Université de Montréal

# Hessian-based Occlusion-Aware Radiance Caching

par
Yangyang Zhao

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

Mémoire présenté à la Faculté des études supérieures
en vue de l'obtention du grade de Maître ès sciences (M.Sc.)
en Informatique

Août, 2014

## RÉSUMÉ

Simuler efficacement l'éclairage global est l'un des problèmes ouverts les plus importants en infographie. Calculer avec précision les effets de l'éclairage indirect, causés par des rebonds secondaires de la lumière sur des surfaces d'une scène 3D, est généralement un processus coûteux et souvent résolu en utilisant des algorithmes tels que le *path tracing* [1986] ou *photon mapping* [1996]. Ces techniques résolvent numériquement l'équation du rendu en utilisant un lancer de rayons Monte Carlo.

Ward et al. [1988] ont proposé une technique nommée *irradiance caching* afin d'accélérer les techniques précédentes lors du calcul de la composante indirecte de l'éclairage global sur les surfaces diffuses. Krivanek [2005] a étendu l'approche de Ward et Heckbert [1992] pour traiter le cas plus complexe des surfaces spéculaires, en introduisant une approche nommée *radiance caching*. Jarosz et al. [2012] et Schwarzhaupt et al. [2012] ont proposé un modèle utilisant le hessien et l'information de visibilité pour raffiner le positionnement des points de la cache dans la scène, raffiner de manière significative la qualité et la performance des approches précédentes.

Dans ce mémoire, nous avons étendu les approches introduites dans les travaux précédents au problème du *radiance caching* pour améliorer le positionnement des éléments de la cache. Nous avons aussi découvert un problème important négligé dans les travaux précédents en raison du choix des scènes de test. Nous avons fait une étude préliminaire sur ce problème et nous avons trouvé deux solutions potentielles qui méritent une recherche plus approfondie.



**Mots clé: éclairage global, cache d'irradiance, cache de radiance, synthèse d'images, lancer de rayon, rendu photoréaliste.**

## ABSTRACT

Efficiently simulating global illumination is one of the most important open problems in computer graphics. Accurately computing the effects of indirect illumination, caused by secondary bounces of light off surfaces in a 3D scene, is generally an expensive process and often solved using algorithms such as path tracing [1986] or photon mapping [1996]. These approaches numerically solve the rendering equation using stochastic Monte Carlo ray tracing.

Ward et al. [1988] proposed irradiance caching to accelerate these techniques when computing the indirect illumination component on diffuse surfaces. Krivanek [2005] extended the approach of Ward and Heckbert [1992] to handle the more complex case of glossy surfaces, introducing an approach referred to as radiance caching. Jarosz et al. [2012] and Schwarzhaupt et al. [2012] proposed a more accurate visibility-aware Hessian-based model to greatly improve the placement of records in the scene for use in an irradiance caching context, significantly increasing the quality and performance of the baseline approach.

In this thesis, we extended similar approaches introduced in these aforementioned work to the problem of radiance caching to improve the placement of records. We also discovered a crucial problem overlooked in the previous work due to the choice of test scenes. We did a preliminary study of this problem, and found several potential solutions worth further investigation.



**Keywords: global illumination, irradiance caching, radiance caching, image synthesis, ray tracing, photo-realistic rendering.**

## CONTENTS

**RÉSUMÉ** . . . .

**ABSTRACT** . . . .

**CONTENTS** . . . .

**LIST OF FIGURES** . . . .

**LIST OF APPENDICES** . . . .

**LIST OF ABBREVIATIONS** . . . .

**DEDICATION** . . . .

**ACKNOWLEDGMENTS** . . . .

**CHAPTER 1: INTRODUCTION** . . . . **1**

1.1 Image Synthesis and Rendering . . . . 1

1.2 Global Illumination . . . . 2

**CHAPTER 2: PREVIOUS WORK IN GLOBAL ILLUMINATION** . . . **4**

2.1 Monte Carlo Integration . . . . 4

2.1.1 Variance Reduction . . . . 5

2.2 Monte Carlo Ray Tracing-based Approach . . . . 5

2.2.1 Path Tracing . . . . 6

2.2.2 Photon Mapping . . . . 6

2.3 Irradiance and Radiance Caching . . . . 7

2.3.1 Irradiance Caching . . . . 8

2.3.2 Radiance Caching . . . . 10

2.3.3 Spherical Harmonics . . . . . 11
2.3.4 Record Placement . . . . . 12

**CHAPTER 3: HESSIAN-BASED OCCLUSION-AWARE RADIANCE CACHING 13**

3.1 Abstract . . . . . 13
3.2 Introduction and Previous Work . . . . . 13
3.3 Background . . . . . 14
3.3.1 Irradiance Caching . . . . . 14
3.3.2 Radiance Caching . . . . . 15
3.4 Record Placement Strategies in Irradiance Caching . . . . . 16
3.4.1 Heuristic-based Approaches . . . . . 17
3.4.2 Error-Estimation-based Approaches . . . . . 17
3.5 Occlusion-Aware Radiance Caching . . . . . 21
3.5.1 Occlusion-Aware Radiance Caching . . . . . 22
3.5.2 Second-Order Interpolation and Extrapolation . . . . . 24
3.5.3 Record Valid Region and Error-Metric . . . . . 25
3.5.4 Gradient and Hessian of Spherical Harmonic Coefficients . . . 29
3.6 Compute the Outgoing Radiance . . . . . 30
3.7 Outgoing Radiance Caching . . . . . 31
3.7.1 View-Independent Record Placement . . . . . 32
3.7.2 View-Dependent Record Placement . . . . . 32
3.8 When Assumptions are Broken . . . . . 34
3.8.1 Demonstration on a Test Case . . . . . 35
3.8.2 Possible Solutions . . . . . 38
3.8.3 Discussion . . . . . 42
3.9 Conclusion and Future Work . . . . . 42

**CHAPTER 4: CONCLUSION . . . . . 44**

**BIBLIOGRAPHY** . . . . . **46**

I.1 Definitions of Spherical Harmonic Basis Function . . . . . xi

I.1.1 Spherical Coordinate System . . . . . xi

I.1.2 Cartesian Coordinate System . . . . . xi

I.2 Derivatives . . . . . xii

I.3 Hessian Matrix . . . . . xiii

## LIST OF FIGURES


1.1 An illustration of a ray tracing. . . . . 1

2.1 A comparison of three sampling techiniques. . . . . 5
2.2 A comparison between bidirectional path tracing and stochastic progressive photon mapping. . . . . 7
2.3 The decomposition of global illumination. . . . . 8
2.4 A image of Cornell Box rendered by Hessian-based irradiance caching strategy. . . . . 10
2.5 Images of Cornell Box scene rendered by Krivanek [2005]'s radiance caching. . . . . 11

3.1 Minimum hit distance strategy in radiance caching. . . . . 16
3.2 Minimum hit distance strategy in irradiance caching. . . . . 17
3.3 The 2D case analyzed by Schwarzhaupt et al. [2012]. . . . . 19
3.4 A reconstructed occlusion free scene for a shade point in Cornell Box. . . . . 20
3.5 The improvement of error using extrapolation. . . . . 24
3.6 The images of simplified Cornell Box: before and after extrapolation. 25
3.7 The images of Cornell Box: before and after extrapolation. . . . . 26
3.8 A comparison of the cache schemes between the first-order error metrics and the second-order error metric. . . . . 27
3.9 A comparison of between incident radiance caching strategy and outgoing radiance caching strategy. . . . . 32
3.10 A comparison of cache scheme of outgoing radiance caching strategy in different scenes. . . . . 33
3.11 A example of fail cases of current radiance caching algorithms. . 34

3.12 This illustration explains why this assumption will cause the artifact we observed above. . . . . 35
3.13 An illustration of the test scene. . . . . 36
3.14 Images of two test scenes rendered using radiance caching. . . . . 37
3.15 A comparison of images rendered with and without the assumption on diffuse surfaces. . . . . 37
3.16 A comparison of images rendered with and without the assumption on glossy surfaces. . . . . 38
3.17 A comparison on a glossy Cornell Box scene showing how missing term affects the results. . . . . 39
3.18 Two images illustrating the contribution from current gradient term and the missing term. . . . . 40

## LIST OF APPENDICES

**Appendix I: Gradient and Hessian of Spherical Harmonic Coefficients x**

## LIST OF ABBREVIATIONS

| | |
|---|---|
| BRDF | Bidirectional reflectance distribution function |
| BSDF | Bidirectional scattering distribution fuction |
| BSSRDF | Bidirectional scattering-surface reflectance distribution function |
| SH | Spherical harmonics |

*I would like to dedicate this thesis to my father who has been a constant source of support and inspiration to me.*

## ACKNOWLEDGMENTS

I would like to thank Derek Nowrouzezahrai, Laurent Belcour, Pierre Poulin, and Wojciech Jarosz for their help during my master's project.

# CHAPTER 1

# INTRODUCTION

A picture is worth a thousand words. Since the beginning of human culture, visual perception has always been one of the most important means for conveying information. After the invention of computers, we have never stopped trying to deliver a vividly virtual world using them.

Computer graphics (CG) is a technology used to generate images using computers. Its development has great influence on many aspects of society and industries, such as films and video games.

## 1.1 Image Synthesis and Rendering

In computer graphics, rendering (also called image synthesis) refers to the process of generating a 2D image from a description of a 3D virtual world. This description includes all the necessary information to synthesize realistic images with rendering algorithms: geometry of objects, surface interactions with light, virtual camera, properties of incident lighting, etc. Geometry information includes the position and shape of all virtual objects. Camera properties include the location of the virtual camera, its view-

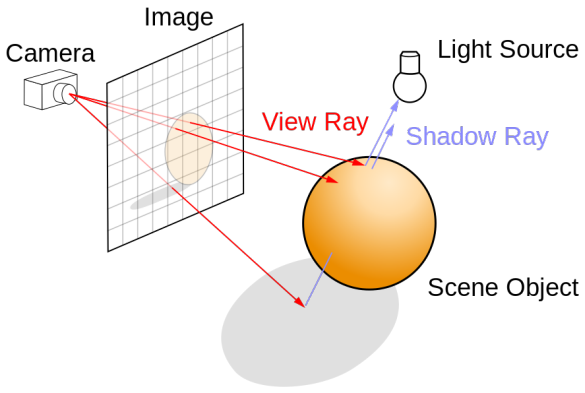


Figure 1.1: An illustration of ray tracing page from Wikipedia.

ing direction, and an image plane divided into pixels. Lighting information includes the position, shape, and emission of light sources.

A rendering algorithm gathers all the aforementioned information and calculates the color of each pixel in the final 2D image. Several algorithms have been developed such as rasterisation, ray casting, and ray tracing.

**Rasterisation.** Rasterisation is the rendering technique currently used by most video games. It geometrically projects objects from the scene onto the image plane, and modifies the pixels it affects. Usually it is faster than pixel-by-pixel approaches, but the light transport effects that it can simulate are limited.

**Ray Casting.** Ray casting is a basic rendering algorithm in computer graphics. It traces rays from the viewpoint to the scene through the image plane, and computes the closest intersection with objects. Using the information of the intersection, it is able to compute the contribution of this ray to the pixel in the image plane.

**Ray Tracing.** Ray tracing is similar to ray casting, the main difference is that ray tracing recursively traces rays from the intersection of primary rays. Ray tracing generates images by shooting rays from the camera and recursively tracing paths of rays while simulating light transport. Ray tracing algorithms are usually used to produce more photo-realistic images. Compared to other rendering algorithms, such as ray casting and rasterisation, ray tracing is able to physically simulate light transport effects that are crucial to generating realistic images. Unfortunately, due to the large number of rays that need to be traced, this technique is far from interactive.

## 1.2 Global Illumination

Image synthesis has shifted towards a physically based rendering model where light transport is simulated using physical rules. This involves many aspects of rendering.

Light interaction with surfaces in the scene is usually described by the bidirectional reflectance distribution function (BRDF). A BRDF is a 4D function that takes incident and outgoing directions as input, thus describing the reflectance of a surface. Because it cannot simulate phenomena such as refraction and subsurface scattering, more sophisticated functions have been introduced: bidirectional scattering distribution function (BSDF) and bidirectional scattering-surface reflectance distribution function (BSSRDF).

In this thesis, we will focus on another very important aspect: global illumination. Global illumination refers to algorithms and techniques dedicated to accurately compute the lighting in a 3D scene.

For an arbitrary point $x$, the radiance scattered towards direction $\omega_o$ can be computed by the integral of the contribution from hemisphere, plus the radiance it emits. This can be expressed mathematically using the "rendering equation" proposed by Kajiya [1986]:

$$L(x,\omega_o) = L_e(x,\omega_o) + \int_\Omega L(x,\omega_i)\, f_r(x,\omega_i,\omega_o)\, (n\cdot\omega_i)\, d\omega_i \tag{1.1}$$

where $\omega_i$ and $\omega_o$ are the directions of incident lighting and towards the camera, $n$ is the surface normal direction, $f$ is the BRDF, $L_e$ is the radiance emitted toward the camera, and $L(x,\omega_i)$ is the incident radiance from direction $\omega_i$ (this term is computed by recursively using rendering equation).

Due to the complexity of the rendering equation, global illumination is very expensive to compute and nearly impossible to solve analytically. A lot of research has been dedicated to solve this problem, and we will give an overview of the work related to this thesis in the next chapter.

# CHAPTER 2

# PREVIOUS WORK IN GLOBAL ILLUMINATION

## 2.1 Monte Carlo Integration

In this thesis, we heavily rely on the Monte Carlo integration to numerically approximate integrals. Monte Carlo integration is proposed by Metropolis and Ulam [1949] to estimate the integral with a set of random samples. The random samples are generated according to the probability density function (PDF).

Given an arbitrary function $f(x)$ we want to integrate in the subset $\Omega$.

$$F = \int_{x \in \Omega} f(x)\, dx. \tag{2.1}$$

The Monte Carlo integration estimates this integration by averaging the results evaluated at $N$ samples of the function $f$:

$$\langle F^N \rangle = \frac{1}{N} \sum_{i=1}^{N} \frac{f(X_i)}{\text{pdf}(X_i)} \tag{2.2}$$

where $X_i \in \Omega$ are random samples generated according to the probability density function $\text{pdf}(X_i)$.

We can easily prove that the expectation of $\langle F^N \rangle$ is equal to $F$, which means $\langle F^N \rangle$ is an unbiased estimator of $F$. This is a very important property: by increasing the number of samples $N$, $\langle F^N \rangle$ becomes closer and closer to $F$. This ensures that with a sufficiently large number of samples we can always get an estimation with the desired accuracy.

The major advantage of Monte Carlo integration is that it converges at same rate $O(\sqrt{N})$ in any number of dimensions. But in many cases this performance can be greatly improved by a variety of techniques.

### 2.1.1 Variance Reduction

As a non-deterministic approach, the variance of Monte Carlo integration with a fixed number of samples relies on the choice of PDF. A PDF $\text{pdf}(X_i) = cf(X_i)$ proportional to the function $f(x)$ being integrated, the variance of the estimator reduces to zero in this perfect situation. However, the computation of the normalization constant $c$ requires solving the integral that we want to estimate. In practice we usually choose a PDF close to $f(x)$ in order to reduce the variance. Unfortunately, function $f(x)$ can be very complex in practice, or be a combination of several functions. In this case, we can choose a PDF proportional to a portion of $f(x)$. For example, for the integrand is $L(x,\omega_i)\, f_r(x,\omega_i,\omega_o)\, (n\cdot\omega_i)$ in Equation 1.1, a cosine-weighted hemispherical sampling can improve the convergence rate of path tracing. Veach and Guibas [1995] introduced multiple importance sampling to generate samples according to a complex function. We can thus generate samples according to multiple PDFs and combine their contributions using multiple importance sampling.

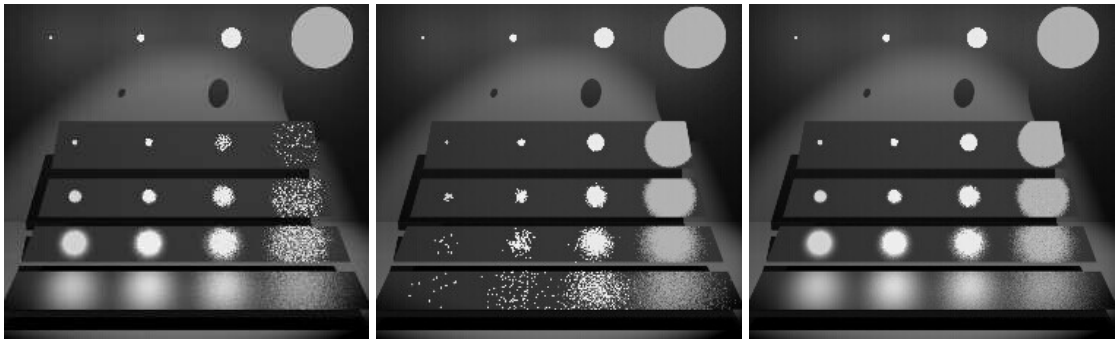

Figure 2.1: A comparison of three images rendered with Monte Carlo ray tracer from Veach and Guibas [1995]. From left to right: sampling the light source, sampling the BRDF, combining the samples of the light source and of the BRDF.

## 2.2 Monte Carlo Ray Tracing-based Approach

It is nearly impossible to analytically solve the rendering equation due to its complexity. Monte Carlo ray tracing provides an unbiased approximation of its solution

with Monte Carlo integration. It has been proven that algorithms based on Monte Carlo ray tracing are capable of solving the rendering equation accurately with a large enough number of samples.

### 2.2.1 Path Tracing

One of the most intuitive Monte Carlo ray tracing algorithms is "path tracing", proposed by Kajiya [1986]. It randomly generates a large number of rays from the camera through the pixel, and recursively traces the path of each ray until it reaches a light source. Then the contribution of this ray is evaluated and added to the pixel. The probability of a ray hitting a light source depends on the number, size, and position of light sources, as well as the camera parameters and the distribution of objects in the 3D scene. A scene with small lights may need an extremely large number of rays to reach a converged result.

Lafortune and Willems [1993] introduced bidirectional path tracing to adapt path tracing to scenes containing caustic effects. It traces ray paths both from the camera and the light source, which makes it easier to find paths from camera to light source to render caustics much more efficiently.

Inspired by genetic algorithms, Veach and Guibas [1997] proposed the Metropolis light transport algorithm. New light paths are generated by mutating existing paths. This algorithm is particularly good at handling localized illumination effects but relatively weak at caustics scenes.

### 2.2.2 Photon Mapping

Photon mapping, introduced by Jensen [1996] is a two-pass global illumination algorithm that uses the concept of photon maps. In first pass, it traces photons emitted from light sources and stores them on the intersected diffuse surfaces to build a photon map. In second pass, it traces camera rays and uses the photon map built in the previous pass to estimate the radiance by interpolating from the nearest photons. Their results

Figure 2.2: A comparison by Georgiev et al. [2012] between bidirectional path tracing (left) and stochastic progressive photon mapping (right) after 30 minutes of rendering. Stochastic progressive photon mapping produces more noisy results and cannot handle the illumination seen in the mirror. Bidirectional path tracing fails to capture the caustics of the vase reflected in the mirror.

show significant improvement compared to previous approaches especially in scenes with caustics.

This approach is further improved by Hachisuka et al. [2008]. They propose a multi-pass rendering algorithm based on photon mapping, which solves the memory usage problem present in the original photon mapping technique. This new approach is referred as progressive photon mapping. Knaus et al. [2011] present a new formulation of progressive photon mapping that converts it in an unbiased approach and further improves memory usage by avoiding to store any local photon statistics.

## 2.3 Irradiance and Radiance Caching

All the previously mentioned approaches focus on how to efficiently compute the global illumination at a given shade point. We can also accelerate rendering by exploring the space correlation between the shade points on surfaces.

Global illumination at shade point $x$ on a surface can be divided into two parts: direct illumination and indirect illumination. The direct illumination at shade point $x$ refers to the radiance contribution directly from light source. The indirect illumination refers to

the radiance contributed from light bounced off other surfaces in the scene:

$$L(x,\omega_o) = L_e(x,\omega_o) + L_{direct}(x,\omega_o) + L_{indirect}(x,\omega_o). \tag{2.3}$$

The direct component of global illumination is relatively easy to compute since we can explicitly sample light sources from the shade points. The indirect component, on the other hand, remains difficult to compute because of the integration over the hemisphere, but its contribution to the final image is relatively low. Therefore, computing indirect illumination precisely for each sample ray is not very cost-effective.

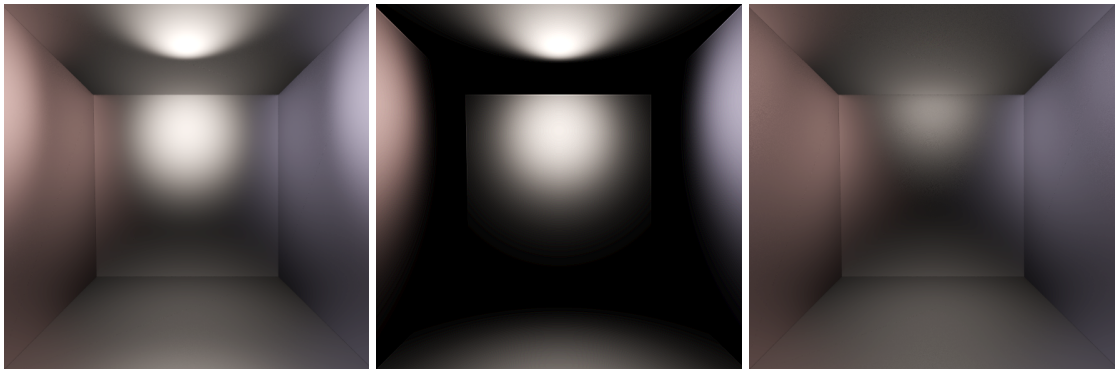

Figure 2.3: From left to right: final image with global illumination, contribution from direct illumination, and contribution from indirect illumination.

### 2.3.1 Irradiance Caching

Diffuse reflection is one of the simplest and common reflection models we use in computer graphics. A diffuse surface scatters light according to Lambert's cosine law, so sometimes it is also referred to as a Lambertian surface. Mathematically, the BRDF $f_r$ of a diffuse surface equal to a constant called albedo divided by $\pi$.

Solving the rendering equation on diffuse surfaces has the advantage of taking the BRDF term out of the integral. On diffuse surfaces we can define the irradiance $E$ by integrating over the hemisphere:

$$E(x,n) = \int_{\Omega} L_i(x,\omega_i)\,(n\cdot\omega_i)\,d\omega_i. \tag{2.4}$$

Then the rendering equation on diffuse surfaces can be rewritten using irradiance and surface albedo $\rho(x)$:

$$L_o(x,\omega_o) = L_e(x,\omega_o) + \frac{\rho(x)}{\pi} E(x,n). \tag{2.5}$$

Ward et al. [1988] observed that the irradiance of indirect illumination changes very slowly on diffuse surfaces due to the integration over the hemisphere. Based on this observation, they propose irradiance caching to accelerate the computation of indirect illumination on diffuse surfaces. The irradiance is only accurately computed by integration in sparse locations. The results are stored in records and reused to interpolate the irradiance when possible.

Irradiance caching greatly accelerates the computation of global illumination because the cost of interpolation is negligible compared to hemispheric integration. This property also causes image resolution to have very little impact on computing time. By only placing records on visible surfaces, the computing power is only invested on visible parts of a scene. Additionally, the cached irradiance is stored separately, which frees irradiance caching from the geometry. Another major advantage is that it can be incorporated easily with other global illumination algorithms; for example, Ward et al. [1988] used it to accelerate a Monte Carlo path tracer and Schwarzhaupt et al. [2012] used it in photon mapping.

Irradiance caching is weaker at handling scenes with complex geometry, due to oversampling. Tabellion and Lamorlette [2004] have addressed this issue by using simplified geometry with irradiance caching algorithms.

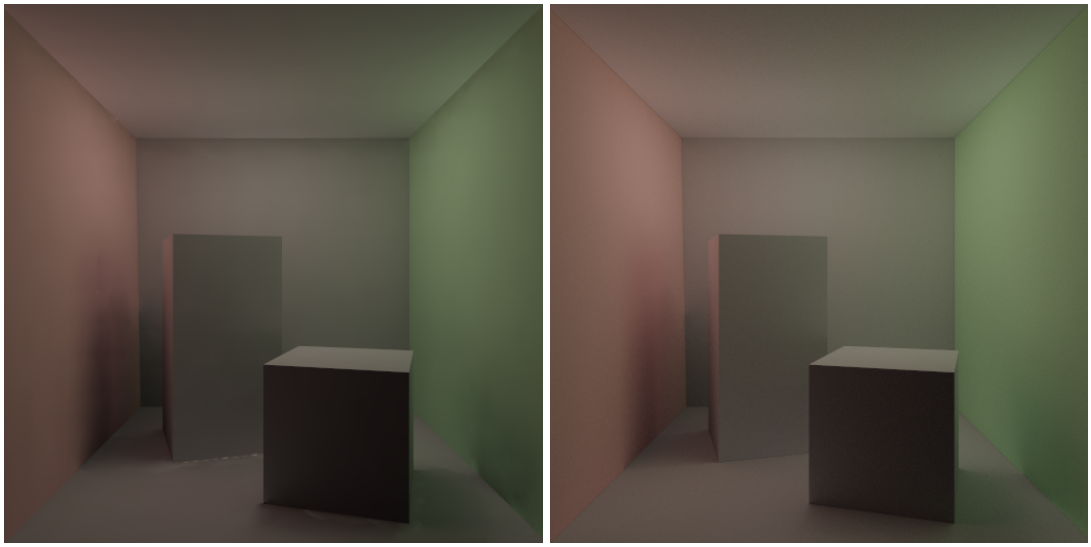

Figure 2.4: A diffuse Cornell Box scene. Left: Hessian-based irradiance caching strategy. Right: the reference image rendered by path tracing with 10240 samples per pixel.

### 2.3.2 Radiance Caching

After the success of irradiance caching, Krivanek [2005] extended this idea to handle the more complex cases of glossy surfaces, introducing an approach referred to as radiance caching. The main idea is similar, as the incident indirect radiance is only accurately computed and stored in sparse locations. For nearby locations, indirect illumination is computed using interpolation and extrapolation.

When extending irradiance caching to radiance caching, finding an efficient way to represent the distribution of incident radiance over the hemisphere is very important. The representation needs to be compact, yet the value at direction $\omega$ needs to be reconstructed easily. An efficient rotation algorithm is also necessary for radiance caching.

In Krivanek [2005]'s work, he chose spherical harmonics, and his results show that spherical harmonics is a very good fit for radiance caching. We decided to take the same approach, using spherical harmonics to store radiance distribution over the hemisphere. As our work heavily relies on spherical harmonics, we will introduce them briefly in the next section.

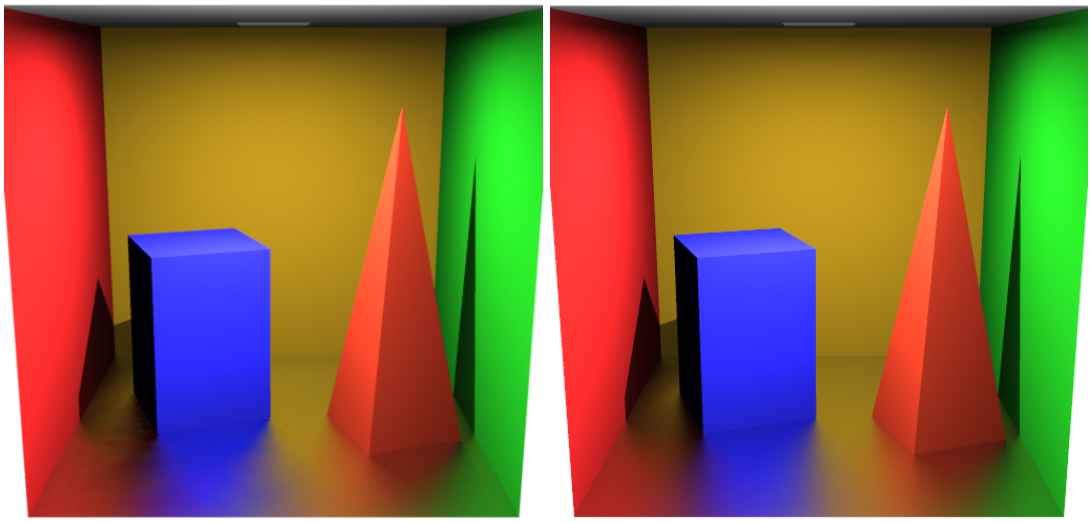

Figure 2.5: Images of Cornell Box scene from Krivanek [2005]'s dissertation. Notice the floor is a glossy surface. Left: radiance caching. Right: reference image rendered by path tracing.

### 2.3.3 Spherical Harmonics

Spherical harmonics are 1D Fourier series of spherical functions defined in spherical coordinates, widely used in the area of physics. They were first introduced by Laplace in 1782. In computer graphics, Sloan et al. [2002] have shown that spherical harmonics are an efficient way to approximate spherical functions.

First we define a series of functions, spherical harmonic basis functions: $y_0^0(\theta,\phi)$, $y_1^{-1}(\theta,\phi)$, ... , $y_l^m(\theta,\phi)$, ... , $y_l^l(\theta,\phi)$, where $l$ is a non-negative integer often referred to as the order of spherical harmonics and $m$ is an integer between $-l$ and $l$. With a set of spherical harmonic coefficients $\lambda_0^0$, $\lambda_1^{-1}$, $\lambda_1^0$, ... , $\lambda_l^m$, ... , $\lambda_l^l$, we can approximate the value of a spherical function $f(\theta,\phi)$ in a given direction.

$$f(\theta,\phi) = \sum_{l=0}^{n-1} \sum_{m=-l}^{M} \lambda_l^m y_l^m(\theta,\phi). \tag{2.6}$$

The coefficients can be computed by projecting the spherical function onto the basis functions:

$$\lambda_l^m = \int_\Omega f(\theta,\phi)\, y_l^m(\theta,\phi)\, d(\omega). \tag{2.7}$$

Spherical harmonic rotations are often required in computer graphics, and therefore, several algorithms have been proposed to rotate spherical harmonic coefficients. Kautz et al. [2002] efficiently perform the *y*-rotation by decomposing it into a *zxz* rotation. In the work of radiance caching, Krivanek [2005] accelerates spherical harmonic rotation by approximating the *y*-rotation with a first-order Taylor expansion. Later Nowrouzezahrai et al. [2012] efficiently decompose each order-$l$ spherical harmonic basis function into a sum of $2l+1$ rotated order-$l$ zonal harmonic lobes.

### 2.3.4 Record Placement

Record placement is crucial to the performance of irradiance caching and radiance caching, it refers to how we place the records in irradiance caching and radiance caching. Many studies have optimized the record placement; for example, in recent work by Jarosz et al. [2012] and Schwarzhaupt et al. [2012], a more accurate visibility-aware Hessian-based model was proposed to greatly improve the placement of records in the scene, significantly improving the quality and performance of the previous approaches.

The goal of this master thesis is to accelerate global illumination computations on glossy surfaces by extending the visibility-aware Hessian-based model introduced in these aforementioned work, in the context of radiance caching.

# CHAPTER 3

# HESSIAN-BASED OCCLUSION-AWARE RADIANCE CACHING

## 3.1 Abstract

Radiance caching is an efficient approach to accelerate the computation of global illumination on glossy surfaces, but it suffers from the absence of an efficient and high-quality record placement strategy. Existing approaches are far from optimal and rely heavily on clamping to avoid excessive records around scene edges. In a recent work on irradiance caching Schwarzhaupt et al. [2012] propose a Hessian-based occlusion-aware approach resulting artifact free image within a reasonable render time. In this paper, we investigate how to extend their error metric to radiance caching and study several record placement strategies.

## 3.2 Introduction and Previous Work

Efficiently simulating global illumination is one of the most important open problems in computer graphics. Accurately computing the effects of indirect illumination, caused by secondary bounces of light off surfaces in a 3D scene, is generally an expensive process that is often solved using algorithms such as path tracing [1986] or photon mapping [1996]. These approaches numerically solve the rendering equation [1986] using computationally intensive stochastic Monte Carlo approaches.

Ward et al. [1988] propose irradiance caching to accelerate these techniques when computing indirect illumination on diffuse surfaces. Krivanek [2005] extends the approach of Ward and Heckbert [1992] to handle the more complex case of glossy surfaces, introducing an approach referred to as radiance caching. In the work of Jarosz et al. [2012] and Schwarzhaupt et al. [2012], a more accurate visibility-aware Hessian-based model is proposed to greatly improve the placement of records in the scene for

use in an irradiance caching context, thus significantly improving the quality and performance of the baseline approach.

## 3.3 Background

As presented in Section 2.3, global illumination can be divided into two parts: direct illumination and indirect illumination. The direct part can be more easily computed, and therefore in this chapiter we only focus on the indirect component of global illumination.

### 3.3.1 Irradiance Caching

Irradiance caching exploits the fact that the indirect illumination component on a diffuse surface changes slowly across the surface due to hemispheric integration. The main idea of irradiance caching is that we compute and store the indirect irradiance accurately only in sparse locations in the scene. The indirect irradiance is only estimated at the other shade points in the scene by interpolating and extrapolating the information stored in nearby records.

Generally, the records are generated by a greedy algorithm. For a shade point $x$, we first find each record where $x$ is inside its valid region $A$. A valid region of a record can be defined by the maximum distance from the record, the maximum allowed normal change, the relative position, and other criteria. Usually the valid region corresponds to a circle or an ellipse. If one or more records are found, the indirect irradiance $E_x$ at $x$ is estimated using these records:

$$E_x \approx \frac{\sum_{i \in A} w_i(x)\, E_i'}{\sum w_i} \tag{3.1}$$

where $E_i'$ is the irradiance estimate given by record $i$, and $w_i$ is the weight of record $i$ to location $x$.

If $x$ is outside all records valid regions, indirect irradiance $E_x$ will be accurately computed using path tracing or another global illumination algorithm, and a new record

will be created at the current location $x$.

### 3.3.2 Radiance Caching

Given the success of irradiance caching, Krivanek [2005] extends it to handle low-frequency glossy surfaces, introducing an approach called radiance caching. On glossy surfaces, the reflected radiance $L_r$ at shade point $x$ toward direction $\omega_o$ can be computed by integrating over the hemisphere:

$$L_r(x,\omega_o) = \int_\Omega L_i(x,\omega_i)\, f_r(\omega_o,\omega_i)\,(n\cdot\omega_i)\, d\omega_i \tag{3.2}$$

where $L_i(x,\omega_i)$ is the incident radiance from direction $\omega_i$ and $f_r$ is the BRDF.

This approach uses the similar idea from irradiance caching: try to interpolate or extrapolate the indirect component of incident radiance from the records when possible. Figure 3.1 shows the performance of radiance caching with a minimum hit distance strategy in a Cornell Box composed of five glossy surfaces. To handle glossy surfaces, Krivanek uses spherical harmonics as an incident radiance representation, which allow fast rotation and provide a smooth approximation of incident radiance. Let $\lambda_0^0$, $\lambda_1^{-1}$, $\lambda_l^0$, ... , $\lambda_l^m$, ... , $\lambda_{l_{\max}}^{l_{\max}}$ be the spherical harmonic coefficients of incident radiance. Using Monte Carlo integration, these coefficients can be computed with Equation 2.7 as

$$\lambda_l^m = \frac{1}{N}\sum_{i=1}^{N}\frac{L_i(\omega_i)\, y_l^m(\omega_i)}{\mathrm{pdf}(\omega_i)} \tag{3.3}$$

where $y_l^m(\omega_i)$ is the value of the spherical harmonic basis function in direction $\omega_i$, $N$ is number of samples used, and $\mathrm{pdf}(\omega_i)$ is the PDF used to generate random directions $\omega_i$.

The computation of the spherical harmonic coefficients is only needed in sparse locations. For the rest of the shade points the coefficients are rotated, then interpolated and extrapolated. Once we have expressed the incident radiance in spherical harmonics, the incident radiance from direction $\omega_i$ can be easily reconstructed with Equation 2.6. Then

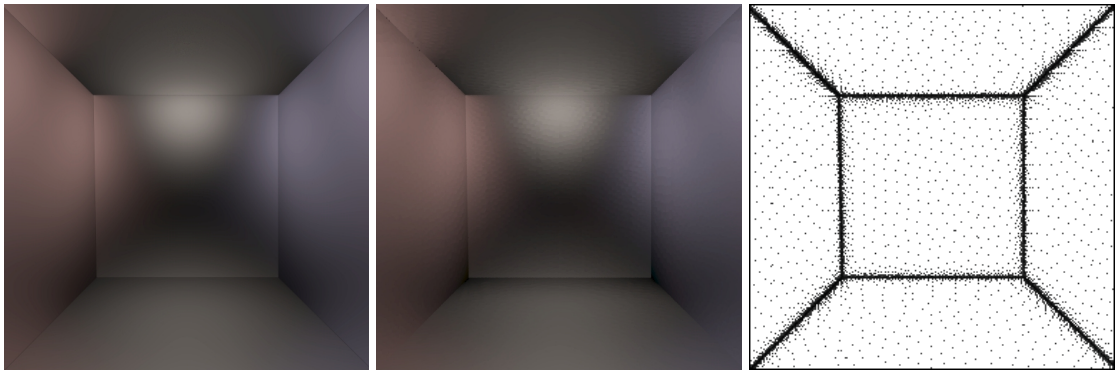

Figure 3.1: Left: the ground truth of indirect illumination generated by path tracing with 10240 samples per pixel. Middle: final image generated by a minimum hit distance strategy radiance caching without extrapolation. Right: the cache scheme generated using the same strategy illustrating the distribution of the records. It uses about 12K records and 8192 hemisphere samples per record. We can observe that with high amount of records (e.g. 12K), this basic strategy can produce relatively high quality results, but the artifacts are still visible.

the outgoing radiance can be computed using Equation 3.2. To simplify computations, Krivanek rewrites the integral in Equation 3.2 as the dot product of coefficient vectors if the BRDFs are projected onto the spherical harmonic basis functions (represented with coefficient vector $\Phi$). Let $\Lambda_i(x)$ denote the incident radiance spherical harmonic coefficients in vector form. Then Equation 3.2 becomes the dot product of two coefficient vectors (the cosine term is included in the BRDF):

$$L_o(x, \omega_o) = \Lambda_i(x) \cdot \Phi(x, \omega_o). \tag{3.4}$$

### 3.4 Record Placement Strategies in Irradiance Caching

In irradiance caching, most of the computing time is spent on creating records. We usually want to use fewer records while maintaining image quality. This can be achieved by optimizing record placement and more accurately estimating the irradiance at shade points. In this section we will focus on the previous work that improved record placement.

### 3.4.1 Heuristic-based Approaches

Intuitively, we want to place more records at locations where irradiance varies rapidly. It usually occurs at geometrically complex locations, such as around object edges. Thus, the size of valid regions of records needs to be reduced at these locations.

Distances between a shade point and visible surfaces can provide a rough estimate of the geometry's complexity. Based on this observation, several heuristics have been tested: the minimum hit distance during hemisphere sampling can be used to derive the maximum radius of a record. We can also use the harmonic mean distance from visible surface to a cache location. This heuristic is derived from the split-sphere model proposed by Ward et al. [1988].

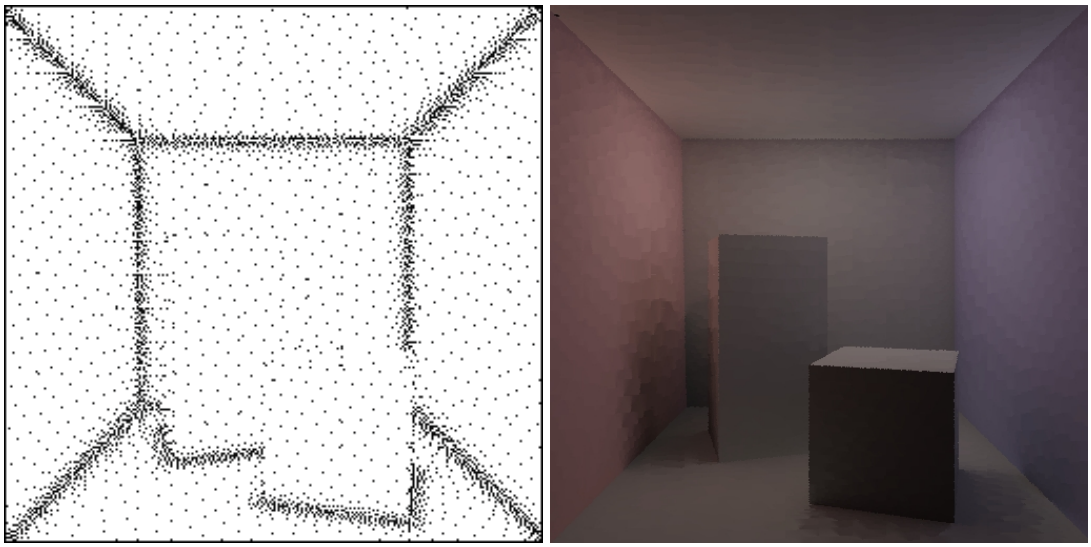

Figure 3.2: Left: the cache scheme generated by the minimum hit distance strategy in a diffuse Cornell Box scene. Right: the associated rendered image.

### 3.4.2 Error-Estimation-based Approaches

Heuristic-based approaches tend to generate an excessive number of records around edges and corners, and cannot detect incident lighting changes. We usually need to clamp the radii of records to adjust the record placement.

A more sophisticated approach controls record placement by limiting the support area $A$ by the upper bound total error $\varepsilon^t$ of a record. It is defined by the integral of the difference between the correct irradiance and the extrapolated irradiance (The interpolated radiance is introduced in Section 3.3.1.):

$$\varepsilon^t = \iint_A \mid E(x_i + \Delta x) - E'(x_i + \Delta x) \mid d\Delta x \tag{3.5}$$

where $E(x_i + \Delta x)$ is the ground-truth irradiance at $x_i + \Delta x$ and $E'(x_i + \Delta x)$ is the irradiance extrapolated using records.

### First-Order Error

Naturally, the ground-truth irradiance can be approximated by the first-order Taylor expansion:

$$\varepsilon^t < \iint_A \mid \nabla_x E(x_i) \Delta x \mid d\Delta x. \tag{3.6}$$

After further simplification, we get $\varepsilon^t < \pi R^2 \mid \nabla_x E(x_i) \mid$. To minimize the number of records, we need to maximize the surface area of $A$, while respecting the upper bound of error $\varepsilon_{max}$ within the area. Then the optimal isotropic radius can be derived with a predefined maximum error $\varepsilon_{max}$:

$$R = \sqrt{\frac{\varepsilon_{max}}{\pi \mid \nabla_x E(x_i) \mid}}. \tag{3.7}$$

### Second-Order Error

Jarosz et al. [2012] propose to compute the second-order derivatives of indirect illumination on diffuse surfaces which leads to a new irradiance error metric. Their new error metric can dramatically improve record placement.

The key to improving record placement is to accurately estimate the variation of indirect illumination. In this Hessian-based approach, they estimate ground-truth irradiance

using second-order Taylor expansion. The error is expressed using the $2 \times 2$ Hessian matrix of irradiance on the tangent plane, and is reformulated with $\lambda_1$ and $\lambda_2$, the two Eigenvalues of the Hessian matrix:

$$\begin{aligned} \varepsilon^t &< \frac{1}{2} \iint_A \mid \Delta x^T H_x(E_i) \Delta x \mid d\Delta x \\ &= \frac{1}{2} \iint_A \left( \mid \lambda_1 \mid x^2 + \mid \lambda_2 \mid y^2 \right) dxdy. \end{aligned} \tag{3.8}$$

To minimize the number of records, they maximize the surface area of $A$, while respecting the upper bound of error $\varepsilon^t$ within the area. By inverting Equation 3.8, they solve the two radii along each Eigenvector of the Hessian matrix:

$$\left( R_i^{\lambda_1}, R_i^{\lambda_2} \right) = \sqrt[4]{\frac{4\varepsilon^t}{\pi}} \left( \sqrt[4]{\frac{1}{\mid \lambda_1 \mid}}, \sqrt[4]{\frac{1}{\mid \lambda_2 \mid}} \right). \tag{3.9}$$

**Improved Error Estimation with Occlusion Information**

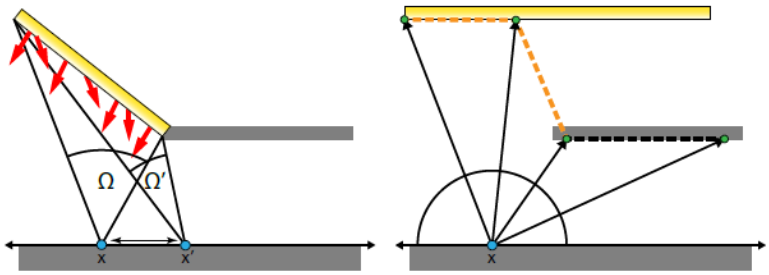


Figure 3.3: A simple 2D case analyzed by Schwarzhaupt et al. [2012] demonstrates how to convert a scene with occlusion to a scene without occlusion, that is radiometrically equivalent. We can observe that for every point $x$ on the bottom surface of both scenes, the distribution of irradiance is exactly the same.

With second-order error, the Hessian-based model increases accuracy and performance of irradiance caching. But the Hessian-based metric neglects visibility changes from each record. An investigation of 2D cases indicates that a occlusion-aware Hessian could further improve irradiance caching. This idea is further investigated by

Schwarzhaupt et al. [2012], who resolve this issue and improve the robustness of irradiance caching in complex scenes with occlusions.

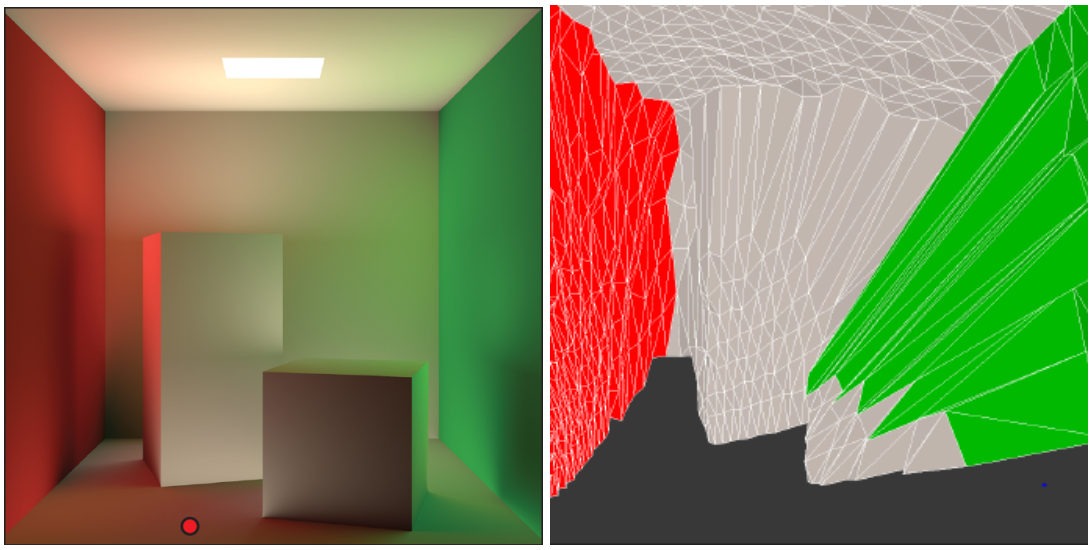

Figure 3.4: Two images from the work of Schwarzhaupt et al. [2012]. Left: the Cornell Box rendered with the occlusion-aware Hessian approach. Right: the converted scene (no occlusion) for a shade point $x$ on the floor (indicated by the red dot in the left image).

The key insight of Schwarzhaupt et al. [2012]'s approach is to convert the scene with occlusions into a radiometrically equivalent scene without occlusion for shade point $x$. Inspired by the 2D case illustrated in Figure 3.3, they construct a triangulated environment by connecting nearby stratified hemispherical samples. Assuming a triangle is small enough so that its exiting radiance is uniform, the irradiance can be computed by summing the contribution from each triangle:

$$E(x,n) \approx \sum_{j=1}^{M} L_{\triangle_j} F_{\triangle_j}(x) \tag{3.10}$$

where $M$ is the number of triangles, $L_{\triangle_j}$ is the exiting radiance, and $F_{\triangle_j}(x)$ is the solid angle covered by triangle $\triangle_j$ which is also referred to as a form factor. Here $L_{\triangle_j}$ is approximated by the radiance from the farthest point of the triangle to $x$.

The gradient and the Hessian irradiance can be easily computed by summing the

gradient and the Hessian of the triangle form factors:

$$\nabla_x E(x,n) \approx \sum_{j=1}^{M} L_{\triangle_j} \nabla_x F_{\triangle_j}(x), \tag{3.11}$$

$$H_x E(x,n) \approx \sum_{j=1}^{M} L_{\triangle_j} H_x F_{\triangle_j}(x) \tag{3.12}$$

where the gradient and the Hessian of the form factor have been derived by Holzschuch and Sillion [1998].

## 3.5 Occlusion-Aware Radiance Caching

As mentioned in Section 3.3.1, instead of accelerating the global illumination computation of single shade points, irradiance and radiance cachings explore the space correlation of shade points: adjacent locations have similar incident lighting.

One advantage of radiance caching is that it can be incorporated with other global illumination algorithms such as path tracing [1986] and photon mapping [1996]. It reduces computation time of these algorithms without losing quality of produced images.

For a long time, radiance caching suffered from many problems including the absence of an efficient record placement strategy. But recent work of Jarosz et al. [2012] and Schwarzhaupt et al. [2012] on irradiance caching indicated that the second-order error metric could be a viable way to produce high-quality cache scheme without heavily relying on clamping.

This idea drove us to investigate the possibility of generating optimized cache scheme using the Hessian-based error metric. We also carefully studied the robustness of radiance caching in arbitrary glossy scenes.

### 3.5.1 Occlusion-Aware Radiance Caching

We develop the new radiance caching algorithm by following the work of Krivanek [2005] and Schwarzhaupt et al. [2012], and we derive new mathematical models for visibility-aware radiance distribution estimation.

With the observation that the incident radiance varies relatively slowly on surfaces due to the hemisphere integral, we only perform sampling over the hemisphere to compute incident radiance in sparse record locations on surfaces. We place a record if it is not within the support area of any record. For the shade points inside the support area, the error of estimated radiance by interpolation and extrapolation is considered within the user-defined upper bound. An optimized record placement can improve image quality while using the same amount of records.

As proven in Krivanek's work [2005], spherical harmonics provide a smooth approximation and fast rotations for spherical functions. We decided to follow their approach: instead of caching scalar irradiance values, we cache spherical harmonic coefficients of the incoming radiance.

We obtain the spherical harmonic coefficients by projecting the incoming radiance $L_i$ from direction $\omega_i$ computed with path tracing onto the spherical harmonic basis function:

$$\lambda_l^m = \int_\Omega L_i(\omega_i)\, y_l^m(\omega_i)\, d(\omega_i). \tag{3.13}$$

We follow the work of Schwarzhaupt et al. [2012] and extend the occlusion-aware Hessian error metric mentioned in Section 3.4.2 to the radiance caching context. This allows us to obtain an optimal record placement. Using the mechanisms of caching and the second-order extrapolation, we are able to generate high-quality images while avoiding the costly computation of radiance at every pixel in the image.

In order to take occlusion changes into account when the shade point is translated on a surface, we construct a scene that is radiometrically equivalent, but without occlusion. For a shade point $x$ we first compute cosine weighted stratified sampling over the

hemisphere. Then we study the approach of Schwarzhaupt et al. [2012]: we go through all the samples and connect neighbors samples into triangles. In this context, the integral in Equation 3.13 is equal to the sum of the contributions from all the triangles thus constructed.

$$\lambda_l^m = \sum_{j=1}^{M} \int_{\omega_i \in \triangle} L_i(x, \omega_i)\,(n \cdot \omega_i)\, y_l^m(\omega_i)\, d\omega_i. \tag{3.14}$$

The integral in Equation 3.14 can be converted to a surface integral using a geometry Jacobian term:

$$\lambda_l^m = \sum_{j=1}^{M} \int_{y \in \triangle} L_i(r)\, y_l^m(r)\, G(x \leftrightarrow y)\, dy \tag{3.15}$$

where $r = y - x$, and $G(x \leftrightarrow y) = -(n \cdot r)(n_y \cdot r) / \mid r \mid^4$ is the geometry Jacobian term.

Considering that the triangles are small enough, we can assume that radiance $L_{\triangle_j}$ exiting from the triangle is uniform and independent of small direction changes. $L_{\triangle_j}$ is defined by the incident radiance from the vertex that is farthest from shade point $x$, because it defines the radiance emitted from the objects that occluded or disoccluded when moves the shade point. These valid assumptions were proven in previous work of Ward and Heckbert [1992], Schwarzhaupt et al. [2012] and Annen et al. [2004].

Then Equation 3.15 can be further simplified:

$$\lambda_l^m = \sum_{j=1}^{M} L_i(r) \int_{y \in \triangle} y_l^m(r)\, G(x \leftrightarrow y)\, dy. \tag{3.16}$$

The above equation is still too complex to be computed analytically. Given the same assumption that triangles are small, we can assume the term $y_l^m(r)$ is constant within the triangle, and we can move $y_l^m(r)$ out of the integral. Let form factor $F_{\triangle_j}(x)$ denote the integral term $\int_{y \in \triangle} -(n \cdot r)(n_y \cdot r) / \mid r \mid^4 dy$. It has been analytically computed in the work of Holzschuch and Sillion [1998]. Then we get:

$$\lambda_l^m = \sum_{j=1}^{M} L_{\triangle_j} F_{\triangle_j}(x)\, y_l^m(\omega_i). \tag{3.17}$$

Several of our assumptions depend on the fact that the triangles are small. In practice, in order to make the results accurate, we need to ensure that the number of hemisphere samples is large enough (in practice we usually use more than 2K hemisphere samples).

### 3.5.2 Second-Order Interpolation and Extrapolation

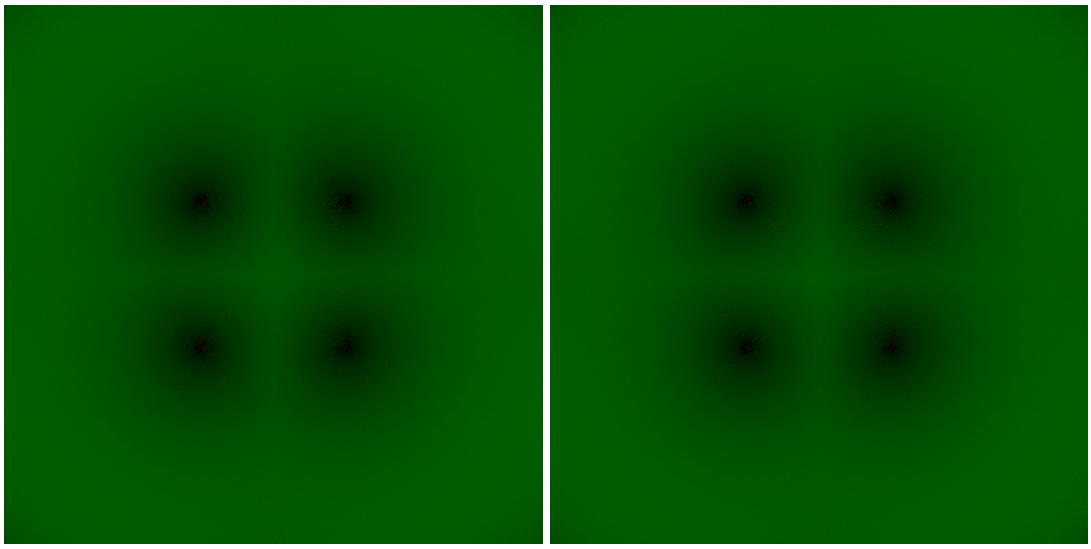

Figure 3.5: A simple scene with a sphere and a plane. Using four fixed records, we are able to show the error improvement using extrapolation. First, the error of linear extrapolation is compared to the error of the first-order extrapolation. The result is shown on the left side: green indicates that the first-order extrapolation has less errors, red indicates that linear extrapolation produces less errors. Right: error comparison between the first-order extrapolation and the second-order extrapolation; green indicates that the second-order extrapolation has less errors, red indicates that the first-order extrapolation has less errors. Image luminance of is gamma corrected to better illustrate the errors.

Given a shade point $x$, the spherical harmonic coefficients can be computed by interpolating the cached records nearby. To improve the quality of the estimation of radiance caching, Krivanek [2005] uses the first-order Taylor expansion to extrapolate the records. In our work, we used the second-order Taylor expansion to more accurately estimate the

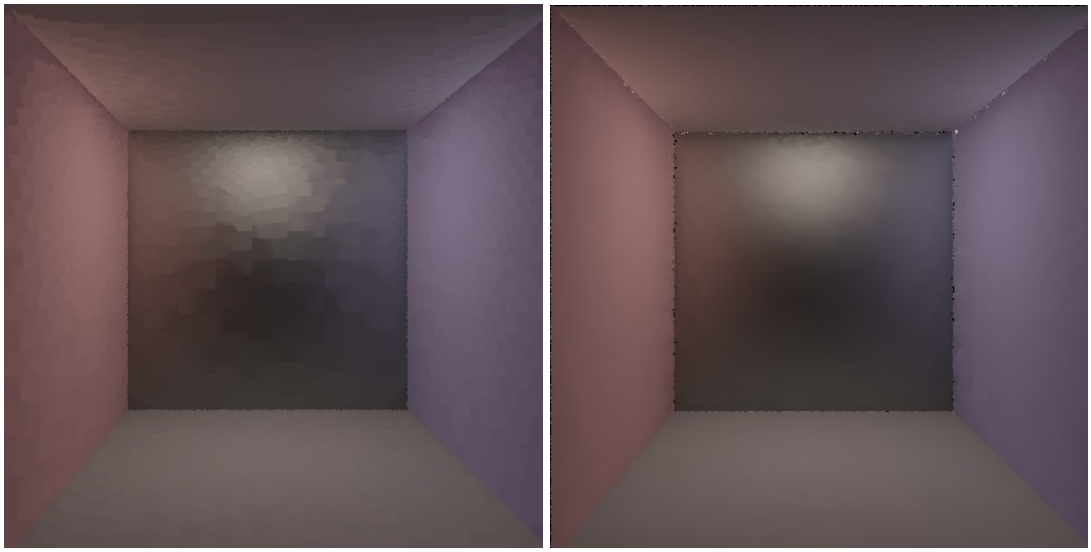

Figure 3.6: A simplified Cornell Box scene (no occlusion), the comparison between before (left) and after (right) extrapolation with same record placement (4K records).

coefficients at point $x$:

$$\lambda_i^n \approx \frac{\sum_{i \in S} w_i(x) \left(\lambda_i^n + \nabla_x \lambda_i^n \Delta x_i + \frac{1}{2} \Delta x_i^T H_x(\lambda_i^n) \Delta x_i\right)}{\sum w_i} \tag{3.18}$$

where $S$ is the set of all records and $x$ is within the valid region. The incident radiance is stored in a global coordinate system, $\lambda_i^n$ is the $n$-th spherical harmonic coefficient of the $i$-th record near point $x$. $w_i$ is the weight of the record that depends on the distance and normal of point $x$.

In practice we convert $\Delta x_i$ to a local coordinate system because the Hessian matrix that we computed in previous steps is in a local coordinate system at each record $x_i$.

### 3.5.3 Record Valid Region and Error-Metric

In radiance caching, computation time mainly involves the generation of records. Time spent on each record is constant, so reducing the number of records makes our approach more efficient. Ideally, we should enlarge the area of the valid region for each record so it can cover more surface area. We still need to avoid visible artifacts caused

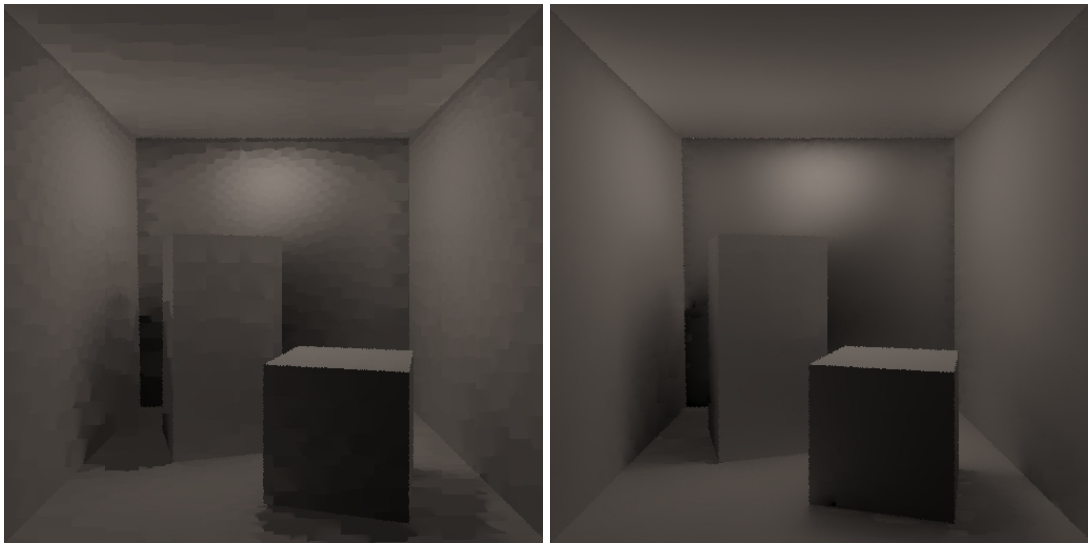

Figure 3.7: A Cornell Box scene. Comparison between before (left) and after (right) extrapolation with same record placement (4K records). Notice that our approach can automatically handle occlusion changes without producing visible artifacts.

by extrapolation errors. To balance between computation time and extrapolation errors, we set an upper bound on error, and find the maximum area of a valid region while respecting this maximum error.

At location $x+\Delta x$, the extrapolation error $\varepsilon(x+\Delta x)$ is the difference between the ground truth of outgoing radiance $L(x+\Delta x)$ and the outgoing radiance $L'(x+\Delta x)$ computed with extrapolated spherical harmonic coefficients. We define spherical harmonic coefficient error of incident radiance as $\varepsilon_l^m$:

$$\varepsilon_l^m(x+\Delta x) = \mid \lambda_l^m(x+\Delta x) - \lambda'^m_l(x+\Delta x) \mid \tag{3.19}$$

where $\lambda_l^m(x+\Delta x)$ is the ground truth of the spherical harmonic coefficient (which we avoid computing) and $\lambda'^m_l(x+\Delta x)$ is our estimation of the spherical harmonic coefficient at location $x$.

Recall that in the outgoing radiance computation in Equation 3.4, $\varepsilon(x+\Delta x)$ can be expressed by $\varepsilon_l^m$:

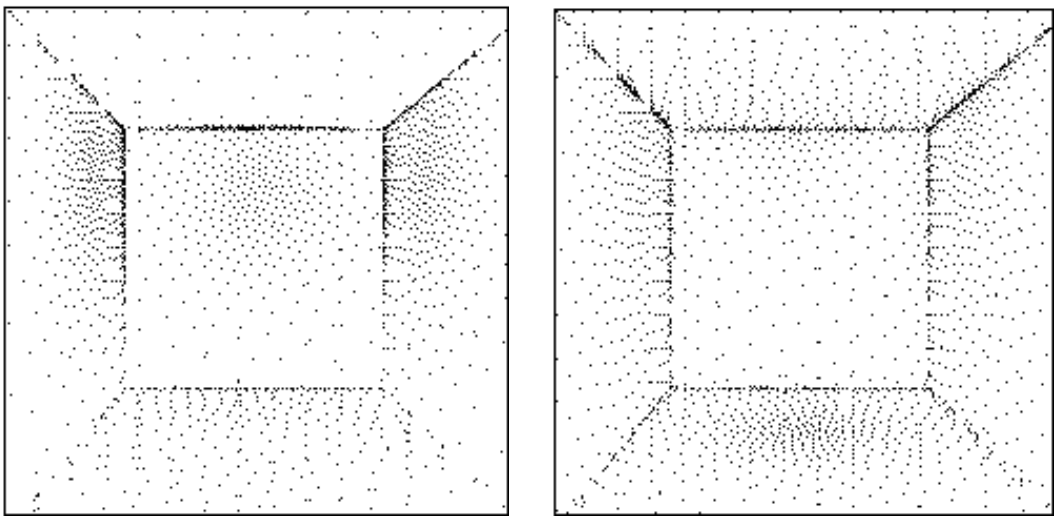

Figure 3.8: Comparison between cache schemes generated by the first-order (left) and the second-order (right) error metric, with each 2K records. The second-order error metric naturally tends to place records more uniformly across the scene and is still able to capture radiance changes in the scene.

$$\varepsilon(x+\Delta x) = \sum_{l=0}^{l_{\max}} \sum_{m=-l}^{l} \varepsilon_l^m \Phi_l^m. \tag{3.20}$$

$\Phi_l^m$ only depends on the surface type and the location. We can give an upper bound to $\varepsilon(x+\Delta x)$:

$$\varepsilon(x+\Delta x) < \mid \Phi \mid\mid \Lambda_i \mid \tag{3.21}$$

where $\mid \Lambda_i \mid = \sqrt{\sum_{l=0}^{l_{\max}} \sum_{m=-l}^{l} \left(\varepsilon_l^m(x+\Delta x)\right)^2}$ is the norm of the error of incident radiance coefficient vector. Here $\mid \Phi \mid$ is a precomputed constant.

The total error of a record is defined by the error integral within the valid region $A$:

$$\varepsilon = \iint_A \varepsilon(x+\Delta x)\, d \triangle x. \tag{3.22}$$

Then we can easily get:

$$\varepsilon < \mid \Phi_{\max} \mid \iint_A \mid \Lambda_i \mid \, d \triangle x. \tag{3.23}$$

By grouping $\mid \Phi_{\max} \mid$ and $\varepsilon$, we get an upper bound $\varepsilon^{\lambda} = \varepsilon / \mid \Phi_{\max} \mid$ for the norm of the error of incident radiance coefficient vector. In practice, we use the user-defined parameter $\varepsilon_{max}$ to set the maximum value of $\varepsilon^{\lambda}$. To compute $\varepsilon_l^m$, we need the ground truth from spherical harmonic coefficients. To avoid computing it directly, we use a Taylor expansion to approximate it.

**First-Order Approximation**

The first step needs to approximate the ground truth of the spherical harmonic coefficients $\lambda_l^m(x+\Delta x)$ by a first-order Taylor expansion. We use a first-order extrapolation to estimate coefficient $\lambda'^m_l(x+\Delta x)$. Then we have:

$$\varepsilon_l^m(x+\Delta x) \approx \mid \nabla_x \lambda_l^m \cdot \Delta x \mid \tag{3.24}$$

where $\nabla_x \lambda_l^m$ is the gradient of the spherical harmonic coefficient $\lambda_l^m$ at $x$. The upper bound of total error $\varepsilon$ can be easily computed as:

$$\begin{aligned} \varepsilon^{\lambda} &\approx \iint_A \sqrt{\textstyle\sum_{l=0}^{l=l_{max}} \sum_{m=-l}^{m=l} \mid \nabla_x \lambda_l^m \cdot \Delta x \mid^2} \, d\Delta x \\ &\leq \pi R^2 \mid \nabla_x \lambda \mid \end{aligned} \tag{3.25}$$

where $R$ is the radius of isotropic record validity region and $\mid \nabla_x \lambda \mid$ is the norm of the gradient of coefficient vector. This leads to the following isotropic record radius equation:

$$R = \sqrt{\frac{\varepsilon_{max}}{\pi \mid \nabla_x \lambda \mid}}. \tag{3.26}$$

In practice, we compute $R$ separately for the three RGB channels, and then choose the minimum value.

**Second-Order Approximation**

To further optimize the record placement, we use the second-order Taylor expansion to approximate the ground truth and estimate the coefficients using the first-order expansion. Here the coefficient error at location $x+\Delta x$ becomes:

$$\varepsilon_l^m \approx \frac{1}{2} \mid \Delta x^T H_x(\lambda_l^m) \Delta x \mid \tag{3.27}$$

where $H_x(\lambda_l^m)$ is the Hessian matrix of spherical harmonic coefficient $\lambda_l^m$ at $x$.

Applying the similar technique explained in Section 3.4.2, the upper bound of total error $\varepsilon$ can be expressed as follows:

$$\varepsilon^\lambda \leq \frac{1}{2} \iint_A \left( \mid u_1 \mid x^2 + \mid u_2 \mid y^2 \right) dxdy \tag{3.28}$$

where $u_1$ and $u_2$ are two Eigenvalues of a $2 \times 2$ matrix $H_x^{sum}$, where $H_x^{sum} = \sum_{l=0}^{l=l_{max}} \sum_{m=-l}^{m=l} \mid H_x(\lambda_l^m) \mid$. Therefore the anisotropic record radius $(R_1, R_2)$ can be obtained as:

$$\left(R_i^{u_1}, R_i^{u_2}\right) = \sqrt[4]{\frac{4\,\varepsilon_{max}}{\pi}} \left( \sqrt[4]{\frac{1}{\mid u_1 \mid}}, \sqrt[4]{\frac{1}{\mid u_2 \mid}} \right) \tag{3.29}$$

### 3.5.4 Gradient and Hessian of Spherical Harmonic Coefficients

To correctly place the records and extrapolate the radiance, we compute the gradient and the Hessian of coefficients. This can be done by computing the derivative and the second-order derivative of Equation 3.17:

$$\nabla_x \lambda_l^m = \sum_{j=1}^{M} L_{\triangle_j} \nabla_x (F_{\triangle_j}(x)\, y_l^m(\omega_i)) \tag{3.30}$$

$$H_x \lambda_l^m = \sum_{j=1}^{M} L_{\triangle_j} H_x (F_{\triangle_j}(x)\, y_l^m(\omega_i)) \tag{3.31}$$

Here we use similar assumptions than in previous work on radiance caching: we

assume the emitted radiance does not vary with insignificant direction changes, which means that in scenes with specular or high-order glossy surfaces, this assumption will no longer be valid. Ignoring this term will potentially cause artifacts which we will discuss in detail in Section 3.8.

The two terms in Equation 3.31: $\nabla_x F_{\triangle_j}$ and $H_x(F_{\triangle_j}(x))$ are already derived from previous work [1998]. We only need to convert the spherical harmonic basis functions to a Cartesian formulation, and then to compute the translation gradient and the translation Hessian. The details and formulation are included in Appendix I. Another solution would be to use the chain rule to derive the gradient and the Hessian from angular formulation of basis functions. But it only works for gradient computation. The Hessian matrices turn out to be too complex to derive analytically.

### 3.6 Compute the Outgoing Radiance

As mentioned in Section 3.3.2, Krivanek [2005] computes the outgoing radiance by converting the integral to the dot product of two coefficient vectors using Equation 3.4. His approach requires projecting the BRDF onto spherical harmonic basis functions. He precomputes the coefficient vectors $\Phi$ for all outgoing directions by sampling the incident and outgoing directions, and stores them in a table.

For some BRDF types, such as Phong and Lambertian, there are analytical solutions for the spherical harmonic representation. The advantage of this approach is that it can be computed very quickly and it is perfectly accurate. Its drawback is that the analytical results of BRDF representations are unclamped, so they will generate errors in the outgoing radiance computations.

In our work, we use a more general approach given by Pharr and Humphreys [2010]. Let $\Lambda_o$ denote the outgoing radiance spherical harmonic coefficients in vector form, and $\Lambda_i$ denote the spherical harmonic coefficients vector of incident radiance. We define a $(l_{\max}+1)^2 \times (l_{\max}+1)^2$ matrix $M$ called BRDF matrix:

$$\Lambda_o = M\,\Lambda_i. \tag{3.32}$$

Pharr and Humphreys [2010] give the analytical expression for $M$:

$$\mathbf{M}_{j,k} = \int_{S^2}\int_{S^2} f(\omega_o, \omega_i)\, y_j(\omega_i)\, y_k(\omega_o)\, d\omega_i\, d\omega_o. \tag{3.33}$$

In practice, we use Monte Carlo integration to precompute $M$ for each BRDF type in the scene and store it in a cache (or in a file). With the coefficient vector of outgoing radiance, the outgoing radiance toward direction $\omega_o$ can be easily reconstructed as follows:

$$L_o(x, \omega_o) = \sum_{l=0}^{l_{\max}} \sum_{m=-l}^{l} y_l^m(\omega_o)\, \omega_o^{l,m}. \tag{3.34}$$

## 3.7 Outgoing Radiance Caching

Until now, the information we store in each record is its incident radiance and its translation gradient and the Hessian. Due to the fact that the record validity region is controlled by the total error of incident radiance coefficients, this current record placement strategy does not depend on the type of surface. Thus, it produces the same record placement on different surfaces. In practice, the outgoing radiance varies slower on diffuse and low-order glossy surfaces under the same incident lighting. Ideally, on high-order glossy surfaces the valid region of a record should be smaller than the same record on a diffuse or low-order glossy surface. This can be handled by storing outgoing radiance instead of incident radiance. Additionally, this can reduce the uses of the BRDF matrix $M$ to compute outgoing radiance.

### 3.7.1 View-Independent Record Placement

Naturally our first step is to adapt the error metric previously presented in Section 3.6 to outgoing radiance. According to previous sections, the spherical harmonic coefficient vector of outgoing radiance is $\Lambda_o = M\Lambda_i$. The BRDF matrix $M$ only depends on the surface type, as the gradient and the Hessian of outgoing radiance can be easily computed by the product of BRDF matrix and the gradient and Hessian of incoming radiance.

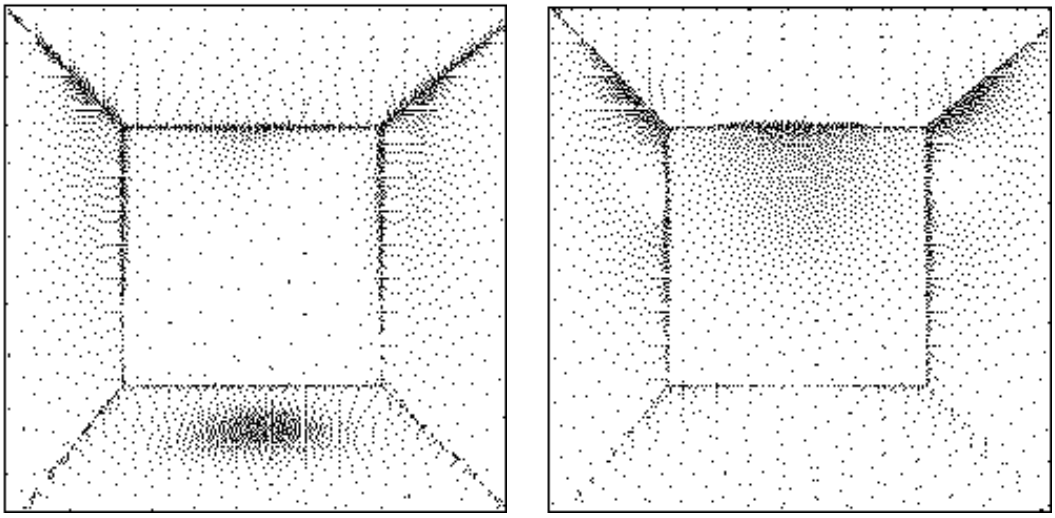

Figure 3.9: Left: cache scheme on the Cornell Box scene using incoming radiance caching. Right: cache scheme on same scene with outgoing radiance caching. Both approaches generated approximately the same numbers of records (4K and 3.6K). Comparing the two images, we can easily find out that outgoing radiance caching puts more records on the glossy surface (the one facing the camera).

### 3.7.2 View-Dependent Record Placement

On glossy surfaces, light bouncing towards the camera can vary significantly depending on the camera position. To further improve the record placement, we should take direction $\omega_o$ from record towards the camera into account instead of placing the records according to the outgoing radiance towards all directions.

This error metric computes the integral of the difference between the correct radi-

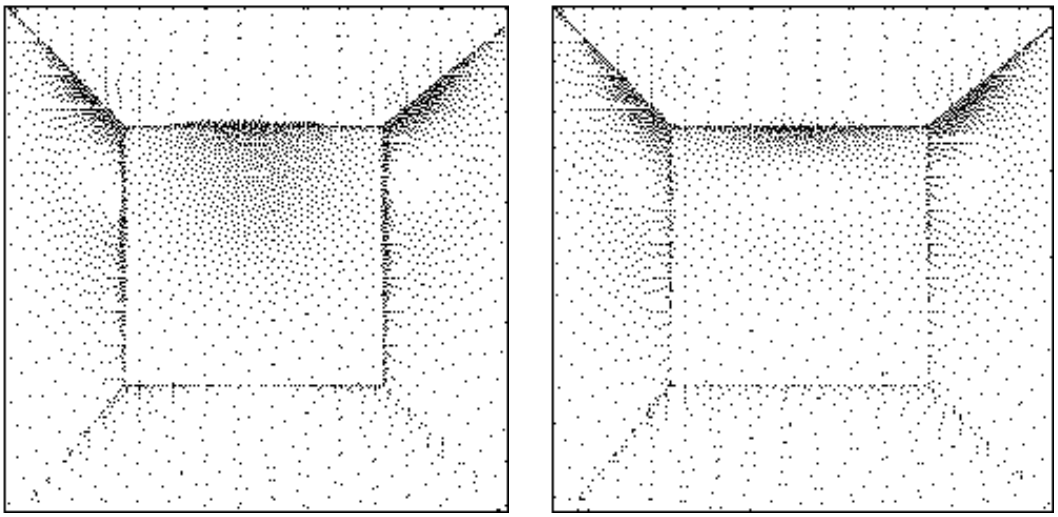

Figure 3.10: Both cache schemes are generated with outgoing radiance caching on different Cornell Box scenes. On right side the surface facing the camera is diffuse instead of glossy (left). We can clearly observe that in the right image, fewer records are placed on the surface facing the camera. Using the same parameters, the scene containing the glossy surface (left) generates 3.6K records and the scene without glossy surface (right) generates 2.9K records.

ance bounce towards the camera and the radiance computed by extrapolated spherical harmonic coefficients

$$\sigma = \iint_A \mid L(x+\Delta x, \omega_x) - L'(x+\Delta x, \omega_x) \mid dx. \tag{3.35}$$

where $L(x_i+\Delta x, \omega_x)$ and $L'(x_i+\Delta x, \omega_x)$ can be computed using the spherical harmonic coefficients.

Again, we used the Taylor expansions to estimate the total error inside the validity region, the first order approximation of $\sigma$ can be expressed as follows:

$$\begin{aligned} \sigma &\approx \iint_A \mid \sum_{l=0}^{l_{max}} \sum_{m=-l}^{l} (\nabla_x \lambda_l^m \cdot \Delta x) y_l^m \mid dx \\ &\leq \pi R^2 \sum_{l=0}^{l_{max}} \sum_{m=-l}^{l} \mid \nabla_x \lambda_l^m \mid \mid y_l^m \mid . \end{aligned} \tag{3.36}$$

The second order approximation of $\sigma$ is:

$$\begin{aligned} \sigma &\approx \iint_A \left| \sum_{l=0}^{l_{max}} \sum_{m=-l}^{l} \left( \tfrac{1}{2} \Delta x^T H_l^m \Delta x \right) y_l^m \right| dx \\ &= \iint_A \left| \left( \tfrac{1}{2} \Delta x^T H_{sum} \Delta x \right) \right| dx \end{aligned} \tag{3.37}$$

where $H_{sum} = \sum_{l=0}^{l_{max}} \sum_{m=-l}^{l} H_l^m y_l^m$, then we used the same technique in Section 3.5.3 to compute the anisotropic radius with Eigenvalues of $H_{sum}$.

## 3.8 When Assumptions are Broken

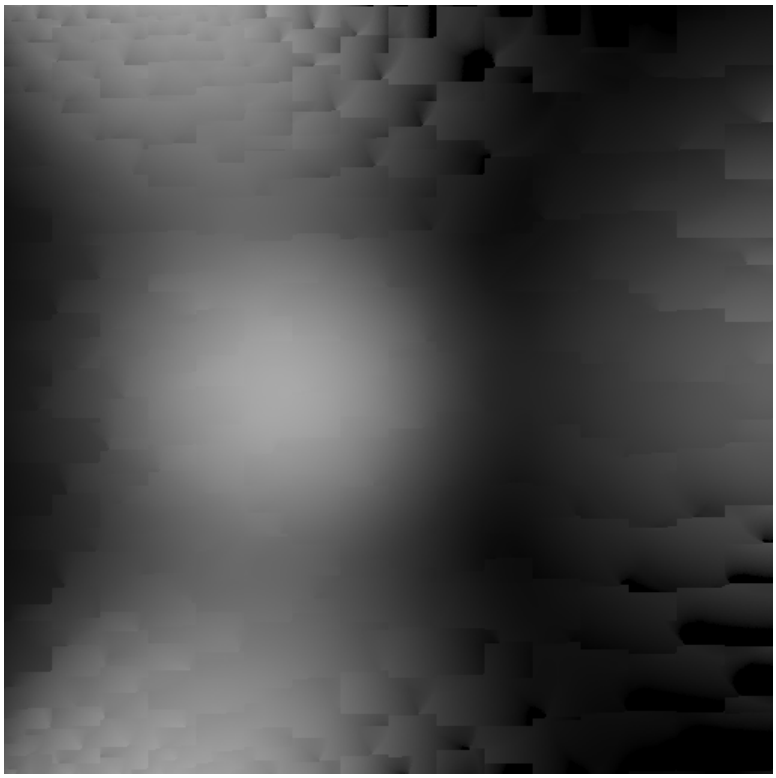

Figure 3.11: A Cornell Box scene with five glossy walls (only the wall facing the camera wall is visible) where current radiance caching algorithm fail.

In some test scenes with multiple glossy surfaces, we observed the artifacts shown in Figure 3.11. We find that extrapolation artifacts occurred in presence of two adjacent glossy surfaces. These artifacts may be caused by the following assumption we made

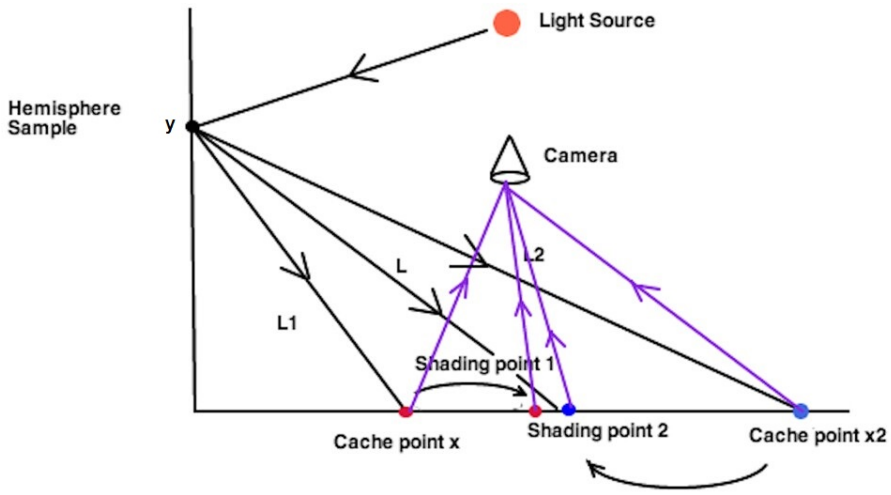


Figure 3.12: This illustration explains why this assumption will cause the artifact we observed above.

during the gradient and the Hessian computation in Section 3.5.4: Emitted radiance does not vary with insignificant direction changes. We assumed that direction changes while translating a shade point on the surface can be ignored.

This assumption is considered valid and used in all previous work about radiance caching. We noticed that none of testing scenes of previous work has more than one glossy surface. The validity of this assumption in more complex test scenes was never actually tested and verified.

### 3.8.1 Demonstration on a Test Case

To further investigate our suspicion about this assumption, we set up two similar test scenes consisting of only two surfaces and a point light source. The bottom surface $A$ captured by the camera is a glossy surface. Surface $B$ is perpendicular to surface $A$: it is not visible from the camera and it can be either diffuse or glossy. We render the scene alternately with glossy and diffuse surface $B$ using radiance caching. The results are shown in Figure 3.14. We can easily observe that the second image with diffuse

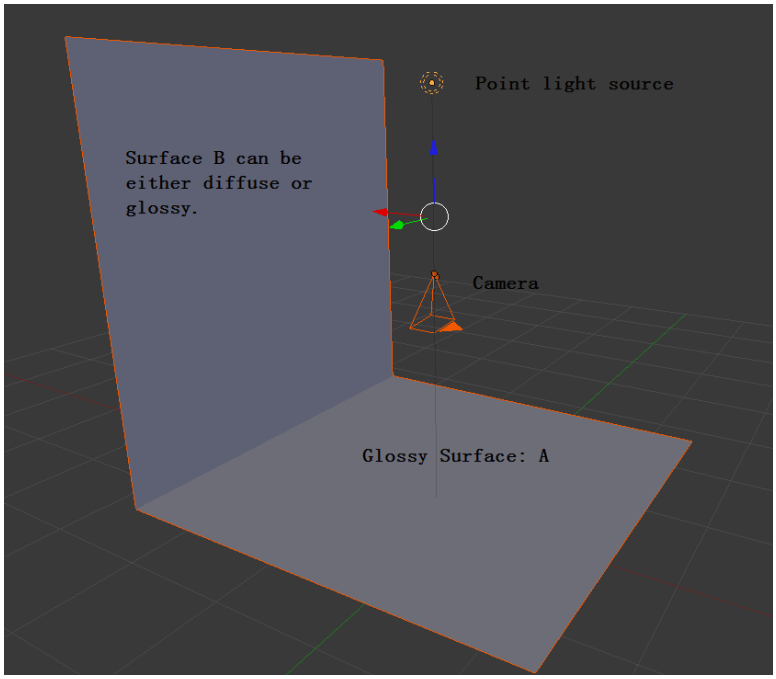


Figure 3.13: An illustration of the test scene.

surface $B$ produces an artifact free image. Artifacts similar to Figure 3.11 appear when we replace the diffuse surface $B$ by a glossy surface.

To show that the artifacts are not caused by other factors in the algorithm, we created a new radiance caching algorithm. For each hemispherical sample of a record, we store the radiance bounce toward record along with the location, normal, and BRDF of the hemispherical sample. At a new shade location, the indirect illumination can be written in the form of a hemisphere integration similar to the rendering equation. We can rewrite it into the form of surface integration with the geometry Jacobian term:

$$L_o(x \to e) = \int_S L_i(y \to x)\, f_r(e \to x \to y)\, G(x \leftrightarrow y)\, dy \tag{3.38}$$

where $G(x \leftrightarrow y)$ is the same geometry Jacobian term in Equation 3.15, and $e$ is the eye position.

Now that the locations of hemispherical samples are stored, we can use the above formula to numerically compute the indirect illumination in this location. If the previous

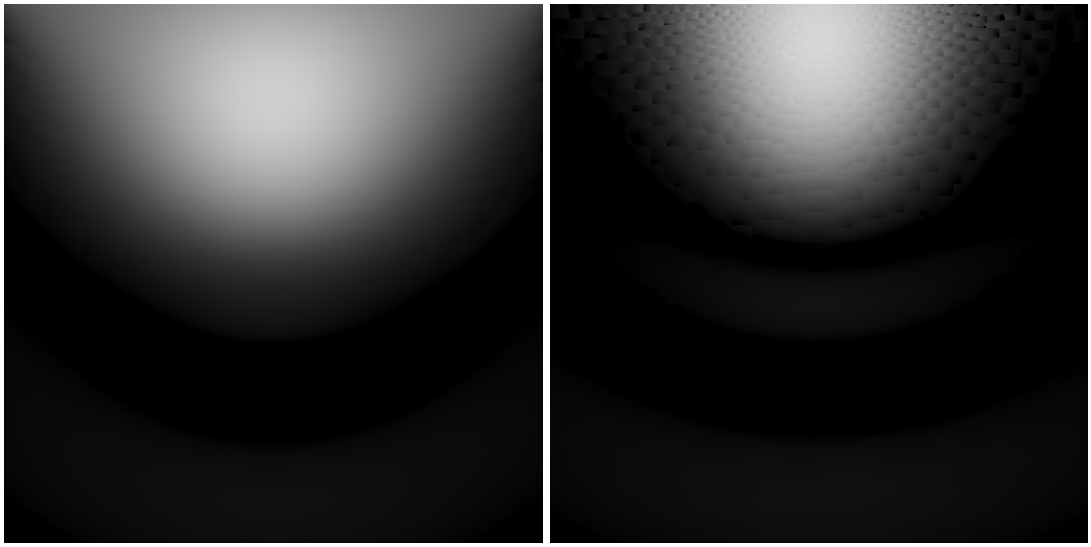

Figure 3.14: Both images are rendered using radiance caching. Left: test scene with diffuse surface $B$. Right: test scene with glossy surface $B$.

mentioned assumption is valid then the term $L_i$ in Equation 3.38 can be approximated by the radiance bounced from hemispherical sample toward the record location without visible artifact. Because the normal and BRDF are also stored, the relative accurate value of $L_i$ can also be computed with path tracing.

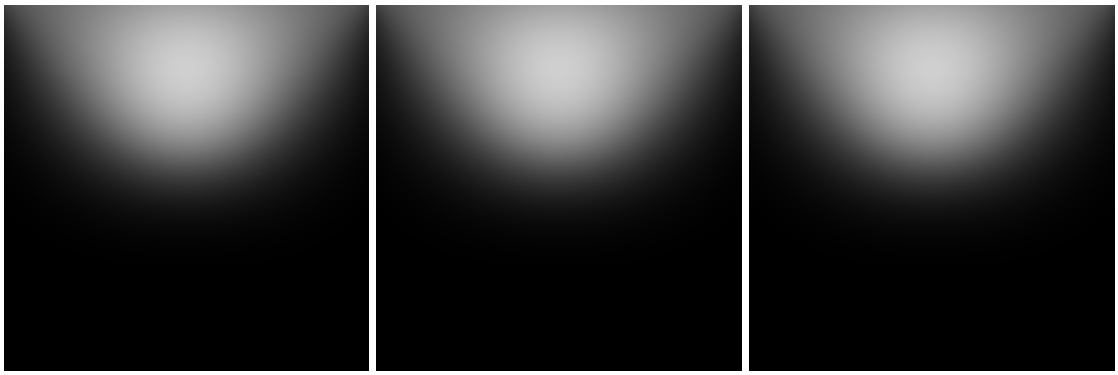

Figure 3.15: Left and middle: comparison of images rendered with and without the assumption when $B$ is diffuse. Right: the reference image rendered by path tracing. We can observe that the three images are identical, which supports that the assumption is valid when surface $B$ is diffuse.

We applied the new algorithm with the assumption on two test scenes. The results

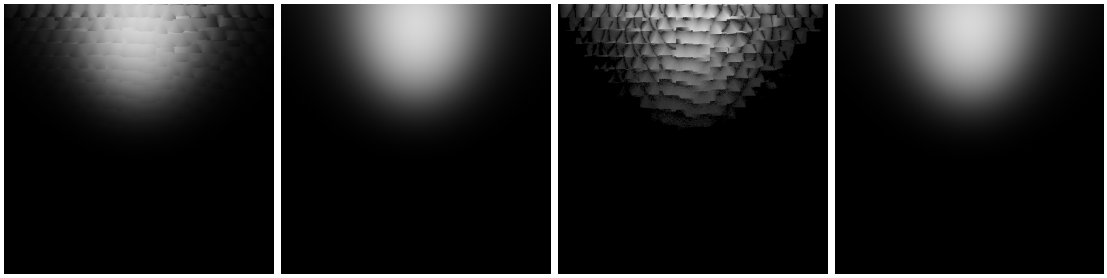

Figure 3.16: Left two images: comparison of images rendered with and without the assumption when $B$ is glossy. Third image: error generated by the assumption (with gamma correction). Fourth image: reference image rendered by path tracing. The artifacts of the first image indicate that the assumption fails when both surfaces in the scene are glossy.

shown in Figure 3.15 and Figure 3.16 indicate that the artifacts remain on the two glossy surface scene. Then we replaced the approximated $L_i$ by $L_i$ computed with path tracing; the images produced with both scenes are now artifact free. Another direct proof is Figure 3.18, we compare the contribution from the current gradient term and the missing term we will introduce in next section. Clearly, the contribution from the missing term cannot be ignored. We can therefore conclude that the assumption of invariant radiance during translation of a shade point is inappropriate on arbitrary scenes with glossy surfaces.

### 3.8.2 Possible Solutions

In order to make the current radiance caching approach more robust with arbitrary scenes, the derivation of the gradient and the Hessian in previous sections must be reformulated without the previous assumption. In this section we discuss two potential solutions.

Back to Equation 3.17, its derivative and second-order derivative are:

$$\nabla_x \lambda_l^m = \sum_{j=1}^{M} \nabla_x (L_{\triangle_j} F_{\triangle_j}(x) y_l^m(\omega_i)). \tag{3.39}$$

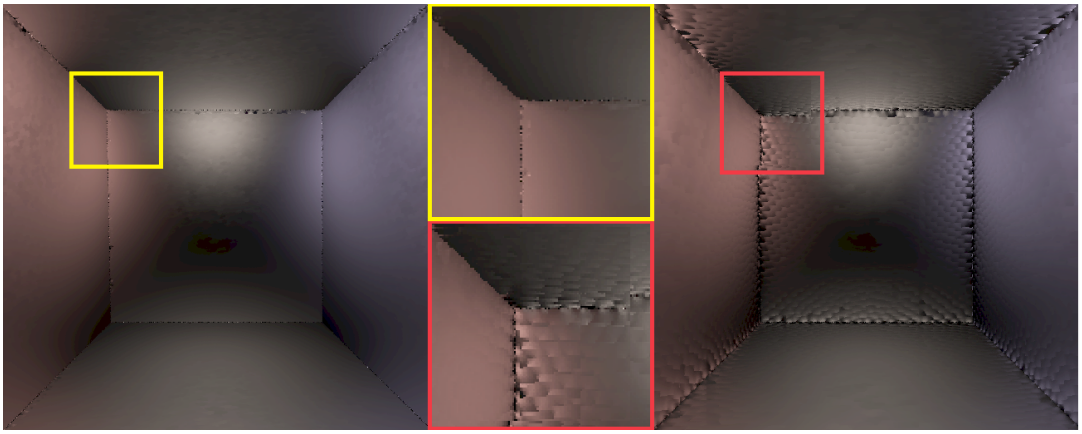

Figure 3.17: Another comparison on a glossy Cornell Box scene, both images are rendered using radiance caching. The left image is rendered with the missing angular term. The right image is rendered without the missing angular term. Notice the artifacts around the edges in the image on the right.

$$H_x \lambda_l^m = \sum_{j=1}^{M} H_x(L_{\triangle_j} F_{\triangle_j}(x)\, y_l^m(\omega_i)). \tag{3.40}$$

In this current step, we only focus on the gradient term. Expanding Equation 3.40, we get:

$$\nabla_x \lambda_l^m = \sum_{j=1}^{M} \left( L_{\triangle_j} \nabla_x \left( y_l^m(\omega_i) F_{\triangle_j}(x) \right) + F_{\triangle_j}(x)\, y_l^m(\omega_i) \nabla_x L_{\triangle_j} \right). \tag{3.41}$$

The first term $F_{\triangle_j}(x)\, y_l^m(\omega_i)$ inside the sum is the gradient term used in the current radiance caching. The second term $F_{\triangle_j}(x)\, y_l^m(\omega_i) \nabla_x L_{\triangle_j}$ was mistakenly believed to be negligeable and was missing in previous computations. Despite the absence of a thorough investigation, we found two promising approaches to compute the missing term.

**Analytical Approach.** One possible solution is to use an analytical approach. Define $L(x, \triangle)$ as $L_{\triangle_j}$, for the record located at $x$, $L(x, \triangle)$ can be analytically computed by

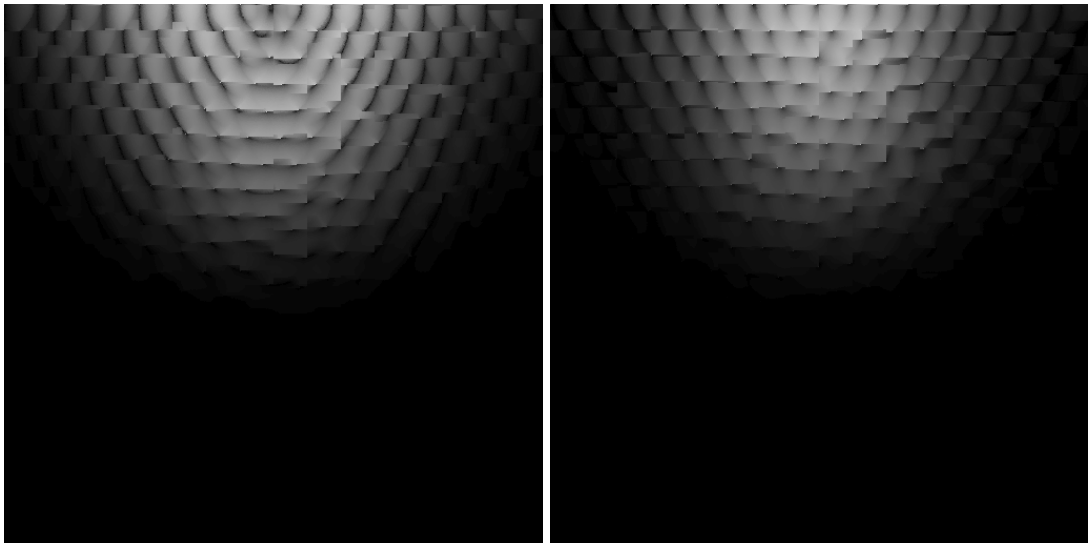

Figure 3.18: Left: contribution of the current gradient term toward the final image in the scene with two glossy surfaces. Right: the contribution of the missing gradient term in the same scene. Clearly, the contribution from the missing term cannot be ignored.

hemispherical integration:

$$L(x, \triangle) = \int_{\Omega} L_i(\triangle, \omega_i)\, f_r(\omega_o, \omega_i)\, (n \cdot \omega_i)\, d\omega_i. \tag{3.42}$$

In the formula above, $L_i(\triangle, \omega_i)$ is the incident radiance from direction $\omega_i$ at the triangle. We can safely assume that it is constant because the location of the triangle does not change, so the derivative of $L_i(\triangle, \omega_i)$ is zero. Then the derivative of $L(x, \triangle)$ is:

$$\frac{\partial L(x, \triangle)}{\partial x} = \int_{\Omega} L_i(\triangle, \omega_i)\, (n \cdot \omega_i)\, \frac{\partial f_r(\omega_o, \omega_i)}{\partial \omega_o}\, \frac{\partial \omega_o}{\partial x}\, d\omega_i. \tag{3.43}$$

It is very difficult to analytically compute the derivative of an arbitrary BRDF $f_r$. To derive Equation 3.43 analytically, we assume that the surface reflection model used is the Phong reflection model. Let $\omega_r$ denote the direction that a mirror reflected ray of light would take from this point on the surface and let $\alpha$ denote the roughness of the surface. The analytical formulation of the Phong BRDF is:

$$f_r(\omega_o, \omega_i) = \rho \frac{(\alpha+1)\,(\omega_o \cdot \omega_r)^{\alpha}}{2\pi}. \tag{3.44}$$

With this assumption, the term $\partial f_r(\omega_o, \omega_i)\,/\partial \omega_o$ can be analytically derived:

$$\frac{\partial f_r(\omega_o, \omega_i)}{\partial \omega_o} = \frac{\rho\,\alpha\,(\alpha+1)}{2\,\pi}\,\omega_r\,(\omega_o \cdot \omega_r)^{\alpha-1}\,. \tag{3.45}$$

Now all we need is use Monte Carlo method to compute the integral in Equation 3.43.

**Spherical Harmonic Approach.** We can also avoid computing derivatives of an arbitrary BRDF using spherical harmonics. First, we project the incident radiance distribution at a hemispherical sample onto the spherical harmonic basis functions to get the spherical harmonic coefficients of incident radiance. Then, we can easily compute outgoing radiance toward location $x$ using the BRDF matrix $M$:

$$L(x, \triangle) = M\,\Lambda_i\,Y \tag{3.46}$$

where $\Lambda_i$ is the spherical harmonic coefficients vector of incident radiance and $Y$ is a vector containing the value of spherical harmonic basis function in direction $\triangle \rightarrow x$. The incident radiance, and the BRDF matrix are both constant, and the gradient, and the Hessian matrix of spherical harmonic basis function is already computed, so the gradient and the Hessian matrix are:

$$\nabla_x L(x, \triangle) = M\,\Lambda_i\,\nabla_x Y, \tag{3.47}$$

$$H_x(L(x, \triangle)) = M\,\Lambda_i\,H_x(Y). \tag{3.48}$$

### 3.8.3 Discussion

The spherical harmonic approach does not depend on any assumption about the surface type. Compared to the analytical approach, it is able to handle arbitrary surface. But this approach relies on accurately computing the spherical harmonic coefficients, which requires a large amount of hemispherical samples. On the other hand, the analytic approach does not involve gradient computation for each spherical harmonic coefficients, which can reduce computing time by a factor of $(l_{\max} + 1)^2$.

Both approaches above require additional hemispherical sampling to gather the incident radiance information for each hemispherical sample of a record. This process is very computationally costly, and certain optimizations or approximations are necessary. We can potentially apply techniques such as importance sampling to reduce the number of samples needed. We can also take only direct illumination into account for the secondary hemispherical sampling, so that we can control the computing time in the same order of magnitude as before.

## 3.9 Conclusion and Future Work

In this chapter, we investigated a new radiance caching algorithm by combining and extending the previous work on irradiance caching and radiance caching. We introduced and discussed new error metrics based on the second-order Taylor expansion in a radiance caching context. We reused the technique of Schwarzhaupt et al. [2012] to take occlusion changes into account.

Similar to irradiance caching, the second-order Taylor expansion of radiance is used as an approximation of the ground truth. Like previous work on radiance caching, spherical harmonics are used to store the information about incident radiance distribution over the hemisphere. We further improved the record placement with the new outgoing radiance caching strategy without any additional computation time. Our results showed that the record density can vary according to incident lighting changes on surface and also

depends on the type of surface.

However, we discovered some visible artifacts in the images produced in some scenes. Our further investigation showed that the assumption that the radiance scattered from hemispherical samples remain constant inside cache valid region caused this type of artifacts. This assumption used in our and previous approaches is not valid in specific configurations of scenes with two adjacent glossy surfaces. We analyzed the failure cases and concluded that this assumption should be discarded to make radiance caching robust in more arbitrary scenes. We also proposed two potential solutions to solve this problem.

Naturally, the next step is to further investigate these two potential solutions that can correctly compute gradient and the Hessian of radiance, and should ultimately produce artifact-free images.

# CHAPTER 4

# CONCLUSION

In this thesis, we successfully combine and extende the previous work on irradiance caching and radiance caching. We propose a new occlusion-aware radiance caching algorithm with new error metrics based on the second-order Taylor expansion.

Similar to irradiance caching, the second-order Taylor expansion of radiance is used as an approximation of the ground truth. Like previous work on radiance caching, spherical harmonics are used to store the information about incident radiance distribution over the hemisphere. We also further improved the records placement with the new outgoing radiance caching algorithm without any additional computation time. Our results showe that record density can vary according to the incident lighting changes on a surface and also depends on the type of surfaces.

However, we discover some visible artifacts in images produced in some scenes. Our further investigation showed that the assumption that the radiance scattered from hemispherical samples remain constant within record valid region, caused this type of artifacts. This assumption used in our approach and previous approaches is not valid in specific scene configurations with two adjacent glossy surfaces. We analyze the failure cases and concluded that this assumption should be discarded to make radiance caching robust in more arbitrary scenes. We propose two potential solutions to solve this problem.

The obvious next step is to conduct a more thorough investigation of the two solutions that we proposed. As we mentioned in Section 3.8.2, due to the secondary hemispherical sampling, both approaches cannot compute the previously ignored missing term in the gradient computation accurately and efficiently yet. The logical first step is to experiment with the new approaches with only direct illumination for secondary hemisphere sampling. Then we can try to incorporate them with advanced sampling

techniques to make our approaches efficient enough to handle both direct and indirect illuminations during the secondary hemispherical sampling.

Currently, irradiance and radiance cachings only place records on visible surfaces. The record placement is non-recursive. If we allow records to be placed recursively, we would be able to use records placed on a non-visible surfaces to accelerate the generation of new records. This approach could potentially accelerate global illumination computations for indoor scenes.

Our ultimate goal is to find a caching strategy that can easily be incorporated in other global illumination algorithms, and naturally generate optimized record placement on all surfaces in scenes with reasonable user-defined parameter tweaking.

## BIBLIOGRAPHY


[1] Thomas Annen, Jan Kautz, Fredo Durand, and Hans-Peter Seidel. Spherical harmonic Gradients for Mid-Range Illumination. In *Rendering Techniques*, pages 331–336. Eurographics Association, 2004.

[2] Iliyan Georgiev, Jaroslav Krivanek, Tomáš Davidovič, and Philipp Slusallek. Light Transport Simulation with Vertex Connection and Merging. *ACM Transactions on Graphics*, 31(6):192:1–10, 2012.

[3] Toshiya Hachisuka and Henrik Wann Jensen. Stochastic Progressive Photon Mapping. *ACM Transactions on Graphics*, 28(5):141:1–8, 2008.

[4] Nicolas Holzschuch and François X. Sillion. An Exhaustive Error-Bounding Algorithm for Hierarchical Radiosity. *Comput. Graph. Forum*, 17(4):197–218, 1998.

[5] Wojciech Jarosz, Volker Schonefeld, Leif Kobbelt, and Henrik Wann Jensen. Theory Analysis and Applications of 2D Global Illumination. *ACM Transactions on Graphics*, 31(5):243–253, 2012.

[6] Henrik Wann Jensen. Global Illumination using Photon Maps. In *Rendering Techniques '96*, pages 21–30. Springer-Verlag, 1996.

[7] James T. Kajiya. The Rendering Equation. *SIGGRAPH Comput. Graph.*, 20(4): 143–150, August 1986.

[8] Jan Kautz, Peter-Pike Sloan, and John Snyder. Fast, Arbitrary BRDF Shading for Low-frequency Lighting Using Spherical Harmonics. In *Proceedings of the 13th Eurographics Workshop on Rendering*, EGRW '02, pages 291–296, 2002.

[9] Claude Knaus and Matthias Zwicker. Progressive Photon Mapping: A Probabilistic Approach. *ACM Transactions on Graphics*, 30(3):25:1–13, 2011.

[10] Jaroslav Krivanek. *Radiance Caching for Global Illumination Computation on Glossy Surfaces*. Ph.d. thesis, Université de Rennes 1 and Czech Technical University in Prague, December 2005.

[11] Eric P. Lafortune and Yves D. Willems. Bi-Directional Path Tracing. In *Proceedings of Compugraphics '93*, pages 145–153, 1993.

[12] N.Metropolis and S.Ulam. The Monte Carlo method. *Journal of the American statistical Association*, 44(247):335–341, 1949.

[13] Derek Nowrouzezahrai, Patricio Simari, and Eugene Fiume. Sparse Zonal Harmonic Factorization for Efficient SH Rotation. *ACM Transactions on Graphics*, 31 (3):23:1–9, 2012.

[14] Matt Pharr and Greg Humphreys. *Physically Based Rendering, Second Edition: From Theory To Implementation*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2010.

[15] Jorge Schwarzhaupt, Henrik Wann Jensen, and Wojciech Jarosz. Practical Hessian-Based Error Control for Irradiance Caching. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia 2012)*, 31(6):193–193, 2012.

[16] Peter-Pike Sloan, Jan Kautz, and John Snyder. Precomputed Radiance Transfer for Real-time Rendering in Dynamic, Low-frequency Lighting Environments. *ACM Transactions on Graphics*, 21(3):527–536, 2002.

[17] Eric Tabellion and Arnauld Lamorlette. An Approximate Global Illumination System for Computer Generated Films. *ACM Transactions on Graphics*, 23(3):469–476, 2004.

[18] Eric Veach and Leonidas J. Guibas. Optimally Combining Sampling Techniques for Monte Carlo Rendering. In *Proceedings of the 22nd Annual Conference on*

*Computer Graphics and Interactive Techniques*, SIGGRAPH '95, pages 419–428, 1995.

[19] Eric Veach and Leonidas J. Guibas. Metropolis Light Transport. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '97, pages 65–76, 1997.

[20] Greg Ward and Paul Heckbert. Irradiance Gradients. In *Eurographics Rendering Workshop*, pages 85–98, May 1992.

[21] Gregory J. Ward, Francis M. Rubinstein, and Robert D. Clear. A Ray Tracing Solution for Diffuse Interreflection. *SIGGRAPH Comput. Graph.*, 22(4):85–92, 1988.

# Appendix I

## Gradient and Hessian of Spherical Harmonic Coefficients

In the proposed radiance caching algorithm, computing the gradient and Hessian of spherical harmonic coefficients on the tangent plane (directions $x$ and $y$) is essential. In this Appendix, I present some details of derivation.

To compute the gradient and the Hessian matrix of coefficients we expand the gradient and the Hessian matrix of Equation 3.17:

$$\nabla_x \lambda_l^m = \sum_{j=1}^{M} L_{\triangle_j} \nabla_x (F_{\triangle_j}(x)\, y_l^m(\omega_i)) \tag{I.1}$$

$$\nabla_x \lambda_l^m = \sum_{j=1}^{M} L_{\triangle_j} (y_l^m(\omega_i)\, \nabla_x F_{\triangle_j}(x) + F_{\triangle_j}(x)\, \nabla_x y_l^m(\omega_i)) \tag{I.2}$$

$$H_x \lambda_l^m = \sum_{j=1}^{M} L_{\triangle_j} H_x (F_{\triangle_j}(x)\, y_l^m(\omega_i)) \tag{I.3}$$

$$H_x \lambda_l^m = A + B + C + D \tag{I.4}$$

where,

$$A = \sum_{j=1}^{M} L_{\triangle_j} \nabla_x y_l^m(\omega_i) \times (\nabla_x F_{\triangle_j}(x))^T \tag{I.5}$$

$$B = \sum_{j=1}^{M} L_{\triangle_j} \nabla_x F_{\triangle_j}(x) \times (\nabla_x y_l^m(\omega_i))^T \tag{I.6}$$

$$C = \sum_{j=1}^{M} L_{\triangle_j} y_l^m(\omega_i)\, H_x F_{\triangle_j}(x) \tag{I.7}$$

$$D = \sum_{j=1}^{M} L_{\triangle_j} F_{\triangle_j}(x) H_x(y_l^m(\omega_i)) \tag{I.8}$$

$\nabla_x F_{\triangle_j}$ and $H_x F_{\triangle_j}$ are computed by Holzschuch and Sillion [1998], so the terms left to compute are $\nabla_x y_l^m$ and $H_x(y_l^m)$.

## I.1 Definitions of Spherical Harmonic Basis Function

### I.1.1 Spherical Coordinate System

First we define the spherical harmonic basis function in angular form:

$$y_l^m = \begin{cases} \dfrac{\sqrt{1+2l}\,P_l(\cos(\theta))}{2\sqrt{\pi}} & m = 0 \\ \sqrt{\dfrac{(1+2l)(l-m)!}{(l+m)!}}\dfrac{\cos(m\phi)\,P_l^m(\cos(\theta))}{\sqrt{2\pi}} & m > 0 \\ -\sqrt{\dfrac{(1+2l)(l+m)!}{(l-m)!}}\dfrac{\sin(m\phi)\,P_l^{-m}(\cos(\theta))}{\sqrt{2\pi}} & m < 0 \end{cases} \tag{I.9}$$

where $P_l^m(x)$ are Legendre polynomials.

### I.1.2 Cartesian Coordinate System

Then we need to define the spherical harmonic basis function in cartesian coordinate system:

$\theta$ and $\phi$ can be computed using $x$, $y$, and $z$:

$$\theta = \arccos\left(\frac{z}{r}\right) = \arccos\left(\frac{z}{\sqrt{x^2+y^2+z^2}}\right) \tag{I.10}$$

$$\phi = \begin{cases} \arccos\left(\frac{x}{r*\sin\theta}\right) = \arccos\left(\frac{x}{\sqrt{x^2+y^2}}\right) & y \geq 0 \\ 2\pi - \arccos\left(\frac{x}{r*\sin\theta}\right) = 2\pi - \arccos\left(\frac{x}{\sqrt{x^2+y^2}}\right) & y < 0 \end{cases} \tag{I.11}$$

Replacing the $\theta$ and $\phi$ in the angular definition, then we have:

$$y_l^m = \begin{cases} \frac{\sqrt{1+2l}}{2\sqrt{\pi}} P_l\left(\frac{z}{r}\right) & m = 0 \\ P_l^m\left(\frac{z}{r}\right) \frac{e_1}{\sqrt{2\pi}} \cos\left(m\left(2\pi - \arccos\left(\frac{x}{\sqrt{x^2+y^2}}\right)\right)\right) & y < 0, m > 0 \\ P_l^m\left(\frac{z}{r}\right) \frac{e_1}{\sqrt{2\pi}} \cos\left(m \arccos\left(\frac{x}{\sqrt{x^2+y^2}}\right)\right) & y \geq 0, m > 0 \\ -P_l^{-m}\left(\frac{z}{r}\right) \frac{e_2}{\sqrt{2\pi}} \sin\left(m\left(2\pi - \arccos\left(\frac{x}{\sqrt{x^2+y^2}}\right)\right)\right) & y < 0, m < 0 \\ -P_l^{-m}\left(\frac{z}{r}\right) \frac{e_2}{\sqrt{2\pi}} \sin\left(m \arccos\left(\frac{x}{\sqrt{x^2+y^2}}\right)\right) & y \geq 0, m > 0 \end{cases}$$

where,

$$e_1 = \sqrt{\frac{(1+2l)(l-m)!}{(l+m)!}}, e_2 = \sqrt{\frac{(1+2l)(l+m)!}{(l-m)!}} \tag{I.12}$$

## I.2 Derivatives

Then we compute the derivative of $y_l^m$ about $x$,

$$\frac{\partial y_l^m}{\partial x} = \begin{cases} -\dfrac{(1+l)\sqrt{1+2l}\,x\,z\,(z P_l(c) - r P_{l+1}(c))}{2\sqrt{\pi}\,e_3^2\,r^2} & m = 0 \\ \dfrac{e_1}{\sqrt{2\pi}}\left(\dfrac{m \sin\phi\, P_l^m(c)\, S}{e_3} - \dfrac{x z C\left((1+l) z P_l^m(c) + (-1-l+m)\, r P_{1+l}^m(c)\right)}{e_3^2\, r^2}\right) & m > 0 \\ \dfrac{e_2}{\sqrt{2\pi}}\left(\dfrac{m \mid \sin\phi \mid C P_l^{-m}(c)}{e_3} + \dfrac{x z\left((1+l) z P_l^{-m}(c) - (1+l+m)\, r P_{l+1}^{-m}(c)\right) S}{e_3^2\, r^2}\right) & m < 0 \end{cases}$$

with:

$$c = \frac{z}{r} \tag{I.13}$$

$$e_3 = \sqrt{x^2 + y^2} \tag{I.14}$$

$$C = \cos(m\phi), \quad S = \sin(m\phi) \tag{I.15}$$

The derivative of $y_l^m$ about $y$ is:

$$\frac{\partial y_l^m}{\partial x} = \begin{cases} -\dfrac{(1+l)\sqrt{1+2l}\,y\,z\,(z P_l(c) - r P_{l+1}(c))}{2\sqrt{\pi}\,e_3^2\,r^2} & m = 0 \\ -\dfrac{y e_1}{\sqrt{2\pi}}\left(\dfrac{z C\left((1+l) z P_l^m(c) + (-1-l+m)\, r P_{l+1}^m(c)\right)}{e_3^2\, r^2} + \dfrac{m x P_l^m(c)\, S}{\mid \sin\phi \mid e_3^3}\right) & m > 0 \\ \dfrac{y e_2}{\sqrt{2\pi}}\left(\dfrac{z\left((1+l) z P_l^{-m}(c) - (1+l+m)\, r P_{l+1}^{-m}(c)\right) S}{e_3^2\, r^2} - \dfrac{m x C P_l^{-m}(c)}{\sin\phi\, e_3^3}\right) & m < 0 \end{cases}$$

### I.3 Hessian Matrix

Computing the derivative of $\dfrac{\partial y_l^m}{\partial x}$ about $x$, we get $\dfrac{\partial^2 y_l^m}{\partial x^2}$.

$$\frac{\partial^2 y_l^m}{\partial x^2} = A_1 \times (B_1 + C_1) \tag{I.16}$$

where,

$$A_1 = \begin{cases} e_4(1+l)\sqrt{1+2l}\,z & m=0 \\ e_4 e_1 & m>0 \\ e_4 e_2 & m<0 \end{cases}$$

$$B_1 = \begin{cases} z\,e_5\,P_l(c) + r\left(x^4 - e_5\right)P_{l+1}(c) & m=0 \\ e_{11}\left(e_{10}x^2\,C\,P^m_{l+2}(c) - r\,P^m_{l+1}(c)\left(\left(e_5 - x^4\right)C - e_8\,S\right)\right) & m>0, y\geq 0 \\ e_{11}\left(e_{10}x^2\,C\,P^m_{l+2}(c) - r\,P^m_{l+1}(c)\left(\left(e_5 - x^4\right)C + e_8\,S\right)\right) & m>0, y<0 \\ -e_9\left(e_{12}x^2\,P^{-m}_{l+2}(c)\,S - r\,P^{-m}_{l+1}(c)\left(e_8\,C + \left(e_5 - x^4\right)S\right)\right) & m<0, y\geq 0 \\ -e_9\left(e_{12}x^2\,P^{-m}_{l+2}(c)\,S + r\,P^{-m}_{l+1}(c)\left(e_8\,C + \left(x^4 - e_5\right)S\right)\right) & m<0, y<0 \end{cases}$$

$$C_1 = \begin{cases} (2+l)\,x^2\,z\,r^2\,P_{l+2}(c) & m=0 \\ -P^m_l(c)\,(e_6\,C + e_8\,e_7\,S) & m>0, y\geq 0 \\ P^m_l(c)\,(-e_6\,C + e_8\,e_7\,S) & m>0, y<0 \\ P^{-m}_l(c)\,(e_8\,e_7\,C - e_6\,S) & m<0, y\geq 0 \\ P^{-m}_l(c)\,(e_8\,e_7\,C + e_6\,S) & m<0, y<0 \end{cases}$$

with:

$$e_4 = \frac{1}{2\sqrt{\pi}\left(x^2+y^2\right)^2\left(x^2+y^2+z^2\right)^2} \tag{I.17}$$

$$e_5 = 3x^4 - y^2\left(y^2+z^2\right) + x^2\left(2y^2 + (2+l)z^2\right) \tag{I.18}$$

$$e_6 = m^2\,y^2\,r^4 - (1+l)\,z^2\,e_5 \tag{I.19}$$

$$e_7 = x^2 + y^2 + (2+l)z^2 \tag{I.20}$$

$$e_8 = 2mx \mid \sin\phi \mid e_3 r^2 \tag{I.21}$$

$$e_9 = (1+l+m)z \tag{I.22}$$

$$e_{10} = (2+l-m)zr^2 \tag{I.23}$$

$$e_{11} = (1+l-m)z \tag{I.24}$$

$$e_{12} = (2+l+m)zr^2 \tag{I.25}$$

Similarly, $\frac{\partial^2 y_l^m}{\partial y^2}$ is the derivative of $\frac{\partial y_l^m}{\partial y}$ about $y$ :

$$\frac{\partial^2 y_l^m}{\partial y^2} = A_2 \times (B_2 + C_2) \tag{I.26}$$

where,

$$A_2 = \begin{cases} e_4(1+l)\sqrt{1+2l}\,z & m = 0 \\ e_4 e_1 & m > 0 \\ e_4 e_2 & m < 0 \end{cases}$$

$$B_2 = \begin{cases} -e_{13} z P_l(c) + r e_{15} P_{l+1}(c) & m = 0 \\ e_{11}\left(e_{10} y^2 C P_{l+2}^m(c) + r P_{l+1}^m(c)\left(e_{15} C - e_8 S\right)\right) & m > 0, y \geq 0 \\ e_{11}\left(e_{10} y^2 C P_{l+2}^m(c) + r P_{l+1}^m(c)\left(e_{15} C + e_8 S\right)\right) & m > 0, y < 0 \\ -e_9\left(e_{12} y^2 P_{l+2}^{-m}(c) S + r P_{l+1}^{-m}(c)\left(e_8 C + e_{15} S\right)\right) & m < 0, y \geq 0 \\ -e_9\left(e_{12} y^2 P_{l+2}^{-m}(c) S - r P_{l+1}^{-m}(c)\left(e_8 C - e_{15} S\right)\right) & m < 0, y < 0 \end{cases}$$

$$C_2 = \begin{cases} (2+l)\,y^2\,z\,r^2\,P_{l+2}(c) & m=0 \\ P_l^m(c)\,(e_8\,e_7\,S - e_{14}\,C) & m>0, y\geq 0 \\ -P_l^m(c)\,(e_{14}\,C + e_8\,e_7\,S) & m>0, y<0 \\ P_l^{-m}(c)\,(e_8\,e_7\,C + e_{14}\,S) & m<0, y\geq 0 \\ -P_l^{-m}(c)\,(e_8\,e_7\,C - e_{14}\,S) & m<0, y<0 \end{cases}$$

with:

$$e_{13} = x^4 - x^2\left(2y^2 - z^2\right) - y^2\left(3y^2 + (2+l)z^2\right) \tag{I.27}$$

$$e_{14} = m^2\,x^2\,r^4 + (1+l)\,z^2\,e_{13} \tag{I.28}$$

$$e_{15} = x^4 + x^2\left(-y^2 + z^2\right) - 2y^2\left(y^2 + (2+l)\,z^2\right) \tag{I.29}$$

Again, computing the derivative of $\frac{\partial y_l^m}{\partial x}$ about $y$ to get $\frac{\partial^2 y_l^m}{\partial x\,\partial y}$:

$$\frac{\partial^2 y_l^m}{\partial x\,\partial y} = A_3 \times (B_3 + C_3) \tag{I.30}$$

where,

$$A_3 = \begin{cases} e_4(1+l)\sqrt{1+2l}\,x\,y\,z & m=0 \\ \dfrac{e_4\,e_1}{y} & m>0 \\ -\dfrac{e_4\,e_2}{y} & m<0 \end{cases}$$

$$B_3 = \begin{cases} z\,e_{17}\,P_l(c) - r\,e_{18}\,P_{l+1}(c) & m = 0 \\ e_{11}\left(e_{10}\,x y^2\,C\,P^m_{l+2}(c) - r\,P^m_{l+1}(c)\left(x y^2\,e_{18}\,C - e_{16} \mid y \mid S\right)\right) & m > 0, y \geq 0 \\ e_{11}\left(e_{10}\,x y^2\,C\,P^m_{l+2}(c) - r\,P^m_{l+1}(c)\left(x y^2\,e_{18}\,C + e_{16} \mid y \mid S\right)\right) & m > 0, y < 0 \\ e_9\left(e_{12}\,x y^2\,P^{-m}_{l+2}(c)\,S - r\,P^{-m}_{l+1}(c)\left(e_{16} \mid y \mid C + x y^2\,e_{18}\,S\right)\right) & m < 0, y \geq 0 \\ e_9\left(e_{12}\,x y^2\,P^{-m}_{l+2}(c)\,S - r\,P^{-m}_{l+1}(c)\left(x y^2\,e_{18}\,S - e_{16} \mid y \mid C\right)\right) & m < 0, y < 0 \end{cases}$$

$$C_3 = \begin{cases} (2+l)\,z\,r^2\,P_{l+2}(c) & m = 0 \\ P^m_l(c)\,(e_{19}\,C - e_{16}\,e_7\,S) \mid y \mid & m > 0, y \geq 0 \\ P^m_l(c)\,(e_{19}\,C + e_{16}\,e_7\,S) \mid y \mid & m > 0, y < 0 \\ P^{-m}_l(c)\,(e_{16}\,e_7\,C + e_{19}\,S) \mid y \mid & m < 0, y \geq 0 \\ P^{-m}_l(c)\,(e_{19}\,S - e_{16}\,e_7\,C) \mid y \mid & m < 0, y < 0 \end{cases}$$

with:

$$e_{16} = m\left(-x^2 + y^2\right) r^2 \tag{I.31}$$

$$e_{17} = 4x^2 + 4y^2 + (3+l)z^2 \tag{I.32}$$

$$e_{18} = 3x^2 + 3y^2 + (5+2l)z^2 \tag{I.33}$$

$$e_{19} = x \mid y \mid \left(m^2\,r^4 + (1+l)\,z^2\,e_{17}\right) \tag{I.34}$$

Finally $\dfrac{\partial^2 y^m_l}{\partial y\,\partial x}$ equals the derivative of $\dfrac{\partial y^m_l}{\partial y}$ about $x$ :

$$\frac{\partial^2 y^m_l}{\partial y\,\partial x} = A_4 \times (B_4 + C_4) \tag{I.35}$$

where,

$$A_4 = \begin{cases} e_4(1+l)\sqrt{1+2l}\,xyz & m=0 \\ e_{20}\,e_1 & m>0 \\ -e_{20}\,e_2 & m<0 \end{cases}$$

$$B_4 = \begin{cases} z\,e_{17}\,P_l(c) - r\,e_{18}\,P_{l+1}(c) & m=0 \\ e_{11}\left(e_{10}\,x \mid y \mid C\,P^m_{l+2}(c) - r\,P^m_{l+1}(c)\,(x \mid y \mid e_{18}\,C + e_{16}\,S)\right) & m>0, y\geq 0 \\ e_{11}\left(e_{10}\,x \mid y \mid C\,P^m_{l+2}(c) - r\,P^m_{l+1}(c)\,(x \mid y \mid e_{18}\,C - e_{16}\,S)\right) & m>0, y<0 \\ e_9\left(e_{12}\,x \mid y \mid P^{-m}_{l+2}(c)\,S + r\,P^{-m}_{l+1}(c)\,(e_{16}\,C - x \mid y \mid e_{18}\,S)\right) & m<0, y\geq 0 \\ e_9\left(e_{12}\,x \mid y \mid P^{-m}_{l+2}(c)\,S - r\,P^{-m}_{l+1}(c)\,(e_{16}\,C + x \mid y \mid e_{18}\,S)\right) & m<0, y<0 \end{cases}$$

$$C_4 = \begin{cases} (2+l)\,z\,r^2\,P_{l+2}(c) & m=0 \\ P^m_l(c)\,(e_{19}\,C + e_{16}\,e_7\,S) & m>0, y\geq 0 \\ P^m_l(c)\,(e_{19}\,C - e_{16}\,e_7\,S) & m>0, y<0 \\ P^{-m}_l(c)\,(e_{19}\,S - e_{16}\,e_7\,C) & m<0, y\geq 0 \\ P^{-m}_l(c)\,(e_{16}\,e_7\,C + e_{19}\,S) & m<0, y<0 \end{cases}$$

with:

$$e_{20} = \frac{y}{r^4\sqrt{2\pi}\sqrt{y^2\left(x^2+y^2\right)^4}} \tag{I.36}$$