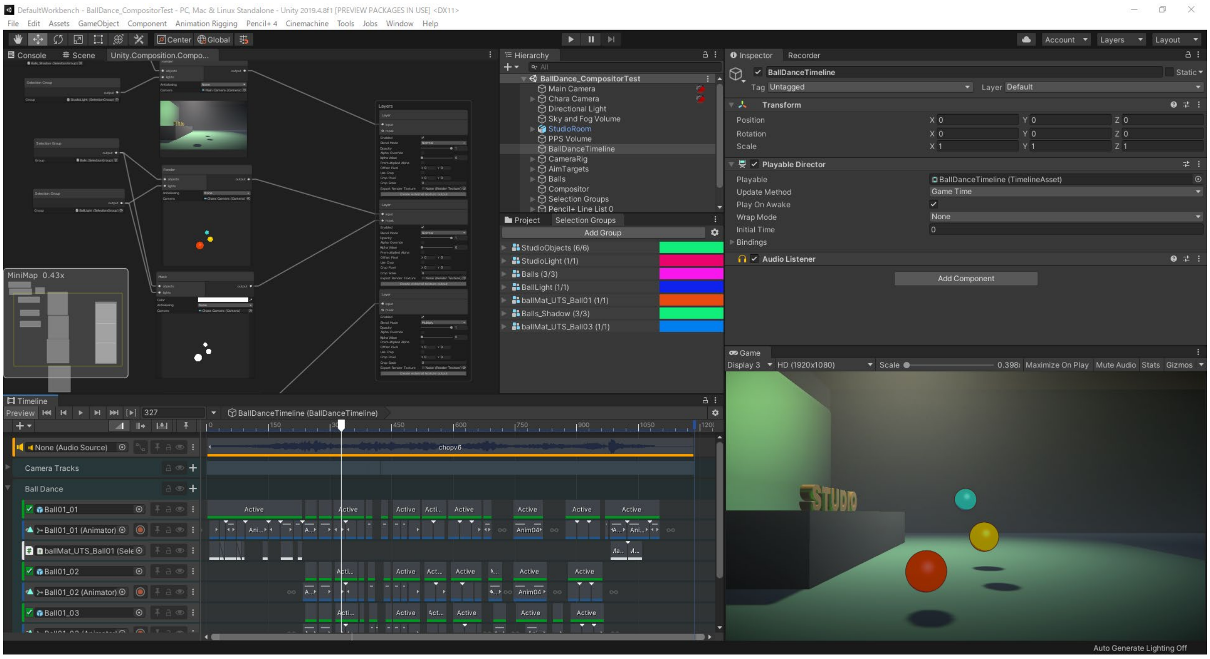
# Ball Dance/Visual Compositor Setup

2020/09/20 N.Kobayashi/ UTJ

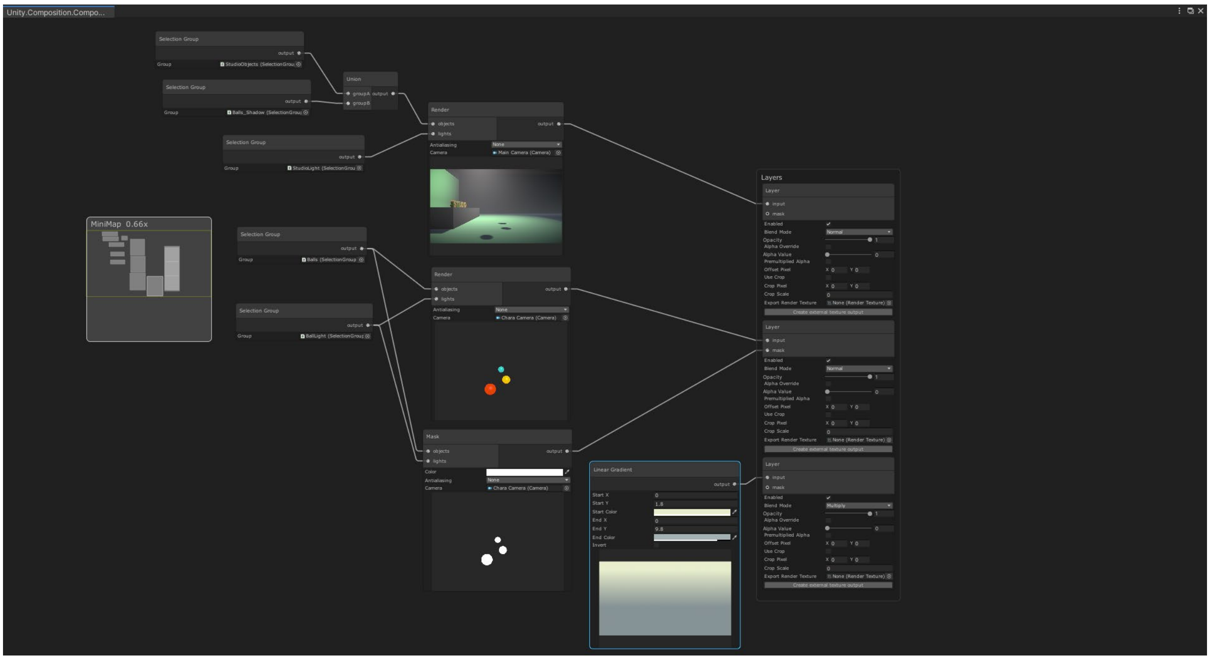
2022/08/10 Updated for BallDanceTemplate-0.13.1-exp. Project



# Introduction

- This document describes how to set up BallDance_CompositorTest included in the BallDanceTemplate-0.xx.x-exp project. BallDance is its original scene that does not include a compositing process with Visual Compositor.
- This document explains how to use Visual Compositor, which is mainly used for "compositing and editing" in the process of producing cell look 3DCG animation.
- Before starting, it is assumed that a scene used for “compositing and editing” work has been completed. Please see "Ball Dance/PreViz Setup" for details.
- A paid license for Pencil+ 4 Line for Unity is required to display all the features of this project properly.
  **In the case that your license is not valid, a watermark will appear in the game view when enabling real-time updates for Pencil+ 4 Line for Unity.**
  If you want to display only rendering results of Pencil+ Line Cache, no license is required.

# What is Visual Compositor

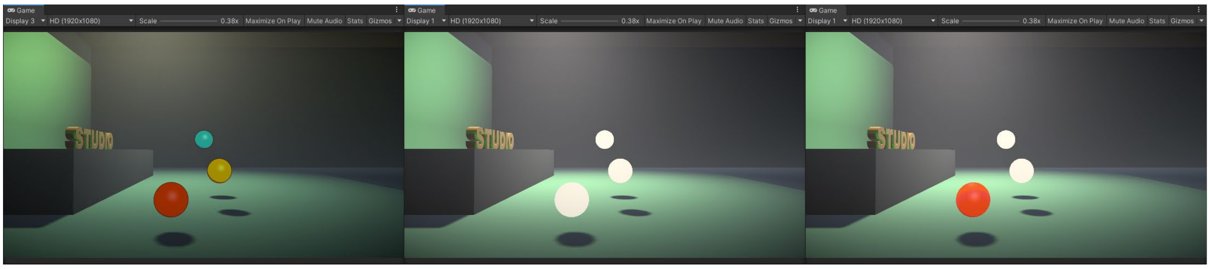


- Visual Compositor is a new feature added to Unity Anime Toolbox. Primarily, it provides 2D-image layer structure and editing functions for each layer without destroying 3D scene settings in Unity. By using Visual Compositor, animation creators do not need to render a composite information of 3D model into output buffer images and then composite them with After Effects each time. As a result, you can check the completed model of 2D image while simultaneously modifying the 3D scene settings in real time.
- Visual Compositor supports not only traditional Unity built-in rendering pipelines, but also the latest rendering pipeline, HDRP and URP. This document describes an example with HDRP.
- Images created with Visual Compositor can be output in formats such as serial number images and Apple ProRes with Unity Recorder. Visual Compositor outputs not only the nicely completed image but also images just processed by Pencil+ Line and various masking images. Those images processed by Visual Compositor can be further edited and processed with existing tools.

# Advantage of Using Visual Compositor

The following is the advantage to perform "compositing and editing" work with Visual Compositor in the process of making cell look 3DCG animations:

1. You can separate light and camera settings for each render target. Traditionally, light linking and camera culling were barely set with only 32 layers in Unity and 8 light layers provided in HDRP. Now, you are released from the tough setting situation.
2. This is especially effective for the case that you want to change the light intensity significantly for each material of an object to be composited, such as in an HDRP environment. In other words, you can freely combine a background object, which you want to shoot in a lighting environment with realistic brightness (very high intensity) that is close to the real world, and character objects, which you want to shoot in a lighting environment good for mat solid colors (within a range of traditional intensity). Color management becomes easier especially for toon shading and cell shading.
3. By compositing rendering results with the layer function of Visual Compositor while adding a mask to each of them in real time, you can easily design 2D-like composite expression that was difficult with traditional 3D models. You can also handle opacity without restrictions.
4. Even when the work results are passed to a post process as serial number images, most of the picture composition can be completed in Visual Compositor. As a result, the number and types of materials which are passed to external editing applications such as After Effects, will be drastically reduced.



The image on the far left is a screen using Visual Compositor. The cell look balls and the background fit into each other. UTS/HDRP is used as their cell shader.

The image in the middle is the result of illuminating the same scene with HDRP light which renders the background. SDR color design of the cell shader often causes overexposure under HDR lighting environment, which HDRP is good at.

The image on the far right has the same environment as the one in the middle, but only the red ball in the foreground shows its color. This is because UTS/HDRP has a feature called Scene Lights Hi-Cut Filter, and it is activated only for the red ball material. Although you can handle HDR environment with special shaders such as UTS/HDRP, it is not a problem anymore to change an intensity of each light which illuminates objects if you use Visual Compositor in the 1st place. And therefore, you can manage colors on object basis as used to be.

# 1st Setup with HDRP

Before using Visual Compositor with HDRP, we should do some preparations for the project. The difference from the built-in pipeline is also briefly discussed.

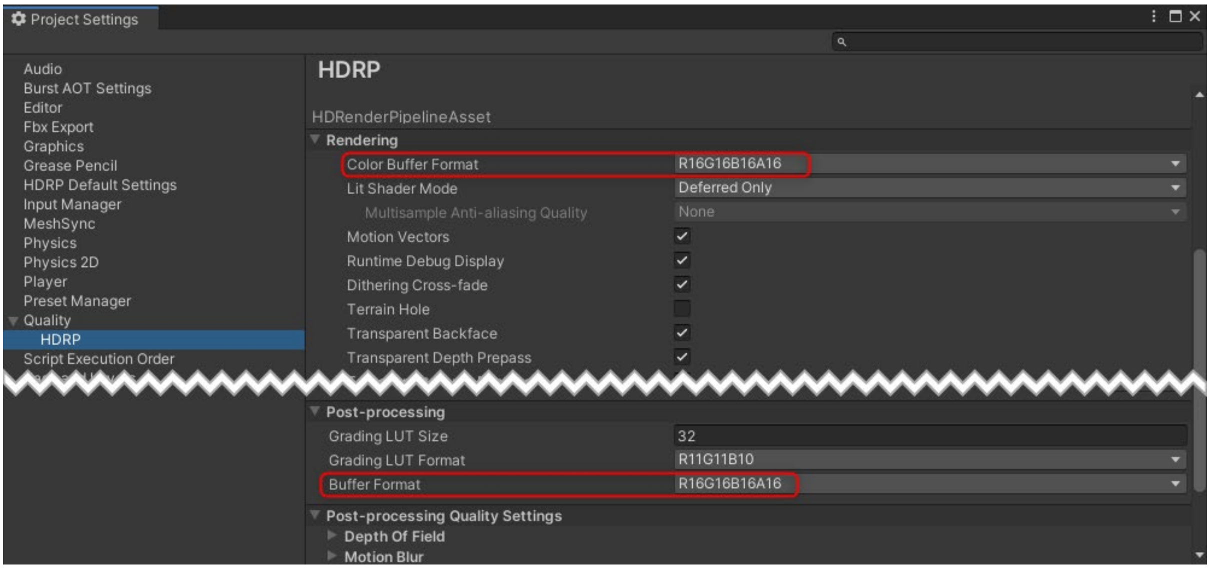
## Settings Project Settings

From the main menu, go to Edit > Project Settings.
Open Project Settings window and set up the following.

### Quality > HDRP

Set HDRenderingPipelineAsset as follows:

- **Rendering > Color Buffer Format**
  **Set it to "R16G16B16A16" (requried)**
- **Post-prosessing > Buffer Format**
  **Set it to "R16G16B16A16" (required for using Pencil+ 4 Line for Unity)**



HDRP

HDRenderPipelineAsset

Rendering
Color Buffer Format: R16G16B16A16
Lit Shader Mode: Deferred Only
Multisample Anti-aliasing Quality: None
Motion Vectors: ☑
Runtime Debug Display: ☑
Dithering Cross-fade: ☑
Terrain Hole: □
Transparent Backface: ☑
Transparent Depth Prepass: ☑

Post-processing
Grading LUT Size: 32
Grading LUT Format: R11G11B10
Buffer Format: R16G16B16A16

Post-processing Quality Settings
Depth Of Field
Motion Blur

### Settings of HDRP Default

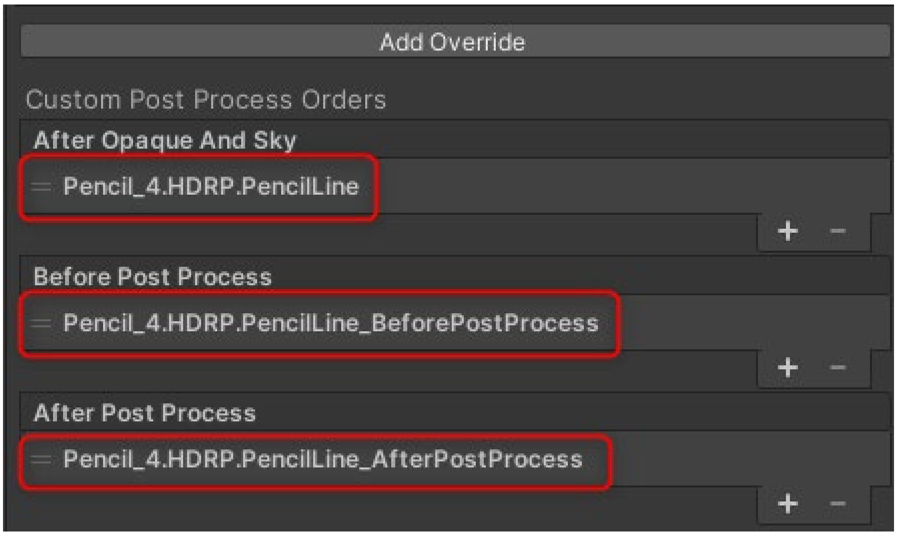
Various settings common to HDRP are decided in HDRP Default Setting. All you need to adjust here is the basic camera settings, the basic settings of post process, and, if using the HDRP version of Pencil+ Line for Unity, an addition of modules to the post processes.

**Unlike the built-in pipeline, in HDRP, post processes are natively set on a camera in the scene. You can define their default settings here. If you have problems with setting up those, you can adjust them as follows.**

- **Frame Settings > Camera > Lighting**
  **Checkbox "Fog" (affecting the behavior of mask node**)

    - **On** ⇒ Fog is enabled on each camera.
      For a masking camera, uncheck “Fog” checkbox individually in Custom Frame Settings of each camera.
    - **Off** ⇒ Fog is disabled on all cameras.
      Fog is disabled even on masking cameras. (recommended)

- **Custom Post Process Orders**
  Add the module of Pencil+ 4 Line for Unity, HDRP version.



Custom Post Process Orders

After Opaque And Sky

Pencil_4.HDRP.PencilLine

Before Post Process

Pencil_4.HDRP.PencilLine_BeforePostProcess

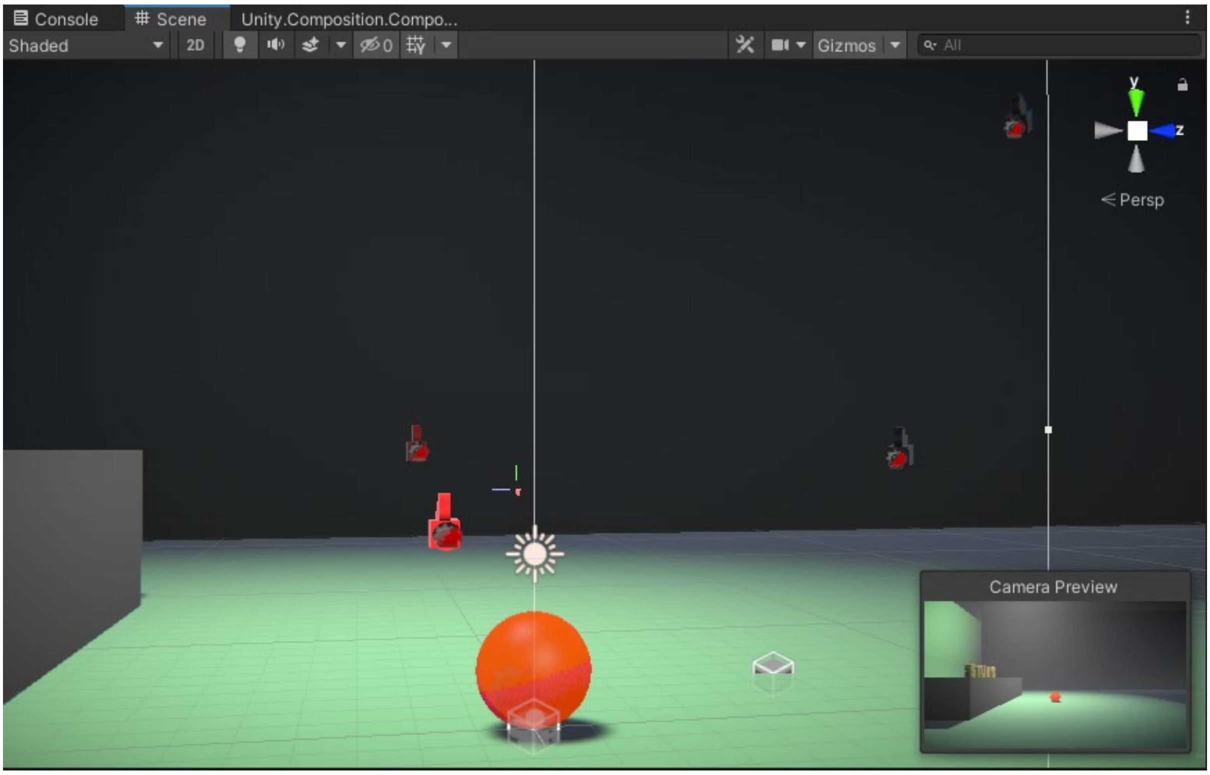
After Post Process

Pencil_4.HDRP.PencilLine_AfterPostProcess

For details on how to use Pencil+ 4 Line for Unity, please see its help file.

## Settings Culling Masks (option)

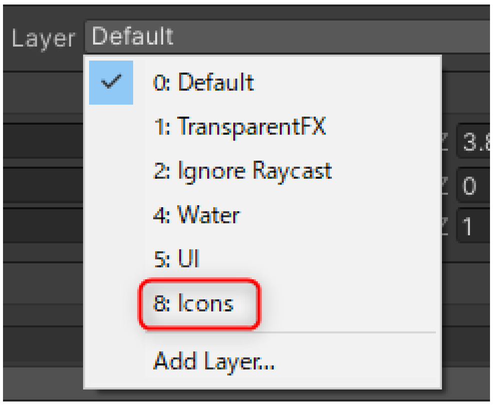
Marking various cameras in a camera rig with a camera-like icon, you can get the camera layout in the scene at a glance. It is convenient. But these camera icons should not be displayed in the game view (it is not good that they are reflected during rendering).



In such a case, there is a way to prevent a specific object from being displayed on cameras in the game view. That is a function called “culling mask” that can be set for each camera component.
You can set a culling mask as described below.

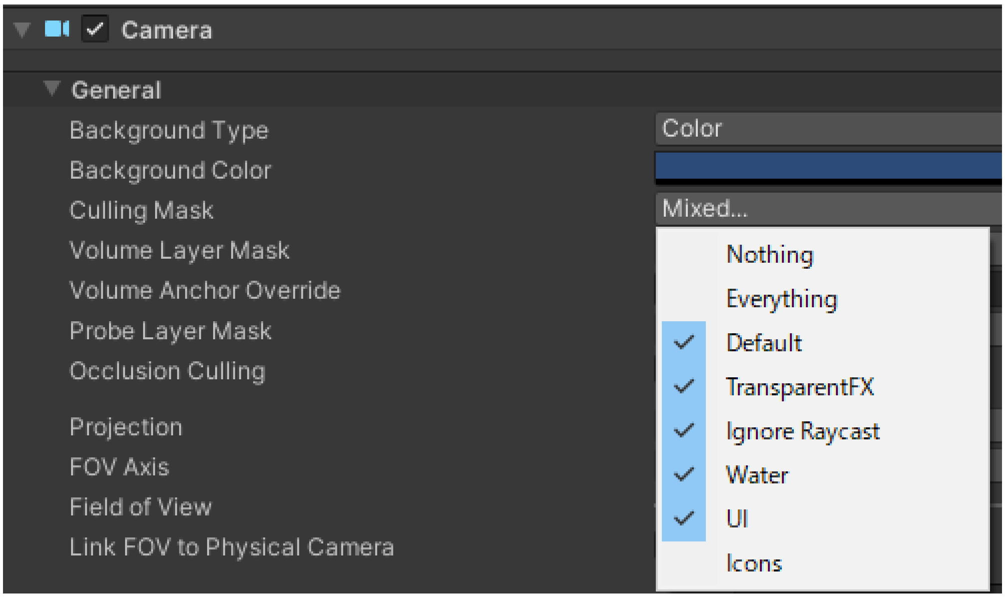
### Set a Layer

From Layer dropdown at the top of Inspector window, press Add Layer button and enter a new layer name to an empty User Layer slot. In this example, we name it "Icons."



## Set a Culling Mask (Camera > Culling Mask)

Open Culling Mask dropdown of the Camera and check Icons.
The screen will look like the one below.



Camera

General

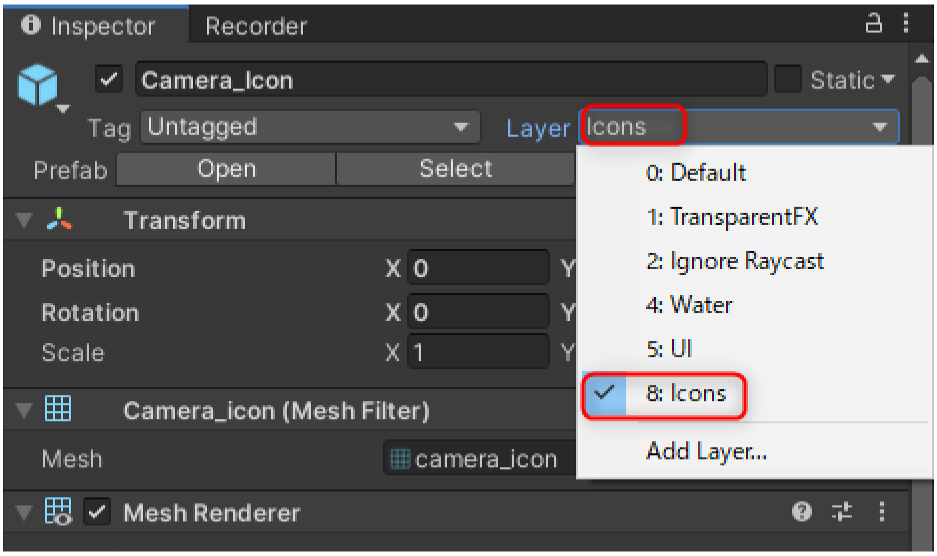
Background Type: Color

Background Color

Culling Mask: Mixed...

## Set the Layer of the object that you want to display only in the scene view, to Icons

With this setting, Camera_Icon is no longer shown in the game view.



Camera_Icon

Tag: Untagged

Layer: Icons

Position: X 0

Rotation: X 0

Scale: X 1

Camera_icon (Mesh Filter)

Mesh: camera_icon

Mesh Renderer

# Before Starting Visual Compositor

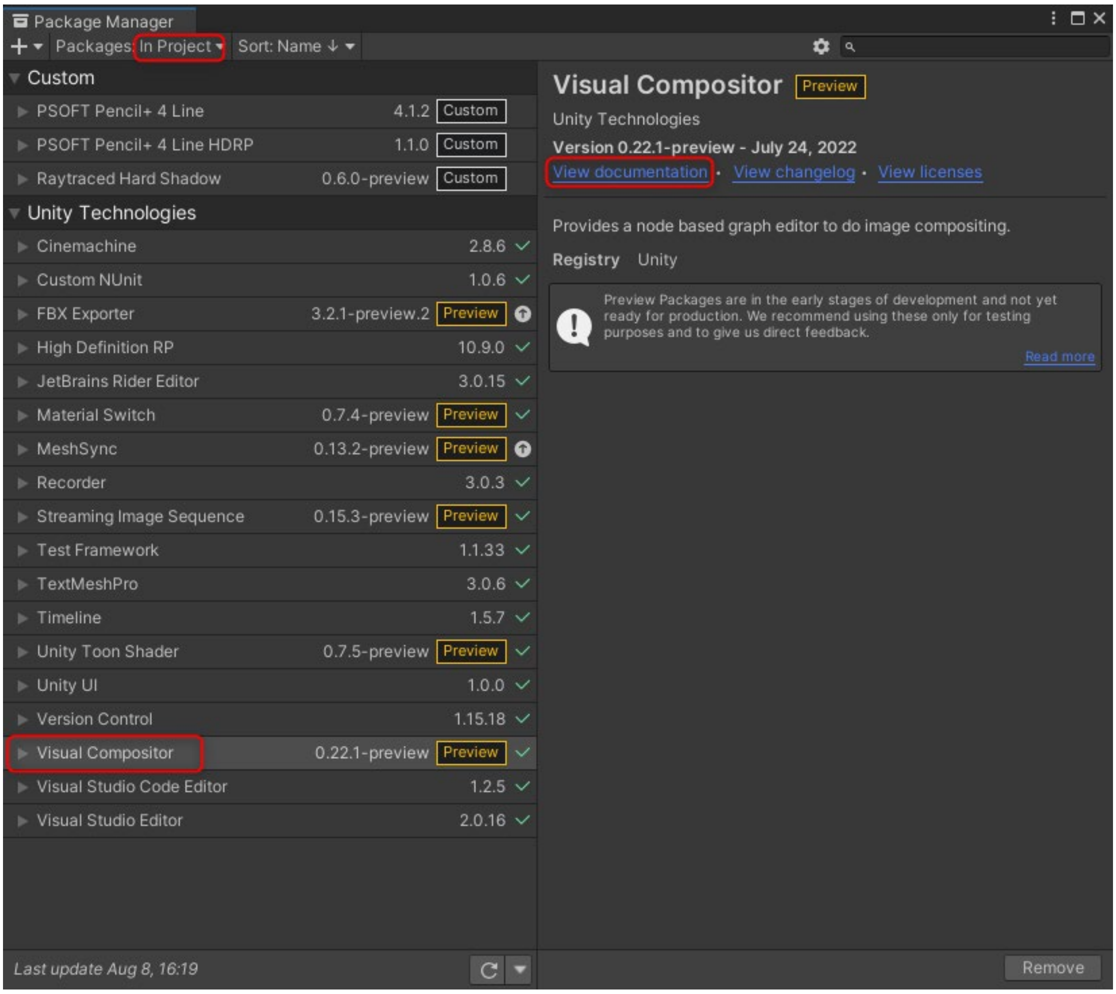
Once you finished the above preparation, we will set up Visual Compositor.

## Seeing references

You can find the document that explains the detailed functions of Visual Compositor as follows:

1. From the menu bar, go to Window > Package Manager.
2. In Package Manager window, confirm that the In Project drop down is selected.
3. All the packages that have been imported into the current project are shown. Search for "Visual Compositor" among them.

When you select Visual Compositor, there will be a section called "Links" on the left side of the window. Clicking “View Documentation” will open the document. It is recommended to read it.



Custom

| | | |
|---|---|---|
| PSOFT Pencil+ 4 Line | 4.1.2 | Custom |
| PSOFT Pencil+ 4 Line HDRP | 1.1.0 | Custom |
| Raytraced Hard Shadow | 0.6.0-preview | Custom |

Unity Technologies

| | | |
|---|---|---|
| Cinemachine | 2.8.6 | ✓ |
| Custom NUnit | 1.0.6 | ✓ |
| FBX Exporter | 3.2.1-preview.2 | Preview |
| High Definition RP | 10.9.0 | ✓ |
| JetBrains Rider Editor | 3.0.15 | ✓ |
| Material Switch | 0.7.4-preview | Preview ✓ |
| MeshSync | 0.13.2-preview | Preview |
| Recorder | 3.0.3 | ✓ |
| Streaming Image Sequence | 0.15.3-preview | Preview ✓ |
| Test Framework | 1.1.33 | ✓ |
| TextMeshPro | 3.0.6 | ✓ |
| Timeline | 1.5.7 | ✓ |
| Unity Toon Shader | 0.7.5-preview | Preview ✓ |
| Unity UI | 1.0.0 | ✓ |
| Version Control | 1.15.18 | ✓ |
| Visual Compositor | 0.22.1-preview | Preview ✓ |
| Visual Studio Code Editor | 1.2.5 | ✓ |
| Visual Studio Editor | 2.0.16 | ✓ |

Visual Compositor Preview

Unity Technologies

Version 0.22.1-preview - July 24, 2022

View documentation • View changelog • View licenses

Provides a node based graph editor to do image compositing.

Registry Unity

Preview Packages are in the early stages of development and not yet ready for production. We recommend using these only for testing purposes and to give us direct feedback.

Read more

Last update Aug 8, 16:19

In the following, we will explain how to start working with Visual Compositor, focusing on functions needed in this project.

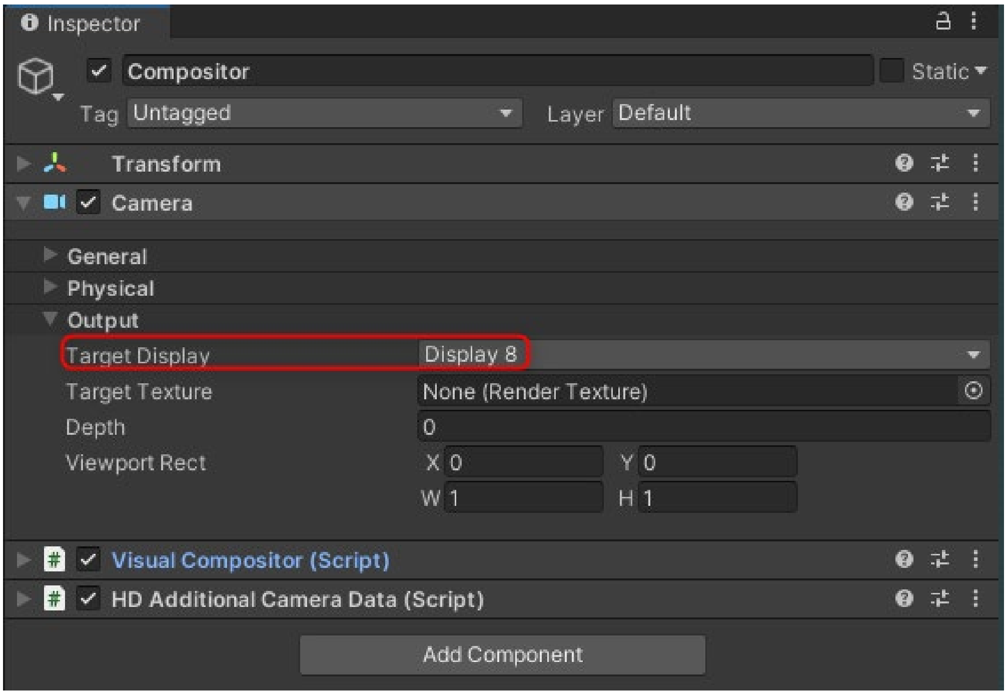
# Creating a Visual Compositor Game Object in Scene

To use Visual Compositor in the scene, create a Compositor game object in the scene. The steps are as follows:

1. In Hierarchy window, click "+" button to create an empty game object (Create Empty)
2. Rename the created empty game object "GameObject" to "Compositor"
3. In Inspector window of the Compositor game object, click Add Component button to attach Visual Compositor Component (type “Compo” in the search field to find it)
4. Make sure the following three components have been added to the Compositor game object
   - Camera
   - Visual Compositor
   - HD Additional Camera Data（HDRP Only）

Now you can use Visual Compositor.

*Note: About the settings to check composition results of Visual Compositor (It will be explained in detail later in this document.) Find the Camera component attached to the Visual Compositor game object in Inspector window. Set “Target Display” under Output to “Display 8.” After this setting, you can check composition results of Visual Compositor by selecting “Display 8” of the game view display.



Inspector

☑ Compositor □ Static

Tag Untagged Layer Default

Transform

☑ Camera

General

Physical

Output

Target Display: Display 8

Target Texture: None (Render Texture)

Depth: 0

Viewport Rect: X 0 Y 0 W 1 H 1

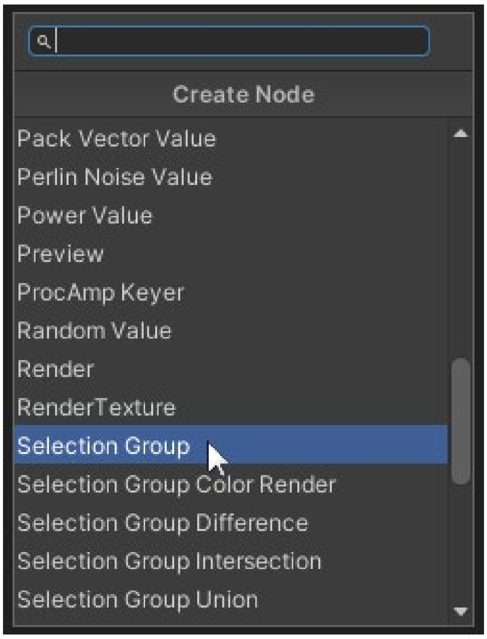
☑ Visual Compositor (Script)

☑ HD Additional Camera Data (Script)

## Opening Visual Compositor Window

Open Visual Compositor window where you actually work on the Visual Compositor as follows:

1. From the main menu, go to Window > Rendering > Visual Compositor, and open Visual Compositor window
2. Dock the Visual Compositor window to a convenient position in Unity
3. In Visual Compositor window, you can perform various compositing and editing works on a node basis. Nodes can be called in the following 2 ways:
    - **Right-click in Visual Compositor window** and select "Create Node"
    - Place the mouse cursor in Visual Compositor window and **press the space bar on the keyboard**



Next, we will explain Selection Groups, which select various elements in the scene and make them available in Visual Compositor window.

# Selecting Various Elements in the Scene by Selection Groups

## What is Selection Groups

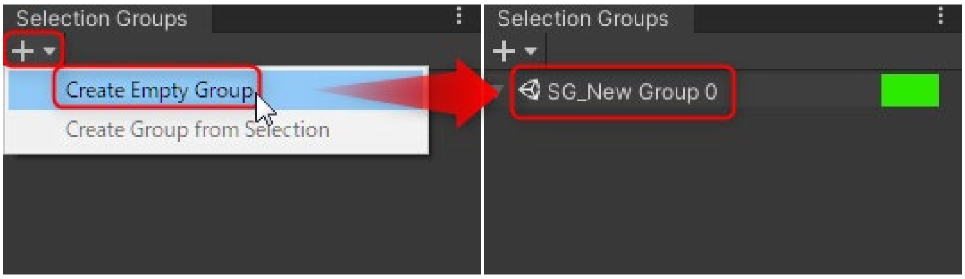
- When compositing and editing in Visual Compositor window, you need to decide “what kind of effect” you want to add to “which objects.” The function to select these "objects" from many game objects existing in the scene and gather them as a group is a function called “Selection Groups.” It was newly added to Unity AnimeToolbox. This function is equivalent to "Selection Set" function of Maya. The usage of Selection Group is also very similar to Selection Set.

- A Selection Group functions with three elements: "Selection Grpups window" which is the use interface of Selection Groups, "Selection Groups game object" which is displayed as a game object in Hierarchy window, and finally "Selection Groups asset" which stores them as assets in Project folder. Selection Groups are saved on project basis, the same Selection Groups are referenced from all the scenes that belong to the same project (note that this is different from Maya's Selection Set).
- Selection Groups operation is executed from Selection Groups window. You can add/delete game objects to/from a group in this window. For more information, see the document of Selection Groups from Package Manager window.

## Setting New Selection Group

To set a new Selection Group, follow the steps below:

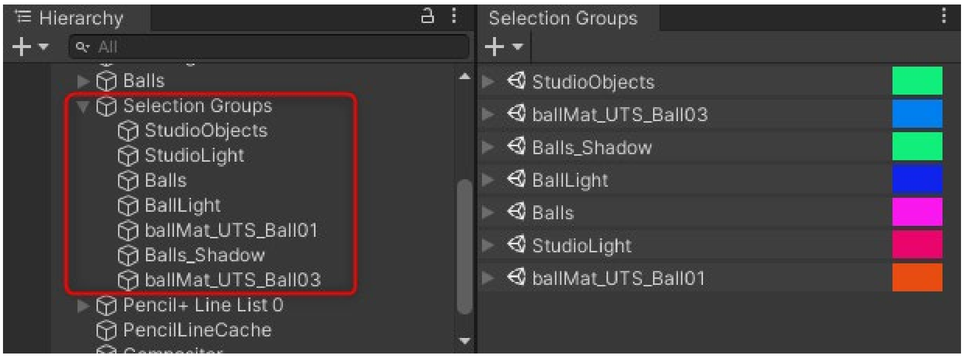
1. From the main menu, go to Window > General > Selection Groups, and open Selection Groups window
2. Once Selection Groups window opens, dock it in a convenient position in Unity. If no group has been created in the window, press “+” dropdown + ▾ to select “Create Empty Group”.



If Selection Groups are already defined in other scenes in the project, the Selection Groups window will open with those groups. Game objects included in each group are queried on a scene basis. And therefore if there are corresponding objects in the scene, they will be registered.

**Tips: Dock Selection Groups window next to Hierarchy window as shown above. It will improve usability since Selection Groups are often used with Hierarchy window.**

3. In this sample scene, you can see a hierarchy of game objects called "Selection Groups" in Hierarchy window. When you expand it, you can see that each element of Selection Groups is registered and referenced from each game object in the scene.



As shown above, in order to refer to each element of Selection Groups from an object in the scene, it is necessary to expose Selection Groups in Hierarchy window.

# How to Set Each Selection Group

## How to register game objects to each Selection Group

There are several ways to register game object(s) to a Selection Group. We will introduce some easy ways.

- **In Hierarchy window, select multiple game objects that you want to gather in a new group, press Add Group button, and put them together into a New Group**
  ⇒ This is the easiest way. However, be careful that if you modify a referenced game object, the reference may be broken. Especially be careful for renaming. Also, with this direct selection method, even game objects that keep the same name across scenes will not be extracted when the scene changes.

- **Drag and drop game object(s) selected in Hierarchy window to an existing group in Selection Groups window.**
  ⇒ This is quite easy, too. Be careful for the same points mentioned above.

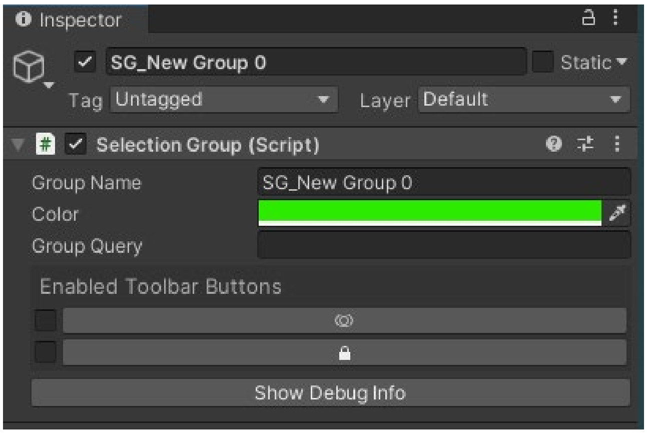
- **Extract corresponding game objects from the scene using a query expression (recommended)**
  ⇒ It is necessary to write a query expression, but this approach is recommended since it is versatile and groups reusable in other scenes as long as the conditions are met.

## How to Write Query Expression

The query expression used in Selection Groups is called “GoQL.” It is unique, but has a very simple format. For details, select the GoQL from the Package Manager window and see the document from Links.

To write the query expression, follow the steps below.

1. In Selection Groups window, click a group to which you want to add a query expression and show the properties of it in Inspector window.



Inspector

☑ SG_New Group 0 □ Static

Tag Untagged Layer Default

☑ Selection Group (Script)

Group Name SG_New Group 0

Color 🟩

Group Query

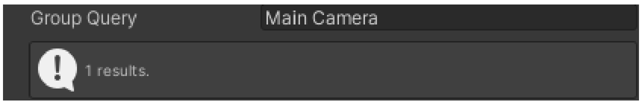
Enabled Toolbar Buttons

□

□

2. Write a query expression in **Group Query** field in the Inspector window.
   Example:

a. To extract the Main Camera in the scene

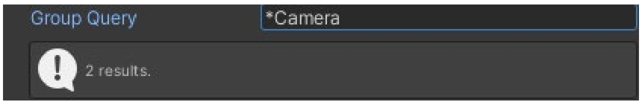


Group Query: Main Camera

1 results.

⇒ Enter the Main Camera's name

b. To extract all game objects in the scene that have a string "Camera" in the latter part of the name (such as "Main Camera" and "Chara Camera")

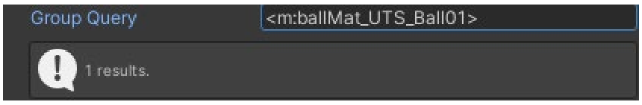


Group Query: *Camera

2 results.

⇒ Represent names with wildcard

c. To extract game objects to which a particular material is attached

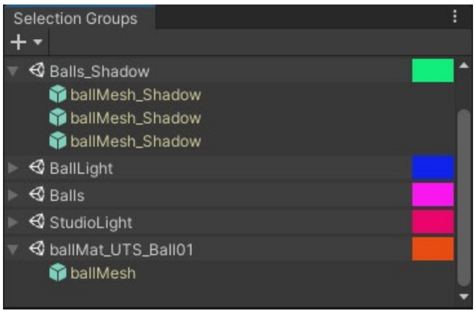


Group Query: <m:ballMat_UTS_Ball01>

1 results.

⇒ Specify the type of the material. Mesh objects attached to the particular material are extracted.

3. The result of the query expression will be shown in Selection Groups window. Confirm the result.

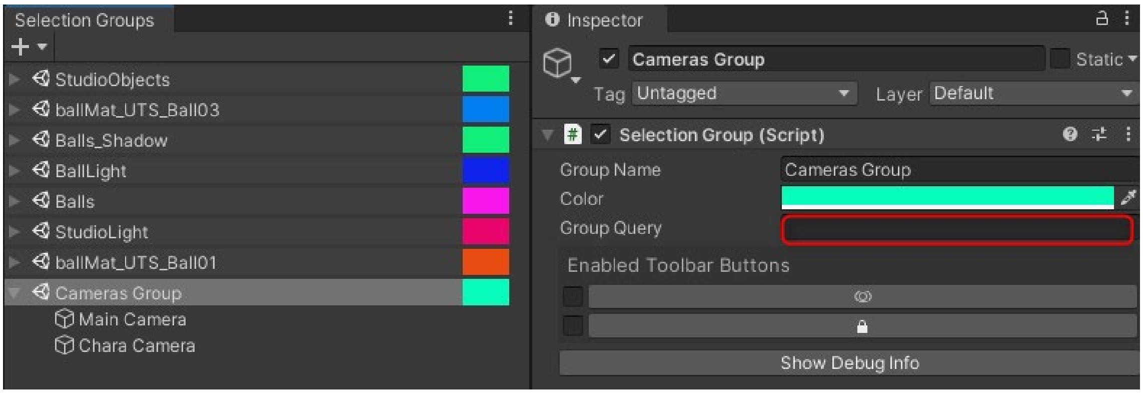


Selection Groups

- Balls_Shadow
  - ballMesh_Shadow
  - ballMesh_Shadow
  - ballMesh_Shadow
- BallLight
- Balls
- StudioLight
- ballMat_UTS_Ball01
  - ballMesh

## How to Delete Registered Game Objects from Selection Group

1. Select a Selection Group to be changed in Selection Groups window and delete the query expression from Group Query field in Inspector window.



Selection Groups

- StudioObjects
- ballMat_UTS_Ball03
- Balls_Shadow
- BallLight
- Balls
- StudioLight
- ballMat_UTS_Ball01
- Cameras Group
  - Main Camera
  - Chara Camera

Inspector

☑ Cameras Group □ Static

Tag: Untagged Layer: Default

☑ Selection Group (Script)

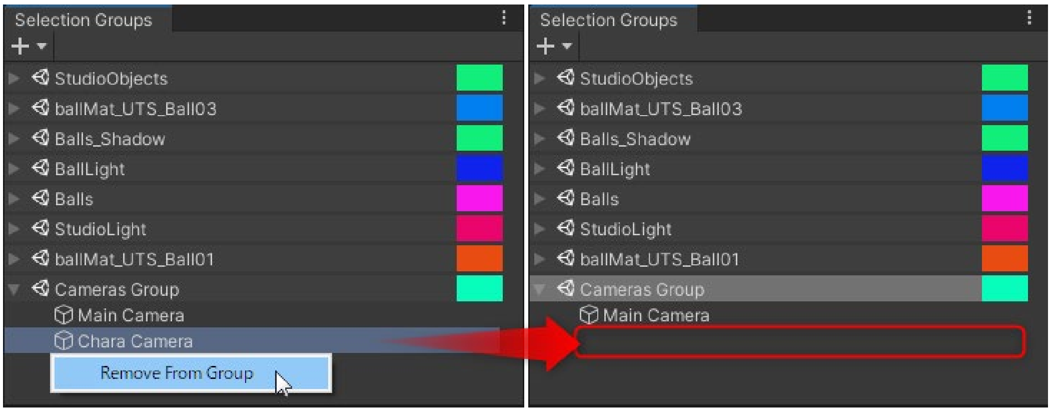
Group Name: Cameras Group

Color

Group Query:

Enabled Toolbar Buttons

2. Select an element you want to delete and right click on it. Select "**Remove From Group**" to delete the element from the Selection Group.



**Tips: After deleting the selected game object, it becomes the same as directly selecting the remaining game objects.**

# Performing Picture Composition with Visual Compositor

**Visual Compositor is a node-based application for composition.** Connect various nodes to which different functions are assigned with a line, and complete the picture as following the flow. Each node has only one function. Because of it, even when overlaying simple layers, you need to use four nodes such as:

- One function node (e.g., Render node)
- Another function node (e.g., another Render node)
- Two Layer nodes (as an output port of blended picture.)

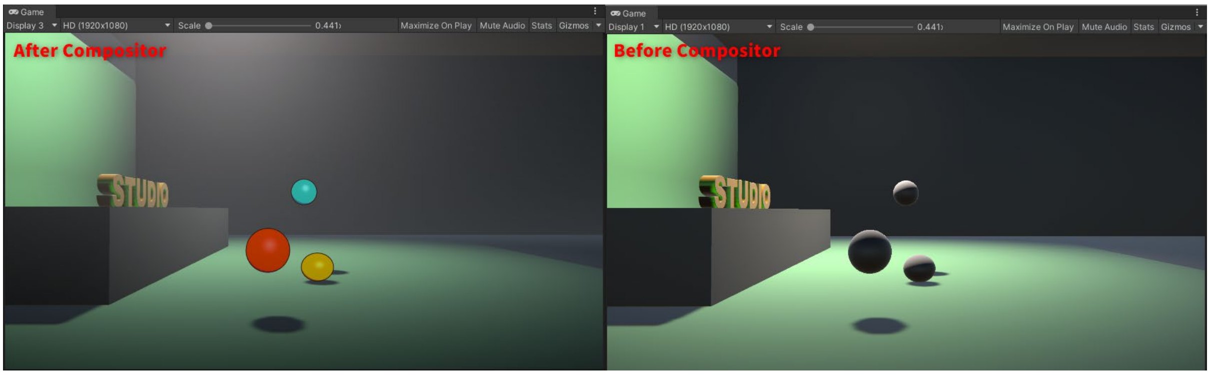
Unlike layer-based, it may seem like a lot of work at 1st. However, as you get used to it, you will realize the combination of nodes facilitates complicated compositing work, because it is retrospectively readable and easy to understand what you did and shows your flow of thoughts. Such a combination of nodes is called "flow" representing a flow toward the goal.

Next, “picture composition” with Visual Compositor will be described.

## Composition of Layer Node Stack

First, let's look at a screen before and after using Visual Compositor in the same scene.

In the image below, the left is the scene after using Visual Compositor, and the right is the scene before using it.



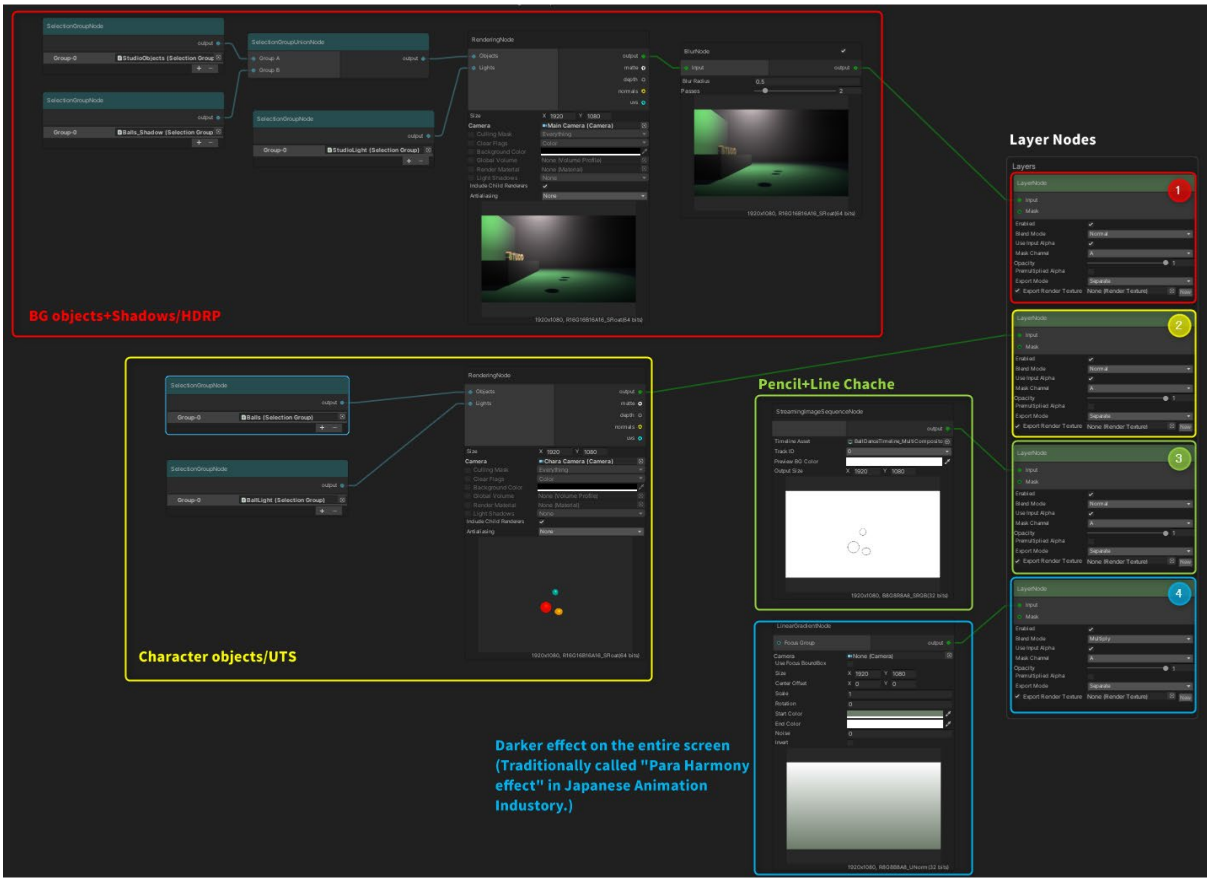
To make the above scene, the following was performed with Visual Compositor:

1. Background (BG) elements: rendering background objects and creating shadows using volumetric lights for HDRP.
2. Cell look character elements: rendering balls with UTS/HDRP material illuminated by the directional light.
3. Toon line composition: compositing with Pencil+ Line Cache and placing a toon line on the ball.
4. Darker effect on the entire screen (traditionally called the "Para(-ffin paper overlaid) Harmony effect" in the Japanese animation industry): using the effect over the entire screen like 2D to make the whole screen look natural.

The final screen is made by overlaying these layers in the order of 1 ⇒ 2 ⇒ 3 ⇒ 4.

Each layer generates an image in real time. And therefore, it is not necessary to output the 3D image buffer beforehand to edit it again with an external Visual Compositor like After Effects as it used to be.

If showing the above composition in the Visual Compositor window, it may look like the following image.



BG objects+Shadows/HDRP

Layer Nodes

Pencil+Line Chache

Character objects/UTS

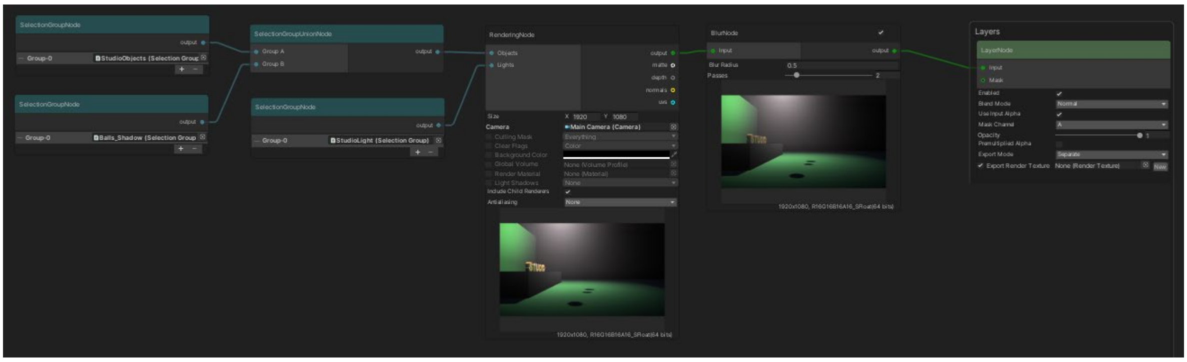
Darker effect on the entire screen (Traditionally called "Para Harmony effect" in Japanese Animation Industry.)

In the Layer Nodes stack, the final output, the bottom node is the layer that is overlaid on top of the screen. The calculation process is to add node 1 and 2 plus node 3, and then add node 4 to the result of 1-3 as if the water flows from top to bottom. Keep in mind that it is a different idea from Photoshop that decides the order of layers according to their stacking order on the Layers panel.

Next, check a node connected to each layer.

Please see the document of Visual Compositor for further information on functions of each node.

# 1st Layer: Rendering Background Objects and Shadow



Group-0 StudioObjects (Selection Group)

Group-0 Balls_Shadow (Selection Group)

Group-0 StudioLight (Selection Group)

Size X 1920 Y 1080
Camera Main Camera (Camera)
Culling Mask Everything
Clear Flags Color
Background Color
Global Volume None (Volume Profile)
Render Material None (Material)
Light Shadows None
Include Child Renderers ✓
Antialiasing None
1920x1080, R16G16B16A16_SFloat(64 bits)

Blur Radius 0.5
Passes 2
1920x1080, R16G16B16A16_SFloat(64 bits)

Enabled ✓
Blend Mode Normal
Use Input Alpha ✓
Mask Channel A
Opacity 1
Premultiplied Alpha
Export Mode Separate
✓ Export Render Texture None (Render Texture) New

Let's start with the 1st layer which is the basis.

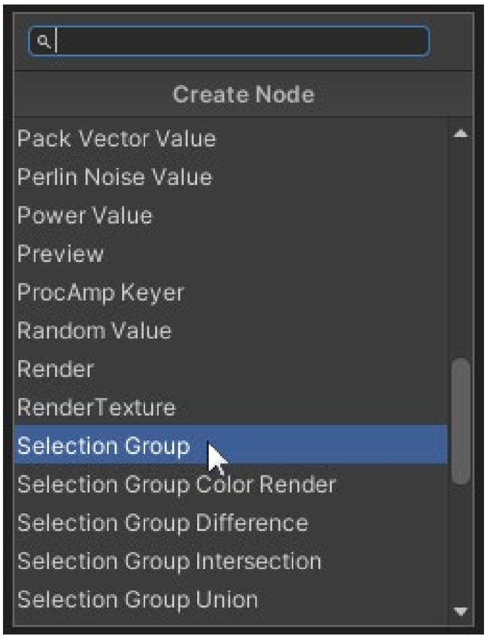
The 1st layer mainly renders background objects. It is considered that a single picture as a background is created in this layer.

The big difference from the 2nd layer, in which mainly characters are rendered, is the type of light used for rendering.

The objects on the 1st layer are illuminated by the volumetric light that underlies the entire scene. Illuminating with the volumetric light makes it possible to render a Japanese-animation-like refined background. The volumetric light also creates shadows of balls (these are characters).

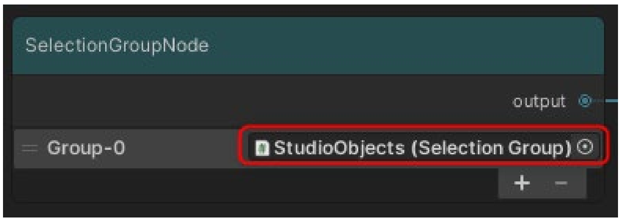
## Adding New Node

Place the mouse cursor in Visual Compositor window and press the space bar or right-click to show the context menu and open Create Node window.
You can add a new node to Visual Compositor window from the Create Node window.



## Inputting Objects in Scene to Visual Compositor - Selection Group Node

In the Create Node window, create a Selection Group node.
A Selection Group node is for referencing an element of Selection Groups window from Visual Compositor window.
In the example below, a Selection Group called “StudioObjects” is referenced.

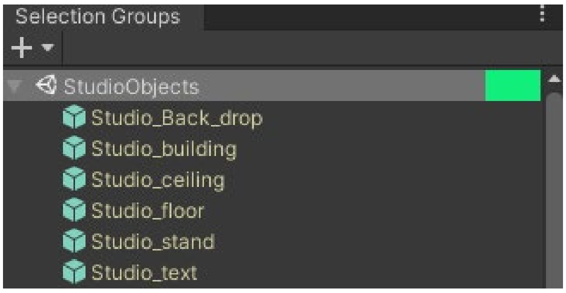


SelectionGroupNode

output

Group-0 StudioObjects (Selection Group)

You can see the contents of StudioObjects in Selection Groups window.



Selection Groups

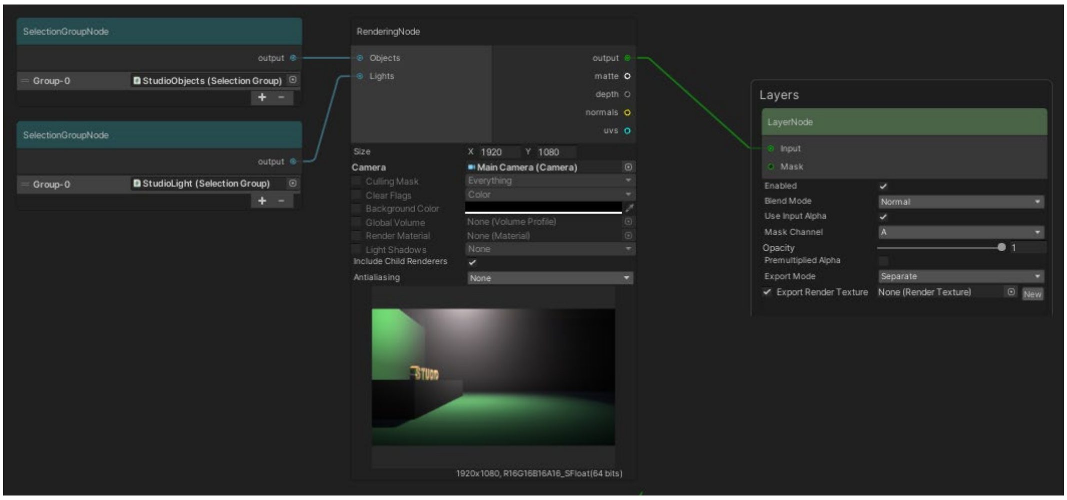
StudioObjects
Studio_Back_drop
Studio_building
Studio_ceiling
Studio_floor
Studio_stand
Studio_text

Game objects in the scene are referenced by different nodes in Visual Compositor window via Selection Group node.

**Tips: To make it easier to search by a query of a Selection Group, the following is best practice: give a name being easy to search with wildcards to objects that are likely to be handled by group, or set a common material to those objects.**

**Tips: In the case you rename an object in Unity, save the scene once and reload it. Then, the change should be reflected in Selection Groups.**

## Showing Result of Visual Compositor in Game View - Layer Node



Group-0 StudioObjects (Selection Group)

Group-0 StudioLight (Selection Group)

Size X 1920 Y 1080

Camera Main Camera (Camera)

Culling Mask Everything

Clear Flags Color

Background Color

Global Volume None (Volume Profile)

Render Material None (Material)

Light Shadows None

Include Child Renderers ✓

Antialiasing None

1920x1080, R16G16B16A16_SFloat(64 bits)

Enabled ✓

Blend Mode Normal

Use Input Alpha ✓

Mask Channel A

Opacity 1

Premultiplied Alpha □

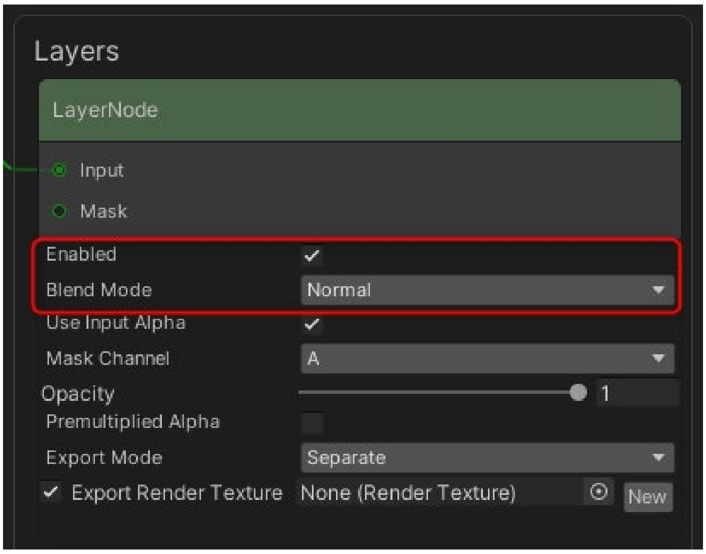
Export Mode Separate

✓ Export Render Texture None (Render Texture)

To show the result of Visual Compositor window in the game view, you eventually need to connect it to the input port of the Layer node. The minimum required node tree is composed of the Selection Group node > Render node > Layer node as shown above.

At this point, set the following:

- check Enabled check box in the Layer node
- select “Normal” in Blend Mode in the Layer node
- in the case of this project, rendered images in Visual Compositor can be displayed in the game view by selecting “Display 8” of the game view.



Enabled ✓

Blend Mode Normal

Use Input Alpha ✓

Mask Channel A

Opacity 1

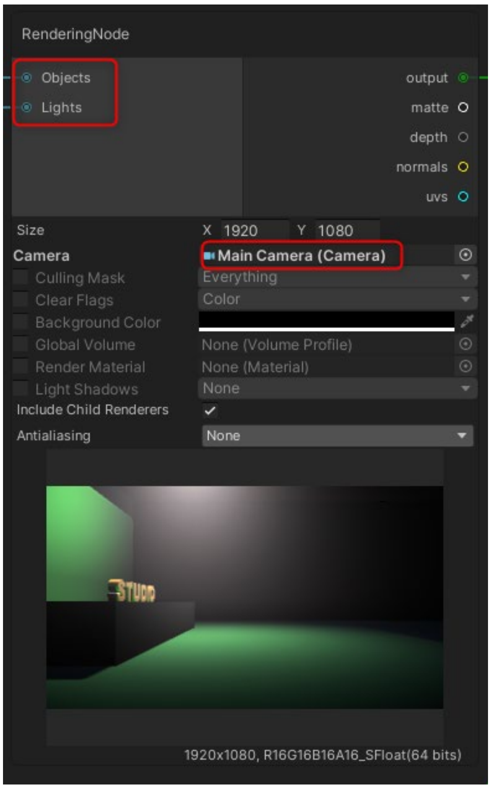
Premultiplied Alpha □

Export Mode Separate

✓ Export Render Texture None (Render Texture)

**Tips: Which display is used to display the result of Visual Compositor is set in “Target Display” of the Camera component to which Visual Compositor component is attached.**

# Generating Rendered Images - Render Node



RenderingNode

Objects
Lights

output
matte
depth
normals
uvs

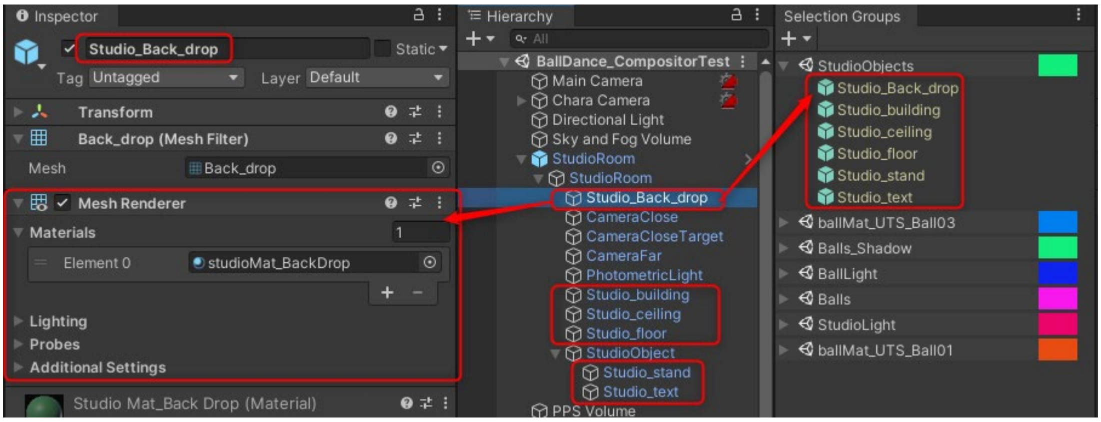
Size X 1920 Y 1080
Camera Main Camera (Camera)
Culling Mask Everything
Clear Flags Color
Background Color
Global Volume None (Volume Profile)
Render Material None (Material)
Light Shadows None
Include Child Renderers ✓
Antialiasing None

1920x1080, R16G16B16A16_SFloat(64 bits)

Render node renders a group of objects and lights connected to the Input port with a specified camera.
When post processes are specified for each camera, they are also applied.

In the above example, Main Camera in Hierarchy window is used as the rendering camera. CinemachineBrain component is attached to the Main Camera, and so images are rendered according to Cinemachine Track of Timeline.

Target rendering objects are retrieved in the StudioObjects selection group. You should not specify the parent on the top of the hierarchy, but each game object that contains Mesh Renderer.

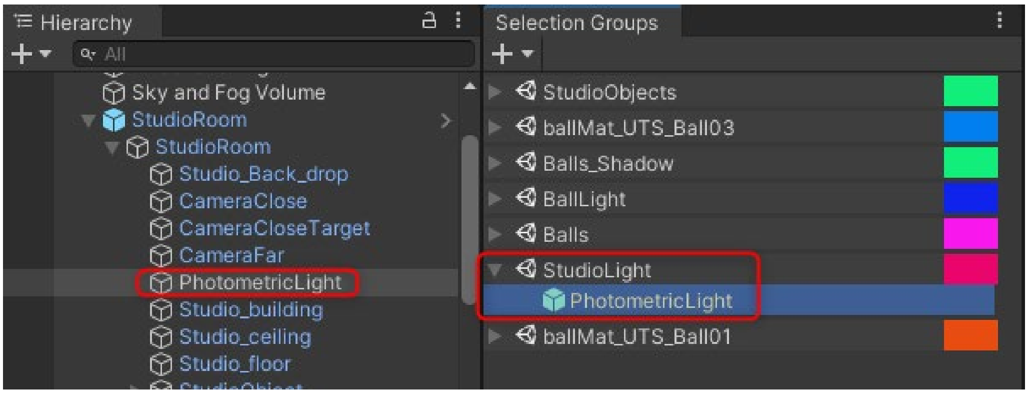


Inspector
✓ Studio_Back_drop Static
Tag Untagged Layer Default
Transform
Back_drop (Mesh Filter)
Mesh Back_drop
✓ Mesh Renderer
Materials 1
Element 0 studioMat_BackDrop
Lighting
Probes
Additional Settings
Studio Mat_Back Drop (Material)

Hierarchy
BallDance_CompositorTest
Main Camera
Chara Camera
Directional Light
Sky and Fog Volume
StudioRoom
StudioRoom
Studio_Back_drop
CameraClose
CameraCloseTarget
CameraFar
PhotometricLight
Studio_building
Studio_ceiling
Studio_floor
StudioObject
Studio_stand
Studio_text
PPS Volume

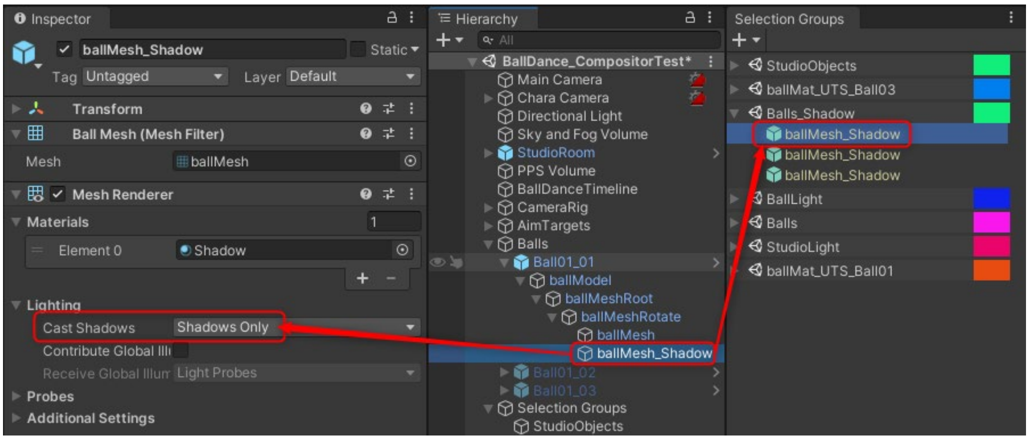
Selection Groups
StudioObjects
Studio_Back_drop
Studio_building
Studio_ceiling
Studio_floor
Studio_stand
Studio_text
ballMat_UTS_Ball03
Balls_Shadow
BallLight
Balls
StudioLight
ballMat_UTS_Ball01

The volumetric light used as a background rendering light is retrieved from Studio Room hierarchy and added to StudioLight selection group.



## Rendering Shadow Model and Adding a Shadow

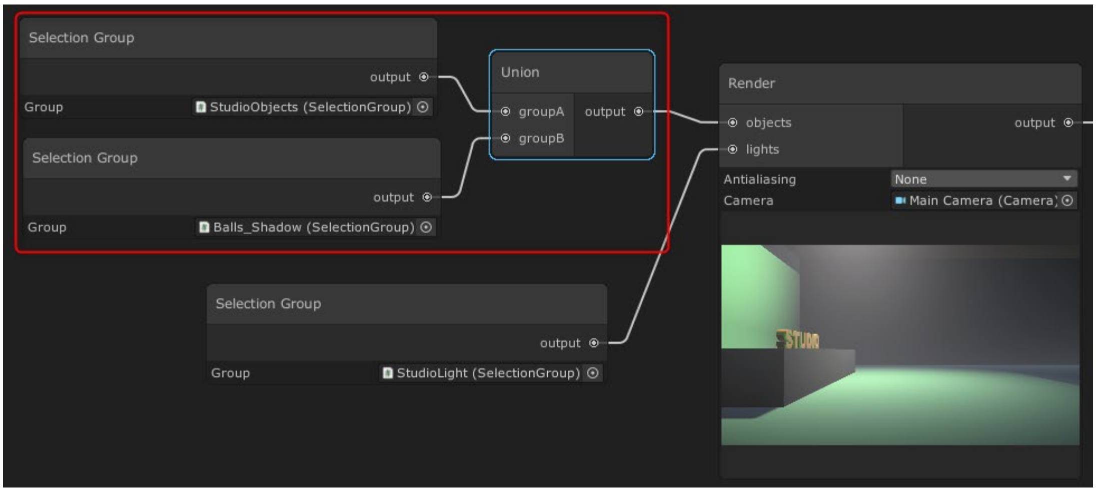
Add shadows of characters (color balls) to the background image.



To make a shadow model, duplicate ballMesh that contains the ball mesh, rename it to ballMesh_Shadow, and then go to Lighting > Cast Shadows in Mesh Renderer and select "Shadows Only." This creates an object that casts only shadows.

Make Balls_Shadow selection group that has only objects for casting shadows, put it together with StudioObjects selection group in the Union node, and connect it to the objects port of Render node.

Since the order of Union node is organized according to the view of camera, the stacking order won`t be changed regardless of whether to connect groupA or groupB. The stacking order can be changed only by the operation in Layer node.



Group StudioObjects (SelectionGroup)

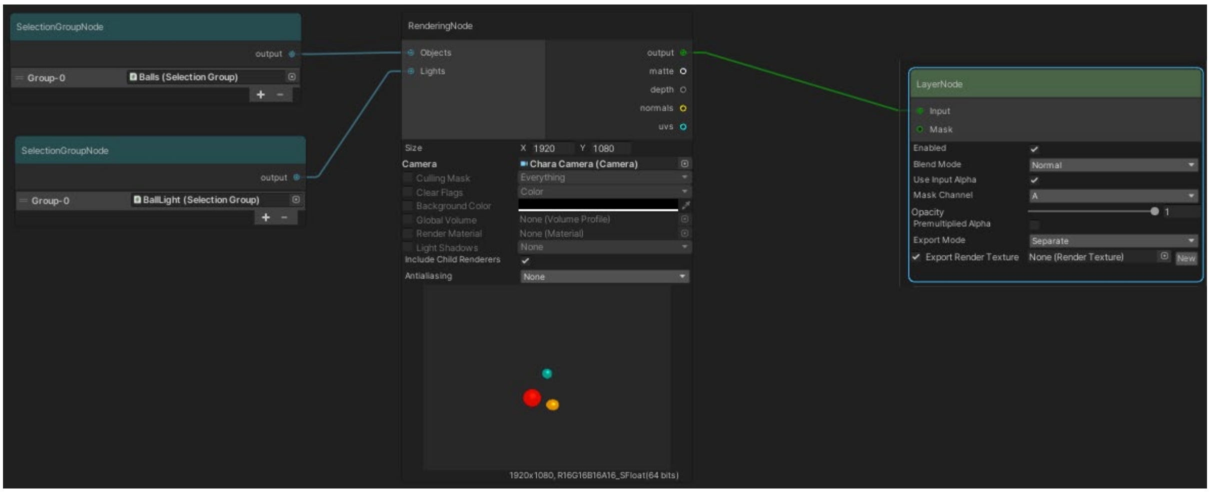
Group Balls_Shadow (SelectionGroup)

Group StudioLight (SelectionGroup)

Antialiasing None

Camera Main Camera (Camera)

## 2nd Layer: Rendering Cell Look Characters



Group-0 Balls (Selection Group)

Group-0 BallLight (Selection Group)

Size X 1920 Y 1080

Camera Chara Camera (Camera)

Culling Mask Everything

Clear Flags Color

Background Color

Global Volume None (Volume Profile)

Render Material None (Material)

Light Shadows None

Include Child Renderers ✓

Antialiasing None

1920x1080, R16G16B16A16_SFloat(64 bits)

Enabled ✓

Blend Mode Normal

Use Input Alpha ✓

Mask Channel A

Opacity 1

Premultiplied Alpha

Export Mode Separate

✓ Export Render Texture None (Render Texture)

In the 2nd layer, cell look characters (in this example, three color balls) will be rendered. Directional light is used for Look Dev of the cell look characters. The direction of the directional light determines the application of the base color and shade color. For cell look characters, design elements add some influence on the application of these colors. In other words, the direction and distribution of 2 different type of colors do not necessarily have to match the direction of the light that illuminates the entire background objects of the scene. **Rather, you should prioritize the appearance of the characters you desire in each scene.**

The rendered image of the character is combined on the background image. For that purpose, a mask for carving the character part is necessary. (Alpha channel is not automatically generated. It is one of the characteristics of node-basis work flow that it is necessary to give one-by-one precise instruction. It has the advantage of allowing you to build a robust composite flow with few errors). In this layer, those masks are also generated.

## Adding Layer Node

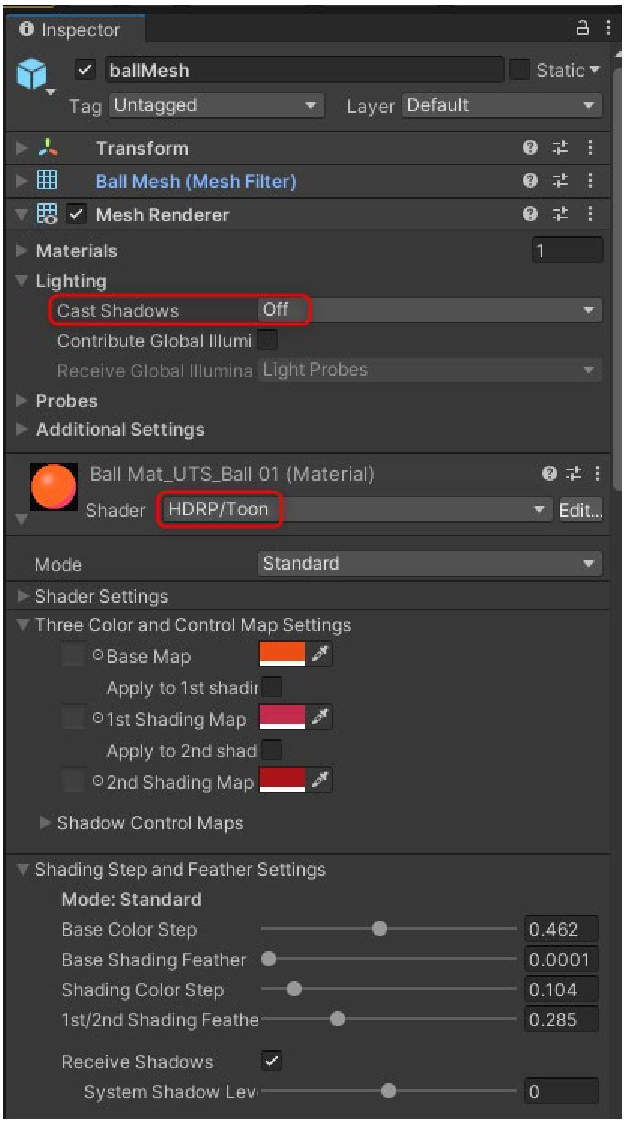
In Create Node window, create a Layer node and drag it below the 1st Layer node in Visual Compositor window. The two Layer nodes then merged and create a Layer Stack node. The 2nd layer is now on top of the 1st layer.

Using the newly added Layer node allows you to combine "a rendered character image with settings of light and camera" that is completely different from the settings for the background objects with the background image.

Now, let's prepare the settings for rendering character images to be displayed in the second layer.

## Setting UTS/HDRP Material

Change the material of ballMesh, the rendering target, to UTS/HDRP. Change the material's “Shader” to HDRP/Toon to create the look you desire. “Cast Shadows” that is for the ball to cast shadows should be turned off in this layer since shadows are handled in the background image side.



Inspector

☑ ballMesh □ Static

Tag Untagged Layer Default

Transform

Ball Mesh (Mesh Filter)

☑ Mesh Renderer

Materials 1

Lighting

Cast Shadows Off

Contribute Global Illumi □

Receive Global Illumina Light Probes

Probes

Additional Settings

Ball Mat_UTS_Ball 01 (Material)

Shader HDRP/Toon Edit...

Mode Standard

Shader Settings

Three Color and Control Map Settings

Base Map

Apply to 1st shadir □

1st Shading Map

Apply to 2nd shad □

2nd Shading Map

Shadow Control Maps

Shading Step and Feather Settings

Mode: Standard

Base Color Step 0.462

Base Shading Feather 0.0001

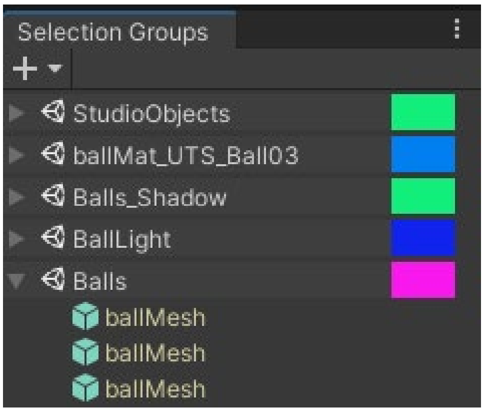
Shading Color Step 0.104

1st/2nd Shading Feathe 0.285

Receive Shadows ☑

System Shadow Lev 0

After adding changes to the material, put ballMesh for each ball together into Balls selection group and make them selectable as a group.

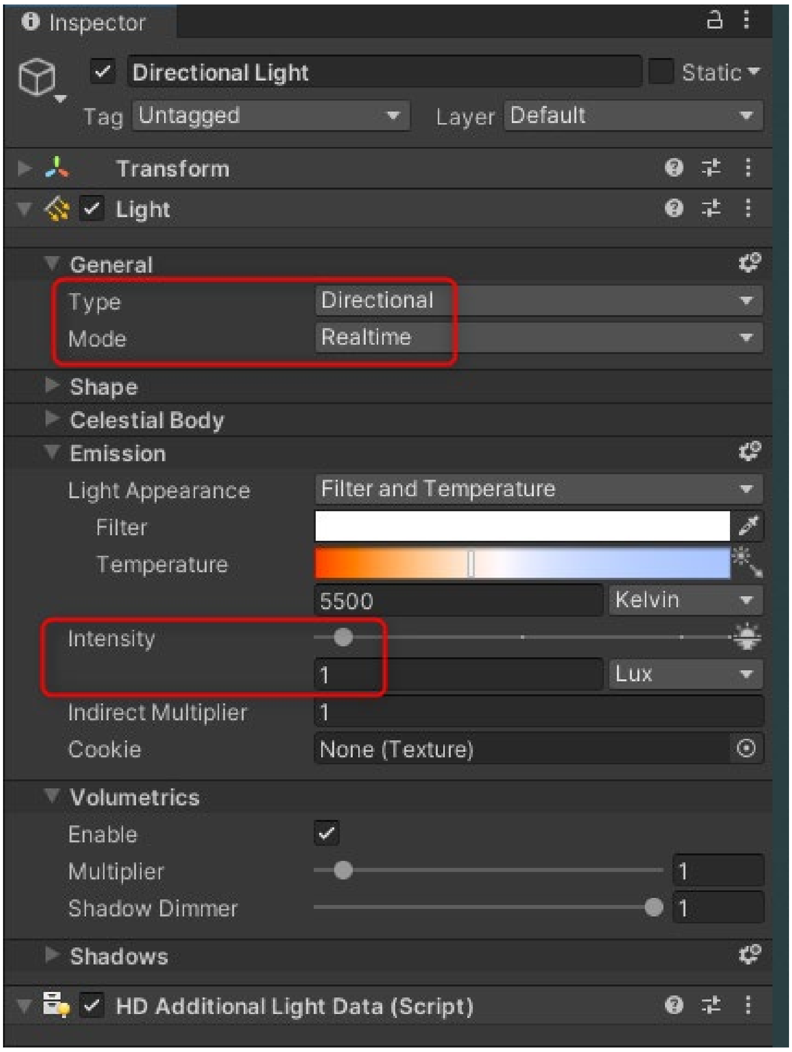


- StudioObjects 🟩
- ballMat_UTS_Ball03 🟦
- Balls_Shadow 🟩
- BallLight 🟦
- Balls 🟪
  - ballMesh
  - ballMesh
  - ballMesh

## Making Character Light

Here, we make a dedicated light for rendering characters. For the light for characters, we use a real-time directional light that works best with cell shaders.

HDRP directional light is set with its default intensity of 10000 Lux for "bright outdoor light source. It is too bright for a cell shader, set Intensity to 1 as shown below.



Inspector

☑ Directional Light □ Static

Tag Untagged Layer Default

Transform

☑ Light

General

Type Directional

Mode Realtime

Shape

Celestial Body

Emission

Light Appearance Filter and Temperature

Filter

Temperature

5500 Kelvin

Intensity

1 Lux

Indirect Multiplier 1

Cookie None (Texture)

Volumetrics

Enable ☑

Multiplier 1

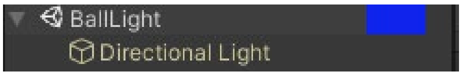
Shadow Dimmer 1

Shadows

☑ HD Additional Light Data (Script)

**Tips: If you use a cell shader with HDRP light of 1 Lux intensity or higher, you can activate Scene Lights Hi-Cut Filter of UTS/HDRP material to effectively suppress overexposure.**

**It is also good idea to include the directional light for characters in BallLight selection group.**



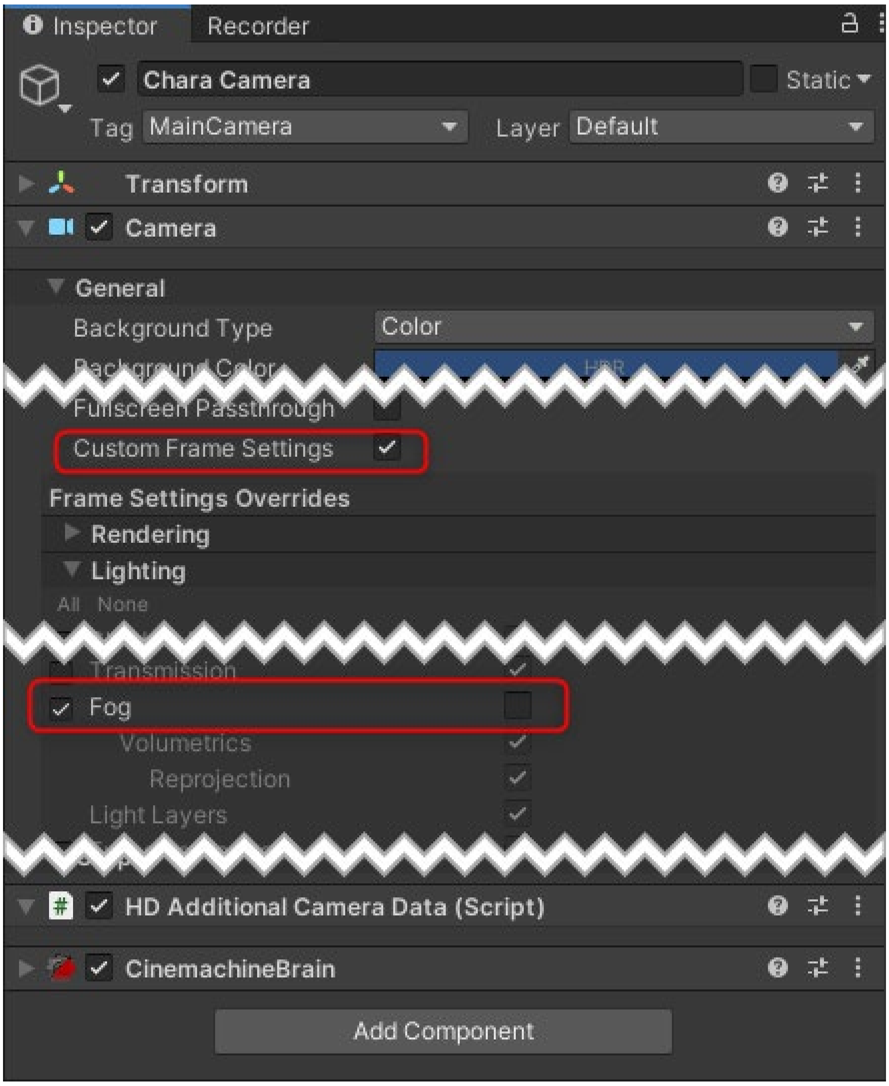
BallLight

Directional Light

## Making Character Camera

Follow the steps below to make a dedicated camera for rendering your character.

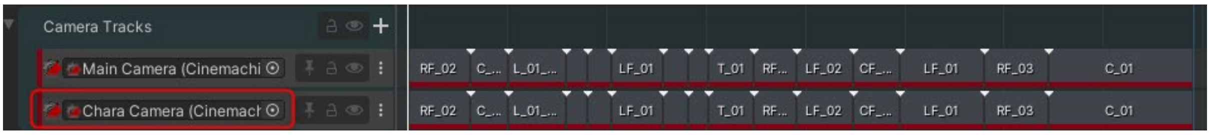
1. Since most of the settings are same with the ones of Main Camera, duplicate Main Camera from Hierarchy window and rename it to Chara Camera. Since CinemachineBrain component is attached to the Main Camera, it is also attached to the duplicated Chara Camera.
2. Open the Inspector window of Chara Camera, expand General in the Camera component and check Custom Frame Settings.
3. Additional section "Frame Settings Overrides" will be appeared. Open Lighting, turn on the left check box of "Fog", and then turn off the right check box of "Fog".
   This disables Fog of Chara Camera. This operation is required for Mask node, which will be explained later, to work properly.



☑ Chara Camera □ Static
Tag MainCamera Layer Default
Background Type Color
Custom Frame Settings ☑
Frame Settings Overrides
Lighting
Transmission ✓
☑ Fog □
Volumetrics ✓
Reprojection ✓
Light Layers ✓

## Registering Character Camera to Cinemachine Track

1. To run Chara Camera on Timeline's Cinemachine Track just like Main Camera, duplicate the Cinemachine Track which the Main Camera is bound.
2. Bind a new Chara Camera to the duplicated Cinemachine Track.
3. Two Cinemachine Tracks are in the Timeline. Put them together in Track Group.



## Carving Rendered Character Image with Mask

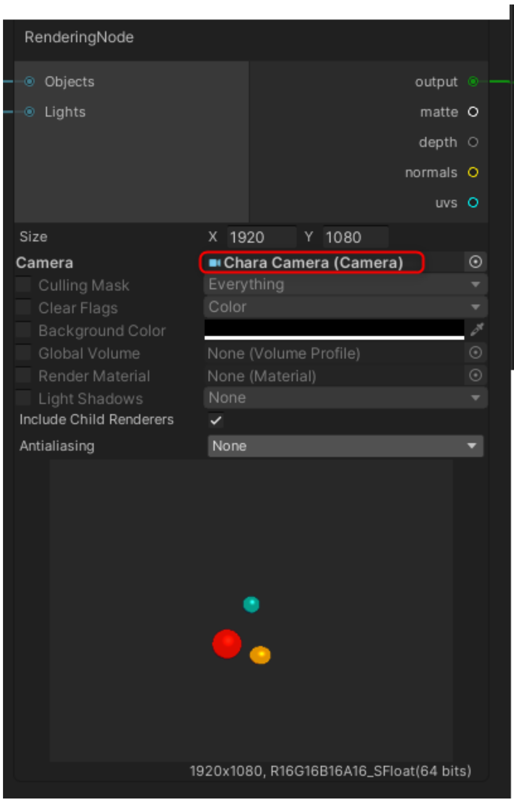
Add two Selection Group nodes and bind each of the following one by one:

- Balls
- BallLight

Next, add Render node to render character images.
At the same time, add Mask node to make a mask that carve character images.
Like Render node, Mask node also needs to connect with Objects and Lights selection groups and specify a camera. Each should be arranged as follows:

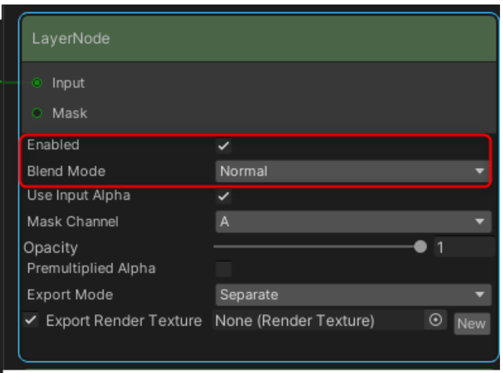
- Objects port ⇒ connecting to **Balls** selection group
- lights port ⇒ connecting to **BallLight** selection group
- Camera ⇒ specify **Chara Camera**

Finally, connect the output from each node to the input port of the Layer node, check Enabled, and set Blend Mode to Normal.



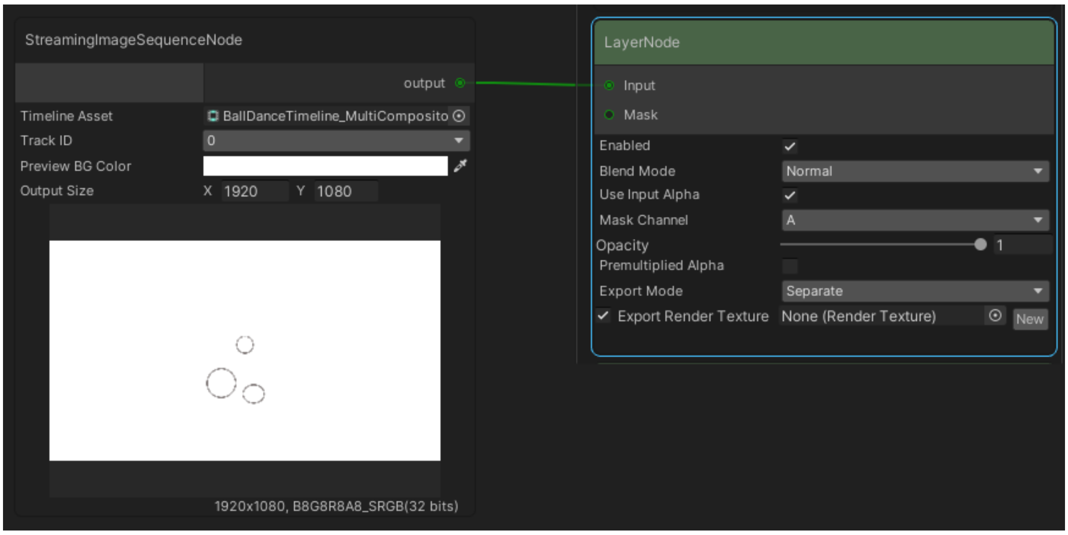
Size X 1920 Y 1080
Camera: Chara Camera (Camera)
Culling Mask: Everything
Clear Flags: Color
Background Color
Global Volume: None (Volume Profile)
Render Material: None (Material)
Light Shadows: None
Include Child Renderers: ✓
Antialiasing: None

1920x1080, R16G16B16A16_SFloat(64 bits)



Enabled: ✓
Blend Mode: Normal
Use Input Alpha: ✓
Mask Channel: A
Opacity: 1
Premultiplied Alpha: □
Export Mode: Separate
✓ Export Render Texture: None (Render Texture) New

# 3rd Layer: Composite of Pencil+ Line Cache



StreamingImageSequenceNode

Timeline Asset: BallDanceTimeline_MultiComposito
Track ID: 0
Preview BG Color
Output Size: X 1920 Y 1080

1920x1080, B8G8R8A8_SRGB(32 bits)

LayerNode

Enabled: ✓
Blend Mode: Normal
Use Input Alpha: ✓
Mask Channel: A
Opacity: 1
Premultiplied Alpha: □
Export Mode: Separate
✓ Export Render Texture: None (Render Texture)

In the 3rd layer, we will composite toon lines, which is important for the design of a cell look character. Toon line uses Pencil+ Line Cache, which was pre-rendered with Pencil+ 4 Line for Unity, as the source of serial number images.

You can make a toon line in real time with Pencil+ 4 Line for Unity every time, but with Pencil + Line Cache, you can have the following merits:

- Even when a character mesh becomes detailed and it gets difficult to be rendered in real time with Pencil+ 4 Line for Unity, you can review the result by scrubbing lightly using the cache.
- It is also possible to modify the cached toon line image manually. Therefore, you don't need to change the settings of Pencil+ 4 Line for Unity every time you make minor modifications.

Pencil+ Line Cache can be used with compositing a screen with Canvas/Image, but this time, we will introduce the way using Visual Compositor node.

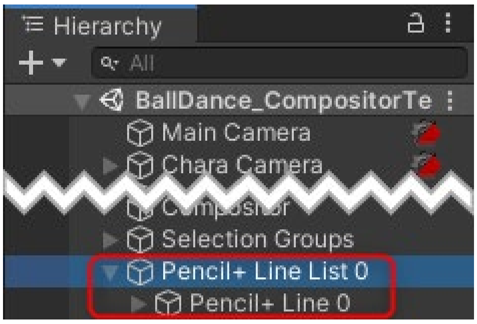
## Preparation of Pencil+ 4 Line for Unity

Now, we will set up Pencil+ 4 Line for Unity for the project.
For detailed steps, see a document “Pencil+ Line Cache Setup” and Help pages of Pencil+ 4 Line for Unity production version.
We will describe the setup for package of HDRP version here.

**Note: Turn on the Read/Write Enabled check box when importing the FBX model to which Pencil+ Line is applied. If you forget this, Pencil+ Line will not appear for meshes imported from outside.**

## Create Pencil+ Line List Game Object in Scene

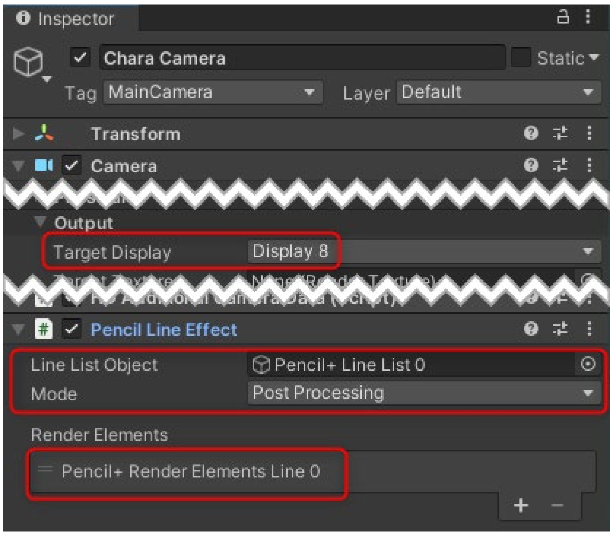
1. First, create an empty game object in the scene and attach Line List Node component to it.
2. From the Inspector, add a new Line List to the Line List Node component and rename the game object to the same name with the Line List.
3. Register materials of the character (to which you want to apply Pencil+ Line) to the created Line List and set up Line (which cannot be displayed at this stage yet).



## Attaching Pencil Line Effect to Character Camera

Since Pencil+ Line is a toon line camera shader, attach it to the camera that renders characters. Since the character camera is Chara Camera, attach the Pencil Line Effect component to it as follows:

1. In Inspector window, expand Output of Chara Camera and set Target Display to Display 8.
2. Next, attach Pencil Line Effect to the Chara Camera using Add Component button.
3. Register the Pencil+ Line List Object created above to Line List Object of Pencil Line Effect. Next, set Mode to “Post Processing” and finally add a new Render Elements.



Chara Camera

Tag MainCamera Layer Default

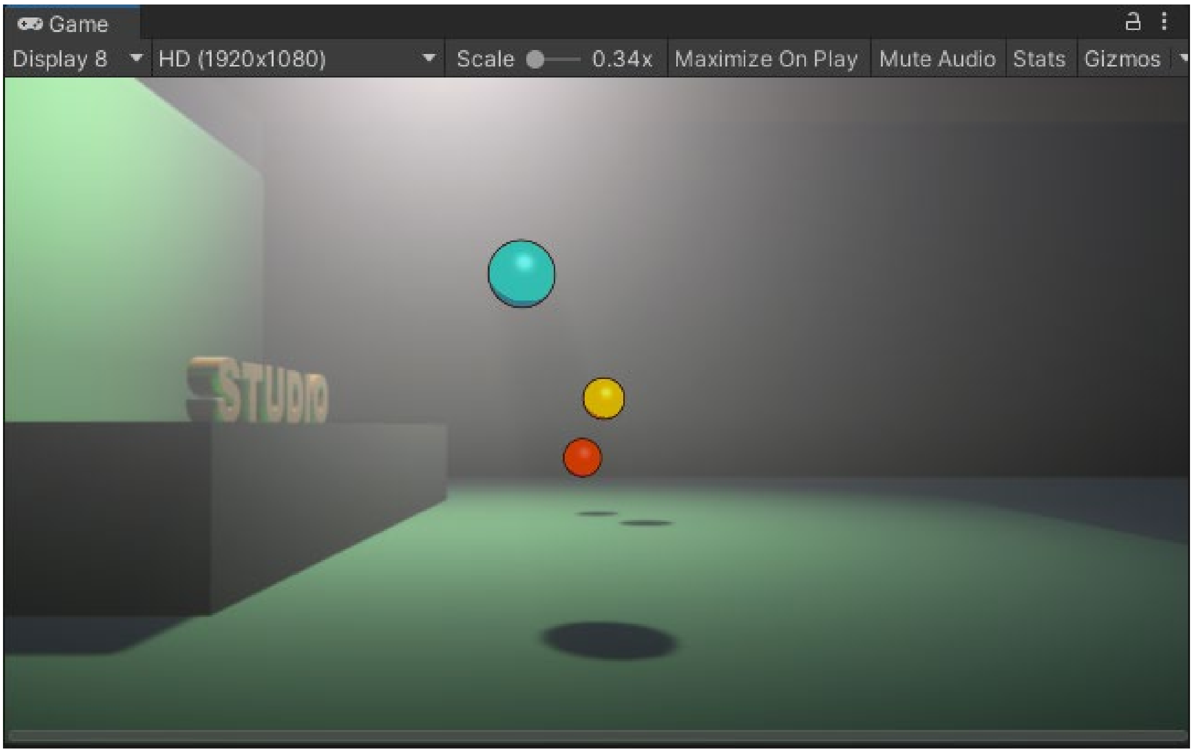
Target Display: Display 8

Line List Object: Pencil+ Line List 0

Mode: Post Processing

Render Elements: Pencil+ Render Elements Line 0

4. At this time, if Pencil+ Line List is set appropriately, you can check the toon line by switching the game view to Display 8.

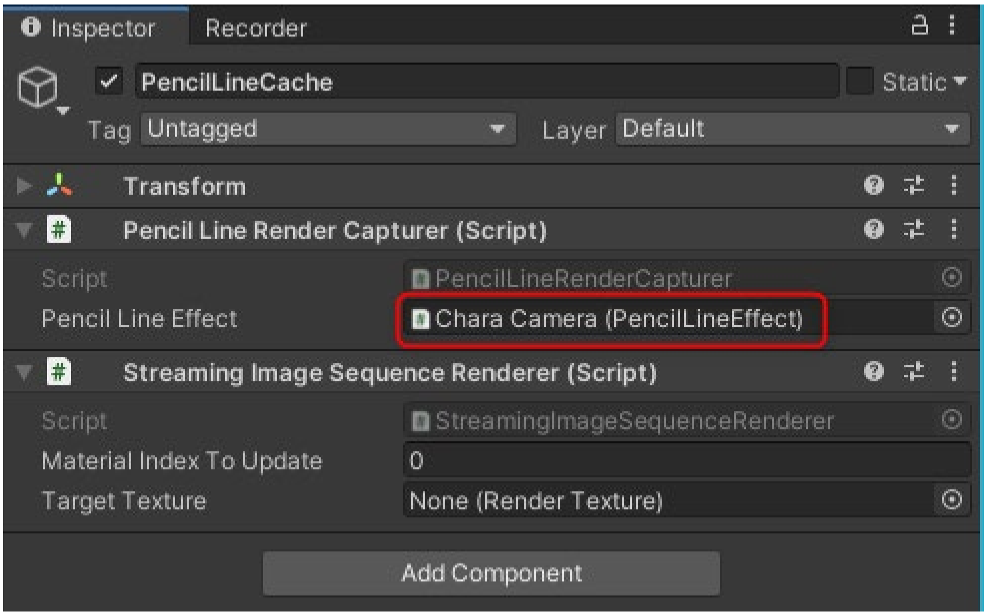

Game
Display 8
HD (1920x1080)
Scale
0.34x
Maximize On Play
Mute Audio
Stats
Gizmos
STUDIO

## Creating Pencil+ Line Cache Game Object

Next, create a Pencil+ Line Cache game object in the scene to reference it from Timeline.

1. Create an empty game object in Hierarchy window and rename it to PencilLineCache.
2. Attach Pencil Line Render Capture component to the created game object.
3. Find “Pencil Line Effect “ in the section of Pencil Line Render Capturer in Inspector window, and specify a camera to which Pencil Line Effect is attached. In this example, it is Chara Camera.
4. Next, attach Streaming Image Sequence Renderer component to the same game object.

As the result of the above, PencilLineCache game object looks like the image below. PencilLineCache game object has functions to create cache images of Pencil+ Line and play them back.



☑ PencilLineCache

Tag Untagged

Layer Default

Pencil Line Render Capturer (Script)

Script PencilLineRenderCapturer

Pencil Line Effect Chara Camera (PencilLineEffect)

Streaming Image Sequence Renderer (Script)

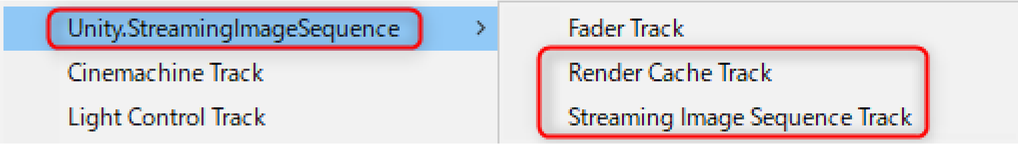
Script StreamingImageSequenceRenderer

Material Index To Update 0

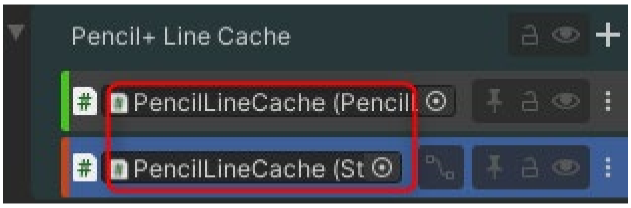
Target Texture None (Render Texture)

## Making Record of Cache Data and Playback Track in Timeline

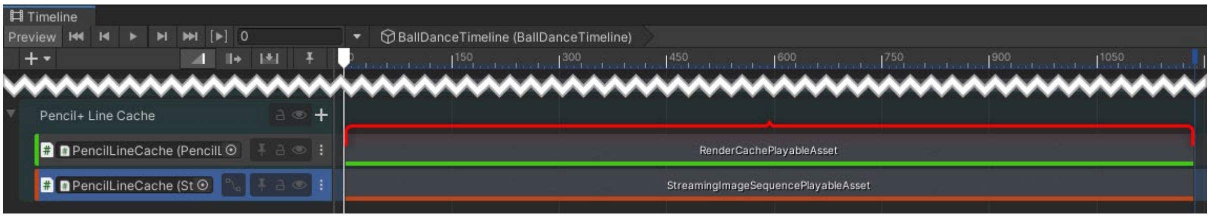
In Timeline, create a Render Cache Track that records cache data of Pencil+ Line and a Streaming Image Sequence Track that plays the cache data. To add these, click “+” button in the upper left in Timeline or right click on any empty space on Timeline track and select the sub items under Unity.StreamingImageSequence in the menu.



1. Make a new Render Cache Track in Timeline window and add a Render Cache Playable Asset to it.
2. Similarly, in Timeline window, make a new Streaming Image Sequence Track and add a Streaming Image Sequence Playable Asset to it.
3. Bind a PencilLineCache game object to both tracks created above and gather them into a track group Pencil+ Line Cache.

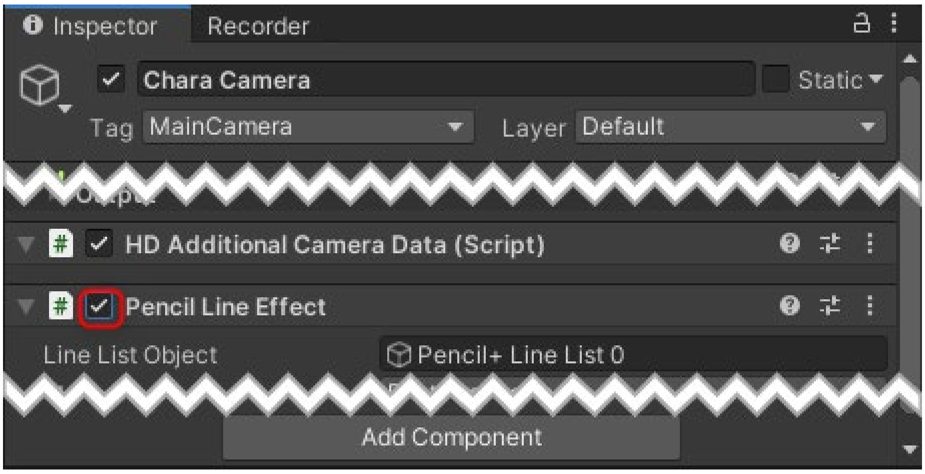


4. Adjust both clips to the same length according to the recording duration of the cache.



## Recording Chache Data

1. Turn on Pencil Line Effect component attached to Chara Camera



☑ Chara Camera

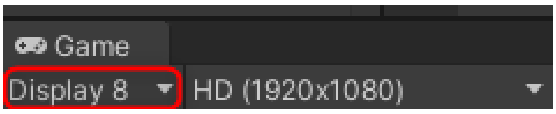
Tag MainCamera

Layer Default

☑ Pencil Line Effect

Line List Object: Pencil+ Line List 0

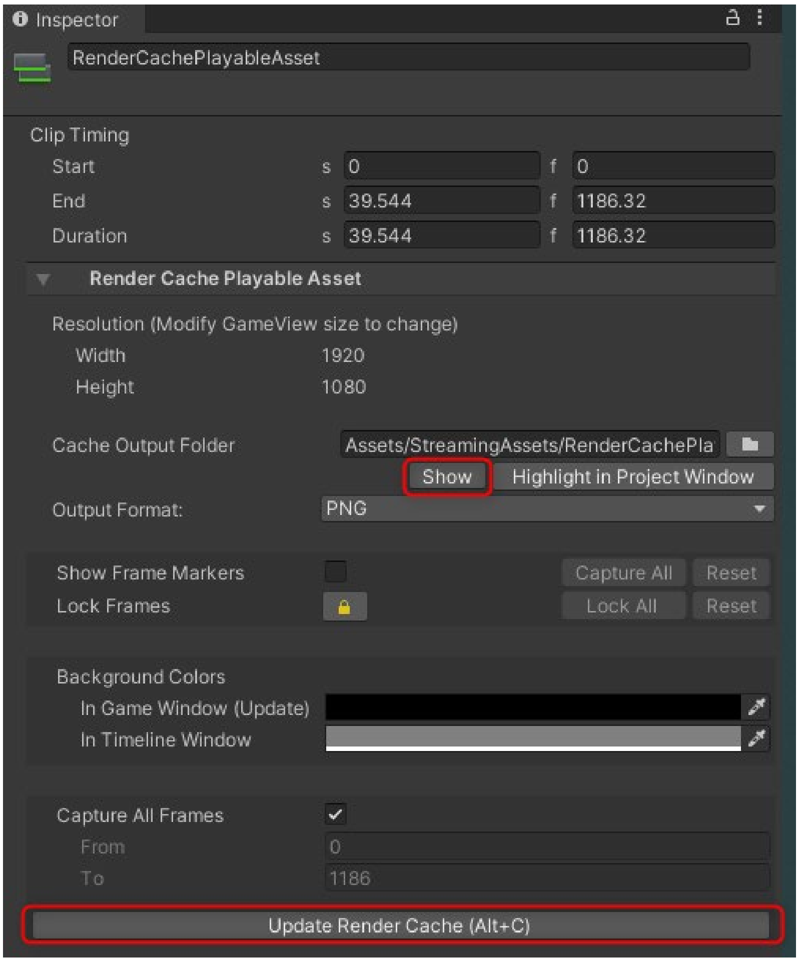
2. Switch the game view to “Display 8”



Display 8

HD (1920x1080)

3. In Timeline window, select “Render Cache Playable Asset” clip
4. Click “Update Render Cache” button to save Pencil+ Line Cache in StreamingAssets folder



RenderCachePlayableAsset

Clip Timing

| | | | | |
|---|---|---|---|---|
| Start | s | 0 | f | 0 |
| End | s | 39.544 | f | 1186.32 |
| Duration | s | 39.544 | f | 1186.32 |

Render Cache Playable Asset

Resolution (Modify GameView size to change)

Width 1920

Height 1080

Cache Output Folder: Assets/StreamingAssets/RenderCachePla

Output Format: PNG

Show Frame Markers □

Lock Frames

Background Colors

In Game Window (Update)

In Timeline Window

Capture All Frames ☑

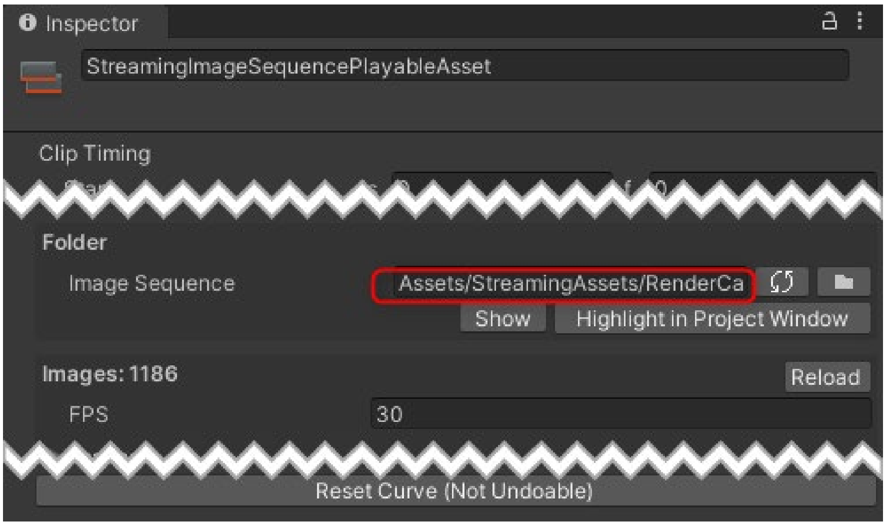
From 0

To 1186

## Showing Cache Data in Visual Compositor

We use Stream Image Sequence Node to handle line cache data in Visual Compositor window. Before that, make the Streaming Image Sequence track in Timeline accessible to the cache data.

1. Select Streaming Image Sequence Playable Asset and go to Image Sequence field in Folder section. Click the folder icon and select the folder under StreamingAssets folder that saves the cache data (meaning, Cache Output Folder of RenderCachePlayableAsset).



Inspector

StreamingImageSequencePlayableAsset

Clip Timing

Folder

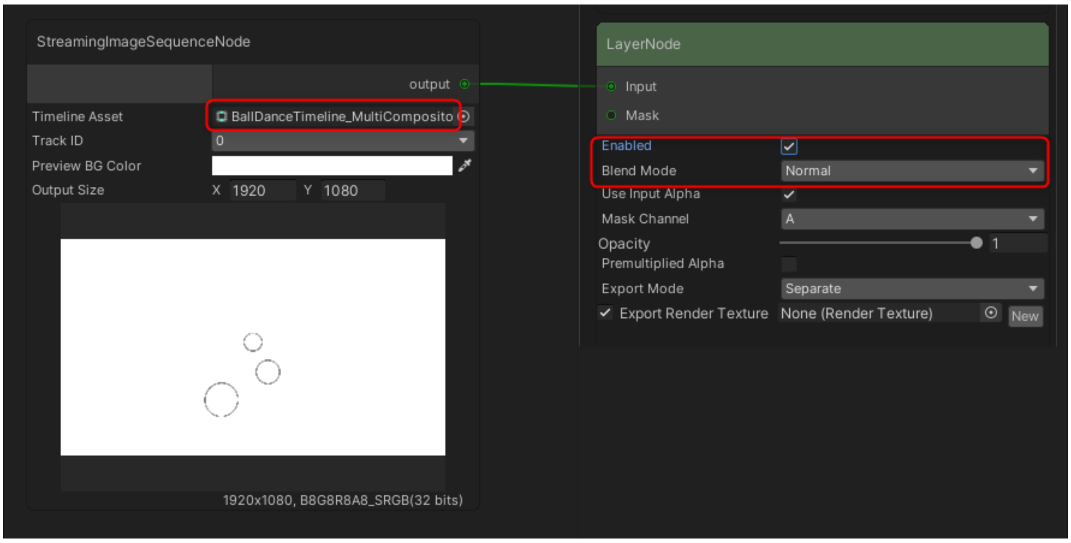
Image Sequence: Assets/StreamingAssets/RenderCa

Images: 1186

FPS: 30

2. In Visual Compositor window, add a new Stream Image Sequence Node and a Layer Node. Connect those two nodes. As same in 2nd layer, drag the layer stack to be the 3rd layer below the layer stack previously created and dock with Layer Stack. Now, the 3rd layer overlays over the result of the 1st and 2nd layers.

3. Specify Timeline Playable Asset as Director in Stream Image Sequence Node. Turn on “Enabled” check box and set Blend Mode to Normal in Layer Node.



StreamingImageSequenceNode

Timeline Asset: BallDanceTimeline_MultiComposito

Track ID: 0

Preview BG Color

Output Size: X 1920 Y 1080

1920x1080, B8G8R8A8_SRGB(32 bits)

LayerNode

Enabled: ☑

Blend Mode: Normal

Use Input Alpha: ✓

Mask Channel: A

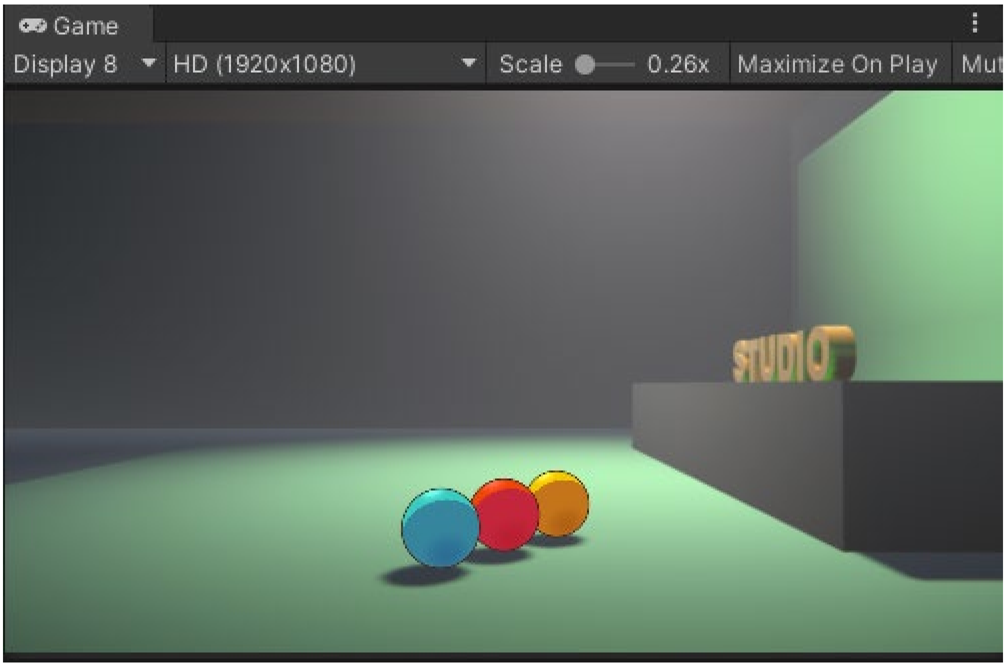
Opacity: 1

Premultiplied Alpha: □

Export Mode: Separate

✓ Export Render Texture: None (Render Texture)

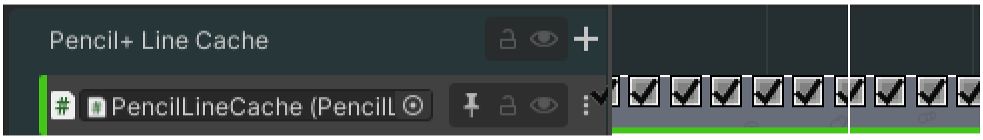
4. Make sure Display 3 is selected for the game view. Scrub the Timeline playhead from side by side to see that the line cache data is composited with an image and played by Visual Compositor.



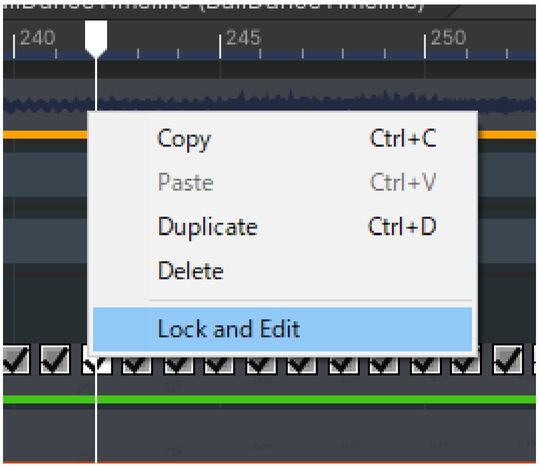
## Modifying Individual Cash Image Data Manually

Finally, we will explain how to modify the cached line image data as serial number images with Photoshop and other software.

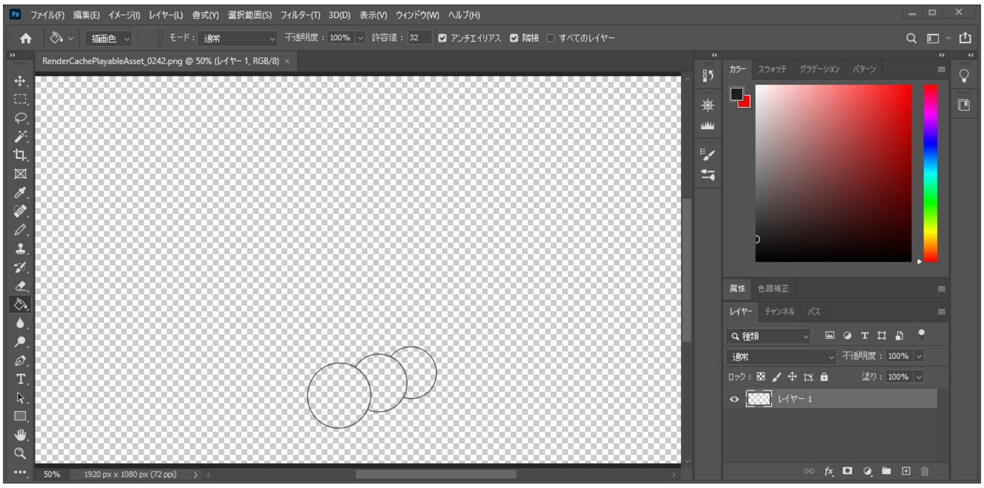
1. Scrub the playhead in Timeline and find a frame you want to modify
2. Zoom in until the gauge for one frame becomes large enough in Timeline
3. Select Render Cache Playable Asset clip that recorded the cached data and check Show Frame Markers in the Inspector. Then, checkboxes of currently enabled cache data frames are checked on the clip.



4. Select a frame you want to modify and right click to show the context menu. Select "Lock and Edit" from the context menu or click "Lock and Edit" button in Inspector window.

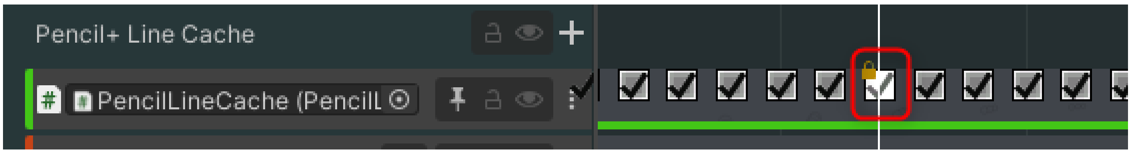


5. The specified frame is read into the image editor such as Photoshop registered in Unity. Modify it on the editor.

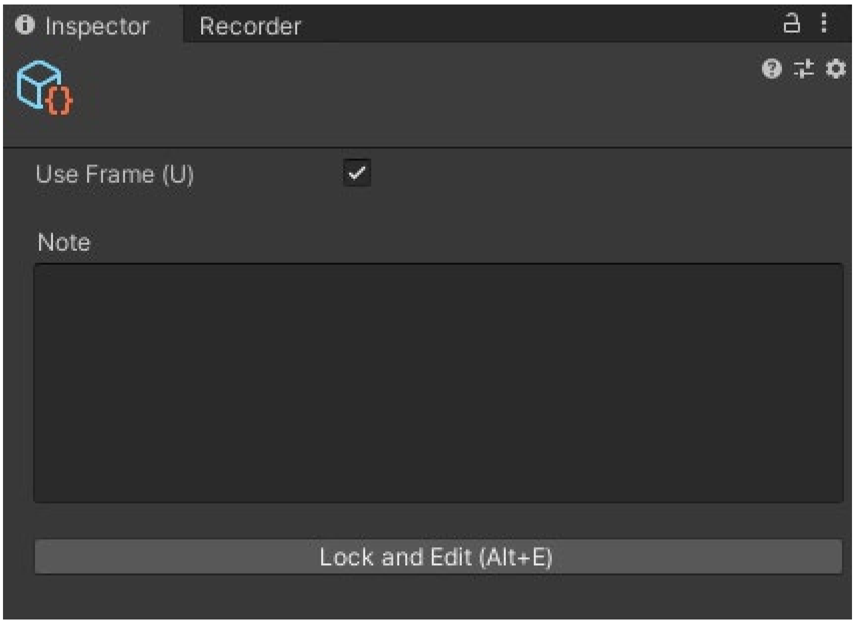

ファイル(F) 編集(E) イメージ(I) レイヤー(L) 書式(Y) 選択範囲(S) フィルター(T) 3D(D) 表示(V) ウィンドウ(W) ヘルプ(H)
RenderCachePlayableAsset_0242.png @ 50% (レイヤー 1, RGB/8)
50%
1920 px x 1080 px (72 ppi)
レイヤー 1

**Tips: To register Photoshop, go to Edit > Preferences > External Tools > Image application**

6. Since the modified frame is locked, so the modified frame is kept as it is even if the cache data is updated.

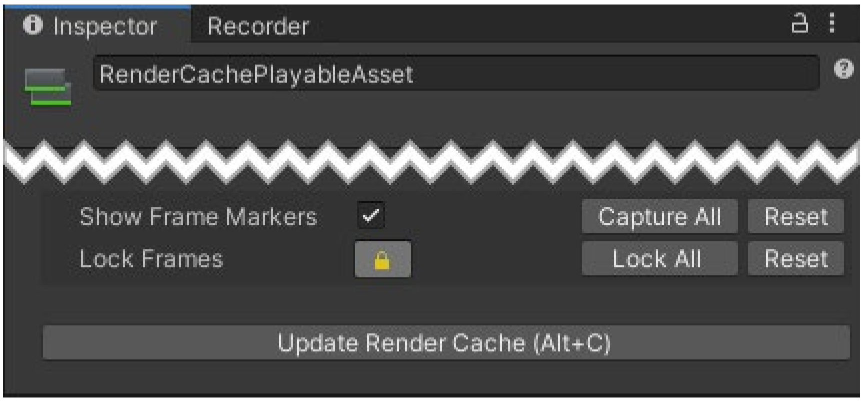


7. You also leave a note in Note area of Inspector window, while selecting the modified frame

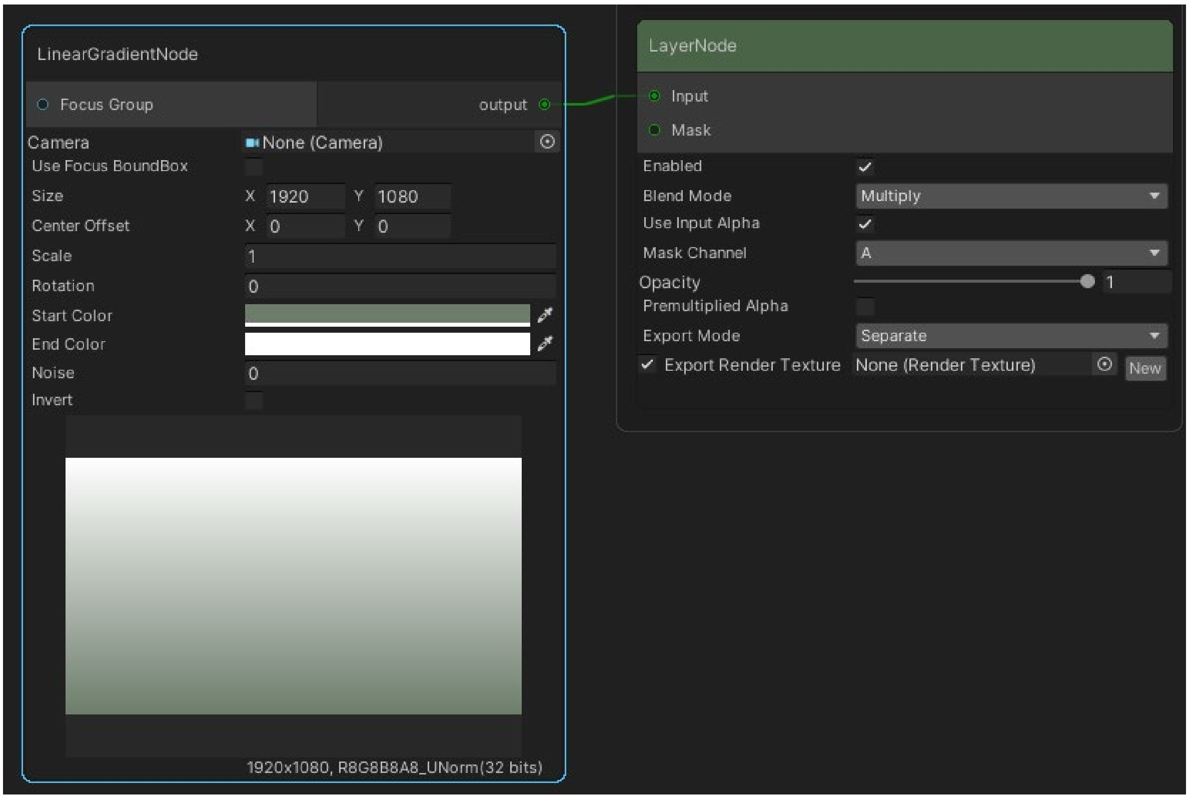


8. Select Render Cache Playable Asset clip again and uncheck “Show Frame Markers.” Then, the checkboxes are disappeared from the clip.

9. To reset the rock, click Lock Frames🔒 button and then click Reset button



## 4th Layer: Adding Para-Harmony Effect(Darker effect on the entire screen) to Make Whole Screen Look Natural



LinearGradientNode

Camera: None (Camera)
Use Focus BoundBox: □
Size: X 1920 Y 1080
Center Offset: X 0 Y 0
Scale: 1
Rotation: 0
Start Color
End Color
Noise: 0
Invert: □

1920x1080, R8G8B8A8_UNorm(32 bits)

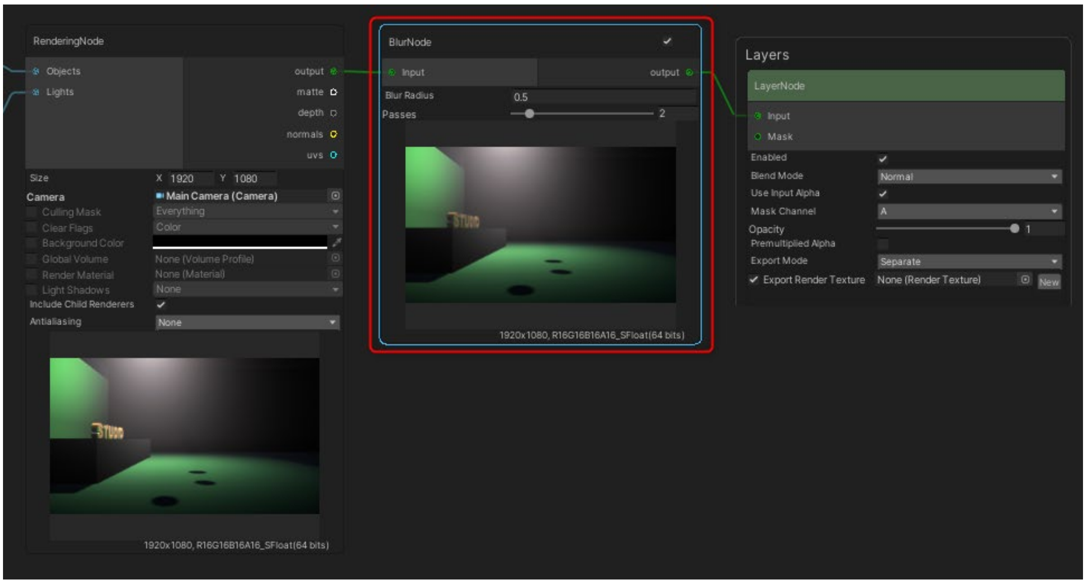
LayerNode

Enabled: ✓
Blend Mode: Multiply
Use Input Alpha: ✓
Mask Channel: A
Opacity: 1
Premultiplied Alpha: □
Export Mode: Separate
✓ Export Render Texture: None (Render Texture) New

In the last 4th layer, we add the process (para-harmony effect) that makes the background image and cell look images look natural.

There are various methods for the process depending on composite cuts you want to make. Here, as an example, we made a top-to-bottom gradient image in Linear Gradient node and blended it in Multiply mode.

It is also a good idea to create a noise image of film grain separately and apply it on the image.



RenderingNode

Size: X 1920 Y 1080
Camera: Main Camera (Camera)
Culling Mask: Everything
Clear Flags: Color
Background Color
Global Volume: None (Volume Profile)
Render Material: None (Material)
Light Shadows: None
Include Child Renderers: ✓
Antialiasing: None

1920x1080, R16G16B16A16_SFloat(64 bits)

BlurNode ✓

Blur Radius: 0.5
Passes: 2

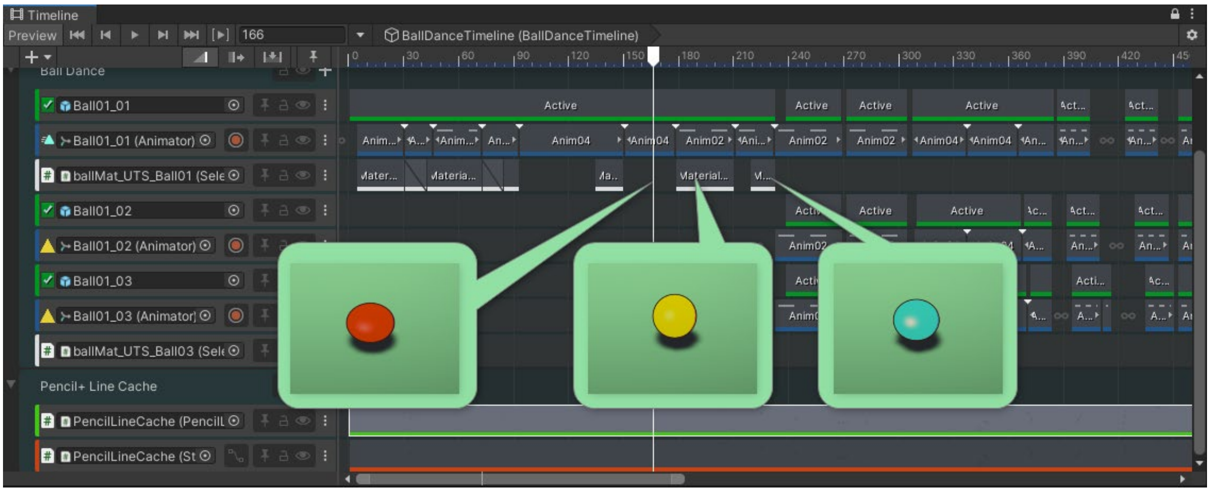
1920x1080, R16G16B16A16_SFloat(64 bits)

Layers – LayerNode

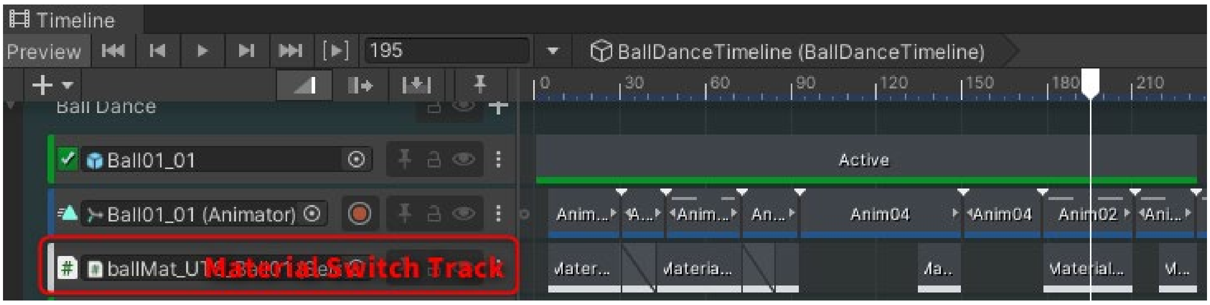
Enabled: ✓
Blend Mode: Normal
Use Input Alpha: ✓
Mask Channel: A
Opacity: 1
Premultiplied Alpha: □
Export Mode: Separate
✓ Export Render Texture: None (Render Texture) New

Besides, instead of using the rendered background image of the 1st layer as it is, you can add Multi Gaussian Blur node after rendering and blur it lighter than the cell look balls to make it look like a "2D background image." In this way, by applying the filters that are normally used for processing 2D images at each stage, it is possible to devise ways to make 3D rendering images look like 2D images.

# Changing Color in Timeline - Material Switch Track



While moving the playhead from frame 0 to 230 on the Timeline, three colored balls appear. In this scene, the color of the red ball is switched to yellow and blue in Timeline.



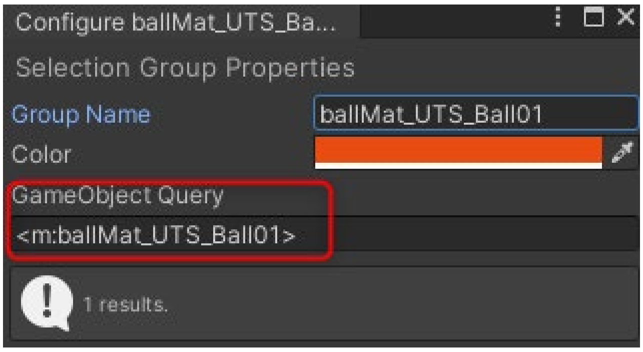
Material Switch feature changes the color by overriding material settings in the Material Switch Track.

Especially for changing colors, considering the convenience of the animation production workflow, it can directly read and change image data from such as a color design document for each scene.

In Material Switch Track also, use selection groups to select and specify materials you want to change.

Set up Material Switch Track as follows.

1. In the Selection Group, select a mesh of the ball to which the material you want to change is attached. You can search it by material name.



Configure ballMat_UTS_Ba...

Selection Group Properties

Group Name: ballMat_UTS_Ball01

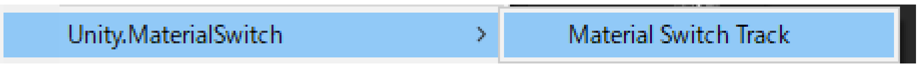
Color

GameObject Query

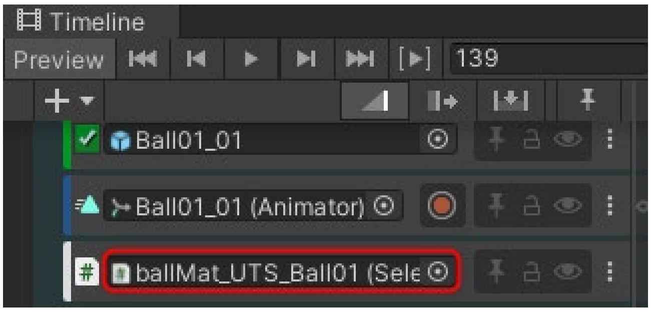
<m:ballMat_UTS_Ball01>

1 results.

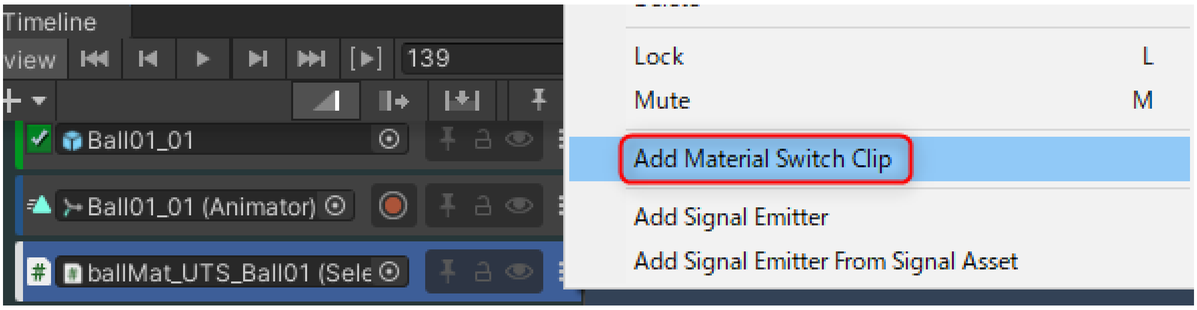
2. In Timeline, you can either click “+” at the upper left of Timeline or right-click on any empty space of Timeline track. Select “Add Track > Unity.MaterialSwitch > Material Switch Track” to make a new track.



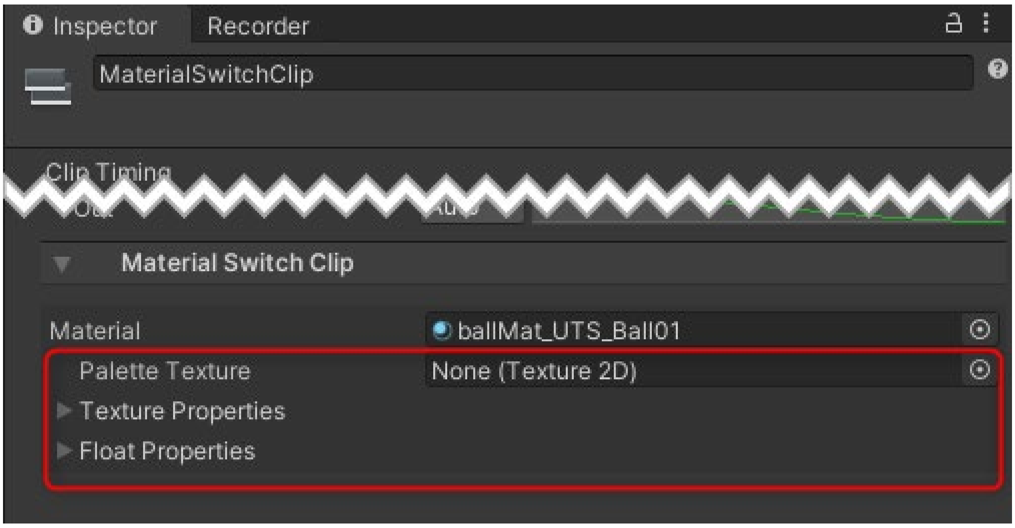
3. Bind the Selection Group which you want to add changes to the newly created Material Switch Track.



4. Right-click on the Material Switch Track and add a Material Switch Clip.



5. The newly created Material Switch Clip in the Inspector looks like the image below. Material is set in the bound Selection Group. You can add changes to the material by overriding the original material settings which is surrounded with a red frame below. Here, we will change the base color, 1st shade color, and 2nd shade colors of UTS/HDRP using a texture.



MaterialSwitchClip

Material Switch Clip

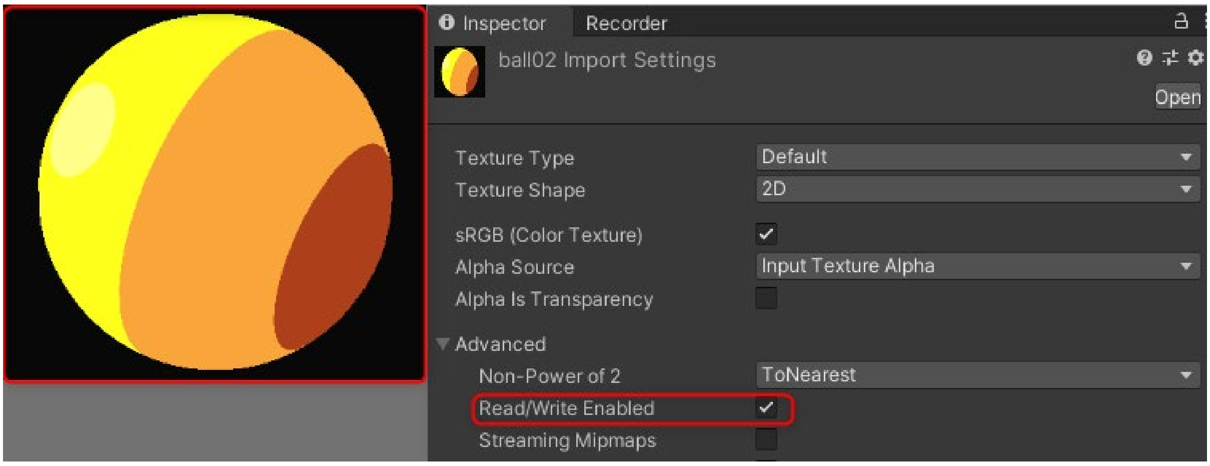
Material: ballMat_UTS_Ball01

Palette Texture: None (Texture 2D)

Texture Properties

Float Properties

6. To change the material of the red ball to yellow, prepare an image like the following. Check on “Read/Write Enabled” in the Import Settings of the image.



ball02 Import Settings

Texture Type: Default

Texture Shape: 2D

sRGB (Color Texture): ✓

Alpha Source: Input Texture Alpha

Alpha Is Transparency: □

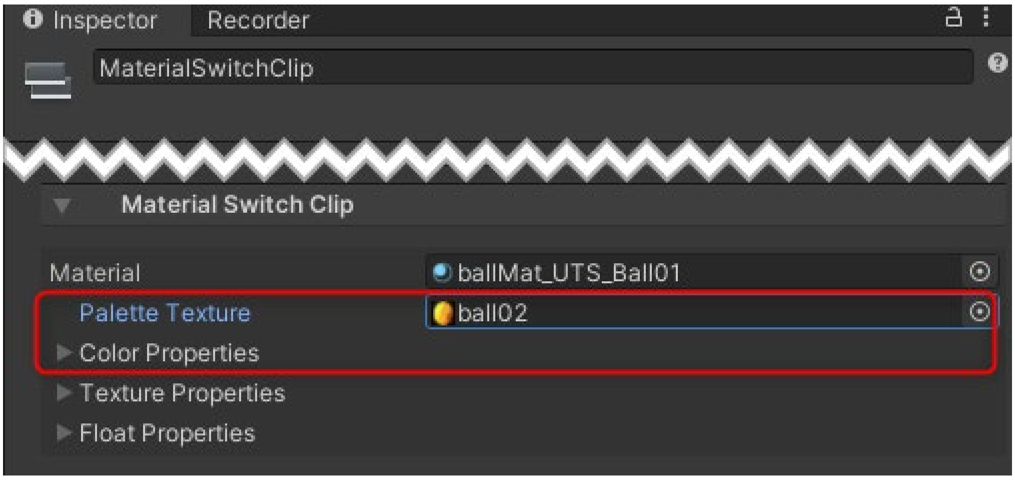
Advanced

Non-Power of 2: ToNearest

Read/Write Enabled: ✓

Streaming Mipmaps: □

7. Select the image imported in the step 6 for Pallet Texture of Material Switch Clip. Color Properties appear.



MaterialSwitchClip

Material Switch Clip

Material: ballMat_UTS_Ball01

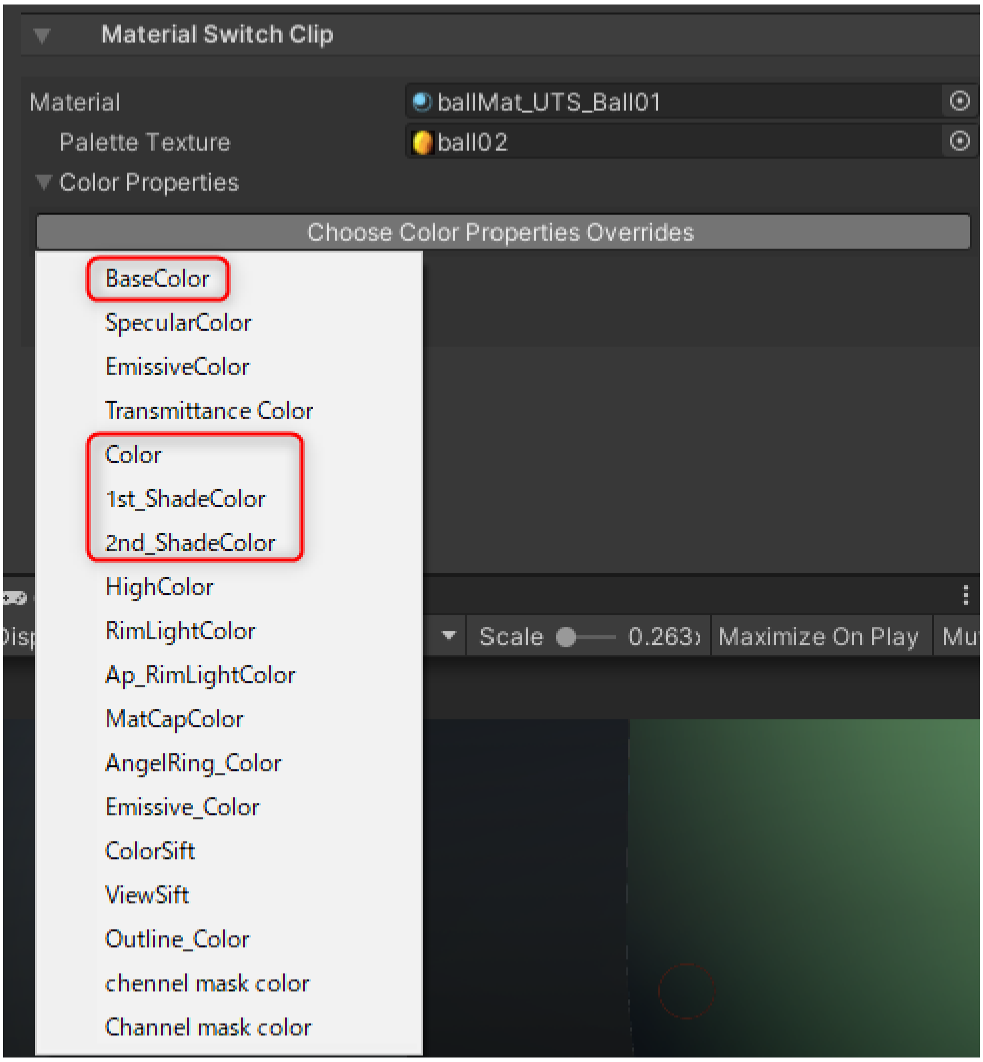
Palette Texture: ball02

Color Properties

Texture Properties

Float Properties

8. Expand Color Properties to display “Choose Color Properties Overrides” button. Select color properties to be changed. In the case of UTS/HDRP;
   - Base color ⇒ set both BaseColor and Color
   - 1st shade color ⇒ set 1st_ShadeColor
   - 2nd shade color ⇒ set 2nd_ShadeColor



Material Switch Clip

Material: ballMat_UTS_Ball01

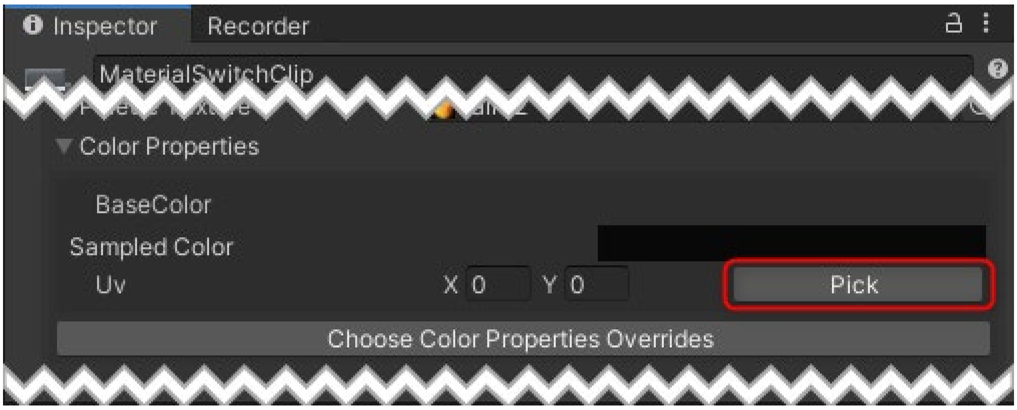
Palette Texture: ball02

Color Properties

Choose Color Properties Overrides

- BaseColor
- SpecularColor
- EmissiveColor
- Transmittance Color
- Color
- 1st_ShadeColor
- 2nd_ShadeColor
- HighColor
- RimLightColor
- Ap_RimLightColor
- MatCapColor
- AngelRing_Color
- Emissive_Color
- ColorSift
- ViewSift
- Outline_Color
- chennel mask color
- Channel mask color

9. First, let’s try with BaseColor. When selecting BaseColor, you will see the following screen. Click Pick button.



MaterialSwitchClip

Color Properties

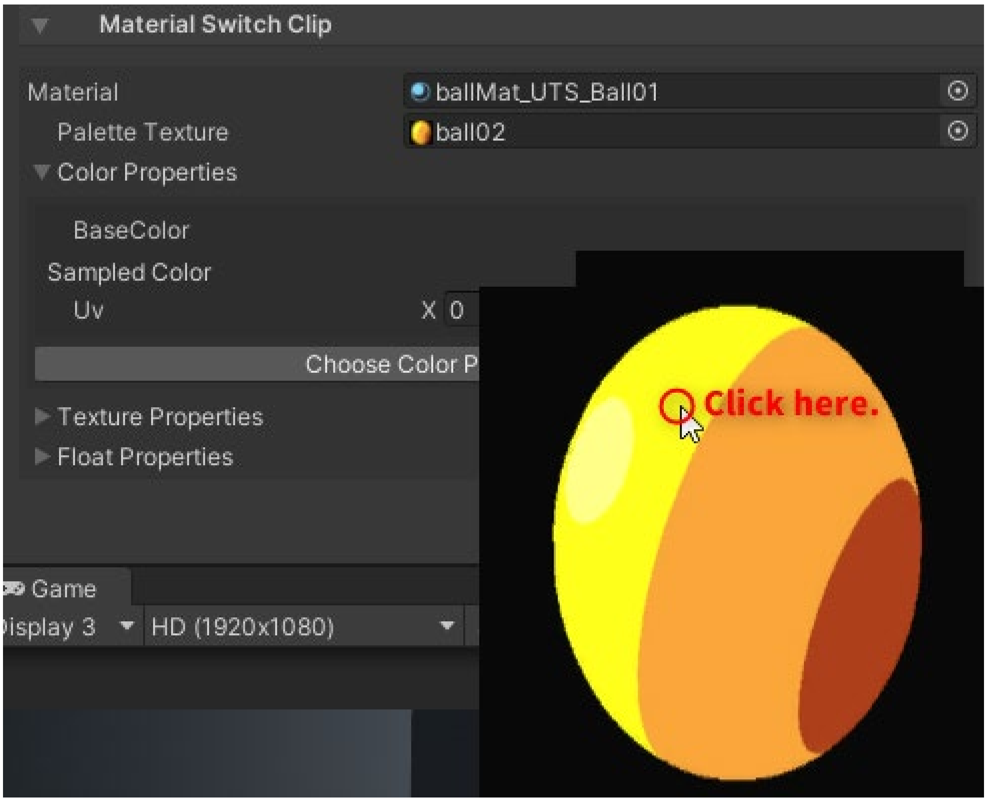
BaseColor

Sampled Color

Uv X 0 Y 0 Pick

Choose Color Properties Overrides

10. When clicking Pick button, the image set in Palette Texture opens. Click a point you want to specify as the base color with the cursor.



Material Switch Clip

Material: ballMat_UTS_Ball01

Palette Texture: ball02

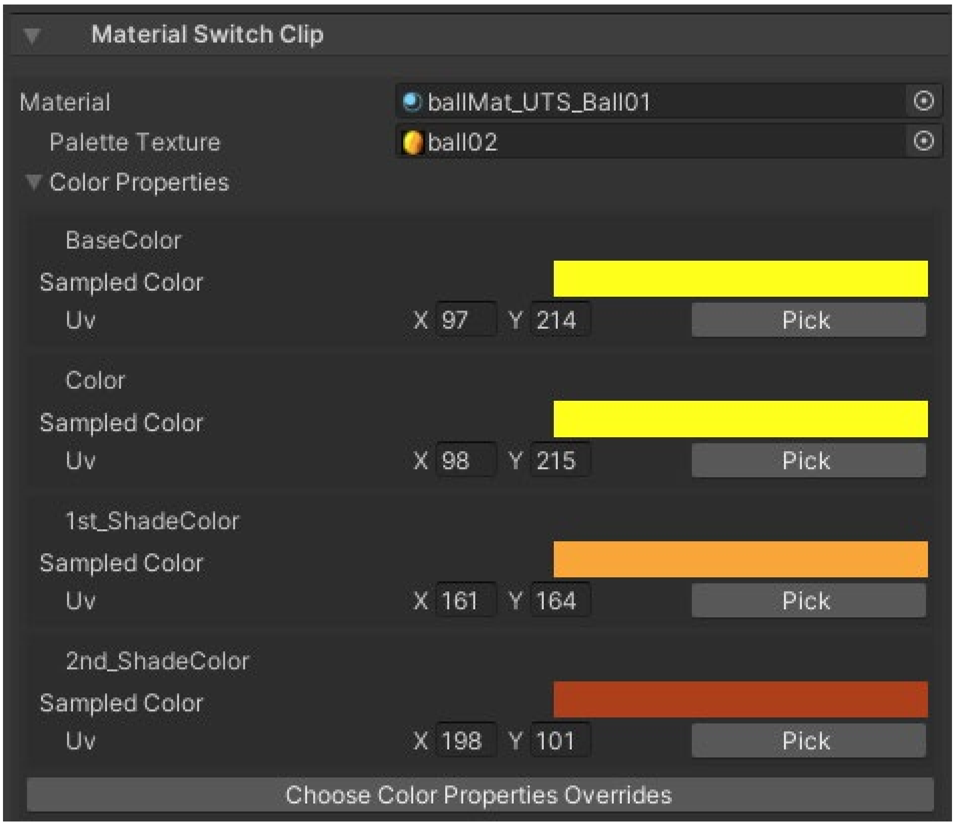
▼ Color Properties

BaseColor

Sampled Color

Uv X 0

11. After clicking, the image automatically closes and X and Y values of UV are filled. The values were taken from the point you clicked in the image. You can set Color, 1st_ShadeColor, and 2nd_ShadeColor, as well.



Material Switch Clip

Material: ballMat_UTS_Ball01

Palette Texture: ball02

▼ Color Properties

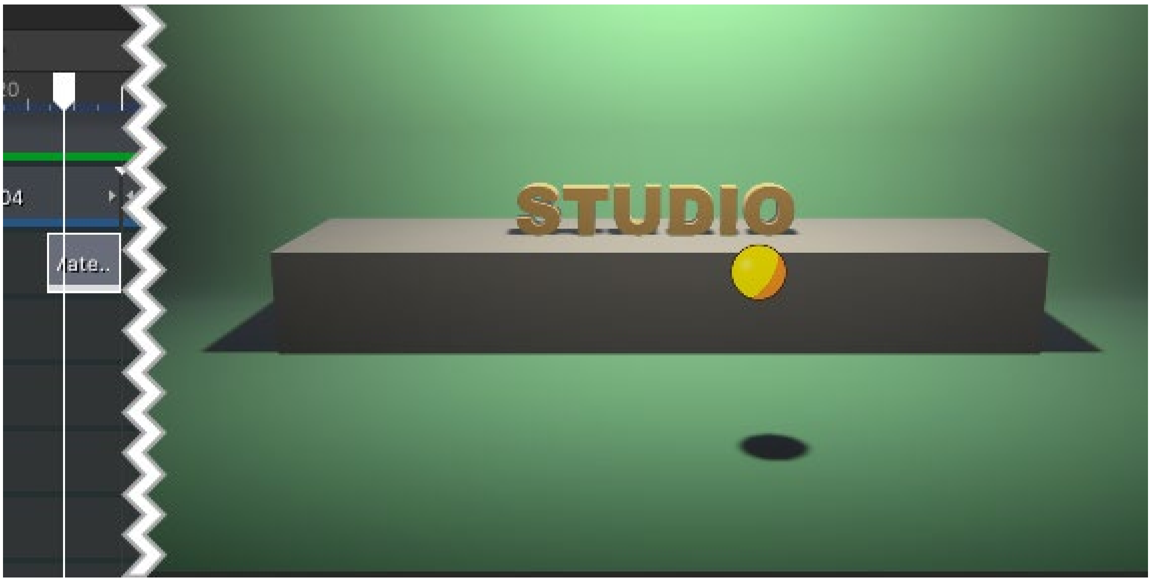
BaseColor
Sampled Color
Uv X 97 Y 214

Color
Sampled Color
Uv X 98 Y 215

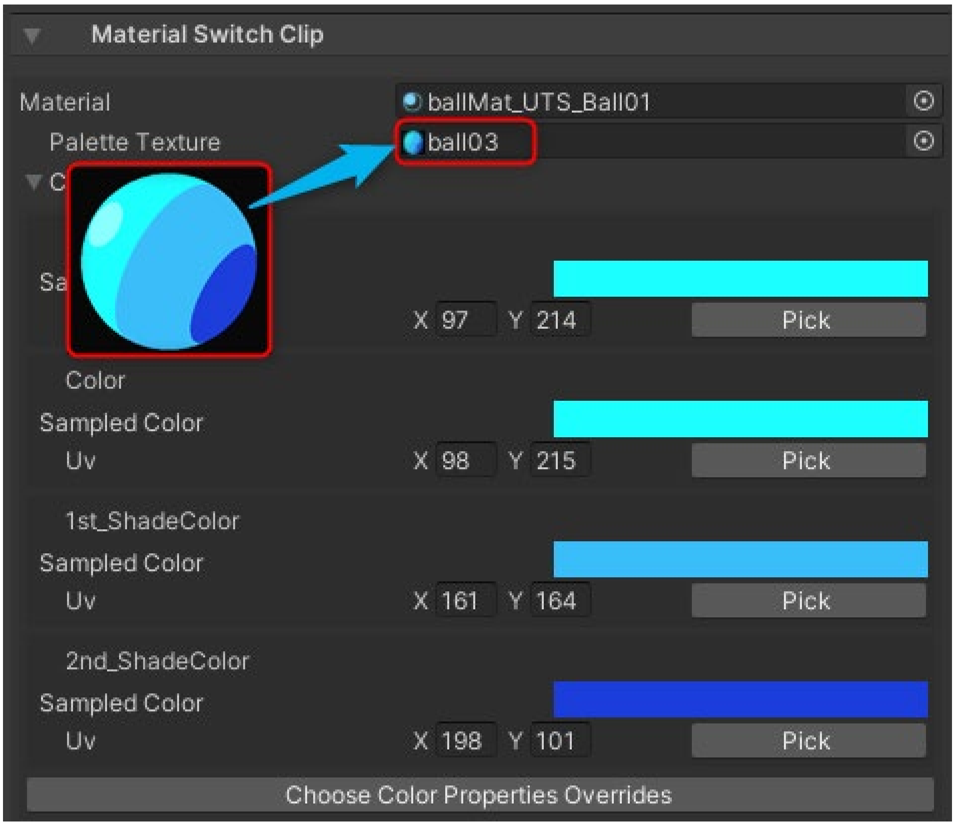
1st_ShadeColor
Sampled Color
Uv X 161 Y 164

2nd_ShadeColor
Sampled Color
Uv X 198 Y 101

12. When the playhead approaches the position of the Material Switch Clip we set above, you can see that the material color is switched to yellow.



13. Duplicate the Material Switch Clip set with yellow and move the clip position. Now let's change the duplicated Material Switch Clip to blue. You prepare a blue Pallet Texture as you did before and replace the texture, the color will be replaced immediately as long as the UV points are the same.



Material Switch Clip

Material ballMat_UTS_Ball01

Palette Texture ball03

C

Sa

X 97 Y 214 Pick

Color

Sampled Color

Uv X 98 Y 215 Pick

1st_ShadeColor

Sampled Color

Uv X 161 Y 164 Pick

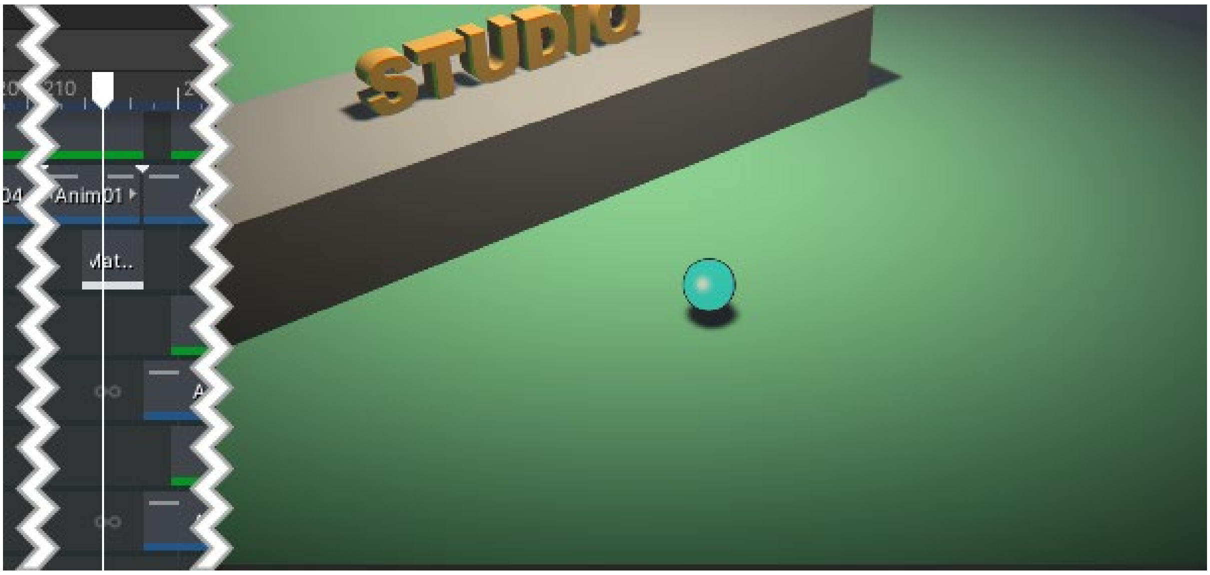
2nd_ShadeColor

Sampled Color

Uv X 198 Y 101 Pick

Choose Color Properties Overrides

Note: **In order to update the color sample display, you need to manually change the display of the Inspector, for example, by re-selecting another track.** However, the internal reference will switch when you switch the Pallet Texture, therefore, it has been already updated in the game view.



With Material Switch Track, in addition to switching the color pallet with color settings of each image as described above, you can also switch the following in Timeline; various maps and textures set in materials, and float values that set a color area.

Material Switch Track allows you to control character’s color design in detail for each scene without changing the material design, which the character is based on.

Tips: Since the shadow control map of UTS/HDRP can also be changed with the Material Switch Track, you can freely control the shade color from the material side.

# Using Unity Recorder for Capture at Accurate Frame Rate

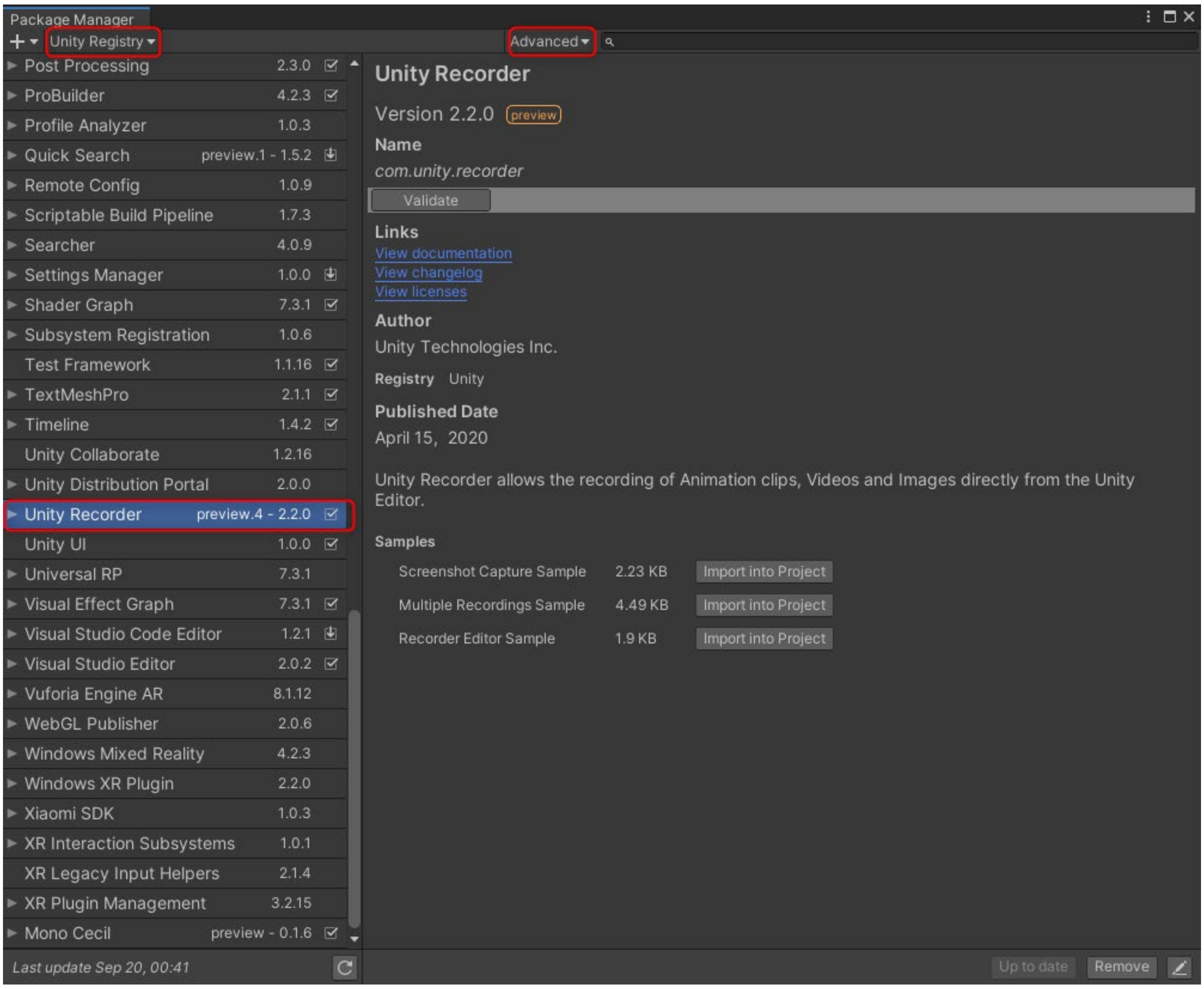
By recording real-time images displayed in the game view with Unity Recorder, it becomes possible to render and output the composite results and the audio data edited in Timeline to serial number images, movies, or audio files.

Here, we will explain how to set up Unity Recorder to capture images at an accurate frame rate and how to operate it to get enough warm-up time when actually capturing.

## Document of Unity Recorder

https://docs.unity3d.com/Packages/com.unity.recorder@2.2/manual/index.html

You can find the document of Unity Recorder in Package Manager window with the following settings.
In the case of DefaultWorkbentch Project, it is installed in default.



| | |
|---|---|
| Post Processing | 2.3.0 ☑ |
| ProBuilder | 4.2.3 ☑ |
| Profile Analyzer | 1.0.3 |
| Quick Search | preview.1 - 1.5.2 |
| Remote Config | 1.0.9 |
| Scriptable Build Pipeline | 1.7.3 |
| Searcher | 4.0.9 |
| Settings Manager | 1.0.0 |
| Shader Graph | 7.3.1 ☑ |
| Subsystem Registration | 1.0.6 |
| Test Framework | 1.1.16 ☑ |
| TextMeshPro | 2.1.1 ☑ |
| Timeline | 1.4.2 ☑ |
| Unity Collaborate | 1.2.16 |
| Unity Distribution Portal | 2.0.0 |
| Unity Recorder | preview.4 - 2.2.0 ☑ |
| Unity UI | 1.0.0 ☑ |
| Universal RP | 7.3.1 |
| Visual Effect Graph | 7.3.1 ☑ |
| Visual Studio Code Editor | 1.2.1 |
| Visual Studio Editor | 2.0.2 ☑ |
| Vuforia Engine AR | 8.1.12 |
| WebGL Publisher | 2.0.6 |
| Windows Mixed Reality | 4.2.3 |
| Windows XR Plugin | 2.2.0 |
| Xiaomi SDK | 1.0.3 |
| XR Interaction Subsystems | 1.0.1 |
| XR Legacy Input Helpers | 2.1.4 |
| XR Plugin Management | 3.2.15 |
| Mono Cecil | preview - 0.1.6 ☑ |

Last update Sep 20, 00:41

Unity Recorder

Version 2.2.0 preview

Name
com.unity.recorder

Validate

Links
View documentation
View changelog
View licenses

Author
Unity Technologies Inc.

Registry Unity

Published Date
April 15, 2020

Unity Recorder allows the recording of Animation clips, Videos and Images directly from the Unity Editor.

Samples

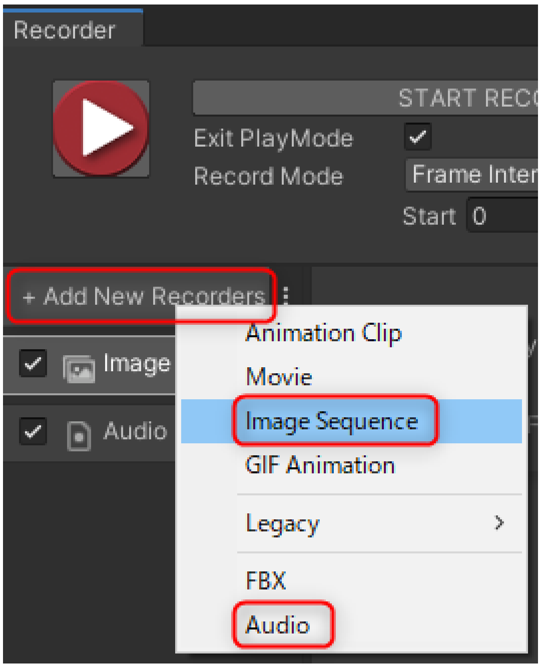
| | | |
|---|---|---|
| Screenshot Capture Sample | 2.23 KB | Import into Project |
| Multiple Recordings Sample | 4.49 KB | Import into Project |
| Recorder Editor Sample | 1.9 KB | Import into Project |

## Setting up Unity Recorder

To operate Unity Recorder, from the main menu, go to Window > General > Recorder > Recorder Windows. Recorder window will open.

### Adding a target to capture

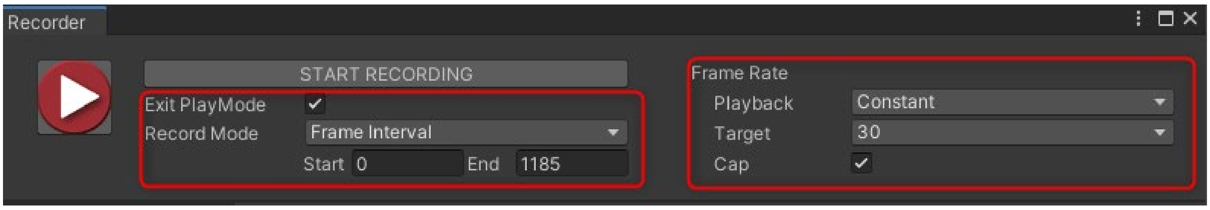
In Recorder window, click "+ Add New Recorders" to select a target to capture from the menu.
Here, we will add Image Sequence and Audio.



Exit PlayMode ✓
Record Mode Frame Inter
Start 0

### Common settings

First, we set up common recording settings. Please refer the image below.



Exit PlayMode ✓
Record Mode Frame Interval
Start 0 End 1185

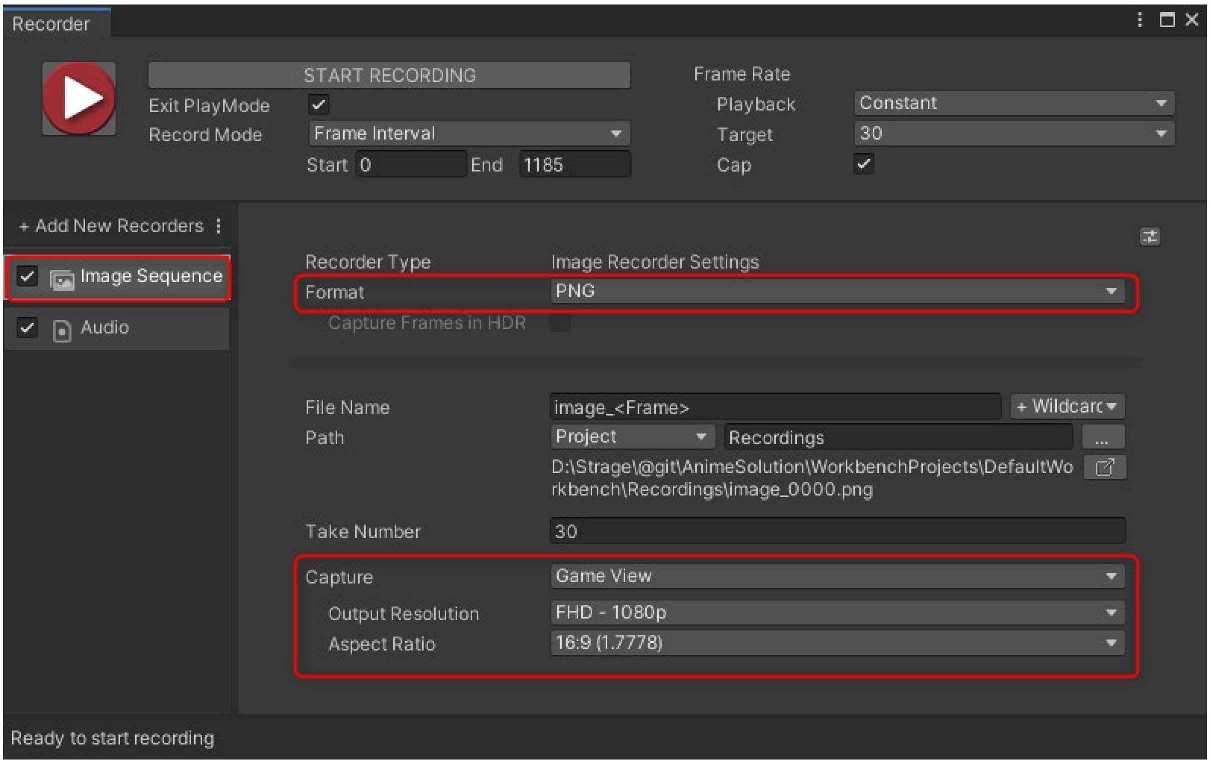
Frame Rate
Playback Constant
Target 30
Cap ✓

- Record Mode ⇒ Set it to "**Frame Interval" and** enter a frame number at the start and end points of Timeline to Start and End fields, respectively.
- Frame Rate ⇒ Set Playback to "**Constant" and** Target to the frame rate set in Timeline. In this example, set to "30."

**Tips: To see frame rate of Timeline, from the gear icon, go to Frame Rate > [frame to check]**

## Setting of Image Sequence

Image Sequence should be set as follows.



Exit PlayMode ☑
Record Mode: Frame Interval
Start 0 End 1185

Frame Rate
Playback: Constant
Target: 30
Cap ☑

☑ Image Sequence
☑ Audio

Recorder Type: Image Recorder Settings
Format: PNG
Capture Frames in HDR □

File Name: image_<Frame>
Path: Project / Recordings
D:\Strage\@git\AnimeSolution\WorkbenchProjects\DefaultWorkbench\Recordings\image_0000.png

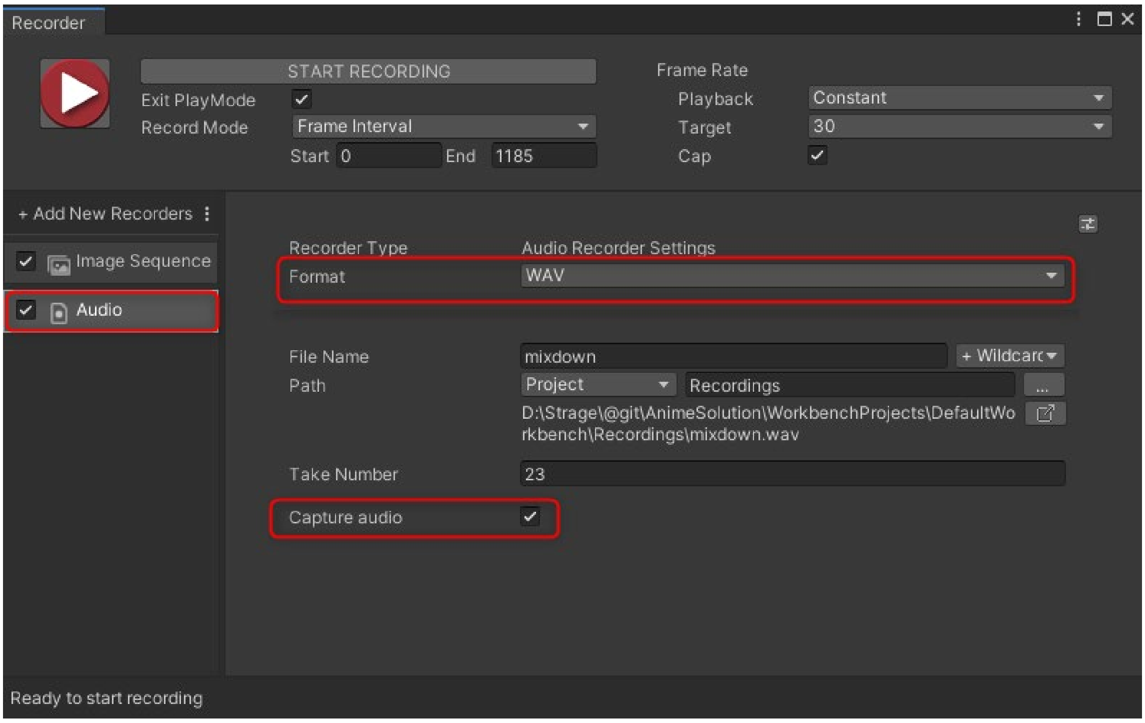
Take Number: 30

Capture: Game View
Output Resolution: FHD - 1080p
Aspect Ratio: 16:9 (1.7778)

- Format ⇒ Set it to “PNG” for a SDR image. Set to “EXR” for a HDR image.
- Capture ⇒ Set it to “Game View”
  Make sure that Game View is set to “Display 8.”
- Output Resolution and Aspect Ratio ⇒ Set them to appropriate values for the final output size.

## Setting of Audio

Set up Audio as follows.



Recorder

START RECORDING

Exit PlayMode ☑

Record Mode: Frame Interval

Start 0 End 1185

Frame Rate

Playback: Constant

Target: 30

Cap ☑

+ Add New Recorders

☑ Image Sequence

☑ Audio

Recorder Type: Audio Recorder Settings

Format: WAV

File Name: mixdown

Path: Project / Recordings

D:\Strage\@git\AnimeSolution\WorkbenchProjects\DefaultWorkbench\Recordings\mixdown.wav

Take Number: 23

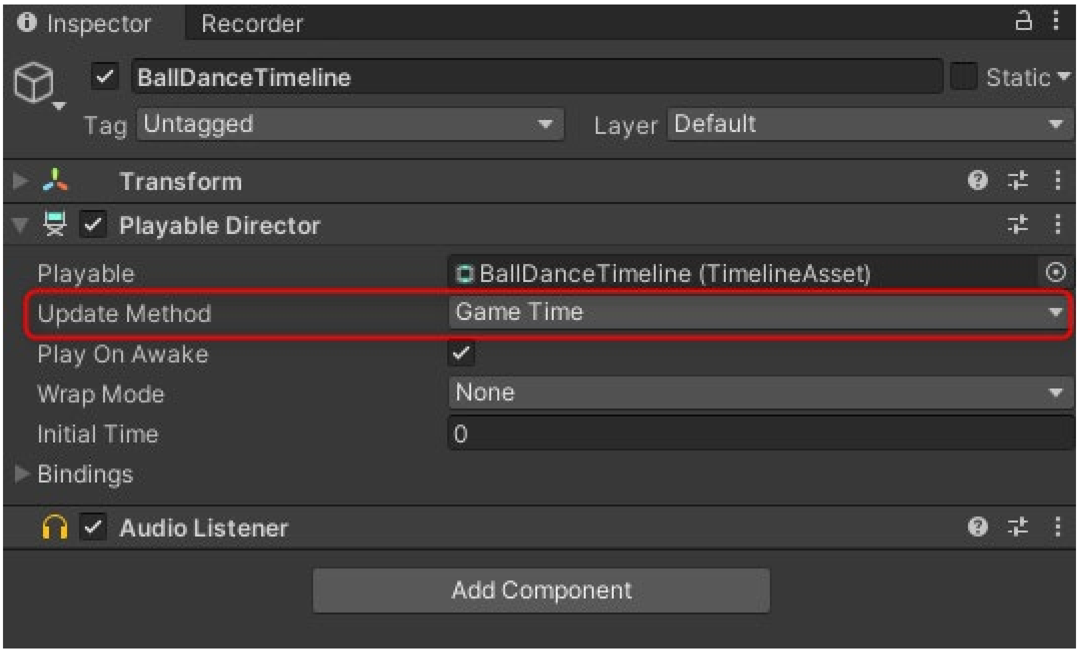
Capture audio ☑

Ready to start recording

- Format ⇒ Set it to “WAV”
- Capture audio ⇒ Check it

## Checking Settings of Playable Director of Timeline

Select a Timeline to capture from Hierarchy window and confirm that the Update Method of Playable Director in the Inspector is "Game Time".



Inspector | Recorder

☑ BallDanceTimeline □ Static

Tag: Untagged Layer: Default

Transform

☑ Playable Director

Playable: BallDanceTimeline (TimelineAsset)

Update Method: Game Time

Play On Awake ☑

Wrap Mode: None

Initial Time: 0

Bindings

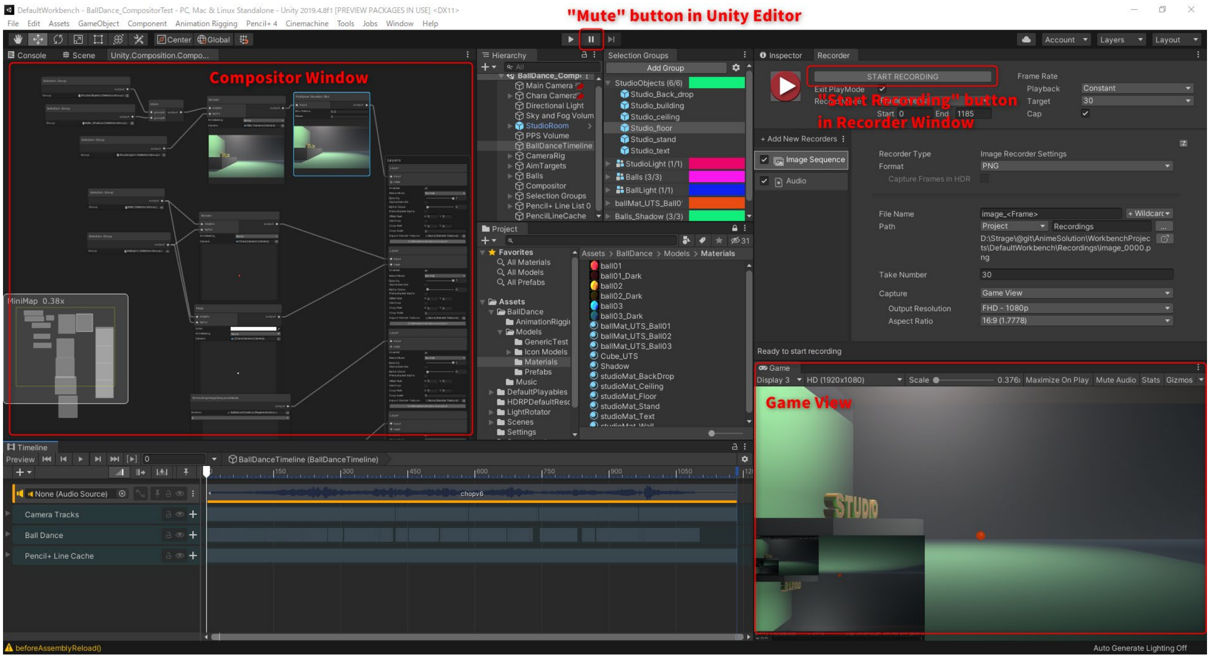
☑ Audio Listener

Add Component

## Steps of Capture Considering Warm-up

Unity, originally a game engine, loads assets into memory as needed. Unity is designed not to suffer users' gameplay experience by changing the frame rate according to the machine specification or the weight of process. In the case of video, capturing a screen in a frame which is updated with a fixed rate, it is better to load all the assets into the memory in advance. In this way, you can prevent unnecessary mistakes.

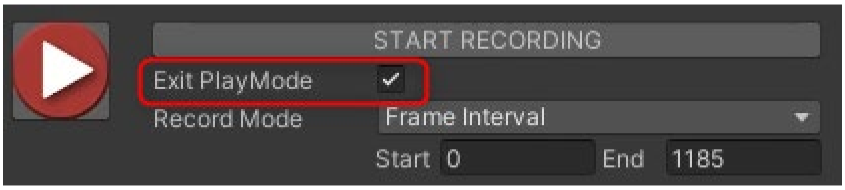
Here, we describe how to use Unity Recorder when capturing images at a fixedframe rate for video. This method provides sufficient warm-up time to accurately reflect the timing design, including synchronization with the audio track in Timeline.


"Mute" button in Unity Editor
Compositor Window
"Start Recording" button in Recorder Window
Game View

START RECORDING
Exit PlayMode ✓
Record Mode Frame Interval
Start 0 End 1185
Frame Rate
Playback Constant
Target 30
Cap ✓
Image Sequence ✓
Audio ✓
Recorder Type Image Recorder Settings
Format PNG
Capture Frames in HDR □
File Name image_<Frame>
Path Project Recordings
D:\Strage\@git\AnimeSolution\WorkbenchProjects\DefaultWorkbench\Recordings\image_0000.png
Take Number 30
Capture Game View
Output Resolution FHD - 1080p
Aspect Ratio 16:9 (1.7778)
Ready to start recording

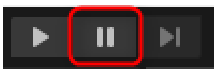
Open the views required to check the capture status, such as Unity Recorder, Visual Compositor, and the game view, and follow the steps below.

1. Confirm "Exit PlayMode" is turned on in Recorder window.

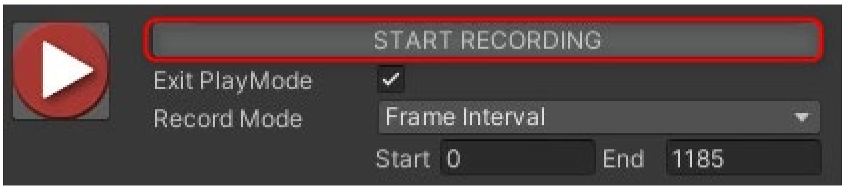


START RECORDING
Exit PlayMode ✓
Record Mode Frame Interval
Start 0 End 1185

2. Click "Pause" button in Unity editor. The editor becomes mute.

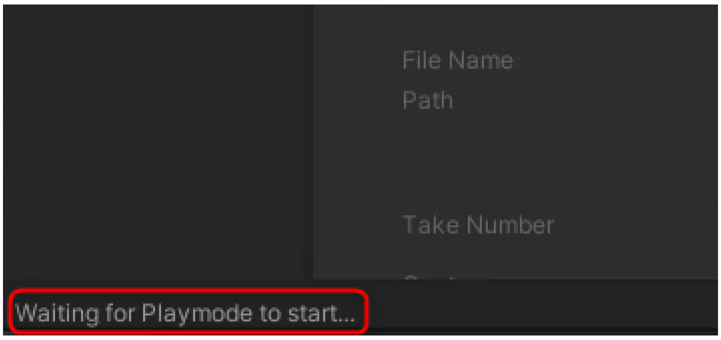


3. Push “**Start Recording**“ button in Recorder window.



Exit PlayMode ✓
Record Mode Frame Interval
Start 0 End 1185

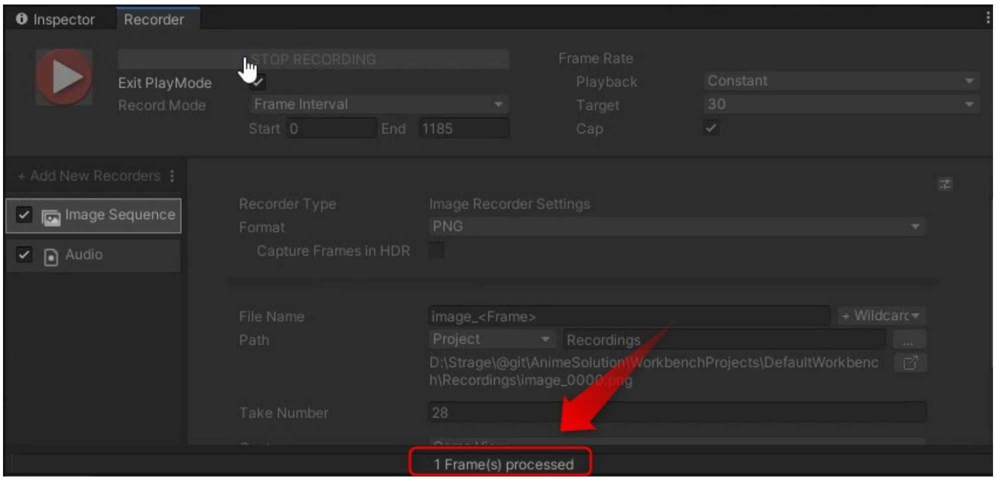
⇒ Capturing is paused because the pause button in the editor has been pressed.
The message "Waiting for Playmode to start ..." is displayed in the lower left corner of the Recorder window.。



Waiting for Playmode to start...

4. **Visual Compositor window or the game view is updated during the pause.**
(In many cases, the preview may momentarily black out and switch to the 1$^{st}$ frame.) During this time, required assets are loaded. This time, which is about one to several seconds at the longest, is called warm-up time. During the warm-up time, Unity will load all large assets such as voice assets into memory.

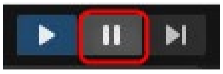
When the warm-up time is over, you will see a message "1 Frame(s) processed" at the bottom of the Recorder window. You should check the recording of the 1st frame has been finished.



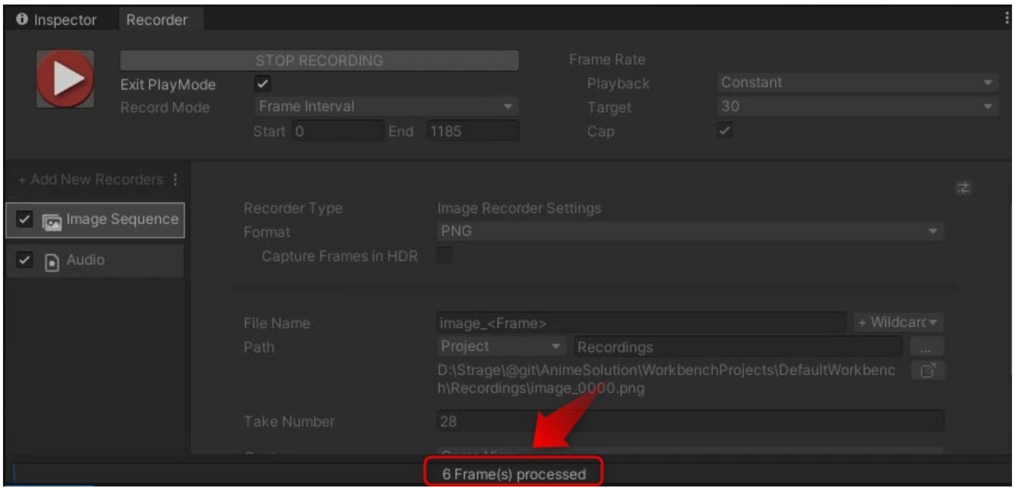
Exit PlayMode ✓
Record Mode Frame Interval
Start 0 End 1185
Frame Rate
Playback Constant
Target 30
Cap ✓
✓ Image Sequence
✓ Audio
Recorder Type Image Recorder Settings
Format PNG
Capture Frames in HDR □
File Name image_<Frame>
Path Project Recordings
D:\Strage\@git\AnimeSolution\WorkbenchProjects\DefaultWorkbench\Recordings\image_0000.png
Take Number 28

1 Frame(s) processed

5. Click the "Pause" button in Unity Editor again to release the paused Recorder. Recording will start。



You can see the "Frame(s) processed" message in Recorder window changes.



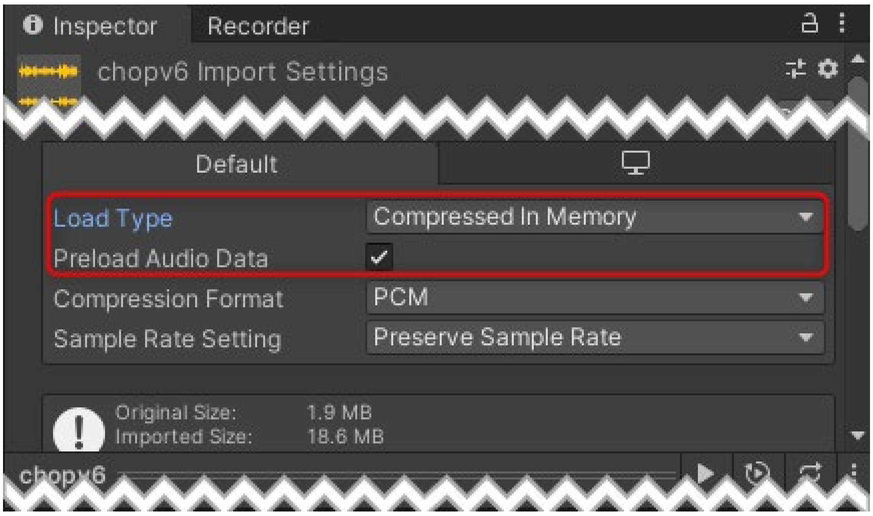
Exit PlayMode ✓
Record Mode Frame Interval
Start 0 End 1185
Frame Rate
Playback Constant
Target 30
Cap ✓
Recorder Type Image Recorder Settings
Format PNG
Capture Frames in HDR □
File Name image_<Frame>
Path Project Recordings
D:\Strage\@git\AnimeSolution\WorkbenchProjects\DefaultWorkbench\Recordings\image_0000.png
Take Number 28

6 Frame(s) processed

Even if the capture starts successfully, the position of the Timeline playhead may not be updated from the initial position for a while, but it will start moving soon. You don't need to worry.

6. **If Exit Play Mode is turned on, play mode will also stop as soon as the recording of capture images is complete. That is all for capture**.

**Tips: In order to load the audio asset into the memory with the previously described way, it is recommended to set audio assets, which are added to audio clips in Timeline, as shown below.**



Load Type Compressed In Memory
Preload Audio Data ✓
Compression Format PCM
Sample Rate Setting Preserve Sample Rate

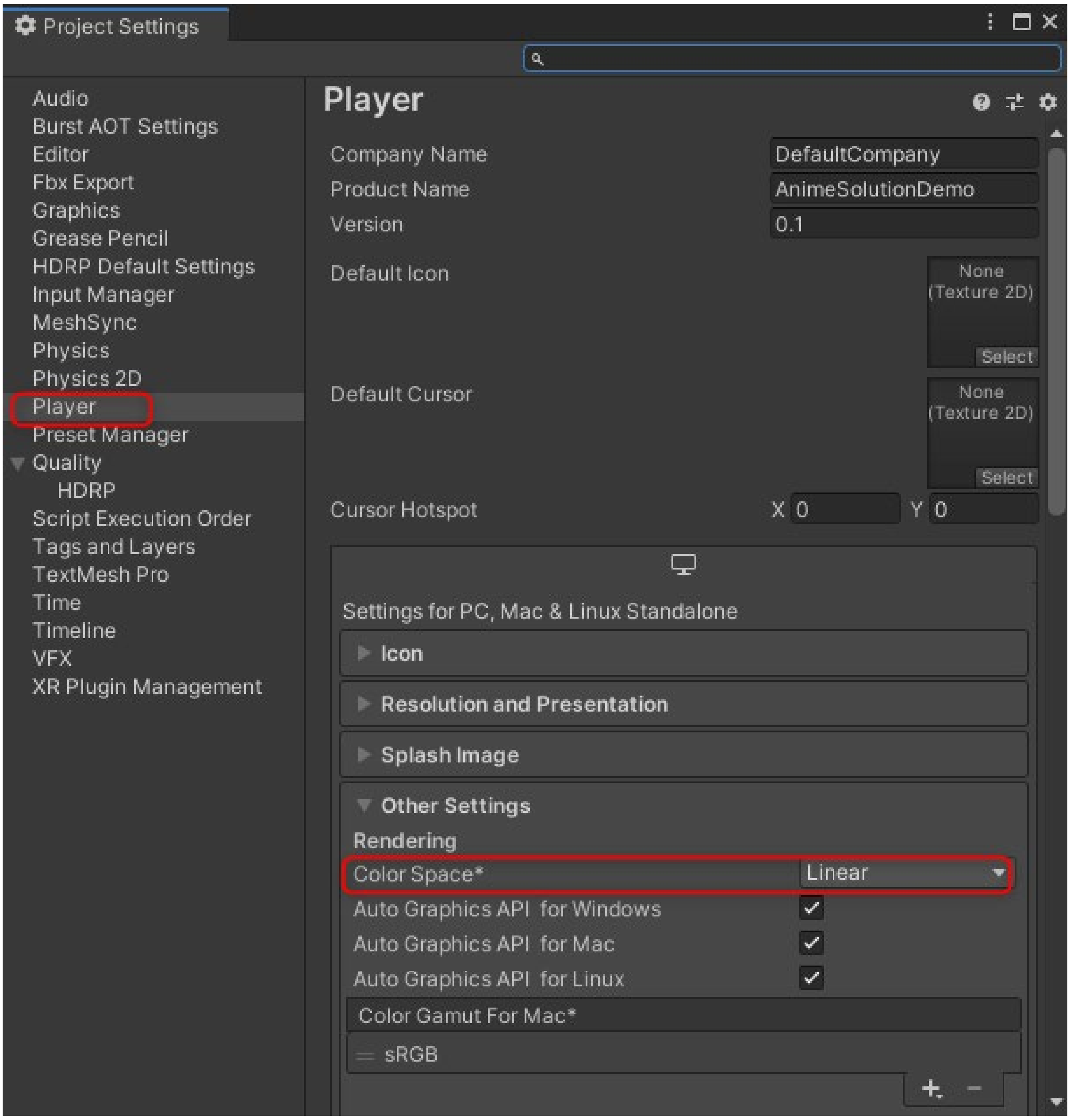
Original Size: 1.9 MB
Imported Size: 18.6 MB

# Reference: About Color Space of Project

When you start a project with HDRP like this project, the color space is set to "linear setting" from the beginning, so you do not need to think about it. However, if you work on the video editing as described in this document with a legacy pipeline like Unity's traditional built-in pipeline, be sure to check the default color space of your project.
In many cases, you can get a better result to edit video in linear space than in gamma space.

To check the color space, go to Editor > Project Settings from the main menu, open the Project Settings window, and find it in the Player page.

- In the case of HDRP (Unity 2019.4.x)



Project Settings

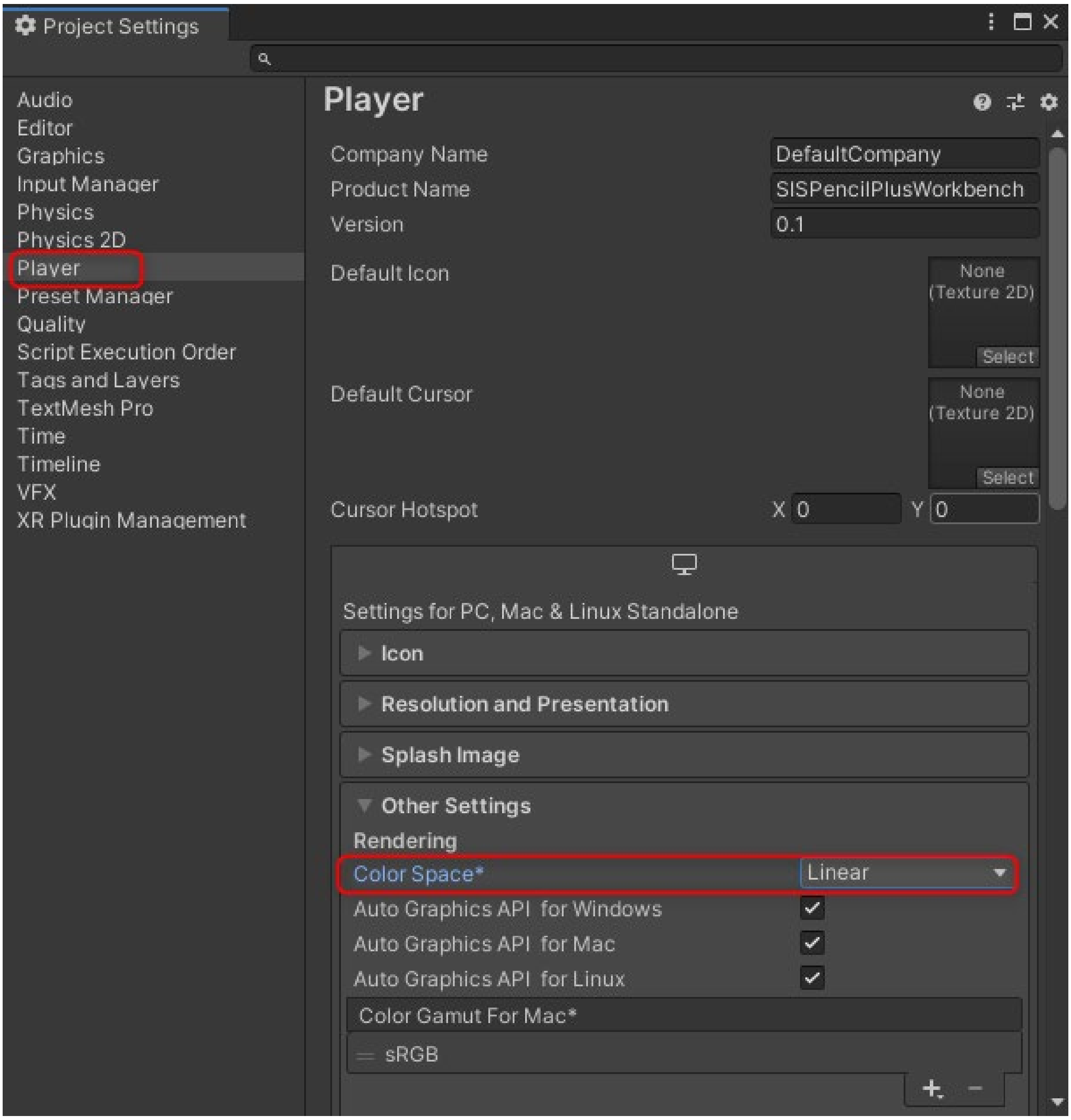
Audio
Burst AOT Settings
Editor
Fbx Export
Graphics
Grease Pencil
HDRP Default Settings
Input Manager
MeshSync
Physics
Physics 2D
Player
Preset Manager
Quality
HDRP
Script Execution Order
Tags and Layers
TextMesh Pro
Time
Timeline
VFX
XR Plugin Management

Player

Company Name: DefaultCompany
Product Name: AnimeSolutionDemo
Version: 0.1
Default Icon: None (Texture 2D) Select
Default Cursor: None (Texture 2D) Select
Cursor Hotspot: X 0 Y 0

Settings for PC, Mac & Linux Standalone
▶ Icon
▶ Resolution and Presentation
▶ Splash Image
▼ Other Settings
Rendering
Color Space*: Linear
Auto Graphics API for Windows ✓
Auto Graphics API for Mac ✓
Auto Graphics API for Linux ✓
Color Gamut For Mac*
sRGB

- In the case of traditional built-in pipeline (Unity 2019.4.x)



Player

Company Name: DefaultCompany

Product Name: SISPencilPlusWorkbench

Version: 0.1

Default Icon: None (Texture 2D) Select

Default Cursor: None (Texture 2D) Select

Cursor Hotspot: X 0 Y 0

Settings for PC, Mac & Linux Standalone

Rendering

Color Space*: Linear

Auto Graphics API for Windows ✓

Auto Graphics API for Mac ✓

Auto Graphics API for Linux ✓

Color Gamut For Mac*

sRGB