

# Lecture 10: Camera

DHBW, Computer Graphics

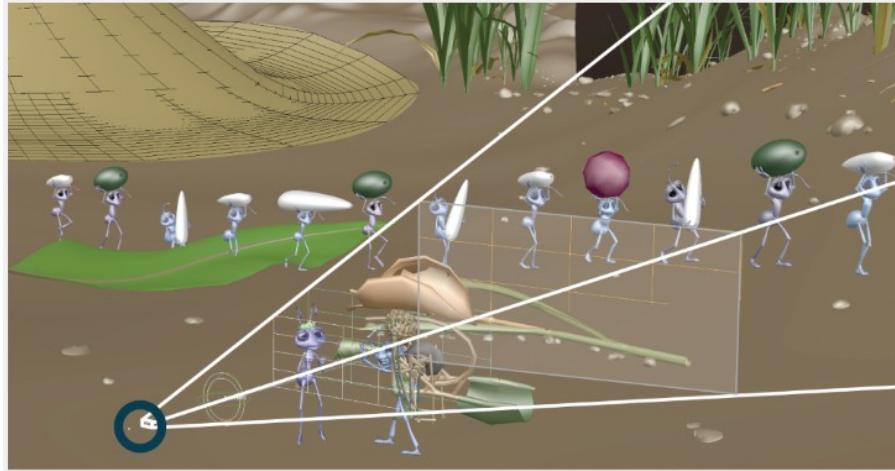
Lovro Bosnar

1.3.2023.

# Syllabus

- 3D scene
    - Object
    - Light
    - Camera
  - Rendering
  - Image
- Camera
    - Real camera system
    - Image formation
    - Pinhole camera model
    - Pinhole camera in rendering
- 

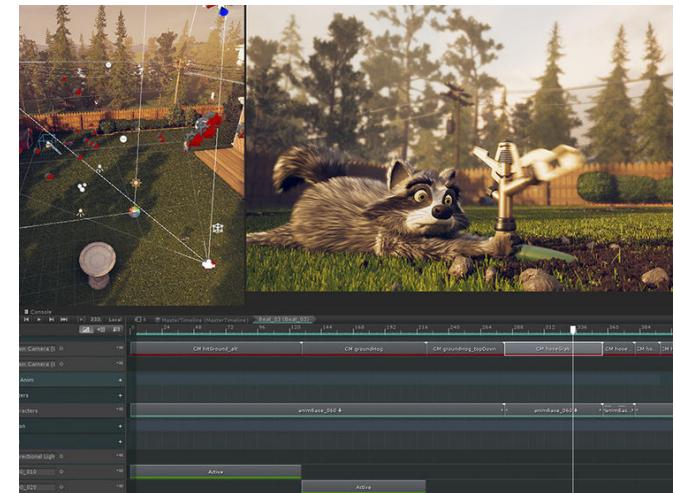
# Where is camera?



<https://sciencebehindpixar.org/pipeline/sets-and-cameras>



<https://www.half-life.com/en/alyx/>



<https://unity.com/solutions/film-animation-cinematics>

# Where is camera?

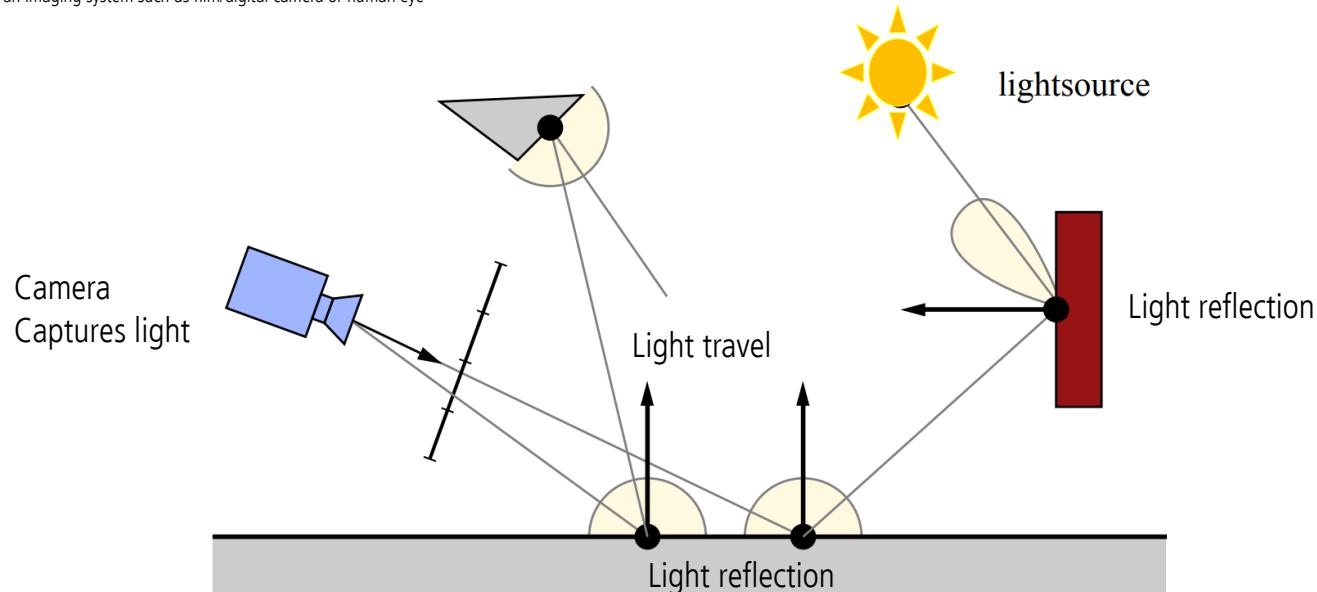
- In rendered images, camera is not seen – it defines what we see.



# Camera in rendering

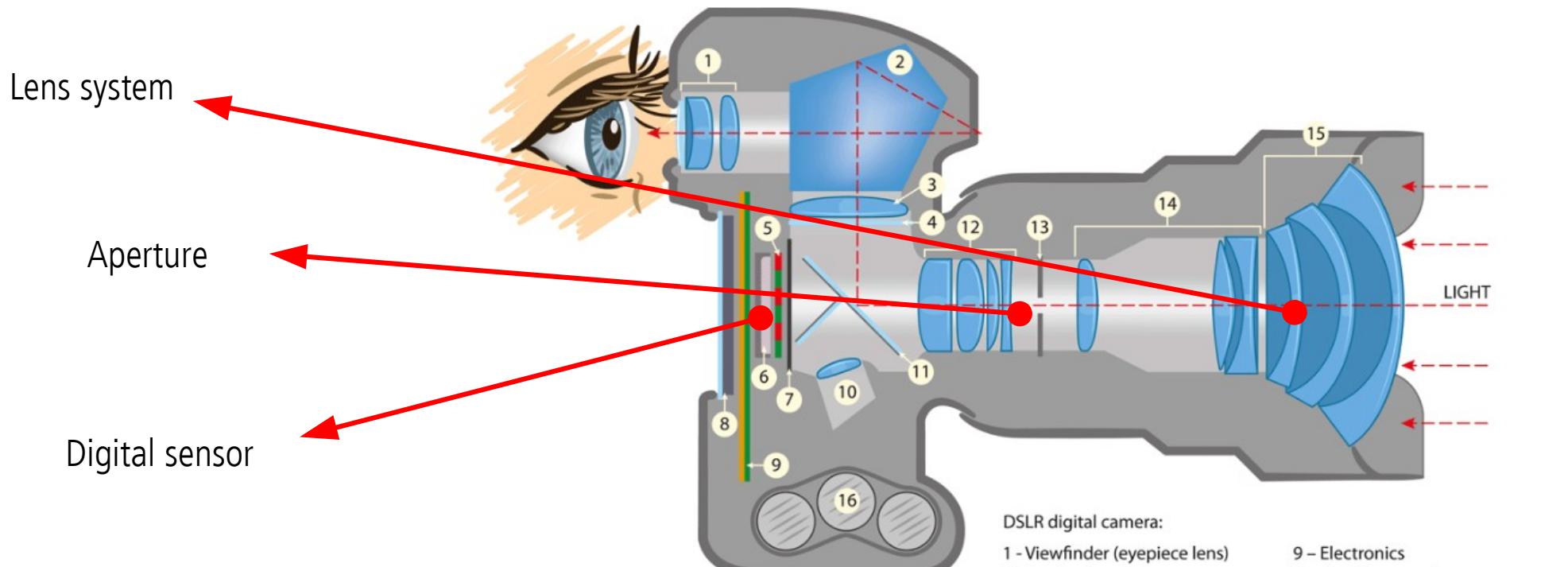
TODO

- Interesting: <https://google.github.io/filament/Filament.html#imagingpipeline?>
- In rendering camera model is required to at least to:
  - Define a portion of visible scene
  - Type of projection: perspective/orthographic
- Camera during image formation:
  - Picks reflected light from 3D scene (light transport and shading)
  - Forms image
- Rendering process is calculating radiance (color) from shaded surface point to camera position.
  - This simulates simplified model of an imaging system such as film/digital camera or human eye
  - <image: object, camera>



# Real camera system

# Real camera systems



DSLR digital camera:

- |                                |                                     |
|--------------------------------|-------------------------------------|
| 1 - Viewfinder (eyepiece lens) | 9 - Electronics                     |
| 2 - Pentaprism                 | 10 - Autofocus system               |
| 3 - Focusing screen            | 11 - Reflex and relay mirror        |
| 4 - Condenser lens             | 12 - Focusing elements              |
| 5 - Color and infrared filter  | 13 - Aperture                       |
| 6 - Digital sensor             | 14 - Zoom elements                  |
| 7 - Shutter                    | 15 - Front light gathering elements |
| 8 - Display                    | 16 - Batteries                      |

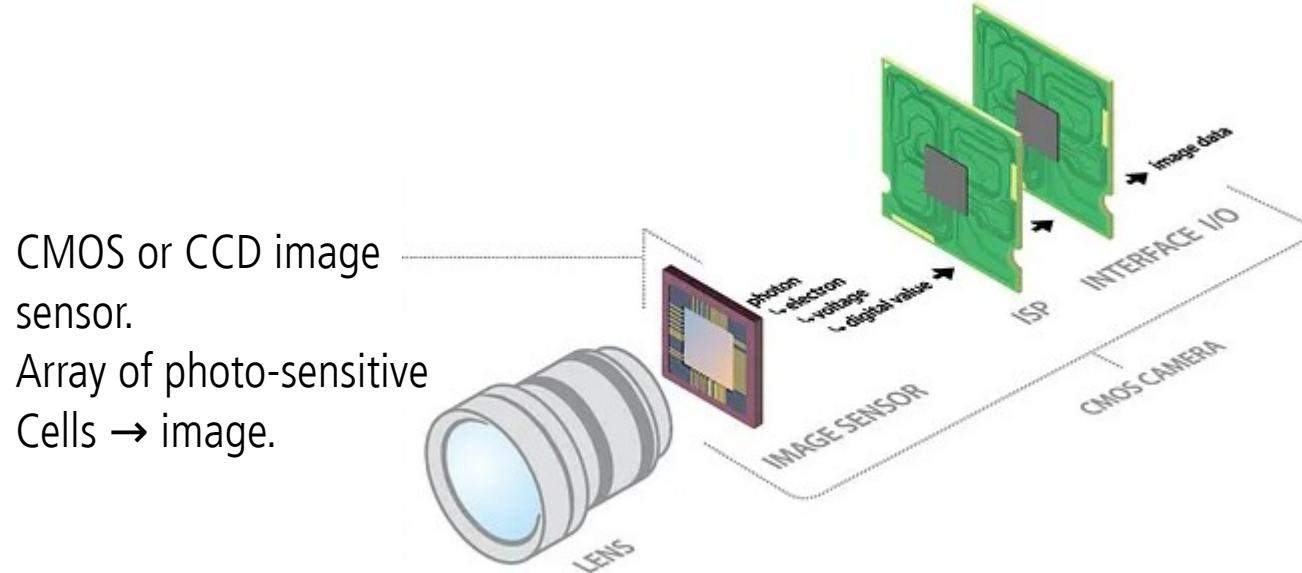
Basic Parts of a DSLR Camera

# Real camera systems

- Camera systems contain many small discrete **sensors**: CMOS/CCD
  - Each measures incoming **radiance** values and converts it to **color** signal
- Exposing sensor to radiance will not produce image without **light-proof enclosure** with small opening – **aperture**.
  - Such setup restricts light where light can enter and strike sensor
- **Lens** is placed at the aperture to focus the light so each sensor receives light from only a small set of incoming directions
  - Lens with aperture average light over small area and small set of incoming directions

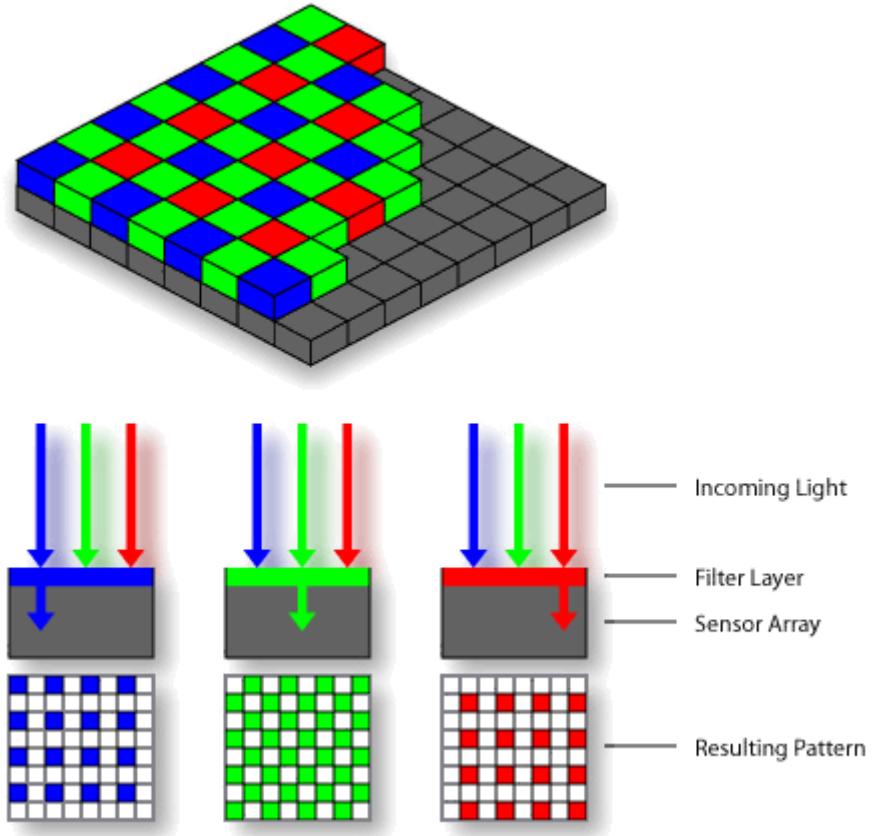
# Camera sensor

- Convert incident light energy into analog signal (e.g., voltage)



# Camera sensor

- Each sensor cell records incoming light energy per second
- Only one signal over whole cell area is captured
- Sensor Measures only R,G,B wavelengths



[https://simple.wikipedia.org/wiki/Bayer\\_filter](https://simple.wikipedia.org/wiki/Bayer_filter)

# Camera and image formation

- Image is created when light falls digital sensor (array of photo-detector cells)
- Sensors measure average radiance which quantifies the brightness and color of single ray of light
- This is why in computer graphics we only care about light rays which are coming to camera
  - That is why we generate rays from camera into the scene → **backward tracing**

# Image formation

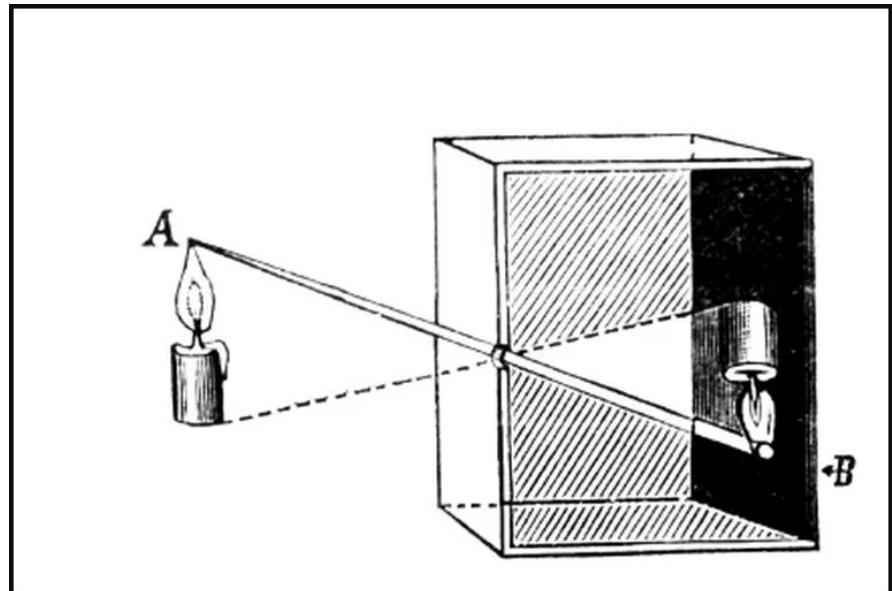
# Principle of photography

- Two main processes:
  - How is image created in camera
  - How is image stored on film/file\*

\* Physical support to store an image, thus simulating the photochemical processes used in traditional film photography is out of scope  
14  
Renderer which simulates image storage on film: <https://maxwellrender.com/>

# Camera obscura

- Lightproof box with:
  - black interior (to prevent light reflection)
  - with very small hole – **aperture**
  - light-sensitive film opposite to aperture
- When aperture is opened, film is exposed to light
  - Light passing through the hole forms an inverted image of the external scene on the opposite side of the box
  - Time which aperture must be opened to capture image on film is called **exposure**
- A foundation for modern photographic cameras



[https://www.inquirer.com/philly/entertainment/20160806\\_Camera\\_Obscura\\_at\\_Fleischner\\_throws\\_new\\_light\\_on\\_photography.html](https://www.inquirer.com/philly/entertainment/20160806_Camera_Obscura_at_Fleischner_throws_new_light_on_photography.html)

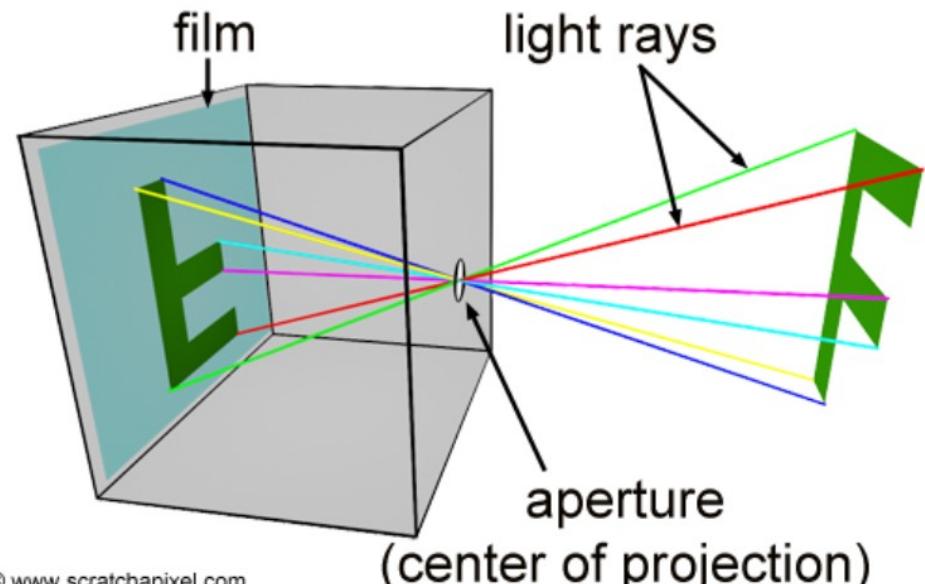
# Camera obscura = Dark room



[https://www.youtube.com/watch?v=gvzpu0Q9RTU&ab\\_channel=NationalGeographic](https://www.youtube.com/watch?v=gvzpu0Q9RTU&ab_channel=NationalGeographic)

# Pinhole camera

- Simplest type of camera in real world is **pinhole camera** (**camera obscura**)
- Main elements:
  - **Aperture**: center of projection: all rays entering converge here and diverge from it to the other side
  - **Film**
- Image formation
  - Aperture is opened to expose film to light
  - Aperture must be very small: only one ray reflected from the world in point P enters the camera and intersects film in one point
  - Each point in the visible portion of the scene corresponds to a single point on the film



© www.scratchapixel.com

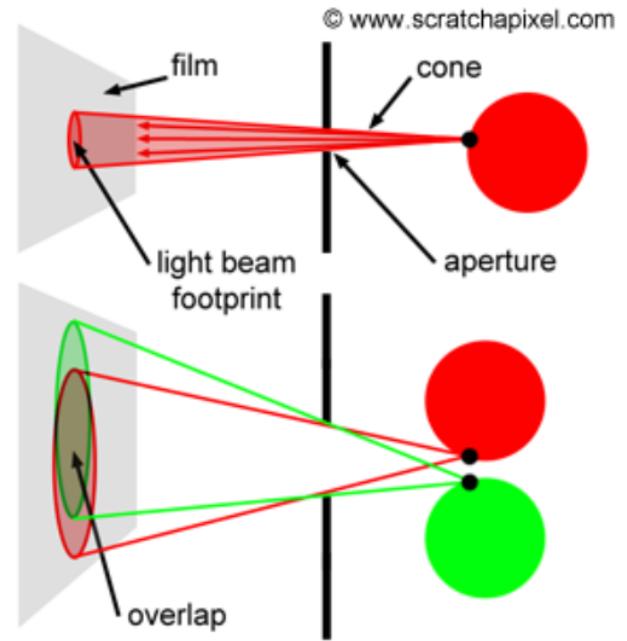
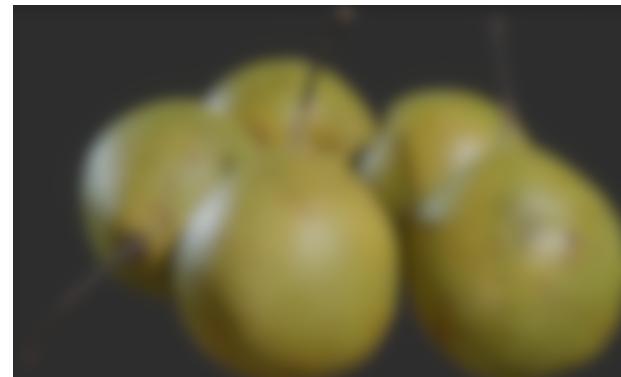
<https://www.scratchapixel.com/lessons/3d-basic-rendering/3d-viewing-pinhole-camera>

# Pinhole camera: aperture size

- Formed image is sharp if each point of the object maps to the one point of the film
- Ideal pinhole: aperture is so small that only one ray passes through it
  - Not possible in a real world because of diffraction of light
- It is never a single ray that passes the aperture – cone of rays
  - the cone of rays (its angle) is determined by a size of aperture.
  - Smaller cone → sharper image
  - Larger aperture → blurred image



[https://polyhaven.com/a/food\\_pears\\_asian\\_01](https://polyhaven.com/a/food_pears_asian_01)

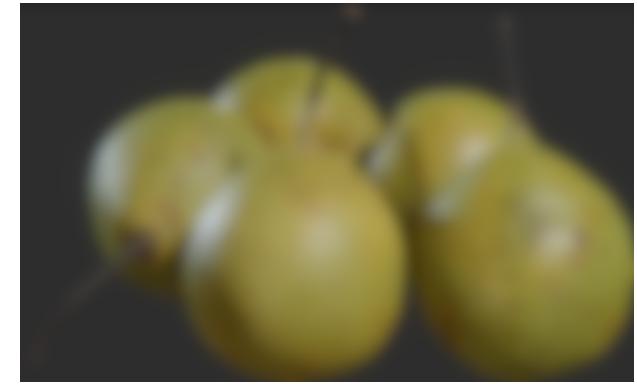
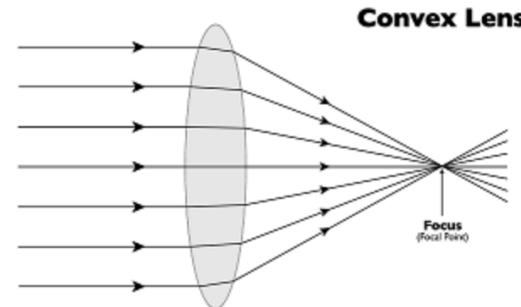


Cones are overlapping and single point appears multiple times on film  
→ blur (out of focus).

<https://www.scratchapixel.com/lessons/3d-basic-rendering/3d-viewing-pinhole-camera>

# Aperture and Lens camera

- Aperture trade-offs:
  - Very small aperture requires long exposure times to form an image: motion blur
  - Large aperture causes blurred images
- Solution is to use lens in front of aperture
  - Rays entering camera are gathered (converged) and focused them to one point on a film plane
  - Aperture can now be larger enabling smaller exposure time while lenses canceling blur



# Depth of field

- Lens cameras introduce distance between nearest and the farthest object from the scene that appears sharp in the formed image – **depth of field**
- Pinhole cameras have infinite depth of field
  - Because of this, images rendered with pinhole cameras are completely sharp and thus look artificial



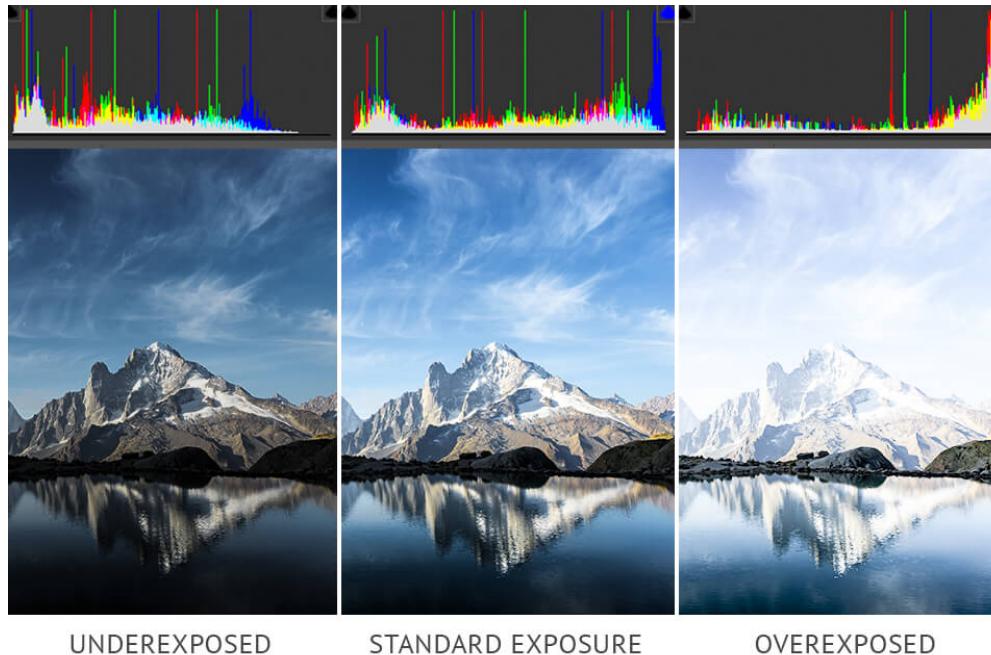
# Bokeh

- Out of focus emissive objects are aesthetically pleasing → bokeh effect.



# Pinhole camera: exposure

- Exposure: amount of light collected by camera → image brightness
- Amount of visible details in lit and shadowed areas
  - Real world has high difference in dynamic range



<https://fixthephoto.com/what-is-exposure-composition.html>

# Exposure

- Exposure is controlled with:
  - Aperture size (f-stop) → depth of field
  - Shutter speed → motion blur
  - Sensitivity (ISO) (gain) → noise



<https://photonify.com/3-steps-for-adjusting-the-depth-of-field-on-your-camera/>

<https://petapixel.com/exposure-triangle/>

<https://actioncamera.blog/2017/02/22/the-exposure-triangle/>

<https://www.shutterbug.com/content/give-photos-dramatic-perspective-motion-and-blur-simple-photoshop-technique-video>

# Exposure

- If aperture is very small, longer time is needed for image to form on a film.
- Time of which the aperture is open is called **exposure** time
  - In real cameras, longer exposure can produce blurred image if camera or objects in the scene are moving (not perfectly still)
  - Simulated camera do not have problem with this since simulated light transport is considered instant, therefore, simulation of motion blur requires additional simulation
- Therefore, generally, shorter exposure time is better, although simulation of motion blur is aesthetically pleasing



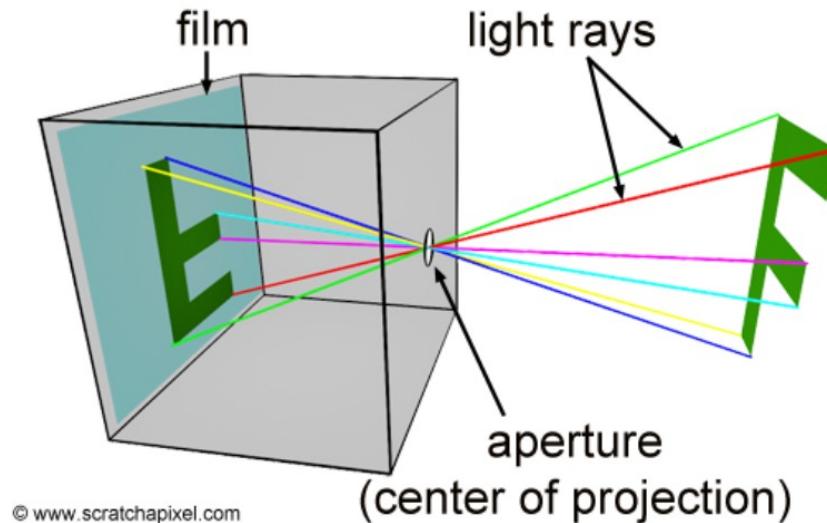
Earth rotation causes motion blur in long-exposure photos



# Pinhole camera model

# Pinhole camera model

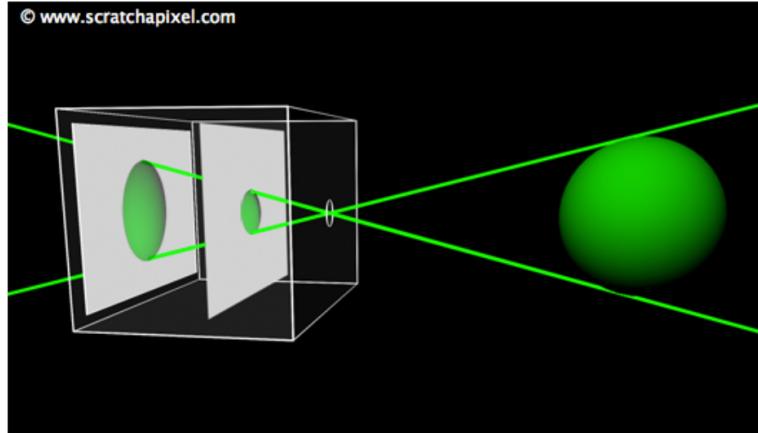
- Real cameras involve complex lens system. In graphics we model much simplified setup
  - Simplest camera model: **pinhole camera**
- Main elements:
  - Film (image plane)
  - Aperture (small opening)
- Parameters:
  - Focal length, angle of view
  - Film size



© www.scratchapixel.com

# Focal length

- Distance of film plane from aperture defines amount of scene that we see
  - Moving image plane (film plane) closer to aperture effectively performs **zoom out**
  - Moving film plane away from aperture effectively performs **zooming in**
- This parameter is called **focal length** or **focal distance**

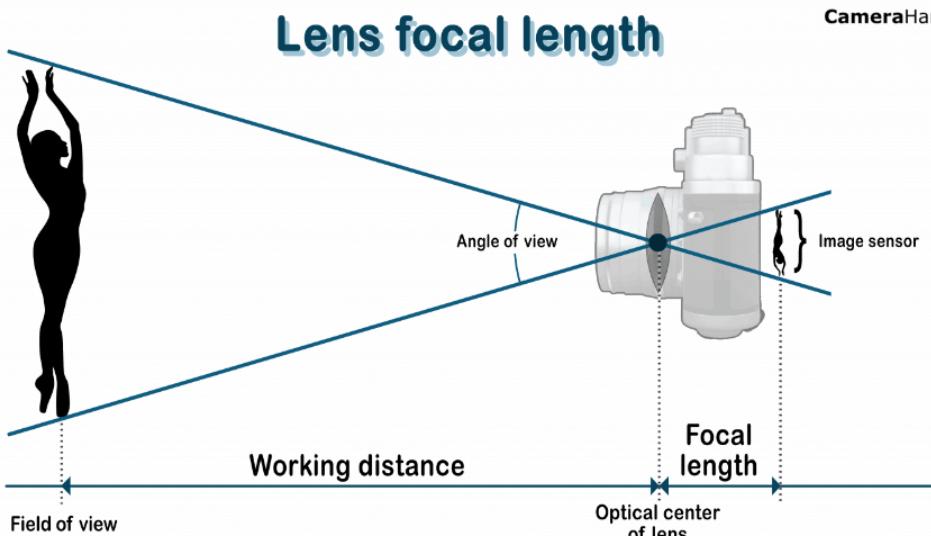


<https://www.nikonusa.com/en/learn-and-explore/a/tips-and-techniques/understanding-focal-length.html>

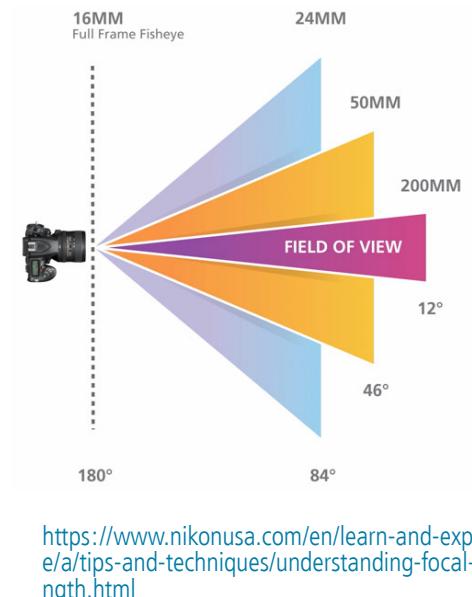


# Angle of view

- Zooming in and zooming out described by focal length (focal distance) can be also described by angle of the apex of **triangle defined with aperture and film edges**
- This angle is called **angle of view or field of view**



<https://cameraharmony.com/focal-length-explained/>



# Focal length and angle of view

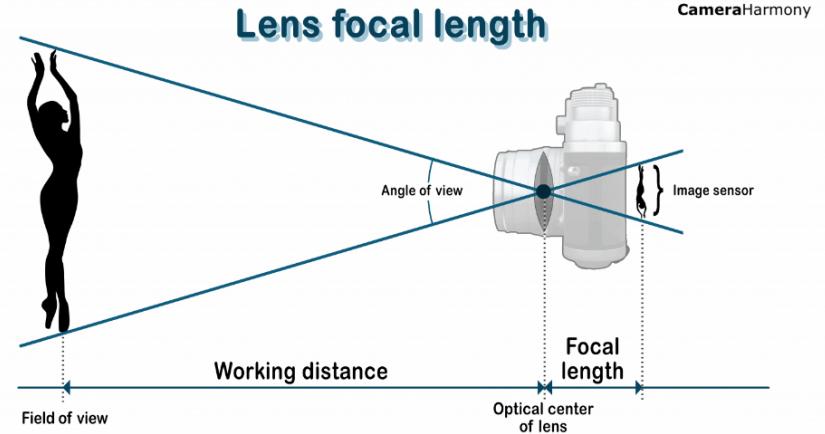
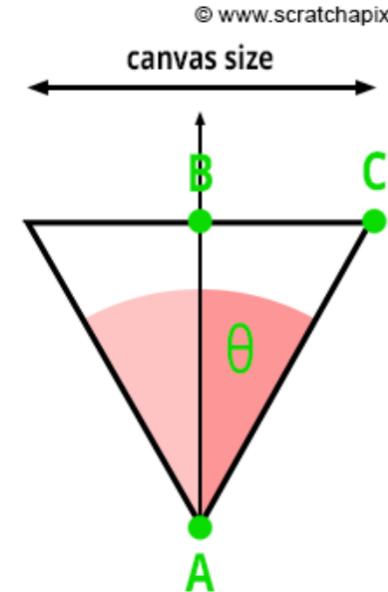
- Relation between focal length and angle of view

$$\tan(\theta) = \frac{BC}{AB}$$

$$BC = \tan(\theta) * AB$$

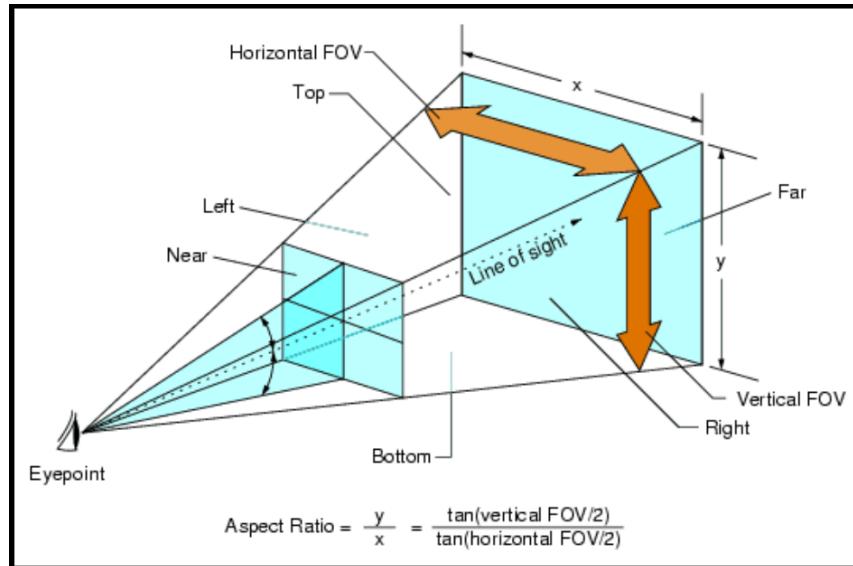
$$\text{Canvas Size} = 2 * \tan(\theta) * AB$$

$$\text{Canvas Size} = 2 * \tan(\theta) * \text{Distance to Canvas}.$$



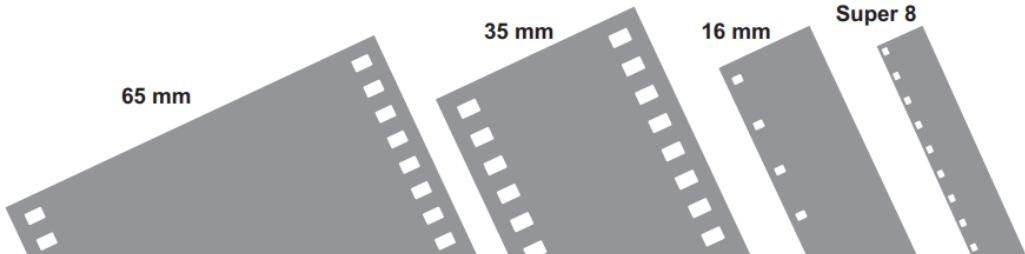
# Field of view

- In 3D, triangle defined with aperture and film edges is a pyramid and we distinguish horizontal and vertical FOV



# Film size

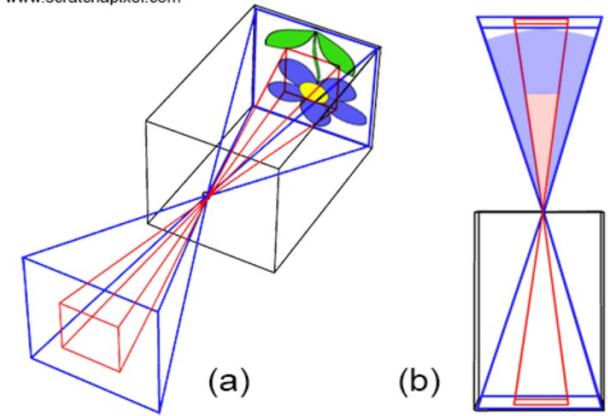
- Amount of scene that is captured also depends on film size (image sensor)
  - Film parameters are horizontal and vertical direction
- Smaller area of film size implies smaller angle of view and vice versa.
- Capturing the same extent of the scene with larger film requires adjusting focal length
- Larger film formats were developed for more details and better image quality
- In digital cameras, film is replaced with sensor, thus we talk about sensor size.
  - Sensor size has the same role as film size



Gauge refers to the width of film. Four are common: 65mm, 35mm, 16mm, Super 8.

[http://www.theodoropoulos.info/attachments/076\\_kodak05\\_Film\\_Types\\_and\\_Formats.pdf](http://www.theodoropoulos.info/attachments/076_kodak05_Film_Types_and_Formats.pdf)

© www.scratchapixel.com



35 mm / 135	120 / Medium Format	4x5 / Large Format
36 mm 24 mm 	4.5 cm 6 cm 645  6 cm 6x6  7 cm 67 	5 inch 4 inch 

<https://www.learnfilm.photography/guide-to-medium-format-photography/>

1920x1080

# Film size and image resolution

- Angle of view:
  - Depends on film size (image sensor size)
  - Doesn't depend on number of pixels on image sensor – **image resolution**
- Size of sensor and number of pixels are independent parameters!
- Image quality depends on:
  - Image sensor size
  - Number of pixels on sensor (resolution)
- Higher resolution images will have more details
  - Resolution is expressed in terms of pixels
  - Width and height resolution



1920x1080 20%



# Image resolution and aspect ratio

- Image aspect ratio (device aspect ratio) is calculated from image resolution

$$\text{Image (or Device) Aspect Ratio} = \frac{\text{width}}{\text{height}}$$

- Film and display devices (computer screens, television) have standard aspect ratios

- 4:3

- Old television systems and computer monitors (e.g., resolution 640x480)
    - Default for digital cameras
    - 35 film formats have aspect ratio 4:3

- 5:3 and 1.85:1

- Aspect ratio used in film

- 16:9

- Aspect ratio used by HD TV, monitors and laptops (resolution: 1920x1080)



<https://www.studiobinder.com/blog/aspect-ratio/>



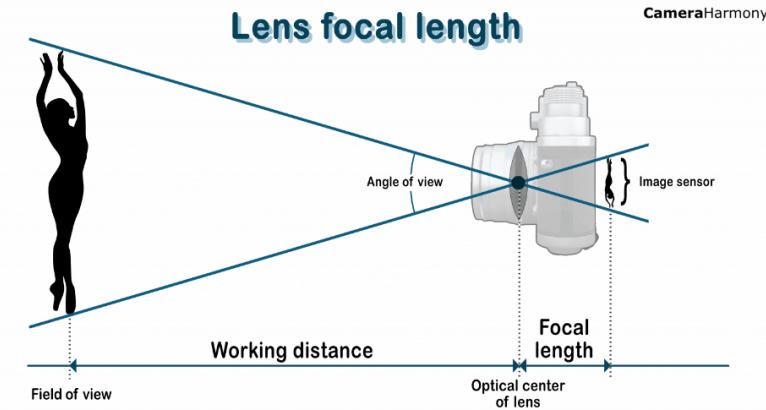
<https://photographylife.com/aspect-ratio>

# Note: film and image aspect ratio

- In computer graphic camera model, sensor (film/canvas) aspect ratio can be different from the image (device) aspect ratio.
- Difference in canvas (film) aspect ratio and the raster image aspect ratio leads to stretching the image in either x or y
- Solution: canvas aspect ratio is often directly computed from the image aspect ratio.
  - Example, if image resolution is 640x480, canvas aspect ratio will be set to 4:3.

# Pinhole camera parameters: recap

- Parameters:
  - Focal length (focal distance)
  - Angle of view (field of view)
  - Film size
  - With two parameters we can always infer the third one.
- Angle of view is determined by:
  - Focal length: longer focal length → narrower angle of view
  - Film size: larger film size → wider angle of view



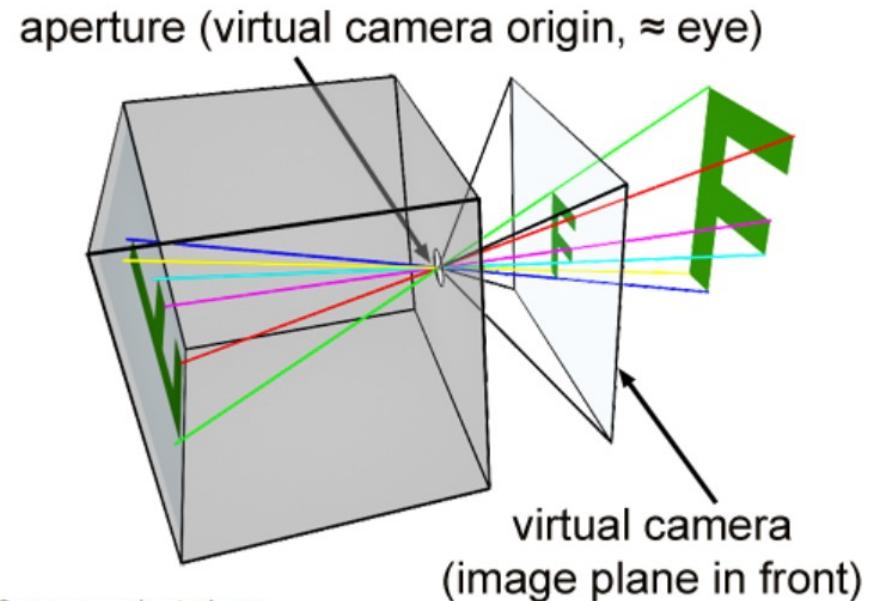
# Camera in rendering

# Camera in rendering

- Pinhole camera model in rendering is used to deliver image similar to those produced by a real camera
  - Size and shape of real and rendered object must be the same
  - photo-realistic rendering which may be combined with live action footage
  - Camera effects enable certain expressive and artistic possibilities
- Pinhole camera model is used in almost all production software
  - Godot: [https://docs.godotengine.org/en/stable/classes/class\\_camera.html](https://docs.godotengine.org/en/stable/classes/class_camera.html)
  - Unity: <https://docs.unity3d.com/ScriptReference/Camera.html>
  - Blender: <https://docs.blender.org/manual/en/latest/render/cameras.html>
  - Houdini: <https://www.sidefx.com/docs/houdini/nodes/obj/cam.html>

# Virtual pinhole camera model

- Real pinhole camera produces images which are upside down
- In computer graphics, this is avoided by setting film plane in front of aperture.
  - This construction gives valid images
  - Aperture is now position of the eye

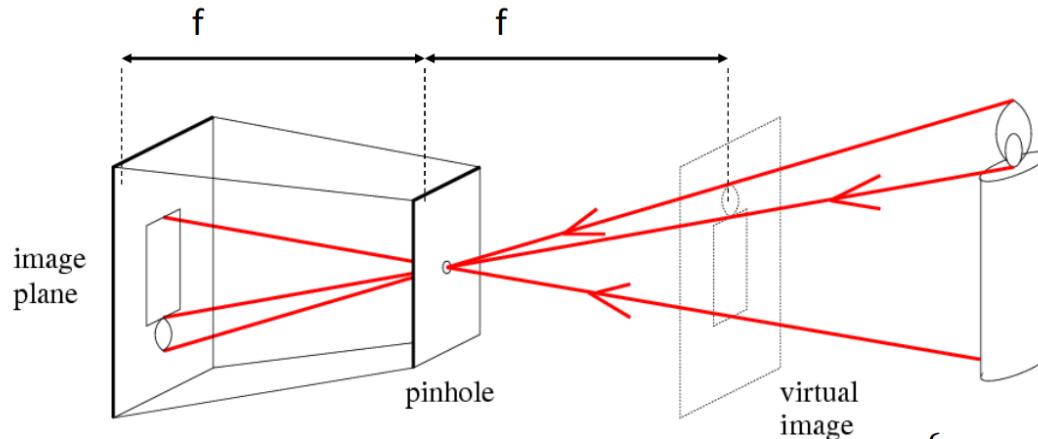


© www.scratchapixel.com

<https://www.scratchapixel.com/lessons/3d-basic-rendering/3d-viewing-pinhole-camera/virtual-pinhole-camera-model>

# Perspective projection

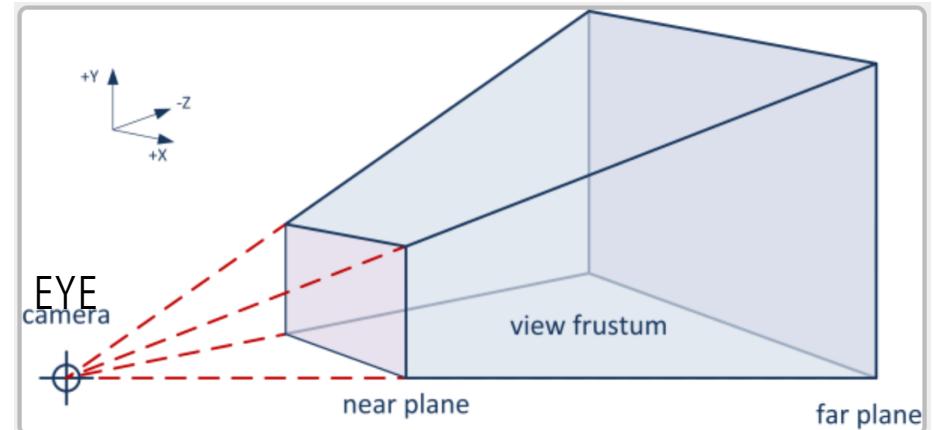
- Image is constructed by following light rays from scene to the eye
- This geometrical problem is called **perspective projection**



[http://vision.stanford.edu/teaching/cs131\\_fall1516/lectures/lecture8\\_camera\\_models\\_cs131.pdf](http://vision.stanford.edu/teaching/cs131_fall1516/lectures/lecture8_camera_models_cs131.pdf)

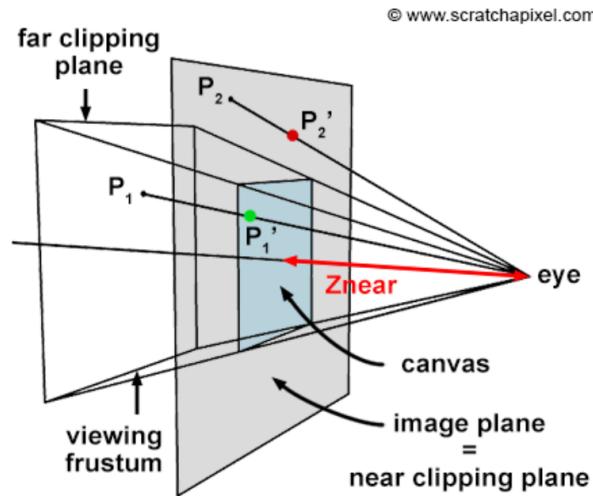
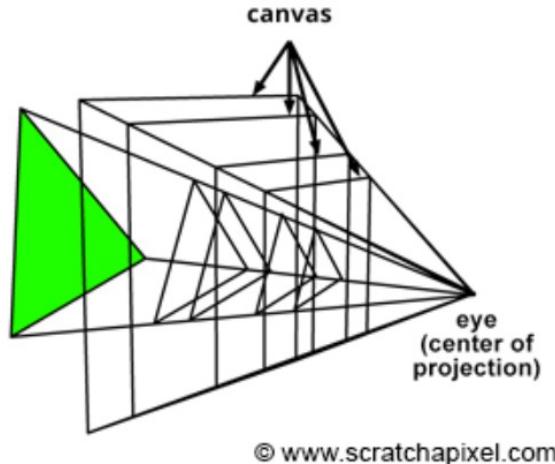
# Viewing frustum

- Near and far clipping planes are virtual planes placed in front of camera and parallel to image plane.
  - Location of clipping planes is determined by camera's line of sight (camera's local z axis)
- Objects in 3D scene closer than near or further than far plane are not visible in image plane.
  - This defines **viewing frustum**
- Near and far clipping planes are needed for resolving precision issues.
  - Used in rasterization-based rendering
  - Not needed for ray-tracing based rendering.



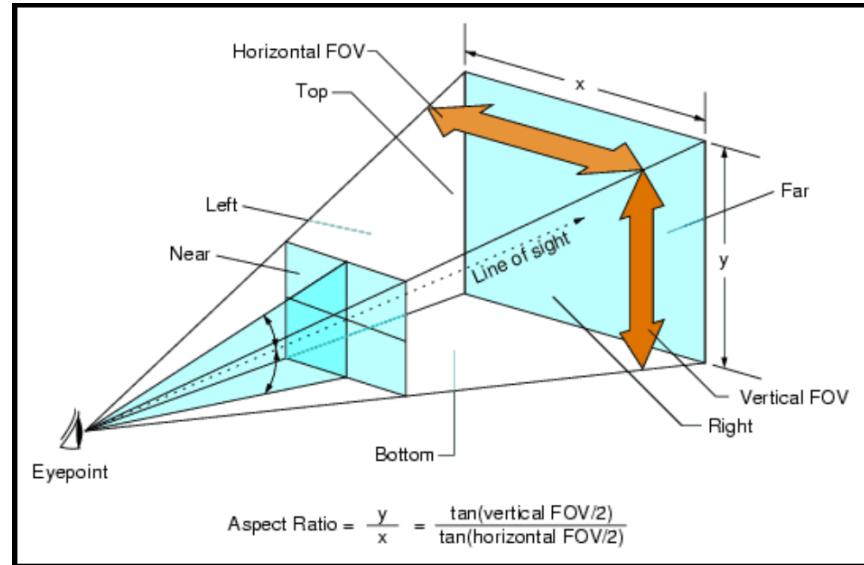
# Image plane size

- Image plane (canvas) can be anywhere on camera's line of sight (camera's local z axis) between near and far clipping planes
  - Smaller distance from eye to image plane → canvas is smaller. And vice versa.



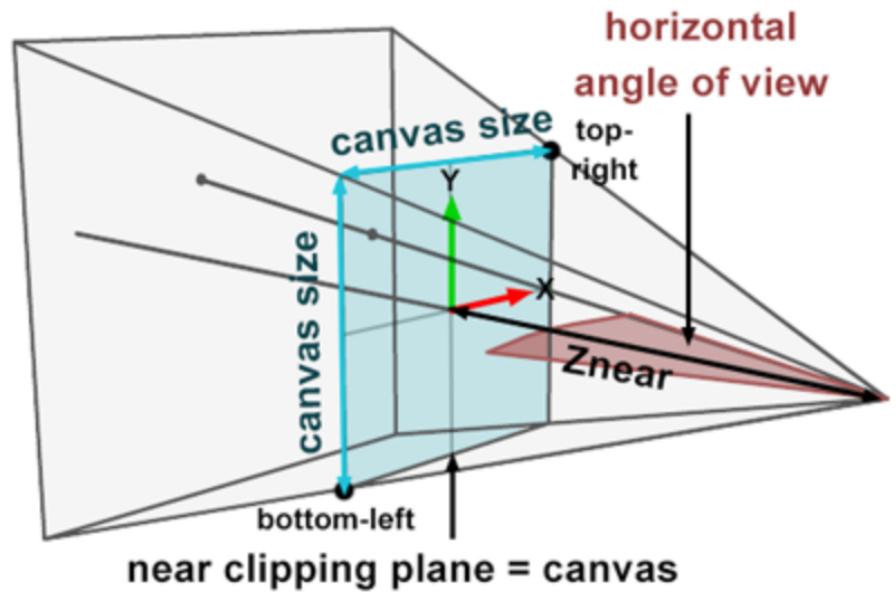
# Recap

- Main elements of pinhole camera are eye and film plane.
- Main parameters of pinhole camera are focal length and sensor (film) size.
- Virtual pinhole camera further introduces near and far clipping planes to tackle precision problems.
  - Viewing frustum shape is defined with aperture point and angle of view
- Virtual image plane (canvas, film) can be anywhere on camera's line of sight
  - We assume it is at near clipping plane
  - Its boundaries are viewing frustum



# Canvas and visibility

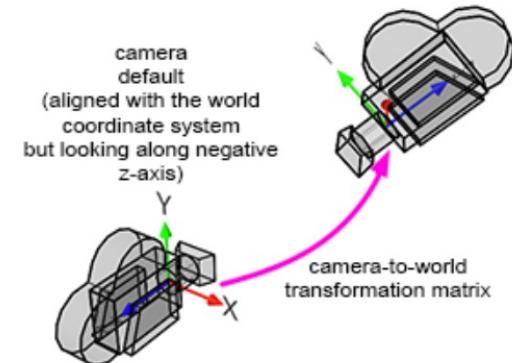
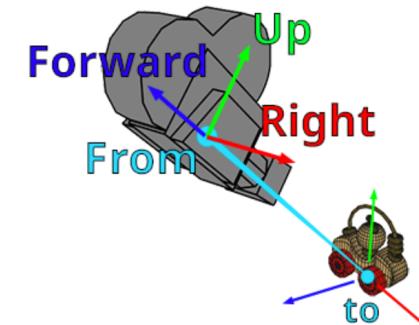
- Point in 3D scene is visible only if lies in canvas limits
- Therefore, point is first projected on image plane and tested for canvas limits to determine visibility
- Canvas is centered about the image plane coordinate system origin.
- The canvas size itself depends on the angle of view and the near clipping plane (since it is placed at near clipping plane)



© www.scratchapixel.com

# Camera movement

- By default camera is located at origin and oriented at negative z axis.
  - Movement of camera is defined with 4x4 transformation matrix.
- Camera direction and orientation is simplified with **look-at method**:
  - Point in space where camera is positioned
  - Point in space where camera is looking
- Look-at method gives **look-at matrix**, which defines 4x4 transformation matrix of camera from its local (camera) space to world space in 3D scene: **camera-to-world** and its inverse **world-to-camera** matrix.

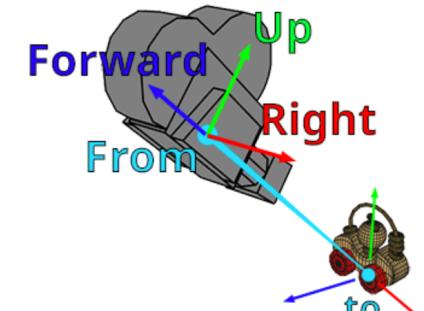


# Camera movement: look at matrix

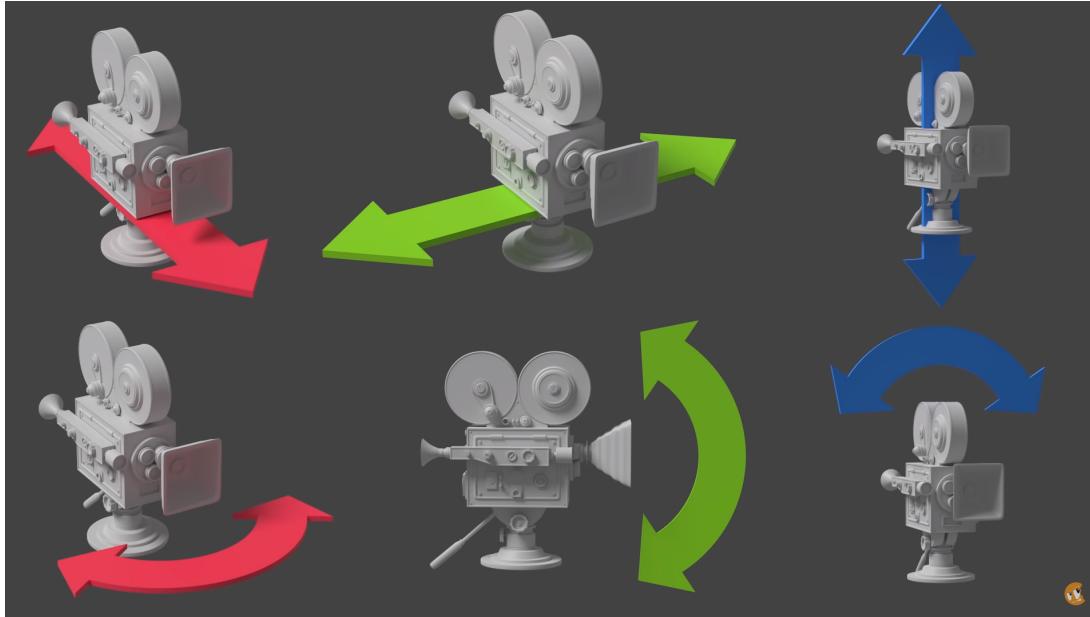
- Forward = normalize(from - to)
- Right = crossProduct(randomVec, forward)
  - RandomVec = (0,1,0)
  - If Forward is close to (0,1,0) or (0,-1,0) take another randomVec
- Up = crossProduct(forward, right)
- 4x4 matrix:

$Right_x$	$Right_y$	$Right_z$	0
$Up_x$	$Up_y$	$Up_z$	0
$Forward_x$	$Forward_y$	$Forward_z$	0
$T_x$	$T_y$	$T_z$	1

$T = \text{from.}$



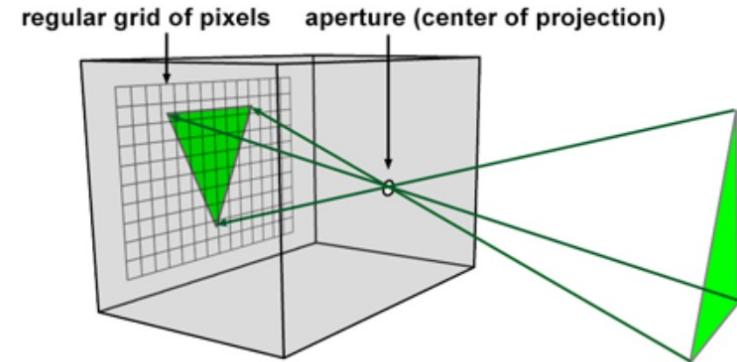
# Camera movement



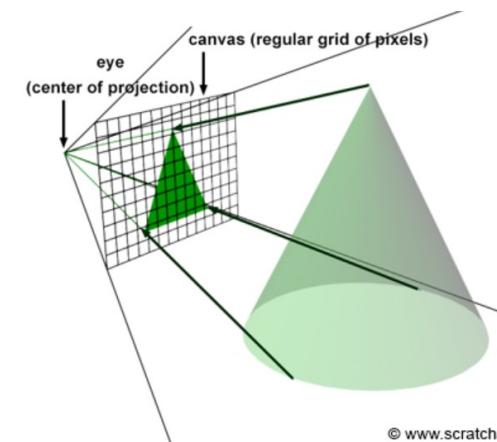
[https://www.youtube.com/watch?v=El0V6fS7mpc&ab\\_channel=CGCookie](https://www.youtube.com/watch?v=El0V6fS7mpc&ab_channel=CGCookie)

# Virtual pinhole camera image formation

- Pinhole image creation:
  - Light rays are emitted from light source, intersect with object surfaces.
  - For each of 3D scene points visible from camera, only some of reflected rays will fall on aperture and strike the film.
  - If we divide film surface into grid of pixels, we obtain digital pinhole camera which we simulate by virtual pinhole camera.
- Virtual pinhole camera follows the same idea but with film place in front of aperture.
- How image is formed using virtual pinhole camera depends on rendering techniques:
  - Ray-tracing-based rendering
  - Rasterization-based rendering



© www.scratchapixel.com

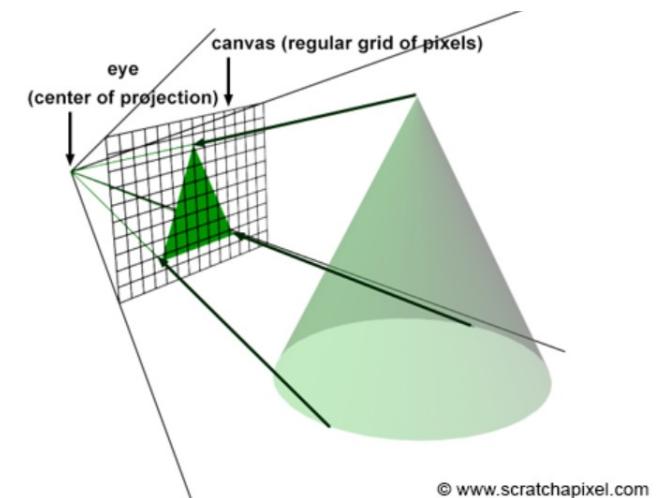


47

© www.scratchapixel.com

# Camera in ray-tracing

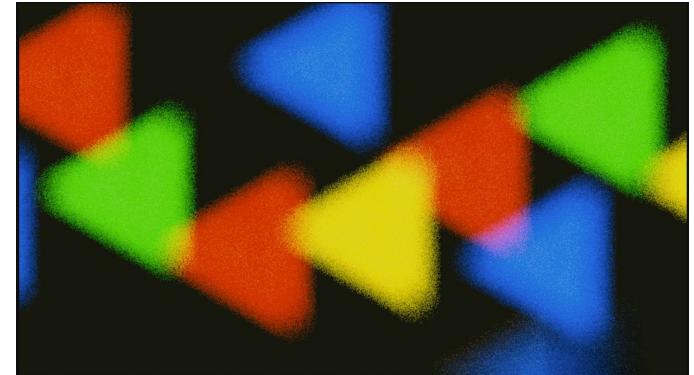
- Real world: light ray reflected from 3D scene point P, passing through aperture is falling on film and is coloring the pixel
- Ray-tracing based rendering simulates this process by associating each pixel with one or multiple rays.
- The process is then reversed:
  - Ray is generated using camera eye position and pixel position in camera image plane.
  - Ray is traced into the 3D scene where it intersects the objects – **ray casting** (foundational method for ray-tracing).
  - Color that is computed from intersecting the object is assigned to pixel



© www.scratchapixel.com

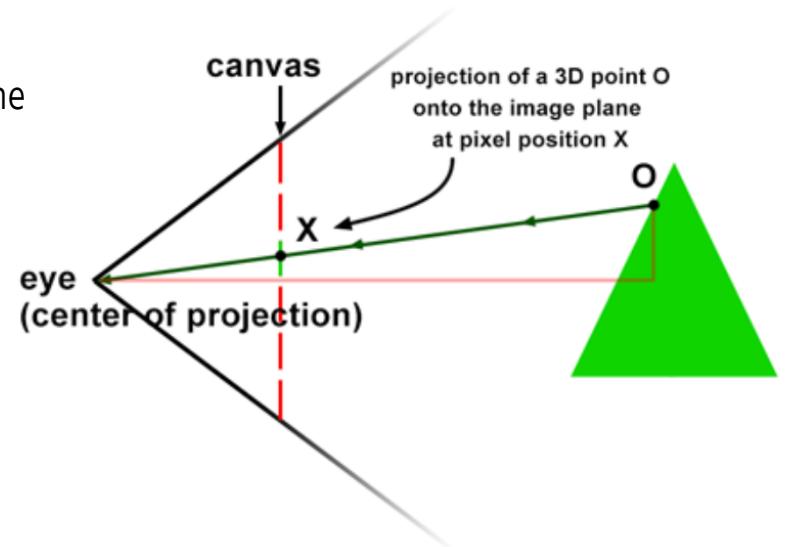
# Camera in ray-tracing

- Ray-tracing is **image-centric**
  - For each pixel in image plane, ray is constructed and traced in 3D scene for intersection and computing color of intersection (shading).
- Aperture is infinitesimally small – a point – from which we generate rays.
  - Image will be perfectly sharp
  - Additional methods are employed from simulating lens effects such as depth of field.



# Camera in rasterization

- Real world: light ray reflected from 3D scene point P, passing through aperture is falling on film and is coloring the pixel
- Rasterization-based rendering is simulating this process by finding a **position of 3D point P from 3D scene in image plane by “projecting” P on image plane.**
  - Perspective projection
- The process of finding pixel coordinates on image plane of 3D point is done as follows:
  - 3D points from world space are transformed to camera space: **world-to-camera matrix**.
  - Apply perspective **projection matrix** or perspective divide.
  - Convert those points to raster space (image plane with pixels).



# Camera in rasterization

- Rasterization is **Object-centric**
  - Rasterization-based rendering starts from 3D scene geometry and projects objects onto image plane to find colored pixels.
- Each projected point from 3D scene has exactly one point on image plane
  - Therefore, image will be perfectly sharp
  - Additional methods are employed to introduce depth of field

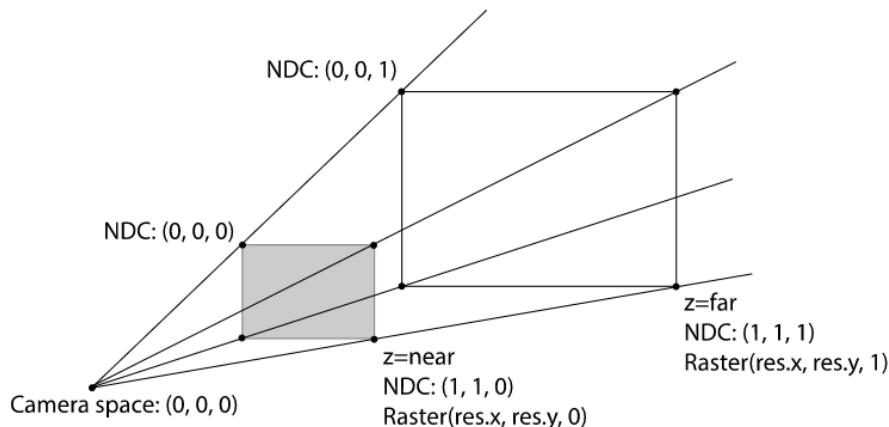
[https://docs.blender.org/manual/en/latest/render/eevee/render\\_settings/depth\\_of\\_field.html](https://docs.blender.org/manual/en/latest/render/eevee/render_settings/depth_of_field.html)



<https://developer.nvidia.com/gpugems/gpugems/part-iv-image-processing/chapter-23-depth-field-survey-techniques>

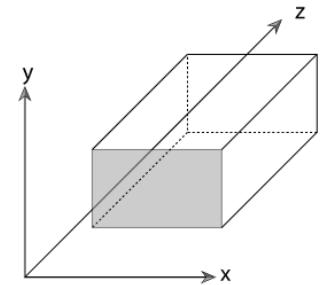
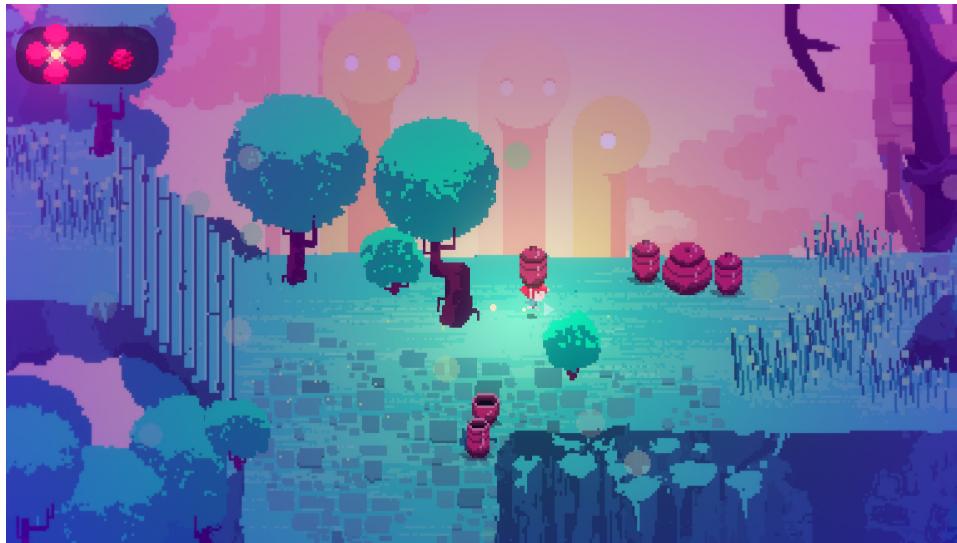
# Perspective camera

- Most often used from film and game rendering

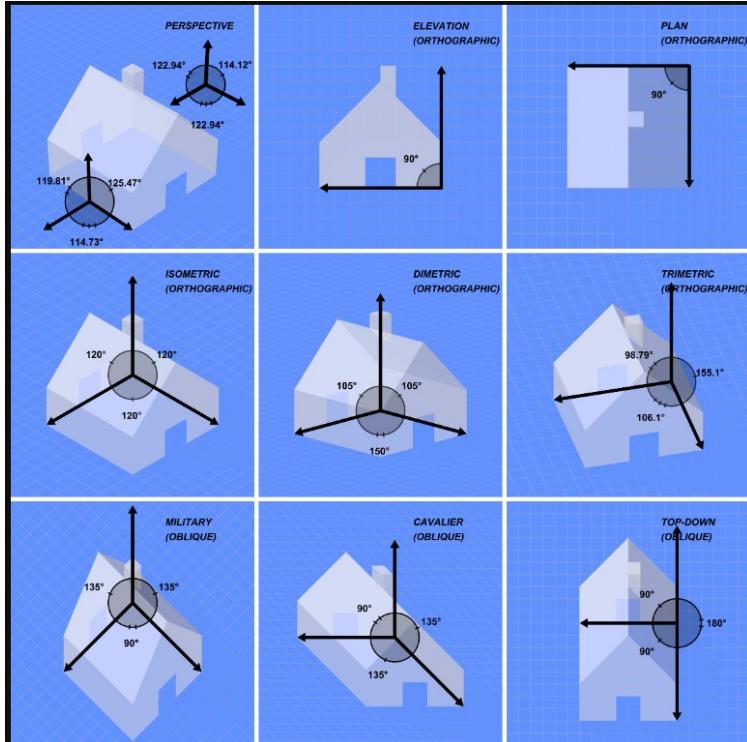


# Orthographic camera

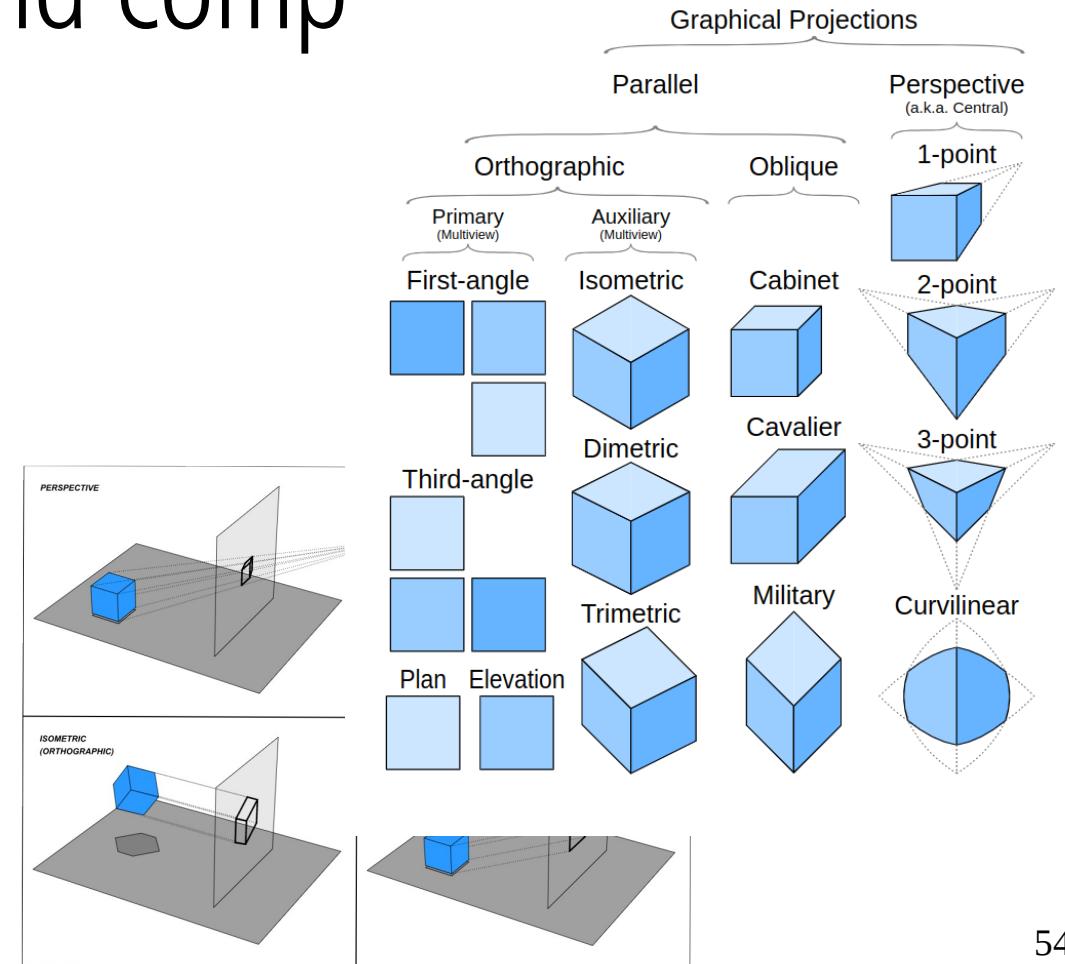
- Useful from modeling in architecture and engineering
- Often used in isometric games



# Other projections and comparisons

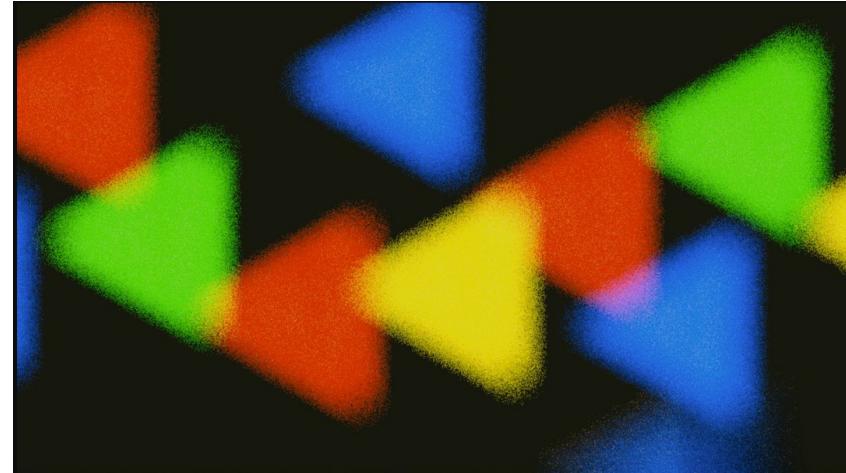


[https://en.wikipedia.org/wiki/Orthographic\\_projection](https://en.wikipedia.org/wiki/Orthographic_projection)



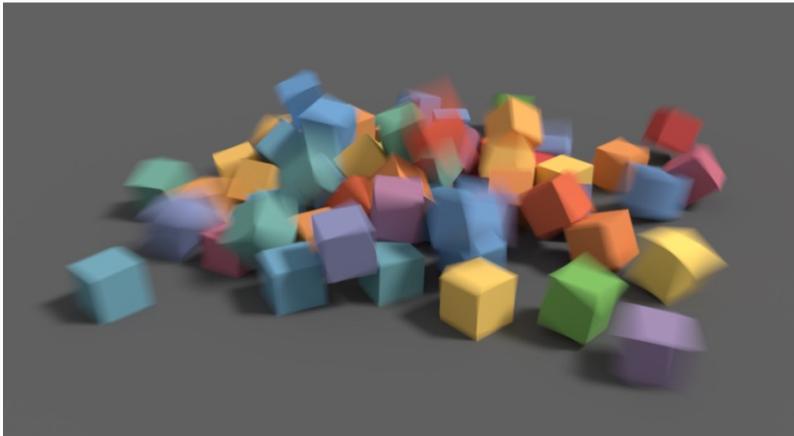
# Simulating lens camera

- Real-world cameras transmit light through a lens that bends and focuses it onto the sensor. Because of this, objects that are a certain distance away are in focus, but objects in front and behind that are blurred.



# Exposure in simulated camera

- Exposure (time in which aperture is opened) in rendering is not used as camera parameter.
  - Exposure is parameter which control brightness of already rendered image
- Shutter speed (determining exposure) is used for motion blur.



[https://docs.blender.org/manual/en/latest/render/cycles/render\\_settings/motion\\_blur.html](https://docs.blender.org/manual/en/latest/render/cycles/render_settings/motion_blur.html)



Different exposures of the same render:  
[https://docs.blender.org/manual/en/latest/render/color\\_management.html](https://docs.blender.org/manual/en/latest/render/color_management.html)

# More into topic

- More about film, aspect ratio:

<https://www.scratchapixel.com/lessons/3d-basic-rendering/3d-viewing-pinhole-camera/how-pinhole-camera-works-part-2>

- Simulating camera lenses:

[https://www.pbr-book.org/3ed-2018/Camera\\_Models/Realistic\\_Cameras](https://www.pbr-book.org/3ed-2018/Camera_Models/Realistic_Cameras)

# Summary questions

- [https://github.com/lorentzo/IntroductionToComputerGraphics/tree/main/lectures/9\\_light](https://github.com/lorentzo/IntroductionToComputerGraphics/tree/main/lectures/9_light)

# Literature

- <https://github.com/lorentzo/IntroductionToComputerGraphics>
- ScrachAPixel lesson:  
<https://www.scratchapixel.com/lessons/3d-basic-rendering/3d-viewing-pinhole-camera>