Digital Material Appearance

TI3235TU Visual Data Processing

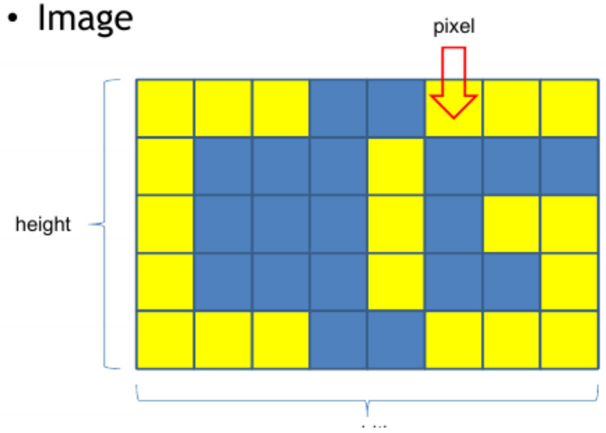
Lecture 7

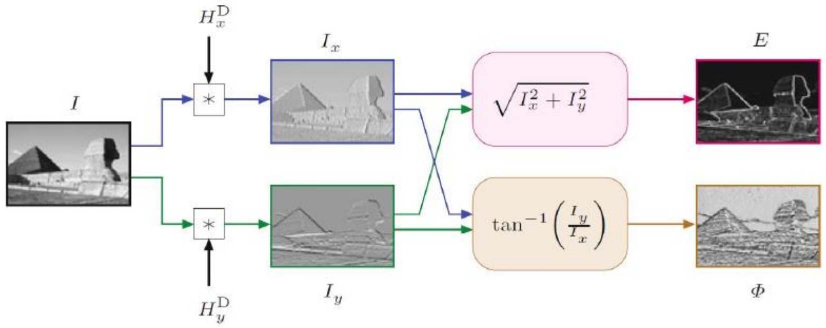
2024/2025

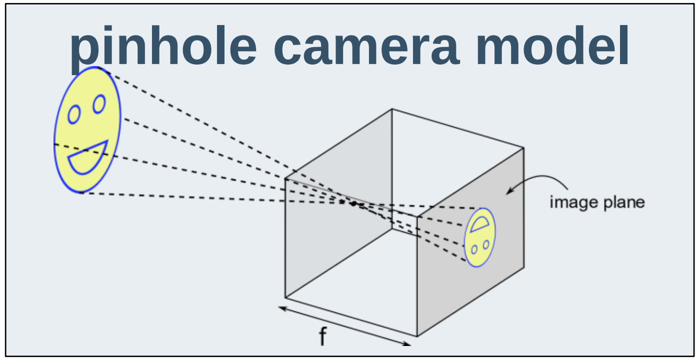
Michael Weinmann

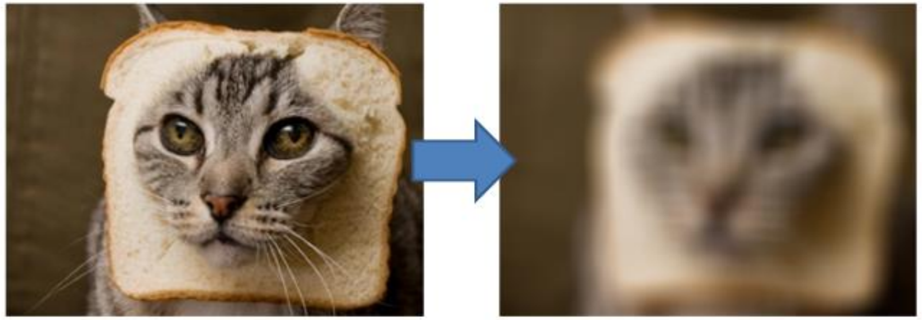
The story so far …

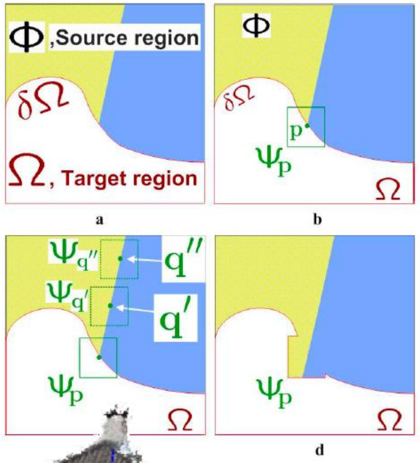
- Image

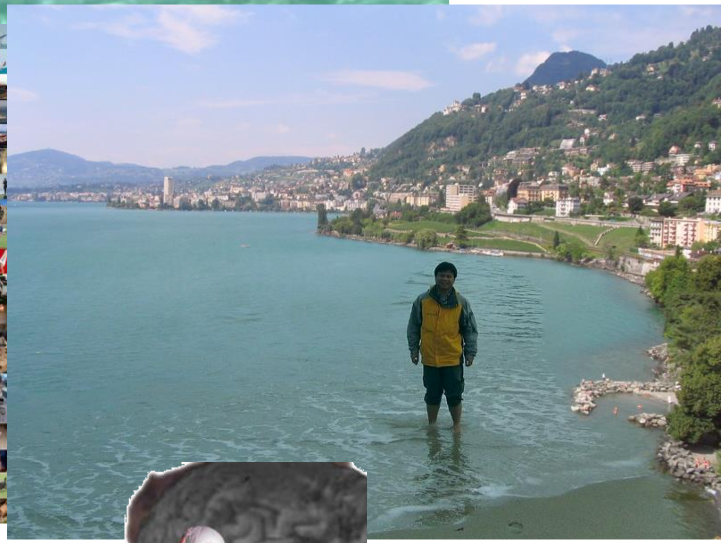


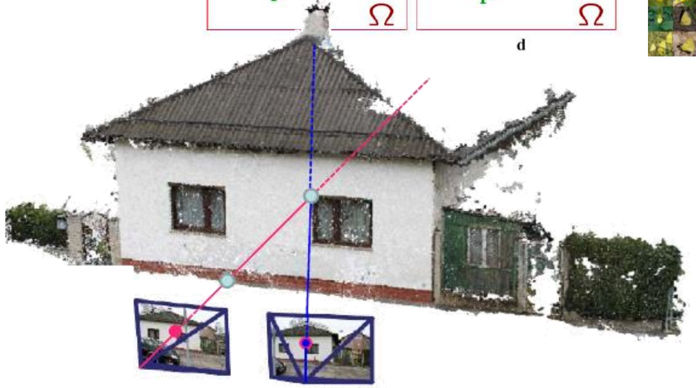


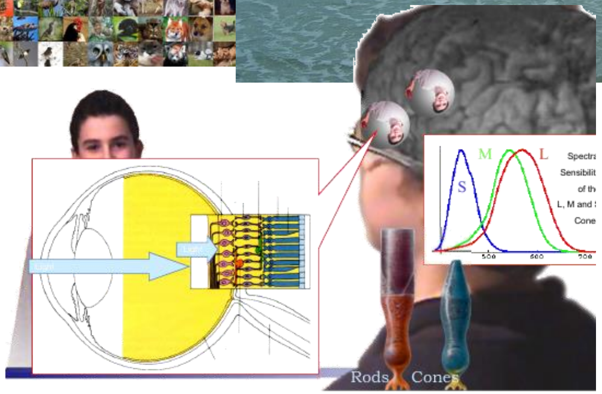






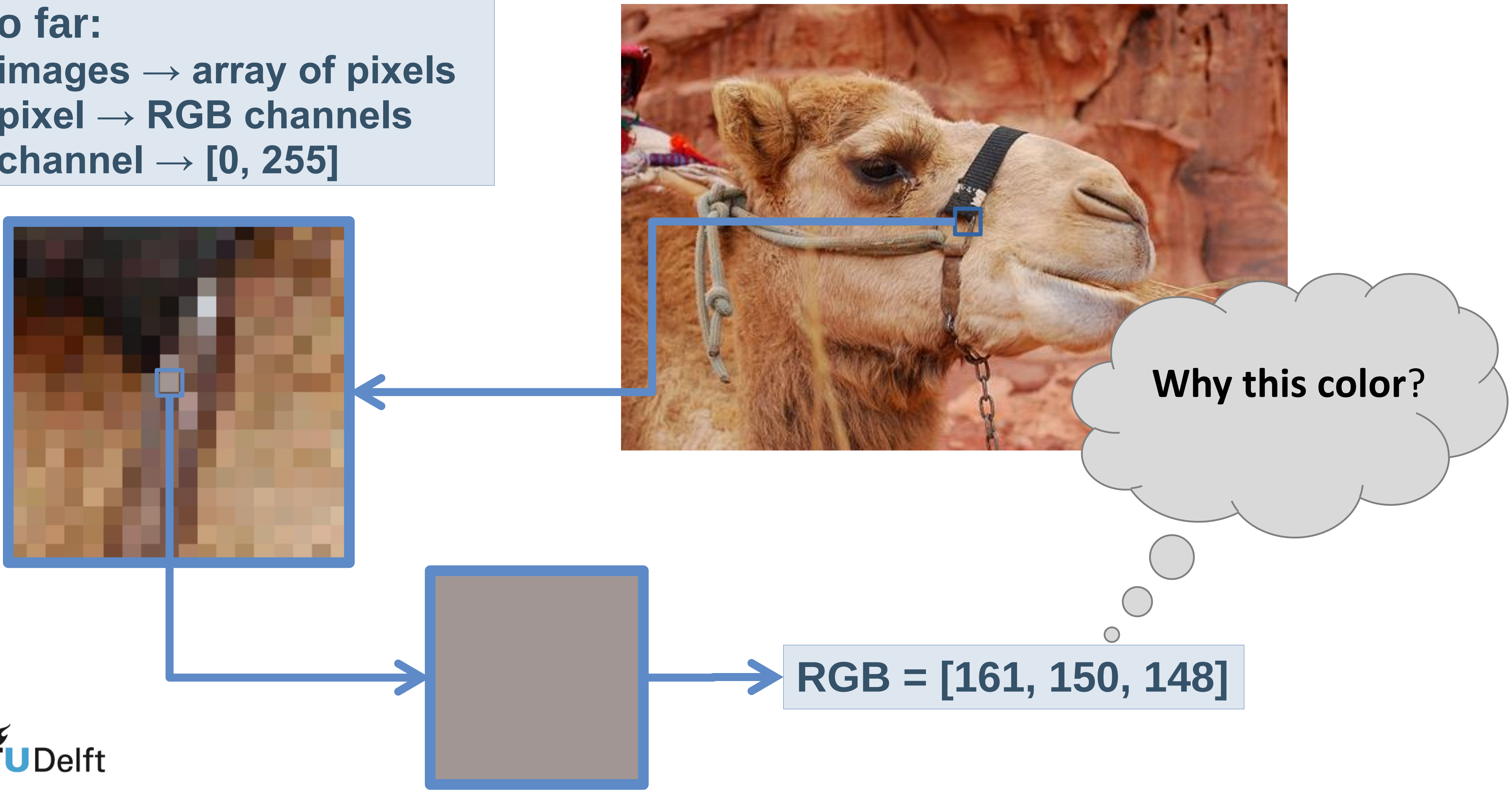






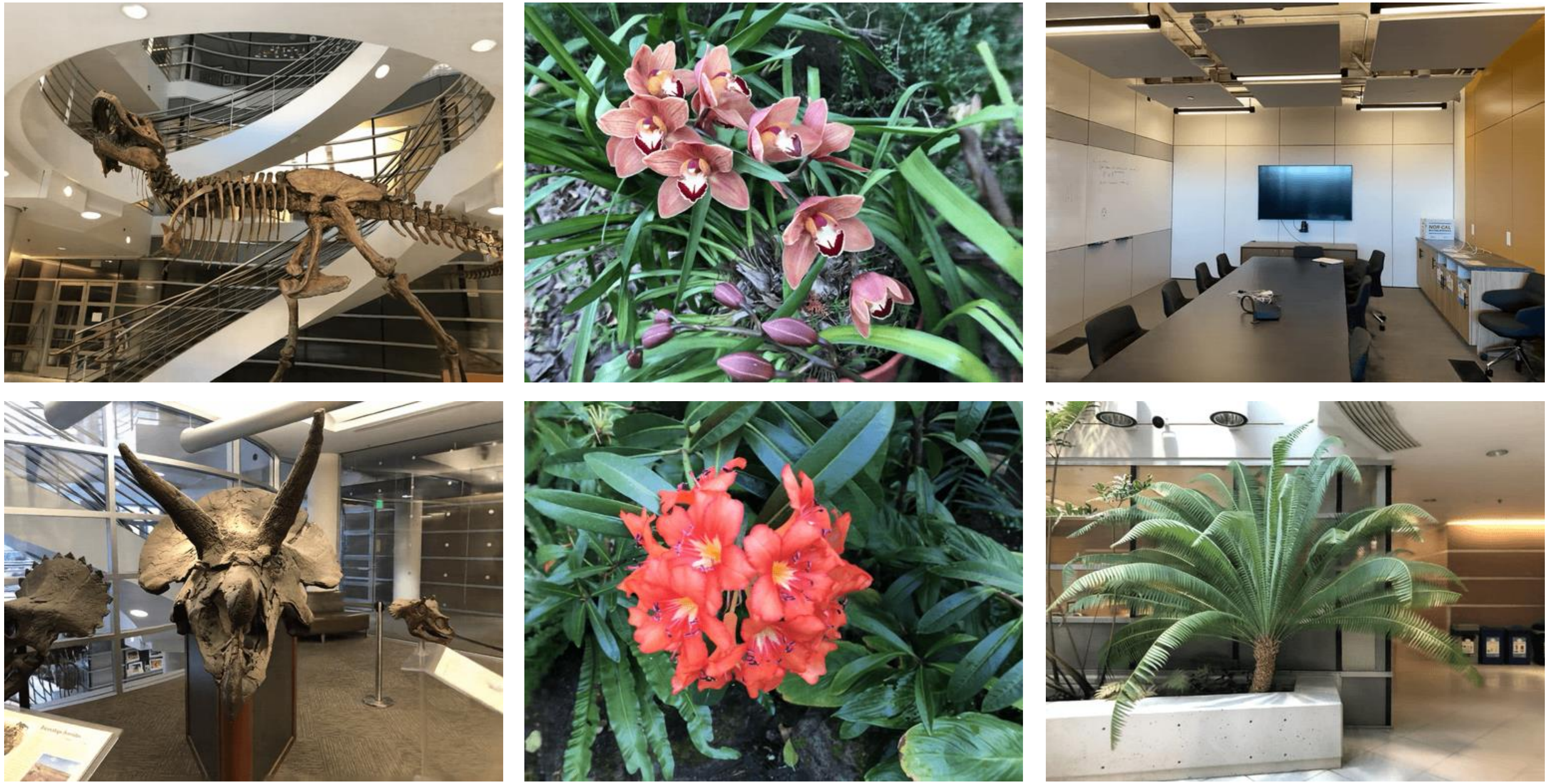
The story so far …

so far:
- images → array of pixels
- pixel → RGB channels
- channel → [0, 255]



But …

› Appearance of surface points may vary for different configurations



Images taken from *Alex Trevithick and Bo Yang. "GRF: Learning a General Radiance Field for 3D Scene Representation and Rendering.“, arXiv/github 2020*

Today's Lecture

Digital materials:

1. **What is relevant** for digitizing (optical) material appearance?
2. **What & how** do we **measure**?
3. **How to infer** a digital representation?

Reality

Virtual scenes

Why do we care about material appearance?

https://visualise.com/2017/11/education-vr-5-examples-bending-reality-enhance-learning

https://medicalview.org/canadas-first-vr-medical-training-centre/

Design

Exploring virtual environments

Education

Entertain-ment

https://geekologie.com/2014/02/count-me-in-playing-skyrim-in-virtual-re.php

https://www.virtualrealityhire.com/wp-content/uploads/2017/02/samsungvrevent.jpg

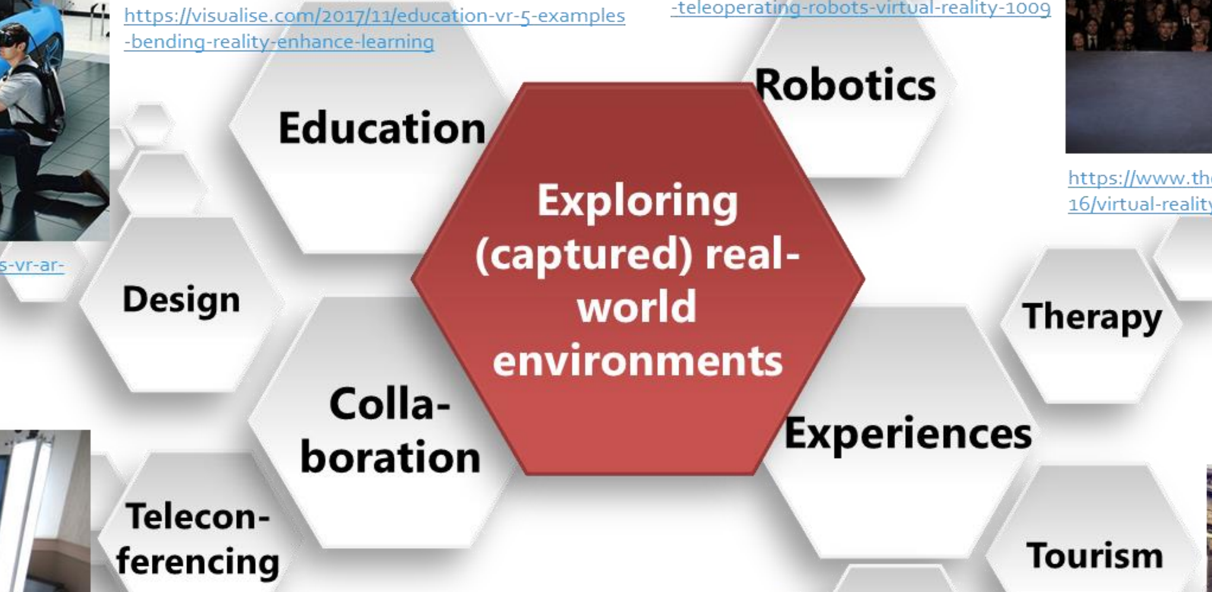
Why do we care about material appearance?

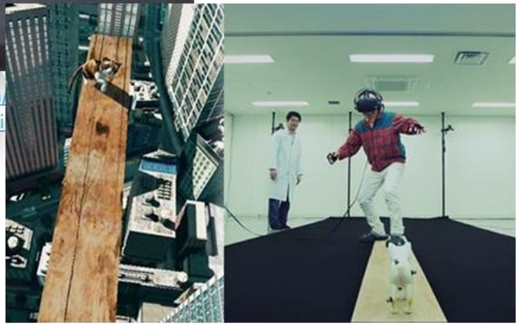
https://visualise.com/2017/11/education-vr-5-examples
-bending-reality-enhance-learning

http://news.mit.edu/2017/mit-csail-new-sy
-teleoperating-robots-virtual-reality-1009

https://www.theguardian.com/
16/virtual-reality-phobias-publi

https://xrsweek.com/2017/08/vrs-focus-vr-ar-design/





https://grapee.jp/en/60459

Holoportation [Orts-Escolano et al. 2016]

https://medium.com/arjs/why-web-apps-are-the-future-of-augmented-reality-c503e796a0c5

https://scooterise.com/modern-way
-exploring-ancient-monuments/

https://medium.com/frulix/virtual-reality-emerging-backbone-of-tourism-industry-7b6000a526c0

Why accuracy matters …

Product Advertisement

Food „Photography“

- Cultural Heritage
- Visual Prototyping
- Advertisement
- Entertainment
- …

Accurate reproduction of characteristic „look“ and „feel“

Cultural Heritage

Entertainment

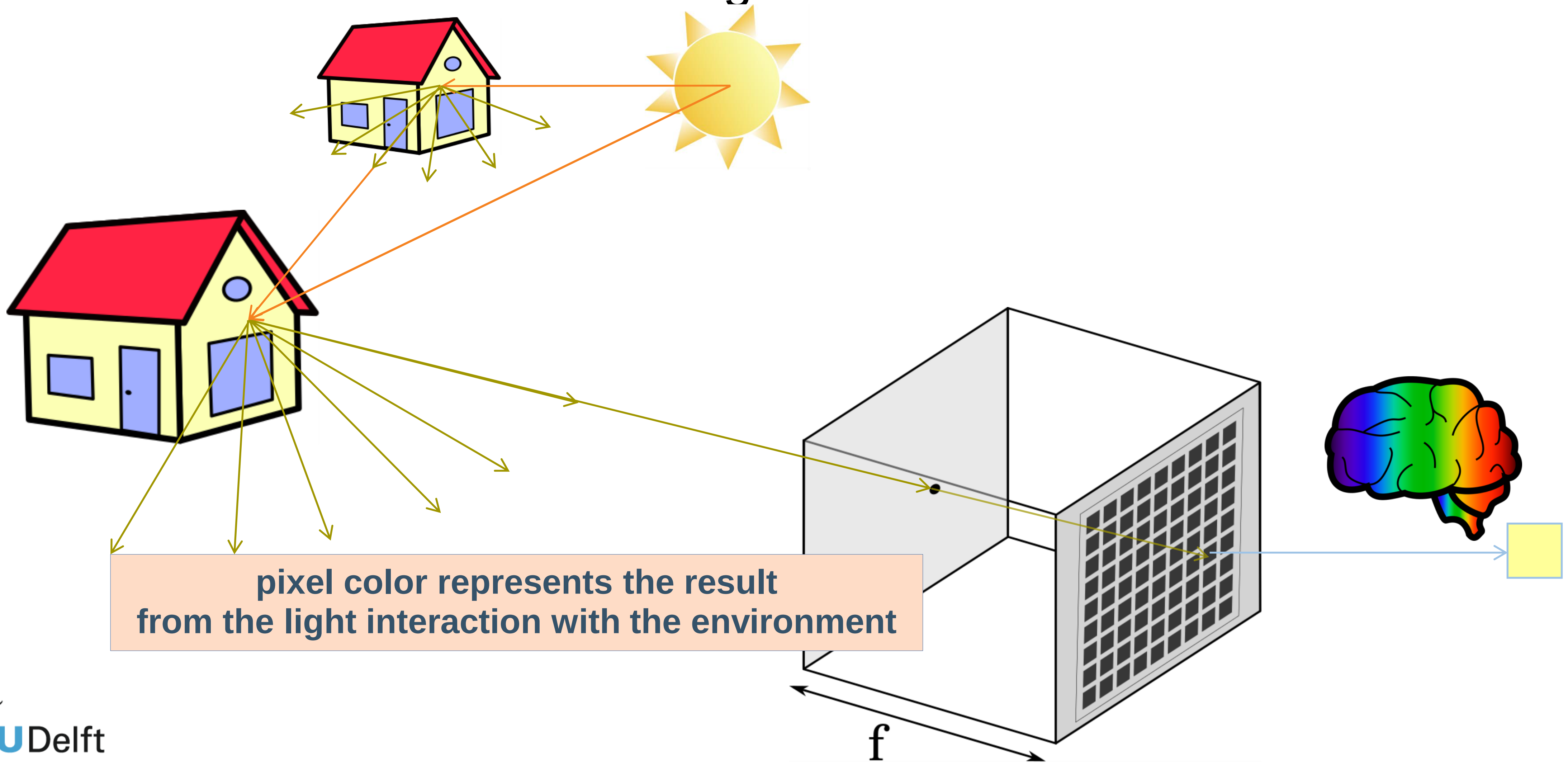
Appetizer

- Which of the depicted objects is a fake?

Basics of Material Appearance

What do we perceive = Image Formation Process

› Perceived color comes from light interaction

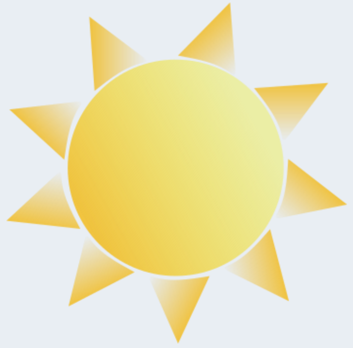


pixel color represents the result
from the light interaction with the environment

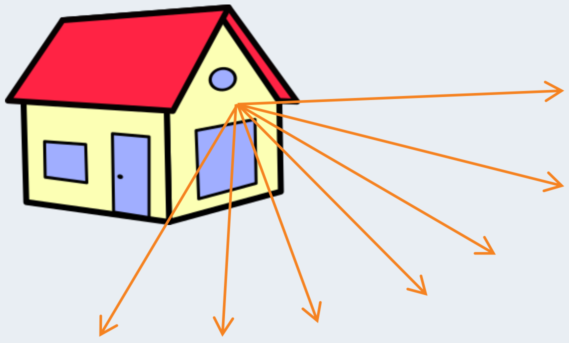
Color = perception

- Pixel colors in a photo depend on:

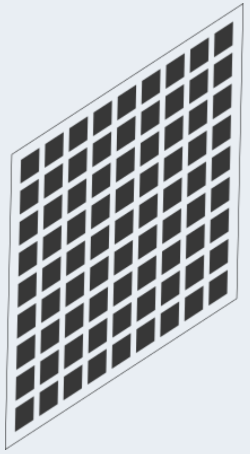
light source(s)
- how it emits light



scene surfaces
- how they reflect light



sensor
- how it captures light



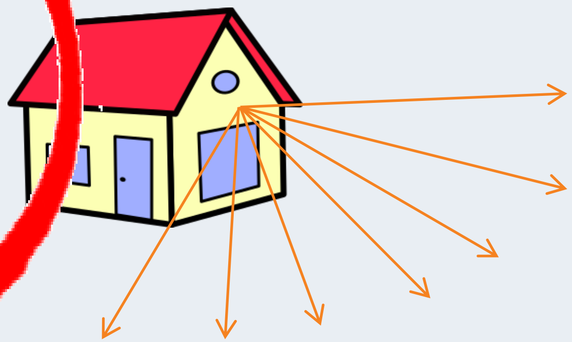
processor
- how it interprets light

Color = perception

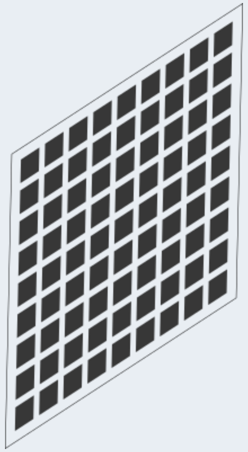
Pixel colors in a photo depend on:

light source(s)
- how it emits light

scene surfaces
- how they reflect light



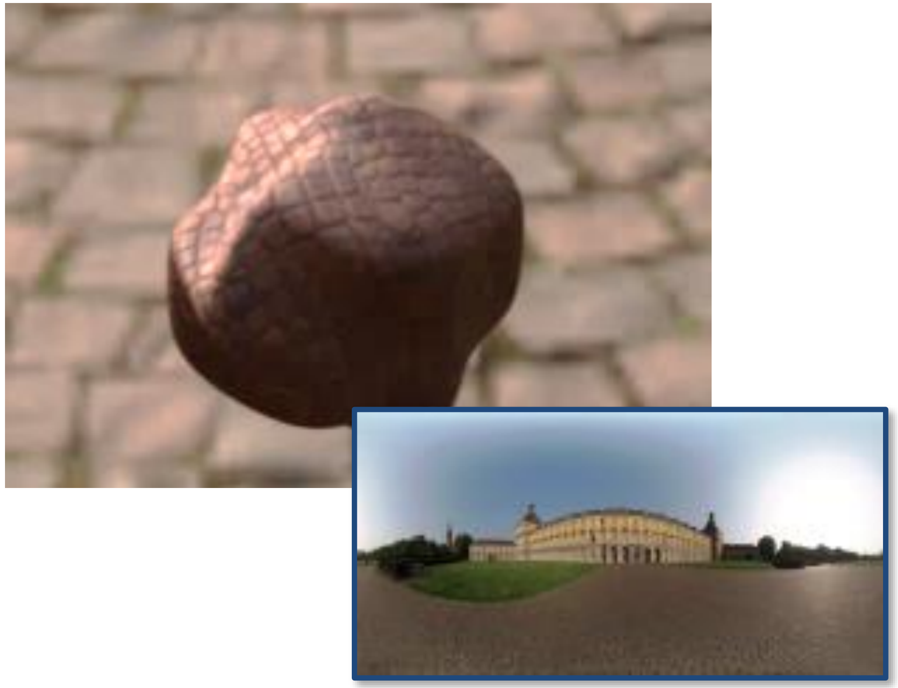
sensor
- how it captures light

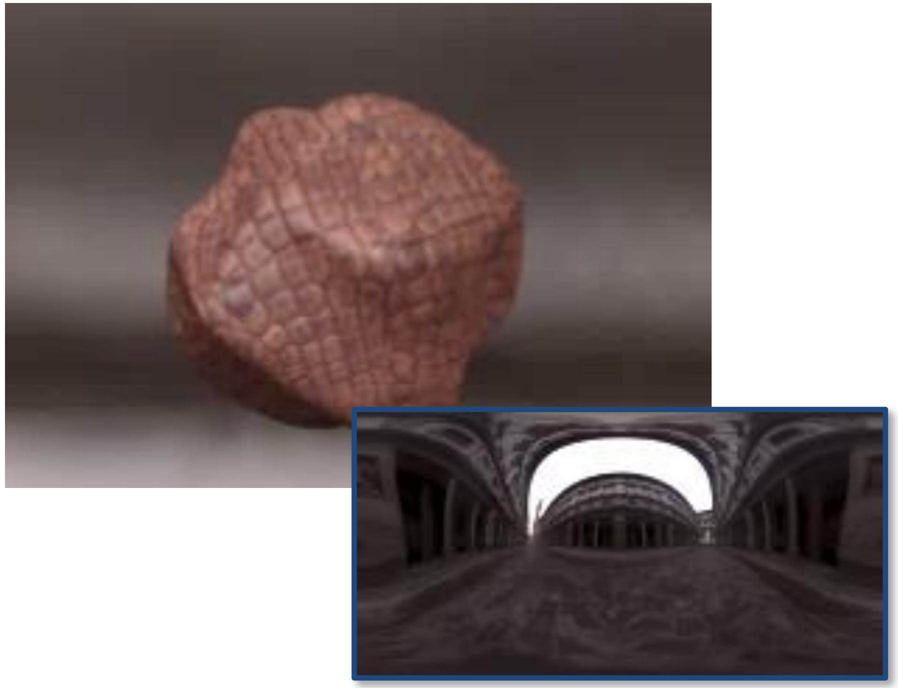


processor
- how it interprets light

Effect of Illumination

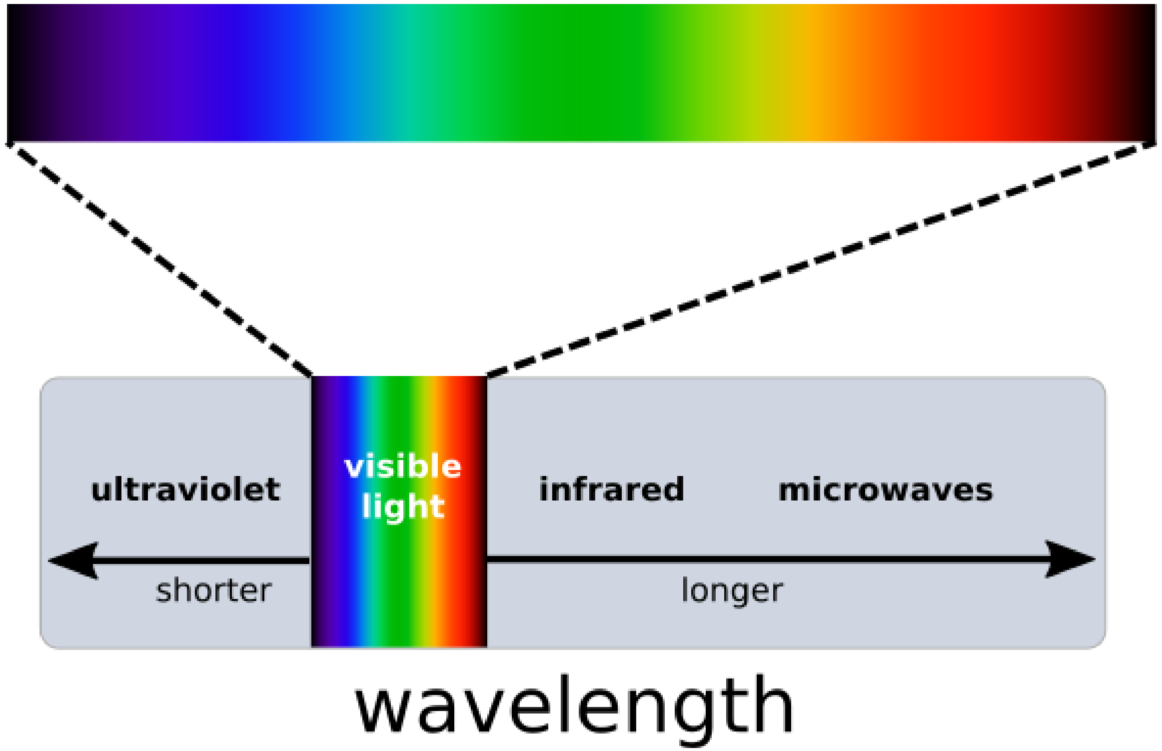
- Observation:
  - Object in photograph appears different under different lighting

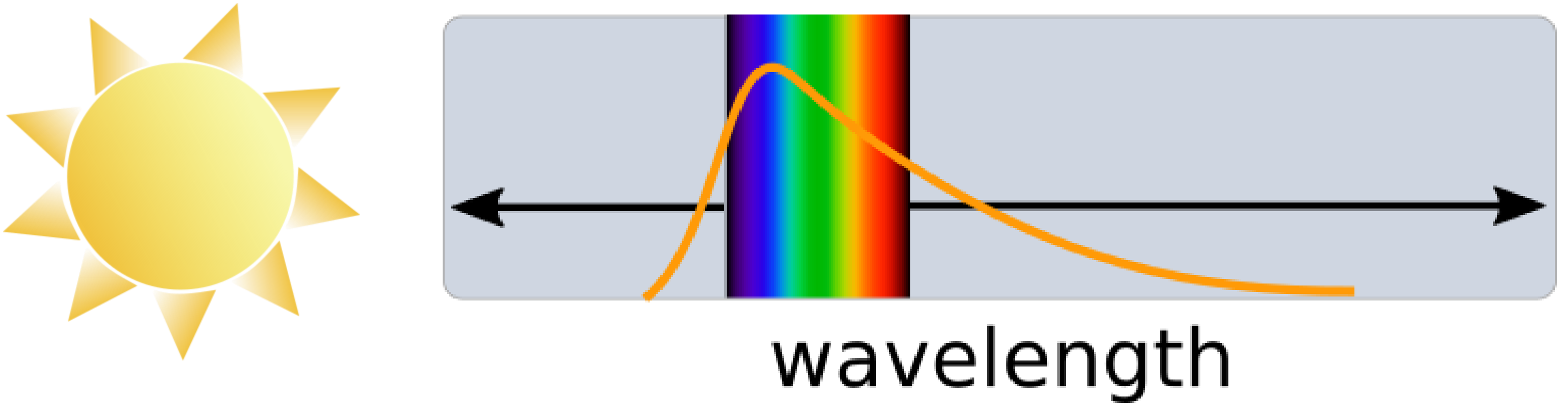




Light Source

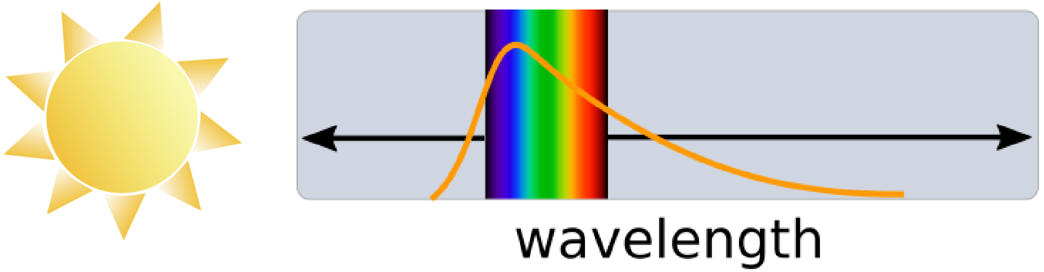
- Light spectrum

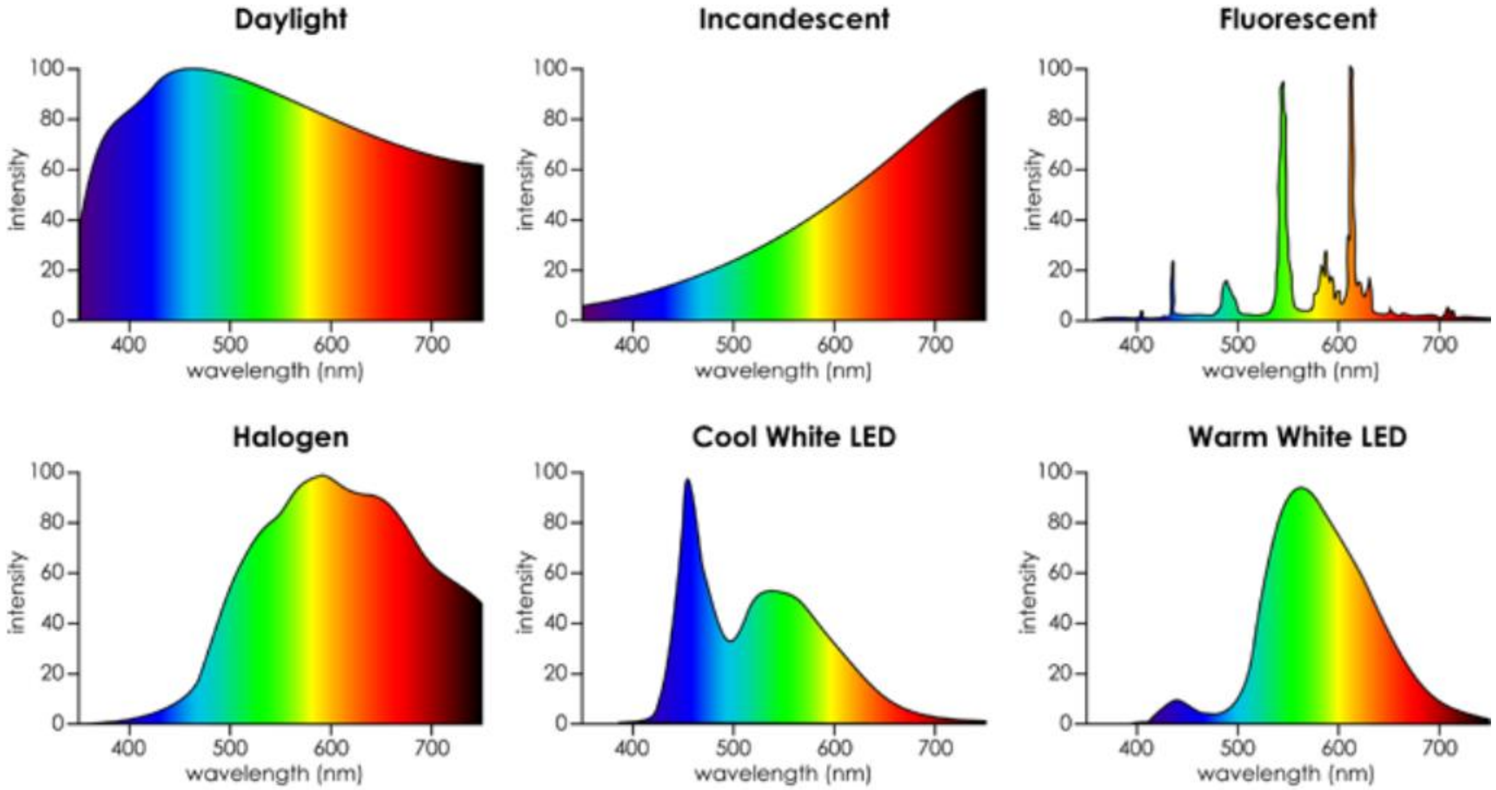




Light Source

› Light spectrum





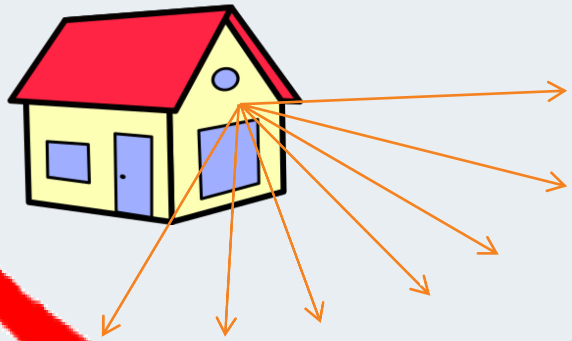
https://www.jfychicago.org/#full-spectrum-led-grow-lights-images

Color = perception

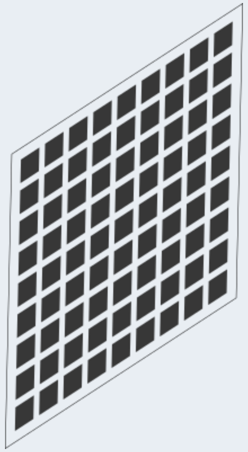
- Pixel colors in a photo depend on:

light source(s)
- how it emits light

scene surfaces
- how they reflect light



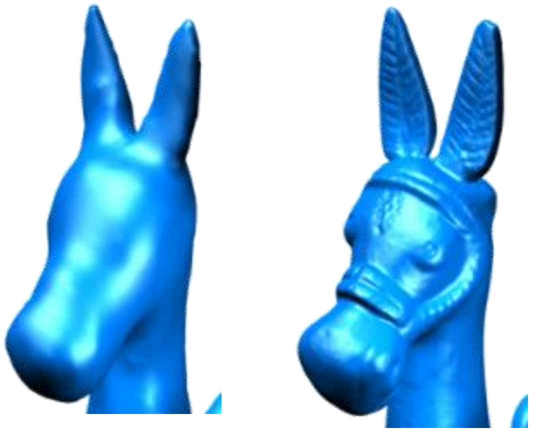
sensor
- how it captures light



processor
- how it interprets light

How do surfaces reflect light?

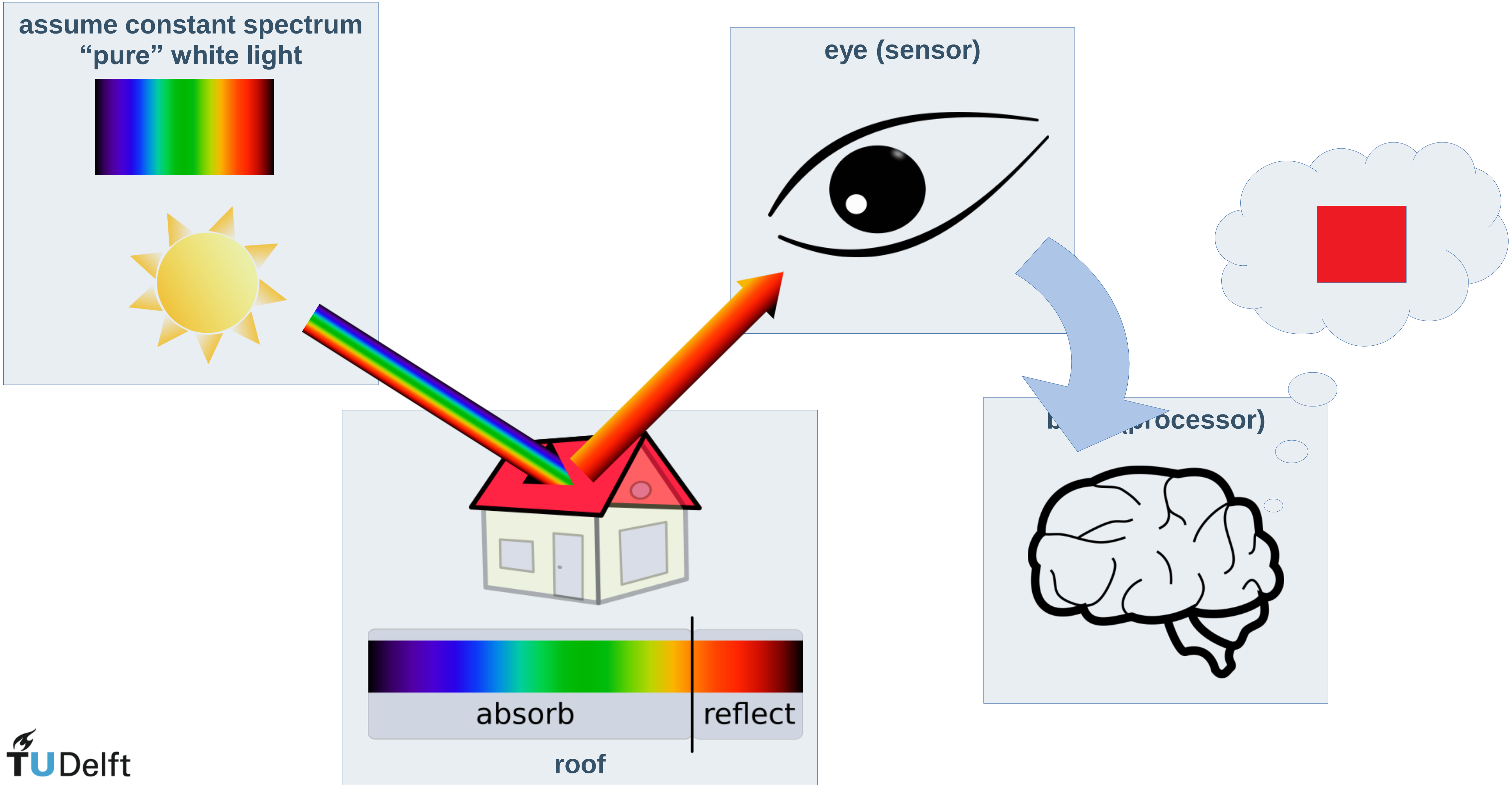
› Depends on surface characteristics



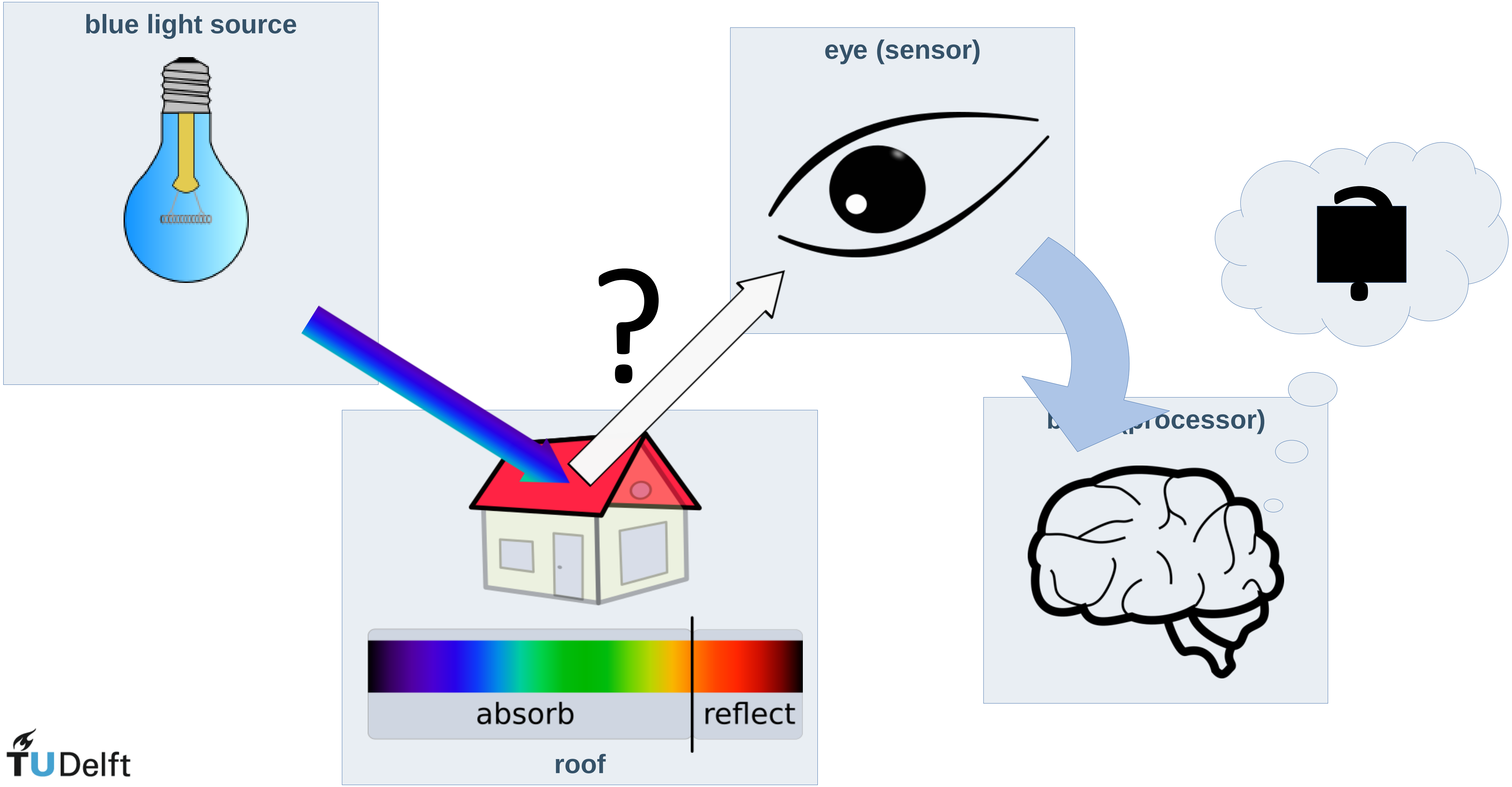
Variation with changing lighting

Geometric surface/object details ***Optical material properties***

What color do we perceive?

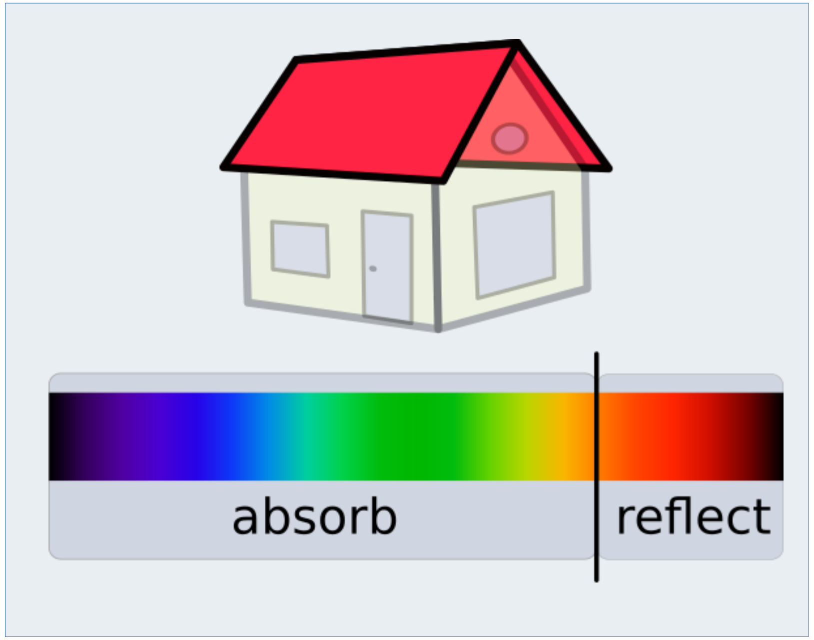


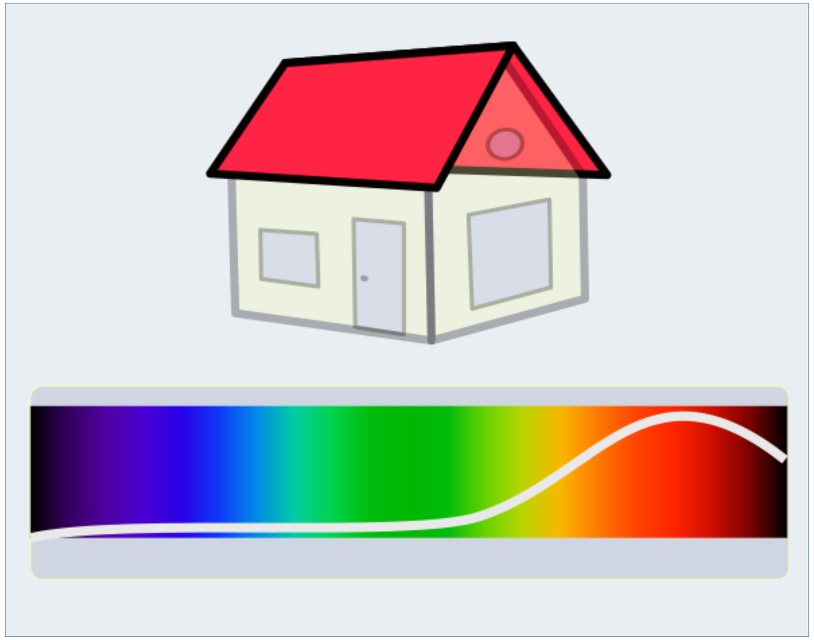
What color do we perceive?



What color do we perceive?

› Reflected wavelength

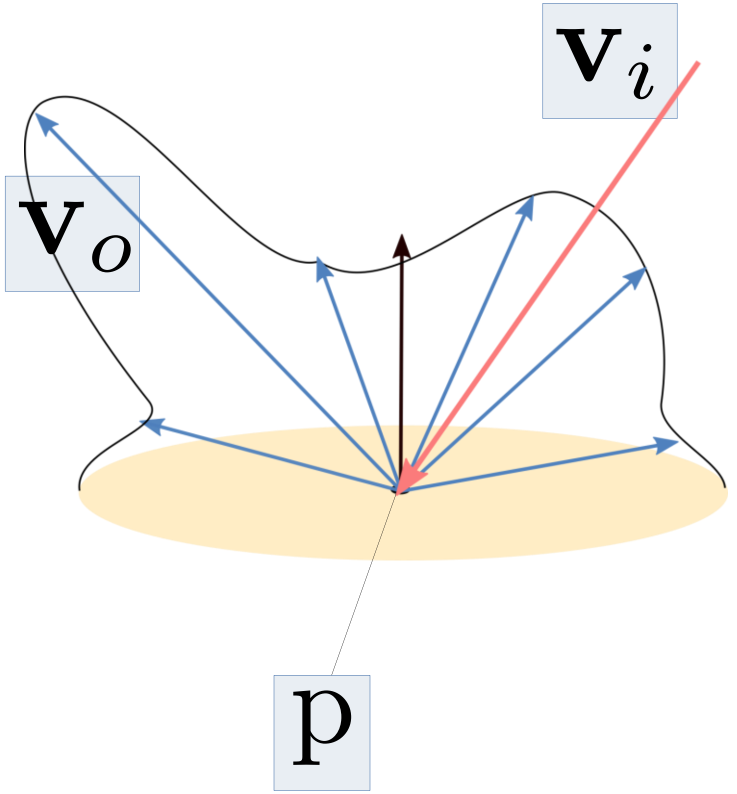


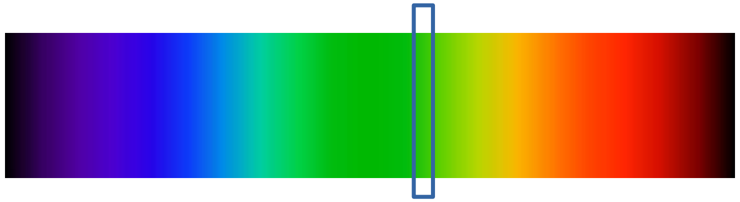


wavelength function is actually continuous

How do surfaces reflect light?

› Reflectance function



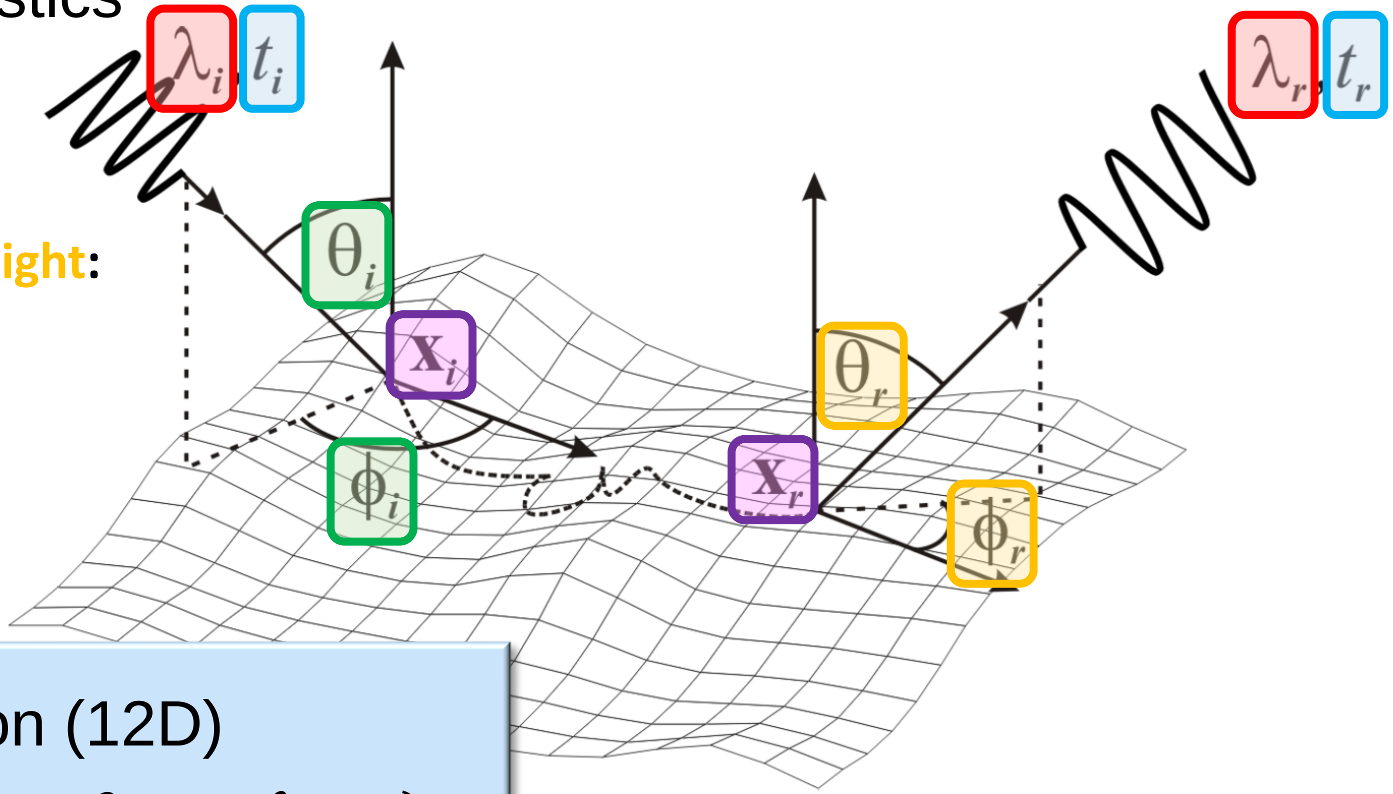


input (depends on):
- point on surface
- incoming light direction
- outgoing light direction
- wavelength

output:
- how much light is reflected in outgoing direction

How to model light exchange at the surface?

- Reflectance function:
  - Describes the material appearance decoupled from the environment, lighting and observer characteristics



Timesteps: t_i, t_r

Directions of incoming/reflected light:

(θ_i, ϕ_i), (θ_r, ϕ_r)

Surface points: x_i, x_r

Wavelengths: λ_i, λ_r

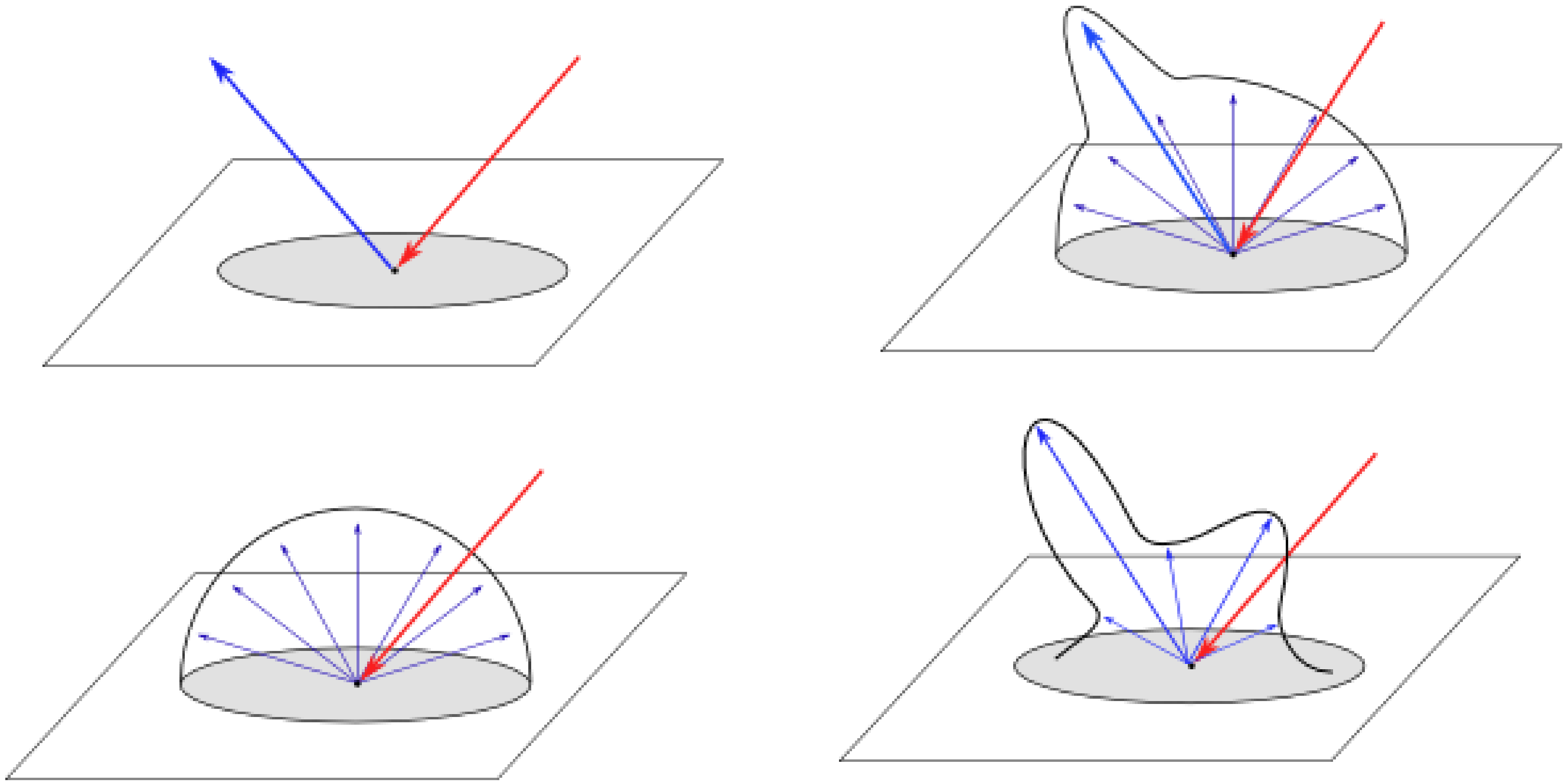
general function (12D)

$$\rho(x_i, y_i, \theta_i, \varphi_i, \lambda_i, t_i, x_r, y_r, \theta_r, \varphi_r, \lambda_r, t_r)$$

TU Delft

How do surfaces reflect light?

› Appearance characteristics vary significantly for different materials

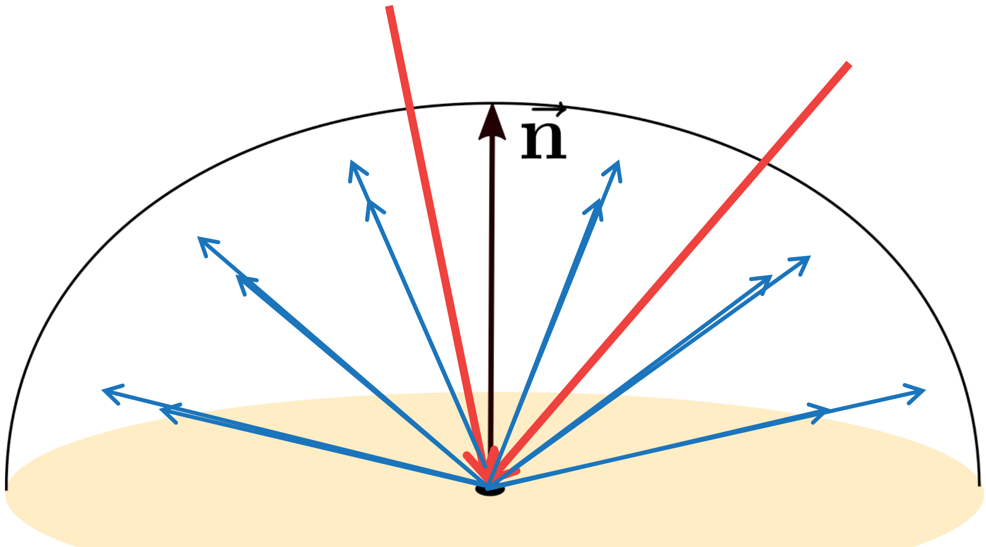


- note: these functions are also wavelength dependent

How do surfaces reflect light?

› Different material properties

light reflects equally
in all directions

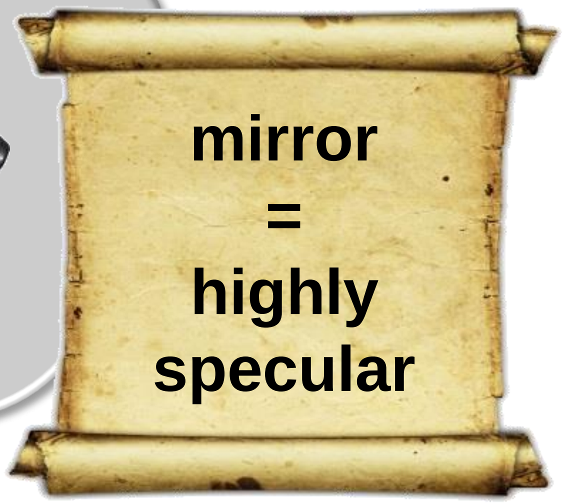


note: intensity of reflection may change

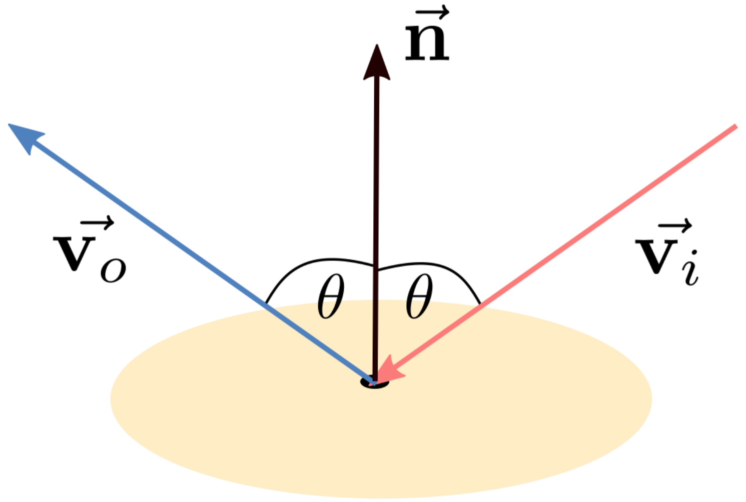
- no highlights
- same perception independent of viewing angle
- depends only on incoming direction

How do surfaces reflect light?

› Different material properties



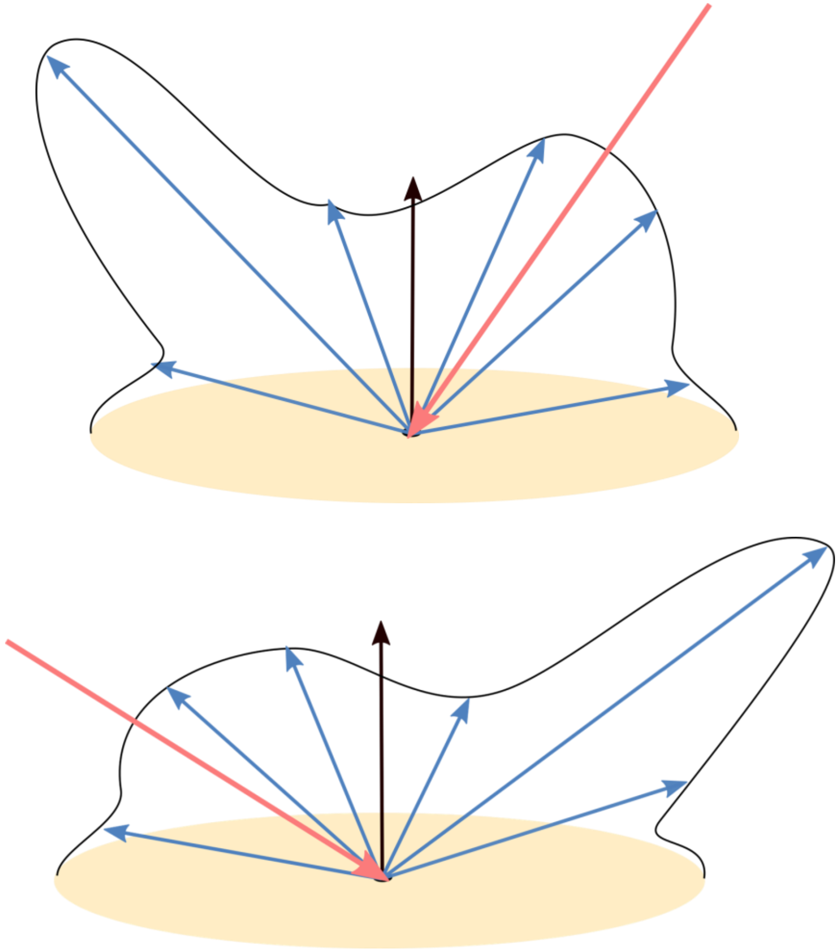
light reflects in a single direction



TUDelft

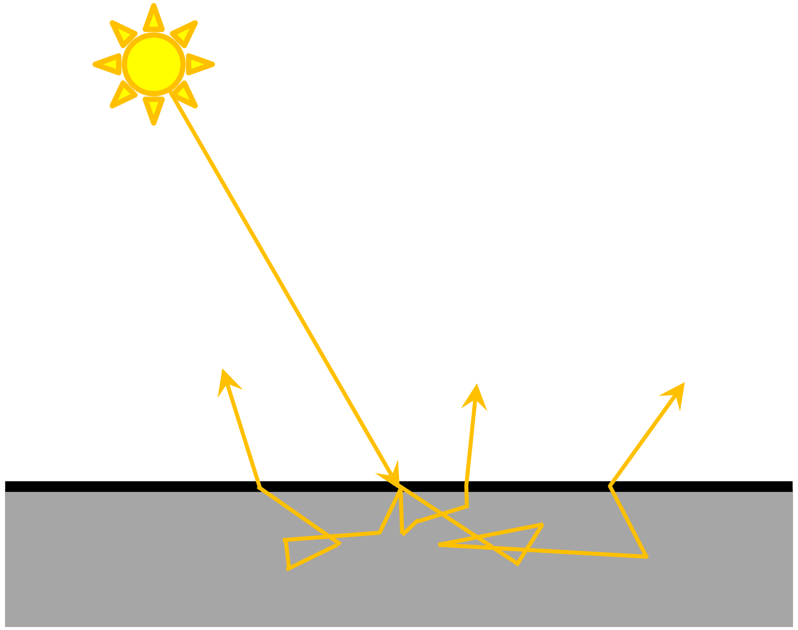
How do surfaces reflect light?

- Different material properties



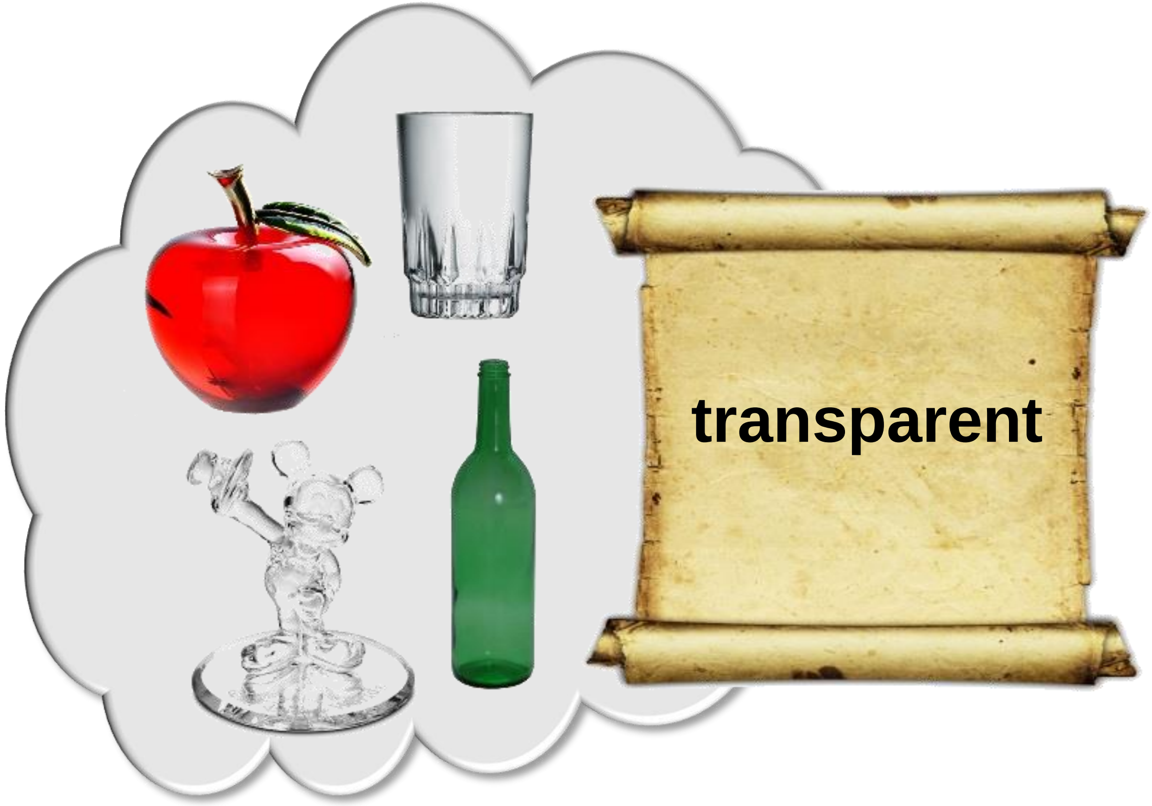
Light exchange at the surface

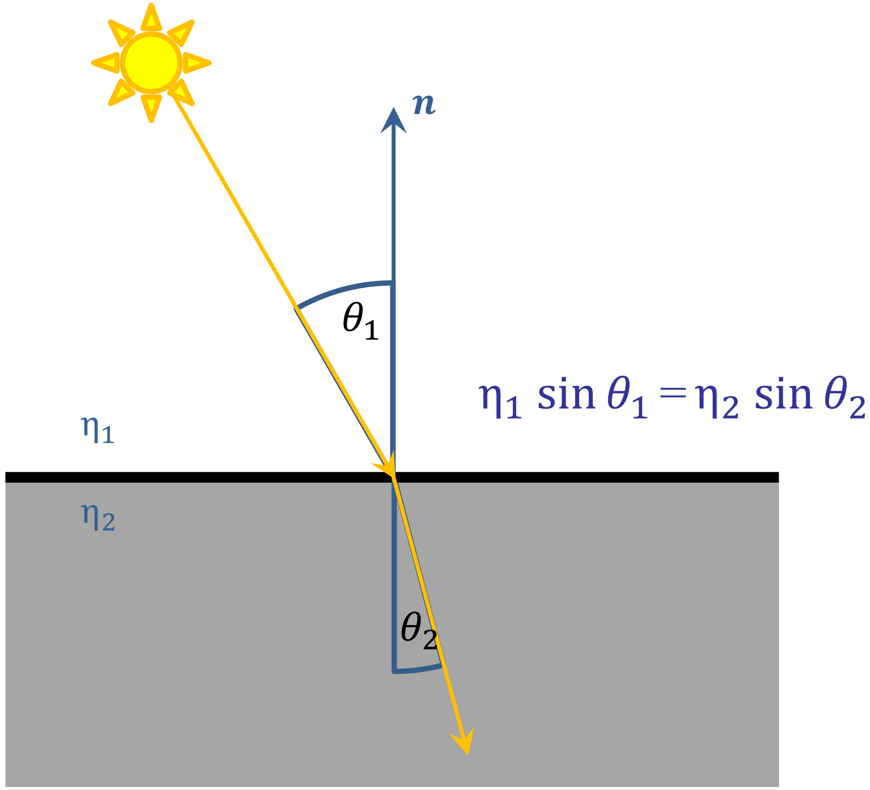
- Different material properties



Light exchange at the surface

- Different material properties



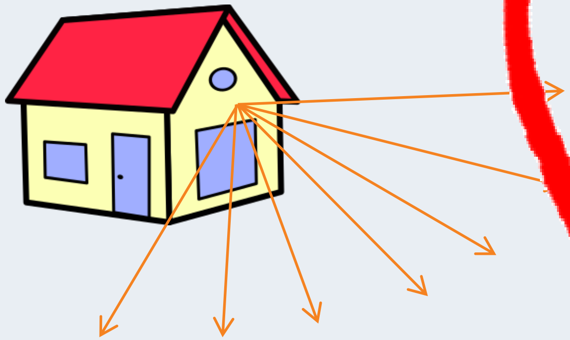


Color = perception

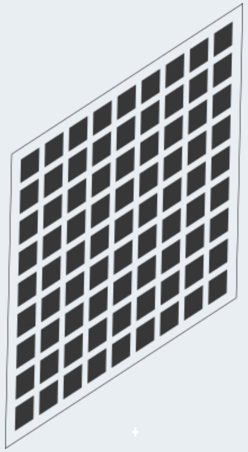
- Pixel colors in a photo depend on:

light source(s)
- how it emits light

scene surfaces
- how they reflect light



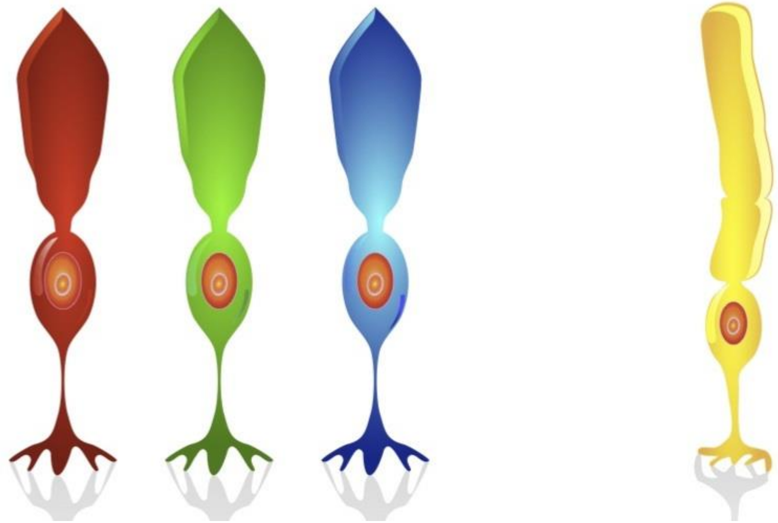
sensor
- how it captures light

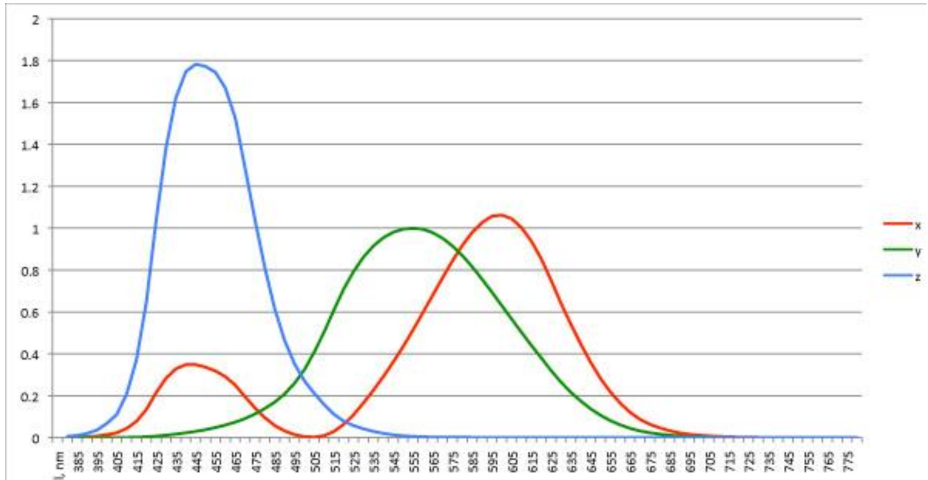


processor
- how it interprets light

How to capture light?

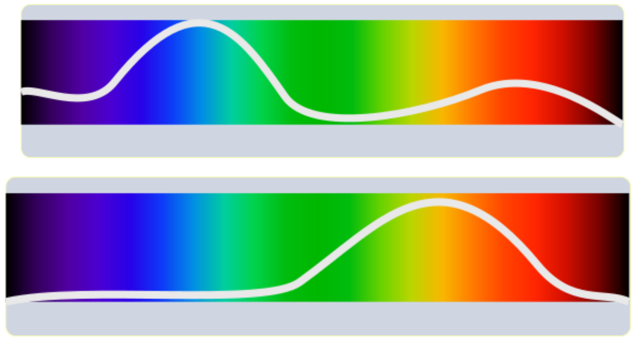
- Inside the eye …

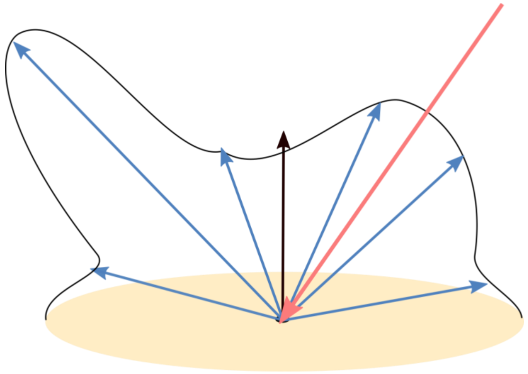




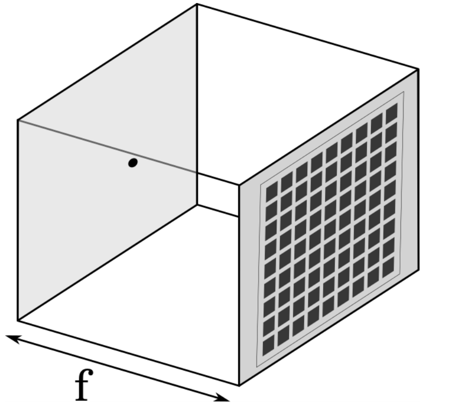
How to capture light?

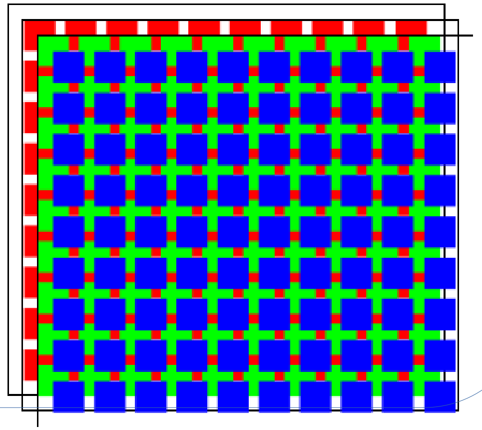
very complex



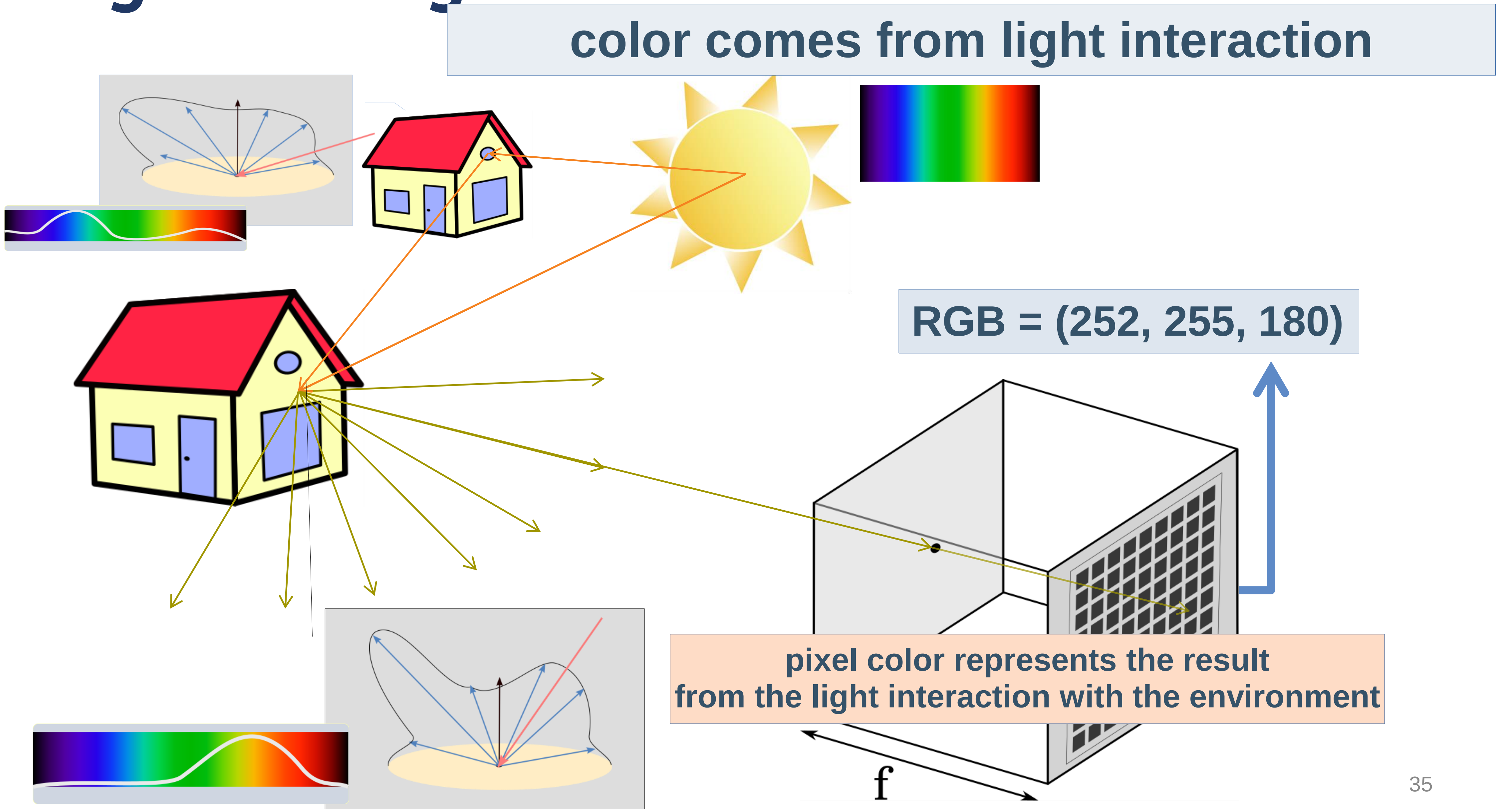


simple





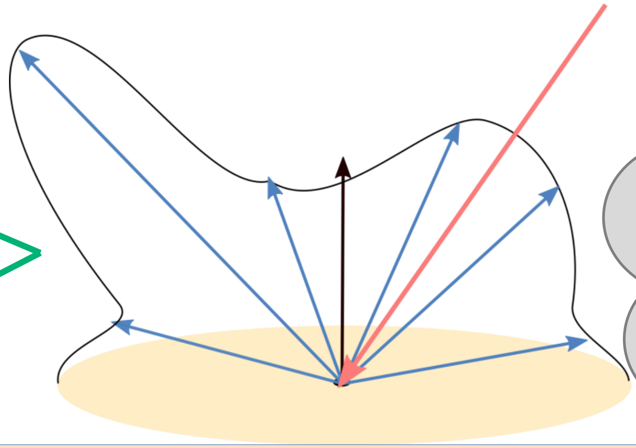
Putting it all together



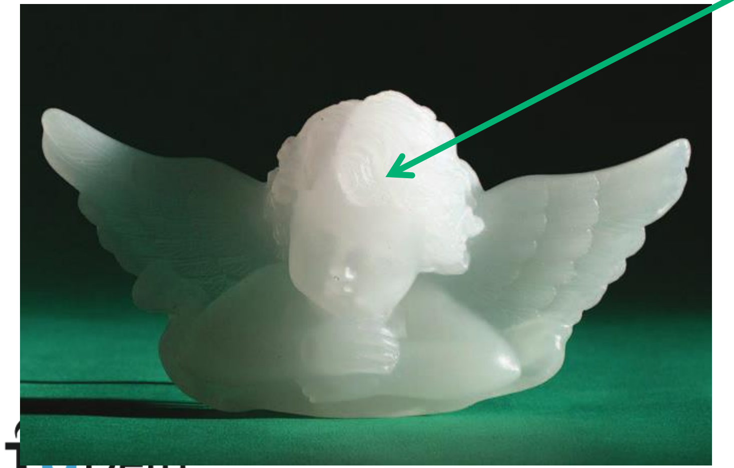
How to capture material appearance?

How do we capture the material?

reflectance function
describes the material appearance independently of the environment



Which setup?
- Camera configurations?
- Lighting configurations?
- Assumptions/parameters to consider?

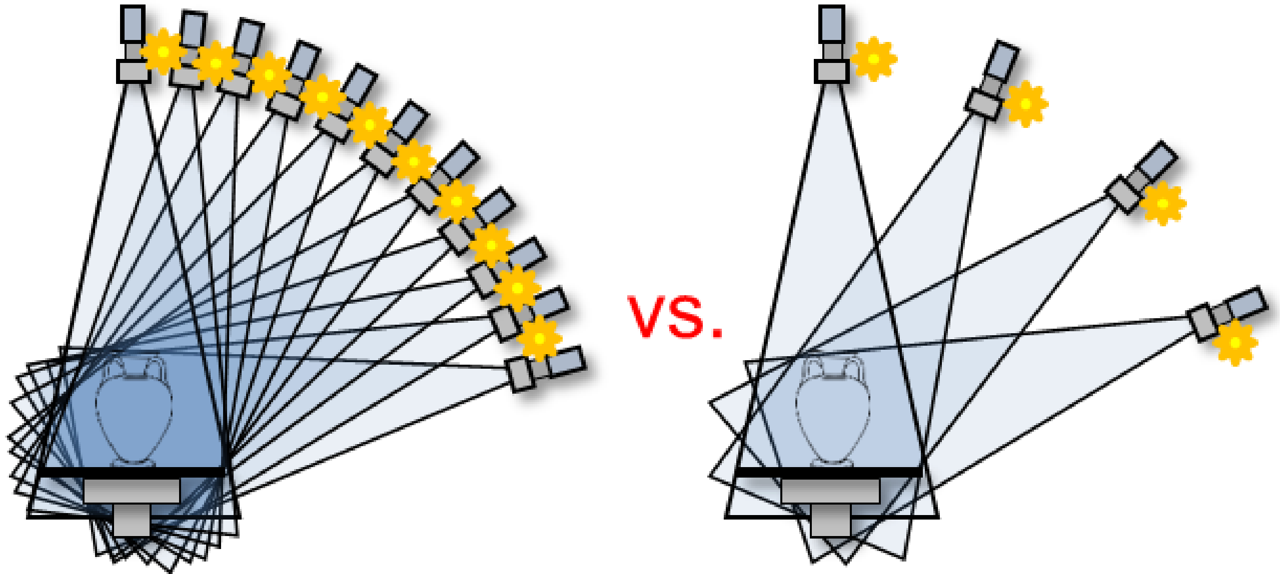


very challenging!!
depends on many variables
- position on surface
- incoming direction
- outgoing direction
- wavelength
- and more ...

Material Appearance Capture

- Idea:
  - Measure appearance of surface points under different conditions

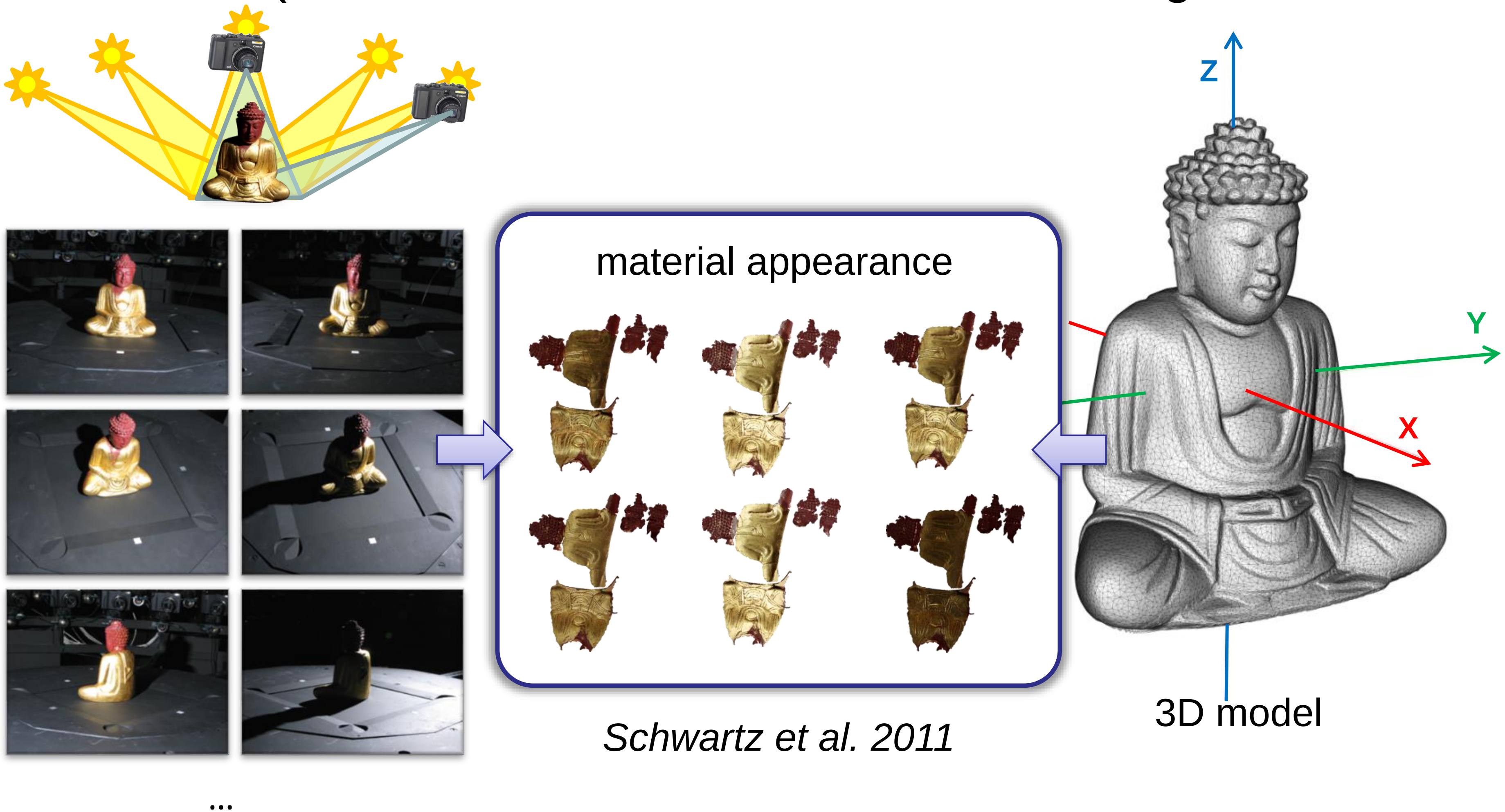
- Challenges:
  - How to capture appearance characteristics:
    - How many measurements? dense vs. sparse
    - Which capture setups? expensive vs. low-cost



How to capture reflectance functions (material appearance)?

› Assumption:

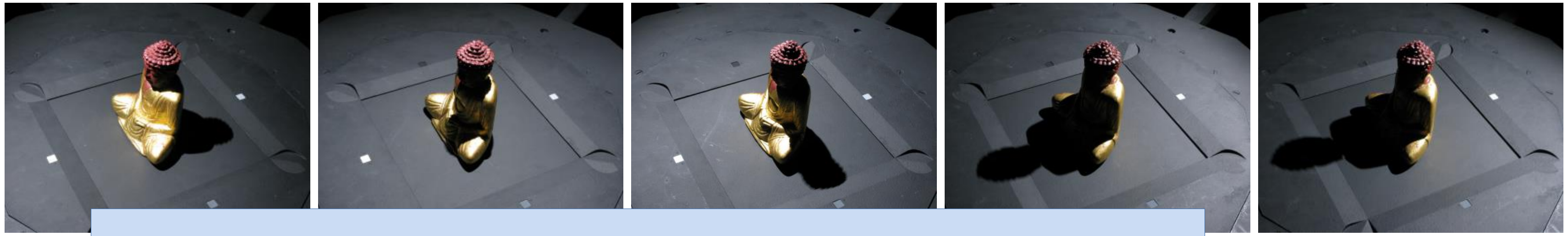
  › known 3D model (multi-view reconstruction, structured light, laser scanner, etc.)



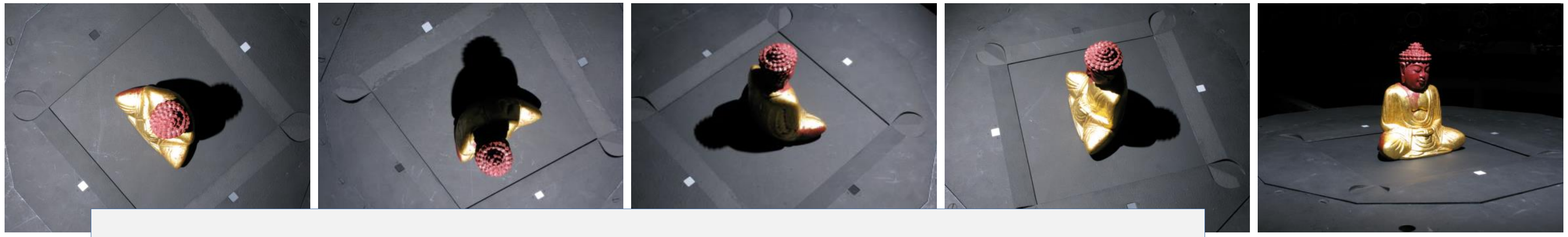
Schwartz et al. 2011

Material Appearance Capture

- Measure appearance of surface points under different conditions:
  - **Assumptions:** no subsurface scattering, no transparencies, no dependence on time and wavelength → measure $\rho(\boldsymbol{x}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_r)$



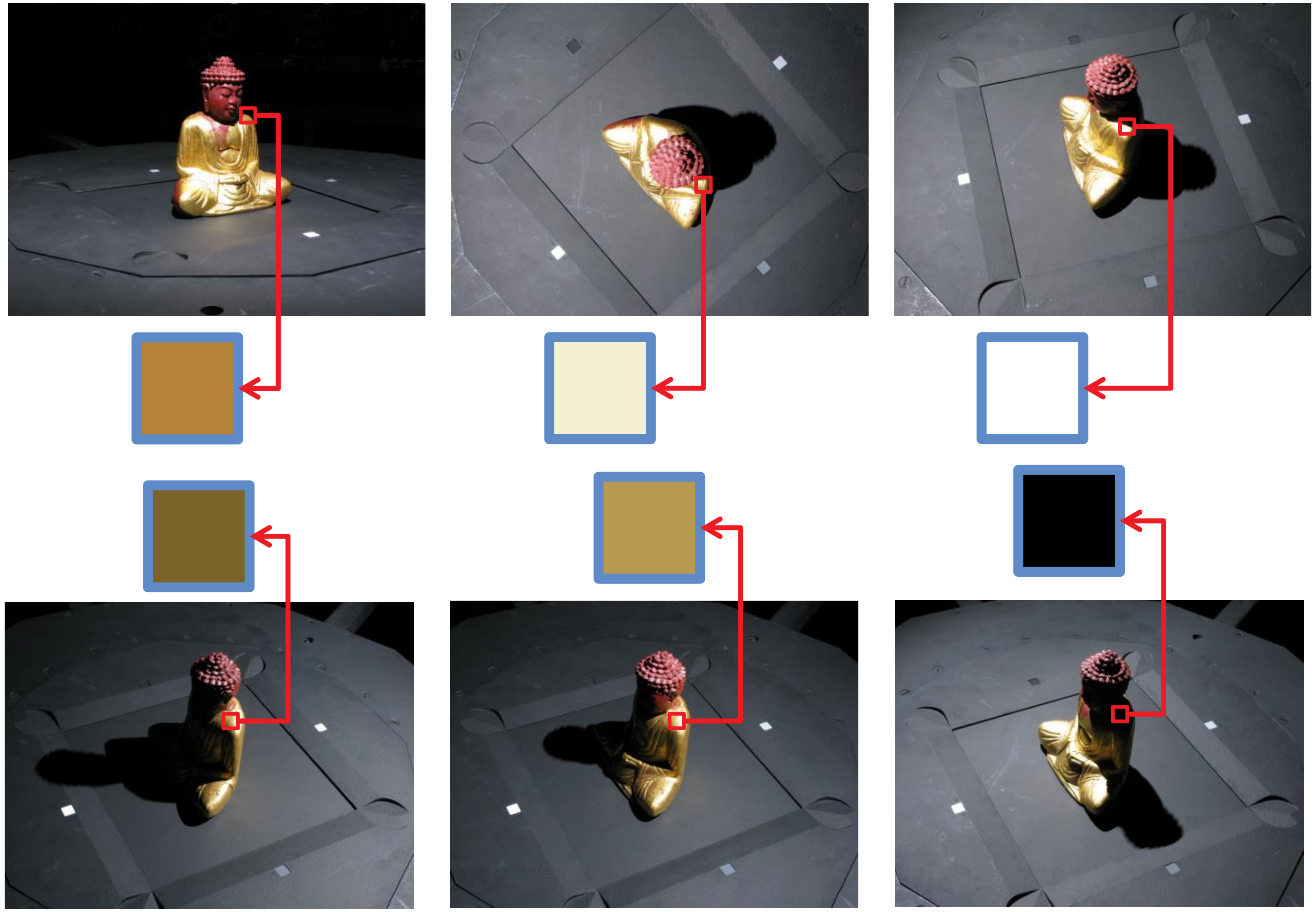
varying incoming direction (light source)



varying outgoing direction (camera)

Material Appearance Capture

- Measure appearance of surface points under different conditions:



Material Appearance Capture

- Setups of individual camera and light source (manual control):
  - Sequential for $\omega_i = (\theta_i, \varphi_i)$ and $\omega_o = (\theta_o, \varphi_o)$
  - Parallel for x

lab settings

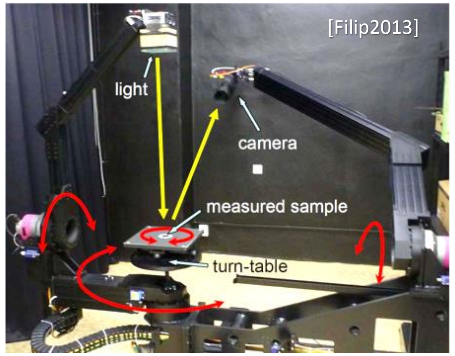
vs.

in-the-wild settings

Material Appearance Capture

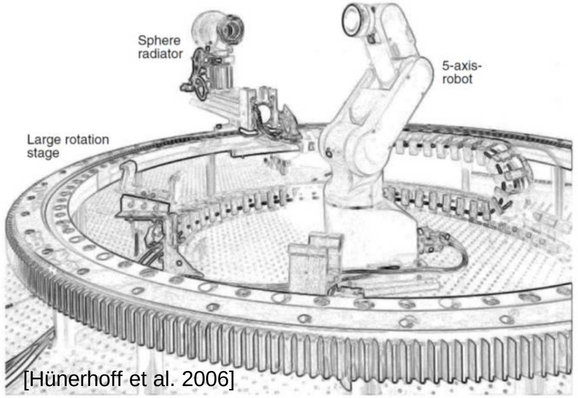
- Setups of individual camera and light source (manual control) → Gonioreflectometer-like setups:
  - Typically 1 camera, 1 light source
  - Sequential for ω_i, ω_o, parallel for x
  - Various flavors:
    - Different number of DOFs
    - Robot rotates sample
      - Fixed light, moving camera
      - Fixed camera, robot moves light
    - Robots move camera **and** light


[Matusik 2003]



[Filip2013]

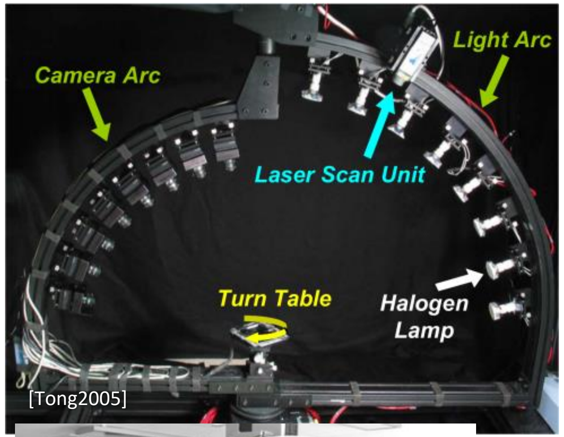

[Holroyd2010]

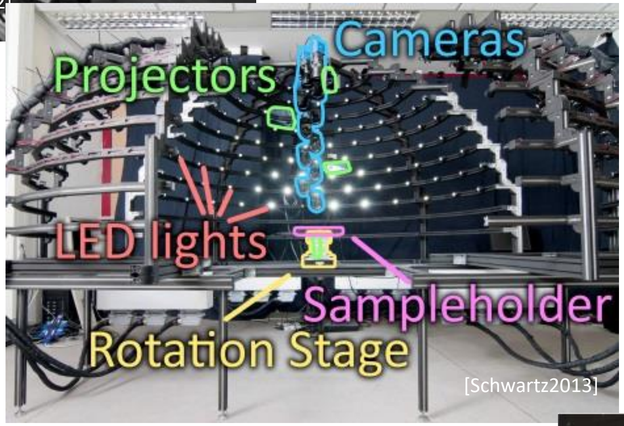


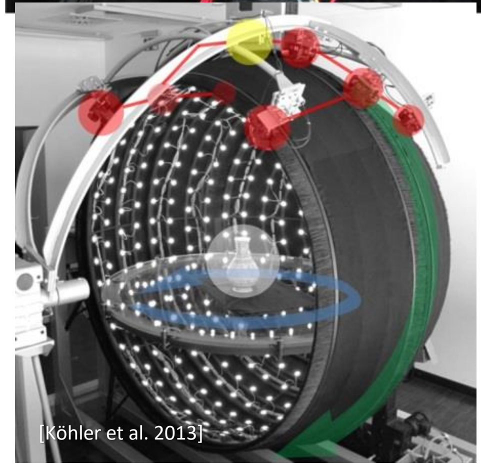
[Hünerhoff et al. 2006]

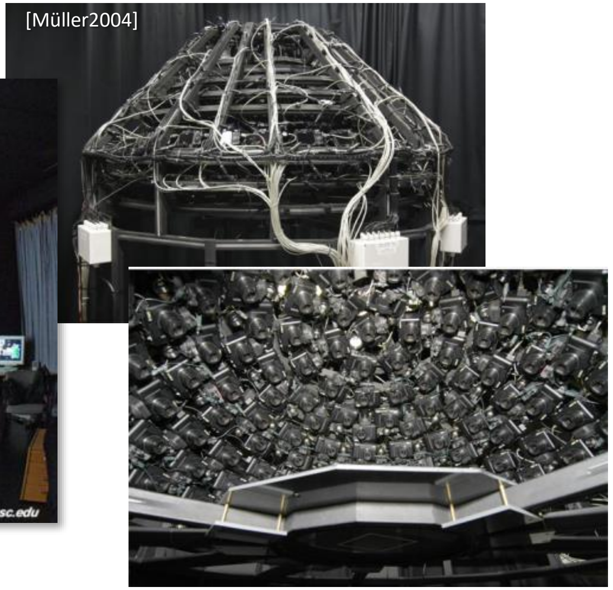
Material Appearan

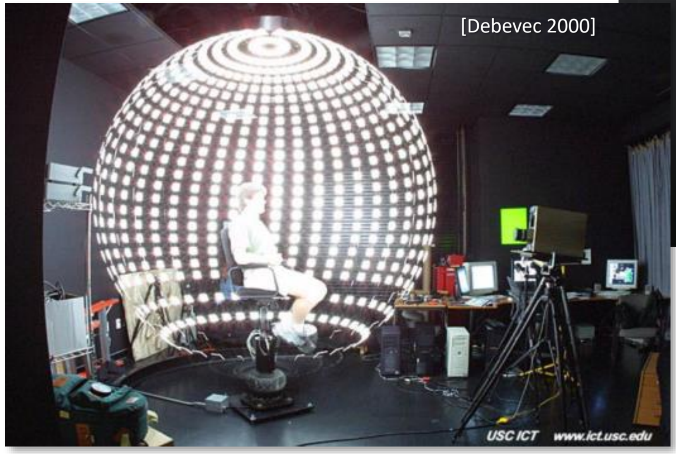
› Camera/light source arrays:
  - Multiple cameras/lights
  - (Semi-)parallel for x **and** ω_o
  - (typically) sequential sampling of ω_i
  - Variants:
    - Turntable rotates object
    - No moving parts

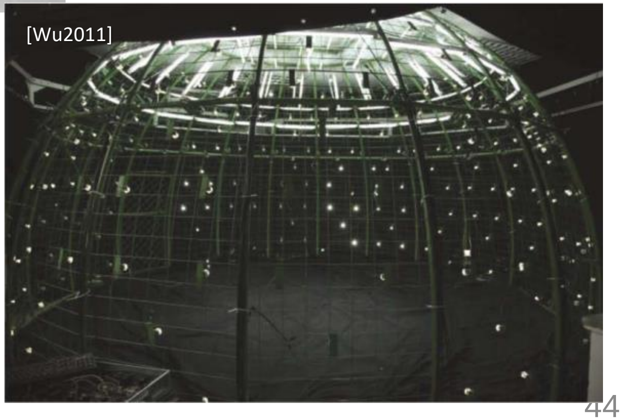








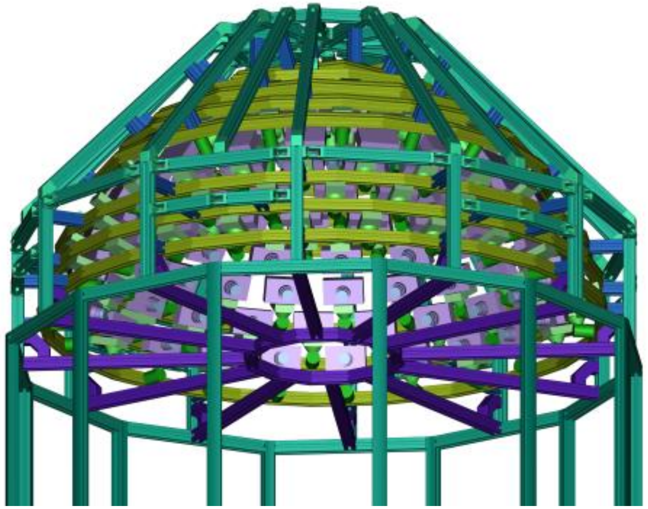


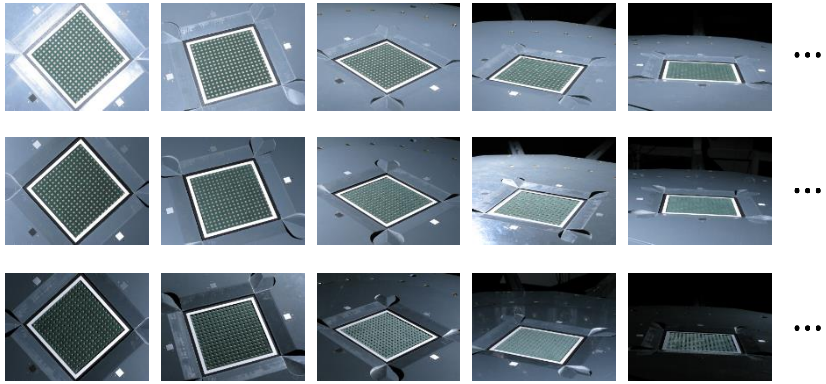


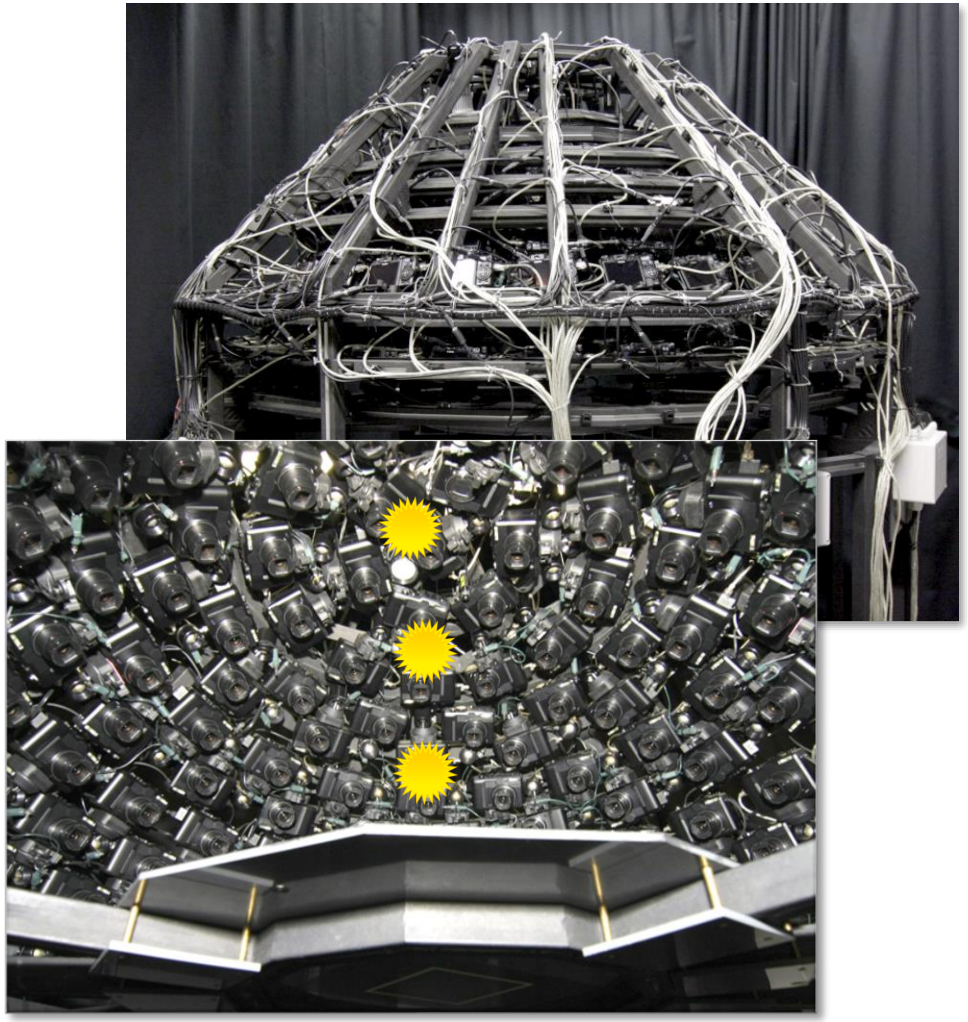
TU Delft

Material Appearance Capture

- Dome setup (highly-parallelized) from Müller et al. 2004:

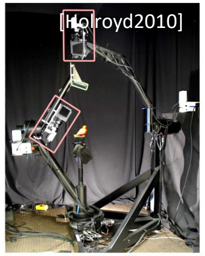


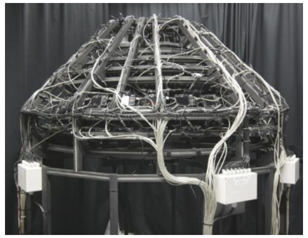




Material Appearance Capture

› Typical setups:

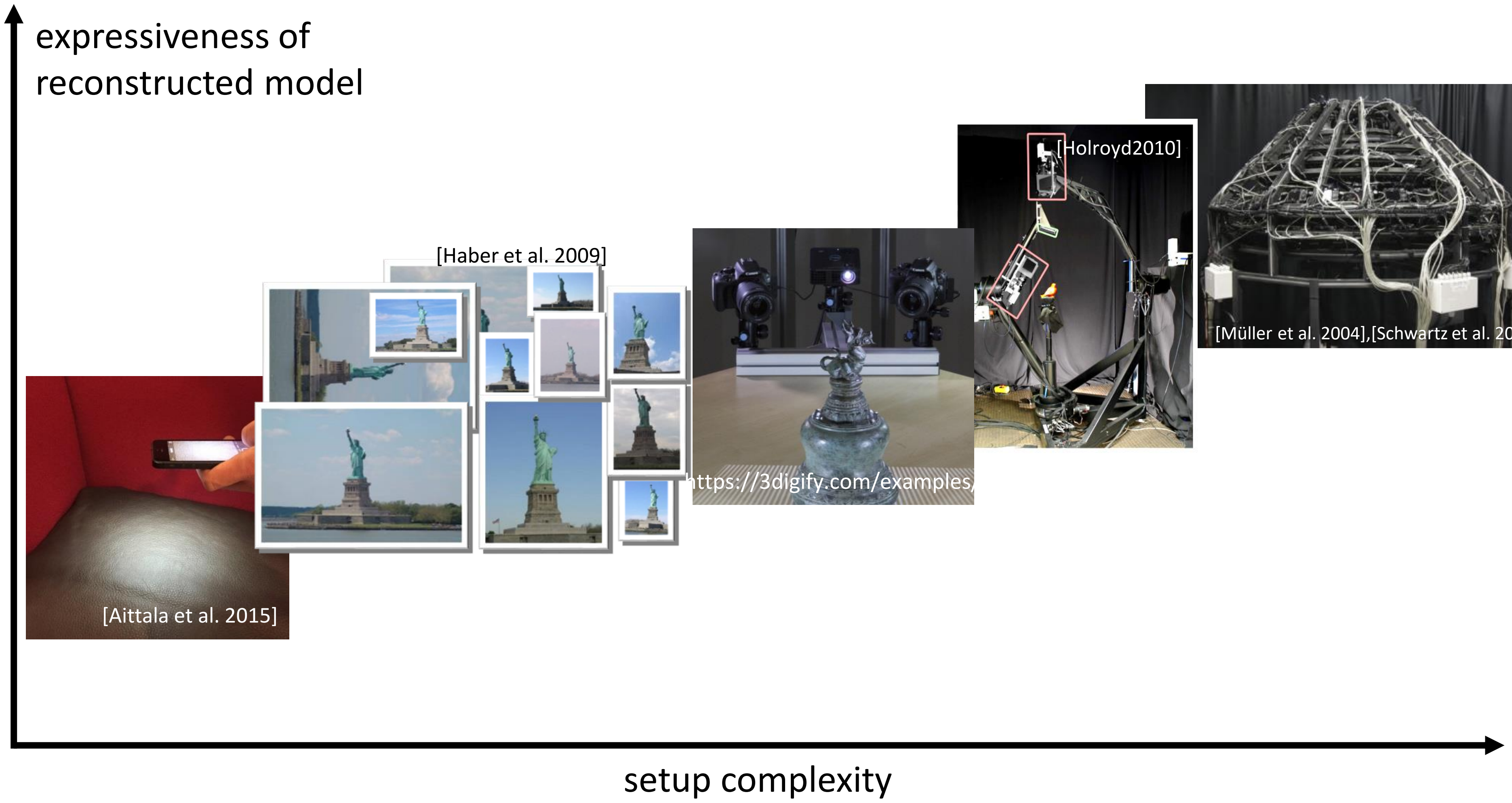




	Setups of individual camera and light source (manual)	gonioreflectometers	camera/light source arrays
measurement process	sequential	sequential	parallel
measurement times	long	long	short
level of automation	manual	automated	automated
costs	low to mid-level	expensive	very expensive

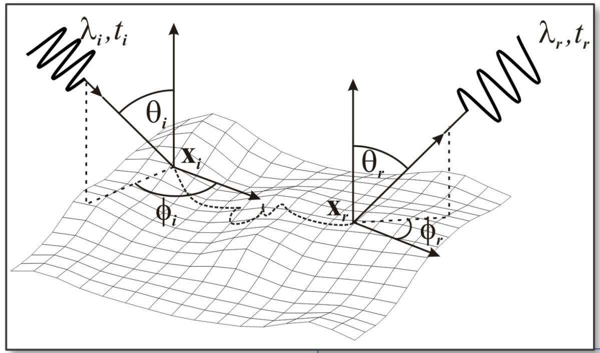
Material Appearance Capture

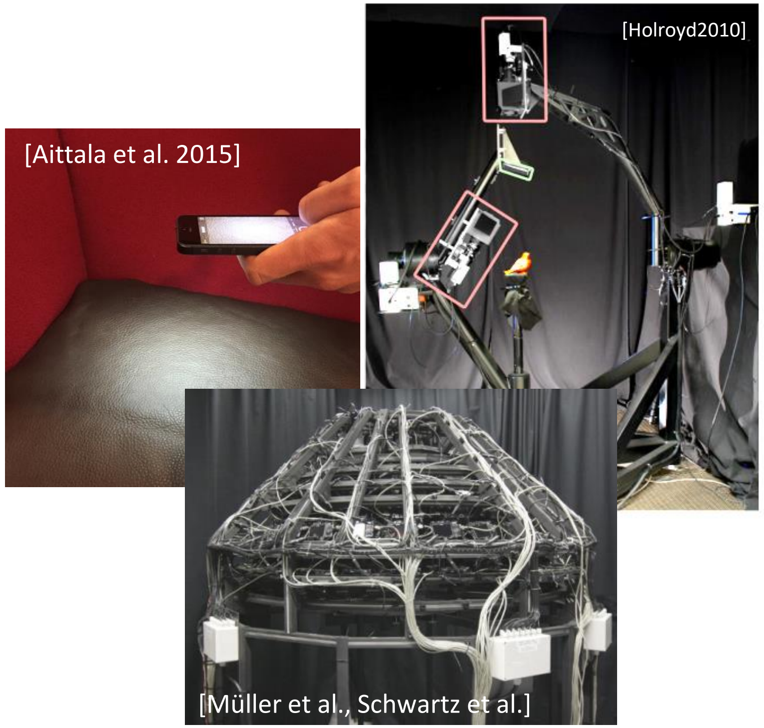
- Trade-off: Acquisition technology vs. expressiveness of models

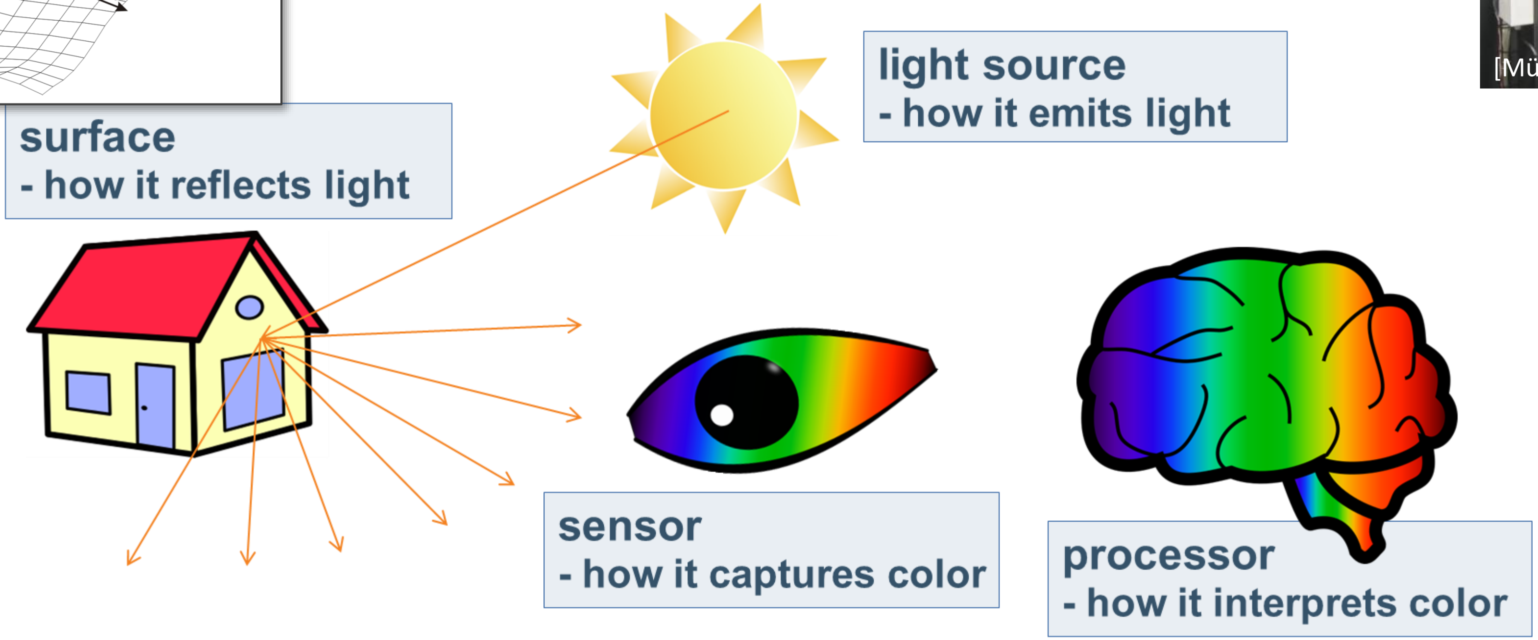


The story so far …

- We have seen:
  - Key factors of material appearance
  - Design of an appearance capture setups

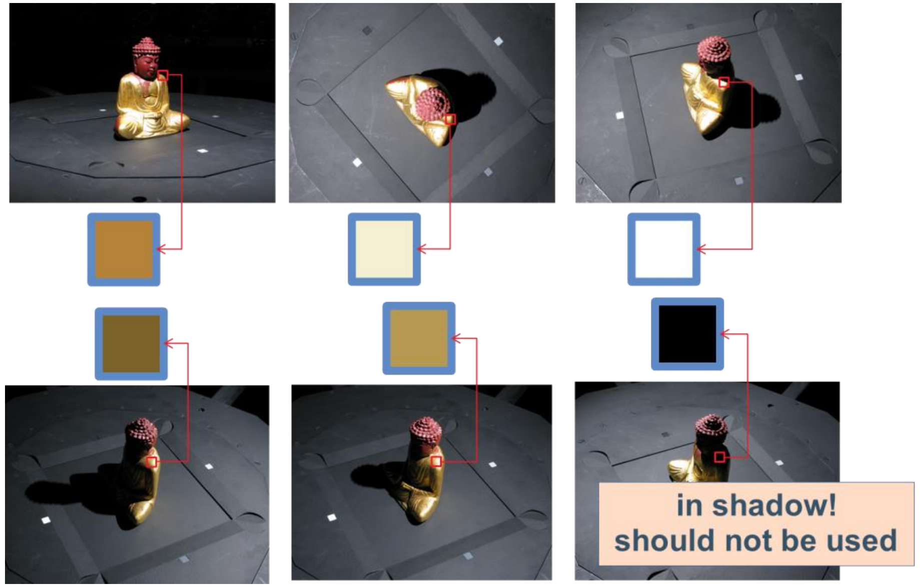




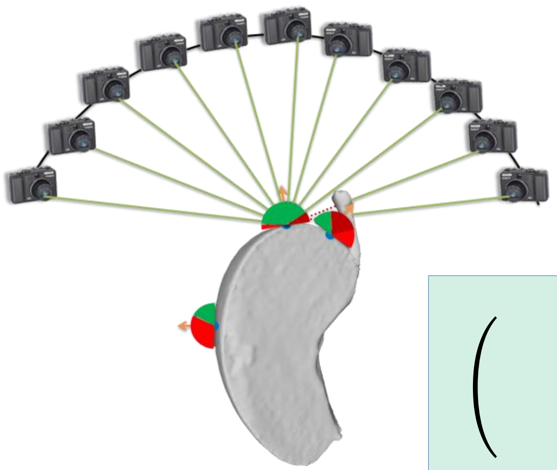


Next …

- Given:
  - Set of appearance measurements



- Task:
  - How to derive digital material representations from such measurements?

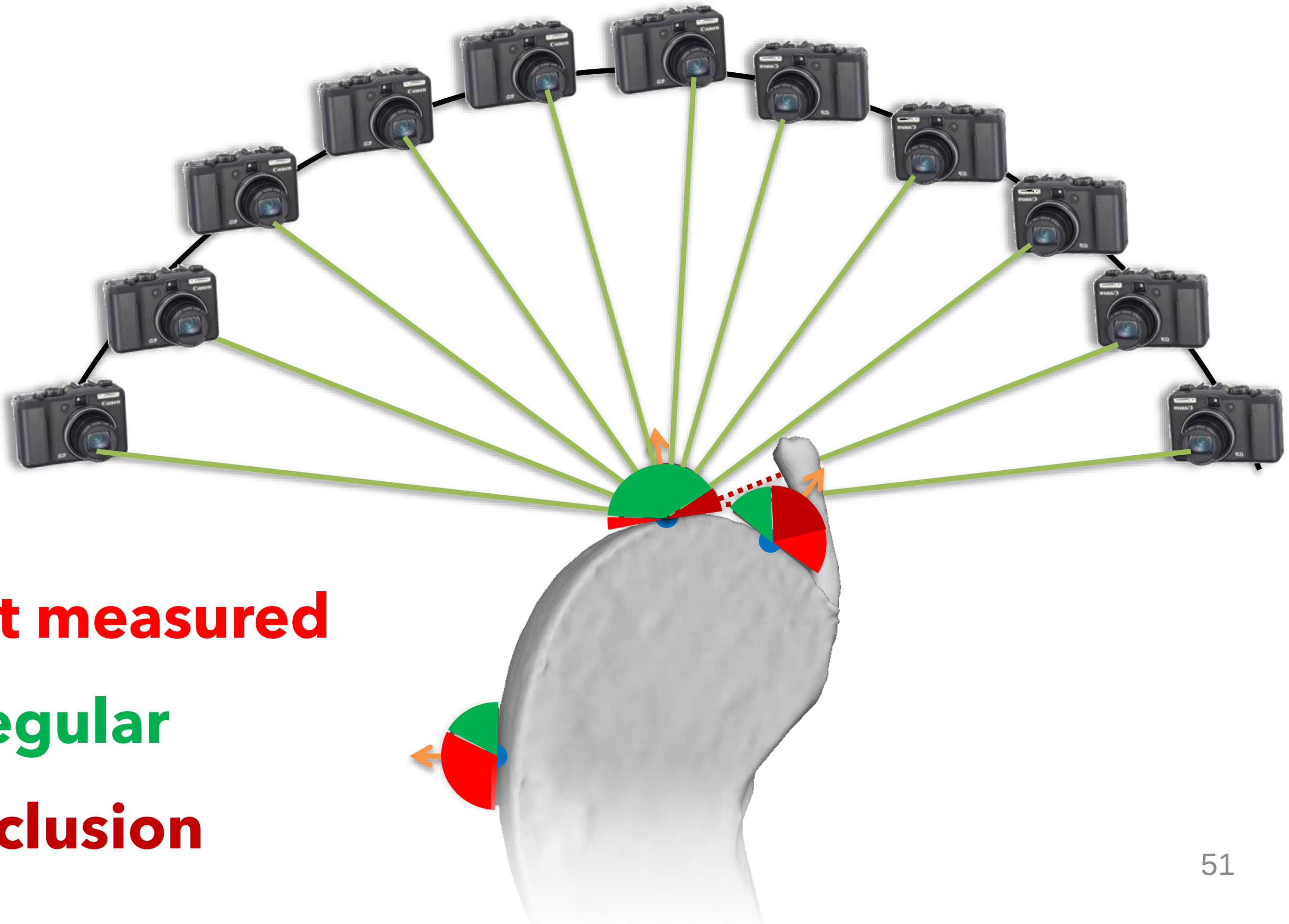


$$\left(A^T A \right)_{6\times6} \left(\boldsymbol{x} \right)_{6\times1} = \left(A^T \boldsymbol{b} \right)_{6\times1}$$

From Measurements …
… to Material Models

How to represent material appearance?

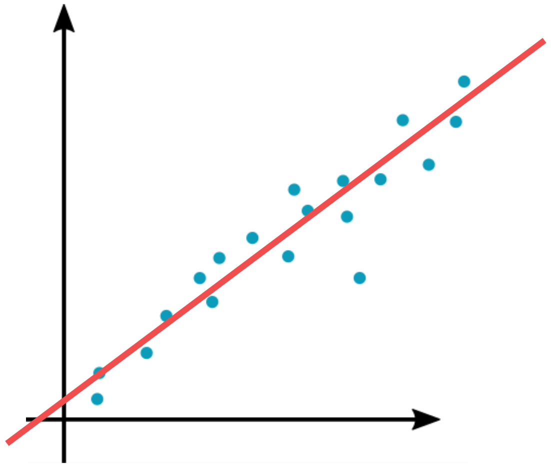
- Idea:
  - Measure appearance of surface points under different view-light conditions



- **Not measured**
- **Irregular**
- **Occlusion**

How to represent material appearance?

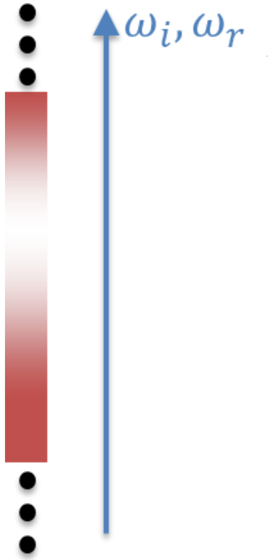
- **Trade-off**
  - How many measurements are needed to capture details?
  - Available memory resources?
  - Can we fill the gaps in between measurements?



Parametric models:
- more compact representation
- analytical model → few parameters

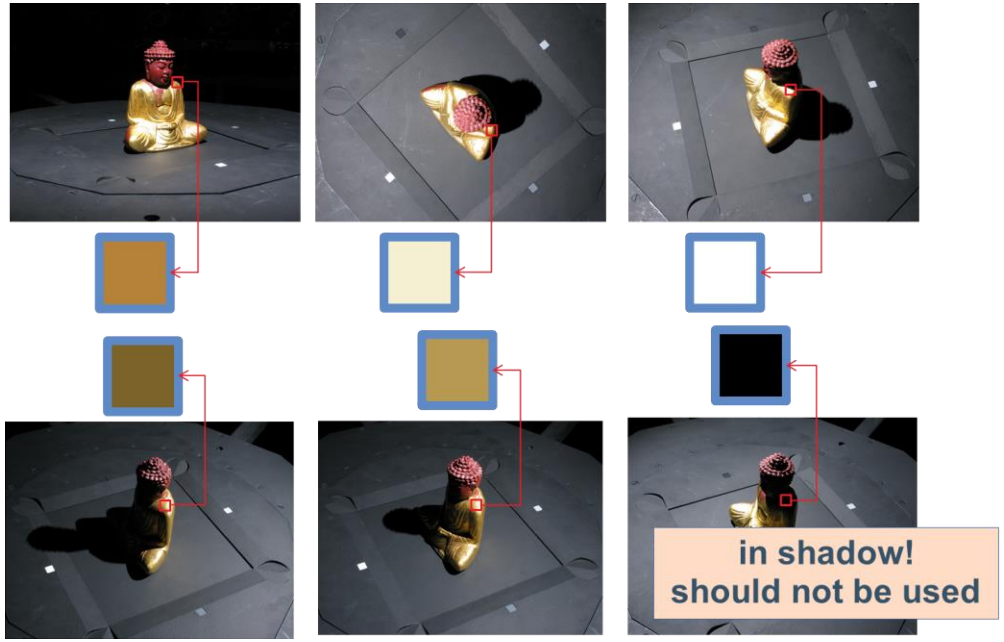
vs.

Dense measurements:
- very large raw data (terabytes!)
- hard to use in many applications

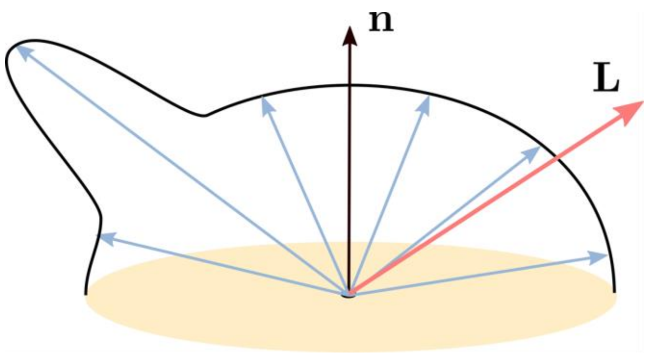


Parametric Reflectance Models

› Example: Mix of diffuse and specular reflectance behavior



?

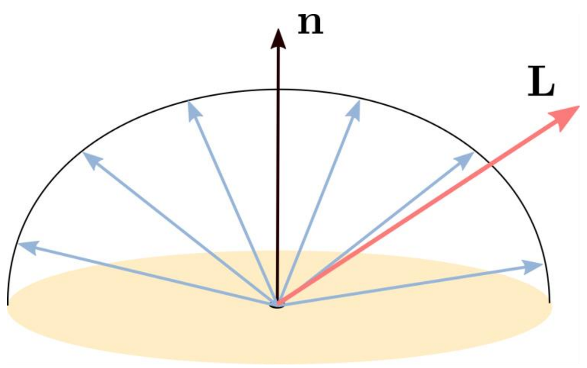


Parametric Reflectance Models

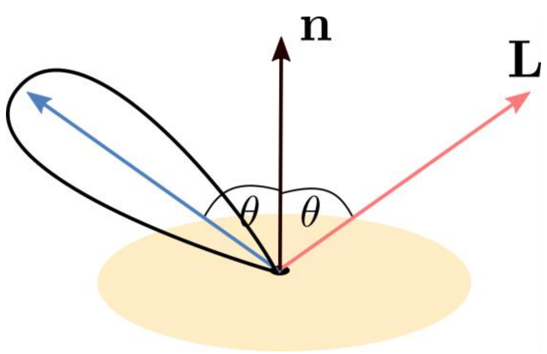
- Phong model

note:
we are now using **L** (direction to light) instead of $\mathbf{v}_i$ (incoming light direction)

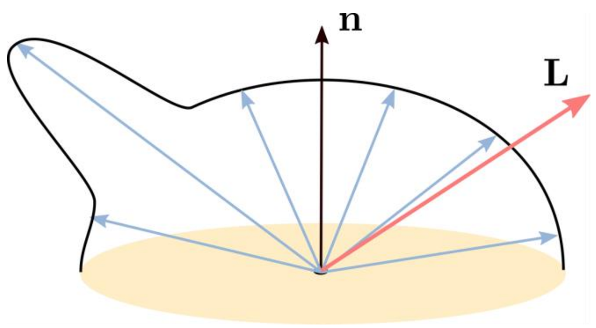
diffuse **specular**

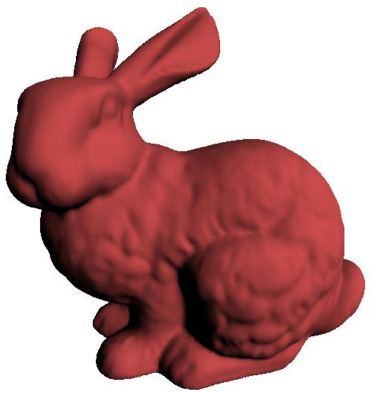


+



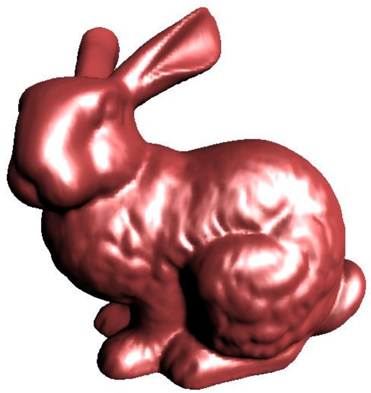
=





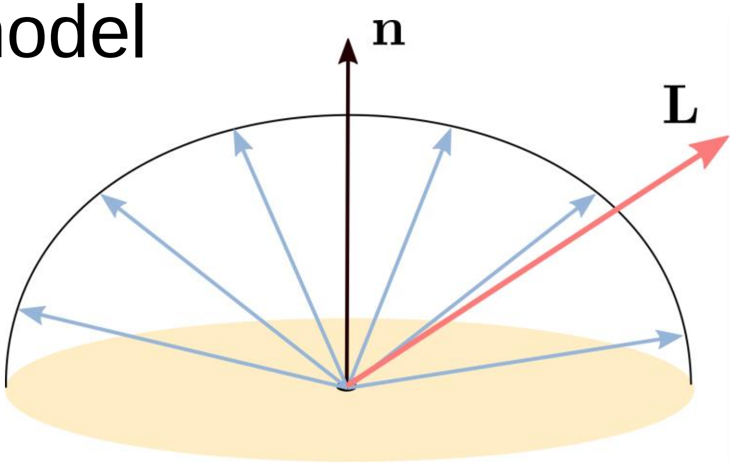
+

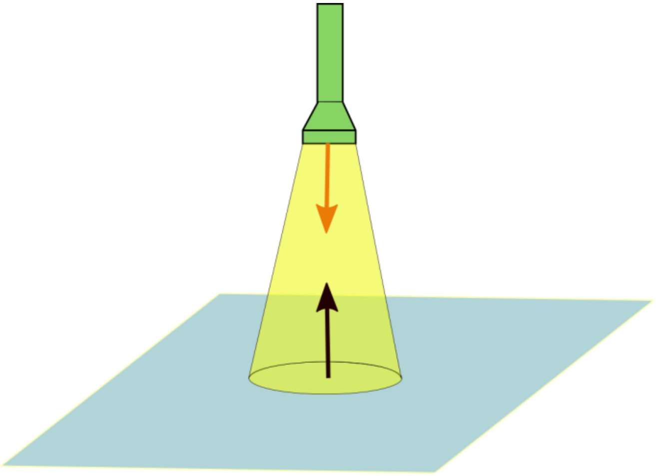
=

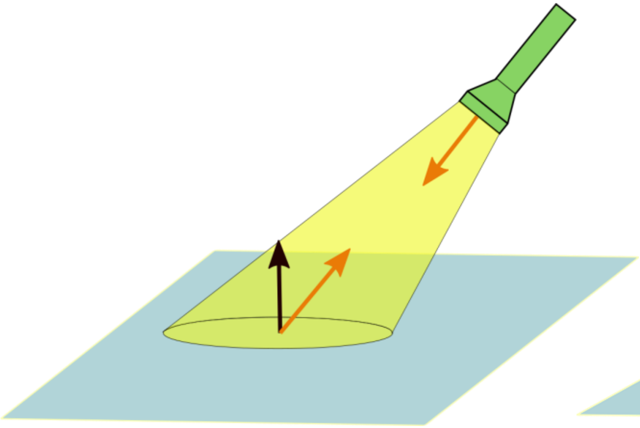


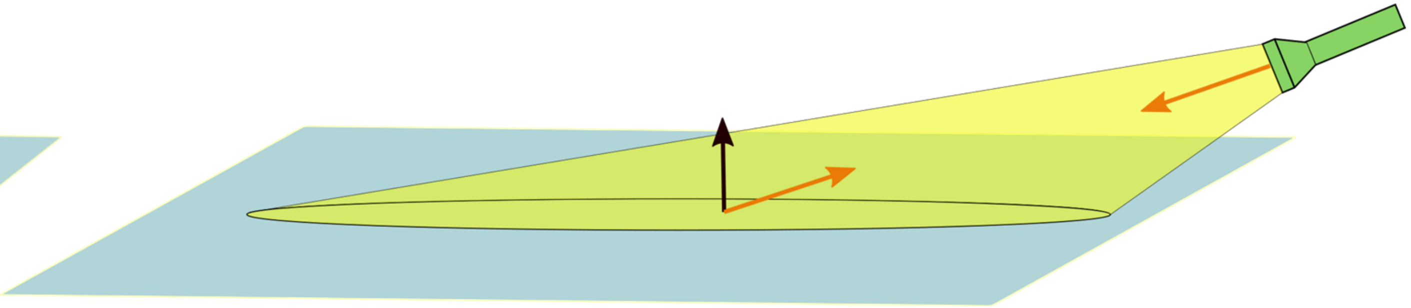
Parametric Reflectance Models

- Phong model







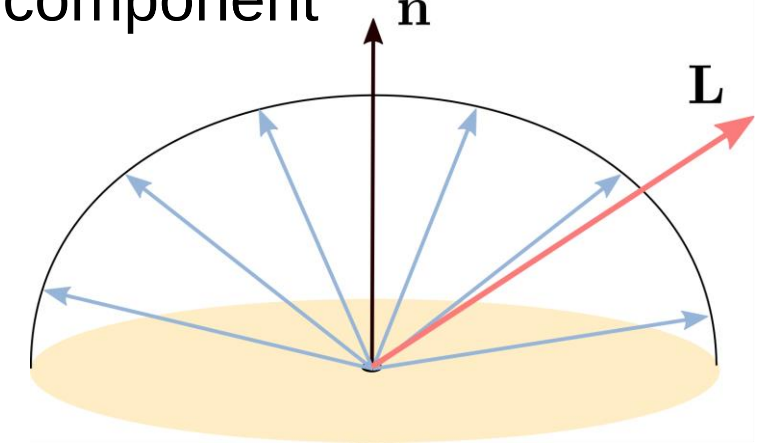


intensity per area decreases with angle between normal and light direction

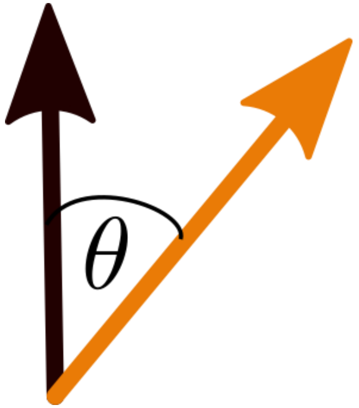
TU Delft

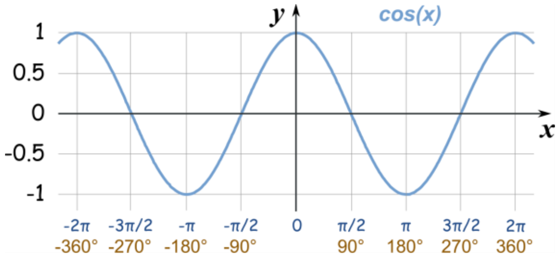
Parametric Reflectance Models

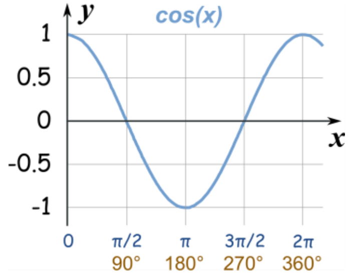
- Phong model
  - Diffuse component

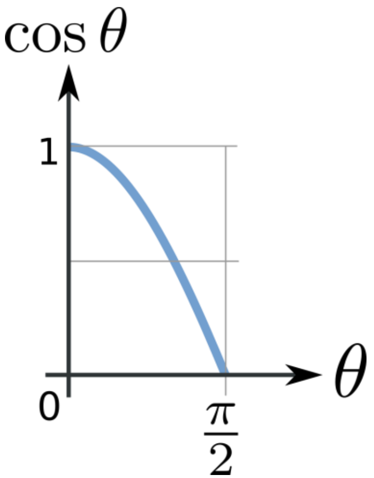


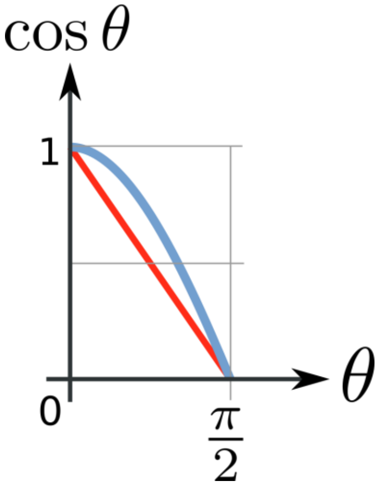
intensity decreases with cosine











Parametric Reflectance Models

› Recap: angle between vectors

dot product!!

$$\mathbf{v} \cdot \mathbf{w} = |\mathbf{v}||\mathbf{w}| \cos\theta$$

$$\mathbf{v} \cdot \mathbf{w} = v_x \cdot w_x + v_y \cdot w_y$$

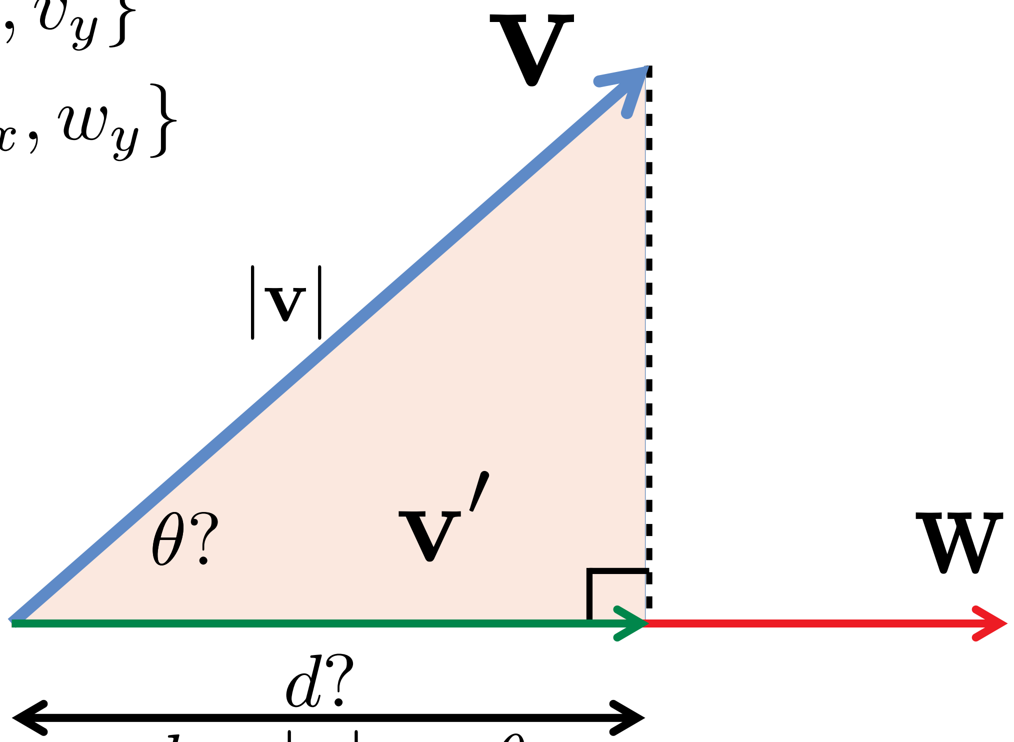
retrieve angle

$$\cos\theta = \frac{\mathbf{v} \cdot \mathbf{w}}{|\mathbf{v}||\mathbf{w}|}$$

projection

$$d = \frac{\mathbf{v} \cdot \mathbf{w}}{|\mathbf{w}|}$$

$$\mathbf{v} = \{v_x, v_y\}$$
$$\mathbf{w} = \{w_x, w_y\}$$

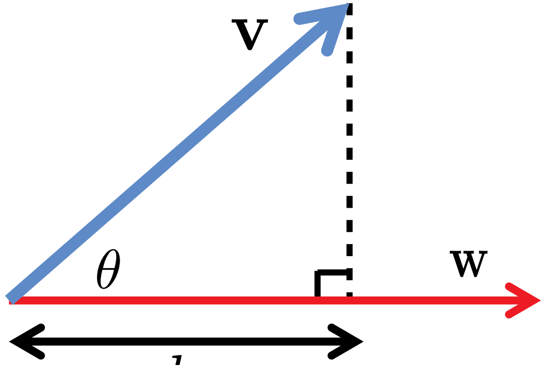


$$d = |\mathbf{v}| \cos\theta$$

$$\mathbf{v}' = d\frac{\mathbf{w}}{|\mathbf{w}|}$$

Parametric Reflectance Models

› Recap: cosine between vectors

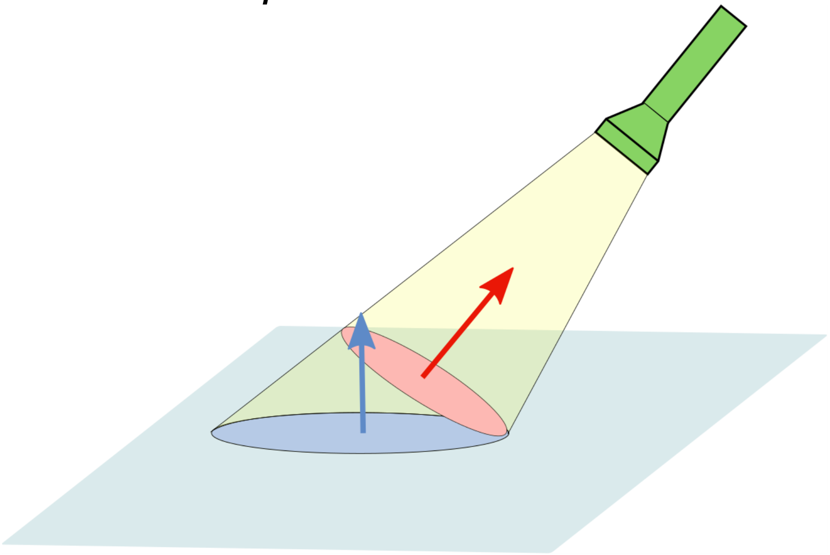


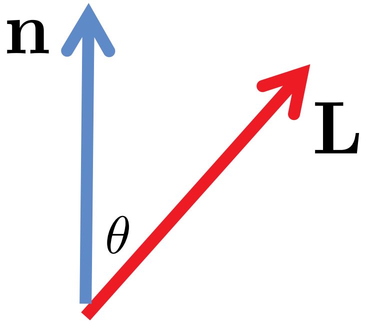
cosine

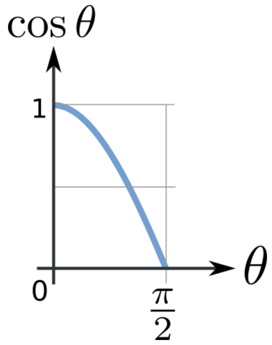
$$\cos\theta = \frac{\mathbf{v}\cdot\mathbf{w}}{|\mathbf{v}||\mathbf{w}|}$$

projection

$$d = |\mathbf{v}|\cos\theta = \frac{\mathbf{v}\cdot\mathbf{w}}{|\mathbf{w}|}$$







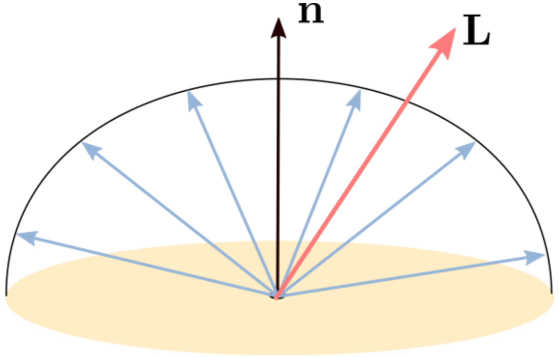
$$\textbf{diffuse dim factor} = \cos\theta$$

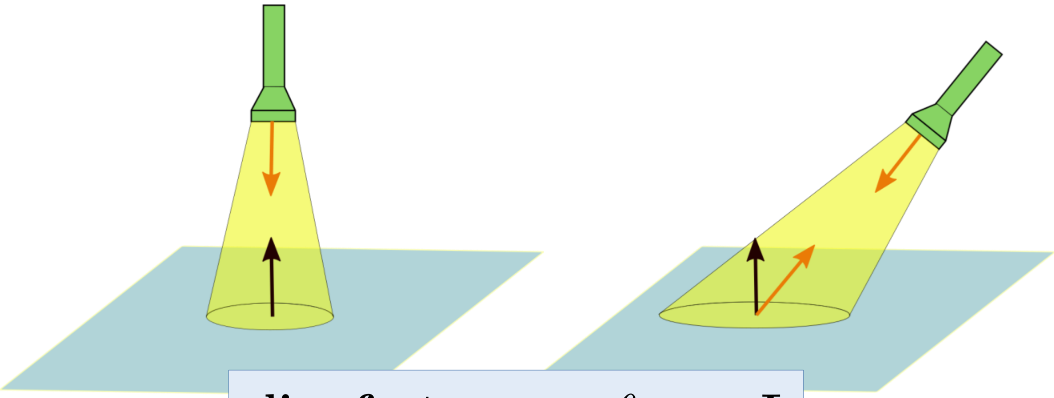
$$\cos\theta = \mathbf{n}\cdot\mathbf{L}$$

($\mathbf{n}$ and $\mathbf{L}$ are normalized)

Parametric Reflectance Models

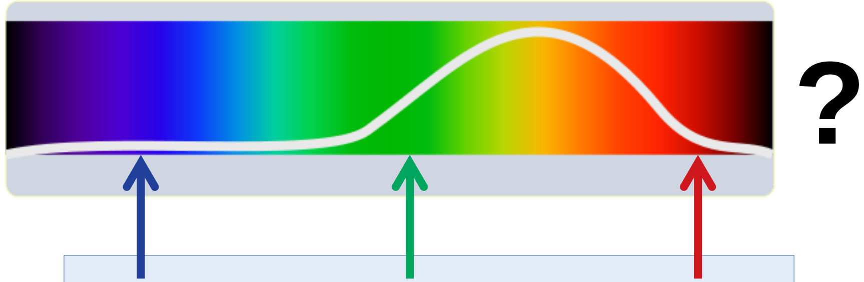
- Phong model
  - Diffuse component





$\mathbf{dim\ factor} = \cos\theta = \mathbf{n} \cdot \mathbf{L}$

($\mathbf{n}$ and $\mathbf{L}$ are normalized)



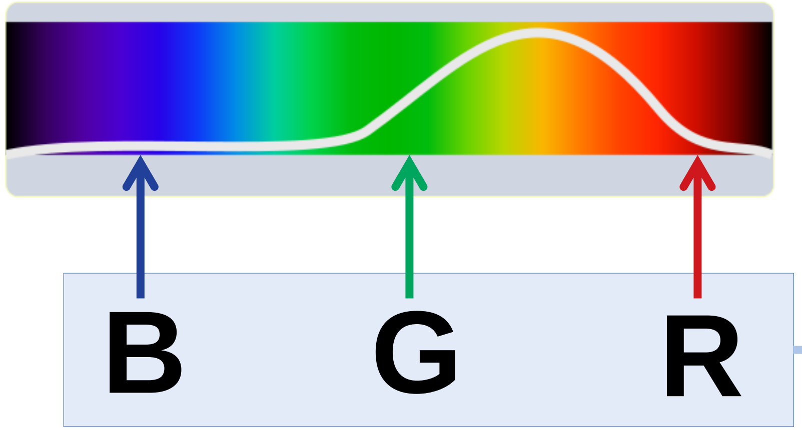
B G R

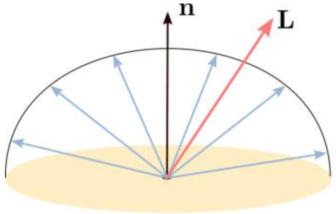
diffuse coefficient

$k_d = (R_d, G_d, B_d)$

Parametric Reflectance Models

- Phong model
  - Diffuse component





example

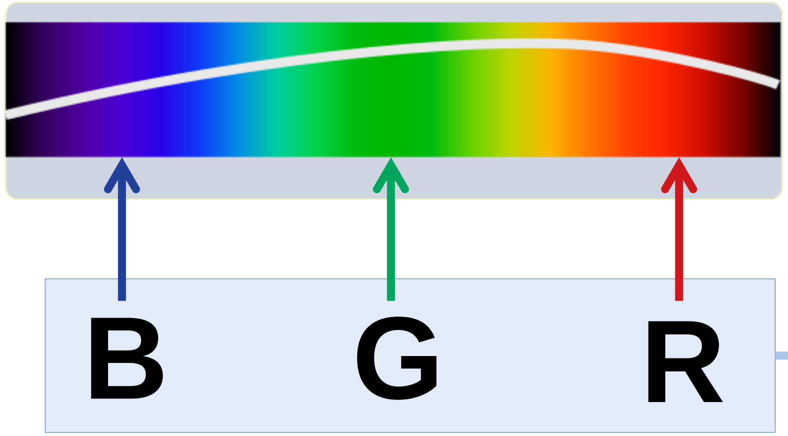
$$\boldsymbol{k}_d = (0.3, 0.2, 0.8)$$

$$I = (0.8, 0.8, 0.6)$$

$$\boldsymbol{k}_d\, I = (0.24, 0.16, 0.48)$$

diffuse coefficient

$$\boldsymbol{k}_d = (R_d, G_d, B_d)$$



light spectrum

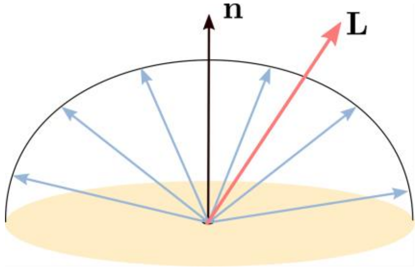
$$I = (R_L, G_L, B_L)$$

reflected spectrum

$$\boldsymbol{k}_d\, I = (R_L R_d, G_L G_d, B_L B_d)$$

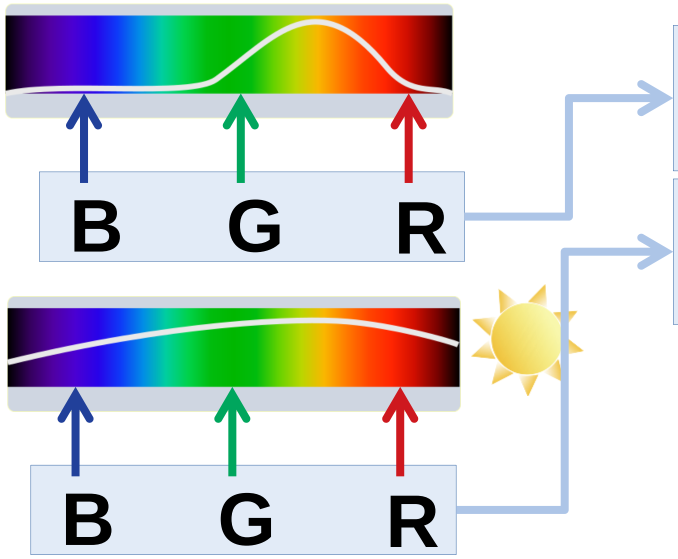
Parametric Reflectance Models

- Phong model
  - Diffuse component



$\textbf{dim factor} = \cos\theta = \mathbf{n} \cdot \mathbf{L}$

($\mathbf{n}$ and $\mathbf{L}$ are normalized)



diffuse coefficient

$\boldsymbol{k}_d = (R_d, G_d, B_d)$

light spectrum

$I = (R_L, G_L, B_L)$

reflected spectrum

$\boldsymbol{k}_d\, I = (R_L R_d, G_L G_d, B_L B_d)$

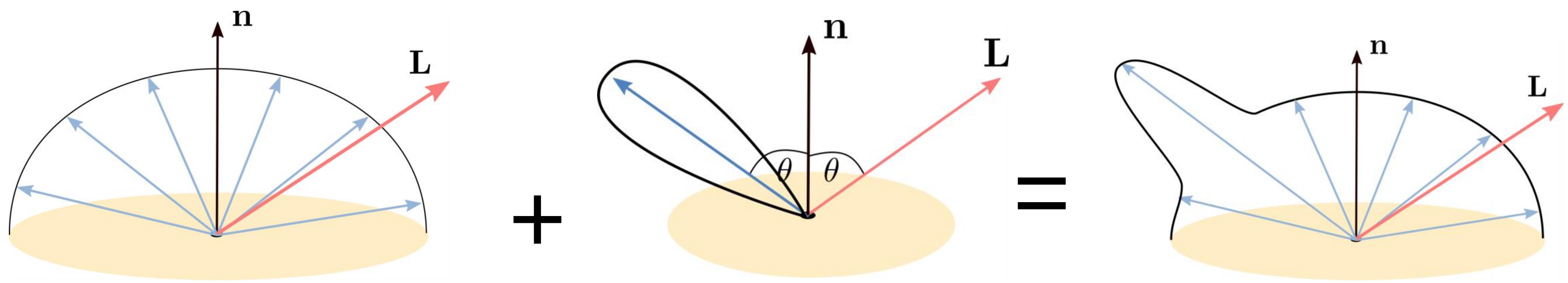
$$\textbf{diffuse} = \mathbf{k}_d \mathrm{I}(\mathbf{n} \cdot \mathbf{L})$$

Parametric Reflectance Models

- Phong model

diffuse

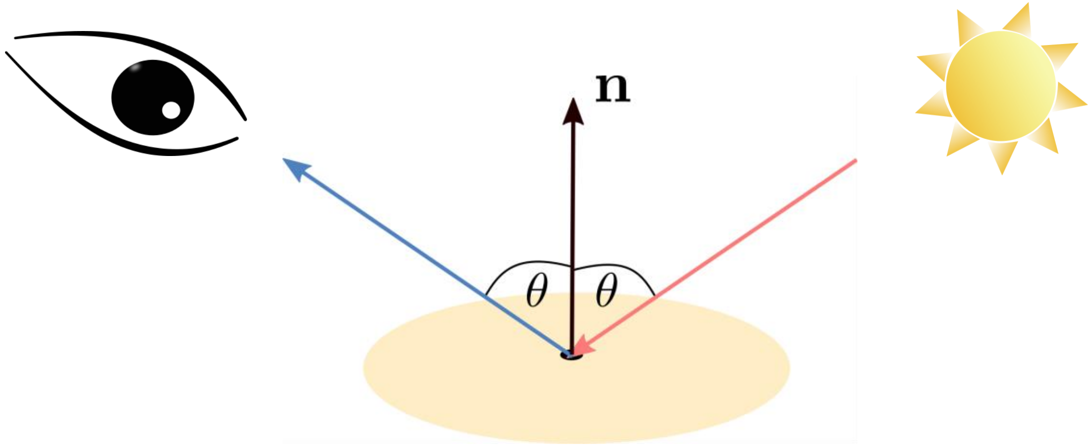
specular

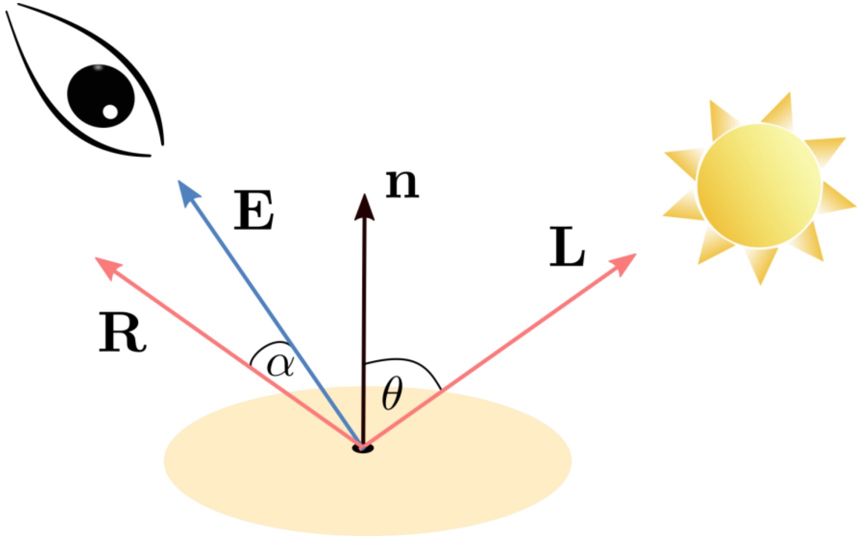


$$\mathbf{diffuse} = \mathbf{k}_d \mathrm{I}(\mathbf{n} \cdot \mathbf{L})$$

Parametric Reflectance Models

- Phong model
  - Specular component

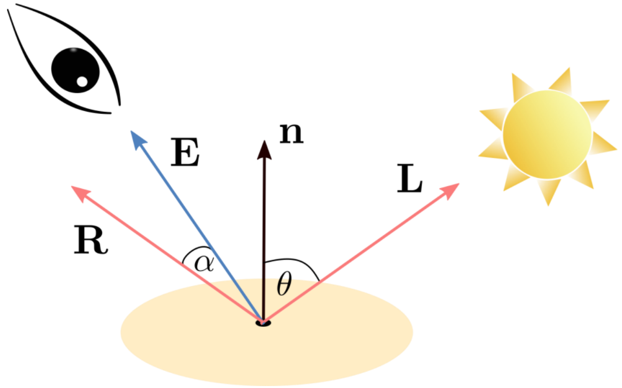


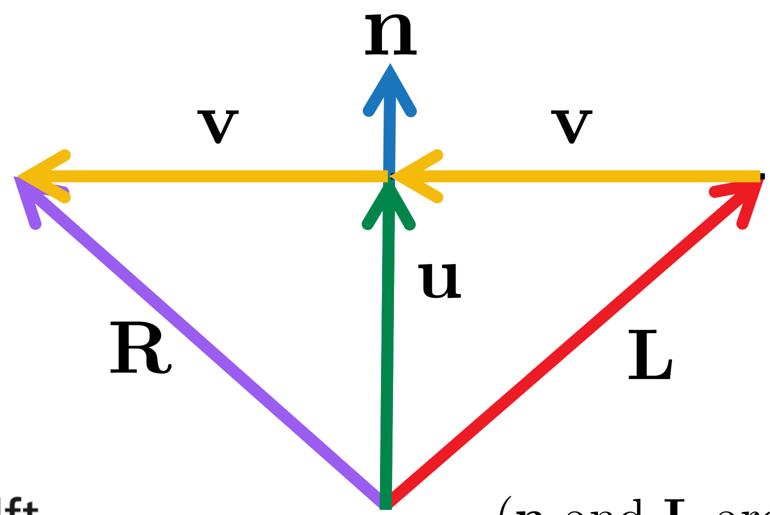


maximum intensity when eye is at reflected light vector direction

Parametric Reflectance Models

- Phong model
  - Specular component





how to compute R?

$$\mathbf{u} = (\mathbf{n} \cdot \mathbf{L})\mathbf{n}$$
$$\mathbf{v} = \mathbf{u} - \mathbf{L}$$
$$\mathbf{R} = \mathbf{L} + 2\mathbf{v}$$

substituting back:

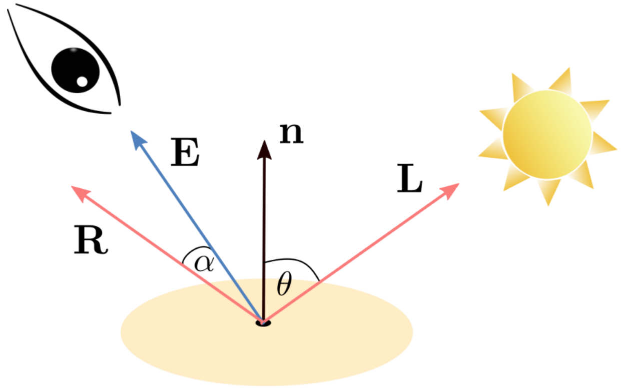
$$\mathbf{R} = \mathbf{L} + 2(\mathbf{u} - \mathbf{L})$$
$$\mathbf{R} = 2\mathbf{u} - \mathbf{L}$$

$$\boxed{\mathbf{R} = 2(\mathbf{n} \cdot \mathbf{L})\mathbf{n} - \mathbf{L}}$$

($\mathbf{n}$ and $\mathbf{L}$ are normalized)

Parametric Reflectance Models

- Phong model
  - Specular component

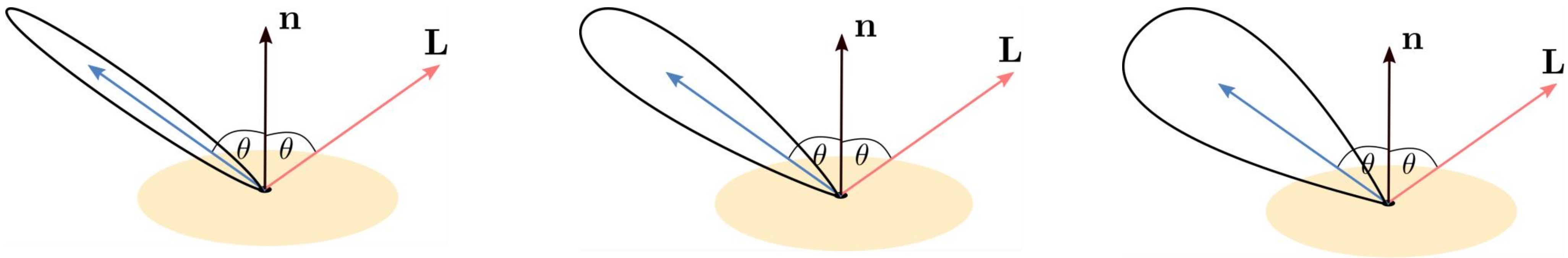


$$\textbf{specular dim factor} = \cos\alpha$$

$$\cos\alpha = \mathbf{E}\cdot\mathbf{R}$$

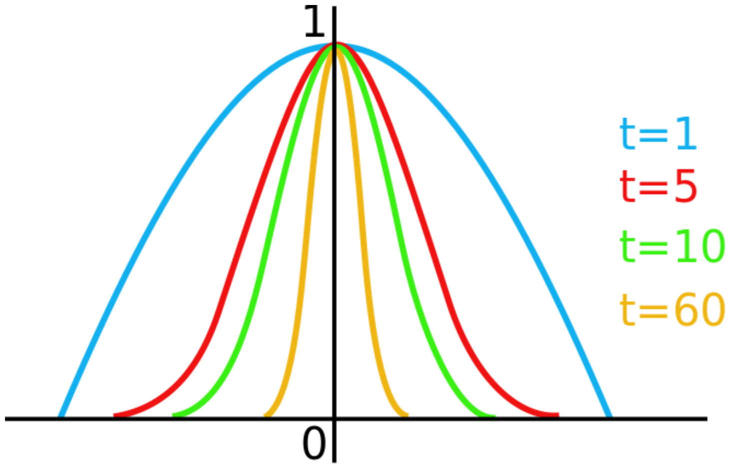
($\mathbf{E}$ and $\mathbf{R}$ are normalized)

how do we control the lobe?



Parametric Reflectance Models

- Phong model
  - Specular component

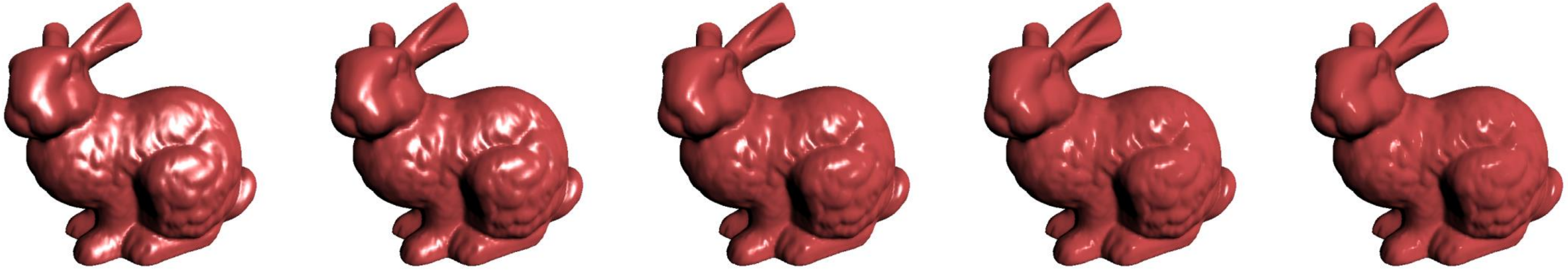


shininess parameter t

$$\textbf{specular dim factor} = (\cos\alpha)^t$$

$$(\cos\alpha)^t = (\mathbf{E}\cdot\mathbf{R})^t$$

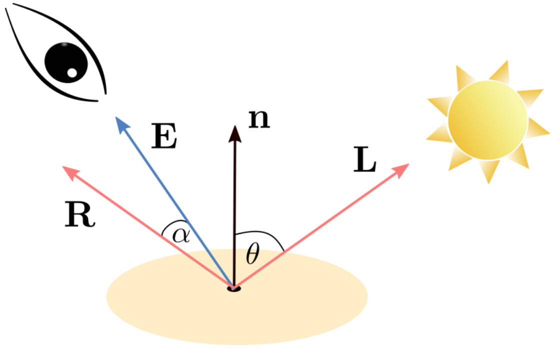
($\mathbf{E}$ and $\mathbf{R}$ are normalized)



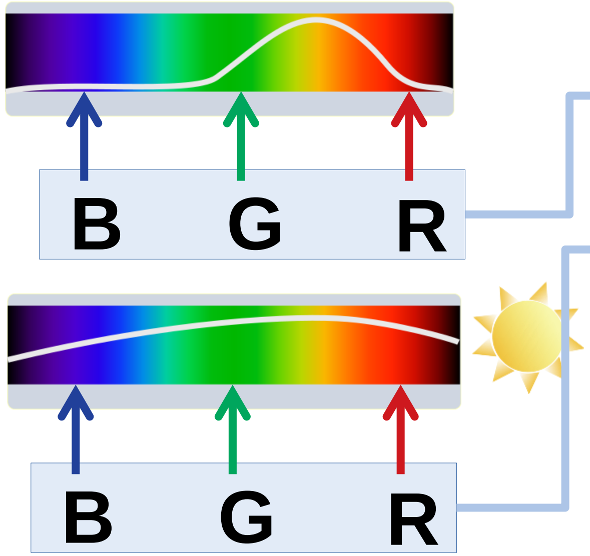
t=8 t=15 t=30 t=60 t=100

Parametric Reflectance Models

- Phong model
  - Specular component



$$\textbf{specular dim factor} = (\cos\alpha)^t$$
$$\cos\alpha = \mathbf{E}\cdot\mathbf{R}$$
($\mathbf{E}$ and $\mathbf{R}$ are normalized)



specular coefficient
$\boldsymbol{k}_s = (R_s, G_s, B_s)$

light spectrum
$I = (R_L, G_L, B_L)$

reflected spectrum
$\boldsymbol{k}_s\, I = (R_L R_s, G_L G_s, B_L B_s)$

$$\textbf{specular} = \mathbf{k}_s \mathrm{I}(\mathbf{E}\cdot\mathbf{R})^t$$

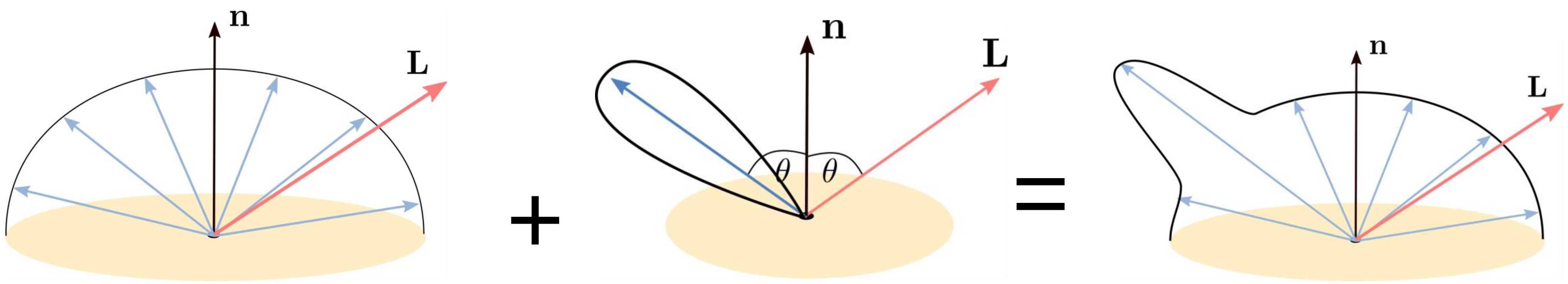
TU Delft

Parametric Reflectance Models

- Phong model

diffuse

specular



$$\mathbf{diffuse} = \mathbf{k}_d \mathrm{I}(\mathbf{n} \cdot \mathbf{L}) \qquad \mathbf{specular} = \mathbf{k}_s \mathrm{I}(\mathbf{E} \cdot \mathbf{R})^t$$

color at point

$$\mathrm{C} = \mathbf{k}_d \mathrm{I}(\mathbf{n} \cdot \mathbf{L}) + \mathbf{k}_s \mathrm{I}(\mathbf{E} \cdot \mathbf{R})^t$$

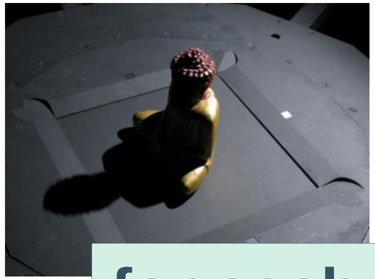
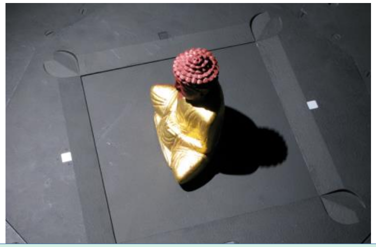
Fitting of Parametric Reflectance Model

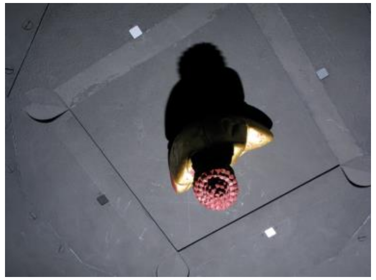
can we recover the Phong parameters from photos?

$$\mathbf{k}_d, \mathbf{k}_s, t \ ?$$

7 parameters

$$\mathrm{C} = \mathbf{k}_d \mathrm{I}(\mathbf{n} \cdot \mathbf{L}) + \mathbf{k}_s \mathrm{I}(\mathbf{E} \cdot \mathbf{R})^t$$

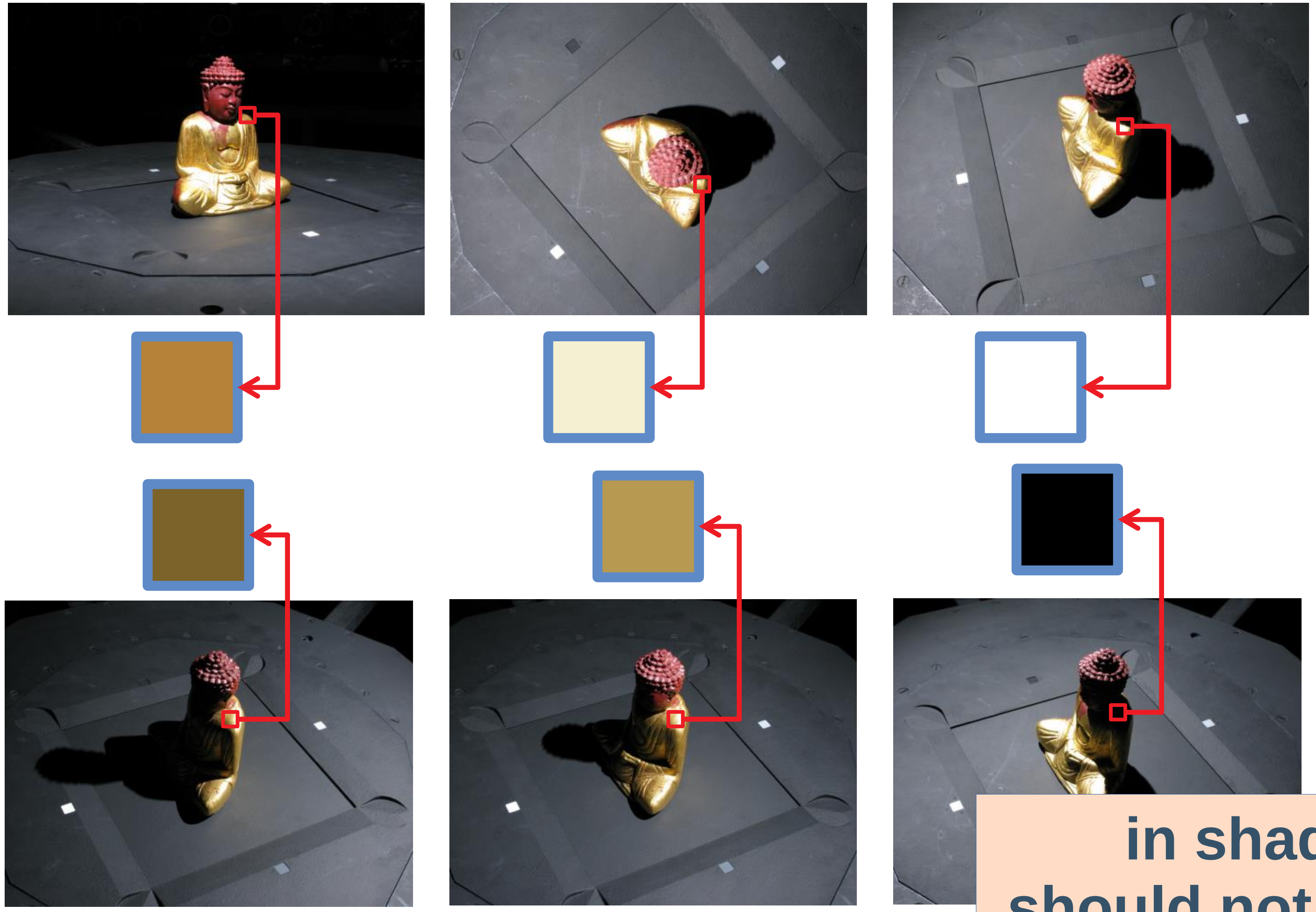


. . .

for each photo:
- color at each pixel
- light direction and intensity
- 3D model (geometry with normals)
- calibrated camera

Fitting of Parametric Reflectance Model

› Data from images: Appearance of surface points under different conditions



in shadow!
should not be used

Fitting of Parametric Reflectance Model

› Data from images

multiple information per 3D point

$$= \mathbf{k}_d \mathrm{I}_1 (\mathbf{n} \cdot \mathbf{L}_1) + \mathbf{k}_s \mathrm{I}_1 (\mathbf{E}_1 \cdot \mathbf{R}_1)^t$$

$$= \mathbf{k}_d \mathrm{I}_2 (\mathbf{n} \cdot \mathbf{L}_2) + \mathbf{k}_s \mathrm{I}_2 (\mathbf{E}_2 \cdot \mathbf{R}_2)^t$$

$$= \mathbf{k}_d \mathrm{I}_3 (\mathbf{n} \cdot \mathbf{L}_3) + \mathbf{k}_s \mathrm{I}_3 (\mathbf{E}_3 \cdot \mathbf{R}_3)^t$$

$$= \mathbf{k}_d \mathrm{I}_4 (\mathbf{n} \cdot \mathbf{L}_4) + \mathbf{k}_s \mathrm{I}_4 (\mathbf{E}_4 \cdot \mathbf{R}_4)^t$$

$$= \mathbf{k}_d \mathrm{I}_n (\mathbf{n} \cdot \mathbf{L}_n) + \mathbf{k}_s \mathrm{I}_n (\mathbf{E}_n \cdot \mathbf{R}_n)^t$$

unknowns

constant per point

change per photo

for each pixel we have:
- light information
- eye vector
- normal direction at 3D point

we want to estimate:
- diffuse coefficient (k_d)
- specular coefficient (k_s)
- shininess coefficient (t)

Fitting of Parametric Reflectance Model

- Optimization:

$$\mathrm{C}_1 = \mathbf{k}_d \mathrm{I}_1 (\mathbf{n} \cdot \mathbf{L}_1) + \mathbf{k}_s \mathrm{I}_1 (\mathbf{E}_1 \cdot \mathbf{R}_1)^t$$

photo 1

$$\mathrm{C}_2 = \mathbf{k}_d \mathrm{I}_2 (\mathbf{n} \cdot \mathbf{L}_2) + \mathbf{k}_s \mathrm{I}_2 (\mathbf{E}_2 \cdot \mathbf{R}_2)^t$$

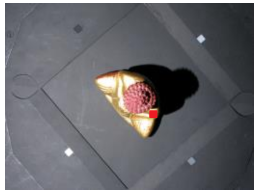


photo 2

$$\mathrm{C}_3 = \mathbf{k}_d \mathrm{I}_3 (\mathbf{n} \cdot \mathbf{L}_3) + \mathbf{k}_s \mathrm{I}_3 (\mathbf{E}_3 \cdot \mathbf{R}_3)^t$$

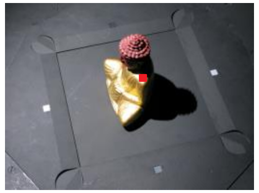


photo 3

$$\mathrm{C}_4 = \mathbf{k}_d \mathrm{I}_4 (\mathbf{n} \cdot \mathbf{L}_4) + \mathbf{k}_s \mathrm{I}_4 (\mathbf{E}_4 \cdot \mathbf{R}_4)^t$$

photo 4

$$\vdots \qquad \vdots \qquad \vdots \qquad \vdots \qquad \vdots$$

$$\mathrm{C}_n = \mathbf{k}_d \mathrm{I}_n (\mathbf{n} \cdot \mathbf{L}_n) + \mathbf{k}_s \mathrm{I}_n (\mathbf{E}_n \cdot \mathbf{R}_n)^t$$

photo n

Fitting of Parametric Reflectance Model

$$C_1 = \mathbf{k}_d I_1(\mathbf{n} \cdot \mathbf{L}_1) + \mathbf{k}_s I_1(\mathbf{E}_1 \cdot \mathbf{R}_1)^{\boxed{t}}$$

photo 1

trouble: cannot design a linear optimization

consider *t* given for now

$$\boxed{C_1} = \boxed{\mathbf{k}_d}\,\boxed{I_1}(\mathbf{n} \cdot \mathbf{L}_1) + \boxed{\mathbf{k}_s}\,\boxed{I_1}(\mathbf{E}_1 \cdot \mathbf{R}_1)^t$$

RGB values

red channel	$C_{r,1} = \mathbf{k}_{r,d} I_{r,1}(\mathbf{n} \cdot \mathbf{L}_1) + \mathbf{k}_{r,s} I_{r,1}(\mathbf{E}_1 \cdot \mathbf{R}_1)^t$
green channel	$C_{g,1} = \mathbf{k}_{g,d} I_{g,1}(\mathbf{n} \cdot \mathbf{L}_1) + \mathbf{k}_{g,s} I_{g,1}(\mathbf{E}_1 \cdot \mathbf{R}_1)^t$
blue channel	$C_{b,1} = \mathbf{k}_{b,d} I_{b,1}(\mathbf{n} \cdot \mathbf{L}_1) + \mathbf{k}_{b,s} I_{b,1}(\mathbf{E}_1 \cdot \mathbf{R}_1)^t$

Fitting of Parametric Reflectance Model

consider ***t*** given for now
and let's look at only one channel

$$C_{r,1} = \mathbf{k}_{r,d} I_{r,1} (\mathbf{n} \cdot \mathbf{L}_1) + \mathbf{k}_{r,s} I_{r,1} (\mathbf{E}_1 \cdot \mathbf{R}_1)^t$$

simplify notation for now, remove ***r***

$$C_1 = \mathbf{k}_d I_1 (\mathbf{n} \cdot \mathbf{L}_1) + \mathbf{k}_s I_1 (\mathbf{E}_1 \cdot \mathbf{R}_1)^t$$

matrix notation

$$\begin{pmatrix} I_1(\mathbf{n} \cdot \mathbf{L}_1) & I_1(\mathbf{E}_1 \cdot \mathbf{R}_1)^t \end{pmatrix} \begin{pmatrix} \mathbf{k}_d \\ \mathbf{k}_s \end{pmatrix} = C_1$$

insert second image in matrix

$$\begin{pmatrix} I_1(\mathbf{n} \cdot \mathbf{L}_1) & I_1(\mathbf{E}_1 \cdot \mathbf{R}_1)^t \\ I_2(\mathbf{n} \cdot \mathbf{L}_2) & I_2(\mathbf{E}_2 \cdot \mathbf{R}_2)^t \end{pmatrix} \begin{pmatrix} \mathbf{k}_d \\ \mathbf{k}_s \end{pmatrix} = \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}$$

photo 1

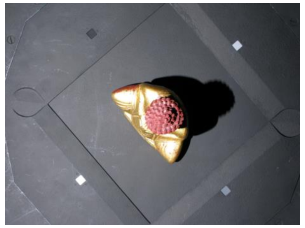


photo 2

Fitting of Parametric Reflectance Model

$$\begin{pmatrix} \mathrm{I}_1(\mathbf{n}\cdot\mathbf{L}_1) & \mathrm{I}_1(\mathbf{E}_1\cdot\mathbf{R}_1)^t \\ \mathrm{I}_2(\mathbf{n}\cdot\mathbf{L}_2) & \mathrm{I}_2(\mathbf{E}_2\cdot\mathbf{R}_2)^t \end{pmatrix} \begin{pmatrix} \mathbf{k}_d \\ \mathbf{k}_s \end{pmatrix} = \begin{pmatrix} \mathrm{C}_1 \\ \mathrm{C}_2 \end{pmatrix}$$

insert the other photos

$$\begin{pmatrix} \mathrm{I}_1(\mathbf{n}\cdot\mathbf{L}_1) & \mathrm{I}_1(\mathbf{E}_1\cdot\mathbf{R}_1)^t \\ \mathrm{I}_2(\mathbf{n}\cdot\mathbf{L}_2) & \mathrm{I}_2(\mathbf{E}_2\cdot\mathbf{R}_2)^t \\ \mathrm{I}_3(\mathbf{n}\cdot\mathbf{L}_3) & \mathrm{I}_3(\mathbf{E}_3\cdot\mathbf{R}_3)^t \\ \mathrm{I}_4(\mathbf{n}\cdot\mathbf{L}_4) & \mathrm{I}_4(\mathbf{E}_4\cdot\mathbf{R}_4)^t \\ \vdots & \vdots \\ \mathrm{I}_n(\mathbf{n}\cdot\mathbf{L}_n) & \mathrm{I}_n(\mathbf{E}_n\cdot\mathbf{R}_n)^t \end{pmatrix} \begin{pmatrix} \mathbf{k}_d \\ \mathbf{k}_s \end{pmatrix} = \begin{pmatrix} \mathrm{C}_1 \\ \mathrm{C}_2 \\ \mathrm{C}_3 \\ \mathrm{C}_4 \\ \vdots \\ \mathrm{C}_n \end{pmatrix}$$

photo 1

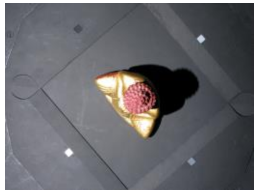
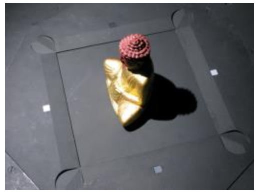


photo 2



⋮

photo n

Fitting of Parametric Reflectance Model

all photos and all channels

$$\begin{pmatrix} \text{diff}_{r,1} & \text{spec}_{r,1} & 0 & 0 & 0 & 0 \\ 0 & 0 & \text{diff}_{g,1} & \text{spec}_{g,1} & 0 & 0 \\ 0 & 0 & 0 & 0 & \text{diff}_{b,1} & \text{spec}_{b,1} \\ \text{diff}_{r,2} & \text{spec}_{r,2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \text{diff}_{g,2} & \text{spec}_{g,2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \text{diff}_{b,2} & \text{spec}_{b,2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \text{diff}_{r,n} & \text{spec}_{r,n} & 0 & 0 & 0 & 0 \\ 0 & 0 & \text{diff}_{g,n} & \text{spec}_{g,n} & 0 & 0 \\ 0 & 0 & 0 & 0 & \text{diff}_{b,n} & \text{spec}_{b,n} \end{pmatrix} \begin{pmatrix} \mathbf{k}_{r,d} \\ \mathbf{k}_{r,s} \\ \mathbf{k}_{g,d} \\ \mathbf{k}_{g,s} \\ \mathbf{k}_{b,d} \\ \mathbf{k}_{b,s} \end{pmatrix} = \begin{pmatrix} \mathrm{C}_{r,1} \\ \mathrm{C}_{g,1} \\ \mathrm{C}_{b,1} \\ \mathrm{C}_{r,2} \\ \mathrm{C}_{g,2} \\ \mathrm{C}_{b,2} \\ \vdots \\ \mathrm{C}_{r,n} \\ \mathrm{C}_{g,n} \\ \mathrm{C}_{b,n} \end{pmatrix}$$

$$\text{diff}_{r,i} = \mathrm{I}_{r,i}(\mathbf{n} \cdot \mathbf{L}_{r,i}) \qquad \text{spec}_{r,i} = \mathrm{I}_{r,i}(\mathbf{E}_{r,i} \cdot \mathbf{R}_{r,i})^t$$

Fitting of Parametric Reflectance Model

all photos and all channels

$$
\begin{pmatrix}
\mathrm{diff}_{r,1} & \mathrm{spec}_{r,1} & 0 & 0 & 0 & 0 \\
0 & 0 & \mathrm{diff}_{g,1} & \mathrm{spec}_{g,1} & 0 & 0 \\
0 & 0 & 0 & 0 & \mathrm{diff}_{b,1} & \mathrm{spec}_{b,1} \\
\mathrm{diff}_{r,2} & \mathrm{spec}_{r,2} & 0 & 0 & 0 & 0 \\
0 & 0 & \mathrm{diff}_{g,2} & \mathrm{spec}_{g,2} & 0 & 0 \\
0 & & & & & \\
\vdots & & & & & \\
\mathrm{diff}_{r,n} & \mathrm{spec}_{r,n} & 0 & 0 & 0 & 0 \\
0 & 0 & \mathrm{diff}_{g,n} & \mathrm{spec}_{g,n} & 0 & 0 \\
0 & 0 & 0 & 0 & \mathrm{diff}_{b,n} & \mathrm{spec}_{b,n}
\end{pmatrix}
\begin{pmatrix}
\mathbf{k}_{r,d} \\ \mathbf{k}_{r,s} \\ \mathbf{k}_{g,d} \\ \\ \mathbf{k}_{b,s}
\end{pmatrix}
=
\begin{pmatrix}
C_{r,1} \\ C_{g,1} \\ C_{b,1} \\ C_{r,2} \\ C_{g,2} \\ C_{b,2} \\ \vdots \\ C_{r,n} \\ C_{g,n} \\ C_{b,n}
\end{pmatrix}
$$

look carefully at this matrix after the lecture!

$$\mathrm{diff}_{r,i} = \mathrm{I}_{r,i}(\mathbf{n} \cdot \mathbf{L}_{r,i}) \qquad \mathrm{spec}_{r,i} = \mathrm{I}_{r,i}(\mathbf{E}_{r,i} \cdot \mathbf{R}_{r,i})^t$$

Fitting of Parametric Reflectance Model

› Linear system:

$$\begin{pmatrix} \text{diff}_{r,1} & \text{spec}_{r,1} & 0 & 0 & 0 & 0 \\ 0 & 0 & \text{diff}_{g,1} & \text{spec}_{g,1} & 0 & 0 \\ 0 & 0 & 0 & 0 & \text{diff}_{b,1} & \text{spec}_{b,1} \\ \text{diff}_{r,2} & \text{spec}_{r,2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \text{diff}_{g,2} & \text{spec}_{g,2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \text{diff}_{b,2} & \text{spec}_{b,2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \text{diff}_{r,n} & \text{spec}_{r,n} & 0 & 0 & 0 & 0 \\ 0 & 0 & \text{diff}_{g,n} & \text{spec}_{g,n} & 0 & 0 \\ 0 & 0 & 0 & 0 & \text{diff}_{b,n} & \text{spec}_{b,n} \end{pmatrix} \begin{pmatrix} \mathbf{k}_{r,d} \\ \mathbf{k}_{r,s} \\ \mathbf{k}_{g,d} \\ \mathbf{k}_{g,s} \\ \mathbf{k}_{b,d} \\ \mathbf{k}_{b,s} \end{pmatrix} = \begin{pmatrix} \mathrm{C}_{r,1} \\ \mathrm{C}_{g,1} \\ \mathrm{C}_{b,1} \\ \mathrm{C}_{r,2} \\ \mathrm{C}_{g,2} \\ \mathrm{C}_{b,2} \\ \vdots \\ \mathrm{C}_{r,n} \\ \mathrm{C}_{g,n} \\ \mathrm{C}_{b,n} \end{pmatrix}$$

$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

Fitting of Parametric Reflectance Model

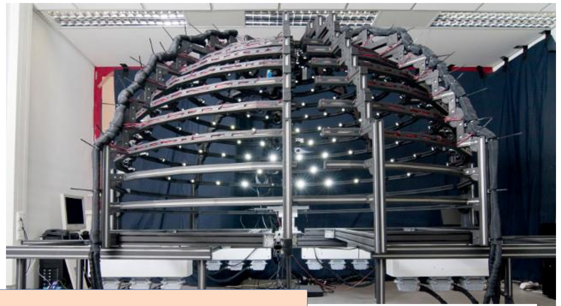
- Large linear system:

how do we solve this?
(how do we find x?)

$$\mathrm{A}\mathbf{x} = \mathbf{b}$$

invert A

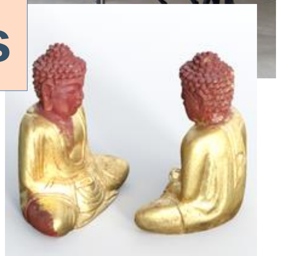
$$\mathbf{x} = \mathrm{A}^{-1}\mathbf{b}$$



size of the linear system

$$\mathrm{A}_{3n\times 6}\mathbf{x}_{6\times 1} = \mathbf{b}_{3n\times 1}$$

n = 151x151 ~ 23K photos



rewrite

$$\boxed{\mathrm{A}^T\mathrm{A}}\,\mathbf{x} = \boxed{\mathrm{A}^T\mathbf{b}}$$

$$\mathrm{A}^T_{6\times 3n}\mathbf{b}_{3n\times 1} = (\mathrm{A}^T\mathbf{b})_{6\times 1}$$

$$\mathrm{A}^T_{6\times 3n}\mathrm{A}_{3n\times 6} = (\mathrm{A}^T\mathrm{A})_{6\times 6}$$

new size of the linear system

$$\left(\quad A^T A \quad \right)_{6\times 6} \left(\boldsymbol{x} \right)_{6\times 1} = \left(A^T \boldsymbol{b} \right)_{6\times 1}$$

Fitting of Parametric Reflectance Model

› Optimization with shininess as parameter:

$$C_1 = k_d I_1 (\mathbf{n} \cdot \mathbf{L}_1) + k_s I_1 (\mathbf{E}_1 \cdot \mathbf{R}_1)^{\boxed{t}}$$

we considered *t* given!
what if we do not know *t*

now, consider k_d and k_s given

$$C_1 = k_d I_1 (\mathbf{n} \cdot \mathbf{L}_1) + k_s I_1 (\mathbf{E}_1 \cdot \mathbf{R}_1)^t$$

take the log of everything and solve for *t*

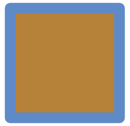
$$k_s I_1 (\mathbf{E}_1 \cdot \mathbf{R}_1)^t = C_1 - k_d I_1 (\mathbf{n} \cdot \mathbf{L}_1)$$

$$(\mathbf{E}_1 \cdot \mathbf{R}_1)^t = \frac{C_1 - k_d I_1 (\mathbf{n} \cdot \mathbf{L}_1)}{k_s I_1} \qquad t = \log_{(\mathbf{E}_1 \cdot \mathbf{R}_1)} \left(\frac{C_1 - k_d I_1 (\mathbf{n} \cdot \mathbf{L}_1)}{k_s I_1} \right)$$

Simple solution: compute the value of *t* for each point and use the average value.
This is left as exercise and is part of the assignment

Fitting of Parametric Reflectance Model

- Full optimization:



$$C_1 = \boxed{\mathbf{k}_d}\mathrm{I}_1(\mathbf{n}\cdot\mathbf{L}_1) + \boxed{\mathbf{k}_s}\mathrm{I}_1(\mathbf{E}_1\cdot\mathbf{R}_1)^{\boxed{t}}$$

what if we really do not know any parameter?

simple strategy:

1) guess *t*
2) with *t*, solve for k_d and k_s
3) with k_d and k_s, solve for *t*
4) repeat 2 and 3 until convergence

Fitting of Parametric Reflectance Model

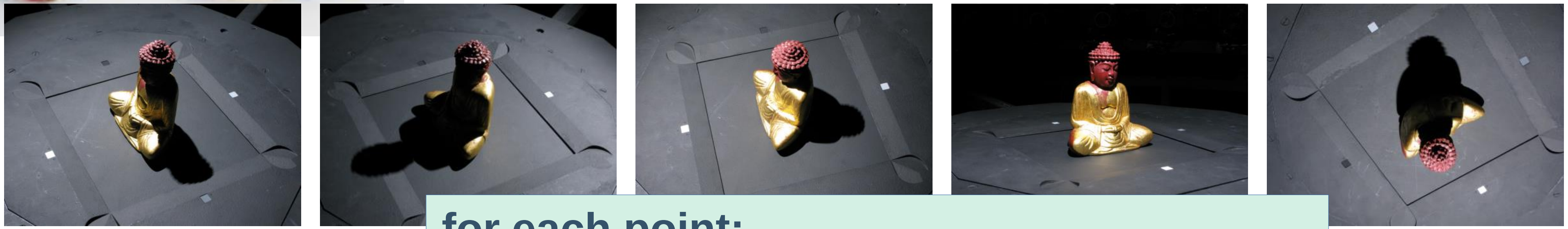
› Recovering parameters of Phong model

we recover the Phong parameters from photos

$$\mathbf{k}_d, \mathbf{k}_s, t$$

7 parameters per point on surface

$$\mathrm{C} = \mathbf{k}_d \mathrm{I}(\mathbf{n} \cdot \mathbf{L}) + \mathbf{k}_s \mathrm{I}(\mathbf{E} \cdot \mathbf{R})^t$$



. . .

for each point:
- color at corresponding pixel for each photo
- light direction and intensity
- surface normal
- calibrated camera

Fitting of Parametric Reflectance Model

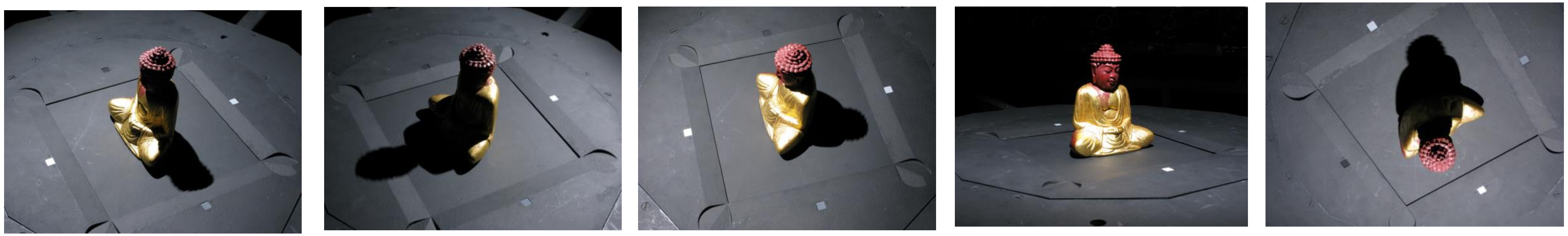
› Recovering parameters of Phong model

we recover the Phong parameters for one surface point

$$\mathbf{k}_d, \mathbf{k}_s, t$$

you can repeat the same procedure
for all surface points

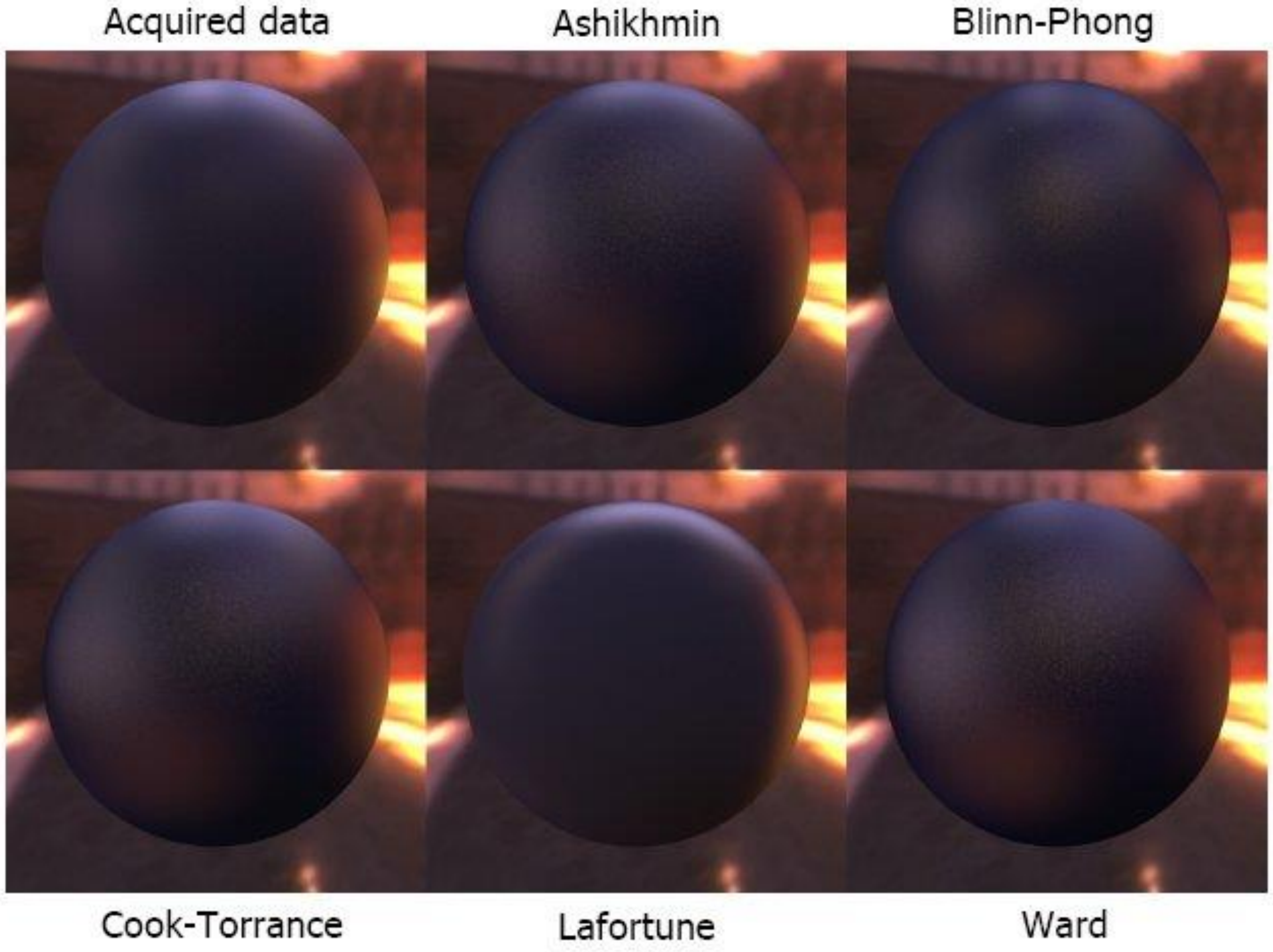
if you know part of the surface is homogeneous,
you can use information from multiple points to
retrieve the material



. . .

Fitting of Parametric Reflectance Model

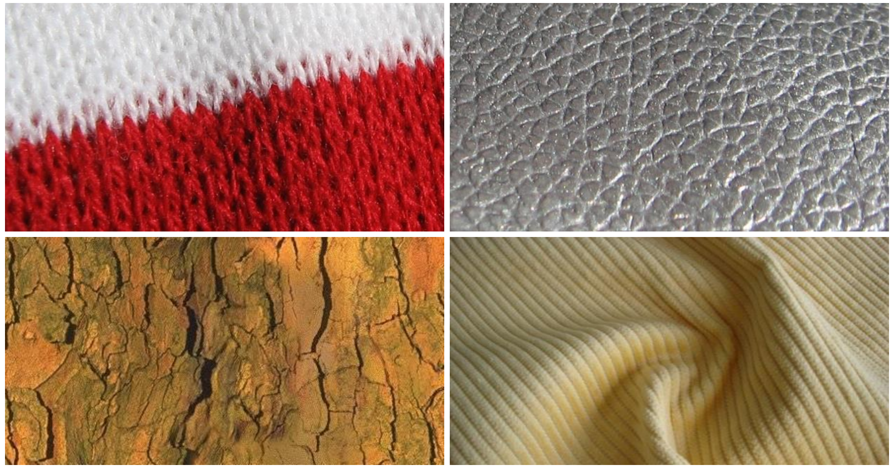
› There are many other reflectance functions / parametric models



Parametric Reflectance Models

- Pros:
  - Only a few parameters
  - Good results for many materials

- Challenges:
  - Relies on accurate reconstruction of surface geometry
  - Difficulty of finding a parametric model for all types of appearance characteristics
  - Cannot handle complex light transport effects (subsurface scattering, interreflections, self-occlusions, self-shadowing)



But: Parametric models are limited ...

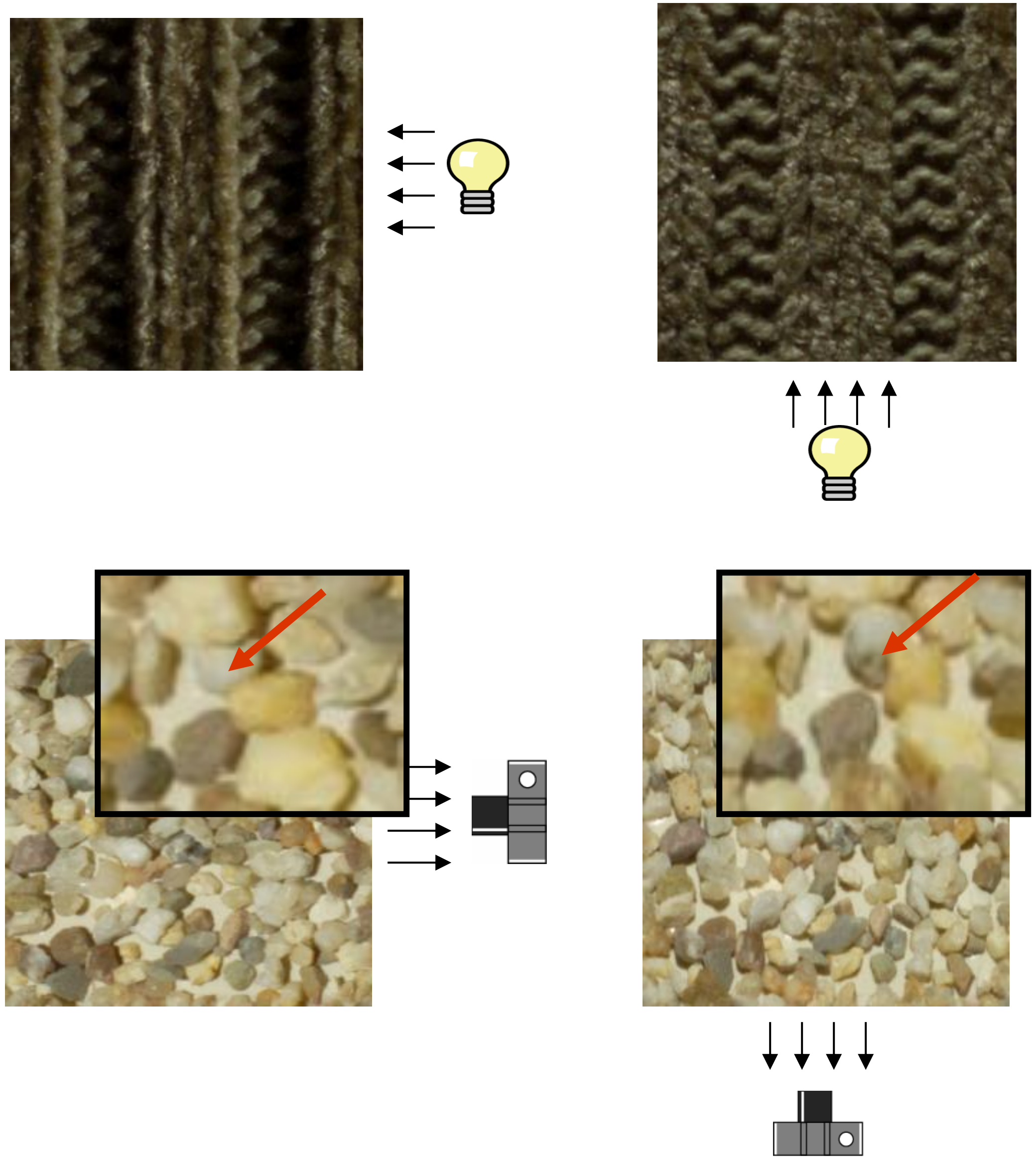
parametric model
(BRDF)

vs.

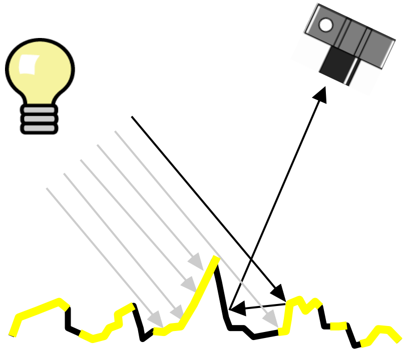
desired result

McAllister 2002

Examples



Shadowing



Occlusions/Parallax

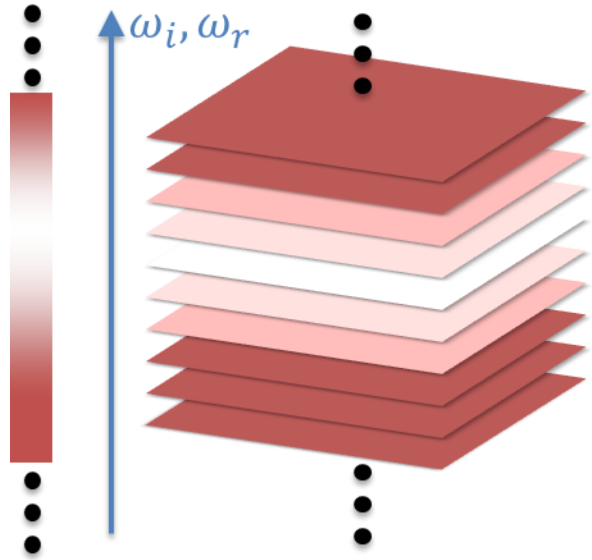
How to represent material appearance?

- **Trade-off**
  - How many measurements are needed to capture details?
  - Available memory resources?
  - Can we fill the gaps in-between measurements?

Parametric models:
- more compact representation
- analytical model → few parameters

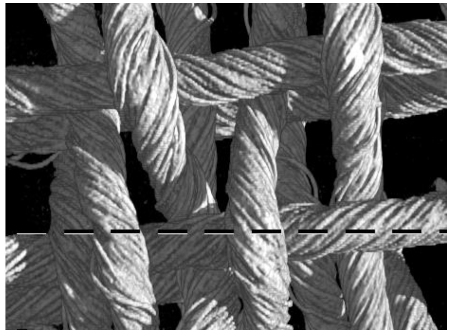
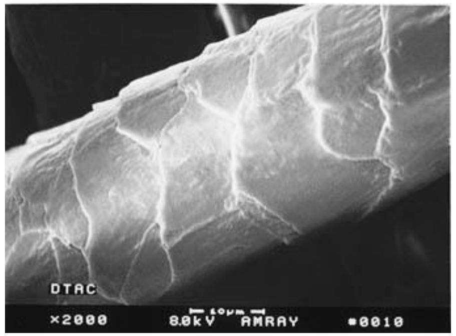
vs.

Dense measurements:
- very large raw data (terabytes!)
- hard to use in many applications



How to achieve a higher accuracy?

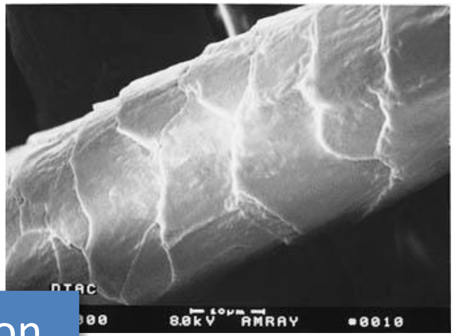
larger ⟷ smaller

		
Macroscopic	**Mesoscopic**	**Microscopic**
"3D shape"	e.g. spun	"microfacets"
e.g. visible	yarns, weaving	e.g. scales on
Explicit surface geometry: polygon mesh	Surface orientation modulation: normal maps, …	Statistical distribution function: BRDF
"meters"	**"millimeters"**	**"micrometers"**

6 orders of magnitude!

How to achieve a higher accuracy?

larger — smaller



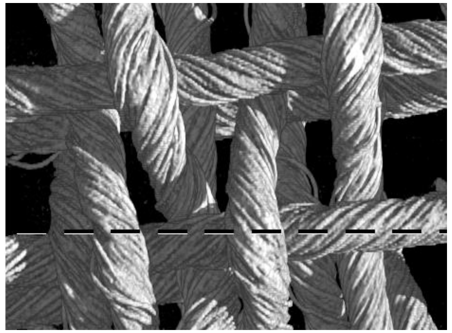
Macroscopi	lesoscopic	Microscopic
"3D shape" e.g. visible	e.g. spun yarns, weaving	"microfacets" e.g. scales on
Explicit surface geometry: polygon mesh	Surface orientation moc normal maps, …	Statistical function: BRDF
"meters"	**"millimeters"**	**"micrometers"**

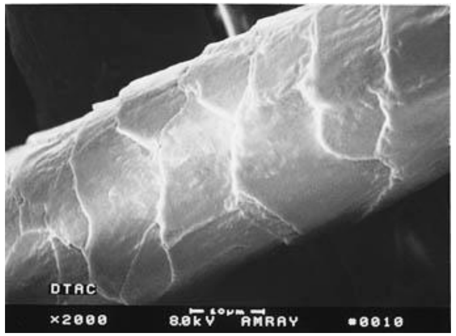
💣 **insufficient**

6 orders of magnitude!

How to achieve a higher accuracy?

larger — smaller





Macroscopic	**Mesoscopic**	**Microscopic**
3D „shape“	e.g. spun	„microfacets“
e.g. visible	yarns, weaving	e.g. scales

inefficient

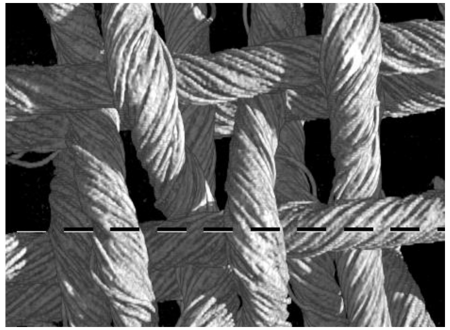
Image based modelling:
light fields, reflectance fields

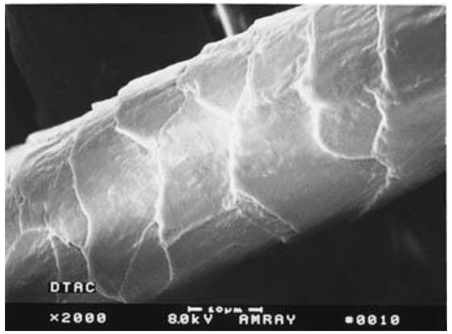
“meters” **“millimeters”** **“micrometers”**

6 orders of magnitude!

How to achieve a higher accuracy?

larger — smaller





Macroscopic	**Mesoscopic**	**Microscopic**
3D „shape“ e.g. visible wrinkles	e.g. spun yarns, weaving patterns	„microfacets“ e.g. scales

Explicit surface geometry: polygon mesh

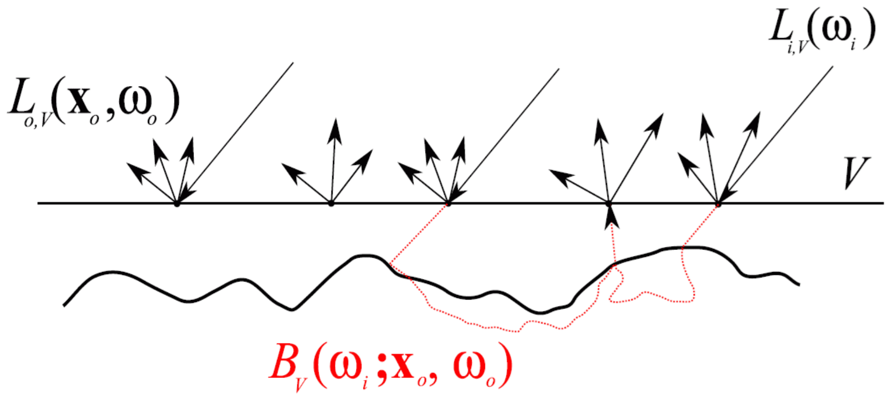
Image based modeling:
surface light/reflectance fields, **BTFs**

“meters” **“millimeters”** **“micrometers”**

6 orders of magnitude!

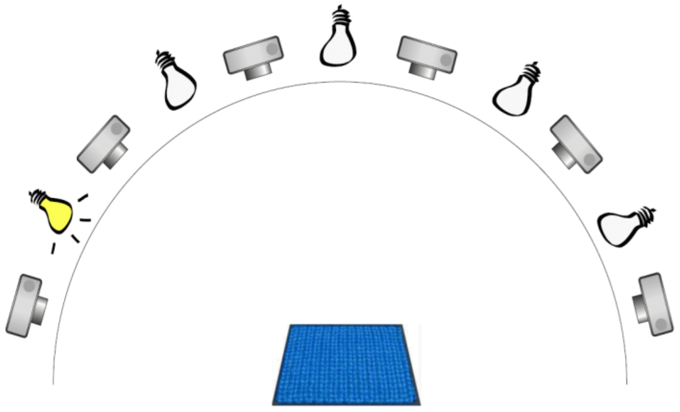
Bidirectional Texture Functions (BTFs)

- **BTF** transfers
  - incident **directional** light field $L_{i,V}(x_i, \omega_i)$
- to
  - corresponding outgoing light field $L_{o,V}(x_o, \omega_o)$
- parametrized on an approx. flat surface V



How do we capture a BTF?

- Sample the continous function $\mathcal{B}_V(\omega_i; \mathbf{x}_o, \omega_o)$:
  - Define
    - basis illuminations $\mathcal{L} = \{\mathbf{l}_i\}_{i \in I}$
    - corresponding outgoing light-fields $\mathcal{R} = \{\mathbf{r}_i\}_{i \in I}$ (one image for each direction)



  - Writing the incoming light-field as linear combination of basis illuminations

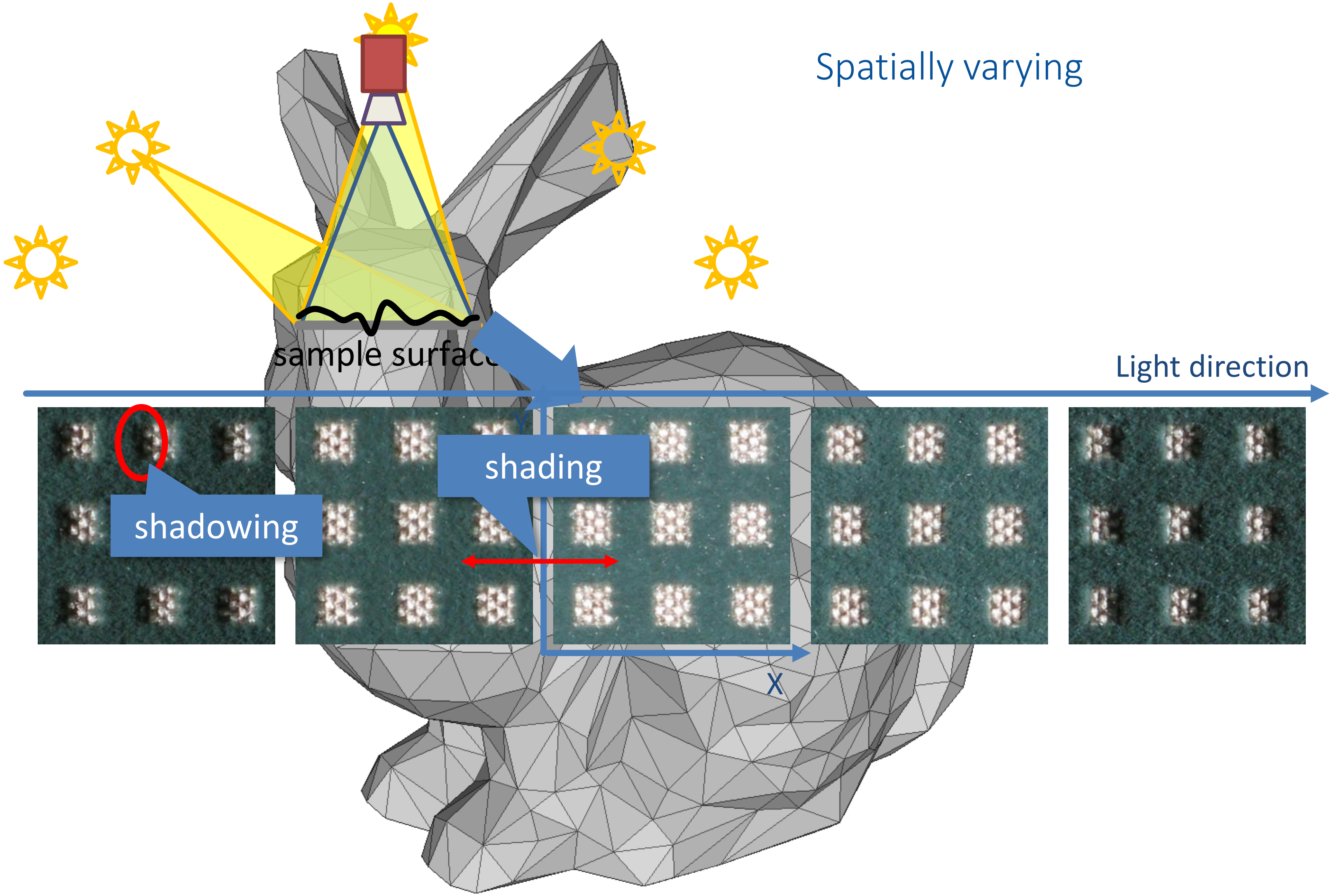
$$\mathbf{l} = \sum_i l_i \mathbf{l}_i$$

we get the discrete image-based relighting equation for BTFs

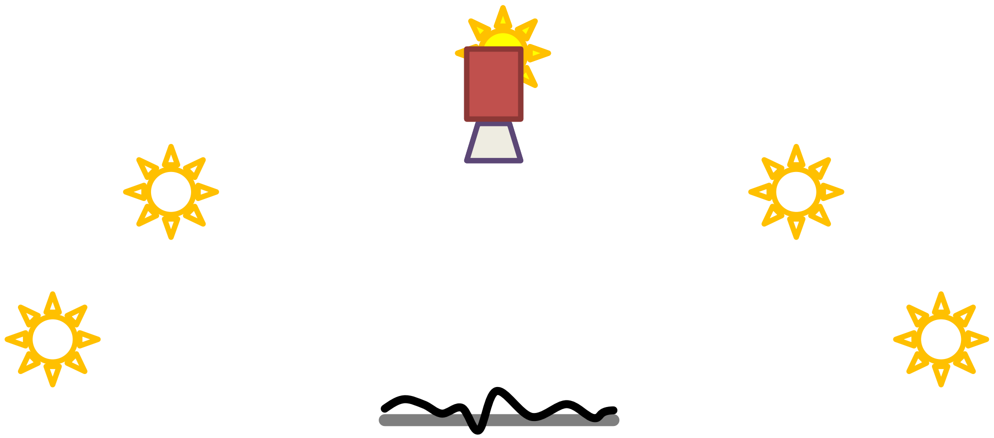
$$\mathbf{l}_o = \sum_{i \in I} l_i \mathbf{r}_i = \underbrace{\mathbf{B}^{(K \times I)}}_{\text{discrete BTF } (\rightarrow \text{ Light Transport Matrix})} \mathbf{l}$$

Bidirectional Texture Functions (BTFs)

Spatially varying

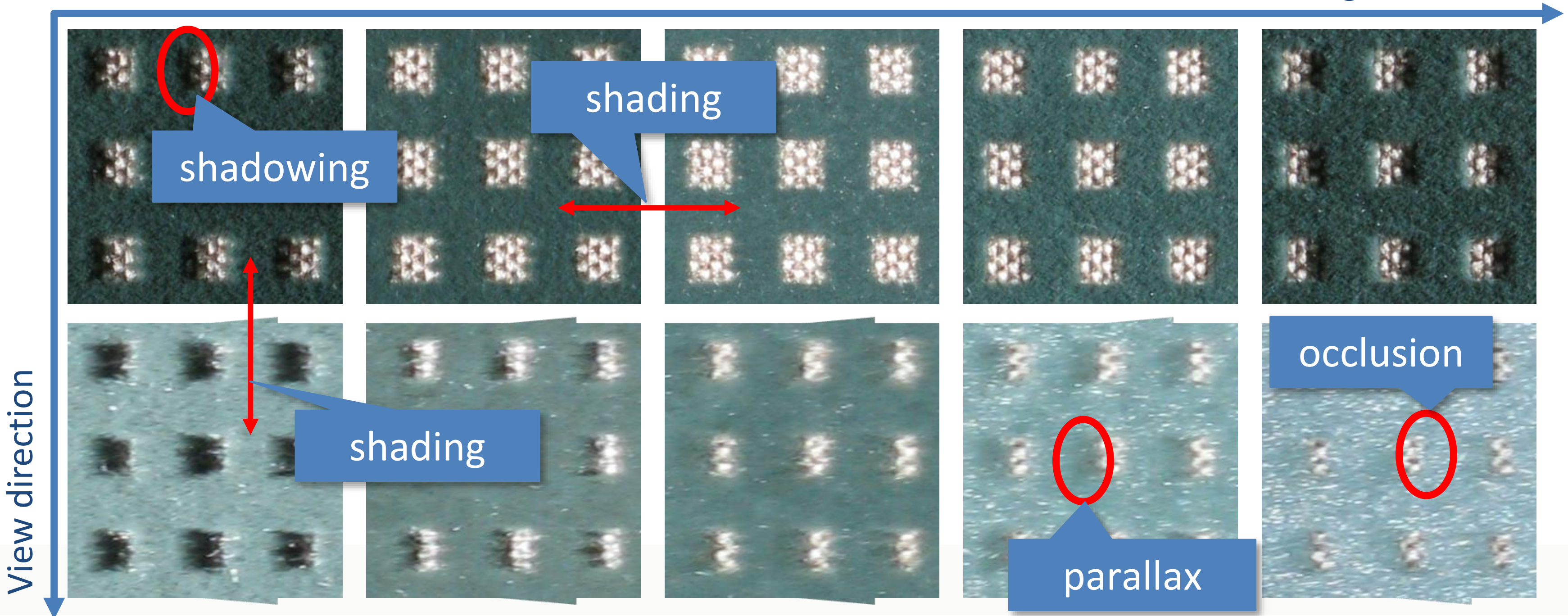


Bidirectional Texture Functions (BTFs)

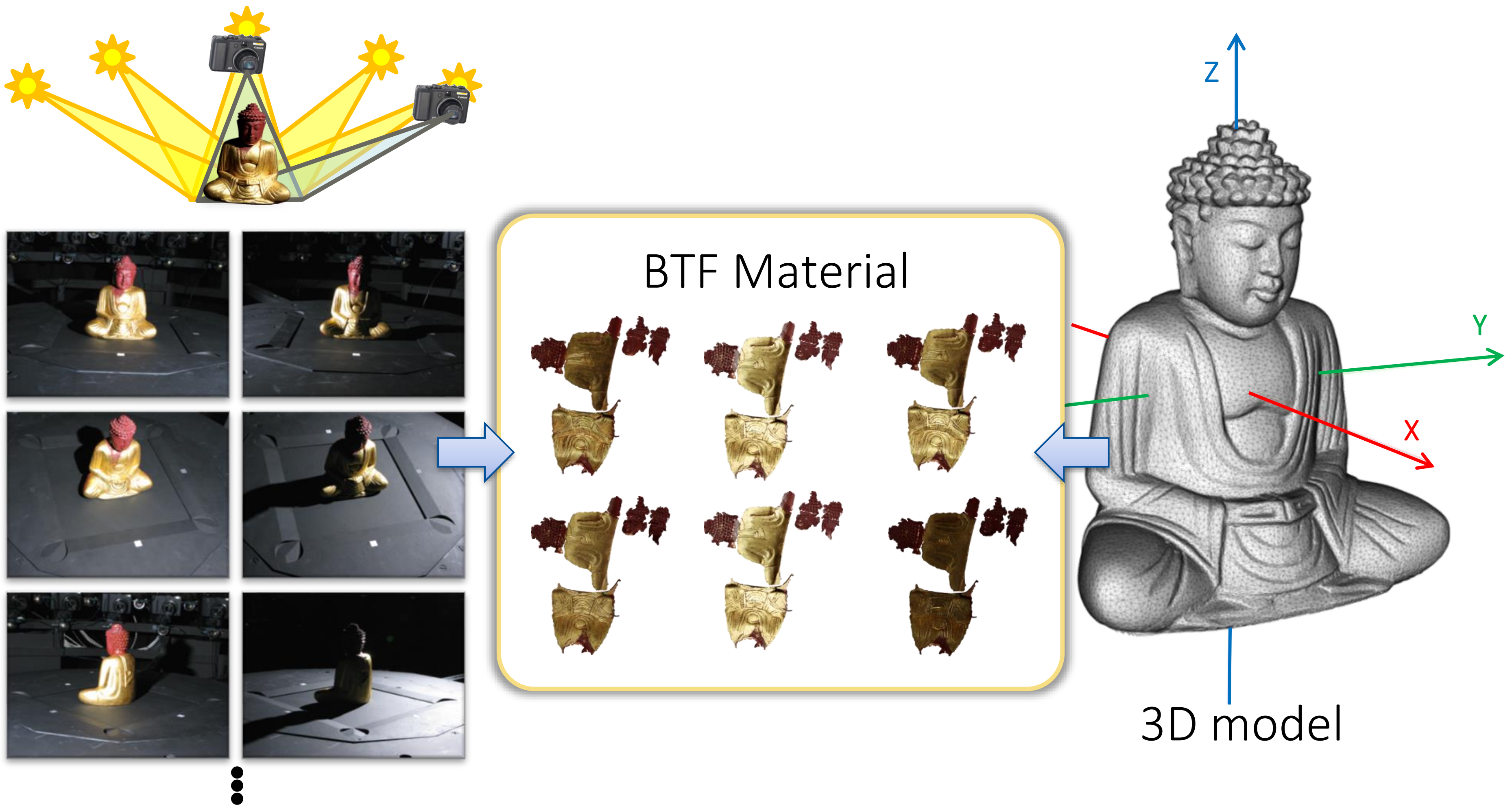


Spatially varying

Bidirectional

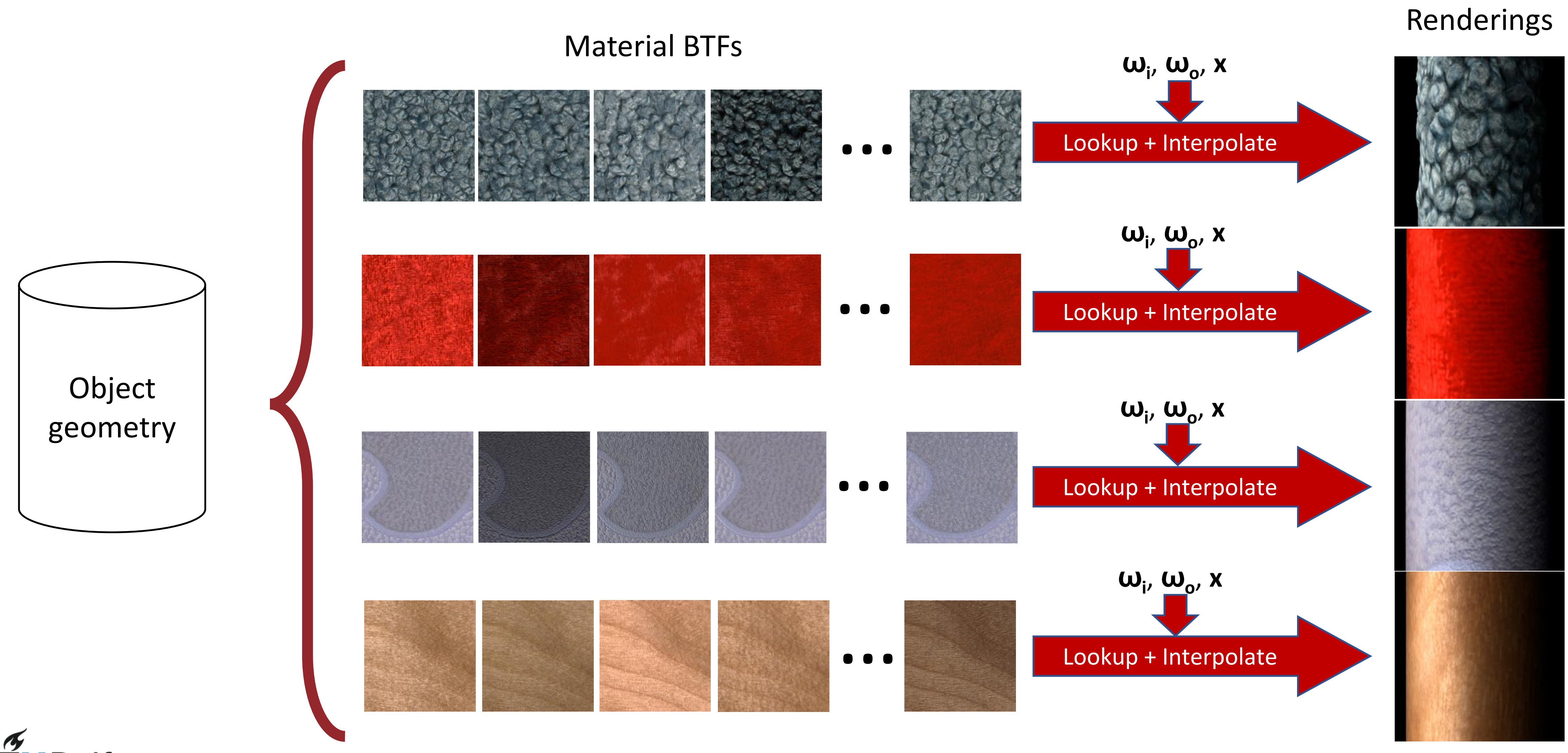


How do we capture a BTF?



C. Schwartz, M. Weinmann, R. Ruiters, and R. Klein, Integrated High-Quality Acquisition of Geometry and Appearance for Cultural Heritage, VAST 2011

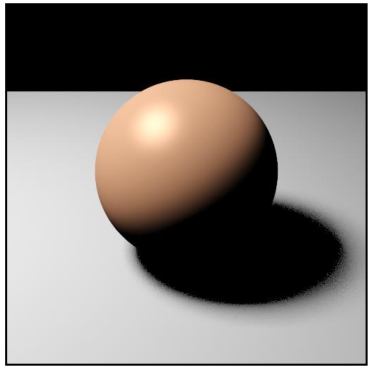
Bidirectional Texture Functions (BTFs)



Rainer, Ghosh, Jakob, Weyrich: Unified Neural Encoding of BTFs

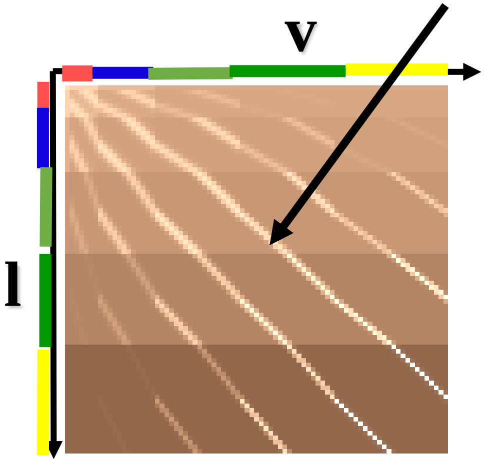
Parametric Models (BRDFs) vs. BTFs

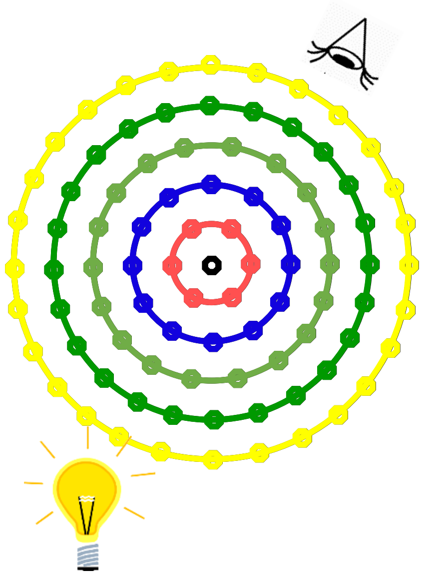
- BRDF:



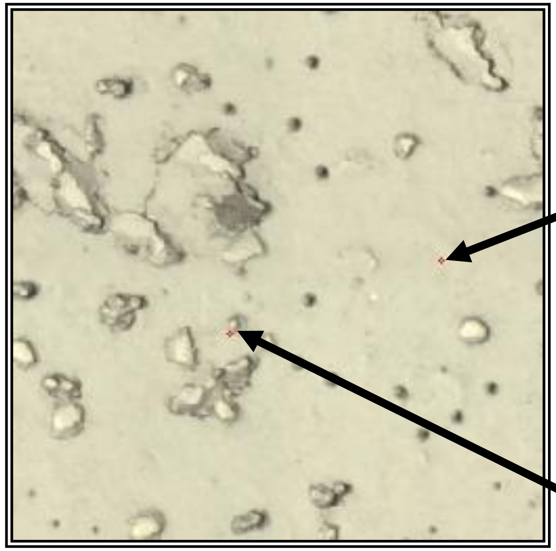
shiny plastic

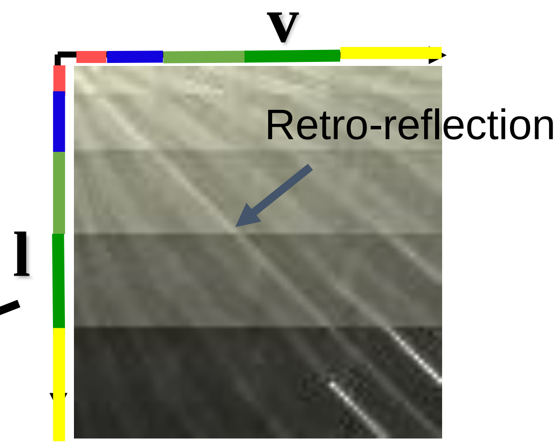
Specular reflection

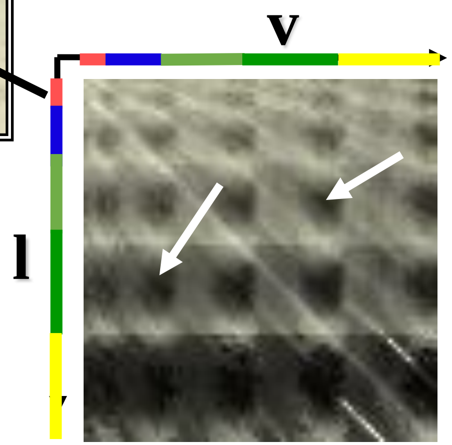




- BTFs:



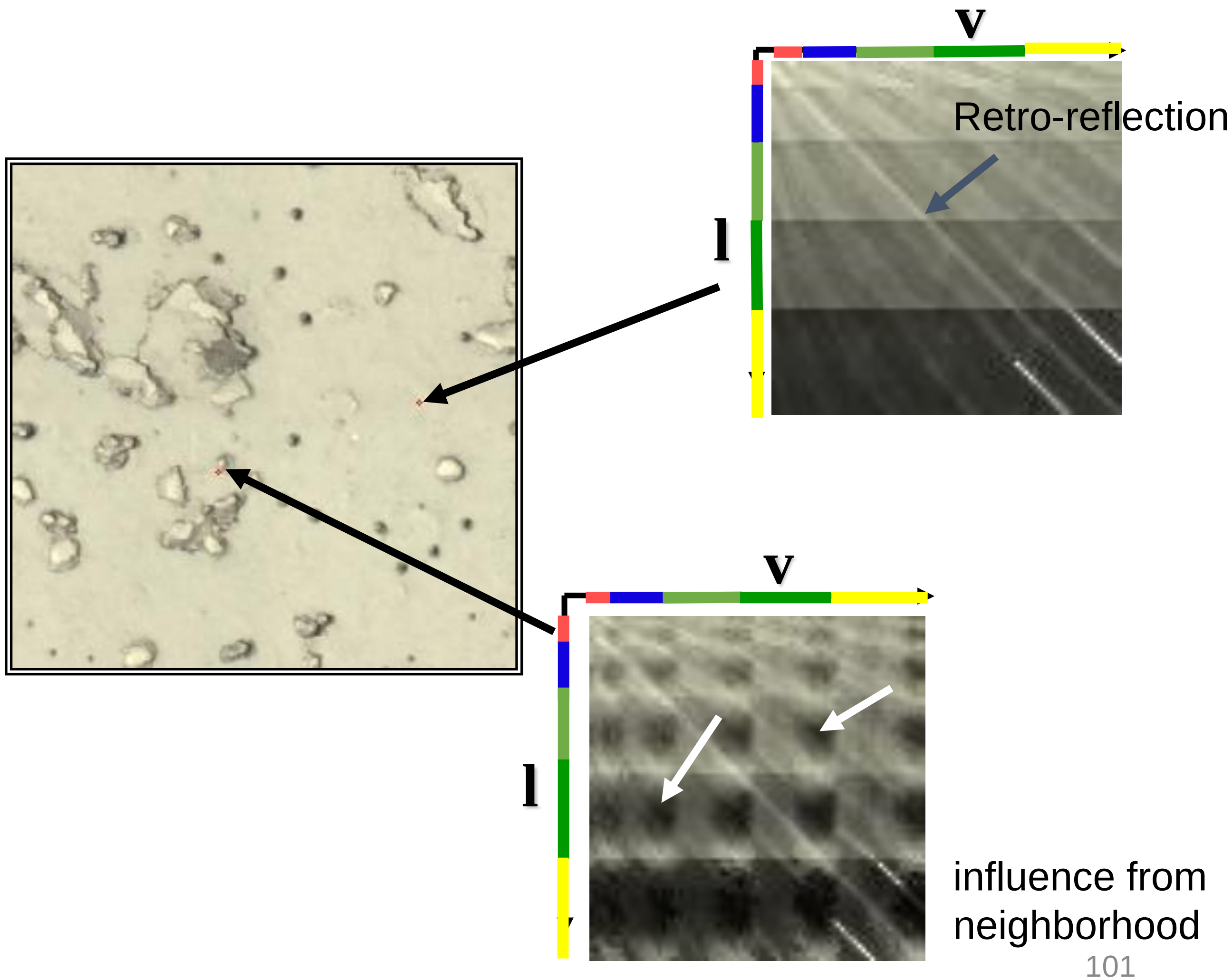




influence from
neighborhood

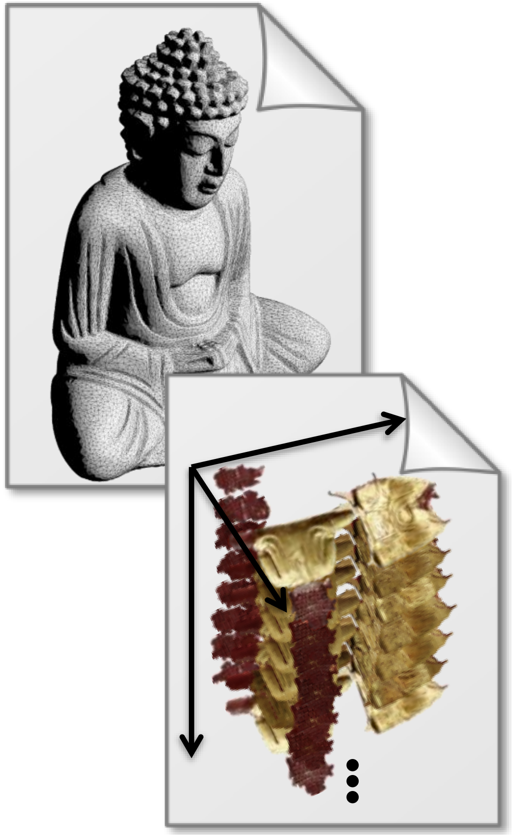
Bidirectional Texture Functions (BTFs)

- Contain also influence from neighborhood:
  - Self-shadowing
  - Self-occlusion
  - (Local) sub-surface scattering
  - Global light exchange on meso- and microscale level
  - …



Bidirectional Texture Functions (BTFs)

- Nice results, …



Photograph

Rendering

C. Schwartz, M. Weinmann, R. Ruiters, and R. Klein, Integrated High-Quality Acquisition of Geometry and Appearance for Cultural Heritage, VAST 2011

Bidirectional Texture Functions (BTFs)

Bidirectional Texture Functions (BTFs)

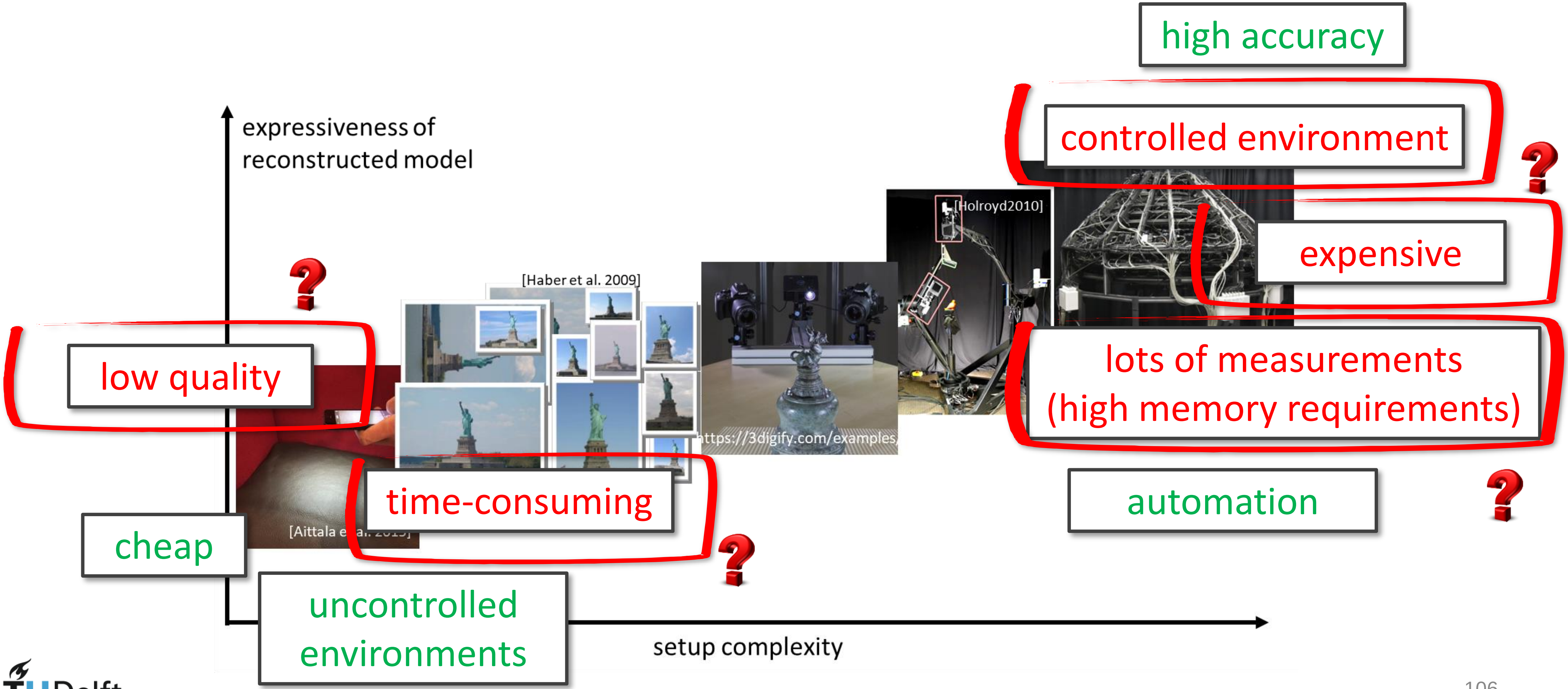
- … **but:**

= 1x

  - High storage requirements!
    - Not suitable for real-time rendering, streaming, etc.

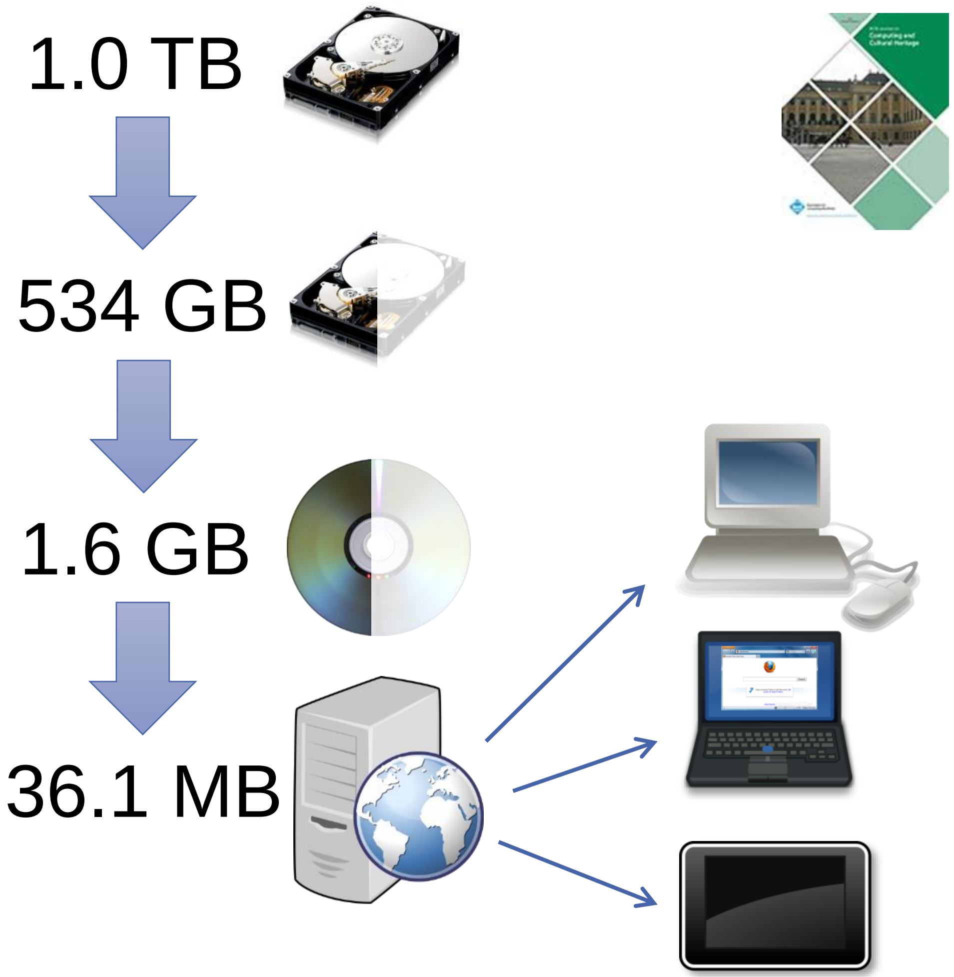
Outlook: Potential of AI?

Outlook: Potential of AI



Outlook: Potential of AI?

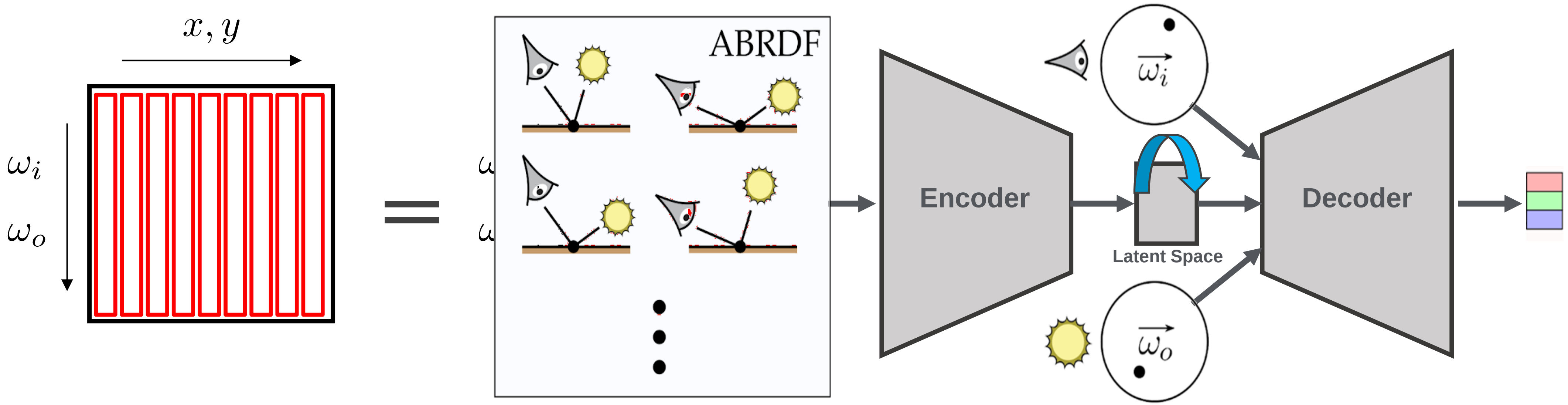
- Compression?



C. Schwartz, R. Ruiters, M. Weinmann, and R. Klein, WebGL-based Streaming and Presentation of Objects with Bidirectional Texture Functions, Journal on Computing and Cultural Heritage, 2013

Outlook: Potential of AI?

- Compression?
  - Encode measured data in compact subspace (=latent space)
  - Decode from latent space
  - Methods:
    - Principal component analysis (PCA)
    - Neural networks



Rainer, Ghosh, Jakob, Weyrich: Unified Neural Encoding of BTFs

Outlook: Potential of AI

› How much information do we get from a single image?

neural network to predict normals, diffuse/specular components, etc.

Input

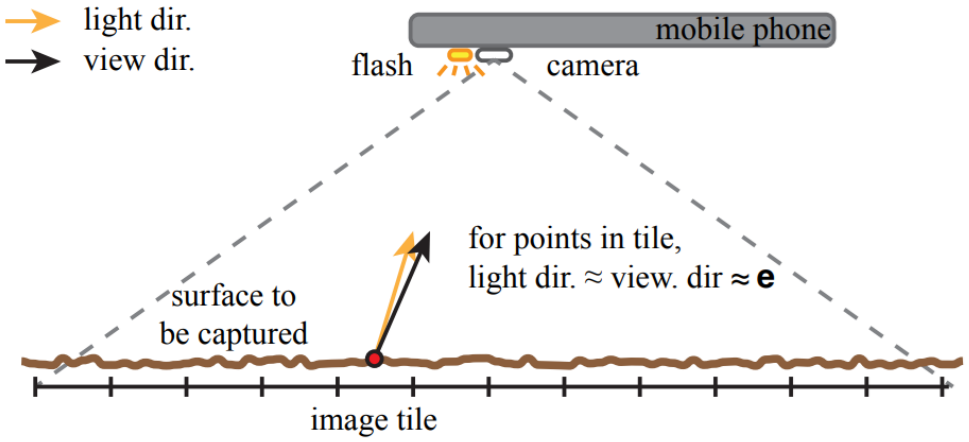
- Idea:
  - Combine AI …
  - … and CG

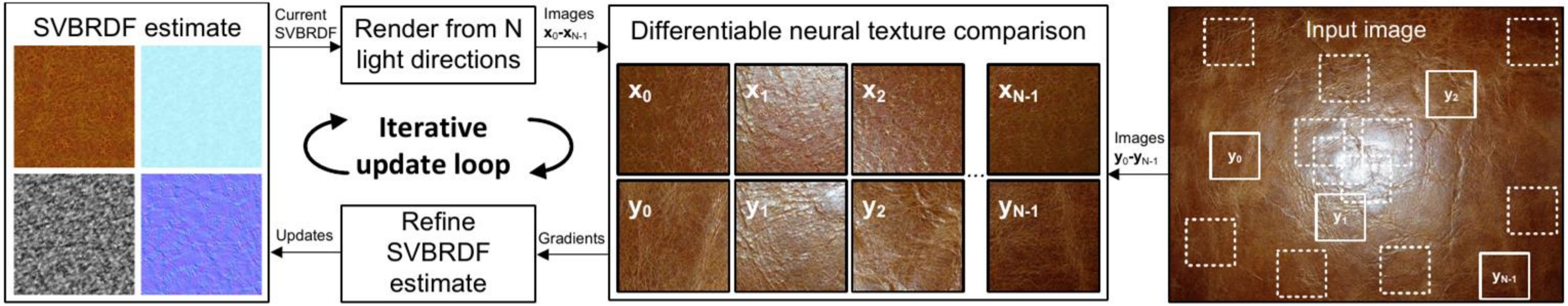
Deschaintre et al., Single-image SVBRDF capture with a rendering-aware deep network." ACM Transactions on Graphics (TOG) 37.4 (2018): 128

Outlook: Potential of AI

› How much information do we get from a single image?

- Further idea:
  - Get multiple configurations from patchwise analysis



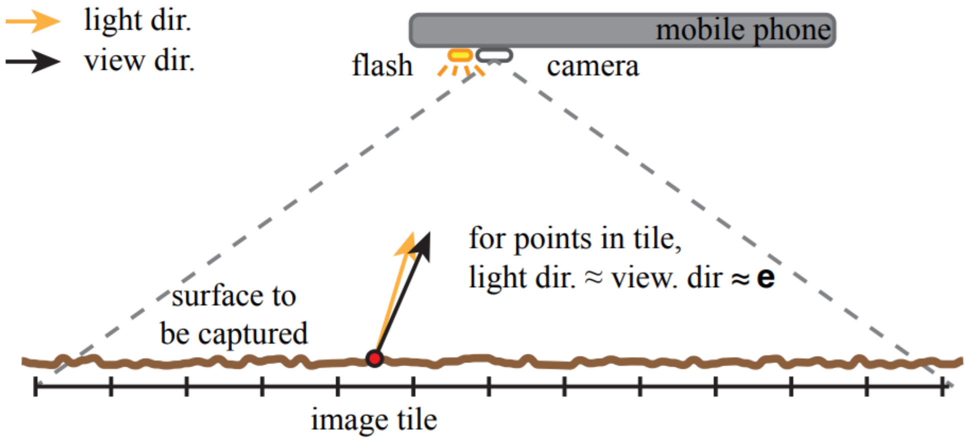


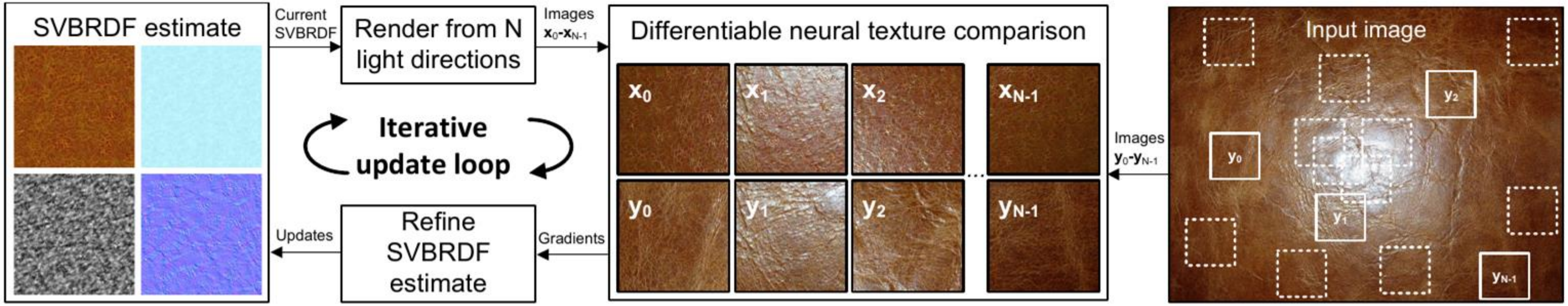
Aittala et al., Reflectance Modeling by Neural Texture Synthesis

Outlook: Potential of AI

› How much information do we get from a single image?

- Further idea:
  - Get multiple configurations from patchwise analysis



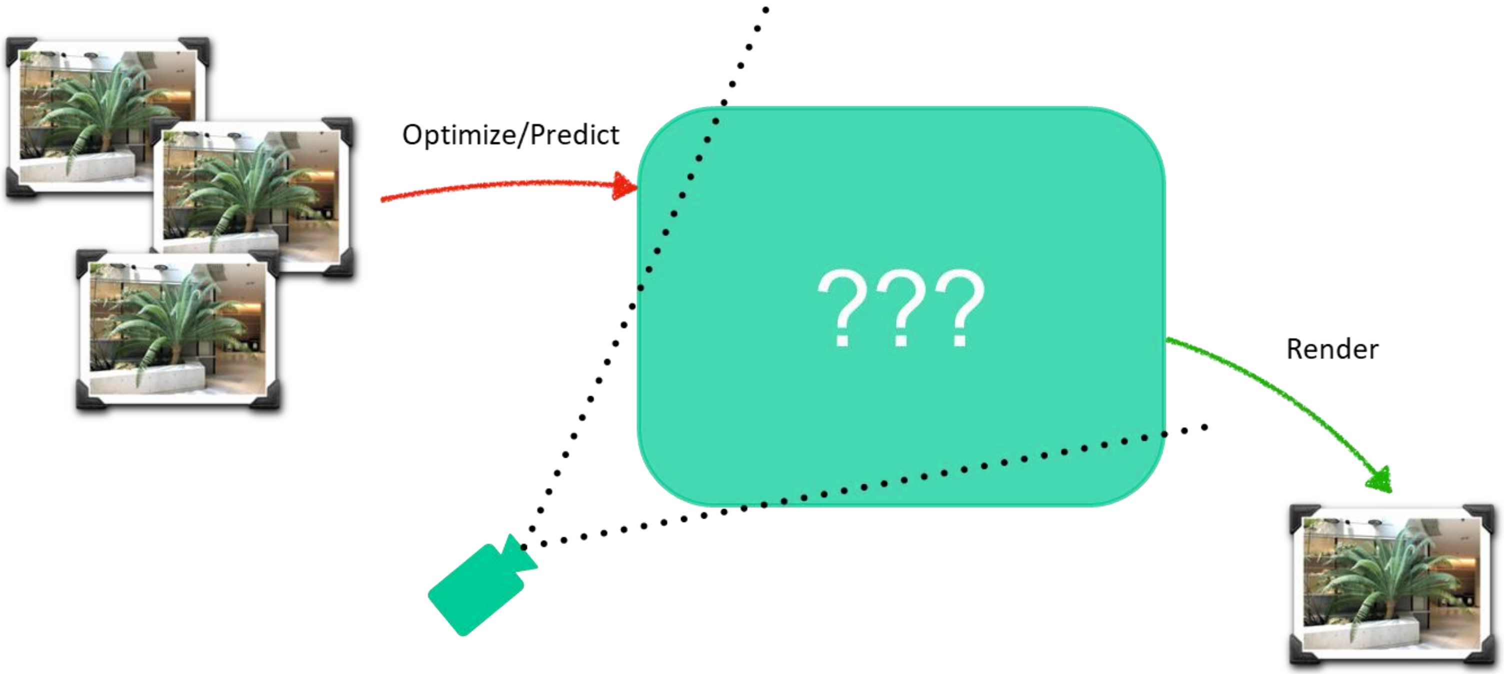


Aittala et al., Reflectance Modeling by Neural Texture Synthesis

Outlook: Potential of AI

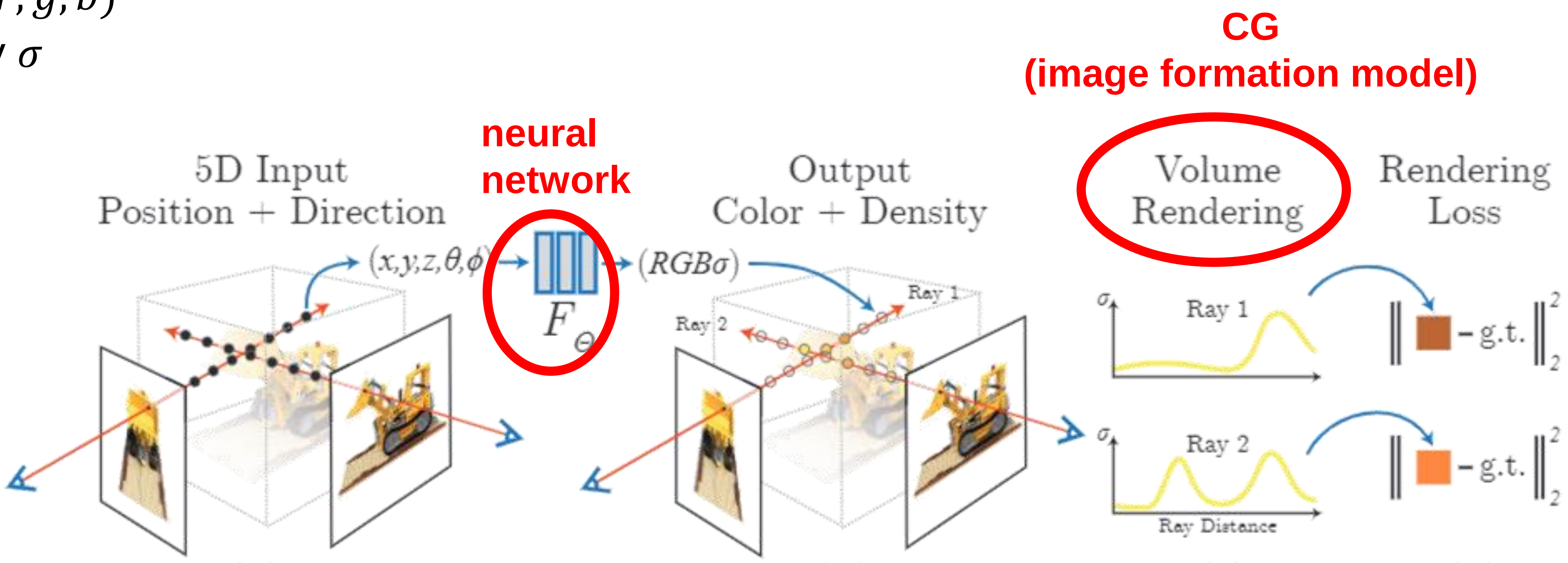
- Learning-based scene representation
  - Embedding of image formation process to allow producing images
  - Make generated images match given photographs

- Assumptions:
  - Known camera poses/ intrinsics
  - Unknown geometry



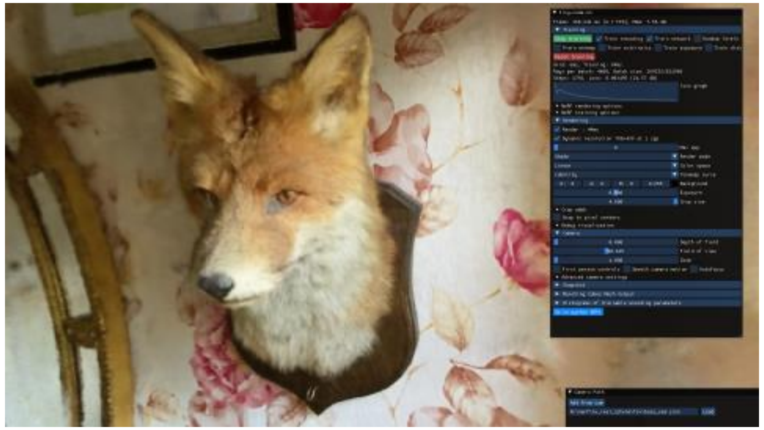
Outlook: Potential of AI

- Neural Radiance Fields (NeRF):
  - Neural network $F_\Theta: (x, y, z, \theta, \phi) \rightarrow (r, g, b, \sigma)$
    - 3D location (x, y, z)
    - 2D viewing direction (θ, ϕ)
    - Color (r, g, b)
    - Density σ

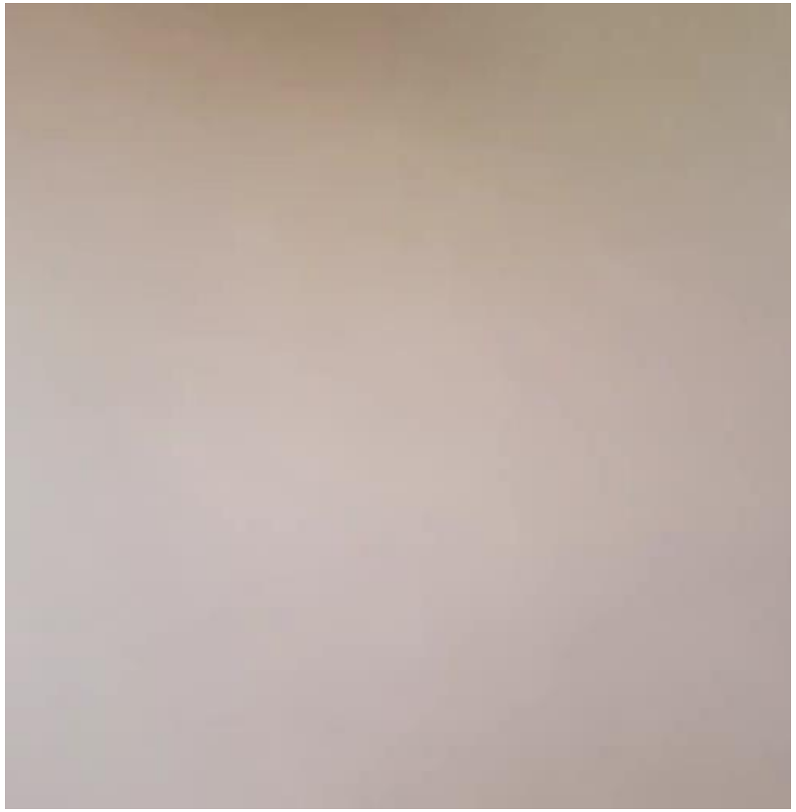


Outlook: Appearance Capture under Complex Conditions

› (Advanced) NeRF variant:



one of the input images



live training

Mueller et al., Instant Neural Graphics Primitives with a Multiresolution Hash Encoding

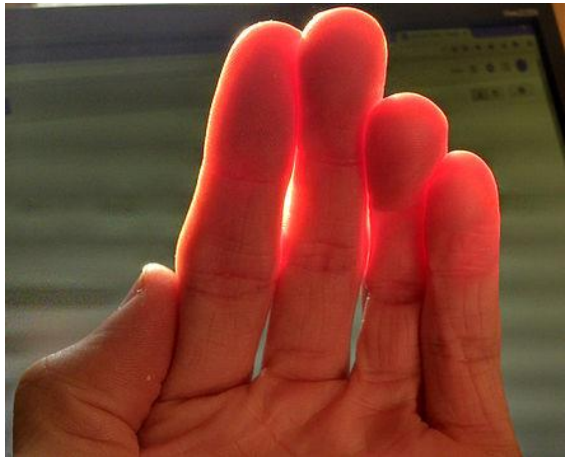
Outlook: Appearance Capture under Complex Conditions

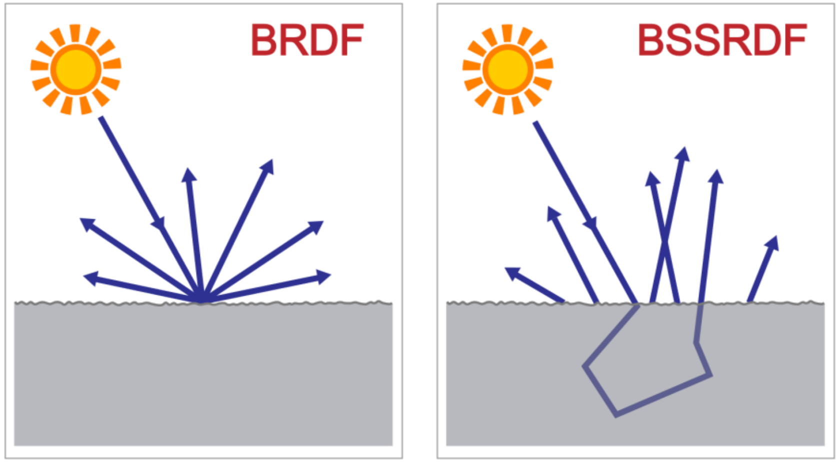
[AI Artists with Instant NeRF | NVIDIA]

What we did not discuss so far ...

- Scattering:

there is more than just reflection





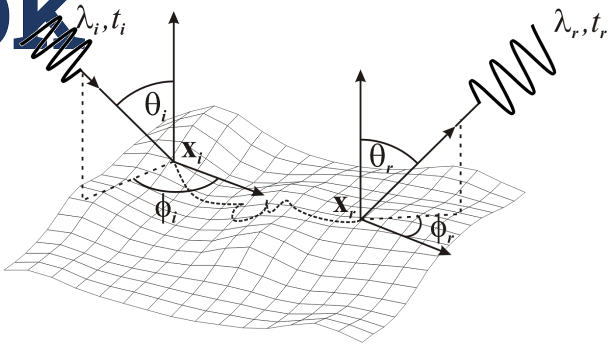
https://www.wikiwand.com/en/Subsurface_scattering

http://www.pxleyes.com/photography-picture/4ca23c04f1412/Street-Refraction.html

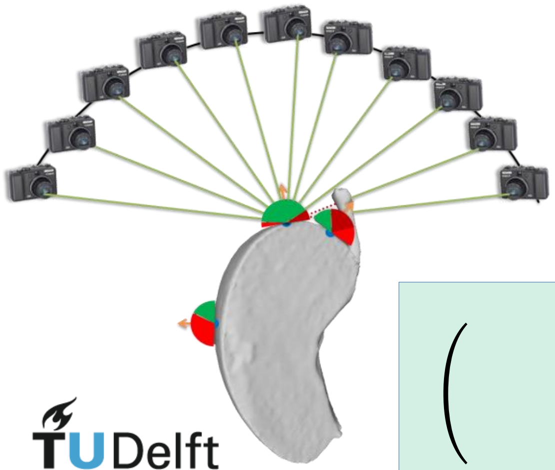
https://aavos.eu/glossary/fluorescence/

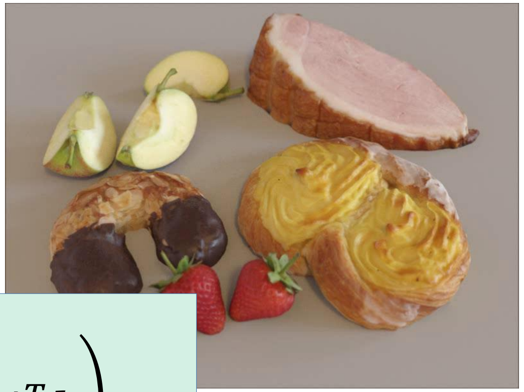
https://commons.wikimedia.org/wiki/File:BSSDF01_400.svg

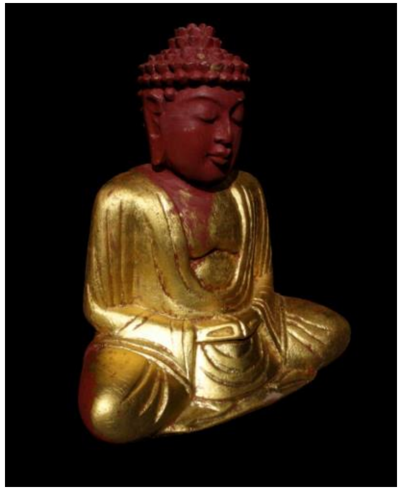
Summary & Outlook



- Summary:
  - Basics of material appearance
  - Appearance capture
  - Inference of digital material representations
    - Parametric models (compact, less accurate)
    - Data-driven models (less compact, more accurate)
    - (Outlook on) potential of AI







$$\left(A^T A \right)_{6\times 6} \left(\boldsymbol{x} \right)_{6\times 1} = \left(A^T \boldsymbol{b} \right)_{6\times 1}$$

TU Delft