacements de l'utilisateur. À droite, la centrale inertielle installée sur un casque audio



---

## BIBLIOGRAPHIE

Aceituno, J., G. Casiez, et N. Roussel

2013. How low can you go? : Human limits in small unidirectional mouse movements. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, Pp. 1383–1386, New York, NY, USA. ACM.

Adhikarla, V. K., P. Woźniak, A. Barsi, D. Singhal, P. T. Kovács, et T. Balogh

2014. Freehand interaction with large-scale 3d map data. In *2014 3DTV-Conference : The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON)*, Pp. 1–4.

Ajaj, R., F. Vernier, et C. Jacquemin

2009. Navigation modes for combined table/screen 3d scene rendering. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, ITS '09, Pp. 141–148, New York, NY, USA. ACM.

Arsenault, R. et C. Ware

2000. Eye-hand co-ordination with force feedback. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '00, Pp. 408–414, New York, NY, USA. ACM.

Arthur, K. W., K. S. Booth, et C. Ware

1993. Evaluating 3d task performance for fish tank virtual worlds. *ACM Transactions on Information Systems*, 11(3) :239–265.

Badler, N. I., K. H. Manoochehri, et D. Baraff

1987. Multi-dimensional input techniques and articulated figure positioning by multiple constraints. In *Proceedings of the 1986 Workshop on Interactive 3D Graphics*, I3D '86, Pp. 151–169, New York, NY, USA. ACM.

Bae, S.-H., R. Balakrishnan, et K. Singh

2008. Ilovesketch : As-natural-as-possible sketching system for creating 3d curve models. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology*, UIST '08, Pp. 151–160, New York, NY, USA. ACM.

Bae, S.-H., T. Kobayash, R. Kijima, et W.-S. Kim

2004. Tangible nurbs-curve manipulation techniques using graspable handles on a large display. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*, UIST '04, Pp. 81–90, New York, NY, USA. ACM.

- Balakrishnan, R., T. Baudel, G. Kurtenbach, et G. Fitzmaurice  
1997. The rockin'mouse : Integral 3d manipulation on a plane. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, CHI '97, Pp. 311–318, New York, NY, USA. ACM.
- Balakrishnan, R., G. Fitzmaurice, G. Kurtenbach, et W. Buxton  
1999a. Digital tape drawing. In *Proceedings of the 12th Annual ACM Symposium on User Interface Software and Technology*, UIST '99, Pp. 161–169, New York, NY, USA. ACM.
- Balakrishnan, R., G. Fitzmaurice, G. Kurtenbach, et K. Singh  
1999b. Exploring interactive curve and surface manipulation using a bend and twist sensitive input strip. In *Proceedings of the 1999 Symposium on Interactive 3D Graphics*, I3D '99, Pp. 111–118, New York, NY, USA. ACM.
- Balakrishnan, R. et G. Kurtenbach  
1999. Exploring bimanual camera control and object manipulation in 3d graphics interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '99, Pp. 56–62, New York, NY, USA. ACM.
- Baudel, T.  
1994. A mark-based interaction paradigm for free-hand drawing. In *Proceedings of the 7th Annual ACM Symposium on User Interface Software and Technology*, UIST '94, Pp. 185–192, New York, NY, USA. ACM.
- Beaudouin-Lafon, M., S. Huot, H. Olafsdottir, et P. Dragicevic  
2014. Glidecursor : Pointing with an inertial cursor. In *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces*, AVI '14, Pp. 49–56, New York, NY, USA. ACM.
- Bérard, E., G. Wang, et J. R. Cooperstock  
2011. On the limits of the human motor control precision : The search for a device's human resolution. In *INTERACT'11*, Pp. 107–122. Springer-Verlag.
- Bordegoni, M., F. Ferrise, M. Covarrubias, et M. Antolini  
2011. Geodesic spline interface for haptic curve rendering. *IEEE Transactions on Haptics*, 4(2) :111–121.
- Boritz, J. et K. S. Booth  
1997. A study of interactive 3d point location in a computer simulated virtual environment. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, VRST '97, Pp. 181–187, New York, NY, USA. ACM.
- Boubekeur, T.  
2014. Shellcam : Interactive geometry-aware virtual camera control. In *2014 IEEE International Conference on Image Processing (ICIP)*, Pp. 4003–4007.
- Brooks, F. P.  
1992. Walkthrough - a dynamic graphics system for simulating virtual buildings. Research Report TR92-026, Department of Computer Science, University of North Carolina.
- Brooks, Jr., F. P.  
1987. Walkthrough - a dynamic graphics system for simulating virtual buildings. In *Proceedings of the 1986 Workshop on Interactive 3D Graphics*, I3D '86, Pp. 9–21, New York, NY, USA. ACM.

BSi

2003. Visual acuity test types - part 1 : Test charts for clinical determination of distance visual acuity - specification. BS 4274-1 :2003, British Standards Institution.

Casiez, G. et N. Roussel

2011. No more bricolage! : Methods and tools to characterize, replicate and compare pointing transfer functions. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, Pp. 603–614, New York, NY, USA. ACM.

Casiez, G. et D. Vogel

2008. The effect of spring stiffness and control gain with an elastic rate control pointing device. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, Pp. 1709–1718, New York, NY, USA. ACM.

Casiez, G., D. Vogel, Q. Pan, et C. Chaillou

2007. Rubberedge : Reducing clutching by combining position and rate control with elastic feedback. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology*, UIST '07, Pp. 129–138, New York, NY, USA. ACM.

Chen, M., S. J. Mountford, et A. Sellen

1988. A study in interactive 3-d rotation using 2-d control devices. In *Proceedings of the 15th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '88, Pp. 121–129, New York, NY, USA. ACM.

Christie, M., P. Olivier, et J.-M. Normand

2008. Camera control in computer graphics. *Computer Graphics Forum*, 27(8) :2197–2218.

Chung, J. C.

1992. A comparison of head-tracked and non-head-tracked steering modes in the targeting of radiotherapy treatment beams. In *Proceedings of the 1992 Symposium on Interactive 3D Graphics*, I3D '92, Pp. 193–196, New York, NY, USA. ACM.

Clay, A., J.-C. Lombardo, J. Conan, et N. Couture

2013. Towards bi-manual 3d painting : Generating virtual shapes with hands. In *Proceedings of the 1st Symposium on Spatial User Interaction*, SUI '13, Pp. 79–79, New York, NY, USA. ACM.

Cohen, J. M., J. F. Hughes, et R. C. Zeleznik

2000. Harold : A world made of drawings. In *Proceedings of the 1st International Symposium on Non-photorealistic Animation and Rendering*, NPAR '00, Pp. 83–90, New York, NY, USA. ACM.

Cohen, J. M., L. Markosian, R. C. Zeleznik, J. F. Hughes, et R. Barzel

1999. An interface for sketching 3d curves. In *Proceedings of the 1999 Symposium on Interactive 3D Graphics*, I3D '99, Pp. 17–21, New York, NY, USA. ACM.

Darken, R. P., W. R. Cockayne, et D. Carmein

1997. The omni-directional treadmill : A locomotion device for virtual worlds. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology*, UIST '97, Pp. 213–221, New York, NY, USA. ACM.

De Amicis, R., F. Bruno, A. Stork, et M. L. Luchi

2002. The eraser pen : a new interaction paradigm for curve sketching in 3D. In *Proceedings of DESIGN 2002, the 7th International Design Conference*, DESIGN 2002, Pp. 465–470.

- De Araùjo, B. R., G. Casiez, et J. A. Jorge  
2012. Mockup builder : Direct 3d modeling on and above the surface in a continuous interaction space. In *Proceedings of Graphics Interface 2012, GI '12*, Pp. 173–180, Toronto, Ont., Canada, Canada. Canadian Information Processing Society.
- Decle, F., M. Hachet, et P. Guitton  
2009. Tech-note : Scruticam : Camera manipulation technique for 3d objects inspection. In *Proceedings of the 2009 IEEE Symposium on 3D User Interfaces, 3DUI '09*, Pp. 19–22, Washington, DC, USA. IEEE Computer Society.
- Dominjon, L., S. Richir, A. Lecuyer, et J. M. Burkhardt  
2006. Haptic Hybrid Rotations : Overcoming Hardware Angular Limitations of Force-Feedback Devices. In *IEEE VR*, Pp. 167–174.
- Fairchild, K. M., B. H. Lee, J. Loo, H. Ng, et L. Serra  
1993. The heaven and earth virtual reality : Designing applications for novice users. In *Proceedings of IEEE Virtual Reality Annual International Symposium*, Pp. 47–53.
- Fei, G., W.-S. Lee, C. Joslin, et Z. Xin  
2008. 3d animation creation using space canvases for free-hand drawing. In *Proceedings of The 7th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry, VRCAI '08*, Pp. 12 :1–12 :6, New York, NY, USA. ACM.
- Fleury, C., A. Chauffaut, T. Duval, V. Gouranton, et B. Arnaldi  
2010. A Generic Model for Embedding Users' Physical Workspaces into Multi-Scale Collaborative Virtual Environments. In *ICAT 2010 (20th International Conference on Artificial Reality and Telexistence)*, Adelaide, Australia.
- Frees, S., G. D. Kessler, et E. Kay  
2007. Prism interaction for enhancing control in immersive virtual environments. volume 14, New York, NY, USA. ACM.
- Frees, S. et D. Lancellotti  
2011. *Workspace-Driven, Blended Orbital Viewing in Immersive Environments*, Pp. 185–193. Berlin, Heidelberg : Springer Berlin Heidelberg.
- Fuhrmann, A., D. Schmalstieg, et M. Gervautz  
1998. *Strolling Through Cyberspace With Your Hands In Your Pockets : Head Directed Navigation In Virtual Environments*, Pp. 216–225. Vienna : Springer Vienna.
- Gain, J., P. Marais, et W. StraÙer  
2009. Terrain sketching. In *Proceedings of the 2009 Symposium on Interactive 3D Graphics and Games, I3D '09*, Pp. 31–38, New York, NY, USA. ACM.
- Garcia, J., T. Carpentier, et J. Bresson  
2017. Interactive-compositional authoring of sound spatialization. *Journal of New Music Research*, 46(1) :74–86.
- Gibbs, C. B.  
1962. Controller design : Interactions of controlling limbs, time-lags and gains in positional and velocity systems. *Ergonomics*, 5(2) :385–402.

- Grossman, T., R. Balakrishnan, G. Kurtenbach, G. Fitzmaurice, A. Khan, et B. Buxton  
2001. Interaction techniques for 3d modeling on large displays. In *Proceedings of the 2001 Symposium on Interactive 3D Graphics, I3D '01*, Pp. 17–23, New York, NY, USA. ACM.
- Grossman, T., R. Balakrishnan, G. Kurtenbach, G. Fitzmaurice, A. Khan, et B. Buxton  
2002. Creating principal 3d curves with digital tape drawing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '02*, Pp. 121–128, New York, NY, USA. ACM.
- Grossman, T., R. Balakrishnan, et K. Singh  
2003. An interface for creating and manipulating curves using a high degree-of-freedom curve input device. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '03*, Pp. 185–192, New York, NY, USA. ACM.
- Guy, E., P. Punpongsanon, D. Iwai, K. Sato, et T. Boubekour  
2015. Lazynav : 3d ground navigation with non-critical body parts. In *3D User Interfaces (3DUI), 2015 IEEE Symposium on*, Pp. 43–50.
- Hachet, M., B. Bossavit, A. Cohé, et J.-B. de la Rivière  
2011. Toucheo : Multitouch and stereo combined in a seamless workspace. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, UIST '11*, Pp. 587–592, New York, NY, USA. ACM.
- Hachet, M., F. Declé, S. Knodel, et P. Guitton  
2008. Navidget for easy 3d camera positioning from 2d inputs. In *Proceedings of the 2008 IEEE Symposium on 3D User Interfaces, 3DUI '08*, Pp. 83–89, Washington, DC, USA. IEEE Computer Society.
- Hachet, M., P. Guitton, et P. Reuter  
2003. The cat for efficient 2d and 3d interaction as an alternative to mouse adaptations. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology, VRST '03*, Pp. 225–112, New York, NY, USA. ACM.
- Hagedorn, B. et J. Döllner  
2008. Sketch-based navigation in 3d virtual environments. In *Proceedings of the 9th International Symposium on Smart Graphics, SG '08*, Pp. 239–246, Berlin, Heidelberg. Springer-Verlag.
- Hand, C.  
1997. A survey of 3d interaction techniques. volume 16, Pp. 269–281. Blackwell Publishers.
- Harrison, C. et A. K. Dey  
2008. Lean and zoom : Proximity-aware user interface and content magnification. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '08*, Pp. 507–510, New York, NY, USA. ACM.
- Harrison, C. et S. E. Hudson  
2008. Pseudo-3d video conferencing with a generic webcam. In *Multimedia, 2008. ISM 2008. Tenth IEEE International Symposium on*, Pp. 236–241.
- Hincapié-Ramos, J. D., X. Guo, P. Moghadasian, et P. Irani  
2014. Consumed endurance : A metric to quantify arm fatigue of mid-air interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '14*, Pp. 1063–1072, New York, NY, USA. ACM.

- Hinckley, K., M. Sinclair, E. Hanson, R. Szeliski, et M. Conway  
1999. The videomouse : A camera-based multi-degree-of-freedom input device. In *Proceedings of the 12th Annual ACM Symposium on User Interface Software and Technology*, UIST '99, Pp. 103–112, New York, NY, USA. ACM.
- Hinckley, K., J. Tullio, R. Pausch, D. Proffitt, et N. Kassell  
1997. Usability analysis of 3d rotation techniques. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology*, UIST '97, Pp. 1–10, New York, NY, USA. ACM.
- Igarashi, T., R. Kadobayashi, K. Mase, et H. Tanaka  
1998. Path drawing for 3d walkthrough. In *Proceedings of the 11th Annual ACM Symposium on User Interface Software and Technology*, UIST '98, Pp. 173–174, New York, NY, USA. ACM.
- Igarashi, T., S. Matsuoka, et H. Tanaka  
1999. Teddy : A sketching interface for 3d freeform design. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '99, Pp. 409–416, New York, NY, USA. ACM Press/Addison-Wesley Publishing Co.
- Ishii, H. et B. Ullmer  
1997. Tangible bits : Towards seamless interfaces between people, bits and atoms. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, CHI '97, Pp. 234–241, New York, NY, USA. ACM.
- ISO  
2010. Ergonomics of human-system interaction – part 210 : Human-centred design for interactive systems. ISO 9241-210 :2010, International Organization for Standardization.
- Jacob, T., G. Bailly, E. Lecolinet, R. Foulon, et E. Corteel  
2014. A design space for three-dimensional curve edition. In *Proceedings of the 26th Conference on L'Interaction Homme-Machine*, IHM '14, Pp. 105–112, New York, NY, USA. ACM.
- Jacquemin, G., T. Coduys, et M. Ranc  
2012. Iannix 0.8. In *Actes des Journées d'Informatique Musicale*.
- Jay, C. et R. Hubbard  
2003. Amplifying head movements with head-mounted displays. *Presence*, 12(3) :268–276.
- Keefe, D., R. Zeleznik, et D. Laidlaw  
2007. Drawing on air : Input techniques for controlled 3d line illustration. *IEEE Transactions on Visualization and Computer Graphics*, 13(5) :1067–1081.
- Keefe, D. F., D. A. Feliz, T. Moscovich, D. H. Laidlaw, et J. J. LaViola, Jr.  
2001. Cavepainting : A fully immersive 3d artistic medium and interactive experience. In *Proceedings of the 2001 Symposium on Interactive 3D Graphics*, I3D '01, Pp. 85–93, New York, NY, USA. ACM.
- Khan, A., B. Komalo, J. Stam, G. Fitzmaurice, et G. Kurtenbach  
2005. Hovercam : Interactive 3d navigation for proximal object inspection. In *Proceedings of the 2005 Symposium on Interactive 3D Graphics and Games*, I3D '05, Pp. 73–80, New York, NY, USA. ACM.
- Khan, A., I. Mordatch, G. Fitzmaurice, J. Matejka, et G. Kurtenbach  
2008. Viewcube : A 3d orientation indicator and controller. In *Proceedings of the 2008 Symposium on Interactive 3D Graphics and Games*, I3D '08, Pp. 17–25, New York, NY, USA. ACM.

- Koller, D. R., M. R. Mine, et S. E. Hudson  
1996. Head-tracked orbital viewing : an interaction technique for immersive virtual environments. In *ACM UIST '96*, Pp. 81–82.
- König, W. A., J. Gerken, S. Dierdorf, et H. Reiterer  
2009. Adaptive pointing — design and evaluation of a precision enhancing technique for absolute pointing devices. In *Proceedings of the 12th IFIP TC 13 International Conference on Human-Computer Interaction : Part I*, INTERACT '09, Pp. 658–671, Berlin, Heidelberg. Springer-Verlag.
- Krueger, W. et B. Froehlich  
1994. The responsive workbench. *IEEE Computer Graphics and Applications*, 14(3) :12–15.
- Kruger, R., S. Carpendale, S. D. Scott, et A. Tang  
2005. Fluid integration of rotation and translation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '05, Pp. 601–610, New York, NY, USA. ACM.
- Kulshreshth, A. et J. J. LaViola, Jr.  
2013. Evaluating performance benefits of head tracking in modern video games. In *Proceedings of the 1st Symposium on Spatial User Interaction*, SUI '13, Pp. 53–60, New York, NY, USA. ACM.
- Lai, C.-Y. et N. Zakaria  
2010. Pressure-based 3d curve drawing. In *Proceedings of the 10th International Conference on Smart Graphics*, SG'10, Pp. 156–159, Berlin, Heidelberg. Springer-Verlag.
- Lai, C.-Y. et N. Zakaria  
2014. Towards an intelligent framework for pressure-based 3d curve drawing. In *Smart Graphics*, M. Christie et T.-Y. Li, eds., volume 8698 of *Lecture Notes in Computer Science*, Pp. 63–72. Springer International Publishing.
- Lapides, P., E. Sharlin, M. C. Sousa, et L. Streit  
2006. The 3d tractus : A three-dimensional drawing board. In *Proceedings of the First IEEE International Workshop on Horizontal Interactive Human-Computer Systems*, TABLETOP '06, Pp. 169–176, Washington, DC, USA. IEEE Computer Society.
- Laviola, J. J. et M. Katzourin  
2007. An exploration of non-isomorphic 3d rotation in surround screen virtual environments. In *2007 IEEE Symposium on 3D User Interfaces*.
- LaViola, Jr., J. J., D. A. Feliz, D. F. Keefe, et R. C. Zeleznik  
2001. Hands-free multi-scale navigation in virtual environments. In *Proceedings of the 2001 Symposium on Interactive 3D Graphics*, I3D '01, Pp. 9–15, New York, NY, USA. ACM.
- Lee, J., R. Post, et H. Ishii  
2011. Zeron : Mid-air tangible interaction enabled by computer controlled magnetic levitation. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, Pp. 327–336, New York, NY, USA. ACM.
- Lee, J., S. Teerapittayanon, et H. Ishii  
2010. Beyond : Collapsible input device for direct 3d manipulation beyond the screen. In *Adjunct Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology*, UIST '10, Pp. 393–394, New York, NY, USA. ACM.

- Lin, M. L., R. G. Radwin, et G. C. Vanderheiden  
1992. Gain effects on performance using a head-controlled computer input device. *Ergonomics*, 35(2) :159–175.
- MacKenzie, I. S., R. W. Soukoreff, et C. Pal  
1997. A two-ball mouse affords three degrees of freedom. In *CHI '97 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '97, Pp. 303–304, New York, NY, USA. ACM.
- Mackinlay, J. D., S. K. Card, et G. G. Robertson  
1990. Rapid controlled movement through a virtual 3d workspace. In *Proceedings of the 17th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '90, Pp. 171–176, New York, NY, USA. ACM.
- Marchal, D. C. P. M., C. Moerman, G. Casiez, et N. Roussel  
2013. Designing Intuitive Multi-touch 3D Navigation Techniques. In *IFIP Interact 2013*, Pp. xx–xx, Cap Town, South Africa.
- Massimino, M. J., T. B. Sheridan, et J. B. Roseborough  
1989. One handed tracking in six degrees of freedom. In *Conference Proceedings., IEEE International Conference on Systems, Man and Cybernetics*, Pp. 498–503 vol.2.
- Miles, W.  
1930. Ocular dominance in human adults. *Journal of General Psychology*, 3 :412–429.
- Mulder, J. D. et R. V. Liere  
2000. Enhancing fish tank vr. In *Proceedings of the IEEE Virtual Reality 2000 Conference*, VR '00, Pp. 91–, Washington, DC, USA. IEEE Computer Society.
- Nabiyouni, M., B. Laha, et D. A. Bowman  
2014. Poster : Designing effective travel techniques with bare-hand interaction. In *3D User Interfaces (3DUI), 2014 IEEE Symposium on*, Pp. 139–140.
- Nigay, L. et J. Coutaz  
1996. Espaces conceptuels pour l'interaction multimédia et multimodale. *Technique et science informatiques*, 15(9).
- Ortega, M., W. Stuerzlinger, et D. Scheurich  
2015. Shocam : A 3d orbiting algorithm. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, UIST '15, Pp. 119–128, New York, NY, USA. ACM.
- Ortega, M. et T. Vincent  
2014. Direct drawing on 3d shapes with automated camera control. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, Pp. 2047–2050, New York, NY, USA. ACM.
- Parmerud, Å.  
2010. Hyperspace.
- Pausch, R., T. Burnette, D. Brockway, et M. E. Weiblen  
1995. Navigation and locomotion in virtual worlds via flight into hand-held miniatures. In *Proceedings of the 22Nd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '95, Pp. 399–400, New York, NY, USA. ACM.

- Perelman, G., M. Serrano, M. Raynal, C. Picard, M. Derras, et E. Dubois  
2015. The roly-poly mouse : Designing a rolling input device unifying 2d and 3d interaction. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI '15*, Pp. 327–336, New York, NY, USA. ACM.
- Pottier, L.  
1998. Dynamical spatialization of sound. holophon : a graphic and algorithmic editor for sigma1. In *Proc. of the International Conference on Digital Audio Effects, DAFx-98*.
- Poupyrev, I., M. Billinghamurst, S. Weghorst, et T. Ichikawa  
1996. The go-go interaction technique : non-linear mapping for direct manipulation in VR. In *ACM UIST '96*, Pp. 79–80.
- Poupyrev, I., S. Weghorst, et S. Fels  
2000. Non-isomorphic 3d rotational techniques. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '00*, Pp. 540–547, New York, NY, USA. ACM.
- Poupyrev, I., S. Weghorst, T. Otsuka, et T. Ichikawa  
1999. Amplifying spatial rotations in 3d interfaces. In *CHI '99 Extended Abstracts on Human Factors in Computing Systems, CHI EA '99*, Pp. 256–257, New York, NY, USA. ACM.
- Reilink, R., G. de Bruin, M. Franken, M. A. Mariani, S. Misra, et S. Stramigioli  
2010. Endoscopic camera control by head movements for thoracic surgery. In *BioRob*, Pp. 510–515. IEEE.
- Roupé, M., P. Bosch-Sijtsema, et M. Johansson  
2014. Interactive navigation interface for virtual reality using the human body. *Computers, Environment and Urban Systems*, 43 :42 – 50.
- Sachs, E., A. Roberts, et D. Stoops  
1991. 3-draw : a tool for designing 3d shapes. *IEEE Computer Graphics and Applications*, 11(6) :18–26.
- Santosa, S., F. Chevalier, R. Balakrishnan, et K. Singh  
2013. Direct space-time trajectory control for visual media editing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '13*, Pp. 1149–1158, New York, NY, USA. ACM.
- Schkolne, S., M. Pruett, et P. Schröder  
2001. Surface drawing : Creating organic 3d shapes with the hand and tangible tools. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '01*, Pp. 261–268, New York, NY, USA. ACM.
- Schumacher, M. et J. Bresson  
2010. Compositional control of periphonic sound spatialization. In *2nd International Symposium on Ambisonics and Spherical Acoustics*, Paris, France.
- Shoemake, K.  
1992. Arcball : A user interface for specifying three-dimensional orientation using a mouse. In *Proceedings of the Conference on Graphics Interface '92*, Pp. 151–156, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

- Simeone, A. L., E. Velloso, J. Alexander, et H. Gellersen  
2014. Feet movement in desktop 3d interaction. In *2014 IEEE Symposium on 3D User Interfaces (3DUI)*, Pp. 71–74.
- Sko, T. et H. J. Gardner  
2009. *Head Tracking in First-Person Games : Interaction Using a Web-Camera*, Pp. 342–355. Berlin, Heidelberg : Springer Berlin Heidelberg.
- Slater, M., A. Steed, et M. Usoh  
1995. The virtual treadmill : A naturalistic metaphor for navigation in immersive virtual environments. In *Selected Papers of the Eurographics Workshops on Virtual Environments '95, VE '95*, Pp. 135–148, London, UK, UK. Springer-Verlag.
- Song, D. et M. Norman  
1993. Nonlinear interactive motion control techniques for virtual space navigation. In *VRAIS'93*, Pp. 111–117. IEEE Comput. Soc.
- Stoakley, R., M. J. Conway, et R. Pausch  
1995. Virtual reality on a wim : Interactive worlds in miniature. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '95*, Pp. 265–272, New York, NY, USA. ACM Press/Addison-Wesley Publishing Co.
- Tan, D. S., G. G. Robertson, et M. Czerwinski  
2001. Exploring 3d navigation : Combining speed-coupled flying with orbiting. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '01*, Pp. 418–425, New York, NY, USA. ACM.
- Teather, R. J. et W. Stuerzlinger  
2007. Guidelines for 3d positioning techniques. In *Proceedings of the 2007 Conference on Future Play, Future Play '07*, Pp. 61–68, New York, NY, USA. ACM.
- Teather, R. J. et W. Stuerzlinger  
2008. Exaggerated head motions for game viewpoint control. In *Future Play '08*, Pp. 240–243. ACM.
- Terziman, L., M. Marchal, M. Emily, F. Multon, B. Arnaldi, et A. Lécuyer  
2010. Shake-Your-Head : Revisiting Walking-In-Place for Desktop Virtual Reality. In *ACM Symposium on Virtual Reality Software and Technology*, Pp. 27–34, Hong Kong, Hong Kong SAR China. ACM.
- Thibodeau, G. et K. Patton  
1996. *Anatomy and Physiology Third Edition*. Mosby.
- Thiel, Y., K. Singh, et R. Balakrishnan  
2011. Elasticurves : Exploiting stroke dynamics and inertia for the real-time neatening of sketched 2d curves. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, UIST '11*, Pp. 383–392, New York, NY, USA. ACM.
- Tsandilas, T., A. Bezerianos, et T. Jacob  
2015. Sketchsliders : Sketching widgets for visual exploration on wall displays. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI '15*, Pp. 3255–3264, New York, NY, USA. ACM.

- Tsandilas, T. et P. Dragicevic  
2016. Accounting for Chance Agreement in Gesture Elicitation Studies. Research Report 1584, LRI - CNRS, University Paris-Sud.
- Usoh, M., K. Arthur, M. C. Whitton, R. Bastos, A. Steed, M. Slater, F. P. Brooks, et Jr  
1999. Walking > walking-in-place > flying, in virtual environments. In *ACM SIGGRAPH '99*.
- van Schooten, B. W., E. M. A. G. van Dijk, E. Zudilova-Seinstra, A. Suinesiaputra, et J. H. C. Reiber  
2010. The effect of stereoscopy and motion cues on 3d interpretation task performance. In *Proceedings of the International Conference on Advanced Visual Interfaces, AVI '10*, Pp. 167–170, New York, NY, USA. ACM.
- Wang, R., S. Paris, et J. Popović  
2011. 6d hands : Markerless hand-tracking for computer aided design. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, UIST '11*, Pp. 549–558, New York, NY, USA. ACM.
- Ware, C.  
2012. *Information Visualization, Third Edition : Perception for Design*, number 978-0123814647 in 3, morgan kaufmann edition. Elsevier.
- Ware, C., K. Arthur, et K. S. Booth  
1993. Fish tank virtual reality. In *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems, CHI '93*, Pp. 37–42, New York, NY, USA. ACM.
- Ware, C. et G. Franck  
1996. Evaluating stereo and motion cues for visualizing information nets in three dimensions. *ACM Transactions on Graphics*, 15(2) :121–140.
- Ware, C. et D. R. Jessome  
1988. Using the bat : a six-dimensional mouse for object placement. *IEEE Computer Graphics and Applications*, 8(6) :65–70.
- Ware, C. et S. Osborne  
1990. Exploration and virtual camera control in virtual three dimensional environments. In *Proceedings of the 1990 Symposium on Interactive 3D Graphics, I3D '90*, Pp. 175–183, New York, NY, USA. ACM.
- Wattenberg, M.  
2001. Sketching a graph to query a time-series database. In *CHI '01 Extended Abstracts on Human Factors in Computing Systems, CHI EA '01*, Pp. 381–382, New York, NY, USA. ACM.
- Wilches, D., P. Figueroa, A. Conde, et E. Samavati  
2012. Sketch-based interface for animation for non-experts. In *Informatica (CLEI), 2012 XXXVIII Conferencia Latinoamericana En*, Pp. 1–8.
- Wobbrock, J. O., M. R. Morris, et A. D. Wilson  
2009. User-defined gestures for surface computing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '09*, Pp. 1083–1092, New York, NY, USA. ACM.

Yamaguchi, K., T. Komuro, et M. Ishikawa

2009. Ptz control with head tracking for video chat. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '09, Pp. 3919–3924, New York, NY, USA. ACM.

Yu, L., K. Efstathiou, P. Isenberg, et T. Isenberg

2012. Efficient structure-aware selection techniques for 3d point cloud visualizations with 2dof input. *Visualization and Computer Graphics, IEEE Transactions on*, 18(12) :2245–2254.

Zeleznik, R. et A. Forsberg

1999. Unicam - 2d gestural camera controls for 3d environments. In *Proceedings of the 1999 Symposium on Interactive 3D Graphics*, I3D '99, Pp. 169–173, New York, NY, USA. ACM.

Zeleznik, R. C., K. P. Herndon, et J. F. Hughes

1996. Sketch : An interface for sketching 3d scenes. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, Pp. 163–170, New York, NY, USA. ACM.

Zhai, S.

1995. *Human performance in six degree of freedom input control*. PhD thesis, University of Toronto.

---

## PUBLICATIONS

Jacob, T., G. Bailly, et E. Lecolinet

2015. A study on 3d viewpoint control through head and shoulders motion. In *Proceedings of the 27th Conference on L'Interaction Homme-Machine (Travaux en cours)*, IHM '15, Pp. 25 :1–25 :6, New York, NY, USA. ACM.

Jacob, T., G. Bailly, E. Lecolinet, G. Casiez, et M. Teyssier

2016. Desktop orbital camera motions using rotational head movements. In *Proceedings of the 2016 Symposium on Spatial User Interaction*, SUI '16, Pp. 139–148, New York, NY, USA. ACM.

Jacob, T., G. Bailly, E. Lecolinet, R. Foulon, et E. Corteel

2014. A design space for three-dimensional curve edition. In *Proceedings of the 26th Conference on L'Interaction Homme-Machine*, IHM '14, Pp. 105–112, New York, NY, USA. ACM.



# Orbital control for 3D trajectory drawing through head movements

Thibaut JACOB

**ABSTRACT :** The field of 3D sound is experiencing rapid growth due to a combination of factors (standardization of new audio formats, equipment for cinemas, etc.). While most of the work in this field has focused on 3D sound processing, creating 3D auditory content interactively remains a challenging task because it requires drawing and editing three dimensional trajectories to control the movement of audio sources in space. In this thesis in Human-Computer Interaction (HCI), we consider the creation of audio source trajectories as a particular case of 3D modelling and propose the following contributions. On a conceptual level, we first present a design space of 3D trajectory creation techniques. We also propose a classification of existing camera controls in relation to the type of control and modalities. On an empirical level, we conducted five user studies in order to design a new interaction technique for orbital viewpoint control. This technique allows users to perform 360 wide rotations by leveraging head roll rotations. Finally, we propose an implementation of our interaction technique and present its integration with two applications : *Blender*, a well-known 3D modelling software, and *Performer*, which is used by *Radio France* to place sound sources in 3D space during live events.

**KEY-WORDS :** 3D sound, trajectory edition, body movements, 3D interaction, viewpoint control, transfer function, elicitation study



# Contrôle orbital pour le tracé de trajectoires 3D à l'aide des mouvements de la tête

Thibaut JACOB

**RESUME :** Le domaine du son 3D est aujourd'hui en pleine effervescence du fait d'une combinaison de facteurs (normalisation de nouveaux formats audio, équipement des salles de cinéma, etc.). Les travaux effectués dans ce domaine se sont principalement concentrés sur le codage et le traitement du son 3D. Cependant, la création interactive de contenus sonores 3D n'est pas un problème trivial, car cette tâche nécessite de tracer ou d'éditer des trajectoires en trois dimensions pour animer des sources sonores dans l'espace. Dans cette thèse en Interaction Homme-Machine (IHM), nous considérons l'édition de trajectoires de sources sonores en 3D comme un cas particulier de tâche de modélisation 3D et proposons plusieurs contributions. Sur le plan conceptuel, nous proposons un espace de conception pour le tracé de trajectoires en trois dimensions. Nous présentons également une classification des contrôles de caméra existants en fonction du type de contrôle et des modalités utilisées. Sur le plan empirique, nous conduisons 5 études utilisateurs dans le but de créer une technique d'interaction pour le contrôle du point de vue orbital permettant d'effectuer une rotation de  $360^\circ$  en utilisant le roulis de la tête. Enfin, nous présentons une implémentation de notre technique ainsi que son intégration dans le logiciel de modélisation 3D *Blender* et dans le logiciel *Performer* utilisé par *Radio France* pour contrôler la position de sources sonores dans le cadre de performances en son 3D.

**MOTS-CLEFS :** Son 3D, tracé de trajectoires, mouvements du corps, interaction 3D, contrôle du point de vue, fonction de transfert, étude par élicitation

