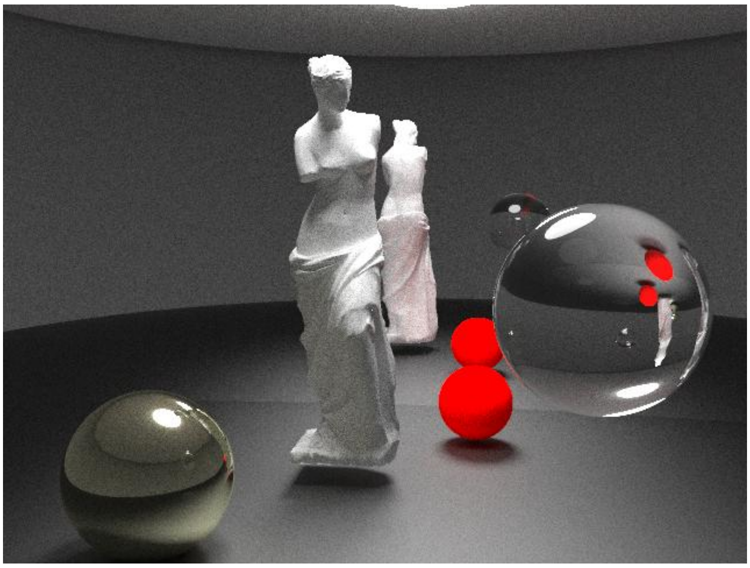
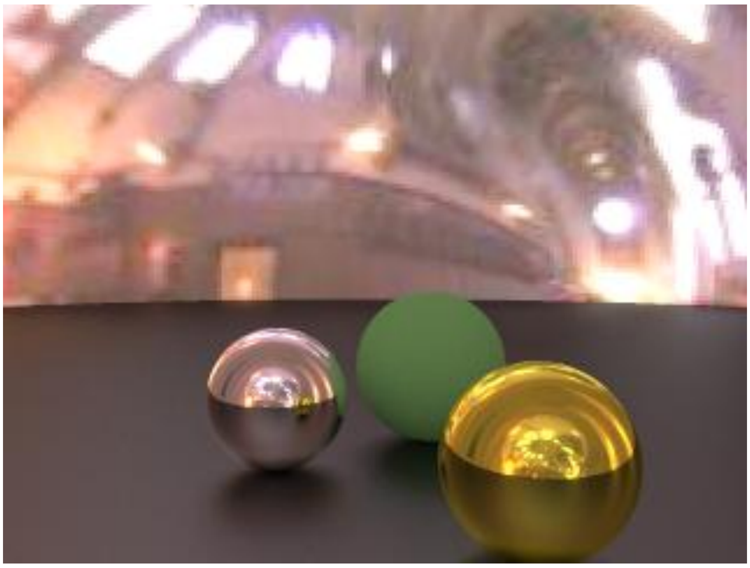
CG (Global Illumination) based on physical models





Global illumination is one of the technologies that produces images that are indistinguishable from the real world (live action).

- Considers mutual reflection of light.

Fulfills the law of conservation of energy.

In other words, the image must be physically correct. What is a physically correct image? In the first place, when a person sees an object, the light from the light source irradiates the object, and the reflected light from the surface of the object enters the visual system and becomes a color stimulus to obtain an image of the real world (in the brain).

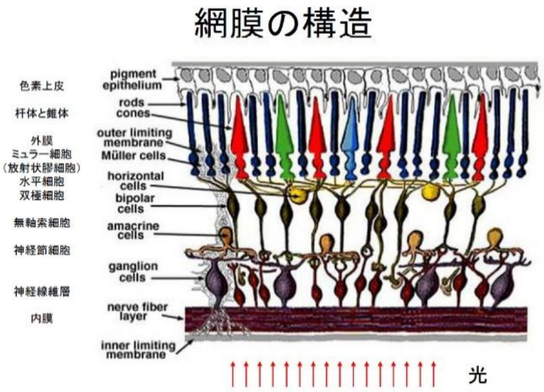


Fig. 2. Simple diagram of the organization of the retina.

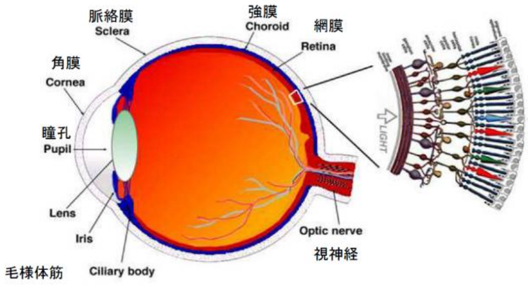


Fig. 1.1. A drawing of a section through the human eye with a schematic enlargement of the retina.

参考図の引用）http://webvision.med.utah.edu/

Sensing Brightness: Rods Cells

Senses red, green, blue: cone cells = >This is where RGB comes from.

In other words, it is necessary to accurately reproduce the physical behavior of light and obtain it by simulating RGB.

Rather than specifying the color, RGB (color) is the result of the simulation. The simulation can be performed by solving the following rendering equations.

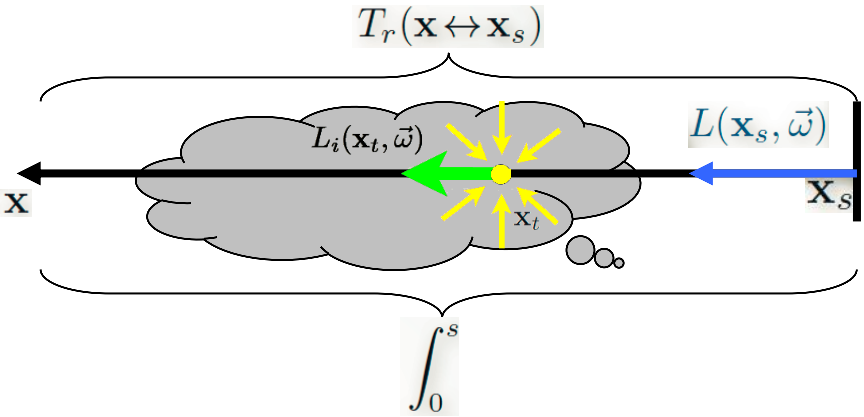
レンダリング方程式(The Rendering Equation)

$$L(x,\omega) = L_e(x,\omega) + \int_{\Omega} L_i(x,\omega')f_r(x,\omega,\omega')\cos(\theta)d\omega'$$

$$\cos(\theta) = n \cdot \omega'$$

Solve rendering equations to accurately reproduce physically correct light behavior
It is calculated by the path tracing method and Monte Carlo integrals.

When dealing with scattering by medium



$$L(x,\omega) = \int_0^s T_r(x \leftrightarrow x_t)\sigma_s(x_t)\left(\int_\Omega p(x_t,\omega_t,\omega')L(x_t,\omega_t)d\omega'\right)dt + T_r(x \leftrightarrow x_s)L(x_s,\omega)$$

$$T_r(x \leftrightarrow x_t) = \exp\left(-\int_0^{\|x_t - x\|} \sigma_t(x + s\omega_{x \to x_t})ds\right)$$

$\sigma_a(x)$: absorption *coefficien t*[1 / *m*]
$\sigma_s(x)$: *scattering coefficien t*[1 / *m*]
$\sigma_t(x) = \sigma_a(x) + \sigma_s(x)$
$p(x_t, \omega_t, \omega')$: *phase function*[1 / *sr*]

restriction

If two or more surfaces overlap, the result may not be correct.
When dealing with involved media and subsurface scattering, the front and back of the model are important, so a complete solid model is required. Otherwise, we may not be able to make the correct calculations.
Involved media and subsurface scattering do not correctly calculate nested states.
For example, there is an object inside an object that is scattered below the surface. The same applies to cases where some of them wrap.
Since it is recursive processing, the stack may overflow when computating in parallel with threads, so it may not be possible to calculate unless the number of threads is set to 1 to prevent parallelization.

■ Description of description

keyword
The settings of the corresponding parameters are described from the keywords.
A blank line ends the description for the keyword.
parameter
It means the description of the parameters. It can be a number or an identifier (string).
parameter
It means the description of the parameters. You can specify a number or a formula.
Formulas cannot contain spaces such as spaces, tabs, or line breaks. This is due to the use of spaces, tabs, and line breaks in the parameter separator.
関数には srqt,exp,log,pow,sin,cos,tan,asin,acos,atan,atan2,floor,ceil が使えます。

Parallel processing

Parallel processing is possible with OpenMP and MPI.

OpenMP uses parallelization by threads. The number of threads (number of parallels) can be set with the thread keyword. Since OpenMP performs parallel processing on a single PC (resource), even if you simply increase the number of parallels, you need to worry about stackover and limitations due to the number of cores.

It may be better to make it about 70% ~ 8% of the number of cores of the PC. If the number of cores is 7, the number of threads is about 5.

If the maximum number of threads is specified in the environment variable OMP_NUM_THREADS, this value takes precedence.

Parallelization by MPI can be calculated by distributing it to multiple PCs with an MPI environment (MPICH2).

Since the scene data is supposed to hold the same data on all nodes, it is necessary to place all the data on the PC to be calculated. The larger the number of PCs, the shorter the time it takes.

■ Output image settings

IMAGE

Image Pixel Width Image Pixel Height

If the environment variable IMAGE_X is set, the horizontal size takes precedence over the environment variable setting.

If the environment variable IMAGE_Y is set, the vertical size takes precedence over the environment variable setting.

OUTPUT

Output image file name

- The output image format follows the extensions .ppm .bmp .hdr.

BMP must be output

The number of Mr./Ms. samples x the number of super Mr./Ms. is added to the output file name.

If the environment variable SUFFIX_SYMBOL is set, its value (string) is appended to the end of the file name.

IMAGEDUMP

Time (seconds)

- Output the results of the calculation at specified intervals.

SCREEN

x y z

- Screen center coordinates when rendering

sampling.

SAMPLING

n

- Number of Mr./Ms. rings.

If the environment variable SAMPLING is set, the environment variable setting takes precedence.

SUPERSAMPLING

n

- Number of super Mr./Ms. samples

If the environment variable SUPERSAMPLING is set, the setting of the environment variable takes precedence.

Parallel processing

THREAD

n

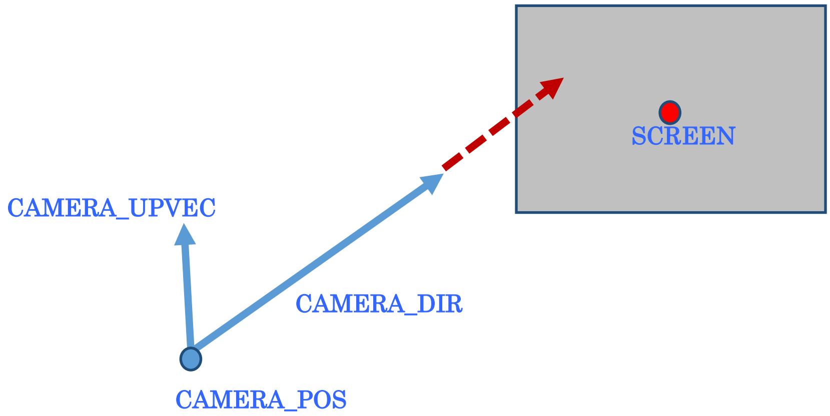
- Number of parallel executions

■ Camera settings

CAMERA_POS

x y z

- Camera Position



CAMERA_UPVEC

x y z

- Camera UP vector

CAMERA_DIR

x y z

- Camera Direction Vector

CAMERA_MATRIX

Matrix Definition

- Change the attitude of the camera according to the specified matrix

If you do not specify a CAMERA_DIR

TARGET_POS

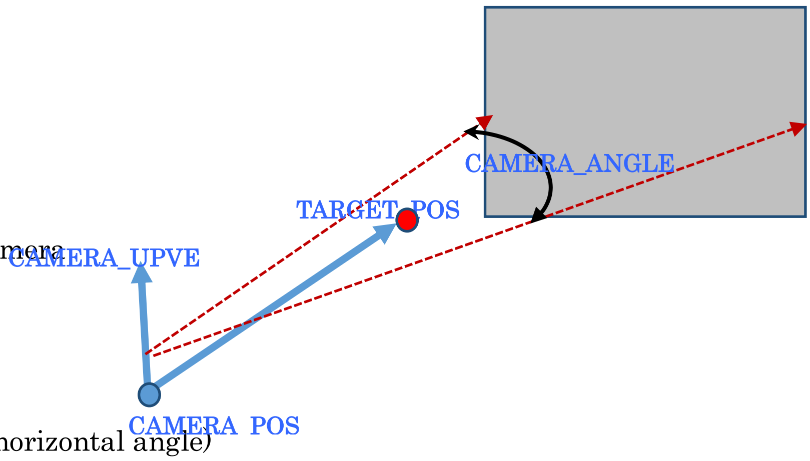
x y z

- Position as seen from the camera

CAMERA_ANGLE

I

- The view from the camera (horizontal angle)



■ Matrix Definition

rotation *θx θy θz*

- Angle of rotation of each axis (degrees)

translate *x y z*

- Travel on each axis

scale *x y z*

- Scaling ratio in each direction

■ Variable definition

VAR *varName*

x

- Set the variable varName to the value x. This variable can be used in the following "Expressions"

■Next Event Estimation

nextEventEstimation

value

- If the value is 0, Next Event Estimation is not performed.

If the environment variable NEXTEVENTESTIMATION is set, the setting of the environment variable takes precedence.

■ Filters

tentfilter

value

- If the value is 0, the tent filter is not performed (default).

■ Gamma Correction

gamma_offset

value

-Final output images undergo this gamma correction. A value of 0 disables gamma correction (default).

■ Noise filter

luminacnce_cuttoff

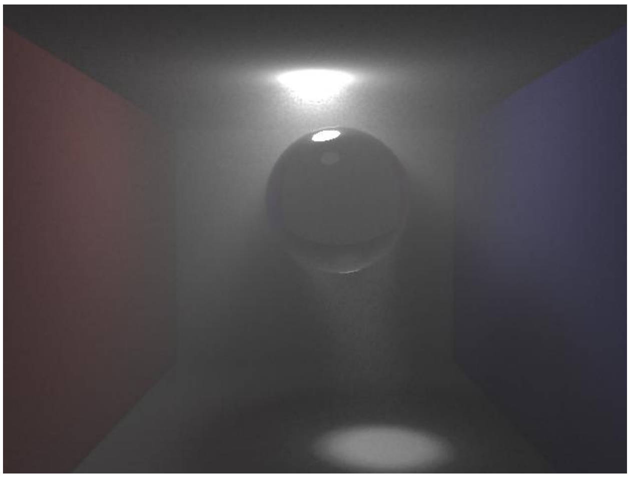
value

- If the value is 0, do nothing (default). The generation of high-brightness pixels is suppressed by limiting the luminance so that it does not exceed the specified value during calculation.

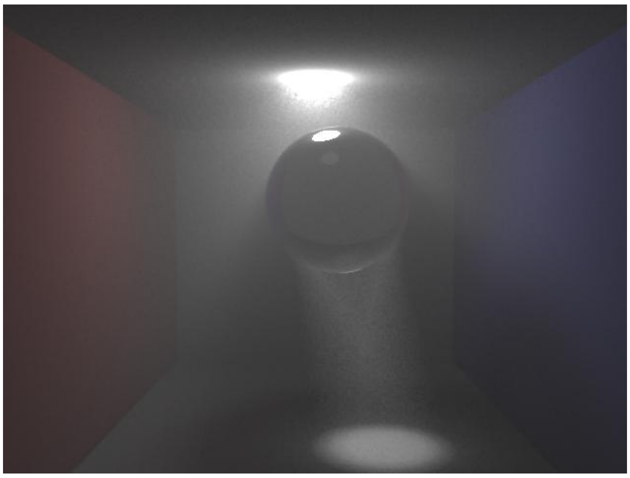
If the environment variable LUMINANCE_CUTTOFF is set, the value takes precedence over the setting of the environment variable.

■ Involved medium

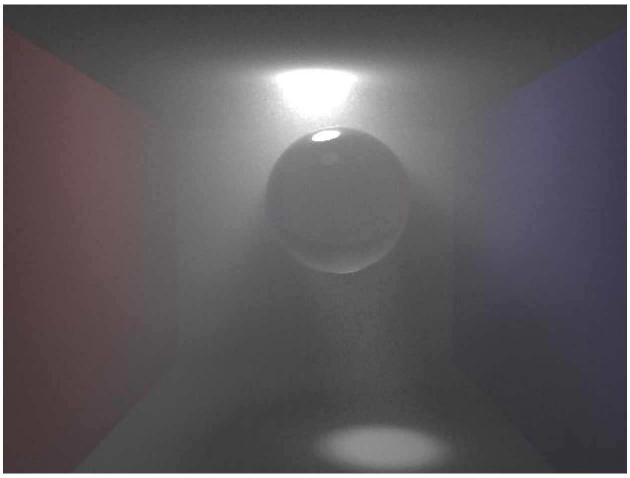
Scattered ahead (Scattering in the direction of the light beam)



Backscattering (scattering in the opposite direction to the direction of incident light)



Isotropic scattering



participatingMedia

value

- If the value is 0, the involved medium is ignored.

SCATTERING

σ s mean (dummy)

σ s[R] σs[G] σs[B]

- Scattering coefficient [1/mm]

ABSORBING

σ s mean (dummy)

σ s[R] σs[G] σs[B]

- Absorption coefficient [1/mm]

ANNOYED

Phase function

PAHASE *phase functionR phase functionG phase functionB*

値が－1 ≦ *phase function* ≦ ＋1 の場合 Henyey-Greenstein phase function. and given by the value of the parameter g. *The number of rank correlations is specified by the parameter g.* Forward scattering when g is close to 1. Back scattering when close to -1. If it is 0, it is isotropic and uniformly scattered.

$$p(\theta) = \frac{1}{4\pi} \frac{1 - g^2}{[1 + g^2 - 2g \cos(\theta)]^{3/2}}$$

However, when g = 0, the uniform scattering phase is obtained.

$$p(\theta) = \frac{1}{4\pi}$$

■ Object Definition

OBJECT

Shape Definition

Material Definition

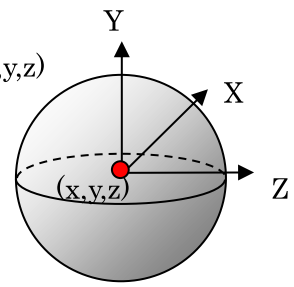
normal *f*

- Whether to invert the normal or not

■ Shape definition

sphere *x y z r*

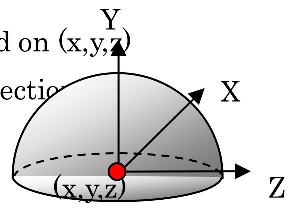
- sphere of radius r centered on (x,y,z)



hemisphere *x y z r*

- A hemisphere of radius r centered on (x,y,z)

  Define the upper half of the Y directio



uvplane x y z

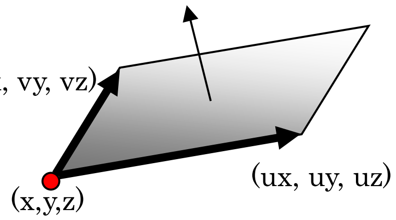
U ux uy uz

V vx vy

- A finite plane with an origin of (x,y,z)(vx, vy, vz)

Locale X(ux, uy, uz)

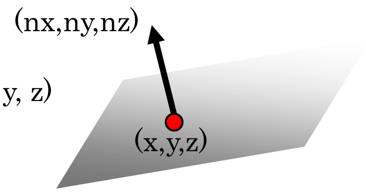
Loacle Y(vx, vy, vz)



plane x y z

normal nx ny nz

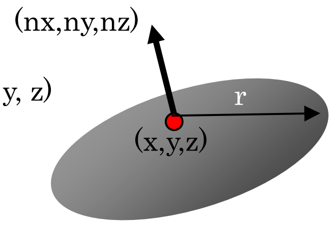
-Normal vector Normal vector passing through (x, y, z) passing through (x, y, z)



disk x y z r

normal nx ny nz

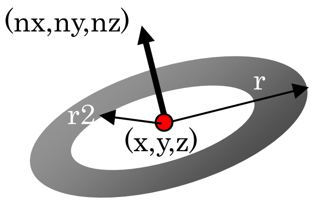
-Normal vector Normal vector passing through (x, y, z) passing through (x, y, z)



disk2 x y z r r2

normal nx ny nz

-Normal vector Normal vector passing through (x, y, z) passing through (x, y, z)



infinity_light nx ny nz

- Light source radiation direction vector (nx,ny,nz)

See also "Light Source"

objfile *file name*

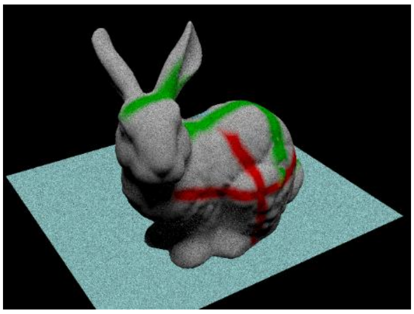
- Mesh data in OBJ format

Vertex color support

As an extended OBJ format, vertex definitions and colors can be set at the same time.

(Example)

```
v -0.037830 0.127940 0.004475 0.752941 0.752941 0.752941
v -0.022605 0.126675 0.007156 0.000000 0.666667 0.000000
```



Material definitions for OBJ files

If there is no material definition, the specified attribute will be set.

Kd *r g b*

-color r g b.

Not *even*

- interpret as refractive index.

Ns *s*

-For Phong, it is interpreted as the specular exponent.

map_Kd *Texture file name*

- Texture files are 8, 16, 24, 32-bit bitmaps, jpge, png, tga, hdr files only

map_Bump *Bump mapping texture filename*

- Texture files are 8, 16, 24, 32-bit bitmaps, jpge, png, tga, hdr files only

It will be the texture of the material specified by newmtl.

If specified in texture, this texture takes precedence.

color, specular is multiplied by the color.

Material Definition Extension for OBJ Files

* These specifications may not function properly at this time.

#REFLECTION *r*

- Specification of reflectance distribution function (BRDF)

r= 0 Perfectly diffuse surface. So-called Lambertian surface。

1, Ideal mirror surface.

2, Ideal vitreous substance.

3, Ward's Glossy

4, Manners

#EMISSION *r g b*

-luminescence

When specifying #EMISSION, please also specify an appropriate value for #COLOR. If you want to apply it to the entire OBJ file, specify it in the attribute.

#COLOR *r g b*

- Reflected light

#WARD *ax ay*

-Ward Reflection Model

#SPECULAR *r g b*

- Specular reflection (see in Ward's Reflection Model)

#ROUGHNESS *r*

- Surface roughness (0~1)

■Attribute definition

Attribute definitions may be set in duplicate, but the last specification takes effect.

■Material definition

color *r g b*

- Reflected light

In reflection models other than full diffusion (Lambert), it is referred to as a diffuse component)

In fully mirrored and glass reflective models, it is referred to as a specular component)

emission *r g b*

-luminescence

specular *r g b*

- Reflected light

Specular components in reflective models other than fully specular and glass surfaces

roughness *r*

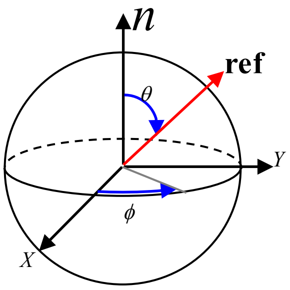
- Surface roughness (0~1)

■Bidirectional Reflectance Distribution Function（BRDF)

reflection diffuse

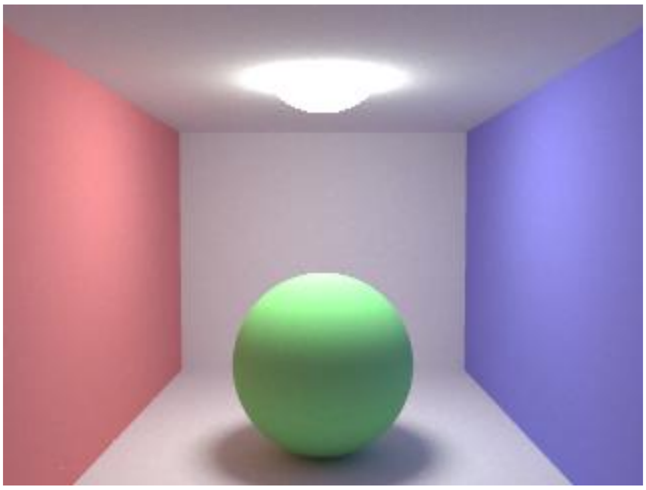
- Diffuse reflection (roughness is 1.0 by default)

$$BRDF = \frac{\mathbf{color}}{\pi}$$



$$\mathbf{ref} = \begin{pmatrix} \sin(\theta)\cos(\phi) \\ \sin(\theta)\sin(\phi) \\ \cos(\theta) \end{pmatrix} \qquad \begin{aligned} \phi &= 2\pi\,\xi_1 \\ \theta &= \cos^{-1}(\sqrt{1-\xi_2}) \end{aligned}$$

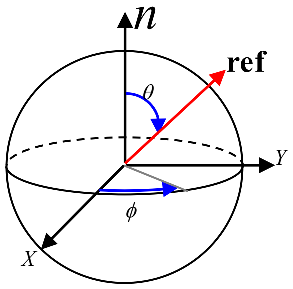
Given two uniform random variables ζ in the range $0 < \zeta < 1$,



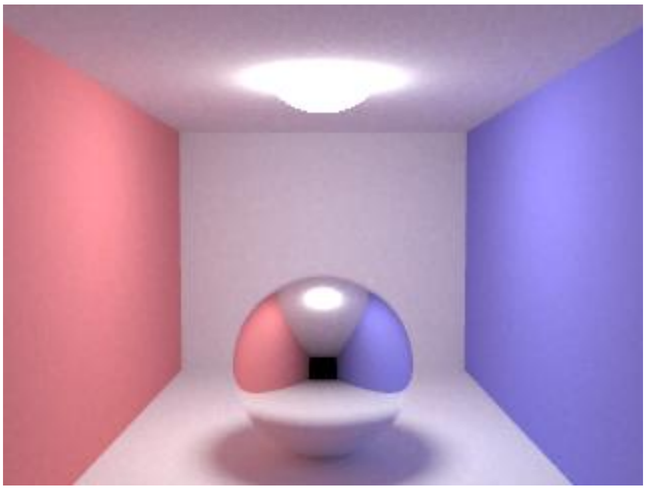
reflection specular

-Specular reflection (roughness is 0.0 by default)

$$BRDF = \mathbf{specular}\ \delta([\mathbf{ray} - 2(\mathbf{n}\cdot\mathbf{ray})\mathbf{n}] - \mathbf{ref})$$



$$\mathbf{ref} = \mathbf{ray} - 2(\mathbf{n}\cdot\mathbf{ray})\mathbf{n}$$



reflection reflection

- Reflection and refraction

Snell's law applies. The external refractive index n_1, the internal refractive index after incidence n_2

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

$$\cos^2(\theta_2) = 1 - \left(\frac{n_1}{n_2}\right)^2 (1 - (-\mathbf{ray} \cdot \mathbf{n})^2)$$

Refractive direction

$$\mathbf{T} = \mathbf{ray}\left(\frac{n_1}{n_2}\right) - \left(\left(\frac{n_1}{n_2}\right)(\mathbf{n} \cdot \mathbf{ray}) + \sqrt{1 - \left(\frac{n_1}{n_2}\right)^2 (1 - (-\mathbf{ray} \cdot \mathbf{n})^2)}\right)\mathbf{n}$$
$$= \mathbf{ray}\left(\frac{n_1}{n_2}\right) - \left(\left(\frac{n_1}{n_2}\right)(\mathbf{n} \cdot \mathbf{ray}) + \sqrt{\cos^2\theta_2}\right)\mathbf{n}$$

The proportion of light carried by reflected light follows the Fresnel → Schlick approximation.

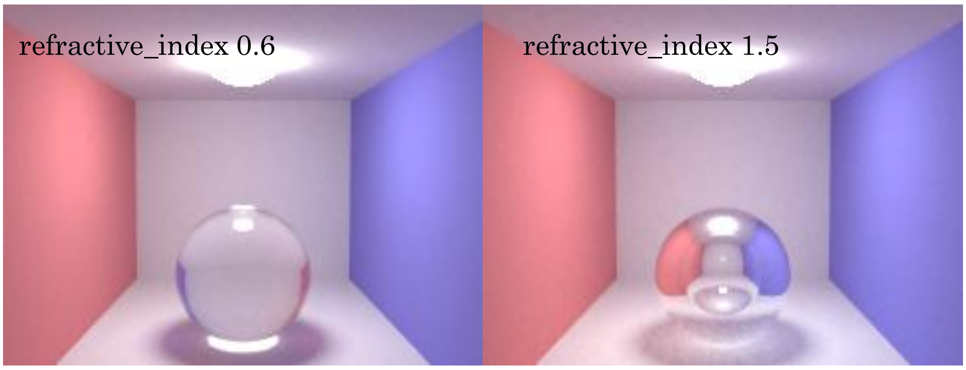
$$F(\theta_1) = F_0 + (1 - F_0)(1 - \cos\theta_1)^5 \qquad F_0 = \frac{(n_1 - n_2)^2}{(n_1 + n_2)^2}$$

$\cos^2(\theta_2) > 0$

$$BRDF = \mathbf{specular}\left(\overbrace{F(\theta_1)\boldsymbol{\delta}([\mathbf{ray} - 2(\mathbf{n} \cdot \mathbf{ray})\mathbf{n}] - \mathbf{ref})}^{\text{reflection}} + \underbrace{(1 - F(\theta_1))\left(\frac{n_1}{n_2}\right)^2}_{\text{refraction}} \boldsymbol{\delta}(\mathbf{T} - \mathbf{ref})\right)$$

$\cos^2(\theta_2) \le 0$

$$BRDF = \mathbf{specular}\ \boldsymbol{\delta}([\mathbf{ray} - 2(\mathbf{n} \cdot \mathbf{ray})\mathbf{n}] - \mathbf{ref})$$



reflection ward_brdf

ward_brdf <u>ρ d ρ s *ax ay*</u>

-Ward Reflection Model

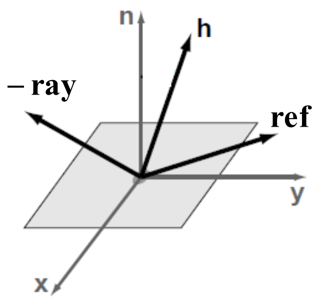
ρ d = diffuse reflectance

ρ s=specular reflectivity ratio

Anisotropic reflection (α x and α y each correspond to roughness with respect to directionality).

$$BRDF = \frac{\mathbf{specular}}{4\pi\alpha_x\alpha_y\sqrt{(-\mathbf{ray}\cdot\mathbf{n})(-\mathbf{ref}\cdot\mathbf{n})}}\exp\left(-\frac{\left(\frac{\mathbf{h}\cdot\mathbf{x}}{\alpha_x{}^2}+\frac{\mathbf{n}\cdot\mathbf{y})}{\alpha_y{}^2}\right)^2}{(\mathbf{h}\cdot\mathbf{n})^2}\right)$$

$$\mathbf{ref} = 2((-\mathbf{ray})\cdot\mathbf{h})\mathbf{h}-(-\mathbf{ray})$$

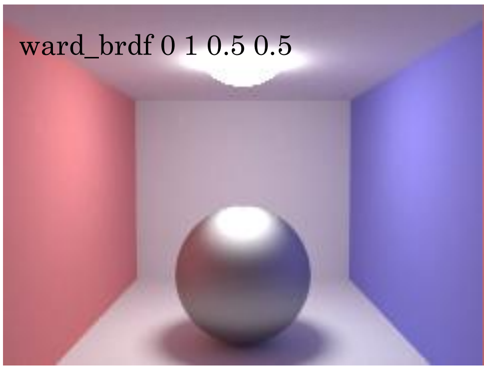


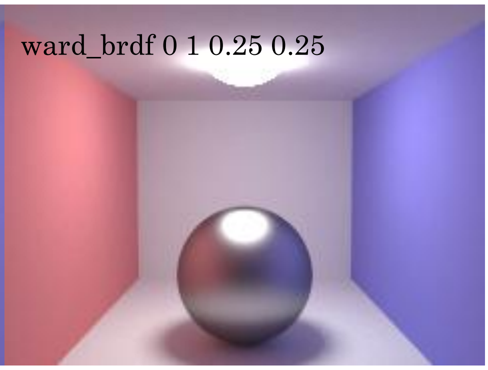
$$\mathbf{h} = \left[\sin\theta_h\cos\phi_h,\ \sin\theta_h\sin\phi_h,\ \cos\theta_h\right]$$

$$\theta_h = \arctan\sqrt{\frac{-\log u}{\cos^2\phi_h/\alpha_x^2+\sin^2\phi_h/\alpha_y^2}}$$

$$\phi_h = \arctan\left(\frac{\alpha_y}{\alpha_x}\tan(2\pi v)\right)$$

Given two uniform random variables u and v in the range 0 < u,v < 1,





reflection phong_brdf

phong_brdf ρ_d ρ_s *specular_exponent*

-Phong Reflection Model

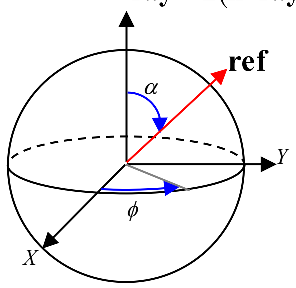
ρ d = diffuse reflectance

ρ s=specular reflectivity ratio

$$BRDF = \frac{n+2}{2\pi}\,\mathbf{specular}\cos^n(\alpha)$$

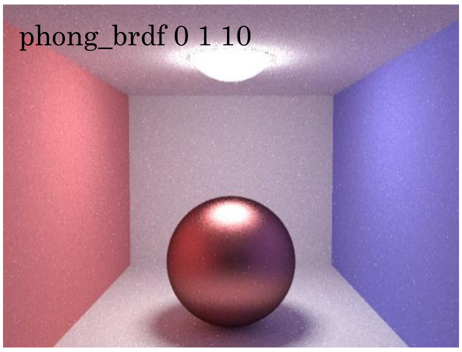
$$\cos(\alpha) = [\mathbf{ray} - 2(\mathbf{n}\cdot\mathbf{ray})\mathbf{n}]\cdot\mathbf{ref}$$

$$\mathbf{ray} - 2(\mathbf{n}\cdot\mathbf{ray})\mathbf{n}$$



$$\mathbf{ref} = \begin{pmatrix} \sin(\theta)\cos(\phi) \\ \sin(\theta)\sin(\phi) \\ \cos(\theta) \end{pmatrix} \qquad \begin{aligned} \phi &= 2\pi\,\xi_1 \\ \theta &= \cos^{-1}(pow(\xi_2, 1/(specular\_exponent+1)) \end{aligned}$$

Given two uniform random variables ζ in the range $0 < \zeta < 1$,



■ Refractive index definition

refractive_index r

- Refractive index r

reflection reflection1

a b c d e f

$$\text{Refractive index} = \sqrt{a + b\lambda^2 + c\lambda^{-2} + d\lambda^{-4} + e\lambda^{-6} + f\lambda^{-8}}$$

reflection reflection2

a b c d e f

$$\text{Refractive index} = \sqrt{1 + \frac{a\lambda^2}{(\lambda^2 - b)} + \frac{c\lambda^2}{(\lambda^2 - d)} + \frac{e\lambda^2}{(\lambda^2 - f)}}$$

The refractive index of various materials is "http://refractiveindex.info/Detailed data is published in the following article.

reflection reflection99

$$\text{Refractive index} = 1.400 + \frac{50.0}{\lambda - 230.0}$$

The unit of λ is nm

Subsurface scattering

reflection Subsurface_Scattering

BSSRDF

f

When -f is 0, it is calculated by scattering simulation. When it is 1 (default), it is calculated using BSSRDF Dipole Model for Subsurface Scattering. This value can also be set with the environment variable "USE_BSSRDF".

SCATTERING

σ s mean (dummy)

σ s[R] σs[G] σs[B]

-Scattering coefficient [1/mm] *See Involved Medium

ABSORBING

σ s mean (dummy)

σ s[R] σs[G] σs[B]

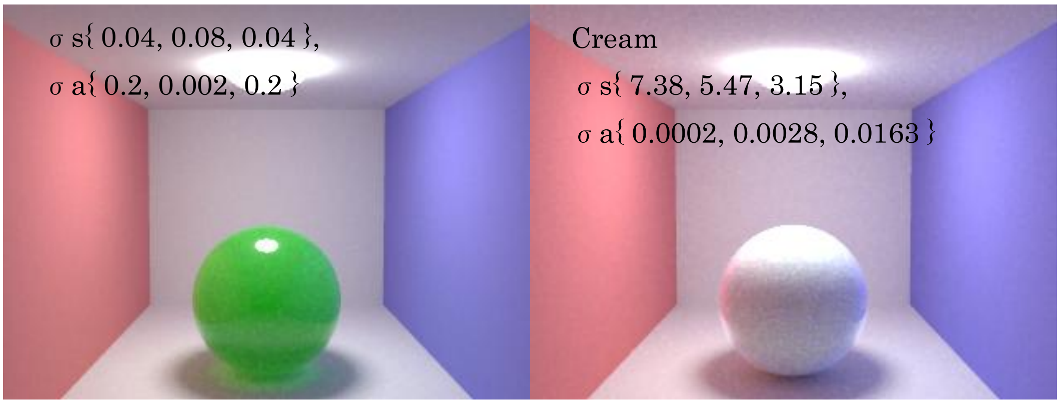
-Absorption coefficient [1/mm] *See Involved Medium

ANNOYED

Phase function

PAHASE *位相関数 R 位相関数 G 位相関数 B*

* Refer to the involved medium



level

Scattering order

0: No clutter.

-1: Full simulation when BSSRDF is zero, single scattering + multiple scattering (default) when BSSRDF is 1.

-2: Multiple scattering by BSSRDF only.

MATERIAL_SYMBOL

Material

A Practical Model for Subsurface scattering" (Jensen et al.)

Material	σ'_s [mm^{-1}]			σ_a [mm^{-1}]			Diffuse Reflectance			η
	R	G	B	R	G	B	R	G	B	
Apple	2.29	2.39	1.97	0.0030	0.0034	0.046	0.85	0.84	0.53	1.3
Chicken1	0.15	0.21	0.38	0.015	0.077	0.19	0.31	0.15	0.10	1.3
Chicken2	0.19	0.25	0.32	0.018	0.088	0.20	0.32	0.16	0.10	1.3
Cream	7.38	5.47	3.15	0.0002	0.0028	0.0163	0.98	0.90	0.73	1.3
Ketchup	0.18	0.07	0.03	0.061	0.97	1.45	0.16	0.01	0.00	1.3
Marble	2.19	2.62	3.00	0.0021	0.0041	0.0071	0.83	0.79	0.75	1.5
Potato	0.68	0.70	0.55	0.0024	0.0090	0.12	0.77	0.62	0.21	1.3
Skimmilk	0.70	1.22	1.90	0.0014	0.0025	0.0142	0.81	0.81	0.69	1.3
Skin1	0.74	0.88	1.01	0.032	0.17	0.48	0.44	0.22	0.13	1.3
Skin2	1.09	1.59	1.79	0.013	0.070	0.145	0.63	0.44	0.34	1.3
Spectralon	11.6	20.4	14.9	0.00	0.00	0.00	1.00	1.00	1.00	1.3
Wholemilk	2.55	3.21	3.77	0.0011	0.0024	0.014	0.91	0.88	0.76	1.3

$$\sigma'_s = \sigma_s(1-g)$$ *g= phase function*

SCATTERING and ABSORBING can be omitted.

MATERIAL_SCALE

scale

Correct the scattering and absorption coefficients by this coefficient. 1.0 if omitted

MATERIAL_R_MAX

R_max

Change of radius of impact. Multiply the specified value by the internally calculated radius. 1.0 if omitted

$$r\_\max = 1.25\sqrt{\log(ratio)} / -\sigma_t \times R\_\max$$
$$\text{ratio} = 1.0\text{e-}4$$

■ Light source

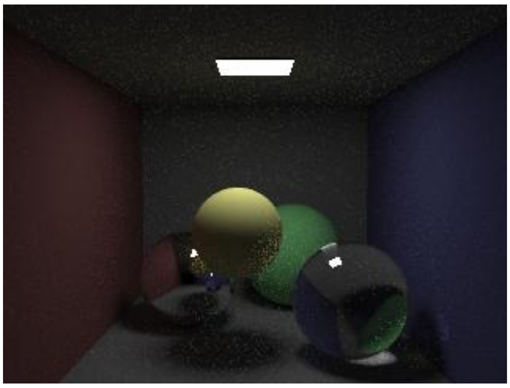
The light source is not particularly identified (emission *r g b*) and an object whose r, g, and b are not all zero emits light, so it is semantically a light source. However, objects with (color *r g b*) and r, g, and b are all zeros, and objects that only emit light are identified internally as light sources, especially when "Next Event Estimation" is enabled. Objects that can be recognized as light sources are limited to spheres, finite planes, and fully diffuse (Lambert) reflections.

Spherical light source

A sphere recognized as a light source is a light source that emits diffuse light in all directions.

Surface light source

The finite plane recognized as a light source is a light source that emits diffuse light based on the normal direction, and does not emit light on the back.



Directional light (directional light)

It is a parallel light source and has no attenuation due to distance. Objects are limited to a finite plane.

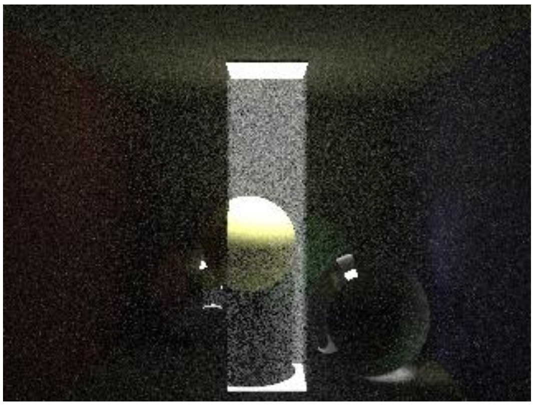
Also, if "**Next Event Estimation**" is not enabled, it will not be calculated correctly.

parallel_light *flag*

Valid only when -flag is non-zero.

parallel_light_dir *x y z*

- Specifies the direction vector of the parallel light source. If omitted, it will be in the normal direction.

Infinity light source

infinity_light *nx ny nz*

It is not a real light source, but functions as a light source that is infinitely far away. It is not a light source setting as an attribute of OBJECT. SEE THE DEFINITION OF OBJECT

Spot light source

There is attenuation due to distance in the spotlight. Objects are limited to spheres. However, the radius of the sphere does not affect. Also, if "**Next Event Estimation**" is not enabled, it will not be calculated correctly.

spot_light *flag*

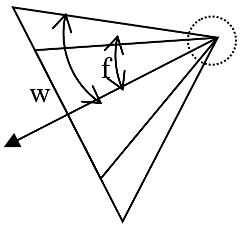
Valid only when -flag is non-zero.

spot_light_dir *x y z*

- Specifies the direction vector of the spot light source. If omitted, it becomes (0,-1,0).

spot_light_angle *w f*

w is the angle of the spot width, and f is the falloff angle. The defaults are 30 and 5, respectively.



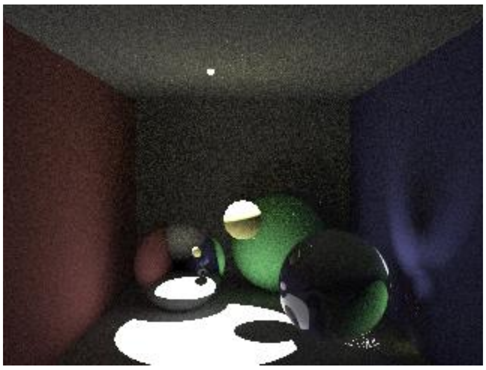
spot_light_falloff *p*

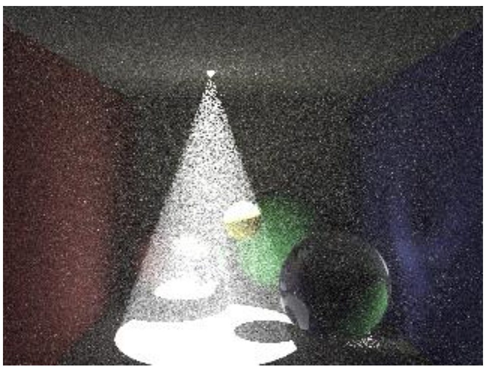
You can specify attenuation outward from falloff. The default value is 2.

spot_attenuation *a b c*

Attenuation with distance (1/t) can be specified.

$t = a + b\,\mathrm{dist} + c\,\mathrm{dist}^2$ It is calculated in: When t=0, there is no attenuation.





Ambient light

ENV_LIGHT *r g b*

It is not a real light source, but a light that illuminates an arbitrary position regardless of the light source.

You can also specify the following environment variables:

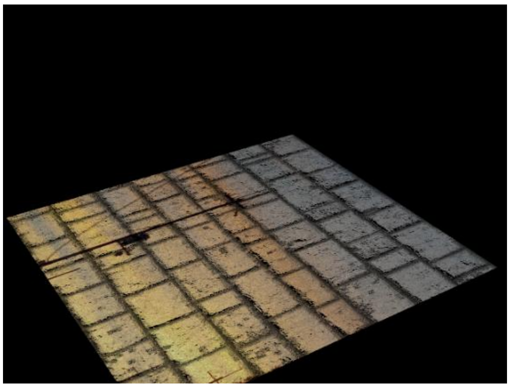
```
ENVLIGHT_R= r
ENVLIGHT_G= g
ENVLIGHT_B= b
```

■ **Texture**

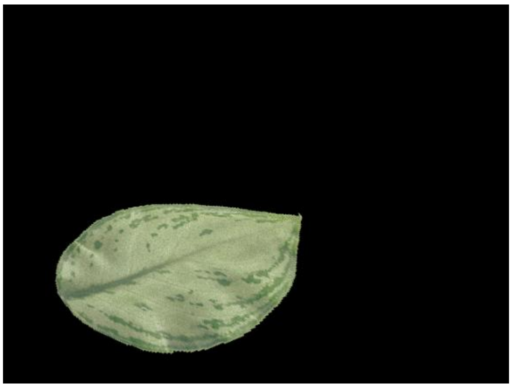
texture *テクスチャファイル名*

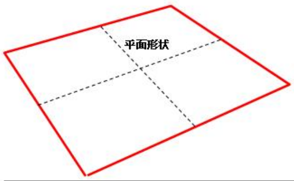
-Texture.　BMP file extension

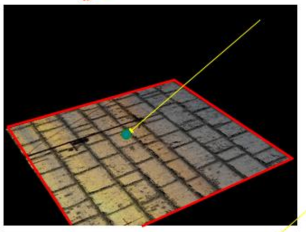
Mesh requires UV designation.

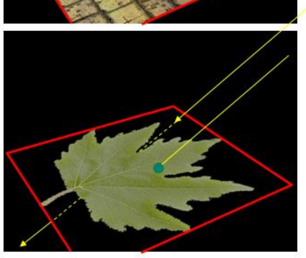


If there is a α channel (transparent, non-transparent), it is assumed that the object does not exist in the transparent position.







■IBL

ibl_texture テクスチャファイル名

- IBL definition. BMP or HDR files
  Mesh requires UV designation.

The definition of IBL must be defined at the end of the definition of the light source.

ibl_texture_coef c

- Coefficient to multiply by the value of IBL

The Color and emittion of the object to set IBL should be set to zero.

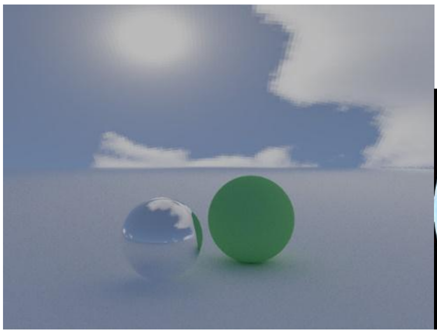
hemisphere_map f

- Whether to map the mapping from the center of the image to the disk shape (0 or 1)

Applications outside the hemisphere are not considered.

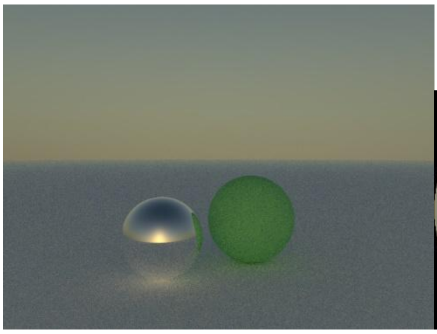
normal f

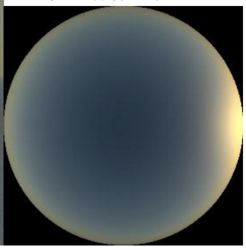
- Whether to invert the normal or not



Map the hemisphere by covering it with a dome.

Flip the normals to make the inside of the hemisphere face outward.

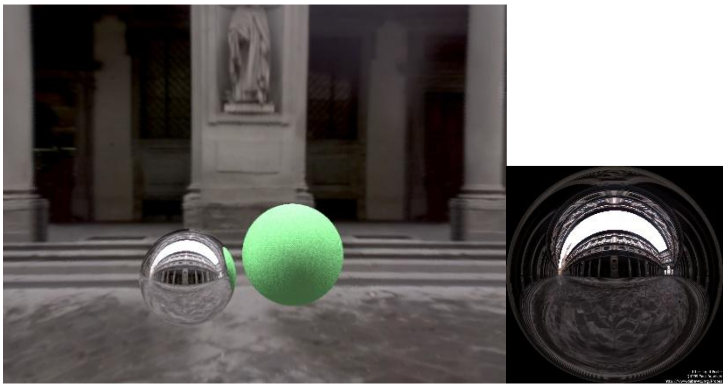




`angular_map` *f*

- Whether to map the mapping from the center of the image to the disk shape (0 or 1)

It is not considered for non-spherical applications.

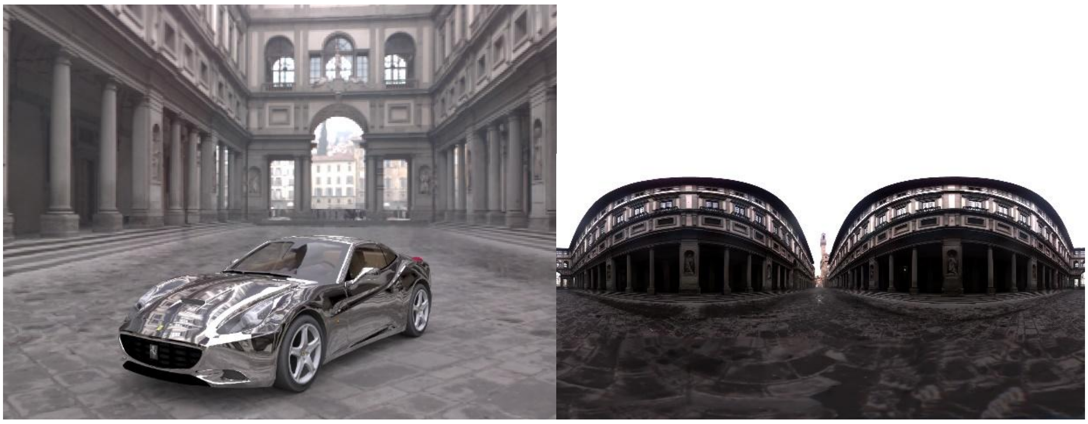


Images Copyright © 1998, 1999 Paul Debevec

`panoramic_map` *f*

- Whether the mapping should be mapped with a panoramic image by latitude and longitude (0 or 1)

It is not considered for non-spherical applications.



Images Copyright © 1998, 1999 Paul Debevec shadow *f*

- Set the shape attribute to the shadow-only attribute. Only planes and finite planes can be set.

Since IBL is a light source, it does not cast a shadow on it, so you can create a shadow with this setting.

On the right, when a flat surface is placed to create a shadow (the body color has been changed). If the shadow setting is set on this plane, it is calculated as follows.

■ Background

background_texture 背景のテクスチャ

The background will be applied as an infinity panorama IBL.

The background will be applied as a panoramic IBL. Set rotation

map_coef a b c d

i=i*a + b

$\phi = \phi * c + d$

■ Mesh

smooth *f*

- Whether to smoothly interpolate the normals of the mesh model (0 or 1)

■ Other

Markov Chain Monte Carlo (**MCMC**)の試行的導入

PSSMLT（**Primary** sample **space MLT**）
In the configuration, SAMPLE is referred to as the number of MLT executions.

`metropolisTransport`

n

-計算アルゴリズムを「**Primary** sample **space MLT**」で行う(n をゼロ以外に設定)。
If set to zero, it does not apply.
If the environment variable METROPOLIS is set, the setting of the environment variable takes precedence.

`mlt_sample`

n

-The **number of initial Mr./Ms. samples used in the "Primary** sample **space MLT**". Specify the number of Mr./Ms. pulls per pixel.
If the environment variable MLTSAMPLE is set, the setting of the environment variable takes precedence.

`mutation`

n

-**Set the number of mutations to be used in the "Primary** sample **space MLT**". Specify the number of mutations per pixel.
If the environment variable MUTATION is set, the setting of the environment variable takes precedence.

`mlt_resampling`

n

-**Generate an initial Mr./Ms. sample to be used in "Primary** sample **space MLT**" every time (set n to non-zero). If set to zero, it does not apply. If the environment variable MLTRESAMPLE is set, the setting of the environment variable takes precedence.

ERPT（Energy-redistribution path tracing）

energyRedistributionPathTracing

n

-計算アルゴリズムを「Energy-redistribution path tracing」で行う(n をゼロ以外に設定)。

If set to zero, it does not apply.

If the environment variable ERPT is set, the setting of the environment variable takes precedence.

mutation

n

- Set the number of mutations to be used in "Energy-redistribution path tracing". Specify the number of mutations per pixel.

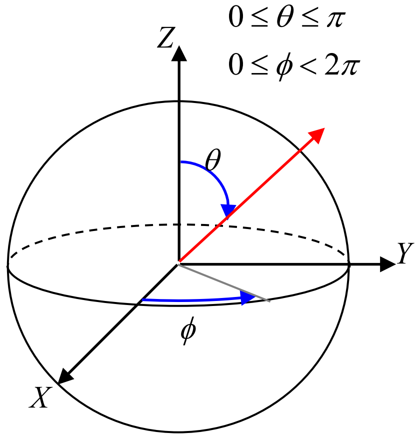
If the environment variable MUTATION is set, the setting of the environment variable takes precedence.

■ **Time limit**

If the environment variable TIMELIMIT is set, rendering is terminated with the value (time) of the environment variable.

In addition, the rendering result is also output every 1 minute when executed in 1 minute.
If there is no time limit, the rendering result will be output every minute where it could be processed in 1 minute.

References



$0 \le \theta \le \pi$

$0 \le \phi < 2\pi$

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = spher(\theta,\phi) = \begin{pmatrix} \sin(\theta)\cos(\phi) \\ \sin(\theta)\sin(\phi) \\ \cos(\theta) \end{pmatrix}$$

$$\begin{pmatrix} r \\ \theta \\ \phi \end{pmatrix} = \begin{pmatrix} \sqrt{x^2+y^2+z^2} \\ \cos^{-1}\left(\dfrac{z}{\sqrt{x^2+y^2+z^2}}\right) \\ \cos^{-1}\left(\dfrac{x}{\sqrt{x^2+y^2}}\right), \sin^{-1}\left(\dfrac{y}{\sqrt{x^2+y^2}}\right) \end{pmatrix}$$

The coordinate axis is not X=(1,0,0),Y=(0,1,0),Z=(0,0,1)
u=(ux,uy,uz),v=(vx,vy,vz),w=(wx,wy,wz)の場合は

$$spher(\theta,\phi) = \sin(\theta)\cos(\phi)\mathbf{u} + \sin(\theta)\sin(\phi)\mathbf{v} + \cos(\theta)\mathbf{w}$$

$$\int_{\text{球面全体}} = \int_0^{2\pi}\int_0^{\pi} d\theta d\phi \sin(\theta)$$

$$\int_{\Omega} d\sigma(\omega')$$ Mathematical (physical) meaning

Integral on a hemisphere with the direction of incidence w' of light as normal (integration on a sphere)

$$\int_{\Omega} d\sigma(\vec{\omega}') = \int_{半球面全体} d\theta d\phi = \int_0^{2\pi} \int_0^{\pi/2} d\theta d\phi \sin(\theta) \quad \left(\int_{球面全体} d\theta d\phi = \int_0^{2\pi} \int_0^{\pi} d\theta d\phi \sin(\theta) \right)$$

BRDF = $f_r(x, \vec{\omega}, \vec{\omega}')$ satisfies the following equation as the law of conservation of energy.

$$\int_{\Omega} d\sigma(\omega')\, f_r(x, \omega, \omega') L_i(x, \omega') \cos(\theta) \leq 1$$

In other words, the law states that the sum of reflected light should not exceed the amount of incident light

In addition, the following Helmholtz reciprocity must be satisfied.

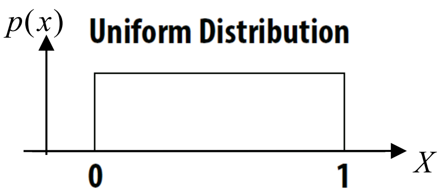
$$f_r(x, \omega, \omega') = f_r(x, \omega', \omega)$$ This is the law that the reflectance is the same even if the incident direction and the reflected direction are swapped

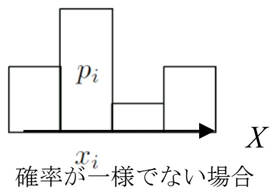
確率密度関数（probability density function(PDF)）

pdf is simply a function that represents the probability of becoming x.

Therefore, the probability density function for the probability, which is said to be a uniform distribution, is as shown in the figure below.

In other words, a random number that appears with the same probability for any x is an example.





確率が一様でない場合

Six-sided die example: $p_i = \frac{1}{6}$

確率が一様：サイコロ

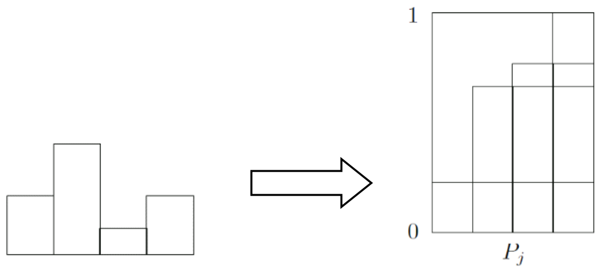
$$p_i \geq 0$$

$$\sum_i p_i = 1$$

累積分布関数（cumulative distribution function, CDF）

cdf represents the probability of a range of x. In other words, it is the accumulation of probabilities and the sum of probability values.

Naturally, if the whole range is 1



Cumulative PDF $P_j = \sum_i^j p_i$

where:

$$0 \leq P_i \leq 1$$

$$P_n = 1$$

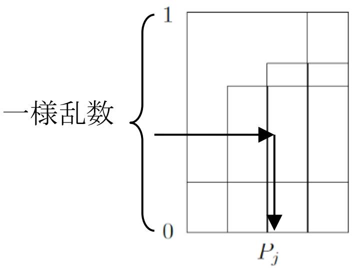
$$P(x) = \int_0^x dx\ p(x) \equiv P_r(X < x)$$

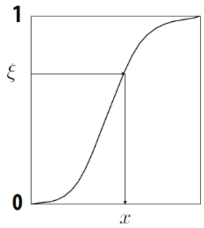
$$P_r(a \leq X \leq b) = \int_a^b dx\ p(x) = \int_0^b dx\ p(x) - \int_0^a dx\ p(x)$$

$$= P(b) - P(a)$$

Once the cumulative distribution function is known, a random number indicating such a distribution can be created using uniform random numbers.

Inverse function method.





Sampling the cosine distribution

$$p(\omega) \propto \cos(\theta)$$

$$x = \cos(2\pi\xi_2)\sqrt{\xi_1}$$

$$y = sin(2\pi\xi_2)\sqrt{\xi_1}$$

$$z = \sqrt{1-\xi_1}$$

$$p(\omega) = c\cos^e(\theta)$$

$$\phi = 2\pi\xi_1$$

$$\theta = \cos^{-1}\left[(1-\xi_2)^{1/(e+1)}\right]$$

Henyey-Greenstein phase function

The probability distribution is $\int_{半球面全体} pdf(\theta,\phi) = \int_0^{2\pi}\int_0^{\pi} d\theta d\phi\; pdf(\theta,\phi)\sin(\theta) = 1$

$pdf(\theta,\phi)$ Find out what kind of random number θ or ϕ of is used to obtain this distribution.

$$pdf(\theta,\phi) = \frac{1}{4\pi}\frac{1-g^2}{\left(1+g^2-2g\cos\theta\right)^{3/2}}$$

From PDF the cumulative probability density function is

$$cdf(\theta,\phi) = \int_{球面全体} PDF = \int_0^{\phi}\int_0^{\theta} d\theta' d\phi' \frac{1}{4\pi}\frac{1-g^2}{\left(1+g^2-2g\cos\theta\right)^{3/2}}\sin(\theta')$$

$$\frac{1-g^2}{4\pi}\int_0^{\phi}\int_0^{\theta} d\theta' d\phi' \frac{\sin(\theta')}{\left(1+g^2-2g\cos\theta\right)^{3/2}}$$

$$= \frac{2\pi(1-g^2)}{4\pi}\int_0^{\theta} d\theta' \frac{\sin(\theta')}{\left(1+g^2-2g\cos\theta\right)^{3/2}}$$

$$= \frac{(1-g^2)}{2}\int_0^{\theta} d\theta' \frac{\sin(\theta')}{\left(1+g^2-2g\cos\theta\right)^{3/2}}$$

$$= \frac{(1-g^2)}{2g}\int_{(11-g)^2}^{1+g^2-2g\cos\theta'} du\, u^{3/2} = \frac{-(1-g^2)}{2\pi g} u^{-1/2}\Bigg|_{(11-g)^2}^{1+g^2-2g\cos\theta'}$$

$$= \frac{1-g^2}{2g}\left(\frac{1}{1-g} - \frac{1}{\sqrt{1+g^2-2g\cos\theta'}}\right)$$

$cdf(\theta,\phi) = F(\theta)G(\phi)$ If you put it with

$$F(\theta) = \frac{1-g^2}{2g}\left(\frac{1}{1-g} - \frac{1}{\sqrt{1+g^2-2g\cos\theta'}}\right)$$

$G(\phi) = 1$

By the inverse function method

$F(\theta) = \xi$ 、 $\theta = F^{-1}(\xi)$

$$
\begin{aligned}
\xi_1 &= \frac{1-g^2}{2g}\left(\frac{1}{1-g} - \frac{1}{\sqrt{1+g^2-2g\cos\theta'}}\right) \\
\Rightarrow \quad \frac{1}{\sqrt{1+g^2-2g\cos\theta'}} &= \frac{1}{1-g} - \frac{2g\xi_1}{1-g^2} \\
\Rightarrow \quad \frac{1}{\sqrt{1+g^2-2g\cos\theta'}} &= \frac{1+g-2g\xi_1}{1-g^2} \\
\Rightarrow \quad \sqrt{1+g^2-2g\cos\theta'} &= \frac{1-g^2}{1+g-2g\xi_1} \\
\Rightarrow \quad 1+g^2-2g\cos\theta' &= \left(\frac{1-g^2}{1+g-2g\xi_1}\right)^2 \\
\Rightarrow \quad \cos\theta' &= \frac{1}{2g}\left(1+g^2-\left(\frac{1-g^2}{1+g-2g\xi_1}\right)^2\right).
\end{aligned}
$$

Monte Carlo integral

It is effective in multiple integrations in discontinuous and high-dimensional fields, which are unmanageable with numerical integrations that are usually used in numerical calculations. The amount of computation is enormous, but it is very simple. It can be calculated at a more realistic cost than numerical integrals, which are usually used in numerical calculations.

$$I = \int_D d\mu(x)\, f(x)$$

In Monte Carlo integrals

$$I = \int_D d\mu(x)\, f(x) \approx \frac{1}{N}\sum_{i=1}^{N}\frac{f(x_i)}{pdf(x_i)}$$

x_i is in the integral range D, pdf Mr./Ms. according to the probability density function. To increase accuracy, increase N. In other words, the simple sum of things can be divided by N.

Sites and documents that were helpful during implementation
http://kagamin.net/hole/simple/index.htm
http://www35.atpages.jp/shocker/memoRANDOM/index.php
The above two sites are grateful sites that have become teachers.
http://blog.tobias-franke.eu/2014/03/30/notes_on_importance_sampling.html
http://scratchapixel.com/old/lessons/3d-basic-lessons/lesson-17-monte-carlo-methods-in-practice/monte-carlo-simulation-2/
http://http.developer.nvidia.com/GPUGems3/gpugems3_ch20.html
http://freespace.virgin.net/hugo.elias/graphics/x_polybm.htm
http://www.sjbrown.co.uk/2012/07/15/bidirectional-path-tracing-in-participating-media/
http://candela.stanford.edu/cs348b-13/
http://www.midnightkite.com/color.html

Philip Dutré
"Global Illumination Compendium"
Computer Graphics, Department of Computer Science Katholieke Universiteit Leuven

"Lecture 16 Global Illumination and Path Tracing"
Image Synthesis Stanford CS348b, Spring 2013

Torsten Moller
James T. Fight
"THE RENDERING EQUATION"
California Institute of Technology Pasadena, Ca. 91125

Michael Doggett
"Path Tracing"
Department of Computer Science Lund university

Eric Lafortune
"Mathematical Models and Monte Carlo Algorithms for Physically Based Rendering"
Department of Computer Science, Faculty of Engineering Katholieke Universiteit Leuven

Petri Häkkinen
"Monte Carlo Path Tracing"
46561N

H. Dammertz1　J. Hanika　A. Keller
"Shallow Bounding Volume Hierarchies for Fast SIMD Ray Tracing of Incoherent Rays"

Christopher Kulla1 and Marcos Fajardo
"Importance Sampling Techniques for Path Tracing in Participating Media"
1Sony Pictures Imageworks, Culver City,　2Solid USAAngle, Madrid, Spain

"The Henyey-Greenstein phase function."

Steve Marschner
"Multiple scattering"
Cornell CS 6630 Fall 2009

"Participating Media"
CMPT 461/761 Image Synthesis

Steve Marschner
"Volumetric Path Tracing"
Cornell University CS 6630 Spring 2012, 8 March

Leonardo Da Vinci
"Light Transport in ParticipatingMedia"

Wojciech Jarosz
"State of the Art in Photon Density Estimation Participating Media Basics"

Diego Gutierrez Henrik Wann Jensen Srinivasa G. Narasimhan Wojciech Jarosz
"Scattering Course notes"
SIGGRAPH Asia 2008

patrick coleman
"subsurface scattering"
january 30, 2006

Olav Madsen Anders Kyster
"Real-time Rendering of Sub-surface Scattering"
IT University of Copenhagen, August 1st, 2002

Gladimir V. G. Baranoski Aravind Krishnaswamy Bradley Kimmel
"Revisiting the Foundations of Subsurface Scattering"
Natural Phenomena Simulation Group, School of Science, Technical Report CS-2003-45ComputerUniversity of Waterloo, Canada

Tom Mertens1 Jan Kautz Philippe Bekaert Frank Van Reeth Hans-Peter Seidel
"Efficient Rendering of Local Subsurface Scattering"

"CIE XYZ"
Janusz Ganczarski

Chris Wyman Peter-Pike Sloan Peter Shirley
"Simple Analytic Approximations to the CIE XYZ Color Matching Functions"
NVIDIA

Brian Smits——————————————
"An RGB to Spectrum Conversion for Reflectances"
University of Utah

Michael Ashikhmin Peter Shirley
"An Anisotropic Phong BRDF Model"
August 13, 2000

David Cline Parris Egbert
"A Practical Introduction to Metropolis Light Transport"

E. Veach, L. J. Guibas
"Metropolis light transport,"
In Proc. of SIGGRAPH '97, pp. 65-76 (1997)

Kelemen C., L. Szirmay-Kalos, Antal G., F. Csonka,
"A Simple and Robust Mutation Strategy for the Metropolis Light Transport Algorithm,"
Computer Graphics Forum, 21(3) (Proc. of EUROGRAPHICS 2002), pp. 531-540 (2002)

D. Cline, J. Talbot, P. Egbert,
"Energy redistribution path tracing,"
ACM Transactions on Graphics, 24(3) (Proc. SIGGRAPH 2005), pp. 1186-1195 (2005)

Christopher Batty
"Implementing Energy Redistribution Path Tracing"
Department of Computer Science
The University of British Columbia