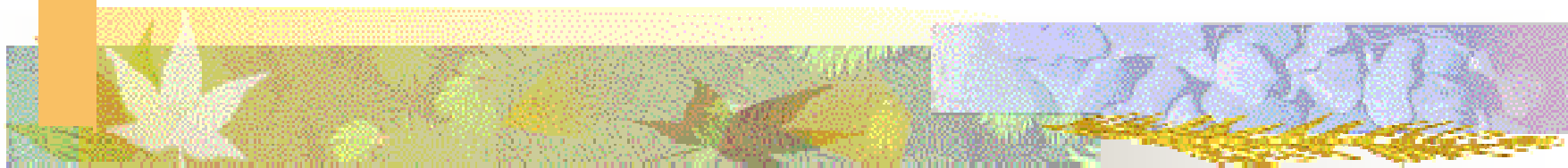


CGの光と陰



光あるところに陰がある

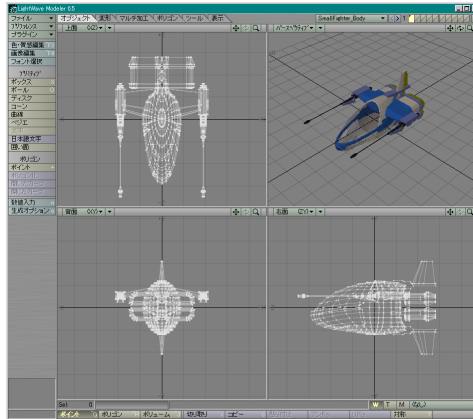
これを知ってればアニメオタク



コンピュータグラフィックス の2つの要素

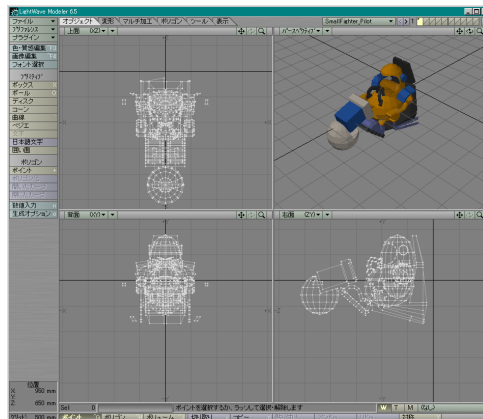
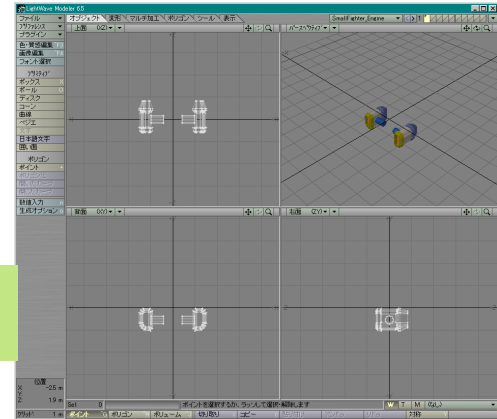
- モデリング
 - データを作成すること
- レンダリング
 - データから映像を生成すること

モデリング



パーツ1

パーツ2



パーツ2

配置



レンダリング



レンダリングの2つの方向

■ リアルタイムに映像を作成する

■ ゲーム等

- 「その場」で映像を生成する
- リアルタイムレンダリング

インタラクティブ
グラフィックス

■ リアルな映像を作成する

■ 映画等

- あらかじめ映像を生成しておく
- フォトリアリスティックレンダリング

プリレンダリング



リアルな映像の作成

- 物理現象の厳密なシミュレーション
 - 非常に時間がかかる
 - 泊り込み
 - レンダリングファーム
- 物に光を当てて初めて姿が見える
 - 照明効果は物理現象
- 形のリアルさ、動きのリアルさ
 - いずれも物理的なモデルから再現される

リアルタイム映像生成

■ 制限時間内に描画を完了する

■ 速度重視

■ 見かけのリアルさは2の次

- ポリゴン（形のデータ）をできるだけ減らすなど

■ 厳密にシミュレーションを行わなくても、それらしく見えればいい

- 陰影計算の手を抜くなど

■ むしろ動きのリアルさが重視される

■ モーションキャプチャ

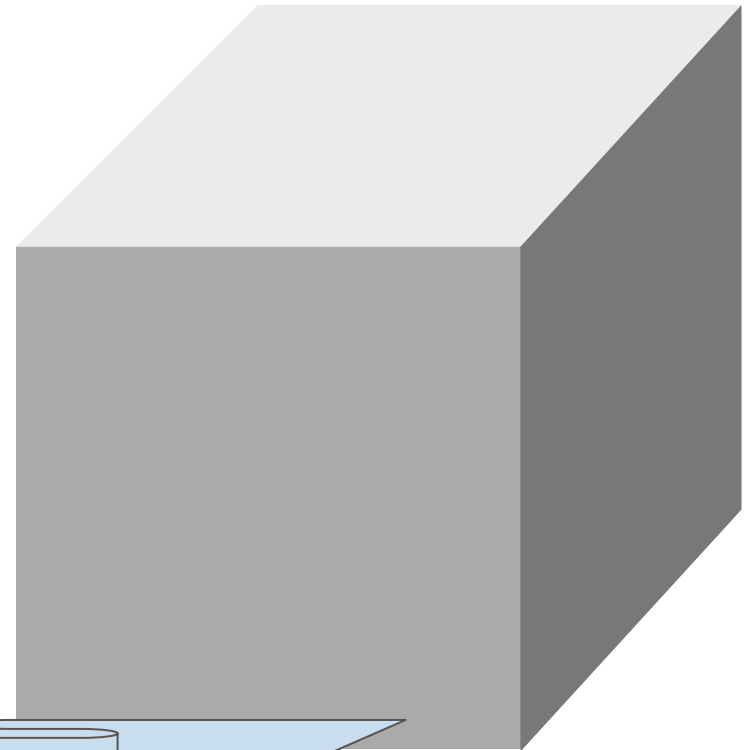
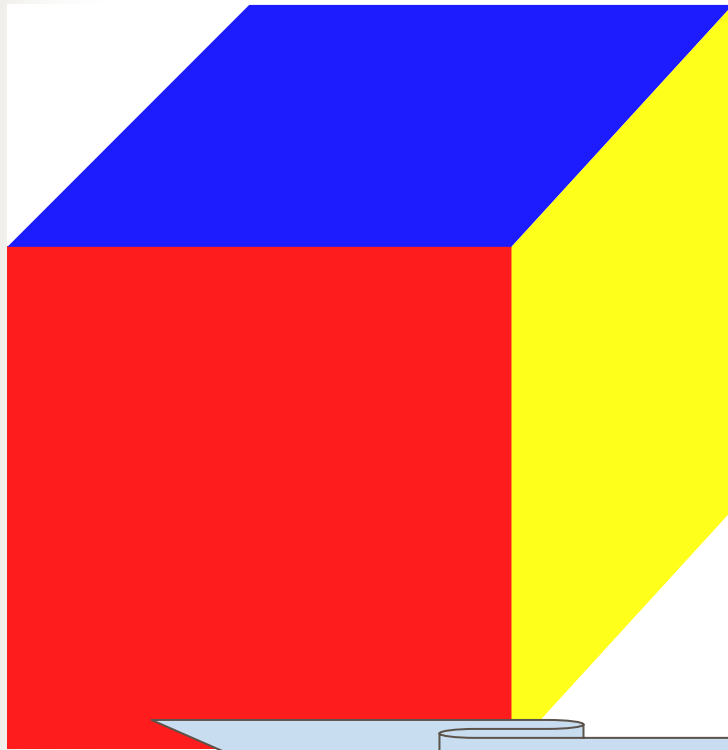
テクスチャで
ごまかす



リアルさとリアルタイム性

- **トレードオフが存在する**
 - リアルさを求めればリアルタイム性の実現が難しくなる
 - リアルタイム性を求めればリアルさを損なう結果になり得る
- **トレードオフの解消が求められている**
 - ゲームのリアリズムの向上
 - CGアニメーション映画のコストダウン

リアルさの基本：陰影



どちらが「リアル」？

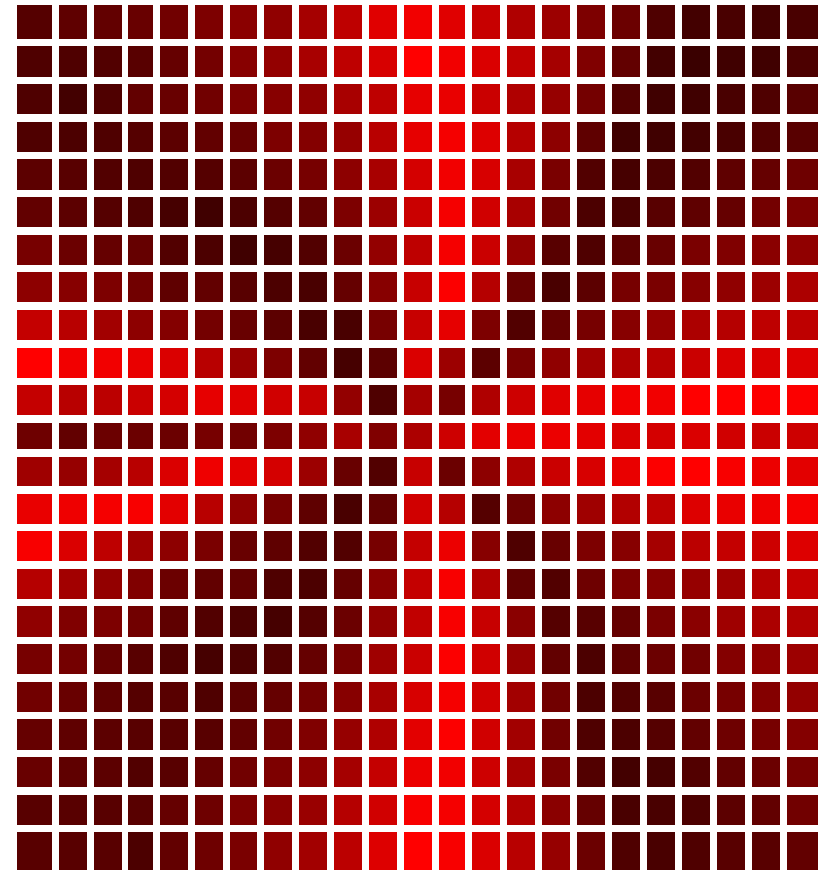


陰影画像の生成

- 3次元形状のスクリーンへの投影像を求める
- その投影像のディスプレイ上での色を計算する
- ディスプレイ全体について色を決定すれば目的の画像が得られる
 - デジタル画像

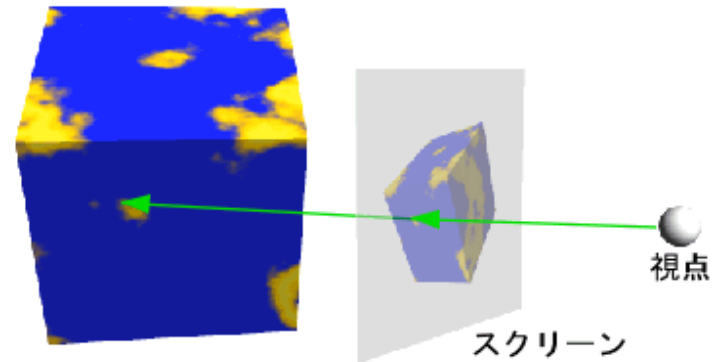
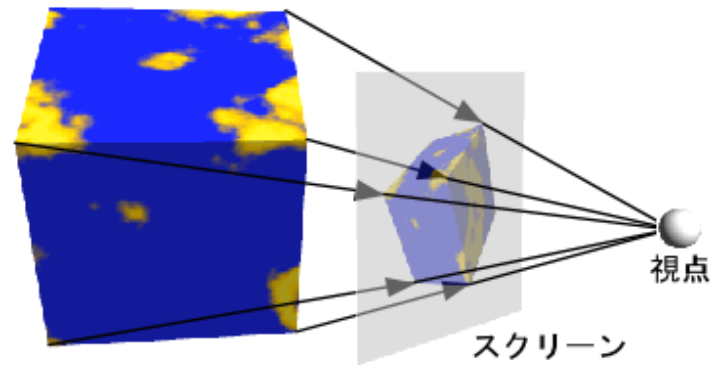
デジタル画像は光の点の集合

- 一つ一つの点の色を決めることで、図形や画像，映像を表現する
- この点のことを画素 (pixel) と言う
- CGの制作は，この一つ一つの画素の色を決める作業



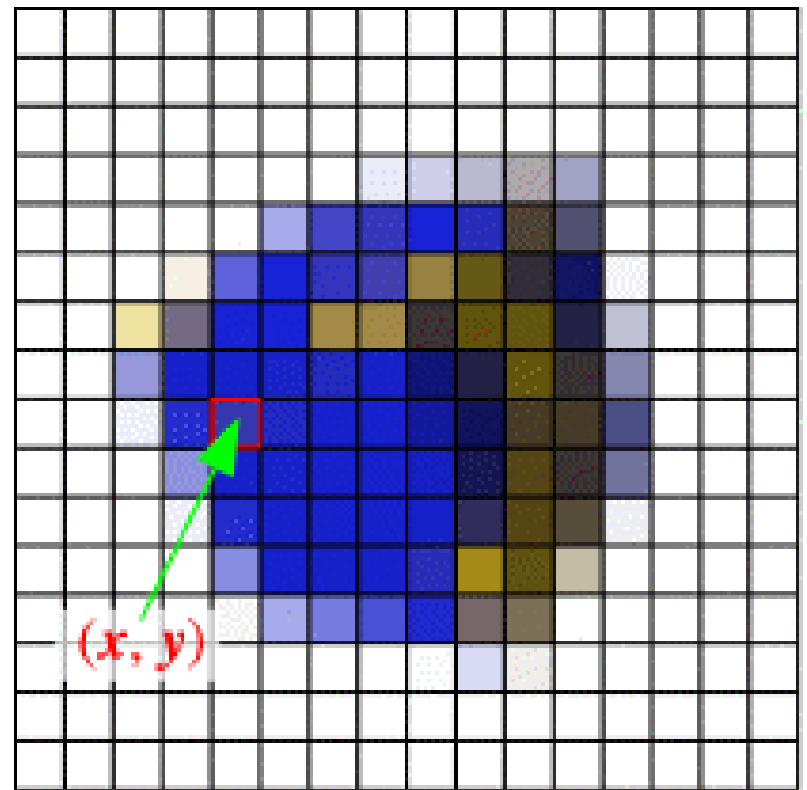
立体図形をスクリーンに投影

- 3次元CG画像は、立体図形の形状のデータをスクリーンに投影して得る
- 視点からスクリーン上のひとつの画素を見て、そこに何が見えるのかを調べる方法もある



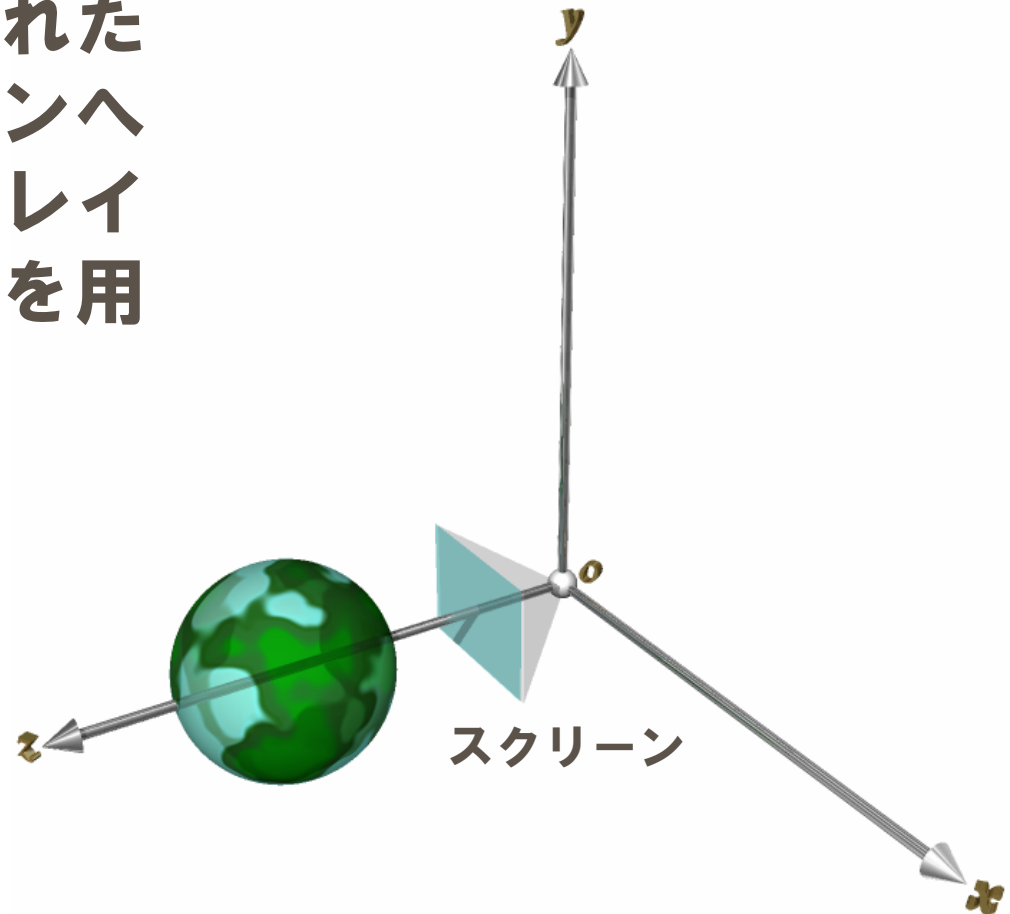
その画素の位置に見えるもの

- その画素のところに何が見えるのかわかれば，画素の色が決定できる
- レイトレーシング



球をライトレーシングで表示

- 空間中に置かれた球のスクリーンへの投影像を，ライトレーシングを用いて作成する



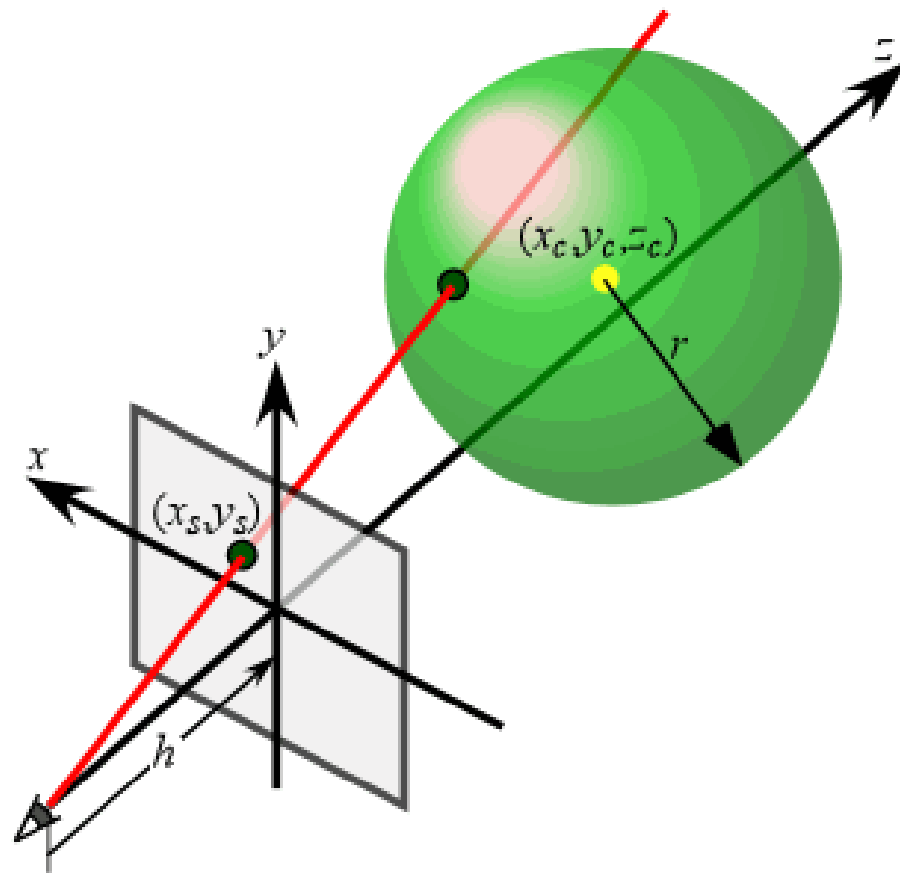
球の方程式

- 中心

- (x_c, y_c, z_c)

- 半径

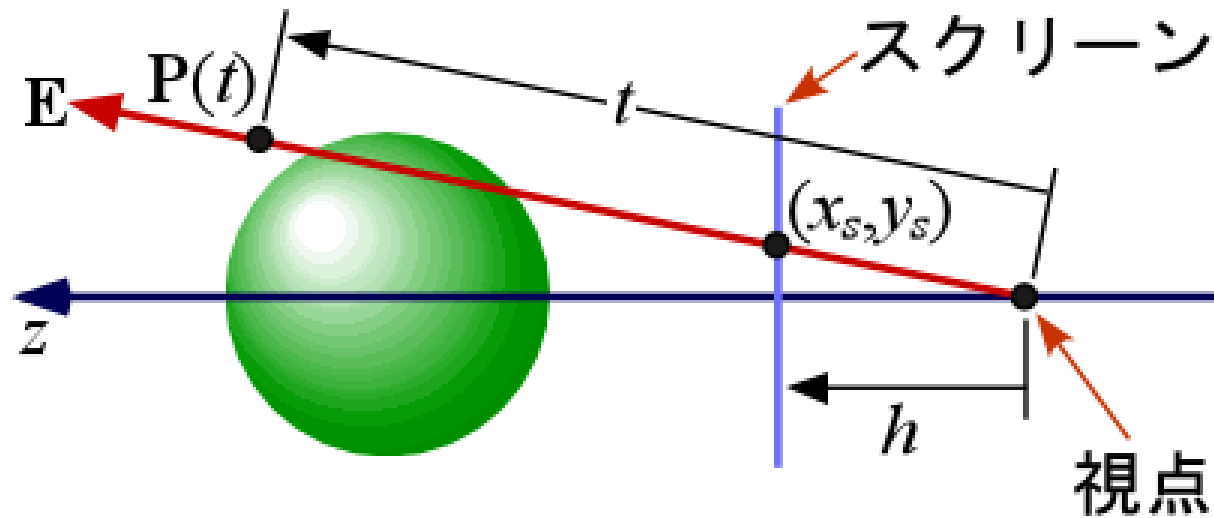
- r



$$(x - x_c)^2 + (y - y_c)^2 + (z - z_c)^2 = r^2$$

視線と球の交点を求める

- この球と，視点から出発して，色を決定したい画素の位置を通る半直線＝視線との交点を求める



視線の方程式

- 原点から出発し，そこから h の位置にあるスクリーン上の1点 (x_s, y_s) を通る半直線の方程式を媒介変数 t で表す

$$x = x_s t$$

$$y = y_s t$$

$$z = ht$$

視線を球の方程式に代入

$$At^2 - 2Bt + C = 0$$

$$A = x_s^2 + y_s^2 + h^2$$

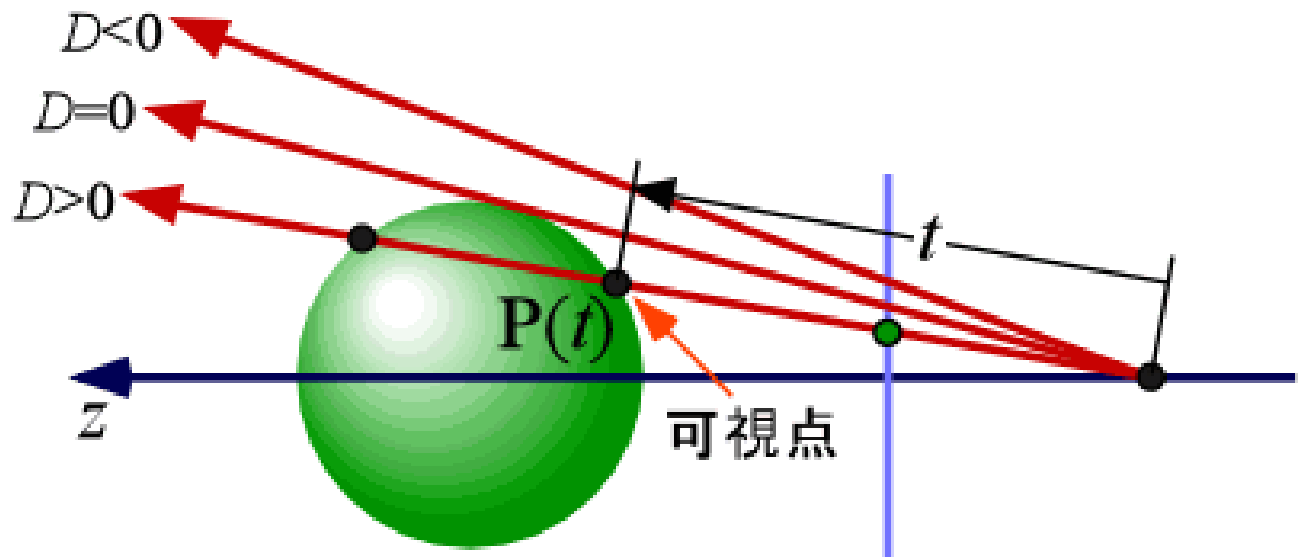
$$B = x_s x_c + y_s y_c + h z_c$$

$$C = x_c^2 + y_c^2 + z_c^2 - r^2$$

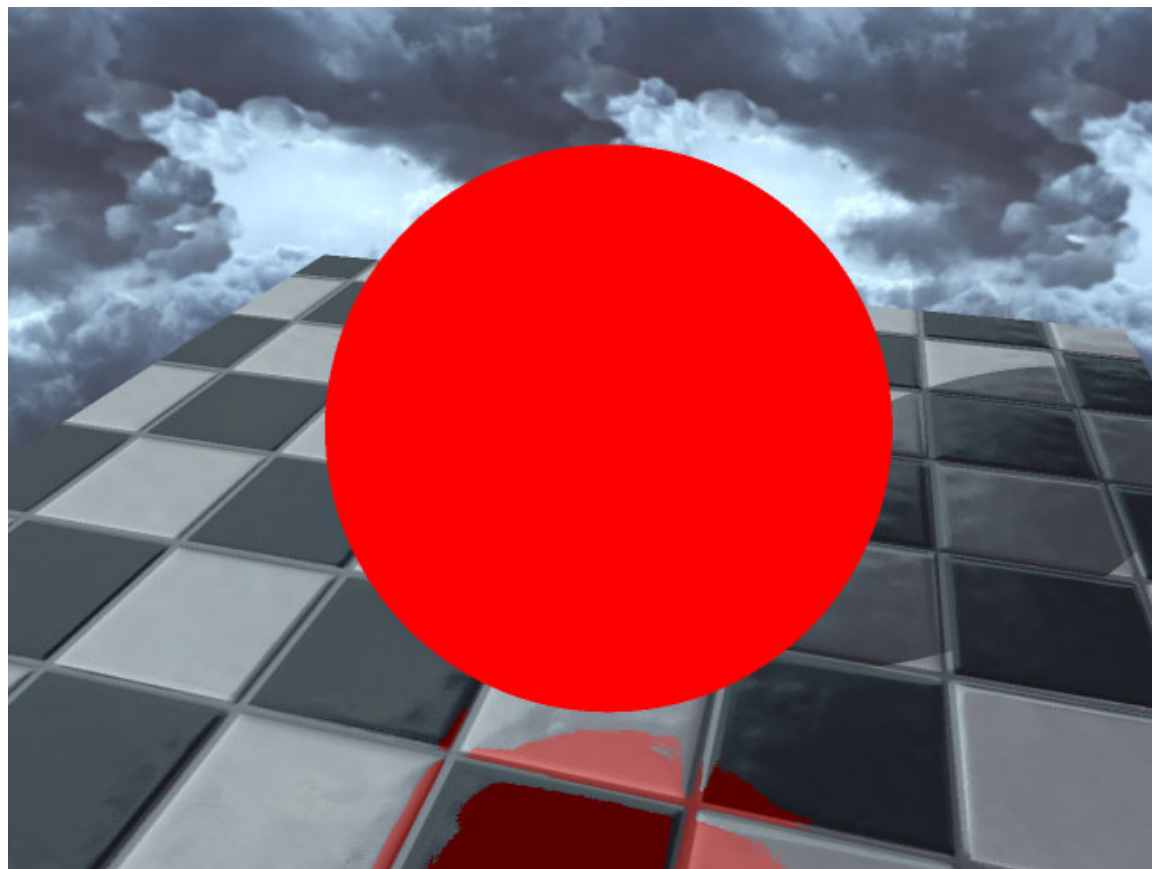
交差の検出

$$At^2 - 2Bt + C = 0$$

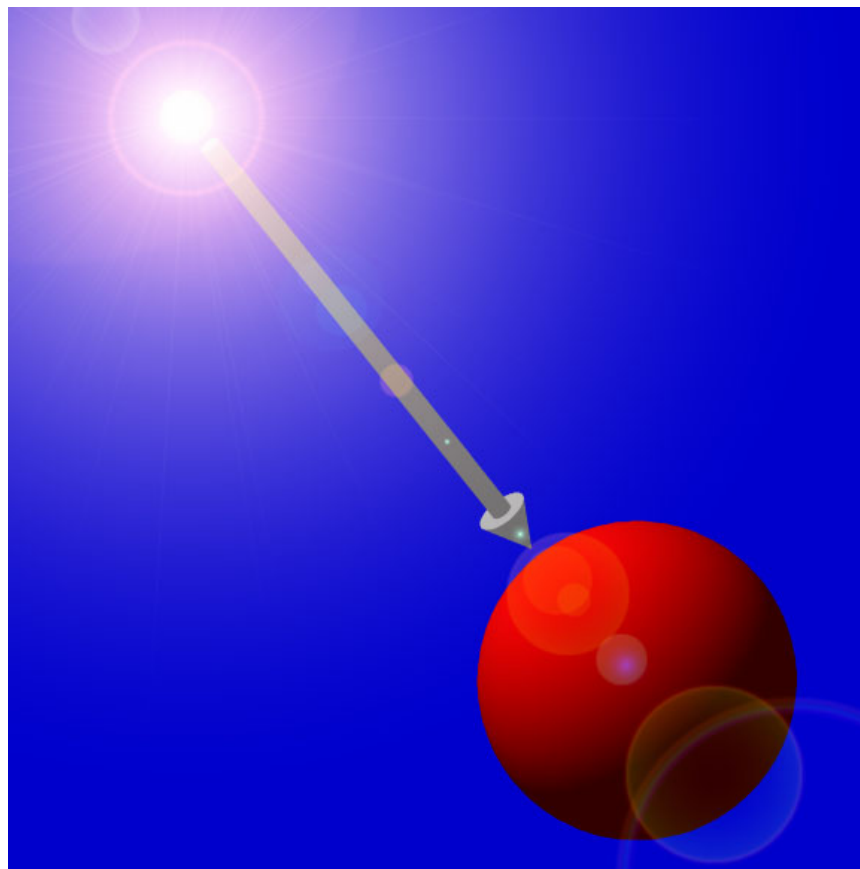
判別式 $D = B^2 - AC$



球と交差した画素を赤にする

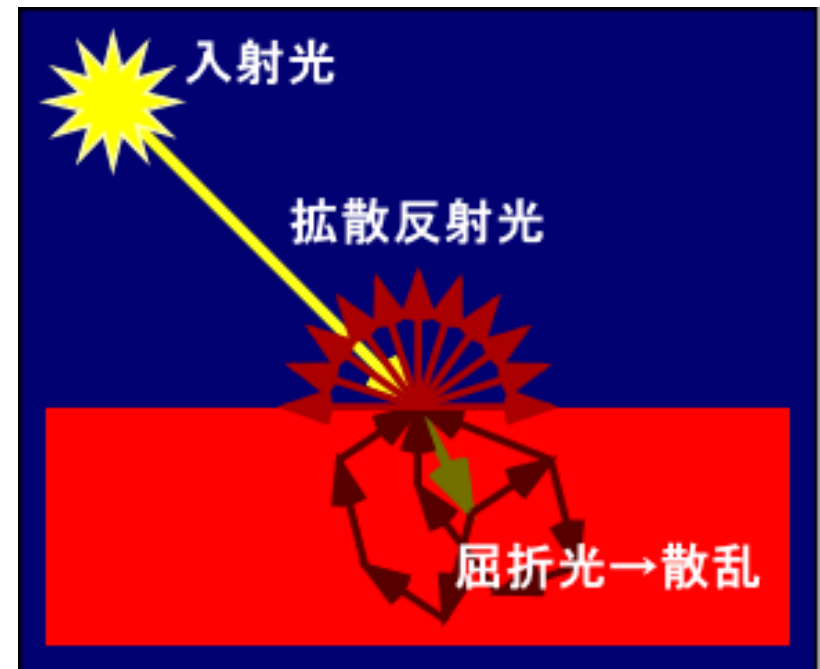


光を当ててみる

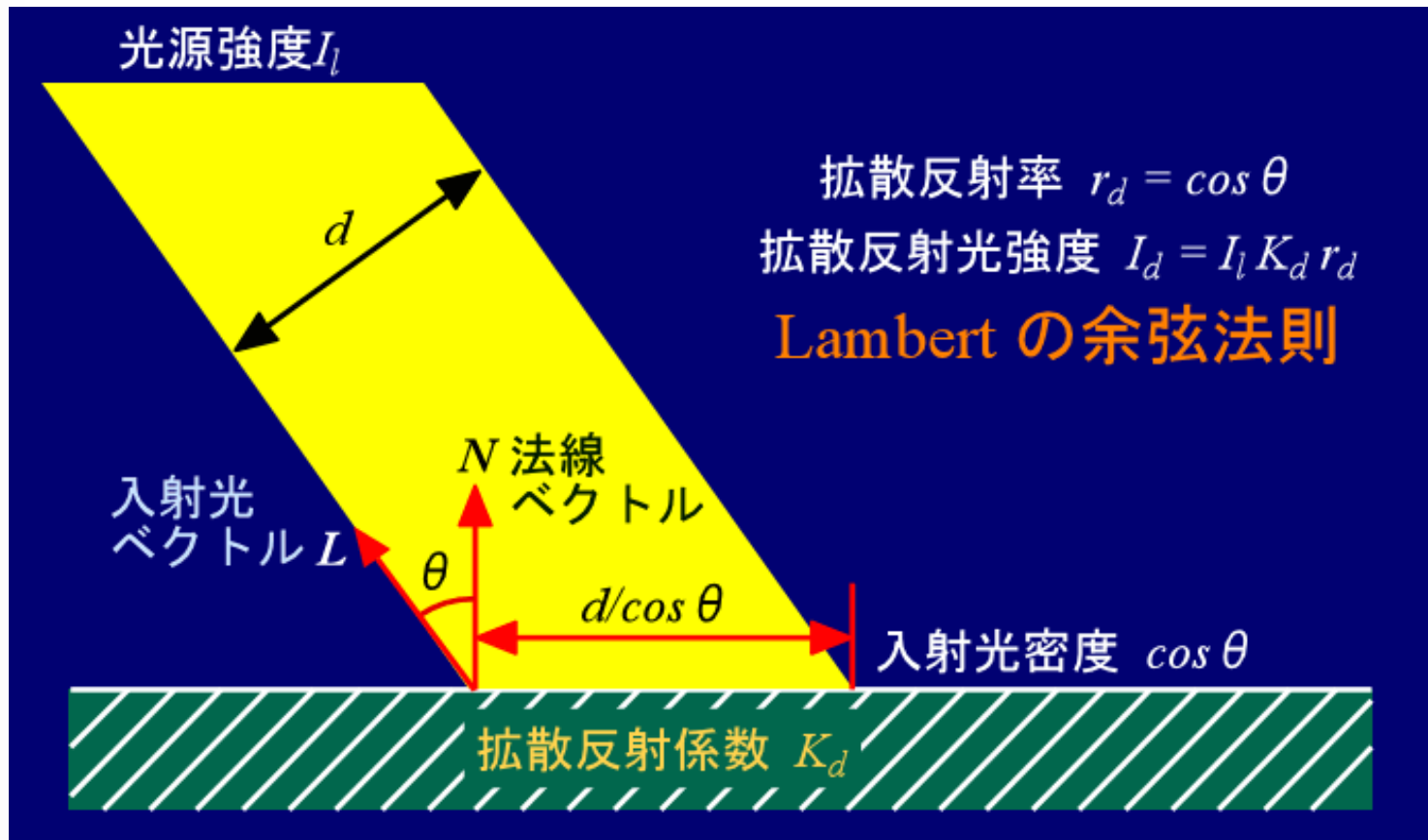


拡散反射光

- 完全拡散反射面
 - 全方向に対して均一に光を反射する
- 反射光強度
 - 入射光強度に比例
- 入射光強度
 - 入射光密度に比例

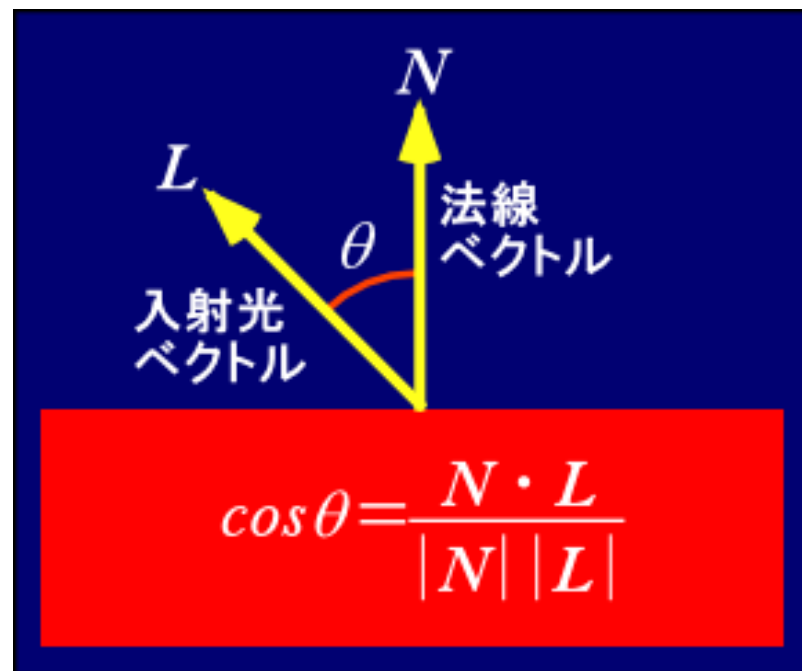
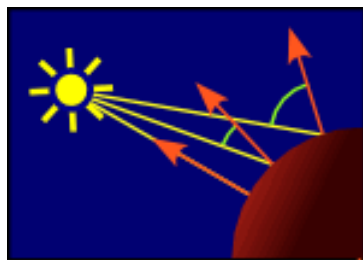


入射光の密度

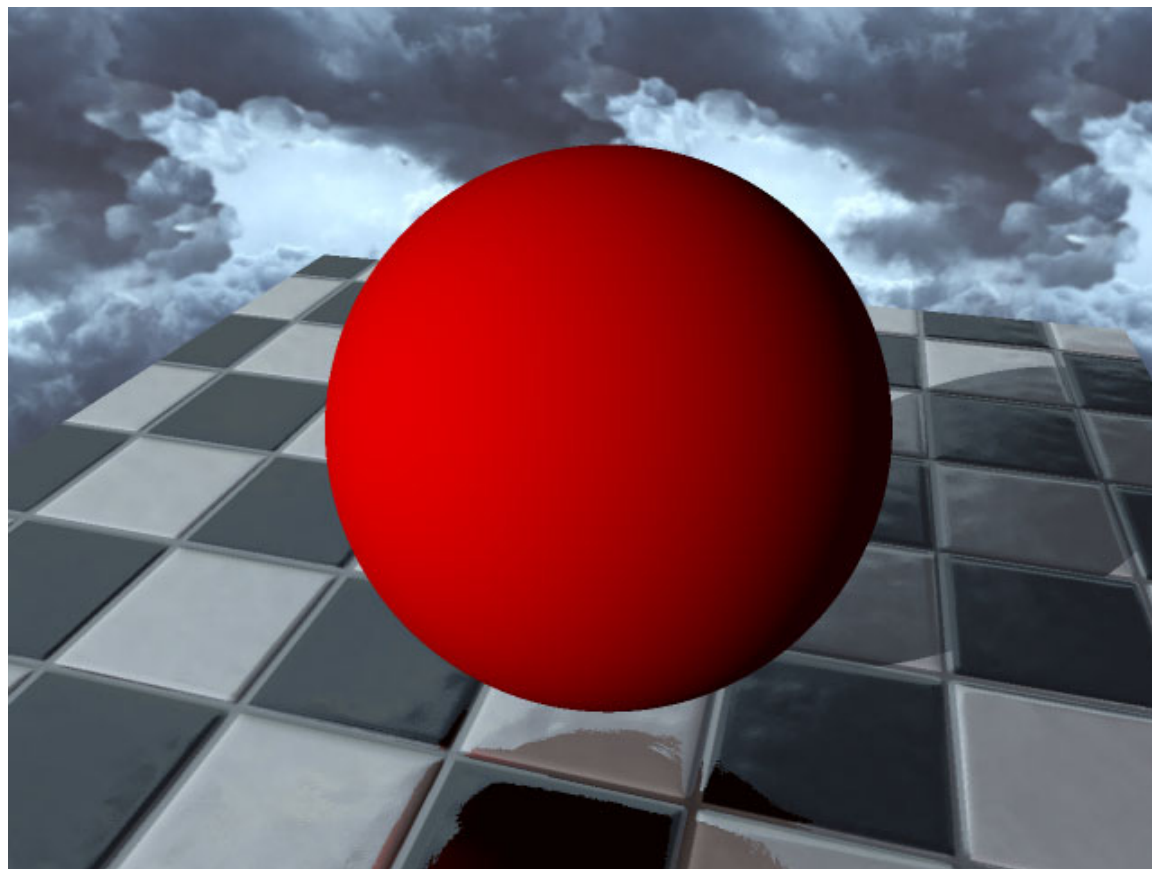


拡散反射光強度

- 入射光ベクトルと法線ベクトルの、内積に比例する
- 照明光の入射方向
 - 真正面→明るい
 - 横方向→暗い

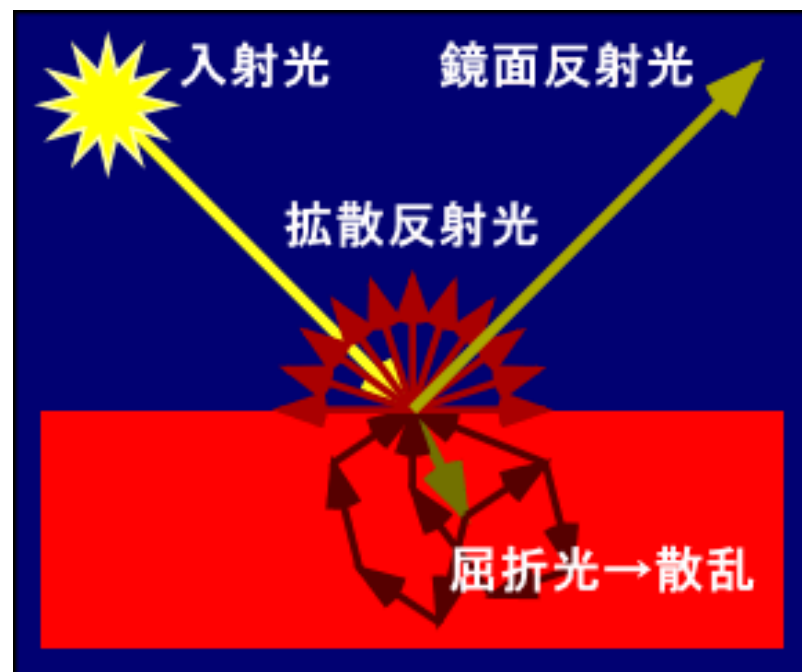


拡散反射光による陰影



ハイライトの表現

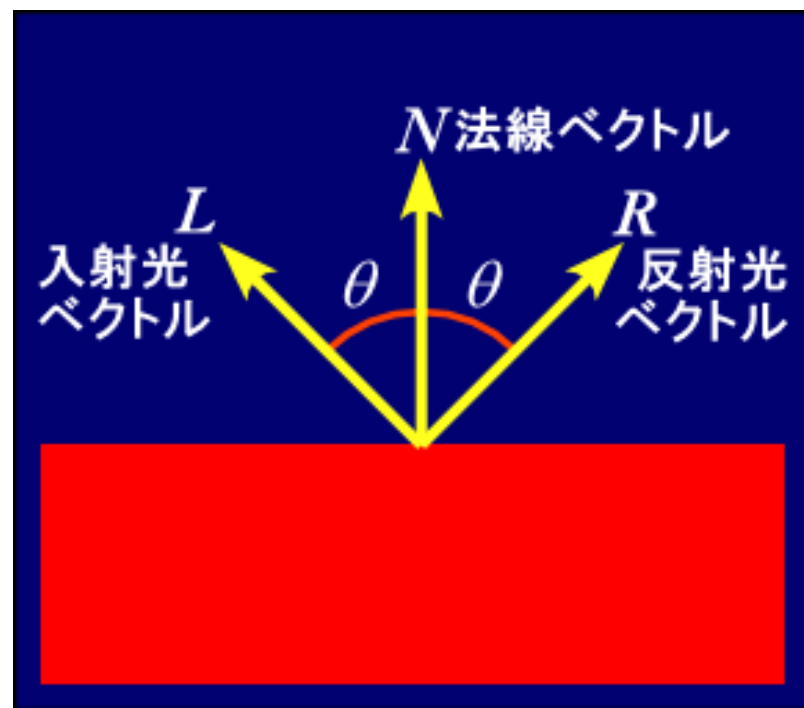
- 2色性反射モデル
 - 拡散反射成分
 - 鏡面反射成分
- 鏡面反射成分
 - ハイライト
 - 光源の映り込み



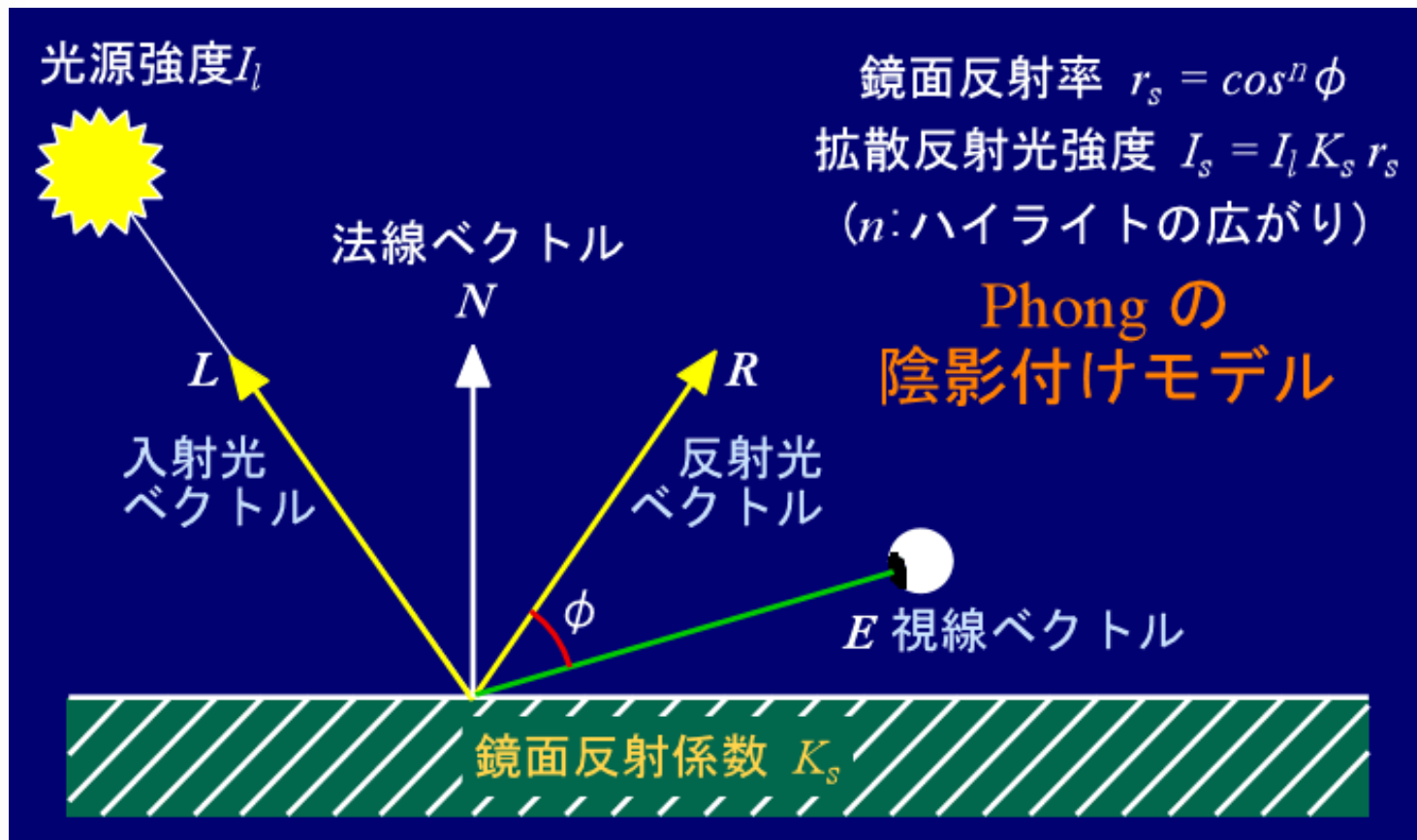
鏡面反射光

- 入射光の正反射光
 - 入射光ベクトル L
 - 法線ベクトル N
 - 反射光ベクトル R

$$R = 2(L \cdot N)N - L$$

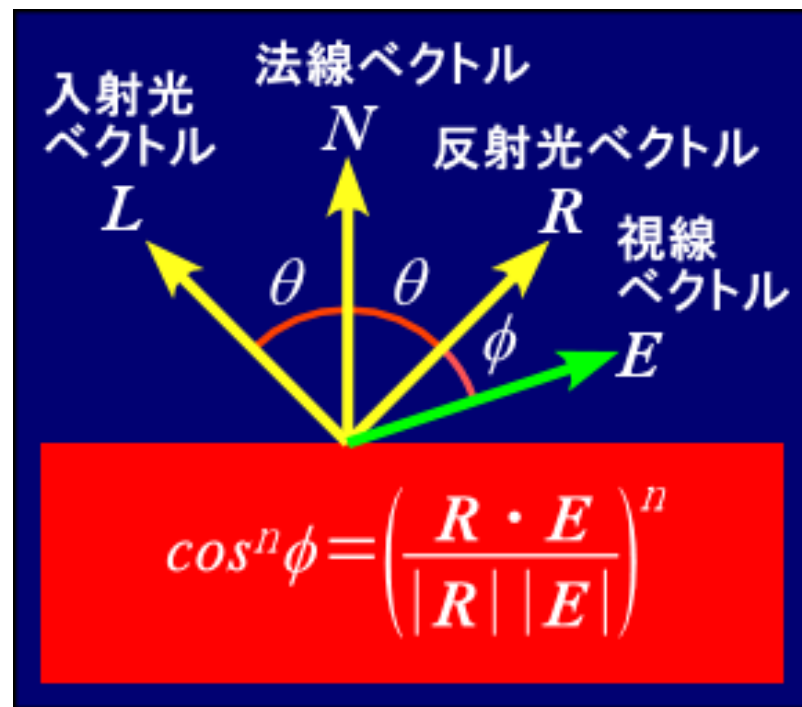


鏡面反射光の分布

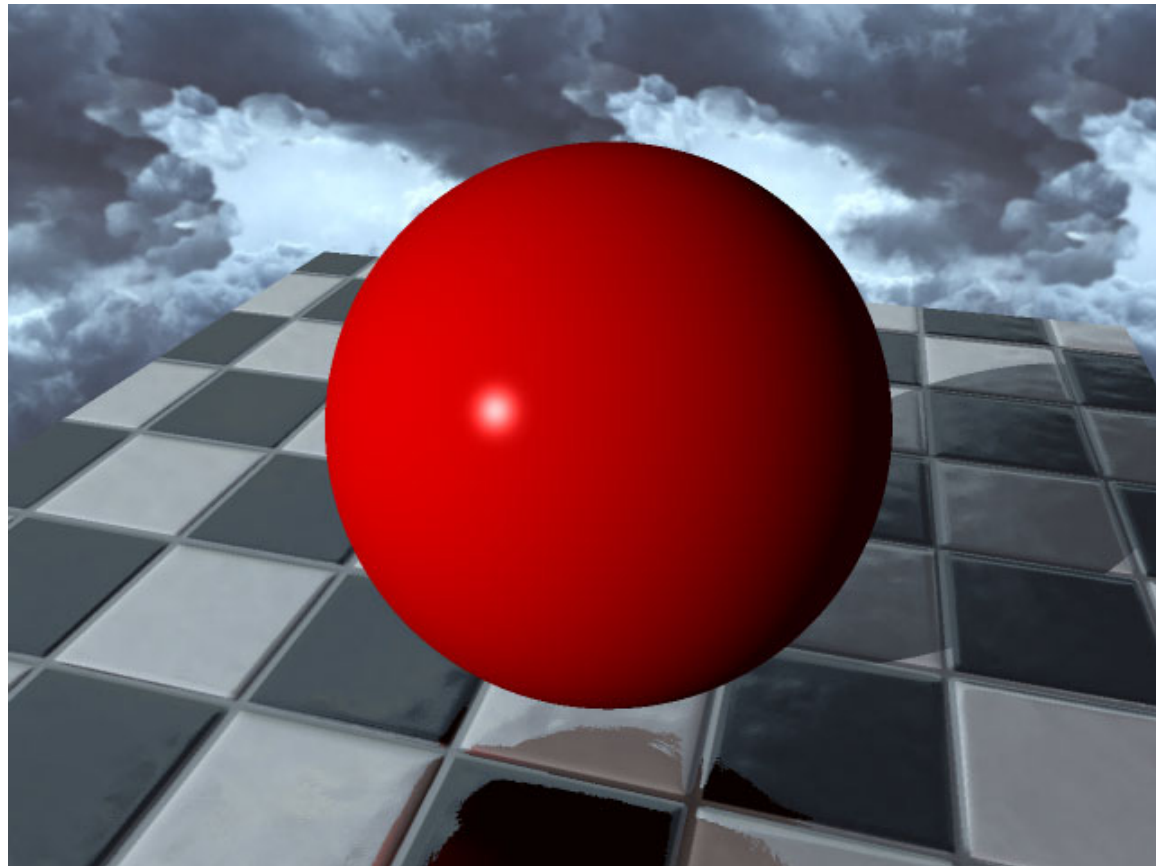


鏡面反射光強度

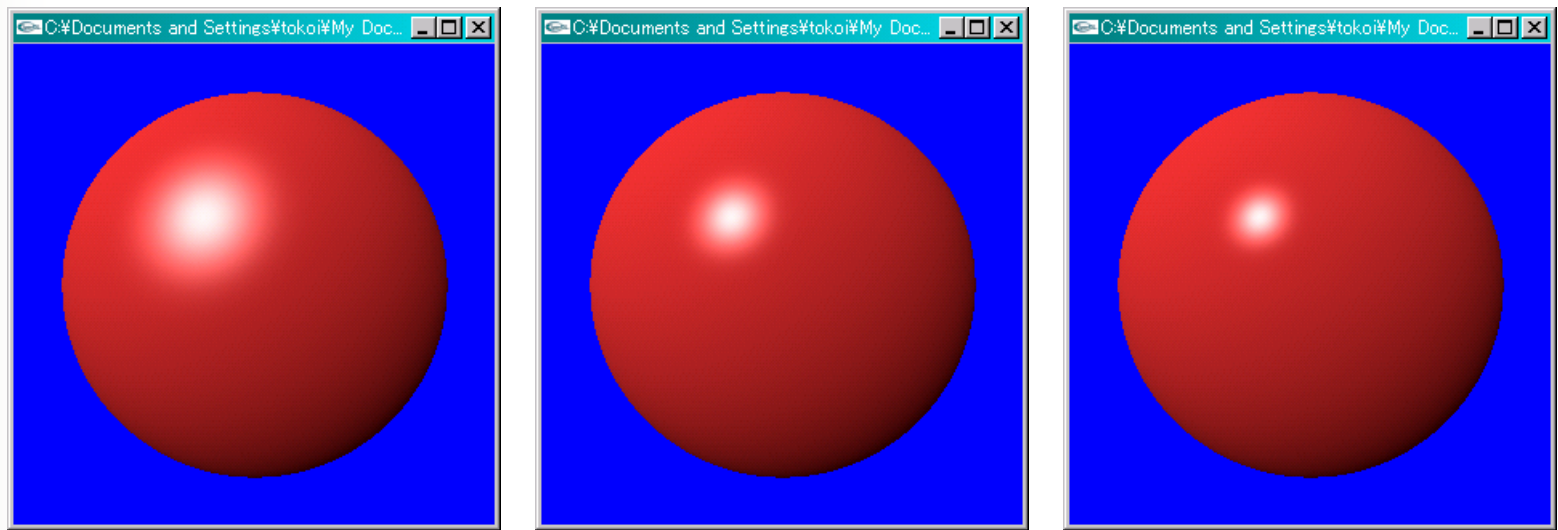
- 反射光ベクトルと視線ベクトルの、内積値の n 乗
- 光源は点光源
 - 面積をもたない
 - 平行光線も同じ
- 係数 n によってハイライトの広がりを制御する



鏡面反射光によるハイライト



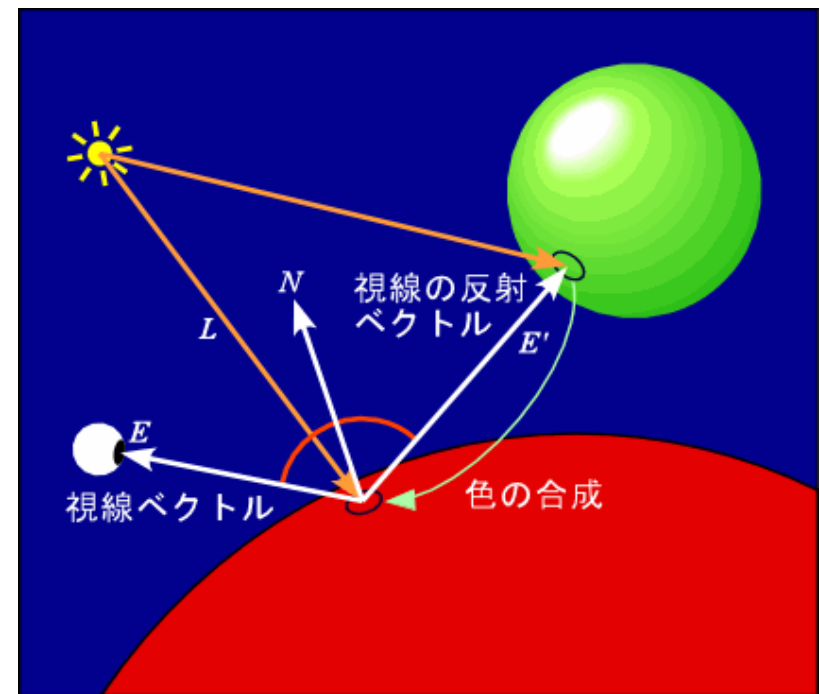
ハイライトの広がりの制御



小 ← n → 大

視線の反射ベクトルを追跡

- 可視点を視点として，反射方向に何が見えるか調べる
- 反射方向に見えた色を可視点の色と合成する

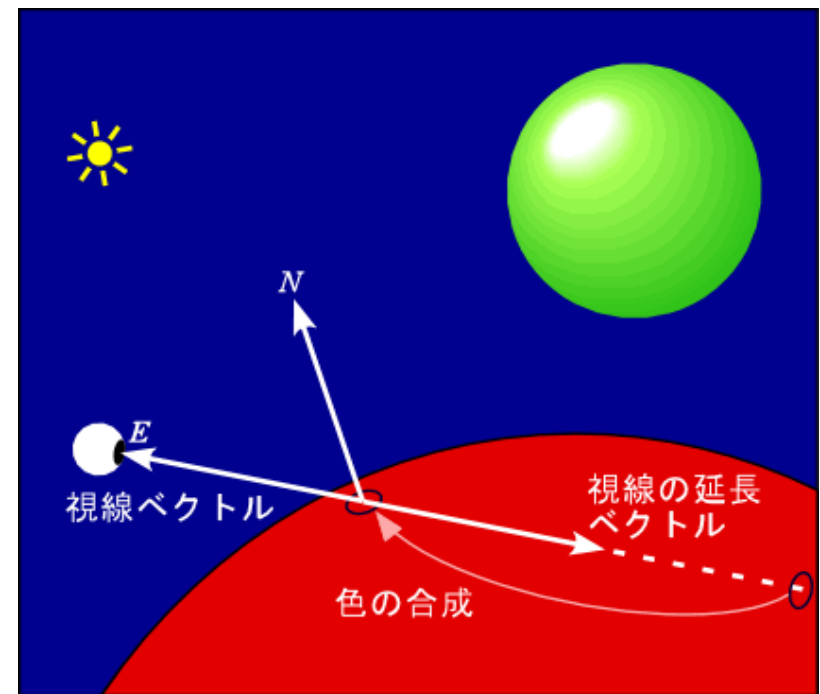


映り込み



視線の延長ベクトルを追跡

- 可視点を視点として，延長方向に何が見えるか調べる
- 延長方向に見えた色を可視点の色と合成する



透明



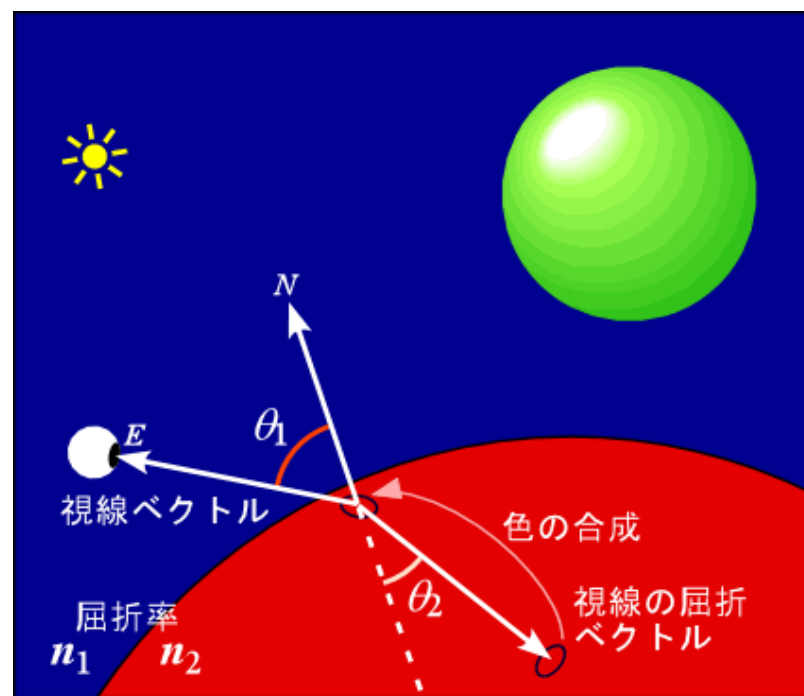
屈折率を考慮する

- 二つの媒質の屈折率から屈折方向を求める

- スネルの法則

$$n_1(\lambda)\sin\theta_1 = n_2(\lambda)\sin\theta_2$$

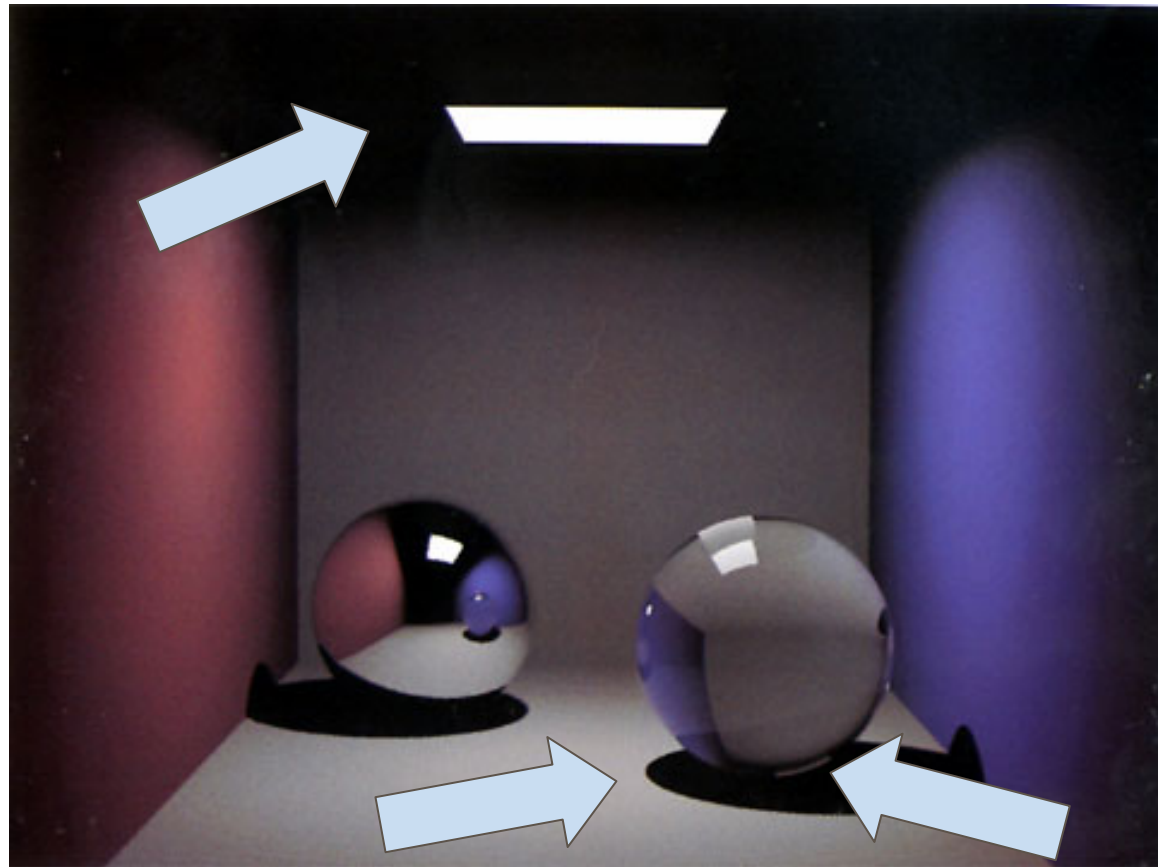
- 屈折方向に見えた色を可視点の色と合成する



屈折

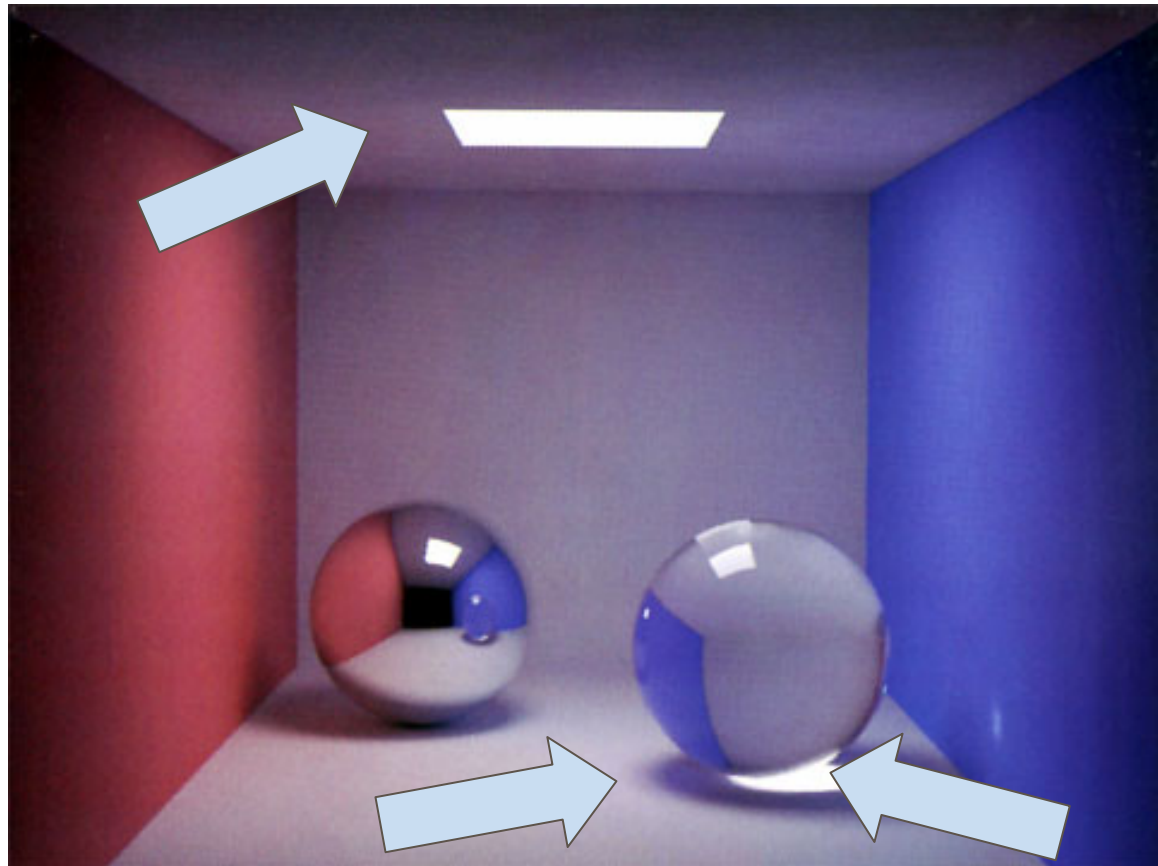


この程度までは表現できる



Henrik Wann Jensen 著、苗村 健 訳: “フォトンマッピング”

これと比べれば・・・



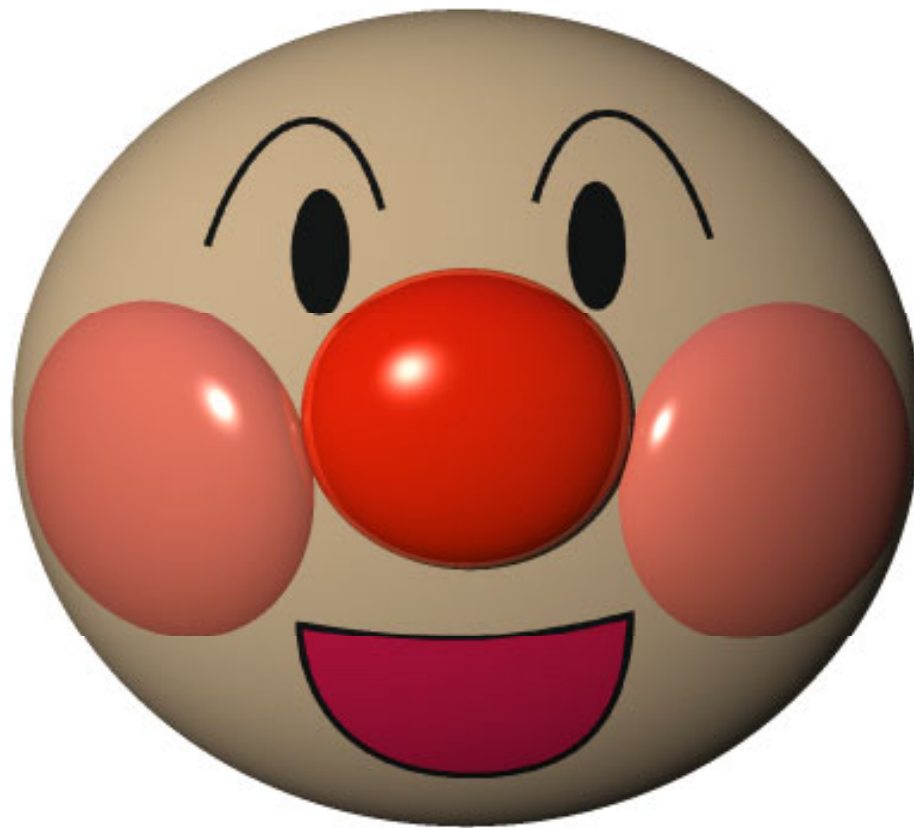
Henrik Wann Jensen 著、苗村 健 訳: “フォトンマッピング”



何が足りない？

- 天井の明るさ（相互反射）
 - 天井には光源の光が直接届かない
 - 天井を照らしているのは壁などによる間接光
- 物体の影（エリアライト）
 - 天井の光源は面積をもっている
 - 半影（ぼやけた影）が生じる
- ガラスの球による集光模様（コースティクス）
 - ガラスは光を透過させる
 - ガラスは視線だけでなく光線も屈折させる

アンパンマンを描いてみた



なんかちがう・・・

これならどうだ



まだちがう・・・というのはいいとして

マンガのアンパンマン

- 元気なアンパンマンのほっぺは「つややか」
 - ハイライトはつややかさを表現する有効な手段
- ハイライト形状は円形とは限らない



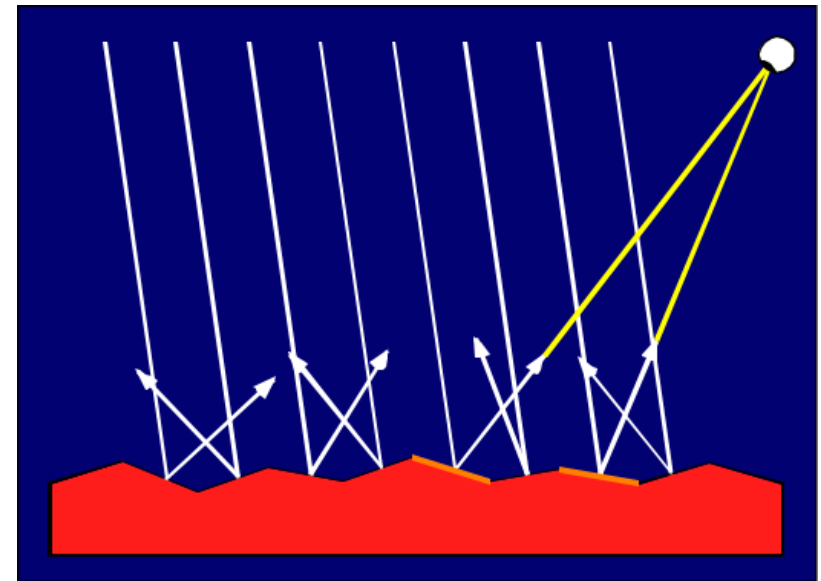


ハイライトの形

- ハイライトの形を決めるもの
 - 光源の形
 - 物体の表面形状によるゆがみ
 - 物体表面の微小形状によるボケ
 - その他
 - 光の伝播路上のちり，ゆらぎ等

粗い物体表面

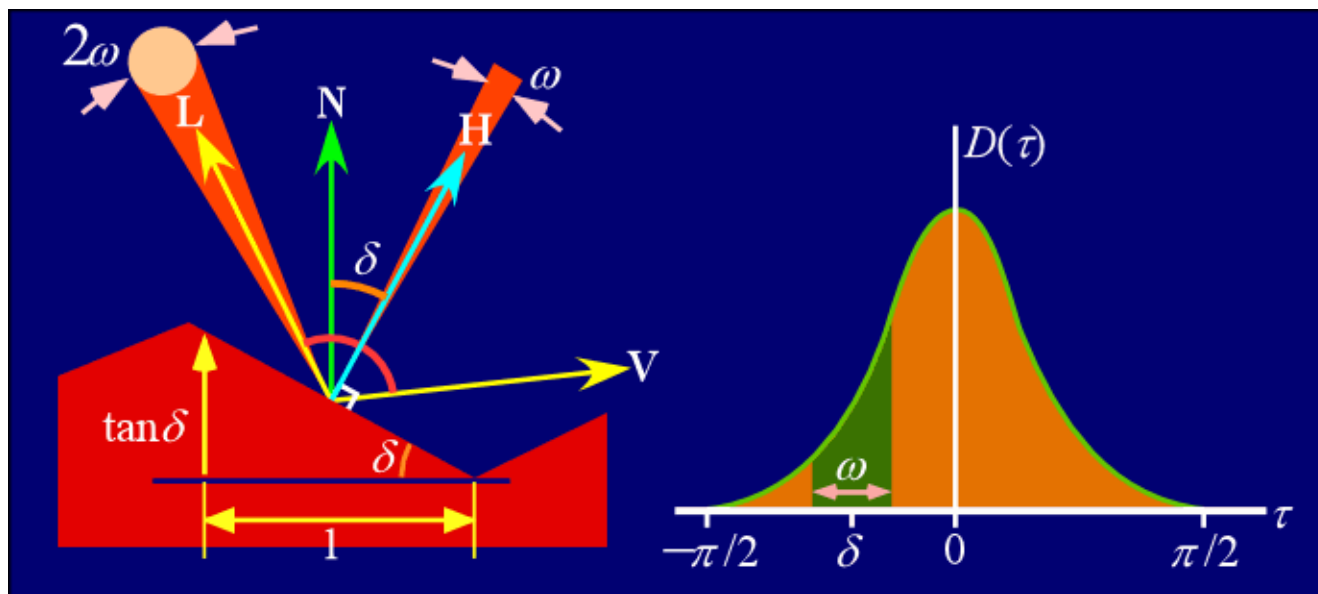
- 微小な凹凸がある
 - 一定割合の微小面がハイライトを生じる
 - 微小面の向きがばらつくから、ハイライトが広がる



凹凸の向ききの分布関数 $D(\tau)$

- 光源の視直径 2ω

- 鏡面反射率 $r_s = \int_{\delta-\omega/2}^{\delta+\omega/2} D(\tau) d\tau$

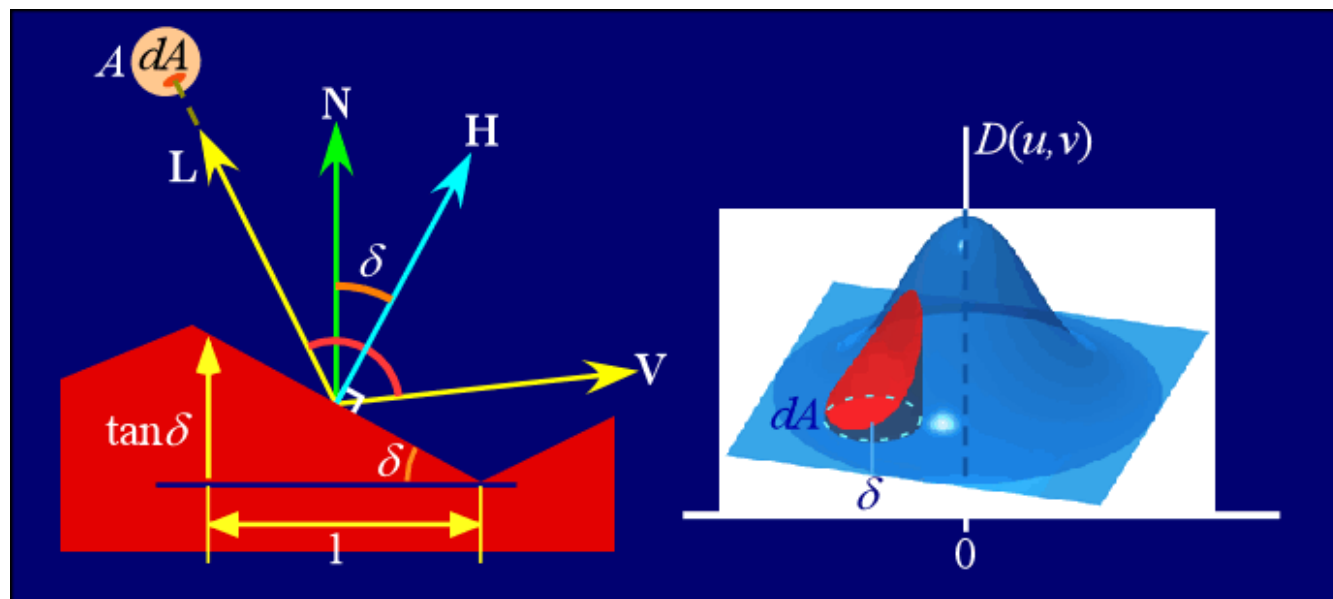


D は2次元分布

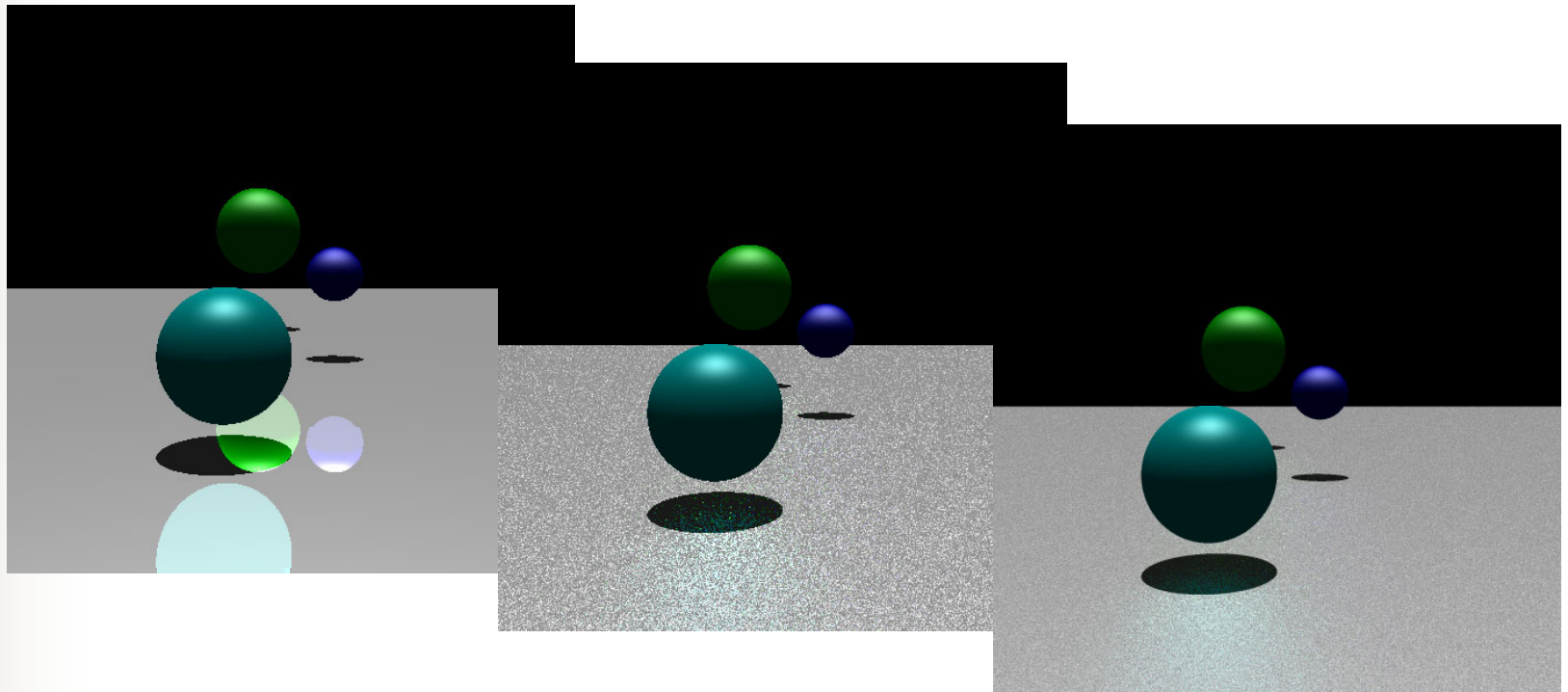
- 光源の面積 A

- 鏡面反射率

$$r_s = \int_A D(u, v) dA$$



粗い表面への映り込み

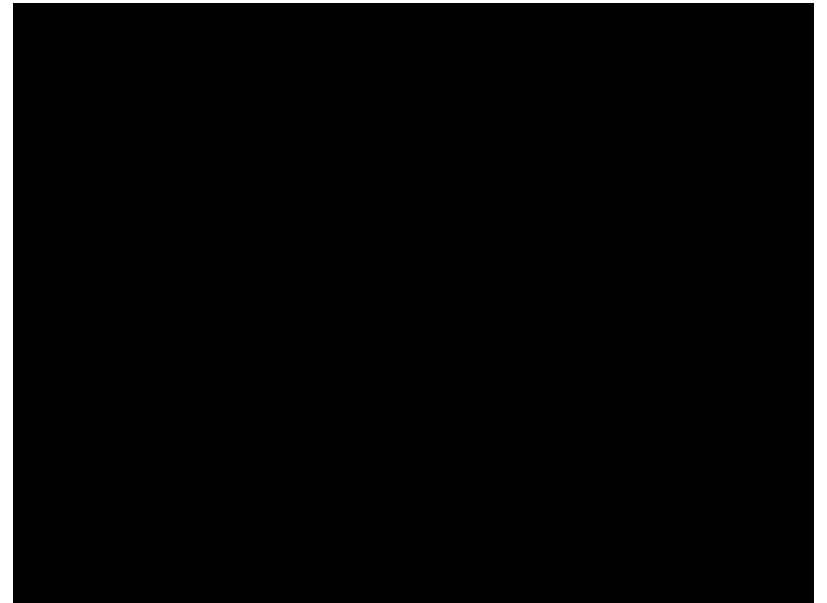


篠原 靖知:“粗い物体の表面形状計測に基づく陰影付けモデルの決定および樹脂成型物の質感設計への応用,” 和歌山大学システム工学部卒業論文 (2001)



粗い表面へのハイライト

■ Phong のモデル

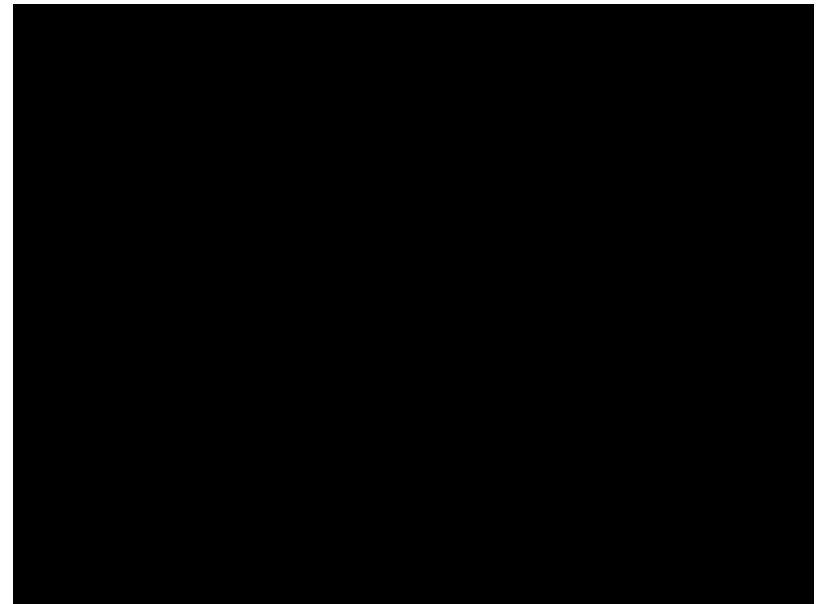


篠原 靖知:“粗い物体の表面形状計測に基づく陰影付けモデルの決定および樹脂成型物の質感設計への応用,” 和歌山大学システム工学部卒業論文 (2001)



粗い表面へのハイライト

- 分布関数を使ったモデル



篠原 靖知:“粗い物体の表面形状計測に基づく陰影付けモデルの決定および樹脂成型物の質感設計への応用,” 和歌山大学システム工学部卒業論文 (2001)

内部散乱まで考える

どっちが「牛乳(スキムミルク)」らしい？



Henrik Wann Jensen ほか: “*A Practical Model for Subsurface Light Transport*,” SIGGRAPH 2001 Conference Proceedings

レポートの質問について



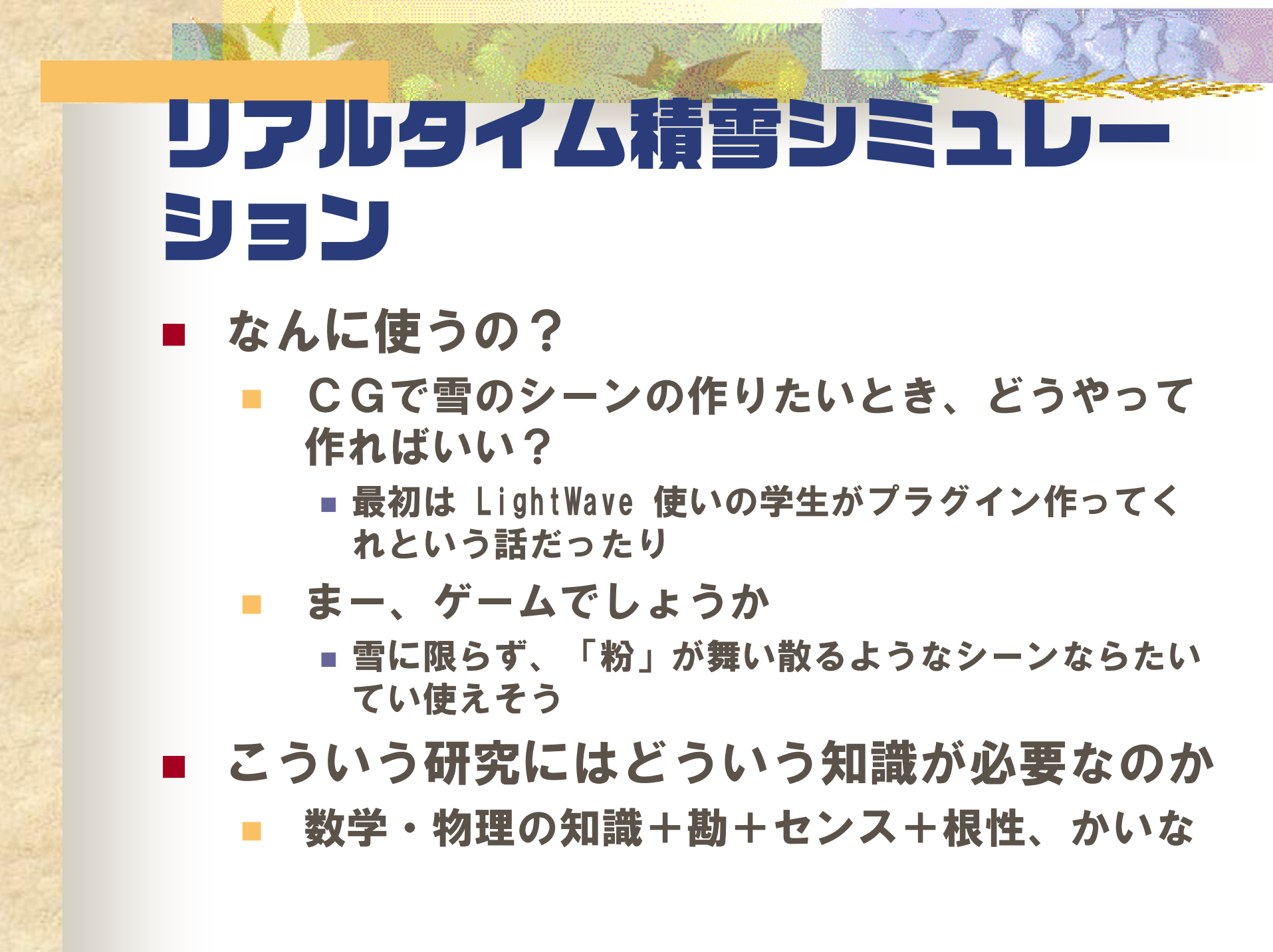
簡単に・・・

VRの「二本ノック」

■ なぜ立体に見えるのか

- スクリーンには右眼用の映像と左眼用の映像が1/120秒ごとに交互に表示されている
- ゴーグル（液晶シャッタメガネ）は、右眼用の映像が表示されているときには左眼のシャッタを閉じ、左眼用の映像が表示されているときには右眼のシャッタを閉じる
- こうして右眼と左眼に異なる映像を見せる
 - 3時間ほどで作ったデモで、研究的意味はありません
 - 立体視を実現するプログラムは3年の実験で作ります

時分割
ステレオ



リアルタイム積雪シミュレーション

- なんに使うの？
 - CGで雪のシーンの作りたいとき、どうやって作ればいい？
 - 最初は LightWave 使いの学生がプラグイン作ってくれという話だったり
 - まー、ゲームでしょうか
 - 雪に限らず、「粉」が舞い散るようなシーンならたいてい使えそう
- こういう研究にはどういう知識が必要なのか
 - 数学・物理の知識＋勘＋センス＋根性、かいな