



**Universidad**  
Zaragoza

# Proyecto Fin de Carrera

Transporte de Luz Transitorio  
en Medios Participativos

Julio Marco Murria

Director: Adrián Jarabo Torrijos  
Ponente: Dr. Diego Gutiérrez Pérez

Escuela de Ingeniería y Arquitectura  
2013



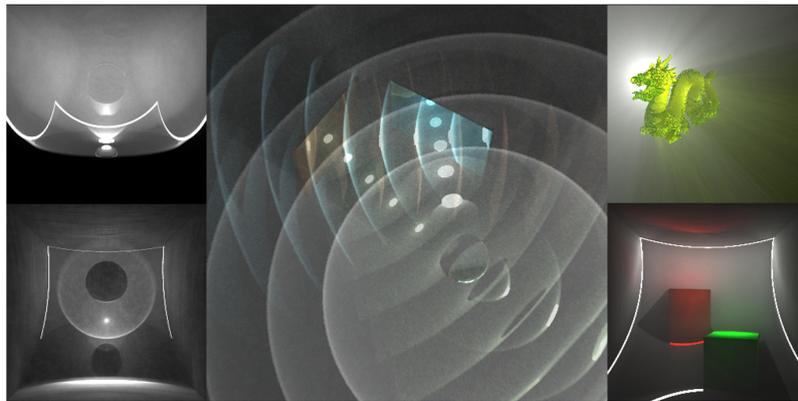


Proyecto Fin de Carrera  
Ingeniero en Informática

---

# Transporte de Luz Transitorio en Medios Participativos

---



JULIO MARCO MURRIA

*Director:*  
ADRIÁN JARABO

*Ponente:*  
DIEGO GUTIÉRREZ

Escuela de Ingeniería y Arquitectura  
Universidad de Zaragoza  
Noviembre de 2013



# Resumen

La simulación del transporte de luz en una escena mediante un ordenador es conocido como *renderizado*. Simular las interacciones físicas de la luz con la materia es una de las tareas más desafiantes en la informática gráfica. Medios participativos cotidianos como la niebla o la piel, interaccionan con la luz produciendo efectos muy interesantes. Para renderizar el transporte de luz en una escena según las propiedades de la materia, es necesario basarse en los modelos físicos que las definen, conllevando el uso de costosos algoritmos que los aproximan. El estudio del transporte de luz es un elemento clave a la hora de mejorar los algoritmos existentes.

Los recientes avances en captura ultrarrápida de imágenes nos permiten observar luz en movimiento a escalas macroscópicas, permitiéndonos ver cómo se propaga e interacciona con la materia y dando lugar a numerosas aplicaciones. Esto demuestra que existe información de utilidad en el dominio temporal que hasta ahora no se había tenido en cuenta. En el campo del render, esta información temporal normalmente es descartada, asumiendo que la luz tiene velocidad infinita, y produciendo imágenes en las que la luz está propagada por toda la escena. Romper esta asunción y considerar la velocidad de la luz finita implica la definición de un modelo de transporte de luz **transitorio**, es decir, teniendo en cuenta el tiempo de propagación de la luz.

Este proyecto se centra en renderizar el transporte de luz transitorio tanto en el vacío como en medios participativos, incluyendo el tiempo en los modelos tradicionales de transporte de luz. Para ello se ha implementado uno de los últimos avances en render de medios participativos, el algoritmo *Photon Beams*, y se ha modificado para incluir el tiempo de propagación de la luz considerando los efectos que tiene la inclusión del tiempo en las propiedades de los medios.

Analizar el transporte de luz transitorio aumenta la complejidad del render, dado que estamos añadiendo una dimensión más al problema. Reducir la dimensionalidad de un problema es una estrategia apropiada para analizarlo, ya que simplifica los modelos matemáticos, reduce los costes de cálculo y permite representar la información de manera más intuitiva. Aplicado al transporte de luz, reducir la dimensionalidad espacial a 2D requiere reconsiderar su formulación. En este proyecto redefinimos las propiedades de los medios participativos para el transporte de luz en 2D y realizamos una adaptación 2D del algoritmo *Photon Beams*. Basados en este enfoque, analizamos el transporte de luz en el tiempo variando las características de los medios y observando los perfiles temporales de la propagación de la luz dentro de medios participativos. Estos análisis nos han permitido detectar limitaciones sobre algoritmos ya existentes, así como observar las desventajas de las técnicas de muestreo utilizadas en los algoritmos estacionario cuando intentamos aplicarlas al transporte de luz transitorio. Finalmente hemos podido comprobar cómo los distintos sistemas de representación y las propiedades de los medios afectan a cómo interpretamos el transporte de luz en una escena.

Este trabajo se ha realizado dentro de un proyecto de investigación en el Graphics and Imaging Lab y en colaboración con Wojciech Jarosz, investigador jefe del grupo de *rendering* en Disney Research, Zürich. Los resultados de este proyecto serán sometidos a la conferencia internacional SIGGRAPH 2014. Adicionalmente resultados previos de este trabajo se presentaron en la Congreso Español de Informática Gráfica 2013.



*A la memoria de mi padre.*



## *Agradecimientos*

Quiero dar las gracias a Adrián por su ayuda y dedicación, y por enseñarme cosas nuevas cada día. A él y a Diego por confiar en mí y darme la oportunidad de participar en este proyecto tan importante. A Wojciech por resolver todas las dudas que le planteé hasta el detalle más mínimo. También a toda la gente del laboratorio por generar tan buen rollo y por sus buenos consejos, especialmente a Adolfo y José por interminables y fructíferas discusiones sobre este trabajo. Y por último, y lo más importante, a mi madre y a mi hermana, por aguantarme en los momentos difíciles todos estos años. Gracias.



# Índice general

<b>1. Introducción</b>	<b>1</b>
<b>2. Trabajo Relacionado</b>	<b>5</b>
<b>3. Teoría de Transporte de Luz</b>	<b>7</b>
3.1. La Ecuación de Render . . . . .	8
3.2. Transferencia Radiativa . . . . .	9
3.3. Algoritmos de Render . . . . .	12
<b>4. Transporte de Luz en Estado Transitorio</b>	<b>15</b>
4.1. Interacción de luz con superficies . . . . .	16
4.2. Interacción de luz con medios participativos . . . . .	17
<b>5. Transporte de Luz en 2D</b>	<b>19</b>
5.1. Geometría . . . . .	20
5.2. Radiometría en superficies . . . . .	21
5.3. Radiometría en medios participativos . . . . .	21
5.3.1. Funciones de fase . . . . .	22
5.3.2. Coeficientes de dispersión y absorción . . . . .	22
<b>6. <i>Photon Beams</i></b>	<b>25</b>
6.1. Trazado de fotones . . . . .	26
6.2. Estimación de radiancia utilizando <i>Photon Mapping</i> . . . . .	27
6.3. Estimación de radiancia usando <i>Photon Beams</i> . . . . .	28
6.3.1. Estimación de radiancia de un punto . . . . .	28
6.3.2. Estimación de radiancia de un rayo . . . . .	30
6.4. <i>Photon Beams</i> en 2D . . . . .	31
6.5. Implementación del algoritmo . . . . .	32
<b>7. Photon Beams Transitorios</b>	<b>35</b>
<b>8. Resultados</b>	<b>39</b>
8.1. Perfiles temporales de radiancia en una escena 2D en free-space . . . . .	39
8.2. Proyección ortográfica de una luz puntual dentro de un medio participativo . . . . .	40
8.3. Renderizado de medios participativos en 2D con un sensor de irradiancia . . . . .	42
8.4. Perfiles temporales gaussianos variando las funciones de fase . . . . .	43
8.5. Problemas de muestreo en renderizado transitorio . . . . .	44
8.6. Efectos de <i>camera warping</i> en 3D . . . . .	45

---

8.7. Emisión delta vs. emisión continua . . . . .	46
<b>9. Conclusiones</b>	<b>49</b>
9.1. Trabajo futuro . . . . .	49
9.2. Desarrollo de proyecto . . . . .	50
9.3. Conclusión personal . . . . .	50
<b>Bibliografía</b>	<b>53</b>
<b>A. Contenido del CD</b>	<b>57</b>
<b>B. Póster en el CEIG 2013</b>	<b>59</b>
<b>C. Detalles de Implementación</b>	<b>63</b>
<b>D. Resultados Adicionales</b>	<b>65</b>

# Símbolos y Terminología

$x$	Punto en el espacio
$\vec{\omega}$	Vector dirección
$\vec{n}$	Vector normal en una superficie o curva
$s, ds$	Distancia, paso diferencial
$t$	Tiempo
$\rho$	Función de fase
$f_r$	Bidirectional Reflectance Distribution Function
$\Phi_p, \Phi_b$	Energía de un fotón o un <i>beam</i>
$L$	Radiancia
$L_i$	Radiancia por dispersión entrante dentro de un medio
$L_r$	Radiancia reflejada por una superficie
$L_e$	Radiancia emitida por una superficie o medio
$T_r$	Transmitancia
$\sigma_s, \sigma_a, \sigma_t$	Coefficientes de dispersión, absorción y extinción <sup>1</sup>
$\kappa_s, \kappa_a$	Secciones eficaces de dispersión y absorción <sup>1</sup>
$C$	Densidad de partículas
$\eta$	Índice de refracción

Debido a que este proyecto forma parte de un proyecto de investigación, la mayor parte de la literatura y terminología relacionada procede del inglés. Por esta razón, muchos de los términos y expresiones técnicas más comunes carecen de traducción alguna al castellano o su traducción no es habitual en este contexto. Durante la escritura de este documento se ha tratado en la medida de lo posible de traducir la mayor parte de estos términos y expresiones, exceptuando casos en los que no existe traducción o su inclusión repetida en el texto afectaba a la fluidez de la lectura.

---

<sup>1</sup>Los subíndices  $s$  y  $t$  vienen de *scattering* (dispersión) y extinción y no tienen relación alguna con distancias  $s$ , pasos diferenciales  $ds$ , o tiempo  $t$ .



# Capítulo 1

## Introducción

*Rendering* es el proceso de sintetizar el modelo de una escena en una imagen en el contexto de los gráficos por ordenador. En particular, el renderizado basado en física es el proceso de simular del transporte de luz en una escena basado en las propiedades físicas de la materia y la luz. Una de las principales asunciones en render, es que la velocidad de la luz es infinita, dando como resultado imágenes en las que la luz está propagada en todos los puntos de la escena. Esta asunción es razonable siempre que queramos representar una escena como si fuera vista por el ojo humano o una cámara tradicional. Sin embargo, la recientemente introducida *Femto-Fotografía* [VWJ<sup>+</sup>12, VWJ<sup>+</sup>13] permite capturar luz en escenas macroscópicas y ver cómo se propaga a frecuencias de un billón de fotogramas por segundo. Capturar luz en movimiento nos permite ver cómo se propaga e interacciona con la materia. Por ejemplo, en la Figura 1.1 se puede ver una botella llena de agua con un poco de leche, contra la que se lanza un pulso de luz, mostrando la evolución temporal de la luz interactuando con el medio. Este tipo de capturas han provocado la aparición de nuevos campos de investigación con numerosas aplicaciones como la captura de geometría oculta [VWG<sup>+</sup>12], la captura de funciones de reflectancia en materiales [NZV<sup>+</sup>11], o el diseño de nuevos sistemas de captura de imagen sin lente [WWB<sup>+</sup>12]. Como se puede ver en la secuencia de la botella, la luz interactúa con la materia a su paso a través del espacio, produciendo efectos muy complejos que dependen de las propiedades físicas de la materia. La materia puede alterar los caminos y propiedades de la luz de formas muy distintas, por ejemplo reduciendo su intensidad (absorción), redirigiéndola a otros lugares (dispersión) o incluso separándola espacialmente en componentes de diferentes longitudes de onda (dispersión cromática). Entre los efectos más interesantes entre luz y materia, los **medios participativos** cubren un amplio rango de fenómenos comunes en el mundo real. Un medio se considera como participativo cuando los efectos de absorción y dispersión dentro de ese medio son significantes, incluyendo medios como la niebla o el agua, y materiales como la piel o la cera (Figura 1.2).



Figura 1.1: Un pulso de luz a través de una botella capturado a un billón de frames por segundo [VWJ<sup>+</sup>12, VWJ<sup>+</sup>13].



(a) Dispersión a través de la niebla.

(b) Dispersión bajo la superficie (*subsurface scattering*) en la cera de la vela, y emisión en la llama.

Figura 1.2: Ejemplos de medios participativos en el mundo real.

Relajar la asunción inicial de que la velocidad de la luz es infinita y considerar los tiempos de propagación de la luz hace que nos movamos del clásico transporte de luz en *estado estacionario* al transporte de luz en *estado transitorio*. Considerar el transporte de luz transitorio en el renderizado conlleva tener en cuenta el tiempo que le cuesta a la luz propagarse a través de la escena, normalmente descartado por los algoritmos tradicionales de render. En este proyecto nos centramos en **renderizar transporte de luz transitorio en medios participativos**. Simular de manera precisa el transporte de luz en este tipo de medios supone un reto, ya que las interacciones de luz ocurren en puntos diferenciales del espacio. Esto conlleva el uso de un gran número de algoritmos numéricos que simulan de manera aproximada los modelos físicos que definen estas interacciones, normalmente con altos costes computacionales. En este proyecto se comparan las ventajas e inconvenientes al incluir el dominio temporal en los algoritmos estacionarios, y se discute cómo la propagación de la luz en el dominio temporal muestra limitaciones en algunos de los métodos existentes. Para ello, se ha implementado el algoritmo *Photon Beams* [JNSJ11], uno de los últimos avances en investigación en el renderizado de medios participativos, y se ha adaptado para incluir el transporte de luz en estado transitorio.

Renderizar transporte de luz transitorio añade una dimensión adicional al problema. Esto incrementa el nivel de complejidad durante el proceso del render, y requiere una representación de los resultados mediante vídeo en vez de las típicas imágenes estáticas del render estacionario. Esto hace que el análisis sea muy complicado si consideramos un mundo 3D, y movernos a 2D supone un enfoque mucho más intuitivo y adecuado para analizar el transporte de luz transitorio en render. Primero, simplifica los modelos matemáticos y las derivaciones, y consecuentemente reduce los costes de cálculo al renderizar. Segundo, permite utilizar sistemas de representación mucho más intuitivos, proporcionando entendimiento práctico igualmente aplicable a 3D. Este enfoque requiere que consideremos el transporte de luz en 2D, necesitando redefinir las propiedades físicas ya existentes en 3D para un mundo 2D. Jarosz et. al [JSKJ12] proponen un modelo radiométrico en 2D para analizar el transporte de luz en el vacío (sin medios participativos). Dado que este proyecto también incluye el análisis del transporte de luz en medios participativos, contribuimos al model de Jarosz et. al con consideraciones y definiciones adicionales sobre radiometría para medios participativos en 2D, y las usamos para implementar una adaptación del algoritmo *Photon Beams* en 2D. Considerar el transporte de luz en estado transitorio abre nuevas formas de analizar las interacciones de la luz con la materia. Dado que todo el proceso de render se realiza normalmente utilizando un **modelo directo**, nos permite modificar las

propiedades de la materia en la escena para obtener datos valiosos en el dominio temporal **de utilidad para resolver problemas inversos**. Esto nos permite obtener información adicional bajo un **entorno controlado** que nos ayuda a entender el transporte de luz y sus interacciones con la materia. Además, incluir el tiempo de propagación de la luz en los algoritmos comunes de render estacionario permite observar su comportamiento en el dominio temporal, y puede ayudar a mejorar las técnicas actuales de render.

Resumiendo, las **principales contribuciones** de este proyecto son:

- Formulación del transporte de luz transitorio para medios participativos, discutiendo la inclusión del tiempo en el modelo físico.
- Definición del transporte de luz en 2D para medios participativos, considerando cambios de dimensionalidad en sus propiedades.
- Implementación del algoritmo *Photon Beams*, y su derivación y adaptación a 2D.
- Derivación y adaptación de una versión transitoria del algoritmo *Photon Beams*, incluyendo el tiempo de propagación de la luz y discutiendo los efectos de esta inclusión en la precisión de los resultados.

Este proyecto está enmarcado dentro de un proyecto de investigación del Graphics and Imaging Lab, y realizado en colaboración con Wojciech Jarosz, investigador jefe en el grupo de *rendering* de Disney Research, Zürich. Parte de los resultados de este proyecto serán incluidos en el artículo del proyecto de investigación al que va ligado, y que será sometido a la conferencia mundial de gráficos SIGGRAPH en enero de 2014. De manera adicional, resultados previos y discusiones de este proyecto fueron incluidos en un póster presentado en el Congreso Español de Informática Gráfica (CEIG) en septiembre de 2013 (ver Anexo B).



## Capítulo 2

# Trabajo Relacionado

Mientras que el transporte de luz estacionario ha sido investigado ampliamente en los gráficos por ordenador tanto para transporte en el vacío [DBB06, KGKCh13] como en medios participativos [GJJD09], su generalización a la propagación de la luz en estado transitorio no ha recibido mucha atención. A pesar de que en general el tiempo de propagación se incluye en la base teórica del transporte de luz [Arv93], la mayoría de los trabajos cambian rápidamente al transporte estacionario, y no se ha propuesto ni discutido ninguna aplicación de render transitorio, asumiendo pues que la velocidad de la luz es infinita.

El término *Renderizado Transitorio* fue acuñado por primera vez por Smith et al. [SSD08]. Presentaron una generalización de la *Ecuación de Render* [Kaj86] modelada como un operador recursivo incluyendo la propagación de la luz a velocidad finita. El modelo presentado forma una base teórica sólida para un modelo directo para ser usado en aplicaciones de visión por computador basadas en los tiempos de propagación de la luz, pero no provee un entorno práctico para render. Jarabo [Jar12] presenta un entorno práctico para analizar transporte de luz en estado transitorio, simulando los dispositivos de captura ultra-rápida de imágenes usados en [VWJ<sup>+</sup>12, VWJ<sup>+</sup>13], proporcionando un sistema funcional para desarrollar algoritmos transitorios de render, utilizado en el desarrollo de este proyecto.

Heide [Hei12] desarrolló una versión transitoria del algoritmo *Radiosity* [GTGB84] en GPU, usándolo como un modelo directo para recuperar geometría oculta, modelado como un problema inverso de render. Este modelo funciona bien para el problema específico de reconstrucción geométrica, pero está limitado a superficies difusas y solo tiene en cuenta interreflexiones entre superficies. Con el mismo objetivo, Keller et al. [KOKP07, KK09] desarrollaron un simulador de un sensor *time-of-flight* acelerado por hardware, que permite generar imágenes con resolución temporal, incluyendo diferentes artefactos que pudieran aparecer en los sensores *time-of-flight*. Este trabajo, sin embargo, solo permite simular iluminación directa, y no puede generar animaciones generales con resolución temporal. Todos estos trabajos se centran en desarrollar modelos directos de transporte de luz para **problemas específicos**; por tanto ninguno se centra en el análisis de transporte de luz transitorio y no exploran como se comportan las diferentes estrategias de render cuando se incluye el dominio temporal. Además, están centrados en simulaciones para transporte de luz entre superficies, sin ningún tipo de medio participativo entre ellas.

D'Eon e Irving [DI11] derivan su modelo estacionario de *Quantized Diffusion* para *subsurface scattering*, basándose en el extenso trabajo previo existente en difusión resuelta en tiempo. Sin embargo, este trabajo está basado en explotar información del transporte de luz en el dominio temporal para aplicarlo únicamente a soluciones en estado estacionario, y considerando solo

medios participativos con alta densidad óptica. Nosotros analizamos el transporte de luz en estado transitorio sin restricciones en las propiedades del medio y considerando su investigación tanto para soluciones estacionarias como transitorias.

Comenzar por una simplificación matemática en 2D y extender seguidamente las derivaciones a 3D ha demostrado ser una estrategia apropiada para analizar el transporte de luz en el dominio primal [LALD12], gradiente [RMB07], Hessiano [JSKJ12] y en el de frecuencias [DHS<sup>+</sup>05, MWRD13, BSS<sup>+</sup>13], proporcionando un mejor entendimiento del transporte de luz y mejorando significativamente la eficiencia en render. En este contexto, Jarosz et al. [JSKJ12] derivaron una teoría autocontenida y consistente del transporte de luz en 2D a través del vacío que cumple los principios de conservación energética. Demuestran que este enfoque ayuda a visualizar intuitivamente el transporte de luz, reduce la complejidad matemática y proporciona un entorno que permite experimentar de forma fácil y eficiente usando cálculos de transporte de luz en 2D. Todos los trabajos mencionados asumen velocidad de la luz finita, ignorando el dominio temporal en si propagación, y únicamente considerando dispersión entre superficies, asumiendo que la luz se propaga en el vacío, por lo que nosotros proponemos una extensión que incluye propagación de la luz en el dominio temporal e incluyendo medios participativos.

## Capítulo 3

# Teoría de Transporte de Luz

El render basado en física simula el transporte de luz en una escena del mismo modo que sucede en el mundo real, y lo sintetiza en una imagen. Un haz de luz alcanzando una región del espacio distinta del vacío interactuará con moléculas de materia que pueden alterar su comportamiento de diferentes maneras (Figura 3.1). Cualquier medio (entendiendo *medio* como materia en cualquier estado sólido, líquido o gaseoso) interactúa con la luz cuando pasa a través de él. Dado que la luz es en realidad radiación electromagnética, la transferencia de energía al propagarse la luz en un medio que absorbe, emite o dispersa luz está modelada por la *Teoría de Transferencia Radiativa (Radiance Transfer Theory)* [Cha60] (Sección 3.2). Sin embargo, en los gráficos por ordenador, normalmente necesitamos hacer simplificaciones sobre los modelos existentes—que son a veces simplificaciones en sí mismos— con el objetivo de obtener soluciones prácticas. La importancia de los efectos de absorción y dispersión depende de la naturaleza de la materia. Algunos medios como los metales o la piedra evitan que la mayor parte de la luz penetre en ellos, así que los efectos de dispersión suelen ser insignificantes y pueden ser ignorados. Estos casos especiales están presentes en el mundo real, y pueden ser representados con modelos simplificados que todavía proporcionen resultados con precisión física. Se les conoce como *medios no participativos* y se asume que el transporte de luz ocurre únicamente en la superficie, en contraste con los *medios participativos* en los que el transporte de luz ocurre en todo su volumen. El transporte de luz en medios no participativos está modelado por la *Ecuación de Render* (Sección 3.1) y en medios participativos por la *Ecuación de Transferencia*

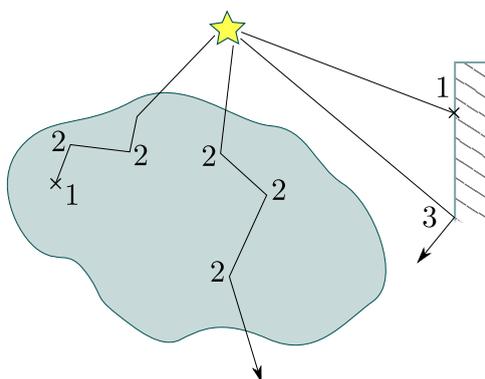


Figura 3.1: Materia interactuando con la luz. La luz viaja libremente a través del vacío hasta que entra en una región con materia, y podrá ser absorbida (1), dispersada (2) o reflejada (3).

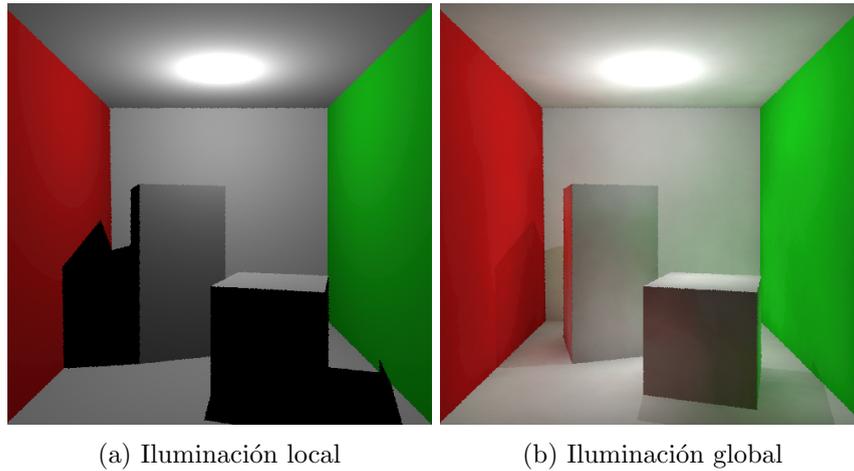


Figura 3.2: La iluminación local solo tiene en cuenta luz directa, mientras que la iluminación global también calcula luz indirecta proveniente de otros puntos en la escena.

*Radiativa* (RTE, Sección 3.2). De aquí en adelante, por cuestiones de brevedad y a menos que se indique lo contrario, cada mención en este documento del término “medio” se refiere a “medio participativo”.

La interacción de luz directa incidente en un punto de una superficie o un medio está definido por las propiedades locales de esa superficie o medio en ese punto, lo que en gráficos por computador se denomina *iluminación local* (Figura 3.2a). Sin embargo, esta luz incidente puede ser absorbida, dispersada y/o transmitida por la superficie o medio, alcanzando otros lugares en la escena. Tener en cuenta este comportamiento al renderizar una escena conduce a modelos de *iluminación global* (Figure 3.2b). Para representar físicamente una escena se requiere un modelo completo de iluminación global. Esto conlleva un proceso infinitamente recursivo en el que superficies y medios dispersan luz sobre otras superficies y medios, repitiéndose el proceso indefinidamente.

### 3.1. La Ecuación de Render

Al proceso de calcular iluminación en un punto se le conoce como *shading* y depende en gran medida en si estamos utilizando un modelo de iluminación global o local. Para calcular la iluminación global en un **medio no participativo**, podemos asumir que el transporte de luz ocurre únicamente en los límites de la materia, y puede ser modelado integrando la luz proveniente de cualquier dirección alrededor de un punto en una superficie. Este problema fue formulado por Kajiyá [Kaj86] como una única integral, la *Ecuación de Render* (Ecuación 3.1), que modela el transporte de luz entre superficies de objetos en términos de radiancia saliente  $L_r(x, \vec{\omega}_o)$  en un punto  $x$  con dirección  $\vec{\omega}$ :

$$L_r(x, \vec{\omega}_o) = L_e(x, \vec{\omega}_o) + \int_{\Omega} f_r(x, \vec{\omega}_i, \vec{\omega}_o) L(x, \vec{\omega}_i) (\vec{n} \cdot \vec{\omega}_i) d\vec{\omega}_i \quad (3.1)$$

$L_e$  indica luz que es emitida por la superficie (p. ej. incandescencia). El segundo término integra la radiancia  $L(x, \vec{\omega}_i)$  en todas las direcciones entrantes  $\vec{\omega}_i$  sobre el hemisferio alrededor del punto  $x$ . El término  $f_r$  está modelado por lo que se denomina *Bidirectional Reflectance Distribution Function* (*BRDF*), e indica cuánta de la luz incidente en  $x$  con dirección  $\vec{\omega}_i$  es reflejada hacia  $\vec{\omega}_o$ .

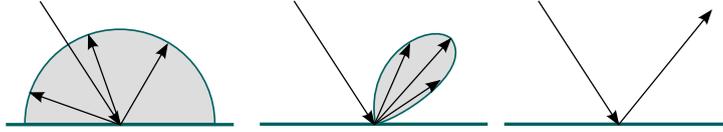


Figura 3.3: Diferentes modelos de  $BRDFs$  mostrando reflexión Lambertiana (izquierda), una reflexión muy direccional típica de superficies brillantes (centro), y reflexión perfectamente especular (derecha).

Las  $BRDFs$  están por definición normalizadas (i.e.  $\int_{\Omega} f_r(x, \vec{\omega}_i, \vec{\omega}_o) d\vec{\omega}_i \leq 1$ ) y tendrán diferente comportamiento dependiendo de las propiedades de la superficie. Ejemplos de diferentes  $BRDFs$  se muestran en la Figura 3.3. El producto escalar del último término dentro de la integral tiene en cuenta un factor geométrico entre la dirección de la luz incidente y la normal de la superficie  $\vec{n}$  en  $x$ . Resolver analíticamente esta ecuación es imposible dado que cada punto de la escena está rodeado de geometría arbitraria. En la última sección del capítulo se explican las bases de los algoritmos más comunes que aproximan esta ecuación.

## 3.2. Transferencia Radiativa

La *Ecuación de Render* original solo tiene en cuenta interreflexiones entre superficies, pero como se ha mencionado antes, para obtener un render físicamente preciso de una escena, también tenemos que tener en cuenta cualquier medio participativo existente. Una región de la escena es considerada como medio participativo si los efectos de absorción y dispersión de la luz dentro de esa región son significativos (Figura 1.2b). La *absorción* modela el efecto de pérdida de energía durante el paso de la luz a través de un medio. La *dispersión* modela el efecto de desviación de la trayectoria de la luz.

Podemos imaginar un medio como un conjunto infinito de pequeñas partículas que al paso de la luz interactúan con ella y pueden absorber su energía o cambiar su dirección de propagación. Los efectos de la luz atravesando un medio son modelados por la *Ecuación de Transferencia Radiativa* [Cha60], que describe el cambio diferencial de energía sufrido por la luz al viajar un paso diferencial en su dirección de propagación. Este cambio incluye tanto pérdida como incremento de energía. Un medio en sí mismo puede emitir luz (Figura 3.5b, p. ej. una vela), produciendo un incremento de energía  $L_e$ . Por otro lado, la absorción y la dispersión producen pérdida de energía en la dirección de propagación. La absorción atenúa la luz (Figura 3.5a) convirtiéndola en otros tipos de energía. La dispersión desvía la luz en otras direcciones, lo que en consecuencia produce una pérdida de energía en la dirección en la que la luz viaja, un efecto propiamente denominado dispersión saliente (Figura 3.5c).

La dispersión y absorción de un medio son representados respectivamente por los coeficientes de dispersión  $\sigma_s(x)$  y absorción  $\sigma_a(x)$  en cada punto  $x$  del medio. Estos coeficientes dependen del medio, y ambos están modelados como cambios proporcionales a la radiancia incidente  $L$  por paso diferencial. La naturaleza de estos coeficientes proviene de dos factores: las secciones eficaces de dispersión/absorción ( $\kappa_s$  y  $\kappa_a$ ) y la concentración de partículas  $C$ , siendo  $\sigma_s = C\kappa_s$  (igual para la absorción). En física, una sección eficaz cuantifica la probabilidad de que ocurra cierta interacción partícula-partícula, y puede ser imaginada como la proyección de el volumen 3D de una partícula sobre un paso diferencial. Así pues, la sección eficaz corresponde a un área (unidades  $m^2$ , ver Figura 3.4a), y en el caso de transporte de luz en un medio, esta interacción puede entenderse como luz interactuando con una partícula en el medio. La concentración de

partículas  $C$  es medida como densidad de partículas por espacio tridimensional, i.e. por volumen 3D ( $\text{m}^{-3}$ ). En consecuencia el coeficiente de dispersión (o absorción) tendrá unidades  $\text{m}^{-1}$ . Dado que ambos coeficientes son modelados de manera similar con iguales unidades físicas, se pueden unir en lo que se denomina *coeficiente de extinción*  $\sigma_t(x) = \sigma_a(x) + \sigma_s(x)$ .

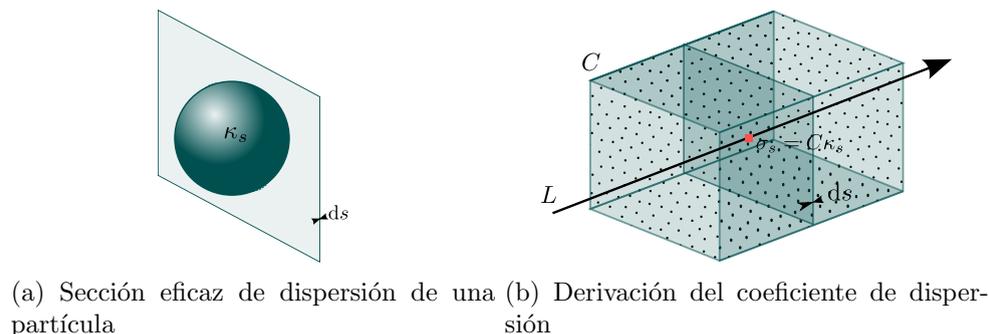


Figura 3.4: Derivación del coeficiente de dispersión  $\sigma_s$ . El coeficiente de absorción  $\sigma_a$  se deriva de forma análoga.

La luz absorbida (Figura 3.5a) se transforma en otros tipos de energía, por ejemplo calor, y se considera perdida. Pero, ¿a dónde va la luz dispersada? La luz dispersada es desviada hacia otras direcciones, y alcanza otros puntos del medio. Así que cuando definimos un cambio de energía en un punto  $x$  también tenemos que tener en cuenta la luz que puede ser dispersada desde otros puntos del medio y que llegue a  $x$ . Este fenómeno es conocido como dispersión entrante (Figura 3.5d) y produce un incremento de energía en la dirección en la que viaja la luz. La dispersión entrante puede provenir de cualquier dirección alrededor de  $x$ . Al igual que en el caso de superficies, donde la luz proveniente de direcciones hemisféricas debe ser recolectada, en un punto dentro de un medio toda la luz proveniente de dispersión entrante debe ser recolectada en todas las direcciones esféricas.

Pero, ¿cuánta de la luz incidente en un punto se dispersa en cada dirección? Cuando la luz en un medio llega a un punto, puede ser dispersada en muchas direcciones alrededor de ese punto. La cantidad de luz dispersada a una dirección arbitraria no es siempre uniforme, y está determinada por la denominada *función de fase*  $\rho(x, \vec{\omega}_i, \vec{\omega}_o)$ . Una función de fase define cuánta luz que entra en un punto  $x$  desde la dirección  $\vec{\omega}_i$  es dispersada en una dirección saliente  $\vec{\omega}_o$ , y depende de la naturaleza del medio. De este modo un medio puede presentar dispersión isotrópica, *backward* o *forward*. Los medios isotrópicos son un caso ideal inexistente en el mundo real, pero representan un caso de estudio de utilidad para observar su comportamiento en render. La luz dispersada en un medio con dispersión *backward* tenderá a tener direcciones salientes opuestas a la dirección entrante. Al contrario, si un medio presenta más dispersión *forward*, la luz dispersada tenderá a

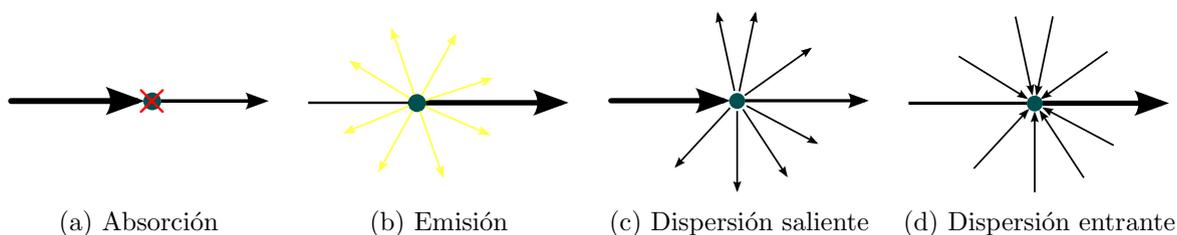


Figura 3.5: Efectos de un medio. La luz viajando a través de un medio puede interactuar produciendo cuatro tipos de efectos: absorción, emisión, dispersión entrante y dispersión saliente.

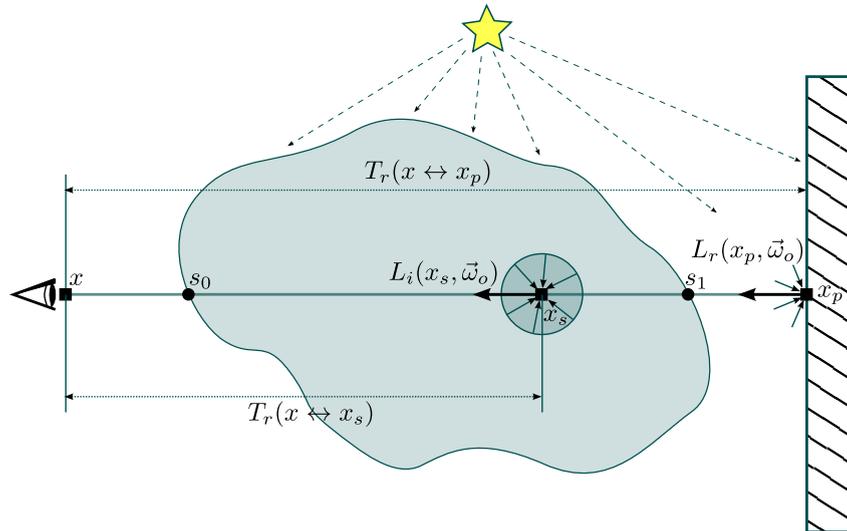


Figura 3.6: Integración de la luz acumulada por un rayo atravesando de un medio, descrita por la Ecuación 3.3.

seguir direcciones salientes similares a la entrante. Las funciones de fase están definidas como distribuciones de probabilidad y en consecuencia están normalizadas.

Volviendo a la integración de la dispersión entrante, dado un punto  $x$  en un medio y la luz  $L(x, \vec{\omega}_i)$  proveniente de  $\vec{\omega}_i$ , la parte proporcional de esa luz que es dispersada en ese punto está definida por  $\sigma_s(x)L(x, \vec{\omega}_i)$ . Además, si queremos saber cuánta de esa luz dispersada sale en la dirección  $\vec{\omega}_o$ , debemos multiplicarla por la función de fase  $\rho(x, \vec{\omega}_i, \vec{\omega}_o)$  del medio. Integrando todas las direcciones en la esfera alrededor de  $x$  tendremos,

$$L_i(x, \vec{\omega}_o) = \sigma_s(x) \int_{\Omega_{4\pi}} \rho(x, \vec{\omega}_i, \vec{\omega}_o) L(x, \vec{\omega}_i) d\vec{\omega}_i \quad (3.2)$$

representando la dispersión entrante en un punto  $x$  del medio. Podemos separar  $\sigma_s(x)$  fuera de la integral ya que solo depende de la posición. Nótese cómo Ecuación 3.2 se parece al término integral de la *Ecuación de Render* (Ecuación 3.1). De forma similar, el primero integra la luz en todas las direcciones esféricas alrededor de un punto de un medio, y el segundo integra todas las direcciones hemisféricas alrededor de un punto en una superficie. En realidad, existe una correspondencia directa entre la función de fase  $\rho(x, \vec{\omega}_i, \vec{\omega}_o)$  de un medio y la *BRDF*  $f_r(x, \vec{\omega}_i, \vec{\omega}_o)$  de una superficie.

Mientras que la RTE está definida en forma diferencial (ver [Cha60] para más información), cuando renderizamos una escena normalmente queremos saber la luz que observa un rayo intersectando una superficie en la escena (o alcanzando un límite de escena). Si hay algún medio entre el punto de intersección (o límite) y el punto de vista, hay que tener en cuenta toda la luz incidente en el rayo y dispersada hacia el punto de vista (Figura 3.6). Esto requiere resolver la RTE para un intervalo finito  $[s_0, s_1]$ . Asumiendo medios no emisivos por simplicidad, la forma integral de la RTE está definida por la luz proveniente de dispersión entrante (Ecuación 3.2), y la luz reflejada desde la superficie más cercana  $L(x_p, \vec{\omega}_o)$ . Por ello la forma integral de la RTE es:

$$L(x, \vec{\omega}_o) = T_r(x \leftrightarrow x_p) L_r(x_p, \vec{\omega}_o) + \int_{s_0}^{s_1} T_r(x \leftrightarrow x_s) L_i(x_s, \vec{\omega}_o) ds \quad (3.3)$$

donde  $L_i(x_s, \vec{\omega}_o)$  corresponde a la Ecuación 3.2, y  $T_r(x \leftrightarrow x')$  representa transmitancia, i.e. la cantidad de luz que no se ha extinguido entre  $x$  and  $x'$ . Nótese que del mismo modo que  $L_r(x_p, \vec{\omega}_o)$  (Ecuación 3.1),  $L_i$  y por consiguiente la integral de la dispersión entrante a lo largo de  $[s_0, s_1]$ , no están resueltas analíticamente. Esto es porque igual que pasa con un punto en una superficie, los elementos alrededor de un punto en el medio son arbitrarios, e integrar analíticamente la dispersión entrante a lo largo del intervalo de un rayo es prácticamente imposible.

**Homogeneidad y heterogeneidad.** La densidad y composición de un medio pueden variar a lo largo del espacio. Una nube de humo puede que no interactúe uniformemente con la luz en todo su volumen, y dependiendo de la densidad de las partículas de humo en cada punto, puede absorber o dispersar luz de manera diferente. De forma similar, la forma, tamaño y densidad de gotas de agua dentro de la niebla puede variar a lo largo del volumen, cambiando la función de fase y produciendo diferentes tipos de dispersión. Según la uniformidad de un medio podemos distinguir entre medios homogéneos y heterogéneos. Un medio se considera *homogéneo* cuando interactúa con la luz de manera uniforme en todo su volumen. Esto significa que la dependencia espacial (representada por  $x$ ) en la dispersión, absorción y función de fase no existe, por lo que pueden ser escritos respectivamente como  $\sigma_s$ ,  $\sigma_a$  and  $\rho(\vec{\omega}_i, \vec{\omega}_o)$ . Un medio es considerado *heterogéneo* cuando alguna de las propiedades que lo define tiene dependencia espacial, y por tanto la luz se comporta diferente según la posición en el medio. Nótese que en el caso de un medio puramente homogéneo, los coeficientes de absorción y dispersión son valores constantes. Esto puede suponer una importante diferencia al renderizar medios participativos, dado que la independencia espacial nos permite hacer algunos cálculos analíticamente. En concreto, el término de transmitancia  $T_r$  mostrado en Ecuación 3.3 se podría computar analíticamente como

$$T_r(x \leftrightarrow x') = e^{-\sigma_t \|x-x'\|} \quad (3.4)$$

### 3.3. Algoritmos de Render

Resolver analíticamente la *Ecuación de Render* y la RTE es prácticamente imposible debido a dos razones principales: geometría arbitraria y medios con propiedades complejas (tanto superficies como medios participativos). Solamente algunas configuraciones de escena bajo ciertas asunciones nos permiten calcular analíticamente la iluminación global en una escena, pero normalmente se necesitan soluciones numéricas para renderizar una escena con un ordenador. En general, soluciones que son a la vez físicamente precisas y visualmente aceptables requieren algoritmos muy lentos o que consumen mucha memoria. Las integrales en las Ecuaciones 3.1, 3.2 y 3.3 se resuelven normalmente mediante integración estocástica por Monte Carlo, teniendo

$$\int_{\Omega} f(x) dx \approx \frac{1}{N} \sum_{x_i \in \Omega} \frac{f(x_i)}{\text{pdf}(x_i)} \quad (3.5)$$

siendo  $\text{pdf}(x_i)$  la distribución de probabilidad del valor  $x_i \in \Omega$ . Por tanto, los resultados presentados pueden presentar efectos como varianza (visto como ruido en las imágenes resultantes) o bias (una diferencia entre la el valor esperado y la solución obtenida) que limitan la calidad y precisión de los algoritmos.

Entre las diversas estrategias para renderizar una imagen, **ray tracing** [Whi79] consiste en trazar rayos desde el punto de vista y a través de cada píxel de la imagen, calcular la iluminación local en la intersección geométrica más cercana y guardar el color calculado en el



(a) Cáusticas producidas por reflexión.

(b) Cáusticas producidas por refracción.

Figura 3.7: Imágenes de cáusticas generadas por Jensen [Jen01]

píxel. Estrictamente hablando, ray tracing no se refiere solo a trazar rayos desde el punto de vista a través de los píxeles, sino desde entre cualquier par de puntos en la escena, incluyendo las posiciones de las luces, superficies o puntos en los medios existentes. Por tanto, puede ser usado para saber si un intercambio de energía es posible entre esos puntos, y es la base de la mayoría de los algoritmos de iluminación basados en Monte Carlo, incluyendo aquellos que resuelven ecuaciones como 3.1 o 3.3.

A la hora de computar la luz en una escena podemos distinguir entre dos enfoques: algoritmos que trazan rayos desde el punto de vista, y algoritmos que trazan rayos desde las fuentes de luz. Los primeros, consisten en trazar caminos desde el punto de vista tratando de encontrar las fuentes de luz. Los segundos consisten en trazar caminos desde las fuentes de luz y almacenar información de irradiancia (i.e. luz incidente) en las superficies y medios de la escena. A su paso a través de la escena, la luz puede generar ciertos caminos de alta frecuencia (i.e. formando patrones estrechos en el espacio) producidos por medios y superficies con alta refracción o reflexión (p. ej. espejos, agua o vidrio). A estos caminos de alta frecuencia se les conoce como cáusticas (Figura 3.7), y son un caso particularmente complejo en render. Los algoritmos que trazan caminos empezando en el punto de vista no manejan bien las cáusticas, porque encontrar estos caminos de luz es estadísticamente muy improbable dada su alta frecuencia. Al contrario, empezar trazando caminos desde las fuentes de luz produce muy buenos resultados con cáusticas, ya que el enfoque no trata de encontrar caminos de luz sino de producirlos. En contraposición a las cáusticas, existen materiales cuya apariencia depende mucho del punto de vista, mostrando alta specularidad. Un ejemplo son los materiales denominados *glossy* (*brillante* en inglés). Ejemplos de materiales *glossy* son las bolas de Navidad o metales pulidos, que poseen la apariencia de un espejo desenfocado (Figura 3.8). En este caso, trazar caminos desde el punto de vista maneja muy bien estos tipos de materiales, mientras que trazarlos desde las luces falla. Elegir

Fuente: <http://en.wikipedia.org/>Figura 3.8: Imágenes de distintos materiales *glossy*.

un enfoque u otro dependerá de la naturaleza de la escena que queramos representar, llevando normalmente al uso de métodos híbridos. Una discusión más detallada sobre cómo afectan estos enfoques al render de medios participativos puede encontrarse en el Capítulo 6.

## Capítulo 4

# Transporte de Luz en Estado Transitorio

Como puede observarse en el Capítulo 3, la teoría del transporte de luz ignora cualquier información sobre el tiempo que le lleva a la luz recorrer el espacio. De hecho, una de las principales asunciones cuando se renderiza una escena es que la velocidad de la luz es infinita, lo cual es razonable siempre que queramos representar una escena como es vista por el ojo humano o por una cámara estándar. Los modelos de transporte de luz usados comúnmente en gráficos por ordenador normalmente no consideran el dominio temporal y sintetizan imágenes con toda la luz integrada en el tiempo, pero es bien sabido que la velocidad de la luz es finita, y si queremos representar cómo la luz se propaga físicamente por el espacio a escalas temporales muy pequeñas, debemos considerar el **tiempo como parte de las ecuaciones**. Con esta diferenciación, podemos distinguir entre el transporte de luz *estacionario* en el que la velocidad de la luz se asume infinita, frente al transporte en *estado transitorio* o simplemente transporte transitorio, en el que se considera el valor finito de la velocidad de la luz. Renderizar una escena considerando transporte de luz transitorio abre nuevas maneras de analizar el transporte de luz, desvelando información escondida en el dominio temporal que hasta ahora era integrada y descartada por las técnicas habituales de render estacionario. Relajar la asunción inicial y considerar velocidad de la luz finita nos ayuda a entender cómo interactúa y se propaga luz a través de una escena y nos permite ver cómo los algoritmos tradicionales de render se comportan al incluir el tiempo en las ecuaciones.

La luz tarda cierta cantidad de tiempo en recorrer una distancia en el espacio. Esto significa que el transporte de luz  $L(x, \vec{\omega}_o)$  desde un punto  $x_0$  hasta un punto  $x_1$  no ocurre de inmediato. Esto se puede expresar como

$$L(x_1, \vec{\omega}_o, t) = L(x_0, \vec{\omega}_o, t - \Delta t)$$

donde  $\vec{\omega}_o$  es una dirección saliendo de  $x_0$  hacia  $x_1$ , y  $\Delta t$  es el tiempo que le cuesta a la luz viajar de  $x_0$  a  $x_1$ . Dado que la velocidad de la luz depende del índice de refracción del medio que atraviesa,  $\Delta t$  está definido como

$$\Delta t(x_0 \leftrightarrow x_1) = \int_{x_0}^{x_1} \frac{\eta(x)}{c} dx \quad (4.1)$$

donde  $\eta(x)$  es el índice de refracción en  $x$  y  $c$  es la velocidad de la luz en el vacío:

$$c = 2,99792458 \times 10^8 \text{ m/s}$$

En un medio con un índice de refracción constante  $\eta(x) = \eta_m$ , entre  $x_0$  y  $x_1$ , la Ecuación 4.1 se puede expresar como

$$\Delta t(x_0 \leftrightarrow x_1) = \frac{\eta_m}{c} \|x_1 - x_0\| \quad (4.2)$$

En la naturaleza podemos encontrar medios con índice de refracción variable, como por ejemplo cerca de superficies muy calientes donde la temperatura del aire colindante cambia constantemente, produciendo efectos como los espejismos en una carretera vista a lo lejos o las ondulaciones cerca de la superficie metálica de un coche a plena luz del sol [GMAS05]. Por simplicidad, este proyecto asume que cada tipo de medio tiene un índice de refracción constante, considerando cambios únicamente en los límites de los medios. Otros efectos como la fase de las ondas electromagnéticas que tendrían efectos a las escalas temporales usadas en este proyecto también son ignorados por simplicidad.

Estrictamente hablando, la definición habitual del transporte de luz estacionario  $L(x, \vec{\omega}_o)$  es una generalización del transporte de luz transitorio  $L(x, \vec{\omega}_o, t)$  en la que la luz saliente de un punto en diferentes tiempos es integrado sobre un intervalo de exposición  $[0, t_{exp}]$ , suficientemente largo comparado con la mínima velocidad de la luz en la escena  $c_{min} = c/\eta_{max}$  y la extensión de la escena  $D_{max}$ :

$$L(x, \vec{\omega}_o) = \int_0^{t_{exp}} L(x, \vec{\omega}_o, t) dt, \quad t_{exp} \gg \frac{D_{max}}{c_{min}} \quad (4.3)$$

de forma que dentro de ese intervalo  $[0, t_{exp}]$  a la luz le ha dado tiempo a llegar a todos los puntos de la escena.

Como una magnitud de referencia, escalas temporales en el orden de picosegundos nos permiten ver luz viajando distancias del orden de milímetros o incluso menores. Por ejemplo, con un tiempo de exposición de 100ps, con un vídeo de 20 segundos a 15 frames/segundo podríamos ver la luz viajando lentamente una distancia de 9 metros.

## 4.1. Interacción de luz con superficies

La *Ecuación de Render* [Kaj86] (Capítulo 3, Ecuación 3.1) modela interacciones de luz entre superficies de la escena, describiendo la luz que sale de un punto de una superficie con dirección  $\vec{\omega}_o$ . Esta ecuación asume que la velocidad de la luz es infinita, por lo que todas las interacciones ocurren simultáneamente. Para incluir el tiempo que la luz tarda en propagarse a lo largo de la escena podemos reformular la Ecuación 3.1 como

$$L(x, \vec{\omega}_o, t) = L_e(x, \vec{\omega}_o, t) + \int_{\Omega} f_r(x, \vec{\omega}_i, \vec{\omega}_o) L(x, \vec{\omega}_i, t) (\vec{n} \cdot \vec{\omega}_i) d\vec{\omega}_i \quad (4.4)$$

Esta versión transitoria de la *Ecuación de Render* es en realidad una simplificación que asume que la luz es reflejada de manera instantánea en la superficie, pero esto no es completamente cierto. Rebotes micrométricos dentro de la microgeometría de la superficie, o retardos temporales debidos a la absorción y re-radiación en medios fluorescentes cobran relevancia a escalas temporales comparables con la velocidad de la luz y deberían ser representadas con una *BRDF*

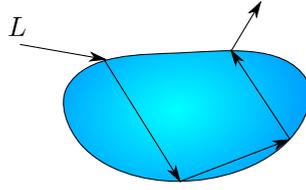


Figura 4.1: Caminos de luz dentro de una gota de agua

$f_r(x, \vec{\omega}_i, \vec{\omega}_o, t)$  dependiente del tiempo. Sin embargo, como se ha mencionado ya, este proyecto está centrado en el transporte de luz transitorio en medios participativos y sólo se consideran modelos de *BRDFs* muy simples, por lo que se ignora este tipo de fenómenos.

## 4.2. Interacción de luz con medios participativos

Del mismo modo que la *Ecuación de Render* modela interreflexiones de luz entre superficies, la RTE [Cha60] modela el transporte de luz en medios participativos. En su forma integral (Ecuación 3.3), podemos observar claras similitudes con la *Ecuación de Render*, por lo que adaptar la RTE para incluir el tiempo sigue un procedimiento similar. Para mostrar dependencia temporal en la RTE podemos reescribirla como

$$L(x, \vec{\omega}_o, t) = T_r(x, x_p)L_r(x_p, \vec{\omega}_o, t) + \int_{s_0}^{s_1} T_r(x, x_s)L_i(x_s, \vec{\omega}_o, t)ds \quad (4.5)$$

donde  $x_p$  es el punto más cercano visto desde la dirección  $\vec{\omega}_o$ , y  $L_i(x, \vec{\omega}_o, t)$  representa dispersión entrante dependiente del tiempo como

$$L_i(x, \vec{\omega}_o, t) = \sigma_s(x) \int_{\Omega_{4\pi}} \rho(x, \vec{\omega}_i, \vec{\omega}_o)L(x, \vec{\omega}_i, t)d\vec{\omega}_i \quad (4.6)$$

De nuevo, debemos reconsiderar las asunciones en esta formulación. Los coeficientes de absorción y dispersión son claramente independientes del tiempo, ya que variaciones en la densidad del medio debidas al tamaño de partículas y su concentración se producen a escalas temporales no comparables con el la velocidad de la luz. En consecuencia, dado que la transmitancia  $T_r$  es una función de la absorción y la dispersión representando el ratio de luz que no se extingue, es también independiente del tiempo. Sin embargo, la independencia temporal de la función de fase es un poco mas delicada. De forma parecida a las superficies, cuando la luz interactúa con una partícula micrométrica, puede seguir caminos dentro de la partícula antes de volver a ser redirigida hacia el exterior. Por ejemplo, las pequeñas gotas de agua en una nube tienen tamaños de alrededor de los 0,02mm o incluso mayores, y la luz es ralentizada debido al índice de refracción del agua  $\eta \approx 1,33$ . Así que cuando todas estas gotas están concentradas en una nube, los caminos de luz dentro de ellas son significativos a escalas temporales pequeñas (Figura 4.1). Si modelamos una nube como un medio participativo, este comportamiento de la luz con las gotas está definido por la función de fase [SML<sup>+</sup>12], y por tanto debería ser dependiente del tiempo  $\rho(x, \vec{\omega}_i, \vec{\omega}_o, t)$ . Esta dependencia añade otro nivel de complejidad al problema de render, por lo que este proyecto asume que las funciones de fase redirigen la luz instantáneamente (i.e. son independientes del tiempo).



## Capítulo 5

# Transporte de Luz en 2D

Como hemos observado, la teoría de transporte de luz está formulada para un espacio tridimensional ya que modela lo que pasa en el mundo real. Sin embargo, cuando representamos estos modelos mediante un ordenador, reducir la dimensionalidad del problema es una estrategia útil que proporciona mejor entendimiento y reduce la complejidad de los cálculos. En particular, cuando modelamos el transporte de luz en una escena, reducir la dimensionalidad espacial a 2D simplifica las formulaciones y permite prototipar y representar de manera mucho más intuitiva los casos de estudio simples.

Como se menciona en el Capítulo 4, uno de los principales objetivos de este proyecto es analizar el transporte de luz en estado transitorio, que esencialmente consiste en incluir el tiempo de propagación de la luz en la escena como una dimensión adicional  $t$  en la teoría de transporte de luz. Una de las principales ventajas es la existencia de ortogonalidad entre la adaptación a 2D y el transporte de luz transitorio. Esto es posible debido a que el transporte de luz transitorio principalmente conlleva computar tiempo en caminos unidimensionales de luz, que están presentes tanto en espacios 3D como 2D. Sin embargo, el incremento de dimensionalidad debido a la inclusión del tiempo hace prácticamente imposible realizar un análisis del transporte de luz en condiciones si consideramos un mundo 3D. La proyección de una escena 3D mediante una cámara se convierte en una imagen 2D (Figura 5.1a). Si añadimos la dimensión temporal,

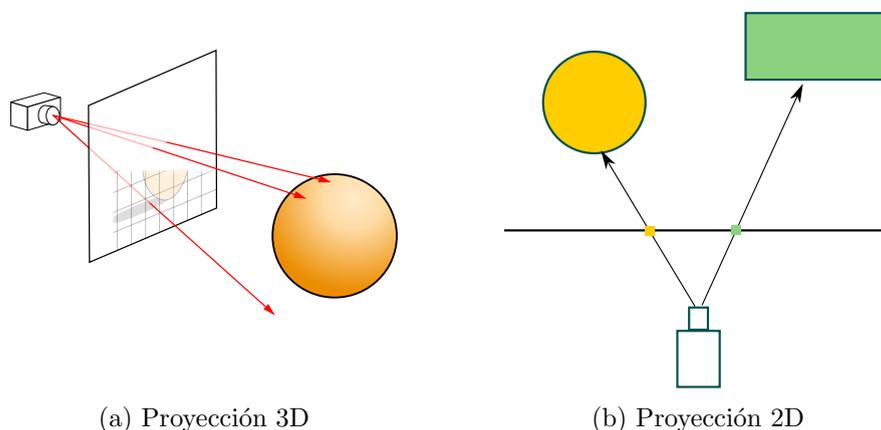


Figura 5.1: La proyección de una cámara visualizando una escena 3D (a) se convierte en una imagen 2D, mientras que en el caso de una escena 2D (b) se convierte en una imagen 1D

tenemos que representarla mediante un vídeo, que puede ser una manera atractiva de visualizar una escena, pero no permite observar variaciones temporales de manera rápida e intuitiva. Por el contrario, una proyección de un mundo 2D se convierte en una imagen 1D con una única fila de pixels (Figura 5.1b), por lo que podemos representar el tiempo en la dimensión sobrante de la imagen. En el Capítulo 8 se explica otro sistema de representación adicional que ha sido requerido para visualizar las escenas 2D.

Para renderizar una escena y definir el transporte de luz en 2D, necesitamos definir equivalencias 3D tanto geométricas como radiométricas para un mundo 2D. En geometría, las primitivas 3D tienen sus equivalencias en 2D con definiciones más simples, y las intersecciones rayo-primitiva—una de las operaciones más repetidas en render—normalmente tienen ecuaciones mucho más simples en 2D, reduciendo significativamente los costes. En radiometría, la integración de la luz sobre direcciones esféricas o hemisféricas 3D se convierte en la integración de círculos y semicírculos en 2D, requiriendo un menor número de muestras para alcanzar más convergencia en los resultados.

Las unidades radiométricas y definiciones también deben ser reconsideradas. Se puede encontrar un modelo radiométrico formalizado en 2D por Jarosz et al. en [JSKJ12] (Figura 5.2), definido dentro de un mundo 2D auto-contenido donde todas las interacciones de luz ocurren dentro de él, y no como una sección plana extraída de un mundo 3D. Aunque parte de este modelo es igualmente válido para medios participativos, está principalmente orientado al transporte de luz en *free-space*, i.e. interreflexiones entre superficies a través del vacío. Este capítulo hace una explicación del modelo radiométrico 2D de Jarosz et. al, centrándose en los principales aspectos que se han tenido en cuenta en este proyecto, y lo **extiende** para incluir consideraciones extra relacionadas con medios participativos.

## 5.1. Geometría

La consecuencia más evidente al reducir la dimensionalidad espacial a 2D es el cambio en primitivas geométricas. En términos generales las superficies se convierten en curvas, y los polígonos se convierten en líneas rectas. Como se ha mencionado antes, las intersecciones rayo-primitiva son una de las operaciones más repetidas en render, y son en general resueltas por sistemas de ecuaciones. Migrar de un espacio  $\{x, y, z\}$  a un espacio  $\{x, y\}$  reduce el número de

Quantity	Symbol	3D		2D	
		Expression	Units	Expression	Units
Spectral Radiant Energy	$Q_\lambda$	$Q_\lambda = n_\lambda \frac{hc}{\lambda}$	J	$Q_\lambda = n_\lambda \frac{hc}{\lambda}$	J
Radiant Energy	$Q$	$Q = \int_0^\infty Q_\lambda d\lambda$	J	$Q = \int_0^\infty Q_\lambda d\lambda$	J
Flux	$\Phi$	$\Phi_{3D}(A) = \frac{dQ(A)}{dt}$	$W = J \cdot s^{-1}$	$\Phi_{2D}(\mathcal{L}) = \frac{dQ(\mathcal{L})}{dt}$	$W = J \cdot s^{-1}$
Irradiance	$E$	$E_{3D}(\mathbf{x}) = \frac{d\Phi_{3D}(A)}{da(\mathbf{x})}$	$W \cdot m^{-2}$	$E_{2D}(\mathbf{x}) = \frac{d\Phi_{2D}(\mathcal{L})}{dl(\mathbf{x})}$	$W \cdot m^{-1}$
Radiosity	$B$	$B_{3D}(\mathbf{x}) = \frac{d\Phi_{3D}(A)}{da(\mathbf{x})}$	$W \cdot m^{-2}$	$B_{2D}(\mathbf{x}) = \frac{d\Phi_{2D}(\mathcal{L})}{dl(\mathbf{x})}$	$W \cdot m^{-1}$
Radiance	$L$	$L_{3D}(\mathbf{x}, \vec{\omega}) = \frac{d^2\Phi_{3D}(A)}{(\vec{n} \cdot \vec{\omega})^+ d\vec{\omega} da(\mathbf{x})}$	$W \cdot m^{-2} \cdot sr^{-1}$	$L_{2D}(\mathbf{x}, \theta) = \frac{d^2\Phi_{2D}(\mathcal{L})}{\cos\theta d\theta dl(\mathbf{x})}$	$W \cdot m^{-1} \cdot rad^{-1}$

Figura 5.2: Equivalencias de unidades radiométricas para 2D y 3D extraídas de [JSKJ12]. Esta tabla se muestra únicamente por completitud, y la notación de este proyecto puede diferir ligeramente.

ecuaciones requeridas para resolver estas operaciones y en consecuencia el número de cálculos. Muchas de las primitivas geométricas no son sólo usadas para geometría visible. Algunas estructuras de datos no-visibles utilizan esferas, cilindros o cajas, que en 2D se convierten en simples círculos y rectángulos con costes de intersección mucho menores. Los costes de memoria también se reducen ya que únicamente se necesitan guardar pares de coordenadas  $\{x, y\}$  para vértices, posiciones o vectores, y parametrizar un vector 3D mediante ángulos  $\{\phi, \theta\}$  se reduce a un único ángulo  $\theta$  en 2D.

## 5.2. Radiometría en superficies

La reflexión de luz en una superficie está caracterizada por la *BRDF* de la superficie (ver Capítulo 3 para más detalles). Dado que una *BRDF* define como se refleja la luz alrededor de un punto, la versión 2D de una *BRDF* define esta reflexión alrededor de un semicírculo en vez de el habitual hemisferio 3D. Las *BRDFs* están por definición normalizadas, lo que significa que la integral alrededor del semicírculo debe ser igual o menor que 1. Si pretendemos definir el equivalente 2D de una *BRDF* debemos asegurar que está normalizado. Si no es así no se garantiza conservación de la energía en los cálculos, ya que la cantidad de luz reflejada por una superficie podría ser mayor que la incidente, llevando a cálculos energéticamente inconsistentes. Una de las *BRDFs* más comunes es la *BRDF Lambertiana*, que simula una superficie reflejando la luz incidente en todas las direcciones uniformemente, i.e.  $f_r=1$  sin normalización. Con este valor, la integral  $\int_{\Omega} f_r \cos(\vec{\omega}, \vec{n}) d\vec{\omega}$  sobre el semicírculo es igual a 2 y sobre un hemisferio es igual a  $\pi$ , que en consecuencia se convierten en los factores de normalización teniendo  $f_{r_{3D}} = \frac{1}{\pi}$  y  $f_{r_{2D}} = \frac{1}{2}$ . Otros modelos proporcionan *BRDFs* mucho más complejas como los modelos de Blinn-Phong o Ward. La normalización 2D de estas *BRDFs* no es trivial y a veces es imposible de computar analíticamente. Dado que este proyecto se centra en medios participativos, hemos elegido representar únicamente modelos simples de *BRDFs*. Por otro lado, integrar la luz incidente en un punto de una superficie se reduce de un hemisferio a un semicírculo, con lo que los algoritmos estocásticos habituales necesitan de un número mucho menor de muestras para converger.

## 5.3. Radiometría en medios participativos

Los medios participativos cubren la mayor parte de este proyecto y por ello hemos prestado especial importancia en adaptar la radiometría a 2D, incluyendo consideraciones sobre la integración de dispersión, normalización de funciones de fase y derivación de coeficientes de dispersión y absorción en un mundo 2D. Estas consideraciones suponen una **contribución** al modelo radiométrico formalizado de Jarosz et al. previamente mencionado.

La luz incidente alrededor de un punto  $x$  en un medio requiere de la integración de todas las direcciones esféricas en 3D, parametrizadas por dos ángulos  $\phi$  y  $\theta$  (Figura 5.3, izquierda). El equivalente 2D se convierte en direcciones circulares alrededor de  $x$ , parametrizado ahora por un único ángulo  $\theta$  (Figura 5.3, derecha). De nuevo, esta reducción de dimensionalidad permite utilizar un menor número de muestras para obtener buenos resultados, y disminuye los costes de cálculo y memoria.

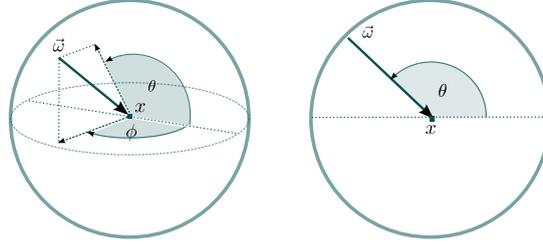


Figura 5.3: Parametrización mediante ángulos de dirección de la dispersión entrante para 3D (izquierda) y 2D (derecha).

### 5.3.1. Funciones de fase

La función de fase define la distribución de la luz que se dispersa hacia otras direcciones en un punto  $x$  dentro de un medio. Está definida como una función de distribución de probabilidad, y por ello su integral sobre las direcciones alrededor de  $x$  debe ser 1. En el caso de 2D, esta integral se debe calcular sobre todas las direcciones esféricas. De manera similar a la integración de *BRDFs*, si no garantizamos esto, podría llevar a modelos energéticamente inconsistentes que dispersarían más luz de la que llega. Uno de los modelos más usados es la función de fase Henyey-Greenstein [HG41]. Originalmente definida en 3D (Ecuación 5.1), la relación entre la dirección entrante y la saliente es parametrizada por dos ángulos  $\phi$  y  $\theta$ . Dado que es una función simétrica alrededor de uno de los dos ejes, su ecuación solo depende de uno de los ángulos  $\theta$ . El parámetro  $g$  representa el valor medio del coseno  $\cos(\theta)$  entre la dirección entrante y todas las direcciones salientes, y define la direccionalidad de la función de fase. En este proyecto se ha comprobado la normalización de una versión 2D de esta función [Dav06], y se ha utilizado para representar diferentes distribuciones de dispersión en 2D.

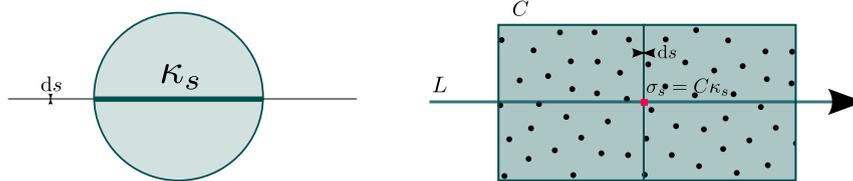
$$\rho(\theta, g) = \frac{1}{4\pi} \frac{1 - g^2}{(1 + g^2 - 2 \cos \theta)^{3/2}} \quad (\text{versión 3D}) \quad (5.1)$$

$$\rho(\theta, g) = \frac{1}{2\pi} \frac{1 - g^2}{1 + g^2 - 2 \cos \theta} \quad (\text{versión 2D}) \quad (5.2)$$

Obsérvese cómo la Ecuación 5.2 no incluye la razonablemente costosa operación de potencia. Esto reduce los costes computacionales, dado que en los algoritmos de render la función de fase es aplicada en cada evento de dispersión, llegando a tener cientos de miles de eventos de dispersión durante la generación de una imagen.

### 5.3.2. Coeficientes de dispersión y absorción

Como se explicó en la Sección 3.2, los coeficientes de dispersión  $\sigma_s$  y absorción  $\sigma_a$  definen el cambio proporcional de radiancia que se dispersa o absorbe en un paso diferencial. Para trasladar esto a 2D, debemos definir los equivalentes 2D de las secciones efectivas ( $\kappa_s$  y  $\kappa_a$ ) y de la concentración de partículas  $C$ . Una sección efectiva será definida ahora como la proyección de una partícula 2D sobre un paso diferencial (Figura 5.4a), convirtiéndose en una longitud ( $\text{m}^1$ ), que es de hecho el equivalente 2D de una sección efectiva en 3D (área). La concentración de partículas ahora será una densidad de partículas por espacio bidimensional, i.e. por área 2D  $\text{m}^{-2}$ , que es el equivalente 2D de un volumen 3D. Finalmente, calculando el equivalente 2D del coeficiente de dispersión (o de absorción) como  $\sigma_s = C\kappa_s$  tendremos que el coeficiente de



(a) La sección efectiva en 2D corresponde a una longitud  $m^{-1}$

(b) Multiplicando la densidad de partículas por la sección efectiva tenemos que  $\sigma_s = C\kappa_s m^{-1}$  coincidiendo con la definición original de coeficientes de dispersión y absorción en 3D.

Figura 5.4: Derivación del coeficiente de dispersión  $\sigma_s$  en 2D. El coeficiente de absorción  $\sigma_a$  se deriva de manera análoga.

dispersión sigue estando definido por longitud, coincidiendo con la naturaleza original de los coeficientes en 3D. Esto puede parecer una conclusión trivial, pero muestra que el modelo 3D de transporte de luz es igualmente adaptable a 2D sin perder significado.



## Capítulo 6

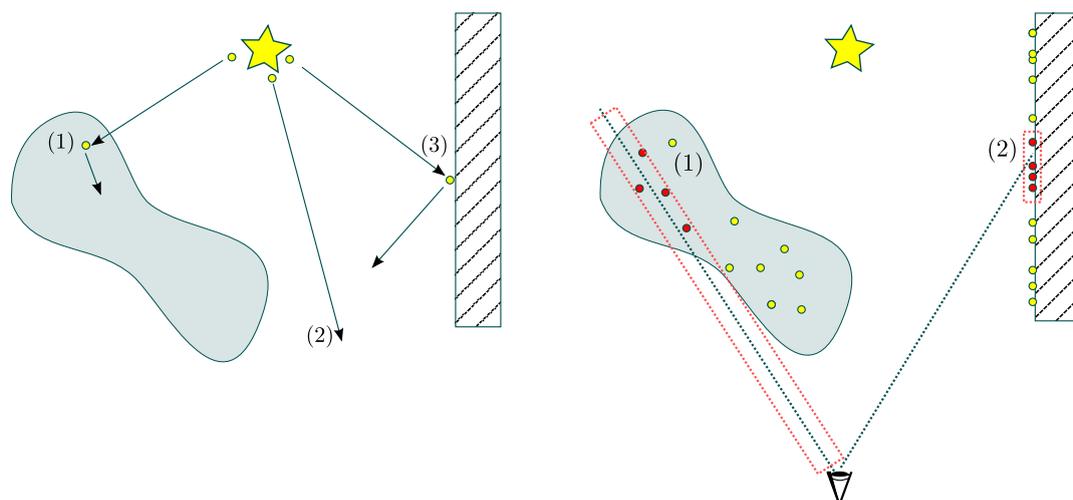
# *Photon Beams*

Como se menciona al final del Capítulo 3, para calcular la iluminación global en una escena debemos proporcionar soluciones numéricas que resuelvan tanto la *Ecuación de Render* (Ecuación 3.1) como la RTE (Ecuación 3.3). *Photon Beams* [JNSJ11] es un algoritmo para calcular iluminación global en medios participativos basado en el algoritmo *Photon Mapping* [Jen01]. Ambos resuelven de manera aproximada la *Ecuación de Render* y la RTE utilizando integración por Monte Carlo, y son algoritmos de dos pasadas. En la primera pasada, la escena es preprocesada para computar la distribución de luz en superficies y medios de la escena. Esta tarea se hace lanzando pequeños paquetes de energía (llamados *fotones*<sup>1</sup>) desde las fuentes de luz. Por medio de un proceso estocástico de Monte Carlo, estos fotones son trazados a través de la escena, y dispersados y absorbidos por la materia, produciendo un proceso de camino aleatorio. Estos fotones son almacenados en diferentes posiciones de la escena en lo que se denomina *mapa de fotones*. En un segundo paso, se trazan rayos desde el punto de vista para calcular información de radiancia en el mapa de fotones. Alcanzar un fotón en el mapa es estadísticamente imposible—ya que son puntos adimensionales—, así que para calcular radiancia en un punto de la escena se asume localidad espacial en un volumen o área alrededor de cada fotón. Esto equivale a realizar una estimación de densidad en cada punto de la escena alcanzado por los rayos del punto de vista. Los caminos aleatorios introducen varianza en el proceso, pero debido a las estimaciones de densidad, esta varianza es reducida y transformada en *bias* (i.e. un sesgo entre el valor calculado y el esperado), de forma que los resultados no muestran la solución correcta. Sin embargo, con un número infinito de fotones el resultado convergería a la solución correcta, por lo que los algoritmos basados en esta técnica son consistentes.

Podemos comparar los algoritmos basados en fotones con algoritmos basados en *Path Tracing* [Kaj86], como *Bidirectional Path Tracing* [LW93] o *Metropolis Light Transport* [VG97]. Estos algoritmos también están basados en un proceso de caminos aleatorios donde los caminos se trazan desde las luces y desde el punto de vista. Sin embargo, no realizan estimaciones de densidad, por lo que estos caminos aleatorios provocan varianza en los resultados (vista como ruido en la imagen final). Además, estos tipos de algoritmos fallan a la hora de calcular caminos de luz de alta frecuencia como las cáusticas, que son muy difíciles o incluso imposibles de encontrar cuando se trazan caminos desde el punto de vista. Por el contrario, los algoritmos basados en fotones son buenos calculando estos caminos de alta frecuencia. No obstante, existe un importante compromiso entre *bias* y varianza, y dependiendo de la naturaleza de los resultados que queramos obtener, elegiremos un tipo de algoritmo u otro.

---

<sup>1</sup>Mencionar que estos *fotones* no representan partículas cuánticas, sino pequeños paquetes de energía proporcional a la emisión de la fuente luz.



(a) Los fotones son trazados desde la luz contra la escena y pueden interactuar con una superficie (3), un medio (1) o perderse (2). El comportamiento después de una interacción dependerá de las propiedades de la materia.

(b) Estimación de radiancia con fotones sobre el mapa resultante. Se buscan fotones alrededor de cada punto en las superficies (2) o alrededor de cada rayo que atraviese un medio (1), y se computa la estimación de radiancia hacia el punto de vista.

Figura 6.1: Esquema del algoritmo *Photon Mapping* con dos fases: (a) Preproceso, (b) Estimación de radiancia

A pesar de que los algoritmos basados en fotones tienen importantes desventajas—principalmente el bias en los resultados y la necesidad de preprocesar la escena—, convergen a soluciones sin ruido mucho más rápido que los algoritmos basados en *Path Tracing*, especialmente cuando se trabaja con medios participativos. *Photon Beams* es uno de los últimos avances en investigación desarrollado por Jarosz et al. [JNSJ11], que explota incluso más todos los beneficios de los algoritmos basados en fotones. En concreto, incrementa la densidad de información en la escena guardando información de la iluminación no solo en puntos discretos de la escena (fotones), sino trayectorias de luz completas (los denominados *photon beams* o haces de fotones), lo que mejora la convergencia a soluciones con menos bias y varianza. Además, intuitivamente podemos ver que el aumento de cobertura espacial resultado del preproceso incrementa a su vez la cobertura temporal, ya que tiempo y espacio están directamente relacionados en el transporte de luz. Este proyecto implementa y modifica el algoritmo *Photon Beams*, y **realiza dos contribuciones** adaptándolo a 2D y al transporte de luz transitorio.

## 6.1. Trazado de fotones

Tanto en *Photon Beams* como en *Photon Mapping* el proceso de trazado es muy similar, y consiste en generar un mapa que represente la distribución de luz en la escena. En primer lugar, un número finito de fotones son lanzados desde las fuentes de luz contra la escena. Estos fotones están compuestos por tres características: posición, dirección y energía. Después de que un fotón ha sido lanzado, puede intersectar con una superficie, alcanzar un medio participativo o perderse si no intersecta con nada (Figura 6.1a). Este trazado se realiza mediante un proceso de Monte Carlo y en cada interacción, el fotón cambiará su comportamiento según las propiedades de la materia. En el caso de una superficie, el fotón puede ser reflejado hacia otra dirección y continuar interactuando con la escena, o ser absorbido, parándose el trazado de ese fotón. En el

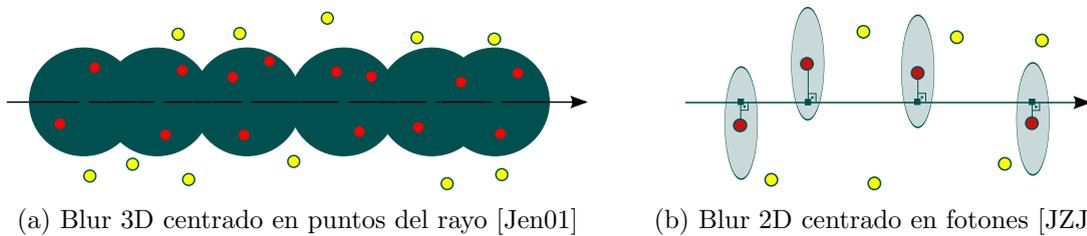
caso de un medio, el fotón continúa una distancia aleatoria, y su energía se reduce dependiendo de la transmitancia del medio. Entonces es absorbido o dispersado por el medio. Si es absorbido, se para el trazado de ese fotón. Si es dispersado, se elige una nueva dirección aleatoria para ese fotón y su energía se reduce por el coeficiente de dispersión  $\sigma_s$  y la función de fase del medio  $\rho$ . A este tipo de interacciones con el medio se las conoce como eventos de dispersión.

En cada interacción de un fotón con la materia, el fotón se almacena en un *mapa de fotones* en la posición donde ocurrió la interacción, incluyendo su energía y dirección. Cuando el trazado termina, obtenemos un mapa similar al que se ve en la Figura 6.1b. Este mapa de fotones representa una aproximación de la **irradiancia** en la escena, i.e. la luz incidente en cada posición de la escena. Más detalles sobre el trazado se pueden encontrar en [Jen01].

## 6.2. Estimación de radiancia utilizando *Photon Mapping*

Después de obtener el mapa de fotones, para calcular la luz reflejada en un punto  $x$  con una dirección  $\vec{\omega}$  (i.e. radiancia), *Photon Mapping* busca un determinado número de fotones cercanos a  $x$  y calcula una estimación de densidad [SG86] en el punto. El radio de búsqueda puede ser constante o adaptativo a número mínimo de fotones. Esta estimación de densidad representa una aproximación de la irradiancia en  $x$ , traducida como un *blur* alrededor de puntos en la escena. La radiancia saliente hacia  $\vec{\omega}$  dependerá de la irradiancia y de las propiedades de la materia. Si un rayo atraviesa un medio, la radiancia acumulada por ese rayo se calcula buscando fotones alrededor de ese rayo (Figura 6.1b). Sin embargo, un rayo está compuesto por infinitos puntos y es irrealizable calcular un número infinito de estimaciones de radiancia. *Photon Mapping* discretiza el rayo en un número finito de puntos, y recorre el rayo realizando una estimación de radiancia en cada uno de esos puntos (Figura 6.2a). Acudir a [Jen01] para más detalles.

Otro enfoque para realizar estimaciones de radiancia en medios participativos es propuesto en [JZJ08]. En este caso, en vez de buscar fotones alrededor de puntos discretizados del rayo, el algoritmo busca el rayo alrededor de cada fotón (Figura 6.2b). De este modo, cada fotón tiene su propia contribución sobre el punto más cercano del rayo, y la dimensionalidad del blur se reduce a 2D frente al blur 3D previamente explicado.



(a) Blur 3D centrado en puntos del rayo [Jen01]

(b) Blur 2D centrado en fotones [JZJ08]

Figura 6.2: Diferentes estimaciones de radiancia sobre el rayo usando fotones. *Photon Mapping* (a) acumula radiancia realizando estimaciones de densidad alrededor de puntos del rayo. Jarosz et al. proponen otro enfoque (b), en el que la búsqueda se centra en fotones alrededor de un disco 2D, por lo que cada fotón tiene una única contribución sobre el punto más cercano del rayo.

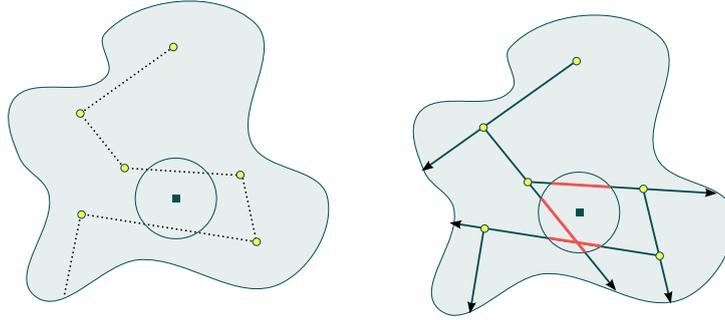


Figura 6.3: Rellenar un medio usando *photon beams* (derecha) en vez de con *photon points* (izquierda) incrementa la densidad de información de la escena, produciendo mejores resultados utilizando el mismo número de *photon points* y *photon beams*.

### 6.3. Estimación de radiancia usando *Photon Beams*

En *Photon Mapping*, se almacenan fotones en posiciones de la materia, representando la irradiancia en esos puntos concretos del espacio. Pero desde que los fotones son emitidos desde una luz hasta que llegan a esos puntos, representan energía viajando a través de la escena. En el vacío esta energía es irrelevante, debido a que potencialmente no existen interacciones. Pero dentro de un medio, un fotón pierde energía debido a la extinción y dispersa energía en otras direcciones. Sin embargo, debido a la naturaleza estocástica del trazado, solo se muestrea un número finito de eventos de dispersión, y el resto de potenciales eventos de dispersión se ignoran. En *Photon Mapping*, esta información potencial simplemente se descarta y solo se usa la energía de un fotón en la posición donde ocurrió el evento para realizar estimaciones de radiancia.

Pero cuando calculamos la radiancia en un punto del medio, se pueden tener en cuenta no solo fotones cercanos, sino todas las trayectorias cercanas de otros fotones, donde potencialmente ocurren eventos de dispersión (Figura 6.3). *Photon Beams* llena el espacio más densamente no solo almacenando fotones en sus posiciones, sino toda su trayectoria en una estructura llamada *photon beam* (haz de fotón). Por ello, podemos distinguir entre *photon points* (hasta ahora llamados fotones), y *photon beams*. La utilización de *photon beams* conlleva nuevos métodos de estimación de radiancia propuestos por Jarosz et al. [JNSJ11]. El proceso de trazado de los nuevos *photon beams* prácticamente es idéntico, salvando que también hay que guardar la distancia máxima que viaja un fotón, aparte de su posición, dirección y energía. De ahora en adelante, por simplicidad en la formulación y dado que este proyecto se centra en medios homogéneos, todas las derivaciones están hechas asumiendo este tipo de medio.

#### 6.3.1. Estimación de radiancia de un punto

Al comienzo del *photon beam*  $x_b$ , un fotón tiene energía  $\Phi_b$ . Debido a la extinción del medio  $\sigma_t$ , este fotón pierde energía conforme viaja a lo largo de la trayectoria del *beam*. Así, en cualquier punto de la trayectoria a una distancia  $s_b$  del comienzo del *beam*, un fotón tiene energía  $\Phi_b e^{-\sigma_t s_b}$ . De manera similar a *Photon Mapping* (Figura 6.2a), podemos calcular una estimación de radiancia alrededor de un punto buscando *photon beams* cercanos. Dado un radio de búsqueda  $r$ , la región de búsqueda  $R$  (i.e. una esfera) alrededor de un punto  $x$  intersecta cada *photon beam* cercano en dos puntos a distancias  $s_b^+$  y  $s_b^-$  (Figura 6.4a). Por tanto, la dispersión

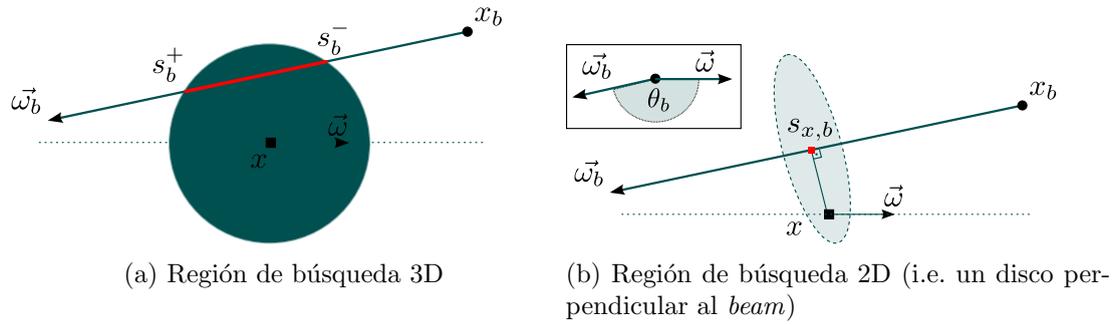


Figura 6.4: Diferentes kernels para estimar radiancia usando *Photon Beams*. Con una región de búsqueda 3D (a), la contribución de un *photon beam* sobre  $x$  está determinada por el segmento  $[s_b^-, s_b^+]$  (Ecuación 6.1). Con una región de búsqueda 2D (b), la contribución está determinada por la distancia  $s_{x,b}$  (Ecuación 6.2).

entrante resultará en una radiancia saliente en  $x$  hacia  $\vec{\omega}$  definida por

$$L_i(x, \vec{\omega}) \approx \frac{1}{\mu_R(r^3)} \rho(\theta_b) \Phi_b \int_{s_b^-}^{s_b^+} e^{-\sigma_t s_b} ds_b \quad (6.1)$$

para un único *photon beam*, donde  $\theta_b$  es el ángulo entre  $\vec{\omega}$  y  $\vec{\omega}_b$ , y  $\mu_R(r^3)$  representa el kernel de blur volumétrico como una función del radio de búsqueda. Un kernel unif, por ejemplo podría ser el volumen de la región de búsqueda  $\frac{4}{3}\pi r^3$ . En un medio homogéneo, la integral puede ser resuelta analíticamente como  $\sigma_t^{-1}(e^{-\sigma_t s_b^-} - e^{-\sigma_t s_b^+})$ . Este enfoque realiza un blur a lo largo del *beam*, producto del método de búsqueda. Sin embargo un *beam* nos proporciona el valor exacto del fotón en cada punto de su trayectoria, y este tipo de blur 3D solo sirve para introducir bias al hacer blur de un segmento del *beam*. Se puede usar a cambio un método de búsqueda similar al mostrado en Figura 6.2b, pero esta vez calculando la radiancia en un punto en vez de un rayo, y utilizando un *photon beam* en vez de un *photon point*. La región de búsqueda es un disco perpendicular al *beam*, dando un distancia de intersección  $s_{x,b}$  proyectada desde el punto  $x$  sobre el *beam* (Figura 6.4b). Ahora, la radiancia en  $x$  hacia  $\vec{\omega}_b$  producida por la dispersión entrante es

$$L_i(x, \vec{\omega}) \approx \frac{1}{\mu_R(r^2)} \rho(\theta_b) \Phi_b e^{-\sigma_t s_{x,b}} \quad (6.2)$$

Nótese que  $\mu_R$  es ahora una función de  $r^2$  en vez de  $r^3$ , dado que estamos haciendo blur sobre un área en vez de un volumen. Además, la región de búsqueda está centrada en el *beam* en vez de en el punto. Esto es análogo a centrar la región en el punto—dado que la proyección sobre  $s_{x,b}$  es la misma—y conduce más intuitivamente a las estimaciones de radiancia sobre el rayo explicadas más adelante. Contabilizando todos los *beams* cercanos cuya región de búsqueda 2D intersecta con  $x$  (expresando esto como  $b \in R_b$ ), podemos expresar la radiancia por la dispersión entrante como una suma:

$$L_i(x, \vec{\omega}) \approx \frac{1}{\mu_R(r^2)} \sum_{b \in R_b} \rho(\theta_b) \Phi_b e^{-\sigma_t s_{x,b}} \quad (6.3)$$

Esta ecuación resuelve de manera aproximada el término de dispersión entrante (Ecuación 3.2) de la RTE (Ecuación 3.3) en un único punto  $x$ .

### 6.3.2. Estimación de radiancia de un rayo

La estimación de radiancia en un punto nos permite calcular la radiancia producida por la dispersión entrante en ese punto, pero en general estamos interesados en calcular la radiancia acumulada por todo un rayo (el término integral en la Ecuación 3.3). Un rayo es en realidad una colección de infinitos puntos a lo largo de una dirección, y la estimación de radiancia en ese rayo puede derivarse extendiendo las estimaciones de radiancia de puntos que utilizan una región 2D (Ecuación 6.2). Empezando por la región de búsqueda 2D (Figura 6.4b), el punto  $x$  corresponderá ahora a un punto dentro del rayo  $x = x_r + s_r \vec{\omega}_r$ . Podemos continuar realizando varias búsquedas similares a lo largo del *beam* obteniendo puntos a lo largo del rayo con sus respectivas proyecciones sobre el *beam*  $x_b + s_b \vec{\omega}_b$  (Figura 6.5a). Si llevamos estas búsquedas alrededor del disco 2D al límite  $\Delta s_b \rightarrow 0$ , obtenemos un cilindro completo centrado en el *beam*. La intersección del rayo con este cilindro se convierte en un segmento delimitado por  $[s_r^-, s_r^+]$ . Podemos imaginar este segmento como un número infinito de puntos del rayo ( $\Delta s_r \rightarrow 0$ ) con sus respectivas proyecciones sobre el *beam*, que componen el correspondiente segmento del *beam*  $[s_b^-, s_b^+]$  (Figura 6.5b). Asumiendo un medio homogéneo, la dispersión entrante integrada sobre un rayo en la RTE (Ecuación 3.3) se puede escribir como

$$L(x_r, \vec{\omega}_r) = \sigma_s \int_{s_{min}}^{s_{max}} e^{-\sigma_t s_r} L_i(x_{s_r}, \vec{\omega}_r) ds_r \quad (6.4)$$

donde  $[s_{min}, s_{max}]$  es el intervalo del rayo dentro del medio.  $L_i$  tiene en cuenta toda la dispersión entrante en cada punto  $x_{s_r}$  a lo largo del rayo, que es aproximada por los *beams* cercanos a ese punto (Figura 6.4b). Utilizando los límites  $[s_r^-, s_r^+]$  de la intersección (Figura 6.5b) como límites de la integral, podemos insertar la Ecuación 6.2 en la Ecuación 6.4 para obtener la radiancia por dispersión entrante de un único *beam* sobre un rayo:

$$\begin{aligned} L(x_r, \vec{\omega}_r) &= \sigma_s \int_{s_r^-}^{s_r^+} e^{-\sigma_t s_r} \underbrace{\frac{1}{\mu_R(r^2)} \rho(\theta_b) \Phi_b e^{-\sigma_t s_b}}_{\text{Ecuación 6.2 con } s_b = s_{x,b}} ds_r \\ &= \frac{\sigma_s}{\mu_R(r^2)} \rho(\theta_b) \Phi_b \int_{s_r^-}^{s_r^+} e^{-\sigma_t s_r} e^{-\sigma_t s_b} ds_r \end{aligned} \quad (6.5)$$

Todos los valores  $s_b^\pm$  y  $s_r^\pm$  son conocidos, obtenidos por la intersección rayo-*beam*. Dado que cada punto  $s_b \in [s_b^-, s_b^+]$  es una proyección perpendicular de un punto  $s_r \in [s_r^-, s_r^+]$ , podemos parametrizar geoméricamente  $s_b$  en función de  $s_r$ , teniendo  $s_b(s_r) = s_b^- + \cos(\theta_b)(s_r - s_r^-)$ . Sustituyendo  $s_b$  en la Ecuación 6.5, la integral se resuelve analíticamente como:

$$\int_{s_r^-}^{s_r^+} e^{-\sigma_t s_r} e^{-\sigma_t s_b(s_r)} ds_r = \frac{e^{-\sigma_t (s_b^- + s_r^-)} - e^{-\sigma_t (s_b^- + \cos(\theta_b)(s_r^+ - s_r^-) + s_r^+)}}{\sigma_t \cos(\theta_b)} \quad (6.6)$$

Para calcular la estimación de radiancia completa, podemos hacer el sumatorio de la Ecuación 6.5 para cada *beam* intersectando al rayo ( $b \in R_b$ ):

$$L(x_r, \vec{\omega}_r) \approx \frac{\sigma_s}{\mu_R(r^2)} \sum_{b \in R_b} \rho(\theta_b) \Phi_b \underbrace{\int_{s_r^-}^{s_r^+} e^{-\sigma_t s_r} e^{-\sigma_t s_b} ds_r}_{\text{Resuelta analíticamente con la Ecuación 6.6}} \quad (6.7)$$

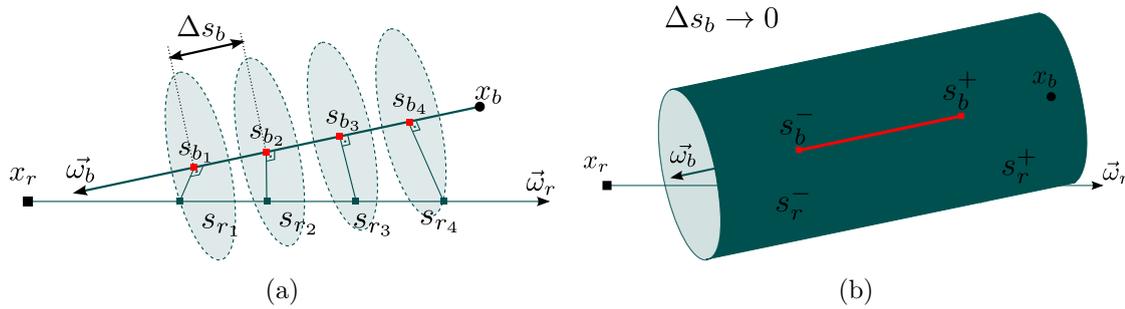


Figura 6.5: Las estimación de radiancia sobre el rayo puede hacerse tomando un infinito número de búsquedas 2D (a)  $\Delta s_b \rightarrow 0$  a lo largo de un *beam* para formar un blur 2D continuo (b)

Éste es un blur 2D, dado que cada punto del rayo tiene la contribución de cada *beam* encontrado en una búsqueda 2D alrededor de un disco. La dimensionalidad del blur puede ser reducida aún más utilizando un blur 1D. En términos generales, la idea es “comprimir” el cilindro del *beam* (Figura 6.5b) en un rectángulo alineado con el rayo. De este modo, obtenemos un único punto  $s_r$  en el rayo, y su proyección perpendicular  $s_b$  sobre el *beam*, representando el par de puntos más cercanos entre el rayo y el *beam* (Figura 6.6). Una vez hemos obtenido estos puntos, la estimación de radiancia de los *beams* cercanos es calculada como

$$L(x_r, \vec{\omega}_r) = \frac{\sigma_s}{\mu_R(r)} \sum_{b \in R_b} \rho(\theta_b) \Phi_b e^{-\sigma_t s_r} e^{-\sigma_t s_b} \quad (6.8)$$

Esta estimación es mucho más simple computacionalmente, y debido a que estamos reduciendo la dimensionalidad del blur—y por tanto el tamaño del kernel—, el bias es menor también. Más detalles sobre su derivación se pueden encontrar en [JNSJ11].

### 6.4. Photon Beams en 2D

Adaptar *Photon Beams* a 2D requiere modificar los métodos de estimación de radiancia. El proceso de trazado requiere reducir la dimensionalidad de las posiciones de los *beams*, trayectorias y funciones de muestreo, a la vez que considerar transporte de luz en 2D (Capítulo 5), pero la naturaleza del trazado estocástico no cambia. Sin embargo, adaptar las estimaciones de radiancia conlleva cambiar la naturaleza de los kernels tanto para *photon points* como para

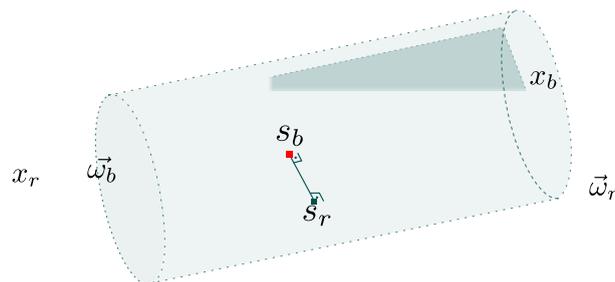


Figura 6.6: Blur kernel 1D resultado de comprimir el cilindro del *beam* mostrado en la Figura 6.5b en un rectángulo alineado con el rayo. La intersección nos da el par de puntos más cercanos entre el rayo y el *beam*.

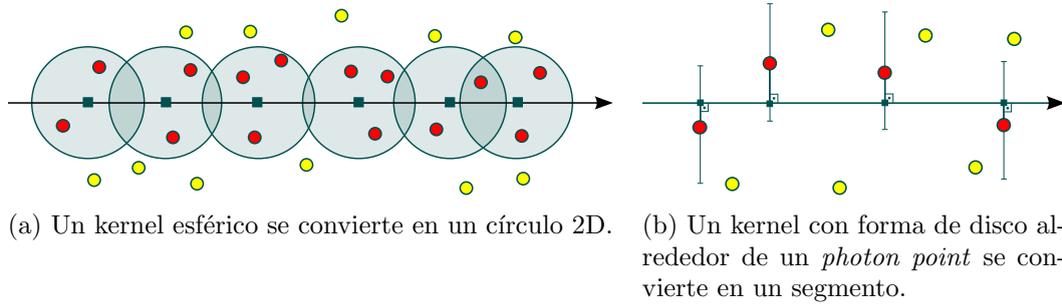


Figura 6.7: Equivalentes 2D de estimaciones de radiancia usando *photon points*.

*photon beams*, llevando a diferentes enfoques que algunas veces no son posibles en un mundo 2D. Si miramos brevemente a las estimaciones de radiancia con *photon points* (Sección 6.2), vemos que las búsquedas en esferas en un mundo 3D (Figura 6.2a) se convierten en círculos 2D en un mundo 2D (Figura 6.7a), manteniendo el proceso de discretización ya que se hace a lo largo del rayo. De manera similar, la búsqueda alrededor de un disco centrado en un *photon point* (Figura 6.2b), se convierte en una búsqueda a lo largo de una distancia en un mundo 2D (Figura 6.7b).

Del mismo modo que en un mundo 3D, podemos almacenar toda la trayectoria de un *photon point* 2D para obtener el *photon beam*. Derivar las estimaciones de radiancia, sin embargo, debe ser llevado a cabo con cuidado. En 3D, para obtener un blur 2D de un *photon beam* se tomaba un número infinito de búsquedas 2D alrededor de un disco a lo largo de la trayectoria del *beam* (Figura 6.5). En 2D podemos seguir el mismo procedimiento usando el equivalente segmento 1D (Figura 6.7b), y empezar a tomar puntos a lo largo del *beam* (Figura 6.8, izquierda). Cogiendo un número infinito de ellos, obtenemos el equivalente 2D del cilindro, que corresponde a un rectángulo (Figura 6.8, derecha). De este modo, el blur 2D del mundo 3D se convierte en un blur 1D en el mundo 2D. Dado que las proyecciones  $s_r \leftrightarrow s_b$  son geoméricamente equivalentes (comparar las Figuras 6.5a y 6.8), la Ecuación 6.6 permanece intacta y la equivalente en 2D de la Ecuación 6.7 solo necesita sustituir  $\mu_R(r^2)$  por  $\mu_R(r)$  para tener en cuenta la reducción de dimensionalidad del blur.

Pero, ¿y si queremos reducir aún más la dimensionalidad del blur? Parece—y en realidad lo es—imposible hacerlo. Un blur 0D no tiene sentido, dado que no estaríamos haciendo blur sobre ninguna dimensión. De hecho, si recordamos cómo se obtenía el blur 1D a partir del 2D en un mundo 3D (Figura 6.6), vemos que se obtiene comprimiendo el cilindro del *beam* en un rectángulo alineado con el rayo. En un mundo 2D, este cilindro “comprimido” estaría en una dimensión que en realidad no existe, por lo que no tiene ningún sentido geométrico realizar esta operación.

## 6.5. Implementación del algoritmo

El preproceso de trazado en la escena se realiza mediante un proceso estocástico de Monte Carlo, en el que se traza un número finito de *photon points* o *photon beams* desde las fuentes de luz contra la escena, y son guardados en una estructura de datos. Después de eso, el algoritmo calcula las estimaciones de radiancia utilizando esta estructura de datos. Desde un punto de vista teórico, los blurs están basados en buscar *photon points* o *beams* dentro de una región N-dimensional alrededor de un punto o rayo. En la práctica, esto puede ser llevado a cabo geoméricamente con intersecciones punto-primitiva o rayo-primitiva, en la que las regiones

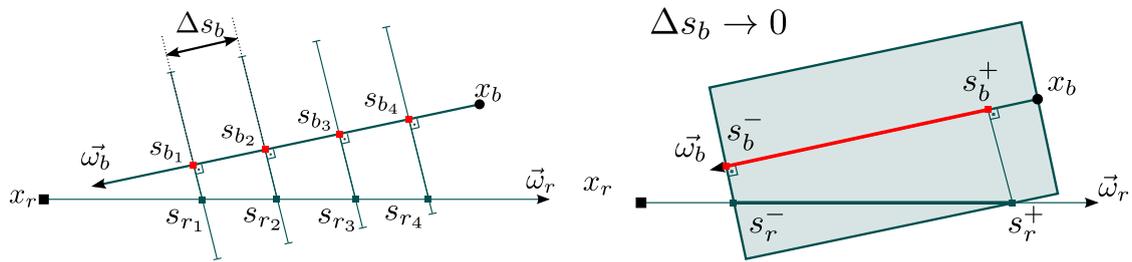


Figura 6.8: Equivalentes 2D de estimaciones de radiancia sobre el rayo 3D. Realizando búsquedas 1D alrededor de puntos del *beam* (izquierda) y llevando este proceso hasta que  $\Delta s_b \rightarrow 0$ , obtenemos el equivalente del blur 2D del mundo 3D para un mundo 2D, i.e. un rectángulo con sus puntos de intersección  $[s_b^\pm, s_r^\pm]$  (derecha).

de búsqueda pueden estar representadas por primitivas como esferas, cilindros o rectángulos. Normalmente se utilizan estructuras de aceleración para obtener mejor rendimiento al aplicar estas operaciones de intersección. Una búsqueda 2D alrededor de un disco (Figura 6.2b) se puede realizar comprobando que la distancia perpendicular entre un rayo y un *photon point* es más corta que el radio del disco. La búsqueda representada en la Figura 6.5b se puede realizar calculando la intersección rayo-cilindro que nos da los valores  $\{s_r^-, s_r^+, s_b^-, s_b^+\}$ . Su equivalente 2D (Figura 6.8) es la intersección rayo-rectángulo.

Dependiendo de si usamos *photon points* o *photon beams*, y qué blur queramos aplicar, la estructura de aceleración elegida para almacenarlos será diferente. *Photon Mapping* almacena los *photon points* en *kd-trees*, ya que están definidos como puntos discretos en el espacio, y los *kd-trees* permiten buscar eficientemente puntos cercanos a ellos. Sin embargo, los *photon beams* no están definidos como puntos discretos, sino como trayectorias con cierta extensión en el espacio. Además, los kernels usados con *photon beams* están basados en intersectar un rayo con primitivas centradas en la trayectoria del *beam*. En este proyecto se han usado BVHs [Hav04] (*bounding volume hierarchies*) para almacenar el mapa de *photon beams* de la escena. Las BVHs son comúnmente utilizadas para obtener el punto de intersección más cercano, pero en *Photon Beams* estamos interesados en obtener todos los beams cuyo cilindro intersecta con el rayo. Por ello, se ha realizado una adaptación del algoritmo original de BVH de modo que se obtienen todas las primitivas (*beams*) intersectadas.

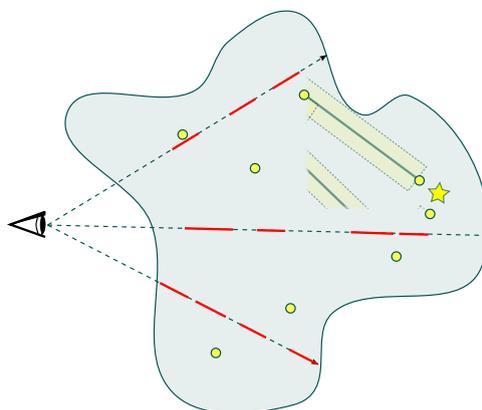


Figura 6.9: En *Photon Beams*, la estimación de radiancia en el medio se realiza intersectando cada rayo contra el mapa de *photon beams*, y usando las intersecciones obtenidas para calcular la estimación de radiancia en cada rayo.

Para calcular las estimaciones de radiancia para cada rayo lanzado desde el punto de vista, se intersecta el rayo contra la estructura de datos del mapa de photon beams, obteniendo los puntos de intersección de cada *beam* con el rayo (Figura 6.9). Después de eso, se estima la radiancia y se acumula para cada *beam* usando estos puntos junto con las soluciones analíticas derivadas anteriormente (Ecuaciones 6.7 y 6.8). En el caso de que un rayo intersecte con una superficie, se estima la radiancia en ese punto y se acumula aplicando la extinción del medio (si existe). En el Anexo C se puede encontrar un diagrama de clases que describe la estructura del motor de render sobre el que está implementado este proyecto.

## Capítulo 7

# Photon Beams Transitorios

En el Capítulo 6 se ha mostrado que el algoritmo *Photon Beams* incrementa la densidad de información espacial dentro de medios participativos, proporcionando una solución práctica para renderizar iluminación global. Pero más allá de eso, llenar un medio con *photon beams* es también muy adecuado para representar transporte de luz transitorio. El tiempo de propagación de la luz está directamente relacionado con las distancias que viaja. Los photon points solo proporcionan información de irradiancia en puntos discretos de la escena y por tanto en instantes también discretos, lo que supone una importante desventaja si se usan resoluciones temporales muy pequeñas. Sin embargo, **un *photon beam* es por definición continuo** ya que su trayectoria está parametrizada por una posición, dirección y longitud. Esto nos permite ser independientes de la resolución temporal, ya que para cada instante podemos saber exactamente donde está un fotón a lo largo del *beam*.

Para un punto a una distancia  $s_{b_i}$  del comienzo del *photon beam*  $s_{b_0} = 0$  (Figura 7.1), a un fotón le costará un tiempo  $\Delta t(s_{b_0} \leftrightarrow s_{b_i}) = \frac{\eta_m}{c} s_{b_i}$  llegar de  $s_{b_0}$  a  $s_{b_i}$  (Ecuación 4.2). Sin embargo, un fotón colocado en  $x_b$  puede haber tomado cierto tiempo en llegar hasta ahí desde que empezó a viajar desde la fuente de luz. Esto significa que el tiempo del fotón al comienzo  $s_{b_0} = 0$  del *photon beam* no tiene que ser necesariamente 0. Este valor  $t_{b_0}$  es calculado durante el proceso de trazado como una acumulación de todas las distancias previas (Figura 7.2). Por ello, para obtener el tiempo total que le cuesta a un fotón llegar de la fuente de luz a una posición dentro de un *photon beam* a distancia  $s_{b_i}$  desde el principio, tendremos que  $t_{b_i} = t_{b_0} + \Delta t(s_{b_0} \leftrightarrow s_{b_i})$ .



Figura 7.1: Un *photon beam* está por definición parametrizado, por lo que podemos saber el instante temporal a cualquier distancia  $s_{b_i}$ .

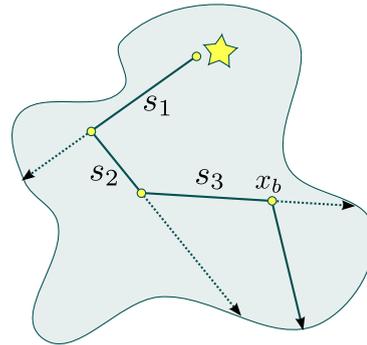


Figura 7.2: A un fotón en  $x_b$  le costará un tiempo  $t_{b_0} = \frac{\eta_m}{c} (s_1 + s_2 + s_3)$  llegar hasta allí desde que se emitió desde la fuente de luz.

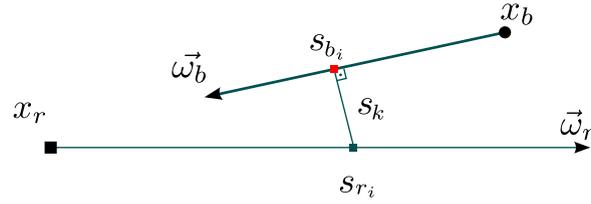


Figura 7.3: La radiancia en  $s_{r_i}$  procedente de  $s_{b_i}$  tardará  $t = t_{b_i} + \frac{\eta_m}{c} s_{r_i}$  en alcanzar  $x_r$ .

Esto permite representar transporte de luz transitorio con un fotón viajando a través del espacio y llegando a un punto dentro de su *photon beam*. El *photon beam* entonces es usado para obtener una estimación de radiancia sobre un rayo. De la misma manera que a un fotón le cuesta un tiempo viajar a través del espacio, a la radiancia acumulada en un punto de un rayo también le llevará un tiempo viajar desde ese punto hasta el punto de vista (Figura 7.3). Este tiempo no está modelado durante el trazado, y se ha de tener en cuenta durante la estimación de radiancia. Sin embargo, este tiempo de rayo puede producir efectos poco intuitivos como el que se muestra en la Figura 7.4, conllevando alteraciones en cómo visualizamos la propagación de la luz en la escena [VWJ<sup>+</sup>13]. A estas alteraciones se las denomina *camera-warping*, y tener en cuenta o no el tiempo de rayo llevará a renders *camera-warped* o *camera-unwarped* respectivamente. Elegir un enfoque u otro es una decisión de diseño que dependerá de cómo queramos visualizar la escena.

Pero, si tenemos en cuenta las distancias  $s_{b_i}$  y  $s_{r_i}$ , ¿por qué no contar también con la distancia  $s_k$  entre  $s_{b_i}$  y  $s_{r_i}$ ? La respuesta es simple: esta distancia es producto de la región de búsqueda de la estimación de radiancia, y no del transporte de luz como tal. Dicho de otro modo, estamos calculando el transporte de luz con una estimación de densidad (radiancia) que es a efectos una **aproximación matemática**, por lo que en realidad estamos modelando la probabilidad de que el transporte de luz ocurra con localidad espacial entre  $s_{b_i}$  y  $s_{r_i}$  (i.e.  $s_k = 0$ ).

Utilizando este enfoque podemos modelar blur kernels temporales para los blur kernel espaciales explicados en el Capítulo 6. Para nuestro principal objetivo (estimaciones de radiancia sobre el rayo) representamos el rango temporal de los blurs espaciales en la Figura 7.5. Como se puede ver, distribuir la radiancia a lo largo del tiempo de manera acorde con el rango espacial solo es posible cuando usamos un blur 2D en un mundo 3D y su equivalente blur 1D en un mundo 2D. Por su naturaleza, el blur 1D de un mundo 3D clacula la estimación de radiancia en base a un único par de puntos en el *beam*  $s_b$  y en el rayo  $s_r$ , por lo que no tiene sentido distribuir radiancia a lo largo del tiempo (ya que el tiempo para ese par de puntos es simplemente

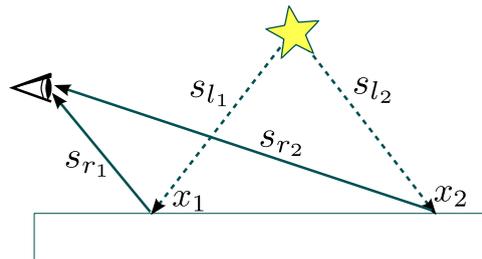


Figura 7.4: *Camera-warping*. Distancias de la luz a la superficie son idénticas  $s_{l_1} = s_{l_2}$ , mientras que las distancias de la superficie al punto de vista son distintas  $s_{r_1} < s_{r_2}$ . Si tenemos el tiempo de rayo en cuenta, veremos que la luz primero llega a  $x_1$  y después llegar a  $x_2$ . Sin embargo, esto no es cierto, ya que la luz llega a la vez  $\frac{\eta}{c} s_{l_1} = \frac{\eta}{c} s_{l_2}$  a ambos  $x_1$  y  $x_2$ .

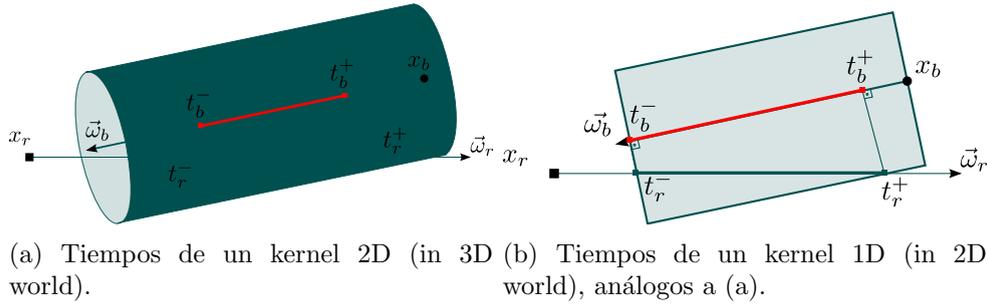


Figura 7.5: Los tiempos  $t_b^\pm, t_r^\pm$  se calculan a partir de las distancias  $s_b^\pm, s_r^\pm$  para ambos kernels 2D y 1D de mundos 2D y 3D respectivamente.

$t_b + t_r$ ). Así pues, para los primeros blurs mencionados, podemos derivar el rango de tiempo de propagación (en inglés *time-of-flight*, ToF) en el que se debe distribuir la radiancia:

$$\begin{aligned}
 c_m &= \frac{c}{\eta_m} && \text{(velocidad de la luz en el medio)} \\
 t_b^\pm &= \frac{s_b^\pm}{c_m} && \text{(ToF de la luz para } s_b^\pm) \\
 t_r^\pm &= \frac{s_r^\pm}{c_m} && \text{(ToF de la luz para } s_r^\pm) \\
 t_r^- &< t_r^+, \quad \Delta t_r = t_r^+ - t_r^- \\
 t^- &= t_r^- + t_b^- && \text{(ToF de la luz total para } s_r^-) \\
 t^+ &= t_r^+ + t_b^+ && \text{(ToF de la luz total para } s_r^+)
 \end{aligned}$$

Por geometría,

$$t^+ = \underbrace{t_r^- + \Delta t_r}_{t_r^+} + \underbrace{t_b^- + \cos(\theta_b)\Delta t_r}_{t_b^+}$$

$$\Delta t = t^+ - t^- = (1 + \cos(\theta_b))\Delta t_r$$

así, dependiendo de  $\cos(\theta_b)$

$$\Delta t \in [0, 2\Delta t_r]$$

Obsérvese que  $\Delta t \approx 0$  cuando el *beam* viaja mirando hacia el rayo, i.e.  $\cos(\theta_b) \approx -1$ . Esto significa que toda la luz llevada por un *photon beam* que está orientado hacia al punto de vista llega en el mismo instante  $t^-$ , visible en los vídeos renderizados (Capítulo 8) como un flash inicial proveniente de los caminos *beam*-rayo más cortos en el momento que se empieza a ver luz en las escenas.

Recordar que las estimaciones de radiancia de cada *beam* se computan analíticamente. Esto significa que el valor de la radiancia obtenida es el de la radiancia integrada dentro del kernel, y por lo tanto no calculamos la radiancia en cada punto separado de la intersección. Por ello, cuando distribuimos esta radiancia integrada a lo largo del intervalo  $\Delta t$ , tenemos que elegir una manera de hacerlo. Usando una aproximación simple, podemos distribuirla uniformemente a lo largo de todo el tiempo que le cuesta a la luz viajar a través de las segmentos de la intersección entre el rayo y el *beam* (Figura 7.6b). Nótese que distribuir la radiancia uniformemente incrementa el bias, dado que en realidad en cada instante temporal la luz tiene una extinción diferente debido a la distancia viajada a través del medio. Un trabajo futuro de este proyecto es proporcionar una solución correcta a este problema.

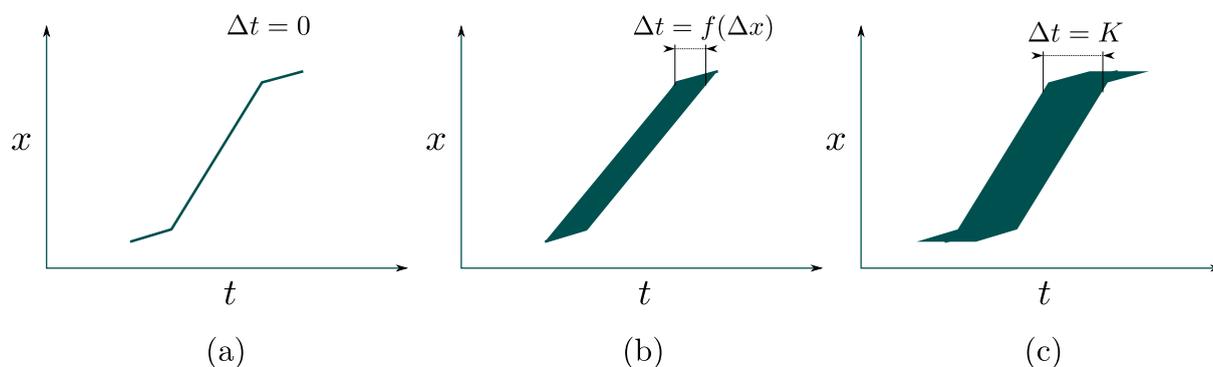


Figura 7.6: Kernels de blur temporal. En los gráficos de arriba representamos la evolución de un *beam* en la dimensión espacial  $x$  y en el tiempo  $t$  (los segmentos iniciales y finales se deben a los límites del cilindro). Cuando computamos las estimaciones de radiancia con blurs espaciales, podemos distribuir la radiancia a lo largo del tiempo con diferentes blurs temporales. En (a) no se aplica ningún blur  $\Delta t = 0$ , colocando la radiancia en el tiempo medio que le cuesta a un fotón atravesar la intersección *beam*-rayo. En (b), el blur en tiempo es dependiente del blur espacial  $\Delta t = f(\Delta x)$ , y la radiancia se distribuye en tiempos coherentes con la longitud de la intersección *beam*-rayo. En (c) el blur temporal es independiente del espacio, y simplemente elegimos un intervalo de tiempo constante  $\Delta t = K$ .

Este blur temporal es coherente con el espacial, por lo que su anchura en tiempo es muy dependiente del tamaño del radio de búsqueda. Esto puede llegar a ser un problema si el radio de búsqueda es muy grande comparado con la escala temporal utilizada, lo que se puede traducir en artefactos estructurados visibles en la secuencia de imágenes obtenida. Esto se puede evitar utilizando un blur temporal independiente del espacio (Figura 7.6c), por ejemplo fijando un rango temporal constante  $\Delta t = K$ , que es de hecho una operación muy simple que se puede realizar como un post-proceso de la imagen. De nuevo, esto incrementa el bias en los resultados y su uso dependerá de la precisión física que queramos obtener.

## Capítulo 8

# Resultados

Este capítulo presenta y discute los resultados más relevantes que hemos obtenido en este proyecto. Aunque este proyecto se centra en medios participativos, empezamos sentando las bases del transporte de luz transitorio con un ejemplo simple en una escena 2D *free-space* (i.e. sin medios participativos), analizando los perfiles temporales de radiancia sobre las superficies. Después de esto, presentamos varios ejemplos del transporte de luz en medios participativos usando *Photon Beams* tanto en escenas 2D como 3D incluyendo: discusión de las ventajas e inconvenientes de los diferentes métodos de medida, representación del transporte de luz dentro de un medio con geometría, comparación de diferentes perfiles de radiancia variando las funciones de fase, comparación de proyecciones *camera-warped* y *unwarped*, análisis de problemas de muestreo en renderizado transitorio comparando diferentes algoritmos, y comparación de la propagación de la luz con emisiones delta de Dirac o función de escalón de Heaviside. En el Anexo D se incluyen renders adicionales de este proyecto así como las versiones estacionarias de algunos renders en 3D de esta sección.

### 8.1. Perfiles temporales de radiancia en una escena 2D en *free-space*

A pesar de que esta escena no tiene medios participativos, explicamos este caso simple de estudio en 2D utilizando superficies Lambertianas para introducir el sistema de render de este proyecto y entender las bases del transporte de luz transitorio. Observando los perfiles temporales de radiancia (Figura 8.1b), podemos ver que la radiancia verde crece hacia la derecha, ya que las partes más a la derecha del muro están más lejos del sensor, y los rebotes de luz llegan más tarde. Por el contrario, dada la posición de la bola, la radiancia roja es simétrica respecto al centro del sensor. Podemos tomar el perfil temporal en un único punto del sensor (líneas amarillas) y mostrarlo en gráficas radiancia-tiempo (Figura 8.1c). En la gráfica de la izquierda podemos ver una función delta representando el frente de onda principal mostrado. Para rebotes de más orden, los perfiles temporales de radiancia son mucho más suaves, principalmente porque en cada rebote, la radiancia se convoluciona con los términos geométricos en el dominio temporal.

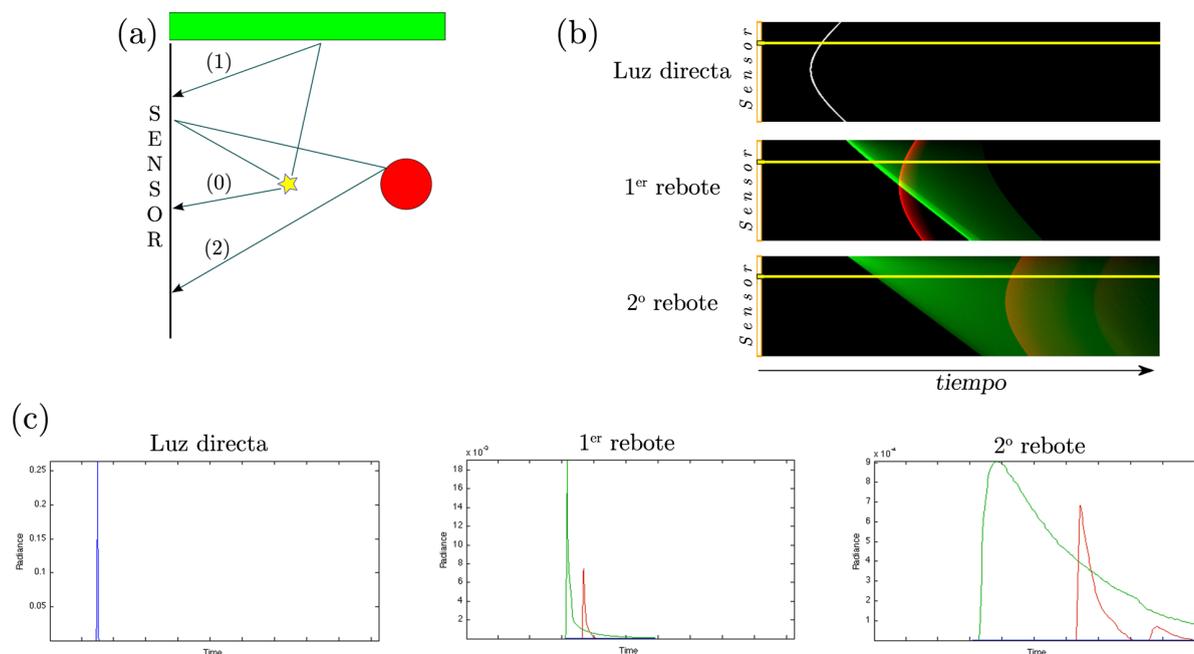


Figura 8.1: En (a) vemos una representación esquemática de una escena 2D, con una luz blanca en el centro emitiendo un pulso delta, un muro verde en la parte superior, y una bola roja en la derecha, todos ellos con materiales difusos (Lambertianos). El sensor a la izquierda actúa como un muro blanco sobre el que medimos radiancia. En (b) representamos luz multiplexada por nivel de rebote, i.e. separando la radiancia dependiendo del número de rebotes que le costó llegar al sensor. El eje vertical representa la parametrización del sensor, y el eje horizontal la evolución de la radiancia en el tiempo. En la imagen de luz directa, solo vemos el primer frente de onda directo (0), con la radiancia llegando más pronto cerca del centro del sensor, ya que está más cerca de la luz. En la imagen del primer rebote, vemos la luz que llegó al muro y la bola, rebotó y llegó al sensor (1). En la imagen del segundo rebote, vemos la radiancia que llegó primero al sensor, se reflejó sobre el muro y la bola y fue reflejada otra vez llegando al sensor (2). Las gráficas (c) representan la evolución de la radiancia en tiempo para un único punto del sensor (marcado en amarillo).

## 8.2. Proyección ortográfica de una luz puntual dentro de un medio participativo

La Figura 8.2, arriba-izquierda muestra una luz puntual dentro de un medio participativa y vista desde una cámara ortográfica. Las gráficas representan single y multiple scattering (luz sin dispersar y luz dispersada al menos una vez) para cuatro funciones de fase distintas en un instante  $t = 2,1\text{ns}$ . En la gráfica de single scattering podemos ver dos características. Los picos en los lados y los lóbulos en el centro, especialmente para funciones de fase con más direccionalidad  $g = \pm 0,6$ .

Los picos de los lados son producto de la acumulación sobre el rayo. La escena representa la emisión de un pulso delta de luz produciendo un frente de onda diferencial  $dx$  (1), pero dado que nuestro tiempo de exposición es finito, en la práctica la anchura del frente de onda también lo es. Cuando un rayo atraviesa el medio cerca de los lados izquierdo o derecho del frente de onda, la cantidad de frente acumulado sobre el rayo es mayor (2), por lo que se observan esos

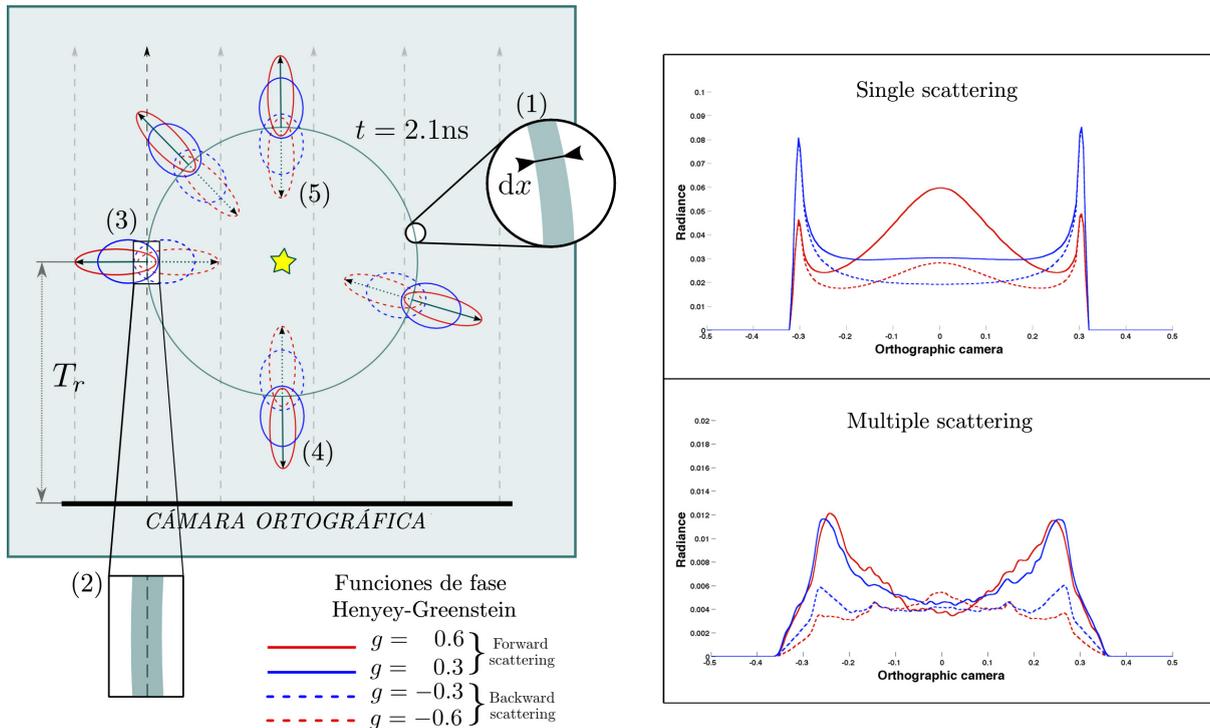


Figura 8.2: Una luz puntual dentro de un medio con  $\sigma_s = 0,6, \sigma_a = 0,1$  (arriba-izquierda). Mostramos diferentes perfiles temporales de radiancia para cuatro funciones de fase Henyey-Greenstein con distintos niveles de direccionalidad. En particular, separamos el single scattering (luz que no ha sufrido eventos de dispersión) del multiple scattering (luz que ha sido dispersada al menos una vez), y en las gráficas mostramos la radiancia vista por la cámara ortográfica en un instante  $t = 2,1\text{ns}$  desde que se lanzó el pulso de luz.

picos con más radiancia. Además para un mismo valor de  $|g|$ , la altura de los picos es idéntica en sus máximos. Esto es porque en esa zona del frente de onda, la luz viaja de forma perpendicular al rayo de cámara. Por ello, las funciones de fase son simétricas respecto a la dirección del rayo de cámara (3), reflejando la misma cantidad de luz hacia el rayo tanto para valores positivos como negativos de un mismo valor  $|g|$ . Esto puede verse en las elipses azules  $|g| = 0,3$  y rojas  $|g| = 0,6$  que muestran la orientación de la función de fase en cada punto del frente de onda.

Los lóbulos también se producen por las características de las funciones de fase. Podemos ver lóbulos en el centro de la cámara ortográfica cuando la función presenta direccionalidad alta  $|g| = 0,6$ . En los rayos cercanos al centro de la cámara, la función de fase se vuelve más paralela a esos rayos (tanto en la parte frontal como la trasera), por lo que la mayor parte de la luz es reflejada en su dirección. Esto sucede en la parte frontal del frente (4) para la dispersión forward  $g = 0,6$ , y en la parte trasera (5) para la dispersión backward. La diferencia en intensidad se debe a que la extinción en la parte frontal es menor que en la parte trasera, y por tanto llega más radiancia de la dispersión forward.

El multiple scattering es en realidad el producto de aplicar muchos procesos de dispersión al single scattering, que depende tanto de la función de fase como de la dispersión y absorción del medio. Intentar obtener conclusiones sobre la gráfica de multiple scattering resulta mucho más complicado, ya que la luz ha sufrido difusión y, como hemos visto, los efectos de acumulación sobre el rayo y la extinción del medio afectan mucho a cómo visualizamos la radiancia. Esto

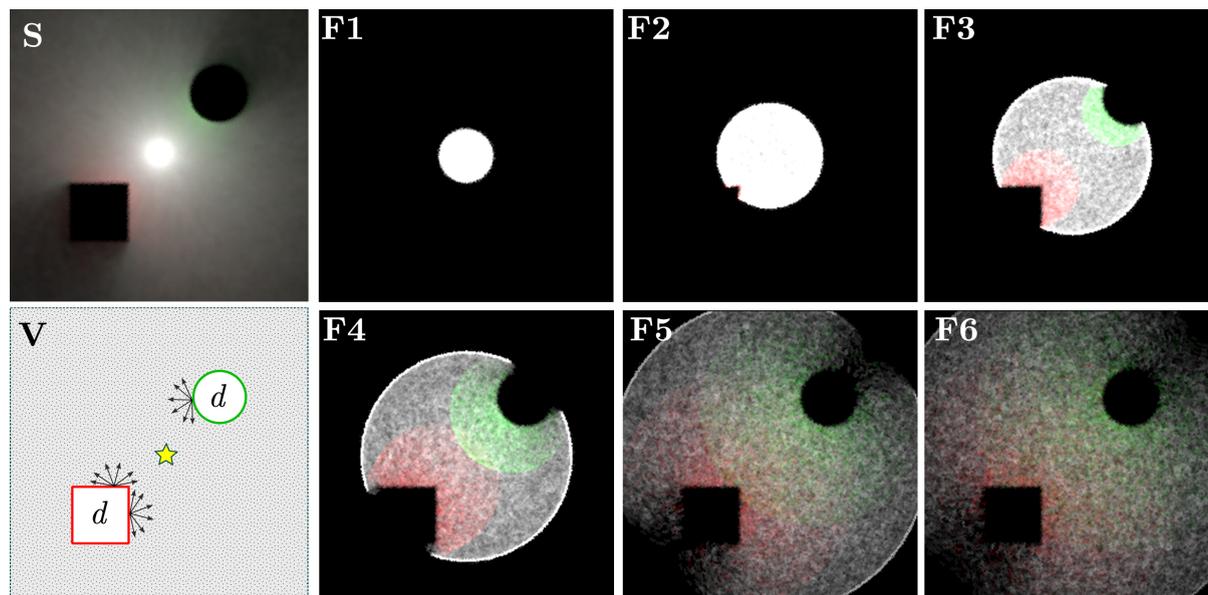


Figura 8.3: V es una representación esquemática de la escena, con una luz en el centro, un círculo verde y una caja roja, con *BRDFs* difusas. En el render estacionario S se puede ver el efecto de dispersión de color en el medio debido a la iluminación indirecta alrededor de la caja y el círculo. In the steady state image S we can see the color bleeding in the medium due to indirect illumination around the box and circle segments. Los fotogramas F1 a F6 muestran la propagación en estado transitorio y están sobreexpuestas para apreciar el multiple scattering.

demuestra la necesidad de un sistema de representación mas adecuado para eliminar estos efectos, explicado en la Sección 8.3.

### 8.3. Renderizado de medios participativos en 2D con un sensor de irradiancia

Si nuestro objetivo es analizar cómo se propaga la luz a través de un medio en 2D, una proyección ortográfica puede no ser el enfoque adecuado, como se ha demostrado en la Sección 8.2. En cambio, calcular la irradiancia en puntos diferenciales de un medio evita los efectos de extinción, acumulación y funciones de fase sobre el rayo, porque en realidad no hay ningún rayo. Esto lleva a un nuevo sistema de representación, donde se coloca una matriz de sensores en la escena y se computa la dispersión entrante en cada uno de ellos. Para ello, solo tenemos que muestrear puntos en la escena y hacer una estimación de radiancia **ignorando la función de fase**, lo que en realidad es una *estimación de irradiancia*. Esta representación simula una “proyección” como si estuviéramos viendo la escena 2D desde arriba, aunque en realidad esa proyección no existe en un mundo 2D. Nótese que este sistema de representación no permite ver los objetos de la escena, ya que una curva tiene anchura diferencial y es estadísticamente imposible muestrear un punto en esa curva.

La Figura 8.3 muestra una escena con una caja y círculos con *BRDFs* difusas. La Figura 8.4 muestra la misma escena variando las *BRDFs* por un círculo de vidrio naranja con  $\eta = 1,5$  y un espejo cian con reflexión especular perfecta. Esto nos permite ver los diferentes perfiles temporales de las cáusticas y la iluminación indirecta. Observando la Figura 8.3, debido a la

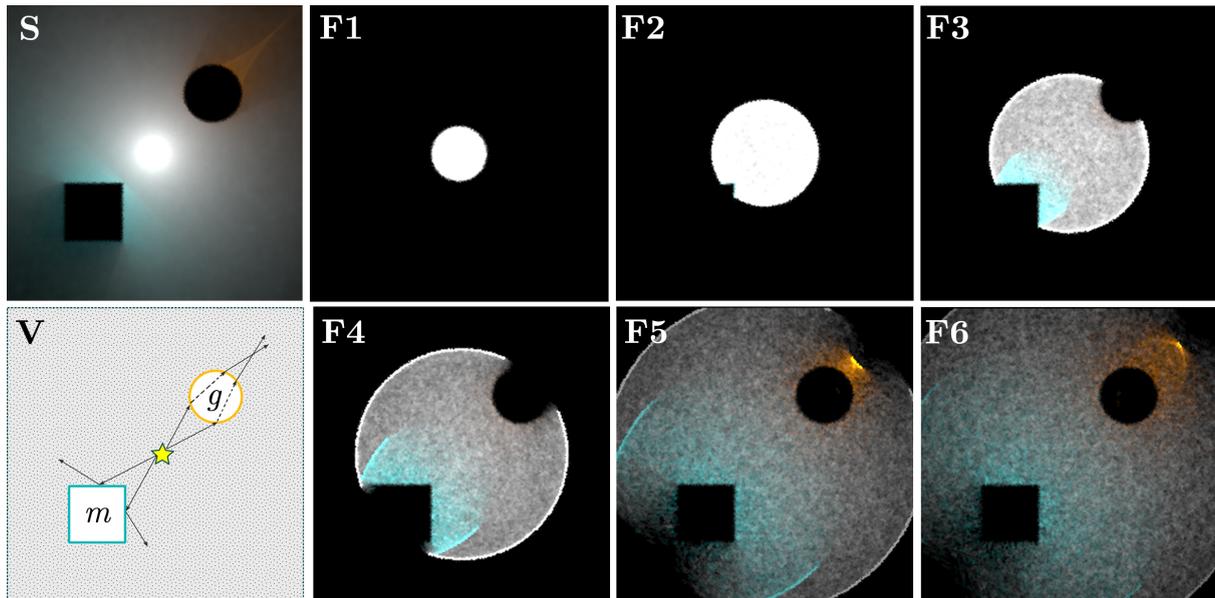


Figura 8.4: Usando la misma escena que en la Figura 8.3, la *BRDF* del círculo es ahora vidrio naranja con índice de refracción  $\eta = 1,5$ , y la *BRDF* de la caja es un espejo cian (reflexión especular perfecta). En la imagen estacionaria S podemos ver las cáusticas formadas por la naturaleza delta de las *BRDFs* transmisiva y especular alrededor del círculo y la caja. Los fotogramas F1 a F6 muestran la propagación de la luz en estado transitorio, y están sobreexpuestas para enfatizar el multiple scattering.

naturaleza difusa de las *BRDFs*, la direccionalidad de el frente de onda principal se pierde cuando rebota con la caja y el círculo (fotogramas F3 y F4) produciendo el efecto de dispersión de color. Si lo comparamos con los fotogramas F3 a F5 en la Figura 8.4, cuando el frente de onda rebota con la caja  $m$ , no se pierde la direccionalidad de la luz y el frente de onda prevalece debido a la direccionalidad de la *BRDF* especular. Toda la difusión alrededor del frente de onda es debida a los eventos dispersión en el medio, pero no a la *BRDF* de la caja. El vidrio naranja  $g$  no muestra ningún tipo de dispersión alrededor ya que toda la luz se transmite hacia el interior del objeto. Podemos observar también como el frente de onda lo rodea, y dado que el índice de refracción reduce la velocidad de la luz, hay que esperar un poco más hasta que vemos la cáustica salir del círculo. Mencionar que los render presentan ruido estructurados debido a problemas con el muestreo explicadas más adelante.

#### 8.4. Perfiles temporales gaussianos variando las funciones de fase

La direccionalidad de la luz presente en los medios participativos depende únicamente de la *BRDF* de las superficies. La función de fase de un medio redirige la luz en cada evento de dispersión con una intensidad distinta en cada dirección (Sección 3.2). En relación con el transporte de luz transitorio, D'Eon e Irving [DI11] proponen un modelo para renderizar materiales translúcidos, en el que mencionan:

[...] *Particles or photons leaving a point source and undergoing diffusion produce Gaussian distributions with a mean displacement proportional to  $\sqrt{tD}$ , where  $t$  is the time since emission and  $D$  is the diffusion coefficient.* [...]

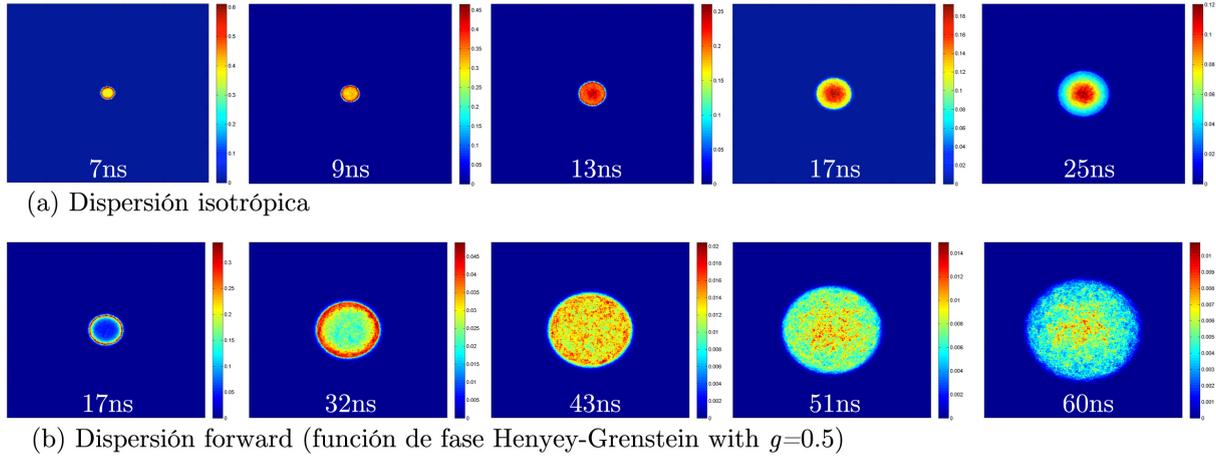


Figura 8.5: Perfiles temporales de irradiancia de una luz puntual dentro de un medio participativo, utilizando una función de fase isotrópica (a) y una función de fase *forward* ( $g=0.5$ ) (b). La escala de color está escalada en cada fotograma para mostrar los valores máximo y mínimo de la irradiancia.

Cuantizando estos perfiles gaussianos a lo largo del tiempo, proporcionan un modelo analítico de difusión. Sin embargo, hemos encontrado limitaciones respecto a este método. Usando el sistema de representación explicado en la Sección 8.3, observamos que a escalas temporales pequeñas este comportamiento no es cierto para los primeros órdenes de dispersión, donde la direccionalidad de la luz todavía prevalece. Dicho de otro modo, en los instantes iniciales—donde suceden los primeros eventos de dispersión—, a distancias muy cercanas a la luz los perfiles gaussianos no se han formado todavía ya que la luz no ha sufrido suficiente difusión debido a la dispersión. En vez de eso, todavía se pueden observar frentes de onda definidos, similares a los formados por el single scattering.

En la Figura 8.5 mostramos una escena con una luz puntual dentro de un medio, representando el **multiple scattering** en escala de color. Comparamos la direccionalidad del medio con funciones de fase isotrópicas ( $g = 0$ ) y forward ( $g=0.5$ ). En el caso de un medio con dispersión isotrópica (Figura 8.5a), se puede observar el comportamiento descrito arriba. Se preserva la direccionalidad de la luz en tiempos  $t \in [0, 24]$ ns y no se pueden observar gaussianas hasta  $t = 25$ ns. Los frentes de onda rojos que se ven en los primeros instantes son causa por de esta direccionalidad. Pero si observamos la configuración con dispersión forward (Figura 8.5b), la direccionalidad se preserva incluso más tiempo, y la gaussiana no se forma hasta  $t = 60$ ns cuando los valores de irradiancia son muy bajos. Esto es debido a que, incluso después de muchos eventos de dispersión, la mayor parte de la luz tiende a seguir su dirección original debido a la naturaleza direccional de la función de fase forward. Esto demuestra otra limitación sobre el modelo de Quantized Diffusion de D'Eon e Irving, añadiendo dependencia parcial en la función de fase.

## 8.5. Problemas de muestreo en renderizado transitorio

Cuando nos movemos a un mundo 3D, los problemas de muestreo de los algoritmos se acentúan ya que estamos en un espacio dimensional mayor. Además, añadir la dimensión temporal hace que las técnicas clásicas de muestreo utilizadas en los algoritmos estacionarios no funcionen muy bien en el transporte transitorio de luz. Esto es principalmente porque estas

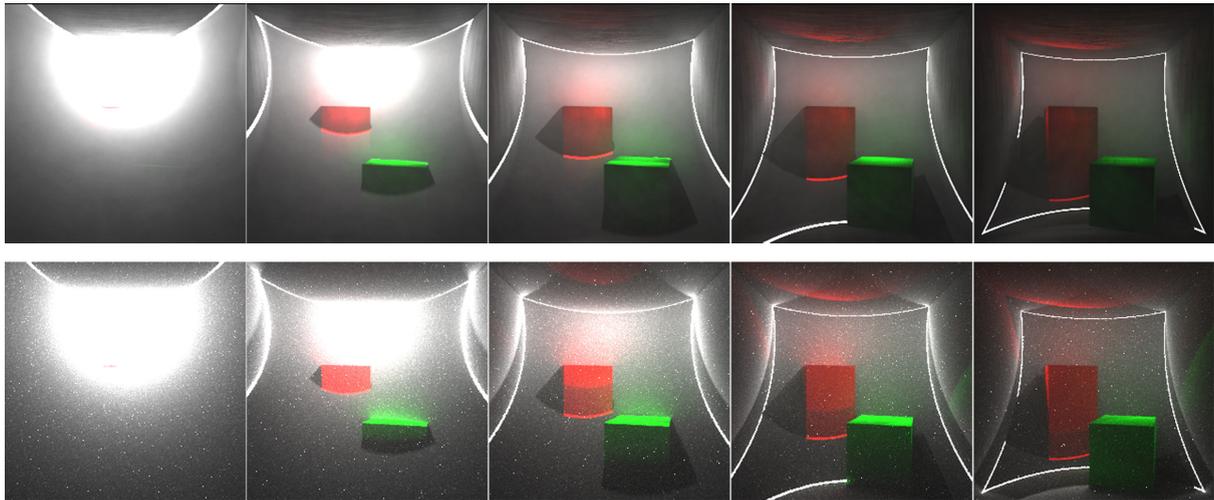


Figura 8.6: Mostramos la misma escena renderizada con dos algoritmos distintos para medios participativos. En la fila superior, la escena está renderizada con *Photon Beams* transitorios. En la escena de abajo, la escena está renderizada con una versión transitoria de *Volumetric Path Tracing*. El ruido en *Volumetric Path Tracing* se acentúa mucho en render transitorio, ya que estamos distribuyendo las muestras de los clásicos render estacionarios a lo largo del tiempo. Podemos ver las ventajas que tienen las estimaciones de radiancia de *Photon Beams* en la reducción de ruido—a pesar de introducir bias—y cómo esto ayuda al render transitorio de escenas con medios participativos.

técnicas están basadas en muestrear por importancia los valores de radiancia, es decir, tratan de obtener los caminos de luz que más contribuyen. Especialmente en medios participativos estos efectos se incrementan significativamente, dado que el tiempo es directamente proporcional a la longitud de los caminos de luz, y por lo tanto tiempos largos normalmente implican mayor extinción. De hecho, podemos comprobar este efecto reescribiendo la transmitancia como función del tiempo, dado que  $t = \frac{\eta_m}{c} s \Leftrightarrow s = \frac{c}{\eta_m} t$ , teniendo  $T_r = e^{-\sigma t \frac{c}{\eta_m}}$ . Sin embargo, en renderizado transitorio normalmente estamos interesados en observar altos órdenes de dispersión, que ocurren en tiempos mas largos.

En la Figura 8.6 se compara la misma escena renderizada con las versiones transitorias de dos algoritmos para medios participativos, *Photon Beams* y *Volumetric Path Tracing*. La versión estacionaria de la escena se puede ver en la Figura D.1a. La ausencia de estimaciones de radiancia de *Volumetric Path Tracing* produce varianza debido al proceso de muestreo por Monte Carlo (visto como ruido en la imagen). Este ruido se acentúa mucho más cuando nos movemos al dominio transitorio, dado que las muestras de radiancia se distribuyen en tiempo, y mayoritariamente en tiempos cortos debido a las técnicas de muestreo por importancia de la radiancia. En cambio, en *Photon Beams* este ruido se enmascara—como bias—debido a las estimaciones de radiancia que producen blur tanto en el dominio espacial como en el temporal.

## 8.6. Efectos de *camera warping* en 3D

Como se menciona en el Capítulo 7, tener en cuenta o no el tiempo de propagación desde un punto de la escena hasta el punto de vista, lleva a renders *camera-warped* o *camera-unwarped* respectivamente. Normalmente, los render *camera-unwarped* muestran de manera más intuitiva lo que realmente pasa en el mundo, ya que los no se alteran los caminos de luz con la distancia

al punto de vista. En la Figura 8.7 se muestran dos secuencias de una escena comparando estos efectos. La versión estacionaria de la escena se puede ver en la Figura D.1b. En la Figura 8.7a (*camera-unwarped*), vemos el frente de onda saliendo desde el instante  $t = 0\text{ns}$  (exactamente cuando se lanza el pulso de luz) y creciendo progresivamente, mientras que en la Figura 8.7b (*camera-warped*) tenemos que esperar hasta  $t = 10\text{ns}$ , y entonces vemos un intenso destello de luz en vez de un frente creciente. La razón de esto es explicada en el Capítulo 7. En la Figura 8.7a  $t \in [4, 6]\text{ns}$  podemos ver dos tipos de cáusticas como “gotas de luz” saliendo de la esfera de vidrio. La cáustica producida por el *single scattering*, empezando a salir de la esfera en  $t = 4\text{ns}$ , y la cáustica producida por el *multiple scattering* saliendo en  $t = 5\text{ns}$ . En  $t = 6\text{ns}$  podemos ver ambas cáusticas una después de la otra. Estas cáusticas apenas se ven en la Figura 8.7b, ya que las alteraciones producidas por *camera-warping* cambian la forma de los caminos de luz. Esto muestra que los renders *camera-unwarped* ayudan a entender mucho mejor cómo se propaga la luz en la escena.

## 8.7. Emisión delta vs. emisión continua

Todos los renders mostrados hasta ahora representan el comportamiento de un pulso de luz saliendo de la fuente de luz, lo que equivale a una emisión delta de Dirac. Esta representación tiene utilidad para ver cómo viaja y se dispersa la luz a través de una escena. Pero para ver cómo se propaga la luz en una escena hasta que se forma una imagen estacionaria típica, necesitamos una luz que emita continuamente, lo que equivale a una emisión con función escalón de Heaviside. Podemos imaginar este enfoque como si estuviéramos tomando una imagen con una cámara tradicional y pudieramos ver toda la exposición durante el tiempo que está abierta la apertura en un video en *ultra slow-motion*. La Figura 8.8 muestra dos secuencias comparando la misma escena con una emisión delta y una continua. La versión estacionaria de la escena se puede ver en la Figura D.1c.

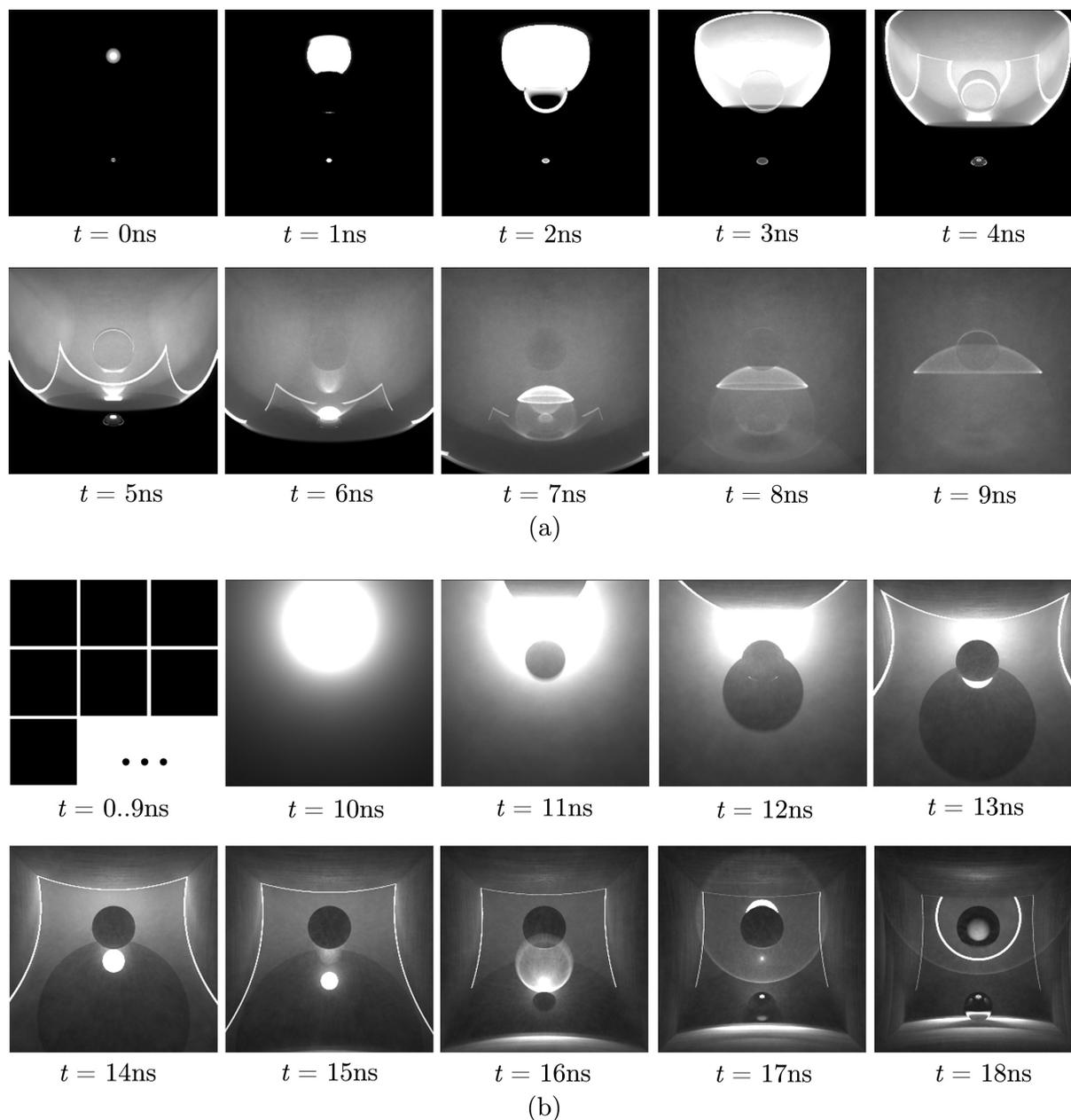
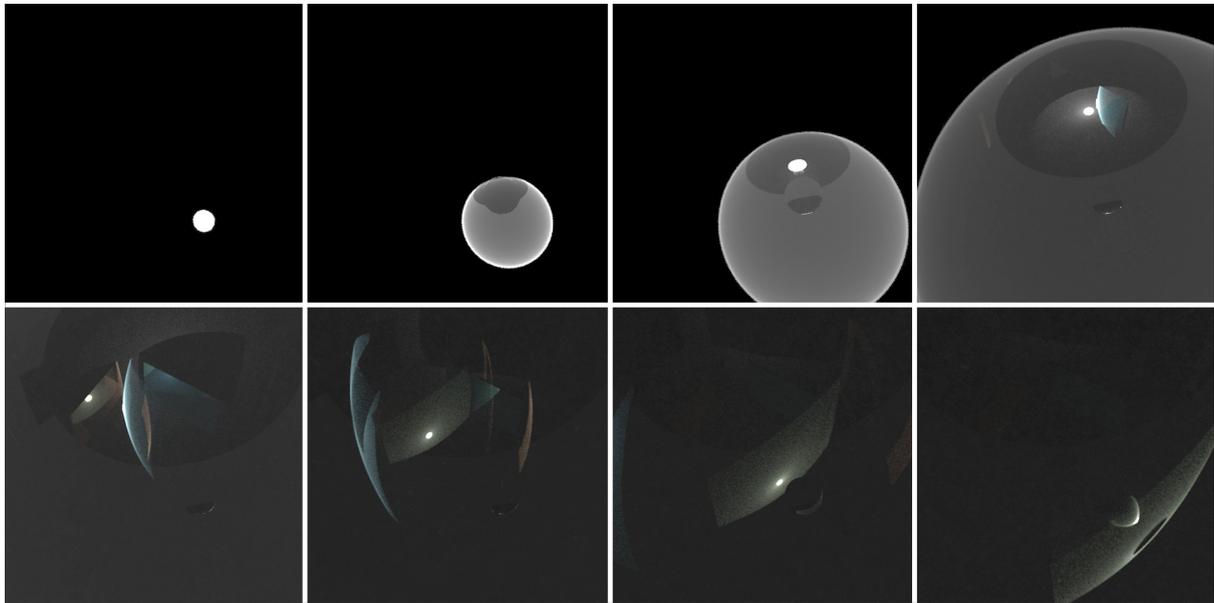
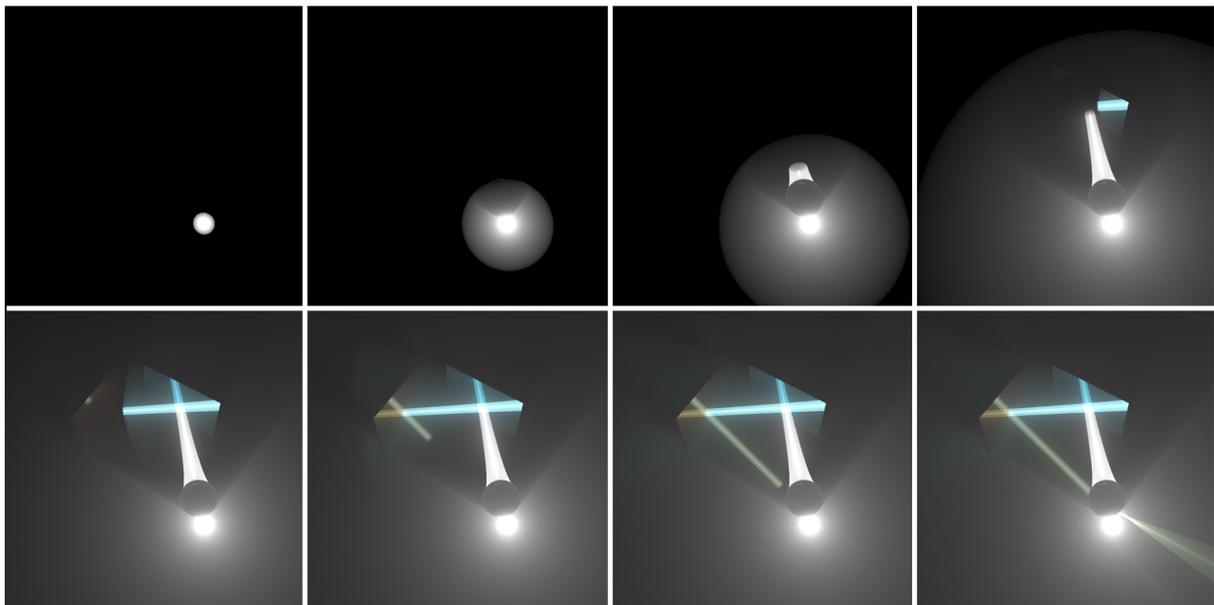


Figura 8.7: Escena con una una Cornell Box, una esfera de vidrio arriba y una esfera perfectamente especular debajo. La secuencia (a) está *camera-unwarped*, i.e. sin tener en cuenta el tiempo entre la luz y el punto de vista, por lo que vemos el frente de onda saliendo progresivamente de la luz. La secuencia (b) está *camera-warped*, teniendo en cuenta esta distancia, por lo que la forma del frente de onda está alterada por el tiempo que le cuesta a la onda recorrer esta distancia, viendo la escena como si hiciéramos un barrido desde el punto de vista hacia las partes más lejanas de la escena. See puede apreciar mejor observando los frentes de onda en las paredes.



(a) Emisión delta de Dirac



(b) Emisión escalón de Heaviside

Figura 8.8: Comparación entre emisión delta de Dirac (a) y emisión escalón de Heaviside (b). (a) está sobreexpuesta para acentuar el multiple scattering. La emisión delta de Dirac nos permite ver cómo un pulso de luz atraviesa y se dispersa por la escena, mientras que la emisión escalón de Heaviside muestra cómo la luz se emite continuamente hasta que alcanza cada punto de la escena, como si tomáramos una fotografía en *ultra slow-motion*.

## Capítulo 9

# Conclusiones

Con este proyecto se ha mostrado que gracias a la inclusión de la dimensión temporal en los modelos físicos de transporte de luz—tanto en el vacío como en los medios participativos—se puede encontrar información de gran utilidad sobre la propagación de la luz, hasta ahora descartada por los modelos clásicos de transporte de luz estacionario utilizados en render. La simulación del transporte de luz mediante un modelo directo nos permite variar parámetros concretos del modelo físico y observar cambios en los perfiles temporales de la propagación de la luz debidos a esas variaciones. Este aspecto unido a la simulación del transporte de luz transitorio en 2D, muestra que la reducción de dimensionalidad es una estrategia útil a la hora de analizar y entender el transporte de luz, especialmente al incluir la dimensión temporal en el problema, obteniendo resultados con conclusiones igualmente extrapolables a 3D. Se ha observado también que esta reducción de dimensionalidad mejora significativamente la eficiencia en el render, necesitando mucho menos tiempo para obtener convergencia en los resultados.

Los resultados obtenidos muestran ciertas limitaciones sobre algoritmos para medios participativos ya existentes, y muestran que las técnicas habituales de muestreo por importancia de la radiancia en los algoritmos de render estacionario no son apropiadas cuando incluimos la dimensión temporal. Los resultados obtenidos con la implementación de *Photon Beams* y su adaptación al dominio temporal demuestran que gracias a las estimaciones de densidad de los algoritmos basados en fotones, se reduce la varianza de las habituales técnicas de muestreo por importancia a la hora de representar escenas con transporte de luz transitorio. Finalmente se ha mostrado que los distintos métodos de representación afectan a la manera de entender el transporte de luz de una escena, especialmente al incrementar la dimensionalidad del problema.

### 9.1. Trabajo futuro

El análisis de este proyecto se ha hecho sobre cambios en las intensidades, observando perfiles temporales de la radiancia de la escena. Sin embargo, cambios temporales en otros tipos de dominios como el de las frecuencias también han sido analizados [WWB<sup>+</sup>12], demostrando la existencia de transferencia de información en el espacio de las frecuencias entre los diferentes dominios espaciales, angulares y temporales. Sin embargo, este estudio no incluye interacciones luz-materia, por lo que un posible trabajo futuro sería extender este análisis para incluir este tipo de interacciones.

Por otro lado, en este proyecto se han realizado muchas simplificaciones sobre las características de los medios. Aspectos como dispersión dependiente de la longitud de onda, medios con índices de refracción variables, o la utilización de modelos complejos de *BRDFs* producen efectos apreciables e interesantes a escalas temporales pequeñas, por lo que la inclusión de estos aspectos en la simulación de transporte de luz transitorio es un aspecto interesante para desarrollar en un futuro.

Además, el blur temporal aplicado en la adaptación temporal de *Photon Beams* incrementa el bias en los resultados, por lo que se requiere de un análisis adecuado con el fin de minimizarlo o al menos dar una estimación fiable del error introducido. Por último, los inconvenientes encontrados en las técnicas habituales de muestreo por importancia al utilizarlas en algoritmos transitorios muestran la necesidad de nuevas técnicas de muestreo que consigan mejorar la convergencia al renderizar transporte de luz transitorio.

## 9.2. Desarrollo de proyecto

Este proyecto se ha realizado dentro del marco de un proyecto de investigación en el Graphics and Imaging Lab de la Universidad de Zaragoza, y en colaboración con Wojciech Jarosz, investigador jefe en el grupo de *rendering* de Disney Research, Zürich. La duración del proyecto ha sido de once meses, desde el 20 de diciembre de 2012 hasta el 20 de noviembre de 2013. En la Figura 9.1 se puede observar un calendario con la evolución del proyecto a lo largo de este tiempo.

Durante todo el proyecto se han realizado reuniones regulares dentro del laboratorio. Desde la incorporación de Wojciech Jarosz al equipo de trabajo, el contacto por correo electrónico ha sido constante con el fin de resolver dudas y comentar detalles. Además se han realizado reuniones por videoconferencia de manera puntual durante los primeros meses, y a partir del 19 de agosto pasaron a ser semanales. En el calendario del proyecto se incluye la organización y celebración del Eurographics Symposium on Rendering (EGSR), evento al que acudieron investigadores de todo el mundo relacionados con campo el render y con los gráficos por ordenador en general. Se aprovechó la asistencia de Wojciech Jarosz al evento como *invited speaker* para realizar una reunión *in situ*. Además de la celebración del EGSR, se han recibido visitas al laboratorio de investigadores internacionales relacionados con los gráficos por ordenador, incluyendo Bill Freeman del MIT o Aaron Hertzmann de Adobe Research, a los que se les explicó la naturaleza de este proyecto y del proyecto de investigación al que va asociado.

Los resultados obtenidos con este proyecto serán sometidos a la conferencia mundial de gráficos por ordenador SIGGRAPH 2014 que publica en la revista ACM Transactions on Graphics (JCR, 1.º tercil, IF: 3.361). Adicionalmente, resultados previos obtenidos durante el desarrollo del proyecto fueron presentados en el *Congreso Español de Informática Gráfica 2013* (Anexo B).

## 9.3. Conclusión personal

Este ha sido mi primer contacto con el mundo de la investigación, y la experiencia está siendo muy positiva. Es una gran oportunidad poder trabajar dentro en un grupo de investigación con tanta repercusión internacional, y poder participar en un proyecto del mayor alcance posible en el mundo de los gráficos por ordenador. Esta experiencia durante los últimos meses ha supuesto

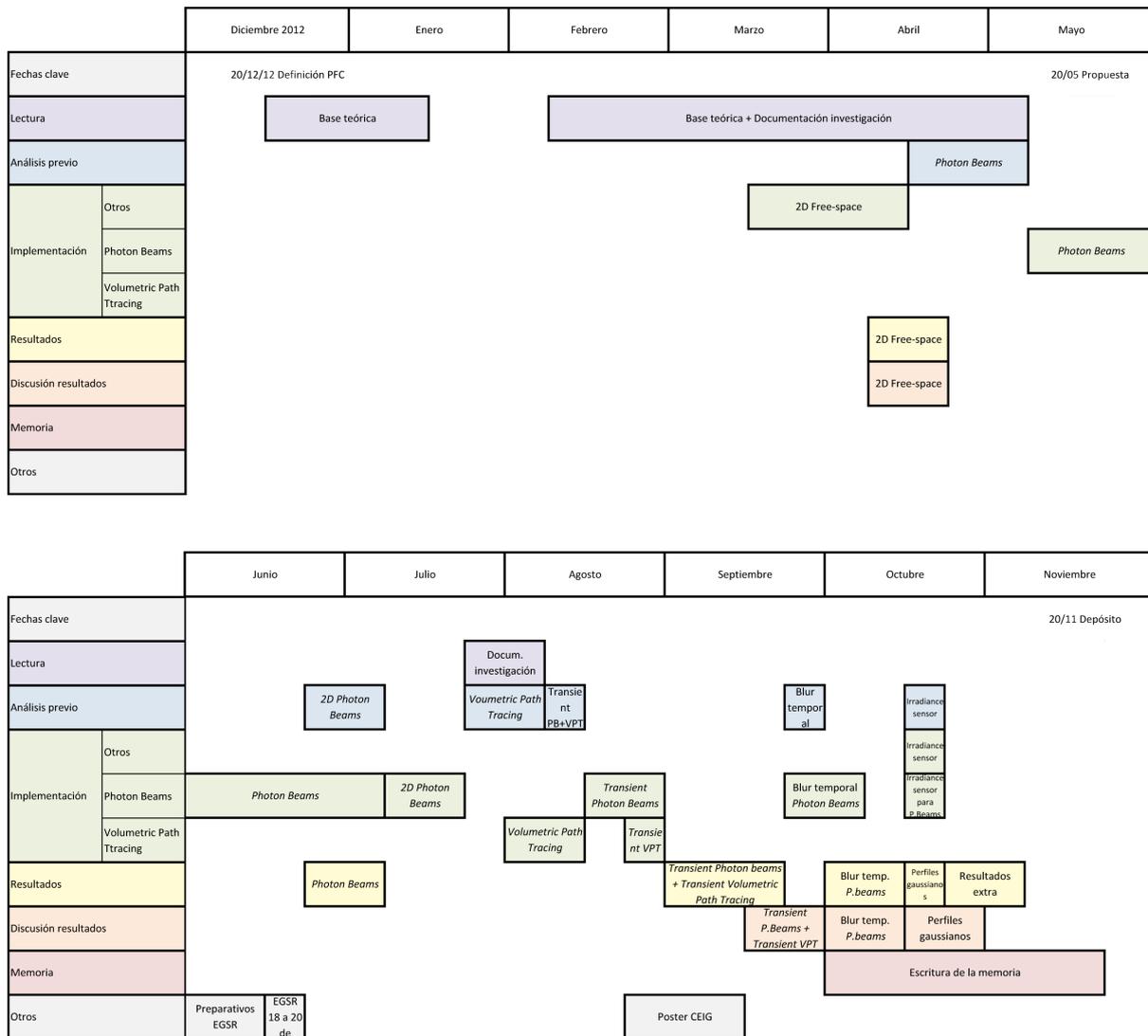


Figura 9.1: Evolución temporal del proyecto desde su comienzo, 20 de diciembre de 2012, hasta su finalización, 20 de noviembre de 2013.

un reto personal en muchos aspectos, y ahora mismo tengo la impresión de que en su momento tomé la decisión más correcta posible eligiendo trabajar con este grupo. La relación tanto con Adrián y Diego como con el resto de miembros del laboratorio ha sido genial, en el grupo se respira muy buen rollo y cada día se aprende algo nuevo. Su pasión por la investigación y las grandes oportunidades que ofrece el grupo han hecho que me decida a continuar en el mundo académico con vistas a hacer el doctorado.



# Bibliografía

- [Arv93] ARVO J. Transfer equations in global illumination. In *Global Illumination, SIGGRAPH'93 Course Notes*, 1993.
- [BSS<sup>+</sup>13] BELCOUR L., SOLER C., SUBR K., HOLZSCHUCH N., AND DURAND F. 5d covariance tracing for efficient defocus and motion blur. *ACM Trans. Graph.*, 32(3), 2013.
- [Cha60] CHANDRASEKHAR S. *Radiative transfer*. Courier Dover Publications, 1960.
- [Dav06] DAVIS A. Effective propagation kernels in structured media with broad spatial correlations, illustration with large-scale transport of solar photons through cloudy atmospheres. In Graziani F., editor, *Computational Methods in Transport*, volume 48 of *Lecture Notes in Computational Science and Engineering*, pages 85–140. Springer Berlin Heidelberg, 2006.
- [DBB06] DUTRÈ P., BALA K., AND BEKAERT P. *Advanced Global Illumination*. AK Peters, 2006.
- [DHS<sup>+</sup>05] DURAND F., HOLZSCHUCH N., SOLER C., CHAN E., AND SILLION F. X. A frequency analysis of light transport. *ACM Trans. Graph.*, 24(3), 2005.
- [DI11] D'EON E. AND IRVING G. A quantized-diffusion model for rendering translucent materials. *ACM Trans. Graph.*, 30(4), 2011.
- [GJJD09] GUTIERREZ D., JENSEN H. W., JAROSZ W., AND DONNER C. Scattering. In *ACM SIGGRAPH ASIA 2009 Courses*, 2009.
- [GMAS05] GUTIERREZ D., MUNOZ A., ANSON O., AND SERON F. J. Non-linear volume photon mapping. In *Eurographics Symposium on Rendering '05*, 2005.
- [GTGB84] GORAL C. M., TORRANCE K. E., GREENBERG D. P., AND BATTAILE B. Modeling the interaction of light between diffuse surfaces. *SIGGRAPH Comput. Graph.*, 18(3), 1984.
- [Hav04] HAVERKORT H. *Results on Geometric Networks and Data Structures*. ASCI dissertation series. 2004.
- [Hei12] HEIDE F. Waveform inversion for lambertian objects using transient imaging. Technical report, University of British Columbia, 2012.
- [HG41] HENYEV L. G. AND GREENSTEIN J. L. Diffuse radiation in the galaxy. *Astrophysical Journal*, 93:70–83, 1941.
- [Jar12] JARABO A. Femto-photography: Visualizing light in motion, 2012.

- [Jen01] JENSEN H. W. *Realistic Image Synthesis Using Photon Mapping*. AK Peters, 2001.
- [JNSJ11] JAROSZ W., NOWROUZEZAHRAI D., SADEGHI I., AND JENSEN H. W. A comprehensive theory of volumetric radiance estimation using photon points and beams. *ACM Trans. Graph.*, 30(1):5:1–5:19, February 2011.
- [JSKJ12] JAROSZ W., SCHÖNEFELD V., KOBELT L., AND JENSEN H. W. Theory, analysis and applications of 2d global illumination. *ACM Trans. Graph.*, 31(5), 2012.
- [JZJ08] JAROSZ W., ZWICKER M., AND JENSEN H. W. The beam radiance estimate for volumetric photon mapping. *Computer Graphics Forum (Proceedings of Eurographics 2008)*, 27(2):557–566, April 2008.
- [Kaj86] KAJIYA J. The rendering equation. In *SIGGRAPH '86*, 1986.
- [KGKCh13] KŘIVÁNEK J., GEORGIEV I., KAPLANYAN A. S., AND CAÑADA J. Recent advances in light transport simulation: theory & practice. In *ACM SIGGRAPH 2013 Courses*, 2013.
- [KK09] KELLER M. AND KOLB A. Real-time simulation of time-of-flight sensors. *Simulation Modelling Practice and Theory*, 17(5), 2009.
- [KOKP07] KELLER M., ORTHMANN J., KOLB A., AND PETERS V. A simulation framework for time-of-flight sensors. In *International Symposium on Signals, Circuits and Systems 2007*, 2007.
- [LALD12] LEHTINEN J., AILA T., LAINE S., AND DURAND F. Reconstructing the indirect light field for global illumination. *ACM Trans. Graph.*, 31(4), 2012.
- [LW93] LAFORTUNE E. P. AND WILLEMS Y. D. Bi-directional path tracing. In *Computer Graphics '93*, 1993.
- [MWRD13] MEHTA S. U., WANG B., RAMAMOORTHY R., AND DURAND F. Axis-aligned filtering for interactive physically-based diffuse indirect lighting. *ACM Trans. Graph.*, 32(4), 2013.
- [NZV<sup>+</sup>11] NAIK N., ZHAO S., VELTEN A., RASKAR R., AND BALA K. Single view reflectance capture using multiplexed scattering and time-of-flight imaging. *ACM Trans. Graph.*, 30, 2011.
- [RMB07] RAMAMOORTHY R., MAHAJAN D., AND BELHUMEUR P. A first-order analysis of lighting, shading, and shadows. *ACM Trans. Graph.*, 26(1), 2007.
- [SG86] SILVERMAN B. W. AND GREEN P. J. *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, London, 1986.
- [SML<sup>+</sup>12] SADEGHI I., MUÑOZ A., LAVEN P., JAROSZ W., SERON F., GUTIERREZ D., AND JENSEN H. W. Physically-based simulation of rainbows. *ACM Trans. Graph.*, 31(1), 2012.
- [SSD08] SMITH A., SKORUPSKI J., AND DAVIS J. Transient rendering. Technical Report UCSC-SOE-08-26, School of Engineering, University of California, Santa Cruz, February 2008.

- [VG97] VEACH E. AND GUIBAS L. J. Metropolis light transport. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 65–76, 1997.
- [VWG<sup>+</sup>12] VELTEN A., WILLWACHER T., GUPTA O., VEERARAGHAVAN A., BAWENDI M. G., AND RASKAR R. Recovering three-dimensional shape around a corner using ultrafast time-of-flight imaging. *Nature Communications*, (3), 2012.
- [VWJ<sup>+</sup>12] VELTEN A., WU D., JARABO A., MASIA B., BARSİ C., LAWSON E., JOSHI C., GUTIERREZ D., BAWENDI M. G., AND RASKAR R. Relativistic ultrafast rendering using time-of-flight imaging. In *ACM SIGGRAPH 2012 Talks*, 2012.
- [VWJ<sup>+</sup>13] VELTEN A., WU D., JARABO A., MASIA B., BARSİ C., JOSHI C., LAWSON E., BAWENDI M., GUTIERREZ D., AND RASKAR R. Femto-photography: Capturing and visualizing the propagation of light. *ACM Trans. Graph.*, 32(4), 2013.
- [Whi79] WHITTED T. An improved illumination model for shaded display. *SIGGRAPH Comput. Graph.*, 13(2), 1979.
- [WWB<sup>+</sup>12] WU D., WETZSTEIN G., BARSİ C., WILLWACHER T., O'TOOLE M., NAIK N., DAI Q., KUTULAKOS K., AND RASKAR R. Frequency Analysis of Transient Light Transport with Applications in Bare Sensor Imaging. In *European Conference on Computer Vision 2012*, 2012.



# Anexo A

## Contenido del CD

En el CD adjunto a esta memoria se incluye una versión digital de la memoria, el póster presentado en el CEIG (Anexo B), los dos artículos [JNSJ11, JSKJ12] en los que principalmente se basa este proyecto, y los vídeos e imágenes correspondientes a los resultados explicados en el Capítulo 8 y resultados adicionales. La estructura de carpetas con la sección a la que corresponden es la siguiente:

```
/
├── Bibliografía
├── Memoria
├── Poster CEIG 2013 .....Anexo B
├── Resultados
│   ├── 2D
│   │   ├── FreeSpace ..... Sección 8.1
│   │   ├── GaussianProfiles ..... Sección 8.4
│   │   └── IrradianceSensor ..... Sección 8.3
│   └── 3D
│       ├── Armadillo ..... Anexo D
│       ├── CornellBlocks .....Sección 8.5
│       ├── CornellCaustics ..... Sección 8.6
│       ├── Mirrors ..... Sección 8.7
│       └── StanfordDragon .....Anexo D
```

La bibliografía contiene los artículos [JNSJ11, JSKJ12] debido a que son en los que se principalmente se basa este proyecto.



## Anexo B

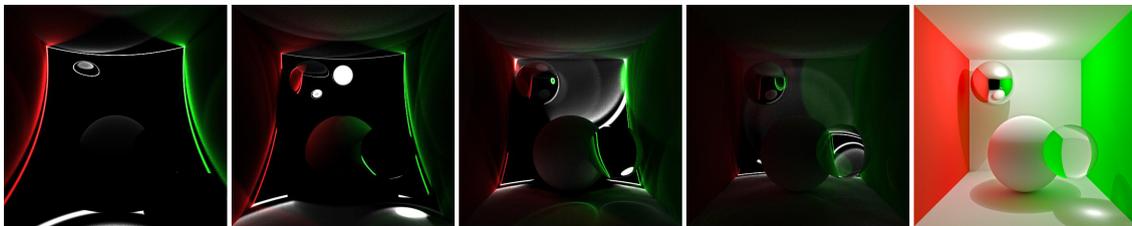
# Póster en el CEIG 2013

Resultados previos y conclusiones este proyecto fueron presentados en el Congreso Español de Informática Gráfica (CEIG) celebrado del 17 al 20 de septiembre de 2013 en un póster con el título “Rendering Light Transport in Transient-State”. En este anexo se incluyen tanto el *Abstract* como el propio póster.

CEIG - Spanish Computer Graphics Conference (2013), pp. 1–1  
 M. Carmen Juan and Diego Borro (Editors)

## Rendering Light Transport in Transient-State

Poster XXXX



**Figure 1:** The first four images show selected frames of a time-resolved rendering. The rightmost image shows the classic view of the scene, with all light integrated on the sensor during the simulated exposure.

### Abstract

In this work we relax the common assumption in rendering of infinite speed of light. We generalize the well-known rendering equation and radiative transfer equation introducing the temporal domain, so they can model the behaviour of light in transient-state i.e. as it propagates through the scene. Based on this generalization, we modify several rendering techniques, both for free-space and participating media, to address the temporal domain in the light transport. This allows rendering synthetic scenes where light propagates through the scene, in a similar fashion as the scenes captured with recent ultra-fast imaging techniques.

A common assumption in the fields of Computer Graphics and Computer Vision is that the light propagates at infinite speed. This is a reasonable assumption, since the usual exposure times in these fields are several orders the magnitude higher than the speed of light. Recent ultra-fast imaging techniques [VWJ\*13] have open the possibility of capturing how the light propagates and interacts with the matter, by capturing at effective exposures in the order of 2 ps. This results in stunning super-slow motion video.

The goal in this work is rendering synthetic animations depicting light propagating through a scene, similar to the scenes captured using ultra-fast imaging. This means that the common assumption of infinite speed of light is relaxed, and thus accounting *time* as a new dimension in the simulation of light transport. We introduce the temporal dimension in the well-established *rendering equation* [Kaj86] and *radiative transfer equation* (RTE) [Cha60], to model the light reflection on surfaces and the light transport in participating media respectively.

We apply these extended light transport models to a set of global illumination algorithms, originally designed for light transport in steady-state, modifying them to include the temporal domain in the simulation. We choose both biased and

unbiased methods to solve the light transport integrands, to example the classical rendering trade-off *variance vs bias*.

The rendered sequences allows us to visualize in synthetic scenes how light propagates through the scene, showing complex effects such as multiple interreflections, caustics, or delay due to transmission through a glass sphere, as exemplified in Figure 1. We hope that the development of transient rendering systems will help the development of new applications based on the novel ultra-fast imaging techniques by giving controlled synthetic scenerios to evaluate and test these new methods, and also to understand in more depth the behaviour of light, which might lead to the development of improved forward and inverse rendering methods.

### References

- [Cha60] CHANDRASEKHAR S.: *Radiative Transfer*. Dover Publications Inc, 1960. 1
- [Kaj86] KAJIYA J.: The rendering equation. In *SIGGRAPH '86* (1986). 1
- [VWJ\*13] VELTEN A., WU D., JARABO A., MASIA B., BARSIC., JOSHI C., LAWSON E., BAWENDI M., GUTIERREZ D., RASKAR R.: Femto-photography: Capturing and visualizing the propagation of light. *ACM Trans. Graph.* 32, 4 (2013). 1

# Rendering Light Transport in Transient-State



Julio Marco<sup>1</sup>

Adrian Jarabo<sup>1</sup>

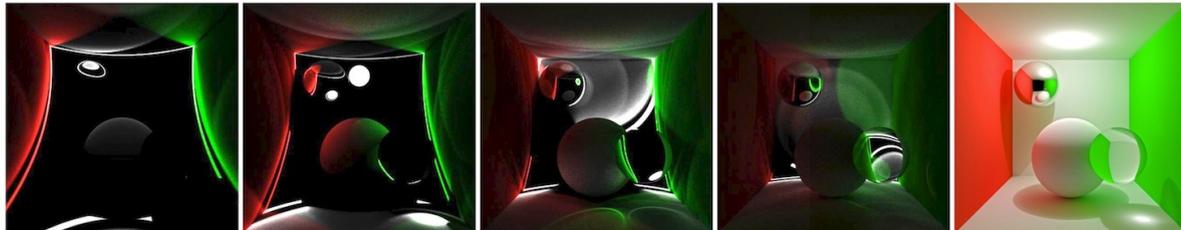
Wojciech Jarosz<sup>2</sup>

Diego Gutierrez<sup>1</sup>

<sup>1</sup>Universidad de Zaragoza

<sup>2</sup>Disney Research Zurich

In this work we relax the common assumption in rendering of infinite speed of light. We generalize the well-known rendering equation and radiative transfer equation introducing the temporal domain, so they can model the behaviour of light in transient-state i.e. as it propagates through the scene. Based on this generalization, we modify several rendering techniques, both for free-space and participating media, to address the temporal domain in the light transport. This allows rendering synthetic scenes where light propagates through the scene, in a similar fashion as the scenes captured with recent ultra-fast imaging techniques.



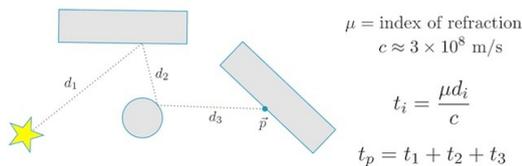
Frame sequence showing the behaviour of light in transient-state propagating through a free-space scene, and the classic steady-state result (rightmost image). Note how a secondary wavefront produced by the indirect reflection at the roof of the scene follows the primary direct wavefront. We can see also that the inclusion of reflective and refractive spheres adds additional light paths, producing, for example, that the reflection of the green wall in the mirror ball appears instants after the wall is illuminated. Additionally, it can be seen how the caustics produced by the lower-right ball appear, including caustics produced by the refraction of the light from green wall.

## Surfaces interreflections in free-space

The introduction of the temporal dimension in the rendering equation [Kaj86] allows us to model the behaviour of light along time as it reflects multiple times between the surfaces.

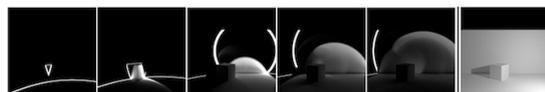
$$L(x, \omega_o, t) = L_o(x, \omega_o, t) + \int_{\Omega^+} L_i(x, \omega_i, t) \rho(x, \omega_i, \omega_o) (-\omega_i \cdot n) d\omega_i$$

Combining this approach with well-established algorithms that solve the steady-state light transport, such as path tracing, we can observe the multiple orders of surface scattering (i.e. interreflections) separated along time. Observing these interreflections allows us to understand how the light propagates through the scene, especially the behaviour of indirect illumination. The length of a path of light increases with the number of bounces, and unless the light is perfectly reflected on a specular surface, its intensity is reduced as well.



In steady state we assume light would travel through the scene and arrive to point  $p$  immediately. If we break this assumption, light actually takes  $t_i$  seconds traveling from the light source until it reaches point  $p$ . In order to render the time-resolved animations, we need to keep track on the total time since light is emitted until it arrives to the camera.

### Box scene



In this frame sequence we can see the direct light wavefront approaching the box. We can also dynamically observe the shadows produced by direct light behind the box (frame 3), and shadows produced by indirect interreflections (frames 3 to 5) that we cannot see in the classic steady-state rendered image (right).

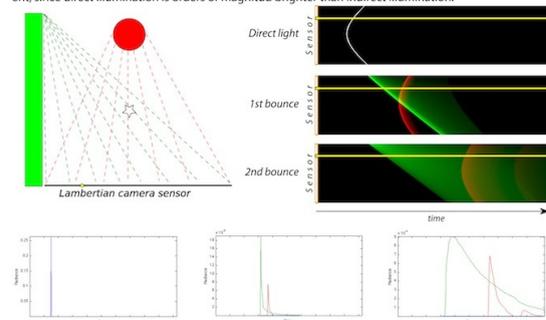
### Irregular surfaces



This bunny scene shows a different type of indirect light wavefront shapes (e.g. on the floor) due to the irregular shape of the bunny surface. We can also see in the third and fourth frame the red backlight on the bunny coming from the bounces of light in the left wall.

## Direct and indirect illumination temporal profiles

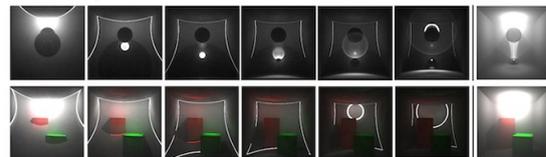
In order to understand the temporal behaviour of direct and indirect illumination, we reduce the spatial dimensionality of the scenes to 2D, which is a useful simplification since adding the temporal dimension to the problem increases its complexity. Using a lambertian camera sensor to capture the incoming irradiance at the floor in the scene depicted in the figure below. By disambiguating the radiance with respect to the number of bounces the light took to reach the sensor we can observe the different shape and frequency of the different illumination components. Note that exposure of the different bounces is different, since direct illumination is orders of magnitude brighter than indirect illumination.



In the radiance maps (which mimic the streak images from [Vel13]) we can distinguish white radiance from the direct illumination, green radiance coming from the wall, and red radiance coming from the ball. Radiance from the wall reaches the leftmost part of the sensor sooner, because the total length of the light paths are closer. Red radiance timings are symmetric due to the symmetry of the projected ball over the sensor. The yellow lines represent a temporal slice for a single point in the sensor. In the graphs we can see the temporal profile in the yellow point for (from left to right) the direct, first bounce and second bounce radiance intensity. Although the temporal profiles depend on the angular and spatial variations of geometry respect to the sensor, we can observe that higher levels of interrefection produce smoother temporal radiance profiles, since the range of paths length increases with the number of bounces.

## Transient light transport in participating media

Simulating light transport in participating media is a challenging task even in common steady-state rendering. However, scattering media is very common in real life, which makes simulate light transport in this type of media very interesting. Different algorithms provide solutions of different nature. While path tracing based algorithms converge to the exact solutions, they fail when representing complex light paths and take long to produce noise-free images. Photon-based algorithms, are good at handling complex light paths, and partially remove high-frequency noise, by introducing bias in the solution. When adding the temporal domain to these algorithms, all these algorithm-inherent properties are accentuated: since radiance samples are distributed along time, the noise produced by path tracing becomes even more evident when observing interreflections, which makes this kind of algorithms impractical. We therefore develop a transient version of Photon Beams [Jar11], to efficiently compute transient light transport in scenes with participating media.



In the top sequence we represent a cornell box with a glass and a mirror sphere, which create caustic light paths and allow us to observe their behaviour through the medium. The third frame shows caustics produced by both direct and scattered light. In the bottom sequence we can observe the evolution of shadows in the medium through the blocks and the scattered indirect light from the blocks.

## References

[Kaj86] Kajija J. The rendering equation. SIGGRAPH '86.  
 [Jar11] Jarosz W, Nowrouzezahrai D, Sadeghi L, and Jensen H. W. A comprehensive theory of volumetric radiance estimation using photon points and beams. ACM Trans. Graph 2011.  
 [Vel13] Velten A., Jarabo A., Masia B., Gutierrez D., Raskar R. et al. Femto-photography: capturing and visualizing the propagation of light. SIGGRAPH'13.

Contact: jmarcomurria@gmail.com  
 ajarabo@unizar.es



## Anexo C

# Detalles de Implementación

Este proyecto ha sido desarrollado sobre un **motor de render** implementado en C++. En la Figura C.1 se puede observar un diagrama de clases **simplificado** de la estructura del motor de render. En este diagrama se incluyen las clases más relevantes con atributos y métodos simplificados de forma que se puede identificar fácilmente la correspondencia con todos los elementos, propiedades y otros aspectos explicados en este documento.

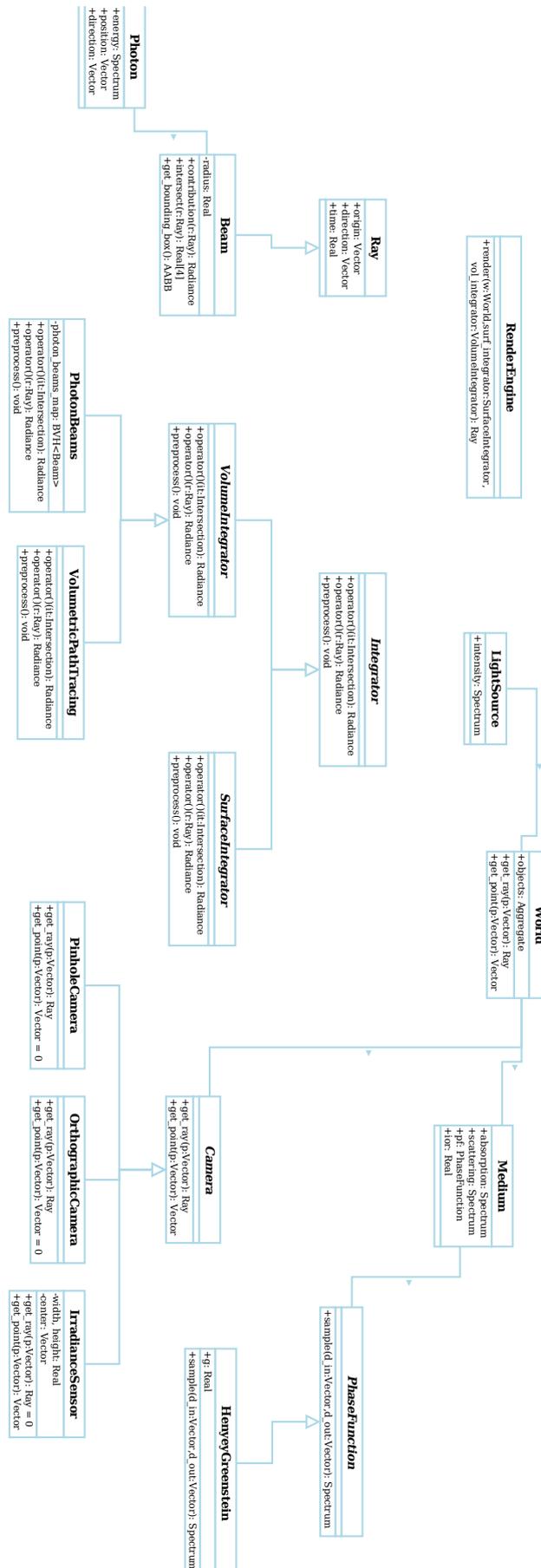
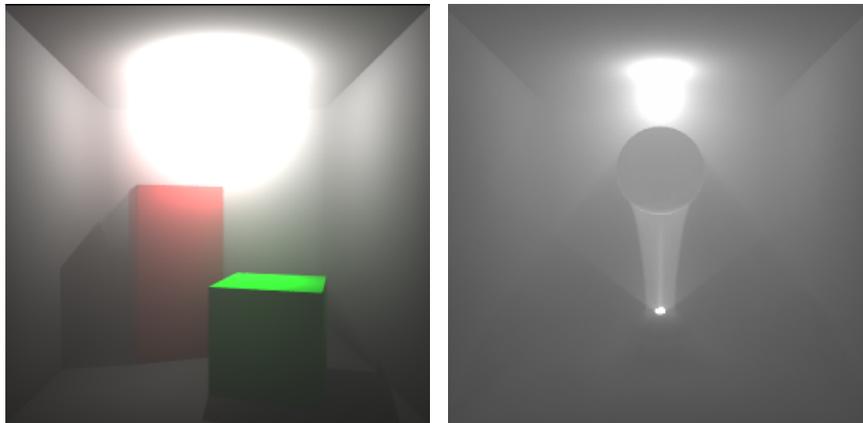


Figura C.1: Diagrama de clases simplificado del motor de render sobre el que se ha trabajado en este proyecto.

## Anexo D

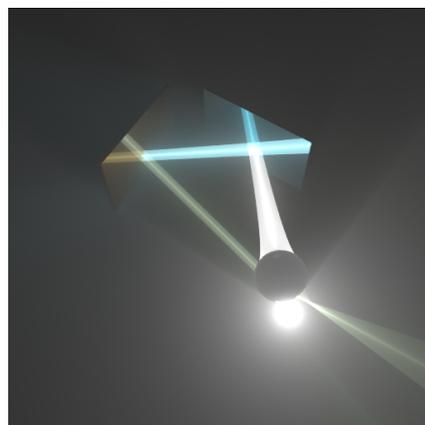
# Resultados Adicionales

En esta sección se muestran parte de los resultados en estado estacionario correspondientes a las secciones del Capítulo 8, y renders adicionales obtenidos durante el desarrollo del proyecto.



(a) Render estacionario de la secuencia explicada en la Sección 8.5.

(b) Render estacionario de la secuencia explicada en la Sección 8.6.



(c) Render estacionario de la secuencia explicada en la Sección 8.7.

Figura D.1: Renders estacionarios del Capítulo 8

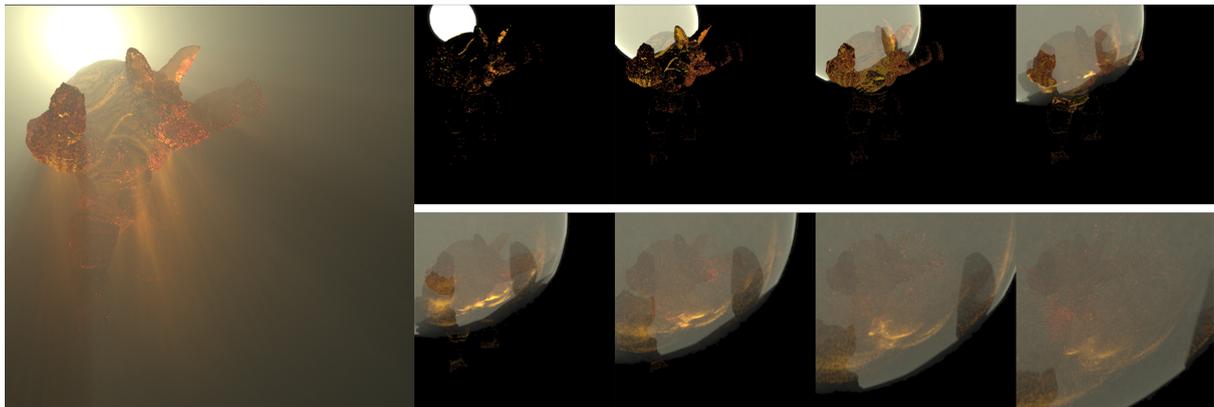


Figura D.2: Secuencia mostrando una emisión delta de luz en una escena con el modelo *armadillo de Stanford*.

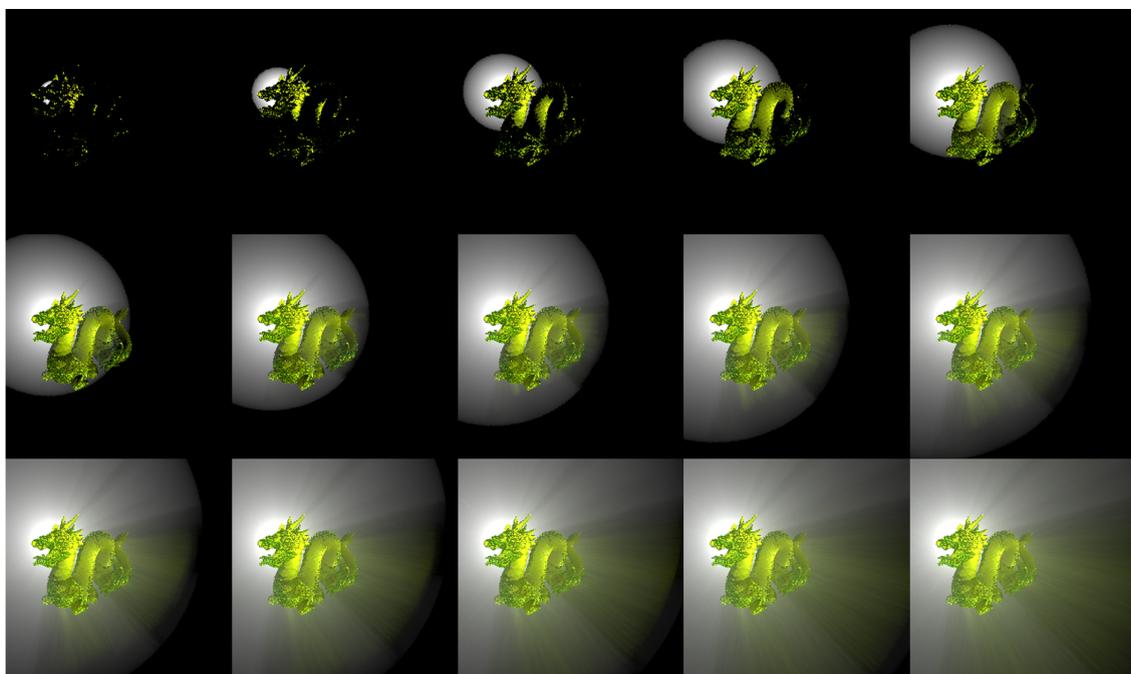


Figura D.3: Secuencia mostrando una emisión continua de luz en una escena con el modelo *dragón de Stanford*.