

Rendu en Portion de Ciel Visible de Gros Nuages de Points 3D

F. Duguet^{1,2}, D. Girardeau-Montaut^{2,3}

¹INRIA Sophia-Antipolis - ²ENST Paris - ³EDF R&D

Florent.Duguet@sophia.inria.fr,
Daniel.Girardeau-Montaut@enst.fr

Résumé : Nous nous intéressons au traitement de très gros nuages de points (plusieurs dizaines de millions). Le rendu de tels nuages s'effectue, de manière rapide, à l'aide de modèles d'éclairage locaux. Mais le réalisme est moindre que celui obtenu avec des modèles d'éclairage globaux. Ces derniers requièrent quant à eux la définition d'une surface à partir du nuage de points. Nous utilisons un modèle d'éclairage intermédiaire, la portion de ciel visible. Nous déclinons ce modèle en deux versions : une version 2D1/2 et une version 3D. La première peut être appliquée à des terrains ou assimilés ; la seconde peut opérer sur des nuages de points 3D issus de mesures, discrétisées dans un arbre octaïdre. Les résultats de ces méthodes ont été utilisés par des archéologues pour guider les travaux d'accostage et d'assemblage de fragments d'une grande colonne grecque. La version 2D1/2 sera utilisée à des fins d'illustration dans des publications archéologiques.

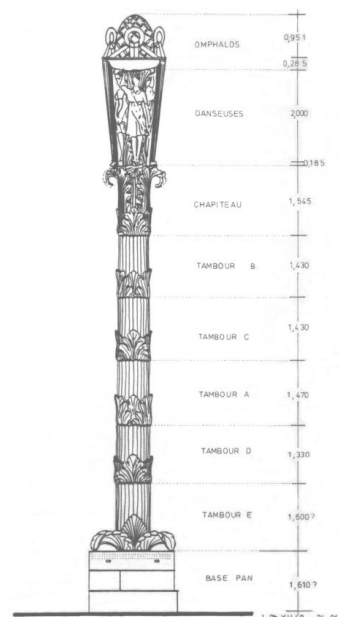
Mots-clés : Portion de Ciel Visible, Nuages de Points, Rendu par Points

1 Introduction

L'acquisition de données 3D pour l'étude et la préservation du patrimoine se répand à la fois chez les spécialistes de la synthèse d'image s'intéressant à ce sujet ([LPC⁺00]), mais aussi et surtout chez les archéologues et conservateurs qui ont de plus en plus recours aux techniques numériques. L'objectif de ces travaux est de mettre à disposition des archéologues des techniques de visualisation performantes. Ces travaux ont été réalisés dans le cadre d'un projet d'étude et de recherche archéologique sur une colonne du site de Delphes en Grèce, appelée la "Colonne des Danseuses". Cette colonne de 14 mètres de haut est aujourd'hui formée de plus de 200 fragments. Les hypothèses archéologiques de "remontage" de la colonne sont difficiles à vérifier en raison de la masse des fragments, et de leur état de conservation. L'outil numérique semble donc approprié.

La colonne (dessinée à droite par J.-Ph. XILLO en 1996) a été scannée fragment par fragment puis les nuages de points correspondant ont été classés, répertoriés, puis consolidés les uns aux autres. Leur mode de représentation est resté tout au long du traitement le nuage de point "brut" tel que fourni par le capteur. Seules des transformations rigides ont été appliquées sur les mesures (points 3D), et aucune reconstruction à base de polygones n'a été effectuée. Cette restriction est une demande expresse des archéologues qui ne souhaitent pas apporter de subjectivité à la mesure, au travers d'une reconstruction en maillage par exemple. Toute la chaîne de traitement, de l'acquisition au rendu final, ne traite donc que des nuages de points.

Afin de remonter la colonne, virtuellement au moins, et ainsi vérifier ou infirmer des hypothèses archéologiques, plusieurs rendus de fragments ont été effectués, puis présentés aux archéologues. Le rendu en portion de ciel visible a donné les résultats les plus illustratifs et les plus utiles aux archéologues. La version 2D a été utilisée à plusieurs reprises pour étudier et affiner le positionnement des fragments et a grandement facilité l'accostage des blocs. Certaines hypothèses de placement des fragments n'avaient pas été envisagées par les archéologues avant l'utilisation de ces outils de rendu.



Le mode de rendu que nous appelons ici "portion de ciel visible", est de plus en plus répandu dans les systèmes de rendu commerciaux. Appelé *Ambient Occlusion*, cette méthode d'éclairage est issue du lancé de rayons. Pour chaque élément à ombrer (que ce soit un pixel dans l'espace image, ou un sommet pour un maillage en espace

objet), la voûte celeste est échantillonné, et des rayons sont lancés pour obtenir l'éclairage. Cette approche peut être accélérée matériellement en utilisant un rendu en projection orthographique et une carte de profondeur comme dans [BCS01]. Notre méthode opérant sur des nuages de points, le rendu en projection orthographique ou le tracé de rayons peuvent certes être effectués, mais requièrent un contrôle fin de plusieurs paramètres.

Dans la section 2, nous présentons la méthode en deux dimensions, puis en section 3 nous décrivons la structure de données utilisée pour l'algorithme en 3D, que nous décrivons en section 4. Nous présentons enfin des résultats dans la section 5.

2 Algorithme en 2D

Dans cette partie, nous décrivons l'algorithme de rendu en portion de ciel visible en 2D utilisé pour aider les archéologues à placer les fragments sur la colonne et pour créer des illustrations des fragments.

2.1 Développées

Nous travaillons sur une colonne composée de plusieurs tambours (9), chaque tambour faisant approximativement 1,5 mètres de haut et 75 cm de diamètre. Les premiers tambours ont une forme cylindrique, avec décorations, et les derniers des formes plus complexes. Afin de les représenter dans des ouvrages descriptifs, les archéologues souhaitent déplier ces cylindres en rectangles, ce qui rend plus simple l'illustration des détails artistiques sculptés sur la colonne. Nous effectuons de ce fait le dépliage, le nuage de points s'apparentant alors à un terrain. Le résultat de ce procédé est appelé une **développée**.

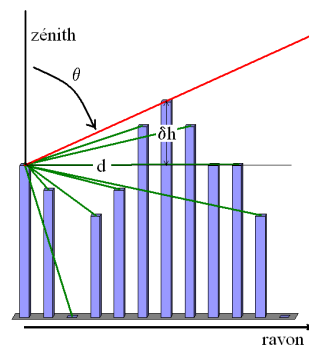
2.2 Image de hauteurs

Une développée peut donc être considérée comme un terrain, avec deux axes pour la position sur le terrain, et la distance radiale qui devient la hauteur. Ce nuage de points est ensuite placé dans une image de résolution donnée. Si la résolution est adaptée à la fréquence d'échantillonnage, tous les pixels sont couverts par au moins un point. Nous stockons pour chaque pixel la hauteur maximale des points présents dans le pixel, ce qui nous donne une image de hauteur. Un premier rendu possible pour cette développée serait d'affecter à chaque pixel une couleur correspondant à sa hauteur (soit en fausses couleurs soit en niveau de gris). Cependant, comme le montre la figure 1, il est difficile de distinguer les détails de sculpture.

2.3 Portion de Ciel Visible

Le principe de l'algorithme est le suivant : pour chaque point de l'image, il s'agit de calculer la portion de ciel visible, c'est-à-dire, la portion de l'hémisphère des directions qui n'est pas stoppée par le terrain. Le terrain étant représenté par une image de hauteur, c'est un problème 2D1/2. Nous quantifions ainsi les directions dans le plan de l'image, et pour chaque direction, nous calculons l'angle de la ligne d'horizon par rapport à l'horizontale. Nous calculons donc la portion de ciel visible pour chaque secteur angulaire.

Dans un secteur angulaire, on se ramène à un problème 1D1/2. Depuis le point d'intérêt (celui pour lequel on souhaite calculer la portion de ciel visible), on calcule la position de la ligne d'horizon en lançant un rayon en 2D dans l'image de hauteur. Lors de ce parcours, chaque cellule touchée par ordre de distance au point est un pas de l'algorithme. On conserve la tangente de l'angle avec le zénith, c'est à dire le rapport entre la différence de hauteurs (notée δh sur la figure) et la distance au point (notée d). Une fois que le rayon a quitté l'image, l'horizon le plus haut donne la portion de ciel visible (voir figure à droite).



Afin d'optimiser les calculs, les parcours relatifs des rayons dans l'image sont précalculés pour chaque direction quantifiée (40 directions typiquement). Pour une direction donnée le parcours des cellules voisines par le rayon est calculé relativement à la cellule centrale, et peut ainsi être réutilisé pour la même direction dans tous les autres pixels.

2.4 Résultats

L'omphalos est le bloc situé au sommet de la colonne. Il a une forme d'ogive et les détails de sculptures sont un entrelacement de chaînes dont les mailles sont de petits motifs décorés. Figure 2, le rendu en Portion de Ciel Visible (PCV) de l'omphalos. Figure 3, le rendu PCV de la développée du tambour 2, qui a servi au placement de fragments du tambour 1. Figure 4, le rendu en PCV de la développée du tambour 6-7. On peut noter les zones gradées entourées en rouge (en bas à gauche).

Ce mode de rendu va être choisi pour les illustrations de la publication des résultats obtenus sur la colonne des danseuses, dans l'ouvrage à paraître [Marre].

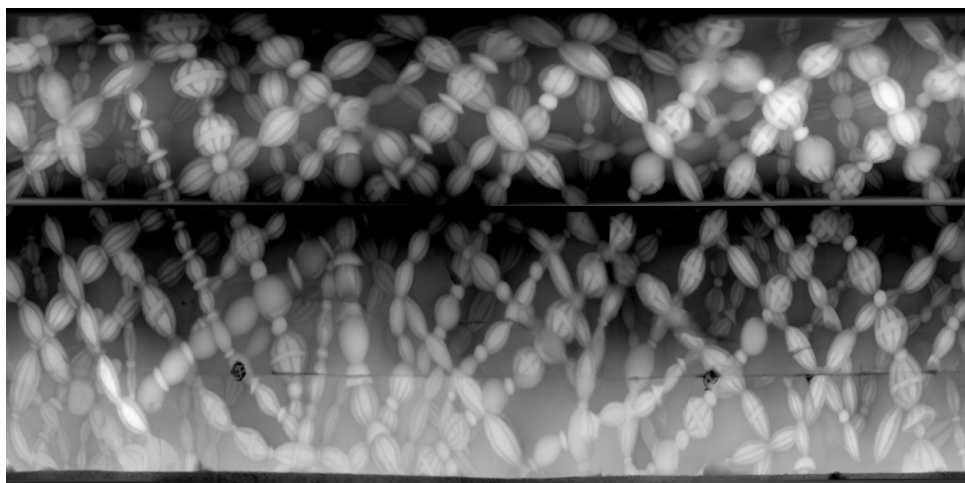


FIG. 1 – Cette image a été obtenue en faisant une développée en deux parties : une partie conique (la partie supérieure), et une partie cylindrique (la partie inférieure). Cette décomposition a permis d'avoir un meilleur contraste. Cette représentation de l'objet peut être comparée avec celle en Figure 2



FIG. 2 – L'omphalos en portion de ciel visible. On peut noter que ce genre d'objet, ayant une forme d'ogive non parfaite, est très difficile à représenter à plat. Les entrelacements des détails sculptés seraient de plus une vraie gageure pour un illustrateur. Enfin, ce mode d'illustration ne souffre pas de la subjectivité d'un dessinateur.

objet	résolution	temps (<i>min</i> : <i>sec</i>)	nb rayons	rayons par sec.
tambour 2	2522 x 1536	12 : 16	154 951 680	210 532
chapiteau	2048 x 1536	13 : 32	125 829 120	154 962
omphalos	2277 x 1024	05 : 07	93 265 920	303 797

TAB. 1 – Résultats des calculs de rendu en PCV 2D effectués sur un Pentium IV à 3.0 GHz, pour 40 directions dans le plan

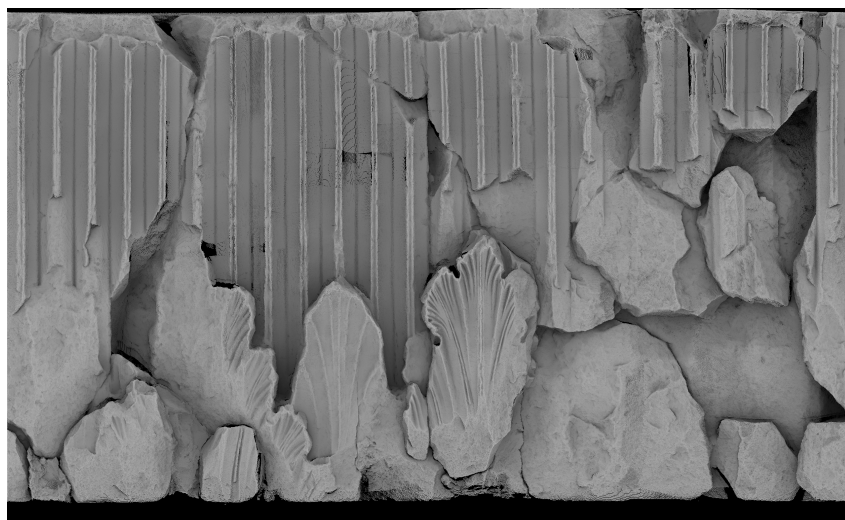


FIG. 3 – Le tambour 2 en rendu PCV. On peut noter la mise en évidence des canelures et des nervures de feuilles que permet ce mode de rendu. Cette image a été utilisée pour accoster des fragments du tambour 1.

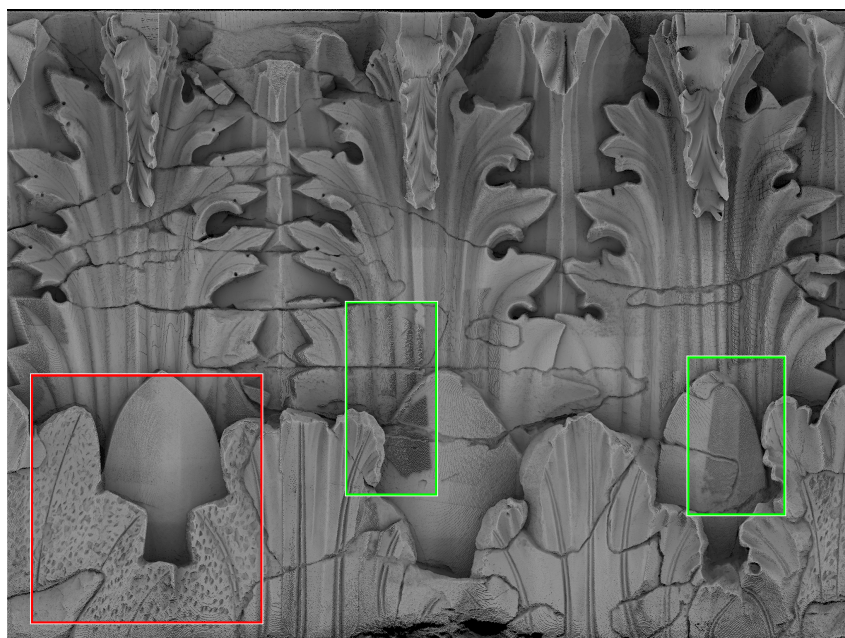


FIG. 4 – Les tambours 6 et 7 en rendu PCV. En rouge (en bas à gauche), on peut nettement voir une zone travaillée à la gradine. Ce mode de rendu permet aussi de mettre en évidence les zones pour lesquelles la consolidation est à améliorer (zones en vert : centre et droite).

3 Structure de Données 3D

Afin de capturer les détails de sculpture des colonnes, les fragments ont été échantillonnés au demi-millimètre ; ceci, rapporté à la taille de la colonne et en prenant en compte les recouvrements, correspond environ à 300 millions de points, soit approximativement 30 millions de points par tambours. Chaque tambour est certes manipulable individuellement sur les machines actuelles ($30 * 12 = 360$ Mo), mais l'ensemble ne l'est pas. Si l'on veut ajouter des informations supplémentaires comme des couleurs ou des normales pour l'éclairage, la mémoire augmente rapidement et dépasse les capacités de la machine. Nous avons donc opté pour une représentation similaire à celle présentée dans [DD04].

Nous travaillons avec un arbre octaïdre. Chaque nœud de l'arbre contient un code sur huit bits (occupation des cellules filles), et un pointeur (ou un indice dans une table) sur la table des cellules filles. Les feuilles contiennent une information libre. La taille d'une cellule est de 40 bits. Les cellules vides ne sont pas allouées : un tableau de la taille nécessaire (nombre de cellules filles) est alloué, et le code permet un parcours de ce tableau pour accéder aux cellules filles. Cette approche est décrite dans [BWK02], et le parcours flexible dans [DD04].

Dans chaque feuille de l'arbre, un seul point est conservé, et sa position est approximée par le centre de la cellule. Cette approximation correspond à une perte en précision, mais dans notre cas, nous utilisons des arbres octaïdres de profondeur 13, et donc une précision de $1/8192$. L'objet étant de taille 1,5 m la taille de la cellule est de 0,18 mm, et l'erreur de quantification est donc inférieure au dixième de millimètre.

4 Algorithme en 3D

Dans cette partie nous décrivons l'algorithme de rendu en portion de ciel visible en 3D utilisé pour simuler la partie directe d'un éclairage ambiant global et ainsi améliorer la visualisation du nuage de points 3D.

4.1 Eclairage isotrope

Le principe est ici encore de calculer la portion de ciel visible pour chaque point du nuage 3D. Nous avons utilisé un raisonnement un peu différent de celui présenté dans la section 2. Le principe de l'algorithme en 2D était de partir de chaque point (donc du niveau local) puis de s'en éloigner progressivement pour "découvrir" la topographie de son voisinage (et donc remonter au niveau global). Or l'analyse du voisinage étendu d'un point en 3 dimensions est très coûteuse et complexe. Le principe de l'algorithme en 3D consiste donc à partir du ciel pour déterminer les points visibles selon chaque direction de la lumière.

On utilise pour initialiser l'algorithme un arbre octaïdre tel que décrit dans la partie 3. Cet arbre est équivalent en terme de structure à une grille régulière en 3 dimensions, comportant 2^n éléments par côté pour un arbre de profondeur n (soient 2^{3n} éléments au total). Ces éléments correspondent aux cellules de l'arbre et nous emploierons donc le même terme par analogie. Chacune de ces cellules peut être vide ou comporter un certain nombre de points, et chacune possède une variable stockant la quantité d'énergie lumineuse reçue.

L'algorithme consiste à illuminer cette grille selon différentes directions (limitées à l'hémisphère supérieur d'une sphère correspondant au ciel). Ces directions sont choisies de manière isotrope : chaque direction correspond à une portion de la voûte céleste de surface égale. Pour chacune des directions, l'illumination des cellules est accumulée après avoir été calculée en prenant en compte leur occultation relative, selon un principe de propagation à travers la grille 3D. Le sens de cette propagation est dépendante de la direction du rayon lumineux (Cf. figure 5). On obtient finalement, pour chaque cellule, une quantification de l'énergie lumineuse reçue sous un éclairage ambiant (discretisé selon un nombre donné de directions, typiquement 32 ou 64). On admet alors que cette quantité est proportionnelle à la portion de ciel visible.

4.2 Propagation

Pour chaque direction d'éclairage, on doit donc à la fois déterminer l'occultation relative des cellules et calculer la quantité d'énergie transmise en fonction de la présence ou non de points à l'intérieur de celles-ci. On utilise pour cela un principe de propagation à travers la grille 3D : quelle que soit la direction du vecteur d'éclairage, il existe

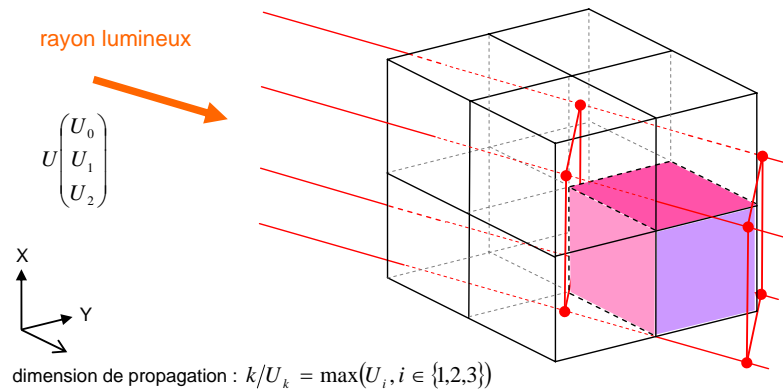


FIG. 5 – Principe de l’illumination selon une direction de la grille 3D correspondant à l’arbre octaïdre calculée sur le nuage de points 3D.

au moins une direction parmi les trois dimensions de la grille selon laquelle la marche de la lumière est maximale. Selon cette direction, on peut découper la grille en "tranches". On détermine alors parmi les deux tranches extrêmes celle qui sera éclairée la première. Pour chacune des cellules de cette tranche initiale, on ajoute à leur variable d’accumulation l’énergie lumineuse initiale du rayon et on se souvient de leur caractère occultant grâce à une structure temporaire. Cette structure temporaire aura la même taille qu’une tranche (en terme d’éléments). On passe ensuite à la tranche suivante, et l’on procède de manière équivalente au calcul de l’énergie lumineuse reçue par chaque cellule en prenant en compte l’occultation des cellules voisines ainsi que celles de la couche précédente, via la structure temporaire. Les informations d’occultation sont au passage mises à jour. Ce processus est répété itérativement jusqu’à la dernière tranche.

Note : pour les informations contenues dans la structure temporaire, on peut soit se limiter à une simple information booléenne du type "cellule transparente / cellule opaque" ou préférer des descriptions plus complexes en fonction de la disposition des points au sein de la cellule. Bien que plus gourmand en mémoire, ceci permet d’évaluer de manière plus précises la quantité d’énergie reçue par les cellules adjacentes et la répartition de la lumière.

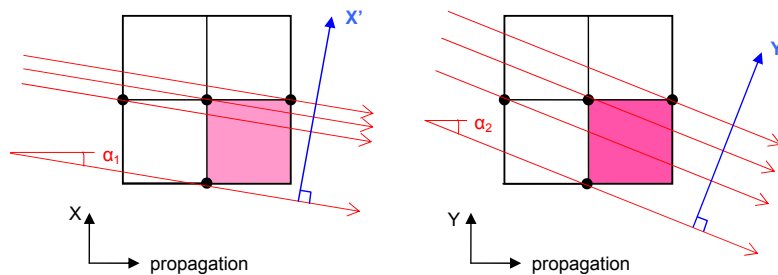


FIG. 6 – Illumination d’une cellule selon une direction et dépendance par rapport à ses voisins.

4.3 Calcul de l’éclairement d’une cellule

On peut remarquer qu’au sein d’une tranche, par construction, la marche de la lumière est inférieure à la largeur d’une cellule selon les deux dimensions de la tranche (car la propagation est faite selon la direction principale qui est la dimension transversale à la tranche). La quantité d’énergie lumineuse reçue par une cellule ne dépend donc que de l’occultation des 3 cellules voisines appartenant à la même couche et des 4 voisines de la couche précédente dans le sens opposé à la marche de la lumière (Cf. figure 6). On en déduit simplement que l’occultation et l’éclairement global d’une cellule selon une direction donnée de la lumière ne dépend que de celles des 7 voisines

selon la direction opposée à la lumière (plus simplement, ce sont les sept cellules de la grille qui seront éclairées "avant"). Or la propagation, si celle-ci est effectuée selon les bonnes directions dans chaque dimension, nous assure la détermination de ces paramètres pour ces 7 cellules avant la cellule en cours. Au bord de la grille, les cellules voisines (non présentes dans la structure) sont considérées comme transparentes : elle transmettent donc l'énergie nominale d'éclairage sans occultation.

Enfin une amélioration supplémentaire de l'algorithme s'appuie sur le fait qu'une cellule n'est occultée en réalité et de manière directe que par 3 cellules parmi les sept décrites précédemment. Ce sont les 3 cellules qui ont une face commune avec la cellule en cours (Cf. figure 7). Les calculs en sont grandement simplifiés.

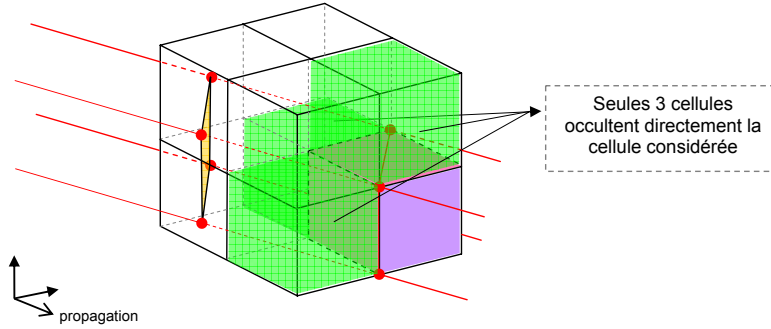
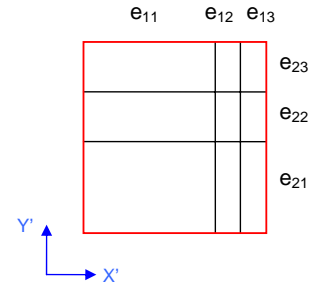


FIG. 7 – Occultation d'une cellule par ses 3 voisines selon le sens opposé à la lumière et ayant une face en commun.

4.4 Modèle d'occultation d'une cellule, découpage adaptatif

Puisqu'on éclaire des cellules cubiques et dans un souci de limitation de la complexité de l'algorithme, on considère un rayon lumineux à section rectangulaire qui englobe entièrement la cellule en cours. Si on se réfère à la figure 6, on observe que le schéma d'occultation d'une cellule par ses voisines est un rectangle divisé en neuf parties (3 sections selon chaque dimension du plan tangent au rayon lumineux). Par un tel découpage (Cf. ci-contre), chaque partie ne dépend que d'une combinaison unique des paramètres des cellules voisines. De plus, ce découpage ne dépend que de la direction du rayon lumineux et est donc commun à toutes les cellules au cours de la propagation.



On peut établir la valeur des différents paramètres. On reprend les notations des figures et on prend comme unité la largeur d'une cellule. Soit δ_i la largeur de la section transversale du rayon lumineux et α_i l'angle d'incidence du rayon selon la dimension i , avec $i : \{X \mapsto 1, Y \mapsto 2\}$. On a alors :

$$\delta_i = (1 + \tan(\alpha_i)) * \cos(\alpha_i) \quad (4.1)$$

$$e_{i1} = \frac{1 - \tan(\alpha_i)}{1 + \tan(\alpha_i)} * \delta_i \quad (4.2)$$

$$e_{i2} = e_{i3} = \frac{\tan(\alpha_i)}{1 + \tan(\alpha_i)} * \delta_i \quad (4.3)$$

$$\sum_1^3 e_{ij} = \delta_i \quad (4.4)$$

Le calcul de l'éclairage d'une cellule se faisant par la superposition des schémas des 3 cellules voisines, cela permet une correspondance élémentaire et une déduction simple des valeurs pour la cellule en cours (Cf. figure 8).

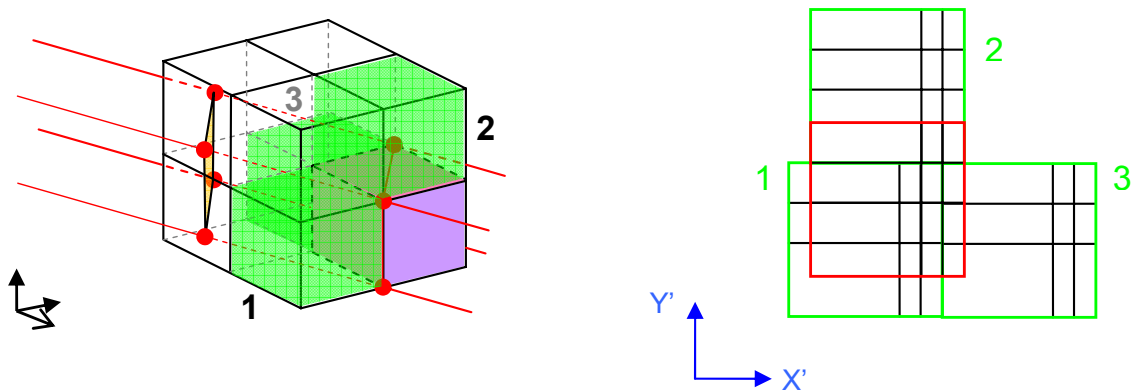


FIG. 8 – Combinaison de schémas d'occultation

La structure cubique de la grille ayant une place prépondérante tout au long de l'algorithme, on limite ainsi sensiblement l'effet très crénelé du résultat que l'on peut observer si aucun découpage n'est effectué (ce qui est le cas si on ne considère qu'une occultation globale et booléenne de la cellule - voir image ci-contre). Plutôt que d'accumuler une information d'éclairage pour chaque cellule, on peut le faire pour chaque point. On raffine ainsi la distribution d'énergie en fonction de la position du point dans la cellule par rapport à ce découpage. Bien que plus gourmand en mémoire, et assez approximatif par rapport à une quelconque réalité "physique", ce raffinement améliore nettement le résultat.



4.5 Découpage par une grille binaire

Nous proposons dans cette partie une optimisation du modèle d'occultation présenté dans la partie précédente.

Plutôt que d'utiliser une partition "flottante" du faisceau lumineux, avec des coefficients non entiers correspondant à l'aire de la section de chaque sous-partie du faisceau, on peut utiliser une grille régulière binaire (Cf. figure 9). Nous avons choisi ici une grille de $8 * 8 = 64$ bits. Cette grille découpe donc le faisceau en 64 parties égales et la valeur du bit correspondant indique si la zone est transparente ou occultante. Les principaux avantages sont le gain en mémoire ainsi réalisé (8 octets contre 9 flottants) et la rapidité des calculs qui se résument alors à des décalages de bits ou aux opérations booléennes élémentaires ET et OU.

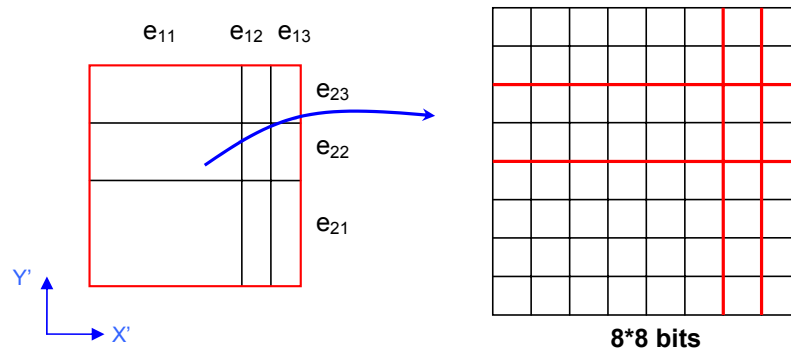


FIG. 9 – Schéma d'occultation par grille binaire régulière

On peut de plus conserver le découpage adaptatif du précédent modèle, du moins dans une certaine mesure. En effet, Si on se réfère aux équations 4.1 à 4.4, on peut exprimer δ_i en bits. Dans le cas particulier d'une grille de 64 bits, on a donc $\delta_i = 8$. Les paramètres e_{ij} vérifient alors :

$$e_{i1} = 8 - 2 * e_{i2} = 8 - 2 * e_{i3} \quad (4.5)$$

Donc, si on exprime toujours ces paramètres en bits, on voit qu'ils ne peuvent prendre que 5 valeurs différentes qui correspondent à 5 angles répartis entre 0 et 45°. Ces valeurs sont relativement proches des valeurs théoriques qu'il faudrait choisir pour un éclairage isotrope (Cf. tableau ci-dessous).

e_{i1}	0	2	4	6	8
e_{i2}, e_{i3}	0	1	2	3	4
$\tan(\alpha_i)$	0	1/7	1/3	3/5	1
α_i (degrés)	0	8.1	18.5	31	45
α_i isotrope	0	10.2	20.7	32	45

TAB. 2 – échantillonnage contraint des paramètres d'illumination par la grille binaire

En pratique, les calculs sont fortement accélérés par le passage en "binaire" (de l'ordre de quatre fois plus rapide d'après nos tests) mais la légère anisotropie contrainte associée au fait que la subdivision du rayon en 64 cellules laisse passer beaucoup de lumière lors de la propagation si le nuage n'est pas très dense, font que le rendu de cette méthode est globalement moins lisse et visuellement moins satisfaisant qu'un rendu avec masque adaptatif en flottants. Ceci n'est pas toujours le cas si les conditions sont très favorables (Cf. figure 11).

5 Résultats

Figure 10, une comparaison entre l'omphalos en ombrage de normales (modèle d'éclairage local), et en portion de ciel visible 3D. Figure 11, la même comparaison pour les tambours 6 et 7. Figure 12, pour les danseuses surmontées de l'omphalos.

L'algorithme n'a pas encore été optimisé pour le moment et les performances ne sont donc pas détaillées. Le rendu en PCV3D des danseuses+omphalos avec 64 directions et une résolution de grille de 2^{3*11} demande 8 heures de calcul, et environ 900 Mo de mémoire. Le rendu final demande 4 secondes par image pour la pleine résolution (40 millions de points sur une image en 2048x2048).

6 Conclusion

Nous avons présenté deux méthodes de rendu de gros nuages de points. La première, le rendu en 2D, permet d'effectuer des illustrations fines ainsi que des guides pour les archéologues et sera utilisée dans une publication archéologique des résultats de recherche sur la colonne des danseuses. La seconde simule un éclairage intermédiaire entre un modèle local et un modèle global et permet ainsi d'obtenir une meilleure visualisation de l'objet en 3D.

7 Remerciements

Electricité de France sponsorise ce projet au bénéfice de l'Ecole Française d'Athènes en offrant son soutien technique et en finançant le partenariat. Electricité de France a conçu le projet et a géré les différents partenariats sous la direction de Guillaume Thibault, que les auteurs souhaitent remercier pour son aide et son soutien. L'équipe Insight dirigée par K. Cain a été choisie pour l'acquisition et la consolidation des données. Nous remercions l'Ecole Française d'Athènes pour avoir participé activement dans la réalisation du projet, et plus particulièrement son directeur, Dominique Mulliez, et Dominique Braustein, conservateur, qui a supervisé la manipulation des éléments. Nous remercions également le Musée du Louvre, en la personne de Jean-Luc Martinez pour ses commentaires

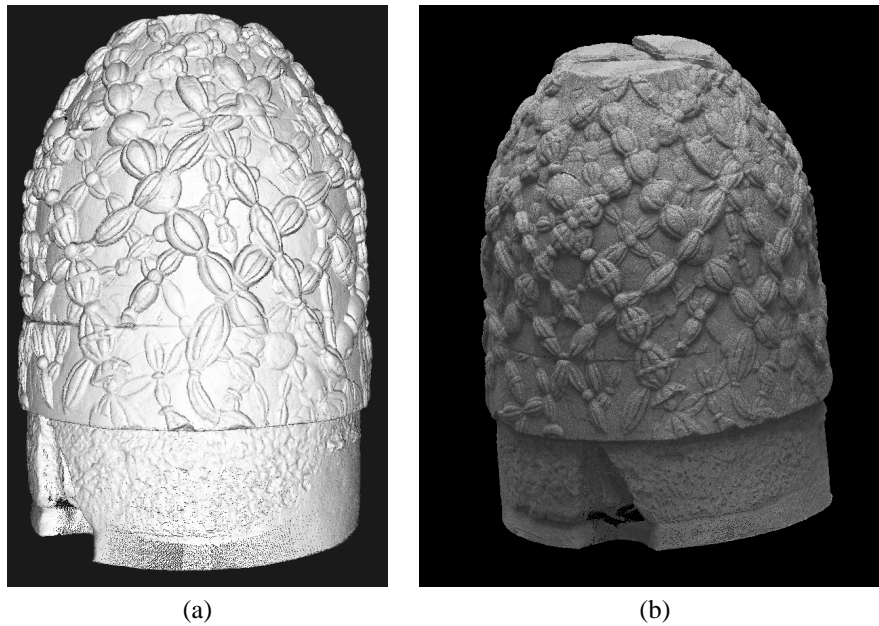


FIG. 10 – L’omphalos en ombrage de normales à gauche (a), et en portion de ciel visible à droite (b).

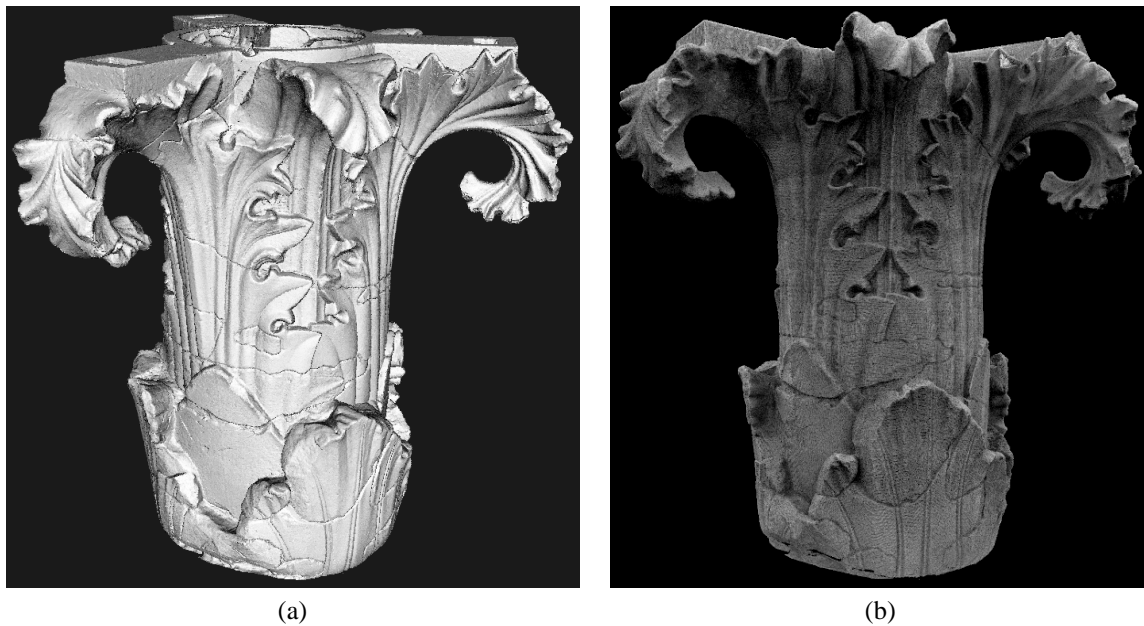


FIG. 11 – Le chapiteau en ombrage de normales à gauche (a), et en portion de ciel visible à droite (optimisation par occultation avec masque binaire de 64 bits) (b).

et directions en qualité d’archéologue. Nous souhaitons également remercier les autorités grecques du musée de Delphes qui ont donné l’accès aux salles alors que le musée était fermé pour travaux, et particulièrement sa directrice Rosina Kolonia, et son assistante, Elena Partida. Alan Chalmers du Department of Computer Science de l’Université de Bristol a fourni un scanner MINOLTA VI910. Deux étudiants de son équipe ont contribué activement au sein de l’équipe d’acquisition et de traitement. Paolo Cignoni du Visual Computing Lab de l’Institut de Sciences Informatiques et de Technologies du Conseil National de la Recherche a contribué à la consolidation des trames 3D. Ces travaux ont été partiellement financés par l’ACI MD “SHOW” du Ministère de la Recherche. Les auteurs souhaitent également remercier Francis Schmitt et George Drettakis pour leurs conseils et leur soutien lors de ce projet.



(a)



(b)

FIG. 12 – Les Danseuses surmontées de l’omphalos en rendu d’ombrage de normales à gauche en en portion de ciel visible à droite (optimisation par occultation avec masque binaire de 64 bits).

Références

- [BCS01] R. Borgo, P. Cignoni, and R. Scopigno. An easy to use visualization system for huge cultural heritage meshes. In *VAST 2001 Conference*, November 2001.
- [BWK02] Mario Botsch, Andreas Wiratanaya, and Leif Kobbelt. Efficient high quality rendering of point sampled geometry. In *Proceedings of the 13th Eurographics Workshop on Rendering*, 2002.
- [DD04] Florent Duguet and George Drettakis. Flexible point-based rendering on mobile devices. *IEEE Computer Graphics and Applications*, 24(4), July-August 2004.
- [LPC⁺00] Marc Levoy, Kari Pulli, Brian Curless, Szymon Rusinkiewicz, David Koller, Lucas Pereira, Matt Gintzton, Sean Anderson, James Davis, Jeremy Ginsberg, Jonathan Shade, and Duane Fulk. The digital michelangelo project : 3d scanning of large statues. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 131–144. ACM Press/Addison-Wesley Publishing Co., 2000.
- [Marre] Jean-Luc Martinez. *La Colonne d’Acanthes de Delphes*, volume IV. Collection des Fouilles de Delphes, à paraître.