

Composition et interpolation de points de vue pour un rendu 3D cubiste

Benjamin Raynal, Venceslas Biri et Jacques Poulain

Université Paris-Est
Laboratoire d'Informatique de l'Institut Gaspard Monge
UMR CNRS 8049
5 bd Descartes, 77454 Marne la Vallée Cedex 2, France
{raynal,biri}@univ-mlv.fr

Abstract

Nous présentons dans cet article une méthode de rendu non photoréaliste simulant le style cubiste, en utilisant un lancer de rayon multiperspective. Nous proposons deux méthodes de composition de l'image finale à partir des différents points de vue, ainsi que des réponses aux différents problèmes de l'interpolation entre ces vues.

Keywords: Rendu non photoréaliste, cubisme, lancer de rayon, vue multiperspective

1. Introduction

Il existe en traitement d'images nombre d'algorithmes permettant d'obtenir des rendus ressemblant à différents styles de dessin ou de peinture (style crayonné, encre, aquarelle, impressionnisme, pour ne citer qu'eux). Cependant, toutes ces techniques ont en commun de pouvoir être obtenues à partir d'une simple image en appliquant des procédés très locaux, et sont peu sensibles au placement spatial des éléments de la scène représentée. Depuis le début des années 2000, un certain nombre d'études ont été menées sur le rendu multiperspective, permettant d'effectuer des rendus non photoréalistes prenant en considération la disposition spatiale de la scène. Le travail présenté dans cet article s'inspire des travaux déjà effectués sur le rendu multiperspective et nous proposons ici une nouvelle méthode de rendu permettant d'obtenir un rendu de style cubiste.

1.1. Cubisme

Le cubisme est un mouvement artistique qui s'est développé de 1907 à 1914 à l'initiative de Braque et de Picasso. Le concept du cubisme est de représenter le volume des objets. Le mouvement peut se découper en 3 périodes :

Précubisme Aussi appelé cubisme césannien. Durant cette période, des peintres tels que Picasso et Braque commencent à représenter un même sujet à partir d'un point de vue multiple.

Cubisme analytique Les tableaux sont décomposés en facettes géométriques, représentant les différentes faces du

sujet. La décomposition du sujet est telle que l'on tend vers l'abstrait le plus complet.

Cubisme synthétique Cette période se caractérise par un retour vers le réel, avec l'introduction de collages, de journaux et d'objets divers (toile cirée, par exemple). Les tableaux sont toujours décomposés en facettes, mais plus larges, et ciblant les faces les plus caractéristiques de l'objet, permettant une meilleure reconnaissance du sujet.

Par la suite, divers peintres s'inspirent du cubisme, et représentent des sujets à partir d'un point de vue multiple. C'est le cas de Salvador Dali, par exemple, qui se démarque surtout en intégrant une continuité entre les points de vue.

2. Etat de l'art

2.1. Rendu cubiste

Différents travaux ont déjà été effectués dans le cadre du rendu cubiste, que l'on peut classer en trois catégories :

Approche 2D L'idée ici est de s'appuyer sur différentes images du sujet pour le rendu final. Dans [CH03], Collo-mosse et al. utilisent des séquences d'images d'un même sujet, et sélectionnent dans chaque image les facettes les plus intéressantes en mesurant la saillance des contours, les déformant ensuite avant de les utiliser pour recomposer une image finale. Aucune continuité n'est assurée entre les différentes facettes.

Approche 2D + temps Le matériau de base de ces méthodes est la vidéo. L'idée générale est de considérer le temps comme une troisième dimension. La superposition spatiale des frames consécutives nous donne alors une *cube vidéo* (voir les travaux de Fels et al. [FLM00,FM99], ainsi

que ceux de Klein et al. [KSC*01]). L'image finale est obtenue en faisant une coupe de ce cube vidéo, non parallèle au plan des frames, bien évidemment.

Approche 3D Cette approche est la plus étudiée. Elle consiste à utiliser une scène 3D et à jouer directement sur les différents points de vues (rendu multiperspective), représentés par des caméras. Nous détaillons les travaux déjà proposés dans la sous section 2.2.

2.2. Rendu multiperspective

On peut distinguer deux familles de méthodes de rendu multiperspective : celles n'utilisant qu'une seule caméra, dont la définition est plus complexe qu'une simple caméra, et celles utilisant un ensemble de caméras, entre lesquelles sont faites des interpolations.

2.2.1. Méthodes à caméra unique

Les méthodes de rendu multiperspective à caméra unique partent toutes de la même idée de base : définir la caméra de façon à ce qu'elle ait un champ de vision différent de celui des caméras habituelles (pinhole ou isométrique, voir figure 1).

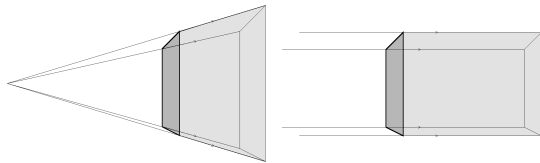


Figure 1: A gauche, une caméra pinhole, à droite, une caméra isométrique. Le plan de la caméra est indiqué en gris foncé, son champ de vision, en gris clair.

Meadows et al. [MA00] proposent d'altérer la vue en modifiant la position de la caméra en fonction du pixel à calculer. Pour cela, l'utilisateur doit fournir des *images de contrôle de caméra*, de mêmes dimensions que l'image à calculer, ou les composantes couleur correspondent aux valeurs des décalages à effectuer. Il en découle qu'il revient à l'utilisateur la charge de s'assurer de la continuité de la vue.

Une méthode plus utilisée, se basant sur le lancer de rayon (voir la sous-section 3.2 pour plus de détails) consiste à définir une surface de projection non planaire, en faisant correspondre chaque pixel à un point de sa surface, et dont le rayon associé correspond à la normale de ce point. Les surfaces peuvent être décrites grâce à divers outils mathématiques : Spindler et al. [SRMS06] utilisent des NURBS, Vallance et al. [VC01], des B-splines, et Loffelmann et al. [LG96] des surfaces décrites par leur équation cartésienne (tore, super-ellipsoïdes, et autres).

Une autre alternative est proposée par Brosz et al. [BSSS07], et consiste à modéliser directement le volume du champ de vision, en décrivant celui-ci à partir de plusieurs surfaces.

2.2.2. Méthodes multi-caméras

Les méthodes de rendu multiperspective utilisant plusieurs caméras doivent répondre à deux principaux problèmes :

Composition Déterminer quelle vue est utilisée dans quelle partie du rendu final.

Interpolation Préserver, autant que possible, la continuité entre les différentes parties du rendu final.

Pour répondre au premier problème, l'approche généralement utilisée est d'en laisser la charge complète à l'utilisateur. L'interpolation, par contre, est abordée de façons très variées.

Smith et al. [SADK04] proposent de répartir les caméras à intervalles réguliers sur une surface, appelée *surface "oeil"*. Les caméras sont toutes orientées vers des points répartis à intervalles réguliers sur une autre surface, appelée *surface "cible"*. Le rendu final est obtenu en juxtaposant les images obtenues par chaque caméra. Une alternative proposée est de laisser l'utilisateur transformer géométriquement les images et les superposer à sa guise. Ici, aucune interpolation n'est proposée, la continuité du rendu final n'est pas assurée.

La méthode de Glassner [Gla04a, Gla04b] demande à l'utilisateur d'attribuer des zones de l'image aux différentes vues (voir figure 2). Les zones non attribuées sont calculées par lancer d'un rayon obtenu par interpolation linéaire entre les rayons des zones attribuées adjacentes, en fonction de la distance minimale de ces zones au point considéré.

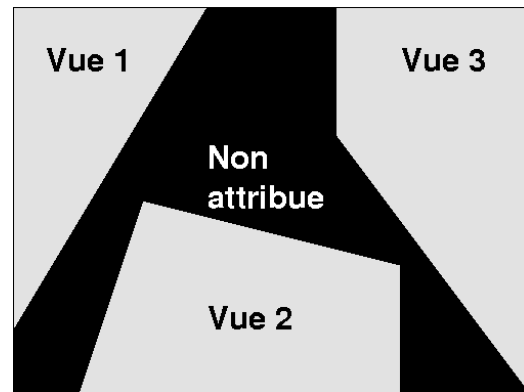


Figure 2: Exemple de composition de points de vue.

L'approche de Yu et al. [YM04a] comporte certaines similarités à celle de Glassner. Ici aussi, l'utilisateur a à sa

disposition les rendus des différentes vues, qu'il peut découper, transformer et positionner dans l'image selon son goût. Cependant, les interpolations sont gérées d'une façon très différente. Les auteurs utilisent un modèle plus générique des caméras : les *caméras linéaires générales* [YM04b] (en anglais, General Linear Camera, abrégé en *GLC*). Ce modèle permet de décrire aussi bien des caméras traditionnelles, comme les caméras pinholes où isométriques, que des caméras multiperspectives usuelles, comme les caméras push-broom, par exemple. Une GLC est définie par trois rayons générateurs. L'idée ici est de réaliser un pavage triangulaire de l'image finale, chaque sommet étant associé à un rayon générateur. Les sommets se trouvant "sous" un morceau positionné par l'utilisateur prendront ainsi les valeurs des rayons lancés pour obtenir les valeurs des pixels les recouvrant. Les autres sommets se feront alors attribuer, de proche en proche, des rayons générateurs, garantissant que chaque triangle corresponde à une caméra de type voulu. De cette façon, la continuité de l'image est garantie, du fait de la continuité entre deux caméras partageant 2 rayons générateurs.

2.3. Positionnement de notre travail

Pour une meilleure interprétation cubiste d'une scène, notre méthode de rendu utilise une approche multi-caméras. Cependant, contrairement aux méthodes décrites ci-dessus, notre ambition tend à limiter le besoin d'interaction avec l'utilisateur au minimum, c'est à dire uniquement pour le choix des points de vue. Pour cela, nous proposons des solutions aux problèmes mentionnés dans la sous section précédente. Ainsi, dans la section 4 nous traiterons du problème de la composition de points de vue, en proposant de nouvelles approches, et dans la section 5 nous proposerons de nouveaux outils, afin de gagner en fluidité et en facilité de paramétrisation de l'interpolation.

Il est important de noter qu'avant l'interpolation entre les vues, la composition de celles-ci, telle que nous les proposons, recouvrent toute la surface de l'image finale, ce qui n'est pas le cas des méthodes décrites dans l'état de l'art. Les compositions en elles même peuvent donc déjà être considérées comme des rendus cubistes à part entière.

3. Implémentation

3.1. Lancer de rayon

Pour implémenter notre rendu cubiste, nous avons choisi le lancer de rayon, et cela pour plusieurs raisons :

popularité du fait du grand nombre de logiciels de graphisme 3D utilisant le lancer de rayon, il sera plus aisé de porter nos travaux sur des scènes existantes de bonne qualité, par le biais de plugins, sur *blender* par exemple.

parallélisme chaque pixel de l'image étant calculé de façon totalement indépendante de ses voisins, on peut aisément

changer les paramètres de la caméra pour chacun, sans risque de perte de cohérence.

réalisme bien que le cubisme soit par définition non photoréaliste, il est constitué de composantes réalistes, répondant à des critères d'illumination, par exemple, difficiles à obtenir avec d'autres méthodes.

3.2. Chaîne de traitements

Le lancer de rayon peut être décomposé en plusieurs étapes. Pour chaque pixel de l'image :

1. déterminer le rayon à lancer (origine et vecteur direction).
2. calculer le premier solide de la scène rencontré.
3. calculer la couleur du point d'intersection en fonction des différentes variables de la scène (brouillard, lumières, et autres)

Notre travail se situe au niveau de la première étape. Si habituellement, tous les rayons trouvent leur origine sur le plan de la caméra, ici, nous allons la définir en fonction de plusieurs caméras, selon des règles que nous définissons dans les sections suivantes.

4. Composition de points de vue

Le premier point de travail a été de déterminer la façon de composer l'image à partir des différents points de vue. Pour un ensemble de caméras, dont les paramètres intrinsèques et extrinsèques sont définis préalablement (soit de façon arbitraire par l'utilisateur, soit automatiquement), nous avons défini deux méthodes de composition de l'image finale.

4.1. Composition par zones d'influence

Pour chaque caméra, on définit (par l'utilisateur ou de façon automatique) un ou plusieurs points de l'image qui lui sont affectés. Soient $\mathcal{C}$ l'ensemble des caméras définies, $\mathcal{I}$ l'ensemble des points de l'image, $\mathcal{P}$ l'ensemble des points de l'image affectés à une caméra, $\mathcal{P}_i$ l'ensemble des points de l'image affectés à la $i^{ème}$ caméra, $\mathcal{Z}_i$ les points de l'image appartenant à la zone d'influence de la $i^{ème}$ caméra, et $d : \mathcal{I} * \mathcal{I} \rightarrow \mathbb{R}$ une fonction distance. Les points de l'image appartenant à la zone d'influence d'une caméra sont définis de la façon suivante :

$$\mathcal{Z}_i = \{p \in \mathcal{I}; \forall c \in \mathcal{P} \setminus \mathcal{P}_i, \exists c' \in \mathcal{P}_i, d(p, c) > d(p, c')\}$$

L'image finale est obtenue en lançant chaque rayon à partir de la caméra correspondant à la zone d'influence contenant le pixel à calculer.

4.1.1. Placement automatique des points

Pour placer automatiquement les points affectés à chaque caméra, il est important de prendre en considération l'ensemble du processus de rendu multiperspective. En effet, la qualité de la répartition des points de vue affecte la qualité

de l'interpolation. Si deux vues très différentes sont positionnées trop proche l'une de l'autre, l'interpolation entre les vues aura un aspect très compact, du fait du faible échantillonnage des vues interpolées, de valeurs trop importantes.

Pour obtenir un rendu le plus fluide possible, il est donc nécessaire de placer les points affectés à chaque vues les uns par rapport aux autres en fonction de la ressemblance de leur point de vue. Cette ressemblance peut-être appréciée selon divers critères, telle que la différence d'orientation des caméras, la distance entre les origines des caméras, la distance entre les points d'intersection entre la scène et le rayon partant du centre de la caméra, ou encore une combinaison de ces critères.

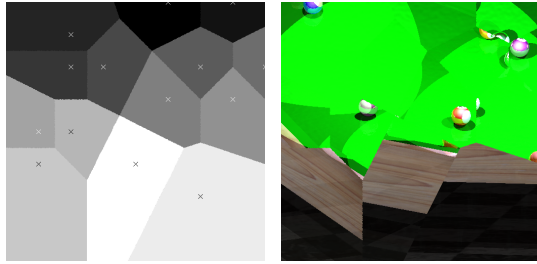


Figure 3: Exemple de composition par zone d'influence : à gauche, la carte des zones d'influence, chaque croix représente un point affecté à une vue. À droite, le résultat obtenu avec cette composition.

4.2. Composition par cartes de profondeurs

Nous proposons ici une méthode adaptée pour le rendu d'un sujet unique. Cette méthode utilise une approche sensiblement différente de la précédente : on considère que les caméras sont toutes disposées à une distance égale du centre du sujet, et orientées vers lui. Pour chaque vue, on procède à un lancer de rayon, très simple, afin de déterminer une carte de profondeurs $\mathcal{D}_i$, où chaque pixel p se voit attribuer la distance (notée $\mathcal{D}_i(p)$) que doit parcourir son rayon associé pour toucher l'objet. Ici, la zone d'influence de la $i^{\text{ème}}$ caméra est définie de la façon suivante :

$$\mathcal{L}_i = \{p \in \mathcal{I}; \mathcal{D}_i(p) \leq \mathcal{D}_v(p), 1 \leq v \leq \text{card}(\mathcal{C})\}$$

Cette méthode a pour principal intérêt, outre le fait de ne nécessiter aucune action de l'utilisateur, de faire ressortir les protubérances de l'objet, souvent primordiales dans sa reconnaissance visuelle.

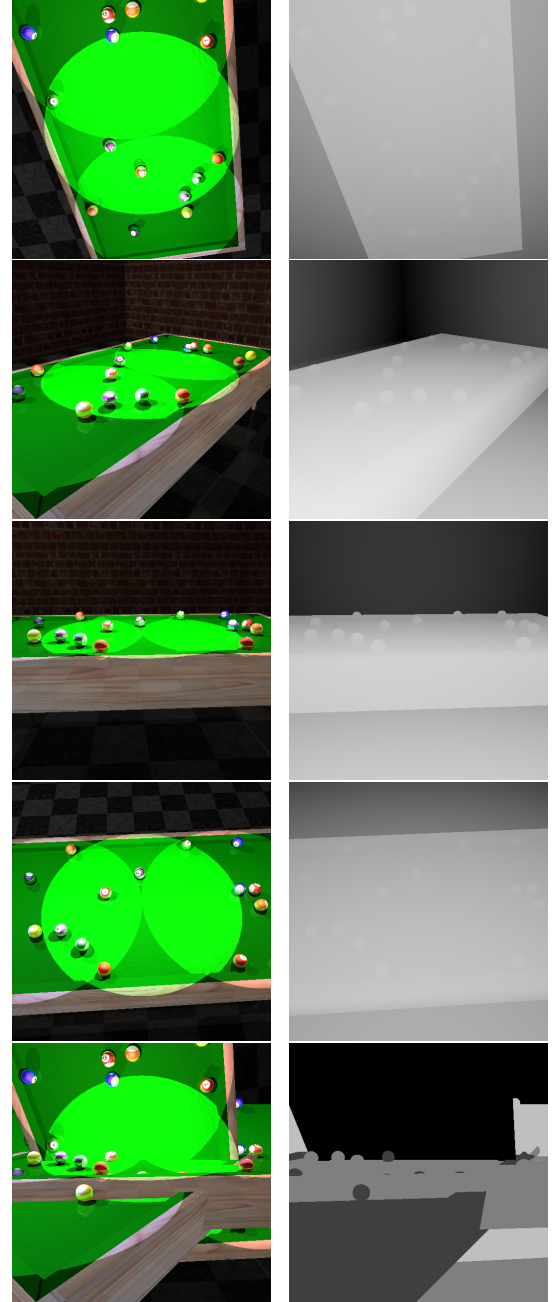


Figure 4: Sur les quatre premières lignes : points de vue de la scène, avec la carte de profondeur associée. Dernière ligne : le rendu obtenu, avec la carte de composition associée.

5. Interpolations

La deuxième partie de notre travail est portée sur l'interpolation entre les différents points de vue. Cette interpolation n'est pas obligatoire pour obtenir un rendu de type cu-

biste, mais permet d'ajouter une large variété de rendus possibles. Nous proposons donc un outil permettant une paramétrisation intuitive et facile de l'interpolation, fournissant des coefficients de pondération de chaque vue, ainsi qu'une alternative à l'interpolation sur les coordonnées cartésiennes des rayons.

5.1. Calcul des valeurs d'interpolation par convolution

Pour permettre une interpolation fluide entre les différentes facettes, il est nécessaire de pouvoir mesurer efficacement dans quelles proportions vont être utilisés les paramètres de chaque point de vue pour chaque pixel. L'outil le plus souvent utilisé est la pondération par rapport aux distances aux différents points de référence, mais cela ne permet pas de paramétrer aisément la largeur et la forme des zones d'interpolation. Pour répondre à ce besoin, nous avons développé une méthode, s'inspirant des différents calculs de flou, en se basant sur les convolutions. L'idée est d'appliquer un noyau de convolution sur la carte des zones d'influence des points de vue (obtenue grâce à une des méthodes décrite dans la section 4).

Notons $\mathcal{N}$ le noyau de convolution utilisé, et $N : \mathcal{N} \rightarrow (R)$ la fonction associant une pondération à chaque point du noyau. Notons $\alpha_i(p)$ la participation de la vue de la $i^{\text{ème}}$ caméra dans le calcul des paramètres du rayon $r(p)$ associé au pixel p . On veut $\sum_{i=1}^{\text{card}(\mathcal{C})} \alpha_i(p) = 1$.

La participation de chaque vue se calcule alors de la façon

$$\alpha_i(p) = \frac{\sum_{n \in \mathcal{N}, (p+n) \in \mathcal{Z}_i} N(n)}{\sum_{n \in \mathcal{N}} N(n)}$$

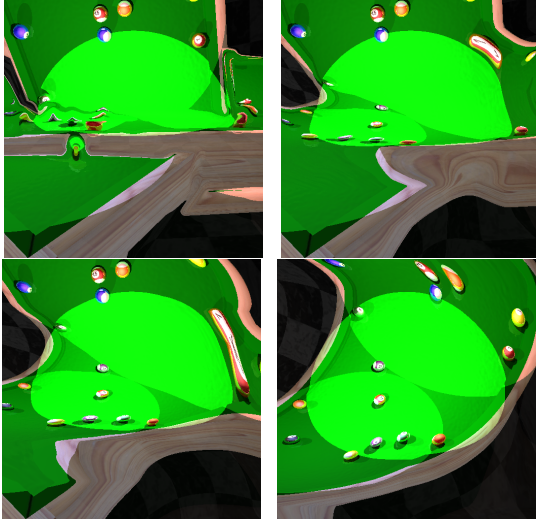


Figure 5: Utilisation de noyaux de taille 25, 125, 255, 511 sur la composition de la figure 4.

L'intérêt d'utiliser les convolutions vient de la variété des noyaux pouvant être utilisés, et dont les différents effets sont faciles à appréhender. On peut ainsi aisément contrôler la taille des zones d'interpolation, en jouant sur la taille du noyau, mais d'autres effets intéressants sont possibles, en utilisant des variations sur les pondérations des points du noyau (voir la figure 6 pour quelques exemples).

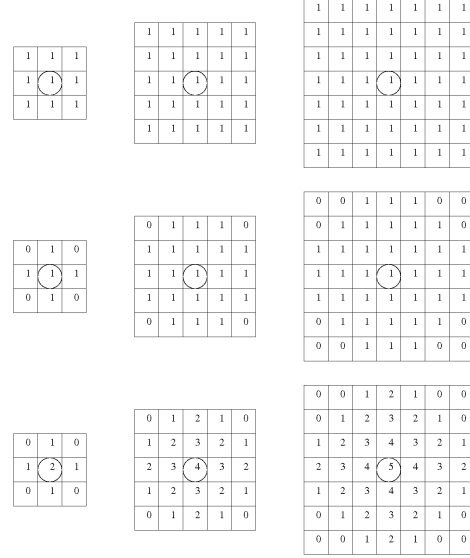


Figure 6: Exemples de noyaux de convolution (le cercle représente la position du centre) : en haut, des noyaux de répartition uniforme rectangulaire, de taille 3*3, 5*5 et 7*7. Au milieu, des noyaux de répartition uniforme circulaire, de taille 3*3, 5*5 et 7*7. En bas, des noyaux de répartition non uniforme.

5.2. Interpolation séparée des paramètres intrinsèques et extrinsèques sphériques

Le problème de l'interpolation directe des coordonnées cartésiennes entre les rayons dans le repère global de la scène vient du fait que les rayons interpolés entre deux rayons r_1 et r_2 auront tous leur origine située sur une même portion de droite, entre les origines de r_1 et r_2 . Si ces deux rayons se trouvent de part et d'autre de l'objet, certaines interpolations risquent de se retrouver dans l'objet, ce qui donne des rendus non désirés. Pour contourner ce problème, on se propose de donner des coordonnées sphériques à chaque caméra. L'interpolation se fait alors dans un premier temps sur les rayons, mais ici dans le référentiel caméra, afin de pouvoir utiliser simultanément de façon fluide divers types de caméra (pinhole, isométrique, fisheye, ou tout autre caméra linéaire), puis de faire une interpolation sur les coordonnées sphériques des caméras, afin de replacer le rayon obtenu lors de la première interpolation dans le référentiel de la scène. Cette seconde interpolation à comme

caractéristique de ne plus couper l'objet, mais de tourner autour, du fait de l'utilisation des coordonnées sphériques.

On note $r(p)$ le rayon associé au pixel p dans le rendu interpolé, avec $O(p)$ son origine, et $V(p)$ son vecteur direction. Notons $r_i(p) = (O_i(p), V_i(p))$ le rayon associé au pixel p pour la i^{eme} caméra et $(\theta_i, \phi_i, \rho_i)$ les coordonnées sphériques de la i^{eme} caméra. En utilisant les participations de chaque vue, calculées comme indiqué dans la sous-section 5.1, on peut calculer r en effectuant les deux interpolations de la façon suivante. Définissons d'abord $r' = (O'(p), V'(p))$ qui correspond à r avant le changement de référentiel caméra vers scène. On a :

$$O'(p) = \sum_{i=1}^{card(\mathcal{C})} \alpha_i(p) * O_i(p)$$

$$V'(p) = \sum_{i=1}^{card(\mathcal{C})} \alpha_i(p) * V_i(p)$$

De la même façon, on peut calculer les coordonnées sphériques (θ, ϕ, ρ) de la caméra virtuelle d'où provient le rayon r' pour donner r .

$$\theta(p) = \sum_{i=1}^{card(\mathcal{C})} \alpha_i(p) * \theta_i(p)$$

$$\phi(p) = \sum_{i=1}^{card(\mathcal{C})} \alpha_i(p) * \phi_i(p)$$

$$\rho(p) = \sum_{i=1}^{card(\mathcal{C})} \alpha_i(p) * \rho_i(p)$$

Il ne reste plus qu'à changer le référentiel de r' , à l'aide de ces coordonnées, pour obtenir r .

Cette méthode d'interpolation est bien adaptée pour le rendu d'un sujet unique, pouvant être contenu dans une sphère. Elle est en revanche complètement inadaptée au rendu d'un sujet non délimité, comme un paysage, par exemple. Dans ce dernier cas, on préférera utiliser l'interpolation linéaire sur les coordonnées cartésiennes.

6. Discussion

Les rendus obtenus avec nos méthodes peuvent être considérés comme cubistes, du point de vue théorique. En pratique, les images obtenues, après des traitements "effet peinture" appropriés, peuvent être apparentés aux tableaux de différents maîtres du cubisme et d'autres courants de peinture multiperspective (voir figure 7). Ainsi, la composition par points de vue par zone d'influence, en prenant un nombre de points de vues suffisant, permet d'obtenir des rendus du style de ceux de la période cubisme analytique. La composition par carte de profondeurs, de son côté, permet d'obtenir des rendus préservant un certain nombre de repères visuels,

et tend plus vers le cubisme synthétique. En utilisant ces méthodes avec une interpolation par noyau de convolution de taille suffisamment grande, on peut obtenir des images faisant penser à certaines oeuvres de Dali, comme ses fameuses montres molles.

7. Conclusion

Dans cet article, nous avons proposé de nouvelles méthodes de composition multiperspective, dans le but de faire des rendus abstraits de type cubiste. Ces méthodes ne requièrent aucune interaction avec l'utilisateur, permettant de produire des séries de rendus cubistes très facilement. Ces méthodes sont bien entendu encore amplement améliorables, notamment l'interpolation sur les coordonnées sphériques, dont on pourrait tenter de raccourcir la trajectoire d'interpolation, en passant par l'utilisation des grands cercles. Même si les résultats déjà obtenus sont intéressants, de nombreuses autres approches sont envisageables, comme par exemple la capacité de cibler le cubisme sur les différents objets de la scène, de façon indépendante et automatique, ou encore d'envisager non plus le cubisme dans une scène 3D fixe, mais animée dans le temps.

References

- [BSS07] BROSZ J., SAMAVATI F. F., SHEELAGH M. T. C., SOUSA M. C. : Single camera flexible projection. In *NPAR '07 : Proceedings of the 5th international symposium on Non-photorealistic animation and rendering* (New York, NY, USA, 2007), ACM, pp. 33–42.
- [CH03] COLLOMOSSE J., HALL P. : Cubist Style Rendering from Photographs. *IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS* (2003), 443–453.
- [FLM00] FELS S., LEE E., MASE K. : Techniques for Interactive Video Cubism. In *Proceedings of ACM Multimedia* (2000), pp. 368–370.
- [FM99] FELS S., MASE K. : Interactive video cubism. In *Proceedings of the 1999 workshop on new paradigms in information visualization and manipulation in conjunction with the eighth ACM international conference on Information and knowledge management* (1999), ACM New York, NY, USA, pp. 78–82.
- [Gla04a] GLASSNER A. : Digital Cubism. *IEEE Computer Graphics and Applications* (2004).
- [Gla04b] GLASSNER A. : Digital Cubism, Part 2. *IEEE Computer Graphics and Applications* (2004).
- [KSC*01] KLEIN A., SLOAN P., COLBURN A., FINKELSTEIN A., COHEN M. : Video Cubism. *Technical Report MSR-TR-2001-45* (2001).
- [LG96] LOFFELMANN H., GROLLER E. : Ray Tracing with Extended Cameras. *The Journal of Visualization and Computer Animation* 7 (1996), 211–227.

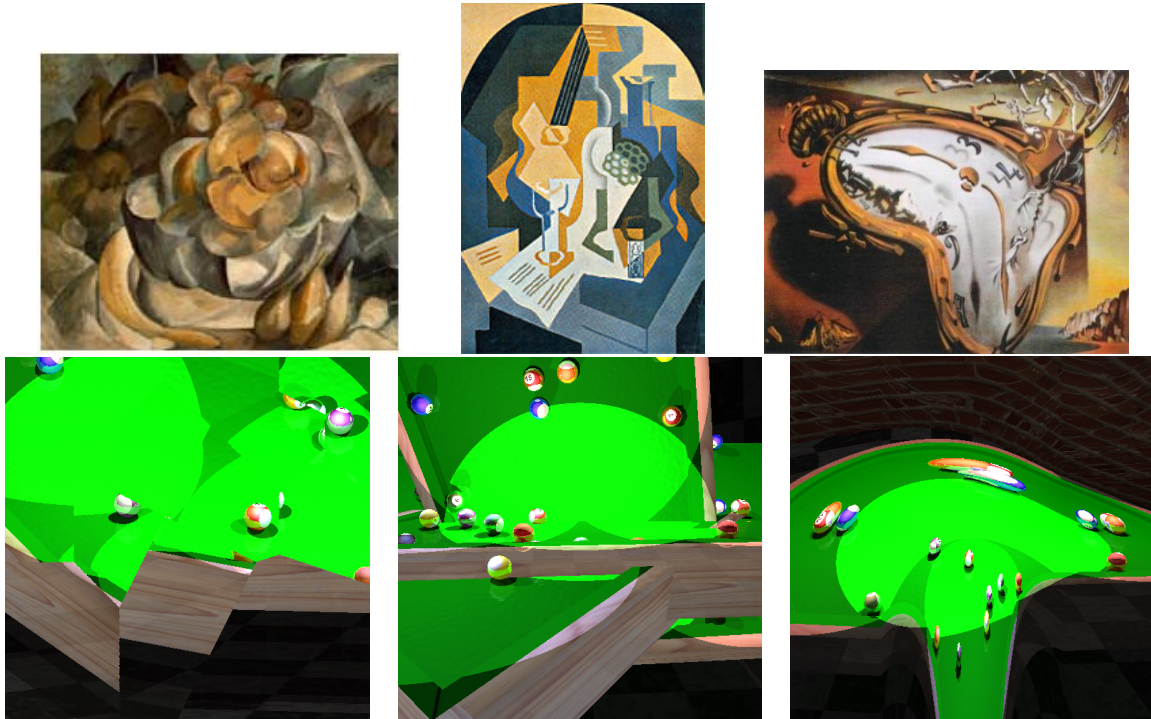


Figure 7: En haut, de gauche à droite : un tableau de la période cubiste analytique, "Le compotier", de Georges Braques. Un tableau de la période synthétique, "Still Life with Fruit Dish and Mandolin" de Juan Gris. Un tableau post cubiste, les fameuses "Montres molles" de Salvador Dali. En bas, des rendus obtenus par nos méthodes. De gauche à droite : composition par zones d'influence, composition par cartes de profondeur, composition par zones d'influence combinée avec une interpolation par convolution utilisant un noyau de grande taille.

[MA00] MEADOWS S., AKLEMAN E. : Creating Abstract Digital Paintings with Painting Camera Technique. In *Information Visualization, 2000. Proceedings. IEEE International Conference on* (2000), pp. 250–254.

[SADK04] SMITH J., AKLEMAN E., DAVISON R., KEYSER J. : MultiCam : A System for Interactive Rendering of Abstract Digital Images. *Bridges : Mathematical Connections in Art, Music, and Science 2004* (2004).

[SRMS06] SPINDLER M., RÖBER N., MALYSZCZYK A., STROTHOTTE T. : *Flexible Film Interactive Cubist Style Rendering*. Univ., Fak. für Informatik, 2006.

[VC01] VALLANCE S., CALDER P. : Multi-perspective images for visualisation. In *Proceedings of the Pan-Sydney area workshop on Visual information processing-Volume 11* (2001), Australian Computer Society, Inc. Darlinghurst, Australia, Australia, pp. 69–76.

[YM04a] YU J., MCMILLAN L. : A Framework for Multiperspective Rendering. In *Eurographics Symposium on Rendering* (2004), pp. 61–68.

[YM04b] YU J., MCMILLAN L. : General Linear Ca-

meras. *LECTURE NOTES IN COMPUTER SCIENCE* (2004), 14–27.