Reconfiguration stéréoscopique

par

Jean-Christophe Houde

Mémoire présenté au Département d'informatique
en vue de l'obtention du grade de maître ès sciences (M.Sc.)

FACULTÉ DES SCIENCES
UNIVERSITÉ DE SHERBROOKE

Sherbrooke, Québec, Canada, juillet 2012

Library and Archives Canada

Published Heritage Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et Archives Canada

Direction du Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 978-0-494-88874-2

Our file Notre référence
ISBN: 978-0-494-88874-2

Le 6 juillet 2012

le jury a accepté le mémoire de Monsieur Jean-Christophe Houde dans sa version finale.

Membres du jury

Professeur Pierre-Marc Jodoin
Directeur de recherche
Département d'informatique

Professeur François Deschênes
Codirecteur de recherche
Université du Québec à Rimouski

Professeur Richard Égli
Membre
Département d'informatique

Professeur Jean-Pierre Dussault
Président rapporteur
Département d'informatique

SOMMAIRE

Au cours des dernières années, le cinéma tridimensionnel a connu un regain de popularité. La réalisation de plusieurs films d'animation 3D de qualité, de même que le succès fulgurant du film *Avatar* aura permis au grand public de constater la qualité de cette nouvelle génération de technologies 3D. Cependant, un problème fondamental ralentit toujours l'adoption à la maison de ce mode de divertissement. En effet, tout contenu visuel produit en se basant sur des techniques de stéréoscopie subira des distorsions visuelles lorsqu'observé dans des conditions différentes de celles considérées lors de la création du contenu. Autrement dit, un film 3D tourné pour un cinéma de grande dimension n'aura pas une richesse de profondeur aussi grande lorsqu'il sera visualisé sur un écran domestique.

Ce mémoire présente un cadre de travail, un modèle mathématique et un ensemble de techniques permettant de « reconfigurer », en générant de nouvelles images, le contenu stéréoscopique original afin que l'effet de profondeur original soit préservé dans les nouvelles conditions de visualisation.

Mots-clés : reconfiguration stéréoscopique, stéréovision, rendu à base d'images, cinéma 3D, modèle de stéréoscopie, emplissage d'image, synthèse de nouvelle vue.

REMERCIEMENTS

Je tiens à remercier mon directeur de recherche, Pierre-Marc Jodoin, pour l'opportunité qu'il m'a donné d'évoluer sous sa supervision avertie. J'ai grandement apprécié cette expérience de recherche en milieu académique. Je le remercie pour nos discussions, autant scientifiques que sur la vie en général. Elles m'ont beaucoup apporté.

Merci à mon codirecteur de recherche, François Deschênes, pour m'avoir mis en contact avec Sensio et pour son intérêt pour le projet malgré son horaire bien rempli. De même, merci à l'entreprise Sensio de m'avoir fourni la problématique de ce mémoire et de m'avoir accueilli à plusieurs reprises dans leurs locaux de Montréal. L'interaction avec cette compagnie m'aura permis de goûter à la recherche dans un contexte industriel.

Merci à mes amis, collègues et professeurs du centre de recherche MOIVRE pour le partage d'idées et les moments de détente n'ayant aucun lien avec les études. Merci à ma famille et mes amis proches d'avoir enduré sans broncher ma réponse répétitive de « bientôt » lorsqu'ils me demandaient à quel moment je terminerais ma maîtrise.

Finalement, merci à Mélissa pour son soutien orthographique dans mes questionnements pointus sur la langue française. Mais surtout, merci à cette femme qui partage ma vie de m'avoir enduré, soutenu et encouragé tout au long de ce processus.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES ALGORITHMES

INTRODUCTION

Au cours des dernières années, le cinéma tridimensionnel (3D) a fait un autre retour en force. Les cinéphiles ont la possibilité de visionner la plupart des films à grand budget en 3D. Certains observateurs du marché considèrent que, cette fois, le cinéma 3D est là pour rester, tandis que d'autres observateurs restent sceptiques. Ces sceptiques se basent sur deux principaux arguments : (1) la qualité discutable de l'effet 3D de certains films, qui pourraient provoquer un désintérêt chez les cinéphiles, et (2) le manque de contenu 3D pouvant être visionné à la maison, qui n'encouragera pas les consommateurs à se procurer de l'équipement supportant le contenu 3D. La première problématique est vérifiée et est principalement due aux films ayant été tournés de manière classique et dont l'effet 3D a été ajouté en post-traitement. Les effets 3D de ce type sont souvent de moins grande qualité, car leur création en post-traitement est limitée par l'information déjà connue. Par contre, cette problématique tend à s'amenuiser, puisque la plupart des nouveaux films 3D ont été produits directement pour ce type d'expérience.

La problématique du manque de contenu 3D pour un visionnement à la maison reste cependant d'actualité. De plus en plus de fabricants de télévisions proposent des équipements supportant directement la technologie 3D. Cependant, le manque de contenu

limite le nombre de consommateurs intéressés à se procurer de tels équipements. Ce manque de contenu est dû au fait qu'un film 3D ne peut pas être directement visionné sur un écran plus petit sans encourir des distorsions visuelles diminuant ou éliminant l'effet 3D. L'entreprise montréalaise Sensio, qui oeuvre dans le domaine de la transmission et de l'encodage de contenu 3D depuis 1999, tente de trouver une solution à ce manque de contenu. Le problème est qu'il est impossible d'acquérir une séquence ayant exactement le même contenu pour deux configurations de visionnement différentes. En effet, il est pratiquement impossible qu'un acteur fasse précisément les mêmes mouvements deux fois de suite. De plus, les coûts qui seraient entraînés par plusieurs acquisitions de la même séquence seraient prohibitifs.

La problématique posée par Sensio est donc la suivante : comment peut-on reconfigurer une séquence 3D acquise pour un visionnement cinéma afin qu'elle produise le même effet 3D lorsque visionnée dans un contexte différent, tel un cinéma maison ou un écran d'ordinateur ? Cette problématique est soumise à la contrainte que pratiquement aucune information sur l'acquisition originale n'est disponible lors de la reconfiguration. Seules les images de la séquence, ainsi que la configuration initialement ciblée (par exemple, pour un cinéma IMAX©) sont disponibles.

Afin de proposer une solution à cette problématique, ce mémoire présentera tout d'abord une revue de la littérature pertinente. Par la suite, un modèle géométrique pour l'acquisition et le visionnement de séquences 3D sera décrit. Les différentes étapes du cadre de traitement proposé seront ensuite décrites, puis celles-ci seront validées à l'aide d'une expérimentation. Finalement, certaines conclusions et pistes d'avenir seront proposées.

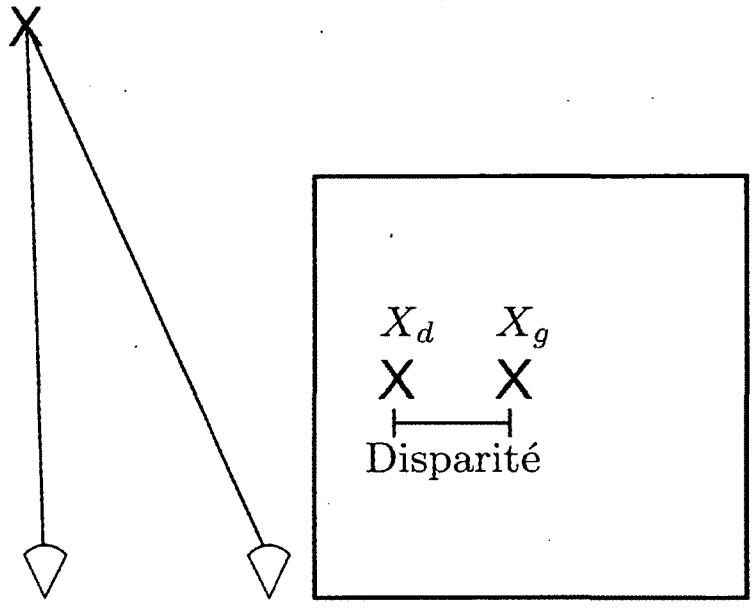
CHAPITRE 1

Mise en contexte et description de la problématique

Afin de bien aborder ce mémoire, la problématique est tout d'abord présentée, tout en identifiant les principaux composants impliqués dans cette problématique. Par la suite, une revue de la littérature concernant ces divers composants est présentée.

1.1 Problématique

Le cinéma tridimensionnel est très souvent basé sur une caractéristique du système visuel humain, le principe de binocularité. En effet, l'un des principaux indices visuels utilisés par notre cerveau pour décoder et comprendre la profondeur de la scène que l'on observe est la disparité (voir [41]). La disparité est la différence de position de la projection d'un objet dans les images perçues par nos deux yeux (voir figure 1.1). Plus la disparité est grande pour un point ou un objet donné, plus celui-ci est près de l'ob-



(a) Modèle du spectateur (b) Perception du spectateur

Figure 1.1 – Concept de disparité binoculaire : (a) représente une scène où se trouvent les deux yeux du spectateur, ainsi qu'un objet en forme de croix, (b) représente les images de gauche et de droite ramenées dans le même repère. Les deux images sont donc superposées, et la distance entre X_g et X_d est la disparité associée à ce point.

servateur. La plupart des technologies de cinéma 3D exploitent cette caractéristique du système visuel humain afin de créer l'illusion de profondeur, comme l'analysent Konrad et Halle [33].

Afin d'exploiter l'effet de disparité lors de la création d'un film ou d'une séquence stéréoscopique, la technique la plus courante est de filmer la scène de deux points de vue légèrement décalés, afin de simuler le positionnement des yeux du spectateur. Cependant, pour obtenir un effet 3D de qualité ne causant pas d'inconfort, le réalisateur doit ajuster certains paramètres du positionnement des caméras. Ces paramètres sont souvent décrits par un modèle tenant compte de diverses variables, comme la dimension de la surface de visualisation de la séquence et la distance entre l'observateur

moyen et cette surface de visualisation (voir [6, 34, 65, 66, 67, 68]). Avant de commencer le tournage, le réalisateur décide de l'effet tridimensionnel qu'il désire créer tout en considérant la scène, le modèle utilisé et les paramètres externes. Il pourra par exemple choisir de configurer son film pour donner un aspect naturel lorsque visionné dans un cinéma IMAX©.

Si le réalisateur suit les recommandations du modèle pour toutes les séquences d'un film, l'effet 3D perçu lorsque qu'observé dans les conditions idéales sera exactement tel que prévu. Par contre, si les conditions de visualisation sont moins qu'idéales (par exemple, si l'écran n'a pas la dimension prévue, ou si le spectateur se trouve trop près ou trop loin de cet écran), quelles sont les répercussions sur la perception qu'aura le spectateur ? En fait, cela dépend du modèle utilisé. En général, on observe que la qualité de l'effet 3D perçu dans une telle situation est grandement diminuée.

Le problème que ce mémoire tente de résoudre est le suivant : comment est-il possible de reconfigurer les images originales afin de retrouver l'effet 3D original dans une autre configuration de visualisation ? Plusieurs éléments sont à considérer lorsque l'on aborde cette problématique. Tout d'abord, seules les images originales sont habituellement disponibles, et il n'est évidemment pas possible de filmer à nouveau la scène en utilisant de nouveaux paramètres. De plus, les paramètres originaux d'acquisition, tels la distance entre les deux caméras, leur orientation et la distance du point d'intérêt dans la scène sont souvent indisponibles. Finalement, même avec une excellente technique de reconfiguration et tous les paramètres originaux connus a priori, il est plausible que des artéfacts de reconfiguration apparaissent dans les images reconfigurées. Il faut donc avoir un moyen de corriger ces artéfacts.

1.2 Littérature et systèmes existants

Le modèle mathématique et le cadre de traitement proposés aux chapitres 2 et 3 et visant à régler la problématique exposée précédemment sont basés sur des techniques provenant de plusieurs domaines. Les principaux domaines touchés dans ce mémoire sont

- la géométrie des systèmes de caméras pour une acquisition stéréoscopique, afin de pouvoir définir le modèle à utiliser et les relations entre les différents paramètres ;
- les techniques de mise en correspondance stéréoscopiques, permettant de pouvoir reconstruire la scène ;
- les méthodes de génération de nouvelles vues, pour pouvoir générer de nouvelles images selon une structure de caméras déterminée ;
- les algorithmes d'emplissage d'images, afin de corriger les artéfacts ayant pu être créés à l'étape précédente.

Une revue de la littérature pertinente à chacun de ces domaines suit dans les prochaines sections.

1.2.1 Structure des systèmes de caméras

L'étape de base lors de la création d'une séquence stéréoscopique est de déterminer le positionnement des caméras, de même que leur orientation et leurs paramètres internes. Plusieurs auteurs se sont attardés à développer des modèles pour déterminer les paramètres à utiliser pour obtenir les meilleures acquisitions possibles, tandis que d'autres ont évalué les distorsions visuelles pouvant être induites par un mauvais choix de paramètres. Comme la problématique à régler dans ce mémoire consiste à obtenir

des images représentant la scène pour une nouvelle configuration de visionnement, il est naturel de s'intéresser à des articles présentant différents modèles.

Tout d'abord, certains auteurs se sont intéressés aux facteurs biologiques et humains de la perception tridimensionnelle. L'article [24] analyse ces facteurs d'un point de vue biologique. Les auteurs ne proposent cependant pas de modèle de configuration de caméras. Dans [28], IJsselsteijn *et al.* évaluent l'impact des paramètres sur la qualité et l'aspect naturel des images stéréoscopiques. Ils concluent que la durée du stimuli n'a pas beaucoup d'effet sur la qualité perçue des images, sauf dans les cas où les disparités sont exagérées. Jones [30] décrit un modèle simple de configuration des caméras et utilise ce modèle pour analyser les facteurs humains qui influencent la perception. Il identifie entre autres la limite moyenne de distance de perception 3D par rapport à un écran d'ordinateur ordinaire.

Kutka [34] établit un modèle géométrique et discute de l'impact du changement de grandeur d'écrans sur la création de l'effet «théâtre de marionnettes». Il établit aussi une formule représentant l'effet d'un changement de la taille d'écran sur la profondeur perçue. Cependant, son modèle ne fonctionne que pour des caméras en configuration parallèle.

Hirokazu Yamanoue et ses collabrateurs étudient les systèmes de télévision stéréoscopique depuis plusieurs années. Dans [66], les auteurs analysent l'effet perceptuel des distorsions dans les images gauche-droite, de même que la tolérance du spectateur à ces distorsions. L'article [68] établit la relation entre les paramètres d'acquisition et la compression du volume spatial perçu. Cette analyse leur permet de mettre en évidence les distorsions visuelles de type «théâtre de marionettes» et d'effet «cartonné». L'effet «cartonné» représente la perte d'impression de profondeur pour un objet. Celui-ci se retrouve à avoir l'air d'être fait en carton et de ne pas avoir d'épaisseur. L'effet

«théâtre de marionettes» représente ce qui se passe lorsqu'il n'y a que quelques valeurs de profondeur perçues dans la scène. Dans ce cas, plusieurs effets «cartonnés» sont combinés à chaque niveau de profondeur. Une version plus fouillée de cette analyse est présentée dans [67]. Les auteurs y apportent de nouvelles constatations. La plus importante indique que, pour éviter l'effet «cartonné», il faut que la mise à l'échelle de la profondeur soit égale à la mise en échelle dans le plan de projection. Dans [65], une évaluation des différences entre une configuration de caméras parallèles et une configuration de caméras convergentes est présentée. La conclusion principale est que le modèle parallèle préserve la linéarité, contrairement à la plupart des modèles convergents. Un modèle de configuration est ensuite établit selon ces conclusions.

Ariyaeeinia [6] présente un modèle extrêmement semblable au modèle utilisé par la compagnie montréalaise Sensio, compagnie qui a soulevé la problématique abordée dans ce mémoire. L'auteur établit un modèle basé sur certaines caractéristiques qui, selon ses expériences, influent la qualité d'un système de télévision 3D. Les paramètres établis sont la disparité perceptible minimale, la dimension de la région confortable de vision stéréoscopique, de même que le facteur de mise à l'échelle de l'image. L'auteur étudie aussi l'effet de la variation de l'échelle de l'image, de l'angle entre les caméras et de la longueur de la ligne de base sur la région de perception stéréoscopique confortable. La région de perception stéréoscopique confortable est le volume d'espace simulé où le spectateur est capable de fusionner les deux images (et donc de percevoir un effet 3D) sans trop ressentir de fatigue oculaire. Le changement de longueur focale ne fait que varier l'étendue de cette région, tandis que la rotation des caméras et le changement de distance entre les deux caméras changeront l'emplacement de cette région. Ce modèle fournit une méthode de calcul de ces paramètres selon le volume de perception 3D et la distance interoculaire. Le modèle qui sera proposé au chapitre

2 est basé sur ce modèle et certaines des contraintes qui y sont posées.

À la différence des modèles génériques et minimisant la distorsion présentés précédemment, certains auteurs s'intéressent à des modélisations permettant de créer des effets particuliers, principalement pour des rendus infographiques. Par exemple, Holliman [26] présente une méthode permettant de déterminer une région d'intérêt où la perception de la profondeur sera améliorée et renforcée. Cependant, cette technique ne s'applique que dans le cas d'images de synthèse, car il faut faire trois rendus séparés avec trois configurations de caméras différentes. Dans un contexte réel, même si trois acquisitions simultanées étaient effectuées, il serait impossible de fusionner le contenu des trois images.

Finalement, certains auteurs proposent des méthodes de correction de l'effet 3D dans certains contextes particuliers. Par exemple, Lee et Kang [35] présentent une méthode de correction pour des caméras convergentes avec lentille grand-angle pour appareils mobiles. Il s'agit donc d'un cas particulier. Selon eux, le modèle serait extensible à d'autres types de matériel, mais ils n'en font pas la preuve.

Dans les articles mentionnés ici, il est souvent configuration convergente et parallèle des caméras. Lorsqu'il est question d'une configuration parallèle, cela signifie que les axes optiques des deux caméras sont parallèles. À l'inverse, une configuration convergente implique que les axes optiques des deux caméras se rencontrent dans l'espace. Ces deux configurations ont différentes conséquences sur l'effet 3D produit. Dans le reste de ce mémoire, il ne sera question que de configuration convergente, car c'est la configuration la plus souvent rencontrée dans un contexte de cinéma. L'utilisation de la configuration parallèle entraînerait aussi une obligation d'utiliser certaines étapes de post-traitement. Sans ce post-traitement, le modèle convergent crée un effet 3D où tous les objets se trouvent devant la surface de projection, ce qui

peut causer une fatigue oculaire.

1.2.2 Stéréovision

Comme il sera expliqué à la section 3.4.2, dans le cadre présenté dans ce mémoire, il est nécessaire d'utiliser un algorithme de stéréovision pour retrouver la structure de la scène. Ces méthodes permettent d'obtenir des cartes de disparités, qui peuvent être converties en cartes de profondeur, ou utilisées pour calculer une représentation 3D de la scène. Ces représentations seront nécessaires lors du calcul de nouvelles vues. Le domaine de la mise en correspondance stéréoscopique est un champ de recherche extrêmement vaste et actif. Des dizaines d'articles sont publiés chaque année par rapport à ce domaine, et les méthodes proposées couvrent un très large spectre de techniques. Il serait donc quasi impossible de présenter une revue de littérature exhaustive de ce domaine.

Dans cette optique, quelques articles souvent cités dans ce domaine seront présentés afin d'avoir un bref aperçu des premières méthodes proposées. Le cadre de validation de Middlebury sera ensuite présenté. Comme il est quasi impossible de faire une revue exhaustive de toutes les techniques existantes, la présentation de ce cadre permettra au lecteur intéressé de trouver le meilleur algorithme au moment de sa lecture. Afin d'obtenir une carte de disparités, il faut utiliser un optimiseur qui trouvera la solution optimale pour une formulation donnée. Dans ce contexte, les deux optimiseurs les plus fréquemment utilisés seront présentés. Une problématique majeure de la mise en correspondance stéréo est la présence d'occultations. Quelques articles proposant d'intégrer la gestion des occultations dans la fonction de coût seront présentés. Récemment, deux familles de techniques ont connu une croissance rapide : les techniques

bàsées sur des régions, de même que des techniques d'ajustement de plans. De telles méthodes seront aussi présentées. Finalement, comme la problématique à résoudre s'applique à des séquences vidéos, certaines techniques de stéréovision incorporant une composante temporelle seront décrites.

Articles fondateurs

Le domaine de la stéréovision a émergé simultanément de plusieurs laboratoires et groupes de recherche. Il n'y a donc pas un fondateur précis. Cependant, certains articles sont fréquemment cités. Il peut être intéressant de mentionner le travail de Hannah [22], dont la thèse de doctorat représente une des premières études fouillées des méthodes de stéréovision. L'auteur y propose certaines métriques de qualité de correspondance, des techniques de détection de zones impossibles à mettre en correspondance (zones reliées à des occultations) et suggère des techniques pour calculer un modèle des caméras selon les correspondances.

Un autre article souvent cité est celui de 1998 de Birchfield et Tomasi [8]. Une limitation lors des calculs de mise en correspondance est le fait que les valeurs de disparités sont discrètes, et le plus souvent entières. Cela peut amener des erreurs de mise en correspondance si le meilleur candidat se situe à une disparité non entière. Les auteurs tentent de résoudre ce problème à l'aide d'une mesure de dissimilarité insensible à l'échantillonage, car basée sur les valeurs d'intensité interpolées des voisins. Cette métrique a souvent été réutilisée par la suite comme base pour des métriques plus poussées.

Dans leur article de 2004 (voir [54]), Scharstein et Szeliski introduisent le concept d'image de l'espace de disparités, ou *Disparity Space Image*. L'idée est inspirée de l'article de Birchfield et Tomasi [8], où il est suggéré d'interpoler selon les intensités

des voisins avant de calculer le coût de mise en correspondance. Les auteurs proposent d'utiliser une interpolation au demi-pixel avant d'échantilloner afin d'obtenir de meilleurs résultats lors de la mise en correspondance. L'image de l'espace de disparités sera reprise comme manière d'organiser l'information de coût dans plusieurs autres algorithmes.

Cadre de validation de Middlebury

En 2002, Scharstein et Szeliski ont publié un article établissant une taxonomie et faisant une évaluation des algorithmes de stéréovision dense les plus connus à ce moment (voir [46]). Ils ont aussi développé un site web [45] répertoriant et organisant les évaluations des algorithmes. Le site permet aussi à d'autres chercheurs de faire évaluer leur nouvel algorithme. Depuis la publication de l'article, cette évaluation est devenue pratiquement incontournable dans le domaine de la stéréovision. Cependant, cela tend aussi à créer un biais dans l'optimisation des paramètres des algorithmes proposés, car les chercheurs peuvent vouloir obtenir de meilleurs résultats dans l'évaluation en utilisant des valeurs de paramètres optimisées pour ces séquences. Le problème est que, lorsqu'un autre individu tente d'utiliser les meilleurs algorithmes sur de nouvelles séquences, il est possible que le résultat soit moins probant, étant donné des paramètres mal ajustés à ces nouvelles séquences. Il faut donc prendre les classements avec un grain de sel. Néanmoins, cette évaluation aura permis de mieux distinguer les tendances et idées qui sous-tendent les meilleurs algorithmes. Il est donc utile de considérer ce classement lorsque vient le temps de choisir un algorithme de mise en correspondance stéréoscopique, comme à la section 3.4.2. De plus, il recense beaucoup d'approches différentes, et le lecteur intéressé est invité à aller le consulter.

Optimiseurs

Pour effectuer la mise en correspondance, il faut assigner une valeur de disparité à chaque pixel des deux images de la paire sréréoscopique. Cette valeur de disparité peut être assimilée à une étiquette. La formulation la plus fréquente de ce problème est faite sous la forme d'une fonction de coût de correspondance que l'on tente de minimiser. Il est possible d'utiliser des techniques locales, telles la programmation dynamique et l'optimisation vorace, mais les approches globales donnent très souvent de meilleurs résultats, car elles incluent fréquemment une modélisation du voisinage. Cette modélisation permet d'obtenir un résultat ayant moins de variations aléatoires. Afin que le problème de minimisation sur l'ensemble de l'image soit traitable, il est souvent représenté sous la forme d'un champ de Markov. Une fois sous cette forme, la fonction doit tout de même être optimisée. Deux techniques sont principalement utilisées pour ce faire : les algorithmes de coupe de graphe («graph-cut») et l'algorithme de propagation de croyance («belief propagation»).

Boykov et Kolmogorov sont deux des chercheurs ayant popularisé les algorithmes de coupe de graphe, qui consistent à trouver la coupe de coût minimal dans un graphe tridimensionnel représentant les coûts de correspondance pour différentes disparités, pour chaque pixel des images d'entrée. En 2001, Boykov *et al.* [10] démontrent que cette famille d'algorithmes permet de trouver rapidement une solution approximative au problème d'optimisation, solution dont l'erreur est bornée. La même année, Kolmogorov et Zabih [31] bâtissent sur des techniques déjà publiées et proposent une formulation optimisée avec un algorithme de coupe de graphe et tenant compte des occultations dans la fonction de coût. En 2004, Boykov et Kolmogorov [9] font équipe pour évaluer différentes implémentations d'algorithmes de coupe de graphe afin d'en évaluer l'efficacité et le taux d'erreur. Ils proposent aussi un nouvel algorithme. Selon

leurs tests sur trois tâches classiques de vision par ordinateur (restoration d'images, stéréovision et segmentation), leur nouvel algorithme est toujours plus rapide que les trois algorithmes classiques, pour des résultats équivalents.

La famille des algorithmes de propagation de croyance simule une communication entre les noeuds d'un graphe afin de calculer la probabilité des noeuds cachés selon l'état des noeuds visibles. Des messages sont passés entre les noeuds, et ceux-ci mettent à jour leur probabilité selon diverses conditions dépendantes de l'algorithme. Yedidia *et al.* [70] ont publié un rapport technique expliquant et vulgarisant les bases de cette famille d'algorithmes. En 2003, Sun *et al.* [53] proposent une des premières méthodes de mise en correspondance stéréoscopique se basant sur la propagation de croyance. Ils ouvrirent la voie à de nombreuses autres approches, et cette famille de techniques est maintenant aussi fréquente que celle basée sur les coupes de graphe.

En 2008, Szeliski *et al.* [55] ont effectué une étude comparative de différentes techniques de minimisation d'énergie sur des champs de Markov. Ils ont entre autres comparé la coupe de graphe à la propagation de croyance pour diverses utilisations. Dans le cas de la mise en correspondance stéréo, les auteurs ont conclu que les deux techniques ont des résultats extrêmement similaires, et que chaque technique remportait la palme du temps de calcul le plus court pour la moitié des tests. Le choix d'une technique de stéréovision ne saurait donc être basé uniquement sur le choix d'un optimiseur.

Gestion des occultations

Une des principales difficultés des algorithmes de stéréovision est due au phénomène d'occultation. Le phénomène d'occultation se produit lorsqu'un objet n'est visible que dans une seule des deux images (voir la figure 1.2). Dans ce cas, aucune correspon-

dance ne pourra être établie pour cet objet. Cela entraîne la création d'artéfacts dans les cartes de disparités, artéfacts qui peuvent se répercuter dans les résultats de la reconfiguration. Il faut donc être capable de détecter et d'éviter ces artéfacts.

En 2000, Zitnick et Kanade [73] publient un algorithme coopératif qui effectue simultanément la mise en correspondance et la détection des occultations. Une itération de mise en correspondance est suivie d'une itération de détection des occultations, qui fournira des indications à l'étape suivante de mise en correspondance. Cela itère jusqu'à stabilisation des résultats. Triantafyllidis *et al.* [57] proposent plutôt de créer un test de Bayes permettant de détecter les occultations. Cependant, il n'y a pas de calcul explicite des disparités ; la méthode est principalement utilisée pour classifier les pixels comme faisant partie de l'avant-plan, de l'arrière-plan ou d'une région occultée. Dans leur article de 2005 (voir [29]), Ince et Konrad évaluent deux méthodes traditionnelles d'estimation des occultations : géométrique (vérification gauche-droite) et photométrique (différences d'intensité). Ils proposent aussi une nouvelle méthode, où ils calculent les valeurs de disparité et analysent la densité des vecteurs dans un voisinage. Si la densité n'est pas constante, cela permet de dire qu'il y a probablement une occultation.

Dans un article publié en 2002 (voir [18]), Egnal et Wildes effectuent une mesure de la qualité de cinq algorithmes de détection des demies-occultations : vérification gauche-droite, bimodalités dans les disparités, variation soudaine de qualité de mise en correspondance, contrainte d'ordonnancement et contrainte d'occlusion. Ils concluent que chacune des méthodes donne de bons résultats dans certains cas précis. Il n'y a donc pas, selon eux, de méthode universelle.

Comme mentionné plus tôt, Kolmogorov et Zabih [32] ont proposé une formulation de mise en correspondance gérant les occultations et optimisée par coupe de graphe.

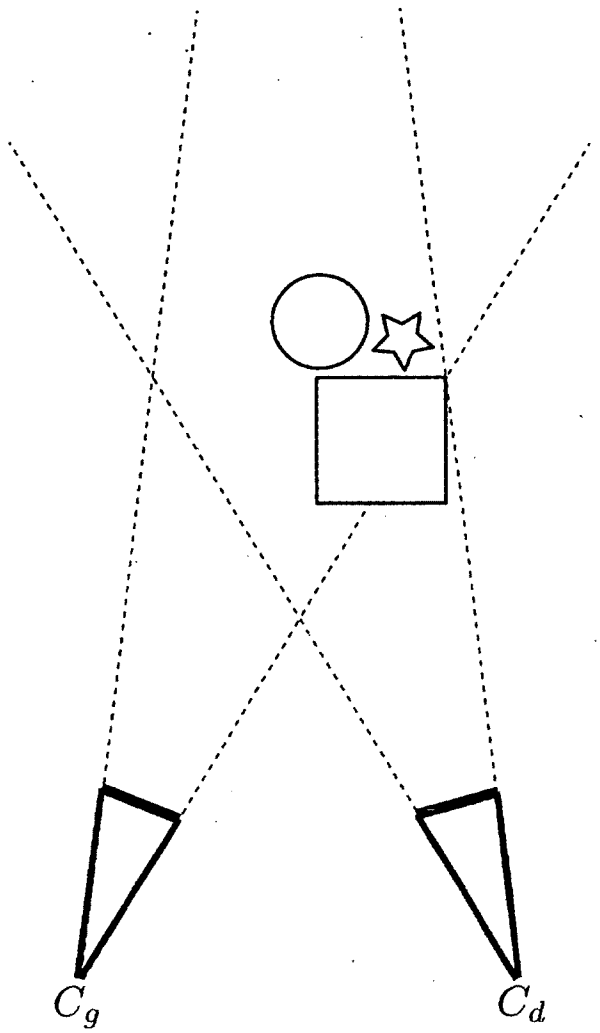


Figure 1.2 – Phénomènes d'occultation lors de la mise en correspondance stéréoscopique : l'objet sphérique est visible seulement pour C_g, et l'objet à forme d'étoile est invisible pour les deux caméras. La sphère ne pourra donc pas être mise en correspondance correctement.

Il faut noter que leur modèle considère implicitement une symétrie dans la présence des occultations. À l'inverse de cela, Wei [61] a publié un algorithme aussi basé sur la coupe de graphe mais modélisant explicitement les occultations comme un phénomène asymétrique en utilisant une seule carte de profondeur. Cette méthode est aussi extensible aux cas où plus de deux images sont connues.

Techniques basées sur les régions

Une famille de techniques de stéréovision ayant connue un gain de popularité dans les dernières années sont les algorithmes de mise en correspondance basés sur des régions. Au tout début de la stéréovision, les algorithmes tentaient de mettre en correspondance des points caractéristiques des images. La correspondance était plus fiable, mais n'était pas dense. Par la suite, les chercheurs ont tenté de mettre en correspondance tous les points d'une image. Ici, la mise en correspondance calculée est dense, mais elle est moins fiable, particulièrement dans les régions ayant peu de texture. Logiquement, l'étape suivante est de vouloir obtenir une carte dense et fiable. Certains chercheurs ont donc décidé de tenter de mettre en correspondance des régions. Cela permet d'avoir une carte de correspondance couvrant toute l'image, mais dont la fiabilité est accrue, étant donné que les régions sont plus faciles à mettre en correspondance.

Un premier algorithme basé sur les régions est celui de Zitnick *et al.* [75]. Dans cet algorithme, la mise en correspondance n'est pas une fin en soi. Il est cependant intéressant d'observer que les images sont tout d'abord segmentées en régions de couleurs similaires, puis mises en correspondance selon la forme, la superficie et la couleur des régions. Dans la première phase de l'algorithme, chaque région ne possède qu'une valeur de disparité. Par la suite, la disparité de chaque pixel de la région est affinée

afin d'obtenir une carte plus régulière, aux transitions moins franches, sauf sur les contours des objets.

Wang et Zheng [60] ont proposé de segmenter les images à l'aide de la technique de décalage de moyenne (*Mean Shift*). Les disparités sont ensuite optimisées localement, en tenant compte de l'information des régions voisines. Finalement, une optimisation globale, dont les termes d'énergie dépendent des différences de couleur des pixels correspondants, de même que d'un test de similitude gauche-droite (afin de gérer les occultations).

Gales *et al.* [20] décrivent une technique utilisant une segmentation basée sur la couleur. Une fois la segmentation effectuée, un processus aléatoire estime différentes approximations de la disparité de chaque pixel selon un modèle de surface. Chaque disparité trouvée obtient un vote, la disparité finale étant obtenue en estimant le mode de la distribution des votes.

Techniques d'ajustement de plans

Une famille de méthodes similaires aux méthodes basées sur les région est la famille des méthodes d'ajustement de plans (*plane fitting*). Dans les techniques de cette famille, on tente d'ajuster des plans sur différentes parties de l'image afin de simuler la structure de la scène. Cet ajustement de plans permet ensuite de calculer des disparités. Plusieurs méthodes basées sur les régions utilisent aussi une technique d'ajustement de plans, que ce soit implicitement ou explicitement.

Dans leur article de 2009 (voir [69]), Yang *et al* proposent d'utiliser une approche hiérarchique dans laquelle il y a une phase d'ajustement de plans, permettant entre autres de bien gérer les problèmes de manque de texture.

Humemberger *et al* [27] proposent une méthode basée sur une segmentation des couleurs et dont la qualité de mise en correspondance dans les endroits moins texturés est améliorée. L'amélioration est due à la mise en correspondance locale qui est faite sur la base d'un recensement des disparités possibles. Les pixels pour lesquels l'algorithme a un faible niveau de confiance sont ensuite réestimés en calculant un plan de disparités pour le segment correspondant. Selon les tests des auteurs, la précision au niveau subpixelique s'en trouve améliorée.

Stéréovision temporelle

L'extension naturelle de la stéréovision spatiale est d'ajouter de l'information temporelle à la méthode. Le raisonnement motivant cette idée est qu'étant donné que plus d'information est disponible pour résoudre le problème, meilleurs seront les résultats. Dans le cas de la problématique de ce mémoire, de telles méthodes pourraient être applicables puisque le but est de reconfigurer une séquence stéréoscopique.

En 1997, Mills [39] établit un court résumé de ce qui existait à ce moment. Il propose d'intégrer la mise en correspondance stéréo et une composante temporelle, et met en évidence une contrainte sur la mise en correspondance découlant de cette intégration. Il propose aussi de formuler la problématique comme un graphe et d'identifier les objets rigides et articulés en utilisant une recherche des composants connectés du graphe.

L'article de 2003 de Curless, Seitz et Li (voir [13]) propose d'optimiser une fonction de coût à laquelle un terme basé sur la différence temporelle est ajouté. L'effet net est que la fenêtre de recherche n'est plus seulement spatiale, mais aussi temporelle. Les auteurs proposent trois formulations : une pour laquelle la fenêtre forme un prisme à base rectangulaire, une qui est oblique, et une troisième qui suit les objets en mouvement.

Ils concluent que leur méthode est efficace pour des scènes statiques dont l'éclairage change, de même que pour des scènes quasi-statiques (telles une chute d'eau). Elle fonctionne aussi très bien pour des patrons de lumière aléatoires projetés sur un objet statique. Cependant, pour une scène dans laquelle les objets se déplacent d'une image à l'autre, les auteurs remarquent qu'allonger la fenêtre temporelle avait le même effet que d'avoir une fenêtre uniquement spatiale plus large. La méthode n'est donc pas aussi efficace que souhaité.

En 2003 et 2005, Davis *et al.* [14, 15] définissent une nouvelle classification des méthodes de stéréovision selon le domaine utilisé. Cela leur permet de définir une nouvelle méthode, basée sur une fenêtre à la fois spatiale et temporelle. Selon leurs tests, il faut faire un compromis entre la grandeur de la fenêtre spatiale et la grandeur de la fenêtre temporelle. La fenêtre est toujours droite. Cette conclusion vient rejoindre ce que Curless *et al.* avait conclu dans [13].

Une autre approche est celle proposée par Adedoyin *et al.* en 2007 (voir [3, 4]). Ils proposent d'optimiser la disparité et la carte de mouvement en même temps à l'aide d'une stratégie évolutive de la famille des algorithmes génétiques. Une stratégie de raffinement sous-pixel est aussi intégrée à cette technique.

Zhang *et al.* [71] présentent une technique permettant de fusionner les cartes de profondeur provenant de différentes sources. Ce n'est donc pas directement de la stéréovision temporelle, mais la technique de fusion proposée pourrait être utilisée pour fusionner des cartes de disparités provenant de différentes techniques, dont certaines pourraient être uniquement spatiales, et d'autres, uniquement temporelles.

Il faut constater que, pour les méthodes présentées, les résultats sur des séquences réelles contenant une grande diversité de mouvements ne sont pas nécessairement

meilleurs que les résultats des méthodes n'utilisant que la stéréovision spatiale. Pour cette raison, aucune méthode de ce type ne sera explicitement utilisée dans le cadre de traitement proposé. Elles pourraient quand même y prendre une place si elles deviennent plus générales.

1.2.3 Génération de nouvelles vues

La solution proposée pour résoudre la problématique posée dans ce mémoire nécessitera de générer de nouvelles images représentant la scène vue d'un nouveau point de vue. Les techniques utilisées pour effectuer une telle opération font partie du domaine de la génération de nouvelles vues et du rendu à base d'images. Différentes approches de ce domaine seront présentées. Tout d'abord, une catégorisation possible de ces méthodes sera présentée. Deux méthodes classiques de rendu à base d'images seront ensuite présentées. Suivront quelques méthodes proposant des variantes de ces deux méthodes classiques. La famille de méthodes basées sur les dictionnaires de parcelles sera aussi présentée, car elle est de plus en plus citée dans la littérature. Finalement, quelques méthodes hybrides seront aussi présentées.

Catégorisation des méthodes de rendu à base d'images

En 2000, Shum et Kang ont fait une revue générale et une catégorisation des méthodes existantes à ce moment [51]. Les méthodes y sont catégorisées en trois grandes familles. Ces familles se distinguent de par leur utilisation et leur représentation de la géométrie de la scène. Les familles d'approches sont les

- Approches sans géométrie, qui ne calculent aucune information géométrique par rapport à la scène ;

- Approches avec géométrie implicite, qui utilisent certains contraintes et font donc une modélisation implicite de la géométrie ;
- Approches avec géométrie explicite, qui modélisent explicitement la géométrie des éléments de la scène.

Ici, le terme géométrie désigne la structure des objets de la scène. Dans ces articles, la configuration des caméras doit être connue.

Selon les auteurs, les approches de la famille sans géométrie tentent de modéliser la fonction plénoptique de façon plus ou moins précise. La fonction plénoptique est une représentation à sept dimensions de l'information lumineuse à tous les points d'un volume déterminé, pour un temps déterminé et selon un intervalle de longueurs d'ondes déterminé (voir [5]). Les principales méthodes de la famille modélisant la fonction plénoptique sont le *Light Field Rendering* [37], le *Lumigraph* [21], les «Concentric Mosaics»[52] et les mosaïques d'images. Selon les auteurs, le principal avantage de ce type de méthodes est l'absence du besoin de la géométrie de la scène, ce qui enlève le besoin d'utiliser un algorithme de stéréovision, et réduit donc le nombre de sources d'erreurs possibles. Par contre, cet avantage induit un inconvénient majeur, qui est le besoin d'une grande quantité d'images en entrée, ce qui rend ce type de systèmes moins pratiques à utiliser dans la réalité.

Les approches à géométrie implicite n'ont pas un modèle explicite de la structure de la scène. Elles utilisent plutôt différentes contraintes qui, misent ensemble, permettent de représenter implicitement la composition de la scène. Certaines informations, telle une carte de correspondances entre les paires d'images, doivent tout de même être connues. Une fois cette carte connue, les méthodes de cette famille sont capables de générer une nouvelle image en utilisant des contraintes sur les relations entre les images en entrée. Les deux méthodes fondatrices de cette famille sont l'interpolation

de vues [11] et la déformation de vues [48]. Ces méthodes ont besoin de moins d'images que les approches sans géométrie. De plus, elles sont plus rapides que les méthodes à géométrie explicite, car elles requièrent moins d'étapes. Il y a également une source d'erreurs de moins que pour les méthodes à géométrie explicite, car il n'y a pas de reconstruction 3D et de construction de maillage. Par contre, pour pouvoir utiliser ces méthodes, l'utilisateur doit posséder une carte de disparités ou de correspondances, et il doit donc dépendre d'une méthode de mise en correspondance et de la précision des résultats de celle-ci.

Finalement, pour les approches à géométrie explicite, les auteurs notent qu'elles procèdent en calculant explicitement l'information géométrique des constituants de la scène. Le plus souvent, cette information est la position 3D des points correspondants à ce qui est projeté sur les pixels des images. Une fois cette information calculée, elle est utilisée de diverses manières pour calculer la nouvelle image. Certaines des méthodes importantes de cette famille sont la déformation 3D [38], les images de profondeur superposées [49] et les cartes de texture dépendantes de l'angle de vue [16, 17]. Ces méthodes ont besoin de peu d'images et elles permettent, dans certains cas, de voir interactivement la nouvelle image. Ce sont aussi les approches qui se mettent en place le plus facilement avec l'aide d'une carte graphique. Cependant, l'utilisateur doit posséder une carte de disparités ou de correspondances, et il dépend donc encore d'une méthode de mise en correspondance et de la précision des résultats de celle-ci.

Dans le contexte de ce mémoire, les approches utilisant une modélisation explicite de la géométrie seront utilisées. En effet, étant donné que les séquences sont déjà acquises et qu'il n'est pas possible d'obtenir de nouvelles informations sur la scène, il est impossible de modéliser la fonction plénoptique. Les méthodes sans géométrie tentant de modéliser cette fonction, elles considèrent implicitement que l'on contrôle

l'acquisition, ce qui n'est pas le cas de là problématique posée ici. De même, il est peu facile d'utiliser une approche à géométrie implicite, puisque la structure des caméras n'est pas connue avec précision. Cela a pour effet de complexifier la définition et l'application des contraintes impliquées dans ce type de modèles, et réduit donc leur efficacité. Ne reste donc que les méthodes à géométrie explicite, qui ont moins de contraintes à satisfaire.

Méthodes classiques

En 1993, Chen et Williams [11] décrivent une méthode afin d'effectuer une transformation entre deux images. Au départ, ils proposent cette technique afin d'interpoler entre deux vues d'une scène, et d'ainsi pouvoir effectuer une transformation entre les points de vue de deux caméras. La méthode peut aussi être utilisée pour interpoler entre deux points de vue sur un même objet. Dans l'article, la méthode n'est présentée que pour des interpolations 1D et 2D, et seulement sur des images synthétiques. Connaissant la correspondance entre les pixels des deux images, la méthode consiste simplement à interpoler linéairement entre les deux positions.

En 1996, Seitz et Dyer [48] proposent une méthode pouvant être utilisée pour changer de point de vue entre deux caméras, transformer entre deux poses du même objet, ou encore entre deux objets différents. Il est nécessaire d'avoir une fonction de correspondance de l'image de gauche vers l'image de droite, et vice-versa. Une fois cette fonction de transfert connue, les images sont interpolées linéairement. Pour calculer la couleur d'un pixel dans l'image interpolée, les auteurs utilisent la moyenne des couleurs des pixels des images interpolées de gauche à droite et de droite à gauche. Les auteurs proposent une première version de l'algorithme nécessitant la connaissance des matrices de projection des deux images. Ils confirment que la méthode de Chen

et Williams [11] est valide dans le cas de caméras parallèles. La deuxième version proposée consiste à rectifier les deux images d'entrée, les interpoler, puis les dérectifier selon une matrice de rectification interpolée à partir des deux matrices de rectification connues.

En résumé, ces deux méthodes classiques nécessitent de connaître une correspondance entre les pixels des images de gauche et de droite, et interpolent de manière différente selon cette correspondance.

Variantes de méthodes classiques

Dans [47], Seitz and Dyer utilisent la méthode de déformation de vues explicitée en [48] afin de découvrir s'il est possible de représenter une scène à l'aide d'images seulement, et non pas de l'entièreté du modèle (comme en infographie, par exemple). Les auteurs définissent une contrainte de monotonie : l'ordre de projection des points doit rester le même d'une image à l'autre. Afin de voir si les images répondent à la contrainte de monotonie, ils proposent de prendre chaque ligne épipolaire de la première image et de vérifier si son conjugué est une déformation monotone de cette même ligne. La conclusion est que, si la contrainte de monotonie est respectée, deux vues perspectives permettent de représenter la scène pour tous les points de vue de trouvant sur la ligne entre les deux centres optiques. Il est donc possible, dans ce cas, d'utiliser la technique de déformation de vues [48] pour générer de nouvelles images de la scène.

L'article [44] décrit une méthode où les auteurs tentent de générer une ou deux images supplémentaires se trouvant entre les images originales. La méthode utilise la déformation de vues [48]. Aucune technique claire de sélection de la distance entre les deux caméras virtuelles n'est donnée dans l'article. Le remplissage des trous est fait en

utilisant la même technique que dans la technique d'interpolation de vues [11].

Méthodes basées sur des dictionnaires de parcelles

Les méthodes basées sur des dictionnaires de parcelles sont de plus en plus citées dans la littérature. Ces méthodes sont de type à géométrie implicite, car elles ne modélisent pas directement la structure de la scène. Elles tentent plutôt de synthétiser les nouvelles images en utilisant de petites parcelles des images d'entrée. Elles peuvent produire des résultats réalistes, mais sont souvent plus lentes que des méthodes à géométrie explicite.

Fitzgibbon *et al.* ont publié une série d'articles sur des méthodes d'interpolation et de génération de vues basées sur des dictionnaires de parcelles d'images. Tout d'abord, dans [64], ils décrivent une méthode utilisant une combinaison de raisonnements géométriques (stéréovision) et un a priori de rendu à base d'images. L'a priori est basé sur la texture de l'image. Le coût des données est fonction de la photoconsistance entre les pixels de sortie et leurs projection dans les images d'entrée, si cette projection n'est pas occultée. La méthode utilise le modèle d'occultation de Wei et Quan [61]. Le coût de lissage est une fonction linéaire basée sur la méthode de Woodford [63] pour pénaliser les discontinuités lorsqu'il n'y a pas de projection dans les images d'entrée. Pour pouvoir construire un graphe soluble avec algorithme de coupe de graphe, les auteurs réduisent la complexité en considérant que, pour le pixel x, tous les autres pixels à une certaine profondeur ont une couleur fixe.

La technique de génération d'une nouvelle vue présentée dans [63] suppose que les matrices de projection des vues initiales sont connues. Une mesure d'énergie est ensuite créée en combinant la photoconsistance du pixel par rapport à son voisinage et à une mesure de similarité de texture. La similarité de texture est évaluée par rapport à une

banque de parcelles estimées à partir des images en entrée. En résumé, cette technique essaie de trouver des zones des images d'entrée semblables à la région courante, et utilise l'information de ces zones pour créer les nouvelles images.

L'article [19] propose une méthode dans la même lignée que [63], tout en précisant certains détails d'implémentation. Les auteurs proposent principalement de construire la banque d'imagettes de référence en se fiant uniquement sur les images en entrée. En effet, dans un monde idéal, ils suggèreraient d'utiliser le plus d'images possible. Cependant, dans la réalité, cela rendrait le problème intraitable à cause du volume de données.

Finalement, l'article [62] décrit une méthode d'accélération du rendu à base d'images avec a priori de texture. La technique consiste à composer des pyramides hiérarchiques de différentes résolutions pour contraindre la recherche aux parcelles près de la parcelle concernée. Il y a un regroupement des parcelles similaires afin de ne pas comparer avec chaque parcelle de la base de données. Selon les résultats présentés, la méthode donne de bons résultats et procure une très grande accélération.

Méthodes hybrides

Certaines méthodes ne peuvent être clairement catégorisées selon leur utilisation de la géométrie. Les articles présentés dans cette section proposent des méthodes hybrides utilisant des concepts venant de différentes méthodes ou d'autres champs de recherche.

Parmi ces méthodes, l'article [50] donne une méthode de génération de plusieurs vues à partir d'images stéréo binoculaires. Pour les auteurs, le but de la méthode est de générer des images qui seront utilisées sur des affichages auto-stéréoscopiques. Ils proposent d'intégrer un algorithme de suivi de mouvement à un algorithme de stéréovision

afin d'améliorer les cartes de disparités. L'algorithme proposé est le *polyline tracking*, qui consiste à détecter les lignes d'une image, à faire une fusion des segments de ligne constituant en fait la même ligne, puis à faire un suivi de mouvement d'une image à l'autre. Les auteurs proposent aussi d'utiliser la carte de disparités calculée au temps t_i pour donner un estimer initial de la carte au temps t_{i+1}. Finalement, le rendu est effectué avec une technique de transfert d'images sans base physique et utilise aussi une reprojection basée sur les disparités.

Une méthode fréquemment citée est celle de Zitnick *et al.* [75] permettant d'effectuer le rendu d'une scène dynamique selon un nouveau point de vue. La méthode est divisée en deux parties : le calcul des disparités et des cartes d'occultations hors-ligne (présenté à la section 1.2.2), puis le rendu en temps réel. Le calcul des cartes de disparités est basé sur une segmentation en régions de couleurs similaires, qui sont ensuite mises en correspondance individuellement. Par la suite, ces correspondances sont affinées en considérant les régions voisines, puis en considérant que les pixels d'une région n'ont pas nécessairement tous la même valeur de disparité. Une technique de transparence d'images est utilisée pour mieux gérer les problèmes de désoccultation sur les bords des objets. Finalement, le rendu est effectué en temps réel, en visionnant la structure reconstruite selon différents points de vue.

Dans l'article [74], Zitnick et Kang proposent de sursegmenter les images afin d'obtenir des résultats plus fiables qu'avec une approche se fiant uniquement sur les pixels. Cette sursegmentation permet aussi de prendre moins de risques de perdre de l'information qu'en utilisant une segmentation normale. Cette segmentation est effectuée en utilisant la technique des k-moyennes sur une image filtrée anisotropiquement. La segmentation est ensuite transformée en un champ de Markov dont les arêtes sont les connexions entre deux segments. Ce champ de Markov est optimisé en considérant

une fonction de coût basée sur un histogramme de confiance de mise en correspondance. Cette approche est appropriée pour le rendu à base d'images, car même si les valeurs de profondeur ne sont pas parfaites, s'il n'y a pas d'artéfacts dans la nouvelle image, les résultats sont considérés comme corrects.

L'article [72] présente une méthode de construction de vues intermédiaires à partir d'une paire d'images stéréoscopiques. Suite à des tests visant à trouver le meilleur modèle statistique d'estimation de disparités par blocs, les auteurs constatent que le modèle laplacien est le plus adapté à ce contexte. En utilisant une mesure de fiabilité développée dans le même article, ils proposent d'effectuer l'interpolation de vues en projetant les images originales selon les cartes de disparités, puis de remplir les artéfacts de désoccultation en interpolant de droite à gauche. Cependant, cela ne fonctionne qu'avec une configuration de caméras parallèles.

Dans l'article [43], les auteurs présentent une méthode de génération de vues basée sur des images stéréoscopiques. La méthode est axée sur la création d'images s'adaptant au point de vue du spectateur, dans un contexte de téléconférence. Il y a utilisation d'un dispositif d'acquisition de la position de la tête, afin d'ajuster l'image au point de vue du spectateur.

Il est important de noter, que peu importe la méthode de rendu à base d'images choisie pour générer les nouvelles images, certains artéfacts peuvent être créés. L'importance de ces artéfacts variera selon le contenu de la scène et la méthode choisie.

1.2.4 Emplissage d'image

Lorsque des nouvelles images de la scène ont été calculées avec une méthode de rendu à base d'images, elles contiennent pratiquement toujours certains artéfacts et certaines

zones vides. Ces zones sont dues au manque d'information lorsque le point de vue est changé. Étant donné que les séquences doivent être visualisées en 3D, il est impératif de remplir ces zones afin de ne pas avoir de désagréments lors du visionnement. Pour ce faire, il est possible d'utiliser une technique d'emplissage d'image (« image inpainting »). Les techniques d'emplissage d'image ont plusieurs utilisations dans le domaine de l'imagerie numérique. Elles peuvent par exemple être utilisées pour réduire ou éliminer des déchirures dans l'image d'une vieille photographie numérisée, ou encore pour faire disparaître un objet indésirable d'une photo de famille. Un résumé de quelques techniques importantes sera présenté, de même que certaines techniques spécialement adaptées au rendu à base d'images.

En 2003, Levin *et al.* [36] constatent que l'emplissage d'image basé sur le contenu local est relativement performant, mais que celui-ci n'est basé que sur le voisinage de la région à remplir. Ainsi, si deux images ont un contenu différent, mais que la forme et le voisinage de la région à emplir dans chacune d'elle est identique, l'algorithme produira le même résultat, même si celui-ci s'intègre mal à une des deux images. Ils proposent donc d'aussi considérer le contenu structurel global de l'image, en considérant un histogramme des caractéristiques de l'image et en utilisant un algorithme d'apprentissage sur une banque d'images.

La même année, Bertalmio *et al.* [7] proposent une méthode d'emplissage d'image basée à la fois sur la structure et la texture de l'image. L'algorithme consiste à décomposer l'image d'entrée en une image de texture et une image de structure, puis à emplir indépendamment ces deux images avec un algorithme adapté à chacune d'elle. L'image de structure est emplie avec un algorithme basique d'emplissage, tandis que l'image de texture est emplie par synthèse de texture. Finalement, les deux images sont recombinées pour former l'image finale. Les auteurs expliquent que les résultats

sont plus réalistes car l'action indépendante des deux algorithmes est plus précise et ciblée qu'en utilisant un seul algorithme pour tout le contenu de l'image.

Toujours en 2003, l'article de Criminisi *et al.* [12] décrit une méthode utilisant de petites parcelles de l'image d'entrée pour éliminer un objet indésirable. La région d'intérêt est décrite par un masque binaire. L'algorithme fonctionne sur la base de petites parcelles. Le contour de la région à emplir est parcouru en considérant, pour chaque pixel du contour, une parcelle contenant à la fois de l'information de l'image d'entrée et une partie de la région à remplir. Ce parcours permet de d'identifier la parcelle à remplir avec la plus haute priorité. Celle-ci est choisie par rapport au nombre de pixels à emplir sur la parcelle et par rapport aux contours entrant dans la région à cet endroit. Cette technique a pour effet d'aider à propager l'information de contour (et donc structurelle) à l'intérieur de la région à emplir. Une fois la parcelle à emplir choisie, l'algorithme cherche dans toute l'image la parcelle la plus similaire à celle à emplir. Le contenu de la parcelle la plus similaire est ensuite utilisé pour emplir les pixels manquants dans la parcelle prioritaire. Le processus est finalement répété jusqu'à ce que la région soit complètement emplie.

Tauber *et al.* [56] ont effectué une revue et des prévisions sur les techniques d'emplissage pour compenser les artéfacts de désoccultation dans le cadre du rendu à base d'images. Leur principal apport est de définir une classification pour les méthodes d'emplissage d'image et d'analyser certaines approches par rapport à cette classification. Leur classification divise les techniques selon qu'elles sont basées sur l'utilisation d'information structurelle ou d'information de texture. Suite à cette classification, ils constatent qu'une méthode utilisant à la fois (ou de manière séparée) la structure et la texture de l'image devrait produire de bons résultats. Ils proposent donc une telle méthode, méthode semblable à celle proposée par Bertalmio *et al.* [7].

Dans leur article de 2008, Wang *et al.* (voir [60]) décrivent une méthode permettant, à partir d'une paire d'images stéréoscopiques, de compléter à la fois la couleur et la profondeur. Ils proposent de compléter les régions d'occultation avec une approche d'emplissage de profondeur basée sur une segmentation. Pour ce faire, les images couleurs complètes et des cartes de disparités avec l'information d'occultation doivent être disponibles. Cette étape permet d'améliorer les cartes de disparités calculées. Par la suite, l'utilisateur identifie la région à enlever et l'information est géométriquement transformée d'une image à l'autre. Les pixels n'ayant pu être emplis le sont par synthèse de texture.

CHAPITRE 2

Développement mathématique et géométrique du modèle de stéréoscopie

Afin d'obtenir un effet 3D contrôlé et de réduire au minimum les distorsions et inconforts visuels d'une séquence stéréoscopique, les paramètres d'acquisition doivent être choisis avec soin. Un modèle mathématique et géométrique des schémas d'acquisition et de visionnement est présenté. À partir de ce modèle et de contraintes perceptuelles, des relations entre les paramètres sont mises en évidence, et celles-ci permettent de décrire le positionnement optimal des caméras pour réduire les distorsions.

2.1 Concepts de stéréoscopie

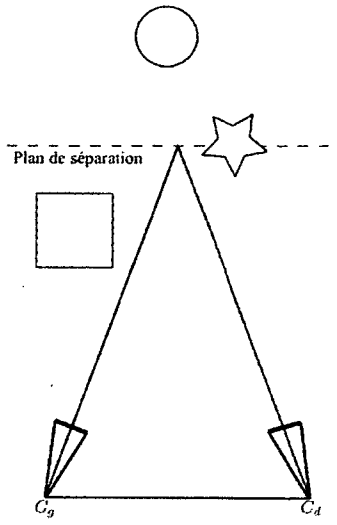
Lors du tournage d'une séquence stéréoscopique, le réalisateur doit décider à l'avance de l'effet 3D qu'il désire donner au spectateur. Pour ce faire, il doit considérer l'emplacement du plan de séparation de l'effet 3D, le volume d'acquisition et l'amplitude de l'effet 3D produit.

2.1.1 Plan de séparation de l'effet 3D

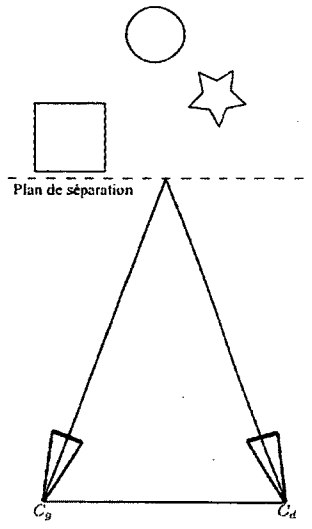
Le plan de séparation de l'effet 3D représente la limite imaginaire entre ce qui sera perçu devant l'écran et ce qui sera perçu derrière l'écran lors de la visualisation de la séquence (voir la figure 2.1). Étant donné les phénomènes de convergence et d'ajustement de la vision, il est recommandé que le plan de séparation soit placé à l'endroit où l'activité principale de la séquence se déroule (voir [25]). Ainsi, le système visuel du spectateur ne sera pas soumis à des indices de profondeur contradictoires, ce qui réduira la fatigue et l'inconfort lors du visionnement. Le réalisateur doit donc déterminer l'emplacement spatial où l'attention du spectateur doit être dirigée.

2.1.2 Volume d'acquisition et amplitude de l'effet 3D

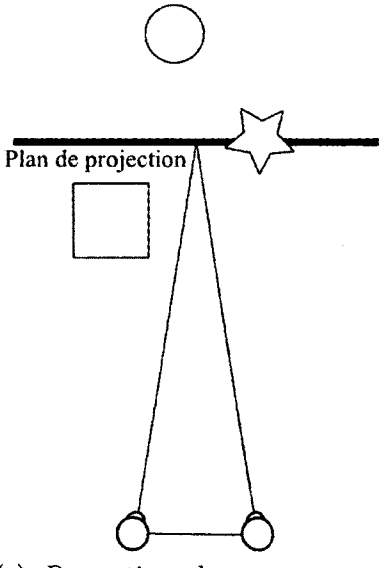
Les paramètres d'acquisition doivent être ajustés selon le volume englobant la scène perçue par le spectateur. Ce volume influence le choix des paramètres d'acquisition, car s'ils sont mal choisis, le spectateur peut percevoir le volume sous une forme compressée ou dilatée. Ces phénomènes de distorsions sont abordés de manière détaillée à la section 2.3.1. Ces types de distorsions nuisent grandement à la qualité et au réalisme de l'expérience 3D. Cependant, dans certains cas, le réalisateur peut vouloir



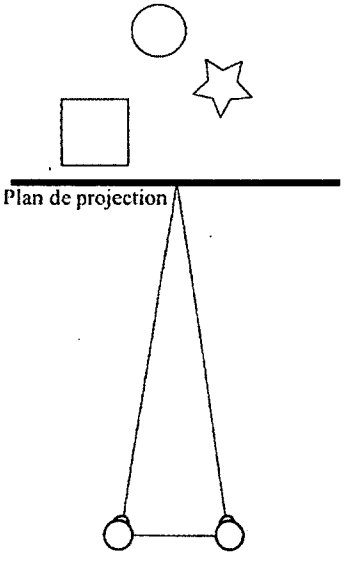
(a) Plan de séparation dans la scène - configuration 1



(b) Plan de séparation dans la scène - configuration 2



(c) Perception du spectateur - configuration 1



(d) Perception du spectateur - configuration 2

Figure 2.1 – Effet du plan de séparation sur la perception 3D : les figures (a) et (b) représentent deux configurations du plan de séparation 3D par rapport aux objets de la scène, tandis que les figures (c) et (d) représentent l'effet 3D perçu par le spectateur par rapport au plan de projection.

accentuer ou diminuer l'effet tridimensionnel, afin de créer un effet visuel percutant ou amusant. Il y a donc un choix à faire entre l'espace filmé et l'amplitude de l'effet 3D produit.

2.2 Modèle d'acquisition stéréoscopique

Un modèle géométrique est nécessaire afin d'exprimer mathématiquement la configuration des caméras. Ce modèle doit permettre de trouver la configuration idéale des caméras induisant le moins de distorsion et d'inconfort visuel possible, étant donnée la position des objets dans la scène. Les éléments de base du modèle proposé, qui s'inspire de [6, 30, 65], sont illustrés à la figure 2.2.

Les principales variables sont θ_i, f_i, k_i, B, D et A. Ici, les indices i représentent une des deux caméras, soit C_g ou C_d, les caméras de gauche et droite, respectivement. B représente la distance entre les deux caméras. θ_i indique l'orientation de la caméra par rapport à la ligne de base, f_i est la longueur focale de la caméra, et k_i est la largeur du plan image. En général, les caméras C_g et C_d prennent les mêmes valeurs de f_i et k_i afin d'éviter un inconfort oculaire. Pour la même raison, les deux caméras sont généralement situées à distance égale de l'origine du repère, et $|\theta_g| = |\theta_d|$. Dans cette notation, A représente la norme du vecteur perpendiculaire à la ligne de base et se rendant jusqu'au plan de séparation de l'effet 3D, tandis que D représente la norme du vecteur entre le centre optique d'une caméra et l'intersection de A et du plan de séparation de l'effet 3D.

Une matrice de projection de caméra sous sa forme classique permet de transformer un point 3D $\mathbf{P}$ vers un pixel $\mathbf{p}$ (voir [23]). Cette matrice de projection est une matrice

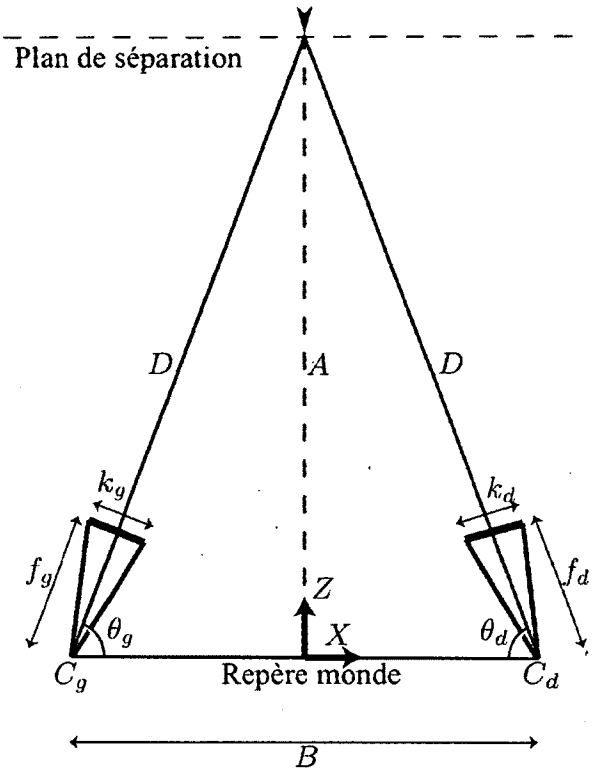


Figure 2.2 – Configuration de base pour l'acquisition stéréoscopique.

homogène 3×4 nommée $\mathtt{M}$, et l'opération de projection est exprimée par

$$\mathbf{p} = \mathtt{M}\mathbf{P} \tag{2.1}$$

où $\mathtt{M}$ est

$$\mathtt{M} = \mathtt{K}[\mathtt{R}|\mathtt{t}] \tag{2.2}$$

avec $\mathtt{K}$ la matrice des paramètres intrinsèques de la caméra, $\mathtt{R}$ la matrice de rotation associée à la caméra et

$$\mathbf{t} = -\mathtt{R}\tilde{\mathbf{C}} \tag{2.3}$$

avec $\tilde{\mathbf{C}}$ la position de la caméra dans le repère monde.

Dans le contexte des acquisitions stéréoscopiques utilisées pour le cinéma 3D, une contrainte est qu'il ne doit pas y avoir de disparité verticale entre les deux images, faute de quoi l'expérience 3D serait diminuée et inconfortable. Une conséquence de cette contrainte est qu'il est possible de représenter l'orientation des caméras comme étant une rotation autour d'un seul axe du repère d'origine. Dans le repère présenté à la figure 2.2, cette rotation s'effectue autour de l'axe Y. Il est donc possible de restreindre $\mathtt{R}_i$ à

$$\mathtt{R}_i = \begin{bmatrix} cos(\theta_i) & 0 & sin(\theta_i) \\ 0 & 1 & 0 \\ -sin(\theta_i) & 0 & cos(\theta_i) \end{bmatrix}. \tag{2.4}$$

Une mâtrice K de paramètres intrinsèques est donnée par

$$\mathtt{K} = \begin{bmatrix} f_x & s & x_0 \\ 0 & f_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.5}$$

où (x_0, y_0) est le centre optique de la caméra en coordonnées pixel, $f_x = -f/m_x, f_y = -f/m_y$, où f est la longueur focale de la caméra et m_x, m_y sont les dimensions horizontale et verticale d'un pixel du capteur. Finalement, s représente le facteur de biais de la caméra. Ce facteur permet de corriger une orientation non perpendiculaire des axes x et y du capteur de la caméra. Pour la plupart des caméras fréquemment utilisées, $s = 0$.

2.2.1 Description de la configuration stéréoscopique par les matrices de projection

Selon les paramètres de la configuration stéréoscopique, on peut déduire les matrices de projection des deux caméras réelles.

Paramètres intrinsèques

En général, (x_0, y_0) est égal à la moitié du nombre de colonnes et de lignes de l'image. Dans ce cas, en supposant des pixels carrés ($m_x = m_y$), il est clair que

$$f_x = f_y = \hat{f} \tag{2.6}$$

et que

$$m_x = \frac{k}{2x_0} \implies \hat{f} = \frac{-2fx_0}{k} \tag{2.7}$$

La matrice K construite avec $\hat{f}$ est notée $\mathtt{K}_{\hat{f}}$.

Paramètres extrinsèques

Afin d'obtenir le meilleur effet tridimensionnel, les caméras doivent être placées à égale distance du point d'intersection de la ligne de base et du vecteur perpendiculaire à celle-ci et joignant le plan de séparation de l'effet 3D. Il s'agit donc de les placer à une distance $\frac{B}{2}$ de ce point d'intersection. Pour les mêmes raisons de diminution de distorsion, l'amplitude de θ_g doit être la même que l'amplitude de θ_d. En déterminant que la rotation s'effectue autour de l'axe Y, et qu'originalement, les caméras regardent

en direction des valeurs positives de Z, les matrices de projection des caméras réelles sont

$$\mathtt{M}_g = \mathtt{K}_{\hat{f}} \left[\mathtt{R}_{\frac{\pi}{2}-\theta} | \mathtt{t}_{-B/2} \right], \tag{2.8}$$

$$\mathtt{M}_d = \mathtt{K}_{\hat{f}} \left[\mathtt{R}_{\theta-\frac{\pi}{2}} | \mathtt{t}_{B/2} \right]. \tag{2.9}$$

2.3 Modèle de visualisation stéréoscopique

De la même manière qu'un modèle géométrique est utilisé pour représenter la configuration d'acquisition, un modèle est utilisé pour illustrer la configuration de visualisation. Le modèle est similaire à celui utilisé lors de l'acquisition de la séquence, tout en étant plus simple. Les paramètres considérés dans ce modèle sont illustrés à la figure 2.3.

Dans ce modèle, B' représente la distance interoculaire du spectateur et θ' représente l'angle de vision du spectateur lorsqu'il fixe le centre de la surface de projection. De même, L est la largeur de cette surface de projection, A' est la distance entre le spectateur et la surface de projection, et D' est la distance entre un oeil du spectateur et la surface de projection.

2.3.1 Phénomène d'agrandissement

La fatigue oculaire, la sensation d'irréalisme et l'inconfort lors du visionnement d'une séquence stéréoscopique sont souvent créés par un phénomène d'agrandissement ou de réduction de la scène. En effet, il est possible d'utiliser la technique d'agrandissement

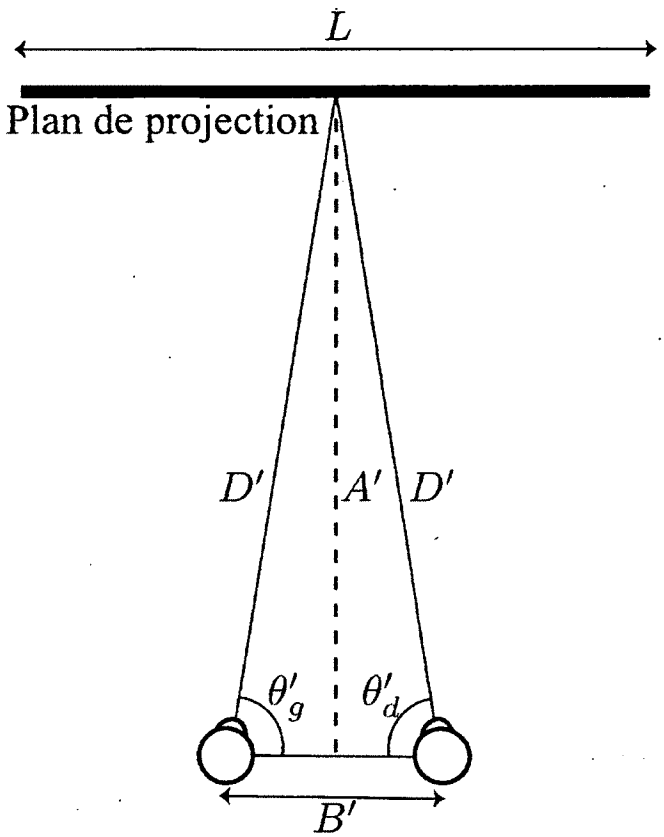
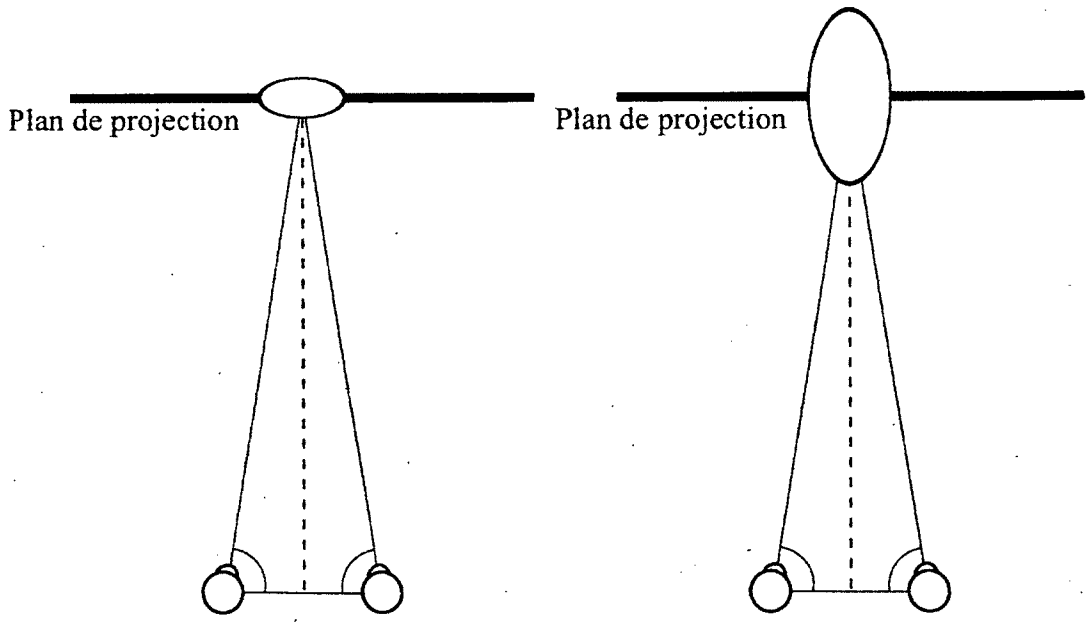


Figure 2.3 – Configuration de base pour le visionnement stéréoscopique.

(«zoom») existant pour une séquence classique dans une séquence stéréoscopique. Cependant, comme la scène est perçue en trois dimensions, l'agrandissement peut lui aussi être perçu en 3D. Si les paramètres sont mal calculés, la valeur d'agrandissement peut ne pas être la même le long des trois axes, ce qui est entraîne une perte de réalisme lors du visionnement. Par exemple, un objet sphérique pourrait paraître aplati comme un disque ou étiré comme un ballon de football (voir figure 2.4). Il faut donc éviter cet effet.

Selon la catégorisation de Ariyaeeinia [6], il existe deux types de phénomènes d'agrandissement : l'agrandissement de profondeur m_z, et l'agrandissement sur le plan XY,



(a) Compression de la profondeur perçue (b) Dilatation de la profondeur perçue

Figure 2.4 – Distorsions possibles de la profondeur : (a) compression de la profondeur de la séquence lorsque $m_z < m_{xy}$, (b) dilatation de la profondeur de la séquence lorsque $m_z > m_{xy}$.

m_{xy}. L'agrandissement de profondeur implique que l'échelle de la scène sur l'axe Z est modifiée. Ce type de distorsion peut entraîner un effet de théâtre de marionnettes, où tous les objets semblent plats, ou une exagération de l'effet de profondeur, où les objets semblent plus longs qu'ils ne le sont réellement. L'agrandissement sur le plan XY, ou plan de projection, est similaire à un effet de «zoom» dans une séquence monoscopique.

Les problèmes de visionnement et de perception apparaissent lorsque l'agrandissement / réduction est anisotropique, c'est-à-dire quand

$$\frac{m_z}{m_{xy}} \neq 1 \tag{2.10}$$

Le ratio $\frac{m_z}{m_{xy}}$ est appelé R_a. Selon l'équation (2.10), le phénomène d'aplatissement apparaît lorsque $R_a < 1$, et le phénomène d'étirement apparaît lorsque $R_a > 1$. Par exemple, si B' est plus grand que ce que les relations mises en évidence à la section 2.3.2 suggèrent, et que les autres paramètres suivent ce qui est recommandé par ces relations, le phénomène d'aplatissement se produit. Au contraire, si B' est plus petit que supposé lors de l'acquisition, il y a étirement de la profondeur des objets.

Toujours selon [6], il est possible de montrer que $m_z(Z)$ et $m_{xy}(Z)$, les facteurs d'agrandissement d'un objet situé à une distance Z d'une caméra, sont respectivement

$$m_{xy}(Z) = \frac{Gf}{Z} \tag{2.11}$$

et

$$m_z(Z) = \frac{D'BfG}{B'Z^2} \tag{2.12}$$

où G est le facteur d'échelle entre le plan image et la surface de projection, Z la distance de l'objet, et f, D' et B sont les paramètres décrits aux sections 2.2 et 2.3.

2.3.2 Relations entre les paramètres d'acquisition et de visionnement

Dans un contexte normal, un réalisateur de séquence stéréoscopique choisit la configuration de ses caméras de manière à minimiser la distorsion des objets se situant sur le plan de séparation de l'effet 3D. En effet, c'est normalement là que l'activité de la scène est concentrée, et c'est à cet endroit que le spectateur focalise son regard. Il importe donc que la distorsion soit minimale pour les objets se situant à $Z = D$. Connaissant les paramètres moyens de visionnement, c'est-à-dire B' et D', de même que la distance entre une caméra et le plan de séparation D, et suivant un concept de triangles semblables, les paramètres de configuration des caméras doivent être calculés selon

$$B = \frac{B'D}{D'}, \tag{2.13}$$

$$\theta = \cos^{-1}\left(\frac{B}{2D}\right) \tag{2.14}$$

afin de minimiser la distorsion des objets au plan de séparation. De même, il est impossible de minimiser la distorsion des objets au plan de séparation si l'angle θ des caméras est différent de l'angle θ' de visionnement. En effet, il est possible de remarquer que

$$B = \frac{B'D}{D'} \implies \frac{B}{D} = \frac{B'}{D'} \tag{2.15}$$

et donc, que

$$\theta = \cos^{-1}\left(\frac{B}{2D}\right) = \cos^{-1}\left(\frac{B'}{2D'}\right) \implies \theta = \theta'. \tag{2.16}$$

Afin de minimiser la distorsion du plan XY, il faut considérer la géométrie de l'espace d'acquisition et celle de l'espace de visionnement présentées aux figures 2.2 et 2.3. Le triangle de la caméra (déterminé par la largeur du capteur k et la longueur focale f) et le triangle spectateur-surface de projection (déterminé par la largeur de la surface L et la distance de oeil-surface de projection D') doivent être approximativement similaires pour minimiser cette distorsion. Cette relation est exprimée par

$$\frac{f}{k} \approx \frac{D'}{L} \rightarrow f \approx \frac{kD'}{L}. \tag{2.17}$$

CHAPITRE 3

Processus de synthèse de nouvelles vues

Maintenant qu'un modèle pour les paramètres d'acquisition et de visionnement est en place, il faut procéder à la reconfiguration des séquences originales. La séquence reconfigurée doit recréer le même effet 3D, dans de nouvelles conditions de visionnement, que la séquence originale dans les conditions originales. Dans ce chapitre, le cadre de traitement général est expliqué, puis chaque étape est détaillée.

3.1 Cadre de traitement

Le cadre de traitement proposé comporte cinq étapes telles que schématisées à la figure 3.1. Chacune de ces étapes possède certaines possibilités de variation. Il faut tout d'abord calculer ou retrouver les paramètres d'acquisition originaux (section 3.2), qui seront ensuite utilisés lors du calcul des paramètres virtuels nécessaires à

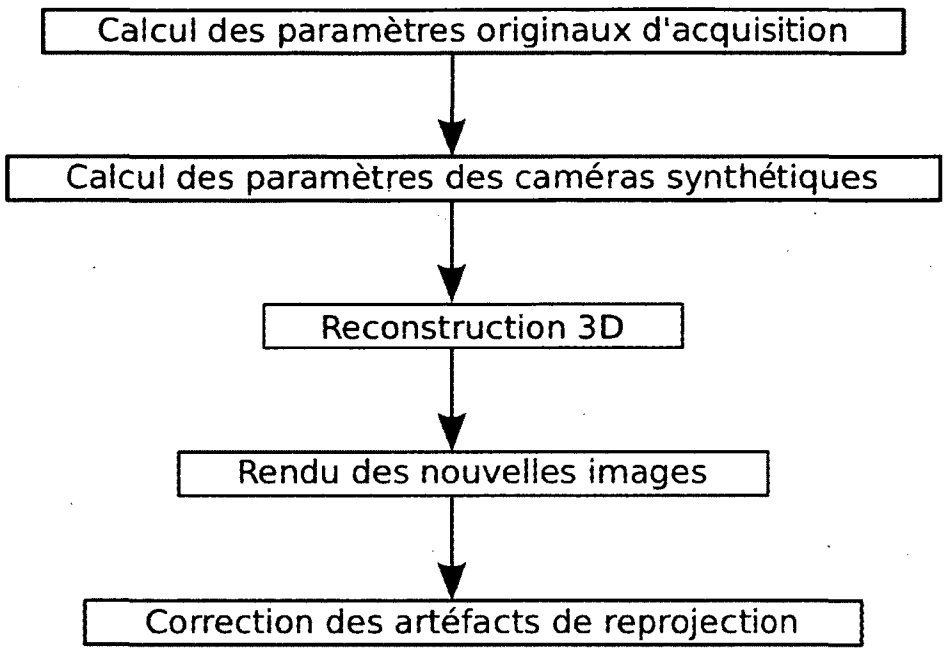


Figure 3.1 – Diagramme des étapes de traitement pour la reconfiguration stéréoscopique.

la bonne visualisation dans de nouvelles conditions (section 3.3). Par la suite, il faut pouvoir obtenir une représentation tridimensionnelle du contenu de la scène (section 3.4), qui sera reprojetée sur les caméras virtuelles à l'étape suivante (section 3.5). Finalement, les images obtenues du processus de reprojection contiendront certains artéfacts, principalement dûs au problème de désoccultation. Il faut donc éliminer ces artéfacts en utilisant une technique d'emplissage de trous (« inpainting », section 3.6).

Le cadre proposé a été choisi parce que la séparation des étapes s'agence avec les données qui sont disponibles au fil des différentes étapes. Au tout début, seuls les paramètres de visionnement des deux configurations sont connus. De ceux-ci, on peut déduire les paramètres originaux d'acquisition (étape 1). Ensuite, on peut déduire les paramètres virtuels d'acquisition selon les paramètres déduits à l'étape 1 et les

paramètres de visionnement de la nouvelle configuration (étape 2). Une fois tous les paramètres calculés, on retrouve la structure de la scène (étape 3), et de celle-ci, on calcule les images virtuelles (étape 4). On termine en corrigeant les artéfacts présents dans ces images (étape 5). Il y a donc un ordonnancement qui apparaît naturellement. De plus, comme les étapes représentent des divisions atomiques du processus, il est possible de remplacer l'algorithme d'une étape par un nouvel algorithme. Cela permet donc au cadre de traitement de représenter les étapes à faire tout en permettant des améliorations progressives selon l'évolution du domaine.

3.2 Calcul des paramètres des caméras originales

Afin de reconfigurer une séquence stéréoscopique en fonction de nouveaux paramètres de visionnement, il est nécessaire de connaître les paramètres d'acquisition originaux afin d'en déduire les nouveaux paramètres. Ces paramètres originaux se divisent en deux : (1) les paramètres reliés à la scène et (2) les paramètres de visionnement originalement choisis. Dans le cadre de ce travail, les paramètres ciblés pour le visionnement sont considérés comme connus au moment de la reconfiguration. En effet, s'ils n'étaient pas connus, il n'y aurait que les rares cas où les paramètres originaux de la scène sont connus où il aurait été possible de procéder à une reconfiguration de qualité. De plus, il est fréquent d'obtenir les paramètres ciblés pour le visionnement, que ce soit via des métadonnées ou par interaction avec le client demandant une reconfiguration. Par exemple, le client peut savoir que son film a été tourné pour un écran de taille IMAX© et ainsi en déduire la distance moyenne entre le spectateur et l'écran, de par les standards IMAX©.

Dans certains cas, il est possible de récupérer les paramètres originaux de la scène. Par

exemple, dans le cadre d'un film d'animation, il est fort probable que ces paramètres soient disponibles dans les métadonnées de la séquence, ou encore dans un fichier externe, comme les notes de production. De même, il est de plus en plus fréquent que certains paramètres, tels l'angulation des caméras, leur longueur focale et leur distance de séparation soient enregistrées à même les métadonnées de la séquence. Cependant, selon divers intervenants du milieu, les pipelines de post-traitement ne sont pas tous adaptés à ce type d'information, et celle-ci est souvent perdue en cours de traitement.

Il serait aussi possible de retrouver les paramètres originaux des caméras si un objet de calibration connu était placé dans la séquence. Cependant, cette technique est impossible à utiliser dans un contexte normal, car l'objet de calibration ne peut pratiquement jamais être intégré à la séquence sans en briser l'aspect réaliste et sans perturber l'expérience de visionnement.

3.2.1 Calculs géométriques

Dans les cas où les paramètres originaux de la scène sont inconnus et où aucun objet de calibration n'est présent dans la scène, il est notoire que les matrices des paramètres de caméra $\mathtt{M}_g$ et $\mathtt{M}_d$ (voir équations (2.8), (2.9)) ne peuvent être retrouvées qu'à une transformation projective près (voir [23]).

Notons qu'à moins d'une grave distorsion géométrique ou optique de la caméra, le centre optique (x_0, y_0) est égal à la moitié de la résolution de la séquence. Par conséquence, $x_0 = \frac{X}{2}$ et $y_0 = \frac{Y}{2}$, où $Y \times X$ est la taille des images de la séquence. De plus, puisqu'il est supposé que les paramètres ciblés initialement pour le visionnement sont disponibles, L, D' et B' sont connus.

Notons aussi qu'il est impossible de retrouver avec précision la valeur originale de B. Tel qu'expliqué à la section 3.4, la valeur de B peut être arbitrairement fixée sans que cela n'affecte les étapes subséquentes.

Cas sans distorsion

Si la séquence a été acquise en utilisant les recommandations de la section 2.3.2, c'est-à-dire en minimisant la distorsion des objets au plan de séparation, il est possible de retrouver les valeurs originales de θ et $\hat{f}$ pour la scène, uniquement par calculs géométriques. En combinant les équations (2.13) et (2.14), il en découle que

$$\begin{aligned} \theta &= \cos^{-1}\left(\frac{B'D}{2DD'}\right) \\ &= \cos^{-1}\left(\frac{B'}{2D'}\right). \end{aligned} \tag{3.1}$$

De même, en combinant les équations (2.7) et (2.17), $\hat{f}$ peut être estimé selon

$$\begin{aligned} \hat{f} &= \frac{-2kD'x_0}{Lk} \\ &= \frac{-2D'x_0}{L}. \end{aligned} \tag{3.2}$$

Il est remarquable que, dans les cas sans distorsion imposée, $\hat{f}$ et θ ne dépendent que des paramètres de visionnement, à savoir B', D' et L.

Cas avec distorsion imposée

Dans certains cas, il est possible que le réalisateur, dans un élan artistique, ou pour créer un effet spectaculaire, décide d'utiliser des paramètres d'acquisition induisant

sciemment une distorsion visuelle. Par exemple, l'effet voulu pourrait être d'accentuer temporairement l'épaisseur du personnage principal, pour démontrer l'effet d'une action sur lui (pensons à un film de super-héros où le personnage utilise son pouvoir). Ces choix de paramètres impliquent que $R_a \neq 1$. Dans ces cas, les déductions de la section 3.2.1 ne sont plus valides.

Afin de trouver une technique pour déduire les valeurs de θ et f ayant été utilisées, rappelons que

$$R_a(Z) = \frac{m_z}{m_{xy}} = \frac{D'BfL}{B'Z^2k} \cdot \frac{kZ}{Lf} \tag{3.3}$$

qui, en simplifiant et en évaluant en $Z = D$ (donc, pour les objets situés sur le plan de séparation de l'effet 3D), devient

$$R_a(D) = \frac{D'B}{B'D}. \tag{3.4}$$

Ici, B et D sont inconnus, et B', D' sont connus. Il est primordial, même dans le cas d'une acquisition ne minimisant pas la distorsion, que l'intersection des axes optiques des caméras se trouve à l'intersection de A et du plan de séparation de l'effet 3D. Cela implique que la distance entre C_g et le plan de séparation de l'effet 3D est la même que celle entre C_d et le plan de séparation. En effet, si cette contrainte n'est pas respectée, le spectateur ne percevra pas les objets au bon endroit par rapport à l'écran. Cette contrainte implique que

$$\theta = \cos^{-1}\left(\frac{B}{2D}\right) \implies 2\cos(\theta) = \frac{B}{D} \tag{3.5}$$

et donc, que

$$
\begin{aligned}
R_a(D) &= \frac{2D' \cos(\theta)}{B'} \\
\cos(\theta) &= \frac{B' R_a(D)}{2D'} \\
\theta &= \cos^{-1}\left(\frac{B' R_a(D)}{2D'}\right).
\end{aligned} \tag{3.6}
$$

De manière intéressante, le lecteur remarquera que l'équation (3.1) est un cas particulier de l'équation (3.6) quand $R_a(D) = 1$.

Pour ce qui est de f, il faut considérer qu'il est plus facile d'estimer globalement R_a que d'estimer individuellement m_{xy} et m_z. Par exemple, il est plus intuitif pour un humain d'estimer qu'un objet sphérique a été étiré par un facteur $R_a = 2$ que d'estimer que $m_{xy} = 0.4$ et $m_z = 0.8$. Étant donné $R_a = \frac{m_z}{m_{xy}}$, les deux facteurs de magnification m_z, m_{xy} sont liés. Ayant estimé une valeur de R_a, il est possible de fixer m_z à une valeur arbitraire afin d'estimer m_{xy}. En utilisant 1 comme valeur arbitraire, on a

$$
R_a = \frac{1}{m_{xy}} \implies m_{xy} = \frac{1}{R_a}. \tag{3.7}
$$

Maintenant, étant donné que $m_{xy}(Z = D) = \frac{Lf}{kD}$, il appert que

$$
f = \frac{m_{xy} k D}{L} \tag{3.8}
$$

et, en combinant avec (3.7), que

$$
f = \frac{kD}{L R_a}. \tag{3.9}
$$

Comme D est inconnu, on doit trouver une manière de l'éliminer. Étant donné que θ a été estimé à l'équation (3.6), et que $\cos(\theta) = \frac{B}{2D} \implies D = \frac{B}{2\cos(\theta)}$, il est possible d'obtenir, en remplaçant D,

$$f = \frac{kB}{2LR_a \cos(\theta)}. \tag{3.10}$$

Si l'on veut obtenir $\hat{f}$, la distance focale en dimension pixel, il suffit d'utiliser l'équation (2.7), et l'on obtient

$$\hat{f} = \frac{-Bx_0}{LR_a \cos(\theta)}. \tag{3.11}$$

Comme dans le cas de θ, le lecteur remarquera que (3.2) est un cas spécial de (3.11) lorsque $R_a = 1$, c'est-à-dire

$$\hat{f} = \frac{-Bx_0}{L \cos(\theta)}. \tag{3.12}$$

Selon la structure classique des caméras, $\cos(\theta) = \frac{B}{2D} \implies \frac{B}{\cos(\theta)} = 2D$. Dans le cas sans distorsion, les équations (2.15) et (2.16) permettent de substituer B, D et θ par B', D' et θ', et ainsi, d'obtenir

$$\frac{B'}{\cos(\theta')} = 2D'. \tag{3.13}$$

Considérant le cas sans distorsion, il est possible de remplacer $\frac{B}{\cos(\theta)}$ dans (3.12) et d'obtenir

$$\hat{f} = \frac{-2D'x_0}{L}, \tag{3.14}$$

qui est exactement le résultat obtenu à l'équation (3.2).

Finalement, en vertu des équations (3.6) et (3.10), on constate que θ et f dépendent de R_a. Il est possible que ce facteur soit déjà disponible, par exemple dans les métadonnées de la séquence ou si celle-ci est reconfigurée avec la participation du réalisateur. Si ce n'est pas le cas, il faut qu'un humain l'estime du mieux qu'il peut.

3.3 Calcul des paramètres des caméras virtuelles

Une fois les paramètres originaux d'acquisition retrouvés ou estimés, il faut calculer les valeurs que ces paramètres auraient dû prendre pour procurer la meilleure expérience possible étant donnée la nouvelle configuration de visionnement. Formellement, les caméras originales C_g et C_d ont été configurées en considérant un écran de largeur L et une distance entre l'oeil du spectateur et l'écran D'. Étant donné une nouvelle grandeur d'écran L^n et une nouvelle distance oeil-écran D'^n, il faut calculer les valeurs des paramètres B^v, $\hat{f}^v$ et θ^v nécessaires pour deux caméras virtuelles C_g^v et C_d^v. Ces caméras virtuelles représentent l'acquisition telle qu'elle aurait dû être faite pour optimiser l'expérience 3D dans la nouvelle configuration.

Il existe deux cas de figure. Tout d'abord, si l'acquisition a été faite dans le but de minimiser la distorsion, les formules sont directes. Si, au contraire, le créateur de la séquence voulait créer un effet de distorsion spécifique, il faut imposer une formulation différente, tel qu'expliqué à la section 3.3.2.

3.3.1 Cas sans distorsion

Dans le cas d'une acquisition effectuée en utilisant des paramètres minimisant la distorsion lors du visionnement, le but est tout simplement de calculer les paramètres des caméras virtuelles permettant de minimiser la distorsion dans la nouvelle configuration de visionnement. Pour ce faire, en supposant que la distance interoculaire B' ne varie pas d'un observateur à l'autre, et en utilisant l'équation (3.1), on en déduit que l'angle de convergence des caméras virtuelles est

$$\theta^v = \cos^{-1}\left(\frac{B'}{2D'^n}\right). \tag{3.15}$$

De même, en utilisant l'équation (3.2), on trouve que

$$\hat{f}^v = \frac{-2D'^n x_0}{L^n}. \tag{3.16}$$

Finalement, la distance plan de séparation-ligne de base A (voir la figure 2.2) ne change pas lors de la reconfiguration. En effet, comme le but de la reconfiguration est de procurer la même expérience de visionnement dans une nouvelle configuration, la distance entre la ligne de base et le plan de séparation de l'effet 3D ne peut être changée. Si elle était changée, l'effet 3D serait modifié de manière appréciable. De manière concrète, les objets perçus n'auraient plus la même relation de position par rapport à la surface de visualisation, ce qui est indésirable dans le contexte de la reconfiguration.

À l'équation (2.13), B est défini comme $B = \frac{B'D}{D'}$. Étant donné que la structure décrite dans les cas sans distorsion se base sur des triangles rectangles semblables, B peut aussi être défini comme

$$B = \frac{B'A}{A'}. \tag{3.17}$$

Dans ce cas, B^v peut être obtenu en utilisant l'équation suivante

$$B^v = \frac{B'A}{A'^n}. \tag{3.18}$$

En isolant A dans les équations (3.17) et (3.18), il apparaît que

$$\frac{BA'}{B'} = A = \frac{B^v A'^n}{B'}, \tag{3.19}$$

et donc,

$$B^v = \frac{BA'}{A'^n}. \tag{3.20}$$

Ici, B^v est défini par rapport à B. Or, tel que précisé à la section 3.2.1, B ne peut être retrouvé avec précision sans connaissance *a priori* de la structure de la caméra ou sans la présence d'un objet de calibration dans la séquence. Cela semblerait indiquer que B^v ne peut pas non plus être calculé avec précision. Dans le cadre de ce projet, cela ne causera cependant aucun problème, pour des raisons qui seront expliquées à la section 3.5.1. B^v est donc estimée en se basant sur la valeur arbitraire choisie pour B ($B = 1$ dans notre cas).

3.3.2 Cas avec distorsion imposée

Dans le cas d'une séquence acquise avec des paramètres impliquant que $R_a \neq 1$, le but de la reconfiguration est de créer une nouvelle séquence ayant la même valeur de

R_a, afin que l'effet soit accentué de la même manière dans la nouvelle configuration de visualisation.

Dans ce cas, il n'est pas possible d'utiliser directement la méthode proposée à la section 3.3.1 pour calculer les paramètres des caméras virtuelles. Cependant, il est possible d'utiliser une autre formulation. Suivant celle-ci, on désire positionner les caméras virtuelles de façon à ce que $m_z = m_z^v$ et $m_{xy} = m_{xy}^v$, où m_{xy}^v, m_z^v sont les facteurs de magnification associés à la structure des caméras virtuelles. Si ces deux égalités sont respectées, on a

$$\frac{m_z}{m_{xy}} = \frac{m_z^v}{m_{xy}^v} \Longrightarrow R_a = R_a^v \tag{3.21}$$

où R_a^v est le ratio de magnification de la séquence virtuelle. Selon le modèle de visualisation présenté à la section 2.3,

$$m_{xy}^v(Z) = \frac{G^v f^v}{Z} \tag{3.22}$$

$$m_z^v(Z) = \frac{D'^n B^v f^v G^v}{B' Z^2}, \tag{3.23}$$

où $G^v = \frac{L^n}{k}$ est le facteur d'agrandissement entre le plan image et la nouvelle surface de projection. En imposant $m_{xy}^v = m_{xy}$, il en découle que

$$\frac{G^v f^v}{Z} = \frac{Gf}{Z} \tag{3.24}$$

$$\frac{L^n f^v}{kZ} = \frac{Lf}{kZ} \tag{3.25}$$

$$f^v = \frac{Lf}{L^n}. \tag{3.26}$$

et, en imposant $m_z^v = m_z$, on obtient

$$\frac{D'^n B^v f^v G^v}{B' Z^2} = \frac{D' B f G}{B' Z^2} \tag{3.27}$$

$$\frac{D'^n B^v f^v L^n}{k B' Z^2} = \frac{D' B f L}{k B' Z^2} \tag{3.28}$$

$$\frac{D'^n B^v}{D' B} = \frac{f L}{f^v L^n}. \tag{3.29}$$

En remplaçant f^v de l'équation (3.29) par sa valeur de l'équation (3.26), on obtient

$$\frac{D'^n B^v}{D' B} = 1 \implies B^v = \frac{D' B}{D'^n}. \tag{3.30}$$

De manière intéressante, (3.30) est le même résultat qu'à l'équation (3.20) lorsque D' et D'^n sont remplacés par A' et A'^n. Ce remplacement est possible et justifié dans le cas de l'acquisition sans distorsion (voir la section 3.3.1 pour la justification). Afin d'obtenir $\hat{f}^v$, on combine les équations (2.7) et (3.26), afin d'obtenir

$$\hat{f}^v = \frac{-2 L f x_0}{L^n k}. \tag{3.31}$$

Pour ce qui est de θ^v, on peut l'exprimer comme

$$\theta^v = \tan^{-1}\left(\frac{2A}{B^v}\right). \tag{3.32}$$

Similairement, on peut exprimer θ comme

$$\theta = \tan^{-1}\left(\frac{2A}{B}\right) \tag{3.33}$$

et donc,

$$A = \frac{B\tan(\theta)}{2}. \tag{3.34}$$

En combinant les équations (3.32) et (3.34), on déduit

$$\theta^v = \tan^{-1}\left(\frac{B\tan(\theta)}{B^v}\right). \tag{3.35}$$

En substituant B^v par l'expression trouvée à l'équation (3.30), on obtient

$$\theta^v = \tan^{-1}\left(\frac{\tan(\theta)D'^n}{D'}\right) \tag{3.36}$$

et, étant donné que θ a été estimé plus tôt, et que D'^n et D' sont connus, on peut calculer la valeur de θ^v suivant l'équation (3.36).

3.3.3 Matrices de projection des caméras virtuelles

Une fois les valeurs des paramètres des caméras virtuelles obtenues, les matrices de projection associées aux caméras virtuelles peuvent être obtenues par

$$\mathtt{M}_g^v = \mathtt{K}_{\hat{f}^v}\left[\mathtt{R}_{\frac{\pi}{2}-\theta^v}|\mathtt{t}_{-B^v/2}\right], \tag{3.37}$$

$$\mathtt{M}_d^v = \mathtt{K}_{\hat{f}^v}\left[\mathtt{R}_{\theta^v-\frac{\pi}{2}}|\mathtt{t}_{B^v/2}\right]. \tag{3.38}$$

Dans ce cas, $\mathtt{K}_{\hat{f}^v}$ est la matrice des paramètres intrinsèques basée sur $\hat{f}^v$ et construite de la même manière qu'à la section 2.2. De même, les matrices contenant la rotation et la translation des caméras sont basées sur les valeurs de θ^v et B^v.

3.4 Calcul de la structure de la scène

Une fois les paramètres des caméras virtuelles calculés, il faut retrouver la structure tridimensionnelle de la scène afin de pouvoir calculer les nouvelles images. Certaines approches préexistantes ne tentent pas de retrouver cette information. Elles font plutôt une transformation directe des images d'entrée, en se basant uniquement sur l'information géométrique de la structure des caméras. Ces approches sont souvent restreintes à certaines applications précises (par exemple, caméras parallèles avec une translation selon un seul axe). Dans le cadre de ce projet, l'approche doit être applicable à plusieurs configurations différentes de scène et de caméras.

Dans ce contexte, l'approche choisie consiste à retrouver une représentation tridimensionnelle du contenu de la scène. La version présentée construit un nuage de points 3D $N = \{\mathbf{P}_1, \mathbf{P}_2, ..., \mathbf{P}_m\}$ où $\mathbf{P}_i \in \mathbb{R}^3$. Chaque point 3D $\mathbf{P}_i$ est généré à partir d'un pixel $\mathbf{p}$ des images originales.

3.4.1 Calcul de la structure de la scène à partir des cartes de profondeur

Si une carte de profondeur est disponible à ce stade du processus (par exemple, dans certains cas de reconfiguration d'une séquence d'animation synthétique), la solution la plus simple est de l'utiliser pour calculer N. Dans ce cas, soient Z_g et Z_d les cartes de profondeur des images de gauche et de droite, respectivement, où $Z_g(\mathbf{p})$ est la profondeur dans la caméra de gauche de l'objet dont $\mathbf{p}$ représente une partie. De même, $Z_d(\mathbf{p})$ est la profondeur de ce même objet par rapport à la caméra de droite.

Connaissant Z_g et Z_d et les paramètres estimés à la section 3.2, chaque pixel de I_g et

I_d peut être « déprojeté » dans l'espace 3D pour former des nuages intermédiaires N_g et N_d. La procédure de « déprojection » pour un pixel $\mathbf{p}_g$ de la caméra de gauche est exprimée par

$$\mathbf{P}'_g = \mathtt{R}_t^{-1}\mathtt{Z}_g\mathtt{K}_g^{-1}\mathbf{p}_g \tag{3.39}$$

où $\mathbf{p}_g$ est le pixel de la caméra de gauche en coordonnées homogènes, $\mathbf{P}'_g \in N_g$ le point 3D associé, et $\mathtt{K}_g$ est la matrice des paramètres intrinsèques de C_g. Ici, $\mathtt{Z}_g$ est une matrice de « déprojection » donnée par

$$\mathtt{Z}_g = \begin{bmatrix} Z_g(\mathbf{p}) & 0 & 0 \\ 0 & Z_g(\mathbf{p}) & 0 \\ 0 & 0 & Z_g(\mathbf{p}) \\ 0 & 0 & 1 \end{bmatrix}, \tag{3.40}$$

et $\mathtt{R}_t$ est une matrice 4x4 représentant la combinaison d'une rotation et d'une translation présentée sous la forme

$$\mathtt{R}_t = \begin{bmatrix} \mathtt{R} & -\mathtt{R}\tilde{\mathbf{C}} \\ 0 & 1 \end{bmatrix}. \tag{3.41}$$

La sous-matrice 3x3 supérieure gauche de $\mathtt{R}_t$ est la matrice $\mathtt{R}$ associée à C_g, le vecteur 3x1 supérieur droit correspond à $-\mathtt{R}\tilde{\mathbf{C}}$ et le vecteur 1x3 inférieur gauche est constitué de zéros. Tel que précisé à la section 2.2, cette matrice représente la position et l'orientation de C_g.

Le processus est similaire pour les pixels $\mathbf{p}_d$ de I_d, qui génère le nuage N_d. Une fois les deux nuages obtenus, ils sont unis, sans filtrage ou simplification, afin de créer un seul nuage N.

3.4.2 Calcul de la structure de la scène à partir de disparités

Lorsqu'une carte de profondeur n'est pas directement disponible, il faut l'estimer à l'aide d'autres techniques. La technique choisie consiste à rectifier les images originales en utilisant les paramètres estimés précédemment et la technique de rectification de [23]. Une fois la paire d'images rectifiée, un algorithme de stéréovision est utilisé pour calculer une carte de disparités dense. Évidemment, comme de nouveaux algorithmes de stéréovision sont publiés à chaque mois, il est possible de choisir celui qui est le meilleur au moment du développement du programme de reconfiguration.

Dans la solution implémentée pour ce projet, l'algorithme de stéréovision de Zitnick *et al* [75] a été utilisé. Comme expliqué à la section 1.2.2, cet algorithme est basé sur une segmentation des images d'entrée en petites régions de couleurs similaires, qui sont ensuite mises en correspondance selon leur forme, leur couleur et leur superficie. Une fois que cette mise en correspondance est faite, un processus d'affinage inter- et intrarégional est effectué, afin d'avoir une variation réaliste de la disparité, tout en conservant les changements brusques entre les régions très différentes.

Une fois la carte de disparités calculée, les points 3D $\mathbf{P}' \in N$ sont calculés en utilisant une technique classique de triangulation basée sur les valeurs de disparité et les paramètres extrinsèques des caméras (voir [23, 58]). Il est à noter que l'utilisation des disparités dans le calcul de N peut entraîner un artéfact particulier lors de la reprojection. Cet artéfact sera expliqué à la section 3.6.3. Malgré la présence de cet artéfact, l'utilisation des cartes de disparités est souvent la seule solution possible, car les cartes de profondeur n'existent pratiquement jamais pour les séquences réelles.

3.5 Rendu des nouvelles images

Une fois le nuage N de points 3D obtenu, la création des nouvelles images I_g^v et I_d^v peut commencer. Il faut tout d'abord étudier l'effet de l'utilisation d'une valeur arbitraire pour B^v, ce qui est fait à la section 3.5.1. Par la suite, deux méthodes de reprojection se présentent. Au premier abord, la solution serait d'effectuer le processus classique de projection de tous les points $\mathbf{P}_i \in N$ sur les deux nouvelles caméras C_g^v et C_d^v, afin d'obtenir I_g^v et I_d^v. C'est ce qu'on appelle une approche par reprojection directe. Comme il sera expliqué à la section 3.5.2, il y a certains inconvénients à utiliser cette approche. Une approche moins intuitive mais produisant de meilleurs résultats consiste à effectuer une reprojection inverse, méthode présentée à la section 3.5.3.

3.5.1 Mise à l'échelle de N

Tel que mentionné à la section 3.2.1, il est impossible de retrouver avec précision la valeur originale de la distance intercaméra B si celle-ci n'est pas connue *a priori*. Si la valeur originale de B était connue, alors

$$B^v = \frac{BD'}{D'^n}. \tag{3.42}$$

En désignant par B_{vrai} la valeur originale de B, par $B_{arbitraire}$ la valeur donnée arbitrairement à B lors du calcul du nuage N, par $B^v_{parfait}$ la valeur de B^v obtenue à partir de B_{vrai}, et par $B^v_{echelle}$ la valeur de B^v obtenue à partir de $B_{arbitraire}$, il appert que

$$\frac{B_{vrai}}{B^v_{parfait}} = \frac{D'^n}{D'} = \frac{B_{arbitraire}}{B^v_{echelle}} \tag{3.43}$$

et donc, que ces deux ratios ont une valeur identique. Le nuage N calculé en se basant sur $B_{arbitraire}$ est donc mis à l'échelle par rapport au nuage qui aurait été obtenu en utilisant B_{vrai}. Cependant, cela ne causera pas de problème lors du rendu. En effet, puisque $\frac{B_{vrai}}{B^v_{parfait}} = \frac{B_{arbitraire}}{B^v_{echelle}}$, l'opération de reprojection est basée sur une valeur de B^v dont le ratio à B est toujours le même. L'effet de mise à l'échelle est donc annulé lors de la reprojection.

3.5.2 Reprojection directe

Connaissant les coordonnées 3D des points $\mathbf{P}_i \in N$ et $\mathtt{M}^v_g$, $\mathtt{M}^v_d$, il est possible de simuler le processus naturel de projection d'une caméra. Dans ce cas, pour chaque point 3D $\mathbf{P}_i \in N$, on obtient les positions en pixels suivantes :

$$\mathbf{p}^v_{ig} = \mathtt{M}^v_g \mathbf{P}_i \tag{3.44}$$

$$\mathbf{p}^v_{id} = \mathtt{M}^v_d \mathbf{P}_i \tag{3.45}$$

où $\mathbf{p}^v_{ig}$, $\mathbf{p}^v_{id}$ sont respectivement les coordonnées dans I^v_g, I^v_d de la projection de $\mathbf{P}_i$. En projetant tous les points du nuage, les images virtuelles sont remplies progressivement. Il est possible que plusieurs points se projettent au même pixel d'une des images virtuelles. Dans ce cas, le pixel provenant du point le plus près de la caméra est conservé pour l'image finale.

La reprojection directe est une technique rapide et simple. Cependant, certains artéfacts visuels peuvent apparaître. Un exemple de résultat de reprojection directe

basée sur les disparités se trouve à la figure 3.2. Tout d'abord, comme certains points risquent de se projeter sur le même pixel, cela implique que d'autres pixels ne seront pas remplis (voir la figure 3.3a). De même, certains types de changements de configuration impliquent que la longueur focale des caméras virtuelles sera plus grande que celle des caméras originales. Lors de la reprojection directe, cela induit des artéfacts de lignes non-remplies (voir la figure 3.3b). Contrairement aux points se projetant sur le même pixel, ici il n'y a tout simplement pas de point qui pourrait se projeter sur les pixels de ces lignes. Cela est causé par l'agrandissement de la focale, ce qui implique un « agrandissement » des images sources. Or, étant donnée la nature discrète des images, cet agrandissement par reprojection se ramène à un agrandissement sans interpolation des pixels vides. Ce sont ces pixels vides qui créent les lignes non-remplies.

De plus, étant donné les changements de certains paramètres des caméras induits par le processus de reconfiguration, il se peut qu'une partie des images virtuelles (normalement, un des coins) ne puisse être remplie, car elle n'était pas perçue par les caméras originales. Dans les faits, cet artéfact découle du même phénomène que les artéfacts de désoccultation se produisant lorsqu'une partie de la scène originalement invisible devient visible (voir les figures 3.3c et 3.3d, la section 1.2.2 pour une description du phénomène, et la section 3.6.2 pour des techniques de réduction de ces artéfacts).

3.5.3 Reprojection inverse

Ayant constaté les problèmes de la méthode de reprojection directe, l'alternative d'une méthode par reprojection inverse est proposée afin d'éviter certains de ces artéfacts. La méthode proposée ici est inspirée par celle de Morvan [40], que nous avons adaptée

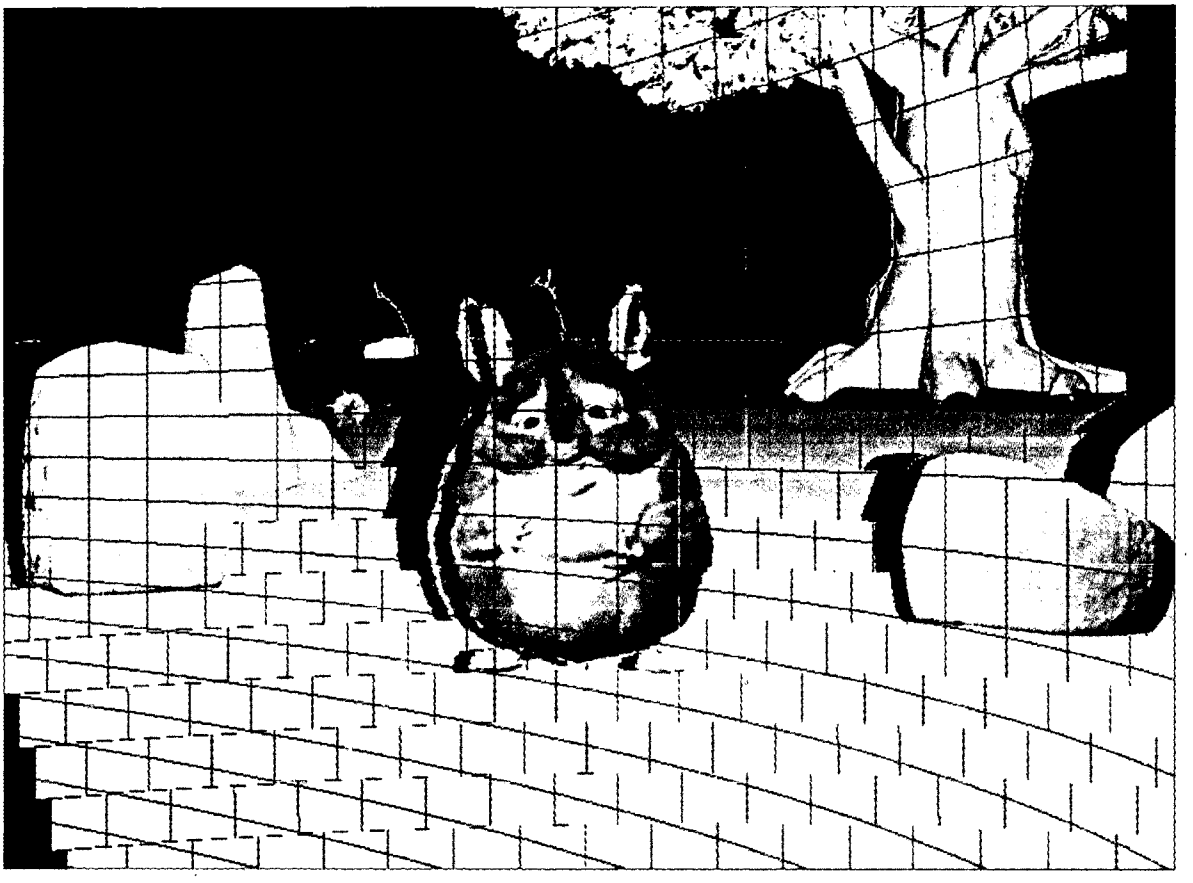
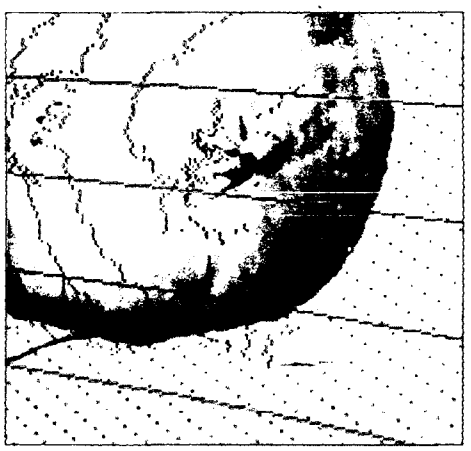
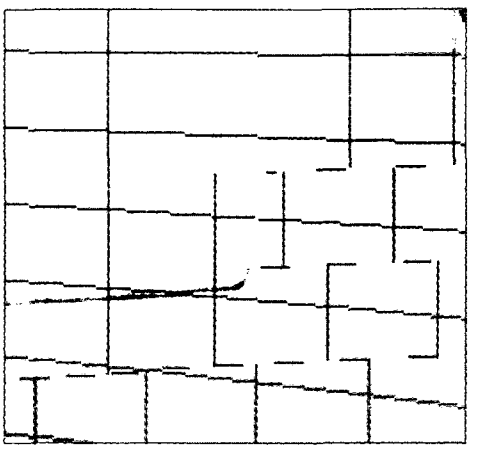


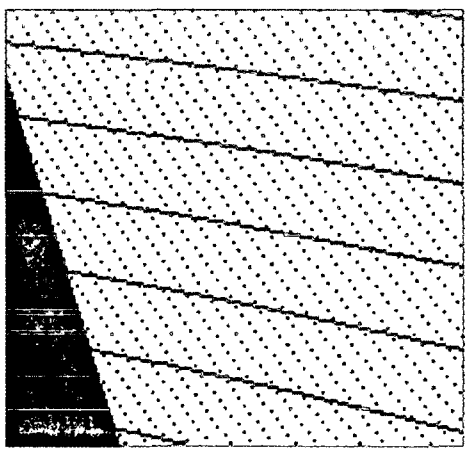
Figure 3.2 – Exemple de résultat de la reprojection directe basée sur les disparités. Les artéfacts de lignes non-remplies, de désoccultation et de fissures dues aux disparités sont présents dans l'image.



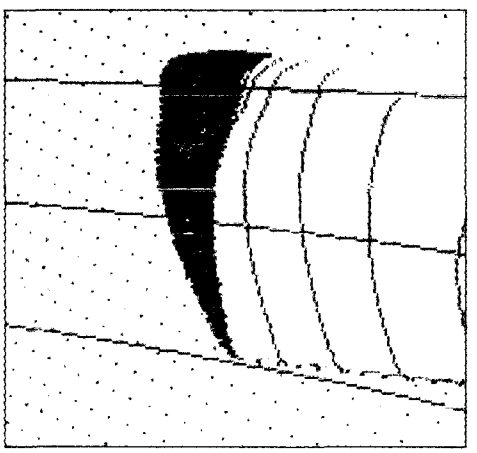
(a) Artéfact de reprojection sur le même pixel



(b) Artéfact de changement de longueur focale



(c) Artéfact du coin de l'image



(d) Artéfact de désoccultation

Figure 3.3 – Artéfacts liés à la reprojection directe : (a) trous causés par plusieurs points se projettant sur le même pixel. Certains points se sont projetés sur le même pixels, laissant d'autres pixels vides. (b) lignes vides causées par le changement de longueur focale lors de la reprojection. Le changement de focale a provoqué des artéfacts de lignes vides. (c) absence d'information dans un des coins de l'image causée par le déplacement des caméras et le changement de leur orientation. (d) information manquante pour la partie de la scène qui était originalement derrière la pierre, et qui est maintenant perçue par les caméras.

aux contraintes du problèmes courant. La méthode comprend cinq étapes que nous avons schématisées à la figure 3.4.

La première étape consiste en une reprojection directe des points de N vers les caméras virtuelles. Cependant, au lieu de créer les images I_g^v et I_d^v, deux images de profondeur Z_g^v et Z_d^v sont créées. Ces images sont des images de profondeur transformées, en ce sens qu'elles contiennent, pour chaque pixel, la distance entre le point 3D générant ce pixel et la caméra originale ayant perçu ce point 3D. Formellement,

$$Z_g^v\left(\mathbf{p}_{ig}^v\right) = |\mathbf{P}_i - Pos(C_g)| \tag{3.46}$$

$$Z_d^v\left(\mathbf{p}_{id}^v\right) = |\mathbf{P}_i - Pos(C_d)| \tag{3.47}$$

où $\mathbf{p}_{ig}^v$, $\mathbf{p}_{id}^v$ sont des pixels de Z_g^v et Z_d^v, $\mathbf{P}_i \in N$ est le point 3D ayant été projeté sur le pixel en question, et $Pos(C_i)$ représente la position 3D de la caméra C_i.

Tout comme les images générées par reprojection directe, les images de profondeur transformées peuvent avoir des trous et certains artéfacts sur les bords des objets. Morvan propose d'effectuer une opération de dilation suivie d'une opération d'érosion afin de combler les trous uniques. Par la suite, pour chaque pixel $\mathbf{p}_g^v$ de Z_g^v, un point 3D $\mathbf{P}_g^v$ est calculé en utilisant

$$\mathbf{P}_g^v = C_g^v + \lambda \mathbf{R}_g^{-v} \mathbf{K}_g^{-v} \mathbf{p}_g^v \tag{3.48}$$

où C_g^v est la position du centre de la caméra virtuelle de gauche,

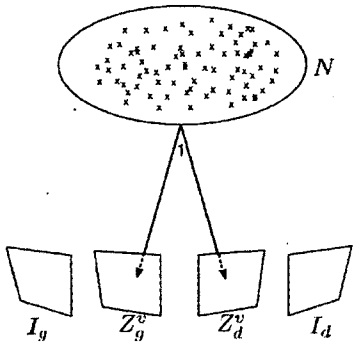
$$\lambda = \frac{Z_g^v\left(\mathbf{p}_g^v\right) - C_{gz}^v}{r_3} \tag{3.49}$$

et

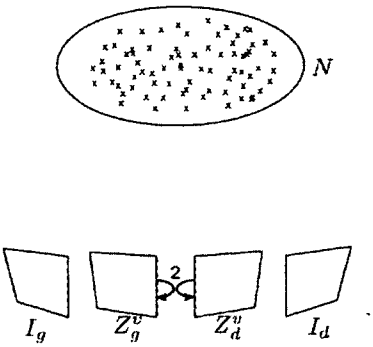
$$\begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix} = \mathtt{R}_g^{-v}\mathtt{K}_g^{-v}\mathbf{p}_g^v. \tag{3.50}$$

Finalement, $\mathbf{P}_g^v$ est projeté sur les caméras originales C_g et C_d. Pour chacune des caméras, la couleur du pixel est interpolée bilinéairement à partir des pixels originaux. La couleur du pixel final est choisie en fonction de la consistance des deux couleurs interpolées précédemment. Si la couleur est consistante, elle est utilisée pour I_g^v. Si elle n'est pas consistante, la couleur correspondant au point le plus près de C_g^v est choisie. I_d^v est créée de façon similaire.

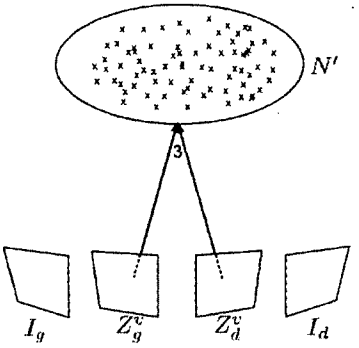
En utilisant la reprojection inverse, les problèmes de trous uniques et de lignes non-remplies sont éliminés (voir la figure 3.5). En effet, puisque le processus tente de remplir chaque pixel des images de destination en retournant dans les images originales pour trouver la couleur associées, il ne peut y avoir de problème de multiples points reprojetés au même endroit. Par contre, comme Z_g^v et Z_d^v sont créées à partir de N, il peut tout de même y avoir des artéfacts de désoccultation, puisqu'il n'y a aucune garantie que tous les pixels de Z_g^v et Z_d^v auront une valeur. Cela s'explique par le fait que, pour les surfaces occultées dans les images originales, il n'y a pas de point généré pour N. Il faut donc s'en remettre aux techniques de la section 3.6.2 pour tenter de corriger ces artéfacts.



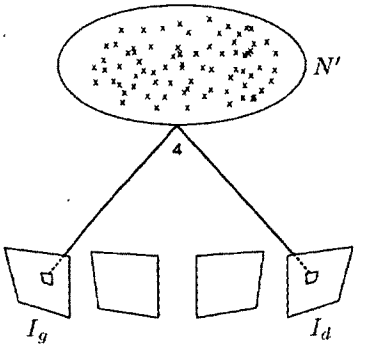
(a) Création des images de profondeur transformées



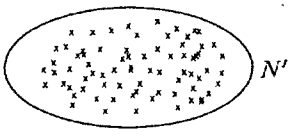
(b) Dilatation et érosion des images de profondeur transformées

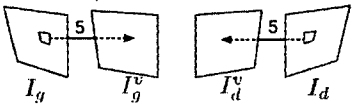


(c) Création du nuage N'



(d) Projection sur les images originales





(e) Interpolation bilinéaire de la couleur de destination, et assignation à l'image virtuelle

Figure 3.4 – Illustration des étapes de la reprojection inverse.

Figure 3.5 – Exemple de résultat de la reprojection inverse basée sur les disparités. Les artéfacts de trous uniques et de lignes non remplies ont été éliminés.

3.6 Correction des artéfacts de reprojection

Tel que mentionné aux sections précédentes, il est probable que le processus de reprojection induise des artéfacts visuels dans les nouvelles images. Comme le but de ce projet est de créer de nouvelles images ayant la plus grande qualité possible et que les artéfacts dans les images générées peuvent causer des désagréments lors du visionnement de la séquence reconfigurée, il faut tenter d'éliminer ces artéfacts. Des techniques de base permettent d'éliminer les artéfacts de trous uniques et de lignes non-remplies, et une technique plus poussée d'emplissage d'image (« image inpainting ») réduit les artéfacts de désoccultation. Certains artéfacts de fissure peuvent aussi se produire dans certains cas précis.

3.6.1 Trous uniques et lignes non-remplies

Si l'utilisateur du cadre de reconfiguration décide d'utiliser la reprojection directe, des artéfacts de trous uniques et de lignes non-remplies apparaîtront (voir les figures 3.3a et 3.3b). Dans ce cas, il est possible de les éliminer en utilisant un filtre médian qui considère les huit pixels voisins du pixel à remplir. Cette opération est pratiquement instantanée, et la différence entre image traité avec ce filtre et la vérité-terrain est imperceptible pour un observateur humain.

3.6.2 Artéfacts de désoccultation

Dans le cas des artéfacts causés par la désoccultation de certaines parties de la scène, les régions à remplir sont souvent plus importantes qu'un unique pixel ou une unique ligne (voir les figures 3.3c et 3.3d). Il faut donc avoir recours à des techniques plus

poussées. Tel qu'expliqué à la section 1.2.4, les techniques d'emplissage de régions d'images peuvent être appliquées à plusieurs domaines. Dans le cadre de ce projet, il a été décidé d'adapter au contexte de rendu stéréoscopique la technique proposée par Criminisi *et al.* [12]. Cette technique a été choisie comme base parce qu'elle utilise l'information de l'image se situant autour de la région à emplir afin de trouver le contenu à utiliser pour l'emplissage. Dans le contexte de la reconfiguration stéréoscopique, l'algorithme peut être facilement adapté afin de n'utiliser que de l'information ne provenant pas de l'objet ayant créé l'occultation.

Le but de la méthode originale est de retirer un objet d'une image, en spécifiant une région d'intérêt à enlever. Par la suite, l'algorithme utilise la couleur et l'information de texture provenant du contour de la région pour la remplir, en donnant la priorité aux pixels vers lesquels les contours pénétrant dans la région se dirigent. Une contrainte est ajoutée à cet algorithme pour pouvoir l'utiliser dans la problématique courante : il est interdit d'utiliser l'information provenant de la région causant l'occultation. Ainsi, la région à remplir le sera en n'utilisant que de l'information provenant de l'arrière-plan et des objets se situant à une plus grande distance que l'objet occultant.

Concrètement, la contrainte a été mise en place en identifiant les régions vides provenant de la désoccultation (les régions larges de moins de deux pixels sont remplies à l'aide de la technique de la section 3.6.1). Une fois ces régions identifiées, le contour de la région est parcouru afin d'identifier les pixels provenant de la région occultante. Ces pixels sont déterminés comme ceux ayant une profondeur dans l'intervalle $[Z_{min}, Z_{min} + \tau]$, où Z_{min} est la plus petite profondeur rencontrée, et τ est un seuil ajusté selon la séquence. Finalement, lorsque tous les pixels occluants sont identifiés, l'algorithme de Criminisi est lancé, en l'empêchant d'utiliser l'information des pixels

identifiés.

3.6.3 Artéfacts de fissures

Dans le cas des images obtenues par reprojection d'un nuage N obtenu à partir de disparités, certaines surfaces peuvent contenir des fissurès. Un exemple de cet artéfact est présenté à la figure 3.6. De telles fissures sont dues à la discrétisation des valeurs de disparités. Étant donné que ces valeurs sont discrètes, elles peuvent soudainement changer de valeur au sein d'un même objet si celui-ci n'est pas constitué d'une seule surface fronto-parallèle à la caméra. Lors du calcul de N, la profondeur est déduite de la valeur de disparité. Ceci implique que deux points voisins sur la surface originale, l'un ayant une disparité d et l'autre ayant une disparité $d + 1$, auront une différence de profondeur marquée dans N. Cette différence entrainera une fissure lors de la reprojection, puisqu'il manquera des points pour représenter toute la surface de façon continue dans N.

Une solution partielle à ce problème serait d'utiliser des cartes de disparités ayant des valeurs $\in \mathbb{R}$. Cependant, il existe très peu d'algorithmes de stéréovision permettant d'obtenir de telles valeurs, car le problème d'optimisation serait rarement soluble. Une solution intermédiaire consiste donc à utiliser un algorithme de stéréovision pouvant fournir des valeurs de disparités dont le pas serait plus petit qu'un pas unitaire. Les résultats pourraient par exemple utiliser un pas de 0.1. Cela ne règle pas entièrement le problème de fissures, mais l'amoindrit. Les fissures restantes sont remplies par l'algorithme de remplissage de trous présenté à la section 3.6.2.

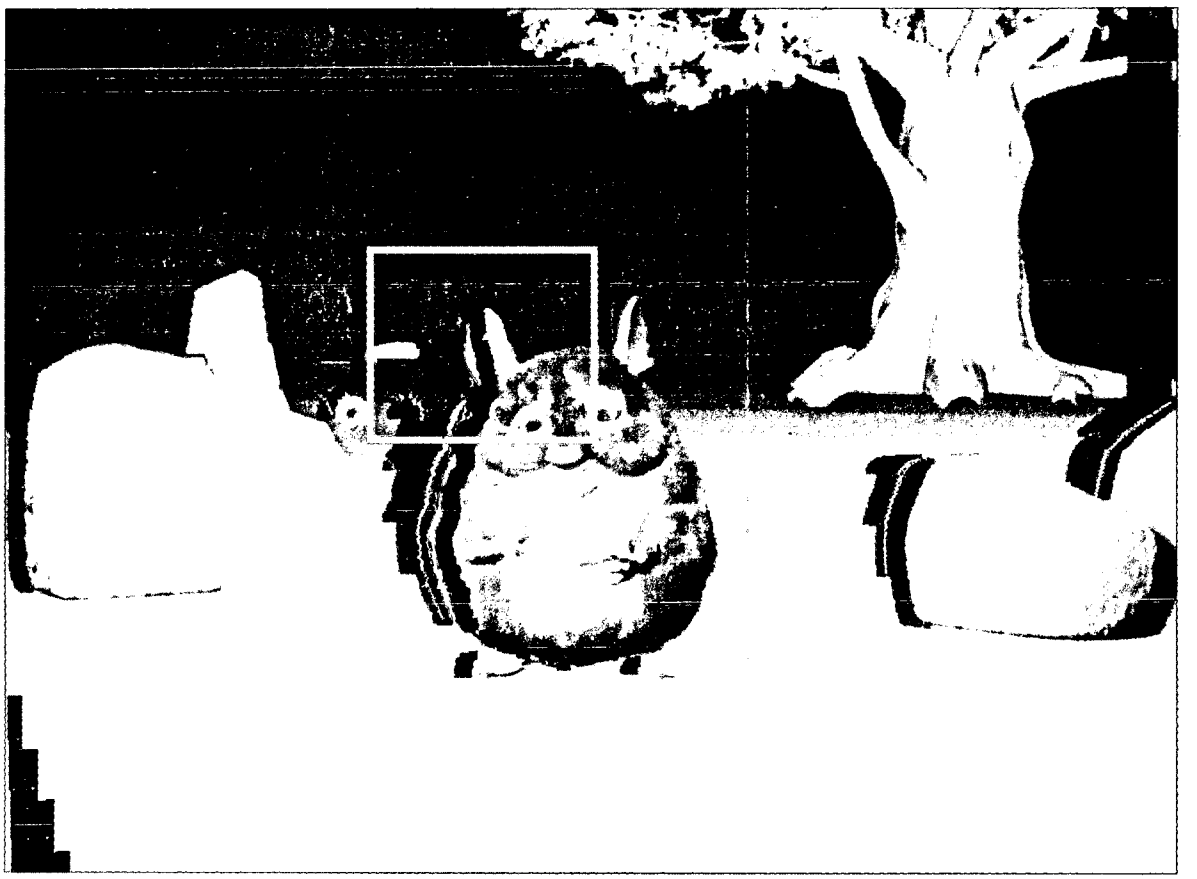


Figure 3.6 – Exemple de l'artéfact de fissure pouvant être généré lors de la reprojection d'un nuage N issu de disparités. La zone entourée de rouge montre l'oreille du chinchilla, qui a été divisée en deux parties, alors qu'elle devrait être unie. Cet artéfact apparaît parce que la valeur (discrète) de disparité fait un bond unitaire au milieu de l'oreille.

3.6.4 Résumé de l'algorithme

Avant de conclure ce chapitre, un résumé de l'algorithme est présenté à l'algorithme 1. Celui-ci permet d'avoir une vue d'ensemble du fonctionnement de la méthode. Ce résumé permet aussi d'avoir une référence rapide aux sections et équations utilisées par chacune des étapes.

Algorithme 1 Reconfiguration stéréoscopique

ENTRÉES : I_g, I_d : images originales de gauche et de droite
ENTRÉES : B', D', L : paramètres de la configuration originale de visionnement
ENTRÉES : B', D'^n, L^n : paramètres de la nouvelle configuration de visionnement

1. Déduire les paramètres originaux d'acquisition $B, \hat{f}, \theta$
si aucune distorsion ($R_a = 1$) **alors**
  Fixer $B = 1$ et utiliser les équations (3.1) et (3.2)
sinon
  Fixer $B = 1$ et utiliser les équations (3.6) et (3.11)
finsi
2. Calculer les paramètres virtuels d'acquisition $B^v, \hat{f}^v, \theta^v$
si aucune distorsion ($R_a = 1$) **alors**
  Utiliser les équations (3.15), (3.16) et (3.20)
sinon
  Utiliser les équations (3.30), (3.31) et (3.36)
finsi
3. Calculer la structure de la scène
si carte de profondeur disponible **alors**
  Calculer N selon l'équation (3.39)
sinon
  Calculer les cartes de disparités avec un algorithme de stéréovision
  Calculer N selon la technique de la section 3.4.2
finsi
4. Reprojeter N sur C_g^v, C_d^v pour obtenir les images virtuelles I_g^v, I_d^v
si par reprojection directe **alors**
  Calculer I_g^v, I_d^v selon l'équation (3.44)
sinon
  Calculer I_g^v, I_d^v selon la technique de la section 3.5.3
finsi
5. Corriger les artéfacts de I_g^v, I_d^v avec l'algorithme de la section 3.6

CHAPITRE 4

Expérimentation et résultats

Le cadre de traitement et les modèles mathématiques développés dans ce mémoire doivent être testés afin d'en assurer la validité et de vérifier la qualité des résultats produits. Une problématique majeure du cinéma stéréoscopique est qu'il est difficile de valider l'effet 3D. En effet, puisque l'effet 3D perçu n'est qu'une illusion de notre système visuel, la quantification de la qualité de l'effet 3D est difficile. De plus, à notre connaissance, aucune séquence stéréoscopique ayant été acquise deux fois en considérant deux configurations de visualisation différentes n'est accessible publiquement. Si de telles séquences existent, elles ne sont malheureusement pas facilement accessibles pour la recherche.

Afin d'avoir des séquences permettant une validation du cadre de traitement et des modèles mathématiques proposés, des séquences synthétiques ont été développées. La validation basée sur ces séquences est présentée à la section 4.1, tandis qu'une validation sur certaines séquences réelles est présentée à la section 4.2. Finalement, une validation de l'algorithme d'emplissage de trous est présentée à la section 4.3.

4.1 Validation sur des séquences synthétiques

Afin d'avoir des séquences stéréoscopiques permettant de contrôler l'effet 3D produit, et permettant ainsi de valider la qualité de la reconfiguration, deux séquences synthétiques ont été développées. Comme elles ont été développées à l'interne, un contrôle total a été exercé sur les paramètres utilisés lors du rendu. De plus, il est possible de générer exactement la même séquence de mouvements et d'actions des personnages lors de plusieurs rendus successifs, ce qui serait ardu avec de vrais acteurs. Le fait d'utiliser des séquences synthétiques permet aussi de calculer la vérité-terrain avec laquelle les résultats de la reconfiguration peuvent être comparés algorithmiquement.

4.1.1 Description des séquences

Deux séquences synthétiques ont été calculées à partir du logiciel libre Blender [2]. Ce logiciel de création de contenu 3D possède une interface de programmation de l'application (« API ») permettant de contrôler le rendu de séquence. Cette interface a permis de générer les séquences selon les points de vue associés à chacune des deux caméras nécessaires. Les informations sur la configuration de visualisation sont fournies au script, et celui-ci calcule les paramètres à utiliser pour le rendu. Un autre avantage d'utiliser ce logiciel est qu'il permet de récupérer les cartes de profondeur suite au rendu. Ceci permettra de tester la technique de calcul de la structure de la scène à partir des valeurs de profondeur.

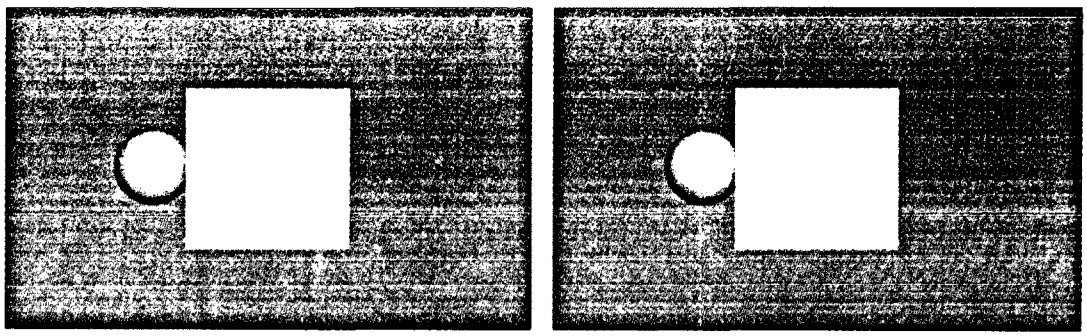
La première séquence test, désignée ci-après par *Cube et sphères*, est une séquence simple, composée de deux sphères et d'un cube gris sur un arrière-plan bleu (voir figures 4.1a et 4.1b). Les trois objets sont animés à travers la séquence, et la caméra se

déplace autour de la scène, afin de permettre d'évaluer l'effet de changement de profondeur. La séquence est constituée de 1050 images. Cette séquence a principalement été utilisée pour les tests de base et l'élaboration de la méthode.

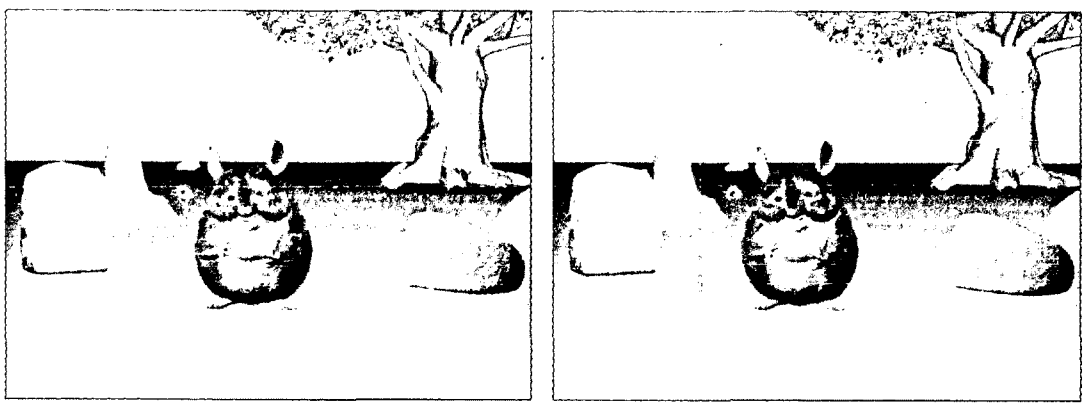
La deuxième séquence synthétique est une séquence plus complexe et « réaliste », en ce sens qu'elle contient des objets complexes et est visuellement plus chargée (voir figures 4.1c et 4.1c). On y trouve entre autres un arbre, un chinchilla et un bonhomme en pain d'épices. Elle sera nommée *Chinchilla*. Cette séquence a été élaborée à partir des modèles provenant du film libre « Big Buck Bunny » [1]. Le film étant libre, les modèles et les scènes peuvent être réutilisés sans contrainte. La séquence a ainsi été bâtie en utilisant certains modèles et en les animant. De plus, un modèle de bonhomme en pain d'épices a été ajouté afin d'intégrer plus de mouvement dans la scène. La séquence complète est constituée de 780 images.

4.1.2 Protocole expérimental

Les deux séquences ont été rendues selon deux configurations de visionnement. La première, nommée ci-après *Cinéma*, représente la configuration de la salle de cinéma de la compagnie Sensio, compagnie avec laquelle le projet a été piloté. Cette configuration est considérée comme la configuration source, c'est-à-dire celle à partir de laquelle les images seront reconfigurées. Dans cette configuration, la distance spectateur-écran de projection $A' = 16,75m$ et la largeur de l'écran $L = 9,75m$. La deuxième configuration est basée sur un moniteur 3D Samsung® 2233RZ. Cette configuration, considérée comme la configuration destination, est désignée comme *Samsung*. Dans ce cas, $A' = 0,8382m$, et $L = 0,475m$. Dans les deux cas, et pour les deux séquences, la distance entre la ligne de base et le plan de séparation de l'effet 3D $A = 7.0m$ et la



(a) *Cube et sphères* - image originale de gauche (b) *Cube et sphères* - image originale de droite



(c) *Chinchilla* - image originale de gauche (d) *Chinchilla* - image originale de droite

Figure 4.1 – Images originales des séquences synthétiques dans la configuration *Cinéma* : (a) *Cube et sphères* - image de gauche, (b) *Cube et sphères* - image de droite, (c) *Chinchilla* - image de gauche, (d) *Chinchilla* - image de droite.

distance interoculaire $B' = 0,065m$.

L'expérimentation sur les séquences synthétiques a été effectuée en plusieurs étapes. Tout d'abord, les deux séquences ont été rendues pour les deux configurations de visionnement. En même temps que le rendu s'effectuait, la carte de profondeur associée à chaque image a été sauvegardée. Une fois le rendu terminé, les deux séquences ont été reconfigurées de la configuration *Cinéma* à la configuration *Samsung* à l'aide de la reprojection directe basée sur les valeurs de profondeur obtenues lors du rendu. Elles ont aussi été reconfigurées avec la reprojection inverse basée sur la profondeur.

Dans le cas des séquences synthétiques, les cartes de disparités ont été calculées à partir des nuages N obtenus à l'étape décrite à la section 3.4. En effet, en reprojetant les points sur l'autre caméra originale (et non pas virtuelle), et en vérifiant la similarité de la couleur, on peut déduire la disparité en soustrayant les coordonnées du pixel obtenu par reprojection de celles du pixel source. Cela permettait de tester la reprojection basée sur les disparités sans introduire d'erreurs supplémentaires dues aux erreurs de calcul des cartes de disparités.

Une fois ces cartes de disparités obtenues, les reprojections par disparité directe et inverse de la configuration *Cinéma* à la configuration *Samsung* ont été calculées pour les deux séquences. Finalement, les techniques de correction d'artéfacts ont été appliquées aux résultats des quatre reprojections pour les deux séquences. Les différents résultats sont présentés aux figures 4.2, 4.3, 4.4 et 4.5.

4.1.3 Évaluation quantitative des résultats

Les vérités-terrain des séquences étant connues, il est possible d'effectuer une analyse de la similarité entre les images de la vérité-terrain et les images obtenues par

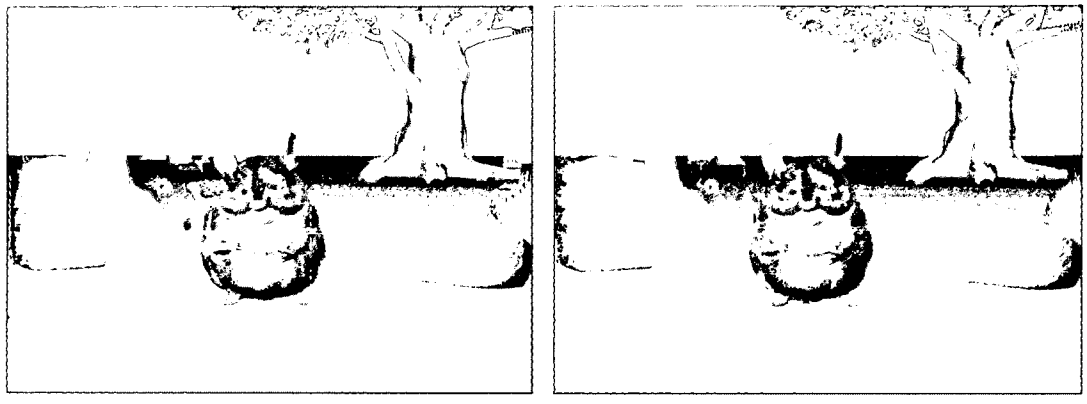
(a) Reprojection directe par les disparités

(b) Reprojection inverse par les disparités

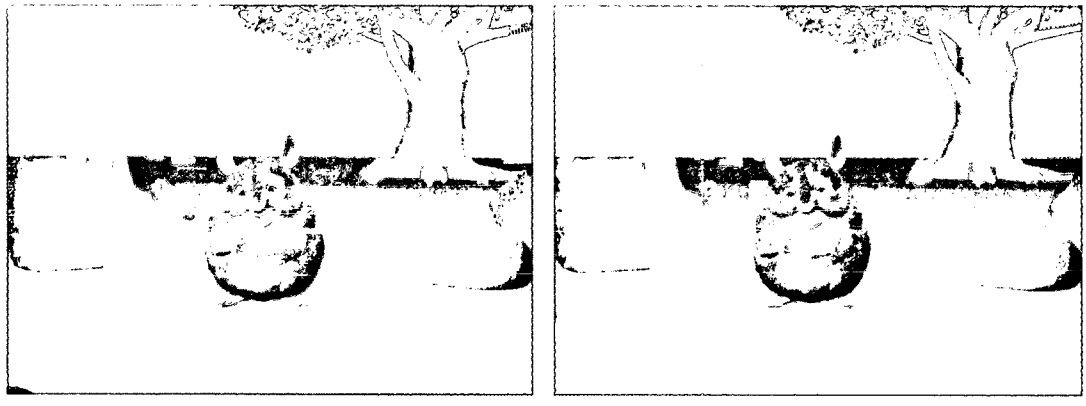
(c) Reprojection directe par la profondeur

(d) Reprojection inverse par la profondeur

Figure 4.2 – Images de la séquence *Chinchilla* reconfigurées pour *Samsung*, sans post-traitement.

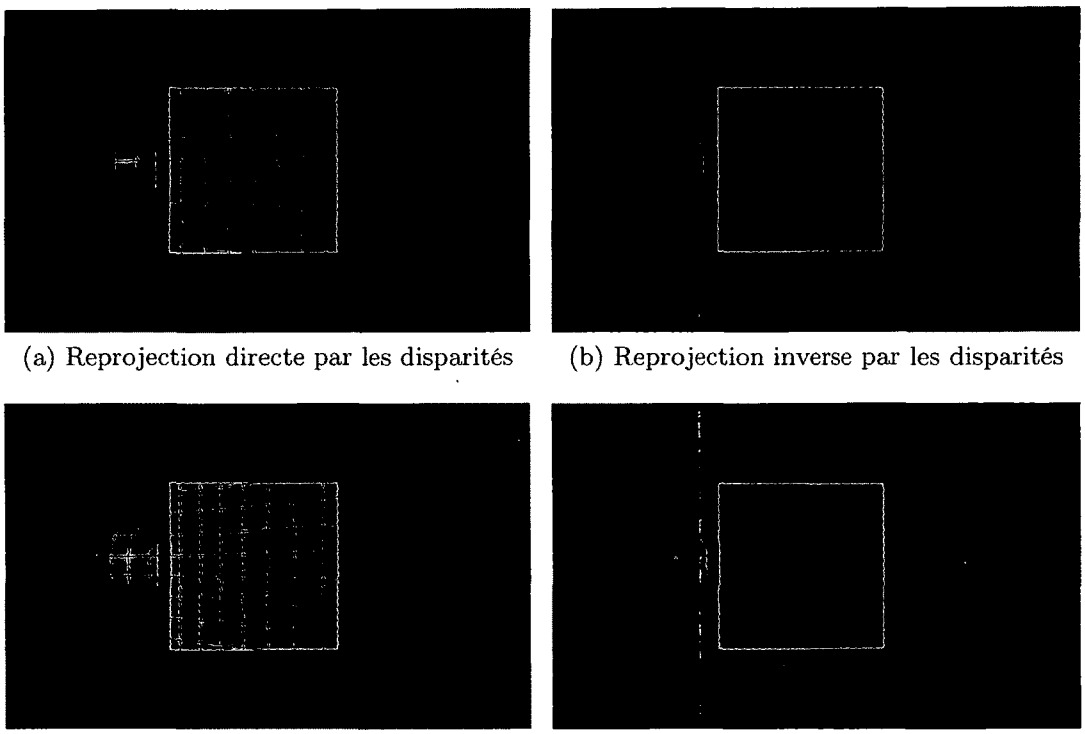


(a) Reprojection directe par les disparités (b) Reprojection inverse par les disparités



(c) Reprojection directe par la profondeur (d) Reprojection inverse par la profondeur

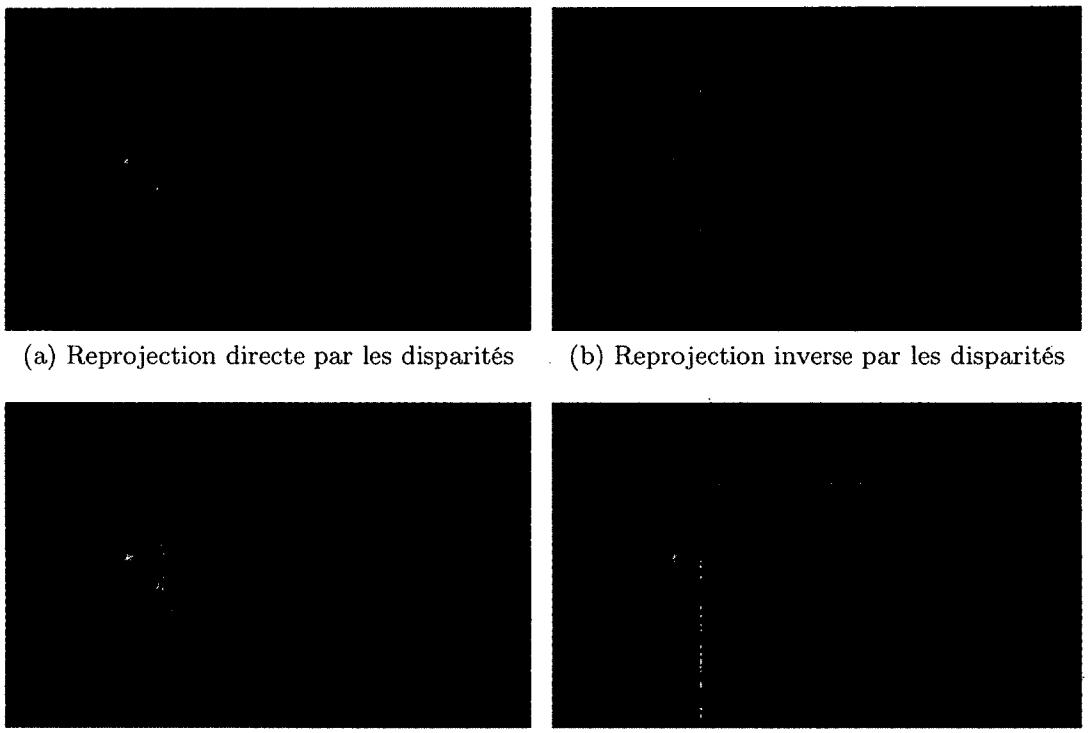
Figure 4.3 – Images de la séquence *Chinchilla* reconfigurées pour *Samsung* dont les artéfacts ont été traités.



(a) Reprojection directe par les disparités (b) Reprojection inverse par les disparités

(c) Reprojection directe par la profondeur (d) Reprojection inverse par la profondeur

Figure 4.4 – Images de la séquence *Cube et sphères* reconfigurées pour *Samsung*, sans post-traitement.



(a) Reprojection directe par les disparités (b) Reprojection inverse par les disparités

(c) Reprojection directe par la profondeur (d) Reprojection inverse par la profondeur

Figure 4.5 – Images de la séquence *Cube et sphères* reconfigurées pour *Samsung* dont les artéfacts ont été traités.

reconfiguration. Pour ce faire, la mesure de ratio signal-bruit maximum (« PSNR ») a tout d'abord été considérée, de par sa simplicité de calcul et d'interprétation. Cette mesure de similarité s'est avérée limitée puisqu'elle est très sensible aux changements spatiaux entre deux images. Dans le cas des images obtenues par reconfiguration, il est fréquent que l'image résultante soit très similaire à la vérité-terrain, tout en étant décalée d'un pixel à cause de l'arrondi des coordonnées lors de la reprojection. La mesure du PSNR étant sensible aux changements spatiaux, les valeurs obtenues n'étaient pas intéressantes et représentaient très peu la similarité des images.

Suite à cette constatation, la mesure de similarité MS-SSIM (« Multi-scale Structural Similarity », voir [59]) a été utilisée. Cette mesure de similarité calcule la similarité structurelle d'images en se fiant sur différentes caractéristiques de l'image, et est de ce fait moins sensible au décalage d'un pixel qui affecte le PSNR. Les valeurs possibles pour la mesure se situent dans l'intervalle $[0, 1]$. Une valeur de similarité de 1 indique que les deux images sont parfaitement identiques, et la similarité décroît lorsque la valeur tend vers 0.

Cette mesure a été calculée sur la séquence *Chinchilla* avec différentes combinaisons de paramètres. Le tableau 4.1a présente la valeur moyenne de la mesure MS-SSIM sur les images reconfigurées pour la configuration *Samsung*. Pour ce tableau, la mesure a été calculée uniquement sur les pixels ayant effectivement été reprojetés, c'est-à-dire que les parties de l'image n'ayant pas reçu une valeur (à cause de la désoccultation ou d'un autre type d'artéfact) n'ont pas été considérées. Il est possible de voir que les images obtenues par la technique basée sur les valeurs de profondeur ont des valeurs moyennes de MS-SSIM légèrement plus élevées que celles provenant de la technique basée sur les disparités.

Le tableau 4.1b présente la valeur moyenne de la mesure MS-SSIM sur les images

reconfigurées et dont les artéfacts ont été traités avec les techniques proposées à la section 3.6. Comme dans le cas du tableau 4.1a, le lecteur peut voir que la reprojection basée sur les valeurs de profondeur donne des mesures de similarité légèrement meilleures que celles obtenues par les disparités. Il faut aussi constater que les valeurs sont toutes inférieures aux valeurs correspondantes lorsque seuls les pixels reprojetés étaient considérés. Cela démontre que l'emplissage de trous et la correction des artéfacts n'est pas parfaite.

Finalement, le tableau 4.1c présente les valeurs moyennes de la mesure de similarité sur les images reconfigurées sans post-traitement. Ici, contrairement au tableau 4.1a, l'entièreté de l'image est considérée. Les artéfacts ne sont donc pas masqués, et l'ensemble des pixels contribuent à l'évaluation de la qualité. Le tableau permet de constater que les artéfacts dus à la reprojection directe ont une influence assez forte sur la valeur de cette mesure de similarité. Cette constatation correspond à la réalité du visionnement de la séquence comme vidéo 3D, car ces artéfacts auront un impact certain sur la qualité de l'expérience. Comme dans les autres cas, la reprojection basée sur les valeurs de profondeur donne un score légèrement meilleur que celle basée sur les disparités.

4.1.4 Évaluation qualitative des résultats

Étant donné que le but du projet est de reconfigurer les séquences afin de conserver la même perception de profondeur, il importe aussi de valider la qualité de la reconfiguration de manière perceptuelle, à travers un humain. Pour ce faire, la qualité des séquences originales a tout d'abord été validée en les visualisant dans la configuration originale, c'est-à-dire dans la salle de cinéma de Sensio. Cette visualisation a

Reprojection	Directe		Inverse	
Basée sur les disparités	0.96933	0.96545	0.97593	0.97646
Basée sur la profondeur	0.99104	0.99178	0.99007	0.99045

(a) Images contenant les pixels reprojetés uniquement

Reprojection	Directe		Inverse	
Basée sur les disparités	0.93159	0.91426	0.93682	0.91614
Basée sur la profondeur	0.96082	0.94887	0.96342	0.94470

(b) Images traitées avec les algorithmes de remplissage

Reprojection	Directe		Inverse	
Basée sur les disparités	0.73077	0.73321	0.86714	0.87534
Basée sur la profondeur	0.78756	0.78658	0.90502	0.90521

(c) Images sans post-traitement, sans masque

Tableau 4.1 – Valeurs moyennes de la mesure de MS-SSIM sur la séquence *Chinchilla* reconfigurée pour la configuration *Samsung*. Sous chaque méthode (Directe et Inverse), la colonne de gauche représente la valeur pour les images de gauche, et celle de droite, pour les images de droite.

pu permettre de s'assurer que les paramètres d'acquisition avaient été bien choisis et qu'aucune distorsion visuelle préexistante n'était présente dans ces séquences.

Par la suite, les deux séquences synthétiques ont été visualisées dans la configuration *Samsung* sans être reconfigurées. Cela permettait de constater les distorsions induites par ce changement de configuration non compensé. Dans les deux cas, les objets de la scène semblaient aplatis et ils semblaient se situer à des profondeurs distinctes, ce que l'on sait ne pas être le cas dans les scènes réelles. Il s'agit là d'une instance de l'effet de « théâtre de marionnettes ». Les vérités-terrain des séquences dans la configuration *Samsung* ont ensuite été visualisées sur l'écran en question. Cela a permis d'observer que l'impression de profondeur était restaurée et très semblable à celle perçue dans la configuration *Cinéma* sur l'écran de cinéma. La première image de chaque séquence avait aussi été rendue en utilisant des valeurs de paramètres ne respectant pas le modèle proposé pour la configuration *Samsung*. Ces images ont été visualisées sur l'écran *Samsung*, et cela a permis de constater que l'effet 3D était dégradé, voire inexistant.

Ensuite, le processus de reconfiguration a été appliqué selon les différentes combinaisons de paramètres (par profondeur ou disparités, directe ou inverse). Une fois les séquences reconfigurées selon toutes les combinaisons de paramètres possibles, elles ont été visualisées une à une. Cette visualisation était divisée en deux étapes : l'utilisateur commençait par n'observer qu'une paire stéréoscopique de la séquence, puis il visualisait la séquence en entier, comme une séquence vidéo. La première partie permettait d'identifier les artéfacts et de bien pouvoir analyser les endroits où la reconfiguration échouait. La deuxième partie simulait le but des utilisateurs normaux du cadre de traitement, c'est-à-dire obtenir une séquence vidéo reconfigurée. Il est important de noter que, dans le cas de la vidéo, certains artéfacts ne sont pas per-

ceptibles, car ils ne se produisent que dans une paire stéréoscopique. Comme chaque paire n'est visible que pendant $\frac{1}{30}$ de seconde, la vision humaine n'est pas assez rapide pour percevoir ces artéfacts.

Cette évaluation a permis d'arriver à certaines conclusions, qui sont présentées dans les paragraphes suivants.

Artéfacts de désoccultation Peu importe la méthode de reconfiguration utilisée, l'artéfact le plus apparent et qui dégrade le plus la qualité de l'expérience est l'artéfact de désoccultation (voir section 3.6.2). Il est habituellement apparent parce que de grande dimension (dépendamment de la dimension de l'objet occultant). De plus, comme les artéfacts de désoccultation ne sont pas aux mêmes endroits dans les images de gauche et de droite, la perception de la scène est dérangée, car il manque de l'information différente pour chaque oeil. Évidemment, ces artéfacts sont normalement réduits ou éliminés par les techniques de réduction d'artéfacts. Cependant, si celles-ci ne sont pas parfaites, il peut arriver que l'information ne soit pas tout à fait consistante entre l'oeil gauche et l'oeil droit, ce qui peut causer un léger inconfort lors de la visualisation de la séquence.

Artéfacts de fissures Le deuxième type d'artéfact qui dégrade la qualité perceptuelle de la séquence est l'artéfact de fissure (voir section 3.6.3). Puisque cet artéfact se produit principalement sur des objets, ceux-ci perdent de leur réalisme. En effet, si un personnage principal de la séquence se retrouve séparé en deux ou trois parties, il est difficile pour le spectateur de percevoir la séquence comme étant réaliste. La technique de remplissage de trous peut parfois remédier à ce problème, mais, selon la largeur de la fissure, il est possible que l'objet ait plutôt l'air étiré que l'air réa-

liste. De plus, dépendamment de la scène, il peut arriver que la fissure aït été remplie lors de la reprojection si de l'information de la scène était disponible. Dans ce cas, l'algorithme de remplissage de trous ne considèrera pas la fissure comme un trou à remplir, et l'artéfact restera donc présent. Le problème de fissures peut être amoindri en utilisant des cartes de disparités pour lesquelles δd, la différence minimale entre deux disparités, est plus petit que 1.

Cas idéal de la reconfiguration Suivant les conclusions des deux derniers paragraphes, la méthode idéale pour reconfigurer est la reprojection inverse basée sur la profondeur. Dans ce cas, les seuls artéfacts présents sont des artéfacts de désocculta-tion, qui seront éliminés par la technique d'emplissage de trous. De plus, il est possible d'utiliser directement les valeurs de profondeur afin de guider cet algorithme d'emplissage et de restreindre les pixels pouvant fournir de l'information à l'algorithme de manière plus précise qu'en se basant sur les disparités. Malheureusement, il est rare de pouvoir directement appliquer la reprojection basée sur la profondeur dans un contexte réel, puisque les valeurs de profondeur sont très rarement disponibles.

Choix entre reprojection directe et inverse Selon les résultats présentés, le lecteur peut constater qu'il est toujours plus avisé d'utiliser la reprojection inverse plutôt que la reprojection directe. Cela permet d'éliminer les artéfacts de trous uniques et de lignes non-remplies, tout en obtenant la même image de la scène. Par contre, la reprojection inverse est six fois plus lente que la reprojection directe. Dans les cas où l'utilisateur a des contraintes de temps de calcul très serrées, il sera possiblement obligé de choisir la reprojection directe et d'utiliser une technique de remplissage pour les trous uniques et les lignes non-remplies. Il s'agira de choisir le meilleur compromis.

Importance de l'algorithme d'emplissage de trous Une autre constatation est l'importance d'avoir un bon algorithme d'emplissage de trous. En effet, si les trous ne sont pas emplis, la séquence n'est souvent pas perceptuellement intéressante. S'ils sont emplis avec un algorithme ne considérant pas le contexte stéréoscopique, le résultat sera à peine plus intéressant à regarder. Si, au contraire, l'algorithme d'emplissage prend compte du contexte stéréoscopique de la séquence, le résultat est beaucoup plus satisfaisant. Il a aussi été constaté qu'il serait intéressant d'avoir un algorithme d'emplissage utilisant le voisinage temporel de la paire courante, afin d'avoir une certaine consistance temporelle, qui améliorerait encore la qualité des séquences produites.

Influence de la visualisation d'une séquence animée Il a été constaté que le fait de visualiser une séquence animée (par rapport à la visualisation image par image) influence la qualité perçue. En effet, lorsque la séquence est vue à 24 ou 30 trames par seconde, certains artéfacts ne sont pas perceptibles à cause des limites du système visuel humain. Par exemple, s'il n'y a qu'un trou de quelques pixels sur une seule trame de la séquence, il ne sera pas perceptible dans la séquence finale. Cela peut donc influencer le choix des algorithmes de correction d'artéfacts.

Importance de la consistance temporelle des cartes de disparités La dernière constatation est qu'il est important que les cartes de disparités soit temporellement consistantes. Si elles ne le sont pas, les images reconfigurées peuvent différer passablement d'une trame à l'autre, ce qui peut causer d'importants désagréments perceptuels. Si les cartes de disparités varient de manière graduelle à chaque trame, le problème ne se présente pas.

4.2 Validation sur des séquences réelles

Une fois la validation sur les séquences synthétiques effectuée, il est nécessaire de valider l'efficacité du cadre proposé sur des séquences réelles. Dans le cadre de la collaboration avec la compagnie Sensio, plusieurs séquences ont été fournies. Malheureusement, pour toutes ces séquences, il n'y avait que la configuration *Cinéma* qui était disponible. En effet, les séquences n'ont été acquises que pour une seule configuration, comme c'est pratiquement toujours le cas pour les séquences réelles. Il est donc impossible de valider quantitativement la qualité de la reprojection en comparant avec une vérité-terrain, puisqu'elle n'existe pas. Il n'y a donc eu qu'une évaluation qualitative des résultats.

Il aurait été possible d'acquérir des séquences suivant le modèle d'acquisition pour les deux configurations. Cependant, le problème d'être capable d'obtenir deux fois une séquence avec exactement le même contenu (même mouvement du sujet, même déplacement des caméras) est toujours présent. De plus, il est difficile de contrôler avec précision l'espacement et l'angulation des caméras dans un contexte non-professionnel. Cette solution a donc été écartée.

4.2.1 Description des séquences

Trois séquences réelles ont été sélectionnées pour tester la reconfiguration sur des séquences réelles. Elles ont été sélectionnées car chacune d'entre elle a un contenu différent des autres, ce qui permet de tester la méthode sur divers contenus de scène.

La première séquence, appelée *Camion*, représente une camionnette qui se déplace à l'intérieur d'une forêt, puis dans un champ (voir figures 4.6a et 4.6b). Cette séquence

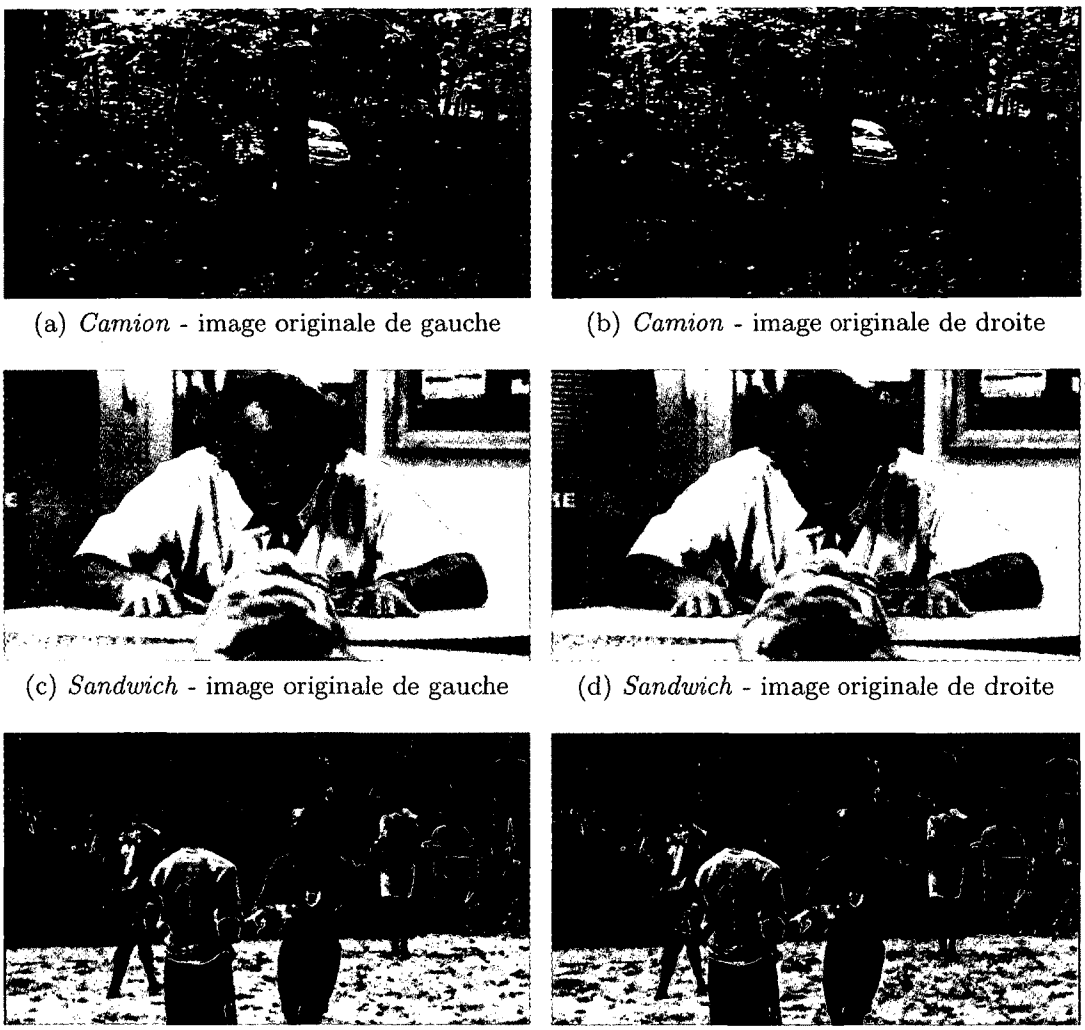
présente deux principales difficultés : la vitesse de déplacement de la camionnette d'une trame à l'autre, et le contenu végétal de la scène. Le contenu végétal pose principalement problème à l'algorithme de mise en correspondance stéréo, car les feuilles sont de petits objets de couleurs très similaires, et donc difficilement différentiables.

La deuxième séquence, nommée *Sandwich*, illustre un employé assis à une table et lorgnant un sandwich (voir les figures 4.6c et 4.6d). Les difficultés de cette séquence sont que les premiers objets sont très près des caméras, et que les couleurs de différents objets sont assez similaires. Cette similarité des couleurs peut encore poser des problèmes à l'algorithme de mise en correspondance stéréo responsable du calcul des cartes de disparités.

La dernière séquence est appelée *Carton jaune* et représente un match de soccer sur sable où un joueur reçoit un carton jaune de la part de l'arbitre (voir les figures 4.6e et 4.6f). Cette séquence comporte plusieurs difficultés. Tout d'abord, il y a beaucoup d'humains repartis dans la scène, à différentes profondeurs. De plus, il y a plusieurs zones d'ombres et quelques fenêtres à l'arrière-plan, ce qui peut aussi causer des problèmes lors de l'obtention des cartes de disparités.

4.2.2 Protocole expérimental

Les séquences réelles utilisées pour l'expérimentation ne sont pas fournies avec des cartes de profondeur. Il est donc obligatoire d'utiliser la reprojection basée sur les disparités. Afin de calculer celles-ci, les images ont tout d'abord été rectifiées à l'aide d'une technique proposée par Julien Prémont [42], puisque cette technique présente une formulation simplifiée de la rectification dans le cas où il n'y a qu'une rotation entre les caméras. Une fois les images rectifiées, les cartes de disparités ont été calculées



(a) *Camion* - image originale de gauche (b) *Camion* - image originale de droite

(c) *Sandwich* - image originale de gauche (d) *Sandwich* - image originale de droite

(e) *Carton jaune* - image originale de gauche (f) *Carton jaune* - image originale de droite

Figure 4.6 – Images originales des séquences réelles dans la configuration *Cinéma* : (a), (b) : *Camion* - images de gauche et droite, respectivement, (c), (d) *Sandwich* - images de gauche et droite, respectivement, (e), (f) *Carton jaune* - images de gauche et droite, respectivement.

avec l'algorithme de stéréovision proposé en [75]. Cet algorithme a été choisi parce que, selon les auteurs de la méthode, il préserve bien les frontières des objets, ce qui est primordial lorsque l'on veut simuler un déplacement des caméras dans le cadre de la reconfiguration. De plus, au moment du choix de l'algorithme, c'était une des méthodes ayant la meilleure performance dans l'évaluation de Middlebury [45]. Les séquences ont ensuite été reprojetées en se basant sur les disparités, à la fois avec la méthode directe et la méthode inverse. Finalement, les techniques de réduction d'artéfacts ont été utilisées, afin de réduire les artéfacts visibles. Les résultats de cette reconfiguration sont visibles à la figure 4.7.

4.2.3 Évaluation humaine de la qualité

Aucune vérité-terrain de la configuration *Samsung* n'étant disponible pour les séquences réelles, il était impossible d'effectuer une évaluation algorithmique de la qualité. Il a donc fallu se fier uniquement sur la validation humaine de la qualité. Comme dans le cas des séquences synthétiques, cette évaluation permet de vérifier si l'effet de profondeur est mieux recréé dans une séquence reconfigurée que lorsque l'utilisateur visualise les séquences originales dans une configuration incompatible. Elle permet aussi d'évaluer l'impact visuel des artéfacts de reprojection.

Afin de pouvoir comparer la qualité de l'effet de profondeur entre séquences originales et séquences reconfigurées, les séquences originales ont été visualisées dans la configuration *Samsung*, afin de constater l'impact de la distorsion. Dans ce cas, comme pour les séquences synthétiques, certains objets avaient l'air aplatis, et les séquences souffraient de l'effet « théâtre de marionnettes ». Par la suite, les séquences reconfigurées ont été visualisées dans la même configuration. L'effet de profondeur était beaucoup

(a) *Camion* - reprojection directe par les disparités

(b) *Camion* - artéfacts traités

(c) *Sandwich* - reprojection directe par les disparités

(d) *Sandwich* - artéfacts traités

(e) *Carton jaune* - reprojection directe par les disparités

(f) *Carton jaune* - artéfacts traités

Figure 4.7 – Images des séquences réelles reconfigurées pour *Samsung*, sans et avec post-traitement.

plus semblable à celui des séquences originales visualisées dans la configuration *Cinéma*. Cependant, les artéfacts présents dans les séquences et principalement causés par les erreurs dans les cartes de disparités causaient certains désagréments lors du visionnement.

Cette visualisation des séquences reconfigurées a permis d'arriver à plusieurs conclusions. Tout d'abord, il a été constaté que toutes les observations faites dans le cas de l'évaluation humaine de la qualité des séquences synthétiques reconfigurées sont aussi valides dans le cas des séquences réelles reconfigurées (voir la section 4.1.4). Certains conclusions supplémentaires sont présentées dans les paragraphes suivants.

Influence de l'algorithme de stéréovision L'algorithme de stéréovision est probablement le composant du cadre de traitement ayant la plus grande influence sur la qualité des résultats sur les séquences réelles. Étant donné que, pour les séquences synthétiques, les cartes de disparités avaient été calculées à partir des cartes de profondeur, les limites de l'algorithme de mise en correspondance stéréoscopique n'avaient pas été mises en évidence. Cependant, pour les séquences réelles, ces limites sont évidentes. Si l'algorithme de mise en correspondance fait une erreur au niveau d'un objet se situant à l'avant-plan, son déplacement dû à la reprojection sera peu réaliste. De plus, si la disparité est mal calculée, il se peut que les zones vides dues aux désocclusions soient très grandes et difficiles à combler pour l'algorithme d'emplissage de trous.

Mise en correspondance de petits objets Si l'algorithme de mise en correspondance a tendance à faire disparaître les petits objets des cartes de disparités, certains problèmes de reprojection peuvent se produire. Par exemple, dans la séquence *Ca-*

mion, le spectateur peut apercevoir le sol de la forêt entre certaines feuilles. Il arrive que, lors de la mise en correspondance, ces zones de sol reçoivent la même valeur de disparité que les feuilles l'entourant, alors que ce ne devrait pas être le cas. Dans ce cas, lors de la reprojection, l'effet de profondeur associé au sol dans cette région paraît invraisemblable. De plus, lors de l'emplissage de trous, il se peut qu'un trou entre les feuilles soit empli de vert, alors qu'il aurait dû être empli de la couleur du sol.

Scènes acquises de loin Pour des séquences qui ont été filmées de loin, l'intervalle des valeurs de disparités peut être relativement petit. Si le passage de la configuration source à la configuration destination implique des changements significatifs dans les paramètres des caméras, il peut se créer de grandes zones vides dues à la désocculation. Plus ces zones sont grandes, plus les artéfacts seront visibles, même après l'emplissage de trous. Cela pourrait donc diminuer l'applicabilité de la méthode pour certaines scènes précises, ou pour certains configurations précises.

Problématiques potentielles Certaines problématiques qui pourraient se poser dans l'utilisation de la méthode n'ont pas été constatées lors de l'expérimentation, mais elles existent tout de même.

Tout d'abord, les séquences à reconfigurer pourraient être pourvues d'un marquage visuel identifiant la compagnie les ayant réalisées. Ce marquage est un texte en superposition dans l'un des coins des trames, et n'apparaît souvent qu'au début ou à la fin de la séquence. Ce marquage cause problème parce qu'il cause des erreurs de mise en correspondance dans cette zone. De plus, le marquage devrait logiquement être appliqué au même endroit dans la séquence reconfigurée, car il ne s'agit pas d'une élément de la scène, mais d'un ajout graphique. Or, si le cadre de traitement est

directement appliqué, ce marquage est souvent déplacé et déformé. Il faudrait créer manuellement un masque, et ne l'appliquer que pour les trames dans lesquelles le marquage est présent.

Il existe aussi certains scénarios où la mise en correspondance peut ne pas fonctionner correctement. Par exemple, si un objet dont la surface est faite de miroir est présent dans la scène, l'algorithme de stéréovision ne pourra probablement pas déduire que c'est un miroir. Les valeurs de disparités ne seront donc pas bonnes, et la reprojection se retrouverait faussée. Il en est de même pour les objets transparents ou possédant des réflexions spéculaires. Ces réflexions doivent normalement suivre la source de lumière les créant, mais lors de la reprojection, elles seraient interpolées au mauvais endroit. Il faudrait donc trouver un moyen de gérer ces cas.

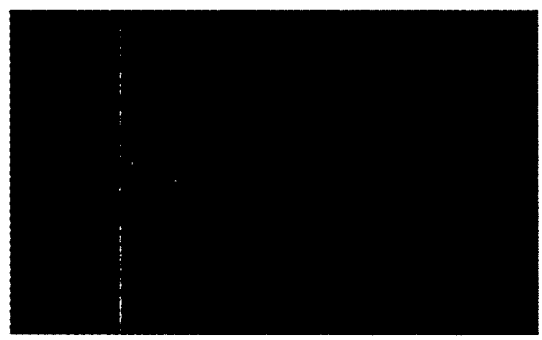
4.3 Évaluation de l'algorithme d'emplissage de trous

L'algorithme d'emplissage de trous présenté à la section 3.6.2 et basé sur les travaux de Criminisi *et al.* [12] a été validé, pour en justifier l'utilisation par rapport à l'algorithme original. La première validation a été effectuée sur une image de la séquence *Cube et sphères* reconfigurée pour la configuration *Samsung* par reprojection inverse basée sur les disparités (voir figure 4.8b). La deuxième validation a été effectuée sur une image de la séquence *Chinchilla* reconfigurée pour l'écran *Samsung* par reprojection inverse basée sur la profondeur (voir la figure 4.9b). Les deux images reconfigurées ont ensuite été traitées avec l'algorithme de Criminisi et l'algorithme modifié présenté dans ce mémoire. Les résultats de ces emplissages de trous sont présentés aux figures 4.8c, 4.8d, 4.9c et 4.9d.

(a) Vérité-terrain

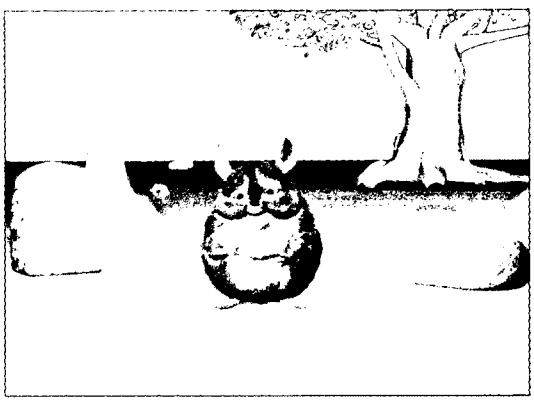
(b) Reprojection sans emplissage de trous

(c) Trous emplis par la technique de Criminisi *et al.* (voir [12])

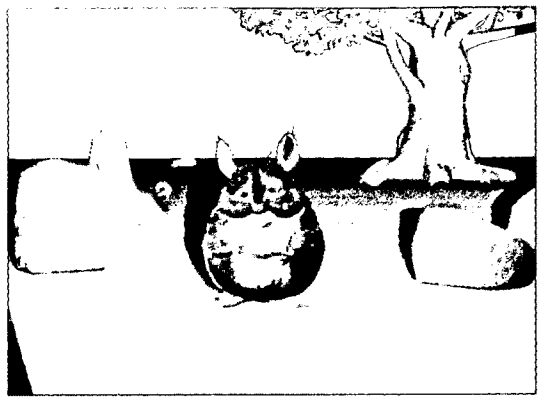


(d) Trous emplis par la version modifiée de l'algorithme

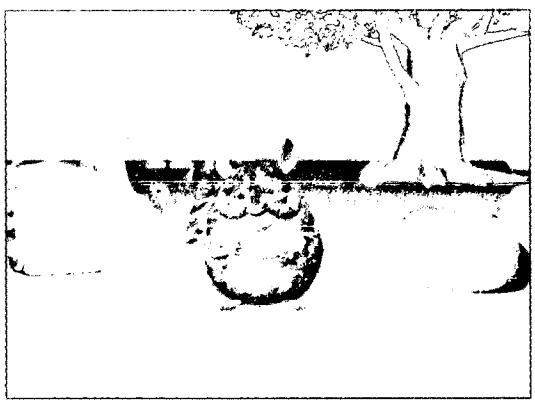
Figure 4.8 – Résultats de la reprojection et de l'emplissage de trous sur une image de gauche de la séquence *Cube et sphères* pour la configuration *Samsung* : (a) vérité-terrain de l'image de gauche dans la configuration *Samsung*, (b) reprojection inverse basée sur les disparités, sans emplissage de trous, (c) image reconfigurée dont les trous ont été emplis avec la technique de Criminisi *et al.* [12], (d) image reconfigurée dont les trous ont été emplis avec la technique proposée dans ce mémoire.



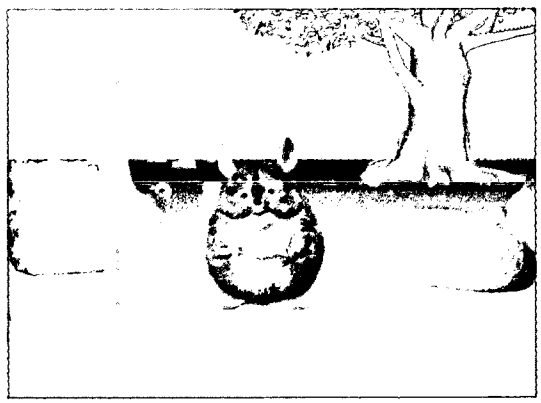
(a) Vérité-terrain



(b) Reprojection sans emplissage de trous



(c) Trous emplis par la technique de Criminisi *et al.* (voir [12])



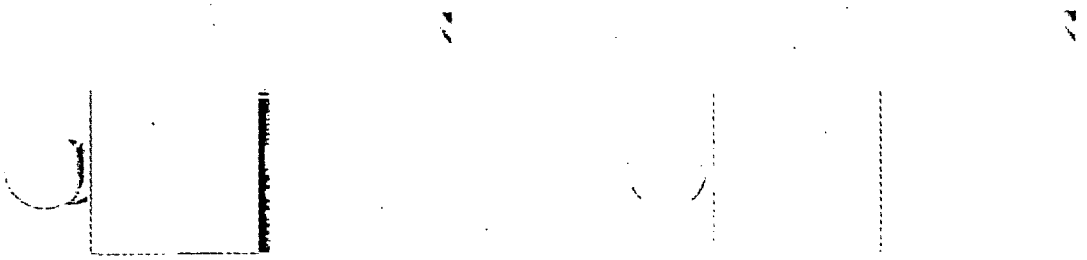
(d) Trous emplis par la version modifiée de l'algorithme

Figure 4.9 – Résultats de la reprojection et de l'emplissage de trous sur une image de gauche de la séquence *Chinchilla* pour la configuration *Samsung* : (a) vérité-terrain de l'image de gauche dans la configuration *Samsung*, (b) reprojection inverse basée sur la profondeur, sans emplissage de trous, (c) image reconfigurée dont les trous ont été emplis avec la technique de Criminisi *et al.* [12], (d) image reconfigurée dont les trous ont été emplis avec la technique proposée dans ce mémoire.

Par la suite, les images de vérité-terrain ont été soustraites de chacune des images finales correspondantes, afin d'identifier les pixels différents. Les résultat de ces comparaisons sont visibles aux figures 4.10 et 4.11.

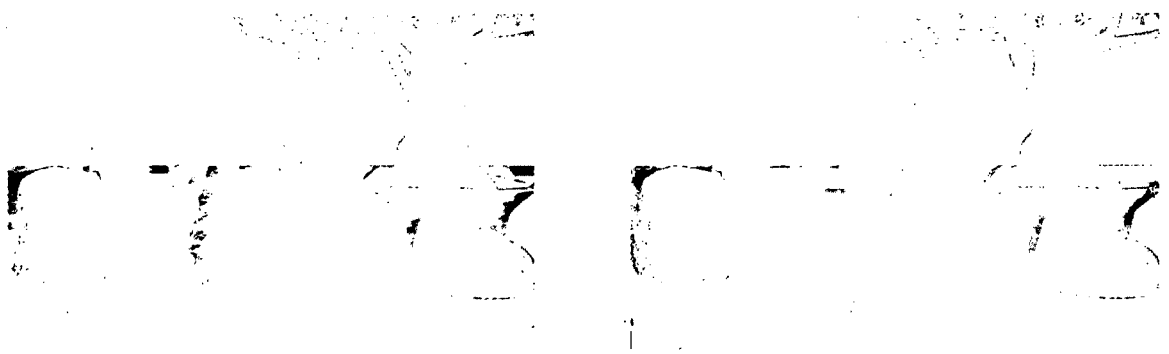
Pour ce qui est de la séquence *Cube et sphères*, deux constatations s'imposent : la méthode proposée dans ce mémoire produit de meilleurs résultats dans le cas de scènes simples, et certaines régions sont problématiques pour les deux méthodes. En effet, le coin supérieur droit, où devrait normalement se trouver une partie de sphère, a été problématique pour les deux méthodes. Cependant, c'est inévitable, puisque qu'aucune information de cette sphère n'était visible avant l'emplissage de trous. Visuellement, la différence semble moins grande pour l'image traitée avec l'algorithme modifié. Afin de s'en assurer, les différences ont été quantifiées. Pour l'image provenant de l'algorithme de Criminisi *et al.*, il y a 17,59% des pixels qui sont erronés, et la différence moyenne de valeur est de 4,71. Pour l'image provenant de l'algorithme modifié, 16,29% des pixels sont erronés, pour une différence moyenne de 1,79. Dans les deux cas, les valeurs des images sont dans l'intervalle $[0, 255]$, et donc, les valeurs de différences moyennes sont aussi dans cet interval.

Dans le cas de la séquence *Chinchilla*, le lecteur peut constater que la méthode de Criminisi *et al.* produit des erreurs perceptuellement plus importantes que la méthode proposée dans ce mémoire. Si la séquence est visualisée comme une vidéo, le fait que l'herbe à gauche du chinchilla soit d'un vert un peu différent de celui dans la vérité-terrain est moins dérangeant que si la figure de ce chinchilla se trouve reproduite à ce même endroit. Évidemment, le succès des deux méthodes d'emplissage de trous sera toujours dépendant du contenu de la scène.



(a) Algorithme de Criminisi *et al.* (b) Algorithme proposé dans ce mémoire

Figure 4.10 – Différences de couleur des pixels entre les images dont les trous ont été emplis et la vérité-terrain sur une image de gauche de la séquence *Cube et sphères* pour la configuration *Samsung*.



(a) Algorithme de Criminisi *et al.* (b) Algorithme proposé dans ce mémoire

Figure 4.11 – Différences de couleur des pixels entre les images dont les trous ont été emplis et la vérité-terrain sur une image de gauche de la séquence *Chinchilla* pour la configuration *Samsung*.

CONCLUSION ET PERSPECTIVES

Conclusion

Ce mémoire présente une méthode permettant de reconfigurer des séquences stéréoscopiques acquises pour une certaine configuration de visionnement afin qu'elles procurent la même qualité d'effet 3D dans une nouvelle configuration de visionnement. Cette méthode est basée sur un modèle géométrique d'acquisition et de visionnement de séquences 3D. Ce modèle permet de choisir la valeur des paramètres d'acquisition à utiliser selon la configuration ciblée pour le visionnement, de même que selon l'accentuation ou la diminution voulue de l'effet 3D.

La méthode est décrite par un cadre de traitement séquentiel pour le traitement des séquences. Les paramètres originaux d'acquisition sont tout d'abord déduits à l'aide du modèle géométrique présenté plus tôt. Par la suite, les paramètres des caméras virtuelles sont calculés selon la nouvelle configuration désirée. La structure du contenu de la scène est ensuite estimée, que ce soit en se basant sur des cartes de profondeur ou sur des cartes de disparités. Les images reconfigurées sont rendues selon la structure calculée, puis elles sont traitées avec des algorithmes de réduction d'artéfacts. Un algorithme d'emplissage de trous adaptant une méthode préexistante au contexte

stéréoscopique a aussi été présenté.

La méthode proposée a été validée sur des séquences synthétiques et réelles. Cela a permis de constater qu'étant donné des cartes de profondeur, la méthode produit des images reconfigurées extrêmement semblables à ce qu'elles auraient dues être en réalité. Dans le cas d'images reconfigurées sur la base de valeurs de disparités, la qualité des résultats est moindre. Cependant, si des cartes de disparités de très grande qualité sont disponibles, les résultats seraient améliorés. Pour ce qui est de l'algorithme d'emplissage de trous, les résultats ne sont pas encore parfaits, mais sont meilleurs qu'en utilisant la version originale du même algorithme.

Il a donc été montré que le modèle géométrique et le cadre de traitement proposés sont valides et que le résultat final de la reconfiguration est dépendant de la qualité des résultats intermédiaires. Meilleurs sont ces résultats intermédiaires, meilleure sera la reconfiguration.

Perspectives

Il serait intéressant de tenter d'appliquer la méthode à une reprojection basée sur la position du spectateur. Ainsi, si chaque spectateur pouvait percevoir sa propre paire d'images stéréoscopiques, celles-ci pourraient être ajustées en fonction de la position de chaque spectateur. Cela permettrait d'avoir un effet 3D n'ayant pratiquement aucune distorsion pour tous les spectateurs.

Certaines opportunités d'amélioration de la méthode seraient à étudier. Parmi celles-ci, notons l'utilisation possible d'un algorithme de mise en correspondance stéréoscopique se basant non seulement sur un raisonnement spatial, mais aussi temporel.

Certaines méthodes de ce type existent déjà, mais ne produisent des résultats intéressants que pour certains types de déplacements. Nous avons tenté de développer une telle méthode, mais les artéfacts introduits n'amélioraient pas la qualité de la reconfiguration. De même, si des cartes de disparités dont l'intervalle des valeurs serait plus petit que 1 étaient disponibles, la qualité des images résultantes serait encore améliorée.

Une autre voie d'amélioration consiste en l'amélioration de l'algorithme d'emplissage de trous. Pour des scènes simples et synthétiques, l'algorithme fonctionne bien. Cependant, pour des scènes plus complexes et réalistes, les contraintes sur l'information à utiliser devraient être posées de manière plus flexible.

Finalement, si une version en temps-réel de l'algorithme est désirée, il faudra poser certaines contraintes sur les paramètres à utiliser. De plus, il faudra qu'une carte de profondeur ou de disparités soit disponible *a priori*, car le calcul des disparités est l'étape la plus longue du traitement. Il faudra aussi simplifier l'algorithme d'emplissage de trous.

Bibliographie

[1] Big buck bunny. `http://www.bigbuckbunny.org/`, 2012.

[2] Blender. `http://www.blender.org/`, 2012.

[3] S. ADEDOYIN, W.A.C. FERNANDO et A. AGGOUN : A joint motion & disparity motion estimation technique for 3D integral video compression using evolutionary strategy. *IEEE Transactions on Consumer Electronics*, 53(2):732–739, 2007.

[4] S. ADEDOYIN, W.A.C. FERNANDO, A. AGGOUN et K.M. KONDOZ : Motion and disparity estimation with self adapted evolutionary strategy in 3D video coding. *IEEE Transactions on Consumer Electronics*, 53(4):1768–1775, 2007.

[5] E.H. ADELSON et J.R. BERGEN : The plenoptic function and the elements of early vision. Dans *Computational Models of Visual Processing*, pages 3–20. MIT Press, 1991.

[6] A. ARIYAEEINIA : Analysis and design of stereoscopic television systems. *Signal Processing : Image Communication*, 13(3):201–208, septembre 1998.

[7] M. BERTALMIO, L. VESE, G. SAPIRO et S. OSHER : Simultaneous structure and texture image inpainting. *IEEE Transactions on Image Processing*, 12(8):882–9, janvier 2003.

[8] S. BIRCHFIELD et C. TOMASI : A pixel dissimilarity measure that is insensitive to image sampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(4):401–406, avril 1998.

[9] Y. BOYKOV et V. KOLMOGOROV : An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–37, septembre 2004.

[10] Y. BOYKOV, O. VEKSLER et R. ZABIH : Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.

[11] S.E. CHEN et L. WILLIAMS : View interpolation for image synthesis. Dans *SIGGRAPH '93 : Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 279–288, New York, NY, USA, 1993. ACM.

[12] A. CRIMINISI, P. PEREZ et K. TOYAMA : Object removal by exemplar-based inpainting. Dans *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 2, pages 721–728, Los Alamitos, CA, USA, 2003.

[13] B. CURLESS, S.M. SEITZ et Z. LI : Spacetime stereo : shape recovery for dynamic scenes. Dans *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 2, pages 367–374, Los Alamitos, CA, USA, 2003.

[14] J. DAVIS, D. NEHAB, R. RAMAMOORTHI et S. RUSINKIEWICZ : Spacetime stereo : a unifying framework for depth from triangulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(2):296–302, mars 2005.

[15] J. DAVIS, R. RAMAMOORTHI et S. RUSINKIEWICZ : Spacetime stereo : a unifying framework for depth from triangulation. Dans *Computer Vision and Pattern*

Recognition, IEEE Computer Society Conference on, vol. 2, pages 359–366, Los Alamitos, CA, USA, 2003. IEEE Comput. Soc.

[16] P. DEBEVEC, Y. YU et G. BORSHUKOV : Efficient view-dependent image-based rendering with projective texture-mapping. Rapport technique, University of California at Berkeley, Berkeley, CA, 1998.

[17] P.E. DEBEVEC, C.J. TAYLOR et J. MALIK : Modeling and rendering architecture from photographs : A hybrid geometry- and image-based approach. Dans *SIGGRAPH '96 : Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 11–20, New York, New York, USA, 1996. ACM Press.

[18] G. EGNAL et R.P. WILDES : Detecting binocular half-occlusions : empirical comparisons of five approaches. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1127–1133, août 2002.

[19] A. FITZGIBBON, Y. WEXLER et A. ZISSERMAN : Image-based rendering using image-based priors. Dans *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, vol. 2, pages 1176–1183, octobre 2003.

[20] G. GALES, A. CROUZIL et S. CHAMBON : A region-based randomized voting scheme for stereo matching. Dans *Proceedings of the 6th international conference on Advances in visual computing - Volume Part II*, ISVC'10, pages 182–191. Springer-Verlag, 2010.

[21] S.J. GORTLER, R. GRZESZCZUK, R. SZELISKI et M.F. COHEN : The lumigraph. Dans *SIGGRAPH '96 : Proceedings of the 23rd annual conference on Computer*

graphics and interactive techniques, pages 43–54, New York, New York, USA, 1996. ACM Press.

[22] M.J. HANNAH : *Computer matching of areas in stereo images.* Thèse de doctorat, Stanford, CA, USA, 1974.

[23] R. HARTLEY et A. ZISSERMAN : *Multiple view geometry in computer vision.* Cambridge University Press, deuxième édition, 2004.

[24] L.F. HODGES et E. THORPE DAVIS : Geometric considerations for stereoscopic virtual environments. Rapport technique, Georgia Institute of Technology, Atlanta, Georgia, USA, 1993.

[25] D.M. HOFFMAN, A.R. GIRSHICK, K. AKELEY et M.S. BANKS : Vergence – accommodation conflicts hinder visual performance and cause visual fatigue. *Journal of Vision*, 8(3):1–30, 2008.

[26] N.S. HOLLIMAN : Mapping perceived depth to regions of interest in stereoscopic images. Dans *Stereoscopic Displays and Virtual Reality Systems XI*, vol. 1, pages 117–128, San Jose, California, 2004.

[27] M. HUMENBERGER, T. ENGELKE et W. KUBINGER : A census-based stereo vision algorithm using modified Semi-Global Matching and plane fitting to improve matching quality. Dans *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, pages 77–84. IEEE, juin 2010.

[28] W.A. IJSSELSTEIJN, H. de RIDDER et J VLIEGEN : Subjective evaluation of stereoscopic images : effects of camera parameters and display duration. *IEEE Transactions on Circuits and Systems for Video Technology*, 10(2):225–233, mars 2000.

[29] S. INCE et J. KONRAD : Geometry-based estimation of occlusions from video frame pairs. Dans *ICASSP '05 : IEEE International Conference on Acoustics, Speech, and Signal Processing.*, pages 933–936. IEEE, 2005.

[30] G.R. JONES : Controlling perceived depth in stereoscopic images. Dans *Stereoscopic Displays and Virtual Reality Systems VIII*, pages 42–53. SPIE, 2001.

[31] V. KOLMOGOROV et R. ZABIH : Computing visual correspondence with occlusions using graph cuts. Dans *Eighth IEEE International Conference on Computer Vision, ICCV 2001*, vol. 2, pages 508–515. IEEE Computer Society, 2001.

[32] V. KOLMOGOROV et R. ZABIH : Graph cut algorithms for binocular stereo with occlusions. Dans *The Handbook of Mathematical Models in Computer Vision*, pages 1–17. Springer, 2005.

[33] J. KONRAD et M. HALLE : 3-d displays and signal processing : An answer to 3-d ills ? *IEEE Signal Processing Magazine*, 24:97–111, 2007.

[34] R. KUTKA : Reconstruction of correct 3-D perception on screens viewed at different distances. *IEEE Transactions on Communications*, 42(1):29–33, 1994.

[35] S. LEE et W. KANG : Horizontal parallax distortion correction method in toed-in camera with wide-angle lens. Dans *2009 3DTV Conference : The True Vision - Capture, Transmission and Display of 3D Video*, pages 1–4. IEEE, mai 2009.

[36] A. LEVIN, A. ZOMET et Y. WEISS : Learning how to inpaint from global image statistics. Dans *Ninth IEEE International Conference on Computer Vision, ICCV 2003*, pages 305–312. IEEE Computer Society, 2003.

[37] M. LEVOY et P. HANRAHAN : Light field rendering. Dans *SIGGRAPH '96 : Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 31–42. ACM Press, 1996.

[38] L. Jr MCMILLAN : *An image-based approach to three-dimensional computer graphics.* Thèse de doctorat, University of North Carolina at Chapel Hill, 1997.

[39] S. MILLS : Stereo-motion analysis of image sequences. Dans *Digital Image Vision Computing : Techniques and Applications (DICTA'97)*, pages 515–520, 1997.

[40] Y. MORVAN : *Acquisition, compression and rendering of depth and texture for multi-view video.* Thèse de doctorat, 2009.

[41] S.E. PALMER : *Vision science : photons to phenomenology.* Bradford Books. MIT Press, 1999.

[42] J. PRÉMONT : Reconstruction de primitives géométriques avec de la lumière non structurée. Mémoire de Maîtrise, U. de Sherbrooke, Sherbrooke, mai 2011.

[43] A. REDERT, E. HENDRIKS et J. BIEMOND : Synthesis of multi viewpoint images at non-intermediate positions. Dans *Acoustics, Speech, and Signal Processing 1997 (ICASSP-97)*, vol. 4, pages 2749–2752. IEEE Computer Society, avril 1997.

[44] S.-M. RHEE, J. CHOI et U. NEUMANN : Stereoscopic view synthesis by view morphing. Dans *Advances in Visual Computing*, vol. 5359 de *Lecture Notes in Computer Science*, pages 924–933. Springer Berlin / Heidelberg, 2008.

[45] D. SCHARSTEIN et R. SZELISKI : The middlebury stereo vision page. URL `http://vision.middlebury.edu/stereo/`. Site visité le 7 septembre 2011.

[46] D. SCHARSTEIN et R. SZELISKI : A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47 (1-3):7–42, avril 2002.

[47] S.M. SEITZ et C.R. DYER : Toward image-based scene representation using view morphing. Dans *Pattern Recognition, 1996., Proceedings of the 13th International Conference on*, vol. 1, pages 84–89 vol.1, août 1996.

[48] S.M. SEITZ et C.R. DYER : View morphing. Dans *SIGGRAPH '96 : Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 21–30, New York, NY, USA, 1996. ACM.

[49] J. SHADE, S. GORTLER, L.-W. HE et R. SZELISKI : Layered depth images. *SIGGRAPH '98 : Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 231–242, 1998.

[50] J. SHAO : Combination of stereo, motion and rendering for 3d footage display. Dans *Stereo and Multi-Baseline Vision, 2001 (SMBV 2001)*, pages 95–102, 2001.

[51] H. SHUM et S.B. KANG : Review of image-based rendering techniques. Dans *Visual Communications and Image Processing*, pages 2–13, 2000.

[52] H.-Y. SHUM et L.-W. HE : Rendering with concentric mosaics. Dans *SIGGRAPH '99 : Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 299–306. ACM Press, 1999.

[53] J. SUN, N.-N. ZHENG et H.-Y. SHUM : Stereo matching using belief propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(7):787–800, juillet 2003.

[54] R. SZELISKI et D. SCHARSTEIN : Sampling the disparity space image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(3):419–25, mars 2004.

[55] R. SZELISKI, R. ZABIH, D. SCHARSTEIN, O. VEKSLER, Vé KOLMOGOROV, A. AGARWALA, M. TAPPEN et C. ROTHER : A comparative study of energy minimization methods for Markov random fields with smoothness-based priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6):1068–1080, juin 2008.

[56] Z. TAUBER, Z. LI et M. DREW : Review and preview : disocclusion by inpainting for image-based rendering. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, 37(4):527–540, 2007.

[57] G.A. TRIANTAFYLLIDIS, D. TZOVARAS et M.G. STRINTZIS : Detection of occlusion and visible background and foreground areas in stereo image pairs. Dans *9th International Conference on Electronics, Circuits and Systems*, pages 1019–1022. IEEE, 2002.

[58] E. TRUCCO et A. VERRI : *Introductory techniques for 3-D computer vision.* Prentice Hall, 1998.

[59] Z. WANG, E.P. SIMONCELLI et A.C. BOVIK : Multiscale structural similarity for image quality assessment. Dans *Proc 37th Asilomar Conf on Signals, Systems and Computers*, vol. 2, pages 1398–1402. IEEE Computer Society, novembre 2003.

[60] Z.-F. WANG et Z.-G. ZHENG : A region based stereo matching algorithm using cooperative optimization. Dans *Computer Vision and Pattern Recognition 2008*

(CVPR 2008), IEEE Computer Society Conference on, pages 1–8. IEEE, juin 2008.

[61] Y. WEI et L. QUAN : Asymmetrical occlusion handling using graph cut for multi-view stereo. Dans *Computer Vision and Pattern Recognition 2005 (CVPR 2005), IEEE Computer Society Conference on*, vol. 2, pages 902–909, June 2005.

[62] O.J. WOODFORD et A.W. FITZGIBBON : Fast image-based rendering using hierarchical image-based priors. Dans *British Machine Vision Conference 2005*, pages 260–269, 2005.

[63] O.J. WOODFORD, I.D. REID et A.W. FITZGIBBON : Efficient new-view synthesis using pairwise dictionary priors. Dans *Computer Vision and Pattern Recognition, 2007 (CVPR 2007), IEEE Conference on*, pages 1–8, juin 2007.

[64] O.J. WOODFORD, I.D. REID, P.H.S. TORR et A.W. FITZGIBBON : On new view synthesis using multiview stereo. Dans *British Machine Vision Conference 2007*, pages 1120–1129, 2007.

[65] H. YAMANOUE : The differences between toed-in camera configurations and parallel camera configurations in shooting stereoscopic images. Dans *2006 IEEE International Conference on Multimedia and Expo*, pages 1701–1704. IEEE, juillet 2006.

[66] H. YAMANOUE, M. NAGAYAMA, M. BITOU, J. TANADA, T. MOTOKI, T. MITUHASHI et M. HATORI : Tolerance for geometrical distortions between L/R images in 3D-HDTV. *Systems and Computers in Japan*, 29(5):37–48, mai 1998.

[67] H. YAMANOUE, M. OKUI et F. OKANO : Geometrical analysis of puppet-theater

and cardboard effects in stereoscopic HDTV images. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(6):744–752, juin 2006.

[68] H. YAMANOUE, M. OKUI et I. YUYAMA : A study on the relationship between shooting conditions and cardboard effect of stereoscopic images. *IEEE Transactions on Circuits and Systems for Video Technology*, 10(3):411–416, avril 2000.

[69] Q. YANG, L. WANG, R. YANG, H. STEWÉNIUS et D. NISTÉR : Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31 (3):492–504, mars 2009.

[70] J.S. YEDIDIA, W.T. FREEMAN et Y. WEISS : Understanding belief propagation and its generalizations. Rapport technique, Mitsubishi Electric Research Laboratories, Cambridge, Massachusetts, 2002.

[71] G. ZHANG, J. JIA, T.-T. WONG et H. BAO : Consistent depth maps recovery from a video sequence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(6):974–988, juin 2009.

[72] L. ZHANG, D. WANG et A. VINCENT : Adaptive reconstruction of intermediate views from stereoscopic images. *Circuits and Systems for Video Technology, IEEE Transactions on*, 16(1):102–113, janvier 2006.

[73] C.L. ZITNICK et T. KANADE : A cooperative algorithm for stereo matching and occlusion detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(7):675–684, juillet 2000.

[74] C.L. ZITNICK et S.B. KANG : Stereo for image-based rendering using image over-

segmentation. *International Journal of Computer Vision*, 75(1):49–65, octobre 2007.

[75] C.L. ZITNICK, S.B. KANG, M. UYTTENDAELE, S. WINDER et R. SZELISKI : High-quality video view interpolation using a layered representation. *ACM Transactions on Graphics*, 23(3):600–608, août 2004.