Custom Pointer README

The Custom Pointer asset allows one to turn any Transform or Leap Motion finger into a pointer that can interact with the UI. This works with 3D Transforms and Rect Transforms by using their screen position. Although you can make your custom pointer click by whatever means you like, hover-click is a built-in feature that shows a progress bar that indicates time-till-click, similar to those found in Kinect games, and the hover-drag feature works similarly.

In addition to allowing any Transform to become a pointer, this asset includes classes that work specifically with the Leap Motion, supporting both Core Assets v2.3.1 which supports macOS, Linux, and Windows, and Core Assets v4.4.0 which supports Windows only and VR. Scenes using Unity's UI will allow a Leap Motion hand to click using hover click, screen tap, or key tap without any changes to the UI.

Note

Leap Motion Orion is still in beta and is changing rapidly. Custom Pointer supports Leap Motion Core Assets v4.1.5. There does appear to be an issue that can be seen in "Custom Pointer Leap Motion Orion Demo." RegisterHandTransition is not always called faithfully so far as I can tell. This often leads to the first time the hand is shown, it won't have a working pointer. If you remove your hand from the sensor, and put it back in, it will work. I believe this issue will be fixed by Leap Motion in subsequent releases of their software.

Requirements

- Unity v5.6.5 or later

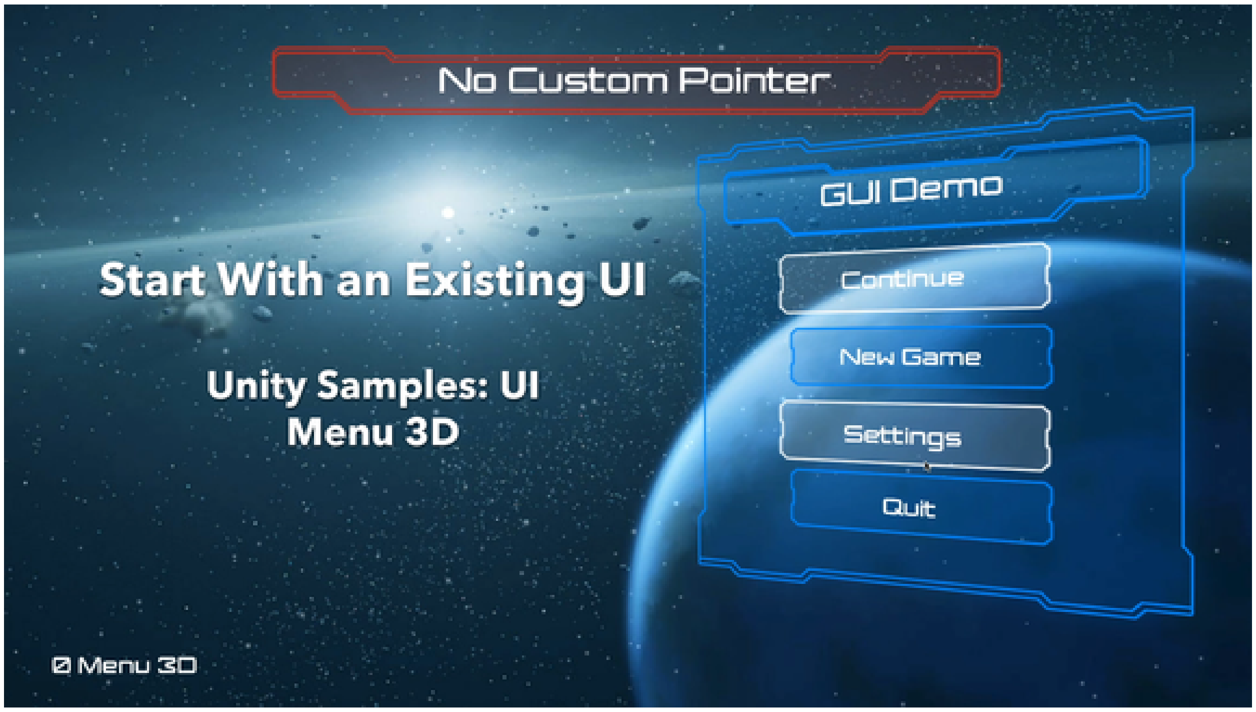
Optional

For Leap Motion support, one of the Leap Motion Core Assets packages for Unity is required. See the Leap Motion section below for more info.

- Leap Motion Core (Orion) Assets v4.4.0 supports Windows and VR.
- Leap Motion Core Assets v2.3.1 supports macOS, Linux, and Windows.
- Leap Motion Orion Beta v4.1.5 provides beta support Windows and VR (deprecated). Prefer v4.4.0 to this one.

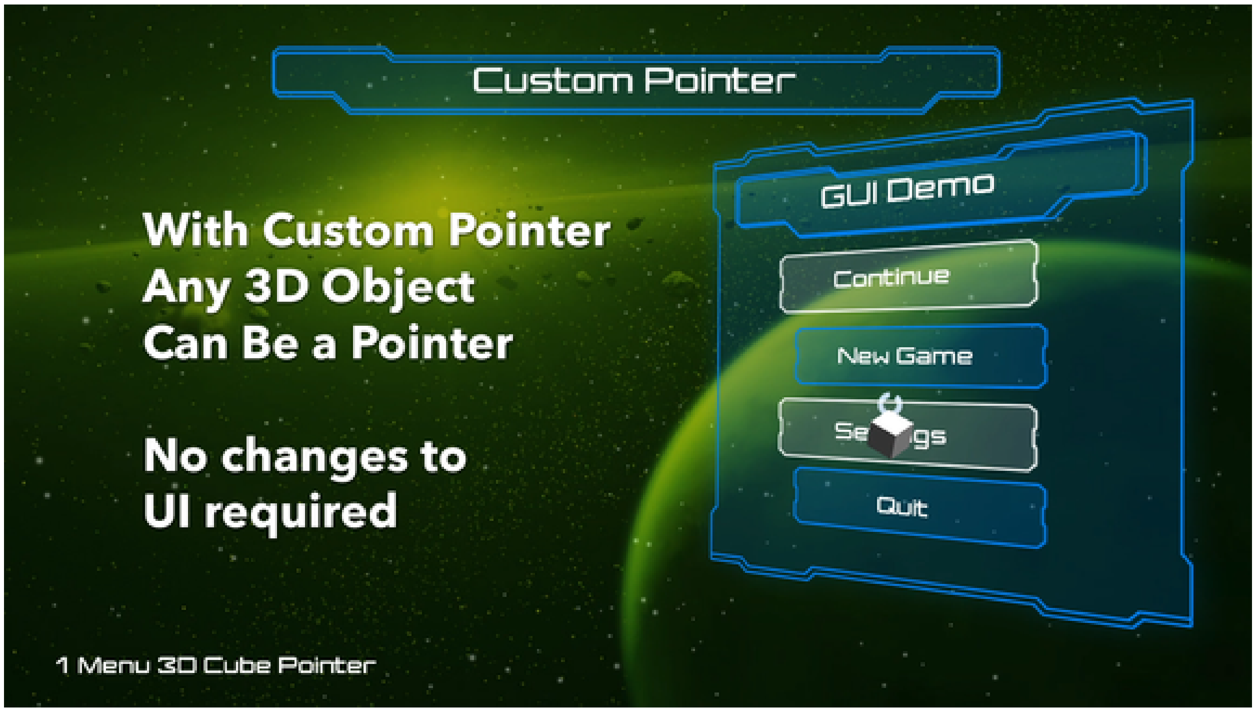
Scenes

The following scenes demonstrate what Custom Pointer can do for you.



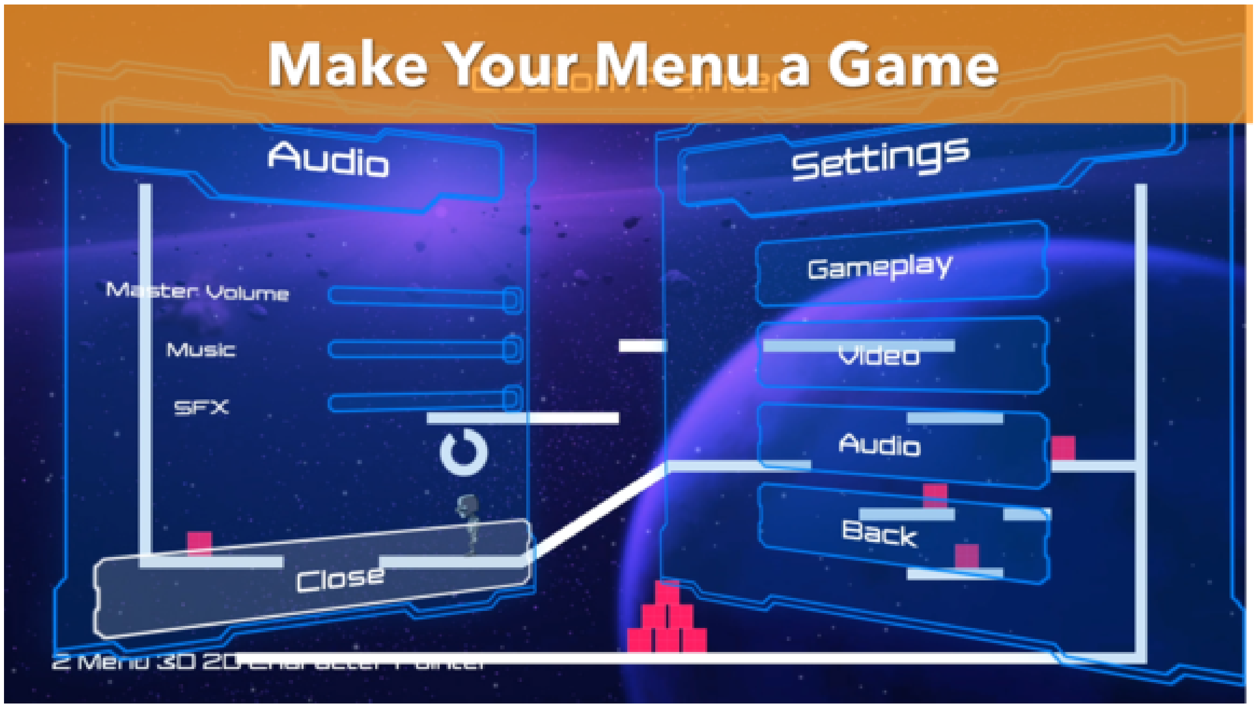
0. Menu 3D

This scene is from Unity Sample: UI. It is essentially unmodified. You can use your mouse or keyboard to interact with it. Custom Pointer is not enabled. Its purpose is to show that an unmodified UI may be used with a variety of pointers that Custom Pointer supports.


Custom Pointer
With Custom Pointer Any 3D Object Can Be a Pointer
No changes to UI required
GUI Demo
Continue
New Game
Settings
Quit
1 Menu 3D Cube Pointer

1. **Menu 3D Cube Pointer**

This scene uses a spinning 3D cube as a pointer. It can click by hovering or pressing space bar. This scene demonstrates that any Transform may be used as a pointer.

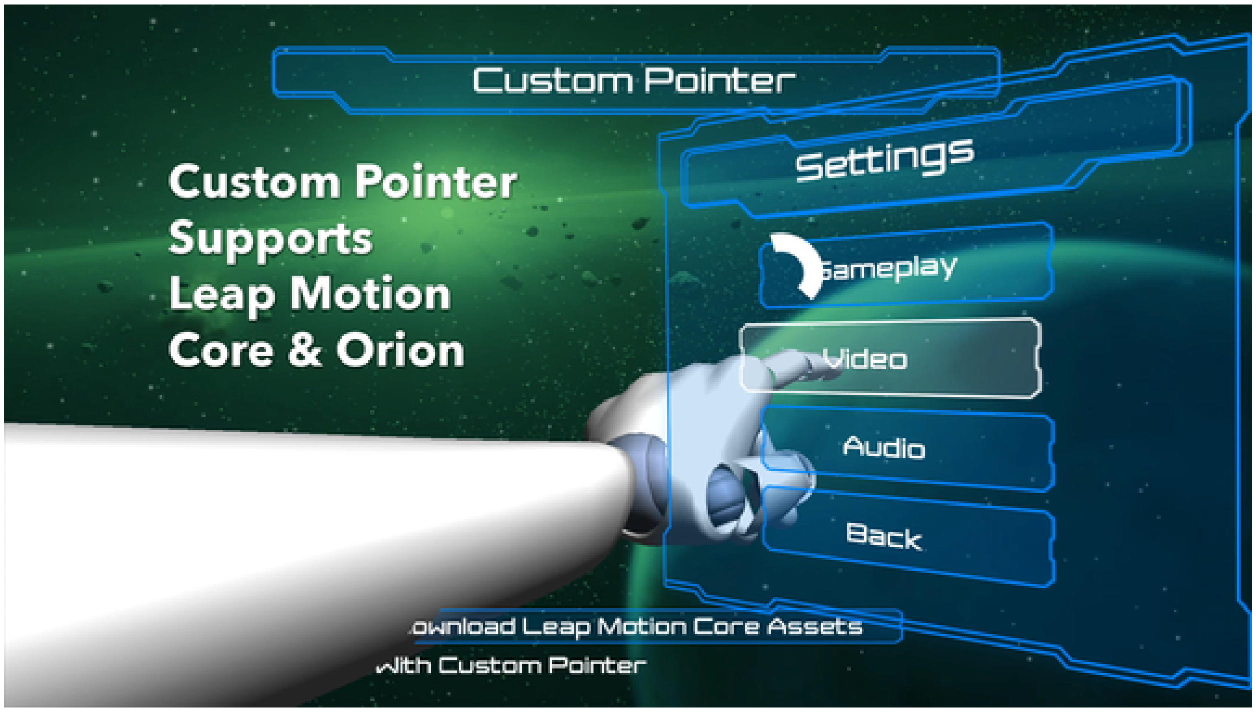


2. Menu 3D 2D Character Pointer

This scene uses a 2D game character as a pointer. Move the character using the arrow keys and space bar to jump. Move the character over a button to hover click it. This demonstrates how you might add some novelty to your menu.

3. Menu 3D Leap Motion Without Custom Pointer

This is nearly identical to scene 0; however, a Leap Motion controller has been added. If you have a Leap Motion, you will see your hand, but your hand will not be able to interact with the menu. This is the basic state of Unity UI and Leap Motion interaction currently without Custom Pointer. This scene only works with Leap Motion Core Assets.


Custom Pointer
Custom Pointer Supports Leap Motion Core & Orion
Settings
Gameplay
Video
Audio
Back
ownload Leap Motion Core Assets
With Custom Pointer

4. Menu 3D Leap Motion With Custom Pointer

Same scene as 1 except Custom Pointer support has been added. Now the Leap Motion hand can interact with the UI using hover click and hover drag. This scene only works with Leap Motion Core Assets.

5. Menu 3D Leap Motion Orion Without Custom Pointer

This scene is the same as 1 but modified for Leap Motion Orion beta software. The hands work but can't interact with the GUI. This scene only works with Leap Motion Orion beta.

6. Menu 3D Leap Motion Orion With Custom Pointer

This scene is the same as 2 but modified for Leap Motion Orion beta software. The hands can now interact with the GUI thanks to Custom Pointer. This scene only works with Leap Motion Orion beta.

Usage

How to Turn a Transform into a Pointer

Create an `EventSystem` if one does not already exist. Select the `EventSystem` in the scene, then select menu Component->Event->Transform Pointer Module. This adds the `TransformPointerModule` component which can be configured.

To turn a Transform into a pointer as is shown in the `Custom Pointer Keyboard Demo`, add a `RegisterTransformPointer` to the GameObject. Set the `Module` variable to the `EventSystem` in the scene.

Enable Hover Click

Typically the hover click is enabled by having a canvas in world space that attaches to the pointer. This will explain how to achieve that effect.

Attach the prefab `Custom Pointer Canvas` to the pointer Transform. Set the `ProgressBar` in `RegisterTransformPointer` to the `Progress Circle`.

The demo `Custom Pointer Keyboard Demo` shows this in action.

Using Rect Transforms as a Pointer

You may use a `RectTransform` as a Pointer; however, be mindful that your `RectTransform` may be "on top of" your UI elements. So it may look like the Custom Pointer code is not working, when it's actually your pointer that is occluding your UI elements. There is a workaround: Turn off any Raycast Targets on your pointer.

Enable Leap Motion

Leap Motion support is not enabled by default. If you do not have or intend on supporting Leap Motion, you can ignore the next two sections.

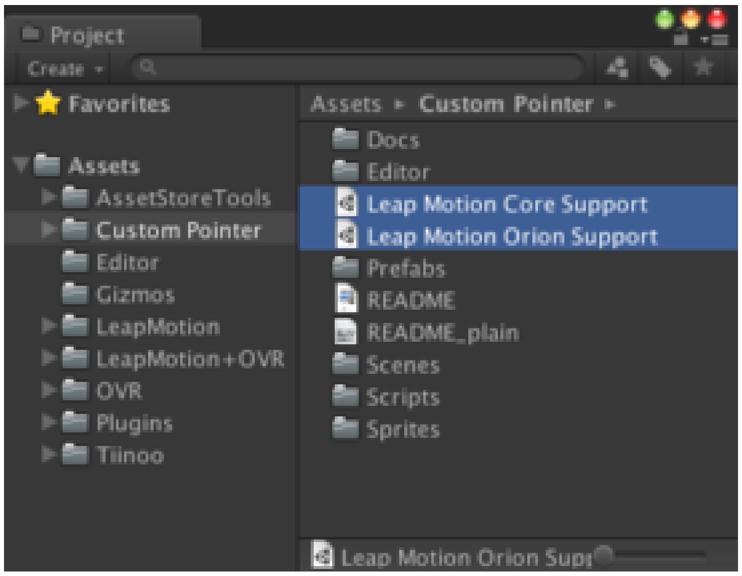
To enable Leap Motion support, download one of Leap Motion's Unity SDKs:

- Leap Motion Core (Orion) Assets v4.4.0 supports Windows and VR.
- Leap Motion Core Assets v2.3.1 supports macOS, Linux, and Windows.
- Leap Motion Orion Beta v4.1.5 provides beta support Windows and VR (deprecated). Prefer v4.4.0 to this one.

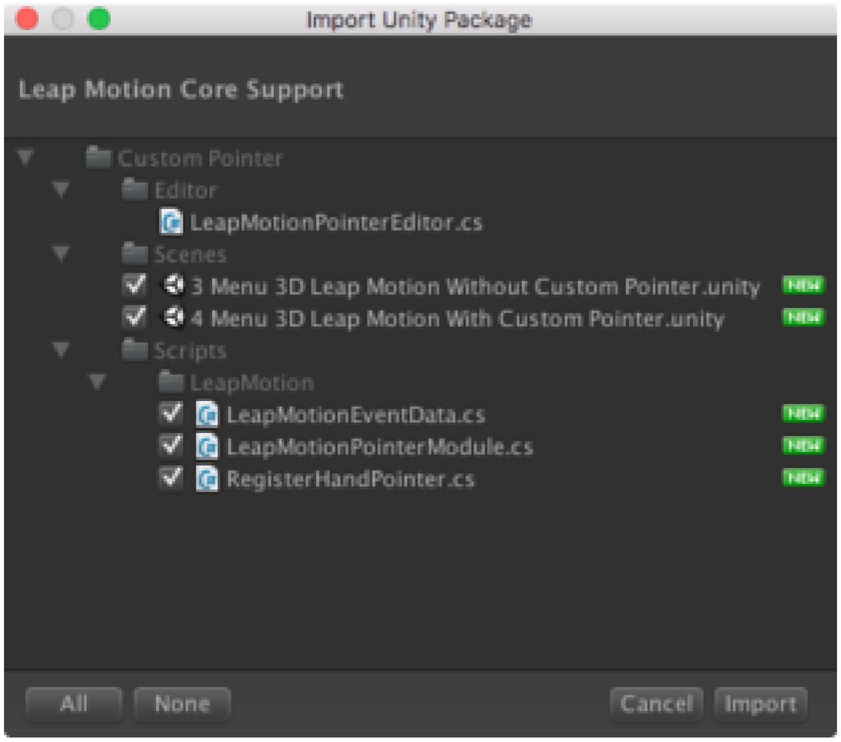
Do not import more than one; they are mutually exclusive.

Note: Leap Motion named their original Unity asset releases "Core" and their newer releases "Orion" and have now switched back somewhat to "Core". This can be a little confusing. Rely on the version numbers in preference to the names if there is ambiguity. Custom Pointer uses "Orion" to signify a Core Assets version >= 4.1.5.

Import Leap Motion Support



In the Custom Pointer directory, there are three Leap Motion Support packages: one for v4.4.0, one for v2.3.1, and one for the deprecated v4.1.5.


Import Unity Package
Leap Motion Core Support
Custom Pointer
Editor
LeapMotionPointerEditor.cs
Scenes
3 Menu 3D Leap Motion Without Custom Pointer.unity NEW
4 Menu 3D Leap Motion With Custom Pointer.unity NEW
Scripts
LeapMotion
LeapMotionEventData.cs NEW
LeapMotionPointerModule.cs NEW
RegisterHandPointer.cs NEW
All
None
Cancel
Import

Open the package corresponding to the Leap Motion assets you've installed and import the files. The script files will have the same names but the contents will differ. The Leap Motion Core support package will have scenes 3 and 4. The Leap Motion Orion support package will have scenes 5 and 6.

How to Turn a Leap Motion Controller Hand into a Pointer

Create an `EventSystem` and `HandController` if one does not already exist. Select the `EventSystem` in the scene, then select menu Component->Event->Leap Motion Pointer Module. This adds the `LeapMotionPointerModule` component which can be configured.

New IHandModel

With Leap Motion's Orion beta, you can turn a hand's index finger into a pointer by adding the `RegisterHandTransition` to the hand.

Old Hand Model

To turn a hand's index finger into a pointer as is shown in the `Custom Pointer Leap Motion Demo`, add a `RegisterTransformPointer` to the GameObject. Set the `Module` variable to the `EventSystem` in the scene. If this is unclear, inspect the `Custom Pointer with Leap Motion (Orion) Demo` to see how it's done.

Enable Hover Click on the Hand

Typically the hover click is enabled by having a canvas in world space that attaches to the pointer. This will explain how to achieve that effect.

Attach the prefab `Custom Pointer Canvas` to the tip of the index finger. Set the `ProgressBar` in `RegisterHandPointer` to the `Progress Circle`.

The demo `Custom Pointer Leap Motion Demo` shows this in action.

Contact

- twitter: @shanecelis
- website: seawisphunter.com