# Viewing

Fall 2023 11/2/2023 Kyoung Shin Park Computer Engineering Dankook University

## **Camera in Unity**

### Camera

- A Unity scene represents GameObjects in a three-dimensional space. Since the viewer's screen is two-dimensional, Unity needs to capture a view and "flatten" it for display. It does this using cameras.
- In Unity, you create a camera by adding a Camera component to a GameObject.



### **Camera Components**

Inspector	🔀 Navigation								а	:
Mair	n Camera							s	tati	c▼
Tag Mai	nCamera		▼ Lay	/er	Default					
Trane	form							0	- <b>+</b>	:
	Iom	v 👝			-		10	•	-+-	
Position		X O		) Y 1 V	1	Z	-10			4
Rotation		X 0		ÌV	0	4	0			
Scale		<u>^  </u>		Ţ	<u> </u>	2				
🔻 💶 🖌 Came	ra							0	갍	
Clear Flags		Skybo	x							
Background										ø
Culling Mask		Every	thing							T
Projection		Persp	ective							
FOV Axis		Vertic	al							
Field of View			•					60	)	
Physical Carr	nera									
Clipping Plan	es	Near	0.3							
		Far	1000							
Viewport Rec		X 0		Y	0					
		W 1		H	1					
Depth		-1								
Rendering Pa	th	Use G	raphics S	Setti	ings					
Target Textu	re	None	(Render 1	ſext	ture)					
Occlusion Cu	lling	~								
HDR		Use G	raphics S	Setti	ings					
MSAA		Use G	raphics S	Setti	ings					T
Allow Dynam	ic Resolution									
Target Displa	У	Displa	iy 1							

### **Camera Components**

Property:	Function:
Clear Flags	Determines which parts of the screen will be cleared. This is handy when using multiple Cameras to draw different game elements. Skybox is the default setting.
Background	The color applied to the remaining screen after all elements in view have been d rawn and there is no <b>skybox</b> .
Culling Mask	Includes or omits layers of objects to be rendered by the Camera. Assigns layers to your objects in the Inspector.
Projection	Toggles the camera's capability to simulate perspective.
Perspective	Camera will render objects with perspective intact.
Orthographic	Camera will render objects uniformly, with no sense of perspective. <b>NOTE:</b> Deferr ed rendering is not supported in Orthographic mode. <b>Forward rendering</b> is alw ays used.
<b>Size</b> (when Orthographic i s selected)	The <b>viewport</b> size of the Camera when set to Orthographic.
<b>FOV Axis</b> (when Perspective e is selected)	Field of view axis.
Horizontal	The Camera uses a horizontal field of view axis.
Vertical	The Camera uses a vertical field of view axis.
Field of view (when Persp ective is selected)	The Camera's view angle, measured in degrees along the axis specified in the <b>FO V Axis</b> drop-down.

### **UI Camera Setting in Unity**

### Capsule(Player) contains Main Camera & UI Camera

- Main Camera uncheck UI in CullMask, set Depth to O
- UI Camera select *Depth only* in Clear Flags, *check only UI* in Cull Mask, set Depth to 1

'≡ Hierarchy		а	
+ - All			
<b>√\$</b> Sa	ampleScene		
\ A A A A A A A A A A A A A A A A A A A	Directional Light Demo_Objects		
∇Ô	Capsule		
¢ ₽ ■	<ul> <li>Main Camera</li> <li>UlCamera</li> <li>Canvas</li> <li>Image</li> <li>Button</li> <li>EventSystem</li> </ul>		

O Inspector		a :
Main Camera		Static 🕶
Tag MainCamera	▼ Layer Default	<b>•</b>
🔻 🙏 Transform		0 ‡ :
Position Rotation Scale	x 0 Y 0.53 x 0 Y 0 x 1 Y 1	Z 0 Z 0 Z 1
🔻 🗖 🗸 Camera		0 ‡ :
Clear Flags Background Culling Mask	Skybox Mixed	*   #   *
Projection FOV Axis Field of View Physical Camera	Nothing Everything V Default TransparentFX	• • 60
Clipping Planes Viewport Rect	Water UI	
Depth	0	
Rendering Path Target Texture Occlusion Culling	Use Graphics Settings None (Render Texture)	• ⊙
HDR MSAA	Use Graphics Settings Use Graphics Settings	* *
Allow Dynamic Resolution		
Target Display	Display 1	▼
🎧 🗹 Audio Listener		0 ≓ :
	Add Component	

Inspector		а	: 🗆 ×
UlCamera			Static 🔻
Tag Untagged	▼ Layer De	efault	•
🔻 🙏 🛛 Transform		(	9 ∓ :
Position	X 0 Y -1.	.41 Z O	
Rotation	X 0 Y 0	Z 0	
	X 1 Y 1	Z 1	
🔻 💶 🗹 Camera	P		9 ≓ :
Clear Flags	Depth only		
Culling Mask	UI		•
Projection	Nothing		
FOV Axis	Everything		-
Field of View	Default		60
Physical Camera	TransparentFX		
Clipping Dispas	Ignore Raycast		
Clipping Planes	Water		
Viewport Pect	VII		
viewportreet	W1 H1	_	,
Denth			
Depth Depdering Dath	1	0	
Target Texture	None (Dender Textury	o)  9	
		e)	
	<ul> <li>Use Graphics Setting</li> </ul>		
MSAA	Use Graphics Setting	IS	•
Allow Dynamic Resolution			
Tanat Dianlass	Display 1		
l arget Display	Display I		
🎧 🗹 Audio Listener			9 ≓ :
	Add Component		

### **UI Camera Setting in Unity**

### Canvas – set Layer to UI, set Render Mode to Screen Space – Camera, set Render Camera to UICamera

- Then, Main Camera renders a scene
- while UI Camera renders UI only.



Canvas   Tag   Untagged   Layer   Layer   Width   Height   100   Width   Height   1008   Scale   X   0   Y   0   0   0   0   0   0   Width   Height   1008   Scale   X   0.5   Y   Y   Y   Y   Y   Y	O Inspector			a:□×
Tag Untagged       Layer UI         Image: Constant Pixel Size       Image: Constant Pixel Size         Sorder Added       Constant Pixel Size         Image: Constant Pixel Size       Image: Constant Pixel Size         Image:	Canvas			
Image: Some values driven by Canvas.         Some values driven by Canvas.         Pos X       Pos Y         0       0         Width       Height         1008       544         Anchors       Pivot         Pivot       X         X       0.5         Y       0.2         Scale       X         V       0.2122611         Y       0.2122611         Plane Distance       100         Sorting Layer       0         Order in Layer       0         Additional Shader Channels       Nothing         V       Canvas Scaler       Image: Im	Tag Untagged	👻 La	yer UI	-
Some values driven by Canvas.       Pos X       Pos Y       Pos Z         0       0       100         Width       Height       100         Width       Height       100         Nothors       Y       0.5       Y       0.5         Rotation       X       0.5       Y       0.5         Scale       X       0.2122611       Y       0.2122611         Canvas       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         Render Mode       Screen Space - Camera       Image: Canvas       Image: Canvas       Image: Canvas         Pixel Perfect       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         Plane Distance       100       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         Viditional Shader Channels       Nothing       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         Viditional Shader Channels       Nothing       Image: Canvas       Image: Canvas       Image: Canvas         Viditional Shader Channels       Nothing       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         Scale Factor <t< th=""><th>🔻 🛟 🛛 Rect Transform</th><th></th><th></th><th>0 ‡ :</th></t<>	🔻 🛟 🛛 Rect Transform			0 ‡ :
Pos X       Pos Y       Pos Z         0       0       100         Width       Height       1008         1008       544       III R         Anchors       Pivot       X       0.5       Y       0.5         Rotation       X       0       Y       0       Z       0         Scale       X       0.5       Y       0.2122611       Z       0         Scale       X       0.2122611       Y       0.2122611       Z       0.2122611         Render Mode       Screen Space - Camera       Image: Camera       Image: Camera       Image: Camera       Image: Camera       Image: Camera         Pixel Perfect       Image: Camera       Ima				
o o 100   Width Height   1008 544   Image: Streen Space - Camera   <				Pos Z
Width       Height         1008       544         Pivot       X         Pivot       X         X       0       Y         Scale       X         X       0       Y         Scale       X       0.2122611         Y       0.2122611       Y       0.2122611         Scale       X       0.2122611       Y       0.2122611         Plane Distance       100       Image: Canvas       Image: Canvas       Image: Canvas         Plane Distance       100       Image: Canvas       Image: Canvas       Image: Canvas         Order in Layer       0       Image: Canvas       Image: Canvas       Image: Canvas         Vill Scale Mode       Constant Pixel Size       Image: Canvas       Image: Canvas         Vill Scale Mode       Constant Pixel Size       Image: Canvas       Image: Canvas         Vill Scale Factor       1       Image: Canvas       Image: Canvas       Image: Canvas         <				100
Anchors       Pivot       X       0.5       Y       0.5         Rotation       X       0       Y       0       Z       0         Scale       X       0.2122611       Y       0.2122611       Z       0.2122611         Image: Canvas       Im				513 D
Pivot       X 0.5       Y 0.5         Rotation       X 0       Y 0       Z 0         Scale       X 0.2122611       Y 0.2122611       Z 0.2122611         Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         Render Mode       Screen Space - Camera       Image: Canvas       Image: Canvas         Pixel Perfect       Image: Canvas       Image: Canvas       Image: Canvas         Plane Distance       100       Image: Canvas       Image: Canvas         Sorting Layer       Default       Image: Canvas       Image: Canvas         Order in Layer       0       Image: Canvas       Image: Canvas         Order in Layer       0       Image: Canvas       Image: Canvas         Order in Layer       0       Image: Canvas       Image: Canvas         Vill Scale Mode       Constant Pixel Size       Image: Canvas       Image: Canvas         Vill Scale Mode       Constant Pixel Size       Image: Canvas       Image: Canvas         Scale Factor       1       Image: Canvas       Image: Canvas       Image: Canvas         Vill Scale Mode       Constant Pixel Size       Image: Canvas       Image: Canvas       Image: Canvas         Vill Scale Factor       1       Image: Canvas<	► Anchore			
Rotation X 0 Y 0 Z 0   Scale X 0.2122611 Y 0.2122611 Z 0.2122611   Image: Canvas Image: Canvas Image: Canvas Image: Canvas Image: Canvas Image: Canvas   Render Mode Screen Space - Camera Image: Canvas Image: Canvas Image: Canvas Image: Canvas   Plane Distance 100 Sorting Layer Default Image: Canvas Image: Canvas   Order in Layer 0 Image: Canvas Image: Canvas Image: Canvas   Order in Layer 0 Image: Canvas Image: Canvas Image: Canvas   Order in Layer 0 Image: Canvas Image: Canvas Image: Canvas   Vill Scale Mode Constant Pixel Size Image: Canvas Image: Canvas   Scale Factor 1 Image: Canvas Image: Canvas   Ignore Reversed Graphics Image: Canvas Image: Canvas   Blocking Objects None Image: Canvas Image: Canvas   Blocking Mask Everything Image: Canvas	Pivot			
Rotation       × 0       Y 0       Z 0         Scale       × 0.2122611       Y 0.2122611       Z 0.2122611         Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         Render Mode       Screen Space - Camera       Image: Canvas       Image: Canvas         Pixel Perfect       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         Plane Distance       100       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         Sorting Layer       Default       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         Order in Layer       0       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         Additional Shader Channels       Nothing       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         VI Scale Mode       Constant Pixel Size       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         Scale Factor       1       Image: Canvas       Image: C				) - <b></b>
Scale       × 0.2122611       ¥ 0.2122611       Z 0.2122611         Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         Render Mode       Screen Space - Camera       Image: Canvas       Image: Canvas         Pixel Perfect       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         Plane Distance       100       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         Sorting Layer       Default       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         Order in Layer       0       Image: Canvas       Canvas       Image: Canvas       Image: Canvas         Additional Shader Channels       Nothing       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         VI Scale Mode       Constant Pixel Size       Image: Canvas       Image: Canvas       Image: Canvas         Scale Factor       1       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         VI Scale Mode       Constant Pixel Size       Image: Canvas       Image: Canvas       Image: Canvas         Scale Factor       1       Image: Canvas       Image: Canvas       Image: Canvas       Image: Canvas         VI Scal	Rotation			Z 0
Image: Canvas       Image: Canvas         Render Mode       Screen Space - Camera         Pixel Perfect       Image: Canvas         Render Camera       ImulCamera (Camera)         Plane Distance       100         Sorting Layer       Default         Order in Layer       0         Additional Shader Channels       Nothing         Image: Canvas Scaler       Image: Canvas Scaler         UI Scale Mode       Constant Pixel Size         Scale Factor       1         Reference Pixels Per Unit       100         Image: Canvas Graphic Raycaster       Image: Canvaster         Script       Image: Graphic Raycaster         Ignore Reversed Graphics       Image: Canvaster         Blocking Objects       None         Blocking Mask       Everything	Scale			Z 0.2122611
Render Mode       Screen Space - Camera         Pixel Perfect         Render Camera         Plane Distance         100         Sorting Layer         Order in Layer         Order in Layer         Order in Layer         O         Additional Shader Channels         Nothing         ✓ Canvas Scaler         ✓ Canvas Scaler         Ø ᅷ I         Ul Scale Mode         Scale Factor         Reference Pixels Per Unit         100         ♥ ♥ ♥ Graphic Raycaster         Ø \$\frackleft \setmathinspace         Ignore Reversed Graphics         Blocking Objects         Blocking Mask	🔻 🗐 🗸 Canvas			0 ± ÷
Pixel Perfect         Render Camera         Plane Distance         100         Sorting Layer         Order in Layer         O         Additional Shader Channels         Nothing         Image: Constant Pixel Size         V Canvas Scaler         Image: Constant Pixel Size         VI Scale Mode         Constant Pixel Size         Scale Factor         1         Reference Pixels Per Unit         100         Image: Constant Pixel Size         Script         Ignore Reversed Graphics         Blocking Objects         Blocking Mask	Render Mode	Screen Space	- Camera	-
Render Camera       ■ UlCamera (Camera)         Plane Distance       100         Sorting Layer       Default         Order in Layer       0         Additional Shader Channels       Nothing         ▼ Canvas Scaler       ♥ ‡ :         UI Scale Mode       Constant Pixel Size         Scale Factor       1         Reference Pixels Per Unit       100         ♥ ᡎ ♥ Graphic Raycaster       ♥ ‡ :         Script       ৸ GraphicRaycaster         Ignore Reversed Graphics       None         Blocking Objects       None         Blocking Mask       Everything	Pixel Perfect			
Plane Distance       100         Sorting Layer       Default         Order in Layer       0         Additional Shader Channels       Nothing         Image: Canvas Scaler       Image: Constant Pixel Size         UI Scale Mode       Constant Pixel Size         Scale Factor       1         Reference Pixels Per Unit       100         Image: Canvas Graphic Raycaster       Image: Canvaster         Ignore Reversed Graphics       Image: Canvaster         Blocking Objects       None         Blocking Mask       Everything	Render Camera	UICamera (	Camera)	$\odot$
Sorting Layer     Default       Order in Layer     0       Additional Shader Channels     Nothing       Image: Canvas Scaler     Image: Constant Pixel Size       UI Scale Mode     Constant Pixel Size       Scale Factor     1       Reference Pixels Per Unit     100       Image: Canvas Graphic Raycaster     Image: Canvaster       Script     Image: Graphic Raycaster       Ignore Reversed Graphics     None       Blocking Objects     None       Blocking Mask     Everything	Plane Distance	100		
Order in Layer       0         Additional Shader Channels       Nothing         Image: Canvas Scaler       Image: Constant Pixel Size         UI Scale Mode       Constant Pixel Size         Scale Factor       1         Reference Pixels Per Unit       100         Image: Law Computer Raycaster       Image: Constant Pixel Size         Script       Image: Constant Pixel Size         Ignore Reversed Graphics       ✓         Blocking Objects       None         Blocking Mask       Everything	Sorting Layer	Default		<b>•</b>
Additional Shader Channels       Nothing         Image: Canvas Scaler       Image: Constant Pixel Size         UI Scale Mode       Constant Pixel Size         Scale Factor       1         Reference Pixels Per Unit       100         Image: Large Canvas Scaler       Image: Canvas Scale Canvas Scal	Order in Layer	0		
Image: Canvas Scaler       Image: Constant Pixel Size         UI Scale Mode       Constant Pixel Size         Scale Factor       1         Reference Pixels Per Unit       100         Image: Lambda Capabic Raycaster       Image: Lambda Capabic Raycaster         Script       Image: Lambda Capabic Raycaster         Ignore Reversed Graphics       ✓         Blocking Objects       None         Blocking Mask       Everything	Additional Shader Channels	Nothing		•
UI Scale Mode       Constant Pixel Size         Scale Factor       1         Reference Pixels Per Unit       100         ↓ ✓ Graphic Raycaster       @ ≠ :         Script       ¤ GraphicRaycaster       ©         Ignore Reversed Graphics       ✓         Blocking Objects       None       ▼         Blocking Mask       Everything       ▼	🔻 🔳 🖌 Canvas Scaler			0 ≓ :
Scale Factor       1         Reference Pixels Per Unit       100         ♥ ↓ ♥ Graphic Raycaster       ● ↓ !         Script       □ GraphicRaycaster       ●         Ignore Reversed Graphics       ♥       ●         Blocking Objects       None       ●         Blocking Mask       Everything       ●	UI Scale Mode	Constant Pixel	Size	-
Reference Pixels Per Unit       100         ↓↓        Graphic Raycaster         Script       ↓ GraphicRaycaster         Ignore Reversed Graphics       ✓         Blocking Objects       None         Blocking Mask       Everything	Scale Factor	1		
↓ ✓ Graphic Raycaster     ● ↓ :       Script     □ GraphicRaycaster     ○       Ignore Reversed Graphics     ✓       Blocking Objects     None     ✓       Blocking Mask     Everything     ✓	Reference Pixels Per Unit	100		
Script     It GraphicRaycaster       Ignore Reversed Graphics       Blocking Objects       Blocking Mask   Everything	🔻 🛱 🖌 Graphic Raycaster			0 ≓ :
Ignore Reversed Graphics Blocking Objects None Blocking Mask Everything				0
Blocking Objects None   Blocking Mask Everything	Ignore Reversed Graphics	<ul> <li>Image: A set of the set of the</li></ul>		
Blocking Mask Everything 🔹	Blocking Objects	None		•
	Blocking Mask	Everything		•

### **Viewer's Perspective**

Before rendering the environment on the screen we consider the camera input such as (field of view, Projection mode [Orthographic or Perspective]).



Perspective camera



Orthographic camera

https://docs.unity3d.com/Manual/CamerasOverview.html

### **Field of View**

• A wide field of view shows more of the scene.



https://gamedevbeginner.com/how-to-zoom-a-camera-in-unity-3-methodswith-examples/

### **Field of View**

A narrow field of view shows less of the camera image, zooming it in scene.



https://gamedevbeginner.com/how-to-zoom-a-camera-in-unity-3-methodswith-examples/

### **Camera Components**

Property:	Function:
<b>Clipping Planes</b>	Distances from the camera to start and stop rendering.
Near	The closest point relative to the camera that drawing will occur.
Far	The furthest point relative to the camera that drawing will occur.
Viewport Rect	Four values that indicate where on the screen this camera view will be drawn. Measured in Viewport Coordinates (values 0–1).
X	The beginning horizontal position that the camera view will be drawn.
Y	The beginning vertical position that the camera view will be drawn.
₩ (Width)	Width of the camera output on the screen.
H (Height)	Height of the camera output on the screen.
Depth	The camera's position in the draw order. Cameras with a larger value will be d rawn on top of cameras with a smaller value.
<b>Rendering Path</b>	Options for defining what rendering methods will be used by the camera.
Forward	Forward is the traditional rendering path.
Deferred Lighting	<i>Deferred Shading</i> is the rendering path with the most lighting and shadow fidelity, and is best suited if you have many realtime lights. It requires a certain level of hardware support.
Legacy Vertex Lit	<i>Legacy Vertex Lit</i> is the rendering path with the lowest lighting fidelity and no support for realtime shadows. It is a subset of Forward rendering path.
Legacy <i>Deferred</i>	<i>Legacy Deferred (light prepass)</i> is similar to Deferred Shading, just using a different technique with different trade-offs.

### Viewport

### Viewport

The space set inside the window. Drawing is restricted to inside the viewport.



Camera1 Viewport Rect X:0 Y: 0 W:0.5 H:1 Camera2 Viewport Rect X:0.5 Y: 0 W:0.5 H:1

### **Camera Components**

Property:	Function:
Target Texture	Reference to a Render Texture that will contain the output of the Camera view . Setting this reference will disable this Camera's capability to render to the sc reen.
Occlusion Culling	Enables Occlusion Culling for this camera. Occlusion Culling means that objec ts that are hidden behind other objects are not rendered, for example if they are behind walls.
Allow HDR	Enables High Dynamic Range rendering for this camera.
Allow MSAA	Enables multi sample antialiasing for this camera.
Allow Dynamic Resoluti on	Enables Dynamic Resolution rendering for this camera.
Target Display	Defines which external device to render to. Between 1 and 8.

- Occlusion Culling is a feature that disables rendering of objects when they are not currently seen by the camera because they are obscured (occluded) by other objects.
- This does not happen automatically in 3D computer graphics since most of the time objects farthest away from the camera are drawn first and closer objects are drawn over the top of them (this is called "overdraw").
- Occlusion Culling is different from Frustum Culling.
- Frustum Culling only disables the renderers for objects that are outside the camera's viewing area but does not disable anything hidden from view by overdraw.
- Note that when you use Occlusion Culling you will still benefit from Frustum Culling.

A maze-like indoor level. This normal scene view shows all visible Game Objects.



#### https://docs.unity.com/Manual/OcclusionCulling.html

### Regular frustum culling renders all Renderers within the Camera's view.



#### https://docs.unity.com/Manual/OcclusionCulling.html

Occlusion culling removes Renderers that are entirely obscured by nearer Renderer.



#### https://docs.unity.com/Manual/OcclusionCulling.html

### **Hidden Surface**

- □ Hidden surfaces provides the occlusion depth cue.
- In computer graphics, the term occlusion refers to objects that are close to the viewer to occlude objects that are far from the viewer.
- In the graphics pipeline, hidden surface removal is performed before shading and rasterization with occlusion culling.



### **Hidden Surface Removal**

### Hidden Surface Removal Algorithm

- Object space technique compare objects or parts of objects to determine which side and line are not visible as a whole.
  - Depth-sorting algorithm After aligning each side of the polygon according to the depth, it is drawn from the far one to front one. Also known as Painter's algorithm.
  - Binary Space Partitioning (BSP) tree Using BSP tree, the space is continuously partitioned by separating front and back according to the viewer direction.
- Image space technique act as part of the projection, and visibility is determined in units of points at the location of object pixels on each projection line.
  - Z-buffer (depth buffer) This is the most commonly used image space technique. By examining the visibility of an object in pixels, it draws the value of the plane with the smallest z (depth) value. We need a depth buffer (z-buffer) to store the z-value.
  - Ray-casting It projects light (ray) through each pixel on the projection surface at the viewpoint, selects the object that first meets this light and draws the pixel. It is an effective hidden surface removal algorithm for curved surface.

### **Z-buffer**





Color buffer

Z-buffer (depth buffer)

### **Z-buffer**

- Polygon rendering means eventually being filled with pixels.
- The color buffer contains RGB color per pixel to be drawn.
- The depth buffer (Z-buffer) has depth information per pixel to be drawn.



Color buffer



Depth buffer

### **Z-buffer Algorithm**

- Whenever a new pixel is drawn, the Z-buffer algorithm compares the new depth information with the previous depth information in the z-buffer.
- Polygons can be drawn in any direction and can intersect.



Color buffer



Depth buffer

## **Depth Fighting**

- **The depth value of the Z-buffer has a limited resolution.**
- The overlap of polygons with a depth value that is very close to the depth buffer creates "depth-fighting".
- This is a phenomenon that occurs due to "floating point round-off errors" when polygons are drawn, where random parts of polygons flight for rendering each other.



## **Depth Fighting**

- There is very high precision at the *near* plane, but very little precision at the *far* plane.
- If the range [-n, -f] is getting larger, it causes a depth precision problem (z-fighting); a small change of z<sub>e</sub> around the *far* plane does not affect on z<sub>n</sub> value.
- The distance between *n* and *f* should be short as possible to minimize the depth buffer precision problem.



## **Depth Fighting**

- Z-fighting can be reduced through the use of a higher resolution depth buffer, by z-buffering in some scenarios, or by simply moving the polygons further apart.
- Z-fighting that is caused by insufficient precision in the depth buffer can be resolved by simply reducing the visible distance in the world. This reduces the distance between the near and far planes and solves the precision issue.
- Another technique that is utilized to reduce or completely eliminate Z-fighting is switching to a logarithmic Z-buffer, reversing Z. Due to the way they are encoded, floating-point numbers have much more precision when closer to 0. Here, reversing Z leads to more precision when storing the depth of very distant objects, hence greatly reducing Z-fighting.

### Planar Shadow [J. Blinn, 88]



$$\begin{bmatrix} s_x \\ 0 \\ s_z \\ 1 \end{bmatrix} = \begin{bmatrix} l_y & -l_x & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -l_z & l_y & 0 \\ 0 & -1 & 0 & l_y \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix}$$

$$\mathbf{s} = l + t(p - l)$$
$$\mathbf{n} \cdot \mathbf{s} + d = 0$$

 $t = \frac{l_y}{l_y - p_y}$ 

### Planar Shadow [J. Blinn, 88]

$$\mathbf{s} = l + \frac{l_y}{l_y - p_y} (p - l)$$

$$s_x = l_x + \frac{l_y}{l_y - p_y} (p_x - l_x)$$

$$s_y = 0$$

$$s_z = l_z + \frac{l_y}{l_y - p_y} (p_z - l_z)$$

$$s_{x} = l_{x} + \frac{l_{y}}{l_{y} - p_{y}}(p_{x} - l_{x}) \qquad s_{z} = l_{z} + \frac{l_{y}}{l_{y} - p_{y}}(p_{z} - l_{z})$$

$$= \frac{l_{x}(l_{y} - p_{y}) + l_{y}(p_{x} - l_{x})}{l_{y} - p_{y}} \qquad = \frac{l_{z}(l_{y} - p_{y}) + l_{y}(p_{z} - l_{z})}{l_{y} - p_{y}}$$

$$= \frac{l_{y}(p_{x} - l_{x}p_{y})}{l_{y} - p_{y}} \qquad = \frac{(l_{z}p_{y} + l_{y}p_{z})}{(l_{y} - p_{y})}$$

$$\begin{bmatrix} s_x \\ 0 \\ s_z \\ 1 \end{bmatrix} = \begin{bmatrix} l_y & -l_x & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -l_z & l_y & 0 \\ 0 & -1 & 0 & l_y \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix}$$

$$s_x = I_y p_x - I_x p_y + 0 p_z$$
  

$$s_y = 0$$
  

$$s_z = 0 p_x - I_z p_y + I_y p_z$$
  

$$w = 0 p_x - p_y + 0 p_z + I_y$$

### **Projection Shadow**



### **Projection Shadow**

$$\begin{split} \mathbf{s} &= l + \frac{n \cdot l + d}{n \cdot l - n \cdot p} (p - l) \qquad s_x = l_x + \frac{n \cdot l + d}{n \cdot l - n \cdot p} (p_x - l_x) \\ s_x &= l_x + \frac{n \cdot l + d}{n \cdot l - n \cdot p} (p_x - l_x) \qquad = \frac{l_x (n \cdot l - n \cdot p) + (n \cdot l + d) p_x - (n \cdot l + d) l_x}{n \cdot l - n \cdot p} \\ s_y &= l_y + \frac{n \cdot l + d}{n \cdot l - n \cdot p} (p_y - l_y) \qquad = \frac{l_x n \cdot l - l_x n_x p_x - l_x n_y p_y - l_x n_z p_z + (n \cdot l + d) p_x - l_x n \cdot l - l_x d}{-n_x p_x - n_y p_y - n_z p_z + n \cdot l} \\ s_z &= l_z + \frac{n \cdot l + d}{n \cdot l - n \cdot p} (p_z - l_z) \qquad = \frac{(n \cdot l + d - l_x n_x) p_x - l_x n_y p_y - l_x n_z p_z - l_x d}{-n_x p_x - n_y p_y - n_z p_z + n \cdot l} \end{split}$$

.....

$$\begin{bmatrix} s_{x} \\ s_{y} \\ s_{z} \\ 1 \end{bmatrix} = \begin{bmatrix} n \cdot l + d - l_{x}n_{x} & -l_{x}n_{y} & -l_{x}n_{z} & -l_{x}d \\ -l_{y}n_{x} & n \cdot l + d - l_{y}n_{y} & -l_{y}n_{z} & -l_{y}d \\ -l_{z}n_{x} & -l_{z}n_{y} & n \cdot l + d - l_{z}n_{z} & -l_{z}d \\ -n_{x} & -n_{y} & -n_{z} & n \cdot l \end{bmatrix} \begin{bmatrix} p_{x} \\ p_{y} \\ p_{y} \\ p_{z} \\ 1 \end{bmatrix}$$

### **Projection Shadow Matrix**

$$s_x$$
 $s_y$ 
 $=$ 
 $n \cdot l - l_x n_x$ 
 $-l_x n_y$ 
 $-l_x n_z$ 
 $-l_x n_w$ 
 $p_x$ 
 $-l_y n_x$ 
 $n \cdot l - l_y n_y$ 
 $-l_y n_z$ 
 $-l_y n_w$ 
 $p_y$ 
 $-l_z n_x$ 
 $-l_z n_y$ 
 $n \cdot l - l_z n_z$ 
 $-l_z n_w$ 
 $p_z$ 
 $-l_w n_x$ 
 $-l_w n_y$ 
 $-l_w n_z$ 
 $n \cdot l - l_w n_w$ 
 $1$ 
 $=$ 
 $\Theta$ 
 $\Theta$ 
 $\Theta$ 
 $=$ 
 $[n_x, n_y, n_z, n_w]$ 
 $\Theta$ 
 $\Theta$ 
 $\Theta$ 
 $=$ 
 $[n_x, n_y, n_z, n_w]$ 
 $\Theta$ 
 $\Theta$ 
 $\Theta$ 
 $H$ 
 $=$ 
 $\Theta$ 
 $\Theta$ 
 $\Theta$ 
 $H$ 
 $=$ 
 $\Theta$ 
 $\Theta$ 
 $\Theta$ 
 $H$ 
 $=$ 
 $[n_y, n_z, n_w]$ 
 $[n_y = 0, l = ]$ 
 $[n_y = 1]$ 

### **Shadow**



Render without shadow

Render with shadow

### Reflection



http://www.gamasutra.com/features/19990723/opengl\_texture\_objects\_02.htm

### **Planar Reflection**

■ How to calculate the reflection point,  $q' = (x_0', y_0', z_0')$  of the point,  $q = (x_0, y_0, z_0)$  for the mirror plane (n, d)

$$q' = q - 2kn$$

$$= q - 2\frac{ax_0 + by_0 + cz_0 + d}{\sqrt{a^2 + b^2 + c^2}}n$$

$$= q - 2(n \cdot q + d)n \text{ when n is unit vector}$$

$$q' = Rq$$

$$q' = Rq$$

$$R = \begin{bmatrix} 1 - 2a^2 & -2ab & -2ac & -2ad \\ -2ab & 1 - 2b^2 & -2bc & -2bd \\ -2ac & -2bc & 1 - 2c^2 & -2cd \\ 0 & 0 & 0 \end{bmatrix}$$
Plane n· p + d = 0
Plan

### **Planar Reflection**

$$\begin{bmatrix} x_{0}'\\ y_{0}'\\ z_{0}' \end{bmatrix} = \begin{bmatrix} x_{0}\\ y_{0}\\ z_{0} \end{bmatrix} - 2(ax_{0} + by_{0} + cz_{0} + d) \begin{bmatrix} a\\ b\\ c \end{bmatrix} \text{ when } n \text{ is unit vector } (a^{2} + b^{2} + c^{2} = 1)$$

$$x_{0}' = x_{0} - 2(ax_{0} + by_{0} + cz_{0} + d)a$$

$$= (1 - 2a^{2})x_{0} - 2aby_{0} - 2acz_{0} - 2ad$$

$$y_{0}' = y_{0} - 2(ax_{0} + by_{0} + cz_{0} + d)b$$

$$= 2abx_{0} + (1 - 2b^{2})y_{0} - 2bcz_{0} - 2bd$$

$$z_{0}' = z_{0} - 2(ax_{0} + by_{0} + cz_{0} + d)c$$

$$= -2acx_{0} - 2bcy_{0} + (1 - 2c^{2})z_{0} - 2cd$$

$$R = \begin{bmatrix} 1 - 2a^{2} & -2ab & -2ac & -2ad \\ -2ab & 1 - 2b^{2} & -2bc & -2bd \\ -2ac & -2bc & 1 - 2c^{2} & -2cd \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

### **Planar Reflection**

Reflection transformation matrix for the plane (yz-, xz-, xy-plane)

