Restitution vidéo stéréoscopique maîtrisée: application à la Réalité Virtuelle

Fabien Goslin

N°: 2010 ENAM 0001

Ecole doctorale n° 432 : Sciences des Métiers de l'Ingénieur

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

l'École Nationale Supérieure d'Arts et Métiers

Spécialité " Informatique "

présentée et soutenue publiquement par

Fabien GOSLIN

le 11 janvier 2010

Restitution vidéo stéréoscopique maîtrisée

Application à la Réalité Virtuelle

Directeur de thèse : **Simon Richir**
Co-encadrement de la thèse : **Florian Schramm**
Etablissement d'accueil : **CLARTE**

Jury

M. Bruno ARNALDI, Professeur, INSA	Président
M. Laurent LUCAS, Professeur, CReSTIC	Rapporteur
M. Youssef MEZOUAR, Maître de conférences, LASMEA	Rapporteur
M. Claude ANDRIOT, Expert Senior, CEA	Examinateur
M. Florian SCHRAMM, Ingénieur Chercheur, CEA	Examinateur
M. Jean-Louis DAUTIN, Directeur, CLARTE	Invité

Arts et Métiers ParisTech - Centre d'Angers
Laboratoire LAMPA, équipe Presence & Innovation

THESE

Remerciements

Cette thèse a été réalisée au sein de l'équipe Presence & Innovation des Arts et Metiers ParisTech sous la direction de Simon Richir. Je souhaite le remercier de m'avoir accueilli dans cette équipe et de m'avoir permis de travailler sur ce sujet passionnant.

Je remercie Claude Andriot de m'avoir proposé de collaborer avec le CEA sur ce sujet, ainsi que Florian Schramm pour l'encadrement de ce travail. Tout au long de cette thèse, ils ne m'ont ménagé ni leurs remarques ni leurs encouragements.

Merci également à Jean-Louis Dautin de m'avoir donné l'opportunité de travailler au sein de CLARTE. Cet environnement scientifique et technique de qualité m'a fourni d'excellentes conditions de travail.

Je tiens à remercier Laurent Lucas et Youcef Mezouar d'avoir accepté d'être les rapporteurs de cette thèse, ainsi que Bruno Arnaldi d'avoir accepté d'en être l'examinateur.

Au cours de ces trois ans passés à Laval, j'ai pu faire des rencontres très enrichissantes. Je pense en particulier à Alexandre Bouchet, Guillaume Brincin, Damien Cariou, Céline Chatelain, Lionel Dominjon, Dimitri Faure, ainsi qu'aux autres membres de l'équipe de CLARTE. Merci pour les nombreuses discussions techniques que nous avons eues ainsi que les bons moments que nous avons passés en diverses occasions.

Merci également à François Druel, Franck Hernoux, Abdelmajid Kadri, Evelyne Klinger, Emilie Loup-Escande, Akihiko Shirai, ainsi qu'aux autres membres de l'équipe Presence & Innovation qu'il serait trop long de citer ici, pour leurs conseils et l'ambiance chaleureuse des réunions d'équipe.

Merci à Jérome Ardouin et Marc Le Renard de l'ESIEA-Ouest de m'avoir, entre autres choses, permis de faire mes premiers pas d'enseignant. J'espère que les étudiants auront perçu une partie du plaisir et de l'intérêt que j'ai éprouvé.

Pour terminer, je souhaite remercier les personnes qui ont contribué par leur soutien et leurs encouragements à la réussite de ce travail, en particulier mes parents et mon frère qui ont toujours été disponibles, Céline qui a vécu cette thèse à mes côtés, ainsi que toute sa famille.

Résumé

La capture en relief d'une scène réelle peut être réalisée grâce à un couple de caméras vidéo (banc stéréoscopique). La capture de ces images vidéo stéréoscopiques et leur restitution sur des systèmes de projection en relief sont à l'interface entre les domaines de la réalité virtuelle, de la vision par ordinateur, et du cinéma en relief.

Placé au sein de cette très vaste thématique, ce travail concerne la projection en relief, sur des systèmes de Réalité Virtuelle, d'images issues d'une capture par un banc stéréoscopique fixe. De très nombreuses contraintes (limitations des configurations de capture et des conditions de restitution notamment) ont restreint l'utilisation de cette technologie. Dans ce mémoire de thèse, nous détaillons les améliorations que nous avons apportées à certaines étapes de la chaîne de transmission stéréoscopique, afin de maîtriser la restitution de vidéos stéréoscopiques. Pour atteindre cet objectif, nous avons réalisé une modélisation mathématique détaillée des caméras, et des différentes configurations de capture et de restitution que nous utilisons.

Disposer d'images stéréoscopiques les moins déformées possible était un point de départ indispensable à la suite de notre travail. Dans ce but, nous avons développé un algorithme de rectification d'images vidéo stéréoscopiques. Afin d'assurer une rectification temps réel, nous avons implémenté cet algorithme sur processeur de carte graphique (GPU ou Graphics Processing Unit), en mettant en place une technique à base de table de référence.

La distance interoculaire de l'utilisateur est un paramètre important pour assurer une bonne restitution des images sur les systèmes de Réalité Virtuelle. Pourtant par commodité, la valeur moyenne de cet écart est souvent prise comme référence, alors que d'importantes différences existent d'un utilisateur à l'autre. Afin d'améliorer la restitution en fixant plus précisément ce paramètre critique, nous avons développé une méthode de calibration de la distance interoculaire de l'utilisateur.

Enfin, alors que les spectateurs des salles de cinéma en relief sont assis dans une zone bien définie devant l'écran, le déplacement des utilisateurs devant le système de projection d'images stéréoscopiques est une caractéristique des systèmes de Réalité Virtuelle. Pour palier aux problèmes que l'on rencontre lors de la projection d'images issues d'un banc stéréoscopique fixe pour un utilisateur en mouvement, nous proposons une méthode pour maîtriser la restitution de la profondeur perçue par cet utilisateur, en nous basant sur une segmentation en profondeur de la scène.

Mots Clés : réalité virtuelle, vidéo stéréoscopique, rectification gpu, calcul distance interoculaire, restitution maîtrisée de la profondeur.

Abstract

Acquiring 3D scenes can be achieved by using a pair of video cameras (stereo rig). The acquisition process of these stereoscopic video images and their projection on 3D display systems is a wide-extending field which borrows from the technologies of virtual reality, computer vision and 3D filming.

Within this wide field, we focused our research work on the projection in 3D of images captured by a fixed stereo rig, on virtual reality systems. The use of this technology was limited by numerous constraints (pertaining essentially to capture configurations and display conditions). In this PhD thesis, we detail the improvements we brought to some of the steps of the stereo transmission process, in order to control more precisely the display of stereoscopic videos and to alleviate some of the limitations mentioned above. We based our work on detailed mathematical modeling of the cameras, and of the capture and the display configurations.

First, it was necessary to work on stereoscopic images as little distorted as possible. To obtain such images, we developed an algorithm that corrects distortions on these images. To ensure real-time rectification, we implemented this algorithm on GPU (Graphics Processing Unit), through the use of a technique based on reference tables.

The interocular distance is a fundamental step to make a full use of virtual reality system. In spite of the fact that significant disparities exist from one user to another, an average value is often used for this parameter. In order to set correctly this critical parameter, we developed a method to calibrate the interocular distance of each user, so that the display of stereoscopic images on these systems will provide a more accurate perception.

Finally, while viewers in 3D theaters sit in a defined area in front of the projection screen, allowing users to move freely is one of the main characteristics of virtual reality systems. To overcome the difficulties that occur when projecting images captured by a fixed stereo rig to a user in motion, we propose a method to improve the restitution of the depth perceived by the user, using a depth segmentation of the captured scene.

Keywords : virtual reality, stereoscopic video, gpu rectification , interocular distance, depth controlled rendering.

Table des matières

Table des figures

Chapitre 1

Introduction

Sommaire

1.1 Contexte de ce travail

Les besoins en matière de simulation numérique ne cessent actuellement d'augmenter dans de nombreux domaines (médecine, nucléaire, design, formation, marketing, ...). Pour certaines applications, telles que la maintenance d'installations, ou la collaboration à distance, il est nécessaire d'enrichir la simulation avec des données du monde réel. Ce domaine particulier apparaît spécialement porteur de développements.

Parmi la variété de techniques d'acquisition de données du monde réel, la capture vidéo stéréoscopique permet de capturer des images numériques d'une scène réelle, grâce à un couple de caméras vidéo (ou banc stéréoscopique). Les techniques employées se rapprochent de celles utilisées dans le monde du cinéma en relief, et empruntent également au domaine de la vision par ordinateur.

Parmi ces techniques, celles de la capture de la profondeur de scènes réelles à partir de caméras vidéo, et de l'affichage des images enregistrées sur un écran de projection ont été plus particulièrement étudiées et développées pour le cinéma 3D, au milieu du siècle dernier. Mais l'attrait pour ce nouveau type de divertissement a été fortement ralenti en raison de nombreux inconforts qui incommodaient le spectateur. Ces inconforts étaient majoritairement liés aux techniques empiriques de paramétrage des caméras lors de l'enregistrement, de même qu'aux moyens de projection rudimentaires de l'époque.

L'avènement du numérique grand public a récemment relancé l'intérêt pour la vidéo en relief. Le cinéma tout d'abord a trouvé dans cette technologie une manière de palier à quelques unes des contraintes qui lui étaient imposées jusqu'alors. Grâce notamment aux possibilités de re-traitement des images en post-production, certains effets visuels indésirables peuvent être corrigés. De plus, les caméras numériques sont moins encombrantes que les caméras argentiques, ce qui facilite la capture des images en permettant de positionner les caméras dans des positions plus proches de l'optimal, par exemple en minimisant la distance inter-caméras. De plus, elles ont une fréquence d'enregistrement plus élevée, et une résolution d'images plus grande. Couplées à la projection numérique, qui permet également une fréquence d'affichage des images plus élevée, le confort de visualisation des images a nettement augmenté.

Mais le cinéma n'est pas le seul domaine à avoir bénéficié de ces avancées techniques. La projection numérique, couplée au développement d'application graphiques pour ordinateur, a permis à de nouvelles disciplines, telles que la Réalité Virtuelle par exemple, de voir le jour. Cette discipline repose sur la simulation (numérique) interactive et immersive d'environnements réels ou virtuels.

Ce mémoire présente les résultats de nos travaux, visant à améliorer l'acquisition, la projection et la perception, en relief et en temps réel, d'images issues d'un banc stéréoscopique fixe sur des systèmes de Réalité Virtuelle.

Nos objectifs principaux ont consisté à mieux comprendre, puis à alléger certaines des nombreuses contraintes qui pesaient sur les configurations de capture et sur les conditions de restitution, et qui représentaient des freins à l'utilisation de cette technologie.

1.2 Organisation du mémoire

Nous avons axé nos travaux sur la projection, en relief et en temps réel, d'images issues d'un banc stéréoscopique fixe sur des systèmes de Réalité Virtuelle. Notre but lors de cette thèse à été d'améliorer plusieurs étapes du processus de transmission stéréoscopique, afin de faciliter l'utilisation de cette technologie, pour enrichir la simulation de données issues du monde réel.

Après un bref historique dans la section suivante, le chapitre 2 présente un état de l'art sur les technologies d'acquisition, de transmission et de restitution de données 3D.

Nous présentons dans le chapitre 3, les principes de la vision en relief chez l'homme, ainsi que les bases de la modélisation mathématique de caméras réelles et virtuelles, en configuration monoscopique (une caméra) et stéréoscopique (banc de deux caméras).

Puis, nous détaillons précisément, dans le chapitre 4, la modélisation complète de chacune des étapes de la chaîne de transmission stéréoscopique (capture/affichage des images et perception de la profondeur). Cette modélisation va servir de base mathématique aux améliorations que nous proposons.

La perception d'un point 3D par un utilisateur peut être simulée à l'aide de deux images, chacune affectée à un oeil (une image pour l'oeil gauche et une autre pour l'oeil droit). La perception 3D du point dépend alors essentiellement de l'écart entre la position ce point dans l'image gauche et sa position dans l'image droite. Lorsque l'on travaille avec des images vidéo, celles-ci se montrent naturellement déformées. Ces déformations sont dues au processus d'acquisition (optique, capteur, CCD). Nous abordons dans le chapitre 5, le développement d'un algorithme de rectification d'images vidéo stéréoscopiques, pour d'éliminer ces déformations non-souhaitées.

Afin d'assurer une rectification temps réel, nous avons implémenté cet algorithme sur processeur de carte graphique (GPU), en mettant en place une technique à base de table de référence.

Ensuite, la détermination de la valeur de l'écart interoculaire des utilisateurs est une étape importante pour l'utilisation de systèmes de Réalité Virtuelle. Par commodité, la valeur moyenne de cet écart est souvent prise comme référence, mais d'importantes différences existent d'un utilisateur à l'autre. Afin de pouvoir améliorer la restitution d'images stéréoscopiques sur ces systèmes, nous proposons une méthode de calibration de la distance interoculaire de l'utilisateur.

Enfin un utilisateur d'un système d'affichage stéréoscopique de type système de Réalité Virtuelle est amené à se déplacer, contrairement à un spectateur d'une salle de cinéma en relief, assis dans son fauteuil. Pour palier aux problèmes que l'on rencontre lors de la projection d'images issues d'un banc stéréoscopique fixe, nous proposons dans le chapitre 6, de modifier la parallaxe entre les images projetées, afin de les adapter au déplacement de l'utilisateur. La perception 3D en est alors améliorée. Cette adaptation nécessite une segmentation des images vidéo en couche de profondeur.

1.3 Rappels historiques : du stéréoscope à la télévision en relief

La réflexion autour de la vision binoculaire en relief a inspiré depuis l'antiquité de nombreux artistes/intellectuels qui cherchaient à représenter le monde en relief.

Les premières représentations picturales comportant une vision d'une même scène pour chaque oeil, semblent avoir été réalisées par Jacopo Chimenti da Empoli, peintre de l'école florentine (1554-1640).

Depuis lors, les recherches se sont donc poursuivies dans ce domaine, jusque l'année 1838, première date de la création d'un instrument qui valide les théories exprimées jusqu'alors sur la perception en 3D.

Charles Wheatstone était un physicien et inventeur britannique. Il décrivit les détails de la perception du relief grâce à la vision stéréoscopique dans [Wheatstone 38]. Il réalisa par la suite les premiers essais grâce au dispositif qu'il construisit : le Stéréoscope (cf. Figure 1.1). L'utilisateur regardait alors deux images (une pour chaque oeil) qui se formaient sur deux miroirs plans suite à la réflection de dessins, placés sur des plaques de bois verticales. Ces deux images lui permettaient ainsi de percevoir le relief de la scène représentée sur les dessins. Mais la technologie du miroir plan adopté sur le Stéréoscope obligeait l'instrument à occuper beaucoup de place et l'handicapait pour sa diffusion. Le succès auprès du grand public vint quelques années plus tard, grâce à la réalisation d'un autre système par David Brewster (cf. Figure 1.2) qui remplaça les miroirs par des lentilles.La taille de ce système était beaucoup plus modeste (10 cm de largeur et 13 cm de hauteur) et permettait d'être pris en main.

La popularité de cette technologie va alors grandir parallèlement à l'invention de la photographie en 1853 et à sa généralisation (le Stéréoscope de Holmes est le plus connu encore aujourd'hui -cf. Figure 1.3-). En 1858, eut lieu la première projection d'images fixes en relief, en utilisant la technique nouvelle de l'anaglyphe. Mais l'engouement pour cette technologie va se voir menacé par l'impression de photos dans les journaux et magazines à la fin du XIX$^{\text{ième}}$ siècle.

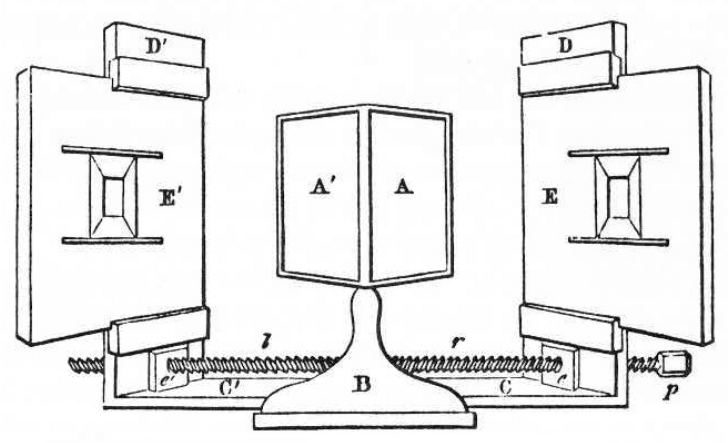
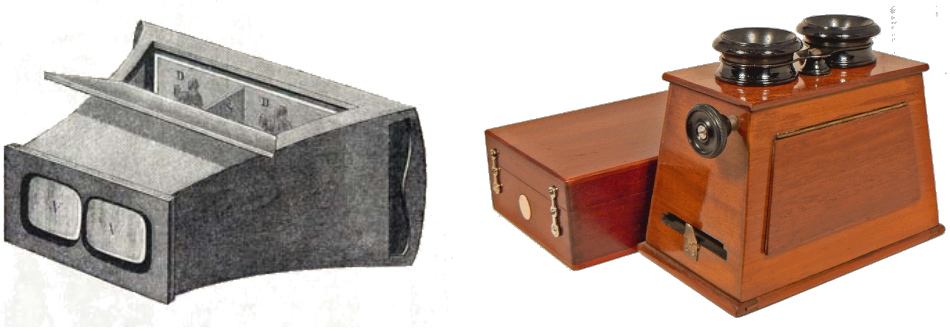


Figure 1.1 – Stéréoscope de Wheatstone



(a) Vision d'artiste du 1er modèle construit (b) Version récente

Figure 1.2 – Stéréoscope de Brewster

Le premier rebond de cette technologie eut brièvement lieu dans les années 1950. L'année 1949 a vu la création du premier appareil photo stéréo construit en grande série : le "Realist" de Seton Rochwite (cf. Figure 1.5). Ensuite, lorsque la télévision fit baisser de moitié la fréquentation des cinémas aux Etats-Unis, les studios américains d'Hollywood cherchèrent à la concurrencer et produisirent durant les années 1952-53 soixante-cinq films stéréoscopiques [Lipton 82], dont les plus célèbres sont "Bwana Devil" (Arch Oboler cf. Figure 1.4), "House of Wax" (Andre de Toth), "Dial M for Murder" (Alfred Hitchcock filmé en 3D mais projeté en 2D). Cette période faste ne dura que peu de temps, en effet les problèmes techniques de capture/projection étaient encore importants : projection simultanée des deux images sur un même écran, synchronisation des images entre elles sans décalage dans le temps, nécessité de projeter sur un écran métallisé,.... Ces problèmes généraient souvent de nombreux inconforts (nausées et maux de tête en particulier) pour le spectateur.

Après une longue période où cette technologie a été abandonnée, on assiste depuis le milieu des années 1980 à un renouveau. Jusque très récemment, les films 3D étaient projetés dans des cinémas de parcs d'attraction où la fréquentation est plus assurée que dans les cinémas normaux. Mais depuis quelques années, grâce notamment à l'arrivée sur le marché de nouveaux projecteurs numériques, une nouvelle "vague 3D" a lieu. En

Figure 1.3 – Stéréoscope de Holmes

Figure 1.4 – Affiche du film "Bwana Devil" considéré comme le premier film couleur 3D

Figure 1.5 – Le "Realist" de Seton Rochwite

Amérique du Nord, le nombre de salles équipées en projecteurs pour diffuser de la 3D est passé de 84 en 2005 à 4112 actuellement (cf Tableau 1.1).

Année / Zone	2005	'06	'07	'08	'09	'10*	'11*	'12*	'13*
Europe Ouest	-	10	175	381	1519	2101	2602	3073	3405
Europe Centrale/Est	-	12	205	556	1774	2460	3044	3610	4002
Amérique du nord	84	206	994	1514	4112	5808	6934	7507	8468
Monde	84	258	1299	2540	6882	9732	11905	13378	15082

Tableau 1.1 – Nombres de salles de cinéma numérique 3D par zone géographique (les * indiquent les chiffres attendus pour ces années et les - des données non disponibles. source : Screendigest.com)

Les trois principales plateformes de cinéma pour projeter en 3D étant les formats RealD [1], XpanD [2] et Dolby 3D Digital [3].

Selon des chiffres de Juin 2009, la technologie Real-D est présente dans 3400 cinémas à travers le monde (elle est largement dominante en Amérique du Nord, où elle occupe plus de 7/8^{e} des cinémas, contre 1/4 en Europe), et son carnet de commande lui assurera un total de 9000 cinémas, d'ici quelques années. Dolby 3D Digital dispose d'un nombre d'installations inférieur : 1000 cinémas équipés (1/10^{e} en Amérique du Nord et 1/3 en Europe) et 500 supplémentaires sont actuellement en cours d'installation. Enfin, XpanD occupe 1000 cinémas à l'heure actuelle majoritairement en Europe (avec 2/5^{e} des cinémas européens et un pourcentage très faible ailleurs dans le monde).

Prévus pour sortir au cours de l'année 2009, pas loin d'une quinzaine de films en reliefs ont été réalisés (Coraline, Monsters vs. Aliens, Up, Avatar,...). La majeure partie

1. http://www.reald.com/Content/proProducts.aspx?pageID=28
2. http://www.xpandcinema.com/products/
3. http://www.dolby.com/professional/motion_picture/solutions_d3ddc.html

de ces films sont des films d'animation, où l'adaptation pour la 3D est plus aisée que pour les films traditionnels grâce à l'utilisation de caméras purement virtuelles. Mais de gros budgets (par exemple, 315M$ pour Avatar) sont actuellement dépensés sur de longs métrages traditionnels, en particulier pour le développement de nouvelles caméras de cinéma stéréoscopiques (Voir le dernier exemple en date : le système Pace Fusion 3D utilisé pour le nouveau film de James Cameron *Avatar*, Figure 1.6).

Figure 1.6 – Caméra stéréoscopique de nouvelle génération pour le cinéma : le système Pace Fusion 3D lors d'un tournage sur un Grand Prix de moto aux USA en Juillet 2009.

Les industriels de matériel télévisuel se placent également dans cette mouvance. De nombreuses entreprises travaillent sur le sujet parmi lesquelles : Philips (pionner dans le domaine), LG, Panasonic, Samsung, Sony,... Même si des essais avaient eu lieu dès les débuts de la télévision moderne, la télévision stéréoscopique connaît un engouement particulier depuis les années 80. Les solutions actuelles utilisent les mêmes techniques de projection couplée à des lunettes pour percevoir le relief. Certains fabricants cherchent à réaliser des téléviseurs permettant de regarder la télévision sans lunettes. Cependant même si le relief est bien restitué et ressenti par les spectateurs, ce développement se heurte pour le moment aux problèmes que posent la visualisation du contenu 3D de différents endroits de la pièce (au contraire d'un cinéma où le placement du public est totalement maîtrisé, le public devant une télévision est très mobile) et la possibilité d'un rendu pour plusieurs spectateurs simultanément.

Pour résumer, l'intérêt actuel pour la 3D se manifeste actuellement par le développement de techniques pour le cinéma et la télévision. Les techniques de capture et de projection ayant évoluées depuis les années 50, nous sommes peut-être à l'aube d'une ère du "tout 3D".

Chapitre 2

Etat de l'art technologique

Sommaire

La restitution du relief d'une scène réelle repose sur la capture des profondeurs de cette scène par le biais de technologies d'acquisition de données 3D. L'affichage en relief de ces données peut être effectué par le biais de systèmes de projection/visualisation qui permettent à un ou plusieurs utilisateurs de percevoir le relief.

Nous allons présenter dans ce chapitre, les différentes technologies existantes liées à l'acquisition et à l'affichage en relief de données 3D.

2.1 Acquisition de données 3D

L'acquisition 3D permet de récupérer, à partir d'un objet physique, des fichiers 3D sous forme de nuages de points ou d'ensemble de facettes. Cette technologie est très utilisée dans de nombreux domaines tels que : la médecine, la rétro-ingénierie, la robotique, la réalité virtuelle, l'archéologie,...

Les techniques d'acquisition de données 3D sont multiples et très variées. Elles peuvent être classées en trois grandes catégories (en suivant la classification de [Curless 00]) :
- acquisition par contact
- acquisition "transmissive"
- acquisition "reflective"

Chacune de ces trois catégories regroupe différents matériels et technologies.

L'acquisition par contact peut être réalisée de manière mécanique (bras à mesurer type cf. Figure 2.1), inertielle (giroscopes, accéléromètres), ou encore à l'aide de trackers à ultrasons ou magnétiques.

Figure 2.1 – Exemple de bras mécanique servant à l'acquisition par contact

L'acquisition "transmissive", quant-à elle, utilise les moyens de l'imagerie par résonance magnétique (IRM), de la tomographie axiale calculée aux rayons X (North Star Imaging[1]), ou des ultrasons. Enfin, l'acquisition "réflective" va se servir des informations que renvoient les objets. Elles peuvent être de nature visible (on parle alors d'acquisition optique ce qui est le cas des images par exemple) ou non (on parle alors d'acquisition non optique de type Radar ou Sonar).

Dans ce chapitre, nous allons plus particulièrement nous intéresser aux techniques les plus utilisées dans le domaine de l'acquisition réflective optique. Cette famille de techniques est elle-même composée de nombreuses méthodes qui peuvent êtres regroupées au sein de deux catégories : les méthodes actives et les méthodes passives.

1. `http://www.4nsi.com/`

2.1.1 Les méthodes actives d'acquisition

Les méthodes actives sont caractérisées par l'utilisation d'un capteur CCD combiné à une source de lumière qui va se réfléchir sur la surface de l'objet. L'image de cette lumière sur l'objet va être récupérée par le capteur. Ensuite, le but est de déterminer le relief de l'objet à partir de l'image obtenue. Les sources de lumière sont soit des lasers soit des lumières structurantes. Ces techniques sont complexes à mettre en oeuvre et requièrent un temps important, notamment pour la phase de numérisation des objets.

- **Télémétrie laser** [Wallace 06] [Jarvis 83] : cette technique est basée sur la mesure de distances par rayon laser. Le rayon laser est projeté sur l'objet. La distance à l'objet peut être mesurée grâce à l'emploi des techniques basées sur l'analyse du temps de vol, de la modulation de fréquence, ou de la comparaison de phase. Ces systèmes permettent une mesure directe de la distance (pas de calculs complexes à réaliser).
  Par exemple, dans le cas du scanner par temps de vol (Time of flight) qui est le plus courant, ce dernier envoie des pulsations de lumière (en général un laser), et mesure le temps que met le faisceau à revenir à la source émettrice de la lumière (cf. Figure 2.2). La distance est déduite grâce à la connaissance de la vitesse de la lumière.
  Les objets à scanner peuvent être très importants (le volume de travail maximum est de l'ordre de 150m) et la précision très bonne (de l'ordre de 5 mm).
  Cependant, il s'agit d'une méthode qui demande un temps de mise en oeuvre très important, en effet le scan s'effectue point par point. D'autres inconvénients de ces systèmes sont par exemple, leur complexité mécanique ou encore le problème de la sécurité des yeux des utilisateurs vis à vis de l'utilisation d'un laser puissant.
  Afin de dévier le faisceau laser, ces télémètres (Leica HDS-3000 [2] ou DeltaSphere-3000 [3]) sont composés de miroirs selon trois différentes configurations :
  - un miroir plan et un miroir tournant : système le plus fréquemment utilisé mais qui souffre d'un temps de latence non négligeable
  - deux miroirs tournants : le temps de latence est faible mais le système est plus coûteux à l'achat
  - deux miroirs plans : le temps de latence est important mais le prix est raisonnable.

Figure 2.2 – Exemple de scanner par temps de vol : le Leica HDS-3000

2. http://www.leica-geosystems.com/en/5574.htm
3. http://www.deltasphere.com/DeltaSphere-3000.htm

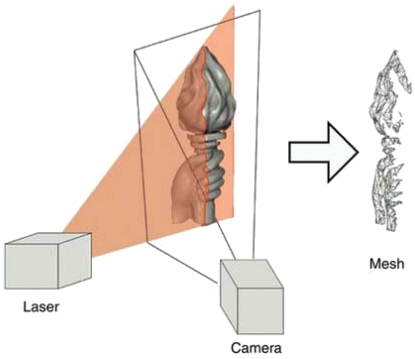


Figure 2.3 – Principe du scanner 3D à triangulation active

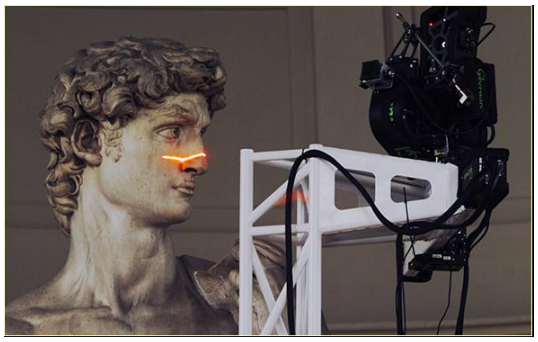


Figure 2.4 – Exemple de scanner 3D à triangulation active [Levoy 00]

– **Capteurs à triangulation active** : le principe consiste à éclairer l'objet avec un faisceau de lumière et observer l'éclairage avec une caméra.
La technique de triangulation repose sur la position en triangle de l'émetteur laser, du capteur photographique, et du point laser (cf. Figure 2.3) et sur la connaissance d'informations sur les distances/angles formés par ce triangle, ce qui permet de déterminer la position du point laser.
Deux types de projections de lumière existent : la projection d'un point lumineux et la projection d'un plan lumineux. Bien souvent, les scanners laser utilisent des bandes de points laser plutôt qu'un point seul pour améliorer la vitesse d'acquisition (cf. Figure 2.4 [Levoy 00]).

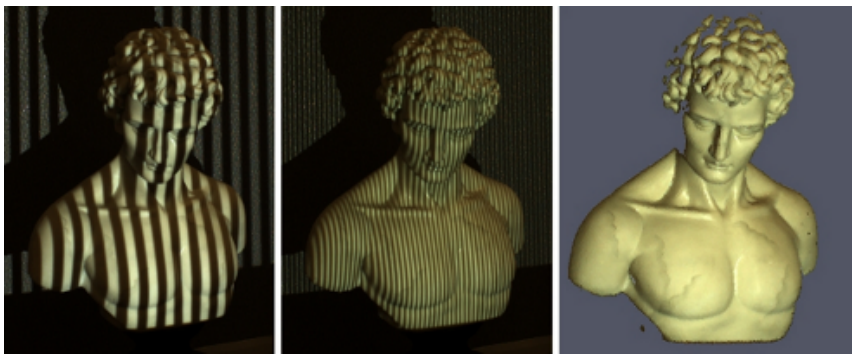


Figure 2.5 – Exemple de scanner à lumière structurée : les deux images à gauches représentent le modèle illuminé par deux motifs différents. A droite, on peut voir le résultat après numérisation.

Différents systèmes utilisent cette technologie :

- Scanner à lumière structurée : système composé d'une caméra et d'un projecteur de lumière. Le projecteur éclaire l'objet avec un motif lumineux et la déformation de ce motif sur l'objet est observée par la caméra [Weise 07] [Scharstein 03] [Rusinkiewicz 02] [Zhang 02] [Rocchini 02] [Bouguet 98]. Le point fort de cette technique est sa rapidité et sa précision (de l'ordre du millimètre). L'intégralité du chant de vision de la caméra est scannée au même instant.

Figure 2.6 – Exemple de scanner manuel : le scanner 3D handyscan

- Scanner à main [Ferreira 02] : ce type de scanner est un dérivé des scanners à triangulation (cf. Figure 2.6 le scanner 3D handyscan [4]). La différence réside dans le fait que l'utilisateur va déplacer le scanner autour de la zone à modéliser. Les positions et les orientations du scanner sont enregistrées soit en se basant sur les premiers points enregistrés soit en utilisant un système propre.

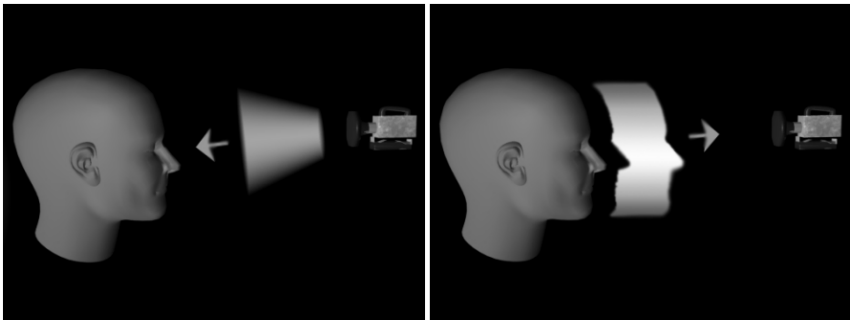


Figure 2.7 – Principe de fonctionnement de la Z-cam : à gauche l'étape d'envoi de la lumière infrarouge pulsée, et à droite la récupération des données de profondeur par la caméra.

– **Camera temps de vol** (scanner matriciel) : caméra qui permet l'enregistrement d'une carte de profondeur et d'une vidéo couleur de la scène filmée à l'aide d'une lumière infrarouge pulsée (cf. Figure 2.7). Exemples de produits commercialisés : Z-cam [5], SwissRanger SR4000 [6], PMDvision CamCube [7]

– **Technologies hybrides** :
  - Fusionner la profondeur acquise avec la technologie "Time-of-flight" à celle obtenue à l'aide de deux caméras stéréo pour obtenir une carte de profondeur plus précise [Zhu 08].
  - "Regular stereo with projected texture" : technique mixte utilisant la projection de texture et la stéréoscopie pour une meilleure précision.

4. http://www.creaform3d.com/en/handyscan3d/default.aspx
5. http://www.3dvsystems.com
6. http://www.mesa-imaging.ch/prodview4k.php
7. http://www.pmdtec.com

Les méthodes actives de scan ont été récemment très utilisées dans le but d'obtenir des nuages de points 3D très denses et précis. Un nuage de points représente un ensemble de sommets positionnés dans l'espace (cf. Figure 2.8). Les nuages de points ne sont pas utilisables en tant que tels par des applications 3D. Il est nécessaire de les transformer soit en grille de triangles, soit en courbes NURBS, soit en modèles CAO. Les méthodes nécessaire à cette conversion sont la triangulation de Delaunay, la technique de *marching triangles*, ou bien encore la technique de *marching cube*. Le processus d'obtention de la hauteur/profondeur, à partir des données enregistrées, est ensuite presque totalement automatique, mais la reconstruction complète d'objets 3D texturés est pour le moment encore peu fiable (même si on constate quelques technologies robustes pour des petits objets/surfaces : scans manuels), et le volume de données à traiter est très important.

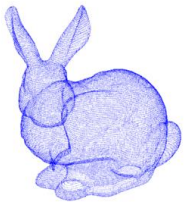


Figure 2.8 – Exemple d'un nuage de points représentant le lapin de Stanford

2.1.2 Les méthodes passives d'acquisition

Les méthodes passives quant-à elles exploitent la lumière ambiante, et n'utilisent que les images issues d'une ou plusieurs caméras. Parmi celles-ci, on trouve les techniques suivantes : "Shape from stereo", "Shape from stereo multi-view" (utilisation d'un ensemble de caméras stéréoscopiques [Allard 06] [Yang 04]),"Shape from multi-view" (utilisation de plusieurs images d'un même objet afin de reconstruire son volume [Pollefeys 99], 3DTVsolutions[8],) "Shape from motion" (repérer des points d'intérêt dans une séquence vidéo [Dellaert 00] [Zheng 00]), "Shape from shading" (reconstruire le relief d'une scène à partir d'une seule image en niveaux de gris [Woodham 80]), et "Shape from silhouette" (construction automatique d'un modèle 3D d'un objet à partir d'une suite d'images prises de cet objet selon différents points de vue [Yemez 07] [Nozick 06] [Wong 02]).

Parmi celles-ci, la technique de "Shape from stereo" [Zheng 03] est celle qui nous intéresse le plus. On utilise un couple de deux caméras afin d'observer un objet selon deux angles différents. Ce dispositif a un fonctionnement semblable à la vision humaine. Pour tous les points de l'image gauche (resp. droite) (cf. Figure 2.9), on cherche leur équivalent dans l'image droite (resp. gauche). La profondeur absolue de chaque point est déterminée par triangulation. Le couple de caméra est appelé banc stéréoscopique. Il est en général composé de deux caméras numériques industrielles. Les modèles commercialisés varient selon la complexité des réglages permis (orientation, longueur de la ligne de base, dispositif de pré-visualisation intégré,. . .). Exemple de produits : les systèmes de Binocle[9], ou ceux de Pace Fusion 3D[10].

8. http://www.3dtvsolutions.com/
9. http://www.binocle.com/
10. http://www.pacehd.com/

Figure 2.9 – Couple d'images stéréoscopiques issues de la techniques "Shape from stereo"

Une fois les formes 3D récupérées, il va être intéressant de les utiliser pour travailler. Dans la section suivante, nous allons voir quelles sont les technologies de projection de données 3D, pour que l'utilisateur les perçoive naturellement en relief, grâce aux propriétés de la vision humaine.

2.2 Projection en relief

Différents dispositifs technologiques permettent à un utilisateur de percevoir des données 3D, en relief. Nous avons choisi de les regrouper en deux catégories : ceux qui utilisent des lunettes, et ceux qui ne s'en servent pas.

2.2.1 Affichage avec lunettes

L'affichage d'images couplé à l'utilisation d'une paire de lunettes est actuellement la solution la plus couramment utilisée lorsqu'il s'agit de recréer une vision en relief par projection d'images. L'utilisation de lunettes permet à chaque oeil de l'utilisateur de percevoir seulement l'image qui lui est destinée.

Parmi les technologies employées, on trouve principalement : la projection d'anaglyphes, la projection polarisée, et la projection active/alternée.

2.2.1.a Projection d'anaglyphes

Un anaglyphe consiste en une superposition d'images stéréoscopiques de couleurs complémentaires. La projection anaglyphe est souvent associée aux lunettes rouge et cyan, que l'on trouve souvent accompagnant les magazines grand public. Le principe de création de ces images repose sur la séparation des composantes colorées de l'image : composante rouge de l'image gauche et composantes verte et bleue de l'image droite. Ce type de projection a été utilisé sur des images fixes dès le XIX^{e} siècle, et au cinéma dès le début du XX^{e} siècle.

Depuis une version améliorée, notamment pour obtenir une meilleur rendu des couleurs (certaines couleurs apparaissent mal en anaglyphe classique), a vu le jour. Cette technologie, nommée Infitec (pour *interference filter technology* du nom également de l'entreprise allemande qui la commercialise Infitec GmbH [11]), est basée sur un changement infime des longueurs d'onde des trois couleurs primaires pour chaque oeil. Ces changements étant très petits, ils sont indiscernables par les yeux humains sans lunettes. Les lunettes et

11. `http://www.infitec.net`

les projecteurs possèdent chacun deux filtres interférométriques avec des variations de longueurs d'onde différentes pour chaque oeil.

2.2.1.b Projection polarisée

La projection polarisée fonctionne également grâce à des filtres au niveau des projecteurs et sur les lunettes. Les filtres au niveau des projecteurs sont chargés de polariser les images droite et gauche dans des sens différents. On trouve ainsi des polarisations perpendiculaires (filtres croisés à 90°) et des polarisations circulaires (technologie plus récente qui offre l'avantage de permettre au spectateur d'incliner la tête en conservant la perception du relief).

Deux variantes existent : la première requiert deux projecteurs, chacun équipé d'un filtre. La seconde se base sur un projecteur numérique équipé d'un filtre actif, capable de polariser dynamiquement et rapidement l'onde lumineuse dans chacune des deux directions. Ce filtre doit également être synchronisé avec l'affichage, afin d'effectuer une bonne polarisation des images [Lipton 82].

Ensuite les filtres montés sur les lunettes (en polarisation inverse par rapport au(x) filtre(s) du projecteur), permettent la visualisation des images pour l'oeil correspondant. Prenons par exemple le cas suivant : le filtre du projecteur polarise l'image de l'oeil gauche. L'oeil gauche reçoit des informations tandis que l'oeil droit est "éteint" (le filtre des lunettes bloque la lumière sur l'oeil droit uniquement).

La contrainte de cette technologie est qu'elle nécessite la projection sur un écran métallisé (écrans spéciaux recouverts d'une peinture contenant des particules de métal). Pour conserver la réflection des images polarisées, il est nécessaire que la lumière garde le sens de polarisation, ce que les écrans traditionnels ne permettent pas. D'autre part, les filtres absorbent une quantité importante de la lumière projetée (jusqu'à 60% [Cowan 07]), ce qui nécessite l'utilisation de projecteurs très lumineux.

2.2.1.c Projection active/alternée

La projection active ou à occultations alternées repose sur la technologie des cristaux liquides et sur la synchronisation avec le système d'affichage [Lipton 97]. A la place des verres des lunettes, deux panneaux de cristaux liquides sont installés. Lorsque le projecteur va afficher l'image gauche, il va transmettre le signal aux lunettes qui vont obscurcir l'oeil droit et laisser l'oeil gauche libre de percevoir l'image. Pour l'oeil droit, l'inverse va se passer, l'oeil gauche va être obscurci et l'oeil droit percevra l'image qui lui est destinée.

Cependant, afin de ne pas gêner l'utilisateur qui pourrait percevoir des alternances entre l'oeil droit et l'oeil gauche, il va être nécessaire d'utiliser des projecteurs à haute fréquence d'affichage (>100Hz) ce qui permet d'afficher chaque image droite et gauche à une fréquence supérieure à 50Hz, le double du seuil du discernement de l'oeil humain (25Hz).

2.2.2 Affichage sans lunettes

Un système est dit auto-stéréoscopique s'il ne nécessite pas que l'utilisateur porte des lunettes spéciales pour percevoir le relief. Les principaux systèmes auto-stéréoscopiques

sont les suivants.

2.2.2.a Ecran auto stéréoscopique

Les écrans auto-stéréoscopiques ont fait une percée récente dans les domaines tels que la télévision ou l'informatique. La plupart des solutions à base d'écrans reposent principalement sur trois technologies : celle des réseaux lenticulaires, celle des barrières de parallaxe, et celle de l'illumination [Schreer 05] (cf. également le projet télé-relief[12]).

La première permet d'adresser à chaque oeil une image différente grâce à une succession de micro-lentilles cylindriques, dit réseau lenticulaire, alors qu'une image entrelacée est projetée sur l'écran. L'image projetée derrière les lentilles doit être composée de micro-images imbriquées représentant la scène filmée sous des points de vue différents. Avec chaque oeil, l'utilisateur perçoit des pixels différents de l'image. Cette technique permet de conserver la luminosité des images sans altérations de couleur.

La technologie des barrières de parallaxe repose sur une structure composée d'une alternance de zones transparentes et de zones opaques. Ces dernières jouent pour chaque point de vue un rôle d'obturateur qui dégrade la luminosité. La perte de luminosité est en général compensée par un surplus d'éclairage qui dégrade les couleurs.

Enfin, l'auto-stéréoscopie à illumination est une variante de la technologie des barrières de parallaxe, et consiste en deux barrières de parallaxe superposées.

2.2.2.b Système holographique

La projection holographique est un procédé qui permet de reproduire un champ de lumière tel qu'il a été enregistré sur la scène réelle. Ce processus a été inventé pour des images fixes, mais une tendance récente porte sur l'étude des holographies en imagerie numérique. De manière générale cette technique est encombrante et complexe à mettre en oeuvre. Elle repose sur la capture et la restitution des ondes lumineuses d'objets à reproduire.

2.2.2.c Affichages volumiques

Les affichages volumiques fonctionnent selon le principe de la représentation d'un objet directement en 3D, la où les autres systèmes recréent un volume en utilisant la stéréoscopie sur des supports 2D. Ils se basent sur le remplissage volumique d'un espace. De ce fait, des points images 3D sont projetés dans un espace fixé.

Les affichages volumiques existent principalement sous deux formes :

- Affichage volumique par balayage : ces systèmes projettent des images sur une surface se déplaçant rapidement. Ils génèrent en conséquence une image tridimensionnelle. Les systèmes les plus répandus se basent sur un miroir rotatif tournant à plus de 900tr/min. Un exemple récent de cette technologie [Jones 07] génère de cette manière une projection à 360° grâce à un projecteur placé verticalement au miroir (cf. Figure 2.10). Le spectateur peut alors tourner autour de la représentation volumique, comme il le ferait autour d'une pièce réelle.

12. `http://www.3dtvsolutions.com/`

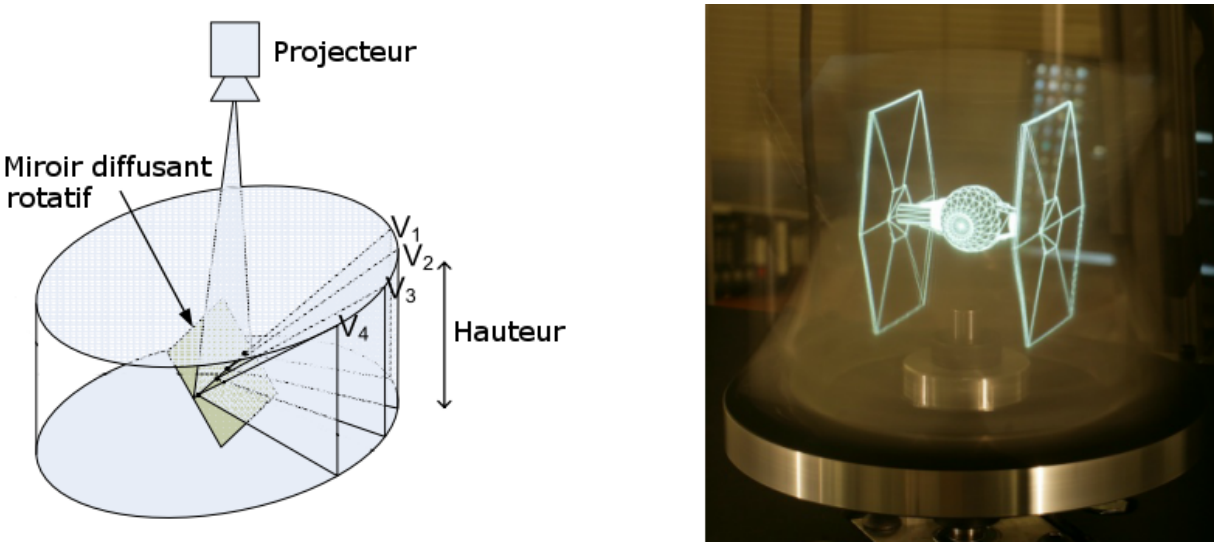


Figure 2.10 – Principe et exemple d'un affichage volumique couleur par balayage à 360 ° [Jones 07]

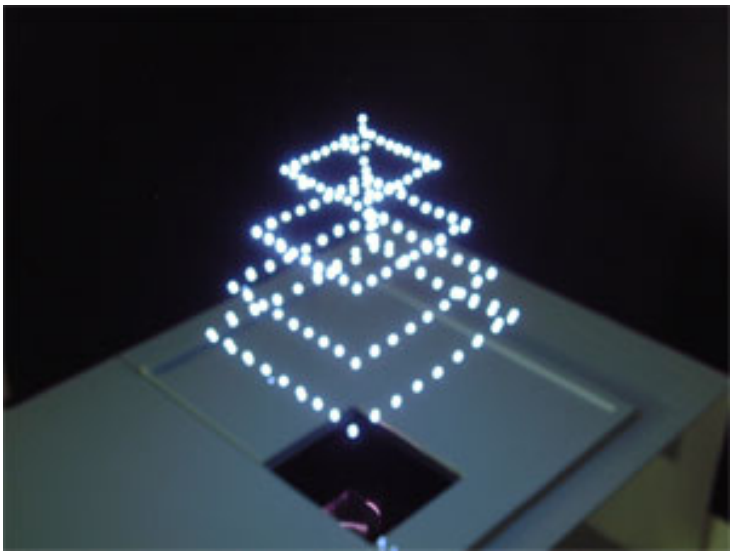


Figure 2.11 – Exemple d'un affichage volumique statique [Saito 08]

– Affichage volumique statique : ce type d'affichage utilise par exemple la lumière laser pour illuminer certaines parties d'un liquide, d'un solide, ou d'un gaz. La lumière laser peut également être utilisée pour diriger des boules de plasma sur des miroirs, pour dessiner des formes dans l'air [Saito 08] (cf. Figure 2.11).

2.2.2.d Projection simultanée : cas des casques de Réalité Virtuelle

Les casques immersifs de Réalité Virtuelle (en anglais HMD, pour Head Mounted Display) sont issus des travaux menés dans l'aviation militaire pour permettre à un pilote d'un avion de chasse de consulter de très nombreuses informations de vol, et de voir également le monde extérieur.

Il existe principalement deux sortes de casques : les casques de projection et les casques

semi-transparents/transparents [Fuchs 06] [Schreer 05]. Nous nous intéresserons ici, seulement aux casques du 1er type. Ils sont composés de deux écrans placés devant les yeux de l'utilisateur (cf. Figure 2.12), grâce à une structure fixée le plus souvent sur la tête de ce dernier. Ces deux écrans vont afficher selon le principe de la stéréoscopie une image différente pour chaque oeil. La position des écrans est ajustable horizontalement pour s'adapter aux différents écartements interoculaire des utilisateurs. Certains de ces casques sont non seulement des systèmes de projection, mais intègrent également des capteurs de mouvement pour transmettre les changements de position de la tête de l'utilisateur au système qui gère l'affichage des images. De plus amples informations peuvent être trouvées à l'adresse ci-dessous [13], notamment une comparaison multi-critères de nombreux casques.

Figure 2.12 – Exemple de casque de Réalité Virtuelle : l'eMagin Z800 de 3Dvisor. On distingue les deux écrans, un devant chaque oeil.

2.3 Utilisation de la projection en relief

2.3.1 Cinéma 3D

Les plateformes utilisées par le cinéma, pour projeter des films en relief, utilisent des technologies différentes :

- XpanD : ce système utilise un projecteur numérique et des lunettes actives à cristaux liquides. Il est adaptable sur les écrans de cinéma existants sans modification. Il est nécessaire cependant de disposer de boîtiers émetteurs d'infrarouges synchronisés sur l'affichage.
- Real D : cette technologie est basée sur la projection passive à polarisation circulaire sur un écran métallisé. L'innovation réside dans la capacité du système à réaliser une projection stéréoscopique à partir d'une seule source où les deux images sont combinées. Cela est permis grâce à un filtre, dont le rôle est de distinguer l'image droite de l'image gauche, en polarisant en circulaire droite pour l'oeil droit et en circulaire gauche pour l'oeil gauche. Le filtre est constitué de cristaux liquides rotatifs

13. http://vresources.org/HMD_rezanalysis.html

Figure 2.13 – XpanD : paire de lunettes stéréoscopiques actives. Au centre, au niveau du nez se trouve le récepteur infrarouge pour la synchronisation des lunettes avec la projection

qui polarisent chaque image à raison de 144 images par seconde. Le problème de cette technologie, outre la nécessité d'un écran de projection métallisé, est qu'elle peut générer des images fantômes (phénomène de *ghosting*), les filtres n'étant pas 100% étanches à une autre polarisation). Afin de résoudre ce problème Real D a mis en place deux parades. Ils ont commencé par créer des films "retouchés" pour limiter au maximum ces effets [14]. Récemment, ils ont développé une technologie temps réel de diminution de cet effet, en comparant les images gauche et droite, et en prédisant les problèmes d'appariement et en supprimant en conséquence les parties d'image les provoquant.

Figure 2.14 – Real D : filtre actif polarisant placé devant le projecteur

- Dolby 3D Digital Cinema : utilise la technologie de projection Infitec décrite précédemment.
- IMAX 3D : cette technologie utilise également le principe de la projection polarisée.

2.3.2 Télévision 3D

Le secteur de la télévision 3D est actuellement en plein essor. De plus en plus de modèles de téléviseur permettent de projeter du contenu en relief. La majorité des constructeurs utilisent les techniques d'affichage avec lunettes (Samsung, Toshiba, DepthQ, nVidia), même si quelque modèles auto stéréoscopiques existent (principalement fabriqués

14. http://www.manice.org/rubrique.php?id_rubrique=59

(a) Système de filtre tournant Dolby 3D Digital

(b) Lunettes utilisées pour la projection Dolby 3D Digital

Figure 2.15 – Eléments de la technologie de projection Dolby 3D Digital

pas Sharp et LG) sous différentes formes (par ex. [Matusik 04]). Des recherches sont également menées par le biais des consortiums tel que SMPTE (Society of Motion Picture and Television Engineers), CEA (Consumer Electronics Association), 3D@home,... pour développer un format standard pour la télévision numérique en relief à partir des techniques existantes pour encoder les flux vidéo stéréoscopiques (changements de couleurs, sous-échantillonnage de pixel, encodage avancé du flux vidéo).

Actuellement, la diffusion télévisuelle en 3D se démocratise. Au Japon par exemple, la chaîne NHK diffuse tous les jours une heure de programmes en relief. Aux Etats-Unis, les matchs de basket de la ligue nationale sont également retransmis en 3D. En mai 2008, Orange (marque de France Télécom, opérateur français de télécommunications) a proposé de suivre quelques matchs du tournoi de tennis de Roland Garros, en direct et en 3D relief, sur le site de Roland Garros et dans deux de ses agences (Champs Élysées et Paris Madeleine). La projection a été réalisée sur des téléviseurs 3D couplés à des lunettes polarisées.

Depuis une première mondiale s'est tenue le 2 juin 2009 avec la capture et la restitution dans plusieurs salles de cinéma (deux à Paris, une à Brest, et une à Avignon) d'un opéra de Mozart, filmé à l'Opéra de Rennes. Cela a mobilisé un dispositif important : quatre caméras bifocales ont été utilisées pour capter en direct. Ensuite, les images étaient envoyées à une régie qui se chargeait de la réalisation et l'encodage des signaux.

De plus amples détails concernant les techniques mises en place permettant la capture, la compression [Schreer 06] et la projection d'images pour la télévision peuvent être trouvés dans [Seuntiëns 06] [Schreer 05] [Meesters 04] [Redert 02].

2.3.3 Réalité Virtuelle

La Réalité Virtuelle utilise principalement la projection d'images virtuelles en relief. Ces images représentent des vues d'un monde virtuel, et sont générées pas le biais de

Figure 2.16 – Exemple de caméra stéréoscopique utilisée sur le tournoi de Roland Garros

caméras virtuelles. Dans ce cas, la transmission stéréoscopique comprend le calcul de la simulation 3D, la génération des images à projeter et l'envoi de ces images aux projecteurs.

Quelques travaux se sont intéressés aux problèmes de la restitution d'images stéréoscopiques pour la téléopération [Fuchs 06] [Maman 98] [Ernadotte 97] ou la collaboration entre sites distants [Jones 07].

D'autres se sont intéressés à l'interaction avec une surface du monde réel, grâce à la projection d'un flux vidéo stéréoscopique et à l'utilisation d'un bras mécanique à retour d'effort. A cet effet, [Scharstein 02] a utilisé le principe de la génération d'une carte de profondeur, qui lui a permis de repérer le placement des objets de la scène.

Néanmoins, nous n'avons pas trouvé mention de travaux qui portent sur l'enrichissement de simulations numériques virtuelles, à l'aide de vidéos stéréoscopiques. Cet aspect représente une des motivations derrière ce travail de thèse.

2.3.4 Jeux vidéo

Le domaine du jeu vidéo a évolué depuis sa création d'univers en 2D vers des univers en 3D. Jusqu'à très récemment, seuls quelques périphériques isolés ont permis la restitution de ces univers en relief. Historiquement, la première tentative fut menée par Nintendo qui développa en 1995 le "Virtual Boy", une console de jeu portative qui offrait un effet de relief en affichant avec des nuances de rouge sur fond noir. Mais le fonctionnement du système rendait très rapidement malades les joueurs.

En 1999, les lunettes Elsa Revelator 3D sont lancées. Fournies avec les cartes graphiques Elsa, elles fonctionnaient selon le principe de la projection active et nécessitaient des moniteurs ayant une fréquence de rafraîchissement minimum de 100 Hz (ce que supportaient la majorité des moniteurs CRT de l'époque). Un émetteur infrarouge était relié à la carte vidéo et permet la synchronisation des lunettes avec l'affichage. Malheureusement, l'arrivée massive des moniteurs LCD quelques années plus tard à condamné ce périphérique, en raison des fréquences d'affichage trop faibles (60Hz maximum à l'époque soit 30Hz par oeil) qui rendaient l'utilisation de ces lunettes douloureuse.

Sur ce même principe nVidia a lancé début 2009, des lunettes actives, en se basant sur l'arrivée de nouveaux moniteurs LCD à 120Hz.

Dans ce travail de thèse, nous nous sommes intéressés aux systèmes de Réalité Vir-

tuelle, qui reposent majoritairement sur des projections actives, afin d'interagir en immersion avec des simulations numériques.

En particulier, nous allons étudier l'affichage, sur ces systèmes, d'images issues d'une capture temps réel par un banc vidéo stéréoscopique.

Chapitre 3

Modélisation

Sommaire

Les systèmes d'affichage, dont nous venons de décrire le fonctionnement dans le chapitre précédent, se basent sur différents aspects de la vision humaine, afin de générer une perception tridimensionnelle chez l'utilisateur.

Les principes de la vision humaine vont être décrits dans la section 3.1. Nous y présentons également les principales sources d'inconfort visuel qui apparaissent lors de la visualisation d'images stéréoscopiques, et dont il faut se prémunir pour que le spectateur ne soit pas gêné.

Ces systèmes d'affichages reposent généralement sur la projection d'images. Dans notre cas, nous nous intéressons au domaine de la projection d'images vidéo stéréoscopiques. Dans le but de maîtriser cette projection, il est important de connaître l'influence de chaque composant de la chaîne de transmission stéréoscopique, qui va nous fournir ces images.

Nous allons détailler dans les sections 3.2 et 3.3, les modèles de caméra utilisés dans la suite de ce travail, pour modéliser la capture de ces images. Dans la section 3.4 nous présentons deux types de configurations de caméras stéréoscopiques, qui sont utilisées pour la capture d'images virtuelles et réelles.

3.1 Vision en relief chez l'homme

3.1.1 Description du système de vision humain

L'oeil humain Le capteur oeil a une forme globalement assimilable à une sphère. Il est à la base du système de vision humain en formant l'image des objets perçus, à partir de la perception des rayons lumineux issus de l'environnement. Il s'agit de l'organe le plus complexe du corps humain. Les informations qu'il reçoit sont transmises au cerveau, pour y être analysées (un tiers du cerveau sert à analyser la perception visuelle [Hecht 01]), par le nerf optique (cf. Figure 3.1 source Wikipedia [1]). Là où les yeux les plus simples ne sont capables que de distinguer les différences de lumière et d'obscurité, l'oeil humain distingue formes et couleurs. Il s'agit de l'un des types d'oeil les plus complexes du monde animal.

Il est principalement composé de la cornée (protection/filtre), du cristallin (lentille à focale variable), de l'iris (diaphragme), de la rétine (récepteur) et du nerf optique (lien avec le cerveau permettant le transfert de l'information).

3.1.2 Les indices visuels humains (depth cues)

La perception de la profondeur par le cerveau humain dépend des indices visuels, déterminés par les images fournies par ses deux yeux. Ces indices visuels peuvent être classés en deux grandes catégories : les indices visuels monoculaires, et les repère binoculaires. [Lipton 82] fournit une liste exhaustive des effets physio-physiques, que nous allons résumer dans les sections suivantes.

1. `http://fr.wikipedia.org/wiki/Oeil_humain`

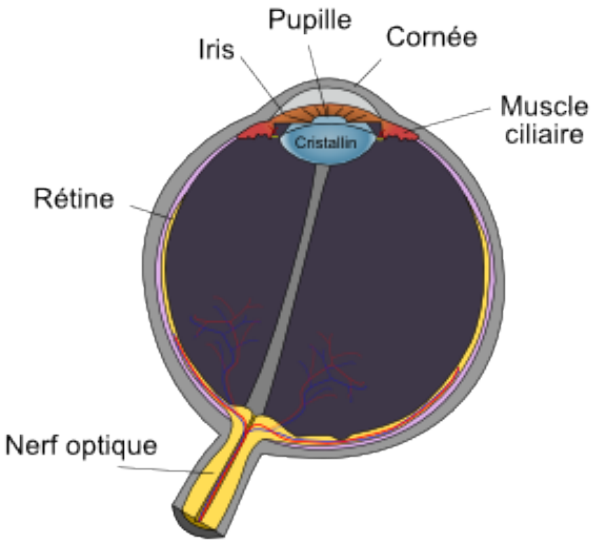


Figure 3.1 – Schéma d'un oeil humain

3.1.2.a Indices visuels monoculaires

- Taille de l'image rétinienne : la taille des images sur la rétine va nous informer sur la proximité d'un objet. En effet, entre deux objets de même taille disposés l'un devant l'autre, celui le plus près apparaît plus gros, que celui qui est plus éloigné (cf Figure 3.2).
- "Motion parallax" : dans le prolongement des notions de tailles différentes, la vitesse des objets les uns par rapport aux autres détermine leur proximité par rapport à l'observateur. Afin d'illustrer cette notion, prenons deux objets ayant la même vitesse, mais situés à des distances différentes de l'utilisateur. L'objet le plus près aura une vitesse apparente plus importante.
- Perspective : La réduction progressive de la taille de l'image d'un objet à mesure que la distance de l'observateur à cet objet augmente. Pour illustrer cette notion, les rails de chemin de fer sont un exemple classique. Ces rails sont rectilignes et parallèles, pourtant lorsque que l'on regarde à l'infini on perçoit une convergence en un point dit *de fuite* (cf. Figure 3.3).
- Occlusion, recouvrement : lorsqu'un objet est placé devant un autre, il va cacher une partie de ce dernier. Le cerveau humain interprète grâce à cette occlusion, le placement relatif des objets. Si un objet cache un second, c'est que le premier est devant le second.
- Lumière et ombre : ces détails sont très importants dans la perception humaine de la profondeur. Ils vont renseigner sur les sources de lumière qui éclairent la scène mais aussi sur la forme des objets et leurs placements les uns par rapport aux autres. Par exemple, un objet se trouvant entièrement dans l'ombre d'un autre est de taille inférieure, sinon des parties de cet objets seraient illuminées (cf. Figure 3.5).

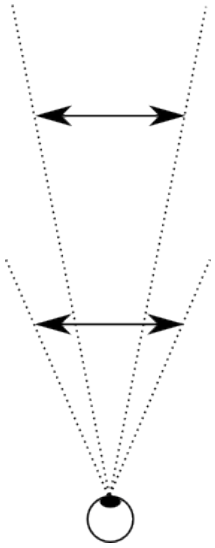


Figure 3.2 – Taille de l'image rétinienne d'un objet : ces deux segments fléchés sont de même taille, mais l'un est perçu plus près que l'autre.

Figure 3.3 – Exemple de perspective

- Altération de la visibilité : la visibilité d'une scène extérieure décroît en fonction de la profondeur (cf Figure 3.4). Cet effet est principalement dû aux propriétés de l'atmosphère. L'effet de flou lointain nous indique que les objets qui s'y trouvent sont éloignés de l'observateur, tandis que les objets nets se trouvent près.
- Gradient de texture : la finesse du détail, de ce qu'un observateur perçoit, décroît plus la distance à ce dernier augmente. Par exemple, si l'on observe une rue pavée, les carrés sont nets et distincts lorsqu'ils sont proches et forment une masse grisâtre, dont on ne distingue que peu de détails, lorsque l'on regarde au loin. En effet, la résolution de la fréquence spatiale de la vision humaine diminue avec la distance.

Ces indices pris indépendamment les uns des autres ne permettent pas de distinguer quel objet est à quelle profondeur. D'autre part, ils peuvent parfois échouer (conditions d'éclairage trompeuses, objets de taille inconnue,...). L'apprentissage au cours des premières années de la vie humaine, de la combinaison de plusieurs de ces indices en même temps, donne une idée de la position des objets les uns par rapport aux autres, mais est insuffisant pour obtenir une connaissance complète des placements en profondeur.

3.1.2.b Indice visuel binoculaire

La perception fine du relief par l'homme est permise grâce au principe de vision dit *stéréoscopique*. D'un point de vue physique, les yeux de l'homme sont écartés en moyenne d'environ $63mm$ ce qui provoque une différence de point de vue ([Dodgson 04] a réalisé une compilation de nombreuses études menées sur ce sujet, auprès d'un large éventail de personnes). Cette distance appelée *distance interoculaire* abrégée DIO (également nommée *distance inter pupillaire*) dépend de l'âge, du sexe, et de l'origine ethnique de la personne.

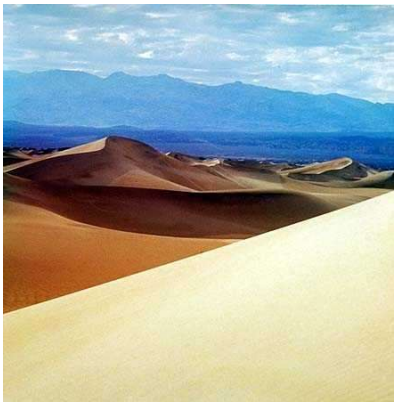


Figure 3.4 – Altération de la visibilité avec la distance

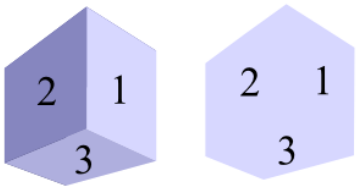


Figure 3.5 – Importance des ombres pour la perception des formes géométriques

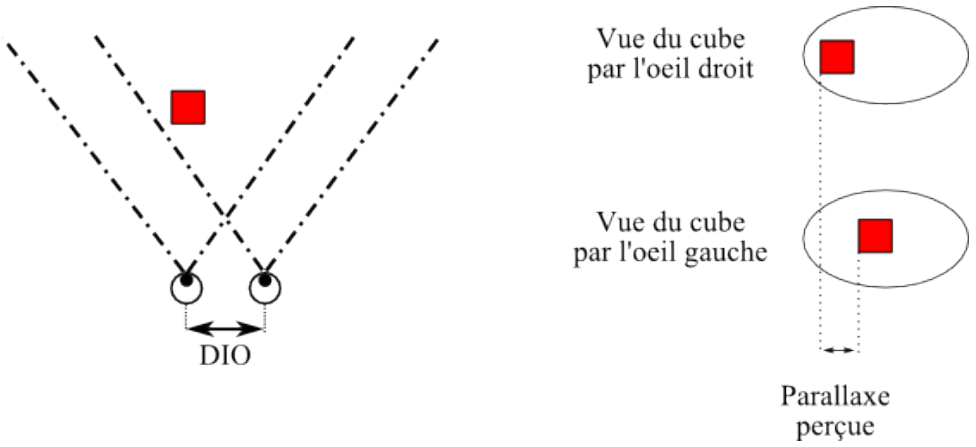


Figure 3.6 – Vision stéréoscopique humaine : à gauche vue du dessus d'une paire d'yeux observant un cube rouge. A droite, la vue de ce cube "à travers les yeux" gauche et droit.

Deux images légèrement différentes sont recueillies par les yeux (cf. Figure 3.6). A partir de ces deux vues, le cerveau va effectuer un raisonnement géométrique, à partir d'un appariement entre des points caractéristiques présents dans les deux images. Ce raisonnement permet à l'homme de percevoir l'environnement en relief et de récupérer des informations sur les espacements entre les objets vus, de même que sur leurs géométries. Grâce à cette action, l'homme sait évaluer des scènes 3D complexes ou inconnues.

Dans la suite nous présentons les deux mécanismes musculaires de l'oeil, l'accommodation et la convergence qui participent à la perception d'une seule image nette par le cerveau, à partir des images gauche et droite (principe de la vision binoculaire). Mais, même s'il n'y a pas de lien physique entre eux, ils agissent de concert dans ce but. Ensuite, nous détaillons la perception en profondeur relative chez l'humain.

Les mécanismes musculaires de l'oeil

- Accommodation : il s'agit du phénomène qui permet le passage de la vision lointaine à la vision proche, et inversement. La rétine de l'oeil permet l'obtention de la netteté de l'objet sur lequel on focalise. Ce phénomène est réalisé par les muscles ciliaires

(cf. Figure 3.1) qui vont intervenir pour changer la forme du cristallin, qui fait alors office d'une lentille biologique, pourvue d'une mise au point variable selon la contraction de ces muscles. Lorsque l'objet est éloigné, le corps ciliaire est contracté, et le cristallin est peu bombé (peu convergent). Si l'objet se rapproche et que l'oeil n'accommode pas, l'objet sera perçu comme flou, car son image se formera derrière la rétine. Les muscles ciliaires vont donc se relâcher, et permettre au cristallin de devenir plus convergent (en augmentant sa courbure). L'image devient se formera sur la rétine et deviendra de ce fait nette autour du point fixé.

– Convergence : la vision humaine donne la possibilité à nos yeux de regarder dans de nombreuses directions. La convergence consiste à orienter simultanément les axes principaux (ou axes visuels) des deux yeux vers le même point 3D dans l'espace, usuellement appelé point de fixation (ou point de convergence). Le point de convergence le plus proche est atteint lorsque nous ne sommes plus en mesure de percevoir une seule image (par exemple, l'observation d'un doigt collé au nez).
Naturellement, lorsque l'on regarde à l'infini, nos deux yeux ont leurs axes parallèles. Mais, si la convergence est un phénomène qui est naturel pour l'humain, la divergence (les deux yeux regardant dans des directions opposées) est très difficile à réussir pour l'homme.
Il est, par conséquent, primordial lors de la restitution d'images stéréoscopiques, de bien s'assurer que le spectateur ne divergera pas, ni ne convergera au delà de sa limite. C'est la raison qui impose que les jaillissements de chaque côté de l'écran, soient bien encadrés et restent dans des limites acceptables pour la grande majorité des personnes.

La perception humaine de la profondeur

Prenons le cas du système visuel humain, les yeux observant deux objets (P et Q) situés à des distances différentes de la tête. Des représentations de ces objets se forment sur les rétines droite et gauche. Un point 3D P et ses projections P_L/P_R sur les rétines gauche/droite (respectivement) forment un angle γ_P (cf. Figure 3.7). Si P est le point de fixation (convergence), les projections P_L/P_R sont situées sur les axes principaux des yeux gauche et droit (respectivement).

Un point Q (autre point de fixation) donne également naissance à deux projections Q_L/Q_R, qui forment un angle γ_Q. Cet angle est en général différent de γ_P et l'on appelle la différence entre ces deux angles, la *disparité rétinienne* : $d = \gamma_P - \gamma_Q$. Un point Q_1 se trouvant devant (resp. derrière) le point de fixation aura donc une disparité négative (resp. positive).

L'ensemble des points 3D H générant une disparité nulle $d = \gamma_P - \gamma_H = 0$, définit l'horoptère $\mathbb{H}$. $\mathbb{H}$ représente un hémisphère de rayon la distance au point de fixation P.

Pour résumer, les indices visuels monoculaires se basent sur la vision d'un seul oeil. Cependant, comme il s'agit d'un mécanisme neuro-biologique simple, ils peuvent parfois être victimes d'ambiguïtés. L'indice visuel binoculaire est une perception plus complexe, liée à des appariements entre des points caractéristiques communs dans les deux images. Mais elle est également plus robuste car elle se base sur un raisonnement géométrique, sans connaissance préalable.

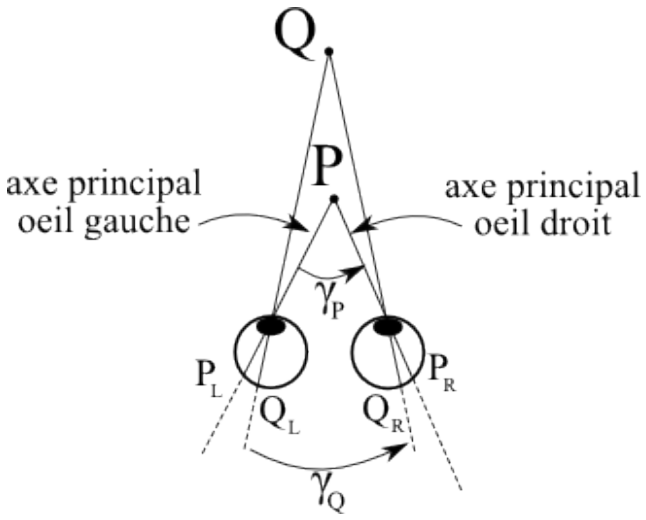


Figure 3.7 – Disparité rétinienne

3.1.3 Perception de la profondeur à partir d'images projetées

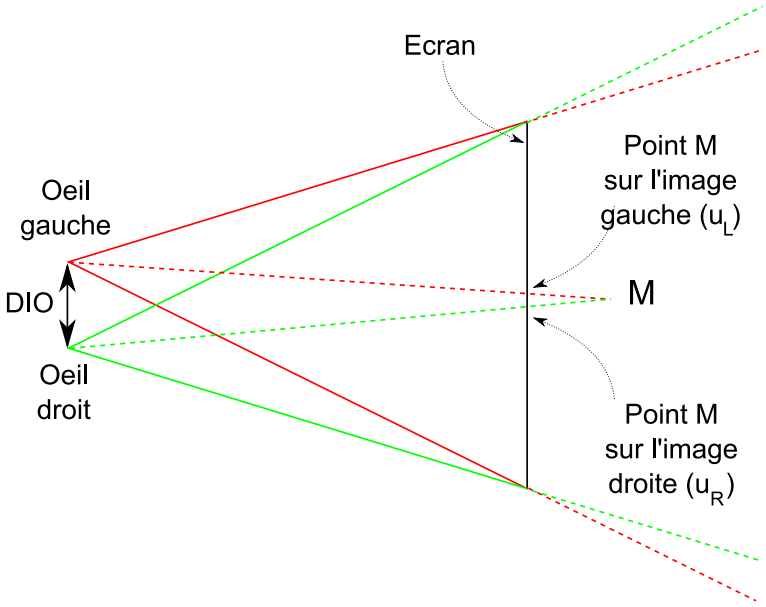


Figure 3.8 – Perception de la profondeur à partir d'images projetées sur un écran

Le principe de la perception de la profondeur, à partir d'images projetées, se base sur le système de vision stéréoscopique humaine décrit précédemment. Nous allons recréer une perception de profondeur chez l'utilisateur en affichant deux images grâce à un système de projection qui va permettre à chaque oeil de percevoir une image équivalente à une vue de la scène réelle (voir la section de l'état de l'art 2.2).

Afin d'illustrer, sur un exemple simple le principe de la perception de la profondeur, nous allons ici nous intéresser aux systèmes qui projettent deux images (repérées par leur couleur sur la Figure 3.8 : rouge pour l'image gauche, vert pour l'image droite) décalées horizontalement, sur un même écran (de type écran de cinéma ou écran de PC).

Un point de la scène, présent sur les deux images (u_L pour le point sur l'image gauche et u_R pour celui de l'image droite), sera donc situé à deux positions différentes sur l'écran. Ce décalage est appelé parallaxe (cf. Figure 3.8).

$$\text{parallaxe} = u_R - u_L$$

La parallaxe peut être soit positive (ou non croisée), soit négative (ou croisée), soit nulle. Ces valeurs de parallaxe ont pour conséquence de faire percevoir, à l'utilisateur du système, le point de la scène respectivement derrière l'écran, devant l'écran, et dans le plan de l'écran (point perçu repéré par la lettre M sur la Figure n°3.9).

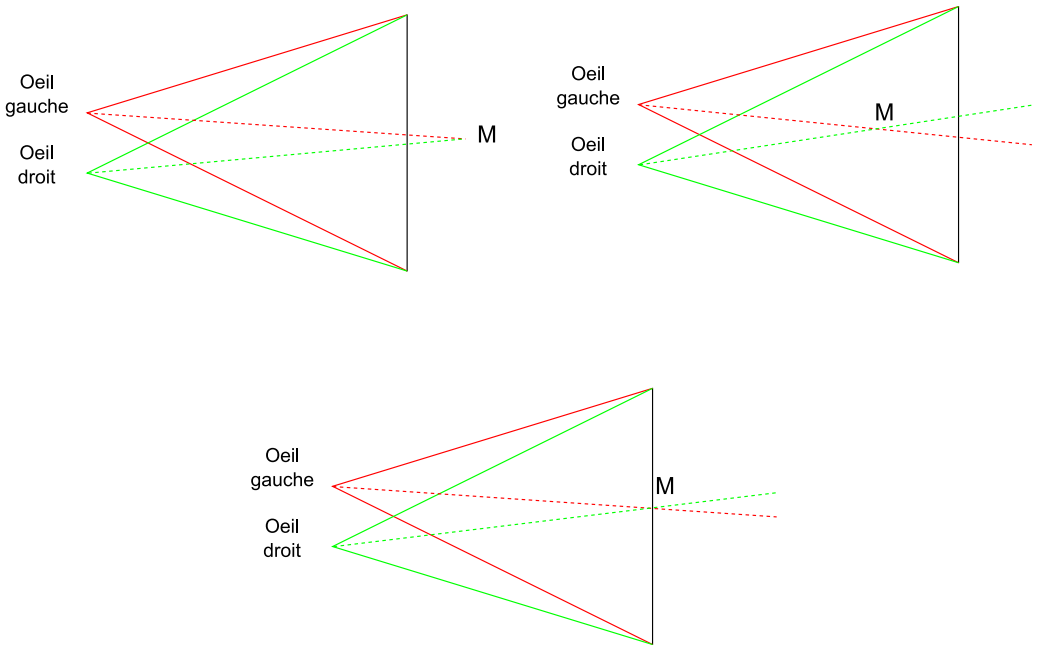


Figure 3.9 – Exemples de parallaxe positive (en haut à gauche), nulle (en bas) et négative (en haut à droite)

En vision normale, les procédés de focalisation de l'oeil humain sur des objets à des distances différentes (accommodation) et de convergence/divergence sont liés par des réflexes musculaires, comme nous l'avons vu dans la partie précédente sur la description du système de vision humaine.

Cependant, un problème se pose lorsque l'on observe un rendu stéréoscopique, car le principe même de la projection va imposer que ces procédés ne soient plus fortement liés. Dans notre cas, les yeux focalisent à une distance fixe sur l'écran pour pouvoir percevoir les images, de manière nette. Mais ils convergent à des distances différentes pour percevoir les objets de la scène en relief.

Etant donné que notre vision naturelle n'est pas habituée à ce phénomène, bien souvent lorsque l'effet de profondeur impose de converger trop loin de l'écran, les utilisateurs ressentent des maux de tête après quelques minutes, si les effets de profondeur sont trop importants. Pour palier à ce problème, il va être primordial que les objets en relief soient ne jaillissent pas trop en dehors de l'écran (autant devant que derrière).

On retrouve dans la littérature que la limite de parallaxe P est liée à une valeur maximale de la disparité. Les limites humaines sont atteintes lorsque la disparité entre le point de fixation sur l'écran et le point de convergence sur l'objet à visualiser est égale à 1.6 ° ([Fuchs 06] [Lipton 82]). Cette disparité apparaît approximativement lorsque la parallaxe atteint une valeur maximale, égale à $|P|_{max} = 0.03W$ où W représente la largeur de l'écran de restitution.

3.1.4 Visualisation d'images stéréoscopiques et inconfort visuel

A chaque étape de la transmission stéréoscopique, la perception finale du relief par l'utilisateur peut être gênée par de nombreux facteurs. [Boev 08] a répertorié de manière très complète tout ces effets.

3.1.4.a Déformation trapézoïdale et courbure de l'espace perçu

Lors de la restitution de contenu stéréoscopique, en particulier lorsque la capture a été réalisée en configuration convergente, des déformations peuvent apparaître sur les images et provoquer des sensations indésirables à l'utilisateur, lors de la restitution de celles-ci.

[Woods 93] font référence à quelques déformations stéréoscopiques typiques que sont la déformation trapézoïdale (keystone distortion) et la courbure du rendu 3D (depth-plane curvature).

Lors de la capture stéréoscopique par des caméras convergentes (les axes optiques des caméras n'étant pas parallèles entre eux), des parallaxes horizontales et verticales vont apparaître lors de la restitution (cf. Figure 3.10).

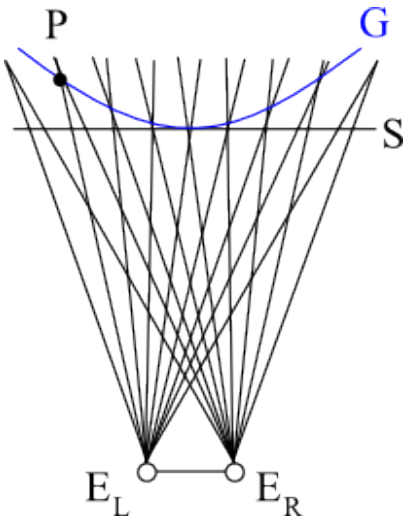
Des valeurs incorrectes de parallaxes horizontales vont engendrer une perception erronées de la profondeur de certains points de l'image, ce qui peut générer des effets de courbure de l'espace perçu. Les points les plus éloignés du centre de l'image seront perçus plus loin par l'utilisateur que ceux du centre de l'image (cf. Figure 3.10a).

Des valeurs incorrectes de parallaxe verticale vont quant-à eux rendre plus difficile la perception d'une seule image relief à partir des images stéréoscopiques droite et gauche. Plus la valeur de la parallaxe verticale est importante, plus le spectateur ressentira de la gène lors de la visualisation du contenu stéréoscopique.

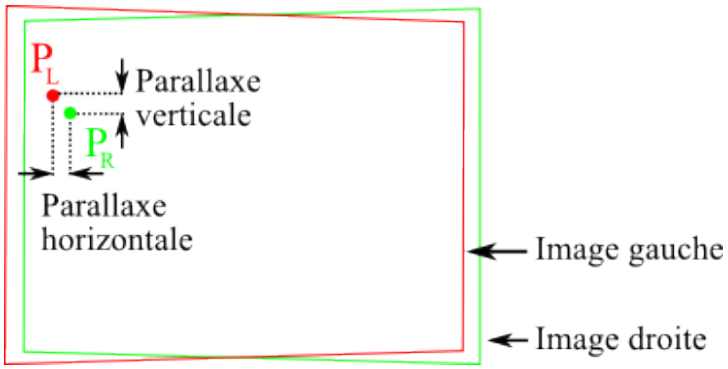
3.1.4.b Effet de grossissement ou de miniaturisation

Les déformations de la taille des objets apparaissent lorsque leur taille perçue par l'oeil et la distance à laquelle ils sont perçus ne correspondent pas à ce que l'on aurait constaté de manière naturelle. Dans le monde "réel", lorsque la taille d'un objet change, cela signifie que sa distance par rapport à nous change également. Cependant, lors d'une projection stéréoscopique, des repère visuels différents peuvent cohabiter et donner en conséquence des informations de profondeur contradictoires, ce qui va directement influencer sur le ressenti de la taille des objets (grossissement ou miniaturisation).

Parmi ces effets déformants, on trouve l'effet carton (Cardboard Effect). Il s'agit d'une impression non naturelle que les objets ou les personnes présentes sur les images stéréoscopiques ont été découpés et collés par couche sur des plaques de carton. Cet effet a principalement pour origine la manière dont a été réalisée la capture stéréoscopique (longueur de focale, distance objet-camera, distance de convergence)



(a) Perception erronée d'un plan G en vision stéréoscopique



(b) Effets de parallaxe verticale et horizontale (exagérées) sur deux points images P_L et P_R, projections d'un point P de la scène.

Figure 3.10 – Deux types de déformations stéréoscopiques [Woods 93]

Un autre effet couramment rencontré est l'effet dit du théâtre de marionnettes (Puppet Theater Effect). Il s'agit d'un problème ennuyeux qui consiste en une miniaturisation de la scène filmée. Les personnes ou objets de la scène ressemblent à des marionnettes animées. Cet effet est créé par une inconsistance entre la position en relief (profondeur) à laquelle apparaît un objet et sa taille perçue à l'écran [Schreer 05].

[Yamanoue 97] a démontré que ce phénomène n'avait pas lieu lorsque des caméras parallèles étaient utilisées pour la capture stéréoscopique, et ne dépendait pas des conditions de projection des images. Dans cet article, il propose également une méthode pour évaluer l'influence de cet effet.

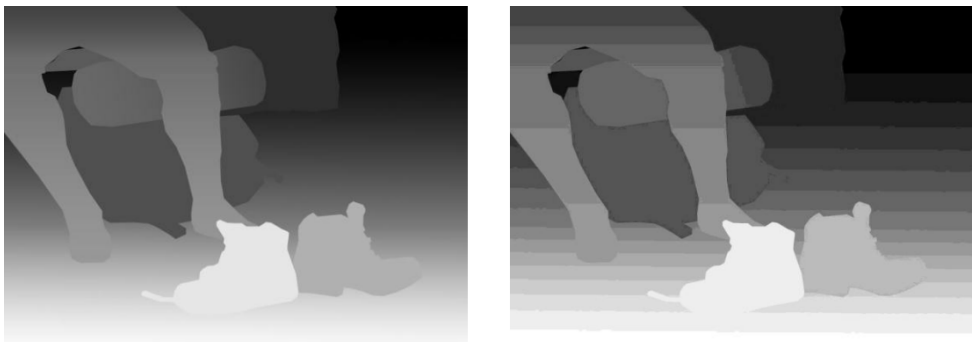


Figure 3.11 – Illustration de l'effet carton (droite) sur une image de carte de profondeur (gauche) [Boev 08]

3.1.4.c Images fantômes

Lorsque l'on étudie la restitution des images, il y a un élément qui peut s'avérer très perturbant, il s'agit de l'apparition du phénomène d'images fantômes (Crosstalk ou ghosting). Ce phénomène vient de la séparation imparfaite entre les images droite et

gauche lors de la restitution. Il dépend beaucoup de la technologie utilisée. Lorsque l'on utilise des tubes cathodiques (CRT : Cathode Ray Tube), le phosphore dont ils sont composés induit une persistance plus longue, ce qui retarde la disparition de l'image précédent l'image courante. Nous voyons donc deux images superposées. L'humain est sensible à ce phénomène même si sa durée est faible.

Ce problème arrive fréquemment lorsque l'on utilise la technique de polarisation droite et que l'utilisateur souhaite incliner sa tête (il va percevoir quelques parties de l'autre image polarisée dans l'autre sens). C'est une des raisons qui pousse l'abandon de cette technologie au profit d'une polarisation circulaire lorsque ces mouvements doivent être permis. Les techniques de projection auto stéréoscopiques sont aussi touchées par ce phénomène, car les lentilles directionnelles souffrent de latence lors du déplacement de l'utilisateur.

3.1.4.d Effet de cisaillement

L'effet de cisaillement est le plus souvent observé lors de l'utilisation d'un système d'affichage stéréoscopique qui ne permet l'affichage d'une scène 3D selon un seul point de vue. Dans ce cas, la vue correcte n'est calculée que pour un point précis d'observation devant le système d'affichage. Si l'utilisateur s'écarte de cette position, il va percevoir une vue déformée de l'environnement (mauvaise perspective). La solution la plus simple a ce problème, réside dans le suivi du mouvement de la tête de l'utilisateur, qui va permettre ensuite d'ajuster l'affichage en fonction de son déplacement, et de projeter les images recalculées.

3.1.4.e Effets de palissade et de feuillettement

Lors de l'utilisation de dispositifs de visualisation auto stéréoscopique à vues multiples, si l'utilisateur se déplace latéralement, il peut percevoir deux effets que sont l'effet de palissade et l'effet de feuillettement [Meesters 04]. Le premier (Picket-Fence Effect en anglais) se caractérise par l'apparition de bandes verticales (sombres et claires alternativement) dans l'image. Le second (Image Flipping en anglais) se distingue par le passage d'une image qui représente une vue calculée à celle qui représente une autre vue.

Ces sources d'inconfort visuel sont un élément primordial à prendre en compte pour améliorer l'expérience de la vision en relief pour un utilisateur. Nous avons donc accordé une attention particulière à la prise en compte des causes de ces inconforts dans notre travail.

3.2 Modèle de caméra monoscopique

Dans cette partie, nous présentons le modèle général d'une caméra vidéo. Il est constitué du modèle de projection du sténopé et de la modélisation des déformations dues aux optiques et au CCD. Ensuite nous détaillons le modèle particulier du modèle canonique et son intérêt.

Enfin, nous intéressons à la modélisation d'un banc stéréoscopique dans le cas général, puis dans le cas canonique.

3.2.1 Modèle de caméra du sténopé

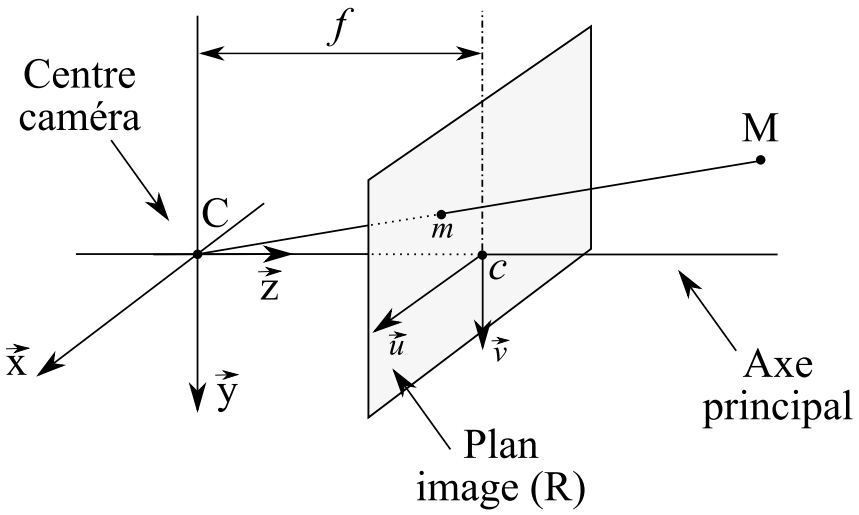


Figure 3.12 – Modèle de caméra sténopé

Une caméra vidéo est modélisée dans le cas général en utilisant le modèle du sténopé (pinhole camera model en anglais [Faugeras 93]). Ce modèle est constitué d'un plan 2D R appelé le plan image et d'un point C, appelé le centre de projection, qui n'appartient pas à R. Le plan image reçoit les rayons de lumière réfléchis par les objets de la scène.

Une caméra permet de capturer sur le plan image les projections 2D d'objets d'une scène 3D. La projection perspective $\underline{m}^n$ d'un point $\underline{X}^v$ de l'espace est formée par l'intersection du rayon optique (C,$\underline{X}^v$) avec le plan image R, et crée l'image de $\underline{X}^v$. L'axe optique est la ligne qui passe par C et qui est perpendiculaire au plan image. L'intersection de cette ligne avec le plan image est nommée c, le point principal. La distance entre le point C et le plan R est appelée la focale f de la caméra (cf. Figure 3.12).

3.2.2 Modélisation des déformations

On définit un système de coordonnées 3D, F_v {C,($\vec{e_x}$,$\vec{e_y}$,$\vec{e_z}$)} centré en C, et appelé système de coordonnées de visualisation.

Soit le point 3D $\underline{X}^v$ de la scène qui appartient à F_v :

$$\underline{X}^v = \begin{bmatrix} x^v \\ y^v \\ z^v \end{bmatrix}$$

Les coordonnées des points images sont exprimés en pixel dans un système de coordonnées 2D, F_i {c,($\vec{e_x}$,$\vec{e_y}$)} centré en c, et appelé système de coordonnées image.

Selon le modèle de caméra du sténopé, la projection perspective d'un point sur un plan 2D virtuel, placé à une distance f de la caméra, appartient au système de coordonnées image et est donnée par :

$$\underline{m}^n = \begin{bmatrix} x^n \\ y^n \end{bmatrix} = \begin{bmatrix} x^v/z^v \\ y^v/z^v \end{bmatrix}$$

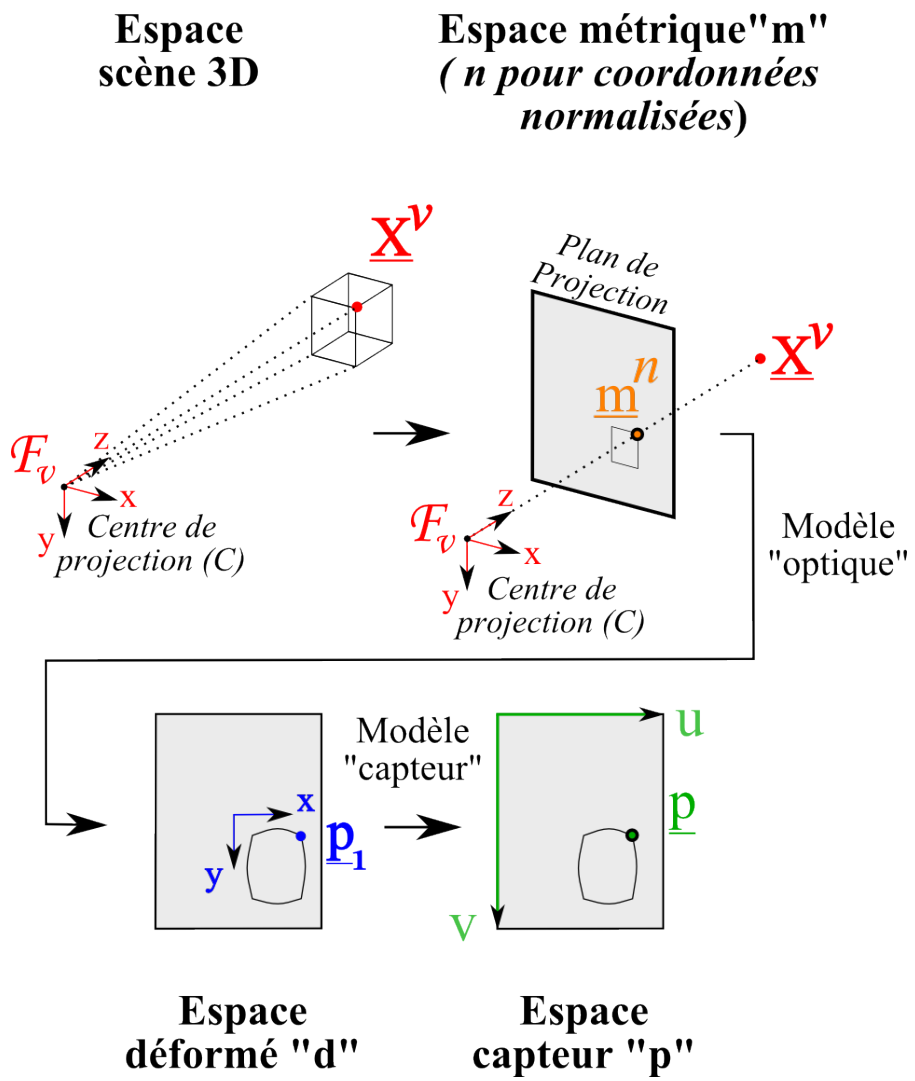


Figure 3.13 – Modélisation de la capture d'une image avec une caméra sténopé

Lorsque la lumière va passer au travers des lentilles, les rayons vont être déformés. Nous devons donc tenir compte des effets de l'optique en utilisant un modèle de distorsion radiale et tangentielle (cf. Figure 3.13).

L'amplitude de ce déplacement dépend de la valeur de r, avec $r^2 = (x^n)^2 + (y^n)^2$ qui représente la distance entre le point $\underline{m}^n$ du plan 2D et le centre de projection perspective.

La position du point, après ce déplacement, est obtenue en utilisant la formule suivante qui utilise les coefficients polynomiaux K_1, K_2, K_3, K_4, and K_5 du modèle des déformations radiales et tangentielles (ces coefficients regroupent les caractéristiques physiques de chaque caméra). Les coefficients K_1, K_2 et K_5 se rapportent aux déformations radiales, alors que K_3 et K_4 se rapportent aux déformations tangentielles.

$$\underline{m}^d = \begin{bmatrix} x^d \\ y^d \end{bmatrix} = (1 + K_1\, r^2 + K_2\, r^4 + K_5\, r^6)\, \underline{m}^n + dx$$

avec

$$dx = \begin{bmatrix} 2\ K_3\ x^n\ y^n + K_4\ (r^2 + 2\ (x^n)^2) \\ 2\ K_4\ x^n\ y^n + K_3\ (r^2 + 2\ (y^n)^2) \end{bmatrix}$$

Chaque caméra est caractérisée par ses paramètres "internes" (paramètres intrinsèques) : les dimensions de la surface de son capteur, l'alignement de son capteur par rapport aux optiques,... Ces paramètres sont représentés par K dans l'équation suivante. Les caractéristiques du capteur CCD déterminent le passage des coordonnées capteur aux coordonnées après déformation :

$$\underline{p}_1 = K \begin{bmatrix} x^d \\ y^d \\ 1 \end{bmatrix} = \begin{bmatrix} f_1 & \alpha_c & u_c \\ 0 & f_2 & v_c \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x^d \\ y^d \\ 1 \end{bmatrix}$$

où $\underline{p}_1$ est exprimé en coordonnées capteur (i.e. unités pixeliques). Ici, f_1 et f_2 décrivent les distances focales en pixels, respectivement horizontaux et verticaux. u_c et u_v correspondent aux coordonnées du point principal, et α_c est la variable de déviation qui décrit la non-orthogonalité des axes (ce paramètre est bien souvent négligé, i.e. on lui attribue une valeur nulle).

On a :

$$\underline{p}_1 = \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix}$$

avec $u_1 \in (0, n_x - 1)$ et $v_1 \in (0, n_y - 1)$ où n_x et n_y représentent respectivement le nombre d'éléments du capteur le long des direction u et v

Pour obtenir les coordonnées du point dans l'espace normalisé associé au dispositif, nous effectuons l'opération suivante :

$$\underline{p} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

avec $u \in (0, 1)$ et $v \in (0, 1)$ et $u = \dfrac{u_1 + 0.5}{n_x}$ $v = \dfrac{v_1 + 0.5}{n_y}$

3.2.3 Modèle de caméra canonique

Le modèle canonique est basé sur le modèle de caméra du sténopé décrit précédemment. Il s'agit d'une configuration particulière du modèle où le centre de projection est situé au centre de la caméra, le plan image est parallèle au plan xy et situé à la distance $f = 1m$ du centre, la caméra est dirigée le long de l'axe z (cf. Figure 3.14).

Ce modèle simplifie le processus de création de l'image au strict minimum. Toute influence des paramètres spécifiques liées à la caméra et à l'optique a été enlevée. Seule reste l'opération de projection perspective, c'est à dire le passage d'un point 3D de la scène à un point 2D sur le plan image.

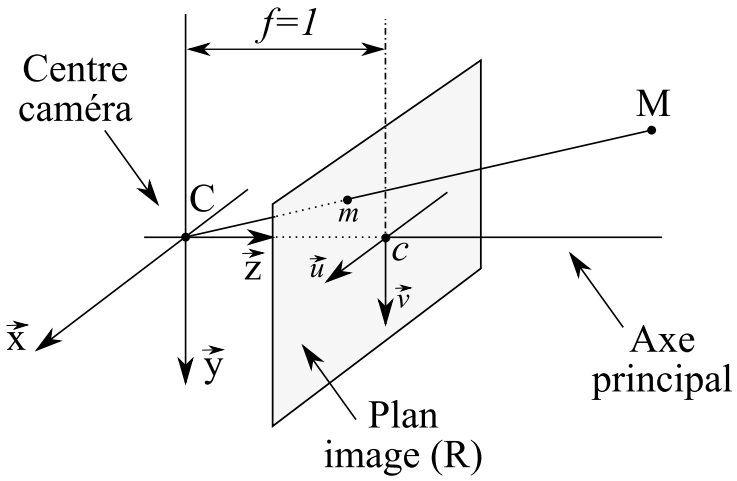


Figure 3.14 – Modèle de caméra canonique

On définit un système de coordonnées 3D, F_v {C,($\vec{e_x}$,$\vec{e_y}$,$\vec{e_z}$)} centré en C, et appelé système de coordonnées de visualisation.

Soit le point 3D $\underline{X}^v$ de la scène qui appartient à F_v :

$$\underline{X}^v = \begin{bmatrix} x^v \\ y^v \\ z^v \end{bmatrix}$$

Les coordonnées des points images sont exprimés en pixel dans un système de coordonnées 2D, F_i {c,($\vec{e_x}$,$\vec{e_y}$)} centré en c, et appelé système de coordonnées image.

Selon le modèle de caméra canonique, la projection perspective d'un point sur un plan 2D virtuel, placé à une distance $f = 1m$ de la caméra, appartient au système de coordonnées image et est donnée par :

$$\underline{m}^n = \begin{bmatrix} x^n \\ y^n \end{bmatrix} = \begin{bmatrix} x^v/z^v \\ y^v/z^v \end{bmatrix}$$

Ce modèle de caméra canonique est également dépourvu de déformations radiales et tangentielles. Soit ces déformations n'existent pas (cas d'une caméra théorique parfaite) soit elles ont été éliminées par correction. C'est à partir de ce modèle de caméra que nous allons travailler pour modéliser la capture vidéo dans le chapitre 4. La section 5.1 présente elle, la méthode que nous avons développée pour permettre de travailler avec des images issues d'un banc stéréoscopique canonique (cf. section 3.3.2), à partir d'un banc stéréoscopique réel (cf. section 3.3.1).

3.3 Modèle de caméra vidéo stéréoscopique

Nous venons de présenter dans la section précédente les modèles de caméra vidéo sténopé et canonique. Nous allons détailler dans cette section, la modélisation d'un banc stéréoscopique dans le cas général, puis dans le cas canonique.

3.3.1 Banc stéréoscopique : modèle général

Un système composé de deux caméras de ce type possède une contrainte géométrique particulière appelée *contrainte épipolaire* (pour plus de détails se référer à [Faugeras 93]). Cette contrainte concerne la projection d'un point 3D $\underline{X}^o$ sur les plans images des caméras gauche et droite, $\underline{m}_l$ and $\underline{m}_r$ (voir la Figure 3.15).

En effet, $\underline{m}_l$ et $\underline{m}_r$ sont contraints de se situer dans le plan de la scène, comprenant le point $\underline{X}^o$ et les centres de projection $\underline{C}_L$ et $\underline{C}_R$, plan que l'on appelle *plan épipolaire* associé à $\underline{X}^o$.

Si l'on connaît la géométrie d'une caméra, pour un point $\underline{x}_l$ donné, la position du point $\underline{x}_r$ correspondant est limitée à l'intersection du plan épipolaire et du plan image. Cette ligne est appelée la *ligne épipolaire* correspondant à $\underline{x}_l$.

En ce qui concerne les projections $\underline{m}_L$ et $\underline{m}_R$ d'un même point $\underline{X}^o$ de la scène, nous avons (d'après [Faugeras 93]) :

$$\left(\underline{m}_L\right)^T E\, \underline{m}_R = 0 \text{ avec } E = [t_e]_x R_e$$

où la matrice essentielle E est définie par le positionnement relatif R_e,t_e des deux caméras.

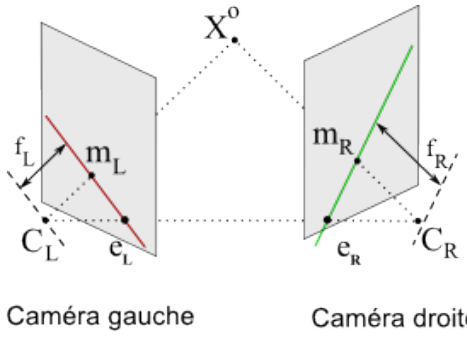


Figure 3.15 – Modèle général de caméras stéréoscopiques. C_l et C_r représentent les points focaux des deux caméras. Le point $\underline{m}_r$ (resp. $\underline{m}_l$) représente l'intersection de la ligne passant par le point X (point d'intérêt) et C_r (resp. C_l) . Le point $\underline{e}_l$ (resp. $\underline{e}_r$), appelé *épipole droit*, est l'intersection de l'autre point focal C_r (resp. C_l) sur le plan image gauche (resp. droit). Les lignes $\underline{e}_l - \underline{m}_l$ et $\underline{e}_r - \underline{m}_r$ sont appelées *lignes épipolaires*.

3.3.2 Banc stéréoscopique : modèle canonique

Selon le modèle de caméra canonique décrit dans la section 3.2.3, les différentes étapes de la transformation des coordonnées images canoniques $\underline{m}^n$ vers des coordonnées images normalisées $\underline{p}$ peuvent-être modélisées simplement par une projection centrale.

$$\lambda \underline{m} = \underline{X}^v \text{ avec } \underline{X}^v = R\underline{X}^o + t$$

où le scalaire λ représente la profondeur en coordonnées caméra et R,t la position de la camera.

Un banc stéréoscopique canonique est modélisé en utilisant deux caméras canoniques. Les images capturées par ce banc sont donc dépourvues de déformations, ce qui rend l'exploitation des images très précise.

De plus, les deux caméras sont alignées horizontalement et leurs plans image sont coplanaires (les lignes épipolaires sont parallèles, et placées à une même hauteur horizontale). Elles ont également les mêmes paramètres intrinsèques.

Ce modèle de banc stéréoscopique canonique présente l'autre avantage de ramener une recherche 2D d'un point caractéristique de l'image droite, sur l'image gauche (entière), à une recherche 1D le long de la ligne épipolaire correspondante. Cet avantage offre un gain de temps considérable, par exemple, lors l'utilisation d'algorithmes d'appariement, pour la reconstruction 3D.

Nous détaillerons, dans la section 5.1.1.b, la rectification d'un banc stéréoscopique, qui permet de passer du modèle général d'un banc stéréoscopique au modèle canonique.

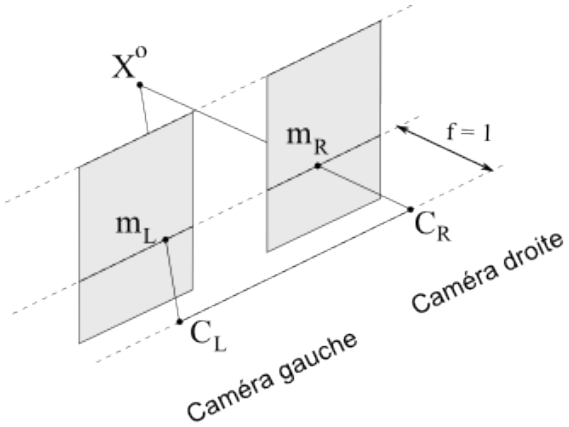


Figure 3.16 – Modèle de banc stéréoscopique canonique.

3.4 Images stéréoscopiques

3.4.1 Deux configurations de capture d'images

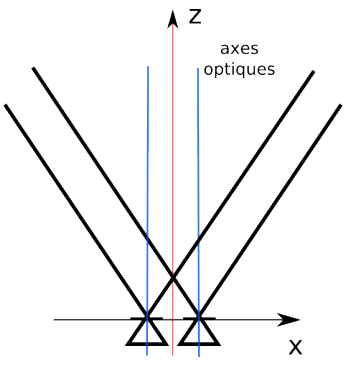


Figure 3.17 – Configuration parallèle d'un banc stéréoscopique

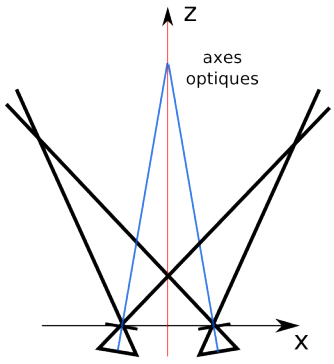


Figure 3.18 – Configuration convergente d'un banc stéréoscopique

Pour capturer des scènes du monde réel ou d'un environnement virtuel avec des caméras stéréoscopiques (réelles ou virtuelles), deux configurations sont possibles et s'opposent.

La première est dite convergente (ou "toed-in"). Les axes optiques des caméras sont convergents (cf. Figure 3.18).

La seconde configuration est dite parallèle. Dans ce cas, les axes optiques des caméras sont parallèles (cf. Figure 3.17).

Chaque configuration possède ses défauts et ses avantages, et l'utilisation de l'une ou de l'autre, dépend en grande partie de l'application recherchée. En effet, un système de caméras convergentes va, par exemple, imposer de retoucher les images pour compenser les déformations dues aux angles (cf section 3.1.4.a). Un système de caméras parallèles va quant-à lui nécessiter de retoucher les images pour éliminer les parties non visibles (cf section 4.2.3).

3.4.2 Capture d'images dans un environnement virtuel : utilisation de caméras virtuelles

La génération d'images de synthèse pour un affichage en stéréoscopie est basée sur le même principe que celui de la vision stéréoscopique des yeux. Deux images (vues gauche et droite) d'une scène sont perçues par notre cerveau qui les fusionne. Par analogie, dans une simulation numérique, deux images (gauche et droite) vont être calculées. Ces deux images seront ensuite projetées chacune sur un oeil, en utilisant les technologies de projection en relief précédemment décrites.

De plus, il va être important de tenir compte de la position de l'utilisateur (fixe ou mobile) par rapport à l'écran, lors de la restitution de ces images.

L'environnement à restituer et à percevoir en relief est un environnement virtuel, composé d'objets 3D. Les vues gauche et droite, de la scène virtuelle, à projeter pour chaque oeil, vont être obtenues par le biais de deux caméras virtuelles. Ces deux caméras sont basées sur le modèle classique utilisé en image de synthèse.

On représente une caméra virtuelle en définissant le volume visible par celle-ci au sein de l'environnement virtuel. Ce volume (*viewing frustum* en anglais) permet de définir les limites (plans *near*, *far*, *right*, *left*, *top*, et *bottom* de la Figure 3.19) au sein desquels les objets de l'environnement seront considérés comme visibles par la caméra. Un caméra virtuelle standard est une caméra virtuelle symétrique. Les plans *right*/*left* et *top*/*bottom* sont situés à égale distance de l'axe de la caméra. Seuls les plans *near* et *far* sont réglables en profondeur le long de cet axe.

Mais il existe des caméras virtuelles dites asymétriques (*asymmetric viewing frustum*). Elles autorisent en plus du réglage de la position des plans *near*/*far*, un réglage précis de la géométrie du viewing frustum, en jouant aussi bien sur sa largeur que sur sa hauteur (cf. Figure 3.20). Ces caméras sont très utilisées dans les applications de Réalité Virtuelle, en particulier sur les systèmes à multi-écrans avec tracking de l'utilisateur.

De plus, pour permettre la projection de deux images différentes pour chaque oeil de l'utilisateur, il va falloir mettre en place deux caméras virtuelles stéréoscopiques. Il y a deux méthodes concurrentes pour mettre en place une capture stéréoscopiques de la scène virtuelle [Bourke 99].

La première dite méthode à caméras convergentes, ou "toed-in", se base sur des caméras symétriques. Cependant, elle n'est que peu utilisée en général car elle peut générer

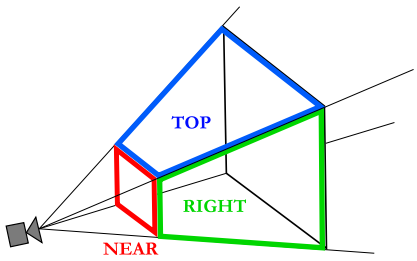


Figure 3.19 – Limites du viewing frustum d'une caméra virtuelle. Les plans right(left), top(bottom), et near(far) délimitent la zone capturée par la caméra.

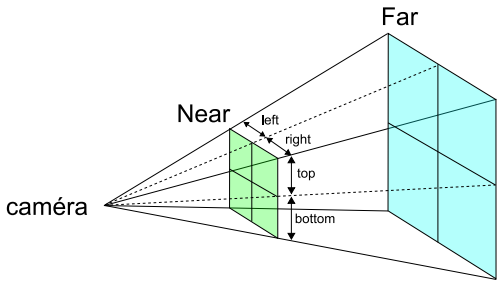


Figure 3.20 – Paramètres de réglage d'une caméra virtuelle asymétrique

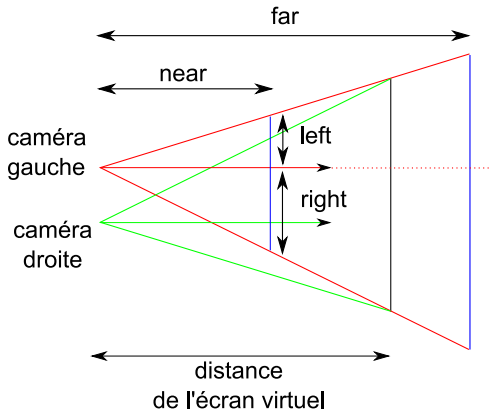


Figure 3.21 – Deux caméras virtuelles asymétriques. L'écran virtuel se situe à l'intersection entre les deux viewing frustum des caméras gauche et droite.

des effets importants de parallaxe verticale, en particulier lorsqu'on s'écarte du centre de l'image en direction des bords.

La seconde méthode dite méthode "parallèle" est mieux adaptée à notre problème. Elle ne génère pas de parallaxe verticale, et permet la création de meilleures images. Cependant, il est nécessaire de faire appel à des caméras asymétriques.

Dans ce modèle, il va être important de connaître la position du plan image dans le champ des caméras (cf. Figure 3.21). En effet sur ce plan, les images droite et gauche seront strictement identiques (parallaxe nulle). Les objets situés au niveau de ce plan image seront perçus comme étant positionnés sur le support de projection des images stéréoscopiques (l'écran).

3.4.3 Capture d'images dans un environnement réel : utilisation d'un banc stéréoscopique

Un système de capture vidéo stéréoscopique (ou banc stéréoscopique) se base sur un couple de caméras vidéo qui sont fixées sur un support sur lequel on pourra régler leur écartement horizontal (la ligne de base stéréoscopique), de même que la position verticale

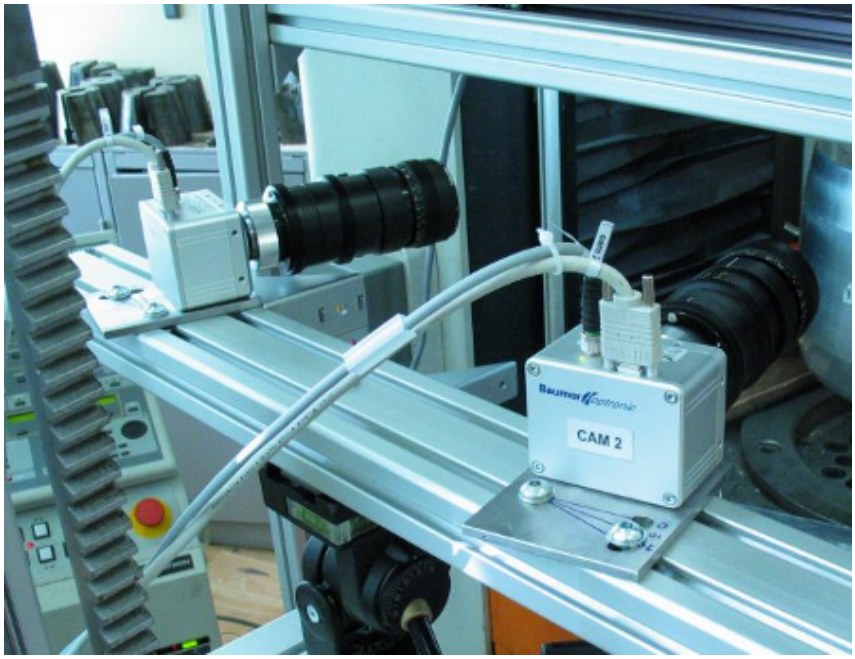


Figure 3.22 – Exemple d'un banc stéréoscopique permettant des réglages en translation et en rotation des caméras sur le plan horizontal

du support, et éventuellement leur convergence en réglant leur rotation selon l'axe vertical (cf. Figure 3.22). Ces caméras vont filmer le monde selon deux points de vue différents. Ces deux vidéos pourront par la suite être projetées en relief grâce aux technologies précédemment décrites.

Nous venons d'aborder les principes de la vision humaine qui permettent de percevoir du relief à partir de deux images, de même que les sources d'inconfort visuels qui peuvent gêner l'utilisateur lors de la visualisation de contenu stéréoscopique.

Nous avons également exposé la modélisation de caméras vidéo monoscopiques et stéréoscopiques, en décrivant les différentes étapes qui mènent à l'obtention d'une image à partir d'un ensemble de points 3D.

En nous basant sur ces modèles de caméra et sur les informations que nous possédons sur la vision humaine, nous allons détailler, dans les chapitres suivants, une modélisation de la chaîne de transmission des images stéréoscopiques, et une méthode de rectification des images pour permettre à un utilisateur de visualiser la scène filmée de manière non déformée.

Nous exposerons également une méthode qui autorise la perception par un utilisateur de profondeurs égales à leurs valeurs réelles (celles issues de la capture de la scène par le banc stéréoscopique) lors de son déplacement devant un système de Réalité Virtuelle.

Chapitre 4

Modélisation de la transmission stéréoscopique

Sommaire

La restitution maitrisée du relief d'une scène réelle nécessite de déterminer les relations qui existent entre les différents éléments qui interviennent lors de la capture et de l'affichage d'images stéréoscopiques.

Dans ce chapitre, nous décrivons la modélisation de l'ensemble du processus, de la capture à l'affichage, qui conduit à une perception en relief, à partir d'images stéréoscopiques. Ces relations mathématiques vont nous permettre de déterminer la position en relief -perçue grâce au système de restitution- d'un objet que l'on filme, en fonction des paramètres de capture de cet objet, dans la scène réelle.

Pour cela, nous avons distingué trois grandes étapes, qui distinguent la modélisation du système de capture, de celle du système de restitution :
- la modélisation de deux configurations de capture (cf. section 4.1)
- la modélisation de l'affichage vidéo stéréoscopique (cf. section 4.2)
- la modélisation de la perception de la profondeur (cf. section 4.3).

Dans cette modélisation, nous avons choisi de séparer les effets liés au matériel de capture/restitution des images (propriétés des caméras, des projecteurs,...), des effets associés à la transmission stéréoscopique proprement dite (écartement des yeux de l'utilisateur, position du plan à parallaxe nulle). Le regroupement des ces deux groupes d'effets (tel qu'on peut le trouver dans les travaux de [Fuchs 06] [Wartell 99] [Woods 93]) présente le désavantage de masquer l'essentiel des détails de la transmission stéréoscopique.

Notre postulat de départ est que notre système de capture (convergent ou parallèle) est constitué d'un banc stéréoscopique de deux caméras canoniques (caméras simplifiées à une projection centrale, suite à une calibration cf. section 3.3.2). La modélisation de la transmission d'images stéréoscopiques, est donc volontairement réduite au cas d'images non déformées.

Nous traiterons le problème de la rectification des images en amont de la modélisation (afin de supprimer les déformations dues au capteur et aux optiques), dans le chaptire 5.

Dans la suite, nous utiliserons plusieurs repères intermédiaires, pour les étapes de capture et de restitution :
- repères associés aux caméras
- repères associés au banc stéréoscopique
- repères associés à l'écran de restitution
- repères associés à l'utilisateur

Nous avons choisi d'adopter les normes de notation anglo-saxonnes pour les variables. Les indices suivants désignent :
- $l \equiv$ *left* (gauche en anglais)
- $r \equiv$ *right* (droite en anglais)
- $c \equiv$ *camera*
- $s \equiv$ *screen* (écran en anglais)
- $i \equiv$ *image*

Les coordonnées d'un objet dans le repère associé au banc stéréoscopique sont exprimées par (X_0, Y_0, Z_0). Cet objet va être projeté sur les deux capteurs CCD des caméras en deux points repérées par leurs coordonnées (X_{Cl}, Y_{Cl}) et (X_{Cr}, Y_{Cr}) sur les plans image gauche et droit. A partir de ces dernières, nous allons calculer les coordonnées dans chacune des images droite et gauche, dans le repère associé à l'écran (X_{Sl}, Y_{Sl}) et (X_{Sr},

Y_{Sr}). Enfin, un utilisateur visualisant ces deux images selon le principe de la stéréoscopie percevra un point en relief, qui aura pour coordonnées (X_i, Y_i, Z_i) dans le repère associé à l'utilisateur.

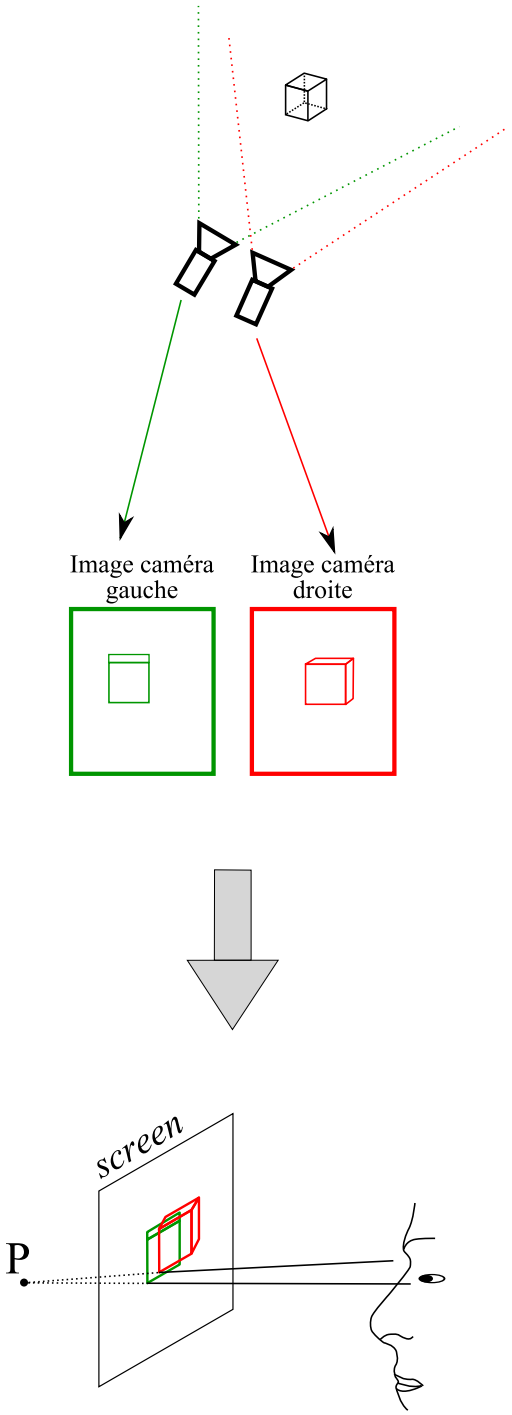


Figure 4.1 – Schéma de la transmission stéréoscopique : de la capture par un banc stéréoscopique à la perception de la profondeur (P) par l'utilisateur

4.1 Deux configurations de capture

4.1.1 Caméras en configuration parallèle

Dans cette partie, nous nous intéressons à mettre en relation les différents paramètres, intervenant lors d'un capture vidéo avec des caméras en configuration parallèle. Dans cette configuration, leurs plans image et leurs axes optiques sont parallèles entre eux. Nous partons d'un point Q de l'espace, et cherchons à déterminer ses coordonnées 2D sur chaque plan image des deux caméras gauche et droite.

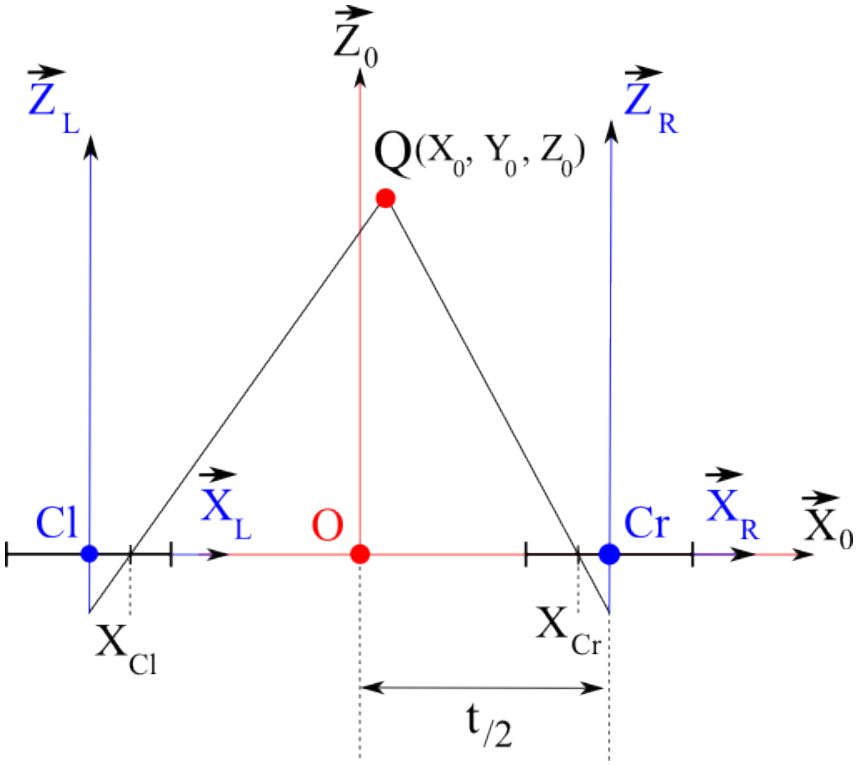


Figure 4.2 – Schéma des repères associés aux deux caméras canoniques en configuration parallèle

Dans le repère F_0(O,$\vec{X_0}$,$\vec{Y_0}$,$\vec{Z_0}$), un point Q de l'objet à filmer a pour coordonnées Q(X_0,Y_0,Z_0).

On définit deux repères F_{Cl} {Cl,($\vec{X_{Cl}}$,$\vec{Y_{Cl}}$,$\vec{Z_{Cl}}$)} et F_{Cr} {Cr,($\vec{X_{Cr}}$,$\vec{Y_{Cr}}$,$\vec{Z_{Cr}}$)} associés à chacune des deux caméras canoniques (cf. Figure 4.2).

Nos caméras étant canoniques, nous repérons la projection du point Q sur le capteur gauche grâce à ces deux coordonnées (X_{Cl},Y_{Cl}) et par (X_{Cr},Y_{Cr}) sur le capteur droit. La distance de la ligne de base est égale à t. Dans les repères associés aux deux caméras, cela donne :

$$\begin{cases} X_{Cl} = \dfrac{(t + 2X_0)}{2\ Z_0} \\ \\ X_{Cr} = -\dfrac{(t - 2X_0)}{2\ Z_0} \end{cases}$$

Mais, alors que nous avons des coordonnées différentes selon X, les coordonnées selon Y seront identiques (en conservant l'hypothèse que les capteurs sont sur un même plan en Y et Z). Nous avons toujours dans ces mêmes repères associés aux capteurs :

$$Y_{Cl} = Y_{Cr} = \frac{Y_0}{Z_0}$$

En résumé :

$$\begin{cases} \boxed{X_{Cl} = \dfrac{(t + 2X_0)}{2\ Z_0}} \\ \boxed{Y_{Cl} = \dfrac{Y_0}{Z_0}} \end{cases} \qquad \begin{cases} \boxed{X_{Cr} = -\dfrac{(t - 2X_0)}{2\ Z_0}} \\ \boxed{Y_{Cr} = \dfrac{Y_0}{Z_0}} \end{cases}$$

Nous avons donc obtenu les coordonnées image du point Q, sur chaque plan image des caméras gauche et droite, en configuration parallèle. Nous allons maintenant déterminer les coordonnées image du point Q, sur chaque plan image des caméras gauche et droite, en configuration convergente, avant de nous intéresser aux relations qui lient ces coordonnées à celle du point correspondant, perçu en relief sur un système immersif.

4.1.2 Caméras en configuration convergente

Dans cette partie, nous allons décrire la capture vidéo avec deux caméras en configuration convergente. Les deux caméras n'ayant plus leurs capteurs image dans le même plan, il va être nécessaire d'introduire les angles de rotation des caméras autour de l'axe Y_0.

Dans le repère $F_0(\mathrm{O},\vec{X_0},\vec{Y_0},\vec{Z_0})$, un point Q de l'objet à filmer a pour coordonnées $\mathrm{Q}(\mathrm{X}_0,\mathrm{Y}_0,\mathrm{Z}_0)$.
On définit deux repères F_{Cl} $\{\mathrm{Cl},(\vec{X_{Cl}},\vec{Y_{Cl}},\vec{Z_{Cl}})\}$ et F_{Cr} $\{\mathrm{Cr},(\vec{X_{Cr}},\vec{Y_{Cr}},\vec{Z_{Cr}})\}$ associés à chacune des deux caméras canoniques, orientées de façon convergente (cf Figure 4.3).
Leurs axes optiques s'intersectent en un point C. Cependant, ils ne forment pas le même angle (β) par rapport au banc stéréoscopique (angles opposés). Nous repérons alors la position du point image Q sur le capteur gauche grâce à ces deux coordonnées $(\mathrm{X}_{Cl},\mathrm{Y}_{Cl})$ et par $(\mathrm{X}_{Cr},\mathrm{Y}_{Cr})$ sur le capteur droit.

Dans les paragraphes suivant, nous allons déterminer les valeurs de ces deux couples de coordonnées à partir de la position du point Q dans l'espace.

Caméra gauche D'après la figure 4.3, nous obtenons la valeur de l'abscisse X_{Cl} dans le repère associé à la caméra gauche :

$$\tan \alpha_{Cl} = X_{Cl}$$

Egalement, d'après la figure 4.3, nous avons

$$\gamma_{Cl} - \beta = \alpha_{Cl}$$

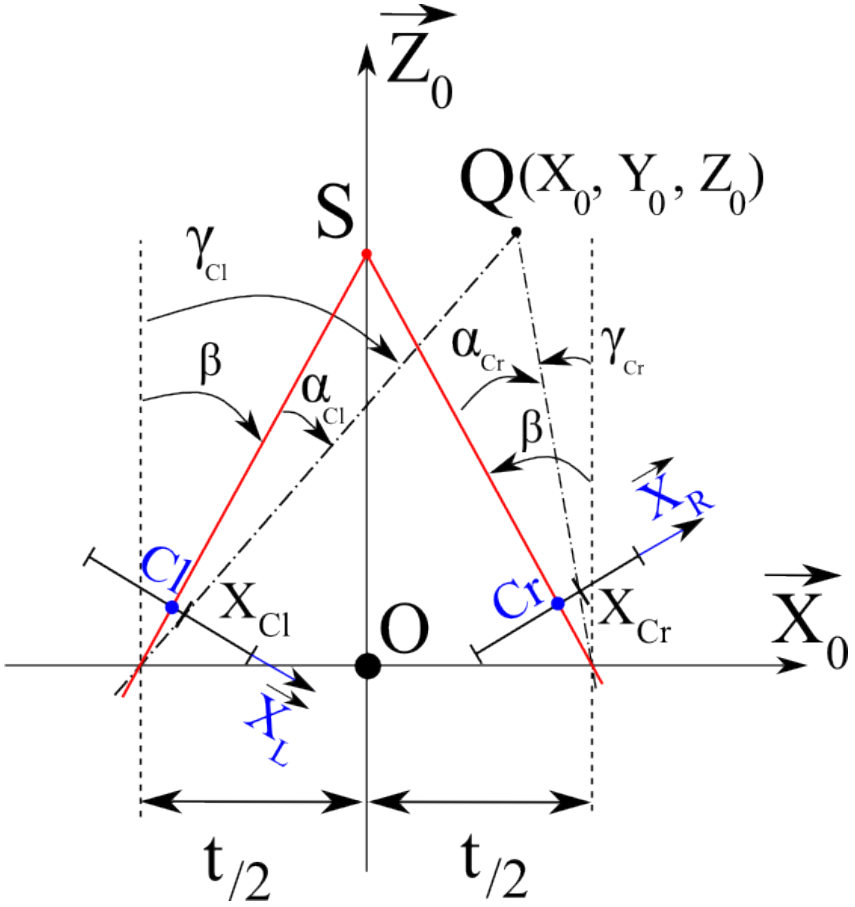


Figure 4.3 – Schéma des repères associés aux deux caméras canoniques en configuration convergente

Dans un autre triangle, nous obtenons

$$\tan \gamma_{Cl} = \frac{\frac{t}{2} + X_o}{Z_o}$$

D'où

$$X_{Cl} = \tan \alpha_{Cl} = \tan(\gamma_{Cl} - \beta)$$

Ce qui nous donne

$$\boxed{X_{Cl} = \tan\left(\arctan(\frac{\frac{t}{2} + X_o}{Z_o}) - \beta\right)} \tag{4.1}$$

Pour calculer la valeur de l'ordonnée Y_{Cl}, nous cherchons d'abord à obtenir la coordonnée de Q en Z. Or, nous avons d'après la figure 4.4 :

$Z_{1L} = Z_0 \cos\beta$ et $Z_{2L} = (X_0 + \frac{t}{2}) \sin\beta$

$Z_{Cl} = Z_{1L} + Z_{2L}$

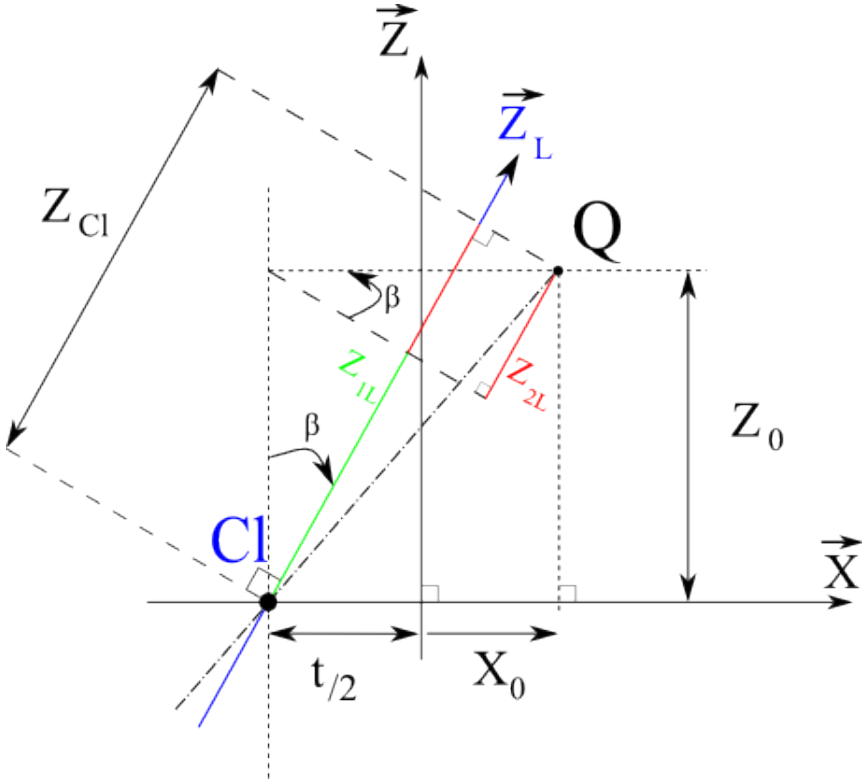


Figure 4.4 – Variables intermédiaires intervenant dans le calcul de Zcl (coordonnée en Z du point Q dans le repère gauche)

Nous obtenons donc

$$\boxed{Y_{Cl} = \frac{Y_0}{Z_{Cl}} = \frac{Y_0}{Z_0 \ \cos\beta + (X_0 + \frac{t}{2}) \ \sin\beta}} \tag{4.2}$$

Nous avons calculé les coordonnées du projeté du point Q, dans le repère lié au plan image gauche. Nous allons maintenant calculer celles du projeté du point Q dans le repère lié au plan image droit.

Caméra droite D'après la figure 4.3, et en nous basant sur les angles, nous cherchons à obtenir la valeur de l'abscisse X_{Cr} dans le repère associé à la caméra droite :

$$\tan\alpha_{Cr} = X_{Cr}$$

Egalement, d'après la figure 4.3, nous obtenons

$$\gamma_{Cr} = \alpha_{Cr} + \beta$$

Dans un autre triangle, nous pouvons aussi calculer :

$$\tan\gamma_{Cr} = \frac{X_o - \frac{t}{2}}{Z_o}$$

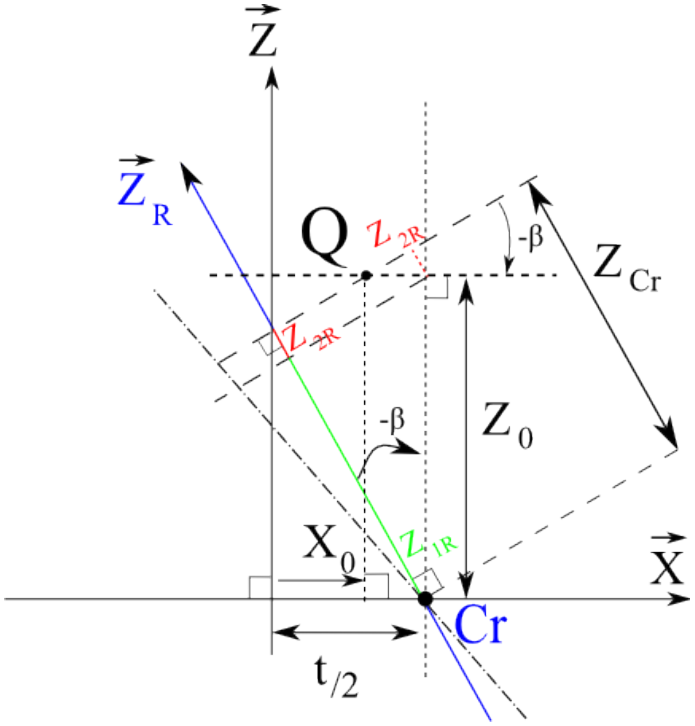


Figure 4.5 – Variables intermédiaires intervenant dans le calcul de Zcr (coordonnée en Z du point Q dans le repère droit)

D'où

$$X_{Cr} = \tan \alpha_{Cr} = \tan(-\beta + \gamma_{Cr})$$

Ce qui nous donne

$$\boxed{X_{Cr} = \tan\left(\arctan(\frac{X_o - \frac{t}{2}}{Z_o}) - \beta\right)} \tag{4.3}$$

Pour calculer la valeur de l'ordonnée Y_{Cr}, nous cherchons d'abord à obtenir la coordonnée de Q en Z. Or, nous avons d'après la figure 4.5 :

$$Z_{1R} = Z_0 \ \cos(-\beta) \text{ et } Z_{2R} = (\tfrac{t}{2} - X_0) \ \sin(-\beta) = -(X_0 - \tfrac{t}{2}) \ \sin\beta$$

$$Z_{Cr} = Z_{1R} + Z_{2R}$$

Nous obtenons donc :

$$\boxed{Y_{Cr} = \frac{Y_0}{Z_{Cr}} = \frac{Y_0}{Z_0 \ \cos\beta - (X_0 - \frac{t}{2}) \ \sin\beta}}$$

Dans cette partie, nous avons déterminé les paramètres qui lient les projections d'un point Q de l'espace sur chaque plan image des caméras gauche et droite. Les couples de coordonnées (X_{Cl},Y_{Cl}) et (X_{Cr},Y_{Cr}) seront les entrées de la modélisation de la position du point perçu, lors de l'affichage des images (cf. 4.3).

Mais avant cette étape, nous devons modéliser la manière dont les images sont projetées sur l'écran, ce que nous allons voir dans la partie suivante (cf. 4.2.1)

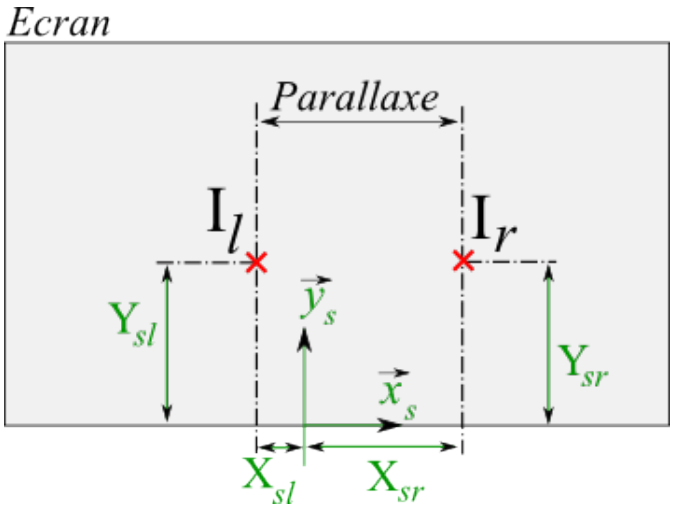


Figure 4.6 – Variables intermédiaires intervenant dans le calcul des coordonnées écran

4.2 Affichage vidéo stéréoscopique

4.2.1 Affichage des images

Le fonctionnement de l'affichage des images sur l'écran ne dépend pas de la configuration parallèle ou convergente des caméras.

En effet, nous avons déterminé la projection du point Q sur chaque plan image des caméras, quelque soit leur configuration.

Les transformations entre les repères CCD/image et image/utilisateur sont indépendantes de la configuration de capture. Ces dernières dépendent uniquement de la configuration du système de restitution. Seule la transformation espace objet/CCD est affectée par la convergence.

Repère écran La projection d'images sur un écran impose de prendre en compte la manière dont ces images sont projetées. La taille de celles-ci par rapport à la taille de l'écran est un paramètre à prendre en compte, de même que la taille de l'écran. Au final, l'effet de ces deux paramètres peut se résumer en un seul que l'on appelle le *facteur de magnification*. Il va représenter les modifications d'échelle apportées aux images, par le système de restitution.

Pour passer des coordonnées exprimées dans le repère image (coordonnées dans le plan image de chaque caméra cf. 4.1) à celles exprimées dans le repère écran (X_{Sl},Y_{Sl}) et (X_{Sr},Y_{Sr}) (cf. Figure 4.6), on multiplie ces coordonnées par le facteur M de magnification de l'affichage.

D'où : $\begin{cases} \boxed{X_{Sl} = M\ X_{Cl}} \\ \boxed{Y_{Sl} = M\ Y_{Cl}} \end{cases} \quad \begin{cases} \boxed{X_{Sr} = M\ X_{Cr}} \\ \boxed{Y_{Sr} = M\ Y_{Cr}} \end{cases}$

Nous venons de calculer les valeurs des coordonnées dans le repère écran des points images du point Q de référence. Avant de nous intéresser à la perception de la profondeur

à partir des images projetées dans la partie 3.1.3, nous allons détailler la manière dont les images sont projetées sur l'écran dans la partie suivante 4.2.3.

4.2.2 Calcul de la parallaxe

Comme nous l'avons vu précédemment, une parallaxe peut-être horizontale ou verticale. La parallaxe horizontale (resp. verticale) est une valeur de distance qui correspond à une différence horizontale (resp. verticale) entre deux points d'un couple d'images stéréoscopiques. Elle est calculée en effectuant la différence entre les positions de ces deux points homologues sur l'écran.

Dans notre cas, les parallaxes horizontale Ph et verticale Pv s'expriment donc de la manière suivante :

$$Ph = X_{S_R} - X_{S_L}$$

$$Pv = Y_{S_R} - Y_{S_L}$$

4.2.3 Ajustement de la restitution

La perception de la profondeur lors d'une projection stéréoscopique dépend de la restitution des images sur le support d'affichage.

Comme nous avons pu le voir dans les chapitres précédents, lorsque des parties des images droite et gauche se superposent parfaitement, on parle de parallaxe nulle (ces parties d'images apparaissent donc dans le plan de l'écran). La capture vidéo stéréoscopique en configuration convergente permet de maîtriser la zone de la scène qui se situera dans le plan de l'écran, grâce au point de convergence. En effet, les axes optiques des deux caméras gauche et droite se croisant en un point de l'espace (i.e. le point de convergence), ce point aura une parallaxe nulle lors de la projection des images.

Malheureusement, la rotation des caméras pour créer cette condition de parallaxe nulle, va produire des déformations trapézoïdales sur les images et par conséquent une parallaxe verticale lors de la projection qui est à l'origine d'un important inconfort visuel [Allison 04] [Woods 93]. Cette solution est actuellement utilisée pour les tournages de cinéma numérique, où la possibilité de corrections de ces déformations trapézoïdales en post-production est offerte et exploitée. Dans le cas de la capture et de la restitution temps réel des images, qui constitue le cadre de notre recherche, ce genre de correction n'est pas envisageable. C'est la raison pour laquelle, des caméras en configuration parallèle sont utilisées.

Afin de créer un plan à parallaxe nulle, avec les images issues d'une capture en configuration parallèle, deux solutions sont envisageables.

La première consiste à décaler le centre des lentilles de manière à recréer un convergence des caméras, tout en maintenant leurs plans images, parallèles entre eux ainsi qu'au support de fixation (cf. Figure 4.7). Cela présente l'avantage de ne pas générer de déformations trapézoïdales mais nécessite par contre de manipuler les composants internes des caméras.

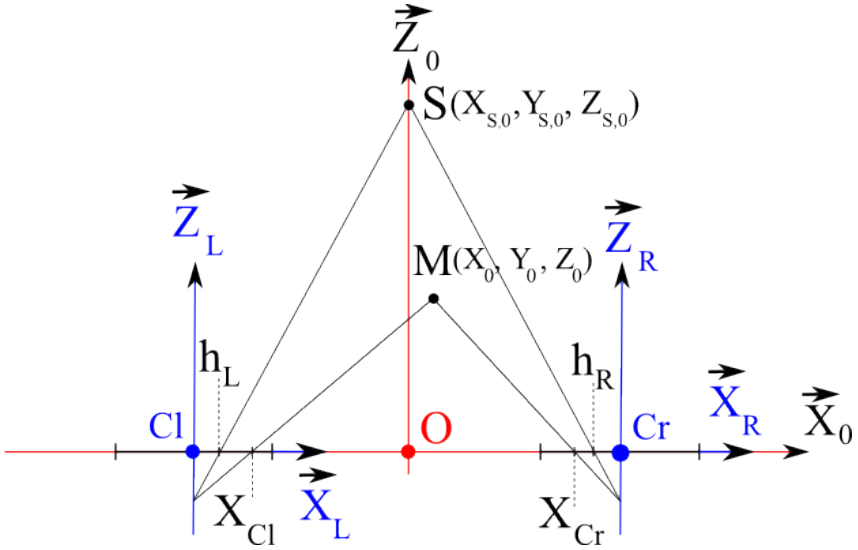


Figure 4.7 – Déplacement des centres des lentilles d'une valeur h pour obtenir un plan à parallaxe nulle passant par le point C_0 avec une configuration parallèle

La seconde solution consiste à modifier les images de manière à générer cette parallaxe nulle. Il s'agit de la solution que nous avons préférée pour nous permettre d'utiliser le plus grand nombre de caméras possible, sans avoir à les modifier physiquement. La solution consiste à déplacer les images horizontalement de manière à modifier la position du plan à parallaxe nulle [Stavrakis 08] [Lipton 97] [McVeigh 96].

Dans la figure 4.8, on obtient des valeurs de décalage pour les images gauche (h_L) et droite (h_R) :

$$h_L = \frac{t}{2\ Z_{S,0}}$$

$$h_R = -\frac{t}{2\ Z_{S,0}}$$

Figure 4.8 – Vues gauche et droite de la scène

Nous allons illustrer cela à partir d'un exemple. Visuellement, à partir d'un couple d'images stéréo superposées l'une sur l'autre (cf. Figure 4.8), nous sommes à même de

déterminer où va se trouver le plan à parallaxe nulle. Il s'agit (si elle existe) de la zone où l'on ne perçoit qu'une seule image. Dans le cas où elle n'existe pas, ou bien si l'on souhaite la modifier, le décalage horizontal va permettre de fixer le plan à parallaxe nulle (cf. Figure 4.9). Des bandes verticales vont alors apparaître à droite et à gauche des images, et correspondent aux pertes dues au déplacement latéral des vues. Le seul inconvénient de cette méthode est l'apparition de ces bandes verticales qui vont devoir être supprimées par un traitement (correspondant à un recadrage de l'image), mais cela ne concernera qu'une faible partie de l'image et sera facilement réalisable en temps réel.

Figure 4.9 – Images droite et gauche superposées

Figure 4.10 – Images droites et gauche superposées mais décalées pour créer le plan à parallaxe nulle : Ici le cube rose et les objets dans le même plan sont à parallaxe nulle.

De manière générale, afin de limiter le placement de tous les objets de la scène devant ou derrière l'écran, on préfère que le plan à parallaxe nulle soit situé au milieu de l'objet ou de la scène que l'on souhaite représenter en 3D. De plus, il est nécessaire de maintenir une parallaxe faible, afin que la restitution reste agréable pour l'accommodation et la convergence (cf. section 3.1.3).

4.3 Perception de la profondeur

Dans cette partie, nous allons nous intéresser à la modélisation de la perception de la profondeur, par projection d'images sur un écran plan faisant face à l'utilisateur.

Nous nous limitons, dans cette partie, à l'étude géométrique de la perception de la profondeur, et non aux illusions d'optique.

Nous prenons comme hypothèse de travail que l'observateur regarde l'écran, droit devant lui. Il est placé de manière à ce que la ligne reliant ses yeux soit parallèle à l'écran et parfaitement horizontale. Les mouvements de l'utilisateur seront étudiés dans le section 6.2.

Nous partons de deux points écran I_l et I_r qui correspondent aux projections du point Q de la scène, dans chacune des deux images gauche et droite (respectivement). Nous

allons donc calculer les coordonnées d'un point M (Xi Yi Zi), perçu dans le repère de l'observateur F_i(I,$\vec{x_i}$,$\vec{y_i}$,$\vec{z_i}$). La position de ce point M dépend de l'écart interoculaire, ainsi que de la position des points image sur l'écran. Elle peut être déterminée géométriquement, comme l'intersection de deux plans verticaux. Le premier plan (le plan gauche) va contenir la position de l'oeil gauche et le point image gauche I_l. Le second plan (le plan droit) contient, lui, la position de l'oeil droit et le point image droit I_r. I_l et I_r sont exprimés dans le repère écran F_s(S,$\vec{x_s}$,$\vec{y_s}$,$\vec{z_s}$).

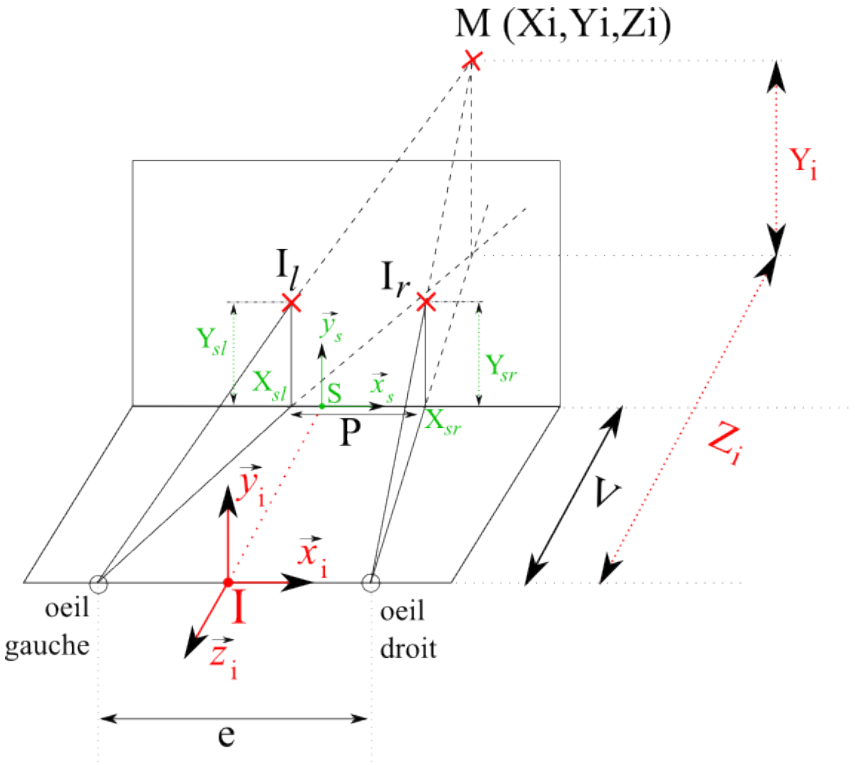


Figure 4.11 – Variables intermédiaires intervenant dans le calcul de la position perçue du point M(X_i,Y_i,Z_i) par un utilisateur ayant un écart interoculaire e et placé à une distance V de l'écran, à partir de deux points images (I_l et I_r) ayant une parallaxe P_h.

Calcul de la composante (Z_i) du point perçu

La valeur Z_i (qui correspond à la profondeur perçue du point M) dépend de l'écart interoculaire, de la distance de l'utilisateur à l'écran, ainsi que de la parallaxe horizontale (P_h) affichée. Dans la Figure 4.11, en appliquant le théorème de Thalès, nous trouvons :

$$\frac{Z_i - V}{Z_i} = \frac{P_h}{e} \quad \text{d'où :} \quad \boxed{Z_i = \frac{V\,e}{e - P_h}}$$

Calcul de la composante (X_i) du point perçu

Afin de calculer l'abscisse du point M, nous posons :

$$\begin{cases} \frac{\frac{e}{2} - X_{Sr}}{\frac{e}{2} - X_i} = \frac{V}{Z_i} \\ \frac{\frac{e}{2} + X_{Sl}}{\frac{e}{2} + X_i} = \frac{V}{Z_i} \end{cases} \Leftrightarrow \begin{cases} \frac{e}{2} - X_{Sr} = \frac{V}{Z_i} \left(\frac{e}{2} - X_i \right) & (1) \\ \frac{e}{2} + X_{Sl} = \frac{V}{Z_i} \left(\frac{e}{2} + X_i \right) & (2) \end{cases}$$

Par soustraction sur (1) et (2), on obtient $X_{Sr} + X_{Sl} = \frac{V}{Z_i} (2\, X_i)$

$$\Leftrightarrow \frac{X_{Sr} + X_{Sl}}{2} = \frac{X_i}{Z_i} V$$

En utilisant la valeur de $Z_i = \frac{V\, e}{e - P_h}$ précédemment calculée, nous obtenons :

$$\Leftrightarrow \frac{X_{Sr} + X_{Sl}}{2} = \frac{e - P_h}{e} X_i$$

$$\Leftrightarrow \boxed{X_i = \frac{e\,(X_{Sr} + X_{Sl})}{2\,(e - P_h)}} \tag{4.4}$$

La valeur de l'abscisse M perçue, dépend donc des abscisses des positions des images ainsi que de l'écart interoculaire, et de la parallaxe des images.

Calcul de la composante (Y_i) du point perçu

Nous allons maintenant calculer l'ordonnée Y_i du point M. Dans la Figure 4.11, nous pouvons déterminer les relations suivantes :

$$\begin{cases} \frac{Y_i}{Y_{Sr}} = \frac{Z_i}{V} \\ \frac{Y_i}{Y_{Sl}} = \frac{Z_i}{V} \end{cases} \Leftrightarrow \begin{cases} Y_{Sr} = \frac{Y_i\, V}{Z_i} & (3) \\ Y_{Sl} = \frac{Y_i\, V}{Z_i} & (4) \end{cases}$$

Nous pouvons donc écrire les équations (3) et (4) sous la forme

$$\begin{cases} Y_{Sr} = Y_i \frac{e - P_h}{e} & (5) \\ Y_{Sl} = Y_i \frac{e - P_h}{e} & (6) \end{cases}$$

Par addition des membres (5) et (6), nous obtenons :

$$\boxed{Y_i = \frac{e\,(Y_{Sr} + Y_{Sl})}{2\,(e - P_h)}} \tag{4.5}$$

La valeur de l'ordonnée de M perçue dépend donc des ordonnées des positions des images ainsi que de l'écart interoculaire, et de la parallaxe des images.

4.4 Synthèse de la transmission stéréoscopique

La transmission stéréoscopique correspond à la chaîne entière de restitution des images, de la capture à la perception en passant par l'affichage. Modéliser cette transmission revient à effectuer la synthèse des étapes précédentes.

4.4.1 Cas de la configuration caméras parallèles

Nous savons, d'après les résultats du 4.2.1, que les coordonnées d'un point sur l'écran peuvent être exprimées à partir des coordonnées de ce même point dans les repères caméra, d'où $X_{Sr} = M\ X_{Cr}$, $X_{Sl} = M\ X_{Cl}$ et $Y_{Sr} = M * Y_{Cr} = Y_{Sl} = M * Y_{Cl}$.

De plus, d'après 4.3, nous sommes en mesure d'exprimer les coordonnées perçues d'un point par rapport à ces coordonnées écran.

Abscisse du point M en fonction des coordonnées du point Q

L'abscisse du point M a pour valeur $X_i = \dfrac{e\ M\ (X_{Cr} + X_{Cl})}{2(e - M(X_{Cr} - X_{Cl}))}$

D'où, $X_i = \dfrac{e\ M(X_{Cr} + X_{Cl})}{2\ (e - P_h)} \Leftrightarrow X_i = \dfrac{e\ M\ X_0}{Z_0\ (e - P_h)}$

or ici P_h représente la parallaxe horizontale et a pour valeur :

$$P_h = X_{Sr} - X_{Sl} = M\ (X_{Cr} - X_{Cl})$$

Nous obtenons comme valeur pour l'abscisse du point M :

$$\boxed{X_i = \frac{e\ M\ X_0}{Z_0\ e - M\ t}} \tag{4.6}$$

Ordonnée du point M en fonction des coordonnées du point Q

Si l'on se place dans le cas où il n'y a pas de parallaxe verticale, l'ordonnée du point M a pour valeur :

$$Y_i = Y_S\ \frac{e}{e - P}$$

D'après la partie 4.2.1, nous pouvons écrire :

$$Y_{Sr} = M\ Y_{Cr} = Y_{Sl} = M\ Y_{Cl} = Y_S$$

D'où $Y_i = \dfrac{e\ M\ Y_0}{Z_0\ (e - P)}$

En exprimant P de la même manière que précédemment pour le calcul de l'abscisse, nous obtenons comme valeur pour l'ordonnée du point M :

$$\boxed{Y_i = \frac{e\ M\ Y_0}{Z_0\ e - M\ t}} \tag{4.7}$$

Profondeur du point M en fonction des coordonnées du point Q

Rappel : que ce soit pour le cas parallèle ou bien le cas convergent, la profondeur du point perçu ne dépend que de l'affichage des images et à pour valeur d'après le 4.3 :

$$Z_i = \frac{V\ e}{e - P}$$

Nous obtenons donc comme valeur pour la coordonnée de profondeur du point M :

$$\boxed{Z_i = \frac{e\ V\ Z_0}{Z_0\ e - M\ t}} \tag{4.8}$$

4.4.2 Cas de la configuration caméras convergentes

Nous choisissons comme hypothèse que l'observateur regarde l'écran, droit devant lui.

Abscisse du point M en fonction des coordonnées du point Q

L'abscisse du point M a pour valeur $X_i = \frac{e\ M\ (X_{Cr} + X_{Cl})}{2(e - M(X_{Cr} - X_{Cl}))}$

D'après les résultats obtenus en 4.1.2, nous avons :

$$X_{Cr} = \tan\left(\arctan(\frac{X_o - \frac{t}{2}}{Z_o}) - \beta\right) \quad \text{et} \quad X_{Cl} = \tan\left(\arctan(\frac{\frac{t}{2} + X_o}{Z_o}) - \beta\right)$$

Nous obtenons par substitution la valeur pour l'abscisse du point M :

$$\boxed{X_i = \frac{e\ M\ \left(\tan\left(\arctan(\frac{X_o - \frac{t}{2}}{Z_o}) - \beta\right) + \tan\left(\arctan(\frac{\frac{t}{2} + X_o}{Z_o}) - \beta\right)\right)}{2\left(e - M\left(\tan\left(\arctan(\frac{X_o - \frac{t}{2}}{Z_o}) - \beta\right) - \tan\left(\arctan(\frac{\frac{t}{2} + X_o}{Z_o}) - \beta\right)\right)\right)}}$$

Ordonnée du point M en fonction des coordonnées du point Q

D'après les résultats obtenus en 4.1.2, nous avons :

$$Y_{Cr} = \frac{Y_0}{Z_0\ \cos\beta - (X_0 - t/2)\ \sin\beta} \quad \text{et} \quad Y_{Cl} = \frac{Y_0}{Z_0\ \cos\beta + (X_0 + t/2)\ \sin\beta}$$

De la même manière, avec $Y_i = \frac{e\ (Y_{Sr} + Y_{Sl})}{2\ (e - P)}$, nous obtenons comme valeur pour l'ordonnée du point M :

$$\boxed{Y_i = \frac{e\ (Y_{Sr} + Y_{Sl})}{2\left(e - M\left(\tan\left(\arctan(\frac{X_o - \frac{t}{2}}{Z_o}) - \beta\right) - \tan\left(\arctan(\frac{\frac{t}{2} + X_o}{Z_o}) - \beta\right)\right)\right)}}$$

Profondeur du point M en fonction des coordonnées du point Q

La coordonnée de profondeur du point M a pour valeur :

$$Z_i = \frac{e\,V}{2\left(e - M\left(\tan\left(\arctan(\frac{X_o - \frac{t}{2}}{Z_o}) - \beta\right) - \tan\left(\arctan(\frac{\frac{t}{2} + X_o}{Z_o}) - \beta\right)\right)\right)}$$

La chaîne de transmission stéréoscopique (de la capture à la profondeur perçue) est à présent entièrement modélisée. Nous sommes donc en mesure de déduire la profondeur perçue d'un point 3D de la scène, à partir des paramètres de capture, d'affichage des images et du placement de l'utilisateur devant le système d'affichage.

Pour cette modélisation, nous avons choisi de séparer les effets du matériel de capture/restitution, de ceux créés par la transmission stéréoscopique. De cette manière, nous avons volontairement considéré que les images étaient non déformées lors de la capture.

Comme cette hypothèse de travail n'est que très rarement atteignable dans la réalité sans retravailler les images (en cause les imperfections des caméras), nous détaillons dans la suite, la méthode que nous avons mise en oeuvre afin de corriger les déformations sur les images capturées.

Lors de nos expérimentations, afin d'obtenir une restitution maîtrisée de vidéos stéréoscopiques en temps réel sur des systèmes de Réalité Virtuelle, nous avons effectué les choix suivants :

- utilisation de caméras stéréoscopiques en configuration parallèle, afin de ne pas avoir de parallaxe verticale à corriger. La parallaxe nulle sera obtenue en modifiant les images (déplacement horizontal)
- limitation du mouvement des caméras en adoptant une position fixe sur leur support.

Chapitre 5

Identification des paramètres critiques pour la restitution d'images vidéo stéréoscopiques

Sommaire

Dans le chapitre précédent, nous avons traité le problème de la modélisation de la transmission stéréoscopique à partir d'images vidéo non déformées. Cependant, les caméras industrielles du marché introduisent bien souvent quelques défauts, qui altèrent les images en les déformant. Sans correction de ces déformations, l'utilisation de ces images déformées pose des problèmes de perception des formes géométriques des objets de la scène.

La méthode que nous avons mise en place pour rectifier, en temps réel, les images en haute définition issues d'un banc stéréoscopique est détaillée dans la section 5.1. Cette section reprend en les complétant, les résultats que nous avons publiés dans *High-resolution stereo video rectification through a cost-effcient real-time GPU implementation using intrinsic and extrinsic camera parameters* [Goslin 09].

La transformation entre l'image déformée et l'image rectifiée est très coûteuse en temps de calcul, principalement car le calcul de la rectification est une transformation non linéaire qu'il faut appliquer à chaque pixel.

De cette façon, la plupart des approches de rectification implémentées sur CPU (Central Processing Unit) sont loin d'être temps-réel. Lorsque le temps réel est requis, les algorithmes de rectification sur CPU trouvés dans la littérature se contentent de travailler sur certaines parties de l'image [Scharstein 02].

Mais l'évolution de la puissance de calcul fournie par les processeurs des cartes graphiques a motivé l'adaptation des algorithmes de rectification sur ce type de matériel.

De nombreux algorithmes de reconstruction 3D à partir d'images stéréoscopiques ont alors été portés sur GPU (Graphics Processing Unit) [Prehn 07] [Yang 06]. Dans [Yang 06] [Mairal 06] [Johnson 06] [Mairal 05] différentes techniques d'appariement sont comparées. On y mentionne que les images sont stéréo-rectifiées pour obtenir des lignes épipolaires alignées et des images non déformées. Cependant, aucun détail de code concernant la rectification n'est fourni dans ces travaux. [Mairal 06] atteint une nombre d'images par seconde affichées après application de la rectification aux alentours de 9 fps, pour un flux vidéo mono standard.

Quelques détails du principe de l'implémentation d'une rectification d'image monoscopique sont présentés dans le volume n°2 des GPUGEMS [Pharr 05], mais aucun n'est donné concernant la rectification d'images stéréoscopiques.

Nous nous sommes donc intéressés aux performances qu'offrent les processeurs des cartes graphiques actuelles en termes de calcul haute performance. Comme les calculs de rectification sont très répétitifs (appliqués à chaque pixel), ils peuvent être parallélisés massivement, ce que réalisent nativement les cartes graphiques récentes. Cette approche, inspirée par les travaux qui se font dans le domaine de la vision par ordinateur, est très innovante dans le domaine de la projection de vidéos stéréoscopiques en relief.

La valeur de la parallaxe affichée des images stéréoscopiques est une autre source importante de gène pour l'utilisateur d'un système de Réalité Virtuelle. Pour adapter au mieux cet affichage, il est important de connaître précisément la valeur de l'écart interoculaire de ce dernier.

Nous présentons, dans la section 5.2, notre méthode de calcul de cette distance interoculaire, basée sur la détermination de la position de chaque oeil, dans un repère associé à la tête de l'utilisateur que l'on calibre.

5.1 Paramètres de rectification temps réel d'images vidéo stéréoscopiques

Une caméra vidéo crée une vue 2D à partir d'une scène 3D. Ceci est réalisé par une projection perspective (cf. section 3.2.1). Outre les distorsions de perspective, les composants physiques qui constituent la caméra (optiques et capteur CCD en particulier) causent des déformations supplémentaires dans l'image. Les images issues d'une capture avec un système de caméras vidéo stéréoscopiques souffrent généralement de déformations tangentielle et radiale.

La modélisation de la capture d'images et celle de la transmission stéréoscopique ont été modélisées séparément, pour permettre de corriger ces distorsions, sans modifier la modélisation de la capture/affichage des images.

Les algorithmes d'appariement de points d'intérêt discrets [Bovic 05] ou d'appariement dense de points à partir d'images stéréo [Scharstein 02] sont facilités si les lignes épipolaires sont parallèles et à la même hauteur dans l'image. A partir d'un banc stéréoscopique canonique, une éventuelle recherche 2D sur l'image entière peut-être ramenée à une recherche 1D le long de la ligne épipolaire correspondante, ce qui offre un gain de temps considérable.

Dans cette section, nous allons détailler l'approche que nous avons développée pour rectifier les images stéréoscopiques lors de l'acquisition, sur GPU et en temps réel. Cette rectification est l'étape préalable indispensable si l'on souhaite utiliser ces images et en retirer des informations précises (sur la profondeur relative des différents objets de la scène, par exemple). En effet, une distorsion radiale de trois pixels par exemple, suffit à biaiser une parallaxe perçue. Nous commencerons par la rectification d'une caméra simple, et ensuite nous verrons la rectification d'un banc stéréoscopique.

5.1.1 Rectification en temps rééel

Notre méthode de rectification d'images s'appuie sur les travaux de [Bouguet 04] et [Fusiello 00]. Mais contrairement à ces travaux, nous raisonnons sur des configurations de caméras canoniques.

Dès lors, nous n'avons pas besoin de calculer un ensemble de paramètres intrinsèques communs à chacune des caméras.

De plus, tous les changements de coordonnées sont pré-calculés et appliqués à chaque pixel, de manière que la manipulation de pixels (interpolation etc), qui est une source possible de détérioration de l'image, soit réduite au maximum.

5.1.1.a Rectification mono caméra

La méthode de rectification que nous avons développée consiste à compenser les déformations (dues aux optiques et au capteur) en redressant les images à l'aide de valeurs calculées de déformations (cf. Figure 5.1).

Au préalable, nous avons effectué une calibration de la caméra vidéo grâce à la *matlab calibration toolbox* [Bouguet 04], qui nous permet de connaître les valeurs des paramètres de calibration.

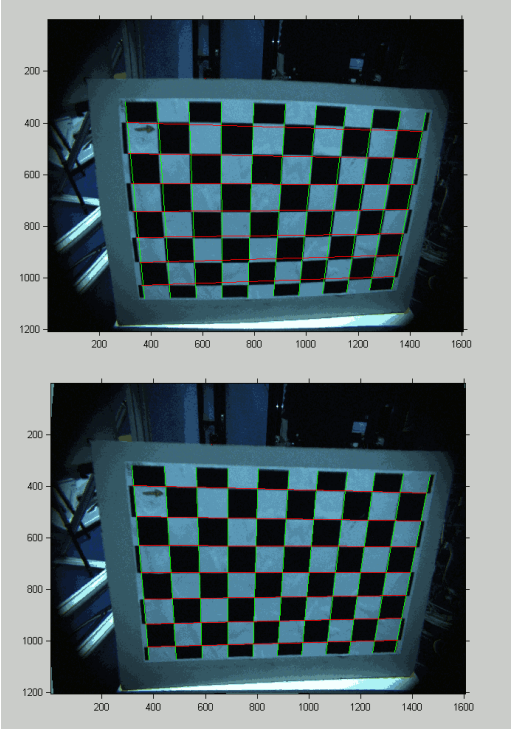


Figure 5.1 – Rectification d'une seule image : l'image du haut est l'image brute (non rectifiée). On peut constater les déformations radiales. L'image du bas est le résultat après application de notre algorithme (image rectifiée).

Ensuite, nous pré-calculons une fois pour toutes une *texture de déplacement* qui va contenir, pour chaque pixel de l'image originale, les déplacements à effectuer sur le pixel, afin de compenser les déformations subies.

Enfin, nous utilisons cette *texture de déplacement* pour rectifier en temps réel les images issues de la caméra vidéo.

Notre méthode pour générer cette texture de déplacement est équivalente à suivre, dans le sens inverse, le cheminement décrit section 3.2, i.e. $\underline{p} \rightarrow \underline{p}_1 \rightarrow \underline{m}^d \rightarrow \underline{m}^n$

Nous partons d'une image déformée, prise par une caméra vidéo. Pour passer d'un point $\underline{p}$ dans l'espace image au point $\underline{m}^d$, nous devons inverser le produit matriciel $L\ K$ pour chaque point de l'image d'origine.

Ensuite, la seconde étape consiste à passer du point $\underline{m}^d$ au point $\underline{m}^n$. Pour cela, il faut inverser l'expression polynomiale $\underline{m}^d = (1 + K_1\, r^2 + K_2\, r^4 + K_5\, r^6)\ \underline{m}^n$ ce qui revient à calculer $\underline{m}^n$ pour un point $\underline{m}^d$ donné, et $r^2 = (x^n)^2 + (y^n)^2$.

Cette inversion n'offre pas de solution unique. Cependant, parmi les quelques solutions trouvées, il n'y en a qu'une seule qui soit physiquement réaliste. Pour déterminer la solution qui nous intéresse nous utilisons la méthode proposée dans [Heikkila 97], qui préconise de résoudre ce problème d'inversion en suivant un processus d'itérations simples, en utilisant un bon point de départ.

Enfin, une fois qu'une solution après inversion a été retenue, il est très probable que la valeur de $\underline{m}^n$ obtenue ne corresponde pas à une coordonnée pixelique entière de la matrice du capteur. Il va donc être nécessaire d'interpoler cette valeur aux pixels adjacents.

5.1.1.b Rectification du banc stéréoscopique

Le but de la rectification d'une paire d'images stéréoscopiques est d'aligner des lignes épi-polaires correspondantes dans les images gauche et droite. Les projections d'un point 3D sur les images gauche et droite se trouvent sur une même ligne horizontale. L'action de rectification peut être décrite mathématiquement [Fusiello 00] par un changement de coordonnées images, de la configuration générale stéréo (indicée o en dessous) à la configuration canonique (indicée n). La projection perspective pour chaque caméra du banc stéréoscopique est décrite par l'équation (5.1).

$$\lambda_\diamond m_\diamond = P_\diamond X^o \text{ avec } P_\diamond = \left[R_\diamond \mid t_\diamond\right] \tag{5.1}$$

où R,t sont les paramètres extrinsèques de chaque caméra $\diamond \in \{Lo, Ro, Ln, Rn\}$. Ici, nous supposons qu'une rectification mono-camera (telle que décrite dans la section 5.1.1.a) a déjà été réalisée. La transformation des coordonnées image de la configuration générale à la configuration canonique est donc réalisée par :

$$\begin{aligned} \lambda_R m_{Ro} &= Q_R m_{Rn}, & \text{où } Q_R &:= R_{Ro} R_{Rn}^T \text{ et } \lambda_R := \frac{\lambda_{Ro}}{\lambda_{Rn}} \\ \lambda_L m_{Lo} &= Q_L m_{Ln}, & \text{où } Q_L &:= R_{Lo} R_{Ln}^T \text{ et } \lambda_L := \frac{\lambda_{Lo}}{\lambda_{Ln}} \end{aligned}$$

Les paramètres extrinsèques sont généralement donnés sous la forme d'une position relative de la caméra gauche par rapport à la caméra droite t_L^R et d'une orientation R_L^R. On pose alors $R_{Ro} := I_3$ et de $R_{Lo} := R_L^R$. Les orientations des caméras droite R_{Rn} et gauche R_{Ln} peuvent être déterminées grâce aux contraintes suivantes [Fusiello 00] :

- les plans image ont la même orientation
- un axe de l'image est choisi de tel manière qu'il soit parallèle à la *ligne de base* du banc stéréoscopique
- les centres optiques C_L, C_R ne changent pas lors de la procédure de rectification.

De cette façon, $R_{Rn} = R_{Ln} = [\underline{r}_1\ \underline{r}_2\ \underline{r}_3]^T$ sont calculés avec

$$\underline{r}_1 = \frac{C_R - C_L}{||C_R - C_L||} \qquad \underline{r}_2 = \frac{\underline{e}_{z,o} \times r_1}{||\underline{e}_{z,o} \times \underline{r}_1||} \qquad \underline{r}_3 = \underline{r}_1 \times \underline{r}_2 \tag{5.2}$$

où la direction choisie de l'axe optique est égale à la moyenne des anciennes directions $\underline{e}_{z,o} = \frac{1}{2}\big(r_{3,Ro} + r_{3,Lo}\big)$.

Donc, si l'on souhaite étendre le processus de rectification d'une image (mono) à celui de la rectification stéréo, la chaîne de transformations suivante $\underline{p} \rightarrow \underline{p}_1 \rightarrow \underline{m}^d \rightarrow \underline{m}^n$ devient $\underline{p} \rightarrow \underline{p}_1 \rightarrow \underline{m}^d \rightarrow \underline{m}_o^n \rightarrow \underline{m}_n^n$.

5.1.2 Résultats expérimentaux

Notre méthode de rectification des images vidéo stéréoscopiques est réalisée en trois temps. Les deux premières étapes sont des étapes réalisées "hors-ligne" et permettent de créer un tableau de correspondance. Seule la troisième est effectuée en temps réel et utilisera ce tableau de correspondance pré-calculé. La première étape consiste en une

calibration des caméras droite et gauche indépendamment, puis du banc stéréoscopique. La seconde étape consiste à pré-calculer une fois pour toutes sur le processeur principal (CPU) les tableaux de correspondance et les coder sous forme de textures de déplacement. Ces textures seront ensuite utilisées pour rectifier, en temps réel, dans la troisième étape (qui sera effectuée par le processeur de la carte graphique (GPU)) l'image vidéo brute (déformée) en une image non déformée.

5.1.2.a Calibration des caméras hors ligne

La calibration des caméras s'effectue d'abord caméra par caméra, puis on calibre le banc stéréoscopique, le tout en utilisant la *matlab calibration toolbox* [Bouguet 04]. Tout d'abord, il va nous falloir prendre des images d'une mire de calibration avec chacune des deux caméras. Pour obtenir une calibration robuste, il nous faudra au moins capturer huit positions différentes de la mire (cf. Figure 5.2). Ensuite, nous allons repérer à la main chaque coin de la mire dans chacune des images capturées (cf. Figure 5.3). Enfin, l'utilisation de la *matlab calibration toolbox*, nous permettra d'obtenir les paramètres de calibration de chaque caméra, et du banc stéréoscopique. Cette étape est à réaliser une fois pour chaque caméra. Les paramètres de calibration restant valables tant qu'on ne change pas les réglages de la caméra ou les optiques. La rapidité de cette étape n'était pas notre priorité, étant donné qu'elle est réalisée hors-ligne.

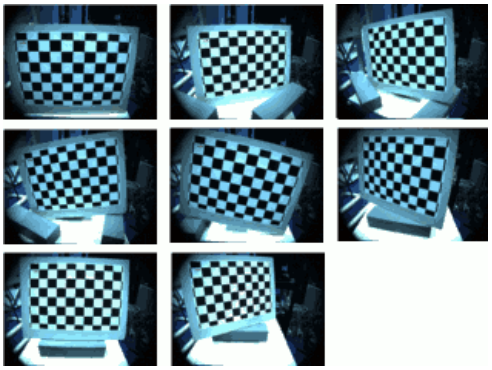


Figure 5.2 – Capture de huit positions différentes de la mire de calibration

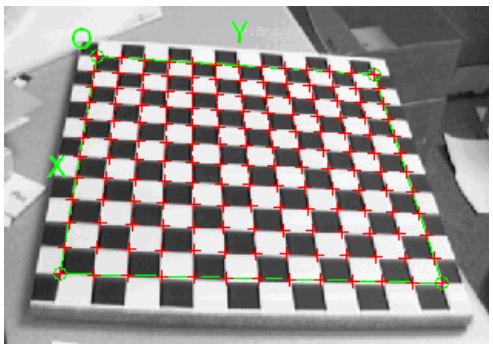


Figure 5.3 – Repérage des quatre coins de la mire dans chacune des images capturées

5.1.2.b Etape de pré-calcul des tableaux de correspondance

Cette étape consiste à pré-calculer une fois pour toutes les tableaux de correspondance et les coder sous forme de textures de déplacement. Pour chaque caméra, ces calculs se basent sur une image brute (déformée) issue de la capture. Ils se déroulent de la façon suivante. Pour chaque pixel de l'image brute, nous allons calculer sa valeur de déplacement (i.e. sa position corrigée après rectification), puis nous allons stocker cette valeur un tableau de correspondance. Ce tableau de correspondance va ensuite être encodé dans un fichier de texture, pour qu'il soit exploitable facilement par le GPU.

Cas mono-caméra

Afin de garantir un remplissage le plus important possible de l'image canonique, nous devons d'abord établir la position de l'image brute dans l'espace canonique. Cela permet de minimiser les détériorations que subit l'image et garantit une taille de pixel constante.

Le calcul du tableau de correspondance est basé sur les coordonnées de l'image rectifiée, ce qui permet un calcul direct des déformations non-linéaires.

La première étape consiste à localiser les positions des quatre coins de l'image brute dans l'espace canonique. Ceci est réalisé en résolvant le problème de l'inversion des termes non-linéaires par itération selon la méthode décrite dans [Heikkila 97] (cf. étape 1 de la Figure 5.4).

La seconde étape est effectuée à partir de l'image canonique, à partir de laquelle nous allons calculer la position de chaque pixel de l'image canonique dans l'image brute après déformation et interpolation entre les pixels adjacents (cf. étape 2 de la Figure 5.4).

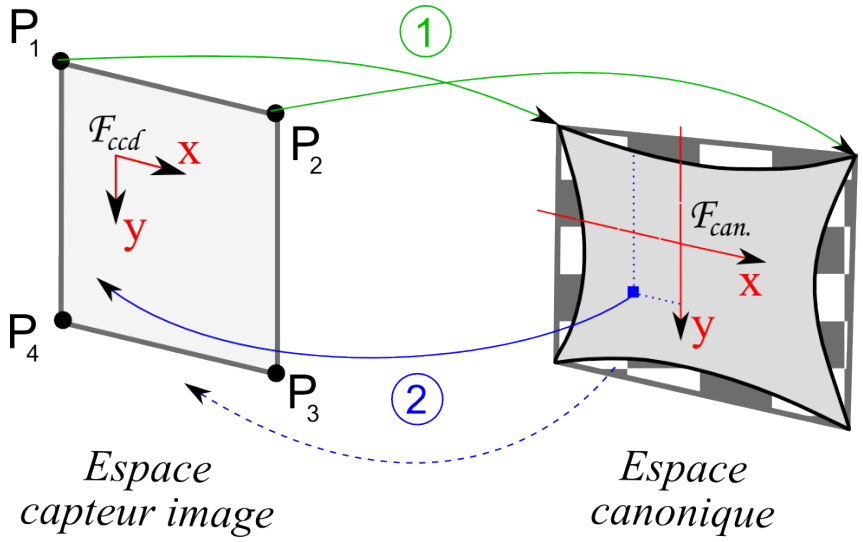


Figure 5.4 – Détermination la position de l'image brute dans l'espace canonique : la première étape consiste à localiser les positions des quatre coins de l'image brute dans l'espace canonique. La seconde étape consiste à calculer la position de chaque pixel de l'image canonique dans l'image brute après déformation et interpolation entre les pixels adjacents.

Les valeurs de déformation sont obtenues après le calcul de la déformation inverse (radiale et tangentielle). Pour chaque pixel du tableau de correspondance, on calcule la position du point rectifié, en se basant sur les valeurs de ses coordonnées, et sur les valeurs des coefficients de déformation (obtenus à l'issue de la calibration de chaque caméra).

Cas banc stéréoscopique

Dans le cas du banc stéréoscopique, les caméras réelles ne sont pas parfaitement parallèles. La calibration stéréoscopique permet de conserver une distance constante de la ligne de base, ainsi qu'une position des centres optiques des caméras constante. Cela permet également de modifier les orientations des deux caméras de sorte que leurs plans images deviennent alignés et centrés sur un même axe horizontal.

Par conséquent, le calcul des tableaux de correspondance pour un banc stéréoscopique, revient à réaliser deux calculs de tableau de correspondance pour chaque caméra indépendamment. Puis, on compense les changements d'orientation des caméras canoniques, en corrigeant les orientations dans les tableaux de correspondance.

Stockage des valeurs des tableaux de correspondance

Afin de faciliter le traitement par les processeurs graphiques, nous avons choisi de coder les valeurs des tableaux de correspondance au sein de fichiers texture (auxquels nous nous référerons par la suite sous le nom de *texture de déformation*). Ces textures ne contiennent donc aucune information visuellement exploitable (cf. Figure 5.5).

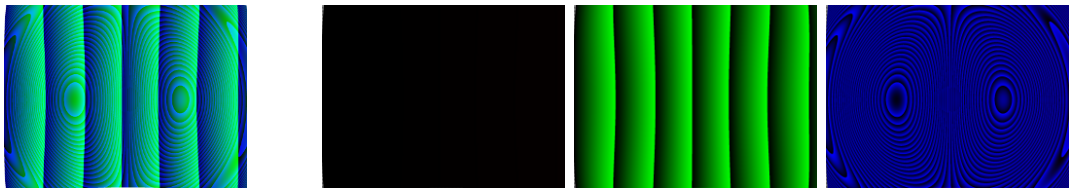


Figure 5.5 – Décomposition d'une texture de déformation (tout à gauche) selon ses trois composantes R (2ème à gauche), G (2ème à droite), et B (tout à droite)

La texture de déformation va ensuite être utilisée pour rectifier l'image brute (déformée). Pour chaque pixel de cette dernière, on interroge la texture de déformation pour obtenir la valeur de déplacement absolu correspondante. Ensuite, la couleur du pixel interpolé est écrite dans l'image rectifiée.

5.1.2.c Rectification temps réel sur GPU

Le calcul massivement parallèle (supporté en natif par les cartes vidéo de dernière génération) permet d'accélérer significativement la rectification des images. Nous allons fournir au GPU une texture à rectifier ainsi que la table de correspondance sous forme de texture, et il sera en mesure de calculer les valeurs de déformations sur plusieurs pixels en même temps.

Un programme d'ombrage (*shader*) a été détourné de son but originel pour réaliser ces calculs.

Comme nous venons de le voir dans la section précédente, la valeur de la déformation est codée en couleur dans la texture de déformation. Notre *shader* va donc récupérer une couleur de l'image brute, couplée à une valeur de déformation, et écrire cette couleur aux coordonnées qu'on lui fournit suite à la rectification.

Le principe général de la rectification de l'image brute est le suivant. Nous allons remplir une image canonique (vide au départ). Afin de créer l'image finale (non déformée), il sera nécessaire pour chaque pixel de l'image de :

- récupérer la valeur de déplacement correspondante à partir de la texture de déformation pré-calculée. Il s'agit de la première entrée du *shader*
- récupérer la couleur de l'image brute aux nouvelles coordonnées qui tiennent compte du déplacement correctif 2D dans la texture. Il s'agit de la deuxième entrée de notre *shader*)

– appliquer la couleur récupérée au pixel courant dans l'image canonique.

5.1.2.d Résultats expérimentaux

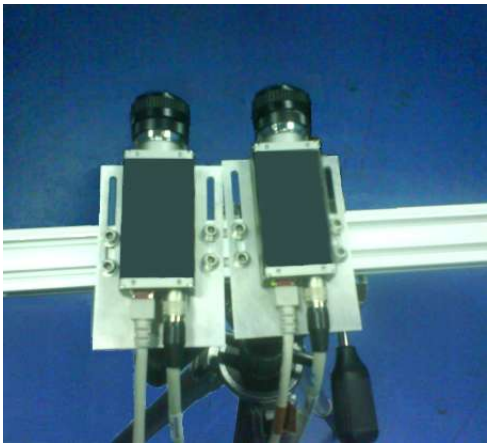


Figure 5.6 – Notre banc de capture stéréoscopique

Matériel utilisé La configuration matérielle utilisée pour ces captures vidéo stéréoscopiques est assez simple. Nous nous sommes servi de deux caméras vidéo, fixées sur un banc stéréoscopique (cf. Figure 5.6), qui permet le changement de l'écart entre les deux caméras ainsi que l'angle de convergence (même si nous utiliserons une capture en configuration parallèle).

Les deux caméras vidéo sont de type *BASLER pilot A1600-35gc*. Ces caméras peuvent capturer des images couleur en Haute Définition (1600 x 1200px) à la fréquence maximum de 35 images/s en utilisant une connexion Ethernet Gigabit. Elles sont peuvent être synchronisées par un signal de synchronisation externe, ce qui est une caractéristique primordiale si l'on souhaite capturer des images stéréo (droite et gauche) exactement au même moment. Cette synchronisation a été câblée en dur sur les entrées/sorties numériques TTL (Transistor-Transistor Logic) des deux caméras.

Dans la section suivante, nous allons présenter les résultats des tests expérimentaux que nous avons menés, sur chacune de nos implémentations (CPU et GPU), tout d'abord avec une caméra puis avec le banc stéréoscopique. Pour ces tests, nous avons utilisé un seul ordinateur (Intel Xeon CPU Quadcore E5405@2Ghz, 4Gb de RAM, et deux NVIDIA GeForce 9800 GX2 (512Mb) montées en SLI) sous Windows XP Pro Sp2 64 bits. Cependant, notre code a été également testé et fonctionne sur des PCs plus lents/anciens, équipés de processeurs et de cartes graphiques "standards".

Nous avons mené ces tests, en utilisant Virtools™(un logiciel [1] de création d'applications 3D temps réel) et Fraps [2] pour enregistrer la valeur moyenne des images par seconde traitées.

Mono-camera Les tableaux suivant montrent les résultats de nos expérimentations dans la configuration mono-caméra en utilisant notre algorithme de rectification sur GPU.

1. `http://www.virtools.com`
2. `http://www.fraps.com`

Les résultats sont comparés en mesurant l'impact de notre algorithme sur le nombre d'images par seconde (framerate) de l'application, tout d'abord sans le *shader* de rectification puis avec.

Pour effectuer de meilleures comparaisons, nous avons également appliqué notre *shader* à différentes sources vidéo : trois fichiers AVI en 1600x1200px (d'une taille égale en Mb mais à des fréquences différentes - 01, 30, et 50 images/seconde -) joués à partir du disque dur de la machine, et un flux vidéo 1600x1200px venant d'une caméra BASLER HD GigE.

Les résultats de la rectification dans la configuration mono-caméra sans le shader de rectification sont présentés dans le tableau 5.1 et avec le shader de rectification dans le tableau 5.2.

Les résultats des tableaux 5.1 et 5.2 mettent en évidence que l'application de notre algorithme, dans le cas mono-caméra, fait baisser le nombre d'images par seconde de seulement 14% par rapport au cas avec les images déformées. Ce faisant, il permet de garder un nombre d'images par seconde -rectifiées- très confortable pour de futurs traitements additionnels.

Mono camera	Virtools - DirectX 2D Frame	
	Mode Fenêtré	Plein Ecran
Avi 01	1853	1371
Avi 30	1661	1283
Avi 50	1584	1247
Basler 25	1546	1211

Tableau 5.1 – Résultats mono-caméra sans le shader de rectification. Les tests ont été réalisés dans les deux modes de rendu de Virtools (mode fenêtré et plein écran). Les résultats sont exprimés en nombre d'image par seconde lors de la lecture de vidéos pré-enregistrées à différents frame rate et du flux vidéo en provenance d'une caméra vidéo BASLER HD. Ce tableau va servir de base pour les comparaisons futures.

Mono camera	Virtools - DirectX 2D Frame	
	Mode Fenêtré	Plein Ecran
Avi 01	1452	750
Avi 30	1339	710
Avi 50	1267	680
Basler 25	1318	717

Tableau 5.2 – Résultats mono-caméra avec le shader de rectification. Similaire au tableau 5.1 mais cette fois les images sont rectifiées en temps réel avec notre algorithme.

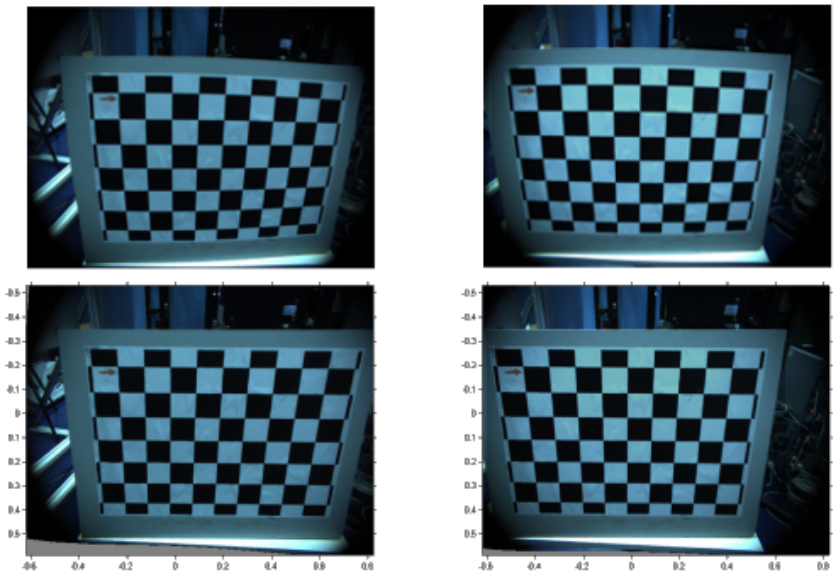


Figure 5.7 – 1er exemple de résultat de notre rectification stéréo : en haut se trouvent les images d'origine. En bas, la même paire d'images après rectification. Les images sont bien rectifiées et correctement alignées horizontalement. Le triangle gris en bas à gauche de l'image correspond à une zone inutilisable de l'image après rectification (il n'y a plus de parties de l'image à cet endroit).

Banc stéréoscopique Les tableaux suivant montrent les résultats de nos expérimentations sur un banc stéréoscopique en utilisant notre algorithme de rectification sur GPU. Les résultats sont comparés en mesurant l'impact de notre algorithme sur le nombre d'images par seconde (framerate) de l'application, tout d'abord sans le *shader* de rectification puis avec.

Pour effectuer de meilleures comparaisons, nous avons également appliqué notre *shader* à différentes sources vidéo stéréoscopiques : six fichiers AVI en 1600x1200px (deux fichiers différents par fréquence et d'une taille égale en Mb mais à des fréquences différentes - 01, 30, et 50 images/seconde -) joués à partir du disque dur de la machine, et deux flux vidéo 1600x1200px venant de deux caméras BASLER HD GigE.

Les résultats dans la configuration stéréo sans le shader de rectification sont présentés dans le tableau 5.3 et avec le shader de déformation dans le tableau 5.4.

Les résultats des tableaux 5.3 et 5.4 mettent en évidence que lors de l'application de notre algorithme, dans le cas d'un banc stéréoscopique, la baisse mesurée du nombre d'images par seconde est d'environ 39% par rapport au cas avec les images déformées. Cette baisse (supérieure à 2x14% de la baisse dans le cas mono) est en partie imputable à la manière dont Virtools gère en mémoire les deux flux vidéo, et leur applique le shader de rectification.

Malgré tout, le nombre d'images par seconde reste suffisamment important pour une rectification stéréoscopique temps réel, par rapport aux solutions existantes.

Nous proposons donc une méthode pour corriger les déformations sur des images stéréoscopiques. Cette approche a été implémentée sur les GPU, pour utiliser au maximum

Stereo cameras	Virtools - DirectX 2D Frame	
	Mode Fenêtré	Plein Ecran
Avi 01	2152	1696
Avi 30	1547	1191
Avi 50	1365	1032
Basler 25	870	797

Tableau 5.3 – Résultats banc stéréoscopique sans le shader de rectification. Les tests ont été réalisés dans les deux modes de rendu de Virtools. Les résultats sont exprimés en nombre d'image par seconde lors de la lecture de vidéos pré-enregistrées à différents frame rate et du flux vidéo en provenance des deux caméras vidéo BASLER HD du banc stéréoscopique. Ce tableau va servir de base pour les comparaisons futures.

Stereo cameras	Virtools - DirectX 2D Frame	
	Mode Fenêtré	Plein Ecran
Avi 01	950	622
Avi 30	830	540
Avi 50	759	470
Basler 25	527	479

Tableau 5.4 – Résultat en stéréo avec le shader de rectification. Même configuration que pour le tableau 5.3 mais cette fois les paires d'images sont rectifiées en temps réel avec notre algorithme.

les capacités naturelles de ces unités de calcul en ce qui concerne les calculs massivement parallèles et le support natif d'interpolateurs de très bonne qualité.

Cette méthode peut être facilement utilisée avec tous type de modèles de déformation aussi complexes soient-ils. En effet, chaque nouveau modèle de déformation requiert seulement le pré-calcul de nouvelles textures de déformations.

5.2 Détermination des paramètres visuels de l'utilisateur

Lorsqu'un utilisateur utilise un système de Réalité Virtuelle, les images qu'il perçoit d'une *scène virtuelle* ont été capturées le biais de deux caméras virtuelles (voir section 3.4.2). Sur une grande majorité de ces systèmes, ces deux caméras virtuelles sont écartées l'une de l'autre d'une distance égale à celle de la distance interoculaire(DIO) d'un humain moyen (63 mm). Lors de la restitution de ces images, la parallaxe est également fixée à cette valeur.

Mais en utilisant la DIO moyenne, on ne tient pas compte des variations qui existent entre les différents êtres humains. D'après [Dodgson 04], la variation de la DIO, chez les

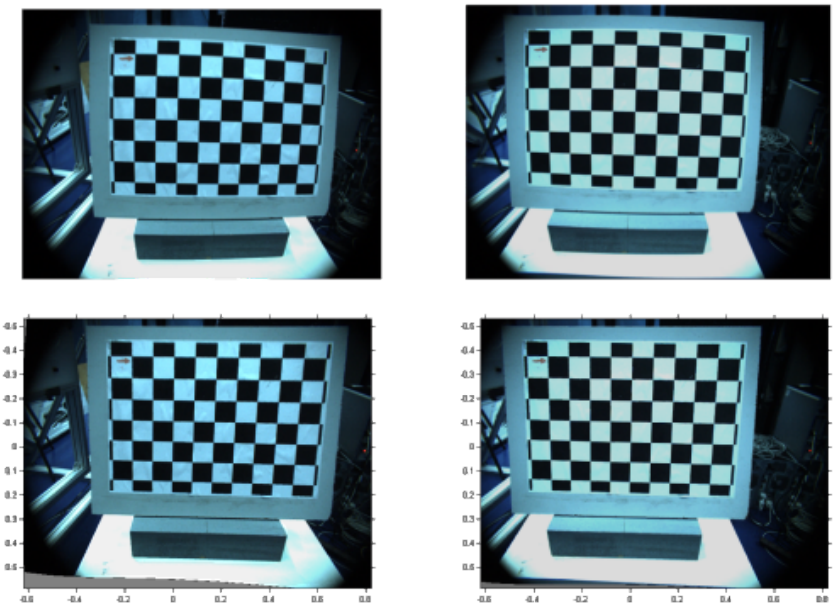


Figure 5.8 – Un autre exemple de notre rectification d'images en stéréo, en utilisant une autre paire d'images. Comme sur la figure 5.7, les images brutes sont en haut, et les images rectifiées en bas

humains adultes dans le monde, est comprise entre 50 et 75 mm. Cette approximation à 63mm -qui reste dans les plages de tolérances du système de perception humain à l'intérieur desquelles il ne ressent pas trop d'inconfort- est néanmoins utilisée pour permettre une plus grande souplesse dans l'utilisation du système.

Cependant, afin de maîtriser la restitution visuelle d'une profondeur, il est nécessaire de connaître l'écart interoculaire de manière précise. La méthode la plus précise consiste à se rendre chez un ophtalmologiste. Cependant, il est difficilement concevable/imaginable d'avoir en permanence un spécialiste prêt à mesurer l'écart inter oculaire de tout nouvel utilisateur, à proximité d'un système de Réalité Virtuelle. D'autres méthodes dans le domaine de la Réalité Virtuelle, se basent sur le recalage de grands cubes 3D (cf. [Andriot 92] par exemple), mais sont complexes à mettre en oeuvre.

Etant donné la difficulté que représentait pour nous la grande variété d'utilisateurs potentiels de notre système, nous avons cherché à déterminer de manière simple et rapide, une mesure précise de l'écart interoculaire de l'utilisateur.

5.2.1 Principe

Nous cherchons à connaître la position de chaque oeil de l'utilisateur dans un repère associé à sa tête. Pour cela, nous avons mis en place un dispositif composé d'un écran, d'une mire physique et d'un marqueur de position/orientation de la tête de l'utilisateur (cf. Figure 5.9).

Afin de déterminer la position d'un oeil, nous allons mesurer plusieurs rayons optiques. La position de l'oeil coïncide avec l'intersection de tous ces rayons visuels. Chaque rayon est défini par l'alignement avec l'oeil d'une mire projetée sur l'écran et d'une mire physique.

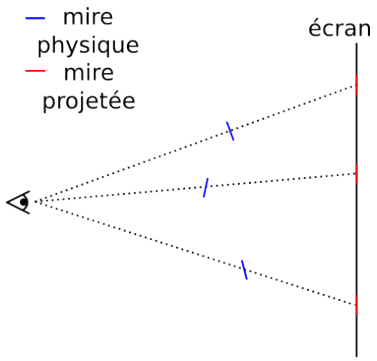


Figure 5.9 – Principe de l'alignement oeil - mire physique - mire projetée

La mire physique est une plaque en plexiglas translucide montée sur pied (pour s'assurer de sa stabilité et permettre un placement vertical précis et fixe) sur laquelle a été gravé une mire (cf Figure 5.10).

Une constellation de marqueurs de position/orientation (référencée en tant que *constellation plaque*) A.R.T. [3] (du nom de l'entreprise de capture de mouvement qui les fabrique) y est rattachée.

La transformation euclidienne pour obtenir les coordonnées de la mire physique, dans le repère lié au système de mesure A.R.T., à partir des mesures de placement de la constellation plaque dans ce même repère, va être détaillée dans la section 5.2.2.a.

5.2.2 Préliminaires à l'expérimentation

Pour mettre en place les expérimentations, liées au paramétrage de la distance interoculaire de l'utilisateur, nous avons utilisé le système de capture de mouvements A.R.T. Ce système est très fréquemment utilisé dans le domaine de la Réalité Virtuelle. Il est composé d'un ensemble de caméras infrarouge qui repèrent les déplacements de constellations de boules réfléchissantes, pour les transmettre à un ordinateur qui se charge de faire les calculs nécessaire à l'obtention des informations 3D/6D des positions et déplacements.

Des corps-rigides (structures à géométrie fixe qui portent les boules réfléchissantes de la constellation) ont été placées sur le marqueur tête que porte l'utilisateur, ainsi que sur la plaque en plexiglas qui supporte la mire.

5.2.2.a Déterminer la position de la mire dans le repère A.R.T.

La position du centre de la mire dans le repère associé à la plaque et porté par le corps rigide A.R.T. se déduit à partir de sa position sur la plaque en plexiglas. Ainsi, nous avons d'après la définition des coordonnées du corps rigide repéré par ART, on détermine A comme étant le marqueur #1, B le #2, et C le #3.

On a donc un repère centré en A dont l'axe $\vec{X}$ est porté par $\vec{AB}$.

Pour déterminer les autres axes, il va falloir connaître l'équation cartésienne du plan contenant A, B, et C.

3. Advanced Realtime Tracking GmbH

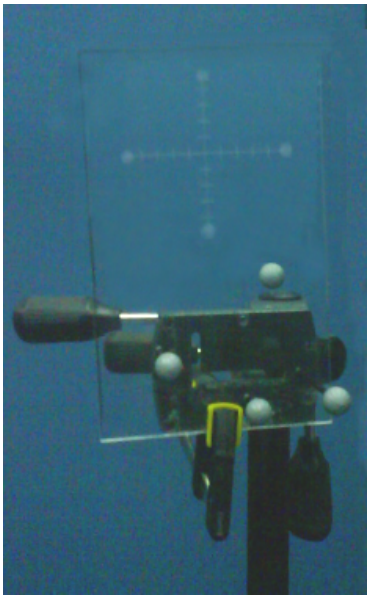


Figure 5.10 – Plaque en plexiglas pour le support de la mire

Cette équation permet d'obtenir les coordonnées de la normale à ce plan. Nous avons alors tout ce qu'il faut pour déterminer entièrement le repère associé au corps rigide du marqueur.

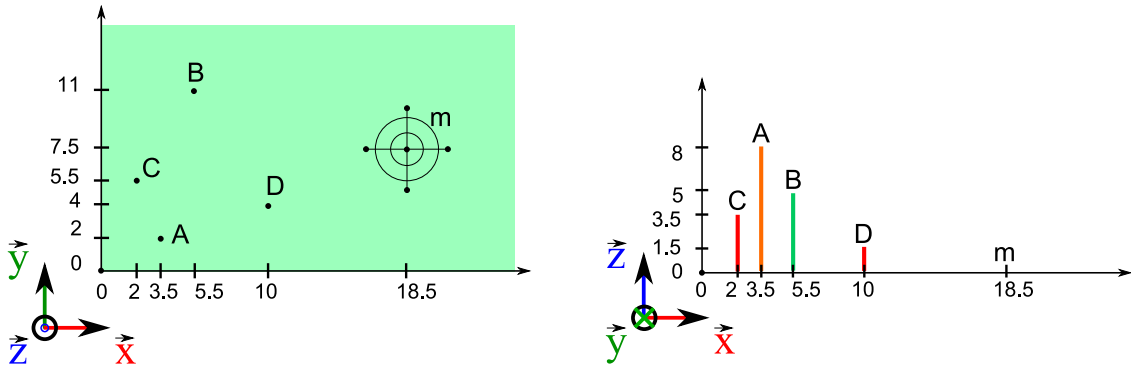


Figure 5.11 – Schéma de repérage des mires A.R.T. sur la plaque : vue de la plaque en plexiglas par le dessus sur la figure de gauche, et par le côté sur la figure de droite. Les coordonnées sont exprimées en cm sur les trois axes.

Nous disposons des 3 points A, B, et C. Nous pouvons donc écrire une équation cartésienne du plan ABC (de type $a.x + b.y + c.z + d = 0$).

De cette équation nous en déduisons les composantes de la normale au plan (ABC) tel que :

$$n = \begin{bmatrix} a \\ b \\ c \end{bmatrix}$$

Dans le repère associé à la plaque plexiglas, nous sommes en mesure de déterminer les coordonnées du vecteur $\vec{AB}$. On en déduit donc $\vec{y}=\vec{AB} \wedge \vec{n}$

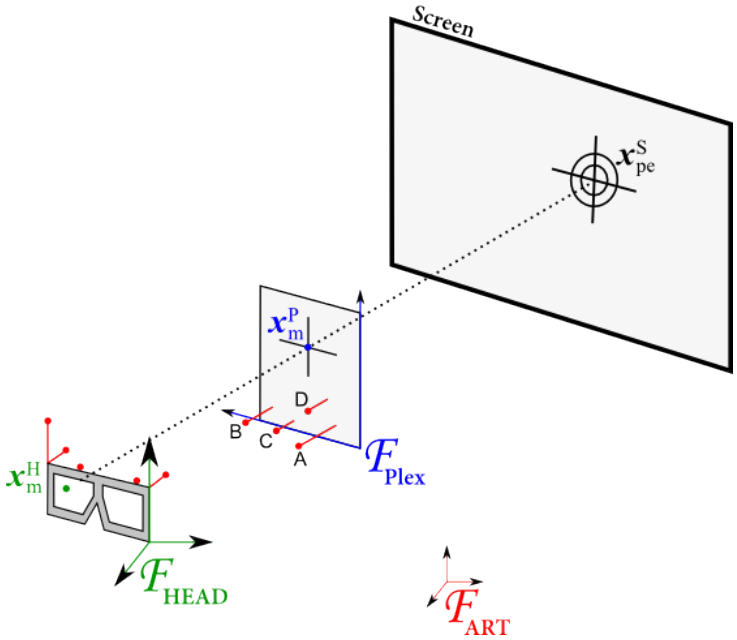


Figure 5.12 – Repères associés à la détermination des positions de chaque oeil de l'utilisateur

Enfin, nous avons le repère orthonormé direct $(\vec{AB}, \vec{y}, -\vec{n})$ porté par le marqueur A.R.T.

La transformation du repère associé à la plaque de plexiglas au repère associé au marqueur plaque s'exprime de la manière suivante :

$$R_{Plex}^{BodyPlex} = \left| \begin{array}{ccc} AB & y & -n \end{array} \right|$$

Par construction de la plaque, nous connaissons le placement du centre de la mire par rapport au repère plaque. Les coordonnées de m dans le repère associé au marqueur plaque sont donc :

$$\underline{x}_m^{BodyPlex} = R_{Plex}^{BodyPlex} \, \underline{x}_m^{Plaque}$$

La matrice de transformation de la mire dans le repère A.R.T. est constituée de $R_{BodyPlex}^{ART}$ et de $t_{BodyPlex}^{ART}$, tel que :

$$\underline{x}_m^{ART} = R_{BodyPlex}^{ART} \, \underline{x}_m^{BodyPlex} + t_{BodyPlex}^{ART}$$

5.2.2.b Déterminer la position de la mire dans le repère tête

La position de la mire dans le repère tête est déterminée par la position de la plaque dans ART sur laquelle est gravé le motif croix. La transformation de passage de la plaque dans le repère ART est constituée de R_{ART}^{Head} et de t_{ART}^{Head}, tel que :

$$\underline{x}_m^{Head} = R_{ART}^{Head} \, \underline{x}_m^{ART} + t_{ART}^{Head}$$

or ici $R_{ART}^{Head} = (R_{Head}^{ART})^T$

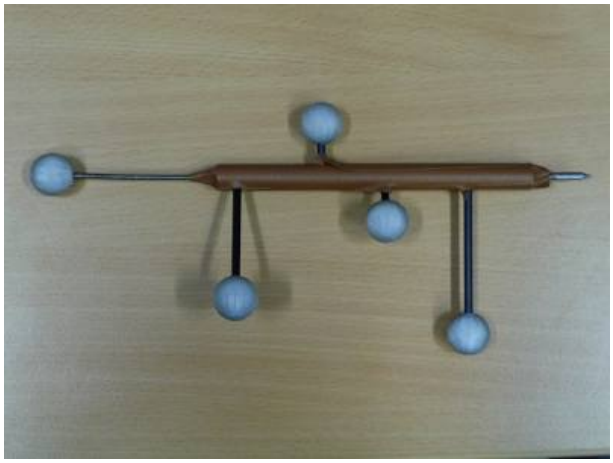


Figure 5.13 – Stylet de mesure avec une configuration A.R.T. qui permet de repérer la position de la pointe (à droite) dans le repère A.R.T.

5.2.2.c Calcul de la position du point écran dans le repère tête

Position dans le repère A.R.T.

La position du point écran est calculée par rapport à sa position en pixels sur la zone d'affichage. Il va donc être nécessaire de calculer sa position relative par rapport à des points écran bien identifiés dans le repère ART. Pour se faire, nous avons choisi de repérer dans A.R.T., la position des quatre coins de l'écran, sur notre surface de projection. Nous avons pour cela utilisé un stylet (développement CEA cf. Figure 5.13) dont la position de la pointe est repérée par A.R.T. Une fois les coordonnées des quatre coins connus, il va nous falloir déterminer la position du point écran ($\underline{x}_{pe}^{ART}$) par rapport aux quatre coins. Cette tâche n'est pas complexe, et nécessite seulement de récupérer en temps réel les coordonnées 2D associées à notre motif (la mire virtuelle).

Ainsi, un simple calcul proportionnel permet d'obtenir les coordonnées en A.R.T. du point courant, en fonction des quatre coins de l'écran préalablement mesurés.

Position dans le repère tête

Le calcul de la position du point écran (pe) dans le repère tête se calcule ensuite par un changement de repère :

$$\underline{x}_{pe}^{Head} = R_{ART}^{Head}\ \underline{x}_{pe}^{ART} + t_{ART}^{Head}$$

5.2.2.d Déterminer la position de chaque oeil dans le repère tête

Une fois, les positions du point écran et de la mire connues dans A.R.T., nous allons chercher à déterminer la position de chacun des deux yeux de l'utilisateur.

Pour cela nous réalisons pour chaque oeil une série d'alignements, entre la mire physique, la mire projetée sur l'écran et l'oeil, en enregistrant à chaque fois la position de chacune des mires, ainsi que la position/orientation du marqueur de tête. Chaque couple de mesures (mire projetée - mire physique) représente un rayon optique. Nous représentons cette droite 3D comme l'intersection de deux plans, décrit chacun par une normale ainsi

qu'une distance à l'origine. L'intersection de tous les plans est équivalente à la position de l'oeil.

Calcul des normales et distances à l'origine

Nous allons exprimer les positions d'un couple de points (positions de la mire projetée et de la mire physique) dans le repère tête.

Nous pouvons déterminer une équation d'un premier plan P_1 passant par le point mire projetée ($\underline{x}_{pe}^{H}$), par le point mire physique ($\underline{x}_{m}^{Head}$) et par l'origine ($\underline{O} = 0,0,0$).

A partir de cette équation, nous obtenons également une première normale à ce plan ($\underline{n}_1$,d_1) :

$\lambda\ \underline{n}_1 = \underline{x}_{pe}^{Head} \wedge \underline{x}_{m}^{Head}$ avec λ tel que $\|\underline{n}_1\| = 1$

On choisit ensuite le plan P_2 perpendiculaire à P_1, que l'on obtient grâce au produit vectoriel entre la normale $\vec{n_1}$ et le vecteur reliant les deux points $\underline{x}_{m}^{Head}$ et $\underline{x}_{pe}^{Head}$, tel que :

$\lambda_2\ \underline{n}_2 = \underline{n}_1\ \wedge\ (\underline{x}_{m}^{Head} - \underline{x}_{pe}^{Head})$

Pour tout point $\underline{t}$ du plan P_1, et en particulier $\underline{x}_{m}^{Head}$ et $\underline{x}_{pe}^{Head}$, on a :

$$\begin{bmatrix} \underline{n}_1 \\ d_1 \end{bmatrix}^T \quad \begin{bmatrix} \underline{x}_{m}^{Head} \\ 1 \end{bmatrix} = 0 \quad \text{et} \quad \begin{bmatrix} \underline{n}_2 \\ d_2 \end{bmatrix}^T \quad \begin{bmatrix} \underline{x}_{pe}^{Head} \\ 1 \end{bmatrix} = 0$$

d'où $d_1 = -\underline{n}_1^T\ \underline{x}_{m}^{Head}$ et $d_2 = -\underline{n}_2^T\ \underline{x}_{pe}^{Head}$

Obtention de la position de l'oeil

Lors de la calibration d'un oeil, pour tous les rayons (positions où l'oeil, la mire physique et la mire projetée sont alignées), on cherche le point $\begin{bmatrix} \underline{x}_{eye}^{Head} 1 \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix}$ tel que :

$$\begin{bmatrix} n_1 & d_1 \\ n_2 & d_2 \\ n_3 & d_3 \\ n_4 & d_4 \\ .. & .. \end{bmatrix} \quad . \quad \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} = 0$$

Ce système homogène linéaire, de type $A\ \underline{x} = 0$, a comme solution non triviale $\underline{x}$, le vecteur colonne $\underline{v}$ de la matrice associée V à la plus petite valeur singulière s, pour la décomposition $A = USV^T$. Cette solution correspond aux coordonnées de l'oeil dans le repère associé à la tête de l'utilisateur, après normalisation.

Ce processus de calibration doit être réalisé une nouvelle fois pour le second oeil (non calibré).

La précision de la résolution dépend principalement des mouvements de la constellation A.R.T. (placée sur la tête de l'utilisateur) par rapport aux yeux et donc par rapport à la tête elle-même. Si ce corps rigide reste fixe d'une mesure à l'autre, il va garantir que l'oeil que l'on calibre se situe toujours à la même position par rapport au repère tête.

Dans le cas où ce repère serait amené à bouger par rapport aux yeux, la résolution du système va fournir une position qui risque d'être un peu écartée de la position de l'oeil ou bien totalement erronée. Il est donc important que la position du marqueur soit la plus fixe possible lors des mesures.

Avant de réaliser notre expérimentation, nous avons mis en place une étude de sensibilité en Matlab pour nous assurer que notre résolution était juste, en nous basant sur des données simulées. Ensuite nous présenterons les résultats de ces expérimentations.

5.2.3 Etude de la sensibilité de notre calibration

Une étude sur la sensibilité de nos calculs à divers paramètres a été réalisée afin d'étudier leurs influences sur le résultat final. Nous avons étudié les sensibilités aux déplacements en translation (sur les trois axes autour de la position originale), ainsi qu'en rotation (autour des trois axes) des trois composants du dispositif d'alignement (cf. section 5.2.1).

La première étape a été de valider, avec des mesures maîtrisées, que notre algorithme de résolution trouvait la bonne position pour chaque oeil réalisant l'alignement. Ceci nous a permis de vérifier le bon fonctionnement de notre algorithme de calibration.

Ensuite, nous avons testé la robustesse de notre calibration à des perturbations. En effet, les conditions réelles de mesure induisent des souvent instabilités de position.

Les équations de propagation d'erreurs étant trop complexes à introduire dans notre calcul, nous avons choisi la solution de la simulation de valeurs avec un bruit maîtrisé. Nous avons donc injecté des valeurs aléatoires dans notre algorithme afin de déterminer leur impact sur chaque composant des alignements.

Le principe de la génération des valeurs aléatoires est le suivant : nous fixons d'abord la position de l'oeil à calibrer, dans le repère tête de l'utilisateur.

Ensuite, nous créons des positions aléatoires pour l'utilisateur (par rapport à l'écran), la mire écran et la mire physique. Cette génération peut être décomposée de la façon suivante :

1. On définit une taille en hauteur/largeur pour l'écran.
2. On génère une position 6D aléatoire de l'écran.
3. On génère une distance aléatoire de l'utilisateur par rapport à l'écran, au sein d'une plage de valeurs fixe.
4. On génère une distance aléatoire de la mire par rapport à l'écran, au sein d'une plage de valeurs fixe.
5. Au sein de l'espace définit par l'utilisateur, la mire et l'écran, on crée des rayons positionnés et orientés aléatoirement.
6. L'intersection des rayons (5) aux plans verticaux placés aux distances (3) (resp. (4)), permettent pour chacun de ces rayons de déterminer la position 6D de l'utilisateur (resp. de la mire physique).
7. On exprime toutes les positions 6D dans le repère tête associé à l'utilisateur.
8. Les positions 6D de l'utilisateur et de la mire vont nous servir de données de simulation pour l'obtention de la position de l'oeil dans le repère tête.

La position arbitraire que nous avons choisi pour l'oeil dans le repère tête était la suivante : [0.05 0.04 0.1]. Nous allons comparer dans les résultats suivants, l'écart par rapport à cette position nominale, en fonction de l'introduction de différents types de bruits sur 25 alignements aléatoires.

5.2.3.a Ajout d'un bruit en translation pure

Nous avons ajouté plusieurs bruits de différentes valeurs de manière à visualiser les conséquences d'erreurs en translation. Nous avons représenté sur les Figures 5.14, 5.15 et 5.16, les évolutions des valeurs(en ordonnée) des trois composantes de la position calculée de l'oeil en fonction des bruits suivants (exprimés en mètres et positionnés en abscisse). Dans les sections suivantes, nous détaillons l'ajout de ces valeurs aux positions de la tête, de la mire écran, et de la mire physique : [-0.010 -0.009 -0.008 -0.007 -0.006 -0.005 -0.004 -0.003 -0.002 -0.001 0 0.001 0.002 0.003 0.004 0.005 0.006 0.007 0.008 0.009 0.010]

Variations de la position de la tête

Nous observons que les valeurs moyennes des composantes X et Y de la position de l'oeil calculées augmentent de façon linéaire d'une valeur égale à la moitié du biais ajouté. Ainsi, par exemple, 0.5cm de décalage a pour conséquence une imprécision de 0.25cm$\pm$0.1 sur la valeur finale de la position de l'oeil. De plus, on observe que la composante Z est très sensible aux perturbations, l'erreur augmentant peu mais les variations autour de la valeur moyenne sont très importantes : $\pm$7-6cm aux extrêmes.

Variations de la position de la mire écran

Les résultats obtenus pour les variations de la position de la mire écran, sont assez similaires à celle de la tête sur les composantes X et Y de la position de l'oeil. Il est nécessaire de noter que les conséquences des variations de la composante Z sont très importantes et induisent des résultats très éloignés de la réalité. Cependant, la mire écran ne se déplaçant qu'en 2D sur l'écran, l'erreur en Z est négligeable, et est considérée comme nulle.

Variations de la position de la mire physique

Les résultats obtenus pour les variations de la position de la mire écran sont inversement proportionnels à la valeur biais introduit. Ainsi, un décalage 0.01m génère une imprécision de -0.01m$\pm$0.004 et un décalage -0.01m génère une imprécision de +0.01m$\pm$0.004sur les composantes X et Y de la position de l'oeil calculées. On observe également, que les erreurs sur la position en Z de la mire physique, génèrent de très importantes erreurs (à l'échelle de nos calculs) : ainsi $\pm$0.01m d'erreur génère une erreur sur la position en Z de l'oeil de -0.09m$\pm$0.06.

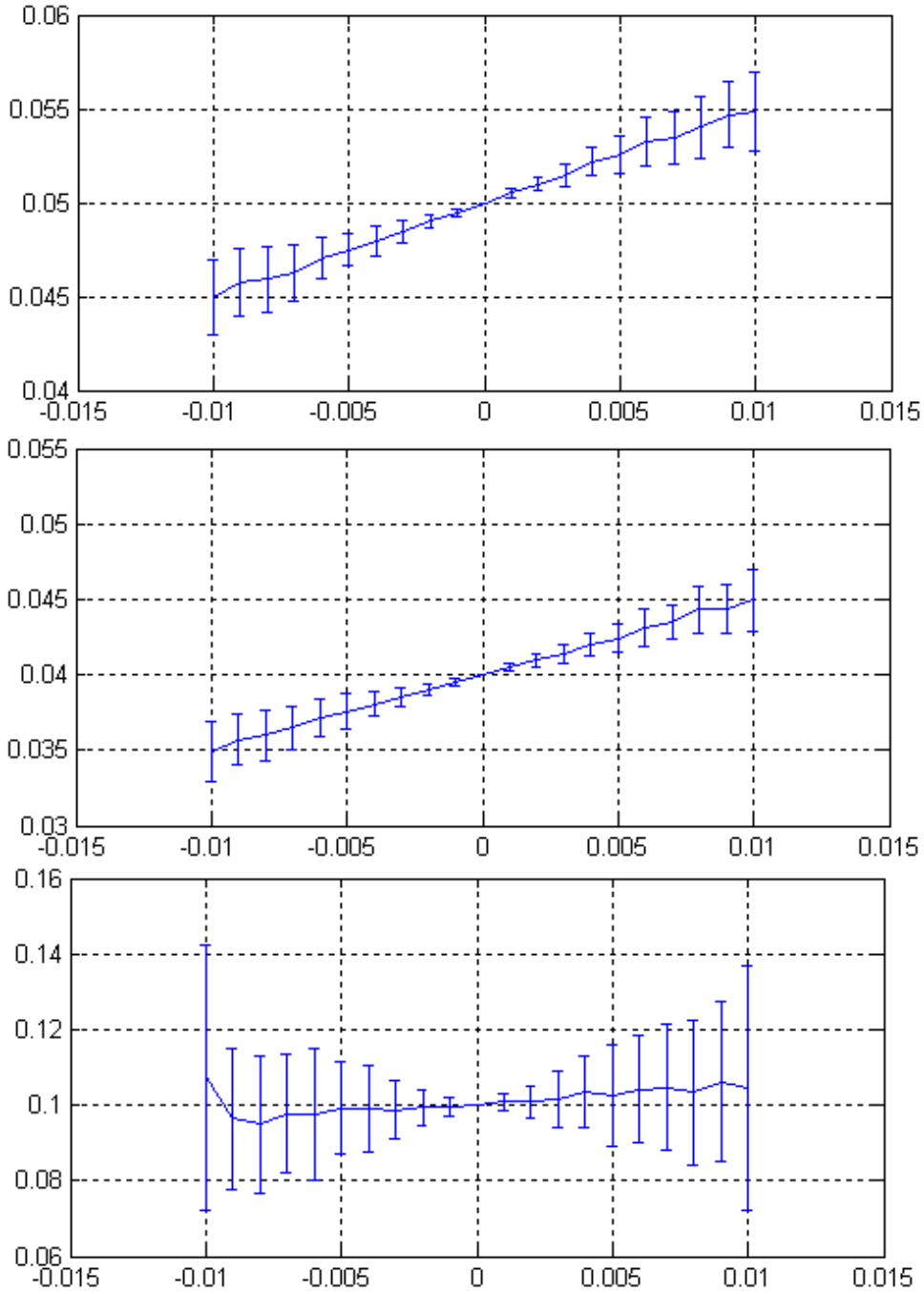


Figure 5.14 – Impact sur la position calculée de l'oeil, suite à l'ajout d'un bruit en translation sur la position de la tête. De haut en bas, les composantes X, Y, et Z. En abscisse on repère la valeur du bruit et en ordonnée la valeur de la composante, exprimés en mètre. La position nominale de l'oeil se trouve en [0.05 0.04 0.1].

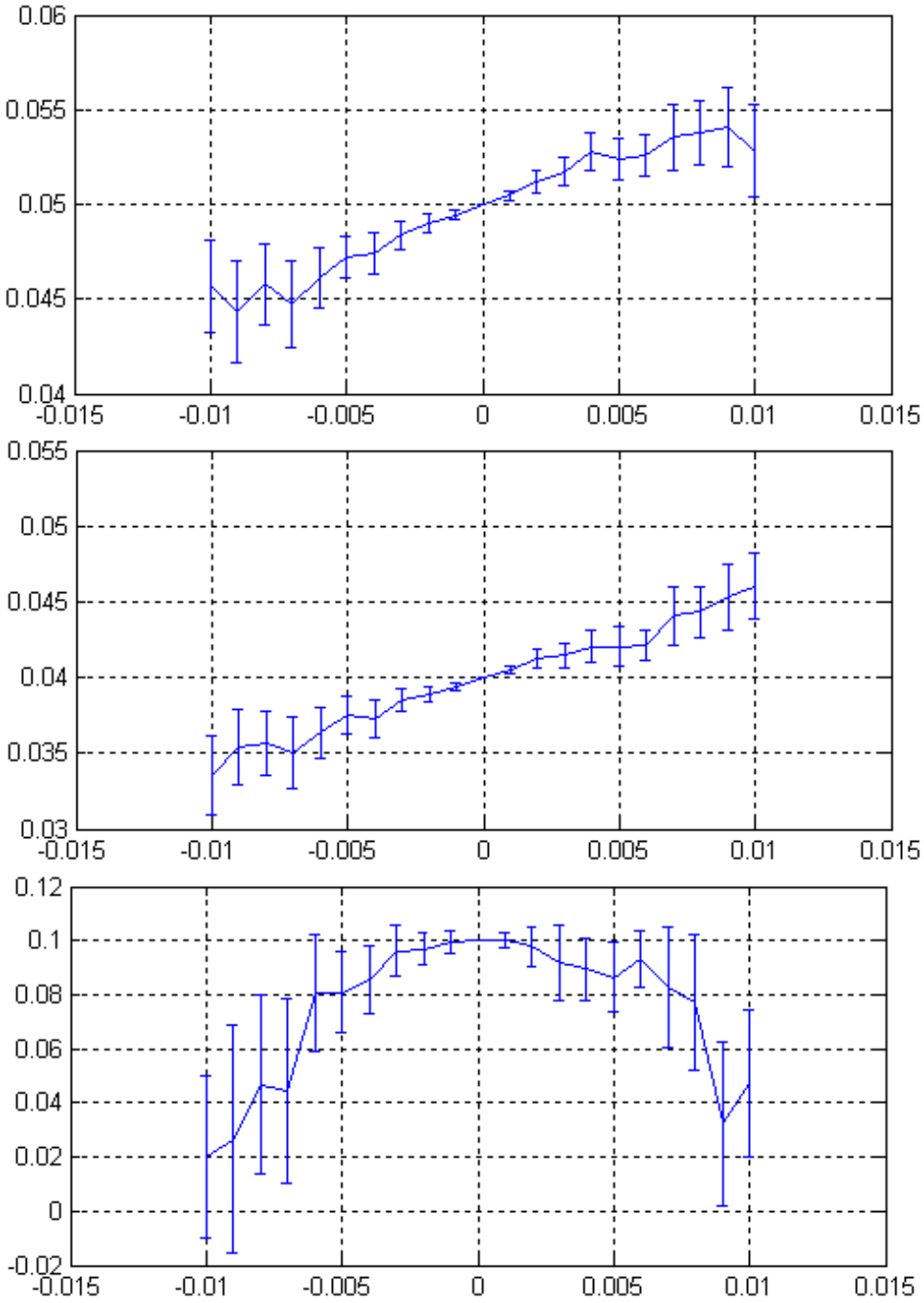


Figure 5.15 – Impact sur la position calculée de l'oeil, suite à l'ajout d'un bruit en translation sur la position de la mire écran. De haut en bas, les composantes X, Y, et Z. En abscisse on repère la valeur du bruit et en ordonnée la valeur de la composante, exprimés en mètre. La position nominale de l'oeil se trouve en [0.05 0.04 0.1].

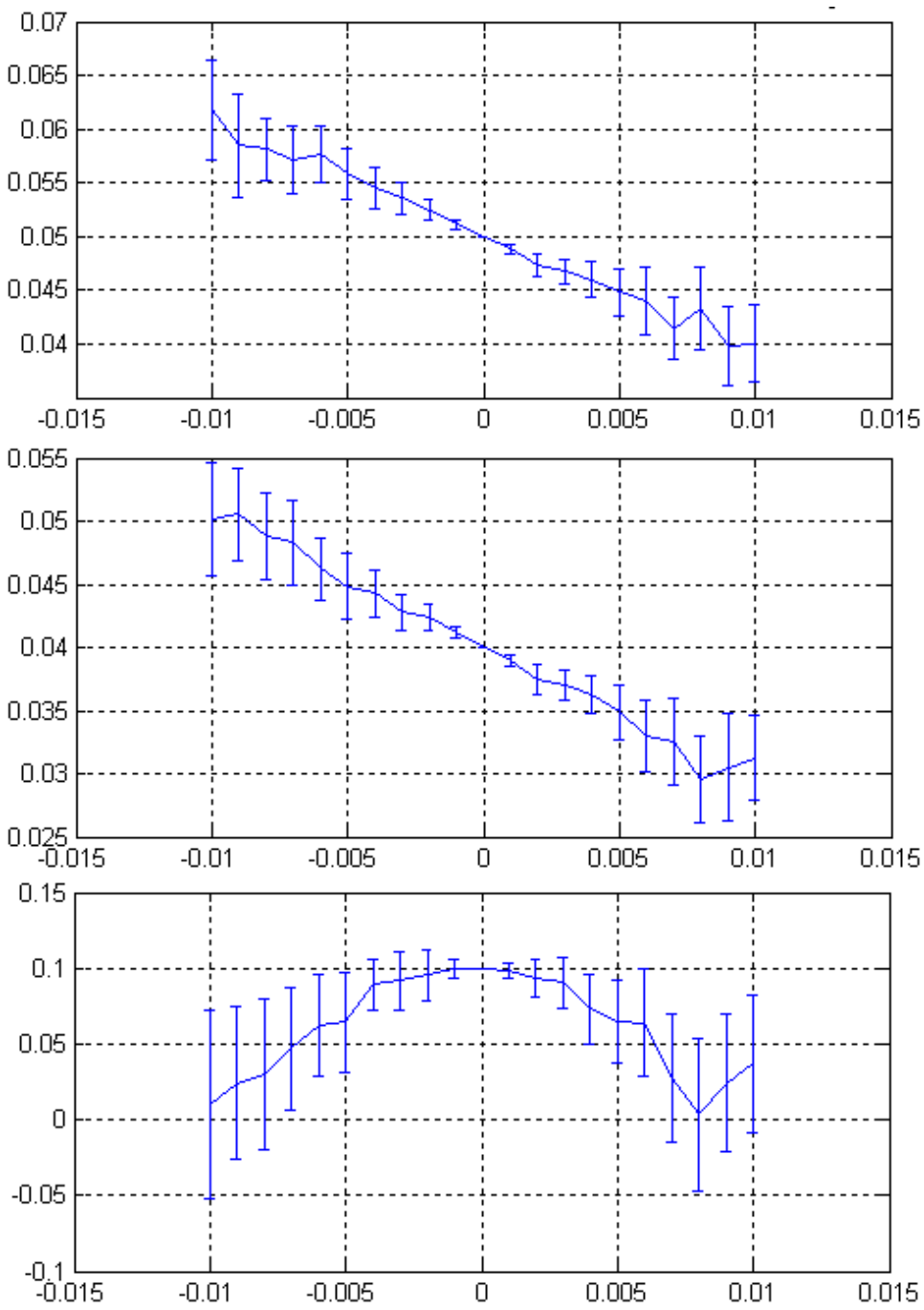


Figure 5.16 – Impact sur la position calculée de l'oeil, suite à l'ajout d'un bruit en translation sur la position de la mire physique. De haut en bas, les composantes X, Y, et Z. En abscisse on repère la valeur du bruit en mètre et en ordonnée la valeur de la composante, exprimés en mètre. La position nominale de l'oeil se trouve en [0.05 0.04 0.1].

5.2.3.b Ajout d'un bruit en rotation pure

Nous avons ajouté plusieurs bruits de différentes valeurs de manière à visualiser l'évolution des erreurs résultant de l'ajout d'un bruit en rotation. Nous avons représenté sur les Figures 5.17, 5.18 et 5.19, les évolutions des résultats en fonction des bruits suivants (exprimés en degrés) ajoutés aux valeurs d'orientation de la tête, de la mire écran, et de la mire physique : [-85 -70 -55 -40 -25 -10 0 10 25 40 55 70 85]

Variations d'orientation de la tête

D'après les résultats obtenus et conformément à ce que l'on s'attend d'une telle situation, nous observons que des erreurs en rotation sur l'orientation de la tête influent de manière très importante sur les trois composantes de la position de l'oeil calculées. Ainsi, si la tête d'un utilisateur est droite, et que l'erreur de mesure est importante ($>10^\circ$), la position de l'oeil sera calculée comme s'il s'était déplacé par rapport au repère tête. Les figures confirment ce raisonnement. La fourchette $0\pm10^\circ$ ne génère que de très faibles écarts par rapport à la position de référence. Cependant, pour tout angle supérieur, l'erreur et l'imprécision sont grandes.

Variations d'orientation de la mire écran et de la mire physique

Les rotations des deux mires, autour de leurs centres, n'impactent pas l'alignement et donc la position calculée de l'oeil dans le repère tête dans la simulation, comme nous pouvons le voir sur les résultats des simulations. En conditions réelles, dans le cas où l'utilisateur peut distinguer le centre de chaque mire, il lui est possible d'aligner son oeil et les deux mires. En cas d'angles trop importants, il lui sera difficile de distinguer les centres de chacune des mires.

Les résultats de notre simulation montrent qu'il est très important de travailler avec un marqueur tête fixe par rapport aux yeux le temps de la prise de mesure. Les mouvements de ce marqueur (en translation et en rotation) impactent directement la précision du résultat trouvé.

Il est également nécessaire de travailler avec un écran en position fixe. En effet, la mire écran étant projetée sur l'écran, si l'alignement se fait sur une position erronée de la mire due à un déplacement de l'écran, l'algorithme perdra en précision pour trouver le point de convergence des droites.

De la même manière, la stabilité de la mire physique est à prendre en compte, même si son importance est moindre comparée aux influences précédemment décrites.

Cependant, lorsque l'on se trouve dans une configuration stable des marqueurs et des mires, nous constatons une robustesse importante de notre algorithme, qui parvient à converger précisément sur la position nominale de l'oeil calibré.

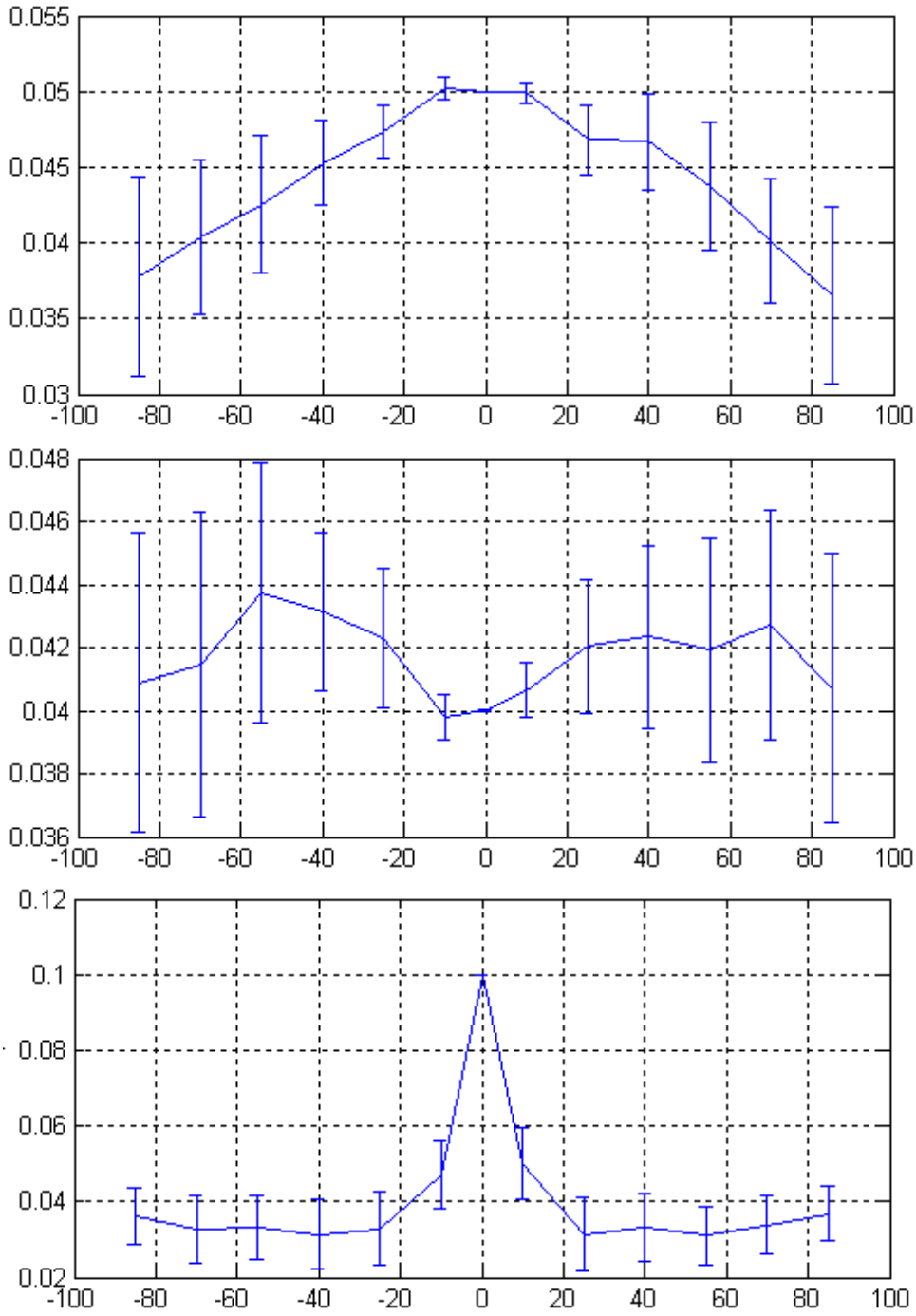


Figure 5.17 – Impact sur la position calculée de l'oeil, suite à l'ajout d'un bruit en rotation sur la position de la tête. De haut en bas, les composantes X, Y, et Z. En abscisse on repère la valeur du bruit en degré et en ordonnée la valeur de la composante. La position nominale de l'oeil se trouve en [0.05 0.04 0.1].

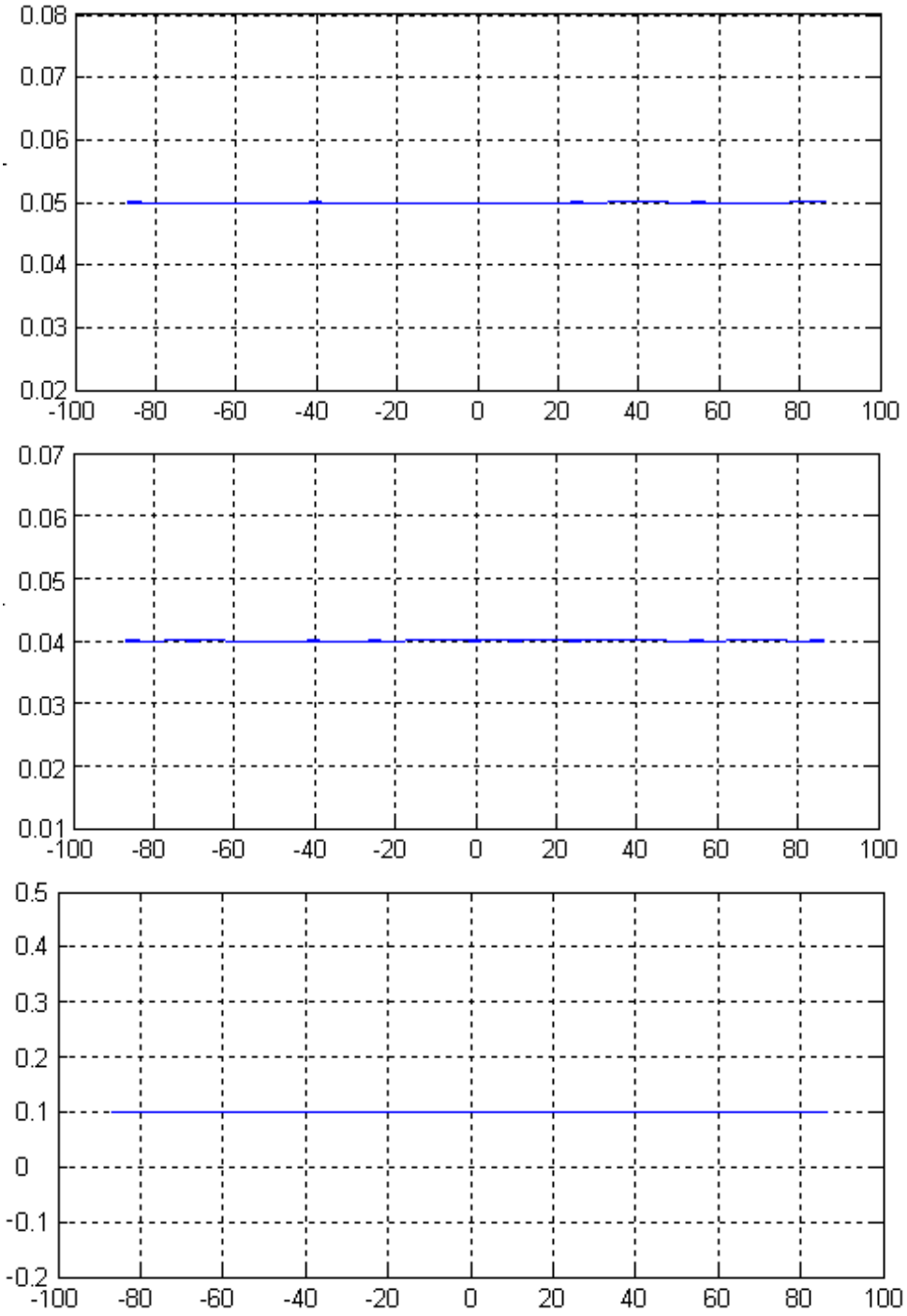


Figure 5.18 – Impact sur la position calculée de l'oeil, suite à l'ajout d'un bruit en rotation sur la position de la mire écran. De haut en bas, les composantes X, Y, et Z. En abscisse on repère la valeur du bruit en degré et en ordonnée la valeur de la composante. La position nominale de l'oeil se trouve en [0.05 0.04 0.1].

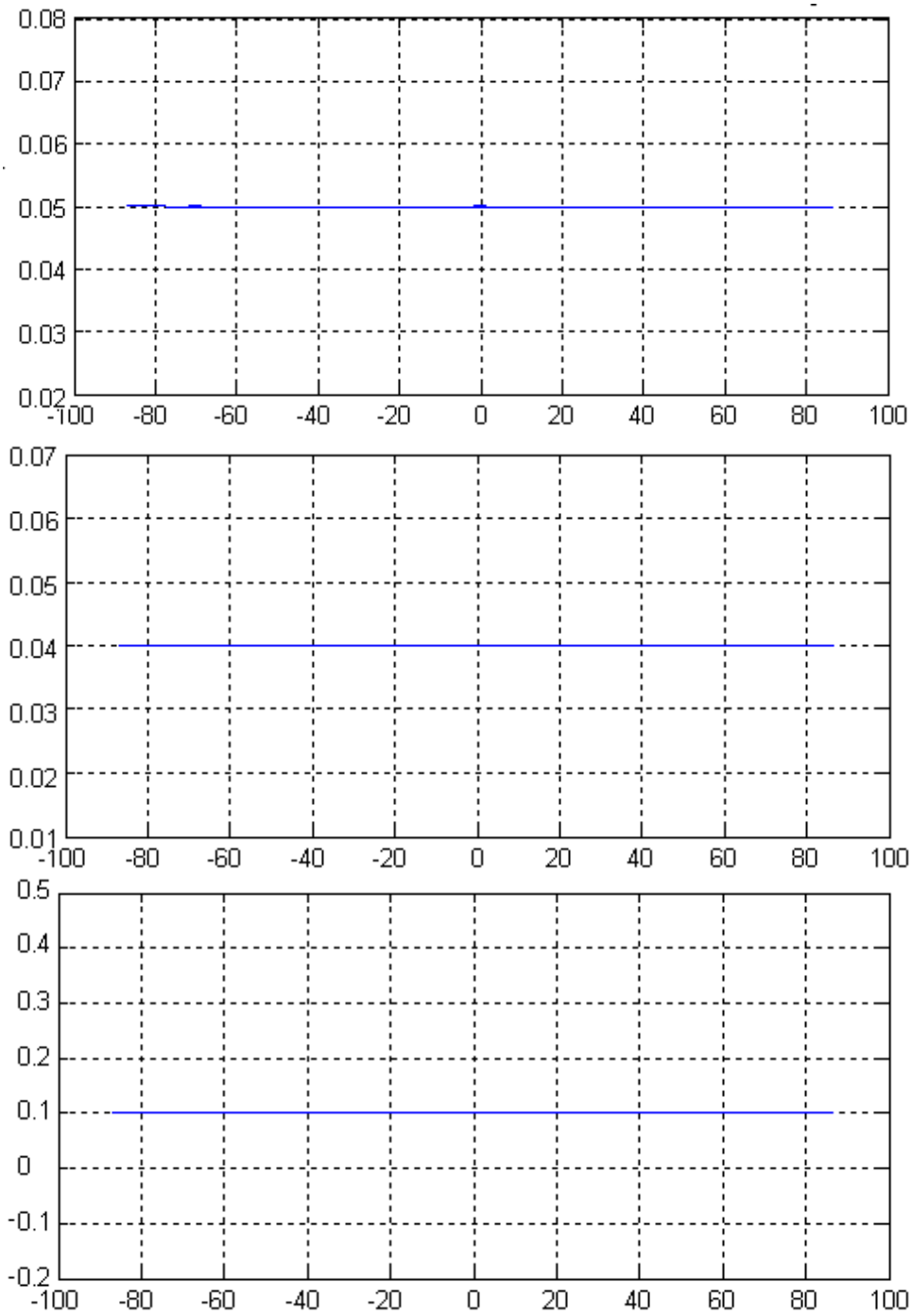


Figure 5.19 – Impact sur la position calculée de l'oeil, suite à l'ajout d'un bruit en rotation sur la position de la mire physique. De haut en bas, les composantes X, Y, et Z. En abscisse on repère la valeur du bruit en degré et en ordonnée la valeur de la composante. La position nominale de l'oeil se trouve en [0.05 0.04 0.1].

5.2.4 Résultats expérimentaux

La mise en place d'expérimentations nous a permis de valider les simulations que nous avions effectuées. Nous présentons dans la suite les résultats de ces expérimentations.

5.2.4.a Validation des hypothèses de mesure

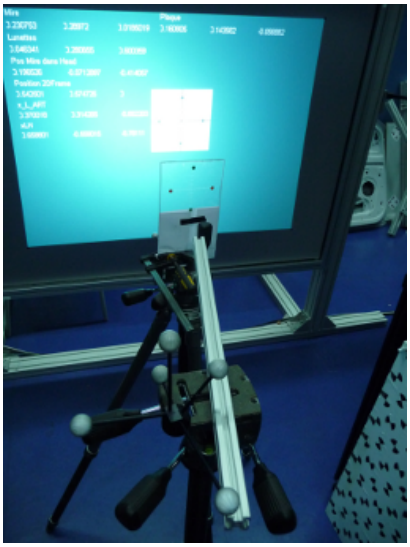


Figure 5.20 – Simulation d'un oeil : la barre métallique est en liaison pivot autour d'un axe vertical, centre de l'oeil simulé

Dans un premier temps, nous avons validé expérimentalement nos hypothèses de travail. Il était nécessaire de nous assurer que les mouvements de l'utilisateur, et des deux mires étaient correctement repérées par le système de mesure. Pour cela, nous avons comparé les positions calculées des deux mires avec celles de la pointe du stylet CEA (cf. Figure 5.13) positionné sur le point central de la mire.

Ensuite, afin de valider la partie résolution de notre algorithme pour la position de l'oeil dans le repère tête, nous avons mis en place un système permettant de simuler un oeil en liaison pivot. En effet, la vérification de la position physique exacte du centre de vision dans l'oeil de l'utilisateur n'est pas facilement vérifiable.

Des alignements ont été réalisés en alignant l'axe longitudinal d'une barre métallique avec les deux mires. Cette barre été fixée en liaison pivot -centre de l'oeil simulé- autour d'un axe vertical, et fixée sur un support dont la position dans la scène était repérée par le système de capture A.R.T. (cf. Figure 5.20). La position du point de pivot sur le support, ainsi que la position de la constellation ont été enregistrées afin de déterminer la position nominale de l'oeil virtuel dans le repère associé au support.

5.2.4.b Réalisation d'une campagne de mesures

Nous avons réalisé une campagne de mesures avec différents utilisateurs, afin de tester notre méthode de résolution. Le dispositif de calibration (cf. Figure 5.21) est composé :

– d'un système de restitution visuelle par projection

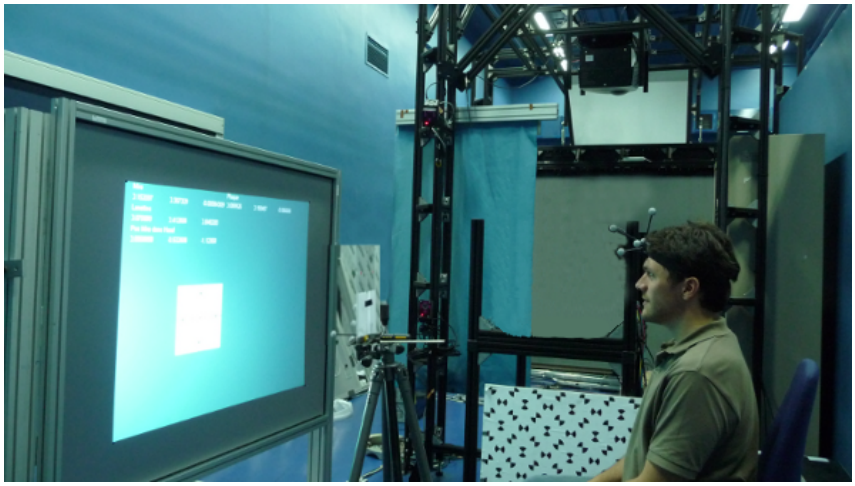


Figure 5.21 – Configuration d'alignement pour la calibration

- d'une mire en plexiglas transparent
- d'un système de capture de mouvements.

Une mesure se déroule de la manière suivante. Une image est affichée, dotée d'un point caractéristique (A) associé à une mire virtuelle projetée sur l'écran. L'utilisateur se place devant le système. Pour chaque oeil, il faudra qu'il aligne le centre de la mire en plexiglas avec plusieurs positions différentes du point A. Ce point est libre de se déplacer partout sur l'écran, et la mire peut également être déplacée dans toutes les directions de manière à ce qu'elle soit correctement alignée.

De plus, lors de notre calibration, le positionnement du marqueur tête pendant la capture n'est pas soumis à des conditions particulières, étant donné que nous ramenons toutes les mesures dans ce repère pour calculer la valeur finale de la position de l'oeil. Il est seulement nécessaire qu'il soit solidaire de la tête pendant la capture.

A chaque instant, grâce aux mesures prises par le système de capture de mouvements A.R.T., nous obtenons les positons/orientations de la tête de l'utilisateur ainsi que celles de la mire physique.

Nous connaissons également, de manière précise, où se trouve le point caractéristique A que nous affichons à l'écran, dans le repère A.R.T.

La fusion de toutes ces informations va nous permettre de déterminer la position de chaque oeil par rapport au marqueur associé à la tête, ce qui nous donnera par la suite l'écart interoculaire. Les résultats que nous avons obtenus se sont révélés très proches des données mesurées par les ophtalmologistes. De plus, on constate des résidus très faibles indiquant une précision importante de la solution trouvée (cf. tableau 5.22).

	DIO	Résidus
Utilisateur 1	0.0701	[0.0018 -0.0017]
Utilisateur 2	0.0628	[0.0023 -0.0012]
Utilisateur 3	0.0641	[0.0012 -0.0019]

Figure 5.22 – Quelques exemples de résultats obtenus pendant une campagne de mesure.

Nous venons de présenter dans ce chapitre la méthode que nous avons développée pour rectifier en temps réel des images déformées en haute défnition, issues d'un banc stéréoscopique. Nous avons pour ce faire utilisé intensivement les capacités de calcul qu'offrent les cartes graphiques modernes. Les images de nos caméras, exemptes de toute déformation, sont maintenant utilisables de manière fiable pour d'autres applications.

De plus, nous avons mis au point un procédé de calibration de l'écart interoculaire d'un utilisateur basé sur des alignements mires-oeil. La détermination de la position de chaque oeil est effectuée dans un même repère associé à la tête de l'utilisateur. Cette calibration (plus simple que les calibrations déjà existantes dans le domaine) nous fournit des mesures de manière rapide et très précise. Nous pouvons ainsi régler finement l'affichage en relief d'images virtuelles issues de simulations numériques, sur des systèmes de Réalité Virtuelle.

Pour améliorer l'affichage d'images stéréoscopiques d'une scène réelle sur ces systèmes (par exemple intégrées au sein d'une simulation numérique), nous allons devoir effectuer davantage de traitements sur ces images.

Le chapitre suivant détaille les solutions que nous proposons pour restituer les profondeurs d'une scène réelle lors de l'affichage des images stéréoscopiques de la capture. Le cas d'un utilisateur mobile, dont les mouvements sont enregistrés par un système de capture de mouvements, est également traité.

Chapitre 6

Réglages des paramètres de la restitution d'images vidéo stéréoscopiques

Sommaire

Dans ce chapitre, nous nous plaçons dans le cas où l'on souhaite visualiser des images vidéo issues d'une capture stéréoscopique. Nous choisissons arbitrairement que les paramètres de configuration du banc (longueur de la ligne de base des caméras, configuration convergente/parallèle,...) ne soient pas modifiables pendant la capture. Sans modification des images lors de la restitution, il n'existe qu'une configuration qui permet une capture et une restitution échelle 1 du contenu filmé. Cette configuration est appelée ortho-stéréoscopique. Cependant, l'utilisateur va être contraint de conserver une position et une orientation donnée afin de percevoir l'environnement à l'échelle filmée. Nous présentons dans la section 6.1 les détails de ces conditions particulières.

Seulement, un problème important de déformation des informations perçues par l'utilisateur (profondeur, géométrie,...) se pose lorsque celui-ci sort de cette position définie. Dans la section 6.2, nous présentons les développements que nous avons mis en place afin de permettre une plus grande souplesse dans la restitution de ces images, lorsque l'utilisateur se déplace.

6.1 Restitution ortho-stéréoscopique

Les conditions ortho-stéréoscopiques sont des conditions de capture/affichage particulières. Elles permettent de percevoir la scène filmée sans compression en profondeur, ni déformations en hauteur/largeur. Cependant, elles limitent le positionnement de l'utilisateur devant son système de restitution, ainsi que la configuration de capture utilisée. En effet, la restitution à l'échelle d'une scène filmée n'est obtenue que si l'on se place à une distance donnée, dont la valeur peut être calculée à l'avance lors de la mise en place du système de capture. Mais il sera également nécessaire que l'on capture les images d'une manière particulière.

6.1.1 Paramétrage du cas ortho-stéréoscopique

La configuration de la restitution ortho-stéréoscopique impose de faire correspondre la vision de l'utilisateur avec celle des caméras. L'utilisateur, visualisant les images restituées, aura un point de vue identique à celui qu'il aurait eu s'il avait été placé en lieu et place des caméras dans l'environnement (virtuel ou réel). La vision restituée sera alors isomorphe à la vision du monde réel [Fuchs 06].

Le cas spécifique de la vision ortho-stéréoscopique impose donc que de nombreux paramètres soient fixés. Ainsi, on va fixer :

- la longueur de la ligne de base du banc stéréoscopique est égale à la distance interoculaire de l'observateur
- la distance entre l'observateur et l'écran doit correspondre au produit $f\ M$ où f est la focale des caméras et M le facteur d'agrandissement
- l'angle du champ de vision de l'observateur au niveau de l'écran est équivalent à celui des cameras.

Néanmoins, la littérature mentionne que d'autres approches [Holliman 04] [Jones 01] [Wartell 99] ont étudié le problème de la restitution de la profondeur d'une scène, d'une autre manière. En effet, les systèmes d'affichage en relief ont un volume de travail limité autour de l'écran, au sein duquel la perception de la profondeur va être perçue sans gène

par l'utilisateur. Il est très rare que ce volume corresponde exactement au volume de la scène capturée. Ceci étant, ces approches proposent de faire correspondre la profondeur capturée de la scène, aux limites de profondeur de l'écran.

Cependant, même si elles permettent d'assurer que les profondeurs perçus conservent les mêmes proportions que celles de la scène filmée, ces approches ne garantissent pas dans tous les cas, la conservation de l'échelle 1.

6.1.2 Expérimentation du cas ortho-stéréoscopique

Pour valider ces hypothèses de capture et restitution ortho-stéréoscopique, nous avons mis en place une expérimentation de restitution temps réel à l'échelle 1. Cette expérimentation a eu lieu lors de l'édition du salon Laval Virtual 2008. Dans le cadre du projet RNTL Part@ge [1], qui traite de l'interaction collaborative entre plusieurs utilisateurs au sein d'un environnement virtuel 3D, nous avons expérimenté l'affichage d'une personne distante en vidéo relief.

Sur un stand, se trouvaient les caméras avec une zone de capture définie, pour laquelle la restitution était mise en place. Sur le second stand se trouvait le dispositif d'affichage.

Nous avons utilisé, pour capturer les images, un banc stéréoscopique fixe composé de deux caméras uEye 2210-C [2] (connexion USB, résolution 640x480, @25Hz) ainsi qu'un système de visualisation de type Workbench (stéréo active, deux écrans, appartenant au centre d'étude et de recherche CLARTE) pour la restitution vidéo du flux stéréoscopique.

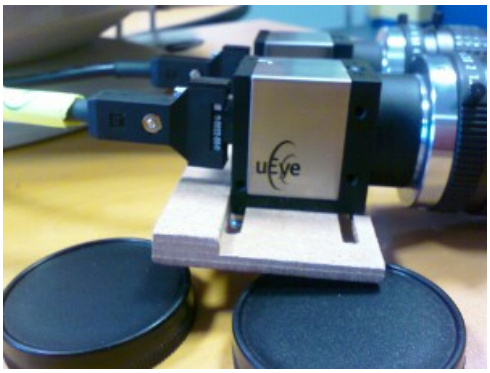


Figure 6.1 – Banc stéréoscopique uEye : le support bloquant la rotation des caméras

Les caméras étaient placées de manière fixe sur un pied d'appareil photo. Leur écartement (qui pouvait être réglé grâce à un déplacement horizontal le long du support) a été fixé à 63mm. Cette support avait été spécialement conçu pour que les caméras soient calées et ainsi empêcher le plus possible leur rotation verticale et horizontale.

L'expérimentation a validé les conditions de capture/affichage stéréoscopique, avec un panel important de personnes (hommes/femmes de tous les âges) qui a pu tester et apprécier la vision en relief temps réel de vidéo stéréoscopiques.

1. http://www.rntl-partage.fr/
2. http://www.ids-imaging.com/

6.1.3 Problème d'un utilisateur mobile : déformations de la vision - mouvements pseudoscopiques

Nous nous plaçons dans le cas d'une configuration stéréoscopique, dans laquelle les déplacements de l'utilisateur ne sont pas pris en compte lors de la capture et/ou affichage des images. Comme nous venons de le voir, l'utilisateur va percevoir les objets de la scène à leur échelle native, ainsi que leurs placements de manière correcte, s'il se trouve à la position exacte (ou proche, le cerveau étant tolérant vis à vis de petits déplacements [Fuchs 06]) de l'ortho-stéréoscopie.

S'il se déplace beaucoup de manière à sortir de la zone de perception ortho-stéréoscopique, il risque de percevoir des déplacements des objets les uns par rapport aux autres, ou bien des déformations de ceux-ci.

Cela va engendrer une perception perturbée ou déformée de la scène. Ce phénomène est appelé "mouvements pseudoscopiques", et est illustré sur la figure 6.2. On peut y voir deux déplacements différents de l'utilisateur.

Celui, perpendiculaire à l'axe de l'écran, va provoquer un changement dans la profondeur perçue de l'objet. Le second cas -lors d'un déplacement en translation le long de l'écran- va donner le sentiment que l'objet a changé de position.

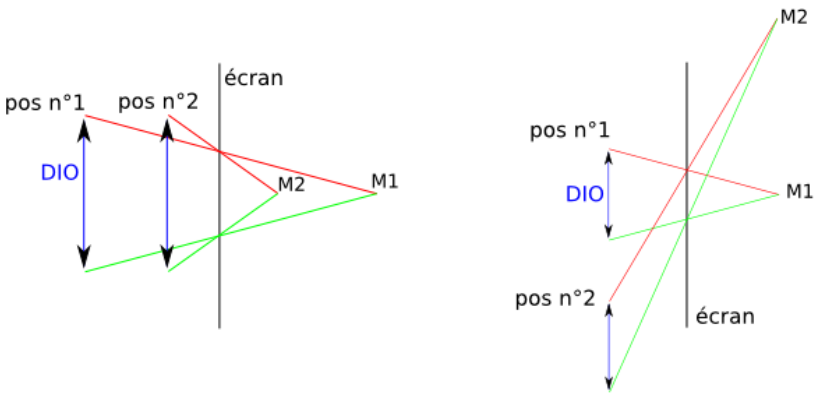


Figure 6.2 – Déformation de la perception d'un point lors du déplacement de l'utilisateur

Pour palier à ces problèmes deux solutions sont possibles.

Soit l'utilisateur reste dans la zone de perception ortho-stéréoscopique, ainsi les mouvements pseudoscopiques des objets sont minimes et lui sont à peine perceptibles.

Soit, l'utilisateur est mobile par rapport à l'écran (configurations immersives de type visiocubes, workbenchs,...), dans ce cas il va être nécessaire de récupérer à chaque instant la position de l'utilisateur grâce à des capteurs de mouvement [Leroy 09] [Wartell 02] et d'adapter ensuite les images projetées (pour plus de détails, se référer à [Cruz-Neira 93]).

6.2 Adapter les paramètres d'une restitution vidéo stéréoscopique

6.2.1 Mise en évidence du problème posé par le mouvement de la tête

Le problème se pose lorsque l'utilisateur sort des limites exactes de l'ortho-stéréoscopie. Si ses yeux ne se trouvent pas exactement à la même place devant l'écran que les caméras devant la scène (position nominale), la profondeur enregistrée et la profondeur perçue auront deux valeurs différentes. Donc la profondeur d'un objet, enregistrée par les caméras, sera perçue différemment par l'utilisateur. La figure 6.3 illustre ce problème sur un exemple. Ici, l'écart interoculaire ($E_L - E_R$) des yeux de l'utilisateur est inférieur à la distance de la ligne de base ($C_L - C_R$) du banc de caméras stéréoscopiques. A partir des points images (x_L et x_R), l'utilisateur percevra un profondeur restituée (P_E) plus importante que celle enregistrée par les caméras (P_C). D'autre part, il aura également le sentiment d'un déplacement horizontal du point.

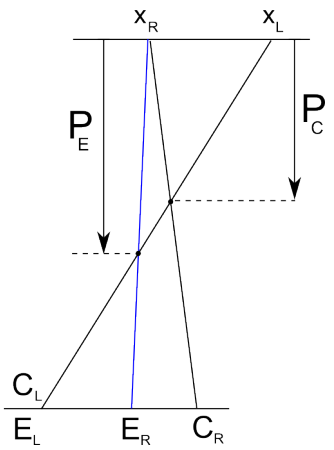


Figure 6.3 – Problème posé par le mouvement des yeux pour la perception de la profondeur restituée. La perception de la profondeur restituée (P_E) va être différente de celle enregistrée par les caméras (P_C). Un déplacement latéral sera également perçu.

Dans l'optique de restituer la profondeur perçue à la valeur réelle et sans générer de faux déplacements, ceci quelque soit la position des yeux de l'utilisateur, nous proposons une méthode de modification de la parallaxe à appliquer aux images affichées.

6.2.2 Adaptation des paramètres de la restitution

Afin de prendre en compte les mouvements de l'utilisateur, nous allons adapter la parallaxe à la position des yeux de l'utilisateur.

Deux cas peuvent être distingués :

– cas des yeux en position nominale, mais la distance interoculaire de l'utilisateur est différente de la distance de ligne de base. Il s'agit d'un cas simple qui va nous servir à illustrer le problème posé.

– cas des yeux écartés de la position nominale et la distance interoculaire de l'utilisateur est différente de la distance de ligne de base. Il s'agit d'un cas proche des conditions réelles.

Nous présentons dans les parties 6.2.2.a et 6.2.2.b, les modélisation associées à ces deux cas. Nous avons limité cette modélisation géométrique au cas des déplacements avant/arrière, gauche/droite. La modélisation en géométrie élémentaire de la rotation de la tête est trop complexe.

Les données d'entrée de notre modélisation vont être les variables suivantes, supposées connues à l'avance :

– la distance $C_R - C_L$ qui représente la longueur de la ligne de base.
– la distance $i_r - i_l$ qui représente la valeur de la parallaxe d'un point de la scène capturé par les caméras.
– la distance $e_{r,x} - e_{l,x}$ qui représente la distance interoculaire de l'utilisateur (voir section 5.2).

6.2.2.a Cas des yeux en position nominale, mais la distance interoculaire de l'utilisateur est différente de la distance de ligne de base

Introduisons les notations suivantes (représentées sur la Figure 6.4) dont nous allons nous servir au cours de cette modélisation :

– Z_s^e : distance de l'écran par rapport aux yeux
  $\hookrightarrow$ fournie par les mesures du système de capture de mouvements

– Z_s^c : distance de l'écran par rapport aux caméras
  $\hookrightarrow Z_s^c = Z_{P_o}$ (distance du plan à parallaxe nulle)

– Z_p^e : profondeur perçue par l'utilisateur
  $\hookrightarrow$ générée par la parallaxe des points images i_r et i_l

– Z_p^c : profondeur d'un point $\mathbf{P}$ de la scène enregistrée par les caméras lors de la capture. Il s'agit d'une valeur enregistrée et à restituer à l'identique

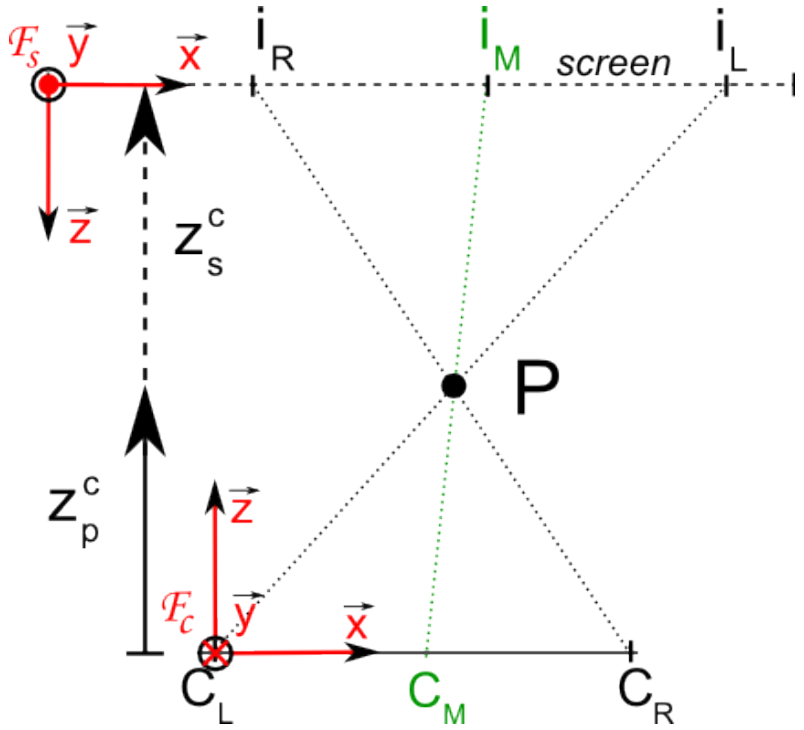
– $C_m = [c_{m,x}, c_{m,y}, c_{m,z}]$: point cyclopéen des caméras (milieu de la baseline) gauche C_l et droit C_r
  $\hookrightarrow$ fournie par la calibration du banc stéréoscopique

– $E_m = [e_{m,x}, e_{m,y}, e_{m,z}]$: point cyclopéen des yeux gauche E_l et droit E_r
  $\hookrightarrow$ fournie par les mesures du système de capture de mouvements, après calibration des yeux

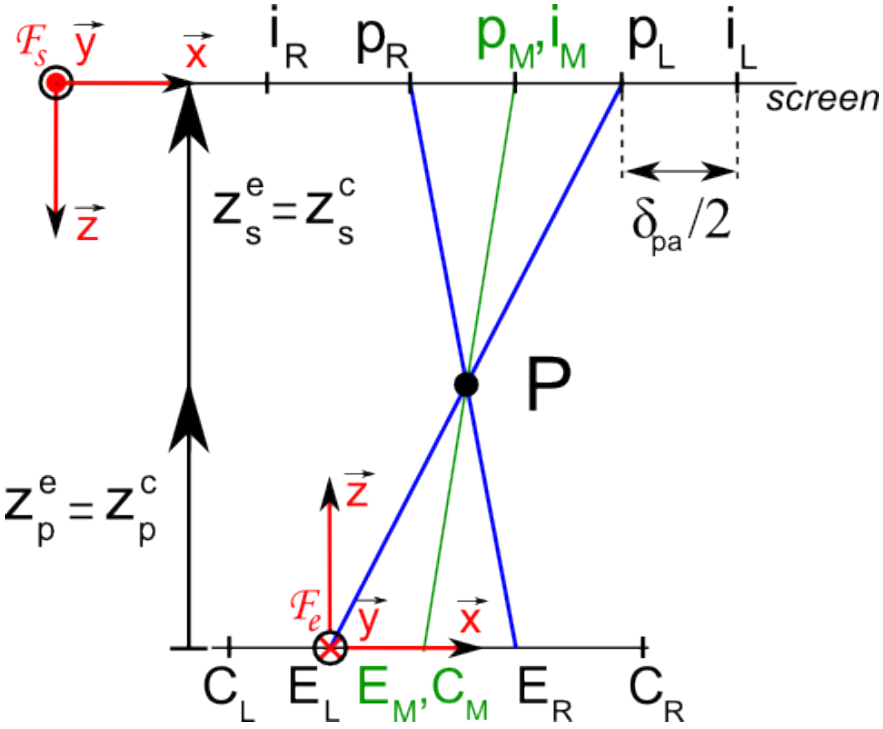
– $I_m = [i_{m,x}, i_{m,y}, i_{m,z}]$: équivalent à la moitié de la parallaxe des points images : point droite i_r point gauche i_l

Nous cherchons à déterminer la valeur de modification de la parallaxe δpa à appliquer aux images affichées de manière à maintenir la profondeur perçue identique à la profondeur enregistrée par les caméras, lorsqu'un utilisateur aura ses yeux situés à la même distance de l'écran que les caméras, mais avec son écart interoculaire inférieur/supérieur à la longueur de la ligne de base du banc stéréoscopique. Cela se traduit par :

$$Z_s^c = Z_s^e \qquad Z_p^c = Z_p^e$$



(a) Notations associées à la capture/restitution avec un banc stéréoscopique



(b) Notations associées à la perception de la profondeur par un utilisateur

Figure 6.4 – Cas des yeux en position nominale, mais la distance interoculaire de l'utilisateur est différente de la distance de ligne de base

Du point de vue géométrique, et en utilisant les variables de la figure 6.4, nous pouvons en déduire les relations suivantes :

$$\frac{Z_s^c - Z_p^c}{Z_p^c} = \frac{i_{l,x} - i_{r,x}}{C_r - C_l} \tag{6.1}$$

ainsi que :

$$\frac{Z_s^e - Z_p^e}{Z_p^e} = \frac{p_{l,x} - p_{r,x}}{e_{r,x} - e_{l,x}} \tag{6.2}$$

$$\Leftrightarrow p_{l,x} - p_{r,x} = (e_{r,x} - e_{l,x}) \, \frac{(Z_s^e - Z_p^e)}{Z_p^e}$$

En substituant avec l'équation 6.1, on obtient :

$$p_{l,x} - p_{r,x} = \frac{(e_{r,x} - e_{l,x}) \, (i_{l,x} - i_{r,x})}{(C_r - C_l)} \tag{6.3}$$

Pour compenser le décalage horizontal des yeux dû à une DIO plus faible/plus large que la ligne de base des caméras, nous allons modifier la valeur de la parallaxe avec $\frac{\delta pa}{2}$

$$\frac{\delta pa}{2} = \frac{(i_{l,x} - i_{r,x}) - (p_{l,x} - p_{r,x})}{2}$$

$$\Leftrightarrow \frac{\delta pa}{2} = \frac{(i_{l,x} - i_{r,x})}{2} \left(1 - \frac{e_{r,x} - e_{l,x}}{C_r - C_l}\right) \tag{6.4}$$

Les nouvelles coordonnées $(i_{r_{new}} \; i_{l_{new}})$ des points images $(i_r \; i_l)$ sont donc :

$$\begin{cases} i_{r_{new},x} = i_{r,x} + \dfrac{\delta pa}{2} \\ \\ i_{l_{new},x} = i_{l,x} - \dfrac{\delta pa}{2} \end{cases}$$

6.2.2.b Cas des yeux écartés de la position nominale, avec une distance interoculaire de l'utilisateur différente de la distance de ligne de base

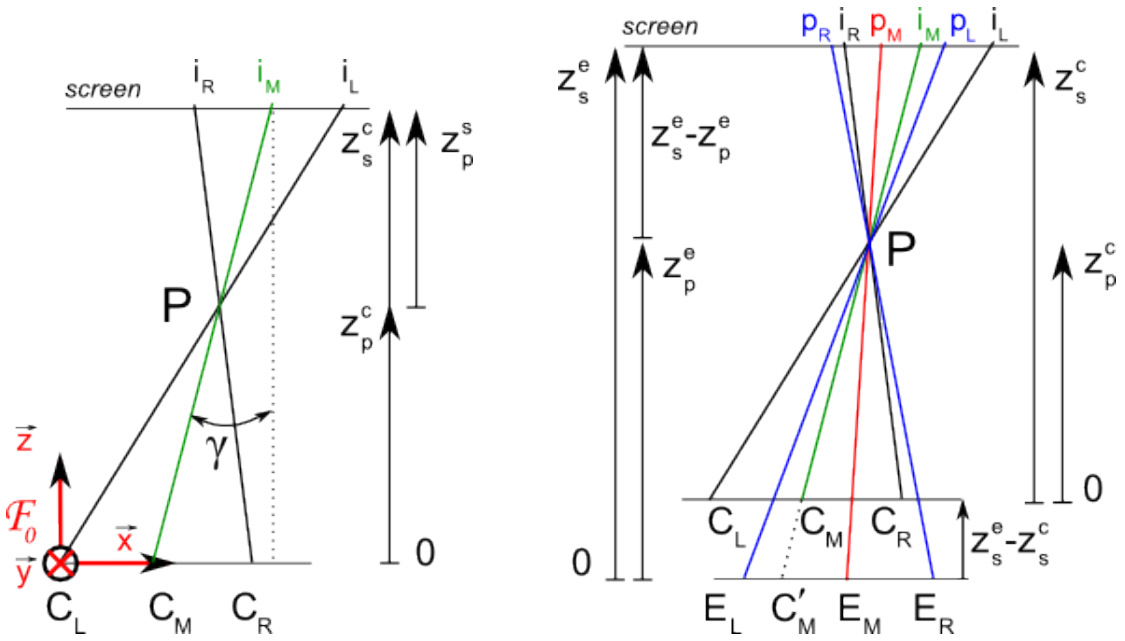
Introduisons la notation suivante (représentée sur la Figure 6.5) dont nous allons nous servir au cours de cette modélisation, en complément des notations déjà définies :

- $C'_m = [c'_{m,x}, c'_{m,y}, c'_{m,z}]$: projection du point cyclopéen des caméras C_m sur la ligne de base des yeux

Dans ce second cas, la tête de l'utilisateur s'est déplacée par rapport à la position de référence, tant en profondeur que latéralement (cf. Figure 6.5)

L'équation 6.1 nous donne :

$$\frac{Z_s^c - Z_p^c}{Z_p^c} = \frac{(i_{l,x} - i_{r,x})}{(C_R - C_L)}$$



(a) Repère associé au banc stéréoscopique

(b) Notations utilisées

Figure 6.5 – Cas yeux écartés de la position nominale et écart interoculaire ≠ ligne de base

d'où

$$Z_p^c = \frac{Z_s^c(C_R - C_L)}{(C_R - C_L) + (i_{l,x} - i_{r,x})}$$

Nous obtenons :

$$Z_p^c = \frac{Z_s^c(\delta C)}{(\delta C) - (\delta I)} \tag{6.5}$$

avec $\delta C = C_R - C_L$ et $\delta I = i_{r,x} - i_{l,x}$

Nous avons également :

$$\frac{Z_s^e - Z_p^e}{Z_p^e} = \frac{(p_{l,x} - p_{r,x})}{(e_{r,x} - e_{l,x})}$$

Nous obtenons :

$$\frac{Z_s^e - Z_p^e}{Z_p^e} = \frac{-\delta p}{\delta dio} \tag{6.6}$$

avec $\delta p = (p_{r,x} - p_{l,x})$ et $\delta dio = (e_{r,x} - e_{l,x})$

De plus,

$$Z_p^e = Z_p^c + (Z_s^e - Z_s^c) \tag{6.7}$$

L'équation 6.6 donne :

$$Z_s^e\ \delta dio\ =\ Z_p^e\ (\delta dio\ -\ \delta p)$$

en utilisant l'équation 6.7 :

$$Z_s^c \left(\frac{\delta C}{\delta C\ -\ \delta I}\ -\ 1\right)\left(\frac{\delta dio - \delta p}{\delta p}\right) = Z_s^e$$

Lors de la capture vidéo stéréoscopique, on fixe la position du plan de parallaxe nulle, dans la scène réelle (cf. section 4.2.3). Cette position ici est représentée par la variable Z_{p_0}. Ainsi, comme

$$Z_s^c = Z_{p0} \tag{6.8}$$

Nous obtenons :

$$Z_s^e = Z_{p_0} \left(\frac{\delta I}{\delta C - \delta I}\right)\left(\frac{\delta dio - \delta p}{\delta p}\right) \tag{6.9}$$

Nous pouvons maintenant écrire Z_p^e sous la forme :

$$Z_p^e = \frac{\delta I \; Z_{p_0} \; \delta dio}{\delta p \; (\delta C - \delta I)} \tag{6.10}$$

Afin de déterminer la modification de parallaxe à apporter aux images, on exprime dans la figure 6.5 :

$$\tan\gamma = \frac{i_{m,x} - c_{m,x}}{Z_s^c} \tag{6.11}$$

ainsi que, d'après la figure 6.5b :

$$c_{m',x} = c_{m,x} + (Z_s^e - Z_s^c).\tan\gamma \tag{6.12}$$

En substituant (6.11) dans (6.12), on obtient

$$c_{m',x} = c_{m,x} \; \left(2 - \frac{Z_s^e}{Z_s^c}\right) + i_{m,x} \; \left(\frac{Z_s^e}{Z_s^c} - 1\right) \tag{6.13}$$

La figure 6.5 nous donne :

$$\frac{e_{m,x} - c_{m',x}}{i_{m,x} - p_{m,x}} = \frac{Z_p^e}{Z_s^e - Z_p^e} \Leftrightarrow p_{m,x} = i_{m,x} - (e_{m,x} - c_{m',x}) \left(\frac{Z_s^e}{Z_p^e} - 1\right)$$

En substituant avec l'équation 6.13 :

$$\Leftrightarrow p_{m,x} = i_{m,x} - \left(e_{m,x} - c_{m,x} \; \left(2 - \frac{Z_s^e}{Z_s^c}\right) + i_{m,x} \; \left(\frac{Z_s^e}{Z_s^c} - 1\right)\right) \; \left(\frac{Z_s^e}{Z_p^e} - 1\right) \tag{6.14}$$

Posons $A = \left(\frac{Z_s^e}{Z_p^e} - 1\right)$, $B = \left(\frac{Z_s^e}{Z_s^c} - 1\right)$ et $D = \left(2 - \frac{Z_s^e}{Z_s^c}\right)$

En utilisant les équations 6.9 et 6.10

$$A = \frac{-\delta p}{\delta dio} \tag{6.15}$$

Egalement, en utilisant les équations 6.8 et 6.9

$$B = \frac{\delta dio \; \delta I - \delta p \; \delta C}{(\delta C - \delta I) \; \delta p} \tag{6.16}$$

De même,

$$D = \frac{2\ \delta C\ \delta p - \delta I\ (\delta dio + \delta p)}{(\delta C - \delta I)\ \delta p} \tag{6.17}$$

L'équation 6.14 peut donc s'écrire :

$$p_{m,x} = i_{m,x}\ (1 - A\ B) - e_{m,x}\ A + c_{m,x}\ A\ D$$

Soit, avec 6.15, 6.16, et 6.17

$$p_{m,x} = i_{m,x}\ \frac{\delta C\ (\delta dio - \delta p)}{\delta dio\ (\delta C - \delta I)} + e_{m,x}\ \frac{\delta p}{\delta dio} + c_{m,x}\ \frac{\delta I\ (\delta dio + \delta p) - 2\ \delta C\ \delta p}{(\delta C - \delta I)\ \delta dio}$$

Or nous avons

$$\delta I = \delta p + \delta pa$$

Nous pouvons donc écrire

$$\delta p = \delta I - \delta pa \tag{6.18}$$

Calculons maintenant $i_{m,x} - p_{m,x}$ en utilisant 6.18

$$i_{m,x} - p_{m,x} = -i_{m,x}\ \frac{\delta C\ (\delta I\ -\ \delta pa)\ -\ \delta dio\ \delta I}{(\delta C\ -\ \delta I)\ \delta dio}$$

$$-e_{m,x}\ \frac{(\delta I\ -\ \delta pa)(\delta C\ -\ \delta I)}{(\delta C\ -\ \delta I)\ \delta dio}$$

$$-c_{m,x}\ \frac{2\ \delta C\ (\delta I\ -\ \delta pa)\ -\ \delta I\ (\delta dio\ +\ \delta I\ -\ \delta pa)}{(\delta C\ -\ \delta I)\ \delta dio} \tag{6.19}$$

Or

$$\delta pa = i_{m,x} - p_{m,x} \tag{6.20}$$

$$\delta pa\ =\ \frac{\delta I\ (\ i_{m,x}(\delta C\ -\delta dio)\ -\ e_{m,x}\ (\delta C - \delta I)\ +\ c_{m,x}\ (2\ \delta C\ -\ \delta I\ -\ \delta dio))}{c_{m,x}\ (2\ \delta C\ -\ \delta I)\ -\ e_{m,x}\ (\delta C - \delta I)\ +\ i_{m,x}\ \delta C\ +\ (\delta C\ -\ \delta I)\ \delta dio} \tag{6.21}$$

Les nouvelles coordonnées $(i_{r_{new}}\ i_{l_{new}})$ des points images $(i_r\ i_l)$ sont donc :

$$\begin{cases} i_{r_{new},x} = i_{r,x} + \dfrac{\delta pa}{2} \\ \\ i_{l_{new},x} = i_{l,x} - \dfrac{\delta pa}{2} \end{cases}$$

6.2.3 Solution proposée

La projection stéréoscopique d'un point 3D en deux points images permet à un utilisateur de percevoir la profondeur de cet objet. La section précédente a présenté la modélisation des changements de parallaxe à effectuer sur ces deux points image, afin de maintenir une restitution de profondeur à échelle constante malgré les mouvements de l'utilisateur.

Cependant, en général une scène est un ensemble d'objets situés à des profondeurs différentes du banc stéréoscopique. Ainsi, modifier la restitution d'une seule profondeur afin qu'elle apparaisse correctement à l'utilisateur n'est pas suffisant. Il est nécessaire de prendre en compte toutes les profondeurs de la scène, lors de la restitution. Mais on ne peut pas modifier la restitution de la parallaxe d'une image, pour ajuster la perception d'une profondeur donnée, sans modifier de la même manière toutes les autres profondeurs.

Afin de résoudre ce problème nous avons mis au point une méthode basée sur le découpage des images selon une segmentation en profondeur. Cette technique est parfois utilisée lors de la conversion de films de la 2D à la 3D (post-traitement), mais rarement dans le cadre d'une transmission stéréoscopique temps réel.

Dans une première partie nous reviendrons succinctement sur la génération de cartes de profondeur à partir d'un banc stéréoscopique. Ensuite, nous présentons nos développements sur le découpage des images stéréoscopiques en couches distinctes à partir d'une carte de profondeur ainsi que sur le déplacement de ces couches afin de s'adapter aux mouvements de l'utilisateur.

6.2.3.a Obtention d'une carte de profondeur

Un banc stéréoscopique permet l'obtention de deux images représentant deux points de vue différents d'une scène réelle. Grâce à la méthode "Shape from stereo" vue en section 2.1, nous allons pouvoir obtenir des informations sur la profondeur des objets filmés par rapport aux caméras. Dans cette partie, nous allons détailler l'algorithme employé afin de réaliser cette opération.

Nous nous sommes basés sur les travaux de synthèse dans ce domaine de l'université de Middlebury [3] afin de récupérer une segmentation de la scène en profondeur. Cette segmentation nous permet de déduire la position de chaque objet par rapport au banc stéréoscopique.

Nous sommes également à même de déterminer la modification de parallaxe à apporter sur deux points images, pour conserver une profondeur donnée suite aux mouvements de l'utilisateur (cf. section 6.2.2.b).

Nous allons détailler dans la section suivante la manière dont nous nous sommes servis de ces données pour parvenir à conserver une profondeur constante malgré les mouvements en translation de l'utilisateur.

6.2.3.b Découpage des images en couche

Une segmentation de la scène en profondeur représente un ensemble de couches de profondeur, chacune rassemblant les différents objets situés à une même distance des

3. `http://vision.middlebury.edu`

caméras. Afin de pouvoir modifier les profondeurs perçues d'une scène, il est nécessaire d'ajuster la parallaxe correspondant à chacune des couches de profondeur.

Pour cela, nous avons choisi de découper nos images, en fonction de la carte de profondeur que nous avons générée, à partir des images vidéo stéréoscopiques.

Tout d'abord, en parcourant l'ensemble des segmentations en profondeur, nous avons construit une table de référence des valeurs de profondeur. Les valeurs référencées dans la carte de profondeur sont codées en couleurs de gris (de 0 à 255, à valeur identique sur les trois composantes RGB). La table de référence contient également le nombre de pixels associé à chaque valeurs de profondeur.

Notre programme de découpage des images prend en entrée deux images qui sont l'image vidéo, et la carte de profondeur associée, ainsi que la table de référence des valeurs de profondeur, et une valeur de seuil.

Avant la génération des images découpées, nous pouvons sélectionner les couches de profondeur avec lesquelles nous souhaitons travailler, grâce à la valeur de seuil qui permet de garder les couches ayant un nombre minimum de pixels suffisant. Par exemple, afin de récupérer toutes les couches de profondeur, une valeur de seuil sera fixée à 0 (toutes les couches de profondeur même les plus petites sont importantes), alors qu'une valeur à 10000 permettra de ne prendre en compte que les couches d'importance majeure.

Pour chaque profondeur ainsi sélectionnée, nous générons un masque de manière à obtenir une sous-image. Cette sous-image contient les parties de l'image qui représentent les objets à la profondeur en cours de traitement (cf. Figure 6.6). Pour obtenir ce résultat, nous allons parcourir l'image de base, et comparer la valeur de la profondeur correspondante à la valeur de la profondeur du pixel en cours. Si la valeur est celle qui est traitée, nous inscrivons la couleur de ce pixel dans la sous-image, sinon nous passons au pixel suivant.

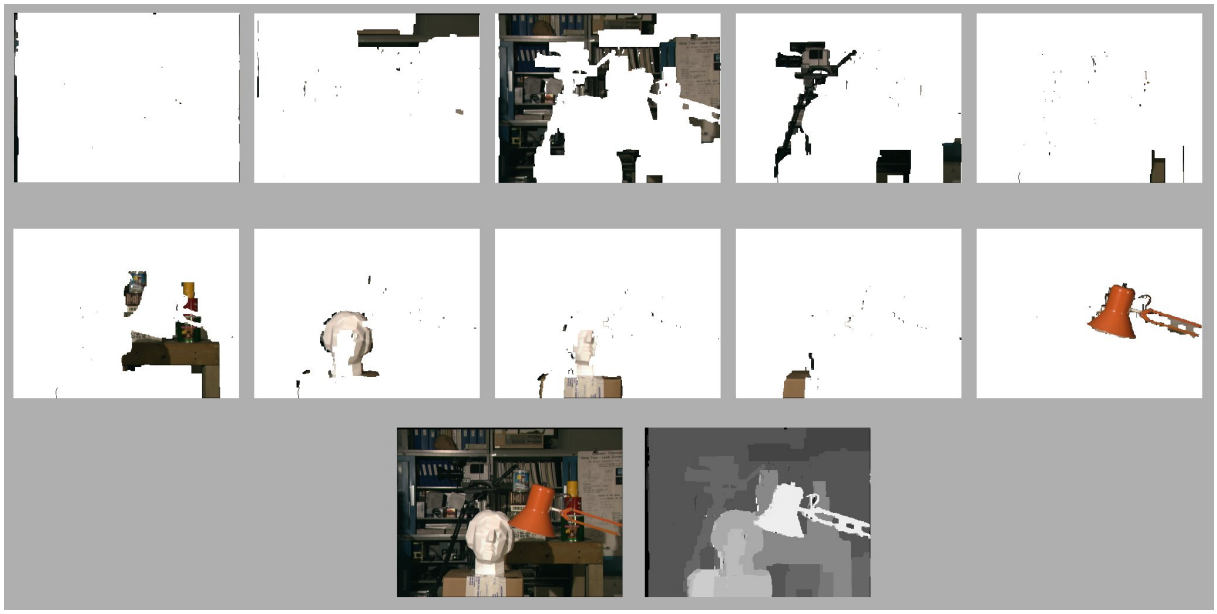


Figure 6.6 – Exemple d'un découpage en dix couches (deux premières lignes d'image) d'une image (en bas à gauche) à partir de la carte de profondeur associée (bas à droite).

Ce processus est à répéter également pour l'image droite, afin d'obtenir ses sous-images. Enfin, une fois les différentes parties de l'image isolées en sous-images, nous avons la liberté de les déplacer à notre guise (un exemple du résultat obtenu pour un déplacement à droite de l'utilisateur est présenté Figure 6.7). Nous avons maintenant la possibilité de régler la parallaxe par couche de même profondeur, en fonction des mouvements de l'utilisateur comme décrit section 6.2.2.b. En outre, il est également possible de régler la position du plan à parallaxe nulle de cette manière.

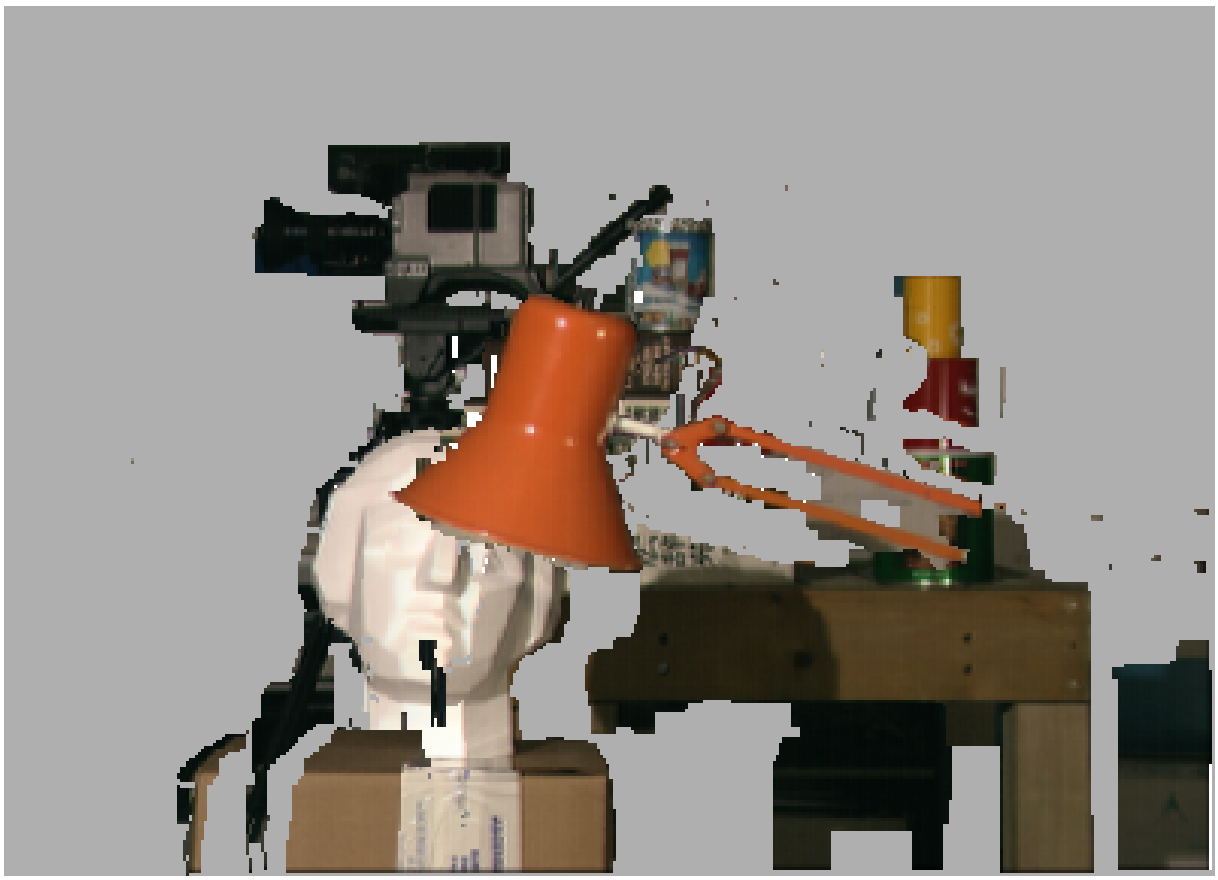


Figure 6.7 – Exemple d'un déplacement des sous-images, lors d'un déplacement à droite de l'utilisateur. La caméra s'est déplacée à droite, alors que la lampe s'est déplacée à gauche. La tête n'a pas bougée car elle se situe sur le plan à parallaxe nulle.

Ce chapitre a énuméré les modifications à apporter aux différentes parties d'une image, issue d'un couple stéréoscopique destiné à être projeté sur un système de Réalité Virtuelle. Ceci afin de conserver une perception par l'utilisateur des profondeurs correspondantes à celles enregistrées lors de la capture de la scène.

En particulier, le changement de parallaxe à apporter à chaque couple de sous-parties de l'image est modélisé pour permettre la conservation des proportions originales de la scène en taille et profondeur, lors que l'utilisateur est fixe ou en mouvement.

Chapitre 7

Conclusions & perspectives

En introduction, nous avions présenté l'intérêt croissant de nombreux domaines pour des simulations numériques qui incluent des éléments du monde réel. Parmi les différentes technologies disponibles, la capture vidéo stéréoscopique permet de capturer ces éléments sous forme d'images vidéo, ce qui présente un avantage important pour la restitution, sur des systèmes de projection en relief, des données enregistrées. Cependant, de nombreuses contraintes restreignent l'utilisation de cette technologie. Il s'agit principalement de limitations des configurations de capture et des conditions de restitution de ces images.

Dans l'optique de s'affranchir de certaines de ces contraintes, et de maîtriser la restitution de vidéos stéréoscopiques sur des systèmes de Réalité Virtuelle, nous avons étudié en détails plusieurs aspects de la chaîne de transmission stéréoscopique. Nous avons, tout d'abord, modélisé complètement cette chaîne ainsi que les caméras que nous avons utilisées.

Notre travail de modélisation sépare les effets liés au matériel utilisé lors de la capture et de l'affichage, des effets causés par la transmission stéréoscopique elle même.

Nous avons volontairement choisi, de concentrer notre travail au cas de la transmission d'images stéréoscopiques non déformées, en modélisant la chaîne de transmission stéréoscopique avec des caméras canoniques (caméras simplifiées car calibrées). La modélisation est donc plus claire : pour chaque caméra, on ne s'intéresse uniquement qu'à l'action essentielle de la projection centrale.

Les images stéréoscopiques étant par nature déformées, nous avons dans un premier temps traité le problème de la rectification de ces images, en amont de la modélisation. Nous avons développé, pour effectuer cette rectification, un algorithme sur GPU. Cette approche issue des travaux menés dans le domaine de la vision par ordinateur est très innovante dans le domaine de la vidéo stéréoscopique en relief. Notre algorithme permet de rectifier en temps réel un flux d'images vidéo stéréoscopiques (2 x 35Hz), en haute résolution (1600 x 1200 pixels) et autorise la conservation d'un nombre d'images par seconde important, utile pour d'autres opérations sur ces images.

Le problème de l'obtention d'images rectifiées étant résolu, le deuxième volet de notre travail a été consacré à la restitution maîtrisée de ces images.

Lors de l'utilisation de systèmes de visualisation en relief, la valeur moyenne de la distance interoculaire (DIO) humaine est souvent utilisée pour la projection. La valeur de cette distance varie d'un être humain à un autre (jusqu'à 2.5 cm de différence aux limites). Un utilisateur dont la DIO est très différente de la valeur moyenne peut percevoir

une profondeur différente de la profondeur réelle, et des mouvements erronés des objets peuvent également apparaître.

Pour éviter les inconvénients ci-dessus, il est nécessaire de paramétrer les systèmes de restitution par la DIO de chaque utilisateur. Nous avons développé une méthode de calibration de la distance interoculaire, basée sur une série de quelques mesures. Notre méthode de calibration repose sur des alignements entre les centres d'une mire virtuelle 2D et d'une mire réelle plane, et se révèle plus simple que les méthodes basées sur le recalage de grands cubes 3D.

Les résultats de cette calibration nous fournissent une information rigoureuse (précision inférieure à 0.5 mm) de la distance interoculaire de chaque utilisateur, et de la position de ses yeux, par rapport au marqueur de position de sa tête.

Enfin, cette mesure calculée de la DIO nous a permis d'adapter, en fonction des déplacements de l'utilisateur devant un système de Réalité Virtuelle, la restitution d'images issues d'un banc stéréoscopique fixe. Ceci permettant de remplir les conditions de restitution particulières nécessaires à la conservation de l'échelle et du placement des objets les uns par rapport aux autres.

Dans le but de restituer la profondeur perçue à la valeur réelle, quelque soit la position des yeux de l'utilisateur, nous avons modélisé la modification de la parallaxe à appliquer aux images affichées. Cette modélisation nous permet également de régler la position du plan à parallaxe nulle (position virtuelle de l'écran dans la scène filmée).

Nous avons mis au point une méthode basée sur le découpage des images selon une segmentation en profondeur de la scène. Les algorithmes de génération de carte de profondeur à partir d'images stéréoscopiques n'étant pas encore temps réel, nous avons limité notre expérimentation en générant une carte de profondeur à intervalles constants.

En conclusion, ce travail a permis de progresser dans le domaine de la restitution maîtrisée de la profondeur et ouvre la voie à de nouvelles perspectives dans la visualisation d'images stéréoscopiques.

A l'issue de ce travail, plusieurs questions nous paraissent demeurer encore largement ouvertes. Ces questions concernent particulièrement la segmentation en profondeur des scènes à restituer.

En premier lieu, la précision des cartes de profondeur obtenues en temps réel, devra être améliorée.

D'autre part, seule une modélisation complète des mouvements en rotation/translation du spectateur permettra de maîtriser totalement la restitution de la profondeur. Les méthodes basées sur la géométrie élémentaire se révéleront probablement trop limitées pour atteindre ce dernier objectif et il sera probablement nécessaire d'utiliser celles de la géométrie projective ou algébrique.

Enfin, la restitution d'images vidéo segmentées en profondeur peut parfois créer une gêne pour le spectateur, par exemple si celui-ci perçoit des vides entre les sous-images. Des techniques de lissage entre ces sous-images devront être développées pour atténuer cette gêne.

L'actuel engouement pour la projection de contenu en relief, que ce soit pour le cinéma ou la télévision, mais également pour la Réalité Virtuelle, demande à ce que les effets indésirables liés à ce type de visualisation soient très fortement diminués voire supprimés.

A la télévision, le direct est une contrainte forte qui impose un traitement des images

le plus rapide possible avant diffusion. La rectification en temps-réel des déformations de l'image apparaît comme primordiale. Cela permettra par exemple de diffuser en direct des évènements sportifs, ou de divertissement avec une plus grande liberté dans les mouvements que l'on peut filmer, sans provoquer de gène chez le spectateur.

Dans le domaine de l'imagerie numérique, les capacités de calcul des ordinateurs de plus en plus importantes permettent chaque jour de réaliser des simulations plus réalistes. Plus particulièrement en Réalité Virtuelle, pour certaines applications, telles que la maintenance d'installations, ou la collaboration à distance, il sera possible d'enrichir la simulation avec des données du monde réel. On perçoit ici les balbutiements d'une nouvelle discipline : la *Virtualité Augmentée*. Dans ce cadre, la maîtrise de la capture de ces données ainsi que leur restitution sans modification d'échelle se révèlent être des pré-requis essentiels. Notre travail a permis de progresser dans l'obtention de certains de ces pré-requis.

Bibliographie

[Allard 06] J. Allard, J.-S. Franco, C. Ménier, E. Boyer & B. Raffin. *The GrImage Platform : A Mixed Reality Environment for Interactions.* Dans 4th International Conference on Computer Vision Systems, ICVS'06, pages 46–46, New York City États-Unis d'Amérique, 2006. IEEE.

[Allison 04] R. S. Allison. *The camera convergence problem revisited.* Dans Andrew J. Woods, John O. Merritt, Stephen A. Benton & Mark T. Bolas, editeurs, Proceedings of the SPIE, volume 5291, pages 167–178. Stereoscopic Displays and Virtual Reality Systems XI, 2004.

[Andriot 92] C. Andriot. *Automatique des systèmes téléopères avec retour d'effort : limitation des performances.* Thèse de doctorat, Université de Paris 6, 1992.

[Boev 08] A. Boev, D. Hollosi & A. Gotchev. *Classification of stereoscopic artefacts.* Rapport technique, MOBILE3DTV project, 2008.

[Bouguet 98] J.-Y. Bouguet & P. Perona. *3D Photography on Your Desk.* Dans ICCV '98 : Proceedings of the Sixth International Conference on Computer Vision, page 43, Washington, DC, USA, 1998. IEEE Computer Society.

[Bouguet 04] J. Bouguet. *Complete Camera Calibration Toolbox for Matlab.* Rapport technique, `http://www.vision.caltech.edu/bouguetj/calibdoc/index.html`, 2004.

[Bourke 99] Paul Bourke. *Calculating Stereo Pairs.* `http://local.wasp.uwa.edu.au/~pbourke/miscellaneous/stereographics/stereorender/`, 1999.

[Bovic 05] A.. Bovic. Handbook of image and video processing (communications, networking and multimedia) 2nd edition. Academic Press, 2005.

[Cowan 07] Matt Cowan. *REAL-D.* Dans SMPTE Tech Conf, 2007.

[Cruz-Neira 93] C. Cruz-Neira, D. Sandin & T. DeFanti. *Surround-screen projection-based virtual reality : the design and implementation of the CAVE.* Dans SIGGRAPH '93 : Proceedings of the 20th annual conference on Computer graphics and interactive techniques, pages 135–142, New York, NY, USA, 1993. ACM.

[Curless 00] Brian Curless. *SIGGRAPH 2000 Course on 3D Photography*, 2000.

[Dellaert 00] F. Dellaert, S. Seitz, C. Thorpe & S. Thrun. *Structure from Motion without Correspondence.* Dans IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'00), June 2000.

[Dodgson 04] N. Dodgson. *Variation and extrema of human interpupillary distance, in Stereoscopic Displays and Virtual Reality Systems.* Dans Proc. SPIE 5291, pages 36–46, 2004.

[Ernadotte 97] D. Ernadotte. *L'intégration d'objets virtuels dans des scènes réelles, application aux tâches de télé-opération.* Thèse de doctorat, Ecole des Mines de Paris, 1997.

[Faugeras 93] O. Faugeras. Three-dimensional computer vision : A geometric viewpoint. MIT Press, 1993.

[Ferreira 02] J. F. Ferreira, J. Lobo & J ? Dias. *Tele-3D : Developing a Handheld Scanner Using Structured Light Projection.* 3D Data Processing Visualization and Transmission, International Symposium on, vol. 0, page 788, 2002.

[Fuchs 06] P. Fuchs. Le traité de la réalité virtuelle. Presse de l'Ecole des Mines de Paris, Février 2006.

[Fusiello 00] A. Fusiello, E. Trucco & A. Verri. *A compact algorithm for rectification of stereo pairs.* Mach. Vis. Appl., vol. 12, no. 1, pages 16–22, 2000.

[Goslin 09] F. Goslin, F. Schramm, C. Andriot, A. Bouchet & S. Richir. *High-resolution stereo video rectification through a cost-efficient real-time GPU implementation using intrinsic and extrinsic camera parameters.* Dans IEEE SSCI - Computational Intelligence in Virtual Environments, 2009.

[Hecht 01] E. Hecht. Optics (4th edition). Addison Wesley, 4 edition, August 2001.

[Heikkila 97] J. Heikkila & O. Silven. *A Four-step Camera Calibration Procedure with Implicit Image Correction.* Dans Conference on Computer Vision and Pattern Recognition (CVPR '97), page 1106, Washington, DC, USA, 1997. IEEE Computer Society.

[Holliman 04] N. S. Holliman. *Mapping perceived depth to regions of interest in stereoscopic images.* Dans Proceedings of SPIE : Stereoscopic displays and virtual reality systems XI, pages 117–128, 2004.

[Jarvis 83] R.A. Jarvis. *A Laser Time-of-Flight Range Scanner for Robotic Vision.* PAMI, vol. 5, no. 5, pages 505–512, September 1983.

[Johnson 06] T. Johnson, F. Gyarfas, R. Skarbez, P. Quirk, H. Towles & H. Fuchs. *Multi-Projector Image Correction on the GPU.* Dans 2006 Workshop on Edge Computing Using New Commodity Architectures (EDGE 2006), May 2006.

[Jones 01] G. Jones, D. Lee, N. Holliman & D. Ezra. *Controlling perceived depth in stereoscopic images.* Dans Stereoscopic Displays and Virtual Reality Systems VIII, pages 200–1, 2001.

[Jones 07] A. Jones, I. McDowall, H. Yamada, M. Bolas & P. Debevec. *Rendering for an interactive 360 ° light field display.* Dans ACM SIGGRAPH 2007, page 40, New York, NY, USA, 2007. ACM.

[Leroy 09] L. Leroy, P. Fuchs, A. Paljic & G. Moreau. *Some experiments about shape perception in stereoscopic displays.* IS&T/SPIE Symposium on

Electronic Imaging, vol. Proceedings of SPIE Vol. 7237, no. 7237-45, pages –, January 2009.

[Levoy 00] M. Levoy, K. Pulli, B. Curless, S. Rusinkiewicz, D. Koller, L. Pereira, M. Ginzton, S. Anderson, J. Davis, J. Ginsberg, J. Shade & D. Fulk. *The digital Michelangelo project : 3D scanning of large statues.* Dans SIGGRAPH, pages 131–144, 2000.

[Lipton 82] L. Lipton. Foundations of the stereoscopic cinema : A study in depth. Van Nostrand Reinhold Company, New York, 1982.

[Lipton 97] L. Lipton. *Stereographics developers handbook.* Rapport technique, Stereographics Corporation, 1997.

[Mairal 05] J. Mairal, R. Keriven & A. Chariot. *A GPU implementation of variational stereo.* Rapport technique, Ecole Nationale des Ponts et Chaussées, 2005.

[Mairal 06] J. Mairal, R. Keriven & A. Chariot. *Fast and Efficient Dense Variational Stereo on GPU.* Dans Proceedings of the Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT'06), pages 97–104, Washington, DC, USA, 2006. IEEE Computer Society.

[Maman 98] D. Maman. *Recalage de modèles tridimensionnels sur des images réelles, application en réalité augmentée : modélisation interactive pour la télé-opération.* Thèse de doctorat, Ecole des Mines de Paris, 1998.

[Matusik 04] W. Matusik & H. Pfister. *3D TV : A Scalable System for Real-Time Acquisition, Transmission, and Autostereoscopic Display of Dynamic Scenes.* ACM Transactions on Graphics, vol. 23, pages 814–824, 2004.

[McVeigh 96] J. S. McVeigh, M. Siegel & A. Jordan. *Algorithm for automated eye strain reduction in real stereoscopic images and sequences.* Human Vision and Electronic Imaging, vol. 2657, pages 307 – 316, February 1996.

[Meesters 04] L. Meesters, W. IJsselsteijn & P. Seuntiëns. *A survey of perceptual evaluations and requirements of three-dimensional TV.* Circuits and Systems for Video Technology, IEEE Transactions on, vol. 14, no. 3, pages 381–391, 2004.

[Nozick 06] Vincent Nozick. *Méthodes de rendu à base de vidéos et applications à la réalité Virtuelle.* Thèse de doctorat, Université Paris-Est, Marne-la-Vallée, France, 2006.

[Pharr 05] M. Pharr & R. Fernando. Gpu gems 2 : Programming techniques for high-performance graphics and general-purpose computation. Addison-Wesley Professional, March 2005.

[Pollefeys 99] M. Pollefeys. *Self-calibration and metric 3D reconstruction from uncalibrated image sequences.* Thèse de doctorat, Katholieke Universiteit Leuven, 1999.

[Prehn 07] Sebastian Prehn. *GPU Stereo Vision*, Dec 2007.

[Redert 02] A. Redert, M. Op de Beeck, C. Fehn, W. IJsselsteijn, M. Pollefeys, L. Van Gool, E. Ofek, I. Sexton & P. Surman. *ATTEST : Advanced*

Three-dimensional Television System Technologies. 3D Data Processing Visualization and Transmission, International Symposium on, vol. 0, page 313, 2002.

[Rocchini 02] C. Rocchini, P. Cignoni, C. Montani, P. Pingi & R. Scopigno. *A low cost 3D scanner based on structured light*. EG Proceedings, vol. 20, pages 299–308, 2002.

[Rusinkiewicz 02] S. Rusinkiewicz, O. Hall-Holt & M. Levoy. *Real-Time 3D Model Acquisition*. ACM Transactions on Graphics (Proc. SIGGRAPH), vol. 21, no. 3, pages 438–446, Juillet 2002.

[Saito 08] H. Saito, H. Kimura, S. Shimada, T. Naemura, J. Kayahara, S. Jarusirisawad, V. Nozick, H. Ishikawa, T. Murakami, J. Aoki, A. Asano, T. Kimura, M. Kakehata, F. Sasaki, H. Yashiro, M. Mori, K. Torizuka & K. Ino. *Laser-plasma scanning 3D display for putting digital contents in free space*. Dans Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, volume 6803 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, Mars 2008.

[Scharstein 02] D. Scharstein & R. Szeliski. *A taxonomy and evaluation of dense two-frame stereo correspondence algorithms*. International Journal of Computer Vision, vol. 47, pages 7–42, 2002.

[Scharstein 03] D. Scharstein & R. Szeliski. *High-Accuracy Stereo Depth Maps Using Structured Light*. Computer Vision and Pattern Recognition, IEEE Computer Society Conference on, vol. 1, page 195, 2003.

[Schreer 05] O. Schreer, P. Kauff & T. Sikora. 3d videocommunication : Algorithms, concepts and real-time systems in human centred communication. John Wiley & Sons, 2005.

[Schreer 06] O. Schreer, C. Fehn, N. Atzpadin, M. Muller, A. Smolic, R. Tanger & P. Kauff. *A Flexible 3D TV System for Different Multi-Baseline Geometries*. Dans ICME, pages 1877–1880. IEEE, 2006.

[Seuntiëns 06] P. J. H. Seuntiëns. *Visual experience of 3D TV*. Thèse de doctorat, Eindhoven University, Eindhoven, The Netherlands, 2006.

[Stavrakis 08] E. Stavrakis & M. Gelautz. *Interactive tools for image-based stereoscopic artwork*. Dans Stereoscopic displays and applications XIX, volume 6803, pages 1–11, January 2008.

[Wallace 06] A.M. Wallace, R.C.W. Sung, G.S. Buller, R.D. Harkins, R.E. Warburton & R.A. Lamb. *Detecting and characterising returns in a pulsed ladar system*. IEEE proceedings. Vision, image and signal processing, vol. 153, no. 2, pages 160–172, April 2006.

[Wartell 99] Z. Wartell, L. Hodges & W. Ribarsky. *Balancing fusion, image depth and distortion in stereoscopic head-tracked displays*. Dans SIGGRAPH '99 : Proceedings of the 26th annual conference on Computer graphics and interactive techniques, pages 351–358, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co.

[Wartell 02] Z. Wartell, L.. Hodges & W. Ribarsky. *An Analytic Comparison of Alpha-False Eye Separation, Image Scaling and Image Shifting in Stereoscopic Displays.* IEEE Transactions on Visualization and Computer Graphics, vol. 8, pages 129–143, 2002.

[Weise 07] T. Weise, B. Leibe & L. Van Gool. *Fast 3D Scanning with Automatic Motion Compensation.* Dans IEEE Conference on Computer Vision and Pattern Recognition (CVPR'07), June 2007.

[Wheatstone 38] C. Wheatstone. *Contributions to the physiology of vision—Part the first. On some remarkable, and hitherto unobserved, phenomena of binocular vision.* Philosophical Transactions of the Royal Society of London, pages 371–394, 1838.

[Wong 02] K. Wong & R. Cipolla. *Reconstruction of Outdoor Sculptures from Silhouettes under Approximate Circular Motion of an Uncalibrated Hand-Held Camera.* Dans IAPR Workshop on Machine Vision Applications, pages 459–462, 2002.

[Woodham 80] R.J. Woodham. *Photometric Method for Determining Surface Orientation from Multiple Images.* OptEng, vol. 19, no. 1, pages 139–144, January 1980.

[Woods 93] A. Woods, T. Docherty & R. Koch. *Image Distortions in Stereoscopic Video Systems.* Dans Proceedings of the SPIE Volume 1915, Stereoscopic Displays and Applications IV, pages 36–47, 1993.

[Yamanoue 97] H. Yamanoue. *The relation between size distortion and shooting conditions for stereoscopic images.* Journal of the SMPTE, pages 225–232, 1997.

[Yang 04] R.G. Yang, M. Pollefeys, H. Yang & G. Welch. *A Unified Approach To Real-time, Multi-resolution, Multi-baseline 2d View Synthesis And 3d Depth Estimation Using Commodity Graphics Hardware.* IJIG, vol. 4, no. 4, pages 627–651, October 2004.

[Yang 06] R. Yang, L. Wang, G. Welch & M. Pollefeys. *Stereovision on GPU.* Dans 2006 Workshop on Edge Computing Using New Commodity Architectures (EDGE 2006), May 2006.

[Yemez 07] Yemez & Wetherilt. *A volumetric fusion technique for surface reconstruction from silhouettes and range data.* Comput. Vis. Image Underst., vol. 105, no. 1, pages 30–41, 2007.

[Zhang 02] L. Zhang, B. Curless & S. M. Seitz. *Rapid Shape Acquisition Using Color Structured Light and Multi-pass Dynamic Programming.* 3D Data Processing Visualization and Transmission, International Symposium on, vol. 0, page 24, 2002.

[Zheng 00] W. Zheng, Y. Kanatsugu, Y. Shishikui & Y. Tanaka. *Robust Depth-map Estimation from Image Sequences with Precise Camera Operation Parameters.* Dans ICIP00, pages Vol II : 764–767, 2000.

[Zheng 03] C. Zheng & H. Hile. *Stereo Video Processing for Depth Map.* On his course site, 2003.

[Zhu 08] J. Zhu, L. Wang, R. Yang & J. Davis. *Fusion of time-of-flight depth and stereo for high accuracy depth maps.* Computer Vision and Pattern Recognition, IEEE Computer Society Conference on, vol. 0, pages 1–8, 2008.

Publications

[Goslin 07] F. Goslin. *Les techniques de la Réalité Virtuelle appliquées au travail collaboratif.* Dans CONFERE 2007 Ecole Nationale Supérieure d'Arts et Métiers, 2007.

[Goslin 08] F. Goslin, C. Andriot, L. Dominjon & S. Richir. *Retour vidéo stéréoscopique pour la réalisation d'une tâche collaborative entre deux utilisateurs distants.* Dans Doctoriales 2008, Université Paris-Sud 11, 2008.

[Goslin 09] F. Goslin, F. Schramm, C. Andriot, A. Bouchet & S. Richir. *High-resolution stereo video rectification through a cost-efficient real-time GPU implementation using intrinsic and extrinsic camera parameters.* Dans IEEE SSCI - Computational Intelligence in Virtual Environments, 2009.

[Sakurai 07] K. Sakurai, A. Shirai, F. Goslin & K. Miyata. *Baked crepe texture generation.* Dans SIGGRAPH '07 : ACM SIGGRAPH 2007 posters, page 27, New York, NY, USA, 2007. ACM.

[Shirai 07] A. Shirai, K. Murakami, F. Goslin, R. Tsuruno, E. Genda & S. Richir. *Wii-Media : Papier Poupee Painter"; a new usage of game controller for infancy art media.* Dans CyberGames2007 : International Conference on Games Research and Development, 2007.

RESTITUTION VIDEO STEREOSCOPIQUE MAITRISEE APPLICATION A LA REALITE VIRTUELLE

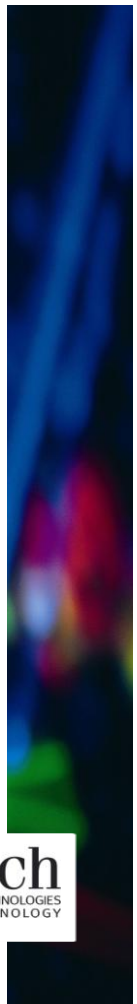
RESUME : La capture en relief d'une scène réelle peut être réalisée grâce à un couple de caméras vidéo (banc stéréoscopique). La capture de ces images vidéo stéréoscopiques et leur restitution sur des systèmes de projection en relief sont à l'interface entre les domaines de la réalité virtuelle, de la vision par ordinateur, et du cinéma en relief. Placé au sein de cette très vaste thématique, ce travail concerne la projection en relief, sur des systèmes de Réalité Virtuelle, d'images issues d'une capture par un banc stéréoscopique fixe. De très nombreuses contraintes (limitations des configurations de capture et des conditions de restitution notamment) ont restreint l'utilisation de cette technologie. Dans ce mémoire de thèse, nous détaillons les améliorations que nous avons apportées à certaines étapes de la chaîne de transmission stéréoscopique, afin de maîtriser la restitution de vidéos stéréoscopiques. Pour atteindre cet objectif, nous avons réalisé une modélisation mathématique détaillée des caméras, et des différentes configurations de capture et de restitution que nous utilisons. Disposer d'images stéréoscopiques les moins déformées possibles était un point de départ indispensable à la suite de notre travail. Dans ce but, nous avons développé un algorithme de rectification d'images vidéo stéréoscopiques. Afin d'assurer une rectification temps réel, nous avons implémenté cet algorithme sur processeur de carte graphique (GPU ou Graphics Processing Unit), en mettant en place une technique à base de table de référence. La distance interoculaire de l'utilisateur est un paramètre important pour assurer une bonne restitution des images sur les systèmes de Réalité Virtuelle. Pourtant par commodité, la valeur moyenne de cet écart est souvent prise comme référence, alors que d'importantes différences existent d'un utilisateur à l'autre. Afin d'améliorer la restitution en fixant plus précisément ce paramètre critique, nous avons développé une méthode de calibration de la distance interoculaire de l'utilisateur. Enfin, alors que les spectateurs des salles de cinéma en relief sont assis dans une zone bien définie devant l'écran, le déplacement des utilisateurs devant le système de projection d'images stéréoscopiques est une caractéristique des systèmes de Réalité Virtuelle. Pour palier aux problèmes que l'on rencontre lors de la projection d'images issues d'un banc stéréoscopique fixe pour un utilisateur en mouvement, nous proposons une méthode pour maîtriser la restitution de la profondeur perçue par cet utilisateur, en nous basant sur une segmentation en profondeur de la scène.

Mots clés : réalité virtuelle, vidéo stéréoscopique, rectification gpu, calcul distance interoculaire, restitution maîtrisée de la profondeur

DEPTH-CONTROLLED RENDERING OF STEREOSCOPIC VIDEO IMAGES FOR VIRTUAL REALITY APPLICATIONS

ABSTRACT: Acquiring 3D scenes can be achieved by using a pair of video cameras (stereo rig). The acquisition process of these stereoscopic video images and their projection on 3D display systems is a wide-extending field which borrows from the technologies of virtual reality, computer vision and 3D filming. Within this wide field, we focused our research work on the projection in 3D of images captured by a fixed stereo rig, on virtual reality systems. The use of this technology was limited by numerous constraints (pertaining essentially to capture configurations and display conditions). In this PhD thesis, we detail the improvements we brought to some of the steps of the stereo transmission process, in order to control more precisely the display of stereoscopic videos and to alleviate some of the limitations mentioned above. We based our work on detailed mathematical modeling of the cameras, and of the capture and the display configurations. First, it was necessary to work on stereoscopic images as little distorted as possible. To obtain such images, we developed an algorithm that corrects distortions on these images. To ensure real-time rectification, we implemented this algorithm on GPU (Graphics Processing Unit), through the use of a technique based on reference tables.
The interocular distance is a fundamental step to make a full use of virtual reality system. In spite of the fact that significant disparities exist from one user to another, an average value is often used for this parameter. In order to set correctly this critical parameter, we developed a method to calibrate the interocular distance of each user, so that the display of stereoscopic images on these systems will provide a more accurate perception. Finally, while viewers in 3D theaters sit in a defined area in front of the projection screen, allowing users to move freely is one of the main characteristics of virtual reality systems. To overcome the difficulties that occur when projecting images captured by a fixed stereo rig to a user in motion, we propose a method to improve the restitution of the depth perceived by the user, using a depth segmentation of the captured scene.

Keywords : virtual reality, stereoscopic video, gpu rectification, interocular distance, depth controlled rendering



ARTS ET MÉTIERS ParisTech

ParisTech INSTITUT DES SCIENCES ET TECHNOLOGIES PARIS INSTITUTE OF TECHNOLOGY