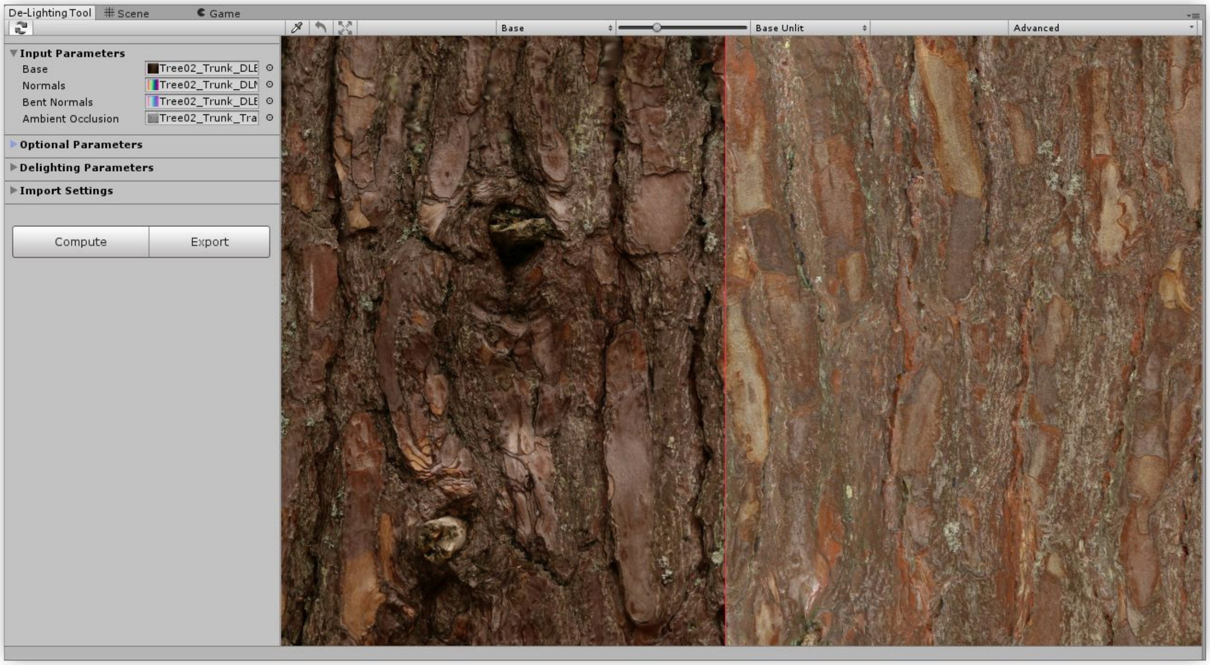
De-Lighting in Unity



Input Parameters

Base: Tree02_Trunk_DLE

Normals: Tree02_Trunk_DLN

Bent Normals: Tree02_Trunk_DLE

Ambient Occlusion: Tree02_Trunk_Tra

Contents

Introduction

Version requirements

Installation

- Download Unity
- Download De-Lighting Tool project

Tool interface overview

De-lighting workflow

- Import data
- De-lighting process
  - Evaluate the de-lit result
  - Navigate the viewport
  - White balance and exposure in the reference zone
- Export the result

Advanced de-lighting

- Optional parameters
  - The position map
  - Mask maps
    - The red channel
    - The green channel
- Advanced mode
  - Optional Parameters
  - De-Lighting Parameters
    - Remove Highlights
    - Remove Dark Noise
    - Separate Dark Areas
    - Force Local De-Lighting
  - Import Settings
  - Comparison channels
- Debug mode
  - Main Env Map section
  - Baked LUT (lookup table) section

Command line

- De-light a texture in a folder:

De-light multiple textures:
De-light textures using a batch script:

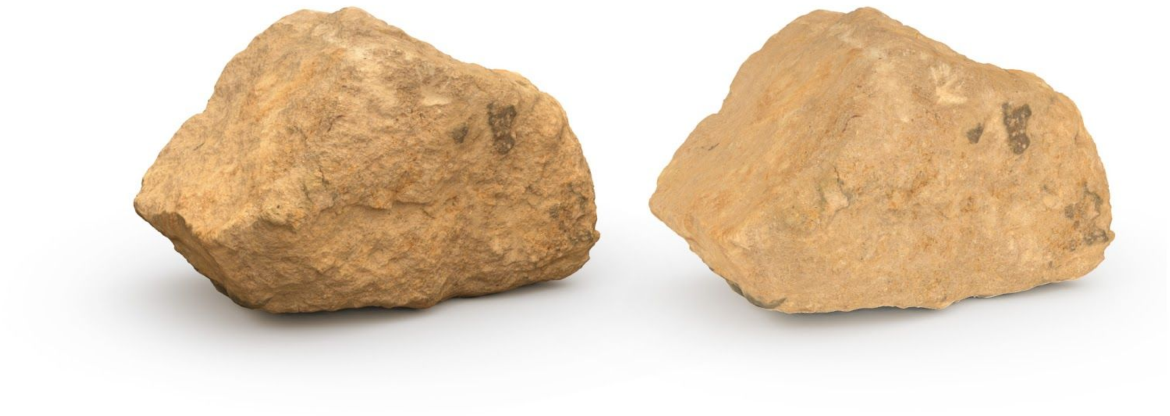
Introduction

Photogrammetry is becoming more and more popular in the CG industry. Being able to use a regular camera as a powerful 3D scanner is now an essential tool when creating realistic 3D assets. However, when you generate raw textures in this way, they often contain a lot of lighting information that you should remove so as to obtain the true base color for rendering. The Unity De-Lighting Tool has been developed to solve this complex problem.

The De-Lighting Tool performs the following steps:

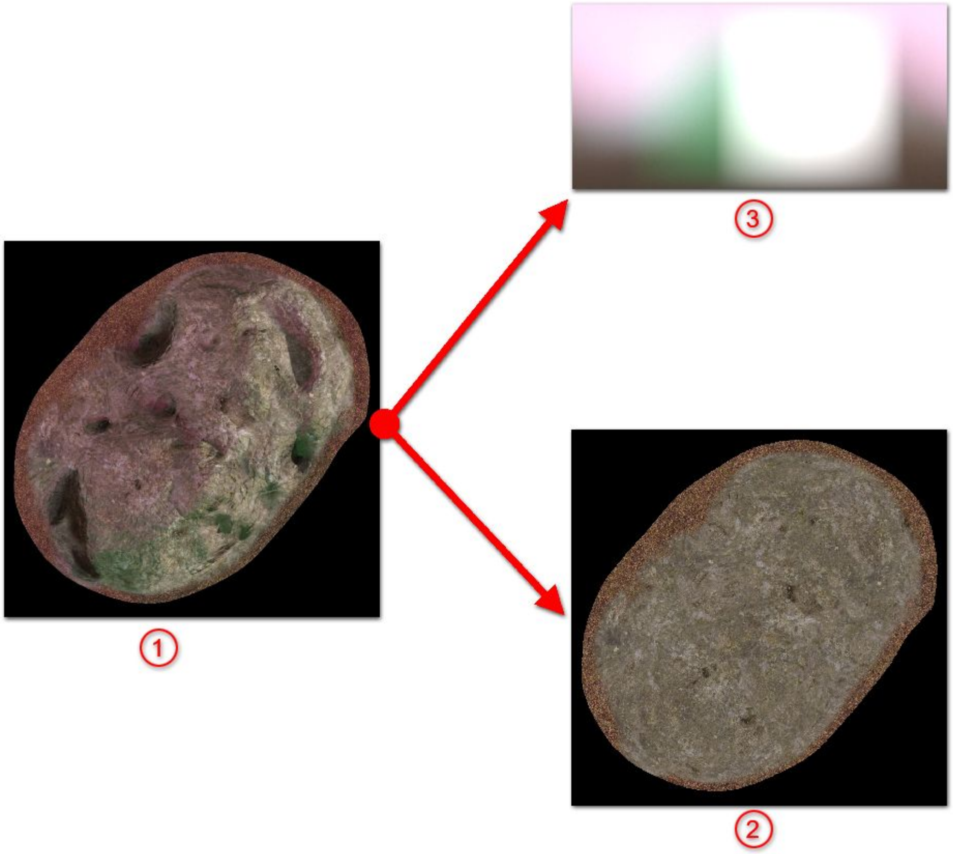
- Environment lighting reconstruction
- Image-based de-lighting
- Ambient and global illumination lighting removal
- Exposure compensation and color correction

After performing these steps, the tool produces a texture that does not have any lighting, a *de-lighted* texture.



Lit and de-lighted object

Additionally, the tool can output an environment map in a latitude-longitude format, which you can use for debugging or validation.



(1) Input texture with lighting information created from photogrammetry, (2) De-lighted texture, (3) Environment lighting in latitude/longitude coordinates

Version requirements

The De-Lighting Tool is compatible with **Unity 5.6** and above. You should use the De-Lighting Tool with the **Linear Color Space** project configuration. To switch to **Linear Color Space**:

1. Select **Edit** > **Project Settings** > **Player**.
2. In the **Other Settings** > **Rendering** section, set **Color Space** to **Linear**.

Installation

Download Unity

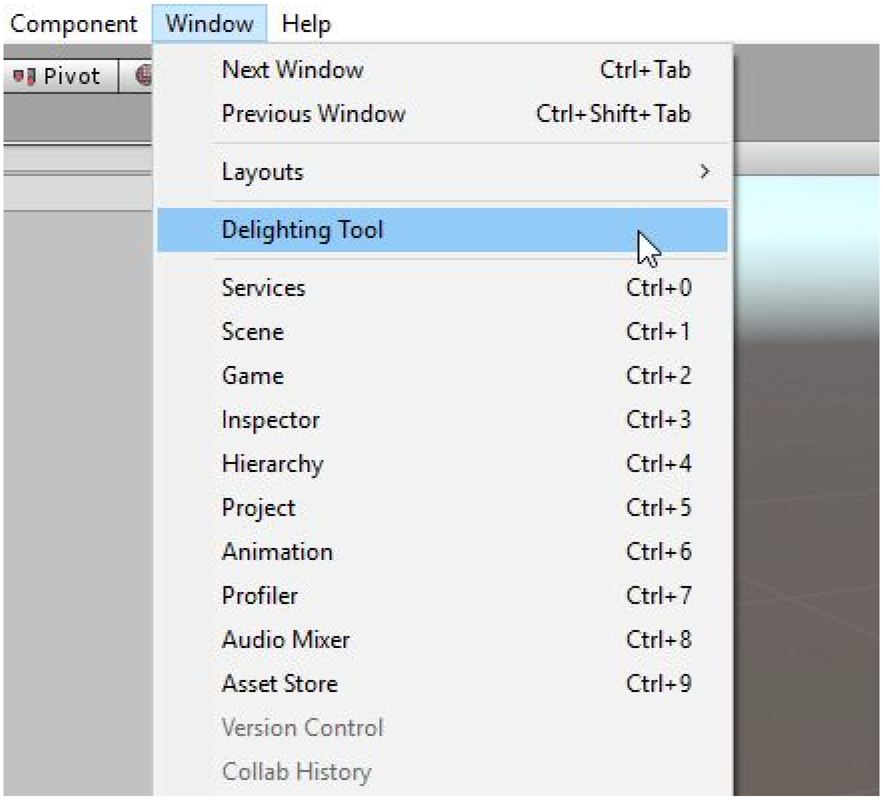
Download and install Unity 5.6 or above by following this link: Unity Download Section. For more information on downloading and installing Unity, see Unity Manual: Download and Install.

Download De-Lighting Tool project

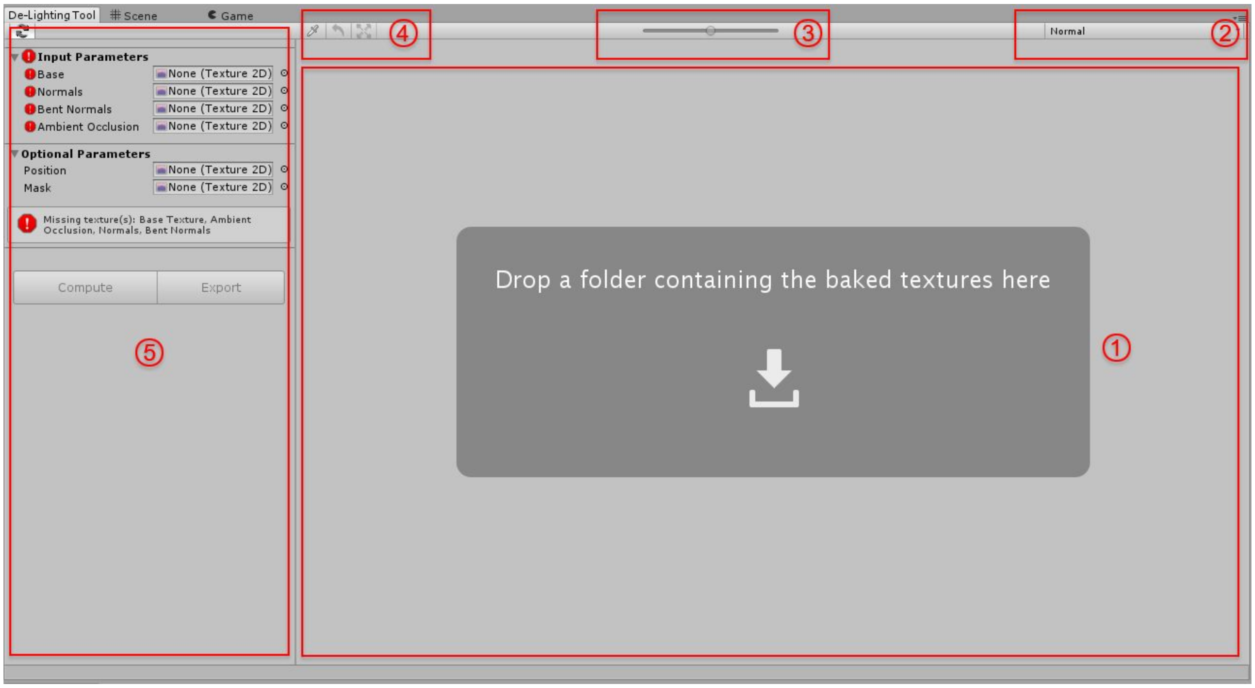
Download the De-Lighting Tool package at: Github Download Link

Tool interface overview

To open the De-Lighting Tool, select **Window** > **De-Lighting Tool**.



Once opened, the tool looks like this:



Input Parameters

Base	None (Texture 2D)
Normals	None (Texture 2D)
Bent Normals	None (Texture 2D)
Ambient Occlusion	None (Texture 2D)

Optional Parameters

Position	None (Texture 2D)
Mask	None (Texture 2D)

Missing texture(s): Base Texture, Ambient Occlusion, Normals, Bent Normals

Drop a folder containing the baked textures here

The interface is split into the following areas:

1 - Preview
2 - View mode switcher
3 - Comparison slider
4 - Preview toolbar:

- Set the reference zone
- Reset the reference zone
- Fit to screen

5 - De-Lighting Tool Inspector (Custom Inspector)

De-lighting workflow

The De-Lighting Tool aims to remove the lighting that remains in 2D textures exported from photogrammetry software, such as Reality Capture and Photoscan. The tool requires specific textures from baking software, such as:

- Ambient occlusion map
- Normal map
- Bent normal map

You can bake these specific maps using xNormal, Knald or Substance Designer. Guidelines for baking are available here: Photogrammetry Workflow (5.4 Textures baking). This document explains the setup that you should use in those tools to get good results with the De-Lighting Tool.

Note: For ambient occlusion and bent normal map baking, you can use uniform and cosine distribution (or weighting in Knald). The baking context should be as close as possible to the real context of the capture.

In baking software, you can reference a list of HD geometries instead of one. If the original object was on the ground, you need to add a ground mesh to the list of HD geometries.

Import data

The tool only processes and uses textures. Because the texture import settings can dramatically change the way a texture is interpreted, the tool generates texture auto-import scripts. That way, you don't need to manage texture settings in the Unity Editor every time a new texture is processed. For texture auto-import to work, you must give textures the right suffix. By default, the naming conventions are:

*_DLBC.tga => Base Map: (RGBA) Original color texture to unlit *(mandatory texture)*
*_DLAO.tga => (Grey) Ambient Occlusion Map *(mandatory texture)*
*_DLN.tga => (RGB) Object-Space Normal Map *(mandatory texture)*
*_DLBN.tga => (RGB) Object-Space Bent Normal Map *(mandatory texture)*

*_mask.tga => (RGB) De-Lighting Parameter Map *(additional texture)*
*_position.exr => (RGB) Texel Position *(additional texture)*

In the Base Map, the alpha channel marks the pixels that need de-lighting. **If the alpha channel isn't correctly authored, the de-lighting result won't be correct.**

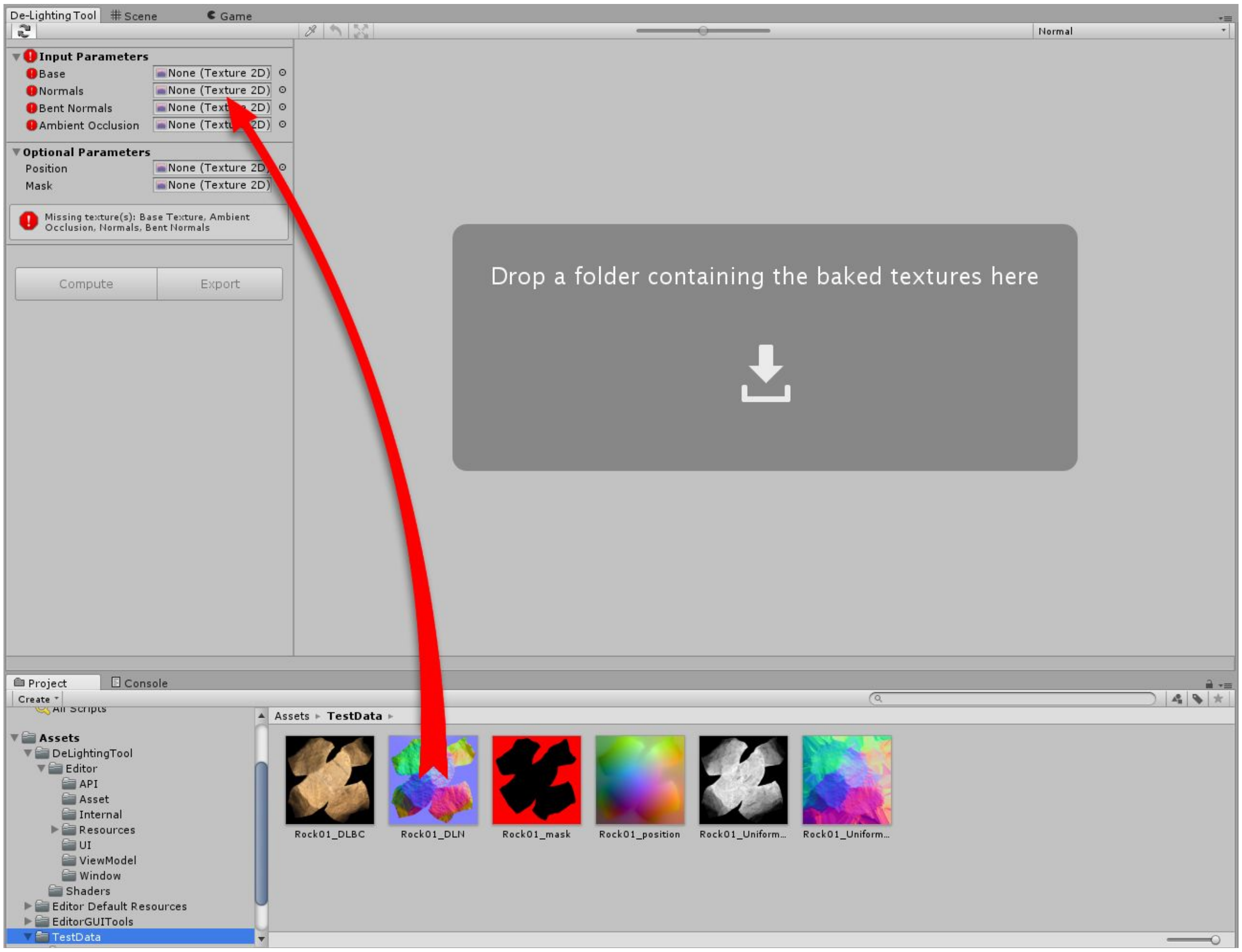
Using textures without this naming convention works fine, but if you don't use the convention then you won't benefit from automatic texture updates in Unity. If you do not use the convention, then all texture settings have to be set manually, like this:

- Set the sRGB import option only on the Base Map (diffuse map).
- Set the maximum size to 8KB and the compression to none for all textures.
- Set the texture type of all the maps to **Default**, even for normal maps.

You can change the naming conventions, as explained in the Import Settings section later in this document.

De-lighting process

Open the tool and drag and drop each required texture in the right slot.



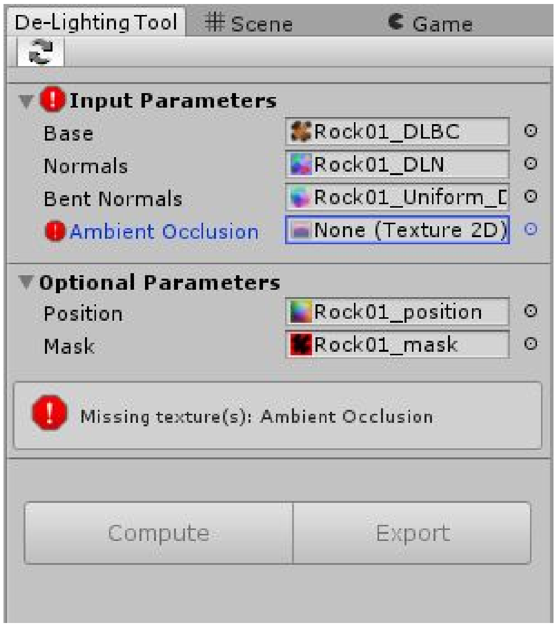
Input Parameters
Base: None (Texture 2D)
Normals: None (Texture 2D)
Bent Normals: None (Texture 2D)
Ambient Occlusion: None (Texture 2D)

Optional Parameters
Position: None (Texture 2D)
Mask: None (Texture 2D)

Missing texture(s): Base Texture, Ambient Occlusion, Normals, Bent Normals

Drop a folder containing the baked textures here

If an exclamation mark appears next to a texture field, then there is an error with the texture. In that case, follow the instructions in the error description. For instance, in this example the **Ambient Occlusion** texture is missing:



Input Parameters

Base: Rock01_DLBC

Normals: Rock01_DLN

Bent Normals: Rock01_Uniform_D

Ambient Occlusion: None (Texture 2D)

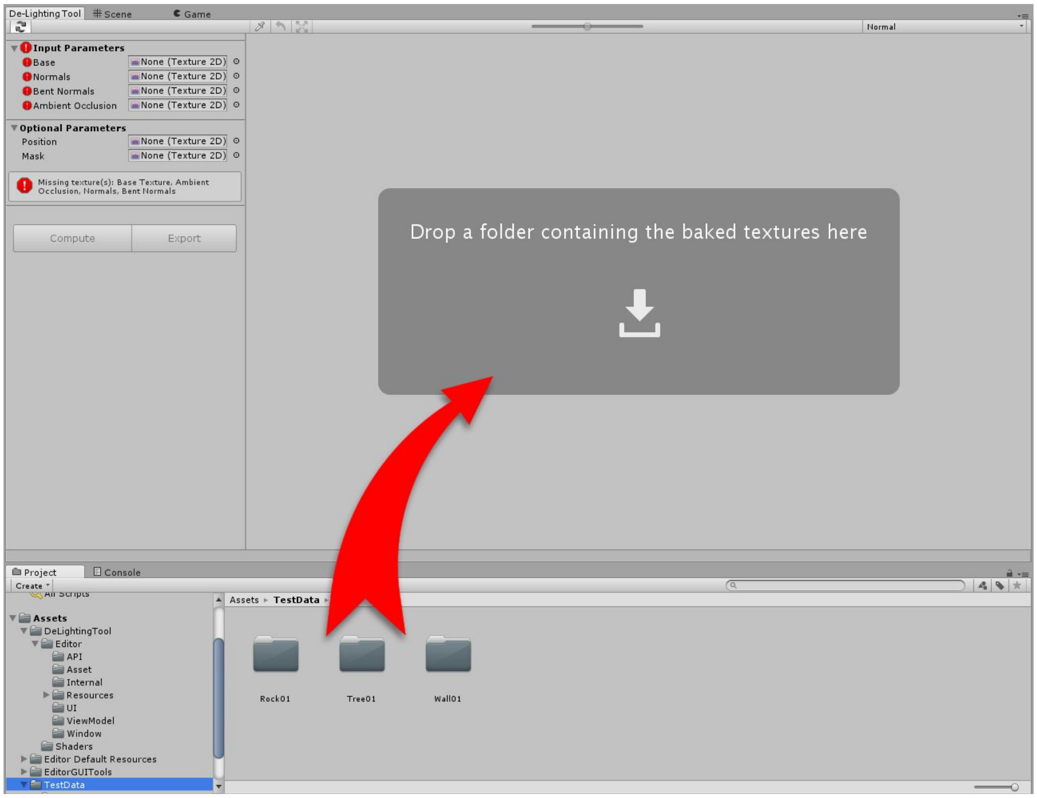
Optional Parameters

Position: Rock01_position

Mask: Rock01_mask

Missing texture(s): Ambient Occlusion

You can also drag and drop the texture folder into the preview area to fill the texture fields:



Input Parameters

Base: None (Texture 2D)

Normals: None (Texture 2D)

Bent Normals: None (Texture 2D)

Ambient Occlusion: None (Texture 2D)

Optional Parameters

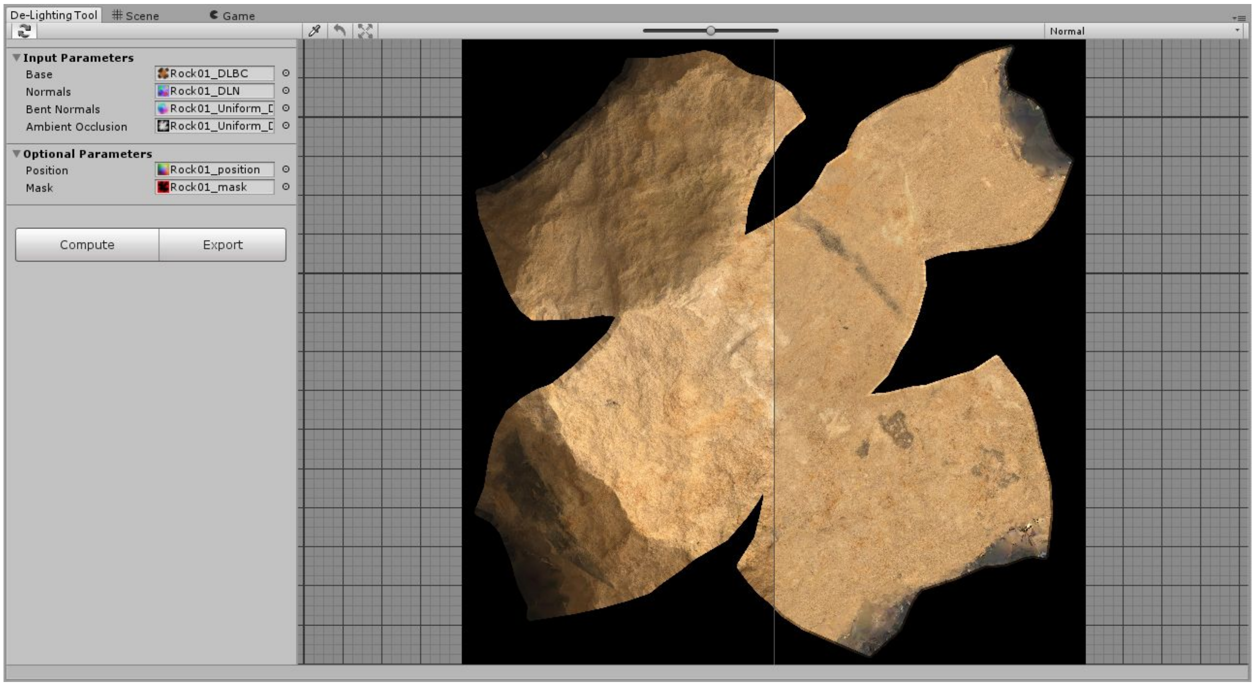
Position: None (Texture 2D)

Mask: None (Texture 2D)

Missing texture(s): Base Texture, Ambient Occlusion, Normals, Bent Normals

Drop a folder containing the baked textures here

When all of the textures in the **Input Parameters** section are filled, the tool processes the De-Lighting automatically.



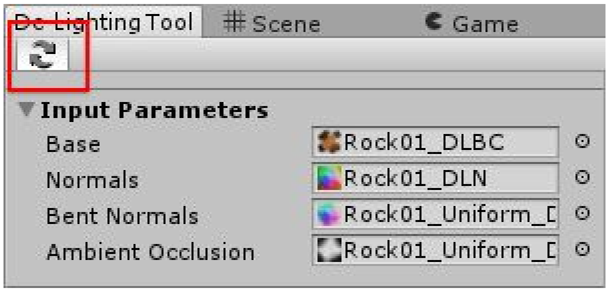
Input Parameters

Base	Rock01_DLBC
Normals	Rock01_DLN
Bent Normals	Rock01_Uniform_[
Ambient Occlusion	Rock01_Uniform_[

Optional Parameters

Position	Rock01_position
Mask	Rock01_mask

When you press the auto-update button, the result updates every time you change a parameter:

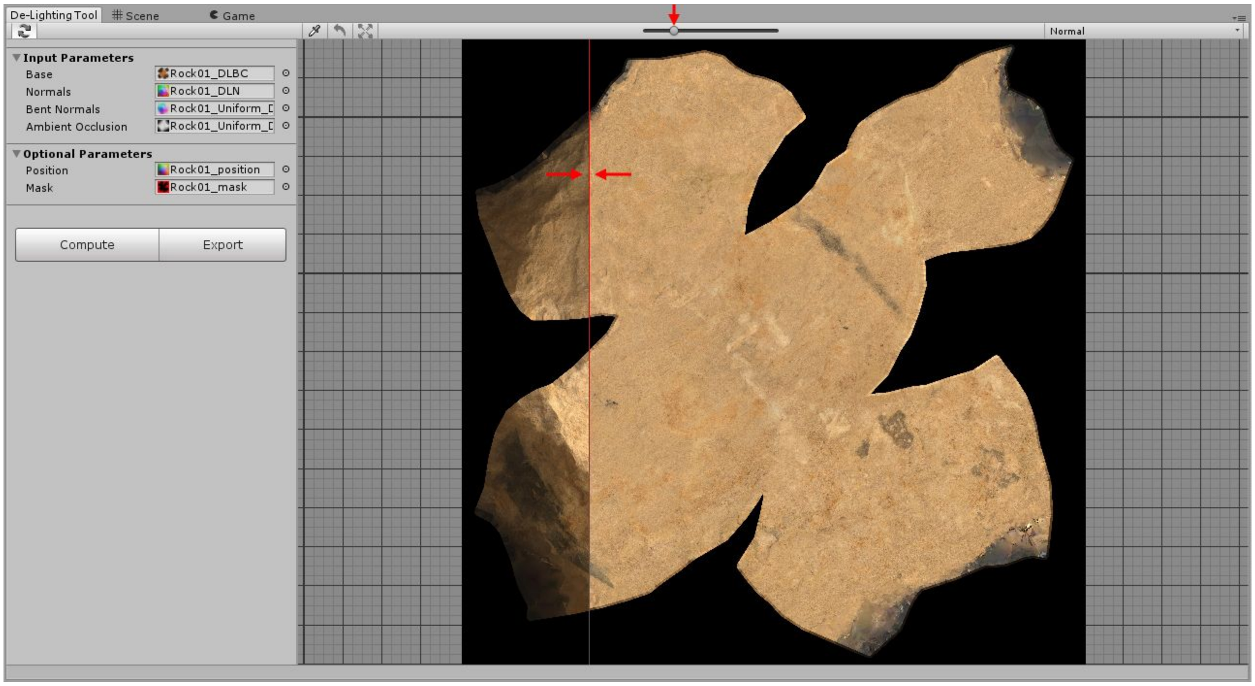


Input Parameters

Base	Rock01_DLBC
Normals	Rock01_DLN
Bent Normals	Rock01_Uniform_[
Ambient Occlusion	Rock01_Uniform_[

Evaluate the de-lit result

The preview area is split into two parts: the left side shows the original texture and the right side shows the de-lit result. Moving the comparison slider moves the preview splitter from left to right. You can also move the splitter by left-clicking anywhere on the result picture:



Input Parameters

Base Rock01_DLBC

Normals Rock01_DLN

Bent Normals Rock01_Uniform_D

Ambient Occlusion Rock01_Uniform_D

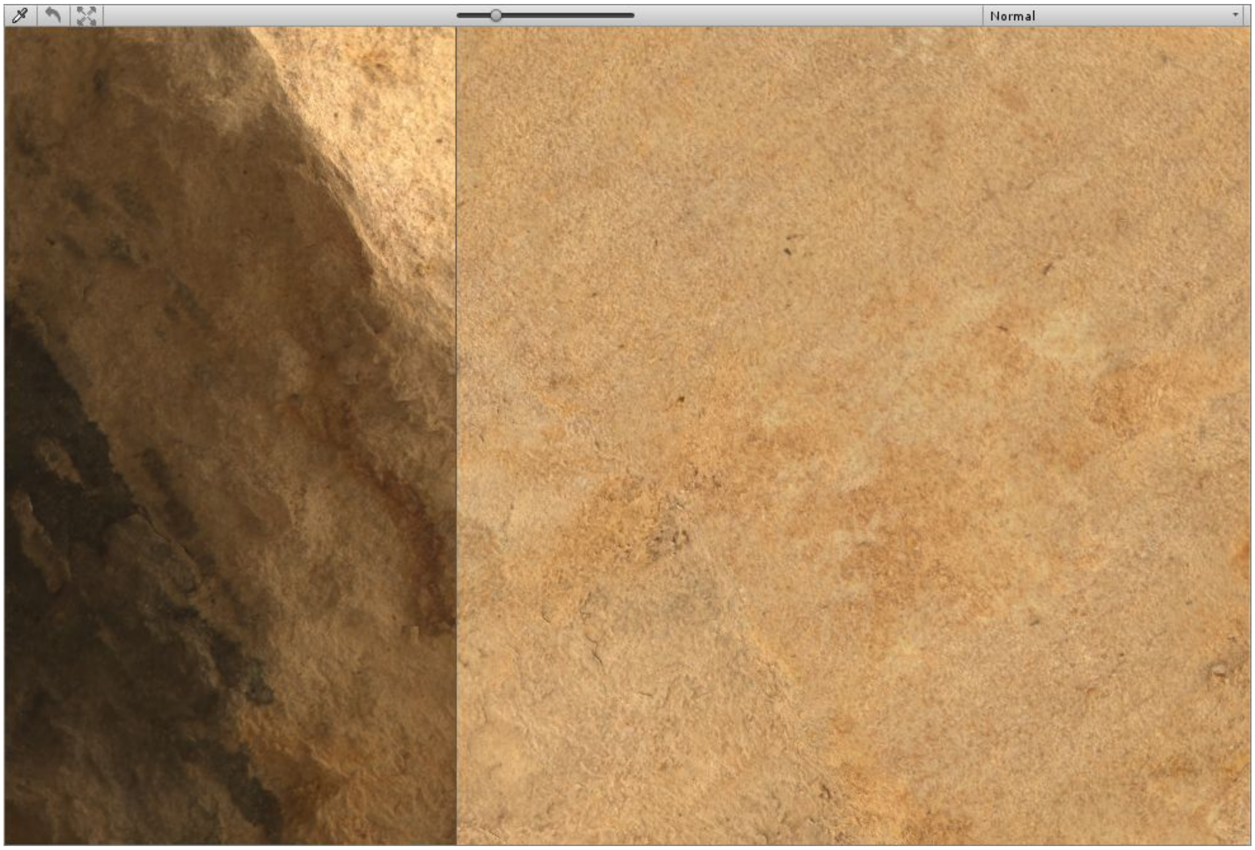
Optional Parameters

Position Rock01_position

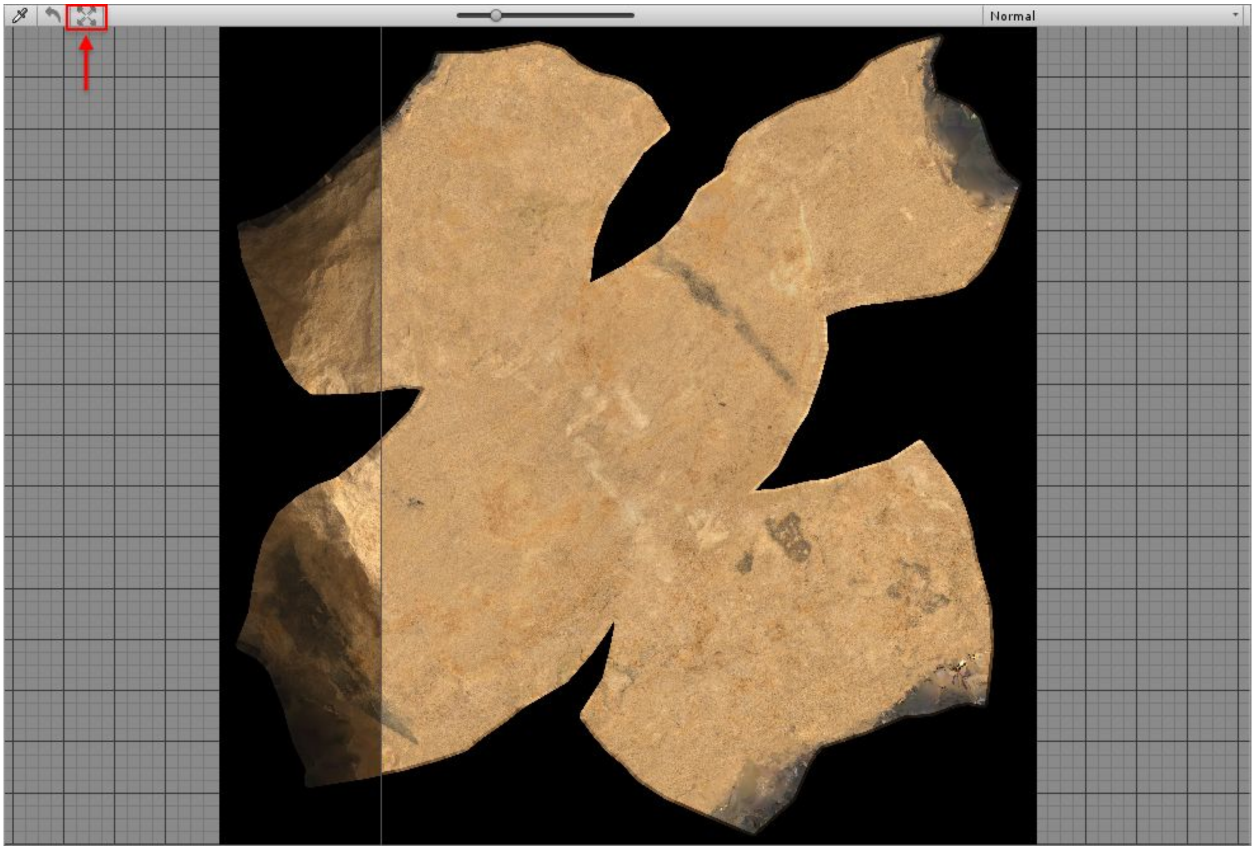
Mask Rock01_mask

Navigate the viewport

Middle-click the mouse to **Pan** in the viewport. Use the mouse wheel to **Zoom** in and out.

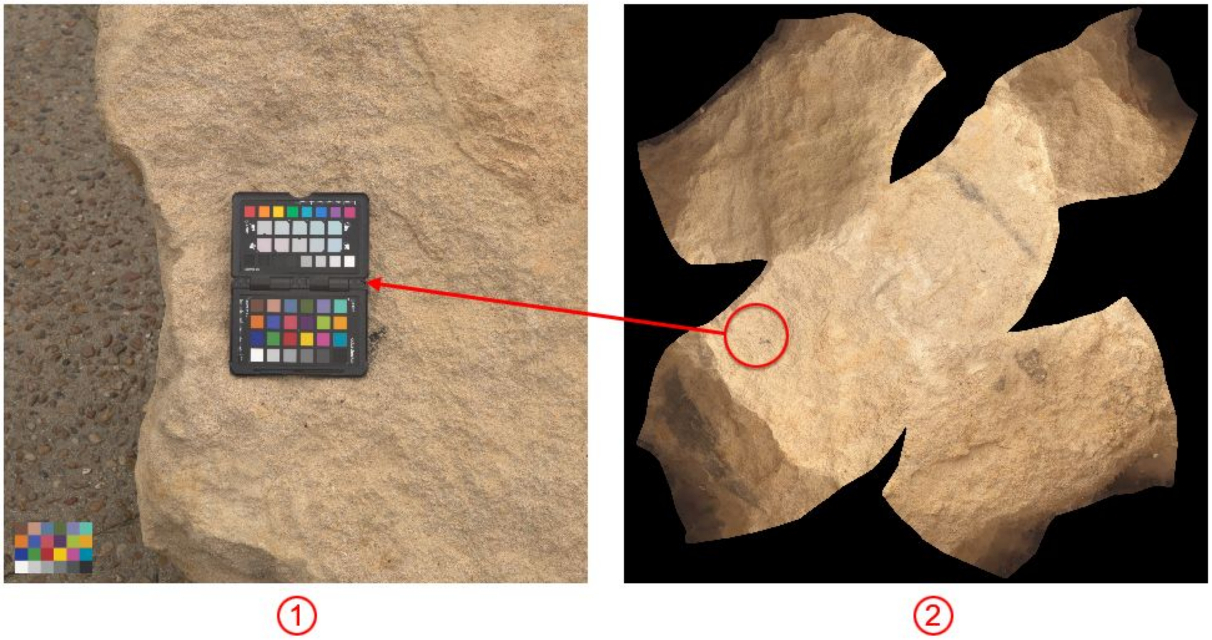


Click the fit to view button in the top left of the preview window to reset pan and zoom:



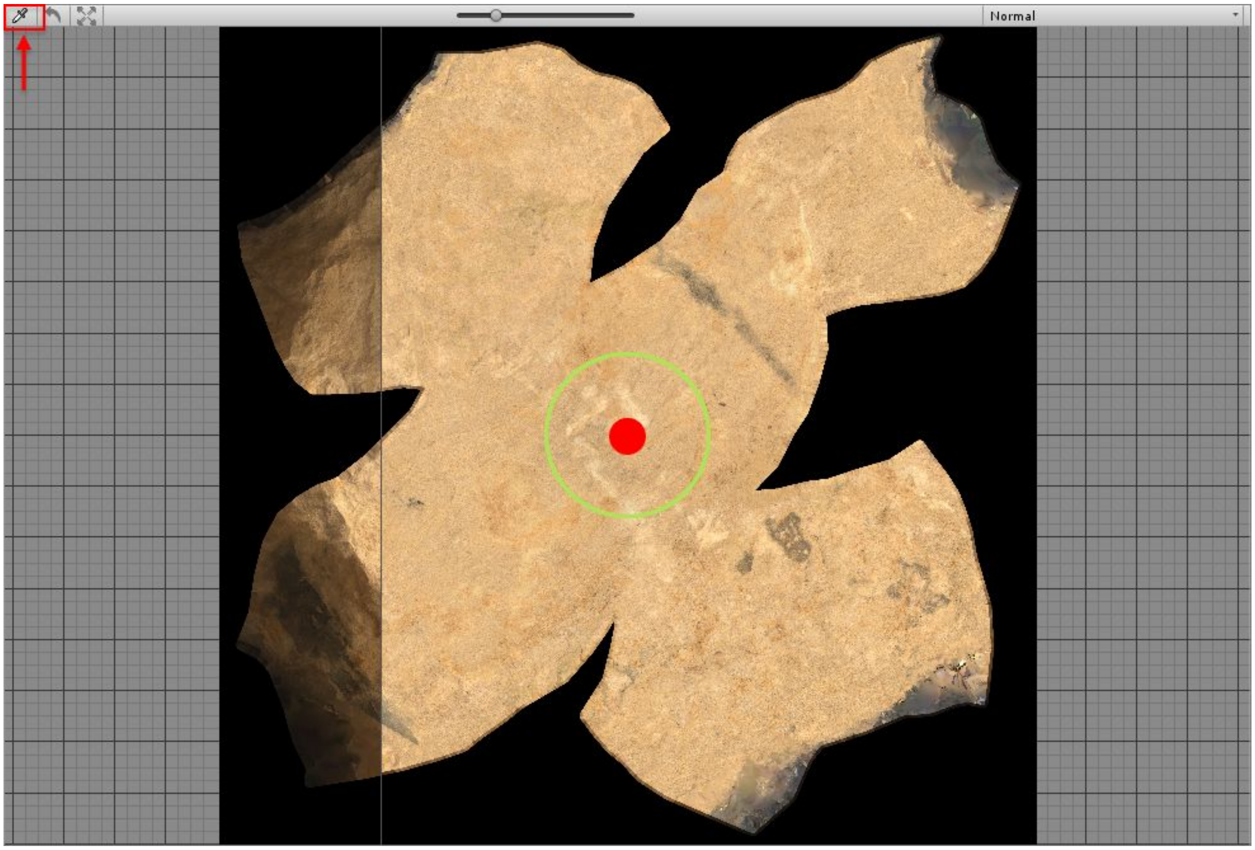
White balance and exposure in the reference zone

If the texture has been color calibrated (see: Photogrammetry Workflow (5.1.3. White balance), a result that is closer to the object's real albedo color appears in the reference zone, which is the calibration area around the color checker.

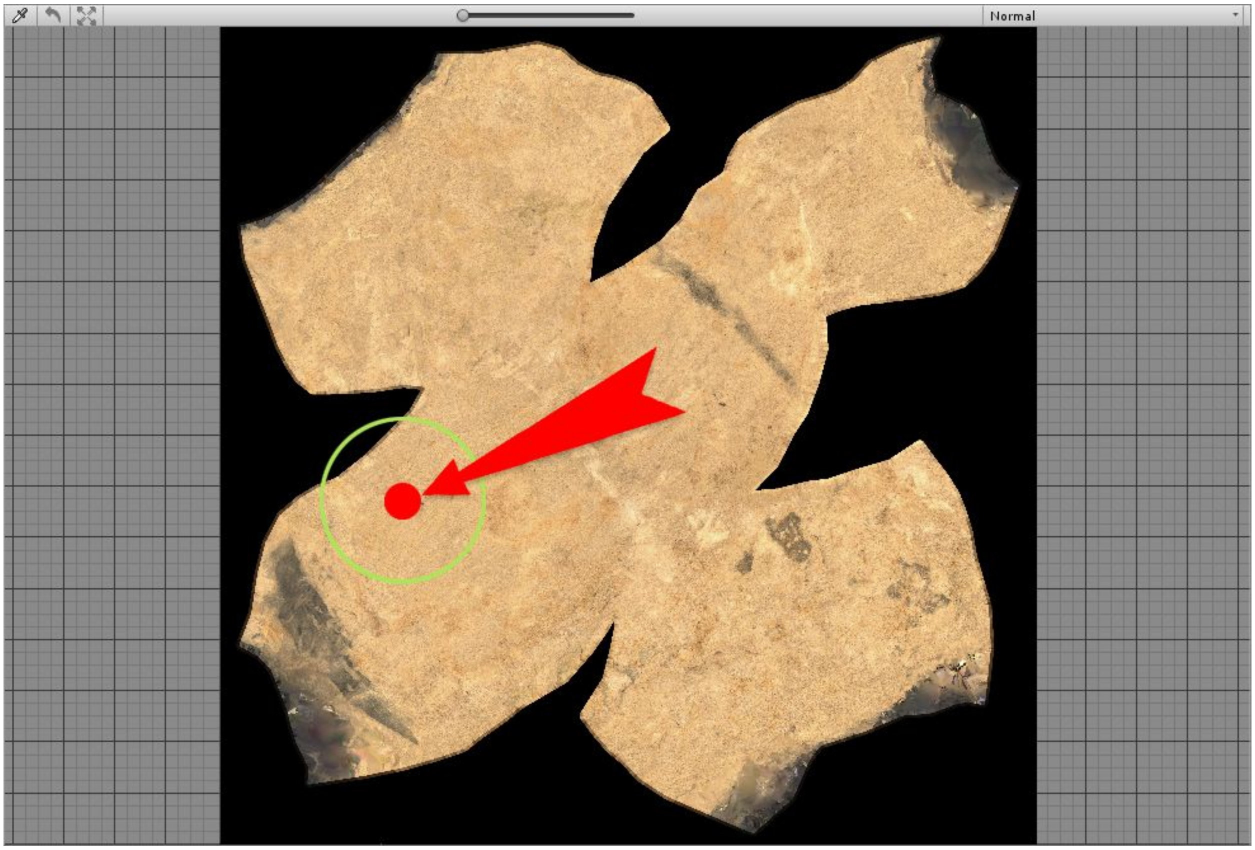


(1) Calibrated photo, (2) Calibrated texture (the red circle marks the calibrated area)

To activate the reference zone, click the picker button in the top left of the preview area. A red dot with a green circle show the reference zone on the texture:



Move the gizmo by left-clicking on the red dot, and place it where the color checker was placed:

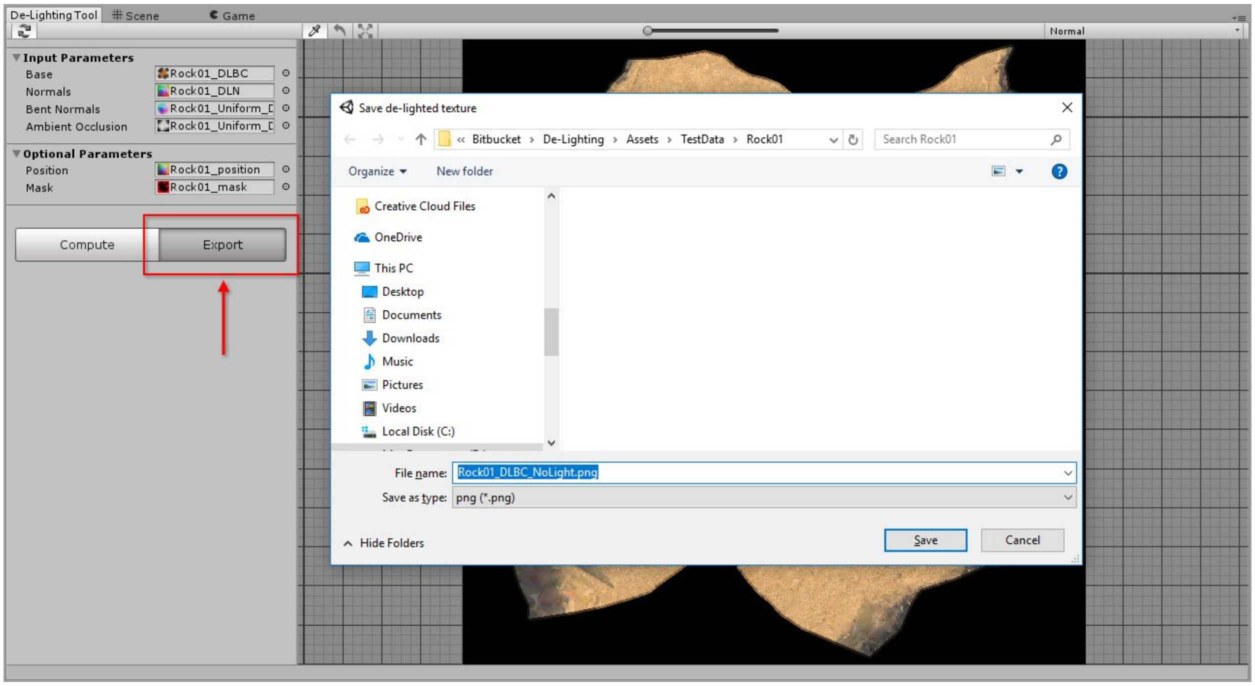


(Here, the compare slider has been set to the far left to only show the de-lighted texture).

Change the size of the green circle by dragging it with the left mouse button. This circle indicates the size of the calibration area. The color and exposure change when you move the gizmo.

Export the result

Simply click the **Export** button in the De-Lighting Tool to save the result as a .png file:



Input Parameters
Base: Rock01_DLBC
Normals: Rock01_DLN
Bent Normals: Rock01_Uniform_D
Ambient Occlusion: Rock01_Uniform_D

Optional Parameters
Position: Rock01_position
Mask: Rock01_mask

Save de-lighted texture

« Bitbucket › De-Lighting › Assets › TestData › Rock01

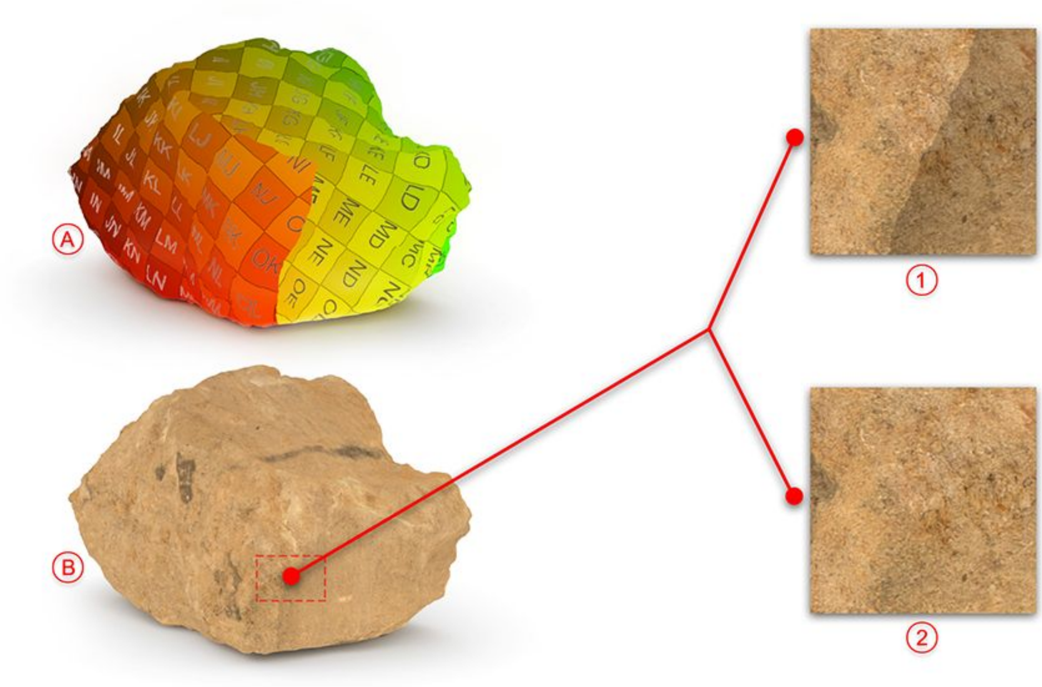
File name: Rock01_DLBC_NoLight.png
Save as type: png (*.png)

Advanced de-lighting

Optional parameters

The position map

As well as 2D space (UV space), you can de-light textures in 3D space by using a position map. This avoids discontinuities that can appear on UV seams during the de-lighting process. Another benefit is that it produces more consistent and more accurate de-lighting results. For advice on generating a position map, see [Photogrammetry Workflow (5.4 Texture baking)](#).



(A) Object UV coordinates, (B) Object with de-lighted texture
(1) Without position map: visible UV seams, (2) With position map: invisible UV seams

Mask maps

A mask map is an RGB texture that you can author with 2D software, like Adobe Photoshop. Each channel value drives a different parameter of the de-lighting process.

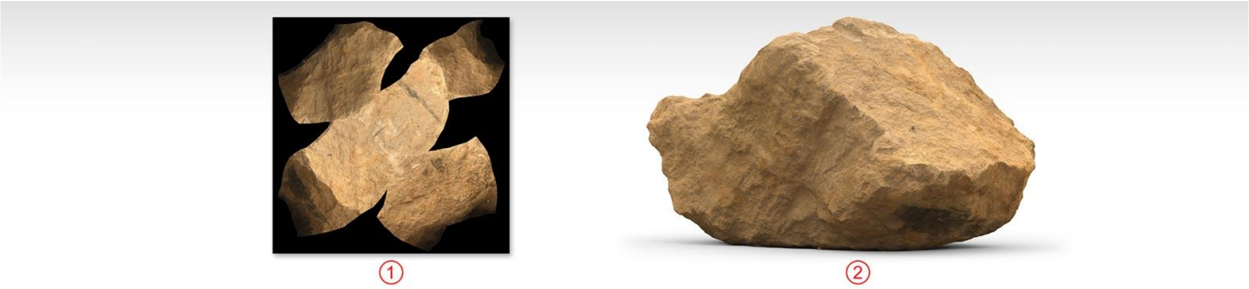
The red channel

The De-Lighting Tool uses the surface itself to reconstruct the environment lighting (that is, a light probe).

Environment lighting is reconstructed based on the pixels that come from the same material. This will be called the reference material. It doesn’t need to be exactly identical but ***very different material parts should be marked in the red channel***.

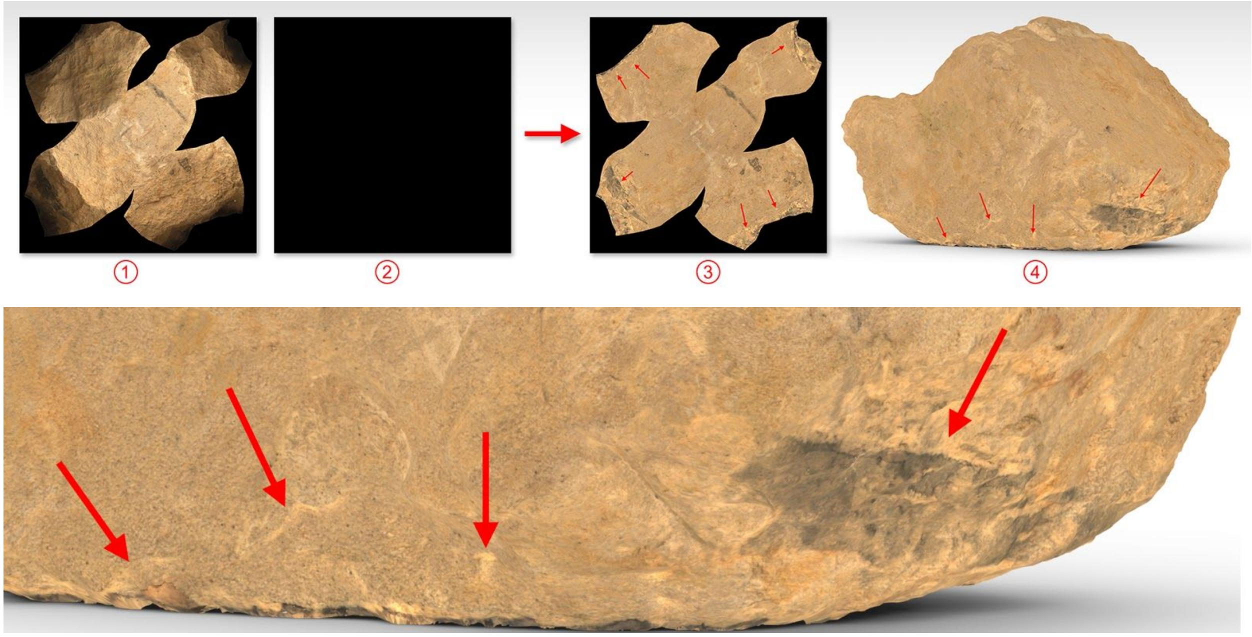
In the rock example, most of the appearance comes from a very similar rock material but a few black parts are paint material and not rock. In the wild, many materials like snow, sand or even moss can cover objects, and they often orient upward due to gravity or light. Such materials can perturb the de-lighting process, so you should mark them in the red channel of the mask. You should also mark any parts that have been incorrectly reconstructed by the photogrammetry process.

The following example shows a texture that needs de-lighting and the 3D mesh that will use it:



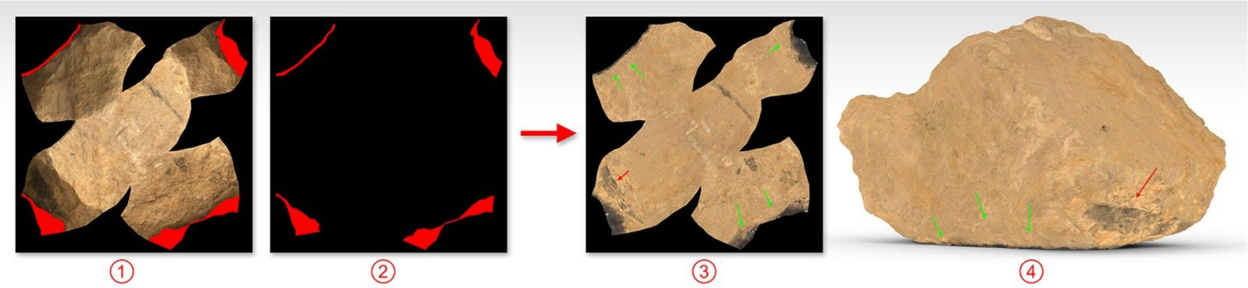
(1) Original lit texture, (2) Lit texture applied onto the mesh

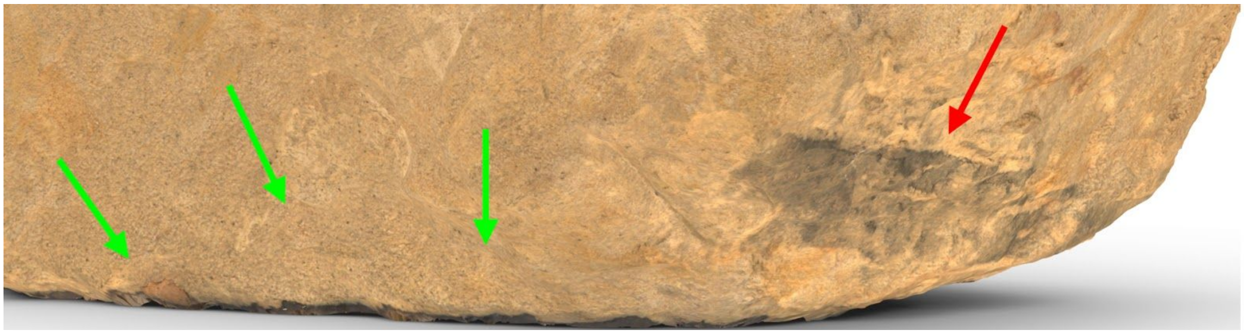
To begin with, the de-lighting process is done without using a mask (or an empty mask). Red arrows show where artifacts have appeared:



(1) Original lit texture, (2) Empty mask, (3) De-lighted texture, (4) De-lighted texture applied onto the mesh

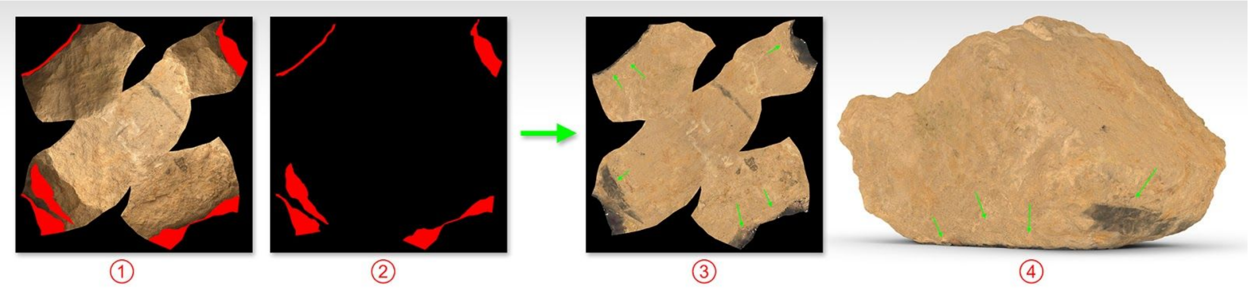
Artifacts appearing like this mean you need to edit the red channel of the mask. Firstly, you need to mark the badly reconstructed parts of the lit texture in the red channel. Most of the arrows will turn green, which means that the artifacts have disappeared. In this case, those bad parts were responsible of nearly all the de-lighting problems. But there is still one red arrow near a black painted area of the rock:





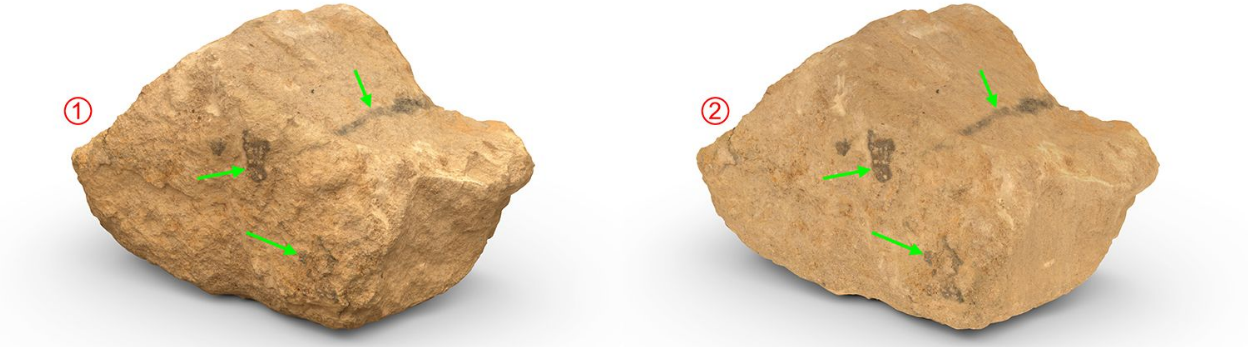
(1) Preview of the authored mask over the original texture, (2) Authored mask, (3) De-lighted texture, (4) De-lighted texture applied onto the mesh

The large black paint mark is very different from the rest of the object, so you can treat is as a very different material. Mark the black part in the red channel (in the bottom left of the mask), so that the tool treats it as a non-reference material. Now, all arrows are green, and the artifacts are gone:



(1) Preview of the authored mask over the original texture, (2) Authored mask, (3) De-lighted texture, (4) De-lighted texture applied onto the mesh

You don't need to mark all of the parts that are small or slightly different. Iterating on mask authoring is a good way to correct artifacts, one after another. The picture below shows the de-lighted result **without** using a mask. On this side of the rock, there is almost no visible artifacts.



*(1) Original texture - applied on the mesh, (2) De-lighted texture **without** mask - applied on the mesh*

If non-reference materials are more than 30% of the pixels, you may need to do the de-lighting in two or more passes. The first pass uses the red channel, as described above, and the second pass uses an inverted red channel. You can then export two results and recombine them in 2D software, like Photoshop.

The green channel

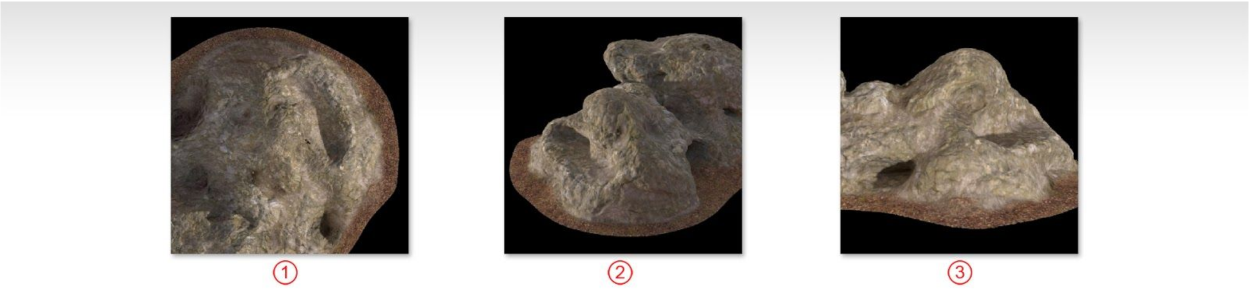
The reconstructed environment lighting is stored in a spatial structure that separates global and local lighting.

Local lighting comes from nearby objects, such as the ground, or from the bounce of light on the object itself (self global illumination). Everything else is considered to be global lighting. The De-Lighting Tool uses the ambient occlusion value to estimate whether it needs to use local or global lighting. More occlusion means more local lighting, and less occlusion means more global lighting.

Sometimes ambient occlusion is not enough to choose between local and global lighting.
The green channel forces the use of local lighting in a specific area of the texture.

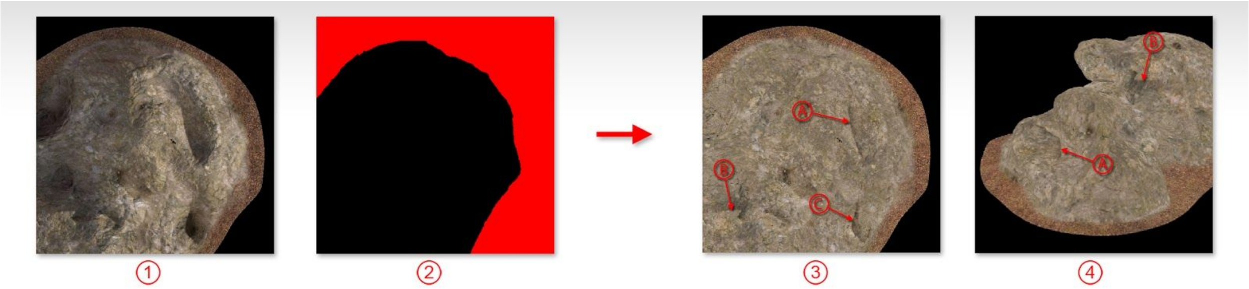
Note: Local de-lighting is less accurate than global de-lighting at separating light color and material albedo. This means that increasing the use of local de-lighting on too much area of the image can decrease the accuracy of the result.

In a similar way to the previous example, the following example shows a texture that needs de-lighting and the 3D mesh that will use it. For better visualisation, the texture is slightly larger.



(1) Zoomed original lit texture, (2) and (3) Two different views of the original lit texture applied onto the mesh

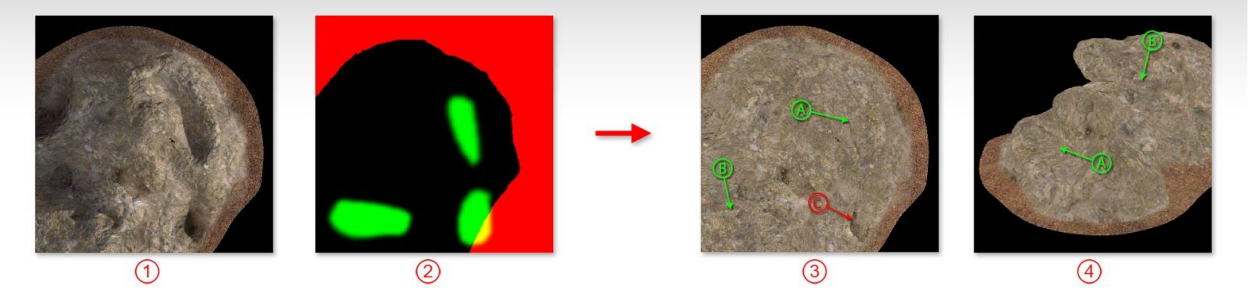
In this example, the previous step has already been done. The red channel of the mask has been edited to mark pine cones and needles as a non-reference material. Despite using the red channel, the result isn't perfect in the areas (A), (B) and (C):



(1) Zoomed original lit texture, (2) Zoomed mask, (3) De-lighted texture, (4) De-lighted texture applied onto the mesh

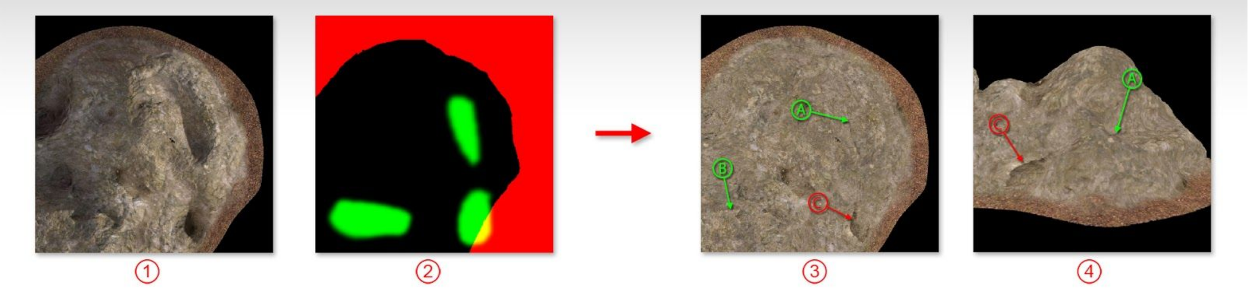
These artifacts are mainly due to cast shadows that come from strong direct lighting. As described in the Photogrammetry Guide (Photogrammetry Workflow - 3.1.3), you should avoid strong direct lighting (such as the Sun) during the capture.

Using the green mask on this kind of artifact can help solve the problem. Artifacts (A) and (B) have almost disappeared:



(1) Zoomed original lit texture, (2) Zoomed mask, (3) De-lighted texture, (4) De-lighted texture applied onto the mesh

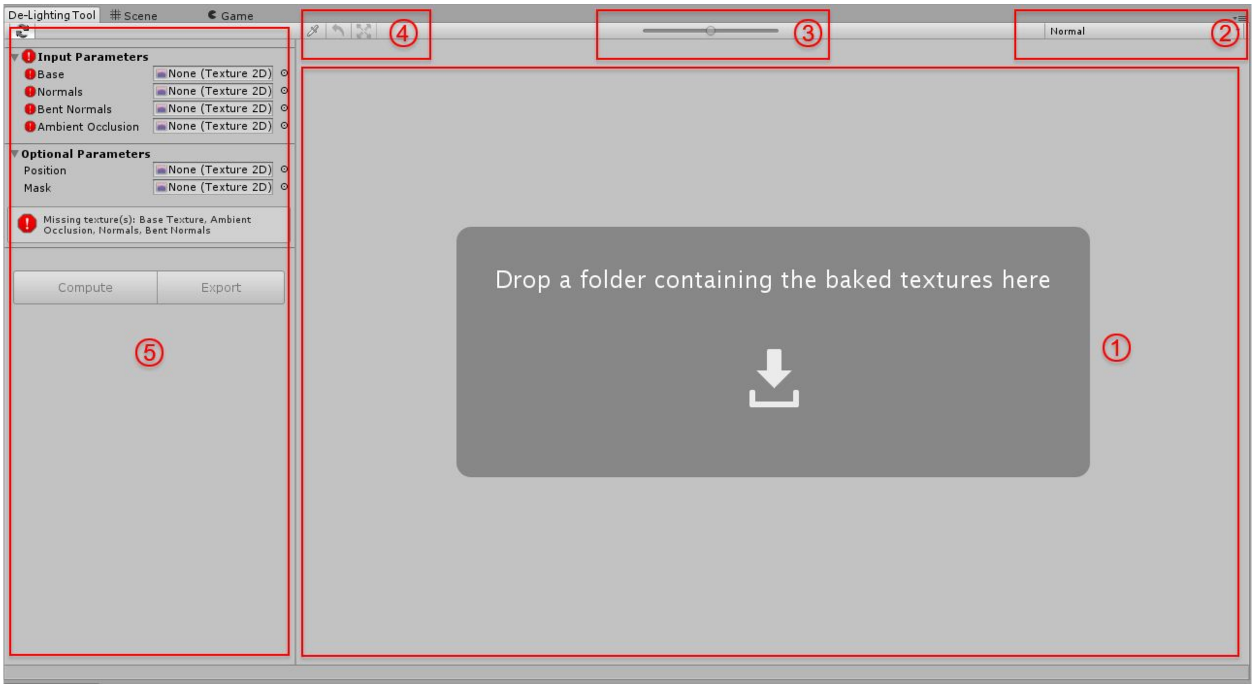
However, artifact (C) still remains. Most of the time, strong cast shadows won't be removed from the texture. You should remove artifacts like this in 2D software like Adobe Photoshop. For better visualisation of area (C), picture (4) below shows the other side of the 3D mesh (other pictures are the same as those above):



(1) Zoomed original lit texture, (2) Zoomed mask, (3) De-lighted texture, (4) De-lighted texture applied onto the mesh

Advanced mode

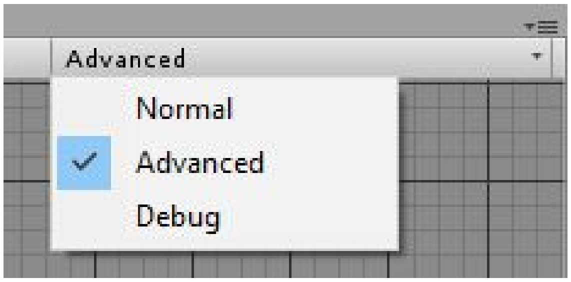
View modes display and hide interface elements to suit your needs. Zone (2) contains the view mode drop-down list:



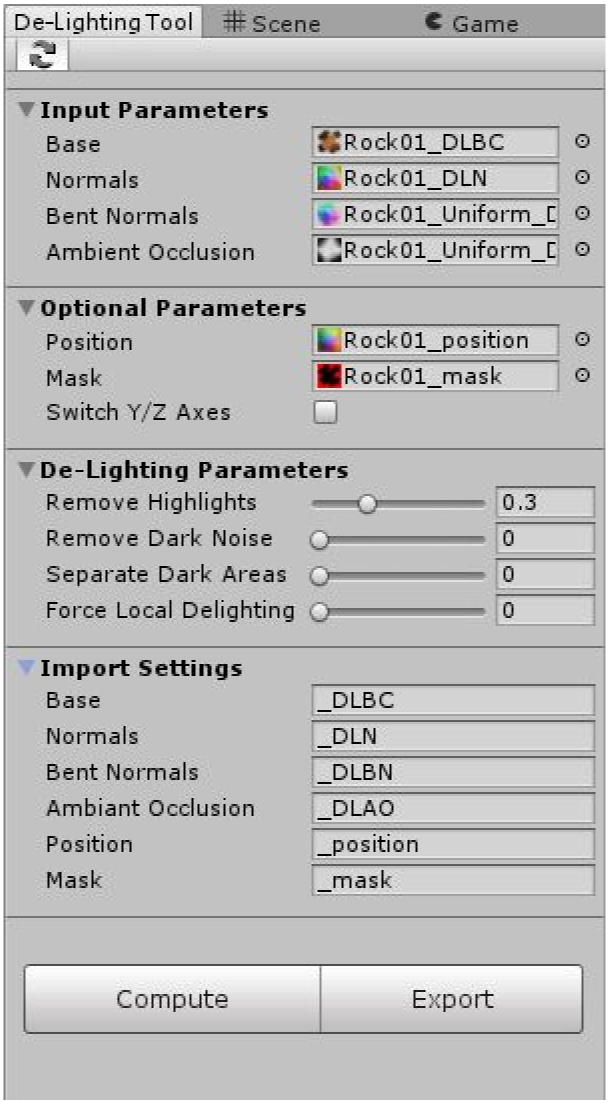
Base: None (Texture 2D)
Normals: None (Texture 2D)
Bent Normals: None (Texture 2D)
Ambient Occlusion: None (Texture 2D)
Position: None (Texture 2D)
Mask: None (Texture 2D)

Missing texture(s): Base Texture, Ambient Occlusion, Normals, Bent Normals

You can switch to **Advanced** mode by selecting from the view mode list:



When the view mode is switched to **Advanced**, the following interface elements and features are available:



▼ Input Parameters

Base	Rock01_DLBC
Normals	Rock01_DLN
Bent Normals	Rock01_Uniform_D
Ambient Occlusion	Rock01_Uniform_D

▼ Optional Parameters

Position	Rock01_position
Mask	Rock01_mask
Switch Y/Z Axes	□

▼ De-Lighting Parameters

Remove Highlights	0.3
Remove Dark Noise	0
Separate Dark Areas	0
Force Local Delighting	0

▼ Import Settings

Base	_DLBC
Normals	_DLN
Bent Normals	_DLBN
Ambiant Occlusion	_DLAO
Position	_position
Mask	_mask

Optional Parameters

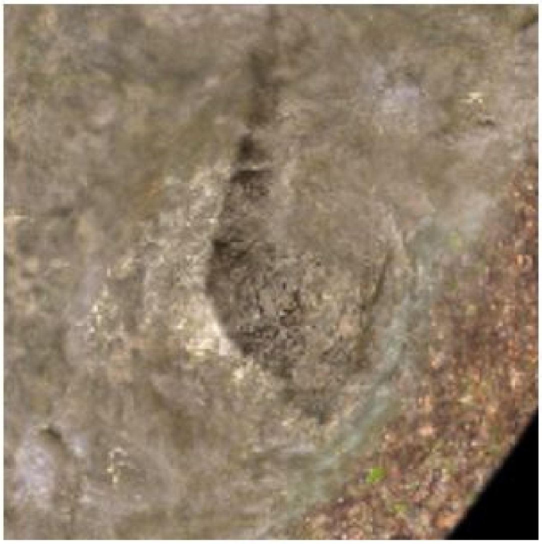
In **Optional Parameters**, a new **Switch Y/Z Axes** checkbox allows the tool to be consistent with the normal data provided (normal map, bent normal map).

De-Lighting Parameters

This section exposes parameters that are useful to correct artifacts.

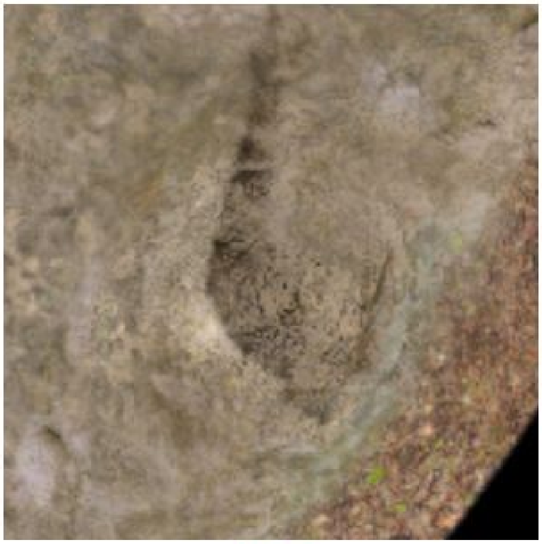
Remove Highlights

This controls highlight detection and processes de-lighting repair on those detected parts. The slider goes from 0 to 1, where 1 is maximum highlight removal. Highlights can appear for many different reasons, and most of the time a 0.3 value is enough, which is why it's the default value. Increasing the value too much makes the de-lighting process less accurate. It's best to try and keep it as close as possible to 0.



1

(1) Remove Highlights value = 0

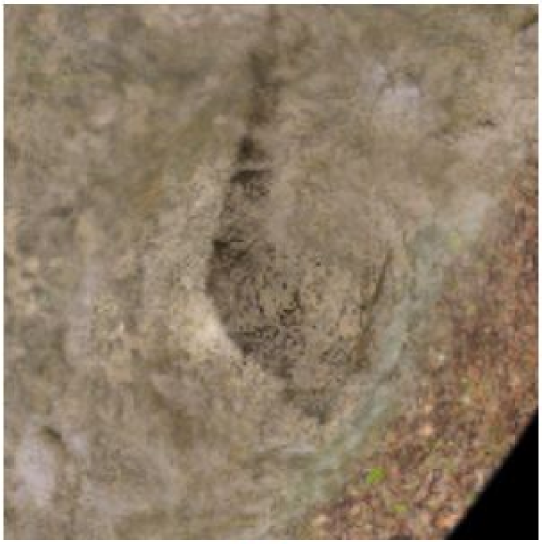


2

(2) Remove Highlights value = 0.3

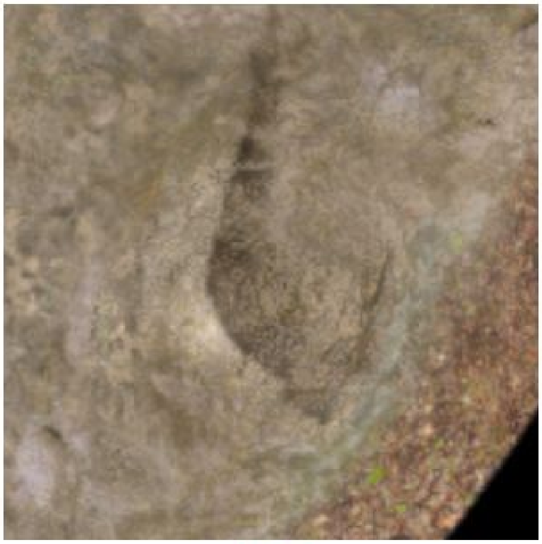
Remove Dark Noise

This removes the small black dots that can, like highlights, appear for many reasons. The slider goes from 0 to 1, where 1 is maximum dark noise removal. Use it the same way as the **Remove Highlights** slider.



1

(1) Remove Dark Noise value = 0



2

(2) Remove Dark Noise value = 0.3

Separate Dark Areas

Photogrammetry is less accurate in dark areas, and all the information in those parts (normal, bent normal, ambient occlusion) won't be as consistent as others. For better results, you could separate the dark parts from the rest of the texture. A slider value of 0.2 separates all the pixels that have a luminance value lower than 0.2 (if we consider that maximum luminance is 1). It could be considered as a procedural mask of dark areas.

Force Local De-Lighting

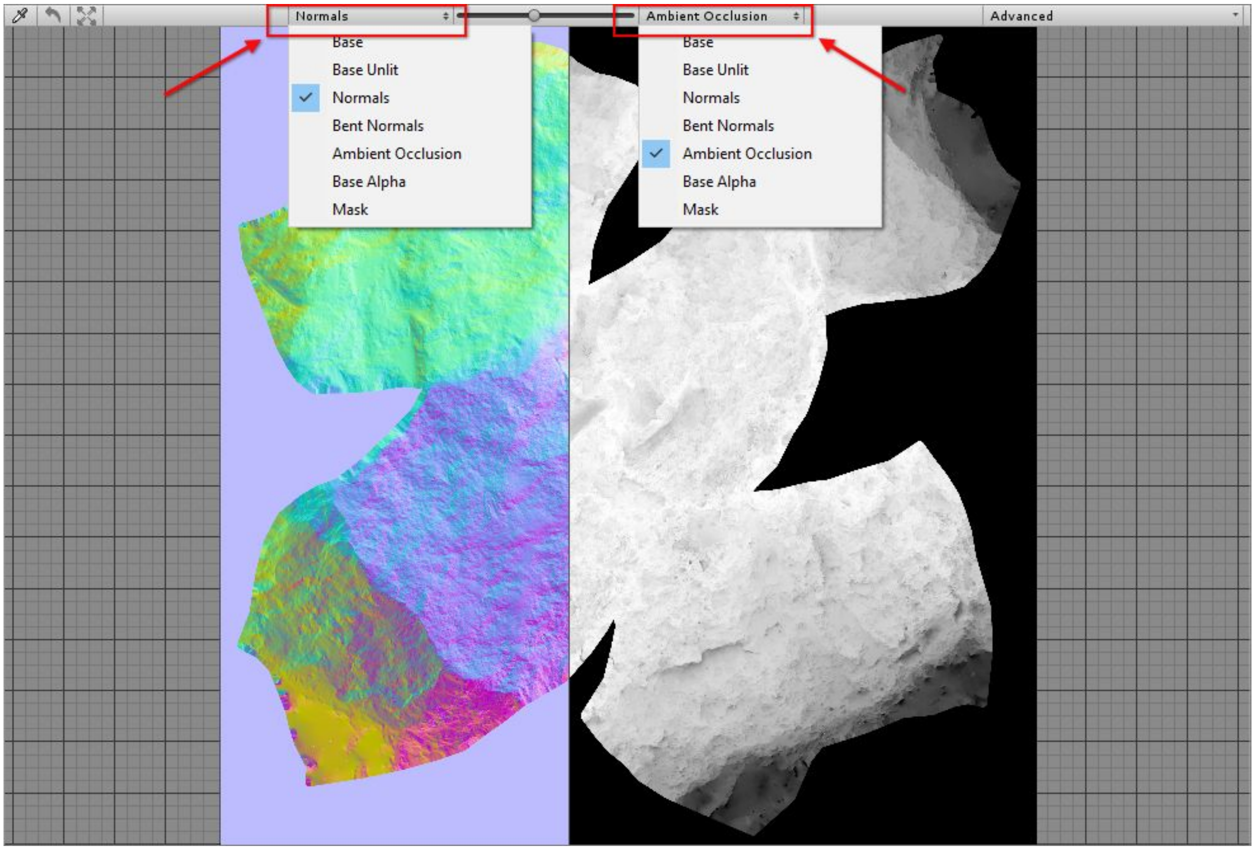
This slider controls the same parameter as the green channel of the mask. As described in the green channel section, when the slider goes from 0 to 1, the tool uses more local lighting information instead of global lighting. Pushing the value to 1 can make the texture appear to be more de-lighted, but in reality the result is less accurate.
This slider and the green channel value are cumulative. So, if the green channel is half grey and the slider is 0.5, the parameter will be equal to 1.

Import Settings

As described in the Import data section, the **Import Settings** expose the name suffix that you should to make the tool correctly import and automatically recognize textures (for example, during folder drag and drop). Changing those suffixes generates new import rules. If there are already import rules (AssetPostprocessor scripts), it's important that you check the suffixes are not in conflict with those used by the De-Lighting Tool.

Comparison channels

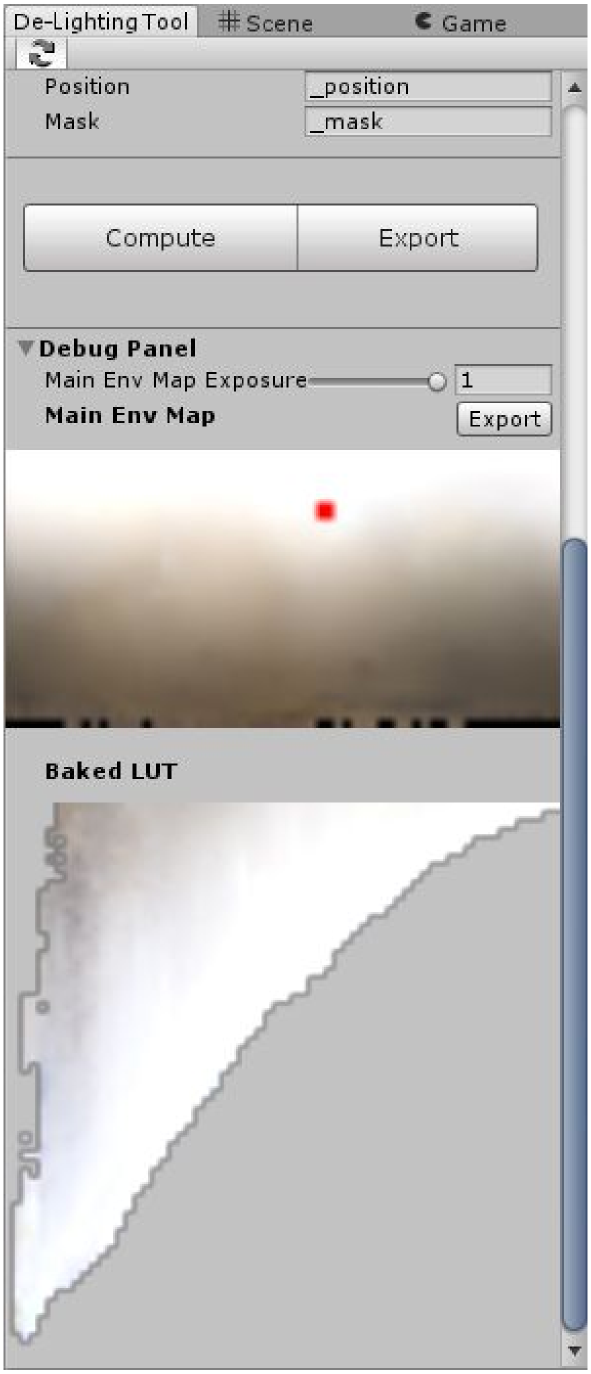
Switching to **Advanced Mode** also adds new comparison channels to the viewport:



On both sides of the comparison slider, there are two dropdown lists that control which channel to display on the left and right side of the viewport splitter. By default, the left side is set to **Base** (the original lit texture) and the right side is set to **Base Unlit** (the de-lighted texture). You can also switch to view **Normals**, **Bent Normals**, **Ambient Occlusion**, **Base Alpha** (the alpha channel of the original lit texture) or **Mask** on either side of the splitter.

Debug mode

If you switch the view mode to **Debug** mode, the inspector will look like this:



Position _position
Mask _mask
Main Env Map Exposure 1

The features available in this mode are designed to help you understand what went right or wrong in the de-lighting process.

Main Env Map section

This section displays the reconstructed global environment lighting as a 64x32 pixel latitude-longitude image. This picture is HDR and can be exported to an .exr file by clicking the **Export** button.

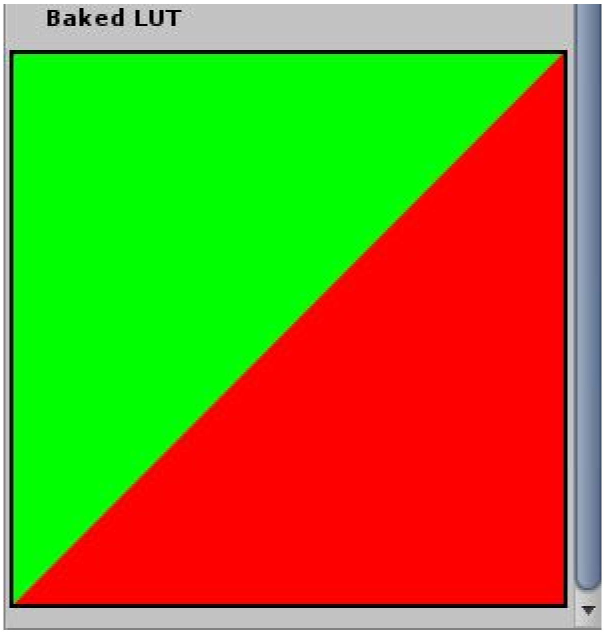
The **Main Env Map Exposure** slider allows you to change the preview exposure of the **Main Env Map**.

The red dot in the **Main Env Map** helps you debug the part of the map the reference zone is pointing at. When you activate the reference zone, the pixels of the normal map under the gizmo become average latitude-longitude coordinates. A red dot appears on the Env Map at those coordinates to indicate a white light reference.

Neither the exposure shift value and the red dot are displayed on the exported Env Map.

Baked LUT (lookup table) section

This section displays the lookup table that is used for ambient lighting and self global illumination. This LUT is specific to every object and it is generated each time a recompute is required. Its shape can give useful information to help determine whether the bent normal map and ambient occlusion map are consistent.



90% of the pixels should be in the green area. They don’t have to cover the whole area, they just need to be inside it. If there are a lot of pixels in the red area, there is probably a problem with the ambient occlusion map and/or bent normal map. In which case, you need to verify that the maps have been baked and imported with the correct settings.

Note: Because the various types of baking software don’t work exactly the same, you should use the same software application to bake all of your maps.

Command line

You can launch the De-Lighting Tool via the command line. Here are some examples:

De-light a texture in a folder:

```
.\Delighting.bat -output test.png -inputFolder Assets\TestData\Rock01
```

De-light multiple textures:

```
.\Delighting.bat -output test.png -base
Assets\TestData\Rock01\Rock01_LRBC.tga -normals
Assets\TestData\Rock01\Rock01_LRN.tga -bentNormals
Assets\TestData\Rock01\Rock01_LRNB.tga -ao
Assets\TestData\Rock01\Rock01_LRAO.tga -position
Assets\TestData\Rock01\rock01_position.exr
```

De-light textures using a batch script:

```
CALL .\Delighting.bat -output test.png -inputFolder Assets\TestData\Rock01

You can define some parameters:
-separateDarkAreas <float>
-forceLocalDelighting <float>
-removeHighlights <float>
-removeDarkNoise <float>
-switchYZ (true|false)
```