

Uniwersytet Wrocławski
Wydział Matematyki i Informatyki
Instytut Informatyki

Marcin Chajec

Mapy fotonów w oświetleniu globalnym

Praca magisterska

Praca wykonana pod kierunkiem
dr Andrzeja Łukaszewskiego

Wrocław 2005

Streszczenie

Celem pracy jest omówienie sposobów wykorzystania mapy fotonów w obliczaniu oświetlenia globalnego dla poprawienia jakości generowanych obrazów i szybkości ich tworzenia w stosunku do metod opartych jedynie na śledzeniu promieni. Zaprezentowane zostały przede wszystkim osiągnięcia ostatnich lat, które odniosły największy sukces i są dziś szeroko wykorzystywane zarówno w aplikacjach komercyjnych, jak i w dalszych badaniach naukowych.

W pracy przedstawiono teoretyczny opis prezentowanych metod. Omówiono także ich implementację, wraz z najczęściej spotykanymi problemami technicznymi i sposobami ich rozwiązywania.

Każde z prezentowanych rozwiązań, o ile jest to możliwe, porównywane jest z jego odpowiednikiem wśród metod wykorzystujących jedynie śledzenie promieni. Porównania dotyczą jakości otrzymywanych wyników, kosztu ich uzyskania (czasowego i pamięciowego), poprawności fizycznej oraz łatwości realizacji rozwiązań.

Praca została podzielona na siedem rozdziałów. W pierwszym zawarto krótkie wprowadzenie do prezentowanego tematu oraz opisano jej cele. Drugi definiuje podstawowe pojęcia, konwencje i oznaczenia. Kolejny rozdział prezentuje w skrócie algorytmy śledzenia promieni, które w tej pracy są punktem odniesienia dla map fotonów. Rozdział czwarty zawiera wprowadzenie do map fotonów. Znajduje się w nim definicja mapy fotonów, opis sposobu jej tworzenia, optymalizowania i wykorzystania. W rozdziale piątym znajduje się przegląd metod wykorzystania map fotonów do wyznaczania oświetlenia bezpośredniego. Szósty mówi o oświetleniu pośrednim oraz globalnym. Rozdział siódmy zawiera podsumowanie.

Większość opisanych metod związanych zarówno z mapami fotonów, jak i śledzeniem promieni, została zaimplementowana w programie dołączonym do pracy na płycie CD. Dokładny opis programu znajduje się w dodatku A. Wszystkie prezentowane w pracy obrazy (jeśli wyraźnie nie zaznaczono, że jest inaczej) zostały utworzone przy pomocy tego programu.

Spis treści

1. Wstęp	6
1.1. Oświetlenie globalne	6
1.2. Zastosowania oświetlenia globalnego	7
1.3. Klasy algorytmów	9
1.4. Cele pracy	10
2. Podstawy teoretyczne	11
2.1. Natura światła a grafika komputerowa	11
2.2. Oznaczenia	12
2.3. Radiometria	13
2.4. Odbicie i załamanie światła oraz funkcja BRDF	14
2.4.1. Odbicie	14
2.4.2. Załamanie	15
2.4.3. Funkcja BRDF	16
2.5. Równanie renderingu	16
3. Śledzenie promieni	17
3.1. Podstawy	17
3.2. Metody monte carlo	21
4. Mapa fotonów	26
4.1. Wprowadzenie	26
4.2. Konstrukcja mapy fotonów	26
4.2.1. Emisja fotonów	27
4.2.2. Śledzenie fotonów	29
4.2.3. Zachowywanie fotonów	30
4.3. Struktura danych	31
4.3.1. Optymalizacja mapy	32
4.3.2. Opis fotonu	33
4.4. Wyszukiwanie fotonów	34
4.5. Przykładowa mapa	35

5. Oświetlenie bezpośrednie	37
5.1. Wprowadzenie	37
5.2. Mapa fotonów w oświetleniu bezpośrednim	38
5.3. Wyznaczanie zbioru światła widocznych	40
5.3.1. Algorytm	40
5.3.2. Przykłady	41
5.3.3. Podsumowanie	45
5.4. Mapa cienia	45
5.4.1. Struktura danych	45
5.4.2. Zastosowanie	46
5.4.3. Przykłady	48
5.5. Podsumowanie	53
6. Oświetlenie pośrednie i globalne	54
6.1. Wprowadzenie	54
6.2. Przyspieszanie zbieżności śledzenia ścieżek	55
6.2.1. Podstawy teoretyczne i algorytm	55
6.2.2. Przykłady	57
6.2.3. Podsumowanie	59
6.3. Wyznaczanie radiancji	60
6.3.1. Estymacja radiancji	60
6.3.2. Kształt obszaru wyszukiwania fotonów	61
6.3.3. Filtrowanie	63
6.3.4. Automatyczne wyznaczanie obszaru wyszukiwania	65
6.4. Renderowanie oświetlenia globalnego	66
6.4.1. Bezpośrednia wizualizacja mapy fotonów	66
6.4.2. Algorytm dwufazowy	71
6.5. Rozszerzenia map fotonów	78
6.5.1. Wstępne obliczanie irradiancji	78
6.5.2. Sterowanie gęstością mapy fotonów	82
6.6. Podsumowanie	90
7. Podsumowanie	91
A Zaimplementowane programy	93
A1. Informacje ogólne	93
A2. mgfview	94
A3. rat	95
A3.1. Składnia języka	95
A3.2. Spis komend	96
A3.3. Przykładowe skrypty i wyniki renderingu	100

Spis rysunków

1.1. Porównanie oświetlenia bezpośredniego i pośredniego.	7
1.2. Wygenerowane komputerowo obrazy wnętrza.	8
2.1. Prawa odbicia i załamania zwierciadlanego.	14
2.2. Odbicie rozproszone i połyskujące.	15
3.1. Drzewo promieni dla algorytmu śledzenia promieni.	19
3.2. Obraz utworzony algorytmem śledzenia promieni Whitted-a.	20
3.3. Drzewo promieni dla algorytmu śledzenia ścieżek.	22
3.4. Obrazy wygenerowane metodą śledzenia ścieżek.	24
4.1. Rodzaje fotonów.	31
4.2. Przykładowa mapa fotonów, mapa cienia oraz wyrenderowany za ich pomocą obraz.	35
5.1. Oświetlenie bezpośrednie w scenie z dwoma światłami. Po lewej rozwiązanie wzorcowe, po prawej uzyskane z mapy fotonów.	38
5.2. Oświetlenie bezpośrednie, wyniki uzyskane w podobnym czasie metodą śledzenia promieni (po lewej) i z mapy fotonów (po prawej).	39
5.3. Oświetlenie bezpośrednie (50 promieni/piksel) we wnętrzu boksu.	41
5.4. Oświetlenie bezpośrednie (50 promieni/piksel), zbliżenie na kwiatek.	43
5.5. Oświetlenie bezpośrednie (50 promieni/piksel) we wnętrzu klasy.	44
5.6. Oświetlenie bezpośrednie w scenie z jednym światłem.	48
5.7. Mapa cienia, obszary półcienia, różnice między obrazami.	49
5.8. Po lewej rozwiązanie wzorcowe z podrozdziału 5.2. utworzone z wykorzystaniem mapy cienia do wykrycia obszarów półcienia. Po prawej u góry podział na obszary światła/półcienia/cienia dla obu światel. Na dole mapa fotonów i obraz różnic między rozwiązaniami.	51
5.9. Po lewej: oświetlenie bezpośrednie wewnątrz biura utworzone z użyciem mapy cienia. Po prawej u góry obszary półcienia dla lamp sufitowych, na dole półcienie dla lampki biurkowej i mapa cienia.	52

6.1. Porównanie tradycyjnego śledzenia ścieżek (górny rząd), z śledzeniem wykorzystującym mapę fotonów do kierowania promieni (dolny rząd). Liczby pod obrazami oznaczają wykorzystaną ilość ścieżek na piksel. . .	57
6.2. Kaustyka uzyskana przy pomocy tradycyjnego śledzenia ścieżek (po lewej) i algorytmu wykorzystującego mapę fotonów do kierowania ścieżek (po prawej).	58
6.3. Irradiancja i radiancja w prostym pokoju biurowym wyznaczone na podstawie mapy zawierającej 10^5 fotonów.	61
6.4. Obraz styku ściany i podłogi przy różnych współczynnikach kompresji.	62
6.5. Porównanie wyników renderingu bez i z użyciem filtra.	64
6.6. Obraz biura utworzony metodą bezpośredniej wizualizacji mapy fotonów.	67
6.7. Obraz biura utworzony poprawioną metodą bezpośredniej wizualizacji.	68
6.8. Obraz biura z dodanymi kaustykami.	69
6.9. Zbliżenie kaustyk na powierzchni biurka.	70
6.10. Oświetlenie pośrednie wewnątrz sześcianu.	71
6.11. Oświetlenie bezpośrednie we wnętrzu pokoju dziennego.	74
6.12. Oświetlenie pośrednie (część rozproszona) we wnętrzu pokoju dziennego.	75
6.13. Zbliżenie kaustyki w pokoju dziennym.	76
6.14. Obrazy uzyskane przy pomocy algorytmu dwufazowego.	77
6.15. Irradiancja wyznaczona na podstawie mapy fotonów (po lewej), oszacowań z odchyleniem do 5° (w środku) i 20° (po prawej).	80
6.16. Oświetlenie pośrednie wyznaczone na podstawie oszacowań o maksymalnym odchyleniu 20°	81
6.17. Porównanie mapy utworzonej bez i z wykorzystaniem kontroli gęstości.	88
6.18. Mapy fotonów wykonane bez (po lewej) i z kontrolą gęstości (w środku) w scenie zawierającej obiekty kaustyczne. Po prawej znajduje się estymacja istotności (na górze) i pełne oświetlenie globalne (na dole) wykonane na podstawie mapy stworzonej z wykorzystaniem kontroli gęstości. . . .	89
A1. Zwierciadlana wersja sceny Cornell box	100
A2. Wnętrze biura oświetlone światłem dziennym.	102

Spis tabel

2.1. Oznaczenia wykorzystane w pracy.	12
2.2. Notacja Heckbert-a.	12
4.1. Algorytm wyszukiwania fotonów.	34
5.1. Statystyki dla metod z rysunku 5.3.	42
5.2. Statystyki dla metod z rysunku 5.4.	43
5.3. Statystyki dla metod z rysunku 5.6.	50
5.4. Porównanie śledzenia promieni z mapą cienia i bez dla rysunku 5.8. . .	51
6.1. Wpływ maksymalnego odchylenia normalnej na czas wyliczania irra- diancji.	80
6.2. Wpływ wykorzystania przybliżeń irradiancji na czas renderingu. . . .	81

Rozdział 1.

Wstęp

Rozdział ten stanowi wprowadzenie do tematyki, którą zajmuje się niniejsza praca. Definiuje problemy, które są przedmiotem rozważań, wylicza istniejące klasy metod ich rozwiązywania, zastosowania tych rozwiązań oraz przedstawia cele pracy.

1.1. Oświetlenie globalne

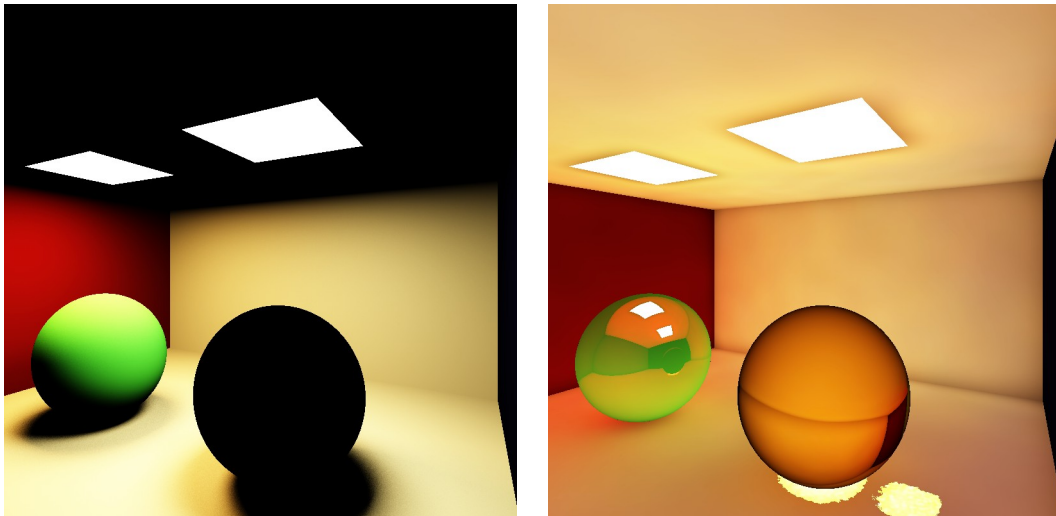
W realistycznej grafice komputerowej rozróżnia się dwa rodzaje oświetlenia:

oświetlenie bezpośrednie – jest to światło pochodzące bezpośrednio od powierzchni je emitującej (źródła światła),

oświetlenie pośrednie (niebezpośrednie) – jest to światło odbite lub załamane przynajmniej raz od momentu, gdy opuściło źródło światła.

W przypadku większości scen decydujący wpływ na jasność i kolor poszczególnych punktów, a przez to na wygląd całego obrazu, ma oświetlenie bezpośrednie. Jest ono zdecydowanie mocniejsze od oświetlenia pośredniego. W naturze odpowiada przede wszystkim za zjawisko cienia oraz półcienia w obszarach, w których widoczna jest tylko część źródła światła. Od dokładności z jaką wyznaczone jest oświetlenie bezpośrednie w tych obszarach zależy w dużym stopniu jakość całego obrazu.

Jednakże oświetlenie pośrednie, mimo iż o wiele słabsze pod względem mocy, także znacząco wpływa na wygląd renderowanej sceny. To właśnie ono odpowiada za całą gamę zjawisk takich, jak np.: krwawienie kolorów, kaustyki. Dociera ono do większości powierzchni i odpowiada za oświetlenie punktów, które pozostają poza zasięgiem oświetlenia bezpośredniego. Zawiera informację o tym, w jaki sposób światło oddziałuje z modelem po opuszczeniu swojego źródła. Obrazy pozbawione elementów oświetlenia pośredniego, szczególnie ostatniego z wyżej wymienionych, wyglądają sztucznie. Wprawne oko bez większego problemu rozpoznaje, że są utworzone przy wykorzystaniu komputera. Ponadto, bez uwzględnienia tego typu oświetlenia w czasie tworzenia obrazu, nie można zapewnić poprawności fizycznej osiągniętego wyniku. To oświetlenie pośrednie powoduje, że generowane obrazy są w pełni realistyczne.



Oświetlenie bezpośrednie

Oświetlenie pośrednie

Rysunek 1.1.: Porównanie oświetlenia bezpośredniego i pośredniego.

Celem algorytmu wyliczania oświetlenia pośredniego jest więc wyznaczenie w każdym punkcie rozważanej sceny, na podstawie opisu jej geometrii, materiałów z jakich jest zbudowana oraz źródeł światła, jak najdokładniejszego przybliżenia zbioru wszystkich odbitych bądź załamanych promieni świetlnych, które go osiagają. Natomiast celem algorytmu wyznaczania oświetlenia bezpośredniego jest ustalenie w każdym punkcie, z jak największą dokładnością, jaki obszar każdego ze źródeł światła i pod jakim kątem jest widoczny.

Celem oświetlenia globalnego jest pełna symulacja rozchodzenia się światła wewnątrz modelu. Zawiera ono oba opisane wyżej typy oświetlenia. Pozwala na tworzenie całkowicie realistycznych i poprawnych fizycznie obrazów, w których stopień dokładności odwzorowania świata rzeczywistego zależy jedynie od jakości wykorzystanych algorytmów.

1.2. Zastosowania oświetlenia globalnego

Algorytmy generowania obrazów z uwzględnieniem oświetlenia bezpośredniego i pośredniego mają bardzo szerokie spektrum zastosowań. W ogólności można je podzielić na dwie grupy:

- generowanie scen statycznych,
- tworzenie animacji.

Pierwszą grupę stanowi tworzenie realistycznych wizualizacji zadanych modeli. Najczęściej jest ono wykorzystywane w projektowaniu różnego rodzaju obiektów trój-

1.2. Zastosowania oświetlenia globalnego



Lodziarnia



Łazienka

Rysunek 1.2.: Wygenerowane komputerowo obrazy wnętrz.

wymiarowych, np.: w architekturze czy projektowaniu urządzeń użytkowych. Umożliwia uzyskanie w bardzo krótkim czasie dokładnych obrazów projektowanego obiektu bez tworzenia kosztownych modeli, wspomagając w ten sposób i przyspieszając proces projektowania. Należy tutaj podkreślić istotną rolę oświetlenia pośredniego, które może w znacznej mierze zmienić wygląd obiektu. Ponadto istnieje wiele dziedzin, jak np.: reklama, gdzie generowany obraz jest produktem końcowym.

W grupie zastosowań dynamicznych czołową rolę odgrywa tworzenie różnego rodzaju symulatorów, np.: lotu, pozwalających na szkolenie pracowników obsługujących niebezpieczne urządzenia w bezpiecznych, ale jednocześnie bardzo realistycznych warunkach. Symulatory należą jednocześnie do innej ważnej podgrupy, zastosowań czasu rzeczywistego. Obraz jest w nich syntezowany w locie, w zależności od zmieniających się warunków. Większość symulacji odbywa się na otwartym terenie, gdzie znaczącą rolę odgrywa oświetlenie bezpośrednie. Realistyczna grafika komputerowa była także od dawna wykorzystywana w przemyśle filmowym, głównie w czasie tworzenia efektów specjalnych, choć od kilku lat powstaje coraz więcej filmów pełnometrażowych zrealizowanych tylko i wyłącznie przy wykorzystaniu grafiki komputerowej. W tym przypadku mamy do czynienia z grupą zastosowań wsadowych, w których poszczególne klatki filmu renderowane są przy pomocy wcześniej ustalonego modelu.

Z osiągnięć obu grup korzysta także nauka, edukacja czy, na razie w ograniczonym zakresie, twórcy gier komputerowych oraz wiele innych gałęzi przemysłu.

W przypadku obu podgrup z zastosowań dynamicznych, ale i, choć w mniejszym stopniu, w przypadku obrazów statycznych, ważnym aspektem jest automatyczny dobór takich parametrów renderowania, które zapewniają wysoką jakość generowanych obrazów przy zachowaniu maksymalnej wydajności algorytmu.

1.3. Klasy algorytmów

W przypadku oświetlenia bezpośredniego jedynym szeroko stosowanym współcześnie w realistycznej grafice komputerowej rozwiązaniem jest próbkowanie w pewien określony sposób powierzchni wszystkich źródeł światła. Pozwala to ustalić ilość światła, która dociera bezpośrednio do badanego punktu. Najważniejszym elementem metody jest algorytm ustalający w jakiej kolejności i z jaką dokładnością próbkowane będą poszczególne powierzchnie emitujące. Poprawnie skonstruowany algorytm pozwala na znaczne zredukowanie ilości próbek potrzebnych do otrzymania satysfakcjonującego wyniku. Mapy fotonów mogą być zastosowane zarówno do wyeliminowania (częściowego lub całkowitego) konieczności próbkowania, jak i w celu wybrania najlepszej kolejności próbkowania poszczególnych światel.

Algorytmy obliczania oświetlenia pośredniego można podzielić na dwie rozłączne klasy:

- śledzenie promieni (próbkowanie punktowe),
- metody energetyczne (radiosity).

Istnieją także algorytmy hybrydowe łączące obie klasy.

Metody śledzenia promieni śledzą promienie światła przecinające scenę. Są to techniki próbkowania punktowego, co czyni je niezależnymi od geometrii modelu, pozwalając na syntezy obrazów scen zawierających obiekty o dowolnym stopniu komplikacji, między innymi obiekty proceduralne. W podstawowym sformułowaniu ([39]) technika ta śledzi promienie wstecz, od obserwatora w kierunku sceny. Daje ona możliwość prostego zwizualizowania oświetlenia bezpośredniego oraz odbić i załamania o charakterze lustrzanym, pomijając pozostałe promienie świetlne. Uzyskanie pełnej informacji o oświetleniu globalnym wymaga zastosowania technik monte carlo, w których promienie rozchodzą się stochastycznie po całej scenie pokrywając wszystkie ścieżki, którymi porusza się światło. Istnieją także algorytmy śledzące promienie od źródeł światła, oraz jednocześnie od obserwatora i źródeł światła.

Do zalet metod opartych na śledzeniu promieni zalicza się przede wszystkim wspomnianą już niezależność od geometrii sceny oraz, z uwagi na rekursywny charakter obliczeń, niewielkie zapotrzebowanie na pamięć. Niestety, techniki te są bardzo czasochłonne. Olbrzymi wpływ na czas renderingu ma właśnie rekursja, zmuszająca do wielokrotnego obliczania podobnych wartości (element ten można wyeliminować kosztem pamięci i poprawności fizycznej rozwiązania) oraz niska zbieżność rozwiązania. Ponadto obrazy syntezywane za pomocą śledzenia promieni charakteryzują się bardzo nienaturalnym, łatwo dostrzegalnym, a trudnym do wyeliminowania, szumem wysokiej częstotliwości.

W metodach energetycznych wszystkie powierzchnie są traktowane jak źródła światła. Model jest dzielony na skończoną ilość pól emitujących i odbijających światło. Oświetlenie na poszczególnych powierzchniach wyliczane jest poprzez iteracyjne rozwiązywanie układów równań opisujących wymianę energii świetlnej pomiędzy wszyst-

kimi płatami. Podstawowe metody radiosity pozwalają na renderowanie scen zawierających jedynie powierzchnie odbijające światło w sposób rozproszony. Obliczają one całkowite rozwiązanie dla odbicia rozproszonego. Niestety, dodanie odbić i załamania zwierciadlanych wymaga zastosowania dodatkowych technik, jak choćby śledzenia promieni.

Z uwagi na konieczność podziału geometrii modelu na płaty, metody energetyczne mają problemy w przypadku powierzchni krzywoliniowych. Stanowi to ich główną wadę, powodując niedokładności w wynikowym obrazie, w szczególności rozmazywanie ostrych krawędzi. Ponadto są one bardzo czasochłonne oraz wymagają znacznej ilości pamięci, zwłaszcza gdy wymagana jest duża dokładność. We współczesnych algorytmach opartych o metodę energetyczną stosuje się progresywne ulepszanie rozwiązania w celu zredukowania czasu oraz zapotrzebowania na pamięć.

Metody z klasy śledzenia promieni mają wiele zalet w stosunku do technik energetycznych. Są od nich elastyczniejsze, pozwalają na bezpośrednie generowanie o wiele większej gamy zjawisk związanych z transportem światła. Są też prostsze w implementacji oraz łatwo rozszerzalne. Z tych powodów to właśnie śledzenie promieni znajduje się w głównym nurcie badań współczesnej realistycznej grafiki komputerowej.

Mapa fotonów jest jednym z rozszerzeń o jakie w ostatnich latach wzbogacone zostały techniki śledzenia promieni. Umożliwia ona przede wszystkim efektywne obliczanie oświetlenia pośredniego. Ponadto, co także zostało przedstawione w pracy, pozwala na wyliczanie różnego rodzaju statystyk, wykorzystywanych w optymalizacji jakości oraz szybkości generowania obu typów oświetlenia.

1.4. Cele pracy

Praca stanowi przegląd najnowszych i najczęściej stosowanych technik wspomagania renderingu opartych na mapie fotonów. Mimo iż mapy fotonów są wykorzystywane przede wszystkim przy obliczaniu oświetlenia pośredniego, zaprezentowane zostały także metody wykorzystania ich przy określaniu oświetlenia bezpośredniego. Przedstawiono również inne sposoby wykorzystania tej struktury danych oraz szczegóły techniczne związane z problemami pojawiającymi się w przypadku typowych zastosowań.

Głównym celem pracy jest dokładne omówienie wybranych metod od strony teoretycznej oraz zbadanie, jak zachowują się w praktyce. W przypadku części teoretycznej celem jest dogłębne zaprezentowanie i przeanalizowanie stanowiącej bazę metody teorii. Celem części praktycznej jest implementacja pozwalająca na zbadanie wszystkich aspektów przedstawianych technik oraz porównanie rezultatów z wynikami teoretycznymi oraz wynikami prezentowanymi przez autorów prac z rozważanej dziedziny.

Podsumowując, praca ma za zadanie scharakteryzować mapy fotonów, porównać algorytmy je wykorzystujące z rozwiązaniami tradycyjnymi i ocenić ich wartość.

Rozdział 2.

Podstawy teoretyczne

Rozdział ten stanowi krótkie wprowadzenie do teorii związanej z transportem światła. Zebrane i wyjaśnione są w nim wszystkie podstawowe pojęcia niezbędne do zrozumienia materiału zamieszczonego w dalszej części pracy. Wzorom prezentowanym w tym rozdziale często towarzyszą jednostki fizyczne, w których wyrażone są wartości. Jednostki i operacje na nich zapisywane są w nawiasach kwadratowych za właściwymi wzorami.

2.1. Natura światła a grafika komputerowa

Światło generowane jest w naturalnych lub sztucznych źródłach światła, takich jak słońce, ogień czy żarówka. Fizyka zna kilka modeli próbujących wyjaśnić naturę rozchodzenia się światła. Są to:

optyka geometryczna – przedstawia światło jako zbiór promieni przemieszczających się w różnorodnych ośrodkach przezroczystych. Prawdłowo modeluje ona większość właściwości światła, jak choćby odbicie czy załamanie;

optyka falowa – opisuje światło jako fale elektromagnetyczne przy pomocy równań Maxwell-a. Modeluje wszystkie zjawiska możliwe do przedstawienia przy pomocy optyki geometrycznej a także te, które zachodzą, gdy światło oddziałuje z obiektami o wielkości zbliżonej do długości jego fali (dyfrakcję, interferencję, polaryzację i dyspersję);

optyka kwantowa – opisuje w jaki sposób światło oddziałuje z powierzchniami, które napotyka na swojej drodze.

W grafice komputerowej wykorzystywana jest praktycznie wyłącznie optyka geometryczna. Pozostałe modele są zdecydowanie zbyt szczegółowe, więc pomija się zjawiska, które mogą być przy ich pomocy wygenerowane. Interakcje światła z otoczeniem opisywane są przy pomocy modeli odbicia i załamania promieni. Ponadto zazwyczaj przyjmuje się, że światło ma nieskończoną szybkość i z chwilą włączenia oświetlenie

Symbol	Znaczenie
$x, \vec{n}$	punkt w przestrzeni, normalna w tym punkcie
$\vec{\omega}, x \rightarrow y$	kierunek, kierunek z punktu x do y
(θ, ϕ)	kierunek we współrzędnych sferycznych
$\vec{\omega}'$	kierunek padającej radiancji
$\vec{\omega}_o$	kierunek odbicia światła
$\vec{\omega}_s$	kierunek odbicia zwierciadlanego światła
$\vec{\omega}_r$	kierunek załamania zwierciadlanego światła
$L(x, \vec{\omega})$	radiancja w punkcie x i kierunku $\vec{\omega}$
$L(x, \vec{\omega}')$	radiancja padająca na punkt x z kierunku $\vec{\omega}'$
$L(x' \rightarrow x)$	radiancja opuszczająca punkt x' w kierunku x
L_e, L_r, L_i	radiancja emitowana, odbita, padająca
ρ_d, ρ_s	współczynnik odbicia rozproszonego, zwierciadlanego
f_r	funkcja BRDF
f_d	część rozproszona funkcji BRDF
f_s	część zwierciadlana funkcji BRDF
Ω	półsfera
η	współczynnik refrakcji
Θ	kąt między wektorami
$\vec{a} \cdot \vec{b}$	iloczyn skalarny wektorów $\vec{a}$ i $\vec{b}$
ξ	liczba losowa z przedziału $[0, 1]$ (rozkład jednorodny)

Tabela 2.1.: Oznaczenia wykorzystane w pracy.

osiąga od razu stan stabilny. Często zakłada się, że światło podróżuje w próżni. Rzadziej, głównie w celu przedstawienia specjalnych zjawisk, jak dym czy mgła, modeluje się inne ośrodki.

2.2. Oznaczenia

Oznaczenia wykorzystywane w pracy zostały zebrane w tabeli 2.1. Jeśli nie zaznaczono inaczej, wszystkie wektory są skierowane zgodnie z normalną do powierzchni (na zewnątrz).

Symbol	Znaczenie	Operator	Znaczenie
L	źródło światła	$(x)*$	dowolna liczba wystąpień x
E	obserwator (kamera)	$(x)+$	co najmniej jedno wystąpienie x
S	odbicie zwierciadlane	$(x)?$	co najwyżej jedno wystąpienie x
D	odbicie rozproszone	$(x x')$	wystąpienie x lub x'

Tabela 2.2.: Notacja Heckbert-a.

W pracy wykorzystano także wprowadzoną w artykule [9] notację pozwalającą opisać ścieżki pokonywane przez światło. Jej szczegóły podane są w tabeli 2.2. Przykładowo, typowa ścieżka kaustyczna ma przebieg: *źródło światła* $\rightarrow$ *jedno lub wiele odbić zwierciadlanych* $\rightarrow$ *odbicie rozproszone* $\rightarrow$ *obserwator*. W notacji Heckbert-a zbiór takich ścieżek opisuje wyrażenie LS+DE.

2.3. Radiometria

Radiometria opisuje wielkości fizyczne związane z transportem światła. W odróżnieniu od fotometrii nie bierze pod uwagę sposobu, w jaki jest ono postrzegane przez człowieka.

Podstawową jednostką radiometrii jest moc promieniowania świetlnego. Określa ona ilość energii przepływającej przez powierzchnię w jednostce czasu i definiowana jest jako:

$$\Phi = \frac{dQ}{dt} \left[W = \frac{J}{s} \right]. \quad (2.1.)$$

Q jest energią fotonów na wszystkich długościach fali:

$$Q = \int_0^\infty n_\lambda \frac{hc}{\lambda} d\lambda, \quad (2.2.)$$

gdzie:

h - stała Planck-a ($\approx 6.63 \cdot 10^{-34} [J \cdot s]$),

c - prędkość światła ($299,792,458 [\frac{m}{s}]$),

λ - długość fali ($[m]$),

n_λ - ilość fotonów o długości fali λ .

Energia padająca na określoną powierzchnię opisywana jest przez irradiancję (natężenie promieniowania), która jest stosunkiem mocy promieniowania do pola powierzchni, na które ono przypada:

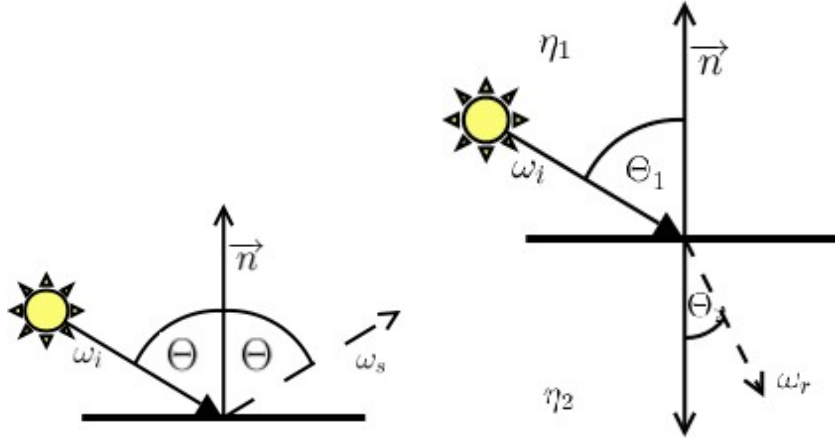
$$E = \frac{d\Phi}{dA} \left[\frac{W}{m^2} \right]. \quad (2.3.)$$

W analogiczny sposób wyraża się energię opuszczającą powierzchnię (promienistość). Intensywność promieniowania jest zależnością pomiędzy kątem bryłowym a strumieniem energii i określa moc promieniowania przypadającą na kąt bryłowy:

$$I = \frac{d\Phi}{d\vec{\omega}} \left[\frac{W}{sr} \right]. \quad (2.4.)$$

Najważniejsza dla grafiki komputerowej jest radiancja, określana jako stosunek mocy promieniowania do kąta bryłowego oraz rzutu powierzchni. Jest to wartość pięciowymiarowa, zależna od pozycji x oraz kierunku $\vec{\omega}$:

$$L(x, \vec{\omega}) = \frac{d^2\Phi}{dAd\vec{\omega} \cos \theta} \left[\frac{W}{m^2 sr} \right]. \quad (2.5.)$$



Rysunek 2.1.: Prawa odbicia i załamania zwierciadlanego.

Największą zaletą radiancji jest fakt, iż reprezentuje ona w bardzo zbliżony sposób kolor obiektu postrzegany przez człowieka jak i urządzenia optyczne. Poza tym w próżni radiancja jest stała wzdłuż linii widzenia. Oznacza to, że radiancja opuszczająca punkt x w kierunku punktu y , jest równa radiancji padającej na punkt y z kierunku x . Fakt ten jest intensywnie wykorzystywany w algorytmach oświetlenia globalnego (zob. 3.1.). Tłumaczy on także, dlaczego kolor i jasność obiektu nie zmieniają się wraz z odległością obserwatora.

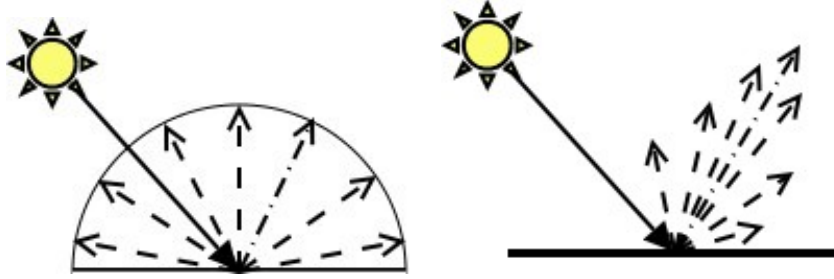
2.4. Odbicie i załamanie światła oraz funkcja BRDF

Światło padające na powierzchnię może ulec odbiciu, załamaniu lub zostać zaabsorbowane. Oczywiście wszystkie trzy wymienione zjawiska mogą zachodzić jednocześnie, każde dla części padającego światła. Zachowanie światła w kontakcie z materią opisywane jest w grafice komputerowej przy pomocy funkcji BRDF (lub ogólniejszych BSSRDF). W kolejnych podrozdziałach wprowadzone są szczegóły dotyczące odbicia i załamania oraz sposobu w jaki są one modelowane przy pomocy funkcji BRDF.

2.4.1. Odbicie

Odbicie światła jest to zmiana kierunku rozchodzenia się promieni świetlnych na granicy dwóch ośrodków. Według prawa odbicia (rys. 2.1.) promień odbity pozostaje w tym samym ośrodku z którego pochodził promień padający. Obydwa leżą na tej samej płaszczyźnie, która jest prostopadła do normalnej do powierzchni w punkcie odbicia. Kąty zawarte między normalną i kierunkami promieni są równe. Na podstawie tego prawa kierunek odbity można wyliczyć w następujący sposób:

$$\vec{\omega}_s = 2(\vec{\omega}' \cdot \vec{n})\vec{n} - \vec{\omega}'. \quad (2.6.)$$



Rysunek 2.2.: Odbicie rozproszone i połyskujące.

Prawo to opisuje jednak tylko odbicie na powierzchniach idealnie gładkich. W przypadku ogólnym odbicie można podzielić na trzy główne rodzaje:

zwierciadlane – zachodzi dla powierzchni doskonale gładkich, których nierówności są bardzo małe w porównaniu z długością padających fal świetlnych. Światło odbijane jest w kierunku wynikającym z prawa odbicia (rys. 2.1.);

rozproszone – występuje w przypadku powierzchni o dużych nierównościach, większych niż długość padających fal świetlnych, nazywanych matowymi. Światło odbijane jest równo we wszystkich możliwych kierunkach (rys. 2.2.);

połyskujące – charakteryzuje powierzchnie pośrednie, w których duża część światła odbijana jest w jednym kierunku (zazwyczaj zgodnym z prawem odbicia), jednak spora jego ilość jest także rozpraszana w innych kierunkach (rys. 2.2.).

Materiały występujące w rzeczywistym świecie mają bardzo często tak skomplikowaną strukturę, że opisanie ich wymaga połączenia wszystkich wymienionych modeli w jeden ogólny model odbicia dla materiału.

2.4.2. Załamanie

Załamane światła (refrakcja) zachodzi na granicy dwóch przezroczystych ośrodków o różnej gęstości i jest wynikiem zmiany szybkości przemieszczania się światła. Podobnie jak w przypadku odbicia, istnieje prawo załamania (rys 2.1.):

$$\eta_1 \sin \Theta_1 = \eta_2 \sin \Theta_2. \quad (2.7.)$$

Na jego podstawie kierunek promienia załamanego można wyliczyć w poniższy sposób:

$$\vec{\omega}_r = -\frac{\eta_1}{\eta_2}(\vec{\omega} - (\vec{\omega} \cdot \vec{n})\vec{n}) - \left(\sqrt{1 - \left(\frac{\eta_1}{\eta_2}\right)^2 (1 - (\vec{\omega} \cdot \vec{n})^2)} \right) \vec{n}. \quad (2.8.)$$

Tak jak w przypadku odbicia rzeczywiste materiały załamują światło w bardziej skomplikowany sposób niż opisuje to przytoczone prawo. Odtworzenie ich zachowania także wymaga tworzenia skomplikowanych modeli załamania światła uwzględniających nierówności powierzchni.

2.4.3. Funkcja BRDF

W celu zamodelowania opisanych w poprzednich podrozdziałach materiałów o skomplikowanej strukturze wykorzystywana jest funkcja BRDF (*ang. Bidirectional Reflectance Distribution Function*). Modeluje ona w najogólniejszy sposób oddziaływanie światła z powierzchnią. Poprzez określenie zależności pomiędzy irradiancją a radiancją odbitą definiuje ona dwukierunkową zdolność odbijania materiału:

$$f_r(x, \vec{\omega}', \vec{\omega}) = \frac{dL_r(x, \vec{\omega})}{L_i(x, \vec{\omega}')(\vec{\omega}' \cdot \vec{n})d\vec{\omega}'} \quad (2.9.)$$

Funkcja BRDF opisuje lokalny model odbicia. Jeśli tylko znana jest irradiancja padająca na dany punkt, możliwe jest wyliczenie przy jej pomocy radiancji opuszczającej go w dowolnym kierunku:

$$L_r(x, \vec{\omega}) = \int_{\Omega} f_r(x, \vec{\omega}', \vec{\omega}) L_i(x, \vec{\omega}') (\vec{\omega}' \cdot \vec{n}) d\vec{\omega}'. \quad (2.10.)$$

Funkcję BRDF można scharakteryzować przy pomocy trzech zasad:

- zakłada ona, że światło uderzające w powierzchnię jest odbijane dokładnie w tym samym miejscu,
- spełnia prawo wzajemności Helmholtz-a, które mówi, że funkcja ta nie zależy od kierunku przepływu światła:

$$f_r(x, \vec{\omega}', \vec{\omega}) = f_r(x, \vec{\omega}, \vec{\omega}'), \quad (2.11.)$$

- spełnia prawo zachowania energii, które mówi, iż powierzchnia może odbijać co najwyżej tyle światła, ile na nią pada:

$$\forall \vec{\omega}. \int_{\Omega} f_r(x, \vec{\omega}', \vec{\omega}) (\vec{\omega}' \cdot \vec{n}) d\vec{\omega}' \leq 1. \quad (2.12.)$$

2.5. Równanie renderingu

Równanie renderingu zostało sformułowane po raz pierwszy w pracy [24]. Opisuje ono światło emitowane z punktu x w kierunku $\vec{\omega}$ w zależności od natężenia światła emitowanego z punktu x w tym kierunku oraz natężenia światła emitowanego do wszystkich innych punktów, które osiąga punkt x i jest od niego odbijane w kierunku $\vec{\omega}$. Równanie to można zapisać w sposób następujący:

$$L_o(x, \vec{\omega}) = L_e(x, \vec{\omega}) + \int_{\Omega} f_r(x, \vec{\omega}', \vec{\omega}) L_i(x, \vec{\omega}') (\vec{\omega}' \cdot \vec{n}) d\vec{\omega}'. \quad (2.13.)$$

Równanie jest poprawne dla dowolnego punktu i kierunku, więc może być wykorzystane do określenia radiancji w dowolnym miejscu renderowanej sceny. Dzięki tej własności jest ono podstawą wszystkich algorytmów oświetlenia globalnego. Niestety, radiancja odbita opisywana jest w nim w sposób rekursywny i nie można jej wyliczyć wprost. Od efektywności z jaką algorytm radzi sobie z rekursją zależy jego szybkość i jakość tworzonych przez niego obrazów.

Rozdział 3.

Śledzenie promieni

W niniejszym rozdziale zawarto krótkie wprowadzenie do wspomnianych już technik śledzenia promieni. Zaprezentowano w nim podstawy tej metody oraz rozszerzenia konieczne w celu uzyskania algorytmu pozwalającego na wizualizację oświetlenia pośredniego oraz bezpośredniego pochodzącego od powierzchniowych źródeł światła. Technikom tym poświęcono osobny rozdział, gdyż stanowią one punkt wyjścia dla większości metod wykorzystujących mapę fotonów.

3.1. Podstawy

Śledzenie promieni jest najogólniejszym znanym aktualnie sposobem generowania realistycznych obrazów. Podstawowy algorytm został zdefiniowany w 1980 roku w pracy [39]. Rozwiązanie zaproponowane w tym artykule pozwalało w prosty sposób zwizualizować oświetlenie bezpośrednie pochodzące od punktowych źródeł światła oraz odbicia i załamania czysto zwierciadlane. Zaprezentowano w nim także prosty, niewiele mający wspólnego z fizyczną poprawnością, sposób symulowania odbicia dla powierzchni błyszczących. Zrezygnowano natomiast całkowicie z odbić o charakterze rozproszonym, które uznano za zbyt kosztowne do wyliczenia przy pomocy tej metody.

Opisany w wymienionym artykule program wykorzystywał pewne istotne właściwości fizyczne światła. Otóż w naturze światło rozchodzi się od swojego źródła w kierunku obiektów znajdujących się w jego zasięgu, następnie załamuje bądź odbija się od nich jeden lub więcej razy, aby w końcu trafić do oka ludzkiego. W rzeczywistych warunkach tylko niewielka ilość emitowanych promieni osiąga oko, przez co symulowanie takiego procesu rozchodzenia się światła wprost byłoby bardzo nieefektywne. Jednakże można wykorzystać dwa istotne fakty:

- radiancja jest stała wzdłuż linii widzenia, czyli odcinka łączącego obiekt i oko, lub dwa obiekty sceny, nie przecinającego żadnego innego obiektu, (zob. 2.3.)¹,

¹Nie jest to prawdą w przypadku niektórych środowisk, w których porusza się światło (np.: dym). Jednak w prostym modelu można tak przyjąć. Informacje o generowaniu obrazów zawierających ośrodki oddziałujące ze światłem można znaleźć w pracy [19].

- rozpraszanie światła (odbicie i załamanie) przez powierzchnie jest w pełni symetryczne.

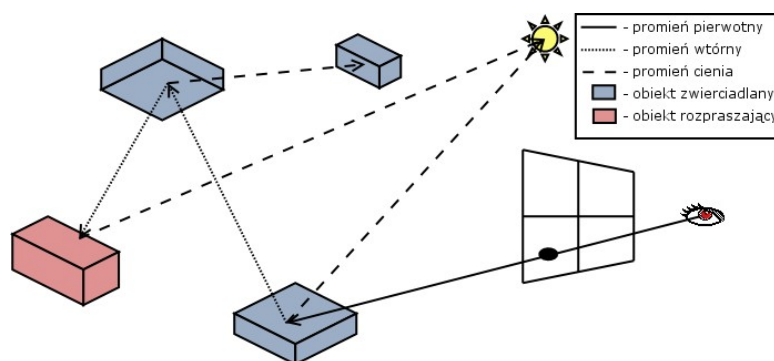
Własności te pozwalają na śledzenie promieni w przeciwną stronę, poczynając od oka, poprzez obiekty wypełniające scenę, na źródłach światła kończąc i otrzymanie w efekcie identycznego wyniku. W ten właśnie sposób działa algorytm zaproponowany w artykule Whitted-a oraz większość współczesnych programów śledzących promienie.

Danymi wejściowymi dla każdego programu śledzącego promienie są:

- opis geometrii wszystkich obiektów sceny wraz z właściwościami fizycznymi materiałów, z jakich są one zbudowane,
- opis wszystkich źródeł światła,
- pozycja obserwatora (kamery),
- płaszczyzna rzutowania, na której tworzony będzie obraz (np.: kierunek patrzenia i kąt widzenia),
- rozdzielczość pionowa i pozioma tworzonego obrazu.

Na podstawie tych danych program oblicza kolor każdego z pikseli. Zadanie to jest realizowane poprzez śledzenie promieni wychodzących z punktu, w którym znajduje się obserwator, w kierunku każdego z pikseli. Promienie te nazywane są pierwotnymi (*ang. primary rays*). W punkcie najbliższego względem obserwatora przecięcia promienia pierwotnego ze sceną, czyli obiekcie widocznym poprzez piksel, obliczana jest radiancja wychodząca. Odpowiada ona kolorowi rozpatrywanego piksela. Oczywiście w celu uzyskania niektórych efektów, np.: antyaliasingu, można wysłać kilka promieni w obszar jednego piksela i uśrednić otrzymaną radiancję.

Do obliczenia wartości radiancji potrzebny jest wektor normalny do powierzchni oraz własności fizyczne materiału w rozpatrywanym punkcie przecięcia. Jest ona obliczana poprzez wyznaczenie wartości irradiancji dla każdego ze źródeł światła sceny, czyli oświetlenia bezpośredniego oraz, w przypadku gdy obiekt odbija lub załamuje światło w sposób zwierciadlany, wartości radiancji przychodzącej z kierunku odbicia i/lub załamania zwierciadlanego. Wszystkie powyższe wartości otrzymuje się poprzez śledzenie kolejnych promieni, nazywanych promieniami wtórnymi (*ang. secondary rays*). W przypadku oświetlenia bezpośredniego jest to promień wysyłany do każdego ze źródeł światła w celu sprawdzenia, czy nie jest ono przesłonięte jakimś innym obiektem sceny. Promienie te nazywane są promieniami cienia (*ang. shadow rays*). Jeśli światło jest widoczne, to jego wkład jest dodawany do radiancji punktu z uwzględnieniem właściwości materiału. Dla materiałów zwierciadlanych generuje się promienie w kierunku idealnie zwierciadlanego odbicia/załamania względem promienia przychodzącego. Następnie rekursywnie śledzi się te promienie i ustala radiancję w punkcie ich najbliższego przecięcia ze sceną. Radiancja ta, jak już wcześniej wspomniano, nie zmienia się wzdłuż linii widzenia, więc może być wykorzystana przy określaniu radiancji w miejscu pierwotnego przecięcia.



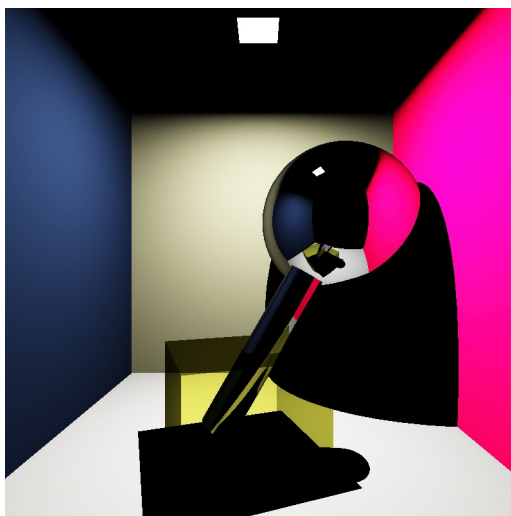
Rysunek 3.1.: Drzewo promieni dla algorytmu śledzenia promieni.

Zaprezentowany algorytm daje możliwość wizualizacji obrazu sceny zawierającej proste źródła światła oraz odbicia i załamania zwierciadlane. Jego działanie rozpoczyna się poprzez wysłanie promienia z punktu obserwacyjnego, a kończy gdy rekursja osiągnie obiekt nie mający właściwości zwierciadlanych, lub gdy promień nie przetnie się z żadnym obiektem sceny. Każdy z promieni generuje po jednym promieniu cienia wysyłanym do każdego ze światel oraz rekursywnie do dwóch kolejnych promieni. Budowane jest w ten sposób (jawnie lub niejawnie) drzewo promieni, w którym jasność każdego węzła jest funkcją jasności jego potomków. Przykład takiego drzewa przedstawiony jest na rysunku 3.1. Widać na nim promień pierwotny wychodzący od obserwatora w kierunku środka piksela zlokalizowanego na płaszczyźnie rzutowania. W każdym punkcie przecięcia wysyłany jest promień cienia w kierunku źródła światła. Należy zwrócić uwagę, że promienie cienia nigdy nie generują dodatkowych promieni, nawet po napotkaniu obiektów zwierciadlanych, co uwidocznione jest u góry schematu. Przecięcia z obiektami zwierciadlanymi generują promienie wtórne.

W opisanym rekursywnym charakterze algorytmu kryje się jego największa słabość. W przypadku naiwnej implementacji wyznaczenie najbliższego punktu przecięcia promienia ze sceną może wymagać przecięcia tego promienia ze wszystkimi obiektami sceny, co dla typowej, zawierającej setki lub tysiące obiektów sceny, jest zadaniem bardzo kosztownym. W celu zwiększenia wydajności konstruuje się struktury danych umożliwiające przeglądanie tylko niewielkiej ilości obiektów znajdujących się w bliskiej okolicy promienia. Przykładami takich struktur są drzewa bsp, kd-drzewa, drzewa ósemkowe. Mimo tych zabiegów obliczenia związane z odnalezieniem najbliższego przecięcia nawet we współczesnych implementacjach pochłaniają około 70% czasu obliczeń.

Przedstawiony algorytm ma bardzo wiele zalet, które czynią go szczególnie atrakcyjnym. Z powodu charakteru obliczeń obsługuje on dowolnie skomplikowane kształty geometryczne, w tym obiekty proceduralne, bez konieczności wykonywania jakiegokolwiek podziału ich powierzchni. Z łatwością oblicza odbicia zwierciadlane, a jakość obrazu może być kontrolowana na poziomie piksela. Jest on także bardzo intuicyjny i prosty w implementacji.

Główną wadą zaprezentowanego modelu jest oczywiście niemożliwość obliczenia



Rysunek 3.2.: Obraz utworzony algorytmem śledzenia promieni Whitted-a.

oświetlenia globalnego. Nie jest on w stanie wyznaczyć oświetlenia pośredniego dla powierzchni innych niż zwierciadlane. Następujące wyrażenie opisuje zestaw wszystkich śledzonych przez niego ścieżek światła:

$$LD \rightarrow S * E$$

Ponadto oświetlenie bezpośrednie jest modelowane jedynie poprzez punktowe źródła światła. Następstwem tego jest brak obszarów półcienia. Wszystkie cienie mają bardzo ostre krawędzie. Oba te fakty powodują, że obrazy, mimo iż dla niewprawnego oka wyglądają dobrze, dalekie są od realistycznych i nie mają szansy poprawnie oddawać interakcji światła z wizualizowaną sceną. Co więcej, wykorzystywany jest tutaj model kamery otworkowej z nieskończenie szybką migawką, w której świat jest widziany przez nieskończenie mały punkt w nieskończenie krótkim czasie. Z tego powodu wszystkie rejestrowane obiekty są tak samo ostre i uchwycone w jednej, nieskończenie krótkiej chwili. Wygenerowanym obrazom brakuje więc efektów związanych z głębią ostrości występujących w przypadku rzeczywistych kamer, a związanych z właściwościami skupiającymi zastosowanych soczewek oraz rozmycia ruchu widocznego w przypadku poruszających się obiektów, a spowodowanego czasem, kiedy migawka pozostaje otwarta. Mimo opisanych niedostatków większość współczesnych programów nadal wykorzystuje ten typ kamery ze względu na jego prostotę i fakt, że powstające przekłamanie nie są tak drażniące i mogą być zlikwidowane przy wykorzystaniu innych metod.

Rysunek 3.2. przedstawia obraz wygenerowany metodą Whitted-a. Widać na nim wyraźnie odbicia i załamania powstałe na powierzchni obiektów zwierciadlanych: kuli, tuby i sześcianu. Odbijające w sposób rozproszony ściany oświetlone są jedynie przez źródło światła znajdujące się na suficie. Z tego powodu obszary, w których światło jest przesłonięte, jak ten za kulą oraz leżący dokładnie w płaszczyźnie źródła światła

sufit, są zupełnie czarne. Wszystkie cienie mają ostre krawędzie, mimo iż nie pochodzą od punktowego źródła światła. Obraz z pewnością nie wygląda realistycznie.

Wszystkie wymienione zjawiska, których nie jest w stanie opisać zaprezentowany przez Whitted-a algorytm, można skutecznie symulować rozszerzając jego konstrukcję o metody monte carlo, o czym mówi następny podrozdział.

3.2. Metody monte carlo

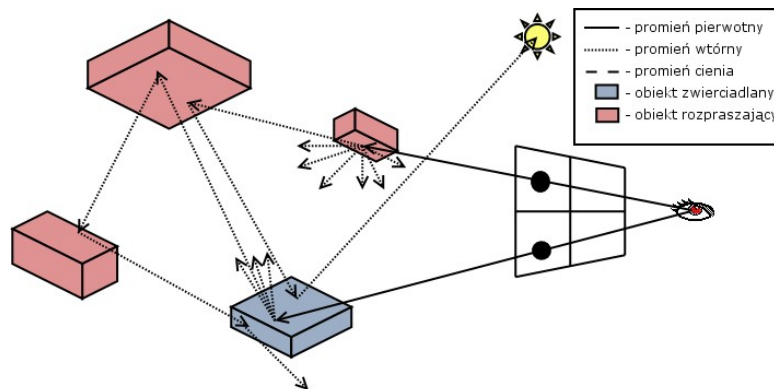
Algorytmy śledzenia promieni oparte na metodach monte carlo są bardzo ogólnymi metodami pozwalającymi na wizualizowanie praktycznie wszystkich istotnych elementów realistycznego obrazu, w tym oświetlenia globalnego. Można je podzielić na trzy klasy:

- śledzenie ścieżek (*ang. path tracing*),
- dwukierunkowe śledzenie ścieżek (*ang. bidirectional path tracing*),
- metody metropolis (*ang. metropolis light transport*).

W tym miejscu opisana jest jedynie pierwsza z metod (śledzenie ścieżek) jako najbardziej podstawowa i stanowiąca rzeczywistą bazę dla dalszej pracy. Informacje na temat metody dwukierunkowej można znaleźć w [27], a metoda metropolis wprowadzona została w [32].

Śledzenie ścieżek jest rozszerzeniem tradycyjnego śledzenia promieni zaproponowanym po raz pierwszy w pracy [24]. Umożliwia ono rozwiązanie zdefiniowanego tam równania renderingu, a tym samym daje możliwość pełnego odtworzenia oświetlenia globalnego. Algorytm wykorzystuje metody stochastyczne do wygenerowania wszystkich możliwych ścieżek światła wewnątrz sceny. W tym celu wykorzystywane są właśnie metody monte carlo. Najogólniej mówiąc pozwalają one za pomocą losowo wybranych próbek badać właściwości nieznanej funkcji. W przypadku śledzenia ścieżek badaną funkcją jest rozkład światła wewnątrz sceny dla jej konkretnego punktu. Jest ona próbkowana poprzez śledzenie wychodzących z tego punktu promieni o losowych kierunkach. Uśrednienie radiancji pochodzących z wielu takich próbek pozwala na oszacowanie rzeczywistej wartości funkcji oświetlenia w badanym punkcie.

W przypadku śledzenia ścieżek oświetlenie globalne w każdym pikselu obrazu jest ustalane poprzez wysłanie bardzo dużej liczby promieni w kierunku jego wnętrza. Natomiast na każdym kolejnym poziomie rekursji jest ono wyznaczane przy użyciu dokładnie jednego promienia odbitego. Jeden promień w obliczeniach rekursywnych może się wydawać wartością zbyt małą, ale jeśli wziąć pod uwagę, że kolor każdego piksela powstaje poprzez uśrednienie dużej liczby ścieżek oraz pamiętając o rekursywnym charakterze algorytmu, rozwiązanie to staje się najbardziej oczywistym. Gdyby każda ścieżka rozgałęziała się na kilkanaście innych, to otrzymany algorytm wraz z głębokością rekursji zwiększałby ich ilość w sposób wykładniczy, przez co niezwykle duży wysiłek obliczeniowy wkładany byłby w dokładne wyznaczanie wartości radiancji w punktach widocznych poprzez wiele odbić, czyli takich, których wpływ na wygląd renderowanego



Rysunek 3.3.: Drzewo promieni dla algorytmu śledzenia ścieżek.

punktu jest mało istotny w porównaniu z tymi widocznymi bezpośrednio. Wybór typu generowanego promienia (odbity/załamany, zwierciadlany/rozproszony) dokonywany jest na podstawie właściwości materiału przy pomocy techniki zwanej rosyjską ruletką. Co ważne technika ta pozwala także śledzić ścieżki o skończonej długości i uzyskać rezultat taki sam, jak przy śledzeniu ścieżek nieskończonych. Więcej szczegółów na ten temat można znaleźć w paragrafie 4.2.2.

Oświetlenie bezpośrednie może być wyznaczane wprost, przy pomocy techniki opisanej w poprzednim akapicie. W tym wypadku wkład światła wliczany jest, gdy któraś ze ścieżek trafi w jedno z nich. Innym, lepszym sposobem jest wyliczanie oświetlenia bezpośredniego przy pomocy specjalnie skonstruowanej metody monte carlo pozwalającej próbkować tylko i wyłącznie powierzchnie źródeł światła. Ten sposób wykorzystywany jest głównie z powodu bardzo istotnego wpływu, jaki ma oświetlenie bezpośrednie na kolor punktu. W tym przypadku ścieżki trafiające w źródła światła są pomijane jako nieprawidłowe w celu uniknięcia sytuacji, w której oświetlenie bezpośrednie wliczane jest dwukrotnie. Obie metody pozwalają na wizualizowanie powierzchniowych źródeł światła i symulowanie obszarów półcienia, jednak druga z nich daje zdecydowanie lepsze rezultaty i pozwala na zastosowanie pewnych optymalizacji, jak schemat przedstawiony w [37], czy te prezentowane w rozdziale 5.

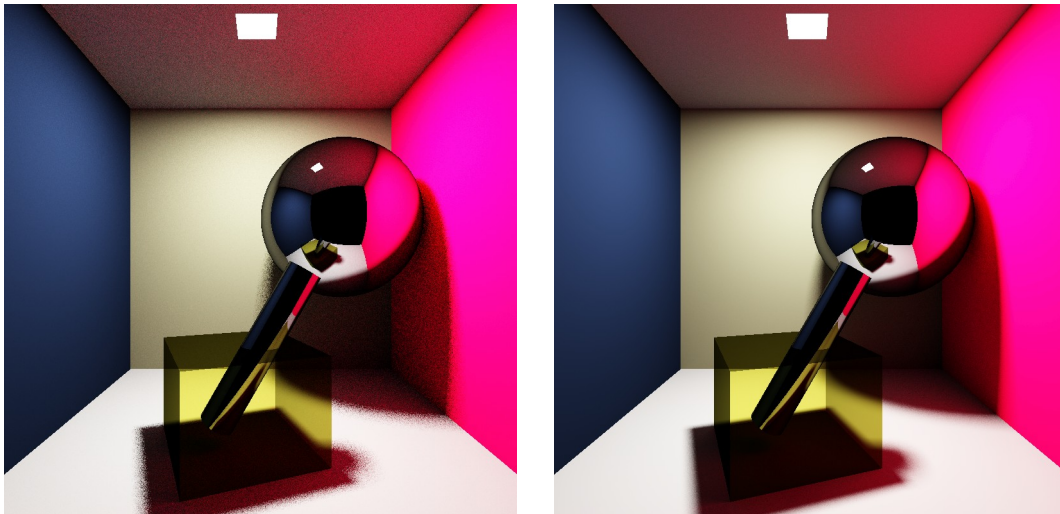
Rysunek 3.3. przedstawia przykładowe drzewo promieni wygenerowane przez algorytm śledzenia ścieżek. W tym przykładzie poprowadzono dwa promienie pierwotne dla dwóch różnych pikseli. W punkcie pierwszego, widocznego poprzez piksel przecięcia ścieżka rozgałęzia się w wielu kierunkach. W każdym następnym przecięciu generowany jest tylko jeden promień wtórny. Dla każdego z promieni pierwotnych przedłużono tylko jedną ścieżkę, pozostałe pokazane są symbolicznie. Warto zwrócić uwagę na różnicę w rozkładzie ścieżek pomiędzy materiałem błyszczącym, dla którego są one skoncentrowane wokół kierunku odbicia zwierciadlanego, a materiałem rozpraszającym, dla którego ich rozkład jest jednorodny. Ponadto należy zauważyć, że obiekty rozpraszające znajdujące się u góry sceny, będące tyłem do źródła światła, nie byłyby w ogóle oświetlone, gdyby użyto prostego algorytmu śledzenia promieni. Zastosowanie metody monte carlo

umożliwia odtworzenie oświetlenia pośredniego na tych powierzchniach. W prezentowanym przykładzie oświetlenie bezpośrednie nie jest wyliczane osobno. Z tego powodu ścieżka trafiająca w źródło światła nie została odrzucona.

Śledzenie ścieżek jest metodą bardzo ogólną. Do głównych zalet należy możliwość wykorzystania dowolnej funkcji BRDF, niskie zapotrzebowanie na pamięć oraz złożoność czasowa, która na podstawie badań empirycznych została określona na $O(\log(N))$, gdzie N oznacza ilość obiektów sceny. Połączenie metod monte carlo z rosyjską ruletką daje dla funkcji oświetlenia estymator nieobciążony. Gwarantuje to, że generowany obraz nie będzie zawierał żadnych dodatkowych artefaktów poza tymi, które powstają na skutek niedokładności przybliżenia funkcji oświetlenia. Metoda ta dzięki swojej prostocie jest także świetnym punktem wyjścia dla algorytmów bardziej wyspecjalizowanych.

Największą wadą śledzenia promieni opartego na metodach monte carlo jest wariancja wśród oszacowań. Ujawnia się ona w postaci szumu wysokiej częstotliwości widocznego w wygenerowanym obrazie. Jest ona rezultatem użycia zbyt małej ilości ścieżek przy przybliżaniu wartości funkcji oświetlenia. Ilość szumu zależy w dużej mierze od charakteru zarówno bezpośredniego, jak i pochodzącego z odbić, oświetlenia sceny. W przypadku scen oświetlonych bardzo jednorodnie, np.: przez światło dużej mocy i powierzchni, jak słońce, oraz nie zawierających dużej ilości obiektów generujących silne zjawiska świetlne, wystarczy około 100-500 ścieżek na piksel w celu uzyskania obrazu o bardzo niskim poziomie szumu. Jednak w scenach charakteryzujących się dużymi i gwałtownymi zmianami w natężeniu oświetlenia, np.: zawierających dużo obiektów połyskliwych, generujących kaustyki, zredukowanie szumu może wymagać użycia 10000 lub nawet 100000 ścieżek na piksel. Wysoki koszt wyeliminowania szumu wynika z faktu, że dla metod monte carlo błąd standardowy dla N próbek jest rzędu $1/\sqrt{N}$. Oznacza to, że aby zmniejszyć go o połowę, trzeba użyć aż czterokrotnie więcej próbek. Warto zaznaczyć, że ilość szumu nie zależy w żaden sposób od poziomu skomplikowania geometrii.

Istnieje wiele technik pozwalających na zredukowanie wariancji, czyli także szumu, w metodach monte carlo. W tym celu wykorzystuje się przede wszystkim zmodyfikowane techniki próbkowania wartości funkcji. Jedną z podstawowych jest koncentrowanie większej ilości próbek w obszarach, w których ma ona szansę przyjąć największe wartości. Wykonuje się to poprzez próbkowanie zgodne z rozkładem funkcji BRDF. Rozwiązanie to sprawdza się bardzo dobrze dla powierzchni błyszczących, dla których funkcja BRDF przyjmuje bardzo duże wartości w niewielkim obszarze. Niestety, wyznaczenie odpowiedniego rozkładu dla większości funkcji BRDF jest bardzo trudne. Ponadto w przypadku odbicia rozproszonego, dla którego wartość BRDF jest jednakoowa w każdym kierunku, nie daje żadnej poprawy. W tym przypadku jednakże można skorzystać z częściowej wiedzy na temat układu radiancji wewnątrz sceny i skierować większą ilość promieni w obszary, gdzie jej wartość jest wysoka. Potrzebną wiedzę na temat radiancji można zaczerpnąć z odpowiednio skonstruowanej mapy fotonów (zob. 6.2.). Innym prostym sposobem zredukowania szumu jest próbkowanie warstwowe (*ang. stratified sampling*). Obszar próbkowania dzielony jest poprzez odpowiednio skon-



100 ścieżek/piksel ośw. pośrednie 1000 ścieżek/piksel ośw. pośrednie
 10 promieni/piksel ośw. bezpośrednie 300 promieni/piksel ośw. bezpośrednie

Rysunek 3.4.: Obrazy wygenerowane metodą śledzenia ścieżek.

struowaną siatkę, a następnie umieszcza się po jednej próbce wewnątrz każdego oczka siatki. Taki sposób próbkowania zapewnia dobry rozkład próbek w badanej przestrzeni i przy odpowiedniej konstrukcji siatki poprawia zbieżność nawet do $1/N$. Niestety, zastosowanie tego typu próbkowania wymaga znajomości ilości próbek przed jego rozpoczęciem, co nie jest możliwe gdy wykorzystywana jest rosyjska ruletka.

Wszystkie opisane dotąd techniki redukcji szumu próbują go wyeliminować poprzez odpowiednie ukierunkowanie ścieżek, pozwalające z wystarczającą dokładnością uchwycić wszystkie szczegóły oświetlenia. Innym, o wiele mniej kosztownym czasowo rozwiązaniem jest zastosowanie technik obróbki gotowego obrazu. Jedną z nich są filtry redukujące wariancję. Niestety, konstrukcja dobrego filtra nie jest łatwa, natomiast proste rozwiązania mają tendencję do uszkodzania obrazu, np.: poprzez rozmywanie ostrych krawędzi. Ponadto zadaniem filtrów jest ukrycie problemów związanych z niedokładnością przybliżenia funkcji oświetlenia, nie wyeliminowanie ich, dlatego powinno się je stosować ostrożnie.

Rysunek 3.4. przedstawia dwa obrazy utworzone algorytmem śledzenia ścieżek. Przy ich generowaniu wykorzystano próbkowanie zgodne z rozkładem funkcji BRDF materiałów. Szum, wyraźnie widoczny w przypadku 100 ścieżek, znika prawie całkowicie po dziesięciokrotnym zwiększeniu ich liczby. Na obu obrazach widoczne są elementy oświetlenia globalnego. Sufit i obszary zacienione nie są całkowicie czarne. Na suficie widać, jak kolory otaczających ścian wpływają na jego barwę. Oświetlenie bezpośrednie utworzono przy pomocy osobnej techniki monte carlo. Cienie mają miękkie krawędzie, wyraźnie zarysowane są obszary półcienia.

W podstawowym sformułowaniu algorytm śledzenia ścieżek nie wykorzystuje praktycznie żadnych dodatkowych informacji możliwych do wyczytania ze sceny. Jest to oczywiście jego dużą wadą, ale nie stoi na przeszkodzie, aby wykorzystać te dane, ewentualnie tworząc dodatkowe struktury danych. Przykłady takich rozwiązań można znaleźć w artykułach Ward-a ([38], [37]). Wiele innych, opierających się na mapie fotonów, jest przedstawionych w dalszej części tej pracy.

Zwięzły przegląd metod monte carlo wykorzystywanych w śledzeniu promieni, a także schematów pozwalających na oddzielne wyliczanie oświetlenia bezpośredniego i pośredniego, dostępny jest w pracy [6]. Dokładny opis można znaleźć w pracy [31].

Rozdział 4.

Mapa fotonów

Rozdział ten wprowadza mapę fotonów jako strukturę danych. Prezentuje elementy z jakich się ona składa oraz szczegóły podstawowych operacji, na których opierają się wszystkie powiązane z nią algorytmy. Przedstawia między innymi sposób konstruowania mapy, optymalizowanie jej struktury w celu osiągnięcia maksymalnej wydajności oraz odzyskiwanie zapisanych w niej informacji.

4.1. Wprowadzenie

Mapa fotonów jest strukturą danych składającą się z fotonów wyemitowanych i zachowanych w scenie. Dwoma głównymi operacjami powiązanymi z nią są:

- konstruowanie mapy (emisja i śledzenie fotonów),
- wyszukiwanie zbioru fotonów najbliższych zadanemu punktowi.

Na bazie tych dwóch podstawowych funkcjonalności zbudowane są wszystkie algorytmy wykorzystujące tą strukturę. W podrozdziałach 4.2. i 4.3. zawarte są informacje dotyczące fazy konstrukcji mapy. Pierwszy z nich prezentuje w sposób abstrakcyjny proces emisji fotonów, a drugi skupia się na detalach implementacyjnych i optymalizacji mapy. Szczegóły dotyczące efektywnego wyszukiwania najbliższych fotonów opisano w podrozdziale 4.4. W podrozdziale 4.5. przedstawiono przykładową mapę fotonów.

4.2. Konstrukcja mapy fotonów

Mapa fotonów konstruowana jest w dwóch fazach. Pierwszą z nich jest śledzenie fotonów. W tym czasie fotony są emitowane ze źródeł światła, śledzone poprzez scenę oraz zapisywane w tymczasowej strukturze danych. Głównym celem tej fazy jest jak najszybsze uzyskanie dobrego rozkładu fotonów, przy wykorzystaniu możliwe niewielkiej ich liczby. W czasie śledzenia nie są wykonywane żadne odwołania do tworzonej mapy, dlatego w celu zwiększenia wydajności fotony zapisywane są w pamięci w sposób liniowy, jeden za drugim. Osobna faza, następująca po wyemitowaniu odpowiedniej

liczby fotonów, optymalizuje ich układ w mapie tak, aby zminimalizować czas potrzebny na wykonanie operacji wyszukiwania. Opisany schemat konstrukcji implikuje fakt, iż po zakończeniu fazy optymalizacji mapa fotonów staje się strukturą wykorzystywaną jedynie do odczytu danych. Dodanie nowych fotonów byłoby możliwe, jednak wymagałoby wykonania ponownie kosztownej operacji optymalizacji układu fotonów. Nie jest to poważnym ograniczeniem, gdyż większość z zastosowań nie wymaga dodawania nowych fotonów.

4.2.1. Emisja fotonów

Fotony emitowane są przez źródła światła. Metoda nie wprowadza żadnych ograniczeń na ich postać, więc mogą być to zarówno typowe stosowane w grafice komputerowej światła, jak punktowe czy kierunkowe o jednorodnym rozkładzie emisji, jak i światła pochodzące z prawdziwego świata o dowolnej geometrii i rozkładzie. Każde źródło światła emituje odpowiednio dużą liczbę fotonów, które następnie są śledzone poprzez scenę. Moc światła jest dzielona równo pomiędzy wszystkie wyemitowane przez nie fotony. Warto podkreślić, że liczba fotonów zachowanych w mapie może być różna od ilości wyemitowanych, ale ich moc zawsze zależy jedynie od drugiej z przytoczonych wartości.

Pojedyncze źródło światła

W przypadku scen zawierających pojedyncze źródło światła fotony są emitowane tylko i wyłącznie z jego powierzchni. Powinny być one emitowane z rozkładem (punkt emisji oraz kierunek wyjścia) zgodnym z rozkładem emisji światła. Zabieg ten zapewnia równą moc wszystkich opuszczających źródło światła fotonów, gdyż moc emitowana z danego punktu w określonym kierunku jest normalizowana poprzez dobranie właściwego rozkładu. Pozwala to oszczędzać zasoby (czas i pamięć) oraz poprawia wyniki osiągnięte w czasie renderingu.

Jak już wspomniano, moc światła powinna być rozdzielona proporcjonalnie pomiędzy wszystkie fotony pochodzące od niego. W przypadku wykorzystania odpowiedniego rozkładu w czasie emisji wystarczy podzielić moc każdego zachowanego w mapie fotonu przez ilość wyemitowanych fotonów.

Wiele źródeł światła

Jeśli model zawiera wiele źródeł światła, fotony muszą być wyemitowane z nich wszystkich. Ilość fotonów emitowanych przez poszczególne źródła światła powinna być dobrana wprost proporcjonalnie do ich mocy. Im większa moc światła, tym więcej fotonów będzie z niego emitowane. Zabieg ten pozwala utrzymać moc wszystkich zapisanych w mapie fotonów na zbliżonym poziomie, co ma znaczący wpływ na jakość renderowanych obrazów. Algorytm emisji dla konkretnych światel jest taki sam jak w przypadku pojedynczego światła.

Punktowe źródło światła

Jest to najprostsze z możliwych źródeł światła. Praktycznie nie jest ono spotykane w świecie rzeczywistym. Powierzchnia emitująca składa się z dokładnie jednego punktu w przestrzeni, a światło rozchodzi się jednorodnie we wszystkich kierunkach. Emisja fotonów powinna zawsze odzwierciedlać charakterystykę źródła światła, toteż wszystkie fotony emitowane są dokładnie z punktu w którym znajduje się źródło światła w kierunku losowanym z rozkładem jednorodnym po jednostkowej sferze otaczającej je. Algorytm losowania kierunku można zapisać w następujący sposób:

$$(\theta, \phi) = (\arccos(1 - 2\xi_1), 2\pi\xi_2). \quad (4.1.)$$

Kierunkowe źródło światła

Światła kierunkowe nie występują w świecie rzeczywistym. Pomimo to są one często wykorzystywane w grafice komputerowej w celu symulowania bardzo odległych światel o dużej mocy, jak na przykład słońce. W tym przypadku fotony wysyłane są zawsze w jednym kierunku (stąd nazwa światło kierunkowe) z odpowiednio wybranego obszaru świecącej płaszczyzny. Obszar ten może być ustalony a priori, lub wybrany poprzez rzutowanie na płaszczyznę figury (np.: sześcianu) otaczającej całą scenę. Płaszczyzna rzutowania powinna znajdować się poza wnętrzem figury otaczającej.

Sferyczne źródło światła

Sferyczne źródło światła jest bardzo powszechnie stosowane w grafice komputerowej. Definiuje się je jako sferę, której każdy punkt emituje światło we wszystkich kierunkach zgodnych z normalną do powierzchni w tym punkcie. Fotony powinny być emitowane z wybieranych z rozkładem jednorodnym punktów znajdujących się na sferze. Dla sfery o środku w punkcie (c_x, c_y, c_z) i promieniu R punkty można generować przy pomocy następującego algorytmu:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} c_x + 2R \cos(2\pi\xi_1) \sqrt{\xi_2(1 - \xi_2)} \\ c_y + 2R \sin(2\pi\xi_1) \sqrt{\xi_2(1 - \xi_2)} \\ c_z + R(1 - 2\xi_2) \end{bmatrix}. \quad (4.2.)$$

W celu uwzględnienia powierzchni rzutowej źródła światła w punkcie padania, kierunek emisji fotonu powinien być wybierany z rozkładem proporcjonalnym do cosinusa kąta między normalną do powierzchni w punkcie emisji, a kierunkiem wyjścia. Otrzymać go można z wzoru:

$$(\theta, \phi) = (\arccos(\sqrt{\xi_1}), 2\pi\xi_2). \quad (4.3.)$$

Kwadratowe i trójkątne źródło światła

Światła kwadratowe i w mniejszym stopniu trójkątne są także bardzo często wykorzystywane w modelach scen. Kierunek fotonu wybierany jest przy pomocy tego samego wzoru, co w przypadku sfery. Losowy punkt wewnątrz trójkąta o wierzchołkach

(x_0, x_1, x_2) można uzyskać przy pomocy wzoru:

$$x = \alpha x_0 + \beta x_2 + \gamma x_3, \quad (4.4.)$$

gdzie:

$$\begin{aligned} \alpha &= 1 - \sqrt{\xi_1}, \\ \beta &= (1 - \xi_2)\sqrt{\xi_1}, \\ \gamma &= \xi_2\sqrt{\xi_1}. \end{aligned}$$

Losowy punkt wewnątrz kwadratu opisanego przez wierzchołek x_0 i wektory $\vec{v}_1, \vec{v}_2$ w kierunku wierzchołków sąsiadujących wyraża się wzorem:

$$x = x_0 + \vec{v}_1 \xi_1 + \vec{v}_2 \xi_2. \quad (4.5.)$$

4.2.2. Śledzenie fotonów

Po wyemitowaniu foton jest śledzony przez scenę. Odbywa się to w sposób zbliżony do tradycyjnego śledzenia promieni. Za każdym razem, gdy foton uderza w jakąś powierzchnię, może zostać odbity, załamany albo zaabsorbowany. Decyzja o rodzaju interakcji podejmowana jest przy pomocy rosyjskiej ruletki na podstawie właściwości powierzchni. W przypadku odbicia i załamania kierunek dalszej wędrówki fotonu ustalany jest w ten sam sposób, co w przypadku metod śledzenia promieni, czyli na podstawie prawa odbicia zwierciadlanego lub rozproszonego, załamania światła, lub próbując funkcję BRDF materiału.

Rosyjska ruletka

Rosyjska ruletka jest standardową techniką monte carlo wprowadzoną do grafiki komputerowej w pracy [1]. Korzystając z odpowiedniego rozkładu prawdopodobieństwa pozwala ona na usunięcie mało znaczących obszarów dziedziny problemu. Mając prawdopodobieństwo p , wyznaczenia następnej próbki L_n określamy:

$$L_n = \begin{cases} \frac{L}{p} & \text{dla } \xi < p \\ 0 & \text{w p.p.} \end{cases}, \quad (4.6.)$$

gdzie L jest wartością wyznaczonej próbki. W omawianym przypadku p będzie prawdopodobieństwem śledzenia kolejnego promienia, a L wartością radiancji wyznaczoną przy jego pomocy. Wartość oczekiwana estymatora wyraża się wzorem:

$$E\{L\} = (1 - p)0 + p \frac{E\{L\}}{p} = E\{L\}. \quad (4.7.)$$

Estymator jest więc nieobciążony, co jest bardzo istotne, gdyż pozwala śledzić promienie (fotony) poprzez skończoną ilość odbić i otrzymać rezultat równoważny z tym, który byłby osiągnięty przy nieskończonej długości ścieżki.

Zastosowanie w śledzeniu fotonów

W przypadku śledzenia fotonów rosyjska ruletka pozwala wcześniej usunąć mało znaczące fotony. Dzięki temu moc fotonów zachowanych w mapie jest bardzo wyrównana. Decyzja na temat dalszych losów fotonu podejmowana jest przy każdej interakcji z napotkaną na jego drodze powierzchnią, na podstawie właściwości materiału, z jakiego jest ona zbudowana. W przypadku obrazów monochromatycznych zależy ona od wartości liczby losowej ξ w następujący sposób:

$$\begin{aligned} \xi \in [0, \rho_d] & \quad - \text{odbicie rozproszone,} \\ \xi \in (\rho_d, \rho_d + \rho_s] & \quad - \text{odbicie zwierciadlane,} \\ \xi \in (\rho_d + \rho_s, 1] & \quad - \text{absorpcja.}^1 \end{aligned}$$

Moc odbitego fotonu nie jest w żaden dodatkowy sposób regulowana, gdyż poprawność wynika z rosyjskiej ruletki oraz dużej liczby fotonów. Dla przykładu od powierzchni odbijającej 40% światła w wyniku rosyjskiej ruletki odbitych zostanie jedynie 40% fotonów, ale o tej samej mocy, zamiast wszystkich o mocy pomniejszonej o 60%.

W przypadku obrazów kolorowych (RGB) decyzja może być podjęta przy pomocy tej samej reguły co poprzednio, a wartości prawdopodobieństwa zajścia określonych zdarzeń można wyliczyć na jeden z dwóch sposobów:

$p_d = \frac{\rho_{d,r} + \rho_{d,g} + \rho_{d,b}}{3}$	$p_d = \frac{\max(\rho_{d,r}P_r + \rho_{d,g}P_g + \rho_{d,b}P_b)}{\max(P_r, P_g, P_b)}$
$p_s = \frac{\rho_{s,r} + \rho_{s,g} + \rho_{s,b}}{3}$	$p_s = \frac{\max(\rho_{s,r}P_r + \rho_{s,g}P_g + \rho_{s,b}P_b)}{\max(P_r, P_g, P_b)}$

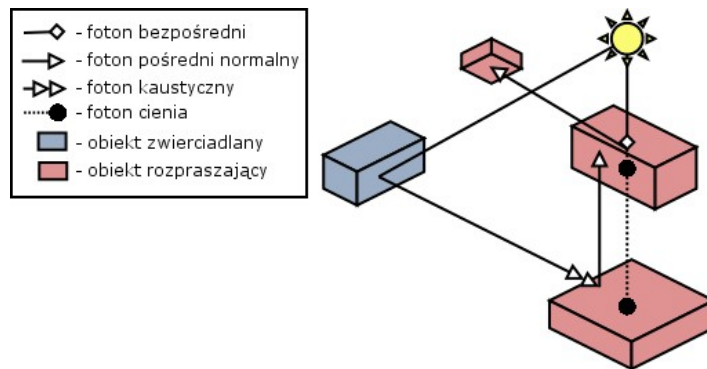
p_d jest tutaj prawdopodobieństwem odbicia rozproszonego, a p_s zwierciadlanego. Prawdopodobieństwo absorpcji jest dopełnieniem do 1 wartości pozostałych prawdopodobieństw. $\rho_{(r|g|b)}$ oznacza odbijalność materiału w zakresie barwy czerwonej, zielonej i niebieskiej, a $P_{(r|g|b)}$ moc fotonu dla tej barwy. W obu przypadkach moc odbitego fotonu musi być przeskalowana w następujący sposób (przykład dla odbicia rozproszonego):

$$\begin{aligned} P_{r,r} &= P_{i,r} \rho_{d,r} / p_d \\ P_{r,g} &= P_{i,g} \rho_{d,g} / p_d \\ P_{r,b} &= P_{i,b} \rho_{d,b} / p_d \end{aligned}$$

4.2.3. Zachowywanie fotonów

Fotony zachowywane są zawsze wtedy, kiedy trafiają w obiekt odbijający bądź załamujący światło w sposób rozproszony. Każdy wyemitowany foton może być zachowany wiele razy, za każdym razem, gdy trafia w odpowiedni obiekt oraz w miejscu, gdzie jest absorbowany. Fotonów nie zachowuje się na powierzchniach zwierciadlanych, gdyż prawdopodobieństwo odnalezienia fotonu przychodzącego z kierunku zbliżonego do kierunku odbicia zwierciadlanego dla danego promienia, a co za tym idzie mającego istotny wpływ na światło odbite od powierzchni w tym kierunku, jest bardzo małe.

¹Oczywiście nic nie stoi na przeszkodzie w dołączeniu załamania światła.



Rysunek 4.1.: Rodzaje fotonów.

Z tego powodu odbicia i załamania zwierciadlane nie mogą być efektywnie symulowane przy pomocy map fotonów i w celu ich uzyskania trzeba zastosować inną metodę².

Fotony można podzielić na kilka rodzajów:

oświetlenia bezpośredniego – fotony mające pierwszy po opuszczeniu źródła światła kontakt z obiektem sceny,

oświetlenia pośredniego – wszystkie fotony odbite przynajmniej raz od obiektu sceny po opuszczeniu źródła światła, wyróżnia się tu dwa rodzaje:

kaustyczne – fotony o ścieżkach LS^*D , odpowiadające za generowanie kaustyk,

normalne – wszystkie pozostałe fotony,

cienia – używane jedynie w celu przybliżenia oświetlenia bezpośredniego.

W zależności od sposobu wykorzystania tworzonej mapy można zachowywać w niej tylko fotony wybranego rodzaju, lub rozdzielać fotony różnego typu do osobnych map w celu zmniejszenia ich wielkości i poprawienia wydajności.

Rysunek 4.1. ilustruje rodzaje fotonów. Strzałkami i kropkami zaznaczono na nim miejsca, w których fotony są zachowywane. Foton nie jest zachowywany w przypadku kontaktu z materiałem zwierciadlanym, a następna interakcja ze sceną daje foton kaustyczny. Fotony cienia generowane są we wszystkich poza pierwszym punktach przecięcia obiektów sceny z promieniem zgodnym z kierunkiem wyjścia fotonu ze źródła światła.

4.3. Struktura danych

Struktura danych, w której przechowywana jest mapa, musi spełniać dwa kryteria: zajmować jak najmniej miejsca oraz działać z maksymalną możliwą szybkością zarówno

²Fakt ten nie stanowi żadnego problemu, gdyż ten typ odbić/załamania może być efektywnie renderowany przy pomocy zwykłego śledzenia promieni.

w czasie emisji, jak i w czasie wykorzystania mapy. Ponadto w celu uniezależnienia od geometrii sceny, fotony nie powinny być przechowywane w strukturach służących do jej opisu.

Skonstruowanie struktury szybkiej zarówno w czasie emisji fotonów, kiedy są one po prostu dodawane do mapy jeden po drugim z losowym rozkładem, jak i w czasie renderingu, kiedy zbiory najbliższych fotonów wyszukiwane są w mapie, byłoby kłopotliwe. Z tego powodu jest ona przechowywana w różnych postaciach w każdej z faz. Jak już wspomniano, w czasie emisji fotony zapisywane są w tablicy, której narzut czasowy i pamięciowy jest równy zeru. Spełnia ona również kryterium niezależności od geometrii modelu.

Przed przystąpieniem do fazy renderingu tablica musi zostać przebudowana w strukturę umożliwiającą szybkie wyszukiwanie zbioru elementów w trójwymiarowej przestrzeni o bardzo niejednorodnym rozkładzie danych. Do zrealizowania tego zadania świetnie nadaje się kd-drzewo ([2]). Jest to wielowymiarowe drzewo poszukiwań binarnych, w którym każdy węzeł dzieli przestrzeń wzdłuż jednego z wymiarów. Mapa fotonów jest trójwymiarowym kd-drzewem. W każdym węźle przechowywany jest foton. W węzłach nie będących liśćmi znajdują się płaszczyzny podziału. Są one zawsze równoległe do jednej z trzech płaszczyzn głównych układu współrzędnych i zawierają punkt, w którym znajduje się foton należący do węzła. Wszystkie fotony z lewego poddrzewa znajdują się poniżej, a z prawego powyżej tej płaszczyzny.

Zbalansowane kd-drzewo umożliwia wyszukanie 1 fotonu w czasie $O(\log(N))$. Wyszukanie k najbliższych fotonów zajmuje średnio $O(k + \log(N))$. Odpowiednio skonstruowana struktura spełnia więc wymóg szybkości. Następny paragraf prezentuje szczegóły dotyczące balansowania kd-drzewa oraz minimalizacji jego reprezentacji.

4.3.1. Optymalizacja mapy

Punktem wyjścia dla algorytmu optymalizującego jest tablica z zachowanymi fotonami. Balansowanie kd-drzewa jest podobne do balansowania drzewa binarnego. Jednak z powodu wielowymiarowości danych na każdym poziomie należy wybrać płaszczyznę podziału. W przypadku mapy fotonów jest ona wybierana na podstawie maksymalnej odległości między punktami (lub jej wariancji) we wszystkich podstawowych kierunkach. Po wybraniu płaszczyzny podziału odnajdywana jest mediana punktów (fotonów) w tym kierunku, która zostaje korzeniem balansowanego zbioru punktów. Lewe i prawe poddrzewo konstruowane są z powstałych podzbiorów.

Zbalansowane drzewo przechowywane jest w pamięci przy pomocy struktury kopcopodobnej, której elementy nadal znajdują się w tablicy. Korzeń jest pierwszym elementem, a element o indeksie k ma następników w komórkach $2k$ (lewy) i $2k + 1$ (prawy). Dzięki temu układowi danych nie trzeba stosować wskaźników, co zdecydowanie redukuje ilość potrzebnej pamięci. Ponadto algorytm zapewnia, że w tablicy nie będzie pustych liści, przez co cała zbalansowana mapa na rozmiar równy wielkości zachowanych w niej fotonów.

Złożoność algorytmu balansowania wynosi $O(N \log N)$. W praktyce wykonanie go na-

wet na bardzo dużej, zawierającej miliony fotonów mapie, zabiera zaledwie kilkadziesiąt sekund. W celu zwiększenia wydajności procedury warto skonstruować dodatkową (tymczasową) tablicę wskaźników do fotonów i w czasie balansowania operować tylko na nich, zmniejszając ilość kopiowanych danych.

4.3.2. Opis fotonu

Jak już wspomniano, fotony zapisywane są w mapie za każdym razem, gdy wyemitowany i śledzony foton trafi w powierzchnię rozpraszającą światło. Z tego względu można powiedzieć, że w mapie przechowywane są nie fotony, lecz informacje o ich interakcjach ze sceną. Wiele z zastosowań wykorzystuje mapy zawierające bardzo dużą liczbę fotonów, dlatego muszą one mieć jak najmniejszy rozmiar. Ponadto w każdej z faz fotony są przechowywane w tablicy, dlatego z powodów wydajnościowych ich rozmiar powinien być równy wielokrotności słowa maszynowego. Do obliczeń potrzebna jest przede wszystkim moc fotonu³ oraz kierunek z którego przybył, używany między innymi do sprawdzenia, czy foton pochodzi z właściwej strony powierzchni.

Na podstawie powyższych obserwacji, przyjmując wykorzystanie maszyny o 32 bitowym rozmiarze słowa oraz przestrzeni barw RGB, foton można zdefiniować następująco:

```
STRUCT Photon {  
    FLOAT x, y, z;    //pozycja  
    CHAR power [4];  //moc  
    CHAR phi, theta; //kierunek  
    CHAR flags [2];  //flagi i wyrównanie  
}
```

Pozycja fotonu reprezentowana jest bez żadnej kompresji, gdyż wykorzystywana jest niezwykle często w operacji wyszukiwania. Moc zapisywana jest w formacie skompresowanym opisanym w [34]. Oryginalny kierunek pochodzenia fotonu mapowany jest na współrzędne sferyczne zapewniające 2^{16} różnych kierunków co, zważywszy na fakt, iż nie jest on bezpośrednio wykorzystywany w obliczeniach oświetlenia, jest wartością w zupełności wystarczającą. Ostatnie pole to dwubajtowe flagi, wyrównujące rozmiar struktury do granicy słowa. Przechowują one dodatkowe informacje wykorzystywane przez różne algorytmy:

- dwa najmniej znaczące bity kodują płaszczyznę podziału w kd-drzewie (00 - X, 01 - Y, 10 - Z),
- trzy najbardziej znaczące bity zawierają informację na temat rodzaju fotonu (100 - foton oświetlenia bezpośredniego; 010 - foton oświetlenia pośredniego, lub foton cienia; 001 - foton kaustyczny),
- na pozostałych 11 bitach przechowywany jest identyfikator (numer) światła, które wyemitowało foton.

³Wyjątkiem są fotony cienia, wykorzystywane w czasie obliczania oświetlenia bezpośredniego.

```

LOCATE (p) {
  IF (2 * p + 1 < ilość fotonów) {
    d := odległość do płaszczyzny podziału węzła n.
    IF (d < 0) {
      LOCATE (2 * p)
      IF (d * d < D2)
        LOCATE (2 * p + 1)
    } ELSE {
      LOCATE (2 * p + 1)
      IF (d * d < D2)
        LOCATE (2 * p)
    }
  }
  d2 := kwadratowa odległość od fotonu p do punktu x.
  IF (d2 < D2) {
    dodaj foton do kolejki h.
    D2 := kwadratowa odległość do fotonu w korzeniu h.
  }
}

```

Tabela 4.1.: Algorytm wyszukiwania fotonów.

4.4. Wyszukiwanie fotonów

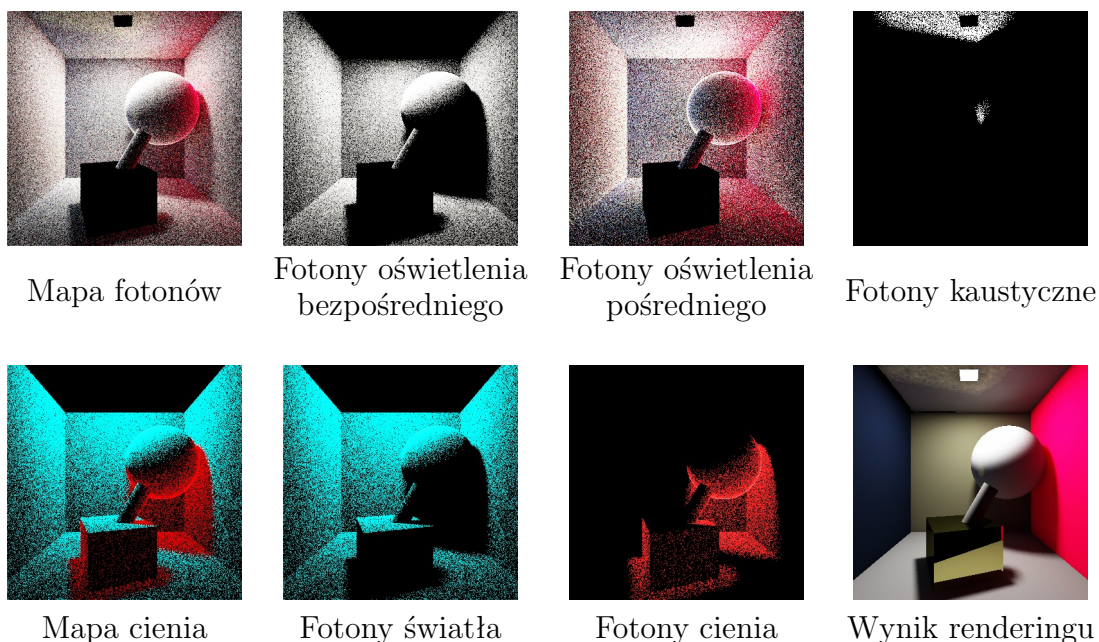
Wyszukiwanie fotonów w mapie jest najczęściej wykonywaną na niej operacją, wykorzystywaną przez praktycznie każdy algorytm stosujący mapę. Z tego powodu jej szybkość (a konkretnie brak) ma decydujący wpływ na całkowitą wydajność. Zapytanie kierowane do mapy ma następującą postać:

Dla danego punktu x w przestrzeni i zadanej maksymalnej odległości od niego d znajdź co najwyżej k najbliższych temu punktowi fotonów.

Zapytanie to można sformułować na wiele odmiennych sposobów (np.: zażądać wszystkich fotonów w zadanym obszarze), jednak podana postać jest najbardziej wydajna i jednocześnie daje najlepsze rezultaty. Ograniczenie maksymalnej odległości wynika z faktu, iż wkład określonej liczby fotonów pochodzących z bardzo dużego obszaru w radiancję odbitą jest znikomy i lepiej posłużyć się ich mniejszą liczbą (oczywiście warto zadbać o to, by nie była to liczba zbyt mała). Z kolei ograniczenie ilości pozwala uniknąć problemu wykorzystania zbyt dużej ich liczby w obszarach o dużej gęstości fotonów. Oczywiście dobranie obu opisywanych wartości nie jest proste i często jest wykonywane metodą prób i błędów. Dlatego bardzo pomocne są metody automatycznego doboru tych parametrów, o czym mowa w paragrafie 6.3.4.

Pseudokod algorytmu wyszukiwania podany jest w tabeli 4.1. Dla zadanego punktu sceny (x) i maksymalnego promienia poszukiwania (D) znajduje on k najbliższych fotonów i zwraca w postaci kopca. Algorytm rozpoczyna swoje działanie na szczycie drzewa zbudowanego na mapie⁴ i dodaje fotony do listy znalezionych, jeśli spełniają kryterium

⁴Parametr p powinien mieć wartość 1 w pierwszym wywołaniu.



Rysunek 4.2.: Przykładowa mapa fotonów, mapa cienia oraz wyrenderowany za ich pomocą obraz.

odległości. Do przetrzymywania i szybkiego uaktualniania listy k najbliższych odnalezionych fotonów wykorzystywana jest kolejka priorytetowa. Po wypełnieniu kolejki k elementami można zmniejszać obszar poszukiwania do wartości równej odległości do najdalszego dotychczas odnalezionego fotonu, zawężając w ten sposób jeszcze bardziej obszar poszukiwań. Ponadto budowa kolejki może być odwleczona do momentu odnalezienia $k + 1$ fotonu.

4.5. Przykładowa mapa

Rysunek 4.2. przedstawia przykładową mapę fotonów oraz mapę cienia. Obie mapy zostały zbudowane dla tej samej sceny, która jest lekką modyfikacją sceny z rysunku 3.4. Kula i walec na tym rysunku odbijają światło w sposób rozproszony, a prostopadłościan w sposób zwierciadlany.

W górnym wierszu zaprezentowano mapę fotonów. Zwizualizowano ją przez rzutowanie każdego z fotonów na płaszczyznę obrazu i sprawdzenie, czy nie jest on zasłonięty przez jakiś obiekt. Zasłonięte fotony są niewidoczne. Kolor fotonu oznacza jego moc. Mapa zawiera wszystkie fotony jakie zostały wygenerowane w fazie konstrukcji. Pierwszy z obrazów przedstawia wszystkie fotony w mapie i stanowi reprezentację promieniowania świetlnego wewnątrz sceny. Zgodnie z tym, co napisano w paragrafie 4.2.3., żaden foton nie został zapisany na powierzchni czysto zwierciadlanego prostopadłościanu oraz źródła światła, które w przypadku tej sceny nie odbija ani nie załamuje światła.

Gęstość mapy wydaje się duża z powodu małego rozmiaru obrazu. Drugi z obrazów przedstawia jedynie fotony pochodzące bezpośrednio od źródła światła, które emituje białe światło. Fotony pokrywają tylko tę część sceny, która jest bezpośrednio widoczna z powierzchni światła. Na kolejnym obrazie widać fotony, które odbiły się przynajmniej raz po wyemitowaniu. Pokrywają one równomiernie całą scenę, a ich kolor zależy od ścieżki jaką przebyły. Ostatni obraz prezentuje fotony kaustyczne pochodzące w tym przypadku jedynie od prostopadłościanu.

Dolny wiersz zawiera obrazy mapy cienia. Pierwszy z nich pokazuje wszystkie fotony z mapy. W tym przypadku kolor fotonu nie ma znaczenia, liczy się jedynie jego typ. Fotony światła mają na tej mapie kolor błękitny, a cienia kolor czerwony. W przypadku tego typu mapy fotony zachowywane są na wszystkich powierzchniach, także tych zwierciadlanych, gdyż służą do oszacowania widoczności źródła światła. Obszary o jednolitej barwie mogą być potraktowane w sposób uproszczony w czasie wyliczania oświetlenia bezpośredniego. Miejsca styku obu rodzajów fotonów muszą być renderowane z o wiele większą dbałością. Więcej informacji na ten temat znajduje się w rozdziale 5. Na drugim i trzecim obrazie oba typy fotonów przedstawione są osobno. Widać na nich dokładnie, że obszary okupowane przez nie pokrywają się z rzeczywistymi obszarami cienia i światła, a także, że fotony cienia zachowywane są w każdym, poza pierwszym, punkcie przecięcia drogi fotonu ze sceną.

Ostatni z obrazów w drugim wierszu prezentuje wynik renderingu przeprowadzonego z wykorzystaniem obu map. Oświetlenie bezpośrednie zostało wyznaczone za pomocą mapy cienia. W obszarach półcienia, wykrytych za pomocą mapy, zastosowano specjalną metodę monte carlo, w celu uzyskania bardzo dokładnego wyniku. Oświetlenie pozostałych punktów wyznaczono wprost z odpowiednich wzorów. Do wyliczenia oświetlenia niebezpośredniego posłużono się mapą fotonów. Obraz ma wysoką jakość, między innymi wyraźnie zarysowana jest dużych rozmiarów kaustyka na suficie.

Rozdział 5.

Oświetlenie bezpośrednie

W niniejszym rozdziale zaprezentowano sposoby wykorzystania mapy fotonów i jej rozszerzeń do wspomagania wyznaczania oświetlenia bezpośredniego. Omówione zostały trzy techniki o różnicowanym stopniu zaawansowania oraz różnych przeznaczeniach.

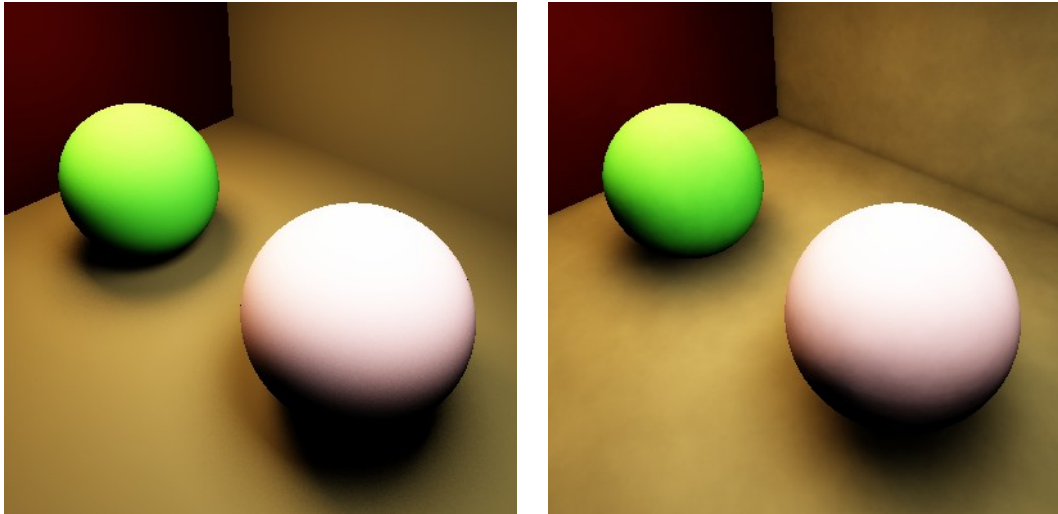
5.1. Wprowadzenie

Jak już wspomniano w rozdziale 1., oświetlenie bezpośrednie jest bardzo istotnym elementem każdego obrazu. Ważna jest przede wszystkim jakość oszacowania. Dużą rolę odgrywa też szybkość, gdyż musi ono być wyliczane w każdym napotkanym punkcie sceny¹. Niestety, tradycyjny dla realistycznej grafiki komputerowej sposób wyznaczania tego elementu polega na zastosowaniu odpowiedniej metody monte carlo do próbkowania powierzchni światła. Daje on wysmienite rezultaty, jednak jest bardzo czasochłonny.

Typowe metody przyspieszania obliczeń polegają na konstruowaniu wyspecjalizowanych funkcji próbkujących dla światła o konkretnych kształtach ([28], [29]). Pozwalają one poprawić wyniki i zmniejszyć liczbę koniecznych próbek. W przypadku scen zawierających wiele światła stosuje się metody ustalające potencjalny wkład każdego ze źródeł światła w danym punkcie. Na podstawie tej wartości przydziela się odpowiednie ilości promieni każdemu ze światła ([29]) oraz bada je w określonej kolejności i tylko do momentu osiągnięcia satysfakcjonującego rezultatu ([37]). Zabieg ten pozwala na zredukowanie całkowitej liczby potrzebnych próbek i na ich lepsze wykorzystanie.

Mapa fotonów zawiera pełną informację o oświetleniu wewnątrz sceny. Oświetlenie bezpośrednie może być z niej wyliczone wprost, o czym mówi podrozdział 5.2. Fotonów bezpośrednich można użyć do wyznaczenia podzbioru światła widocznych w dowolnym punkcie sceny. Szczegóły dotyczące otrzymania i wykorzystania tej informacji oraz używanych korzyści omawia podrozdział 5.3. Wreszcie spokrewniona z mapą fotonów mapa cienia pozwala między innymi odnaleźć wszystkie generowane przez światło obszary półcienia. Opis tej struktury oraz jej zastosowań znajduje się w podrozdziale 5.4.

¹Np.: w rekursywnym algorytmie śledzenia ścieżek oświetlenie bezpośrednie trzeba wyznaczyć w każdym punkcie przecięcia promienia ze sceną na dowolnym poziomie rekursji.



Rysunek 5.1.: Oświetlenie bezpośrednie w scenie z dwoma światłami. Po lewej rozwiązanie wzorcowe, po prawej uzyskane z mapy fotonów.

5.2. Mapa fotonów w oświetleniu bezpośrednim

Informacja o oświetleniu bezpośrednim zawarta jest w fotonach pochodzących wprost od źródła światła. W celu wyznaczenia tego oświetlenia na podstawie mapy fotonów wystarczy więc zastosować metodę estymacji radiancji opisaną w podrozdziale 6.3., biorąc pod uwagę jedynie fotony bezpośrednie.

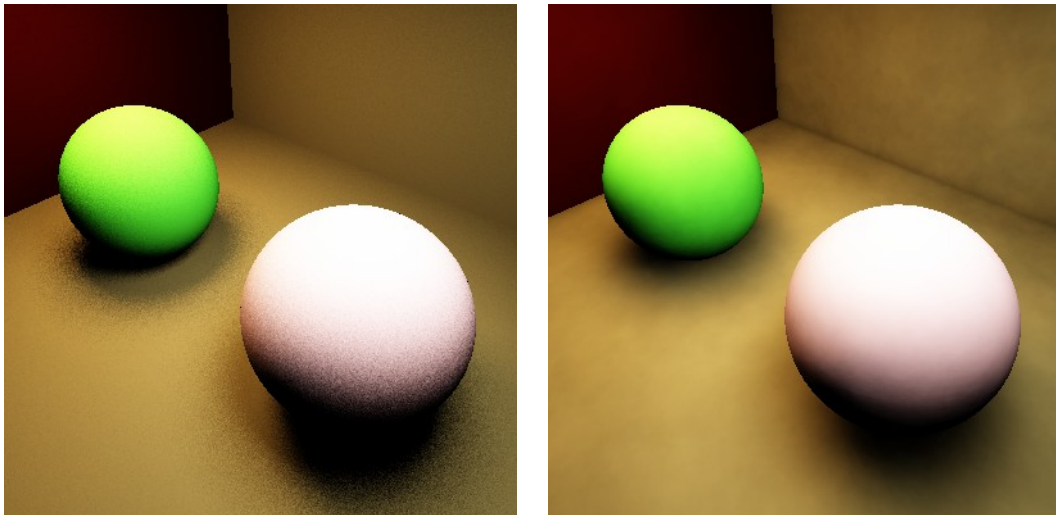
Rysunek 5.1. pozwala na porównanie wyników osiągniętych przy pomocy mapy fotonów oraz metod tradycyjnych. Przedstawiony na nim obraz ukazuje sześcian o boku długości $20m$, w którego wnętrzu znajdują się dwie odbijające w sposób rozproszony kule o promieniu $2.5m$. Scena jest oświetlona przy pomocy dwóch kwadratowych źródeł światła o wymiarach $4m \times 4m$ i bardzo zbliżonej mocy. Opisana konfiguracja generuje duże, trudne do prawidłowego uchwycenia, obszary półcienia. Obydwa obrazy zawierają jedynie oświetlenie bezpośrednie.

Po lewej stronie omawianego rysunku znajduje się rozwiązanie wzorcowe utworzone metodą śledzenia promieni. W każdym punkcie przecięcia promienia pierwotnego ze sceną oświetlenie bezpośrednie było wyliczane przy pomocy 1000 promieni/światło. Użyty obraz w sposób bardzo dokładny odwzorowuje układ cieni. Praktycznie nie zawiera defektów, w szczególności typowego dla śledzenia promieni szumu. Niestety, osiągnięcie tak dobrego rezultatu było bardzo kosztowne (45min 55s) i gdyby chcieć zastosować powyższe parametry w algorytmie rekursywnym, czas tworzenia obrazu stałby się nieakceptowalnie długi.

Obraz po prawej stronie rysunku 5.1. wygenerowany został z wykorzystaniem mapy fotonów. Zawierała ona $4 \cdot 10^5$ fotonów, z czego 198853 bezpośrednich, a około 77800 zostało wykorzystanych przy tworzeniu tego obrazu. W estymacji radiancji używano do 10^3 fotonów znajdujących się w odległości do $1m$ od renderowanego punktu.

5.2. Mapa fotonów w oświetleniu bezpośrednim

Mimo tych parametrów wynik nie jest satysfakcjonujący. W obszarach o jednorodnym natężeniu oświetlenia widoczny jest charakterystyczny dla mapy fotonów szum niskiej częstotliwości. Problemu tego praktycznie nie można wyeliminować. Granice cienia zarówno pod kulami, jak i na ich powierzchniach, są rozmyte. Obszary półcienia i cienia zdecydowanie różnią się od tych z rozwiązania wzorcowego. Przyczyną tego problemu są kłopoty z estymacją radiancji. Uzyskanie dobrego przybliżenia wymaga użycia dużej liczby fotonów. Można ją otrzymać poprzez zbieranie fotonów z dużych obszarów. Powoduje to przedostawanie się fotonów w obszary zacienione, co zmienia ich kształt. Drugim sposobem jest zdecydowane zwiększenie gęstości mapy. To rozwiązanie jest jednak niepraktyczne, gdyż wymagałoby tworzenia zbyt dużych map. Ostatni problem widoczny jest na styku ścian. Estymacja daje w tych miejscach wyraźnie gorsze wyniki, przez co punkty w tej okolicy są nienaturalnie ciemne.



Rysunek 5.2.: Oświetlenie bezpośrednie, wyniki uzyskane w podobnym czasie metodą śledzenia promieni (po lewej) i z mapy fotonów (po prawej).

Ważną zaletą wykorzystania mapy fotonów jest natomiast czas renderingu. Wliczając czas potrzebny na zbudowanie mapy (9.23s) utworzenie obrazu zajęło zaledwie 101.97s. Rysunek 5.2. ukazuje prezentowany wcześniej wynik otrzymany przy użyciu mapy fotonów oraz obraz uzyskany w porównywalnym czasie (101.72s) przy pomocy śledzenia promieni². Różnica pomiędzy nimi jest ciągle dość duża, jednak śledzenie promieni w tym przypadku także generuje defekty wykluczające możliwość użycia go w celu renderowania obiektów widzianych bezpośrednio przez obserwatora.

Jednakże w algorytmach wyliczających oświetlenie globalne w sposób rekursywny, do obliczenia wartości oświetlenia w jednym widzianym przez obserwatora punkcie wykorzystuje się wiele próbek. Dzięki temu można pozwolić na zmniejszenie dokładności

²Wykorzystano 36 promieni/światło.

każdej z nich. Umożliwia to zastosowanie mniej dokładnej metody wyliczania oświetlenia bezpośredniego w przypadku obiektów odnalezionych przez promienie wtórne. Wykorzystanie mapy fotonów pozwala w takich punktach połączyć wyznaczanie obu typów oświetlenia poprzez jednoczesne wyszukiwanie fotonów bezpośrednich i pośrednich. Zabieg ten pozwala praktycznie całkowicie wyeliminować czas potrzebny na wyliczenie oświetlenia bezpośredniego.

Podsumowując — mapa fotonów zawiera informacje na temat oświetlenia bezpośredniego. Nie może być jednak użyta do wyliczenia go wprost w miejscach widzianych przez obserwatora, gdyż otrzymane wyniki są zbyt mało dokładne, a ich poprawienie wiąże się ze zbyt dużym kosztem, który czyni metodę niepraktyczną. Jakość przybliżeń jest jednak satysfakcjonująca w przypadku obiektów osiąganym przez promienie wtórne. W tych punktach mapa fotonów powinna być wykorzystana także do obliczenia oświetlenia bezpośredniego, gdyż pozwala to zredukować czas renderingu o część potrzebną na jego wyznaczenie.

5.3. Wyznaczanie zbioru światel widocznych

Rozkład oświetlenia bezpośredniego w scenach zawierających dużą liczbę źródeł światła jest bardzo skomplikowany. Zazwyczaj poszczególne światła widoczne są jedynie w niewielkich obszarach modelu. Ponadto wśród widocznych w danym punkcie światel znajdują się zarówno takie, które mają dla niego duże znaczenie (mocno go oświetlają), jak i mniej ważne. Oczywiście zbiór światel widocznych oraz istotnych może być różny dla różnych punktów sceny i zmieniać się gwałtownie (np.: na granicy cienia, lub ostrym załamaniu powierzchni). Z tego powodu efektywne wyznaczanie oświetlenia bezpośredniego wymaga konstrukcji metod zdolnych do szybkiego oszacowania widoczności oraz istotności poszczególnych źródeł światła w każdym punkcie modelu.

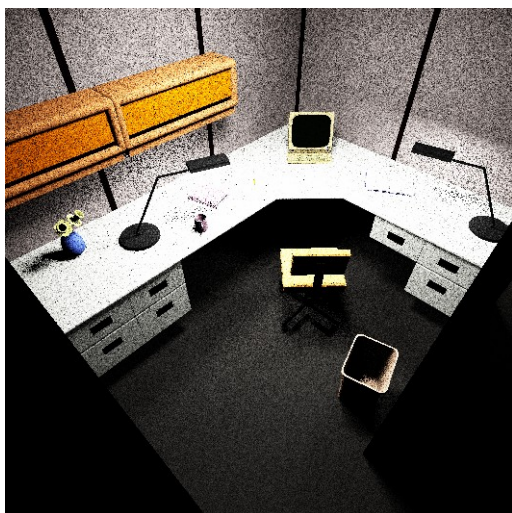
W kolejnych paragrafach tego podrozdziału opisano algorytm selekcji światel zaczerpnięty z pracy [25]. Wykorzystano w nich następujące definicje:

selekcja – oznacza podział światel na widoczne i niewidoczne z określonego punktu,

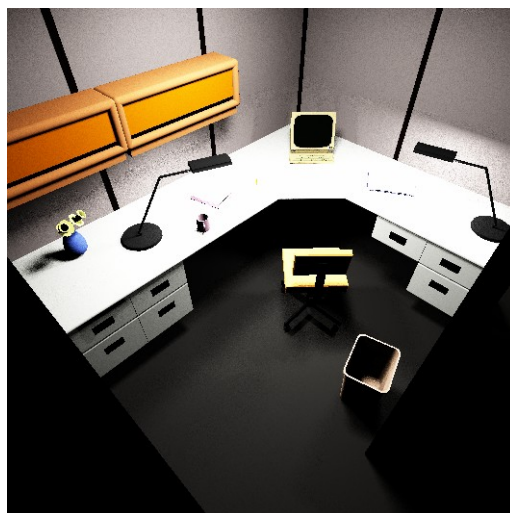
segregacja – oznacza ustalenie istotności światel w określonym punkcie.

5.3.1. Algorytm

Mapa fotonów okazuje się być narzędziem pomocnym przy szybkim podziale zbioru światel na widoczne i niewidoczne. W każdym punkcie sceny, w którym wyliczane jest oświetlenie bezpośrednie, wykonuje się operację wyszukiwania fotonów pochodzących bezpośrednio od źródeł światła. Światła, których identyfikator znajduje się w uzyskanym zbiorze fotonów, uznaje się za widoczne. O pozostałych zakłada się, że są niewidoczne. Następnie, w czasie renderowania, obydwa zbiory światel rozpatruje się oddzielnie. Pomędzy widoczne źródła światła rozdziela się większość dostępnych próbek,



Selekcja: brak
Segregacja: moc światła



Selekcja: mapa fotonów
Segregacja: moc światła

Rysunek 5.3.: Oświetlenie bezpośrednie (50 promieni/piksel) we wnętrzu boksu.

najlepiej wykorzystując w tym celu osobny algorytm ustalający ich potencjalny wkład. Natomiast światła ze zbioru zasłoniętych są praktycznie całkowicie pomijane.

Dokładniej — przy pomocy danych uzyskanych z mapy fotonów, zbiór światel dzielony jest na dwa rozłączne podzbiory, światel widocznych ($\mathcal{L}_v$) oraz niewidocznych ($\mathcal{L}_o$). Każdy z nich jest renderowany przy pomocy osobnego zestawu próbek ($\mathcal{S}_v$, $\mathcal{S}_o$), których ilość jest niezależna od liczby światel. Jeśli $\mathcal{S}_v < |\mathcal{L}_v|$, to w celu uzyskania estymatora nieobciążonego, należy rozdzielić próbki pomiędzy losowo wybrane źródła światła. W przeciwnym razie można wykorzystać funkcje segregujące, aby lepiej zagospodarować dostępne próbki. W przypadku zbioru światel zasłoniętych przyjęcie $\mathcal{S}_o = 1$, czyli wysłanie jednego promienia do losowo wybranego punktu na losowym świetle, zapewnia otrzymanie estymatora nieobciążonego. Wyliczona wartość jest jednak skalowana przez $|\mathcal{L}_o|$, co może być przyczyną wysokiej wariancji w przypadku, gdy próbka trafi w światło mylnie zaklasyfikowane jako niewidoczne. Z tego powodu w czasie selekcji światel należy wykorzystywać odpowiednio duże obszary wyszukiwania fotonów w celu zminimalizowania możliwości popełnienia błędu. Innym sposobem jest pominięcie zbioru światel zasłoniętych. Wynikowy estymator będzie co prawda obciążony, ale przy odpowiednio dobranych parametrach selekcji nie wpłynie to na wygląd utworzonego obrazu.

5.3.2. Przykłady

Pierwsza scena testowa reprezentuje typowe wnętrze biura. Jest ono podzielone na sześć identycznych boksów. Każdy z nich jest oświetlony przez dwie lampki biurkowe o równej mocy oraz jedną podwieszaną lampę sufitową, o mocy trzy razy większej od mocy

5.3. Wyznaczanie zbioru światel widocznych

Metoda segregacji światel	moc światła	moc światła
Metoda selekcji światel	brak	mapa fotonów
Koszt selekcji	0.0s (0%)	16.21s (11%)
Całkowita ilość promieni	14045000	12420600
Promienie utracone	11702164 (83.3%)	329299 (2.6%)

Tabela 5.1.: Statystyki dla metod z rysunku 5.3.

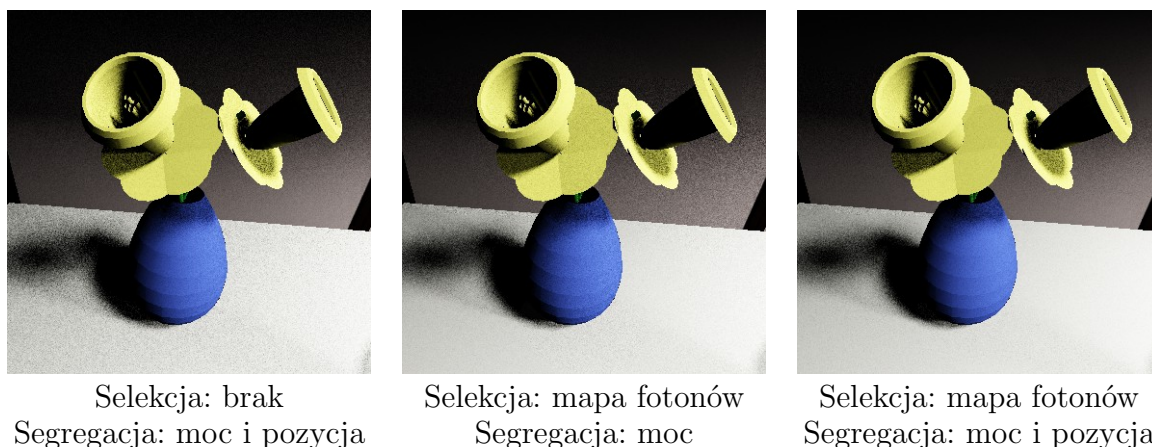
lampki. Boksy oddzielone są od siebie ściankami w taki sposób, że światło pochodzące z lamp znajdujących się w jednym z boksów nie oświetla bezpośrednio pozostałych. Mamy więc 18 źródeł światła, ale w każdym punkcie sceny istotne są co najwyżej trzy z nich.

Rysunek 5.3. przedstawia widok jednego z boksów utworzony przy pomocy dwóch różnych metod. Obydwa obrazy zawierają jedynie oświetlenie bezpośrednie wygenerowane przy pomocy 50 promieni. Po lewej stronie znajduje się wynik powstały przy wykorzystaniu metody tradycyjnej, która rozdziela promienie pomiędzy światła wyłącznie na podstawie ich mocy. Przy generowaniu drugiego z rysunków posłużono się dodatkowo zwykłą mapą fotonów, w celu odnalezienia światel widocznych. Różnica pomiędzy osiągniętymi rezultatami jest ogromna. Obraz po lewej stronie zawiera olbrzymią ilość występującego wszędzie z tym samym natężeniem szumu. Jego poziom jest niezależny od stopnia skomplikowania oświetlenia (cień, półcień). Sugeruje to, że zastosowana metoda jest bardzo słaba. Na drugim z obrazów, mimo wykorzystania tej samej liczby promieni, szum jest znacznie zredukowany. Obszary równomiernie oświetlone są praktycznie od niego wolne. Widoczny jest on jedynie w trudnych do prawidłowego zrenderowania obszarach półcienia. Jednak nawet tam jego poziom jest zdecydowanie niższy niż w przypadku poprzedniej metody.

Tabela 5.1. zawiera statystyki porównujące oba algorytmy. Jak widać metoda nie wykonująca selekcji światel wygenerowała i przetestowała olbrzymią ilość promieni cienia skierowanych do światel niewidocznych. Zastosowanie mapy fotonów pozwoliło tego uniknąć. Promienie utracone przez drugą z metod wygenerowane zostały na granicach boksów i są wynikiem przedostania się fotonów z sąsiednich boksów do oszacowania. Wyeliminowanie tego typu strat jest praktycznie niemożliwe, gdyż zapobiegający im mniejszy promień wyszukiwania może powodować wykluczanie widocznych światel w punktach, gdzie mapa ma mniejszą gęstość, lub w obszarach półcienia. To z kolei może być przyczyną powstawania artefaktów. Czas potrzebny na selekcję światel wydaje się dość duży. Jednak przy założeniu, że do wyznaczenia oświetlenia pośredniego wykorzystywana jest mapa fotonów³, jest on stały ze względu na rozdzielczość obrazu. Stąd wniosek, że czas ten będzie taki sam bez względu na jakość generowanego oświetlenia bezpośredniego (liczbę promieni na piksel) i jego procentowy udział będzie malał wraz ze wzrostem jakości.

³Zważywszy na fakt, iż mapa i tak musiała być zbudowana, jest to dość oczywiste założenie.

5.3. Wyznaczanie zbioru światel widocznych



Rysunek 5.4.: Oświetlenie bezpośrednie (50 promieni/piksel), zbliżenie na kwiatek.

Istnieją oczywiście lepsze sposoby segregacji światel niż ten użyty w poprzednim przykładzie. Rysunek 5.4. zawiera zbliżenie kwiatka znajdującego się po lewej stronie prezentowanego wcześniej boksu. Ponownie wykorzystano 50 promieni/piksel, jednak zastosowano odmienne konfiguracje. Obraz po lewej stronie wygenerowano bez selekcji światel. Do segregacji zastosowano bardzo dokładną wersję metody ustalającej potencjalny wkład światła na podstawie jego mocy oraz położenia w stosunku do renderowanego punktu ([29]). W środkowym obrazie selekcja była wykonana przy pomocy mapy fotonów a segregacja na podstawie mocy światel. Obraz po prawej stronie to złożenie selekcji za pomocą mapy fotonów oraz segregacji na podstawie położenia światła.

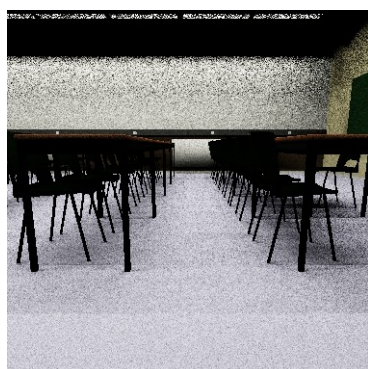
Jak widać zastosowanie lepszej metody segregacji światel zdecydowanie poprawia wyniki. Obrazy po lewej i środkowy są bardzo podobnej jakości. Pierwszy z nich charakteryzuje się większym szumem w obszarze stołu, drugi w okolicy ściany. Metoda selekcji światel jest przezroczysta dla technik segregacji, toteż może być z nimi dowolnie składana. Wynik takiej kombinacji przedstawia trzeci z obrazów rysunku 5.4., gdzie do segregacji użyto mocy i pozycji światła. Jest on zdecydowanie najlepszy, dzieliczy dobre cechy pozostałych obrazów.

Przyjrzyjmy się bliżej różnicom pomiędzy dwoma pierwszymi obrazami. Środkowy (selekcja: mapa, segregacja: moc) jest zdecydowanie gorszy w obszarze ściany. Obraz

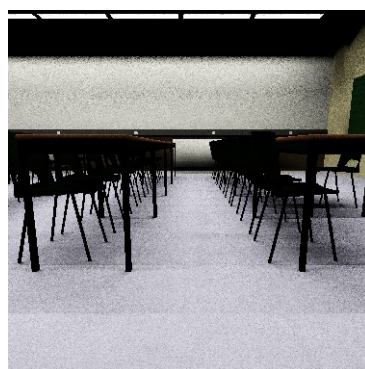
Metoda segregacji światel	moc i pozycja	moc	moc i pozycja
Metoda selekcji światel	brak	mapa fotonów	mapa fotonów
Koszt selekcji i segregacji	3.99s (4%)	8.49s (7%)	11.20s (8.9%)
Całkowita ilość promieni	7070600	7020650	7020650
Promienie utracone	1268075 (18%)	24187 (0.34%)	23545 (0.33%)

Tabela 5.2.: Statystyki dla metod z rysunku 5.4.

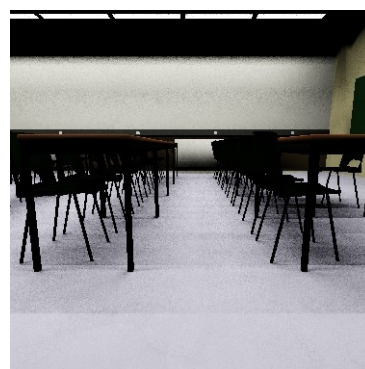
5.3. Wyznaczanie zbioru światel widocznych



Selekcja: brak
Segregacja: moc



Selekcja: mapa fotonów
Segregacja: moc



Selekcja: mapa fotonów
Segregacja: moc i pozycja

Rysunek 5.5.: Oświetlenie bezpośrednie (50 promieni/piksel) we wnętrzu klasy.

po lewej (brak selekcji, segregacja: moc i pozycja) wykazuje natomiast zdecydowanie większy szum na powierzchni blatu biurka. Skąd te różnice? Wyjaśnienie można znaleźć w statystykach zawartych w tabeli 5.2. Otóż w czasie generowania obrazu środkowego i prawego utracono praktycznie taką samą, zaniedbywalną liczbę promieni. Wynika stąd, że jedyną przyczyną niższej jakości tego obrazu jest słabość funkcji segregującej światła, która pogarsza rozkład promieni cienia. Problem ten szczególnie mocno uwiadamia się na powierzchni ściany, która z powodu kąta padania promieni jest oświetlana praktycznie tylko przez lampę sufitową. Natomiast nie jest tak wyraźny na płaszczyźnie biurka położonej pod bardzo podobnym kątem w stosunku do wszystkich trzech lamp. W przypadku obrazu z lewej strony liczba zmarnowanych promieni jest ciągle bardzo duża i to ona ma decydujący wpływ na obniżenie jakości. Zwiększanie liczby światel w scenie będzie powiększało straty (gdyż więcej światel będzie się potencjalnie liczyć), co przyniesie dalszy spadek jakości obrazu. Metody selekcji są więc zbyt słabe w przypadku skomplikowanych układów cieni wynikających z wzajemnego położenia obiektów, a metody segregacji dają gorsze wyniki z powodu tracenia sporej części promieni. Obie techniki uzupełniają się jednak wzajemnie i zastosowane jednocześnie prowadzą do najlepszych rezultatów.

Ostatni z przykładów (rys. 5.5.) to model wnętrza klasy. Pomieszczenie oświetlone jest przez 24 wiszące pod sufitem lampy o identycznej mocy. Jak widać w tym przypadku selekcja światel także pozwoliła poprawić jakość obrazu, ale w mniejszym stopniu. Gorsze wyniki są rezultatem specyficznego układu oświetlenia. W większości punktów sceny widoczne są prawie wszystkie lampy, co nie pozwala na odrzucenie praktycznie żadnego źródła światła. Ponadto wszystkie cienie mają bardzo mały rozmiar, co dodatkowo zmniejsza skuteczność, gdyż dość duży obszar wyszukiwania fotonów sprowadza ich powierzchnię praktycznie do zera. Niestety, zmniejszenie obszaru wyszukiwania fotonów wprowadziłoby artefakty. Alternatywą dla globalnie ustalonego rozmiaru obszaru wyszukiwania fotonów mogłoby być dynamiczne dostosowywanie jego wielkości. Jed-

nak stworzenie odpowiednio ogólnego i szybkiego algorytmu adaptacji wydaje się niemożliwe. Tak jak w poprzednich przykładach wykorzystanie lepszej techniki segregacji światła przyczyniło się do jeszcze większego wzrostu jakości obrazu.

5.3.3. Podsumowanie

Zaprezentowana metoda selekcji światła na podstawie mapy fotonów jest bardzo skuteczna. Jeśli założyć, że mapa fotonów jest wykorzystywana do wygenerowania oświetlenia pośredniego, nie wymaga budowania żadnej dodatkowej struktury danych. Koszt czasowy zastosowania tej techniki jest niewielki, a biorąc pod uwagę otrzymywaną poprawę jakości, pozwala ona wręcz zredukować czas renderingu. Technika ta szczególnie dobrze spisuje się w przypadku dużych scen, zawierających wiele źródeł światła rozsianych po odseparowanych od siebie pomieszczeniach. Gorsze efekty daje dla modeli, w których światła są nieodseparowane i jest raczej bezużyteczna w przypadku, gdy scena zawiera tylko kilka światła widocznych praktycznie wszędzie. Przedstawiona metoda może i powinna być łączona z uzupełniającymi ją technikami segregacji światła, które pozwalają na dalszą poprawę jakości.

5.4. Mapa cienia

Mapa cienia została wprowadzona do realistycznej grafiki komputerowej w pracy [16]. Jest to struktura danych oparta na mapie fotonów. Rozszerza ona jej funkcjonalność pozwalając, w zależności od sposobu wykorzystania, znacząco zredukować ilość potrzebnych promieni cienia, lub całkowicie wyeliminować konieczność ich stosowania przy obliczaniu oświetlenia bezpośredniego. Kolejne paragrafy zawierają: opis struktury danych (5.4.1.), jej zastosowania (5.4.2.) oraz przykłady ilustrujące (5.4.3.).

5.4.1. Struktura danych

Mapa fotonów przechowuje fotony pochodzące bezpośrednio od źródła światła (bezpośrednie) oraz pośrednie (zwykle i kaustyczne) czyli takie, które odbiły się przynajmniej raz od jakiegoś obiektu sceny po opuszczeniu źródła światła. Mapa cienia wprowadza nowy typ fotonu, foton cienia. Fotony cienia tworzone są przez śledzenie promieni biorących swój początek w źródle światła poprzez całą scenę. W punkcie pierwszego przecięcia promienia ze światłem zapisywany jest zwykły foton oświetlenia bezpośredniego. W pozostałych punktach przecięcia powstają fotony cienia. W mapie tego typu zarówno fotony bezpośrednie, jak i cienia, muszą przechowywać identyfikator źródła światła, od którego pochodzą.

Mapa cienia może być tworzona równocześnie z mapą fotonów. W tym przypadku w punkcie przecięcia fotonu bezpośredniego ze sceną jego ścieżka rozdziela się na dwie całkowicie niezależne. Pierwsza z nich związana jest z oświetleniem globalnym. Jej przebieg wyznaczany jest przez rosyjską ruletkę i funkcje BRDF napotkanych obiektów. Druga generuje wzdłuż promienia, którym podążał foton bezpośredni, fotony cienia.

Zawartość tak tworzonej mapy cienia (czyli fotony bezpośrednie i fotony cienia) można przechowywać na kilka sposobów. Najprostsza implementacja będzie zapisywać wszystkie fotony w jednej, połączonej mapie. Zaletą tego rozwiązania jest oszczędność miejsca. Każdy foton jest przechowywany dokładnie jeden raz. Wadą jest natomiast zwiększona ilość fotonów w mapie. Jak opisano w podrozdziale 4.3., mapa przechowywana jest w postaci kd-drzewa. Większa ilość fotonów oznacza większą gęstość i przez to dłuższy czas wyszukiwania. Ponadto fotony cienia nigdy nie są wyszukiwane w tym samym zapytaniu co fotony oświetlenia pośredniego. Zapisywanie obu rodzajów fotonów w tej samej mapie powoduje oddalanie się od siebie fotonów tego samego typu w kd-drzewie. Wymusza to przeszukiwanie większych obszarów drzewa. Rozwiązaniem tego problemu jest przechowywanie fotonów bezpośrednich w zwykłej mapie, a fotonów cienia w osobnej, przeznaczonej specjalnie dla nich. Pozwala to zredukować rozmiar mapy fotonów, nie zwiększając całkowitego rozmiaru utworzonych struktur danych. Niestety, wyznaczanie oświetlenia bezpośredniego w tej konfiguracji wymaga przeszukiwania dwóch oddzielnych map, co w większości przypadków zwiększa całkowity czas działania. Ostatnie możliwe rozwiązanie to zapisanie fotonów bezpośrednich zarówno w mapie fotonów jak i mapie cienia. Powoduje to utratę pewnej ilości pamięci. Jednak zarówno wyliczanie oświetlenia bezpośredniego, jak i pośredniego (lub globalnego), wykorzystuje tylko jedną, specjalnie do tego celu przeznaczoną mapę, co czyni to rozwiązanie najszybszym.

5.4.2. Zastosowanie

W podstawowym zastosowaniu mapa cienia wykorzystywana jest do szybkiego określenia, w jaki sposób źródło światła jest widoczne w dowolnym punkcie sceny. W celu otrzymania informacji na temat oświetlenia pochodzącego od światła l w punkcie x , należy odnaleźć k fotonów najbliższych temu punktowi. Oczywiście pod uwagę powinny być brane jedynie fotony bezpośrednie oraz cienia pochodzące od światła l . Wynikiem zapytania jest zbiór fotonów zawierający pewną ilość fotonów cienia (k_s) oraz pewną ilość fotonów reprezentujących oświetlenie bezpośrednie (k_d). Na podstawie zawartości tego zbioru oświetlenie jest zaliczane do jednej z trzech grup:

- $k_d = 0, k_s = k$ – punkt x znajduje się całkowicie w cieniu światła l , to światło nie oświetla go wcale,
- $k_d = k, k_s = 0$ – punkt x jest oświetlany przez światło l , to światło oświetla go całą widoczną z tego miejsca powierzchnią,
- $0 < k_d, k_s < k$ – punkt x znajduje się w obszarze półcienia dla światła l , lub bardzo blisko tego obszaru, to światło najprawdopodobniej oświetla go jedynie częścią widoczną z tego miejsca powierzchni.

Powyższa klasyfikacja jest wykorzystywana w następujący sposób. Wszystkie źródła światła zaliczone do zbioru niewidocznych są całkowicie pomijane w obliczeniach, ich wkład w oświetlenie punktu jest zerowy. Wkład światła widocznych całkowicie jest

równy 100%. Wyliczenie oświetlenia pochodzącego od takiego światła wymaga jedynie określenia kąta bryłowego, pod którym światło jest widziane. Wartość radiancji w punkcie x generowanej przez w pełni widoczne światło l może być przybliżona na podstawie następującego wzoru:

$$L(x, \vec{\omega}) = \frac{A_l}{m} \sum_{i=1}^m \left(\underbrace{\frac{\cos(x \rightarrow y_i, n_x) \cos(y_i \rightarrow x, n_{y_i})}{r_{x,y_i}^2}}_{G(x,y_i)} L_e(y_i \rightarrow x) f_r(x, x \rightarrow y_i, \vec{\omega}) \right), \quad (5.1.)$$

gdzie:

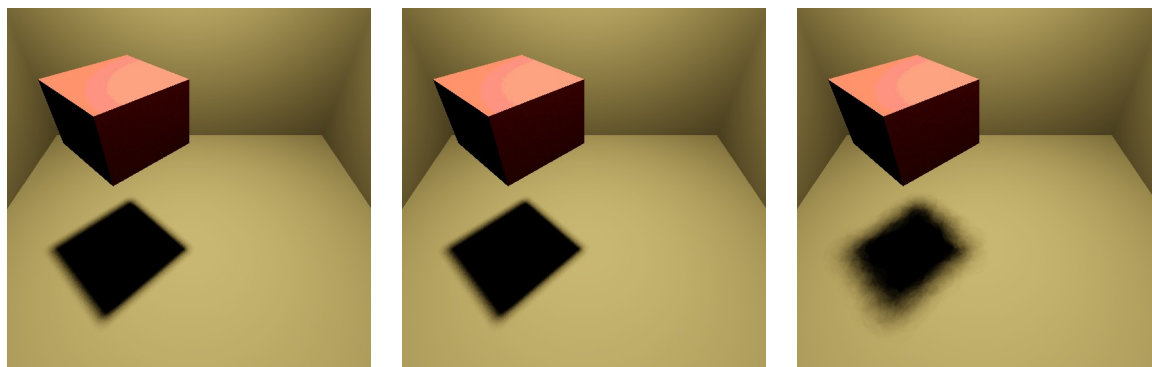
- x, n_x – punkt dla którego wykonywane jest przybliżenie, normalna w tym punkcie,
- y_i, n_{y_i} – losowy punkt na powierzchni światła, normalna w tym punkcie,
- r_{x,y_i} – odległość pomiędzy punktami x i y_i ,
- $G(x, y_i)$ – współczynnik geometryczny,
- A_l – powierzchnia światła l ,
- m – ilość wykorzystanych losowych próbek.

Powyższy wzór jest często stosowany przy przybliżaniu oświetlenia bezpośredniego pochodzącego od jednego źródła światła za pomocą śledzenia promieni. W opisywanym przypadku wymaga on jedynie wylosowania z rozkładem jednorodnym m punktów na powierzchni źródła światła oraz ustalenia normalnych w tych punktach. Nie zachodzi natomiast potrzeba żadnego śledzenia promieni, gdyż z oszacowania wiadomo, że całe światło jest widoczne. Pozwala to zaoszczędzić mnóstwo czasu.

Parametr m umożliwia sterowanie jakością przybliżenia. Im większa ilość próbek, tym jest ono dokładniejsze. Wartość 50 jest wystarczająca nawet dla światel o dużych powierzchniach.

Zastosowanie rozkładu jednorodnego może powodować trawienie próbek, szczególnie w przypadku światel o skomplikowanych kształtach. Dzieje się tak, gdyż część z wylosowanych punktów znajdzie się na niewidocznych ścianach światła. W obszarach tych zarówno czynnik $G(x, y)$ jak i funkcja BRDF przyjmą wartość zero⁴. Problem ten można wyeliminować przez konstruowanie funkcji rozkładu dedykowanych dla światel konkretnego kształtu. W praktyce jednak skonstruowanie takiej funkcji jest bardzo trudne, a ona sama jest kosztowna obliczeniowo. Dlatego lepiej jest wykorzystać rozkład jednorodny i większą ilość próbek.

⁴Oczywiście pod warunkiem, że ściana zawierająca punkt y jest odwrócona tyłem do punktu x . Inaczej oba czynniki będą dodatnie, a punkt y zostanie błędnie zaliczony jako widoczny. Na szczęście światła o tak skomplikowanym kształcie zdarzają się bardzo rzadko i błąd ten można zazwyczaj zaniedbać.



Rozwiązanie wzorcowe
200 pr./piks.

Mapa cienia
200 pr./piks. w półcieniu

Mapa cienia
Półcienie z wzoru 5.2.

Rysunek 5.6.: Oświetlenie bezpośrednie w scenie z jednym światłem.

Obydwa omówione dotychczas przypadki odpowiadają sytuacjom prostym z punktu widzenia oświetlenia bezpośredniego, gdyż nie mamy w nich do czynienia z półcieniem. Powierzchnia światła zaklasyfikowanych do trzeciego ze zbiorów jest w rozpatrywanym punkcie widoczna tylko częściowo. Oznacza to, że w tym miejscu światła te generują półcień. Z tego powodu powinny one być potraktowane dokładniej. Możliwe są dwa rozwiązania. W prostszym procent widocznej powierzchni światła wylicza się na podstawie następującego wzoru:

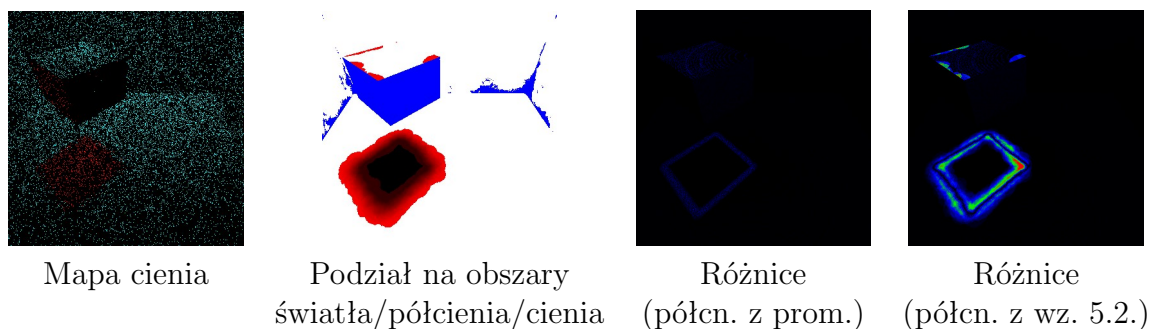
$$V = \frac{k_d}{k_d + k_s}, \quad (5.2.)$$

a następnie wykorzystuje otrzymaną wartość do przeskalowania radiancji otrzymanej przy pomocy wzoru 5.1. To rozwiązanie jest oczywiście mało dokładne. Alternatywą dla niego jest wykorzystanie promieni cienia w celu bardzo dokładnego wyliczenia oświetlenia pochodzącego od każdego z podejrzanych światła. Ten sposób jest zdecydowanie lepszy, ale i o wiele kosztowniejszy.

5.4.3. Przykłady

Pierwsza scena testowa składa się z niewielkiego, zawieszonego w powietrzu, czerwonego sześcianu. Jest on oświetlony przez koliste światło znajdujące się przy suficie. Jedynym elementem rzucającym cień jest sześcian. Z powodu małych rozmiarów lampy jego boczne ściany są bardzo słabo oświetlone. Obszary półcienia są niewielkie. Z punktu widzenia oświetlenia bezpośredniego scena jest prosta.

Rysunek 5.6. przedstawia obraz opisanego modelu utworzony trzema różnymi technikami. Po lewej stronie znajduje się rozwiązanie wzorcowe. Utworzono je poprzez badanie widoczności światła przy pomocy 200 promieni/piksel w każdym punkcie sceny. W przypadku obrazu środkowego zastosowano mapę cienia do wykrycia obszarów półcienia. Następnie obszary te zrenderowano wykorzystując 200 promieni/piksel. W po-



Rysunek 5.7.: Mapa cienia, obszary półcienia, różnice między obrazami.

zostałych miejscach oświetlenie bezpośrednio wyznaczono wprost ze wzoru 5.1. Obraz po prawej stronie wygenerowano jedynie przy użyciu mapy cienia. Do utworzenia półcienia wykorzystano wzór 5.2.

Jak można się było spodziewać, zarówno obszary całkowicie oświetlone, jak i całkowicie zacienione, wyglądają identycznie na wszystkich trzech obrazach. Różnice są niewykrywalne, mimo iż metoda klasyczna wykorzystywała aż 200 promieni, a metody szacujące za pomocą mapy tylko 50 próbek. Do renderowania obszarów oświetlonych wystarczyłoby więc w zupełności 50 promieni. Jednak metoda tradycyjna nie wie, gdzie znajdują się wymagające większej dokładności półcienie. Również różnice w wyglądzie półcienia na obrazie wzorcowym i środkowym są niedostrzegalne. Obszary półcienia zostały poprawnie wykryte i zastosowano w nich metodę dokładną. Natomiast półcienie utworzone przy pomocy wzoru 5.2. (obraz po prawej) zdecydowanie odbiegają od wzorca. Są zbyt rozmyte i postrzępione, ich wygląd jest bardzo nienaturalny.

Rysunek 5.7. pozwala dokładniej przyjrzeć się procesowi tworzenia obrazów przy wykorzystaniu mapy cienia oraz zrewidować uzyskane wyniki. Pierwszy od lewej obraz przedstawia mapę cienia (obydwa obrazy wygenerowano przy pomocy tej samej mapy). Na błękitno zaznaczono fotony oświetlenia, a na czerwono cienia. Jak widać układ fotonów dokładnie odwzorowuje pozycję cienia.

Drugi z obrazów przedstawia utworzony na podstawie mapy podział sceny na obszary całkowicie oświetlone (biały), półcienia (czerwony), całkowicie zacienione (czarny) oraz te, w których zbyt niska gęstość mapy uniemożliwiła klasyfikację (niebieski). Widać, że prawdziwe półcienie zostały wykryte. Jednak ich rozmiar jest większy niż w rzeczywistości. Zjawisko to jest konsekwencją przyjęcia strategii konserwatywnej. Bezpieczniej jest zakwalifikować większe obszary do badanego półcienia, niż przypadkowo przeoczyć jakiś szczegół. Także w miejscach, gdzie mapa ma niską gęstość i udaje się zebrać tylko niewielką liczbę fotonów (np.: w rogach sceny), lepiej użyć metody dokładnej, niż ryzykować błędną klasyfikację. Na górnej, w pełni oświetlonej, ścianie sześcianu pojawiły się fałszywe półcienie. Są one wynikiem przeskakiwania fotonów z sąsiednich powierzchni.

Dwa pozostałe obrazy przedstawiają różnice w wartości radiancji odbitej pomiędzy rozwiązaniem wzorcowym, a tym uzyskanym z wykorzystaniem map. Obszary czar-

	rozwiązanie wzorcowe	m. cienia + promienie w półcieniu	m. cienia
Promienie w półcieniu	200	200	50 (próbek)
Promienie w cieniu	200	0	0
Promienie w pełnym świetle	200	50 (próbek)	50 (próbek)
Wykorzystane promienie	64297800	11783800	0
Fotony w oszacowaniu cienia	0	20-50	20-50
Fotony (fotony cienia)	0(0)	10^5 (1696)	10^5 (1696)
Czas renderingu	555.3s	123.69s	74.61s

Tabela 5.3.: Statystyki dla metod z rysunku 5.6.

ne oznaczają brak różnic, czerwone odpowiadają różnicy o wartości 227.25. Kolory pośrednie (niebieski, zielony, żółty) mapowane są w sposób liniowy na ten przedział. Obrazy te potwierdzają wcześniejszą tezę o bardzo dużej dokładności odwzorowania oświetlenia w miejscach całkowicie oświetlonych bądź zacienionych. W przypadku generowania półcieni za pomocą promieni cienia także różnice w obszarach półcienia są bardzo niewielkie (maksymalna wynosiła 53.625). Błędy w klasyfikacji wywołane przez zbyt niską gęstość mapy i przeskakiwanie fotonów również nie wpłynęły na jakość wyniku. Jest to konsekwencją przyjęcia zasady używania metody dokładnej (śledzenia promieni) w każdym niepewnym punkcie. Inaczej jest w przypadku estymacji opartej na wzorze 5.2. (rysunek po prawej). Tu rozbieżności są znacznie większe i obejmują zdecydowanie większy obszar. Pojawiają się one nie tylko w okolicy rzeczywistych półcieni, ale także w innych błędnie zakwalifikowanych miejscach. Bardzo charakterystyczny jest obszar gwałtownego spadku błędu otoczony ze wszystkich stron błędem większym. Pokrywa się on z powierzchnią zajmowaną przez rzeczywisty półcień. Stąd wniosek, że prawdziwy półcień przybliżany jest dobrze. Pojawiający się w jego sąsiedztwie błąd wynika z przenikania okolicznych fotonów do sąsiednich obszarów, co zaburza estymację w tych miejscach.

Tabela 5.3. zawiera statystyczne porównanie wszystkich wykorzystanych metod. Rozmiar mapy cienia może wydawać się duży, jednak zawierała ona zaledwie 1696 fotonów cienia. Jeśli założyć wykorzystanie także zwykłej mapy fotonów i przechowywanie fotonów bezpośrednich tylko raz, rzeczywisty narzut pamięciowy był znikomy. Skorzystanie z pomocy mapy cienia pozwoliło zredukować ilość potrzebnych promieni 5.5 krotnie oraz czas renderingu 4.5 raza bez jakiejkolwiek utraty jakości⁵. Niestety, całkowite wyeliminowanie promieni cienia wymagałoby, podobnie jak w przypadku algorytmów omawianych w podrozdziale 5.2., użycia o wiele gęstszej mapy, która zmniejszyłaby efekt przenikania fotonów do sąsiednich obszarów. Fakt ten czyni metodę wyznaczania półcieni na podstawie wzoru 5.2. raczej bezużyteczną.

W drugim teście wykorzystano ponownie scenę z podrozdziału 5.2. Zawiera ona znacznie trudniejsze do poprawnego uchwycenia duże i nieregularne obszary półcienia,

⁵Dla metody wykorzystującej promienie cienia w obszarach półcienia.

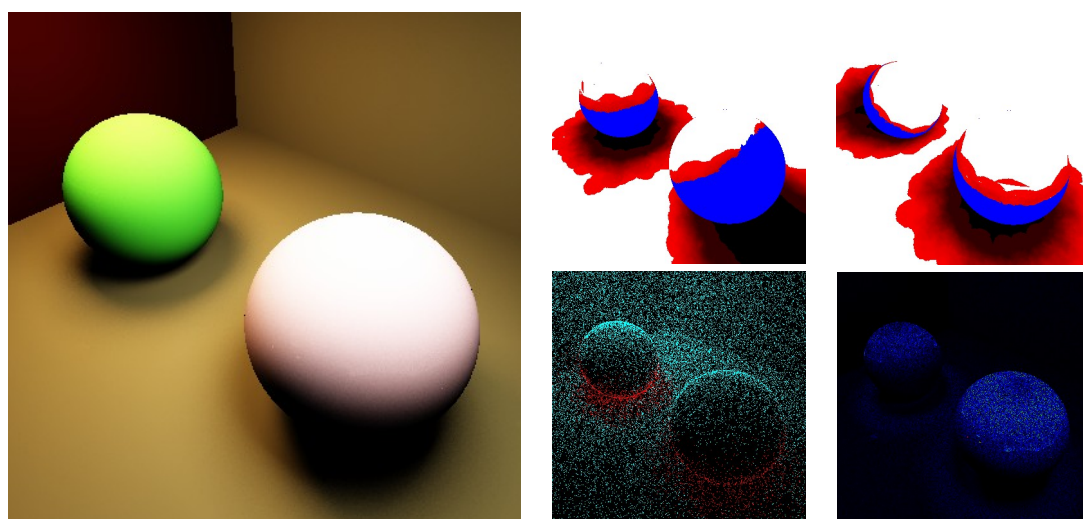
5.4. Mapa cienia

	śledzenie promieni		mapa cienia + śledzenie promieni	
Promienie w półcieniu	36	1000	36	1000
Promienie w cieniu	36	1000	0	0
Promienie w pełnym świetle	36	1000	50 (próbek)	300 (próbek)
Wykorzystane promienie	11520000	320000000	3661416	101706000
Fotony w oszacowaniu cienia	0	0	30-50	30-50
Fotony (fotony cienia)	0(0)	0(0)	$10^5(14082)$	$10^5(14082)$
Czas renderingu	101.72s	45min 55s	97.75s	18min 4s

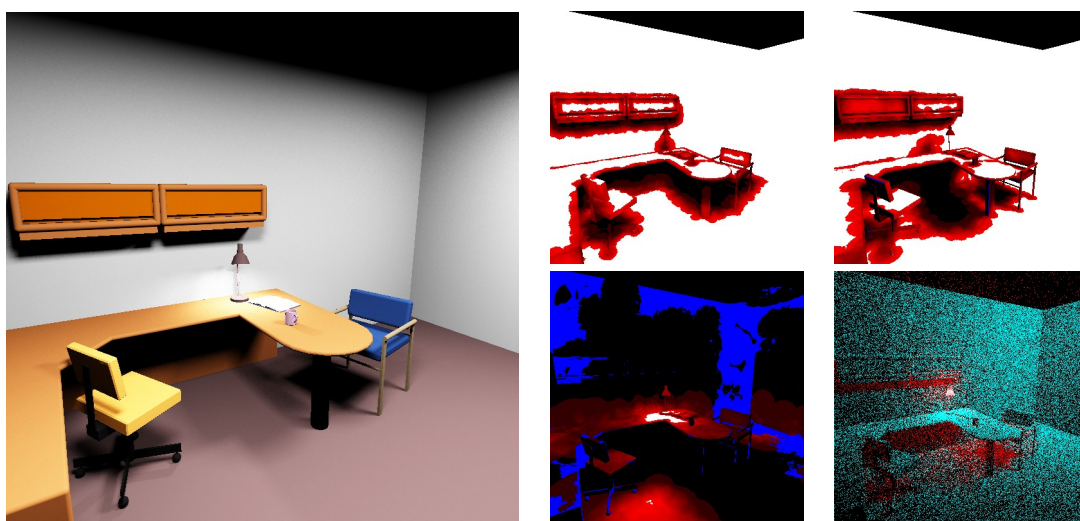
Tabela 5.4.: Porównanie śledzenia promieni z mapą cienia i bez dla rysunku 5.8.

które dodatkowo znajdują się na powierzchniach krzywoliniowych (kulach). Ponownie użyto mapy zawierającej 10^5 fotonów. Tym razem znalazło się w niej o wiele więcej fotonów cienia (patrz tabelka 5.4.). Liczba tych fotonów zależy od całkowitej powierzchni obszarów cienia generowanych przez wszystkie światła, stąd tak duży wzrost w porównaniu z poprzednim przykładem. Mimo to zajmowały one zaledwie 275kb pamięci.

W pierwszym teście wykorzystano zaledwie 36 promieni/światło w obszarach półcienia, czyli zaprezentowaną w podrozdziale 5.2. konfigurację szybką. Jak wynika z danych przedstawionych w tabeli 5.4., ilość potrzebnych promieni cienia udało się zmniejszyć 3.1 raza. Niestety, praktycznie nie obniżyło to czasu renderingu. Długi czas działania



Rysunek 5.8.: Po lewej rozwiązanie wzorcowe z podrozdziału 5.2. utworzone z wykorzystaniem mapy cienia do wykrycia obszarów półcienia. Po prawej u góry podział na obszary światła/półcienia/cienia dla obu światel. Na dole mapa fotonów i obraz różnic między rozwiązaniami.



Rysunek 5.9.: Po lewej: oświetlenie bezpośrednie wewnątrz biura utworzone z użyciem mapy cienia. Po prawej u góry obszary półcienia dla lamp sufitowych, na dole półcienie dla lampki biurkowej i mapa cienia.

jest wynikiem wysokiego kosztu klasyfikacji poszczególnych punktów. Całkowity czas pochłonięty przez tą operację wynosił 78.68s i wynikał z konieczności wykorzystania większej liczby fotonów zbieranych z dość dużych obszarów sceny. Wpłynęło to także na zwiększenie obszarów półcienia badanych metodą dokładną (rysunek 5.8.). Problemy z klasyfikacją były efektem większej liczby światel, która spowodowała spadek ilości fotonów emitowanych przez każdą z lamp. Dodatkowe kłopoty sprawiały kule. Duża część ich powierzchni nie została sklasyfikowana wcale, a w obszarach półcienia znajdują się charakterystyczne, świadczące o problemach, wyrwy. Wszystkie opisane trudności nie miały jednak wpływu na jakość wyniku, prawdziwe obszary półcienia zawierają się bowiem wewnątrz tych wykrytych przez mapę cienia.

Czas klasyfikacji punktów przy ustalonych parametrach jest stały, bez względu na ilość wykorzystanych promieni. Dlatego mimo kłopotów z klasyfikacją, wraz ze wzrostem jakości obrazu, metoda wykorzystująca mapę fotonów będzie uzyskiwała coraz wyraźniejszą przewagę czasową. Ukazuje to zaprezentowany na rysunku 5.8. wynik drugiego z testów, w którym konfigurację wzorcową z podrozdziału 5.2. (1000 promieni/piksel) rozszerzono o mapę cienia. Wykorzystano tą samą mapę co poprzednio, więc liczba potrzebnych promieni cienia została zredukowana o dokładnie ten sam współczynnik. W tym przypadku udało się jednak 2.5 krotnie skrócić czas działania. Niewielkie odchylenia widoczne na obrazie różnic wynikają z zastosowania tylko 300 losowych próbek w oszacowaniu pełnego oświetlenia. Podniesienie ich liczby do 1000 eliminuje je kosztem dodatkowych 132s. Nie jest to jednak konieczne, gdyż różnice w wynikowym obrazie są niedostrzegalne.

Ostatni z przykładów (rys. 5.9.) prezentuje oświetlenie bezpośrednie wewnątrz po-

mieszczania biurowego. Jest ono generowane przez dwie lampy sufitowe i jedną biurkową. W czasie renderingu wykorzystano 131782 fotony cienia (2.5Mb). Pozwoliło to 2.1 raza zredukować czas oraz 3.5 krotnie ilość potrzebnych promieni cienia w stosunku do dającej podobne rezultaty metody wykorzystującej położenie światła w celu określenia jego istotności ([29]). Obraz nie zawiera żadnych widocznych artefaktów w obszarach półcienia. Wszystkie, nawet małe i skomplikowane cienie, jak te znajdujące się pod nogami krzeseł, pod biurkiem, generowane przez kubek czy oparcia krzeseł, zostały zrenderowane prawidłowo. Mimo to w większości obszarów sceny udało się całkowicie wyeliminować konieczność stosowania promieni cienia. Przykład ten pokazuje, że mapa cienia może być z powodzeniem wykorzystywana we wspomaganiu renderowania rzeczywistych scen o dużych rozmiarach i skomplikowanym układzie oświetlenia bezpośredniego.

5.5. Podsumowanie

Mimo że mapa fotonów została zaprojektowana przede wszystkim jako struktura służąca wspomaganiu wyznaczania oświetlenia pośredniego, może być także wykorzystana do znacznego przyspieszenia obliczania oświetlenia bezpośredniego. W niezmienionej formie można jej użyć do zawężenia zbioru rozpatrywanych światel oraz do renderowania oświetlenia bezpośredniego w miejscach, które nie są widziane bezpośrednio przez obserwatora. Pierwsze z zastosowań pozwala lepiej zagospodarować promienie cienia oraz zredukować całkowitą ilość koniecznych promieni w scenach zawierających dużą ilość światel. Drugie praktycznie całkowicie eliminuje czas potrzebny na wyznaczenie oświetlenia bezpośredniego. Oba rozwiązania znacząco redukują czas obliczeń, nie wymagając przy tym wykorzystania żadnej dodatkowej pamięci ani tworzenia nowych struktur danych.

Będąca modyfikacją mapy fotonów mapa cienia pozwala podzielić całą scenę na obszary, w których łatwo wyliczyć oświetlenie bezpośrednio bez użycia promieni cienia (miejsca całkowicie oświetlone lub zacienione) oraz te, w których śledzenie promieni jest konieczne (półcienie). Pozwala to na dalszą oszczędność czasu, jednak wiąże się z koniecznością wykorzystania dodatkowej pamięci potrzebnej do przechowania fotonów cienia. Dodatkowy koszt jest jednak niewielki w porównaniu z osiąganym przyspieszeniem. Wyniki uzyskiwane przy pomocy mapy cienia są bardzo wysokiej jakości i z powodzeniem mogą być wykorzystywane w przypadku powierzchni widzianych bezpośrednio przez obserwatora.

Rozdział 6.

Oświetlenie pośrednie i globalne

W pierwszej części tego rozdziału przedstawiono przegląd metod wykorzystania mapy fotonów do obliczania oświetlenia pośredniego. Wprowadzone tutaj i w poprzednim rozdziale techniki wystarczają do stworzenia bardzo efektywnych algorytmów renderowania oświetlenia globalnego, które opisano w dalszej części rozdziału.

6.1. Wprowadzenie

Oświetlenie pośrednie jest drugim ze składników oświetlenia. Podobnie jak oświetlenie bezpośrednie, jest ono bardzo istotne. W świecie rzeczywistym odpowiada ono za całą gamę zjawisk, takich jak: oświetlenie punktów, na które nie pada bezpośrednio żadne światło, kaustyki, krwawienie kolorów, etc. Niestety, dokładne wyznaczanie tego elementu obrazu wiąże się z wieloma złożonymi problemami, a tradycyjne metody są zazwyczaj bardzo mało wydajne, lub ograniczone do prostych modeli. Renderowanie obrazów dobrej jakości przy ich pomocy wymaga tworzenia algorytmów hybrydowych, w których za każdy z elementów oświetlenia pośredniego odpowiada inna, wyspecjalizowana technika.

Mapa fotonów jest strukturą danych umożliwiającą między innymi tworzenie algorytmów wspomagających wyznaczanie oświetlenia pośredniego. Choć koncepcja wykorzystania w tym celu cząstek światła była obecna w grafice komputerowej od bardzo dawna, same mapy fotonów w takiej postaci, w jakiej zaprezentowano je w tej pracy, wprowadzone zostały po raz pierwszy stosunkowo niedawno, w artykule [18]. Okazały się one być narzędziem bardzo skutecznym, a ponadto szybkim i eleganckim. Umożliwiły odtwarzanie praktycznie wszystkich, nawet bardzo trudnych w symulacji, elementów oświetlenia pośredniego. Na ich bazie udało się zbudować wiele algorytmów wyliczania oświetlenia globalnego o niespotykanej dotychczas szybkości i dokładności. Ich dodatkowymi zaletami są prostota i duża elastyczność, umożliwiające tworzenie coraz to nowszych zastosowań.

6.2. Przyspieszanie zbieżności śledzenia ścieżek

Jak już wspomniano w rozdziale 3., metody śledzenia promieni połączone z technikami monte carlo, zwłaszcza śledzeniem ścieżek, formułują najogólniejsze i najelastyczniejsze algorytmy obliczania oświetlenia globalnego. Są one stosowane jako podstawa wielu bardziej zaawansowanych technik (w tym większości metod opartych na mapie fotonów). Niestety, obliczanie oświetlenia pośredniego na powierzchniach odbijających światło w sposób rozproszony z wykorzystaniem tych metod jest bardzo mało efektywne. Szczególnie słabe wyniki otrzymywane są w przypadku, gdy rozkład oświetlenia pośredniego wewnątrz sceny jest bardzo nierównomierny.

Podrozdział ten wprowadza opartą na mapie fotonów metodę, pozwalającą poprawić zbieżność algorytmów śledzenia promieni. Korzysta ona z faktu, że mapa fotonów jest przybliżeniem rozkładu oświetlenia wewnątrz sceny, a każdy zapisany w niej foton przechowuje między innymi kierunek, z jakiego przybył. Informacje te służą do generowania lepszych, uwzględniających charakterystykę oświetlenia, kierunków śledzenia promieni wtórnych. Opisywany algorytm pochodzi z pracy [12].

6.2.1. Podstawy teoretyczne i algorytm

Wartość dowolnej całki postaci¹:

$$I = \int f(x) dx, \quad x \in D \subset R^n, \quad (6.1.)$$

może być przybliżona przy pomocy techniki monte carlo nazywanej średnią próbek (*ang. sample-mean*). Wartość całki jest wtedy reprezentowana przez wartość oczekiwaną zmiennej losowej X o rozkładzie $p(x)$, $x \in D$ takim, że $p(x) > 0$ dla $f(x) \neq 0$:

$$I = \int \frac{f(x)}{p(x)} p(x) dx = E \left\{ \frac{f(x)}{p(x)} \right\}. \quad (6.2.)$$

Całkę przybliża się poprzez wykorzystanie n próbek $x_1, x_2, \dots, x_n$ wylosowanych z rozkładem $p(x)$:

$$I \approx \frac{1}{n} \sum_{i=1}^n \frac{f(x_i)}{p(x_i)}. \quad (6.3.)$$

Dokładność opisanego przybliżenia zależy od ilości próbek n oraz od funkcji rozkładu prawdopodobieństwa $p(x)$. Wykorzystywany rozkład powinien być jak najbardziej związany z przybliżaną funkcją. Większa ilość próbek powinna być kierowana w obszary, w których $f(x)$ przyjmuje największe wartości, gdyż pozwala to szybciej zredukować błąd. Optymalnym wyborem, dającym zawsze błąd standardowy równy zero, jest oczywiście $p(x) = \frac{f(x)}{I}$. Wymaga on jednak znajomości poszukiwanej przez nas wartości całki.

¹Porównaj z równaniem 2.10.

W przypadku oświetlenia pośredniego obliczaną całką jest wartość radiancji odbitej z równania renderingu (2.13.). W celu jej wyznaczenia generowanych jest n promieni rekursywnie badających oświetlenie osiągające rozpatrywany punkt. Aby zapewnić dobrą zbieżność rozkład $p(x)$ musi być odpowiednio skonstruowany. Wybór odpowiedniej funkcji jest dość prosty w przypadku obiektów o charakterze zwierciadlanym. Aby znacząco zawęzić obszar poszukiwań wystarczy związać ją z funkcją BRDF, która dla materiałów zwierciadlanych przyjmuje wysokie wartości tylko w niewielkim przedziale. Niestety, strategia ta nie sprawdza się dla odbicia rozproszonego, gdyż jego BRDF jest identyczny we wszystkich kierunkach. W tym przypadku optymalnym rozwiązaniem byłoby związanie rozkładu z promieniowaniem świetlnym, które niestety nie jest znane². Do zbudowania suboptymalnego rozkładu można się natomiast posłużyć przybliżeniem promieniowania zawartym w mapie fotonów.

W celu wzięcia pod uwagę rozkładu oświetlenia w czasie renderowania punktu x odnajdywanych jest k najbliższych mu fotonów reprezentujących oświetlenie pośrednie. Każdy foton p przenosi cząstkę promieniowania $\Delta\Phi_{i,p}$ z kierunku $\vec{\omega}_p$. Jeśli założyć, że wszystkie odnalezione fotony trafiają w punkt x , to wkład każdego z nich w promieniowanie odbite wynosi:

$$\Delta\Phi_r = f_r(\vec{\omega}_p) \Delta\Phi_{i,p}(\vec{\omega}_p). \quad (6.4.)$$

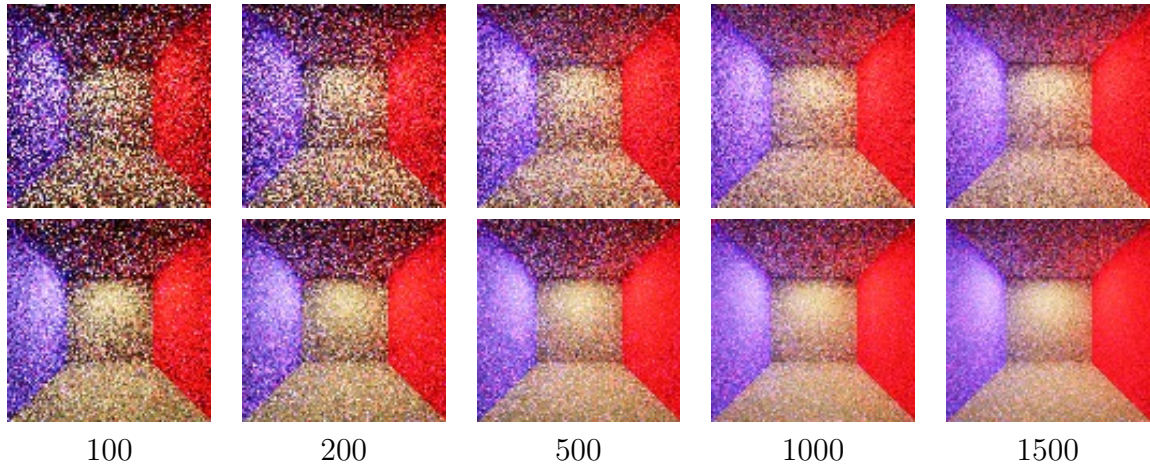
Zbiór odszukanых fotonów zawiera więc przybliżenie rozkładu oświetlenia padającego na rozpatrywany punkt. Informację tą odzyskuje się za pomocą następującej konstrukcji. Jednostkowy kwadrat dzielony jest na $m \times n$ równych obszarów (m wierszy i n kolumn). Każdy punkt wewnątrz tego kwadratu odpowiada dokładnie jednemu kierunkowi, z jakiego może pochodzić foton. Kierunki mapowane są na punkty wewnątrz kwadratu przy pomocy specjalnego, odwracalnego przekształcenia $T(\theta, \phi)$. Każdemu fotonowi p odpowiada dokładnie jeden punkt $(x_p, y_p) = T(\theta_p, \phi_p)$. Energia każdego z odnalezionych fotonów akumulowana jest w obszarze wyznaczonym przez wiersz i kolumnę, do której należy jego punkt (x_p, y_p) . W rezultacie poszczególne obszary kwadratu zawierają przybliżenie energii pochodzącej z pewnego zbioru kierunków.

W kolejnym kroku, na podstawie danych zgromadzonych w kwadracie, tworzona jest jednowymiarowa, dyskretna funkcja rozkładu, w której prawdopodobieństwo wyboru konkretnego regionu jest wprost proporcjonalne do ilości energii w nim zgromadzonej. Wynikiem wykorzystania tej funkcji w połączeniu z odpowiednim mapowaniem wylosowanego punktu na kierunek jest wysyłanie promieni w obszary mocniej oświetlone.

W celu wyznaczenia pojedynczej próbki (promienia) najpierw, za pomocą opisanej w poprzednim akapicie funkcji, ustalany jest odpowiedni region wewnątrz kwadratu. Następnie wybierany jest wewnątrz niego z rozkładem jednorodnym punkt. Punkt ten zamieniany jest na odpowiedni kierunek przy pomocy przekształcenia T^{-1} . Dla powierzchni o charakterze czysto rozproszonym przekształcenie to powinno mieć następującą postać:

$$T(\theta, \phi) = (\cos^2 \theta, \frac{\phi}{2\pi}), \quad T^{-1}(x, y) = (\arccos(1 - x), 2\pi y). \quad (6.5.)$$

²Dokładniej rzecz ujmując, właśnie jest wyznaczane.



Rysunek 6.1.: Porównanie tradycyjnego śledzenia ścieżek (górny rząd), z śledzeniem wykorzystującym mapę fotonów do kierowania promieni (dolny rząd). Liczby pod obrazami oznaczają wykorzystaną ilość ścieżek na piksel.

Z powodu niejednorodnego próbkowania radiancja otrzymana w wyniku śledzenia tak wybranego promienia musi być przeskalowana przez następujący czynnik:

$$s = \frac{E_T}{E_R * m * n}, \quad (6.6.)$$

gdzie:

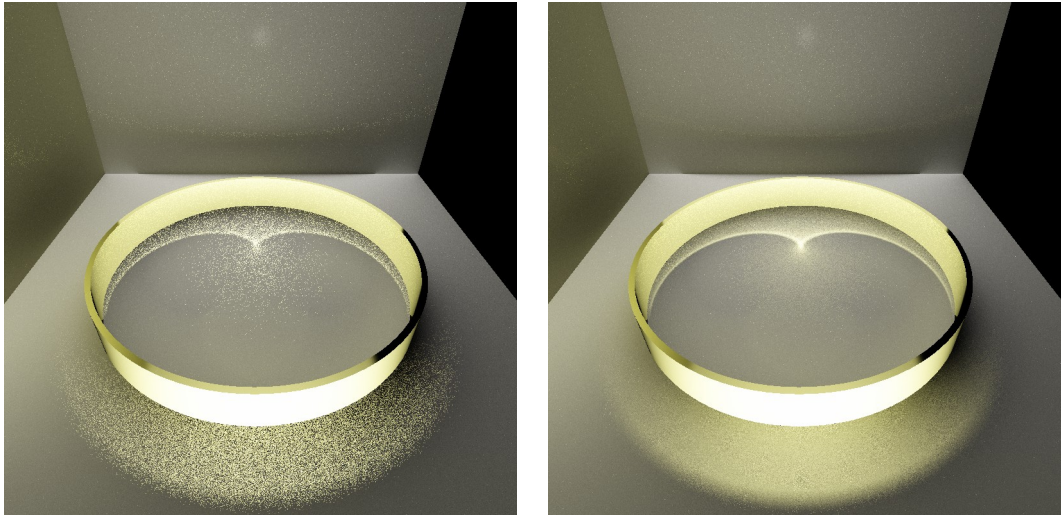
- E_T - całkowita energia zgromadzona w kwadracie,
- E_R - energia wewnątrz wybranego regionu.

6.2.2. Przykłady

Pierwsza scena testowa to pusty sześcián, którego wnętrze jest oświetlone przy pomocy jednego, niewielkiego źródła światła. Lampa jest umieszczona bardzo blisko górnej ściany i emituje światło jedynie w jej kierunku. Dzięki takiej konfiguracji większość modelu jest oświetlona tylko poprzez światło odbite. Wszystkie ściany odbijają światło w sposób idealnie rozproszony. Lewa ściana jest niebieska, prawa czerwona, a pozostałe są białe. Układ ten powoduje silne krwawienie kolorów na sąsiadujących powierzchniach o różnym kolorze.

Celem testów przeprowadzonych na opisanej scenie było zbadanie poprawy jakości, jaką można uzyskać przy pomocy opisanego algorytmu. Rysunek 6.1. przedstawia porównanie tradycyjnego śledzenia ścieżek (górny rząd), z śledzeniem, w którym promienie kierowane są przy pomocy mapy fotonów.

Na podstawie obrazów można wywnioskować, że opisana metoda zapewnia znaczną poprawę zbieżności w przypadku scen o dość jednorodnym rozkładzie oświetlenia.



Rysunek 6.2.: Kaustyka uzyskana przy pomocy tradycyjnego śledzenia ścieżek (po lewej) i algorytmu wykorzystującego mapę fotonów do kierowania ścieżek (po prawej).

Uzyskanie podobnych wyników przy pomocy algorytmu tradycyjnego wymaga wykorzystania ponad trzykrotnie większej liczby ścieżek³.

Opisana metoda posiada trzy parametry. Są to rozmiar mapy fotonów, ilość fotonów wykorzystywanych do utworzenia funkcji rozkładu oraz gęstość podziału półsfery. Testy wykazały, że mapa fotonów o niskiej gęstości (taka, jak ta opisywana w paragrafie 6.4.1.) w zupełności wystarcza do zbudowania dobrej funkcji rozkładu we wszystkich punktach sceny. Podział półsfery na cztery komórki wzdłuż kierunku θ , oraz 16 wzdłuż ϕ zapewnia odpowiednią ziarnistość utworzonego rozkładu. Ilość fotonów wykorzystywanych do utworzenia rozkładu powinna oscylować pomiędzy 50–100. Nie powinna być ona mniejsza niż łączna ilość komórek, na jakie podzielona została półsfera. Przy zastosowaniu tych parametrów czas działania rozszerzonej metody jest około 30% dłuższy w porównaniu z tradycyjnym śledzeniem ścieżek. Jeśli jednak wziąć pod uwagę wspomnianą w poprzednim akapicie poprawę jakości, rozszerzony algorytm okazuje się być znacznie szybszy.

Drugi z przeprowadzonych testów posłużył do zbadania, jak prezentowany algorytm zachowuje się w przypadku scen zawierających obszary o bardzo niejednorodnym rozkładzie oświetlenia pośredniego. Wykorzystany model składa się z odbijającego światło w sposób zwierciadlany pierścienia, oświetlonego z przodu przez lampę o dużej mocy. W efekcie na znajdującej się pod nim powierzchni powstaje bardzo silna kaustyka. To właśnie obszar kaustyki charakteryzuje się bardzo ukierunkowanym oświetleniem pośrednim, pochodzącym przede wszystkim z odbicia lampy w pierścieniu.

Rysunek 6.2. zawiera wyniki renderingu uzyskane przy pomocy tradycyjnego śledze-

³Porównaj obraz dla 100 i 500 ścieżek z dolnego rzędu odpowiednio z obrazem dla 500 i 1500 ścieżek z górnego rzędu.

6.2. Przyspieszanie zbieżności śledzenia ścieżek

nia ścieżek oraz przy wykorzystaniu mapy fotonów do wyboru kierunków generowanych promieni. Dla zwiększenia czytelności przedstawiono jedynie oświetlenie pośrednie. Oba obrazy utworzono z użyciem 1500 ścieżek/piksel.

Jak widać różnice pomiędzy otrzymanymi wynikami są olbrzymie. W przypadku tradycyjnego śledzenia ścieżek wszystkie widoczne kaustyki dopiero zaczęły się zarysowywać. Zarówno ich rogi, jak i wnętrza, są bardzo nieregularne i postrzępione. Zauważalna jest też znaczna ilość szumu w okolicy środka lewej ściany oraz na ścianie tylnej powyżej odbicia pierścienia. W rzeczywistości w tych miejscach powinna pojawić się bardzo wąska kaustyka pochodząca z odbicia światła w górnej ścianie pierścienia. Wyeliminowanie opisanych niedoskonałości wymagałoby użycia o wiele większej ilości ścieżek.

Kaustyki uzyskane przy pomocy wprowadzonego algorytmu są o wiele lepsze. Mniejsza, znajdująca się wewnątrz pierścienia, ma bardzo wyraźnie zarysowane brzegi z mocnym rozbłyskiem pośrodku. Zewnętrzna jest także bardzo wyraźna, a szum w jej obszarze o wiele mniejszy. Szum na ścianach jest też znacznie słabszy. Niestety, tu nie pojawiła się kaustyka. Oznacza to, że nawet rozszerzona metoda śledzenia ścieżek nie jest w stanie szybko i dokładnie oddawać wszystkich zjawisk świetlnych.

Przykład ten pokazuje, jak znaczący wpływ na ostateczny wynik ma właściwy dobór kierunków śledzenia promieni. Algorytm wykorzystujący mapę fotonów okazał się znakomicie spełniać swoje zadanie, umożliwiając nawet odtworzenie niektórych kaustyk, co w tradycyjnym śledzeniu ścieżek jest praktycznie nieosiągalne. Niestety, uzyskanie kaustyki o akceptowalnej jakości jest ciągle zbyt kosztowne czasowo i wiąże się z koniecznością stosowania antyaliasingu w celu wyeliminowania postrzępionych brzegów. Poza tym, niektóre bardzo małe kaustyki są ciągle niemożliwe do zwizualizowania. Z tych powodów renderowanie skomplikowanych elementów oświetlenia globalnego warto pozostawić o wiele wydajniejszym metodom wyspecjalizowanym, a opisany algorytm stosować jedynie do poprawiania zbieżności śledzenia ścieżek.

6.2.3. Podsumowanie

Zaprezentowana w tym podrozdziale metoda wykorzystania mapy fotonów do przyspieszania zbieżności śledzenia ścieżek okazała się być bardzo skuteczna. Jej zastosowanie zdecydowanie poprawia rozkład generowanych promieni, kierując ich większą ilość w obszary silnie oświetlone. Pozwala to znacznie zmniejszyć szum, a także zdecydowanie poprawić jakość wizualizacji zjawisk takich jak krwawienie kolorów czy kaustyki. Zwiększony koszt związany z wyszukiwaniem fotonów i tworzeniem odpowiednich funkcji rozkładu jest niewielki, a jeśli wziąć pod uwagę uzyskiwaną poprawę jakości, algorytm rozszerzony jest znacznie szybszy. Dodatkowym atutem metody jest jej prostota i przezroczystość. Może ona być w łatwy sposób zaimplementowana jako specjalna funkcja BRDF.

6.3. Wyznaczanie radiancji

Radiancja opuszczająca punkt jest najważniejszą z wartości obliczanych w algorytmach oświetlenia globalnego. Kolejne paragrafy tego podrozdziału zawierają opis metody wyznaczania radiancji na podstawie mapy fotonów (6.3.1.), sposoby poprawiania otrzymywanych rezultatów poprzez modyfikację obszaru wyszukiwania fotonów (6.3.2.) oraz filtrowanie (6.3.3.). Opisano także prostą metodę automatycznego doboru optymalnego rozmiaru obszaru wyszukiwania fotonów (6.3.4.).

6.3.1. Estymacja radiancji

Mapa fotonów jest reprezentacją rozkładu promieniowania świetlnego wewnątrz sceny. Każdy z fotonów przenosi część tego promieniowania. Punkty, w które uderzają fotony, oznaczają miejsca odbierające promieniowanie. Na podstawie pojedynczego fotonu nie można jednak ustalić, jak dużo światła pada na dany punkt. Wartość tę oszacowuje się za pomocą gęstości fotonów w okolicy punktu.

Dokładniej — jak już wspomniano w paragrafie 2.4.3., radiancję opuszczającą punkt x w kierunku $\vec{\omega}$ opisuje następujący wzór:

$$L_r(x, \vec{\omega}) = \int_{\Omega} f_r(x, \vec{\omega}', \vec{\omega}) L_i(x, \vec{\omega}') (\vec{\omega}' \cdot \vec{n}_x) d\vec{\omega}'.$$

Rozwiązanie powyższej zależności wymaga znajomości radiancji padającej na rozpatrywany punkt. Mapa fotonów reprezentuje jedynie promieniowanie świetlne wewnątrz sceny i nie przechowuje bezpośrednio żadnej informacji na temat radiancji. Istnieje jednak prosta zależność pomiędzy radiancją padającą a promieniowaniem. Wyraża się ona następującym wzorem:

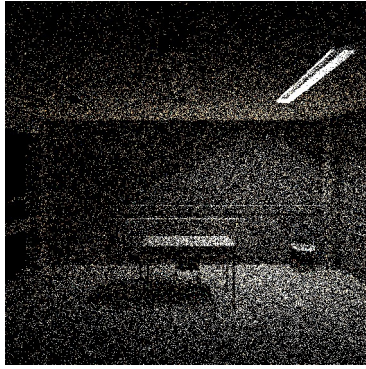
$$L_i(x, \vec{\omega}') = \frac{d^2\Phi_i(x, \vec{\omega}')}{(\vec{n}_x \cdot \vec{\omega}') d\vec{\omega}' dA_i}. \quad (6.7.)$$

Stąd nowa postać radiancji opuszczającej punkt:

$$\begin{aligned} L_r(x, \vec{\omega}) &= \int_{\Omega_x} f_r(x, \vec{\omega}', \vec{\omega}) \frac{d^2\Phi_i(x, \vec{\omega}')}{(\vec{n}_x \cdot \vec{\omega}') d\vec{\omega}' dA_i} (\vec{\omega}' \cdot \vec{n}_x) d\vec{\omega}' \\ &= \int_{\Omega_x} f_r(x, \vec{\omega}', \vec{\omega}) \frac{d^2\Phi_i(x, \vec{\omega}')}{dA_i}. \end{aligned} \quad (6.8.)$$

Promieniowanie padające jest przybliżane za pomocą pewnej określonej liczby fotonów (k) najbliższych punktowi x . Każdy foton p ma moc $\Delta\Phi_p(\vec{\omega}_p)$. Zakładając, że wszystkie wybrane fotony trafiają w punkt x otrzymujemy:

$$L_r(x, \vec{\omega}) \approx \sum_{p=1}^k f_r(x, \vec{\omega}_p, \vec{\omega}) \frac{\Delta\Phi_p(x, \vec{\omega}_p)}{\Delta A}. \quad (6.9.)$$



Mapa fotonów



Irradiancja



Radiancja

Rysunek 6.3.: Irradiancja i radiancja w prostym pokoju biurowym wyznaczone na podstawie mapy zawierającej 10^5 fotonów.

W powyższym równaniu pozostał jeszcze czynnik ΔA powiązany z gęstością fotonów w okolicy punktu x . Przyjmując, że powierzchnia wokół tego punktu jest płaska, ΔA można wyznaczyć jako pole koła powstałego poprzez zrzutowanie najmniejszej sfery zawierającej wszystkie fotony:

$$\Delta A = \pi r^2,$$

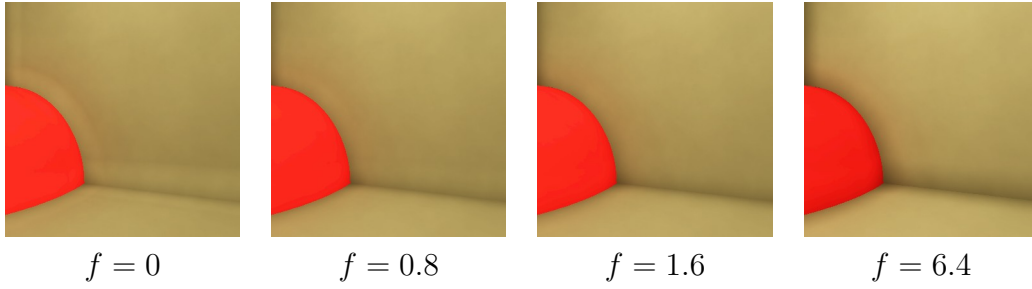
gdzie r jest promieniem tej sfery, czyli odległością do najdalszego z uwzględnionych w oszacowaniu fotonów. Ostateczny kształt wzoru przybliżającego radiancję jest więc następujący:

$$L_r(x, \vec{\omega}) \approx \frac{1}{\pi r^2} \sum_{p=1}^k f_r(x, \vec{\omega}_p, \vec{\omega}) \Delta \Phi_p(x, \vec{\omega}_p). \quad (6.10.)$$

Zaprezentowana metoda estymacji radiancji jest oparta na bardzo wielu założeniach. Jej jakość uzależniona jest od tego, jak bardzo są one prawdziwe. Największy wpływ na dokładność ma ilość fotonów znajdujących się w mapie oraz biorących udział w oszacowaniu. W pracy [15] wykazano, że wykorzystując opisany powyżej algorytm, przy zaniedbaniu niedokładności reprezentacji pozycji i kierunku fotonu, mocy strumienia oraz zwiększając liczbę fotonów w mapie i oszacowaniu do nieskończoności, jesteśmy w stanie otrzymać dowolnie dobre przybliżenie wyniku dokładnego i to wykorzystując w oszacowaniach radiancji nieskończenie mniej fotonów, niż jest ich w mapie.

6.3.2. Kształt obszaru wyszukiwania fotonów

Z powodu wydajności w rzeczywistych implementacjach liczba fotonów, szczególnie tych wykorzystywanych w oszacowaniu, musi być utrzymana na jak najniższym poziomie. W tej sytuacji znaczący wpływ na jakość estymacji ma kształt obszaru, z którego zbierane są fotony. W przytoczonej w poprzednim paragrafie postaci, algorytm



Rysunek 6.4.: Obraz styku ściany i podłogi przy różnych współczynnikach kompresji.

estymacji radiancji wykorzystuje do tego celu sferę, dodając po prostu najbliższe w sensie euklidesowskim fotony. Niewątpliwą zaletą tego rozwiązania jest prostota obliczeń. Wyznaczenie odległości fotonu od rozważanego punktu, jak i powierzchni obszaru rzutowania, jest w tym przypadku bardzo łatwe i szybkie. Niestety, zastosowanie sfery może prowadzić do powstawania dużych przekłamań wszędzie tam, gdzie dwa lub więcej obiektów znajduje się bardzo blisko siebie. Dobrym przykładem takiej sytuacji są punkty na styku ściany i podłogi. W tych obszarach założenie o lokalnej płaskości powierzchni wokół rozpatrywanego punktu przestaje być prawdziwe, gdyż część fotonów znajdujących się na ścianie przenika do oszacowań irradiancji wykonywanych dla podłogi, co powoduje przekłamanie.

Rozwiązaniem tego problemu jest modyfikacja kształtu obszaru wyszukiwania fotonów. Jednym z prostszych, a jednocześnie dających bardzo dobre wyniki sposobów, jest kompresja sfery w kierunku normalnej do powierzchni w rozważanym punkcie. Efektem tego zabiegu jest elipsoida, która rośnie znacznie szybciej w kierunku prostopadłym do normalnej, niż w kierunku z nią zgodnym. Dzięki temu dołączane są przede wszystkim fotony znajdujące się na tej samej płaszczyźnie, co rozpatrywany punkt. Zapobiega to przenikaniu fotonów z sąsiednich powierzchni.

Opisaną wyżej kompresję można uzyskać poprzez zastąpienie w czasie wyszukiwania fotonów zwykłej odległości euklidesowskiej wynikiem otrzymanym przy pomocy następującego wzoru:

$$d_{pn} = d_p + f d_p |\cos(\vec{n}_x, x \rightarrow p)|, \quad (6.11.)$$

gdzie:

- d_p – euklidesowska odległość fotonu p od punktu x ,
- d_{pn} – zmodyfikowana odległość fotonu p od punktu x ,
- f – współczynnik kompresji, $f \geq 0$.

Zaletą powyższego wzoru jest prostota i duża skuteczność. W porównaniu ze sferą wymagane jest dodatkowo wyliczenie kąta pomiędzy dwoma wektorami oraz normalizacja wektora w kierunku fotonu ($x \rightarrow p$). Przy odpowiedniej kolejności obliczeń wystarczy osiem dodatkowych mnożeń zmiennopozycyjnych, trzy dzielenia, pięć dodawań i jeden

pierwiastek kwadratowy. Czynniki ΔA odpowiedzialny za gęstość fotonów we wzorze na oszacowanie radiancji (6.9.) pozostaje bez zmian, gdyż rzut obszaru wyszukiwania nadal jest kołem.

Jedynym parametrem metody jest współczynnik kompresji f . Dla $f = 0$ kompresja nie jest wykonywana, a obszarem wyszukiwania fotonów jest kula. Dla $f \neq 0$ stopień kompresji rośnie wprost proporcjonalnie do wzrostu współczynnika.

Rysunek 6.4. prezentuje wymieniony wcześniej przykład styku ściany i podłogi. Został on wygenerowany za pomocą mapy fotonów przy wykorzystaniu kompresji różnego stopnia. Przy $f = 0$ (brak kompresji) wyraźnie widoczne są jaśniejsze obszary w okolicach styku dwóch powierzchni. Są one efektem przeszacowania wartości radiancji w tej okolicy. Artefakt ten znacznie traci na sile nawet przy wykorzystaniu niewielkiego współczynnika kompresji i już dla $f = 0.8$ jest prawie niewidoczny. Dla $f = 1.6$ błąd został całkowicie wyeliminowany. Dalsze zwiększanie współczynnika kompresji jest bezcelowe, a może prowadzić do problemów z odnalezieniem odpowiedniej liczby fotonów, co z kolei powoduje niedoszacowanie radiancji ($f = 6.4$). Wspomniane problemy z odnalezieniem fotonów wynikają z błędów reprezentacji ich pozycji.

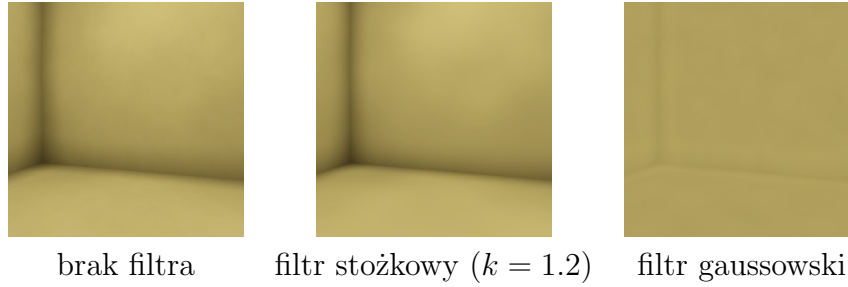
Narzut czasowy związany z wyliczaniem odległości przy pomocy wzoru 6.11. jest minimalny, jednak trzeba się liczyć z dodatkowymi kosztami spowodowanymi przeszukiwaniem większych obszarów kd-drzewa. Rosną one wraz ze wzrostem współczynnika kompresji, szczególnie na powierzchniach krzywoliniowych.

Innym kształtem dającym podobne do elipsoidy wyniki przy porównywalnym koszcie obliczeniowym jest cylinder. Niektórzy autorzy proponują także wykorzystanie płaskiego dysku, lecz sprawuje się on bardzo źle na powierzchniach krzywoliniowych (np.: kulach), gdzie praktycznie nie ma fotonów leżących na jednej płaszczyźnie. Istnieją także bardziej zaawansowane techniki, jak na przykład ta opisana w pracy [10], w której jako obszar wyszukiwania stosowany jest specjalnie skonstruowany prostopadłościan. Są one jednak znacznie kosztowniejsze obliczeniowo, a osiągnięta poprawa jakości nie rekompensuje poniesionego nakładu czasowego. Powinny one być stosowane jedynie w aplikacjach wyspecjalizowanych, wymagających bardzo wysokiej dokładności uzyskiwanych przybliżeń.

6.3.3. Filtrowanie

Estymacja gęstości fotonów bazująca na polu najmniejszego koła, w którym się one znajdują, daje w większości przypadków wystarczająco dobre rezultaty. Niestety, metoda ta powoduje niedoszacowanie gęstości w pobliżu krawędzi powierzchni, gdyż w takich miejscach fotony pokrywają jedynie część wyznaczonego obszaru. W efekcie okolicę krawędzi są znacznie ciemniejsze niż reszta powierzchni. Dobrym, lecz zbyt kosztownym rozwiązaniem tego problemu byłoby wykorzystanie zamiast koła otoczki wypukłej wszystkich fotonów. Innym, równie skutecznym, a jednocześnie szybkim narzędziem eliminującym ten problem są filtry.

Filtry znajdują także zastosowanie wszędzie tam, gdzie zbyt niska gęstość mapy fotonów powoduje rozmazywanie estymacji radiancji na granicy obszarów o różnej cha-



Rysunek 6.5.: Porównanie wyników renderingu bez i z użyciem filtra.

rakterystyce oświetlenia (np.: na granicy kaustyki).

Zadaniem filtra jest przypisanie wagi do każdego fotonu wykorzystywanego w oszacowaniu radiancji. Jest ona dobierana tak, aby fotony znajdujące się blisko jądra (punktu, dla którego ustalana jest radiancja) miały wkład większy niż te, które znajdują się w dużej odległości. Stosowane filtry są dwuwymiarowe, gdyż zarówno estymacja gęstości, jak i powierzchnie, na których zachowywane są fotony, są dwuwymiarowe.

Filtr stożkowy

Filtr ten bazuje na odległości d_p fotonu od punktu, dla którego wyznaczana jest radiancja. Wyraża się on wzorem:

$$w_{ps} = 1 - \frac{d_p}{kr}. \quad (6.12.)$$

k jest parametrem filtra zachowującym zależność $k \geq 1$, a r odległością najdalszego z wykorzystywanych w estymacji fotonów od punktu x . Zastosowanie tego filtra wymaga znormalizowania otrzymanego przybliżenia radiancji przez czynnik:

$$N_{ws} = 1 - \frac{2}{3k}. \quad (6.13.)$$

Ostateczny kształt równania estymacji radiancji jest więc następujący:

$$L_r(x, \vec{\omega}) \approx \frac{\sum_{p=1}^k f_r(x, \vec{\omega}_p, \vec{\omega}) \Delta \Phi_p(x, \vec{\omega}_p) w_{ps}}{\pi r^2 N_{ws}}. \quad (6.14.)$$

Filtr gaussowski

W przypadku filtra gaussowskiego wykorzystuje się założenie o płaskości powierzchni w okolicy punktu x i stosuje prosty filtr gaussowski. Ma on następującą postać:

$$w_{pg} = \alpha \left(1 - \frac{1 - e^{-\beta \frac{d_p^2}{2r^2}}}{1 - e^{-\beta}} \right). \quad (6.15.)$$

W tym konkretnym zastosowaniu współczynniki α oraz β powinny mieć wartości odpowiednio 0.918 i 1.953. Filtr nie wymaga dodatkowej normalizacji, stąd następująca postać wzoru na estymację radiancji:

$$L_r(x, \vec{\omega}) \approx \sum_{p=1}^k f_r(x, \vec{\omega}_p, \vec{\omega}) \Delta\Phi_p(x, \vec{\omega}_p) w_{pg}. \quad (6.16.)$$

6.3.4. Automatyczne wyznaczanie obszaru wyszukiwania

Jak już wspomniano w paragrafie 6.3.1., ustalenie irradiancji, jak i radiancji odbitej, na podstawie mapy fotonów, odbywa się za pomocą estymacji gęstości fotonów w okolicy rozpatrywanego punktu (x). Dane dla tej estymacji przygotowywane są poprzez odnalezienie co najwyżej k fotonów najbliższych punktowi x i znajdujących się w odległości nie większej niż r od tego punktu. Pierwsze z ograniczeń pozwala uniknąć niepotrzebnego rozważania zbyt wysokiej liczby fotonów w obszarach, gdzie mapa ma bardzo dużą gęstość. Drugie zapobiega z kolei przeszukiwaniu olbrzymich partii kłdrzewa w okolicach, gdzie gęstość mapy jest niska.

Maksymalna ilość rozpatrywanych fotonów jest dość prosta do ustalenia. Zazwyczaj użycie około 100 fotonów w przypadku zbierania końcowego (*ang. final gather*) oraz 500 przy bezpośredniej wizualizacji mapy daje zadowalające wyniki. Niestety, dobranie drugiego z parametrów nie jest tak łatwe. Zależy on między innymi od gęstości mapy, rozkładu fotonów oraz oczywiście pożądanej jakości. Im mniejszy jest maksymalny przeszukiwany obszar, tym algorytm jest szybszy. Jednak zbyt mała wartość powoduje powstanie charakterystycznej mozaiki jaśniejszych i ciemniejszych punktów, będącej efektem kiepskiej estymacji. Rozwiązaniem problemów z doбором rozmiaru wyszukiwania fotonów jest formuła, która wyznacza go automatycznie.

Niewielka ilość fotonów rozproszonych na dużym obszarze ma mały wkład w radiancję odbitą, gdyż ich gęstość jest znikoma. Z tego powodu maksymalny rozmiar obszaru poszukiwania fotonów powinien być dobrany tak, aby gęstość fotonów była zawsze utrzymana co najmniej na pewnym minimalnym poziomie. Niech więc r_m będzie promieniem maksymalnego obszaru wyszukiwania odpowiadającym ustalonej minimalnej gęstości, a L_t uzyskiwaną przy nim radiancją. Niech ponadto P_m będzie maksymalną mocą fotonu zachowanego w mapie. Przyjmując pole koła jako współczynnik określający gęstość fotonów ($A = \pi r^2$) oraz ograniczając moc dowolnych k fotonów z mapy przez kP_m , a wartość funkcji BRDF dla odbicia rozproszonego przez $1/\pi$, otrzymujemy następujące ograniczenie górne na wartość radiancji odbitej, wyliczonej na podstawie mapy fotonów:

$$L_r = \frac{\sum_{p=1}^k f_r(x, \vec{\omega}_p, \vec{\omega}) P_p}{A} \leq \frac{kP_m}{\pi^2 r^2}. \quad (6.17.)$$

Oszacowanie L_t przyjmuje więc wartość:

$$L_t = \frac{kP_m}{\pi^2 r_m^2}. \quad (6.18.)$$

Z tej zależności można wyznaczyć maksymalny promień wyszukiwania fotonów:

$$r_m = \frac{1}{\pi} \sqrt{\frac{k P_m}{L_t}}. \quad (6.19.)$$

Pozostaje jeszcze ustalenie wartości L_t , jednak jest to zadanie dość proste. Dobrym wyborem jest na przykład 5% maksymalnej wyświetlanej wartości koloru.

6.4. Renderowanie oświetlenia globalnego

Do tego miejsca w pracy został zaprezentowany sposób tworzenia mapy fotonów. Pokazano też wiele metod jej wykorzystania. W tym podrozdziale przedstawione są dwa algorytmy używające bezpośrednio każdej z opisywanych dotychczas metod w celu renderowania obrazów zawierających wszystkie elementy oświetlenia globalnego.

6.4.1. Bezpośrednia wizualizacja mapy fotonów

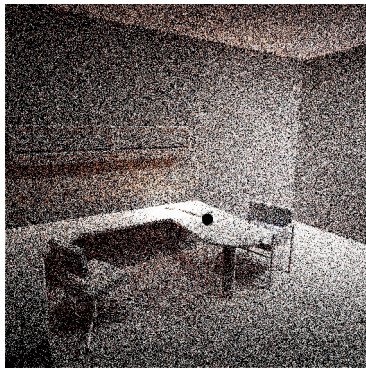
Jak już wspomniano, mapa fotonów jest przybliżeniem pełnej informacji o rozkładzie oświetlenia wewnątrz sceny. Na jej podstawie, przy pomocy algorytmu opisanego w paragrafie 6.3.1., można wyznaczyć radiancję opuszczającą dla dowolnego punktu sceny. Odpowiednie zastosowanie tego algorytmu wystarcza do zbudowania programu tworzącego realistyczne obrazy.

Algorytm

Opisywane rozwiązanie opiera się na prostym programie śledzącym promienie. Tak jak w metodzie tradycyjnej (podrozdz. 3.1.), dla każdego punktu obrazu generowany jest promień pierwotny, przechodzący przez ten punkt i skierowany w stronę wnętrza sceny. W punkcie przecięcia tego promienia ze sceną podejmowana jest jedna z dwóch akcji. Jeśli napotkana powierzchnia odbija światło w sposób rozproszony, to na podstawie mapy fotonów wykonywana jest estymacja radiancji w tym punkcie. Jeśli natomiast ma ona charakter zwierciadlany, to w celu wyznaczenia radiancji śledzony jest promień wtórny. Oczywiście na powierzchniach o charakterystyce mieszanej wykonywane są obydwie wymienione operacje. Ponadto, jeśli odbicie ma charakter błyszczący, lecz nie jest czysto zwierciadlane, można wykorzystać kilka promieni wtórnych dla osiągnięcia lepszego efektu.

Mimo swojej prostoty algorytm pokrywa wszystkie ścieżki, jakimi może przemieszczać się światło. Śledzenie promieni odpowiada za ścieżki opisywane przez wyrażenie $(LS^*E) \mid (DS^*E)$, obsługując samodzielnie źródła światła widoczne bezpośrednio oraz przez kilka odbić zwierciadlanych. Natomiast połączenie z mapą fotonów, zawierającą ścieżki $L(S \mid D)^*D$, pozwala we wszystkich punktach poprawnie zrenderować światło widziane przez przynajmniej jedno odbicie rozproszone.

Dzięki wykorzystaniu mapy fotonów algorytm jest bardzo wydajny. Dla większości pikseli generowanego obrazu śledzony jest dokładnie jeden promień. Jest to oczywiście



Mapa fotonów



Widok ogólny



Zbliżenie biurka

Rysunek 6.6.: Obraz biura utworzony metodą bezpośredniej wizualizacji mapy fotonów.

olbrzymia oszczędność w porównaniu z tysiącami rekursywnych promieni używanych w tradycyjnym śledzeniu ścieżek. Wykorzystanie rekursji do renderowania obiektów błyszczących nie ma większego znaczenia, gdyż takich obiektów jest zazwyczaj bardzo niewiele, a ilość promieni potrzebnych w tym przypadku nieporównywalnie mniejsza⁴.

Na rysunku 6.6. przedstawiono wyniki działania opisanego algorytmu. Zawiera on mapę fotonów oraz wygenerowane na jej podstawie dwa różne ujęcia tej samej sceny. Jest to zmodyfikowana wersja prezentowanego już wnętrza biura. Zmiany polegały na zastąpieniu kubka przez szklaną, mocno załamującą światło kulę, oraz nadaniu lekkiego połysku blatu biurka.

Niestety, wyniki renderingu mocno rozczarowują. Co prawda obrazy zawierają elementy oświetlenia globalnego, w szczególności sufit, na który nie pada bezpośrednio żadne światło, nie jest czarny, dostrzegalne jest krwawienie kolorów. Mimo to ogólne wrażenie jest fatalne. Najgorzej wypadły cienie, praktycznie ich nie ma, występują jedynie obszary trochę ciemniejsze. Rogi ścian są o wiele zbyt jasne, a brzegi biurka zbyt ciemne. Poza tym cały obraz charakteryzuje się szumem niskiej częstotliwości, który powoduje, że np.: ściany sprawiają wrażenie brudnych.

Na szczęście wszystkie opisane niedociągnięcia wiążą się z prezentowanymi już wcześniej problemami i można je w łatwy sposób wyeliminować.

Ulepszenia algorytmu

Kiepskiej jakości cienie są wynikiem renderowania oświetlenia bezpośredniego poprzez estymację radiancji wprost z mapy fotonów. Dokładniej o problemach z tym związanych mówi podrozdział 5.2. Potrzebna jest więc następująca modyfikacja algorytmu. W czasie estymacji radiancji należy wykorzystywać jedynie fotony pośrednie. W ten sposób pokrywane są następujące ścieżki: $L(S|D)+P$, gdzie P oznacza punkt, w którym wykonywana jest estymacja. Pozostaje jeszcze dołączenie oświetlenia bezpośredniego

⁴Np.: w przypadku odbicia czysto zwierciadlanego wystarczy dokładnie jeden promień.

6.4. Renderowanie oświetlenia globalnego



Widok ogólny

Zbliżenie biurka

Rysunek 6.7.: Obraz biura utworzony poprawioną metodą bezpośredniej wizualizacji.

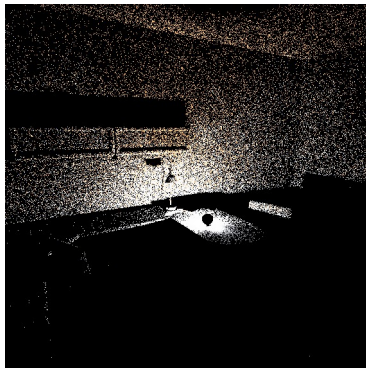
(ścieżki LP). Wykonuje się je przez wykorzystanie osobnej, wyspecjalizowanej metody, która zapewnia wysoką jakość generowanych cieni. Dobrym rozwiązaniem jest mapa cienia w połączeniu ze śledzeniem promieni w obszarach półcienia. Dodatkowy koszt związany z zastosowaniem osobnej metody do generowania oświetlenia bezpośredniego jest minimalny, gdyż algorytm praktycznie nie śledzi promieni rekursywnych.

Wykorzystanie osobnej metody do generowania oświetlenia bezpośredniego powoduje, że fotony bezpośrednie stają się bezużyteczne. W związku z tym nie ma potrzeby zachowywania ich w mapie. Fotonów bezpośrednich nie trzeba więc zapisywać, a wolne miejsce można wykorzystać na dodatkowe fotony pośrednie.

Rozwiązaniem części problemów z niedokładnością generowanego oświetlenia pośredniego jest zastosowanie elipsoidy jako obszaru wyszukiwania fotonów (6.3.2.) oraz filtrowania (6.3.3.). Redukują one między innymi opisane wcześniej problemy ze zbyt dużą, lub zbyt małą jasnością niektórych obszarów oraz zmniejszają ogólny poziom szumu.

Rysunek 6.7. przedstawia obrazy wygenerowane z użyciem wszystkich opisanych udoskonaleń. Ich jakość jest o wiele wyższa. Oświetlenie bezpośrednie wyznaczone zostało z należytą dokładnością. Cienie są wyraźne i mają ostro zarysowane krawędzie. Większość pozostałych artefaktów także zniknęła. Powierzchnia biurka, podłogi i ścian jest jednolicie oświetlona. Górna część lampki nie jest już w dziwny sposób rozświetlona.

Jednak i ta wersja posiada wady. Oświetlenie na suficie nadal nie jest perfekcyjne. Widoczny na jego powierzchni szum niskiej częstotliwości jest cechą charakterystyczną obrazów generowanych przy pomocy mapy fotonów. Wynika on z niedokładności przybliżeń, i jako że sufit nie jest bezpośrednio oświetlony, nie znika pod warstwę mocniejszego światła bezpośredniego. Niestety, przy pomocy tego algorytmu nie można całkowicie wyeliminować szumu. Drugą usterką jest brak kaustyk. Model zawiera obiekty odbijające światło w sposób zwierciadlany, a mimo to nie można dostrzec żadnej kaustyki. Następny paragraf wskazuje przyczyny tego problemu i sposób jego rozwiązania.



Mapa kaustyczna



Widok ogólny



Zbliżenie biurka

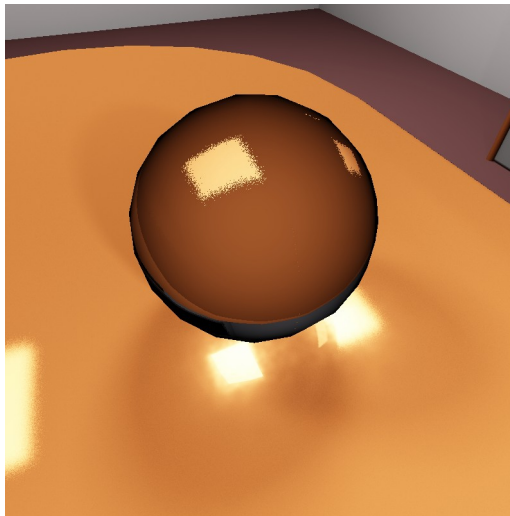
Rysunek 6.8.: Obraz biura z dodanymi kaustykami.

Generowanie kaustyk

W renderingu z wykorzystaniem mapy fotonów używa się map niskiej gęstości. Taka mapa jest bardzo atrakcyjna. Buduje się ją niezwykle szybko, zajmuje bardzo mało pamięci, a na dodatek w zupełności wystarcza do wygenerowania całkowicie poprawnego obrazu. Niestety, kaustyki tworzone na podstawie takiej mapy są nieprawidłowe. Zbyt mała gęstość powoduje, że są one zazwyczaj całkowicie niewidoczne, lub mają postrzępione rogi i niejednolite wnętrza. Jedynie w prostych scenach, zawierających niewiele więcej ponad obiekt kaustyczny, udaje się uzyskać satysfakcjonujące wyniki.

Z tych powodów do tworzenia kaustyk wykorzystuje się dodatkową, specjalnie do tego celu przeznaczoną, mapę kaustyczną. Mapa ta zawiera jedynie fotony kaustyczne, czyli takie, które pokonały ścieżki o wzorcu LS^*D . Mapę tą buduje się w osobnym przebiegu algorytmu poprzez śledzenie fotonów w kierunku obiektów zwierciadlanych. Dzięki takiemu sposobowi konstrukcji ma ona odpowiednio dużą gęstość, umożliwiającą dokładne generowanie kaustyk, a jednocześnie może być tworzona szybko. Oczywiście moc zachowywanych fotonów musi być przeskalowana tak, aby wziąć pod uwagę specyficzny sposób emisji światła.

Renderowanie dowolnego punktu sceny z wykorzystaniem nowego typu mapy przebiega następująco. Najpierw wykonywana jest próba wygenerowania kaustyki przy pomocy mapy kaustycznej. W tym celu wyszukuje się fotony w tej właśnie mapie. Aby zapobiec tworzeniu fałszywych kaustyk, należy wymagać odpowiedniej gęstości mapy wokół rozpatrywanego punktu. Zapewnić ją można poprzez określenie pewnej minimalnej liczby fotonów, która musi być znaleziona w bardzo niewielkim obszarze otaczającym rozpatrywany punkt. Kaustyka jest tworzona tylko i wyłącznie wtedy, kiedy gęstość mapy w okolicy punktu jest odpowiednio wysoka. Po próbie utworzenia kaustyki generowane jest, na podstawie zwykłej mapy, oświetlenie pośrednie. Ta mapa także zawiera fotony kaustyczne, powstałe w wyniku zwykłej emisji. Dlatego jeśli tworzenie kaustyki powiodło się, to aby nie generować jej ponownie, należy użyć jedynie fotonów niekaustycznych. Jeżeli jednak kaustyka nie była wygenerowana, to w celu zapewnie-



Rysunek 6.9.: Zbliżenie kaustyk na powierzchni biurka.

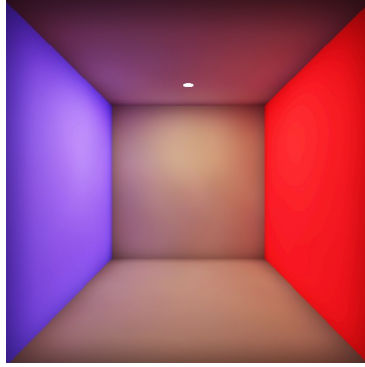
nia poprawności fizycznej konieczne jest dodanie fotonów kaustycznych znajdujących się w zwykłej mapie. Typ fotonu może być odzyskany z informacji zapisanych we flagach fotonu (podrozdz. 4.3.2.).

Rysunek 6.8. zawiera obrazy wygenerowane przy pomocy algorytmu rozszerzonego o tworzenie kaustyk. Po lewej stronie widoczna jest mapa kaustyczna. Bardzo duża liczba fotonów znajdujących się na ścianach i suficie jest konsekwencją błyszczącego charakteru biurka. Minimalna gęstość mapy została określona na 50 fotonów w kuli o promieniu 0.1m. Jak widać kaustyka została wygenerowana jedynie w okolicach szklanej kuli. W pozostałych miejscach mapa miała zbyt małą gęstość i tworzenie w nich kaustyk za pomocą specjalnej metody wprowadziłoby jedynie szum. W tych miejscach oświetlenie pochodzące ze ścieżek kaustycznych zostało dodane za pomocą zwykłej mapy fotonów. Utworzona kaustyka jest lekko rozmazana na rogach. Jest to wynikiem błyszczącego charakteru biurka.

Opisane rozszerzenie wymaga pewnych dodatkowych obliczeń, w szczególności w każdym punkcie konieczne jest wykonanie dwóch, a nie jak wcześniej jednego, wyszukiwań fotonów w mapie. W praktyce jednak narzut jest bardzo niewielki, gdyż poszukiwania w mapie kaustycznej ograniczają się jedynie do bardzo niewielkiego obszaru, który jest sprawdzany bardzo szybko.

Podsumowanie

W paragrafie tym opisano opartą na mapie fotonów, szybką i skuteczną metodę generowania realistycznych obrazów komputerowych zawierających wszystkie elementy oświetlenia globalnego. Przedstawiono też wady prezentowanego algorytmu. Część z nich udało się wyeliminować przez dodanie prostych rozszerzeń. Niestety, niektóre pozostały, co uzmysławia konieczność posiadania lepszego rozwiązania. nierozwiązane



Rysunek 6.10.: Oświetlenie pośrednie wewnątrz sześciangu.

problemy mogą się wydawać dyskwalifikującymi, jednak tak nie jest. Dowodem tego jest rysunek 6.10. Przedstawia ona pierwszą scenę testową z paragrafu 6.2.2. Jej obraz jest perfekcyjny, a utworzenie go zajęło jedynie kilkanaście sekund! Przyczyną tak różnego zachowania algorytmu jest stopień skomplikowania modelu. Zaprezentowane rozwiązanie jest dość proste i ma problemy w przypadku bardzo złożonych scen. Nadaje się natomiast idealnie do prostszych modeli, jak ten przedstawiony tutaj.

6.4.2. Algorytm dwufazowy

Wprowadzona w poprzednim paragrafie metoda bezpośredniej wizualizacji mapy fotonów nie sprawdza się w przypadku scen o bardzo skomplikowanym układzie oświetlenia. Zaprezentowany tutaj algorytm dwufazowy jest jej rozwinięciem, pozwalającym całkowicie wyeliminować wszystkie niedoskonałości.

Opisywane rozwiązanie składa się z dwóch etapów. Pierwszym z nich jest konstrukcja wszystkich potrzebnych map fotonów. W drugim kroku mapy te wykorzystywane są w renderingu, który opiera się na ograniczonym śledzeniu ścieżek. W czasie renderingu oświetlenie każdego punktu rozdzielane jest na kilka niezależnych składowych, z których każda jest traktowana osobno, przez najlepiej dobraną metodę. W kolejnych punktach zaprezentowano teoretyczne uzasadnienie podziału oświetlenia na składowe oraz sposób generowania każdej z nich.

Składowe oświetlenia

Przedstawione już w podrozdziale 2.5. równanie renderingu reprezentuje radiancję opuszczającą dowolny punkt w dowolnym kierunku jako sumę dwóch składowych, radiancji emitowanej przez punkt (L_e) oraz radiancji odbitej (L_r):

$$L_o(x, \vec{\omega}) = L_e(x, \vec{\omega}) + \underbrace{\int_{\Omega} f_r(x, \vec{\omega}', \vec{\omega}) L_i(x, \vec{\omega}') (\vec{\omega}' \cdot \vec{n}) d\vec{\omega}'}_{L_r(x, \vec{\omega})}. \quad (6.20.)$$

Wyznaczenie L_e jest proste i opiera się na właściwościach materiału. Wylczenie L_r jest natomiast o wiele trudniejsze, gdyż wiąże się w koniecznością przybliżenia wartości odpowiedniej całki. W celu podniesienia efektywności obliczeń warto skorzystać z własności funkcji BRDF opisującej materiał w rozważanym punkcie. Funkcja ta składa się zwykle z dwóch oddzielnych elementów. Jeden z nich jest odpowiedzialny za odbicie rozproszone ($f_{r,D}$), a drugi za błyszczące ($f_{r,S}$). Sposób renderowania obydwu komponentów jest diametralnie różny, dlatego warto rozważać funkcję BRDF jako sumę dwóch osobnych komponentów:

$$f_r(x, \vec{\omega}', \vec{\omega}) = f_{r,D}(x, \vec{\omega}', \vec{\omega}) + f_{r,S}(x, \vec{\omega}', \vec{\omega}). \quad (6.21.)$$

Podobnie radiancja padająca na punkt może być rozdzielona na sumę trzech niezależnych składników:

$$L_i(x, \vec{\omega}') = L_{i,L}(x, \vec{\omega}') + L_{i,C}(x, \vec{\omega}') + L_{i,D}(x, \vec{\omega}'), \quad (6.22.)$$

gdzie:

- $L_{i,L}(x, \vec{\omega}')$ – oznacza oświetlenie pochodzące bezpośrednio od źródeł światła,
- $L_{i,C}(x, \vec{\omega}')$ – oznacza oświetlenie pochodzące od promieni odbitych bądź załamanych jedynie w sposób zwierciadlany, czyli kaustyki,
- $L_{i,D}(x, \vec{\omega}')$ – oznacza oświetlenie pochodzące od promieni odbitych przynajmniej raz w sposób rozproszony.

Połączenie nowej postaci funkcji BRDF oraz radiancji padającej daje następujący podział radiancji odbitej na składowe:

$$\begin{aligned} L_r(x, \vec{\omega}') &= \int_{\Omega} f_r(x, \vec{\omega}', \vec{\omega}) L_i(x, \vec{\omega}') (\vec{\omega}' \cdot \vec{n}) d\vec{\omega}' \\ &= \int_{\Omega} f_r(x, \vec{\omega}', \vec{\omega}) L_{i,L}(x, \vec{\omega}') (\vec{\omega}' \cdot \vec{n}) d\vec{\omega}' + \\ &\quad \int_{\Omega} f_{r,S}(x, \vec{\omega}', \vec{\omega}) (L_{i,C}(x, \vec{\omega}') + L_{i,D}(x, \vec{\omega}')) (\vec{\omega}' \cdot \vec{n}) d\vec{\omega}' + \\ &\quad \int_{\Omega} f_{r,D}(x, \vec{\omega}', \vec{\omega}) L_{i,C}(x, \vec{\omega}') (\vec{\omega}' \cdot \vec{n}) d\vec{\omega}' + \\ &\quad \int_{\Omega} f_{r,D}(x, \vec{\omega}', \vec{\omega}) L_{i,D}(x, \vec{\omega}') (\vec{\omega}' \cdot \vec{n}) d\vec{\omega}'. \end{aligned} \quad (6.23.)$$

Pierwszy ze składników sumy jest odpowiedzialny za oświetlenie bezpośrednie, drugi za odbicia zwierciadlane i połyskujące. Trzeci dodaje kaustyki na powierzchniach o charakterze rozproszonym, a czwarty delikatne oświetlenie pośrednie pochodzące od światła odbitego w sposób rozproszony. Suma wszystkich elementów zawiera całkowitą informację o oświetleniu. Jak już wspomniano, każdy z odseparowanych składników jest wyznaczany i dodawany do oświetlenia każdego punktu osobno.

Mapy fotonów

W pierwszej fazie algorytmu tworzone są wszystkie wykorzystywane mapy fotonów. W zależności od sposobu generowania oświetlenia bezpośredniego potrzebne są dwie lub trzy mapy. Pierwszą z nich jest zwykła mapa fotonów. Przechowuje ona pełną informację o rozkładzie oświetlenia wewnątrz sceny i wykorzystywana jest przede wszystkim w czasie obliczania oświetlenia pochodzącego od promieni odbitych w sposób rozproszony. Ponadto może być użyta do wyznaczania oświetlenia bezpośredniego w punktach niewidocznych dla obserwatora oraz wykrywania zasłoniętych światel. Mapa kaustyczna, opisana w 6.4.1., jest drugą z tworzonych map. Wykorzystuje się ją do generowania kaustyk widzianych bezpośrednio przez obserwatora. Trzecią, tworzoną opcjonalnie, jest mapa cienia. Wykorzystuje się ją do przyspieszania obliczeń związanych z oświetleniem bezpośrednim.

Rendering

W drugiej fazie obraz jest generowany z wykorzystaniem wcześniej opisanych map oraz śledzenia ścieżek o ograniczonej głębokości rekursji. Kolor każdego piksela obrazu wyliczany jest poprzez śledzenie promienia z punktu, w którym znajduje się obserwator, poprzez piksel, w kierunku sceny. W punkcie przecięcia promienia ze sceną wyznaczana jest, na podstawie wzoru 6.23., radiancja opuszczająca. Obliczenia wykonywane są na jeden z dwóch sposobów. W punktach widzianych przez obserwatora bezpośrednio, lub za pośrednictwem kilku odbić zwierciadlanych, wszystkie składowe równania wyznaczane są bardzo dokładnie. W pozostałych punktach wykonywane są jedynie obliczenia przybliżone.

Oświetlenie bezpośrednie

Oświetlenie bezpośrednie wyliczane jest w następującym składniku sumy:

$$\int_{\Omega} f_r(x, \vec{\omega}', \vec{\omega}) L_{i,L}(x, \vec{\omega}') (\vec{\omega}' \cdot \vec{n}) d\vec{\omega}'.$$

Jak już wspomniano, ma ono zazwyczaj bardzo duży wkład w całkowitą radiancję odbitą od punktu, toteż musi być wyznaczone dokładnie. W przypadku punktów widzianych bezpośrednio przez obserwatora, lub poprzez odbicia zwierciadlane, wykonywane są obliczenia dokładne. Polegają one na wykorzystaniu odpowiedniej ilości promieni cienia i specjalnej techniki monte carlo. Oczywiście w celu przyspieszenia obliczeń można wykorzystać wszystkie opisane w rozdziale piątym rozszerzenia, jak wykrywanie światel niewidocznych i mapa cienia. W pozostałych punktach oświetlenie bezpośrednie wyliczane jest poprzez dołączenie fotonów bezpośrednich do oszacowania radiancji.

Rysunek 6.11. zawiera przykład ukazujący oświetlenie bezpośrednie wewnątrz pokoju obliczone metodą dokładną i przybliżoną. Różnice między obrazami są dość wyraźne, jednak obydwa rozwiązania spełniają swoje zadanie. Wersja dokładna w sposób



Wynik dokładny



Wynik przybliżony

Rysunek 6.11.: Oświetlenie bezpośrednie we wnętrzu pokoju dziennego.

bardzo precyzyjny odzwierciedla układ oświetlenia bezpośredniego, dodając obrazowi realizmu. Natomiast wersja przybliżona jest wyliczana niezwykle szybko, a mimo to dobrze oddaje ogólny układ oświetlenia.

Odbicia zwierciadlane

Odbicia zwierciadlane zawarte są w przytoczonym poniżej elemencie równania:

$$\int_{\Omega} f_{r,s}(x, \vec{\omega}', \vec{\omega})(L_{i,C}(x, \vec{\omega}') + L_{i,D}(x, \vec{\omega}'))(\vec{\omega}' \cdot \vec{n})d\vec{\omega}'.$$

Tego typu odbicia są w dużym stopniu determinowane poprzez funkcję BRDF ($f_{r,s}$). Przybiera ona bardzo duże wartości jedynie w niewielkim obszarze, zbliżonym do kierunku odbicia lustrzanego. W pozostałych miejscach jest bliska, lub równa zero. Z tego powodu światło pochodzące z kierunków nie mieszczących się w obszarze dużych wartości BRDF można całkowicie zaniedbać.

Ze względu na swój charakter mapa fotonów nie jest wykorzystywana do obliczania odbić zwierciadlanych. Co prawda byłoby to możliwe, jednak wymagałoby wykorzystania mapy o olbrzymiej gęstości tak, aby dało się odnaleźć fotony pochodzące z odpowiedniego dla danej powierzchni i kąta patrzenia zbioru kierunków. Odbicia zwierciadlane i błyszczące są wyznaczane poprzez rekursywne śledzenie promieni. W tym zastosowaniu jest to metoda efektywna, gdyż zwykle wystarcza zaledwie kilka promieni rekursywnych (w przypadku perfekcyjnego zwierciadła dokładnie jeden), a sceny zawierają niewiele obiektów o właściwościach zwierciadlanych.



Rysunek 6.12.: Oświetlenie pośrednie (część rozproszona) we wnętrzu pokoju dziennego.

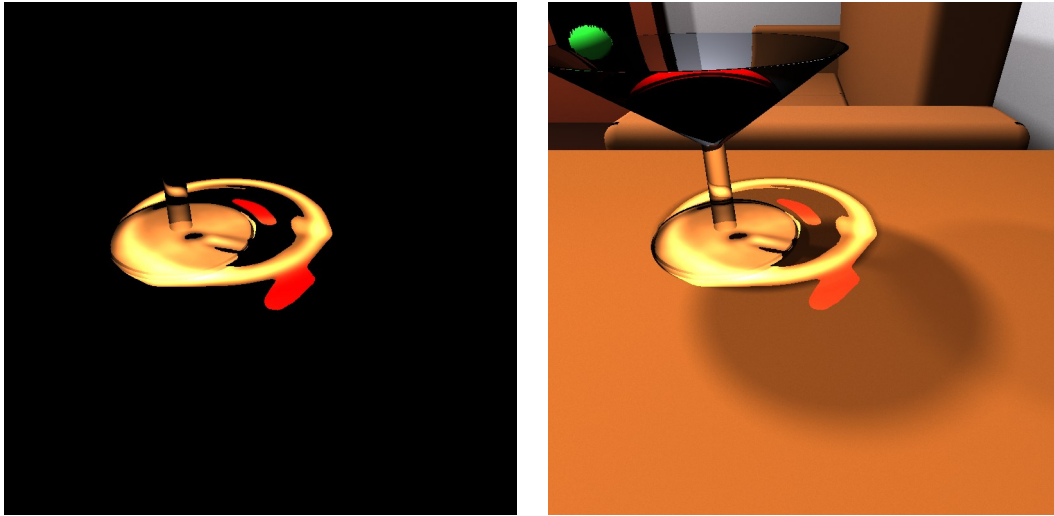
Odbicia rozproszone

Oświetlenie pochodzące od światła odbitego w sposób rozproszony opisuje następująca część sumy:

$$\int_{\Omega} f_{r,D}(x, \vec{\omega}', \vec{\omega}) L_{i,D}(x, \vec{\omega}') (\vec{\omega}' \cdot \vec{n}) d\vec{\omega}'.$$

W przypadku obliczeń dokładnych do wyznaczenia tej składowej wykorzystywane jest rekursywne śledzenie promieni wtórnych poprzez zastosowanie odpowiedniej metody monte carlo. W normalnych warunkach byłoby to rozwiązanie bardzo nieefektywne, jednak ten konkretny przypadek posiada kilka własności, które pozwalają wprowadzić uproszczenia. Po pierwsze kaustyki, które są głównym źródłem szumu w oświetleniu pośrednim, są obliczane osobno. Dzięki temu uzyskiwane przybliżenia są o wiele dokładniejsze przy wykorzystaniu znacznie mniejszej liczby ścieżek. Ponadto należy wykorzystać algorytm opisany w podrozdziale 6.2. Pozwoli to skierować promienie przede wszystkim w miejsca mające duży wkład w oświetlenie punktu. Dodatkowe przyspieszenie z pewnością zapewnią techniki zapamiętywania oświetlenia pośredniego i jego interpolacji ([38]), które dzięki odrzuceniu kaustyk będą się spisywać bardzo dobrze.

Największe przyspieszenie otrzymywane jest jednak dzięki wykorzystaniu obliczeń przybliżonych. Jak już wspomniano, są one wykonywane w miejscach, które nie są bezpośrednio widziane przez obserwatora. Dzięki temu większość ścieżek ma głębokość dokładnie jeden. Wyjątkiem są jedynie promienie wtórne natrafiające na obiekty zwierciadlane. W tym przypadku konieczne jest śledzenie kolejnego promienia rekursywnego. Obliczenia przybliżone wykonywane są poprzez wyznaczenie radiancji na podstawie mapy fotonów. Jako że na wartość radiancji w punkcie widzianym bezpośrednio przez obserwatora składa się wiele wartości wyliczonych w sposób przybliżony, można wykorzystać wstępne wyznaczanie irradiancji przedstawione w paragrafie 6.5.1. Pozwala to jeszcze bardziej przyspieszyć obliczenia.



Obraz kaustyki

Pełne oświetlenie globalne

Rysunek 6.13.: Zbliżenie kaustyki w pokoju dziennym.

Rysunek 6.12. przedstawia ponownie wnętrze pokoju, ale tym razem zawiera oświetlenie pośrednie pochodzące jedynie z odbić rozproszonych. Po lewej stronie znajduje się wynik obliczeń dokładnych. Został on utworzony przy pomocy śledzenia 5000 promieni wtórnych w każdym pikselu obrazu. Rezultat jest bardzo dobry, zniknęły wszystkie artefakty będące wynikiem bezpośredniej wizualizacji mapy fotonów. Środkowy obraz zawiera wizualizację wartości przybliżonych, wykorzystywanych w miejscach przecięcia promieni wtórnych ze sceną. Ich dokładność jest dość niska. Są one o wiele gorsze od tych wykorzystywanych w algorytmie wizualizacji bezpośredniej. Jednak taki stopień dokładności całkowicie wystarcza do otrzymania poprawnego wyniku końcowego. Obraz po prawej stronie przedstawia oświetlenie pośrednie otrzymane przy pomocy tradycyjnego śledzenia ścieżek z wykorzystaniem 5000 rekursywnych ścieżek. Różnica między tym obrazem, a rozwiązaniem dokładnym uzyskanym przy wykorzystaniu mapy fotonów jest ogromna. Pozwala to zorientować się, jak duży wpływ ma zastosowanie mapy fotonów na ostateczny wygląd oświetlenia pośredniego.

Kaustyki

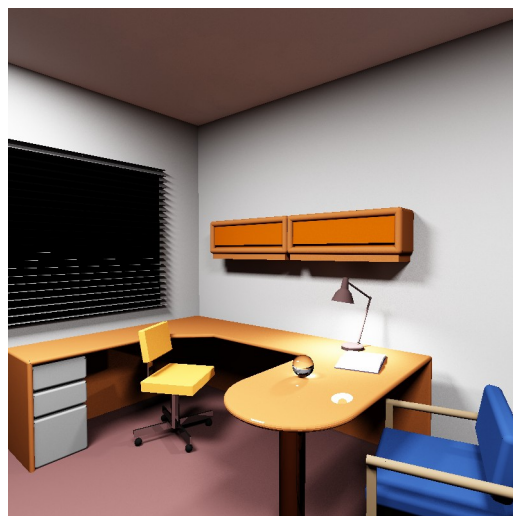
Kaustyki są reprezentowane przez następujący składnik równania:

$$\int_{\Omega} f_{r,D}(x, \vec{\omega}', \vec{\omega}) L_{i,C}(x, \vec{\omega}') (\vec{\omega}' \cdot \vec{n}) d\vec{\omega}'. \quad (6.24.)$$

W przypadku obliczeń dokładnych całka ta jest przybliżana za pomocą metody bezpośredniej wizualizacji mapy kaustycznej. Algorytm opisano w paragrafie 6.4.1. Zapewnia to wysoką jakość kaustyk widzianych bezpośrednio przez obserwatora oraz w odbiciach zwierciadlanych. W przypadku obliczeń przybliżonych oświetlenie pochodzące od kau-



Pokój dzienny



Wnętrze biura

Rysunek 6.14.: Obrazy uzyskane przy pomocy algorytmu dwufazowego.

styk dodawane jest poprzez dołączenie fotonów kaustycznych do estymacji radiancji dla odbicia rozproszonego (6.4.2.).

Rysunek 6.13. przedstawia obraz kaustyki utworzonej przez załamanie światła w kieliszku ponczu stojącym na stoliku w rogu prezentowanego wcześniej pokoju dziennego. Kieliszek jest oświetlany przez trzy źródła światła. Jednak tylko umiejscowiony bezpośrednio nad nim reflektor znajduje się pod kątem powodującym utworzenie kaustyki poprzez załamanie światła w szkłe. Pozostałe lampy są źródłem szerokiej czerwonej kaustyki pochodzącej od światła załamanego w ponczu.

Uzyskany obraz potwierdza dużą skuteczność metody opartej na mapie kaustycznej. Kaustyki są bardzo wyraźne i mają ostre, naturalne krawędzie. Nie ma żadnych fałszywych rozblasków. Utworzenie ich zajmuje zaledwie kilka sekund. Wszystkie te fakty są dowodem na to, że nawet w dużych i skomplikowanych scenach można z bardzo dużą dokładnością wizualizować kaustyki, nawet te o niewielkich rozmiarach.

Podsumowanie

W paragrafie tym opisano algorytm renderowania realistycznych obrazów pozwalający z bardzo dużą dokładnością odtworzyć wszystkie elementy oświetlenia globalnego. Stanowi on połączenie śledzenia promieni oraz wykorzystania różnego typu map fotonów w celu uzyskania jednego, silnego narzędzia. Wynikowa metoda jest bardzo ogólna. Pozwala ona renderować sceny o dowolnym stopniu skomplikowania geometrii oraz układu oświetlenia. Najważniejszą cechą algorytmu jest rozdzielenie oświetlenia na kilka składowych i renderowanie ich osobno, przy pomocy najodpowiedniejszej metody. Zabieg ten pozwala zdecydowanie zredukować czas i poprawić jakość wyników. Jego pobocznym efektem jest fakt, że otrzymany algorytm należy do klasy rozwiązań hybrydowych.

Mimo to zachowuje on prostotę i nie wprowadza żadnych sztucznych ograniczeń. Wykorzystanie mapy fotonów pozwala wręcz na rozszerzenie podstawowego rozwiązania o nowe, jeszcze bardziej zaawansowane możliwości, jak renderowanie materiałów mokrych ([21], [22]), odbijających światło w sposób podpowierzchniowy ([23]), czy ośrodków rozpraszających światło ([19]).

6.5. Rozszerzenia map fotonów

W podrozdziale tym opisano dwa rozszerzenia map fotonów. Pierwsze z nich (6.5.1.) pozwala zdecydowanie przyspieszyć obliczanie oświetlenia pośredniego w algorytmie dwufazowym. Drugie (6.5.2.) poprawia jakość konstruowanej mapy fotonów i może być powiązane z dowolnym algorytmem renderowania.

6.5.1. Wstępne obliczanie irradiancji

Najkosztowniejszą czasowo operacją wykonywaną w celu wyznaczenia irradiancji w punkcie (potrzebnej do wyliczenia radiancji) na podstawie mapy fotonów, jest wyszukiwanie zbioru najbliższych fotonów w kd-drzewie. Niestety, algorytm dwufazowy, w czasie wyznaczania oświetlenia pośredniego, wylicza irradiancję w olbrzymiej ilości punktów sceny, więc także wykonuje mnóstwo poszukiwań w kd-drzewie. Ze względu na charakter rozwiązywanego problemu obliczenia są wielokrotnie wykonywane w praktycznie tych samych miejscach. Toteż wstępne wyznaczenie potrzebnych wartości w określonych punktach pozwoliłoby znacznie zredukować czas działania.

W kolejnych podrozdziałach opisano modyfikację metody wstępnego wyliczania irradiancji opisanej w pracy [3].

Algorytm

Prostym rozwiązaniem byłoby wyliczenie irradiancji na każdej powierzchni w punktach o określonej odległości. Niestety, wykorzystanie tego rozwiązania w renderingu prowadziłoby do dużych przekłamań wszędzie tam, gdzie irradiancja zmienia się gwałtownie pomiędzy wybranymi punktami (np.: na granicy kaustyki). Ponadto wymaga ono zbudowania i zarządzania kolejną strukturą danych.

Lepszym sposobem jest wyliczanie irradiancji w punktach, w których zachowane są fotony. To rozwiązanie o wiele lepiej przystosowuje się do gwałtownych zmian oświetlenia, gdyż fotony z bardzo dużym prawdopodobieństwem będą znajdować się w interesujących punktach. Poza tym odpada konieczność tworzenia kolejnej struktury danych. Wyliczoną irradiancję można przechować razem z fotonem. Przechowywanie dodatkowych wartości wymaga powiększenia struktury fotonu o sześć bajtów, cztery na skompresowaną irradiancję i dwa na również skompresowaną reprezentację normalnej. Daje to przyrost rozmiaru fotonu, a co za tym idzie całej mapy, o 30%.

Obliczenia wstępne można wykonać przed właściwym renderingiem, poprzez zwykłe wyliczenie irradiancji na podstawie mapy w każdym punkcie, w którym znajduje się fo-

ton. Wyznaczenie irradiancji w mapie zawierającej N fotonów zajmuje $O(\log N)$ czasu, czyli złożoność tej operacji wykonanej na wszystkich fotonach wynosi $O(N \log N)$.

Znacznie lepszym rozwiązaniem jest wyznaczanie oszacowania na żądanie. We flagach fotonu można przechowywać informację o tym, czy irradiancja już została dla niego wyznaczona. Jeżeli w czasie renderingu jako źródło oszacowania wybrany zostanie foton, w którym irradiancja nie została jeszcze policzona, oblicza się ją, zachowuje uzyskaną wartość w fotonie oraz uaktualnia wpis w flagach. Przy każdym następnym wykorzystaniu tego fotonu używana jest wcześniej uzyskana wartość. Przewagą tego rozwiązania jest to, że pozwala ono nie liczyć irradiancji w fotonach, które nigdy nie zostaną wykorzystane, nie wnosząc przy tym żadnych dodatkowych kosztów.

Wyliczone wartości wykorzystuje się podczas renderowania w następujący sposób. W czasie wyznaczania irradiancji dla punktu x , zamiast wykonywać standardowe przeszukiwanie kd-drzewa w celu odnalezienia k pobliskich fotonów, lokalizuje się tylko jeden, najbliższy foton, którego normalna jest zbliżona do normalnej $\vec{n}_x$. W większości przypadków pierwszy odnaleziony foton będzie spełniał postawione kryteria. Jedyne na bardzo zakrzywionych powierzchniach, lub w rogach, może się zdarzyć, że konieczne będzie sprawdzenie kilku fotonów. Do dalszych obliczeń wykorzystywana jest wartość irradiancji zapisana w tym fotonie. Wymaganie zgodności normalnych jest potrzebne, gdyż dzięki niemu wyklucza się możliwość wykorzystania fotonu, który trafił w drugą stronę cienkiej powierzchni, bądź w inny, bardzo bliski obiekt. Poza tym, jeśli w czasie wstępnych obliczeń stosowana była kompresja obszaru wyszukiwania, pomaga ono w dobraniu oszacowania, które powstało na bazie najbardziej optymalnych fotonów.

Rozszerzona procedura wyszukiwania fotonów wymaga wyznaczania iloczynu skalarnego normalnej zapisanej w fotonie oraz $\vec{n}_x$, w celu sprawdzenia ich zgodności. Są to jednak zaledwie trzy mnożenia i dwa dodawania zmiennopozycyjne przypadające na jedno sprawdzenie. Stanowi to bardzo niewielki koszt w porównaniu z wyszukiwaniem fotonu w kd-drzewie. Poza tym, jeżeli pierwszy odnaleziony foton jest akceptowany, to są to jedyne dodatkowe obliczenia.

Zastosowanie opisanej metody oznacza, że irradiancja jest przybliżana za pomocą funkcji przedziałami stałej. Dokładniej, fotony dzielą powierzchnie wewnątrz sceny na diagramy Voronoi reprezentujące obszary o tej samej irradiancji. To przybliżenie jest jednak akceptowalne, gdyż irradiancja pomiędzy fotonami zazwyczaj nie zmienia się bardzo gwałtownie, a ostateczna wartość radiancji w punktach widzianych bezpośrednio przez obserwatora powstaje przez uśrednienie wielu wartości.

Przykłady

Scena testowa to prezentowany już w poprzednim rozdziale model wnętrza biura. Rysunek 6.15. przedstawia wizualizację irradiancji policzonej na podstawie mapy zawierającej $5 \cdot 10^5$ fotonów dokładnie w punktach przecięcia promieni pierwotnych ze sceną (po lewej), oraz wyznaczonej za pomocą oszacowań, w których normalna w wykorzystanym fotonie różniła się nie więcej niż 5° (w środku) i 20° (po prawej), od tej w rozpa-

6.5. Rozszerzenia map fotonów



Rysunek 6.15.: Irradiancja wyznaczona na podstawie mapy fotonów (po lewej), oszacowań z odchyleniem do 5° (w środku) i 20° (po prawej).

trywanym punkcie. Różnice między wszystkimi trzema obrazami są bardzo niewielkie. Jedynym poważnym błędem są czarne kółka krzesła oraz róg oparcia fotela na obrazie środkowym. Problem ten nie występuje na obrazie po prawej stronie, stąd wniosek, że wynikał on z niemożności odnalezienia oszacowania (fotonu) o odpowiednio podobnej normalnej. Zastosowanie słabszego kryterium pozwala wyeliminować ten problem, czego dowodem jest obraz po prawej stronie. Dokładniejsze oględziny ujawniają także drobne problemy na powierzchniach krzywoliniowych oraz w rogach. Zarysowane są tam wspomniane wcześniej diagramy Voronoi, wyraźne tym bardziej, im większa jest dozwolona różnica w normalnych. Zaburzenia te nie stanowią jednak wielkiego problemu, gdyż ostatecznie radiancja jest wyznaczana na podstawie wielu próbek.

Tabela 6.1. przedstawia porównanie wykorzystania różnych maksymalnych odchylen normalnej. Jak można się było tego spodziewać, im większa dozwolona różnica, tym mniejszy dodatkowy narzut czasowy. Optymalnym wyborem dla tej sceny jest odchylenie nie większe niż 20° . Dodatkowy koszt związany z wyszukiwaniem odpowiedniego fotonu jest wtedy bardzo mały, a otrzymywane przybliżenia ciągle bardzo dokładne. Zmniejszenie maksymalnej różnicy kątów powoduje zdecydowany przyrost czasu obliczeń, dając jedynie nieznaczną poprawę jakości wyników. Na dodatek może prowadzić do opisanych wcześniej problemów z odnalezieniem odpowiedniego oszacowania na powierzchniach krzywoliniowych o niewielkiej gęstości fotonów. Większe maksymalne odchylenia skutkują dużymi przekłamaniami w otrzymywanych wynikach, dając

Maksymalne odchylenie normalnej	5°	10°	20°	30°	45°
Średnia ilość sprawdzonych fotonów	7.05	3.65	2.06	1.67	1.48
Wydłużenie czasu działania w stosunku do wyszukania najbliższego fotonu	10%	4.2%	1.9%	1.4%	0.9%

Tabela 6.1.: Wpływ maksymalnego odchylenia normalnej na czas wyliczania irradiancji.



Rysunek 6.16.: Oświetlenie pośrednie wyznaczone na podstawie oszacowań o maksymalnym odchyleniu 20° .

jedynie znikomą poprawę wydajności.

Charakterystyczna dla tej sceny jest wysoka średnia fotonów sprawdzanych przed odnalezieniem tego z odpowiednią normalną. Nawet przy odchyleniach dochodzących do 45° konieczne było sprawdzenie średnio 1.48 fotonu. Jest to wynikiem dużej ilości niewielkich powierzchni krzywoliniowych, na których gęstość fotonów jest niezbyt duża. Skutkiem tego jest dość chaotyczne przeszukiwanie całej okolicy w poszukiwaniu odpowiedniego kandydata, które pogarsza wydajność i jakość wyników. W scenach nie zawierających tego typu obiektów problem nie występuje, np.: średnia dla kulek prezentowanych w podrozdziale 5.2. wynosi 1.02 przy odchyleniach do 20° . Właśnie ta różnica kątów jest najbardziej optymalną wartością w większości przypadków.

Rysunek 6.16. zawiera wizualizację oświetlenia pośredniego w dwóch różnych scenach, wyznaczonego na podstawie przybliżeń irradiancji. W obu przypadkach wykorzystano algorytm wprowadzony w paragrafie 6.4.2. oraz przybliżenia, których maksymalne odchylenie od normalnej wynosiło nie więcej niż 20° . Jak widać opisane w poprzednim akapicie artefakty nie przeniosły się na wynikowe obrazy. Oświetlenie pośrednie

Scena	Kulki	Biuro
Czas renderingu z użyciem przybliżeń	1135.9s	1534.93s
Czas wyznaczania przybliżeń	86.55s (7.6%)	490.08s (31.9%)
Czas renderingu bez użycia przybliżeń	8817.5s	14893.63s
Uzyskane przyspieszenie	$7.76\times$	$9.7\times$

Tabela 6.2.: Wpływ wykorzystania przybliżeń irradiancji na czas renderingu.

oddane jest w sposób perfekcyjny. Warto zwrócić uwagę na wyjątkową gładkość i delikatność obrazów oraz krwawienie kolorów, zaznaczone bardzo wyraźnie w przypadku wizualizacji kulek. W tabelce 6.2. przedstawiono porównanie czasów tworzenia omówionych obrazów z wykorzystaniem przybliżeń irradiancji (wyliczanych na żądanie) oraz bez nich. Jak widać wykorzystanie przybliżeń pozwala zaoszczędzić olbrzymie ilości czasu.

Podsumowanie

W paragrafie tym opisano metodę wstępnego wyznaczania irradiancji w wybranych punktach sceny na podstawie mapy fotonów. Pozwala ona znacząco zredukować czas renderowania obrazów z wykorzystaniem mapy fotonów. W przypadku algorytmów rekursywnych, jak ten prezentowany w paragrafie 6.4.2., osiągane jest około ośmiokrotne skrócenie czasu renderingu, bez żadnej widocznej utraty jakości. Metoda jest całkowicie przezroczysta dla innych technik i może być z nimi bez żadnych ograniczeń łączona. Wyjątkiem są techniki bezpośredniej wizualizacji mapy (paragraf 6.4.1.), które nie mogą jej stosować ze względu na wyraźnie widoczne błędy w oszacowaniu radiancji.

6.5.2. Sterowanie gęstością mapy fotonów

Jak już wspomniano w podrozdziale 4.2., w czasie konstrukcji mapy fotony są emitowane ze wszystkich źródeł światła zgodnie z ich właściwościami fizycznymi. Następnie podróżują one poprzez scenę i przy każdej interakcji z jej obiektami odbijają się zgodnie z właściwościami napotkanych materiałów. Dzięki takiej metodzie emisji wszystkie zachowane w mapie fotony mają bardzo podobną moc. Różnice w ilości światła padającego na poszczególne punkty odzwierciedlone są jedynie w gęstości mapy. Im wyższa gęstość w okolicy punktu, tym mocniej jest on oświetlony. Duża gęstość gwarantuje także bardzo dużą dokładność przybliżenia.

Niestety, ten sposób emisji fotonów posiada istotne wady. Mapa zbudowana dla sceny o zróżnicowanym oświetleniu ma bardzo niejednorodną gęstość. W okolicach punktów mocno oświetlonych jest ona bardzo duża, często zdecydowanie większa niż potrzeba. Natomiast w pozostałych miejscach, szczególnie tych nie oświetlonych bezpośrednio przez żadne światło, gęstość jest niska. Próby zwiększenia gęstości w tych obszarach poprzez wyemitowanie dodatkowych fotonów nie odnoszą pożądanego skutku, gdyż większość z nich ponownie trafia w okolice silnie oświetlone. Problem ten można skutecznie rozwiązać jedynie przy pomocy algorytmu konstrukcji mapy kontrolującego gęstość tworzonej struktury. Metoda taka jest opisana w kolejnych paragrafach. Jest to modyfikacja algorytmu opisanego w pracy [30] przedstawiona na kursie 38 w czasie konferencji *Siggraph 2001* ([20]).

Konstrukcja mapy fotonów

W rozszerzonym algorytmie konstrukcji mapy fotony są emitowane i śledzone dokładnie tak samo jak w wersji oryginalnej. Zmieniona jest jedynie procedura ich zachowywania.

Fotony są zapisywane w mapie tylko dopóki ich gęstość w danej okolicy nie jest odpowiednio duża. Jeśli foton trafi w obszar o wystarczająco dużej gęstości, to nie zostanie zapisany.

Precyzyjniej — założmy że mamy funkcję $D_r(x)$, która dla dowolnego punktu sceny x podaje gęstość mapy wymaganą do dokładnego przybliżenia oświetlenia w tym miejscu. Niech foton k trafi w punkt y . W celu sprawdzenia, czy ma on być zachowany, konieczne jest ustalenie bieżącej gęstości mapy w okolicy tego punktu ($D_c(y)$). Wartość tą można wyznaczyć przy pomocy standardowej estymacji gęstości, lokalizując N najbliższych fotonów:

$$D_c(y) = \frac{N}{\pi r_N^2}, \quad (6.25.)$$

gdzie r_N oznacza odległość najdalszego z odnalezionych fotonów od punktu y . Foton k zostanie zachowany w mapie tylko, jeżeli jej bieżąca gęstość nie jest wystarczająca, czyli:

$$\frac{D_c(y)}{D_r(y)} \leq 1. \quad (6.26.)$$

W przeciwnym wypadku foton zostanie odrzucony. Jednak w celu zachowania zgodności pomiędzy mocą wyemitowaną a zachowaną w mapie, jego moc musi zostać odpowiednio rozdysponowana. Dobrym sposobem jest rozdzielenie jej pomiędzy okoliczne fotony. Gdyby foton k został zachowany w mapie, to radiancja wyznaczona w punkcie y na podstawie $N + 1$ fotonów miałaby następującą wartość:

$$L_r(x, \vec{\omega}) = \frac{\sum_{p=1}^N f_r(x, \vec{\omega}_p, \vec{\omega}) \Phi_p + f_r(x, \vec{\omega}_k, \vec{\omega}) \Phi_k}{\pi r_N^2}. \quad (6.27.)$$

r_N pozostaje niezmiennione, gdyż foton k trafił dokładnie w punkt y . Moc pozostałych fotonów w oszacowaniu powinna być dobrana tak, aby estymacja radiacji z pominięciem fotonu k dała dokładnie ten sam wynik:

$$L_r(x, \vec{\omega}) = \frac{\sum_{p=1}^N f_r(x, \vec{\omega}_p, \vec{\omega}) (\Phi_p + \Delta\Phi_{k,p})}{\pi r_N^2}. \quad (6.28.)$$

Wartość $\Delta\Phi_{k,p}$ definiuje się w następujący sposób:

$$\Delta\Phi_{k,p} = \begin{cases} \Phi_k/N' & \forall p. (\vec{\omega}_p \cdot \vec{n}_y) > 0 \\ 0 & \text{w p.p.} \end{cases}, \quad (6.29.)$$

gdzie N' oznacza liczbę fotonów spełniających postawiony warunek. Dzięki takiemu wyborowi moc zostanie rozdysponowana równo pomiędzy te fotony, które rzeczywiście biorą udział w estymacji radiancji. Możliwe są oczywiście inne algorytmy rozdzielania mocy pomiędzy fotony, np.: bazując na ich odległości od punktu y , czy zgodności $\vec{\omega}_p$ z $\vec{n}_y$. Jednak wykorzystanie tej prostej metody pozwala utrzymać moc wszystkich fotonów na w miarę równym poziomie.

Modyfikacja mocy okolicznych fotonów powoduje zmianę wyników estymacji radiancji w pobliskich punktach. Może to być źródłem artefaktów, szczególnie w okolicach

gwałtownych zmian D_r . W tych miejscach moc fotonów pochodzących z obszaru o niskiej wartości D_r będzie rozdzielana także pomiędzy fotony należące do sąsiedniego obszaru o wysokim D_r , powodując zdecydowany wzrost mocy skumulowanej na granicy. Rozwiązaniem tego problemu jest rozdzielanie mocy jedynie pomiędzy kilka (15) najbliższych fotonów.

Do wyznaczenia wartości funkcji D_c konieczne jest wykonanie operacji wyszukiwania zbioru N fotonów najbliższych punktowi y już w czasie tworzenia mapy. Aby zredukować czas wyszukiwania zapisane w mapie fotony powinny być przechowywane w (niezbilansowanym) kd-drzewie, a nie jak dotychczas w zwykłej tablicy. Oczywiście drzewo to należy zbilansować przed rozpoczęciem renderingu.

Opisana metoda kontroli gęstości mapy jest parametryzowana przez funkcję $D_r(x)$, odpowiedzialną za określenie wymaganej gęstości. Konstrukcję takiej funkcji opisuje następny paragraf.

Funkcja wymaganej gęstości mapy

Skonstruowanie dobrej funkcji określającej odpowiednią gęstości mapy wymaga zbadania wpływu, jaki ma gęstość mapy na błąd estymacji radiancji wykonanej na jej podstawie oraz na ostateczny błąd widoczny w poszczególnych pikselach obrazu.

Zdefiniujmy funkcję emitowanego potencjału:

$$W_e(x, \vec{\omega}) = \begin{cases} 1 & \begin{cases} \text{jeśli } x \text{ jest równe pozycji obserwatora} \\ \text{oraz } \vec{\omega} \text{ prowadzi przez ekran}^5 \end{cases} \\ 0 & \text{w p.p.} \end{cases} \quad (6.30.)$$

Potencjał jest odpowiednikiem radiancji, podobnie istotność jest równoważna irradiancji. Istotność jest emitowana przez obserwatora i śledzona poprzez scenę, dopóki nie napotka źródła światła. Równanie transportu potencjału wygląda więc następująco:

$$W(x, \vec{\omega}) = W_e(x, \vec{\omega}) + \int_{\Omega} f_r(x, \vec{\omega}', \vec{\omega}) W_i(x, \vec{\omega}') (\vec{\omega}' \cdot \vec{n}) d\vec{\omega}'. \quad (6.31.)$$

Ponadto, podobnie jak dla radiancji, zachodzi zasada zachowania potencjału:

$$W_i(x \leftarrow y) = W(x \rightarrow y). \quad (6.32.)$$

Istotność padająca na punkt x wyraża się wzorem:

$$\Gamma(x) = \int_{\Omega} W_i(x, \vec{\omega}) (\vec{\omega} \cdot \vec{n}) d\vec{\omega}. \quad (6.33.)$$

Całkowita istotność osiągnięta daną powierzchnią opisywana jest poniższym wzorem:

$$\Psi = \int_A \Gamma(x) dA. \quad (6.34.)$$

⁵Ekran można zdefiniować na dwa sposoby: jest to cała powierzchnia obrazu, lub pojedynczy piksel.

Znajomość potencjału umożliwia wyznaczenie strumienia światła, który trafia do obserwatora poprzez ekran:

$$\Phi_{scr} = \int_A \int_{\Omega} W_i(x, \vec{\omega}) L_e(x, \vec{\omega}) (\vec{\omega} \cdot \vec{n}) d\vec{\omega} dA, \quad (6.35.)$$

gdzie A oznacza całkowanie po wszystkich źródłach światła, a Ω po wszystkich kierunkach. W analogiczny sposób definiuje się oświetlenie pośrednie osiagające ekran, z tym że całkowanie odbywa się po wszystkich powierzchniach sceny. Rekonstrukcja radiancji na podstawie mapy fotonów jest jednak obciążona błędem, który ostatecznie uwidacznia się w poszczególnych pikselach obrazu:

$$\Phi_{pix,D} + \epsilon(\Phi_{pix,D}) = \int_A \int_{\Omega} W_D(x, \vec{\omega}) (L_D(x, \vec{\omega}) + \epsilon(L_D(x, \vec{\omega}))) (\vec{\omega} \cdot \vec{n}) d\vec{\omega} dA, \quad (6.36.)$$

czyli błąd równy jest:

$$\begin{aligned} \epsilon(\Phi_{pix,D}) &= \int_A \int_{\Omega} W_D(x, \vec{\omega}) \epsilon(L_D(x, \vec{\omega})) (\vec{\omega} \cdot \vec{n}) d\vec{\omega} dA \\ &= \int_A \left(\int_{\Omega} W_D(x, \vec{\omega}) (\vec{\omega} \cdot \vec{n}) d\vec{\omega} \right) \epsilon(L_D(x, \vec{\omega})) dA \\ &= \int_A \Gamma_D(x) \epsilon(L_D(x, \vec{\omega})) dA. \end{aligned} \quad (6.37.)$$

Błąd zależy więc od istotności, a błąd wnoszony przez pojedynczy punkt sceny x wynosi:

$$\epsilon(\Phi_{pix,D}(x)) = \Gamma_D(x) \epsilon(L_D(x)). \quad (6.38.)$$

Stąd następujące ograniczenie na błąd estymacji:

$$\epsilon(L_D(x)) \leq \frac{\epsilon(\Phi_{pix,D}^{max}(x))}{\Gamma_D(x)}. \quad (6.39.)$$

Jeśli założyć, że każdy punkt sceny może wnieść taki sam błąd do całkowitego błędu widzianego w pikselu, to:

$$\epsilon(L_D(x)) \leq \frac{\epsilon(\Phi_{pix,D}^{max}(x))}{A_t \Gamma_D(x)}, \quad (6.40.)$$

gdzie A_t oznacza całkowitą powierzchnię wszystkich obiektów sceny.

Ostatnim krokiem jest związanie oszacowania błędu estymacji z funkcją wymaganej gęstości mapy:

$$D_r(x) = \Gamma(x) \frac{S_r}{A_{pix}}, \quad (6.41.)$$

gdzie A_{pix} oznacza powierzchnię piksela, a S_r ilość fotonów. Czynniki S_r/A_{pix} oznacza więc docelową gęstość pikseli, która odpowiada jednostkowej istotności. Parametr S_r jest dobierany ręcznie.

Pozostaje jeszcze skonstruowanie metody wyznaczania istotności w każdym punkcie sceny ($\Gamma(x)$). Szczegółowy opis znajduje się poniżej.

Mapa istotności

Mapa istotności jest strukturą bliźniaczą mapie fotonów. Metoda jej tworzenia jest bardzo podobna do oryginalnego algorytmu konstrukcji mapy. W tym przypadku scena zawiera zawsze jedno źródło importonów⁶, które można potraktować jak specjalne źródło światła. Znajduje się ono dokładnie w tym samym punkcie co obserwator. Importony emitowane są z rozkładem jednorodnym w kierunku płaszczyzny obrazu, ograniczonej ze wszystkich stron przez granice ekranu, na którym generowany jest obraz. Algorytm śledzenia importonów przez scenę jest taki sam jak w przypadku fotonów. Podobnie jak fotony, są one zachowywane jedynie na powierzchniach odbijających światło w sposób rozproszony. Importony zapisywane są w dwóch osobnych mapach: normalnej i kaustycznej. W mapie kaustycznej zachowywane są wszystkie importony, które pokonały ścieżki ES*. Mapa globalna zawiera importony, które pokonały następujące ścieżki ES*DS*. Pozostałe importony (ES*DS*D(D|S)+) nigdy nie są generowane, gdyż ścieżki tego typu nie występują w czasie renderingu z użyciem mapy fotonów.

Dokładnie tak samo jak w przypadku mapy fotonów, istotność wszystkich zachowanych w mapie importonów jest skalowana przez ilość wyemitowanych cząstek. Istotność wyemitowana przez jeden piksel wynosi $\Psi_{pix} = A_{pix}$, a przez cały obraz $\Psi_{scr} = A_{scr}$.

Estymacja istotności w dowolnym punkcie x na podstawie mapy jest dokładnie taka sama, jak estymacja radiancji i bazuje na N najbliższych importonach:

$$\Gamma(x) \approx \frac{\sum_{p=1}^N \Psi_p}{\pi r_N^2}. \quad (6.42.)$$

Wskazówki implementacyjne

Dołączenie opisanej metody sterowania gęstością mapy fotonów do algorytmów renderowania jest bardzo proste. Wystarczy dodać do oryginalnego rozwiązania nowy pierwszy krok, w którym zostanie stworzona mapa istotności.

Mapa istotności powinna być zbudowana raz dla całego obrazu. Oznacza to, że minimalizowany będzie błąd dla całego obrazu: $\epsilon(\Phi_{scr}^{max}(x))$. Wadą tego rozwiązania jest fakt, że błąd w pojedynczych pikselach może ciągle być wysoki. Alternatywnym rozwiązaniem mogłoby być zbudowanie osobnej mapy dla każdego piksela. Istotność w konkretnym punkcie określana byłaby wtedy jako maksymalna istotność dla wszystkich pikseli. Jednak tworzenie mapy istotności dla każdego piksela oddzielnie byłoby zdecydowanie zbyt kosztowne, zarówno czasowo, jak i pamięciowo.

Mapa istotności jest bardzo intensywnie wykorzystywana w czasie emisji fotonów, przed każdym zapisaniem fotonu w mapie wykonywana jest estymacja istotności. Dlatego warto, podobnie jak w przypadku mapy fotonów, obliczyć wstępnie wartości istotności we wszystkich miejscach, w które trafiły importony i wykorzystywać te oszacowania w czasie tworzenia mapy.

Jak już wcześniej wspomniano, miejsca gwałtownych zmian wymaganej gęstości mapy mogą być przyczyną artefaktów. Aby zminimalizować ryzyko ich powstania istotność

⁶Tak nazywają się fotony zachowywane w mapie istotności.

powinna być oszacowywana przy pomocy dość dużej (250) ilości importonów. W przypadku zwykłej mapy fotonów współczynnik gęstości jednostkowej S_r powinien mieć wartość 2. Dla map kaustycznych oraz przy bezpośredniej wizualizacji mapy wymagana jest większa dokładność, dlatego powinno to być 30. Do oszacowania bieżącej gęstości mapy wystarczy około 15–20 najbliższych fotonów.

Czas tworzenia mapy zawierającej ustaloną liczbę fotonów jest zdecydowanie większy w przypadku metody wykorzystującej kontrolę gęstości. Przyrost czasu jest zależny od sceny oraz ilości fotonów, które są zachowywane w mapie. Generalnie można się spodziewać około czterokrotnego wzrostu czasu, jednak zdarzają się przypadki nawet ponad dziesięciokrotnego wydłużenia całkowitego czasu tworzenia mapy. Mimo wszystko czas budowy mapy pozostaje niewielki w stosunku do czasu renderingu. Zwiększony koszt jest efektem konieczności stworzenia dodatkowej mapy istotności, wykonywania estymacji gęstości każdorazowo przed zapisaniem fotonu oraz emitowania większej liczby fotonów (gdyż część z nich nie jest zachowywana w mapie). Jednak osiągnięcie zadowalającej gęstości mapy w każdym punkcie sceny przy pomocy tradycyjnego algorytmu wymaga wyemitowania i zapisania zdecydowanie większej ilości fotonów. Fakt ten równoważy zapotrzebowanie na czas. Podobnie jest z pamięcią. Co prawda algorytm kontrolujący gęstość musi przechowywać dodatkową mapę istotności, której rozmiar powinien być dość duży (około $2-5 \cdot 10^4$ dla średnio skomplikowanej sceny). Jednak zazwyczaj zapisuje zdecydowanie mniej fotonów, przez co całkowita ilość wykorzystanej pamięci jest podobna. Ponadto po wyemitowaniu wszystkich fotonów mapę istotności można usunąć z pamięci.

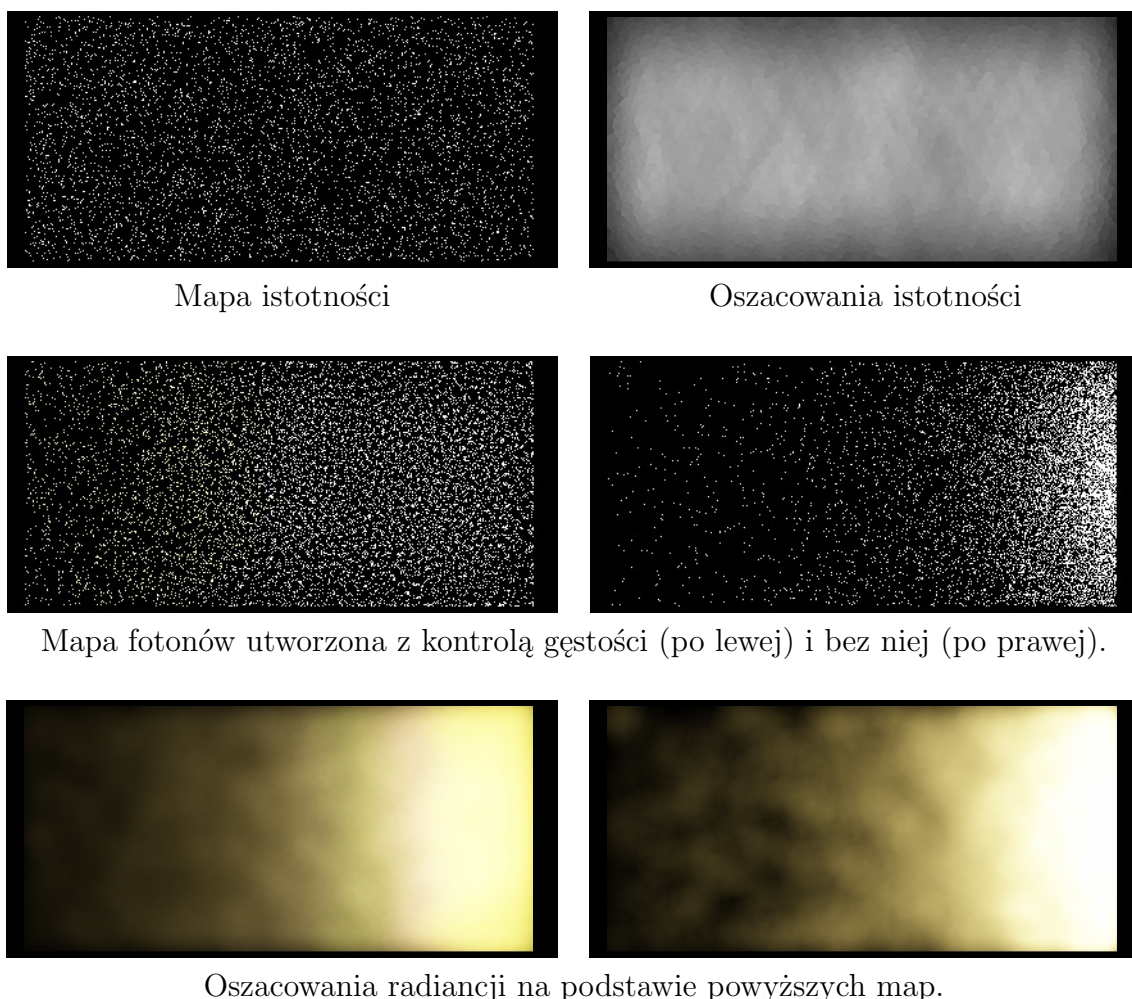
Ważną zaletą kontroli gęstości jest fakt, iż nakłada ona górne ograniczenie na całkowitą ilość fotonów zapisanych w mapie. Po osiągnięciu wymaganej gęstości w każdym punkcie sceny żaden dodatkowy foton nie zostanie dodany do mapy. Dzięki temu możliwe jest stworzenie algorytmu automatycznie dobierającego optymalną liczbę fotonów w mapie. W tym celu wystarczy kontrolować stosunek fotonów zapisanych w mapie do wszystkich interakcji z powierzchniami i po osiągnięciu odpowiednio niskiego odsetku zakończyć emisję nowych cząstek.

Przykłady

Rysunek 6.17. zawiera pierwszą scenę testową. Jest to płaszczyzna oświetlona z góry przez jedno źródło światła. Lampa umieszczona jest tuż przy prawej krawędzi płaszczyzny.

Pierwszy wiersz tego rysunku zawiera wizualizację mapy istotności oraz wyliczonych na jej podstawie oszacowań istotności w poszczególnych punktach. Mapa zawiera $5 \cdot 10^3$ importonów. Istotność w każdym punkcie była wyznaczana przy pomocy 250 najbliższych importonów. Jest ona w przybliżeniu równa dla całej płaszczyzny, gdyż obserwator miał jej całą powierzchnię w polu widzenia w czasie emitowania importonów.

Drugi wiersz prezentuje mapy fotonów. Obydwie mapy zawierają $1.1 \cdot 10^4$ fotonów. Mapa po lewej stronie została utworzona z wykorzystaniem kontroli gęstości opartej



Mapa fotonów utworzona z kontrolą gęstości (po lewej) i bez niej (po prawej).

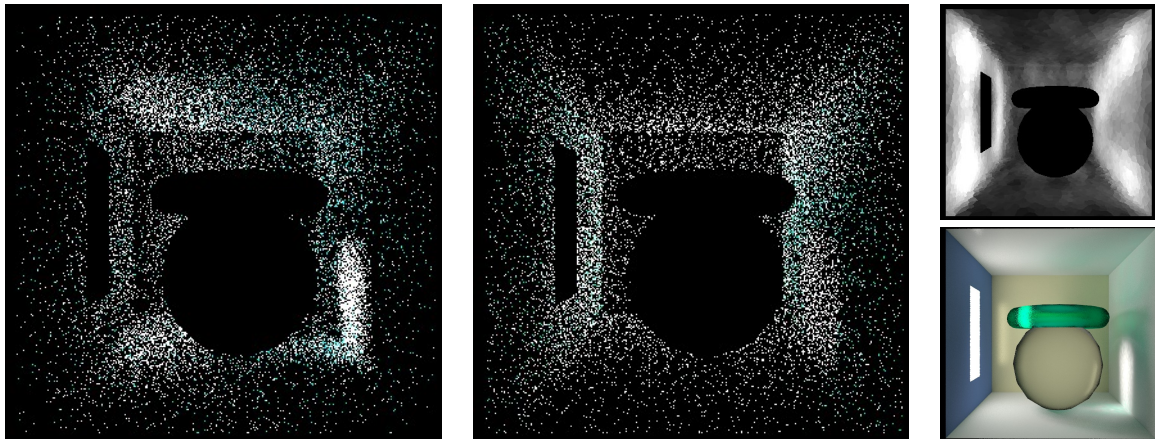
Oszacowania radiancji na podstawie powyższych map.

Rysunek 6.17.: Porównanie mapy utworzonej bez i z wykorzystaniem kontroli gęstości.

na przedstawionej wyżej mapie istotności, a mapa po prawej przy pomocy tradycyjnego sposobu emisji. Różnica w jakości obydwu map jest bardzo duża. W przypadku tradycyjnej emisji większość fotonów została zapisana w pobliżu prawej krawędzi płaszczyzny. Ich gęstość jest w tym miejscu bardzo duża. Natomiast z lewej strony fotonów jest niewiele. Wynika to z charakterystyki lampy, świeci ona z największą mocą w dół, oraz jej specyficznego położenia.

Dodanie kontroli gęstości zdecydowanie poprawiło układ fotonów. Rozkład cząstek na całej powierzchni płaszczyzny jest mniej więcej taki sam. W tym wypadku różnice w oświetleniu odzwierciedlane są nie w gęstości, lecz w mocy fotonów. Maleje ona wraz ze zwiększaniem odległości od lampy. Co ważne, zmiana jest bardzo łagodna, dzięki czemu nie generuje żadnych błędów oszacowań irradiancji.

Utworzenie mapy przy pomocy tradycyjnego sposobu emisji zajęło 1.2s. Wykorzystanie kontroli gęstości zwiększyło czas dziewięciokrotnie (łączy czas tworzenia mapy



Rysunek 6.18.: Mapy fotonów wykonane bez (po lewej) i z kontrolą gęstości (w środku) w scenie zawierającej obiekty kaustyczne. Po prawej znajduje się estymacja istotności (na górze) i pełne oświetlenie globalne (na dole) wykonane na podstawie mapy stworzonej z wykorzystaniem kontroli gęstości.

istotności i fotonów wynosił 11s). Jednak osiągnięcie podobnego stopnia zagęszczenia fotonów po lewej stronie płaszczyzny przy pomocy zwykłego algorytmu emisji wymaga zapisania około $6 * 10^4$ fotonów. Taka mapa tworzona jest w czasie 6s i zawiera ponad pięć razy więcej fotonów. Rendering z jej wykorzystaniem jest sporo wolniejszy z powodu zdecydowanie większego kd-drzewa oraz wykonania większej ilości wstępnych oszacowań irradancji (6.5.1.).

Wykorzystanie mapy o jednolitej gęstości pozwoliło uzyskać o wiele lepszą estymację radiancji (ostatni wiersz rysunku). Utworzony obraz jest gładniejszy na całej powierzchni płaszczyzny. Jest to efektem rozdzielania mocy nadmiarowych fotonów pomiędzy ich sąsiadów, które działa podobnie do filtrowania. Wyniki te pokazują, że nawet w prostych scenach gęstość mapy ma wielki wpływ na ostateczny rezultat renderingu.

Rysunek 6.18. zawiera drugą scenę testową. Są to dwa przezroczyste obiekty oświetlone z boku przez jedną bardzo dużą lampę. Obydwa obiekty generują duże i wyraźne kaustyki. Na rysunku przedstawiono zwykłe mapy fotonów, które zawierają $1.6 * 10^4$ cząstek, oszacowania istotności, uzyskane przy pomocy mapy zawierającej $7 * 10^3$ importonów oraz wyniki renderingu wykonanego z wykorzystaniem mapy utworzonej z kontrolą gęstości. Czas tworzenia mapy przy pomocy tradycyjnego schematu emisji wyniósł 0.67s. Dodanie kontroli gęstości zwiększyło całkowity czas do 5.65s, czyli ponad ośmiokrotnie.

Jak można się było spodziewać, różnice w rozkładach fotonów są bardzo duże. W przypadku mapy utworzonej bez wykorzystania kontroli gęstości większość cząstek znalazła się w okolicach kaustyk. Zachowanie tak dużej liczby fotonów w tych obszarach jest całkowicie niepotrzebne, gdyż kaustyki są generowane przy pomocy osobnej mapy kaustycznej. W pozostałych regionach sceny mapa ma bardzo zróżnicowaną gęstość,

w szczególności w prawym dolnym rogu znajduje się bardzo mało fotonów, a w lewym górnym bardzo dużo.

Kontrola gęstości ponownie przyczyniła się do poprawy rozkładu fotonów. Cała scena jest bardzo równomiernie wypełniona cząstkami, a w miejscu gdzie znajduje się kaustyka, nie powstało żadne skupisko fotonów. Jest to oczywiście efektem wymagania niezbyt dużej gęstości zwykłej mapy fotonów. Wyniki renderingu wyraźnie pokazują, że brak tych dodatkowych fotonów nie powoduje utraty jakości odtworzenia kaustyki. Dzieje się tak dlatego, że w przypadku mapy kaustycznej (nie pokazanej na rysunkach) wymagana jest piętnastokrotnie większa gęstość.

Podsumowanie

W paragrafie tym opisano modyfikację algorytmu tworzenia mapy fotonów, umożliwiającą skuteczne kontrolowanie gęstości tworzonej mapy. Zaprezentowana metoda pozwala ograniczyć gęstość mapy w dowolnym miejscu sceny przez pewną z góry ustaloną wartość. Dzięki temu możliwe jest osiągnięcie wymaganej gęstości mapy w każdym punkcie sceny bez konieczności zapisywania olbrzymiej ilości fotonów. Przedstawione rozwiązanie eliminuje wszystkie problemy wynikające z nieodpowiedniej gęstości mapy, przez co wspaniale uzupełnia zestaw zaprezentowanych w tym rozdziale algorytmów.

6.6. Podsumowanie

W rozdziale tym zaprezentowano przegląd metod wykorzystania mapy fotonów do wizualizacji oświetlenia pośredniego. Opisano w nim, w jaki sposób mapa fotonów może być użyta do wyznaczenia radiancji w dowolnym punkcie sceny, zbudowania dobrej funkcji rozkładu promieni wtórnych na powierzchniach o charakterystyce rozproszonej, czy generowania bardzo dokładnych kaustyk. Pokazano również kilka prostych usprawnień pozwalających poprawić jakość uzyskiwanych wyników i szybkość ich generowania. Znalazły się w nim w końcu dwa algorytmy tworzenia obrazów zawierających pełne oświetlenie globalne. Wykorzystują one mapy fotonów na wiele sposobów, także w czasie obliczania oświetlenia bezpośredniego. Obydwa charakteryzują się prostotą i efektywnością. Pierwszy nadaje się szczególnie do prostszych modeli oraz szybkiego generowania podglądu oświetlenia globalnego. Drugi jest prawdopodobnie najlepszym znanym aktualnie algorytmem renderowania oświetlenia globalnego. Dzięki wielu rozszerzeniom ([19], [21], [22], [23]) pozwala odtwarzać praktycznie wszystkie zjawiska świetlne zachodzące w przyrodzie.

Mapa fotonów została zaprojektowana głównie z myślą o wyznaczaniu oświetlenia pośredniego. Zgromadzony w tym rozdziale materiał udowadnia, że jest ona najskuteczniejszym i najwszechstronniejszym istniejącym narzędziem wykorzystywanym w tym celu.

Rozdział 7.

Podsumowanie

Tworzenie realistycznych obrazów świata rzeczywistego z wykorzystaniem komputerów jest przedmiotem badań naukowych od bardzo dawna. W wielu gałęziach tej dziedziny rozwój postępował bardzo dynamicznie. Między innymi szybko skonstruowano struktury danych pozwalające w efektywny sposób reprezentować geometrię wizualizowanej sceny. Powstały modele odbicia dokładnie odwzorowujące interakcję, jaka zachodzi pomiędzy światłem a obiektami, które napotyka ono na swojej drodze. Stworzono urządzenia mierzące właściwości rzeczywistych obiektów.

Również techniki renderingu rozwijały się bardzo szybko. W początkowej fazie tworzone głównie algorytmy dające akceptowalne rezultaty, nie zważając na poprawność fizyczną generowanych obrazów. Powstawały również techniki symulacji określonych elementów oświetlenia globalnego, jak wielokrotnych odbić zwierciadlanych czy delikatnych odbić czysto rozproszonych. Gwałtowny wzrost mocy obliczeniowych komputerów umożliwiał łączenie tych technik i tworzenie coraz dokładniejszych rozwiązań.

Niestety, przez długi czas nie udawało się wynaleźć szybkiej i dokładnej metody pełnego odwzorowywania oświetlenia globalnego. Szczególne trudności sprawiało delikatne oświetlenie pochodzące od wielokrotnych odbić rozproszonych oraz specjalne efekty świetlne, jak chociażby kaustyki. Dopiero zaprezentowana w roku 1995 w pracy [18] mapa fotonów okazała się być w tej dziedzinie przełomem. Struktura ta, skonstruowana początkowo z myślą o oświetleniu pośrednim, okazała się być bardzo efektywna i niezwykle elastyczna. W krótkim czasie wymyślono dla niej szereg nowych zastosowań. Powstało też mnóstwo jej modyfikacji, pozwalających symulować coraz bardziej skomplikowane efekty świetlne z niespotykaną dotąd dokładnością i szybkością.

W pracy zawarto przegląd technik tworzenia i wykorzystania map fotonów oraz pokrewnych im struktur danych w renderingu. Opisano metody wspierania wyznaczania oświetlenia bezpośredniego i pośredniego, tworzenia specjalnych efektów świetlnych oraz kompletne algorytmy renderowania. Omówione techniki zostały zaimplementowane i dokładnie przetestowane, a wyniki tych testów przedstawiono w pracy. Wszystkie opisane algorytmy okazały się spełniać pokładane w nich nadzieje. W przypadku oświetlenia bezpośredniego wykorzystanie map fotonów pozwoliło znacząco skrócić czas renderingu bez utraty jakości wyniku. Natomiast przedstawione metody generowania

oświetlenia pośredniego i globalnego wprowadziły wręcz nową, niespotykaną w przypadku innych rozwiązań jakość i to zarówno biorąc pod uwagę czas renderingu jak i efekt końcowy.

Przytoczone w pracy algorytmy stanowią jedynie trzon metod wykorzystujących mapy fotonów. W zaledwie dziesięć lat skonstruowano wokół niego olbrzymią ilość dodatkowych rozwiązań, które obejmują między innymi szybki rendering specjalnych efektów świetlnych oraz animacji, rendering rozproszony przy pomocy wielu komputerów, czy tworzenie realistycznych obrazów w czasie rzeczywistym. Większość z tych technik nie pozostała jedynie w sferze teoretycznych rozważań, lecz szybko znalazła zastosowanie w systemach wykorzystywanych produkcyjnie, między innymi przy tworzeniu filmów animowanych.

Mapy fotonów są bez wątpienia bardzo wartościową strukturą danych. Jej wprowadzenie przyniosło olbrzymi postęp w realistycznej grafice komputerowej.

Dodatek A

Zaimplementowane programy

W celu przetestowania opisywanych w pracy algorytmów stworzono dwa programy komputerowe. Dodatek ten stanowi skrót dokumentacji technicznej oraz instrukcji użytkownika tych programów.

A1. Informacje ogólne

W trakcie tworzenia pracy zaimplementowano dwa programy komputerowe. Stanowią one jej integralną część i pochłonęły spory odsetek czasu przeznaczonego na jej wykonanie. Są to:

- **mgfview** – interaktywny program służący do wyświetlania trójwymiarowych scen zapisanych w formacie **mgf**,
- **rat** – wsadowy program renderujący obrazy na podstawie skryptu sterującego oraz opisu sceny (w formacie **mgf**).

Obydwa programy zrealizowano w języku C, zgodnie ze standardem ISO C 99. Jest to współczesne rozszerzenie dobrze znanego standardu ANSI, implementowane przez większość nowoczesnych kompilatorów. Obydwa programy są niezależne od platformy użytkownika. Jednak rozwijane były przy wykorzystaniu systemu operacyjnego Linux opartego na jądrze 2.6.7 oraz kompilatora gcc w wersji 3.3.4. i w tej konfiguracji działają najstabilniej.

Obydwa programy wykorzystują popularny w realistycznej grafice komputerowej format opisu sceny **mgf** (*Materials and Geometry Format* [36]). Do poprawnej kompilacji i działania wymagają one następujących dodatkowych bibliotek:

- **mgflib** – zawiera implementację i interfejsy parsera formatu **mgf**, korzystają z niej obydwie programy w celu wczytania sceny,
- **gl** – zawiera implementację oraz interfejsy **OpenGL**, korzysta z niej tylko program **mgfview**, w celu wyświetlenia trójwymiarowego modelu sceny,

- **glut** – zawiera implementację oraz interfejsy prostego systemu okien, korzysta z niej tylko program **mgfview**, w celu wyświetlenia interfejsu użytkownika,
- **glu** – zawiera implementację dodatkowych funkcji graficznych, korzystają z niej obydwie programy w celu podziału sceny na trójkąty.

Pierwsza z wymienionych bibliotek została dołączona do źródeł programu. Trzy pozostałe dostarczane są przez twórców konkretnego systemu operacyjnego bądź sprzętu (najczęściej karty graficznej).

A2. mgfview

Program służy do wyświetlania trójwymiarowych scen. Wykorzystuje do tego biblioteki z rodziny **OpenGL**. Formatem wejściowym jest **mgf**. Program musi być uruchamiany z konsoli, gdyż działa poprzez interpretowanie danych przekazanych na standardowe wejście. Przykładowe wywołanie programu może wyglądać następująco:

```
mgfview < file.mgf
```

Spowoduje ono uruchomienie programu oraz zinterpretowanie i wyświetlenie modelu zawartego w pliku **file.mgf**. Niektóre modele zapisane są w kilku osobnych plikach. Wyświetlenie takiego modelu może być zrealizowane poprzez przekierowanie połączonych plików na wejście programu:

```
cat file1.mgf file2.mgf | mgfview
```

Ten sam efekt można uzyskać poprzez stworzenie specjalnego pliku **mgf** włączającego w swoim ciele wszystkie potrzebne pliki:

file.mgf:

```
i file1.mgf
i file2.mgf
```

i wczytanie tego pliku do programu.

Program umożliwia swobodne poruszanie się po całej scenie we wszystkich kierunkach, oglądanie jej w różnych konfiguracjach wyświetlania oraz zapis informacji o bieżącym położeniu kamery w formacie zgodnym z programem **rat**. Aplikację obsługuje się przy pomocy następujących klawiszy:

← – ruch w lewo	Ctrl+← – obrót w lewo (oś y)	F1 – tryb wireframe/solid
→ – ruch w prawo	Ctrl+→ – obrót w prawo (oś y)	F2 – zmienia tryb oświetlenia
↑ – ruch do przodu	Ctrl+↑ – obrót w górę (oś x)	F3 – zmienia tryb cieniowania
↓ – ruch w tył	Ctrl+↓ – obrót w dół (oś x)	F5 – zapis informacji o bieżącym
Page Up – ruch w górę	Insert – obrót w lewo (oś z)	położeniu kamery na
Page Down – ruch w dół	Home – obrót w prawo (oś z)	standardowe wyjście programu

A3. rat

Program renderujący realistyczne obrazy komputerowe na podstawie opisu scen w formacie **mgf** oraz skryptów sterujących napisanych w specjalnie stworzonym języku wewnętrznym. Wynikiem działania programu są obrazy zapisane w formacie **hdr**. Istnieje wiele narzędzi pozwalających na oglądanie i obróbkę obrazów zapisanych w tym formacie. Jedno z nich (**vrpic**) dołączono do źródeł programu. Przykładowe wywołanie programu wygląda następująco:

```
rat script.rat
```

W tym przypadku zostanie zinterpretowany skrypt zawarty w pliku **script.rat**. Program akceptuje tylko jeden skrypt w pojedynczym wywołaniu. Może on być podany jako pierwszy parametr wywołania programu, lub przekazany na jego standardowe wejście. Druga z opcji umożliwia pracę interaktywną. Na płycie dołączonej do pracy znajdują się skrypty pozwalające wyrenderować wszystkie prezentowane w niej obrazy.

A3.1. Składnia języka

Język skryptowy wykorzystywany przez program składa się z listy poleceń, które wykonywane są w kolejności pojawiania się w pliku. Każde polecenie musi być zapisane w jednym wierszu, którego długość nie przekracza 1024 znaków. Polecenie składa się z nazwy i listy parametrów. Parametr jest parą nazwa–wartość rozdzieloną znakiem **=**. Parametr może być jednego z czterech typów: liczba całkowita, liczba zmiennopozycyjna, ciąg znaków lub referencja. Format obydwu typów liczbowych jest zdefiniowany jako dowolny ciąg znaków poprawnie parsowany przez funkcję **scanf** z biblioteki standardowej języka C. Ciąg znaków składa się z dowolnych znaków zawartych między cudzysłowami **""**. Referencja składa się z poprawnego identyfikatora zawartego pomiędzy znakami **'?'**. Identyfikator musi odnosić się do wartości zwróconej przez wcześniej wykonaną komendę i zapisanej w środowisku skryptu pod tą samą nazwą. Możliwe jest przekazanie pustej referencji poprzez użycie konstrukcji **'??'**. Każdy wiersz zawierający polecenie musi kończyć się znakiem **';**. Wszystko, co znajduje się za tym znakiem, jest ignorowane. Linie rozpoczynające się znakiem **#** zawierają komentarze i są ignorowane.

Pojedyncza komenda ma więc następujący format:

```
KOMENDA (= REZULTAT)? (PARAMETR = (int|float|"x*"|?y*?))* ;
```

gdzie:

- **KOMENDA** – jest identyfikatorem komendy,
- **REZULTAT** – nazwa pod jaką w środowisku zostanie zapisany wynik, jeśli ta część jest pominięta wynik nie zostanie zapisany wcale,
- **PARAMETR** – jest identyfikatorem parametru,

Identyfikator może składać się jedynie z dużych i małych liter. Zapisanie w środowisku wyniku komendy pod nazwą, pod którą już wcześniej była zapisana jakaś wartość, spowoduje usunięcie poprzedniej wartości. Próba wykorzystania jako parametru referencji do nazwy, która nie została wcześniej zapisana, zakończy się błędem. Wszystkie referencje zapisane w środowisku posiadają typ. Przekazanie jako parametru referencji złego typu spowoduje błąd wykonania skryptu. Referencja pusta (??) jest automatycznie konwertowana do odpowiedniego typu i zawsze może być użyta.

A3.2. Spis komend

W paragrafie tym umieszczono spis komend rozpoznawanych przez program renderujący **rat**. Z powodu dużej ilości dostępnych poleceń ograniczono się do opisanego zestawu najistotniejszych z nich, dotyczących ładowania sceny i renderowania obrazów przy pomocy mapy fotonów. Pominęto natomiast całkowicie komendy umożliwiające renderowanie z użyciem śledzenia ścieżek oraz większość związanych z wyliczaniem oświetlenia bezpośredniego. Polecenia są podane w następującej formie:

- **Komenda** : **TYP WYNIKU** — nazwa komendy, po której znajduje się typ zwracanego wyniku i krótki opis działania.

1. **Parametr 1** : **TYP PARAMETRU** — lista parametrów wraz z typami i krótkim opisem znaczenia.

Komendy operujące na modelu sceny:

- **scenelnit** : **SCENE REF** — tworzy nowy obiekt sceny.
- **sceneLoadFile** — ładuje model z pliku do podanej sceny. Komenda może być wywołana kilka razy dla tego samego obiektu sceny w celu załadowania modelu znajdującego się w wielu plikach.
 1. **object** : **SCENE REF** — referencja do obiektu sceny, do którego ma zostać załadowany model.
 2. **file** : **STRING** — nazwa pliku zawierającego model.
 3. **divs** : **INT** — program wykonuje podział skomplikowanych kształtów geometrycznych na trójkąty. Parametr określa ilość wierzchołków przypadających na zmianę kąta o 90°.
 4. **minTrArea** : **FLOAT** — minimalna wielkość trójkąta. Trójkąty o mniejszym polu są odrzucane.
- **sceneFinalize** — wykonuje wstępne obliczenia na wczytanym modelu. Komendę tą trzeba wywołać po załadowaniu wszystkich plików, a przed rozpoczęciem innej operacji (jak rendering czy tworzenie mapy fotonów). Po wykonaniu tej operacji do obiektu sceny nie można już wczytać nowego modelu.
 1. **object** : **SCENE REF** — referencja do obiektu sceny.
- **sceneSetShader** — ustawia obiekt odpowiedzialny za renderowanie oświetlenia bezpośredniego.
 1. **object** : **SCENE REF** — referencja do obiektu sceny.
 2. **shader** : **SHADER REF** — referencja do obiektu shader.
- **sceneSetGlobalMap** — ustawia mapę fotonów.
 1. **object** : **SCENE REF** — referencja do obiektu sceny.
 2. **map** : **PHOTONMAP REF** — referencja do obiektu mapy fotonów.
- **sceneSetCausticMap** — ustawia mapę kaustyczną.
 1. **object** : **SCENE REF** — referencja do obiektu sceny.
 2. **map** : **PHOTONMAP REF** — referencja do obiektu mapy kaustycznej.

- **sceneSetImportanceMap** — ustawia mapę istotności.
 1. **object** : SCENE REF — referencja do obiektu sceny.
 2. **map** : PHOTONMAP REF — referencja do obiektu mapy istotności.
- **sceneSetCamera** — ustawia kamerę, ustawienia tej kamery będą wykorzystane przy tworzeniu mapy istotności.
 1. **object** : SCENE REF — referencja do obiektu sceny.
 2. **camera** : CAMERA REF — referencja do obiektu kamery.
- **sceneSetRefraction** — ustawia obiekt odpowiedzialny za obsługę refrakcji.
 1. **object** : SCENE REF — referencja do obiektu sceny.
 2. **refraction** : REFRACTION REF — referencja do obiektu refrakcji.

Komendy sterujące ustawieniami kamery:

- **cameralnit** : CAMERA REF — tworzy nowy obiekt kamery.
 1. **aspect** : FLOAT — stosunek wysokości do szerokości obrazu.
 2. **vAngle** : FLOAT — kąt widzenia w poziomie (w stopniach).
 3. **nearcp** : FLOAT — przednia płaszczyzna odcinająca.
 4. **farcp** : FLOAT — tylna płaszczyzna odcinająca.
- **cameraMove** — przemieszcza kamerę we wskazane miejsce.
 1. **object** : CAMERA REF — referencja do obiektu kamery.
 2. **x** : FLOAT — współrzędna x nowego położenia.
 3. **y** : FLOAT — współrzędna y nowego położenia.
 4. **z** : FLOAT — współrzędna z nowego położenia.
- **cameraSetCs** — ustawia kierunek patrzenia.
 1. **object** : CAMERA REF — referencja do obiektu kamery.
 2. **vx** : FLOAT — współrzędna x wektora w kierunku patrzenia.
 3. **vy** : FLOAT — współrzędna y wektora w kierunku patrzenia.
 4. **vz** : FLOAT — współrzędna z wektora w kierunku patrzenia.
 5. **ux** : FLOAT — współrzędna x wektora w górę.
 6. **uy** : FLOAT — współrzędna y wektora w górę.
 7. **uz** : FLOAT — współrzędna z wektora w górę.
- **cameraSetRes** — ustawia rozdzielczość generowanego obrazu, ta rozdzielczość będzie wykorzystana do określenia rozmiaru piksela w czasie tworzenia mapy fotonów z wykorzystaniem kontroli gęstości.
 1. **object** : CAMERA REF — referencja do obiektu kamery.
 2. **xres** : INT — rozdzielczość pozioma.
 3. **yres** : INT — rozdzielczość pionowa.

Komendy operujące na mapie fotonów:

- **photonmaplnit** : PHOTONMAP REF — tworzy nową mapę fotonów. Wywołanie tej operacji spowoduje wyemitowanie fotonów oraz zbudowanie zbalansowanej struktury danych gotowej do renderingu. Komenda obsługuje tworzenie wszystkich typów map fotonów. W przypadku wywołania dla zwykłej mapy fotonów, jeśli wcześniej była utworzona i zachowana w scenie mapa istotności, to mapa fotonów będzie tworzona z wykorzystaniem kontroli gęstości.

1. `mapType` : `STRING` — typ mapy, dozwolone typy to: `global` (zwykła mapa), `caustic` (mapa kaustyczna), `shadow` (mapa cienia) oraz `importance` (mapa istotności).
 2. `scene` : `SCENE REF` — referencja do sceny, dla której ma być zbudowana mapa.
 3. `numPhotons` : `INT` — liczba fotonów, które będą zachowane w mapie.
 4. `incDirIllum` : `INT` — 0 oznacza, że fotony oświetlenia bezpośredniego nie będą zachowane w mapie.
 5. `useAvgPower` : `INT` — 0 oznacza wykorzystanie algorytmu `max` do wyznaczania typu interakcji fotonu z powierzchnią, każda inna wartość algorytm `avg` (zob. 4.2.2.).
 6. `maxPerRay` : `INT` — maksymalna ilość fotonów zachowywanych wzdłuż jednej ścieżki.
 7. `mcaustic` : `FLOAT` — minimalna wartość odbicia zwierciadlanego dla obiektów kaustycznych. Parametr istotny tylko w przypadku mapy kaustycznej.
- `photonmapCreateFlags` : `PHOTONFLAGS REF` — tworzy flagi służące do filtrowania fotonów.
 1. `photonType` : `STRING` — typ fotonu, dozwolone typy to: `direct` (bezpośredni), `caustic` (kaustyczny), `pure indirect` (pośredni, ale nie kaustyczny), `indirect` (dowolny pośredni), `direct and caustic` (bezpośredni lub kaustyczny), `direct and pure indirect` (bezpośredni lub pośredni ale nie kaustyczny), `any` (dowolny).
 2. `lightNumber` : `INT` — numer źródła światła, -1 oznacza wszystkie.
 - `photonmapGatherOptsInit` : `PHOTONGATHER REF` — tworzy obiekt zawierający opcje wyszukiwania fotonów.
 1. `filter` : `STRING` — wykorzystywany filtr, dozwolone typy: `NULL` (brak), `cone` (stożkowy), `gaussian` (Gaussa).
 2. `flags` : `PHOTONFLAGS REF` — flagi wykorzystywane do odfiltrowywania fotonów.
 3. `maxDist` : `FLOAT` — promień kuli, w której wyszukiwane są fotony.
 4. `flattenFrac` : `FLOAT` — współczynnik kompresji kuli (zob. 6.3.2.).
 5. `maxPhotons` : `INT` — maksymalna liczba odnajdywanych fotonów.
 6. `minPhotons` : `INT` — minimalna liczba fotonów. Wyszukiwanie zakończy się niepowodzeniem, jeśli w podanym obszarze jest ich mniej.
 7. `allowZero` : `INT` — jeśli $\neq 0$, to dozwolone są obszary, w których nie ma żadnego fotonu. Parametr używany tylko w mapie cienia.
 8. `useEstimates` : `INT` — jeśli $\neq 0$, to w czasie renderingu wykorzystane zostaną oszacowania irradiancji zapisane w fotonach (zob. 6.5.1.), jeśli oszacowania nie były wyliczone wcześniej dla całej mapy (polecenie `photonmapCalcIrradiances`), to będą automatycznie wyznaczane w trybie „na żądanie”.
 9. `normDevCos` : `FLOAT` — cosinus maksymalnego odchylenia normalnej wykorzystywanego oszacowania. Parametr istotny tylko jeśli używane są oszacowania irradiancji.
 - `photonmapCalcIrradiances` — powoduje wyliczenie irradiancji we wszystkich punktach, w które trafiły fotony.
 1. `object` : `PHOTONMAP REF` — mapa fotonów.
 2. `gatherOpts` : `PHOTONGATHER REF` — opcje odnajdywania fotonów.
 - `photonmapCalcImportances` — powoduje wyliczenie istotności we wszystkich punktach, w które trafiły importony. Uwaga, oszacowania istotności nie są wyznaczane w trybie na żądanie, więc ta operacja musi być wywołana przed rozpoczęciem tworzenia mapy fotonów.
 1. `object` : `PHOTONMAP REF` — mapa fotonów.
 2. `gatherOpts` : `PHOTONGATHER REF` — opcje odnajdywania importonów.
 - `photonmapVisSimple` : `RGBBUFFER REF` — tworzy obraz sceny przy pomocy algorytmu bezpośredniej wizualizacji mapy fotonów.
 1. `scene` : `SCENE REF` — referencja do sceny.
 2. `camera` : `CAMERA REF` — kamera, definiuje pozycję obserwatora i kierunek patrzenia.
 3. `gatherOpts` : `PHOTONGATHER REF` — opcje odnajdywania fotonów.
 4. `causticOpts` : `PHOTONGATHER REF` — opcje renderowania kaustyk.

5. `rlights` : INT — jeśli $\neq 0$, to do utworzenia oświetlenia bezpośredniego wykorzystany zostanie shader zapisany w scenie.
 6. `npaths` : INT — ilość promieni wtórnych generowanych w przypadku napotkania obiektu zwierciadlanego.
 7. `width` : INT — szerokość obrazu.
 8. `height` : INT — wysokość obrazu.
- `photonmapRender` : RGBBUFFER REF — tworzy obraz sceny przy pomocy algorytmu dwufazowego.
 1. `scene` : SCENE REF — referencja do sceny.
 2. `camera` : CAMERA REF — kamera, definiuje pozycję obserwatora i kierunek patrzenia.
 3. `gatherOpts` : PHOTONGATHER REF — opcje odnajdywania fotonów.
 4. `causticOpts` : PHOTONGATHER REF — opcje renderowania kaustyk.
 5. `dgatherOpts` : PHOTONGATHER REF — opcje odnajdywania fotonów do tworzenia funkcji rozkładu promieni (zob. 6.2.).
 6. `tdivs` : INT — ilość przedziałów funkcji rozkładu promieni w kierunku θ .
 7. `pdivs` : INT — ilość przedziałów funkcji rozkładu promieni w kierunku ϕ .
 8. `rlights` : INT — jeśli $\neq 0$, to do utworzenia oświetlenia bezpośredniego wykorzystany zostanie shader zapisany w scenie.
 9. `dpaths` : INT — ilość promieni wtórnych generowanych w przypadku napotkania obiektu o odbiciu rozproszonym.
 10. `spaths` : INT — ilość promieni wtórnych generowanych w przypadku napotkania obiektu zwierciadlanego.
 11. `width` : INT — szerokość obrazu.
 12. `height` : INT — wysokość obrazu.

Komendy sterujące generowaniem oświetlenia bezpośredniego:

- `shaderInit` : SHADER REF — tworzy obiekt odpowiedzialny za renderowanie oświetlenia bezpośredniego.
 1. `scene` : SCENE REF — referencja do sceny, dla której tworzony jest obiekt.
 2. `photonMap` : PHOTONMAP REF — referencja do mapy fotonów (niektóre z algorytmów korzystają z niej do wyznaczenia oświetlenia).
 3. `shadowMap` : PHOTONMAP REF — referencja do mapy cienia (niektóre z algorytmów korzystają z niej do wyznaczenia oświetlenia).
- `shaderPFacOpts` : POSFACTOR REF — tworzy obiekt zawierający opcje ustalania istotności światła
 1. `method` : STRING — metoda ustalania istotności, dozwolone wartości: `position` (na podstawie pozycji), `photon map` (na podstawie mapy fotonów), `shadow map` (na podstawie mapy cienia) (zob. 5.3., 5.4., [29]).
 2. `gatherOpts` : PHOTONGATHER REF — opcje zbierania fotonów, wykorzystywane w przypadku używania mapy fotonów lub cienia.
 3. `usePosition` : INT — jeśli $\neq 0$, to wybrana metoda oparta na mapie będzie złożona z metodą pozycyjną.
- `shaderSetupRT` — ustawia konfigurację wyliczania oświetlenia bezpośredniego przy pomocy śledzenia promieni.
 1. `object` : SHADER REF — referencja do obiektu shader.
 2. `depth` : INT — głębokość rekursji, na której ta konfiguracja będzie używana. Shader może wykorzystywać inną konfigurację dla każdej głębokości rekursji. -1 oznacz konfigurację domyślną wykorzystywaną zawsze, gdy nie jest zdefiniowana konfiguracja dla bieżącej głębokości.
 3. `posFactorOpts` : POSFACTOR REF — referencja do obiektu opcji wykorzystywanych do obliczenia względnej istotności światła.
 4. `method` : STRING — metoda, dozwolone wartości: `visibility` (ustal tylko widoczność środka światła), `single light` (renderuj każde światło osobno), `multi light uniform` (renderuj wszystkie światła razem, wszystkie są tak samo istotne), `multi light power` (renderuj wszystkie światła razem, mocniejsze światła są bardziej istotne), `multi light importance` (renderuj wszystkie światła razem, wykorzystaj opcje `posFactorOpts` do określenia istotności światła) (zob. [6], [29]).

5. `numberOfRays` : INT — ilość wykorzystywanych promieni.

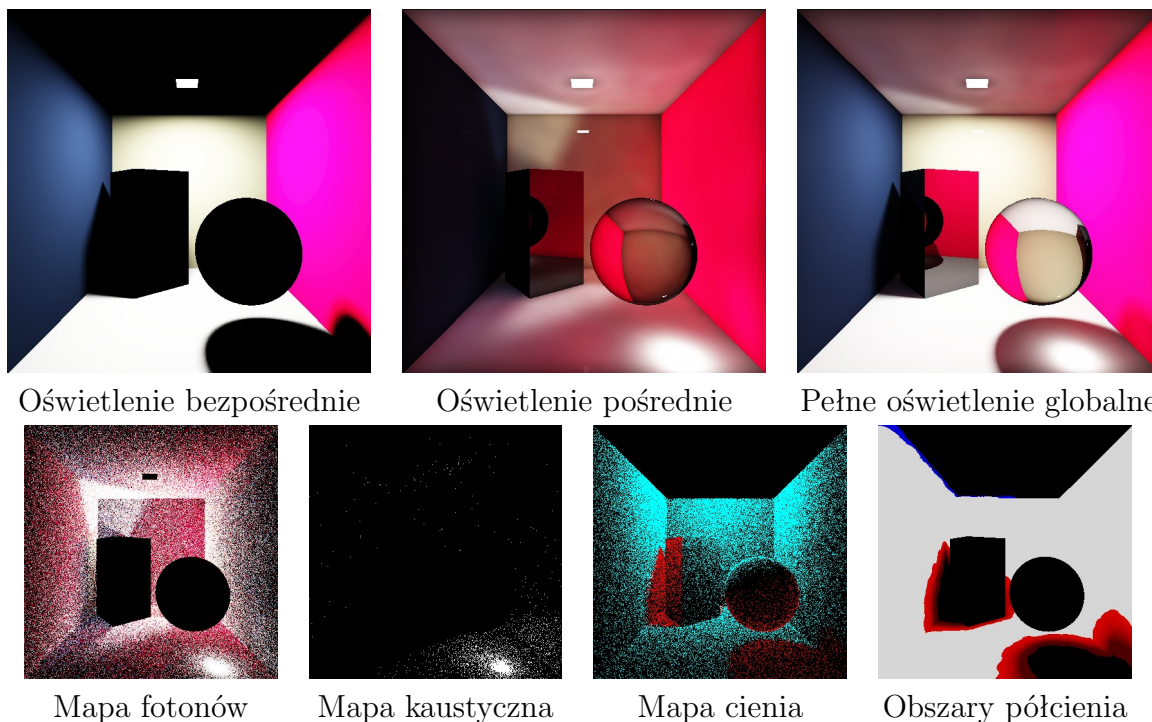
Inne komendy:

- `refractionInit` : REFRACTION REF — tworzy obiekt odpowiedzialny za obsługę refrakcji.
 1. `environment` : FLOAT — współczynnik refrakcji ośrodka, w którym znajduje się scena.
 2. `stackSize` : INT — maksymalna głębokość zagnieżdżenia obiektów załamujących światło.
- `hdrWriteData` — zapisuje do pliku zawartość bufora z wyrenderowanym obrazem.
 1. `file` : STRING — nazwa pliku.
 2. `buffer` : RGBBUFFER REF — referencja do bufora zawierającego obraz.
 3. `msg` : STRING — komentarz (dodawany do pliku).

A3.3. Przykładowe skrypty i wyniki renderingu

W tym paragrafie przedstawiono dwa skrypty programu `rat`, tworzone przez nie obrazy oraz trochę statystyk dotyczących renderingu. Pomiary czasu wykonano na procesorze AMD Duron 850MHz. W przytoczonych skryptach zbyt długie wiersze kodu podzielono na kilka osobnych. Takie wiersze oznaczono znakiem $\leftrightarrow$ w punkcie podziału.

Zwierciadlana wersja sceny Cornell box



Rysunek A1.: Zwierciadlana wersja sceny Cornell box.

```

sceneInit = scene ;
sceneLoadFile object = ?scene? file = "caustic.mgf" divs = 25 minTrArea = 0.0 ;
sceneFinalize object = ?scene? ;

cameraInit = camera aspect = 1.00000 vAngle = 70.00000 nearcp = 0.10000 farcp = 25.00000 ; .
cameraMove object = ?camera? x = 5.00000 y = -7.10001 z = 5.05000 ; .
cameraSetCs object = ?camera? vx = 0.00000 vy = 1.00000 vz = 0.00000 ux = -0.00000 ↔
    uy = -0.00000 uz = 1.00000 ; .

refractionInit = refraction environment = 1.0 stackSize = 32 ;
sceneSetRefraction object = ?scene? refraction = ?refraction? ;
photonmapInit = global mapType = "global" scene = ?scene? numPhotons = 300000 ↔
    incDirIllum = 0 useAvgPow = 0 maxPerRay = 10 mcaustic = 0.0 ;
sceneSetGlobalMap object = ?scene? map = ?global? ;

photonmapInit = caustic mapType = "caustic" scene = ?scene? numPhotons = 5000 ↔
    incDirIllum = 1 useAvgPow = 0 maxPerRay = 10 mcaustic = 0.8 ;
sceneSetCausticMap object = ?scene? map = ?caustic? ;

photonmapInit = shadow mapType = "shadow" scene = ?scene? numPhotons = 100000 ↔
    incDirIllum = 1 useAvgPow = 0 maxPerRay = 10 mcaustic = 0.0 ;

photonmapCreateFlags = flagsAll photonType = "any" lightNumber = -1 ;
photonmapShowPhotons = vis object = ?global? flags = ?flagsAll? camera = ?camera? ↔
    width = 395 height = 395 ;
hdrWriteData file = "global-photons.hdr" buffer = ?vis? msg = "Photons" ;
colorFreeRGBBuffer object = ?vis? ;

photonmapCreateFlags = flagsAll photonType = "any" lightNumber = -1 ;
photonmapShowPhotons = vis object = ?caustic? flags = ?flagsAll? camera = ?camera? ↔
    width = 395 height = 395 ;
hdrWriteData file = "caustic-photons.hdr" buffer = ?vis? msg = "Caustic photons" ;
colorFreeRGBBuffer object = ?vis? ;

photonmapCreateFlags = flagsAll photonType = "any" lightNumber = -1 ;
photonmapShowPhotons = vis object = ?shadow? flags = ?flagsAll? camera = ?camera? ↔
    width = 395 height = 395 ;
hdrWriteData file = "shadow-photons.hdr" buffer = ?vis? msg = "Shadow photons" ;
colorFreeRGBBuffer object = ?vis? ;

photonmapGatherOptsInit = gopts filter = "NULL" flags = ?? maxDist = 0.8 flattenFac = 1.0 ↔
    maxPhotons = 100 minPhotons = 10 allowZero = 1 useEstimates = 0 normDevCos = 0.1 ;
photonmapVisShadowrays = vis scene = ?scene? camera = ?camera? map = ?shadow? ↔
    gatherOpts = ?gopts? lnum = 0 width = 395 height = 395 ;
hdrWriteData file = "rays-all.hdr" buffer = ?vis? msg = "Estimated shadow regions" ;
colorFreeRGBBuffer object = ?vis? ;

shaderInit = shader scene = ?scene? photonMap = ?? shadowMap = ?shadow? ;
shaderSetupRT object = ?shader? depth = -1 posFactorOpts = ?? method = "NULL" ↔
    numberOfRays = 1 ;
sceneSetShader object = ?scene? shader = ?shader? ;

photonmapGatherOptsInit = gopts filter = "cone" flags = ?? maxDist = 1.0 flattenFac = 0.1 ↔
    maxPhotons = 1500 minPhotons = 1 allowZero = 0 useEstimates = 0 normDevCos = 0.1 ;
photonmapGatherOptsInit = copts filter = "cone" flags = ?? maxDist = 0.075 flattenFac = 0.0 ↔
    maxPhotons = 300 minPhotons = 50 allowZero = 0 useEstimates = 0 normDevCos = 0.1 ;
photonmapVisSimple = vis scene = ?scene? camera = ?camera? gatherOpts = ?gopts? ↔
    causticOpts = ?copts? rlights = 1 npaths = 1 width = 567 height = 567 ;
hdrWriteData file = "caustic-indirect.hdr" buffer = ?vis? msg = "Indirect illumination" ;
colorFreeRGBBuffer object = ?vis? ;

photonmapGatherOptsInit = gopts filter = "NULL" flags = ?? maxDist = 0.8 flattenFac = 1.0 ↔
    maxPhotons = 100 minPhotons = 10 allowZero = 1 useEstimates = 0 normDevCos = 0.1 ;
shaderSetupSM object = ?shader? depth = -1 method = "absolute" gatherOpts = ?gopts? ↔
    numberOfSamples = 300 numberOfRays = 300 ;
tracePaths = vis scene = ?scene? camera = ?camera? numberOfPaths = 0 excludeDirect = 0 ↔
    excludeCaustics = 1 width = 567 height = 567 ;
hdrWriteData file = "caustic-direct.hdr" buffer = ?vis? msg = "Direct illumination" ;
colorFreeRGBBuffer object = ?vis? ;

```

```

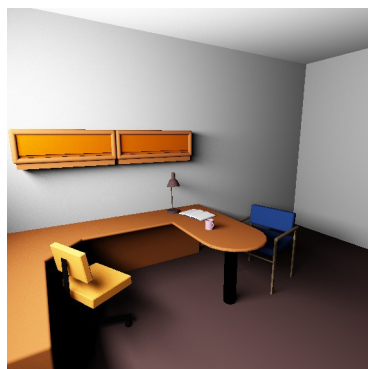
photonmapGatherOptsInit = gopts filter = "cone" flags = ?? maxDist = 1.0 flattenFac = 0.1 ↔
    maxPhotons = 1500 minPhotons = 1 allowZero = 0 useEstimates = 0 normDevCos = 0.1 ;
photonmapGatherOptsInit = copts filter = "cone" flags = ?? maxDist = 0.075 flattenFac = 0.0 ↔
    maxPhotons = 300 minPhotons = 50 allowZero = 0 useEstimates = 0 normDevCos = 0.1 ;
photonmapVisSimple = vis scene = ?scene? camera = ?camera? gatherOpts = ?gopts? ↔
    causticOpts = ?copts? rlights = 1 npaths = 1 width = 567 height = 567 ;
hdrWriteData file = "caustic.hdr" buffer = ?vis? msg = "Full global illumination" ;

```

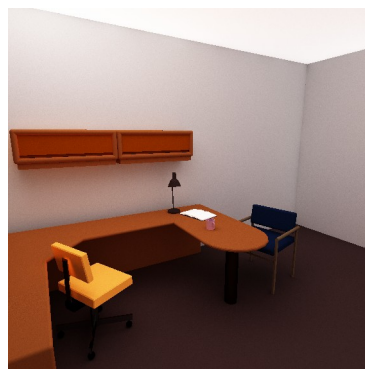
Scena została wyrenderowana w rozdzielczości 567×567 pikseli. Oświetlenie bezpośrednie utworzono przy pomocy mapy cienia zawierającej 1×10^5 fotonów cienia oraz 300 promieni cienia w obszarach półcienia. Oświetlenie pośrednie wygenerowano algorytmem bezpośredniej wizualizacji mapy fotonów. Mapa zawierała 3×10^5 fotonów. W czasie renderingu wykorzystano do 1500 najbliższych fotonów. Konieczność użycia aż tylu fotonów wynikała z dużej ilości obiektów o charakterze zwierciadlanym, które generują delikatne kaustyki (np.: na suficie). Kaustyka na podłodze pochodząca od światła załamane w szklanej kuli została utworzona przy pomocy oddzielnej mapy kaustycznej zawierającej 5×10^3 fotonów.

Czas tworzenia mapy fotonów wynosił 14.22s, mapy kaustycznej 0.74s a mapy cienia 2.24s. Wykrywanie obszarów półcienia trwało 20.01s, a całkowity czas tworzenia oświetlenia bezpośredniego to 2m 2.47s. Generowanie oświetlenia pośredniego zajęło 33m 39.84s. Całkowity czas tworzenia obrazu to 37m 22.54s.

Wnętrze biura oświetlone światłem dziennym



Oświetlenie bezpośrednie



Oświetlenie pośrednie



Pełne oświetlenie globalne



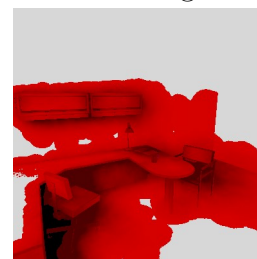
Mapa fotonów



Irradiancja



Istotność



Obszary półcienia

Rysunek A2.: Wnętrze biura oświetlone światłem dziennym.

```

sceneInit = scene ;
sceneLoadFile object = ?scene? file = "office.mgf" divs = 5 minTrArea = 0.0 ;
sceneFinalize object = ?scene? ;

cameraInit = camera aspect = 1.00000 vAngle = 70.00000 nearcp = 0.10000 farcp = 25.00000 ; .
cameraMove object = ?camera? x = 0.69904 y = 2.77383 z = 1.62624 ; .
cameraSetCs object = ?camera? vx = -0.84710 vy = -0.20791 vz = -0.48907 ux = -0.18006 ↔
    uy = 0.97815 uz = -0.10396 ; .
cameraSetRes object = ?camera? xres = 300 yres = 300 ;
sceneSetCamera object = ?scene? camera = ?camera? ;

refractionInit = refraction environment = 1.0 stackSize = 32 ;
sceneSetRefraction object = ?scene? refraction = ?refraction? ;

photonmapInit = importance mapType = "importance" scene = ?scene? numPhotons = 30000 ↔
    incDirIllum = 1 useAvgPow = 0 maxPerRay = 10 mcaustic = 0.0 ;
sceneSetImportanceMap object = ?scene? map = ?importance? ;

photonmapGatherOptsInit = gopts filter = "cone" flags = ?? maxDist = 2.0 flattenFac = 0.1 ↔
    maxPhotons = 350 minPhotons = 1 allowZero = 1 useEstimates = 0 normDevCos = 0.1 ;
photonmapCalcImportances object = ?importance? gatherOpts = ?gopts? ;

photonmapInit = global mapType = "global" scene = ?scene? numPhotons = 100000 ↔
    incDirIllum = 1 useAvgPow = 0 maxPerRay = 10 mcaustic = 0.0 ;
sceneSetGlobalMap object = ?scene? map = ?global? ;

photonmapInit = shadow mapType = "shadow" scene = ?scene? numPhotons = 100000 ↔
    incDirIllum = 1 useAvgPow = 0 maxPerRay = 10 mcaustic = 0.0 ;

photonmapCreateFlags = flagsAll photonType = "any" lightNumber = -1 ;
photonmapShowPhotons = vis object = ?global? flags = ?flagsAll? camera = ?camera? ↔
    width = 395 height = 395 ;
hdrWriteData file = "photons.hdr" buffer = ?vis? msg = "Photons" ;
colorFreeRGBBuffer object = ?vis? ;

photonmapGatherOptsInit = gopts filter = "cone" flags = ?? maxDist = 2.0 flattenFac = 0.1 ↔
    maxPhotons = 350 minPhotons = 1 allowZero = 1 useEstimates = 0 normDevCos = 0.1 ;
photonmapVisImportances = irrads scene = ?scene? camera = ?camera? width = 395 height = 395 ;
hdrWriteData file = "importances.hdr" buffer = ?irrads? msg = "Estimated importance" ;

photonmapGatherOptsInit = gopts filter = "cone" flags = ?? maxDist = 0.2 flattenFac = 0.1 ↔
    maxPhotons = 300 minPhotons = 1 allowZero = 1 useEstimates = 1 normDevCos = 0.94 ;
photonmapVisIrradiances = irrads scene = ?scene? camera = ?camera? gatherOpts = ?gopts? ↔
    width = 395 height = 395 ;
hdrWriteData file = "irradiance.hdr" buffer = ?irrads? msg = "Estimated irradiances" ;
colorFreeRGBBuffer object = ?irrads? ;

photonmapGatherOptsInit = gopts filter = "NULL" flags = ?? maxDist = 0.8 flattenFac = 1.0 ↔
    maxPhotons = 100 minPhotons = 10 allowZero = 1 useEstimates = 0 normDevCos = 0.1 ;
photonmapVisShadowrays = vis scene = ?scene? camera = ?camera? map = ?shadow? ↔
    gatherOpts = ?gopts? lnum = 0 width = 395 height = 395 ;
hdrWriteData file = "rays-all.hdr" buffer = ?vis? msg = "Estimated shadow regions" ;
colorFreeRGBBuffer object = ?vis? ;

shaderInit = shader scene = ?scene? photonMap = ?? shadowMap = ?? ;
shaderSetupRT object = ?shader? depth = -1 posFactorOpts = ?? method = "NULL" ↔
    numberOfRays = 1 ;
sceneSetShader object = ?scene? shader = ?shader? ;

photonmapGatherOptsInit = gopts filter = "cone" flags = ?? maxDist = 0.2 flattenFac = 0.1 ↔
    maxPhotons = 300 minPhotons = 1 allowZero = 1 useEstimates = 1 normDevCos = 0.94 ;
photonmapCreateFlags = indirectPhotons photonType = "indirect" lightNumber = -1 ;
photonmapGatherOptsInit = dgopts filter = "cone" flags = ?indirectPhotons? maxDist = 0.5 ↔
    flattenFac = 0.0 maxPhotons = 65 minPhotons = 10 allowZero = 0 useEstimates = 0 ↔
    normDevCos = 0.1 ;
photonmapRender = vis scene = ?scene? camera = ?camera? gatherOpts = ?gopts? ↔
    causticOpts = ?? dgatherOpts = ?dgopts? tdivs = 4 pdivs = 16 rlights = 0 dpaths = 3000 ↔
    spaths = 1 width = 567 height = 567 ;
hdrWriteData file = "office-indirect.hdr" buffer = ?vis? ↔
    msg = "Indirect illumination in office scene lit by daylight" ;
colorFreeRGBBuffer object = ?vis? ;

```

```

refractionInit = refraction environment = 1.0 stackSize = 32 ;
sceneSetRefraction object = ?scene? refraction = ?refraction? ;

shaderInit = shader scene = ?scene? photonMap = ?? shadowMap = ?shadow? ;
sceneSetShader object = ?scene? shader = ?shader? ;

photonmapGatherOptsInit = gopts filter = "NULL" flags = ?? maxDist = 0.8 flattenFac = 1.0 ↔
    maxPhotons = 100 minPhotons = 10 allowZero = 1 useEstimates = 0 normDevCos = 0.1 ;
shaderSetupSM object = ?shader? depth = -1 method = "absolute" gatherOpts = ?gopts? ↔
    numberOfSamples = 1000 numberOfRays = 1000 ;
tracePaths = vis scene = ?scene? camera = ?camera? numberOfPaths = 0 excludeDirect = 0 ↔
    excludeCaustics = 1 width = 567 height = 567 ;
hdrWriteData file = "office-direct.hdr" buffer = ?vis? ↔
    msg = "Direct illumination in office scene lit by daylight" ;
colorFreeRGBBuffer object = ?vis? ;

photonmapGatherOptsInit = gopts filter = "cone" flags = ?? maxDist = 0.2 flattenFac = 0.1 ↔
    maxPhotons = 300 minPhotons = 1 allowZero = 1 useEstimates = 1 normDevCos = 0.94 ;
photonmapCreateFlags = indirectPhotons photonType = "indirect" lightNumber = -1 ;
photonmapGatherOptsInit = dgopts filter = "cone" flags = ?indirectPhotons? maxDist = 0.5 ↔
    flattenFac = 0.0 maxPhotons = 65 minPhotons = 10 allowZero = 0 useEstimates = 0 ↔
    normDevCos = 0.1 ;
photonmapRender = vis scene = ?scene? camera = ?camera? gatherOpts = ?gopts? ↔
    causticOpts = ?? dgatherOpts = ?dgopts? tdivs = 4 pdivs = 16 rlights = 0 dpaths = 3000 ↔
    spaths = 1 width = 567 height = 567 ;
hdrWriteData file = "office.hdr" buffer = ?vis? ↔
    msg = "Full global illumination in office scene lit by daylight" ;
colorFreeRGBBuffer object = ?vis? ;

```

Scena została wyrenderowana w rozdzielczości 567×567 pikseli. Światło dzienne zasymulowano poprzez ustawienie dużego prostokątnego źródła światła za oknem. Oświetlenie bezpośrednie utworzono przy pomocy mapy cienia zawierającej $1 \cdot 10^5$ fotonów cienia oraz 1000 promieni cienia w obszarach półcienia. Konieczność wykorzystania tak dużej mapy oraz ilości promieni wynikała z rozmiarów źródła światła, które powodowało powstawanie dużych półcieni. Oświetlenie pośrednie wygenerowano algorytmem dwufazowym. Wykorzystano mapę fotonów zawierającą $1 \cdot 10^5$ cząstek. Mapa ta została zbudowana z wykorzystaniem kontroli gęstości opartej na mapie istotności ($3 \cdot 10^4$ importonów). Oświetlenie pośrednie w poszczególnych punktach wyznaczano przy pomocy 5000 promieni wtórnych.

Czas tworzenia mapy cienia wyniósł 1.85s, a całkowity czas wyznaczania oświetlenia bezpośredniego to 26m 57.96s, z czego 45.1s zajęło wyszukiwanie obszarów półcienia. Mapa istotności została zbudowana w 22.8s, z czego 0.5s trwało emitowania importonów, a 22.3s wstępne przeliczanie istotności. Tworzenie mapy fotonów zajęło 69.25s. W czasie renderingu wykorzystano wyznaczenie oszacowań irradiancji w trybie na żądanie. Czas generowania oświetlenia pośredniego wynosił 2h 34m 11s, a całkowity czas renderingu 3h 5m 57s.

Bibliografia

- [1] James Arvo, David Kirk. Particle transport and image synthesis. *Computer Graphics*, 24(4):63–66, Sierpień 1990.
- [2] Jon Louis Bentley. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9):509–517, Wrzesień 1975.
- [3] Per Henrik Christensen. Faster photon map global illumination. *Journal of Graphics Tools*, 4(3):1–10, Kwiecień 2000.
- [4] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest. *Wprowadzenie do algorytmów*. Wydawnictwa Naukowo-Techniczne, Warszawa, wydanie 4, 2001.
- [5] Antoni Diller. *L^AT_EX wiersz po wierszu*. Wydawnictwo Helion, Gliwice, 2001.
- [6] Philip Dutré. Global illumination compendium, Wrzesień 2003.
- [7] James D. Foley, Andries van Dam, Steven K. Feiner, John F. Hughes, Richard L. Phillips. *Wprowadzenie do grafiki komputerowej*. Wydawnictwa Naukowo-Techniczne, Warszawa, wydanie 2, 2001.
- [8] Vlastimil Havran. *Heuristic Ray Shooting Algorithms*. Praca doktorska, Faculty of Electrical Engineering, Czech Technical University, Prague, Maj 2001.
- [9] Paul S. Heckbert. Adaptive radiosity textures for bidirectional ray tracing. *Computer Graphics*, 24(4):145–154, Sierpień 1990.
- [10] Heinrich Hey, Werner Purgathofer. Global illumination with photon map compensation. Raport instytutowy, Styczeń 2001.
- [11] Peter Ingmar, Georg Pietrek. Importance driven construction of photon maps. *Rendering Techniques '98 (Proceedings of the Ninth Eurographics Workshop on Rendering)*, strony 269–280. Springer Verlag, 1998.
- [12] Henrik Wann Jensen. Importance driven path tracing using the photon map. *Rendering Techniques '95 (Proceedings of the Sixth Eurographics Workshop on Rendering)*, strony 326–335. Springer Verlag, 1995.

- [13] Henrik Wann Jensen. Global illumination using photon maps. *Rendering Techniques '96 (Proceedings of the Seventh Eurographics Workshop on Rendering)*, strony 21–30. Springer Verlag, 1996.
- [14] Henrik Wann Jensen. Rendering caustics on non-lambertian surfaces. *Proceedings of Graphics Interface '96*, strony 116–121, Maj 1996.
- [15] Henrik Wann Jensen. *Realistic Image Synthesis Using Photon Mapping*. AK Peters, Natick, Massachusetts, 2001.
- [16] Henrik Wann Jensen, Niels Jørgen Christensen. Efficiently rendering shadows using the photon map. *Proceedings of Compugraphics'95*, strony 285–291. Alvor, Grudzień 1995.
- [17] Henrik Wann Jensen, Niels Jørgen Christensen. Optimizing path tracing using noise reduction filters. *Proceedings of WSCG 95*, strony 134–142. Plzen, 1995.
- [18] Henrik Wann Jensen, Niels Jørgen Christensen. Photon maps in bidirectional monte carlo ray tracing of complex objects. *Computers & Graphics*, 19(2):215–224, 1995.
- [19] Henrik Wann Jensen, Per Henrik Christensen. Efficient simulation of light transport in scenes with participating media using photon maps. *Proceedings of SIGGRAPH '98, Computer Graphics Proceedings, Annual Conference Series*, strony 311–320. Addison Wesley, 1998.
- [20] Henrik Wann Jensen, Per Henrik Christensen, Frank Suykens. A practical guide to global illumination using photon mapping. Sierpień 2001. Siggraph 2001 Course 38.
- [21] Henrik Wann Jensen, Julie Dorsey, Alan Edelman, Justin Legakis, Hans Køhling Pedersen. Modeling and rendering of weathered stone. *Proceedings of SIGGRAPH '99*, strony 225–234. Siggraph, 1999.
- [22] Henrik Wann Jensen, Justin Legakis, Julie Dorsey. Rendering of wet materials. *Rendering Techniques '99*. Springer Verlag, 1999.
- [23] Henrik Wann Jensen, Steve Marschner, Marc Levoy, Pat Hanrahan. A practical model for subsurface light transport. *Proceedings of SIGGRAPH 2001*. Siggraph, 2001.
- [24] James T. Kajiya. The rendering equation. *Computer Graphics*, 20(4):143–150, Sierpień 1986.
- [25] Alexander Keller, Ingo Wald. Efficient importance sampling techniques for the photon map. *Proceedings of the 18th spring conference on Computer Graphics*, strony 107–114, 2002.

- [26] Brian W. Kernigham, Dennis M. Ritchie. *Język ANSI C*. Klasyka informatyki. Wydawnictwa Naukowo-Techniczne, Warszawa, wydanie 7, 2002.
- [27] Eric P. Lafortune, Yves D. Willems. Bidirectional path tracing. *Compugraphics '93*, strony 95–104, 1993.
- [28] Peter Shirley, Changyaw Wang. Direct lighting by monte carlo integration. *Proceedings of the Second Eurographics Workshop on Rendering*, strony 85–98. Eurographics, 1991.
- [29] Peter Shirley, Changyaw Wang, Kurt Zimmerman. Monte carlo techniques for direct lighting calculations. *ACM Transactions on Graphics*, 15(1):1–36, Styczeń 1996.
- [30] Frank Suykens, Yves Willems. Density control for photon maps. *Rendering Techniques 2000 (Proceedings of the Eleventh Eurographics Workshop on Rendering)*, strony 11–22. Springer Verlag, 2000.
- [31] Eric Veach. *Robust Monte Carlo Methods For Light Transport Simulation*. Praca doktorska, Stanford University, Grudzień 1997.
- [32] Eric Veach, Leonidas J. Guibas. Metropolis light transport. *Proceedings of SIGGRAPH '97*, strony 65–76, Sierpień 1997.
- [33] Ingo Wald. *Realtime Ray Tracing and Interactive Global Illumination*. Praca doktorska, Saarland University, Saarbrücken, Germany, Styczeń 2004.
- [34] Gregory J. Ward. Real pixels. James Arvo, redaktor, *Graphics Gems II*, strony 80–83. San Diego: Academic Press, 1991.
- [35] Gregory J. Ward. Measuring and modeling anisotropic reflection. *Computer Graphics*, 26(2):265–272, Lipiec 1992.
- [36] Gregory J. Ward. The materials and geometry format. <http://radsite.lbl.gov/mgf/>, Styczeń 1996. Version 1.1.
- [37] Gregory J. Ward, Paul Heckbert. Adaptive shadow testing for ray tracing. *Proceedings of the Second Eurographics Workshop on Rendering*. Eurographics, 1991.
- [38] Gregory J. Ward, Paul Heckbert. Irradiance gradients. *Third Eurographics Workshop on Rendering*, strony 85–98. Eurographics, 1992.
- [39] Turner Whitted. An improved illumination model for shaded display. *Communications of the ACM*, 32(6):343–349, Czerwiec 1980.