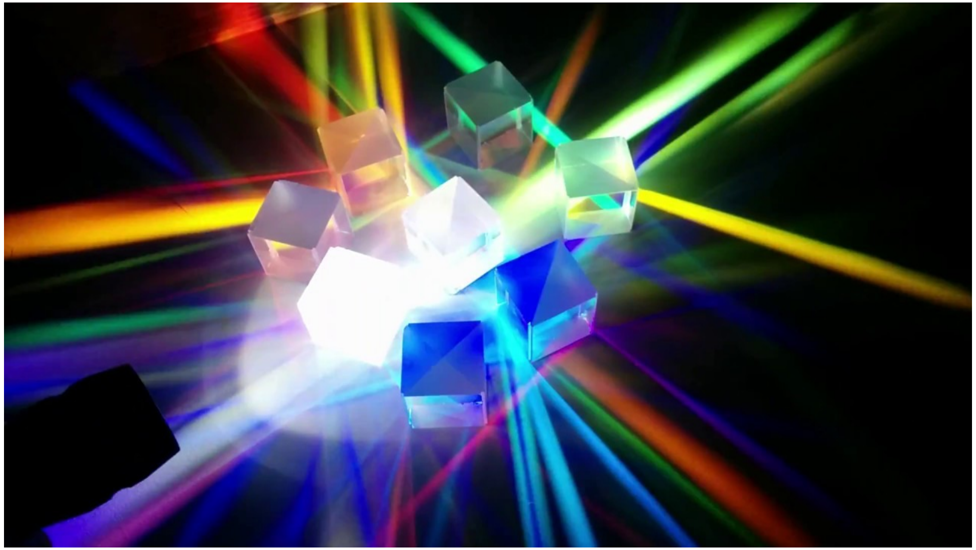
advanced material

CS4515 3D Computer Graphics and Animation



Ricardo Marroquim

Delft University of Technology (TU Delft)

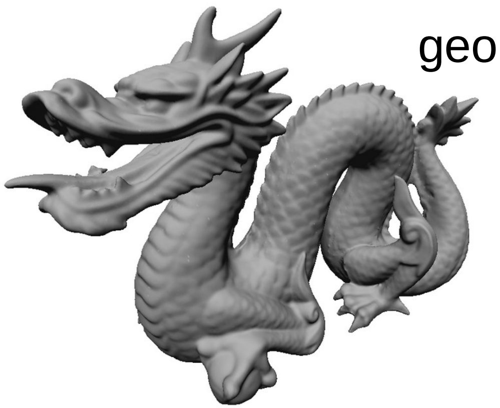
today

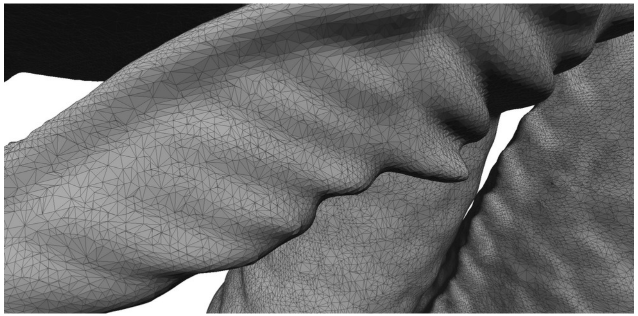
- light-matter interaction
- Physically Based Rendering/Shading
- common implementations and usage

https://www.gamersnexus.net/gg/1866-what-is-pbr-cryengine-star-citizen

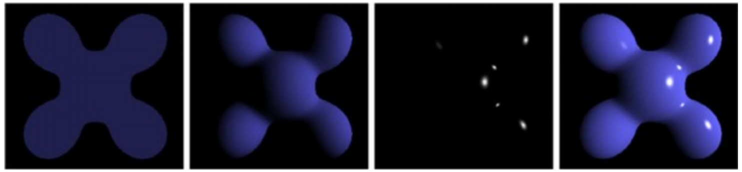
scales

geometry

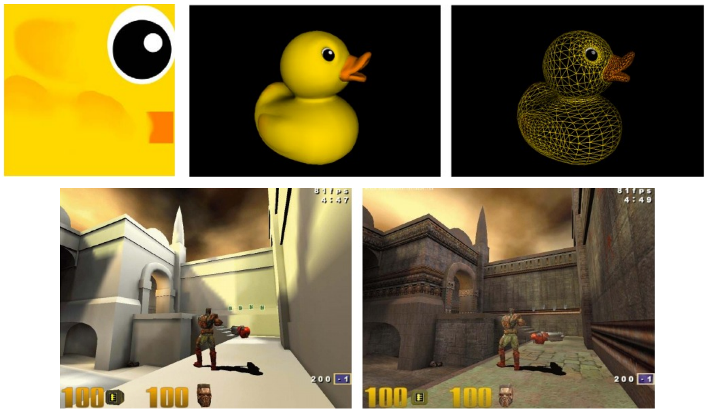


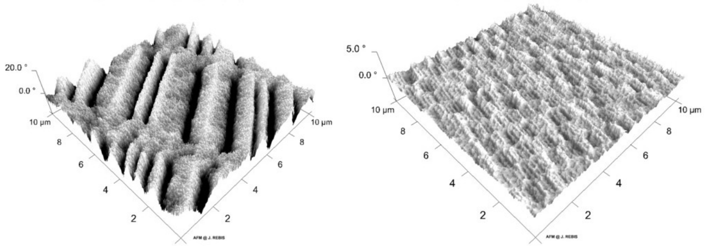


shading



textures



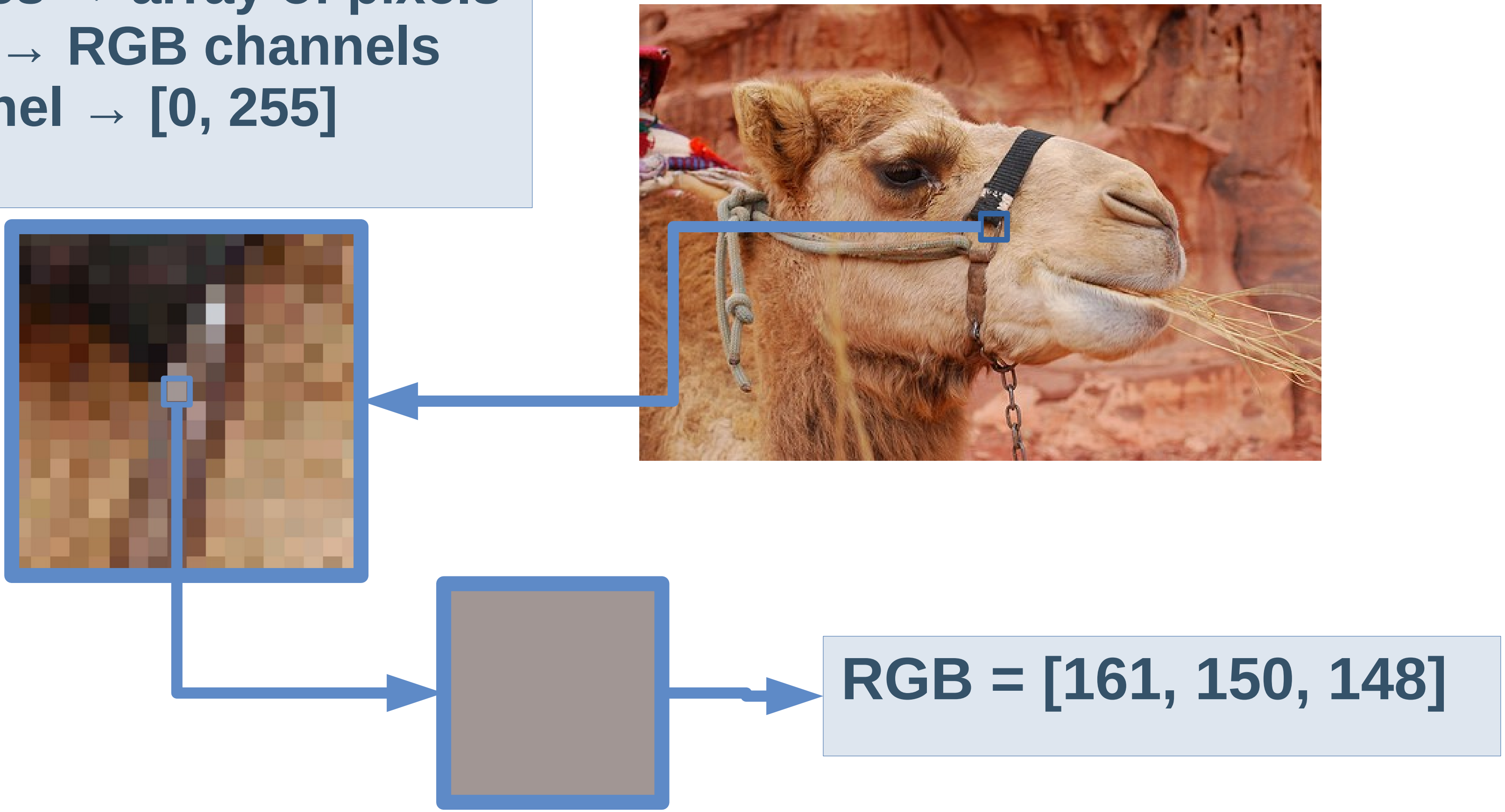


today, we get even closer to the surface

images

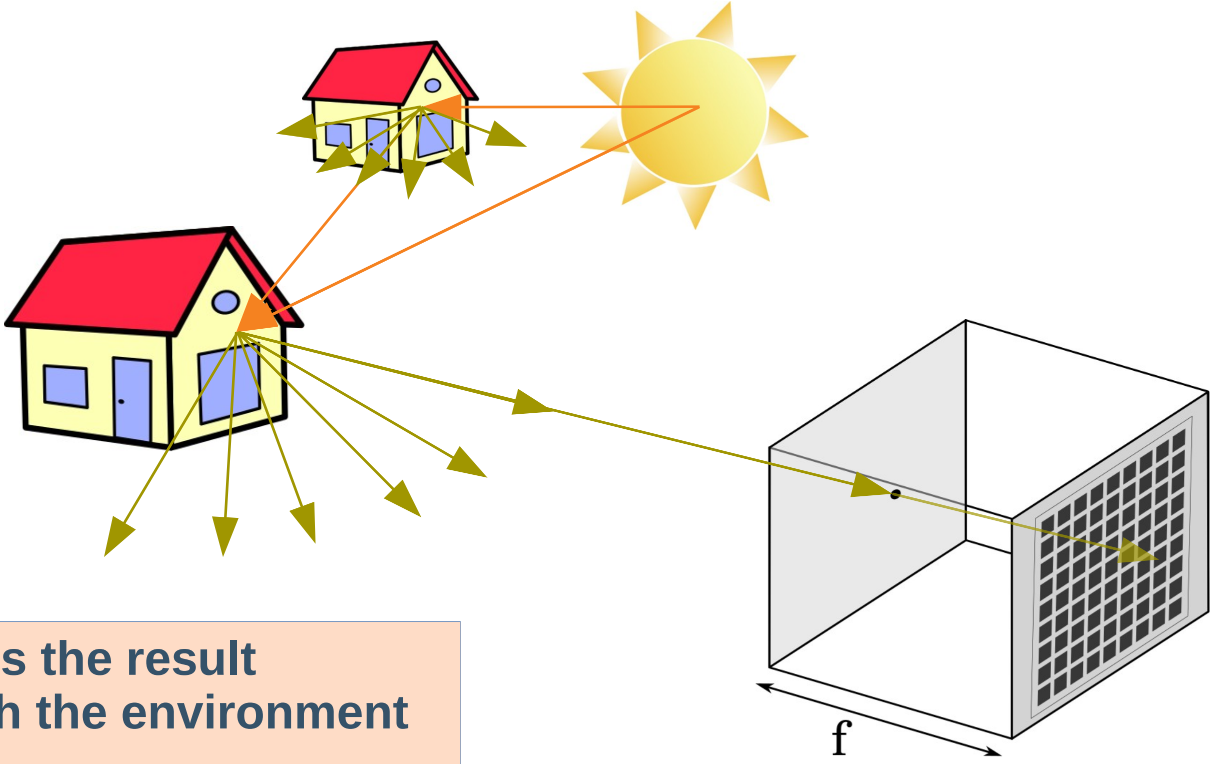
pixel colour:

- **images → array of pixels**
- **pixel → RGB channels**
- **channel → [0, 255]**



colour

colour comes from light interaction



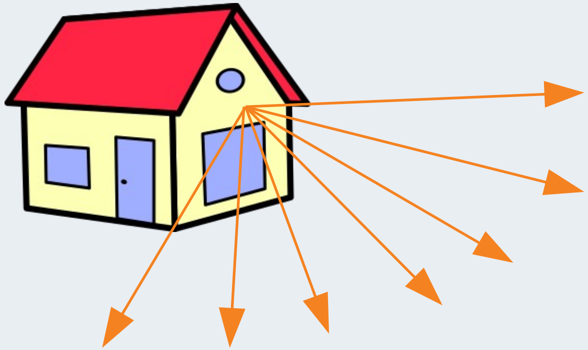
pixel colour represents the result from the light interaction with the environment

colour

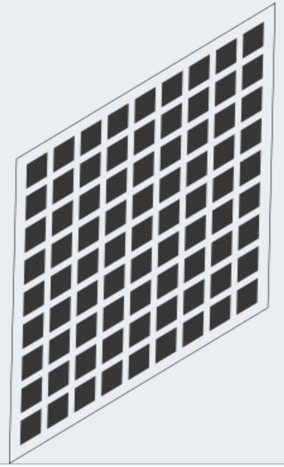
pixel colour of a photo depends on:

light source
- how it emits light

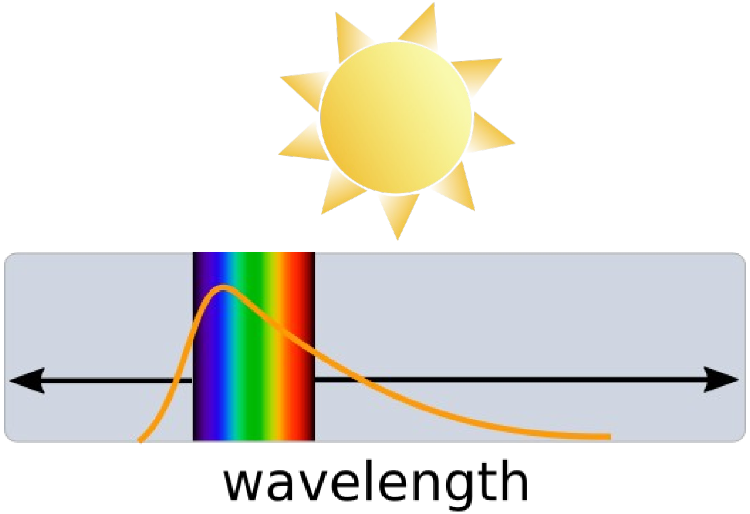
surface
- how it reflects light

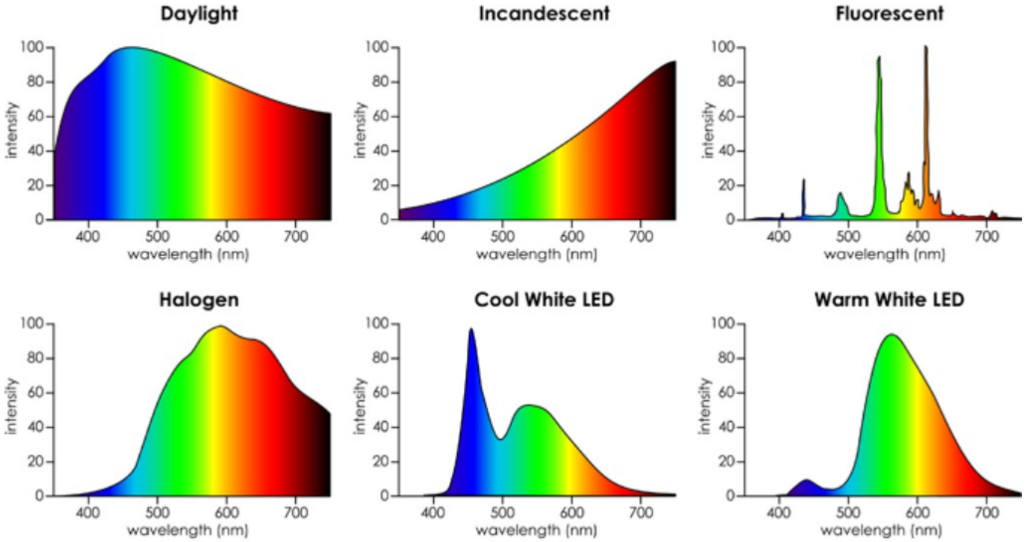


sensor
- how it captures colour



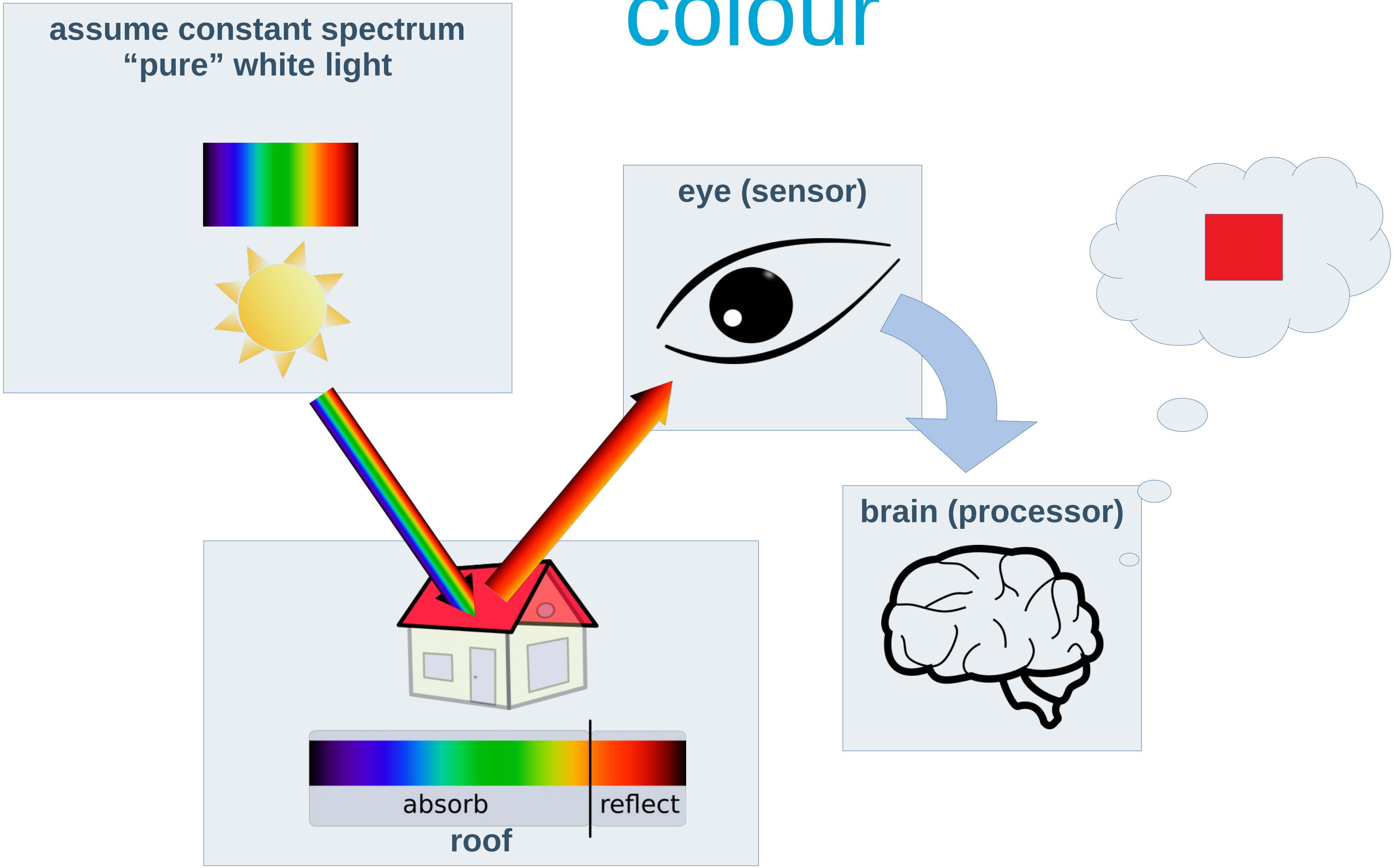
light spectrum



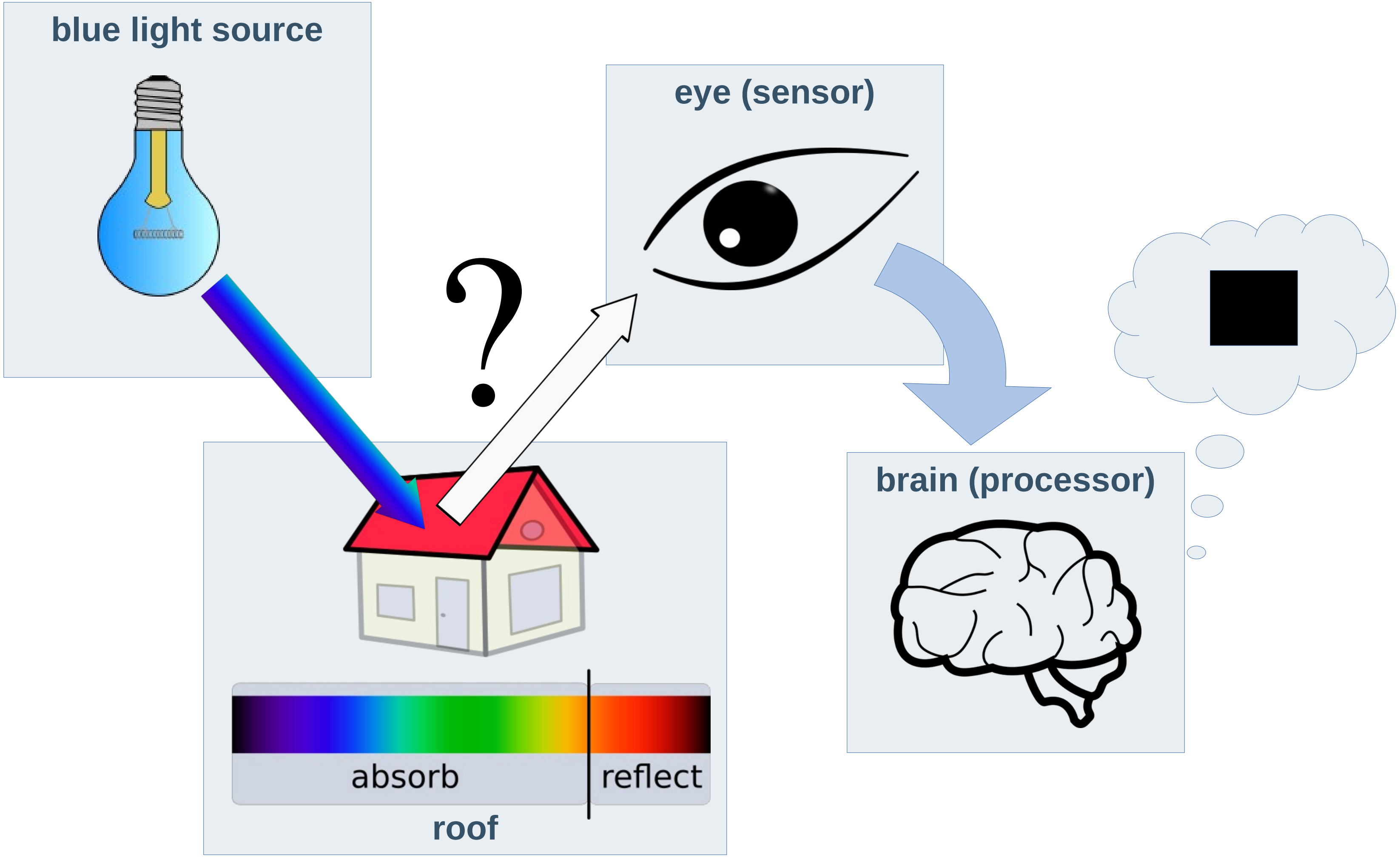


https://www.jfychicago.org/#full-spectrum-led-grow-lights-images

colour

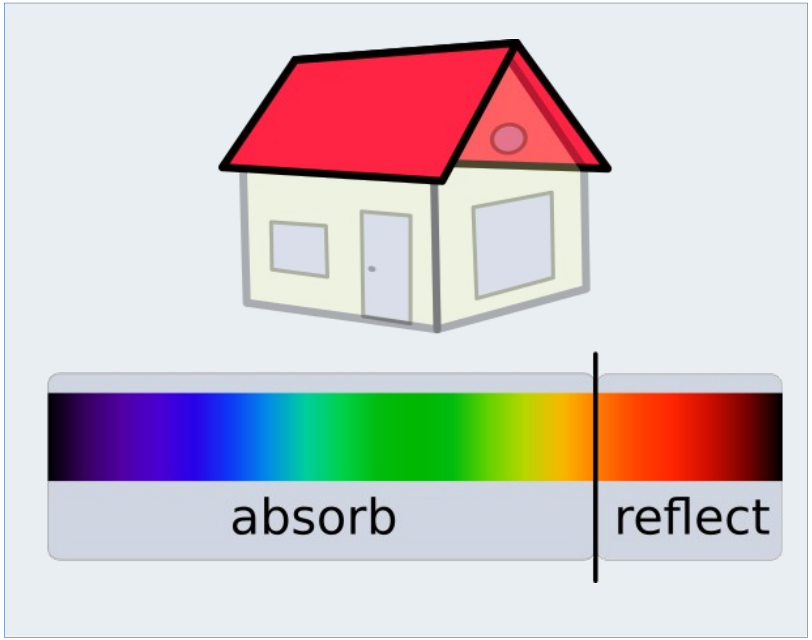

assume constant spectrum
“pure” white light
eye (sensor)
brain (processor)
absorb
reflect
roof

colour



blue light source
eye (sensor)
?
brain (processor)
absorb
reflect
roof

reflected wavelength

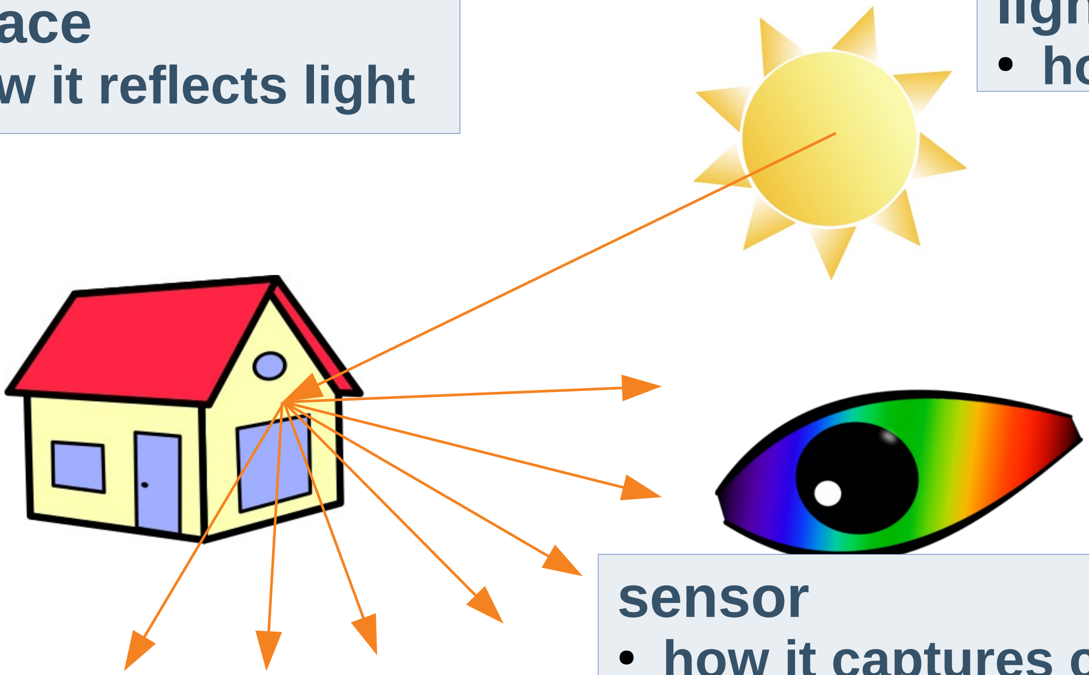


wavelength function is actually continuous

colour = perception

surface
- **how it reflects light**

light source
- **how it emits light**



processor
- **how it interprets colour**

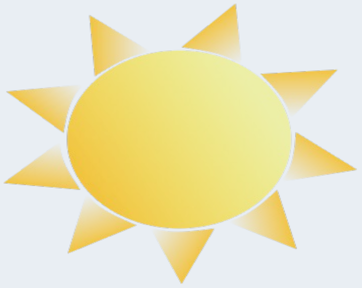
sensor
- **how it captures colour**

eyes capture light
brain processes light information

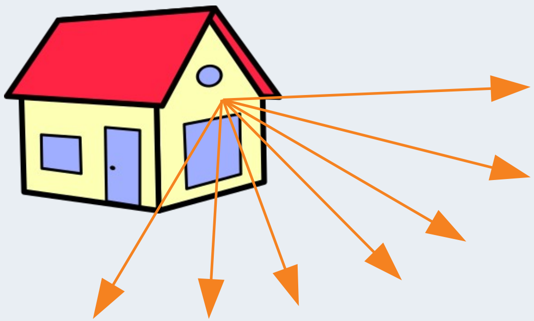
appearance

how to isolate the material (surface)?

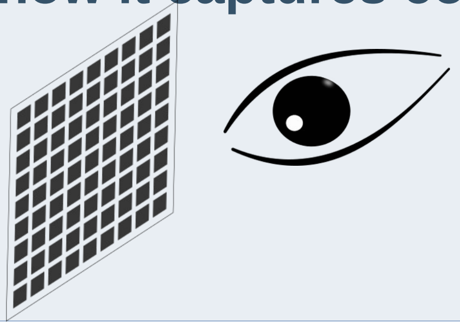
light source
- how it emits light



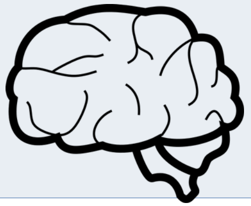
surface
- how it reflects light



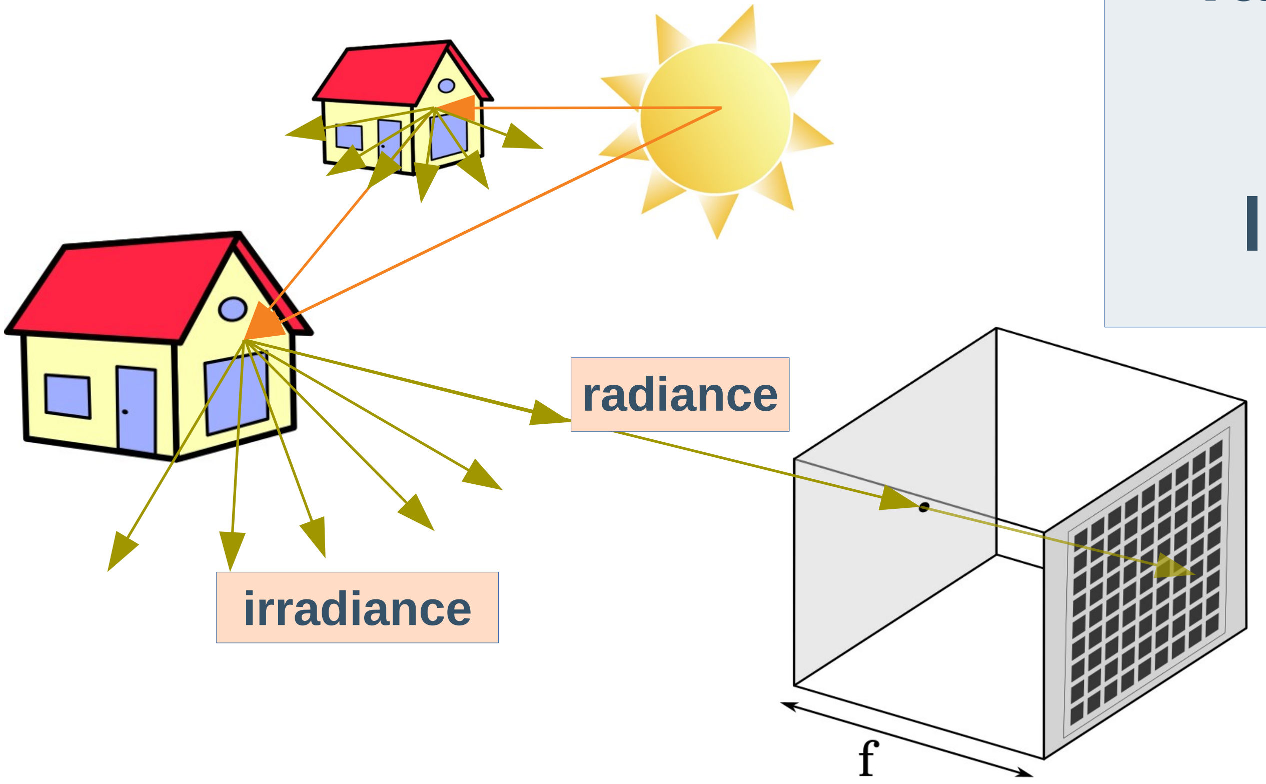
sensor
- how it captures colour



processor
- how it interprets colour



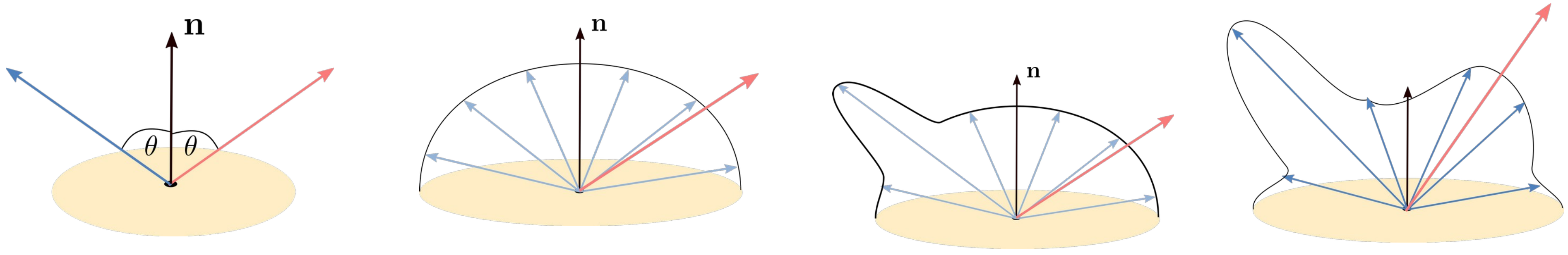
~~colour~~ radiometry



radiance captured by sensor comes from light interaction

reflectance functions (BRDF)

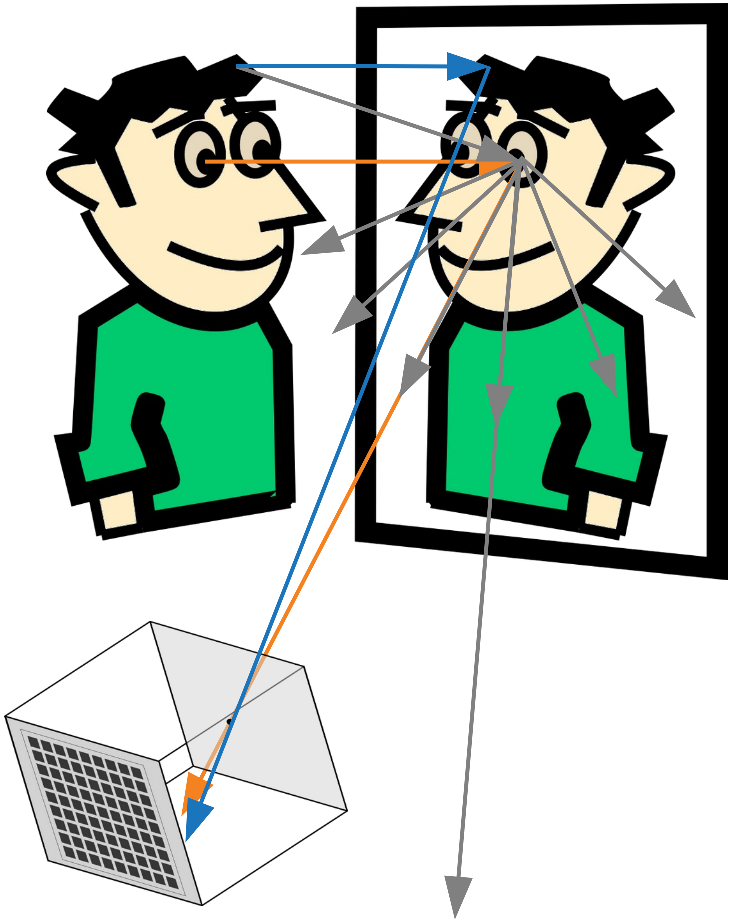
how light is reflected?



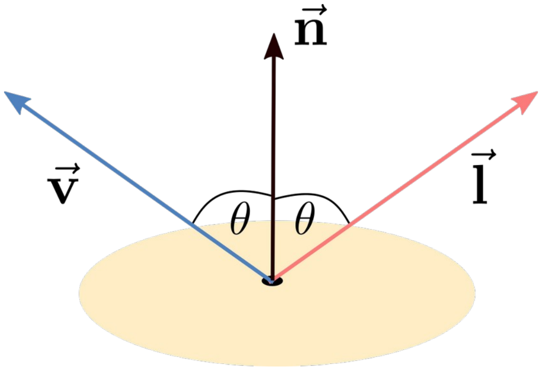
$$f_r(p, \mathbf{v}, \mathbf{l})$$

describe the material appearance independently of the environment

perfect mirror

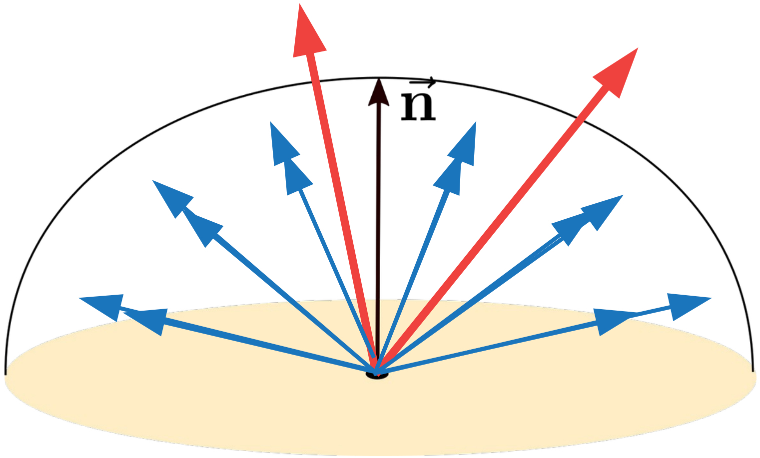


light reflects in a single direction

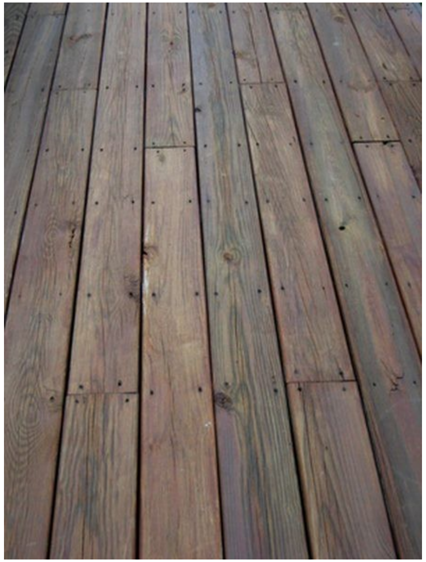


what would happen otherwise?

perfect diffuse



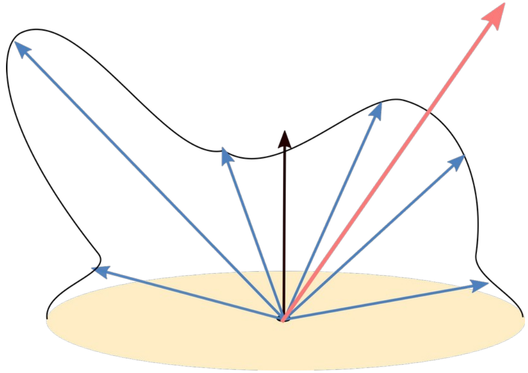
light reflects equally in all directions

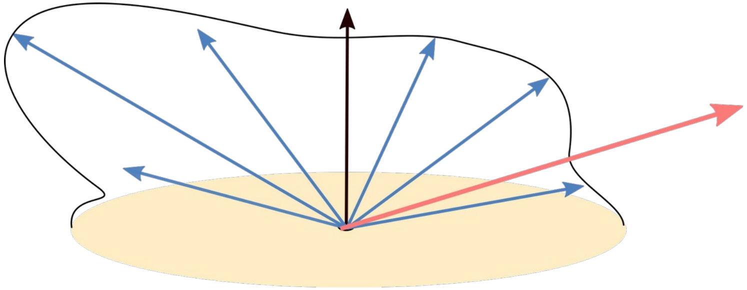


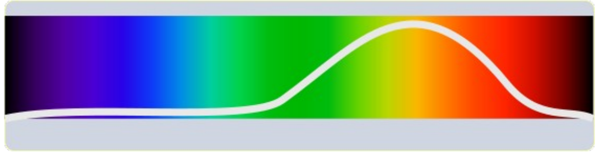
note: intensity of reflection may change

- no highlights
- same perception independent of viewing angle
- depends only on incoming direction

glossy



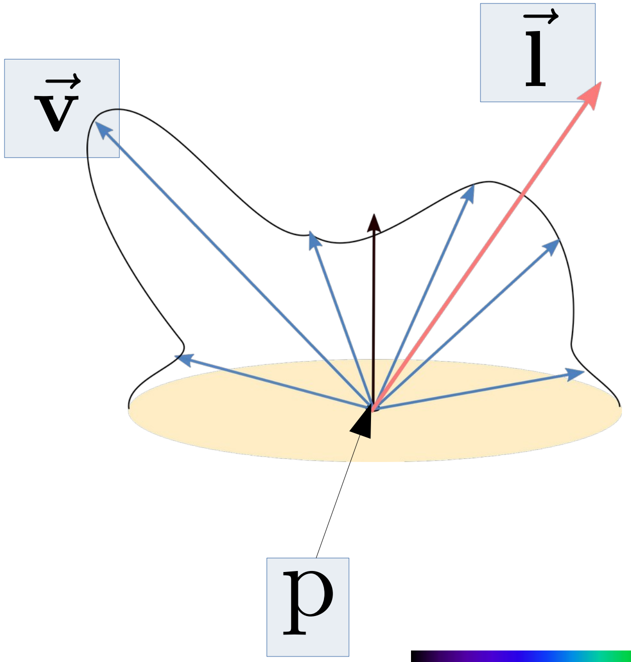


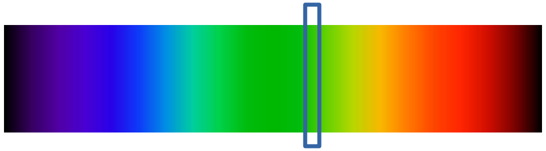


- **note: these functions are also wavelength dependent**

reflectance function

BRDF (Bidirectional Reflectance Distribution Function)





input (depends on):
- **point on surface**
- **incoming light direction**
- **outgoing light direction**
- **wavelength**

output:
- **ratio of reflected light**

putting it all together

captured radiance comes from light interaction

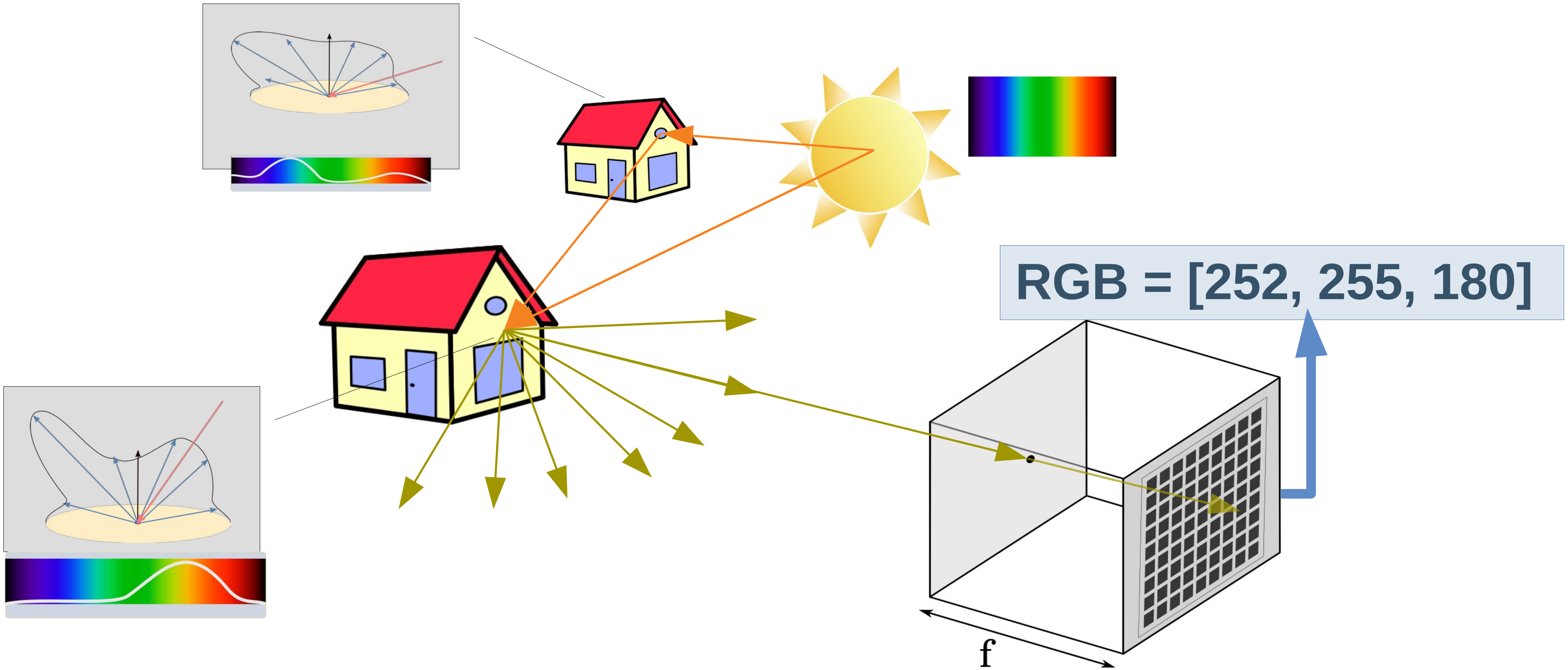
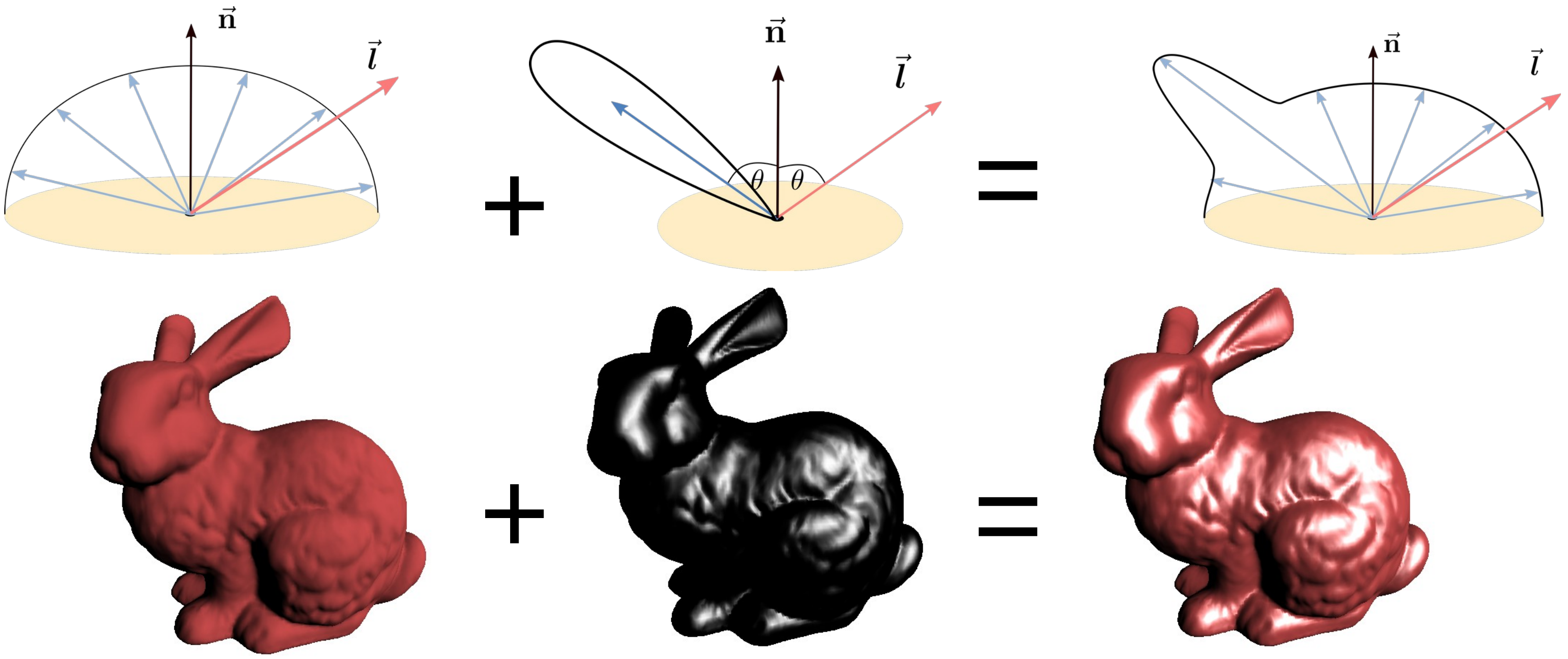


photo captures “radiance”

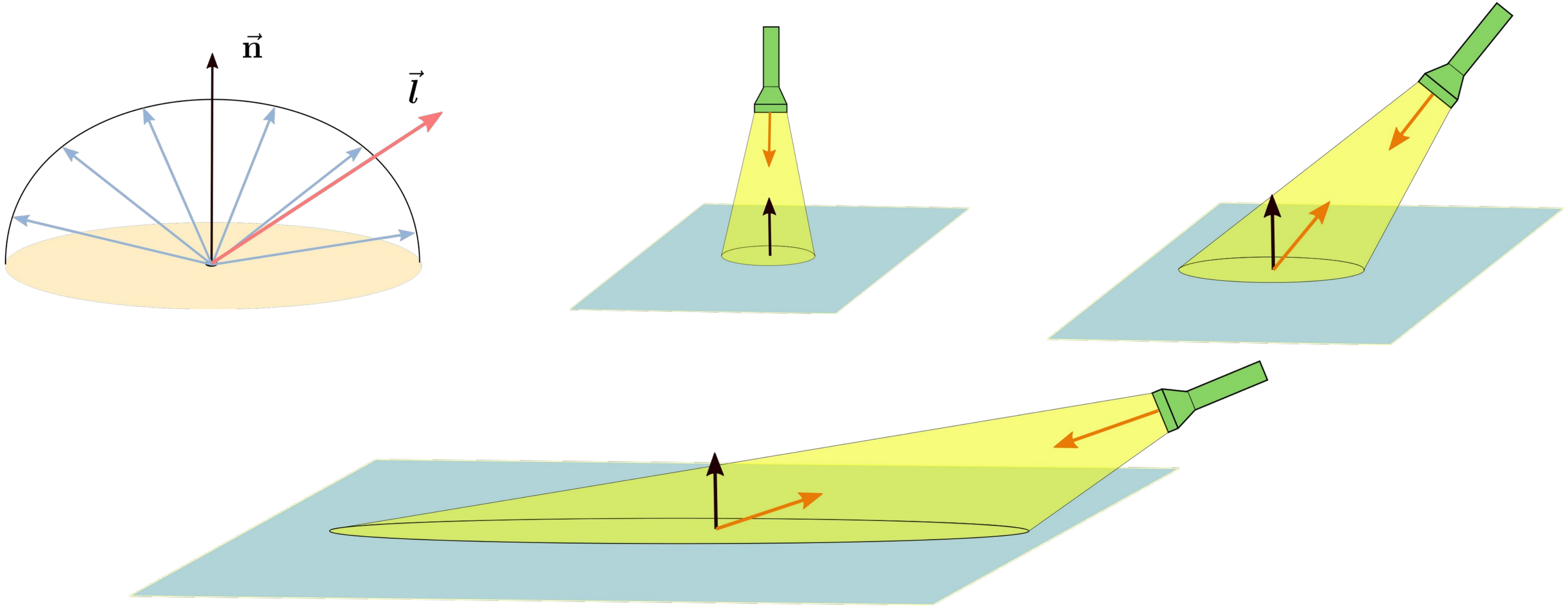
Phong model

diffuse

specular

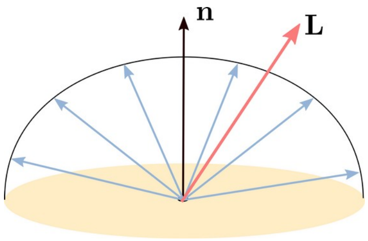


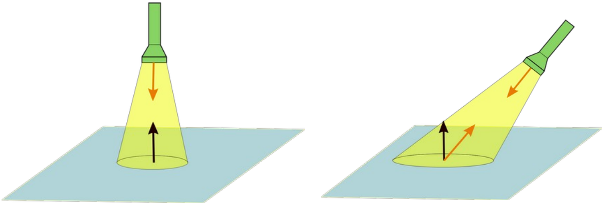
Phong model - diffuse



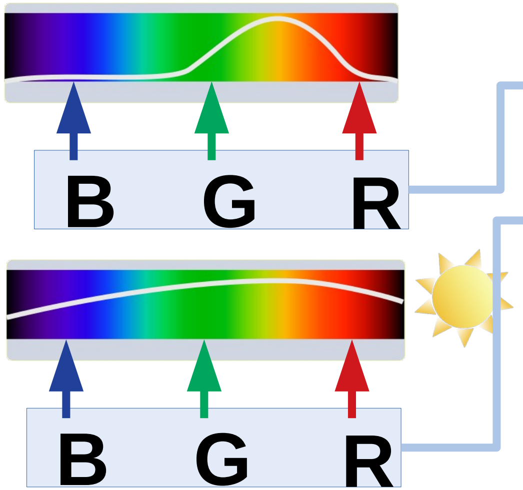
intensity per area decreases with angle between normal and light direction

Phong model - diffuse





dim factor $= \cos\theta = \vec{\mathbf{n}} \cdot \vec{\mathbf{l}}$
($\mathbf{n}$ and $\mathbf{l}$ are normalized)



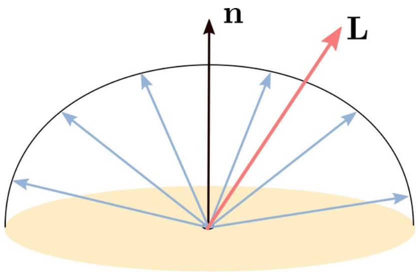
diffuse coefficient
$\mathbf{k}_d = [\mathsf{R}_d, \mathsf{G}_d, \mathsf{B}_d]$

light spectrum
$\mathrm{I} = [\mathsf{R}_L, \mathsf{G}_L, \mathsf{B}_L]$

reflected spectrum
$\mathbf{k}_d\mathrm{I} = [\mathsf{R}_L\mathsf{R}_d, \mathsf{G}_L\mathsf{G}_d, \mathsf{B}_L\mathsf{B}_d]$

$$\textbf{diffuse} = \mathbf{k}_d\mathrm{I}(\mathbf{n} \cdot \mathbf{l})$$

is this physically plausible?



$$\mathbf{diffuse} = \mathbf{k}_d \mathrm{I}(\mathbf{n} \cdot \mathbf{l})$$

e.g. consider a white surface and white light

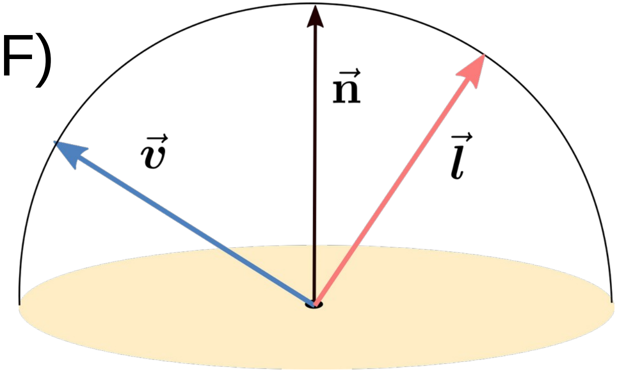
$$\mathbf{I} = \mathbf{k}_d = (1, 1, 1)$$

when $\vec{\mathbf{l}}$ is aligned with $\vec{\mathbf{n}}$, all light is reflected in all directions!

no energy conservation!

closer look

- let's say we know the reflectance function (BRDF)



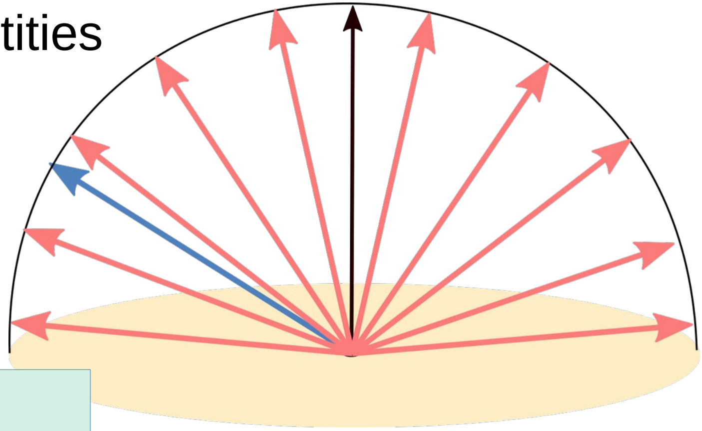
$$f_r(p, \mathbf{v}, \mathbf{l})$$

- what about other incoming directions?
  - integrate over hemisphere (all incoming directions)
  - note: here we assume rays, but really differential quantities

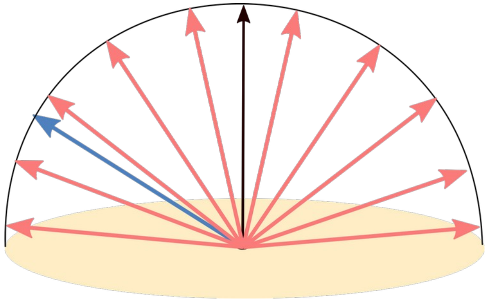
$$L_o(p, \mathbf{v}) = \int_H f_r(p, \mathbf{v}, \mathbf{l}) L_i(p, \mathbf{l}) \cos(\theta) d\mathbf{l}$$

radiance

Rendering Equation



the missing PI



$$\textbf{diffuse} = \mathbf{k}_d \mathrm{I}(\mathbf{n} \cdot \mathbf{l})$$

- how much light is reflected in outgoing direction (considering all incoming directions)

$$L_o(p, \mathbf{v}) = \int_H f_r(p, \mathbf{v}, \mathbf{l}) L_i(p, \mathbf{l}) \cos(\theta) d\mathbf{l}$$

- how much light is arriving at the point?
- consider a white light source for all incoming directions

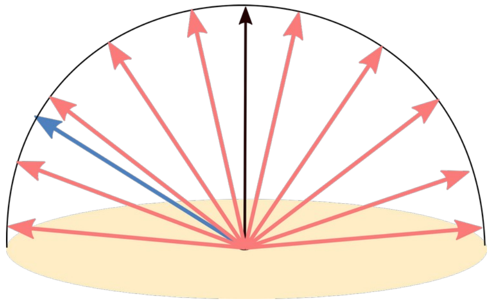
total energy not 1 , it is PI

$$E = \int_H L_i(p, \mathbf{l}) \cos(\theta) d\mathbf{l} = \int_H \cos(\theta) d\mathbf{l} = \pi$$

irradiance

normalized diffuse term:

$$\textbf{diffuse} = \frac{1}{\pi} \mathbf{k}_d \mathrm{I}(\mathbf{n} \cdot \mathbf{l})$$



the missing PI

normalized diffuse term:

$$\textbf{diffuse} = \frac{1}{\pi}\mathbf{k}_d \mathrm{I}(\mathbf{n}\cdot\mathbf{l})$$

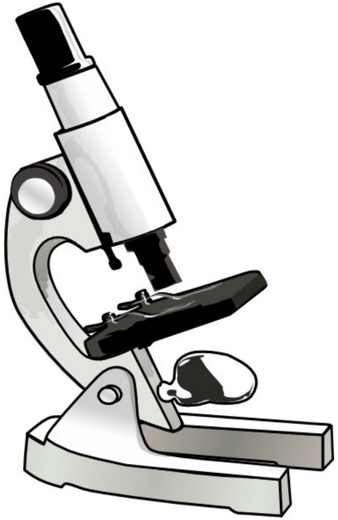
- in practice, for many applications normalization is not important
- sometimes, we can also assume that I=PI, and not I=1

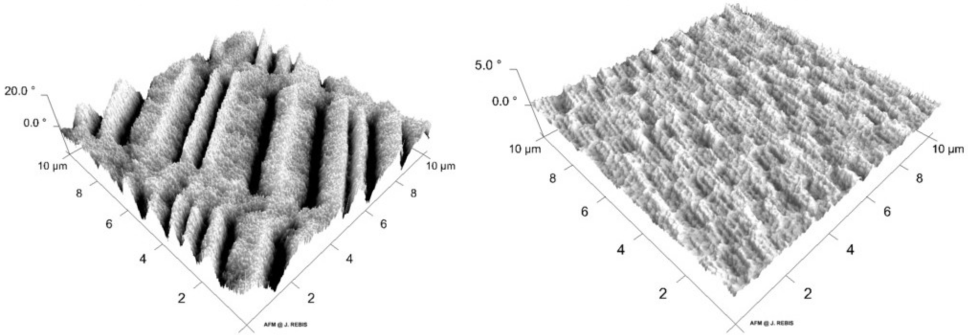
$$L_o(p,\mathbf{v}) = \int_H f_r(p,\mathbf{v},\mathbf{l}) L_i(p,\mathbf{l}) \cos(\theta) d\mathbf{l}$$

- for more realistic renderings, normalization is important
  - lack can result in images that are too bright

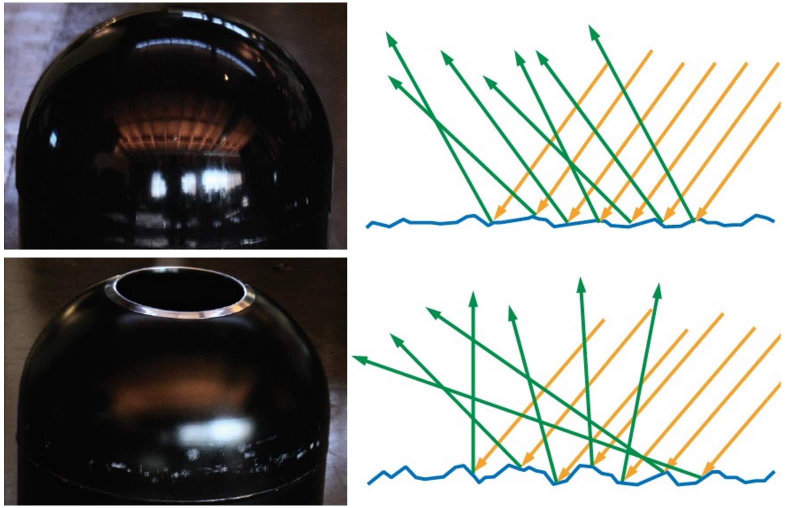
- there is also a normalization factor for the Phong specular part

time do get closer ...

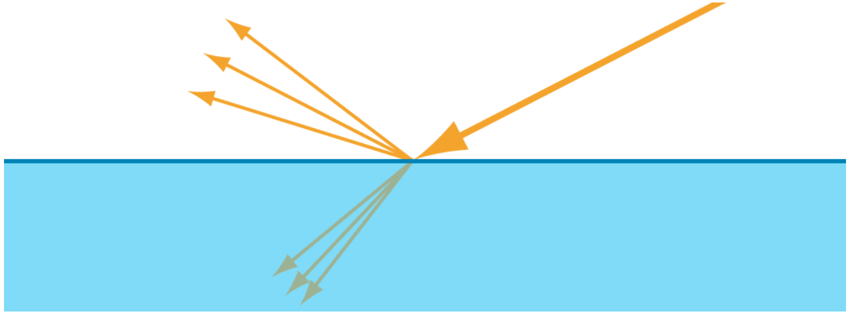




scattering - geometry



surface roughness at
micro-geometry scale



statistically, we model as spread
both for reflection and refraction

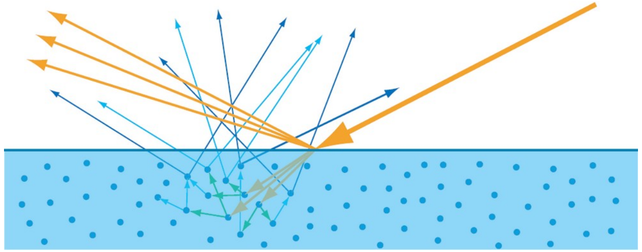
dielectric (non-metals)

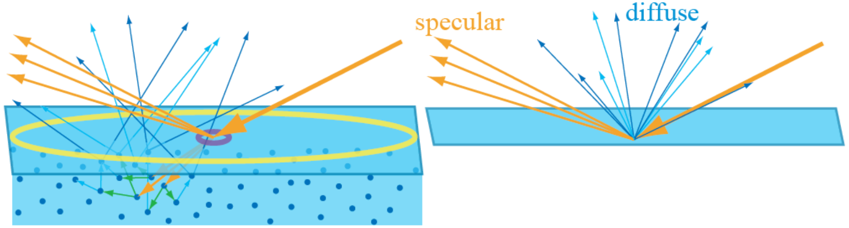
conductor (metals)

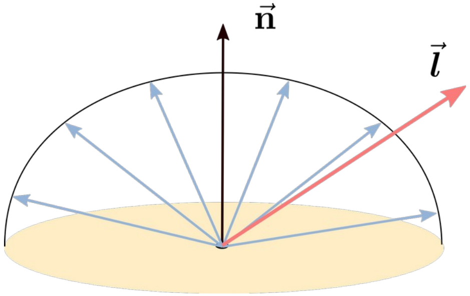
https://www.chaosgroup.com/blog/understanding-metalness

scattering - material

- **metals** (conductor): reflection + absorption
- **non-metals** (dielectric): reflection + refraction + absorption





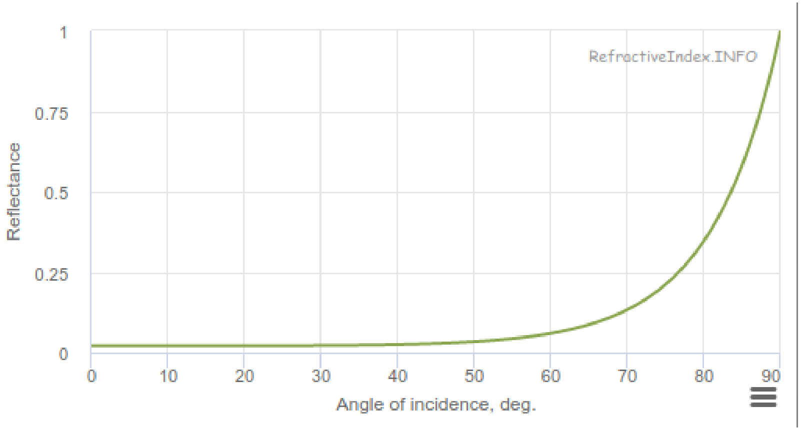


the diffuse/specular partition is physically motivated!

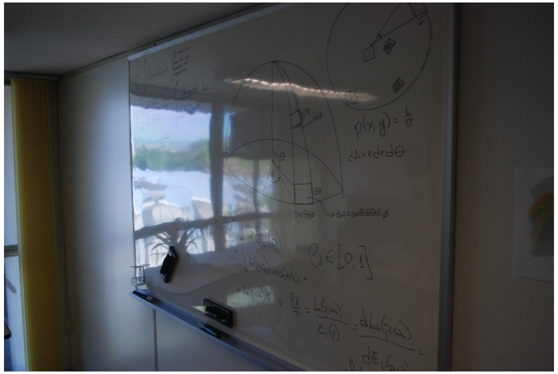
but … how much is reflected and how much is refracted?

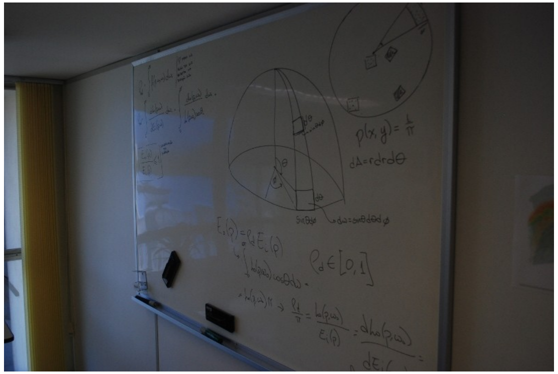
Fresnel

https://shanesimmsart.wordpress.com/2018/03/29/fresnel-reflection/



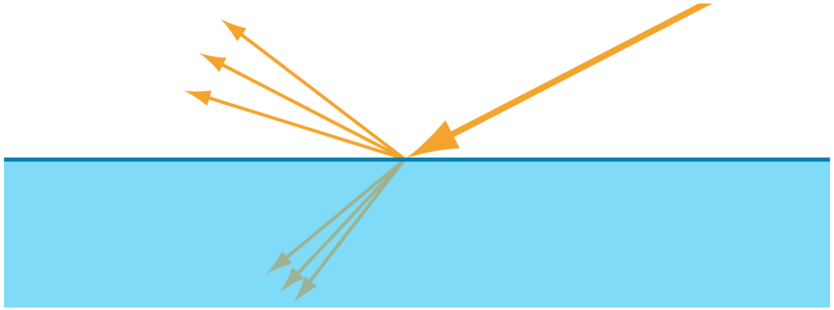
https://refractiveindex.info/?shelf=3d&book=liquids&page=water





but we will not cover polarization here!

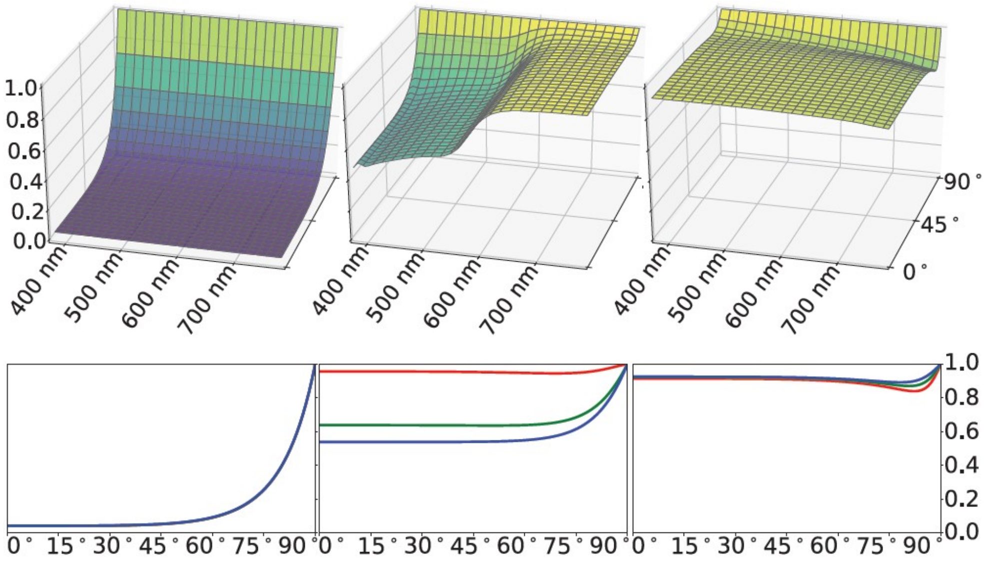
Fresnel



ratio of reflected/refracted light

note: for metals only absorption

glass, copper, aluminium

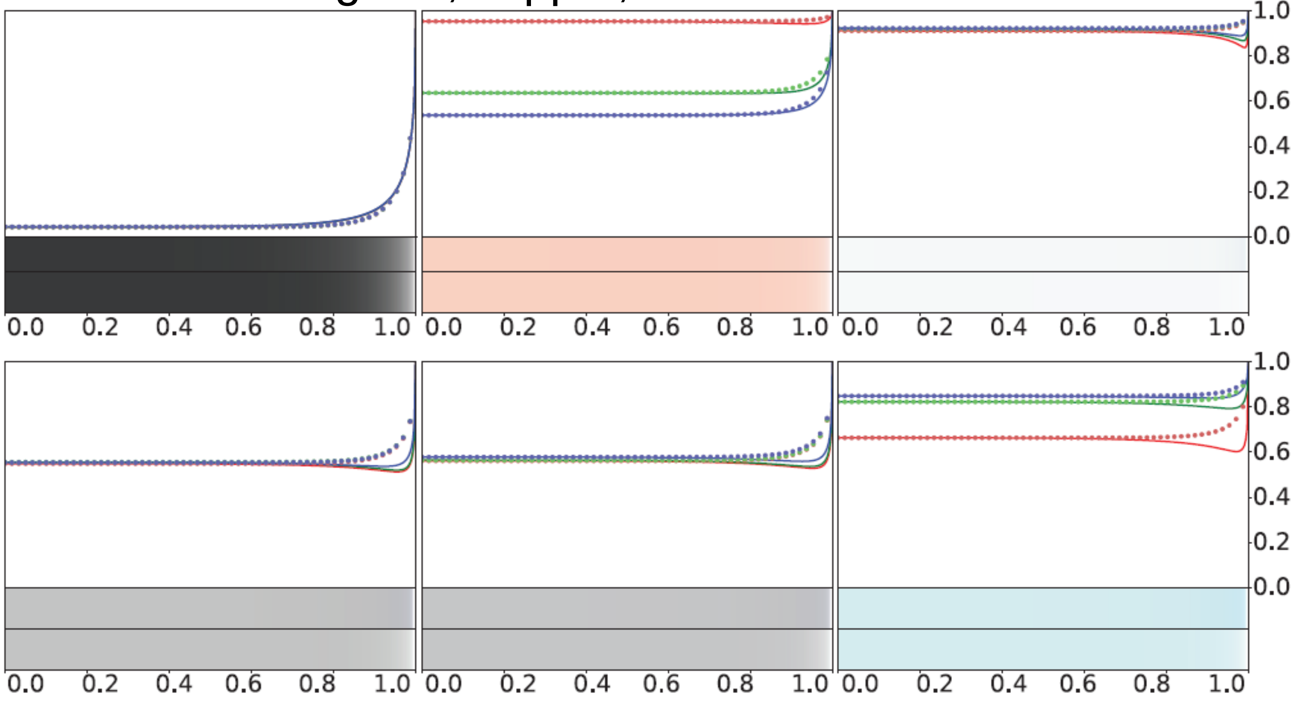


1. for dielectric, no wavelength dependency
2. at grazing angle, 100% reflection

Fresnel

Schlick approximation: $F(\mathbf{n},\mathbf{l}) \approx F_0 + (1-F_0)\left(1-(\mathbf{n}\cdot\mathbf{l})^+\right)^5$

glass, copper, aluminium



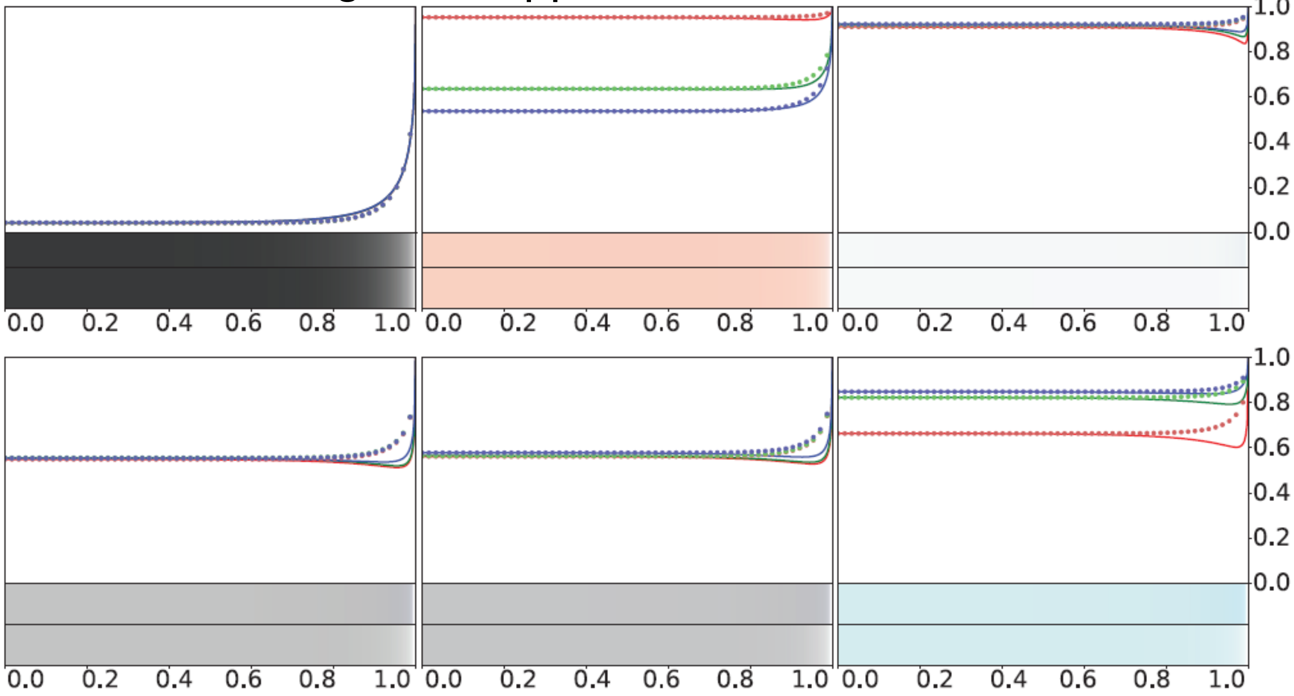
chromium, iron, zinc

original paper: Schlick, An Inexpensive BRDF Model for Physically-based Rendering, 1994

Fresnel

Schlick approximation: $F(\mathbf{n}, \mathbf{l}) \approx F_0 + (1 - F_0)\left(1 - (\mathbf{n} \cdot \mathbf{l})^+\right)^5$

glass, copper, aluminium



chromium, iron, zinc

original paper: Schlick, An Inexpensive BRDF Model for Physically-based Rendering, 1994

Fresnel

Schlick approximation: $F(\mathbf{n}, \mathbf{l}) \approx F_0 + (1 - F_0)\left(1 - (\mathbf{n} \cdot \mathbf{l})^+\right)^5$

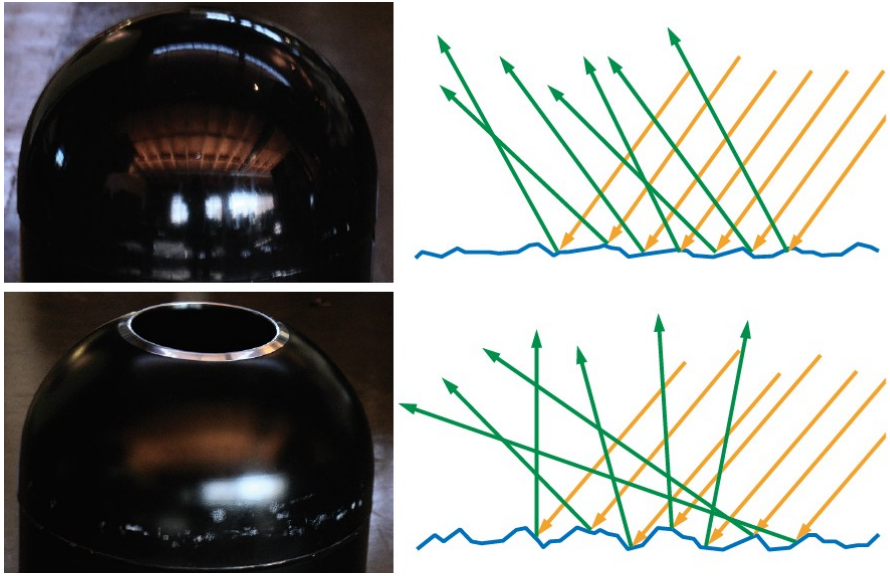
specular colour from index of refraction (IOR): $F_0 = \left(\frac{n-1}{n+1}\right)^2$

Material	Reflectance	IOR
Water	2%	1.33
Fabric	4% to 5.6%	1.5 to 1.62
Common liquids	2% to 4%	1.33 to 1.5
Common gemstones	5% to 16%	1.58 to 2.33
Plastics, glass	4% to 5%	1.5 to 1.58
Other dielectric materials	2% to 5%	1.33 to 1.58
Eyes	2.5%	1.38
Skin	2.8%	1.4
Hair	4.6%	1.55
Teeth	5.8%	1.63
Default value	4%	1.5

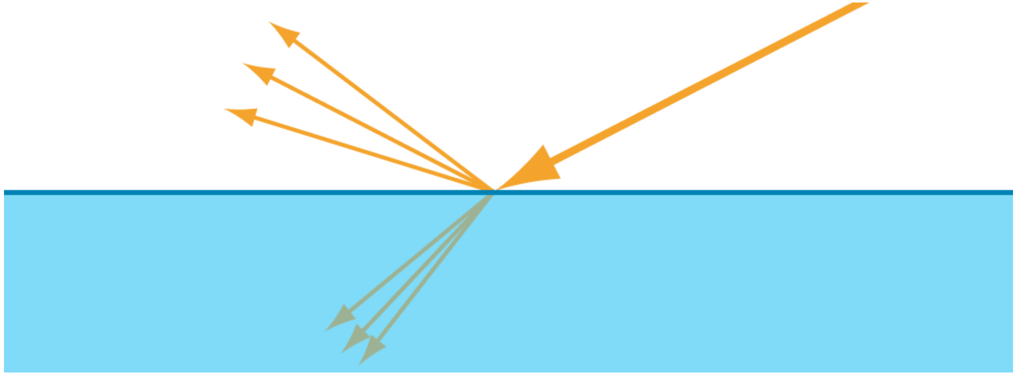
Metal	f_0 in sRGB	Hexadecimal	Color
Silver	0.97, 0.96, 0.91	#f7f4e8	
Aluminum	0.91, 0.92, 0.92	#e8eaea	
Titanium	0.76, 0.73, 0.69	#c1baaf	
Iron	0.77, 0.78, 0.78	#c4c6c6	
Platinum	0.83, 0.81, 0.78	#d3cec6	
Gold	1.00, 0.85, 0.57	#ffd891	
Brass	0.98, 0.90, 0.59	#f9e596	
Copper	0.97, 0.74, 0.62	#f7bc9e	

https://google.github.io/filament/Filament.md.html

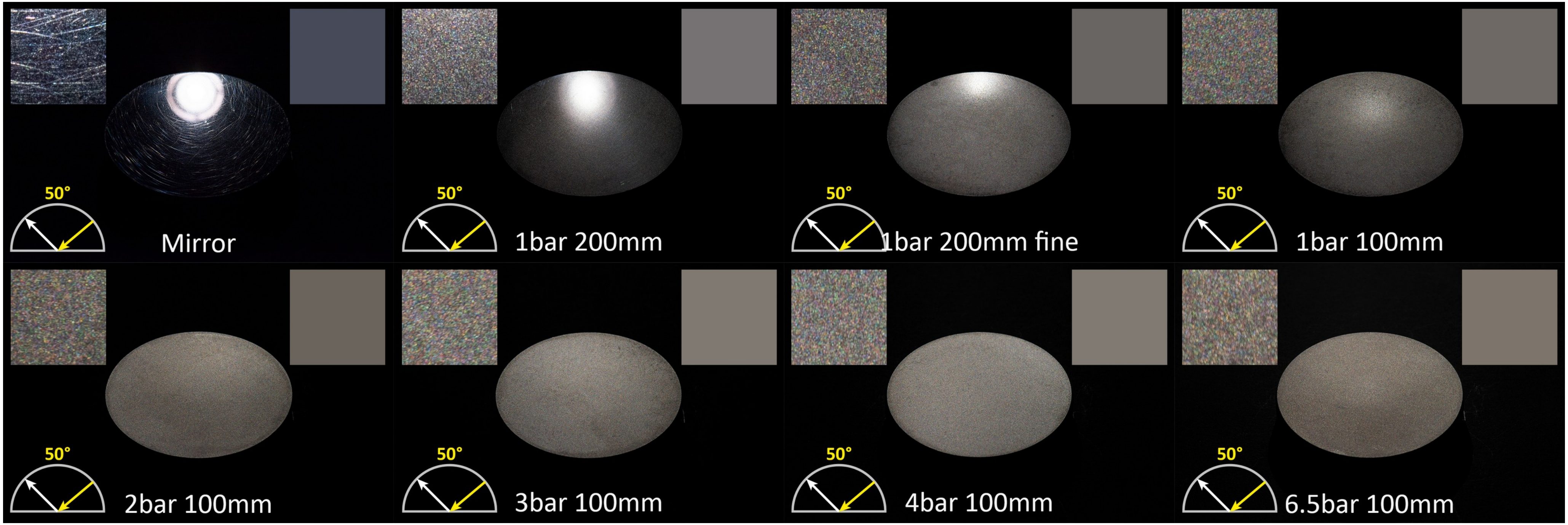
scattering



surface roughness at
micro-geometry scale



statistically, we model as spread
both for reflection and refraction



Investigation and Simulation of Diffraction on Rough Surfaces, Clausen et al.

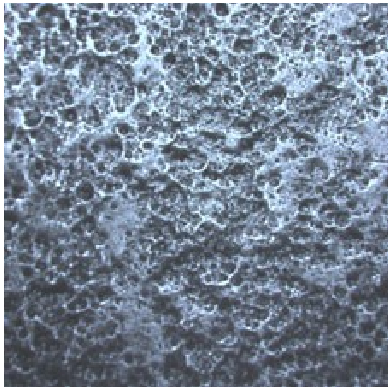
https://onlinelibrary.wiley.com/doi/10.1111/cgf.14717

Microfacets

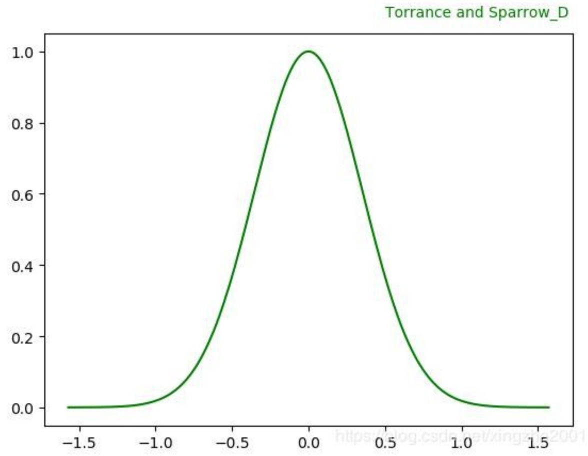
model micro-geometry statistically

macro-surface normal **n**
microfacets normal **m**

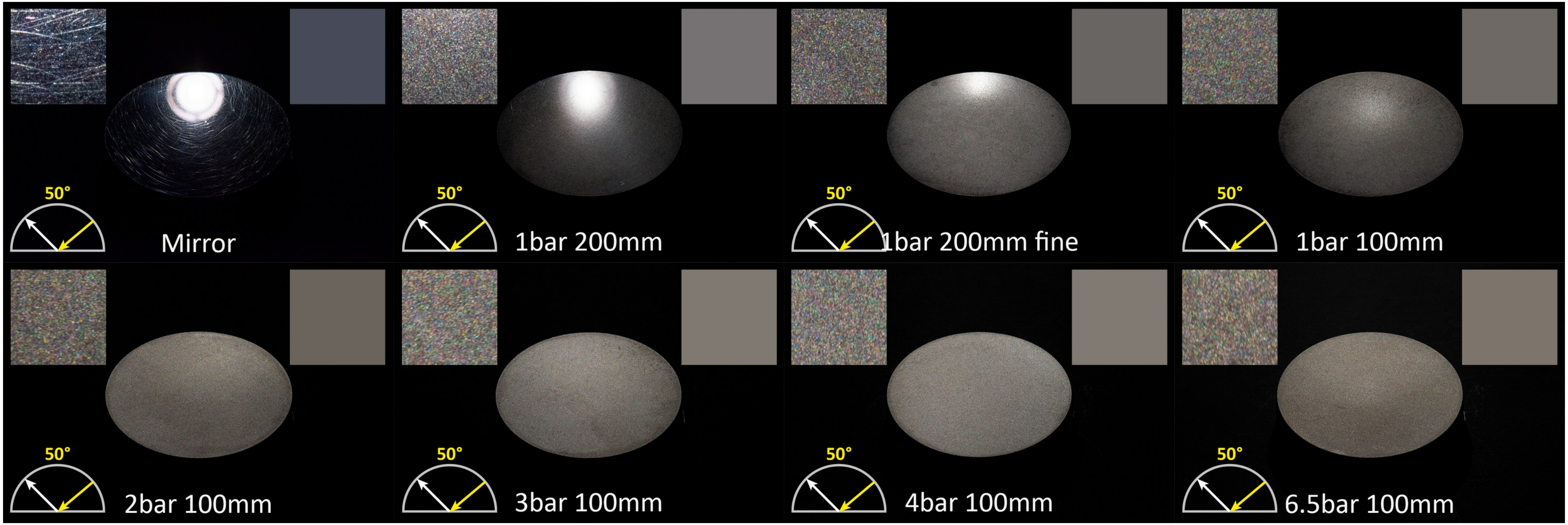
Normal Distribution Function (NDF): $D(\mathbf{m})$

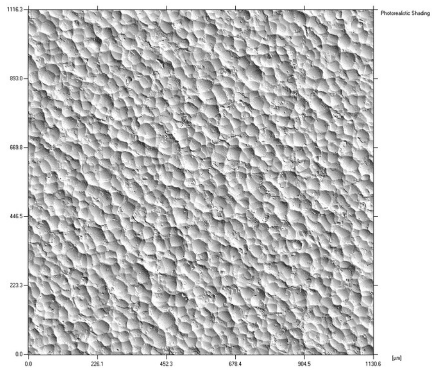


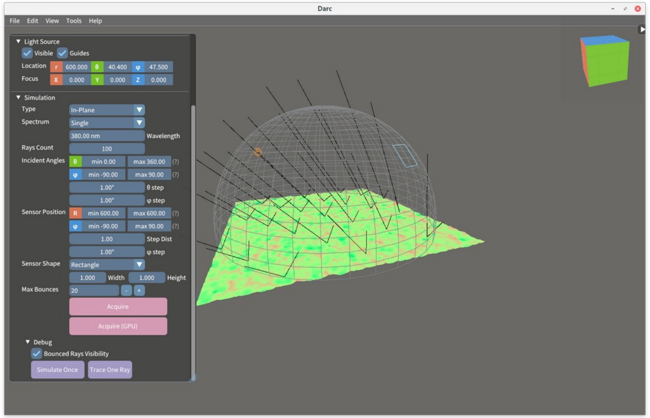
https://www.microscopeworld.com/t-microscope_camera_images.aspx

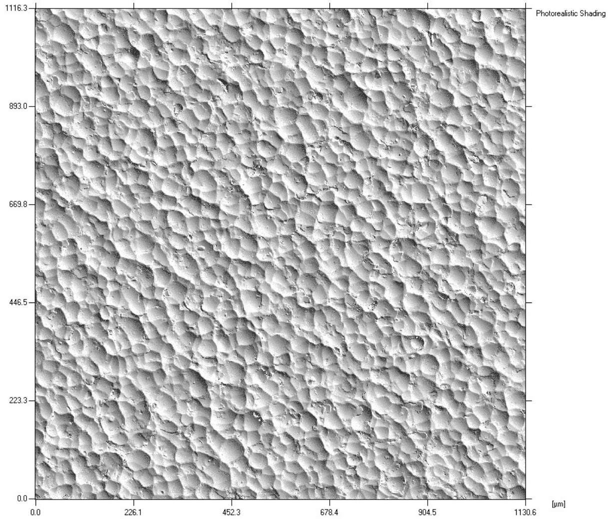


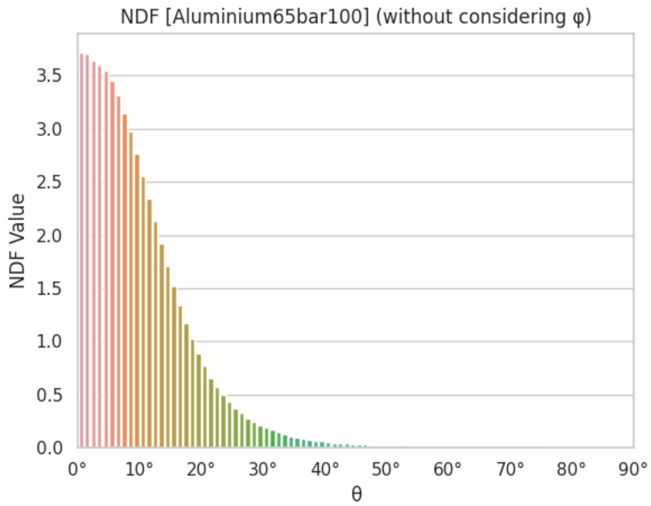
https://www.programmersought.com/article/5773129614/

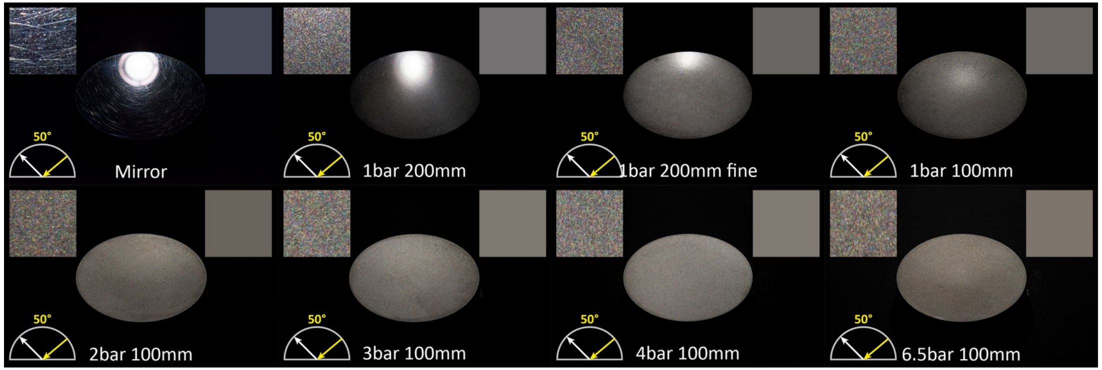

50°
Mirror
50°
1bar 200mm
50°
1bar 200mm fine
50°
1bar 100mm
50°
2bar 100mm
50°
3bar 100mm
50°
4bar 100mm
50°
6.5bar 100mm

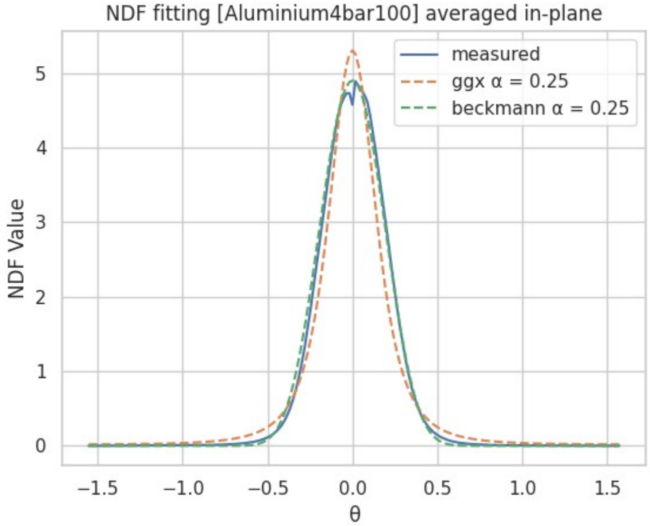

Photorealistic Shading
1116.3
893.0
669.8
446.5
223.3
0.0
0.0
226.1
452.3
678.4
904.5
1130.6
[µm]


Darc
File Edit View Tools Help
Light Source
Simulation
Acquire
Acquire (GPU)
Debug
Simulate Once
Trace One Ray

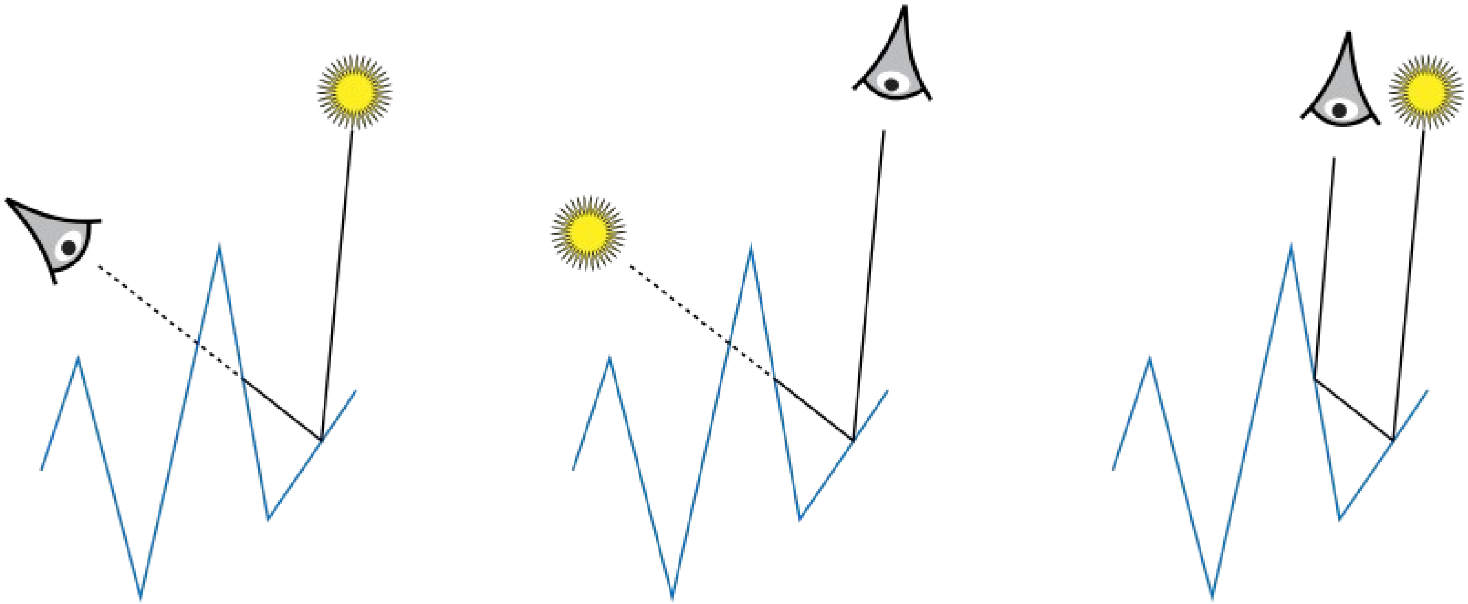

Photorealistic Shading
1116.3
893.0
669.8
446.5
223.3
0.0
0.0
226.1
452.3
678.4
904.5
1130.6
[µm]


NDF [Aluminium65bar100] (without considering φ)
NDF Value
θ
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
0°
10°
20°
30°
40°
50°
60°
70°
80°
90°


50°
Mirror
50°
1bar 200mm
50°
1bar 200mm fine
50°
1bar 100mm
50°
2bar 100mm
50°
3bar 100mm
50°
4bar 100mm
50°
6.5bar 100mm


NDF fitting [Aluminium4bar100] averaged in-plane
measured
ggx α = 0.25
beckmann α = 0.25
NDF Value
θ
0
1
2
3
4
5
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5

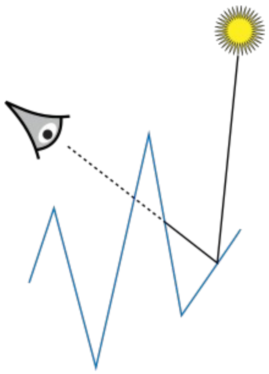
Microfacets



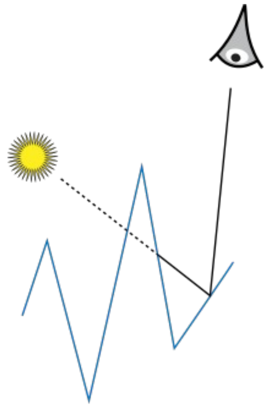
masking shadowing inter-reflection

inter-reflection is quite hard, let’s look only
at masking and shadowing

Microfacets



masking



shadowing

$$G_1(\mathbf{m}, \mathbf{v})$$

masking function:
fraction of microfacets
with normal **m**
visible from direction **v**

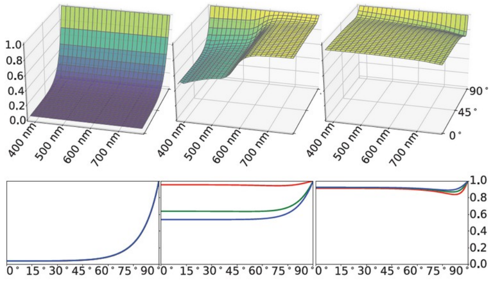
$$G_1(\mathbf{m}, \mathbf{l})$$

shadowing function:
fraction of microfacets
with normal **m**
visible from direction **l**

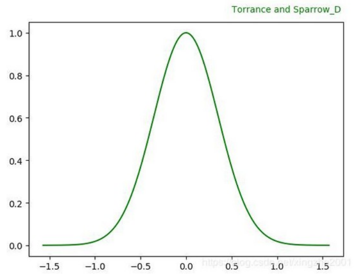
$$G_2(\mathbf{m}, \mathbf{l}, \mathbf{v})$$

masking-shadowing
function

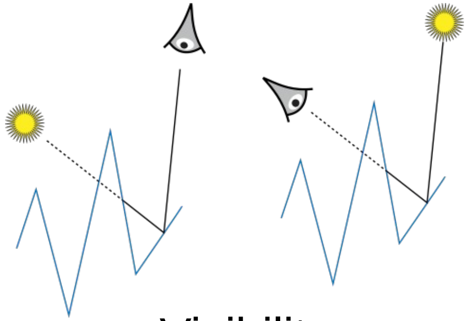
putting everything together



Fresnel



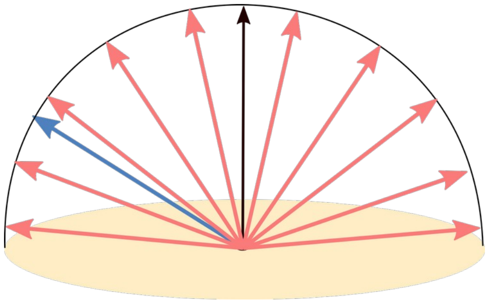
NDF



Visibility

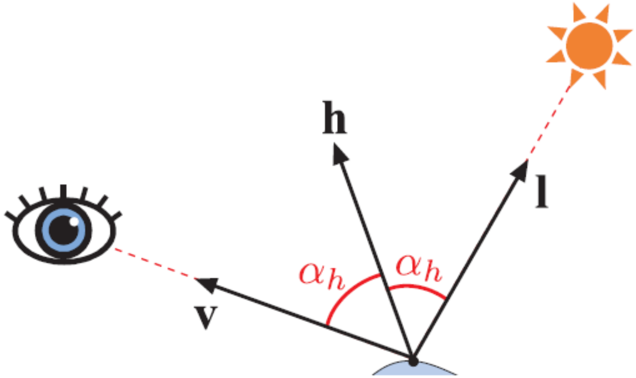
For full computation, needs to integrate the rendering equation over microfacet normals :(

$$L_o(p, \mathbf{v}) = \int_H f_r(p, \mathbf{v}, \mathbf{l}) L_i(p, \mathbf{l}) \cos(\theta) d\mathbf{l}$$



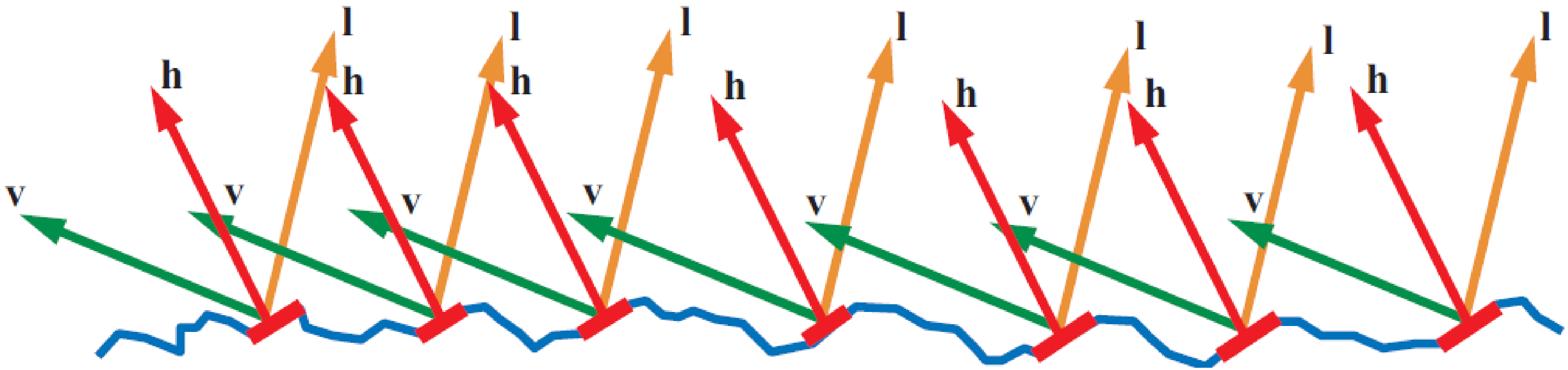
Microfacets - Specular

specular surface reflection: each facet is a Fresnel mirror

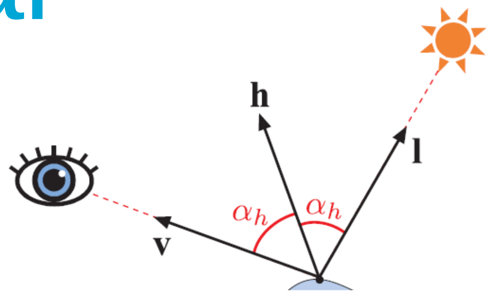


Cook-Torrance

$$f_r(\mathbf{l},\mathbf{v}) = \frac{F(\mathbf{h},\mathbf{l})G_2(\mathbf{l},\mathbf{v},\mathbf{h})D(\mathbf{h})}{4|\mathbf{n}\cdot\mathbf{l}||\mathbf{n}\cdot\mathbf{v}|}$$



Microfacets - Specular



specular surface reflection: each facet is Fresnel mirror

Fresnel

shadowing masking

NDF

$$f_r(\mathbf{l}, \mathbf{v}) = \frac{F(\mathbf{h}, \mathbf{l}) G_2(\mathbf{l}, \mathbf{v}, \mathbf{h}) D(\mathbf{h})}{4|\mathbf{n} \cdot \mathbf{l}||\mathbf{n} \cdot \mathbf{v}|}$$

normalization factor

derivation of the denominator:

- http://www.pbr-book.org/3ed-2018/Light_Transport_I_Surface_Reflection/Sampling_Reflection_Functions.html#sec:microfacet-sample
- Walter et al., Microfacet Models for Refraction through Rough Surfaces

GGX/Smith (Walter et al. 2007)

$$f_r(\mathbf{l}, \mathbf{v}) = \frac{F(\mathbf{h}, \mathbf{l}) G_2(\mathbf{l}, \mathbf{v}, \mathbf{h}) D(\mathbf{h})}{4|\mathbf{n} \cdot \mathbf{l}||\mathbf{n} \cdot \mathbf{v}|}$$

looks very complicated, but only one parameter: roughness α_g

Schlick approximation:

$$F(\mathbf{h}, \mathbf{l}) \approx F_0 + (1 - F_0) \left(1 - (\mathbf{h} \cdot \mathbf{l})^+\right)^5$$

$$\mathbf{G}_2(\mathbf{l}, \mathbf{v}, \mathbf{h}) = \mathbf{G}_1(\mathbf{v}, \mathbf{h}) \mathbf{G}_1(\mathbf{l}, \mathbf{h})$$

Smith Visibility Function

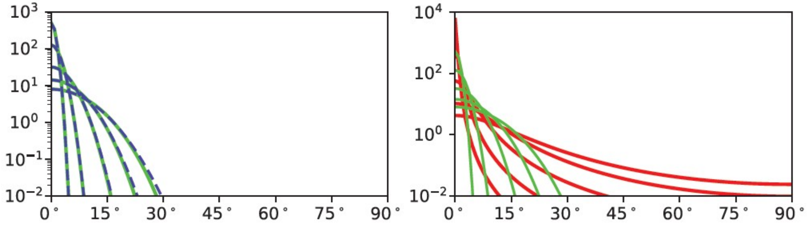
$$\mathbf{G}_1(\mathbf{v}, \mathbf{h}) = \chi^+ \left(\frac{\mathbf{v} \cdot \mathbf{h}}{\mathbf{v} \cdot \mathbf{n}}\right) \frac{2}{1 + \sqrt{1 + \alpha_g^2 \tan^2 \theta_v}}$$

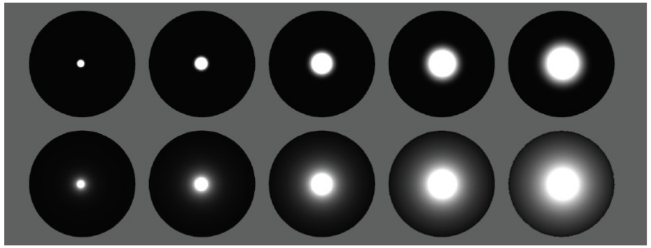
GGX Function:

$$D(\mathbf{h}) = \frac{\alpha_g^2 \chi^+(\mathbf{h} \cdot \mathbf{n})}{\pi \cos^4 \theta_h (\alpha_g^2 + \tan^2 \theta_h)^2}$$

other popular NDF:

- Blinn-Phong (blue)
- Beckmann (green)





Beckmann

GGX

GGX/Smith (Walter et al. 2007)

$$f_r(\mathbf{l},\mathbf{v}) = \frac{F(\mathbf{h},\mathbf{l})G_2(\mathbf{l},\mathbf{v},\mathbf{h})D(\mathbf{h})}{4|\mathbf{n}\cdot\mathbf{l}||\mathbf{n}\cdot\mathbf{v}|}$$

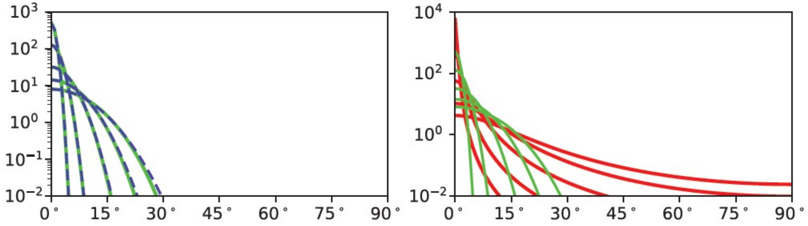
looks very complicated, but only one parameter: roughness α_g

$$F(\mathbf{h},\mathbf{l}) \approx F_0 + (1-F_0)\left(1-(\mathbf{h}\cdot\mathbf{l})^+\right)^5$$

other popular NDF:

- Blinn-Phong (blue)
- Beckmann (green)

$$\mathbf{G}_2(\mathbf{l},\mathbf{v},\mathbf{h}) = \mathbf{G}_1(\mathbf{v},\mathbf{h})\mathbf{G}_1(\mathbf{l},\mathbf{h})$$



Smith Visibility Function:

$$\mathbf{G}_1(\mathbf{v},\mathbf{h}) = \chi^+\left(\frac{\mathbf{v}\cdot\mathbf{h}}{\mathbf{v}\cdot\mathbf{n}}\right)\frac{2}{1+\sqrt{1+\alpha_g^2\tan^2\theta_v}}$$

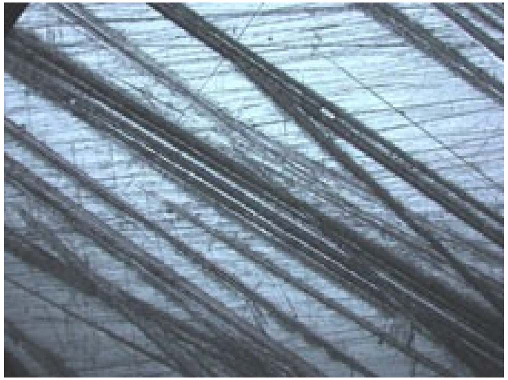
GG

mann

$D($

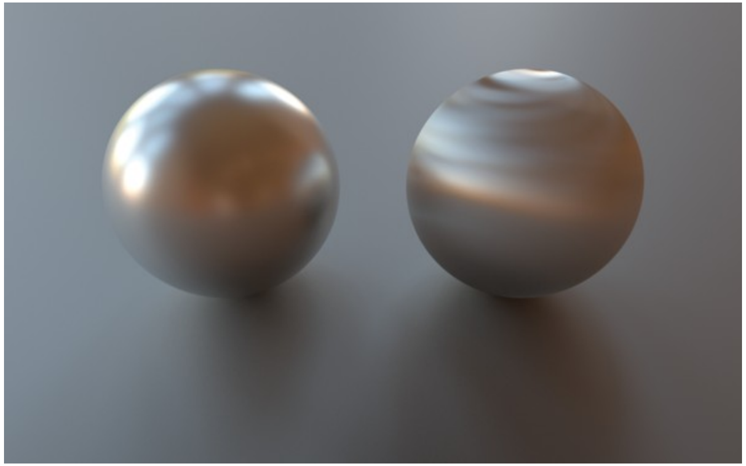
different D’s, G’s and F’s:
https://graphicrants.blogspot.com/2013/08/specular-brdf-reference.html

anisotropic specular reflection



https://www.microscopeworld.com/t-microscope_camera_images.aspx

https://www.blenderguru.com/tutorials/an-introduction-to-anisotropic-shading



http://www.pbr-book.org/3ed-2018/Reflection_Models/Microfacet_Models.html

two directional roughness parameters

Diffuse

there are also Microfacets models for diffuse (local subsurface scattering)

Lambertian

Oren-Nayar

http://www.pbr-book.org/3ed-2018/Reflection_Models/Microfacet_Models.html

Oren-Nayar

https://hapzunglam.wordpress.com/2017/03/13/oren-nayar-reflection/

Oren-Nayar

- also has roughness parameter
- accounts for retro-reflection

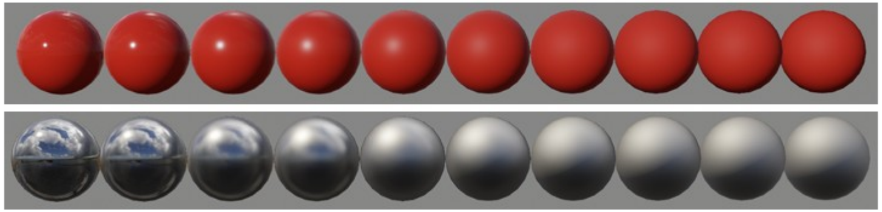
https://hapzunglam.wordpress.com/2017/03/13/oren-nayar-reflection/

Unreal Engine

Material	BaseColor (R, G, B)
Iron	(0.560, 0.570, 0.580)
Silver	(0.972, 0.960, 0.915)
Aluminum	(0.913, 0.921, 0.925)
Gold	(1.000, 0.766, 0.336)
Copper	(0.955, 0.637, 0.538)
Chromium	(0.550, 0.556, 0.554)
Nickel	(0.660, 0.609, 0.526)
Titanium	(0.542, 0.497, 0.449)
Cobalt	(0.662, 0.655, 0.634)
Platinum	(0.672, 0.637, 0.585)

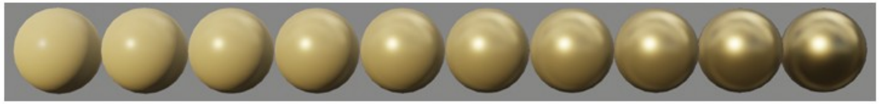
Roughness

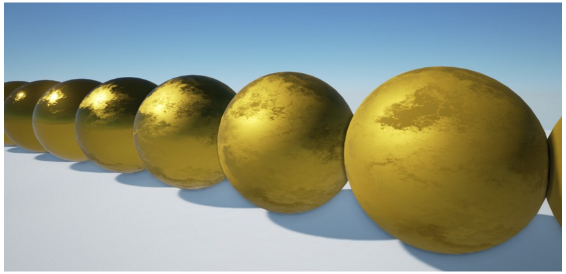
The **Roughness** input controls how rough or smooth a Material's surface is. Rough Materials scatter reflected light in more directions than smooth Materials, which controls how blurry or sharp a reflection is (or how broad or tight a specular highlight is). A Roughness of 0 (smooth) results in a mirror reflection and roughness of 1 (rough) results in a diffuse (or matte) surface.



Metallic

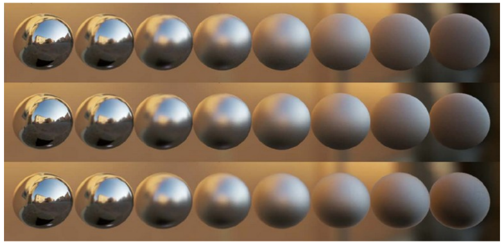
The **Metallic** input controls how 'metal-like' your surface will be. Nonmetals have Metallic values of 0 and metals have Metallic values of 1. For pure surfaces, such as pure metal, pure stone, pure plastic, etc. this value will be 0 or 1, not anything in between. When creating hybrid surfaces like corroded, dusty, or rusty metals, you may find that you need some value *between* 0 and 1.





https://docs.unrealengine.com/en-US/RenderingAndGraphics/Materials/PhysicallyBased/index.html

Unreal Engine



Lambertian Diffuse

$$f(\mathbf{l}, \mathbf{v}) = \frac{\mathbf{c}_{\text{diff}}}{\pi}$$

modified Schlick Fresnel

$$F(\mathbf{v}, \mathbf{h}) = F_0 + (1 - F_0)\, 2^{(-5.55473(\mathbf{v}\cdot\mathbf{h}) - 6.98316)(\mathbf{v}\cdot\mathbf{h})}$$

Cook-Torrance Specular Microfacets

$$f(\mathbf{l}, \mathbf{v}) = \frac{D(\mathbf{h})\, F(\mathbf{v}, \mathbf{h})\, G(\mathbf{l}, \mathbf{v}, \mathbf{h})}{4\,(\mathbf{n}\cdot\mathbf{l})\,(\mathbf{n}\cdot\mathbf{v})}$$

Smith Shadowing/Masking

$$k = \frac{(Roughness + 1)^2}{8}$$

GGX NDF

$$D(\mathbf{h}) = \frac{\alpha^2}{\pi\left((\mathbf{n}\cdot\mathbf{h})^2\,(\alpha^2 - 1) + 1\right)^2}$$

$$G_1(\mathbf{v}) = \frac{\mathbf{n}\cdot\mathbf{v}}{(\mathbf{n}\cdot\mathbf{v})(1-k) + k}$$

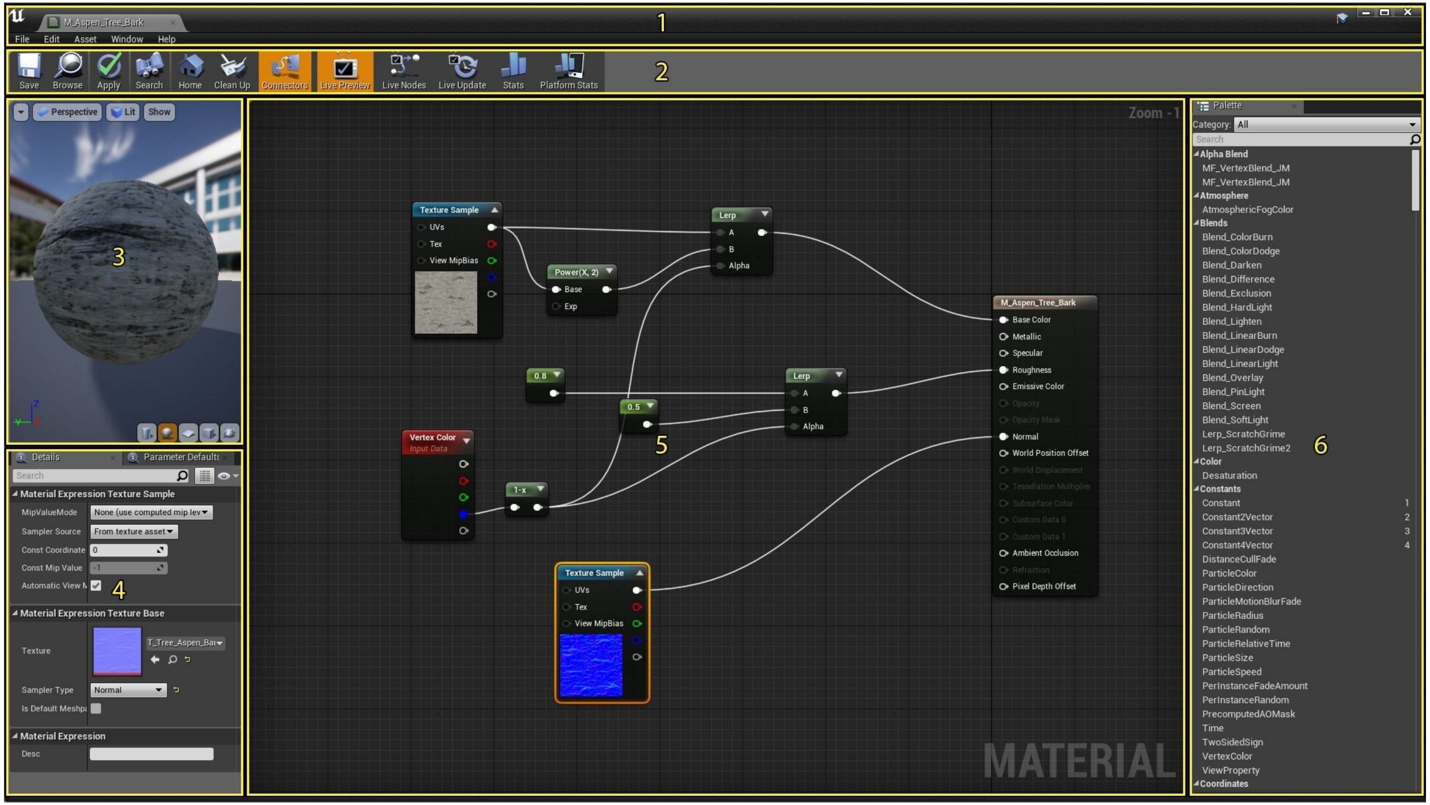
$$G(\mathbf{l}, \mathbf{v}, \mathbf{h}) = G_1(\mathbf{l})\, G_1(\mathbf{v})$$

https://de45xmedrsdbp.cloudfront.net/Resources/files/2013SiggraphPresentationsNotes-26915738.pdf

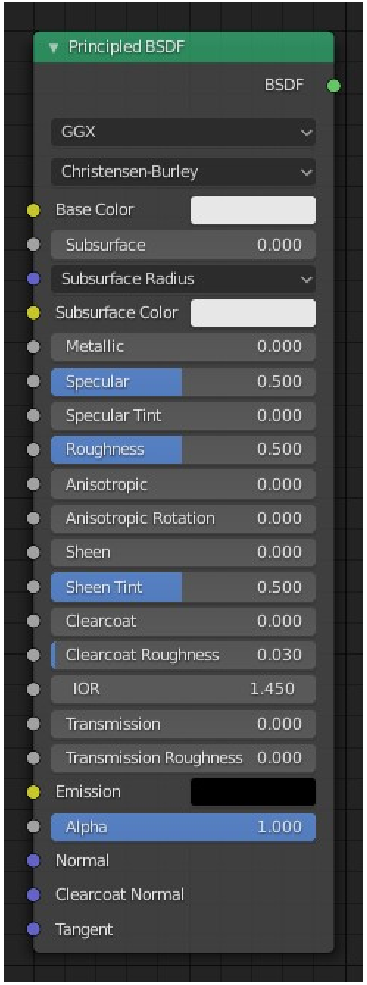



material (shader) editor

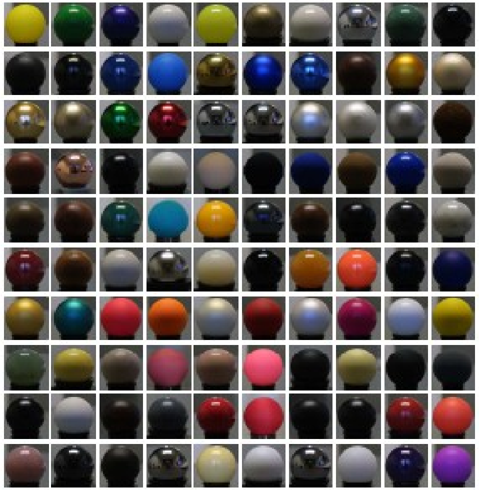
Blender

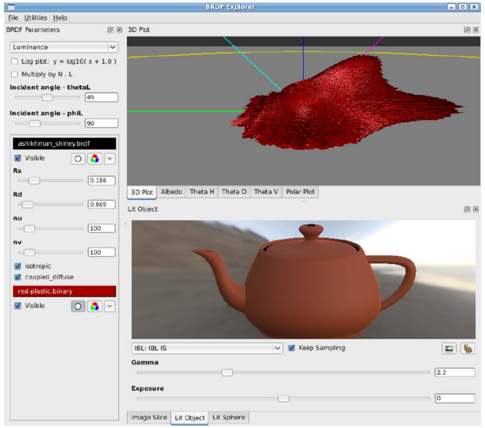


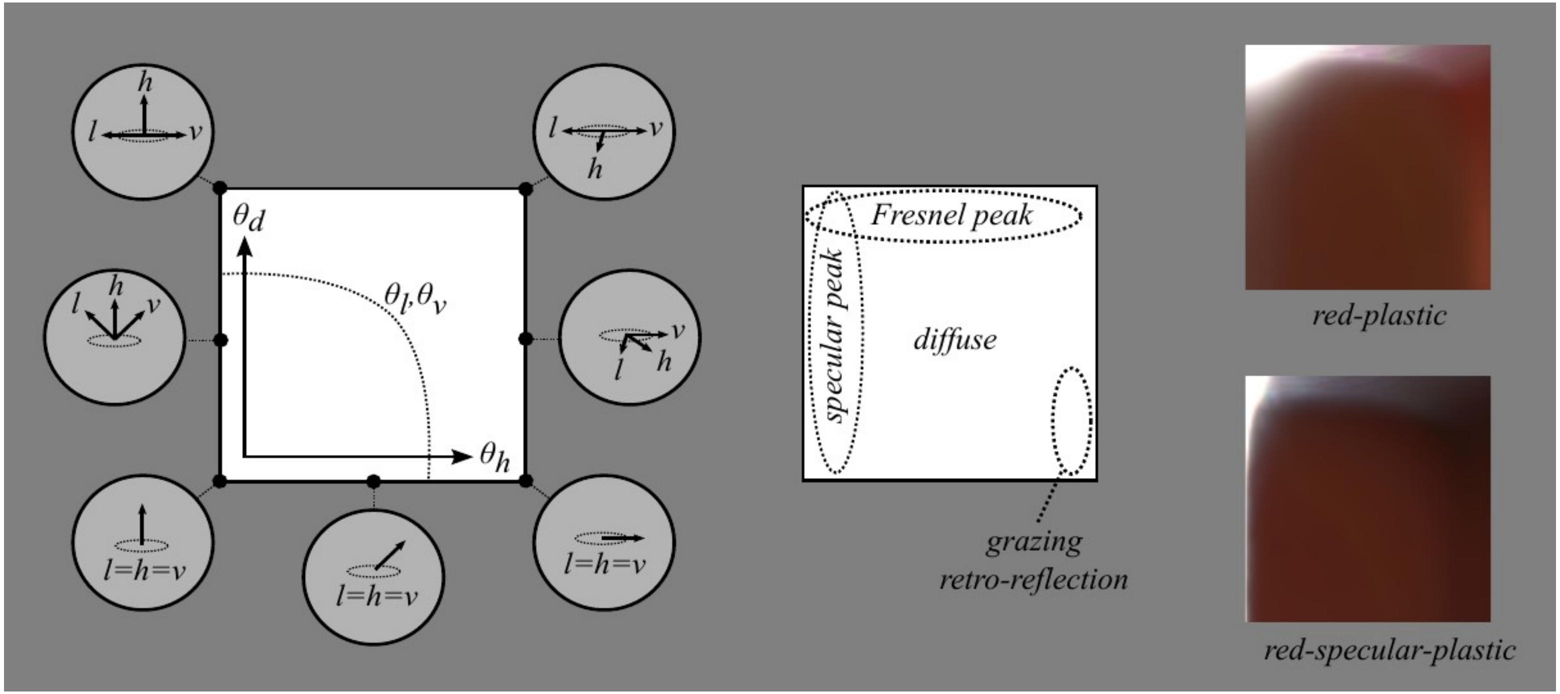
UE4



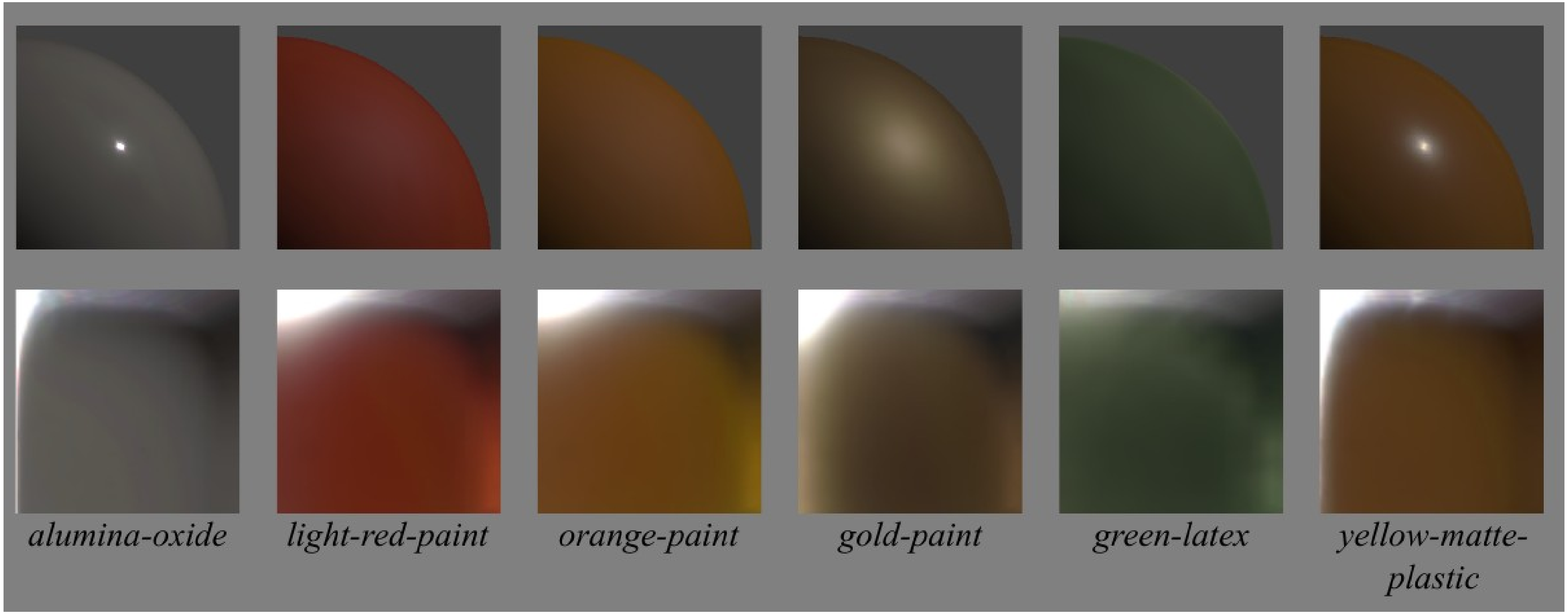
Principled BRDFs

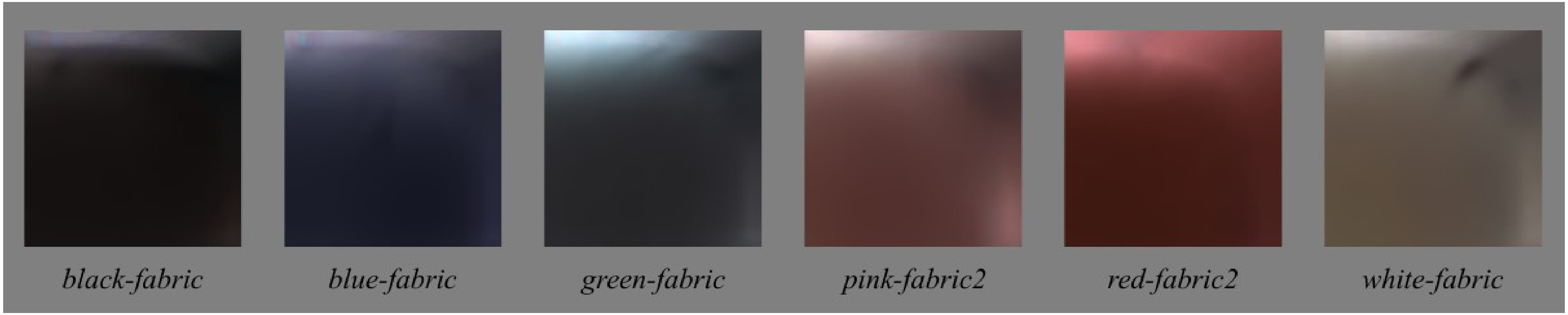




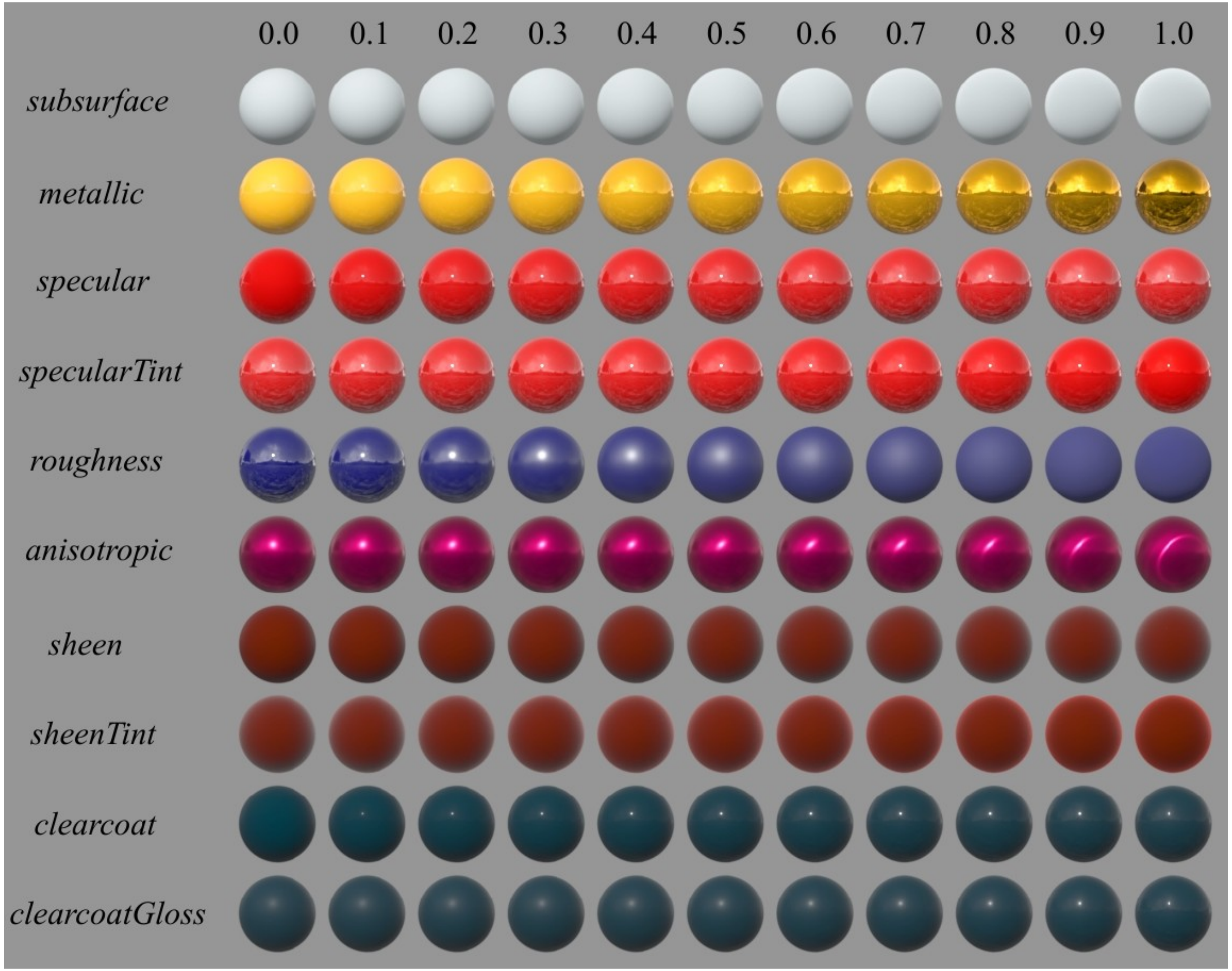


Burley et al, Physically-Based Shading at Disney



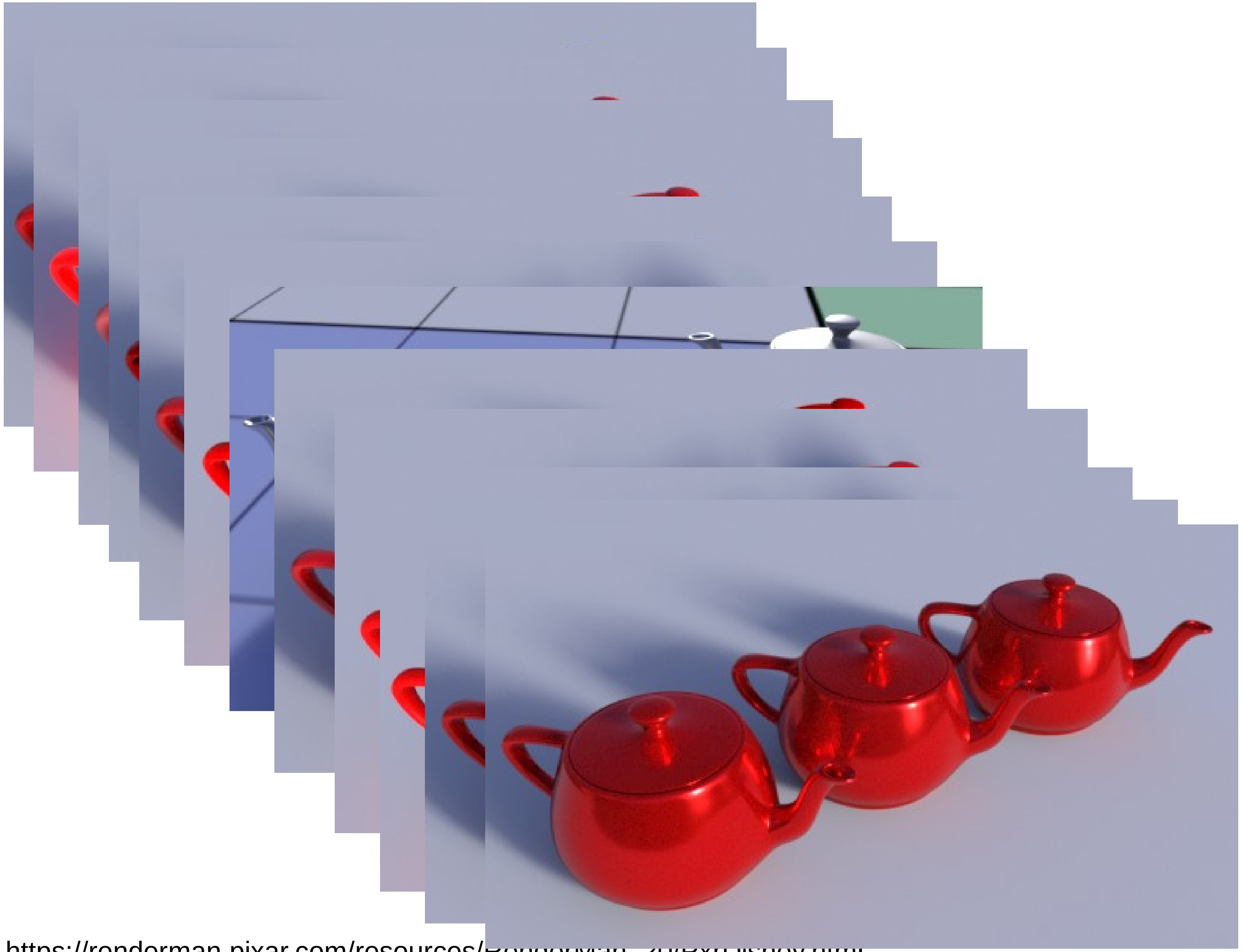


Burley et al, Physically-Based Shading at Disney



Burley et al, Physically-Based Shading at Disney

RENDERMAN



base color (RGB or texture)

emit color

subsurface

metallic

specular (IOR)

specular tint

roughness

anisotropic

sheen

sheen tint

clearcoat

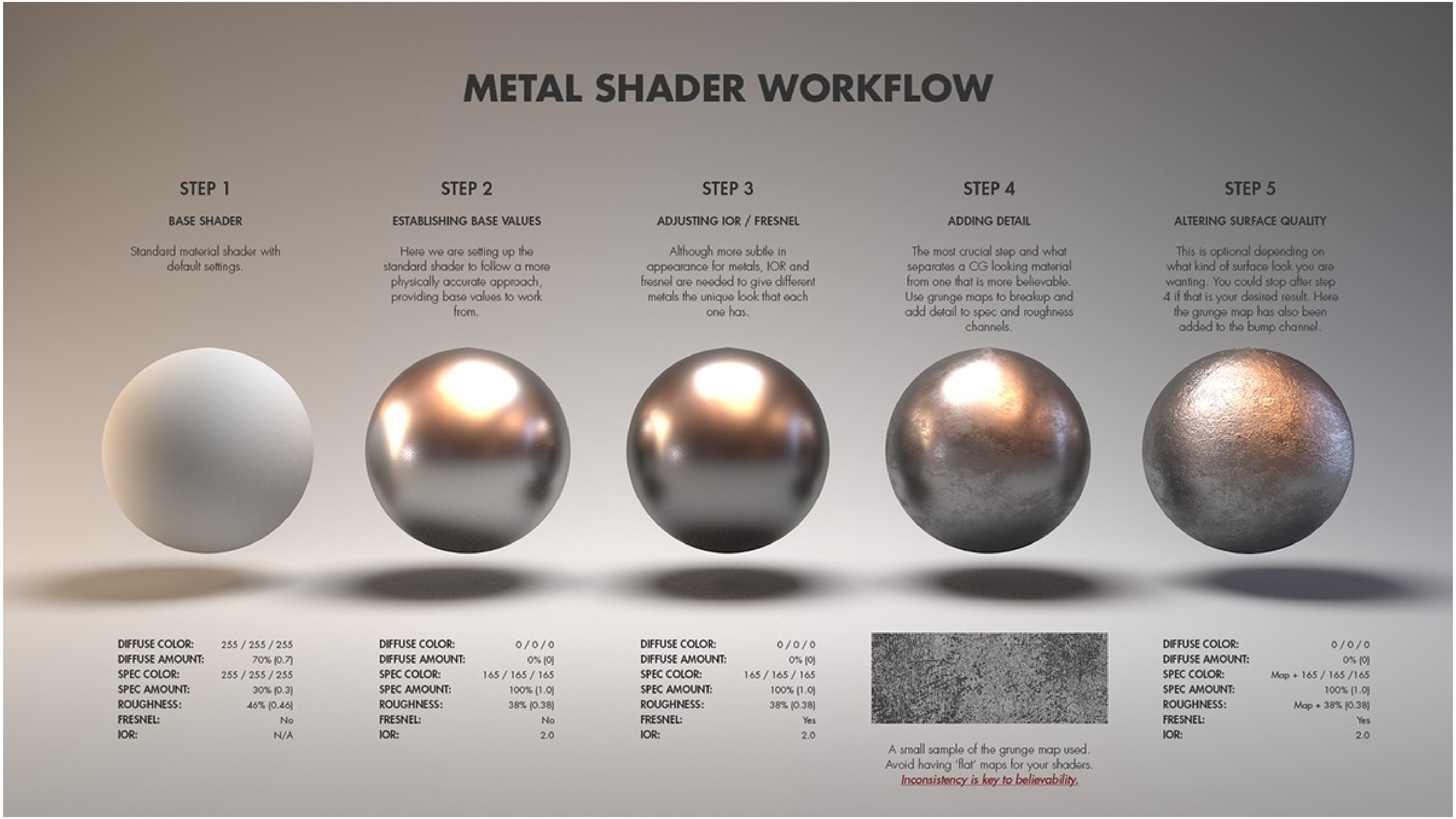
clearcoat gloss

https://renderman.pixar.com/resources/RenderMan_20/PxrDisney.html

why is it so important?

Ray Tracing for the Movie ‘Cars’, Christensen et al.

Physically Based Lighting at Pixar, Hery and Villemin



https://www.behance.net/gallery/35636521/Material-Studies-Metals

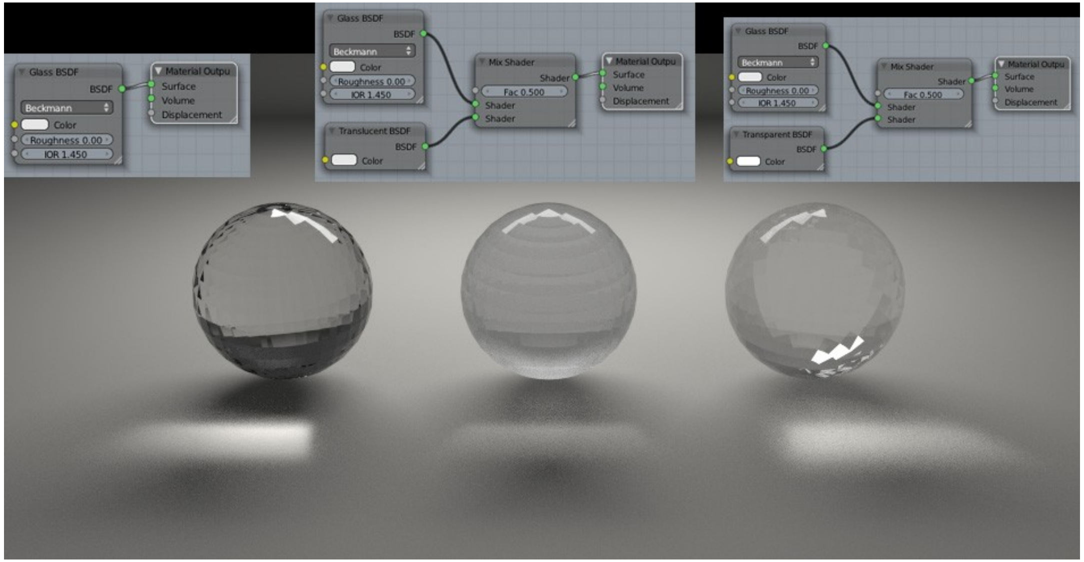

COPPER
MATERIAL STUDY: STUDIO
BRUSHED
DIFFUSE COLOR: 0 / 0 / 0
DIFFUSE AMOUNT: 0% (0)
SPEC COLOR: 174 / 124 / 92
SPEC AMOUNT: 100% (1.0)
ROUGHNESS: 143 / 143 / 143
FRESNEL: Yes
IOR: 1.1
DENTED
DIFFUSE COLOR: 0 / 0 / 0
DIFFUSE AMOUNT: 0% (0)
SPEC COLOR: 174 / 124 / 92
SPEC AMOUNT: 100% (1.0)
ROUGHNESS: 114 / 114 / 114
FRESNEL: Yes
IOR: 1.1
OLD
DIFFUSE COLOR: 0 / 0 / 0
DIFFUSE AMOUNT: 0% (0)
SPEC COLOR: 212 / 174 / 151
SPEC AMOUNT: 100% (1.0)
ROUGHNESS: 82 / 82 / 82
FRESNEL: Yes
IOR: 1.1
SANDBLASTED
DIFFUSE COLOR: 0 / 0 / 0
DIFFUSE AMOUNT: 0% (0)
SPEC COLOR: 174 / 124 / 92
SPEC AMOUNT: 100% (1.0)
ROUGHNESS: 114 / 114 / 114
FRESNEL: Yes
IOR: 1.1
WROUGHT
DIFFUSE COLOR: 0 / 0 / 0
DIFFUSE AMOUNT: 0% (0)
SPEC COLOR: 174 / 124 / 92
SPEC AMOUNT: 100% (1.0)
ROUGHNESS: 143 / 143 / 143
FRESNEL: Yes
IOR: 1.1


GOLD
MATERIAL STUDY: STUDIO
GOLD
DIFFUSE COLOR: 0 / 0 / 0
DIFFUSE AMOUNT: 0% (0)
SPEC COLOR: 255 / 213 / 128
SPEC AMOUNT: 100% (1.0)
ROUGHNESS: 35 / 35 / 35
FRESNEL: Yes
IOR: 0.47
GOLD PAINT
DIFFUSE COLOR: 0 / 0 / 0
DIFFUSE AMOUNT: 0% (0)
SPEC COLOR: 226 / 193 / 112
SPEC AMOUNT: 100% (1.0)
ROUGHNESS: 57 / 57 / 57
FRESNEL: Yes
IOR: 0.47
GOLD PAINT CRACKED
DIFFUSE COLOR: 0 / 0 / 0
DIFFUSE AMOUNT: 0% (0)
SPEC COLOR: 165 / 165 / 165
SPEC AMOUNT: 100% (1.0)
ROUGHNESS: 70 / 70 / 70
FRESNEL: Yes
IOR: 0.47
GOLD ROUGH
DIFFUSE COLOR: 0 / 0 / 0
DIFFUSE AMOUNT: 0% (0)
SPEC COLOR: 146 / 124 / 72
SPEC AMOUNT: 100% (1.0)
ROUGHNESS: 100 / 100 / 100
FRESNEL: Yes
IOR: 0.47
GOLD FOIL ROUGH
DIFFUSE COLOR: 0 / 0 / 0
DIFFUSE AMOUNT: 0% (0)
SPEC COLOR: 255 / 213 / 128
SPEC AMOUNT: 100% (1.0)
ROUGHNESS: 65 / 65 / 65
FRESNEL: Yes
IOR: 0.47

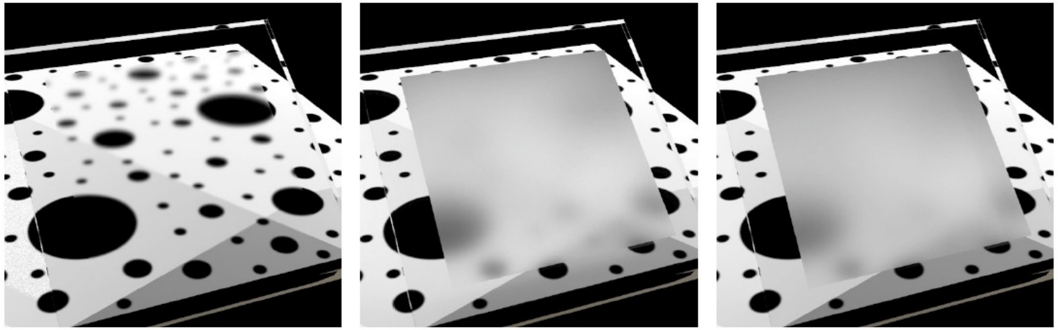
BSDF = BRDF + BTDF

reflection transmission

https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/hair.html



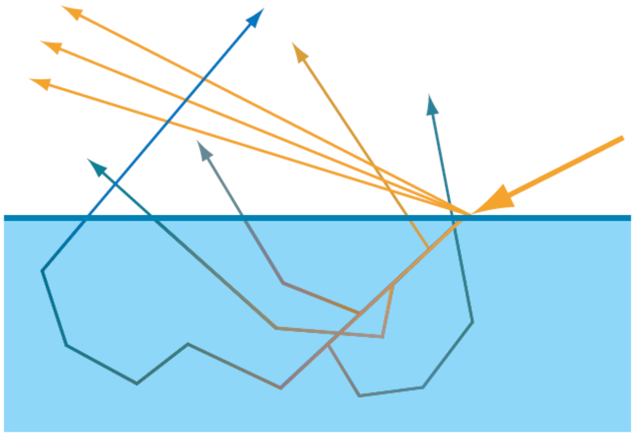
https://blenderartists.org/t/cycles-glass-transparent-shadows/541771/3



anti-glare (Beckman, $\alpha_b = 0.023$) ground (GGX, $\alpha_g = 0.394$) etched (GGX, $\alpha_g = 0.553$)

Walter et al., Microfacet models for refraction through rough surfaces

BSSDF: Bidirectional Surface-Scattering Distribution Function

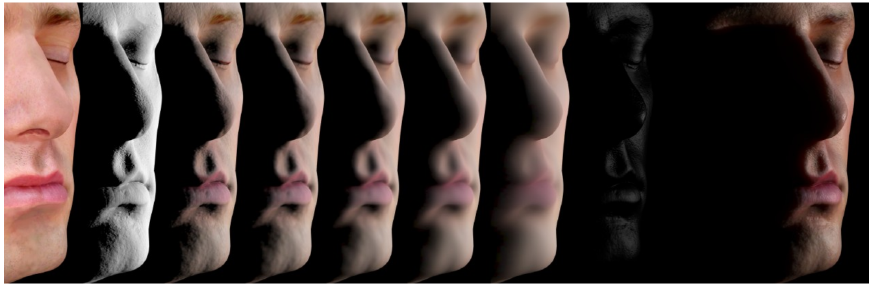


$$f_r(p_o, p_i, \mathbf{v}, \mathbf{l})$$

Jensen et al., A Practical Model for Subsurface Light Transport

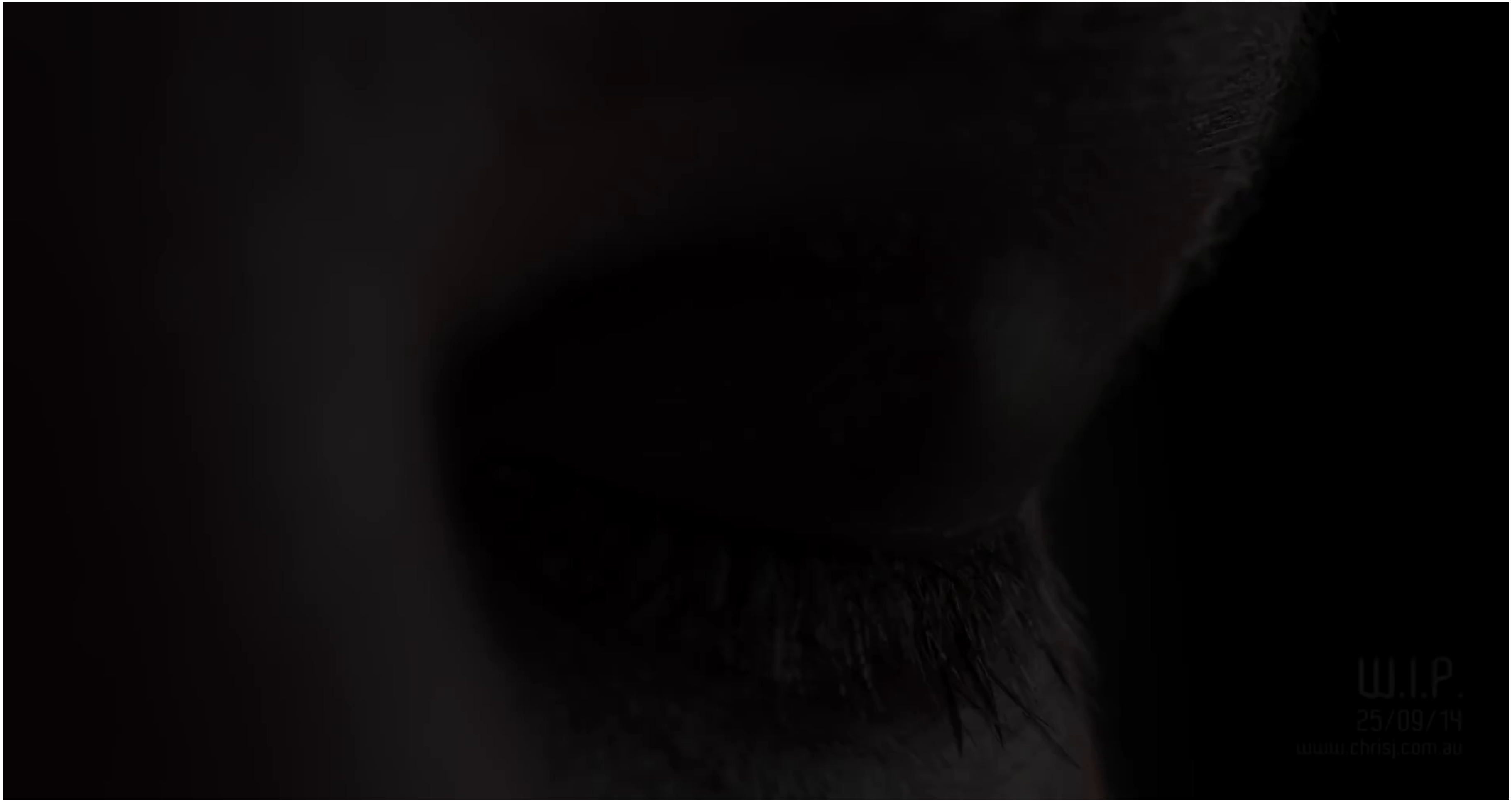
Anyone can Cook: Inside Ratatouille's Kitchen

BSSDF: Bidirection Surface-Scattering Distribution Function



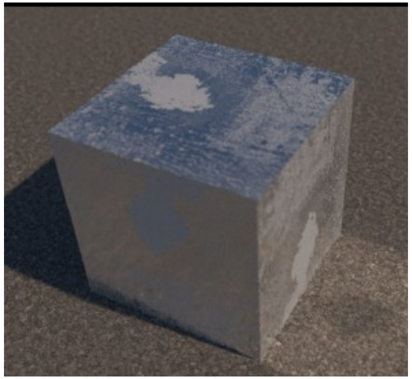
d’Eon et al, Efficient Rendering of Human Skin

https://docs.unrealengine.com/en-US/Resources/Showcases/DigitalHumans/index.html



https://www.youtube.com/watch?v=HjHiC0mt4Ts

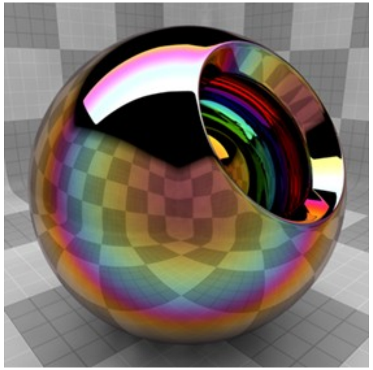
still not done ...

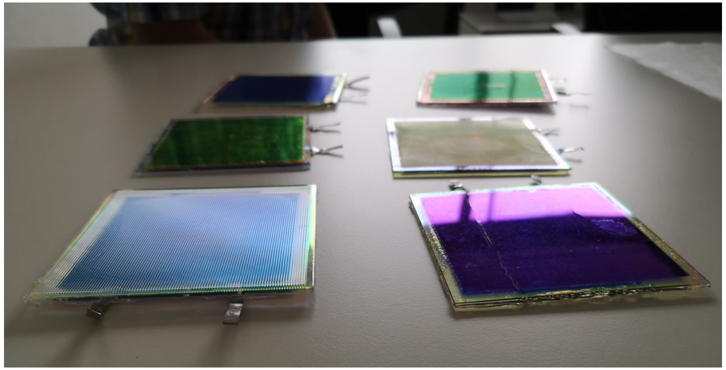


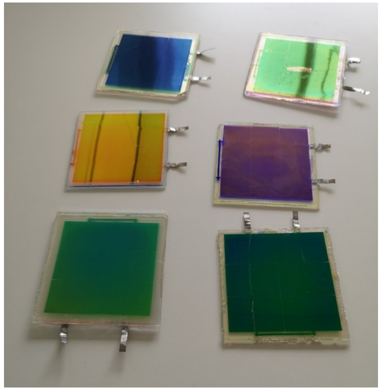
A Comprehensive Framework for Rendering Layered Materials, Jakob et al.

still not done ...

Iridescence





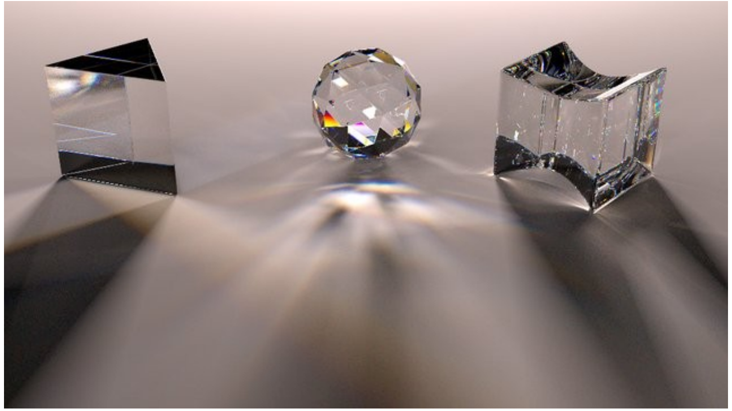


https://learn.foundry.com/modo/content/help/pages/shading_lighting/shader_items/iridescence.html

Fluorescence

https://eclat-digital.com/fluorescence-brighten-your-renderings/

Spectral Rendering



Efficient Spectral Rendering on the GPU for Predictive Rendering, Murray et al.

today

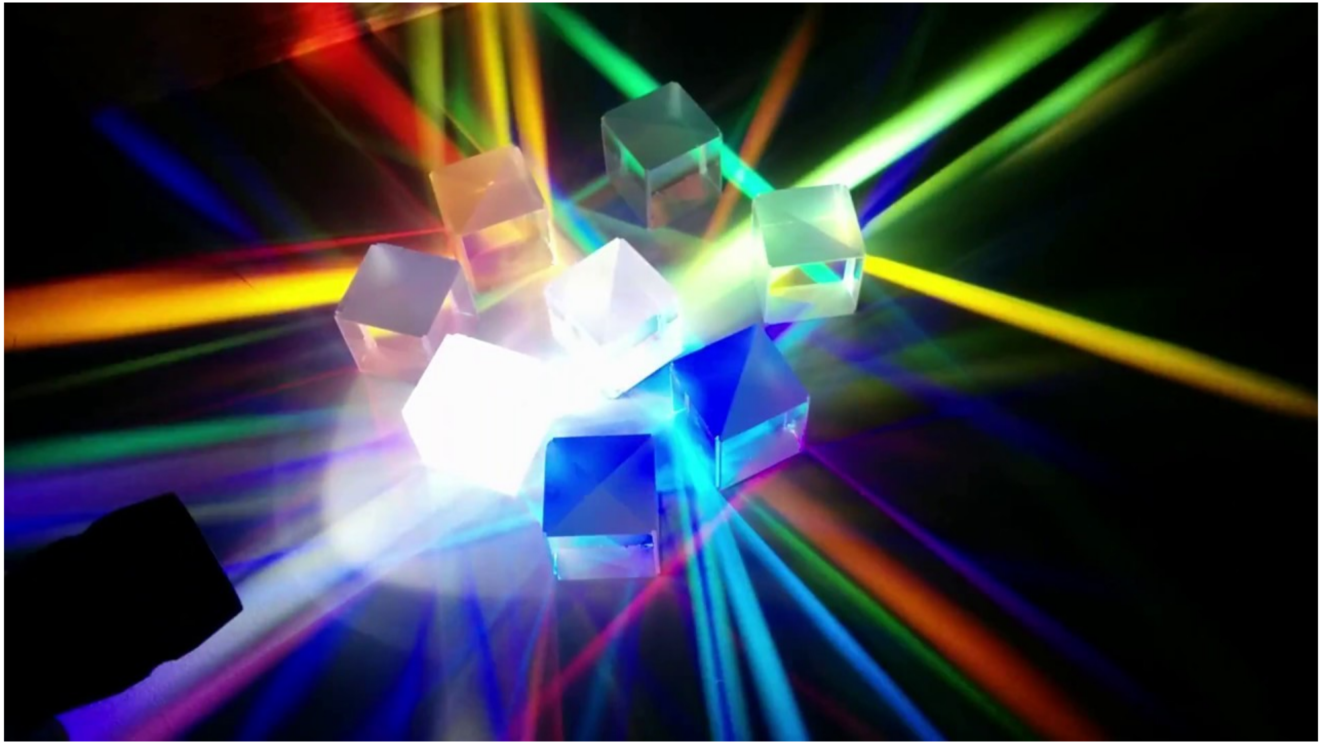
- overview of light-matter interaction
- modelling appearance (gentle intro)
- topic is very vast (and fascinating) ...
- important take-away: known well the theory before developing/using approximations

UE5

Pixar

material appearance

CS4515 3D Computer Graphics and Animation



Ricardo Marroquim

Delft University of Technology (TU Delft)

images

- non-authored images that were not referenced were taken from
  - Real Time Rendering Book
  - Physically Based Rendering Book