Ray Tracing
in
Entertainment Industry

Tanaboon Tongbuasirilai
Dept. Computer Science
Kasetsart University

Week 3
Image formation and rays

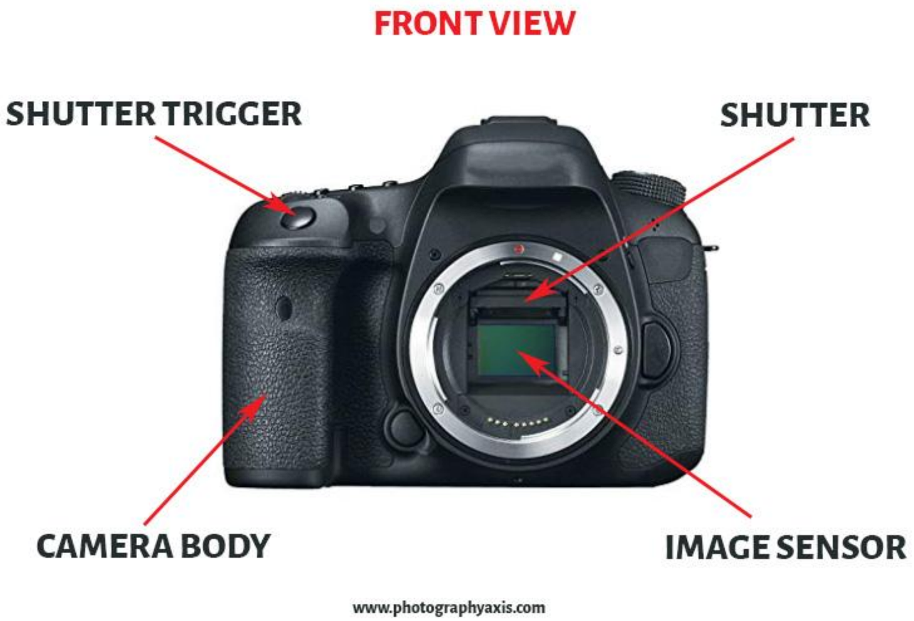
Image formation and rays

Camera and Lens classes

Ray class

Utility class

1. Recapped – Basic camera components



www.photographyaxis.com

- Image plane, sensor
- Shutter
- Aperture
- Lens

Camera class

- Attributes
    - Image width and height
    - Aspect ratio
    - Field of view
    - Camera center or look from (position)
    - Camera look at (position)
    - Camera Lens
    - Viewport width and height
    - Camera frame coordinates
    - Delta pixel (spacing between each pixel)
    - Film or image sensor

Image C1

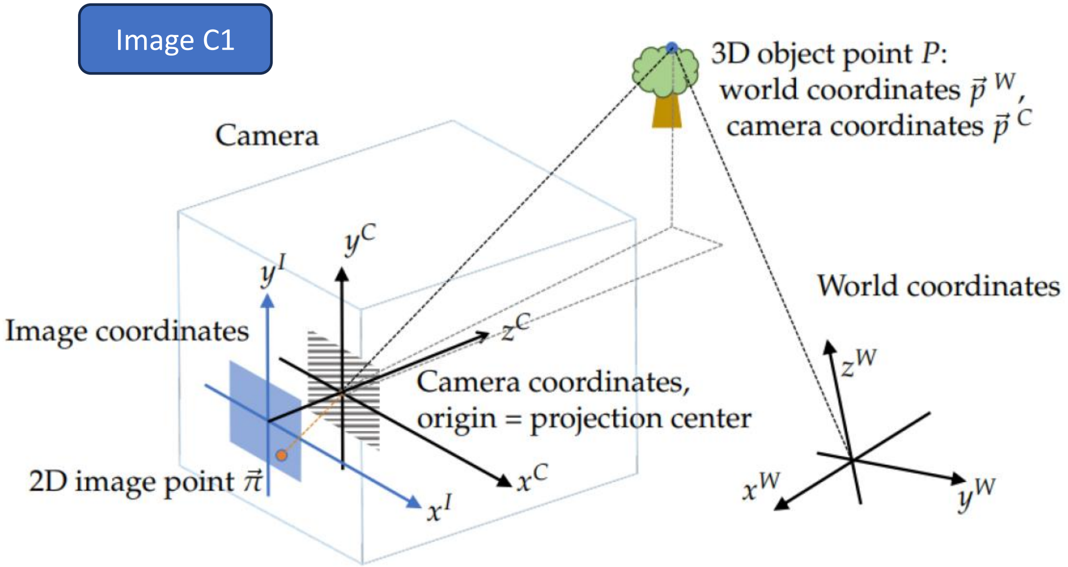

3D object point P:
world coordinates $\vec{p}^{\,W}$,
camera coordinates $\vec{p}^{\,C}$
Camera
y^C
y^I
World coordinates
Image coordinates
z^C
z^W
Camera coordinates,
origin = projection center
2D image point $\vec{\pi}$
x^C
x^I
x^W
y^W

Image C2

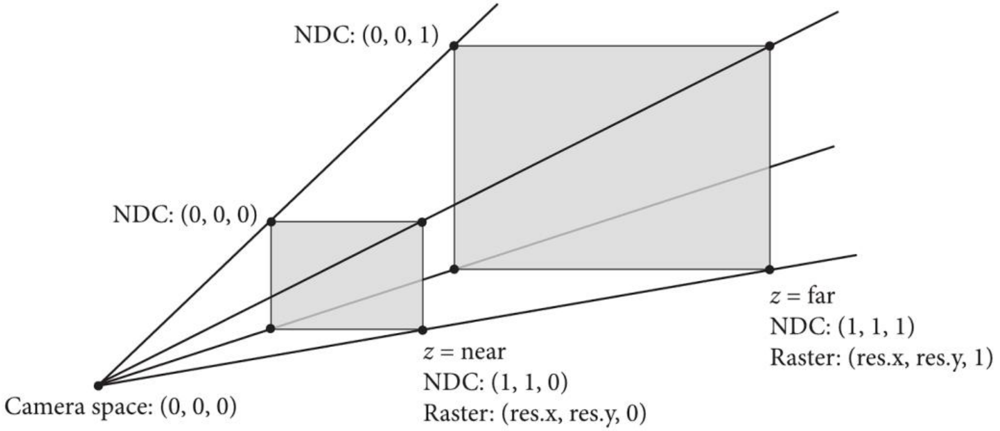
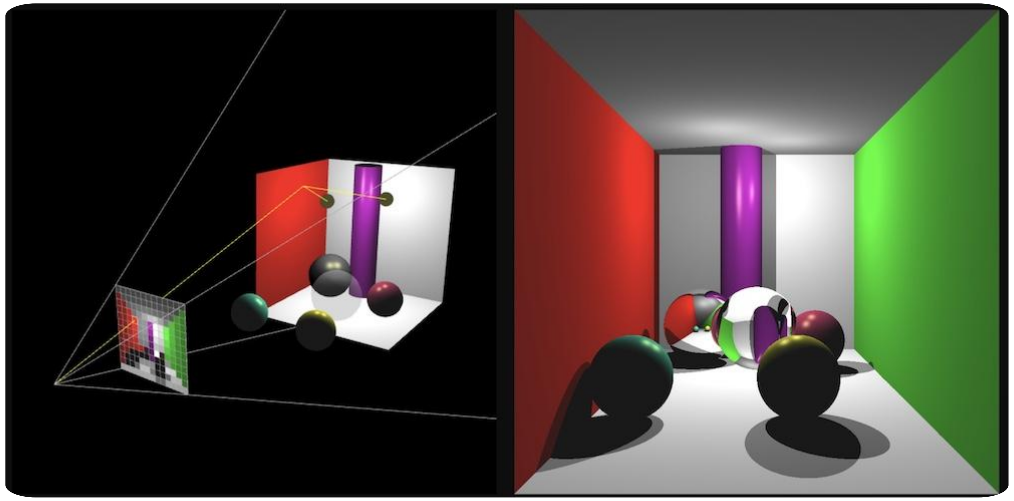


Image C1 from Pak, A., Reichel, S., & Burke, J. (2022). Machine-Learning-Inspired Workflow for Camera Calibration. Sensors, 22(18), 6804. https://doi.org/10.3390/s22186804

Image C2 from Matt Pharr, Wenzel Jakob, and Greg Humphreys. Physically Based Rendering: From Theory To Implementation, 2004-2023.

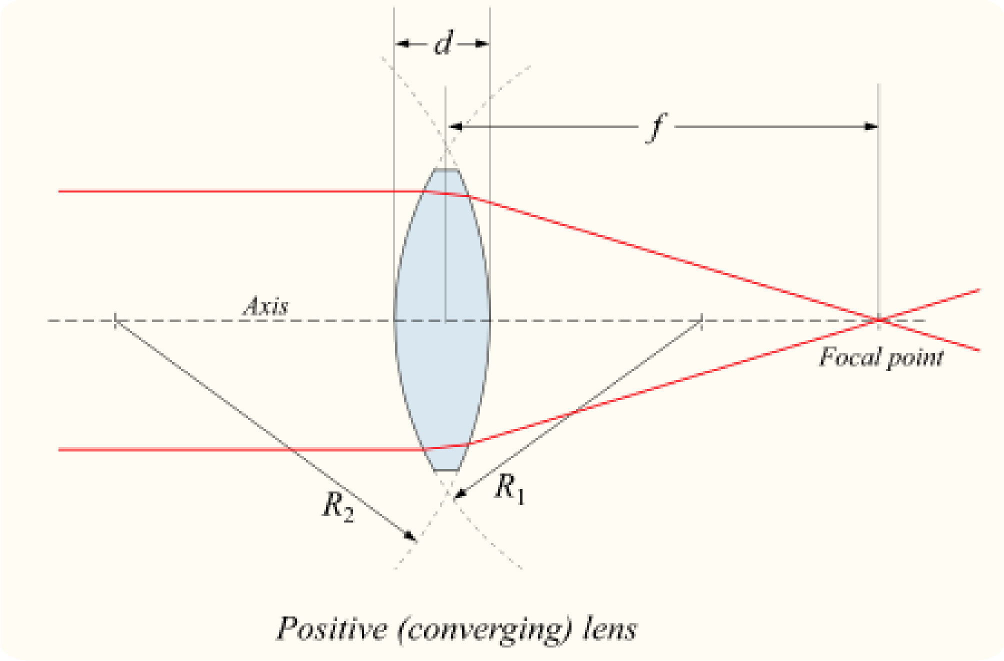
Camera class (cont.)



- Methods
  - Lens setup
  - Camera setup
  - Ray shooting
  - Film writing

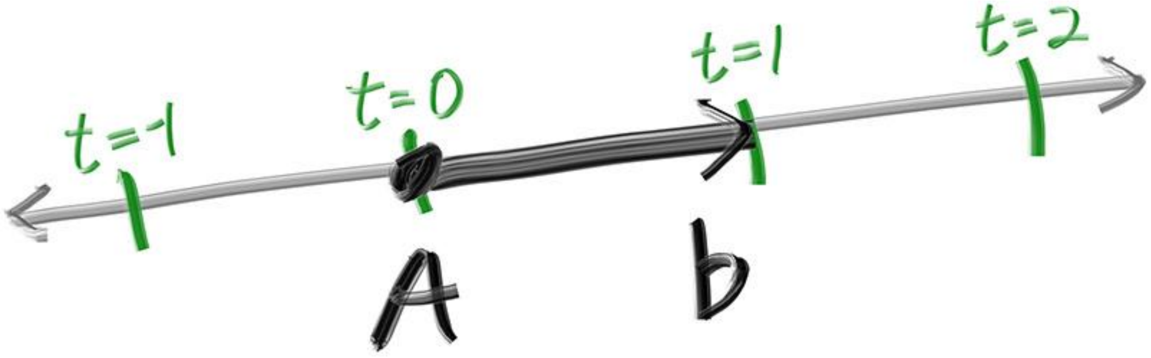
Image from Assignment 6 - Ray Tracer, https://people.engr.tamu.edu/sueda/courses/CSCE441/2022S/assignments/A6/index.html.

Lens class



- This Lens approximation
  - Attributes
    - Defocus angle
    - Focus distance
  - Methods will be later explained in a late chapter.

Image from File:Lens1.svg. (2022, April 26). Wikimedia Commons. Retrieved December 9, 2025, from https://commons.wikimedia.org/w/index.php?title=File:Lens1.svg&oldid=651839294.



Ray class

- Attributes
  - Origin : ray origin position (in 3d)
  - Direction : ray direction (in 3d)

- Methods
  - Ray at t : To compute where a ray position is given time t.

Image from Peter Shirley, Trevor David Black, Steve Hollasch . Ray Tracing in One Weekend. Version 4.0.2, 2025-04-25. Copyright 2018-2024 Peter Shirley. All rights reserved.

Utility classes

Vector class	Define cartesian coordinates
Color class	Define color spectrum (RGB or multi spectrum)
Interval class	Handle a value in a specified range.
Hit info class	Keep information of the ray intersection.

Vector class

- Name : Vec3
- Attributes
  - Array e[3]
- Methods
  - Coordinate values : x(), y(), z()
  - Length of a vector
  - Unit vector / normalize
  - Dot product
  - Cross product
  - Mathematical binary operations : + - * / (add, minus, multiply, div)
  - Unary operation : - (negation)

Color class

- Vec3 as a base (parent) class
- Color is a child class of Vec3
- Attributes
  - Array e[3] (derived from Vec3)
- Methods
  - RGB color : r(), g(), b()
  - Binary operations : + - * /
  - Unary operation : negation
  - Conversion : ex. [0,1) --> [0, 255]

Interval class

- Limit any arbitrary scalar value in the range.
- Attributes
  - Min value
  - Max value
- Methods
  - Is the value inside ?
  - Is the value bounded ?
  - Is the value near zero ?
  - Clamping the value
  - Empty interval
  - Unlimited interval

Hit info class

- Point of intersection.
- When the intersected ray hits.
- Surface normal at the intersection point.
- What material is hit.
- Is it front face or back face ?

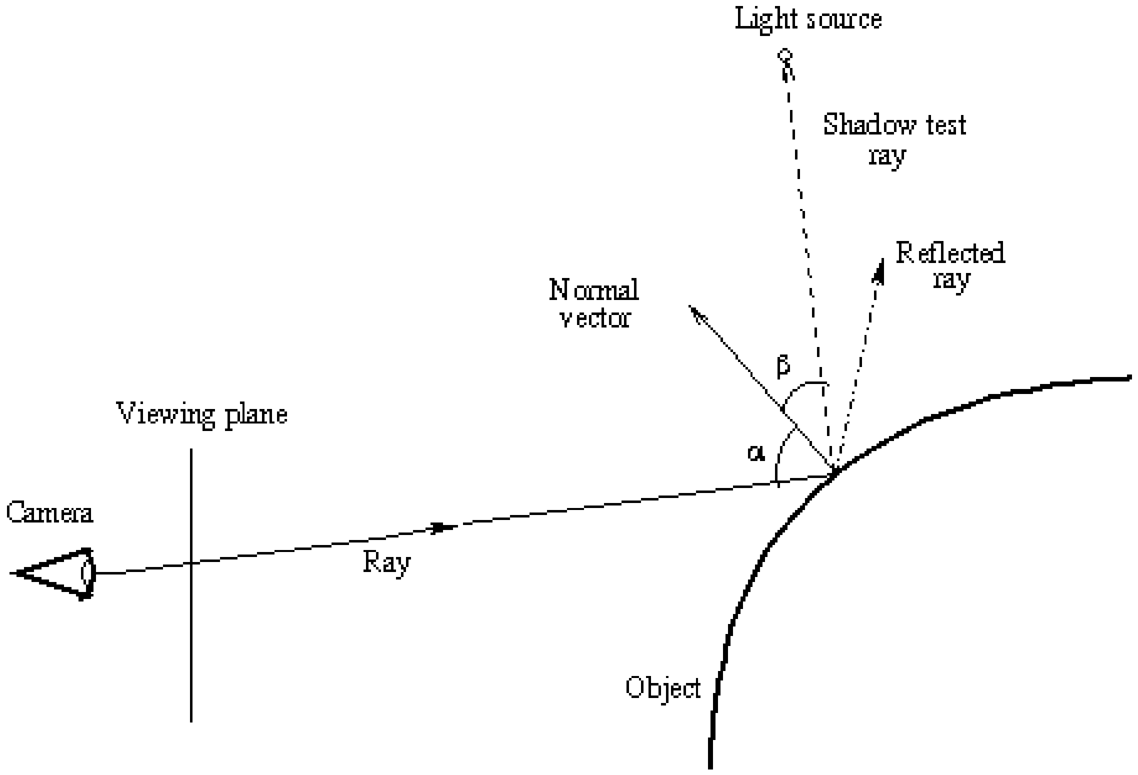


Image from POV-Ray SDL tutorial: A raytracer. https://www-f9.ijs.si/~matevz/docs/PovRay-3.50b/povdoc_133.html.

Static method

A static method is a class method which can be called without instantiating.

A non-static method is a class method which can be called from instances of the class.

Image writer (python)

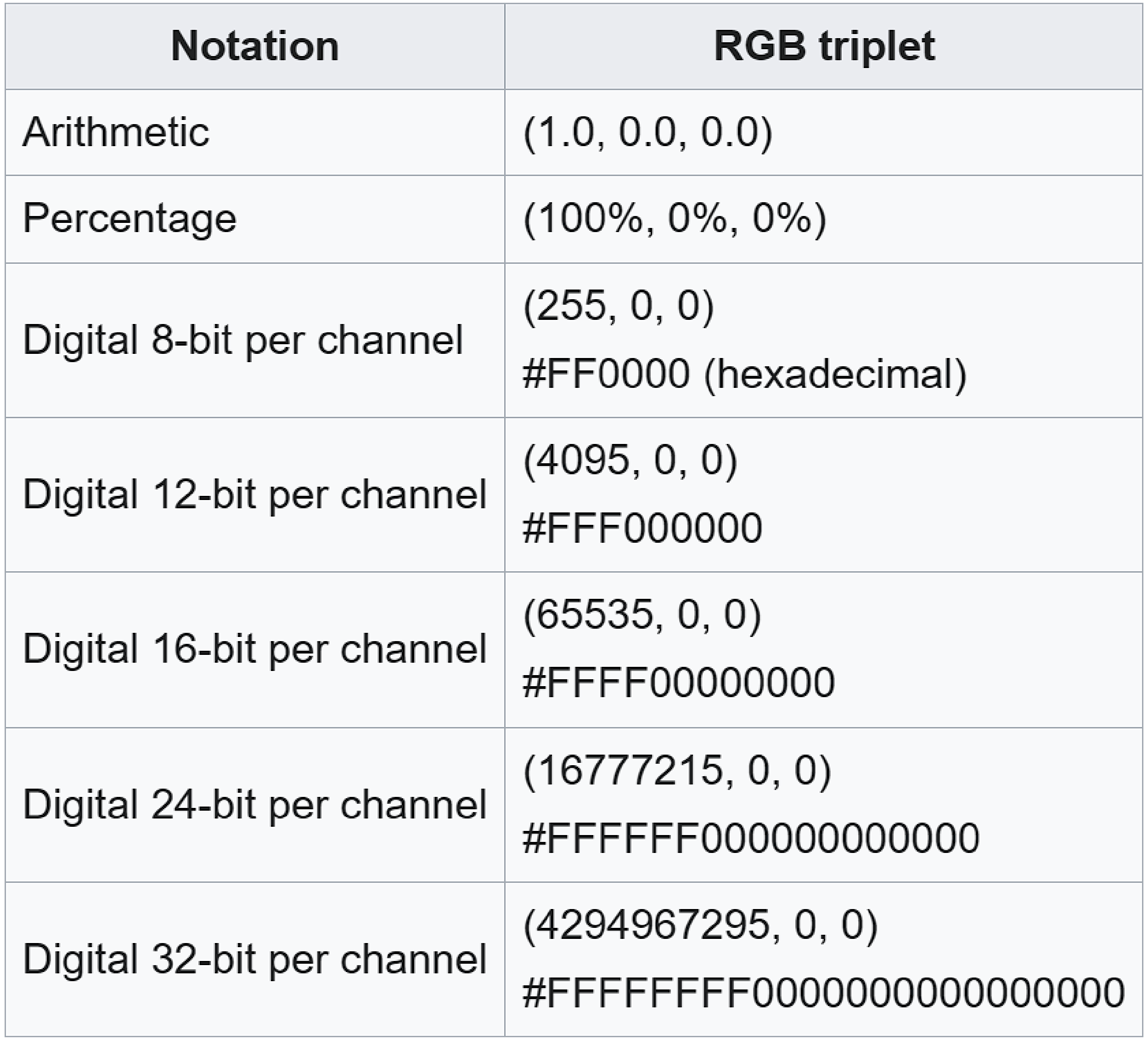
- Python imaging library (PIL)
  - https://pypi.org/project/Pillow/

- Example (python)

```
from PIL import Image as im
png_film[y,x,color] = color
data = im.fromarray(png_film.astype(np.uint8))
data.save(strPng_filename)
```

8-bit color

- Each color channel has 256 values.
- 8-bit RGB color ~ 16m colors.
- Color ranging from [0,1] --> [0,255].
- Clamping is sometimes needed.



Notation	RGB triplet
Arithmetic	(1.0, 0.0, 0.0)
Percentage	(100%, 0%, 0%)
Digital 8-bit per channel	(255, 0, 0) #FF0000 (hexadecimal)
Digital 12-bit per channel	(4095, 0, 0) #FFF000000
Digital 16-bit per channel	(65535, 0, 0) #FFFF00000000
Digital 24-bit per channel	(16777215, 0, 0) #FFFFFF000000000000
Digital 32-bit per channel	(4294967295, 0, 0) #FFFFFFFF0000000000000000

Table from https://en.wikipedia.org/wiki/RGB_color_model.

RGB Color space

- An RGB triplet describes the exact position of a specific color in a 3D color cube.
- Each number in the triplet (red, green, blue) corresponds to a coordinate, locating the color either inside the cube, on its surfaces, or along its edges.

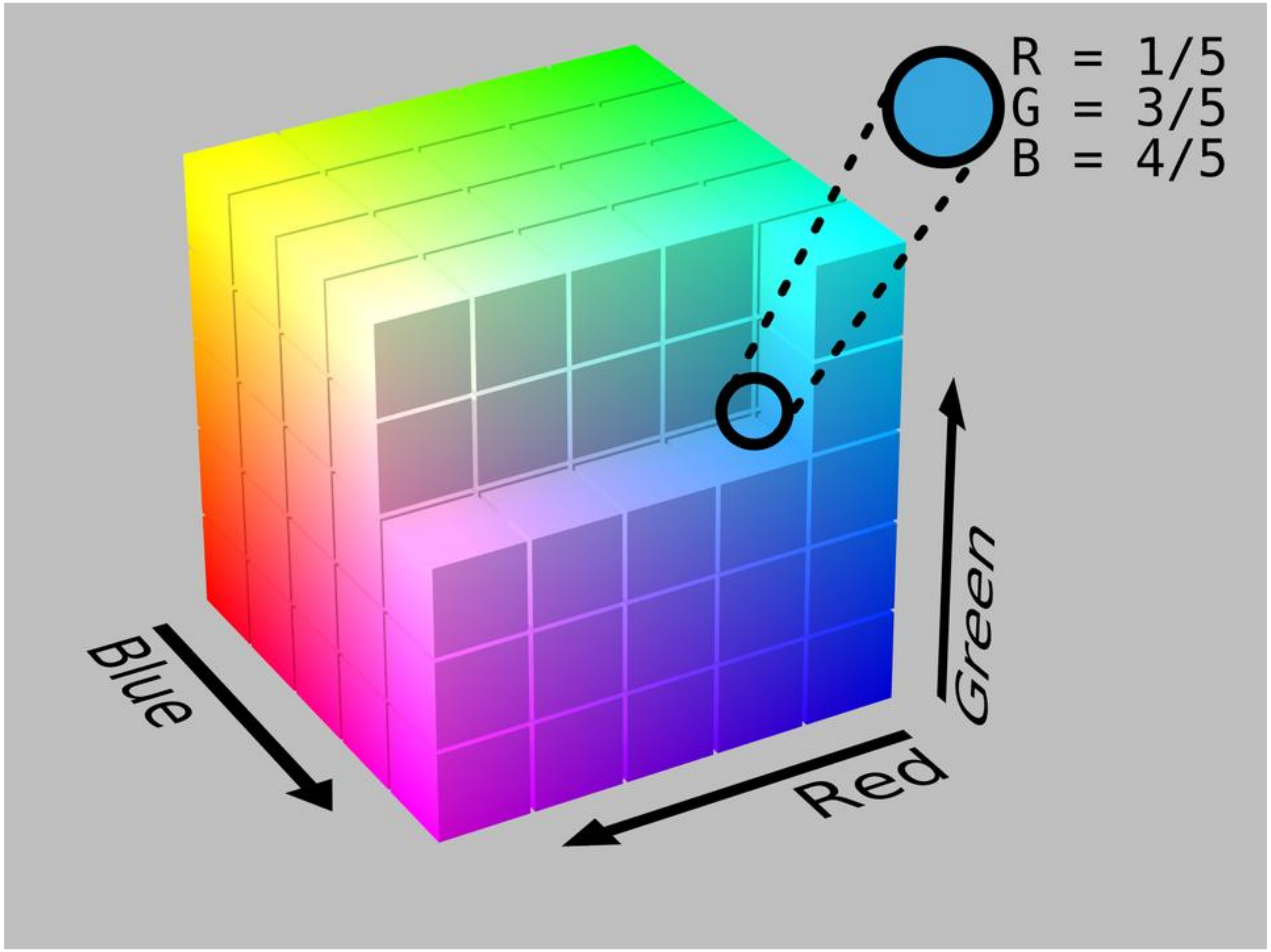
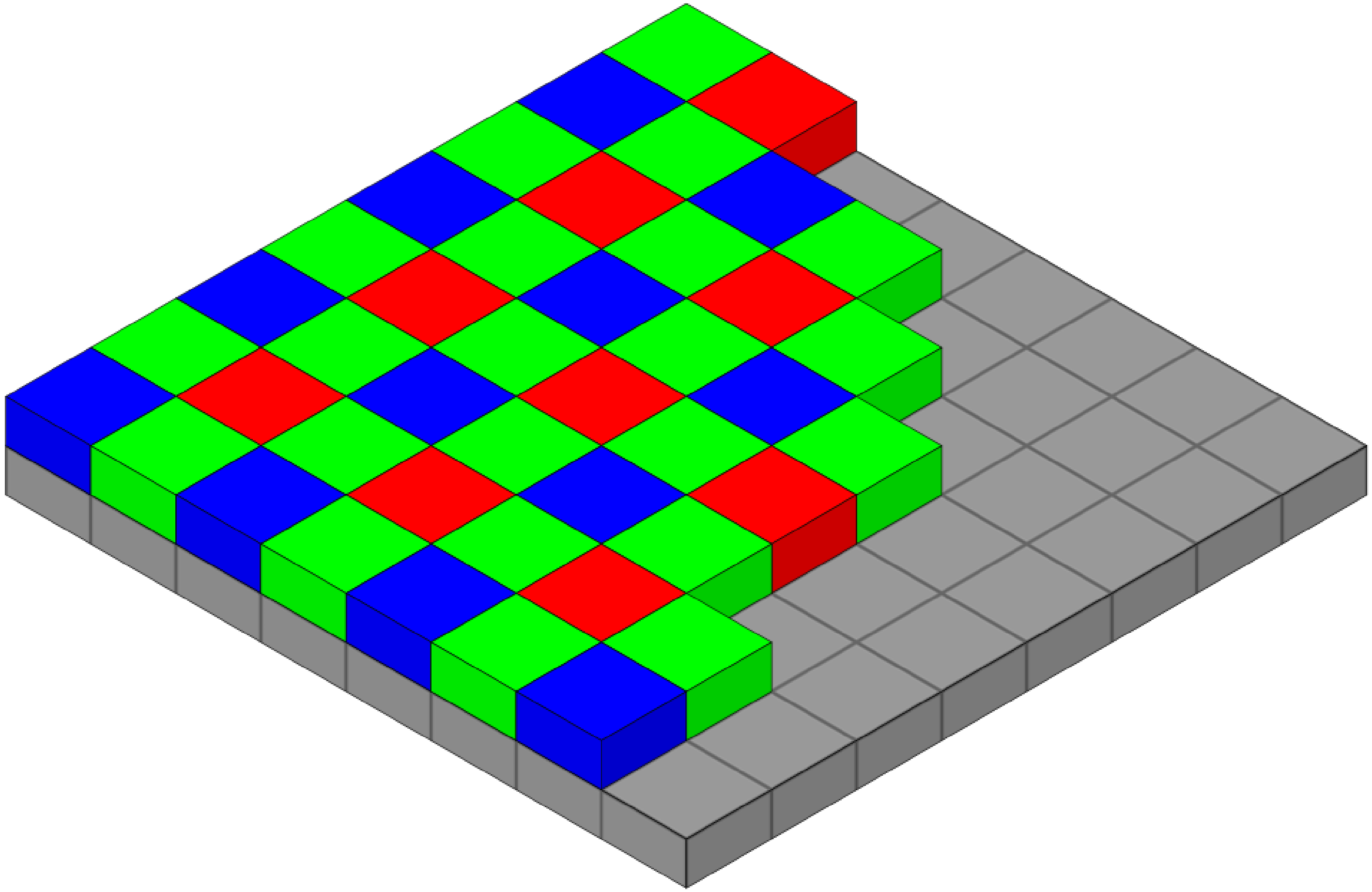


Image from File:RGB Cube Show lowgamma cutout b.png. (2024, October 7). Wikimedia Commons. Retrieved December 9, 2025, from https://commons.wikimedia.org/w/index.php?title=File:RGB_Cube_Show_lowgamma_cutout_b.png&oldid=934666855.

A Bayer filter mosaic in image sensor



- Cameras typically employ various **demosaicing algorithms** to reconstruct a full-color image from the incomplete color information captured by the image sensor.
- The image sensor usually has a **Bayer filter array**, which means each pixel only records one color (red, green, or blue).
- **Demosaicing** is the process of interpolating the missing color information for each pixel based on its neighbors.

Image from File:Bayer pattern on sensor.svg. (2025, September 8). Wikimedia Commons. Retrieved December 9, 2025, from https://commons.wikimedia.org/w/index.php?title=File:Bayer_pattern_on_sensor.svg&oldid=1082533639.

Shooting a ray (Class assignment (3.))

Shoot

Shoot a ray at the center of the pixel.

- get_center_ray()

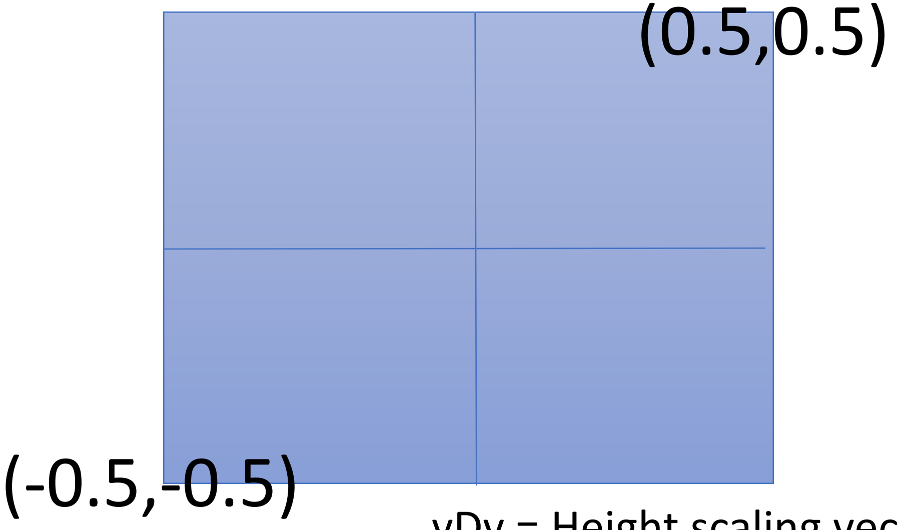
Shoot

Shoot a ray at a random location inside the pixel.

- get_ray()

Shoot a ray at a random location

- Random a location inside a rectangle.
- Method : random_pixel_in_square(vDu, vDv)
- Pixel center is at (0,0)
- Pixel corners are (-0.5,-0.5) (-0.5,0.5) (0.5,-0.5) (0.5,0.5).



vDu = Width scaling vector

vDv = Height scaling vector

Renderer class

- Regarding the class diagram, the renderer defines how we combine all ray tracing components.
- For now, it includes the following instances
  - Camera
- We also need to implement some functions to manage
  - Background color
  - Ray color
  - Simple ray intersection
- See RT_renderer.py

Codes and class assignment!

- Github : RT-python-week03
    - https://github.com/KUGA-01418283-Raytracing/RT-python-week03