



UNIVERSIDAD
DE GRANADA

Los fundamentos de Ray Tracing

Doble grado en ingeniería informática y matemáticas

asmilex.github.io/Raytracing

Presentado por: Andrés Millán Muñoz,

Tutorizado por: Carlos Ureña Almagro, María del Carmen Segovia García

Escuela Técnica Superior de Ingenierías Informática y de
Telecomunicación
Facultad de Ciencias

March 29, 2022



Contents

Abstract	1
Dedicatoria	3
1 Introducción	4
1.1 ¿Qué es ray tracing?	5
1.2 Vale, ¿y qué vamos a hacer entonces?	6
2 Las bases	7
2.1 Eligiendo direcciones	9
2.2 Intersecciones rayo - objeto	10
2.2.1 Superficies implícitas	10
2.2.2 Superficies paramétricas	12
2.2.3 Intersecciones con esferas	13
2.2.4 Intersecciones con triángulos	14
3 Integración de Monte Carlo	18
3.1 Repaso de probabilidad	18
3.1.1 Variables aleatorias discretas	19
3.1.2 Variables aleatorias continuas	21
3.1.3 Esperanza y varianza de una variable aleatoria	22
3.2 El estimador de Monte Carlo	25
3.3 Escogiendo puntos aleatorios	28
3.3.1 Método de la transformada inversa	28
3.3.2 Método del rechazo	29

4	Radiometría	31
4.1	Unidades básicas	31
4.1.1	Potencia	32
4.1.2	Irradiancia	32
4.1.3	Ángulos sólidos	33
4.1.4	Intensidad radiante	36
4.1.5	Radiancia	36
4.2	Integrales radiométricas	38
4.2.1	Una nueva expresión de la irradiancia y el flujo	38
4.3	Integrando sobre área	40
4.4	Dispersión de luz: las familias de funciones de distribución bidireccionales	41
4.4.1	La función de distribución de reflectancia bidireccional (BRDF) .	41
4.4.2	La función de distribución de transmitancia bidireccional (BTDF)	42
4.4.3	Juntando la BRDF y la BTDF	43
4.4.4	<i>Subsurface scattering</i>	43
5	<i>Importance sampling</i>	45
6	Metodología; o cómo se hizo este trabajo	46
6.1	Github	46
6.1.1	Github Actions	46
6.1.2	Github Projects	46
7	Anexo: Glosario de términos y expresiones	47
7.1	Notación	47
7.2	Radiometría	48
	Bibliografía	50

Abstract

Se procederá a analizar los algoritmos modernos de visualización 3D realista usando métodos de Monte-Carlo, y su implementación en hardware gráfico moderno (GPUs) específicamente diseñadas para aceleración de Ray-Tracing. Se diseñará e implementará un sistema software de síntesis de imágenes realistas por path tracing y muestreo directo de fuentes de luz, que haga uso del hardware gráfico, y se analizará su eficiencia en tiempo en relación a la calidad de las imágenes y en comparación con una implementación exclusivamente sobre CPU.

Se realizará una revisión bibliográfica de los métodos de Montecarlo que se aplican de manera habitual para la visualización de imágenes 3D. Se examinarán los puntos fuertes y débiles de cada una de las técnicas, con el objetivo de minimizar el error en la reconstrucción de la imagen sin que esto suponga un alto coste computacional. Se investigarán las soluciones propuestas para el futuro del área.

Translation. It'll be left as is until there's a definitive abstract

Dedicatoria

Aquí es donde me pongo ñoño

¡Parece que has llegado un poco pronto! Si lo has hecho voluntariamente, ¡muchas gracias! Este proyecto debería estar finalizado en verano de 2022.

Mientras tanto, actualizaré poco a poco el contenido. Si quieres ir comprobando los progresos, puedes visitar [Asmilex/Raytracing](#) en Github para ver el estado del desarrollo.

1 Introducción

Ser capaces de capturar un momento.

Desde siempre, este ha sido uno de los sueños de la humanidad. La capacidad de retener lo que ven nuestros ojos comenzó con simples pinturas ruprestres. Con el tiempo, el arte evolucionó, así como la capacidad de retratar nuestra percepción con mayor fidelidad.

A inicios del siglo XVIII, se caputaron las primeras imágenes con una cámara gracias a Nicéphore Niépce. Sería una imagen primitiva, claro; pero era funcional. Gracias a la compañía Kodak, la fotografía se extendió al consumidor rápidamente sobre 1890. Más tarde llegaría la fotografía digital, la cual simplificaría muchos de los problemas de las cámaras tradicionales.

Hablando de digital. Los ordenadores personales modernos nacieron unos años más tarde. Los usuarios eran capaces de mostrar imágenes en pantalla, que cambiaban bajo demanda. Y, entonces, nos hicimos una pregunta...

¿Podríamos **simular la vida real** para mostrarla en pantalla?

Como era de esperar, esto es complicado de lograr. Para conseguirlo, hemos necesitado crear abstracciones de conceptos que nos resultan naturales, como objetos, luces y seres vivos. “Cosas” que un ordenador no entiende, y sin embargo, para nosotros *funcionan*.

Así, nació la geometría, los puntos de luces, texturas, sombreados, y otros elementos de un escenario digital. Pero, por muchas abstracciones elegantes que tengamos, no nos basta. Necesitamos visualizarlas. Y como podemos imaginarnos, esto es un proceso costoso.

La **rasterización** es el proceso mediante el cual estos objetos tridimensionales se transforman en bidimensionales. Proyectando acordemente el entorno a una cámara, conseguimos colorear un pixel, de forma que represente lo que se ve en ese mundo.

TODO insertar imagen rasterización. NOTE ¿quizás debería extender un poco más esta parte? Parece que se queda algo coja la explicación.

Aunque esta técnica es bastante eficiente en términos de computación y ha evolucionado mucho, rápidamente saturamos sus posibilidades. Conceptos como *shadow maps*, *baked lightning*, o *reflection cubemaps* intentan solventar lo que no es posible con rasterización: preguntarnos *qué es lo que se encuentra alrededor nuestra*.

En parte, nos olvidamos de la intuitiva realidad, para centrarnos en aquello computacionalmente viable.

Y, entonces, en 1960 el trazado de rayos con una simple idea intuitiva.

1.1 ¿Qué es ray tracing?

En resumidas cuentas, *ray tracing* (o trazado de rayos en español), se basa en disparar fotones desde nuestras luces digitales y hacerlos rebotar en la escena.

De esta forma, simulamos cómo se comporta la luz. Al impactar en un objeto, sufre un cambio en su trayectoria. Este cambio origina nuevos rayos, que vuelven a dispersarse por la escena. Estos nuevos rayos dependerán de las propiedades del objeto con el que hayan impactado. Con el tiempo necesario, lo que veremos desde nuestra cámara será una representación fotorealista de lo que habita en ese universo.

Esta técnica, tan estúpidamente intuitiva, se ha hecho famosa por su simpleza y su elegancia. Pues claro que la respuesta a “¿Cómo simulamos fielmente una imagen en un ordenador?” es “Representando la luz de forma realista”.

Aunque, quizás intuitiva no sea la palabra. Podemos llamarla *natural*, eso sí. A fin de cuentas, fue a partir del siglo XVIII cuando empezamos a entender que podíamos capturar la luz. Nuestros antepasados tenían teorías, pero no podían explicar por qué *veíamos* el mundo.

Ahora sí que sabemos cómo funciona. Entendiendo el por qué lo hace nos permitirá programarlo. Y, resulta que funciona impresionantemente bien.

Atrás se quedan los *hacks* necesarios para rasterización. Los cubemaps no son esenciales para los reflejos, y no necesitamos cámaras virtuales para calcular sombras. Ray tracing permite simular fácilmente efectos como reflejos, refracción, desenfoque de movimiento, aberración cromática... Incluso fenómenos físicos propios de las partículas y las ondas.

Espera. Si tan bueno es, ¿por qué no lo usamos en todos lados?

Por desgracia, el elefante en la sala es el rendimiento. Como era de esperar, disparar rayos a diestro y siniestro es costoso. **Muy costoso.**

A diferencia del universo, nosotros no nos podemos permitir el lujo de usar fotones de tamaño infinitesimal y dispersiones casi infinitas. Nos pasaríamos una eternidad esperando. Y para ver una imagen en nuestra pantalla necesitaremos estar vivos, claro.

Debemos evitar la fuerza bruta. Dado que la idea es tan elegante, la respuesta no está en el “*qué*”, sino en el “*cómo*”. Si **disparamos y dispersamos rayos con cabeza** seremos capaces de obtener lo que buscamos en un tiempo razonable.

Hace unos años, al hablar de tiempo razonable, nos referiríamos a horas. Quizás días. Producir un *frame* podría suponer una cantidad de tiempo impensable para un ordenador de consumidor. Hoy en día también ocurre esto, claro está. Pero la tecnología evoluciona.

Podemos bajarlo a milisegundos.

Hemos entrado en la era del **real time ray tracing**.

1.2 Vale, ¿y qué vamos a hacer entonces?

TODO hablar de los objetivos del trabajo.

Wikipedia contributors (2022j), Wikipedia contributors (2022h), Wikipedia contributors (2022d), Wikipedia contributors (2022c), Caulfield (2020), Wikipedia contributors (2022b), “Rendering” (n.d.), Haines and Akenine-Möller (2019), p. 16

2 Las bases

Empecemos por definir lo que es un rayo.

Un rayo es una función $P(t) = O + tD$, donde O es el origen, D la dirección, y $t \in \mathbb{R}$. Podemos considerarlo una interpolación entre dos puntos en el espacio, donde t controla la posición en la que nos encontramos.

Por ejemplo, si $t = 0$, obtendremos el origen. Si $t = 1$, obtendremos el punto correspondiente a la dirección. Usando valores negativos vamos *hacia atrás*.

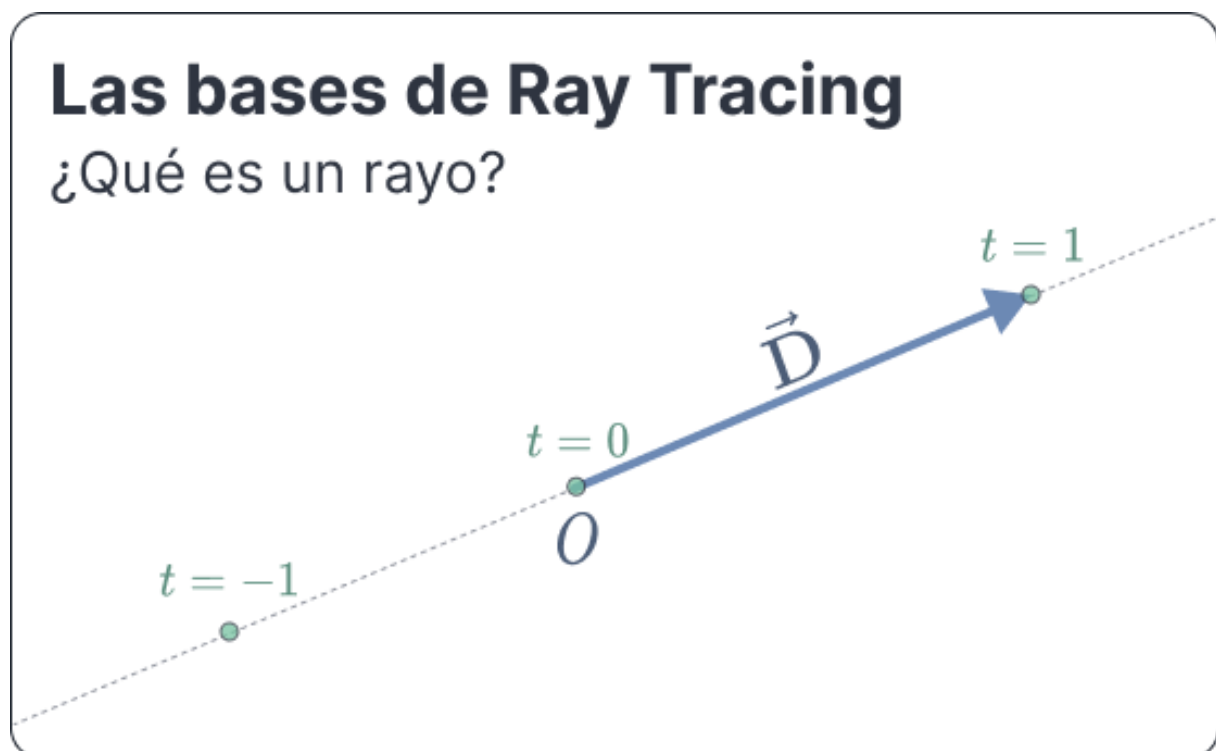


Figure 2.1: El parámetro t nos permite controlar los puntos del rayo

Dado que estos puntos estarán generalmente en \mathbb{R}^3 , podemos escribirlo como

$$P(t) = (O_x, O_y, O_z) + t(D_x, D_y, D_z)$$

Estos rayos los *dispararemos* a través de una cámara virtual, que estará enfocando a la escena. De esta forma, los haremos rebotar con los objetos que se encuentren en el camino del rayo. A este proceso lo llamaremos **ray casting**.

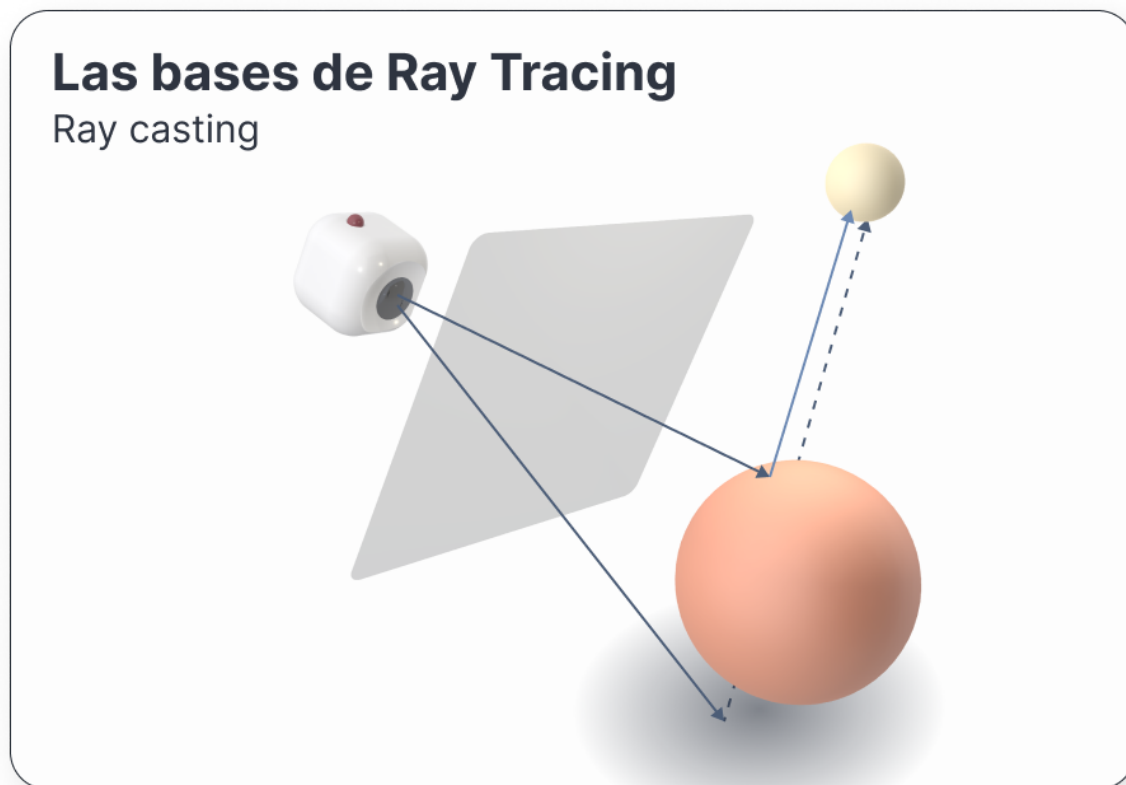


Figure 2.2: Diagrama de ray casting

Generalmente, nos quedaremos con el primer objeto que nos encontremos en su camino. Aunque, a veces, nos interesará saber todos con los que se encuentre.

Cuando un rayo impacta con un objeto, adquirirá parte de las propiedades lumínicas del punto de impacto. Por ejemplo, cuánta luz proporciona la lámpara que tiene encima la esfera de la figura anterior.

Una vez recojamos la información que nos interese, aplicaremos otro raycast desde el nuevo punto de impacto, escogiendo una nueva dirección determinada. Esta dirección dependerá del tipo de material del objeto. Y, de hecho, algunos serán capaces de invocar varios rayos.

Por ejemplo, los espejos reflejan la luz casi de forma perfecta; mientras que otros elementos como el agua o el cristal reflejan y refractan luz, así que necesitaremos generar dos nuevos raycast.

Usando suficientes rayos obtendremos la imagen de la escena. A este proceso de **ray casting recursivo** es lo que se conoce como ray tracing.

Como este proceso puede continuar indefinidamente, tendremos que controlar la profundidad de la recursión. A mayor profundidad, mayor calidad de imagen; pero también, mayor tiempo de ejecución.

2.1 Eligiendo direcciones

Una de las partes más importantes de ray tracing, y a la que quizás dedicaremos más tiempo, es a la elección de la dirección.

Hay varios factores que entran en juego a la hora de decidir qué hacemos cuando impactamos con un nuevo objeto:

1. **¿Cómo es la superficie del material?** A mayor rugosidad, mayor aleatoriedad en la dirección. Por ejemplo, no es lo mismo el asfalto de una carretera que una lámina de aluminio impecable.
2. **¿Cómo de fiel es nuestra geometría?**
3. **¿Dónde se encuentran las luces en la escena?** Dependiendo de la posición, nos interesará muestrear la luz con mayor influencia.

Estas cuestiones las exploraremos a fondo en las siguientes secciones.

2.2 Intersecciones rayo - objeto

Como dijimos al principio del capítulo, representaremos un rayo como

$$\begin{aligned} P(t) &= (O_x, O_y, O_z) + t(D_x, D_y, D_z) = \\ &= (O_x + tD_x, O_y + tD_y, O_z + tD_z) \end{aligned}$$

Por ejemplo, tomando $O = (1, 3, 2)$, $D = (1, 2, 1)$:

- Para $t = 0$, $P(t) = (1, 3, 2)$.
- Para $t = 1$, $P(t) = (1, 3, 2) + (1, 2, 1) = (2, 5, 3)$.

Nos resultará especialmente útil limitar los valores que puede tomar t . Restringiremos los posibles puntos del dominio de forma que $t \in [t_{min}, t_{max}]$, con $t_{min} < t_{max}$. En general, nos interesará separarnos de las superficies un pequeño pero no despreciable ε para evitar errores de redondeo.

TODO dibujo origen - epsilon == tmin -> rayo -> t_max = 1-epsilon

Una de las principales cuestiones que debemos hacernos es saber cuándo un rayo impacta con una superficie. Lo definiremos analíticamente.

2.2.1 Superficies implícitas

Generalmente, cuando hablemos de superficies, nos referiremos [superficies diferenciables](#), pues nos interesará conocer el vector normal en cada punto.

Una superficie implícita es una superficie en un espacio euclidiano definida como

$$F(x, y, z) = 0$$

Esta ecuación implícita define una serie de puntos del espacio \mathbb{R}^3 que se encuentran en la superficie.

Por ejemplo, la esfera se define como $x^2 + y^2 + z^2 - 1 = 0$.

Consideremos una superficie S y un punto regular de ella P ; es decir, un punto tal que el gradiente de F en P no es 0. Se define el vector normal \mathbf{n} a la superficie en ese punto como

$$\mathbf{n} = \nabla F(P) = \left(\frac{\partial F(P)}{\partial x}, \frac{\partial F(P)}{\partial y}, \frac{\partial F(P)}{\partial z} \right)$$

TODO: dibujo de la normal a una superficie.

Dado un punto $Q \in \mathbb{R}^3$, queremos saber dónde interseca un rayo $P(t)$. Es decir, para qué t se cumple que $F(P(t)) = 0 \iff F(O + tD) = 0$.

Consideremos por ejemplo un plano, como en (Shirley and Morley 2003). Para ello, nos tomamos un punto Q_0 del plano y un vector normal a la superficie \mathbf{n} .

La ecuación implícita del plano será

$$F(Q) = (Q - Q_0) \cdot \mathbf{n} = 0$$

Si pinchamos nuestro rayo en la ecuación,

$$\begin{aligned} F(P(t)) &= (P(t) - Q_0) \cdot \mathbf{n} \\ &= (O + tD - Q_0) \cdot \mathbf{n} = 0 \end{aligned}$$

Resolviendo para t , esto se da si

$$\begin{aligned} O \cdot \mathbf{n} + tD \cdot \mathbf{n} - Q_0 \cdot \mathbf{n} &= 0 && \iff \\ tD \cdot \mathbf{n} &= Q_0 \cdot \mathbf{n} - O \cdot \mathbf{n} && \iff \\ t &= \frac{Q_0 \cdot \mathbf{n} - O \cdot \mathbf{n}}{D \cdot \mathbf{n}} \end{aligned}$$

Es decir, hemos obtenido el único valor de t para el cual el rayo toca la superficie.

Debemos tener en cuenta el caso para el cual $D \cdot \mathbf{n} = 0$. Esto solo se da si la dirección y el vector normal a la superficie son paralelos.

TODO: dibujo de dos rayos con un plano: uno corta a la superficie, mientras que el otro es paralelo.

2.2.2 Superficies paramétricas

Otra forma de definir una superficie en el espacio es mediante un subconjunto $D \subset \mathbb{R}^2$ y una serie de funciones, $f, g, h : D \rightarrow \mathbb{R}$, de forma que

$$(x, y, z) = (f(u, v), g(u, v), h(u, v))$$

En informática gráfica, hacemos algo similar cuando mapeamos una textura a una superficie. Se conoce como UV mapping

Demos un par de ejemplos de superficies paramétricas: - El grafo de una función $f : D \rightarrow \mathbb{R}$,

$$G(f) = \{(x, y, f(x, y)) \mid (x, y) \in D\}$$

define una superficie diferenciable siempre que f también lo sea. - Usando coordenadas esféricas (r, θ, ϕ) , podemos parametrizar la esfera como $(x, y, z) = (\cos \phi \sin \theta, \sin \phi \sin \theta, \cos \theta)$

TODO añadir imagen de coordenadas esféricas. U otro capítulo con coordenadas.

NOTE: estoy usando (radial, polar, azimuthal). θ corresponde con la apertura con respecto a la vertical

El vector normal \mathbf{n} a la superficie en un punto (u, v) del dominio viene dado por

$$\mathbf{n}(u, v) = \left(\frac{\partial f}{\partial u}, \frac{\partial g}{\partial u}, \frac{\partial h}{\partial u} \right) \times \left(\frac{\partial f}{\partial v}, \frac{\partial g}{\partial v}, \frac{\partial h}{\partial v} \right)$$

Encontrar el punto de intersección de una superficie paramétrica con un rayo es sencillo. Basta con encontrar aquellos puntos (u, v) y t para los que

$$O_x + tD_x = f(u, v)$$

$$O_y + tD_y = g(u, v)$$

$$O_z + tD_z = h(u, v)$$

Es posible que el rayo no impacte en ningún punto. En ese caso, el sistema de ecuaciones no tendría solución. Otra posibilidad es que intersequen en varios puntos.

2.2.3 Intersecciones con esferas

Estudiemos ahora cómo intersecan una esfera con nuestro rayo. Una esfera de centro C y radio r viene dada por aquellos puntos $P = (x, y, z)$ que cumplen

$$(P - C) \cdot (P - C) = r^2$$

Podemos reescribir esta ecuación en términos de sus coordenadas para obtener

$$(x - C_x)^2 + (y - C_y)^2 + (z - C_z)^2 = r^2$$

Veamos para qué valores de t de nuestro rayo se cumple esa ecuación:

$$\begin{aligned} (P(t) - C) \cdot (P(t) - C) &= r^2 \iff \\ (O + tD - C) \cdot (O + tD - C) &= r^2 \iff \end{aligned}$$

Aplicando las propiedades del producto escalar de la conmutatividad ($a \cdot b = b \cdot a$) y la distributiva ($a \cdot (b + c) = a \cdot b + a \cdot c$), podemos escribir

$$\begin{aligned} ((O - C) + tD) \cdot ((O - C) + tD) &= r^2 \iff \\ (O - C)^2 + 2 \cdot (O - C) \cdot tD + (tD)^2 &= r^2 \iff \\ D^2 t^2 + 2D \cdot (O - C)t + (O - C)^2 - r^2 &= 0 \iff \end{aligned}$$

Así que tenemos una ecuación de segundo grado. Resolviéndola, nos salen nuestros puntos de intersección:

$$t = \frac{-D \cdot (O - C) \pm \sqrt{(D \cdot (O - C))^2 - 4(D^2)((O - C)^2 - r^2)}}{2D^2}$$

Debemos distinguir tres casos, atendiendo al valor que toma el discriminante $\Delta = (D \cdot (O - C))^2 - 4(D^2)((O - C)^2 - r^2)$:

1. Si $\Delta < 0$, $\sqrt{\Delta} \notin \mathbb{R}$, y el rayo no impacta con la esfera
2. Si $\Delta = 0$, el rayo impacta en un punto, que toma el valor $t = \frac{-D \cdot (O - C)}{2D \cdot D}$. Digamos que *pegaría* justo en el borde.
3. Si $\Delta > 0$, existen dos soluciones. En ese caso, el rayo atraviesa la esfera.

TODO dibujo explicativo de la intersección con una esfera.

Para estos dos últimos, si consideramos t_0 cualquier solución válida, el vector normal resultante viene dado por

$$\mathbf{n} = 2(P(t_0) - C)$$

o, normalizando,

$$\hat{\mathbf{n}} = \frac{(P(t_0) - C)}{r}$$

2.2.4 Intersecciones con triángulos

Este tipo de intersecciones serán las más útiles en nuestro path tracer. Generalmente, nuestras geometrías estarán compuestas por mallas de triángulos, así que conocer dónde impacta nuestro rayo será clave. Empecemos por la base:

Un triángulo viene dado por tres puntos, A , B , y C ; correspondientes a sus vértices. Para evitar casos absurdos, supongamos que estos puntos son afinmente independientes; es decir, que no están alineados.

2.2.4.1 Coordenadas baricéntricas

Podemos describir los puntos contenidos en el plano que forman estos vertices mediante **coordenadas baricéntricas**. Este sistema de coordenadas expresa cada punto del plano como una combinación convexa de los vértices. Es decir, que para cada punto P del triángulo existen α, β y γ tales que $\alpha + \beta + \gamma = 1$ y

$$P = \alpha A + \beta B + \gamma C$$

TODO: triángulo con coordenadas baricéntricas.

Debemos destacar que existen dos grados de libertad debido a la restricción de que las coordenadas sumen 1.

Una propiedad de estas coordenadas que nos puede resultar útil es que un punto P está contenido en el triángulo si y solo si $0 < \alpha, \beta, \gamma < 1$.

Esta propiedad y la restricción de que sumen 1 nos da una cierta intuición de cómo funcionan. Podemos ver las coordenadas baricéntricas como la contribución de los vértices a un punto P . Por ejemplo, si $\alpha = 0$, eso significa que el punto viene dado por $\beta B + \gamma C$; es decir, una combinación lineal de B y C . Se encuentra en la recta que generan.

Por proponer otro ejemplo, si alguna de las coordenadas fuera mayor que 1, eso significaría que el punto estaría más allá del triángulo.

TODO: dibujo con explicación de cómo funciona (libreta Shinrin - Yoku)

2.2.4.2 Calculando la intersección

Podemos eliminar una de las varibales escribiendo $\alpha = 1 - \beta - \gamma$, lo que nos dice

$$\begin{aligned} P &= (1 - \beta - \gamma)A + \beta B + \gamma C \\ &= A + (B - A)\beta + (C - A)\gamma \end{aligned}$$

bajo la restricción

$$\begin{aligned}
\beta + \gamma &< 1 \\
0 &< \beta \\
0 &< \gamma
\end{aligned} \tag{2.1}$$

Un rayo $P(t) = O + tD$ impactará en un punto del triángulo si se cumple

$$P(t) = O + tD = A + (B - A)\beta + (C - A)\gamma$$

cumpliendo [2.1]. Podemos expandir la ecuación anterior en sus coordenadas para obtener

$$\begin{aligned}
O_x + tD_x &= A_x + (B_x - A_x)\beta + (C_x - A_x)\gamma \\
O_y + tD_y &= A_y + (B_y - A_y)\beta + (C_y - A_y)\gamma \\
O_z + tD_z &= A_z + (B_z - A_z)\beta + (C_z - A_z)\gamma
\end{aligned}$$

Reordenamos:

$$\begin{aligned}
(A_x - B_x)\beta + (A_x - C_x)\gamma + tD_x &= A_x - O_x \\
(A_y - B_y)\beta + (A_y - C_y)\gamma + tD_y &= A_y - O_y \\
(A_z - B_z)\beta + (A_z - C_z)\gamma + tD_z &= A_z - O_z
\end{aligned}$$

Lo que nos permite escribir el sistema en forma de ecuación:

$$\begin{pmatrix} A_x - B_x & A_x - C_x & D_x \\ A_y - B_y & A_y - C_y & D_y \\ A_z - B_z & A_z - C_z & D_z \end{pmatrix} \begin{pmatrix} \beta \\ \gamma \\ t \end{pmatrix} = \begin{pmatrix} A_x - O_x \\ A_y - O_y \\ A_z - O_z \end{pmatrix}$$

Calcular rápidamente la solución a un sistema de ecuaciones lineales es un problema habitual. En (Shirley and Morley 2003) se utiliza la regla de Cramer para hacerlo, esperando que el compilador optimice las variables intermedias creadas. Nosotros no nos tendremos que preocupar de esto en particular, ya que el punto de impacto lo calculará la GPU gracias

a las [herramientas aportadas por KHR](#).

Para obtener el vector normal, podemos hacer el producto vectorial de dos vectores que se encuentren en el plano del triángulo. Como, por convención, los vértices se guardan en sentido antihorario visto desde fuera del objeto, entonces

$$\mathbf{n} = (B - A) \times (C - A)$$

Wikipedia contributors ([2022e](#)), Wikipedia contributors ([2021c](#)), Wikipedia contributors ([2022f](#)), Alfonso Romero Sarabia ([2021](#))

3 Integración de Monte Carlo

TODO: este capítulo seguramente debería ir más tarde. De esa forma, puedo introducir otros conceptos antes. De momento, se queda aquí.

Una de las partes más importante de nuestro ray tracer es saber calcular la cantidad de luz en un punto de la escena. Para ello, necesitaríamos hallar la radianza en dicha posición mediante la *rendering equation*. Sin embargo, es *muy* difícil resolverla; tanto computacional como analíticamente. Por ello, debemos atacar el problema desde otro punto de vista.

Las técnicas de Monte Carlo nos permitirán aproximar el valor que toman las integrales mediante una estimación. Utilizando muestreo aleatorio para evaluar puntos de una función, seremos capaces de obtener un resultado suficientemente bueno.

Una de las propiedades que hacen interesantes a este tipo de métodos es la **independencia del ratio de convergencia y la dimensionalidad del integrando**. Sin embargo, conseguir un mejor rendimiento tiene un precio a pagar. Dadas n muestras, la convergencia a la solución correcta tiene un orden de $\mathcal{O}(n^{-1/2})$. Es decir, para reducir el error a la mitad, necesitaríamos 4 veces más muestras.

En este capítulo veremos el fundamento de la integración de Monte Carlo, cómo muestrear distribuciones específicas y cómo afinar el resultado final.

3.1 Repaso de probabilidad

Necesitaremos unas cuantas nociones de variable aleatoria para poder entender la integración de Monte Carlo, así que vamos a hacer un breve repaso.

Una **variable aleatoria** X (v.a.) es, esencialmente, una regla que asigna un valor numérico a cada posibilidad de proceso de azar. Formalmente, es una función definida en un espacio de probabilidad (Ω, \mathcal{A}, P) asociado a un experimento aleatorio:

$$X : \Omega \rightarrow \mathbb{R}$$

A Ω lo conocemos como espacio muestral (conjunto de todas las posibilidades), \mathcal{A} es una σ -álgebra de subconjuntos de Ω que refleja todas las posibilidades de eventos aleatorios, y P es una función probabilidad, que asigna a cada evento una probabilidad.

NOTE: no sé hasta qué punto debería meterme en la definición formal de variable aleatoria. Es una movida tremenda para poca cosa que necesitamos. De momento, voy con lo más interesante.

Una variable aleatoria X puede clasificarse en discreta o continua, dependiendo de cómo sea su rango $R_X = \{x \in \mathbb{R} \mid \exists \omega \in \Omega : X(\omega) = x\}$:

3.1.1 Variables aleatorias discretas

Las v.a. discretas son aquellas cuyo rango es un conjunto discreto.

Para comprender mejor cómo funcionan, pongamos un ejemplo: Consideremos un experimento en el que lanzamos dos dados, anotando lo que sale en cada uno. Los posibles valores que toman serán

$$\begin{aligned} &\{(1, 1), (1, 2), (1, 3), (1, 4), (1, 5), (1, 6), \\ &\quad (2, 1), (2, 2), (2, 3), (2, 4), (2, 5), (2, 6), \\ &\quad (3, 1), (3, 2), (3, 3), (3, 4), (3, 5), (3, 6), \\ &\quad (4, 1), (4, 2), (4, 3), (4, 4), (4, 5), (4, 6), \\ &\quad (5, 1), (5, 2), (5, 3), (5, 4), (5, 5), (5, 6), \\ &\quad (6, 1), (6, 2), (6, 3), (6, 4), (6, 5), (6, 6)\} \end{aligned}$$

Cada resultado tiene la misma probabilidad de ocurrir (claro está, si el dado no está truco). Como hay 36 posibilidades, la probabilidad de obtener un cierto valor es de $\frac{1}{36}$.

La v.a. X denotará la suma de los valores obtenidos en cada uno. Así, por ejemplo, si al lanzar los dados hemos obtenido $(1, 3)$, X tomará el valor 4. En total, X puede tomar todos los valores comprendidos entre 2 y 12. Este sería el **espacio muestral**. Cada pareja no está asociada a un único valor de X . Por ejemplo, $(1, 2)$ suma lo mismo que $(2, 1)$. Esto nos lleva a preguntarnos... ¿Cuál es la probabilidad de que X adquiera un cierto valor?

La **función masa de probabilidad** nos permite conocer la probabilidad de que X tome un cierto valor x . Se denota por $P(X = x)$.

En este ejemplo, la probabilidad de que X tome el valor 4 es

$$\begin{aligned} P(X = 4) &= \sum \text{nº parejas que suman 4} \cdot \text{probabilidad de que salga la pareja} \\ &= 3 \cdot \frac{1}{36} = \frac{1}{12} \end{aligned}$$

Las parejas serían $(1, 3)$, $(2, 2)$ y $(3, 1)$.

Por definición, si el espacio muestral de X es $\Omega = \{x_1, \dots, x_n\}$, la función masa de probabilidad debe cumplir que

$$\sum_{i=1}^n P(X = x_i) = 1$$

Muchas veces nos interesará conocer la probabilidad de que X se quede por debajo de un cierto valor x (de hecho, podemos caracterizar distribuciones aleatorias gracias a esto). Para ello, usamos la **función de distribución**:

$$F_X(x) = P(X \leq x) = \sum_{\substack{k \in \Omega \\ k \leq x}} P(X = k)$$

Es una función continua por la derecha y monótona no decreciente. Además, se cumple que $0 \leq F_X \leq 1(x)$ y $\lim_{x \rightarrow -\infty} F_X = 0$, $\lim_{x \rightarrow \infty} F_X = 1$.

En nuestro ejemplo, si consideramos $x = 3$:

$$\begin{aligned}
 F_X(x) &= \sum_{i=1}^3 P(X = i) = P(X = 1) + P(X = 2) + P(X = 3) \\
 &= \frac{1}{36} + \frac{2}{36} + \frac{3}{36} = \frac{1}{12}
 \end{aligned}$$

3.1.2 Variables aleatorias continuas

Este tipo de variables aleatorias tienen un rango no numerable; es decir, el conjunto de valores que puede tomar abarca un intervalo de números.

Un ejemplo podría ser la altura de una persona.

Si en las variables aleatorias discretas teníamos funciones masa de probabilidad, aquí definiremos las **funciones de densidad de probabilidad** (o simplemente, funciones de densidad). La idea es la misma: nos permite conocer la probabilidad de que nuestra variable aleatoria tome un cierto valor del espacio muestral.

Es importante mencionar que, aunque *la probabilidad de que la variable aleatoria tome un valor específico* es 0, ya que nos encontramos en un conjunto no numerable, sí que podemos calcular la probabilidad de que se encuentre entre dos valores. Por tanto, si la función de densidad es f_X , entonces

$$P(a \leq X \leq b) = \int_a^b f_X(x) dx$$

La función de densidad tiene dos características importantes:

1. f_X es no negativa; esto es, $f_X(x) \geq 0 \forall x \in \Omega$
2. f_X integra uno en todo el espacio muestral:

$$\int_{\Omega} f_X(x) = 1$$

Intuitivamente, podemos ver esta última propiedad como *si acumulamos todos los valores que puede tomar la variable aleatoria, la probabilidad de que te encuentres en el conjunto debe*

ser 1. Si tratamos con un conjunto de números reales, podemos escribir la integral como $\int_{-\infty}^{\infty} f_X(x) = 1$.

Una de las variables aleatorias que más juego nos darán en el futuro será la **v.a. con distribución uniforme en $[0, 1)$** . La denotaremos como ξ , y escribiremos $\xi \sim U([0, 1))$. La probabilidad de que ξ tome un valor es constante, por lo que podemos definir su función de densidad como

$$f(\xi) = \begin{cases} 1 & \text{si } \xi \in [0, 1) \\ 0 & \text{en otro caso.} \end{cases}$$

La probabilidad de que ξ tome un valor entre dos elementos $a, b \in [0, 1)$ es

$$P(\xi \in [a, b]) = \int_a^b 1 dx = b - a$$

Como veremos más adelante, definiendo correctamente una función de densidad conseguiremos mejorar el rendimiento del path tracer.

La función de distribución $F_X(x)$ podemos definirla como:

$$F_X(x) = P(X \leq x) = \int_{-\infty}^x f_X(t) dt$$

Es decir, dado un x , ¿cuál sería la probabilidad de que X se quede por debajo de x ?

El Teorema Fundamental del Cálculo nos permite relacionar función de distribución y función de densidad directamente:

$$f_X(x) = \frac{dF_X(x)}{dx}$$

3.1.3 Esperanza y varianza de una variable aleatoria

La **esperanza de una variable aleatoria**, denotada $E[X]$, es una generalización de la media ponderada. Nos informa del *valor esperado* de dicha variable aleatoria.

En el caso de las variables discretas, se define como

$$E[X] = \sum_{x_i \in \Omega} x_i p_i$$

donde x_i son los posibles valores que puede tomar la v.a., y p_i la probabilidad asociada a cada uno de ellos; es decir, $p_i = P[X = x_i]$

Para una variable aleatoria continua real, la esperanza viene dada por

$$E[X] = \int_{-\infty}^{\infty} x f_X(x) dx$$

aunque, generalizando a una v.a. con espacio muestral Ω , la esperanza se puede generalizar como

$$E[X] = \int_{\Omega} x f_X(x) dx$$

Pongamos un par de ejemplos del cálculo de la esperanza. En el **ejemplo de las variables discretas**, la esperanza venía dada por

$$E[X] = \sum_{i=2}^{12} i \cdot P[X = i] = 2 \cdot \frac{1}{36} + 3 \cdot \frac{2}{36} + \dots + 12 \cdot \frac{1}{36} = 7$$

Para variables aleatorias uniformes en (a, b) (es decir, $X \sim U(a, b)$), la esperanza es

$$E[X] = \int_a^b x \cdot \frac{1}{b-a} dx = \frac{a+b}{2}$$

La esperanza tiene unas cuantas propiedades que nos resultarán muy útiles. Estas son:

- **Linealidad:**

- Si X, Y son dos v.a., $E[X + Y] = E[X] + E[Y]$
- Si a es una constante, X una v.a., entonces $E[aX] = aE[X]$
- Análogamente, para ciertas X_1, \dots, X_k , $E\left[\sum_{i=1}^k X_i\right] = \sum_{i=1}^k E[X_i]$

- Estas propiedades no necesitan que las variables aleatorias sean independientes. Este hecho será clave para las técnicas de Monte Carlo.
- La **Ley del estadístico inconsciente** (*Law of the unconscious statistician*, o LOTUS): dada una variable aleatoria X y una función medible g , la esperanza de $g(X)$ se puede calcular como

$$E[g(X)] = \int_{\Omega} g(x) f_X(x) dx$$

Esta propiedad será clave en nuestro desarrollo.

Será habitual encontrarnos con el problema de que no conocemos la distribución de una variable aleatoria Y . Sin embargo, si encontramos una transformación medible de una variable aleatoria X de forma que obtengamos Y (esto es, $\exists g$ función medible tal que $g(X) = Y$), entonces podemos calcular la esperanza de Y fácilmente. Esta propiedad hará que las variables aleatorias con distribución uniforme adquieran muchísima importancia. Generar números aleatorios en $[0, 1)$ es muy fácil, así **que obtendremos otras v.a.s a partir de ξ .**

Otra medida muy útil de una variable aleatoria es **la varianza**. Nos permitirá medir cómo de dispersa es la distribución con respecto a su media. La denotaremos como $Var[X]$, y se define como

$$Var[X] = E[(X - E[X])^2]$$

Si desarrollamos esta definición, podemos conseguir una expresión algo más agradable:

$$\begin{aligned} Var[X] &= E[(X - E[X])^2] = \\ &= E[X^2 + E[X]^2 - 2XE[X]] = \\ &= E[X^2] + E[X]^2 - 2E[X]E[X] = \\ &= E[X^2] - E[X]^2 \end{aligned}$$

Hemos usado que $E[E[X]] = E[X]$ y la linealidad de la esperanza.

Enunciemos un par de propiedades que tiene, similares a la de la esperanza:

- La varianza saca constantes al cuadrado: $Var[aX] = a^2 Var[X]$
- $Var[X + Y] = Var[X] + Var[Y] + 2Cov[X, Y]$, donde $Cov[X, Y]$ es la covarianza de X y Y .
 - En el caso en el que X e Y sean incorreladas (es decir, la covarianza es 0), $Var[X + Y] = Var[X] + Var[Y]$.

La varianza nos será útil a la hora de medir el error cometido por una estimación de Monte Carlo.

3.2 El estimador de Monte Carlo

Tras este breve repaso de probabilidad, estamos en condiciones de definir el estimador de Monte Carlo. Primero, vamos con su versión más sencilla.

Los estimadores de Monte Carlo nos permiten hallar la esperanza de una variable aleatoria, digamos, Y , sin necesidad de calcular explícitamente su valor. Para ello, tomamos unas cuantas muestras Y_1, \dots, Y_N que sigan la misma distribución que Y con media μ . Entonces, consideramos el estimador de μ (Owen 2013):

$$\hat{\mu}_N = \frac{1}{N} \sum_{i=1}^N Y_i \quad (3.1)$$

Haciendo la esperanza de este estimador, vemos que

$$\begin{aligned} E[\hat{\mu}_N] &= E \left[\frac{1}{N} \sum_{i=1}^N Y_i \right] = \frac{1}{N} E \left[\sum_{i=1}^N Y_i \right] \\ &= \frac{1}{N} \sum_{i=1}^N E[Y_i] = \frac{1}{N} \sum_{i=1}^N \mu = \\ &= \mu \end{aligned}$$

A este tipo de estimadores se les llama insesgados.

Generalmente nos encontraremos en la situación en la que $Y = f(X)$, donde X sigue una distribución con función de densidad $p_X(x)$, y $f : S \rightarrow \mathbb{R}$. En ese caso, sabemos que la esperanza de Y se puede calcular como

$$\mu = E[Y] = E[f(X)] = \int_S f(x)p_X(x)dx$$

Lo que estamos buscando es calcular $\int_S f(x)dx$. Entonces, ¿qué ocurre si intentamos compensar en [3.1] con la función de densidad?

$$\begin{aligned} E \left[\frac{1}{N} \sum_{i=1}^N \frac{f(X_i)}{p_X(X_i)} \right] &= \frac{1}{N} \sum_{i=1}^N E \left[\frac{f(X_i)}{p_X(X_i)} \right] = \\ &= \frac{1}{N} \sum_{i=1}^N \left(\int_S \frac{f(x)}{p_X(x)} p_X(x) dx \right) = \\ &= \frac{1}{N} N \int_S f(x) dx = \\ &= \int_S f(x) dx \end{aligned}$$

¡Genial! Esto nos da una forma de calcular la integral de una función usando muestras de variables aleatorias con cierta distribución. Llamaremos al estimador de Monte Carlo

$$\hat{F}_N = \frac{1}{N} \sum_{i=1}^N \frac{f(X_i)}{p_X(X_i)} \quad (3.2)$$

Es importante mencionar que $p_X(x)$ debe ser distinto de 0 cuando f también lo sea.

Podemos particularizar el caso en el que nuestras muestras X_i sigan una distribución uniforme en $[a, b]$. Si eso ocurre, su función de densidad es $p_X(x) = \frac{1}{b-a}$, así que podemos simplificar un poco [3.2]:

$$\hat{F}_N = \frac{b-a}{N} \sum_{i=1}^N f(X_i)$$

Elegir correctamente la función de densidad p_X será clave. Si conseguimos escogerla de-

bidamente, reduciremos mucho el error que genera el estimador. Esto es lo que se conoce como *importance sampling*.

TODO: añadir enlace al capítulo de importance sampling.

Podemos calcular el error cuadrático medio de la estimación si volvemos al estimador de la media $\hat{\mu}_N$ [3.1]. Para ello, necesitamos la varianza: como $\hat{\mu}_N$ es insesgado, tenemos que

$$\begin{aligned} \text{Var}[\hat{\mu}_N] &= \text{Var}\left[\frac{1}{N} \sum_{i=1}^N Y_i\right] = \frac{1}{N^2} \text{Var}\left[\sum_{i=1}^N Y_i\right] = \\ &= \frac{1}{N^2} \sum_{i=1}^N \text{Var}[Y_i] = \frac{1}{N^2} N \text{Var}[Y] = \\ &= \frac{\text{Var}[Y]}{N} \end{aligned}$$

El error cuadrático medio es

$$\sqrt{\text{Var}[\hat{\mu}_N]} = \sqrt{\frac{\text{Var}[Y]}{N}} = \frac{\sqrt{\text{Var}[Y]}}{\sqrt{N}}$$

así que, como adelantamos al inicio del capítulo, la estimación tiene un error del orden $\mathcal{O}(N^{-1/2})$. Esto nos dice que, para reducir el error a la mitad, debemos tomar 4 veces más muestras.

Pongamos un ejemplo de estimador de Monte Carlo para una caja de dimensiones $[x_0, x_1] \times [y_0, y_1] \times [z_0, z_1]$. Si queremos estimar la integral de la función $f : \mathbb{R}^3 \rightarrow \mathbb{R}$

$$\int_{x_0}^{x_1} \int_{y_0}^{y_1} \int_{z_0}^{z_1} f(x, y, z) dx dy dz$$

mediante una variable aleatoria $X \sim U([x_0, x_1] \times [y_0, y_1] \times [z_0, z_1])$ con función de densidad $p(x, y, z) = \frac{1}{x_1 - x_0} \frac{1}{y_1 - y_0} \frac{1}{z_1 - z_0}$, tomamos el estimador

$$\hat{F}_N = \frac{1}{(x_1 - x_0) \cdot (y_1 - y_0) \cdot (z_1 - z_0)} \sum_{i=1}^N f(X_i)$$

3.3 Escogiendo puntos aleatorios

Una de las partes clave del estimador de Monte Carlo [3.2] es saber escoger la función de densidad p_X correctamente. En esta sección, veremos algunos métodos para conseguir distribuciones específicas partiendo de funciones de densidad sencillas.

3.3.1 Método de la transformada inversa

En resumen: Para conseguir una muestra de una distribución específica F_X :

1. Generar un número aleatorio $\xi \sim U(0, 1)$.
2. Hallar la inversa de la función de distribución deseada F_X , denotada $F_X^{-1}(x)$.
3. Calcular $F_X^{-1}(\xi) = X$.

Este método nos permite conseguir muestras de cualquier distribución continua a partir de variables aleatorias uniformes, siempre que se conozca la inversa de la función de distribución.

Sea X una variable aleatoria con función de distribución F_X ¹. Queremos buscar una transformación $T : [0, 1] \rightarrow \mathbb{R}$ tal que $T(\xi) \stackrel{d}{=} X$, siendo ξ una v.a. uniformemente distribuida. Para que esto se cumpla, se debe dar

$$\begin{aligned} F_X(x) &= P[X < x] = \\ &= P[T(\xi) < x] = \\ &= P(\xi < T^{-1}(x)) = \\ &= T^{-1}(x) \end{aligned}$$

Este último paso se debe a que, como ξ es uniforme en $(0, 1)$, $P[\xi < x] = x$. Es decir, hemos obtenido que F_X es la inversa de T .

TODO: dibujo similar a [este](#): p.52

¹En su defecto, si tenemos una función de densidad f_X , podemos hallar la función de distribución haciendo $F_X(x) = P[X < x] = \int_{x_{min}}^x f_X(t)dt$

Como ejemplo, vamos a muestrear la función $f(x) = x^2$, $x \in [0, 2]$.

Primero, normalizamos esta función para obtener una función de densidad $p_X(x)$. Es decir, buscamos $p_X(x) = cf(x)$ tal que

$$\begin{aligned} 1 &= \int_0^2 p_X(x)dx = \int_0^2 cf(x)dx = c \int_0^2 f(x)dx = \\ &= \left. \frac{cx^3}{3} \right|_0^2 = \frac{8c}{3} \\ \Rightarrow c &= \frac{3}{8} \\ \Rightarrow p_X(x) &= \frac{3x^2}{8} \end{aligned}$$

A continuación, integramos la función de densidad para obtener la de distribución F_X :

$$F_X(x) = \int_0^x p_X(x)dx = \int_0^x \frac{3x^2}{8} = \frac{x^3}{8}$$

Solo nos queda conseguir la muestra. Para ello,

$$\begin{aligned} \xi &= F_X(x) = \frac{x^3}{8} \iff \\ x &= \sqrt[3]{8\xi} \end{aligned}$$

Sacando un número aleatorio ξ , y pasándolo por la función obtenida, conseguimos un elemento con distribución $f(x)$.

3.3.2 Método del rechazo

En resumen: Para conseguir una muestra de una variable aleatoria X con función de densidad p_X :

1. Obtener una muestra y de Y , y otra ξ de $U(0, 1)$.
2. Comprobar si $\xi < \frac{p_X(y)}{Mp_Y(y)}$. Si es así, aceptarla. Si no, sacar otra muestra.

El método anterior presenta principalmente dos problemas:

1. No siempre es posible integrar una función para hallar su función de densidad.
2. La inversa de la función de distribución, F_X^{-1} no tiene por qué existir.

Como alternativa, podemos usar este método (en inglés, *rejection method*). Para ello, necesitamos una variable aleatoria Y con función de densidad $p_Y(y)$. El objetivo es conseguir una muestra de X con función de densidad $p_X(x)$.

La idea principal es aceptar una muestra de Y con probabilidad p_X/Mp_Y , con $1 < M < \infty$. En esencia, estamos jugando a los dados: si la muestra de y que hemos obtenido se queda por debajo de la gráfica de la función $Mp_Y < p_X$, estaremos obteniendo una de p_X .

TODO dibujo de la gráfica $\frac{p_X(y)}{Mp_Y(y)}$.

¿Quizás haga falta una demostración también? No estoy satisfecho con este apartado ahora mismo. Necesita trabajo.

El algoritmo consiste en:

1. Obtener una muestra de Y , denotada y , y otra de $U(0, 1)$, llamada ξ .
2. Comprobar si $\xi < \frac{p_X(y)}{Mp_Y(y)}$.
 1. Si se cumple, se acepta y como muestra de p_X
 2. En caso contrario, se rechaza y y se vuelve al paso 1.

Wikipedia contributors (2021a), Wikipedia collaborators (2021), Wikipedia collaborators (2022c), Wikipedia collaborators (2022a), Wikipedia contributors (2022g), Galvin (n.d.), Wikipedia contributors (2022i), Wikipedia collaborators (2022d), Wikipedia collaborators (2022b), Wikipedia contributors (2022a), Shirley and Morley (2003), Pharr, Jakob, and Humphreys (2016), Owen (2013), Berkeley cs184 (n.d.)

4 Radiometría

En este capítulo estudiaremos las bases de la radiometría. Esta área de la óptica nos proporcionará una serie de herramientas con las cuales podremos responder a la pregunta *cuánta luz existe en un punto*.

4.1 Unidades básicas

Nota: cuando usemos un paréntesis tras una ecuación, dentro denotaremos sus unidades de medida.

Antes de comenzar a trabajar, necesitamos conocer *qué entendemos* por luz. Aunque hay muchas formas de trabajar con ella (a fin de cuentas, todavía seguimos discutiendo sobre *qué es exactamente la luz*¹), nosotros nos quedaremos con algunas pinceladas de la cuántica. Nos será suficiente quedarnos con la concepción de fotón. Una fuente de iluminación emite una serie de fotones. Estos fotones tienen(Shirley and Morley 2003) una posición, una dirección de propagación y una longitud de onda λ . Un fotón también tiene asociado una velocidad c que depende del índice de refracción del medio, n .

La unidad de medida de λ es el nanómetro (nm). También nos vendrá bien definir una frecuencia, f . Su utilidad viene del hecho de que, cuando la luz cambia de medio al propagarse, la frecuencia se mantiene constante.

$$f = \frac{c}{\lambda}$$

¹No entraremos en detalle sobre la naturaleza de la luz. Sin embargo, si te pica la curiosidad, hay muchos divulgadores como [QuantumFracture](#) que han tratado el tema con suficiente profundidad.

Un fotón tiene asociada una carga de energía, denotada por Q :

$$Q = hf = \frac{hc}{\lambda} (\text{J})$$

donde $h = 6.62607004 \times 10^{-34} \text{J} \cdot \text{s}$ es la constante de Plank y $c = 299792458 \text{m/s}$ la velocidad de la luz.

En realidad, **todas estas cantidades deberían tener un subíndice λ** , puesto que dependen de la longitud de onda. La energía de un fotón Q , por ejemplo, debería denotarse Q_λ . Sin embargo, en la literatura de informática gráfica, **se ha optado por omitirla**. ¡Tenlo en cuenta a partir de aquí!

4.1.1 Potencia

A partir de la energía anterior, podemos estimar *la tasa de producción de energía*. A esta tasa la llamaremos (Pharr, Jakob, and Humphreys 2016) **potencia**, o **flujo radiante** Φ . Esta medida nos resultará más útil que la energía total, puesto que nos permite estimar la energía en un instante:

$$\Phi = \lim_{\Delta t \rightarrow 0} \frac{\Delta Q}{\Delta t} = \frac{dQ}{dt} (\text{J/s})$$

Su unidad es julios por segundo, comúnmente denotado vatio (*watts*, W). También se utiliza el lumen. Podemos encontrar la energía total en un periodo de tiempo $[t_0, t_1]$ integrando el flujo radiante:

$$Q = \int_{t_0}^{t_1} \Phi(t) dt$$

4.1.2 Irradiancia

La **irradiancia** o **radiancia emitida** es el flujo radiante que recibe una superficie. Dada un área A , se define como

$$E = \frac{\Phi}{A} (W/m^2)$$

TODO: dibujo como el de <https://cs184.eecs.berkeley.edu/public/sp22/lectures/lec-11-radiometry-and-photometry/lec-11-radiometry-and-photometry.pdf>, p.28

Ahora que tenemos la potencia emitida en una cierta área, nos surge una pregunta: ¿y en un cierto punto p ? Tomando límites en la expresión anterior, encontramos la respuesta:

$$E(p) = \lim_{\Delta A \rightarrow 0} \frac{\Delta \Phi}{\Delta A} = \frac{d\Phi}{dA} (W/m^2)$$

De la misma manera que con la potencia, integrando $E(p)$ podemos obtener el flujo radiante:

$$\Phi = \int_A E(p) dp$$

El principal problema de la irradiancia es que *no nos dice nada sobre las direcciones* desde las que ha llegado la luz.

4.1.3 Ángulos sólidos

Con estas tres unidades básicas, nos surge una pregunta muy natural: ¿cómo mido cuánta luz llega a una superficie?

Para responder a esta pregunta, usaremos los **ángulos sólidos**.

Un ángulo sólido es la extensión del concepto de **ángulo planar** (en dos dimensiones). Para ilustrar el sentido de estos ángulos, imaginemos que tenemos un cierto objeto en dos dimensiones delante de nosotros. Dependiendo de cómo de lejos se encuentre, nos parecerá más grande o más pequeño. Si trazamos un par de líneas desde nuestra posición a las partes más alejadas de este objeto, y las cortamos con una circunferencia de radio r , obtendremos un par de puntos en dicha circunferencia. Al arco que encapsulan ambos puntos le corresponde un cierto ángulo: el ángulo planar.

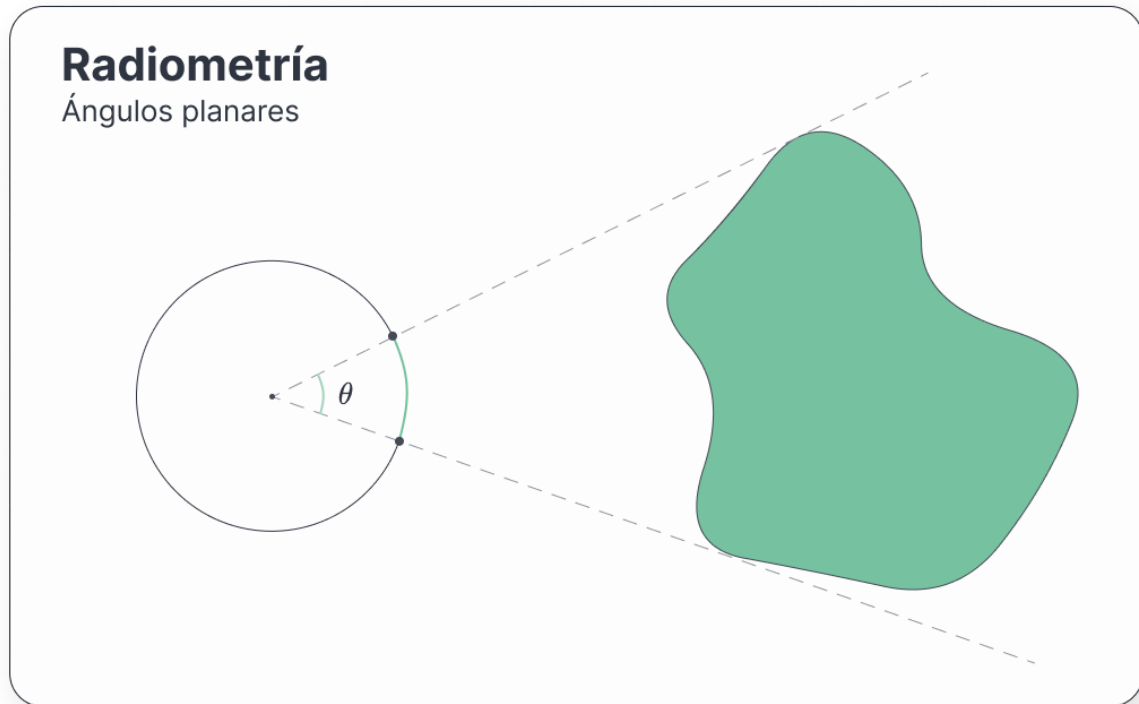


Figure 4.1: La idea intuitiva de un ángulo planar

Llevando esta idea a las tres dimensiones es como conseguimos el concepto de **ángulo sólido**. Su unidad de medida es el estereorradián (sr). Corresponde a la superficie generada por las rectas proyectantes que van desde un objeto hasta un punto, cortando con una esfera de radio r . Se define como

$$\omega = \frac{A}{r^2}(\text{sr})$$

siendo A la superficie cubierta por el objeto. Por tanto, un estereorradián corresponde una superficie con área r^2 : $1\text{sr} = \frac{r^2}{r^2}$.

TODO: foto de un ángulo sólido.

Si 2π radianes corresponden a la circunferencia completa, para la esfera se tiene que 4π estereorradianes cubren toda la superficie de esta. Además, 2π sr cubren un hemisferio.

Denotaremos a los ángulos sólidos por ω . Como hemos visto, $\omega \in [0, 4\pi]$.

De vez en cuando, usaremos ω **un vector dirección unitario en la esfera**.

TODO: y si meto un xkcd <https://xkcd.com/1276/>

Usualmente emplearemos coordenadas esféricas cuando trabajemos con ellos, dado que resulta más cómodo.

$$\begin{cases} x = \sin \theta \cos \phi \\ y = \sin \theta \sin \phi \\ z = \cos \theta \end{cases}$$

A θ se le denomina ángulo polar, mientras que a ϕ se le llama acimut. Imaginémonos un punto en la esfera de radio r ubicado en una posición (r, θ, ϕ) . Queremos calcular un área chiquitita dA_h , de forma que el ángulo sólido asociado a dicha área debe ser $d\omega$. Así, $d\omega = \frac{dA_h}{r^2}$. Si proyectamos el área, obtenemos $d\theta$ y $d\phi$: pequeños cambios en los ángulos que nos generan nuestra pequeña área.

dA_h debe tener dos lados $lado_1$ y $lado_2$. Podemos hallar $lado_1$ si lo trasladamos al eje z de nuevo. Así, $lado_1 = r \sin \theta d\theta$. De la misma manera, $lado_2 = r d\phi$.

TODO: foto que explique todo esto, porque si no, no hay quien se entere. Quizás me sirva la de <https://cs184.eecs.berkeley.edu/public/sp22/lectures/lec-11-radiometry-and-photometry/lec-11-radiometry-and-photometry.pdf>, p.16 siempre que adapte ϕ .

Poniendo estos valores en $d\omega$:

$$\begin{aligned} d\omega &= \frac{dA_h}{r^2} = \frac{lado_1 lado_2}{r^2} = \\ &= \frac{r \sin \theta d\theta r d\phi}{r^2} = \\ &= \sin \theta d\theta d\phi \end{aligned} \tag{4.1}$$

¡Genial! Acabamos de añadir un recurso muy potente a nuestro inventario. Esta expresión nos permitirá convertir integrales sobre ángulos sólidos en integrales sobre ángulos esféricos.

4.1.4 Intensidad radiante

Los ángulos sólidos nos proporcionan una variedad de herramientas nuevas considerable. Gracias a ellos, podemos desarrollar algunos conceptos nuevos. Uno de ellos es la **intensidad radiante**.

Imaginémonos un pequeño punto de luz encerrado en una esfera, el cual emite fotones en todas direcciones. Nos gustaría medir cuánta energía pasa por la esfera. Podríamos entonces definir

$$I = \frac{\Phi}{4\pi} (\text{W/sr})$$

Otra unidad de medida es el lumen por esterradián, (lm/sr). La anterior definición mide cuántos fotones pasan por toda la esfera. ¿Qué ocurre si *cerramos* el ángulo, restringiéndonos así a un área muy pequeña de la esfera?

$$I = \lim_{\Delta\omega \rightarrow 0} \frac{\Delta\Phi}{\Delta\omega} = \frac{d\Phi}{d\omega}$$

De la misma manera que con los conceptos anteriores, podemos volver a la potencia integrando sobre un conjunto de direcciones:

$$\Phi = \int_{\Omega} I(\omega) d\omega$$

4.1.5 Radiancia

Finalmente, llegamos al concepto más importante. La **radiancia espectral** (o radiancia a secas²) es una extensión de la radiancia emitida teniendo en cuenta la dirección:

²Recuerda que estamos omitiendo la longitud de onda λ .

$$L(p, \omega) = \lim_{\Delta\omega \rightarrow 0} \frac{\Delta E_\omega(p)}{\Delta\omega} = \frac{dE_\omega(p)}{d\omega}$$

siendo $E_\omega(p)$ la radiancia emitida a la superficie perpendicular a ω .

TODO: foto como la de <https://cs184.eecs.berkeley.edu/public/sp22/lectures/lec-11-radiometry-and-photometry/lec-11-radiometry-and-photometry.pdf>, página 10.

Podemos dar otra expresión de la radiancia en términos del flujo:

$$L(p, \omega) = \frac{d^2\Phi(p, \omega)}{d\omega dA^\perp} = \frac{d^2\Phi(p, \omega)}{d\omega dA \cos \theta}$$

donde dA^\perp es el área proyectada por dA en una hipotética superficie perpendicular a ω :

TODO: figura similar a pbr figura 5.10 https://www.pbr-book.org/3ed-2018/Color_and_Radiometry/

Cuando un rayo impacta en una superficie, L puede tomar valores muy diferentes en un lado y otro de dicha superficie. Por ejemplo, si nos imaginamos un espejo, el valor un poco por encima y un poco por debajo de un punto del espejo es muy diferente. Para solucionarlo, podemos tomar límites para distinguir a ambos lados:

$$\begin{aligned} L^+(p, \omega) &= \lim_{t \rightarrow 0^+} L(p + t\mathbf{n}_p, \omega) \\ L^-(p, \omega) &= \lim_{t \rightarrow 0^-} L(p + t\mathbf{n}_p, \omega) \end{aligned} \tag{4.2}$$

donde \mathbf{n}_p es la normal en el punto p .

Otra forma de solucionarlo (y preferible, puesto que simplifica entender lo que ocurre) es distinguir entre la radiancia que llega a un punto –la incidente–, y la que sale.

La primera se llamará $L_i(p, \omega)$, mientras que la segunda será $L_o(p, \omega)$. Es importante destacar que ω apunta *hacia fuera* de la superficie. Quizás es contraintuitivo en L_i , puesto que $-\omega$ apunta *hacia* la superficie. Depende del autor se utiliza una concepción u otra.

Utilizando esta notación y usando [4.2], podemos escribir L_i y L_o como

$$L_i(p, \omega) = \begin{cases} L^+(p, -\omega) & \text{si } \omega \cdot \mathbf{n}_p > 0 \\ L^-(p, -\omega) & \text{si } \omega \cdot \mathbf{n}_p < 0 \end{cases}$$

$$L_o(p, \omega) = \begin{cases} L^+(p, \omega) & \text{si } \omega \cdot \mathbf{n}_p > 0 \\ L^-(p, \omega) & \text{si } \omega \cdot \mathbf{n}_p < 0 \end{cases}$$

TODO: <https://cs184.eecs.berkeley.edu/public/sp22/lectures/lec-11-radiometry-and-photometry/lec-11-radiometry-and-photometry.pdf>, p.36

Una propiedad a tener en cuenta es que, si cogemos un punto p del espacio donde no existe ninguna superficie, $L_o(p, \omega) = L_i(p, -\omega) = L(p, \omega)$

La importancia de la radiancia se debe a un par de propiedades:

La primera de ellas es que, dado L , podemos calcular cualquier otra unidad básica mediante integración. Además, su valor se mantiene constante en rayos que viajan en el vacío. Parece natural usarla en un ray tracer.

4.2 Integrales radiométricas

En esta sección, vamos a explorar las nuevas herramientas que nos proporciona la radiancia. Veremos también cómo integrar ángulos sólidos, y cómo simplificar dichas integrales.

4.2.1 Una nueva expresión de la irradiancia y el flujo

Como dijimos al final de **la sección de la irradiancia**, esta medida no tiene en cuenta las direcciones desde las que llegaba la luz. A diferencia de esta, la radiancia sí que las utiliza. Dado que una de las ventajas de la radiancia es que nos permite obtener el resto de medidas radiométricas, ¿por qué no desarrollamos una nueva expresión de la irradiancia?

Para obtener cuánta luz llega a un punto, debemos acumular la radiancia incidente que nos llega desde cualquier dirección.

TODO: dibujo como el de la libreta roja. Me lo mandé por Telegram, por si no lo encuentro

Dado un punto p que se encuentra en una superficie con normal \mathbf{n} en dicho punto, la irradiancia se puede expresar como

$$E(p, \mathbf{n}) = \int_{\Omega} L_i(p, \omega) |\cos \theta| d\omega \quad (4.3)$$

El término $\cos \theta$ aparece en la integral debido a la derivada del área proyectada, dA^\perp . θ es el ángulo entre la dirección ω y la normal \mathbf{n} .

Generalmente, la irradiancia se calcula únicamente en el hemisferio de direcciones asociado a la normal en el punto, $H^2(\mathbf{n})$.

Podemos eliminar el $\cos \theta$ de la integral mediante una pequeña transformación: proyectando el ángulo sólido sobre el disco alrededor del punto p con normal \mathbf{n} , obtenemos una expresión más sencilla: como $d\omega^\perp = |\cos \theta| d\omega$, entonces

$$E(p, \mathbf{n}) = \int_{H^2(\mathbf{n})} L_i(p, \omega) d\omega^\perp$$

Usando lo que aprendimos sobre la derivada de los ángulos sólidos [4.1], se puede reescribir la ecuación anterior como

$$E(p, \mathbf{n}) = \int_0^{2\pi} \int_0^{\pi/2} L_i(p, \theta, \phi) \cos \theta \sin \theta d\theta d\phi$$

Haciendo el mismo juego con el flujo emitido de un cierto objeto al hemisferio que encapsula la normal, conseguimos:

$$\begin{aligned} \Phi &= \int_A \int_{H^2(\mathbf{n})} L_o(p, \omega) \cos \theta d\omega dA = \\ &= \int_A \int_{H^2(\mathbf{n})} L_o(p, \omega) d\omega^\perp dA \end{aligned}$$

TODO: a lo mejor merece la pena hacer un ejemplo sobre los diferentes tipos de luz, como en <https://cs184.eecs.berkeley.edu/public/sp22/lectures/lec-11-radiometry-and-photometry/lec-11-radiometry-and-photometry.pdf> p.41? O a lo mejor un capítulo para hablar de luces en general.

4.3 Integrando sobre área

Una herramienta más que nos vendrá bien será la capacidad de convertir integrales sobre direcciones en integrales sobre área. Hemos hecho algo similar en las secciones anteriores, así que no perdemos nada por generalizarlo.

Considera un punto p sobre una superficie con normal en dicho punto \mathbf{n} . Supongamos que tenemos una pequeña área dA con normal \mathbf{n}_{dA} . Sea θ el ángulo entre \mathbf{n} y \mathbf{n}_{dA} , y r la distancia entre p y dA .

Entonces, la relación entre la diferencial de un ángulo sólido y la de un área es

$$d\omega = \frac{dA \cos \theta}{r^2}$$

TODO: figura como la de pbr book 5.16.

Esto nos permite, por ejemplo, expandir algunas expresiones como la de la irradiancia [4.3] si partimos de un cuadrilátero dA :

$$\begin{aligned} E(p, \mathbf{n}) &= \int_{\Omega} L_i(p, \omega) |\cos \theta| d\omega = \\ &= \int_A L \cos \theta \frac{\cos \theta_o}{r^2} dA \end{aligned}$$

siendo θ_o la radiancia de salida de la superficie del cuadrilátero.

4.4 Dispersión de luz: las familias de funciones de distribución bidireccionales

Cuando una fuente de luz emite fotones hacia una superficie impactando en ella, ocurren un par de sucesos: parte de la luz se refleja en ella, saliendo disparada hacia alguna dirección; mientras que otra parte se absorbe.

En este capítulo vamos a modelar la primera. Estudiaremos qué es lo que ocurre cuando los fotones alcanzan una superficie, en qué dirección se reflejan, y cómo cambia el comportamiento dependiendo de las propiedades del material.

4.4.1 La función de distribución de reflectancia bidireccional (BRDF)

La **función de distribución de reflectancia bidireccional** (en inglés, *bidirectional reflectance distribution function*, BRDF) describe cómo la luz se refleja en una superficie opaca. Se encarga de informarnos sobre cuánta radiancia sale en dirección ω_o debido a la radiancia incidente desde la dirección ω_i , partiendo de un punto p en una superficie con normal \mathbf{n} .

TODO: esquema como el de pbr fig 5.18.

Si consideramos ω_i como un cono diferencial de direcciones, la irradiancia diferencial en p viene dada por

$$dE(p, \omega_i) = L_i(p, \omega_i) \cos \theta_i d\omega_i$$

Debido a esta irradiancia, una pequeña parte de radiancia saldrá en dirección ω_o , proporcional a la irradiancia:

$$dL_o(p, \omega_o) \propto dE(p, \omega_i)$$

Si lo ponemos en forma de cociente, sabremos exactamente cuál es la proporción de luz. A este cociente lo llamaremos $f_r(p, \omega_o, \omega_i)$; la función de distribución de reflectancia bidireccional:

$$f_r(p, \omega_o, \omega_i) = \frac{dL_o(p, \omega_o)}{dE(p, \omega_i)} = \frac{dL_o(p, \omega_o)}{L_i(p, \omega_i) \cos \theta_i d\omega_i} (1/\text{sr})$$

Las BRDFs físicamente realistas tienen un par de propiedades importantes:

1. **Reciprocidad:** para cualquier par de direcciones ω_i, ω_o , se tiene que $f_r(p, \omega_i, \omega_o) = f_r(p, \omega_o, \omega_i)$.
2. **Conservación de la energía:** La energía reflejada tiene que ser menor o igual que la incidente:

$$\int_{H^2(\mathbf{n})} f_r(p, \omega_o, \omega_i) \cos \theta_i d\omega_i \leq 1$$

4.4.2 La función de distribución de transmitancia bidireccional (BTDF)

Si la BRDF describe cómo se refleja la luz, la *bidirectional transmittance distribution function* (abreviada BTDF) nos informará sobre la transmitancia; es decir, cómo se comporta la luz cuando entra en un medio. Son caras de la misma moneda: cuando la luz impacta en una superficie, y parte de ella, se reflejará, y otra parte se transmitirá.

Denotaremos a la BTDF por

$$f_t(p, \omega_o, \omega_i)$$

Al contrario que en la BRDF, ω_o y ω_i se encuentran en hemisferios diferentes.

4.4.3 Juntando la BRDF y la BTDF

Convenientemente, podemos unir la BRDF y la BTDF en una sola expresión, llamada **la función de distribución de dispersión bidireccional** (*bidirectional scattering distribution function*, BSDF). A la BSDF la llamaremos

$$f(p, \omega_o, \omega_i)$$

Usando esta definición, podemos obtener

$$dL_o(p, \omega_o) = f(p, \omega_o, \omega_i) L_i(p, \omega_i) |\cos \theta_i| d\omega_i$$

NOTE: En wikipedia integran con respecto a ω_o , y no con la incidente. ¿Quizás afecte en algo? Además, el término $\cos \theta_i$ aparece en valor absoluto porque las normales no siempre están orientadas hacia fuera. ¿Podríamos omitirlo?

Esto nos deja a punto de caramelo una nueva expresión de la radiancia en términos de la radiancia incidente en un punto p . Integrando la expresión anterior, obtenemos

$$L_o(p, \omega_o) = \int_{\mathbb{S}^2} f(p, \omega_o, \omega_i) L_i(p, \omega_i) |\cos \theta_i| d\omega_i$$

siendo \mathbb{S}^2 la esfera.

Esta forma de expresar la radiancia es muy importante. Generalmente se le suele llamar la *ecuación de dispersión* (*scattering equation*, en inglés). Dado que es una integral muy importante, seguramente tengamos que evaluarla repetidamente. ¡Los métodos de Monte Carlo nos vendrán de perlas!

4.4.4 Subsurface scattering

NOTE: quizás no merezca la pena hablar de esto. De momento, se queda para recordar su existencia. No es tampoco difícil, pero si no lo vamos a usar...

Wikipedia contributors (2021b), StudySession (2021)

- https://www.wikiwand.com/es/Funci%C3%B3n_de_distribuci%C3%B3n_de_reflectancia_bidireccional
- <https://www.wikiwand.com/en/Transmittance>
- <https://cs184.eecs.berkeley.edu/public/sp22/lectures/lec-11-radiometry-and-photometry/lec-11-radiometry-and-photometry.pdf>

5 *Importance sampling*

- (*berkeley-cs184*) <https://cs184.eecs.berkeley.edu/public/sp22/lectures/lec-12-monte-carlo-integration/lec-12-monte-carlo-integration.pdf>
- <https://cs184.eecs.berkeley.edu/sp22>

6 Metodología; o cómo se hizo este trabajo

TODO - hablar de las fases de desarrollo. Interpretación propia de Agile. Documentación y código desarrollado a la par, mediante issues. Adaptación de los requisitos conforme se avanza. Beneficios de una página web (seguramente debería ser su propia sección)

6.1 Github

TODO - Hablar de cómo se utiliza Github y sus tecnologías para agrupar todo el trabajo. Hablar de la guía de estilos, y cómo los emojis ayudan a identificar rápidamente secciones.

6.1.1 Github Actions

TODO - Hablar de cómo se usa el sistema de integración continua para construir la web y el pdf

6.1.2 Github Projects

TODO - Hablar de cómo se gestiona el trabajo mediante issues, recapitulados todos con Projects.

7 Anexo: Glosario de términos y expresiones

It's dangerous to go alone, take this.

Tener en mente *todos* los conceptos y las expresiones que pueden tomar es prácticamente imposible. Tampoco hay necesidad de ello, realmente. Por eso, aquí tienes recopilada una lista con las fórmulas y enlace a las secciones correspondientes.

7.1 Notación

Concepto	Notación
Puntos	Letras mayúsculas: P, Q, \dots
Escalares	Letras minúsculas: a, b, c, k, \dots
Vectores	Letras minúsculas en negrita: $\mathbf{v}, \mathbf{w}, \mathbf{n}, \dots$. Si están normalizados, se les pone gorrito (por ejemplo, $\hat{\mathbf{n}}$)
Matrices	Letras mayúsculas en negrita: \mathbf{M} . Por columnas.
Producto escalar	$\mathbf{v} \cdot \mathbf{w}$. Si es el producto escalar de un vector consigo mismo, a veces pondremos \mathbf{v}^2
Producto vectorial	$\mathbf{v} \times \mathbf{w}$

Concepto	Notación
Variables aleatorias	X, Y . ξ representa una variable aleatoria con distribución uniforme en $[0, 1)$.

7.2 Radiometría

Concepto	Expresiones
Ángulo sólido, derivada [4.1]	$\omega = \frac{A}{r^2}$ $d\omega = \sin \theta \, d\theta \, d\phi$
Hemisferio de direcciones alrededor de un vector	$H^2(\mathbf{n})$
Carga de energía	$Q = hf = \frac{hc}{\lambda}$
Flujo radiante, potencia	$\Phi = \frac{dQ}{dt}$
Irradiancia, radiancia emitida	$E = \frac{\Phi}{A}$ $E(p) = \frac{d\Phi}{dA}$ $E(p, \mathbf{n}) = \int_{\Omega} L_i(p, \omega) \cos \theta d\omega$ $E(p, \mathbf{n}) = \int_0^{2\pi} \int_0^{\pi/2} L_i(p, \theta, \phi) \cos \theta \sin \theta \, d\theta \, d\phi$
Intensidad radiante	$I = \frac{d\Phi}{d\omega}$

Concepto	Expresiones
	$L(p, \omega) = \frac{dE_{\omega}(p)}{d\omega}$
Radiancia	$L = \frac{d\Phi}{d\omega dA^{\perp}}$
	$L^{+}(p, \omega) = \lim_{t \rightarrow 0^{+}} L(p + t\mathbf{n}_{\mathbf{p}}, \omega)$
	$L^{-}(p, \omega) = \lim_{t \rightarrow 0^{-}} L(p + t\mathbf{n}_{\mathbf{p}}, \omega)$
Radiancia incidente	$L_i(p, \omega) = \begin{cases} L^{+}(p, -\omega) & \text{si } \omega \cdot \mathbf{n}_{\mathbf{p}} > 0 \\ L^{-}(p, -\omega) & \text{si } \omega \cdot \mathbf{n}_{\mathbf{p}} < 0 \end{cases}$
Radiancia de salida	$L_o(p, \omega) = \begin{cases} L^{+}(p, \omega) & \text{si } \omega \cdot \mathbf{n}_{\mathbf{p}} > 0 \\ L^{-}(p, \omega) & \text{si } \omega \cdot \mathbf{n}_{\mathbf{p}} < 0 \end{cases}$

Bibliografía

- Alfonso Romero Sarabia. 2021. *Apuntes de La Asignatura Curvas y Superficies*.
- Berkeley cs184. n.d. “Monte Carlo Integration Cs184/284a.” Accessed March 20, 2022. <https://cs184.eecs.berkeley.edu/sp22/lecture/12/monte-carlo-integration>.
- Caulfield, Brian. 2020. “What’s the Difference Between Ray Tracing, Rasterization?” May 22, 2020. <https://blogs.nvidia.com/blog/2018/03/19/whats-difference-between-ray-tracing-rasterization/>.
- Galvin. n.d. “Random Variables.” Accessed March 20, 2022. https://www3.nd.edu/~dgalvin1/10120/10120_S16/Topic17_8p4_Galvin_class.pdf.
- Haines, Eric, and Tomas Akenine-Möller, eds. 2019. “Ray Tracing Gems.” Apress. 2019. <http://raytracinggems.com>.
- Owen, Art B. 2013. *Monte Carlo Theory, Methods and Examples*.
- Pharr, Matt, Wenzel Jakob, and Greg Humphreys. 2016. “Physically Based Rendering: From Theory to Implementation (3rd Ed.).” San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. November 2016. <https://www.pbr-book.org/3ed-2018/contents>.
- “Rendering.” n.d. Accessed March 20, 2022. <https://sciencebehindpixar.org/pipeline/rendering#:~:text=They%20said%20it%20takes%20at,to%20render%20that%20many%20frames>.
- Shirley, Peter, and R. Keith Morley. 2003. *Realistic Ray Tracing*. 2nd ed. USA: A. K. Peters, Ltd.
- StudySession. 2021. “Solid Angle Derivation & Intuition.” March 4, 2021. <https://www.youtube.com/watch?v=WksgBElPWA>.

- Wikipedia colaboradores. 2021. “Variable Aleatoria.” August 30, 2021. https://es.wikipedia.org/wiki/Variable_aleatoria.
- . 2022a. “Función de Probabilidad.” February 10, 2022. https://es.wikipedia.org/wiki/Funci%C3%B3n_de_probabilidad.
- . 2022b. “Método de La Transformada Inversa.” February 20, 2022. https://es.wikipedia.org/wiki/M%C3%A9todo_de_la_transformada_inversa.
- . 2022c. “Distribución de Probabilidad.” February 28, 2022. https://es.wikipedia.org/wiki/Distribuci%C3%B3n_de_probabilidad.
- . 2022d. “Estimador.” March 14, 2022. <https://es.wikipedia.org/wiki/Estimador?oldformat=true>.
- Wikipedia contributors. 2021a. “Rendering Equation.” March 9, 2021. https://en.wikipedia.org/wiki/Rendering_equation.
- . 2021b. “Radiometry.” September 12, 2021. <https://en.wikipedia.org/wiki/Radiometry>.
- . 2021c. “Parametric Surface.” December 15, 2021. https://en.wikipedia.org/wiki/Parametric_surface.
- . 2022a. “Rejection Sampling.” February 4, 2022. https://en.wikipedia.org/wiki/Rejection_sampling.
- . 2022b. “Ray Tracing (Graphics).” March 7, 2022. [https://en.wikipedia.org/wiki/Ray_tracing_\(graphics\)](https://en.wikipedia.org/wiki/Ray_tracing_(graphics)).
- . 2022c. “Rendering (Computer Graphics).” March 8, 2022. [https://en.wikipedia.org/wiki/Rendering_\(computer_graphics\)](https://en.wikipedia.org/wiki/Rendering_(computer_graphics)).
- . 2022d. “Computer.” March 13, 2022. <https://en.wikipedia.org/wiki/Computer>.
- . 2022e. “Implicit Surface.” March 13, 2022. https://en.wikipedia.org/wiki/Implicit_surface.
- . 2022f. “Barycentric Coordinate System.” March 14, 2022. https://en.wikipedia.org/wiki/Barycentric_coordinate_system.

- . 2022g. “Expected Value.” March 14, 2022. https://en.wikipedia.org/wiki/Expected_value.
- . 2022h. “Kodak.” March 14, 2022. <https://es.wikipedia.org/wiki/Kodak>.
- . 2022i. “Probability Density Function.” March 17, 2022. https://en.wikipedia.org/wiki/Probability_density_function.
- . 2022j. “History of Photography.” March 18, 2022. https://en.wikipedia.org/wiki/History_of_photography.