



**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET
POPULAIRE**

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

Université Mohamed Khider – BISKRA

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

Département d'informatique

N° d'ordre : IVA18/M2/2022

Mémoire

Présenté pour obtenir le diplôme de master académique en

Informatique

Parcours : Image et Vie Artificielle (IVA)

Optimisation du rendu des ombres doucees en temps réel

Par :

HACHAICHI MARIA

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

Boucetta Mebarek

MAA

Présidenta

Zerari abd El Mouméne

MCB

Rapporteur

Bentrah Ahlem

MCB

Examineur

Année universitaire 2021-2022

RÉSUMÉ

Les ombres résultent d'une interaction entre la lumière et les objets, et nous permettent de capter la profondeur de la scène et surtout confèrent un poids aux objets 3D. Les ombres se classent en deux grandes catégories : les ombres douces et les ombres dures. Dans la réalité de tous les jours, les ombres dures n'existent pas, seules les ombres douces ont une existence réelle. Produire des ombres douces est un sujet traité depuis longtemps en synthèse d'image, elles se caractérisent essentiellement par leur réalisme mais sont relativement complexes à reproduire en temps réel. L'ajout de la contrainte temps réel est relativement récent. De nombreuses applications interactives récentes ont incorporée des ombres en temps réel. Généralement, ces applications utilisent des volumes d'ombres [19], des cartes d'ombres [16] ou des techniques apparentées. Ces méthodes utilisent des sources lumineuses ponctuelles qui ne projettent que des ombres dures. Étant donné que les lumières du monde réel n'occupent pas un point mais une zone finie, les images réalistes nécessitent des ombres douces. Ainsi, à mesure que les systèmes graphiques interactifs deviennent plus réalistes, des procédés pour rendre rapidement des ombres douces sont nécessaires. Il est en fait directement lié aux possibilités de programmation des cartes graphiques de dernières générations.

L'objectif de notre travail est d'optimiser et d'afficher des ombres douces de haute qualité en temps réel, en mettant en évidence les problèmes liés aux algorithmes d'ombres douces en temps réel.

ABSTRACT

Shadows result from an interaction between light and objects, and allow us to capture the depth of the scene and above all give weight to 3D objects. Shadows fall into two broad categories: soft shadows and hard shadows. In everyday reality, hard shadows do not exist, only soft shadows have real existence. Producing soft shadows has long been a subject in computer graphics, they are essentially characterized by their realism but are relatively complex to reproduce in real time. The addition of the real-time constraint is relatively recent. Many recent interactive applications have incorporated real-time shadows. Typically, these applications use shadow volumes[19], shadow maps[16] or related techniques. These methods use point light sources that cast only hard shadows. Since real-world lights do not occupy a point but a finite area, realistic images require soft shadows. Thus, as interactive graphics systems become more realistic, methods for quickly rendering soft shadows are needed. It is in fact directly linked to the programming possibilities of the latest generation graphics cards. The goal of our work is to optimize and display high-quality soft shadows in real time, highlighting issues with real-time soft shadow algorithms.

ملخص

تنتج الظلال عن تفاعل بين الضوء والأشياء ، وتسمح لنا بالتقاط عمق المشهد وقيل كل شيء إعطاء وزن للكائنات ثلاثية الأبعاد. تنقسم الظلال إلى فئتين عريضتين: الظلال الناعمة والظلال القاسية. في الواقع اليومي ، لا توجد ظلال صلبة ، فقط الظلال الناعمة لها وجود حقيقي. لطالما كان إنتاج الظلال الناعمة موضوعًا في رسومات الكمبيوتر ، فهي تتميز بشكل أساسي بواقعيته ولكنها معقدة نسبيًا لإعادة إنتاجها في الوقت الفعلي. تعد إضافة قيود الوقت الفعلي حديثة نسبيًا. أدرجت العديد من التطبيقات التفاعلية الحديثة الظلال في الوقت الفعلي. بشكل عام ، تستخدم هذه التطبيقات مجلدات الظل [19] أو خرائط الظل [16] أو التقنيات ذات الصلة. تستخدم هذه الطرق مصادر الضوء النقطية التي تلقي بظلال قاسية فقط. نظرًا لأن الأضواء الواقعية لا تشغل نقطة ولكنها تشغل مساحة محدودة ، فإن الصور الواقعية تتطلب ظلالًا ناعمة. وبالتالي ، عندما تصبح أنظمة الرسوم التفاعلية أكثر واقعية ، هناك حاجة إلى طرق لعرض الظلال الناعمة بسرعة. إنه في الواقع مرتبط بشكل مباشر بإمكانيات البرمجة لأحدث جيل من بطاقات الرسومات ، الهدف من عملنا هو تحسين وعرض الظلال الناعمة عالية الجودة في الوقت الفعلي ، وإبراز المشكلات المتعلقة بخوارزميات الظل الناعمة في الوقت الفعلي .

Table des figures

1.1	Comment former une ombre	15
1.2	Schéma vision binoculaire.....	15
1.3	Le tapis volant.....	16
1.4	Mobiliers urbains	16
1.5	Information sur le positionnement relatif des objets	16
1.6	Information sur la forme du receveur.....	16
1.7	La Grande Lessive	17
1.8	Ombre d'une fenêtre.....	17
1.9	Une source de lumière Lointaine.....	17
1.10	Une source de lumière proche.....	17
1.11	Les trois catégories d'ombre	19
1.12	L'occultation par un objet opaque	20
1.13	L'occultation par un objet transparent	20
1.14	Ombre dure vs. Ombre douce	21
1.15	Shadow mapping	24
1.16	Volume d'ombre généré par un triangle	25
2.1	Les trois composants d'une ombre douce.....	27
2.2	La relation entre la taille de la source et l'étendue de la pénombre.....	28
2.3	L'effet de l'espace de source lumineuse a l'ombre.....	30
2.4	L'ombre d'un objet avec deux sources lumineuses.....	30
2.5	Image de comparaison de zoom de deux régions d'ombre. Rangée du haut : obturateur lointain. Rangée du bas : près de l'obturateur. Colonne de gauche : pénombre fixe (4x4 PCF) Colonne de droite : pénombre variable.....	36
2.6	Application de Filtrage PCF.....	36
2.7	Une approche de subdivision adaptative de source de lumière.....	37
3.1	plan de travail.....	41
3.2	Fractionnement du tronc de la vue de 0 % à 100 % dans la direction z [38].....	42
3.3	Partitionnement arbitraire du tronc de vue[38].....	43
3.4	Galerie sud de la scène du cloître d'elne rendue à l'aide de Cascade Shadow Map.....	43
3.5	Subdiviser la pyramide de vue [38].....	44
3.6	Subdivision de la source lumineuse de la zone.....	45
3.7	une région de micropolygone à quatre côtés.(L'algorithmme PCF).....	49
3.8	Utilisez un multifiltre PCF.(A chaque champ un taux de filtre spécifique).....	53
3.9	Tableau des images d'une sphère avec différents noyaux de PCF	54
3.10	Tableau pour différents objets avec différentes tailles et effet de multi pcf.....	56

Table des matières

Table des figures	04
Table des matièresl.	06
List of Abbreviations.....	09
Introduction générale	10
1. Chapitre 1 : les Ombres	13
1. Introduction	14
2. Définition.....	14
3. L'importance de l'ombre.....	15
4. L'ombre du point vue géométrique.....	17
5. Classification des ombres.....	18
6. Ombres temps réel.....	20
6.1 Les ombres dures(hard shadow).....	21
6.1.2 Techniques de rendu temps réel d'ombre du.....	21
6.2 Les ombres douces(Shadow soft).....	22
6.3 Le Mappage de d'ombre (Shadow Mapping).....	22
6.4 Le volume d'ombre (Shadow volume)	24
7. Conclusion.....	25
Chapitre 2: Les ombres douces (soft shaows)	26
1. Introduction.....	27
2. les ombres douces (soft-shadow)	27
2.1 Pénombre	27
2.2 Antombre.....	28
3. La relation entre la source et l'ombre.....	28
4. Ombres dures vs ombres douces.....	29
5. Problèmes importants dans le calcul des ombres douces.....	30
5.1 Composition de plusieurs ombres.....	30
5.1.1 Ombres de plusieurs sources lumineuses.....	30
5.1.2 Ombres de plusieurs objets.....	30
5.2 Ombres physiquement exactes ou fausses.....	31
5.2.1 Ombres d'une source de lumière étendue.....	31
5.2.2 Rapprochement de la région de pénombre.....	32
5.2.3 Illumination dans la région d'ombre.....	32
5.2.4Ombres de différents objets.....	32
5.3 Ombres d'objets spéciaux.....	33

5.4 Contraintes sur la scène	33
6. calcule les ombres douces.....	33
7. Travaux connexes sur les ombres douces en temps réel.....	34
8. Filtrage PCF (Percentage Closure Filtering)	35
9. Ombres douces rapides et précises avec échantillonnage adaptatif de la source de lumière surfacique	37
10. Conclusion.....	38
Chapitre 3: Conception , Implémentation et résultats.....	39
1. Introduction	40
2. Description de la technique des soft shadow maps en temps réel basée sur plusieurs PCF.....	40
2.1 Subdivision tronconique.....	42
2.2 Calcul du niveau d'ombre.....	43
2.3 Approximation des ombres douces avec échantillonnage de la source lumineuse de la zone.....	45
2.3.1 Subdivision de la source lumineuse de la zone.....	45
2.4 Rendu de la carte d'ombre.....	45
2.5 Évaluation des informations sur la carte d'ombre.....	46
2.5.1 Attribuer des pondérations de contribution à Shadow Map.....	46
2.5.2 Visualisation des ombres douces à l'aide de n Shadow Maps.....	47
2.5.3 Évaluation des ombres douces à l'aide de tableaux de texture.....	47
2.5.4 Filtrage PCF « Percentage-Closer Filtering(pcf) ».....	48
3. Mise en œuvre.....	49
3.1 Résultats et évaluation.....	53
3.2 Résultats visuels.....	53
4. Conclusion.....	57
Conclusion.....	58
Bibliographie.....	61

List of Abbreviations

2D: Bidimensionnel .

3D: Tridimensionnel .

GPU: Processeur graphique .

PCF: Filtrage plus proche en pourcentage .

INTRODUCTION GÉNÉRAL

Les ombres jouent un rôle crucial dans la compréhension de la forme, de la taille et de l'emplacement relatif de la géométrie dans l'espace tridimensionnel. De nombreuses propriétés des lanceurs et des récepteurs d'ombre peuvent être dérivées en inspectant une silhouette d'ombre et son interaction avec les surfaces de la scène. Le rendu des ombres nécessite de résoudre un problème de visibilité point à surface. En cas d'ombre dure, cela revient à résoudre un simple problème de visibilité entre deux points pour chaque pixel. Cependant, le rendu d'une ombre douce nécessite de connaître le pourcentage d'occlusion dans chaque pixel. Les méthodes d'illumination globale ne conviennent pas au rendu en temps réel, en raison du coût de la visibilité qui est la principale dépense dans la plupart des méthodes. Cependant, les méthodes d'approximation d'illumination globale [32] peuvent générer des ombres en temps réel. Cependant, les applications en temps réel utilisent généralement des algorithmes de mappage d'ombre [16] et de volume d'ombre [19]. Le mappage des ombres est très flexible en termes de géométrie de scène, mais il présente des problèmes importants tels que le crénelage de perspective. Récemment, plusieurs approches ont été essayées pour réduire ces artefacts d'aliasing, Cascaded Shadow Maps (CSMs)[33] est l'une d'entre elles. La génération d'une ombre douce physiquement précise avec plusieurs cartes d'ombres nécessitait d'échantillonner la source lumineuse de la zone en n échantillons. Une texture d'ombre est créée pour chaque point lumineux à partir d'une position de vue dans une passe de rendu de scène. Par conséquent, aucune passe de rendu ne sera nécessaire pour générer l'ombre douce. Le temps de calcul augmente si le nombre de cartes d'ombres est grand, Cependant, pour les applications temps réel ce nombre doit être réduit pour réduire le temps de calcul, plusieurs méthodes sont créées pour accélérer la génération des ombres douces en réduisant le nombre d'échantillons. Une méthode qui y parvient a déjà été développée par Schwärzler et al. [30], mais uniquement en réduisant le nombre d'échantillons par la technique de subdivision adaptative de la source lumineuse. Cette méthode commence par échantillonner la source de lumière de surface sur un petit nombre d'échantillons et générer des points d'échantillonnage supplémentaires si nécessaire. Le problème avec cette méthode est que la taille des zones de pénombre peut changer radicalement en peu de temps, le nombre d'échantillons nécessaires peut également varier considérablement, rendant cette approche inadaptée aux applications en temps réel.

Pour les applications en temps réel, la puissance de calcul disponible est strictement limitée et peut nécessiter une large gamme d'approximations et d'hypothèses telles que

l'utilisation de modèles d'intégration de Monte-Carlo afin de rendre les ombres douces. Il est pour cette raison préférable de réduire le temps de calcul des ombres non seulement pour produire des ombres de meilleure qualité pour le même coût de calcul mais aussi pour libérer du temps de rendu pour d'autres effets qui n'ont pas été essayés auparavant en raison du volume de calcul intensif qu'ils impliquent afin d'améliorer la qualité globale de l'image.

Objectif et contribution

L'objectif de ce mémoire est de créer des ombres douces précises en temps réel en lissant les ombres des cartes d'ombres, le problème principal étant de réduire la netteté des bords des ombres afin de masquer le crénelage, Reeves et al ont proposé un algorithme de filtrage appelé Pourcentage Closer Filter (PCF) [26] qui effectue un test de profondeur Z sur plusieurs pixels adjacents de la shadow map. Acquisition de tons de gris à la limite de l'ombre, cette technique est une simulation d'ombre douce pour obtenir une haute qualité de rendu en temps réel , ce que nous allons présenter dans ce travail.

Structure de ce mémoire

Le mémoire est organisé en trois chapitres :

- Le premier chapitre définit la notion d'ombre et son importance, ainsi que nous expliquerons les deux types d'ombres : les ombres douces ; les ombres dures. Après cela, nous décrivons les techniques d'affichage de ces ombres, Shadow volume et Shadow Mapping
- Le deuxième chapitre, consacrant la description et l'étude sur les ombres douces, comparées aux ombres dures, mentionnant les problèmes qui entravent le calcul de ce type d'ombres, détaillant chaque problème, et présentant quelques travaux similaires dans ce contexte.
- Le chapitre trois ,présente une méthode pour créer des ombres douces et physiquement précises à l'aide de la technologie PCF dans des applications en temps réel. Nous commençons par une description générale de notre technique, suivie des différents détails de chaque partie de notre technique. Enfin, nous présentons nos résultats en étudiant les concepts obtenus.

Chapitre 1 :L'ombre

1. Introduction

Les ombres sont importantes pour la synthèse d'image car elles ajoutent réalisme et aider le spectateur à identifier les relations spatiales. De nombreuses techniques de génération d'ombres développées au cours des années ont été utilisées avec succès dans la production de films hors ligne. Cependant, il est toujours difficile de calculer des ombres de haute qualité en temps réel pour des scènes dynamiques.

2. Définition

« Ombre (nom, féminin, du latin umbra) : Zone sombre résultant de l'interception de la lumière ou de l'absence de lumière : "Au soleil couchant, la vallée est dans l'ombre". Silhouette sombre, plus ou moins déformée, que projette sur une surface un corps qui intercepte la lumière : "Les ombres des arbres s'allongent vers le soir." »

Une ombre est une zone sombre de l'espace résultant de l'obstruction (partielle ou complète) par un occulteur de la lumière émise par une source lumineuse. Les ombres sont des indices primordiaux pour le système visuel humain. Elles permettent d'appréhender le positionnement relatif des objets dans l'espace, donnent des informations sur leur forme, sur l'emplacement des sources de lumière, etc... Ainsi les ombres sont indispensables pour permettre la bonne interprétation d'une image par le cerveau. C'est pourquoi elles sont nécessaires au réalisme des images de synthèse, donc la présence des ombres dans une image fournit des informations essentielles pour la compréhension d'une scène et notamment sur la position relative des objets entre eux et des sources lumineuses [3].

On peut définir les ombres simplement : ce qui est visible depuis une source lumineuse est éclairé, ce qui ne l'est pas est ombré. Du point de vue géométrique, cela revient à déterminer s'il existe, entre une source lumineuse s et un point p d'une scène, un élément géométrique qui intersecte le segment (ps) (Figure 1.1) [2].

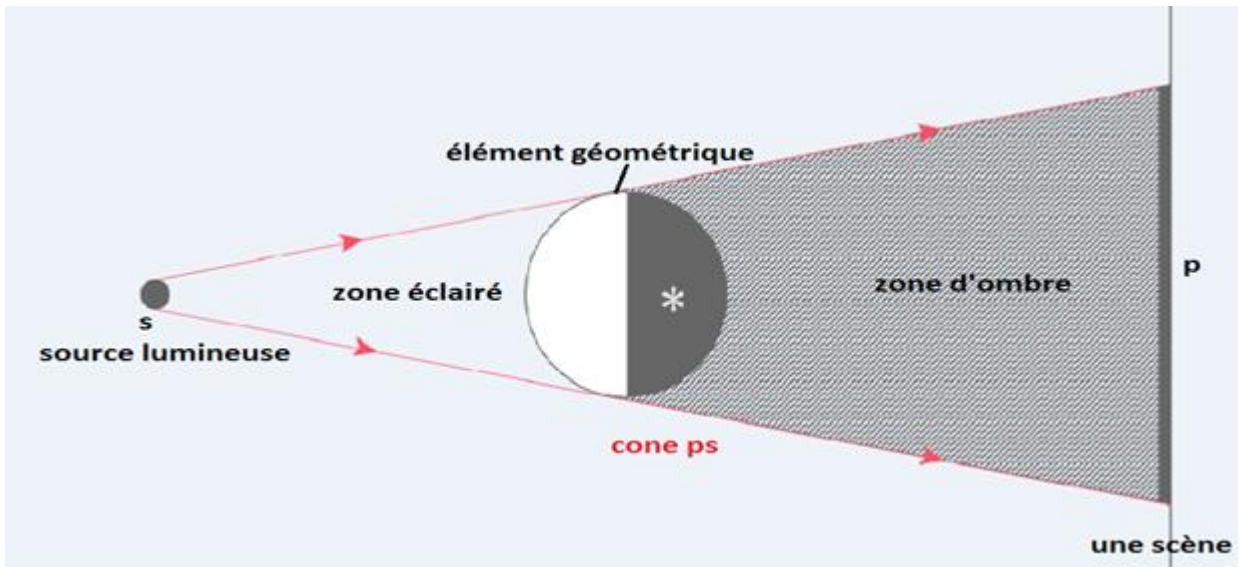


Figure 1.1 comment former une ombre[2]

3. Importance des ombres

Pour bien comprendre l'importance de l'ombre, particulièrement en synthèse d'images, il faut revenir à la base du fonctionnement de la perception visuelle humaine. Ainsi pour percevoir un objet.

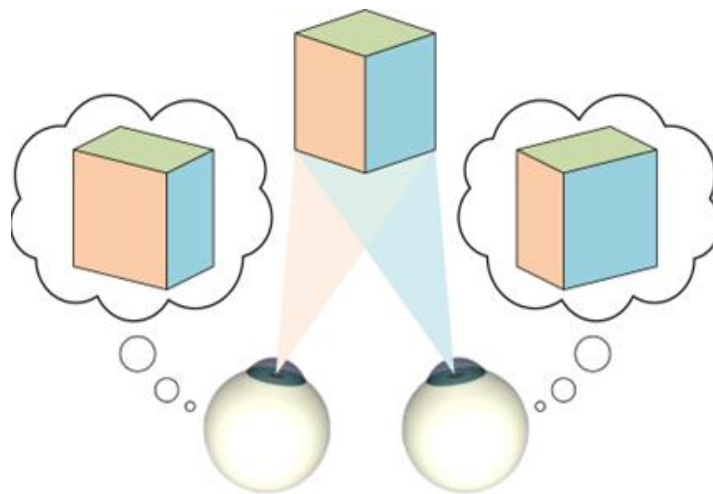


Figure 1.2 Schéma vision binoculaire

Le cerveau fusionne les 2 images bidimensionnelles issues des 2 yeux pour générer une perception unique avec du relief. Dans le cas où l'on regarde une image plane comme sur un écran, le cerveau ne peut pas extraire d'information sur le relief (cf. figure 1.2). Il doit alors déduire le relief et les distances entre les objets par d'autres moyens. Observé par un individu

privé de vision binoculaire, une image en deux dimensions donnera des perceptions fausses, comme sur les figures 1.3 et 1.4. Dans ces 2 exemples, c'est la mauvaise interprétation des ombres réelles ou fausses qui aboutit à la vision de situations non réalistes.

En pratique, l'ombre permet au cerveau de déterminer un certain nombre d'informations :

— la position relative des objets les uns par rapport aux autres, comme sur la figure 1.5 :

l'ombre donne une estimation de la distance entre le personnage et le sol .

— le relief d'un objet sur lequel une ombre se projette comme l'ondulation du sol (figure1.6).

— la perception d'un objet absent du champ visuel, mais dont l'ombre est visible comme sur les figures 1.7 et 1.8.



Figure 1.3 – Le tapis volant



Figure 1.4 -mobiliers urbains



Figure 1.5 – Information sur le positionnement forme du receveur[1].

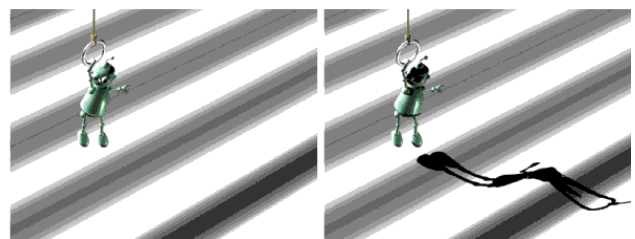


Figure 1.6– Information sur la Source

- L'ombre est donc un élément essentiel pour le réalisme d'une image de synthèse et son interprétation correcte par le cerveau [2].



Figure 1.8 Ombre d'une fenêtre



Figure 1. 7 la Grande Lessive

4. L'ombre du point vue géométrique

- La problématique du calcul d'ombre est une question de visibilité : un point non visible depuis la lumière sera dans l'ombre. Il faut donc être capable de savoir s'il existe un élément géométrique qui coupe le segment de droite reliant la source lumineuse et un point de la scène[2].

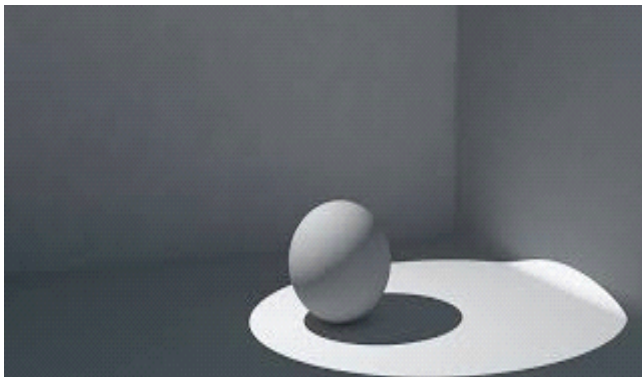


Figure 1.9 – Une source de lumière
Lointaine

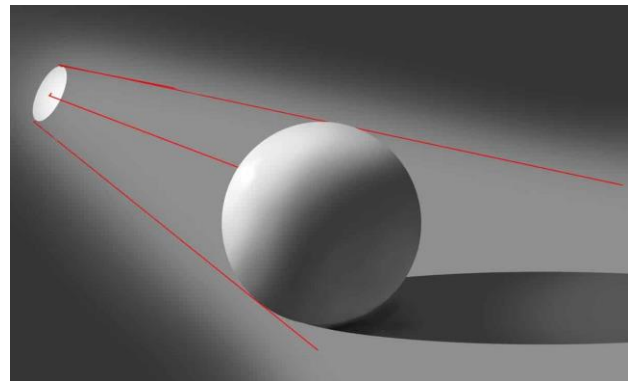


Figure 1.10 – Une source de lumière
proche

- au cas d'une source ponctuelle, On considérera par exemple que, proportionnellement à la distance Terre-soleil, une petite ampoule ou le soleil sont assimilables à des sources ponctuelles. Un élément géométrique va générer une zone d'ombre et le calcul de l'ombre se ramène à un choix binaire : totalement dans l'ombre ou totalement éclairée.

- l'intersection du rayon lumière-point avec toute la géométrie de la scène, et dès qu'on trouve un élément qui coupe ce rayon on sait que le point est dans l'ombre; sinon une fois que toute la géométrie est testée on peut conclure que le point est illuminé. L'application de ce genre d'algorithmes est inenvisageable pour du rendu temps réel : il devient vite très coûteux de tester toute la géométrie pour chaque pixel d'une image[2].

- En synthèse d'images en temps réel sur GPU (Graphics Processing Unit ou carte graphique), on représente la géométrie d'une scène avec t triangles. Pour une image de hauteur h et de largeur l et une scène de t triangles, le nombre de tests à faire pour une image est :

$$h \times l \times t$$

5. Classification des ombres

Il existe une multitude d'ombres différentes, tant par leurs formes que par leurs couleurs. Il est alors nécessaire de définir quelques caractéristiques communes de ces ombres afin de mieux les maîtriser. Les ombres dans l'infographie sont le résultat de l'interaction entre trois composants:

- sources lumineuses
- récepteurs d'ombre
- un occultant.

Les ombres se classent tout d'abord en trois catégories bien distinctes:

- **Les ombres propres** : sont les parties des objets non éclairées par une source lumineuse. Si un objet projette une ombre sur lui-même, il produit de l'ombre propre (self-shadowing).
- les ombres portées** : sur une surface sont le résultat de l'obstruction de la lumière par un objet.
- **La pénombre** : se situe à la périphérie des ombres portées, quand un point n'est éclairé que par une partie de la source lumineuse. Elle constitue la transition entre une zone éclairée et une ombre[1].

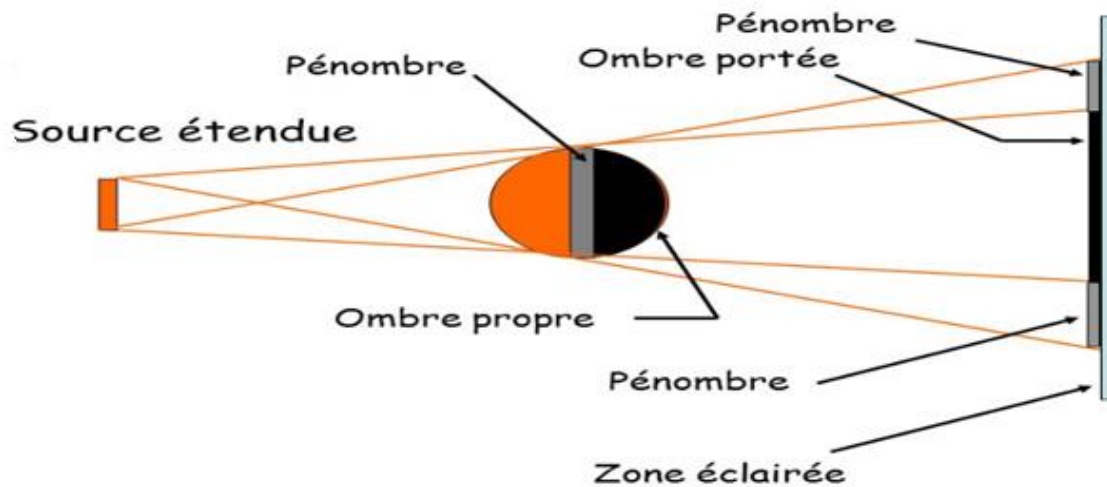


Figure 1.11 les trois catégories d'ombre

Les méthodes de rendu en synthèse d'images se divisent en deux catégories :

- celles qui résolvent (ou approximent) l'équation de l'illumination globale. Dans ce cas, les ombres font naturellement partie de la solution trouvée.
- celles qui, par souci de performance, effectuent une résolution locale et ne gèrent que les éclairages directs ou les trajets des rayons (potentiellement) les plus énergétiques. Dans ce cas, il devient nécessaire de traiter explicitement le calcul des ombres (Z-buffer, lancé de rayons élémentaires). [1]

On peut donc commencer à classer les différents types d'ombres :

- les ombres directes sont les zones sombres résultantes de l'occultation directe d'une source de lumière par un objet.
- les ombres indirectes sont les zones sombres résultantes de l'occultation de rayonnements indirects par un objet

Dans les deux cas, la disparition de l'objet occultant doit également faire disparaître l'ombre qui y est associée.

Le phénomène d'occultation peut prendre également deux formes :

- soit l'objet est opaque, les rayons touchant l'objet sont déviés par sa surface.
- soit l'objet est transparent, et dans ce cas, les rayons subissent soit une absorption partielle (diminuant ainsi leurs intensités), soit une focalisation (en concentrant les rayons dans une région au sortir de l'objet), soit les deux [1].

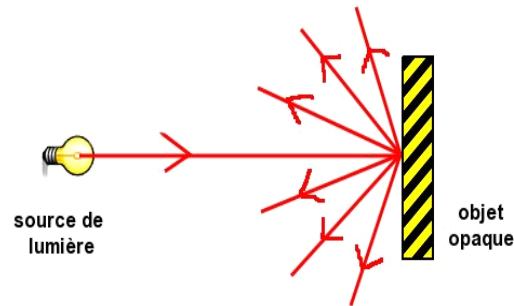


Figure 1.12 l'occultation par un objet opaque.

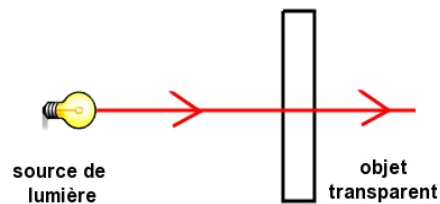


Figure 1.13 l'occultation par un objet transparent

6. Ombres temps réel

L'ombre en temps réel est un vrai défi en infographie, et il n'y a aucune solution générale qui peut être appliquée sur n'importe quelle scène. Les méthodes de l'illumination globale pour générer les ombres telles que: le tracé de chemins et la radiosité ne sont pas adaptés au temps réel. Par contre, les méthodes pour générer les ombres comme la shadow mapping (basée image) et le volume d'ombre (basé objet) sont bien adaptés au temps réel.

- La recherche des nouvelles méthodes pour le rendu des jeux vidéo, environnements virtuels et applications graphiques interactives plus réalistes, a vu un essor récent de travaux portant sur la génération d'ombres en temps réel [7; 8;9].

-Les techniques de calcul d'ombres peuvent être, dans la majorité des cas, regroupées en plusieurs catégories : les techniques traditionnelles [10; 11; 12], les techniques basées sur le lancer de rayons [15], les techniques de l'illumination globale [13; 14], les techniques basées sur les tampons de profondeur (depth buffers ou shadow maps) [16;17; 18] et finalement, celles basées sur les volumes d'ombre [19][1].

6.1 Les ombres dures

Les ombres dures engendrées par des lumières ponctuelles ou directionnelles. Le principe sur lequel est basée ce rendu d'ombres est simple, il s'agit d'un test binaire : soit la source est visible depuis la surface, soit elle ne l'est pas et dans ce cas on se trouve dans une zone d'ombre. Ce principe laisse apparaître des changements brusque dans la visibilité de la source lumineuse, ce qui a pour conséquence de provoquer des ombres aux contours nets. En réalité, il n'existe pas dans la nature de sources réellement ponctuelles de dimension 0, c'est pourquoi les ombres dures manquent de réalisme [20] [4].

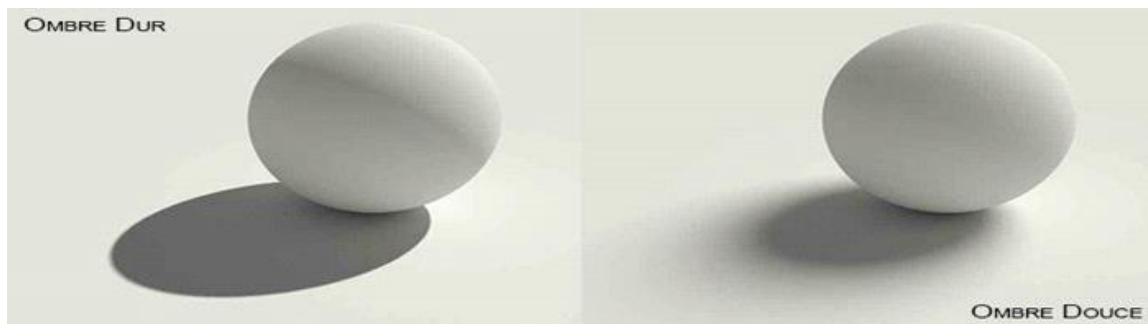


Figure 1.14 : Ombre dure vs. Ombre douce

6.1.2 Techniques de rendu temps réel d'ombre dure

Cette partie se focalisera sur les méthodes de rendu en temps réel des ombres dures. La problématique du rendu des ombres s'est posée dès l'apparition des premières méthodes de rendu. Quasiment tous les travaux dans le domaine se basent sur des techniques imaginées à la fin des années 70 :

- une méthode basée image utilisant le tampon de profondeur, connue sous sa dénomination anglo-saxonne de technique des Shadow Maps .
- une méthode basée sur la géométrie, la technique des volumes d'ombre ou Shadow Volumes.

— quelques approches alternatives pouvant mélanger plus ou moins les deux premières approches .Pour classer et comparer les diverses méthodes, il faut d’abord définir plusieurs critères.

— L’exactitude

— Le comportement face à la complexité géométrique

— Le comportement face à une hausse de la résolution image

— Le coût en mémoire

— Le type de source supportée[2]

6.2 Les ombres douces

les ombres douces (soft-shadow) engendrées quant à elles par les sources de lumière étendues (comme dans la réalité). Dans ce cas, on parlera de zone d’ombre, où la source de lumière est complètement occultée, et de zone de pénombre, où la source ici n’est vu que partiellement. Les ombres douces (taille de la zone de pénombre) dépendent de la taille de la source, de sa distance à l’objet, et de la distance de l’objet à la surface qui reçoit l’ombre[4].

6.3 Shadow Mapping

Le shadow mapping est de loin la technique la plus répandue pour le rendu d’ombre en temps-réel. Introduite par Williams en 1978 [16], la méthode a progressivement été intégrée aux cartes graphiques dans les années 90 jusqu’à bénéficier d’un support matériel complet, disponible sur les cartes graphiques grand public dès le début des années 2000. Dans sa définition initiale, le shadow mapping présente une complexité linéaire. La puissance et le parallélisme des GPU ont toutefois permis de porter la méthode à un très haut niveau de performances, avec des temps de calculs très stables et une très faible sensibilité vis à vis de la complexité géométrique. Ceci explique sa large adoption dans l’industrie et notamment dans les applications temps réel, où il est primordial de respecter les budgets de temps alloués pour chaque étapes du rendu d’une image. Par sa nature basée image, le shadow mapping souffre cependant de nombreux artefacts visuels qui rendent difficile, voire impossible, l’obtention de résultats exacts en toutes circonstances.[3]

Principe:

Le shadow mapping repose sur l'observation suivante : un point visible depuis la caméra est éclairé uniquement s'il est visible depuis la source de lumière. Il s'agit donc de calculer

la visibilité depuis la caméra d'une part, depuis la lumière d'autre part, puis de déterminer les éléments communs à ses deux vues. Le calcul des deux vues est effectué en deux passes d'algorithme du Z-buffer, produisant deux buffers de profondeur, celui correspondant à la vue depuis la lumière constituant la shadow map. Une shadow map est donc une image calculée depuis le point de vue de la lumière qui contient dans chacun de ses pixels la profondeur de l'objet visible le plus proche. Lors du rendu de l'ombre, chaque point visible depuis la caméra est projeté dans l'espace image de la shadow map, et un test de comparaison entre la profondeur stockée dans le pixel correspondant et celle du point permet de déterminer sa visibilité. Le point est dans l'ombre s'il possède une profondeur supérieure à celle stockée dans le pixel, sinon il est éclairé.[3]

Le calcul d'une shadow map repose donc sur l'algorithme de rasterisation qu'implémentent les cartes graphiques. Celui-ci consiste à projeter chaque primitive géométrique de la scène (de nos jours principalement des triangles) sur le plan écran de la caméra afin de déterminer les pixels qu'elle recouvre. Seule la primitive projetée la plus proche contribue à la couleur d'un pixel puisque les primitives plus éloignées seront masquées (en considérant uniquement des objets opaques). La rasterisation doit pour cela employer un mécanisme qui permet de déterminer pour chaque pixel la visibilité des primitives qui s'y projettent. Ce problème est résolu par l'emploi d'un tampon de profondeur (Z-buffer), qui stocke pour chaque pixel la profondeur de la primitive projetée la plus proche. Une shadow map est ainsi générée très rapidement par simple rasterisation de la scène depuis la lumière.[3]

Shadow Mapping

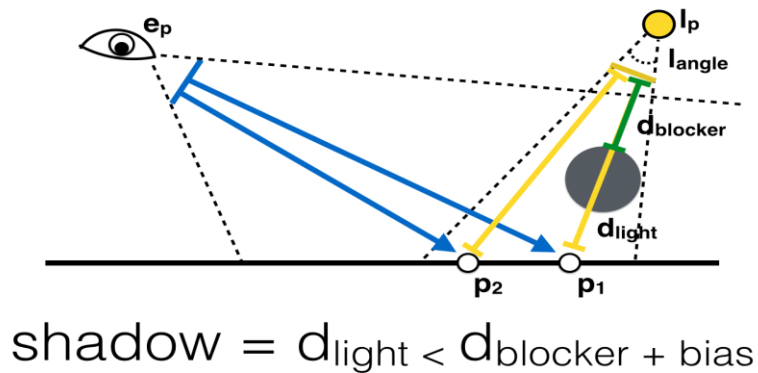


Figure 1.15 la méthode de Shadow mapping

6.4 Le volume d'ombre (Shadow volume)

La méthode des volumes d'ombre (ou Shadow Volumes) est une méthode analytique qui, contrairement au shadow mapping, utilise la géométrie de la scène pour déterminer si un point se trouve dans l'ombre. Les volumes d'ombre opèrent donc dans l'espace objet et sont à ce titre opposés aux shadow maps qui opèrent dans l'espace image. Introduite en 1977 par Crow, cette approche permet d'obtenir des résultats exacts en toute circonstance, au prix toutefois de temps de calculs plus importants et moins stables en raison de leur dépendance à la complexité géométrique et visuelle de la scène.[3]

Principe de fonctionnement

Géométriquement, l'ombre projetée par un objet donne lieu à un volume. Pour un triangle, ce volume est une pyramide tronquée formée par les trois plans qui passent par la lumière et chaque arête du triangle, avec comme base le triangle lui-même (figure dans Chapitre 2 N3). Un point de la scène est ombré s'il se trouve à l'intérieur d'au moins un volume d'ombre. À l'inverse, s'il n'est contenu dans aucun volume d'ombre, il est éclairé. Le but des méthodes dites de volumes d'ombre est donc de représenter la géométrie qui compose ces volumes afin de déterminer si les échantillons de vue sont contenus de dans ou non. Une approche naïve consisterait à tester pour chaque pixel l'ensemble des volumes d'ombre jusqu'à trouver une inclusion du point dans l'un d'eux. La complexité de ces requêtes en $O(n)$ pour n triangles n'est

évidemment pas viable en pratique. Bien que la notion de volumes d'ombre fut introduite en 1977, il faudra attendre 1991 pour voir émerger la première implémentation adaptée aux cartes graphiques, et 2003 pour qu'elle soit appliquée de manière à obtenir des performances temps-réelles sur des scènes suffisamment complexes.[3]

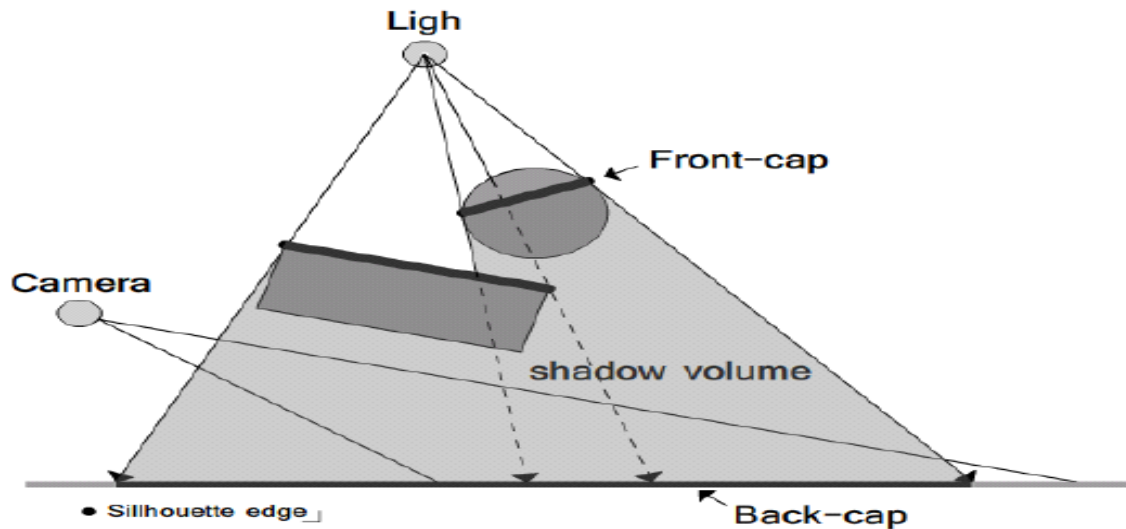


Figure 1.16 Volume d'ombre généré par un triangle

7. Conclusion

Les ombres sont nécessaires au réalisme des images de synthèse, si bien que les premiers algorithmes pour le rendu de scènes tridimensionnelles adressaient déjà cette problématique. Deux concepts introduits à la fin des années 70 ont nourri ce domaine de recherche durant les décennies suivantes. D'un côté, les shadow maps souffrent de nombreux artefacts visuels dus à la projection de la géométrie sur un écran, mais elles offrent d'excellentes performances grâce au support matériel des cartes graphiques. De l'autre côté, les volumes d'ombre produisent un résultat visuel exact par pixel en exploitant la géométrie de la scène dans son espace original (l'espace objet). Mais ils sont ainsi plus coûteux et moins stables que les shadow maps. Dans un contexte temps-réel, les shadow maps ont généralement été privilégiées pour leur vitesse et pour la garantie qu'elles offrent de respecter un budget de temps et de mémoire alloué.

Chapitre 2:

Les ombres douces (soft shaows)

1. Introduction

Le calcul d'ombre douce dans une application d'infographie est une tâche très coûteuse en temps de calcul, elle peut prendre plusieurs millisecondes jusqu'à plusieurs minutes ou même heures. Cela dépend de la méthode de rendu, l'algorithme d'ombre utilisé, le matériel disponible, la qualité de l'ombre, et la complexité de la scène. Le matériel d'aujourd'hui est capable de rendre diverses techniques de génération d'ombres en temps réel.

2. Les ombres douces (soft-shadow)

Les ombres douces créent une arête douce qui devient plus visible à mesure que la distance entre l'objet et l'ombre augmente, les Parties d'une ombre :

- Ombre
- la pénombre
- anti-ombrage

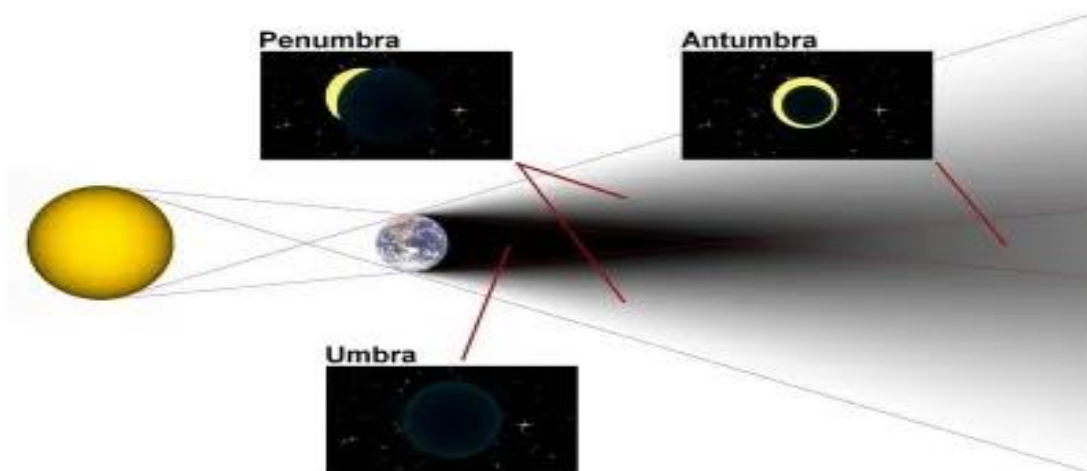


Figure 2.1 : les trois composants d'une ombre douce

2.1 Pénombre

Le mot pénombre vient du latin « paene » qui signifie « presque ou presque ». La pénombre est une partie de l'ombre d'un objet où seule une certaine partie du faisceau de lumière est obstruée par le corps occlusif. Un observateur situé dans la région de pénombre ne voit qu'une éclipse partielle. La pénombre est souvent qualifiée à juste titre de sous-ensemble de l'ombre car elle représente la partie de l'ombre formée par une partie de la source de lumière qui est obscurcie. Selon la Navigation and Ancillary Information Facility de la NASA, la partie

présente dans l'ombre se trouve également dans la pénombre.[21]

2.2 Antombre

Le mot « antumbra » vient du latin « ante » qui signifie « avant ». L'antumbra fait référence à la partie de l'ombre à partir de laquelle l'objet obstruant semble être placé complètement dans la région centrale de la source lumineuse. Si un observateur est placé dans la région de l'antumbra, alors il/elle peut voir une éclipse annulaire, c'est-à-dire un anneau lumineux de lumière entourant le corps qui s'éclipse. Avec la réduction de la distance entre l'observateur et la source lumineuse, la taille relative du corps obstruant augmente jusqu'au moment où il forme une ombre complète.[21]

3. La relation entre la source et l'ombre

Les ombres douces engendrées par les sources étendues. On alors trois cas :

- la source est totalement visible (zone éclairée)
- la source est partiellement visible (zone de pénombre)
- la source est complètement occultée (zone d'ombre portée). Voir aussi la (figure 2.10)

pour comprendre la dépendance entre la taille de la source et l'étendue de la pénombre (plus la taille de la source lumineuse est importante, plus la largeur de la zone de transition est grande). Dans le cas plus réaliste d'une lumière surfacique, un point sur un receveur pourrait ne voir qu'une fraction de la lumière. C'est ici que se fait la distinction entre la zone complètement éclairée, la pénombre et la zone d'ombre. Le calcul exact de ces zones est particulièrement difficile (et donc habituellement plus long), mais les ombres douces donnent des images beaucoup plus près de la réalité [1] (figure 2.2).

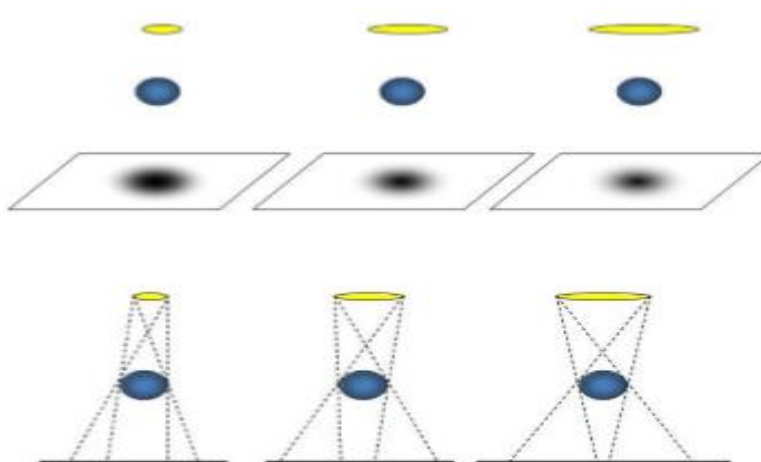


Figure 2.2 la relation entre la taille de la source et l'étendue de la pénombre

4. Ombres dures vs ombres douces

La notion d'ombre au sens commun est un statut binaire, c'est-à-dire qu'un point est "dans l'ombre" ou non. Cela correspond aux ombres dures, telles que produites par les sources lumineuses ponctuelles : en effet, une source lumineuse ponctuelle est soit visible, soit occluse de tout point de réception. Cependant, les sources lumineuses ponctuelles n'existent pas dans la pratique et les ombres dures donnent une sensation plutôt irréaliste aux images (voir Figure 1.14). Notez que même le soleil, probablement la source de lumière créant des ombres la plus courante dans notre vie quotidienne, a une étendue angulaire significative et ne crée pas d'ombres dures. Pourtant, les sources lumineuses ponctuelles sont faciles à modéliser en infographie et plusieurs algorithmes permettent de calculer les ombres dures en temps réel.[22]

Dans le cas plus réaliste d'une source lumineuse d'étendue finie, un point du récepteur peut avoir une vue partielle de la lumière, c'est-à-dire que seule une fraction de la source lumineuse est visible depuis ce point. On distingue la région d'ombre (si elle existe) dans laquelle la source lumineuse est totalement bloquée du récepteur, et la région de pénombre dans laquelle la source lumineuse est partiellement visible. La détermination de l'ombre et de la pénombre est une tâche difficile en général, car elle revient à résoudre des relations de visibilité en 3D, un problème notoirement difficile. Dans le cas d'objets polygonaux, la forme des régions d'ombre et de pénombre est intégrée dans un maillage de discontinuité¹³ qui peut être construit à partir des bords et des sommets de la source lumineuse et des obturateurs (voir Figure 1.14).[22]

En infographie, nous pouvons approximer une source lumineuse petite ou distante en tant que sources ponctuelles uniquement lorsque la distance entre la lumière et l'obturateur est beaucoup plus grande que la distance entre l'obturateur et le récepteur, et la résolution de l'image finale ne permet pas un rendu correct de la pénombre. Dans tous les autres cas, de grands avantages peuvent être attendus d'une représentation correcte des ombres douces.[22]

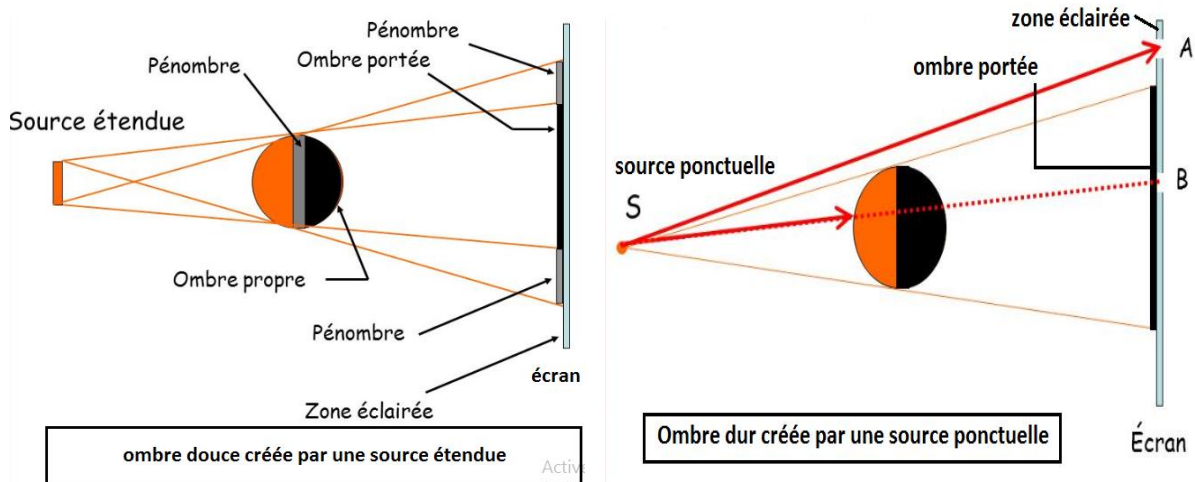


Figure 2.3 L'effet de l'espace de source lumineuse a l'ombre

5. Problèmes importants dans le calcul des ombres douces

5.1 Composition de plusieurs ombres

5.1.1 Ombres de plusieurs sources lumineuses

Ombres produites par plusieurs sources lumineuses sont relativement faciles à obtenir si nous savons comment traiter avec une source unique (voir Figure 2.3). Exigible à la nature linéaire du transfert de lumière, nous additionnons simplement la contribution de chaque lumière (pour chaque longueur d'onde ou bande de couleur)[22].

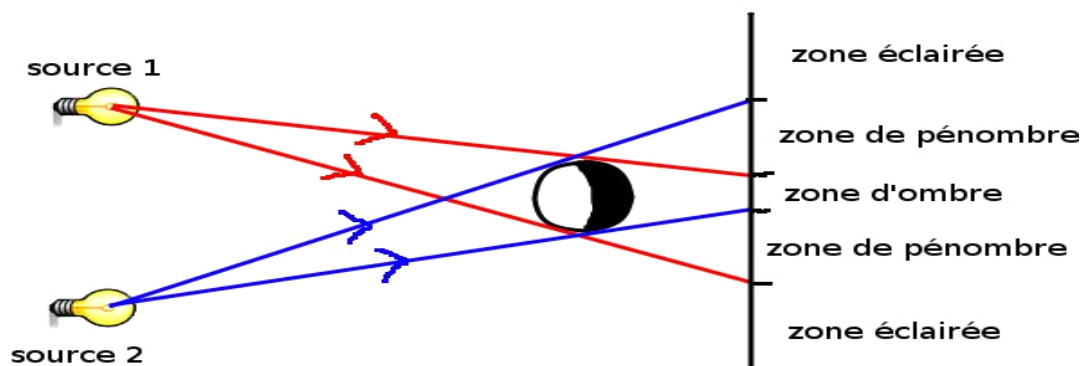


Figure 2.4 l'ombre d'un objet avec deux sources lumineuses

5.1.2 Ombres de plusieurs objets

Pour les sources lumineuses ponctuelles, les ombres dues à différents obturateurs peuvent être facilement combinées puisque la zone d'ombre (où la source lumineuse est invisible) est l'union de toutes les ombres individuelles.

Avec une source lumineuse surfacique, combiner les ombres de plusieurs occulteurs est plus compliqué. Rappelons que l'éclairage contribution de la source lumineuse sur le récepteur implique une fonction de visibilité partielle : un problème majeur est qu'il n'est pas simple la combinaison des fonctions de visibilité partielle d'obturateurs distincts peut donner la fonction de visibilité partielle de l'ensemble des bloqueurs considérés ensemble. Par exemple il peut y avoir points de la scène où la source de lumière n'est pas occluse par tout objet pris séparément, mais est totalement occlus par l'ensemble des objets pris ensemble. La corrélation entre les fonctions de visibilité partielle des différents obturateurs ne peuvent pas être prédite facilement, mais peut parfois être approximée ou délimité En conséquence, l'ombre de l'union des objets peut être plus grand que l'union des ombres des objets. Cet effet est bien réel, mais n'est pas très visible sur des scènes typiques, surtout si les objets projetant des ombres sont animés[22].

5.2 Ombres physiquement exactes ou fausses

5.2.1 Ombres d'une source de lumière étendue

Ombres douces proviennent de sources lumineuses étendues dans l'espace. Pour bien modéliser l'ombre projetée par de telles sources lumineuses, il faut tenir compte de toutes les parties de l'obturateur qui bloquent la lumière provenant de la source lumineuse. Cela nécessite d'identifier toutes les pièces de l'objet projetant une ombre visible d'au moins un point de la source lumineuse étendue, ce qui est algorithmiquement beaucoup plus compliqué que d'identifier des parties de la obturateur visible d'un seul point .Parce que cette information de visibilité est beaucoup plus difficile

pour calculer avec des sources lumineuses étendues qu'avec une lumière ponctuelle sources, la plupart des algorithmes d'ombre douce en temps réel calculent les informations de visibilité à partir d'un seul point (généralement le centre de la source lumineuse) puis simuler le comportement de la source lumineuse étendue à l'aide de cette information de visibilité (calculée pour un point).

Cette méthode produit des ombres qui ne sont pas physiquement exactes, bien sûr, mais peuvent

être suffisamment proches des ombres réelles pour applications les plus pratiques. La différence entre l'approximation et l'ombre réelle est plus difficile à remarquer si les objets et leur ombre sont animés - un phénomène courant dans les algorithmes en temps réel. Alors que les grandes sources de lumière ne sont pas fréquentes dans les algorithmes en temps réel, le même problème se produit également si l'objet projetant.[22]

5.2.2 Rapprochement de la région de pénombre

Quand en temps réel les algorithmes d'ombre douce se rapprochent des sources de lumière étendues en utilisant des points, ils calculent en fait une ombre dure, et l'étendent pour calculer une ombre douce. Il existe plusieurs algorithmes possibles :

- étendre la région d'ombre vers l'extérieur, en calculant un extérieur région de pénombre,
- rétrécir la zone d'ombre et la compléter avec une zone intérieure région de pénombre,
- calculer à la fois la pénombre intérieure et la pénombre extérieure

5.2.3 Illumination dans la région d'ombre

Une question importante est l'illumination dans les régions qui sont dans l'ombre complètement cachées de la source lumineuse. Il n'y a pas de lumière atteignant ces régions, elles devraient donc apparaître entièrement noires, en théorie. Cependant, dans la pratique, une certaine forme d'éclairage ambiant est utilisé pour éviter les régions complètement sombres et pour simuler fait que la lumière finit par atteindre ces régions après plusieurs reflets. Les méthodes d'ombre en temps réel sont généralement combinées avec des calculs d'éclairage, par exemple en utilisant le modèle d'éclairage Open GL simple. Selon que la méthode de l'ombre opère avant ou après la phase d'illumination, l'éclairage ambiant sera présent ou absent. Dans ce dernier cas, la zone d'ombre apparaît complètement sombre, un effet qui peut être perceptible. Une solution consiste à ajouter l'ombrage ambiant en tant que passe ultérieure ; cette passe supplémentaire ralentit l'algorithme, mais la réutilisation intelligente du tampon Z-buffer sur le matériel graphique récent rend le coût supplémentaire gérable[22].

5.2.4 Ombres de différents objets

En présence de sources lumineuses étendues, l'ombre de l'union de plusieurs objets est plus

grande que l'union des ombres individuelles. De plus, la limite de l'ombre causée par la combinaison de plusieurs objets polygonaux peut être une ligne courbe. Ces effets étant liés au fait que la source lumineuse est étendue, ils ne peuvent pas apparaître dans les algorithmes qui utilisent un seul point pour calculer les surfaces visibles depuis la source lumineuse. Tous les algorithmes d'ombre douce en temps réel souffrent donc de cette approximation. Cependant, alors que ces effets sont clairement identifiables sur les images fixes, ils ne sont pas aussi visibles dans les scènes animées. Il n'existe actuellement aucun moyen de modéliser ces effets avec des algorithmes d'ombre douce en temps réel[22].

5.3 Ombres d'objets spéciaux

La plupart des algorithmes d'ombrage utilisent une représentation explicite des formes de l'objet, soit pour calculer les silhouettes des obturateurs, soit pour créer des images et des cartes d'ombre. Objets très complexes et volumétriques tels que nuages, cheveux, l'herbe, etc. nécessitent généralement un traitement spécial[22].

5.4 Contraintes sur la scène

Les algorithmes de shadowing peuvent imposer des contraintes particulières sur la scène. Les exemples incluent le type de modèle d'objet (les techniques qui calculent une ombre en tant que carte de texture nécessitent généralement un objet paramétrique, sinon un polygone), ou la nécessité/possibilité d'identifier un sous-ensemble de la scène en tant qu'occluseurs ou récepteurs d'ombre. Cette dernière propriété est importante dans adapter les performances de l'algorithme pour maintenir le temps réel[22].

6. Calcule d'ombre douce

Le calcul des ombres douces précis exige de résoudre l'intégration de l'équation de Kajiya [23]. Cette intégrale est très complexe à résoudre analytiquement. C'est pourquoi, beaucoup d'efforts sont consacrés à la recherche de méthodes approximatives. Parmi ces méthodes, nous nous intéressons particulièrement aux méthodes stochastiques de l'illumination globale du type Monte-Carlo [23]. La méthode de Monte-Carlo est utilisée essentiellement pour estimer cette intégrale, par un procédé probabiliste. L'estimation de la fonction intégrale est obtenue par la moyenne pondérée des valeurs de la fonction en un ensemble de points échantillonnés d'une manière stochastique[1] :

$$VL(x) \approx \sum_{i=1}^N w_i V(x, l_i) \implies VL(x) = \int_L V(x, l) \pi G(x, l) dl$$

Avec l'augmentation de nombre d'échantillons N, cet estimateur de Monte-Carlo converge vers la solution exacte :

$$Shading = \int_L \rho(x) \pi G(x, l) dl$$

VL(x) est le facteur de visibilité au point x:

$$VL(x) = \int_L V(x, l) dl$$

7. Travaux connexes sur les ombres douces en temps réel

Comme le calcul d'ombres douces physiquement correctes est généralement considéré comme trop coûteux pour les applications interactives ou en temps réel, la plupart des ces approches estiment la visibilité de la zone complexe (par exemple sur une surface) en calculant une seule ombre dure à partir du centre de la zone de la source lumineuse, et simulant la pénombre en utilisant des heuristiques approximatives. Les simplifications utilisées dans ces approches dites à échantillon unique ne conduisent généralement pas à des ombres douces physiquement correctes.

Un grand nombre d'algorithmes d'ombres douces en temps réel sont publiés au cours des dernières années, la plupart basés sur des extensions de l'algorithme de shadow mapping ou sur l'algorithme du volume d'ombre.

* Dans le travail de Wyman et al. [24], non seulement une carte d'ombre, mais aussi une carte dite de carte pénombre (Penumbra Map) est générée en analysant les silhouettes d'objets à partir de la position de la source lumineuse, permettant d'estimer une région de pénombre dans la passe d'illumination. Randima Fernando [25] suggère d'utiliser une technique appelée Percentage Closer Soft Shadows (PCSS), où Percentage Closer Filtering (PCF) de [26] est appliqué et combiné avec une recherche du bloqueur d'ombre. Le PCF adoucit les limites des ombres dures en comparant non seulement la profondeur actuelle à une seule valeur dans la shadow map, mais en faisant cela avec les pixels voisins dans la shadow map

* Le PCF adoucit les limites des ombres dures en comparant non seulement la profondeur actuelle à une seule valeur dans la shadow map, mais en faisant cela avec les pixels voisins dans

la shadow map. Le pourcentage de tests d'ombre réussis spécifie l'intensité de l'ombre. Il aide à réduire les artefacts d'aliasing aux limites des ombres adoucies, mais la pénombre est loin d'être précis, car elle a toujours la même taille.

* Plusieurs travaux [27; 28; 29] sont publiés, qui proposent des variantes d'une technique appelée back projection. L'idée est d'utiliser une seule carte d'ombre non seulement pour la comparaison de profondeur, mais pour l'utiliser comme une représentation discrétisée de la scène. Afin de calculer le facteur de visibilité v pour un pixel p de l'espace de l'écran, les texels de la carte d'ombre sont rétro-projetés de p sur la source de lumière, où la quantité d'occultation est estimée. Ces approches peuvent produire des résultats plus précis que PCSS et leurs variations, mais sont sujettes à des artefacts (par exemple, lorsque les bloqueurs d'ombre se chevauchent, lorsque la source de lumière est trop proche ou lorsque la pénombre est extrêmement grande) et il se peut qu'un nombre important de texels de la carte d'ombre soit projetée, ce qui réduit les performances[1].

8. PCF (Pourcentage Closure Filtering)

Le principe est simple et correspond à un moyennage sur les voisins des pixels courant. C'est ce principe que nous avons utilisé pour atténuer les ombres des shadow maps. Le problème majeur de l'algorithme est la possibilité qu'un aliassage très visible apparaisse au niveau des contours des ombres. Plusieurs techniques sont donc proposées pour atténuer ce problème. Comme cet aliassage ne sera présent qu'aux frontières de l'ombre, un réflexe intuitif serait de filtrer le résultat de la carte d'ombre, de façon à masquer l'aliassage. Reeves et al. proposent un algorithme de filtrage appelé le Percentage Closer Filtering (PCF) [26], qui effectue le test de profondeur Z-buffer sur plusieurs pixels voisins de la shadow map, de façon à obtenir des tons de gris aux frontières de l'ombre. Mais cette technique n'est qu'une simulation d'ombre douce.[5][1]

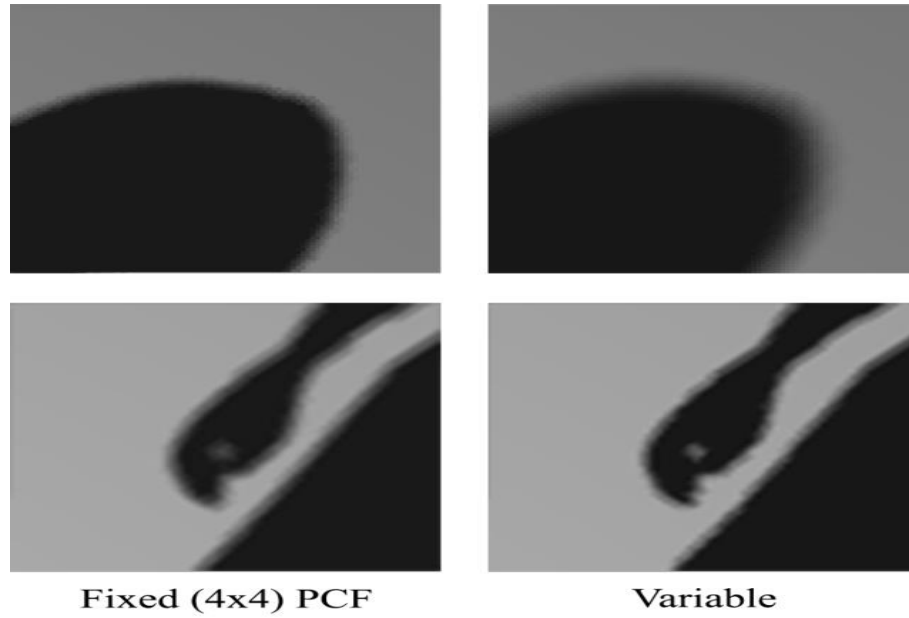


Figure 2.5 Image de comparaison de zoom de deux régions d'ombre. Rangée du haut : obturateur lointain. Rangée du bas : près de l'obturateur. Colonne de gauche : pénombre fixe (4x4 PCF). Colonne de droite : pénombre variable

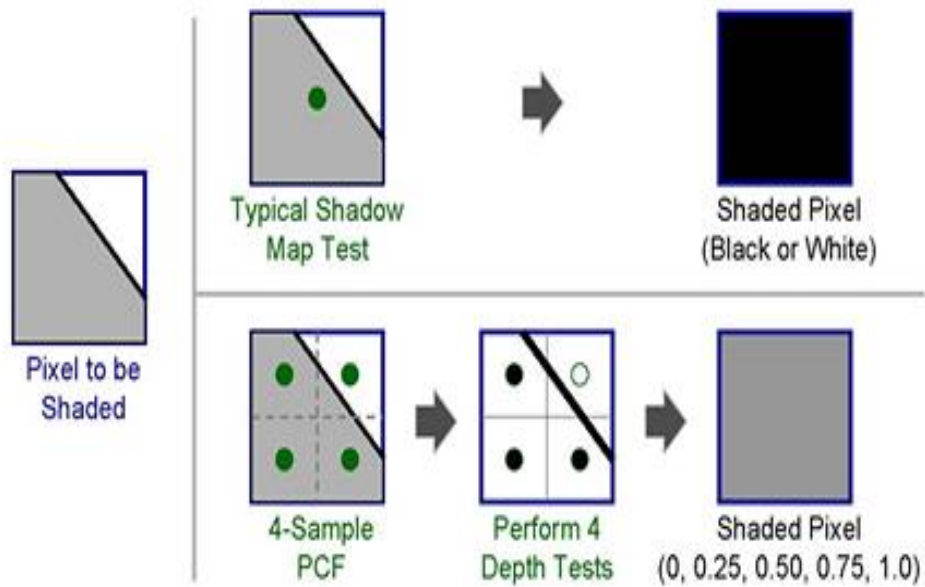


Figure 2.6 Application de Filtrage PCF

9. Ombres douces rapides et précises avec échantillonnage adaptatif de la source de lumière surfacique

Michael Schwärzler et al [30] dans leur article intitulé Fast Accurate Soft Shadows with Adaptive Light Source Sampling proposent une nouvelle approche de création d'ombres douces physiquement correctes en temps réel. Dans cet article les auteurs implémentent les ombres douces générées par une source de lumière surfacique. Ils proposent une approche de subdivision adaptative de source de lumière pour choisir les points d'échantillonnage de manière adaptative. L'idée principale est de commencer par avec quelques échantillons sur la zone de la lumière, évaluant à des différences utilisation les Hardware Occlusion Queries [31], et en ajoutant plus de points d'échantillonnage si nécessaire. Ils diminuent le nombre d'échantillons de la source de lumière. En s'appuyant sur la nouvelle génération des GPU, qui permettent, pour chaque pixel affiché à l'écran, de réaliser des opérations programmables, accroissant ainsi le réalisme en minimisant le coût. [1]

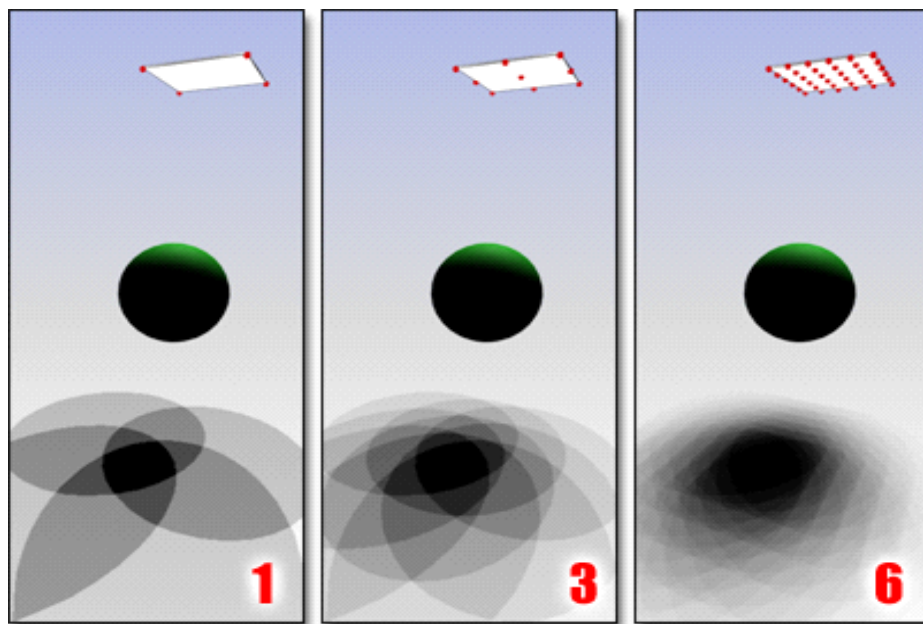


Figure 2.7 une approche de subdivision adaptative de source de lumière

10. Conclusion

Dans ce chapitre, nous avons présenté les notions utilisées pour générer les ombres en temps réel afin d'ajouter du réalisme aux scènes 3D. Nous nous sommes concentré principalement sur les ombres douces physiquement correctes en temps réel. En réalité, malgré le fait que les méthodes basées sur les ombres douces donnent une bonne approximation. Il y a des limites par les artefacts dus à l'échantillonnage de la source de lumière surfacique, particulièrement quand nous générons un petit nombre d'échantillons. Ainsi, pour couvrir ce problème, nous proposons dans le chapitre 3 une nouvelle technique permettant de diminuer le nombre d'échantillons de la lumière surfacique sans toucher aux performances.

Chapitre 3:

Conception , Implémentation et résultats

1. Introduction

Le mappage des ombres est la méthode de choix pour créer des ombres dans un rendu haut de gamme pour les films et la télévision. Cependant, il a été problématique d'utiliser le shadow mapping dans les applications en temps réel, telles que les jeux vidéo, en raison de problèmes d'aliasing. Ce chapitre montre comment réduire significativement l'aliasing shadow map dans un shader. Il décrit comment implémenter une version simplifiée du filtrage plus proche en pourcentage qui tire le meilleur parti du matériel de mappage des ombres du GPU pour restituer des ombres à bords doux et anticrénelées à des taux en temps réel.[34]

Le mappage d'ombre consiste à projeter une carte d'ombre sur la géométrie et à comparer les valeurs de la carte d'ombre avec la profondeur de la vue de lumière à chaque pixel. Si la projection provoque l'agrandissement de la carte d'ombre, un crénelage sous la forme de gros crénelages disgracieux apparaîtra aux bordures d'ombre. Le crénelage peut généralement être réduit en utilisant des cartes d'ombre à plus haute résolution et en augmentant la résolution de la carte d'ombre, en utilisant des techniques telles que les cartes d'ombre en perspective (Stamminger et Drettakis 2002). Cependant, l'utilisation de techniques de mappage d'ombre en perspective et l'augmentation de la résolution de la carte d'ombre ne fonctionnent pas lorsque la lumière se déplace presque parallèlement à la surface ombragée, car le grossissement approche l'infini.[34]

Un logiciel de rendu haut de gamme résout le problème d'aliasing en utilisant une technique appelée filtrage de pourcentage plus proche.

2. Description de la technique des soft shadow maps en temps réel basée sur plusieurs PCF

La technique proposée est inspirée des travaux de Schwarzler et al. [36] Il consiste à minimiser le nombre d'échantillons de source lumineuse pour des ombres douces précises en temps réel. Notre idée est de partitionner la source lumineuse de surface en un plus petit nombre d'échantillons que celle de Schwärzler et al. Méthode [35] et en utilisant la technique de Zerari [36]. Nous utilisons plusieurs filtres PCF pour supprimer les artefacts et lisser les ombres douces résultantes, ce qui aide à fournir une résolution plus élevée pour la texture de profondeur avec le grand nombre d'échantillons de points lumineux près du spectateur et une

résolution plus faible avec moins d'échantillons loin du spectateur. En divisant la tige de vue de la caméra en sections distinctes et en créant des cartes de profondeur d'échantillons de lumière ponctuelle pour chaque section afin d'augmenter la qualité et de réduire le temps de calcul.

Nous appliquons la technique proposée comme suit : Tout d'abord, nous divisons la source lumineuse de surface en n sources lumineuses ponctuelles pour chaque corde. Ensuite, nous visualisons la scène du point de vue de chaque source de lumière ponctuelle pour créer des cartes d'ombre. Ensuite, nous divisons la largeur du tronc en plusieurs troncs (sections). . Ensuite, nous utilisons plusieurs filtres PCF pour lisser les ombres résultantes et comparer le temps de calcul résultant.

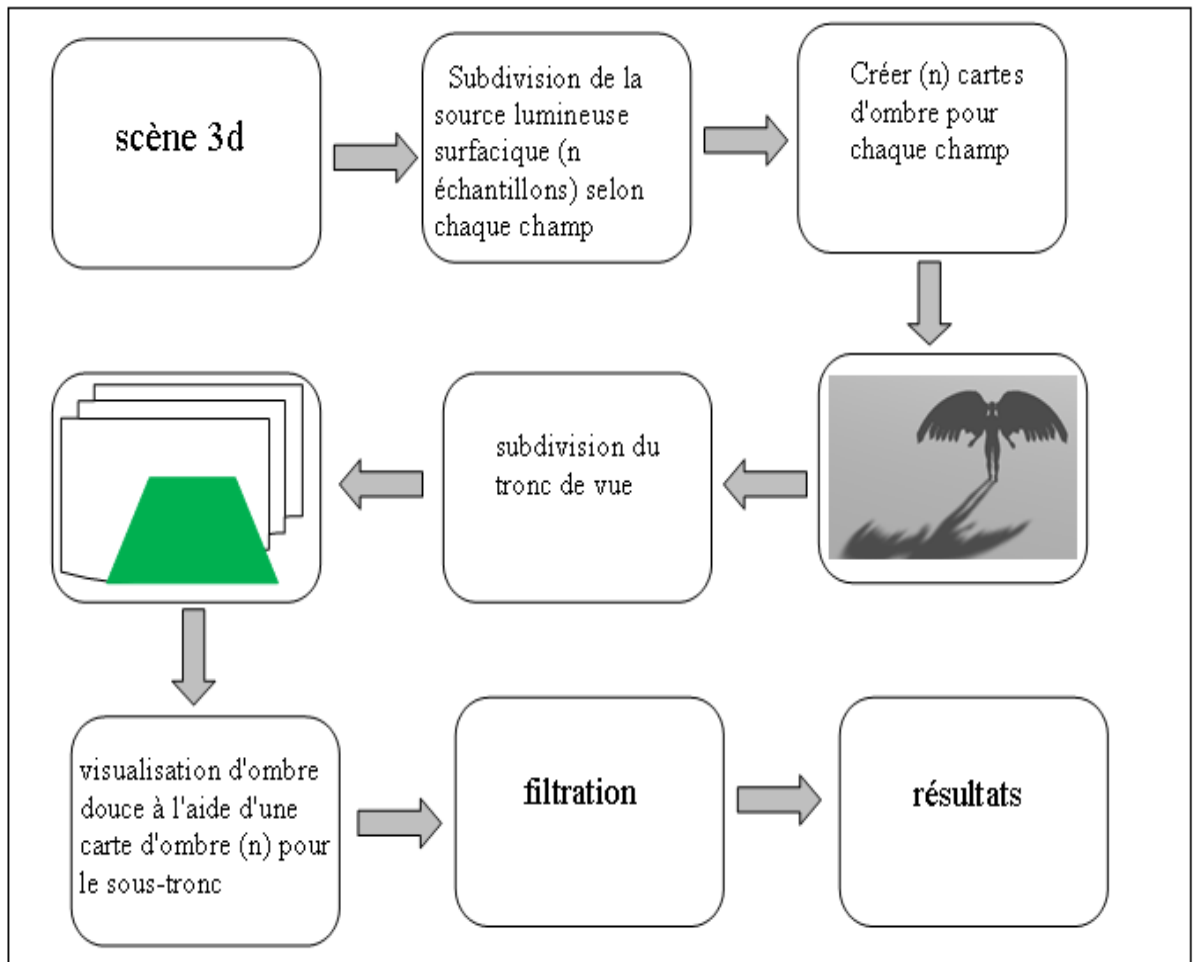


Figure 3.1 plan de travail

2.1 Subdivision tronconique

Partitionner le frustum est l'acte de créer des subfrusta. Une technique pour diviser le tronc consiste à calculer des intervalles de zéro pour cent (0 %) à cent pour cent (100 %) dans la direction Z (voir Figure 3.2). Chaque intervalle représente alors un plan proche et un plan éloigné en pourcentage de l'axe Z.

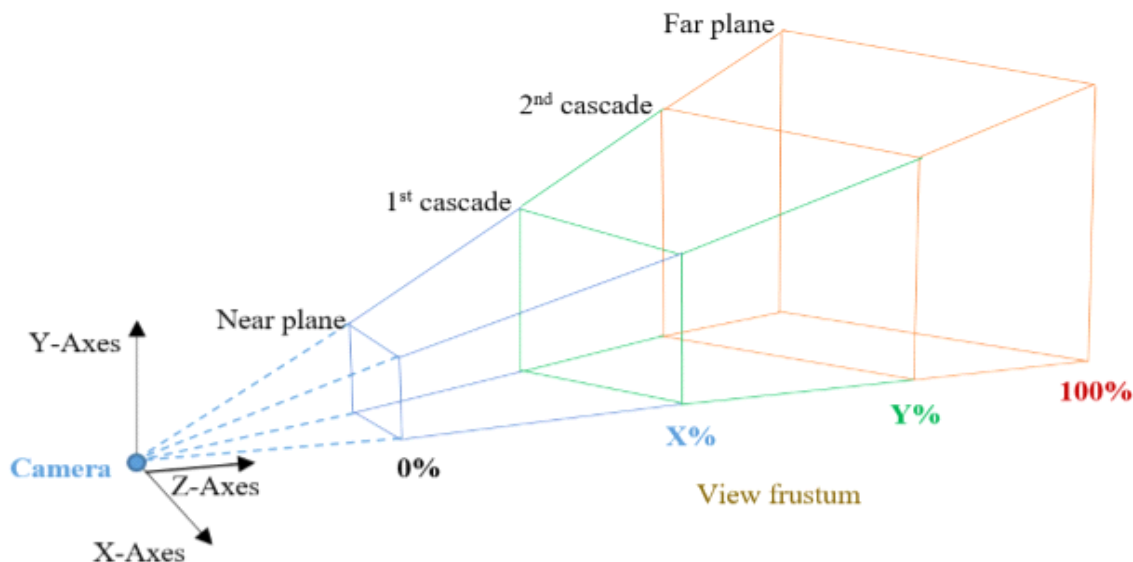
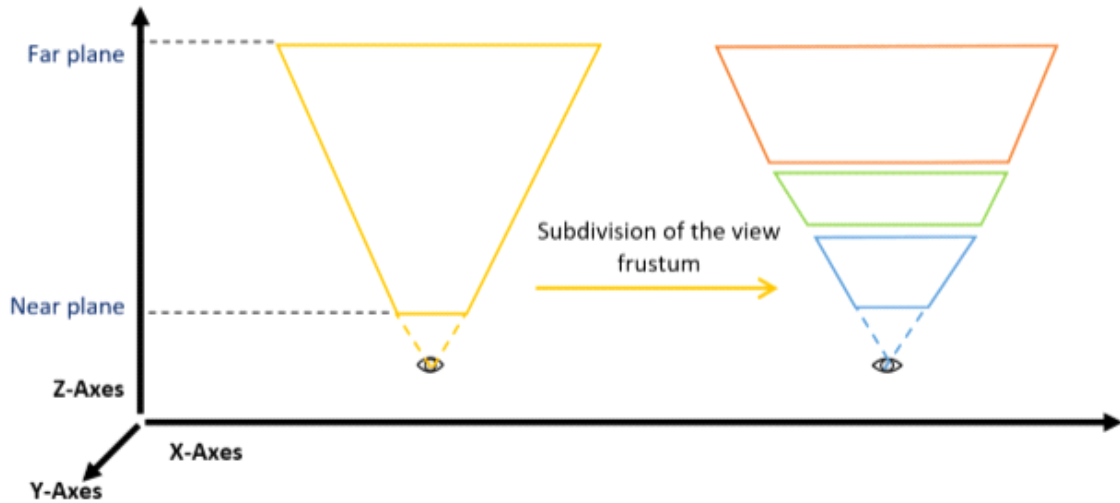


Figure 3.2 Fractionnement du tronc de la vue de 0 % à 100 % dans la direction z [38]

La subdivision du tronc de caméra se fait soit linéairement (chaque sous-tronc a la même longueur) soit logarithmiquement (la longueur du premier sous-tronc est inférieure à la longueur du dernier). Il peut également s'agir d'une combinaison de découpage linéaire et logarithmique ou il peut être partitionné arbitrairement comme nous l'avons fait dans ce travail (voir Figure 3.3).



Figur. 3.3. Partitionnement arbitraire du tronc de vue.[38]

2.2 Calcul du niveau d'ombre

A partir d'une vue de haut niveau, nous allons découper le tronc de la vue en plusieurs cascades (voir Figure 3.3). Chaque cascade sera rendue dans ses propres cartes d'ombre privées (une carte d'ombre pour chaque échantillon de lumière ponctuelle). L'algorithme d'ombre lui-même restera le même, mais lors de l'échantillonnage de la profondeur à partir des cartes d'ombre, nous devons sélectionner les cartes appropriées en fonction de la distance du spectateur

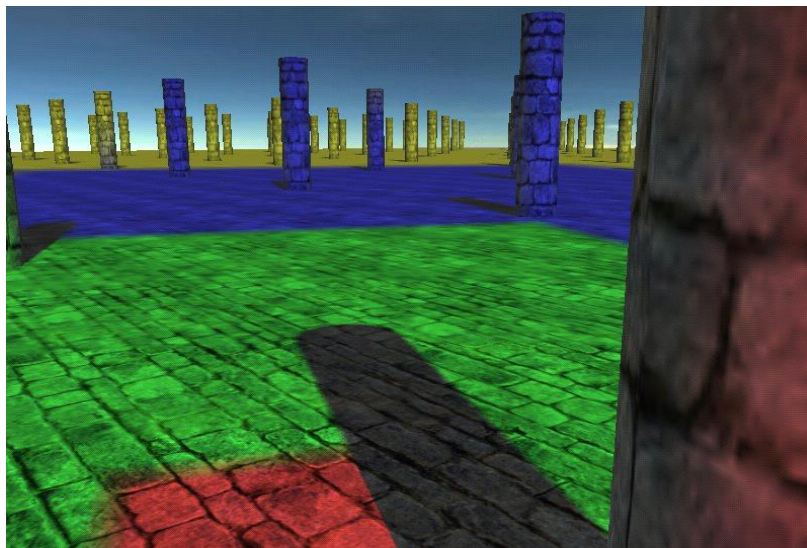


Figure 3.4 Galerie sud de la scène du cloître d'elne rendue à l'aide de Cascade Shadow Map

Pour les besoins de ce travail, nous avons utilisé trois groupes consécutifs: proche, moyenne et lointaine. Dans le premier champ, nous avons échantillonné la source lumineuse de la zone en un grand nombre d'échantillons, une carte d'ombre haute résolution est générée pour chaque source lumineuse ponctuelle afin de garantir que les objets les plus proches ont une haute qualité d'ombres (voir Figure 3.5). Dans le deuxième champ, nous avons échantillonné la source lumineuse de zone en un nombre moyen d'échantillons, une carte d'ombre à résolution moyenne est générée pour chaque source lumineuse ponctuelle. Enfin, dans le dernier champ, nous avons échantillonné la source lumineuse de la zone en un petit nombre d'échantillons, une carte d'ombre à petite résolution est générée pour chaque source lumineuse ponctuelle afin de réduire le temps de calcul car nous n'avons pas besoin d'utiliser une haute qualité pour les objets qui sont loin de la caméra, mais si nous le faisons, ce serait une perte de temps de calcul.

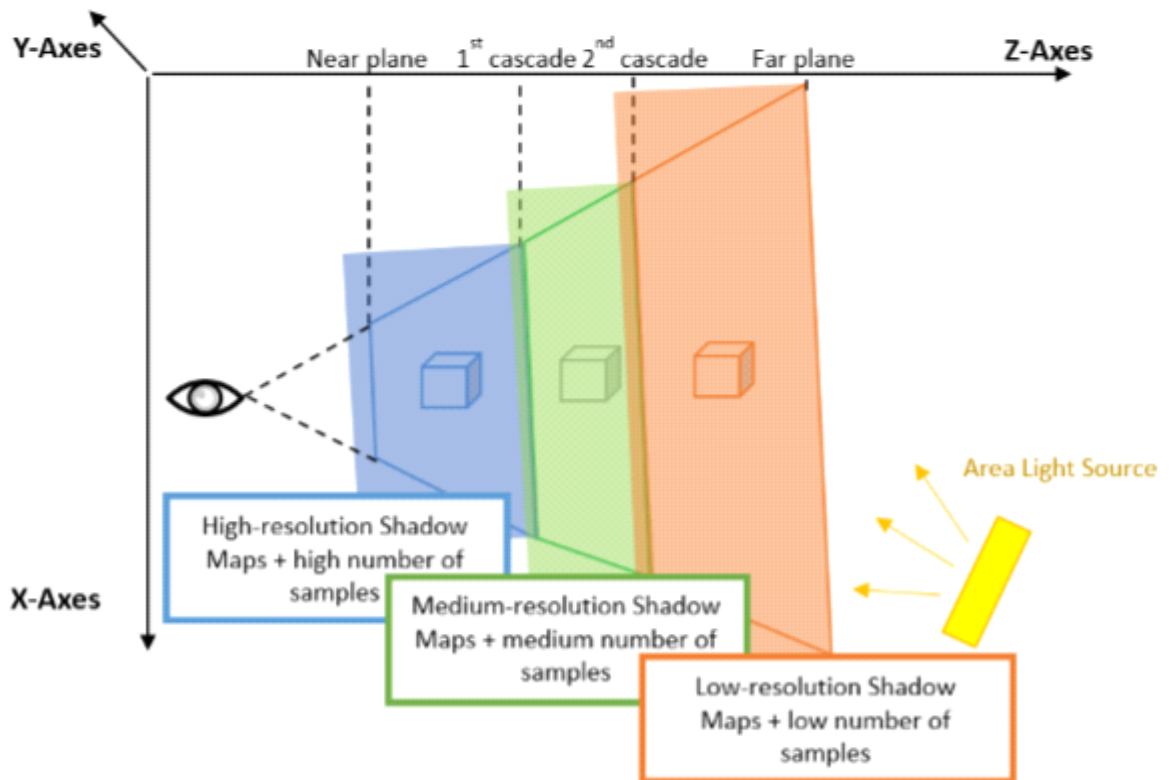


Figure 3.5: Subdiviser la pyramide de vue[38]

2.3 Approximation des ombres douces avec échantillonnage de la source lumineuse de la zone

Dans cette section, nous allons approximer les ombres douces par la technique de subdivision de la source de lumière surfacique

2.3.1 Subdivision de la source lumineuse de la zone

Nous utiliserons la technique de Zerari [35] pour subdiviser uniformément la zone de la source lumineuse. Nous supposons que la source lumineuse de surface est carrée. La première étape consiste toujours à subdiviser la source lumineuse de surface en quatre lumières ponctuelles initiales dans les coins de la source lumineuse de surface carrée, puis d'autres lumières ponctuelles seront créées par l'équation suivante 3.1 si cette lumière de surface doit être subdivisée (voir Figure 3.5).

$$n = k^2 \quad (3.1)$$

Où :

n : est le nombre de sources lumineuses ponctuelles,

$k \times k$: est le noyau de subdivision uniforme

$k \geq 2$.

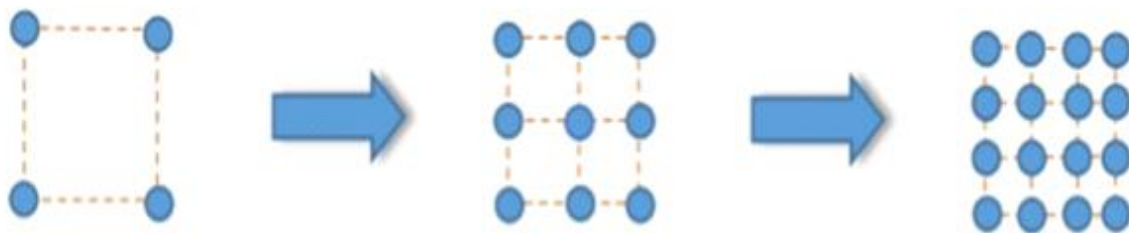


Figure 3.6. Subdivision de la source lumineuse de la zone.

2.4 Rendu de la carte d'ombre

Le calcul d'ombre consiste à identifier les parties de la scène qui sont cachées à la source lumineuse. Il s'agit donc intrinsèquement d'un calcul de visibilité du point de vue de la lumière. Dans ce qui suit, les shadow maps sont produites à partir de chaque point lumineux (échantillon)

de chaque champ en utilisant la technique de shadow mapping avec une projection orthographique. Pour chaque lumière ponctuelle de chaque cascade, une carte d'ombre est créée, après avoir subdivisé la source lumineuse de la zone en un ensemble de n lumières ponctuelles, donc n cartes d'ombre seront créées. La génération d'ombres douces avec plusieurs cartes d'ombres par lumière est coûteuse en calcul en raison de la densité d'échantillonnage élevée qui est nécessaire pour rendre les régions de pénombre lisses et visuellement attrayantes. Si la densité est trop faible, des artefacts de bandes sont susceptibles d'apparaître et le système visuel humain ne perçoit plus une ombre douce, mais plusieurs ombres dures.

2.5 Évaluation des informations sur la carte d'ombre

Les cartes d'ombre doivent être évaluées dans une passe de rendu d'illumination, cette étape est identique à la deuxième passe de rendu dans l'algorithme de mappage d'ombre standard et vient après avoir toutes les cartes d'ombre nécessaires. Dans cette section, nous montrons comment résoudre les obstacles qui peuvent apparaître en raison des différentes profondeurs de subdivision, et en raison du grand nombre de textures de profondeur qui doivent être échantillonnées.

2.5.1 Attribuer des pondérations de contribution à Shadow Map

Nous appliquons des poids aux points d'échantillonnage. Dans les zones de nombreuses subdivisions, les échantillons individuels se voient attribuer un poids plus petit et ne contribueront pas autant à l'obscurité de la pénombre que ceux avec un poids important. Dans le cas d'une source lumineuse surfacique subdivisée selon la technique de Zerari [35], les poids w_i attribués à la i ème carte d'ombre sont calculés avec:

$$w_i = 1/k^2$$

Où :

k^2 : est le nombre d'échantillons,

La somme de tous les poids est 1.

Sinon, les poids doivent être normalisés pour s'assurer que les valeurs d'ombre accumulées finales se situent entre 0 (entièrement éclairé) et 1 (entièrement ombragé).

2.5.2 Visualisation des ombres douces à l'aide de n Shadow Maps

La carte d'ombre dans l'algorithme de mappage d'ombre est évaluée dans un deuxième passage du point de vue de la caméra en utilisant les données du premier passage pour le calcul d'ombre pour calculer si les fragments sont dans l'ombre en fonction de leur occlusion. Rétroprojetez chaque pixel dans le pixel shader sur chacun des troncs de lumière ponctuelle en comparant la profondeur de ce pixel particulier à partir des points de vue de la lumière pour calculer l'ombre douce. Si nous obtenons une valeur (0 ou 1) selon que le pixel est dans l'ombre ou non par rapport à une source lumineuse ponctuelle particulière i , ces valeurs sont multipliées par leur poids w_i et additionnées, ce qui donne une estimation du pourcentage d'occlusion.

L'utilisation d'un nombre élevé de shadow maps peut entraîner des problèmes dus à des limitations techniques : par exemple, dans certaines API de rendu plus anciennes, il n'est pas possible d'échantillonner plus de 16 textures en une seule passe de rendu. Cela limite le nombre théorique de cartes d'ombre à 16. Le système de rendu différé introduit par [37], ainsi qu'un soi-disant tampon d'accumulation, est un moyen de résoudre ce problème, qui est un tampon d'espace écran avec un seul canal de données. Pour chaque shadow map, nous rendons la scène dans une passe de rendu séparée. Au lieu d'utiliser la valeur d'ombre dure obtenue d'un fragment d'espace d'écran $f(x, x_0)$ directement pour l'éclairage, nous la multiplions par son poids et l'ajoutons au tampon d'accumulation à la position $f_{\text{accumulationBuffer}}(x, x_0)$.

Toutes les cartes d'ombre ont été évaluées et le tampon d'accumulation est rempli après n passes de rendu. Maintenant, dans une dernière passe de rendu, la scène est éclairée : pour chaque fragment d'espace d'écran $f(x, x_0)$, la valeur de tampon d'accumulation correspondante $f_{\text{accumulationBuffer}}(x, x_0)$ est échantillonnée et utilisée comme pourcentage d'occlusion.

2.5.3 Évaluation des ombres douces à l'aide de tableaux de texture

L'une des extensions utiles introduites dans OpenGL est "Array Textures". Cette extension introduit l'idée de textures de tableau à une et deux dimensions. Un tableau de texture est une collection d'images unidimensionnelles et bidimensionnelles de taille et de format identiques, organisées en couches. `TexImage2D` est utilisé pour spécifier une texture de tableau à une dimension, où la hauteur spécifie le nombre de couches pour la texture de tableau. De

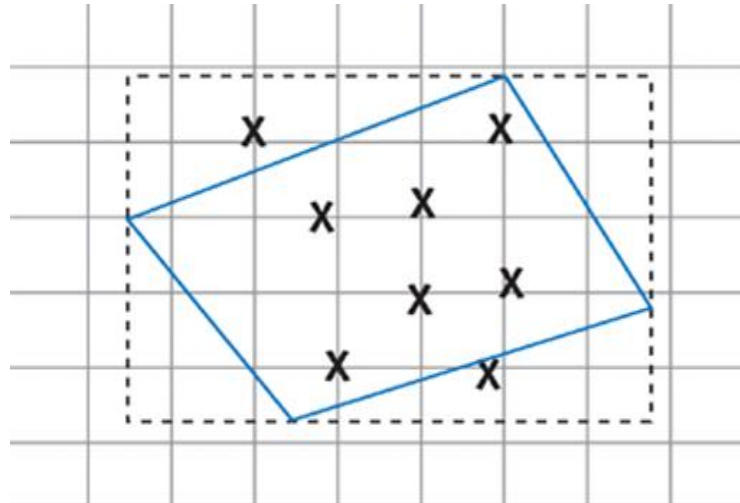
même, TexImage3D est utilisé pour déclarer une texture de tableau à deux dimensions, où la profondeur spécifie le nombre de couches de textures 2D. Array Textures nous permet d'envoyer jusqu'à 512 textures.

De cette façon, l'évaluation des shadow maps en plusieurs passes n'est plus nécessaire. Rendre toutes les cartes d'ombres accessibles dans un seul shader en utilisant les fonctionnalités Arrays Texture sert parfaitement notre objectif. Dans le cas de textures de tableau 2D, un calque est sélectionné en spécifiant la coordonnée de texture r. De plus, on y accède comme s'il s'agissait d'une texture unidimensionnelle ou bidimensionnelle. Les recherches de texture ne filtrent pas entre les calques, bien que la même chose puisse être obtenue dans les shaders. Le rendu en textures 2D ne peut être réalisé qu'en liant une texture de tableau 2D à un objet tampon de trame. De plus, l'utilisation d'un tampon d'accumulation n'est plus nécessaire : nous obtenons la valeur d'occlusion du fragment en additionnant simplement les valeurs d'ombres dures multipliées par leurs poids correspondants. Cette valeur peut être directement utilisée dans la même passe pour modifier l'éclairage. La valeur d'occlusion du fragment courant peut ainsi être obtenue sans rajouter de passes, en économisant n opérations de lecture/écriture ainsi que la mémoire précédemment consommée par le tampon d'accumulation.

2.5.4 Filtrage PCF « Percentage-Closer Filtering »

Contrairement aux textures normales, les textures de carte d'ombre ne peuvent pas être préfiltrées pour supprimer le crénelage. Au lieu de cela, plusieurs comparaisons de cartes d'ombre sont effectuées par pixel et moyennées ensemble. Cette technique est appelée filtrage en pourcentage plus proche (PCF) car elle calcule le pourcentage de la surface qui est plus proche de la lumière et donc pas dans l'ombre.

L'algorithme PCF original, décrit dans Reeves et al. 1987, a appelé à cartographier la région à ombrager dans l'espace de la carte d'ombre et à échantillonner cette région de manière stochastique (c'est-à-dire de manière aléatoire). L'algorithme a d'abord été implémenté à l'aide du moteur de rendu REYES, de sorte que la région à ombrer signifiait un micropolygone à quatre côtés. La figure 11-1 montre un exemple de cette implémentation



La figure 3.7 :une région de micropolygone à quatre côtés.(L'algorithme PCF)

3. Mise en œuvre

Nos expérimentations sont implémentées sur une station graphique qui a les configurations suivantes :

Configuration matérielle: La station graphique est équipée d'un processeur Intel(R)

Processeur Core i35005U 2.16 GHZ, une RAM de 2.00 GO, et une carte graphique Intel(R) HD Graphics 5500.

Configuration logicielle: Pour la réalisation de notre application, nous avons utilisé les outils logiciels et plateformes suivants

- Système d'exploitation : Windows 08 professionnel, 32 bits
- Langage de programmation : Visual Studio C++ 2010
- API graphique : OpenGL 2.02.

Nous avons implémenté la technique de soft shadow mapping décrite dans la section 3.2, et les deux méthodes décrites dans la section 3.2.5 (3.2.5.2 et 3.2.5.3) en utilisant une source de lumière rectangulaire bidimensionnelle et différentes résolutions de shadow map, et appliqué un filtre PCF avec différents noyaux changeant à chaque fois (5*5, 3*3, 2*2) avec calcul du temps d'exécution dans chaque cas

Nous avons déclaré les textures 2D-Array comme suit :

```
//////Générer une texture opengl
```

```
glGenTextures(1, &ShadowMap);
```

```
glBindTexture(GL_TEXTURE_2D_ARRAY, ShadowMap)
```

```
//Mise en place des paramètres.
```

```
glTexParameteri(GL_TEXTURE_2D_ARRAY, GL_TEXTURE_MIN_FILTER, GL_LINEAR);  
glTexParameteri(GL_TEXTURE_2D_ARRAY, GL_TEXTURE_MAG_FILTER, GL_LINEAR);  
glTexParameteri(GL_TEXTURE_2D_ARRAY, GL_TEXTURE_WRAP_S, GL_CLAMP_TO_EDGE);  
glTexParameteri(GL_TEXTURE_2D_ARRAY, GL_TEXTURE_WRAP_T, GL_CLAMP_TO_EDGE);  
glTexParameteri(GL_TEXTURE_2D_ARRAY, GL_TEXTURE_COMPARE_MODE,  
GL_COMPARE_R_TO_TEXTURE);
```

```
glTexParameteri(GL_TEXTURE_2D_ARRAY, GL_TEXTURE_COMPARE_FUNC, GL_LEQUAL);  
glTexParameteri(GL_TEXTURE_2D_ARRAY, GL_DEPTH_TEXTURE_MODE, GL_INTENSITY);
```

```
//Créer de l'espace pour la texture du tableau.
```

```
/*La profondeur dans notre cas est le nombre de cartes d'ombre (chaque source de lumière ponctuelle a  
une carte d'ombre)*/
```

```
glTexImage3D(GL_TEXTURE_2D_ARRAY, 0, GL_DEPTH_COMPONENT24, ShadowResolution,  
ShadowResolution, Samples , 0, GL_DEPTH_COMPONENT, GL_FLOAT, NULL);
```

Nous évaluons les cartes d'ombre dans une deuxième passe de rendu en utilisant les données de la première passe pour le calcul d'ombre au shader de fragment. . Tout en évaluant le pourcentage d'occlusion du fragment actuel, nous filtrons les ombres dures de chaque source lumineuse ponctuelle à l'aide d'un filtre PCF pour brouiller les bords des ombres dures afin de supprimer les artefacts. À la fin, nous obtenons des ombres douces de haute qualité dans la première cascade, et dans la deuxième et dernière cascade, nous obtenons des ombres douces avec moins de qualité en raison du nombre d'échantillons dans cette cascade et de la résolution des cartes d'ombres.

Nous représentons ici une partie de notre code fragment shader correspondant à ce que nous avons expliqué précédemment(3 noyaux différents) :

```
#extension GL_EXT_texture_array : enable // activation du tableau de texture  
  
uniform sampler2D Texture;  
  
uniform sampler2DArrayShadow ShadowMapCascade1;  
  
for(int i = 0; i < 9; i++){  
  
    ShadowTexCoord1[i] = LightTextureMatricesCascade1[i] * vert;}
```

```

float ShadowCascade1=0.0 ; ombre = 0;

if(po<=(A)){float ShadowCascade1=0.0 ; ombre = 0;

    for(int i = 0; i < 9; i++){ // Filtre PCF utilisant le noyau 5*5

        for (y = -2 ; y <=2 ; y+=1.0) // 5*5

            for (x = -2 ; x <=2 ; x+=1.0) // 5*5

                ombre += shadow2DArray(ShadowMapCascade1, vec4(ShadowTexCoord1[i].xy /
                ShadowTexCoord1[i].w, i, ShadowTexCoord1[i].z / ShadowTexCoord1[i].w)+ vec4(x *
                xPixelOffset1 * ShadowTexCoord1[i].w, y * yPixelOffset1 * ShadowTexCoord1[i].w, 0.005, 0.0)).r;
                ombre /= 25.0 ; // 5*5////////fonction de recherche

    }

    ShadowCascade1 += ombre;}

ShadowCascade1 /= 9.0;

float NdotLD = max(dot(normalize(LightDirection), Normal), 0.0) * ShadowCascade1;

float Spec = pow(max(dot(normalize(LightDirectionReflected), normalize(CameraDirection)), 0.0),
2500.0) * ShadowCascade1;

float diff = max(dot(normalize(LightDirection), Normal), 0.0);

if(sol==0)

gl_FragColor = vec4( ShadowCascade1,ShadowCascade1,ShadowCascade1,1.0)*diff*couleur+ia +
vec4(0.1, 0.0, 0.0 0.0);} // Filtre PCF utilisant le noyau 3*3

if(po<=(B) && po>=(A)){

    float ShadowCascade1=0.0 ;

    ombre = 0;

    for(int i = 0; i < 9; i++){ // Filtre PCF utilisant le noyau 3*3

        for (y = -1.0 ; y <=1.0 ; y+=1.0) // 3*3

            for (x = -1.0 ; x <=1.0 ; x+=1.0 ) // 3*3

                ombre += shadow2DArray(ShadowMapCascade1,
                vec4(ShadowTexCoord1[i].xy / ShadowTexCoord1[i].w, i, ShadowTexCoord1[i].z /
                ShadowTexCoord1[i].w)+ vec4(x * xPixelOffset1 * ShadowTexCoord1[i].w, y * yPixelOffset1 *
                ShadowTexCoord1[i].w, 0.005, 0.0)).r;

    }

    ombre /= 9.0 ; // 3*3////////fonction de recherche
}

```

```

        ShadowCascade1 += ombre;}

    ShadowCascade1 /= 9.0;

float NdotLD = max(dot(normalize(LightDirection), Normal), 0.0) * ShadowCascade1;

float Spec = pow(max(dot(normalize(LightDirectionReflected), normalize(CameraDirection)), 0.0),
2500.0) * ShadowCascade1;

    float diff =  max(dot(normalize(LightDirection), Normal), 0.0);

        if(sol==0)

gl_FragColor = vec4( ShadowCascade1,ShadowCascade1,ShadowCascade1,1.0)*diff*couleur+ia +
vec4(0.0, 0.0, 0.1, 0.0);} ////////////// écran d'affichage

    if(po>=(B)){

        float ShadowCascade1=0.0 ;

        ombre = 0;

            for(int i = 0; i < 9; i++){ ////////////// Filtre PCF utilisant le noyau 2*2

                for (y = -0.5 ; y <=0.5 ; y+=1.0) // 2*2

                    for (x = -0.5 ; x <=0.5 ; x+=1.0) // 2*2

        ombre += shadow2DArray(ShadowMapCascade1, vec4(ShadowTexCoord1[i].xy /
ShadowTexCoord1[i].w, i, ShadowTexCoord1[i].z / ShadowTexCoord1[i].w)+ vec4(x *
xPixelOffset1 * ShadowTexCoord1[i].w, y * yPixelOffset1 * ShadowTexCoord1[i].w, 0.005, 0.0)).r;

        ombre /= 4.0 ; // 2*2//////////fonction de recherche

    ShadowCascade1 += ombre;}

        ShadowCascade1 /= 9.0;

float NdotLD = max(dot(normalize(LightDirection), Normal), 0.0) * ShadowCascade1;

float Spec = pow(max(dot(normalize(LightDirectionReflected), normalize(CameraDirection)), 0.0),
2500.0) * ShadowCascade1;

    float diff =  max(dot(normalize(LightDirection), Normal), 0.0);

        if(sol==0){

gl_FragColor  = vec4( ShadowCascade1,ShadowCascade1,ShadowCascade1,1.0)* diff* couleur+ia
+ vec4(0.0, 0.1, 0.0, 0.0);} ////////////// écran d'affichage

```

3.1 Résultats et évaluation

Dans cette section, nous présenterons les résultats de la mise en œuvre où différentes mesures de performance et de qualité visuelle seront comparées et revues. Pour comparer nos résultats, nous avons choisi le critère standard de FPS (Frame Per Second) et le temps nécessaire pour générer une image. Nous avons évalué la technique proposée en utilisant le T (temps de calcul). Nous validons nos résultats qualitativement et quantitativement sur des images de scènes en temps réel.

Nous présentons quelques résultats obtenus en utilisant la technique proposée pour rendre des ombres douces physiquement correctes basées sur l'échantillonnage de la source lumineuse de la zone et en divisant l'angle de vision de la caméra en sections séparées pour augmenter la qualité de l'ombre des objets proches et réduire le temps de calcul. Nous avons comparé les images obtenues entre elles et l'effet des différents noyaux PCF sur l'amélioration des ombres douces et le réalisme de l'image tout en atteignant le temps réel.

3.2 Résultats visuels

Dans cette partie, nous présentons ci-dessous les résultats de notre algorithme sur différentes scènes utilisant différentes configurations. Nous testons également la validité de la technique proposée en termes de qualité et de temps de calcul et nous discutons enfin les points forts et faibles à l'aide de métriques perceptives.

Pour vérifier la précision de la technique proposée, nous comparons les résultats finaux obtenus.

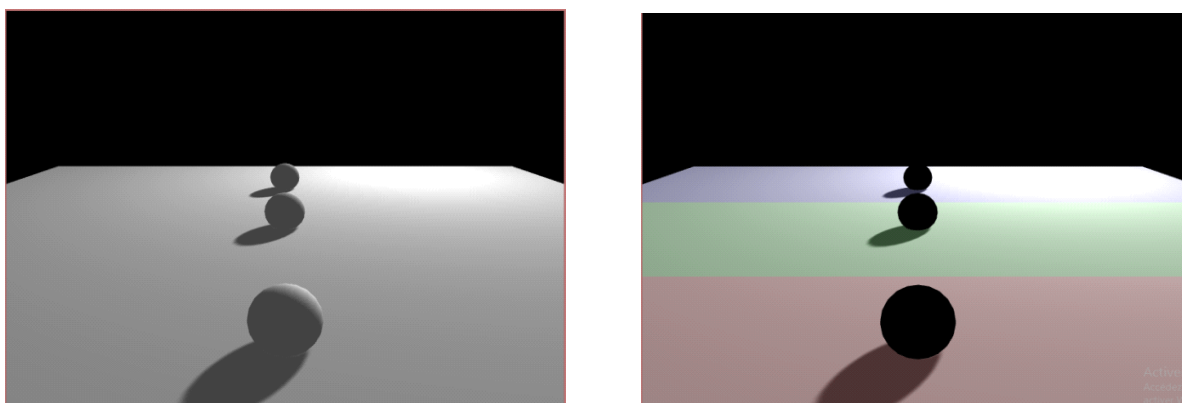
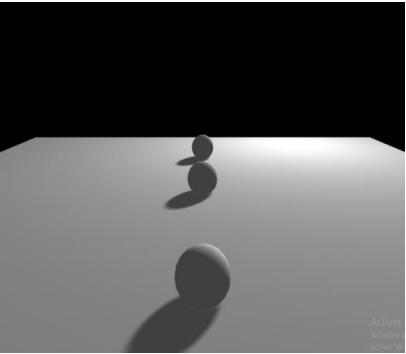
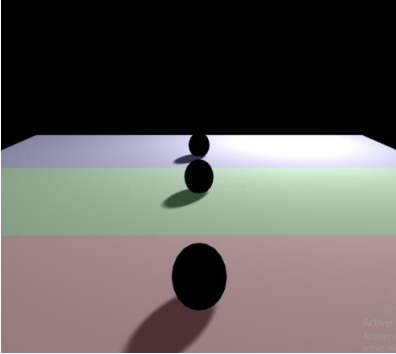
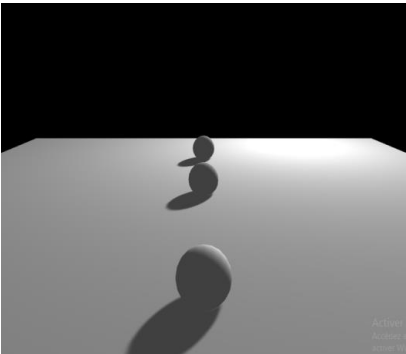
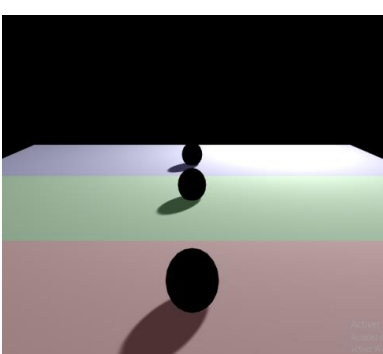


Figure 3.8: Utilisez un multifiltre PCF. (A chaque champ un taux de filtre spécifique)

La figure (3.8) montre le résultat de la division dépendante (3.2.1) montrant 3 sous-couches, nous plaçons une sphère dans chaque sous-couche. Nous avons utilisé un filtre PCF spécifique pour chaque sous-couche. Le premier champ est défini par (5*5), le deuxième par (3*3) et le troisième (2*2). Plus en s'éloignant de la caméra, on réduit le filtre PCF. L'utilisation de ces différents filtre à permis d'accélérer le temps de calcul. La figure de gauche est la même image avec les zones de division masquées, dans la première sous-couche, le champ est éclairé par 16 sources lumineuses ponctuelles à l'aide d'une carte d'ombre par un point lumineux de résolution 1024. Dans le deuxième sous-couche, le champ est éclairé par 9 sources lumineuses ponctuelles et 512 cartes d'ombres précises et dans la dernière sous-couche est éclairée par 4 sources lumineuses ponctuelles utilisant 128 cartes d'ombres précises, avec un FPS égal à 7 et un temps de calcul estimé à 0,28 ms.

Ce tableau résume les résultats obtenus:

modèle		Pcf	Résultat
		5*5	Fps :2 T :0.39ms
		3*3	Fps :7 T :0.13ms

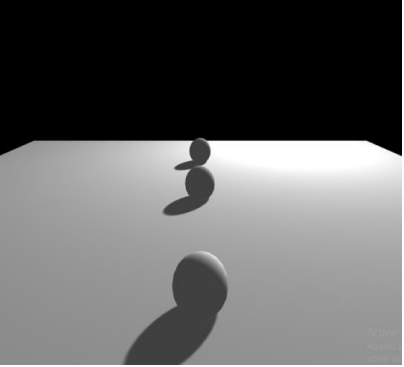
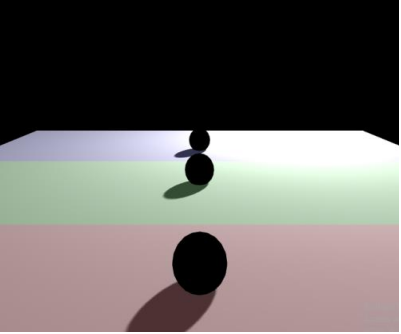
		2*2	Fps :14 T :0.07ms
---	--	-----	----------------------

Figure 3.9 :Tableau des images d'une sphère avec différents noyaux de PCF

Dans ce tableau, nous nous sommes appuyés sur la fixation du ratio pcf dans chaque image dans les trois d'image. Dans la première image, nous l'avons implémenté avec un pcf (5 * 5), la seconde était pcf (3 * 3) et la dernière était pcf (2 * 2). Il existe une relation directe entre le temps de calcul et le pourcentage de fps. Plus le temps de calcul diminue en fps et vice versa, et nous avons également remarqué que plus les choses s'éloignent de la caméra, plus ou moins la différence d'utilisation du filtre pcf n'affecte pas l'intégrité de l'image, ce qui nous a permis de réduire le temps de calcul sans compromettre la qualité de l'image.

En comparant l'image (3.9) avec les résultats du tableau, on remarque qu'en utilisant un rapport pcf élevé (5*5), on obtient des ombres lisses sans d'alliasage, mais avec une longue durée dans le calcul, $t = 0,39ms$; et lors de l'utilisation d'un rapport pcf bas, les nuances n'étaient pas de bonne qualité. Élevé, mais c'était avec les choses proches de la caméra comme pour les plus éloignées. Il n'y a pas de différence entre eux et un filtre à rapport élevé et un filtre bas où le temps de calcul $t = 0,07ms$, c'est ce sur quoi nous avons travaillé si nous l'avons divisée en champs pour allouer à chaque champ un ratio pcf approprié dans le champ près de la caméra, nous fixons le pcf à une grande valeur (5*5) et pour les portées lointaines nous avons réduit la valeur du pcf et ainsi préservé la qualité de l'image, notamment la qualité des ombres, et réduit le temps de calcul pour être adapté à la visualisation en temps réel.

Nous vous présentons ce tableau Avec pcf fixe(5*5) et défèrent modèles avec défèrent taille :

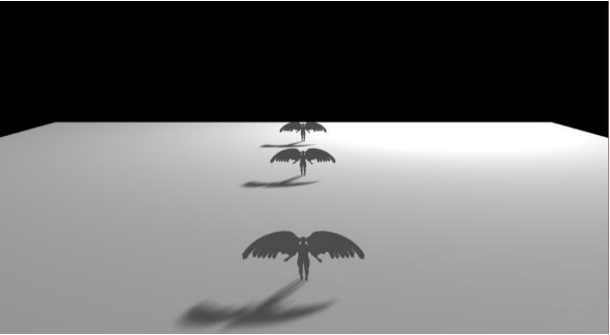
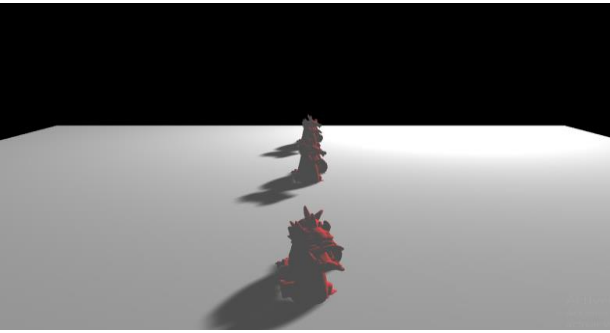
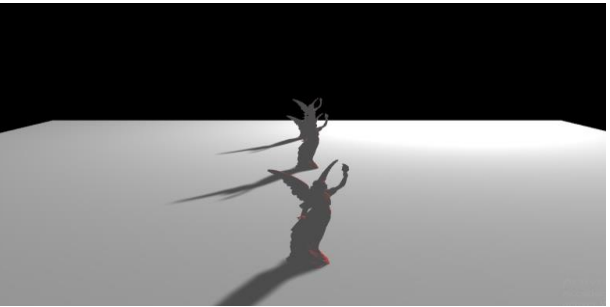
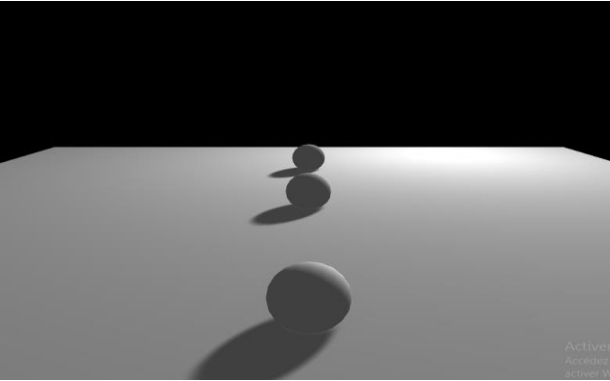
modèle	Taill de modèle	Résultat	Multer pcf
	13,1 Mo	Fps :0 T =2.54ms	T= 2.42ms
	6,70 Mo	Fps :0 T :2.39ms	T = 2.43ms
	3,47Mo	Fps :1 T :0.93ms	T = 0.75ms
	octuor glutSolidS phere	Fps :2 T :0.39ms	T = 0.28ms

Figure 3.10Tableau pour différents objets avec différentes tailles et effet de multi pcf

Dans ce tableau, nous nous sommes appuyés sur un changement des formes utilisées avec leurs différentes tailles. Plus la taille du modèle été grande, et le temps de calcul était long. En revanche, avec la technique proposée, nous obtenions beaucoup moins de temps sans aucune différence en qualité d'image.

4. Conclusion

A travers cet projet, nous avons présenté une méthode pour rendre des ombres douces précises en utilisant la technologie de mappage des ombres à l'aide d'un filtre multi-pcf pour lisser l'image et la rendre plus réaliste, ainsi que pour réduire le temps de calcul dans notre application de test, nous ont pu fournir des ombres douces de haute qualité, Cette méthode prouve son efficacité si nous pouvons correctement diviser l'image en champs en fonction de sa proximité ou de sa distance avec la caméra et déterminer le rapport pcf approprié pour chaque champ.

Conclusion

Le rendu réaliste est devenu incontournable dans l'industrie, la simulation et les jeux vidéo. Cela nécessite de simuler l'interaction entre la lumière et les objets dans une scène 3D, un calcul connu sous le nom d'illumination globale est souvent très coûteux en calcul. En effet, pour générer une image, le calcul principal a été formalisé dans une équation dite équation de rendu. Cette équation reflète la conservation de l'énergie dans le transport de la lumière. elle ne peut être obtenue que par approximations. L'un des effets compliqués les plus importants simulés par ces approximations est celui des ombres douces.

Contrairement aux ombres dures, les ombres douces ne sont pas projetées par des lumières ponctuelles sans étendue, mais par des sources lumineuses de surface. Ils sont donc constitués de régions d'ombre et de pénombre, et malgré le problème des coûts de calcul accrus causés par le grand nombre d'échantillons nécessaires pour obtenir une bonne qualité d'ombres douces, leur utilisation en vaut la peine. Presque toutes les ombres ont en réalité des limites douces, donc l'utilisation d'ombres douces dans les applications de rendu augmente considérablement le réalisme des images générées.

Des ombres douces et subtiles peuvent être simulées physiquement dans les applications de synthèse d'images en échantillonnant et en agrégeant plusieurs échantillons d'une source lumineuse de zone entière, de sorte que chaque échantillon représente une source lumineuse ponctuelle. Dans ce travail, nous avons proposé une solution aux problèmes d'ombres douces subtiles dans les applications en temps réel. La contribution que nous avons apportée est de concevoir une méthode pour calculer efficacement les ombres douces physiquement précises causées par l'échantillonnage d'une source lumineuse de région dans un petit nombre d'échantillons à l'aide de la technologie Zerari qui est basée sur la division uniforme de la source lumineuse de région et le remplacement de l'effet pour un grand nombre d'échantillons par un filtre multi-PCF, Pour réduire l'échantillonnage de la source lumineuse de surface dans les objets éloignés de la caméra afin de réduire le temps de calcul et d'améliorer la qualité de l'image présentée. Les résultats expérimentaux montrent que notre contribution peut améliorer l'efficacité et produire des nuances de haute qualité à grande échelle avec des performances en temps réel.

Comme travaux futurs, nous suggérons de réduire le temps de calcul requis pour l'étape de comparaison qui est la deuxième passe dans l'algorithme de carte d'ombre qui coûte trop cher en

réduisant le nombre de requêtes d'occlusion. Nous pouvons y parvenir en limitant les recherches sur les cartes d'ombre des échantillons de source lumineuse de zone pour la seule zone de pénombre qui a besoin du pourcentage d'occlusion pour obtenir des ombres douces, et rendre l'ombre de la zone d'ombre à partir d'une source lumineuse ponctuelle de la source lumineuse de zone . Cela conduit automatiquement à limiter le filtrage aux seules régions avec des artefacts. Cette suggestion augmentera les performances de rendu des ombres douces physiquement précises.

Bibliographie

- [1] « Occultation ambiante basée sur l'échantillonnage préférentiel multiple pour la génération des ombres douces en temps réel » Abd El Mouméne Zerari ; Thèse de docteur en sciences Université Mohamed Khider BISKRA 2018
- [2] « une alternative pour le rendu d'ombres en temps réel » Julien Gerhards ; thèse de docteur de l'université de limoges 2017
- [3] « complexité géométrique et stratégie de partitionnement » Frédéric Mora et Djamchid Ghazanfarpour, thèse de docteur de l'université de limoges 2019
- [4] « star eurographics2003 » « ombres en temps réel »
- [5] « gestion des ombres en temps réel » dupuis Emmanuel, travail d'étude et de recherche master STIC-informatique, 2003
- [7] « a General version of crow shadow volumes » philippe bergeron, published in IEEE computer graphics and applications
- [8] « a Survey of shadow algorithms » anderew woo. Pierre poulin and alain frounrier ,LEEE computer graphics & applications
- [9] « real-time soft shdows using temporal coherence » daniel scherzer, michael schwarzler, oliver mattauch and michael wimmer ; in visual computing : 5th international symposium on visual computing .
- [10] « me and my (fake) shadow » jim blinn ; LEEE computer graphics & applications
- [11] « output sensitive extraction of silhouettes from polygonal geometry » benichou fabien and gershon elber ; 7th pacific conference on computer graphics and applications 1999
- [12] « polygon shadow generation » atherton peter , kevin weiler and donald gree nberg ; program of computer graphics 1979
- [13] « radiosity and realistic image synthesis » michael F. cohen and john R. wallace ; academic press professional, USA 1993
- [14] « photon maps in bidirectional monte carlo ray tracing of complex objects » henrik wann jensen and niels jorgen chistensen , computer & graphics 1995
- [15] « an improved illumination model for shaded display » turner whitted ; magazine communications of the ACM 1980.
- [16] « casting curved shadows on curved surfaces » lance williams, ACM SIG GRAPH 1978
- [17] « fast shadows and lighting effects using texture mapping » mark segal , carl korobkin, rolf van widentft, jim foran and paul E. haeberli ; in proceedings SIGGRAPH 1992

- [18] « adaptive shadow maps » Randima Fernando, Sebastian Fernandez, Kavita Bala and Donald P. Greenberg. In SIGGRAPH 01 ; the 28th annual conference on Computer graphics and interactive techniques 2001
- [19] « Shadow algorithms for computer graphics » Franklin C. Crow. on Computer graphics and interactive techniques SIGGRAPH 77, 1977
- [20] « A hierarchical shadow volume algorithm » Timo Aila and Tomas Akenine-Möller , In HWWS '04 Proceedings of the ACM ; Eurographics Association 2004
- [21] « qu'est-ce qu'une ombre » sanchari chakraborti ; article de LAMBDA GEEKS page web
- [22] « A survey of real-time soft shadows algorithms » J-M Hasenfratz, M Lapierre, N Holzschuch and F Sillion ; 2003
- [23] « The rendering equation » James T Kajiya ; 13th annual conference on Computer graphics and interactive techniques, 1986
- [24] « Penumbra Maps: Approximate Soft Shadows in Real-Time » Chris Wyman and Charles D. Hansen, the 14th Eurographics workshop on Rendering Techniques, 2003
- [25] « Percentage-closer Soft Shadows » Randima Fernando, In: ACM SIGGRAPH 2005 Sketches and Applications
- [26] « Rendering Antialiased Shadows with Depth Maps » William T. Reeves, David H. Salesin, and Robert L. Cook ; the 14th annual conference on Computer graphics and interactive techniques (1987)
- [27] « Real-time Soft Shadow Mapping by Backprojection. » Gaël Guennebaud Loïc Barthe and Mathias Paulin, In Eurographics Symposium on Rendering (EGSR 2006).
- [28] « High-Quality Adaptive Soft Shadow Mapping » Gaël Guennebaud Loïc Barthe and Mathias Paulin, Computer Graphics Forum, Eurographics 2007 proceedings
- [29] « Soft Shadow Maps: Efficient Sampling of Light Source Visibility » Lionel Atty, Nicolas Holzschuch, Marc Lapierre, Jean-Marc Hasenfratz, François X. Sillion and Hansen Charles, Computer Graphics Forum 2006
- [30] « Fast Accurate Soft Shadows with Adaptive Light Source Sampling » Michael Schwärzler, Oliver Mattausch, Daniel Scherzer and Michael Wimmer, the 17th International Workshop on Vision, Modeling, and Visualization. 2012
- [31] « Extending Graphics Hardware for Occlusion Queries in OpenGL » Dirk Bartz, Michael Meißner and Tobias Hüttner ; the ACM SIGGRAPH/EUROGRAPHICS workshop, 1998.
- [32] « The state of the art in interactive global illumination » Tobias Ritschel, Carsten

Dachsbacher, Thorsten Grosch, and Jan Kautz In Computer Graphics Forum 2012

[33] « Advanced Rendering Techniques»Wolfgang Engel, Charles River Media,2006

[34] « Shadow Map Antialiasing»Michael Bunnell,Fabio Pellacini;GPU Gems Book,2007

[35] « Fast accurate soft shadows with adaptive light source sampling » Michael Schwärzler ,Oliver Mattausch,Daniel Scherzer,and Michael Wimmer.. In Proceedings of the 17th International Workshop on Vision, Modeling, and Visualization2012

[36] « Ombres douces avec source de lumière surfacique» AbdElMouméneZerari, MedBabahenini ,BahiNaima, et SarraKhemliche.2014

[37] « The triangle processor and normal vector shader » Michael Deering, Stephanie Winner, Bic Schediwy, Chris Duffy, and Neil Hunt. ACM Siggraph computer graphics1988

[38] « Cascaded Soft Shadow Maps Based On PCF Filtering »Nadia AZRI,Master Thesis 2020

الجمهورية الجزائرية الديمقراطية الشعبية

وزارة التعليم العالي والبحث العلمي

جامعة محمد خيضر بسكرة

تصريح شرفي

(خاص بالالتزام بقواعد النزاهة العلمية لإنجاز بحث)

أنا الممضي أسفله،

السيد (ة): الصفة:

طالب.....

الحامل لبطاقة التعريف الوطنية رقم: والصادرة بتاريخ:

.....

المسجل بكلية: العلوم الدقيقة وعلوم الطبيعة والحياة

قسم:

والمكلف بإنجاز مذكرة تخرج في الماستر عنوانها:

.....

.....

.....

.....

أصرح بشرفي أنني ألتزم بمراعاة المعايير العلمية والمنهجية ومعايير الأخلاقيات المهنية والنزاهة الأكاديمية المطلوبة في إنجاز البحث المذكور أعلاه.

التاريخ:

توقيع المعني: