Vidéo 3D : quel débit pour la profondeur ?

Emilie Bosc, Vincent Jantet, Luce Morin, Muriel Pressigout, Christine Guillemot

▶ **To cite this version:**

Emilie Bosc, Vincent Jantet, Luce Morin, Muriel Pressigout, Christine Guillemot. Vidéo 3D : quel débit pour la profondeur ?. CORESA, Oct 2010, Lyon, France. hal-00533843

HAL Id: hal-00533843
https://hal.science/hal-00533843

Submitted on 8 Nov 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Vidéo 3D : quel débit pour la profondeur ?

Emilie Bosc[1] Vincent Jantet[2] Luce Morin[1] Muriel Pressigout[1] Christine Guillemot[2]

[1] IETR - INSA Rennes - 20 avenue des Buttes de Coësmes - 35043 Rennes, France
[2] INRIA Rennes, Bretagne Atlantique - Campus de Beaulieu - 35042 Rennes, France
[1]{Emilie.Bosc, Luce.Morin, Muriel.Pressigout}@insa-rennes.fr
[2]{Vincent.Jantet, Christine.Guillemot}@irisa.fr

Résumé

Cet article s'intéresse à la répartition du débit entre la texture et la profondeur lors de la compression de séquences multi-vues plus profondeur (MVD). Les effets de la quantification sur les deux types de données sont étudiés. La distorsion est mesurée sur les images synthétisées à partir des séquences MVD encodées et décodées par la méthode MVC. Bien que la profondeur soit codée sur une seule composante (contre trois pour la texture), allouer 25% du débit à la profondeur n'est pas le choix optimal. Les résultats montrent que plus de la moitié du débit doit être réservée aux données de profondeur.

Mots clefs

télévision 3D, vidéo multi-vues, synthèse de vue, compression, MVC, allocation de débit, cartes de profondeur

1 Introduction

Les vidéos 3D sont considérées comme l'évolution de la télévision conventionnelle actuelle. Le changement radical attendu est comparé à celui qu'occasionna l'introduction de la couleur à la télévision. Aujourd'hui, l'innovation majeure vient de l'apport de l'impression de profondeur générée par l'exploitation du phénomène de stéréopsie (perception de la profondeur de champ relative de deux stimuli présentés dans le champ visuel).

Ainsi, les représentations telles que les vidéos multi-vues (*MultiView Video* en anglais, MVV) permettent la création de vidéos 3D. Il s'agit de plusieurs séquences vidéo conventionnelles prises avec plusieurs caméras synchronisées et à des positions différentes dans la scène. Lorsque l'on associe ces vidéos à des vidéos dites de profondeur on parle de données *Multiview Video-plus Depth*, MVD. La Figure 1 illustre ce type de données constitué de séquences en couleur et de séquences de profondeur.

FIGURE 1 – *Les données de type MVD comportent des séquences de texture (à gauche) et de profondeur (à droite)*

La connaissance de la géométrie de la scène (issue des vidéos de profondeur) facilite la génération d'images virtuelles selon des points de vue différents de ceux réellement acquis par les caméras. Ces représentations de données permettent des applications telles que la télévision tridimensionnelle (3DTV) et le libre choix du point de vue (*Free viewpoint TeleVision*, FTV). 3DTV donne une impression de profondeur ou de relief à la scène alors que FTV offre à l'utilisateur la possibilité de choisir interactivement un point de vue arbitraire.
Les données traitées sont de taille considérable et une méthode efficace de compression est indispensable. De nombreuses solutions ont été proposées pour réaliser l'encodage des données de type MVD. Une problématique récurrente reste l'évaluation de l'allocation de débit qui optimise la qualité visuelle des images reconstruites à partir des données de texture et de profondeur.
Nous présentons dans ce document les études existantes ainsi que nos propres expériences réalisées pour répondre à ce problème. La section 2 revient sur les principes de synthèse de vue basée sur les cartes de profondeur. La section 3 rappelle les stratégies de compression de données MVD couramment utilisées. Enfin la section 4 montre les résultats des expériences réalisées pour déterminer le budget binaire optimal à attribuer aux vidéos de profondeur pour assurer des synthèses de vue de bonne qualité.

2 Synthèse de vue

La représentation MVD des vidéos 3D consiste en plusieurs séquences vidéo standard dites "vidéos de texture", et les vidéos de profondeur associées. Les cartes de pro-

fondeur (ou images de profondeur) qui composent les vidéos de profondeur sont des représentations bidimensionnelles de la scène 3D saisie. Ce sont des images en niveaux de gris, et de mêmes dimensions que les images de texture. Chaque pixel de la carte de profondeur indique la valeur de la profondeur du pixel de texture correspondant. Les valeurs de profondeur sont comprises dans un champ $[z_{near} - z_{far}]$, quantifié sur 8 bits. z_{near} est le point le plus proche de la caméra, et correspond à la valeur 255. z_{far} est le point le plus éloigné de la caméra, et correspond à la valeur 0.
Grâce à la géométrie projective [1], une carte de profondeur peut être convertie en une représentation tridimensionnelle. L'algorithme que nous utilisons pour cette opération de synthèse de vue à partir de cartes de profondeur est l'algorithme VSRS (View Synthesis Reference Software [2], version 3.5) fourni par MPEG. La Figure 2 illustre la relation entre un point 3D réel de la scène X et un point x_1 dans l'image de la caméra C_1 et son correspondant x_2 dans l'image de la caméra C_2. Cette transformation géométrique est possible grâce à la connaissance de la profondeur des pixels et des paramètres intrinsèques et extrinsèques des caméras. Ces dernières informations sont estimées lors du calibrage des caméras.

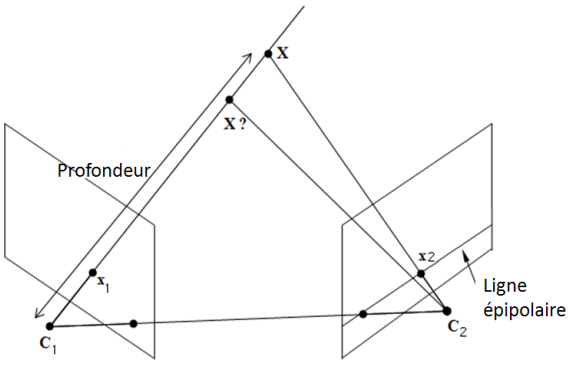


FIGURE 2 – *Relation entre les points de l'image et le monde réel [3]*

Selon le même principe, on peut projeter des points du monde réel sur le plan image d'une caméra virtuelle ayant un point de vue quelconque. Dans ce cas, l'algorithme VSRS nécessite les paramètres C_v de la caméra virtuelle. A partir des textures des deux vues adjacentes et des cartes de profondeur associées, l'algorithme estime la vue rendue selon le point de vue intermédiaire désiré.
La Figure 3 illustre le procédé utilisé par VSRS. Les deux cartes de profondeur sont projetées dans le point de vue intermédiaire. Il en résulte deux nouvelles cartes de profondeur correspondant au point de vue désiré. Ces cartes comportent des zones sans valeurs : ce sont les zones pour lesquelles on ne dispose pas d'informations dans la vue de référence (gauche, ou respectivement droite). Les zones manquantes de petites tailles peuvent être comblées par l'application d'un filtre médian. Les images de texture de référence sont ensuite elles-mêmes projetées dans le point de vue désiré, en fonction des profondeurs précédemment calculées. Il en résulte deux images de texture pour le même point de vue virtuel. Les vastes zones manquantes sont comblées en y attribuant les informations disponibles dans la deuxième image de texture générée, et vice versa. Enfin, ces deux images sont fusionnées, et on applique une méthode d'inpainting pour combler les dernières zones manquantes.

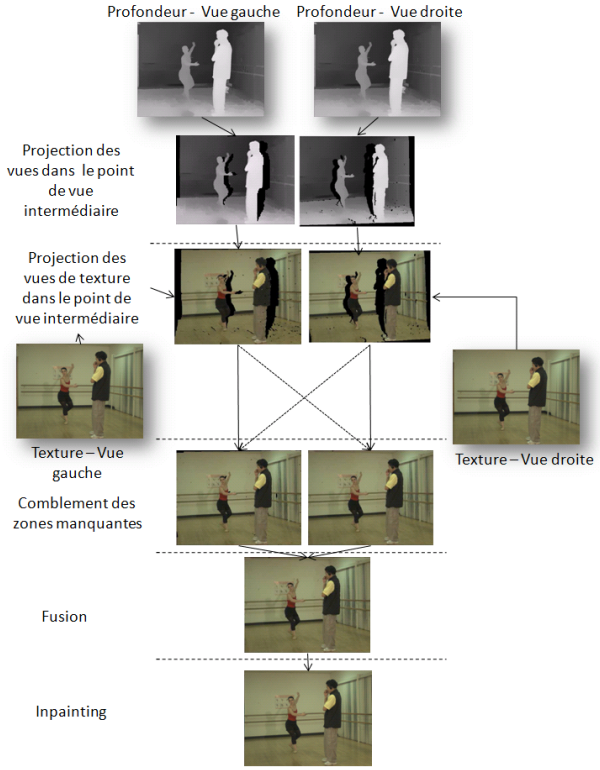


FIGURE 3 – *Synthèse d'une image intermédiaire avec VSRS*

Lors de la construction, expliquée précédemment, les valeurs de texture T_v de la vue virtuelle rendue sont attribuées aux pixels selon leur profondeur. Il y a bien trois cas possibles :
– Les profondeurs des pixels des vues adjacentes sont nulles, ce qui correspond à une zone occultée.
– Seule une des valeurs de profondeur est non-nulle, qui correspond à des régions occultées selon la nouvelle perspective.
– Les valeurs de profondeur des deux pixels sont non-nulles.
Ceci se traduit par l'expression suivante :

$$T_v = \begin{cases} 0, & \text{si } (u,v) \text{ n'est pas visible} \\ T_1(u,v), & \text{si } d_1(u,v) \neq 0 \\ & \text{et } d_2(u,v) = 0 \\ T_2(u,v), & \text{si } d_1(u,v) = 0 \\ & \text{et } d_2(u,v) \neq 0 \\ (1-\alpha)T_1(u,v) + \alpha T_2(u,v), & \\ & \text{si } d_1(u,v) \neq 0 \\ & \text{et } d_2(u,v) \neq 0 \end{cases}$$

où (u, v) fait référence aux coordonnées d'un pixel de l'image synthétisée, $d_1(u, v)$ la profondeur de ce pixel par rapport à la caméra C_1, $d_2(u, v)$ la profondeur de ce pixel par rapport à la caméra C_2, et α un facteur de pondération dépendant de la distance ($\alpha < 1$). A ce stade certaines causes d'artefacts peuvent être identifiées. Elles sont d'ordre géométrique : les régions occultées entraînent des zones dont la texture est nulle sur l'image rendue. Elles peuvent être dues aux arrondis des coordonnées des pixels calculées lors de la projection du monde réel vers le plan de l'image. L'inpainting fait partie des techniques de post-traitement inclues dans l'algorithme utilisé, qui permettent de pallier ce type d'artefacts.

3 Compression de vidéos multi-vues plus profondeur

Cette section aborde la compression de données de type MVV dans un premier temps, puis celle de données de type MVD. Enfin, les problématiques d'allocation de débit sont présentées.

3.1 Compression multi-vues (H.264/MVC)

L'idée la plus simple pour l'encodage des vidéos multi-vues consiste à encoder indépendamment chaque vue en utilisant un codeur de l'état de l'art. On appelle cette méthode *simulcast coding*. Ce procédé conduit à négliger l'exploitation des redondances existantes entre chaque vue, or ces redondances permettent de réaliser un gain significatif. Ainsi, en vue de surpasser les performances du codage simulcast, des modes de prédiction inter-vues sont ajoutés aux modes de prédiction spatiale et temporelle existant dans les codeurs simples de l'état de l'art. La compression H.264/AVC est considérée comme la meilleure méthode de compression pour les vidéos simples. Le codage multi-vues (*Multiview Video Coding* [4] en anglais) est basé sur cette méthode et a donné naissance au standard H.264/MVC [5]. La compression H.264/AVC [6, 7] exploitant déjà les redondances spatiales et temporelles au sein d'une même vue, l'algorithme MVC y ajoute l'utilisation des corrélations entre vues adjacentes.

La structure du codeur MVC consiste en une matrice de codeurs simples qui utilisent les redondances spatiales et temporelles. Les structures de bases des méthodes de codage sont conservées. Il s'agit des images codées *intra* (I) qui font appel aux pixels voisins à l'intérieur de la même image, des images *prédictives* (P) et des images *bidirectionnelles* (B) qui font référence aux pixels d'images voisines précédemment encodées ou à encoder.

L'originalité de la compression H.264 vient du fait qu'elle utilise des prédictions hiérarchiques, dont l'efficacité est prouvée [8]. Il existe des niveaux de prédictions, certaines images B pouvant devenir des références pour d'autres images B.

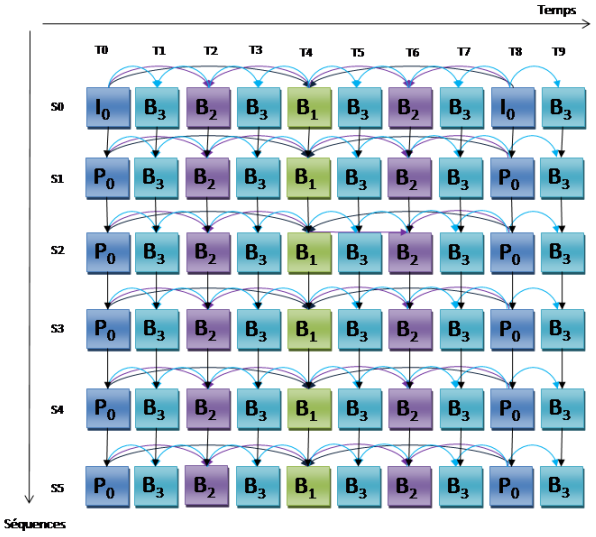


FIGURE 4 – *Schéma de la structure de la méthode H.264/MVC*

La prédiction inter-vues utilise la compensation en mouvement. La Figure 4 montre que les images encodées à un niveau de prédiction (et appartenant à une certaine vue) peuvent contribuer à la prédiction des images des vues adjacentes.

3.2 Compression de données MVD

Les séquences de profondeur peuvent être codées de façon particulière, en tenant compte de leurs caractéristiques [9], ou en réutilisant le codeur MVC, comme le proposent de nombreux travaux [10, 11, 12]. Lors d'une réutilisation du codeur MVC, rien n'impose d'utiliser les mêmes jeux de paramètres pour les cartes de profondeurs que pour les textures. Pour une utilisation judicieuse des canaux de transmission, restreints en bande passante, il convient de définir la distribution du budget binaire entre les séquences de texture et celles de profondeur.

En effet, compte tenu des propriétés des vidéos de profondeur, bien que ces dernières puissent être traitées comme des signaux monochromes et comprimées par les méthodes de l'état de l'art, il convient de leur appliquer un taux de compression adapté. Les codeurs de l'état de l'art sont souvent optimisés pour respecter des propriétés statistiques ou bien pour respecter la perception humaine des couleurs. Cependant, comme expliqué précédemment, les cartes de profondeur sont foncièrement différentes des images de texture, ce ne sont pas des images réelles. Par conséquent les artéfacts liés à la compression des cartes de profondeur n'impliquent pas les mêmes gênes que celles occasionnées par la compression d'images réelles. Pour cette raison, il convient de bien évaluer la quantification limite à appliquer aux vidéos de profondeur pour une qualité de synthèse optimale.

La synthèse de vue, décrite dans la section précédente, implique déjà des artefacts de projection à cause des zones

d'occultation et des imprécisions possibles de l'estimation de la carte de profondeur : certains pixels de la vue virtuelle peuvent ne pas trouver de point correspondant dans les vues de référence adjacentes. Ils nécessiteront une opération d'inpainting (interpolation par rapport aux valeurs des pixels voisins). La compression ajoute à cela de nouveaux artefacts de projection dans la vue rendue. Des expériences, menées dans [10], consistent à faire varier la compression des séquences de profondeur et à en observer l'impact sur les vues de synthèse rendues (existantes ou virtuelles). Les comparaisons sont basées sur des mesures de PSNR.

Les résultats confirment l'influence sérieuse de la qualité de l'information de la profondeur sur la qualité de la vue intermédiaire rendue. Les taux de compression élevés de la profondeur conduisent à des distorsions importantes au niveau des discontinuités, sur les bords des objets. En revanche, les auteurs constatent que la variation de la qualité de la texture influence moins la qualité de la vue rendue. Pourtant, des mesures objectives et subjectives menées dans [13] indiquent que les observateurs tolèrent plus facilement les artefacts de profondeur que ceux de texture, ce qui suggère que l'on peut favoriser une compression plus importante pour les informations de profondeur.

3.3 Allocation texture/profondeur

Dans la plupart des méthodes d'encodage, le débit affecté aux séquences de profondeur est restreint, et fixé à un certain pourcentage du débit total (généralement entre 10% et 15%). Dans [14], les auteurs estiment que seulement 5% à 10% du débit total suffisent pour l'encodage des séquences de profondeur avec la méthode H.264/AVC.

Dans les travaux présentés dans [15], le budget est fixe mais les images de profondeur sont sous-échantillonnées. Cette approche est basée sur le fait que sous-échantillonner une image vers une résolution moindre, puis l'encoder et l'agrandir à la résolution initiale améliore les performances en termes de PSNR. Il s'agit de réaliser un compromis entre la distorsion introduite par le sous-échantillonnage et la distorsion introduite par la quantification. A budget binaire fixé, le sous-échantillonnage permet une quantification moins brutale. Les résultats montrent un gain de 33% du débit total.

Face à l'approche consistant à attribuer un budget fixe, il existe également des méthodes d'optimisation de l'allocation du débit. Elles se basent sur des critères de qualité de l'information de la profondeur, ou bien de qualité de l'image de synthèse. C'est le cas dans [16], où le rapport optimal entre le débit et la distorsion est calculé en sélectionnant le groupe de paramètres de quantification qui favorise la meilleure qualité de vue rendue. Il y est proposé un algorithme d'optimisation de la répartition du débit par la recherche hiérarchique des paramètres de quantification idéaux. Pour un débit total de 75kbits, un gain de 1dB est enregistré sur la qualité de l'image rendue (en PSNR), comparé à une méthode d'allocation fixe (10% pour la profondeur).

Des études ont cherché à évaluer quantitativement la façon dont la compression de la texture et de la profondeur affecte la qualité des images rendues, en utilisant la méthode de compression H.264 comme dans [17]. Les auteurs proposent un modèle mathématique de la distorsion maximum possible. Cette dernière dépend de l'erreur induite par la compression de la profondeur et de l'erreur induite par la compression de la texture. Les résultats de ces expériences montrent que le modèle présenté est une indication approximative de la qualité de la vue virtuelle rendue et peut constituer un outil pour optimiser le rapport entre le débit et la distorsion due à la compression des deux types de séquences. La méthode d'optimisation d'allocation proposée obtient une amélioration de 0.3 à 1dB sur la qualité de l'image de synthèse comparée à une méthode d'allocation constante de débit (20% du débit total alloué à la profondeur).

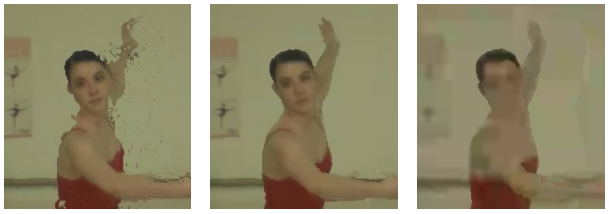
D'autres travaux ont mis en évidence les effets de la compression de la profondeur sur le rendu de vues [18] : comparé à une méthode de compression basée *platelet*, H.264/MVC obtient de meilleurs résultats en termes de PSNR. En revanche, la méthode basée platelet, adaptée aux caractéristiques de la carte de profondeur, préserve mieux les contours et donc la géométrie de la scène, d'où une meilleure qualité visuelle de l'image rendue.

La distribution optimale du débit entre les séquences de texture et de profondeur dépend fortement de l'utilisation qui sera faite des vidéos. Dans la section suivante, nous présentons une série d'expérimentations permettant d'évaluer le ratio optimal entre les débits de la texture et de la profondeur, pour une utilisation de la séquence MVD par VSRS.

4 Simulation et résultats expérimentaux

Nous avons souhaité évaluer la qualité d'images compressées par l'algorithme MVC en faisant varier la répartition du débit entre la texture et la profondeur [19]. Nous avons testé différents paramètres de quantification pour les textures et les cartes de profondeurs, afin d'observer la qualité du rendu obtenu par l'algorithme VSRS (version 3.5). Comme expliqué précédemment, VSRS est un algorithme développé par MPEG qui permet de synthétiser une vue intermédiaire à partir de deux vues adjacentes (textures) et des cartes de profondeur associées.

Allouer des débits différents à la texture et à la profondeur a un impact important sur la qualité finale d'une vue synthétisée. Pour un débit total fixé à 13 Mbit/s, la Figure 5 montre quelques images synthétisées à partir de vidéo multi-vues ("*Ballet*" de Microsoft Research), mais avec des proportions Texture/Profondeur différentes. Elles ont été construites avec VSRS, à partir de textures et de profondeur préalablement encodées par l'algorithme MVC.



(a) PSNR = 30.0dB ; Prof. = 3% débit ; (b) PSNR = 33.8dB ; Prof. = 60% débit ; (c) PSNR = 30.8dB ; Prof. = 95% débit ;

FIGURE 5 – *Images synthétisées à partir de séquences MVD à 13 Mbit/s, et pour différents rapports de débit entre texture et profondeur.*

– Avec 3% du débit alloué à la profondeur, la Figure 5(a) montre une importante dégradation des contours. Une trop forte compression de la carte de profondeur induit une perte de précision sur la position des discontinuités, provoquant des erreurs lors de la projection. Même si une grande partie de l'arrière plan est correct, le PSNR chute à cause des erreurs importantes commises sur les pixels de contours.
– Avec 95% du débit alloué à la profondeur, la Figure 5(c) montre des contours nets, mais des textures floues. La quantification trop forte des textures a entraîné une perte d'information importante, et une baisse de la qualité visuelle. Le PSNR chute également, à cause des erreurs commises sur tous les pixels de l'image.
– Avec 60% du débit alloué à la profondeur, la Figure 5(b) montre une qualité visuelle bien supérieure aux exemples précédents. Les cartes de profondeur sont suffisamment précises pour permettre les calculs de projections, et les textures sont suffisamment détaillées pour éviter des artéfacts de compression.

La qualité visuelle de ces images laisse penser qu'il existe un budget binaire optimal à attribuer à la profondeur.

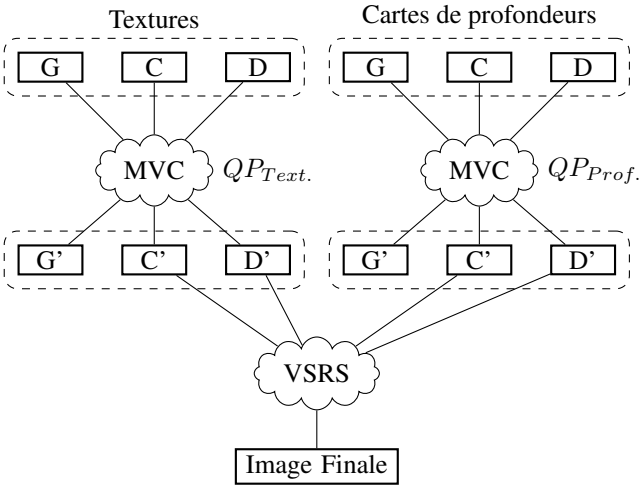


FIGURE 6 – *Représentation schématique du processus expérimental.*

La Figure 6 présente le protocole expérimental qui a permis d'évaluer le rapport de débit idéal entre la texture et la profondeur. Ce protocole se scinde en deux étapes distinctes :

– Dans un premier temps, les premières images des vues 2, 4 et 6 de la vidéo "*Ballet*" de Microsoft Research ont été sélectionnées. Sur le schéma 6, elles correspondent aux notations D pour droite, C pour centre, et G pour gauche. MVC a été utilisé pour compresser et décompresser ces textures d'une part, et les cartes de profondeur associées d'autre part. Notons $QP_{Text.}$, le pas de quantification des séquences de texture, et $QP_{Prof.}$ le pas de quantification des séquences de profondeur. Dans les deux cas, la vue 4 est la vue de référence, utilisée pour prédire les vues 2 et 6. Le débit de la séquence, calculé comme la somme des débits de la texture et de la profondeur, est directement lié aux pas de quantifications choisis.
– Dans un second temps, les vues décompressées 2 et 4 sont utilisées par le logiciel VSRS pour la synthèse d'une vue intermédiaire 3. Dans le cadre d'une utilisation en synthèse de vue, le PSNR entre cette vue générée et la vue 3 originale est le critère d'évaluation de la qualité de la vidéo multi-vues (nous disposons d'une vérité terrain).

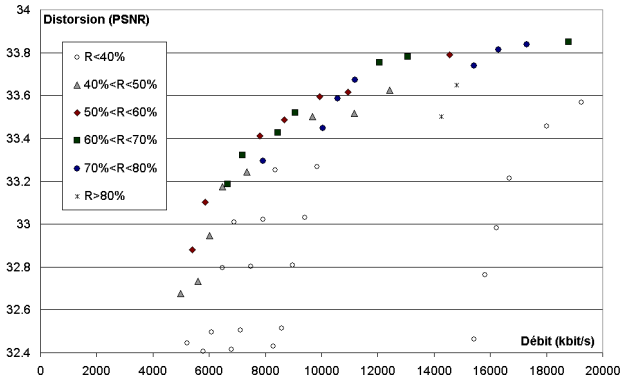


FIGURE 7 – *Courbe débit-distorsion pour différents rapports du débit profondeur sur le débit total.*

La Figure 7 rassemble les résultats de ce protocole expérimental, appliqué à un large choix de paramètres de quantification différents. Chaque point du graphique est représenté par un symbole, qui traduit le rapport R entre le débit alloué à la profondeur et le débit total. La position du point dans le graphique représente le PSNR (en dB) de la vue synthétisée en fonction du débit total. L'ensemble des points est délimité par une enveloppe convexe qui définit le rapport idéal entre le débit texture et profondeur.

La texture est une image à trois composantes, alors que la profondeur est représentée par une image à une seule composante. En conséquence, une hypothèse simple consisterait à penser que la profondeur nécessite moins de bande passante. Pourtant, les échantillons dont la profondeur représente moins de 40% du débit total sont très mal posi-

tionnés dans ce graphique. Au contraire, ce sont les échantillons de rapport R entre 50% et 70% qui approchent le mieux l'enveloppe. C'est ce rapport R voisin de 60% qui permet d'obtenir les meilleurs compromis entre le débit et la distorsion de l'image reconstituée. Ce rapport optimal n'est pas dépendant du débit total ciblé, ou de la distorsion maximale autorisée. Il apparait nécessaire d'allouer à la profondeur plus de la moitié, voir les deux tiers, du débit, pour assurer une bonne qualité de synthèse de vue.

5 Conclusion

A partir de vidéos MVD, nous avons utilisé le codeur MVC pour compresser les textures d'un coté, et les cartes de profondeur de l'autre. En faisant varier les pas de quantifications, on observe l'évolution du PSNR lors de la synthèse de vue par VSRS. La compression des textures fait apparaitre des zones floues, et la compression des cartes de profondeur entraîne des distorsions géométriques. A débit total fixé, il existe un rapport idéal entre le débit alloué à la texture et celui alloué à la profondeur, maximisant la qualité visuelle de l'image générée. Pour assurer la meilleure qualité d'image lors de la synthèse de vue avec VSRS, nous avons montré qu'il est nécessaire d'attribuer aux cartes de profondeur environ 60% du débit total.

Remerciements

Ces travaux ont été réalisés dans le cadre des projets ANR-PERSEE, ANR-CAIMAN et DGE-Région FUTURIM@GE. Nous souhaitons également remercier Microsoft Research pour la mise à disposition de la séquence multi-vues "*Ballet*".

Références

[1] R. Hartley et A. Zisserman. *Multiple view geometry in computer vision*. Cambridge Univ Pr, 2003.

[2] M. Tanimoto, T. Fujii, K. Suzuki, N. Fukushima, et Y. Mori. Reference softwares for depth estimation and view synthesis. Avril 2008.

[3] C. Lee et Y. S Ho. View synthesis tools for 3D Video.ISO/IEC JTC1/SC29/WG11 MPEG2008/M15851, Octobre 2008.

[4] Y. Chen, Y. K Wang, K. Ugur, M. M Hannuksela, J. Lainema, et M. Gabbouj. The emerging MVC standard for 3D video services. *EURASIP Journal on Advances in Signal Processing*, 2009(1), 2009.

[5] ISO/IEC JTC1/SC29/WG11,. Text of ISO/IEC 14496-10 :200X/FDAM 1 multiview video coding, 2008.

[6] ITU-T Recommendation H.264. Advanced video coding for generic Audio-Visual services, 2009.

[7] T. Wiegand, G. J Sullivan, G. Bjontegaard, et A. Luthra. Overview of the h. 264/AVC video coding standard. *IEEE Transactions on circuits and systems for video technology*, 13(7) :560–576, 2003.

[8] H. Schwarz, D. Marpe, et T. Wiegand. Analysis of hierarchical b pictures and MCTF. Dans *Proc. ICME*, page 1929–1932, 2006.

[9] Y. Morvan, D. Farin, et P. H.N De. Depth-Image compression based on an RD optimized quadtree decomposition for the transmission of multiview images. Dans *IEEE Int. July*, page 105–108, 2007.

[10] P. Merkle, A. Smolic, K. Muller, et T. Wiegand. Multi-view video plus depth representation and coding. Dans *Proceedings of ICIP*, page 201–204, 2007.

[11] P. Merkle, K. Muller, A. Smolic, et T. Wiegand. Efficient compression of multi-view video exploiting inter-view dependencies based on h. 264/MPEG4-AVC. Dans *Proc. ICME*, page 9–12, 2006.

[12] I. Daribo, M. Kaaniche, W. Miled, M. Cagnazzo, et B. Pesquet-Popescu. Dense disparity estimation in multiview video coding. Dans *Proc. of the IEEE Workshop on Multimedia Signal Processing (MMSP), Rio de Janeiro, Brazil*, 2009.

[13] A. Tikanmaki, A. Gotchev, A. Smolic, et K. M\üller. Quality assessment of 3D video in rate allocation experiments. Dans *IEEE Int. Symposium on Consumer Electronics (14-16 April, Algarve, Portugal)*, 2008.

[14] E. Martinian, A. Behrens, J. Xin, et A. Vetro. View synthesis for multiview video compression. Dans *Picture Coding Symposium*, volume 37, page 38–39, 2006.

[15] E. Ekmekcioglu, S. T Worrall, et A. M Kondoz. Bitrate adaptive down-sampling for the coding of multiview video with depth information. Dans *3DTV Conference : The True Vision-Capture, Transmission and Display of 3D Video, 2008*, page 137–140, 2008.

[16] Y. Morvan, D. Farin, et P.H.N. de With. Joint depth/texture bit-allocation for multi-view video compression. *Picture Coding Symposium*, 10(1.66) :4349, 2008.

[17] Y. Liu, Q. Huang, S. Ma, D. Zhao, et W. Gao. Joint video/depth rate allocation for 3D video coding based on view synthesis distortion model. *Signal Processing : Image Communication*, 24(8) :666–681, Septembre 2009.

[18] P. Merkle, Y. Morvan, A. Smolic, D. Farin, K. Muller, et T. Wiegand. The effects of multiview depth video compression on multiview rendering. *Singal Processing : Image Communication*, 24(1-2) :73–88, 2009.

[19] K. Muller, A. Smolic, K. Dix, P. Merkle, et T. Wiegand. Coding and intermediate view synthesis of multiview video plus depth. pages 741–744, Novembre 2009.