

Thank you for purchasing VR Hand Gesture Recognizer (Oculus Quest Hand Tracking).
If you have any questions, comments, or requests please email
support@blacklumenvr.com

Please Note:

You must have imported the Oculus Quest Hand Tracking SDK before using this asset as it is dependent on having the Quest hand tracking objects in your scene in order to function
https://developer.oculus.com/documentation/unity/unity-handtracking/?locale=en_US

Please follow the directions in the link above to setup Oculus Hand Tracking with your project before proceeding with the Asset.

Saving Gestures:

1. Ensure that the Hand Tracking SDK is enabled and working within your project.
2. Add the GestureSaver prefab to your scene.
3. Press Play in your scene and after you see your hands in the scene you can save a gesture by hitting the appropriate Save Gesture button or the chosen SaveKey on the keyboard.
4. Your gesture will be saved as an asset file to Assets/Resources/"GestureName".asset



(Figure 1: Gesture Saver Default Configuration)

Settings (Found on GestureSaver)

HandToSave: Choose which hand this GestureSaver is looking at

Save Key: Choose which keyboard key can be used to automatically save a gesture for the selected HandToSave

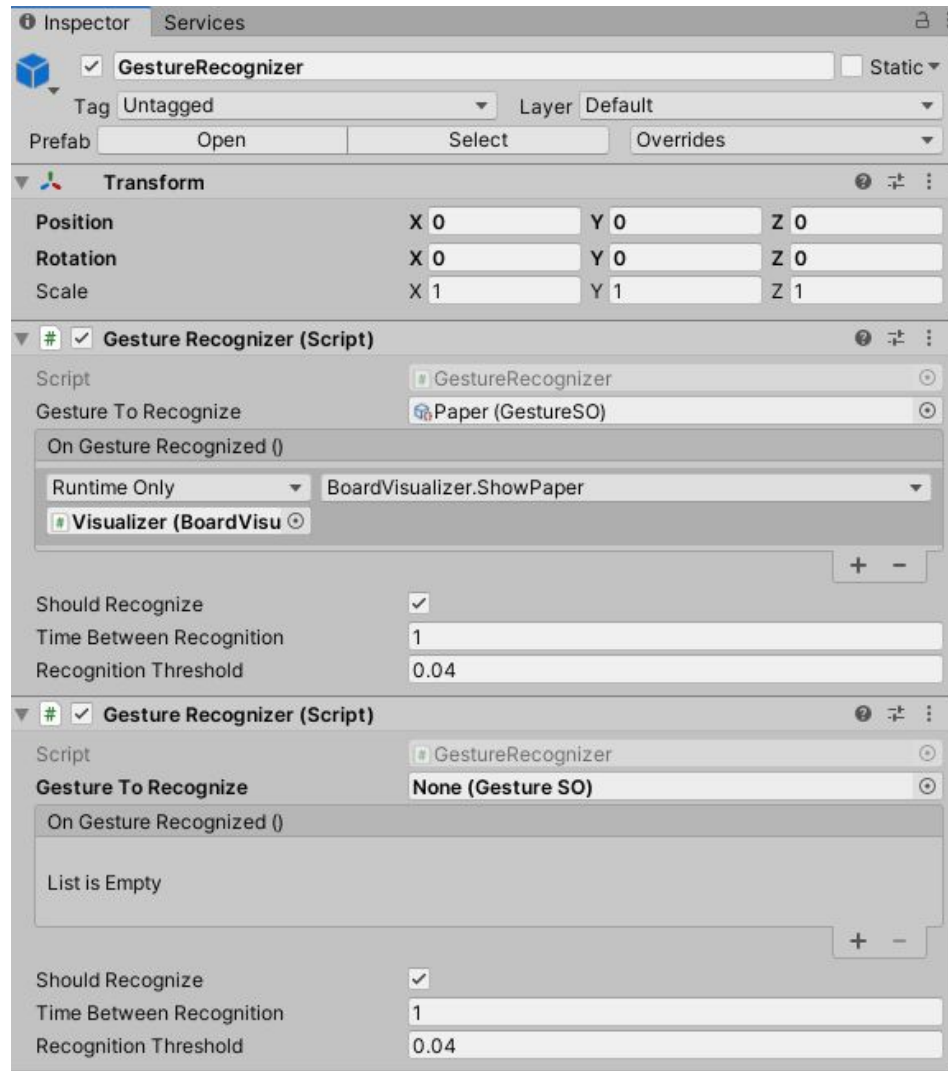
GestureName: Set the desired GestureName that will be saved into Assets/Resources

SaveGesture: Alternate way of saving a gesture with specified GestureName

Saving gestures works by capturing the selected hand when you hit the save button or the save key. Be sure to set a GestureName before saving a gesture otherwise you may have difficulty finding your saved gesture. Currently if you want a gesture to be recognized by both hands you'll have to save a gesture file for each hand.

Recognizing Gestures:

1. Add the GestureRecognizer prefab to your scene
2. Drag a saved gesture file (Located in Assets/Resources) into the GestureToRecognize slot of the GestureRecognizer
3. Hit the + button to add an event listener to OnGestureRecognized so that you can react to a recognition
4. If you'd like to recognize more than one gesture then AddComponent > GestureRecognizer to add each additional gesture recognizer you want.



Example setup of a GestureRecognizer (Top is ready to go and bottom is blank)

Settings (Found on GestureRecognizer)

Gesture To Recognize: Choose the gesture file you'd like to recognize

OnGestureRecognized: Let's you specify methods or functions that will be called when a gesture is recognized by the system.

Should Recognize: Disabling this will pause recognition for that gesture

Time Between Recognition: Cooldown time between firing successive gesture recognition events (recommended setting is at least .5 seconds)

Recognition Threshold: Governs precision required to initiate a successful gesture recognition event. A lower number will require fingers to be positioned much closer to the original saved

hand gesture. A higher number allows for looser recognition but may not be ideal with multiple gestures. (Recommended value is .04)

When a gesture is recognized an event is fired called OnGestureRecognized. Any event listeners you have hooked up to that event will be called and you can take action in your scene based on the result. For more information on this see the following link:

<https://docs.unity3d.com/Manual/UnityEvents.html>

Handling Gesture Recognition Events:

The sample scene showcases how you can have each gesture recognition tied to a different method that has logic in it to change elements of your scene. In our case we are changing the material of the object in the picture below to give the user positive feedback that their gesture was recognized.



FAQ:

Can I save both hands at the same time?

This is not currently supported but if this is something the community desires then I will add it as a feature.

Adding multiple GestureRecognizers is a pain. Is there a faster way to do this?

Currently there is not but that is something that I will be adding in the next update. Please email me if you'd like early access to updates or have feature requests you'd like to see rolled in.