

Université de Montréal

Modes de représentation pour l'éclairage en synthèse d'images

par
Romain Pacanowski

Département d'informatique et de recherche opérationnelle
Faculté des arts et des sciences

Thèse présentée à la Faculté des études supérieures
en vue de l'obtention du grade de Philosophiæ Doctor (Ph.D.)
en informatique

Septembre, 2009

© Romain Pacanowski, 2009.

Université de Montréal
Faculté des études supérieures

Cette thèse intitulée:

Modes de représentation pour l'éclairage en synthèse d'images

présentée par:

Romain Pacanowski

a été évaluée par un jury composé des personnes suivantes:

Pascal Guitton,	président-rapporteur
Christophe Schlick,	directeur de recherche
Pierre Poulin,	codirecteur
Xavier Granier,	codirecteur
Bernard Péroche,	examineur externe
Xavier Pueyo,	examineur externe
Victor Ostromoukhov,	représentant du doyen de la FES

Thèse acceptée le: 25 Septembre 2009

À Danielle Pacanowski

Présente chaque jour dans mes pensées.

RÉSUMÉ

En synthèse d'images, le principal calcul à effectuer pour générer une image a été formalisé dans une équation appelée équation du rendu [Kajiya1986]. Cette équation est la intègre la conservation de l'énergie dans le transport de la lumière. Elle stipule que l'énergie lumineuse renvoyée, par les objets d'une scène, dans une direction donnée est égale à la somme de l'énergie émise et réfléchi par ceux-ci. De plus, l'énergie réfléchi par un élément de surface est définie comme la convolution de l'éclairement incident avec une fonction de réflectance. Cette dernière modélise le matériau (au sens physique) de l'objet et joue le rôle d'un filtre directionnel et énergétique dans l'équation du rendu, simulant ainsi la manière dont la surface se comporte vis-à-vis d'une réflexion. Dans ce mémoire de thèse, nous introduisons de nouvelles représentations pour la fonction de réflectance ainsi que pour la représentation de l'éclairement incident.

Dans la première partie de ce mémoire, nous proposons deux nouveaux modèles pour représenter la fonction de réflectance. Le premier modèle s'inscrit dans une démarche artistique et est destiné à faciliter la création et l'édition des reflets spéculaires. Son principe est de laisser l'utilisateur peindre et esquisser les caractéristiques (forme, couleur, gradient et texture) du reflet spéculaire dans un plan de dessin paramétrisé en fonction de la direction de la réflexion miroir de la lumière. Le but du second modèle est de représenter de manière compacte et efficace les mesures des matériaux isotropes. Pour ce faire, nous introduisons une nouvelle représentation à base de polynômes rationnels. Les coefficients de ces derniers sont obtenus à l'aide d'un processus d'approximation qui garantit une solution optimale au sens de la convergence.

Dans la seconde partie de ce mémoire, nous introduisons une nouvelle représentation volumétrique pour l'éclairement indirect représenté directionnellement à l'aide de vecteurs d'irradiance. Nous montrons que notre représentation est compacte et robuste aux variations géométriques et qu'elle peut être utilisée comme système de cache pour du rendu temps réel ou non, ainsi que dans le cadre de la transmission progressive des données (*streaming*). Enfin, nous proposons deux types de modifications de l'éclairement incident afin de mettre en valeur les détails et les formes d'une surface. Le première

modification consiste à perturber les directions de l'éclairement incident tandis que la seconde consiste à en modifier l'intensité.

Mots-clés :

Eclairage global, BRDF, Modélisation de l'apparence, Bases de fonctions sphériques.

ABSTRACT

In image synthesis, the main computation involved to generate an image is characterized by an equation named rendering equation [Kajiya1986]. This equation represents the law of energy conservation. It stipulates that the light emanating from the scene objects is the sum of the emitted energy and the reflected energy. Moreover, the reflected energy at a surface point is defined as the convolution of the incoming lighting with a reflectance function. The reflectance function models the object material and represents, in the rendering equation, a directional and energetic filter that describes the surface behavior regarding the reflection. In this thesis, we introduce new representations for the reflectance function and the incoming lighting.

In the first part of this thesis, we propose two new models for the reflectance function. The first model is targeted for artists to help them create and edit highlights. Our main idea is to let the user paint and sketch highlight characteristics (shape, color, gradient and texture) in a plane parametrized by the incident lighting direction. The second model is designed to represent efficiently isotropic material data. To achieve this result, we introduce a new representation of the reflectance function that uses rational polynomials. Their coefficients are computed using a fitting process that guarantees an optimal solution regarding convergence.

In the second part of this thesis, we introduce a new volumetric structure for indirect illumination that is directionally represented with irradiance vector. We show that our representation is compact and robust to geometric variations, that it can be used as caching system for interactive and offline rendering and that it can also be transmitted with streaming techniques. Finally, we introduce two modifications of the incoming lighting to improve the shape depiction of a surface. The first modification consists in warping the incoming light directions whereas the second one consists in scaling the intensity of each light source.

Keywords: Global Illumination, BRDF, Appearance Modeling, Spherical basis functions.

REMERCIEMENTS

Ces quatre années de doctorat n'auraient pu être aussi enrichissantes et passionnantes sans le soutien inconditionnel et sans faille de mes trois directeurs : Xavier Granier, Pierre Poulin et Christophe Schlick. Ils ont su me transmettre non seulement leur sens de la rigueur scientifique et rédactionnelle, mais aussi une méthodologie pour le travail particulier, qu'est la recherche en synthèse d'images. A travers nos *brainstorming* sans fin sur les notations mathématiques, j'ai surtout appris qu'une idée doit sans cesse être travaillée, développée et retravaillée afin de la transformer en résultats. Leurs nombreuses qualités humaines en font des encadrants hors pair qui ont su, tout au long de mon doctorat, entretenir ma motivation, ce qui m'amène aujourd'hui à écrire ces lignes.

Merci à Bernard Péroche et Xavier Pueyo de m'avoir fait l'honneur d'être les rapporteurs de ce mémoire. Merci aussi à Victor Ostromoukhov d'avoir participé au jury avec une perception différente mais toujours enrichissante. Merci enfin, à Pascal Guitton, chef d'équipe dynamique, qui a présidé mon jury et qui a réussi à faire en sorte que le cadre de travail et d'étude des doctorants soit agréable et fructifiant.

Je tiens à remercier Romain Vergne et Pascal Barla qui m'ont fait participer à une expérience particulière : du LSD au *Light Warping* ; Mickaël Raynaud et Mathieu Gauthier dont l'aide a été essentielle sur la mise en application de l'IVG (Irradiance Vector Grid) et Vincent Lebret-Soler pour son regard HDR.

Au quotidien, je tiens à remercier mes camarades de recherche dont la bonne humeur a largement contribué, dans le laboratoire, à la bonne ambiance qui est si bénéfique et motivante pour notre travail.

A Bordeaux, je n'oublie pas Julien Hadim, mon camarade de bureau 2⁸, qui réussit, en pleine *deadline*, à me transmettre sa *zénitude*, Joachim Pouderoux qui a toujours d'excellentes idées et questions de conception ou de programmation et Jérôme Baril, *Wavelets Man*, pédagogue et patient lorsque je le bombarde de questions.

A Montréal, un grand merci à Luc Leblanc pour ses conseils toujours avisés et optimisés en infographie, Simon Bouvier-Zappa qui m'a transmis le virus Blender, Philippe

Beudoin toujours clairvoyant quand il explique les concepts mathématiques les plus obscurs, François Duranleau qui m'a préparé de la meilleure façon au prédoc écrit et tous mes camarades de *lab* et de 5 à 7 : Marie-Elise Cordeau, Di Jiang, Michelle Laprade et Yann Rousseau.

Enfin, je tiens à remercier Alexia pour une relecture littéraire de ce mémoire scientifique ; Claire, toujours présente et patiente, qui supporte mes divagations sur l'éclaircissement aussi bien au milieu d'un repas qu'en vacances sous les palmiers.

TABLE DES MATIÈRES

RÉSUMÉ	ii
ABSTRACT	iv
I BRDF	1
CHAPITRE 1 : ETAT DE L'ART	2
1.1 Généralités	2
1.2 Notations et grandeurs radiométriques	5
1.3 Paramétrisation	9
1.4 Représentation de la BRDF	12
1.5 Edition et visualisation de la BRDF	21
1.6 Conclusion	25
CHAPITRE 2 : UNE NOUVELLE REPRÉSENTATION POUR L'UTILISATEUR	27
2.1 Aperçu du système proposé	27
2.2 Interface utilisateur	28
2.3 Représentation du reflet spéculaire	30
2.4 Scénarii d'utilisation et résultats	34
2.5 Discussion	37
2.6 Conclusion et travaux futurs	39
CHAPITRE 3 : BRDF ISOTROPES MESURÉES	42
3.1 Introduction	42
3.2 Paramétrisation	43
3.3 <i>Isotropic Shading Function</i>	47
3.4 Formulation analytique de l'ISF	51

3.5	Utilisation pratique de l'ISF	53
3.6	Approximation des matériaux mesurés avec l'ISF	54
3.7	Résultats	57
3.8	Perspectives futures	63
Conclusion de la première partie		65
II Eclairage incident		66
CHAPITRE 4 : ETAT DE L'ART		67
4.1	Éclairage global en synthèse d'images	67
4.2	Aperçu des algorithmes d'éclairage global	71
4.3	Tendances récentes en éclairage global	75
4.4	Travaux antérieurs en édition	78
4.5	Systèmes de cache	81
4.5.1	Formulation mathématique	81
4.5.2	Travaux antérieurs en <i>caching</i>	83
4.6	Conclusion	89
CHAPITRE 5 : STRUCTURES POUR L'ÉCLAIREMENT INDIRECT .		90
5.1	Aperçu du système proposé	90
5.2	Grille de vecteurs d'irradiance (<i>Irradiance Vector Grid</i>)	92
5.3	Applications et résultats	97
5.4	Optimisations	112
5.5	Travaux futurs	115
CHAPITRE 6 : MANIPULER L'ÉCLAIREMENT INCIDENT		118
6.1	Travaux antérieurs	118
6.2	Descripteur géométrique	119

6.3	Modification des directions de l'éclairement incident	120
6.4	Modification de l'intensité de l'éclairement incident	129
6.5	Travaux futurs	134
Conclusion de la seconde partie		136
Conclusion et directions futures		137
ANNEXE I : RADIOMÉTRIE ET PHOTOMÉTRIE		
I.1	Quantités radiométriques	
I.2	Définition quantique de l'énergie rayonnante spectrale	vii
I.3	Définition du flux énergétique spectral en fonction de la température . . .	viii
ANNEXE II : MÉTHODES NUMÉRIQUES D'APPROXIMATION À		
 BASE DE POLYNÔMES RATIONNELS		ix
II.1	Introduction	ix
II.2	DCA pour l' <i>Isotropic Shading Function</i>	ix
II.3	Formulation dans la base monomiale	xi
II.4	Initialisation optimisée	xii
ANNEXE III : REPRÉSENTATION DIRECTIONNELLE : ÉTUDE DE		
 BASES DE FONCTION SUR LA SPHÈRE UNITÉ		xiv
III.1	Bases de fonction sur la sphère	xiv
III.2	Application à l'éclairement	xvii
III.3	Nouvelles bases de fonction	xix
ANNEXE IV : DÉRIVATION DU JACOBIEN xxiv		
BIBLIOGRAPHIE		xxvii

LISTE DES FIGURES

1.1	Exemple de matériaux représentables par une BRDF variant spatialement, ou SVBRDF (<i>Spatially Varying BRDF</i>). Images tirées de [LBAD ⁺ 06].	5
1.2	Un exemple de matériau représentable par une BTF. Images tirées de [MMS ⁺ 06].	5
1.3	Repérage des directions pour la BRDF. La direction incidente ω_i de la lumière et la direction d'observation ω_o sont exprimées dans le repère local $(x, t, \mathbf{n}, \mathbf{b})$ de la surface. ω_i (resp. ω_o) substantive un angle solide différentiel $d\omega_i$ (resp. $d\omega_o$) sur l'hémisphère Ω_n	6
1.4	Illustration de la propriété d'isotropie et d'anisotropie d'une BRDF. (a) BRDF isotrope, (b) BRDF anisotrope dont la direction principale d'anisotropie est verticale ou (c) horizontale. Images extraites de [AS00b].	10
1.5	Deux paramétrisations courantes en infographie. (a) Classique, (b) Rusinkiewicz [Rus98].	10
1.6	Construction d'une BRDF complexe à partir de BRDFs simples. De gauche à droite : la BRDF lambertienne, la BRDF <i>glossy</i> (brillante) aussi appelée directionnelle et la BRDF spéculaire parfaite décrivant le comportement d'un miroir parfait.	14
1.7	Illustration de l'effet de Fresnel sur un matériau comme le bois ciré. (a) La table en bois vue à incidence normale, le coefficient diffus domine le coefficient spéculaire. (b) Pour une incidence à 45 degrés, les coefficients sont plus ou moins égaux. (c) A incidence rasante, le coefficient spéculaire domine. Photographies tirées de [LFTG97].	15
1.8	Comparaison entre le modèle de Ward [War92] et le modèle d'Ashikhmin <i>et al.</i> [APS00]. Images tirées de [APS00].	17

1.9	L'effet d'ombrage des micro-facettes. En fonction de la direction de vue (ω_o) et de la direction de la lumière (ω_i), une proportion des micro-facettes n'est pas active : celles-ci sont masquées par d'autres micro-facettes.	17
1.10	(gauche) Exemple de gonioréfectomètre, image tirée de [LFTW05]. (droite) Dispositif proposé par Matusik <i>et al.</i> [MPBM03b].	19
1.11	Illustration du système d'édition, à l'aide de courbes, de BRDFs de Ben-Artzi <i>et al.</i> Image tirée de [BAOR06].	23
1.12	Utilisation d'une texture pour définir la distribution des micro-facettes du modèle d'Ashikhmin <i>et al.</i> [APS00].	23
1.13	<i>BRDF Shop</i> . Images de gauche : le résultat de l'édition de la BRDF sur un objet. Images de droite : la sphère que l'utilisateur édite pour créer sa BRDF représentée en bas à droite. En haut, une édition de BRDF avec un seul lobe. En bas, édition d'une BRDF avec trois lobes. Images tirées de [CPK06].	25
2.1	Le plan de dessin ou <i>Sketching Plane</i> . L'utilisateur esquisse la forme du reflet spéculaire et peint sa couleur directement sur le plan de dessin qui est orienté perpendiculairement à la direction miroir (\mathbf{r}) de la lumière (\mathbf{l}).	29
2.2	Vue principale de l'interface utilisateur du système proposé. A : (de gauche à droite) zones d'édition de la forme, du gradient de la couleur et résultat final de l'édition dans le plan de dessin. B : outils de dessins. C : visualisation de la direction-clé couramment active. D : sélection de la direction-clé pour laquelle l'édition dans la zone A sera faite. E : résultat final sur l'objet 3D affiché en temps interactif.	31

- 2.3 Décomposition de l'interface proposée à l'utilisateur. (a) Sélection de l'utilisateur. (b) La courbe cyan représente l'esquisse de l'utilisateur tandis que la courbe bleue représente la courbe finale approximée par le processus de minimisation aux moindres carrés. (c) Définition d'un gradient de couleur vectoriel. (d) Vue du plan de dessin montrant le reflet résultant de (b) et de (c). 31
- 2.4 Reflets dessinés avec un courbe de type selon (a) forme étoilée généralisée, (b) une ellipse, (c) une fonction analytique 4-pétales et une réflexion et (d) une fonction analytique 10-pétales. 33
- 2.5 (a) Les directions \mathbf{x} et \mathbf{y} forment le repère local du plan de dessin. Elles sont calculées en fonction de la direction miroir \mathbf{r} de la lumière \mathbf{l} et la tangente géométrique \mathbf{t} . Afin de calculer l'intensité du reflet spéculaire, nous projetons la direction de vue \mathbf{v} sur le plan de dessin. (b) Le champ de distance $d(\rho, \theta)$ défini par une courbe polaire paramétrise l'ensemble du plan de dessin. 35
- 2.6 Illustration des directions-clés. La couleur et la taille du reflet spéculaire sont définies en fonction de la direction de la lumière. Lorsque la lumière est colinéaire à la normale de la surface, (a) le reflet exhibe une couleur verte alors que, lorsque la lumière se déplace vers la direction tangentielle, (b) (resp. bitangentielle (c)) le reflet devient plus bleu (resp. rouge). Notons que pour les configurations (b) et (c), la taille du reflet est plus grande que celle en (a) afin d'augmenter l'effet de changement de couleur. 35
- 2.7 Une forme de reflet plus complexe. La couleur du reflet a été obtenue en combinant un dégradé vectoriel (jaune vers noir) et des dessins effectués directement à l'aide de pinceaux numériques. 36
- 2.8 Dispersion. (a)-(c) Le reflet exhibe un comportement simulant un effet de dispersion qui dépend de la direction de la lumière. La ligne supérieure (resp. inférieure) montre la configuration de la couleur et de la forme pour l'incidence normale (resp. rasante). 38

2.9	Exemple de déformation d'une texture bitmap. (a) La forme initiale du highlight, générée à l'aide d'une texture bitmap contenant un smiley, est changée en (b) un smiley triste en modifiant la courbe polaire qui englobe cette forme.	38
2.10	(a)-(c) Reflet spéculaire rendu pour différentes directions de lumière. (d) La première (resp. deuxième et troisième) ligne représente la configuration d'éclairage lorsque la lumière est colinéaire avec la normale (resp. tangente et bitangente) de la surface. Toutes les configurations utilisent une forme circulaire.	38
3.1	L' <i>Isotropic Shading Function</i> (ISF) permet d'approximer des matériaux isotropes du monde réel. Cette figure montre le résultat obtenu en approximant le nickel extrait de la base de BRDFs MERL-MIT avec seulement deux polynômes rationnels 1D.	44
3.2	Trois modèles analytiques de BRDFs : (haut) exprimés dans la paramétrisation (r, s) ; (bas) exprimés dans l'espace de l'ISF.	46
3.3	Une BRDF lambertienne représentée dans l'espace ISF. L'approximation de Schlick [Sch93a] a été utilisée pour modéliser l'effet de Fresnel.	48
3.4	Projection de la BRDF de Blinn dans l'espace ISF. Le coefficient de Fresnel est constant sur une même ligne alors que la rugosité de la specularité augmente de gauche à droite. L'effet de Fresnel relève la surface à l'approche du point $(s, r) = (1, 0)$. L'augmentation de la specularité du matériau provoque l'aplatissement de la surface, au point qu'elle se réduit pratiquement à une variation le long de l'axe s	49
3.5	Visualisation dans l'espace ISF de neuf matériaux extraits de la base de BRDFs MERL-MIT.	50
3.6	Résultat de la procédure d'approximation sur les courbes 1D pour le matériau metallic-blue. Les mesures sont affichées à l'aide de croix.	51
3.7	Reparamétrisation homographique sur la courbe limite σ_s pour le matériau <i>chrome-steel</i> de la base de données MERL-MIT.	58

3.8	Rendus obtenus avec le matériau <i>fabric-beige</i> (a-d) et le <i>chrome-steel</i> (e-h).	61
3.9	Approximation par composante chromatique du matériau <i>metallic-blue</i> . Les courbes ISFs approximées sur les images de gauche ont 5 (resp. 10) coefficients et l'erreur RMS sur le canal bleu est cinq fois meilleure que celle de la meilleure approximation de Ngan <i>et al.</i>	61
3.10	Différents types de matériaux créés avec l'ISF représentée par des courbes rationnelles quadratiques. La courbe σ_r en vert contrôle la specularité du matériau qui peut varier du diffus au spéculaire parfait. La courbe σ_s en rouge contrôle les effets de Fresnel à angle rasant ainsi que le type de réflexion spéculaire.	63
4.1	Différence de résultats entre (a) une méthode utilisant uniquement l'éclairage direct et (b) l'autre utilisant l'éclairage global. Images tirées de [Cha05].	69
4.2	Rendus avec <i>Photon Mapping</i> . (a) Utilisation directe du <i>Photon Map</i> . (b) Utilisation du <i>Final Gathering</i> . Images tirées de [Jen96].	84
4.3	Comparaison entre de l'éclairage indirect reconstruit avec : (a) gradients de Ward <i>et al.</i> et (b) approximation de meilleure qualité obtenue en échantillonnant l'hémisphère. Images tirées de [KGPB05].	85
4.4	Comparaison du positionnement des échantillons dans le cache en fonction du critère utilisé. (a) Critère de Tabellion et Lamorlette en utilisant la géométrie détaillée. (b) Critère de Tabellion et Lamorlette en utilisant la géométrie faiblement détaillée. (c) Critère de Ward <i>et al.</i> Images tirées de [TL04].	86
4.5	Comparaison de la qualité visuelle obtenue entre : (a) méthode de Ward et Heckbert [WH92] stockant l'irradiance. (b) Méthode de Krivanek <i>et al.</i> [KBPv06] stockant la radiance. Images tirées de [KBPv06].	87

4.6 (gauche à droite) Comparaison de la projection d’une carte d’environnement en utilisant la base des harmoniques sphériques, celle des ondelettes de Haar et celle des *Spherical Radial Basis Functions*. Pour chaque image le même nombre de coefficients (972) a été utilisé. La reconstruction à l’aide d’harmoniques sphériques peine à reconstruire les hautes fréquences de la carte d’environnement : l’effet de Gibbs est visible. Les ondelettes de Haar font apparaître un effet discontinu : les blocs sont visibles. Les *Spherical Radial Basis Functions* ont un comportement plus continu. Images tirées de [TS06]. 89

5.1 Illustration du type de structures utilisées en fonction du type d’objet d’une scène. En plus de la structure globale (en rouge) de la scène, des structures locales 3D (en vert) sont utilisées pour les objets volumineux tandis que des structures locales 2D sont utilisées pour les objets de type planaire. 92

5.2 Illustration 2D du calcul des vecteurs d’irradiance à l’aide d’un tracé de photons. Lorsqu’un photon intersecte une face d’un voxel, sa contribution est ajoutée au vecteur d’irradiance associé au sommet le plus proche de la grille. La contribution du photon est ajoutée au sommet $I_{-x}^{i,j}$ (gauche) et $I_{+y}^{i,j+1}$ (droite). 94

5.3 Stratégie d’échantillonnage pour les grilles. (gauche) Une légère translation dans la direction de la normale est effectuée sur les sommets de la grille. (droite) Certains échantillons peuvent désormais être à l’intérieur de l’objet. On ne tient donc pas compte de la première intersection des rayons partant de ces sommets dans le calcul de l’éclairage indirect. 96

5.4 Texture d’atlas non paramétrisée. (gauche) Sans paramétrisation globale, les normales sont compactées dans un atlas de textures. (droite) Les régions en blanc représentent des normales de surfaces non-définies. (Encadré en bas à gauche) Ceci introduit de légères discontinuités pour les normales reconstruites. 99

- 5.5 Taille en polygones des objets originaux et de leur version simplifiée avec leur temps de pré-calcul pour les représenter sous la forme d'un APO et d'atlas de textures. Ces mesures ont été effectuées sur un processeur AMD 3500+ avec 2 GB de mémoire vive, et une NVIDIA GeForce 8800 GTS. Comme le pré-calcul peut utiliser jusqu'à 4 GB de mémoire, le temps passé par le système à transférer les données de la mémoire vive à celle de *swap* a été supprimé. 99
- 5.6 Scènes de test pour les APOs et les grilles d'éclairage. La première scène est dédiée à la statue de Lucy éclairée de manière indirecte par un spot lumineux. Le sol est modélisé avec une normal map. La seconde scène est constituée de trois statues provenant de divers endroits et époques autour la mer méditerranée et d'un sol avec une normal map. Le tout est éclairé de manière indirecte par un spot lumineux orienté vers le plafond. 101
- 5.7 Salle méditerranéenne. Rendu avec APO, à gauche, comparé avec un rendu avec des atlas de textures à droite. Aucune différence n'est visible sur les statues. Remarquez les réflexions de l'éclairage indirect des murs de droite et de gauche sur les statues. De plus, l'éclairage indirect suit les variations des normales. 102
- 5.8 Pré-calcul de l'éclairage indirect avec et sans géométrie simplifiée. La résolution de la grille 3D autour de la statue est de $8 \times 16 \times 8$. En (b) et (c) nous utilisons un rendu software pour mettre en évidence l'irradiance indirecte reconstruite à l'aide des grilles en utilisant respectivement la géométrie faiblement détaillée (b) et celle détaillée (c). (d) Calcul de différence dans l'espace de couleur *Lab* entre (b) et (c). Ce calcul montre que l'erreur maximale est de 69 (15% de l'erreur maximale possible) et très localisée. L'erreur moyenne est de 0,4 (0,09%). 102
- 5.9 Éclairage indirect sur GPU. La caméra et le dragon peuvent être déplacés avec une interpolation tri-linéaire (resp. tri-cubique) à 91 fps (resp. 87 fps). 103

- 5.10 Notre système de visualisation client/serveur permet d'envoyer de manière progressive de la géométrie et de l'éclairage représenté sous forme de grilles de vecteurs d'irradiance. (a) Une scène avec 0.2% de la géométrie transférée et sans éclairage direct (aucune ombre). (b) La même quantité de géométrie mais avec 25% d'éclairage indirect : les effets de transferts de couleur entre le mur et la statue sont désormais visibles, le bouddha apparaît plus rouge. (c) Raffinement supplémentaire de la géométrie (2%). (d) Résolution maximale atteinte pour la géométrie (50 Mo) et la grille d'éclairage (1 Mo). Le taux de rafraîchissement pour une telle scène est de 50 fps avec une carte NVIDIA GeForce Go 7800 GTX . . . 104
- 5.11 Comparaison de l'éclairage indirect reconstruit sur le côté droit du dragon (a) sans l'algorithme du push-pull, (b) avec l'algorithme du push-pull et (c) avec filtrage pyramidal et sans filtrage pyramidal (c). La dimension de la grille est de $16 \times 16 \times 16$ et la moitié des sommets de la grille ont été transférés. Le push-pull a comblé les régions noires du dragon avec des valeurs d'éclairage continues. 105
- 5.12 Temps total de téléchargement pour une grille de $32 \times 32 \times 32$ avec différentes tailles de buffer. La taille de la grille 3D est de 3 Mo en utilisant (a) des nombres flottants et (b) la taille quantifiée est de 1,3 Mo. Le temps de traitement indiqué inclut l'étape de déquantification et la copie vers la mémoire CPU. 106
- 5.13 Comparaison des temps de téléchargement avec et sans filtrage lors de l'étape du push-pull pour une grille quantifiée de $32 \times 32 \times 32$. L'étape de filtrage représente 80% du temps passé lors de l'étape de *push-pull*. . 107

- 5.14 Transfert progressif de la géométrie avec une grille d'éclairage fixe ($16 \times 16 \times 16$ (563 Ko)). De gauche à droite : l'éclairage indirect s'adapte lorsque la géométrie est raffinée (utilisation d'un buffer de 250 sommets). Pour chaque image, le temps indiqué en-dessous représente le temps total requis pour atteindre la taille indiquée de géométrie dans l'image à sa gauche. Les faces orientées vers le mur rouge sont toujours rouges et les effets de couleur de l'éclairage indirect sont représentés de manière cohérente. 107
- 5.15 Transfert progressif d'une grille d'éclairage de $16 \times 16 \times 16$ (déquantifiée) pour une géométrie fixe. (a) La grille d'éclairage initiale, avec seulement les 8 vecteurs d'irradiance (iv) localisés dans les coins, est raffinée (b)-(d) avec 64 échantillons par couche et par requête. Pour chaque image, les temps indiqués en-dessous représentent le temps total pour atteindre la taille de grille atteinte dans l'image à sa gauche. 108
- 5.16 Temps de téléchargement pour la géométrie complète du bouddha de la figure 5.10 pour différentes tailles de paquets (un paquet correspond à un nombre fixe de sommets). La taille de la géométrie complète (haute résolution) est de 1,1 M polygones. Le temps de traitement inclut la mise-à-jour des indices de sommets à chaque étape de la transmission. 110
- 5.17 Scène majoritairement éclairée par un éclairage direct. 16 rayons par pixel sont utilisés. (gauche) Notre technique avec une grille d'éclairage de $40 \times 50 \times 40$ (24.5 Mo), sans compression des vecteurs d'irradiance, utilisée de manière directe avec un schéma d'interpolation tricubique. (centre) *Photon Mapping* avec le cache de Christensen utilisant 50 des 5 millions de photons (124 Mo) stockés pour pré-calculer les échantillons d'irradiance. (droite) Solution de référence obtenue en effectuant un *Path Tracing* avec 1 600 rayons par pixel. Les effets d'éclairage sont similaires entre les différentes images mais notre technique est bien plus rapide pour le calcul de l'éclairage indirect (51 s vs. 5527 s) et réduit nettement le temps total de rendu (1 341 s vs. 6 618 s). 111

- 5.18 Scène avec principalement un éclairage indirect. (gauche) Notre technique en utilisant une grille de $30 \times 20 \times 92$ (11,4 Mo), où les vecteurs d'irradiance ne sont pas compressés, construite en utilisant 80 millions de photons. (centre) Image de référence avec la méthode de Christensen en utilisant 3 200 rayons pour échantillonner l'hémisphère et 5 millions de photons. (droite) Notre technique en utilisant une grille de $12 \times 8 \times 20$ accédée de manière indirecte où le même nombre de photons et le même nombre de rayons ont été utilisés que pour l'image du centre. 113
- 5.19 Illustration sur l'objet Lucy des sommets réellement utilisés pour l'éclairage de l'objet. (a) Sommets à l'intérieur de la grille. (b) Sommets à l'extérieur de l'objet et utilisés pour le calcul d'éclairage, *i.e.*, tangents à la surface extérieure de l'objet. (c) Sommets tangents et ceux extérieurs non utilisés lors du calcul de l'éclairage. (d) Ensemble des sommets de la grille montrant le faible ratio (12,032%) de sommets réellement utiles par rapport au nombre total de 32 768 pour une grille de $32 \times 32 \times 32$. . . 113
- 6.1 (a) Les compressions des motifs lumineux réfléchis par un objet parfaitement miroir révèlent des informations sur la courbure de l'objet pour un point de vue donné. Les surfaces planaires réfléchissent de plus petites régions que les surfaces courbes. Pour la même aire, une surface courbe compresse plus les motifs lumineux qu'une surface plane. (b) Afin d'améliorer l'information sur la courbure provenant de la réflexion des motifs lumineux réfléchis par la surface, nous agrandissons la région réfléchie par la surface en modifiant la direction incidente de la lumière. 121
- 6.2 (gauche) Illustration 1D du processus de modification des directions incidentes. La direction \mathbf{l} de la lumière est (1) projetée dans l'espace stéréographique, (2) modifiée dans cet espace et (3) reprojétée sur la sphère des directions. (droite) Illustration de l'étape (2) en 2D. Notez la symétrie autour des directions \mathbf{u} et \mathbf{v} 123

- 6.3 Mise en évidence des détails en utilisant un rendu avec éclairage global à un rebond : l'*Armadillo* est rendu avec (gauche) un matériau diffus et (droite) un matériau légèrement spéculaire. Chaque côté de la figure montre la différence entre le rendu sans et avec *Light Warping*. Pour les deux cas, les paramètres sont identiques. Remarquez comment les détails de la surface sont mis en évidence de manière cohérente dans les deux cas. 126
- 6.4 Comparaison de l'Ambient Occlusion sur l'*Armadillo*. Lors du calcul de l'AO, nous modifions les directions de la lumière mettant ainsi en avant les détails de l'objet. 127
- 6.5 Comparaison de l'éclairage indirect à un rebond calculé avec 25 rayons par hémisphère. En comparant les deux images, nous constatons que le warping a pour effet d'accentuer le bruit. 127
- 6.6 (a) Illustration du descripteur géométrique révélant certains artefacts du scan (digitalisation) de l'objet. (b) Rendu avec réfraction et réflexion spéculaire parfaite sans application du warping. (c) Rendu avec warping, accentuant certains détails géométriques de l'objet. 128
- 6.7 Utilisation d'un environnement pré-filtré conjointement avec de l'Ambient Occlusion. Dans cette figure, une carte d'environnement pré-filtrée pour une BRDF lambertienne, et des valeurs d'Ambient Occlusion pré-calculées par sommet ont été utilisées. (a) Rendu sans modification de l'intensité de la radiance incidente. (b) Application de la transformation qui montre que, pour ce type de scénario, les détails géométriques de la surface sont mis en évidence. (c) Comparaison entre les deux formulations : (droite) la modification de l'intensité de la lumière incidente donne lieu à moins de bruit que (gauche) la modification de sa direction. 131
- 6.8 L'image de gauche montre la version originale du rendu (sans visibilité). Dans l'image de droite, l'adaptation d'intensité appliquée à la composante spéculaire de la BRDF permet de faire ressortir les détails de la surface autour des yeux, de la bouche et des oreilles. 133

6.9	Illustration de l'adaptation d'intensité dans le cas de la réfraction approximée avec un simple rayon interrogeant directement la carte d'environnement. La perception des yeux, des oreilles et de la bouche est nettement améliorée sur l'image de droite comparativement à l'image d'origine à gauche.	134
I.1	(a) Notion de flux. (b) Repérage de la radiance. La radiance est un scalaire valable pour un angle solide $d\boldsymbol{\omega}$ repéré par une direction $\boldsymbol{\omega}$	
I.2	Repérage hémisphérique des directions.	ii
I.3	Angle solide généré par une surface A au point x	iii
I.4	(a) Angle solide différentiel $d\boldsymbol{\omega}$ pour une direction $\boldsymbol{\omega}$. (b) Quantités en jeu pour la formulation surfacique de l'angle solide différentiel $d\boldsymbol{\omega}_x$	iv

LISTE DES TABLEAUX

3.I	Comparaisons des erreurs d'approximation (dans l'espace ISF) pour différents matériaux en utilisant la représentation 2D ou 3D de l'ISF.	60
5.I	Taux de rafraîchissement (fps) pour une taille écran de 800 × 800 pixels. Pour une scène donnée, les mêmes grilles sont utilisées pour l'éclairage indirect mais avec différentes représentations pour la géométrie de la scène. Mesures effectuées sur un processeur Intel Q6600 (4 GB de RAM) avec une carte graphique NVIDIA GeForce 8800 GTX.	100
5.II	Taille des différentes structures et fps moyen. La taille de la géométrie correspond au nombre de polygones. La taille des structures d'éclairage est faible comparée à celle de la géométrie. Nous avons utilisé une méthode stochastique avec 16000 rayons pour pré-calculer les vecteurs d'irradiance pour chaque sommet. Le taux de rafraîchissement a été évalué pour une image de 800×800 pixels. Pour une scène donnée, les mêmes grilles sont utilisées mais avec une représentation de la géométrie différente.	103
I.I	Quantités radiométriques et photométriques	v

Première partie

BRDF

CHAPITRE 1

ETAT DE L'ART

Dans ce chapitre, nous présentons le concept de BRDF (*Bidirectional Reflectance Distribution Function*) et donnons un rappel des définitions mathématiques qui lui sont associées. Parmi elles, nous détaillons la notion de paramétrisation qui est centrale pour les BRDFs (e.g. [Rus98, KM99, SA05]). Dans toute cette première partie nous considérons, par abus de langage, que le terme BRDF englobe aussi bien les fonctions de réflectance physiquement exactes (utilisées dans les simulations d'éclairage) que celles utilisées dans des domaines plus artistiques. Ces dernières sont souvent regroupées sous le terme de modèle d'éclairage (*Shading Model*). Avant de présenter nos contributions au domaine, nous donnons aussi un aperçu des différents domaines de recherche associés aux BRDFs dans le cadre de la synthèse d'images. A la suite de cette analyse, nous introduisons au chapitre 2 un nouveau modèle destiné à un rendu de type artistique. A contrario, nous introduisons, au chapitre 3, un modèle destiné avant tout à des simulations physiques des matériaux. Nous concluons cette première partie en donnant des pistes futures qui permettraient d'unifier les deux types d'usages de modèle.

1.1 Généralités

Dans les mêmes conditions d'éclairage, un mur peint avec de la peinture satinée sera plus «brillant» que lorsqu'il est peint avec une peinture mate. La fonction de réflectance, appelée BRDF, modélise cette différence de comportement de la matière par rapport à la lumière.

D'un point de vue physique, la BRDF modélise les interactions entre la lumière et la matière à l'échelle microscopique. En première approximation, pour une direction incidente de lumière, la BRDF représente la quantité d'énergie lumineuse¹ réfléchiée dans une direction d'observation. Si l'on considère la lumière comme un signal, on peut in-

¹Le lecteur soucieux d'une définition physique plus rigoureuse est invité à se référer à l'annexe I.

interpréter que la BRDF agit comme un filtre angulaire [RH01b] sur la lumière incidente (signal initial) en la transformant en lumière réfléchie (signal transformé).

En infographie basée sur le réalisme, la principale volonté est de reproduire les effets lumineux dus à de la matière présente sur des objets géométriques, observables à l'œil nu, formés à l'échelle macroscopique. Une manière de modéliser la matière à l'échelle microscopique est de considérer qu'elle est composée d'un ensemble de micro-facettes et la BRDF est alors utilisée pour modéliser le comportement «moyen» de la lumière vis-à-vis de cette micro-géométrie. La BRDF est donc reliée à la surface réelle de l'objet mais est souvent supposée uniforme sur la surface macroscopique perçue. Lorsque cette supposition est vérifiée, le matériau modélisé par la BRDF ne possède pas de détails individuellement observables.

Lorsque le matériau d'un objet, au sens physique du terme, n'est pas uniforme sur sa surface, on ne parle plus de BRDF mais de SVBRDF (*Spatially Varying BRDF*). Les images de la figure 1.1 montrent un exemple de matériau modélisé par une SVBRDF. Bien que la variation spatiale du matériau soit prise en compte, la SVBRDF reste une fonction qui se place à l'échelle microscopique et ne permet donc pas de prendre en compte des phénomènes qui surviennent à l'échelle mésoscopique² (cf. 1.2). Ces effets, situés à l'échelle mésoscopique, sont capturés par la BTF (*Bidirectional Texture Function*). Comme le montre la figure 1.2, les détails individuels tels que les plis ou les ombres générés par le matériau sur lui-même sont visibles à l'échelle mésoscopique. Un bon aperçu de la recherche sur les SVBRDFs peut être lu dans le travail de Lawrence *et al.* [LBAD⁺06] ou encore dans la thèse de McAllister [McA02]. De même, les travaux les plus importants sur les BTFs sont discutés dans quelques travaux récents [FCGH08, FH09].

La BRDF, la SVBRDF et la BTF modélisent des matériaux opaques et ne sont donc pas capables de représenter des phénomènes tels que la transparence, la diffusion sous la surface ou encore la réfraction. Il en va de même pour les phénomènes dépendants du temps ou du transfert d'énergie d'une longueur d'onde à une autre, tels que la phosphorescence (réémission de lumière dans le temps), la fluorescence (réémission de la lumière

²L'échelle mésoscopique se situe entre l'échelle macroscopique et l'échelle microscopique.

sur une longueur d'onde différente de celle initiale) ou encore la diffraction (diffusion de l'onde par des points de l'objet). En infographie, pour prendre en compte ce type de phénomène, des traitements particuliers sont donc nécessaires. Par exemple, pour traiter les phénomènes de diffraction, Stam [Sta99] introduit un modèle qui prend en compte le fait que la lumière doit être modélisée par une onde électromagnétique (plutôt qu'un rayon géométrique). Par la suite, il simplifie son modèle pour le rendu temps réel [Sta04]. Un autre exemple de tel traitement particulier afin de simuler des phénomènes de réfraction est l'approximation introduite par Wyman [Wym05]. Cette approximation de la réfraction, calculable en temps réel, montre que seuls les deux premiers rebonds à l'intérieur de l'objet réfractif sont plus importants pour notre perception du phénomène.

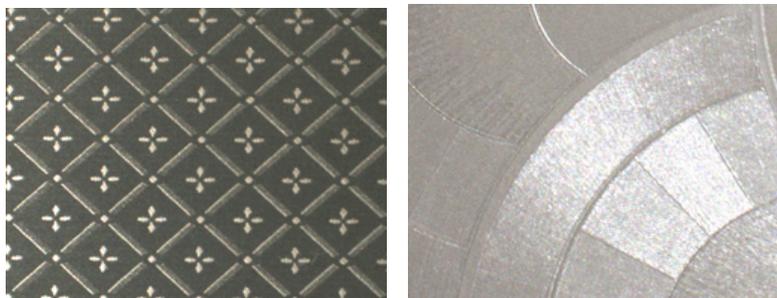


Figure 1.1 – Exemple de matériaux représentables par une BRDF variant spatialement, ou SVBRDF (*Spatially Varying BRDF*). Images tirées de [LBAD⁺06].



Figure 1.2 – Un exemple de matériau représentable par une BTF. Images tirées de [MMS⁺06].

1.2 Notations et grandeurs radiométriques

Repérage

La BRDF étant une fonction locale à la surface de l'objet, elle est exprimée dans un repère local orthonormé $(x, \mathbf{t}, \mathbf{n}, \mathbf{b})$ où :

- x le point de la surface sur laquelle la BRDF s'applique
- \mathbf{t} la tangente de la surface au point x
- \mathbf{n} la normale au point x
- \mathbf{b} la binormale obtenue par le produit vectoriel de la tangente avec la normale.

Exprimées dans ce repère, on note $\boldsymbol{\omega}_i$ la direction incidente de la lumière et $\boldsymbol{\omega}_o$ la direction d'observation (réfléchi) du point x . L'hémisphère centré en x et orienté selon

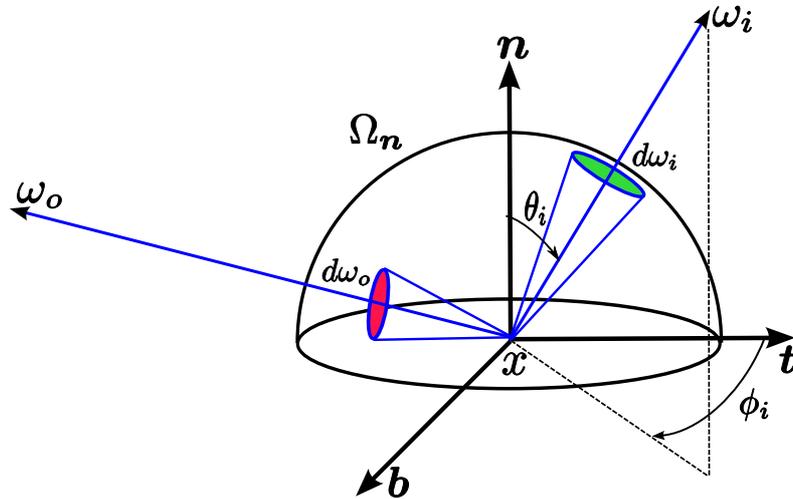


Figure 1.3 – Repérage des directions pour la BRDF. La direction incidente ω_i de la lumière et la direction d'observation ω_o sont exprimées dans le repère local (x, t, n, b) de la surface. ω_i (resp. ω_o) subtend un angle solide différentiel $d\omega_i$ (resp. $d\omega_o$) sur l'hémisphère Ω_n .

n est noté Ω_n (cf. figure 1.3).

On peut définir la BRDF, notée f_r , comme étant une fonction de ces deux directions et à valeurs non négatives :

$$f_r(\omega_i, \omega_o) : (\omega_i, \omega_o) \rightarrow \mathbb{R}_+. \quad (1.1)$$

Étant donné que la BRDF ne modélise que des matériaux opaques, seules les directions (ω_i, ω_o) contenues dans l'hémisphère orienté du côté de la normale sont pertinentes. On peut donc raffiner l'équation précédente pour obtenir le domaine de définition de la BRDF :

$$f_r(\omega_i, \omega_o) : \Omega_n \times \Omega_n \rightarrow \mathbb{R}_+. \quad (1.2)$$

En général, les directions (ω_i, ω_o) sont orientées de manière à avoir comme origine le point x .

Définition physique

Les travaux de Nicodemus *et al.* [NRH⁺77] donnent la définition de la BRDF en

fonction de quantités radiométriques élémentaires :

$$f_r(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = \frac{dL(x \rightarrow \boldsymbol{\omega}_o)}{L(x \leftarrow \boldsymbol{\omega}_i) \cos(\boldsymbol{n}, \boldsymbol{\omega}_i) d\boldsymbol{\omega}_i} \quad (1.3)$$

où $L(x \rightarrow \boldsymbol{\omega}_o)$ représente la radiance émise dans la direction $\boldsymbol{\omega}_o$ au point x tandis que $L(x \leftarrow \boldsymbol{\omega}_i)$ représente la radiance incidente au même point. La radiance est la quantité fondamentale en infographie réaliste. Elle représente la quantité d'énergie par unité de surface et de direction qui arrive ou quitte une surface : elle est exprimée en watt par mètre carré par stéradian ($Wm^{-2}sr^{-1}$). Le lecteur non familier avec les notions fondamentales de radiométrie est invité à se référer à l'annexe I. L'unité de la BRDF est l'inverse du stéradian (sr^{-1}).

Contraintes physiques

L'ensemble des fonctions définies par l'équation 1.2 contient un sous-ensemble regroupant les BRDFs dites «physiquement réalistes». Ces BRDFs sont caractérisées par deux propriétés supplémentaires : la réciprocité et la conservation de l'énergie.

1. Réciprocité d'Helmholtz :

$$f_r(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = f_r(\boldsymbol{\omega}_o, \boldsymbol{\omega}_i). \quad (1.4)$$

La réciprocité d'Helmholtz force une BRDF à être symétrique par rapport aux directions de vue et de lumière.

2. Conservation de l'énergie :

$$\forall \boldsymbol{\omega}_i, \int_{\Omega_n} f_r(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) \cos(\boldsymbol{n}, \boldsymbol{\omega}_o) d\boldsymbol{\omega}_o \leq 1 \quad (1.5)$$

où $d\boldsymbol{\omega}_o$ est l'angle solide différentiel (cf. annexe I).

On utilise une inégalité pour représenter le fait que l'énergie lumineuse incidente peut être réfléchiée, absorbée sous forme de chaleur ou encore transmise (réfraction). Lorsque la totalité de l'énergie incidente est réfléchiée, sans absorption ou transmission, l'inégalité précédente devient une égalité. La conservation de l'énergie est nécessaire

pour s'assurer de la convergence des algorithmes d'éclairage global, tandis que la réciprocity facilite les techniques d'échantillonnage basé sur l'importance (*importance sampling*).

Longueur d'onde

En toute rigueur, la BRDF dépend de la longueur d'onde λ de la lumière incidente :

$$f_r(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o, \lambda) : \Omega_n \times \Omega_n \times \mathbb{R}_+ \rightarrow \mathbb{R}_+$$

et sa valeur n'est pas forcément égale pour chacune des longueurs d'onde. En infographie, sauf cas particuliers, pour des raisons de stockage et de performance, on n'utilise que trois couleurs $(R, G, B)^3$ pour représenter l'ensemble du spectre visible de la lumière. La BRDF s'exprime alors par trois composantes $(f_r(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o, R), f_r(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o, G), f_r(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o, B))$.

La BRDF modélise non seulement un filtre énergétique mais aussi un filtre colorimétrique. Dans la suite de ce document, on omettra la notation colorimétrique lorsque cet aspect de la BRDF n'aura pas ou peu d'importance.

Isotropie et anisotropie

Lorsque la BRDF est invariante par rotation autour de la normale :

$$\forall t, f_r(\boldsymbol{\theta}_i, \phi_i; \boldsymbol{\theta}_o, \phi_o) = f_r(\boldsymbol{\theta}_i, \phi_i + t; \boldsymbol{\theta}_o, \phi_o + t) \quad (1.6)$$

on dit que la BRDF est isotrope ; dans le cas contraire, on parle de BRDF anisotrope. Une conséquence de l'isotropie est l'invariance par réflexion par rapport au plan d'incidence (cf. [SA05]) de la BRDF :

$$f_r(\boldsymbol{\theta}_i, 0; \boldsymbol{\theta}_o, \phi_o) = f_r(\boldsymbol{\theta}_i, 0; \boldsymbol{\theta}_o, -\phi_o). \quad (1.7)$$

Des exemples de BRDFs isotropes et anisotropes sont montrés à la figure 1.4. Lors-

³RGB pour *Red, Green, Blue*. Le sigle français équivalent est RVB pour Rouge, Vert, Bleu.

qu'une BRDF est isotrope, elle ne dépend donc plus que de trois paramètres :

$$f_r(\theta_i; \theta_o, |\phi_o - \phi_i|) : \mathbb{R}^3 \rightarrow \mathbb{R}_+.$$

Pour une BRDF anisotrope il est nécessaire de construire un repère local (tangente, normale, binormale) qui varie de la manière la plus cohérente possible sur l'ensemble de la surface. On cherche le plus souvent à orienter les tangentes avec les caractéristiques anisotropes de la surface. Pour une BRDF isotrope, cette contrainte n'est pas nécessaire ce qui la rend plus facile à utiliser.

1.3 Paramétrisation

La manière classique d'exprimer les directions (ω_i, ω_o) est d'utiliser les coordonnées sphériques (θ, ϕ) où θ représente l'angle co-latitudinal et ϕ l'angle azimutal (cf. figure 1.3) :

$$\begin{aligned} f_r(\theta_i, \phi_i; \theta_o, \phi_o) : \Omega_n \times \Omega_n &\rightarrow \mathbb{R}_+ \\ \theta_i, \theta_o &\in \left[0, \frac{\pi}{2}\right] \\ \phi_i, \phi_o &\in [0, 2\pi]. \end{aligned}$$

Exprimée de cette manière, la BRDF est une fonction de \mathbb{R}^4 vers \mathbb{R}_+ . La façon de repérer les directions sur l'hémisphère Ω_n est appelée «paramétrisation de la BRDF» et constitue à elle seule un domaine de recherche. Les paramétrisations les plus utilisées sont présentées à la figure 1.5.

Paramétrisation classique

La manière classique de repérer les vecteurs ω_i et ω_o est d'utiliser les coordonnées sphériques (θ, ϕ) dans le repère local de la surface (cf. figure 1.5(a)). La transformation classique de coordonnées cartésiennes en coordonnées sphériques est donnée à l'équation 1.8.

Une direction $\omega = (x, y, z)$ normalisée et repérée sur l'hémisphère Ω_n peut s'exprimer

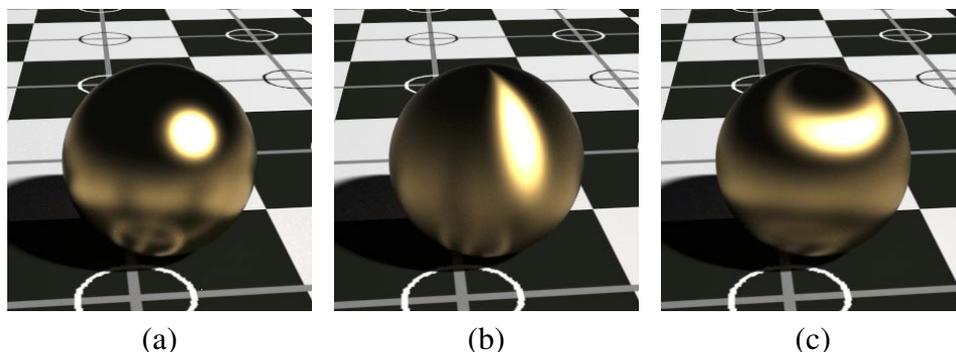


Figure 1.4 – Illustration de la propriété d’isotropie et d’anisotropie d’une BRDF. (a) BRDF isotrope, (b) BRDF anisotrope dont la direction principale d’anisotropie est verticale ou (c) horizontale. Images extraites de [AS00b].

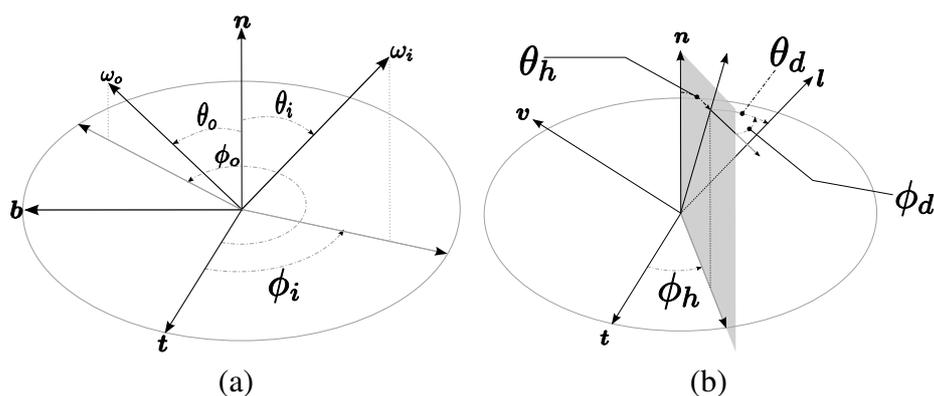


Figure 1.5 – Deux paramétrisations courantes en infographie. (a) Classique, (b) Rusinkiewicz [Rus98].

en coordonnées sphériques (θ, ϕ) en utilisant le système d’équations suivant :

$$\begin{cases} \theta = \arccos(z) \\ \phi = \arctan2(x, y) \\ \theta \in [0, \frac{\pi}{2}] \quad \phi \in [0, 2\pi] \end{cases} \quad (1.8)$$

où la direction $\boldsymbol{\omega} = x\mathbf{b} + y\mathbf{t} + z\mathbf{n}$ est définie centré en x et orienté selon \mathbf{n} dans le repère local $(\mathbf{b}, \mathbf{t}, \mathbf{n})$. Le système d’équations 1.8 met en évidence les problèmes de continuité qui surviennent lorsqu’on utilise la paramétrisation classique. En effet, l’angle ϕ n’est plus

défini lorsqu'une direction est confondue (ou numériquement proche) avec la normale de la surface, puisque (x, y) devient le vecteur nul. En revanche, l'angle θ est toujours défini. Pour tous les modèles analytiques isotropiques qui ne dépendent pas de ϕ , aucun problème ne se pose en pratique.

Paramétrisation de Rusinkiewicz

La paramétrisation de Rusinkiewicz [Rus98] est illustrée à la figure 1.5(b). Elle utilise le vecteur bissecteur \mathbf{h} et un vecteur différence \mathbf{d} définis comme suit :

$$\mathbf{h} = \frac{\boldsymbol{\omega}_i + \boldsymbol{\omega}_o}{\|\boldsymbol{\omega}_i + \boldsymbol{\omega}_o\|} \quad (1.9)$$

et \mathbf{d} est défini par :

$$\begin{cases} \mathbf{u} = -\frac{(\mathbf{n} - (\mathbf{n} \cdot \mathbf{h}) \mathbf{h})}{\|(\mathbf{n} - (\mathbf{n} \cdot \mathbf{h}) \mathbf{h})\|} \\ \mathbf{v} = \mathbf{h} \times \mathbf{u} \\ \mathbf{d} = [\boldsymbol{\omega}_i \cdot \mathbf{h}, \boldsymbol{\omega}_i \cdot \mathbf{u}, \boldsymbol{\omega}_i \cdot \mathbf{v}] \end{cases} \quad (1.10)$$

où \times est le produit vectoriel et \cdot le produit scalaire.

Dans cette paramétrisation, lorsque $\mathbf{h} = \mathbf{n}$, une discontinuité apparaît puisque \mathbf{u} devient le vecteur nul de même que \mathbf{v} . Il faut bien noter que $\mathbf{d} = \boldsymbol{\omega}_i$. En fait, il s'agit de projeter $\boldsymbol{\omega}_i$ dans le repère $(\mathbf{h}, \mathbf{u}, \mathbf{v})$. On appelle les angles (θ_h, ϕ_h) , associés au vecteur \mathbf{h} , demi-angles, tandis que les angles (θ_d, ϕ_d) associés au vecteur \mathbf{d} sont appelés angles de différence. De plus, le vecteur \mathbf{h} est très employé dans les modèles de BRDFs (e.g. [Bli77, APS00, NDM05]) parce que la paramétrisation de Rusinkiewicz décrit mieux le comportement physique de la BRDF.

Paramétrisation de Kautz et McCool

La paramétrisation proposée par Kautz et McCool [KM99] est très proche de celle de Rusinkiewicz si ce n'est que la discontinuité apparaît lorsque \mathbf{h} est rasant à la surface. La seule différence se situe au niveau de la définition de \mathbf{u} et \mathbf{v} :

$$\begin{cases} \mathbf{u} = -\frac{(\mathbf{t} - (\mathbf{t} \cdot \mathbf{h})\mathbf{h})}{\|(\mathbf{t} - (\mathbf{t} \cdot \mathbf{h})\mathbf{h})\|} \\ \mathbf{v} = \mathbf{h} \times \mathbf{u}. \end{cases} \quad (1.11)$$

La discontinuité apparaît désormais lorsque \mathbf{h} est colinéaire à \mathbf{t} , ce qui arrive dans le cas de la rétro-réflexion. Il n'y a donc aucune différence fondamentale entre la paramétrisation de Kautz et McCool [KM99] et celle de Rusinkiewicz [Rus98], si ce n'est la construction du repère $(\mathbf{h}, \mathbf{u}, \mathbf{v})$ qui ne donne pas lieu au même cas de discontinuité. Kautz et McCool justifient leur paramétrisation en rappelant que les cas de rétro-réflexion sont moins fréquents que ceux où $\mathbf{h} = \mathbf{n}$.

Cependant, la discontinuité de cette paramétrisation constitue, selon nous, une limitation dans un contexte d'édition. L'utilisateur pourrait très bien vouloir spécifier un lobe pour le cas rétro-réflexif. Remarquons que les panneaux de signalisation routière constituent un exemple simple de rétro-réflexion. Nous détaillerons cela au chapitre 2. De plus, dans le cadre de l'approximation de BRDFs, la paramétrisation de Stark *et al.* [SA05], pour les BRDFs isotropes, semble très appropriée (cf. chapitre 3). Au chapitre 2, nous introduirons une nouvelle paramétrisation pour l'édition.

1.4 Représentation de la BRDF

Dans cette section nous donnons un aperçu des différents modèles de BRDF. La littérature sur le sujet étant très vaste, il est difficile d'être exhaustif. Nous invitons le lecteur à se reporter aux différents états de l'art [Sch94a, SMSA01, DR08] pour compléter cette section.

BRDFs particulières

Il existe deux BRDFs particulières qui définissent les cas limites de comportement d'une BRDF. La BRDF lambertienne, ou parfaitement diffuse, réfléchit la lumière de manière uniforme en intensité. Une définition analytique d'une BRDF lambertienne est donnée par :

$$f_r(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = \frac{\rho_d}{\pi} \quad (1.12)$$

où ρ_d , l'albédo diffus, est une constante positive dépendante de la surface qui caractérise le pourcentage d'énergie réfléchié par rapport à l'énergie incidente. Les albédos sont sans dimension et compris entre zéro et un (*i.e.*, $\rho_d \in [0, 1]$).

A l'opposé, la BRDF qui modélise le miroir parfait correspond à celle pour laquelle la lumière incidente est réfléchié dans une, et une seule, direction :

$$f_r(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = \delta_{\boldsymbol{\omega}_r}(\boldsymbol{\omega}_o) \quad (1.13)$$

où δ représente la fonction delta de Dirac étant uniquement non-nulle dans la direction $\boldsymbol{\omega}_r$:

$$\boldsymbol{\omega}_r = (\theta_r, \phi_r) = (\theta_i, \pi + \phi_i) = -\boldsymbol{\omega}_i + 2(\boldsymbol{\omega}_i \cdot \mathbf{n})\mathbf{n}.$$

Ces deux BRDFs théoriques sont souvent utilisées conjointement pour définir de nouvelles BRDFs plus complexes comme illustré à la figure 1.6.

Modèles analytiques

On peut catégoriser les modèles suivant leur développement (empirique ou physique) ou encore selon leur paramétrisation du lobe spéculaire : autour de la direction $\boldsymbol{\omega}_r$ de la lumière réfléchié ou autour de la normale en utilisant le vecteur bissecteur $\mathbf{h} = \frac{\boldsymbol{\omega}_o + \boldsymbol{\omega}_i}{\|\boldsymbol{\omega}_o + \boldsymbol{\omega}_i\|}$.

L'un des premiers modèles développés de manière empirique et qui reste encore largement employé est celui proposé par Phong [Pho75], dont la formulation est la suivante :

$$f_r(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = \rho_d + \rho_s (\boldsymbol{\omega}_r \cdot \boldsymbol{\omega}_o)^k \quad (1.14)$$

avec :

- ρ_d albédo diffus
- ρ_s albédo spéculaire
- $k \in \mathbb{N}$ contrôle la taille du reflet spéculaire (*highlight*).

Le modèle de Phong est un exemple typique de création de BRDFs en utilisant le terme lambertien (composante diffuse) et une composante spéculaire allant de la réflexion miroir parfaite ($k = \infty$) à la réflexion diffuse uniforme ($k = 0$). Une alternative,

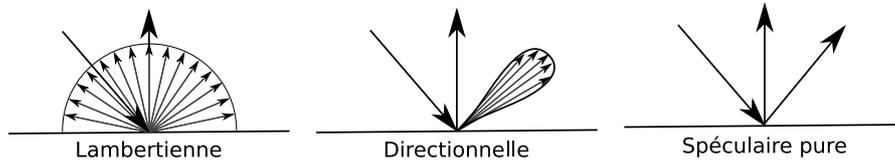


Figure 1.6 – Construction d’une BRDF complexe à partir de BRDFs simples. De gauche à droite : la BRDF lambertienne, la BRDF *glossy* (brillante) aussi appelée directionnelle et la BRDF spéculaire parfaite décrivant le comportement d’un miroir parfait.

moins coûteuse en termes de calcul et permettant de modéliser une BRDF anisotrope, a été proposée par Schlick [Sch94c] en utilisant des fonctions rationnelles pour calculer la composante spéculaire du modèle.

Le modèle de Phong ne conserve pas l’énergie ; il ne constitue pas une BRDF qui satisfait les contraintes de propriétés physiques. Lafortune et Willems [LW94] ont proposé un modèle de Phong alternatif qui satisfait les contraintes physiques grâce à une reformulation du terme ρ_s :

$$f_r(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = \frac{\rho_d}{\pi} + \rho_s \frac{k+2}{2\pi} (\boldsymbol{\omega}_r \cdot \boldsymbol{\omega}_o)^k .$$

Lafortune *et al.* ont également proposé une généralisation de ce modèle [LFTG97], dont le principe est de construire une BRDF qui possède plusieurs lobes, pour pouvoir modéliser des comportements anisotropes. D’autres modèles de BRDFs anisotropes, n’utilisant pas forcément le vecteur $\boldsymbol{\omega}_r$, ont été proposés, notamment [Kaj85, PF90, Sch94b].

Le premier modèle à paramétriser le lobe spéculaire en utilisant le bissecteur \mathbf{h} a été introduit par Blinn [Bli77] suite aux travaux de Torrance et Sparrow [TS67b]. Sa formulation est très proche de celle proposée par Phong :

$$f_r(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = \rho_d + \rho_s (\mathbf{n} \cdot \mathbf{h})^k . \quad (1.15)$$

Ward [War92] a dérivé de manière empirique un modèle où l’anisotropie est contrôlée

par deux paramètres définissant une ellipse et qui utilise également le vecteur \mathbf{h} :

$$f_r(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = \frac{\rho_d}{\pi} + \frac{\rho_s}{4\pi \alpha_x \alpha_y \sqrt{(\mathbf{n} \cdot \boldsymbol{\omega}_i)(\mathbf{n} \cdot \boldsymbol{\omega}_o)}} \exp\left(-2 \frac{\frac{(\mathbf{h} \cdot \mathbf{t})^2}{\alpha_x^2} + \frac{(\mathbf{h} \cdot \mathbf{b})^2}{\alpha_y^2}}{(1 + \mathbf{h} \cdot \mathbf{n})}\right)$$

où α_x et α_y contrôlent la forme du lobe spéculaire qui varie d'une forme elliptique à circulaire lorsque $\alpha_x = \alpha_y$.

Une observation faite par certains chercheurs montre que les albédos ρ_d et ρ_s ne sont pas constants en général et dépendent de l'angle d'incidence de la direction d'observation. A savoir, plus on observe certaines surfaces à angle rasant, plus le coefficient spéculaire devient important, tandis que le coefficient diffus diminue de manière à ce que la conservation de l'énergie soit respectée. Ce phénomène est appelé effet de Fresnel et plusieurs modèles analytiques [Sch94b, LFTG97, AS00b, APS00] en tiennent compte. L'effet de Fresnel est en général dû à la superposition d'une ou plusieurs couches semi-transparentes et dépend de l'indice de réfraction des couches. La figure 1.7 montre un exemple de matériau pour lequel l'effet de Fresnel est visuellement important.

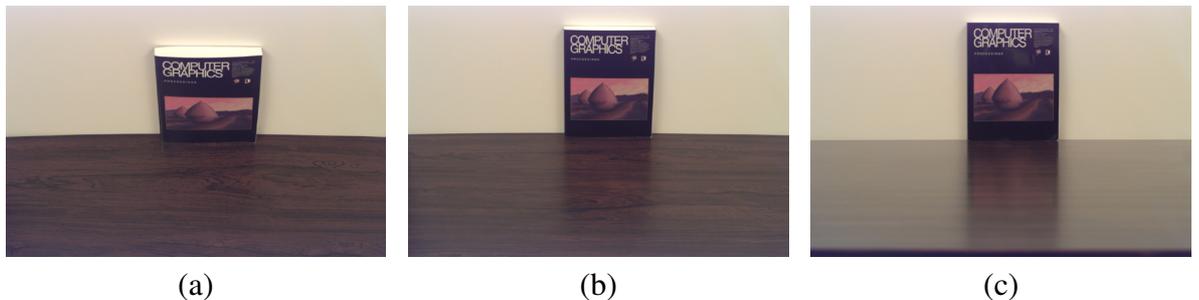


Figure 1.7 – Illustration de l'effet de Fresnel sur un matériau comme le bois ciré. (a) La table en bois vue à incidence normale, le coefficient diffus domine le coefficient spéculaire. (b) Pour une incidence à 45 degrés, les coefficients sont plus ou moins égaux. (c) A incidence rasante, le coefficient spéculaire domine. Photographies tirées de [LFTG97].

Enfin, signalons qu'un modèle diffus plus général a été introduit par Oren et Nayar [ON94] afin d'approximer le comportement d'une surface composée de micro-facettes diffuses. C'est par exemple le cas pour certains matériaux diffus plus rugueux où le modèle lambertien est inadéquat.

Parallèlement à la recherche sur les modèles empiriques, des modèles physiques ont également été développés, tels celui de Cook et Torrance [CT82a], celui de He *et al.* [HTSG91], et plus récemment celui d’Ashikhmin *et al.* [APS00]. La figure 1.8 compare l’apparence obtenue sur une sphère en utilisant le modèle de Ward [War92] et le modèle d’Ashikhmin *et al.* [APS00].

Les modèles physiques s’assurent de respecter les propriétés de réciprocité et de conservation de l’énergie. Certains de ces modèles comme celui de He *et al.* [HHP⁺92] tiennent aussi compte de la polarisation de lumière. En effet, la quasi-majorité des modèles se basent sur une modélisation de la lumière à base d’optique géométrique. Ce type de modélisation ne tient pas compte des phénomènes de polarisation qui doivent être modélisés à l’aide d’un modèle ondulatoire de la lumière. Certains matériaux, comme un diamant, constituent de bons exemples où il est nécessaire de prendre en compte la polarisation de la lumière. Le modèle proposé par He *et al.* [HHP⁺92] est certainement le modèle le plus complet. En pratique, il reste peu utilisé parce qu’il contient de nombreux paramètres moins intuitifs à contrôler pour un utilisateur non-expert.

Sans entrer dans le détail de chacun des modèles physiques, il convient de rappeler l’approche prise pour modéliser la BRDF. A l’exception du modèle de He *et al.* [HHP⁺92], les modèles physiques utilisent le concept de distribution de micro-facettes pour caractériser la surface de l’objet à l’échelle microscopique. Une micro-facette est une petite surface planaire orientée selon une normale et qui se comporte comme un miroir parfait. Une fonction de distribution des micro-facettes donne la proportion de micro-facettes qui entrent en jeu pour le calcul de la BRDF en fonction des directions de vue et de lumière. Il faut tenir compte du masquage (ou ombrage) des micro-facettes à l’aide d’un terme supplémentaire dans le modèle, soit un coefficient d’atténuation géométrique ou un terme d’ombrage. La figure 1.9 illustre le concept du terme d’ombrage. Le plus souvent, on factorise la dépendance sur les directions de vue et de lumière en utilisant uniquement le vecteur \mathbf{h} comme seul paramètre du terme d’ombrage et de la distribution de micro-facettes. Le modèle de Torrance et Sparrow [TS67a] ou celui de Cook et Torrance [CT82b] constitue un bon exemple de modèle basé sur des

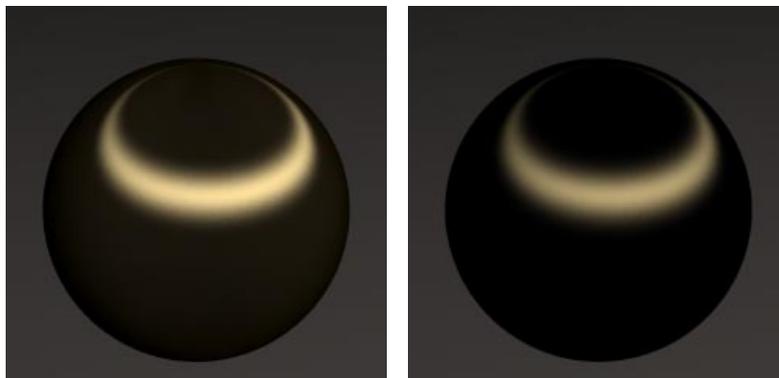


Figure 1.8 – Comparaison entre le modèle de Ward [War92] et le modèle d’Ashikhmin *et al.* [APS00]. Images tirées de [APS00].

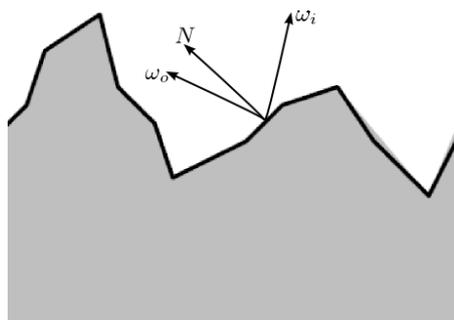


Figure 1.9 – L’effet d’ombrage des micro-facettes. En fonction de la direction de vue (ω_o) et de la direction de la lumière (ω_i), une proportion des micro-facettes n’est pas active : celles-ci sont masquées par d’autres micro-facettes.

micro-facettes (supposées lisses) et permettant de modéliser l’effet d’ombrage :

$$f_r(\omega_i, \omega_o) = \frac{F(\omega_i \cdot \mathbf{h})}{\pi} \frac{D(\mathbf{n}, \mathbf{h}, m) G(\omega_i, \omega_o)}{(\mathbf{n} \cdot \omega_i)(\mathbf{n} \cdot \omega_o)}$$

où :

- $F(\omega_i \cdot \mathbf{h})$: le terme de Fresnel qui décrit la réflectance des micro-facettes.
- $D(\mathbf{n}, \mathbf{h}, m)$: la distribution des micro-facettes qui représente la proportion de micro-facettes orientées dans la direction du vecteur \mathbf{h} .
- $G(\omega_i, \omega_o)$: le terme géométrique qui capture les effets de masquage et d’auto-ombrage.

Dans le modèle de Cook et Torrance, G et D sont définis par :

$$G(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = \min\left\{1, \frac{2(\mathbf{n} \cdot \mathbf{h})(\mathbf{n} \cdot \boldsymbol{\omega}_o)}{\boldsymbol{\omega}_o \cdot \mathbf{h}}, \frac{2(\mathbf{n} \cdot \mathbf{h})(\mathbf{n} \cdot \boldsymbol{\omega}_i)}{\boldsymbol{\omega}_o \cdot \mathbf{h}}\right\}$$

$$D(\mathbf{n}, \mathbf{h}, m) = \frac{1}{m^2(\mathbf{n} \cdot \mathbf{h})^4} e^{-\left(\frac{\tan \theta_h}{m}\right)^2}$$

avec m représentant la racine quadratique (*RMS*) de la pente. Outre le terme $\boldsymbol{\omega}_i \cdot \mathbf{h}$, le terme de Fresnel F dépend aussi de η , l'indice de réfraction, et de k , le coefficient d'atténuation. Le détail de la dérivation de F , suivant le type de matériau, peut être trouvé dans le livre de Pharr et Humphrey [PH04].

Cet aperçu ne pourrait être complet sans citer les travaux sur les modèles de BRDFs multi-couches. Les modèles multi-couches essaient de simuler l'interaction de la lumière avec plusieurs couches de matériaux en les factorisant en une seule BRDF. Par exemple, comment simuler, en une seule fonction, un matériau composé de la superposition d'une couche lisse comme du vernis et d'une couche comme du bois. Un des premiers modèles de cette famille, juste après les travaux précurseurs de Neumann et Neumann [NN89], est celui de Dorsey et Hanrahan [DH05] qui simule le vieillissement d'une surface. Ceux de Hirayama *et al.* [HKY⁺99, HYK⁺00, HKYM01] sont utilisés pour simuler les phénomènes d'interférences entre différentes couches de films et, dans certains, cas avec de la dispersion [GH03]. Enfin, récemment, Wilkie *et al.* [WWLP06] ont introduit un modèle pour simuler les surfaces diffuses et fluorescentes ainsi qu'un autre modèle [WW07] permettant de factoriser, en un seul modèle, le comportement des différentes couches de matériau.

En résumé, les deux approches de développement de modèles analytiques sont intéressantes et chaque type de modèle a sa place. Lorsqu'on souhaite modéliser un matériau où la diffusion de la lumière (*light scattering*) entre les différentes couches du matériau est négligeable, les modèles empiriques sont suffisants. Si la diffusion est une donnée importante, les modèles physiques sont plus appropriés.

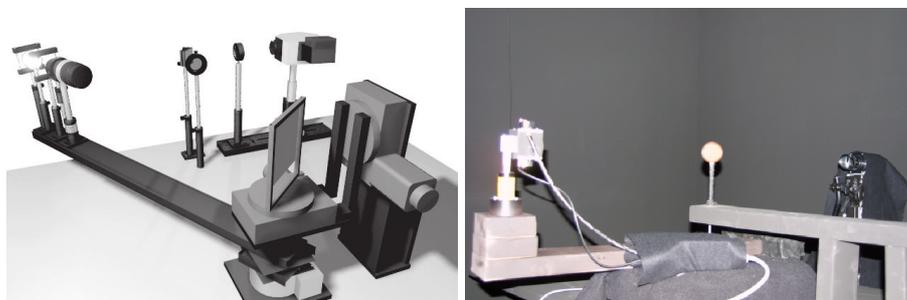


Figure 1.10 – (gauche) Exemple de gonioréfectomètre, image tirée de [LFTW05]. (droite) Dispositif proposé par Matusik *et al.* [MPBM03b].

Mesure et *fitting* de BRDF

Une voie explorée pour construire des BRDFs, redevenue populaire, est de mesurer directement la réflectance des matériaux à l'aide de dispositifs spécialisés [CMS87a, WAT92, War92, MPBM03b, GCG⁺05, MWAM05]. Le plus classique est le gonioréfectomètre [CMS87a, GTS⁺97] qui permet de faire varier la position de la source de lumière et celle d'un capteur pour acquérir des valeurs discrètes de la BRDF. La figure 1.10 montre un exemple de gonioréfectomètre. Récemment, des mesures plus complètes, en espace colorimétrique, ont aussi été faites à l'aide d'appareils photo ou de caméras numériques (e.g. [War92, DvGNK99, MWL⁺99]).

Outre le fait que l'acquisition des données de réflectance est un procédé long et fastidieux, l'un des principaux problèmes est la robustesse des mesures acquises pour les angles rasants de la surface ou encore pour les configurations où la rétro-réflexion⁴ entre en jeu. Dernièrement, Matusik *et al.* [MPBM03a] ont mesuré une centaine de BRDFs isotropes. Conjointement avec Ngan *et al.* [NDM05], un nouveau système de mesure a été développé pour mieux acquérir les BRDFs anisotropiques.

En général, la taille des données acquises est trop volumineuse pour être directement exploitables. Une manière d'utiliser ces données est d'essayer, par un mécanisme de minimisation d'erreur, de trouver un modèle analytique qui approxime les données mesurées [War92, LFTG97, MPBM03a, NDM05, CBP07]. La minimisation est majo-

⁴On parle de rétro-réflexion lorsque les directions incidente et de réflexion de la lumière sont confondues.

ritairement non-linéaire et les solveurs itératifs de type Levenberg-Marquart restent les plus utilisés. Malheureusement, en présence d'une minimisation non-linéaire, il est difficile de garantir une convergence globale : la qualité de la minimisation dépend fortement de la solution initiale donnée comme condition de départ. De leur étude récente, Ngan *et al.* [NDM05] ont conclu que le modèle d'Ashikhmin *et al.* [APS00] et la paramétrisation du lobe spéculaire utilisant le bissecteur \mathbf{h} possède de meilleures propriétés de représentation pour les algorithmes de *fitting*. Ceci est notamment dû au fait que la paramétrisation avec le vecteur \mathbf{h} permet de mieux représenter les effets de Fresnel à angles rasants. Une autre conclusion tirée par Ngan *et al.* est que, pour les matériaux isotropes, les modèles analytiques de BRDFs réussissent bien à modéliser l'apparence du phénomène alors que, pour les BRDFs anisotropes, ils échouent dans de nombreux cas à bien représenter l'apparence du matériau. Enfin, tout comme Lawrence *et al.* [LRR04], Ngan *et al.* [NDM05] constatent que l'ajout d'un lobe spéculaire au modèle analytique permet d'améliorer la qualité (environ 25%) de l'approximation mais qu'au-delà de deux lobes, le processus de minimisation devient numériquement instable et donc impraticable.

Projection dans une base de fonction directionnelle

Une autre approche pour représenter la BRDF est de la projeter dans une base de fonction. La plus utilisée à l'heure actuelle reste certainement celle des harmoniques sphériques [CMS87b, WAT92]. Ceci est notamment dû à la forte utilisation des harmoniques sphériques dans les systèmes utilisant les *Precomputed Radiance Transfers* (PRTs) [KSL05]. Nous reviendrons sur de tels systèmes à la section 4.3. Dans ce type de système, il est plus efficace de représenter la BRDF et l'éclairage incident dans la même base de fonctions. En effet, pour les harmoniques sphériques, une fois ces deux grandeurs projetées, le produit de la BRDF par l'éclairage incident revient à un simple produit scalaire. D'autres approches utilisant les ondelettes sphériques [SS95], les polynômes de Zernike [KDS96] ou encore les bases de fonctions radiales [ZERB05] ont aussi été utilisées.

Représentations à l'aide de techniques de réduction de dimensionnalité

Parallèlement, il est possible d'appliquer des méthodes classiques de réduction de dimensionnalité telles que l'Analyse en Composantes Principales [KM99, MPBM03a, VT04], la décomposition en valeur propre (SVD), et son extension, la factorisation de matrice non-négative ou *Non-negative Matrix Factorization* [LRR04, LBAD⁺06, WWHL07]. Ces techniques ont été utilisées soit pour du rendu temps réel [MAA01], soit pour utiliser des schémas d'importance ou *importance sampling* [LRR04] dans le contexte du rendu non temps réel. Ces techniques s'appliquent aussi bien à un modèle analytique qu'à un ensemble de mesures. En revanche, elles partagent toutes le même inconvénient. Elles nécessitent toutes un nombre de coefficients élevé pour représenter les BRDFs spéculaires. Mahajan *et al.* [MTR08] ont d'ailleurs montré que le nombre de coefficients varie de manière quadratique en fonction de la fréquence de la BRDF. Par conséquent, le nombre de coefficients devient rapidement trop important pour les matériaux très spéculaires.

1.5 Edition et visualisation de la BRDF

Dans cette section, nous nous intéressons à l'édition des BRDFs. Si la visualisation est une fonctionnalité requise aussi bien dans le cadre de la simulation physique que dans celui de l'infographie, le besoin d'éditer les BRDFs provient avant tout de la communauté infographique. En effet, ce sont les artistes provenant de divers horizons (dessins animés, jeux vidéo, film d'animation, etc.) qui sont les plus friands de fonctionnalités d'édition afin de contrôler au plus près le résultat qu'ils, ou que leur direction artistique, souhaitent obtenir. Les travaux d'édition et de contrôle de BRDFs peuvent être divisés en deux catégories : ceux portant sur l'édition interactive de modèles analytiques dans des conditions particulières et ceux mettant en œuvre un processus inverse pour répondre aux besoins de l'artiste. Dans le cas d'édition directe des paramètres, l'artiste doit effectuer un apprentissage, pas toujours évident, du comportement des paramètres. Cet apprentissage est facilité en amont si les paramètres du modèle développé possèdent une sémantique précise pour chacun d'eux. Dans le cas d'édition inverse, l'artiste spé-

cifie à l'aide de techniques de dessin ou d'esquisses le résultat qu'il souhaite obtenir. La difficulté pour le système est alors de comprendre et satisfaire les contraintes spécifiées. Nous commençons par donner un aperçu des techniques d'édition des modèles analytiques avant d'aborder celles d'édition inverse.

Édition de modèles analytiques/paramétriques

Les modèles paramétriques étant le plus souvent difficiles d'approche, un certain nombre de systèmes interactifs ont été proposés afin de permettre à l'utilisateur d'apprendre à mieux cerner l'influence des paramètres du modèle. L'exemple le plus probant est le système proposé par He *et al.* [HHP⁺92], censé donner une intuition sur l'action des paramètres du modèle de BRDFs proposé par les mêmes auteurs l'année précédente. Le système de Ben-Artzi *et al.* [BAOR06] propose à l'utilisateur d'éditer la BRDF d'un objet statique éclairé par une carte d'environnement (*environment map*), à la différence des systèmes de visualisation qui ne font usage que d'une seule lumière. Une représentation de la BRDF à base de courbes 1D (cf. figure 1.11) permet à l'utilisateur de manipuler les paramètres de la BRDF. En fait, l'utilisateur ne change pas les paramètres du modèle analytique sous-jacent mais une représentation de ces paramètres, et le système se charge alors de la transformation. Cela permet une manipulation linéaire de la BRDF. Le même système d'édition de BRDF est utilisé dans l'*Inverse Shade Tree* de Lawrence *et al.* [LBAD⁺06]. Ben-Artzi *et al.* [AKFR07] ont proposé une extension de leur système [BAOR06] afin d'éditer dans un contexte de lumière encore plus complexe comme les scènes d'intérieur utilisées en éclairage global. Enfin, signalons que le modèle proposé par Ashikhmin *et al.* [APS00] permet une forme limitée d'édition. La distribution des micro-facettes peut être spécifiée par l'artiste en utilisant une simple texture 2D et permet d'obtenir des lobes spéculaires de tout type (cf. figure 1.12). Malheureusement, la flexibilité laissée à l'utilisateur est limitée puisqu'il ne peut effectuer qu'une seule action : choisir sa texture qui est 2D pour spécifier la BRDF qui est 4D. De plus, il nous semble que, dès que la direction de vue change, le comportement du lobe spéculaire risque de ne plus correspondre à la texture fournie et pourrait devenir incohérent.

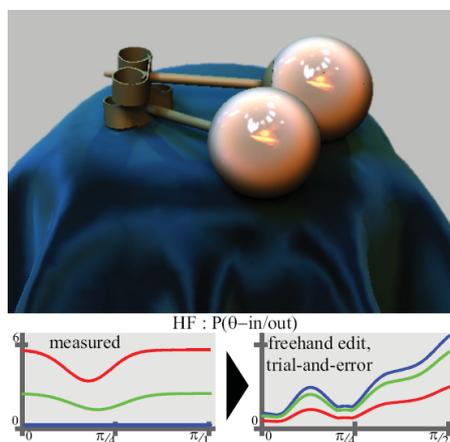


Figure 1.11 – Illustration du système d’édition, à l’aide de courbes, de BRDFs de Ben-Artzi *et al.* Image tirée de [BAOR06].

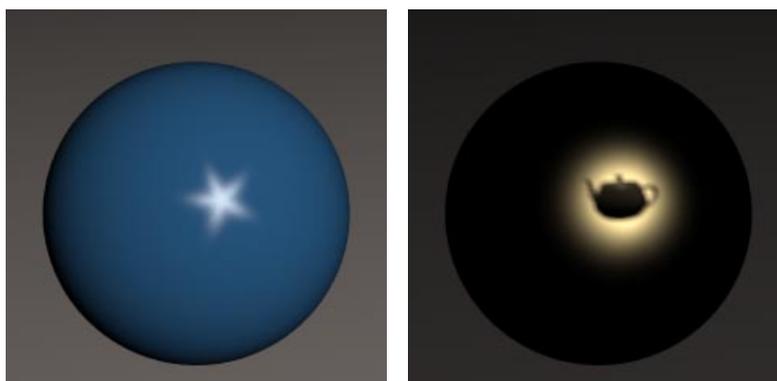


Figure 1.12 – Utilisation d’une texture pour définir la distribution des micro-facettes du modèle d’Ashikhmin *et al.* [APS00].

Edition par processus inverse

De manière générale en rendu inverse, l’utilisateur spécifie le résultat qu’il souhaite obtenir (spécification des contraintes) et le système tente de les satisfaire. Un des premiers systèmes introduit par Poulin et Fournier [PF92] et amélioré dans [PRJ97] laisse l’utilisateur dessiner les *highlights* et les ombres, et le système se charge de modifier les lumières en conséquence. Plus en rapport avec les BRDFs, le système proposé dans [PF95] laisse l’utilisateur peindre des couleurs sur la surface et le système, via une minimisation non linéaire, calcule les paramètres du modèle de Phong qui correspondent

au mieux au dessin de l'utilisateur. Reprenant la même approche, Colbert *et al.* [CPK06] ont récemment proposé de laisser l'utilisateur peindre les *highlights* d'une BRDF utilisant le modèle analytique de Ward [War92]. Un processus inverse construit un lobe tel que défini par la fonction de Ward. Afin de permettre plus de liberté de création, la BRDF n'est pas contrainte à un seul lobe mais à une somme de lobes qui respectent les contraintes de conservation d'énergie et de réciprocité. L'intérêt d'avoir une BRDF composée de plusieurs lobes est d'augmenter les possibilités de création artistique. Une capture d'écran du système (*BRDF Shop*) proposé par Colbert *et al.* est présentée à la figure 1.13. Contrairement à ce qui est réalisé par Poulin et Fournier [PF95], l'absence de processus de minimisation de l'erreur au carré permet à *BRDF Shop* d'être un système interactif.

L'une des principales limitations du système précédent est son manque de liberté dans les outils de dessin proposés. Ceci est dû à l'approche choisie, à savoir résoudre un problème inverse. Pour s'assurer que le problème inverse soit toujours résoluble, les outils d'édition sont limités à des fonctions inversibles, comme un filtre gaussien. Il en résulte que les formes de *highlights* dessinables sont moins variées. De plus, bien que l'artiste voie le résultat de son dessin sur l'objet qui l'intéresse, il ne peint pas directement dessus mais sur une sphère affichée en même temps.

Autres formes d'édition

D'autres systèmes d'édition dans le cadre du rendu non-photoréaliste existent comme le *Lit Sphere* proposé par Sloan *et al.* [SMGG01] où l'utilisateur peint également sur une sphère. Les données acquises sur la sphère sont ensuite reprojétées sur la surface 3D en prenant en compte les similarités entre le point de vue et les normales, limitant ainsi le système à un éclairage statique. Enfin, signalons deux systèmes qui proposent un paradigme de peinture sans pour autant travailler sur la BRDF. Celui proposé par Anjyo *et al.* [AWB06] laisse l'utilisateur peindre des ombres ou des reflets de manière artistique et en déduit des formes de lumière. Celui proposé par Okabe *et al.* [OMSI07] déduit la valeur d'une carte d'environnement (*environment map*) en fonction de la couleur apposée sur la surface de l'objet par l'utilisateur.

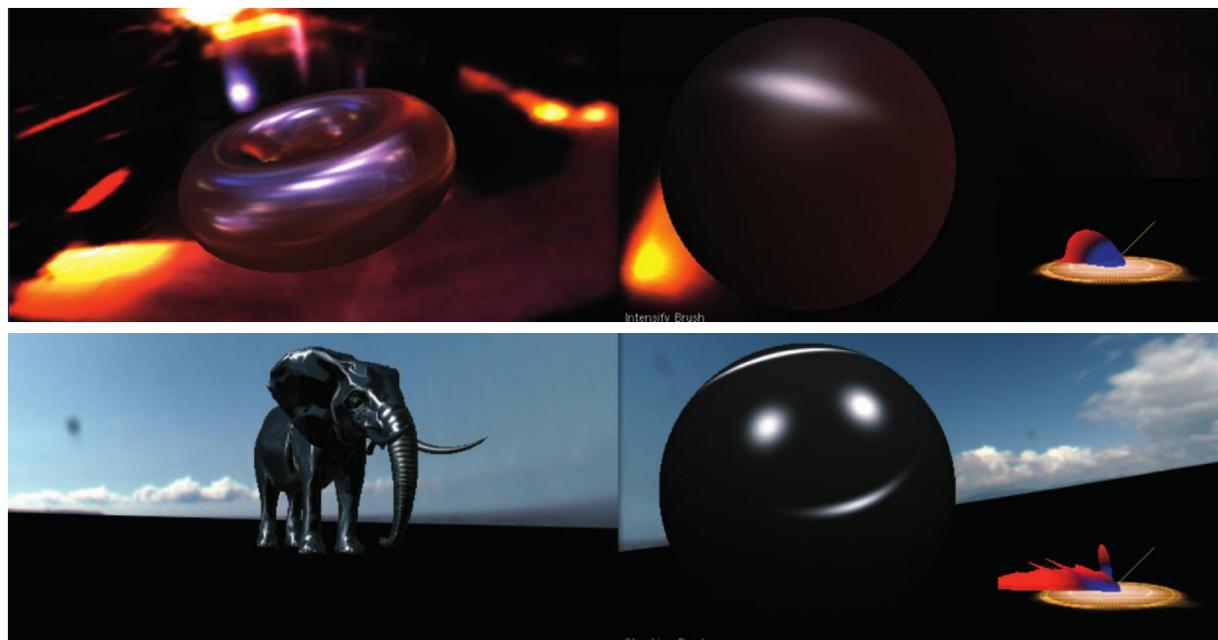


Figure 1.13 – *BRDF Shop*. Images de gauche : le résultat de l'édition de la BRDF sur un objet. Images de droite : la sphère que l'utilisateur édite pour créer sa BRDF représentée en bas à droite. En haut, une édition de BRDF avec un seul lobe. En bas, édition d'une BRDF avec trois lobes. Images tirées de [CPK06].

1.6 Conclusion

Force est de constater que la problématique de l'édition des BRDFs a été peu abordée comparativement à celle du développement de modèles analytiques ou de mesures. Les systèmes d'édition de modèles paramétriques limitent la créativité de l'utilisateur aux formes de lobes spéculaires définies par le modèle. Cependant, la précision d'édition proposée par le changement de courbes ou de valeurs est indéniable. Les systèmes inverses, qui permettent de peindre, présentent une interface d'édition qui semble nettement plus conviviale, mais restent eux aussi limités soit par la résolution inverse du problème, soit dans la diversité de création de formes. Les deux formes d'édition présentent des avantages et des inconvénients et le paradigme idéal se situe probablement à mi-chemin. Dans le chapitre suivant, nous proposons une première étape vers une représentation de la BRDF orientée utilisateur et artiste.

Comme le montre l'étude de Ngan *et al.* [NDM05], les différents modèles analytiques peinent à représenter efficacement les données mesurées. De plus, les méthodologies et techniques utilisées pour obtenir de telles représentations demeurent longues et fastidieuses. Au chapitre 3, nous proposons un nouveau modèle de BRDFs destiné à améliorer la représentation des mesures de BRDFs tout en offrant un bon pouvoir d'expression.

CHAPITRE 2

UNE NOUVELLE REPRÉSENTATION POUR L'UTILISATEUR

Dans ce chapitre, nous proposons une nouvelle représentation pour les BRDFs destinée aux artistes. Les taches spéculaires ou les reflets spéculaires (*highlights*) d'une BRDF constituent une partie importante de l'apparence de la surface d'un objet. Partant du constat que les reflets spéculaires obtenus par les modèles de Phong ou de Blinn sont des taches circulaires dont l'intensité varie de manière radiale, nous introduisons une nouvelle représentation afin que l'artiste ait plus de contrôle sur sa forme, sa couleur et son dégradé. Cette représentation est combinée à un nouveau système de d'édition et de création (*design*). Au centre de ce système se situe un plan de dessin, ou *sketching plane*, dans lequel l'artiste peut facilement esquisser et spécifier les caractéristiques du *highlight*. Cette approche a été validée par une publication [PGSP08].

2.1 Aperçu du système proposé

Le reflet spéculaire d'une BRDF peut être défini par sa puissance (*shininess*), sa couleur et sa forme. Comme on l'a vu au chapitre précédent, la plupart des modèles analytiques de BRDFs proposent peu de contrôle sur la forme ou la couleur du reflet spéculaire. Le système et la représentation proposés permettent à l'artiste de définir son reflet à l'aide d'opérations :

- d'esquisses pour en définir la forme
- de peintures et dessins pour en définir la couleur et la puissance.

Cette interface (cf. figure 2.2) est couplée à une représentation (cf. figure 2.3) qui combine une courbe pour la forme et une texture pour la couleur.

Le principe général de notre approche est illustré à la figure 2.1. L'utilisateur édite directement les caractéristiques du reflet affiché sur un plan perpendiculaire à la direction de réflexion miroir de la lumière. Sur ce plan, l'artiste peut changer la forme et la variation à l'aide d'outils d'édition 2D. En conséquence, les caractéristiques du highlight

sont définies pour une certaine direction-clé de lumière (*key-direction* par analogie à la *key-frame* utilisée en animation). L'utilisation de directions-clés est une solution assez similaire à ce qui a été proposé par Todo *et al.* [TAB107] dans un contexte d'application légèrement différent. L'ensemble des caractéristiques du reflet associées à une direction-clé est appelé configuration d'éclairage (*lighting configuration*). Un exemple typique d'usage du système est de définir le reflet spéculaire avec les outils 2D (d'abord sa forme, puis sa couleur, et son gradient comme illustré à la figure 2.3), puis ensuite d'observer son comportement pour différentes directions de lumière et de vue. Par défaut, notre système réplique les mêmes caractéristiques pour toutes les configurations d'éclairage, mais l'utilisateur peut toujours les modifier pour une direction-clé particulière. Pour toutes les autres directions de lumière et de vue, le système interpole automatiquement et de façon lisse les caractéristiques, de manière à obtenir un comportement cohérent. Enfin, le système proposé est entièrement interactif afin de donner un maximum de retour immédiat à l'artiste pendant le processus d'édition.

2.2 Interface utilisateur

Afin de fournir des outils intuitifs pour modéliser le reflet, notre approche repose sur trois types d'interaction adaptés aux différentes caractéristiques qui doivent être éditées. Pour faciliter l'édition, chaque interaction est associée à une partie bien identifiée de l'interface (cf. figure 2.2). Comme illustré à la figure 2.3, l'utilisateur peut esquisser (*sketch*) la forme, peindre la couleur ou encore éditer son gradient à l'aide d'un outil vectoriel.

Les techniques d'esquisses ou de *sketching* (e.g. [IMT99]) sont reconnues comme étant efficaces et intuitives pour définir l'aspect général d'une forme et par conséquent notre système implémente ces dernières pour les actions de création, d'édition et de modification de la forme du reflet. A l'aide de traits dans une zone spécialisée (cf. figure 2.2(a)), l'utilisateur peut sélectionner une partie ou la totalité de la forme du reflet (cf. figure 2.3(a)). Une fois la sélection effectuée, il dessine une nouvelle courbe. Un processus de minimisation essaie alors d'approximer au plus près le nouveau dessin (cf.

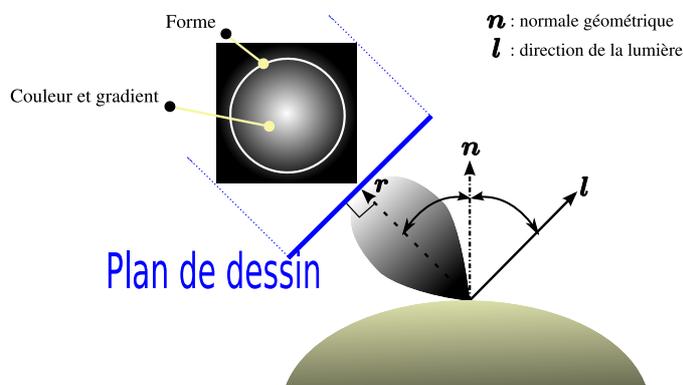


Figure 2.1 – Le plan de dessin ou *Sketching Plane*. L'utilisateur esquisse la forme du reflet spéculaire et peint sa couleur directement sur le plan de dessin qui est orienté perpendiculairement à la direction miroir (r) de la lumière (l).

figure 2.3(b)) en utilisant la représentation sous-jacente décrite à la section 2.3. La longueur de la courbe contrôle aussi la brillance du reflet spéculaire : plus la taille de la courbe est petite, plus la surface du reflet est concentrée et plus le reflet apparaît donc comme brillant. Pour augmenter (resp. diminuer) la brillance l'utilisateur a uniquement besoin de réduire (resp. augmenter) la longueur de la courbe à l'aide d'une homothétie.

Pour définir les variations de couleur et de gradient, les outils inspirés de la peinture (*painting*) constituent la meilleure approche à notre sens. Ainsi, l'artiste peut utiliser un ensemble de brosses, pinceaux et outils pour contrôler le gradient permettant aussi de raffiner la forme du highlight et d'en contrôler sa couleur. Les brosses permettent à l'utilisateur d'éditer la couleur tandis que les filtres classiques (gaussien, moyenneur, etc.) l'autorisent à ajuster l'intensité du highlight. Ces actions sont également effectuées dans des zones spécialisées : l'une pour la manipulation du gradient (cf. figure 2.2(b) et 2.3(c)), l'autre pour une édition précise du résultat final (cf. figure 2.2(a) et 2.3(d)). Nous détaillons à la section 2.3 la combinaison de la forme et de la texture de gradient. Enfin, si les outils fournis ne sont pas assez expressifs, un artiste expérimenté peut toujours charger une image créée avec un logiciel d'édition (e.g. GIMP [GIM]). Pour un système de modélisation interactif, il est crucial que l'utilisateur ait un retour sur ses actions en temps réel. Grâce à notre représentation (cf. section 2.3) chaque modification effectuée par l'utilisateur est affichée immédiatement sur un modèle géométrique 3D (cf.

figure 2.2(e)). Étant donné que le reflet dépend à la fois des directions de vue et de lumière, nous fournissons également une représentation visuelle de la direction-clé éditée par l'artiste (cf. figure 2.2(c)). De plus, l'utilisateur peut sélectionner une portion de l'objet 3D afin de récupérer de manière locale quels sont les paramètres les plus importants, telles que la couleur ou la forme autour de sa sélection. Bien que cela ne constitue pas une interface 3D de dessin, ces informations facilitent le travail de l'utilisateur quand il souhaite modifier une partie précise du reflet.

2.3 Représentation du reflet spéculaire

Le reflet spéculaire est représenté sous différentes formes qui sont combinées lors de l'étape de rendu. Pour chaque configuration d'éclairage, un champ de distance et une texture servent à définir ses caractéristiques. Le champ de distance, défini par une courbe limite, sert à représenter sa forme tandis que la texture représente ses couleurs et son gradient.

Forme du reflet spéculaire

Une observation des modèles de Phong [Pho75] et de Blinn [Bli77] nous montre que la forme des reflets produits est circulaire et leur taille est directement reliée à l'exposant. Pour le modèle de Ward [War92], la forme peut être un peu plus variée, à savoir elliptique, sans pour autant être complètement libre. Ce sont ces différentes observations qui nous ont conduits à vouloir donner à l'utilisateur un contrôle plus expressif pour la forme du reflet. Ainsi, la forme est définie par une courbe dont la taille en contrôle la brillance (*glossiness*). L'idée essentielle est d'utiliser cette courbe comme contour principal pour l'ensemble du reflet. A l'extérieur de cette courbe, l'intensité est nulle tandis qu'à l'intérieur, elle est modulée par la texture couleur. Afin de remplir l'intérieur de la forme, la représentation choisie doit permettre un accès simple et intuitif de la texture de couleur. De plus, la représentation doit également permettre une interpolation aussi lisse que possible entre les directions-clés. Ceci est nécessaire afin de pouvoir reconstruire l'ensemble des caractéristiques du reflet pour les directions n'étant pas associées à des

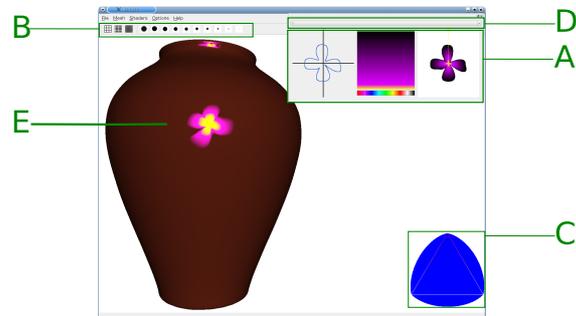


Figure 2.2 – Vue principale de l’interface utilisateur du système proposé. A : (de gauche à droite) zones d’édition de la forme, du gradient de la couleur et résultat final de l’édition dans le plan de dessin. B : outils de dessins. C : visualisation de la direction-clé couramment active. D : sélection de la direction-clé pour laquelle l’édition dans la zone A sera faite. E : résultat final sur l’objet 3D affiché en temps interactif.

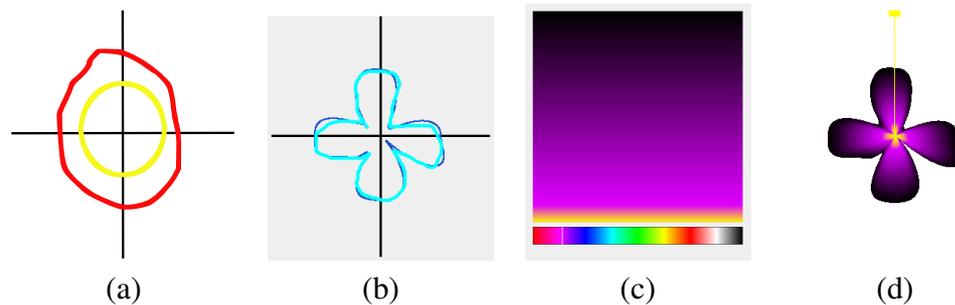


Figure 2.3 – Décomposition de l’interface proposée à l’utilisateur. (a) Sélection de l’utilisateur. (b) La courbe cyan représente l’esquisse de l’utilisateur tandis que la courbe bleue représente la courbe finale approximée par le processus de minimisation aux moindres carrés. (c) Définition d’un gradient de couleur vectoriel. (d) Vue du plan de dessin montrant le reflet résultant de (b) et de (c).

directions-clés.

Nous avons choisi d’utiliser des courbes polaires de type spline telles qu’elles sont proposées par Crespin *et al.* [CBS96]. Chaque point de contrôle de la courbe est défini par un angle θ , un rayon ρ ainsi que par des tangentes à gauche et à droite. Tous ces paramètres peuvent facilement être interpolés. Ce type de courbe offre un maximum de contrôle mais il est tout à fait possible d’utiliser des courbes analytiques pré-définies (cf. figure 2.4). De plus, associée à une fonction de distance anisotrope, cette représentation de courbe permet de définir aisément un champ de distance $d(\rho, \theta)$. Ce champ de distance a pour origine le centre du plan de dessin (cf. figure 2.5(b)) et est ensuite utilisé

pour remplir l'intérieur de la forme à l'aide de la couleur de la texture.

Texture couleur

La texture couleur représente la couleur et l'intensité du reflet à l'intérieur de sa forme. Les texels noirs représentent donc une intensité nulle. En modifiant les texels, l'artiste peut raffiner la forme qu'il a créée à l'aide de la courbe. De plus, en contrôlant le gradient de l'intensité, l'utilisateur peut en modifier la *glossiness*.

Nous utilisons deux paramétrisations (polaire et cartésienne) pour accéder à la texture couleur. De manière empirique, nous avons trouvé qu'il était plus simple de travailler en coordonnées polaires lorsqu'il s'agit de définir un gradient vectoriel de couleurs. A contrario, les coordonnées cartésiennes s'avèrent plus intéressantes pour les textures habituellement créées avec les outils de dessin. En paramétrisation polaire (cf. figure 2.3(c)), l'axe horizontal de la texture couleur représente la variation angulaire alors que l'axe vertical représente la variation radiale.

Puisque la texture couleur est délimitée par la courbe polaire qui définit une forme étoilée généralisée, l'utilisateur peut modifier la texture pour créer des formes plus complexes. Pour faciliter ce processus, nous séparons la texture couleur à l'aide de deux calques multiplicatifs comme cela est proposé dans la plupart des logiciels d'édition d'images. Le premier calque stocke la couleur alors que le second stocke son intensité en niveaux de gris.

Paramétrisation

Il nous reste à définir les paramètres ρ et θ en fonction des directions de vue et de lumière. Pour ce faire, nous introduisons une nouvelle paramétrisation de la BRDF, définie comme suit :

$$\begin{cases} \rho = \mathbf{v} \cdot \mathbf{r} \\ \cos \theta = \mathbf{v} \cdot \mathbf{x} \\ \sin \theta = \mathbf{v} \cdot \mathbf{y} \end{cases} \quad (2.1)$$

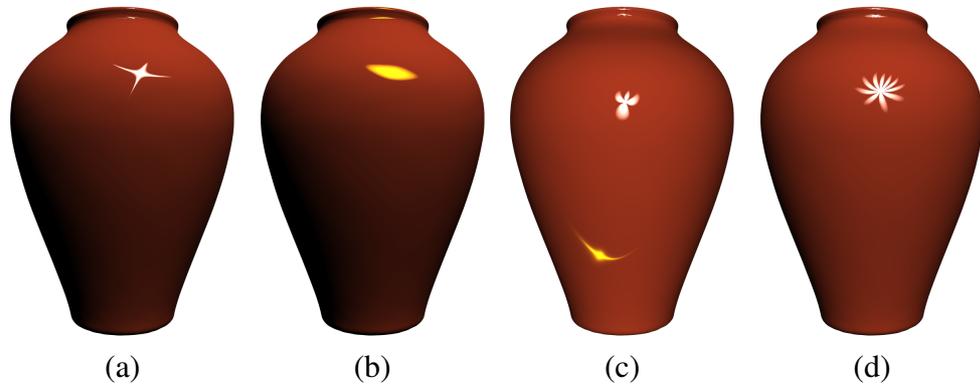


Figure 2.4 – Reflets dessinés avec un courbe de type selon (a) forme étoilée généralisée, (b) une ellipse, (c) une fonction analytique 4-pétales et une rétro-réflexion et (d) une fonction analytique 10-pétales.

où « \cdot » représente le produit scalaire, \mathbf{r} la direction miroir réfléchie de la lumière \mathbf{l} et \mathbf{v} la direction de vue. Les directions \mathbf{x} et \mathbf{y} sont définies par :

$$\begin{cases} \mathbf{x} = \mathbf{t} \times \mathbf{r} \\ \mathbf{y} = \mathbf{r} \times \mathbf{x} \end{cases} \quad (2.2)$$

où « \times » représente le produit vectoriel, et \mathbf{t} une tangente à la surface (cf. figure 2.5(a)).

Enfin, les coordonnées textures (u, v) utilisées pour accéder à la texture couleur sont fonction du champ de distance $d(\rho, \theta)$:

$$\begin{cases} u = \frac{\rho}{1 - d(\rho, \theta)} \\ v = \frac{\theta}{2\pi} \end{cases} \quad (2.3)$$

En résumé, pour une direction de lumière donnée, notre système interpole un champ de distance $d(\rho, \theta)$ et une couleur de texture tous deux utilisés lors de l'étape du rendu en fonction de la direction de vue.

Implémentation

La seule opération effectuée sur CPU est la minimisation au sens des moindres carrés qui, en fonction du trait dessiné par l'utilisateur, trouve la meilleure courbe polaire

approximante. Afin d'obtenir un système interactif, les autres opérations sont implémentées sur la carte graphique, GPU¹, à l'aide du *OpenGL Shading Language*.

La forme du reflet est stockée à l'aide d'une texture 3D. Une couche 2D de la texture 3D contient les paramètres (rayon et tangentes) d'un point de contrôle de la courbe polaire pour chacune des configurations d'éclairage. Le nombre total de couches est égal au nombre de points de contrôle de la courbe polaire. Ainsi, une couche 2D donnée paramétrise un triangle sphérique qui représente l'ensemble des configurations d'éclairage éditables par l'utilisateur. Enfin, nous concaténons l'ensemble des textures couleur (une pour chaque direction-clé) dans un tableau de textures en utilisant l'extension *OpenGL ARB_texture_array_extension*.

En pratique, nous utilisons entre 20 et 60 points de contrôle avec 3 configurations de direction de lumière. Il faut donc moins de 40 Ko pour stocker les caractéristiques du reflet et notre solution présente donc un coût mémoire faible. Ainsi, il est tout à fait envisageable de stocker plusieurs modèles directement en mémoire GPU pour les appliquer à différents objets 3D d'une même scène.

2.4 Scénarii d'utilisation et résultats

Le système a été testé sur une machine de type Intel Core 2 Duo T7500 avec une carte graphique GeForce 8600M GT. Pour une résolution écran de 1280×1024 pixels avec un objet 3D de 35 000 triangles, nous obtenons un taux de rafraîchissement de l'ordre de 60Hz. Sur le plan de la rapidité des interactions, toutes les actions prennent moins de 100ms pour mettre à jour le résultat. Pour tous les résultats présentés sur les différents objets 3D, nous combinons une composante diffuse colorée au reflet spéculaire.

La figure 2.6 présente un scénario où l'utilisateur dessine une forme de highlight de type trèfle à 4 feuilles dont la taille et le gradient de couleur varient en fonction de la direction-clé de lumière (cf. figure 2.6(d)). Lorsque la direction de la lumière est colinéaire à la normale, la texture couleur est un gradient de couleur verte ; lorsque la direction est colinéaire à la tangente (resp. bitangente), la texture couleur est un gradient

¹*Graphical Processor Unit*. Processeur présent sur la carte graphique. Cette unité de calcul est devenue facilement programmable à partir de 2002.

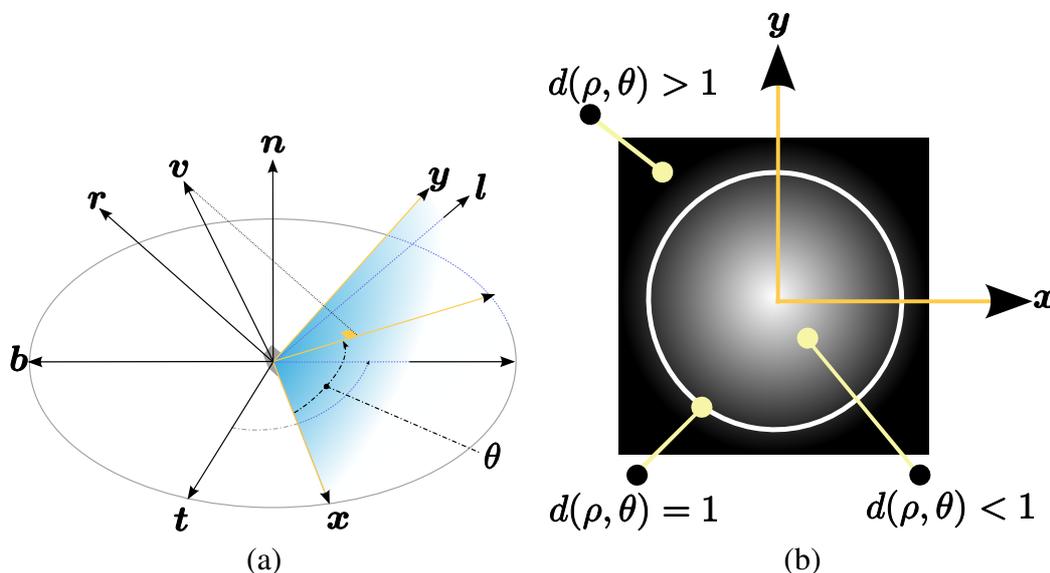


Figure 2.5 – (a) Les directions x et y forment le repère local du plan de dessin. Elles sont calculées en fonction de la direction miroir r de la lumière l et la tangente géométrique t . Afin de calculer l'intensité du reflet spéculaire, nous projetons la direction de vue v sur le plan de dessin. (b) Le champ de distance $d(\rho, \theta)$ défini par une courbe polaire paramétrise l'ensemble du plan de dessin.

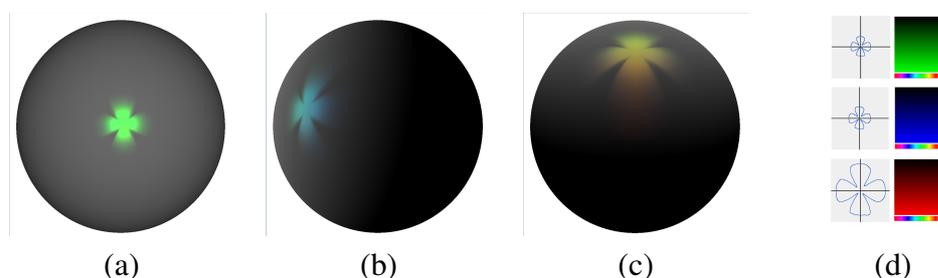


Figure 2.6 – Illustration des directions-clés. La couleur et la taille du reflet spéculaire sont définies en fonction de la direction de la lumière. Lorsque la lumière est colinéaire à la normale de la surface, (a) le reflet exhibe une couleur verte alors que, lorsque la lumière se déplace vers la direction tangentielle, (b) (resp. bitangentielle (c)) le reflet devient plus bleu (resp. rouge). Notons que pour les configurations (b) et (c), la taille du reflet est plus grande que celle en (a) afin d'augmenter l'effet de changement de couleur.

de couleur bleue (resp. rouge). Comme illustré aux figures 2.6(b) et 2.6(c) lorsque la direction de la lumière change pour se rapprocher de la tangente (resp. bitangente), le reflet spéculaire devient de plus en plus bleu (resp. rouge). La couleur jaune présente à la figure 2.6(c) provient de l'interpolation entre les couleurs rouge et verte stockées pour

deux des directions-clés. Ceci explique pourquoi la partie supérieure de la sphère, où les configurations de lumière normale et bitangente ont la même importance, est plus jaune alors que la partie inférieure, où la configuration de la bitangente est la plus importante, apparaît plus rouge.

L'utilisateur peut dessiner des formes plus complexes comme montré à la figure 2.7. A la différence de l'exemple précédent, des outils de peinture (non-vectoriels) sont utilisés pour créer une texture couleur nettement plus complexe qu'un simple dégradé. Pour ces deux derniers exemples, l'utilisateur se sert d'une paramétrisation polaire de la texture couleur.

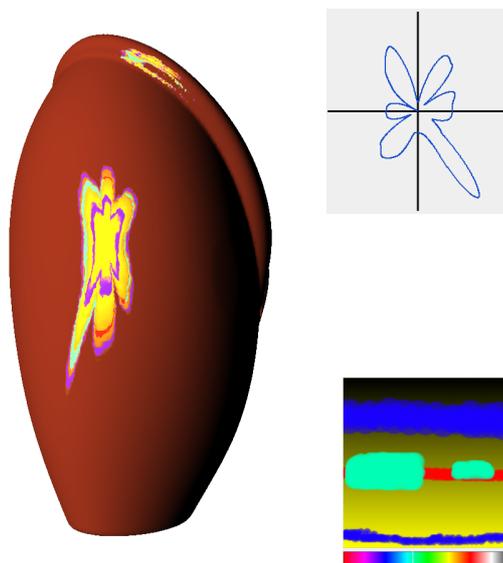


Figure 2.7 – Une forme de reflet plus complexe. La couleur du reflet a été obtenue en combinant un dégradé vectoriel (jaune vers noir) et des dessins effectués directement à l'aide de pinceaux numériques.

Avec une approche assez similaire, l'artiste peut créer des apparences assez proches de la réalité, comme l'effet de dispersion présenté à la figure 2.8. Si l'on observe d'un point de vue macroscopique les phénomènes de dispersion de couleur à la surface d'un objet, on constate que, d'une part, lorsque la lumière est colinéaire à la normale, la variation de couleur est réduite à une couleur blanche (concentration des couleurs) et, que d'autre part, lorsque la lumière est de plus en plus rasante, les couleurs se dispersent créant ainsi l'arc-en-ciel. On remarque aussi que la forme du reflet est circulaire à inci-

dence normale et qu'elle devient plutôt rectangulaire à incidence rasante. Pour obtenir cet effet de dispersion, on utilise donc deux textures couleur (en coordonnées cartésiennes) et deux courbes différentes. Pour la configuration à incidence normale de lumière, la texture est un simple dégradé blanc alors qu'à incidence rasante il s'agit d'une image représentant un dégradé de type arc-en-ciel. Enfin, à incidence normale, la courbe est un cercle, alors qu'elle est de forme rectangulaire à incidence rasante.

Les deux calques de la texture couleur sont utilisés pour raffiner la forme du reflet (cf. figures 2.9 et 2.10). Les reflets spéculaires de ces images sont modélisés à l'aide d'une texture bitmap de telle sorte que les points du plan de dessin qui tombent en dehors de la texture bitmap se voient assigner une intensité nulle. Comme la texture couleur est encapsulée par la courbe polaire, une modification de celle-ci entraîne une modification de cette texture. Ceci permet donc d'obtenir de nouveaux effets tels que celui d'un *smiley* triste à la figure 2.9(b). La bouche a été déformée suite à la transformation du cercle (cf. figure 2.9(a)) en une forme de sablier (cf. figure 2.9(b)). Enfin, la figure 2.10 présente un exemple avec des textures de couleur plus complexes. Le calque couleur est rempli en fonction de la configuration de la lumière alors que celui de l'intensité, qui représente aussi la forme, reste constant (cf. 2.10(d)).

2.5 Discussion

Le système introduit est une première étape vers un système complet de création et d'édition de BRDFs. L'une des limitations de l'approche est de dessiner ou de peindre non pas directement sur la surface de l'objet mais sur le plan de dessin. La projection de l'image du plan de dessin sur la géométrie 3D, qui n'est pas nécessairement plane, induit des déformations qui peuvent perturber la volonté de l'artiste. En revanche, comme notre système est interactif, l'artiste peut ajuster la forme du highlight en temps réel afin de compenser pour ces déformations. C'est d'ailleurs ce qui est effectué aux figures 2.9(a) et (b) afin de compenser la courbure du vase.

Malgré l'utilisation d'une interpolation lisse de la courbe polaire, les artefacts qui apparaissent sur la figure 2.10 sont dus au problème, plus général, d'interpolation entre

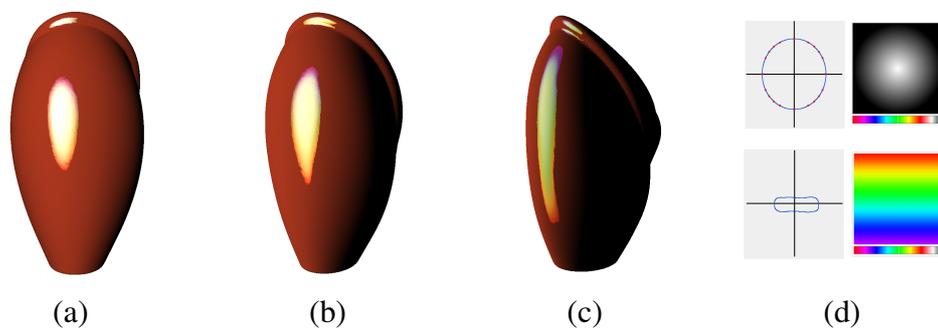


Figure 2.8 – Dispersion. (a)-(c) Le reflet exhibe un comportement simulant un effet de dispersion qui dépend de la direction de la lumière. La ligne supérieure (resp. inférieure) montre la configuration de la couleur et de la forme pour l'incidence normale (resp. rasante).

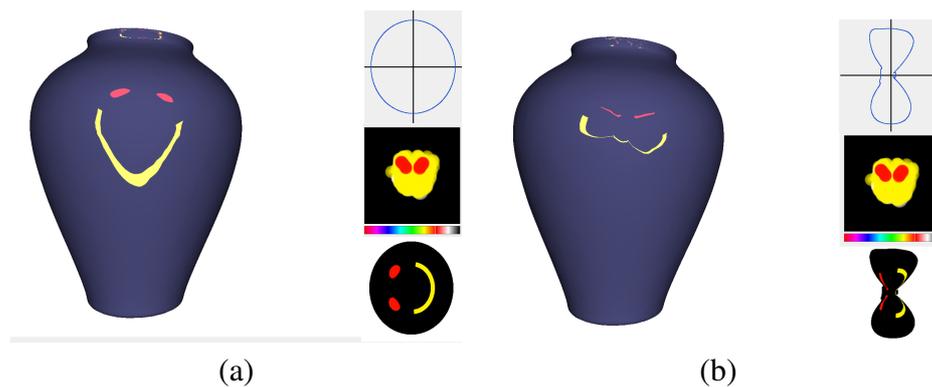


Figure 2.9 – Exemple de déformation d'une texture bitmap. (a) La forme initiale du highlight, générée à l'aide d'une texture bitmap contenant un smiley, est changée en (b) un smiley triste en modifiant la courbe polaire qui englobe cette forme.

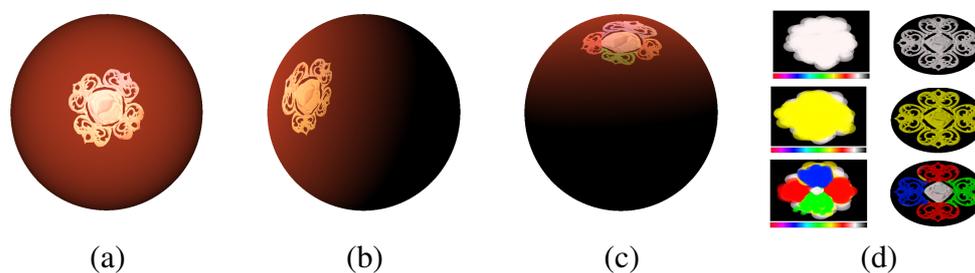


Figure 2.10 – (a)-(c) Reflet spéculaire rendu pour différentes directions de lumière. (d) La première (resp. deuxième et troisième) ligne représente la configuration d'éclairage lorsque la lumière est colinéaire avec la normale (resp. tangente et bitangente) de la surface. Toutes les configurations utilisent une forme circulaire.

deux images quelconques comme cela est fait en *morphing*. Enfin, notre système modélise un highlight comme une fonction des directions de vue et de lumière repérées de manière locale à la géométrie. Ainsi, une surface complexe qui exhibe de nombreuses variations de surface subira aussi de nombreuses variations de reflets, mais pas nécessairement sur une zone assez large pour que leur forme soit perceptible. Ceci explique pourquoi les reflets spéculaires des figures 2.9 et 2.10 présents sur le rebord supérieur du vase sont difficilement reconnaissables.

Bien qu'il puisse paraître complexe de créer un highlight à partir d'esquisses 2D plutôt que peindre directement sur la surface de l'objet, nous pensons que cet inconvénient est compensé par l'interactivité d'édition fournie par notre système. Dans tous nos exemples, nous avons utilisé uniquement trois directions de lumière ; il serait possible (dans les limites de RAM GPU) d'utiliser plus de configurations mais nous avons trouvé, de manière empirique, qu'augmenter le nombre de configurations rend le processus de création nettement plus complexe pour l'utilisateur.

2.6 Conclusion et travaux futurs

Dans ce chapitre, nous avons proposé un nouveau système de création de reflets spéculaires à base de métaphores de peinture et d'esquisses. La forme du highlight est représentée sous la forme d'une courbe polaire tandis que son gradient de couleur par une texture. La représentation proposée est efficace et adaptée à un rendu sur carte graphique permettant ainsi une édition en temps réel. Grâce à des outils simples, les possibilités d'édition et de création sont variées, peuvent même créer des effets proches du réalisme, et elles nous apparaissent comme plus puissantes que les approches précédentes. Enfin, nous pensons que notre système est une première étape vers des systèmes plus intuitifs pour contrôler le rendu et de l'apparence en général.

Les limitations du système actuel, discutées à la section 2.5, font apparaître différentes pistes pour de futurs travaux. Du point de vue de l'interface graphique, les solutions basées sur la métaphore de peinture semblent être parmi les plus intéressantes.

On peut les qualifier de solutions *WYSIWYG*². Ce type de solution possède une courbe d'apprentissage très rapide. En somme, l'utilisateur impose des contraintes au système qui tente de les satisfaire. La question primordiale, soulevée par ce besoin IHM³, est de définir sur quelle surface l'artiste peint et dans quelles conditions d'éclairage.

Pour ce faire, nous décomposons les différentes situations d'édition en fonction du type de surface et des conditions d'éclairage. Si l'on se place dans un contexte extrêmement libre, où l'artiste peut peindre sur l'objet 3D lui-même éclairé par différentes lumières, l'artiste peut effectuer des actions contradictoires. En effet, sur un objet 3D quelconque, il existe, en général, plusieurs endroits (points de recouvrements) différents sur la surface de l'objet, pour lesquels les directions de vue et de lumière sont identiques sans pour autant que l'artiste ait peint la même couleur aux mêmes endroits. Face à un tel problème, plusieurs choix s'offrent à nous. Soit le système remplit dynamiquement les zones de recouvrement afin d'éviter le problème des contradictions avec le risque de déstabiliser l'utilisateur qui constatera qu'une action de sa part peut entraîner de multiples changements. Soit la BRDF n'est plus la fonction que l'on cherche à modéliser. Dans ce cas, on cherche à modéliser une fonction plus complexe telle qu'une SVBRDF, ce qui induit de nouvelles difficultés (e.g. paramétrisation de l'objet) puisque la dimension du problème vient d'augmenter.

Une façon intéressante de ne pas augmenter la dimension du problème est de restreindre l'espace où l'utilisateur peint sur deux hémisphères, un peu comme cela a été proposé par Sloan *et al.* [SMGG01] pour leur *Lit Sphere*. De cette manière, l'utilisateur peint la BRDF sur l'espace des directions (aucune ambiguïté n'est possible). Couplée à un rendu temps réel sur l'objet 3D final auquel est associée la BRDF, cette solution nous semble très intéressante du point de vue de sa facilité d'utilisation. Pour ce faire, une nouvelle représentation, éventuellement avec la même paramétrisation que celle introduite dans ce chapitre, serait nécessaire. Il serait aussi intéressant que la représentation permette de démarrer le processus d'édition à partir d'une BRDF analytique existante afin que l'artiste puisse partir d'une solution plausible.

²WYSIWYG : What You See Is What You Get

³IHM : Interface Homme-Machine

Une autre question fondamentale est de savoir si l'utilisateur peint la BRDF ou s'il peint le résultat de la convolution de la lumière par la BRDF. Lorsque la lumière est blanche, les deux actions sont identiques, mais, dans la plupart des situations, un objet n'est pas éclairé par une lumière blanche mais par une lumière colorée. Il pourrait être utile de laisser l'utilisateur changer la chromaticité de la lumière et de le laisser peindre le résultat de la convolution.

Enfin, un objet n'est pas nécessairement éclairé par une seule lumière (e.g. environnement naturel modélisé par des cartes d'environnement), mais il nous apparaît extrêmement complexe de laisser l'artiste éditer et créer la BRDF dans un tel contexte d'éclairage. En revanche, nous pensons qu'un rendu interactif ou temps réel du résultat final est nécessaire pour guider l'artiste dans sa démarche créatrice.

Une autre ligne de recherche pertinente serait de dissocier les normales de la géométrie des normales utilisées pour le *shading*. Ceci pourrait notamment permettre de mieux percevoir les reflets spéculaires sur des zones à forte variation géométrique. De manière générale, de nouveaux outils qui permettraient de faire varier les caractéristiques du reflet de manière spatiale sur l'objet constituent un axe de recherche à explorer.

Pour conclure, il apparaît important d'effectuer un travail d'étude utilisateur afin de valider que le nouveau système proposé apporte une nette amélioration sur le plan de la facilité et l'intuitivité de ce dernier. Ce type d'étude sort du cadre de la synthèse d'images mais constitue une excellente piste de collaboration avec d'autres équipes de recherche spécialisées dans le domaine.

CHAPITRE 3

UNE REPRÉSENTATION ADAPTÉE POUR LES BRDFS ISOTROPES MESURÉES

Après avoir considéré la propriété de réflectance sous un angle créatif, nous abordons dans ce chapitre un nouveau contexte : nous nous intéressons à un nouveau modèle de BRDFs qui permet de représenter avec grande précision les matériaux mesurés. En effet, comme expliqué au chapitre 1, une représentation efficace des matériaux mesurés est nécessaire afin de pouvoir les utiliser au sein de moteurs de rendu réaliste ou de simulation d'éclairage.

3.1 Introduction

Les matériaux mesurés se présentent sous la forme de données 4D puisqu'il y a une mesure pour chaque direction de vue et de lumière : leur taille varie donc en fonction de la précision de la mesure. A titre d'exemple, chaque matériau mesuré par Matusik *et al.* [MPBM03a] occupe une place mémoire de 30 Mo. Dès lors, il apparaît rapidement difficile d'utiliser dans des scènes 3D ce genre de représentation puisque leur impact mémoire est loin d'être négligeable pour un matériau, et qu'une scène réelle en nécessite un grand nombre. Quand bien même on posséderait une machine avec une quantité de mémoire suffisante, lorsque la BRDF doit être modulée de manière spatiale, comme dans le cas d'une Spatially-Varying BRDF (SVBRDF), la solution directe n'est pas viable. De plus, à la différence des modèles analytiques, une telle représentation rend difficile l'utilisation de techniques de schéma d'importance (*importance sampling*) (e.g. [KC08]) notamment utilisées pour accélérer le rendu.

L'étude présentée par Ngan *et al.* [NDM05] démontre que les modèles de BRDF analytiques réussissent plus ou moins à représenter les matériaux isotropes alors qu'ils éprouvent des difficultés à représenter les matériaux anisotropes. La procédure d'ap-

proximation (*fitting*¹) utilisée pour déterminer la valeur des paramètres est non-linéaire et possède des inconvénients tels que l’absence de garantie d’une convergence globale ou encore le temps d’exécution de la procédure.

Dans ce chapitre, nous introduisons un modèle destiné à représenter les matériaux isotropes (cf. figure 3.1). Plus qu’un modèle de BRDFs, nous proposons une nouvelle fonction d’éclairage pour ce type de matériau : l’ISF pour *Isotropic Shading Function*. L’idée principale de l’ISF est de combiner la puissance de description des polynômes rationnels et de les exprimer dans l’espace de Stark *et al.* [SA05]. Contrairement aux approches précédentes, l’utilisation de polynômes rationnels bivariés, fonctions non-linéaires, permet d’utiliser des techniques d’approximation linéaires à convergence globale garantie. L’ISF introduit aussi une unification de la notion de lobe diffus et de lobe spéculaire. Enfin, nous montrons que l’ISF, décomposée en courbes 1D, présente également un intérêt sur le plan de l’édition.

Avant de présenter en détail l’ISF aux sections 3.3, 3.4 et 3.5, nous détaillons la paramétrisation de Stark *et al.* à la section 3.2. A la section 3.6, nous détaillons la procédure utilisée pour approximer les matériaux mesurés issus de la base de données du MERL-MIT (cf. [MPBM03b]). Nous présentons à la section 3.7 nos résultats sur le fitting des matériaux en les comparant à ceux de Ngan *et al.* [NDM05]. Cette même section présente aussi des résultats pour l’édition. Nous concluons ce chapitre par une série d’extensions pour l’ISF. A court terme, des extensions pour gérer les matériaux anisotropes et multi-couches sont introduites à la section 3.8 et pourraient être utilisées, à plus long terme, comme pierre angulaire de la factorisation de l’apparence pour les SVBRDFs et les BTFs.

3.2 Paramétrisation

Comme expliqué au chapitre 1, une BRDF est une fonction 4D, dans le cas général des matériaux anisotropes, et 3D dans le cas particulier des matériaux isotropes. Toute opération de projection dans un espace de dimension inférieure induit un recouvrement

¹De l’anglais *to fit* qui se traduit, dans notre contexte, par «approximer». Il s’agit bien ici de trouver les paramètres d’une fonction afin que cette dernière s’adapte, du mieux possible, aux données mesurées.

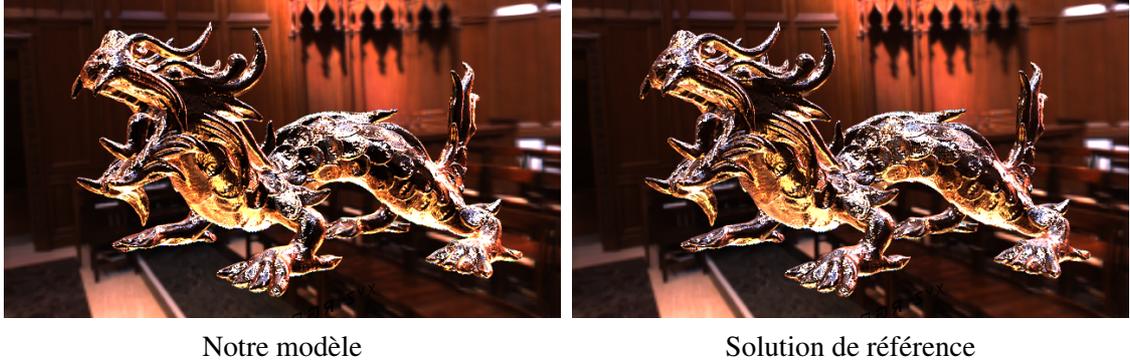


Figure 3.1 – L’*Isotropic Shading Function* (ISF) permet d’approximer des matériaux isotropes du monde réel. Cette figure montre le résultat obtenu en approxinant le nickel extrait de la base de BRDFs MERL-MIT avec seulement deux polynômes rationnels 1D.

de l’espace de départ sur l’espace d’arrivée. Formellement, on étudie les classes d’équivalence entre les deux espaces pour caractériser ce recouvrement. Dans le cas des matériaux isotropes, la projection de l’espace 4D dans l’espace 3D s’effectue sans introduire d’artefacts car il suffit de moyenner les valeurs des configurations identiques par rotation autour de la normale.

Stark *et al.* [SA05] ont montré que l’on pouvait réduire encore plus la dimension d’une BRDF isotrope à un espace à deux dimensions et ce, sans perdre le pouvoir d’expression de l’espace 3D original. La paramétrisation introduite par Stark *et al.* [SA05] permet de réduire l’espace 3D des BRDFs isotropes en espace 2D. D’après leur étude des classes d’équivalence, la projection induite par la paramétrisation proposée n’induit pas d’artefacts visuels majeurs. Parmi les trois paramétrisations proposées, nous choisissons d’utiliser la deuxième, qui est celle qui correspond le plus aux paramétrisations courantes.

La deuxième paramétrisation (r, s, t) de Stark *et al.* est définie par :

$$\begin{cases} r = (\mathbf{n} \cdot \mathbf{h})^2 - (\mathbf{n} \cdot \mathbf{l})(\mathbf{n} \cdot \mathbf{v}) \\ s = 1 - (\mathbf{n} \cdot \mathbf{h})^2 \\ t = (\mathbf{n} \cdot \mathbf{l})(\mathbf{n} \cdot \mathbf{v}) \end{cases} \quad \text{avec } (r, s, t) \in [0, 1]^3 \text{ et } r + s + t = 1$$

avec \mathbf{l} , la direction de la lumière, \mathbf{v} , la direction de vue et \mathbf{h} le bissecteur de \mathbf{l} et de \mathbf{v}

(i.e., $\mathbf{h} = \frac{\mathbf{l}+\mathbf{v}}{\|\mathbf{l}+\mathbf{v}\|}$).

Cette paramétrisation a l'avantage de pouvoir représenter une BRDF isotrope, fonction à trois dimensions ($\mathbb{R}^3 \rightarrow \mathbb{R}$), avec une fonction à deux dimensions ($\mathbb{R}^2 \rightarrow \mathbb{R}$). Notons que (r, s, t) forment les coordonnées barycentriques d'un triangle. Leurs principales propriétés sont :

- $t = 1$: les directions de la lumière et de la vue sont colinéaires à la normale.
- $t = 0$: la direction de la lumière ou la direction de la vue est perpendiculaire à la normale (i.e., rasant à la surface de l'objet).
- $s = 0$: le vecteur \mathbf{h} est colinéaire à la normale. La réflexion spéculaire est alors maximale.
- $s = 1$: les directions de la lumière et de la vue sont colinéaires et perpendiculaires à la normale.
- $r = 0$: les directions de la lumière, de la vue et du bissecteur sont colinéaires. Le matériau est rétro-reflectif.
- $r = 1$: les directions de la lumière et de vue sont perpendiculaires à la normale. De plus, elles sont opposées de manière à ce que le vecteur \mathbf{h} soit colinéaire à la normale.

Afin de faciliter la compréhension de cette paramétrisation, nous avons tracé à la figure 3.2 la surface 2D correspondant à la projection de quelques modèles analytiques dans cet espace. Pour améliorer la lisibilité du tracé, nous avons quantifié sur 32 valeurs les variables $r \in [0, 1]$ et $s \in [0, 1]$. Pour chacun des 32 intervalles, nous calculons la moyenne des valeurs contenues dans l'intervalle et affichons l'histogramme correspondant. L'utilisation de la valeur moyenne est justifiée par le fait que l'on doit calculer une valeur représentative de la classe d'équivalence.

Comme mentionné par Stark *et al.* [SA05] la paramétrisation (r, s) offre par construction des avantages indéniables. Premièrement, comme les directions de vue et de lumière jouent un rôle symétrique, la BRDF résultante satisfait nécessairement la condition physique de réciprocité. Deuxièmement, certains phénomènes physiques classiques de réflexion de la lumière peuvent être facilement et clairement identifiés sur la surface (r, s) :

- La réflexion spéculaire parfaite (miroir) se trouve sur l'axe $s = 0$.

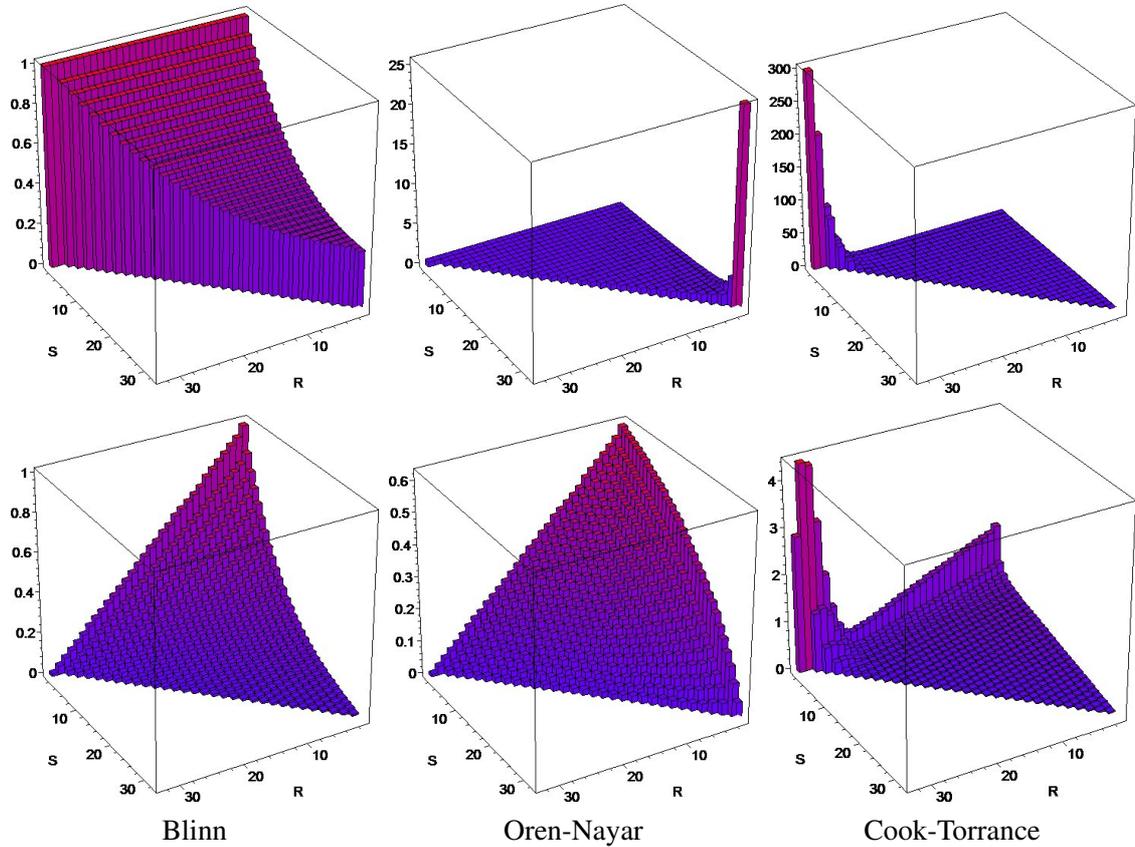


Figure 3.2 – Trois modèles analytiques de BRDFs : (haut) exprimés dans la paramétrisation (r,s) ; (bas) exprimés dans l'espace de l'ISF.

- Les phénomènes de rétro-réflexion et de rétro-diffusion sont localisés le long de l'axe $r = 0$.
- Le point $(r,s) = (0,0)$ correspond à la configuration particulière où $\mathbf{n} = \mathbf{v} = \mathbf{l}$.
- La specularité du lobe principal (resp. rétro-réflexif) est liée au gradient de la surface le long de l'axe $s = 0$ (resp. $r = 0$).
- Les effets de Fresnel observés à angle rasant sont localisés au voisinage du point $(r,s) = (1,0)$.
- La droite $r + s = 1$ correspond à l'ensemble des configurations difficiles à acquérir, aussi bien du point de vue de la théorie de la réflexion de la lumière que du point de vue des dispositifs physiques d'acquisition.

Malheureusement, toutes ces bonnes propriétés ne résolvent pas un problème pra-

tique majeur partagé par l'ensemble des représentations de BRDFs : du fait de la conversion de l'énergie, les valeurs des BRDFs deviennent très élevées au centre du lobe spéculaire ainsi que pour les directions rasantes. En conséquence, la dynamique de la BRDF peut être très grande (plusieurs ordres de magnitude même pour les matériaux de basses fréquences), ce qui rend le processus de fitting très difficile voire impraticable. Une solution standard pour améliorer la qualité de l'approximation est d'appliquer un filtre (réversible) aux valeurs afin de réduire cette dynamique. Le processus d'approximation est alors effectué sur les valeurs à faible dynamique et, lors de l'utilisation de la fonction approximée on applique alors le filtre inverse. Ceci a notamment été fait par McCool *et al.* [MAA01] à l'aide d'une fonction logarithmique ou encore avec une racine cubique [NDM05].

3.3 *Isotropic Shading Function*

Les approches de type filtrage précédentes sont arbitraires dans le contexte des BRDFs. Nous introduisons une transformation qui nous semble plus pertinente et cohérente avec la théorie des micro-facettes. Cette transformation est la suivante :

$$\sigma(r, s) = (1 - r - s) * \rho(r, s)$$

où $\rho(r, s)$ correspond à la BRDF transformée dans l'espace de Stark *et al.* En d'autres termes, nous filtrons la BRDF par le paramètre $(1 - r - s)$ de la paramétrisation de Stark *et al.* [SA05]. Nous appelons $\sigma(r, s)$ l'*Isotropic Shading Function* (ISF). Elle ne correspond pas à une BRDF mais elle décrit tout de même le *shading* d'un point sur la surface d'un objet. Comme illustré au bas de la figure 3.2, lorsqu'on projette les modèles analytiques dans l'espace ISF, la dynamique des valeurs est grandement réduite. De plus, dans les configurations difficiles ($r + s = 1$) de mesure et aussi là où la théorie des micro-facettes atteint ses limites, à cause de la discontinuité induite à angle rasant, l'ISF contourne le problème en devenant égale à zéro. La figure 3.3 (resp. 3.4) met en évidence comment l'espace ISF représente l'effet de Fresnel pour une BRDF lambertienne (resp. spéculaire). La figure 3.4 montre l'effet d'une augmentation de l'exposant

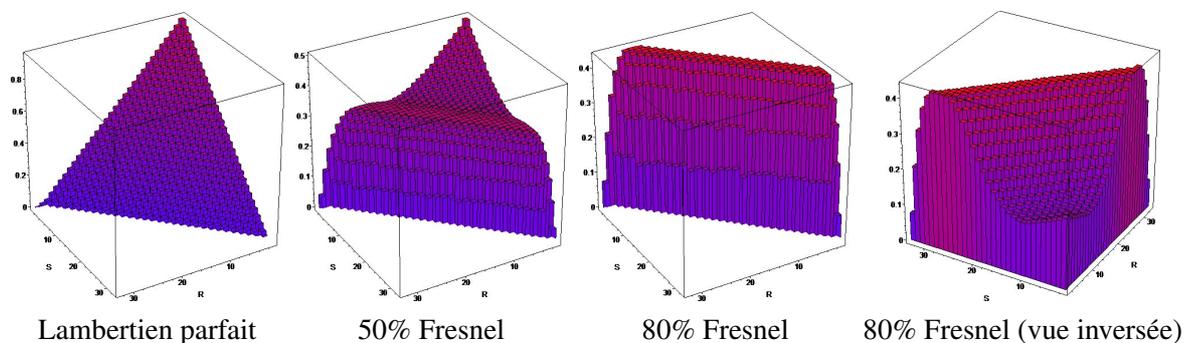


Figure 3.3 – Une BRDF lambertienne représentée dans l’espace ISF. L’approximation de Schlick [Sch93a] a été utilisée pour modéliser l’effet de Fresnel.

spéculaire d’un modèle de Blinn [Bli77] où la surface 2D décroît de plus en plus en vite à partir de la courbe $s = 0$.

L’espace ISF n’est pas seulement intéressant pour comprendre les modèles analytiques ; il permet aussi d’éclairer notre compréhension des matériaux mesurés dans le monde réel. Mise à part la base de données du laboratoire de Cornell [Cor], la base de données de BRDFs isotropes la plus complète est celle du MERL-MIT [MPBM03b] qui contient plus d’un milliard de mesures. Parmi ces 100 matériaux, nous avons choisi d’extraire neuf caractéristiques représentant un large spectre de phénomènes physiques. La figure 3.5 présente la projection de ces matériaux dans l’espace ISF. Afin d’obtenir un seul histogramme par figure, et bien que les mesures aient été effectuées en RGB, nous affichons la moyenne des trois canaux. Tout comme à la figure 3.2, les propriétés physiques des matériaux (réflexion diffuse, brillance et effet de Fresnel) peuvent être identifiées en étudiant les histogrammes.

Enfin, une autre manière de représenter l’ISF d’un matériau donné est de visualiser les courbes 1D σ_r et σ_s pour les axes $s = 0$ et $r = 0$, où se concentrent les phénomènes les plus importants (cf. figure 3.6). Cette visualisation, où l’on affiche une courbe par composante RGB, permet également d’identifier facilement les matériaux qui présentent un comportement dépendant de la longueur d’onde. Par exemple à la figure 3.6, la courbe représentant le canal bleu du matériau *metallic-blue* ne possède pas du tout le même comportement que les courbes R et G. Ceci met aussi en évidence le fait que représenter une BRDF par un triplet (R, G, B) et une fonction modélisant seulement les variations de

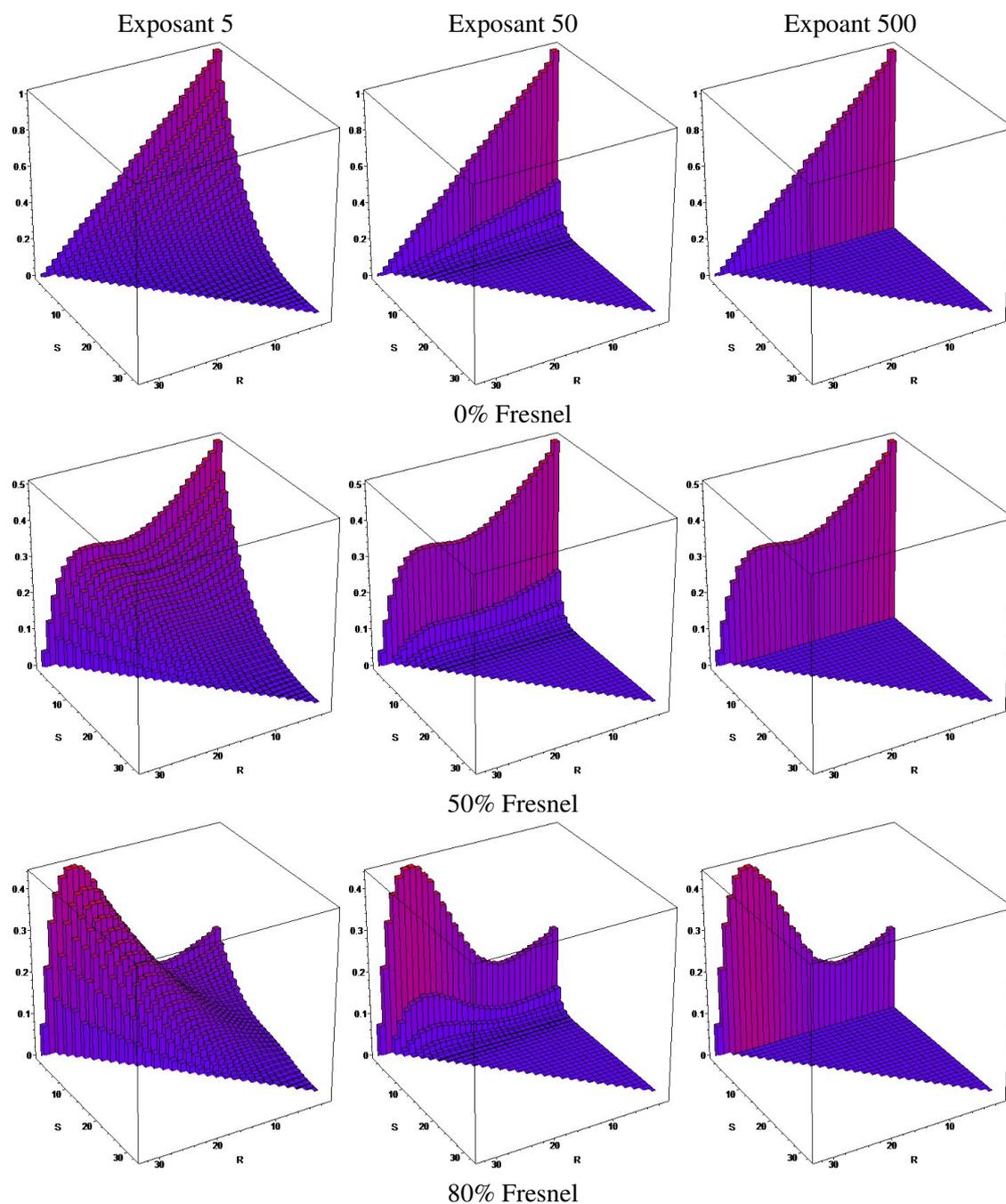


Figure 3.4 – Projection de la BRDF de Blinn dans l’espace ISF. Le coefficient de Fresnel est constant sur une même ligne alors que la rugosité de la spécularité augmente de gauche à droite. L’effet de Fresnel relève la surface à l’approche du point $(s, r) = (1, 0)$. L’augmentation de la spécularité du matériau provoque l’aplatissement de la surface, au point qu’elle se réduit pratiquement à une variation le long de l’axe s .

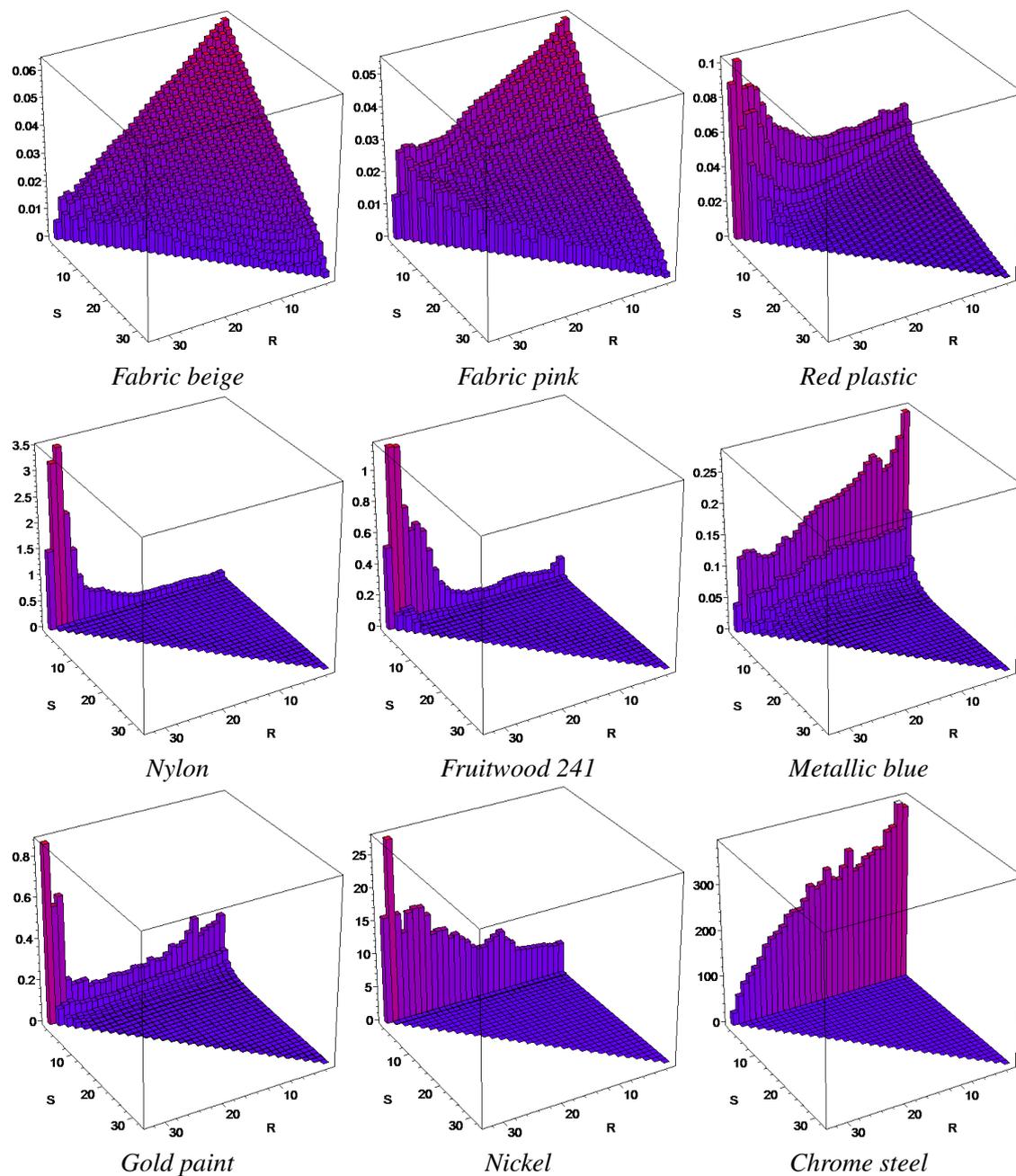


Figure 3.5 – Visualisation dans l'espace ISF de neuf matériaux extraits de la base de BRDFs MERL-MIT.

luminance n'est pas une approche valide pour certains matériaux.

Une autre information importante mise en évidence par les différents histogrammes de la figure 3.5 est la relative simplicité des surfaces obtenues dans l'espace ISF. Il est

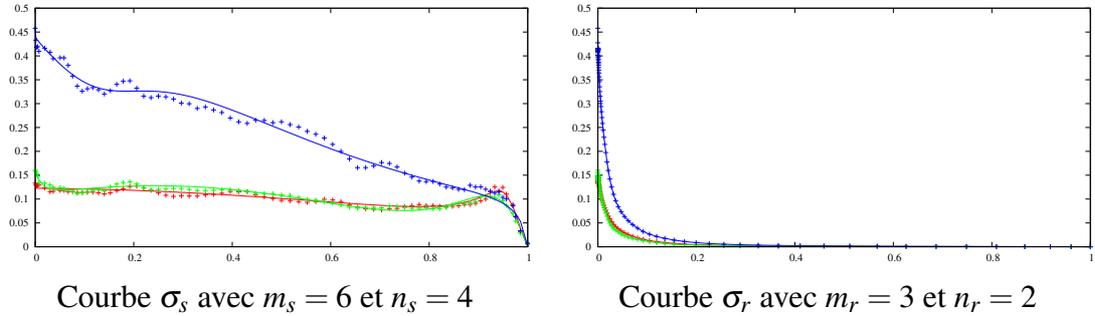


Figure 3.6 – Résultat de la procédure d’approximation sur les courbes 1D pour le matériau metallic-blue. Les mesures sont affichées à l’aide de croix.

donc tentant de vouloir approximer ces mesures par une fonction analytique. De plus, on remarque qu’un matériau diffus, représenté par un simple plan, peut tout à fait être représenté par une fonction 2D tout comme les matériaux les plus spéculaires. Ceci nous a conduits à proposer une représentation de lobes qui unifie la composante diffuse et spéculaire. Cette distinction qui est faite d’ordinaire n’a pas de justification physique excepté dans le cas des matériaux multi-couches. Dans la section suivante, nous introduisons la formulation analytique de l’ISF qui est assez flexible pour représenter les différents matériaux.

3.4 Formulation analytique de l’ISF

Polynôme rationnel bi-varié

Nous nous proposons de modéliser l’ISF comme un simple polynôme rationnel bi-varié $\sigma(r,s)$ qui prend en compte l’ensemble des phénomènes lumineux, d’ordinaire séparés en plusieurs lobes :

$$\sigma(r,s) = \frac{\sum_{i=1}^M p_i b_i(r,s)}{\sum_{j=1}^N q_j b_j(r,s)}$$

avec $M = \frac{(m+1)(m+2)}{2}$ et $N = \frac{(n+1)(n+2)}{2}$, et où m (resp. n) représente le degré du numérateur (resp. dénominateur). Les p_i (resp. q_i) représentent les coefficients réels du numérateur (resp. dénominateur) tandis que les $b_*(r,s)$ représentent une base polyno-

miale (e.g. monomiale, Bernstein, Chebyshev, etc.). Par conséquent, pour un polynôme rationnel de degrés m sur n , le nombre de coefficients requis est de $M + N$.

L'un des avantages des polynômes rationnels est qu'il faut moins de coefficients pour modéliser les hautes fréquences, par rapport aux polynômes classiques ou des bases trigonométriques. Un autre avantage est la possibilité d'utiliser des procédures d'approximation à convergence globale garantie à la différence d'autres fonctions non-linéaires telles que l'exponentielle. Nous détaillerons à la section 3.6 les algorithmes d'approximation pour polynômes rationnels que nous avons utilisés.

Interpolation à l'aide de courbes limites

Nous avons mentionné précédemment que les courbes $s = 0$ et $r = 0$ de l'espace ISF semblent contenir les caractéristiques les plus importantes de la BRDF. Afin de réduire le processus d'approximation à un processus 1D au lieu de 2D, nous nous proposons d'approximer ces courbes limites et de reconstruire une surface interpolante à partir de celles-ci. De plus, ces courbes fournissent un contrôle simplifié pour l'utilisateur (cf. section 3.7).

On note $\sigma_r(r) = \sigma(r, 0)$ et $\sigma_s(s) = \sigma(0, s)$. Une fois ces deux courbes approximées, il est nécessaire de reconstruire l'ISF pour n'importe quel couple (r, s) . Après plusieurs tests, nous avons trouvé que l'expression ci-dessous donne de très bons résultats pour un grand nombre de matériaux :

$$\bar{\sigma}(r, s) = \frac{\sigma_s(s) \cdot \sigma_r(r/(1-s))}{\sigma(0, 0)}.$$

Afin d'assurer une continuité dans la surface reconstruite, les contraintes nécessaires pour les courbes limites sont :

$$\sigma_r(0) = \sigma_s(0) = \sigma(0, 0) \quad \text{et} \quad \sigma_r(1) = \sigma_s(1) = 0.$$

Ceci garantit que $\bar{\sigma}(0, 0) = \sigma(0, 0)$ et $\bar{\sigma}(r, s) = 0$ pour $r + s = 1$ (configurations rasantes). L'idée derrière ce schéma interpolant est de laisser glisser la courbe de l'axe $r = 0$ le long de l'axe $s = 0$ tout en la contractant. La façon dont est gérée la contraction est donnée

par le terme $r/(1-s)$ qui pourrait être simplifié à l'extrême par simplement r et nous aurions alors une fonction séparable. Pour ce même terme, d'autres formulations moins brutales sont possibles :

$$\sigma_r \left(\frac{r}{(1-s)^2} \right) \quad \text{ou} \quad \sigma_r \left(\frac{r}{(1-s)^3} \right) \quad \text{ou} \quad \sigma_r \left(\frac{r}{(1-s)^4} \right).$$

Enfin, si l'on compare la représentation à base de courbes limites à celle du polynôme rationnel bi-varié, celle-ci ne nécessite que $m_s + n_s + m_r + n_r + 4$ coefficients, où m_s et n_s (resp. m_r et n_r) représentent les degrés choisis pour σ_s (resp. σ_r).

3.5 Utilisation pratique de l'ISF

A l'étape du rendu, il est nécessaire d'évaluer, entre autres, la BRDF ou la BRDF multipliée par le facteur cosinus. La relation liant la BRDF à l'ISF :

$$\rho(r, s) = \frac{\sigma(r, s)}{1 - r - s}$$

produit des instabilités numériques lorsque $(1 - r - s) \simeq 0$. Ces instabilités numériques existent dans la plupart des modèles de BRDF physiquement corrects (e.g. [HTSG91, War92, EBJ⁺06, AP07]) et sont généralement résolues par une simple troncature ou encore en remplaçant le dénominateur par un développement de Taylor afin d'assurer un meilleur comportement (e.g. [War92, AP07]). Nous choisissons de réduire ces instabilités en introduisant une fonction de continuité C^1 , $\gamma_\varepsilon(x)$, qui remplace la division $1/x$ et qui décroît vers zéro lorsque $x < \varepsilon$:

$$\gamma_\varepsilon(x) = \frac{1}{\varepsilon} \begin{cases} 3(x/\varepsilon) - 2(x/\varepsilon)^2 & x \in [0, \varepsilon[\\ \varepsilon/x & x \in [\varepsilon, 1]. \end{cases}$$

De cette manière, on peut évaluer la BRDF de manière robuste :

$$\rho(r, s) \simeq \sigma(r, s) \gamma_\varepsilon(1 - r - s). \quad (3.1)$$

Il est aussi possible d'évaluer la BRDF multipliée par le facteur cosinus, *i.e.*, $\rho(r,s) \cos \theta_l$ en utilisant une formulation simplifiée. Pour ce faire, remarquons que $1 - r - s = \cos \theta_l \cos \theta_v$, ce qui permet d'obtenir :

$$\rho(r,s) \cos \theta_l \simeq \sigma(r,s) \gamma_{\epsilon}(\cos \theta_v). \quad (3.2)$$

Lors de nos expérimentations, nous avons opté pour cette deuxième formulation. Elle introduit peu d'erreurs (et uniquement sur les silhouettes qui sont assombries), comme le montre la figure 3.8(h). Enfin, la formulation proposée aux équations 3.1 et 3.2 n'est pas restreinte à notre nouveau modèle de BRDFs mais pourrait être utilisée pour les modèles physiquement exacts introduits précédemment.

3.6 Approximation des matériaux mesurés avec l'ISF

Pré-traitement des données

Il a été noté par plusieurs chercheurs, dont Ngan *et al.* [NDM05], que les données de la base de BRDFs du MERL-MIT [MPBM03b] ne sont pas forcément exploitables directement (bien que déjà filtrées par Matusik *et al.* pour réduire le bruit dû aux appareils de mesure). Par exemple, le bruit de mesure est de plus en plus intense lorsque que l'on s'approche des configurations rasantes et par conséquent Ngan *et al.* ont choisi de simplement ignorer toutes les mesures au-delà de 80 degrés. Nous préférons faire un choix moins drastique en projetant toutes les mesures dans l'espace ISF où les configurations rasantes génèrent des valeurs faibles, comme expliqué précédemment. L'espace ISF possède l'avantage de réduire les valeurs des mesures aux endroits où elles sont les moins fiables. Ceci rend aussi le processus d'approximation plus robuste. A la différence de Ngan *et al.*, qui ont travaillé sur les données initiales non filtrées, nous n'enlevons aucun *outlier*² (puisque les données fournies par MERL-MIT sont déjà filtrées).

Les mesures de réflectance fournies par la base de données MERL-MIT sont exprimées dans la paramétrisation de Rusinkiewicz [Rus98] avec un échantillonnage an-

²Terme anglais employé en statistique. Il représente une donnée qui diffère significativement des autres données alors que toutes proviennent du même phénomène mesuré.

gulaire de 1 degré pour les angles différentiels (θ_d, ϕ_d) et θ_h . Lorsque l'on projette les mesures dans l'espace ISF, cet échantillonnage angulaire génère un regroupement (*clusterisation*) naturel des paramètres (r, s) qui ne prennent qu'un nombre fini de valeurs. Ces groupes (*clusters*) peuvent être vus comme les classes d'équivalence de la projection 3D vers 2D générées par la paramétrisation de Stark *et al.*

L'analyse de ce processus de *clusterisation* montre que la variance de chaque *cluster* diffère selon le type de matériau. Dans le cas de matériaux diffus ou légèrement glossy, la variance dans chaque *cluster* est toujours petite comparativement à la valeur absolue de la magnitude de la réflectance correspondante, tandis que, pour les matériaux plus spéculaires, cette variance augmente. Ceci peut s'expliquer par deux raisons. Premièrement, la résolution angulaire peut devenir insuffisante pour correctement échantillonner les matériaux les plus spéculaires. Deuxièmement, ce type de matériaux est nettement plus sensible aux bruits de mesures. Dans ce dernier cas, filtrer les mesures permet de réduire la variance avant de lancer le processus d'approximation. Cependant, pour éviter d'introduire un biais statistique, nous effectuons un filtrage sur les données uniquement dans chacun des *clusters* et non sur un voisinage de *clusters*. Après avoir testé plusieurs procédures (minimum, maximum, moyenne, etc.) pour trouver un échantillon représentatif au sein d'un *cluster*, nous avons opté pour la réflectance médiane qui apparaît comme le choix le plus robuste puisqu'il permet d'enlever naturellement les *outliers* contenus dans un *cluster*.

Approximation globale vs approximation aux limites

Une fois les données projetées, regroupées et filtrées dans l'espace ISF, nous pouvons lancer le processus d'approximation. Nous avons choisi d'utiliser le DCA, pour *Differential Correction Algorithm* [PBR71], qui convertit le processus d'approximation par un polynôme rationnel en un simple problème linéaire de type simplex. Cet algorithme assure une convergence au sens de la norme L^∞ et assure également que le dénominateur n'a pas de pôles sur le domaine d'approximation considéré. Un pôle au dénominateur générerait des instabilités numériques, voire des résultats faux. De plus, comme nous cherchons à approximer une BRDF qui est une fonction positive ou nulle, nous ajoutons

au problème linéaire original des contraintes supplémentaires pour forcer le numérateur à être toujours non négatif. Enfin, d'autres contraintes linéaires peuvent être ajoutées au besoin si le processus d'approximation doit être affiné.

Afin d'accélérer la vitesse de convergence du DCA, nous utilisons l'algorithme de Papamarkos [Pap88], qui permet d'obtenir une solution proche de la solution optimale. Nous utilisons alors cette première approximation comme initialisation pour le DCA, ce qui permet d'en réduire le nombre d'itérations. L'annexe II présente en détail l'algorithme du DCA.

Rappelons que nous avons proposé deux représentations analytiques pour l'ISF, l'une étant bi-variée $\sigma(r, s)$ et l'autre étant une interpolation $\bar{\sigma}(r, s)$ obtenue à partir de courbes 1D limites σ_r et σ_s . Nous utilisons le DCA pour les deux formulations. Dans le cas de l'expression bi-variée, nous qualifions de *global* le processus d'approximation tandis que dans le cas de la paire de courbes 1D, nous le qualifions de *limite*. Dans le premier cas, le DCA assure une convergence globale pour la norme L^∞ alors que, dans le second cas, la minimisation de l'erreur est uniquement garantie sur les courbes limites. Cependant, pour un nombre donné de coefficients $M + N$, les degrés du polynôme rationnel bi-varié sont en général beaucoup plus faibles que ceux des courbes. C'est pour cette raison que nous avons observé, que pour les matériaux glossy ou spéculaires, l'erreur finale peut être plus faible avec la version interpolée.

Reparamétrisation homographique

En observant attentivement les histogrammes présentés en figure 3.5, on remarque que, pour les matériaux spéculaires, l'ISF présente des zones avec des transitions très abruptes où la réflectance varie très rapidement de valeurs très élevées à des valeurs proches de zéro. Ce n'est pas une surprise puisque les matériaux spéculaires sont connus pour avoir des hautes fréquences directionnelles.

Même si les polynômes rationnels sont capables de représenter des courbes ou des surfaces avec de telles variations abruptes, la technique du DCA n'est pas assez efficace pour une telle configuration des données. Récemment, les travaux en approximation se sont concentrés pour traiter ce type de cas (e.g. [AG00]). Dans notre implémentation,

nous n'avons pas utilisé de telles techniques avancées ; nous utilisons une solution nettement plus simple qui repose sur la reparamétrisation homographique (RH).

Une reparamétrisation homographique consiste à appliquer un polynôme rationnel linéaire sur les paramètres d'une fonction avant de l'évaluer. Dans notre cas, une propriété intéressante est qu'elle ne modifie pas le degré du polynôme rationnel résultant. Ainsi, on peut voir cette étape de reparamétrisation comme une étape de pré-conditionnement des données en vue d'améliorer la qualité du processus d'approximation. Nous avons uniquement appliqué la reparamétrisation homographique sur le paramètre s puisque c'est sur ce paramètre que les variations sont les plus fortes. Par exemple à la figure 3.7(a), la courbe présentée est celle de l'isoline $r = 0$ du matériau *chrome-steel*³. On peut voir sur cette même figure que la variation de la réflectance se concentre sur un tout petit intervalle du paramètre et crée donc une configuration très complexe pour le processus d'approximation. Si l'on applique une RH (cf. figure 3.7(b)), les variations de la fonction s'étendent sur un intervalle beaucoup plus large et rendent le processus d'approximation plus aisé et plus robuste. Une fois le processus d'approximation effectué, le polynôme rationnel obtenu est pré-multiplié par le polynôme homographique pour obtenir la fonction approximante finale.

3.7 Résultats

Nous avons testé notre nouveau modèle sur les neuf matériaux présentés à la figure 3.5. Ces matériaux exhibent des comportements variés allant du lambertien (e.g. *fabric-beige*) au miroir parfait (e.g. *chrome-steel*). De plus, certains matériaux exhibent des comportements colorimétriques spécifiques (e.g. *metallic-blue*) alors que d'autres présentent des effets de Fresnel (e.g. le nylon).

Résultats quantitatifs

Le tableau 3.I présente les erreurs obtenues en approximant les mesures à l'aide de l'ISF pour plusieurs configurations. Nous présentons nos résultats avec deux types d'er-

³Il s'agit d'acier chromé.

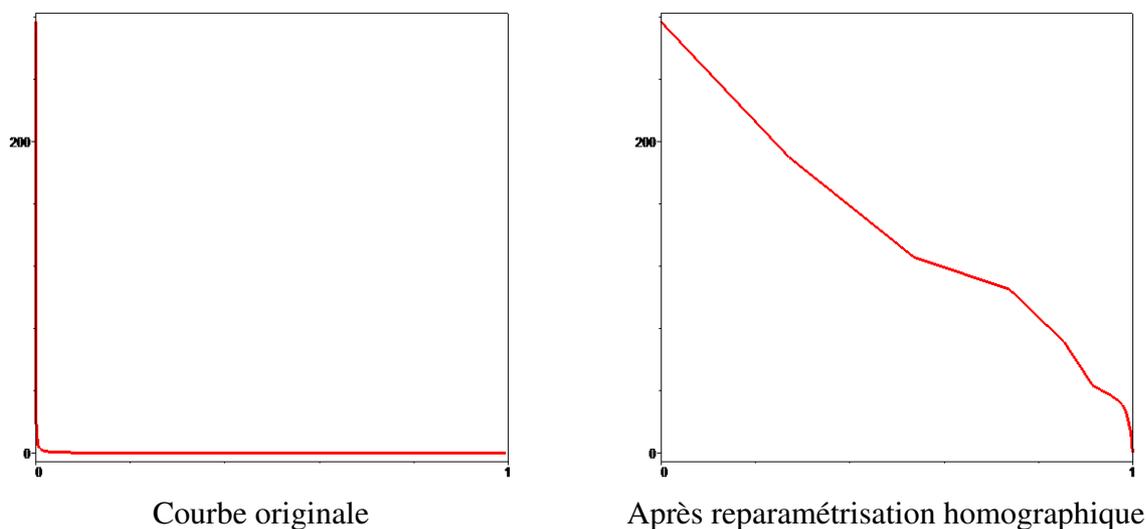


Figure 3.7 – Reparamétrisation homographique sur la courbe limite σ_s pour le matériau *chrome-steel* de la base de données MERL-MIT.

reurs : une première de type L^∞ qui est calculée comme le maximum des erreurs élevées au carré ; une seconde de type L^2 qui est calculée comme la moyenne des erreurs élevées au carré. Afin de comparer la qualité de l'approximation entre les différents matériaux, chacune des métriques est divisée par la moyenne des intensités prises au carré de la BRDF. Nous avons aussi inclus, dans la première colonne, l'erreur générée dans l'espace ISF par la meilleure approximation de Ngan *et al.* [NDM05]. Cependant, nous ne considérons pas que cette colonne contient l'erreur de référence puisqu'ils ont eu accès aux données originales alors que nous n'avons accès qu'aux données filtrées (fournies dans la paramétrisation de Rusinkiewicz). Cela signifie que les approximations données dans l'annexe de leur article peuvent ne pas être optimales au regard des données filtrées contenues dans la base MERL-MIT. Néanmoins, nous pensons que ces résultats donnent tout de même une indication importante de la qualité de notre approximation à l'aide de notre nouveau modèle.

Le tableau 3.I montre la différence entre utiliser l'ISF directement en formulation surfacique ou à l'aide des deux courbes limites. Nous avons testé une surface rationnelle de type quadratique par linéaire (9 coefficients) et une autre de type cubique par quadratique (15 coefficients). Nous montrons aussi l'erreur obtenue en utilisant la re-

présentation à l'aide de courbes avec 5 (colonne «Courbe 1») ou maximum 10 (colonne «Courbe 2») coefficients par courbe. Contrairement aux approches de minimisation non-linéaire, une fois que le nombre de coefficients est fixé, le processus d'approximation est entièrement automatique. Ainsi, pour un nombre fixé de coefficients, le système teste toutes les combinaisons de degrés possibles pour le numérateur et le dénominateur et retourne celle avec l'erreur L^∞ la plus petite. Comme mentionné précédemment, la plupart des matériaux exhibent un comportement plus complexe sur la courbe σ_s que sur la courbe σ_r . Aussi, nous ajoutons principalement des coefficients sur la courbe σ_s à l'exception des matériaux spéculaires qui ont de grandes valeurs pour σ_r . Pour ce type de matériaux, il est crucial d'approximer au mieux la vitesse de décroissance de σ_r et la reparamétrisation homographique revêt donc toute son importance.

Augmenter le nombre de coefficients provoque automatiquement une amélioration de l'approximation quand on utilise la représentation 2D surfacique de l'ISF. En effet, nous résolvons un problème de programmation linéaire et ceci garantit de converger vers une erreur globale minimale. En revanche, ce n'est pas forcément le cas avec la représentation à l'aide de paires de courbes limites. Bien qu'augmenter le nombre de coefficients sur chaque courbe améliore l'approximation au regard de la norme L^∞ , cela n'améliore pas forcément l'erreur globale, comme on peut le voir pour le matériau *gold-paint* au tableau 3.I. Néanmoins, ce même tableau montre que nos approximations à l'aide des courbes limites sont très bonnes comparées à celles obtenues avec l'approximation surfacique.

Résultats qualitatifs

En tant qu'erreur qualitative, nous comparons les rendus (cf. figure 3.8) d'une sphère éclairée avec un environnement lumineux complexe avec les données originales, notre approximation et la meilleure approximation de Ngan *et al.* A l'exception du *chrome-steel*, nous échantillons l'environnement lumineux de manière uniforme sous la forme de lumières directionnelles. Pour les matériaux les plus spéculaires, nous utilisons un schéma d'importance (*importance sampling*) piloté par la meilleure approximation de Ngan *et al.* pour sélectionner les directions de lumière. La couleur orangée présente à

	Ngan <i>et al.</i>	Surfacique 1	Surfacique 2	Courbe 1	Courbe 2
<i>fabric-beige</i>	0.047 - 0.165	0.006308 - 0.096670	0.005149 - 0.082060	0.002 - 0.081	0.00224 - 0.077956
<i>fabric-pink</i>	0.029 - 0.284	0.035368 - 0.166898	0.024684 - 0.143480	0.012186 - 0.201150	0.0128 - 0.192
<i>red-plastic</i>	0.176 - 5.072	0.212565 - 3.555914	0.265402 - 3.292972	0.03567 - 4.564350	0.092744- 3.358226
<i>nylon</i>	0.603 - 66.501	0.975136 - 52.844093	0.740938 - 42.556693	0.493416 - 50.073309	0.459 - 47.82
<i>fruitwood-241</i>	0.938 - 70.974	2.159609 - 16.492461	0.161203 - 15.25347	0.153417 - 19.510875	0.157 - 19.510332
<i>metallic-blue</i>	0.329 - 3.179	0.016832 - 0.529237	0.019530 - 0.272510	0.011319 - 0.240038	0.005 - 0.164
<i>gold-paint</i>	0.353 - 137.575	0.784637 - 92.681384	0.473731 - 106.506454	0.083155 - 73.999200	0.087605 - 78.980105
<i>nickel</i>	0.356 - 20.581	0.111108 - 12.796227	0.6295 - 19.796	0.110608 - 3.530724	0.106637 - 3.500720
<i>chrome-steel</i>	0.677 - 57.739	0.502911 - 27.546729	0.423603 - 28.041568	0.240442 - 28.861142	0.2170 - 26.147666

Tableau 3.I – Comparaisons des erreurs d’approximation (dans l’espace ISF) pour différents matériaux en utilisant la représentation 2D ou 3D de l’ISF.

la figure 3.8 pour les résultats de Ngan *et al.* peut sembler surprenante mais elle provient directement des résultats, pour les albédos diffus et spéculaires reportés par ces derniers dans l’annexe de leur article [NDM05]. Enfin, la figure 3.9 met en évidence le fait que certains matériaux tels que *metallic-blue* ne peuvent être représentés avec une simple approximation de la luminance et d’un triplet RGB. D’autres matériaux, comme le *nickel*, exhibent aussi des comportements chromatiques spécifiques pour le canal bleu.

Edition à l’aide des courbes limites

Comparée à la formulation bi-variée, la formulation de l’ISF avec les courbes limites offre une représentation bien plus intuitive puisque l’aspect global de la surface interpolée peut facilement être déduite à partir des courbes σ_r et σ_s . Sans avoir développé une solution interactive complète, nous avons testé les résultats obtenus lorsqu’on laisse l’utilisateur manipuler les courbes limites afin de créer des ISFs manuellement.

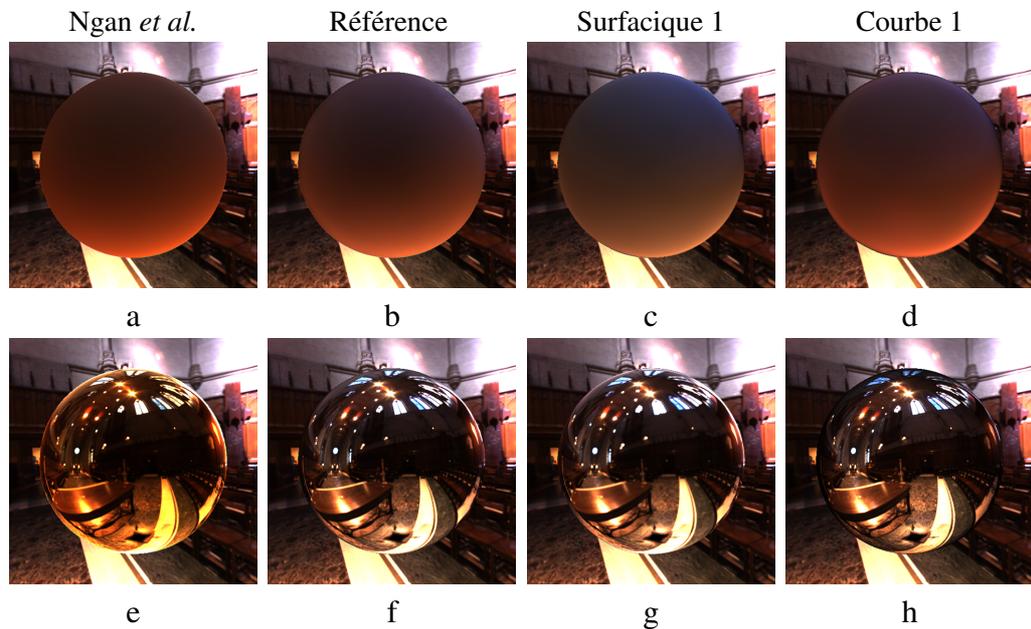


Figure 3.8 – Rendus obtenus avec le matériau *fabric-beige* (a-d) et le *chrome-steel* (e-h).

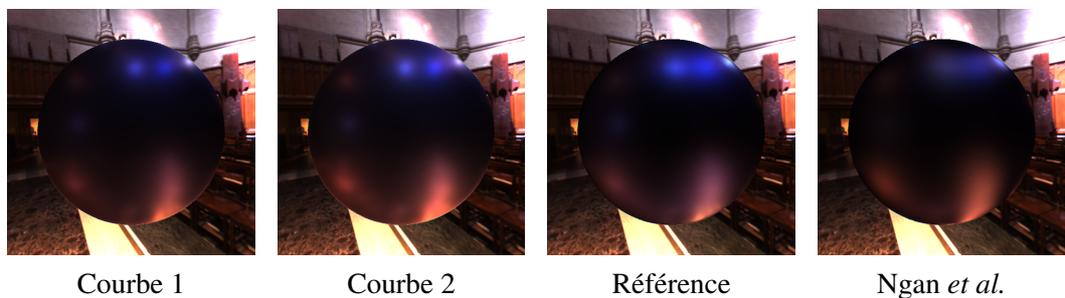


Figure 3.9 – Approximation par composante chromatique du matériau *metallic-blue*. Les courbes ISFs approximées sur les images de gauche ont 5 (resp. 10) coefficients et l'erreur RMS sur le canal bleu est cinq fois meilleure que celle de la meilleure approximation de Ngan *et al.*

Afin d'avoir un processus d'édition assez flexible, nous avons développé une solution pour laquelle chaque courbe limite est représentée par une Bézier rationnelle où la position des points de contrôle et les poids peuvent être modifiés par l'utilisateur. Le système force la continuité des courbes au point $(0,0)$. Nous avons été surpris de constater que cette solution naïve offre déjà une grande flexibilité et permet de créer une grande variété

de formes même lorsque l'on se limite à des courbes de Bézier rationnelles quadratiques.

La figure 3.10 présente de nombreux matériaux créés en éditant deux courbes rationnelles quadratiques. Tous les matériaux sont rendus dans les mêmes conditions d'éclairage. Le type de matériau obtenu varie du lambertien au spéculaire pur, avec ou sans effet de Fresnel. Ceci démontre la richesse, du point de vue de l'édition, de l'ISF avec une seule couleur et deux courbes limites. Naturellement, on pourrait, comme cela a été fait ailleurs [BAOR06, LBAD⁺06], laisser l'utilisateur éditer une courbe par composante chromatique pour lui laisser encore plus de flexibilité.

Discussion

Nous avons montré que la représentation ISF introduite, avec deux formulations, présente de nombreux avantages dont l'un est la garantie de convergence globale du processus d'approximation lorsqu'on utilise la formulation bi-variée. Nous avons aussi montré que la formulation à base de courbes limites présente un intérêt du point de vue de l'édition. Une question que l'on est en droit de se poser est de savoir si la représentation à base de courbes permet de décrire tous les matériaux isotropes. Si l'on se place sur un plan purement mathématique, nous avons déjà constaté que certains matériaux synthétiques qui font intervenir plusieurs lobes spéculaires et rétro-réfléctifs seront difficilement représentables. Par exemple, un tel matériau peut être obtenu en combinant un lobe avant et un lobe arrière avec le modèle de Lafortune *et al.* [LFTG97]. Dans une telle configuration, la surface générée dans l'espace ISF présente des variations de formes complexes au centre du triangle barycentrique qui ne pourront être déduites de la forme des courbes limites. En revanche, il est clair que la formulation bi-variée de l'ISF permettra de représenter de tels matériaux. Enfin, la réponse sur le plan mathématique a certainement un intérêt mais elle ne saurait être complète sans une confirmation expérimentale de mesures de matériaux rétro-réfléctifs. En effet, à l'heure actuelle, la communauté graphique ne dispose pas de mesures de tels matériaux et il est donc impossible de donner une réponse définitive à la question posée. Une extension possible pour la formulation à base de courbes 1D serait d'ajouter une troisième courbe $r = s$ pour mieux contrôler l'interpolation au milieu du triangle barycentrique.

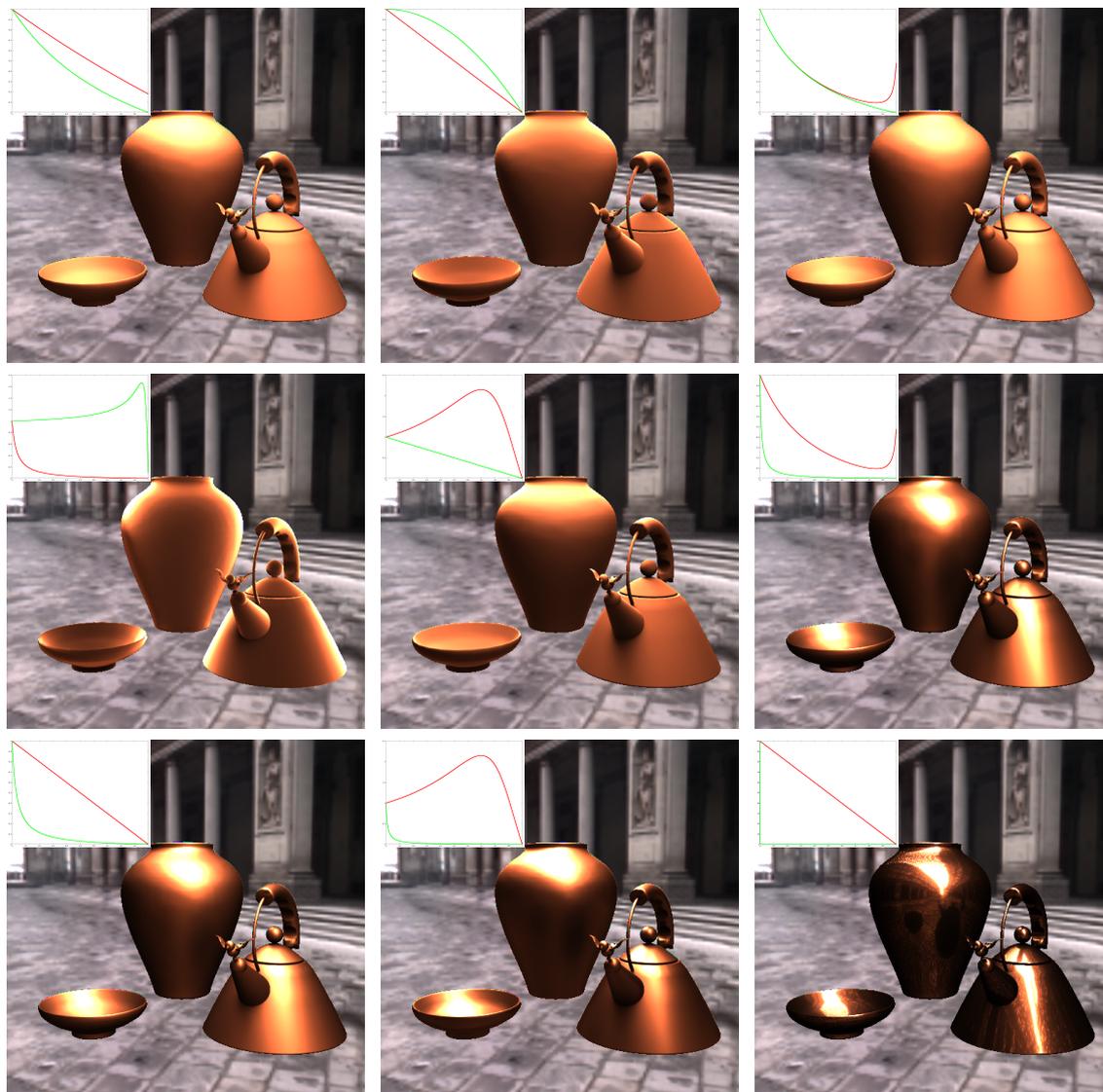


Figure 3.10 – Différents types de matériaux créés avec l’ISF représentée par des courbes rationnelles quadratiques. La courbe σ_s , en vert contrôle la spécularité du matériau qui peut varier du diffus au spéculaire parfait. La courbe σ_f , en rouge contrôle les effets de Fresnel à angle rasant ainsi que le type de réflexion spéculaire.

3.8 Perspectives futures

Comme son nom l’indique, l’ISF introduite peut uniquement représenter des matériaux isotropes. Une première extension serait d’étendre cette représentation à un modèle

complètement anisotrope. Une méthode est d'étendre le polynôme rationnel bi-varié à un polynôme rationnel tri-varié dont le troisième paramètre contrôlerait l'anisotropie. Le triangle barycentrique utilisé deviendrait alors un tétraèdre avec certaines redondances dues à la périodicité de l'anisotropie.

Une fois l'anisotropie proprement modélisée, il serait intéressant d'étendre le modèle pour représenter une SVBRDF. On pourrait coupler une analyse en ondelettes de la variation spatiale des BRDFs afin de compresser au maximum les données. Le même type d'extension pourrait être envisagé avec les BTFs mais avec la réserve que les ombres portées introduites par la méso-structure nécessiteront d'être représentées avec plus de coefficients qu'une simple BRDF. Afin de mener à bien ces extensions du modèle, il nous paraît important d'améliorer la paramétrisation de Stark *et al.* qui n'est pas totalement uniforme.

Enfin, il serait intéressant de travailler sur un schéma d'importance pour échantillonner le modèle dans le cadre du rendu non temps réel. Ce schéma pourrait être numérique (e.g. [LRR04]), bien qu'une dérivation analytique serait plus intéressante en pratique.

CONCLUSION DE LA PREMIÈRE PARTIE

Après avoir fait un état des lieux des différents aspects de la recherche sur les BRDFs nous avons, dans cette première partie, proposé deux représentations. Une première, résolument destinée aux artistes, dont l'édition est facile et intuitive. La représentation du lobe spéculaire à base de courbes et de textures permet à un utilisateur de facilement spécifier, et ce en temps réel, l'apparence du reflet. La seconde représentation est plus destinée aux mesures de BRDFs et autorise une forme simple d'édition. Elle utilise la puissance d'expressivité des polynômes rationnels définis dans l'espace paramétrique de Stark *et al.* L'utilisation des polynômes rationnels permet alors d'utiliser des techniques d'approximation linéaire à convergence globale garantie.

Ainsi, la question qui se pose de manière naturelle est de savoir s'il serait possible d'obtenir une représentation unifiée qui servirait à la fois pour le travail d'un artiste et celui d'un architecte. Dans le cas de l'artiste, les contraintes physiques sont souvent un frein à l'expressivité alors qu'elles sont nécessaires pour les applications de type architecture, industrie automobile, etc. où la prédiction du rendu est importante. Garantir la réciprocité d'Helmholtz et la conservation de l'énergie d'un modèle de BRDFs ne saurait être l'unique étape pour l'utilisation d'un modèle dans un contexte industriel. En effet, il s'agit de conditions nécessaires mais non suffisantes à la réalisation physique du modèle. Il paraît donc important, dans un contexte applicatif, de développer une méthodologie qui permettrait de s'assurer que le modèle soit réalisable.

Enfin, la BRDF joue un rôle essentiel dans l'apparence d'une scène mais elle n'en est pas le seul élément. L'environnement lumineux, la géométrie des objets et l'interaction de ceux-ci avec la lumière sont aussi importants. Dans un contexte d'édition, une prévisualisation rapide du rendu de la scène avec le modèle édité paraît donc une voie de recherche à approfondir.

Pour ce faire, dans la partie suivante, nous nous intéressons à l'éclairage incident. Après avoir passé en revue les différentes techniques de rendu ainsi que les représentations de l'éclairage incident utilisées, nous en proposons une nouvelle représentation.

Deuxième partie

Eclairage incident

CHAPITRE 4

ETAT DE L'ART

Rappelons qu'en infographie, une scène est constituée de matériaux associés à des objets et de sources de lumière qui génèrent un éclairage. Dans cette seconde partie, nous nous intéressons à l'éclairage incident d'une scène. L'éclairage incident, en un point de la scène, est le résultat de la propagation directe et/ou indirecte de la lumière réfléchiée par les différents matériaux. En un point donné, il est nécessaire d'évaluer l'éclairage incident pour le multiplier à la BRDF et calculer ainsi sa contribution finale dans l'image. Dû aux multiples indirections que la lumière peut prendre, l'éclairage incident reste un terme difficile à évaluer. Le représenter ou le traiter efficacement constitue donc un axe de recherche aussi important que l'axe de recherche sur les BRDFs abordé en première partie. Dans ce chapitre, nous présentons le cadre mathématique associé à l'éclairage et l'éclairage, les différentes représentations utilisées pour l'éclairage ainsi que les différents algorithmes utilisés pour le calculer. Nous donnons également les tendances et les problématiques actuelles associées à l'éclairage et abordons en détail deux d'entre elles avec les systèmes de cache (cf. section 4.5) et d'édition (cf. section 4.4). Au chapitre 5, nous présentons une nouvelle structure de données avec ses applications en tant que système de cache. Au chapitre 6, nous présentons une technique qui modifie l'éclairage incident afin de mettre en évidence les détails de la surface. Cette technique peut être considérée comme une forme d'édition de l'éclairage.

4.1 Éclairage global en synthèse d'images

En infographie, la notion d'éclairage incident fait partie intégrante d'un domaine appelé éclairage global. Les algorithmes de rendu par éclairage global prennent en compte l'ensemble des interactions possibles entre la lumière et la géométrie de la scène. Ils tiennent compte non seulement de l'éclairage incident *direct* (éclairage dit *local*) d'un objet par les sources lumineuses, mais aussi de l'éclairage dû aux autres objets

(éclairage incident *indirect*) réfléchissant de la lumière. La figure 4.1 illustre, pour une même scène, la différence entre un rendu obtenu en tenant compte de l'éclairage local par rapport à l'éclairage global.

Formulation mathématique

En première approximation, un algorithme de rendu doit calculer pour chaque pixel d'une image une valeur de radiance (cf. annexe I). Kajiya [Kaj86] a montré que tout algorithme de rendu tente en fait de résoudre une même équation appelée «équation du rendu», qui découle de la loi de la conservation de l'énergie en un point. Intuitivement, cette équation formalise le fait que l'énergie réfléchiée par un point est au maximum égale à la somme de l'énergie émise et réfléchiée en ce point :

$$L(x \rightarrow \omega_o) = L_e(x \rightarrow \omega_o) + L_r(x \rightarrow \omega_o) \quad (4.1)$$

où :

- $L_e(x \rightarrow \omega_o)$ la radiance émise par la surface au point x dans la direction ω_o
- $L_r(x \rightarrow \omega_o)$ la radiance réfléchiée, par la surface au point x dans la direction ω_o .

En réexprimant la radiance réfléchiée, on obtient l'équation du rendu [Kaj86] :

$$L(x \rightarrow \omega_o) = L_e(x \rightarrow \omega_o) + \int_{\Omega_n} L(x \leftarrow \omega_i) f_r(x, \omega_i, \omega_o) (n \cdot \omega_i) d\omega_i \quad (4.2)$$

où $f_r(x, \omega_i, \omega_o)$ représente la BRDF au point x , Ω_n l'hémisphère centré en x et orienté selon n et $L(x \leftarrow \omega_i)$ la radiance incidente en x . Le terme $L_r(x \rightarrow \omega_o)$ est donc le résultat de la convolution de l'éclairage incident par la BRDF qui joue le rôle d'un filtre.

La simplicité apparente de cette équation masque la complexité de sa résolution puisque l'inconnue (la radiance) apparaît dans les deux membres de l'équation ; ceci met en évidence son caractère hautement récursif. Tout algorithme de rendu résout de manière plus ou moins partielle et plus ou moins efficace l'équation du rendu.

Éclairage indirect

Pour mettre en évidence les notions d'éclairage indirect et d'éclairage local, on

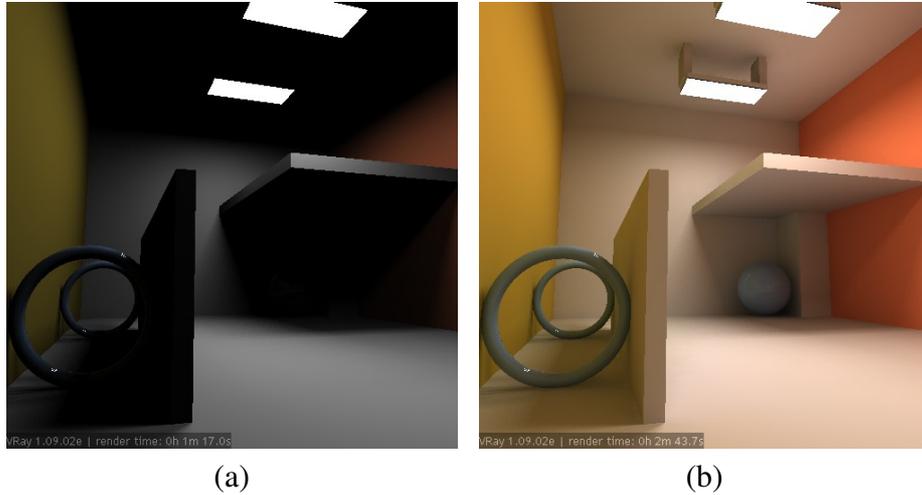


Figure 4.1 – Différence de résultats entre (a) une méthode utilisant uniquement l'éclairage direct et (b) l'autre utilisant l'éclairage global. Images tirées de [Cha05].

décompose la radiance réfléchie $L_r(x \rightarrow \omega_o)$ deux termes :

$$\begin{aligned}
 L(x \rightarrow \omega_o) = & L_e(x \rightarrow \omega_o) + \underbrace{\int_A L_e(y \rightarrow -\omega_i) f_r(x, \omega_o, \omega_i) V(x, y) G(x, y) dA_y}_{\text{Eclairage direct}} \\
 & + \underbrace{\int_{\Omega_n} L(x \leftarrow \omega_i) f_r(x, \omega_i, \omega_o) (\mathbf{n} \cdot \omega_i) d\omega_i}_{\text{Eclairage indirect}}
 \end{aligned} \tag{4.3}$$

où :

- A représente l'ensemble des surfaces appartenant aux sources de lumière.
- $V(x, y)$ représente le terme de visibilité entre x et y et qui est défini par $V(x, y) = 1$ si x et y sont mutuellement visibles, 0 dans le cas contraire.
- $G(x, y)$ est le terme géométrique défini par :

$$G(x, y) = \frac{\max(\mathbf{n} \cdot \omega_i, 0) \max(\mathbf{n}_y \cdot \omega_i, 0)}{\|x - y\|^2}$$

où \mathbf{n}_y représente la normale au point y .

Nous avons donc décomposé la radiance réfléchie en deux intégrales. L'une surfacique représentant l'éclairage direct et l'autre hémisphérique représentant l'éclairage

indirect. Contrairement à l'intégrale de l'éclairage indirect, l'intégrale de l'éclairage direct utilise uniquement la radiance émise L_e et n'est donc pas récursive ; elle est donc des deux intégrales, la plus simple à évaluer. Si l'on reformule la radiance incidente $L(x \leftarrow \omega_i)$ on peut faire apparaître à nouveau la radiance réfléchie montrant ainsi le caractère récursif de l'intégrale :

$$L(x \leftarrow \omega_i) = L_r(r(x, \omega_i) \rightarrow -\omega_i)$$

où $r(x, \omega_i)$ est l'opération de lancer de rayon, telle qu'introduite par Dutré *et al.* [DBB06], qui renvoie le point de la scène le plus proche de x dans la direction ω_i :

$$r(x, \omega_i) = \{y \mid y = x + t_{min} \omega_i\} \quad \text{avec} \quad t_{min} = \min\{t \mid t > 0, x + t \omega_i \in A\}.$$

Comme nous allons le voir dans la prochaine section, c'est principalement l'intégrale de l'éclairage indirect à laquelle s'attaquent les algorithmes d'éclairage global.

Avant d'aborder les systèmes existants qui permettent d'éditer l'éclairage (cf. section 4.4), nous passons brièvement en revue les techniques classiques qui permettent de calculer l'éclairage global (cf. section 4.2) et leurs évolutions récentes (cf. section 4.3). A la section 4.5, nous présentons également les techniques de *caching*¹. Nous verrons que ces techniques permettent d'accélérer les temps de rendu au prix de l'introduction d'un biais dans la solution calculée.

¹Le terme «*caching*» désigne la technique tandis que le terme «*cache*» désigne la structure qui permet d'implémenter la technique.

4.2 Aperçu des algorithmes d'éclairage global

Algorithmes déterministes

Historiquement, les premiers algorithmes font l'hypothèse que la scène est diffuse (cf. section 1.4) et résolvent de manière déterministe l'ensemble des interactions possibles entre les divers éléments de la scène. Ces algorithmes, basés sur les méthodes d'éléments finis, se regroupent sous le terme de méthodes de radiosité. Un aperçu de ces méthodes se trouve dans les livres de Sillion et Puech [SP94] et de Cohen *et al.* [CWH93]. Ils ont l'avantage d'être déterministes, d'être indépendants du point de vue, et leurs résultats peuvent donc être réutilisés pour la visualisation de la scène 3D lorsque le point de vue change. L'intérêt des algorithmes déterministes est l'absence de bruit (au profit d'un biais) dans la solution obtenue. Malheureusement, ils nécessitent une géométrie subdivisible (*tessellable*) et ne prennent en compte des scènes avec des BRDFs *glossy*, voire miroir, qu'au prix d'un grand surcoût mémoire, tel que celui de Granier *et al.* [GDW00]. Les techniques de radiosité ont connu leur essor au début des années 80. Depuis les travaux pionniers de Goral *et al.* [GTGB84], souvent regroupés sous le terme de *radiosité classique*, les améliorations qui se sont succédé visent, pour la plupart, à améliorer les problèmes de stockage, sous forme matricielle, des facteurs de forme et d'adaptations automatiques de la *tessellation* de la scène. Le facteur de forme entre deux éléments de la scène représente la proportion d'énergie échangée en tenant compte de la visibilité, de l'orientation et de la proximité des deux éléments. Puisque la visibilité intervient dans leur calcul, ces derniers restent coûteux et numériquement complexes à évaluer. Les techniques de radiosité progressive [CCWG88, HSA91, Sch93b] apportent un début de solution au problème du stockage en ne recalculant à chaque fois qu'une ligne/colonne de la matrice d'interactions alors que les techniques de *clustering* (regroupement) [SAG94, SDS95, Sil95] rassemblent volumétriquement les éléments de la scène afin de réduire le nombre d'interactions et donc le temps de calcul. Grâce à l'utilisation de chemins lumineux aléatoires, les méthodes de radiosité stochastique [NFKP94, NNB⁺96, BNN⁺98] permettent d'éviter complètement l'évaluation et le stockage de ces facteurs de forme. La technique d'estimation de densité de particules

sur un maillage, proposée par Walter [Wal98], peut aussi être considérée comme une forme de radiosit  stochastique puisque l'irradiance et la radiosit  sont  valu es sans calcul et stockage de facteurs de forme. En somme, on peut voir la radiosit  stochastique comme une technique essayant de tirer le meilleur parti des techniques d terministes et des techniques stochastiques pr sent es ci-dessous.

Algorithmes stochastiques

L'id e principale des algorithmes stochastiques est de g n rer des chemins lumineux,  valuer la radiance transmise le long de ceux-ci, et accumuler leur contribution dans l'image. On utilise les m thodes d'int gration de Monte-Carlo pour g n rer les chemins lumineux et  valuer de mani re stochastique l' quation du rendu. Le principal avantage de ces m thodes est leur simplicit  d'impl mentation et leur g n ralit , car toutes les BRDFs sont utilisables et l'ensemble des interactions est pris en compte. Leur principal inconv nient est leur lenteur   converger vers une solution non bruit e. Le bruit provient de divers facteurs dont l'un des principaux est l' valuation de l'int grale de l' clairement indirect voire de l' clairement direct. En effet, dans le cas o  le nombre de sources lumineuses est tr s  lev ,  chantillonner toutes les sources devient trop co teux et on a souvent recours   une  valuation stochastique de l' clairement direct introduisant *de facto* du bruit.

De mani re g n rale, les techniques stochastiques essaient de choisir les chemins lumineux qui contribuent le plus   la g n ration de l'image. Un aper u des m thodes stochastiques est donn  dans le livre de Dutr  *et al.* [DBB06]. La m thode la plus simple et la plus connue, le trac  de rayons ou *Ray Tracing* [Whi80], consiste   lancer des rayons depuis une cam ra et   les propager dans la sc ne. Les rayons se propagent uniquement pour la r flexion sp culaire parfaite et la r fraction parfaite ; par cons quent seul un sous-ensemble des interactions lumineuses est pris en compte. Une g n ralisation du *Ray Tracing*, appel e *Path Tracing*, prend en compte l'ensemble des chemins lumineux ; elle a  t  introduite par Kajiya [Kaj86] comme solution   son  quation du rendu. Partant du constat que certains chemins lumineux sont plus faciles    chantillonner depuis les sources de lumi re, le trac  de rayons bi-directionnel propos  par Lafortune et

Willems [LW93] ainsi que Veach et Guibas [VG94] permet d'accélérer la convergence de la solution. En effet, la probabilité d'atteindre une source lumineuse depuis la caméra peut être très faible, notamment en présence d'objets fortement spéculaires ou réfractifs alors qu'il est simple de construire les chemins lumineux depuis les sources de lumière. La principale difficulté est de connecter les chemins construits depuis le point de vue et les sources de lumière sans introduire de biais. Veach et Guibas ont également introduit une extension, appelée Metropolis [VG97], au tracé de rayons bi-directionnel. L'idée est d'introduire des perturbations locales aux chemins lumineux afin de les faire converger plus rapidement vers les chemins plus importants.

Ces diverses méthodes réduisent le bruit de manière plus ou moins efficace. Une autre manière de réduire le bruit est d'utiliser des schémas d'importance (*importance sampling*). Il s'agit de concentrer les calculs pour les chemins lumineux les plus importants. Définir quels sont ces chemins peut se faire en utilisant la BRDF (e.g. [LRR04]), en connaissant l'importance de l'éclairage incident (e.g. [ARBJ03]), ou en combinant les deux (e.g. [BGH05, WA09]). Le but d'un échantillonnage reposant sur une BRDF est de concentrer les rayons proportionnellement à la spécularité de la BRDF. Échantillonner le signal lumineux dans les directions où la lumière est la plus importante peut se faire pour l'éclairage direct (e.g. [CAM08]) et aussi pour l'éclairage indirect, où l'on a souvent recours à une structure de cache (cf. section 4.5), préalablement calculée, pour obtenir cette information d'importance. Pour traiter un nombre élevé de sources de lumière, plusieurs techniques accélérant les temps de calcul ont été introduites (e.g. [PPD98, WFA⁺05, HPB07]). Enfin, les méthodes de Metropolis ont été améliorées afin d'augmenter la cohérence de l'algorithme et de pouvoir utiliser les instructions SIMD [SIP07].

Aussi bien pour les méthodes déterministes que stochastiques, le terme de visibilité de l'équation (4.3) constitue en général le principal goulot d'étranglement. Une bonne évaluation de l'intégrale de l'éclairage indirect sur l'hémisphère requiert un nombre élevé de requêtes de visibilité. On peut distinguer deux axes de recherche dont le but est d'accélérer le calcul de l'éclairage. Le premier, abordé dans la section suivante, est guidé par une utilisation massive du GPU. Le second, abordé à la section 4.5, et qui

peut servir au précédent, vise à développer des techniques de *caching*. Le principe de ces techniques est de pré-calculer l'éclairage indirect à certains endroits de la scène et de l'interpoler aux endroits manquants. Ceci accélère le temps de calcul et a pour effet de remplacer le bruit, visuellement déplaisant, par un biais (perceptible comme du flou) qui est plus acceptable pour notre système visuel.

4.3 Tendances récentes en éclairage global

Depuis la démocratisation de la programmation sur carte graphique, notamment grâce aux langages de type Cg, GLSL, HLSL, et même ceux de plus haut niveau tels que CUDA ou OpenCL, on assiste à un développement de nouveaux algorithmes d'éclairage global. Cet engouement pour un développement GPU s'explique par le fait que la puissance de calcul des GPUs augmente plus rapidement que celle des CPUs. En revanche le modèle SIMD des GPUs est moins flexible que les modèles des CPUs. De nouvelles implémentations d'algorithmes d'éclairage global apparaissent donc essayant de le rendre interactif au maximum. Le but ultime est de développer une technique où tous les paramètres de la scène (objets géométriques, lumières, matériaux, point de vue) seraient modifiables en temps réel tout en obtenant un rendu réaliste ou, du moins, aussi réaliste que possible. D'un point de vue applicatif, on peut distinguer les techniques orientées édition de l'éclairage, des techniques uniquement orientées rendu d'une ou plusieurs frames. Classifier les différentes techniques n'est pas aisé et certains travaux montrent d'ailleurs qu'elles résolvent toutes le même problème mais sous des angles différents (cf. [Leh07]).

Precomputed Radiance Transfer

Nous commençons par les Pré-calculs de Transfert de Radiance ou *Precomputed Radiance Transfer* (PRT) qui ont connu un énorme succès ces dernières années. Elles reposent toutes sur un pré-calcul effectué sur CPU et permettent une certaine forme d'interactivité. Les PRTs pré-calculent la réponse lumineuse d'un objet à un environnement lumineux distant en la stockant sur chaque sommet de la géométrie ou dans une image. L'intérêt des PRTs est de laisser l'utilisateur modifier l'éclairage distant et d'observer la réponse de l'objet en temps interactif ou réel.

Ramamoorthi et Hanrahan [RH01a] sont les premiers à avoir utilisé une telle approche pour une BRDF diffuse. Les premiers travaux de Sloan *et al.* [SKS02] autorisaient uniquement des environnements lumineux de basses fréquences et des BRDFs diffuses ou glossy. Cette limitation a été rapidement éliminée avec les travaux de Ng *et*

al. [NRH03] d’abord en fixant la vue dans le cas de BRDFs glossy, puis en enlevant cette contrainte [NRH04] au prix d’une interactivité largement sacrifiée. Les travaux de Wang *et al.* [WTL04, WTL06a, WNLH06] et ceux de Liu *et al.* [LSSS04] améliorent l’interactivité du système tandis que ceux de Sloan permettent d’intégrer du *normal mapping* [Slo06].

A l’exception de quelques travaux novateurs [PWL⁺07], la plupart des techniques présupposent une scène statique. La limitation d’un éclairage uniquement distant n’a pas vraiment été enlevée mis à part par Pan *et al.* [PWL⁺07]. Le principal problème des PRTs est le coût mémoire requis pour stocker toutes les informations. Comme les informations sont souvent stockées par sommet, les techniques deviennent rapidement inutilisables si la complexité géométrique de la scène est trop élevée. Ce problème est en partie résolu par la technique proposée par Sloan *et al.* [SGNS07] mais en contraignant l’environnement lumineux à être diffus.

Eclairage global interactif sans pré-calcul

A contrario des PRTs, de nouvelles techniques ont été développées et essaient de tout calculer à la volée sur le GPU sans passer par un pré-calcul, souvent très long. Malgré la puissance du GPU, les techniques dites d’éclairage global interactif effectuent des approximations en limitant les indirections de la lumière à un rebond [LSK⁺07] ou en ignorant la visibilité des sources secondaires [DS05, DS06], ou encore en simplifiant les requêtes de visibilité [GKD07, RGK⁺08] qui restent très coûteuses. Certaines d’entre elles et les plus récentes reposent sur l’algorithme de Keller [Kel97], *Instant Radiosity*, où des rayons sont tracés depuis les sources lumineuses pour créer sur les surfaces diffuses intersectées des sources de lumière secondaires. Une fois cette première passe effectuée, la contribution des sources de lumière secondaires est ajoutée à l’éclairage direct. En théorie, les objets et les lumières peuvent être dynamiques au prix d’un nombre élevé de passes sur GPU. Ceci a conduit à différentes évolutions de la technique, telle celle de Laine *et al.* [LSK⁺07] qui réutilise les différentes sources de lumière secondaires afin de ne pas avoir à recalculer la visibilité. En revanche, la technique est limitée à une seule indirection d’éclairement et bien que l’on puisse bouger les objets de la scène, ils

n'influenceront pas l'éclairage indirect. Ritschel *et al.* [RGK⁺08] enlèvent cette limitation en introduisant des *Imperfect Shadow Maps* qui permettent, au prix d'une grossière approximation de la géométrie de la scène, de calculer de manière très rapide les requêtes de visibilité. Bien que la technique supporte les indirections multiples les performances se dégradent très rapidement au-delà d'une indirection. Enfin, les techniques qui évaluent implicitement la visibilité (e.g. [DSDD07, DKTS07]) restent souvent limitées en termes d'adaptativité vis-à-vis de la complexité géométrique de la scène.

La gestion de la complexité géométrique de la scène est en partie résolue en utilisant des stratégies de rendu de type Rendu Différé ou *Deferred Shading* qui stockent, lors d'une première passe, dans des buffers (*G-Buffers* [ST90a]) les informations nécessaires au *shading* (normale, position, matériaux...) puis accumulent lors de passes supplémentaires les contributions des lumières dans un autre buffer. Le *Deferred Shading* [DWS⁺88] est, désormais, souvent couplé à de l'échantillonnage entrelacé ou *Interleaved Sampling* [SIMP06] afin d'augmenter les performances de rendu puisque pour un même nombre d'échantillons, il réduit la répétition des motifs d'échantillonnage visuellement déplaisants. Les *G-Buffers* sont aussi utilisés dans le cadre de moteurs de rendu destinés à l'industrie du film [PVL⁺05] où l'on utilise le GPU pour avoir une prévisualisation rapide de l'image finale.

Dans la même lignée, les travaux de Walter *et al.* [WFA⁺05, WABG06], qui utilisent surtout le CPU, reposent sur la construction d'un arbre hiérarchique de lumière qui est évalué efficacement en faisant une coupe sur la profondeur dans celui-ci. Enfin, terminons ce tour d'horizon par les systèmes entièrement CPU tels que ceux pour le *Ray Tracing* interactif (e.g. [WKB⁺02]) ou encore hybrides (e.g. [BBS⁺09]). Si les premières techniques telles que celles de Purcell *et al.* [PDC⁺05] ou Carr *et al.* [CHCH06] restaient moins performantes que les systèmes SIMD CPU, les dernières avancées, pour les systèmes d'éclairage global sur GPU les concurrencent, voire les surpassent désormais. Citons, par exemple, la technique de Popov *et al.* [PGSS07] qui implémente un algorithme de *Ray Tracing* sur GPU, avec géométrie statique, et celle de Zhou *et al.* [ZHWG08] qui implémente la technique du *Photon Mapping* (cf. section 4.5.2) pour obtenir des caustiques avec géométrie dynamique. Cette dernière vient d'être améliorée par Wang *et*

al. [WWZ⁺09] qui prend en compte tous les chemins lumineux. Celle de McGuire et Luebke [ML09], technique hybride CPU-GPU, implémente un algorithme de *Photon Mapping* basé image et obtient de très bonnes performances au prix d'une qualité dégradée par rapport à l'implémentation originale et entièrement CPU du *Photon Map*.

Malgré toutes ces avancées technologiques et algorithmiques, ces différentes méthodes peinent encore à gérer la complexité géométrique. Les scènes utilisées pour les tester restent simples comparativement à celles qui sont utilisées dans l'industrie des effets spéciaux ou du cinéma.

4.4 Travaux antérieurs en édition

Les systèmes d'édition d'éclairage peuvent aussi se décomposer en plusieurs catégories :

- les systèmes interactifs d'éclairage global présentés à la section précédente.
- les systèmes de ré-éclairage ou *relighting* qui peuvent être considérés comme des cas particuliers des précédents : leur interactivité permet de modifier les paramètres des sources de lumière en observant directement les effets mais ils sont souvent limités en termes d'interactivité du point de vue.
- les systèmes dits à «rendu inverse». Dans ce type de système, l'utilisateur spécifie le résultat qu'il souhaite obtenir à l'aide de certaines métaphores d'interaction comme celle de peinture (*painting*).

Historiquement, les systèmes de *relighting* fixent la direction de vue afin d'accélérer la mise-à-jour de la solution. L'édition possible est de déplacer les sources de lumière. Les premiers systèmes (e.g. [GH00, PVL⁺05]) ne prenaient pas du tout en compte l'éclairement indirect. Celui d'Hasan *et al.* [HPB06] remédie à ce problème en calculant une matrice de transfert de l'éclairement direct en éclairage indirect : avec un calcul interactif de l'éclairement direct, celui de l'éclairement indirect devient interactif. Un système très abouti qui permet de déplacer les sources de lumière, les objets et de modifier les BRDFs en tenant compte de l'éclairement indirect est celui introduit par Sun *et al.* [SZC⁺07]. Dans ce système, l'édition d'un modèle de BRDF analytique est

également possible.

Conscients de la limitation d'un système à point de vue fixe, les chercheurs ont repoussé cette limite. Le système proposé par Kristensen *et al.* [KAMJ05] nécessite un pré-calcul substantiel pour pouvoir déplacer ensuite en temps réel les sources de lumière dans des scènes quelconques mais où la caméra reste libre. Postava *et al.* [CPWAP08] proposent une extension aux *Lights Cuts* de Walter *et al.* [WFA⁺05, WABG06] afin de laisser l'utilisateur éditer les matériaux et les lumières pour n'importe quel angle de vue.

L'intérêt de ces systèmes réside dans une prévisualisation rapide du rendu d'une scène qui serait proche de ce qu'une simulation physique donnerait après de longues minutes ou même des heures de calcul, tout en ayant la possibilité d'ajuster des paramètres. Un exemple typique d'utilisation, dans le domaine de l'architecture, est le placement de sources de lumière dans une pièce qui n'a pas encore été construite. Malheureusement, étant donné que l'éclairage global (et donc indirect) est un phénomène global, il reste difficile de spécifier localement une forme d'éclairage en ne faisant qu'ajuster des paramètres. Le processus d'ajustement des paramètres est pénible pour l'utilisateur qui effectue une série d'essais jusqu'à être satisfait du résultat. L'intérêt des systèmes de rendu inverse est de faciliter la tâche de l'utilisateur en lui laissant spécifier directement son résultat.

Le plus souvent, dans un contexte de rendu inverse, l'utilisateur peint le résultat qu'il souhaite obtenir directement sur la surface d'un objet. Le système se charge alors d'effectuer une résolution inverse. Cette résolution inverse est en général trop complexe à résoudre (problème sur ou sous-dimensionné) pour porter sur l'ensemble des paramètres de la scène. Ainsi dans le système de Schoeneman *et al.* [SDS⁺93], les seuls paramètres que le système inverse retrouve (minimisation au sens des moindres carrés) sont : les intensités des sources de lumière et leurs couleurs dans une scène aux surfaces lambertiennes. De cette manière, le problème inverse à résoudre devient linéaire. Cependant, il n'y a pas toujours de solutions aux contraintes dessinées par l'utilisateur. De plus, lorsque l'utilisateur spécifie qu'il souhaite modifier une zone de la scène, d'autres zones vont également voir leur luminosité changer par effet de bord. Une fois de plus, ceci est dû au fait que l'éclairage est un phénomène global. Une solution serait alors d'ajouter

des contraintes supplémentaires. Malheureusement, plus on ajoute de contraintes, plus l'espace de solutions possibles se restreint.

Le système de Kawai *et al.* [KPC93] permet à un utilisateur de spécifier plus de contraintes de haut niveau, mais la métaphore de spécification est nettement moins précise. A savoir, l'utilisateur ne peint pas sur la surface mais définit des objectifs globaux de perception lumineuse tels que «je voudrais une ambiance plus feutrée». Une extension du travail de Kawai *et al.* [KPC93] a été proposée par Costa *et al.* [CSF99] où le système tente de retrouver aussi la géométrie des sources de lumière et autorise l'utilisateur à spécifier des contraintes complexes. Plus récemment, Pellacini *et al.* [PBMF07] ont proposé un système où la position des sources de lumière n'est pas fixée (processus inverse non-linéaire). En se restreignant à un modèle d'éclairage local, le système est alors capable de spécifier de manière interactive les différents paramètres, au sens cinématographique [Bar97], des sources de lumière. Comme l'optimisation linéaire est complexe à résoudre, Pellacini *et al.* [PBMF07] laissent l'utilisateur ajouter des lumières (une par une) au fur et à mesure plutôt que d'essayer de résoudre les contraintes sur un ensemble de sources de lumière prédéfinies. L'utilisateur a donc des outils d'édition directe (ajout d'une nouvelle lumière) et des outils d'édition inverse (le *painting* sur les surfaces).

En somme, il y a peu de travaux réalisés sur le rendu inverse comparé au rendu direct parce que le problème est difficile à résoudre. Dès lors, il est important de subdiviser les outils d'édition proposés à l'artiste pour mieux conditionner la résolution inverse. Une autre composante importante de l'édition inverse est de réussir à convaincre l'utilisateur que les «pincesaux» qui lui sont proposés correspondent à sa façon de spécifier son résultat. Enfin, mis à part le système proposé par Obert *et al.* [OKP⁺08], il semble qu'aucun système de rendu inverse n'ait attaqué de front l'édition de l'éclairage indirect en temps interactif. De manière générale, l'édition de l'éclairage indirect semble avant tout motivé pour compenser les effets de matériaux ou résoudre un problème de positionnement des lumières. Au prochain chapitre, nous proposons de modifier l'éclairage dans un but différent, à savoir mettre en avant les détails géométriques d'un objet.

4.5 Systèmes de cache

Historiquement, les systèmes de cache sont utilisés pour accélérer les calculs de l'éclairage indirect. Comme nous l'avons vu à la section 4.3, un certain nombre de techniques GPU telles que les PRTs ou les systèmes de *relighting* peuvent être vues comme des systèmes de cache de nouvelle génération. Dans cette section, nous présentons la problématique mathématique associée à la mise en cache ainsi que les différents systèmes de cache existants.

4.5.1 Formulation mathématique

La stratégie des techniques de *caching* est de précalculer l'éclairage incident à différents endroits et de l'interpoler aux endroits manquants.

Malheureusement, il est complexe de représenter l'éclairage incident de manière efficace, quelle que soit la propriété de réflectance des surfaces intervenant dans l'éclairage. Cette difficulté est liée au large éventail de BRDFs. Les besoins de représentations de l'éclairage incident ne sont pas les mêmes pour les BRDFs de basses fréquences (lambertiennes) indépendantes du point de vue que pour les BRDFs spéculaires (miroir) qui sont entièrement dépendantes du point de vue. Ainsi, les stratégies pour représenter l'éclairage incident sont souvent dépendantes de la palette de la BRDF sous-jacente du matériau. Aussi les premiers systèmes de cache étaient principalement utilisés pour les BRDFs diffuses. Nous en donnons ci-dessous la justification mathématique.

Si l'on considère qu'une BRDF peut être séparée en une composante diffuse et une composante spéculaire, nous pouvons réécrire l'équation de l'éclairage indirect par :

$$\begin{aligned}
& \int_{\Omega_n} L(x \leftarrow \boldsymbol{\omega}_i) f_r(x, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o) (\mathbf{n} \cdot \boldsymbol{\omega}_i) d\boldsymbol{\omega}_i \\
&= \int_{\Omega_n} L(x \leftarrow \boldsymbol{\omega}_i) f_{r,S}(x, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o) (\mathbf{n} \cdot \boldsymbol{\omega}_i) d\boldsymbol{\omega}_i + \int_{\Omega_n} L(x \leftarrow \boldsymbol{\omega}_i) f_{r,d}(x, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o) (\mathbf{n} \cdot \boldsymbol{\omega}_i) d\boldsymbol{\omega}_i \\
&= \int_{\Omega_n} L(x \leftarrow \boldsymbol{\omega}_i) f_{r,S}(x, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o) (\mathbf{n} \cdot \boldsymbol{\omega}_i) d\boldsymbol{\omega}_i + f_{r,d}(x) \int_{\Omega_n} L(x \leftarrow \boldsymbol{\omega}_i) (\mathbf{n} \cdot \boldsymbol{\omega}_i) d\boldsymbol{\omega}_i \\
&= \int_{\Omega_n} L(x \leftarrow \boldsymbol{\omega}_i) f_{r,S}(x, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o) (\mathbf{n} \cdot \boldsymbol{\omega}_i) d\boldsymbol{\omega}_i + f_{r,d}(x) E_n(x)
\end{aligned} \tag{4.4}$$

où $f_{r,S}$ représente la composante spéculaire (pas nécessairement spéculaire parfaite) de la BRDF et $f_{r,d}$ la composante lambertienne (donc le cas diffus uniforme). Puisque la composante diffuse est constante, on peut la sortir de l'intégrale. La quantité restante est appelée **irradiance** au point x , souvent notée $E_n(x)$, et qui dépend de la normale \mathbf{n} en x . Dans le cas d'une BRDF diffuse, il y a donc un résultat exact qui permet en fonction de l'irradiance de calculer la radiance réfléchie :

$$L(x \rightarrow \boldsymbol{\omega}_o) = f_{r,D}(x)E(x) \quad \text{avec} \quad E(x) = \int_{\Omega_n} L(x \leftarrow \boldsymbol{\omega}_i) (\mathbf{n} \cdot \boldsymbol{\omega}_i) d\boldsymbol{\omega}_i. \tag{4.5}$$

C'est là toute la stratégie des premiers schémas de *caching* : précalculer $E(x)$ à différents endroits et l'interpoler aux autres parce que recalculer l'intégrale de l'irradiance pour chaque pixel est extrêmement coûteux si l'on veut éliminer le bruit.

Le *caching* est énormément utilisé dans les logiciels commerciaux pour obtenir plus rapidement un rendu en éclairage global d'une scène. La technique la plus utilisée est celle proposée par Ward et Heckbert [WH92] (ou une de ses dérivées) que nous détaillerons dans la prochaine section.

Pour résumer, les points importants d'un algorithme de *caching* sont :

- la robustesse à reconstruire le type d'éclairage incident ainsi que son indépendance par rapport à la BRDF. Ceci est directement lié à la quantité radiométrique stockée et à la représentation utilisée. A la fin de cette section, nous passons en revue les différentes représentations possibles pour l'éclairage.

- la robustesse lorsqu'on augmente les détails géométriques.
- la continuité du schéma d'interpolation qui lui est associée.
- la consommation mémoire et l'adaptabilité vis-à-vis de la taille de la scène utilisée.

4.5.2 Travaux antérieurs en *caching*

Photon Mapping

Le *Photon Map* introduit par Jensen [Jen01] est une structure de cache qui stocke des photons. Les photons sont émis à partir des sources de lumière, propagés dans la scène et chaque impact sur les surfaces diffuses est stocké dans un *kd-tree*. Le *Photon Map* est une structure 3D qui «suit» la répartition des impacts sur les surfaces et donc la géométrie de la scène. L'efficacité de la recherche dépend énormément du choix effectué par l'utilisateur sur certains paramètres comme la taille maximale du voisinage et le nombre maximum de photons dans ce même voisinage. Lorsque l'éclairage indirect doit être évalué à une position donnée, une recherche de photons dans son voisinage est effectuée. La densité de photons permet alors de calculer l'éclairage indirect. Lorsqu'on accède directement au *Photon Map* et en fonction des paramètres affectés par l'utilisateur, du bruit de reconstruction peut apparaître. Pour remédier à ce problème, la structure de cache est interrogée de manière indirecte lors du rendu final ou *Final Gathering*, ce qui permet de filtrer le bruit par une passe supplémentaire de réflexion non-spéculaire. La figure 4.2 montre la différence de qualité obtenue entre un accès direct au *Photon Map* et avec une utilisation du *Final Gathering*.

On obtient, au prix d'un coût réduit mais toujours élevé de temps de rendu, une très bonne qualité de reconstruction de l'éclairage indirect. Afin de réduire le coût de calcul principalement lié à la recherche de photons voisins, le mécanisme de Christensen [Chr99] ou celui de Ward *et al.* [WRC88] (cf. ci-dessous) est utilisé.

Méthode de Ward *et al.*

La méthode de Ward *et al.* [WRC88] stocke l'irradiance en certains points de la scène. Dans une première passe, la position et le nombre d'échantillons sont déterminés à l'aide



Figure 4.2 – Rendus avec *Photon Mapping*. (a) Utilisation directe du *Photon Map*. (b) Utilisation du *Final Gathering*. Images tirées de [Jen96].

d'une fonction dépendante de critères géométriques (variation des normales et proximité d'autres objets) et du point de vue de la scène. Dans une seconde passe, l'irradiance est interpolée en un point en prenant les échantillons voisins dans le cache. La valeur stockée par échantillon est l'irradiance, grandeur dépendante de la normale et qui doit donc être recalculée dès que la normale varie.

L'albédo diffus peut donc être modifié dynamiquement, comme le montre l'équation (4.5), permettant ainsi une certaine édition avec une approximation car on ne prend pas en compte l'influence de la modification de l'albédo sur le pré-calcul. La continuité de la reconstruction du schéma original de Ward *et al.* n'est pas garantie parce que les échantillons ne sont pas organisés de manière spatiale. L'utilisation de gradients rotationnels et translationnels [WH92] améliore la qualité de la reconstruction mais des discontinuités peuvent encore survenir, comme le montrent les images de la figure 4.3.

La fonction de positionnement des échantillons dépend notamment de la variation de la normale, ce qui pose des problèmes de concentration des échantillons dès que de nombreux détails géométriques [TL04] sont présents dans la scène (cf. figure 4.4(c)).

Extension de Tabellion et Lamorlette à la méthode de Ward *et al.*

Le mécanisme proposé par Tabellion et Lamorlette [TL04] vise à améliorer le positionnement des échantillons, et notamment à éviter une accumulation dans les coins en modifiant la fonction de positionnement des échantillons, comme le montrent les fi-

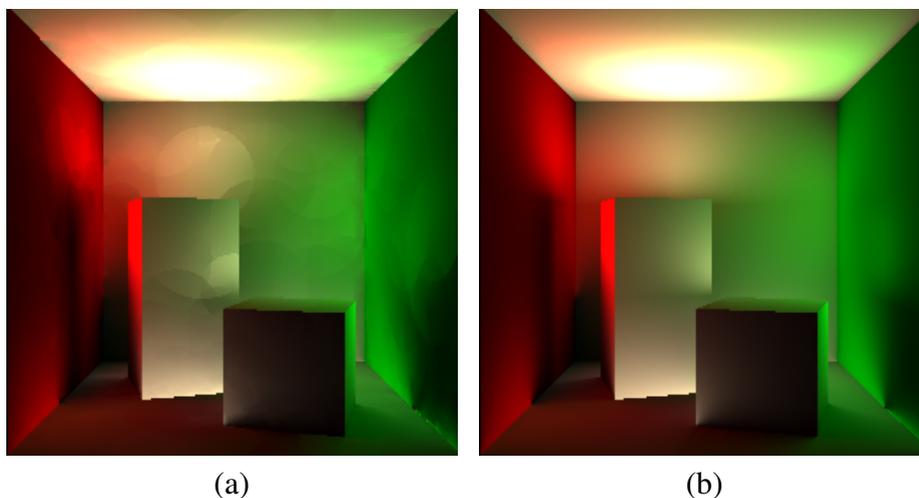


Figure 4.3 – Comparaison entre de l'éclairage indirect reconstruit avec : (a) gradients de Ward *et al.* et (b) approximation de meilleure qualité obtenue en échantillonnant l'hémisphère. Images tirées de [KGPB05].

gures 4.4(a) et (b). De plus, lors de la première étape, ils n'utilisent pas la géométrie détaillée par *bump maps* ou des *displacement maps*² mais une version simplifiée (un maillage de basse résolution) pour limiter le nombre d'échantillons. Cependant, la fonction de positionnement des échantillons reste dépendante des critères géométriques et ne limiterait donc pas le nombre élevé d'échantillons sur un maillage de haute définition.

Il semblerait aussi que Tabellion et Lamorlette stockent non seulement une valeur d'irradiance mais également une direction de lumière incidente moyenne afin de pouvoir utiliser n'importe quelle BRDF (bien que cela soit physiquement incorrect pour certaines BRDFs non-diffuses). La continuité du schéma de reconstruction semble grandement améliorée. Cependant, il est difficile de prouver que le schéma d'interpolation garantisse une continuité supérieure à C^0 .

Enfin, précisons que les techniques basées sur le critère de Ward *et al.* présentent l'avantage d'être consistantes, c'est-à-dire que l'augmentation du nombre d'échantillons fait tendre la reconstruction vers la solution exacte. Si chaque pixel possède un échantillon stockant l'irradiance, alors aucune interpolation n'est effectuée et ceci revient à

²Le *bump map* et le *displacement map* sont des techniques qui simulent des détails géométriques à l'aide d'une texture 2D sur les normales ou sous forme d'une fonction analytique de déplacement de la surface.

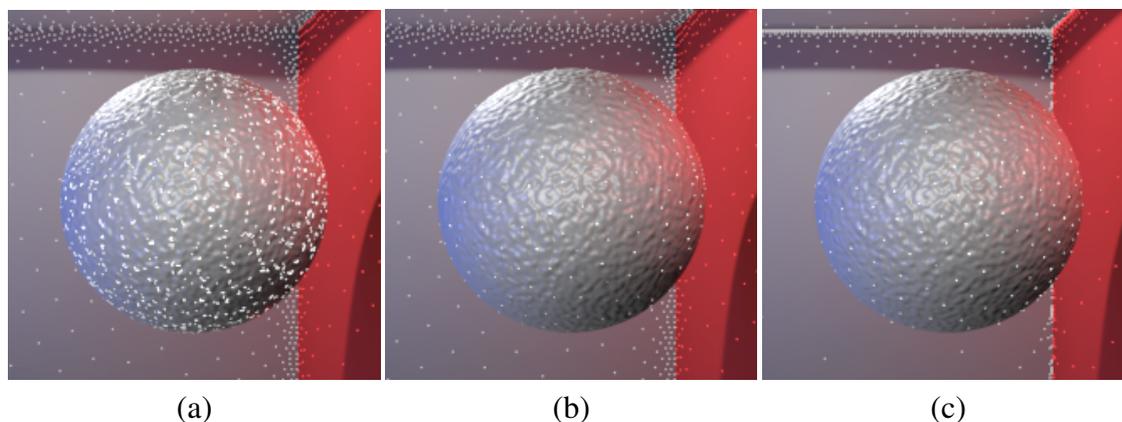


Figure 4.4 – Comparaison du positionnement des échantillons dans le cache en fonction du critère utilisé. (a) Critère de Tabellion et Lamorlette en utilisant la géométrie détaillée. (b) Critère de Tabellion et Lamorlette en utilisant la géométrie faiblement détaillée. (c) Critère de Ward *et al.* Images tirées de [TL04].

calculer la solution exacte de l'irradiance.

Systèmes stockant la radiance incidente

Comme mentionné ci-dessus, stocker l'irradiance n'est physiquement correct que si la BRDF est diffuse. Récemment, des algorithmes stockant la radiance ont été développés pour permettre d'utiliser des BRDFs plus *glossy*. Comme la radiance est uniquement valable pour un angle solide donné, il faut une représentation adéquate dépendante des directions sur la sphère.

Arikan *et al.* [AFO05] projettent la radiance dans la base des harmoniques sphériques. Leurs échantillons sont disposés dans un volume autour de la surface. L'interpolation lors de la reconstruction n'est pas garantie (pour les mêmes raisons que pour le schéma de Ward *et al.*) et surtout le critère de convergence n'est pas facilement contrôlable, voire pire, il augmente les discontinuités dans l'image finale.

Le schéma de Krivanek *et al.* [KGPB05] utilise également les harmoniques sphériques ou hémisphériques. Dans le cas de BRDFs *glossy*, Krivanek *et al.* [KBPv06] redistribuent la position des échantillons pour mieux répartir l'erreur visuelle. Ceci a pour effet d'améliorer la qualité visuelle de la reconstruction comme le montrent les images de la figure 4.5.

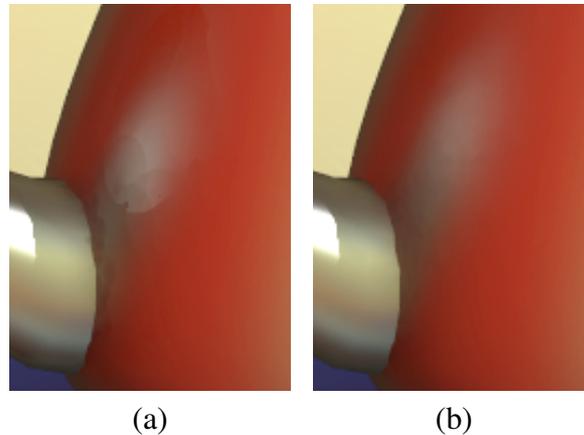


Figure 4.5 – Comparaison de la qualité visuelle obtenue entre : (a) méthode de Ward et Heckbert [WH92] stockant l’irradiance. (b) Méthode de Krivanek *et al.* [KBPv06] stockant la radiance. Images tirées de [KBPv06].

Cependant, les schémas d’interpolation d’Arikan *et al.* et de Krivanek *et al.* ne positionnent pas leurs échantillons de manière régulière et la continuité de la reconstruction n’est pas garantie. Malheureusement, aucune de ces deux techniques ne peut être utilisée dès que la scène contient trop de détails géométriques.

Autres systèmes de cache et dérivés

Préalablement, Greger *et al.* [GSHG98] ont proposé de stocker les échantillons dans un volume sur une grille régulière, appelé *Irradiance Volume*. Contrairement au nom de la structure, c’est bien la radiance incidente qui est stockée. Comme les échantillons sont stockés sur une grille régulière suivant le type de filtre utilisé, la reconstruction peut être C^0 , C^1 ou encore C^2 . Greger *et al.* discrétisent la sphère sur 578 directions et stockent pour chacune une valeur de radiance. L’utilisation d’un volume a également été proposée par Chiu *et al.* [CZS96]. Leur *Light Volume* est très proche et précurseur de l’*Irradiance Volume* de Greger *et al.* et stocke bien des valeurs de radiance. On peut voir le *Light Volume* et l’*Irradiance Volume* comme une forme d’arbre 5D (trois dimensions pour la localisation spatiale et deux dimensions pour la localisation directionnelle) tel que proposé par Lafortune et Willems [LW95]. A la différence des structures traditionnelles de cache, celle de Lafortune et Willems est uniquement utilisée pour faire de l’importance

sampling. Enfin, Christensen et Batali [CB04] ont également proposé une structure volumique mais adaptative utilisant des grilles de taille fixe plongées dans un octree. La taille de la structure est intéressante comparée à plusieurs *Photon Maps*, mais reste prohibitive comparée à celle de Greger *et al.* sans compter les temps de rendu.

Représentations en éclairage global

Dans ce paragraphe, nous résumons les principales représentations utilisées en éclairage global. Nous avons déjà présenté au chapitre 1 les représentations pour les BRDFs. Similairement, pour l'éclairage incident, les harmoniques sphériques sont aussi beaucoup utilisées pour les PRTs ou dans les systèmes de *caching*. Les harmoniques sphériques sont simples à utiliser mais sont sujettes aux effets de Gibbs lorsque l'on cherche à les utiliser pour compresser des signaux de hautes fréquences. Les ondelettes n'ont pas ce problème mais leur utilisation sur la sphère (e.g. [SS95]) reste complexe et, bien souvent, on préfère utiliser leur version 2D, par exemple pour représenter l'éclairage incident d'une carte d'environnement. Les bases de fonctions radiales sur la sphère ou *Spherical Radial Basis Functions* (SRBFs) sont une alternative aux ondelettes et aux harmoniques sphériques. Elles ont été utilisées dans le domaine des PRTs (e.g. [TS06]). La figure 4.6 présente les différences de qualité obtenue entre les différentes bases de fonctions pour une carte d'environnement. Bien que les SRBFs semblent très prometteuses, elles restent peu utilisées probablement à cause de la lenteur de leur projection dans la base.

Notons que, le plus souvent, toutes ces représentations sont utilisées pour stocker l'irradiance ou la radiance sous forme de scalaires. Il existe quelques cas d'utilisation de grandeurs vectorielles. Citons notamment les *Light Vectors* [ZSP98, SP01] et, récemment, le système de cache proposé par Lehtinen *et al.* [LZT⁺08]. Bien qu'aucune comparaison rigoureuse n'ait été faite entre les représentations vectorielles et scalaires, les représentations vectorielles semblent être plus robustes [LZK⁺07] que les scalaires. Enfin, signalons qu'il existe des bases vectorielles telles que les harmoniques sphériques vectorielles (e.g. [Arf05]), généralisation des harmoniques sphériques classiques, qui ne sont pas utilisées en infographie.

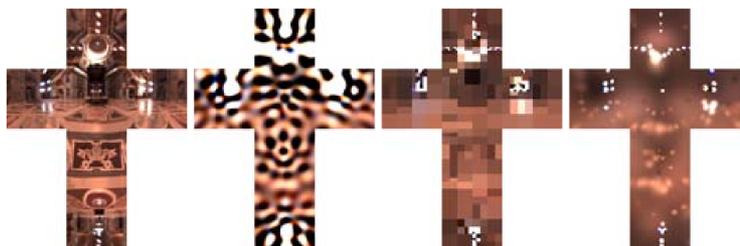


Figure 4.6 – (gauche à droite) Comparaison de la projection d’une carte d’environnement en utilisant la base des harmoniques sphériques, celle des ondelettes de Haar et celle des *Spherical Radial Basis Functions*. Pour chaque image le même nombre de coefficients (972) a été utilisé. La reconstruction à l’aide d’harmoniques sphériques peine à reconstruire les hautes fréquences de la carte d’environnement : l’effet de Gibbs est visible. Les ondelettes de Haar font apparaître un effet discontinu : les blocs sont visibles. Les *Spherical Radial Basis Functions* ont un comportement plus continu. Images tirées de [TS06].

4.6 Conclusion

Dans ce chapitre, nous avons introduit les principales notions et techniques qui tournent autour de l’éclairage global. Parmi elles, nous avons détaillé les systèmes de cache ainsi que les représentations utilisées pour l’éclairage incident. Nous avons mis en évidence que les systèmes proposés sont peu robustes aux fortes variations géométriques contenues dans une scène. Dans le prochain chapitre, nous proposons une nouvelle structure qui vise à résoudre ce problème. Nous avons également présenté les différentes méthodes pour éditer l’éclairage et nous avons montré que le domaine de l’édition reste peu exploré comparé à celui du rendu. De plus, les systèmes proposés ont tous pour but de modéliser les sources de lumière ou les BRDFs de la scène. Dans le chapitre 6, nous proposons une approche novatrice où la modification de l’éclairage incident a pour but de faire ressortir les détails géométriques de la surface de l’objet.

CHAPITRE 5

STRUCTURES POUR L'ÉCLAIREMENT INDIRECT

Dans ce chapitre, nous nous intéressons à une nouvelle représentation de l'éclairage indirect dans le contexte du *caching*. Comme expliqué au chapitre précédent, les systèmes de cache présentent encore des lacunes, notamment lorsqu'il s'agit d'être utilisés dans des scènes contenant de nombreux détails géométriques. Notre objectif est d'obtenir une représentation robuste aux variations géométriques et facilement utilisable avec un accès peu coûteux. Nous optons pour une solution qui reconstruit l'éclairage de manière lisse, préférant une solution biaisée mais visuellement plus plaisante qu'une solution bruitée.

A la lecture de ce qui a été fait précédemment dans le domaine, nous pensons qu'il faut un système de cache qui :

- garantisse une continuité au moins C^1 dans la reconstruction de l'éclairage
- utilise une quantité radiométrique qui permette une bonne mise à l'échelle lorsque les détails géométriques dans une scène augmentent drastiquement
- utilise une base de reconstruction/projection qui autorise une certaine indépendance par rapport à la BRDF utilisée (dans l'idéal du cas diffus au cas spéculaire).

5.1 Aperçu du système proposé

Le système de cache proposé est basé sur l'utilisation d'une grille régulière qui encode l'éclairage incident. Plus précisément, l'éclairage incident indirect est pré-calculé lors d'une première étape et stocké sur chaque sommet de la grille. Lors de l'étape de rendu, la grille est interrogée pour chaque point de la scène dont on cherche à connaître l'éclairage indirect. L'éclairage indirect pourra alors être reconstruit par interpolation. Nous utilisons plusieurs types de grilles en fonction du type d'éléments contenus dans la scène afin d'améliorer la qualité de reconstruction de l'éclairage. Ainsi, des grilles 2D sont utilisées pour les objets de type planaire ou quasi-planaire

(plan et *bump map* ou *displacement map*) tandis que des grilles 3D sont utilisées pour les objets pleins (e.g. statues) ou semi-pleins (e.g. arbres). Ces grilles sont définies de manière locale, *i.e.*, dans le référentiel de l'objet auquel elles sont rattachées et elles sont immergées dans une grille 3D globale à toute la scène (cf. figure 5.1).

En résumé, le système de *caching* que nous proposons possède les caractéristiques suivantes :

1. Une grille 2D ou 3D régulière pour pouvoir facilement effectuer une interpolation spatiale qui sera au moins C^1 .
2. Une grandeur radiométrique vectorielle (vecteur d'irradiance) robuste pour s'abstraire de la normale.
3. Une indépendance par rapport à la BRDF sous-jacente.
4. Une base de vecteurs d'irradiance stockée aux sommets de la grille qui permet de reconstruire l'irradiance pour n'importe quelle normale sous-jacente.
5. Une interpolation directionnelle en chaque point de l'espace afin d'avoir à tous les endroits une reconstruction C^1 de l'éclairage incident.

Ce chapitre est organisé comme suit. A la section 5.2 nous présentons en détail nos structures ainsi que la base de vecteurs d'irradiance. Nos résultats sont présentés à la section 5.3 sous forme de trois cas pratiques d'utilisation :

- Système de cache pour le rendu temps réel
- Transfert progressif (*Streaming*) de l'éclairage dans un contexte de rendu distribué
- Système de cache pour le rendu de haute qualité.

Nous concluons ce chapitre en proposant des optimisations (cf. section 5.4) ainsi que des pistes (cf. section 5.5) de recherche pour de futurs travaux. Notre approche a été validée pour les deux premières applications par deux publications [PRG⁺08, PRL⁺08] et la troisième par un rapport de recherche [PGSP09].

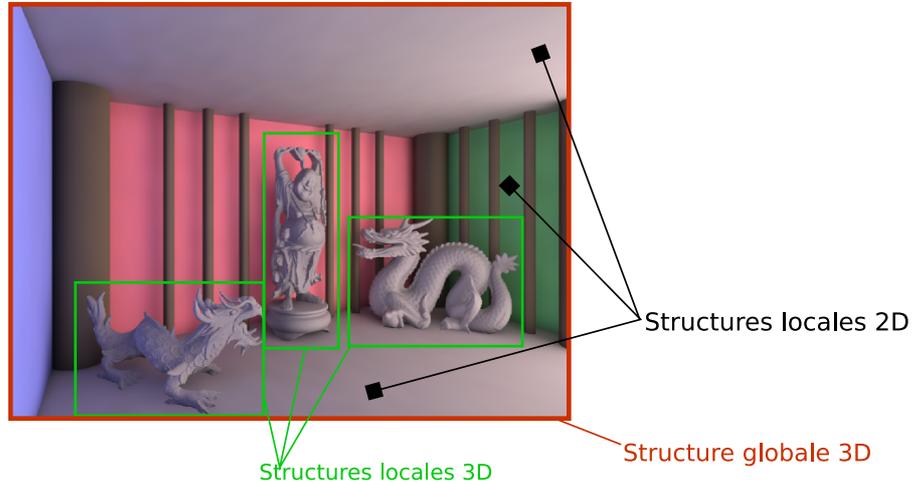


Figure 5.1 – Illustration du type de structures utilisées en fonction du type d’objet d’une scène. En plus de la structure globale (en rouge) de la scène, des structures locales 3D (en vert) sont utilisées pour les objets volumineux tandis que des structures locales 2D sont utilisées pour les objets de type planaire.

5.2 Grille de vecteurs d’irradiance (*Irradiance Vector Grid*)

Vecteur d’irradiance

Nous choisissons de stocker à chaque sommet de la grille un vecteur d’irradiance. Pour une longueur d’onde donnée, le vecteur d’irradiance, tel qu’introduit par Arvo [Arv94], est défini pour un point p et une normale \mathbf{n} par :

$$\mathbb{R}^3 \times \Omega \rightarrow \mathbb{R} : \mathbf{I}_n(\mathbf{p}) = \int_{\Omega_n} L(\mathbf{p} \leftarrow \boldsymbol{\omega}_i) \boldsymbol{\omega}_i d\boldsymbol{\omega}_i \quad (5.1)$$

où $L(\mathbf{p} \leftarrow \boldsymbol{\omega}_i)$ représente la radiance incidente au point \mathbf{p} provenant de la direction $\boldsymbol{\omega}_i$, $d\boldsymbol{\omega}_i$, l’angle solide différentiel généré par $\boldsymbol{\omega}_i$, et Ω_n , l’hémisphère centré en \mathbf{p} et orienté vers \mathbf{n} . Le vecteur d’irradiance représente une information à la fois géométrique et radiométrique et il est directement relié à la radiance réfléchie dans le cas diffus par :

$$L_r(\mathbf{p} \rightarrow \boldsymbol{\omega}_o) = \frac{\rho_D}{\pi} (\mathbf{I}_n(\mathbf{p}) \cdot \mathbf{n}) \quad (5.2)$$

où ρ_D représente l’albédo diffus de la BRDF au point \mathbf{p} et « \cdot » le produit scalaire. Intuitivement, un vecteur d’irradiance représente l’intensité de l’éclairage incident et la

direction moyenne d'où il provient. Le principal avantage du vecteur d'irradiance réside dans le fait que, pour une variation locale de la normale, la radiance réfléchie peut être ajustée, rendant ainsi la représentation plus **robuste aux variations géométriques**. Remarquons que nous voulons calculer la radiance réfléchie $L_r(\mathbf{p})$, où la normale au point \mathbf{p} peut être quelconque. Il nous faut donc une représentation qui stocke l'éclairement incident pour n'importe quelle direction. Ainsi, nous subdivisons l'espace des directions en six hémisphères où chaque hémisphère est orienté selon une direction principale $\boldsymbol{\delta} = \pm\mathbf{x} | \pm\mathbf{y} | \pm\mathbf{z}$. Nous pré-calculons un vecteur d'irradiance $I_{\boldsymbol{\delta}}$ pour chaque hémisphère pour représenter l'éclairement incident.

Pré-calcul des vecteurs d'irradiance

Pour évaluer l'équation 5.2, il est possible d'utiliser n'importe quel algorithme d'éclairage global comme le tracé de photons (*Photon Tracing* [Jen01]) ou le tracé de rayons (*Path Tracing* [Kaj86]). Nous avons expérimenté avec les deux approches.

Pour utiliser le tracé de photons, nous choisissons de propager les photons depuis la lumière à travers la grille en s'assurant qu'ils aient subi au moins une réflexion. A chaque fois qu'un photon traverse une face d'un voxel de la grille, sa contribution est ajoutée au vecteur d'irradiance $I_{\boldsymbol{\delta}}^{ijk}$ associé au sommet de la grille \mathbf{v}^{ijk} le plus proche et pour une direction $\boldsymbol{\delta}$ donnée par la normale de la face (cf. figure 5.2). Lorsqu'un photon intersecte la géométrie de la scène, une réflexion stochastique, à l'aide de la roulette russe, est utilisée afin de déterminer si le photon est absorbé ou s'il est réfléchi de manière cohérente à la BRDF.

Le tracé de photons est effectué en deux étapes. La première utilise une structure d'accélération pour trouver l'intersection la plus proche entre le photon et la scène. Ensuite, nous propageons dans la grille 3D (pour une grille 2D cette étape est inutile) la contribution du photon sans effectuer aucun test d'intersection. Une fois tous les photons tracés, nous effectuons une passe de normalisation, similaire à une estimation de

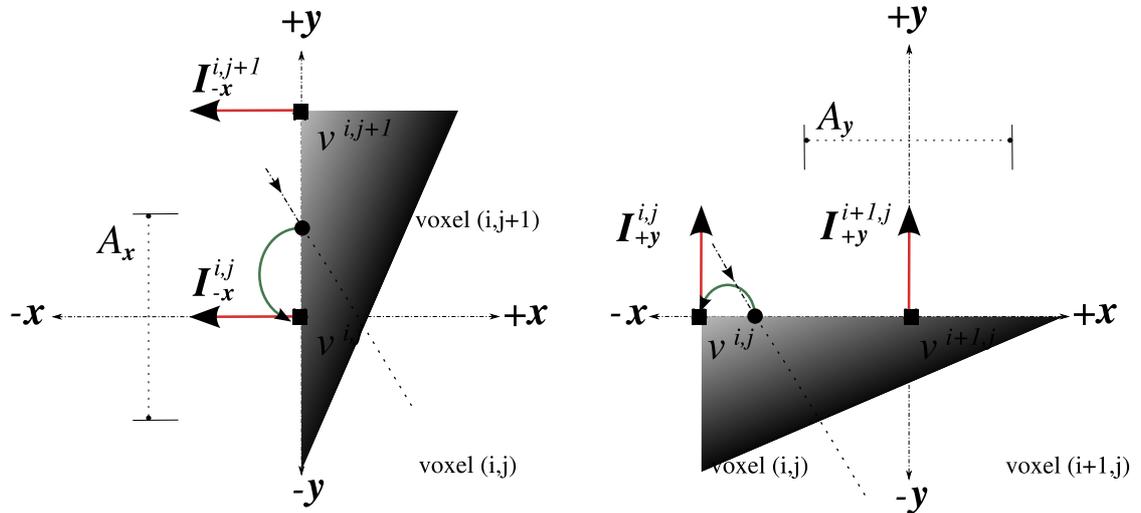


Figure 5.2 – Illustration 2D du calcul des vecteurs d'irradiance à l'aide d'un tracé de photons. Lorsqu'un photon intersecte une face d'un voxel, sa contribution est ajoutée au vecteur d'irradiance associé au sommet le plus proche de la grille. La contribution du photon est ajoutée au sommet $I_{-x}^{i,j}$ (gauche) et $I_{+y}^{i,j+1}$ (droite).

densité, sur les vecteurs d'irradiance :

$$I_{\delta}^{ijk} = \frac{1}{A_{\delta}^{ijk}} \sum_{l=1}^{N_{\delta}^{ijk}} \phi_l \omega_l$$

où :

1. N_{δ}^{ijk} représente le nombre de photons qui ont contribué au vecteur d'irradiance du sommet v^{ijk} dans la direction δ .
2. A_{δ}^{ijk} représente l'aire de la face du voxel centrée en v^{ijk} et orientée selon δ .
3. ϕ_l représente la puissance du l -ième photon.
4. ω_l représente la direction du l -ième photon.

Les deux seuls cas où l'aire n'est pas celle de la face du voxel sont pour les sommets aux coins de la grille ; dans ce cas, l'aire doit être divisée par 4 ainsi que les sommets sur les bords de la grille où l'aire doit être divisée par 2. Remarquons que notre approche ne souffre pas du biais sur les bords assez classique lorsqu'on utilise le *Photon Map*. Ce biais se manifeste notamment dans les coins d'une scène où la recherche des photons

les plus proches peut inclure des photons provenant d'une géométrie différente de celle dont on cherche à calculer l'éclairage. Enfin, contrairement à l'approche d'Havran *et al.* [HBHS05], notre solution ne nécessite pas de stocker tous les rayons issus de la propagation des photons.

Pour calculer les vecteurs d'irradiance, il est aussi possible d'utiliser une technique de Monte-Carlo du type *Path Tracing*. Pour ce faire, pour chaque sommet, on échantillonne de manière uniforme l'ensemble des directions possibles sur la sphère. Il est à noter que chaque rayon tiré contribue au maximum à trois vecteurs d'irradiance. Etant donnée la simplicité des structures utilisées, cette approche peut être massivement parallélisée sur une machine multi-coeurs ou encore sur un *cluster* de PCs. Comme illustré à la figure 5.3, un problème qui se pose lorsqu'on utilise cette approche est de ne pas échantillonner des sommets qui se trouvent à l'intérieur ou qui sont trop proches de la surface. Pour rendre le calcul plus robuste, nous perturbons légèrement la position initiale du sommet dans la direction de la normale. comme proposé par Arikan *et al.* [AFO05]. Comme un sommet peut être à l'intérieur de l'objet, nous choisissons d'ignorer les intersections des rayons qui se produisent à l'intérieur de la géométrie. De cette façon, tous les sommets de la grille possèdent une valeur d'éclairage cohérente.

Interpolation spatiale et directionnelle

Afin de reconstruire de manière continue l'éclairage indirect, nous interpolons un vecteur d'irradiance pour chaque point $\mathbf{p} = (p_x, p_y, p_z)$ avec normale $\mathbf{n} = (n_x, n_y, n_z)$ dont on cherche à calculer la radiance. Cette interpolation s'effectue en deux étapes : une interpolation spatiale relative à la position de \mathbf{p} dans la grille et ensuite une interpolation directionnelle relative à la normale \mathbf{n} .

Pour l'interpolation spatiale, le vecteur d'irradiance $\mathbf{I}_\delta(\mathbf{p})$ est obtenu par interpolation tri-linéaire ou tri-cubique des vecteurs d'irradiance stockés sur les sommets de la grille autour de \mathbf{p} .

L'interpolation directionnelle est uniquement effectuée à l'aide de trois directions δ parmi les six possibles. Le choix entre $\pm\mathbf{x}$ (resp. $\pm\mathbf{y}$ et $\pm\mathbf{z}$) est fait par rapport au signe de la normale n_x (resp. n_y and n_z). Ces trois directions sont alors combinées à l'aide

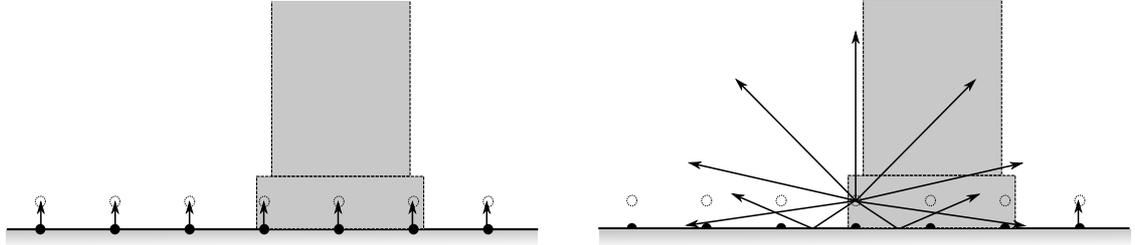


Figure 5.3 – Stratégie d'échantillonnage pour les grilles. (gauche) Une légère translation dans la direction de la normale est effectuée sur les sommets de la grille. (droite) Certains échantillons peuvent désormais être à l'intérieur de l'objet. On ne tient donc pas compte de la première intersection des rayons partant de ces sommets dans le calcul de l'éclairage indirect.

d'une interpolation sur la sphère des directions pour reconstruire le vecteur d'irradiance $I_n(\mathbf{p})$:

$$I_n(\mathbf{p}) = I_x(\mathbf{p})n_x^2 + I_y(\mathbf{p})n_y^2 + I_z(\mathbf{p})n_z^2.$$

Remarquons que notre interpolation est exacte lorsque la direction de la normale est alignée avec $\boldsymbol{\delta}$. De plus, cette interpolation peut aussi bien être effectuée sur CPU que sur GPU.

Compression

Afin de réduire la place mémoire occupée par nos structures sur la carte graphique, nous compressons les vecteurs d'irradiance avant transmission au GPU. En effet, l'équation 5.1 définit un vecteur d'irradiance pour une longueur d'onde donnée. En infographie, nous utilisons en pratique une définition colorée avec trois composantes (R, G, B), et nous devons alors stocker pour chaque direction $\boldsymbol{\delta}$, trois vecteurs d'irradiance sous la forme d'une matrice 3×3 :

$$\mathbf{M} = (\mathbf{I}_{\boldsymbol{\delta}}^R | \mathbf{I}_{\boldsymbol{\delta}}^G | \mathbf{I}_{\boldsymbol{\delta}}^B).$$

Pour un sommet donné de la grille, nous compressons \mathbf{M} sous la forme d'un produit scalaire entre une direction \mathbf{d} et une couleur \mathbf{c} définies comme suit :

$$\mathbf{d} = \frac{\mathbf{I}^R + \mathbf{I}^G + \mathbf{I}^B}{\|\mathbf{I}^R + \mathbf{I}^G + \mathbf{I}^B\|} \quad \mathbf{c} = \left[\frac{\mathbf{I}^R \cdot \boldsymbol{\delta}}{\mathbf{d} \cdot \boldsymbol{\delta}}, \frac{\mathbf{I}^G \cdot \boldsymbol{\delta}}{\mathbf{d} \cdot \boldsymbol{\delta}}, \frac{\mathbf{I}^B \cdot \boldsymbol{\delta}}{\mathbf{d} \cdot \boldsymbol{\delta}} \right].$$

Ceci garantit que lorsque la normale \mathbf{n} est alignée avec δ , nous préservons les intensités RGB originales : $\mathbf{Mn} = \mathbf{c}$. Expérimentalement, nous avons vérifié visuellement que cette compression n'introduit pas d'artefacts supplémentaires dans l'interpolation de l'éclairage indirect.

Afin de réduire encore plus l'utilisation de la bande passante CPU-GPU, il est également possible de quantifier la direction sur 24 bits en utilisant une quantification classique de 8 bits pour chacune des trois coordonnées de la direction et 32 bits pour la couleur. Pour la couleur, il est plus efficace d'utiliser le format R9_G9_B9_E5 qui permet de compresser une couleur HDR de manière similaire au format RGBE proposé par Ward [War91] et qui est supporté par les GPU modernes¹.

Echantillonnage spatial

Comme illustré à la figure 5.1, nous utilisons plusieurs types de grilles en fonction du type d'éléments contenus dans la scène afin d'améliorer la qualité de reconstruction de l'éclairage.

Notre structure 2D est une grille régulière positionnée légèrement en avant de la surface, le long de sa normale. Cette surface peut résulter d'une *bump map*, d'une *normal map* ou d'une *displacement map*. Dans le cas d'une *displacement map* analytique, le positionnement de notre structure peut être calculé de manière exacte en s'assurant qu'elle se trouve toujours en avant des points déplacés. De plus, pour éviter les problèmes d'échantillonnage de l'hémisphère liés à un positionnement dans un coin, les échantillons sont légèrement déplacés vers le centre de leur cellule (cas adaptatif) ou de la grille (cas régulier).

5.3 Applications et résultats

Dans cette section, nous présentons trois types d'applications pour notre structure de données.

¹Cette compression de couleur en grande dynamique (HDR) est supportée dans la version 10 de DirectX et en OpenGL avec l'extension *GL_EXT_texture_shared_exponent*.

Système de cache pour rendu interactif

En infographie, les scènes contiennent de plus en plus de détails géométriques provenant par exemple des scanners 3D. Calculer en temps réel l'éclairage indirect reste un challenge pour ce type de scène. Les systèmes de cache utilisent le fait que l'éclairage indirect dépende de la géométrie de la scène mais pas de tous les détails géométriques [WRC88]. Dès lors, une géométrie simplifiée suffit pour le pré-calcul de l'éclairage indirect et permet aussi de réduire le temps de pré-calcul. En revanche, il est important de pouvoir observer les détails géométriques de la scène pour les points de vue rapprochés.

Notre représentation est parfaitement adaptée pour ce type d'approche. Nous pré-calculons la structure en utilisant la géométrie simplifiée, mais à l'étape du rendu, nous réintroduisons les détails géométriques. Dans les figures 5.6 et 5.7, nous utilisons des grilles 2D pour les murs et des grilles 3D pour les différentes statues. Les détails géométriques sont stockés sous forme de textures de normales (*normal maps*) ou encore de textures 3D en utilisant les *Appearance-Preserving Octree* (APO [LBJS07]). Comme un APO est une structure volumétrique, il n'est pas nécessaire de paramétrer globalement l'objet. Une paramétrisation globale est un processus particulièrement coûteux lorsque l'objet contient un nombre élevé de triangles. Un APO encode une *normal map* dans une texture représentant un octree.

Cet octree peut être transformé en un atlas de textures 2D. Chaque triangle de la géométrie simplifiée est rendu dans une texture à l'aide d'un *fragment shader* qui associe une couleur à chaque fragment du triangle (cf. figure 5.4). Comme illustré dans la figure 5.7, qui compare les deux types de rendu (APO vs atlas), l'utilisation d'un atlas ne génère pas d'artefacts majeurs et permet d'améliorer grandement les performances. Le tableau 5.I compare les différentes performances de rendu pour les figures 5.6 et 5.7, et confirme l'intérêt de notre représentation combinée avec un atlas de textures ou un APO pour des scènes à géométries très complexes où l'utilisateur veut pouvoir changer le point de vue de manière interactive.

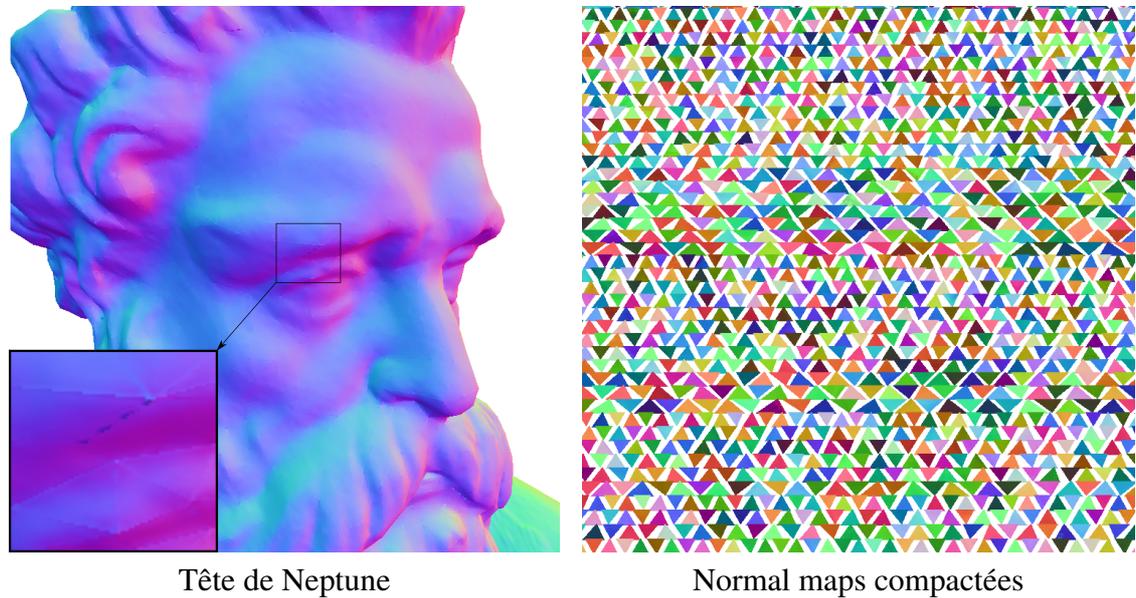


Figure 5.4 – Texture d’atlas non paramétrisée. (gauche) Sans paramétrisation globale, les normales sont compactées dans un atlas de textures. (droite) Les régions en blanc représentent des normales de surfaces non-définies. (Encadré en bas à gauche) Ceci introduit de légères discontinuités pour les normales reconstruites.

	Ramses	Neptune	David	Lucy
Originale/Simplifiée	1.6M/80K	4M/100K	7.2M/100K	15M/150K
APO	4 min	6 min	4 min	7 min
Texture	6 min	5 min	10 min	10 min

Figure 5.5 – Taille en polygones des objets originaux et de leur version simplifiée avec leur temps de pré-calcul pour les représenter sous la forme d’un APO et d’atlas de textures. Ces mesures ont été effectuées sur un processeur AMD 3500+ avec 2 GB de mémoire vive, et une NVIDIA GeForce 8800 GTS. Comme le pré-calcul peut utiliser jusqu’à 4 GB de mémoire, le temps passé par le système à transférer les données de la mémoire vive à celle de *swap* a été supprimé.

Le tableau 5.II indique les temps de pré-calcul et la consommation mémoire de nos grilles. Comparée à la place occupée par la géométrie, cette consommation mémoire reste faible, ce qui rend notre approche d'autant plus intéressante. La seule partie coûteuse reste le temps de pré-calcul de nos grilles.

Pour valider notre approche de pré-calcul de l'éclairage incident sur une géométrie simplifiée, nous comparons le rendu d'une scène pour les deux stratégies. Ainsi la figure 5.8 montre la localisation spatiale de l'erreur introduite par un pré-calcul sur géométrie simplifiée comparée à un pré-calcul avec la géométrie détaillée.

Nous avons également testé notre approche avec une grille 3D globale à toute la scène dans laquelle on peut déplacer une géométrie complexe. Évidemment, dans ce cas la géométrie est passive (elle n'influence pas l'éclairage de la scène) mais elle peut tout de même recevoir les effets de l'éclairage indirect comme illustré à la figure 5.9. Dans cette figure, le cube et le dragon reçoivent de l'éclairage indirect en provenance du mur vert. La figure met également en évidence la différence de qualité obtenue lorsqu'on utilise un schéma d'interpolation tri-cubique par rapport à un schéma tri-linéaire pour une grille contenant des vecteurs d'irradiance compressés.

Streaming d'éclairage indirect

Nous avons également expérimenté nos structures dans le cadre d'une application client/serveur pour laquelle la géométrie et l'éclairage sont transmis de manière progressive. Notre système exploite une séparation du calcul entre l'éclairage direct et indi-

	APO	Atlas de textures	Géométrie hautement détaillée.
Salle Lucy	35 fps	60 fps	3-8 fps
Salle méditerranéenne	30 fps	60 fps	5-8 fps

Tableau 5.I – Taux de rafraîchissement (fps) pour une taille écran de 800 × 800 pixels. Pour une scène donnée, les mêmes grilles sont utilisées pour l'éclairage indirect mais avec différentes représentations pour la géométrie de la scène. Mesures effectuées sur un processeur Intel Q6600 (4 GB de RAM) avec une carte graphique NVIDIA GeForce 8800 GTX.

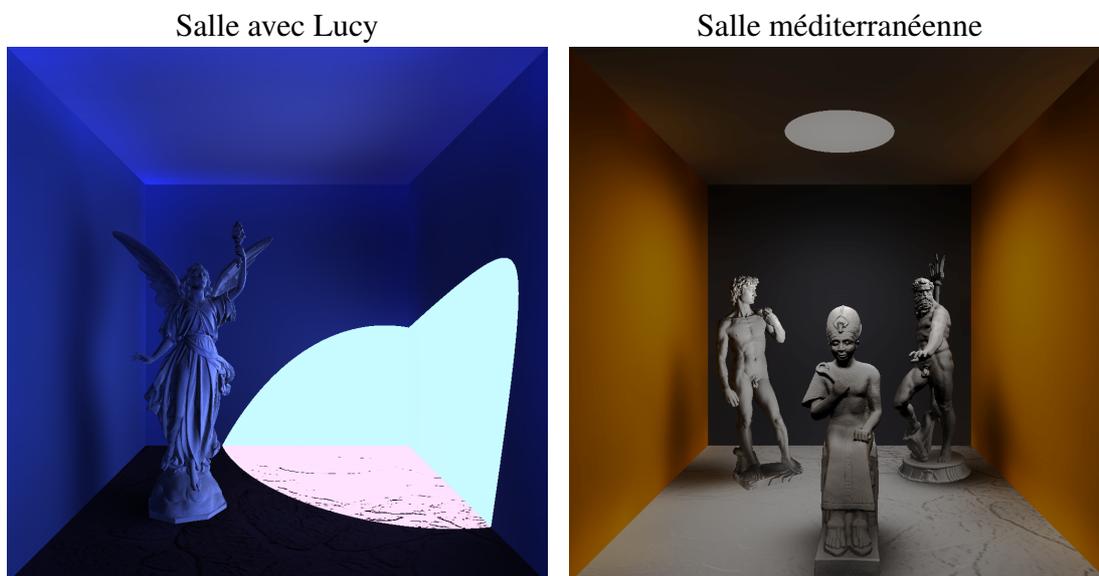


Figure 5.6 – Scènes de test pour les APOs et les grilles d’éclairage. La première scène est dédiée à la statue de Lucy éclairée de manière indirecte par un spot lumineux. Le sol est modélisé avec une normal map. La seconde scène est constituée de trois statues provenant de divers endroits et époques autour la mer méditerranée et d’un sol avec une normal map. Le tout est éclairé de manière indirecte par un spot lumineux orienté vers le plafond.

rect. L’éclairage direct est calculé en temps réel par le client en utilisant les ressources GPU disponibles tandis que l’éclairage indirect est évalué grâce aux grilles. Le serveur pré-calculé et stocke les structures d’éclairage (les grilles) ainsi que les niveaux de détails (LODs) de chaque géométrie complexe représentée sous la forme d’un maillage progressif. Le serveur envoie donc soit de la nouvelle géométrie soit un raffinement de l’éclairage. Après chaque réception de données, le client effectue des opérations sur les structures d’éclairage et géométriques avant de les envoyer sur le GPU. Notre approche permet d’utiliser différentes stratégies d’envoi de données tel que l’entrelacement entre les données d’éclairage et de géométrie (cf. figure 5.10). Ceci offre donc une visualisation progressive jusqu’à ce que la qualité désirée soit obtenue.

Une solution classique pour le transfert progressif d’une texture est d’utiliser une décomposition hiérarchique utilisant des bases de fonctions récursives comme les ondelettes. Cependant, pour reconstruire chaque niveau de la hiérarchie, cette technique de

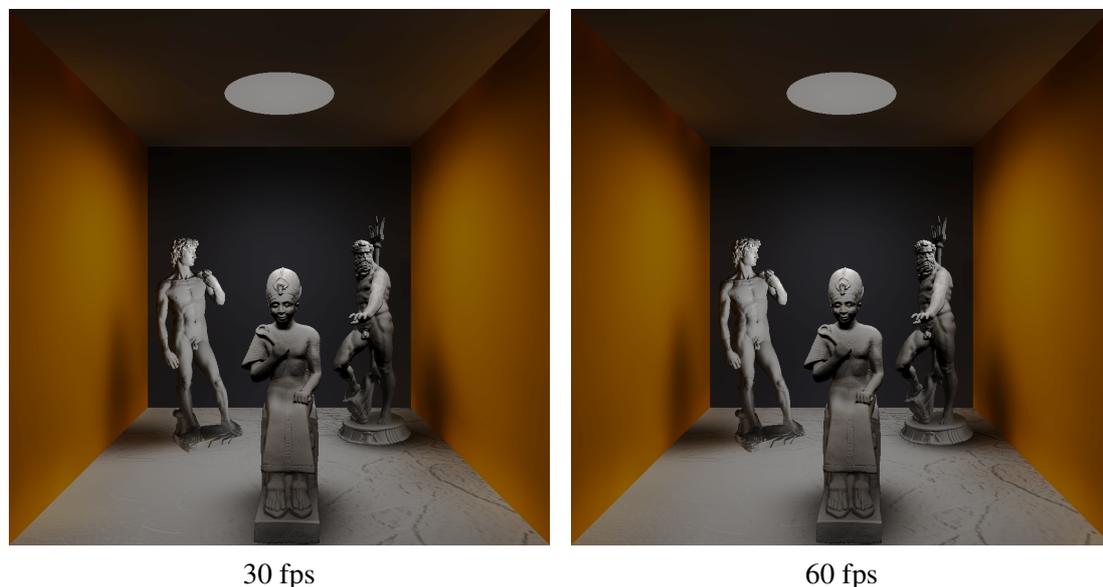


Figure 5.7 – Salle méditerranéenne. Rendu avec APO, à gauche, comparé avec un rendu avec des atlas de textures à droite. Aucune différence n'est visible sur les statues. Remarquez les réflexions de l'éclairage indirect des murs de droite et de gauche sur les statues. De plus, l'éclairage indirect suit les variations des normales.

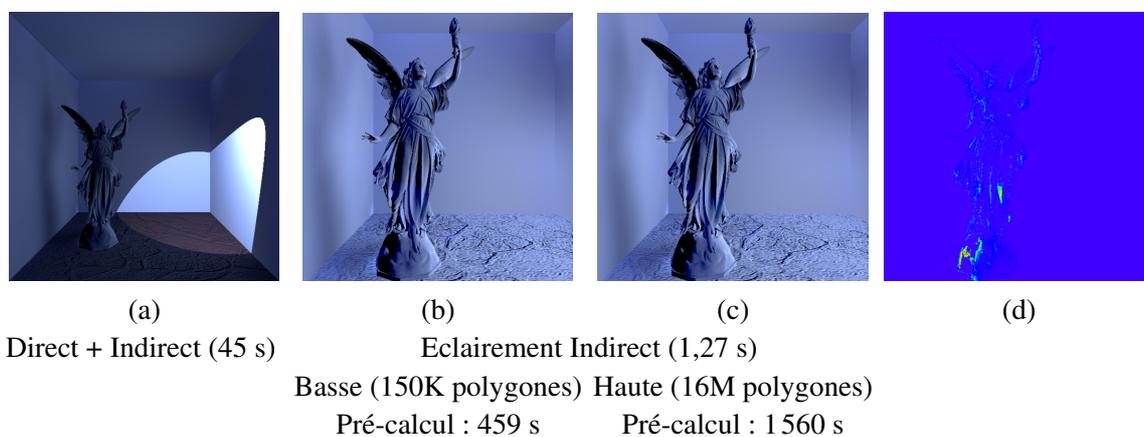


Figure 5.8 – Pré-calcul de l'éclairage indirect avec et sans géométrie simplifiée. La résolution de la grille 3D autour de la statue est de $8 \times 16 \times 8$. En (b) et (c) nous utilisons un rendu software pour mettre en évidence l'irradiance indirecte reconstruite à l'aide des grilles en utilisant respectivement la géométrie faiblement détaillée (b) et celle détaillée (c). (d) Calcul de différence dans l'espace de couleur *Lab* entre (b) et (c). Ce calcul montre que l'erreur maximale est de 69 (15% de l'erreur maximale possible) et très localisée. L'erreur moyenne est de 0,4 (0,09%).

	Salle avec Lucy	Salle méditerranéenne
Géométrie originale / simplifiée	15 M / 150 K	12,8 M / 280 K
APO	21 Mo	94 Mo
Grilles 3D	1 / 8×16×8 / 144 Ko	3 / 16×16×16 / 1.7 Mo
Grilles 2D	6 / 32×32 / 264 Ko	6 / 32×32 / 264 Ko
Pré-calcul de l'éclairage	55 min	260 min
Géométrie originale	3-8 fps	5-8 fps
APO	35 fps	30 fps

Tableau 5.II – Taille des différentes structures et fps moyen. La taille de la géométrie correspond au nombre de polygones. La taille des structures d'éclairage est faible comparée à celle de la géométrie. Nous avons utilisé une méthode stochastique avec 16000 rayons pour pré-calculer les vecteurs d'irradiance pour chaque sommet. Le taux de rafraîchissement a été évalué pour une image de 800×800 pixels. Pour une scène donnée, les mêmes grilles sont utilisées mais avec une représentation de la géométrie différente.

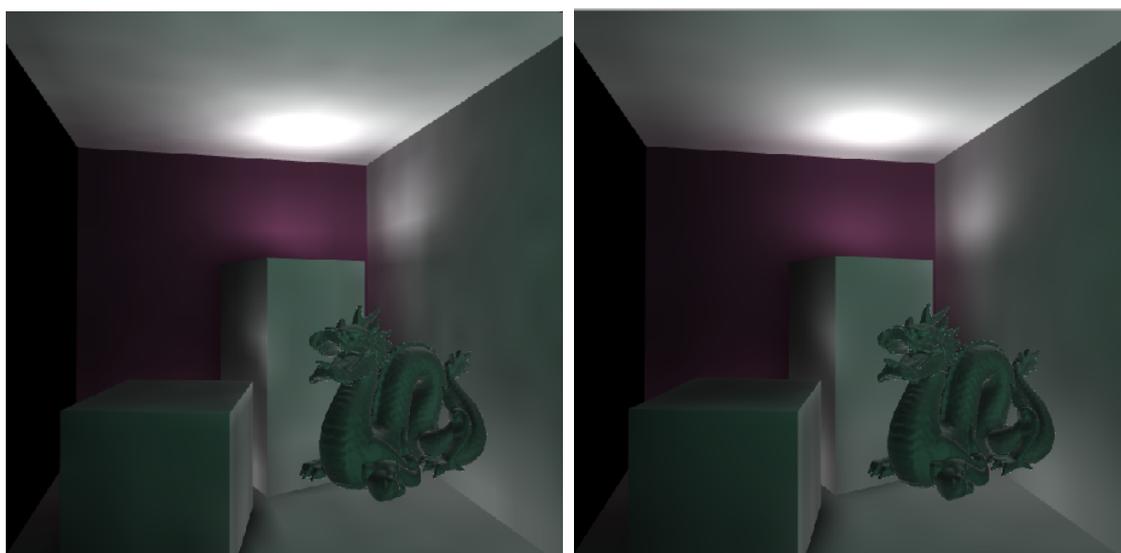
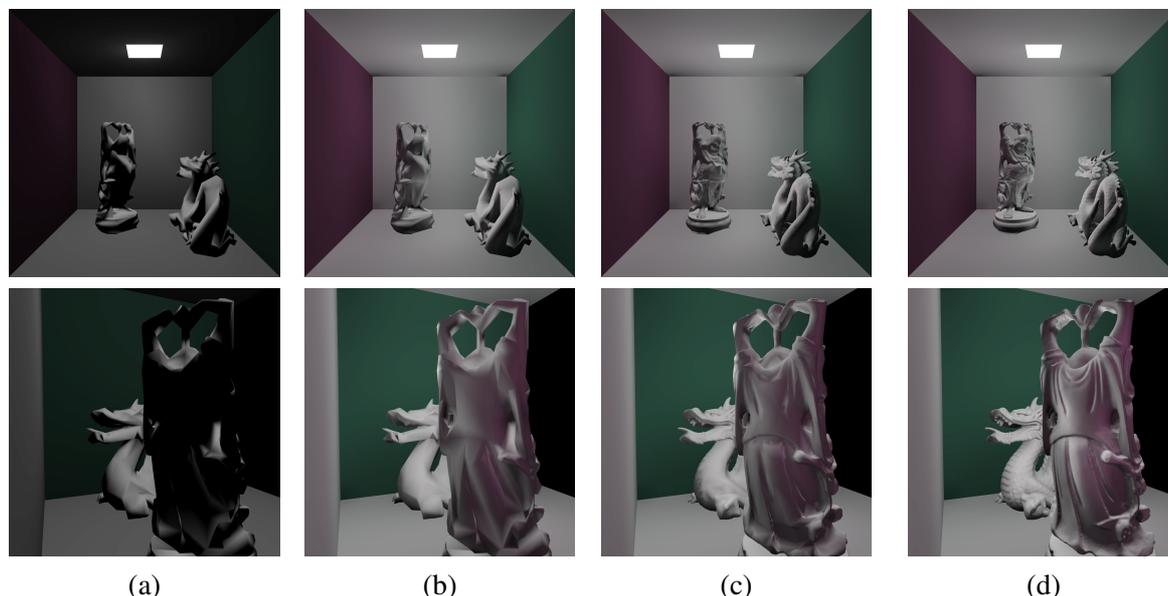


Figure 5.9 – Éclairage indirect sur GPU. La caméra et le dragon peuvent être déplacés avec une interpolation tri-linéaire (resp. tri-cubique) à 91 fps (resp. 87 fps).

décomposition nécessite d'attendre la réception de tous les coefficients correspondant au niveau de détails en cours de transmission. Ainsi, le temps requis pour obtenir un nouveau niveau de la hiérarchie augmente avec la taille du niveau. Nous proposons une approche alternative qui permet une transmission d'un groupe de voxels de taille fixe.

La transmission est initialisée en transférant les huit coins de la grille. Ensuite,



(a) 0,2% maillage, no IVG (b) 0,2% maillage, 25% IVG (c) 2% maillage, 25% IVG (d) 100% maillage, 100% IVG

Figure 5.10 – Notre système de visualisation client/serveur permet d’envoyer de manière progressive de la géométrie et de l’éclairage représenté sous forme de grilles de vecteurs d’irradiance. (a) Une scène avec 0.2% de la géométrie transférée et sans éclairage direct (aucune ombre). (b) La même quantité de géométrie mais avec 25% d’éclairage indirect : les effets de transferts de couleur entre le mur et la statue sont désormais visibles, le bouddha apparaît plus rouge. (c) Raffinement supplémentaire de la géométrie (2%). (d) Résolution maximale atteinte pour la géométrie (50 Mo) et la grille d’éclairage (1 Mo). Le taux de rafraîchissement pour une telle scène est de 50 fps avec une carte NVIDIA GeForce Go 7800 GTX .

chaque requête du client consiste en un nombre constant de vecteurs d’irradiance. Le nombre de vecteurs d’irradiance par requête peut être changé de manière dynamique en fonction des capacités CPU/GPU du client ainsi que la bande passante et la fiabilité du réseau utilisé. Afin d’obtenir une mise-à-jour cohérente de l’éclairage, nous avons implémenté un échantillonnage aléatoire stratifié de la grille. Dans notre implémentation, la grille 3D est divisée en plusieurs couches 2D le long du plus grand axe de la grille 3D. A chaque requête client, le serveur choisit de manière aléatoire sur chaque couche, le nombre de vecteurs d’irradiance demandés par le client.

Malheureusement, lors du transfert progressif de la grille 3D, des trous peuvent apparaître dans la reconstruction de l’éclairage, et ce, jusqu’à la transmission complète

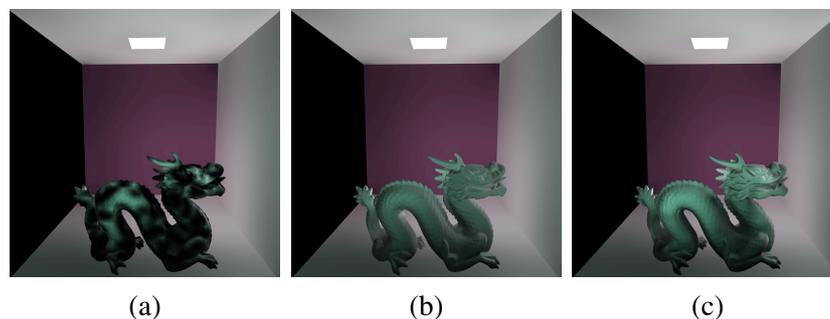


Figure 5.11 – Comparaison de l'éclairage indirect reconstruit sur le côté droit du dragon (a) sans l'algorithme du *push-pull*, (b) avec l'algorithme du *push-pull* et (c) avec filtrage pyramidal et sans filtrage pyramidal (c). La dimension de la grille est de $16 \times 16 \times 16$ et la moitié des sommets de la grille ont été transférés. Le *push-pull* a comblé les régions noires du dragon avec des valeurs d'éclairage continues.

de la grille. Ceci vient du fait que certains sommets de la grille n'ont pas encore reçu de valeur cohérente pour les vecteurs d'irradiance qui leur sont associés. Pour remédier à ce problème, nous avons choisi d'implémenter un algorithme de *push-pull* qui remplit de manière continue les données manquantes de la grille 3D. Les trous sont remplis avec un schéma d'interpolation des données déjà reçues (cf. figure 5.11). L'algorithme du *push-pull* peut être vu comme la version simplifiée d'une diffusion de poisson sans conditions à la frontière du domaine. Remarquons que notre version de l'algorithme ne modifie aucune donnée reçue. Afin d'obtenir des résultats encore plus lisses, un filtre pyramidal est appliqué à chaque étape de l'algorithme. Enfin, cette étape de *push-pull* est optionnelle si les capacités CPU du client sont limitées. Les caractéristiques minimales requises au niveau du matériel sont uniquement un GPU programmable avec support pour les textures 3D.

La direction et la couleur de chaque vecteur d'irradiance sont envoyées soit sous forme de nombres à virgule flottante, soit sous une forme quantifiée. Nos expérimentations montrent que la quantification réduit le temps de transfert d'un facteur environ égal à 2,5 sans introduire d'artefacts visibles. En effet, pour la scène de la figure 5.10, la différence moyenne entre une image reconstruite avec et sans procédé de quantification est de 0,008% en espace *Lab*. Naturellement, l'étape de déquantification doit être effectuée côté client afin de pouvoir effectuer l'étape du *push-pull*. Comme montré sur

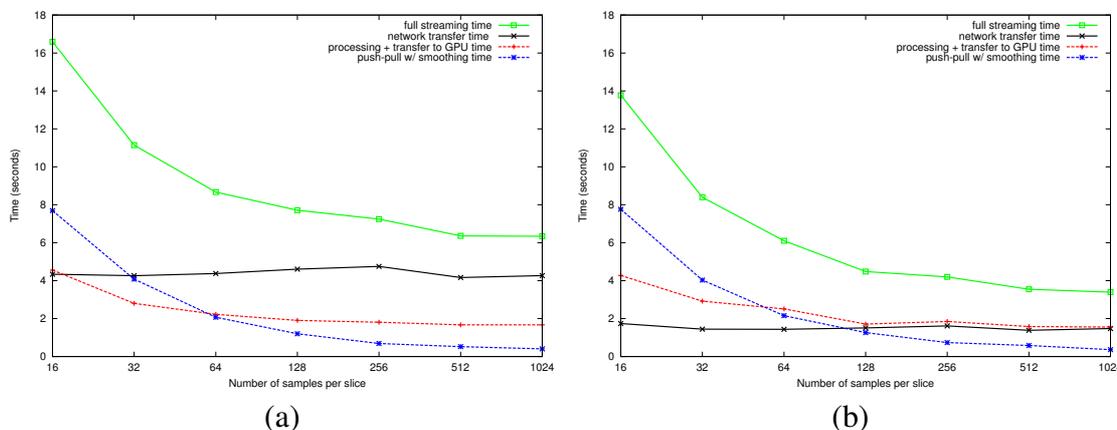


Figure 5.12 – Temps total de téléchargement pour une grille de $32 \times 32 \times 32$ avec différentes tailles de buffer. La taille de la grille 3D est de $3 Mo$ en utilisant (a) des nombres flottants et (b) la taille quantifiée est de $1,3 Mo$. Le temps de traitement indiqué inclut l'étape de déquantification et la copie vers la mémoire CPU.

le graphique de la figure 5.12, cette étape introduit un faible ralentissement.

Nos grilles permettent de choisir n'importe quel type de représentation progressive pour la géométrie. Nous choisissons d'utiliser une représentation similaire à celle proposée par Melax [Mel98]. Cette représentation permet de transmettre une première version grossière du maillage qui se raffinerà à l'aide d'une opération de *vertex split* [Hop96]. Pour une transmission progressive du maillage multi-résolution, nous réordonnons les sommets de manière à ce que les sommets les plus significatifs (présents dans la version grossière du maillage) soient envoyés en premier. La transmission consiste à envoyer des séquences de sommets et de faces. Ce procédé, similaire à celui de Guézic *et al.* [GTHL99], offre $n - 2$ niveaux de détails où n représente le nombre total de sommets.

Nous avons testé notre système de visualisation avec un serveur dont le processeur est un Intel Q6600 (4 GB de RAM) et un client dont le processeur est un Intel Pentium M 2.26 GHz (2 GB de RAM). Nous avons mesuré toutes les transmissions réseau sur un réseau WiFi 802.11g. Chaque mesure a été répétée plusieurs fois et moyennée.

Pour démontrer l'indépendance de la géométrie et de l'éclairage, nous avons testé notre système avec deux stratégies extrêmes de transmission. Dans la première (cf. figure 5.14), nous transférons d'abord une géométrie basse résolution avec une grille

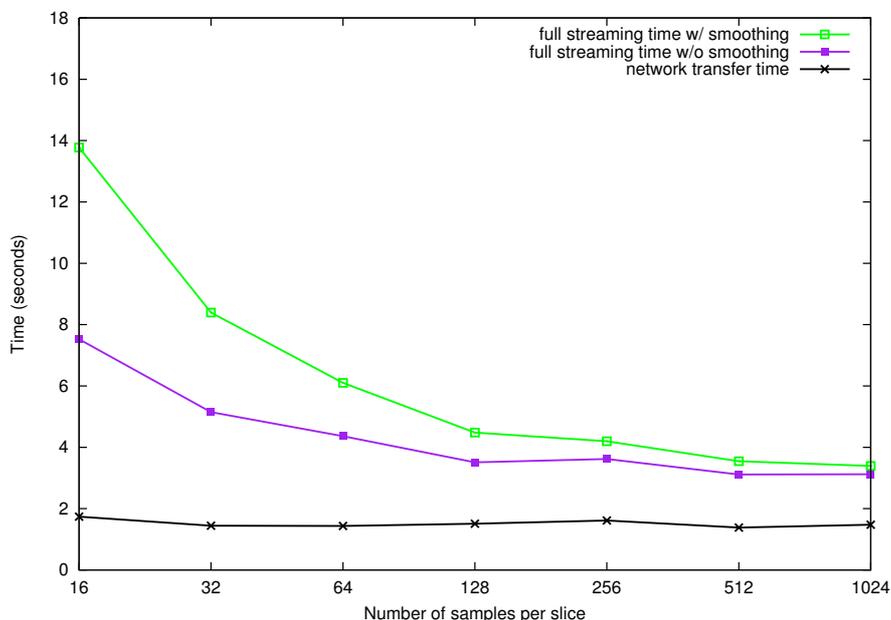


Figure 5.13 – Comparaison des temps de téléchargement avec et sans filtrage lors de l'étape du push-pull pour une grille quantifiée de $32 \times 32 \times 32$. L'étape de filtrage représente 80% du temps passé lors de l'étape de *push-pull*.

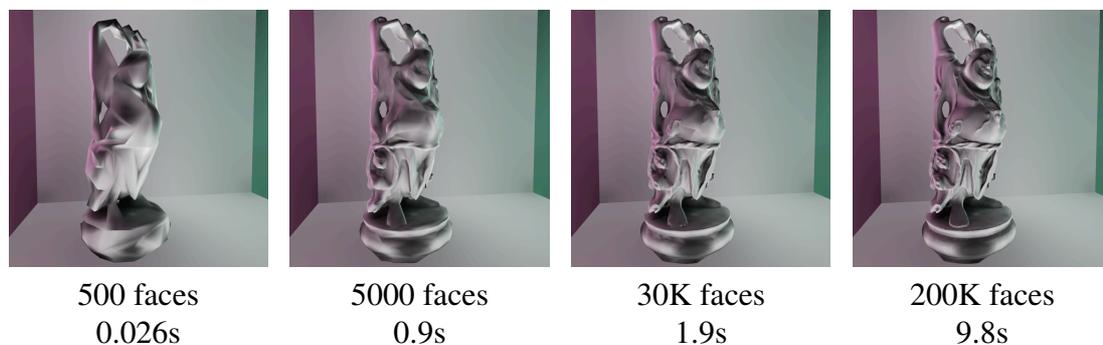


Figure 5.14 – Transfert progressif de la géométrie avec une grille d'éclairage fixe ($16 \times 16 \times 16$ (563 Ko)). De gauche à droite : l'éclairage indirect s'adapte lorsque la géométrie est raffinée (utilisation d'un buffer de 250 sommets). Pour chaque image, le temps indiqué en-dessous représente le temps total requis pour atteindre la taille indiquée de géométrie dans l'image à sa gauche. Les faces orientées vers le mur rouge sont toujours rouges et les effets de couleur de l'éclairage indirect sont représentés de manière cohérente.

d'éclairage complète. Ensuite, nous transmettons progressivement le reste de la géométrie sous forme de détails. Grâce à notre représentation vectorielle, l'éclairage

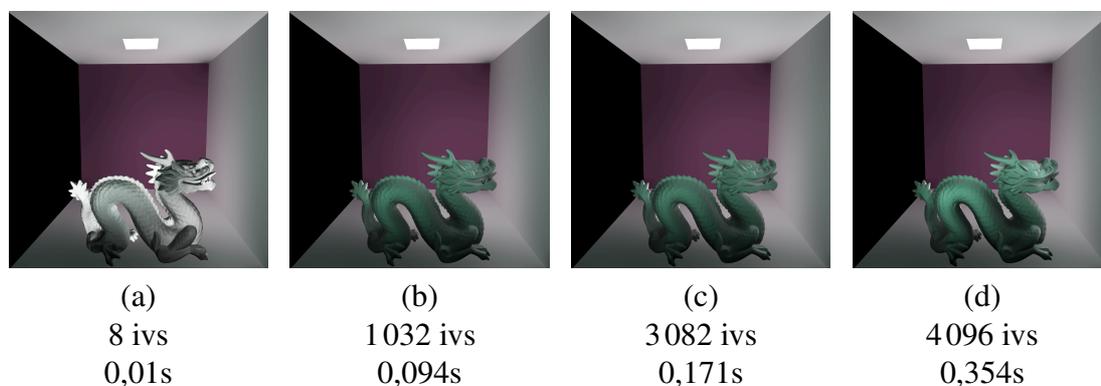


Figure 5.15 – Transfert progressif d’une grille d’éclairage de $16 \times 16 \times 16$ (déquantifiée) pour une géométrie fixe. (a) La grille d’éclairage initiale, avec seulement les 8 vecteurs d’irradiance (iv) localisés dans les coins, est raffinée (b)-(d) avec 64 échantillons par couche et par requête. Pour chaque image, les temps indiqués en-dessous représentent le temps total pour atteindre la taille de grille atteinte dans l’image à sa gauche.

s’adapte de manière lisse aux variations locales de la géométrie pendant qu’elle se raffine sans avoir à utiliser des informations complémentaires.

Dans la seconde stratégie (cf. figure 5.15), nous transmettons progressivement une grille de vecteurs d’irradiance sur une géométrie qui est déjà à pleine résolution. Cette stratégie est intéressante lorsque le serveur raffine l’éclairage dans des environnements dynamiques qui utilisent une solution incrémentale pour le calcul de l’éclairage global (e.g. [DBMS02]). Ceci est aussi utile si l’on souhaite avoir un serveur qui calcule à distance l’éclairage et transmet progressivement les vecteurs d’irradiance pour différentes grilles calculées en parallèle.

Un avantage de notre séparation entre la structure d’éclairage et celle pour la géométrie est que le client contrôle le raffinement des données en fonction de ses capacités matérielles et celles du réseau. La stratégie classique qui offre la visualisation progressive la plus lisse et continue est d’entrelacer la transmission progressive de géométrie et d’éclairage (cf. figure 5.10). Nos tests montrent que, côté client, le taux de rafraîchissement de l’image reste constant lorsque l’on met à jour la géométrie ou l’éclairage. Ainsi, notre système fournit un rendu temps réel avec des mises à jour continues au client. De plus, pour une scène donnée, nous n’avons pas mesuré des pertes de perfor-

mance lorsque l'on introduit les grilles d'éclairage.

Nous avons aussi testé séparément les temps de *streaming* pour la géométrie et la grille d'éclairage. Afin de tester l'influence du *streaming* sur le temps de transmission, nous avons comparé le temps requis pour télécharger la structure complète avec celui requis pour transférer la géométrie/grille sans la pénalité du *streaming* (cf. la courbe horizontale cyan de la figure 5.16). Pour la géométrie, illustrée à la figure 5.16, la pénalité maximale est de 70% mais seulement en utilisant des paquets de taille faible. Augmenter la taille des paquets réduit rapidement cette pénalité à 10%. C'est d'ailleurs un très bon compromis pour une solution de *streaming* comparée à l'approche classique utilisant un mécanisme de LODs avec VRML où la pénalité peut atteindre 100%. Cette pénalité provient principalement du traitement progressif pour la transmission de la géométrie et le temps de transférer les données au GPU.

Pour l'éclairage, nous avons testé la transmission de grilles quantifiées et non-quantifiées utilisant un encodage en nombres flottants. Comme attendu, le temps de transfert sur le réseau est réduit quand on utilise des grilles quantifiées (cf. la courbe noire sur les deux graphiques de la figure 5.12). De plus, la comparaison des deux courbes rouges de la même figure démontre que l'étape de déquantification des données est rapide (approximativement 4%). Le temps passé pour effectuer le *push-pull* dépend du nombre de requêtes effectuées par le client. Ainsi, la taille des paquets doit être choisie en accord avec les capacités du serveur et des clients et celui du réseau. D'après la figure 5.12, lorsqu'on utilise une grille $32 \times 32 \times 32$ quantifiée, la taille appropriée de paquets est atteinte en utilisant environ 100 échantillons par couche (l'abscisse correspondant à l'endroit où la courbe noire croise la courbe bleue). Finalement, comme illustré à la figure 5.13, la majorité de la pénalité introduite par le *streaming* provient de la passe du *push-pull* de la grille 3D.

Système de cache pour rendu CPU

Nos grilles d'éclairage avec vecteurs d'irradiance peuvent aussi être utilisées dans le contexte de moteur de rendu d'éclairage global en tant que mécanisme de cache. En effet, les reconstructions directionnelles et spatiales de la structure peuvent être uti-

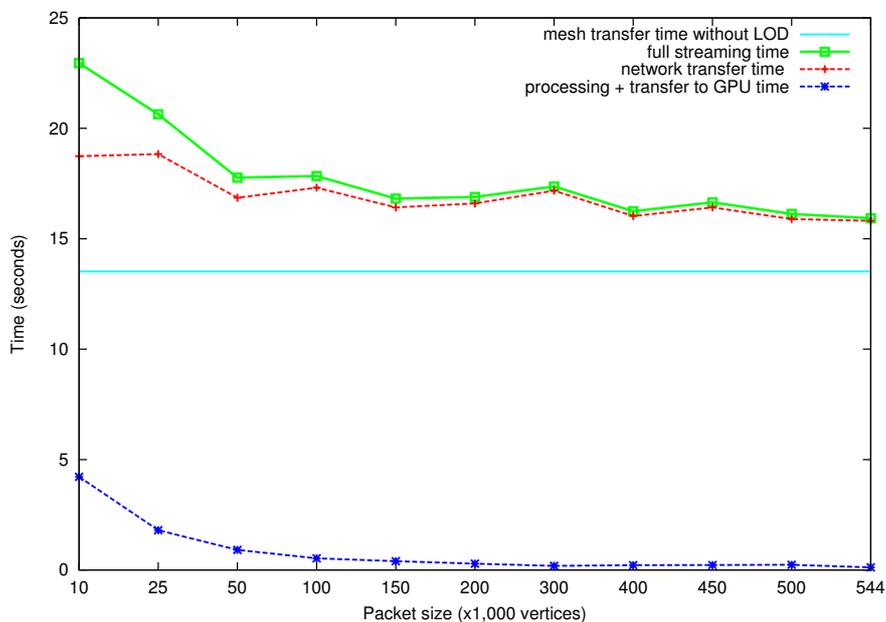


Figure 5.16 – Temps de téléchargement pour la géométrie complète du bouddha de la figure 5.10 pour différentes tailles de paquets (un paquet correspond à un nombre fixe de sommets). La taille de la géométrie complète (haute résolution) est de 1,1 M polygones. Le temps de traitement inclut la mise-à-jour des indices de sommets à chaque étape de la transmission.

lisées pour reconstruire l'éclairage indirect incident pour une géométrie diffuse, voire glossy. La continuité de la reconstruction de l'éclairage permet d'utiliser la structure avec un accès direct et sans passe de *Final Gathering* comme cela est fait pour le *Photon Map*. Les techniques de Ward *et al.* [WRC88, WH92] et Krivanek *et al.* [KBPZ06] peuvent également être utilisées de manière directe mais ne garantissent pas la même continuité que notre structure, et restent dépendantes de la résolution de la géométrie.

Pour évaluer notre approche, nous utilisons deux scènes types avec des configurations d'éclairage complexes. La première configuration (cf. figure 5.17) représente une scène principalement éclairée de manière directe par 11 sources surfaciques. La seconde (cf. figure 5.18) est un cas classique d'éclairage indirect dans lequel une pièce est éclairée de manière indirecte par une lumière positionnée dans une pièce voisine. Les deux scènes totalisent plus de 8 millions de polygones consécutifs à la présence d'objets hautement détaillés utilisés pour illustrer la robustesse géométrique de notre approche.



Figure 5.17 – Scène majoritairement éclairée par un éclairage direct. 16 rayons par pixel sont utilisés. (gauche) Notre technique avec une grille d'éclairage de $40 \times 50 \times 40$ (24.5 Mo), sans compression des vecteurs d'irradiance, utilisée de manière directe avec un schéma d'interpolation tri-cubique. (centre) *Photon Mapping* avec le cache de Christensen utilisant 50 des 5 millions de photons (124 Mo) stockés pour pré-calculer les échantillons d'irradiance. (droite) Solution de référence obtenue en effectuant un *Path Tracing* avec 1 600 rayons par pixel. Les effets d'éclairage sont similaires entre les différentes images mais notre technique est bien plus rapide pour le calcul de l'éclairage indirect (51 s vs. 5 527 s) et réduit nettement le temps total de rendu (1 341 s vs. 6 618 s).

Nous comparons notre technique à celle du *Photon Mapping* combinée à la technique de pré-calcul de Christensen [Chr99]. Le temps de pré-calcul pour notre technique englobe le temps de propagation des photons et leur accumulation dans la grille d'éclairage tandis que pour celle de Christensen, il englobe le temps de propagation des photons, la construction du kd-tree et le pré-calcul des échantillons d'irradiance. Pour toutes les images générées avec la méthode de Christensen, nous avons fixé le nombre d'échantillons d'irradiance pré-calculés de manière à ce qu'il corresponde au quart du nombre total de photons tirés tel que suggéré par l'auteur.

Les résultats des figures 5.17 et 5.18 ont tous été calculés sur un processeur AMD 64 bits 3500+ avec 2 Go de mémoire. Pour ces mêmes figures, la résolution des images est de 640×480 pixels.

La figure 5.17 compare les résultats obtenus avec notre technique et celle de Christensen pour la configuration d'éclairage direct. Afin d'avoir un point de comparaison, nous avons aussi calculé, à l'aide d'un *Path Tracing*, une solution de référence. Pour un

temps de pré-calcul identique, notre technique permet d'effectuer le rendu de l'éclairage indirect nettement plus rapidement (51 s vs 5527 s) grâce à un accès direct de la structure. Dans notre technique, lors de l'étape finale de rendu, le CPU passe principalement son temps à calculer l'éclairage direct alors que la technique de Christensen doit effectuer un accès indirect à sa structure. Sans cet accès indirect, l'utilisation des échantillons d'irradiance pré-calculés produit de sérieux artefacts : zones d'éclairage constant similaires à un diagramme de Voronoi. Ces artefacts sont connus et discutés par Christensen [Chr99].

Pour la scène à éclairage indirect (cf. figure 5.18), nous avons aussi essayé d'utiliser un accès direct à notre structure mais pour capturer les zones étroites d'ombre générées par les colonnes, une grille de plus haute résolution ($30 \times 20 \times 92$) a été utilisée. Dès lors, un nombre plus conséquent de photons doit être utilisé. Ceci accroît de manière significative le temps de pré-calcul de notre structure mais notre méthode avec un accès direct reste toujours plus rapide (2311 s vs 47971 s) que celle de Christensen pour une qualité similaire. De manière alternative, nous avons aussi testé notre approche avec un accès indirect aux vecteurs d'irradiance (cf. figure 5.18) en utilisant une résolution de grille plus petite ($12 \times 8 \times 20$) construite avec le même nombre de photons que celui utilisé pour la technique de Christensen. Notre technique réduit le temps de pré-calcul (150 s vs 466 s) et le temps de reconstruction de l'éclairage indirect (2376 s vs 10050 s) tout en retenant les mêmes caractéristiques d'éclairage. L'accès à notre structure est constant par rapport au nombre d'échantillons contenus dans le cache alors qu'il est logarithmique pour le kd-tree utilisé par Christensen ; et ceci explique donc notre gain de performances.

5.4 Optimisations

Dans cette section nous présentons des optimisations au niveau du calcul de la grille et de la place mémoire occupée. Ces optimisations constituent des travaux futurs à court terme.

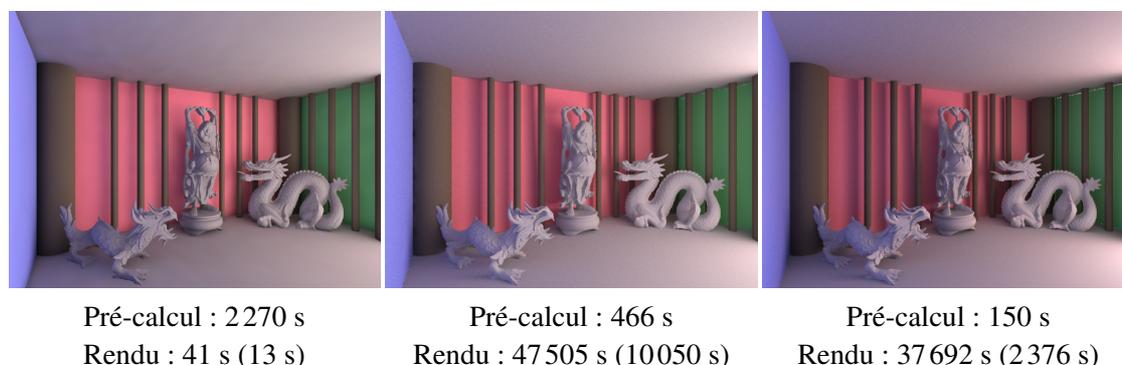


Figure 5.18 – Scène avec principalement un éclairage indirect. (gauche) Notre technique en utilisant une grille de $30 \times 20 \times 92$ (11,4 Mo), où les vecteurs d’irradiance ne sont pas compressés, construite en utilisant 80 millions de photons. (centre) Image de référence avec la méthode de Christensen en utilisant 3 200 rayons pour échantillonner l’hémisphère et 5 millions de photons. (droite) Notre technique en utilisant une grille de $12 \times 8 \times 20$ accédée de manière indirecte où le même nombre de photons et le même nombre de rayons ont été utilisés que pour l’image du centre.

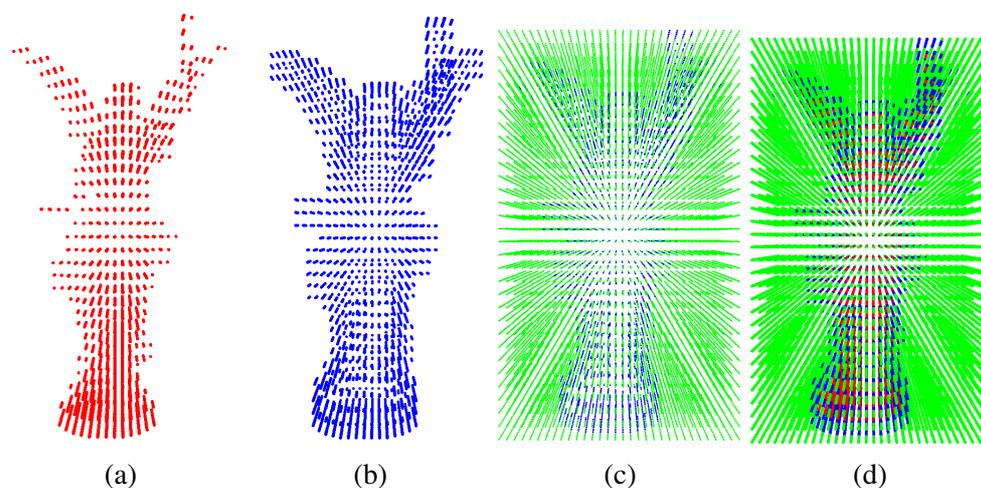


Figure 5.19 – Illustration sur l’objet Lucy des sommets réellement utilisés pour l’éclairage de l’objet. (a) Sommets à l’intérieur de la grille. (b) Sommets à l’extérieur de l’objet et utilisés pour le calcul d’éclairage, *i.e.*, tangents à la surface extérieure de l’objet. (c) Sommets tangents et ceux extérieurs non utilisés lors du calcul de l’éclairage. (d) Ensemble des sommets de la grille montrant le faible ratio (12,032%) de sommets réellement utiles par rapport au nombre total de 32 768 pour une grille de $32 \times 32 \times 32$.

Optimisation spatiale

Nous avons choisi d’utiliser une grille régulière comme structure spatiale. Une grille

régulière possède de nombreux avantages comparée à des structures adaptatives comme le kd-tree ou l'octree : principalement celui d'être facilement utilisable sur GPU. En revanche, lorsqu'on analyse le nombre de sommets de la grille réellement utilisés pour calculer l'éclairage indirect sur la géométrie détaillée, nous nous apercevons (cf. figure 5.19) que seul un tiers environ des sommets est utile. Ce ratio peut être encore plus faible à mesure que l'on augmente la résolution de la grille. Pour l'exemple de la figure 5.19, on atteint même 12% de sommets utiles (tangents) avec 9% de sommets intérieurs, le reste (79%) étant des sommets extérieurs non utiles. La structure apparaît donc comme nettement creuse et il serait donc sûrement intéressant d'utiliser des structures de type table de hachage. La question plus délicate est de réussir à utiliser une table de hachage sur le GPU (e.g. [LH06]).

Optimisation du pré-calcul

Nos différentes expérimentations montrent que le pré-calcul de la grille constitue un goulot d'étranglement pour de futures applications interactives comme l'édition ou la pré-visualisation rapide de simulation d'éclairage global.

Une première manière d'accélérer le calcul qui utilise l'échantillonnage stochastique à partir des sommets de la grille est de factoriser le coût des chemins lumineux. En effet, pour deux sommets voisins, de nombreux chemins lumineux peuvent être réutilisés. La formulation mathématique du calcul des vecteurs d'irradiance devient alors :

$$\mathbf{L}_{ij} = \frac{1}{p(x_j)} L(x_j \leftarrow \boldsymbol{\omega}_j) \mathbf{D}_{ij} V(s_i, x_j) G(s_i, \mathbf{D}_{ij}, \mathbf{n}_j) f_r(x_j, \boldsymbol{\omega}_j, -\mathbf{D}_{ij})$$

où :

- s_i représente un sommet de la grille.
- x_j un point de la scène, tiré aléatoirement visible depuis s_i et dont la normale est \mathbf{n}_j .
- \mathbf{L}_{ij} la radiance vectorielle accumulée au sommet s_i de la grille en provenance de l'échantillon x_j .
- $\boldsymbol{\omega}_j$ direction aléatoire pour laquelle est évaluée la radiance incidente en x_j

- \mathbf{D}_{ij} la direction formée par les points (s_i, x_j) .
- V la visibilité point à point.
- G le facteur géométrique avec

$$G(s_i, \mathbf{D}_{ij}, \mathbf{n}_j) = \frac{\max(-\mathbf{D}_{ij} \cdot \mathbf{n}_j, 0) \max(\boldsymbol{\omega}_j \cdot \mathbf{n}_j, 0)}{\|s_i - x_j\|^2}.$$

- $f_r(x_j, \boldsymbol{\omega}_j, -\mathbf{D}_{ij})$ la BRDF au point x_j évaluée pour les directions $\boldsymbol{\omega}_j$ et $-\mathbf{D}_{ij}$.
- $p(x_j)$ la probabilité de choisir le point x_j donnée par :

$$p(x_j) = \frac{1}{4\pi N} \sum_{k=1}^N V(s_k, x_j) \frac{\max(-\mathbf{D}_{kj}, \mathbf{n}_j, 0)}{\|s_k - x_j\|^2}$$

où N représente le nombre de sommets contenus dans la grille.

Cette formulation met clairement en évidence que l'échantillon x_j et sa radiance $L(x_j \leftarrow \boldsymbol{\omega}_j)$ vont pouvoir être utilisés et réutilisés pour tous les sommets s_i de la grille que l'on cherche à pré-calculer, résultant très probablement dans une accélération du temps de pré-calcul.

Une autre manière d'accélérer ce calcul serait d'utiliser le GPU pour construire la grille. En prenant une approche où l'on part de la lumière et l'on accumule l'énergie lumineuse dans la grille, le calcul pourrait être factorisé et accéléré par le GPU. Enfin, la combinaison du tracé de rayons et du tracé de photons (ou du *Bidirectional Path Tracing*) reste la manière la plus robuste pour pré-calculer des grilles dans les conditions d'éclairage les plus complexes (*i.e.*, avec de multiples indirections) et devraient donc constituer la meilleure stratégie pour le pré-calcul des grilles.

5.5 Travaux futurs

Dans ce chapitre, nous avons introduit de nouvelles structures 2D et 3D pour représenter l'éclairage indirect. Ces structures sont des grilles régulières dont les sommets stockent six vecteurs d'irradiance capables de reconstruire l'irradiance pour n'importe quelle normale sous-jacente. Ainsi, les structures proposées offrent une grande robustesse par rapport à la géométrie de la scène et nous avons présenté plusieurs types d'ap-

plications possibles.

L'une des limitations actuelles de notre approche est liée à la représentation de l'éclairage incident. En effet, nos six vecteurs d'irradiance sont peu coûteux en espace mémoire mais restent insuffisants pour des matériaux qui sont glossy, voire spéculaires. Une piste de recherche est donc de développer des nouvelles représentations pour l'éclairage incident. L'annexe III esquisse plusieurs pistes de ce que pourraient être de telles représentations.

Nous aimerions aussi développer une structure adaptative qui pourrait se simplifier (resp. se raffiner) si la variation de l'éclairage incident est trop faible (resp. forte). Ceci faciliterait l'utilisation des structures puisque la résolution optimale serait automatiquement inférée en fonction de l'éclairage et non plus choisie a priori. La principale difficulté avec de telles structures est alors de garantir une reconstruction spatiale continue lorsque la géométrie est à cheval sur différents voxels dont les profondeurs pourraient être très variées. Une autre piste serait aussi de stocker en plus d'une représentation directionnelle une représentation du gradient de l'éclairage afin d'utiliser des schémas d'interpolation plus continus tels que celui d'Hermite.

La question d'utiliser une structure avec une représentation vectorielle afin d'éditer l'éclairage, et plus particulièrement le flux, est également une piste de recherche qui nous apparaît comme valable. Dans un contexte d'édition, comme pour les BRDFs, c'est la sémantique des actions de l'artiste/utilisateur qui doit guider la ligne de recherche. A savoir, une sémantique simple et puissante telle que «plus de rouge ici» semble être une voie prometteuse. Cependant, il serait bon de vérifier qu'il y a un réel besoin de la part des artistes. Naturellement, développer un nouvel outil d'édition peut créer le besoin. La difficulté avec une édition de l'éclairage indirect reste à notre sens de déterminer si l'édition d'autres caractéristiques/variables de la scène ne permettrait pas d'obtenir les mêmes résultats. Par exemple, l'édition d'une SVBRDF permet de localiser précisément des effets de lumière ou encore l'utilisation de lumières virtuelles associées uniquement à certains objets de la scène sont des techniques simples et déjà utilisées. En somme, il convient de déterminer quel est le but réellement recherché par une édition de l'éclaire-

ment indirect, phénomène global et très complexe a priori à manipuler par un utilisateur novice.

CHAPITRE 6

MANIPULER L'ÉCLAIREMENT INCIDENT

Dans ce chapitre, nous nous intéressons à la possibilité de manipuler l'éclairage incident, et ce, dans le but de contrôler la manière dont la forme des objets ressort dans une image. Ce but diffère des méthodes d'édition précédentes qui tentent de retrouver les propriétés d'éclairage ou matérielles d'une scène. Notre objectif est d'améliorer la perception de la forme sans avoir à en changer les caractéristiques matérielles.

Le travail présenté dans ce chapitre est le fruit d'une collaboration avec Romain Vergne et Pascal Barla de l'équipe INRIA Iparla. Une partie des résultats a été validée dans une publication [VPB⁺09] et le reste est en cours de soumission.

6.1 Travaux antérieurs

Pour la mise en avant de la forme, les travaux précédents ne se sont principalement intéressés qu'aux dessins à base de traits. Depuis les premiers travaux en infographie de Saito et Takahashi [ST90b], plusieurs améliorations ont été proposées, telles celle de Nienhaus et Döllner [ND04] qui travaillent sur les silhouettes et les plis ou celle de DeCarlo *et al.* [DFRS03] introduisant les contours suggestifs. Malgré plusieurs travaux récents [KST08, ZHXC09], aucun consensus (cf. [CGL⁺08]) n'a été trouvé sur la «bonne» définition de l'ensemble des lignes qui doivent être dessinées afin d'améliorer la perception de la forme de l'objet. De plus, à l'exception des techniques proposées par Lee *et al.* [LMLH07] et Goodwin *et al.* [GVH07], les méthodes à base de traits effacent les caractéristiques de l'éclairage et des matériaux. Les transformations présentées dans ce chapitre veulent précisément conserver ces caractéristiques tout en mettant en avant la forme des objets.

Une autre manière d'améliorer la perception de la forme est d'utiliser l'occultation ambiante ou *Ambient Occlusion* [PG04]. Cette méthode tend à assombrir les régions de l'objet qui sont les moins visibles comme les concavités et est à rapprocher de la

notion d'accessibilité telle qu'introduite par Miller [Mil94]. L'*Ambient Occlusion* améliore principalement la perception des volumes, mais est difficilement contrôlable par un utilisateur. De plus, elle ignore les détails géométriques présents sur la surface de l'objet.

La technique (*3D unsharp masking*) proposée par Ritschel *et al.* [RSI⁺08] possède les mêmes avantages et inconvénients que ceux cités précédemment. Elle augmente le contraste de la radiance réfléchie sans tenir compte des variations de géométrie, matériaux ou d'éclairage changeant ainsi fortement la perception des matériaux brillants.

Les techniques les plus proches des transformations que nous proposons sont celles qui altèrent le *shading* en utilisant une information locale de la surface. Elles sont souvent inspirées de techniques d'illustrations stylisées venant de domaines spécifiques tels que celui de l'illustration mécanique [GGSC98], l'archéologie [KWTM03], le dessin technique au crayon [CST05], la cartographie [RBD06] ou encore l'anatomie [LHV06]. Toutes ces techniques sont contraintes à une combinaison fixe de style, d'éclairage et de matériau limitant ainsi leur cadre d'utilisation.

A contrario, nous proposons de modifier l'éclairage incident en minimisant le changement perceptuel du matériau et ce, pour les différentes sortes d'éclairage et sans restreindre le type de rendu. Nous proposons deux formes de modifications de l'éclairage incident. La première change la direction d'incidence de la lumière alors que la seconde change son intensité. Une modification de la direction correspond à une modification fréquentielle du signal incident tandis qu'une modification de l'intensité correspond à une modification de l'amplitude du signal incident.

6.2 Descripteur géométrique

Afin de mettre en avant les variations de la forme de l'objet, les deux méthodes s'appuient sur un descripteur géométrique. Le descripteur géométrique est extrait à partir des images de normales et des profondeurs rendues depuis le point de vue. C'est un tenseur de courbure H défini pour chaque point p de la surface, visible depuis le point

de vue et à l'extérieur des silhouettes et des plis de l'objet :

$$H(p) = \nabla^T \mathbf{g}(p)$$

où $\mathbf{g}(p) = (-n_x/n_z, -n_y/n_z)^T$ est le gradient de la profondeur calculé à partir des normales $\mathbf{n} = (n_x, n_y, n_z)^T$ repérées en espace image. Les caractéristiques saillantes de la surface telles que les courbures principales κ_u et κ_v sont extraites à partir de H . L'intérêt d'un tenseur de courbure défini en espace image est qu'il peut être utilisé directement et sans pré-calcul. On peut également l'utiliser avec des géométries sur lesquelles des cartes de normales ont été plaquées. Enfin, il est facile d'en faire une version multi-résolution. Pour des raisons de performance, ce descripteur est calculé sur GPU à partir de buffers géométriques (*G-Buffer*) générés lors d'une première passe de rendu. Plus de détails sur le descripteur peuvent être consultés dans l'article associé [VPB⁺09]. La perturbation introduite sur l'éclairage incident sera proportionnelle à l'intensité de courbure du descripteur.

6.3 Modification des directions de l'éclairage incident

Intuition

L'idée de modifier la direction de l'éclairage incident afin de mettre plus en avant les détails vient des travaux de Fleming *et al.* [FTA04, FTA09]. Ces travaux de vision par ordinateur, effectués sur des objets parfaitement miroirs, montrent que la compression des motifs lumineux réfléchis par la surface de l'objet influence notre perception de sa courbure. Dans la figure 6.1(a), les parties courbes de la surface compressent une plus grande partie de l'environnement lumineux que les surfaces planes, produisant ainsi des motifs plus compressés depuis un point de vue particulier. L'idée (cf. figure 6.1(b)) est de perturber ces motifs lumineux en modifiant les directions incidentes de la lumière de telle sorte que la compression des motifs lumineux réfléchis par la surface améliore notre perception de la courbure de l'objet.

Nous introduisons le terme de *Light Warping* (LW) qui regroupe l'ensemble des modifications effectuées à l'éclairage incident. Pour construire une fonction qui modifie

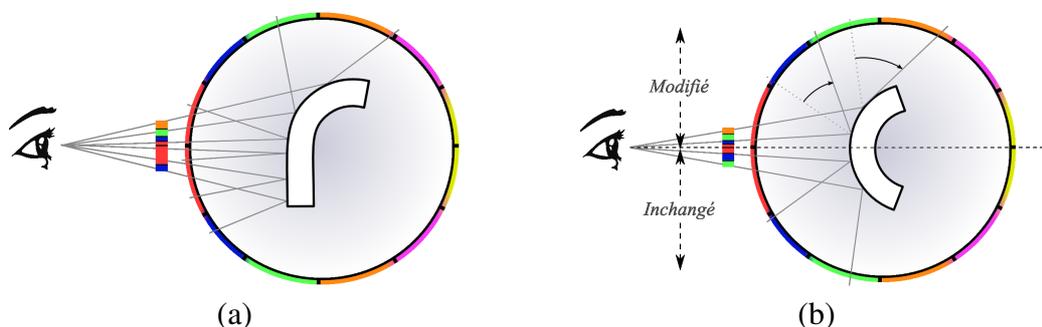


Figure 6.1 – (a) Les compressions des motifs lumineux réfléchis par un objet parfaitement miroir révèlent des informations sur la courbure de l’objet pour un point de vue donné. Les surfaces planes réfléchissent de plus petites régions que les surfaces courbes. Pour la même aire, une surface courbe comprime plus les motifs lumineux qu’une surface plane. (b) Afin d’améliorer l’information sur la courbure provenant de la réflexion des motifs lumineux réfléchis par la surface, nous agrandissons la région réfléchie par la surface en modifiant la direction incidente de la lumière.

l’éclairage incident, nous utilisons le descripteur géométrique qui nous permet d’identifier les caractéristiques principales de l’objet et à plusieurs niveaux d’échelle. Une fois l’éclairage incident modifié, nous évaluons simplement l’équation du rendu afin d’obtenir l’image finale.

Reformulation de l’équation de réflexion

Formellement, le *Light Warping* est basé sur une modification de la direction de la radiance incidente :

$$L(p \rightarrow \omega_o) = \int_S f_r(p, \omega_o, \mathbf{l}) (\mathbf{n} \cdot \mathbf{l}) L(p \leftarrow W_p(\mathbf{l})) d\mathbf{l} \quad (6.1)$$

où S est la sphère des directions, \mathbf{l} la direction incidente de la lumière et $W : S \rightarrow S$ est la fonction de transformation des directions qui est dépendante du point p et de la direction \mathbf{l} . Cette équation peut être directement utilisée dans un moteur d’éclairage global. En revanche, dans le cadre d’un rendu sur carte graphique il n’est pas performant de l’utiliser directement. Pour pallier ce problème, nous introduisons l’équation inverse :

$$L(p \rightarrow \omega_o) = \int_S f_r(p, \omega_o, W^{-1}(\mathbf{l}')) (\mathbf{n} \cdot W^{-1}(\mathbf{l}')) L(p \leftarrow \mathbf{l}') J d\mathbf{l}' . \quad (6.2)$$

Dans cette dernière équation, on suppose que les directions des lumières ont déjà été modifiées. Le jacobien J est introduit pour compenser la perte ou le gain d'énergie dû à la modification des directions. Cette équation permet d'implémenter de manière efficace le *Light Warping* sur une carte graphique en utilisant un pré-échantillonnage fixe des cartes d'environnement. Bien évidemment, le jacobien est directement relié à la fonction W qui sera présentée au prochain paragraphe.

Formulation de W

Afin de pouvoir utiliser l'équation inverse, W doit être une transformation bijective dans l'espace des directions. De plus, comme la compression des motifs lumineux réfléchis suit l'anisotropie de courbure [FTA04] (définie comme le ratio des courbures principales), nous voulons que la transformation de l'éclairement incident suive les directions des courbures principales en \mathbf{u} et \mathbf{v} fournies par le descripteur. Comme ce descripteur fournit une information symétrique en espace image, la fonction W doit être symétrique par rapport à \mathbf{u} et \mathbf{v} et invariante selon la direction de vue que nous noterons \mathbf{z} . Chaque direction \mathbf{l} de lumière doit donc d'abord être projetée dans le référentiel $\{\mathbf{u}, \mathbf{v}, \mathbf{z}\}$ avant d'être modifiée par la fonction W pour être à nouveau re-transformée dans son espace d'origine.

La direction de la lumière et les courbures ne sont pas exprimées dans le même référentiel. Afin d'établir facilement une correspondance entre l'espace cartésien et l'espace angulaire, nous utilisons une projection stéréographique sur le plan image. Le procédé est illustré dans la figure 6.2 :

- (1) la direction \mathbf{l} de la lumière est projetée stéréographiquement sur le plan $\mathbf{z} = 1$:
 $\bar{\mathbf{l}} = \mathcal{P}(\mathbf{l})$ où \mathcal{P} est l'opérateur de projection stéréographique ;
- (2) $\bar{\mathbf{l}}$ est modifiée en fonction des informations de courbures : $\bar{\mathbf{l}}' = W(\bar{\mathbf{l}})$;
- (3) $\bar{\mathbf{l}}'$ est re-transformée dans l'espace des directions : $\mathbf{l}' = \mathcal{P}^{-1}(\bar{\mathbf{l}}')$.

Soit $\mathbf{l} = (l_u, l_v, l_z)$ une direction incidente de lumière exprimée dans le référentiel $\{\mathbf{u}, \mathbf{v}, \mathbf{z}\}$, la projection stéréographique de \mathbf{l} par \mathcal{P} s'exprime par :

$$\mathcal{P}(\mathbf{l}) = (a, b, c) = \left(\frac{2l_u}{l_z + 1}, \frac{2l_v}{l_z + 1}, 1 \right).$$

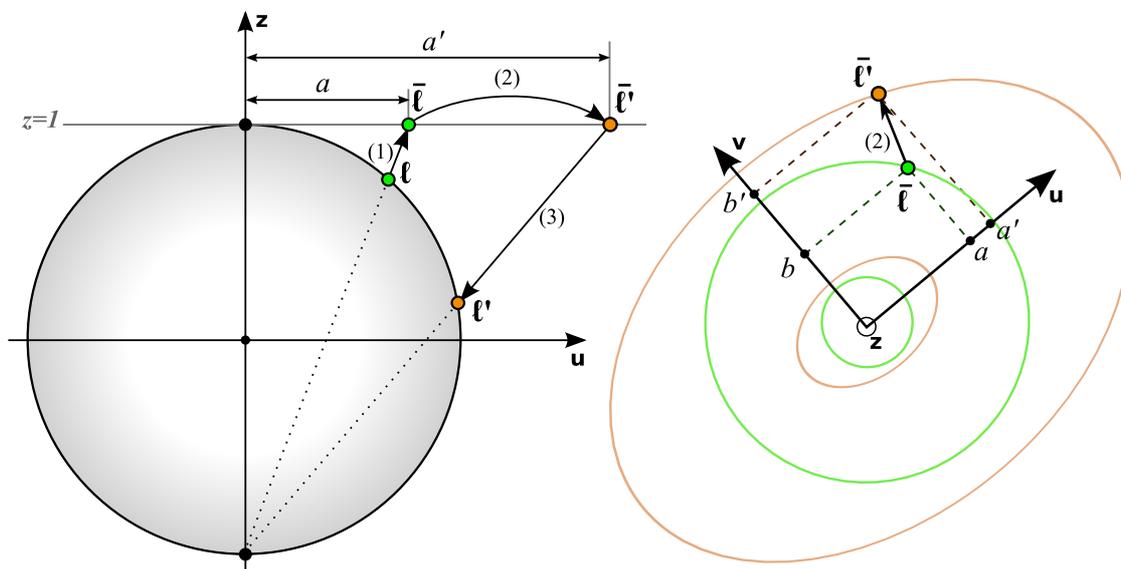


Figure 6.2 – (gauche) Illustration 1D du processus de modification des directions incidentes. La direction \mathbf{l} de la lumière est (1) projetée dans l'espace stéréographique, (2) modifiée dans cet espace et (3) reprojétée sur la sphère des directions. (droite) Illustration de l'étape (2) en 2D. Notez la symétrie autour des directions \mathbf{u} et \mathbf{v} .

Intuitivement, la direction $(0,0,1)$ est projetée à l'origine du plan stéréographique. Toutes les autres directions sont projetées plus loin au fur et à mesure qu'elles se rapprochent de la direction $(0,0,-1)$, qui est projetée à l'infini.

W est simplement définie comme une mise à l'échelle non-linéaire dépendant des courbures κ_u et κ_v :

$$W(\bar{\mathbf{l}}) = (a', b', c') = (\lambda_u a, \lambda_v b, 1)$$

où les facteurs λ_u et λ_v sont respectivement fonctions de κ_u et κ_v :

$$\lambda_{u|v} = \tan\left(\arctan(\alpha\kappa_{u|v})/6 + \frac{\pi}{4}\right). \quad (6.3)$$

Cette formulation garantit qu'au maximum la moitié de l'énergie lumineuse incidente d'un hémisphère est envoyée dans l'hémisphère opposé. Le paramètre α est contrôlé par l'utilisateur et définit à quel point les directions doivent être modifiées en fonction de la courbure.

La transformation stéréographique inverse \mathcal{P}^{-1} est définie par :

$$\mathcal{P}^{-1} = (l'_u, l'_v, l'_z) = (a't, b't, 2t - 1)$$

où t est donné par :

$$t = \frac{4}{(4 + a'^2 + b'^2)}.$$

t représente la distance paramétrique sur la droite (\mathbf{l}') lorsque cette dernière intersecte la sphère des directions.

Naturellement, on peut concaténer toutes ces transformations en une seule afin d'obtenir directement W :

$$\begin{aligned} W(\mathbf{l}) &= \mathcal{P}^{-1}(\mathbf{l}) \circ W(\mathbf{l}) \circ \mathcal{P}(\mathbf{l}) \\ &\Leftrightarrow \\ W(\mathbf{l}) &= \left(\frac{2t\lambda_u l_u}{1+l_z}, \frac{2t\lambda_v l_v}{1+l_z}, 2t-1 \right) \quad \text{avec} \quad t = \frac{(1+l_z)^2}{(1+l_z)^2 + \lambda_u^2 l_u^2 + \lambda_v^2 l_v^2}. \end{aligned}$$

La fonction inverse de W^{-1} est définie en prenant l'inverse de λ_u et λ_v , ce qui revient à changer α par $-\alpha$ dans l'équation 6.3.

La fonction de transformation étant définie, nous pouvons désormais donner une formulation du jacobien J qui dépend des courbures et de la direction \mathbf{l} :

$$J = \frac{4\lambda_u^3 \lambda_v^3 (1+l'_z)^2}{(\lambda_u^2 \lambda_v^2 (1+l'_z)^2 + \lambda_v^2 l_u'^2 + \lambda_u^2 l_v'^2)^2}.$$

La dérivation complète du jacobien peut être trouvée en annexe IV.

Résultats

L'implémentation du *Light Warping* dans le cadre du rendu temps réel se fait en utilisant l'équation inverse du rendu, évaluée directement sur la carte graphique, en ne tenant pas compte de la visibilité des sources de lumière. La carte d'environnement est échantillonnée sous forme de lumières directionnelles qui sont passées au GPU à l'aide de l'extension OpenGL *Bindable uniform*. Le nombre de passes effectuées une fois le des-

cripteur géométrique calculé dépend du nombre de sources de lumière. Typiquement, 16 sources de lumière sont évaluées par passe.

Dans le cadre d'un rendu avec éclairage global ou lors de l'utilisation de l'*Ambient Occlusion* (AO), le moteur de rendu utilise à la fois le CPU et le GPU. Dans un premier temps, le CPU calcule à l'aide d'une passe de *Ray Casting* les buffers géométriques (normales, position, profondeur), stockés sous la forme de textures, en prenant entre 9 et 16 rayons par pixel. Ces textures sont ensuite passées au GPU qui calcule le descripteur géométrique et le stocke aussi sous forme de texture. Enfin, le moteur d'éclairage global utilise la texture du descripteur pour effectuer la transformation des directions incidentes par pixel.

La figure 6.3 présente nos résultats sur l'*Armadillo* pour deux types de matériaux. La partie gauche de chaque image montre le rendu obtenu sans effectuer de modifications de l'éclairage incident à la différence de la partie droite. Remarquez que les détails de l'objet sont nettement mis en avant pour les deux types de matériaux bien que pour le glossy, l'effet soit plus atténué que pour le diffus. Ces deux images utilisent un éclairage indirect à un seul rebond. Empiriquement, nous n'avons pas trouvé nécessaire de modifier les autres rebonds. D'ailleurs, les effets de l'éclairage indirect s'atténuent fortement au fur et à mesure des indirections. Notons que, lorsque l'on modifie la direction de l'éclairage incident, on modifie également la fonction de visibilité pour cette direction. Ce changement de visibilité semble améliorer la perception de l'objet. Ceci peut notamment se voir dans la figure 6.4 qui présente un rendu uniquement avec *Ambient Occlusion*. Rappelons que l'*Ambient Occlusion* $A(p)$ est définie comme la visibilité moyenne du point par rapport à son environnement. Elle donne une information scalaire sur l'occultation du point p :

$$A(p) = \frac{1}{\pi} \int_{\Omega_n} (\mathbf{n} \cdot \mathbf{l}) V(\mathbf{l}) d\mathbf{l} \quad (6.4)$$

où V est la fonction de visibilité et Ω_n l'hémisphère centré au point p et orienté selon la normale \mathbf{n} . L'*Ambient Occlusion* modifiée par la transformation $W(A(p))$ est alors

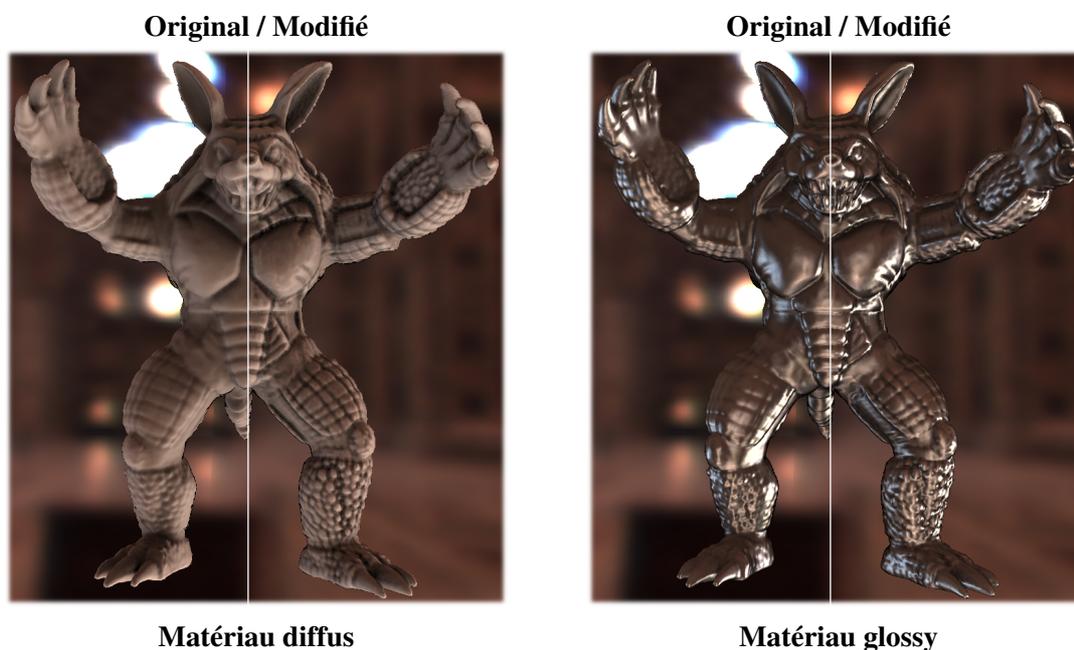


Figure 6.3 – Mise en évidence des détails en utilisant un rendu avec éclairage global à un rebond : l'*Armadillo* est rendu avec (gauche) un matériau diffus et (droite) un matériau légèrement spéculaire. Chaque côté de la figure montre la différence entre le rendu sans et avec *Light Warping*. Pour les deux cas, les paramètres sont identiques. Remarquez comment les détails de la surface sont mis en évidence de manière cohérente dans les deux cas.

donnée par :

$$W(A(p)) = \frac{1}{\pi} \int_{\Omega_n} \max(\mathbf{n} \cdot \mathbf{W}(\mathbf{l}), 0) V(\mathbf{W}(\mathbf{l})) d\mathbf{l}.$$

La figure 6.5 montre le résultat du *Light Warping* appliqué uniquement sur l'éclairage indirect. Tout comme dans la figure 6.4, la modification des directions amplifie le bruit présent dû à un échantillonnage trop faible de l'éclairage indirect. La pénalité induite par son introduction dans le cadre d'un moteur d'éclairage global est négligeable alors qu'elle peut atteindre 50% pour un rendu sur carte graphique (uniquement avec l'éclairage direct).

Discussion

L'approche proposée pour mettre en évidence les détails de la surface d'un objet est nettement plus souple que les approches précédentes. Par exemple, celle de Rusinkie-

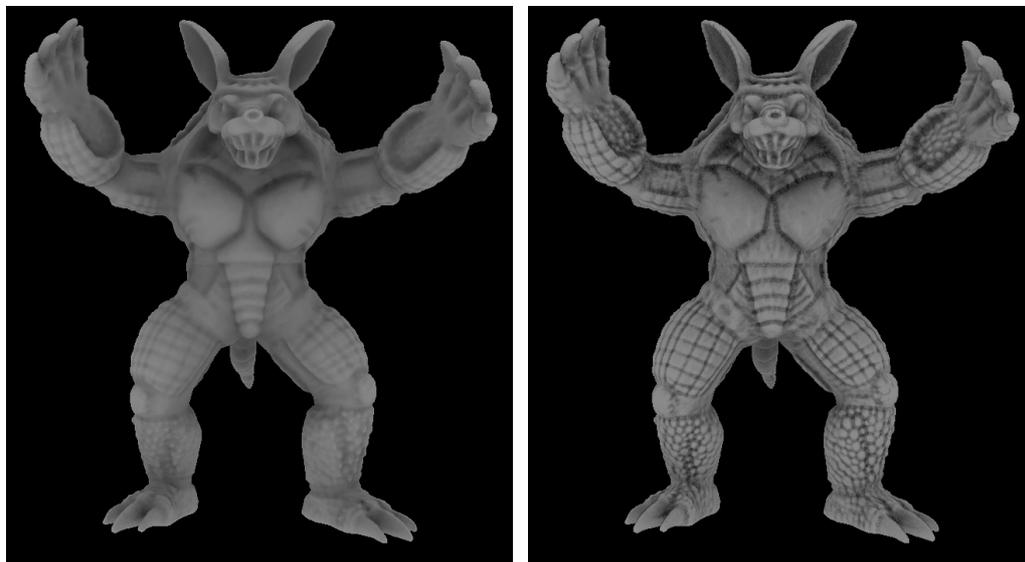


Figure 6.4 – Comparaison de l’Ambient Occlusion sur l’Armadillo. Lors du calcul de l’AO, nous modifions les directions de la lumière mettant ainsi en avant les détails de l’objet.

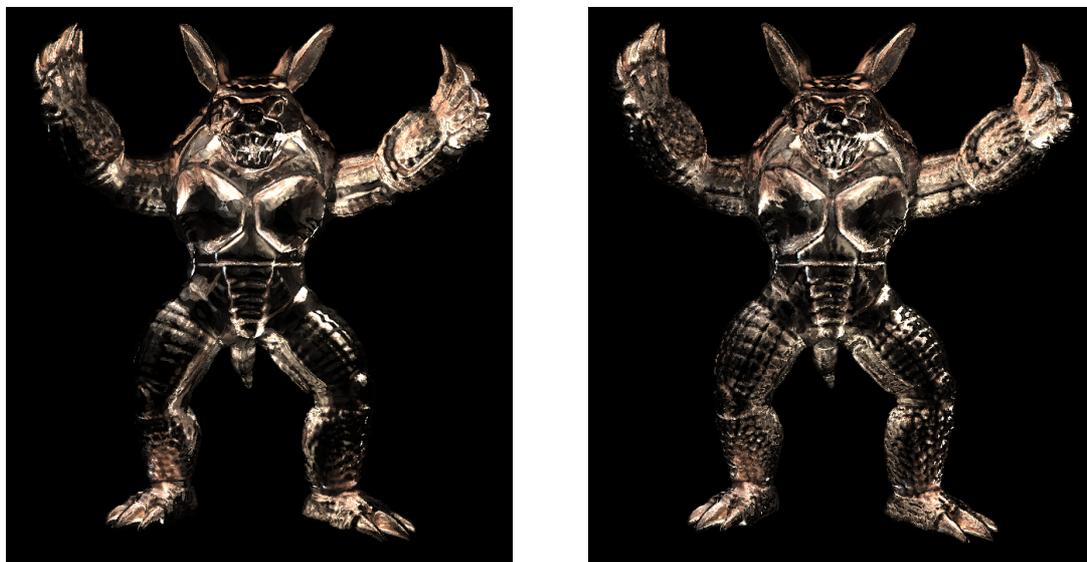


Figure 6.5 – Comparaison de l’éclairage indirect à un rebond calculé avec 25 rayons par hémisphère. En comparant les deux images, nous constatons que le warping a pour effet d’accroître le bruit.

wiecz *et al.* [RBD06] est limitée à une seule source de lumière et à un seul style et celle de Ritschel *et al.* [RSI⁺08] modifie sans discrimination la perception des matériaux, de

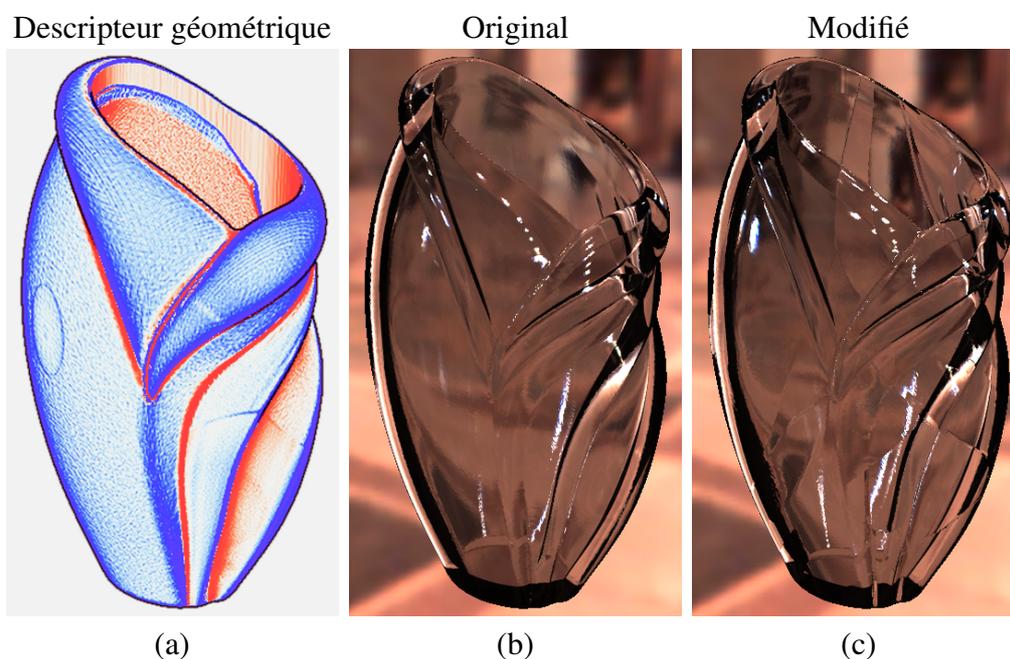


Figure 6.6 – (a) Illustration du descripteur géométrique révélant certains artefacts du scan (digitalisation) de l’objet. (b) Rendu avec réfraction et réflexion spéculaire parfaite sans application du warping. (c) Rendu avec warping, accentuant certains détails géométriques de l’objet.

la géométrie et l’éclairage incident. Le faible nombre de paramètres de notre approche nous apparaît aussi comme un avantage du point de vue de son utilisation. Néanmoins, comme montré dans les différentes figures, notre approche possède certaines limitations. Premièrement, elle nécessite un environnement lumineux qui possède suffisamment de variations, comme c’est souvent le cas dans les environnements lumineux naturels. En présence d’un environnement lumineux minimal, la technique présentera moins d’effets. De plus, bien qu’il soit possible d’appliquer le warping pour la réfraction et la réflexion spéculaire parfaite, la déformation des motifs lumineux obtenue ne constitue pas nécessairement une amélioration de la perception de la surface de l’objet. En effet, comme illustré à la figure 6.6, le warping met plutôt en avant les motifs réfractés rendant la compréhension de l’objet réfractif encore plus complexe. Puisque l’on transforme les directions autour du point de vue de manière bijective, si les directions proches du point de vue sont contractées, celles qui sont opposées seront dilatées. De plus, lorsque le pa-

ramètre α est trop élevé, les transitions dans le *shading* deviennent de plus en plus dures, ce qui a pour effet de modifier la perception du matériau de l'objet. Enfin, comme précédemment expliqué, lorsqu'on utilise le warping dans un moteur de rendu non temps réel, cela amplifie le bruit.

Ainsi, ces différentes limitations ont conduit à une seconde formulation qui ne modifie pas la direction incidente de la lumière mais son intensité.

6.4 Modification de l'intensité de l'éclairage incident

Si l'on fait l'hypothèse qu'une modification de l'intensité du signal lumineux incident est équivalente à une modification de la direction du signal, alors il est possible de remplacer la transformation des directions proposée à la section précédente par une adaptation d'intensité des sources de lumière. Ceci nous conduit à une reformulation de l'équation de réflexion et à l'introduction d'une nouvelle fonction de transformation qui s'avère être plus efficace et plus simple à évaluer.

Reformulation

Modifier l'intensité de l'éclairage incident peut être formulé de la manière suivante :

$$L(p \rightarrow \omega_o) = \int_S f_r(p, \omega_o, \mathbf{l}) (\mathbf{n} \cdot \mathbf{l}) L(p \leftarrow \mathbf{l}) W_p(\delta(\mathbf{l})) d\mathbf{l}$$

où $\delta(\mathbf{l})$ est une fonction d'importance sur la direction :

$$\delta(\mathbf{l}) : S \rightarrow [0, 1].$$

$W_p(\delta(\mathbf{l}))$ est la nouvelle fonction qui est un facteur multiplicatif appliqué à la radiance incidente $L(p \leftarrow \mathbf{l})$. La définition de la fonction d'importance $\delta(\mathbf{l})$ se fait en fonction des contextes d'utilisation. De plus, cette fonction peut varier en fonction de la composante (diffuse ou spéculaire) de la BRDF.

La nouvelle fonction W_p doit, comme la précédente, dépendre des courbures de la surface et être contrôlable par l'utilisateur, mais également posséder les propriétés sui-

vantes :

- **Invariance** : la fonction doit être invariante (*i.e.*, égale à 1) pour les surfaces planes et pour les zones de même importance.
- **Réciprocité** : les concavités et convexités doivent avoir une influence réciproque.
- **Symétrie** : la transformation doit avoir un effet symétrique par rapport à l'importance de la valeur renvoyée par $\delta(\mathbf{l})$.

Le polynôme rationnel linéaire ci-dessous satisfait ces propriétés :

$$W_{\alpha,\beta}(t) = \frac{\alpha e^\beta + t(1 - \alpha(1 + e^\beta))}{\alpha + t(e^\beta - \alpha(1 + e^\beta))}$$

où $t = \delta(\mathbf{l})$. Plus précisément, α agit comme une balance entre les concavités et les convexités. Quant à β , il est directement dépendant de la position p et des courbures κ_u et κ_v :

$$\beta_{\kappa_u,\kappa_v}(\mathbf{l}) = \left(\frac{\kappa_u + \kappa_v}{2} + \lambda \frac{\kappa_u - \kappa_v}{2} \right) l_u^2 + \left(\frac{\kappa_u + \kappa_v}{2} + \lambda \frac{\kappa_u - \kappa_v}{2} \right) l_v^2 + \left(\frac{\kappa_u + \kappa_v}{2} \right) l_z^2$$

où $\lambda \in [0, 1]$ contrôle l'anisotropie du warping. Si $\lambda = 0$, le warping devient isotrope alors que pour $\lambda = 1$ il est anisotrope (e.g. $\beta_{\kappa_u,\kappa_v}(\mathbf{u}) = \kappa_u$), de même que lorsque $\lambda = -1$ mais les directions sont alors inversées (e.g. $\beta_{\kappa_u,\kappa_v}(\mathbf{u}) = \kappa_v$). Lorsque \mathbf{l} est alignée avec l'axe \mathbf{z} , la transformation est forcée à être isotrope puisque la direction de la projection de \mathbf{l} dans le plan image est indéfinie.

En résumé, seulement deux paramètres α et λ sont contrôlés par l'utilisateur. En pratique, les courbures κ_u et κ_v définies sur \mathbb{R} doivent être ramenées sur l'intervalle $[0, 1]$ avec une fonction telle que celle proposée par Vergne *et al.* [VBGS08]. Remarquons aussi que cette formulation ne nécessite aucun calcul de jacobien et que la même équation peut être aussi bien utilisée pour le rendu temps réel que le rendu avec éclairage global. Dans le prochain paragraphe, nous présentons nos résultats avec cette nouvelle formulation ainsi que les différentes fonctions δ utilisées.

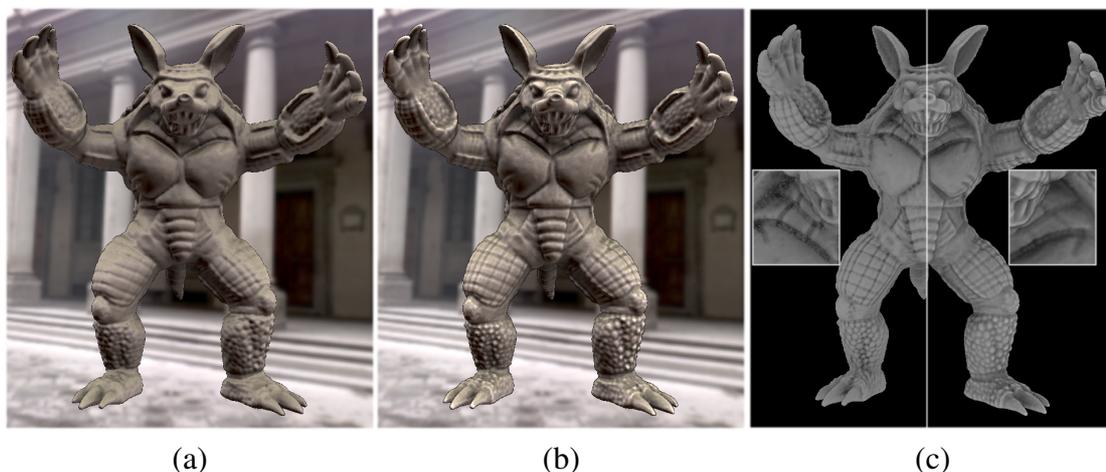


Figure 6.7 – Utilisation d’un environnement pré-filtré conjointement avec de l’Ambient Occlusion. Dans cette figure, une carte d’environnement pré-filtrée pour une BRDF lambertienne, et des valeurs d’Ambient Occlusion pré-calculées par sommet ont été utilisées. (a) Rendu sans modification de l’intensité de la radiance incidente. (b) Application de la transformation qui montre que, pour ce type de scénario, les détails géométriques de la surface sont mis en évidence. (c) Comparaison entre les deux formulations : (droite) la modification de l’intensité de la lumière incidente donne lieu à moins de bruit que (gauche) la modification de sa direction.

Résultats

Dans cette section, nous comparons à la première formulation, les résultats obtenus avec la nouvelle formulation pour l’*Ambient Occlusion*, l’éclairage direct et la réfraction.

La figure 6.7 montre la différence entre l’utilisation de l’*Ambient Occlusion* selon les deux types de transformations. Avec la seconde formulation, il est possible d’utiliser directement des environnements pré-filtrés (e.g. [KM00]), chose impossible avec la première formulation. Dans ce cas précis, l’équation discrète de l’éclairage devient alors :

$$L(p \rightarrow \omega_o) = W_{\alpha, \beta}(\delta(\mathbf{l}))A(p) \underbrace{\left(\frac{2\pi}{n_l} \sum_k^{n_l} f_r(p, \omega_o, \mathbf{l}_k) (\mathbf{n} \cdot \mathbf{l}_k) L(p \leftarrow \mathbf{l}_k) \right)}_{T(\mathbf{n}, \omega_o)}$$

où $A(p)$ représente l’*Ambient Occlusion* (cf. équation 6.4), n_l le nombre de sources de lu-

mière contenues dans la carte d'environnement et W le facteur multiplicatif de warping. Le terme $T(\mathbf{n}, \boldsymbol{\omega}_o)$ peut être pré-calculé de manière plus ou moins efficace en fonction de la spécularité de la BRDF (e.g. [KM00, KC08]). Dans le cas d'une BRDF diffuse, T ne dépend plus que de la normale \mathbf{n} et l'équation se simplifie encore :

$$L(p \rightarrow \boldsymbol{\omega}_o) = T(\mathbf{n}) \frac{\rho_d}{\pi} W_{\alpha,\beta}(\boldsymbol{\delta}(\mathbf{l})) A(p)$$

où ρ_d représente l'albédo diffus et $T(\mathbf{n})$ l'irradiance au point p pour la normale \mathbf{n} . Pour obtenir les images de la figure 6.7, nous avons utilisé l'irradiance moyenne de l'environnement pré-filtré comme fonction d'importance. Nous appliquons un opérateur de ton pour ramener sa valeur dans l'intervalle $[0, 1]$. De cette manière, la modification des intensités lumineuses s'effectue en considérant que l'irradiance maximale est la principale caractéristique de l'environnement lumineux. Comme aucun échantillonnage n'est effectué à la volée, la qualité de l'*Ambient Occlusion* obtenue est moins bruitée que celle obtenue précédemment avec les modifications directionnelles.

Dans la figure 6.8, l'adaptation d'intensité est utilisée avec le modèle d'Ashikhmin et Shirley [AS00a] en prenant une fonction d'importance $\boldsymbol{\delta}(\mathbf{l}) = \max(\mathbf{n} \cdot \mathbf{l}, 0)$ dans le cas de la réflexion diffuse, et $\boldsymbol{\delta}(\mathbf{l}) = \max(\mathbf{n} \cdot \mathbf{h}, 0)$ dans le cas de la réflexion spéculaire. Ceci donne plus de contrôle sur la mise en avant des détails de la surface suivant son éclairage et sa BRDF. Dans cette figure 6.8, le warping s'effectue principalement sur la composante spéculaire du modèle d'Ashikhmin et Shirley [AS00a].

Enfin, la figure 6.9 montre également qu'il est possible d'utiliser l'adaptation d'intensité dans le cadre de la réfraction et d'obtenir une amélioration de la perception des détails de la surface.

Discussion

La seconde formulation présente plusieurs avantages sur la première. Nos tests montrent que pour un nombre fixé de sources de lumière, la seconde formulation est toujours plus efficace en termes de rendu sur carte graphique. En outre, elle est plus simple à utiliser et à implémenter. Elle offre une certaine souplesse dans l'utilisation de

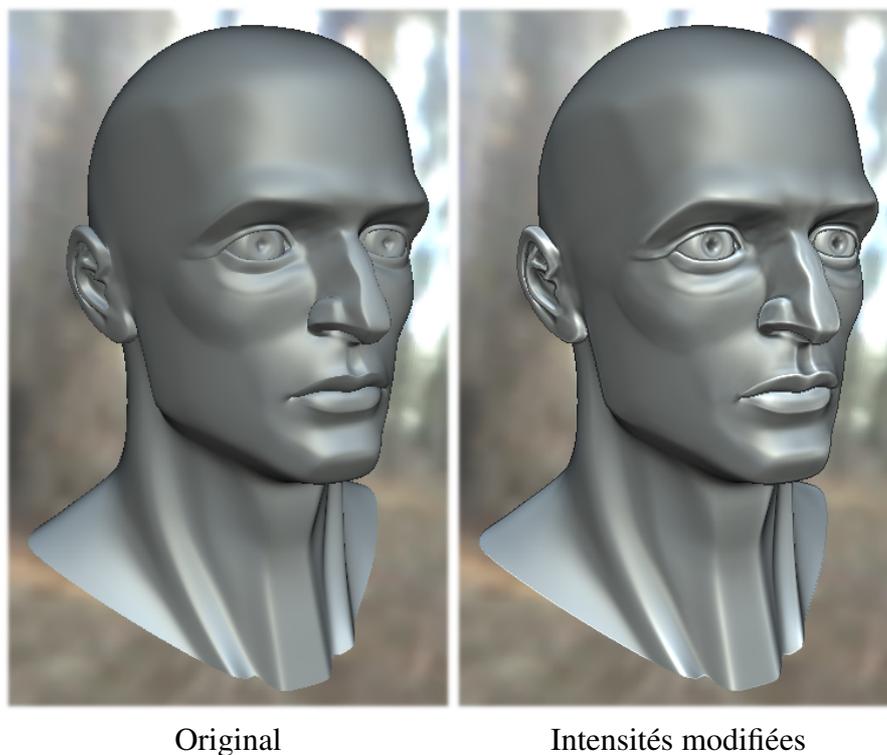


Figure 6.8 – L'image de gauche montre la version originale du rendu (sans visibilité). Dans l'image de droite, l'adaptation d'intensité appliquée à la composante spéculaire de la BRDF permet de faire ressortir les détails de la surface autour des yeux, de la bouche et des oreilles.

cartes d'environnement pré-filtrées ou encore de la réfraction.

Lorsque l'objet est hautement réfléchissant, la compression et la dilation des motifs lumineux réfléchis par la surface semblent le meilleur moyen de donner de l'information sur la forme de l'objet, résultat impossible à obtenir avec la seconde formulation. Si l'objet contient beaucoup de détails géométriques et que son matériau est plutôt de type glossy, la seconde formulation donne de meilleurs résultats avec, notamment, moins d'aliasing. Néanmoins, contrairement à la première formulation, les changements de perception du matériau de la surface nous apparaissent comme nettement plus forts et accentués.

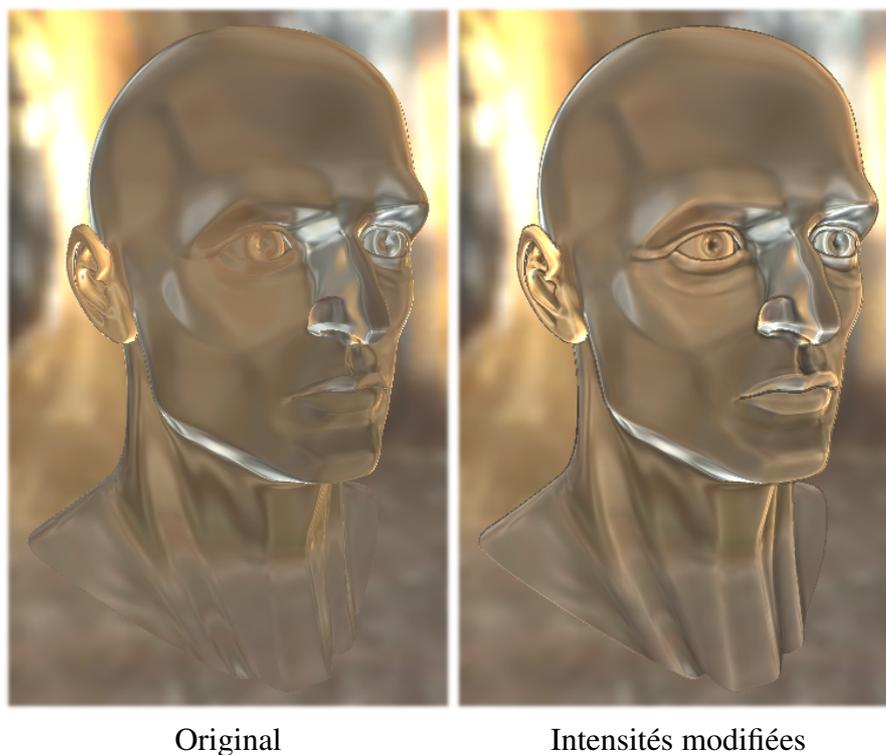


Figure 6.9 – Illustration de l’adaptation d’intensité dans le cas de la réfraction approximée avec un simple rayon interrogeant directement la carte d’environnement. La perception des yeux, des oreilles et de la bouche est nettement améliorée sur l’image de droite comparativement à l’image d’origine à gauche.

6.5 Travaux futurs

Une possibilité pour essayer de repousser les limitations soulevées dans la section précédente serait d’essayer de combiner les deux formulations de warping. Une première étape à effectuer serait de faire une analyse des cas d’équivalence entre les deux méthodes de warping proposées. Cette analyse pourrait se faire dans le domaine fréquentiel ou avec une analyse au premier ordre telle que proposée par Ramamoorthi *et al.* [RMB07].

Réussir à extraire la classe des fonctions de transformation qui permettraient de mettre en avant les détails géométriques d’une surface tout en garantissant l’absence de modifications de la perception du matériau attaché à la surface reste un travail d’analyse perceptuelle à effectuer.

Dans un contexte strict d'édition, il serait aussi intéressant d'essayer de créer des environnements lumineux virtuels spécifiquement utilisés pour mettre en avant certains détails à des endroits précis de la géométrie de l'objet tels que les salles de néons qui permettent d'observer les courbures sur les voitures.

CONCLUSION DE LA SECONDE PARTIE

Après nous être intéressés à la réflexion de l'éclairement incident dans la première partie, nous avons abordé, dans cette seconde partie, des problèmes directement liés à l'éclairement incident. Dans le chapitre 5, nous avons proposé une nouvelle structure et une représentation de l'éclairement indirect et nous avons présenté différentes applications telles que le transfert progressif de l'éclairement ou encore un système de cache pour un rendu de haute qualité. Pour toutes ces applications, notre méthode a prouvé sa robustesse à la variation géométrique, la rendant très adaptée pour traiter des objets très détaillés. Nous avons proposé des directions futures de recherche pour optimiser l'organisation spatiale de notre structure et l'utiliser comme outil d'édition de l'éclairement. Enfin, nous avons aussi mis en évidence le besoin de développer de nouvelles représentations directionnelles.

Dans le chapitre 6, nous nous sommes ensuite intéressés à deux formulations pour modifier l'éclairement incident en vue de mieux faire ressortir les détails d'une surface. Ces transformations sur les directions et l'intensité des sources de lumière semblent bien aboutir à ce but mais leur implémentation pose des questions plus fondamentales sur notre perception des formes qu'il conviendrait d'approfondir. Notre représentation vectorielle présentée au chapitre 5 semble aussi pouvoir être intégrée dans cette approche. Cette combinaison nous semble prometteuse dans le futur pour aboutir à une nouvelle forme d'édition de l'éclairement plus localisée.

CONCLUSION ET DIRECTIONS FUTURES

Résumé des contributions

Dans ce mémoire de thèse, nous nous sommes intéressés à la représentation des matériaux via la fonction de réflectance (BRDF), et à l'éclairage incident.

Nous avons introduit deux nouveaux modèles pour les BRDFs. Le premier est destiné à faciliter la création des reflets spéculaires grâce à une interface intuitive reposant sur des outils d'esquisse et de peinture. Le second modèle autorise une meilleure représentation et compression des mesures de BRDFs isotropes. Une édition intuitive à base de courbes 1D est également proposée.

Nous avons présenté une nouvelle structure de données ainsi qu'une représentation vectorielle pour capturer l'éclairage incident. La représentation vectorielle est robuste aux variations géométriques de la scène. Enfin, nous avons introduit deux transformations de l'équation de la réflexion afin de mettre en avant les détails géométriques. Chacune de ces transformations modifie soit la direction, soit l'intensité de l'éclairage incident, et leur effet est contrôlable par l'utilisateur.

Travaux futurs

La réflexion

Le travail sur les mesures de BRDFs fait ressortir la rareté (relative) des données sur les matériaux en infographie surtout, considérant la grande richesse de ceux qui nous entourent. Par exemple, aucune donnée n'est disponible pour les matériaux rétro-réfléchissants ; les matériaux anisotropes n'ont pas assez été mesurés, à l'instar des matériaux issus de plusieurs couches non-homogènes. A court terme, il nous apparaît donc important d'effectuer de nouvelles mesures afin de raffiner le nouveau modèle introduit et de généraliser aux fonctions d'apparence de plus haute dimension (SVBRDF, BSSRDF).

A moyen terme, nous aimerions proposer un modèle de BRDFs qui unifie les besoins artistiques et ceux liés aux simulations physiques. Dans une démarche artistique, l'inter-

face utilisateur ainsi que les métaphores d'édition et de création sont aussi importantes que le modèle lui-même. Notre modèle d'édition de reflets spéculaires nous donne des indications sur le type d'interface à utiliser, tels les outils de dessin et d'esquisse. En revanche, la question du type d'éclairage le plus approprié pour éditer ou créer une BRDF reste à résoudre. Il semble plus simple pour un artiste d'éditer une BRDF avec une seule source de lumière. En revanche, la BRDF ainsi modélisée, excepté dans certains cas particuliers (e.g. rendu expressif), est rarement utilisée dans un contexte d'éclairage aussi simple. L'environnement lumineux utilisé sera souvent complexe (*all-frequency*) comme celui d'une carte d'environnement ou celui d'une scène avec éclairage global ou direct avec de multiples sources de lumière. Si l'on souhaite éditer dans de tels environnements, des améliorations au niveau des techniques de rendu sont nécessaires afin de donner un retour continu à l'artiste sur ses actions. Un exemple simple d'application dans le cadre de l'industrie automobile serait la création des peintures de carrosseries de voiture dont l'artiste souhaite voir le résultat aussi bien avec un ciel d'été que d'hiver, ou encore de nuit. Enfin, des études utilisateurs restent nécessaires pour valider l'interface utilisateur et les métaphores d'édition proposées. Ces études étant spécialisées, elles devraient se faire en partenariat avec une équipe dont la recherche est axée sur les interactions.

A plus long terme, nous aimerions travailler sur la multi-résolution de l'apparence. Bien qu'il nous semble complexe d'obtenir un modèle s'adaptant à toutes les échelles de l'apparence, nous pensons qu'il serait intéressant de dériver différents modèles pouvant être combinés, au besoin, de manière lisse et dynamique. Dans le cadre de l'édition, nous aimerions également travailler sur la question d'existence d'un matériau créé par un artiste. Naturellement, il est possible de restreindre les actions de l'artiste afin de garantir que la matériau virtuellement créé sera physiquement réalisable. Cependant, nous pensons qu'il devrait être possible, dans certains cas, de répondre à la question de l'existence si les contraintes de réalisation du matériau peuvent être correctement modélisées.

L'éclairage incident

La représentation directionnelle utilisant six vecteurs pour reconstruire l'irradiance offre une robustesse géométrique et ce, pour un faible impact en mémoire. Afin de pouvoir utiliser notre structure d'éclairage avec des matériaux dont la fréquence est plus élevée, nous devons utiliser une représentation de l'éclairage incident plus fine. Ainsi, nous aimerions réfléchir à de nouvelles représentations directionnelles, qui pourraient ou non être vectorielles, mais qui autoriseraient une reconstruction de la radiance incidente. Notons que la place destinée à stocker l'éclairage incident doit rester faible et ce, aussi bien pour les jeux vidéo que pour les applications industrielles. En revanche, pour les simulations physiques effectuées le plus souvent sur des grilles (*clusters*) de calcul, il est plus important que le système de cache proposé soit facilement distribuable et à mise à jour facile via le réseau.

Dans la droite lignée de ces travaux, nous aimerions proposer un nouveau système de cache dont le calcul pourrait facilement utiliser les nouvelles architectures matérielles parallèles. Ceci permettrait d'avoir des pré-visualisations de rendu avec éclairage global nettement plus rapides et dont le cache serait plus facilement réutilisable en fonction des actions (changement de point de vue, modification des sources de lumière, édition des matériaux, déplacement de la géométrie, etc.) effectuées par l'utilisateur. Exception faite des jeux vidéo, ou plus généralement d'un contexte où tous les éléments doivent être dynamiques, les temps de mises à jour du rendu ne doivent pas forcément être identiques mais peuvent dépendre de l'action d'édition effectuée (e.g. *Shading Cache* [TPWG02]).

A plus long terme, dans un cadre d'édition du flux lumineux, il semblerait que la représentation vectorielle soit la plus «intuitive», probablement parce qu'elle est proche de celle utilisée en visualisation scientifique.

L'énergie réfléchie

Nous avons utilisé l'équation du rendu et ses deux principaux termes, l'éclairage incident et la BRDF, comme fil conducteur pour les travaux de recherche de cette thèse. Rappelons que la radiance finale d'un pixel est le résultat d'une convolution de l'éclai-

rement incident par la fonction de réflectance d'un point vu à travers ce pixel. Nous avons travaillé sur chacun des termes et proposé des extensions et perspectives futures pour chacun d'eux. Plutôt que de contraindre l'utilisateur à éditer uniquement un seul des deux termes ou les deux à la fois, nous pensons qu'il serait intéressant de laisser l'utilisateur éditer le résultat de la convolution. Le système se chargerait alors de le guider vers l'édition du terme le plus approprié, voire de lui proposer une solution exacte lorsque cela est possible. Une telle approche de plus haut niveau ouvrirait sûrement la voie vers de nouveaux outils d'édition.

ANNEXE I

RADIOMÉTRIE ET PHOTOMÉTRIE

La radiométrie est l'étude de la lumière et plus particulièrement de sa mesure physique sur l'ensemble du spectre. La photométrie qui en découle prend en compte la perception de l'énergie lumineuse par le système visuel humain qui est uniquement capable d'observer les phénomènes compris approximativement entre 380 et 780 nm.

On trouvera dans le tableau I.I une correspondance entre les quantités et unités radiométriques et photométriques.

I.1 Quantités radiométriques

Flux énergétique spectral ou puissance spectrale

La quantité de lumière émanant/quittant/traversant une surface est appelée énergie rayonnante spectrale (cf. annexe I.2). Si l'on considère la variation de cette dernière par rapport au temps, on établit le flux énergétique spectral¹ Φ_λ dont le concept est illustré à la figure I.1(a).

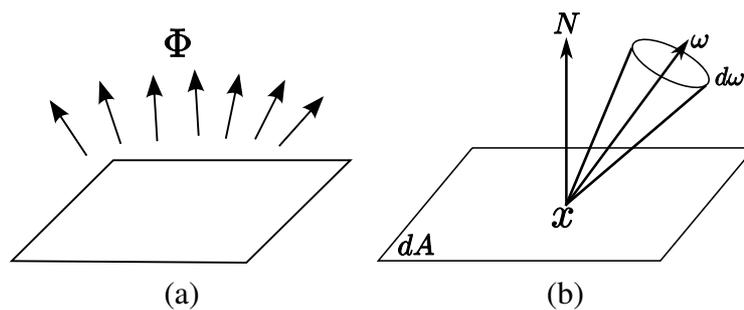


Figure I.1 – (a) Notion de flux. (b) Repérage de la radiance. La radiance est un scalaire valable pour un angle solide $d\omega$ repéré par une direction ω .

En synthèse d'images, on considère que le système a atteint un état énergétique lumineux stable ; l'aspect temporel du flux est donc ignoré.

¹La littérature anglo-saxonne parle de *spectral radiant flux* en ce qui concerne le flux énergétique spectral et de *spectral radiant power* pour la puissance énergétique.

Irradiance et radiosit  spectrales

Les quantit s permettant de calculer l' nergie lumineuse qui part ou qui arrive sur une surface donn e sont l'irradiance et la radiosit  spectrales. Si l'on ne consid re pas la provenance de l' nergie, on parle de densit  de flux  nerg tique.

L'irradiance² spectrale, $E_\lambda(x)$, en un point x , exprim e en Watt par m tre carr  par nanom tre ($W \cdot m^{-2} \cdot nm^{-1}$) correspond au flux  nerg tique spectral incident sur une surface par aire de surface unitaire :

$$E_\lambda(x) = \frac{d\Phi_\lambda}{dA} \quad (I.1)$$

avec dA surface unitaire  l mentaire.

La radiosit  spectrale B_λ , parfois appel e excitance  nerg tique, correspond au flux  nerg tique partant d'une surface par aire de surface unitaire. Tout comme l'irradiance, elle est aussi exprim e en Watt par m tre carr  par nanom tre ($W \cdot m^{-2} \cdot nm^{-1}$). Elle est d finie par l' quation suivante :

$$B_\lambda(x) = \frac{d\Phi_\lambda}{dA} . \quad (I.2)$$

Notion d'angle solide et d'angle solide diff rentiel

Dans ce document, les coordonn es sph riques sont rep r es par les angles θ et ϕ , o  θ est   valeur sur l'intervalle $[0, \frac{\pi}{2}]$ et repr sente l'angle co-latitudinal, tandis que ϕ est   valeur sur l'intervalle $[0, 2\pi]$ et repr sente l'angle azimutal (longitudinal), comme illustr    la figure I.2.

Une quantit  fondamentale en  clairage global associ e   la notion de direction est l'angle solide (cf. figure I.3) qui permet d'exprimer   la fois une portion de direction et une aire infinit simale.

L'angle solide Ω au point x , d fini par une aire A sur la sph re, est d fini par :

$$\Omega_x = \frac{A}{r^2} \quad (I.3)$$

²Le terme irradiance est un anglicisme courant. Le terme fran ais officiel est «  clairage  nerg tique ».

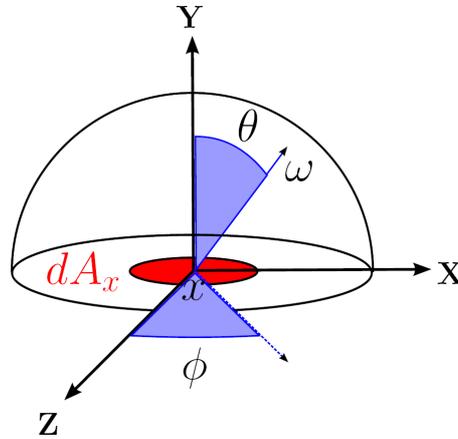


Figure I.2 – Repérage hémisphérique des directions.

avec r le rayon de la sphère.

Bien que sans dimension, les angles solides sont exprimés en stéradian (sr). Pour calculer l'angle solide d'un objet quelconque, on doit projeter la surface de cet objet sur la sphère unité.

Pour pouvoir intégrer des fonctions sur l'hémisphère, on définit l'angle solide différentiel, dont une illustration est donnée en figure I.4(a), centré autour d'une direction $\boldsymbol{\omega} = (\theta, \phi)$, notée $d\boldsymbol{\omega}$, par :

$$d\boldsymbol{\omega} = d\phi d\theta \sin \theta. \quad (\text{I.4})$$

Comme le montre la figure I.4(a), le facteur $\sin \theta$ s'explique par le fait que la variation d'angle solide est plus grande à l'horizon qu'au pôle.

Pour pouvoir passer d'une formulation hémisphérique à surfacique, on a besoin d'exprimer l'élément différentiel d'angle solide en fonction de l'élément différentiel de surface :

$$d\boldsymbol{\omega}_x = \frac{\cos \theta_y dA_y}{d_{xy}^2} \quad (\text{I.5})$$

avec

- θ_y angle entre la normale au point y et $\boldsymbol{\omega}_x$
- d_{xy} la distance euclidienne entre x et y
- dA_y l'élément différentiel de surface au point y

dont une illustration est donnée à la figure I.4(b).

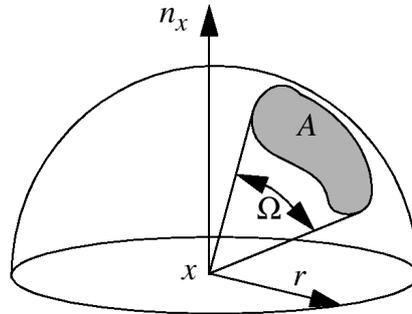


Figure I.3 – Angle solide généré par une surface A au point x .

Grâce à la notion d'angle solide, on peut définir de nouvelles quantités radiométriques qui tiennent compte de la direction de la lumière.

Intensité énergétique spectrale et radiance spectrale

L'intensité énergétique spectrale³ $I_\lambda(\boldsymbol{\omega})$, exprimée en Watt par stéradian ($W \cdot sr^{-1}$), correspond au flux énergétique par angle solide unitaire :

$$I_\lambda(\boldsymbol{\omega}) = \frac{d\Phi_\lambda}{d\boldsymbol{\omega}}. \quad (\text{I.6})$$

La radiance⁴ spectrale L_λ , exprimée en Watt par mètre carré par stéradian par nanomètre ($W \cdot m^{-2} \cdot sr^{-1} \cdot nm^{-1}$), est définie par le flux énergétique spectral par surface unitaire projetée et par angle solide différentiel :

$$L_\lambda = \frac{d^2\Phi_\lambda}{\cos \theta dA d\boldsymbol{\omega}}$$

avec θ angle entre la direction $\boldsymbol{\omega}$ et la normale de la surface.

Afin de différencier la radiance incidente de la radiance sortante, on utilise la notation suivante :

- $L_\lambda(x \rightarrow \boldsymbol{\omega})$ représente la radiance quittant le point x dans la direction $\boldsymbol{\omega}$.
- $L_\lambda(x \leftarrow \boldsymbol{\omega})$ représente la radiance incidente au point x de direction $\boldsymbol{\omega}$.

³Spectral radiant intensity dans la littérature anglo-saxonne.

⁴Il s'agit une fois encore d'un anglicisme courant dans la communauté scientifique de l'éclairage global. Le terme français adéquat est «luminance énergétique».

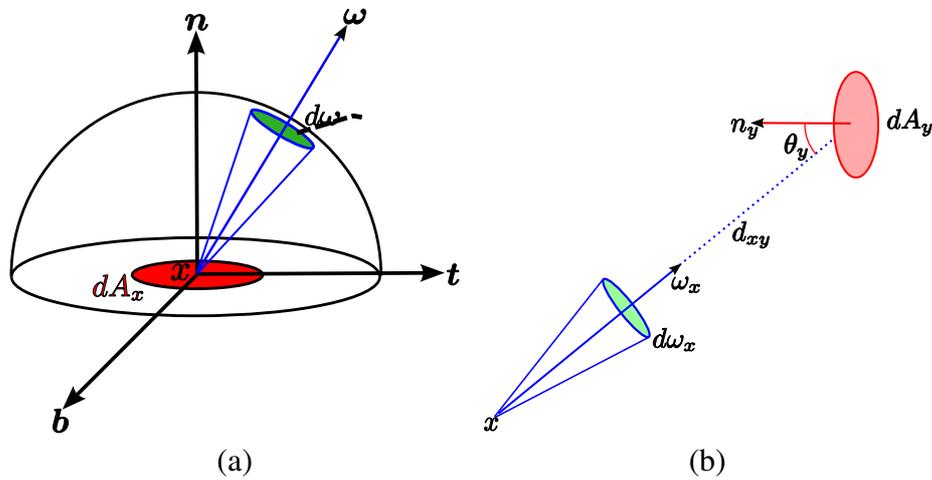


Figure I.4 – (a) Angle solide différentiel $d\boldsymbol{\omega}$ pour une direction $\boldsymbol{\omega}$. (b) Quantités en jeu pour la formulation surfacique de l'angle solide différentiel $d\boldsymbol{\omega}_x$.

L'une des propriétés fondamentales de la radiance est son invariance dans le vide le long d'un rayon, c'est-à-dire :

$$L_\lambda(x \rightarrow y) = L_\lambda(y \leftarrow x). \quad (\text{I.7})$$

La radiance spectrale est sans conteste la quantité fondamentale en éclairage global. Notre système visuel, comme les capteurs, y sont sensibles. Malheureusement, c'est une quantité complexe puisque c'est une fonction de dimension 6 (3 pour la position, 2 pour la direction et 1 pour la longueur d'onde).

Equations reliant les diverses quantités radiométriques

Les quantités radiométriques peuvent s'exprimer en fonction de la radiance spectrale :

$$\Phi_\lambda = \int_A \int_\Omega L_\lambda(x \rightarrow \boldsymbol{\omega}) \cos \theta \, d\boldsymbol{\omega} \, dA_x \quad (\text{I.8})$$

$$E_\lambda(x) = \int_\Omega L_\lambda(x \leftarrow \boldsymbol{\omega}) \cos \theta \, d\boldsymbol{\omega} \quad (\text{I.9})$$

$$B_\lambda(x) = \int_\Omega L_\lambda(x \rightarrow \boldsymbol{\omega}) \cos \theta \, d\boldsymbol{\omega} \quad (\text{I.10})$$

avec :

- Φ_λ le flux énergétique spectral
- $E_\lambda(x)$ l'irradiance spectrale
- $B_\lambda(x)$ la radiosité spectrale
- L_λ la radiance spectrale.

Quantité radiométrique	Unité	Quantité photométrique	Unité
Flux énergétique	W	Flux lumineux	lm
Irradiance	$W \cdot m^{-2}$	Illuminance	$lm \cdot m^{-2}$ i.e. lx
Intensité énergétique	$W \cdot sr^{-1}$	Intensité lumineuse ⁵	$lm \cdot sr^{-1}$ i.e. cd
Radiance	$W \cdot m^{-2} \cdot sr^{-1}$	Luminance ⁵	$lm \cdot m^{-2} \cdot sr^{-1}$ i.e. $cd \cdot m^{-2}$ i.e. nit

Tableau I.I – *Tableau comparatif des quantités radiométriques et photométriques. Watt (W). Lumen (lm). Watt par mètre carré ($W \cdot m^{-2}$). Candela (cd). Watt par mètre carré par stéradian ($W \cdot m^{-2} \cdot sr^{-1}$). Lumen par stéradian ($lm \cdot sr^{-1}$).*

Correspondance entre les grandeurs radiométriques et photométriques

Le tableau I.I présente les principales correspondances entre les grandeurs radiométriques et les grandeurs photométriques. Notons que, contrairement à la radiance, la luminance ne fait pas partie du Système International d'unité (SI) et que le candela est l'une des sept quantités de base du SI⁶. Le lumen défini à partir du candela représente le flux lumineux émis par une source ponctuelle isotropique d'intensité lumineuse égale à 1 candela.

Emission et réception de la lumière

La lumière émise sous forme de photons provient de sources lumineuses comme le soleil, les lampes ou encore des phénomènes naturels tels que le feu. Pour chaque source lumineuse, il est normalement nécessaire de connaître la distribution :

⁵Il s'agit aussi d'un anglicisme. Le terme français approprié est «luminance lumineuse».

⁶Les sept quantités de base du SI sont : m (distance exprimée en mètre), kg (masse exprimée en kilogramme), s (temps exprimé en seconde), A (intensité électrique exprimée en ampère), K (température exprimée en degré Kelvin), mol (quantité de matière, nombre de moles) et cd (intensité lumineuse exprimée en candela).

- spatiale,
- directionnelle,
- spectrale.

Un exemple simple est celui d'une source lumineuse ponctuelle placée en O_L qui émet donc dans toutes les directions avec une puissance spectrale Φ_λ . L'irradiance reçue en un point x d'une surface est alors donnée par :

$$E_\lambda(x) = \frac{\Phi_\lambda \cos \theta}{4\pi r^2}$$

avec :

- r distance entre x et la source lumineuse
- θ est l'angle entre la normale de la surface au point x et le vecteur \mathbf{xO}_L .

Intuitivement, on peut voir que le facteur $\cos \theta$ est nécessaire pour tenir compte du fait qu'une surface qui fait face à la lumière recevra plus d'énergie lumineuse qu'une surface qui lui tourne le dos.

Formulation implicite des longueurs d'onde. Espace de couleur

En infographie, on représente en général ⁷ la radiance en utilisant trois couleurs. Si l'on se place dans l'espace de couleur RGB⁸, la radiance sera alors stockée et représentée par le triplet (L_R, L_G, L_B) .

$L_{R,G,B}$ peut prendre n'importe quelle valeur sur \mathbb{R}_+ . Par conséquent, afin d'afficher ces valeurs sur un dispositif à plage limitée tels nos écrans habituels, il convient d'effectuer une transformation appropriée appelée correction de ton ou *tone-mapping* (e.g. [War94, Sch94d, TR93, DD00]). Lorsqu'il n'y a aucune ambiguïté permise, on omet la notation spectrale des quantités radiométriques.

⁷A l'exception de moteur de rendu spectral tel que *Maxwell*

⁸RGB pour *Red, Green, Blue*. Le sigle français équivalent est RVB pour Rouge, Vert, Bleu.

I.2 Définition quantique de l'énergie rayonnante spectrale

L'équation reliant la constante universelle de Planck et l'énergie d'un photon, au sens physique du terme, quantité radiométrique élémentaire, permet d'établir la relation entre l'énergie du photon pour une longueur d'onde donnée :

$$e_{\lambda} = \frac{hc}{\lambda} \quad (\text{I.11})$$

avec

- h la constante universelle de Planck ($h \simeq 6,6261 \cdot 10^{-34} J \cdot s$)
- c la vitesse de la lumière dans le vide ($c \simeq 299\,792\,458 \text{ ms}^{-1}$).

Il en découle deux quantités radiométriques :

- l'énergie rayonnante spectrale⁹ Q_{λ} , exprimée en Joule par nanomètre ($J \cdot \text{nm}^{-1}$), qui est définie pour un ensemble de n photons ayant la même longueur d'onde :

$$\begin{aligned} Q_{\lambda} &= n e_{\lambda} \\ Q_{\lambda} &= n \frac{hc}{\lambda}. \end{aligned} \quad (\text{I.12})$$

- l'énergie rayonnante¹⁰ Q , exprimée en Joule (J) qui est l'énergie obtenue en intégrant l'énergie rayonnante spectrale sur tout le spectre :

$$Q = \int_{\lambda=0}^{\infty} Q_{\lambda} d\lambda. \quad (\text{I.13})$$

⁹Le terme utilisé dans la littérature anglo-saxonne est *spectral radiant energy*.

¹⁰Le terme utilisé dans la littérature anglo-saxonne est *radiant energy*.

I.3 Définition du flux énergétique spectral en fonction de la température

Si la lumière est donnée par sa température, la formule de Planck rappelée ci-dessous permet de calculer son flux énergétique spectral par :

$$\Phi_{\lambda} = \frac{2\pi C_1}{\lambda^5 \left(e^{\frac{C_2}{\lambda T}} - 1 \right)} \quad (\text{I.14})$$

avec

- T la température du corps donnée en degré Kelvin (K)
- $C_1 = hc^2$
- $C_2 = \frac{hc}{k}$
- k la constante de Boltzmann ($k = 1,3806505 \cdot 10^{-23} J \cdot K^{-1}$).

ANNEXE II

MÉTHODES NUMÉRIQUES D'APPROXIMATION À BASE DE POLYNÔMES RATIONNELS

II.1 Introduction

Dans cette annexe, nous présentons les formulations mathématiques des algorithmes d'approximation utilisés au chapitre 3 pour approximer l'ISF à partir de mesures. Le *Differential Correction Algorithm* (DCA) [PBR71], détaillé dans la prochaine section, est un algorithme à convergence garantie mais dont la vitesse peut être faible. Dès lors, nous utilisons comme première approximation, pour initialiser le DCA, la solution retournée par l'algorithme proposé par Papamarkos [Pap88] expliqué à la section II.2.

II.2 DCA pour l'*Isotropic Shading Function*

De manière générale nous voulons approximer un ensemble de données (mesures) que l'on notera $\{(x_i, f_i), i = 1..I\}$ à l'aide d'un polynôme rationnel $R(x)$ exprimé sous la forme :

$$R(x) = \frac{P(x)}{Q(x)} = \frac{\sum_{m=0}^M p_m b_m(x)}{\sum_{n=0}^N q_n b_n(x)}$$

où $b_i(x)$ est une base polynomiale. Au chapitre 3, dans le cas 2D, les mesures ont été projetées dans l'espace ISF (cf. section 3.2), par conséquent $x = (r, s)$ et nous cherchons la meilleure approximation des mesures, au sens de la norme L^∞ , en minimisant l'expression suivante :

$$\Delta = \max_{i=1..I} \{|R(x_i) - f_i|\}. \quad (\text{II.1})$$

Afin d'obtenir une représentation valide sur le domaine d'observation, il est nécessaire d'effectuer la minimisation sous la contrainte suivante :

$$Q(x_i) > 0, i = 1..I. \quad (\text{II.2})$$

Notons que cette contrainte garantit que le dénominateur est toujours strictement positif sur les points échantillonnés (mesures) mais pas entre eux. Ceci peut poser des problèmes si l'échantillonnage est insuffisant ou lorsque les degrés de $R(x)$ sont trop élevés, par rapport à la variation des données, et introduisent alors de fortes oscillations. En pratique, nous n'avons pas eu de problèmes d'échantillonnage avec les données de la base MERL-MIT (cf. [MPBM03b]).

Comme les fonctions $P(x)$ et $Q(x)$ peuvent être réduites ou augmentées sans changer la valeur $R(x)$, on peut alors introduire une condition de normalisation sur les coefficients de $Q(x)$:

$$\max_{n=1..N} \{|q_n|\} \leq 1 \Leftrightarrow q_n \in [-1, 1], n = 0..N. \quad (\text{II.3})$$

Dans le contexte d'approximation de BRDF, nous devons ajouter une contrainte supplémentaire au DCA original. En effet, la BRDF étant une quantité non-négative, et puisque $Q(x) > 0$, il est nécessaire de forcer le numérateur $P(x)$ à être positif ou nul aux positions d'échantillonnage :

$$P(x_i) \geq 0, i = 1..I. \quad (\text{II.4})$$

En résumé, le problème linéaire à résoudre est de type *minmax* avec les contraintes données en II.2, II.3 et II.4.

Notons Δ_k , l'erreur L^∞ à l'étape k du DCA :

$$\Delta_k = \max_i |f_i - R_k(x_i)| \quad \text{avec} \quad R_k(x) = P_k(x)/Q_k(x).$$

Après chaque étape, nous résolvons, à l'étape $k + 1$, le problème d'optimisation suivant :

minimiser Δ sous les contraintes :

$$\left\{ \begin{array}{l} [f_i + \Delta_k] Q(x_i) - P(x_i) + Q_k(x_i) \Delta \geq 0, \quad i = 1..I \\ [-f_i + \Delta_k] Q(x_i) + P(x_i) + Q_k(x_i) \Delta \geq 0, \quad i = 1..I \\ P(x_i) \geq 0, \quad i = 1..I \\ Q(x_i) > 0, \quad i = 1..I \\ p_m \in \mathbb{R}, \quad m = 0..M \\ q_n \in [-1, 1], \quad n = 0..N \\ \Delta \in \mathbb{R}. \end{array} \right.$$

Ce système linéaire avec contraintes peut être résolu avec n'importe quelle librairie (e.g. [GLP, CGA]). Le DCA s'arrête lorsque $\Delta_{k+1} > \Delta_k$. Le DCA a uniquement besoin d'un polynôme rationnel de départ qui satisfasse les contraintes du problème linéaire comme graine d'initialisation de l'algorithme.

II.3 Formulation dans la base monomiale

Si l'on utilise la base monomiale, la formulation du problème est légèrement simplifiée. En effet, le polynôme rationnel $R(x)$ s'écrit alors :

$$R(x) = \frac{P(x)}{Q(x)} = \frac{\sum_{m=0}^M p_m x^m}{\sum_{n=0}^N q_n x^n}.$$

Cependant, le numérateur doit toujours être positif, et cela force $q_0 > 0$. Dans ce cas précis, la condition de normalisation se simplifie à $q_0 = 1$. Les contraintes sur les coefficients q_n (cf. équation II.3) deviennent superflues et il y a un coefficient de moins à estimer pour dénominateur. En conséquence, on peut réécrire $R(x)$:

$$R(x) = \frac{P(x)}{Q(x)} = \frac{\sum_{m=0}^M p_m x^m}{1 + \sum_{n=1}^N q_n x^n} = \frac{P(x)}{1 + Q'(x)}$$

et le problème linéaire :

minimiser Δ sous les contraintes :

$$\left\{ \begin{array}{l} [f_i + \Delta_k] Q'(x_i) - P(x_i) + Q_k(x_i)\Delta \geq -[f_i + \Delta_k], \quad i = 1..I \\ [-f_i + \Delta_k] Q'(x_i) + P(x_i) + Q_k(x_i)\Delta \geq [f_i - \Delta_k], \quad i = 1..I \\ P(x_i) \geq 0, \quad i = 1..I \\ Q'(x_i) > -1, \quad i = 1..I \\ p_m \in \mathbb{R}, \quad m = 0..M \\ q_n \in \mathbb{R}, \quad n = 0..N \\ \Delta \in \mathbb{R}. \end{array} \right.$$

II.4 Initialisation optimisée

La méthode introduite par Papamarkos [Pap88] est une alternative au DCA pour trouver une solution au problème *minmax*. L'idée principale est de maximiser l'inverse de l'erreur L^∞ . Ceci induit un changement de variables par rapport à la formulation du DCA :

$$\varepsilon = 1/\Delta$$

$$\delta = \min_{i=1..I} \{Q(x_i)/\varepsilon\}$$

$$p'_m = \varepsilon p_m, \quad m = 0..M$$

$$q'_n = \varepsilon q_n, \quad n = 0..N,$$

ainsi que les nouveaux polynômes \bar{Q} et \bar{P} , tels que $R = \bar{P}/\bar{Q}$ avec :

$$\bar{Q}(x) = \sum_{n=0}^N q'_n b_n(x) \quad \text{et} \quad \bar{P}(x) = \sum_{n=0}^N p'_n b_n(x).$$

La seule différence dans notre contexte est que nous devons nous assurer que $R(x)$ sera tout le temps non-négatif. Avec $\varepsilon \geq 0$ nous devons modifier l'équation II.4 afin de

tenir compte de cette nouvelle contrainte :

$$\varepsilon P(x_i) = \bar{P}(x_i) \geq 0, \quad i = 1..I. \quad (\text{II.5})$$

L'approximation, non-optimale, du problème *minmax* est donc la solution du problème linéaire suivant :

maximiser δ sous les contraintes :

$$\left\{ \begin{array}{l} f_i \varepsilon + f_i \bar{Q}(x_i) - \bar{P}(x_i) \leq 1, \quad i = 1..I \\ -f_i \varepsilon - f_i \bar{Q}(x_i) + \bar{P}(x_i) \leq 1, \quad i = 1..I \\ \qquad \qquad \qquad -\bar{P}(x_i) \leq 0, \quad i = 1..I \\ \qquad \qquad \qquad -\varepsilon - \bar{Q}(x_i) + \delta \geq 0, \quad i = 1..I . \end{array} \right.$$

$$\left\{ \begin{array}{l} p'_m \in \mathbb{R}, \quad m = 0..M \\ q'_n \in \mathbb{R}, \quad n = 0..N \\ \delta \geq 0 \\ \varepsilon \geq 0. \end{array} \right.$$

Comme pour le DCA, dans le cas de la base monomiale, les mêmes simplifications peuvent être effectuées.

ANNEXE III

REPRÉSENTATION DIRECTIONNELLE : ÉTUDE DE BASES DE FONCTION SUR LA SPHÈRE UNITÉ

Dans cette annexe, nous présentons différentes bases de fonction pour approximer l'éclairement incident sur la sphère ou l'hémisphère. Nous rappelons (section III.1) la théorie mathématique liée à l'utilisation des bases de fonction pour approcher une grandeur radiométrique (section III.2) avant d'introduire différentes bases (section III.3).

III.1 Bases de fonction sur la sphère

Principe général

Une base de fonction sur un espace fonctionnel¹ permet de représenter une fonction à l'aide d'une combinaison linéaire des éléments de la base. Un élément de la base est alors appelé fonction de base.

Dans notre cas, l'espace auquel nous nous intéressons est la sphère et la fonction que nous cherchons à représenter est celle de l'éclairement incident. L'intérêt de construire une telle base est de pouvoir approximer un signal défini sur la sphère à l'aide d'un nombre choisi de coefficients. Le nombre de coefficients est directement relié au nombre d'éléments de la base. Leur calcul est effectué à l'aide d'une opération dite de «projection» tandis que l'opération inverse, l'approximation de la fonction originale à l'aide des coefficients et des fonctions de base, est appelée «reconstruction». Une propriété importante caractérisant l'erreur introduite par l'opération de reconstruction est que, lorsque la base est orthonormale la reconstruction est optimale au sens des moindres carrés.

Définitions

Soit $\boldsymbol{\omega} = (\omega_x, \omega_y, \omega_z)$ une direction sur la sphère unité. Soit $F = \{f_1(\boldsymbol{\omega}), f_1(\boldsymbol{\omega}), \dots, f_p(\boldsymbol{\omega})\}$ un ensemble regroupant p fonctions définies sur la sphère et

¹Un espace fonctionnel peut être un espace vectoriel, un espace topologique ou les deux à la fois.

à valeurs dans \mathbb{R} :

$$\forall i \in [1, p] \quad f_i(\boldsymbol{\omega}) : S \rightarrow \mathbb{R}.$$

où S représente la sphère unité et donc tout l'espace des directions.

Une manière de déterminer si F est une base ainsi que sa caractérisation (orthogonale, orthonormale, etc.) est de calculer la matrice de corrélation² M dont les éléments m_{ij} sont définis par :

$$m_{ij} = \int_S f_i(\boldsymbol{\omega}) f_j(\boldsymbol{\omega}) d\boldsymbol{\omega}. \quad (\text{III.1})$$

L'intégrale de l'équation (III.1) est la définition du produit scalaire de deux fonctions sur leur domaine de définition.

En fonction de M on peut caractériser F :

- Si M est inversible alors F est une base, sinon F est au mieux une famille génératrice.
- Si M est diagonale (mais n'est pas la matrice identité), alors M est une base orthogonale.
- Si M est la matrice identité, alors F est une base orthonormale.

Base duale

Lorsque M est inversible, on peut définir la base duale de F notée \hat{F} . Les éléments \hat{f}_k de la base duale sont définis à partir des éléments de F et de la matrice de corrélation :

$$\int_S f_i(\boldsymbol{\omega}) \hat{f}_k(\boldsymbol{\omega}) d\boldsymbol{\omega} = \begin{cases} 1 & \text{si } i = k \\ 0 & \text{sinon} \end{cases} \quad (\text{III.2})$$

et sont obtenus par combinaison linéaire des f_j :

$$\hat{f}_k(\boldsymbol{\omega}) = \sum_{j=1}^p m_{kj}^{-1} f_j(\boldsymbol{\omega}) \quad (\text{III.3})$$

²Cette matrice est parfois appelée matrice de Gram par référence au procédé d'orthogonalisation de Gram-Schmidt.

où m_{kj}^{-1} représente les éléments de la matrice inverse de M .

Projection

Soit $g(\boldsymbol{\omega})$ une fonction dont on cherche la projection sur S à l'aide d'une base F et de sa duale \hat{F} . Le résultat de la projection de g sur S est un ensemble $\hat{C} = \{c_1, \dots, c_p\}$ de coefficients donnés par :

$$\forall i \in [1, p], \quad c_i = \int_S \hat{f}_i(\boldsymbol{\omega}) g(\boldsymbol{\omega}) d\boldsymbol{\omega}. \quad (\text{III.4})$$

Notons que les c_i sont des grandeurs scalaires et que les fonctions f_i sont à valeurs dans \mathbb{R} .

Reconstruction

La reconstruction de la fonction $g(\boldsymbol{\omega})$, notée $g_r(\boldsymbol{\omega})$, se fait à partir des coefficients c_i et des fonctions de base de F :

$$g_r(\boldsymbol{\omega}) \approx \sum_{i=1}^p c_i f_i(\boldsymbol{\omega}). \quad (\text{III.5})$$

Notions d'orthogonalité et d'orthonormalité d'une base

Lorsque que F est une base orthogonale, la matrice de corrélation est forcément diagonale. Soient λ_i les éléments diagonaux de M . On peut alors établir de manière immédiate une relation entre les fonctions dans la base F et leurs correspondantes dans la base duale \hat{F} :

$$\forall i \in [1, p] \quad \lambda_i \in \mathbb{R}$$

$$\hat{f}_i(\boldsymbol{\omega}) = \frac{1}{\lambda_i} f_i(\boldsymbol{\omega}).$$

Dans le cas où la base F est orthonormale, la relation entre les f_i et les \hat{f}_i se simplifie :

$$\hat{f}_i(\boldsymbol{\omega}) = f_i(\boldsymbol{\omega}).$$

D'un point de vue pratique, il apparaît donc plus intéressant d'utiliser des bases or-

thonormales puisque, lors de la phase de reconstruction, les calculs sont réduits par rapport au cas général. Remarquons qu'une base orthogonale mais non orthonormale peut toujours être normalisée en posant comme nouvelles fonctions de base f'_i :

$$f'_i(\boldsymbol{\omega}) = \frac{f_i(\boldsymbol{\omega})}{\sqrt{\lambda_i}}.$$

III.2 Application à l'éclairage

En général, n'importe quel terme de l'équation du rendu étant défini sur la sphère peut être approximé à l'aide d'une base de fonction. Le but est toujours d'accélérer le rendu et/ou d'optimiser le stockage. En éclairage global, on cherche le plus souvent à représenter efficacement sur la sphère (ou l'hémisphère) la radiance incidente ou l'irradiance.

Projection de la radiance

En reprenant l'intégrale de l'équation, on peut calculer la projection de la radiance dans une base F contenant p éléments :

$$l_i = \int_{\Omega_n} \hat{f}_i(\boldsymbol{\omega}) L(x \leftarrow \boldsymbol{\omega}) d\boldsymbol{\omega} \quad (\text{III.6})$$

et sa reconstruction :

$$L(x \leftarrow \boldsymbol{\omega}) \approx \sum_{i=1}^p l_i f_i(\boldsymbol{\omega}). \quad (\text{III.7})$$

A l'aide de cette dernière équation, nous pouvons remplacer la radiance incidente dans l'équation du rendu afin d'obtenir :

$$\begin{aligned} L(x \rightarrow \boldsymbol{\omega}_o) &\approx \int_{\Omega_n} \left(\sum_{i=1}^p c_i f_i(\boldsymbol{\omega}) \right) \rho(\boldsymbol{\omega}_o, \boldsymbol{\omega}) (\mathbf{n} \cdot \boldsymbol{\omega}) d\boldsymbol{\omega} \\ L(x \rightarrow \boldsymbol{\omega}_o) &\approx \sum_{i=1}^p l_i \underbrace{\left(\int_{\Omega_n} f_i(\boldsymbol{\omega}) \rho(\boldsymbol{\omega}_o, \boldsymbol{\omega}) (\mathbf{n} \cdot \boldsymbol{\omega}) d\boldsymbol{\omega} \right)}_{T_i(\mathbf{n}, \boldsymbol{\omega}_o)} \end{aligned} \quad (\text{III.8})$$

où $\rho(\boldsymbol{\omega}_o, \boldsymbol{\omega})$ représente la BRDF au point x et où \mathbf{n} représente la normale en x . Le terme T ne dépend plus que de la BRDF au point x et ne dépend plus de l'éclairage incident. Les techniques à base de PRT reposent sur un pré-calcul et un stockage optimisé de T .

Projection de l'irradiance

Lorsque que la BRDF est lambertienne, il est plus intéressant de projeter l'irradiance $E(x, \mathbf{n})$ [RH01a] puisque la radiance réfléchi est directement fonction de l'irradiance :

$$L(x \rightarrow \boldsymbol{\omega}_o) = \frac{\rho_d}{\pi} E(x, \mathbf{n}).$$

Rappelons que l'irradiance au point x est donnée par :

$$E(x, \mathbf{n}) = \int_{\Omega_{\mathbf{n}}} L(x \leftarrow \boldsymbol{\omega}) (\mathbf{n} \cdot \boldsymbol{\omega}) d\boldsymbol{\omega}$$

où $\Omega_{\mathbf{n}}$ est l'hémisphère orienté selon \mathbf{n} . En remplaçant la radiance incidente $L(x \leftarrow \boldsymbol{\omega})$ par son approximation, nous obtenons :

$$E(x, \mathbf{n}) \approx \sum_{i=1} c_i \underbrace{\left(\int_{\Omega_{\mathbf{n}}} f_i(\boldsymbol{\omega}) (\mathbf{n} \cdot \boldsymbol{\omega}) d\boldsymbol{\omega} \right)}_{T_i(\mathbf{n})}. \quad (\text{III.9})$$

$T_i(\mathbf{n})$ peut être vu comme la convolution de la fonction $f_i(\boldsymbol{\omega})$, définie sur la sphère, par un filtre $h_{\mathbf{n}}(\boldsymbol{\omega})$ hémisphérique orienté selon \mathbf{n} tel que :

$$h_{\mathbf{n}}(\boldsymbol{\omega}) = \max(\mathbf{n} \cdot \boldsymbol{\omega}, 0).$$

Aussi bien dans le cas de la radiance que de l'irradiance, et selon la base F , T peut être dérivé de manière analytique ou encore pré-calculé dans un tableau. Enfin, bien que le terme de visibilité n'apparaisse pas de manière explicite, il ne faut pas perdre de vue qu'il est induit par l'angle solide différentiel.

Intégration de la BRDF

La formulation générale de la fonction $T(\mathbf{n}, \boldsymbol{\omega}_o)$ présentée à l'équation (III.8) montre que T est une fonction à 4 dimensions difficilement stockable d'un point de vue pratique. D'ailleurs les premiers travaux (e.g. [NRH03]) à base de PRT fixaient $\boldsymbol{\omega}_o$ afin de simplifier le problème. Cependant, les dernières avancées [WTL04, WTL06b] préfèrent utiliser la factorisation (reliée à celle de McCool *et al.* [MAA01]) où la BRDF est séparée comme le produit de deux fonctions bi-dimensionnelles :

$$\rho(\boldsymbol{\omega}, \boldsymbol{\omega}_o) = \sum_{k=1}^K \rho_k^1(\boldsymbol{\omega}) \rho_k^2(\boldsymbol{\omega}_o) \quad (\text{III.10})$$

où K est le nombre de termes utilisés pour décomposer la BRDF. Naturellement, plus K est élevé, plus l'approximation de la BRDF est précise. En utilisant l'équation (III.10) dans la formulation de $T(\mathbf{n}, \boldsymbol{\omega}_o)$, l'équation de la radiance réfléchie devient :

$$\begin{aligned} L(x \rightarrow \boldsymbol{\omega}_o) &\approx \sum_{i=1}^p c_i \left[\int_{\Omega_n} f_i(\boldsymbol{\omega}) \left(\sum_k^K \rho_k^1(\boldsymbol{\omega}) \rho_k^2(\boldsymbol{\omega}_o) \right) (\mathbf{n} \cdot \boldsymbol{\omega}) d\boldsymbol{\omega} \right] \\ L(x \rightarrow \boldsymbol{\omega}_o) &\approx \sum_{i=1}^p c_i \sum_k^K \rho_k^2(\boldsymbol{\omega}_o) \underbrace{\left(\int_{\Omega_n} f_i(\boldsymbol{\omega}) \rho_k^1(\boldsymbol{\omega}) (\mathbf{n} \cdot \boldsymbol{\omega}) d\boldsymbol{\omega} \right)}_{T_{i,k}(\mathbf{n})}. \end{aligned} \quad (\text{III.11})$$

III.3 Nouvelles bases de fonction

Dans cette section, nous présentons de nouvelles bases de fonction dont le but est de reconstruire la radiance incidente représentée par exemple par une carte d'environnement. Les nouvelles bases doivent aussi être capables de reconstruire la fonction constante sur la sphère de directions. Pour chaque base, nous donnons également la matrice M de corrélation associée.

Dans la suite de cette section, la fonction $\text{sign}(x) : \mathbb{R} \rightarrow \{-1, 1\}$ est définie par :

$$\text{sign}(x) = \begin{cases} -1 & \text{si } x < 0 \\ 1 & \text{sinon.} \end{cases}$$

Bases utilisant la fonction *sign*

constante par morceau

$$F_0 = \begin{cases} f_0(\boldsymbol{\omega}) = 1 \\ f_1(\boldsymbol{\omega}) = \text{sign}(\omega_x) \\ f_2(\boldsymbol{\omega}) = \text{sign}(\omega_y) \\ f_3(\boldsymbol{\omega}) = \text{sign}(\omega_z) \end{cases} \quad M_0 = \begin{pmatrix} 4\pi & 0 & 0 & 0 \\ 0 & 4\pi & 0 & 0 \\ 0 & 0 & 4\pi & 0 \\ 0 & 0 & 0 & 4\pi \end{pmatrix}$$

quadratique

$$F_1 = \begin{cases} f_0(\boldsymbol{\omega}) = 1 \\ f_1(\boldsymbol{\omega}) = \omega_x^2 \text{sign}(\omega_x) \\ f_2(\boldsymbol{\omega}) = \omega_y^2 \text{sign}(\omega_y) \\ f_3(\boldsymbol{\omega}) = \omega_z^2 \text{sign}(\omega_z) \end{cases} \quad M_1 = 4\pi \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{5} & 0 & 0 \\ 0 & 0 & \frac{1}{5} & 0 \\ 0 & 0 & 0 & \frac{1}{5} \end{pmatrix}$$

La base F_1 peut s'écrire sans utiliser la fonction *sign* :

$$F_1 = \begin{cases} f_0(\boldsymbol{\omega}) = 1 \\ f_1(\boldsymbol{\omega}) = \omega_x |\omega_x| \\ f_2(\boldsymbol{\omega}) = \omega_y |\omega_y| \\ f_3(\boldsymbol{\omega}) = \omega_z |\omega_z| \end{cases} .$$

On peut ajouter des termes bilinéaires et trinéaires en conservant son orthogonalité :

$$F_3 = \begin{cases} f_0(\boldsymbol{\omega}) = 1 \\ f_1(\boldsymbol{\omega}) = \omega_x |\omega_x| \\ f_2(\boldsymbol{\omega}) = \omega_y |\omega_y| \\ f_3(\boldsymbol{\omega}) = \omega_z |\omega_z| \\ f_4(\boldsymbol{\omega}) = \omega_x \omega_y \\ f_5(\boldsymbol{\omega}) = \omega_x \omega_z \\ f_6(\boldsymbol{\omega}) = \omega_y \omega_z \\ f_7(\boldsymbol{\omega}) = \omega_x \omega_y \omega_z \end{cases} \quad M_3 = 4\pi \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{5} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{5} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{5} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{15} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{15} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{15} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{105} \end{pmatrix} .$$

Bases polynomiales

Les harmoniques sphériques constituent certainement la base polynomiale sur la sphère la plus connue et la plus utilisée, notamment grâce à sa propriété d'orthonormalité. F_4 représente les éléments de la base d'ordre 2 des harmoniques sphériques :

$$F_4 = \begin{cases} f_0(\boldsymbol{\omega}) = 1 \\ f_1(\boldsymbol{\omega}) = \omega_x \\ f_2(\boldsymbol{\omega}) = \omega_y \\ f_3(\boldsymbol{\omega}) = \omega_z \end{cases} \quad M_4 = 4\pi \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{3} & 0 & 0 \\ 0 & 0 & \frac{1}{3} & 0 \\ 0 & 0 & 0 & \frac{1}{3} \end{pmatrix} .$$

Les harmoniques sphériques ont l'inconvénient de générer un effet de Gibbs (cf. [Gel97]) dès qu'on les utilise pour représenter une fonction discontinue. Il est intéressant d'essayer de dériver de nouvelles bases qui, tout en restant polynomiales, ne posséderaient pas ce problème. Si l'on souhaite conserver le fait que la base reste polynomiale, on doit nécessairement renoncer à l'orthogonalité de la base sous peine de retomber sur les harmoniques sphériques.

En utilisant le fait que $\boldsymbol{\omega} = (\omega_x, \omega_y, \omega_z)$ est une direction sur la sphère unité, on peut

construire une base quadratique quasi-orthogonale :

$$F_5 = \begin{cases} f_0(\boldsymbol{\omega}) = \omega_x \\ f_1(\boldsymbol{\omega}) = \omega_y \\ f_2(\boldsymbol{\omega}) = \omega_z \\ f_3(\boldsymbol{\omega}) = \omega_x \omega_y \\ f_4(\boldsymbol{\omega}) = \omega_x \omega_z \\ f_5(\boldsymbol{\omega}) = \omega_y \omega_z \\ f_6(\boldsymbol{\omega}) = (\omega_x)^2 \\ f_7(\boldsymbol{\omega}) = (\omega_y)^2 \\ f_8(\boldsymbol{\omega}) = (\omega_z)^2 \end{cases} \quad M_5 = 4\pi \begin{pmatrix} \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{15} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{15} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{15} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{5} & \frac{1}{15} & \frac{1}{15} \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{15} & \frac{1}{5} & \frac{1}{15} \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{15} & \frac{1}{15} & \frac{1}{5} \end{pmatrix} .$$

Puisque $\omega_x^2 + \omega_y^2 + \omega_z^2 = 1$, on peut toujours reconstruire la fonction constante sur la sphère. Bien que la base F_5 ne soit pas orthogonale, la quasi-diagonalité de la matrice M_5 montre qu'elle est quasi-orthogonale. Par conséquent, la base primale et la base duale sont très proches.

En appliquant le même principe que précédemment, on peut construire une base

non-orthogonale de degré 3 :

$$F_6 = \left\{ \begin{array}{l} f_0(\boldsymbol{\omega}) = \omega_x \omega_y \\ f_1(\boldsymbol{\omega}) = \omega_x \omega_z \\ f_2(\boldsymbol{\omega}) = \omega_y \omega_z \\ f_3(\boldsymbol{\omega}) = \omega_x^2 \\ f_4(\boldsymbol{\omega}) = \omega_y^2 \\ f_5(\boldsymbol{\omega}) = \omega_z^2 \\ f_6(\boldsymbol{\omega}) = \omega_x \omega_y^2 \\ f_7(\boldsymbol{\omega}) = \omega_x \omega_z^2 \\ f_8(\boldsymbol{\omega}) = \omega_y \omega_x^2 \\ f_9(\boldsymbol{\omega}) = \omega_y \omega_z^2 \\ f_{10}(\boldsymbol{\omega}) = \omega_z \omega_x^2 \\ f_{11}(\boldsymbol{\omega}) = \omega_z \omega_y^2 \\ f_{12}(\boldsymbol{\omega}) = \omega_x \omega_y \omega_z \\ f_{13}(\boldsymbol{\omega}) = \omega_x^3 \\ f_{14}(\boldsymbol{\omega}) = \omega_y^3 \\ f_{15}(\boldsymbol{\omega}) = \omega_z^3 \end{array} \right. \quad (\text{III.12})$$

Dans ce cas précis, nous avons utilisé les relations algébriques suivantes :

$$\omega_x = \omega_y^2 \omega_x + \omega_z^2 \omega_x + \omega_x^3 \quad \omega_y = \omega_x^2 \omega_y + \omega_z^2 \omega_y + \omega_y^3 \quad \omega_z = \omega_x^2 \omega_z + \omega_y^2 \omega_z + \omega_z^3 \quad (\text{III.13})$$

afin de retirer les fonctions f_0 , f_1 et f_2 de la base F_5 .

ANNEXE IV

DÉRIVATION DU JACOBIEN

Le but de cette annexe est de dériver la formulation du jacobien introduite au chapitre 6.

Le jacobien J est défini par $J = \frac{dl}{d\mathbf{l}}$ et il intervient dès qu'un changement de variable est effectué dans un calcul d'intégrales.

Dans le repère local $\{\mathbf{u}, \mathbf{v}, \mathbf{z}\}$ où \mathbf{u} et \mathbf{v} sont les directions des courbures principales exprimées en espace image (la caméra pointant vers $-\mathbf{z}$), on note les directions \mathbf{l} et \mathbf{l}' de la lumière de la façon suivante :

$$\mathbf{l} = \begin{pmatrix} l_u \\ l_v \\ l_z \end{pmatrix} = \begin{pmatrix} \cos(\phi) \sin(\theta) \\ \sin(\phi) \sin(\theta) \\ \cos(\theta) \end{pmatrix} \quad (\text{IV.1})$$

$$\mathbf{l}' = \begin{pmatrix} l'_u \\ l'_v \\ l'_z \end{pmatrix} = \begin{pmatrix} \cos(\phi') \sin(\theta') \\ \sin(\phi') \sin(\theta') \\ \cos(\theta') \end{pmatrix} \quad (\text{IV.2})$$

Dans cette paramétrisation $d\mathbf{l} = \sin \theta d\theta d\phi$.

En introduisant $\tau = \tan(\theta/2)$, on remarque que

$$d\tau = 2/(1 + \tau^2)d\theta$$

et que

$$\sin(\theta) = 2\tau/(1 + \tau^2).$$

Ce qui conduit à :

$$d\mathbf{l} = \frac{4\tau}{(1 + \tau^2)^2} d\tau d\phi.$$

La projection stéréographique introduit deux nouvelles variables :

$$a = l_u / (1 + l_z) = \cos(\phi) \tau$$

et

$$b = l_v / (1 + l_z) = \sin(\phi) \tau.$$

Puisque $a^2 + b^2 = \tau^2$ et que $da db = \tau d\tau d\phi$, cette nouvelle substitution nous permet d'obtenir :

$$d\mathbf{l} = \frac{4}{(1 + a^2 + b^2)^2} da db. \quad (\text{IV.3})$$

Dans l'espace stéréographique, le warping de la direction correspond simplement à une mise à l'échelle de a et b : $a' = \lambda_u a$ et $b' = \lambda_v b$. Ce qui donne :

$$d\mathbf{l} = \frac{4}{(1 + a'^2/\lambda_u^2 + b'^2/\lambda_v^2)^2} \frac{da' db'}{\lambda_u \lambda_v} \quad (\text{IV.4})$$

$$= \frac{4\lambda_u\lambda_v}{(\lambda_u^2\lambda_v^2 + \lambda_v^2a'^2 + \lambda_u^2b'^2)^2} da' db'. \quad (\text{IV.5})$$

Comme à l'instar de l'équation IV.3, on peut écrire :

$$d\mathbf{l}' = \frac{4}{(1 + a'^2 + b'^2)^2} da' db'. \quad (\text{IV.6})$$

En combinant ce résultat avec l'équation IV.5 on obtient alors :

$$d\mathbf{l} = \frac{\lambda_u^3\lambda_v^3(1 + a'^2 + b'^2)^2}{(\lambda_u^2\lambda_v^2 + \lambda_v^2a'^2 + \lambda_u^2b'^2)^2} d\mathbf{l}'. \quad (\text{IV.7})$$

a' et b' correspondent à la projection stéréographique de la direction \mathbf{l}' , $a' = l'_u / (1 + l'_z)$ et $b = l'_v / (1 + l'_z)$ et on obtient donc :

$$d\mathbf{l}' = \frac{\lambda_u^3\lambda_v^3((1 + l'_z)^2 + l'^2_u + l'^2_v)^2}{(\lambda_u^2\lambda_v^2(1 + l'_z)^2 + \lambda_v^2l'^2_u + \lambda_u^2l'^2_v)^2} d\mathbf{l}' = J d\mathbf{l}'. \quad (\text{IV.8})$$

Comme \mathbf{l}' est une direction on a : $1 - l_z'^2 = l_u'^2 + l_v'^2$, ce qui permet d'obtenir :

$$J = \frac{4\lambda_u^3 \lambda_v^3 (1 + \ell_z')^2}{(\lambda_u^2 \lambda_v^2 (1 + \ell_z')^2 + \lambda_v^2 l_u'^2 + \lambda_u^2 l_v'^2)^2}.$$

BIBLIOGRAPHIE

- [AFO05] O. Arikan, D.A. Forsyth, and J.F. O'Brien. Fast and Detailed Approximate Global Illumination by Irradiance Decomposition. In *Proc. SIGGRAPH 2005*, pages 1108–1114. ACM, 2005.
- [AG00] Dominique Apprato and Christian Gout. A result about scale transformation families in approximation : application to surface fitting from rapidly varying data. *Numerical Algorithms*, 23(2-3) :263–279, 2000.
- [AKFR07] Ben-Artzi Aner, Egan Kevin, Durand Fredo, and Ramamoorthi Ravi. A precomputed polynomial representation for interactive brdf editing with global illumination. *To appear in ACM Trans. Graph.*, 2007.
- [AP07] Michael Ashikhmin and Simon Premoze. Distribution-based BRDFs. Technical Report <http://www.cs.utah.edu/~premoze/dbrdf/>, University of Utah, 2007.
- [APS00] M. Ashikhmin, S. Premoze, and P. Shirley. A microfacet-based BRDF generator. In *ACM SIGGRAPH '00*, pages 65–74, 2000.
- [ARBJ03] Sameer Agarwal, Ravi Ramamoorthi, Serge Belongie, and Henrik Wann Jensen. Structured importance sampling of environment maps. *ACM Trans. Graph.*, 22(3) :605–612, 2003.
- [Arf05] G. Arfken. *Mathematical Methods for Physicists*. Academic Press, 2005.
- [Arv94] James R. Arvo. The irradiance Jacobian for partially occluded polyhedral sources. In *Proc. SIGGRAPH '94*, annual conference on Computer graphics and interactive techniques, pages 343–350. ACM, 1994.
- [AS00a] Michael Ashikhmin and Peter Shirley. An anisotropic phong brdf model. *J. Graph. Tools*, 5(2) :25–32, 2000.

- [AS00b] M. Ashikmin and P. Shirley. An anisotropic phong light reflection model, 2000.
- [AWB06] Ken-Ichi Anjyo, Shuhei Wemler, and William Baxter. Tweakable light and shade for cartoon animation. In *NPAR '06 : Proc. international symposium on Non-photorealistic animation and rendering*, pages 133–139. ACM Press, 2006.
- [BAOR06] Aner Ben-Artzi, Ryan Overbeck, and Ravi Ramamoorthi. Real-time BRDF editing in complex lighting. *ACM Trans. Graph.*, 25(3) :945–954, 2006.
- [Bar97] Ronen Barzel. Lighting controls for computer cinematography. *journal of graphics tools*, 2(1) :1–20, 1997.
- [BBS⁺09] Brian Budge, Tony Bernardin, Jeff A. Stuart, Shubhabrata Sengupta, Kenneth I. Joy, and John D. Owens. Out-of-core data management for path tracing on hybrid resources. *Comput. Graph. Forum*, 28(2) :385–396, 2009.
- [BGH05] David Burke, Abhijeet Ghosh, and Wolfgang Heidrich. Bidirectional importance sampling for direct illumination. In *Rendering Techniques*, pages 147–156, 2005.
- [Bli77] James F. Blinn. Models of light reflection for computer synthesized pictures. In *ACM SIGGRAPH '77*, pages 192–198, 1977.
- [BNN⁺98] P. Bekaert, L. Neumann, A. Neumann, M. Sbert, and Y.D. Willems. Hierarchical monte carlo radiosity. In *Proceedings of 9th Eurographics Workshop on Rendering*. Fifth Eurographics Workshop on Rendering, 1998.
- [CAM08] Petrik Clarberg and Tomas Akenine-Möller. Practical Product Importance Sampling for Direct Illumination. *Computer Graphics Forum (Proceedings of Eurographics 2008)*, 27(2) :681–690, 2008.

- [CB04] P.H. Christensen and D. Batali. An Irradiance Atlas for Global Illumination in Complex Production Scenes. In *Proc. Eurographics Symposium on Rendering*, pages 133–141, 2004.
- [CBP07] Luc Claustres, Loïc Barthe, and Mathias Paulin. Wavelet encoding of brdfs for real-time rendering. In *GI '07 : Proceedings of Graphics Interface 2007*, pages 169–176, New York, NY, USA, 2007. ACM Press.
- [CBS96] Benoît Crespín, Carole Blanc, and Christophe Schlick. Implicit sweep objects. *Computer Graphics Forum*, 15(3) :165–174, 1996.
- [CCWG88] Michael F. Cohen, Shenchang Eric Chen, John R. Wallace, and Donald P. Greenberg. A progressive refinement approach to fast radiosity image generation. In *SIGGRAPH '88 : Proc. 15th annual conference on Computer graphics and interactive techniques*, pages 75–84, 1988.
- [CGA] CGAL. Computational Geometry Algorithms Library.
- [CGL⁺08] Forrester Cole, Aleksey Golovinskiy, Alex Limpaecher, Heather Stoddart Barros, Adam Finkelstein, Thomas Funkhouser, and Szymon Rusinkiewicz. Where Do People Draw Lines? *ACM Trans. Graph. (Proc. SIGGRAPH 2008)*, 27(3) :1–11, 2008.
- [Cha05] Chaosgroup. The chaos group. Internet Ressource, 2005 ?
- [CHCH06] Nathan A. Carr, Jared Hoberock, Keenan Crane, and John C. Hart. Fast gpu ray tracing of dynamic meshes using geometry images. In *GI '06 : Proceedings of Graphics Interface 2006*, pages 203–209, Toronto, Ont., Canada, Canada, 2006. Canadian Information Processing Society.
- [Chr99] P.H. Christensen. Faster global photon map global illumination. *Journal of Graphics Tools*, 4(3) :1–10, 1999.
- [CMS87a] Brian Cabral, Nelson Max, and Rebecca Springmeyer. Bidirectional reflection functions from surface bump maps. In *SIGGRAPH '87 : Procee-*

- dings of the 14th annual conference on Computer graphics and interactive techniques*, pages 273–281, New York, NY, USA, 1987. ACM Press.
- [CMS87b] Brian Cabral, Nelson Max, and Rebecca Springmeyer. Bidirectional reflection functions from surface bump maps. In *SIGGRAPH '87 : Proc. 14th annual conference on Computer graphics and interactive techniques*, pages 273–281, 1987.
- [Cor] The cornel reflectance database.
- [CPK06] M. Colbert, S. Pattanaik, and J. Krivánek. BRDF-Shop : Creating Physically Correct Bidirectional Reflectance Distribution Functions. *IEEE Computer Graphics and Applications*, 26(1) :30–36, 2006.
- [CPWAP08] Ewen Cheslack-Postava, Rui Wang, Oskar Akerlund, and Fabio Pellacini. Fast, realistic lighting and material design using nonlinear cut approximation. *ACM Trans. Graph.*, 27(5) :1–10, 2008.
- [CSF99] António Cardoso Costa, António Augusto Sousa, and Fernando Nunes Ferreira. Lighting design : A goal based approach using optimisation. In *Rendering Techniques*, pages 317–328, 1999.
- [CST05] Paolo Cignoni, Roberto Scopigno, and Marco Tarini. A simple Normal Enhancement technique for Interactive Non-photorealistic Renderings. *Comp. & Graph.*, 29(1) :125–133, 2005.
- [CT82a] R. L. Cook and K. E. Torrance. A reflectance model for computer graphics. *ACM Trans. Graph.*, 1(1) :7–24, 1982.
- [CT82b] Robert L. Cook and Kenneth E. Torrance. A Reflectance Model for Computer Graphics. *ACM Trans. Graph.*, 1(1) :7–24, 1982.
- [CWH93] Michael F. Cohen, John Wallace, and Pat Hanrahan. *Radiosity and realistic image synthesis*. Academic Press Professional, Inc., San Diego, CA, USA, 1993.

- [CZS96] Ken Chiu, Kurt Zimmerman, and Peter Shirley. The light volume : an aid to rendering complex environments. In *Proceedings of the eurographics workshop on Rendering techniques '96*, pages 1–ff., London, UK, 1996. Springer-Verlag.
- [DBB06] Philip Dutré, Kavita Bala, and Philippe Bekaert. *Advanced Global Illumination*. A. K. Peters, Ltd., 2006.
- [DBMS02] Kirill Dmitriev, Stefan Brabec, Karol Myszkowski, and Hans-Peter Seidel. Interactive global illumination using selective photon tracing. In *EGRW '02 : Proceedings of the 13th Eurographics workshop on Rendering*, pages 25–36, Aire-la-Ville, Switzerland, Switzerland, 2002. Eurographics Association.
- [DD00] Frédo Durand and Julie Dorsey. Interactive tone mapping. 2000. Held in Brno, Czech Republic.
- [DFRS03] Doug DeCarlo, Adam Finkelstein, Szymon Rusinkiewicz, and Anthony Santella. Suggestive Contours for Conveying Shape. *ACM Trans. Graph. (Proc. SIGGRAPH 2003)*, 22(3) :848–855, July 2003.
- [DH05] Julie Dorsey and Pat Hanrahan. Modeling and rendering of metallic patinas. In *SIGGRAPH '05 : ACM SIGGRAPH 2005 Courses*, page 2, New York, NY, USA, 2005. ACM.
- [DKTS07] Zhao Dong, Jan Kautz, Christian Theobalt, and Hans-Peter Seidel. Interactive global illumination using implicit visibility. In *PG '07 : Proceedings of the 15th Pacific Conference on Computer Graphics and Applications*, pages 77–86, Washington, DC, USA, 2007. IEEE Computer Society.
- [DR08] Julie Dorsey and Holly Rushmeier. Light and materials in virtual cities. In *SIGGRAPH 2008 Course Notes*, 2008.

- [DS05] Carsten Dachsbacher and Marc Stamminger. Reflective shadow maps. In *I3D '05 : Proceedings of the 2005 symposium on Interactive 3D graphics and games*, pages 203–231, New York, NY, USA, 2005. ACM.
- [DS06] Carsten Dachsbacher and Marc Stamminger. Splatting indirect illumination. In *I3D '06 : Proceedings of the 2006 symposium on Interactive 3D graphics and games*, pages 93–100, New York, NY, USA, 2006. ACM.
- [DSDD07] Carsten Dachsbacher, Marc Stamminger, George Drettakis, and Frédo Durrant. Implicit visibility and antiradiance for interactive global illumination. *ACM Trans. Graph.*, 26(3) :61, 2007.
- [DvGNK99] Kristin J. Dana, Bram van Ginneken, Shree K. Nayar, and Jan J. Koenderink. Reflectance and texture of real-world surfaces. *ACM Trans. Graph.*, 18(1) :1–34, 1999.
- [DWS⁺88] Michael Deering, Stephanie Winner, Bic Schediwy, Chris Duffy, and Neil Hunt. The triangle processor and normal vector shader : a vlsi system for high performance graphics. *SIGGRAPH Comput. Graph.*, 22(4) :21–30, 1988.
- [EBJ⁺06] Dave Edwards, Solomon Boulos, Jared Johnson, Peter Shirley, Michael Ashikhmin, Michael Stark, and Chris Wyman. The halfway vector disk for BRDF modeling. *ACM Trans. Graph.*, 25(1) :1–18, 2006.
- [FCGH08] J. Filip, M.J. Chantler, P.R. Green, and M. Haindl. A psychophysically validated metric for bidirectional texture data reduction. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2008)*, 27(5), December 2008.
- [FH09] Jiří Filip and Michal Haindl. Bidirectional texture function modeling : A state of the art survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 99(1), 2009.

- [FTA04] Roland W. Fleming, Antonio Torralba, and Edward H. Adelson. Specular reflections and the perception of shape. *J. Vis.*, 4(9) :798–820, 9 2004.
- [FTA09] Roland W. Fleming, Antonio Torralba, and Edward H. Adelson. *Three dimensional shape perception*, chapter Shape from sheen. Springer Verlag, 2009. to appear.
- [GCG⁺05] Johannes Günther, Tongbo Chen, Michael Goesele, Ingo Wald, and Hans-Peter Seidel. Efficient acquisition and realistic rendering of car paint. In Günther Greiner, Joachim Hornegger, Heinrich Niemann, and Marc Stamminger, editors, *Vision, Modeling, and Visualization 2005 (VMV'05)*, pages 487–494, Erlangen, Germany, November 2005. Aka.
- [GDW00] Xavier Granier, George Drettakis, and Bruce Walter. Fast global illumination including specular effects. In B. Péroche and H. Rushmeier, editors, *Proc. Eurographics Workshop on Rendering 2000*, pages 47 – 59. Eurographics, 2000.
- [Gel97] Anne Gelb. The resolution of the gibbs phenomenon for spherical harmonics. *Math. Comput.*, 66(218) :699–717, 1997.
- [GGSC98] Amy Gooch, Bruce Gooch, Peter Shirley, and Elaine Cohen. A Non-Photorealistic Lighting Model For Automatic Technical Illustration. In *Proc. ACM SIGGRAPH '98*, pages 447–452. ACM, 1998.
- [GH00] Reid Gershbein and Pat Hanrahan. A fast relighting engine for interactive cinematic lighting design. In *SIGGRAPH '00 : Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 353–358, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co.
- [GH03] Xavier Granier and Wolfgang Heidrich. A simple layered rgb brdf model. *Graph. Models*, 65(4) :171–184, 2003.

- [GIM] GIMP. GNU Image Manipulation Program.
- [GKD07] Paul Green, Jan Kautz, and Frédo Durand. Efficient reflectance and visibility approximations for environment map rendering. *Comput. Graph. Forum*, 26(3) :495–502, 2007.
- [GLP] GLPK. GNU Linear Programming Kit.
- [GSHG98] G. Greger, P. Shirley, P.M. Hubbard, and D.P. Greenberg. The irradiance volume. *IEEE Computer Graphics and Applications*, 18(2) :32–43, 1998.
- [GTGB84] Cindy M. Goral, Kenneth E. Torrance, Donald P. Greenberg, and Bennett Battaile. Modeling the interaction of light between diffuse surfaces. In *SIGGRAPH '84 : Proc. 11th annual conference on Computer graphics and interactive techniques*, pages 213–222, 1984.
- [GTHL99] André Guéziec, Gabriel Taubin, Bill Horn, and Francis Lazarus. A framework for streaming geometry in vrml. *IEEE Comput. Graph. Appl.*, 19(2) :68–78, 1999.
- [GTS⁺97] Donald P. Greenberg, Kenneth E. Torrance, Peter Shirley, James Arvo, Eric Lafortune, James A. Ferwerda, Bruce Walter, Ben Trumbore, Sumanta Patanaik, and Sing-Choong Foo. A framework for realistic image synthesis. In *SIGGRAPH '97 : Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 477–494, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co.
- [GVH07] Todd Goodwin, Ian Vollick, and Aaron Hertzmann. Isophote distance : a shading approach to artistic stroke thickness. In *NPAR '07 : Proc. international symposium on Non-photorealistic animation and rendering*, pages 53–62. ACM, 2007.
- [HBHS05] V. Havran, J. Bittner, R. Herzog, and H.-P. Seidel. Ray Maps for Global Illumination. In *Proc. Eurographics Symposium on Rendering 2005*, pages 43–54, 2005.

- [HHP⁺92] Xiao D. He, Patrick O. Heynen, Richard L. Phillips, Kenneth E. Torrance, David H. Salesin, and Donald P. Greenberg. A fast and accurate light reflection model. *SIGGRAPH Comput. Graph.*, 26(2) :253–254, 1992.
- [HKY⁺99] H. Hirayama, K. Kaneda, H. Yamashita, Y. Yamaji, and Y. Monden. Visualization of optical phenomena caused by multilayer films with complex refractive indices. *Computer Graphics and Applications, Pacific Conference on*, 0 :128, 1999.
- [HKYM01] Hideki Hirayama, Kazufumi Kaneda, Hideo Yamashita, and Yoshimi Monden. An accurate illumination model for objects coated with multilayer films. *Computers & Graphics*, 25(3) :391–400, 2001.
- [Hop96] Hugues Hoppe. Progressive meshes. In *Proceedings of ACM SIGGRAPH 1996*, pages 99–108. ACM, 1996.
- [HPB06] Miloš Hašan, Fabio Pellacini, and Kavita Bala. Direct-to-indirect transfer for cinematic relighting. *ACM Trans. Graph.*, 25(3) :1089–1097, 2006.
- [HPB07] Miloš Hašan, Fabio Pellacini, and Kavita Bala. Matrix row-column sampling for the many-light problem. *ACM Trans. Graph.*, 26(3) :26, 2007.
- [HSA91] P. Hanrahan, D. Salzman, and L. Aupperle. A rapid hierarchical radiosity algorithm. In *Computer Graphics Proceedings*, volume 25, pages 197–206, 1991.
- [HTSG91] Xiao D. He, Kenneth E. Torrance, François X. Sillion, and Donald P. Greenberg. A comprehensive physical model for light reflection. In *ACM SIGGRAPH '91*, pages 175–186, 1991.
- [HYK⁺00] H. Hirayama, Y. Yamaji, K. Kaneda, H. Yamashita, and Y. Monden. Rendering iridescent colors appearing on natural objects. *Computer Graphics and Applications, Pacific Conference on*, 0 :15, 2000.

- [IMT99] T. Igarashi, S. Matsuoka, and H. Tanaka. Teddy : a sketching interface for 3d freeform design. In *Proc. ACM SIGGRAPH '99*, pages 409–416, 1999.
- [Jen96] Henrik Wann Jensen. Global illumination using photon maps. In *Proceedings of the eurographics workshop on Rendering techniques '96*, pages 21–30, London, UK, 1996. Springer-Verlag.
- [Jen01] Henrik Wann Jensen. *Realistic Image Synthesis using Photon Mapping*. A.K. Peters, 2001.
- [Kaj85] James T. Kajiya. Anisotropic reflection models. In *SIGGRAPH '85 : Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, pages 15–21, New York, NY, USA, 1985. ACM Press.
- [Kaj86] J.T. Kajiya. The rendering equation. In *Proc. SIGGRAPH'86*, pages 143–150, 1986.
- [KAMJ05] Anders Wang Kristensen, Tomas Akenine-Möller, and Henrik Wann Jensen. Precomputed local radiance transfer for real-time lighting design. In *SIGGRAPH '05 : ACM SIGGRAPH 2005 Papers*, pages 1208–1215, New York, NY, USA, 2005. ACM Press.
- [KBPv06] J. Křivánek, K. Bouatouch, S.N. Pattanaik, and J. Žára. Making radiance and irradiance caching practical : Adaptive caching and neighbor clamping. In *Proc. Eurographics Symposium on Rendering 2006*, pages 127–138, 2006.
- [KBPZ06] J. Křivánek, Kadi Bouatouch, Sumanta Pattanaik, and Jirí Zára. Making radiance and irradiance caching practical : Adaptive caching and neighbor clamping. *Symposium on Rendering*, 17 :127–138, 2006.
- [KC08] Jaroslav Křivánek and Mark Colbert. Real-time shading with filtered importance sampling. In *Computer Graphics Forum (Proc. of Eurographics Symposium on Rendering*, volume 27, 2008.

- [KDS96] Jan J. Koenderink, Andrea J. van Doorn, and Marigo Stavridi. Bidirectional Reflection Distribution Function Expressed in Terms of Surface Scattering Modes. In *Proc. European Conference on Computer Vision-Volume II*, 1996.
- [Kel97] Alexander Keller. Instant radiosity. In *SIGGRAPH '97 : Proc. 24th annual conference on Computer graphics and interactive techniques*, pages 49–56, 1997.
- [KGPB05] J. Křivánek, P. Gautron, S. Pattanaik, and K. Bouatouch. Radiance caching for efficient global illumination computation. *IEEE Trans. Visualization and Computer Graphics*, 11(5) :550–561, 2005.
- [KM99] Jan Kautz and Michael D. McCool. Interactive rendering with arbitrary brdfs using separable approximations. In *SIGGRAPH '99 : ACM SIGGRAPH 99 Conference abstracts and applications*, page 253, 1999.
- [KM00] Jan Kautz and Michael D. McCool. Approximation of glossy reflection with prefiltered environment maps. In *Graphics Interface*, pages 119–126, 2000.
- [KPC93] John K. Kawai, James S. Painter, and Michael F. Cohen. Radiotimization : goal based rendering. In *SIGGRAPH '93 : Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 147–154, New York, NY, USA, 1993. ACM Press.
- [KSL05] Jan Kautz, Peter-Pike Sloan, and Jaakko Lehtinen. Precomputed radiance transfer : theory and practice. In *SIGGRAPH '05 : ACM SIGGRAPH 2005 Courses*, page 1, New York, NY, USA, 2005. ACM.
- [KST08] Michael Kolomenkin, Ilan Shimshoni, and Ayellet Tal. Demarcating Curves for Shape Illustration. *ACM Trans. Graph. (Proc. SIGGRAPH Asia 2008)*, 27(5) :1–9, 2008.

- [KWTM03] G Kindlmann, R Whitaker, T Tasdizen, and T Möller. Curvature-Based Transfer Functions for Direct Volume Rendering : Methods and Applications. In *Proc. IEEE Visualization 2003*, pages 513–520, October 2003.
- [LBAD⁺06] Jason Lawrence, Aner Ben-Artzi, Christopher DeCoro, Wojciech Matusik, Hanspeter Pfister, Ravi Ramamoorthi, and Szymon Rusinkiewicz. Inverse shade trees for non-parametric material representation and editing. *ACM Trans. Graph.*, 25(3) :735–745, 2006.
- [LBJS07] Julien Lacoste, Tamy Boubekur, Bruno Jobard, and Christophe Schlick. Appearance preserving octree-textures. In *GRAPHITE '07 : Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia*, pages 87–93, New York, NY, USA, 2007. ACM.
- [Leh07] Jaakko Lehtinen. A framework for precomputed and captured light transport. *ACM Trans. Graph.*, 26(4) :13, 2007.
- [LFTG97] Eric P. F. Lafortune, Sing-Choong Foo, Kenneth E. Torrance, and Donald P. Greenberg. Non-linear approximation of reflectance functions. In *ACM SIGGRAPH '97*, pages 117–126, 1997.
- [LFTW05] Hongsong Li, Sing Choong Foo, Kenneth E. Torrance, and Stephen H. Westin. Automated three-axis gonioreflectometer for computer graphics applications. In A. Duparre, B. Singh, and Z-H Gu, editors, *Advanced Characterization Techniques for Optics, Semiconductors, and Nanotechnologies II*, volume Proceedings of SPIE Vol. 5878, pages 5878–29. SPIE, SPIE, Bellingham, WA, July 2005.
- [LH06] Sylvain Lefebvre and Hugues Hoppe. Perfect spatial hashing. In *SIGGRAPH '06 : ACM SIGGRAPH 2006 Papers*, pages 579–588, New York, NY, USA, 2006. ACM.

- [LHV06] Chang Ha Lee, Xuejun Hao, and Amitabh Varshney. Geometry-dependent lighting. *IEEE Transactions on Visualization and Computer Graphics*, 12(2) :197–207, 2006.
- [LMLH07] Yunjin Lee, Lee Markosian, Seungyong Lee, and John F. Hughes. Line drawings via abstracted shading. *ACM Trans. Graph.*, 26(3) :18, 2007.
- [LRR04] Jason Lawrence, Szymon Rusinkiewicz, and Ravi Ramamoorthi. Efficient BRDF importance sampling using a factored representation. *ACM Trans. Graph.*, 23(3), 2004.
- [LSK⁺07] Samuli Laine, Hannu Saransaari, Janne Kontkanen, Jaakko Lehtinen, and Timo Aila. Incremental Instant Radiosity for Real-Time Indirect Illumination. In *Proc. EUROGRAPHICS Symposium on Rendering 2007*, pages 277–286, 2007.
- [LSSS04] Xinguo Liu, Peter-Pike Sloan, Heung Y. Shum, and John Snyder. All-Frequency Precomputed Radiance Transfer for Glossy Objects. In *Proc. EUROGRAPHICS Symposium on Rendering 2004*, pages 337–344. EUROGRAPHICS, 2004.
- [LW93] E.P. Lafortune and Y.D. Willems. Bi-directional path tracing. In *Proc. Third International Conference on Computational Graphics and Visualization Techniques (Compugraphics '93)*, pages 145–153, 1993.
- [LW94] Eric P. Lafortune and Yves D. Willems. Using the Modified Phong BRDF for Physically Based Rendering. Technical Report CW197, Katholieke Universiteit Leuven, Belgium, 1994.
- [LW95] Eric P. Lafortune and Yves D. Willems. A 5d tree to reduce the variance of monte carlo ray tracing. In *Rendering Techniques*, pages 11–20, 1995.
- [LZK⁺07] Jaakko Lehtinen, Matthias Zwicker, Janne Kontkanen, Emmanuel Turquin, François Sillion, and Timo Aila. Meshless finite elements for hierarchical

- global illumination. Technical Report TML-B7, Helsinki University of Technology, may 2007.
- [LZT⁺08] Jaakko Lehtinen, Matthias Zwicker, Emmanuel Turquin, Janne Kontkanen, Frédo Durand, François Sillion, and Timo Aila. A meshless hierarchical representation for light transport. *ACM Trans. Graph.*, 27(3), 2008.
- [MAA01] Michael D. McCool, Jason Ang, and Anis Ahmad. Homomorphic factorization of BRDFs for high-performance rendering. In *ACM SIGGRAPH '01 : Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 171–178. ACM Press, 2001.
- [McA02] David K. McAllister. *A generalized Representation of Surface Appearance*. PhD thesis, University of North Carolina at Chapel Hill, 2002.
- [Mel98] Stan Melax. A simple, fast and effective polygon reduction algorithm. *Game Developer Magazine*, pages 209–216, November 1998.
- [Mil94] Gavin Miller. Efficient Algorithms for Local and Global Accessibility Shading . In *Proc. ACM SIGGRAPH '94*, pages 319–326. ACM, 1994.
- [ML09] Morgan McGuire and David Luebke. Hardware-accelerated global illumination by image space photon mapping. In *Proceedings of the 2009 ACM SIGGRAPH/EuroGraphics conference on High Performance Graphics*, New York, NY, USA, August 2009. ACM.
- [MMS⁺06] G. Müller, J. Meseth, M. Sattler, R. Sarlette, and R. Klein. Acquisition, synthesis, and rendering of bidirectional texture functions. *Computer Graphics Forum*, 24(1) :83–109, 2006.
- [MPBM03a] W. Matusik, H. Pfister, M. Brand, and L. McMillan. A Data-Driven Reflectance Model. *ACM Transactions on Graphics*, 22(3) :759–769, July 2003.

- [MPBM03b] Wojciech Matusik, Hanspeter Pfister, Matthew Brand, and Leonard McMillan. Efficient isotropic brdf measurement. In *Proc. Eurographics workshop on Rendering*, pages 241–247. Eurographics, 2003.
- [MTR08] Dhruv Mahajan, Yu-Ting Tseng, and Ravi Ramamoorthi. An Analysis of the In-Out BRDF Factorization for View-Dependent Relighting. *Comp. Graph. Forum*, 27(4), 2008.
- [MWAM05] Stephen R. Marschner, Stephen H. Westin, Adam Arbree, and Jonathan T. Moon. Measuring and modeling the appearance of finished wood. *ACM Trans. Graph.*, 24(3) :727–734, 2005.
- [MWL⁺99] Stephen R. Marschner, Stephen H. Westin, Eric P. F. Lafortune, Kenneth E. Torrance, and Donald P. Greenberg. Image-based brdf measurement including human skin. In *Eurographics Rendering Workshop 1999*, 1999.
- [ND04] Marc Nienhaus and Jürgen Döllner. Blueprints : illustrating architecture and technical parts using hardware-accelerated non-photorealistic rendering. In *Graphics Interface (GI'04)*, pages 49–56. Canadian Human-Computer Communications Society, 2004.
- [NDM05] A. Ngan, F. Durand, and W. Matusik. Experimental Analysis of BRDF Models. In *Proc. Eurographics Symposium on Rendering*, pages 117–226, 2005.
- [NFKP94] Laszlo Neumann, Martin Feda, Manfred Kopp, and Werner Purgathofer. A new stochastic radiosity method for highly complex scenes. In *Proceedings of Fifth Eurographics Workshop on Rendering*, pages June 1994. TALK : W. Purgathofer 14.6.1994.
- [NN89] L. Neumann and A. Neumann. Photosimulation interreflection with arbitrary reflection models and illumination. *Comput. Graph. Forum*, 8(1) :21–34, 1989.

- [NNB⁺96] Attila Neumann, László Neumann, Philippe Bekaert, Yves D. Willems, and Werner Purgathofer. Importance-driven stochastic ray radiosity. In *Proceedings of the eurographics workshop on Rendering techniques '96*, pages 111–ff., London, UK, 1996. Springer-Verlag.
- [NRH⁺77] F. E. Nicodemus, J. C. Richmond, J. J. Hsia, I. W. Ginsberg, and T. Limperis. *Geometrical Considerations and Nomenclature for Reflectance*. National Bureau of Standards, 1977.
- [NRH03] Ren Ng, Ravi Ramamoorthi, and Pat Hanrahan. All-frequency shadows using non-linear wavelet lighting approximation. *ACM Trans. Graph.*, 22(3) :376–381, 2003.
- [NRH04] Ren Ng, Ravi Ramamoorthi, and Pat Hanrahan. Triple product wavelet integrals for all-frequency relighting. *ACM Trans. Graph.*, 23(3) :477–487, 2004.
- [OKP⁺08] Juraj Obert, Jaroslav Krivánek, Fabio Pellacini, Daniel Sýkora, and Sumanta N. Pattanaik. icheat : A representation for artistic control of indirect cinematic lighting. *Comput. Graph. Forum*, 27(4) :1217–1223, 2008.
- [OMSI07] Makoto Okabe, Yasuyuki Matsushita, Li Shen, and Takeo Igarashi. Illumination brush : Interactive design of all-frequency lighting. In *Proc. Pacific Conference on Computer Graphics and Applications*, pages 171–180. IEEE Computer Society, 2007.
- [ON94] Michael Oren and Shree K. Nayar. Generalization of lambert’s reflectance model. In *SIGGRAPH '94 : Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 239–246, New York, NY, USA, 1994. ACM Press.
- [Pap88] N. Papamarkos. On the optimum approximation of real rational functions via linear programming. *Applied Mathematics and Computation*, 26, 1988.

- [PBMF07] Fabio Pellacini, Frank Battaglia, R. Keith Morley, and Adam Finkelstein. Lighting with paint. *ACM Trans. Graph.*, 26(2) :9, e 07.
- [PBR71] M. J. D. Powell, I. Barrodale¹, and F. D. K. Robert. The differential correction algorithm for rational L^∞ approximation. In *Lect. Notes in Math.*, volume 228, 1971.
- [PDC⁺05] Timothy J. Purcell, Craig Donner, Mike Cammarano, Henrik Wann Jensen, and Pat Hanrahan. Photon mapping on programmable graphics hardware. In *SIGGRAPH '05 : ACM SIGGRAPH 2005 Courses*, page 258, New York, NY, USA, 2005. ACM.
- [PF90] Pierre Poulin and Alain Fournier. A model for anisotropic reflection. *Computer Graphics*, 24(4) :273–282, August 1990.
- [PF92] Pierre Poulin and Alain Fournier. Lights from highlights and shadows. *Computer Graphics*, 25(2) :31–38, March 1992.
- [PF95] P. Poulin and A. Fournier. Painting Surface Characteristics. In *Proc. Eurographics Workshop on Rendering*, pages 160–169. Springer, June 1995.
- [PG04] Matt Pharr and Simon Green. *GPU Gems*, chapter Ambient Occlusion. Addison-Wesley, 2004.
- [PGSP08] Romain Pacanowski, Xavier Granier, Christophe Schlick, and Pierre Poulin. Sketch and paint-based interface for highlight modeling. In *EUROGRAPHICS Workshop on Sketch-Based Interfaces and Modeling*. Eurographics, jun 2008.
- [PGSP09] Romain Pacanowski, Xavier Granier, Christophe Schlick, and Pierre Poulin. Volumetric Vector-Based Representation for Indirect Illumination Caching. Research Report RR-6983, INRIA, 2009.
- [PGSS07] Stefan Popov, Johannes Günther, Hans-Peter Seidel, and Philipp Slusallek. Stackless kd-tree traversal for high performance GPU ray tracing. *Com-*

- puter Graphics Forum*, 26(3) :415–424, September 2007. (Proceedings of Eurographics).
- [PH04] Matt Pharr and Greg Humphreys. *Physically Based Rendering : From Theory to Implementation*. Morgan Kaufmann, 2004.
- [Pho75] Bui Tuong Phong. Illumination for computer generated pictures. *Commun. ACM*, 18(6) :311–317, 1975.
- [PPD98] Eric Paquette, Pierre Poulin, and George Drettakis. A light hierarchy for fast rendering of scenes with many lights. In M. Gobel and F. Nunes Ferreira, editors, *Computer Graphics Forum (EuroGraphics '98 Conference Proceedings)*, pages 63–74. EuroGraphics, September 1998. held in Lisbon, Portugal, 02-04 September 1998.
- [PRG⁺08] Romain Pacanowski, Mickaël Raynaud, Xavier Granier, Patrick Reuter, Christophe Schlick, and Pierre Poulin. Efficient Streaming of 3D Scenes with Complex Geometry and Complex Lighting. In *Web3D '08 : Proc. International Symposium on 3D Web Technology*, pages 11–17. ACM, 2008.
- [PRJ97] Pierre Poulin, Karim Ratib, and Marco Jacques. Sketching shadows and highlights to position lights. In *Proceedings of Computer Graphics International 97*, pages 56–63. IEEE Computer Society, June 1997.
- [PRL⁺08] Romain Pacanowski, Mickaël Raynaud, Julien Lacoste, Xavier Granier, Patrick Reuter, Christophe Schlick, and Pierre Poulin. Compact Structures for Interactive Global Illumination on Large Cultural Objects. *International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST 2008) : Shorts and Projects*, 2008.
- [PVL⁺05] Fabio Pellacini, Kiril Vidimče, Aaron Lefohn, Alex Mohr, Mark Leone, and John Warren. Lpics : a hybrid hardware-accelerated relighting engine for computer cinematography. In *SIGGRAPH '05 : ACM SIGGRAPH 2005 Papers*, pages 464–470, New York, NY, USA, 2005. ACM Press.

- [PWL⁺07] Minghao Pan, Rui Wang, Xinguo Liu, Qunsheng Peng, and Hujun Bao. Precomputed Radiance Transfer Field for Rendering Interreflections in Dynamic Scenes. *Comp. Graph. Forum (Proc. EUROGRAPHICS 2007)*, 26(3) :485–493, 2007.
- [RBD06] Szymon Rusinkiewicz, Michael Burns, and Doug DeCarlo. Exaggerated Shading for Depicting Shape and Detail. *ACM Trans. Graph. (Proc. SIGGRAPH 2006)*, 25(3) :1199–1205, 2006.
- [RGK⁺08] T. Ritschel, T. Grosch, M. H. Kim, H.-P. Seidel, C. Dachsbacher, and J. Kautz. Imperfect shadow maps for efficient computation of indirect illumination. *ACM Trans. Graph.*, 27(5) :1–8, 2008.
- [RH01a] Ravi Ramamoorthi and Pat Hanrahan. An efficient representation for irradiance environment maps. In *SIGGRAPH '01 : Proc. 28th annual conference on Computer graphics and interactive techniques*, pages 497–500, 2001.
- [RH01b] Ravi Ramamoorthi and Pat Hanrahan. A signal-processing framework for inverse rendering. In *SIGGRAPH '01 : Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 117–128, New York, NY, USA, 2001. ACM Press.
- [RMB07] Ravi Ramamoorthi, Dhruv Mahajan, and Peter Belhumeur. A first-order analysis of lighting, shading, and shadows. *ACM Trans. Graph.*, 26(1) :2, 2007.
- [RSI⁺08] Tobias Ritschel, Kaleigh Smith, Matthias Ihrke, Thorsten Grosch, Karol Myszkowski, and Hans-Peter Seidel. 3D Unsharp Masking for Scene Coherent Enhancement. *ACM Trans. Graph. (Proc. SIGGRAPH 2008)*, 27(3) :1–8, 2008.
- [Rus98] Szymon Rusinkiewicz. A New Change of Variables for Efficient BRDF

- Representation. In *Proc. Eurographics Rendering Workshop*, pages 11–22. Springer, 1998.
- [SA05] M.M. Stark and B. Arvo, J. and Smits. Barycentric parameterizations for isotropic brdfs. volume 11–2, pages 126–138, 2005.
- [SAG94] Brian Smits, James Arvo, and Donald Greenberg. A clustering algorithm for radiosity in complex environments. In *SIGGRAPH '94 : Proc. 21st annual conference on Computer graphics and interactive techniques*, pages 435–442, 1994.
- [Sch93a] Christophe Schlick. A customizable reflectance model for everyday rendering. In *In Fourth Eurographics Workshop on Rendering*, pages 73–83, 1993.
- [Sch93b] Peter Schröder. Numerical integration for radiosity in the presence of singularities. In *FOURTH EUROGRAPHICS WORKSHOP ON RENDERING*, 1993.
- [Sch94a] Christophe Schlick. A Survey of Shading and Reflectance Models. *Comp. Graph. Forum*, 13(2), 1994.
- [Sch94b] Christophe Schlick. An Inexpensive BRDF Model for Physically-Based Rendering. *Computer Graphics Forum*, 13(3) :233–246, 1994.
- [Sch94c] Christophe Schlick. A fast alternative to phong’s specular model. pages 385–387, 1994.
- [Sch94d] Christophe Schlick. Quantization techniques for visualization of high dynamic range pictures. pages 7–18, 1994.
- [SDS⁺93] Chris Schoeneman, Julie Dorsey, Brian Smits, James Arvo, and Donald Greenburg. Painting with light. In *SIGGRAPH '93 : Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 143–146, New York, NY, USA, 1993. ACM Press.

- [SDS95] François Xavier Sillion, George Drettakis, and Cyril Soler. A clustering algorithm for radiance calculation in general environments. In *Rendering Techniques*, number 6 in Proceedings of Eurographics Rendering Techniques, 1995.
- [SGNS07] Peter-Pike Sloan, Naga K. Govindaraju, Derek Nowrouzezahrai, and John Snyder. Image-based proxy accumulation for real-time soft global illumination. In *PG '07 : Proceedings of the 15th Pacific Conference on Computer Graphics and Applications*, pages 97–105, Washington, DC, USA, 2007. IEEE Computer Society.
- [Sil95] François X. Sillion. A unified hierarchical algorithm for global illumination with scattering volumes and object clusters. *IEEE Transactions on Visualization and Computer Graphics*, 1(3) :240–254, 1995.
- [SIMP06] B. Segovia, J. C. Iehl, R. Mitanchey, and B. Péroche. Non-interleaved deferred shading of interleaved sample patterns. In *GH '06 : Proceedings of the 21st ACM SIGGRAPH/EUROGRAPHICS symposium on Graphics hardware*, pages 53–60, New York, NY, USA, 2006. ACM.
- [SIP07] B Segovia, JC Iehl, and B. Péroche. Coherent metropolis light transport with multiple-try mutations. Technical report, Laboratoire d’InfoRmatique en Image et Systèmes d’information (LIRIS), 2007.
- [SKS02] Peter-Pike Sloan, Jan Kautz, and John Snyder. Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. *ACM Trans. Graph.*, 21(3) :527–536, 2002.
- [Slo06] Peter-Pike Sloan. Normal Mapping for Precomputed Radiance Transfer. In *I3D '06 : Proc. symposium on Interactive 3D graphics and games*, pages 23–26. ACM, 2006.
- [SMGG01] Peter-Pike Sloan, William Martin, Amy Gooch, and Bruce Gooch. The Lit

- Sphere : A Model for Capturing NPR Shading from Art. In *Proc. Graphics Interface*, pages 143–150, 2001.
- [SMSA01] P. Shirley, S. Marschner, J. Stam, and M. Ashikhmin. State of the art in modeling and measuring of surface reflection. In *SIGGRAPH 2001 Course Notes*, 2001.
- [SP94] François Xavier Sillion and Claude Puech. *Radiosity and Global Illumination*. Morgan Kaufmann, 1994.
- [SP01] Xavier Serpaggi and Bernard Péroche. An adaptive method for indirect illumination using light vectors. In A. Chalmers and T.-M. Rhyne, editors, *Proc. Eurographics 2001*, volume 20(3), pages 278–287, 2001.
- [SS95] Peter Schröder and Wim Sweldens. Spherical wavelets : efficiently representing functions on the sphere. In *SIGGRAPH '95 : Proc. 22nd annual conference on Computer graphics and interactive techniques*, pages 161–172, 1995.
- [ST90a] Takafumi Saito and Tokiichiro Takahashi. Comprehensible rendering of 3-d shapes. *SIGGRAPH Comput. Graph.*, 24(4) :197–206, 1990.
- [ST90b] Takafumi Saito and Tokiichiro Takahashi. Comprehensible Rendering of 3-D Shapes. In *Proc. ACM SIGGRAPH '90*, pages 197–206. ACM, 1990.
- [Sta99] Jos Stam. Diffraction shaders. In *SIGGRAPH '99 : Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 101–110, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co.
- [Sta04] Jos Stam. Simulating diffraction. In *GPU Gems*, chapter 8. Addison-Wesley Professional, 2004.

- [SZC⁺07] Xin Sun, Kun Zhou, Yanyun Chen, Stephen Lin, Jiaoying Shi, and Baining Guo. Interactive relighting with dynamic brdfs. *ACM Trans. Graph.*, 26(3) :27, 2007.
- [TABI07] Hideki Todo, Ken-Ichi Anjyo, William Baxter, and Takeo Igarashi. Locally controllable stylized shading. *ACM Trans. Graph.*, 26(3) :17, 2007.
- [TL04] E. Tabellion and A. Lamorlette. An approximate global illumination system for computer generated films. *ACM Trans. on Graphics*, 23(3) :469–476, 2004.
- [TPWG02] Parag Tole, Fabio Pellacini, Bruce Walter, and Donald P. Greenberg. Interactive global illumination in dynamic scenes. In *SIGGRAPH '02 : Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 537–546, New York, NY, USA, 2002. ACM Press.
- [TR93] Jack Tumblin and Holly Rushmeier. Tone reproduction for realistic images. *IEEE Comput. Graph. Appl.*, 13(6) :42–48, 1993.
- [TS67a] K. Torrance and E. Sparrow. Polarization, directional distribution, and off-specular peak phenomena in light reflected from roughened surfaces. *J. Opt. Soc. Am.*, 57(9) :1105–1114, 1967.
- [TS67b] K. E. Torrance and E. M. Sparrow. Theory for off-specular reflection from roughened surfaces. pages 32–41, 1967.
- [TS06] Yu-Ting Tsai and Zen-Chung Shih. All-frequency precomputed radiance transfer using spherical radial basis functions and clustered tensor approximation. *ACM Trans. Graph.*, 25(3) :967–976, 2006.
- [VBGS08] Romain Vergne, Pascal Barla, Xavier Granier, and Christophe Schlick. Apparent relief : a shape descriptor for stylized shading. In *NPAR '08 : Proc. international symposium on Non-photorealistic animation and rendering*, pages 23–29. ACM, 2008.

- [VG94] Eric Veach and Leonidas J. Guibas. Bidirectional estimators for light transport. In *Proc. Eurographics Workshop Rendering*, pages 147–162, 1994.
- [VG97] E. Veach and L.J. Guibas. Metropolis light transport. In *Proc. SIGGRAPH'97*, pages 65–76, 1997.
- [VPB⁺09] Romain Vergne, Romain Pacanowski, Pascal Barla, Xavier Granier, and Christophe Schlick. Light warping for enhanced surface depiction. *ACM Transaction on Graphics (Proceedings of SIGGRAPH 2009)*, Aug 2009. To Appear.
- [VT04] M. Alex O. Vasilescu and Demetri Terzopoulos. Tensortextures : multilinear image-based rendering. In *Proc. SIGGRAPH*, 2004.
- [WA09] Rui Wang and Oskar Akerlund. Bidirectional importance sampling for unstructured direct illumination. *Computer Graphics Forum*, 28(2) :269–278, 2009.
- [WABG06] Bruce Walter, Adam Arbre, Kavita Bala, and Donald P. Greenberg. Multi-dimensional lightcuts. In *SIGGRAPH '06 : ACM SIGGRAPH 2006 Papers*, pages 1081–1088, New York, NY, USA, 2006. ACM.
- [Wal98] Bruce Walter. *Density Estimation Techniques for Global Illumination*. PhD thesis, Cornell University, 1998.
- [War91] Gregory Ward. Real pixels. In *Graphics Gems II*, pages 80–83. 1991.
- [War92] Gregory J. Ward. Measuring and modeling anisotropic reflection. In *ACM SIGGRAPH '92*, pages 265–272, 1992.
- [War94] Greg Ward. A contrast-based scalefactor for luminance display. pages 415–421, 1994.
- [WAT92] Stephen H. Westin, James R. Arvo, and Kenneth E. Torrance. Predicting reflectance functions from complex surfaces. *SIGGRAPH Comput. Graph.*, 26(2) :255–264, 1992.

- [WFA⁺05] Bruce Walter, Sebastian Fernandez, Adam Arbree, Kavita Bala, Michael Donikian, and Donald P. Greenberg. Lightcuts : a scalable approach to illumination. In *SIGGRAPH '05 : ACM SIGGRAPH 2005 Papers*, pages 1098–1107, New York, NY, USA, 2005. ACM.
- [WH92] Gregory J. Ward and Paul Heckbert. Irradiance gradients. In *Proc. Eurographics Workshop on Rendering 1992*, pages 85–98, 1992.
- [Whi80] Turner Whitted. An improved illumination model for shaded display. In *Communications of the ACM*, 23(6), June 1980.
- [WKB⁺02] Ingo Wald, Thomas Kollig, Carsten Benthin, Alexander Keller, and Philipp Slusallek. Interactive global illumination using fast ray tracing. In *EGRW '02 : Proceedings of the 13th Eurographics workshop on Rendering*, pages 15–24, Aire-la-Ville, Switzerland, Switzerland, 2002. Eurographics Association.
- [WNLH06] Rui Wang, Ren Ng, David Luebke, and Greg Humphreys. Efficient wavelet rotation for environmentmap rendering. In *Rendering Techniques 2006 : 17th Eurographics Workshop on Rendering*, pages 173–182, June 2006.
- [WRC88] G.J. Ward, F.M. Rubinstein, and R.D. Clear. A ray tracing solution for diffuse interreflection. In *Proc. SIGGRAPH'88*, pages 85–92, 1988.
- [WTL04] Rui Wang, John Tran, and David Luebke. All-frequency relighting of non-diffuse objects using separable brdf approximation. In *Proc. Eurographics Symposium on Rendering 2004*, pages 321–330, 2004.
- [WTL06a] Rui Wang, John Tran, and David Luebke. All-frequency relighting of glossy objects. *ACM Trans. Graph.*, 25(2) :293–318, 2006.
- [WTL06b] Rui Wang, John Tran, and David Luebke. All-frequency relighting of glossy objects. *ACM Trans. Graph.*, 25(2) :293–318, 2006.

- [WW07] Andrea Weidlich and Alexander Wilkie. Arbitrarily layered micro-facet surfaces. In *GRAPHITE '07 : Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia*, pages 171–178, New York, NY, USA, 2007. ACM.
- [WWHL07] R. Peter Weistroffer, Kristen R. Walcott, Greg Humphreys, and Jason Lawrence. Efficient Basis Decomposition for Scattered Reflectance Data. In *Proc. Eurographics Symposium on Rendering*, 2007.
- [WWLP06] Alexander Wilkie, Andrea Weidlich, Caroline Larboulette, and Werner Purgathofer. A reflectance model for diffuse fluorescent surfaces. In *Proceedings of Graphite 2006*, pages 321–328, 11 2006.
- [WWZ⁺09] Rui Wang, Rui Wang, Kun Zhou, Minghao Pan, and Hujun Bao. An efficient gpu-based approach for interactive global illumination. In *SIGGRAPH '09 : ACM SIGGRAPH 2009 Papers*. ACM, 2009.
- [Wym05] Chris Wyman. An approximate image-space approach for interactive refraction. *ACM Trans. Graph.*, 24(3) :1050–1053, 2005.
- [ZERB05] Todd Zickler, Sebastian Enrique, Ravi Ramamoorthi, and Peter Belhumeur. Reflectance sharing : Image-based rendering from a sparse set of images. In *Proc. Eurographics Symposium on Rendering*, pages 253–265, 2005.
- [ZHWC08] Kun Zhou, Qiming Hou, Rui Wang, and Baining Guo. Real-time kd-tree construction on graphics hardware. *ACM Trans. Graph.*, 27(5) :1–11, 2008.
- [ZHXC09] Long Zhang, Ying He, XueXiang Xie, and Wei Chen. Laplacian Lines for Real Time Shape Illustration. In *I3D '09 : Proc. symposium on Interactive 3D graphics and games*. ACM, 2009.

- [ZSP98] J. Zaninetti, X. Serpaggi, and B. Péroche. A vector approach for global illumination in ray tracing. In *Proc. Eurographics 1998*, pages 149–158, 1998.