



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

**Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique**

Université Mohamed Khider-BISKRA

Faculté des Science Exactes, des Science de la Nature et de la Vie

Département d'informatique

N° d'ordre : IVA17 /M2/2021

Mémoire

Présenté pour obtenir le diplôme de master académique en

Informatique

Parcours : **Images et vie artificielle**

**Occultation ambiante dans l'espace
écran basé sur la technique du bump
mapping**

Par :

Nouar Amira

Soutenu en 28/06/2022, devant le jury composé de :

Chighoub Rabiaa

MCB

Président

Zerari abd El Mouméne

MCB

Rapporteur

Berima Salima

MAA

Examineur

Année universitaire 2021-2022

Dédicace

Je exprimer ma très profonde gratitude mes parent

Mes frères Et Mes sœurs,

Ma famille et mes amis meilleurs

Cette réalisation n'aurait pas été possible sans eux. Merci. . .

Remerciement

Tout d'abord, je tiens à remercier Allah de m'avoir donné le courage et l'engagement d'accomplir ce travail.

Je tiens à exprimer ma sincère gratitude à mon encadreur Dr. ABD EL MOUMÉNE ZERARI pour avoir toujours été leur pour moi lorsque j'ai rencontré des problèmes ou que j'ai eu des questions sur mes recherches, pour ses conseils que j'ai eu tout au long de ce projet ou même avant, pour corriger les différentes fautes que j'ai pu faire ou l'incompréhension que j'avais pour plusieurs concepts.

Je tiens à exprimer mes remerciements à Md. Chighoub Rabiaa pour m'avoir donnée le grand honneur de présider le jury de ma soutenance et Je remercie également Md. Berima Salima pour le temps et l'effort qu'elles ont fourni pour lire et évaluer ce travail.

Je tiens également à remercier tous les professeurs d'informatique de l'Université Mohamed Khidher pour leurs précieux enseignements et conseils pendant mon temps d'étude.

Je remercie mes camarades pour les belles années que nous avons passées ensemble et tout le plaisir que nous avons eu.

Résumé

L'illumination global en temps réel a toujours été l'intérêt de l'infographie où il faut des approximations pour un rendu des scènes réalistes en tenant compte toutes sortes d'interactions entre les objets de la scène ou entre eux et une source de lumière ici nous nous intéressons à l'une des méthodes utilisées pour approximer l'illumination globale qui est l'occlusion ambiante qui génère des ombres douces dans des endroits à faible contribution lumineuse afin d'améliorer le réalisme de la scène bien qu'elle ait d'excellents résultats pour l'approximation de l'illumination globale, elle est pas pratique à utiliser lorsque le temps réel est demandé en raison du fait qu'il dépend de la géométrie dans ses calculs.

l'occultation ambiante dans l'espace écran considéré comme l'une des différentes solutions pour approximer l'obscurance ambiante qui est une version étendue de l'occlusion ambiante en utilisant l'espace écran alors qu'il a des résultats efficaces en temps réel, elle provoque un faible performance, donne des résultats bruyants sans parler sur l'ignorance des contributions importantes de la scène.

Notre but est d'implémenter la technique de SSAO en se basant la technique de bump mapping .

mots clés: illumination globale, obscurance ambiante , occlusion ambiante , espace écran, filtrage, échantillonnage, temps-réel .

Abstract

Real-time global illumination has always been the interest of computer graphics where approximations are needed to render realistic scenes taking into account all kinds of interactions between objects in the scene or between them and a light source here we are interested in one of the methods used to approximate the global illumination which is the ambient occlusion which generates soft shadows in places with low light contribution in order to improve the realism of the scene although it has d excellent results for the global illumination approximation, it is impractical to use when real time is required due to the fact that it is geometry dependent in its calculations.

Ambient occlusion in screen space considered as one of the different solutions to approximate ambient obscurance which is an extended version of ambient occlusion using screen space while it has effective results in real time , it causes low performance, gives noisy results not to mention about ignoring important scene contributions.

Our goal is to implement the SSAO technique based on the bump mapping technique

Keywords: global illumination, ambient obscurity, ambient occlusion, screen space, filtering, sampling, real-time,

ملخص

لطالما كانت الإضاءة العالمية في الوقت الفعلي من مصلحة رسومات الكمبيوتر حيث يلزم التقريب لتقديم مشاهد واقعية مع مراعاة جميع أنواع التفاعلات بين الكائنات في المشهد أو بينها وبين مصدر الضوء هنا نحن مهتمون بإحدى الطرق المستخدمة لتقريب الإضاءة العالمية وهي الإغلاق المحيط الذي يولد ظلالاً ناعمة في الأماكن ذات الإضاءة المنخفضة من أجل تحسين واقعية المشهد على الرغم من أنه يحتوي على نتائج ممتازة لتقريب الإضاءة العالمي ، فمن غير العملي استخدامه عندما يكون الوقت الحقيقي مطلوباً نظراً لحقيقة أنه يعتمد على الهندسة في حساباته

يعتبر الانسداد المحيط في مساحة الشاشة أحد الحلول المختلفة لتقريب الغموض المحيط وهو نسخة ممتدة من الانسداد المحيط باستخدام مساحة الشاشة بينما تؤدي إلى نتائج فعالة في الوقت الفعلي ، فهي تسبب أداءً منخفضاً ، وتعطي نتائج صاخبة ناهيك عن تجاهل المهم مساهمات المشهد

بناءً على تقنية رسم الخرائط SSAO هدفنا هو تنفيذ تقنية

الكلمات الرئيسية: الإضاءة العالمية ، الغموض المحيط ، الانسداد المحيط ، مساحة الشاشة ، التصفية ، أخذ العينات ، الوقت الفعلي.

Table des matières

Table des matières.....	i
Liste des tableaux.....	iii
Liste des figures :	iv
Introduction générale.....	1
1 Occultation ambiante	3
1.1 Introduction.....	3
1.2 Illumination locale.....	3
1.2.1 Types de réflexion locale	4
1.2.2 Les techniques d'illumination locale	4
1.3 Illumination global	5
1.4 Equation de rendu global	5
1.4.1 Fonction de Distribution de la Réflectance Bidirectionnelle	5
1.4.2 Equation de rendu	6
1.4.3 Résolution Équation de rendu	6
1.4.4 Techniques d'illumination globale	6
1.5 Les techniques d'approximations d'illuminations globales	7
1.5.1 Radiosité Instantanée	7
1.5.2 Reflective Shadow Maps	8
1.5.3 Imperfect Shadow Maps	8
1.5.4 Occlusion Ambiante	9
1.6 Brève historique sur occultation ambiante.....	10
1.6.1 Obscurance ambiante.....	10
1.6.2 Occultation ambiante	11
1.6.3 Définition d'Occlusion Ambiante	14
1.7 Synthèse.....	14
1.8 Conclusion	15
2 Occultation ambiante dans l'espace écran	16
2.1 Introduction.....	16
2.2 Les méthodes d'occultation Ambiante en temps réel	16
2.2.1 Méthode basée sur l'objet	16
2.2.2 Méthode basée sur les sommets	17
2.2.3 Méthode basée sur l'espace d'écran	19
2.2.4 Techniques d'Occultation ambiante dans l'espace d'écran	20

2.3	Bilan	23
2.4	Les filtres pour SSAO	25
2.4.1	Filtre Anti Aliasing.....	25
2.4.2	Filtre Gaussien	26
2.4.3	Filtre Bilatéral	26
2.5	Bump Mapping	27
2.6	Synthèse.....	30
2.7	Conclusion.....	31
3	Mise en œuvre, résultats et perspective	32
3.1	Introduction.....	32
3.2	Objectifs et description.....	32
3.3	Conception Générale	32
3.3.1	Modélisation de la scène 3D	33
3.3.2	Placer la caméra	34
3.3.3	Calculer le G-Buffer	34
3.3.4	Bump mapping.....	34
3.3.5	Création des échantillons	35
3.3.6	Calculer le SSAO	35
3.3.7	Illumination.....	36
3.3.8	Application du Filtre.....	36
3.4	Réalisation	37
3.4.1	Configurations matérielles et logiciels utilisés.....	37
3.4.2	Structure de l'application.....	38
3.5	Algorithmes	42
3.6	Résultats	46
3.7	Conclusion.....	49
	Conclusion générale	51
	Bibliographie	52

Liste des tableaux

Tableau 2. 1 : Comparaison entre les techniques de SSAO.....	25
--	----

Liste des figures :

Figure 1. 1 : Illumination locale & Illumination global [37]	4
Figure 1. 2 : Types de réflexion locale [36].....	4
Figure 1. 3 : La réflexion de Phong [35]	5
Figure 1. 4 : Illumination locale & Illumination global [37]	5
Figure 1. 5: Paramètres d'une BRDF au point x [4].....	6
Figure 1. 6 : Figure 1.6 : Radiosité Instantanée [9].....	8
Figure 1. 7 : A gauche, image sans RSM. Au milieu, la même image à l'aide de RSM. A droite, l'image montre la différence entre les deux images [38].	8
Figure 1. 8 : Les ISM peuvent gérer des scènes géométriquement complexes. À gauche : cartes d'ombre classiques. À droite : ISM [11].	9
Figure 1. 9: Éclairage direct (à gauche), lumière ambiante (au milieu) et éclairage direct avec lumière ambiante (à droite) [12].....	9
Figure 1. 10 : Accessibilité d'un point par obscursans [32].	11
Figure 1. 11 : Lancer de rayon pour résoudre l'intégrale. Les rayons verts sont visibles ($V=1$) alors que les rayons rouges sont occlus ($V=0$) [12].	12
Figure 1. 12 : Occlusion ambiante [14].	14
Figure 2. 1 : Les ombres sont rendues en utilisant la méthode Champs d'occlusion ambiante [15].	16
Figure 2. 2 : Exemple des ombres de contact [16].....	17
Figure 2. 3 : Ajouté un réalisme avec l'occlusion ambiante et éclairage indirect [17].	18
Figure 2. 4 : Stanford dragon sous l'éclairage de la plage dynamique élevée avec l'utilisation d'occlusion ambiante, qui a été calculé sur le matériel graphique [18].	18
Figure 2. 5 : Images rendues avec la méthode SSDO [19].....	19
Figure 2. 6 : Occultation ambiante dans l'Espace Ecran [39].	19
Figure 2. 7 : Crytek OA, les échantillons verts et rouges sont visibles et occlus, respectivement. Un échantillon est aussi montré projetée sur la surface de la scène [22].	20
Figure 2. 8 : Starcraft 2 OA, Les échantillons verts et rouges sont visibles et occlus,	21
Figure 2. 9 : Horizon basée Occlusion ambiante avec Ray Marching [22].	22
Figure 2. 10 Volumétrique Obscurcissement. Les échantillons sont présentés sous forme de carrés gris. Les segments de ligne verts et rouge indiquent des volumes visibles et occlus, respectivement. On notera que p contribue elle-meme en tant que position de l'échantillon [22]......	22
Figure 2. 11 : Alchemy AO. Les vecteurs verts v , sont utilisés pour le calcul de la durée OA. Les points d'échantillonnage rouge ont été rejetés parce qu'ils se situent en dehors(ou sur le bord de) l'hémisphère [22]......	23
Figure 2. 12 : Comparaison entre l'entrée (à gauche) sans filtre anti aliasing. (Au milieu) avec filtre anti aliasing (FXAA) et (à droite) avec filtre anti aliasing (SMAA) [41].	25
Figure 2. 13 : Comparaison entre : (à gauche) sans filtre Gaussien et (à droite) avec filtre Gaussien [42]......	26
Figure 2. 14 : Comparaison entre : (à gauche) avec filtre Bilatéral et (à droite) sans filtre Bilatéral [43].	27
Figure 2. 15 : (À gauche) Une sphère sans bump mapping. (Au milieu) Une bump map à appliquer à la sphère. (À droite) La sphère avec la bump mapping appliquée [45].	28
Figure 2. 16: (a gauche) bump mapping, (a droite) texture map[46]	30
Figure 2. 17: Image rendue avec Texture Map + Bump Map[46]	30

Figure 3. 1 : Conception générale.	33
Figure 3. 2 : G-buffer [50]	34
Figure 3. 3 : Bump Mapping [51].....	35
Figure 3. 4 : Échantillonnage dans un hémisphère.....	36
Figure 3. 5 : Filtrage.	37
Figure 3. 6 : Structure général.....	41
Figure 3. 7 : Résultats de G-Buffer et du SSAO.....	46
Figure 3. 8 : Résultats du noyau d'échantillonnage, appliqué au modèle 3D sibenik.....	46
Figure 3. 9 : SSAO sans bump mapping, (à gauche). SSAO avec bump mapping (à droite). .	47
Figure 3. 10 : SSAO sans bump mapping, (à gauche). SSAO avec bump mapping (à droite). Model de sponza0.....	47
Figure 3. 11 : SSAO sans filtre bilatéral, (à gauche). SSAO avec filtre bilatéral (à droite).	48
Figure 3. 12 : Comparaison entre les noyaux.....	49
Figure 3. 13 : Filtre.....	49

Introduction générale

Les images de synthèse obtenues en utilisant les techniques d'illumination globale qui représente non seulement les sources de lumière directes, mais aussi l'éclairage indirect qui correspond à la réflexion et la réfraction de la lumière sur d'autres objets dans le modèle.

L'illumination globale permet un éclairage réaliste notamment dans des espaces intérieurs. Mais elles sont également beaucoup plus lentes et plus coûteuses en termes de temps de calcul et ça représente la problématique de notre projet.

On a quelques approximations d'illumination globale simples et faciles à gérer parmi eux l'occultation ambiante qui est une technique pour mesurer la quantité de lumière ambiante au rayon qui est bloqué par les objets à l'entour. Cette approximation représente une solution d'équation de rendu, mais n'est pas pratique pour le rendu en temps réel pour une scène 3D complexe.

Les méthodes d'occlusion ambiantes en temps réel sont des solutions pour résoudre le problème, Parmi eux nous avons choisi l'occultation ambiante dans espace écran (Screen Space Ambient Occlusion) qui mesure l'occlusion ambiante en profondeur de pixel sur une partie de l'écran, cette technique apporte un défaut c'est qu'elle produit un bruit dans la scène, pour cela nous avons besoin d'appliquer un filtre pour éliminer les bruits et donne une qualité d'image proche de la réalité.

L'implémentation de technique d'occlusion ambiante dans l'espace écran en temps réels on se basant sur la technique du Bump Mapping est l'objectif principal de notre travail.

Pour réaliser cet objectif, nous avons organisé notre mémoire en trois chapitres.

Nous avons présenté dans le premier chapitre le problème d'illumination globale qui formalisé par l'équation de rendu et suggéré les techniques d'approximation d'illumination globale interactive pour résoudre ce problème. Ensuite nous avons introduisons l'occlusion ambiante et les fondements mathématiques sur lesquels elle repose.

Dans le chapitre deux nous commençons par les différentes méthodes d'occlusion ambiante en temps réel, puis la Techniques d'Occultation ambiante dans l'espace d'écran (SSAO) et les techniques de filtrages, à la fin on définit la technique du bump mapping.

INTRODUCTION GENERALE

Le troisième chapitre décrit les objectifs et description de projet et la conception générale étape par étape puis la réalisation et la structure de notre application, ainsi que les résultats obtenus des différentes étapes de l'implémentation.

CHAPITRE 1 : OCCULTATION AMBIANTE

1 Occultation ambiante

1.1 Introduction

L'illumination définit le transport de la lumière dans la scène, est la projection de lumière sur les surfaces d'objets qui permet de les voir à l'œil nu.

On distingue deux types d'illumination : locale et globale.

L'illumination globale est un terme utilisé en infographie pour faire référence à tous les phénomènes d'éclairage causés par l'interaction entre les surfaces, Il y a quelques approximations de l'illumination globale qui sont plus faciles à gérer.

Parmi ces approximations l'occultation ambiante est un modèle d'éclairage qui calcule la luminosité d'un pixel par rapport à des objets proches de la scène.

Dans ce chapitre, nous commençons par les modèles d'illumination, puis on définit l'équation du rendu de l'illumination global et on essaye de trouver une solution analytique et on détaille l'illumination globale en se basant sur les techniques d'approximation de l'illumination globale.

1.2 Illumination locale

C'est une méthode de calcul d'illumination utilisée dans l'informatique graphique. Elle permet de calculer la luminosité apparente d'un volume en ne considérant que les interactions entre les sources lumineuses et les objets. L'étude de l'illumination se base sur deux modèles : le modèle d'illumination qui permet de déterminer la couleur d'un point de la surface, ce modèle s'exprime à l'aide d'équations dont les variables sont associées au point de l'objet, et le modèle d'ombrage (ou shading) qui détermine la manière d'appliquer le modèle d'illumination sur la surface [1].

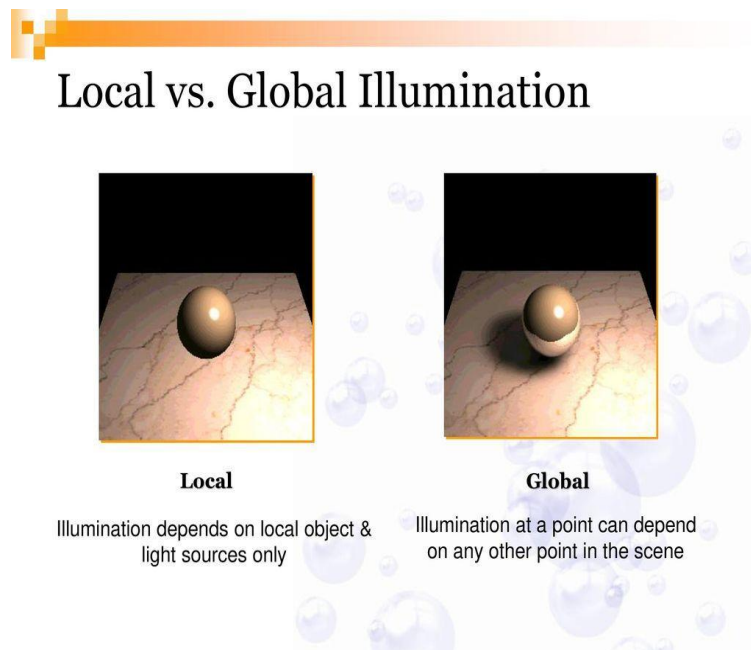


Figure 1. 1 : Illumination locale & Illumination global [37]

1.2.1 Types de réflexion locale

Un modèle de réflexion décrit l'interaction entre la lumière et une surface en fonction des propriétés du matériau constitutif de la surface ainsi que de la nature de la lumière et de son incidence. On considère trois types de réflexion : [2]

- a. Réflexion ambiante
- b. Réflexion diffuse
- c. Réflexion spéculaire

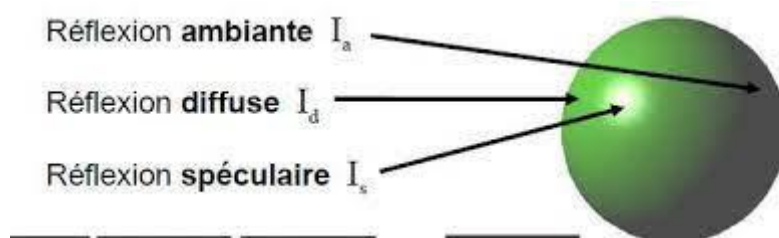


Figure 1. 2 : Types de réflexion locale [36]

1.2.2 Les techniques d'illumination locale

1.2.2.1 Modèle d'illumination de Phong

Le modèle le plus fameux parmi les modèles d'illumination locale est le modèle Phong, c'est un modèle local simple, rapide à calculer et très utilisé par de nombreux systèmes de synthèse d'image. Il permet de calculer la lumière réfléchie par le point de la surface étudié, pour cela

CHAPITRE 1 : OCCULTATION AMBIANTE

il combine trois éléments : la lumière ambiante, la lumière diffuse et la lumière spéculaire comme illustré dans la figure 1.3.

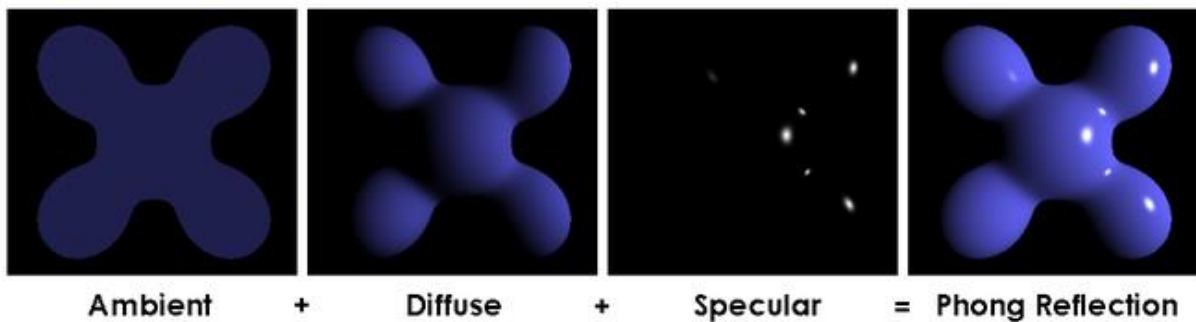


Figure 1. 3 : La réflexion de Phong [35]

1.3 Illumination global

Les algorithmes d'illumination globale sont ceux qui, en déterminant la lumière tombant sur une surface, tiennent compte non seulement de la lumière qui a pris un chemin partant directement d'une source lumineuse (illumination directe), mais également la lumière ayant subi la réflexion d'autres surfaces dans la scène à 3 dimensions (illumination indirecte) [3].

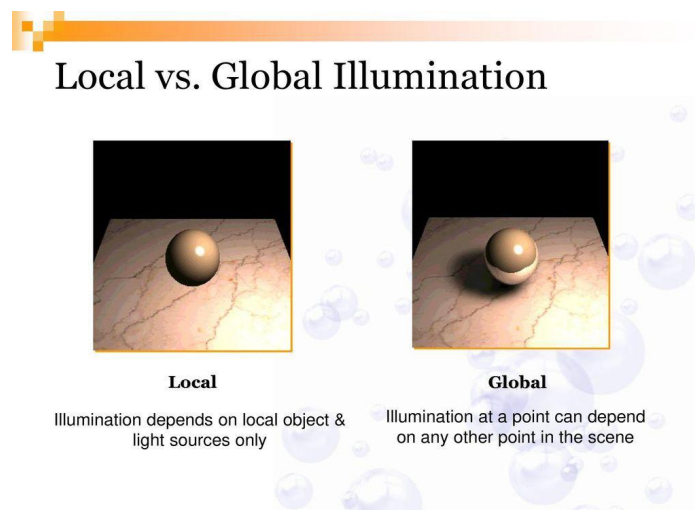


Figure 1. 4 : Illumination locale & Illumination globale [37]

1.4 Equation de rendu global

1.4.1 Fonction de Distribution de la Réflectance Bidirectionnelle

Une fonction de distribution de réflectance bidirectionnelle (BRDF) décrit comment la lumière incidente est dispersée à un point de surface sur un hémisphère qui est orienté par rapport à sa normale. Il a été spécifié pour la première fois par Nicodème en 1970 [4].

CHAPITRE 1 : OCCULTATION AMBIANTE

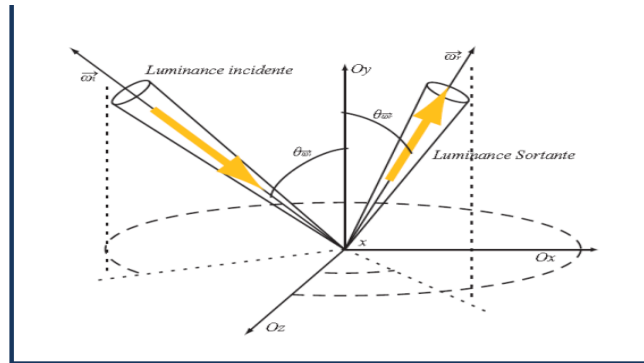


Figure 1. 5: Paramètres d'une BRDF au point x [4].

1.4.2 Equation de rendu

En informatique graphique, la distribution de l'énergie dans une scène a été formulée mathématiquement pour la première fois par Kajiya [5] en 1986, sous forme d'une équation qui s'appelle l'équation de rendu. Cette dernière, permet de calculer la luminance sortante d'un point de la surface en se basant sur la luminance émise par la surface, la luminance incidente et les propriétés des matériaux de la surface.

1.4.3 Résolution Équation de rendu

Pour calculer l'illumination globale, il faut donc résoudre l'équation du rendu global, pour cela il existe trois catégories de solutions [6] :

- a. Solution analytique
- b. Solutions déterministes
- c. Solutions Stochastiques

1.4.4 Techniques d'illumination globale

Il existe des méthodes de l'illumination globale pour rendre les environnements en trois dimensions plus réaliste, nous pouvons citer quelque algorithme :

a. Lancer de photon (Path Tracing)

Est une amélioration de lancer de rayon (ray tracing), utilisée pour déterminer l'illumination globale. Le lancer de photons consiste à envoyer des photons depuis les sources lumineuses dans des directions aléatoires. Les photons sont envoyés en nombre fini dans la scène et la puissance de la source est divisée équitablement pour chacun d'entre eux. Lorsqu'un photon rencontre une face, il est stocké dans une structure de données appelée carte de photons (en anglais, photon map) puis est relancé dans la scène de manière

CHAPITRE 1 : OCCULTATION AMBIANTE

aléatoire en ayant perdu une partie de son énergie : celle que la face a absorbée. Il existe de nombreuses heuristiques pour la destruction des photons : lorsque leur énergie restante est inférieure à un certain seuil, quand un nombre maximum de rebonds est atteint, où encore en tirant un nombre aléatoire pour savoir s'il continue à vivre. Ces traitements sont indépendants du point de vue et la création d'une image peut ensuite être réalisée soit à l'aide d'un lancer de rayons, soit par des méthodes à base de Z-buffer [7].

b. Radiosité

La radiosité est définie comme la quantité totale d'énergie émise par un point de surface, par unité de surface. Les premiers algorithmes de radiosité apparaissent en 1984 dans le but de modéliser les interactions lumineuses entre les surfaces diffuses. Ces algorithmes proviennent de l'application de la théorie de l'énergie radiative au problème d'illumination globale. La radiosité a un avantage par rapport au ray tracing et c'est qu'elle est indépendante de la position de l'observateur, elle permet le calcul d'une solution qui est indépendante de celui-ci. Après les échanges d'énergies entre les surfaces, si le point de vue de l'observateur change, un minimum de changement est demandé en utilisant la radiosité. Cela n'est pas le cas avec le ray tracing qui calcul une solution dépendante du point de vue de l'observateur. La radiosité utilise les principes d'émission, de réflexion et d'inter-réflexion pour simuler les réflexions et inter-réflexions de la lumière dans la scène pour arriver finalement à un équilibre et une convergence vers la bonne solution de radiosité [8].

1.5 Les techniques d'approximations d'illuminations globales

Les techniques d'illuminations globales sont très compliquées et demande beaucoup de temps de calcul et pas pratique pour un rendu en temps réel. Les chercheur en proposé des techniques d'approximation d'illumination global interactive pour résoudre l'équation du rendu et accélérer les calculs. Parmi ces méthodes on peut citer :

1.5.1 Radiosité Instantanée

La radiosité instantanée a été introduite dans le domaine de l'infographie par Keller [9], qui consiste à remplacer le rebond de lumière secondaire par un point lumineux virtuel appelé (VPL).

Les VPL sont créés à l'intersection de photons émis par une source lumineuse avec la géométrie et ils encapsulent la lumière qui est réfléchié sur les surfaces. Après leur

CHAPITRE 1 : OCCULTATION AMBIANTE

génération, la scène est rendue normale et les VPL sont utilisés comme feux de point réguliers.

La contribution de ces sources lumineuses (indirectes) est calculée en prenant en compte l'occlusion avec l'utilisation d'une technique de génération d'ombre, comme les cartes d'ombre. Les méthodes de Radiosité Instantanée souffrent généralement d'artefacts temporels lorsqu'un ensemble gérable de VPL est utilisé, en particulier dans des conditions d'éclairage dynamique [37].



Figure 1. 6 : Figure 1.6 : Radiosité Instantanée [9]

1.5.2 Reflective Shadow Maps

Les cartes d'ombre réfléchissantes sont définies comme considérant chaque pixel des pixels des cartes d'ombre comme une source de lumière indirecte qui génère l'éclairage indirect à un rebond dans une scène avec une seule source de lumière où toutes les surfaces visibles dans ses cartes d'ombre sont considérées comme indirectes à un rebond, ce qui fait que les shadow maps contiennent des informations d'illumination indirecte, ce qui conduit à considérer que les shadow maps réfléchissantes sont des collections de textures qui contiennent des informations de visibilité des surfaces à partir d'une source lumineuse où cette collection est échantillonnée pour sélectionner les surfaces qui sont utilisés comme des sources de lumière virtuel [10].



Figure 1. 7 : A gauche, image sans RSM. Au milieu, la même image à l'aide de RSM. A droite, l'image montre la différence entre les deux images [38].

1.5.3 Imperfect Shadow Maps

CHAPITRE 1 : OCCULTATION AMBIANTE

Imperfect Shadow Maps Shadow Maps (ISM) qui a été introduit en 2008[11], qui utilise une représentation ponctuelle à basse résolution afin de capturer la visibilité dans la scène plutôt que des cartes d'ombre à haute résolution [48].

Là où les cartes d'ombres imparfaites utilisent une représentation de géométrie de scène différente pour rendre les cartes d'ombres rapidement améliorées en utilisant des points de géométrie plutôt qu'en utilisant des polygones qui peuvent fournir les cartes d'ombres imparfaites pour le calcul d'éclairage indirect[11], afin de rendre un grande échelle de cartes d'ombres en un seul appel de tirage avec moins de réduction de la géométrie [11].

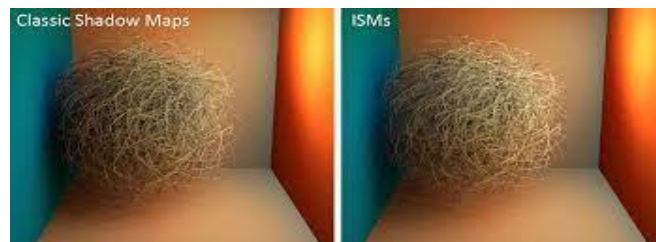


Figure 1. 8 : Les ISM peuvent gérer des scènes géométriquement complexes. À gauche : cartes d'ombre classiques. À droite : ISM [11].

1.5.4 Occlusion Ambiante

1.5.4.1 Lumière ambiante

Une méthode traditionnelle d'approximation, l'éclairage indirect, consiste à remplacer le calcul par une constante. Ce terme constant est appelé lumière ambiante dans les modèles d'ombrage tels que Blinn-Phong ou Cook-Torrance. En utilisant une constante, nous supposons que la lumière atteint chaque point dans toutes les directions et avec une même intensité.

Ces faits impliquent que la quantité de lumière ambiante incidente sur un point de surface est indépendante de l'orientation de la surface, et la quantité de réflexion perçue est indépendante de la direction de la vue.



Figure 1. 9: Éclairage direct (à gauche), lumière ambiante (au milieu) et éclairage direct avec lumière ambiante (à droite) [12].

Dans la Figure 1.10 à gauche, les zones où la lumière n'atteint pas sont représentées complètement en noir et aucun détail géométrique n'est affiché.

CHAPITRE 1 : OCCULTATION AMBIANTE

Dans la Figure 1.10 à droite, la scène est totalement éclairée même les endroits dans l'ombre sont également visibles.

Le terme ambiant constant rend la scène terne et inintéressante, et il est difficile pour l'observateur d'apprécier la profondeur de la scène. C'est exactement le même problème que "Ambient Occlusion" tente de résoudre. C'est ce dont nous parlerons en détail dans la prochaine section [12].

1.6 Brève historique sur occultation ambiante

1.6.1 Obscurance ambiante

Zhukov et al. [33] introduisent la notion d'obscurance, les auteurs proposent un modèle empirique pour expliquer une meilleure approximation de l'éclairage ambiant que l'on nomme obscurance ambiante (en anglais Ambient Obscurance). Dans ce travail, les auteurs simulent la composante diffuse de l'illumination uniforme éloignée de l'illumination globale, en calculant le nombre de points occlus par géométrie voisine.

Dans une scène, les algorithmes d'illumination globale simulent le transport de la lumière en utilisant des méthodes approchées de l'équation du rendu proposées par James Kajiya[34], tandis que l'obscurance ambiante calcule un facteur d'atténuation de la lumière en tout point de la surface. L'obscurité d'un point p est définie par l'équation.

$$L_0(p \rightarrow \omega_0) = \frac{1}{\pi} \int_{\Omega} \rho(d(p \leftarrow \omega_i)) \cos\theta_i d\omega_i \quad (1)$$

La définition ci-dessus pour l'occultation ambiante révèle un problème subtil. Quelle est la valeur de l'occultation ambiante pour une scène fermée ? La réponse est zéro, ou noir. C'est parce que chaque rayon de chaque point p finit par toucher une autre surface, auquel cas $V(p \leftarrow w^i)$ est nul pour tout p et pour tout w^i .

Le problème ci-dessus a une solution très simple. Au lieu d'utiliser la fonction de visibilité $V(p \leftarrow w^i)$, la distance entre p et le potentiel occluant doit être calculé dans la direction w^i , notée $d(p \leftarrow w^i)$, puis appliquer une fonction ρ à cette distance pour donner une valeur d'occultation.

Cette équation modifiée est l'équation d'obscurance ambiante qui peut être vue comme une forme d'occultation ambiante limitée dans le temps.

En pratique, la fonction ρ dans l'équation 1 réduit typiquement la valeur d'occultation à zéro lorsque la distance est supérieure à un certain seuil, c'est-à-dire :

CHAPITRE 1 : OCCULTATION AMBIANTE

$$\rho(d) = \begin{cases} f(d) \in [0; 1] & d < \text{seuil} \\ 0 & \text{sinon} \end{cases} \quad (2)$$

La fonction ρ sert donc trois objectifs. Premièrement, ρ rend le travail de l'obscurité ambiante pour les scènes fermées en donnant 0 pour les occulteurs qui se trouvent à une distance supérieure à un certain seuil, la valeur d'obscurité n'est pas nulle dans les scènes fermées, contrairement à l'occultation ambiante. Deuxièmement, puisque ρ réduit la valeur d'occultation à zéro pour des points éloignés, l'obscurité ambiante d'un point donné devient un calcul relativement local. C'est-à-dire que la valeur d'obscurité ambiante n'est qu'une fonction de la géométrie proche du point, ce qui se traduit par des implémentations plus efficaces en pratique. Finalement, ρ donne aux occulteurs voisins une valeur d'occultation plus élevée (une valeur plus proche de 0) que les occulteurs distants.

Enfin, il est important de noter que dans la littérature, les termes occultation ambiante et obscurité ambiante sont souvent utilisés de façon interchangeable pour désigner cette dernière.

Par la suite, Bredow [31] et Landis [32] simplifient ce modèle en supprimant la notion de distance. Ce nouveau modèle, appelé Occultation Ambiante. La formule 1 est simplifiée en remplaçant la fonction de distance ρ par une fonction de visibilité V (équation 6).

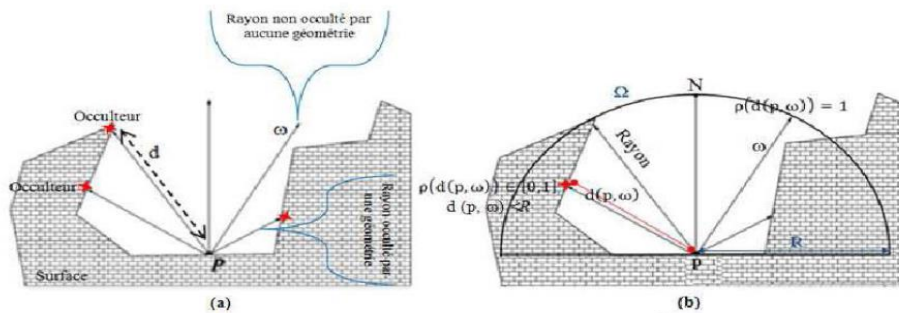


Figure 1. 10 : Accessibilité d'un point par occulteurs [32].

1.6.2 Occultation ambiante

L'occultation ambiante peut être considérée comme une forme d'éclairage indirect à mi-chemin entre la lumière ambiante et l'éclairage indirect réel. Au lieu d'utiliser un terme ambiant constant, l'occultation ambiante produit un terme ambiant pour chaque point de la scène. Cette valeur dépend de la position et d'une fonction de la visibilité du point, et l'occultation ambiante fait deux hypothèses lors du calcul de cette valeur.

CHAPITRE 1 : OCCULTATION AMBIANTE

- La première hypothèse est que la surface est une surface diffuse parfaite, c'est-à-dire que la surface réfléchit la lumière de manière égale dans toutes les directions. Dans ce cas, la BRDF de la surface $f_r(p, w^i, w^o)$ devient une constante k .

$$L_0(p \rightarrow \omega_0) = k \int_{\Omega} L_i(p \leftarrow \omega_i) \cos\theta_i d\omega_i \quad (3)$$

- La deuxième hypothèse est que la lumière atteint potentiellement un point x dans toutes les directions (voir figure 1.12). En d'autres termes, la direction dans laquelle la lumière atteint effectivement x devient non pertinente, et nous supposons que l'intensité de la lumière atteignant potentiellement x est égale dans toutes les directions.

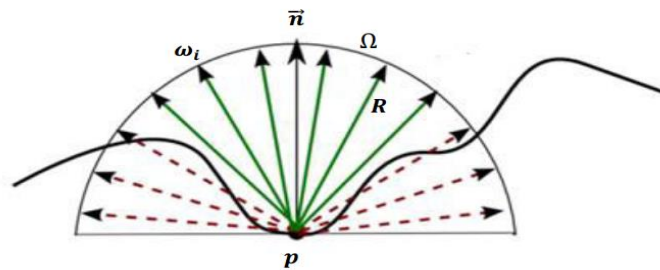


Figure 1. 11 : Lancer de rayon pour résoudre l'intégrale. Les rayons verts sont visibles ($V=1$) alors que les rayons rouges sont occlus ($V=0$) [12].

L'occultation ambiante n'ajoute pas simplement la contribution de la lumière de toutes les directions, car cela se traduirait par un terme de lumière ambiante constant. Au lieu de cela, l'occultation ambiante prend en compte la visibilité du point.

Le calcul de l'OA d'une scène s'effectue typiquement en pré-calcul par la méthode du lancer de rayons. Pour chaque direction w^i dans laquelle la lumière peut potentiellement atteindre p , l'occultation ambiante détermine si un rayon dans la direction w^i touche une autre surface (voir figure). Si c'est le cas, on dit que le point p est occlus dans la direction w^i , dans ce cas, aucune contribution de lumière dans la direction w^i n'est ajoutée. D'un autre côté, si le rayon n'intersecté pas une surface, la lumière est déterminée pour atteindre p dans la direction w^i , dans ce cas, cette lumière contribue à la valeur finale de la lumière réfléchi L^o . Cette description est plus succinctement exprimée par l'équation suivante :

$$L_0(p \rightarrow \omega_0) = k \int_{\Omega} V(p \leftarrow \omega_i) \cos\theta_i d\omega_i \quad (4)$$

Où

CHAPITRE 1 : OCCULTATION AMBIANTE

$$V(p \leftarrow \omega_i) = \begin{cases} 0 & \text{si } p \text{ est occlus dans la direction } \omega_i \\ 1 & \text{sinon} \end{cases} \quad (5)$$

La différence entre les équations 3 et 4 est que le terme de lumière entrant L_i ($\leftarrow w^i$) de l'équation 3 est remplacé par une fonction de visibilité V ($\leftarrow w^i$). Cette fonction de visibilité retourne 0 si p est occlus dans la direction w^i et 1 sinon. De cette façon, l'occultation ambiante ajoute la contribution de la lumière dans la direction w^i seulement si p est non occlus dans cette direction.

Finalement, nous dérivons la valeur de la constante k pour arriver à l'équation finale d'occultation ambiante. Pour simplifier, nous définissons l'occultation ambiante comme une valeur dans la plage $[0 ; 1]$. Pour un point complètement non occlus, la fonction de visibilité $V(p \leftarrow w^i)$ vaut 1 pour chaque w^i , donc :

$$\begin{aligned} L_0(p_{\text{non occlus}} \rightarrow \omega_0) &= k \int_{\Omega} V(p_{\text{non occlus}} \rightarrow \omega_i) \cos\theta_i d\omega_i \\ &= k \int_{\Omega} \cos\theta_i d\omega_i \\ &= k\pi \end{aligned}$$

Nous voulons que la valeur d'occultation ambiante pour un point non occlus soit de 1, donc :

$$k = \frac{1}{\pi}$$

Et l'équation d'occultation ambiante finale devient :

$$L_0(p \rightarrow \omega_0) = \frac{1}{\pi} \int_{\Omega} V(p \leftarrow \omega_i) \cos\theta_i d\omega_i \quad (6)$$

Dans la pratique, toutes les méthodes de l'occultation ambiante utilisent l'équation 6 pour le calcul du facteur d'occultation ambiante.

L'OA d'un élément p de surface muni d'un vecteur normal n est définie comme l'intégrale sur l'hémisphère de la visibilité :

$$OA(p, n) = \frac{1}{\pi} \int_{\Omega} V(p, \omega) n \cdot \omega d\omega_i \quad (7)$$

Où Ω est l'hémisphère défini par p et w .

L'intégrale OA peut être évaluée par l'intégration de Monte-Carlo [33] :

$$OA(p, n) = \frac{1}{N} \sum_{m=1}^N V \cdot (n \cdot \omega_m) \quad (8)$$

Où N est le nombre de rayons, w^m est une direction choisie uniformément au hasard pour chaque m .

L'effet principal de ce facteur est l'amélioration du réalisme par l'ajout des ombres douces et des auto-ombres. De plus, ce calcul permet d'obtenir un effet intéressant à moindre

CHAPITRE 1 : OCCULTATION AMBIANTE

coût par rapport au calcul de l'illumination globale avec des méthodes de radiosité ou de lancer de rayon type (ray tracing) [12].

1.6.3 Définition d'Occlusion Ambiante

L'Occlusion Ambiante(OA) est souvent utilisée comme une approximation de l'illumination globale. Plus spécifiquement, elle est utilisée comme une approximation de l'ombre de la lumière indirecte. L'occlusion ambiante est grossièrement définie comme la quantité de la géométrie à proximité est obstruée un point. Il est un scalaire compris entre zéro et un qui ne dépend que de la géométrie, et non sur toutes les sources lumineuses. La valeur de l'occlusion ambiante peut ensuite être utilisée à l'échelle de l'éclairage, par exemple le terme ambiant constant dans le modèle d'éclairage de phong. Le résultat est des ombres de proximité entre les objets qui sont proches les uns des autres, et les coins et les crevasses qui sont plus sombres, qui peuvent être interprétés comme ces points obtenant moins de lumière indirecte ou comme des ombres provenant de sources de lumière surfacique [13].



Figure 1. 12 : Occlusion ambiante [14].

1.7 Synthèse

Dans le premier chapitre, nous avons présenté l'illumination globale et ses différents aspects qui sont nécessaires pour parvenir à l'équation de rendu d'illumination globale qui est compliquée et doit être résolue, Les techniques d'illuminations globales est une résolution mais elles sont plus compliquées et demande beaucoup de temps de calcul et pas pratique pour un rendu en temps réel. Pour cela les chercheurs en proposent des techniques

CHAPITRE 1 : OCCULTATION AMBIANTE

d'approximation d'illumination Global pour résoudre l'équation du rendu et accélérer les calculs. Parmi ces méthodes nous avons choisi l'occultation ambiante là où semble être la technique d'approximation d'illumination globale la plus rapide et la plus simple qui rend l'amélioration du réalisme de la scène plus vivante qu'auparavant, que ce soit de manière statique ou dynamique.

1.8 Conclusion

Dans ce chapitre, nous avons mis en évidence l'illumination locale, l'illumination globale et les techniques de chacun, nous avons ensuite présenté l'équation de rendu globale, pour résoudre cette équation on utilise les techniques d'approximation d'illumination globale.

Parmi ces techniques nous nous sommes intéressés à l'occultation ambiante l'occlusion ambiante, puis nous avons présenté une brève historique sur occultation ambiante qui définit la définition mathématique d'OA en détail.

CHAPITRE 2 : OCCULTATION AMBIANTE DANS L'ESPACE ÉCRAN

2 Occultation ambiante dans l'espace écran

2.1 Introduction

L'occlusion ambiante en temps réel était hors de portée jusqu'à ce que Screen Space Ambient Occlusion (SSAO) soit apparu, développée par Crytek en 2007 [20].

L'occlusion ambiante dans l'espace écran (SSAO) c'est un mécanisme simple permettant d'estimer efficacement l'effet d'occlusion ambiante en temps réel, est la forme la plus couramment utilisée, mais c'est aussi la plus basique.

Nous discutons plus en détail la technique de SSAO et présentons les différentes techniques SSAO.

2.2 Les méthodes d'occlusion Ambiante en temps réel

2.2.1 Méthode basée sur l'objet

Ces méthodes calculent l'occlusion ambiante au niveau des objets. Nous présentons les méthodes basées sur l'objet :

2.2.1.1 Ambient Occlusion Fields

Champs d'Occlusion Ambiante c'est une technique en temps réel introduit par Kont kanen et Laine en 2005 pour le calcul de l'occlusion ambiante au niveau des objets. Où chaque objet occlus a un champ de pré-calcul dans son espace environnant qui conduit à une approximation de l'occlusion qui a été causée par un objet de la scène. Où chaque champ qui est une information volumétrique est ensuite utilisé pour la projection d'ombre sur les objets qui ont l'influence de l'occlusion [15].



Figure 2. 1 : Les ombres sont rendues en utilisant la méthode Champs d'occlusion ambiante [15].

CHAPITRE 2 : OCCULTATION AMBIANTE DANS L'ESPACE ECRAN

2.2.1.2 Fast Precomputed Ambient Occlusion for Proximity Shadows

Occlusion ambiante pré calculée rapide pour la proximité des ombres considérée comme une nouvelle méthode d'évaluation de l'occlusion ambiante ou un stockage plus précis, efficace et simple, qui n'utilise qu'une petite quantité de mémoire graphique pour stocker et récupérer les valeurs d'occlusion ambiante qui n'ont aucun rapport avec la complexité du bloqueur ce qui rend cette méthode recommandée pour les besoins en temps réel [16].



Figure 2. 2 : Exemple des ombres de contact [16].

2.2.2 Méthode basée sur les sommets

Ce groupe de techniques OA fonctionne au niveau des sommets, nous présentons les méthodes basées sur le sommet :

2.2.2.1 Dynamic Ambient Occlusion and Indirect Lighting

L'occlusion ambiante dynamique et indirecte est une technique de calcul du transfert de lumière diffuse et montre comment elle peut être utilisée pour calculer l'illumination globale des scènes animées qui a été introduite par Bunnell de Nvidia. Cette technique est suffisamment efficace lorsqu'elle est implémentée sur un GPU rapide pour calculer à la volée les données d'occlusion ambiante et d'éclairage indirect pour chaque image rendue [17].

Cette méthode est si efficace, car elle fonctionne sans calculer la visibilité d'un élément à un autre. Au lieu de cela, il utilise une technique beaucoup plus simple et plus rapide basé sur l'observation approximative pour tenir compte de l'occlusion (blocage) la géométrie [17].

CHAPITRE 2 : OCCULTATION AMBIANTE DANS L'ESPACE ECRAN



Figure 2. 3 : Ajouté un réalisme avec l'occlusion ambiante et éclairage indirect [17].

La scène sur la gauche utilise uniquement l'éclairage direct de l'environnement. La scène du milieu ajoute des ombres douces en utilisant l'occlusion ambiante. La scène sur la droite ajoute un éclairage indirect pour un niveau de réalisme supplémentaire [17].

2.2.2.2 Hardware Accelerated Ambient Occlusion Computation

Le Calcul d'Occlusion Ambiante Accéléré par le Matériel utilise le processeur graphique (GPU) pour accélérer la visibilité entre les points de surface et le calcul des sources lumineuses directionnelles ce qui donne une approximation de l'équation de rendu à l'aide de (GPU) en accumulant des tests de profondeur de chaque fragment de sommet lorsqu'il est capturé à partir d'une direction de source lumineuse, où cette méthode ne nécessite pas d'étape de prétraitement de l'objet scène et aucun coût de perte de mémoire pouvant être utilisé avec une géométrie statique ou dynamique et peut gérer de grands modèles polygonaux [18].



Figure 2. 4 : Stanford dragon sous l'éclairage de la plage dynamique élevée avec l'utilisation d'occlusion ambiante, qui a été calculé sur le matériel graphique [18].

CHAPITRE 2 : OCCULTATION AMBIANTE DANS L'ESPACE ECRAN

2.2.3 Méthode basée sur l'espace d'écran

2.2.3.1 Occultation Directionnelle dans l'Espace Écran

(Screen Space Directional Occlusion(SSDO)). L'occlusion directionnelle de l'espace d'écran permet de calculer l'illumination indirecte locale et la diffusion directionnelle sans l'aide de cartes environnementales, ce qui permet une plus grande variété de nuances en prenant les angles à partir desquels la lumière s'approche de l'objet ainsi que le rebond de la lumière d'un objet derrière l'objet initial en compte [19].



Figure 2. 5 : Images rendues avec la méthode SSDO [19].

2.2.3.2 Occultation ambiante dans l'Espace Écran

(Screen Space Ambient Occlusion (SSAO)) était le type d'occlusion introduit par Vladimir Kajalin tout en travaillant chez Crytek dans le jeu vidéo Crysis en 2007 .Cette technique Permettant d'obtenir une approximation efficace de l'effet d'occlusion ambiante en temps réel. Il utilise le tampon de profondeur de la scène comme une approximation de la géométrie existante, en travaillant avec elle comme si elle était un champ de hauteur. Le voisinage de chaque pixel est échantillonné pour l'occlusion. Si l'échantillon se trouve sous le champ de hauteur, il est considéré comme occlus, sinon il est visible. Alors que la technique était loin d'être exacte, Il a ouvert la voie à d'autres modèles d'occlusion ambiante dans l'espace écran [20].



Figure 2. 6 : Occultation ambiante dans l'Espace Ecran [39].

CHAPITRE 2 : OCCULTATION AMBIANTE DANS L'ESPACE ECRAN

2.2.4 Techniques d'Occultation ambiante dans l'espace d'écran

2.2.4.1 Crytek Occlusion Ambiante

Crytek Occlusion est la première technique d'approximation d'occultation ambiante basé sur l'espace d'écran qui a été introduit en CryEngine 2 par Crytek en 2007 en tant Que version réduite de l'occlusion ambiante dans Crysis, il utilise le tampon de profondeur comme échantillonnage de profondeur à l'intérieur d'une sphère autour du point occlus qui est présenté par un pixel p ou nous calculons le facteur d'occlusion ambiante à par une simple comparaison entre la valeur de profondeur de point d'origine S^z la profondeur d'un échantillon qui est projeté dans la scène S^d après avoir choisi au hasard l'échantillon de noyau résultant de l'approximation du facteur ambiant qui donne beaucoup plus de réalisme sur la scène [21].

Où la base originelle de SSAO est donnée par la formule :

$$AO \approx \frac{1}{N} \sum_N^{n=1} V'(S_n) \quad (9)$$

Avec :

N : noyau d'échantillon qui est le nombre d'échantillons utilisés.

S^n : est un point d'échantillonnage qui a une position aléatoire autour du centre de la sphère p où nous voulons calculer le SSAO.

V' : fonction de visibilité de l'occlusion ambiante mais avec un calcul simplifié cela se fait entre la profondeur de pixel plutôt que la géométrie comme dans AO, qui est donnée comme suivre :

$$V'(s) = \begin{cases} 1 & \text{si } S_d > S_z \\ 0 & \text{sinon} \end{cases} \quad (10)$$

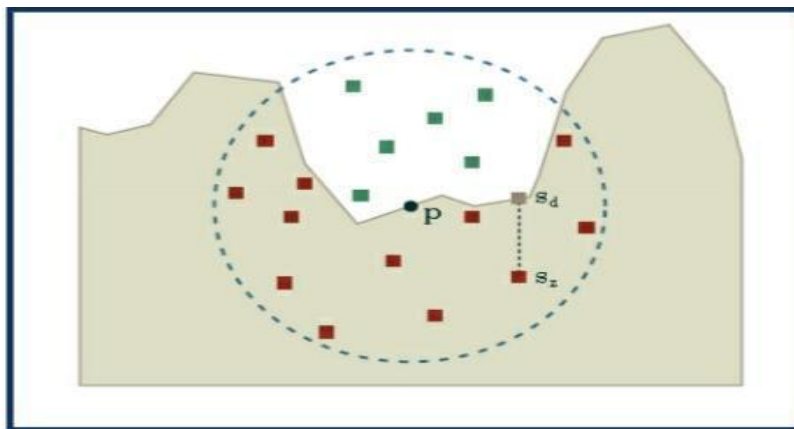


Figure 2. 7 : Crytek OA, les échantillons verts et rouges sont visibles et occultés, respectivement. Un échantillon est aussi montré projeté sur la surface de la scène [22].

CHAPITRE 2 : OCCULTATION AMBIANTE DANS L'ESPACE ECRAN

2.2.4.2 Starcraft 2 Occlusion Ambiante

StarCraft2 a été introduit en 2008 dans la société de développement de jeux de divertissement BLIZZARD qui définit le noyau de l'échantillon, Cette approche utilise une technique similaire à celle trouvée dans Crytek Occlusion ambiante mais avec quelques différences clés. Tout d'abord, les échantillons sont générés dans un hémisphère qui est vrai à la formulation théorique d'OA. En outre, les échantillons sont générés dans l'espace monde, puis transformés en espace écran [23].

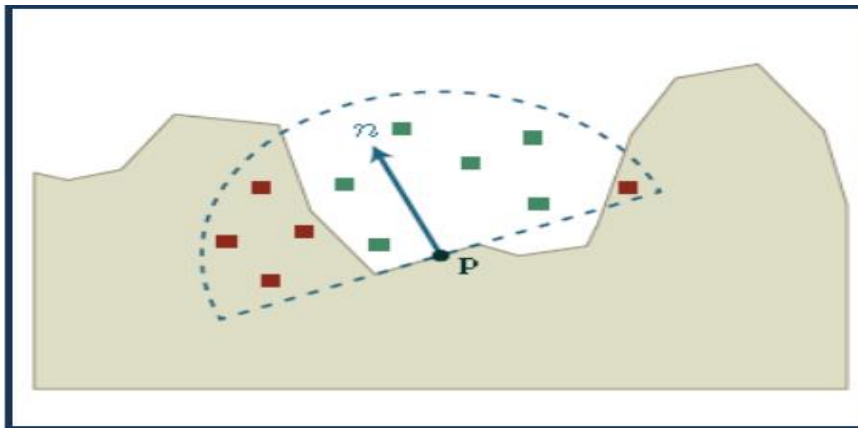


Figure 2. 8 : Starcraft 2 OA, Les échantillons verts et rouges sont visibles et occlus,

2.2.4.3 Horizon Basée Occlusion Ambiante avec Ray Marching

Horizon Based Ambient Occlusion ou HBAO est une technique qui a été introduite en 2008 par Louis Bavoil et Miguel Sainz chez NVIDIA, qui utilise le tampon de profondeur afin de rayonner la marche qui est évidemment dans l'espace de l'écran pour trouver un angle d'horizon visible h . Tout d'abord, une direction du rayon est choisie perpendiculaire à la direction de la vue (la ligne en pointillés gris). Ensuite, plusieurs échantillons sont prélevés le long du dit rayon dans la progression. A chaque étape, la valeur de profondeur est lue. Si la valeur de la profondeur est inférieure à l'étape précédente, l'angle d'horizon est mis à jour. Plusieurs rayons sont coulés et l'angle d'horizon libre moyen [24]. Dans la figure 2.10 les échantillons verts montrent lorsque l'angle de l'horizon est mis à jour. Les échantillons rouges indiquent quand il n'est pas. Le résultat est une mesure très précise de l'horizon libre h [22].

CHAPITRE 2 : OCCULTATION AMBIANTE DANS L'ESPACE ECRAN

2.2.4.5 Alchemy Ambient Obscurance

L'Alchemy Ambient Obscurance est une technique introduite en 2011 par NVIDIA et Vicarious Visions Studio où elle a été implémentée dans leur moteur de jeu Alchemy. Théoriquement, la méthode fonctionne en définissant une fonction d'atténuation intelligente plutôt que la fonction d'occlusion qui simplifie l'intégrale dans le terme AO. Procédé gagne son efficacité du fait qu'il ne coûte pas cher à calculer les vecteurs v [26].

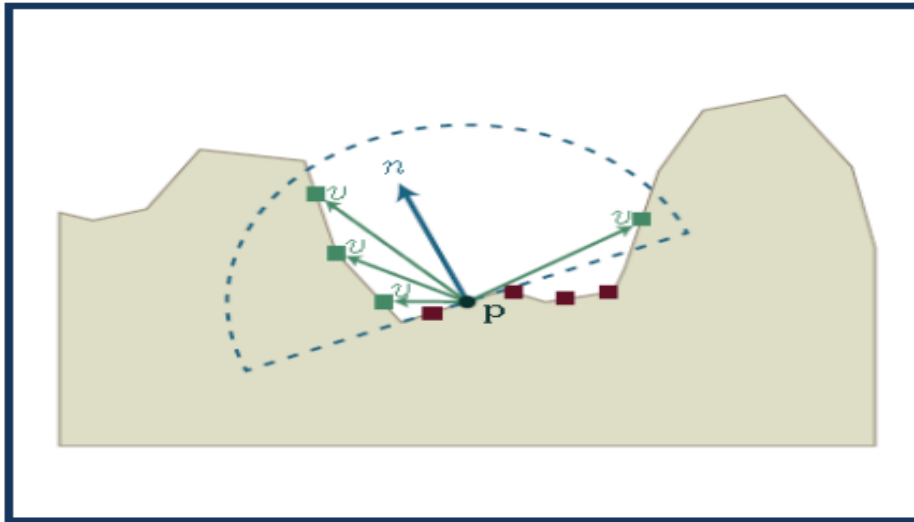


Figure 2. 11: Alchemy AO. Les vecteurs verts v , sont utilisés pour le calcul de la durée OA. Les points d'échantillonnage rouge ont été rejetés parce qu'ils se situent en dehors(ou sur le bord de) l'hémisphère [22].

2.3 Bilan

On présente et explique dans le tableau 2.1, les noms des différentes techniques de SSAO, les avantages et les inconvénients de ces techniques et une liste d'utilisations industrielles.

Technique	Description	Avantages	Inconvénients	Industrie
CryENGINE 2 AO [21]	Faire une approximation d'OA entre les points d'échantillonnage proches visibles et les points occlus. Les échantillons choisis au hasard sont dans une sphère autour de p en coordonnées d'écran. Les auteurs révèlent en outre qu'un système intelligent (en fonction de la	-Résultats rapides.	-Échantillonnage sphérique.	Crysis (2007)

CHAPITRE 2 : OCCULTATION AMBIANTE DANS L'ESPACE ECRAN

	profondeur) nécessaire pour réduire le bruit			
Starcraft II SSAO [23]	Similaire à [21] les différences sont dans les détails. Le plus remarquable est ce point échantillons se trouve dans les coordonnées de monde et pas les coordonnées d'écran.	-Échantillonnage de l'hémisphère (Échantillonnage amélioré).	- Auto-occlusion. - Cas particulier	Starcraft II (2010)
Horizon basée AO avec Ray Marching [24]	AO approximative par la taille de l'horizon non occlus dans l'hémisphère défini par p et n. La méthode ray-marche utilise le tampon de profondeur pour obtenir les estimations de l'horizon. En outre, les auteurs ont suggéré l'étape du ray-marche de choisir des directions au hasard pour éviter le bruit systématique.	-Meilleur résultat d'ombrage.(en raison de l'hypothèse du champ de hauteur).	-Directions des rayons aléatoires (en raison de l'utilisation du hasard vecteurs).	Tomb Raider 2013
Obscurcissement volumétrique [25]	Les auteurs définissent le terme volumétrique obscurcissement et l'utilisent pour trouver AO. Ils utilisent le rapport du visible au volume occlus autour p. Une autre nouvelle idée est de combiner les calculs AO d'une petite sphère et une grande sphère de l'échantillon. Cette technique reflète à la fois faible échelle ainsi que des détails à grande échelle.	- Meilleure approximation du volume (en raison des échantillons de ligne).	- Performances faibles lorsque il s'agit d'échantillons de zone. - Absence d'objets fins influence.	
Alchemy AO [26]	Une méthode unique qui utilise une fonction pour obtenir une formule simple pour AO.	- Intuitif - Efficace - Robuste	- Le filtrage bilatéral provoque un faible niveau de rendement.	

CHAPITRE 2 : OCCULTATION AMBIANTE DANS L'ESPACE ECRAN

			- Écart d'échantillon.	/
--	--	--	------------------------	---

Tableau 2. 1 : Comparaison entre les techniques de SSAO

2.4 Les filtres pour SSAO

Quand nous utilise les techniques d'occultation ambiante dans l'espace écran à l'aide d'échantillonnage de profondeur aléatoires produit un effet connu sous le nom de bruit qui peut être réduit à l'aide de techniques de filtrage .En peut citer :

2.4.1 Filtre Anti Aliasing

L'Anti Aliasing est une technique utilisée pour minimiser la propagation du crénelage et supprimer les effets d'escaliers qui apparaissent sur certaines lignes obliques en raison de la nature discrète des pixels. En attendant d'avoir des écrans de résolution infinie, ou suffisamment élevée pour que les pixels ne soient plus discernables à l'œil nu, il faut se rabattre sur ces algorithmes capables d'adoucir les contours sans pour autant trop flouter l'image dans son ensemble.

Plusieurs méthodes d'anti aliasing en peuvent citer :

FXAA (Fast Approximate Anti-Aliasing), SMAA (Subpixel Morphologique Anti-Aliasing), MSAA (MultiSample Anti-Aliasing), et TXAA (Temporal Anti-Aliasing). La raison de ce choix de méthodes anti-aliasing est que chacun a ses avantages et ses inconvénients, tant en termes de qualité et de performance image. Il n'y a pas de méthode unique anti aliasing optimale recommandée, car elle dépend en grande partie sur une combinaison des capacités de votre système et goût personnel [40].



Figure 2. 12 : Comparaison entre l'entrée (à gauche) sans filtre anti aliasing. (Au milieu) avec filtre anti aliasing (FXAA) et (à droite) avec filtre anti aliasing (SMAA) [41].

CHAPITRE 2 : OCCULTATION AMBIANTE DANS L'ESPACE ECRAN

2.4.2 Filtre Gaussien

Le filtre gaussien est un filtre passe-bas non uniforme, étudié dans le traitement de l'image et la vision par ordinateur, qui permettent d'améliorer beaucoup plus les images du fait qu'un passage de l'intervalle de continuité à l'intervalle discret ou à l'aléatoire doit être fait en infographie ce qui nous laisse avec une quantité assez importante de bruit connu sous le nom de flou qui peut être réduite par lissage[42].

Le filtre gaussien basé sur la formule de la fonction gaussienne la plus générale est :

$$g(x,y) = \frac{1}{2\pi\sigma^2} e^{-\left(x^2+y^2\right)/2\sigma^2} \quad g(x,y) = \frac{1}{2\pi\sigma^2} e^{-\left(x^2+y^2\right)/2\sigma^2} \quad (11)$$

Avec :

σ^2 est la variance du filtre gaussien respectivement σ comme écart type.

(x,y) sont les coordonnées cartésiennes de l'image.



Figure 2. 13 : Comparaison entre : (à gauche) sans filtre Gaussien et (à droite) avec filtre Gaussien [42].

2.4.3 Filtre Bilatéral

Un filtre bilatéral est un filtre de lissage non linéaire, préservant les bords et réduisant le bruit pour les images. Développé par Tomasi et Manduchi en 2009. Il remplace l'intensité de chaque pixel par une moyenne pondérée des valeurs d'intensité des pixels voisins. Ce poids peut être basé sur une distribution gaussienne. Fondamentalement, les poids dépendent non seulement de la distance euclidienne des pixels, mais également des différences radiométriques [43].

Le filtre bilatéral est défini comme :

$$I^{\text{filtered}}(x) = I^{\text{filtered}} \frac{1}{w_p} \sum_{x_i \in \Omega} I(x_i) \text{fr}(\|I(x_i) - I(x)\|) g_s(\|x_i - x\|)$$
$$\frac{1}{w_p} \sum_{x_i \in \Omega} I(x_i) \text{fr}(\|I(x_i) - I(x)\|) g_s(\|x_i - x\|) \quad (12)$$

Et terme de normalisation, $w_p w_p$ est défini comme :

CHAPITRE 2 : OCCULTATION AMBIANTE DANS L'ESPACE ECRAN

$$w_p w_p = \sum_{x_i \in \Omega} f_r(\|I(x_i) - I(x)\|) g_s(\|x_i - x\|) \sum_{x_i \in \Omega} f_r(\|I(x_i) - I(x)\|) g_s(\|x_i - x\|) \quad (13)$$

Où :

I_{filtered} : est l'image filtrée.

I : est l'image d'entrée d'origine à filtrer.

x : sont les coordonnées du pixel courant à filtrer.

Ω : La fenêtre est-elle centrée sur x , donc $x_i \in \Omega$ est un autre pixel.

f_r : est le noyau de plage de lissage des différences d'intensité (cette fonction peut être une fonction gaussienne).

g_s : est le noyau spatial (ou domaine) de lissage des différences de coordonnées (cette fonction peut être une fonction gaussienne).

w_p : est le facteur de normalisation.



Figure 2. 14 : Comparaison entre : (à gauche) avec filtre Bilatéral et (à droite) sans filtre Bilatéral [43].

2.5 Bump Mapping

Le Bump mapping [45] est une technique de mappage de texture en infographie introduite par James Blinn en 1978 [46], permettant de simuler des bosses et des rides à la surface d'un objet. Ceci est réalisé en perturbant les normales de surface de l'objet et en utilisant la normale perturbée lors des calculs d'éclairage. Le résultat est une surface apparemment bosselée plutôt qu'une surface lisse bien que la surface de l'objet sous-jacent ne soit pas modifiée.

Le bump mapping est une technique d'infographie permettant de rendre une surface rendue plus réaliste en simulant de petits déplacements de la surface. Cependant, contrairement au

CHAPITRE 2 : OCCULTATION AMBIANTE DANS L'ESPACE ECRAN

mapping de déplacement, la géométrie de la surface n'est pas modifiée. Au lieu de cela, seule la normale à la surface est modifiée comme si la surface avait été déplacée. La normale de surface modifiée est ensuite utilisée pour les calculs d'éclairage (en utilisant, par exemple, le modèle de réflexion de Phong) donnant l'apparence de détails au lieu d'une surface lisse [47]. Le bump mapping est beaucoup plus rapide et consomme moins de ressources pour le même niveau de détail par rapport au mappage de déplacement car la géométrie reste inchangée [47].

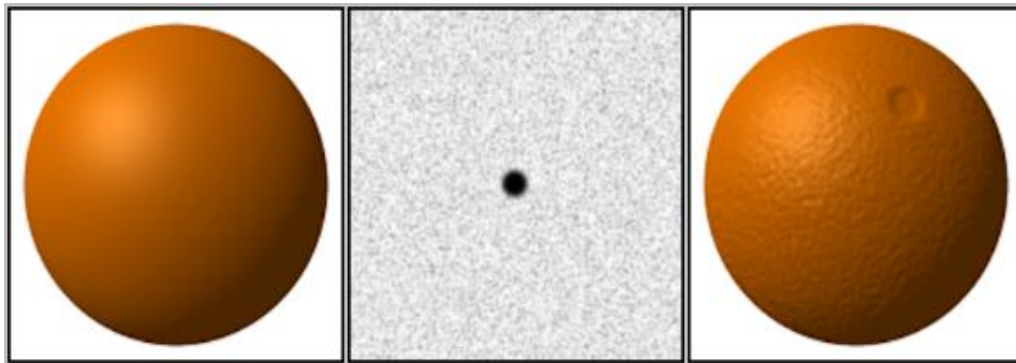


Figure 2. 15 : (À gauche) Une sphère sans bump mapping. (Au milieu) Une bump map à appliquer à la sphère. (À droite) La sphère avec la bump mapping appliquée [45].

La surface est marbrée ressemblant à une orange. Les reliefs permettent d'obtenir cet effet en modifiant la façon dont une surface éclairée réagit à la lumière, sans modifier la taille ou la forme de la surface.

- **Algorithme de bump mapping**

Nous utilisons une carte hauteur/bump (comme l'image montrée à la fin) pour simuler le déplacement de surface donnant la normale modifiée comme indiqué par Blinn. Ainsi, une bump map est une texture dans laquelle sont stockés les déplacements de la surface. La hauteur réelle des déplacements dépendra de la façon dont nous utilisons la bump map (seule la plage $[0,1]$ y est stockée). Dans les calculs d'éclairage, nous utilisons cette normale perturbée. La perturbation à ajouter à la normale est la première dérivée de la bump map et peut être calculée comme suit [46],

CHAPITRE 2 : OCCULTATION AMBIANTE DANS L'ESPACE ECRAN

$$\vec{N}_x = \frac{-\frac{\partial F}{\partial x}}{\sqrt{\frac{\partial F^2}{\partial x} + \frac{\partial F^2}{\partial y} + 1}}$$

$$\vec{N}_y = \frac{-\frac{\partial F}{\partial y}}{\sqrt{\frac{\partial F^2}{\partial x} + \frac{\partial F^2}{\partial y} + 1}}$$

$$\vec{N}_z = \frac{1}{\sqrt{\frac{\partial F^2}{\partial x} + \frac{\partial F^2}{\partial y} + 1}}$$

N reflète les modifications stockées dans la carte en relief. Afin de calculer les dérivées partielles mentionnées dans la formule, la méthode des différences finies est utilisée. La différence finie consiste simplement à prendre une différence entre deux échantillons suffisamment proches l'un de l'autre. Par exemple, trouver la dérivée partielle de F par rapport à x peut être calculé avec $h = 1$ et cela nous donne [46],

$$\frac{\partial F}{\partial x} = F(x + 1) - F(x)$$

Nous faisons cela pour une fonction 2-D telle que $F(u,v)$ auquel cas la dérivée w.r.t u est $F(u+1,v) - F(u,v)$ et la dérivée w.r.t. v est $F(u,v+1) - F(u,v)$.

Avant d'effectuer un calcul d'éclairage pour chaque pixel de la surface de l'objet, l'algorithme effectue les étapes suivantes :

- Recherchez la hauteur dans la carte des hauteurs qui correspond à la position sur la surface.
- Calculez la surface normale de la carte de hauteur en utilisant la méthode des différences finies comme indiqué ci-dessus.
- Combinez la normale de surface de l'étape 2 avec la vraie normale de surface ("géométrique") de sorte que la normale combinée pointe dans une nouvelle direction.
- Calculez l'interaction de la nouvelle surface "bosselée" avec les lumières de la scène à l'aide du modèle de réflexion de Phong.

Le résultat est une surface qui semble avoir une profondeur réelle. L'algorithme garantit également que l'apparence de la surface change lorsque les lumières de la scène sont déplacés.

CHAPITRE 2 : OCCULTATION AMBIANTE DANS L'ESPACE ECRAN

Comme on peut le remarquer à partir de l'algorithme, la surface elle-même est toujours plate, cette méthode n'affectera que les calculs d'éclairage et non la géométrie réelle [46].

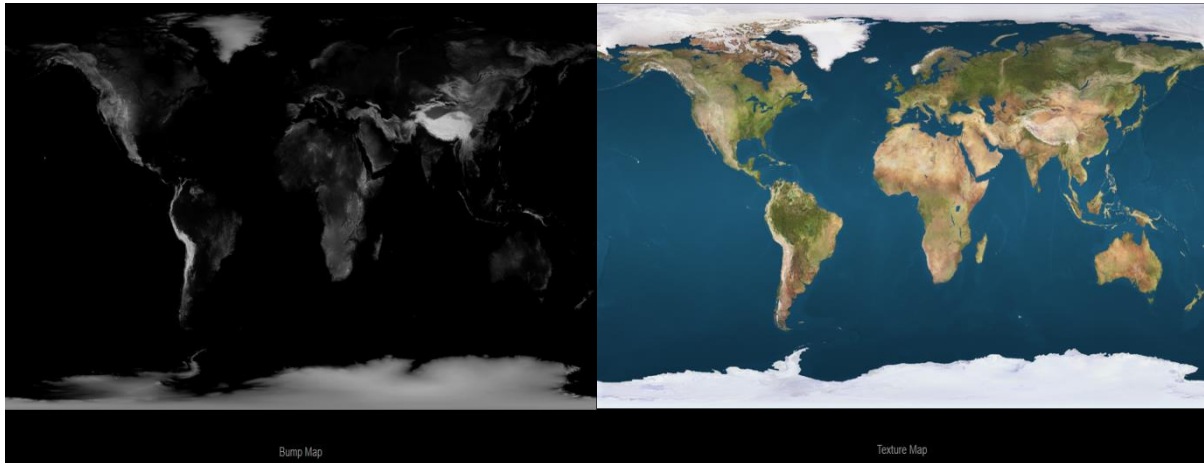


Figure 2. 16: (a gauche) bump mapping, (a droite) texture map[46]

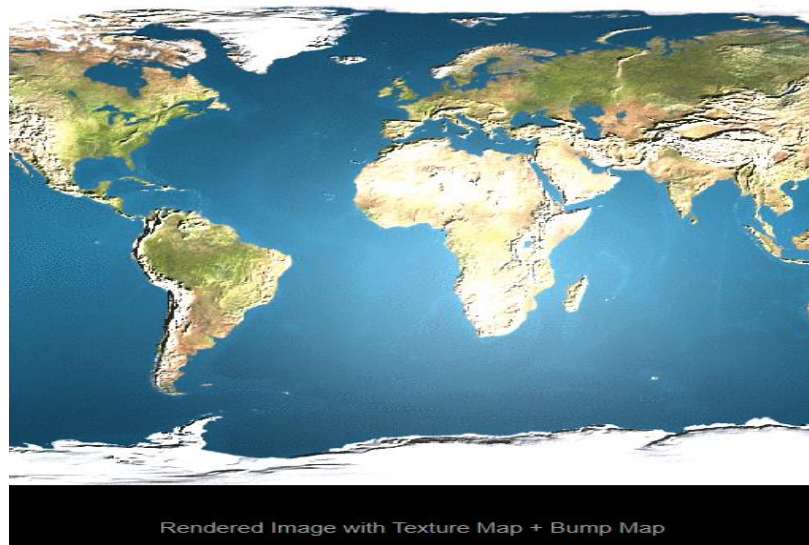


Figure 2. 17: Image rendue avec Texture Map + Bump Map[46]

2.6 Synthèse

Au cours de ce chapitre, nous avons présenté la plupart des travaux de l'occultation ambiante dans l'espace écran.

Parmi les méthodes basées sur l'espace d'écran la technique d'occultation ambiante dans l'espace écran (SSAO) qui permet d'obtenir une approximation efficace de l'effet d'occlusion ambiante en temps réel. Cette technologie utilise le tampon de profondeur et le

CHAPITRE 2 : OCCULTATION AMBIANTE DANS L'ESPACE ECRAN

tampon de normale de scène pour déterminer la quantité d'occlusion, mais au lieu d'utiliser le tampon de normale nous avons proposé d'utiliser les normales de bump mapping qui nous parlerons en détail.

Nous avons également discuté de la technique de filtrage du bruit et du crénelage qui peut entraîner une assez bonne douceur connue sous le nom de flou pour réduire la quantité ou même éliminer les artefacts visuels que l'œil humain peut distinguer avec une comparaison résumée qui récapitule leurs différences en ce qui concerne temps, mémoire et résultat efficacité.

2.7 Conclusion

Dans ce chapitre, nous avons commencé par présenter les méthodes d'occlusion ambiante en temps réel, ensuite nous avons exposé la méthode de l'occultation ambiante dans l'espace écran (SSAO), puis nous avons discuté des différentes méthodes de SSAO existantes, nous avons également proposé un bilan des différentes méthodes de SSAO existantes, ces avantages et ces inconvénients. Apres nous avons discuté de la technique de filtrage du bruit et du crénelage qui peut entraîner une assez bonne douceur connue sous le nom de flou pour réduire la quantité ou même éliminer les artefacts visuels que l'œil humain peut distinguer avec une comparaison résumée qui récapitule leurs différences en ce qui concerne le temps, le mémoire et l'efficacité des résultats. A la fin nous parlons en détaille la technique de bump mapping et leur algorithme.

CHAPITRE 3 : MISE EN ŒUVRE, RÉSULTATS ET PERSPECTIVE

3 Mise en œuvre, résultats et perspective

3.1 Introduction

Dans ce chapitre nous commençons par les objectifs et description de notre projet et la conception générale. Ensuite la réalisation de notre application par décrire les outils matériels et logiciels, et la structure de notre application que nous avons utilisé, puis les algorithmes et à la fin nous avons présenté les résultats.

3.2 Objectifs et description

Notre objectif principal de ce projet est d'implémenter la technique d'occlusion ambiante dans l'espace écran en temps réels on se basant sur la technique du Bump Mapping qui permettant de rendre une surface plus réaliste, son principe est de perturber les normales de surface de l'objet et en utilisant la normale perturbée lors des calculs d'éclairage. On utilise le GPU et les shaders pour une telle tâche.

La technique de SSAO permet d'ajouter le réalisme dans la scène 3D, mais l'utilisation de l'échantillonnage de profondeur aléatoires produit un effet connu sous le nom de bruit qui peut être réduit à l'aide de l'utilisation des filtres particuliers.

3.3 Conception Générale

La conception générale est discutée où nous divisons la réalisation d'objectifs en plusieurs étapes où chaque étape a un rôle important.

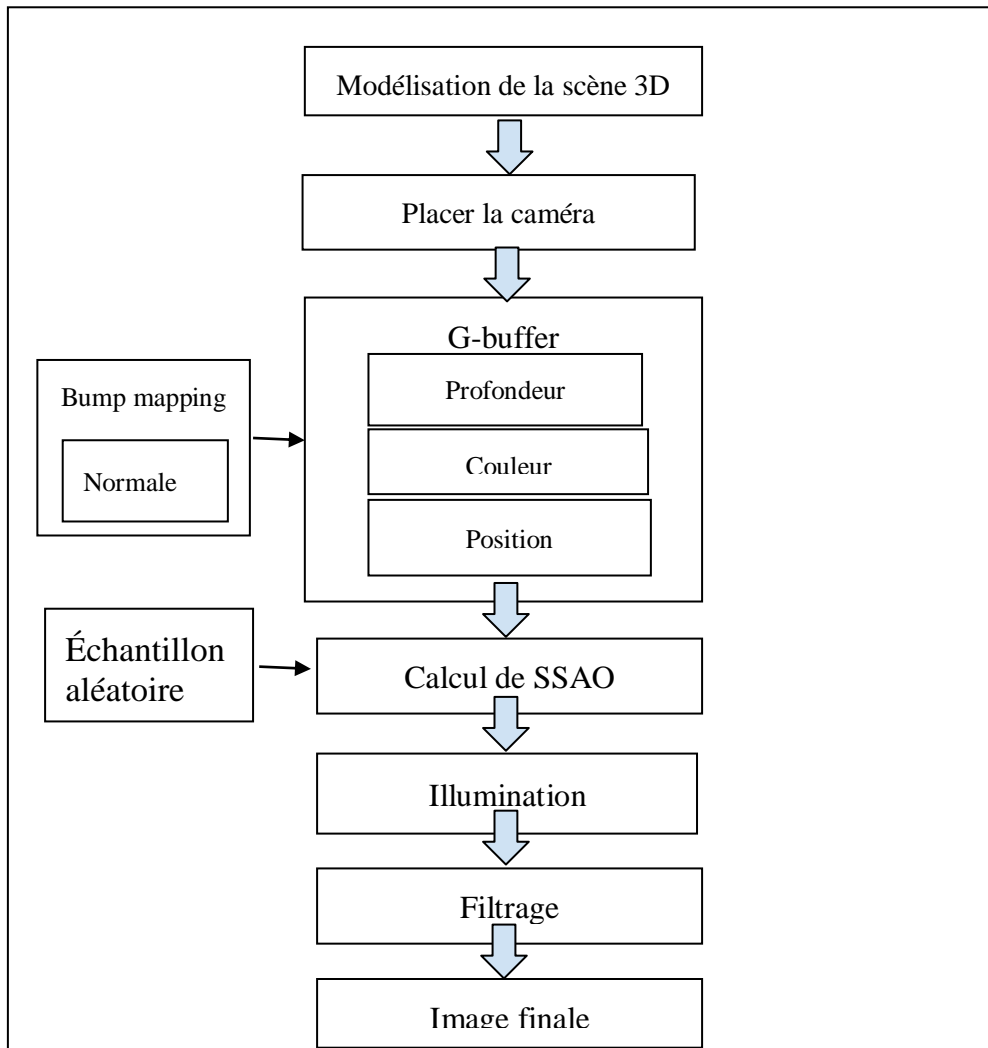


Figure 3. 1 : Conception générale.

3.3.1 Modélisation de la scène 3D

La première étape est la modélisation, nous modélisons la scène 3D ou utilisons des modèles 3D modélisés par les modelleurs à partir d'un fichier de format (fichier .OBJ).

OBJ est un format de fichier contenant la description d'une géométrie 3D. Il a été défini par la société Wavefront Technologies dans le cadre du développement de son logiciel d'animation Advanced Visualizer. Ce format de fichier est ouvert et a été adopté par d'autres logiciels 3D (tels que 3D Turbo de iluac software Poser de e-frontier, Maya de Autodesk, Blender, MeshLab, 3D Studio Max, LightWave 3D de Newtek, GLC Player etc) pour des traitements d'import / export de données[49].

CHAPITRE 3 : MISE EN ŒUVRE, RESULTATS ET PERSPECTIVE

3.3.2 Placer la caméra

On doit placer la caméra pour extraire les informations nécessaires (position, profondeur et couleur) pour calculer le SSAO.

3.3.3 Calculer le G-Buffer

L'une des choses les plus importantes qui a permis de calculer l'occlusion ambiante dans l'espace écran est le G-Buffer où il stocke toutes les informations de géométrie (positions, couleurs et profondeurs) dans des textures spécifiées attachées à le même FBO pour une utilisation ultérieure.

Le G-Buffer est utilisé soit pour approximer le facteur d'occlusion ambiante, soit pour calculer l'illumination finale à l'aide de l'un des modèles d'illumination locale simplifiée où elle est considérée comme différée plutôt que directe en raison de l'utilisation du G-Buffer qui est considérée comme une passe de géométrie.

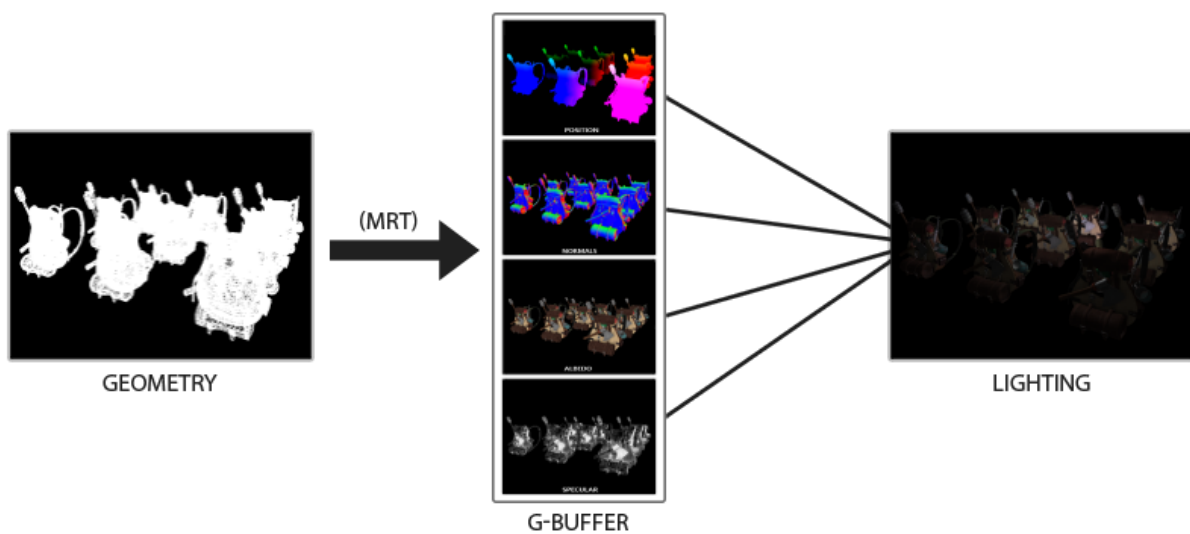


Figure 3. 2 : G-buffer [50]

3.3.4 Bump mapping

La terminologie appropriée pour la technique de mappage en relief (bump mapping) que nous utilisons est appelée mappage normal. La raison en est que nous utilisons une texture spéciale appelée carte normale qui est essentiellement une table de recherche pour les normales de surface. Chaque pixel de cette carte normale indique la direction de la lumière pour le pixel correspondant sur la carte de couleurs de texture [51].

CHAPITRE 3 : MISE EN ŒUVRE, RESULTATS ET PERSPECTIVE

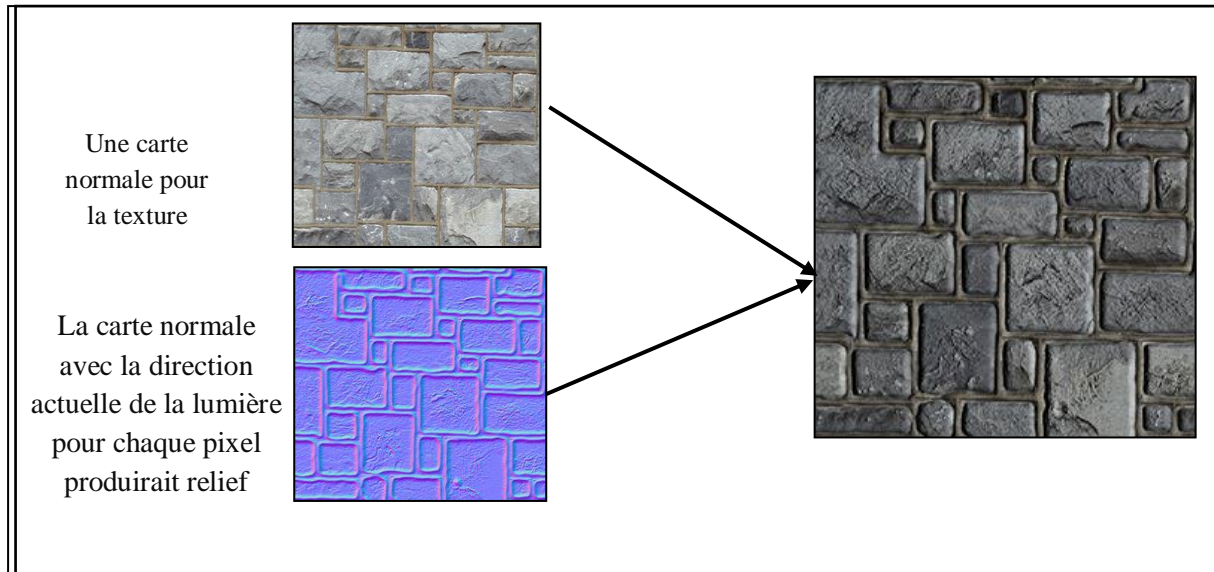


Figure 3. 3 : Bump Mapping [51].

3.3.5 Création des échantillons

L'approximation du facteur d'occlusion ambiante dans l'espace de l'écran nécessite des échantillons de noyau qui sont un nombre limité de points qui entourent un point, où ces points sont générés de manière à former un hémisphère orienté par la normale à la surface qui pointe dans la direction Z.

L'utilisation de ces techniques uniquement ne nous fournit pas d'excellents résultats sans avoir un nombre élevé d'échantillons tout en utilisant une faible quantité d'échantillons.

Ce qui entraîne également l'effet de bande qui est perceptible à la sensation de l'œil humain. Nous avons utilisé des vecteurs de rotation aléatoires pour faire pivoter les échantillons le long de l'axe z ces vecteurs aléatoires qui sont générés et enregistrés dans la texture connue sous le nom de texture de bruit sont utilisés pour obtenir d'excellents résultats sans avoir l'effet de bande qui a été remplacé par l'effet de bruit qui est moins perceptible par la vision humaine.

3.3.6 Calculer le SSAO

Pour chaque fragment sur un quadrilatère qui couvre l'écran, nous calculons un facteur d'occlusion basé sur les valeurs de profondeur environnante du fragment. Le facteur d'occlusion est ensuite utilisé pour réduire ou annuler la composante d'éclairage ambiant du fragment. Le facteur d'occlusion est obtenu en prenant plusieurs échantillons de profondeur dans un noyau d'échantillons d'hémisphère entourant la position du fragment et en comparant

CHAPITRE 3 : MISE EN ŒUVRE, RESULTATS ET PERSPECTIVE

chacun des échantillons avec la valeur de profondeur du fragment actuel. Le nombre d'échantillons qui ont une valeur de profondeur supérieure à la profondeur du fragment représente le facteur d'occlusion.

En fin en sauvegardant le résultat SSAO dans un Frame Buffer pour l'afficher ensuite sur l'écran.

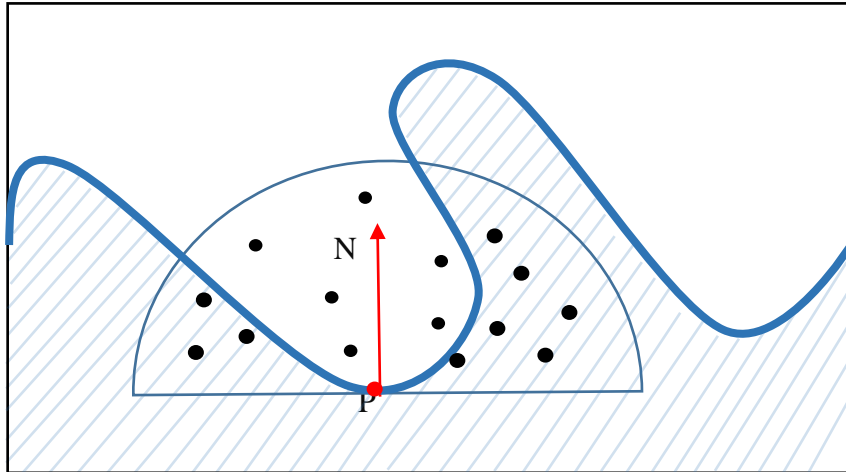


Figure 3. 4 : Échantillonnage dans un hémisphère

3.3.7 Illumination

Nous avons utilisé le modèle d'illumination de Phong pour éclairer notre scène car il est rapide et simple à calculer, il combine trois éléments : la lumière ambiante, la lumière diffuse (modèle lambertien) et la lumière spéculaire.

La formule finale de Phong est la somme des (ambiante, diffuse, spéculaire) :

$$I = I_a + I_d + I_s$$

3.3.8 Application du Filtre

En raison de la rotation aléatoire des échantillons le long de l'axe z en utilisant les vecteurs aléatoires, l'effet de bande qui est l'artefact le plus observable est supprimé et remplacé par un bruit moins perceptible à l'œil humain. En ordre pour éliminer ce type de bruit, des techniques de traitement d'image sont nécessaires.

Il y a plusieurs types de filtres, parmi eux le filtre original qui est définie comme le filtre bilatéral qui est une technique de filtrage puissante dans la préservation des contours. Pour

CHAPITRE 3 : MISE EN ŒUVRE, RESULTATS ET PERSPECTIVE

implémenter ce filtre, nous aurions besoin d'un poids bidimensionnel que nous pouvons obtenir à partir d'une équation de courbe gaussienne à deux dimensions. Mais le problème avec cette approche est toutefois qu'il devient rapidement lent et affecte les performances.

Heureusement, l'équation gaussienne a une propriété propre qui nous permet de séparer l'équation bidimensionnelle en deux équations plus petites : une qui décrit les poids horizontaux et l'autre qui décrit les poids verticaux. Nous allons d'abord faire un filtre horizontal avec les poids horizontaux sur une texture entière, puis sur la texture résultante, puis faire un filtre vertical et afficher l'image finale.

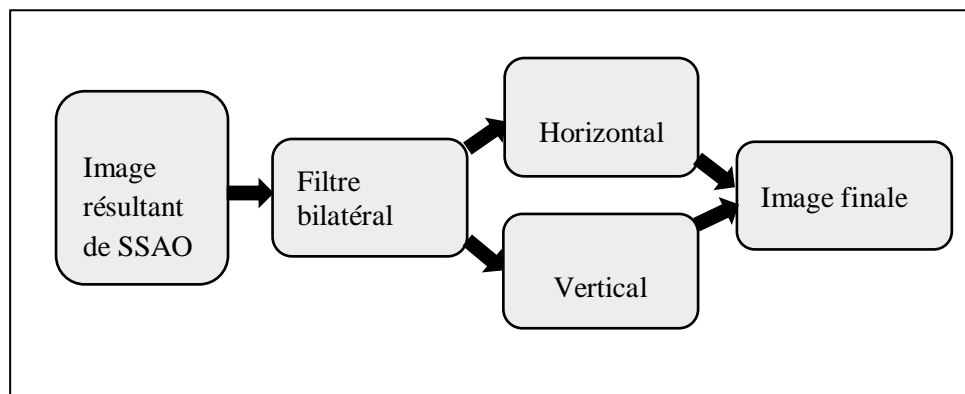


Figure 3. 5 : Filtrage.

3.4 Réalisation

3.4.1 Configurations matérielles et logiciels utilisés

Pour une meilleure évaluation de notre application, nous l'avons testée sur une station graphique qui possède les configurations suivantes :

3.4.1.1 Configuration Matérielle

- Machine : Hp.
- Système d'exploitation : Windows 8.1, 64bits.
- Processeur : Intel(R) Core(TM) i5-5300U CPU@2.30 GHz2.30GHz.
- Mémoire (RAM) : 4,00 Go.
- Carte graphique : Intel(R) HD Graphics 5500

3.4.1.2 Configuration logicielle

- Langage de programmation : Visual Studio C++ 2010.
- API graphique : OpenGL 4.4 (220 ext) - Intel(R) HD Graphics 5500
- GLSL (The OpenGL Shading Language)

CHAPITRE 3 : MISE EN ŒUVRE, RESULTATS ET PERSPECTIVE

3.4.2 Structure de l'application

Afin de calculer SSAO et d'utiliser des techniques de filtrage, nous avons utilisé des shaders (un vertex, un fragment), il est nécessaire de l'insérer au sein d'une application OpenGL. Ces fichiers seront ensuite chargés dans une application OpenGL, puis compilés avant de pouvoir être utilisés.

Il exige 4 passes de rendu : un premier passe de rendu pour sauvegarder les informations géométriques (position, couleur, profondeur) dans les cartes de texture, et un deuxième rendu pour calculer le facteur SSAO, et les deux derniers rendus pour filtrer l'image sous forme horizontale et verticale. Les étapes de l'algorithme principal de notre application sont comme suit :

3.4.2.1 Premier shader : Création du G-Buffer

L'utilisation de espace écran pour approximer le facteur AO nécessite l'utilisation de textures, ce qui est un fait connu où il s'agit en fait d'un ensemble d'images contenant les informations de géométrie stockées à l'aide du shader G-buffer en créant un frame buffer contenant ces différentes textures nécessaires : Positions, Profondeur et on ajoute les normales qui pré-calculées avec la technique de bump mapping, où ces informations sont obtenues à partir du modèle chargé.

La position, la couleur et les normales du bump mapping sont des informations d'entrées pour le vertex shader qui permet de convertir les sommets dans l'espace monde vers l'espace écran (en utilisant la matrice monde/view/projection (MVP), puis transmet la profondeur, position, couleur, et la normale qui pré-calculées par le bump mapping de chaque vertex vers le fragment shader comme des variables varying.

Dans le fragment shader les informations géométriques (G-buffer) sont des entrées, on va remplir les textures de la couleur et la position, la profondeur dans les cartes de texture. Les sorties du premier passe utilisé par le deuxième shader comme variable de type Uniform pour être utilisées pour calculer le facteur SSAO.

3.4.2.2 Deuxième shader : calculer le SSAO

Afin de calculer le facteur SSAO, plusieurs étapes sont nécessaires dans le shader, où elles sont expliquées ci-dessous.

CHAPITRE 3 : MISE EN ŒUVRE, RESULTATS ET PERSPECTIVE

● Le noyau de l'échantillon

Tout d'abord on va envoyer un vecteur 2D du CPU vers le vertex shader SSAO comme variable Uniform, tel que le vecteur est égale à la largeur et la longueur de la fenêtre d'affichage divisé par n qui représente le nombre d'échantillons.

On calcul les positions des sommets en espace 3D, puis en multipliant ces positions par le vecteur 2D pour obtenir les positions dans les blocs. Ensuite on va faire un échantillonnage du fragment dans un hémisphère de manière aléatoire en utilisant un vecteur de noyau échantillon 16, 32 et 64.

● Calculer le facteur SSAO

Nous calculons le facteur d'occlusion ambiante en utilisant l'échantillonnage qui est implémenté dans le GPU avec un programme shader. On va calculer la position des points échantillonnés à partir à la position de centre de noyau de l'échantillon, puis on obtient les profondeurs des points échantillonnées à partir à la position de point échantillonnée et l'information existant dans la texture de profondeur.

On calcul la distance de point échantillon et comparer avec la valeur stockée dans tampon de profondeur .Si la distance du point échantillon supérieure à la valeur stockée alors le point échantillon dans une situation d'occlusion.

Après le calcul de facteur SSAO nous appliquons l'éclairage de phong .Nous appliquons ce modèle pour deux cas différents :

Nous appliquons l'éclairage phong avec éclairage direct qui utilise une lumière ambiante, ou nous utilisons l'éclairage phong avec éclairage indirect qui utilise le facteur d'SSAO au lieu de lumière ambiante.

Enfin nous enregistrons le résultat dans le Frame Buffer pour le filtrer et l'afficher sur l'écran.

La formule complète est :

$$I = I_a \cdot K_a + I_d \cdot K_d (\vec{L} \cdot \vec{N}) + I_s \cdot K_s (\vec{R}_n \cdot \vec{V})^\alpha$$

● Suppression du bruit

L'utilisation d'échantillonnage de SSAO produit des bruits et pour qualité visuelle et résultats acceptables, nous avons besoin d'un nombre minimum d'échantillons.

CHAPITRE 3 : MISE EN ŒUVRE, RESULTATS ET PERSPECTIVE

On peut éliminer ce bruit par la technique de filtrage, nous utilisons le filtre bilatéral qui nécessite filtre vertical et filtre horizontal pour l'appliquer.

3.4.2.3 Troisième shader : Filtre horizontale

Le filtre horizontal s'applique sur les lignes (x) plutôt les colonnes (y) de la coordonnée de texture, puis on va diviser la valeur appliquée sur la largeur de fenêtre d'affichage. Le résultat est un vecteur 2D pour calculer la profondeur et le nouveau poids d'un pixel.

la valeur absolue de la profondeur ancienne moins la valeur de profondeur calculé, si le résultat inférieure strictement qu'un facteur alors calculer la valeur poids d'un pixel, puis on va diviser cette valeur sur la somme de la taille de noyau pour obtenir la moyenne de valeur des poids d'un pixel voisins et remplacé par la valeur ancien .

3.4.2.4 Quatrième shader : filtre verticale

Dans le filtre verticale on utilise les mêmes étapes que le filtre horizontal sauf l'horizontal applique sur les lignes (x) pour diviser la valeur sur la largeur de fenêtre d'affichage, et le vertical appliqué sur les colonnes (y) pour diviser la valeur sur la longueur de fenêtre d'affichage.

CHAPITRE 3 : MISE EN ŒUVRE, RESULTATS ET PERSPECTIVE

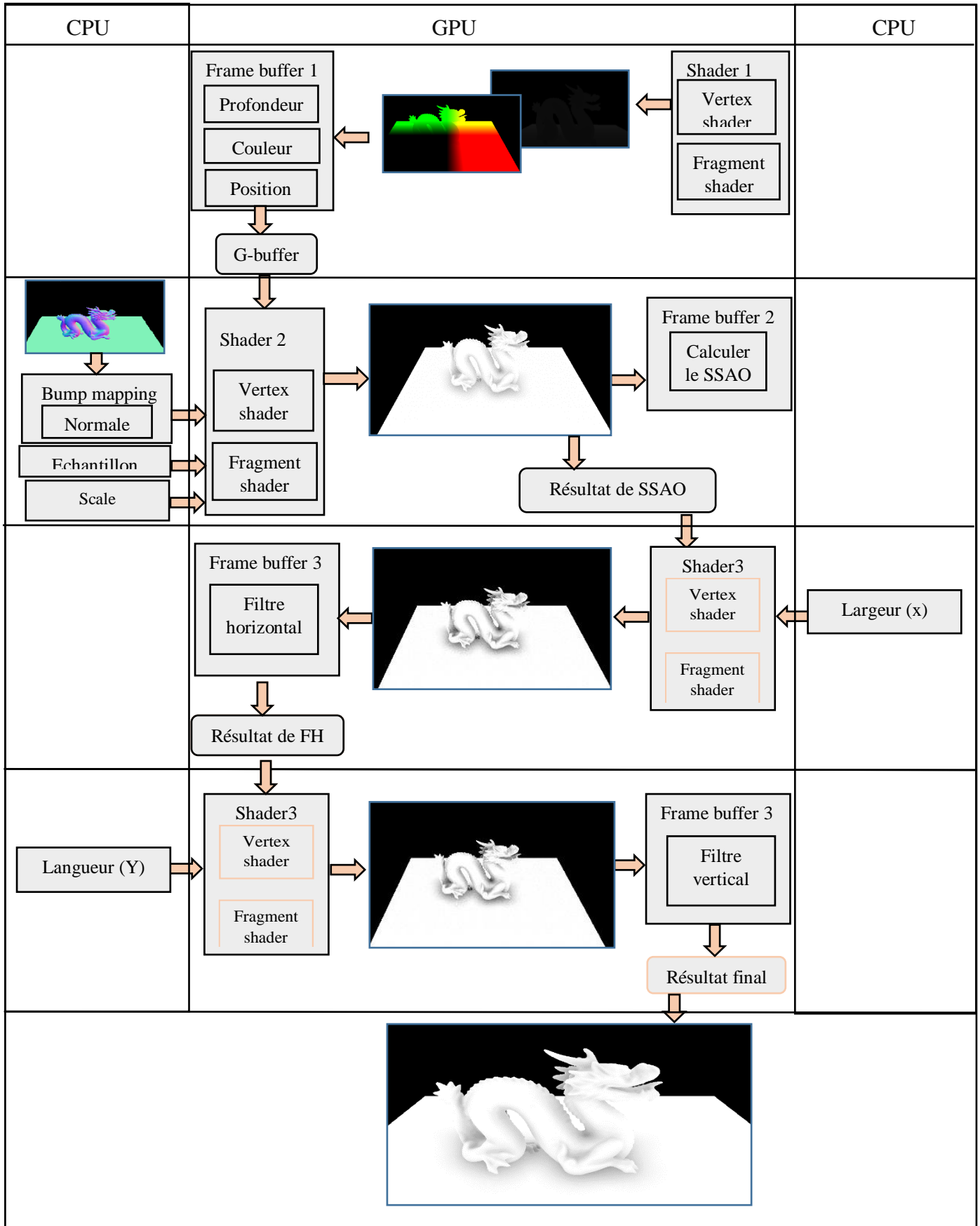
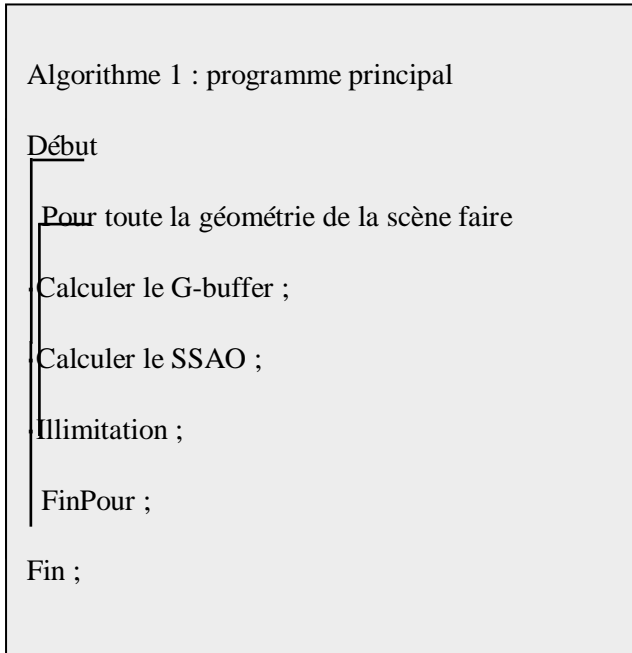


Figure 3. 6 : Structure général.

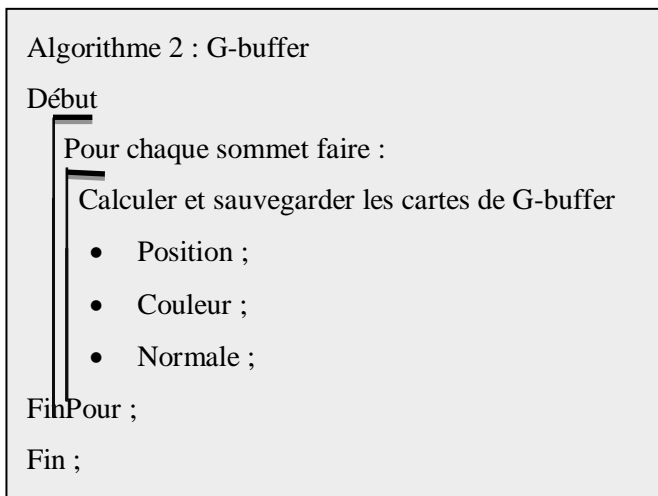
CHAPITRE 3 : MISE EN ŒUVRE, RESULTATS ET PERSPECTIVE

3.5 Algorithmes

- Algorithme 1 : représente le programme principal



- Algorithme 2 : représente le G-buffer



CHAPITRE 3 : MISE EN ŒUVRE, RESULTATS ET PERSPECTIVE

- Algorithme 3 : représente le calcul de SSAO

Algorithme 3 : SSAO

Début

```
Tab : Samples [32] ; // noyau d'échantillon
Pour chaque sommet faire :
    SSAO ← 0;
    i=0;
    Tanque (i < 32) faire
        TexCoord = clamp(ScaleRotationMatrix * Samples[i] + gl_TexCoord[0].st,0.0,0.999999);
        depth = texture2D(DepthBuffer, TexCoord).r;
        position = ProjectionBiasMatrixInverse * vec4(TexCoord, depth, 1.0);
        position.xyz /= position.w ;
        P2P = position.xyz - Position.xyz;
        Distance2 = dot(P2P, P2P);
        Weight = 1.0 - Distance2 * 0.25;
        Si (Weight > 0.0) alors
            P 2P = P2P /sqrt (Distance2);
            NdotP2P = dot(Normal, P2P);
            Si (NdotP2P > 0.342) alors ;
            SSAO = SSAO + NdotP2P * Weight;
        Finsi;
    Finsi;
    i++;
FinTanque ;
FinPour ;
Fin ;
```

- **Explication d'algorithme de SSAO**

SSAO : facteur d'occlusion

CHAPITRE 3 : MISE EN ŒUVRE, RESULTATS ET PERSPECTIVE

1) Obtenir la position échantillon par rapport à la position sommet que veut calculer AO
 $\text{Vec2 TexCoord} = \text{clamp} (\text{ScaleRotationMatrix} * \text{Samples}[i] + \text{gl_TexCoord} [0].\text{st}, 0.0, 0.999999);$

Tel que

- $\text{clamp} ()$ fonction renvoie une valeur qui se situe forcément entre minimum et maximum. $\text{clamp} (\text{var}, \text{minimum}, \text{maximum}) ;$
- $\text{ScaleRotationMatrix}$ en le fait comme une matrice parce que la transformation besoin une multiplication des textures.
- $\text{gl_TexCoord} [0].\text{st}$ positions de chaque échantillon dans un bloc.

2) Obtenir la profondeur de l'échantillon par rapport à la position de cet échantillon

$\text{float depth} = \text{texture2D} (\text{DepthBuffer}, \text{TexCoord}).r;$

Tel que

- DepthBuffer : variable Uniform envoyer vers le shader SSAO, comme un frame buffer qui sauvgarde les profondeurs.

3) Obtenir la position de l'échantillon

$\text{Vec4 position} = \text{ProjectionBiasMatrixInverse} * \text{vec4} (\text{TexCoord}, \text{depth}, 1.0) ;$

Tel que

- $\text{ProjectionBiasMatrixInverse}$: matrice permet de transfere les position en 3D avant la projection de camera

4) normalization de vectrice position

$\text{position}.\text{xyz} /= \text{position}.\text{w} ;$

5) Distance entre SURFACE-POSITION et SAMPLE-POSITION

$\text{Vec3 P2P} = \text{position}.\text{xyz} - \text{Position}.\text{xyz};$

Tel que

- P2P : la position de point échantillonné.

6) le produit scalaire de deux mêmes vecteurs donne un angle

$\text{float Distance2} = \text{dot} (\text{P2P}, \text{P2P}) ;$

Tel que

- $\text{Distance2} = \text{Distance}$ entre les deux point position et Position

CHAPITRE 3 : MISE EN ŒUVRE, RESULTATS ET PERSPECTIVE

float Weight = 1.0 - Distance2 * 0.25 ;

Tel que

- Weight = poids pour atténuation les zone sombre. La loi dit : $w = 1 - r \cdot r$. $r = \|P2P\| / R$
- Tester le poids si Weight >0 alors
- normalize(P2P)
 $P2P = P2P / \text{sqrt}(\text{Distance2}) ;$

float NdotP2P = dot (Normal, P2P) ;

Tel que

- NdotP2P = L'angle entre la normal et le point échantillonné doit appartenir à l'intervalle [0,70]
 - Tester NdotP2P si NdotP2P >0.342 pour ignorer les échantillons très près de la base de la sphère (0.342 = cos(70)).
- Algorithme 4 : représente l'illumination de phong

```
Algorithme 4 :illumination
Début
  Si appliquer phong avec un éclairage direct alors (ambiante)
     $I = i_a \cdot k_a + i_d \cdot k_d + i_s \cdot k_s$ 
  Sinon ;
    Si appliquer phong avec un éclairage indirect alors (SSAO au lieu ambiant)
       $I = SSAO + i_d \cdot k_d + i_s \cdot k_s$ 
    Finsi ;
  Finsi ;
Fin ;
```


3.6 Résultats

Nous avons implémenté dans cette partie une technique d'occultation ambiante dans l'espace écran basant sur la technique de Bump Mapping, nous commençons par le calcul de G-buffer puis nous avons calculé SSAO avec l'utilisation d'un l'échantillon aléatoires, ce qui a produit du bruit dans les zone de SSAO. Pour résoudre ce problème nous allons utiliser le filtre bilatéral, ce qui nous permettra de supprimer ce bruit dans le résultat obtenu.

- **L'implémentation du G-Buffer et résultat de SSAO**

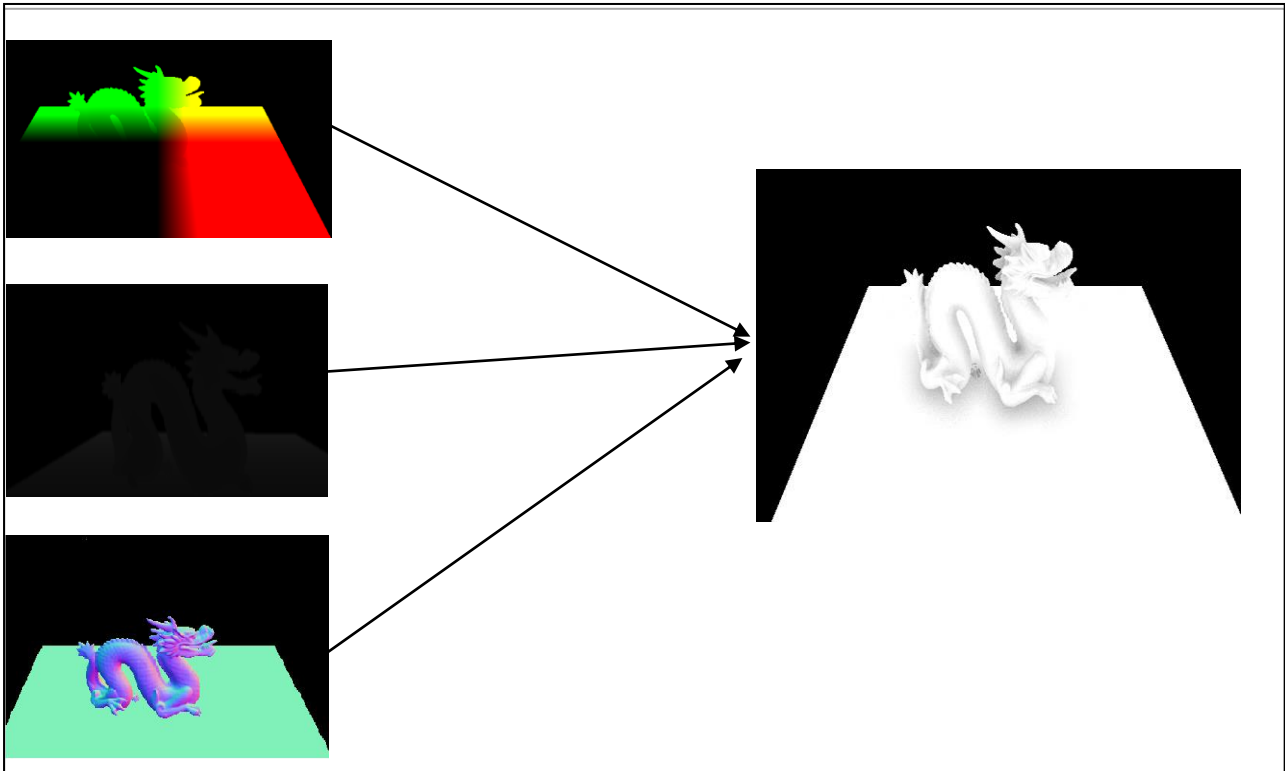


Figure 3. 7 : Résultats de G-Buffer et du SSAO.

- **Noyau d'échantillonnage**

On va essayer de comparer les différentes images obtenues par la variation de noyau d'échantillonnage (16 ,32 ,64).

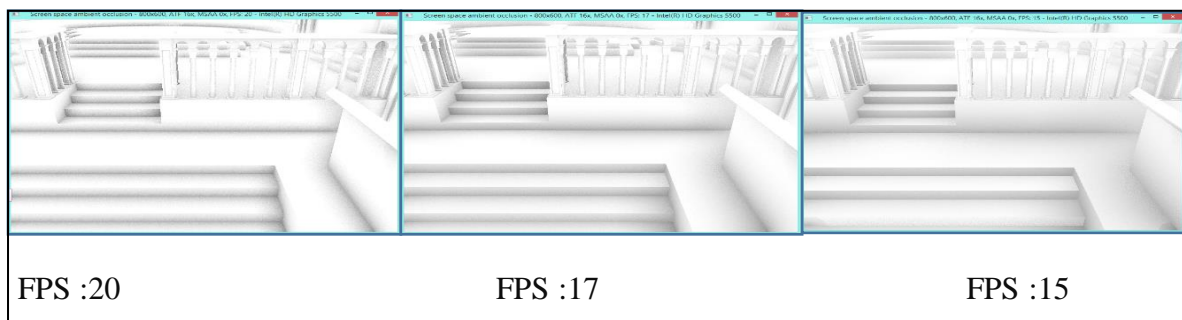


Figure 3. 8 : Résultats du noyau d'échantillonnage, appliqué au modèle 3D sibenik.

CHAPITRE 3 : MISE EN ŒUVRE, RESULTATS ET PERSPECTIVE

Les résultats de figure 3.7 montrent que l'augmentation du nombre de noyau d'échantillonnage réduit le bruit.

- **Bump mapping**

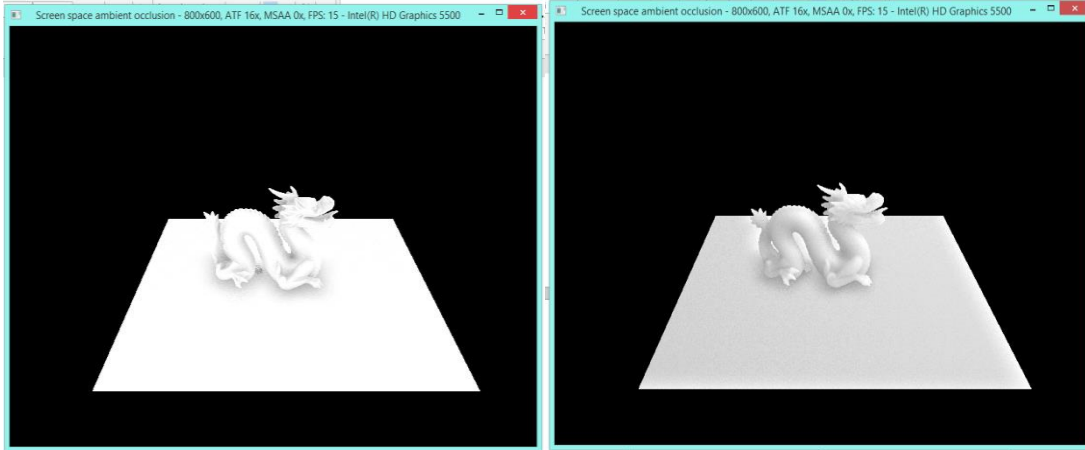


Figure 3. 9 : SSAO sans bump mapping, (à gauche). SSAO avec bump mapping (à droite).

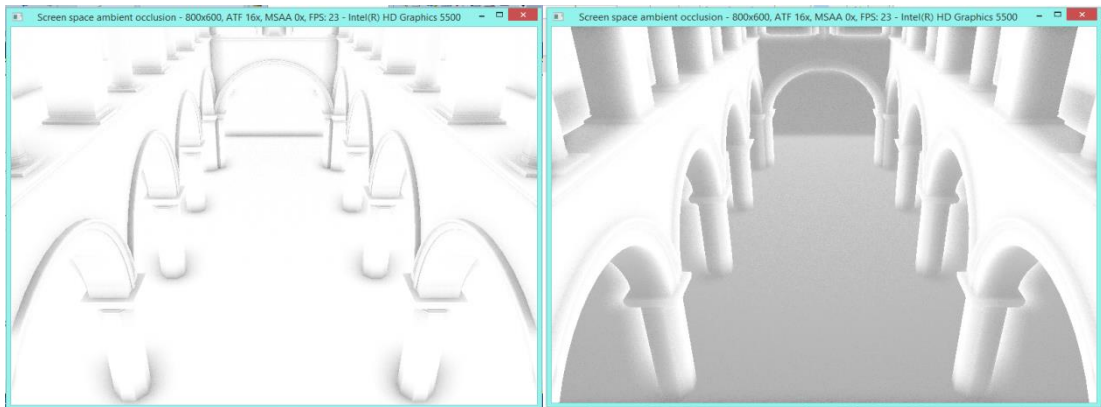


Figure 3. 10 : SSAO sans bump mapping, (à gauche). SSAO avec bump mapping (à droite).
Model de sponza0.

Les figures 3.8 et 3.9 représentent la technique de SSAO avec et sans la technique de bump mapping.

Nous avons remarqué que l'image de SSAO basé sur le bump mapping contient plus de détail contrairement à l'image sans le bump mapping.

C'est ce qui distingue le bump mapping est une petite variation dans le calcul de l'intensité, produisant l'illusion que la surface a une bosse. Cela se fait avec une image, généralement en niveaux de gris, qui utilise la couleur pour spécifier la variation.

CHAPITRE 3 : MISE EN ŒUVRE, RESULTATS ET PERSPECTIVE

● **Technique de Filtrage**

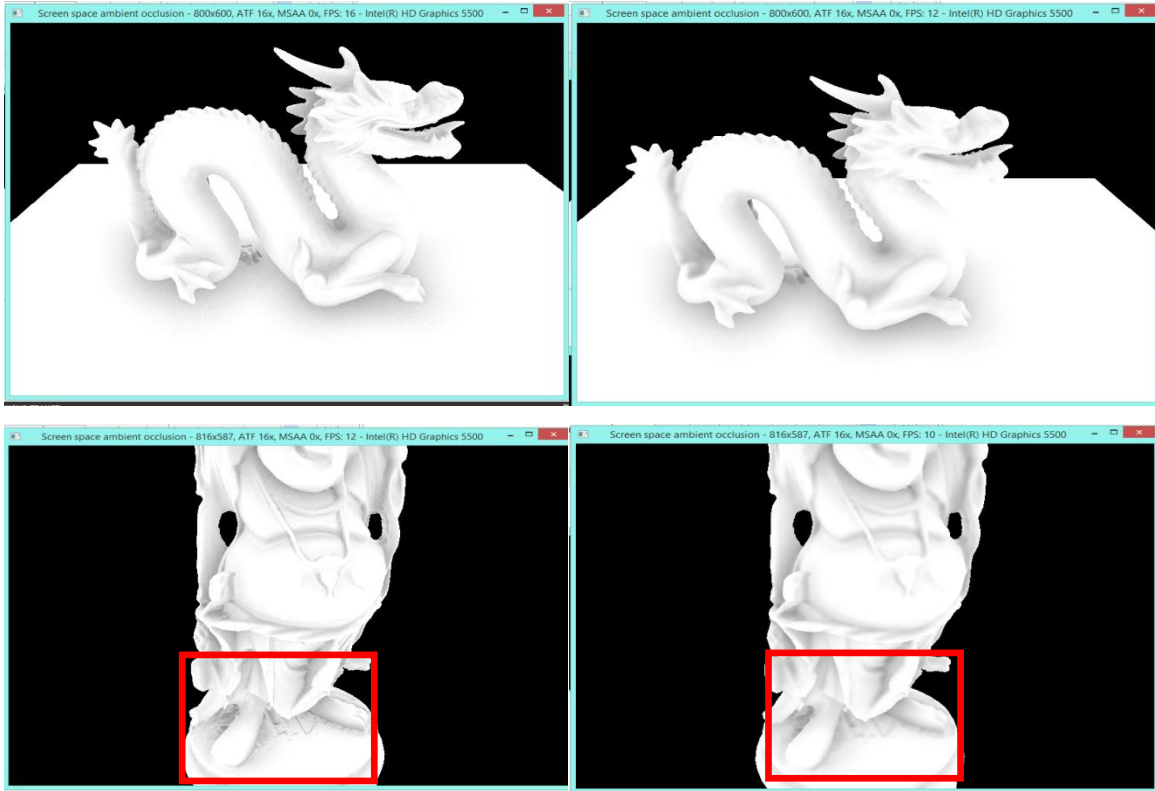


Figure 3. 11 : SSAO sans filtre bilatéral, (à gauche). SSAO avec filtre bilatéral (à droite).

Le figure 3.11 représente la comparaison entre l'image résultat de SSAO avec et sans le filtre bilatéral.

L'utilisation de ce filtre donne de meilleurs résultats et diminue considérablement le bruit.

Tel que le SSAO sans filtre génère 16 FPS, par contre SSAO avec filtre bilatéral diminue 12 FPS et diminue le bruit.

● **Discussion**

Nous avons comparé les résultats des noyaux d'échantillonnage dans la figure 3.7 et on a remarqué que chaque fois qu'on a un accroissement des nombres des noyaux d'échantillonnages on aura un décroissement des FPS et du bruit.

Pour comparer on a choisi le critère FPS et la quantité de bruit dans l'image.

CHAPITRE 3 : MISE EN ŒUVRE, RESULTATS ET PERSPECTIVE

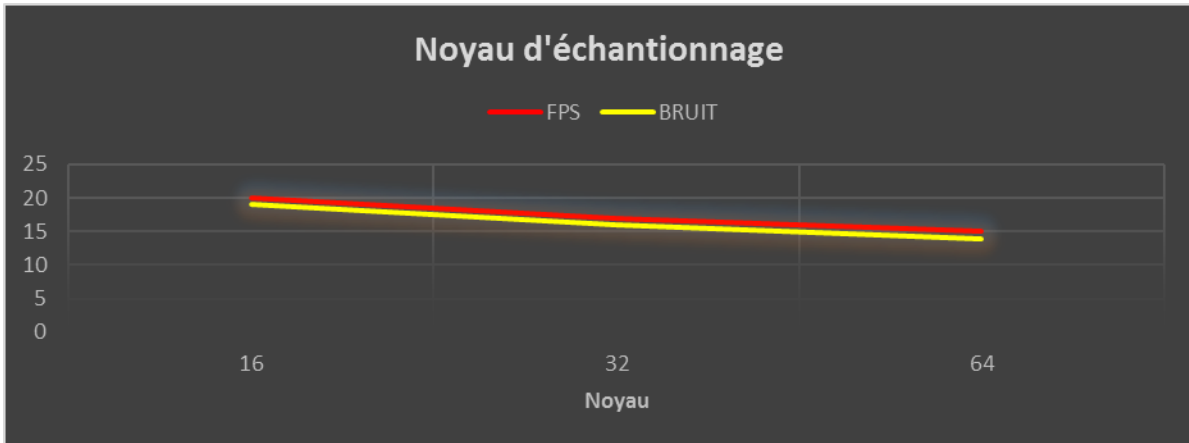


Figure 3. 12 : Comparaison entre les noyaux.

Dans les résultats figure 3.11 qui représentaient le SSAO avec et sans filtre bilatéral. On a remarqué quand l'utilisation du filtre conduit à la diminution du frame per seconds(FPS). Cependant la qualité des images s'améliore considérablement.

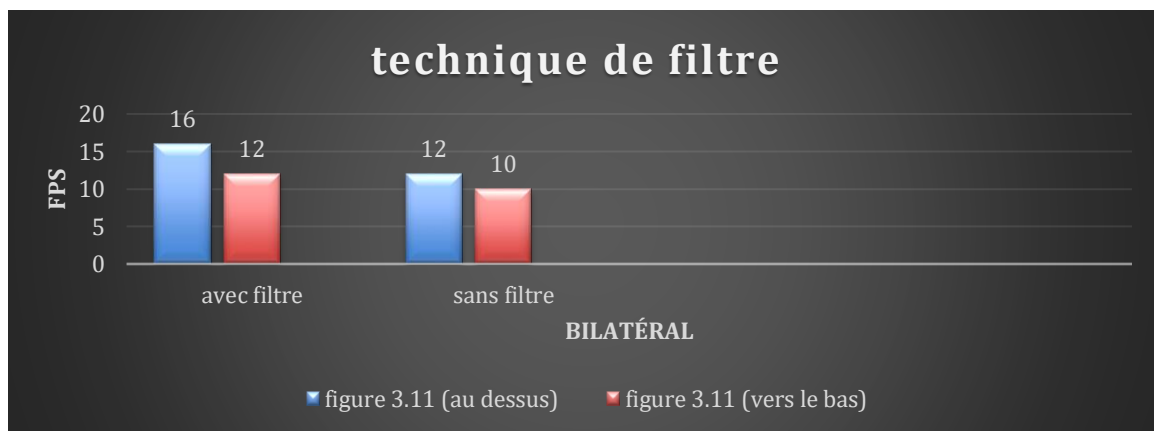


Figure 3. 13 : Filtre.

3.7 Conclusion

Dans ce chapitre, nous avons présenté l'implémentation de technique d'occultation ambiante dans espace écran où nous avons utilisé les shaders programmables intégrés dans les cartes graphiques et des G-buffer qui nous ont facilité la tâche.

Ensuite nous avons présenté nos résultats, les comparaisons ont été faites (résultats d'échantillonnage, résultats de SSAO avec et sans bump mapping et résultats de SSAO sans et avec des filtre bilatéral).

CHAPITRE 3 : MISE EN ŒUVRE, RESULTATS ET PERSPECTIVE

L'utilisation de la technique du bump mapping pour le calcul de SSAO à montrer ces efficacités dès les résultats obtenus dans les images générées qui contient plus de détail que celle avec la technique de SSAO standard.

Conclusion générale

Nous avons présenté dans ce projet les fondements de l'illumination globale et ses différentes méthodes d'approximation qui ont différents éléments essentiels. L'une de ces méthodes est l'occlusion ambiante qui est une technique de rendu qui fournit une approximation de la luminosité de la lumière sur différentes parties de la surface visible, ce qui n'est pas pratique dans une application en temps réel.

Pour résoudre ce problème, on propose l'occlusion ambiante dans espace écran(SSAO) comme une solution qui examine les pixels autour de l'élément et leur profondeur et est plus rapide à calculer. Mais l'implémentation de cette technique produit un certain bruit dans le résultat final pour cela on applique un filtre.

L'objectif est de modéliser une scène 3D, en calculant le G-buffer et les normales par la technique de bump mapping après pour le calcul du SSAO, puis insérer un modèle d'éclairage (phong). Le résultat peut être bruité à cause d'échantillonnage aléatoire, pour cela on applique un filtre pour améliorer la qualité visuel du résultat final.

Dans ce travail nous avons mise en place un rendu multi passes : le G-buffer et bump mapping où nous avons calculé le SSAO, en utilisant aussi les shaders programmables ce qui a permis d'accélérer le temps de calcul du rendu et d'avoir des résultats en temps réel.

Notre application peut avoir plusieurs améliorations dans le futur comme :

- Utiliser des textures d'Occlusion Ambiante pré calculé pour l'accélération des calculs.
- Remplacer l'échantillonnage régulier autour du pixel par un autre type d'échantillonnage, afin de nous donner des résultats plus précis, comme par exemple un échantillonnage par importance.

Bibliographie

- 1 D. METZ, qu'est-ce que la radiométrie, en ligne : URL : <http://www.blogcouleur.com/Qu-est-ce-que-la-radiométrie>, date de consultation : 13/03/2022.
- 2 Abd El Mouméne Zerari. "Volume D'ombre en Rendu Temps Réel" thèse de Magister Université Mohamed Khider de Biskra, 18 septembre 2011.
- 3 SIGGRAPH 2012 Course: Practical Physically Based Shading in Film and Game Production.
- 4 Christophe Cassagnabère. "Étude et Implémentation d'Algorithmes d'Illumination Globale Stochastiques Accélérés par le Matériel". Thèse Doctorat de l'Université du Littoral Côte d'Opale, 29 janvier 2007.
- 5 KAJIYA J. T. "The rendering equation" In Proc. SIGGRAPH (1986), vol. 20, pp. 143–150.
- 6 N. Takouachet, Utilisation de critères perceptifs pour la détermination d'une condition d'arrêt dans les méthodes d'illumination globale, thèse de doctorat 2009, Université Littoral Côte d'Opale.
- 7 David FRADIN. "Modélisation et Simulation d'Eclairage à Base Topologique : Application aux Environnements Architecturaux Complexes", L'Université de Poitiers, 25 avril 2002.
- 8 Pierre Y. Chatelier "Une approche de la radiosité par voxels, application à la synthèse d'images "Thèse Doctorat de l'Université d'Auvergne 4 décembre 2006.
- 9 Alexander Keller. "Instant Radiosity". In: Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '97. New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 1997,

pp. 49–56. ISBN: 0-89791-896-7. DOI: 10 . 1145 / 258734. 258769. URL:
<https://doi.org/10.1145/258734.258769>.

10 Dachsbacher c. Stamminger m. "Reflective Shadow Maps". In Proc. I3D (2005).

ISBN:1-59593-013-2 doi : 10.1145/1053427.1053460.

11 Tobias Ritschel, Thorsten Grosch, Min H. Kim, Hans-Peter Seidel, Carsten

Dachsbacher, and Jan Kautz. "Imperfect Shadow Maps for Efficient Computation of Indirect Illumination". ACM Transactions on Graphics, 2008.

ISBN: 978-1-4503-1831-0 doi : 10.1145/1457515.1409082

12 Abd El Mouméne Zerari. "Occultation ambiante basée sur l'échantillonnage

préférentiel multiple pour la génération des ombres douces en temps réel" thèse Université Mohamed Khider de Biskra, 04/ 11 /2018.

13 Joel Jansson. "Ambient Occlusion for Dynamic Objects and Procedural Environments", Master's Thesis Linköpings universitet, 10 June 2013

14 Marc Sunet. "Ambient Occlusion on Mobile: An Empirical Comparison", Master

in Innovation and Research in Informatics, Facultat d'Informàtica de Barcelona Department of Computer Science, January 25 2016.)

15 Janne Kontkanen and Samuli Laine. "Ambient Occlusion Fields". In SI3D '05:

Proceedings of the 2005 symposium on Interactive 3D graphics and games, New York, NY, USA, 2005. ACM Press.

16 Mattias Malmer, Fredrik Malmer, Ulf Assarson, and Nicolas Holzschuch. "Fast

Precomputed Ambient Occlusion for Proximity Shadows". Journal of Graphics Tools,2006.

17 Michael Bunnell. "GPU Gems 2". Addison-Wesley Longman, har/cdr (1. april2005) edition, 2005)

18 Mirko Sattler, Ralf Sarlette, Gabriel Zachmann, and Reinhard Klein. "HardwareAccelerated Ambient Occlusion Computation". In 9th Int'l Fall Workshop VISION, MODELING, AND VISUALIZATION (VMV), Stanford (California), USA, November16– 18 2004.

19 Tobias Ritschel, Thorsten Grosch, and Hans-Peter Seidel. "Approximating Dynamic Global Illumination in Image Space". In I3D '09: Proceedings of the 2009symposium on Interactive 3D graphics and games, New York, NY, USA, 2009.ACM. ISBN: 978-1-60558-429-4 doi : 10.1145/1507149.1507161.

20 Martin Mittring. "Finding Next Gen: CryEngine 2". In: ACM SIGGRAPH 2007 Courses. SIGGRAPH '07. San Diego, California: ACM, 2007. ISBN: 978-1-4503 1823-5. DOI: 10.1145/1281500.1281671.

21 MITTRING, M. 2007.Finding next gen: Cryengine 2 (course notes). In ACM SIGGRAPH 2007 courses, ACM, New York, NY, USA, SIGGRAPH '07, 97–121.

22 Frederik Peter Aalund DTU."A Comparative Study of Screen Space Ambient Occlusion Methods". Technical University of Denmark Informatics and MathematicalModelling

23 Dominic Fillion and Rob McNaughton. "Effects & Techniques". In: ACM SIGGRAPH 2008 Games. SIGGRAPH '08. Los Angeles, California: ACM,

2008.

DOI : 10.1145/1404435.1404441

24 Louis Bavoil, Miguel Sainz, and Rouslan Dimitrov. "Image Space Horizon

Based Ambient Occlusion". In: ACM SIGGRAPH 2008 Talks. SIGGRAPH '08. Los Angeles, California: ACM, 2008, 22:1–22:1. ISBN : 978-1-60558-343- 3. DOI : 10.1145/1401032.1401061.

25 (LOOS, B. J. AND SLOAN, P.-P. "Volumetric Obscurance" In Proceedings of the 2010 ACM SIGGRAPH symposium on Interactive 3D Graphics and Games ACM, New York, NY, USA, I3D '10, 151–156.)

26 (Morgan McGuire et al. "The Alchemy Screen Space Ambient Obscurance

Algorithm". In: High-Performance Graphics 2011. Vancouver, BC, Canada, 2011.

ISBN: 978-1-4503-0896-0 Doi: 10.1145/2018323.2018327.)

27 (BAVOIL, L., SAINZ, M., AND DIMITROV, R. 2008"Image-space horizon-based ambient occlusion".In ACM SIGGRAPH 2008 talks, ACM, New York, NY, USA, SIGGRAPH '08, 22 :1–22 :1.)

28 (MCGUIRE, M., OSMAN, B., BUKOWSKI, M., AND HENNESSY, P. 2011"The alchemy screen-space ambient obscurance algorithm" In Proceedings of the ACM SIGGRAPH Symposium on High Performance Graphics, ACM, New York, NY, USA, HPG '11, 25–32.)

29 (MITTRING, M., 2012. The technology behind the unreal engine 4 elemental demo.)

- 30** FILION, D., AND MCNAUGHTON, R. 2008. Effects and techniques. In *ACM SIGGRAPH 2008 Games*, ACM, New York, NY, USA, SIGGRAPH '08, 133–164.
- 31** Hayden Landis. Production-ready global illumination. In *RenderMan in Production. Course 16 : ACM SIGGRAPH 2002 Course Notes*. ACM, Boston (2002).
- 32** Rob Bredow. Renderman on film. *Course 16: RenderMan in Production. ACM SIGGRAPH Course Notes*, July 2002.
- 33** Sergej Zhukov, Andrej Inoes, and Grigorij Kronin. An ambient light illumination model. In: *Drettakis, G., Max, N. (eds.) Rendering Techniques '98 (Proceedings of Eurographics Rendering Workshop '98)*. Pages: 45–56, Springer, Berlin Heidelberg New York, 1998. https://doi.org/10.1007/978-3-7091-6453-2_5.
- 34** James T Kajiya. The rendering equation. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, volume 20, pages: 143-150, ACM August 1986. <https://doi.org/10.1145/15922.15902>.
- 35** University of Utah School of Computing, <http://www.cs.utah.edu/school/history/#phong-ref>
- 36** Synthèse d'images 2. Illumination Licence Pro 2005-2006 Sébastien THON IUT de l'Université de Provence, site d'Arles Département Informatique <https://slideplayer.fr/slide/3023037/>
- 37** Samuli Laine, Hannu Saransaari, Janne Kontkanen, Jaakko Lehtinen, and Timo Aila. «Incremental Instant Radiosity for Real Time Indirect Illumination". In *Proceedings of Eurographics Symposium on Rendering 2007*, pages 277–286. Eurographics Association, 2007.

38 Reflective Shadow Maps: Part 2 – The implementation APRIL 13, 2016
BY ERICPOLMAN

39 Joey de Vries SSAO - LearnOpenGL <https://learnopengl.com/Advanced-Lighting/SSAO>

40 Jimenez, j Gutierrez, d Yang, j Reshetov, a Demoreuille, p Berghoff, t Perthuis, c
yu, h Mcguire, m Lottes, t Malan, h Persson, e Andreev,d and Sousa,t.2011.
“Filtering

Approaches for Real Time Anti-Aliasing". In ACM SIGGRAPH Courses,ACM, New York.

41 Les différentes techniques d’anti-aliasing 19 novembre 2018

42 Les différentes techniques d'anti-aliasing MatBlog
<https://blog.materiel.net/les-differentes-techniques-danti-aliasing/> 19 novembre 2018

43 Guang Deng and LW Cahill. “An adaptive Gaussian filter for noise reduction and edge detection”. In: Nuclear Science Symposium and Medical Imaging Conference, 1993. 1993 IEEE Conference Record. IEEE. 1993, pp. 1615–1619.

44 Sylvain Paris et al. “Bilateral filtering: Theory and applications”. In: Foundations and Trends R in Computer Graphics and Vision 4.1 (2009), pp. 1–73.

45 "Bump and Environment Mapping" (PDF). ics.uci.edu.

46 James F. Blinn. 1978. Simulation of wrinkled surfaces. In *Proceedings of the 5th annual conference on Computer graphics and interactive*

techniques (SIGGRAPH '78). ACM, New York, NY, USA, 286-292.
DOI=<http://dx.doi.org/10.1145/800248.507101>

47 Lengyel, Eric (July 2019). Foundations of Game Engine Development, Volume 2: Rendering. Terathon Software LLC. ISBN 978-0-9858117-5-4.

48 Oliver Mattausch. “Visibility Algorithms for Real-Time Rendering in General 3D Environments”. PhD thesis. Vienna University of Technology, 2010.

49 Article Objet 3D (format de fichier) 1 juin 2020 à 19:43 Donatello29
[https://fr.wikipedia.org/wiki/Objet_3D_\(format_de_fichier\)](https://fr.wikipedia.org/wiki/Objet_3D_(format_de_fichier))

50 Joey de Vries Deferred Shading - LearnOpenGL
<https://learnopengl.com/Advanced-Lighting/Deferred-Shading>

51 Tutorial 20: Bump Mapping <http://www.rastertek.com/dx11tut20.html>
January 3, 2016.