



Introduction

Thank you for purchasing the HoloLens Shader Pack. You can now apply optimized shaders to your Spatial Mapping or spice up your models with one of the Surface Shaders that are included in the package.

Support for the Mixed Reality Toolkit V2 and the old HoloToolkit

The shaders and materials in this package can be used without the Mixed Reality Toolkit (MRTK2) or HoloToolkit as shown in the demo scenes. However, there are a number of scripts that provide helper functionality to use these shaders with the HoloLens for instance to update a gaze location in the shaders.

Automatic toolkit detection

The HoloLens Shader Pack includes a utility script that detects which toolkit is already available in the project. This will add the Scripting Define Symbols HAVE_MRTK for Mixed Reality Toolkit V2 and HAVE_HTK for the older HoloToolkit. These Scripting Define Symbols activate the parts of code that depend on these toolkits. If both toolkits are found the MRTK2 will have preference in the scripts.

Using spatial mapping shaders with the MRTK2

Microsoft's Mixed Reality Toolkit V2 is not included in this package and can be downloaded as unitypackage from GitHub (or by using Nuget).

Mixed Reality Toolkit download location

<https://github.com/Microsoft/MixedRealityToolkit-Unity/releases>

After configuration you will have a MixedRealityToolkit object in your scene with a MixedRealityToolkit script component. This component contains an Active Profile that configures which services will be used. Typically, you will start with one of the default profiles like the DefaultHoloLens1ConfigurationProfile or DefaultHoloLens2ConfigurationProfile. Both will have a DefaultMixedRealitySpatialAwarenessSystemProfile, but note that spatial awareness is disabled by default for these two default profiles.

Create a new configuration profile by duplicating the default and enable the Spatial Awareness System:

MixedRealityToolkit GO -> MixedRealityToolkit Component -> Spatial Awareness -> Enable Spatial Awareness System

The spatial Awareness System by default contains a Windows Mixed Reality Spatial Mesh Observer that generates the spatial mapping mesh at runtime. To change the material/shader that is shown on the spatial mesh unfold the properties and replace the Visible Material with a material of your choice.

MixedRealityToolkit GO -> MixedRealityToolkit Component -> Spatial Awareness -> Windows Mixed Reality Spatial Mesh Observer -> Display Settings -> Visible Material

Using spatial mapping shaders with the HoloToolkit

If you have a Unity project that still uses the older HoloToolkit you can still use the included scripts. When using shaders on the SpatialMapping you need to add the SpatialMappingCalcNormals script to the scene. Either in an empty gameobject or on the SpatialMapping prefab that you imported from the HoloToolkit.

You can use one of the predefined materials and set them as the SurfaceMaterial on the SpatialMappingManager component. Or you can create a new material that uses one of the shaders under the shader folder *HoloLens Shader Pack*.

Pulse animation

If *Pulse* is enabled in the used SpatialMapping material you can choose to animate the pulse based on gaze by adding a GazeTransitionAnimator script to your scene. This will automatically generate animated pulses from the location that you are looking at.

If you want to animate a pulse by tapping, add the TapTransitionAnimator script to the SpatialMapping prefab that you imported from the Mixed Reality Toolkit.

Gaze Animation

The GazeAnimator is a small script that copies the gaze hitpoint location to the specified material so gaze can be visualized in the shader instead of using a separate cursor object. This script works with both the MRTK2 and the HoloToolkit. The *Fragments* material is a material that makes use of this feature.

Troubleshooting

SpatialMapping textures are stretched along some walls when using MRTK2

Make sure that normal calculation is enabled:

MixedRealityToolkit GO -> MixedRealityToolkit Component -> Spatial Awareness -> Windows Mixed Reality Spatial Mesh Observer -> Physics Settings -> Recalculate Normals

SpatialMapping textures are stretched along some walls when using HoloToolkit

Check if you added the SpatialMappingCalcNormal script to scene as described above. The generated textures depend on properly generated normals to calculate the best texture coordinates.

The intersection of the SlicingGradient shader does now work properly on HoloLens

Make sure to build your project with a DepthBuffer of 24-bits.

Scripts that update the shaders don't seem to work

Check if the Scripting Define Symbols for the toolkit that you are using are generated properly:

Player Settings -> Other Settings -> Configuration -> Scripting Define Symbols -> HAVE_MRTK (or HAVE_HTK)

If the Scripting Define Symbols does not contain the HAVE_MRTK or HAVE_HTK you can add it by hand to fix it.

Support or feature requests

For help or feature requests please contact Roland Smeenk

roland.smeenk@gmail.com (preferred for support requests)

<http://www.smeenk.com>

<https://twitter.com/rolandsmeenk>

<https://www.youtube.com/user/RolandSmeenkNL>

Documentation changelog

1.0 Initial version

2.0 Changed HoloToolkit to Mixed Reality Toolkit to match Microsoft's renaming

Mentioned MRTK 2017.4.0.0 as version tested with the HoloLens Shader Pack

Added "Troubleshooting" section with common support issues

3.0 Updates to explain the automatic handling of MRTK2 or HoloToolkit support and how to work with the new Mixed Reality Toolkit.