Game Animation
Game Programming

- Rendering
- Looping and control
- Math
- Animation
- Physics
- Behaviour and navigation (AI)
- Effects
- Networking
Game Programming

- Rendering
- Looping and control
- Math
- **Animation**
- Physics
- Behaviour and navigation (AI)
- Effects
- Networking
2D Game Kit
Zero code game design

FREE
Add to My Assets

License
Extension Asset
File size
299.9 MB
Latest version
1.9.3
Latest release date
Jul 21, 2020
Supported Unity versions
2017.3.0 or higher
Support
Visit site
Demo : 2D Game Kit

- Play
- Introduce scene assets
2D sprite animation

- Keyframes / image sequence
Sprite renderer

- Sprite: Anim_IDLE_00001
- Color:
- Flip: Off, Off
- Draw Mode: Simple
- Mask Interaction: None
- Sprite Sort Point: Center
- Material:Sprites-Default
- Additional Settings:
  - Sorting Layer: Default
  - Order in Layer: 2
Demo: 2D Game Kit

- Inspect Ellen [Prefab]
Animation clip: keyframe animation

![Diagram showing animation clip and keyframes]
Demo: 2D Game Kit

- Inspect Ellen prefab
- Inspect Ellen_Idle animation
## Sprite Editor

### Sprite Editor Interface

- **Sprite Editor**
- **Slice**
- **Trim**

### Sprite Settings

<table>
<thead>
<tr>
<th>Setting</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Player_SpriteSheet_0</td>
</tr>
<tr>
<td>Position</td>
<td>X 0, Y 448, W 64, H 64</td>
</tr>
<tr>
<td>Border</td>
<td>L 0, T 0, R 0, B 0</td>
</tr>
<tr>
<td>Pivot</td>
<td>Bottom</td>
</tr>
<tr>
<td>Custom Pivot</td>
<td>X 0.5, Y 0</td>
</tr>
</tbody>
</table>

### Source

*Source: Unity Learn*
Tight?
Root motion
Root motion
Root motion

<table>
<thead>
<tr>
<th>Sprite Editor</th>
<th>Slice</th>
<th>Trim</th>
<th>Revert</th>
<th>Apply</th>
</tr>
</thead>
</table>

**Sprite**

<table>
<thead>
<tr>
<th>Name</th>
<th>Player_SpriteSheet_0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position</td>
<td></td>
</tr>
<tr>
<td>X</td>
<td>0</td>
</tr>
<tr>
<td>Y</td>
<td>448</td>
</tr>
<tr>
<td>W</td>
<td>64</td>
</tr>
<tr>
<td>H</td>
<td>64</td>
</tr>
<tr>
<td>Border</td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>0</td>
</tr>
<tr>
<td>T</td>
<td>0</td>
</tr>
<tr>
<td>R</td>
<td>0</td>
</tr>
<tr>
<td>B</td>
<td>0</td>
</tr>
<tr>
<td>Pivot</td>
<td>Bottom</td>
</tr>
<tr>
<td>Custom Pivot</td>
<td></td>
</tr>
<tr>
<td>X</td>
<td>0.5</td>
</tr>
<tr>
<td>Y</td>
<td>0</td>
</tr>
</tbody>
</table>
Root motion
Demo: 2D Game Kit

- Inspect Ellen prefab
- Inspect Ellen_IDLE animation
Parametric animation
Parametric animation

Interpolation between consecutive frames?
Parametric animation

Smooth or sharp?
Animation clip: Curves

- (DEMO) Ellen [Prefab] > Animation > Ellen_IdleWithGun
State machine

- Current state?
- On state enter/leave?
State machine

- Current state?
- On state enter/leave?
- Transition between states?
Animation state machine

- Current animation?
- On animation play / finished?
- Blending between animations?
Animator controller

Controller: Ellen
Avatar: None (Avatar)
Apply Root Motion: Off
Update Mode: Animate Physics
Culling Mode: Always Animate

Clip Count: 51
Curves Pos: 19 Quat: 0 Euler: 0 Scale: 0 Muscles: 0 Generic: 98 PPtr: 51
Curves Count: 206 Constant: 137 (66.5%) Dense: 0 (0.0%) Stream: 69 (33.5%)
Animator states
(DEMO)
Animator transition
BlendTree
Join GitHub today

GitHub is home to over 50 million developers working together to host and review code, manage projects, and build software together.

Sign up

About

Samples for 2D Animation features.
Inverse kinematics (IK)
Inverse kinematics (IK)

Set desired target
Inverse kinematics (IK)

Set desired target

Always possible?
(DEMO)
3D Model
Skinned mesh renderer

Bounds
Center: X 0.00682 Y 0 Z -0.0092
Extent: X 0.87855 Y 0.76564 Z 0.25213

Quality: Auto
Update When Offscreen
Mesh: Ellen_Body
Root Bone: Ellen_Hips (Transform)

Materials
Size: 6
Element 0: Ellen_Hair_Mat
Element 1: Ellen_Body_Mat
Element 2: Ellen_Head_Mat
Element 3: Ellen_Eyes_Mat
Element 4: Ellen_Tear_Mat
Rig

Animation Type: Humanoid

Avatar Definition: Create From This Model

Skin Weights: Standard (4 Bones)

Optimize Game Objects: [ ]

A Biped was detected, but cannot be configured properly because of an unsupported hierarchy. Adjust Biped settings in 3DS Max before exporting to correct this problem.

- Invalid parent for Bip01 Head. Expected Bip01 Neck, but found Bip01 Neck2. Preferred is one Neck Links
(DEMO)
(DEMO)
Root motion
(DEMO)
3D Game Kit

FREE

Add to My Assets

License
File size
Latest version
Latest release date
Supported Unity versions
Support

Extension Asset
2.2 GB
1.9.2
Jul 22, 2020
2018.1.0 or higher
Visit site
(DEMO)
BlendTree
Retargeting
Retargeting and Reusing Animation

Project • Beginner • +300 XP • 2 Hours 40 Mins • ¥402

Overview

Summary

This project will take us through the process of reusing Animation Clips, both imported and custom. We will also review how to retarget imported animation clips on different biped, humanoid models.

One of the most powerful features of Mecanim is retargeting of humanoid animations. This means that with relative ease, you can apply the same set of animations to various character models. Keep in mind that retargeting is only possible for humanoid models, where an Avatar has been configured, because this gives us a correspondence between the models' bone structure.
(DEMO)
Inverse kinematics (IK)
Inverse kinematics (IK)
Inverse kinematics (IK)
Q & A
References

● Getting Started with Unity’s 2D Animation Package @ Unity Blog

● Mecanim Humanoids @ Unity Blog
  ○ https://blogs.unity3d.com/2014/05/26/mecanim-humanoids/

● Retargeting and Reusing Animation @ Unity Blog
  ○ https://learn.unity.com/project/re-targeting-and-re-using-animation