The Mathematics of Interior Integral Derivative

Zichen Wang

July 19, 2023

1 Introduction

In the *path integral formulation* [1], the recursive integrals of the light transport equation are expanded to a single integral over the path space

$$I = \int_\Omega g(\vec{x})\ d\mu(\vec{x})$$

where Ω denotes the path space, $\vec{x} = (x_0,\ x_1,\ ...,\ x_k,\ z) \in \Omega$ denotes a light path, $g(\vec{x})$ is the path contribution of $\vec{x}$, and μ is some measure over the path space.

Zhao et. al. [2] later proposed the *differential path integral*

$$\frac{\partial I}{\partial p} = \int_\Omega \frac{\partial}{\partial p} g(\vec{x})\ d\mu(\vec{x}) + \int_\Gamma \langle n(\vec{x}), \frac{\partial \vec{x}}{\partial p} \rangle g(\vec{x})\ d\sigma(\vec{x})$$

where Γ is the boundary path space, $n(\vec{x})$ is the normal of $\vec{x}$ in Γ, Δg is the value difference on two sides of the boundary, and σ is a measure over Γ induced by μ.

2 Interior Integral

2.1 2D Example

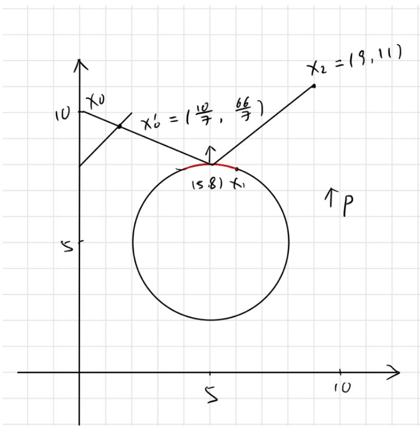

Figure 1: 2D Example

Consider a simple 2D scene consisting of a sphere at $(5, 5)$ with radius 3, a camera at $(0, 10)$ with its image plane from $(0, 8)$ to $(2, 10)$, and a point light at $(9, 11)$. The light integral is then

$$\begin{aligned} I &= \int_P W(x_0')Li(x_0', x_0 \leftarrow x_0')\ dx_0' \\ &= \int_P W(x_0')L_o(x_1, x_0 \leftarrow x_1)\ dx_0' \\ &= \int_P W(x_0')f(x_0 \leftrightarrow x_1 \leftrightarrow x_2)G(x_1 \leftrightarrow x_2)I(x_2)\ dx_0' \end{aligned}$$

where x_0 is the camera view point, x'_0 is the image plane sample, x_1 is the intersection point with the sphere, and x_2 is the point light; P is the image plane, W is the ray weight, f is the BSDF, G is the geometry term associated with the point light, and I is the point light intensity.

Suppose that we limit the integral to a single pixel so that $W = 1$, the BSDF is always 1 in all directions, and the light intensity is 1, then the integral simplifies to only the geometry term. The sphere now moves up by a single parameter p, and we want to compute the gradient of the geometry term with respect to the parameter.

On one hand, in this simple example we can solve analytically the gradient

$$\begin{aligned}\frac{\partial}{\partial p} G(x_1 \leftrightarrow x_2) &= \frac{\partial}{\partial p} \frac{(0,1) \cdot (x_2 - x_1)}{||x_2 - x_1||^3} \\ &= \frac{\partial}{\partial p} \frac{(0,1) \cdot (4, 3-p)}{(16 + (3-p)^2)^{1.5}} \\ &\approx 0.00064 \text{ when } p = 0\end{aligned}$$

On the other hand, note that

$$\frac{\partial}{\partial p} G(x_1 \leftrightarrow x_2) = \frac{\partial}{\partial x_1} G(x_1 \leftrightarrow x_2) \cdot \frac{\partial x_1}{\partial p}$$

Autodiff can handle $\partial G(x_1 \leftrightarrow x_2)/\partial x_1$ correctly, so we only need to figure out $\partial x_1/\partial p$. We can do so by replacing the gradient of x_1 to be $(0, p)$. This amounts to moving the intersection point together with the sphere. If we now forward parameter p to I, we will obtain the same value.

2.2 3D Case

In 3D cases, the general idea remains the same: replace the gradients of the intersection points to be $(0, p, 0)$ to move the paths with the sphere. The main difference is that in the 2D example, we consider the derivative of the path contribution with respect to the single parameter, but here we need to consider the derivative of the pixel value with respect to the parameter. This asks us to also compute the gradients of the image plane samples x'_0.

The light integral from above still holds here

$$\begin{aligned}I &= \int_P W(x'_0) L_o(x_1, x_0 \leftarrow x_1)\ dx'_0 \\ &\approx \frac{1}{N} \sum_i W(x_0'^{(i)}) L_o(x_1^{(i)}, x_0^{(i)} \leftarrow x_1^{(i)}) / P(x_0'^{(i)})\end{aligned}$$

Differentiate and we get

$$\frac{\partial I}{\partial p} \approx \frac{1}{N} \sum \frac{\frac{\partial}{\partial p} W \cdot L0 \cdot P + W \cdot \frac{\partial}{\partial p} L_o \cdot P - W \cdot L_o \cdot \frac{\partial}{\partial p} P}{P^2}$$

We are left to handle the three partial derivatives above and then we are done. Although during forward rendering we sample x'_0 first and then deterministically compute x_1, when computing gradients, the order is actually reversed: since we want the path to move with the sphere, we replace the gradient of x_1 first and then compute the gradient of x'_0 with respect to x_1.

Note that the partial derivative of L_o now only depends on x_1 but not x'_0. Thus we can follow a similar logic to the 2D example to obtain

$$\frac{\partial}{\partial p} L_o(x_1, x_0 \leftarrow x_1) = \frac{\partial}{\partial x_1} L_o(x_1, x_0 \leftarrow x_1) \cdot \frac{\partial x_1}{\partial p}$$

For the rest two, note that their primal values only depend on x'_0 but not x_1. However, since the gradient of x'_0 now depends on the gradient of x_1, we need one more step

$$\begin{aligned}\frac{\partial}{\partial p} W(x'_0) &= \frac{\partial}{\partial x'_0} W(x'_0) \cdot \frac{\partial x'_0}{\partial x_1} \cdot \frac{\partial x_1}{\partial p} \\ \frac{\partial}{\partial p} P(x'_0) &= \frac{\partial}{\partial x'_0} P(x'_0) \cdot \frac{\partial x'_0}{\partial x_1} \cdot \frac{\partial x_1}{\partial p}\end{aligned}$$

Finally, to compute $\partial x_0'/\partial x_1$, we project the intersection point x_1 back onto the image plane to get a new x_0' and transform it from world space to image space. We can compute the new x_0' by using the camera's FOV. It seems that in Mitsuba3, the y-axis of the image plane points downward, so we take the opposite of the y coordinate to transform to image space.

3 Results

Here we demonstrate that our interior gradients match well with the finite difference. We render a bunny and forward a single parameter that moves up the bunny to the rendered image. All images are under the same normalization, with grey for 0, pink for positive gradients, and dark blue for negative gradients. The silhouette pink gradients typically center around 1200, while in the difference image our maximum error is controlled within 100.

(a) finite diff (b) interior (c) finite diff - interior

Figure 2: Comparison with finite difference.