3D Computer Graphics

Multimedia Techniques & Applications

Yu-Ting Wu

(with slides borrowed from Prof. Yung-Yu Chuang, Prof. Tzu-Mao Li, and Dr. I-Chao Shen)

Outline

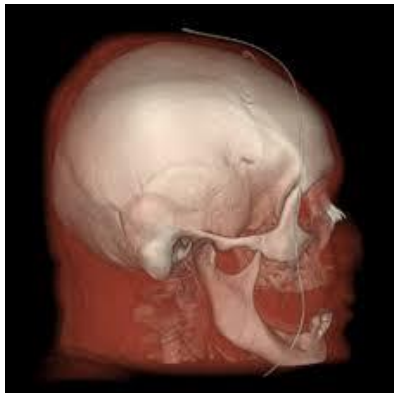
- Overview
- Modeling
- Animation
- Rendering
- Film production pipeline
- Ray tracing and rasterization

Outline

- **Overview**
- Modeling
- Animation
- Rendering
- Film production pipeline
- Ray tracing and rasterization

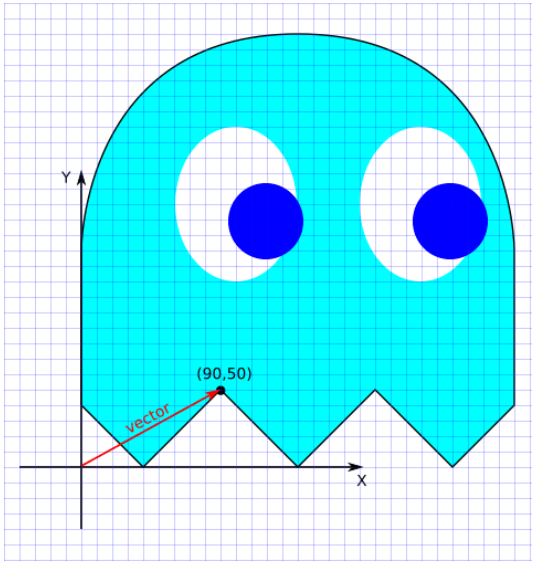
What is Computer Graphics

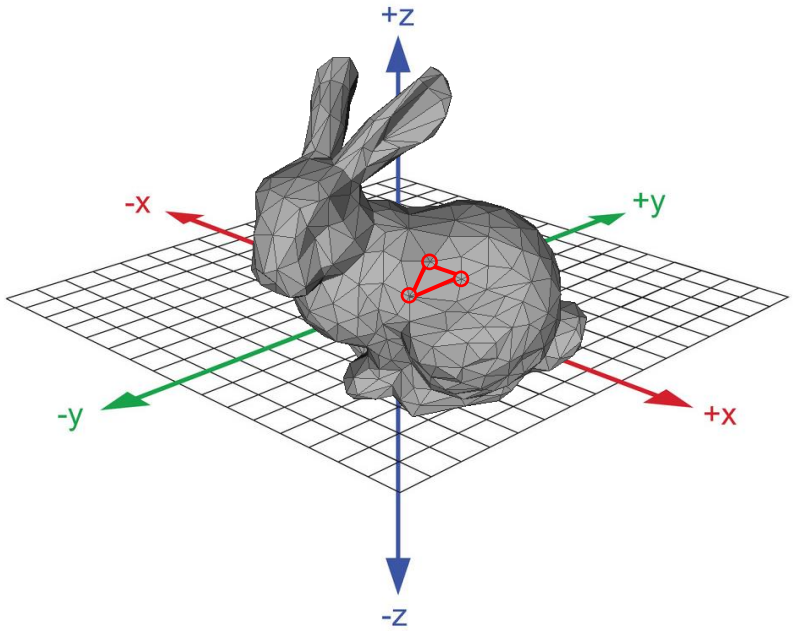
- Computer graphics are pictures and films created using computers
- Computer graphics is the process of creation, storage and manipulation of models and images using data structure and algorithms



From 2D Graphics to 3D Graphics

- We have talked about 2D vector graphics, now we will extend it to the **3D** world





2D coordinate (x, y)

2D shapes

2D transformation

3D coordinate (x, y, z)

3D shapes

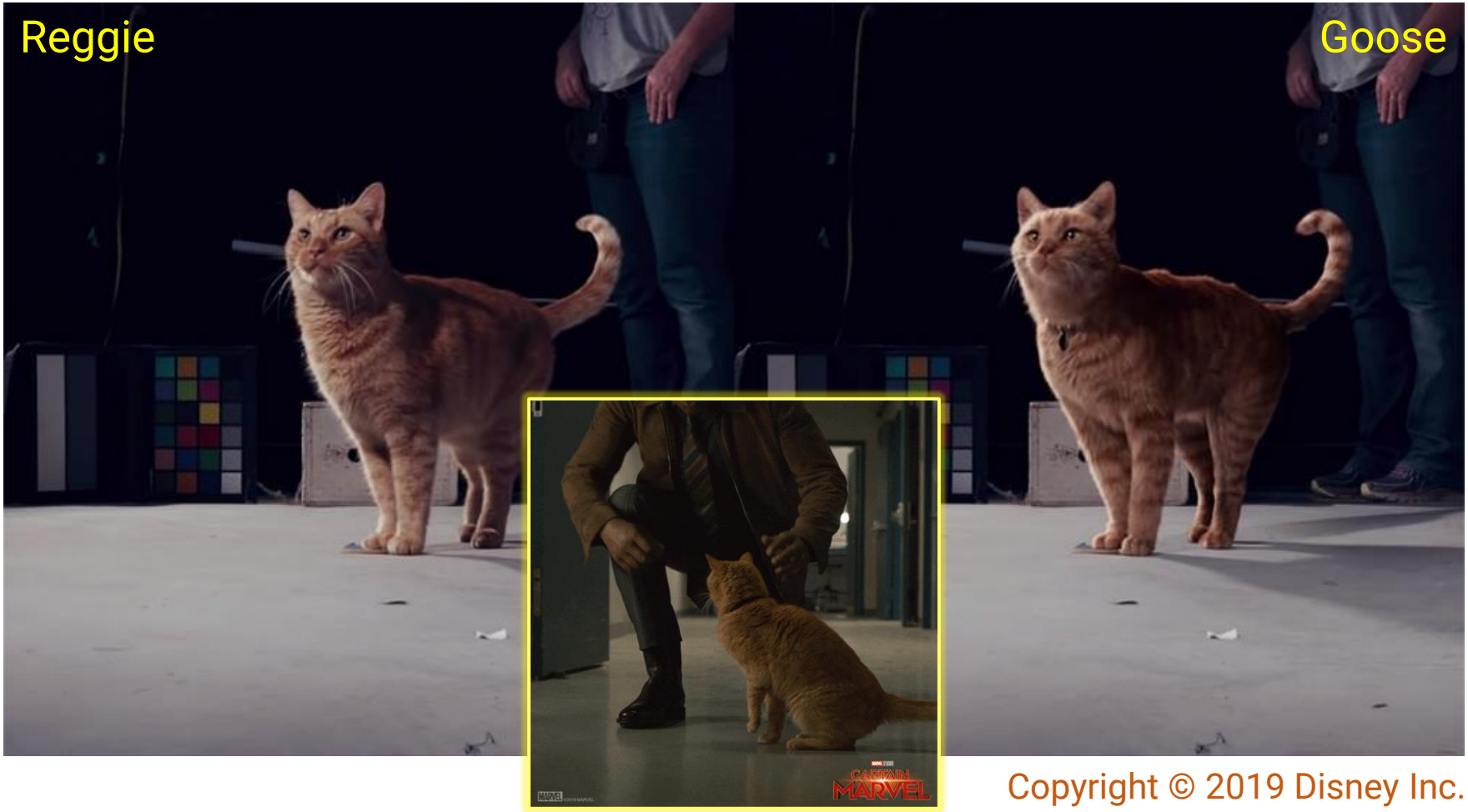
3D transformation

What Happened in Previous 20 Years

Resident Evil 3 (1999)

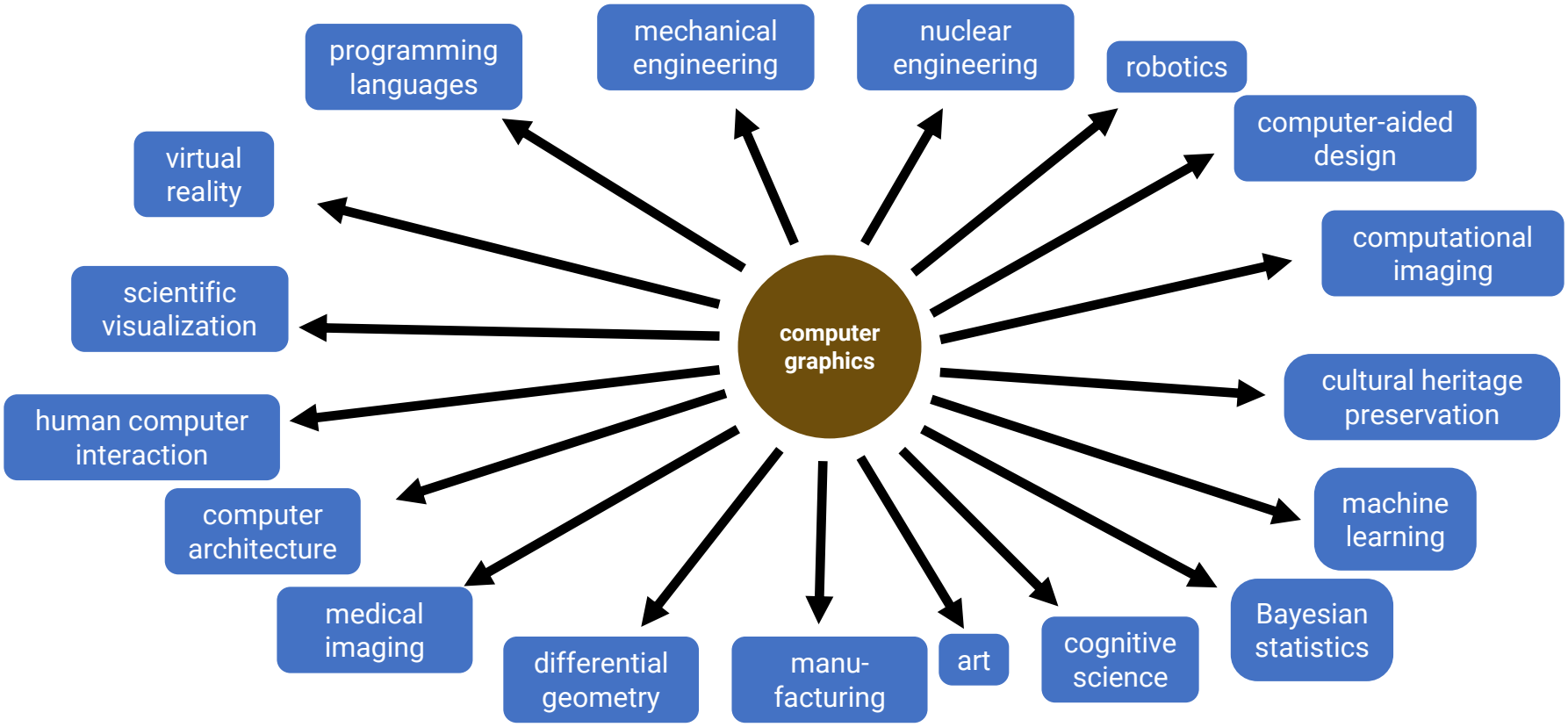
Resident Evil 3 Remake (2020)

Which Cat is Real? (Goose or Reggie)



Why Computer Graphics is Important

- Graphics push advances in many fields



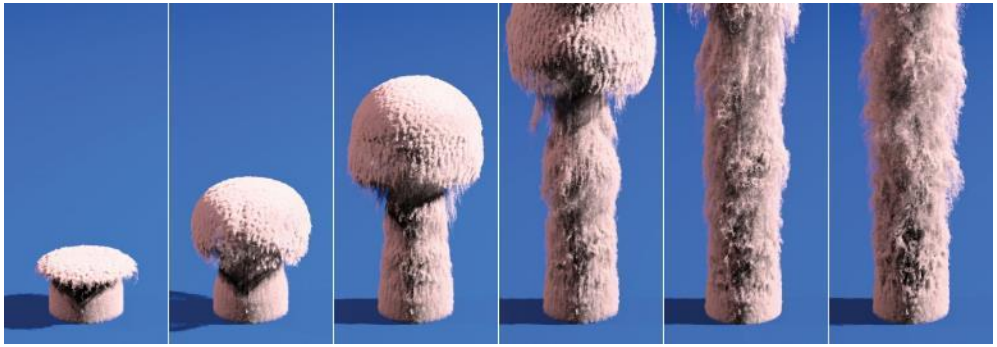
Applications of Computer Graphics

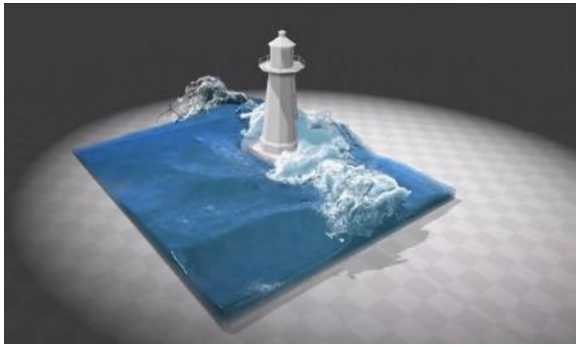
- Lighting and architecture design




Applications of Computer Graphics (cont.)

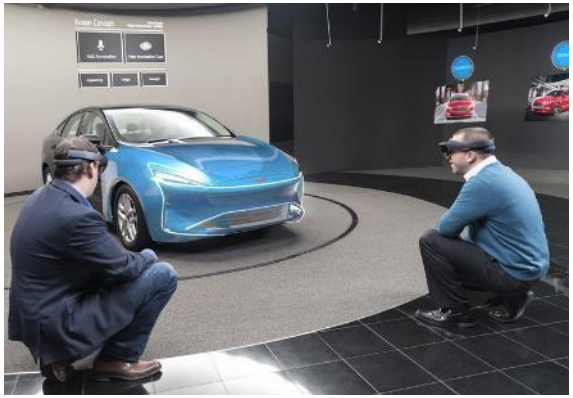
- Visualization of scientific data and physical simulation





Applications of Computer Graphics (cont.)

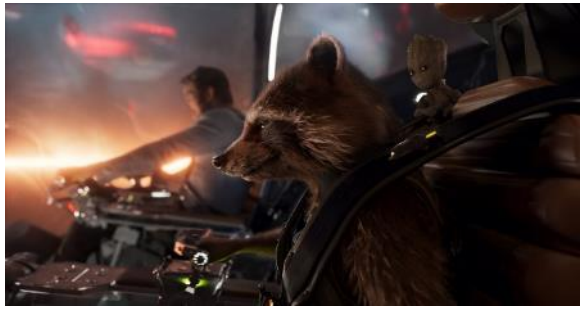
- Games, AR, MR, and VR



Applications of Computer Graphics (cont.)

- Film production

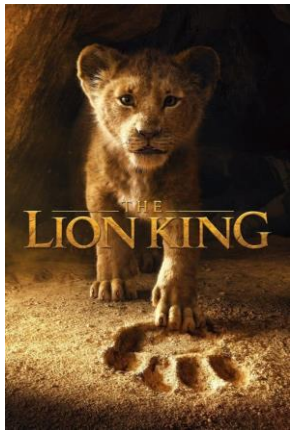






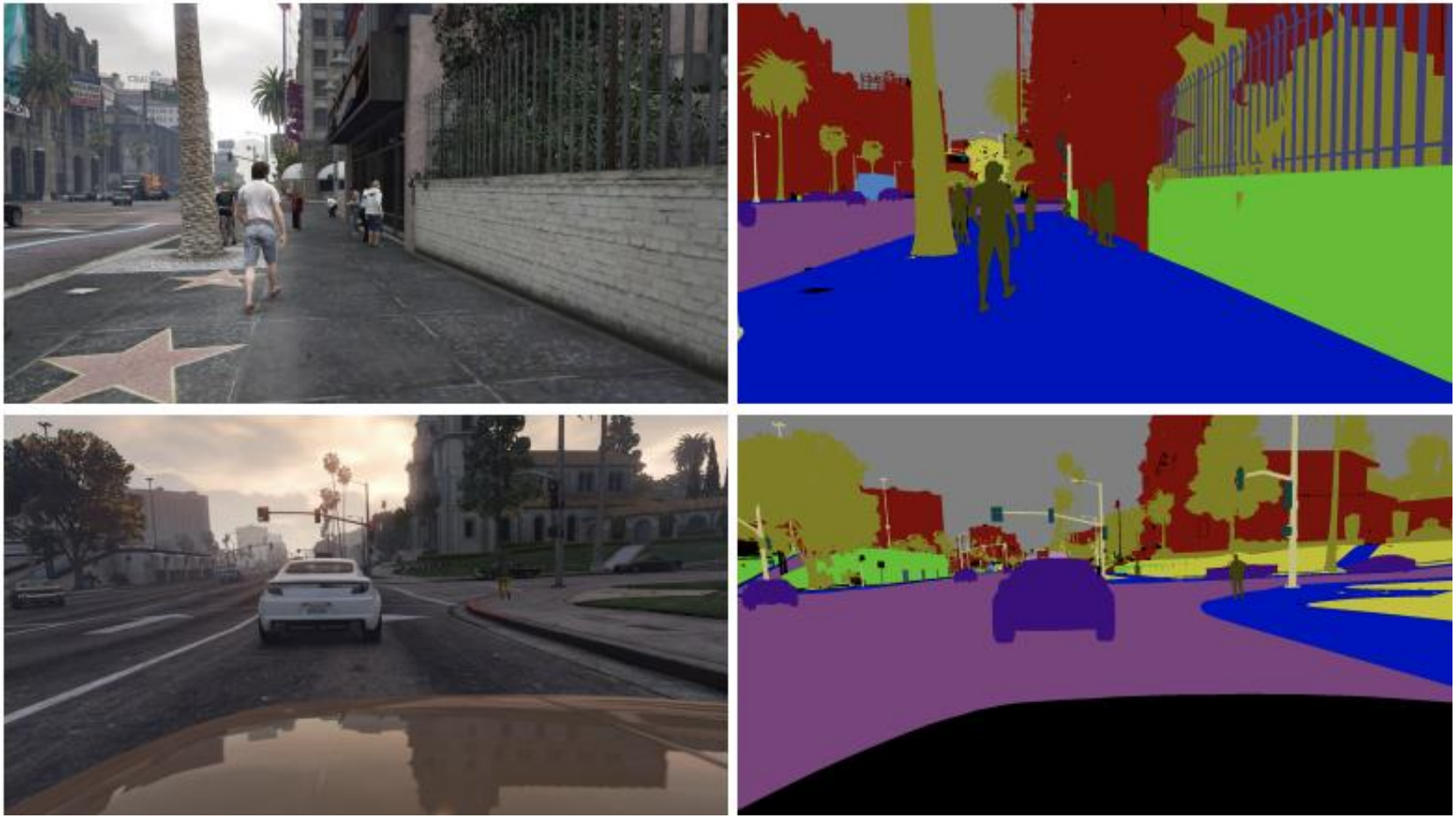

Applications of Computer Graphics (cont.)

- Film production



Applications of Computer Graphics (cont.)

- Training data generation for deep learning



Description of a 3D World

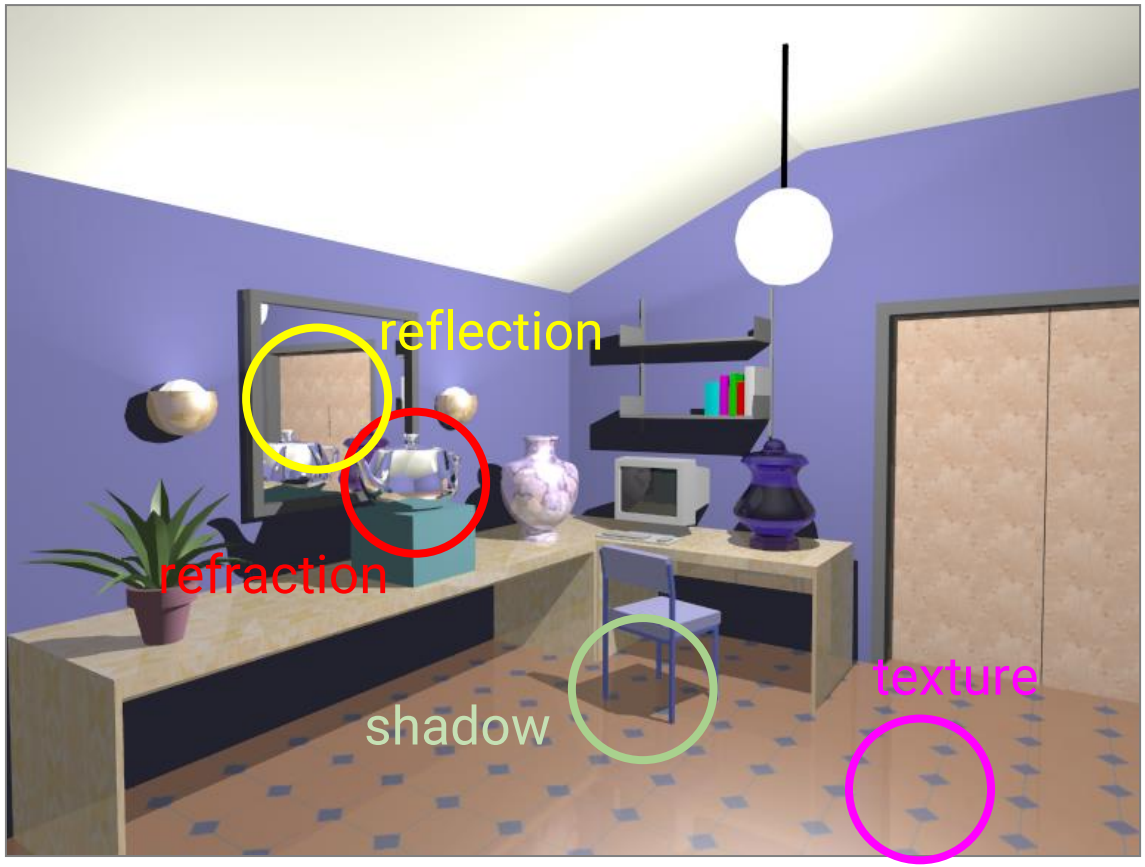
- Define **geometry** of the objects (or scene)

Description of a 3D World (cont.)

- Model **materials** of the 3D objects and simulate **lighting**

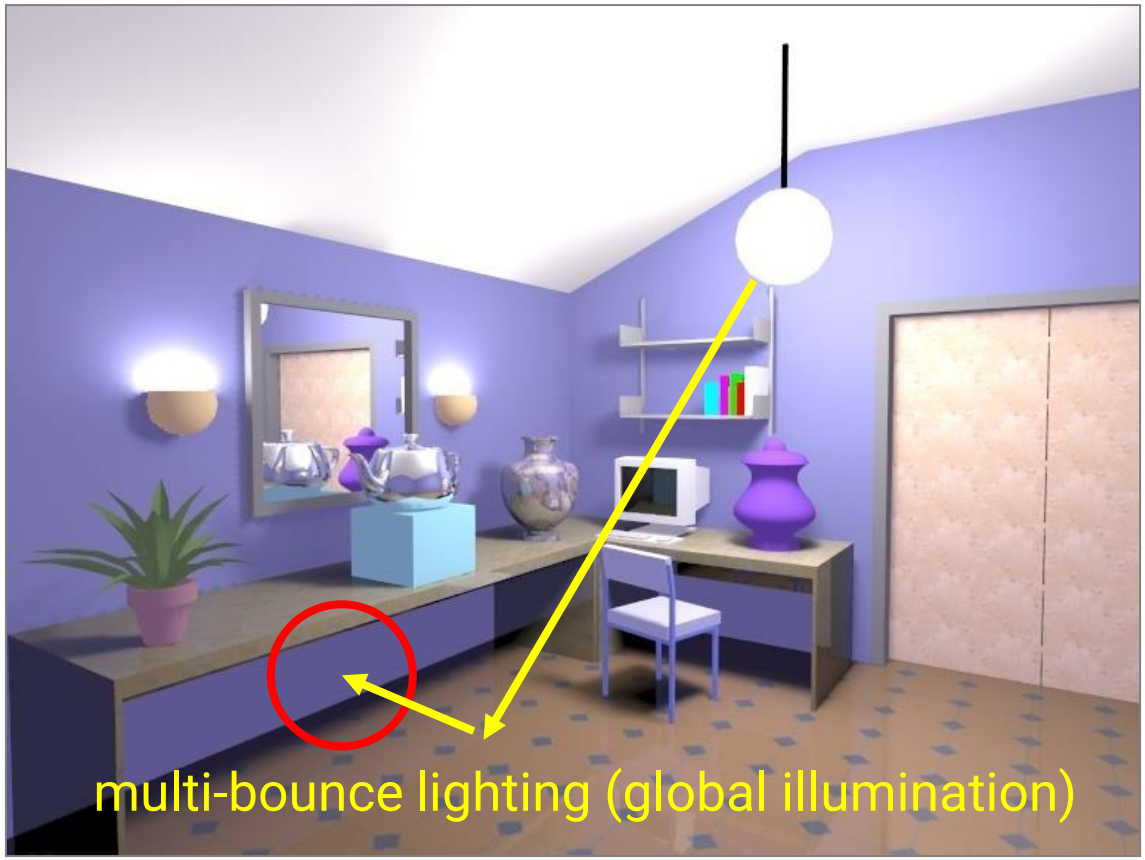
Description of a 3D World (cont.)

- Simulate more realistic materials and lighting phenomena



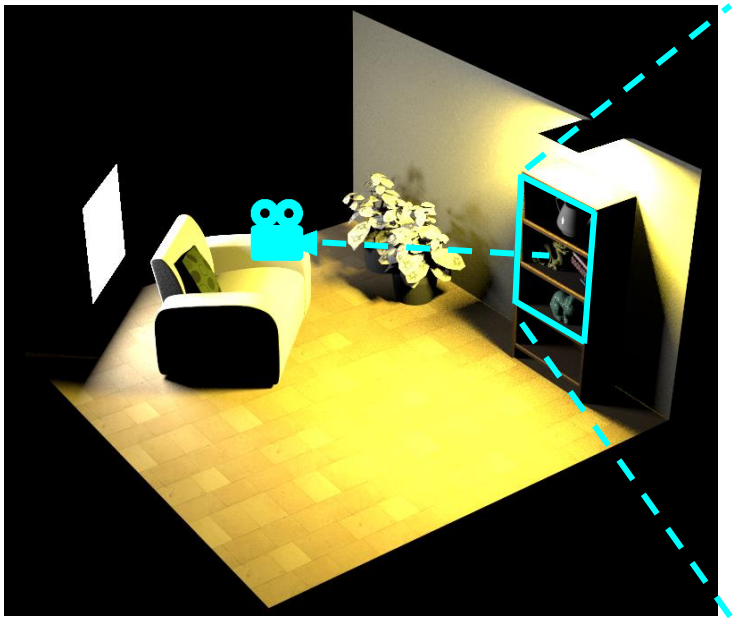
Description of a 3D World (cont.)

- Simulate more complex **light paths**



Generate Images from the 3D World

- Most displays are 2D, so we need to generate images from the 3D world
- Just like taking a picture with a camera in our daily lives
  - But with a **virtual camera** and a **virtual film**



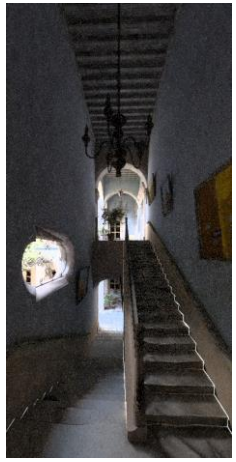
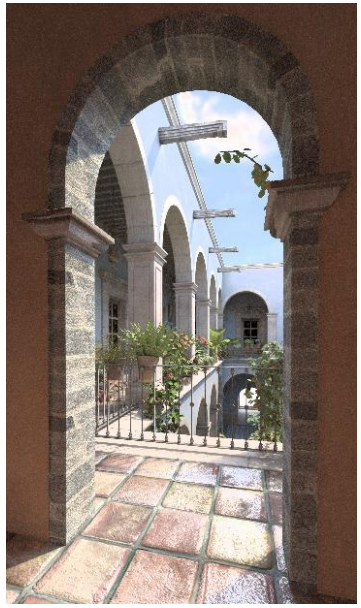
3D virtual world

rendered image

Generate Images from the 3D World (cont.)





Are These 3D?


あきら
たかし
あーちゃん
たかし
あきら
ケンジ
あーちゃん


移動
魔法
攻撃
撤退
道具
戦技
HP 4209
EP 160
CP 90
HP 3934
EP 160
CP 90
HP 7240
EP 227
CP 200
HP 7782
EP 304
CP 200
SBREAK X


小鱼儿
好，全给我包起来吧！

The Differences between Relevant Fields

- Traditionally we will categorize ***computer graphics***, ***computer vision***, and ***image processing*** by their inputs and outputs:

inputs \ outputs	descriptions	images
descriptions		***computer graphics***
images	***computer vision***	***Image processing***

- However, the gaps are much vaguer now!

Major Subfields of Computer Graphics

Modeling
How to create and store geometry objects

Rendering
How to generate images of geometry objects

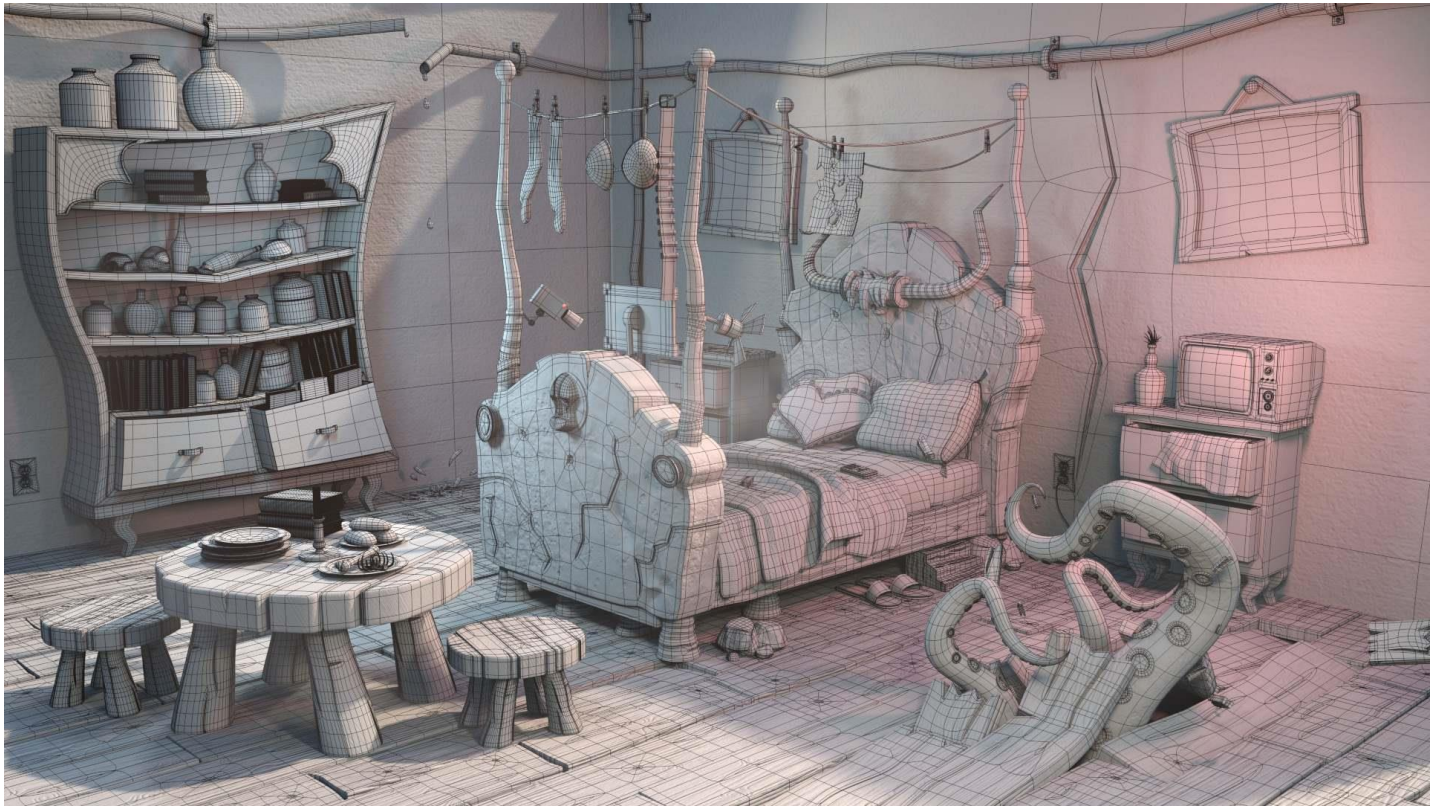
Animation
How to manipulate geometry objects

Outline

- Overview
- **Modeling**
- Animation
- Rendering
- Film production pipeline
- Ray tracing and rasterization

Modeling

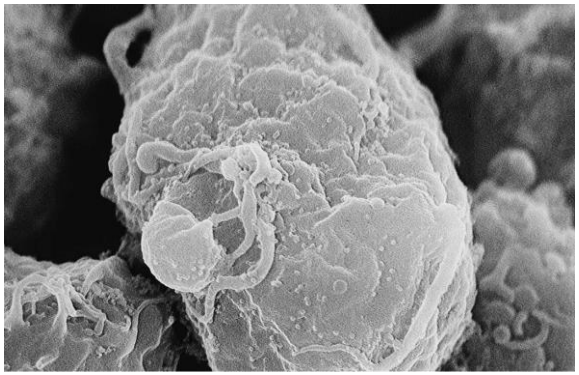
- Build 3D representation of the virtual world
- The process of generating “data” in computer graphics



Modeling (cont.)

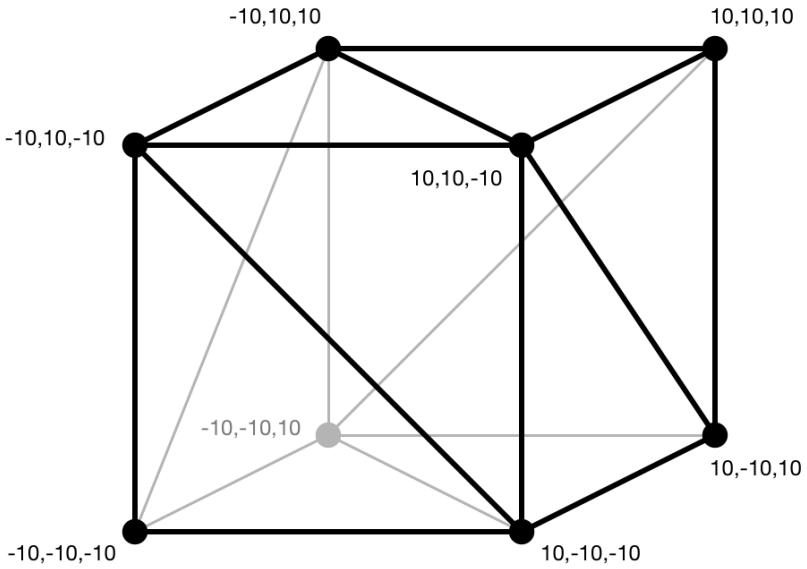
- World geometries are diverse!






Triangle Mesh

- The most popular way to represent the 3D geometry

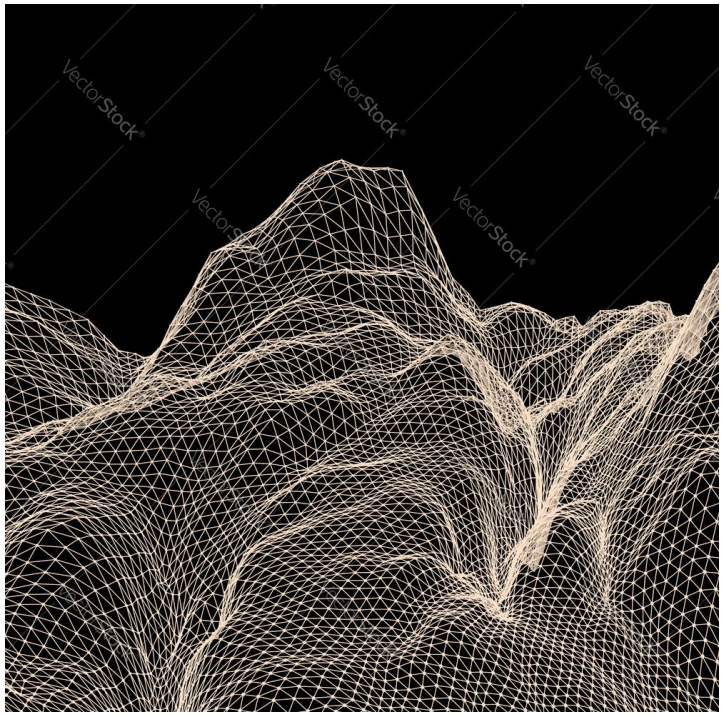


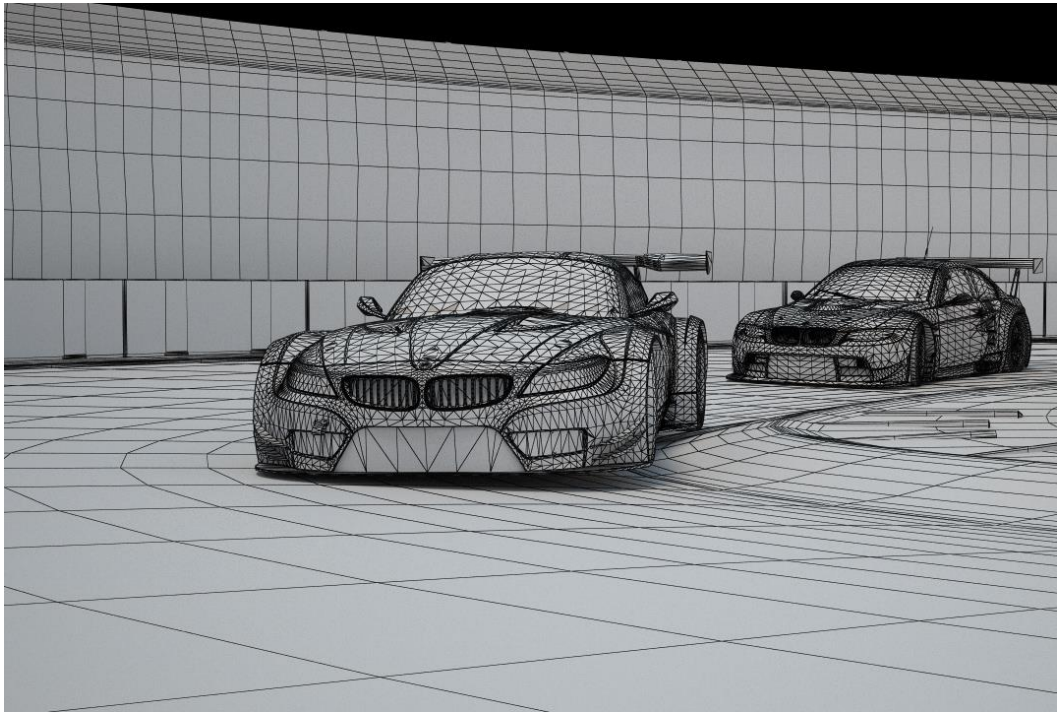
12 triangles

10K triangles

Triangle Mesh (cont.)

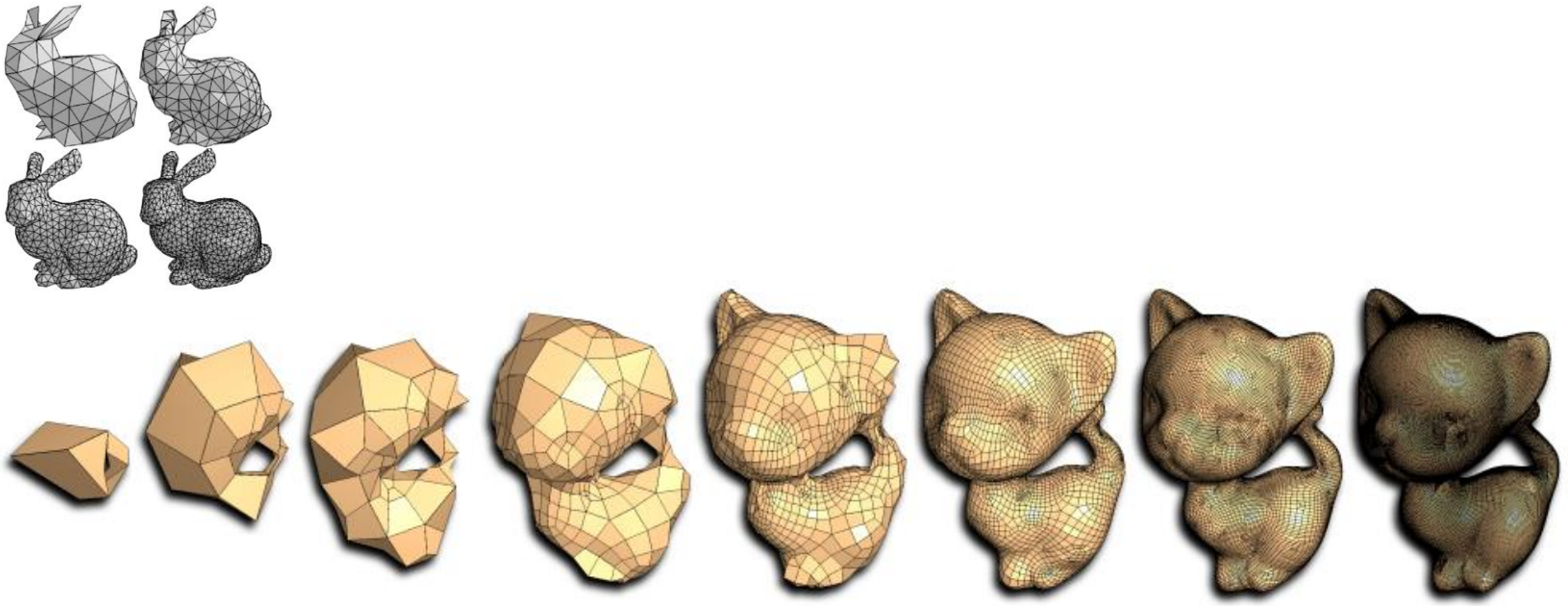
- The most popular way to represent the 3D geometry
  - Most objects can be modeled by triangles
  - **All vertices of a triangle lie on the same plane**



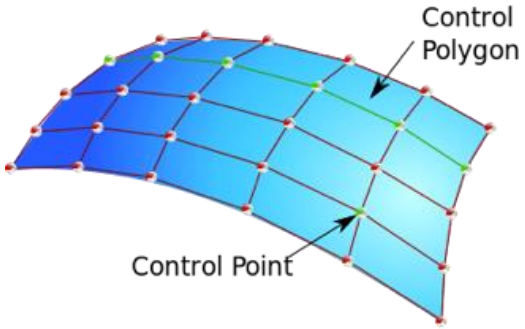


Triangle Mesh (cont.)

- The most popular way to represent the 3D geometry
  - Most objects can be modeled by triangles
  - All vertices of a triangle lie on the same plane
  - More triangles: more fidelity but also more expensive!



Other Geometry Representation



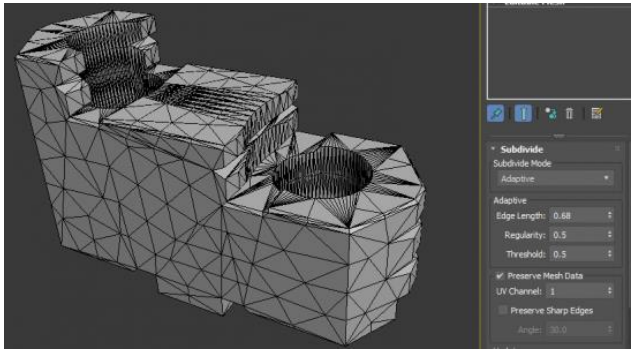
patches

curves

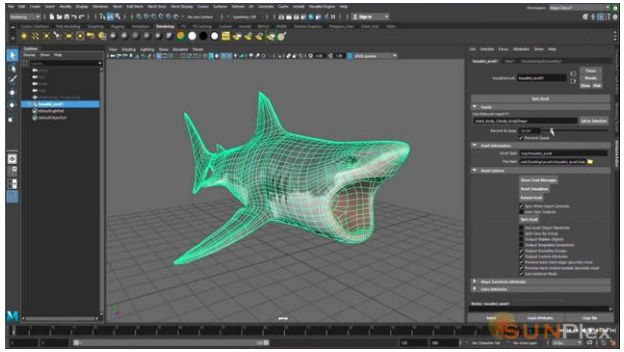
volume data

Obtain 3D Models

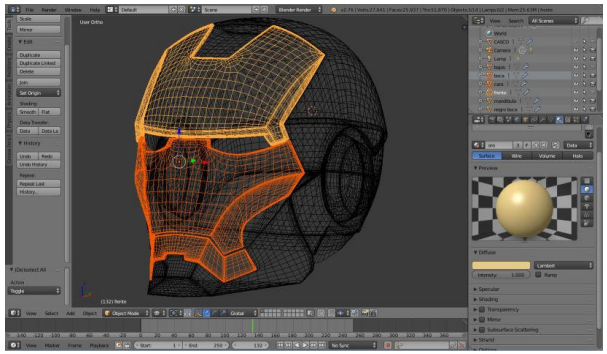
- 3D models are usually obtained by professional manipulations in 3D modeling tools
- Popular 3D editing software



Blender



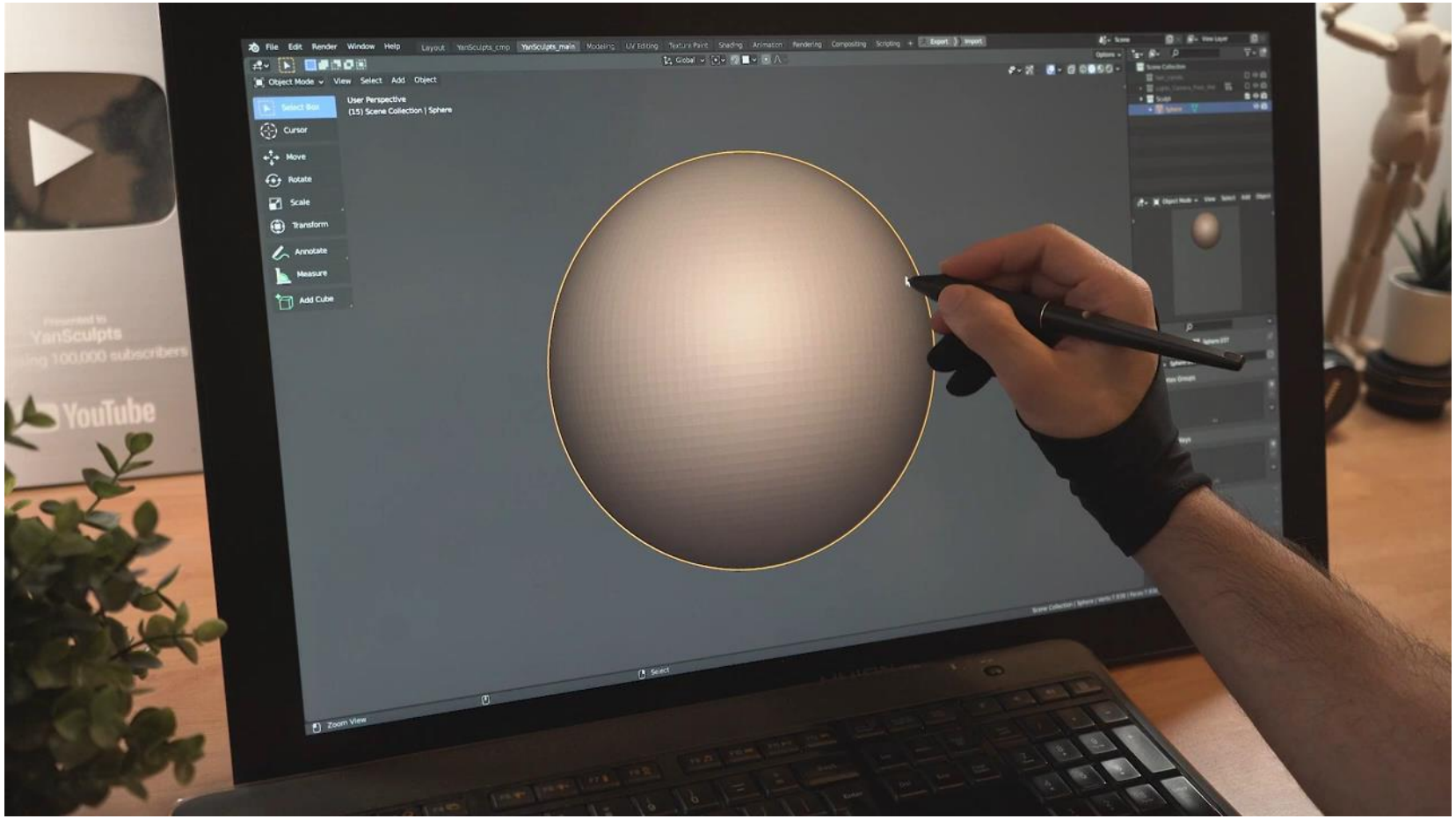
Maya



AUTODESK 3DS MAX

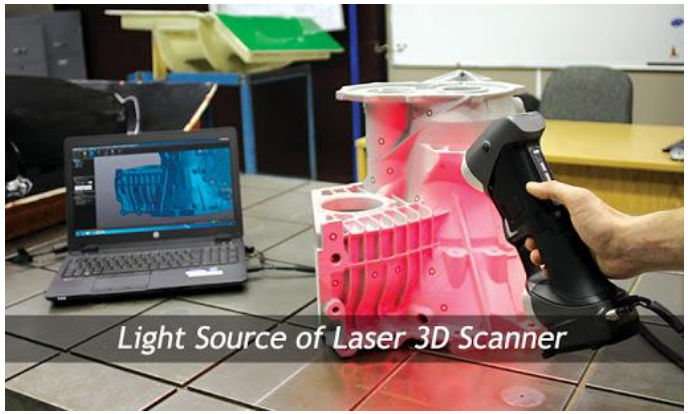
Obtain 3D Models (cont.)

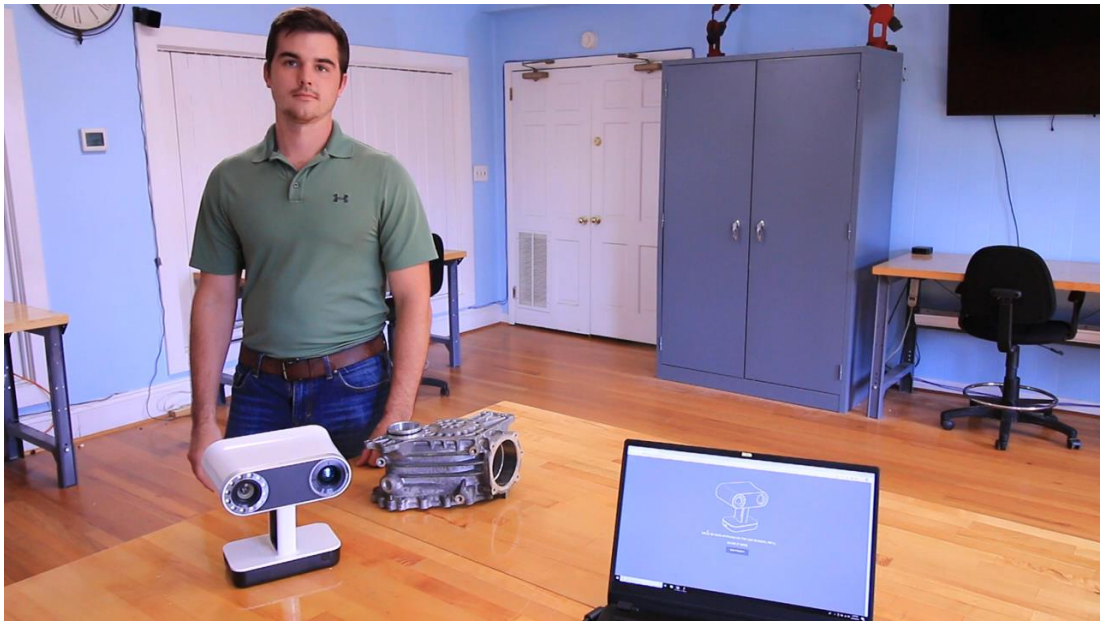
- Example: create a 3D character model in Blender [Link]



Obtain 3D Models (cont.)

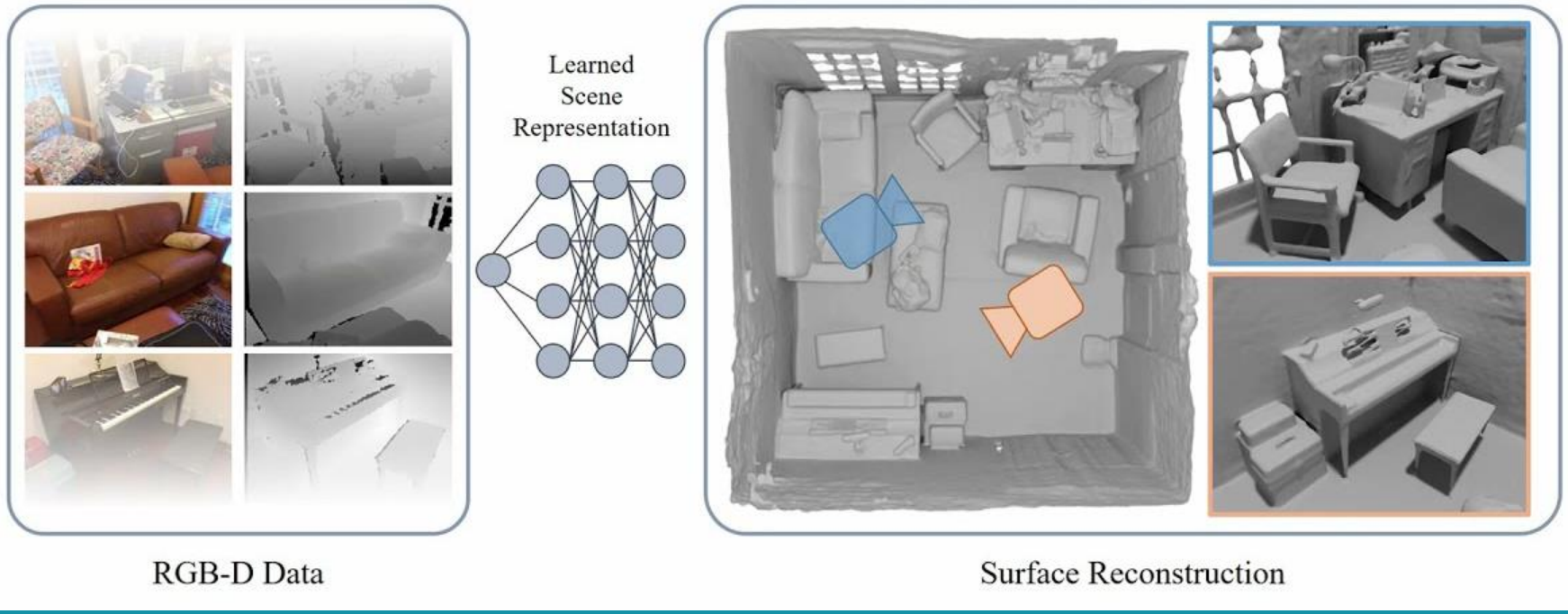
- Alternative: capture the real-world geometries with special hardware devices (cameras)





Obtain 3D Models (cont.)

- Alternative: capture the real-world geometries with special hardware devices (cameras)

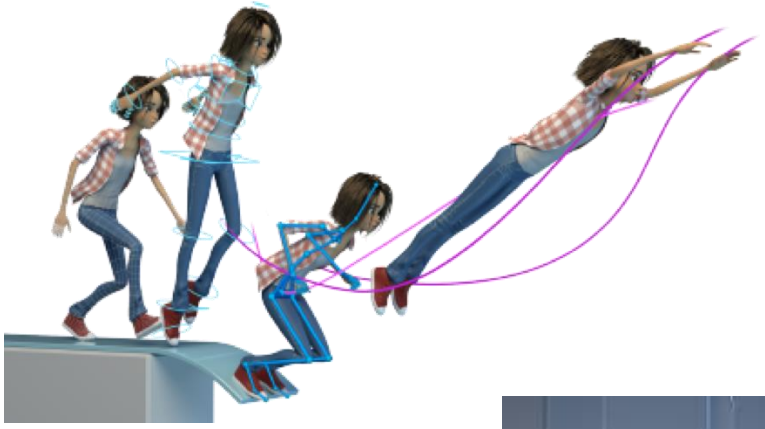


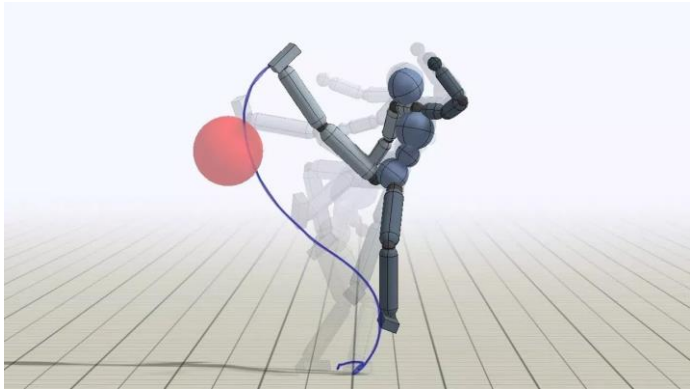
Outline

- Overview
- Modeling
- **Animation**
- Rendering
- Film production pipeline
- Ray tracing and rasterization

Animation

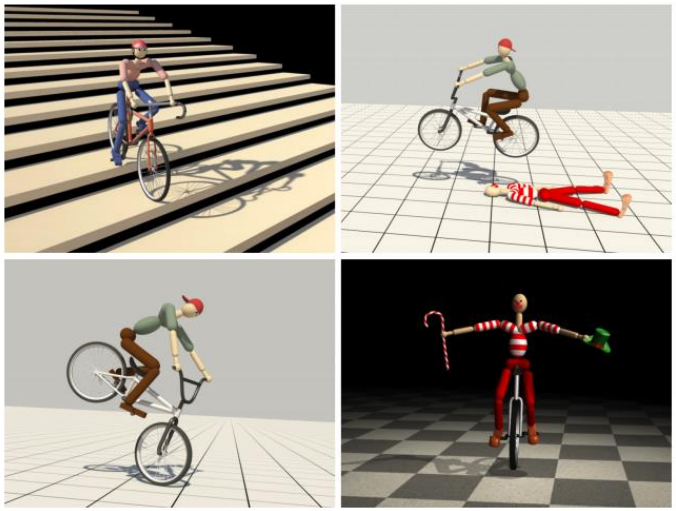
- How do the geometry change / move over time



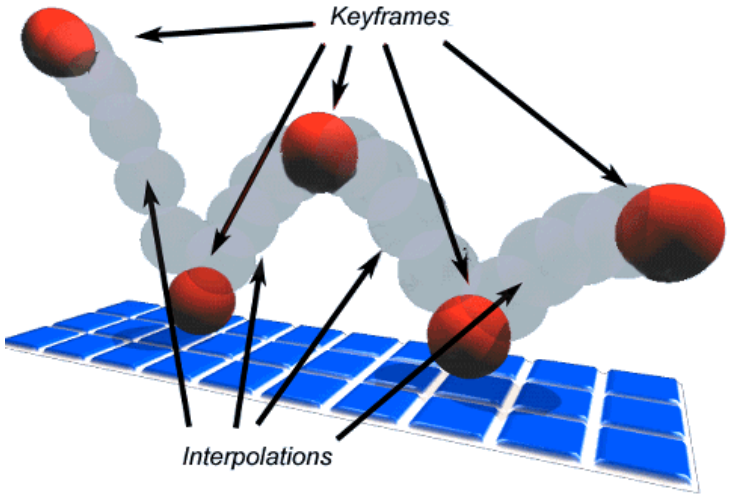


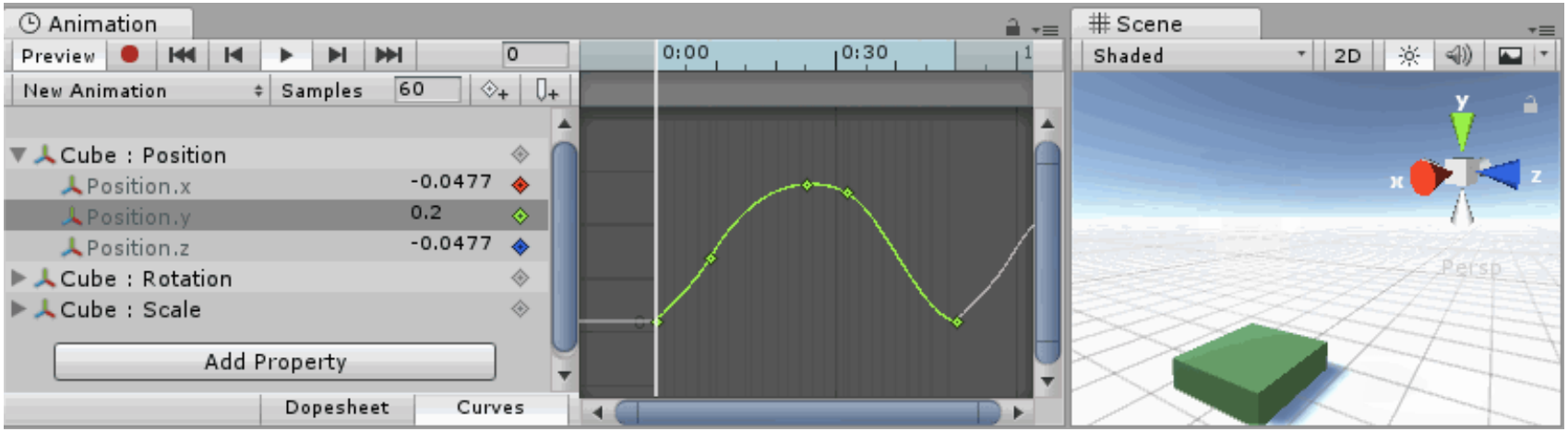
Physically-Based Animation

T-shirt Dressing



Making Animation: Keyframe Animation





Making Animation: Keyframe Animation

- 1 Minute example:
  https://www.youtube.com/watch?v=TjJLIuFKA20

Capturing Motion

- Motion capture

Motion Capture (cont.)

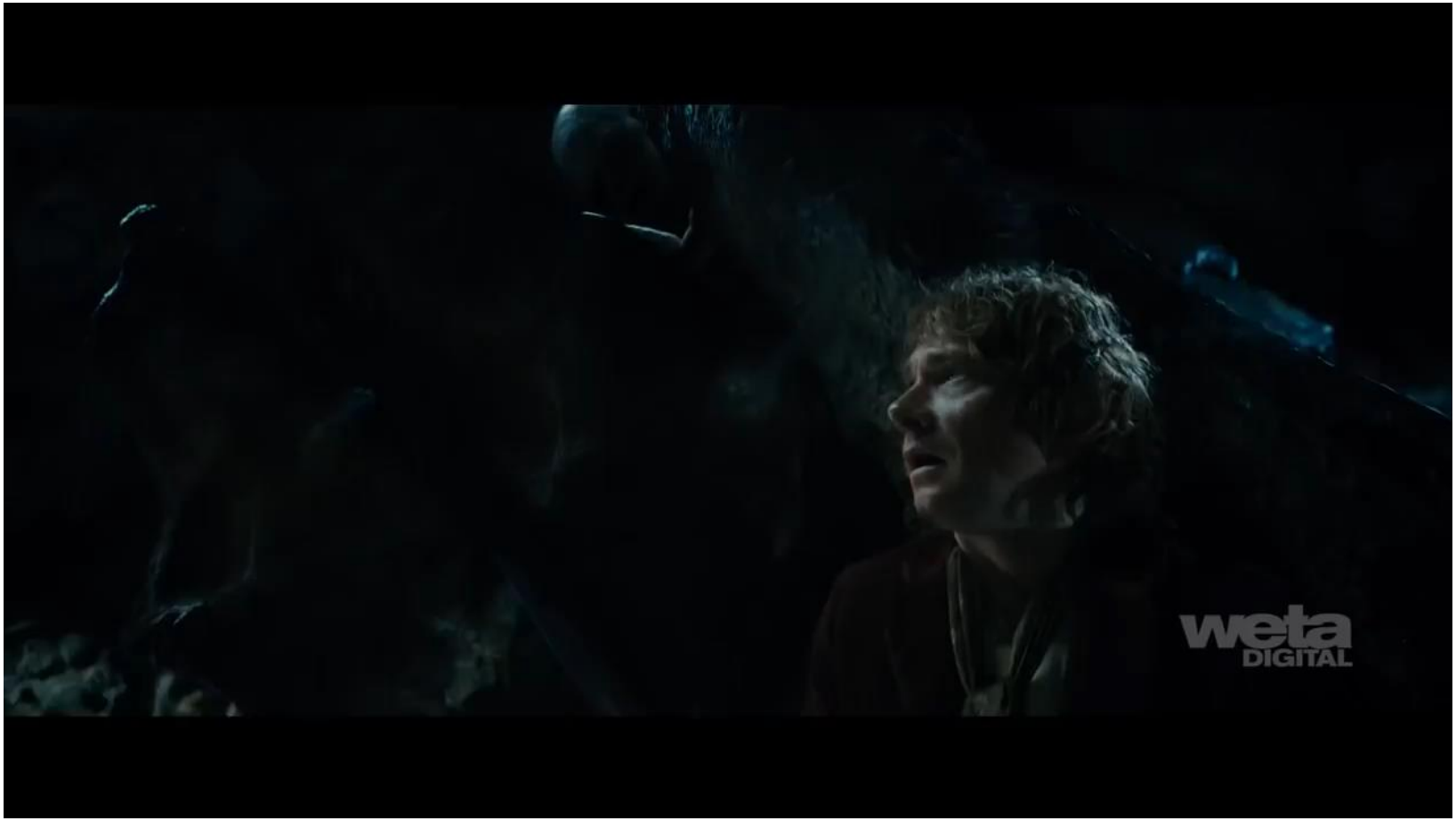
- Example: motion capture in game production

Facial Animation Capture

- Facial capture

Facial Animation Capture (cont.)

- Example: facial capture in movie production

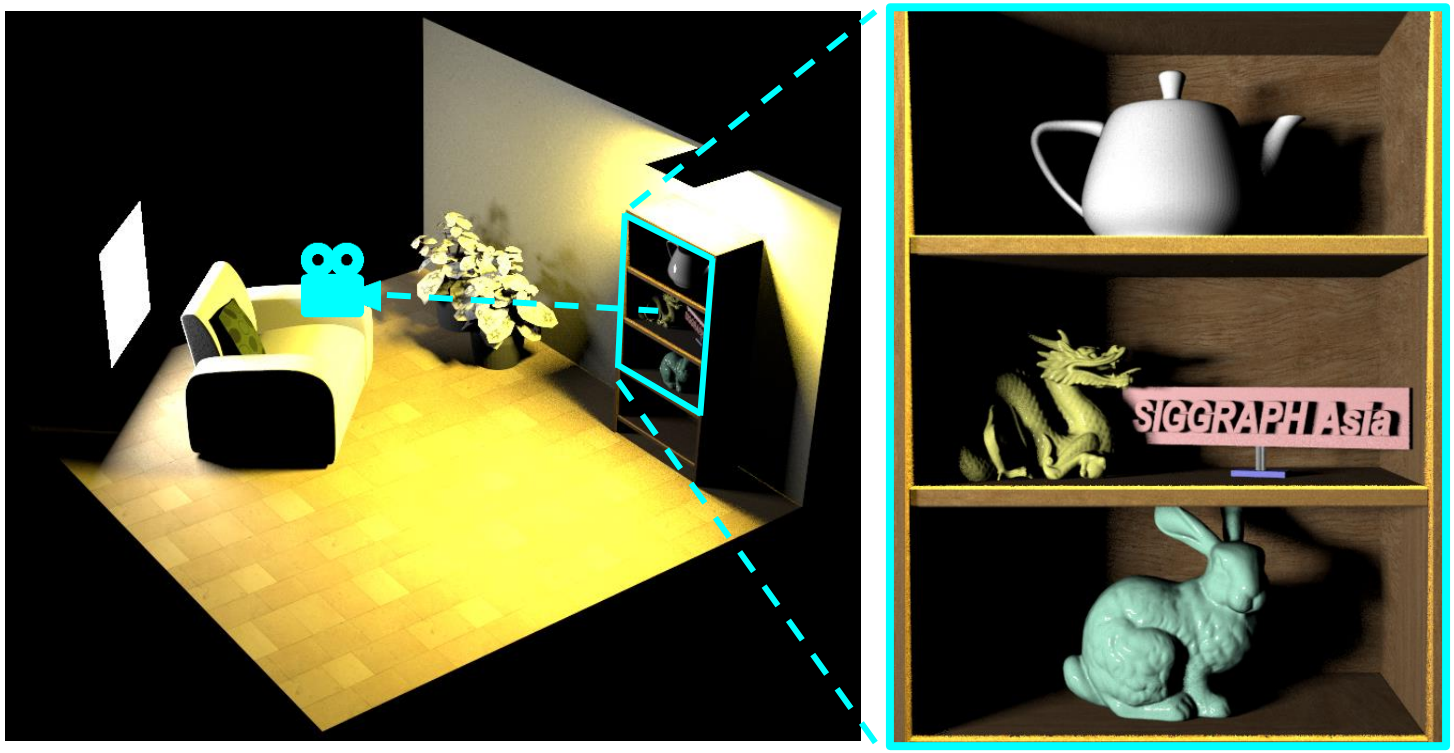


Outline

- Overview
- Modeling
- Animation
- **Rendering**
- Film production pipeline
- Ray tracing and rasterization

Rendering

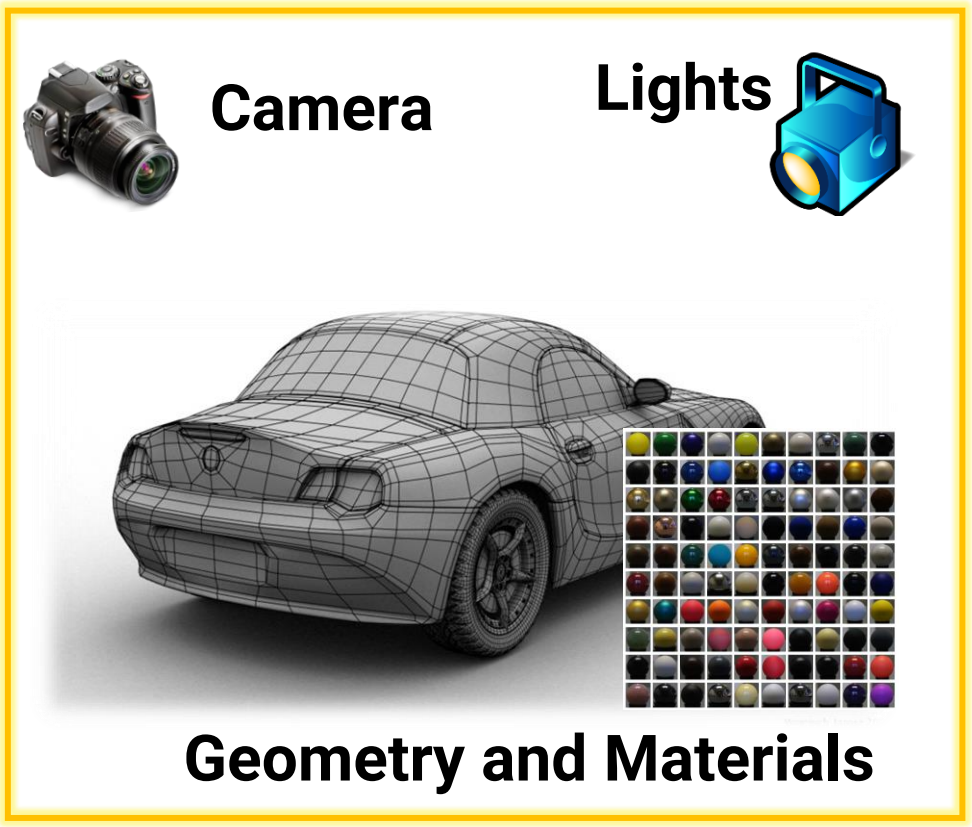
- How do we model appearance and perceive things



3D virtual world rendered image

Rendering (cont.)

- Generate a 2D image from a 3D world description

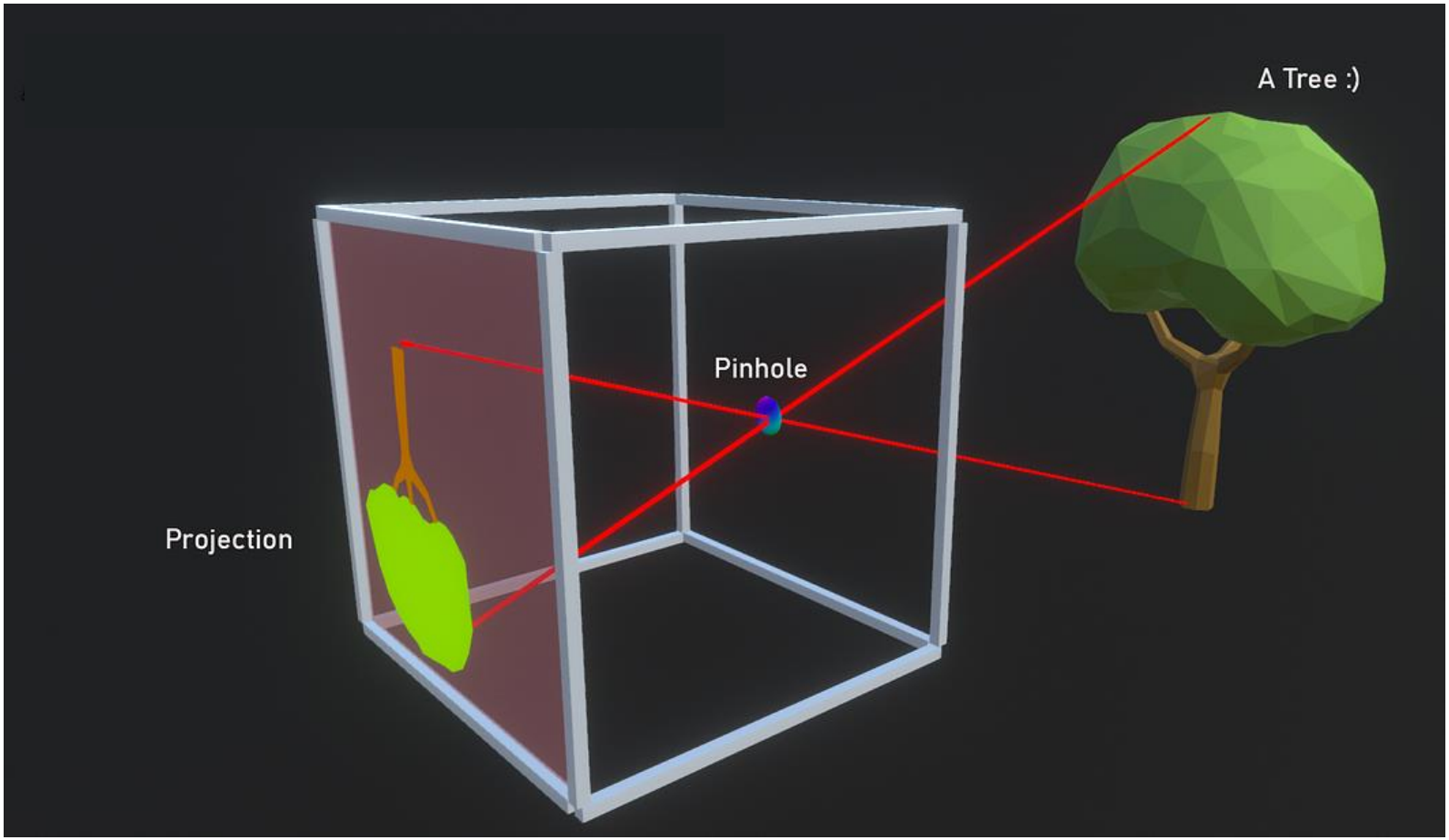


input: 3D description of a scene

output: 2D synthetic image

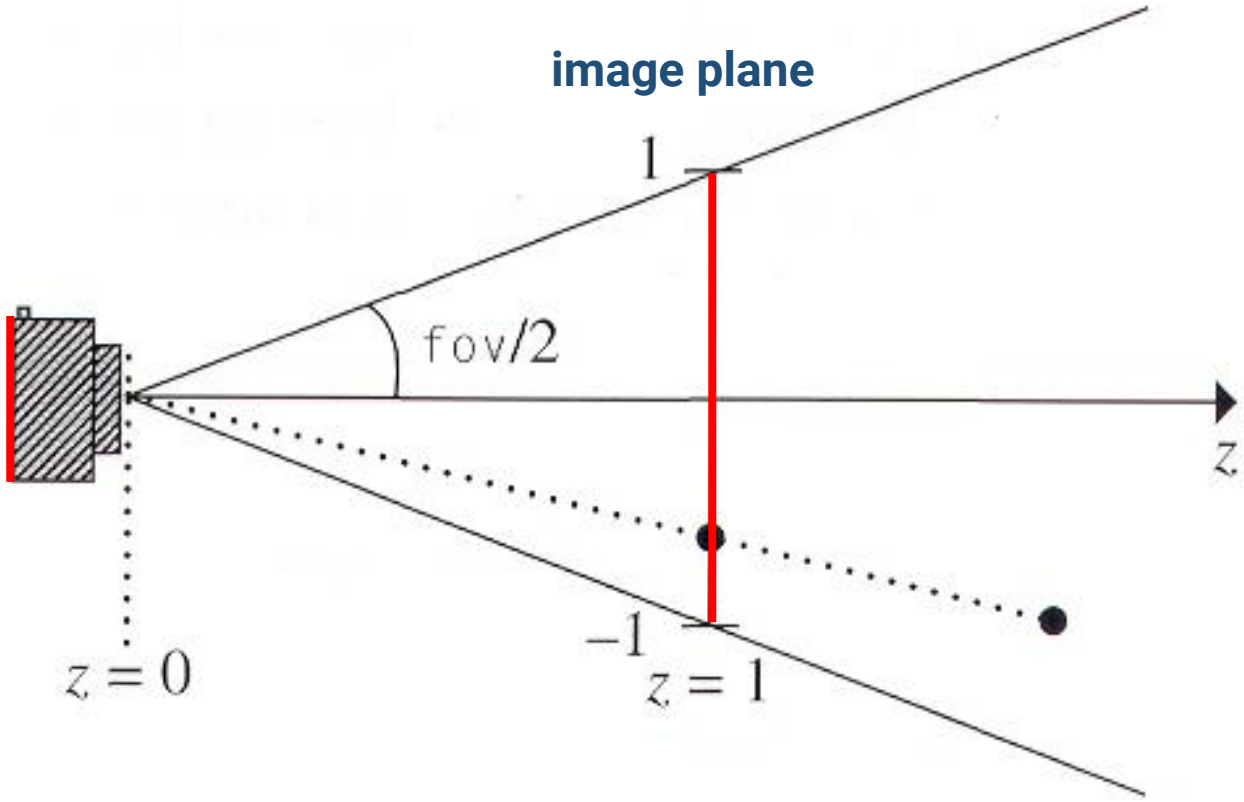
Rendering (cont.)

- Rendering with **perspective pinhole camera**



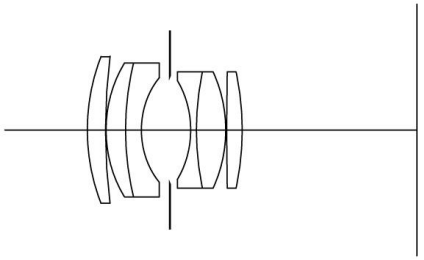
Rendering (cont.)

- **Perspective pinhole camera** in graphics
  - The image plane is put in front of the pinhole to avoid an up-side-down image



Rendering (cont.)

- Lens camera for simulating depth of field

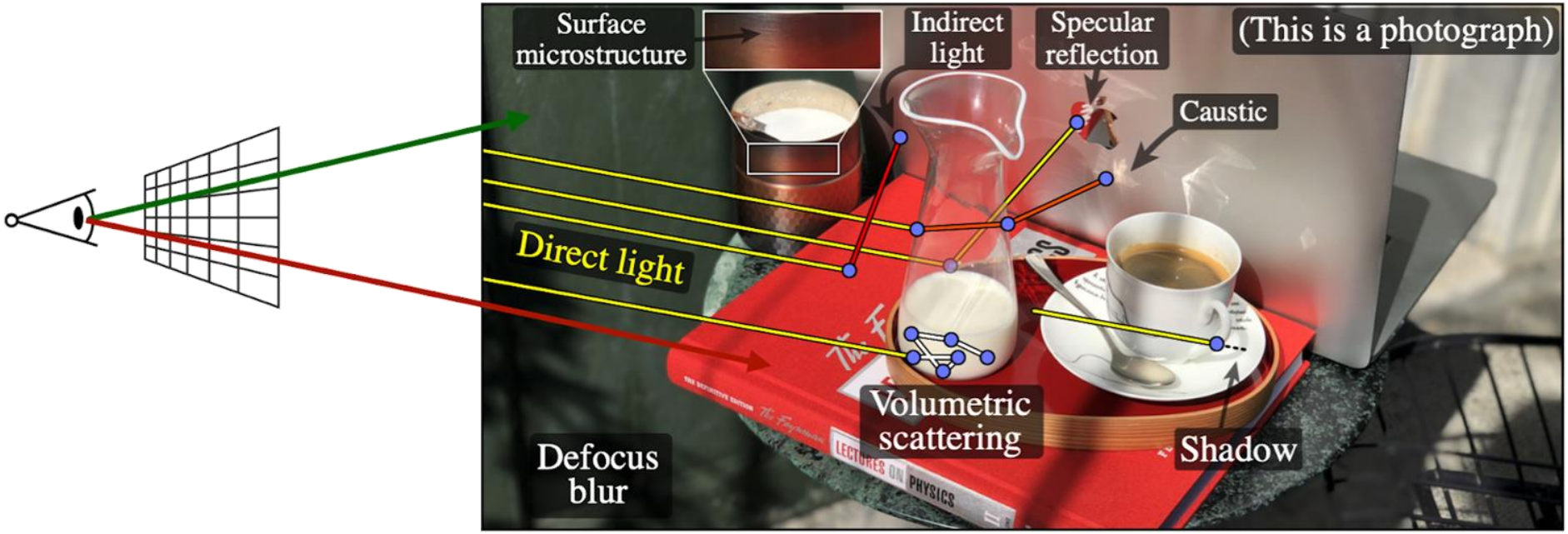


Rendering (cont.)

- Simulation of motion blur

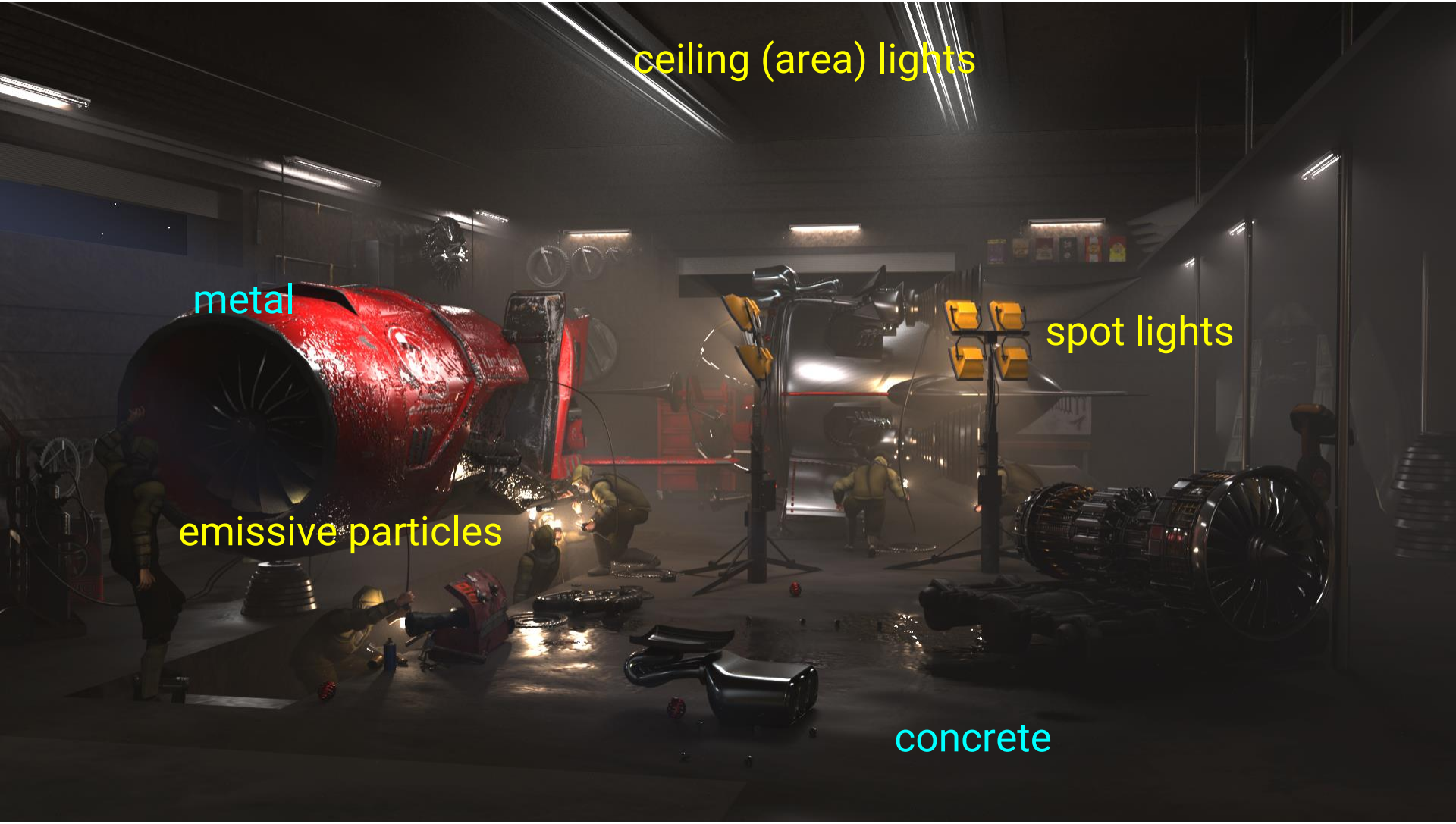
Physically-Based Rendering (PBR)

- Uses **physics** and **math** to simulate the interaction between matter and lights
- **Realism** is the primary goal



Non-Photo-Realistic Rendering (NPR)

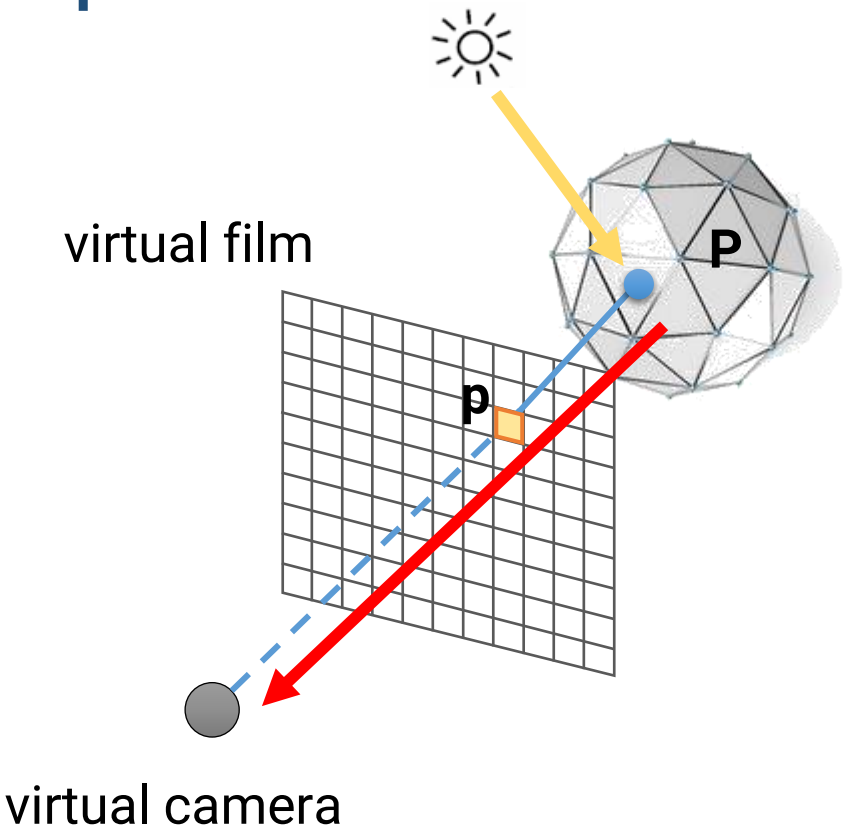
Shading: Materials and Lighting

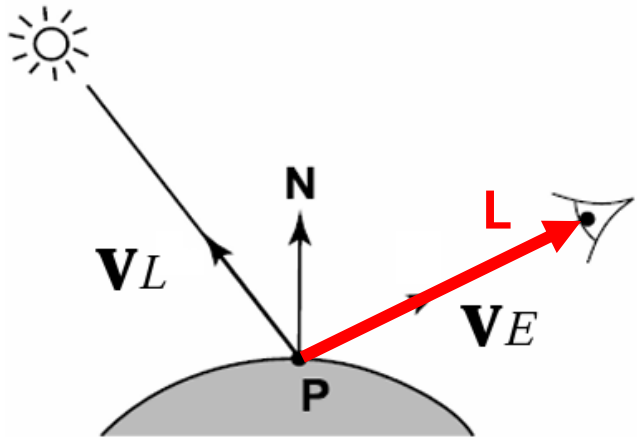


Shading: Materials and Lighting (cont.)

Shading

- Simulate the **interaction** between a **light** and a **surface point**





Point **P** on a surface through a pixel **p**

Normal **N** at **P**

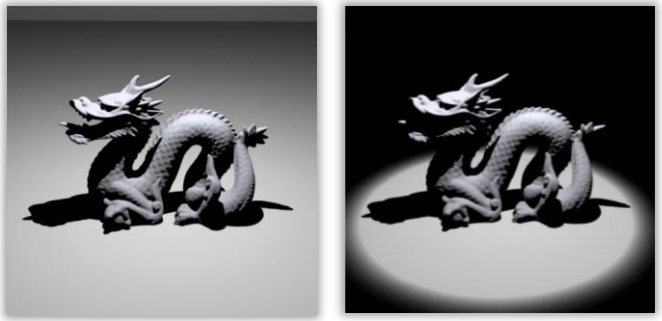
Lighting direction **V***L*

Viewing direction **V***E*

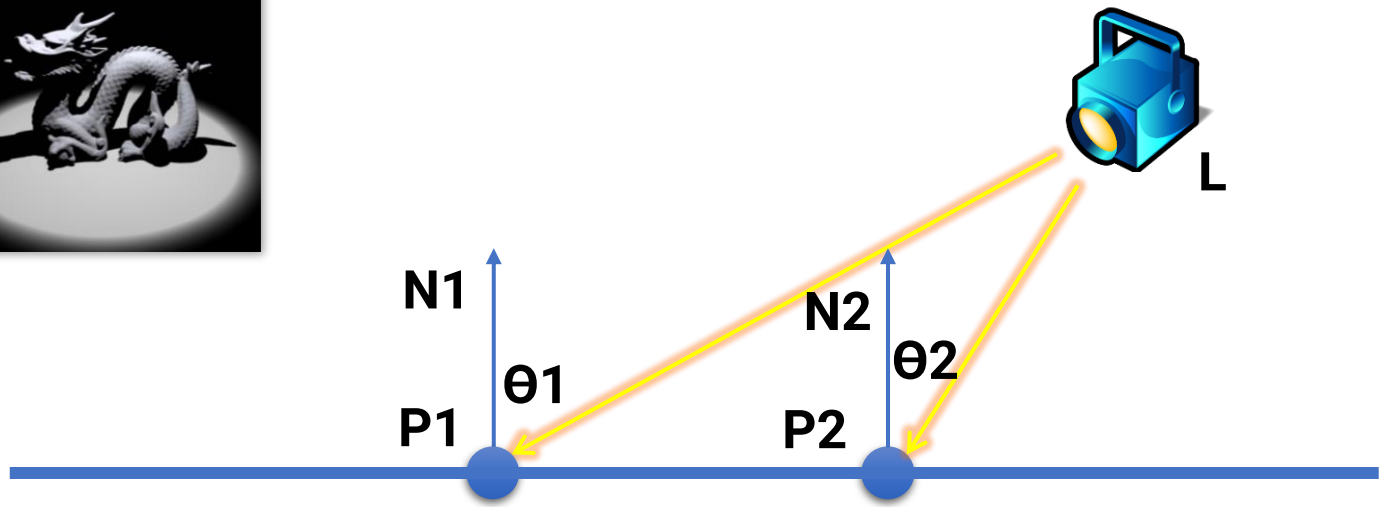
Goal: compute color **L** for pixel **p**

Local Lights

- The distance between a light and a surface is **NOT** long enough compared to the scene scale
- The position of a light need to be taken into account during shading
  - **Lighting direction = |L – P|**

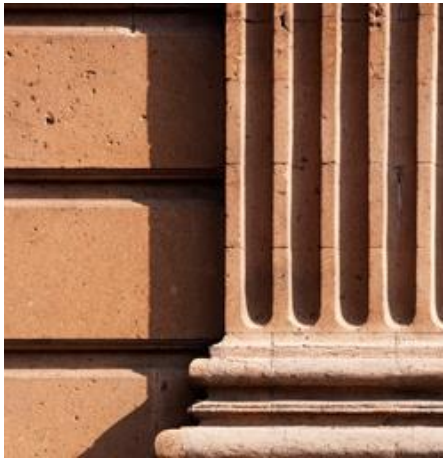


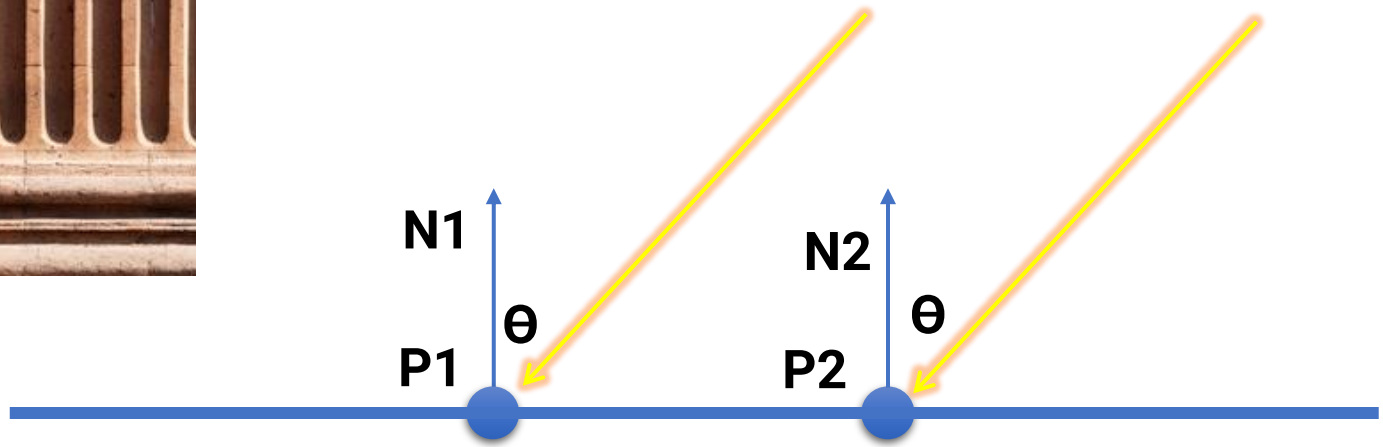
point/spot light



Distant Lights

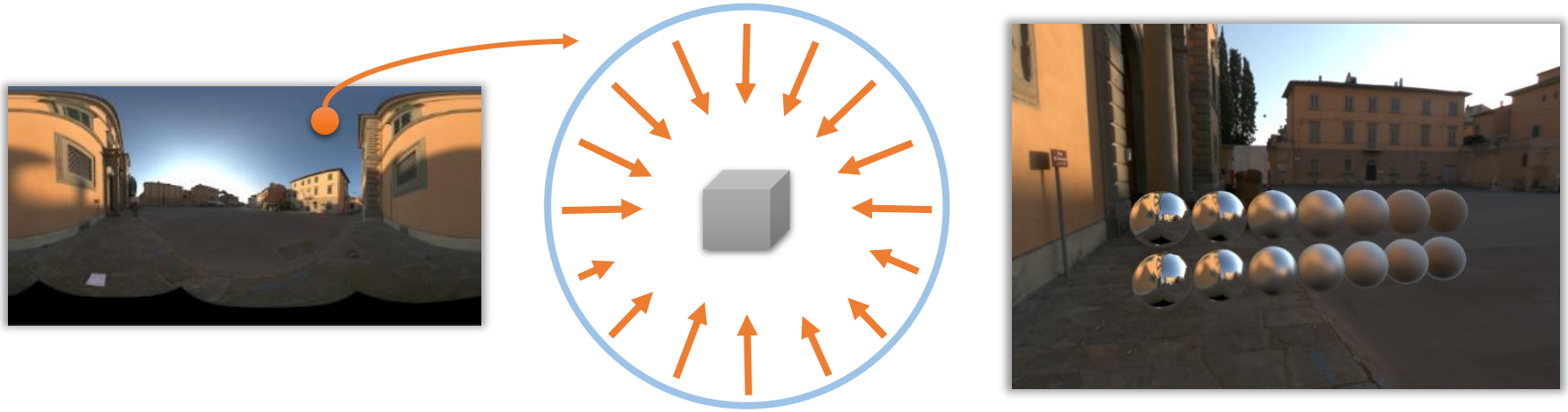
- The distance between a light and a surface is long enough compared to the scene scale and **can be ignored**
  - **Lighting direction** is **fixed**
- **Directional light (sun)** is the most common distant light





Environment Light

- Environment light illuminates the scene from a **virtual sphere at infinite distance**
- The spherical energy distribution is usually represented with longitude-latitude images
- Also called **image-based lighting (IBL)**

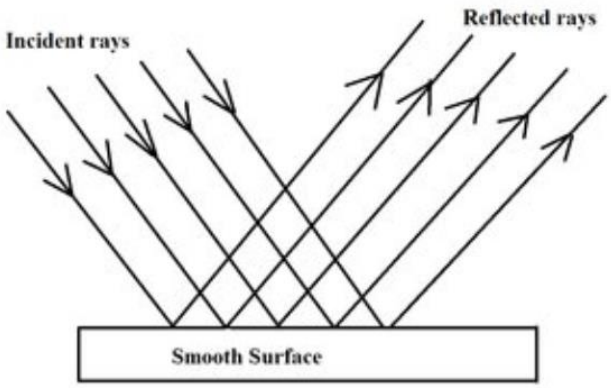


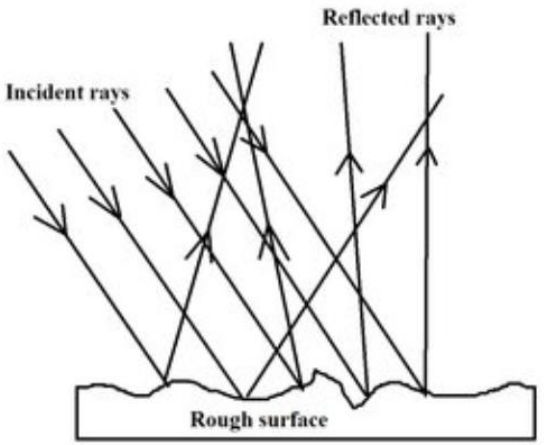
Environment Light (cont.)

- Widely used in digital visual effects and film production

Materials

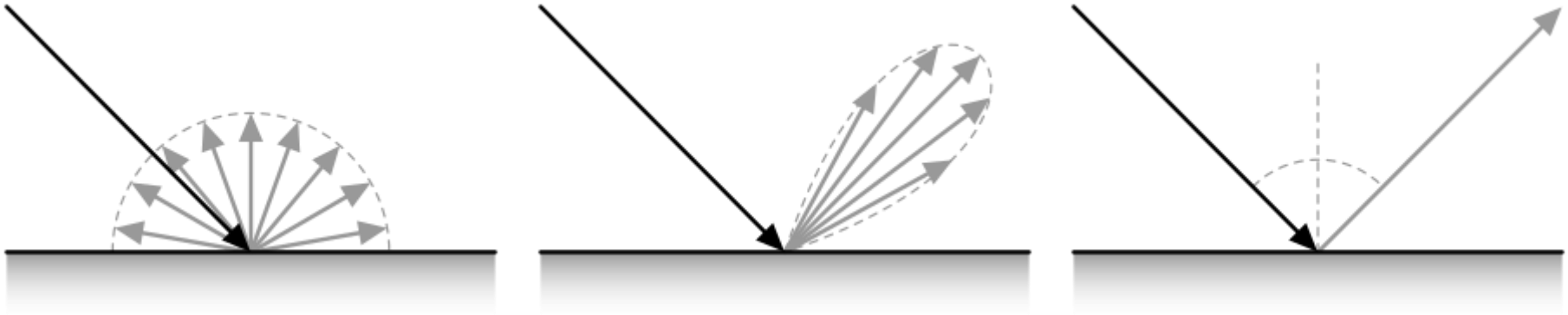
- Highly related to surface types
- The **smoother** a surface, the more reflected light is concentrated in the direction a **perfect mirror** would reflect the light





Materials (cont.)

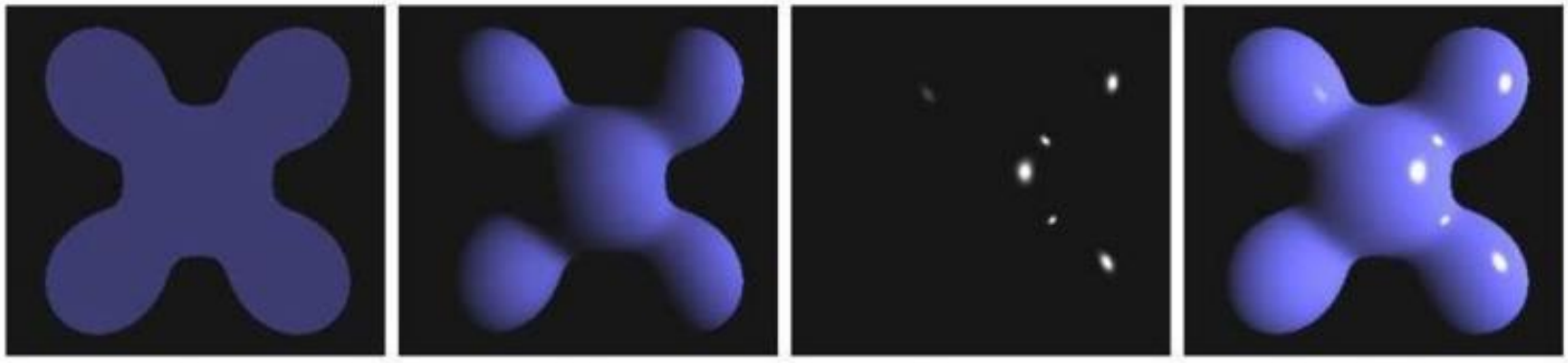
- Highly related to surface types
- The **smoother** a surface, the more reflected light is concentrated in the direction a **perfect mirror** would reflect the light



diffuse glossy specular

Phong Lighting Model

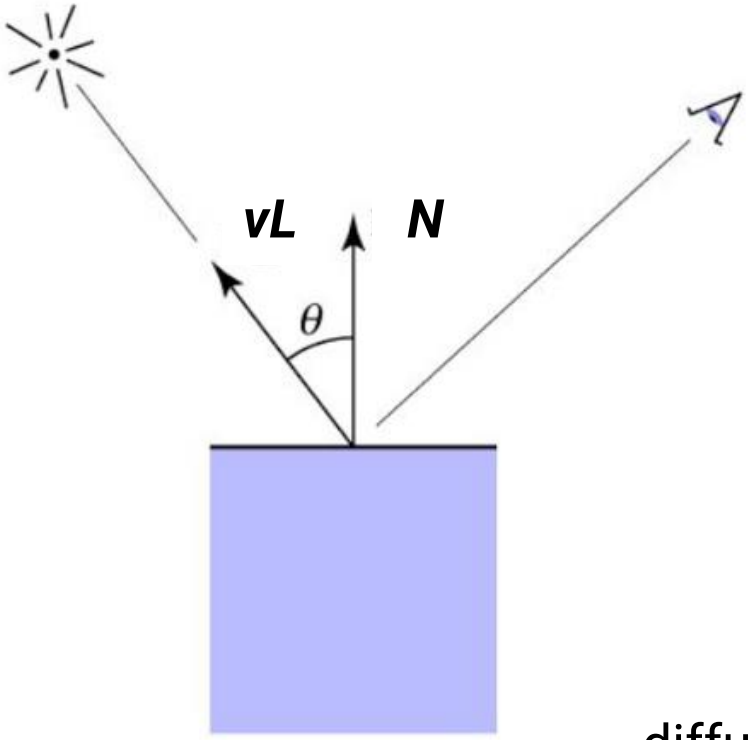
- **Diffuse reflection**
  - Light goes everywhere; colored by object color
- **Specular reflection**
  - Happens only near mirror configuration; usually white
- **Ambient reflection**
  - Constant accounted for global illumination (cheap hack)



ambient diffuse specular

Diffuse Shading (cont.)

- Applies to diffuse or matte surface



illumination from source

$$L_d = k_d \cdot I \cdot \max(0, N \cdot vL)$$

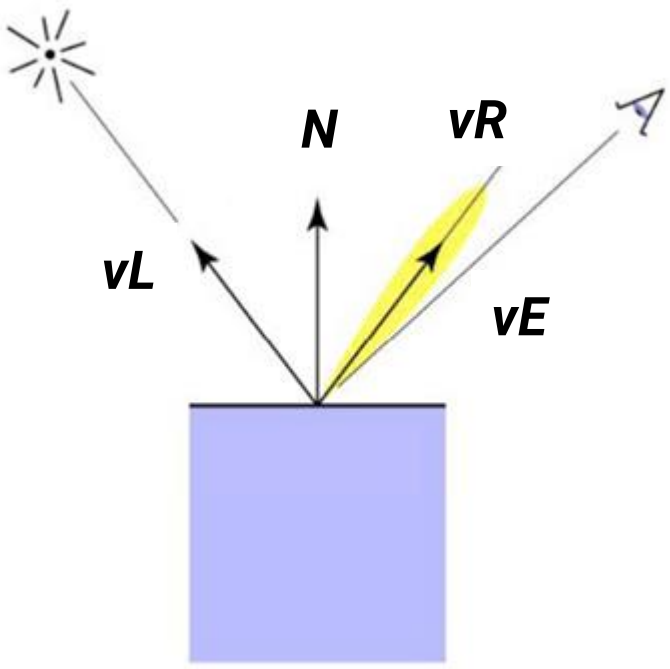
Lambertian law

diffuse coefficient

diffusely reflected light

Specular Shading (cont.)

- **Phong specular model [1975]**
  - Fall off gradually from the perfect reflection direction



$$vR = vL + 2((N \cdot vL)N - vL)$$
$$= 2(N \cdot vL)N - vL$$

specular exponent

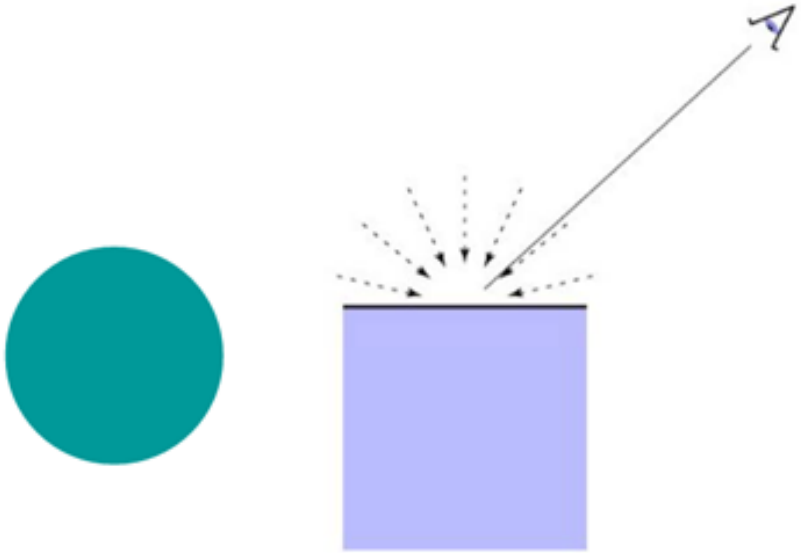
$$L_s = k_s \cdot I \cdot \max(0, \cos\sigma)^n$$
$$= k_s \cdot I \cdot \max(0, vE \cdot vR)^n$$

specular coefficient

specularly reflected light

Ambient Shading

- Add constant color to account for disregarded illumination and fill black shadows



the **intensity** of ambient light

$$L_a = k_a \cdot I_a$$

ambient coefficient

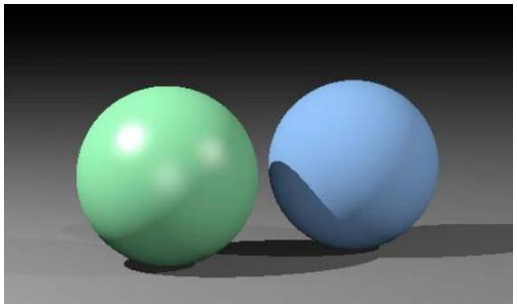
reflected ambient light

Complete Phong Lighting Model

- Compute the contribution from a light to a point by including **ambient**, **diffuse**, and **specular** components

$$L = L_a + L_d + L_s$$

- Consider the RGB channels for color



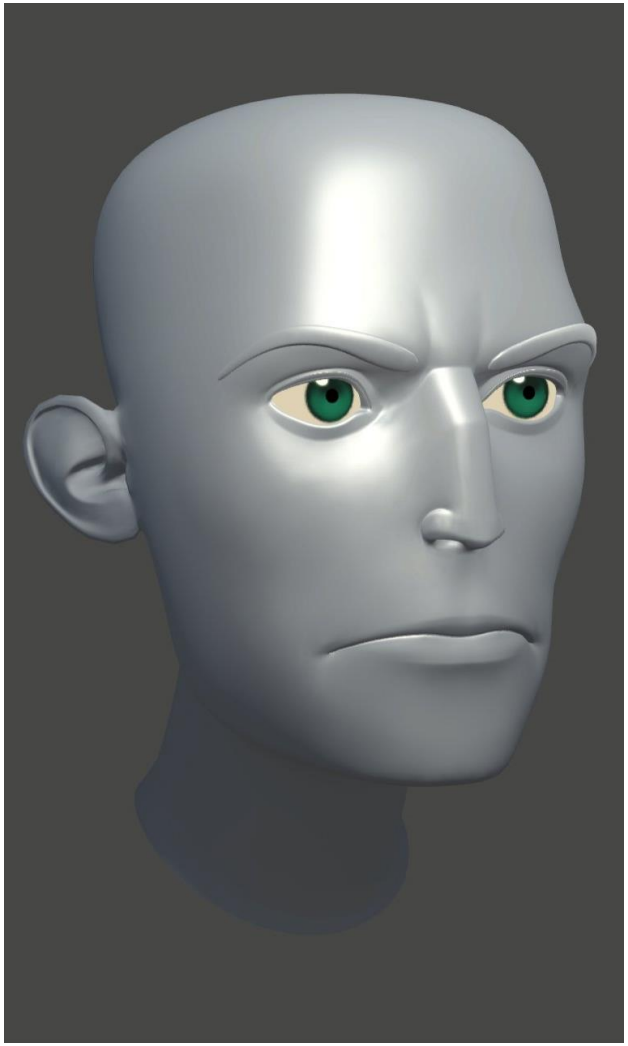
white light: (0.9, 0.9, 0.9)
yellow light: (0.8, 0.8, 0.2)

$$L_d = k_d \cdot I \cdot \max(0, N \cdot vL)$$

green ball: (0.2, 0.7, 0.2)
blue ball: (0.2, 0.2, 0.7)

- If there are multiple lights, just sum over all the lights because the lighting is **linear**

Some Results with Phong Lighting Model

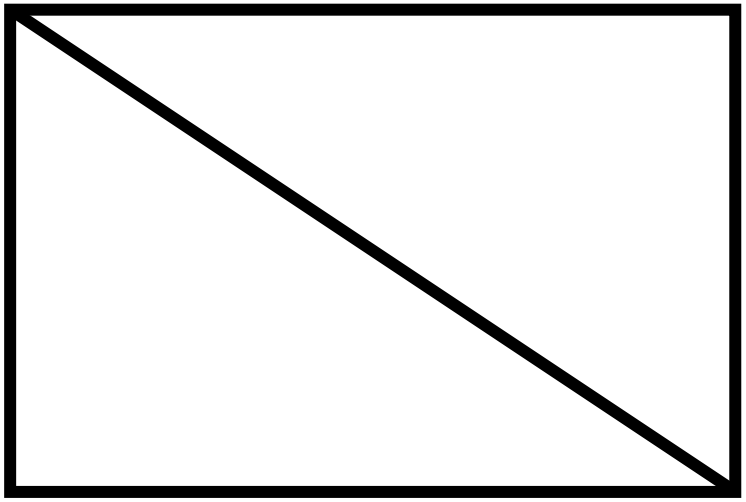


Texture

- Consider the following cases
  - Do we need (or can we) to finely subdivide the object?

Textures

- Can be used to represent **spatially-varying** data
- Can **decouple** materials from the geometry



Geometry: two triangles
Material: Kd(1, 1, 1)

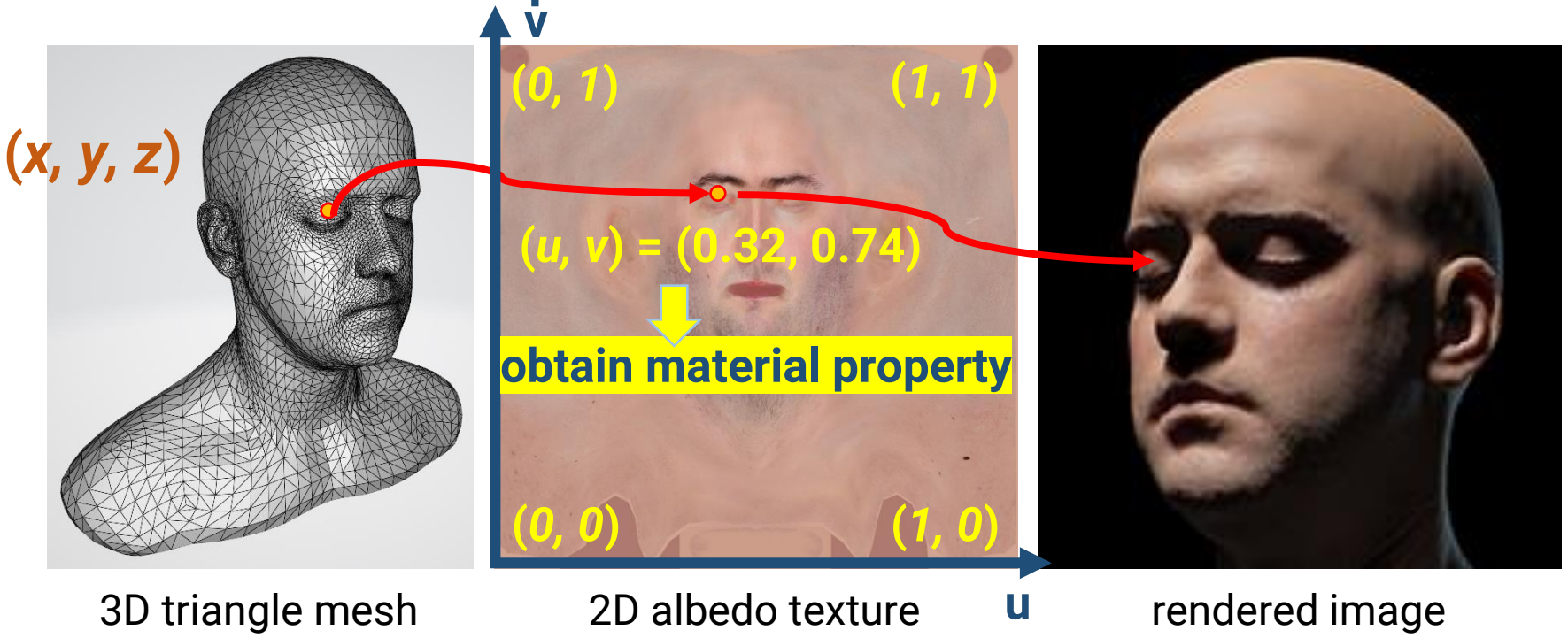
+

2D image texture

= **complex appearance**

Texture Coordinate

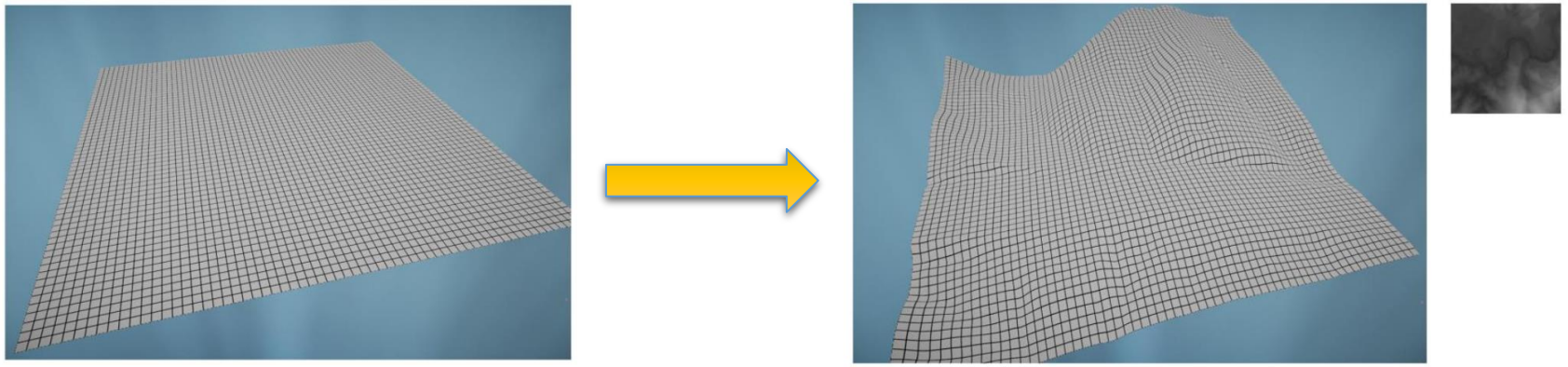
- A coordinate to look up the texture
- The way to map a point on an **arbitrary 3D surface** to a pixel (texel) on an **image** texture
  - Need **surface parameterization**



Texture Coordinate (cont.)

- A coordinate to look up the texture
- The way to map a point on an **arbitrary 3D surface** to a pixel (texel) on an **image** texture
  - Need **surface parameterization**
  - Usually produced by 3D artists

More Texture Types



Outline

- Overview
- Modeling
- Animation
- Rendering
- **Film production pipeline**
- Ray tracing and rasterization

Animation Production Pipeline

story

text treatment

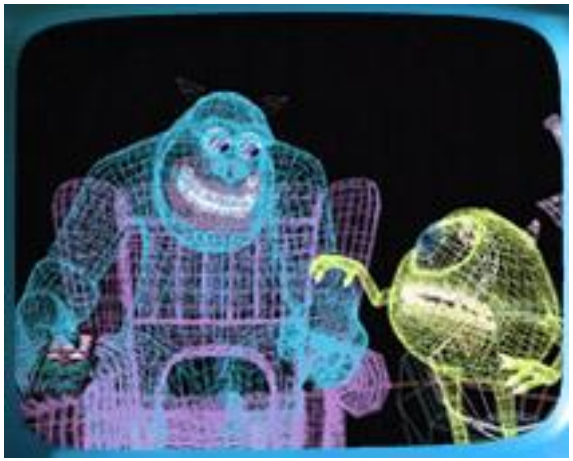
storyboard

voice

storyreel

look and feel

Animation Production Pipeline (cont.)



modeling / articulation

layout

animation

shading / lighting

rendering

final touch

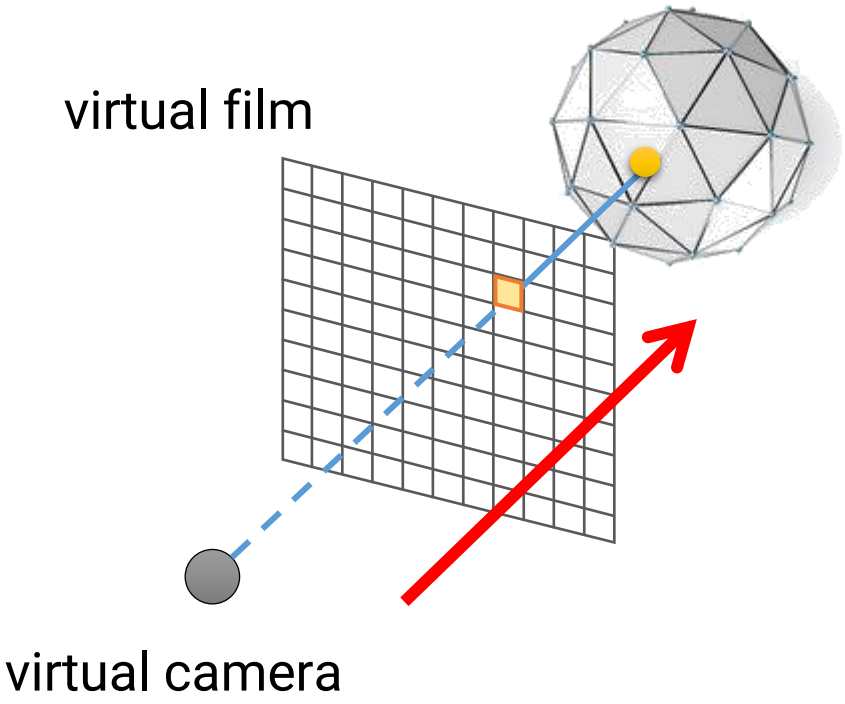
Animation Production Pipeline

Outline

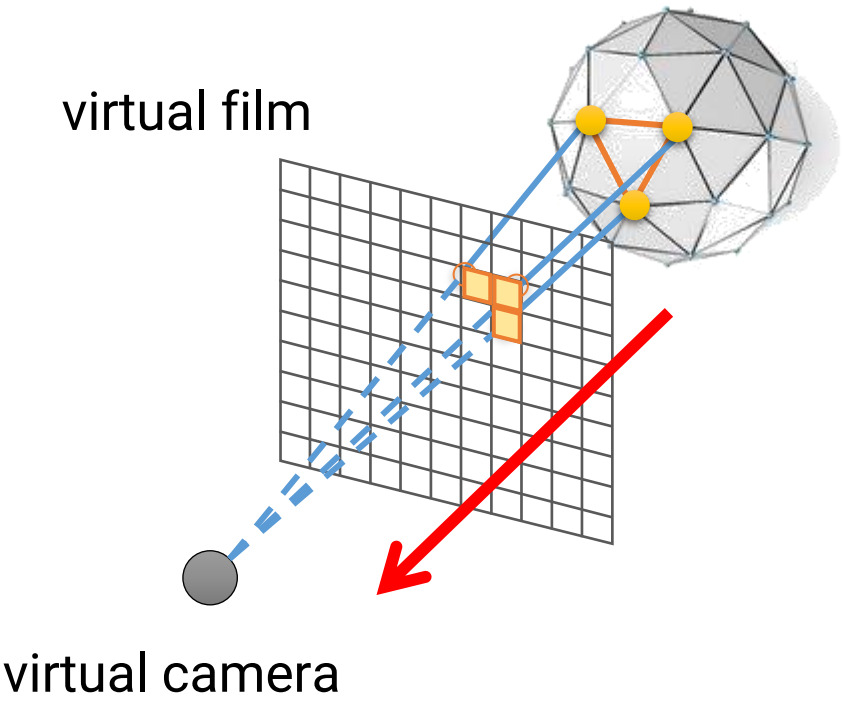
- Overview
- Modeling
- Animation
- Rendering
- Film production pipeline
- **Ray tracing and rasterization**

Bring Triangles into Pixels

Ray Tracing

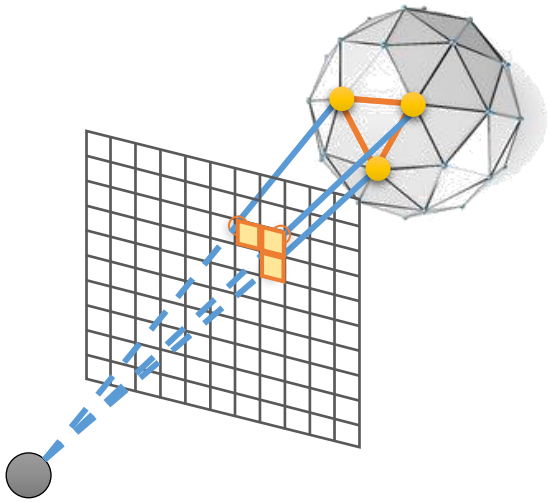


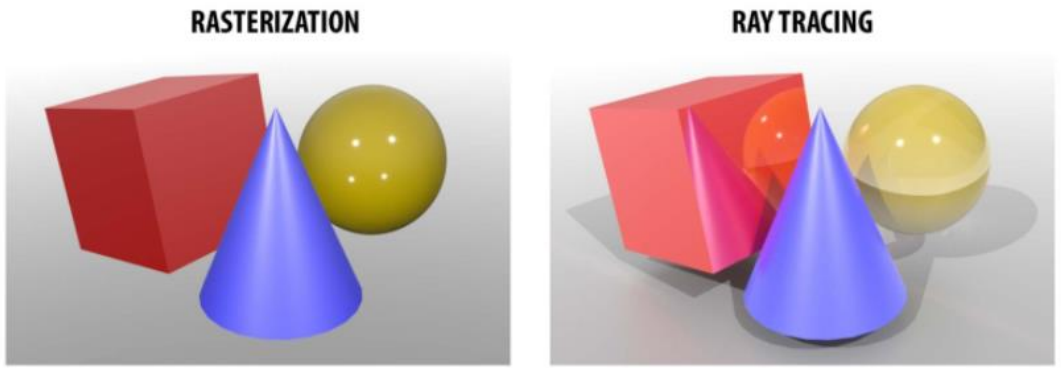
Rasterization



Rasterization

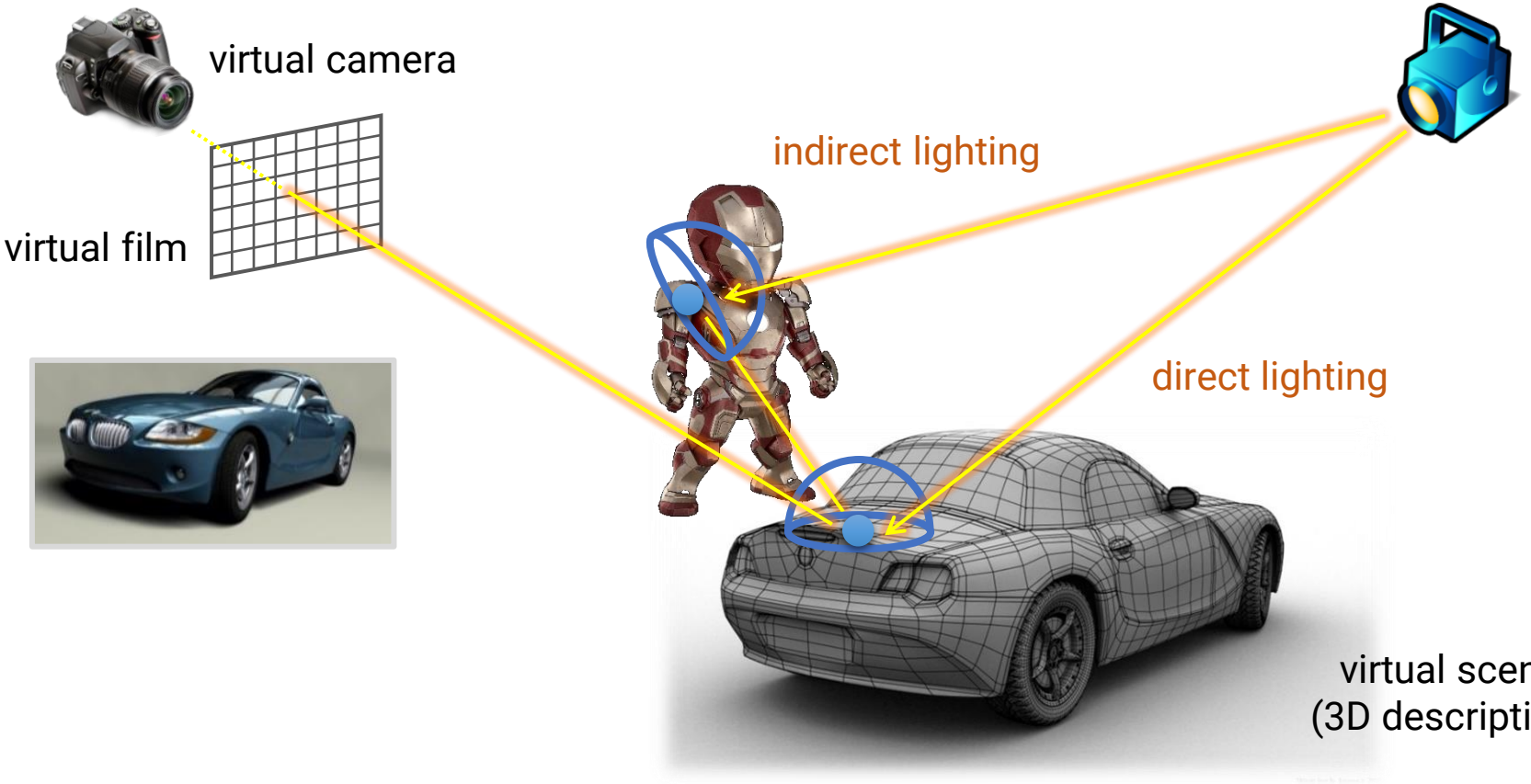
- Rasterization is **more friendly to hardware** and usually has higher parallelism
- But it is more difficult to simulate effects such as reflection, refraction, shadows, and global illumination
  - Need specialized algorithms





Ray Tracing

- Ray tracing is more general for simulating a wide variety of light transport paths

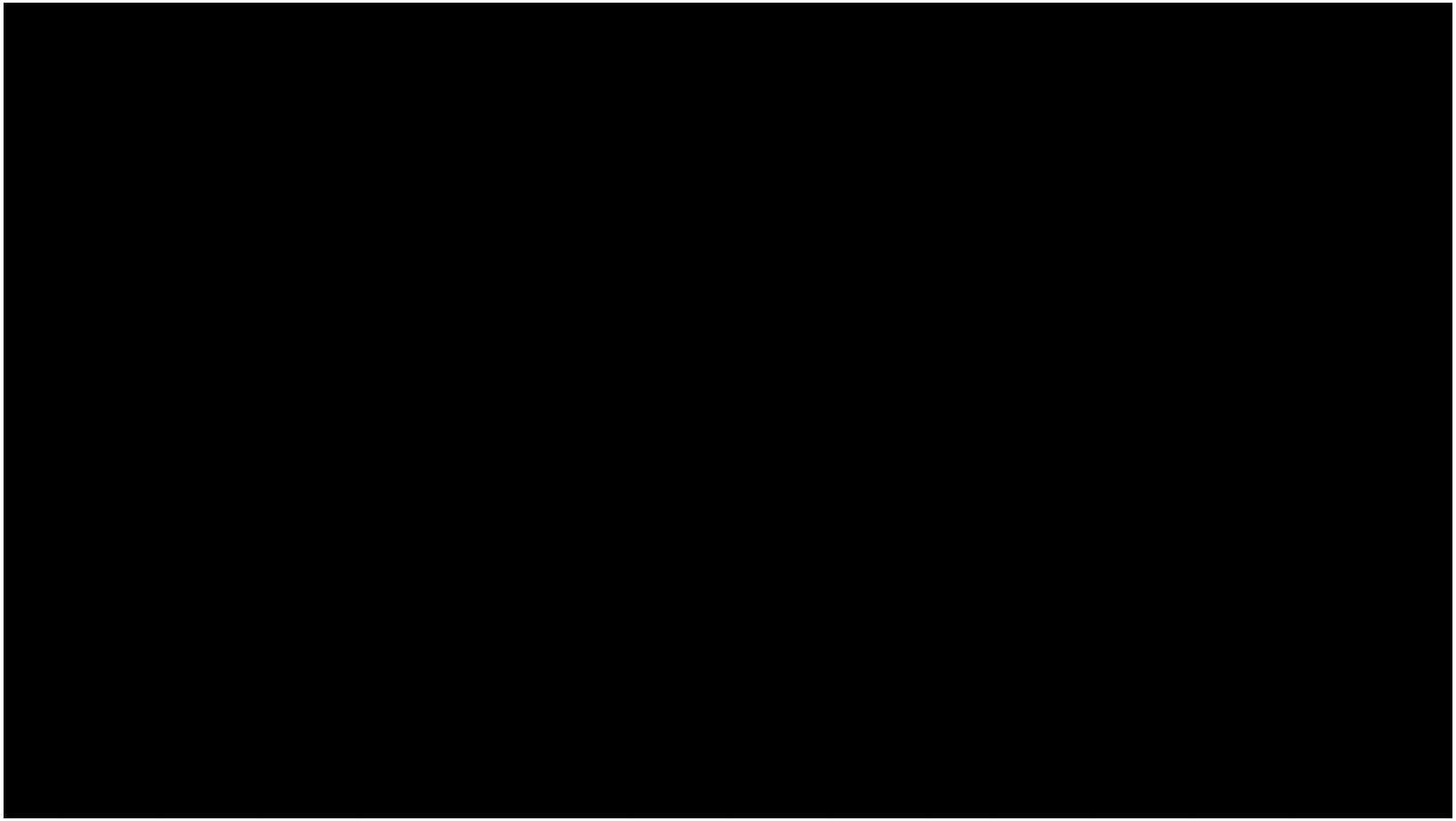


Ray Tracing

- However, its simulator usually has a **slow convergence rate** and produces lots of **noise** when the samples are not enough

Real-Time Ray Tracing

- FIRST DAY: A Star Wars short film made with UE5



Real-Time v.s. Offline Graphics

real-time **offline**


END OF PRESENTATION
ANY QUESTIONS?