レイトレーシングと数学と物理

小野涼大

2024 年 10 月 17 日

概要

Ray Tracing in One Weekend(週末レイトレーシング) と言う書籍 (サイト？) ではレイトレーシングを C++ を使って簡単に実装する方法について解説されている．その中では当たり前だが多少の計算が必要で, プログラミングやレイトレーシングに興味を持った未来の高専生たちがいざ作ってみようと思っても (コード例が載っているので作ることはできるが) 完全に理解した状態で作るには説明が足りないと思われる．そこでその計算方法やその他屈折に関する興味深い事実をまとめた．

1　はじめに

最近のゲームや映画の映像はものによっては現実と区別がつかないほど高度な域にまで達している (特に金属や機械など)．商業的に使われているものほど細かいものは難しいが, 単純な金属球やガラス球といったものを現実のような美しさで 1 から描画するのは実はさほど難しくない．

レイトレーシングという 3D CG の描画手法を使えば (プログラミングを実行できる環境を持っていれば) 誰でも簡単に以下のような画像を生成させることができる．印刷の関係で白黒になっているだろうが美しい画像を生成できることは確認できるだろう．念を押すが誰でも Blender や Unreal Engine といったソフトウェアを使わずに 3D CG を描画できるのだ．

こちらではこれの作り方は説明しない．既に Ray Tracing in One Weekend という書籍 (サイト？) で作り方を無料公開しているからだ．こちらではそのサイトでは説明不足に感じた光の屈折の計算方法とそれに関する蛇足情報をまとめてある．

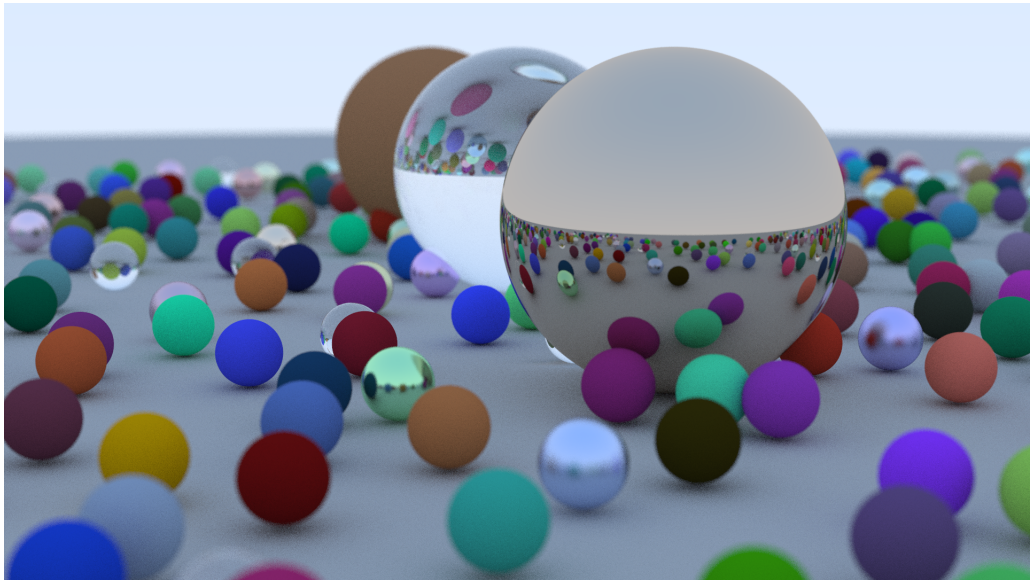

図1: 実際に描画した画像

2　光の屈折とは

この宇宙で最速 (量子もつれ[*1]とかは無視) の光くんは” 真空中” で速度 $c \simeq 300,000\mathrm{km/s}$ とされている. これは 1 秒で地球を 7 周半できるほどの速さである [*2]. しかしこれは真空中の話で他の物質中 (空気, 水, ガラス) を通っているときは速度が落ちる. 光は粒子” でも” あるので実際に進むときは図のように横 1 列になって進んでいると考えてほしい.

このように斜めに進んでいると他の物質に入るのに時間差がうまれる. この速度差と時間差が影響して人の目からは光が他の物質へ入るときに屈折するように見えるのだ.

ところで適当に流していたが, そもそもなぜ「他の物質中 (空気, 水, ガラス) を通っているときは速度が落ちる」のだろう.

3　屈折光の計算

実際に屈折光をスネルの法則を中心に計算していくが計算過程でいくつか高校数学で習うものを使う. といってもちょろいので簡単な説明をしておく.

3.1　必要になる知識の説明

三角比

図のような直角三角形がある. $\angle \mathrm{ABC} = \theta$ とする. それぞれ向かい合う辺を a, b, c とすると三角比は以下のように定義される.

$$\sin\theta = \frac{b}{c} \qquad \cos\theta = \frac{a}{c} \qquad \tan\theta = \frac{b}{a}$$

なお今回使うのは sin だけである. また定義から明らかなことではあるが斜辺の長さをかければそれぞれ高さ, 底辺の長さを求めることができる.

ベクトル

今まで使ってきた長さや速さといったものは 1 つの値で表してきた. このようなものをスカラーと言う. これに方向といった情報を追加し, 複数の値を使って表すものをベクトルという. $\vec{a}$ や $\mathbf{a}$ などを使う. 今回は後者を使う. 格好良いから. ベクトルの演算方法については, 加算, 内積, 大きさの計算方法がわかっていれば良いので, その 3 つについて説明する.

加算については図のようなイメージを持ってもらえば良い. マイナスがついた場合方向が真逆になると思って欲しい.

内積は 2 つのベクトル $\mathbf{a}, \mathbf{b}$ のなす角を θ とするとき以下のように定義される.

$$\mathbf{a} \cdot \mathbf{b} = |\mathbf{a}||\mathbf{b}| \cos\theta$$

$|\mathbf{a}|$ は $\mathbf{a}$ の大きさを意味しており, イメージとしては図の通りである. 式を見てもわかるが内積で得られる値はスカラーである.

[*1] よくわかってないので自分で調べて

[*2] 速い, が宇宙の最速としては遅い, と思う

3.2 スネルの法則

3.3 全反射

3.4 フレネル反射率

4 最後に

一番最初に紹介したのに最後に紹介するのはあまり良くないかもしれないが, 最後に Ray Tracing in One Weekend について布教したいと思う. そもそも

参考文献

[1] Ray Tracing in One Weekend.