Physically Based Rendering: Phong BRDF

Carlos Fuentes

What’s Physically Based Rendering ?

- Physically Based Rendering aka **PBR** are techniques focused on understanding of physics that are used to model how **light and materials** (that have different physical properties) **interacts**.
- Since this interactions are sometimes very complex and happens in a very fine level, **statistical models** are often used to manage this complexity.

A model for light and material interaction

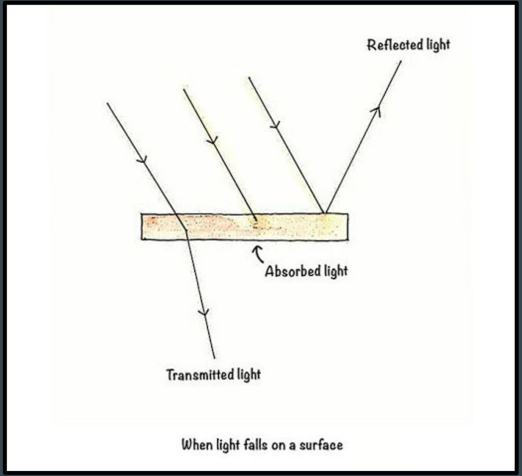
- Material and light interactions are described mathematically using a Bidirectional Scattering Distribution Function aka **BSDF**.
- This functions describes how **light scatters** upon contact with a surface based on surface properties
- Actually this functions are an **statistical model** to describe how **likely** the light is scattered in a specific **outgoing direction**.

Bidirectional Scattering Distribution Functions

- **BSDF** sounds complicated but it really means:
  - **Bidirectional** refers that light comes in one direction and goes out in other direction
  - **Scattering** means that light coming from one direction can be splitted into a range of outgoing directions
  - **Distribution** refers to scattered light is described by an statistical distribution function. This functions describes how **likely** the light is **scattered** in certain outgoing direction based on physical properties of the surface

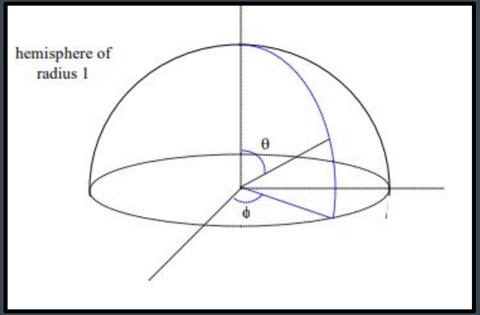
Bidirectional Reflectance Distribution Functions

- BSDFs are composed of two other functions:
  - Bidirectional Reflectance Distribution Functions aka **BRDF** are the part of BSDF that describes how light is **reflected**
  - Bidirectional Transmittance Distribution Functions aka **BTDF** are the part of BSDF that describes how light is **transmitted** through a surface
- We aim to model commonly encountered surfaces so, we will ignore BTDFs and focus on **BRDFs**



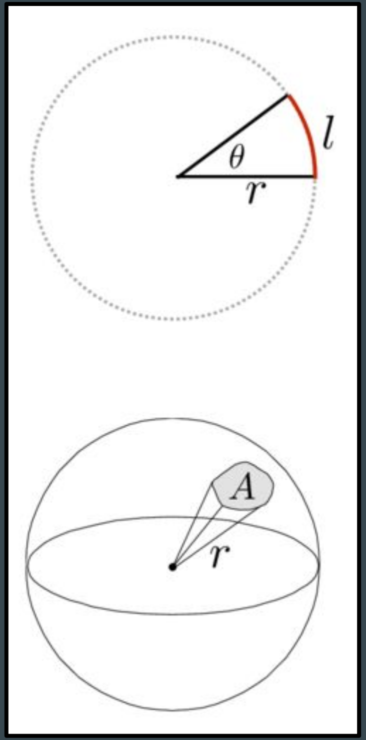
Radiometry fundamentals

- In **radiometry** (science of measurement amounts of light), measurements are taken with respect to the **unit hemisphere** surrounding a point in a surface.
- We want to measure amount of light going through one direction **ω.** (Note that direction ω can be expressed using two angles **ϕ** and ***θ*** aka **spherical coordinates**)



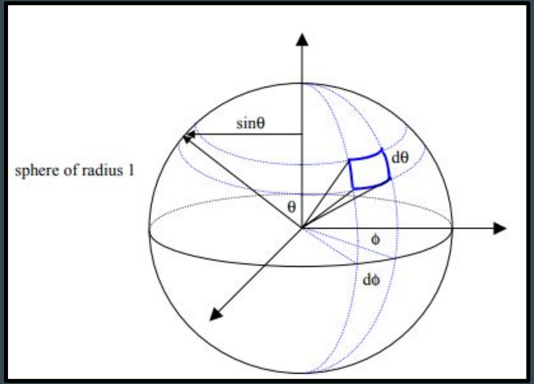
Radiometry fundamentals: Solid Angle

- Radiometry measures light passing through a **solid angle**
- A solid angle is an extension of **2D angle** to **3D**.
- **Plane Angle** (Θ) is an arc length (l) on a circle divided by radius (r)
  - $\theta = \frac{l}{r}$
  - The length of a circle is $2\pi r$ so a circle has **2π radians**
- **Solid angle** (□) is an area on a sphere divided by radius squared
  - $\Omega = \frac{A}{r^2}$
  - The area of a sphere is $4\pi r^2$ so a sphere has **4π steradians (sr)**



Radiometry fundamentals: Differential Solid Angle

- Generic solid angles has no limitations about **shape**.
- Some of lighting equations are integrals around a hemisphere so we are specially interested in **differential solid angles**
- You can see a differential solid angle as the **small square area** on the unit sphere defined by differential angles **dϕ** and **dθ** in spherical coordinates



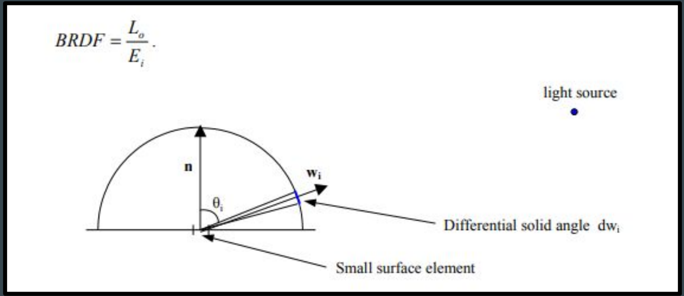
Radiometry fundamentals: Radiance and Irradiance

- **Irradiance (E_i)** measures light arriving at a small surface area from all incoming directions.
- **Radiant exitance (E_o)** measures light leaving a small surface area to all outgoing directions.
- **Incoming Radiance (L_i)** measures incoming light passing through a **solid angle** from a specific direction
- **Outgoing Radiance (L_o)** measures outgoing light passing through a **solid angle** from a specific direction

BRDF: Definition

- BRDF is the ratio of the **outgoing radiance** (leaving a small surface area) to the **differential incoming irradiance** (arriving to the same surface area from a differential solid angle):

$$BRDF = \frac{L_o}{\delta Ei}$$

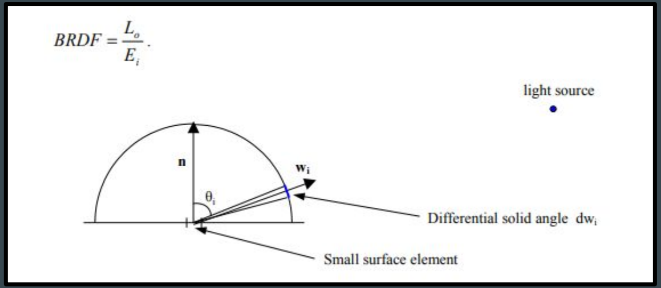


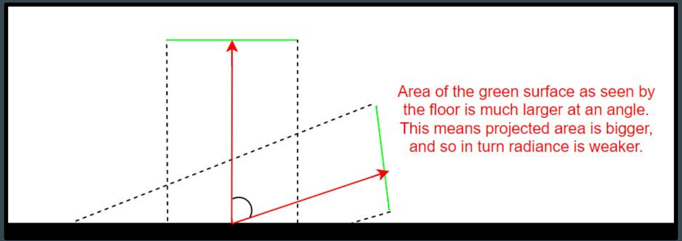
BRDF: Solid angle projection

- We can rewrite differential irradiance in terms of incoming **radiance L_i** arriving at dw_i **solid angle** and then **projected** into our small surface area:

$$\delta Ei = L_i \cdot cos\theta_i \cdot dwi$$

- Similar to what happens with Lambert, **projection** of solid angle means scaling by **$\cos\theta_i$** from surface normal to light direction.



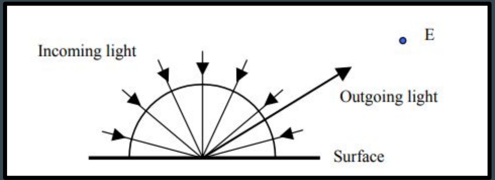


The Rendering Equation

- For rendering, we aim to use BRDF to compute **outgoing illumination** from a small surface point into a specific outgoing direction, usually the view direction.
- We can rewrite BRDF equation to compute **outgoing illumination** from one differential incoming direction

$$L_o = BRDF \cdot L_i \cdot cos\theta_i \cdot dwi$$

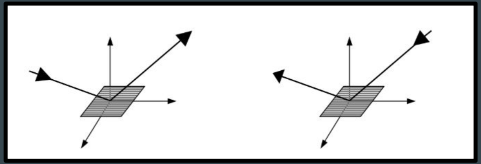
- And, finally, for all possible incoming directions



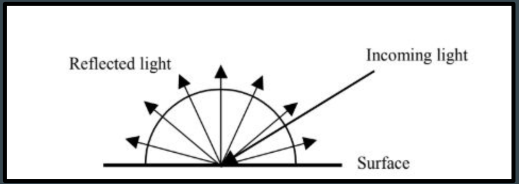
$$L_o = \sum_{\Omega} BRDF \cdot L_i \cdot cos\theta_i \cdot dwi$$

Physically based BRDF constraints

1. **Helmholtz Reciprocity rule**: If we change incoming and outgoing directions we obtain the same BRDF



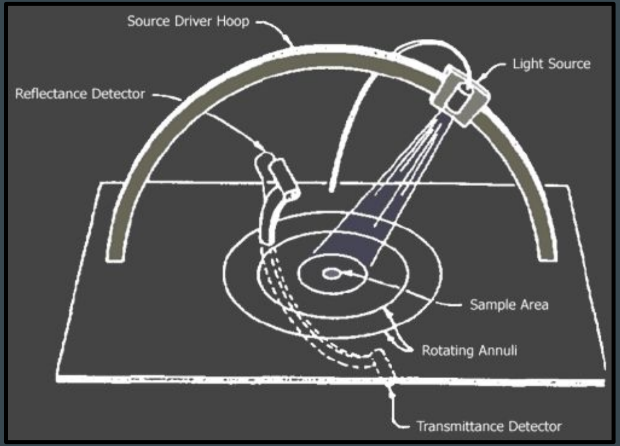
2. **Energy conservation law:** The total reflected light from a single incoming direction can not exceed the amount of light arrived ⇒ The sum of the BRDF times projected outgoing solid angle is less or equal to 1.



$$\int_{out} BRDF \cdot cos\theta_o \cdot dw_o \leq 1$$

Acquiring BRDFs

- There are **a lot of BRDFs**. Each of them are used to describe different physical phenomena: **fresnel reflections, subsurface scattering, microgeometry, etc.**
- BRDFs can be **captured** using sensitive instruments
- BRDFS can also be **analytic models**. Analytic statistical models are commonly used in video games



Gonioreflectometer

Lambert and Phong BRDF

- Can we use Lambert and Phong functions as a **BRDF** for modelling reflections?

$$L_o = BRDF \cdot L_i \cdot cos\theta_i \cdot dwi$$

- Our shading equation

$$L_o = (C_d + (V \cdot R)^n) \cdot L_i \cdot (N \cdot L)$$

- So, diffuse **Lambert BRDF**

$$BRDF_{lambert} = C_d$$

- And specular **Phong BRDF**

$$BRDF_{phong} = (V \cdot R)^n$$

Energy conserving Phong

- Lambert and Phong are not energy conserving. Can we multiply them by a **normalization factor** to ensure energy conservation ? Remember **energy conservation** means:

$$\int_{out} BRDF \cdot cos\theta_o \cdot dw_o \leq 1$$

- Lambert

$$\int_{out} C_d \cdot cos\theta_o \cdot dw_o \leq 1 \Rightarrow C_d \cdot \pi \leq 1$$

$$\Rightarrow Normalization = \frac{1}{\pi}$$

Energy conserving Phong

- Phong

$$\int_{out} (V \cdot R)^n \cdot cos\theta_o \cdot dw_o \leq 1$$

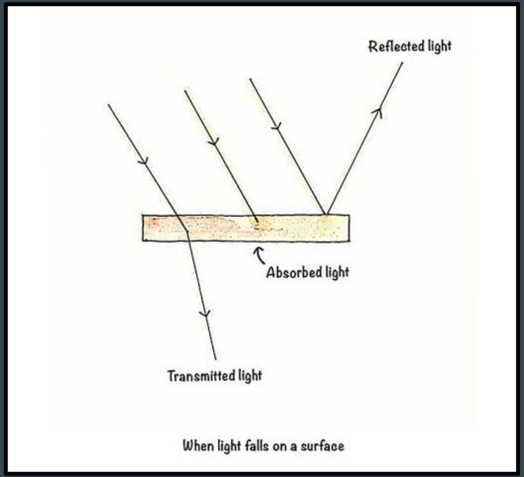
$$\Rightarrow Normalization = \frac{n+2}{2\pi}$$

- So, our final Render Equation with **energy conservation** factors is

$$L_o = \left(\frac{C_d}{\pi} + \frac{n+2}{2\pi} \cdot (V \cdot R)^n \right) \cdot L_i \cdot (N \cdot L)$$

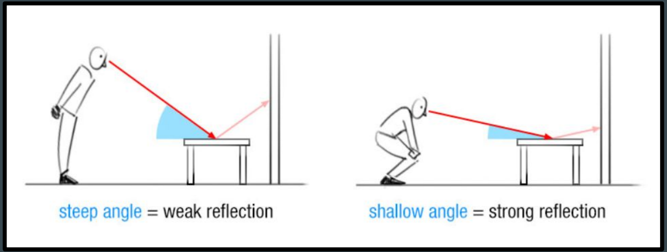
How much light is reflected ?

- Following **energy conservation** rules we know that when light arrives to a surface:
  - Some light is **reflected**
  - Some light is **absorbed**
  - Some light is **transmitted**
- But, what's the **amount of light reflected**?

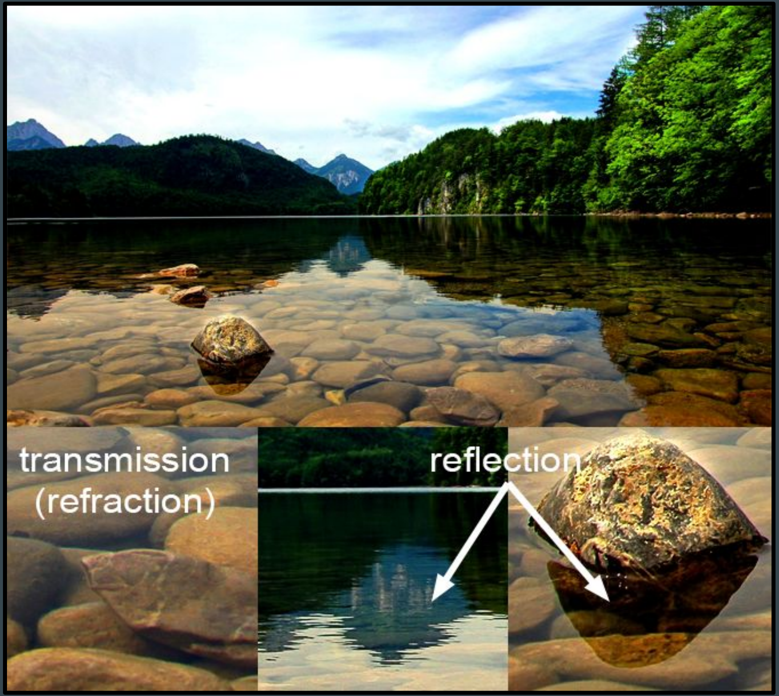


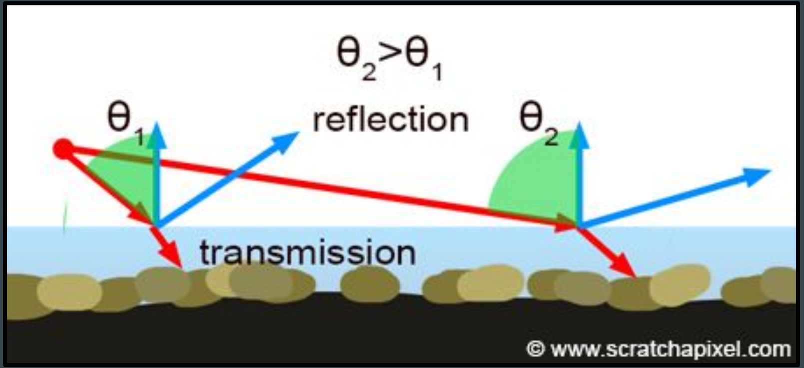
Fresnel effect

- The amount of light reflected depends only on
  - The **refractive index** of the surface
  - The light incoming angle: Light arriving at glancing angles (perpendicular to normal) is reflected more ⇒ **Fresnel effect**



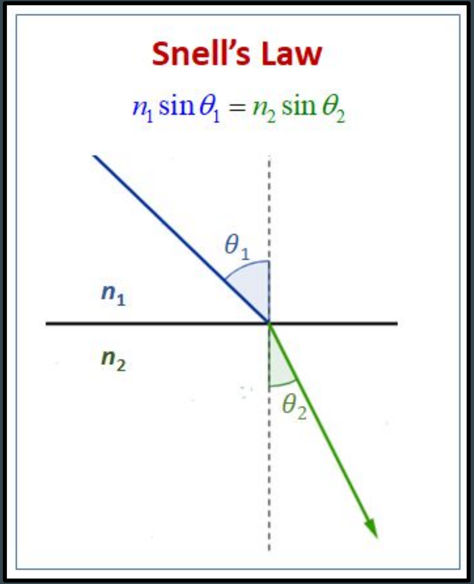
Fresnel effect





Fresnel equations: Dielectrics

- Fresnel equations depends on surface types:
  - **Dielectrics**: glass, plastic, ceramics. Transmits electricity but without conducting it. Having n_1, n_2 refractive indexes (n_1 is usually the air)
    - **Parallel polarized light** $$r_{\parallel} = \frac{n_1 cos\theta_2 - n_2 cos\theta_1}{n_1 cos\theta_2 + n_2 cos\theta_1}$$
    - **Perpendicular polarized light** $$r_{\perp} = \frac{n_1 cos\theta_1 - n_2 cos\theta_2}{n_1 cos\theta_1 + n_2 cos\theta_2}$$
    - **Unpolarized light** $$r = \frac{r_{\perp}^2 + r_{\parallel}^2}{2}$$



Fresnel equations: Metals

- **Conductors:** metals. Don't transmits light, absorbs some light transferred as heat. The amount of absorbed light is modelled using an **absorption coefficient k**.

  - **Parallel polarized light** $$r_{\parallel}^2 = \frac{(n^2+k^2)cos\theta_1 * 2 - 2ncos\theta_1 + 1}{(n^2+k^2)cos\theta_1 * 2 + 2ncos\theta_1 + 1}$$

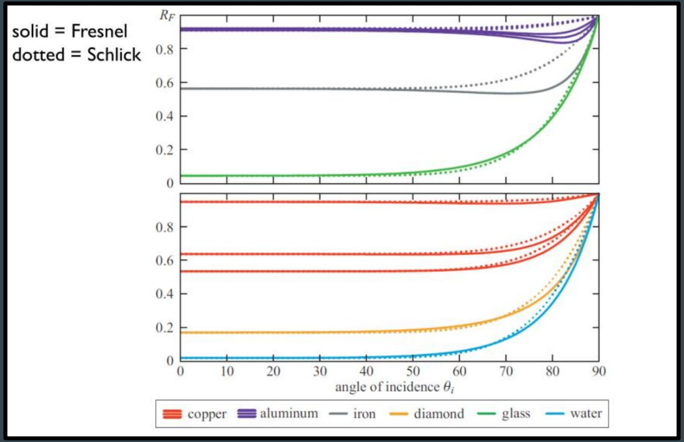
  - **Perpendicular polarized light** $$r_{\perp}^2 = \frac{(n^2+k^2)cos\theta_1 * 2 - 2ncos\theta_1 + cos\theta_1^2}{(n^2+k^2)cos\theta_1 * 2 + 2ncos\theta_1 + cos\theta_1^2}$$

Schlick's approximation

- Fresnel equations are **too complex** to use for rendering
- Plotting fresnel equation **$R_f(\theta_i)$ from incidence angle θ_i**
  - **$R_f(0^o)$** (perpendicular to surface normal) depends of each material
  - **$R_f(\theta_i)$** remains stable until 60^0-70^0 where fresnel begins an approach to 1 until 90° (glancing angles)
- **Schlick approximation**

$$R_f(\theta_i) \approx R_f(0^o) + (1 - R_f(0^o))(1 - cos\theta_i)^5$$

- Where $R_f(0^o) = \left(\frac{n-1}{n+1}\right)^2$ and **n is the index of refraction**



Fresnel as specular color

- Our Render equation with fresnel is

$$L_o = \left(\frac{C_d}{\pi} + \frac{n+2}{2\pi} \cdot R_f(\theta_i) \cdot (V \cdot R)^n\right) \cdot L_i \cdot (N \cdot L)$$

- where **C_d (diffuse color) and n (shininess) and $R_f(0°)$** are material properties.
- $R_f(0°)$ is represented as a color, like a **specular color**
- $R_f(0°)$ is grayscale for dielectrics and can be colored for metals. It can be computed from databases like [this]

Material	$F(0°)$ **(Linear)**	$F(0°)$ **(sRGB)**	**Color**
Water	0.02,0.02,0.02	0.15,0.15,0.15	
Plastic / Glass (Low)	0.03,0.03,0.03	0.21,0.21,0.21	
Plastic High	0.05,0.05,0.05	0.24,0.24,0.24	
Glass (High) / Ruby	0.08,0.08,0.08	0.31,0.31,0.31	
Diamond	0.17,0.17,0.17	0.45,0.45,0.45	
Iron	0.56,0.57,0.58	0.77,0.78,0.78	
Copper	0.95,0.64,0.54	0.98,0.82,0.76	
Gold	1.00,0.71,0.29	1.00,0.86,0.57	
Aluminum	0.91,0.92,0.92	0.96,0.96,0.97	
Silver	0.95,0.93,0.88	0.98,0.97,0.95	

Combine diffuse and specular

- From **energy conservation rule** we know that the more specular the less diffuse reflection

$$C_d + R_f(\theta_i) \leq 1$$

- So, energy conservation rule could be easily accomplished using diffuse color

$$C'_d = C_d \cdot (1 - R_f(\theta_i))$$

- But this brokes **Helmholtz Reciprocity rule** for diffuse BRDF because $R_f(\theta_i)$ depends only on incoming light. We can use $R_f(0°)$ as an approximation without this dependency

$$C'_d = C_d \cdot (1 - R_f(0^o))$$

Engine integration

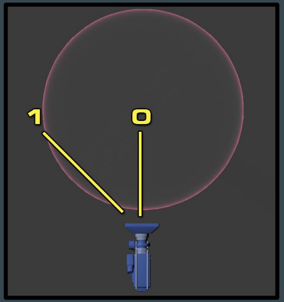
- Replace phong shading equation for **Normalised Phong BRDF with Fresnel** equation

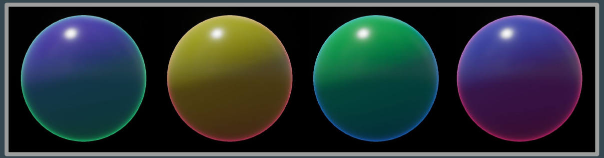
$$L_o = \left(\frac{C_d \cdot (1 - R_f(0^o))}{\pi} + \frac{n+2}{2\pi} \cdot R_f(\theta_i) \cdot (V \cdot R)^n \right) \cdot L_i \cdot (N \cdot L)$$

- Note: For the moment, in our final shading equation, keep adding **ambient color as a constant light** from environment
- Note: Is usual to find implementations without **π divisions**, they are compensated in diffuse and specular colors.

Expected results

- Using energy conserving Phong will help you to create scenes with **coherent lighting** and without empirical tweek of K_d and K_s constants.
- Fresnel effect is visible at **glancing angles** and is more visible on dielectric objects





References

- Useful links for learning more about BRDFs and Normalized Phong
    - BRDFs by Chuck Moidel here
    - An Introduction to BRDF-Based Lighting by Chris Wynn here
    - The Rendering Equation and BRDFs here
    - Phong energy conservation here
    - About Fresnel equations here