



Photogrammetry Workflow

Contents

1. Overview
 - 1.1. Introduction
 - 1.2. Game photogrammetry workflow overview
2. Equipment and Software
 - 2.1. Equipment
 - 2.2. Software
3. Acquisition
 - 3.1. The 3 W's: What, Where and When?
 - 3.1.1. What ?
 - 3.1.2. Where?
 - 3.1.3. When?
 - 3.2. Video or photos ?
 - 3.3. Camera setting
4. Capture process
 - 4.1. How to capture an object (except small ones)
 - 4.1.1. Assets selection
 - 4.1.2. Shoot
 - 4.2. How to capture a surface
 - 4.2.1. Prepare the area
 - 4.2.2. Shoot
 - 4.2.3. Detail textures
 - 4.3. How to capture a small object
 - 4.3.1. Set up a support
 - 4.3.2. Shoot
 - 4.4. How to capture the diffuse albedo only
 - 4.5. How to capture foliage
5. Processing
 - 5.1. Images processing - DCraw & Photoshop
 - 5.1.1. Determine photo set required for reconstruction
 - 5.1.2. Convert RAW to TIFF
 - 5.1.3. White balance

- 5.2. Reconstruction - Reality Capture
 - 5.2.1. Add all images
 - 5.2.2. Align images
 - 5.2.3. Check the components
 - 5.2.4. Frames from video
 - 5.2.5. Control points
 - 5.2.6. Set a scale
 - 5.2.7. Create the model
 - 5.2.8. Colorize or texture
 - 5.2.9. Simplify
 - 5.2.10. Mesh export
- 5.3. Create a low resolution mesh for baking - 3dsMax
 - 5.3.1. Tileable surface
 - 5.3.2. Detail material
 - 5.3.3. Object
 - 5.3.3.1. Geometry
 - 5.3.3.2. UV layout
- 5.4. Textures baking - Knald - Xnormal - Designer
 - 5.4.1. Knald v1.2.1 workflow
 - 5.4.2. Xnormal v3.19.3 workflow
 - 5.4.3. Substance Designer 6
- 5.5. Remove lighting - Unity - Photoshop
 - 5.5.1. Unity
 - 5.5.2. Photoshop
- 5.6. Tileable material - Artomatix - Substance painter
 - 5.6.1. Artomatix
 - 5.6.2. Substance painter
- 5.7. Create a game asset - 3dsMax
- 5.8. Foliage - Photoshop
 - 5.8.1. Base color
 - 5.8.2. Normal map

1. Overview

1.1. Introduction

What is photogrammetry?

Photogrammetry is the process of authoring a digital asset using multiple photos of the original real-world object. There are different uses for photogrammetry, and the workflow can vary depending on the context.

This document aims to describe a photogrammetry workflow dedicated to authoring game assets with an affordable budget for a game studio. While it is possible to create extremely high quality assets for non-game usage using photogrammetry, this documents focuses on using photogrammetry for game development, and therefore describes how to get the best results within the typical time and budget constraints of game development.

A selection of equipment and software is listed in the next section, which provide a balance between efficiency, cost and quality.

1.2. Game photogrammetry workflow overview

The workflow is divided in two major parts: Capture and processing.

Capture is about taking photos of an object covering all of it. Processing is about generating the mesh and texture data from the photos.

Capture

The first step is to gather multiple photos of the object from various angles. It is also necessary to gather extra information for reference, scale metric and correctly exposing the photos.

Processing

First the photos need to be calibrated (white balanced), then sent to a reconstruction application. The reconstruction application compares the shapes in the photos (Alignment) to generate a high resolution 3D mesh. The color contained in the pictures is then transferred to either the mesh's vertex colors (Colorize) or to textures used on the surface of the mesh.



Note: The reconstruction software may generate a very high-resolution mesh, which is not suitable for use in real time engines, therefore it's necessary to generate a lower resolution for this purpose. It is possible to create a lower resolution mesh while preserving the details of the high resolution mesh by generating a normal map with baking tools. The baking tools will transfer high frequency information from the high resolution mesh to the low resolution mesh's normal map textures.

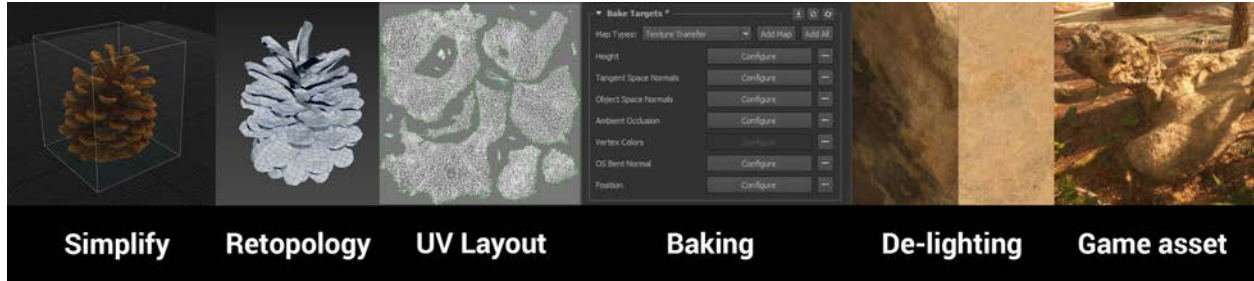
Next, a low resolution mesh for the baking process need to be created: A medium to low resolution mesh is exported from the reconstruction software and modified in a 3D software tool to be use as destination of the baking tools (Retopology and UV layout step).

Baking tools are then used to generate textures.

Then, the textures need to have light removed. This is because the textures are generated from real world photos, which will inevitably have visible lighting and shadows present. To be able to have an asset work in any lighting condition, it is required to remove the lighting (light removal step, a step also know as de-lighting) information baked into generated textures. Sometimes in this step the textures are also made tileable.

Finally the low resolution mesh for baking is converted to a game ready mesh (Mesh with correct orientation, pivot and uv set) in a 3D software application.

Unity Photogrammetry Workflow



Production time

Photogrammetry is a very efficient way to author high resolution meshes and textures, but it does not help out with other processes which are necessary for game mesh preparation (such as the retopology step, UV layout step). Whether or not it saves time compared with traditional 3D asset authoring depends on the complexity of the object in question. Using the photogrammetry process in production is not a simple task as it varies according to the object's complexity and retopology time. It is usually a win but it can be difficult to quantify the time saved.

Here are some rough examples of the potential time saving when using photogrammetry:

Days	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Classic workflow	High Mesh					Texturing			Retopology	UV + baking	Material	Import IG	LOD	
Photogrammetry workflow	Photos	HM + T	Retopology	UV + baking	Material+ Delighting	Import IG	LOD	Time saved						

2. Equipment and Software

2.1. Equipment

The following is a recommended list of equipment to capture good data for high quality photogrammetry. You can also use this as a checklist each time the team travels to a shooting location.

Unity Photogrammetry Workflow



1- Camera: Canon 6D.

Technically any camera can be used for photogrammetry, even a phone camera. However for best quality it is important to have a camera that allows you to capture high resolution sharp images. A full frame camera is recommended.

High resolution also means that fewer photos are required for the reconstruction process.

2- Lens: Sigma 24-70 mm f/2.8 EX-DG HSM

In a natural environment some objects can be inaccessible, too far or too close. It's then easier to have a variable zoom than a fixed focal length lens to capture objects.

The software we recommend for reconstruction (below) works fine with photos of various focal length.

3- CANON ring flash MR-14EX II with rechargeable batteries (2850 mAh).

Ring flash is preferred over cobra flash as it remove most of the light shadows.

4- Memory Card 256 Go (Class 10).

256 Go is a good capacity for a day of capture. Class 10 is needed to minimize the latency.

5- Laptop: Macbook Pro.

The laptop is dedicated to storage and to check data during a long trip.

6- Color checker passport photo (X-rite).

The color checker should be use at each capture to white balance the photos.

7- HOYA Polarizer filter PRO 1 DIGITAL 67 mm.

Lens polarizer is used to remove reflection when trying to capture diffuse albedo.

8- Linear polarizer filter film.

The linear polarizer filter film is a small coating that is put on the flash to polarize the lighting. It allows you to remove reflection for better diffuse albedo capture.

Example:

http://polarization.com/polarshop/product_info.php?cPath=21&products_id=28&osCsid=j6hsje6ll1us9n11jggu4hle05

9- Secondary camera battery.

The photogrammetry process requires a lot of photos and a second battery is recommended for a full day of capture.

10- Bag.

11- Lenspen (Lens cleaner tools).

12- LASTOLITE Background support kit support H 3.52 x L 3m.

13 - PHOCUSLINE Black background 3 m x 3 m.

14- Lastolite Cubelite.

15 - GODOX reflector 180 x 120 cm.

16- Markers like tent pegs and chalk.

17- GOPro

18- DJI drone Phantom 4 RTF

2.2. Software



1- **DCRaw**: Convert RAW to TIFF

Windows: <http://www.centrostudiprogressofotografico.it/en/dcraw/>

Mac OS: <http://www.cybercom.net/~dcoffin/dcraw/>

2- **Adobe Photoshop CC**: White balancing, artefacts correction

<http://www.photoshop.com/products/photoshop>

3- **Reality Capture**: Photogrammetry processing software

<https://www.capturingreality.com/>

Note: A good GPU card is recommended and Reality Capture expects an NVidia card for better compatibility with its GPU processing.

4- **3dsMax (or any related 3D software)**: Retopology, UV, LOD

<https://www.autodesk.fr/products/3ds-max/overview>

PLY import/export plugin: <http://www.guruware.at/main/plyImp/index.html>

5- **Instant Mesh**: Retopology.

<https://github.com/wjakob/instant-meshes>

6- **Knald**: Baking textures

<https://www.knaldtech.com/>

Note: A good GPU card is recommended. 8GB of Vram is advised for very high meshes (>400M of polygons)

7- **Xnormal**: Baking textures

<http://www.xnormal.net/downloads.aspx>

8- **Substance designer:** Baking textures

<https://www.allegorithmic.com/products/substance-designer>

9- **Unity Light Removal Tool:** remove lighting from albedo baking.

<https://github.com/Unity-Technologies/DeLightingTool>

10- **Artomatix:** automatic tiling material.

<https://artomatix.com/>

11- **Substance painter:** Manual tiling material.

<https://www.allegorithmic.com/products/substance-painter>

12- **VLC media player:** Video to frames

<http://www.videolan.org/index.fr.html>

Recommended configuration for computer:

32 GB of RAM, an Nvidia video card with 4GB of Vram, and a hard drive of 2TB.

Unity team use CPU intel i7-5960x, 64 GB of RAM, 980 GTX, 2TB hard drive.

3. Acquisition

Before organizing a photogrammetry travel trip it is recommended to be familiar with the process presented in this tutorial and have good camera practice.

3.1. The 3 W's: What, Where and When?

Prepare a photogrammetry travel trip by answering the 3 W's.

3.1.1. What?

The first step is to know the kind of environment that will be used in the game and start to gather a list of assets that need to be produced for a game level. It is necessary to check what assets are compatible with the photogrammetry process and which ones really require it.

Understanding how reconstruction software works will help you to understand what kind of surfaces best fit a photogrammetry process. To reconstruct geometry from a set of pictures, a reconstruction software recovers the 3D world position of the various pixels of the pictures based on camera information, then if there is spatial group of pixels with enough similarities it makes a point. The result of this process is called a "point cloud".

There are some types of surface which are bad candidates for photogrammetry. If the subject is moving (for example, foliage in the wind), or if it is shiny or has strong reflections (such as metallic surfaces, wet surfaces), if it is liquid, or if it is transparent (such as glass surfaces), then the reconstruction software can't match pixel colours across all the photos of the object. It can be difficult or impossible to get good enough result with such objects. The image below shows some examples of subjects that are not suitable for photogrammetry:



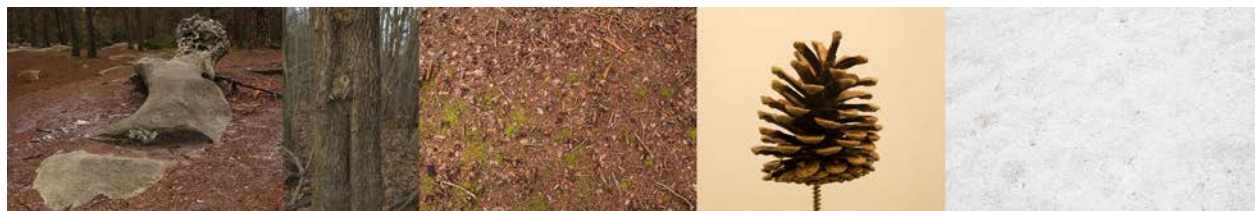
In addition, if the object has large flat areas of color, the reconstruction software doesn't have enough

information to distinguish groups of pixels similarities as they all have the same color.

Note: For flat colored objects, it is possible to help the reconstruction software by painting on the objects or by projecting fixed light patterns. However in this case, only the geometry is extracted and the textures need to be produced separately. The example below shows on the left a flat-colored asset that is too difficult to reconstruct correctly. The result is that multiple separate point clouds are created which is not desired. On the right, same flat object has had patterns of paint applied to it. which gives a good result - only one point cloud is created:



Photogrammetry works very well with a static, solid, rough asset. The image below shows some examples of subjects that are well-suited to photogrammetry:



It is important to consider the production cost when creating an asset with photogrammetry. Even if an

object matches the requirements for a photogrammetry process it doesn't mean that it is the best way to go. Some simple or manufactured objects are often more efficient to author with traditional 3D modeling techniques. Below is an example of a mug that is faster to model in 3D software, as it is flat colored simple shape, than using photogrammetry.



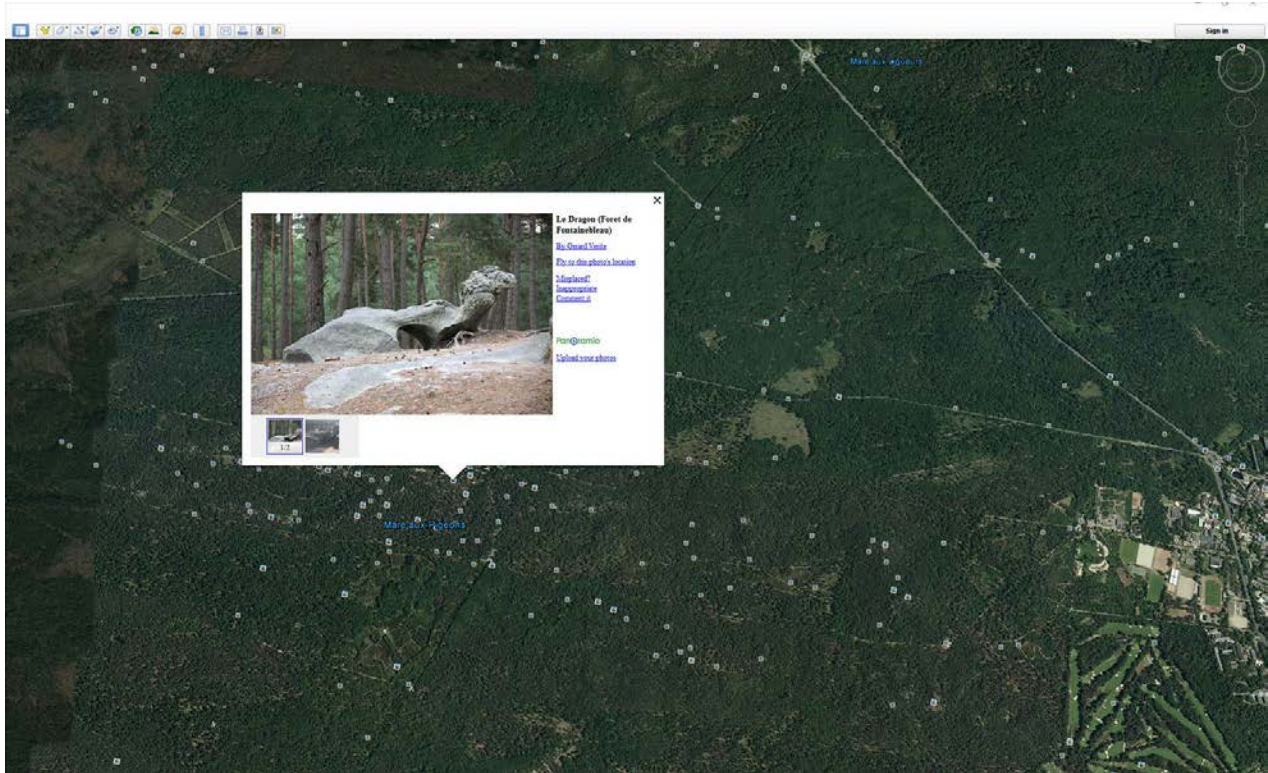
Pay attention to complex objects, especially with fine parts like little roots. The processing time of a complex asset can be really expensive. Below is an example of an asset that will take a long time to process:



3.1.2. Where?

Travelling to various locations to photograph objects takes time, as well as the deployment of the various capture devices. It is necessary to carefully plan the shooting trip to minimize the amount of travel and prevent wasted time. Good areas are where the maximum number of assets can be captured at the same location. If you are capturing environmental geography, Google Earth can be a great help in finding the right location.

Unity Photogrammetry Workflow



Note: There are private or protected properties everywhere that must be respected. Many areas are also forbidden (like military areas) and major cities tend to forbid the use of drones. Be careful about the local laws and rules of the trip location. Moreover, have in mind that when the subject is in a public space, it can be difficult to avoid people moving in front of the camera.



Look for quiet areas with enough space to capture the subject.

3.1.3. When?

When shooting in outdoor environments, the **weather** is the most important factor to take into account. Weather has a large impact on the photogrammetry process.

Rain and snow are the worst situations. Surfaces can be moving (water sliding on them, snow falling down), reflection is stronger and the lens is modified by water drops or snowflakes. It is recommend to cancel the Shoot in these conditions.

Wind is another element difficult to deal with. The wind can cause trees, grass, dust, and sand to move position in your images, which can result in a poor reconstruction result. For small objects (wood pieces, branches, pine cones) you can collect them and bring them back to an indoor area to photograph them in a safer place. However, some objects that are susceptible to moving in the wind can only be captured on-site, such as trees, bushes and grass. For these you will need to make sure you photograph them on a day with no wind so the objects remain completely still.

The sun produces highlights and strong directional shadows. A situation that must be avoided as it is difficult to have a good exposition with resulting high exposure range. In a high-contrast photo, noise tends to appear in strong shadows. Moreover, highlights or directional shadows are difficult to remove from the generated diffuse albedo texture, a step that is required to be able to relight a captured object. Below is an example of a location with strong directional lighting coming from the sun through the leaves. The generated texture from this location exhibits a lot of lighting and strong occlusion that are difficult to remove.



Unity Photogrammetry Workflow

Shooting on an overcast day without rain provides a constant illumination and very soft shadows which makes it easy to remove light from the image later. However the luminosity is low and the ISO should be raised (See camera setting section later).



The preferred conditions for the Shoot is during sunny days but with the use of an occluder to isolate the subject from the directional lighting. It produces softer photos similar to an overcast sky, but with a better luminosity level for the capture.



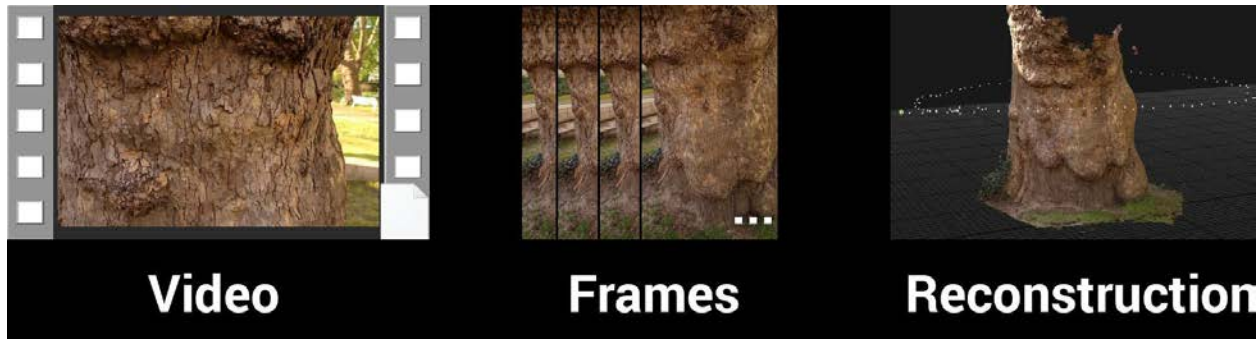
Lastly, having stable lighting is important to have consistent pictures for the reconstruction software. Fast moving clouds passing in front of the sun can become a problem.

The second most important factor is the **season**. Seasonal lighting differences influence the appearance of organic/living objects like leaves, trees or flowers (particularly in autumn). Also, EV range varies

seasonally and in winter days the EV range available for capturing subjects is narrower. Available daylight also varies with the seasons.

3.2. Video or photos?

The photogrammetry process relies on photographic source images. However, all captures could be done with a GoPro or any similar 4K video device, from which still frames can be extracted. Because these kinds of video recording devices are small, they allow you to cover all parts of an object very quickly and can be very close to the object, in ways that can be more difficult for a larger camera.



Unfortunately the GoPro and other similar devices tend to be very sensitive to high lighting conditions, which results in bloom appearing around bright areas on the footage. They also tend to not have manual controls for ISO and aperture. The captured video can also contain more motion blur due to hand shaking or fast movements:



It is recommended to use a full frame still camera to get best results.

A video recording device is efficient when used to capture the geometry only (material being captured with another method (See [How to capture a surface](#)) or created procedurally) or when use for coarse layout and quick test (See [Huge object](#)). This is why it is recommended to use a GoPro as a backup rather than main device (See [Shoot](#)).

3.3. Camera setting

The best results during reconstruction are obtained from sharp pictures with no clamping of the color range. Sharpness is a very important factor here.

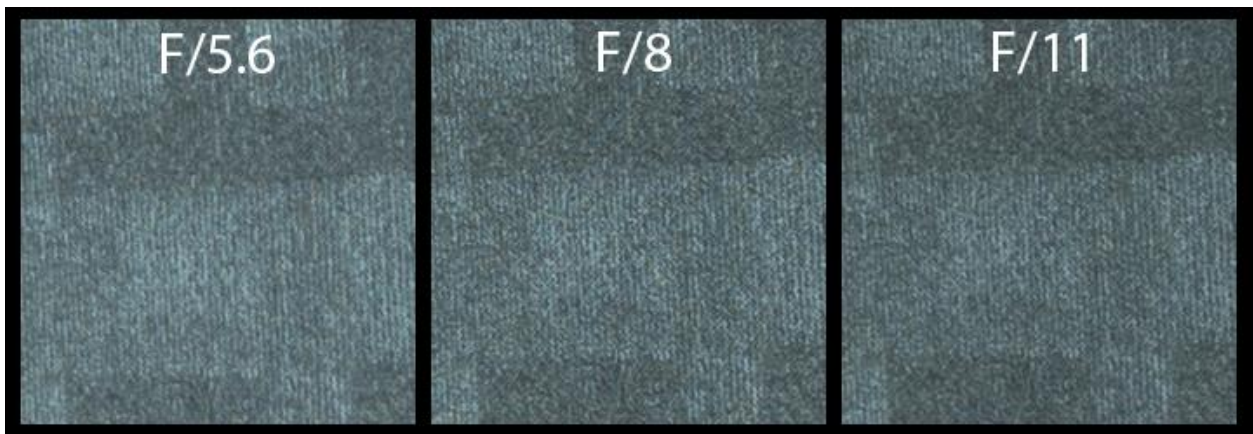


1- Example of bad exposure. Dark or clamped areas provide bad reconstruction.

2- Example of a bad pictures with too much defocus, aperture of f/2.8. The border of the pine cone is totally blurry due to the depth of field and the reconstruction is a fail.

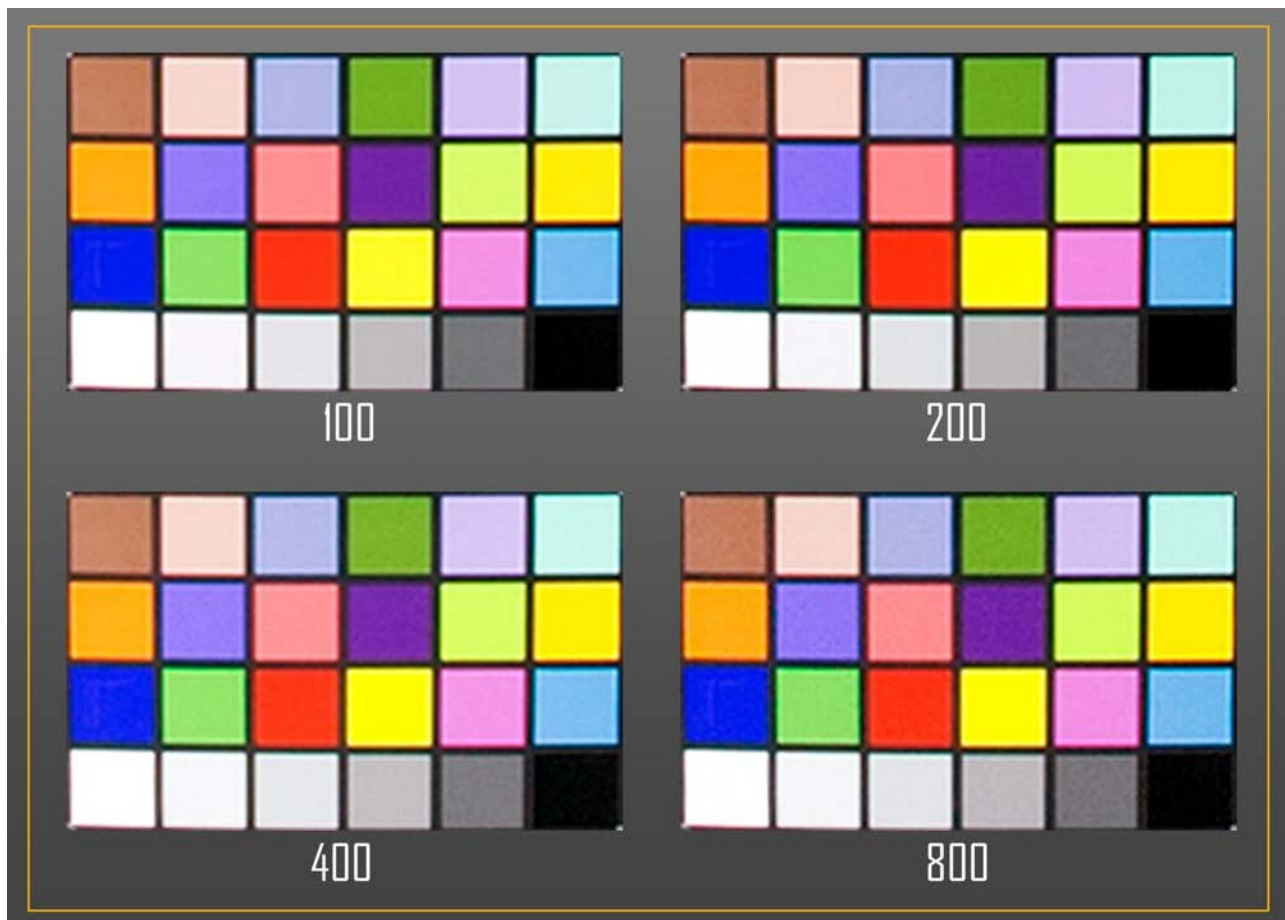
3- Example of correct focus and exposure.

The default recommended settings aim to have the sharpest image possible: an aperture of **f8** is used with the autofocus set to on as highlighted by this comparison of various aperture:



If you are shooting outdoors, lighting conditions can vary due to the sun or clouds moving. Therefore, to save time, the use of a tripod is not recommended. Also don't spend a day to capture a rock! The **shutter speed** must be at **1/160** or less to accommodate for this fast hand shooting while maintaining a sharp image.

ISO should be set to **100 or lower** to avoid noise especially in darker areas which are not good for the reconstruction software.



In practice, in the case of low lighting conditions, you should first increase the ISO, then the aperture. It can be necessary to adjust the aperture a bit to counterbalance the noise produced by high ISO. With a tripod, it is possible to play with the shutter speed - however using a tripod can add excessive time between shots as described above.

Set the output format to RAW. RAW format has higher accuracy for the reconstruction and allows better white balance in the pictures.

Before you start the asset capture, the exposure level must be controlled with the camera histogram to not have clamped pixels on the subject and to have a good exposure range. The example below shows on the left an image where the range is too small to do a reconstruction and many pixels are clamped.

The right image represents a good range to process a reconstruction. The subject is represented in the middle of the histogram:



It is important to have an exposure level in the middle of the lighting range because the recommended reconstruction software (Reality Capture) doesn't use the full 14-bit range captured by the camera and photos are converted to 8-bit.

4. Capture process

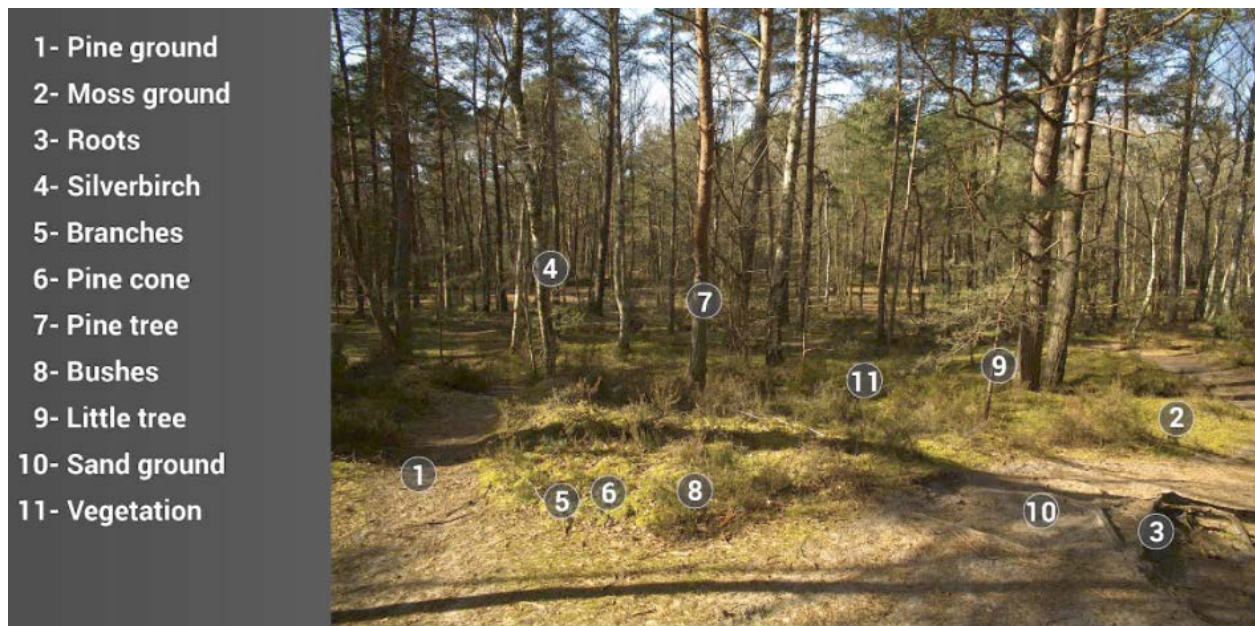
Before travelling to your image capture location, look at the check list. Take care that batteries are charged, lenses are clean and memory cards are empty.

4.1. How to capture an object (except small ones)

4.1.1. Assets selection

Remember that photogrammetry is best suited to objects that are time consuming to produce in 3D sculpting software. Don't use photogrammetry for simple shapes like a cube. Don't try to capture everything in the location, think in term of usable pieces. Select a subset that allows you to reconstruct the environment. Focus on quality instead of quantity of captures.

The example below shows identification of pieces of the environment that will provide a toolkit of elements that will allow you to build a game level with several instanced assets:





Note: Large objects which cannot be fully captured from all angles should be captured in the position they will be used in game. Rotations will show unreconstructed parts. Making such objects reusable for every rotation angle requires artists to re-author the missing parts of the asset like in a traditional pipeline and is not covered in this documentation.

When capturing an object, take care that there is no obstacle that prevents you from seeing all sides of the object:



Pay attention to shadows and highlights. Use an occluder if it is appropriate as describe in [When?](#) section.

4.1.2. Shoot

Color checker setup

A color checker is used to white balance the photos, but also as a scale metric for the virtual asset. Knowing the size of reconstructed objects is important as the photogrammetry process loses this information. Find a good location for the color checker. Close enough to the subject, but it shouldn't be shadowed by it. Find a place in the scene that can be easily removed during the process, or digitally paint it out. The color checker should usually be facing the sky.



Distance/resolution choice

It is also necessary to evaluate the importance of the object to capture. Assets that appear near to the camera in your game will require more texture resolution in game than background assets. You must define texture budgets for your various assets, and these decisions will drive the choices you make on your Shoot. The shooting distance from the object depends on the size of the object itself, the lens

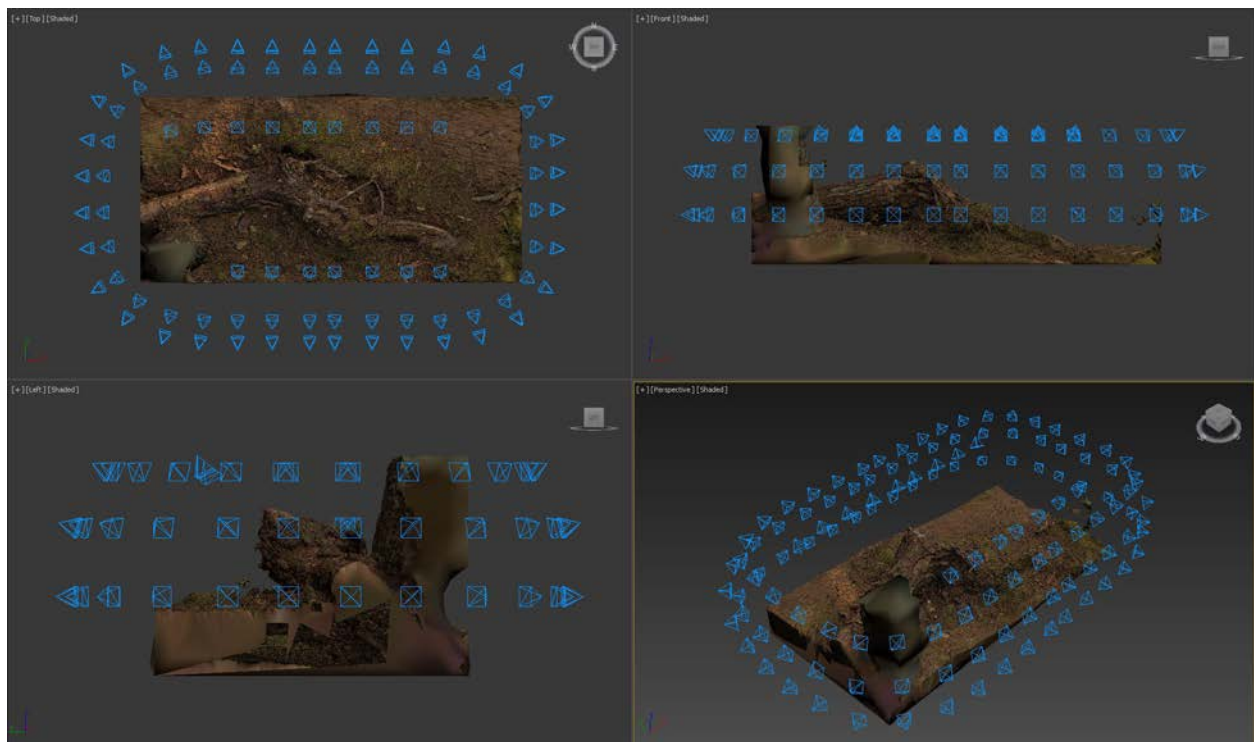
zoom, and the texture budget. The object should fill most of your images to maximize the reconstruction quality.

When higher resolution is wanted, closer pictures are needed. Note that it can take a long time to cover a big object like a tree stump with a micro resolution (See [Detail texture](#) section). Moreover Reality Capture has a limit on the number of photos you can provide as a source for reconstruction depending on the licence chosen.

In practice, for the capture of a big object like this tree stump with a final texture at 4096x4096, the photos should be taken at a distance of between 1 and 2 meters.

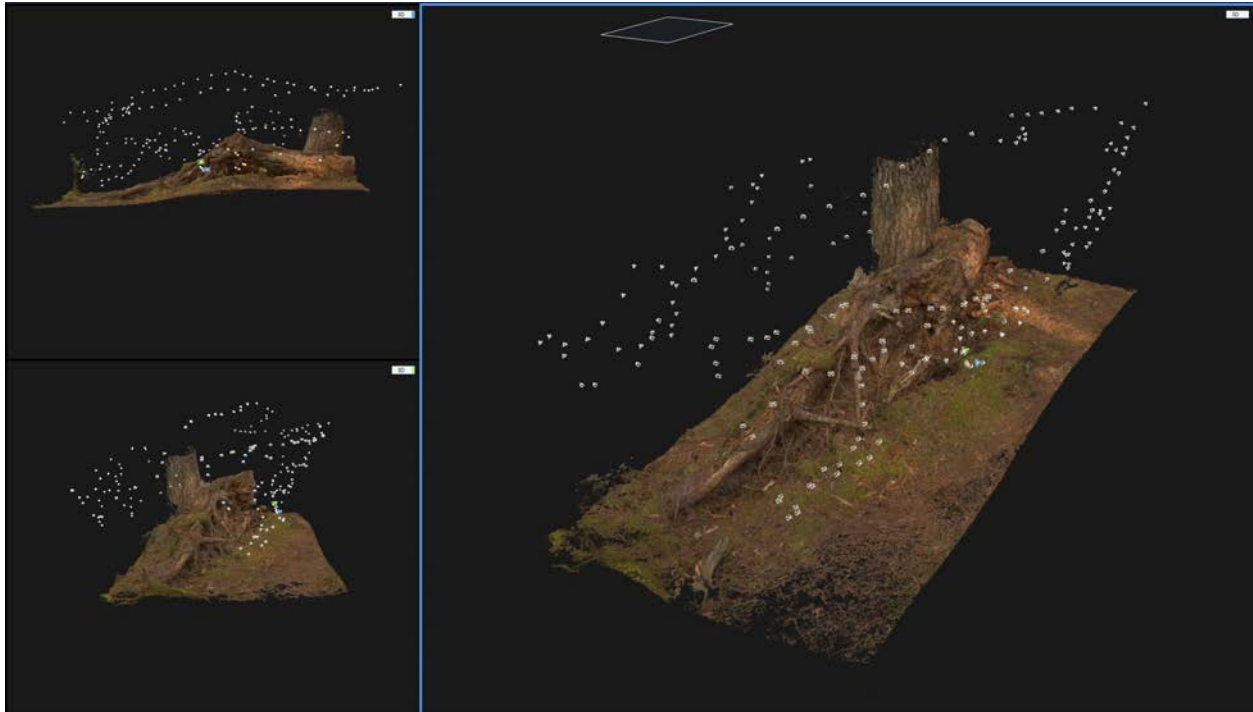
Shots location

When shooting an object, it is important to have **full coverage** of it. All parts of the object must be covered by several pictures so the reconstruction software can create groups of similar pixels, so that the final asset doesn't have holes or undersampled areas. Ideally the shots location should look like this (Screenshot from 3dsMax):



The pattern is done by moving around the object and taking many far shots, and some fewer close shots to provide more detail at the reconstruction step. At each horizontal camera location, it is important to also cover the subject vertically.

In practice it can be difficult to follow this ideal path and you may end up with something that looks more like the image below (Screenshot from Reality Capture):



It is easy to miss elements during the shoot. Don't hesitate to take a lot of pictures and validate them on a computer if possible (See [4.1.1. Test the images alignment](#)). Once the trip is finished it is expensive to return to the same location and hard to match with weather and lighting conditions.

Additionally a quick GoPro 4k video of the object can be done by following a similar path. This video can later be used as a backup to extract missing area or in case of undersampled or badly reconstructed area (See [Align images](#)). It is fast to do and can save many days of reconstruction correction. The color checker must be on the video to be able to white balance the frames extracted.

The Gopro can be used also to cover difficult areas for a camera. For example the bottom of the stump can't be done by the camera.



Shots Coverage

The reconstruction software needs to have enough similar pixels within the set of pictures to produce the point cloud. For a precise reconstruction it is recommended to have 90% overlap between pictures.



Practically speaking it simply means that between each shot, do a side step of around 30 centimeters and maintain the camera focus on the subjects, then do top to bottom pictures to cover the objects. Don't hesitate to take more pictures than needed. Do take care to only shoot when the **camera is stable**. Any movement when shooting will result in a blurry photo.

Scale metric

It is usually important to keep the same ratio between real world and your reconstructed virtual assets, depending on the context of your game. To achieve this, it is required to have a reference object of a known size in the reconstructed mesh that can be used to define the scale. Any object can be used. The color checker is a good candidate as it is already located near the subject and includes a 5 cm ruler:



For bigger objects like a castle, you could measure a part of it so that you know the real-world scale.

4.1.3. Huge objects

Very large objects - for example, cliffs, castles, boats - will require the use of a drone to capture images:



It is recommended to use drone video recording instead of pictures for such huge object as the quality is not significantly better and video will allow you to have easy coverage of the objects.

Note: It is possible to mix close shots taken by hand with far shots taken by the drone.

Alternatively, the drone can be used to create a placeholder area or temporary element.. The following example shows an area reconstructed with Reality Capture with photos extracted from a drone video. It can be use as a quick placeholder layout:



4.2. How to capture a surface

The photogrammetry process can be used to capture a surface that is then used in a game as a material.

4.2.1. Prepare the area

First, find a location without shadows or use an occluder to protect it from significant direct lighting. Remove the undesired elements. Pay attention to significant elements or patterns, especially if the goal is to create a tileable material. Avoid walking in the area that will be captured so that you do not make footprints in it or break natural elements.

At the location, use markers (Ex: tent pegs for soft surfaces, chalk for hard surfaces) to define the area to capture (marked by red arrows on the photo below). For a 2Kx2K textures and a texel ratio of 1K texel per meter in a game, use an area of 2mx2m. It is important to **keep the same ratio between the real world and the virtual asset**. Place the Color checker close to the area to capture:



4.2.2. Shoot

Capture a ground

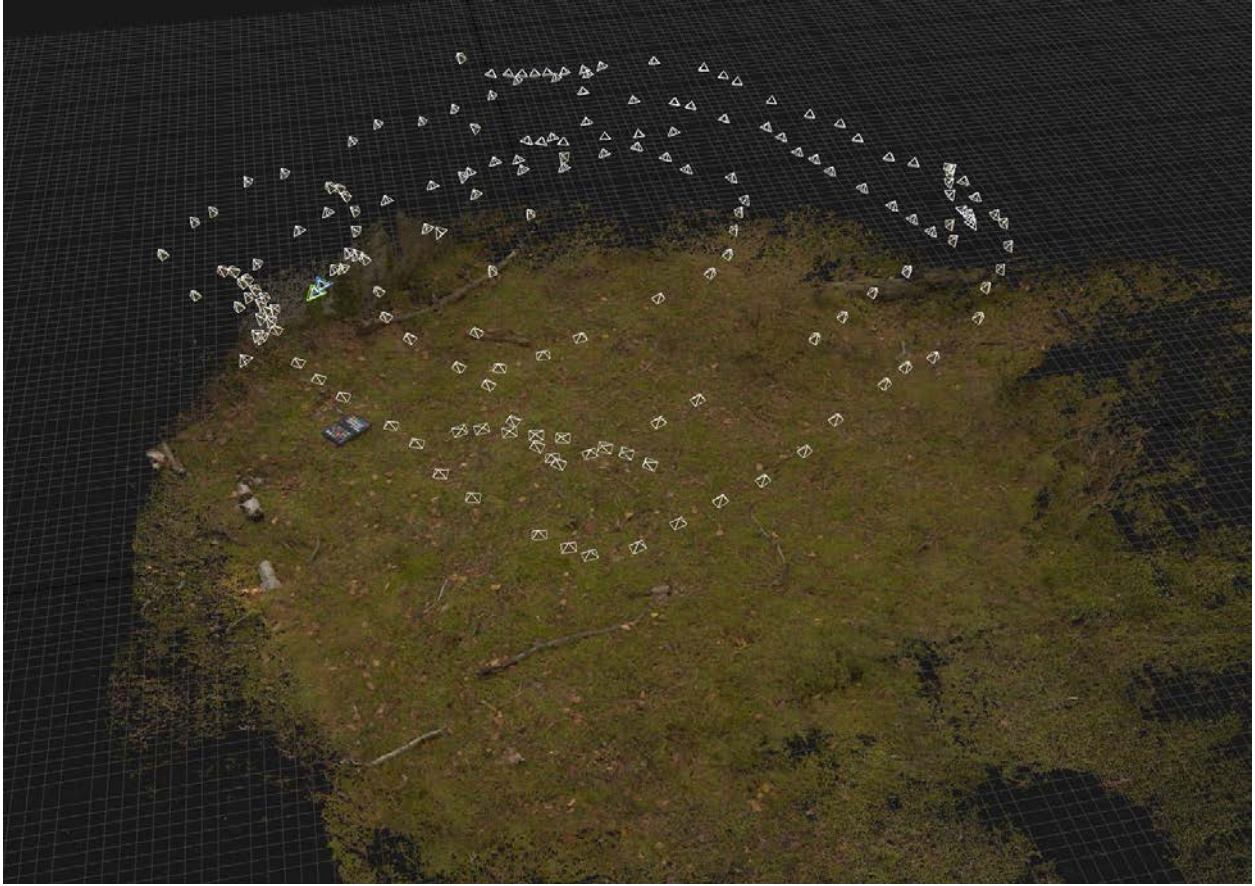
As a reference(location reminder or to keep track of shoot), it is recommended to take pictures of the surrounding environment before scanning the ground surface.

To scan a ground surface, take top-view photos following a spiral path from the exterior to center. Move sideways with your head oriented toward the center. Be sure to not interact with ground elements on the ground and pay attention to your body shadow and your own feet, otherwise reconstruction will suffer from it:

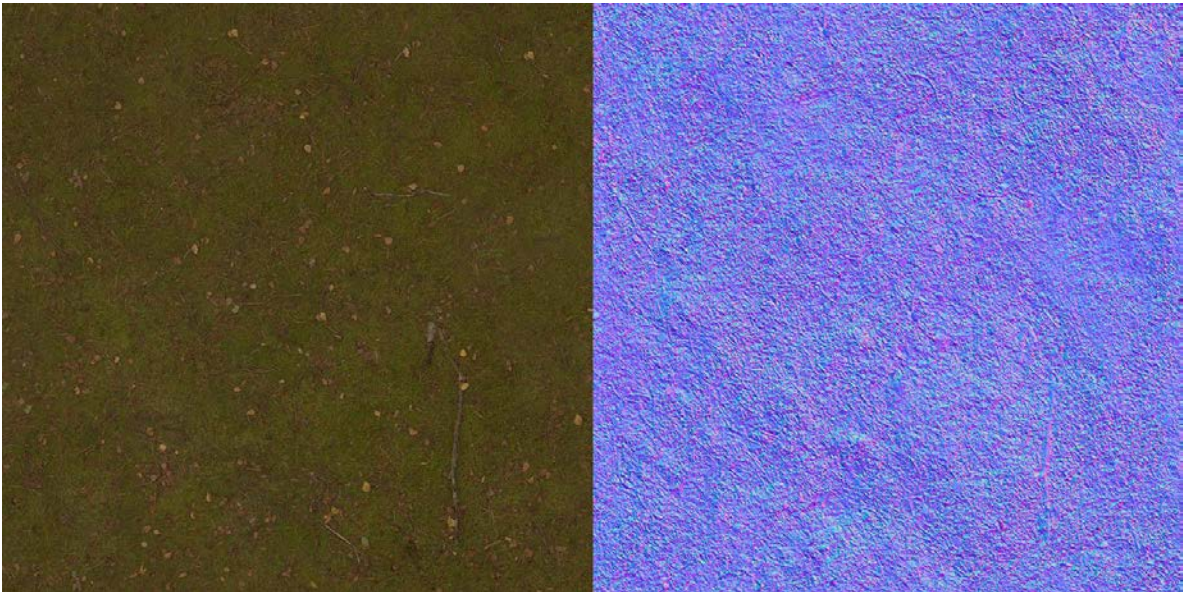


Below is an example result in Reality Capture. As expected, it is not possible to have a perfect path, you should just ensure that all of the area is covered. 179 photos taken:

Unity Photogrammetry Workflow



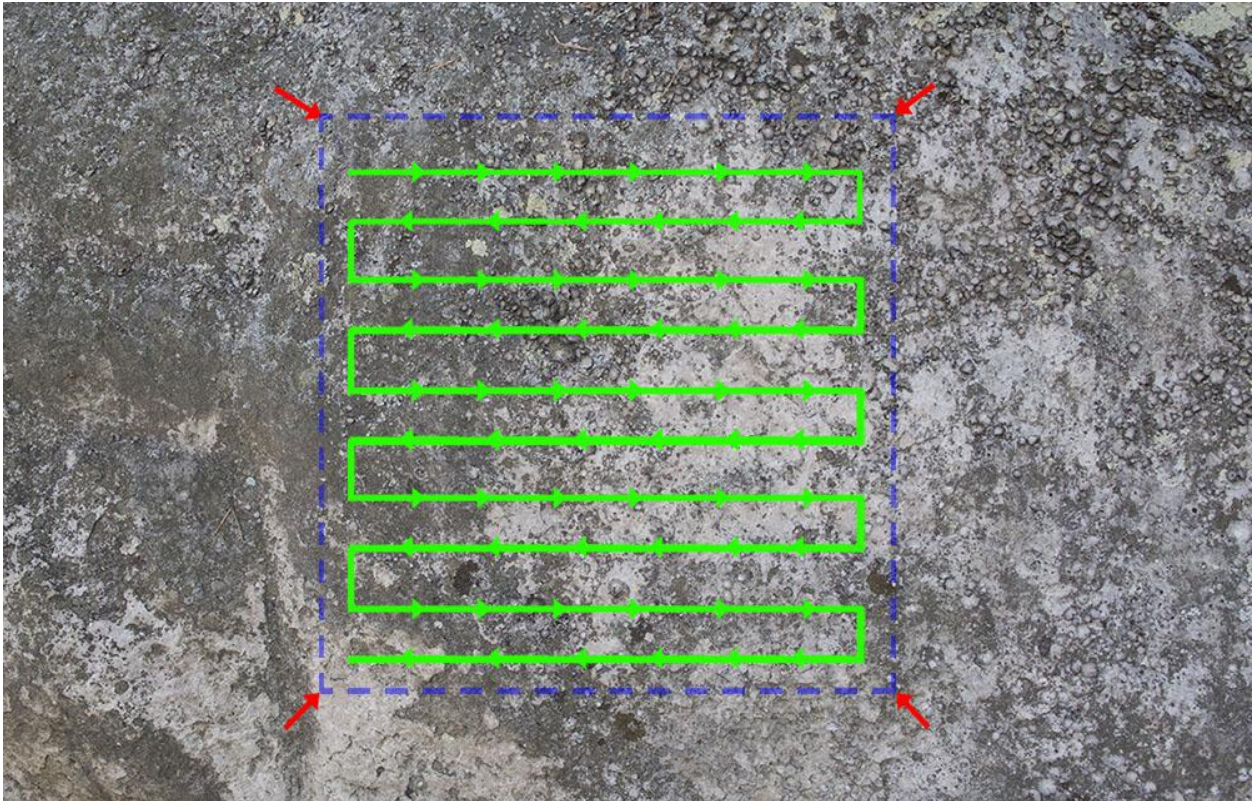
Resulting textures after full processing:



Capturing a vertical surface:

As a reference (location reminder or to keep track of the shoot) it is recommended to take pictures of the surrounding environment before scanning the surface.

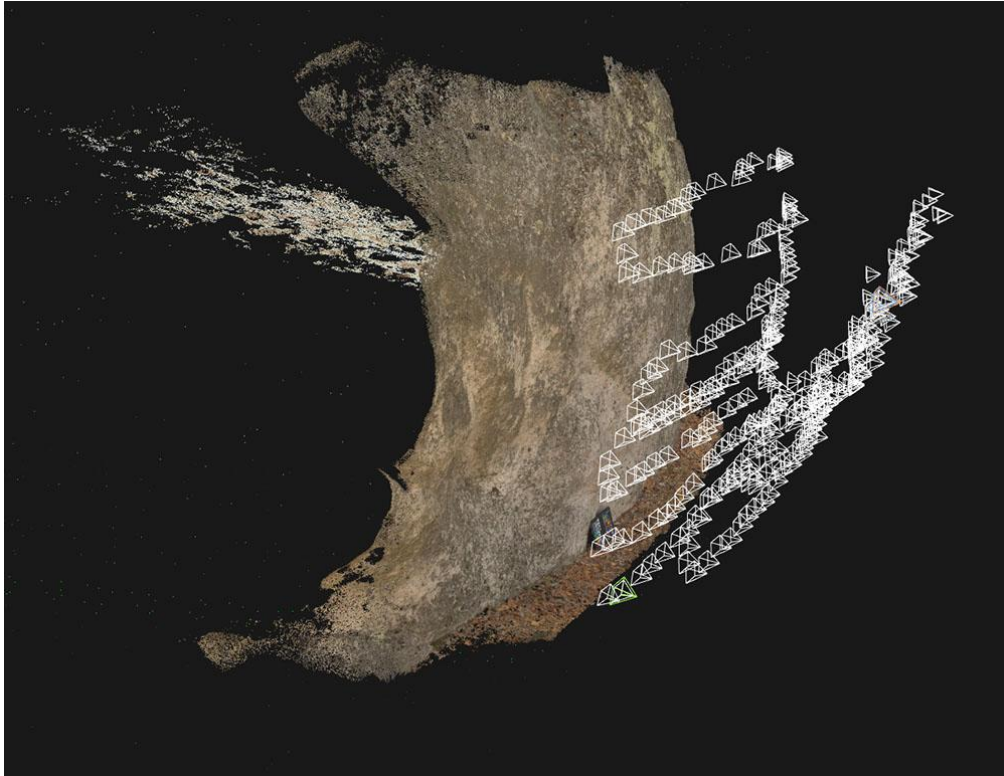
To scan a vertical surface, take pictures following a side-to-side path as shown in the example below. Place the color checker parallel to the surface if possible.



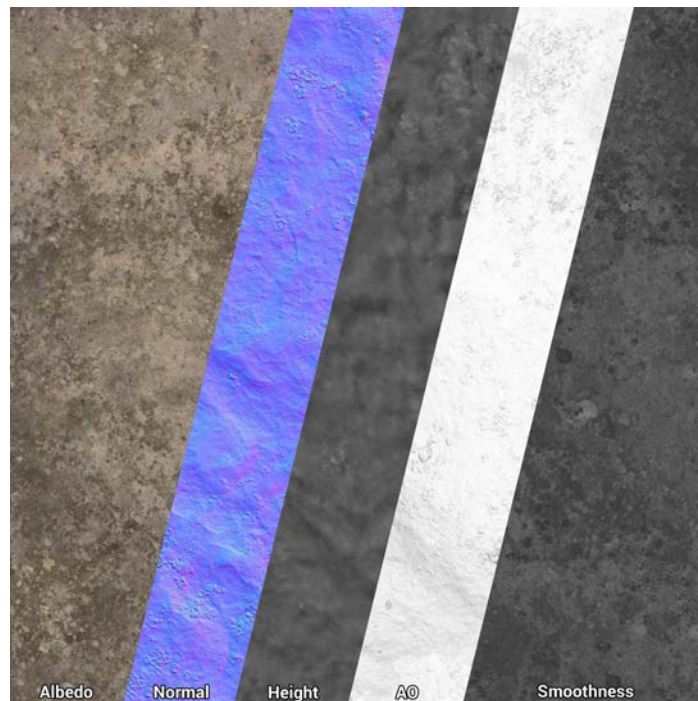
The capture can be done twice, the second time at closer position in order to provide more details to the material.

Unity Photogrammetry Workflow

Below is an example result in Reality Capture. As expected, it is not possible to have a perfect path, you should just ensure that all of the area is covered:



The reproduction of a surface is a good way to produce tileable materials:



These materials can then be use on handmade geometries or geometries reconstructed from a GoPro or a drone video. Video in this case is a really fast way to author assets as it is really fast to capture without sacrificing the asset quality. The geometry will provide the coarse detail information while the materials provide the fine details.

4.2.3. Detail textures

Detail textures are a secondary layer of textures used in combination with the main texture to add detail at a different scale. For example, a rock texture which covers a large area of a model might be enhanced by adding a higher resolution detail texture at a much smaller scale which shows small cracks and imperfections in the rock. (For more information, see <https://docs.unity3d.com/Manual/StandardShaderMaterialParameterDetail.html>)

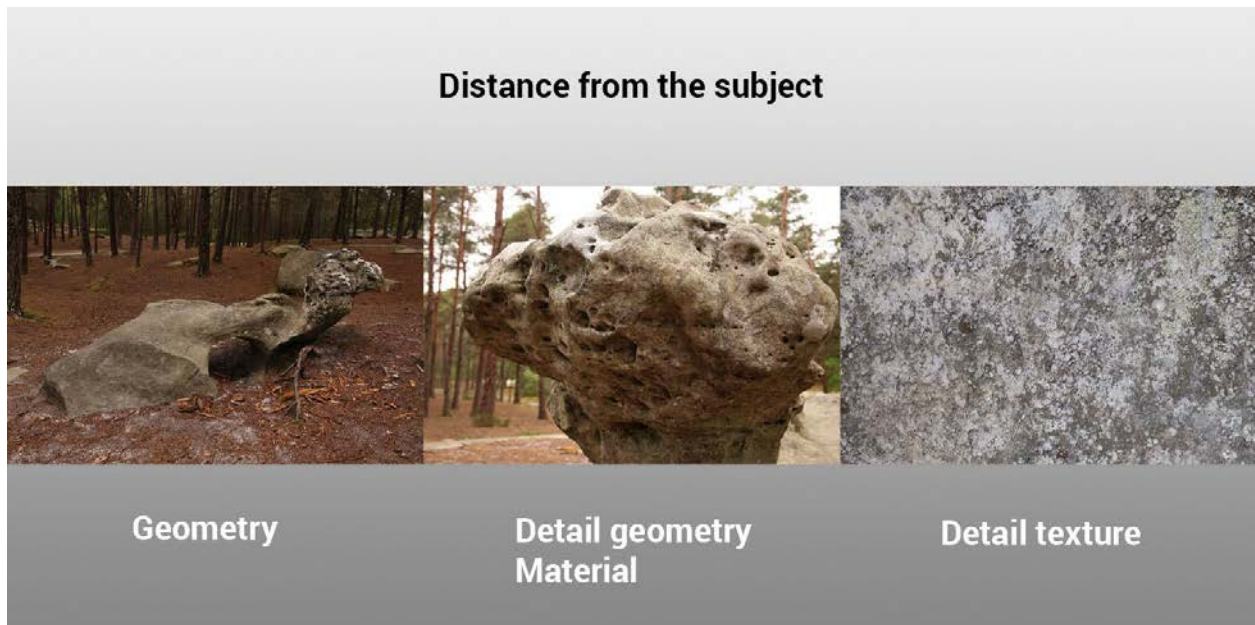
Scanned material can benefit from the same technique, scanning first at a coarse level, then performing a second detailed scan that can be combined when drawing the material in the game.

The capture process for detail textures is the same as for regular surfaces, except that shots are focused more closely on a small area. The area chosen to produce the details texture should be as generic as possible because detail textures are typically tiled many times across the object compared with the main texture. In practice, the shots are done at the closest autofocus limit. The example below shows on left the coarse subject, used to create the geometry and base material, and on the right the detailed area scanned to produce detail textures. The distance is the minimum allowed by the auto focus:



You should only select a small area for use as a detail texture. Attempting to use large areas of your object as the detail texture will result in textures that are too high-resolution and will waste time during the shooting process.

The image below is a summary of the correct distance from the subject for each purpose:



Geometry: Global photos to capture the main shape of an object.

Detail geometry/Material: Closer photos to bring more detail in the geometry and produce a usable material.

Detail texture: Photos done close-up at the autofocus limit. Used to produce detail textures.

4.3. How to capture a small object

Small objects are often used to populate a scene by duplicating many of the same small objects around in different areas. They can be rotated, scale and mirrored to create variety. For such objects, it is required to cover all parts from all directions to reconstruct full assets.

4.3.1. Set up a support

To be able to capture a small object from all angles it is necessary to be able to turn it around from all angles. Small objects can be held in front of the camera using a support clamp, ideally with as little contact with the object as possible. So, the first step is to find a good support. The exact nature of the support you will need depends on the shape and structure of the object you are capturing. Since it is difficult to carry all the equipment during a trip, it can often be preferable to gather small objects, bring them back to your studio, and capture them in a safe place close to the computer with the reconstruction software. This also allows you to have full control of the lighting.

The image below shows some examples of supports. Left, a tripod and a screw are used to capture all parts of the pine cone. Right, clamping pliers are used to put the ivy root in a good position for the capture.



The type of support required varies depending on the object, but common guidelines are:

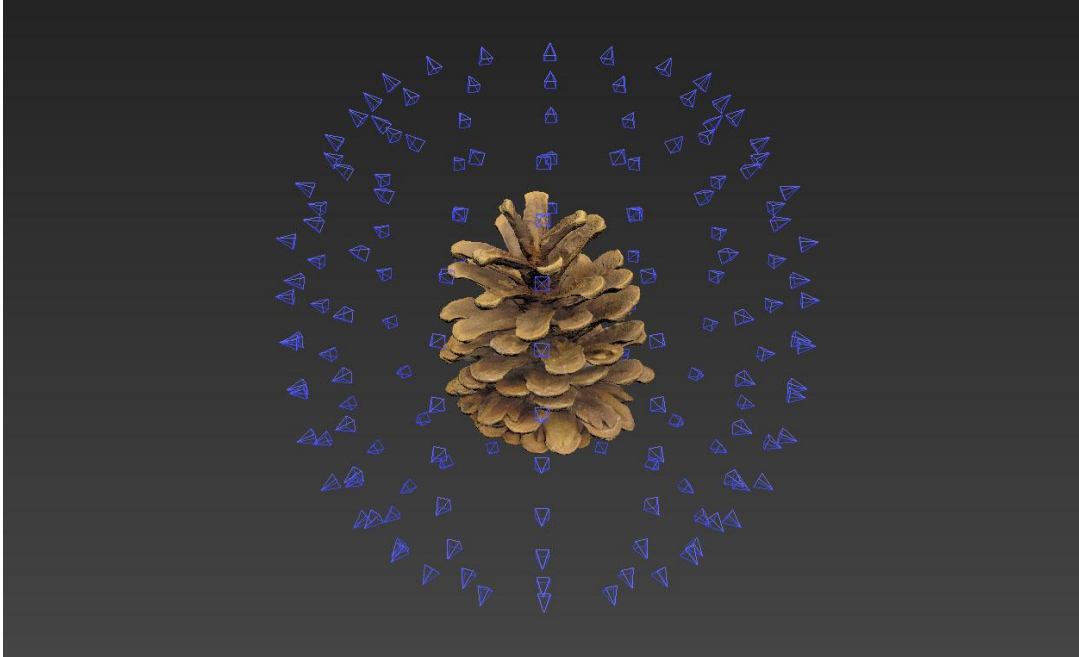
- Put the object in a position where all parts to capture are visible
- Minimize the part of the support that should be erased later.
- The support and the subject must be stable.

4.3.2. Shoot

There is no specific recommendation for the number of pictures required to fully capture an object. Complex objects like a pine cone can require thousand of photos. Use a snaking left-to-right path around the object to get good coverage.

Example of shot location (Theoretical positions simulated with 3dsMax):

Unity Photogrammetry Workflow



Leave the scene in place and check the photo alignment (See [Align images](#)). If the reconstruction software is not able to make the images match each other to generate a single point cloud (and therefore generate multiple meshes instead of a single one) then do additional photos where it fails. In the image below, red cameras represent a second shot session to increase image matching and produce a single point cloud. Don't hesitate to do more photos to solve the problem, it is faster than using the software's features and helpers.

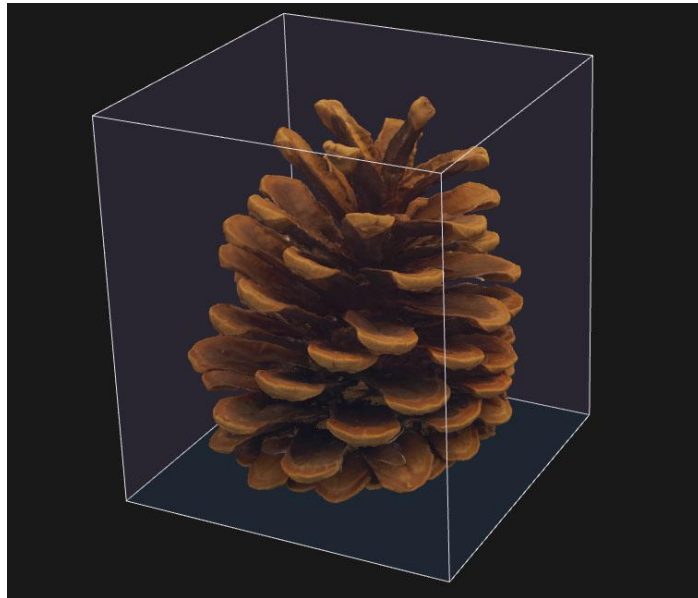


Unity Photogrammetry Workflow

Note: In this pine cone sample, the first short session produced 564 photos. In Reality capture, the first image alignment created 10 components (i.e 10 meshes). Five more sessions were needed to have all the photos in the same components (i.e a single mesh). At the end 1008 photos was taken for this complex asset.

If the scene setup can't be maintained or the images alignment can't be checked, do a video with a GoPro to be able to extract some frames and save processing.

Final result in Reality Capture:



4.4. How to capture the diffuse albedo only

Diffuse albedo is the view-independent color of the object. The regular process to recover the diffuse albedo from generated textures of the photogrammetry process is explained in another section and implies a light removal step (See [Remove Lighting](#)).

The process detailed in this section allows you to create a texture that is close to diffuse albedo of an object directly from the photos. The process is not simple and can't be used with hundreds of photos - it is aimed at shooting 1 or 2 photos that can be used directly as a diffuse albedo texture for small elements (See [How to capture foliage](#)) or as a reference to validate the result of the light removal step.

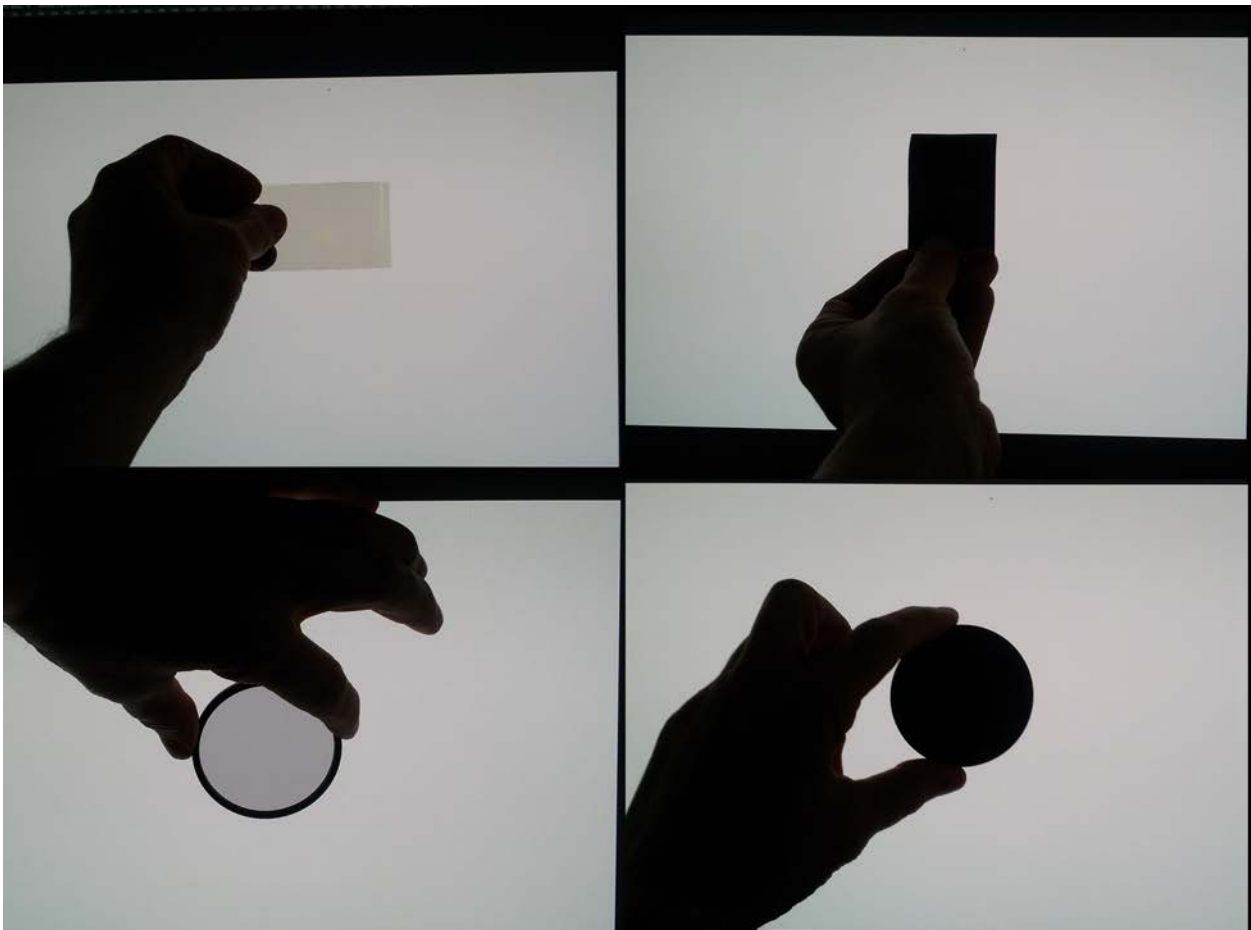
After a Shoot it is recommended to take a close picture of the diffuse albedo of the object with the method describe here. As explain, it will help to validate the reconstructed albedo after the light removal process.



To capture good diffuse albedo, you need to use specific devices: a ring flash and two polarizer filters. One lens polarizer for the camera, one linear polarizer filter for the light source. The objective is to remove any view-dependent effects like reflection.

Setup polarizer filters

Polarizing filters can be rotated to allow more or less admission of polarised light. The first step is to identify the orientation of the filters. This can be done with an LCD screen. LCD screens use linear polarized filters, so if another polarizing filter covers the screen with an angle of 90° , the lighting from the screen is blocked. At 0° the lighting will pass through. With this test the polarized film and the polarized filter can be aligned easily.



To be able to capture a color close to the diffuse albedo it is necessary to remove as much directional light information as possible. The goal of this section's method is to take photos where only controlled filtered lighting from a flash is visible on the subjects. This means the lighting coming from the flash should be more bright than the one from the environment and thus more visible.

Unity Photogrammetry Workflow

A ring flash is recommended over a cobra flash as it produces parallel shadows to the view that are not very visible on the photo. The ring flash is augmented with a linear polarizer filter film on led lights to linearly polarize the flash lighting. Then a lens polarizer is used for the camera to remove flash reflection.



Unity Photogrammetry Workflow

With this setup, the lens polarizer filter can be rotated at 0° or 90° to get photos with or without specular (Meaning that both lens filter and flash filter are either parallel or perpendicular).



Unity Photogrammetry Workflow

The following example illustrates the result of using the above setup. On the left, with 0° lens filter orientation (i.e the lens and light polarizer filters have the same orientation), the specular from the flash is visible. On the right, with 90° lens filter orientation (i.e the lens and light polarizer filters have different orientation) most of the reflection from the flash is filtered.



To minimize the environment reflection, set the exposure with a shutter speed under 1/180 (CANON flash MR-14EX II speed). With this set up, only the lighting coming from the flash should be present in the photo.

These photos can provide information to produce a roughness/smoothness texture.

The polarized photo should be white balanced with the color checker to have correct exposure (see [White balance](#)). The result is a photo that represents what the albedo looks like.

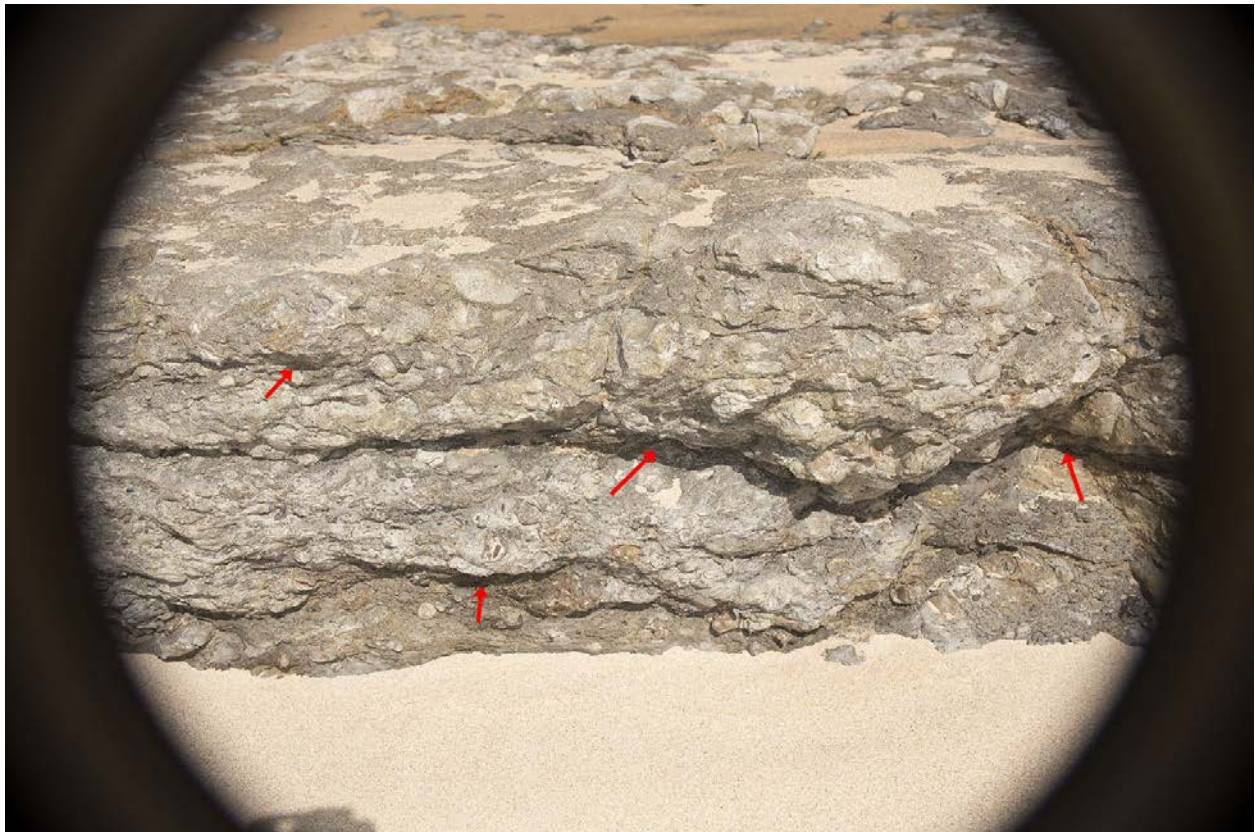


If all the shots could be taken with this method, the light removal process may not be necessary.

However there are a few problems with this approach:

- Occlusion from the lens filter support implies that more photos are required as there is less coverage.
- The batteries of the flash are not powerful enough to be used intensively. Batteries are empty after few shots.
- Capture time increases a lot as it is necessary to wait for flash reloading, making the process impractical.
- This setup doesn't work if the subject is under a strong lighting like the sun. The sun is so powerful that it still produces shadows and specular (red arrows on the example below) on flash illuminated photos. The solution is to use an occluder.

Example of photo shot using ring flash with shutter speed at 1/4000:



4.5. How to capture foliage

Foliage is difficult to capture, mainly because it is sensitive to wind and movement during the capture. Still, it is recommended to capture it on-site because cutting and bringing them to another location may change their look in a short amount of time. There are various techniques with additional custom equipment and under different lighting conditions that can be used, but then it becomes difficult to travel efficiently.

To solve these issues, it is recommended to use a photogrammetry process only to capture the albedo. Most games use simple plane polygons to represent foliage for performance reasons. Trying to extract geometry in this context will only help for the normal map. You will need to find a balance between required equipment, capture duration, reconstruction, and manual authoring. It is recommended to manually build the geometry in a 3D software.

First, create a location protected from any significant direct lighting like the sun:

Unity Photogrammetry Workflow



Next, place the foliage and the color checker on a matte black background to quickly create a texture mask (See [Foliage - Photoshop](#)) without any environment lighting. A colored background will cause problems because the color will encroach on to the edges of the foliage.

Third, use the camera with the ring flash and the polarized filters at 90° (See [How to capture diffuse albedo](#)):



After that, the photo is white balanced and the result is close to a diffuse albedo. The geometry can be modeled in a 3D software and the normal map authored manually or generated.

5. Processing

For the reconstruction software, which is the most important part of the photogrammetry workflow, we recommend that you use Reality Capture from CapturingReality. The efficiency and the quality of the reconstruction is far better than other software (Photoscan from Agisoft, 123D Catch from Autodesk) for an affordable price (with the 3-month Indie license). We explored other solutions like visualSFM but discounted them for various reasons - for example visualSFM can't be used commercially.

5.1. Images processing - DCraw & Photoshop

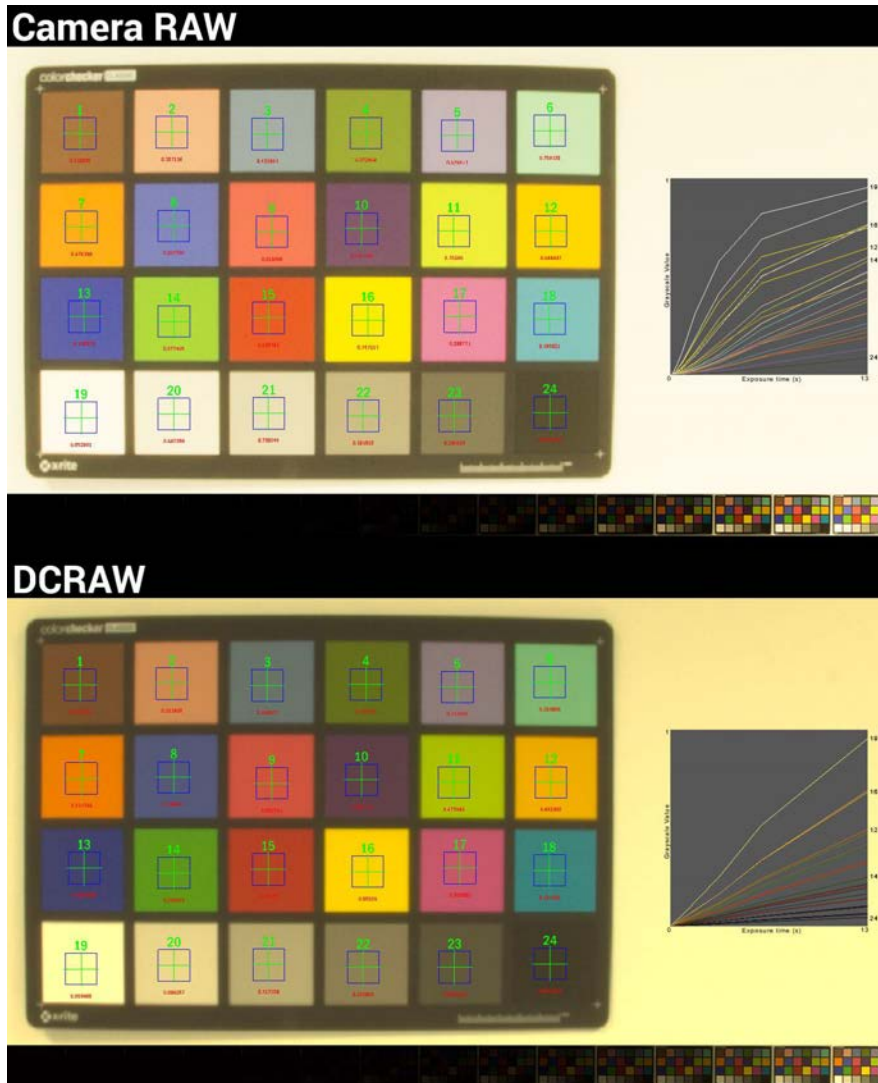
5.1.1. Determine photo set required for reconstruction

Before any processing begins, it is good to validate that the set of photos you are going to provide to the reconstruction software is sufficient. With Reality Capture it means that images should be aligned and a single component created. Check the alignment of the image set directly with the RAW data in Reality Capture. If more images are needed, a new photo-shoot session can be done or they can be extracted from a 4K video take at the end of the shoot session as detailed in section [How to capture an object](#). It is necessary to do this step first, so all the following manipulation can be done on the entire final photos set.

5.1.2. Convert RAW to TIFF

Reality Capture accepts RAW files directly. However, to get better results when extracting diffuse albedo, it is required to correctly convert your images to linear format and to white balance the photos set. This is a step performed outside of Reality Capture.

DCraw is used instead of Photoshop (Camera RAW) because it allows you to have correct linear conversion when output to TIFF:



It means that Camera RAW isn't used for white balancing. Photoshop is used instead.

To convert the raw files to TIFF, open a command prompt in the folder with the photos and the DCraw executable. Run the command "dcrw -v -4 -T *.CR2" (CR2 as RAW format).

-v: To verbose because the process is long

-4: To do 16-bit linear format with camera white balance and a gamma at 1

-T: Write TIFF format

Move all the TIFF (That are in 16-bit) into a dedicated folder.

5.1.3. White balance

White balance on photos is performed on a 32-bit format inside Photoshop before being sent to the reconstruction Software. The process for doing this is as follows:

- Open a TIFF in photoshop including the color checker
- Convert the TIFF into 32-bit
- Pick the color of the second grey patch (60%). It is better to use the second patch especially if the first is close to 1 (255, 255, 255)
- Close the TIFF without saving
- Apply a batch script to the TIFF folder:
 - Convert to 32-bit
 - Make a layer with solid color at 0.6, 0.6, 0.6 (60% grey)
 - Set the solid color layer to multiply
 - Make a layer with the color previously picked
 - Set the color layer to divide
 - Merge all
 - Convert to 8 bit
 - Save
 - Close

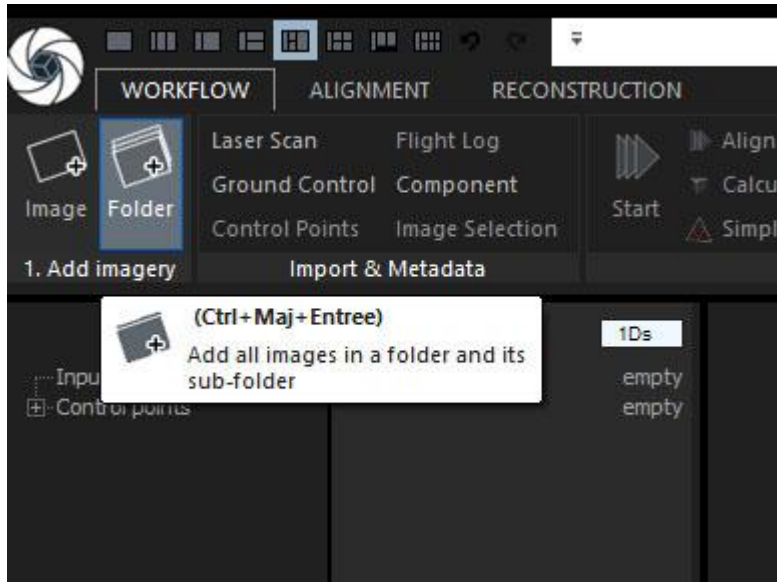
Photos are converted to 8-bit to avoid any issue with gamma correction in Reality Capture. Also 8-bit format are smaller to store which is helpful in case of a huge amount of photos.

Note: During the reconstruction Reality Capture converts images to an 8-bit format (JPeg). So there is no loss of information when storing in 8-bit.

5.2. Reconstruction - Reality Capture

5.2.1. Add all images

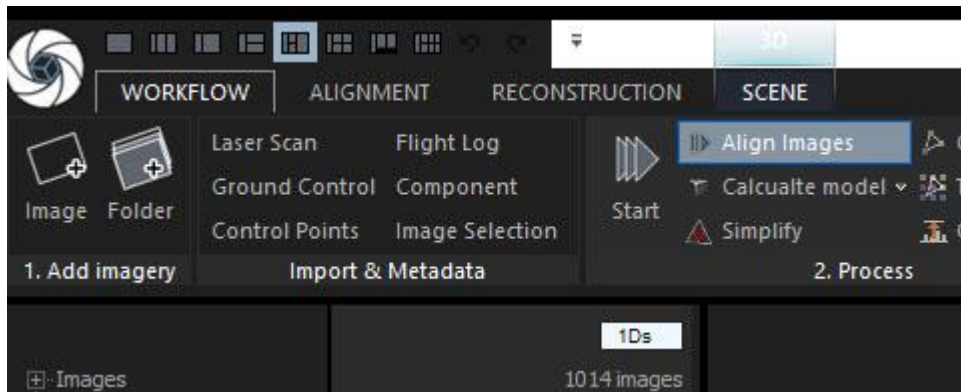
Use Add image or folder to load a set of photos or drag and drop from file browser.



At any time, photos can be added, especially if alignment produces several components.

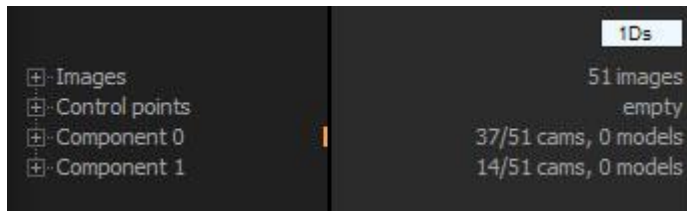
5.2.2. Align images

Before you start the reconstruction, the images must be aligned together. Whenever some images are added, the alignment must be done.

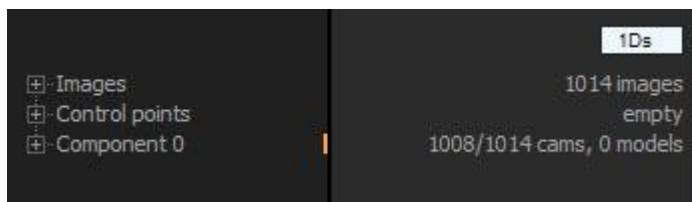


5.2.3. Check the components

Check if all the images are grouped into the same component.



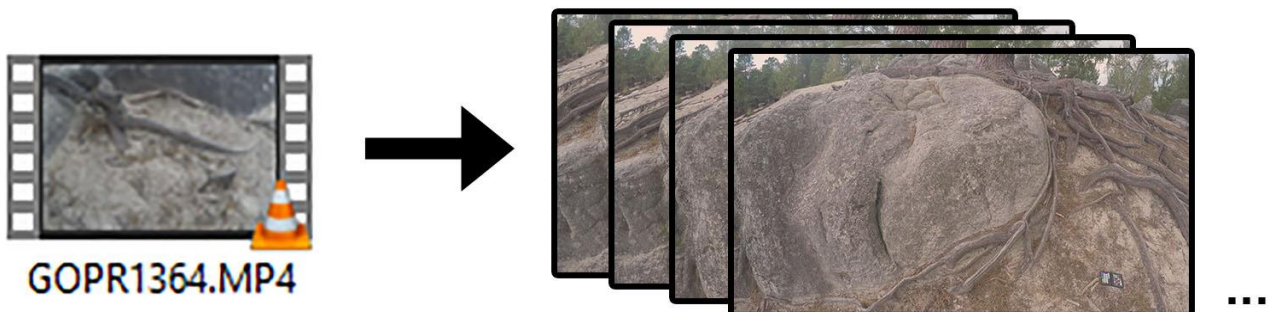
If not, add more images from new capture sessions (See [How to capture a small object](#)) until all images are linked in the same component:



When you have a single component, it can be used for the reconstruction.

5.2.4. Frames from video

VLC can be used to extract frames from a video:



- Launch VLC
- Click Tools -> Preferences
- Click Show settings -> All
- Video/Filters -> active check box "Scene video filter". Don't forget to uncheck after the work is done
- Video/Filters/Scene filter -> Add directory where to save the frames
- -> Set recording ratio. 10 saves 1 frame every 10 frames.

- -> Save
- Open and run a video

Frames will be extracted during the playback in the directory that you selected.

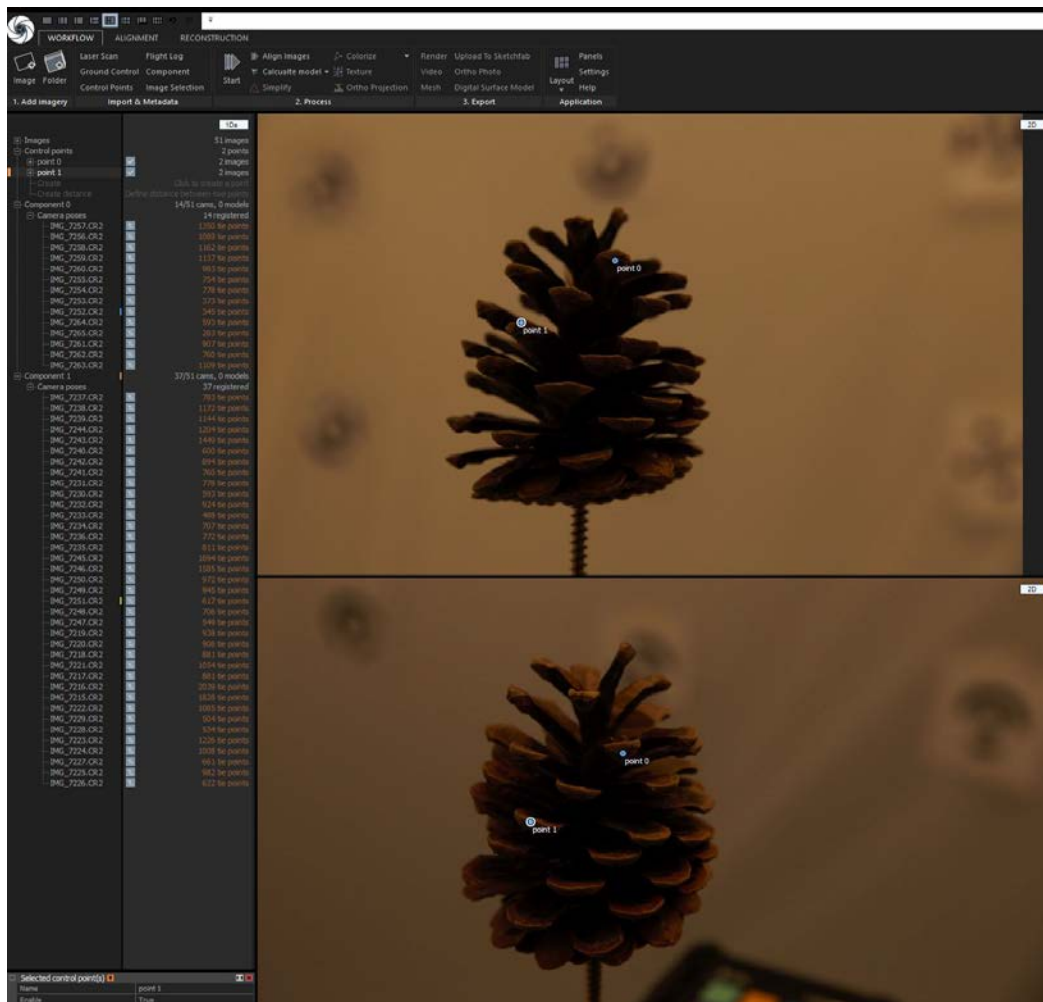
Note: VLC's default extract format (png) is supported by Reality Capture.

Images extracted from the video must also be white balanced. The video capture must include the color checker on some frames.

5.2.5. Control points

If all images are not linked in the same component, and no more images are available, you can create control points manually to correct the missing links (See: Reality Capture help, "Merging components using control points").

Create some control points and drag them on the images into the different components. The control points on the different images should represent the same position in the 3D space.



After you have added a few control points, test the image alignment to check if they resolve the missing links. Repeat the process until you obtain an efficient component for the reconstruction.

The addition of control points is a long process. It's recommended to add photos to resolve the missing links whenever possible, because this is a faster way of fixing the capture.

Note: Reality Capture provides other alignment controls (see: Reality Capture help - Merging options).

5.2.6. Set a scale

Control points can be used to set the real scale of an object (See: Reality Capture help: "Scaling a Scene/Distance constraint").

Create two control points and place them on images that show the color checker ruler.

These control points must be assigned to two or more images. The more images that are used, the more precise the result. For natural assets, two images is good enough.

Activate "Define distance" button in the alignment tab.

Link the two control points by clicking on the first, then joining it to the second by holding the left button. When the button is released, a distance constraint is created. By default the distance unit is meters (: the unit can be modified in the local.xml file in the Reality Capture installation folder.)



In constraint properties, set the distance to 0.05. The passport color checker ruler size is 5 centimeters. If you have used a different size reference on the photos, enter its measurement.

Click on the update button under the define distance button to refresh the actual distance.

Selected constraint(s) [?] [↕] [✕]	
Name	distance 0
A	point 1
B	point 0
Enable	True
Distance	0.050000
Actual distance	0.050002

The scale setting can be done after the reconstruction.

For surface and detail materials the re-scale is not needed. Their size is constrained by the markers during the capture and by the resolution of the texture. For example, a 2048x2048 does 2 square meters with a game ratio at 1024/m.

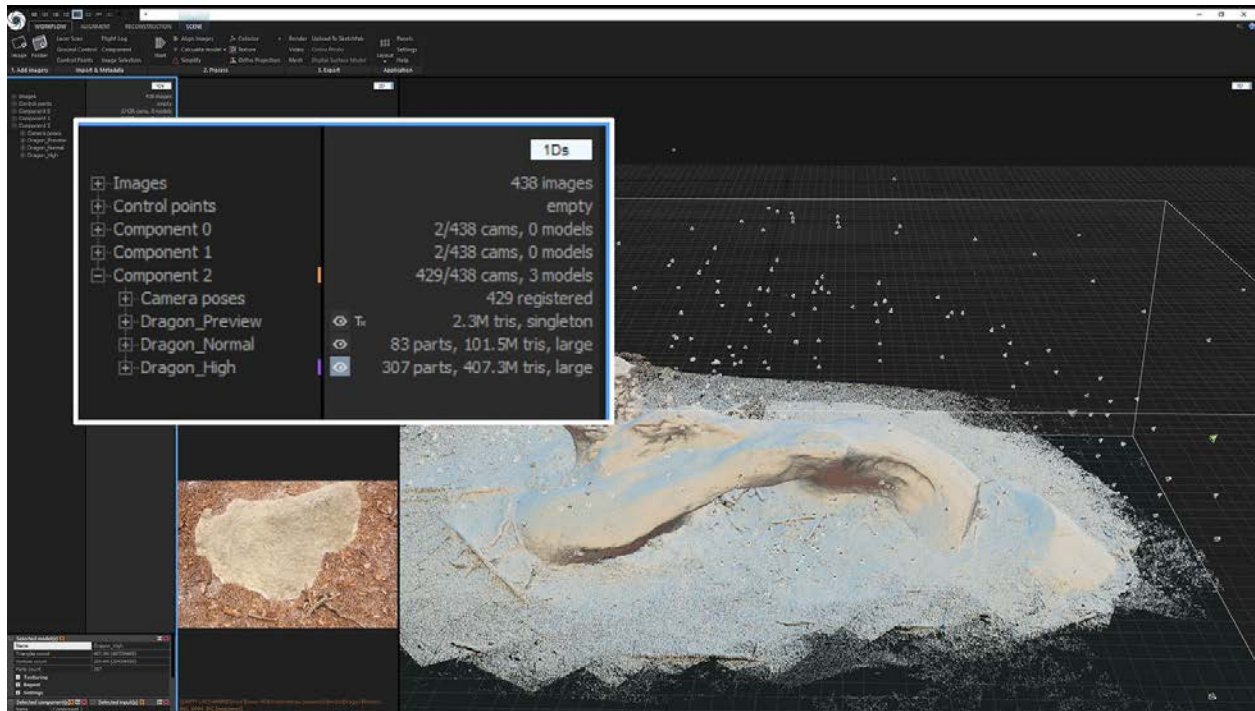
5.2.7. Create the model

Select the component used to create the model (component 2 in the image below).

Unity Photogrammetry Workflow

Set the reconstruction region. For surface and detail materials set the reconstruction region close to the markers (See [How to capture a surface](#)).

Select the model quality (Preview, Normal, High). Ideally you should choose the highest quality mode but you may want to select a lower level depending on your computer's processing power.



The previous image illustrates a rock reconstruction with three levels of details.

The preview quality produces 2.3 M of polygons in 36 seconds. This model can be used to bake a texture of up to 1K x 1K.

The normal quality produces 101.5 M of polygons in 28 minutes. This model can be used to bake a texture of up to 8Kx8K.

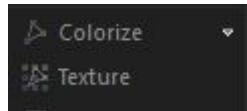
The high quality produces 407.3 M of polygons in 7 hours and 21 minutes. This model can be used up to bake a texture of up to 16Kx16K.

Note: The above timings were done using the computer configuration listed in the [Equipment](#) section.

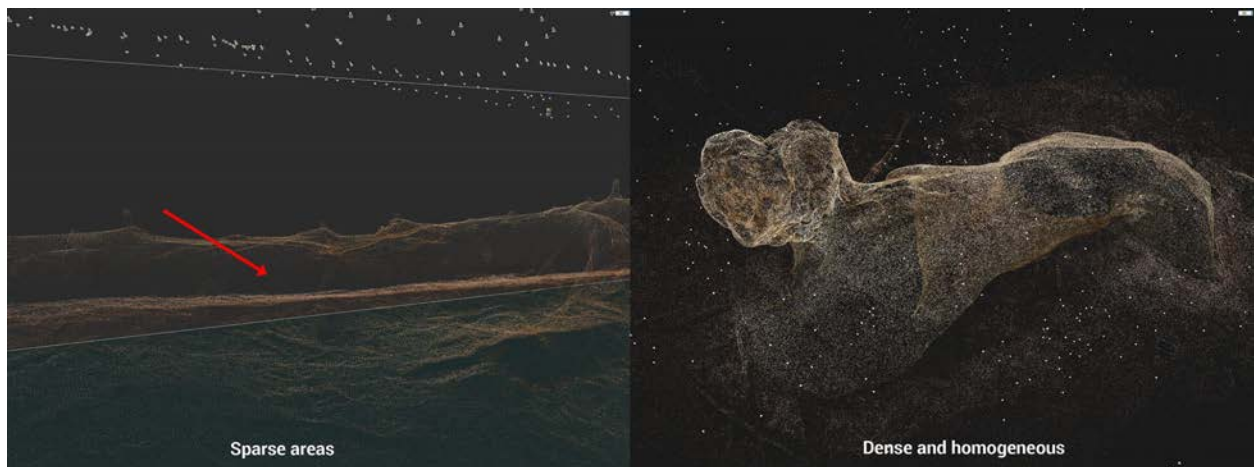
The high quality mode may sound like too much, but it is a good way to ensure you have the correct resolution in all parts of the object. Normal quality can produce average results in some parts. The normal map baking quality depends directly on the high quality mesh.

5.2.8. Colorize or texture

With Reality Capture, it is possible to store color information either by vertex (Colorize) or in a texture.



Which option you choose will depend on the reconstruction result. When the reconstructed point cloud is dense and homogenous, it is recommended to use colorize. Reality Capture displays the mesh as a point cloud, but internally it converts it to a mesh with vertices. The photo color is transferred to the vertex colors. This requires only a .ply file (which handles vertex color), which is easier to manipulate compared to an .obj file with a big texture. Another benefit of storing color in vertices is that it doesn't put any limit on the size of the model. With a 16Kx16K texture only 268 Mpixels can be stored, and high quality mode can produce models bigger than 268M vertices. The time required to colorize vertices depends on the amount of vertices.



If the point cloud is sparse or if there are significant gaps in the reconstruction, it is recommended to use textures instead of vertex colors. This is because the color from the source photos are transferred to textures, and the textures span the gaps between vertices because they are drawn across the mesh's surface. The default texture size is 16Kx16K, however this can be changed in the mesh properties. It is best to keep as much as quality as possible before the baking process (i.e use high resolution texture).

Note: This process of creating color textures from the source photographs is a separate process to the creation of normal maps. These colour textures relate to the diffuse albedo texture, whereas the the normal map creation process only depends on the quality of the reconstructed geometry.

5.2.9. Simplify

The high resolution mesh produced with Reality Capture can't be used in real time and most 3D software will have trouble handling such huge meshes, therefore a lower resolution mesh is required. The goal is to make the lower resolution mesh look as similar as possible to the high resolution mesh by generating a normal map using baking tools. The baking tools will transfer high frequency information (small details) of the high resolution mesh that are otherwise lost when converting low resolution meshes to textures.

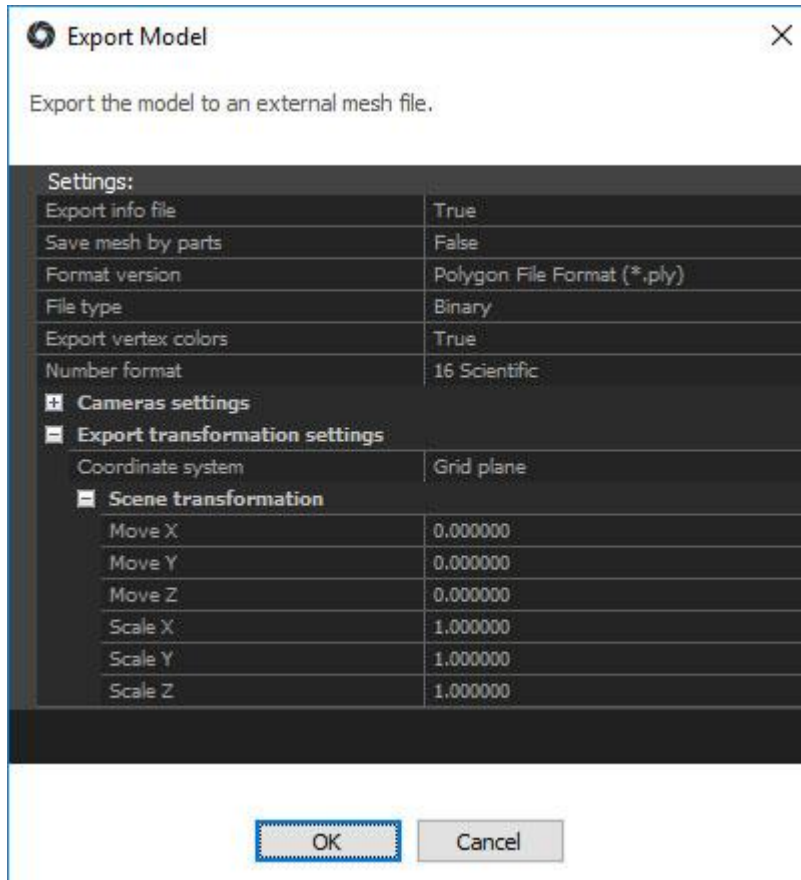
Reality Capture provides the *Simplify* process to optimize a mesh (See: Reality Capture help - *Simplify*). *Simplify* allows you to define the number of desired polygons. If the *Simplify* process gives a good result, it can be used as a base to produce the low resolution mesh for baking. If the result is not good enough, or if the mesh is too complex, use the *Simplify* process to do a medium resolution mesh with 1 or 2 millions polygons which can be imported easily in a 3D software like 3dsMax. (See [Create a low mesh for baking - 3dsMax](#)). You can use this in 3dsMax to produce the low resolution mesh for baking.

5.2.10. Mesh export

Both the high resolution and the medium or low resolution mesh (depending on the *Simplify* process quality) must be exported. The high resolution mesh will be used as source for baking the normal maps, and the low or medium resolution mesh can be reworked in 3D software to generate a proper low resolution mesh used as destination of the baking.(See: [Create a low mesh for baking - 3dsMax](#)).

Click on Mesh button in workflow or reconstruction tab.

Select .ply format if the reconstruction is dense and homogeneous. The mesh must be colorized.

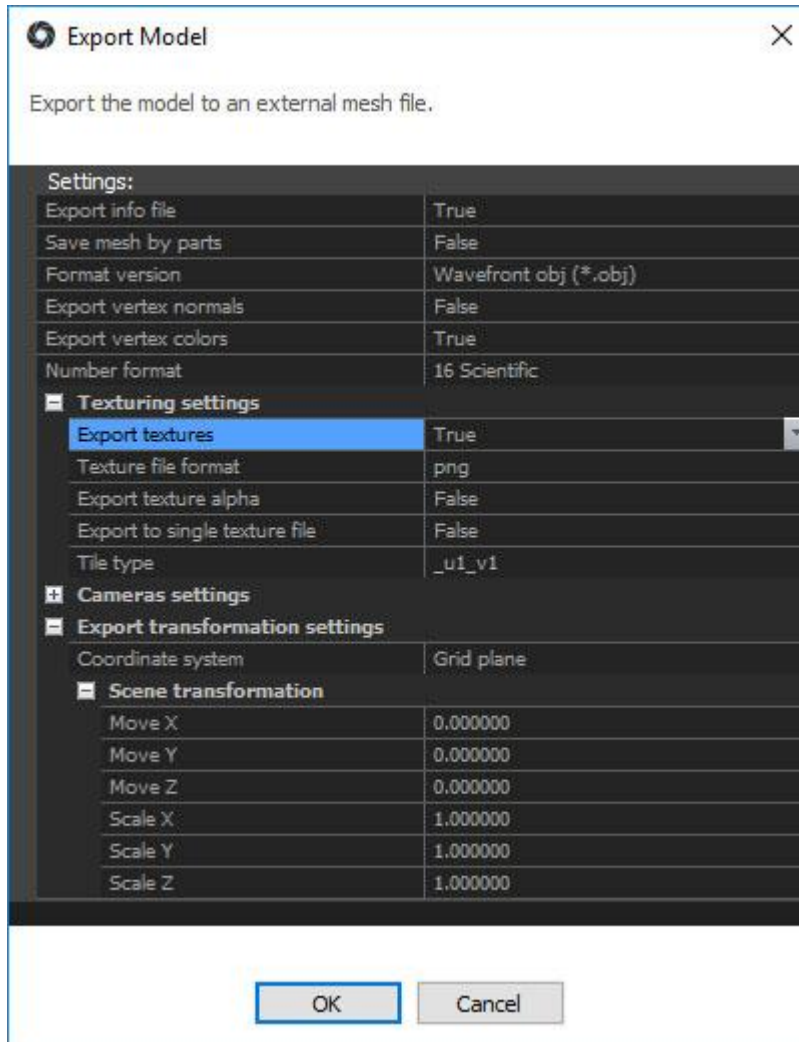


Save mesh by parts should be false.

Export vertex color must be true.

No move and no scale are applied.

Select obj in others cases. The mesh must be textured.



Save mesh by parts should be false.

Export textures must be true.

No move and no scale are applied.

On the previous rock sample, the high resolution mesh once exported to .ply format is 7.77 GB and in .obj format is 47 GB. To save time and disk space, it is recommended to use the .ply format.

5.3. Create a low resolution mesh for baking - 3dsMax

Note: The PLY import/export plugin is needed for 3ds Max.

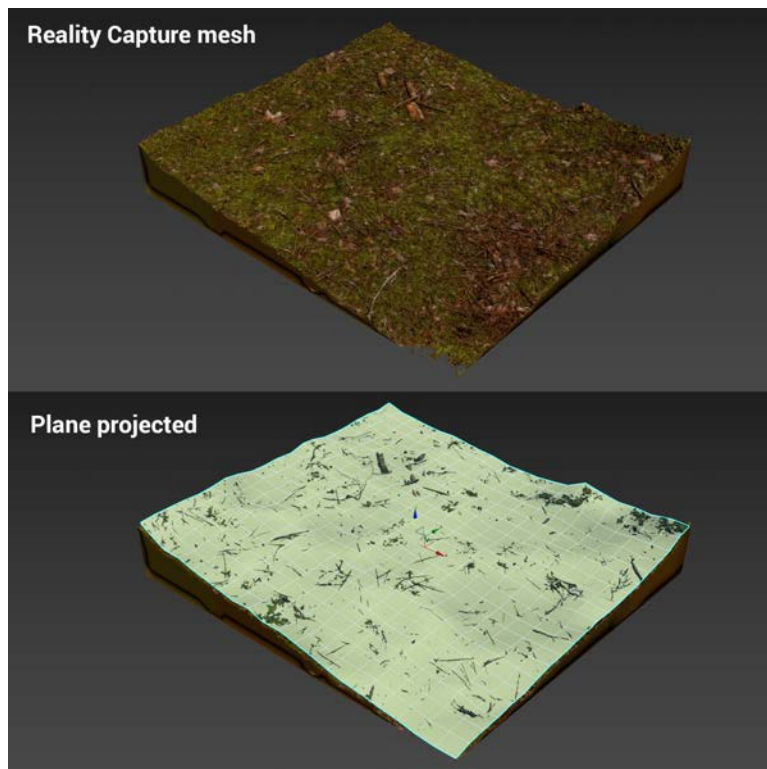
This low resolution mesh will be used as destination for baking. The high resolution mesh and the low resolution mesh must be at the same place and size because the baking process is a projection from high to low.

Import the medium or low resolution mesh from Reality Capture in 3dsMax. Set the scale to 1 and make sure flip axis is unchecked.

5.3.1. Tileable surface

Create a square plane over the Reality Capture mesh with enough divisions to match the global shape. Then apply planar UV (It is recommended to do this step before the projection step below).

Project the square plane on the Reality Capture mesh. The conform tool is a fast way to do it (See: 3ds Max "conform" documentation).



Export the plane in .obj or .fbx with scale at 1 and flip axis unchecked. It will be used as a low resolution mesh for baking.

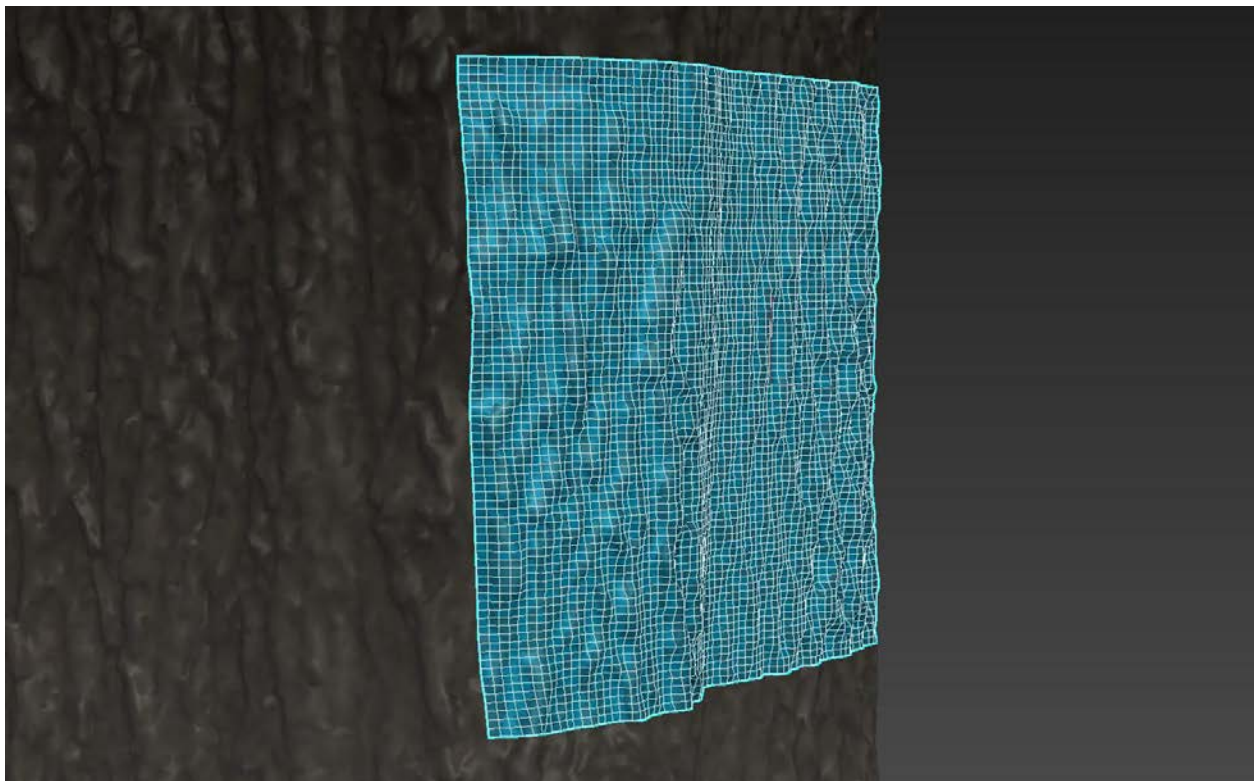
5.3.2. Detail material

For detail material the goal is to bake only high frequency information in the detail normal map.

Create a square plane with enough division to match the low and medium frequency information of the mesh. More division in the mesh means there will be less information to store inside the normal map (as the information is present in the mesh). With a mesh division that matches the low and medium frequency information, only high frequency information will remain during the baking process of the details textures.

Then apply planar UV (it is recommended to do this step before the projection step below).

Place and project the square plane on the Reality Capture mesh. Conform tool is a fast way to do it (See: 3ds Max "conform" documentation).

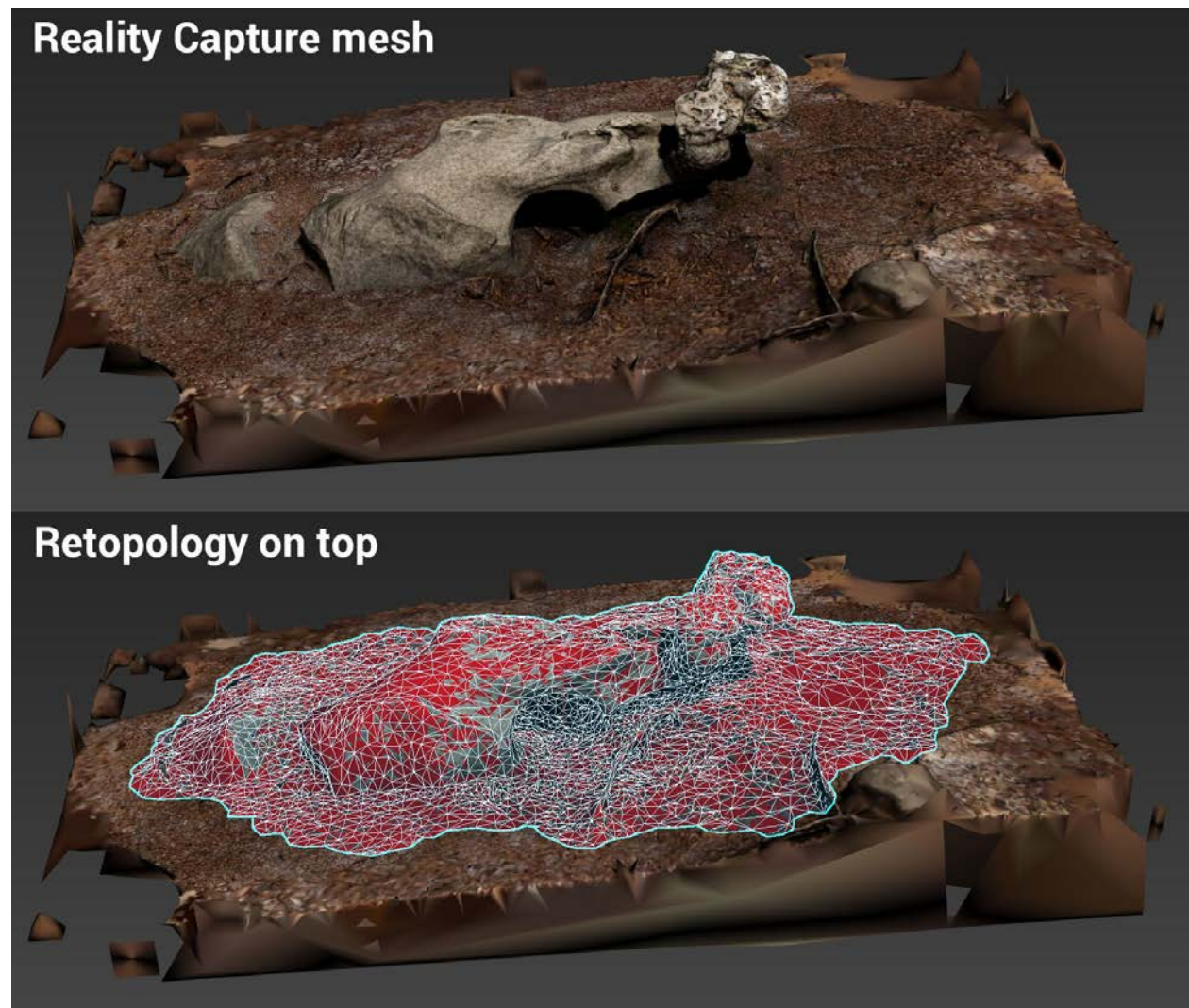


Export the plane in .obj or .fbx as low resolution mesh for baking with scale at 1 and flip axis unchecked.

5.3.3. Object

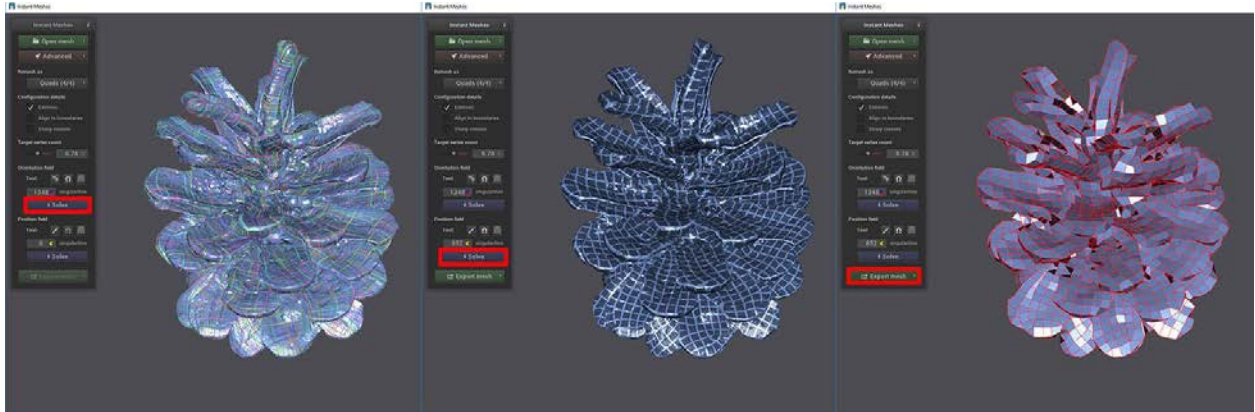
5.3.3.1. Geometry

Start from the Reality Capture low resolution mesh and clean it, or do the retopology based on the medium mesh from Reality Capture. This is the usual process of authoring a game asset, and therefore it is not covered in this guide.

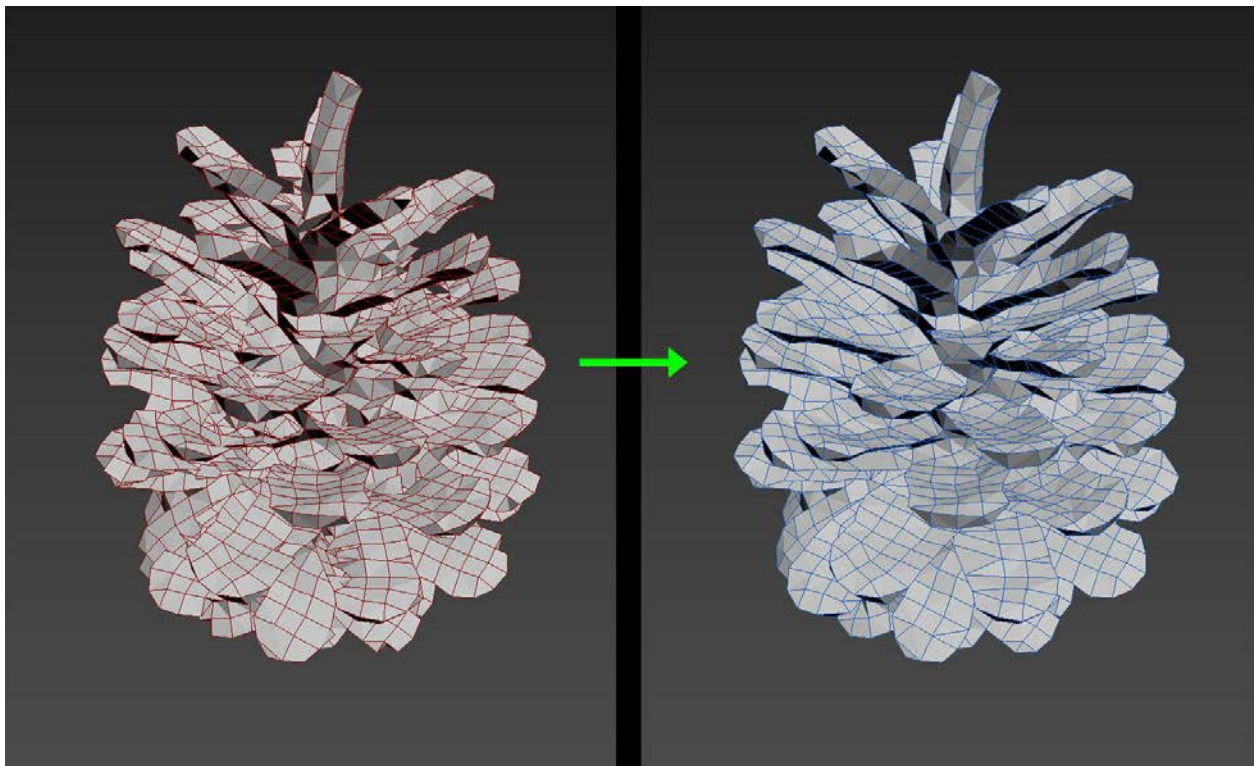


To save time, the medium resolution mesh can be processed with Instant Mesh software to produce the low resolution mesh. The result is often of good quality. Even if it isn't, the result can still be a good base to author the low resolution mesh for baking with 3dsMax.

Unity Photogrammetry Workflow



3dsMax cleaning:



5.3.3.2. UV layout

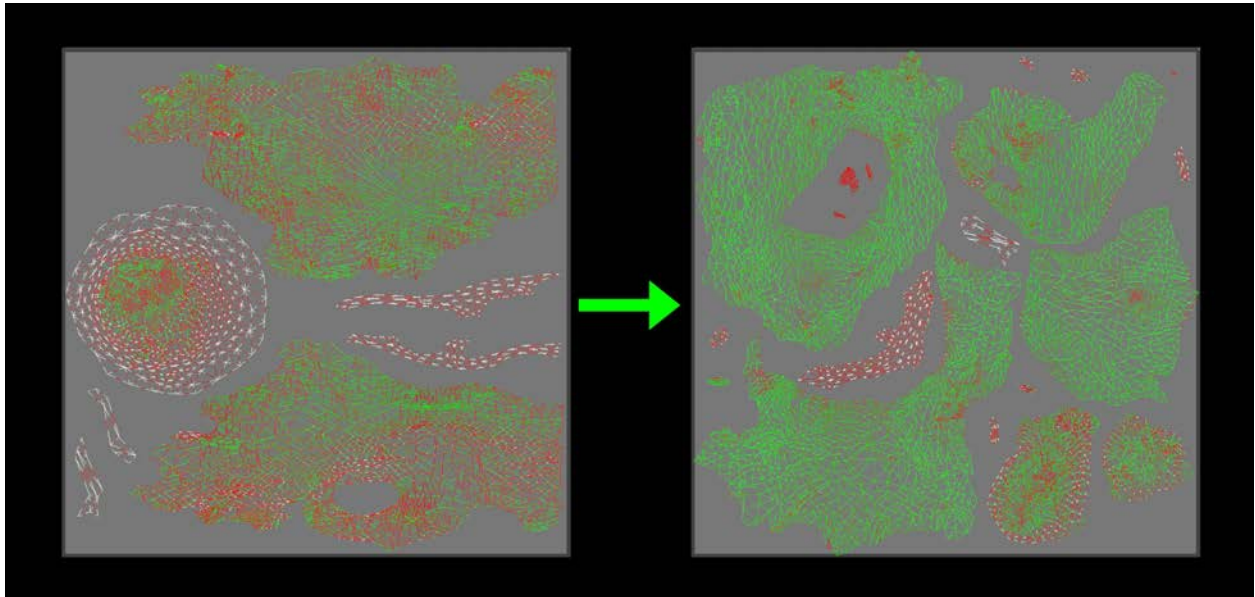
Note: With .ply format the UV1 channel is not used. With a .obj format that have a texture generated with Reality Capture UV1 is used.

Any UV tool can be used to do UV layout.

Try to have a minimum of seams (by lowering the number of UV clusters) and use all the available space. Keep a homogeneous ratio between polygon size and equivalent UV space. It is important to

have similar texture resolution for all visible parts of the mesh. The UVs have a large influence on the quality of the final asset, so don't hesitate to invest time on it.

Check edge distortion (Unwrap/display/Show Edges distortion) and minimize it by creating new seams and using UV relax.



(Red color represents a significant edge distortion).

When the UV layout is done, export the mesh in .obj or .fbx format with no scale, rotation or position modification. Take care to export with a scale at 1 and ensure that flip axis is unchecked.

5.4. Textures baking - Knald - Xnormal - Designer

We recommend Knald for the baking process as it is both fast and accurate. Of course other tools like Xnormal or Allegorithmic Substance Baker can be used, however they take more time to process the baking. It is recommended to use the same baking software for all the textures of an asset.

Note: Allegorithmic Substance Designer doesn't support .ply format which is a limitation for the presented workflow. To achieve baking with Designer, export to .obj format + color texture ([5.2.8. Colorize or texture](#)).

Note 2: For now, smoothness maps are not baked. Automatic authoring of smoothness maps is a difficult thing, and for now in the presented workflow it needs to be done by hand, using photos with flash as references.

5.4.1. Knald v1.2.1 workflow

Note: In the 1.2.1 version high mesh is limited to 350 M vertices in one mesh.

Open Knald.

CTRL+B to open baker tab.

Baker tab:

Set Anti-aliasing to 16X (for textures with a game ratio at 1024/m and lower.

Set the size wanted.

Assign the high mesh.

Assign the low mesh for baking.

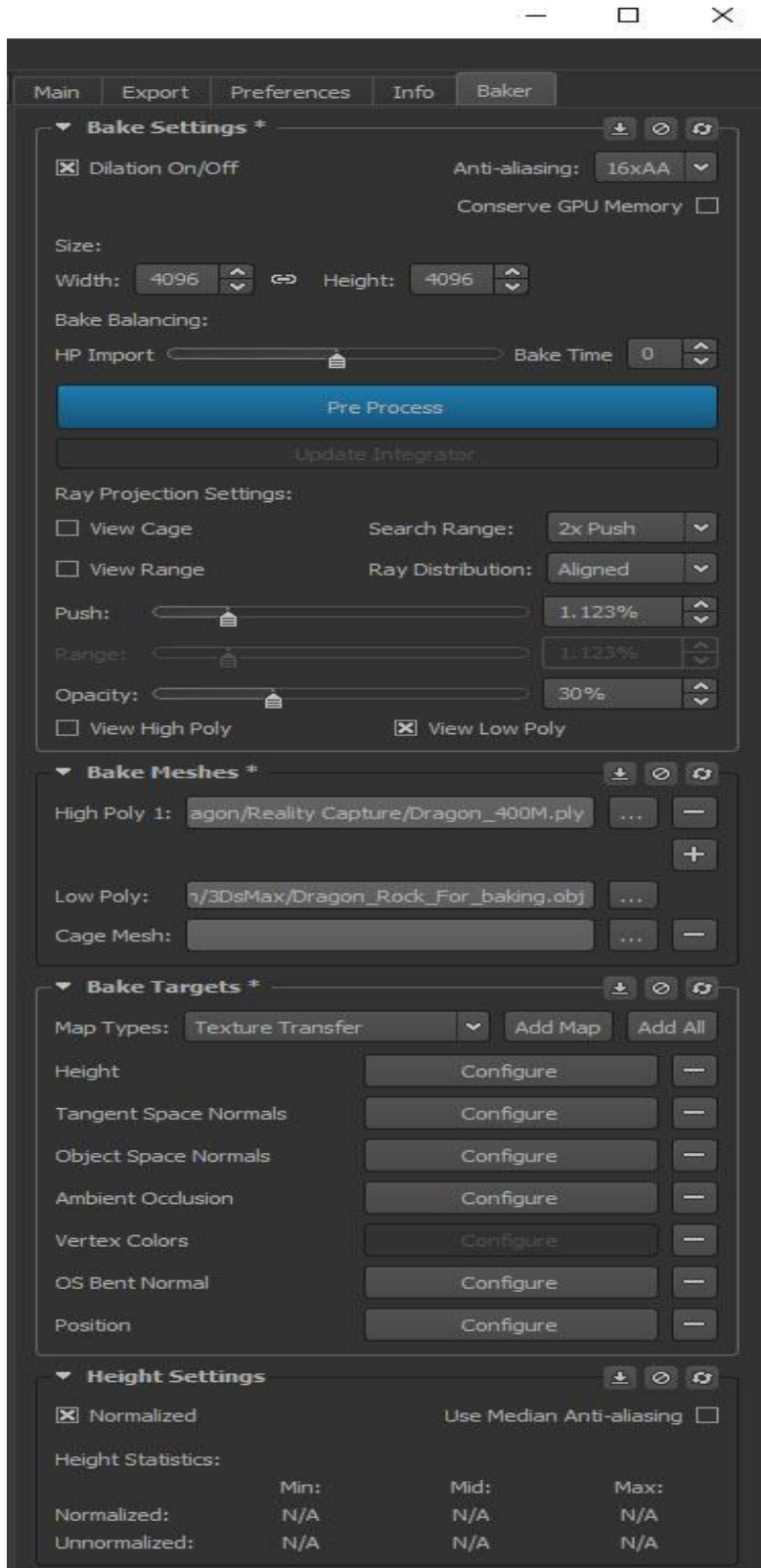
Add Bake target with default setting:

- Height: used in real time material.
- Tangent space Normals: used in real time material as normal map.
- Object space Normals: used to remove lighting.
- Ambient Occlusion: used to remove lighting and used in real time material.
- Vertex color: The information from the vertex color will be transferred to a texture, and allows creation of the lit diffuse albedo texture.
- Object space Bent Normal: used to remove lighting.

Unity Photogrammetry Workflow

- Position: Used to remove lighting. Needed only if UV are done with a lot of clusters. Normalization should be set to "Uniform".

Unity Photogrammetry Workflow



Picture above: Baker tab

Preferences tab:

Set your prefix/suffix nomenclature for textures.

Export tab:

Set global Export as TGA format. TGA is a safe 8-bit format correctly managed by all software.

TIFF format has some different gamma settings in different software.

PNG format has no alpha channel when it is saved by Photoshop.

Choose an export path.

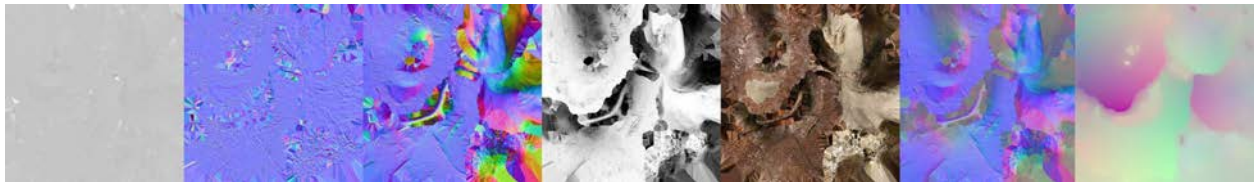
Main Tab:

Enter an export name. The prefix and suffix set in preferences are added to this name.

Use the preprocess button into the baker tab to start the baking.

Export all.

Example of all exported map:



5.4.2. Xnormal v3.19.3 workflow

High definition meshes tab:

- Add high resolution mesh.
- To generate the base color from the .ply vertex color: uncheck Ignore per-vertex-color. This option must be checked for all other baking.

Low definition meshes tab:

- Add low resolution mesh.

Baking option tab:

Name the output. A suffix will be added depending on the map baked.

Use edge padding, and specify a non-zero value.

Set the size.

Set Anti-aliasing. It is advised for textures with a game ratio at 1024/m and lower.

Maps to render:

Unity Photogrammetry Workflow

- Normal map: set tangent space on/off, depending if it will be the tangent or world normal map wanted. The two normal maps can't be done at the same time.
- Height map: if Raw values are used, exr format is needed and middle is at 0. Interactive and Manual use 8-bit format with a middle at 0.5.
- Ambient occlusion: distribution sets to cosine. Spread angle set to 179.5°.
- Bent normal: distribution sets to cosine. Spread angle set to 179.5°.

Unity Photogrammetry Workflow



- Bake high-poly vertex colors. This map must be generated alone due to a different setting in the high mesh (Ignore per-vertex-color)

Unity Photogrammetry Workflow



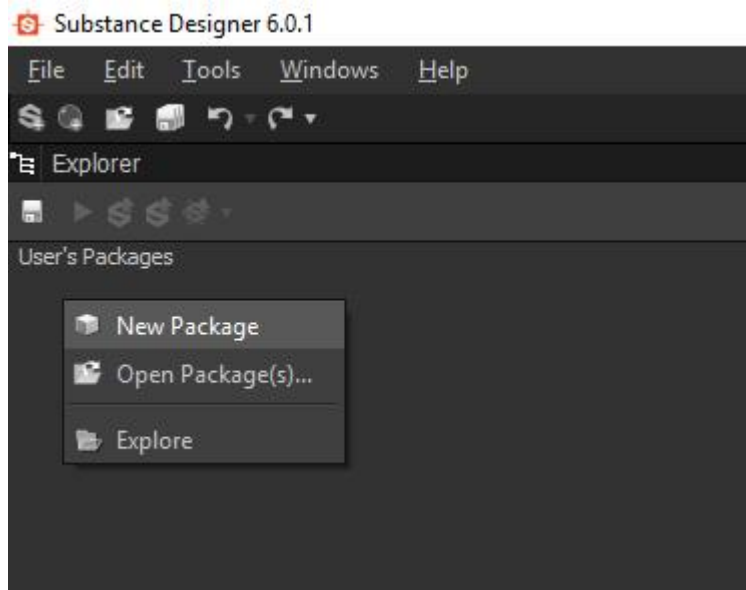
Note: Position map doesn't exist in Xnormal.

Use "Generate Maps" button to start baking

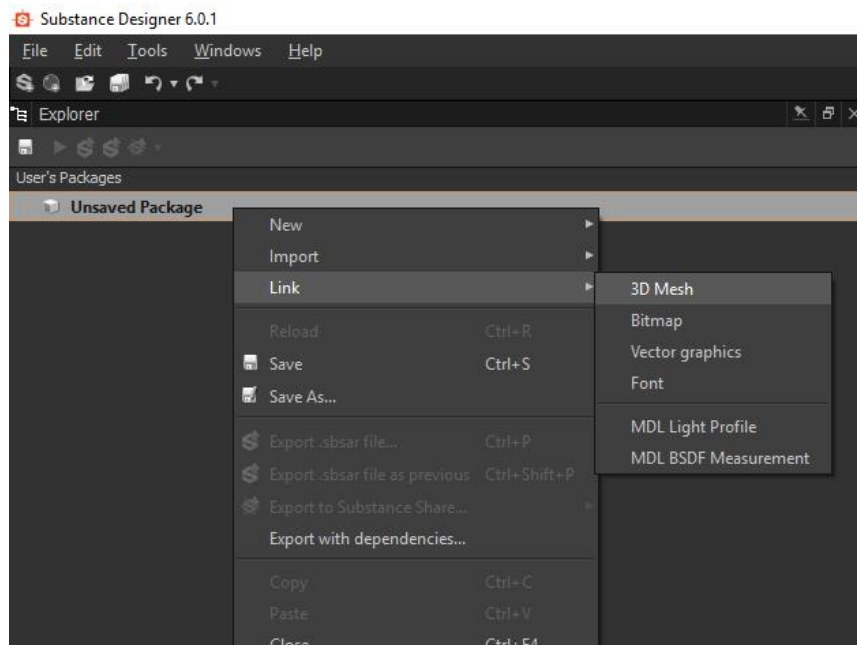
5.4.3. Substance Designer 6

Create a new package:

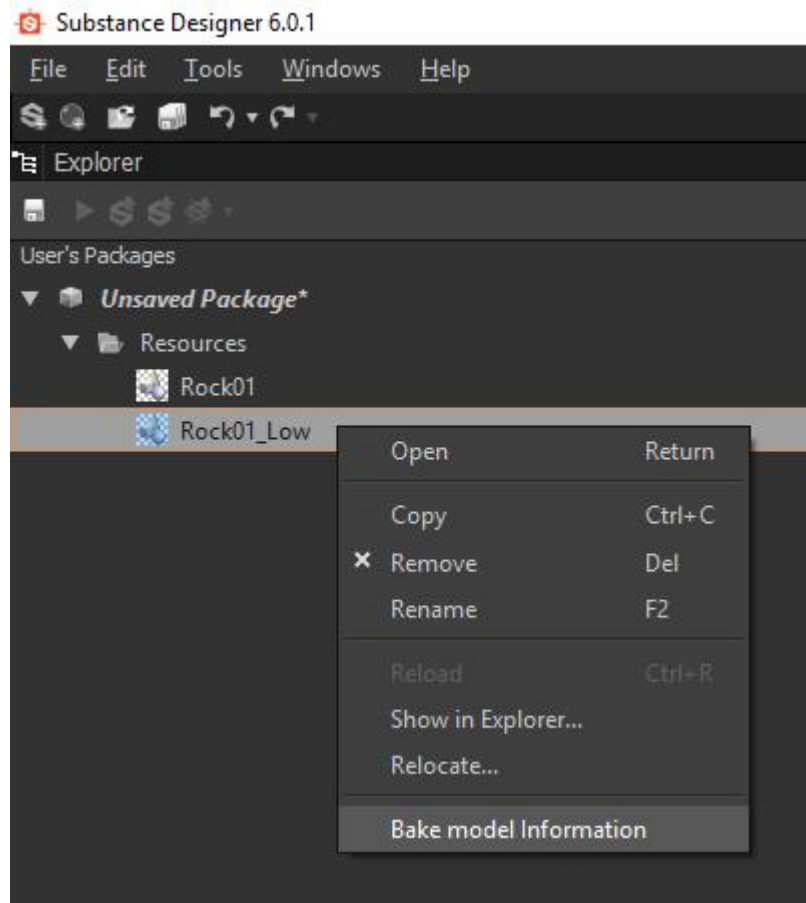
- Right-click in the explorer view and select “new package”



- Right-click on the new package and select Link / 3D Mesh

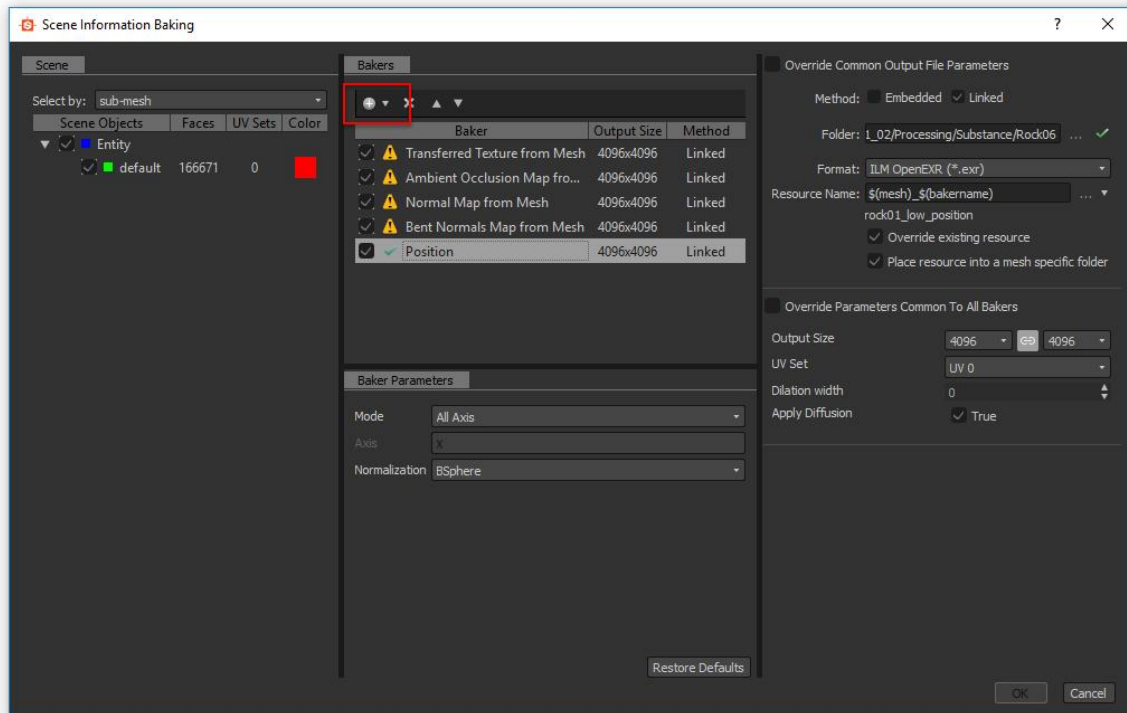


- Right-click on the low polygon mesh and choose Bake Model Information



Unity Photogrammetry Workflow

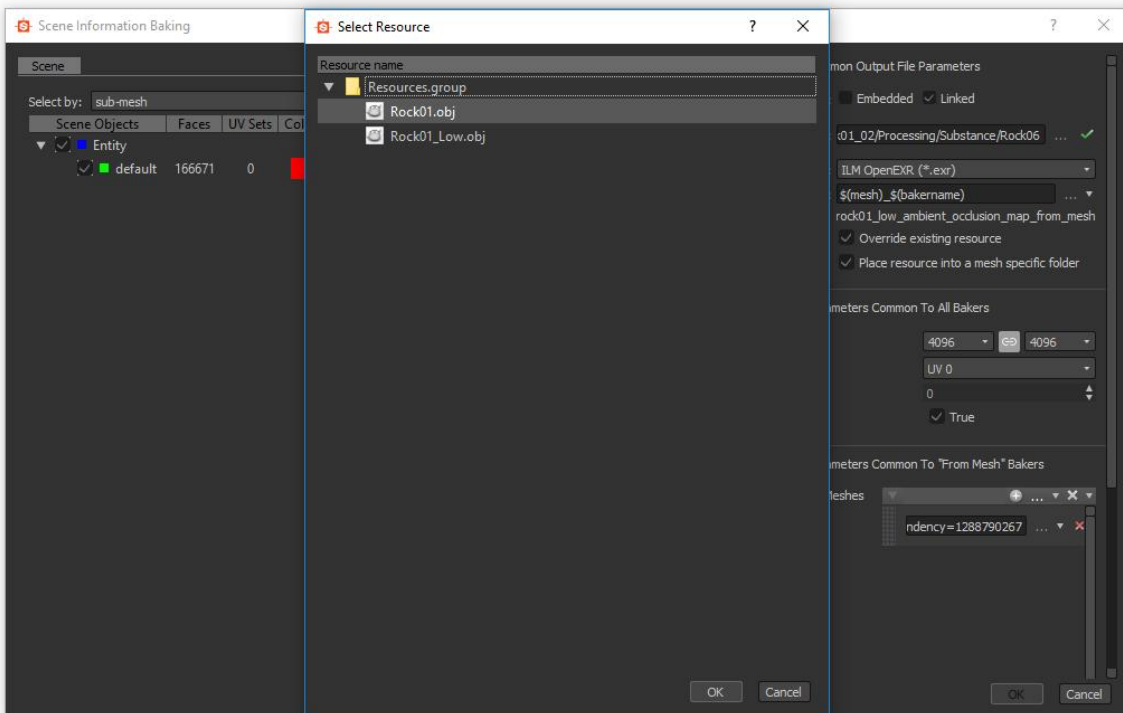
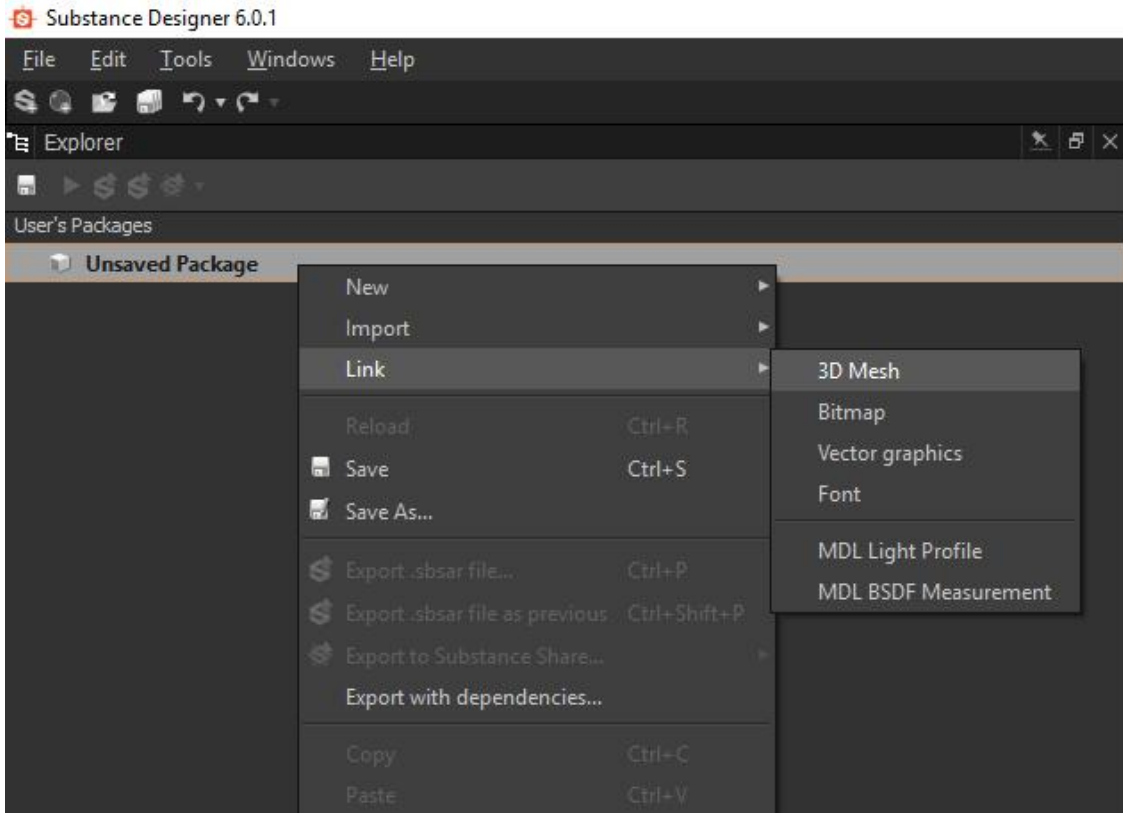
- Add those bakers:
 - Color Map from Mesh
 - Ambient Occlusion from Mesh
 - Normal Map from Mesh
 - Bent Normals Map from Mesh
 - Position



Note that all the bakers that are “from Mesh” display a warning sign. This is because the HD geometry is not referenced.

Unity Photogrammetry Workflow

- Click on one of those, then reference the HD mesh in the right panel



- Here are the settings for each channel:

Bakers

Baker	Output Size	Method
<input checked="" type="checkbox"/> ✓ Transferred Texture from Mesh	4096x4096	Linked
<input checked="" type="checkbox"/> ✓ Ambient Occlusion Map fro...	4096x4096	Linked
<input checked="" type="checkbox"/> ✓ Normal Map from Mesh	4096x4096	Linked
<input checked="" type="checkbox"/> ✓ Bent Normals Map from Mesh	4096x4096	Linked
<input checked="" type="checkbox"/> ✓ Position	4096x4096	Linked

Baker Parameters

Texture File:

UV Set:

Filtering Mode:

Normal Map: False

Map Type:

Normal Orientation:

Bakers

Baker	Output Size	Method
<input checked="" type="checkbox"/> ✓ Transferred Texture from ...	4096x4096	Linked
<input checked="" type="checkbox"/> ✓ Ambient Occlusion Map f...	4096x4096	Linked
<input checked="" type="checkbox"/> ✓ Normal Map from Mesh	4096x4096	Linked
<input checked="" type="checkbox"/> ✓ Bent Normals Map from ...	4096x4096	Linked
<input checked="" type="checkbox"/> ✓ Position	4096x4096	Linked

Baker Parameters

Secondary Rays:

Min Occluder Distance:

Max Occluder Distance:

Relative to Bounding Box: True

Spread Angle:

Distribution:

Ignore Backface: True

Self Occlusion:

Attenuation:

Normal Map:

World Space: False

Normal Orientation:

Unity Photogrammetry Workflow

Bakers

Baker	Output Size	Method
<input checked="" type="checkbox"/> ✓ Transferred Texture from Mesh	4096x4096	Linked
<input checked="" type="checkbox"/> ✓ Ambient Occlusion Map fro...	4096x4096	Linked
<input checked="" type="checkbox"/> ✓ Normal Map from Mesh	4096x4096	Linked
<input checked="" type="checkbox"/> ✓ Bent Normals Map from Mesh	4096x4096	Linked
<input checked="" type="checkbox"/> ✓ Position	4096x4096	Linked

Baker Parameters

Secondary Rays: 64

Min Occluder Distance: 0.00001

Max Occluder Distance: 0.1000

Relative to Bounding Box: True

Spread Angle: 180.0

Distribution: Cosine

Ignore Backface: True

Self Occlusion: Always

Map Type: World Space

Normal Orientation: DirectX

Restore Defaults

Bakers

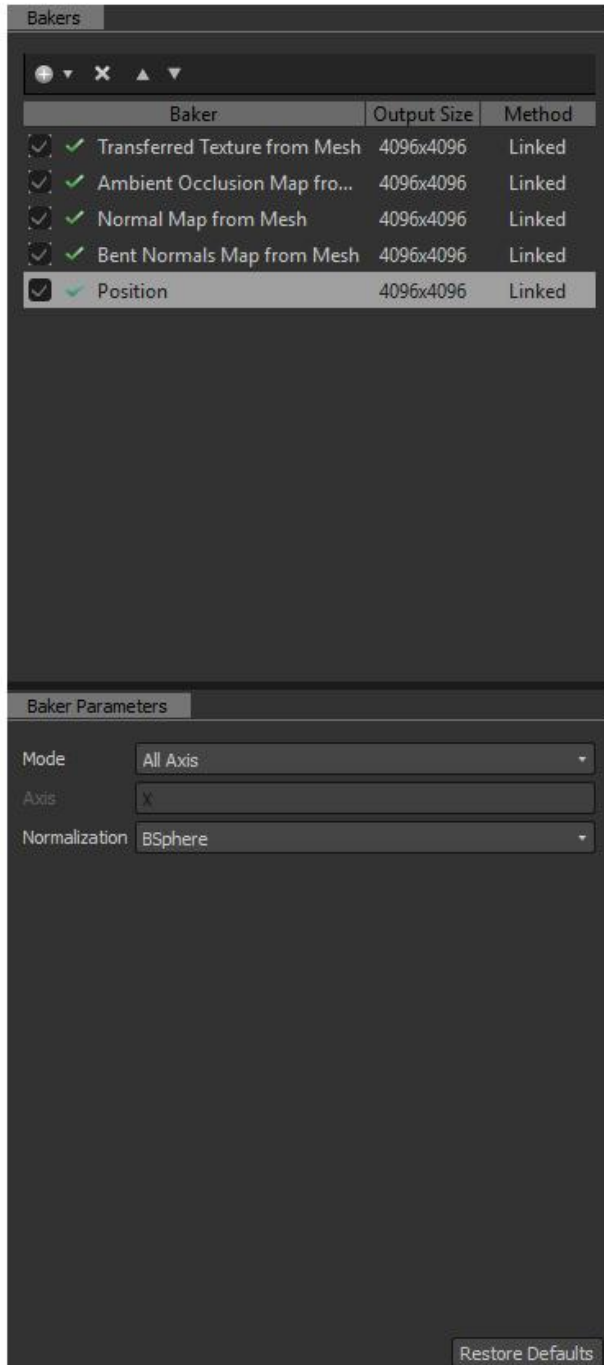
Baker	Output Size	Method
<input checked="" type="checkbox"/> ✓ Transferred Texture from Mesh	4096x4096	Linked
<input checked="" type="checkbox"/> ✓ Ambient Occlusion Map fro...	4096x4096	Linked
<input checked="" type="checkbox"/> ✓ Normal Map from Mesh	4096x4096	Linked
<input checked="" type="checkbox"/> ✓ Bent Normals Map from Mesh	4096x4096	Linked
<input checked="" type="checkbox"/> ✓ Position	4096x4096	Linked

Baker Parameters

Map Type: World Space

Normal Orientation: DirectX

Restore Defaults



- The last step is to choose an output folder and to set the right format for the baked maps. All should be 8 bit (e.g. tga) except position map (8 bit work but 32 bit (e.g. exr) provide more precision).

5.5. Remove lighting - Unity - Photoshop

The workflow presented in this document is for game assets that will be dynamically lit in a game level. Photos contain lighting information from the environment of the shot at the time of the shoot. In order to be able to apply virtual lighting, it is required to remove all lighting information from the created textures. The process is explained in this section.

5.5.1. Unity

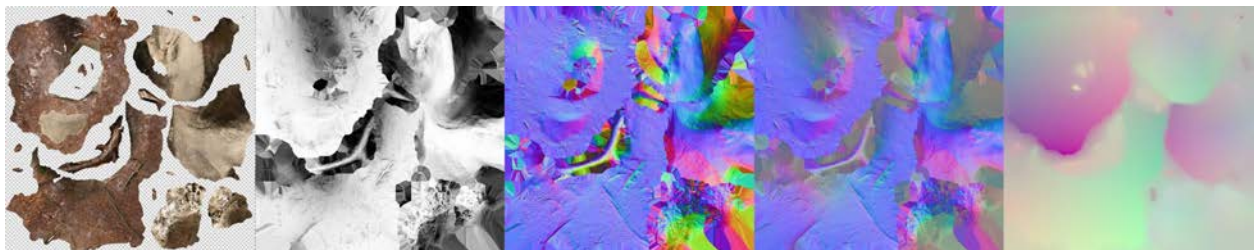
There are multiple ways of un-lighting a texture. Some of these ways require more information about the environment lighting while doing the shoot. Time saving is important for production and Unity has developed a tool to do light removal using only the base color coming from the baking (See: De-Lighting Tool guide).

The advantages of this tool are:

- Fast
- Removes many light sources
- Removes colored lights
- Removes environment lighting

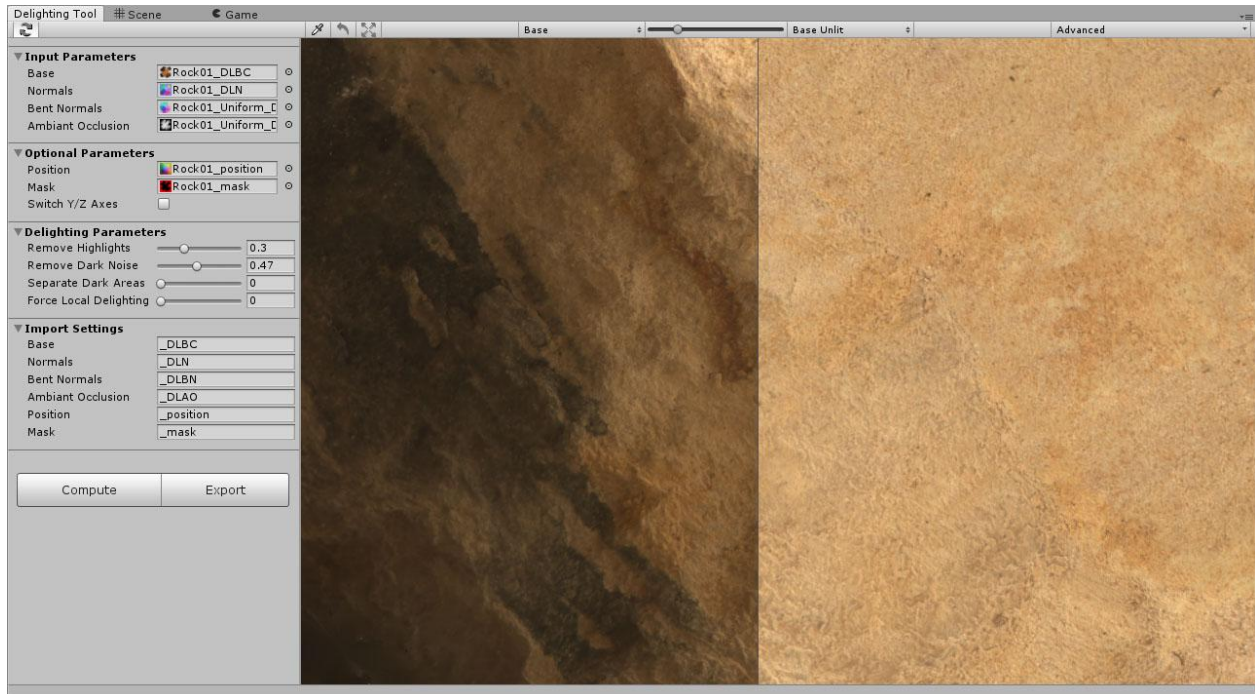
Import in Unity the 4/5 maps needed to do light removal:

- Base color with alpha representing the clusters
- Ambient occlusion
- Object space normal
- Object space bent normal
- Position if needed



Use imported textures in the light removal tool.

Unity Photogrammetry Workflow



Edit masks if needed (See: De-Lighting tool documentation).

Export the result.

If some artefacts are still present, use Photoshop to clean them.

The result depends on the quality of the baking (and de facto, on the quality of the reconstruction). Dark areas give bad results.



Some artefacts can appear and shadows are difficult to remove (red shapes on the previous image).

5.5.2. Photoshop

The Unity light removal tool is not perfect and some textures can be challenging to remove light automatically. In this case Photoshop can be used to remove lighting artistically (i.e based on artists skills rather than an automatic process). It is more time consuming and in cases with many lights, colored lights or significant global illumination, it can be difficult to do. The light removal process in Photoshop is based on the world normal and ambient occlusion map. The light removal is done on a 32-bit format.

Light removal process:

- Load the base color from the baking.
- Switch to 32-bit format
- Create an exposure layer with the ambient occlusion map inverted as mask. It is inverted because it is better to lighten dark areas than darken the rest of the image.
- Create an exposure layer with the world normal red channel.
- Create an exposure layer with the world normal green channel.
- Create an exposure layer with the world normal blue channel.
- Manage the global and directional lighting with these exposure layers.
- Add some exposure layer to manage highlights. These highlights can be isolated easily with high pass filters (Filters/Others/High Pass).
- Add a last exposure layer to manage all the exposure compared to the albedo photo reference (See section [How to capture the diffuse albedo texture only](#))

In this example, you can see how environment lighting is still influencing the base color (red shapes):

Unity Photogrammetry Workflow



Combining both light removal solutions in Photoshop

A third path is to use the output of the Unity light removal tool and if it is not good enough, it can be combined with the manual Photoshop method to produce the desired result:



5.6. Tileable material - Artomatix - Substance painter

Some game assets require tileable textures to be reusable. Tiling is time consuming and to help this process, we recommend two different approaches: Artomatix can be used as an automatic workflow to produce tileable materials. Allegorithmic with Substance painter can be used as an artistic workflow.

5.6.1. Artomatix

At the time of the writing, the Artomatix services are done via their own web site.

Workflow:

Access the Artomatic web site.

Choose remove seams.

Upload the textures:

- Base color from light removal process
- normal map from baking
- height map from baking
- Ambient occlusion map from baking

Set Border and masks.

In few minutes a tileable material can be created.



The right image is a tileable texture completely regenerated by Artomatix based on the original at left image. If some patterns are too visible, they can be removed by some masking.

The process is fast, but with the current version, the result can exhibit too many identical patterns. They can be removed with Substance painter.

5.6.2. Substance painter

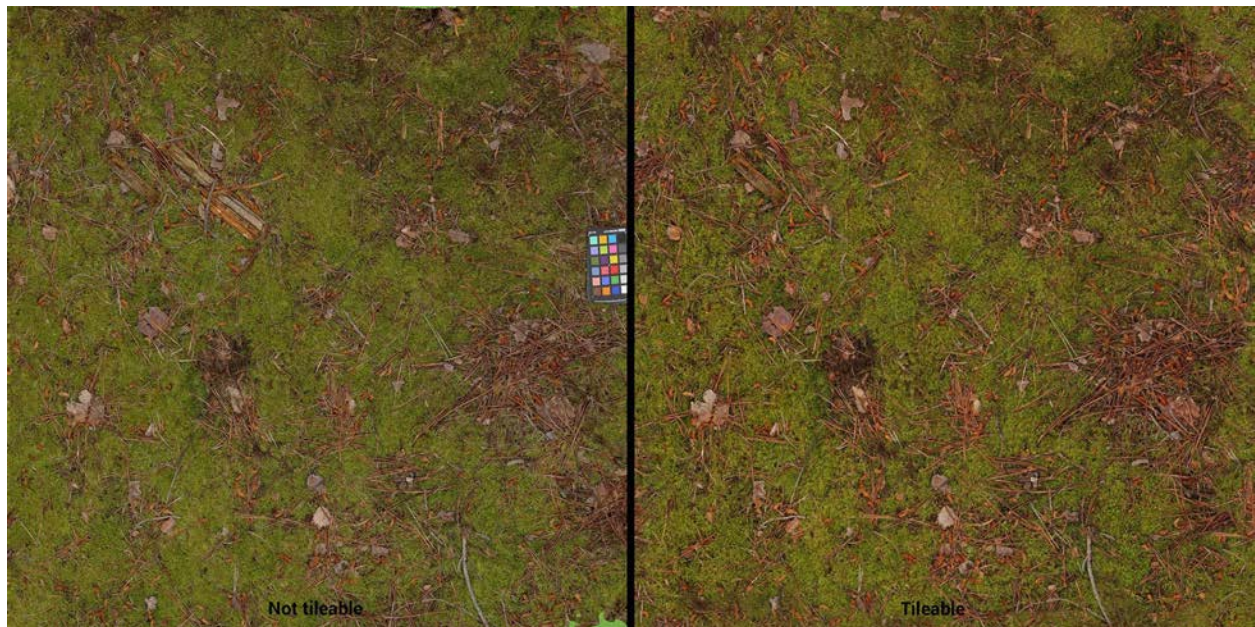
Substance painter allows the artists to paint a material on a 3D object.

It can be used to remove Artomatix artefacts or use directly to produce the tileable material.

Use a plane with a 3x3 tile as 3D mesh. The tile is needed to manage the transitions between the borders.

Set your material in painter (See Substance Painter help).

Use Painter clone tool to do a tileable material (See Substance Painter help).



With the clone tool the borders are tiled, the color checker is erased, and some unwanted patterns are replaced.

5.7. Create a game asset - 3dsMax

Use the low resolution mesh done for baking textures as the base for a game asset.

Don't modify the UV1 channel used for baking.

Place the pivot.

Set the mesh in 0, 0, 0.

Rotate it like it should be in game.

Apply a reset Xform.

A second UV set can be needed in case of material layering with tileables materials in Unity. In this case pay attention about UV direction.

Do Level of detail if needed.

In game asset result:



Rock asset plus tileable material layered, all are done with photogrammetry.

5.8. Foliage - Photoshop

The photogrammetry process for foliage requires some custom elements that needs to be done manually. It is possible to do more complex workflow but presently a process based on photos without any 3D reconstruction is advised to save time. The results are good enough for video games.

5.8.1. Base color

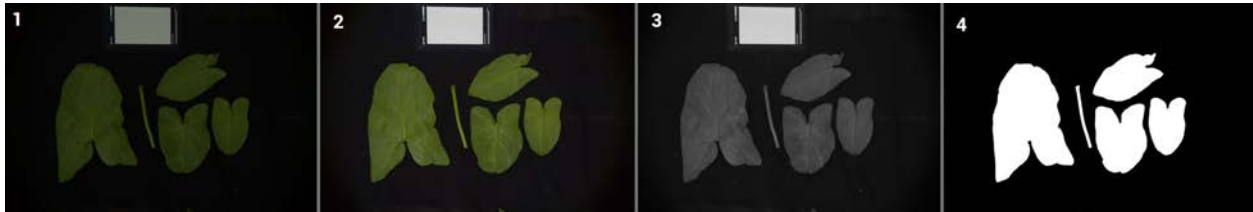
- 1- Open the photo. Convert to 32-bit.
- 2- White balance it. (See: [White balance](#)).

At this step the photo represents something close to the albedo.

The next step is the production of an alpha mask.

3-Convert the image to 8-bit then use Hue/saturation to set saturation at 0. This gives a grayscale image.

4- use Brightness/Contrast many times to do a black and white image. Gaussian blur can be used to erase the little artefacts from the dust. Remove the color checker and clean the last artefacts by hand.



This black and white texture can be used as a mask. Store it in the alpha channel of the diffuse albedo texture.

When there is a lot of foliage, it is more efficient to pack them into an atlas:



5.8.2. Normal map

To author a normal map for foliage, two normal maps are produced and combined. This approach is a bit of a hack but the results are often good enough. The Xnormal Photoshop plug-in is required.

The first normal map is based on the mask generated in the previous process. It will define the global shape.

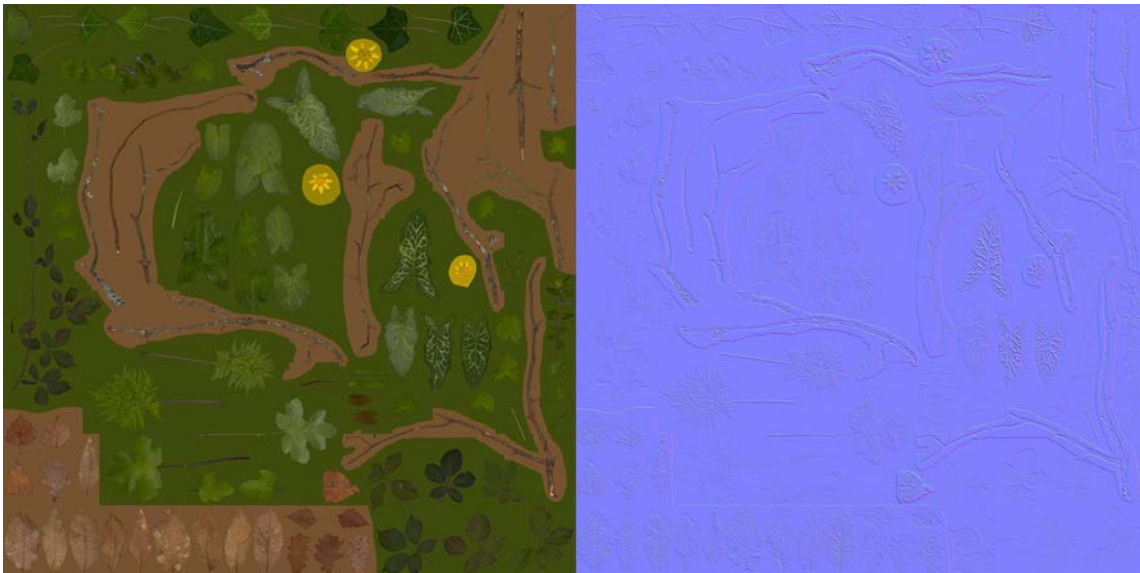
- Copy the mask to a new document
- Set mode as RGB Color
- Apply a gaussian blur to do rounded shapes
- Use Xnormal filter "Heights2Normals"



If the foliage is more complex, a simple 3D mesh can be done and baked to reproduce the global shape.

The second normal map uses the diffuse albedo to bring more details.

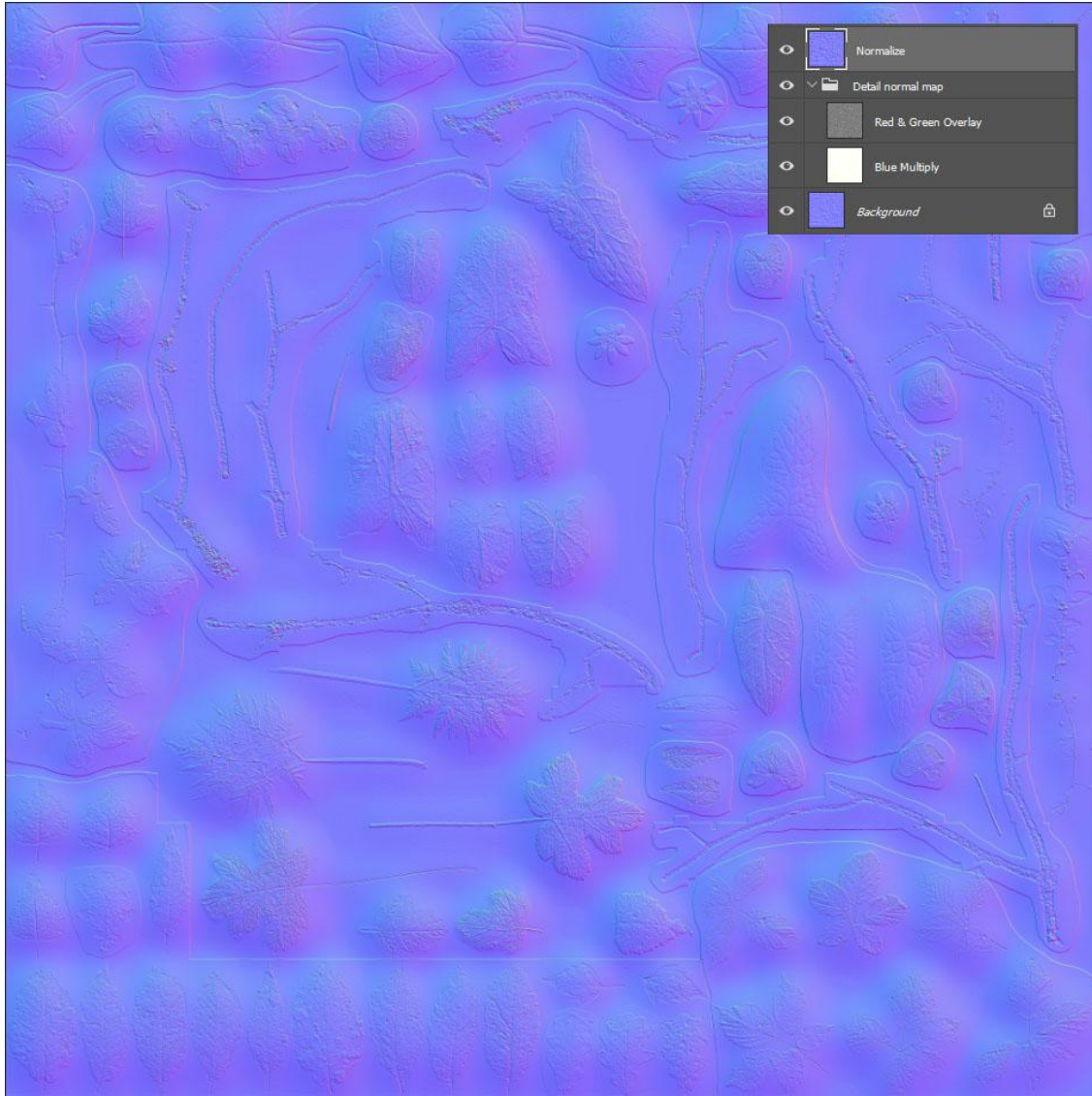
- Use Xnormal filter "Heights2Normals"



Mix the two normal maps:

- Load the normal map from shape,
- Create a group,
- Add a new layer with the detail normal map from albedo,
- Fill in white the red and green channel,
- Set the layer as multiply. Only the blue channel is kept.
- Add a new layer with the detail normal map from albedo,
- Fill in middle grey (128, 128, 128) the blue channel,
- Set the layer as overlay. Only the red and green channels are kept.
- Manage the group opacity to bring more or less details.
- Copy paste the result in a new top layer,
- Use Filter/Xnormal/Normalize normal map.

Unity Photogrammetry Workflow



Set your prefix/suffix nomenclature for textures.

The result is not accurate, but it is the simplest way to produce something good enough.

In game rendering:

