

Lumigraphe et reconstruction géométrique

Bruno Mercier, Daniel Meneveaux, Alain Fournier

Laboratoire IRCOM-SIC, Université de Poitiers, BP 30179
86962 Futuroscope Chasseneuil Cedex, France

{mercier,daniel}@sic.univ-poitiers.fr

Résumé :

Depuis quelques années, les méthodes de rendu basé-image connaissent un succès croissant car elles permettent de visualiser une scène réelle ou virtuelle presque uniquement à partir d'images. Grâce à un compromis entre simplicité de stockage et richesse de représentation, le lumigraphe est probablement aujourd'hui la structure de données basée-image la plus utilisée dans la communauté graphique. Actuellement, l'une de nos préoccupations concerne le rééclairage de lumigraphes, éventuellement pour les intégrer à l'intérieur d'environnements virtuels. Le point clé de cette étude est la normale de la surface associée à l'objet puisque d'une part elle permet de retrouver la position des sources de lumière, leurs caractéristiques ainsi que celles de la surface de l'objet et d'autre part elle est primordiale pour prendre en compte un nouvel éclairage. Dans cet article, notre contribution concerne une méthode de reconstruction géométrique dédiée aux lumigraphes et deux algorithmes d'estimation des normales. La première étape consiste à estimer la géométrie de l'objet selon une méthode de sculpture basée sur la silhouette, le résultat produit est une hiérarchie de voxels représentant sa forme. Nous proposons également de reconstruire un maillage grâce à une version adaptée de l'algorithme des marching cubes et l'évaluation de la normale est réalisée soit directement à partir du maillage produit, soit uniquement à l'aide de la hiérarchie de voxels. Une étude comparative permet d'estimer la qualité des résultats obtenus à la fois pour la reconstruction géométrique et pour les deux méthodes d'estimation des normales.

Mots-clés : lumigraphes, reconstruction géométrique, géométrie discrète, estimation de normale.

1 Introduction

En synthèse d'images, la dernière décennie a vu naître de nouveaux modèles pour lesquels des photographies d'objets réels ne sont plus seulement utilisées comme des textures indépendantes du point de vue. Elles servent de support pour des méthodes interactives de visualisation d'environnements ou d'objets réels complexes. Ces méthodes de visualisation, identifiées sous le nom de *rendu basé-image* (ou RBI) permettent de réduire de manière drastique une description géométrique généralement très complexe tout en conservant un réalisme important.

Naturellement, lorsque les images initiales sont des photographies d'objets réels, la visualisation est également réaliste, moyennant divers problèmes d'échantillonnage. Mais le concept du rendu basé-image est également intéressant dans le cas d'objets virtuels complexes et difficiles à afficher en des temps raisonnables. La création d'un lumigraphe (par exemple) correspond alors à un précalcul permettant ensuite de visualiser l'objet de manière interactive.

En pratique, les modèles proposés pour le rendu basé-image permettent d'exprimer les flux lumineux émis en chaque point de l'espace et dans chaque direction, indépendamment des objets qui composent la scène. Cela correspond à la fonction *plénoptique* introduite par Adelson et Bergen en 1991 (figure 1.a). Elle est usuellement définie dans un espace de dimension 6 : 3 coordonnées représentent la position, 2 angles définissent une direction, le dernier paramètre étant la longueur d'onde. Notons que dans le cas de scènes animées, une variable de temps peut également faire partie des paramètres.

Naturellement il est illusoire de trouver une formulation analytique à cette fonction pour prendre en compte tous les transferts lumineux dans un environnement, même s'il est simple. En revanche, un échantillonnage judicieux de l'espace permet de représenter une partie exploitable de cette fonction. Les lumigraphes [GGSC96, LH96], au même titre que les textures dépendant du point de vue [DTM96] ou encore la méthode de modélisation plénoptique [MB95] correspondent à un tel échantillonnage. La représentation la plus utilisée aujourd'hui reste probablement le lumigraphe, également appelé *champ de lumière* (ou *lightfield*) [LH96]. Depuis son apparition, elle suscite un intérêt croissant pour de nombreuses études [MGS03, WCCG02, CLF98, VMW03]. Un lumigraphe est composé

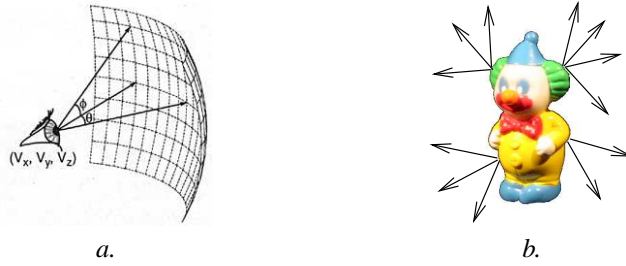


FIG. 1 – a. La fonction plénoptique exprime le flux lumineux émis par un point (V_x, V_y, V_z) et dans toutes les directions ; b. Cette fonction peut être utilisée pour représenter les flux lumineux émis par la surface d'un objet.

de plusieurs paires de polygones parallèles, appelés *slabs* et maillés de manière régulière. Chaque slab représente la lumière traversant le volume délimité par ses deux polygones et pour toutes les directions définies par le maillage (figure 2).

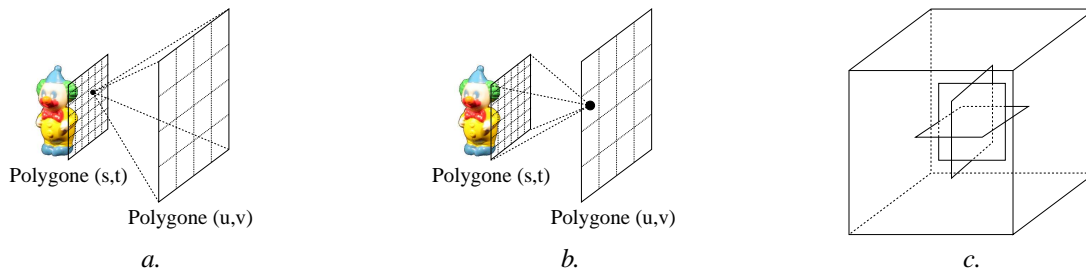


FIG. 2 – Echantillonnage de la fonction plénoptique par un slab (u, v) et (s, t) ; a. pour le flux lumineux passant par un point (s, t) vers toutes les directions (u, v) ; b. pour une image de l'objet à partir d'un point (u, v) ; c. plusieurs slabs pour représenter la fonction plénoptique autour d'un volume

Dans cet article, notre contribution concerne :

1. l'utilisation et l'adaptation aux lumigraphes d'une méthode de sculpture pour reconstruire, selon plusieurs niveaux de détail, une surface à base de voxels correspondant à l'objet visualisé ;
2. une extension de l'algorithme des marching cubes à l'intérieur des voxels pour reconstruire une surface polygonale aussi proche que possible de celle de l'objet ;
3. deux méthodes différentes pour estimer la normale à sa surface en chaque voxel ;
4. une étude statistique permettant de comparer les résultats obtenus pour la reconstruction géométrique et pour l'estimation des normales.

Ce travail est motivé par une étude plus large, visant à modifier l'éclairage initial d'un objet de type lumigraphe. La normale est l'un des points clés de ce travail car elle permet de retrouver la position des sources de lumières et par conséquent une information précise sur les caractéristiques photométriques de l'objet. Une description géométrique peut également être utile dans ce contexte, par exemple pour estimer les ombres auto-portées lors du rééclairage.

Ce document est organisé de la manière suivante : nous présentons dans un premier temps les travaux concernant le rendu basé-image, les dernières études relatives aux lumigraphes et à la reconstruction géométrique. Dans la section 3, nous expliquons l'algorithme de sculpture utilisé dans notre application, avant de détailler nos modèles d'estimation de normale (section 4). Nous proposons une analyse comparative de ces méthodes dans la section 5 avant de conclure et présenter nos objectifs à plus long terme.

2 Références

Historiquement, les méthodes de visualisation s'appuient sur des images d'objets réels ou virtuels pour accroître le réalisme et réduire les temps de calcul. Les utilisations les plus connues sont probablement le plaquage de tex-

tures et les cartes d'environnement, mais d'autres applications telles que les cartes animées (*movie-maps*) [Lip80], quicktime VR [Che95], ou le musée virtuel (*the virtual museum*) [MOC⁺92] associent également une description géométrique à des séquences vidéo ou à des séries d'images.

Les méthodes de rendu basé-image (RBI) ont réellement émergé au début des années 1990, avec la définition de la fonction plénoptique. En 1995, L. McMillan et G. Bishop ont proposé la méthode de *modélisation plénoptique* [MB95] pour représenter les flux lumineux selon un échantillonnage cylindrique. Les lumigraphes et le rendu des champs de lumières (*lightfield redering*) ont fait leur apparition l'année suivante [LH96, GGSC96], ainsi que les textures dépendantes du point de vue [DTM96]. Depuis, les lumigraphes ont servi de support à diverses applications, par exemple pour stocker les informations goniométriques de sources lumineuses [MGS03], ou pour représenter la luminance émise par la surface d'un objet [WCCG02]. Dans [VMW03], les auteurs proposent de modifier l'éclairage d'un objet réel à partir de lumigraphes grâce à des prises de vues réalisées sous plusieurs conditions d'éclairage différentes.

Notre objectif est de modifier l'éclairage de lumigraphes *standards*, c'est-à-dire acquis dans des conditions d'éclairage uniques. Une estimation de la forme et des propriétés photométriques de l'objet est alors nécessaire. Il s'agit d'un problème de reconstruction traité depuis de nombreuses années en robotique, en vision ou en imagerie médicale. Une solution classique et efficace consiste à sculpter l'objet selon sa silhouette de manière hiérarchique (avec un arbre octal). Cette méthode a été proposée par R. Szeliski en 1993 [Sze93] - C. Esteban aux journées AFIG de 2001 [CHE01] a proposé une amélioration permettant d'affiner la reconstruction des cavités de l'objet. Ce type de sculpture produit un ensemble de voxels à l'intérieur desquels une surface est reconstruite. Par exemple la méthode des *marching cubes* proposés par W. Lorensen en 1987 [LC87] génère un maillage polygonal. Des surfaces paramétrées peuvent également être construites de manière efficace selon la méthode décrite dans [LB03].

3 Reconstruction Géométrique

Dès 1996, avec les lumigraphes, l'équipe de Microsoft Research utilise estime la géométrie de l'objet à l'aide de l'algorithme de sculpture de R. Szeliski [Sze93] pour résoudre certains problèmes d'échantillonnage. Nous avons choisi d'utiliser et d'adapter ce même algorithme pour notre travail en y ajoutant une étape supplémentaire permettant de créer une surface précise de l'objet par la méthode des *Marching cubes*.

3.1 Algorithme de sculpture

La sculpture proposée dans [Sze93] nécessite de définir un premier volume cubique englobant l'objet. Ce volume est subdivisé en 8 sous-volumes, projetés successivement sur toutes les images disponibles. Ces volumes sont des voxels pouvant être classés de 3 manières :

- un voxel est *externe* ou *hors de l'objet* si il existe une image sur laquelle il se trouve totalement à côté de l'objet (figure 3.a).
- un voxel est *interne* ou *dans l'objet* si sur toutes les images il est vu à l'intérieur de l'objet (figure 3.b).
- un voxel est *ambigu* dans tous les autres cas : il contient la surface de l'objet (figure 3.c).

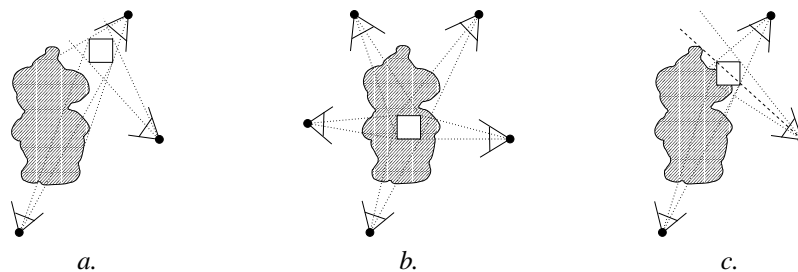


FIG. 3 – a. voxel externe et plusieurs points de vues ; b. voxel interne, toujours vu à l'intérieur de l'objet ; c. voxel ambigu.

Tous les voxels ambigus sont à leur tour découpés en 8 sous-volumes, avec une classification initialisée à *interne*. Une nouvelle itération de l'algorithme est appliquée à ce nouvel ensemble de voxels et ce processus de sculpture est répété jusqu'à atteindre une précision donnée par l'utilisateur (figure 4).

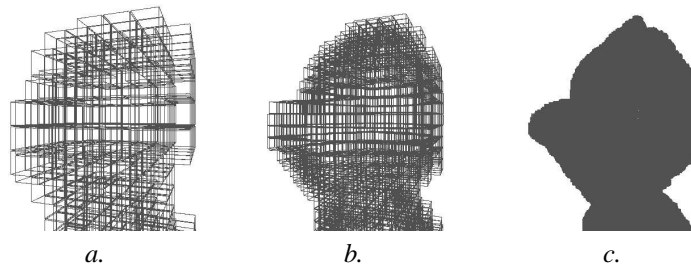


FIG. 4 – Vues des voxels de l'objet, a. au niveau 5 de la hiérarchie, b. au niveau 6, c. au niveau 9.

Dans le cas des lumigraphes, ce type de reconstruction est particulièrement bien adapté car le nombre d'images est très important. Généralement, les slabs sont positionnés tout autour de l'objet afin de pouvoir le visualiser à partir de n'importe quel point de l'espace. Par conséquent l'objet peut être reconstruit de manière relativement précise. Cependant, la silhouette n'est pas toujours suffisante pour retrouver toutes les cavités d'un objet. Par exemple l'anse d'une tasse vide est bien reconstruite, mais l'intérieur ne peut être creusé.

En pratique, nous considérons les échantillons (u, v, s, t) comme une liste de rayons partant d'un plan uv et atteignant soit l'objet (les *rayons-objet*), soit l'arrière-plan (les *rayons-fond*) comme le montre l'image de la figure 5. Un rayon atteint l'objet uniquement lorsque la couleur correspondante n'est pas le fond (noir pour nos lumigraphes). Tous les voxels traversés par un rayon voient leur classification modifiée selon le tableau de la figure 5. Notons que les rayons-objet correspondent également à un flux lumineux allant d'une petite partie de surface de l'objet vers le plan uv .

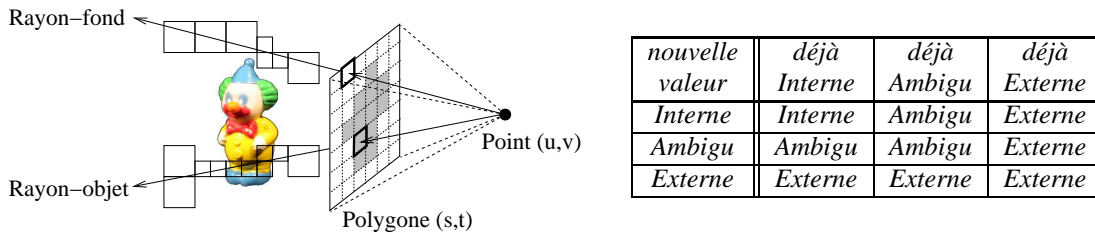


FIG. 5 – Intersection des voxels par un rayon-objet et un rayon-fond et table de classification des voxels pour les rayons de chaque nouvelle image.

Nous appliquons l'algorithme des *marching cubes (MC)* sur l'ensemble des voxels ambigus pour reconstruire une surface polygonale à l'intérieur de chaque voxel. Cet ensemble de voxels ambigus, appelé *surface discrète*, doit impérativement être 6-connexe pour obtenir au final un maillage polygonal représentant la surface de l'objet. Nous avons donc défini un traitement permettant de respecter la configuration souhaitée pendant l'algorithme de sculpture.

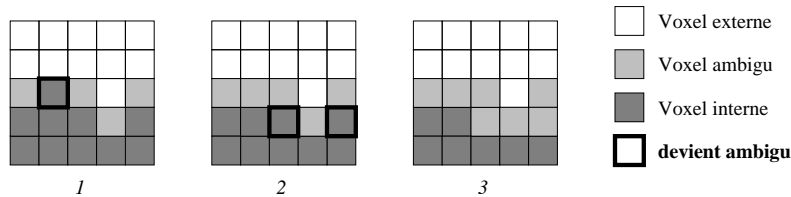


FIG. 6 – Traitement de l'épaisseur de la surface discrète reconstruite. a. Un voxel interne est en contact direct avec un voxel externe ; b. la surface n'est pas 6-connexe ; c. surface discrète finale.

Si un voxel interne V_{in} est directement voisin d'un voxel externe V_{out} , la surface discrète est trop fine. Pour résoudre le problème lorsque ces deux voxels correspondent à des niveaux différents de la hiérarchie, le plus petit étant issu d'un ancien voxel ambigu, nous choisissons de lui réaffecter cette classification. En revanche lorsque V_{in} et V_{out} ont la même taille, le voxel interne est reclassé ambigu afin de sculpter l'objet autant que possible, suivant les règles de l'algorithme de R. Szeliski. Cette étape d'épaississement est réalisée après chaque itération. De plus, à la fin de l'algorithme, lorsque toutes les itérations sont terminées, un parcours réduit l'épaisseur de la surface discrète (en conservant la 6-connexité) en supprimant les voxels ambigus les plus à l'extérieur de la surface.

Remarque : cette méthode de sculpture produit une représentation très précise de l'objet puisqu'après 8 itérations, la taille d'un voxel est de l'ordre de celle d'un pixel pour une image de résolution 256×256 .

3.2 Reconstruction surfacique

Avec la classification obtenue par les traitements de la section précédente, il est possible de reconstruire un maillage de l'objet en utilisant la méthode des *marching cubes* [LC87] à l'intérieur de tous les voxels ambigus. Ce maillage peut alors être utilisé soit pour estimer la normale à la surface de l'objet, soit pour produire une représentation polygonale aussi précise que possible d'un objet réel dont on ne dispose que de photographies.

La méthode des marching cubes (MC) a été développée en 1987 par W. Lorensen et H. Cline. L'objectif est de reconstruire une surface à partir d'un ensemble de voxels en connaissant la position de leurs sommets par rapport à l'objet, soit à l'intérieur soit à l'extérieur. Avec 8 sommets et à une symétrie près, il existe 14 combinaisons pour définir la surface à l'intérieur d'un voxel (figure 7, [LC87]).

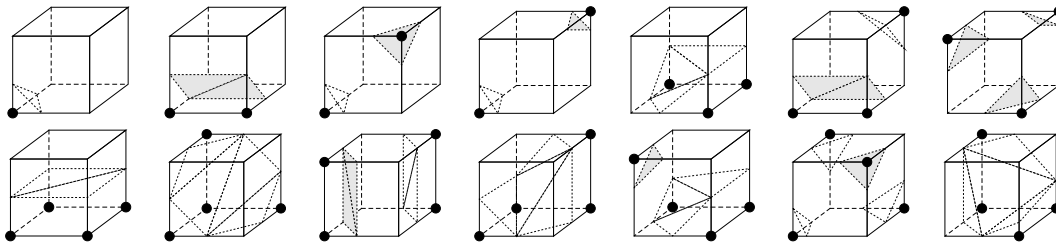


FIG. 7 – Les 14 combinaisons possibles pour les marching cubes.

Il est également possible d'associer une pondération à chaque sommet du voxel pour ajuster au mieux chaque triangle du maillage. Dans le cas des lumigraphes, la classification des sommets est directement réalisée pour tous les voxels ambigus à partir des relations de voisinage, mais aucune pondération ne peut exister contrairement à des images numériques médicales. En revanche, les échantillons (u, v, s, t) des lumigraphes permettent d'affiner la représentation. Pour que les rayons-fond n'atteignent pas la surface reconstruite, nous proposons de prendre en compte leurs points d'intersection avec les faces de chaque voxel ambigu. Sur chaque face, cet ensemble de points définit une zone dans laquelle aucun triangle ne peut être défini (figure 8).

Nous utilisons conjointement la classification des sommets et ces points d'intersection pour construire un *segment de coupe* sur chaque face (figure 8). Entre deux faces adjacentes du voxel, les segments ainsi créés ne coïncident généralement pas et nous les ajustons de manière à les rapprocher autant que possible de la surface (figure 9.a). Le problème de cohérence est résolu de la même façon entre deux voxels voisins 6-connexes (figure 9.b). Notons qu'avec les données des lumigraphes, il est possible de lever directement les ambiguïtés liées à l'algorithme MC (figures 8.b et 8.c).

A l'intérieur de chaque voxel ambigu, une petite partie de la surface est ainsi reconstituée. L'ensemble de ces parties permet de reconstituer un maillage complet de l'objet. De cette manière, nous affinons la géométrie de l'objet.

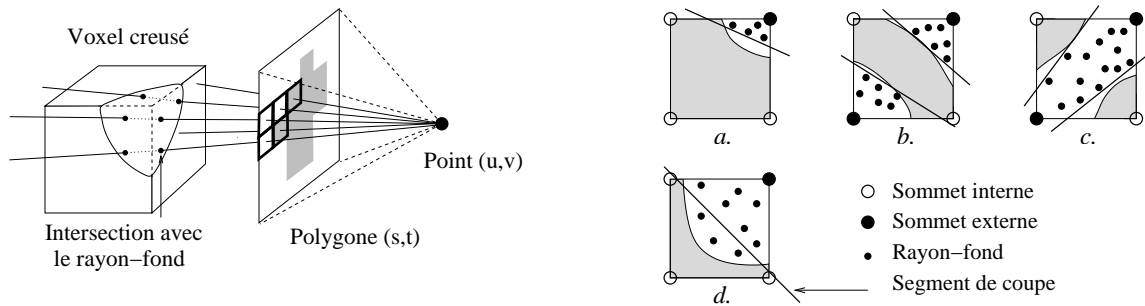


FIG. 8 – Intersection d'un voxel par plusieurs rayons-fond/objet. Estimation de l'intersection entre la surface de l'objet et une face d'un voxel atteint par plusieurs rayons-fond. a. Un seul sommet est à l'extérieur, et plusieurs rayons-fond; b. et c. deux sommets opposés sont à l'extérieur de l'objet et les rayons-fond sont disposés selon deux configurations différentes; d. reconstruction erronée de la surface.

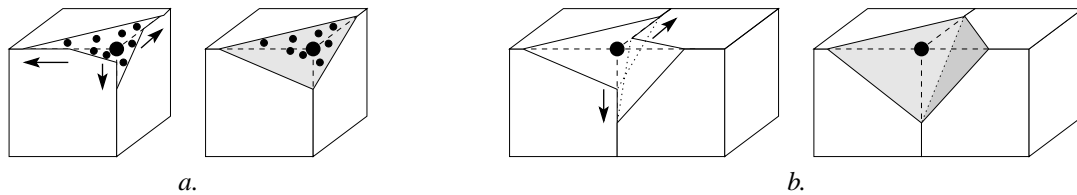


FIG. 9 – a. Choix du segment à modifier pour construire la surface à l'intérieur d'un voxel; b. surface entre deux voxels.

4 Estimation de la normale

La surface ainsi générée peut évidemment être utilisée pour réaliser des calculs de visualisation ou de simulation d'éclairage. En pratique, au niveau de précision maximal (une hiérarchie de profondeur 9 dans nos exemples), elle est définie par plusieurs centaines de milliers de triangles (tableau 1). Notre objectif n'est pas de l'utiliser en l'état, mais plutôt d'estimer une normale pour chaque voxel ambigu. Pour cette estimation, nous proposons une méthode *surfactive* et une méthode *discrète*. La première est basée sur les triangles obtenus grâce à l'algorithme MC et la seconde repose sur la classification des voxels et leur voisinage.

4.1 Normale à partir de la surface

A l'intérieur de chaque voxel ambigu, l'ensemble des triangles définit une estimation de la surface. Le calcul de la normale à partir de ces triangles est réalisée simplement à l'aide d'une moyenne des normales de tous les triangles, pondérée par leur aire (figure 10.a). Avec cette méthode, la normale est estimée directement à partir du maillage produit. Cependant, si la surface paraît précise pour l'objet complet, localement l'algorithme MC produit des irrégularités importantes.

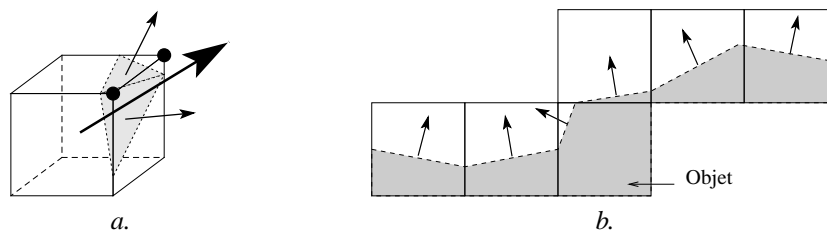


FIG. 10 – a. Estimation de la normale à l'intérieur d'un voxel; b. normales de plusieurs voxels ambigus, la surface est accidentée.

Finalement, ce moyennage à l'intérieur d'un voxel ne produit souvent la normale attendue. Cela est dû aux traitements trop localisés sur la surface. Pour résoudre ce problème, nous proposons de réaliser un lissage basé sur les voxels ambigus voisins. Naturellement, plus le voisinage pris en compte est large, moins la normale est représentative des détails de l'objet.

4.2 Normale discrète

Obtenir une estimation à l'aide du maillage produit semble être une bonne solution mais le résultat est très irrégulier et nécessite un lissage. Si la surface polygonale n'est pas utile, une méthode d'estimation discrète plus rapide et directement basée sur le voisinage des voxels peut également donner des résultats très cohérents (figure 11).

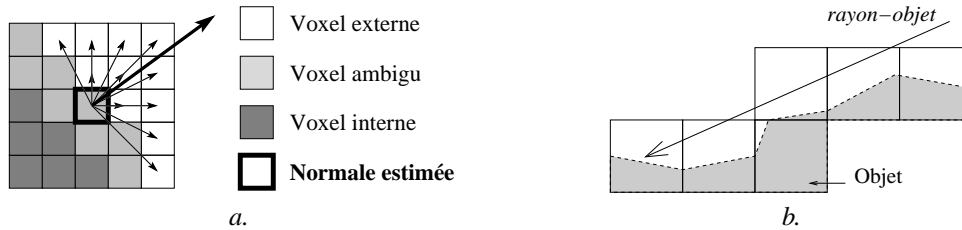


FIG. 11 – a. Estimation de la normale discrète pour un voxel ambigu ; b. un rayon-objet traverse plusieurs voxels ambigus, il est difficile de déterminer à quel endroit il atteint effectivement la surface.

Pour estimer une normale discrète de chaque voxel ambigu, nous utilisons directement les voxels externes voisins V_e . La normale discrète est calculée de la manière suivante : $N = (\sum_i V_i) / i$, V_i étant le vecteur unitaire allant du centre du voxel V_a au centre du voxel V_e . Plus le voisinage est étendu, plus la normale représente un objet d'aspect lisse.

Dans certaines configurations les voxels ambigus n'ont pour voisin que d'autres voxels ambigus ou des voxels internes. Cela arrive par exemple si un tube très fin traverse l'objet. Dans ce cas, nous proposons de réaliser une moyenne des normales avec les voxels ambigus voisins. Il existe cependant toujours quelques configurations où ce traitement reste inefficace et le niveau de précision doit alors être augmenté. Si néanmoins cela reste impossible, alors pour ce voxel aucune normale cohérente n'est identifiable.

5 Mise en œuvre et résultats

La reconstruction d'un objet à partir d'images reste un problème délicat, même dans la configuration idéale où les lumigraphes sont issus d'images synthétiques. Dans cette section, nous soulevons quelques problèmes concernant l'acquisition de lumigraphes et détaillons les principes de mise en œuvre. Nous présentons également les résultats obtenus pour la reconstruction et l'estimation des normales.

5.1 Acquisition de lumigraphes

Nous avons travaillé d'une part avec deux lumigraphes synthétiques, réalisés à partir d'objets virtuels et d'autre part avec un lumigraphe représentant un objet réel. Les lumigraphes synthétiques sont simples et rapides à construire, mais surtout dans notre cas ils permettent de valider les algorithmes développés car l'environnement de la prise de vue est parfaitement défini et calibré. Nous avons choisi d'adapter le logiciel PovRay et d'utiliser deux objets modélisés par Stewart Cowley (<http://www.povray.org>) : le *quad* et le *cochon à roulettes* (figure 12). Pour cela, nous stockons autant d'images que d'échantillons (u, v) . Chaque image correspond à tous les échantillons (s, t) pour une valeur fixée de (u, v) .

Cependant, l'un des intérêts majeurs d'un lumigraphe est qu'il permet de visualiser des objets réels. C'est pourquoi nous avons conçu un système d'acquisition simple et très peu onéreux pour construire un lumigraphe. Idéalement, nous aimerions poser l'objet sur une mire et prendre à la main de nombreuses photographies tout autour. En pratique, cela est difficilement réalisable pour plusieurs raisons : le photographe se trouve souvent entre l'objet

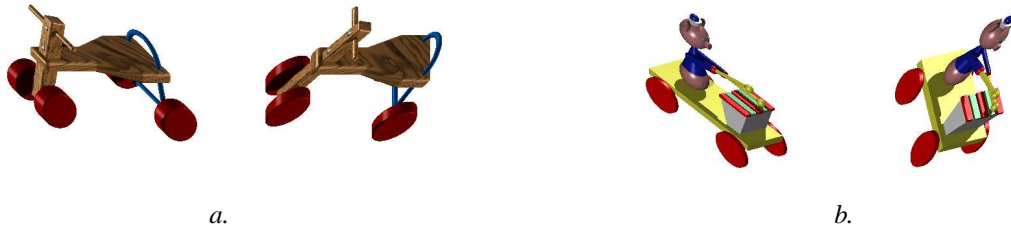


FIG. 12 – Images du lumigraphe pour un échantillon (u, v) donné ; a. le quad ; b. le cochon à roulettes

et l'une des sources lumineuses, l'objet est trop gros ou trop petit pour la mire, la segmentation permettant de séparer l'objet de la mire impose des contraintes importantes sur la forme et la couleur des objets choisis, etc. C'est pourquoi notre choix s'est plutôt porté sur le principe des plateformes d'acquisition plus classiques. La base du support est un plateau tournant sur lequel des sources lumineuses peuvent être fixées (figure 13). L'objet est disposé au centre de ce plateau, sur un pied légèrement surélevé. Un appareil photo numérique est positionné sur un trépied à côté du plateau, sa position (hauteur, inclinaison) est fixe pour les prises de vues d'un tour de plateau complet. Après chaque tour, l'appareil est monté de quelques centimètres et pour un nouveau tour de plateau, des photos sont prises tous les 5 degrés (76 photographies par tour de plateau pour construire un lumigraphe de 4 slabs 8×8 uv). Dans notre cas, ce système est entièrement manuel, l'acquisition d'un lumigraphe nécessite deux jours de prise de vue. Pour chaque position donnée du plateau, deux photos sont prises. La première est surexposée pour faciliter la segmentation de l'objet sur le fond de l'image, la seconde est prise dans les conditions normales d'éclairage du lumigraphe (figure 14.a).

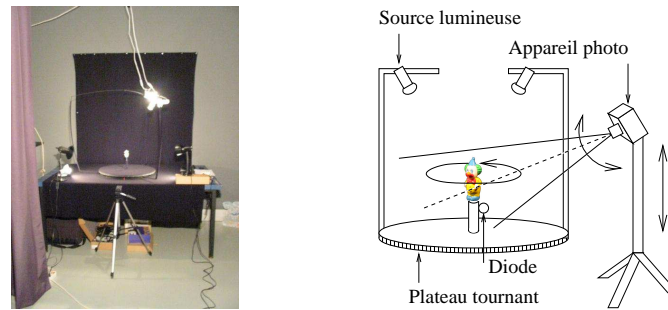


FIG. 13 – Images de notre système d'acquisition.

Une fois la prise de vue terminée, pour reconstituer tous les échantillons (u, v, s, t) du lumigraphe, nous recherchons les pixels des photographies les plus proches. Cette approche permet de reconstruire tous les échantillons du lumigraphe en choisissant les images les plus adaptées.

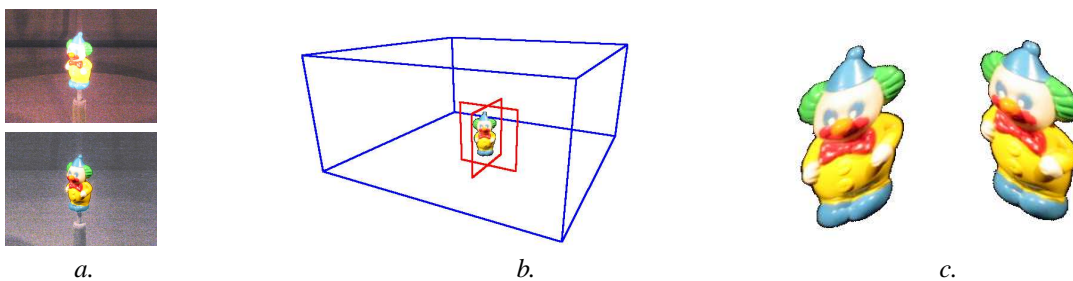


FIG. 14 – a. Photographies du clown en surexposition pour la segmentation et en exposition normale pour le lumigraphe ; b. programme de visualisation interactive de lumigraphes ; c. images des échantillons (s, t) pour deux valeurs fixes de (u, v) .

Nous avons choisi d'utiliser une représentation non compressée des lumigraphes car la méthode proposée par Levoy et Hanrahan [LH96] génère une perte d'information relativement importante. Les images finales du lumigraphe représentent un objet déformé par rapport à une image classique de l'objet car pour l'échantillonnage des polygones uv et st , la direction de visée n'est pas perpendiculaire au plan projectif. Pour les objets synthétiques (le quad et le cochon à roulettes), 6 slabs sont positionnés tout autour de l'objet. Le polygone (u, v) définit 8×8 échantillons et le polygone (s, t) définit 350×350 échantillons soit 384 images par objet. Le clown est représenté uniquement par les 4 slabs illustrés sur la figure 14.b (aucun slab ne représente les parties supérieure et inférieure de l'objet), avec le même échantillonnage pour le polygone (u, v) et 256×256 pour le polygone (s, t) . Ce stockage n'est pas idéal pour une visualisation interactive car les images doivent être chargées en mémoire au fur et à mesure des déplacements du point de vue. Un cache interne au programme est associé au cache du système pour permettre d'obtenir une fréquence d'affichage raisonnable (entre 5 et 25 images par seconde, selon la précision d'affichage choisie).

5.2 Reconstruction par sculpture

Nos algorithmes de reconstruction sont appliqués aux lumigraphes précédents. Lors de la sculpture, le choix du voxel initial pour l'arbre octal est prépondérant pour la suite de l'algorithme. La seule information géométrique disponible du lumigraphe est la description des polygones (u, v) et (s, t) . Nous utilisons une boîte englobant les polygones (s, t) . Nous avons également accéléré le parcours de l'octree par les rayons correspondants aux échantillons (u, v, s, t) grâce à une grille régulière. La taille de cette grille est variable et dépend du niveau de hiérarchie traité à chaque étape de l'algorithme.

La reconstruction du maillage à l'intérieur des voxels ambigus nécessite de mémoriser tous les rayons-fond atteints. Pour des questions d'espace mémoire, nous avons fixé à 255 le nombre de rayons par voxel ambigu. Lorsque le nombre d'échantillons est trop important, une seule image (u, v) sur deux est utilisée. Cela permet d'obtenir un échantillonnage uniformément réparti des rayons mémorisés. Par ailleurs, après plusieurs itérations, la taille des voxels diminue et certains voxels ne sont plus atteints par les rayons correspondants aux échantillons (u, v, s, t) du lumigraphe. Ces voxels sont alors projetés sur les images du lumigraphe comme le prévoit l'algorithme classique de R. Szeliski.

5.3 Algorithme des marching cubes

Pour savoir si la reconstruction de la surface est fidèle à l'objet du lumigraphe, nous proposons de confronter les échantillons (u, v, s, t) avec le maillage. Le tableau 1 représente le nombre de rayons-objet n'atteignant jamais la surface et le nombre rayons-fond atteignant la surface alors qu'ils ne le devraient pas.

	Quad (47 040 000 rayons)		Clown (16 777 216 rayons)	
	niveau 8	niveau 9	niveau 8	niveau 9
nb voxels ambigus	35 971	148 986	37 584	156 685
nb triangles MC	71 926	297 980	75 170	313 378
nb rayons-fond erronés	1 894 (0.6%)	275 (2.7%)	147 (1%)	173 (16%)
nb rayons-objet erronés	533 836 (8%)	565 124 (9%)	1 246 636 (57%)	1 247 900 (59%)

TAB. 1 – Résultats obtenus pour la reconstruction géométrique. Le pourcentage de rayons-fond erronés n'est pas réalisé sur le nombre total de rayons-fond, mais sur le nombre de rayons-fond atteignant au moins un voxel ambigu.

Visuellement, la surface ressemble à l'objet représenté par le lumigraphe. Les résultats obtenus montrent que le nombre de rayons-objet n'atteignant jamais les voxels ambigus ni le maillage est très important car à chaque étape, l'algorithme sculpte l'objet autant que possible. Pour la même raison, le nombre de rayons-fond erronés (atteignant la surface reconstruite alors qu'ils ne le devraient pas) est relativement faible. Il n'est néanmoins pas nul car dans certains cas, les triangles construits par l'algorithme MC ne peuvent représenter la surface de l'objet de manière suffisamment fidèle (voir la figure 8.d).

Concernant l'objet réel, les mesures des paramètres du systèmes d'acquisition sont suffisamment précises pour

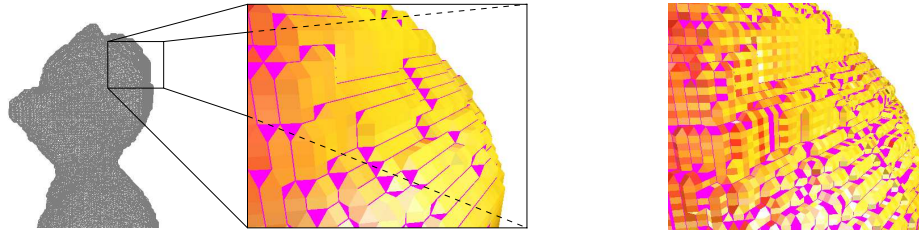


FIG. 15 – Maillage obtenu pour le clown aux niveaux 8 et 9 de la hiérarchie.

une visualisation réaliste. Le maillage de l'objet semble visuellement correct, mais confronté aux données du lumigraphe, il n'est pas satisfaisant (figures 15 et 16). Cela montre combien l'acquisition de lumigraphes d'objets réels est un problème sensible.

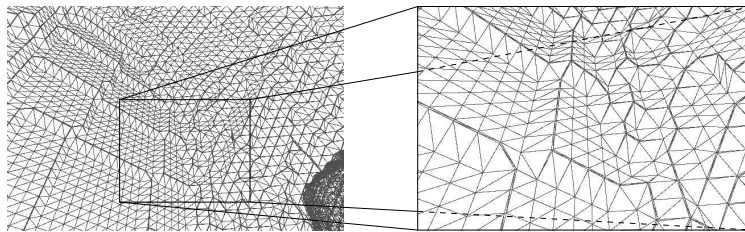


FIG. 16 – Sur cette figure, un zoom montre que la surface est très accidentée au niveau des courbures, et plane par palliers le reste du temps.

5.4 Estimation des normales

Nous avons proposé deux méthodes d'estimation des normales à l'intérieur des voxels ambigus. D'une part, à l'aide d'une surface reconstruite, d'autre part selon une méthode discrète basée sur le voisinage des voxels. Intuitivement, la première méthode semble plus exacte car outre la représentation des voxels, nous exploitons autant que possible les informations du lumigraphe. Cependant, comme nous l'avons remarqué dans la section précédente, la surface reconstruite est localement accidentée et produit une estimation perturbée de la normale.

Pour réduire la fréquence des variations de normale liées au traitement local de l'algorithme MC, nous avons choisi de lisser les valeurs selon un voisinage plus ou moins étendu. Cela produit une normale plus représentative de l'objet. La taille du voisinage influe naturellement sur le résultat final, le tableau 2 permet de comparer les deux méthodes (surfactive et discrète) d'estimation avec et sans lissage. Pour ce tableau, un vecteur de normale est estimé à l'intérieur de chaque voxel ambigu selon les deux méthodes avec un lissage plus ou moins important. Nous estimons ensuite la différence angulaire entre les deux vecteurs obtenus pour tous les voxels. La comparaison est réalisée avec des voisinages d'étendue comparables pour les deux méthodes. Naturellement, plus la taille du voisinage est importante, plus l'écart-type diminue car les deux méthodes convergent vers un même résultat. Cependant au niveau 8, l'écart-type des variations ne reste pas stable lorsque le voisinage est trop important. Cela est dû à un lissage exagéré de la normale, non représentatif dans les parties courbes de l'objet. Au niveau 9, et jusqu'à un lissage de voisinage 5, l'écart-type diminue de manière régulière car la même taille de voisinage représente une partie moins importante de la surface.

Malgré les différences observables dans le tableaux 2, l'objet est visuellement représenté de manière relativement fidèle. Si l'objectif est uniquement d'obtenir une estimation des normales sans avoir un maillage polygonal de la surface, la méthode discrète est beaucoup plus rapide et suffisamment précise.

Voisinage surfacique	0	1	1	2	2	3	3	4	4	5	
Voisinage discret	1	1	2	2	3	3	4	4	5	5	
Quad niveau 8	moyenne	8.4°	10.2°	6.2°	5.9°	4.2°	4.9°	3.6°	5.2°	4.0°	5.6°
	écart-type	9.2°	8.2°	4.7°	4.0°	3.4°	3.5°	3.8°	8.1°	8.1°	11.9°
Quad niveau 9	moyenne	8.2°	8.8°	6.7°	5.1°	4.1°	4.0°	3.1°	3.5°	2.7°	3.3°
	écart-type	10.3°	8.0°	5.3°	3.7°	3.0°	2.7°	2.3°	2.4°	2.2°	2.3°

TAB. 2 – Comparaison (moyenne et écart-type) des variations angulaires des normales entre les méthodes surfacique et discrète, avec différentes tailles de voisinages.

Taille du voisinage	Quad niveau 8			Quad niveau 9		
	1	2	3	1	3	5
Précalcul pour la méthode surfacique	1h20'			2h40'		
Méthode surfacique	7"	9"	14"	26"	55"	2'40"
Méthode discrète	< 1"	4"	13"	2"	16"	1'01"

TAB. 3 – Temps de calcul pour estimer les vecteurs normaux avec la méthode surfacique et avec la méthode discrète.

6 Conclusion et perspectives

Le travail présenté dans cet article est une étape nécessaire pour parvenir au rééclairage d'un lumigraphe dans des conditions correctes. Le problème traité concerne l'estimation d'une normale pour un ensemble de points sur la surface de l'objet visualisé. Dans un premier temps, une adaptation aux lumigraphes de l'algorithme de R. Szeliski [Sze93] donne lieu à un ensemble de voxels représentant l'objet. Cette géométrie approximative sert de support pour la construction d'un maillage polygonal suivant une version étendue de l'algorithme des marching cubes [LC87]. La normale à la surface à l'intérieur de chaque voxel est déduite soit de ce maillage, soit des relations de voisinage entre les voxels de la hiérarchie. Contrairement à notre attente, le maillage résultant n'est pas adapté à une estimation parfaitement adéquate des normales. La méthode discrète fournit des résultats sensiblement équivalents en des temps de calcul bien moins importants. Les méthodes de reconstruction géométrique et d'estimation de normales décrites ci-dessus sont dédiées aux lumigraphes mais elles sont naturellement applicables à d'autres types de structures de données. Pour retrouver les sources de lumière et modifier l'éclairage de l'objet, le maillage polygonal ne doit pas être utilisé car il contient trop d'irrégularités et ce niveau de précision est en réalité inutile.

7 Remerciements

Nous avons débuté ce travail avec Alain Fournier en 1999 avant sa disparition, cet article présente la première étape d'un travail aujourd'hui plus abouti. Nous remercions également Stewart Cowley pour avoir accepté de nous laisser utiliser les modèles du *quad* et du *cochon à roulettes*.

Références

- [Che95] Shengang Eric Chen. Quicktime vr - an image-based approach to virtual environment navigation. *Computer Graphics*, 29(Annual Conference Series) :29–38, August 1995.
- [CHE01] Francis Schmitt Carlos H. Esteban. Reconstruction 3d d'objets par multi-stéréo. *Actes des journées AFIG*, pages 27–36, December 2001.
- [CLF98] Emilio Camahort, Apostolos Leros, and Donald Fussell. Uniformly sampled light fields. In G. Drettakis and N. Max, editors, *Rendering Techniques '98 (Proceedings of Eurographics Rendering Workshop '98)*, pages 117–130, New York, NY, 1998. Springer Wien.

- [DTM96] Paul E. Debevec, Camillo J. Taylor, and Jitendra Malik. Modeling and rendering architecture from photographs : A hybrid geometry- and image-based approach. *ACM Computer Graphics*, 30(Annual Conference Series) :11–20, August 1996.
- [GGSC96] Steven J. Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F. Cohen. The lumigraph. *ACM Computer Graphics*, 30(Annual Conference Series) :43–54, August 1996.
- [LB03] Adriano Lopes and Ken Brodlie. Improving the robustness and accuracy of the marching cubes algorithm. *IEEE Transactions on Visualization and Computer Graphics*, 9 :16–29, 2003.
- [LC87] William E. Lorensen and Harvey E. Cline. Marching cubes : A high resolution 3d surface construction algorithm. *ACM Computer Graphics*, 21(Annual Conference Series) :163–169, July 1987.
- [LH96] Marc Levoy and Pat Hanrahan. Lightfield rendering. *Computer Graphics*, 30(Annual Conference Series) :31–42, August 1996.
- [Lip80] Andrew Lippman. Movie-maps : An application of the optical videodisc to computer graphics. In ACM, editor, *SIGGRAPH*, August 1980.
- [MB95] L. McMillan and G. Bishop. Plenoptic modeling : An image-based rendering system. *Computer Graphics*, 29(Annual Conference Series) :39–46, 1995.
- [MGS03] W. Heidrich M. Goesele, X. Granier and H. Seidel. Accurate light source acquisition and rendering. *ACM Siggraph*, pages 621–630, July 2003.
- [MOC⁺92] G. E. Miller, S. E. Offert, E. Chen, D. Patterson, S. Blacketter, S. A. Rubin, J. Yim, and J. Hanan. The virtual museum : Interactive 3d navigation of a multimedia database. *The Journal of Visualization and Computer Animation*, 3(3) :183–197, 1992.
- [Sze93] Richard Szeliski. Rapid octree construction from image sequences. In 58, editor, *CVGIP : Image Understanding*, volume 1, pages 23–32, July 1993.
- [VMW03] Philip Dutré Vincent Masselus, Pieter Peers and Yves D. Willems. Relighting with 4d incident light fields. *ACM Siggraph*, 22(3) :613–620, July 2003.
- [WCCG02] Michael H. Chu Wei-Chao Chen, Jean-Yves Bouguet and Radek Grzeszczuk. Light field mapping : efficient representation and hardware rendering of surface light fields. *ACM Siggraph*, pages 447–456, 2002.