

INTRO

DISCLAIMER: This technique appears in many games nowadays, I consider IOI one of the first pioneers of adequately using this technique in Hitman Absolution that was in development in 2009, so:

I'M NOT REINVENTING THE WHEEL, or sell it as it was mine,
I'm just showing you how the wheel works :)

Back in the days, words like deferred rendering (see the end of the tutorial for more info about it), render, deferred shading pipeline, buffer, trim sheets etc were words only for super technical monsters of the industry :). Nowadays we are living in a time where a lot of information is in what people call the internet, and because of that that people in the industry are learning and getting more and more technical than artists years ago used to be, with that said information becomes easier to understand and to apply. At the end of this pdf, you will find all the links and references that can help you expand and better understand this topic but also all the information that is present in this tutorial scattered around.

Wrapping up I'm just tidily putting together understandable information that will help you understand the topic sourcing what is already present in various tutorials and articles online.

Ideally, with this, I want you to quickly learn the technical stuff that the job requires so that you can focus more on the art rather than finding solutions or fixing problems.

NORMAL EDGE DECALS

NORMAL EDGE DECALS are very useful when you want to break the rigidity of the geometrical lines that compose an object. These are commonly applied to architectural elements and props that use tileable textures. This allows you to keep the tileable material untouched whilst achieving some extra details such as scratches or minor destruction.

A SIMPLE EXAMPLE WITHOUT AND WITH NORMAL EDGE DECALS

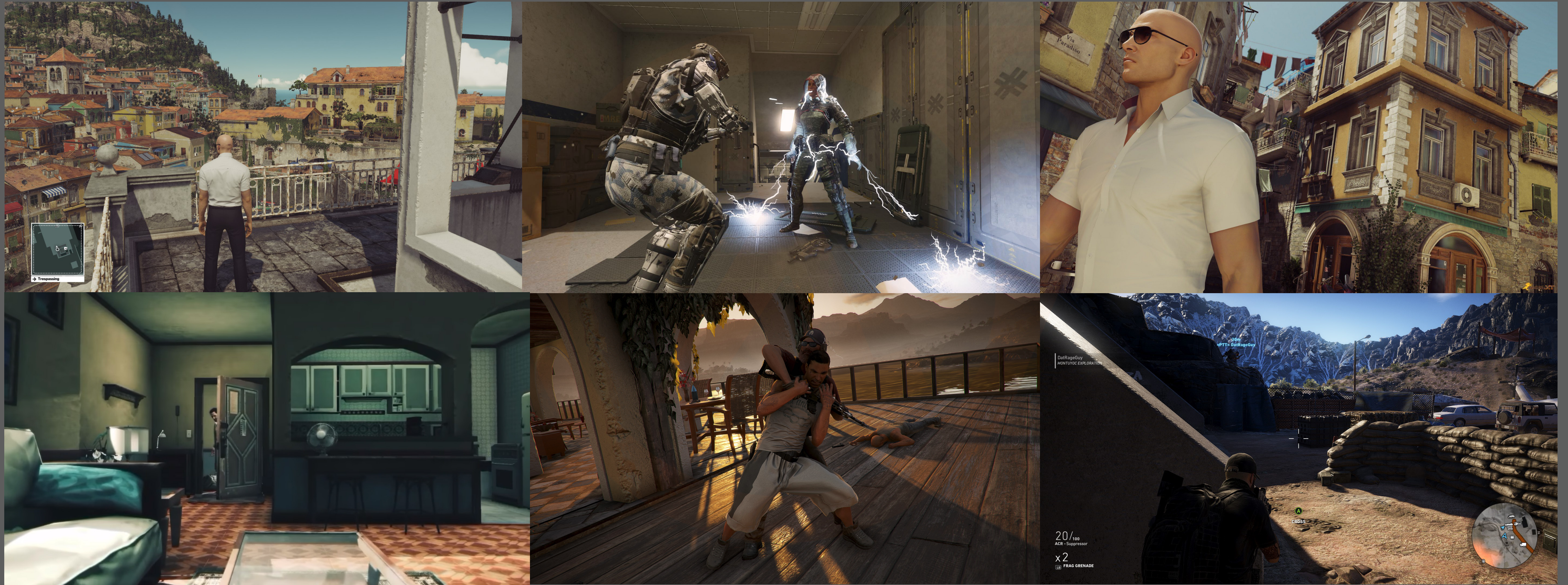


GEOMETRY ONLY



GEOMETRY WITH NORMAL EDGE DECAL

If you pay attention, you will find normal edge decals in pretty much all the next gen games you play as well as some oldy buy goldy ones. Here are some examples:



Let's have a look at the most obvious ones:



PLAN IT FIRST

Search for reference and plan what edge decals you want to create.

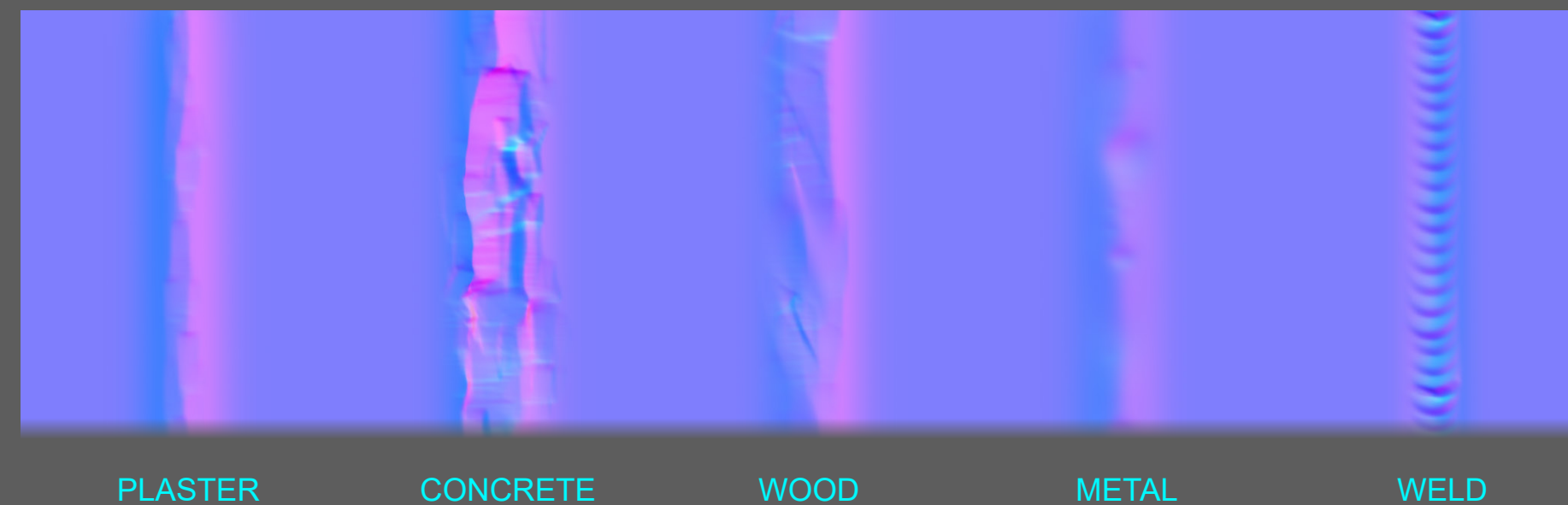


Ideally, you want enough variations to cover the majority of the tileable materials you are going to use on big scale objects such as architectural meshes: Walls corners, stairs, trims. These examples vary based on the type of project you are working on.

Based on the project you are working on and the number of details you want to get, you have two different directions when planning the trim:

I <3 PERFORMANCE

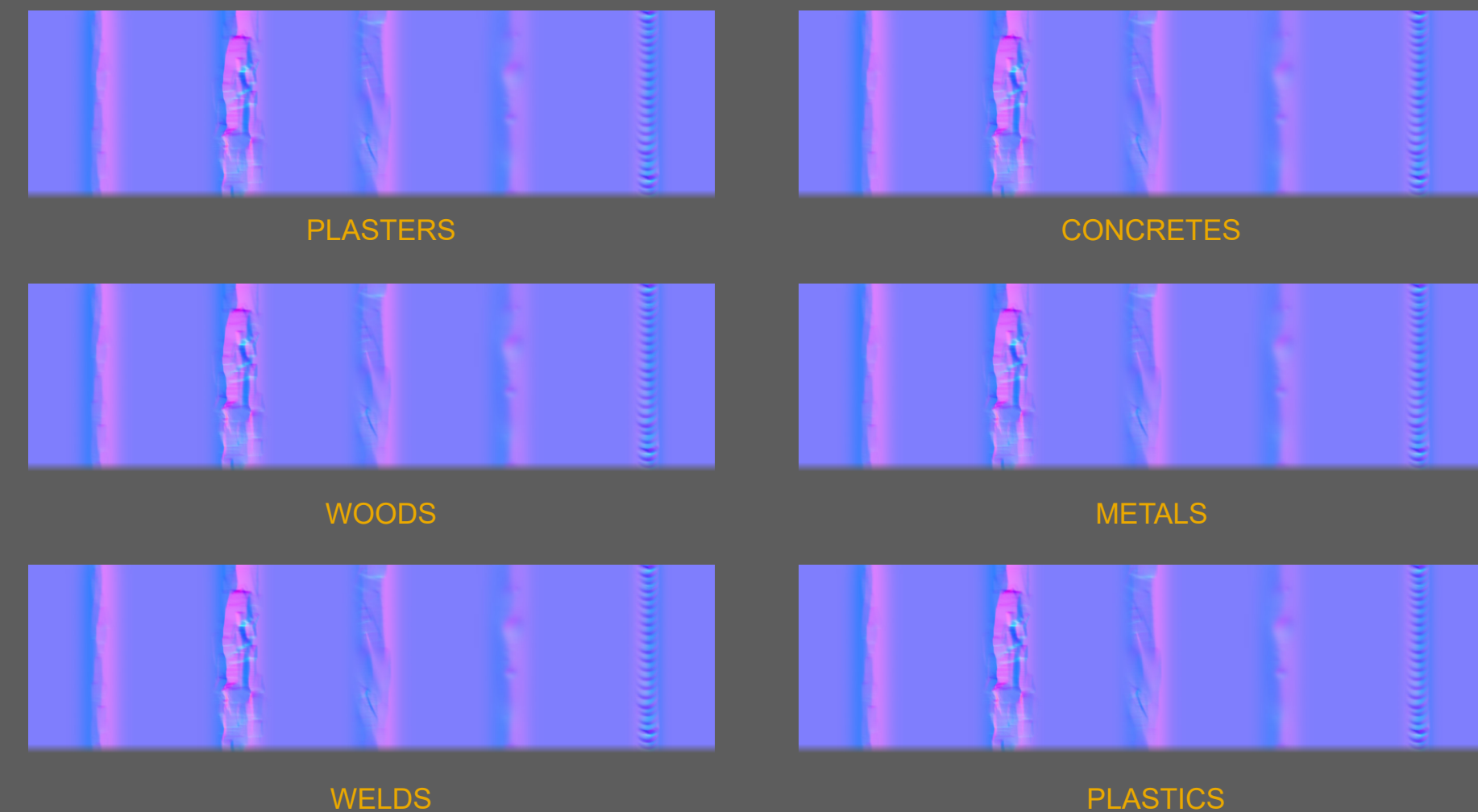
One texture that contains all the common type of materials: such as plaster, concrete, metal, wood etc. with a medium level of destruction



DETAILS MANIAC

One texture with destruction levels for each type of generic material.

* This is the same texture repeated to give you a better idea of what I'm trying to explain



PRO

BETTER PERFORMANCE

MORE VARIATION

THE TRUTH LIES SOMEWHERE IN THE MIDDLE

LIGHT DESTRUCTION

One texture that contains all the common type of materials with a light/soft state of destruction



PLASTER

CONCRETE

WOOD

METAL

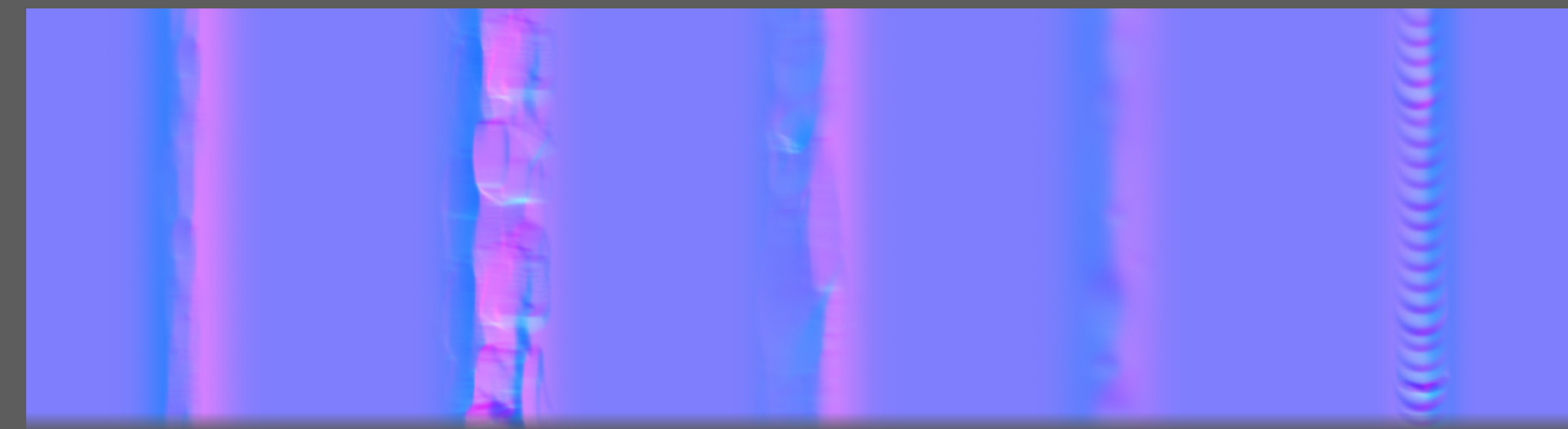
WELD



HEAVY DESTRUCTION

The same texture but with and havier state of destruction

* For the sake of this tutorial I just changed the intensity, but if you go with this scenario ideally you sould make another sculpt that match better with the state of destruction and not just changing the intensity ;)



PLASTER

CONCRETE

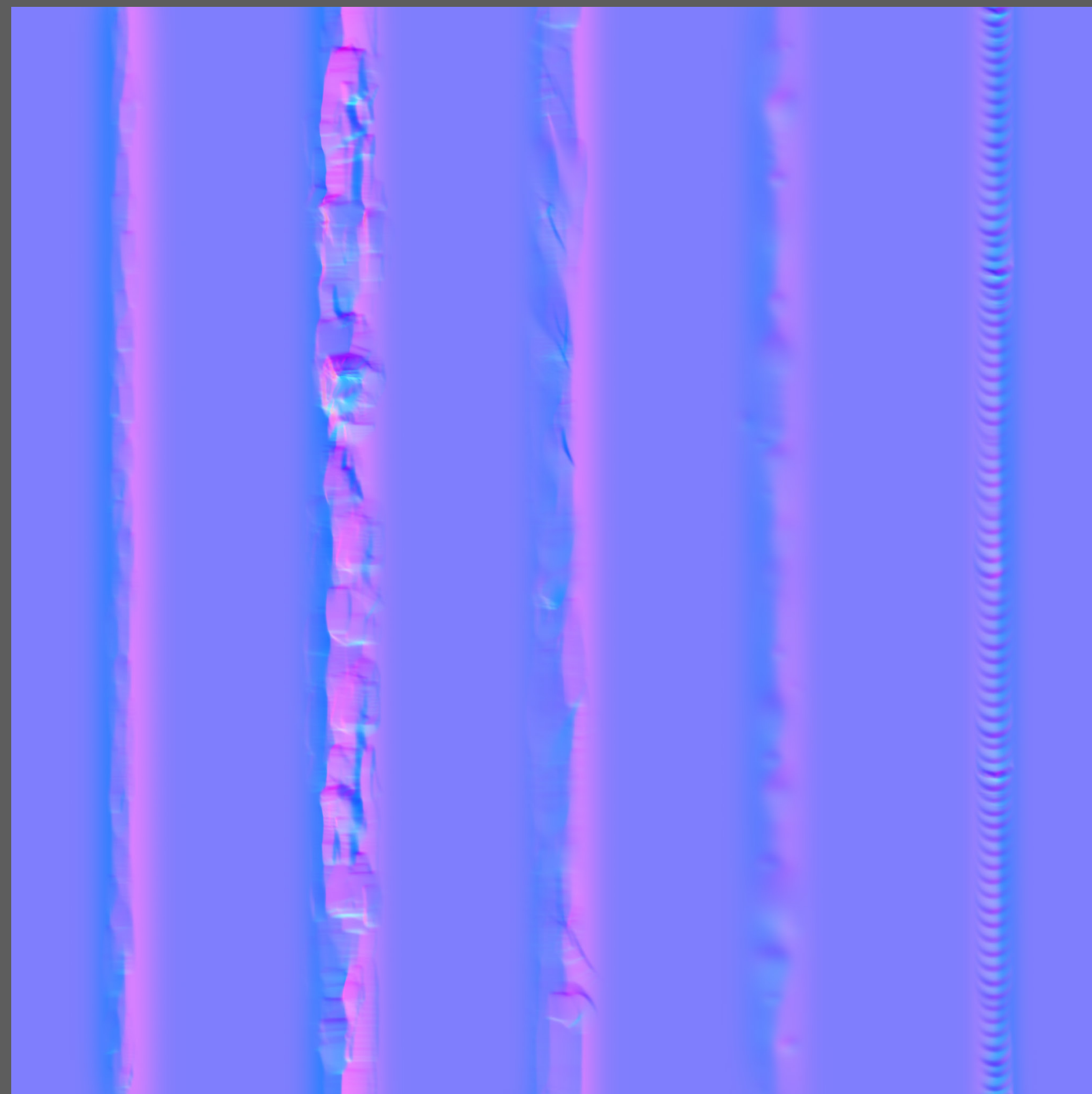
WOOD

METAL

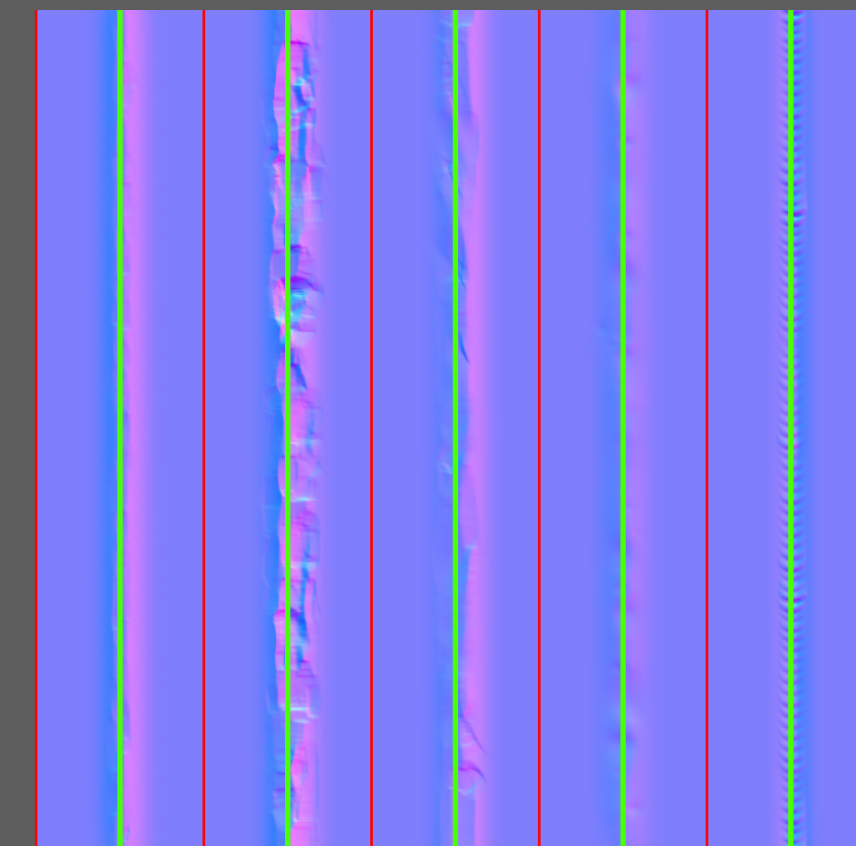
WELD

With this option, you have a wide range of destructions type for most of the conventional materials whilst ensuring that the number of textures is low. **NOTE: I OBVIOUSLY CHOOSE THESE MATERIALS AS THEY ARE THE MOST COMMON GENERIC TYPE OF MATERIALS**

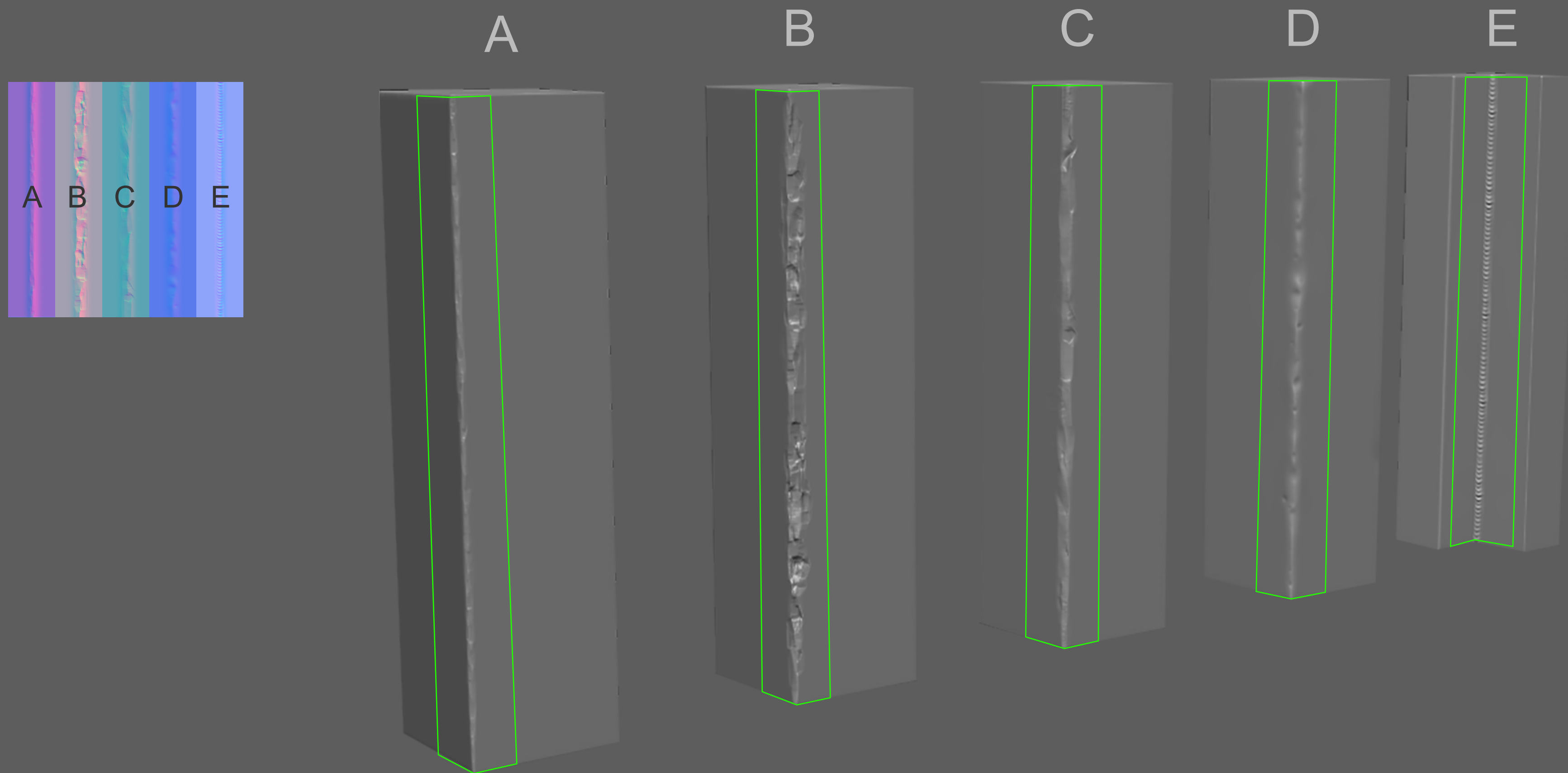
HOW TO CREATE A NORMAL SHEET



Our final goal is to reach a map that look like this (left), to do that divide the texture in 5 (or more) slots **A**, **B**, **C**, **D**, **E** and get the normal information map in the center of each slot giving enough room to fade before reaching in the near slots. **NOTE: THE TRIMS HAVE BEEN OBVIOUSLY PLACE VERTICAL BECAUSE IT MAKE SENSE SINCE WE ARE GOING TO APPLY MAINLY ON VERTICLA CORNERS, FEEL FREE TO PLACE DIAGONALLY IF YOU FEEL LIKE IT :)**



! WE ONLY CREATE ONE (OR TWO DEPENDING WHAT YOU PLANNED) NORMAL SHEET, FOR EVERYTHING ONCE. YOU SHOULDN'T CREATE A NEW ONE FOR EACH ASSET YOU MAKE; INSTEAD, YOU SHOULD RE-USE THIS ONE.



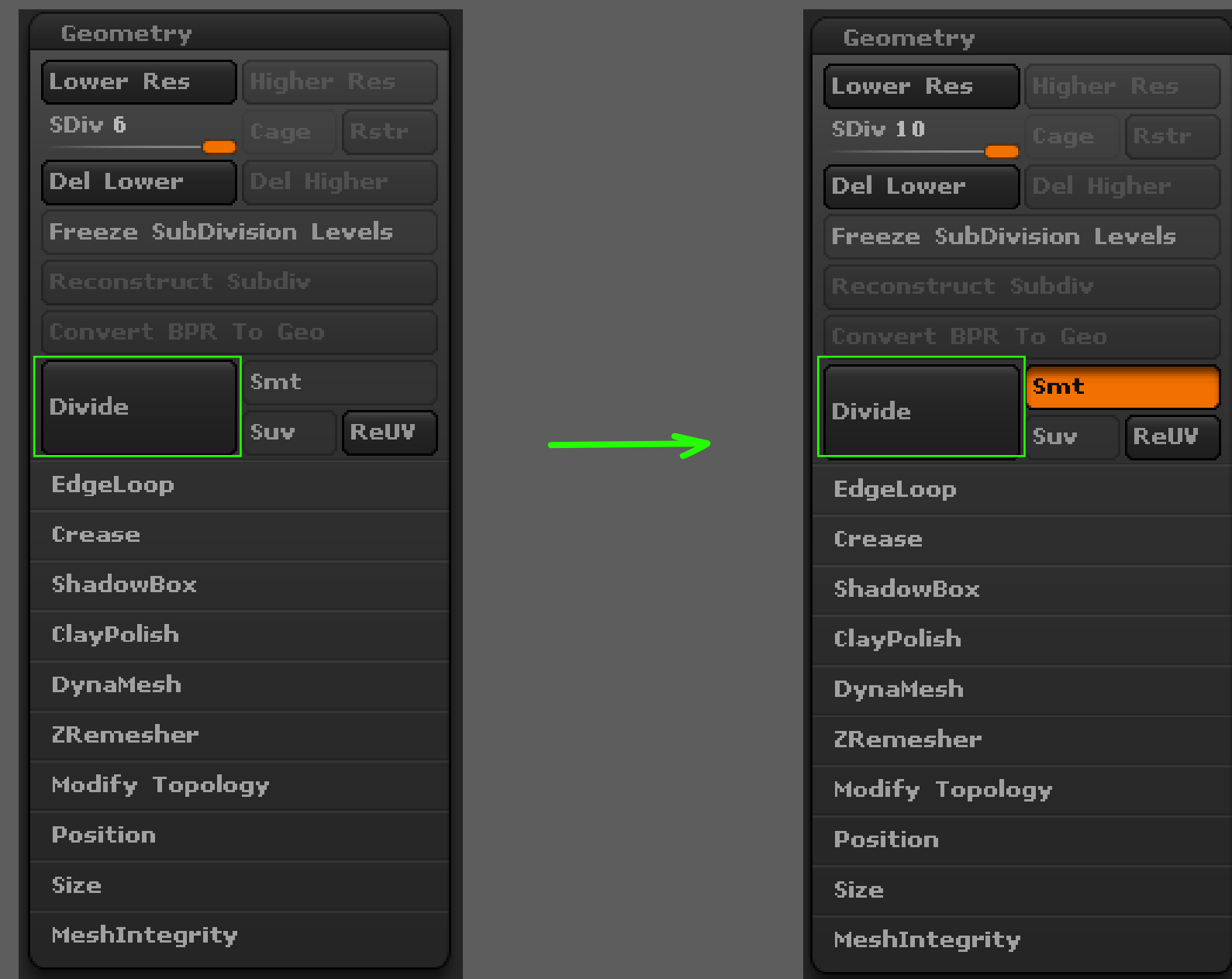
Simply create some primitive meshes and sculpt the corners the way you want the normal map on the decal to look like. Then create the low poly version of that corner. Make sure that you cover the area of each slot with a lowpoly mesh (see "low poly and bake" page).

NOTE: THERE ARE DIFFERENT WAYS TO ACHIEVE THIS, THAT'S THE ONE THAT IT'S MOST STRAIGHT FORWARD TO UNDERSTAND

ZBRUSH WORKFLOW 101

OPTION A

Import your mesh in ZBrush and under Geometry press divide few times (5-6 times) then activate **SMT** (smooth) option and press other few times so that you can smooth a bit the edges which will help when sculpting it



OPTION B

Alternatively, you can subdivide your mesh a few times with the SMT (smooth) option disabled and then under Deformation you play with the first deformer. This will allow you to have better control over the smoothness of the edge.



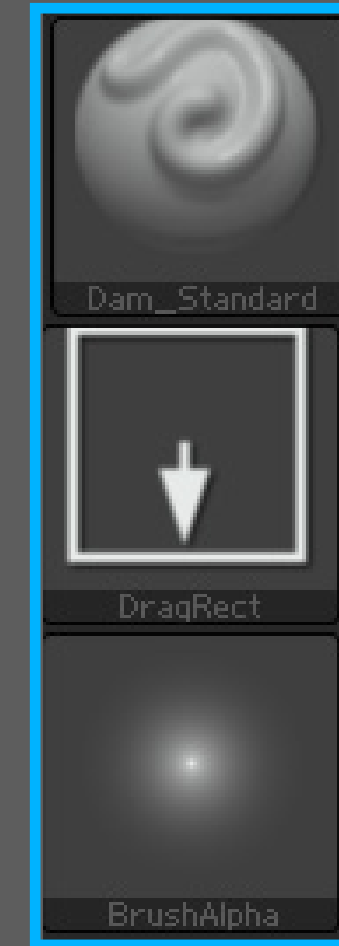
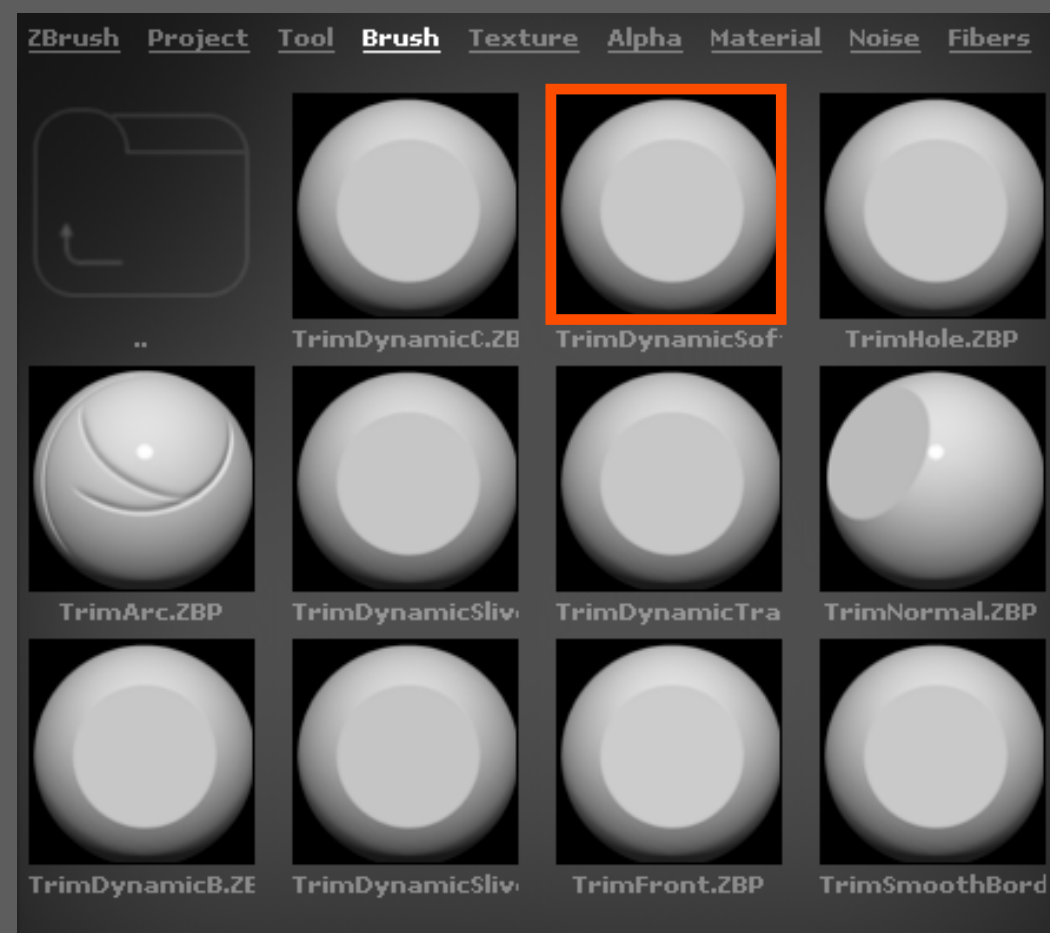
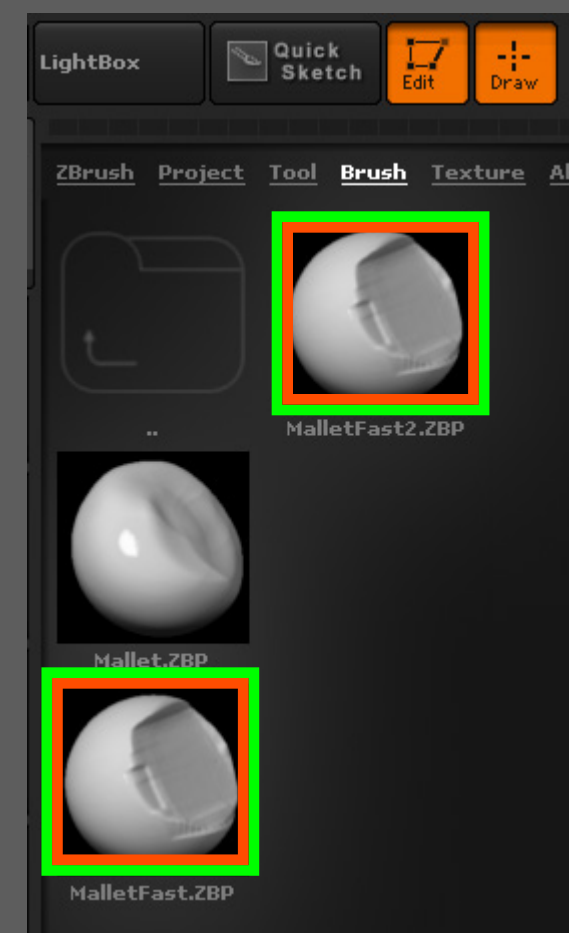
Here are some of the brushes that I used to create the normal map for this tutorial. There are endless ways to achieve the same thing in ZBrush.

WOOD AND PLASTIC

CONCRETE

METAL

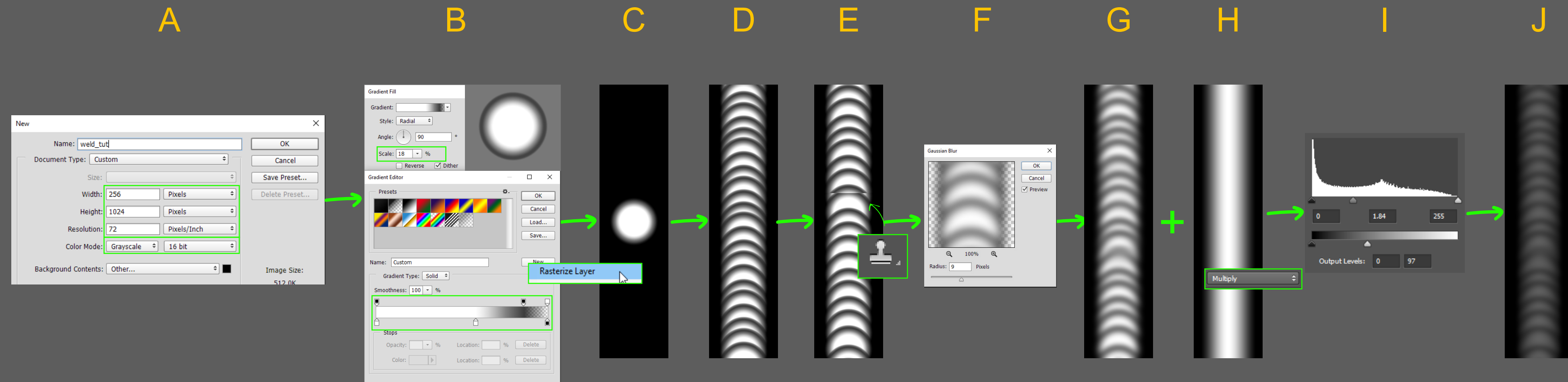
WELD



TO ACHIEVE A BETTER WELD I'D SUGGEST TO SMOOTH A BIT THE CORNER YOU ARE GOING TO SCULPT BEFORE STARTING

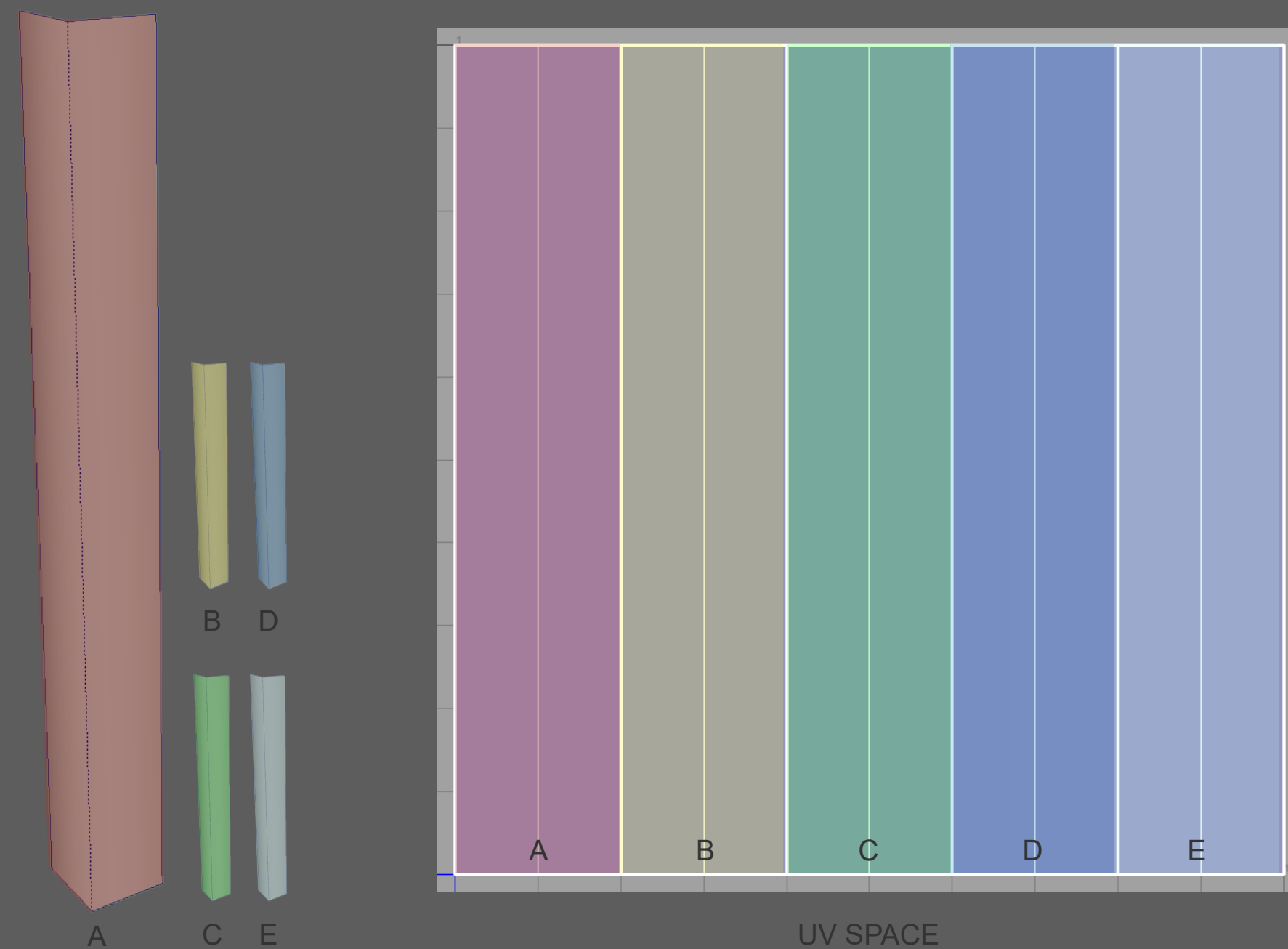
HOW TO CREATE WELD ALPHA IN PHOTOSHOP

This technique can be applied to other software such as Substance Designer



Create a new document (A), Using the gradient layer you can create a spherical shape with a smooth black outline, might be better to set the background mid-grey to see the dark ring around the circle (B), Remove the background and rasterise the layer (C). Duplicate the layers vertically to fill the whole image like in the image D and merge all the layer then press Ctrl+A and from the top menu press **Image > Crop**. At this point go to **Filter > Other > Offset** and set horizontal at 0 and vertical offset half of the size (512 px). If you get a seam you can easily remove it with the Clone Stamp Tool you can remove it (E). Now go to **Filter > Blur > Gaussian Blur** and try to get the result I got (G). Now create a new gradient layer (H) and set the blending mode to Multiply on top of everything. The last step is to reduce the intensity of the map with some level (I). You should get something like mine (J). Depends on the type of weld you want to deform more the shapes as well as adding more variation on it.

LOW POLYS AND BAKE

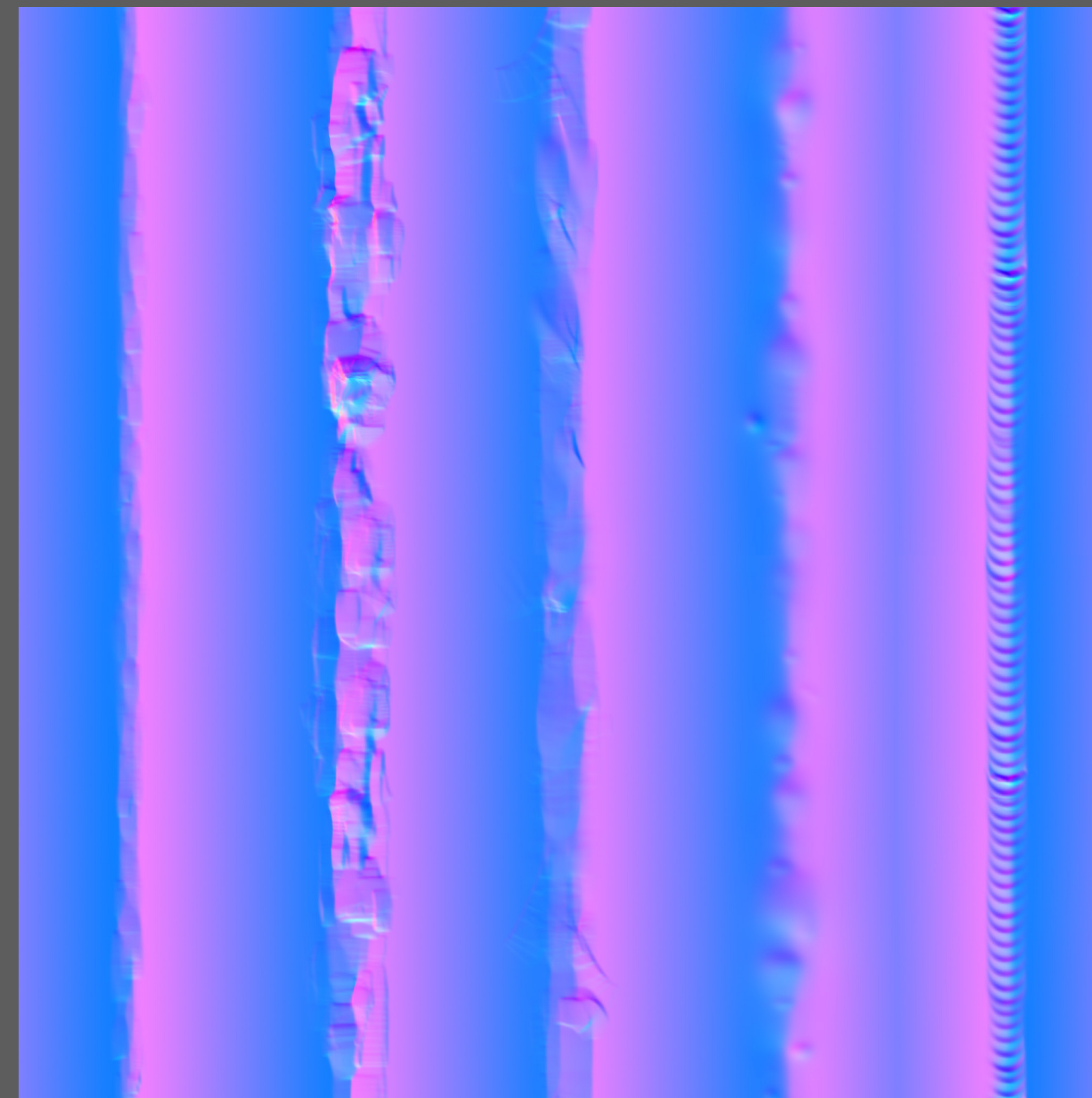
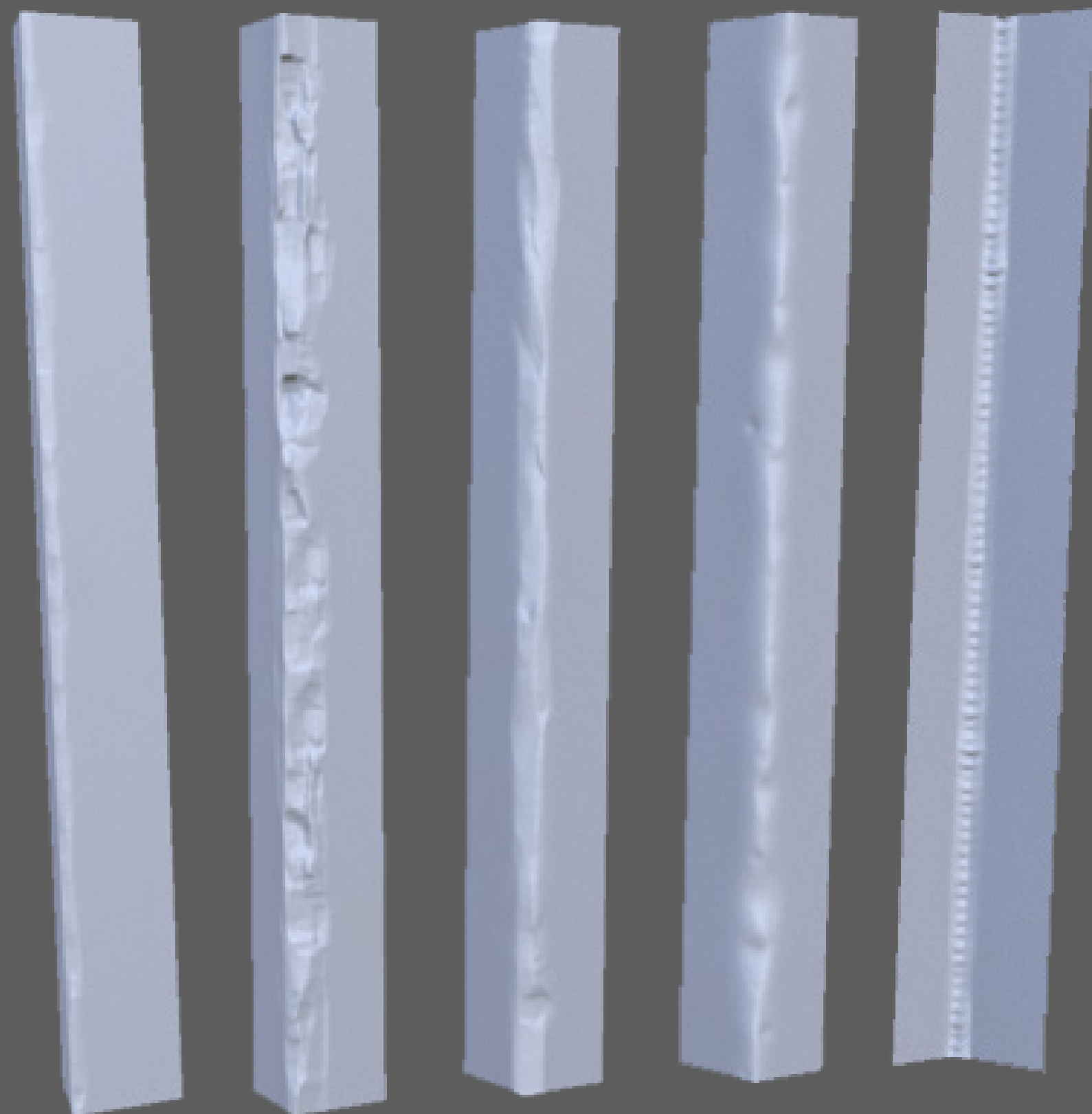


- Needs to have a single uv shell/island

- Needs to have the same smoothing groups (soften edge) Having the same smoothing groups will also allow us to use the edge normal decal on every angle we want.

- Each uvs need to be unfolded covering the entire slot

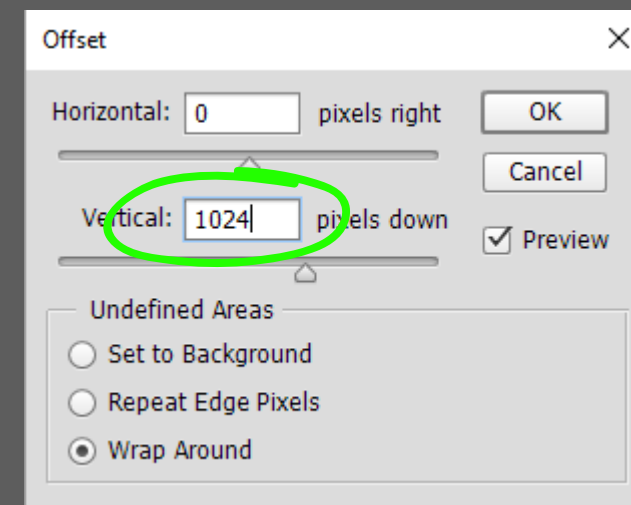
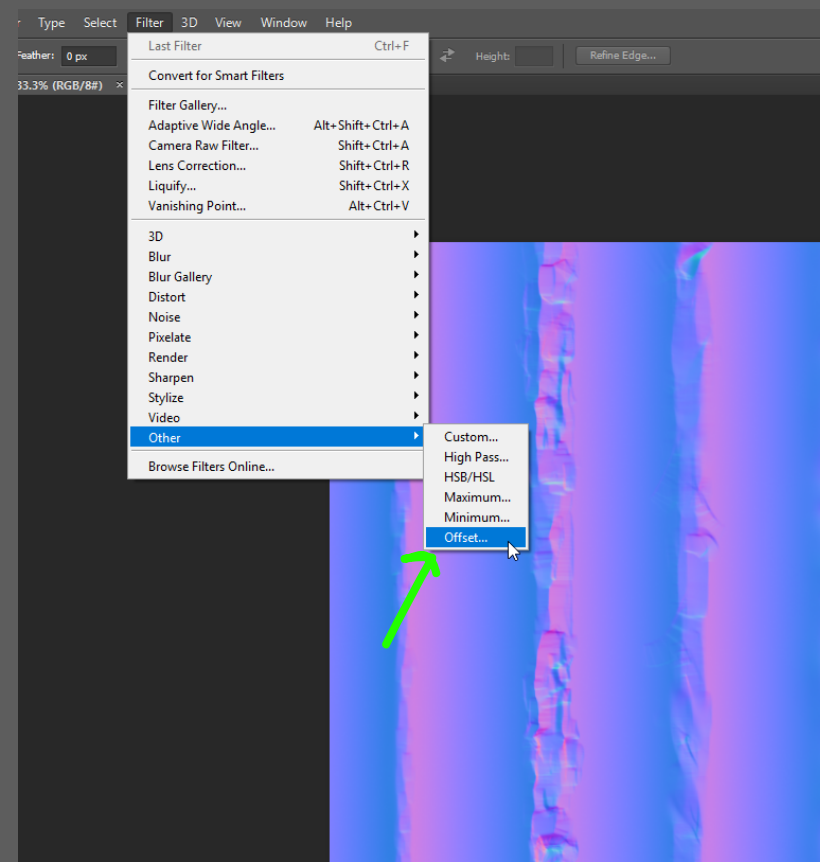
-Try to place uvs snapping with the grid. it's going to be much easier to work with later on. Snapping things on grid, in general, it's a good practice to keep everything nice and tidy. Every software should allow the snap to grid functionality both if you are working in 3d or 2d with uvs.



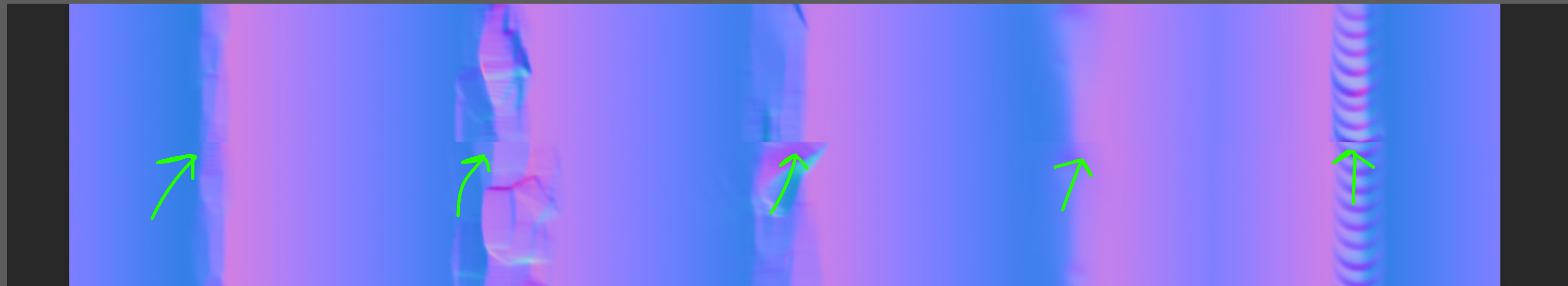
Finally bake the high poly on the low poly (if you don't know how to do this I made a tutorial explaining how to **bake normal maps**, go check it out.) You should get something similar to image above. Unfortunately you will have to remove some seams on the vertical axis (which is a problem because later we are going to use tile this texture)



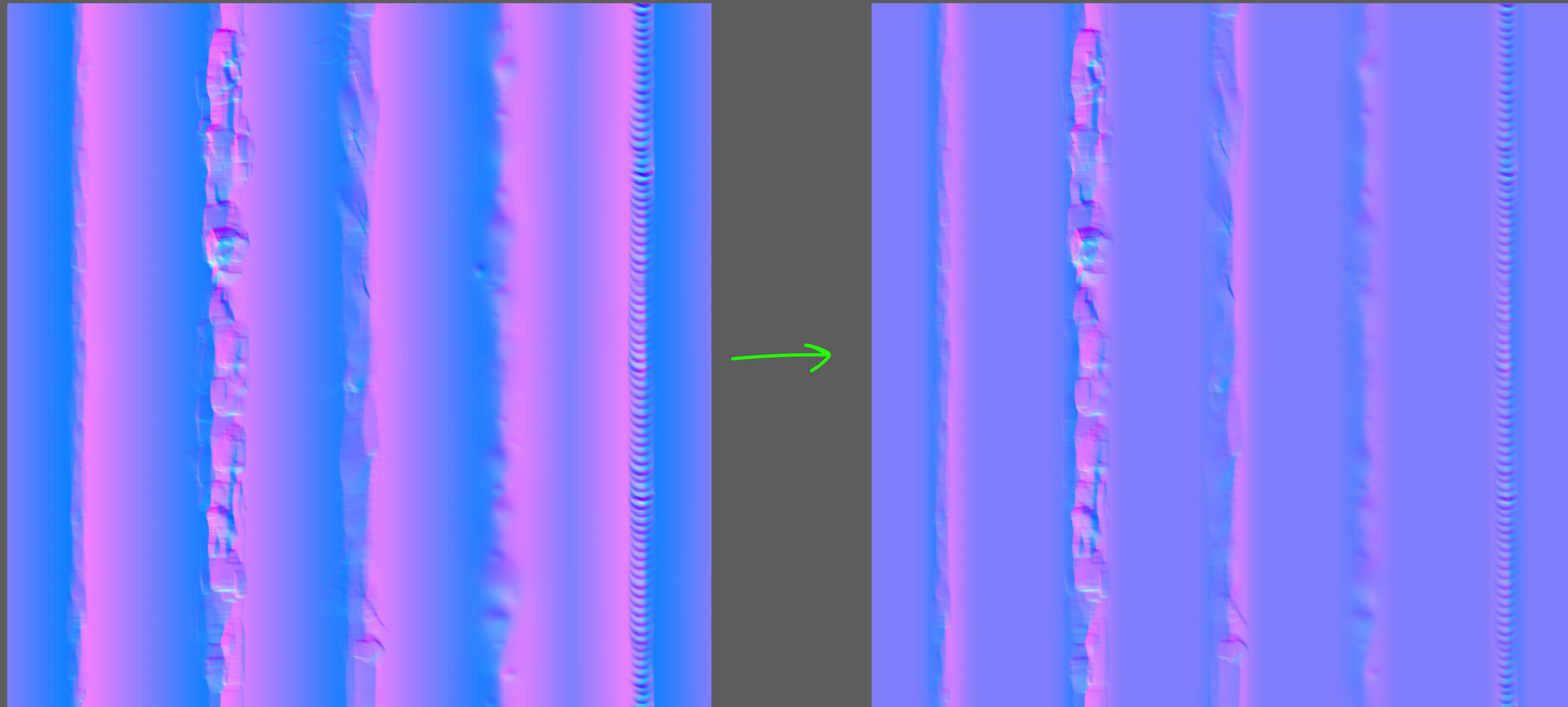
Time to fix the seams. You can quickly fix that with photoshop or similar software.



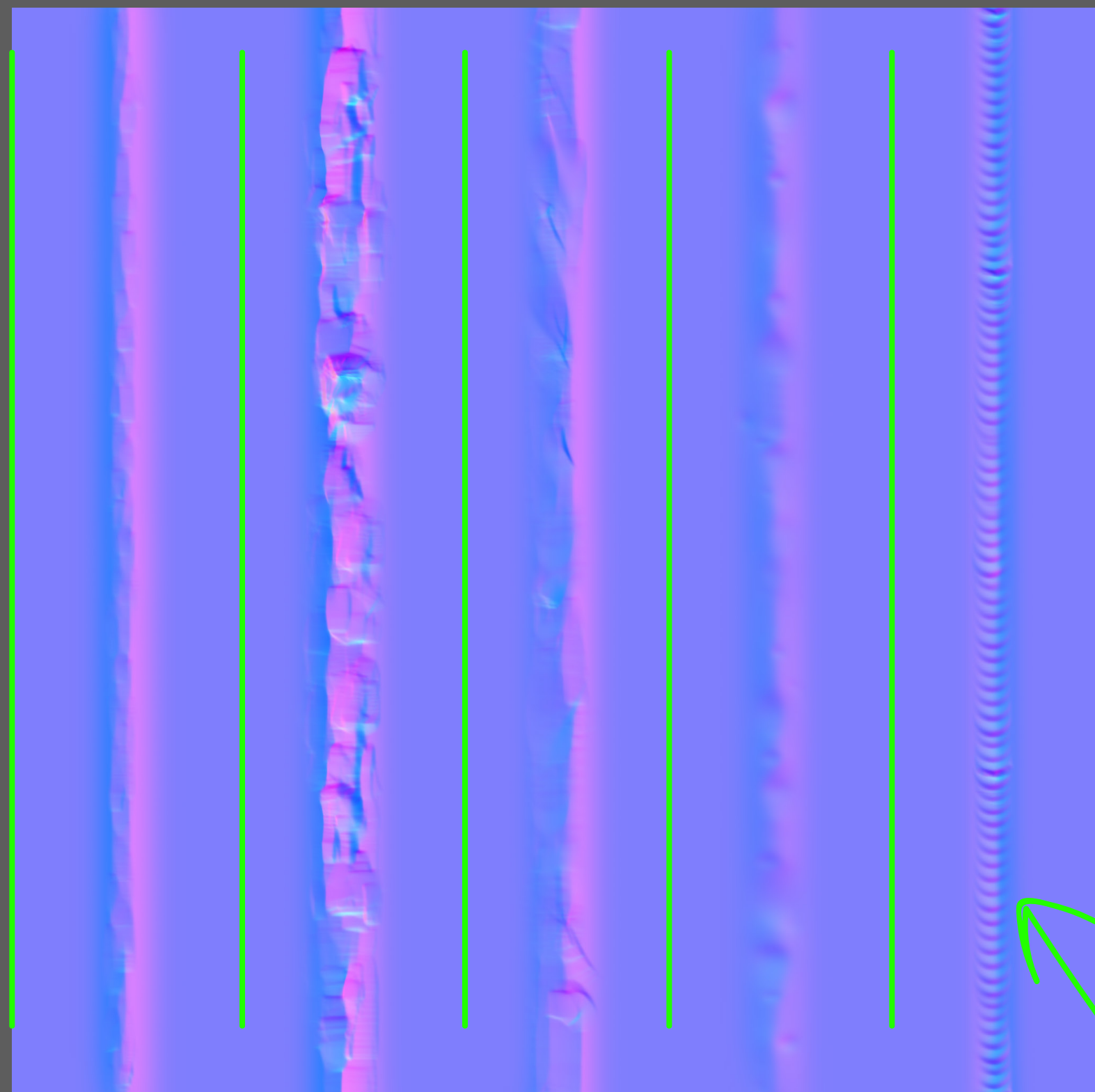
SET THE VERTICAL
VALUE OF HALF OF
YOUR TEXTURE HEIGHT



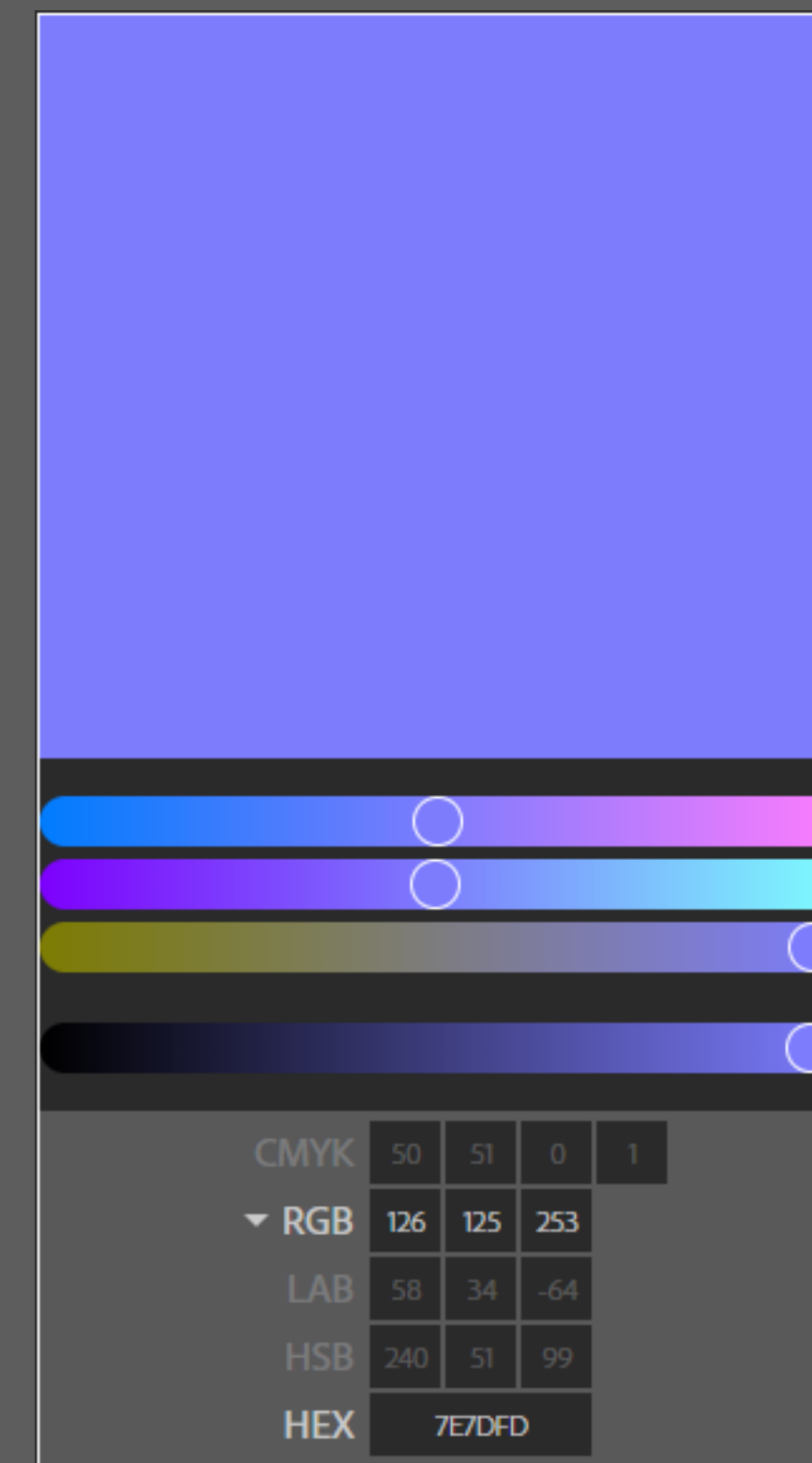
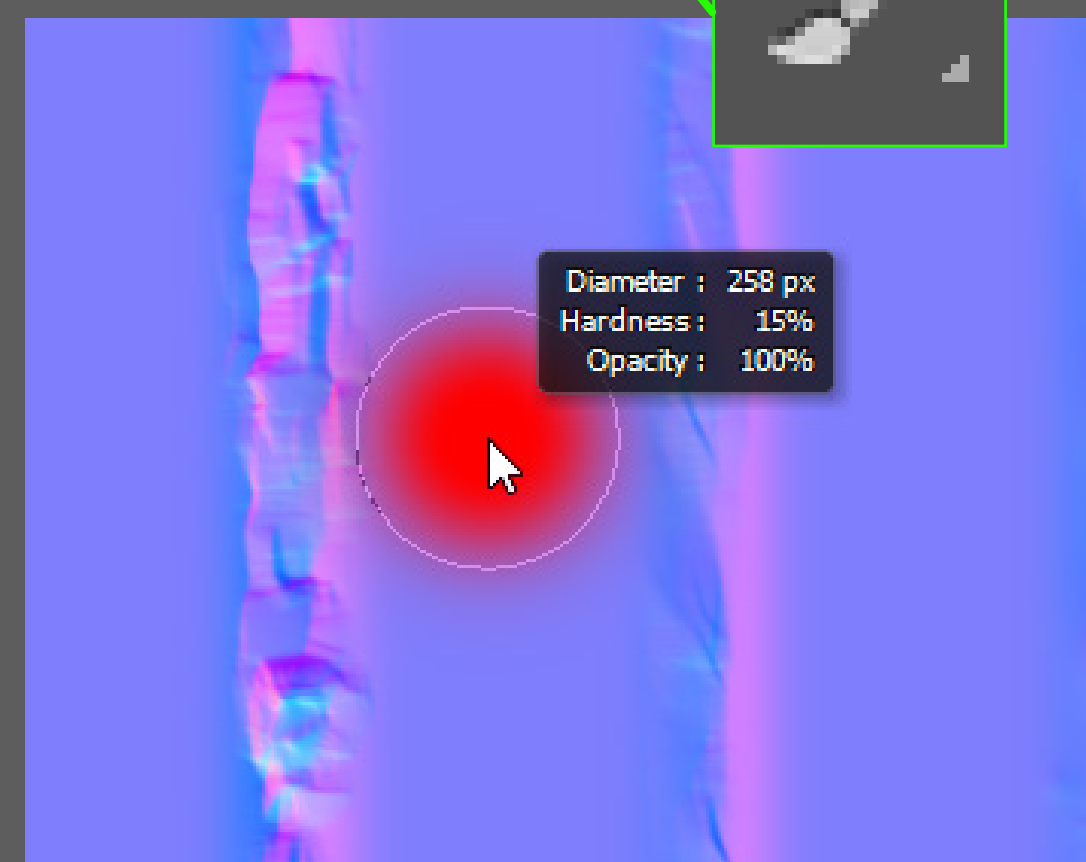
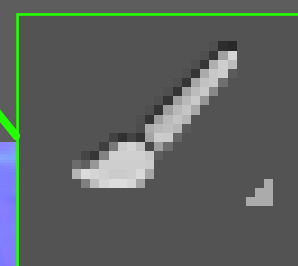
The Clone Stamp Tool will do the job



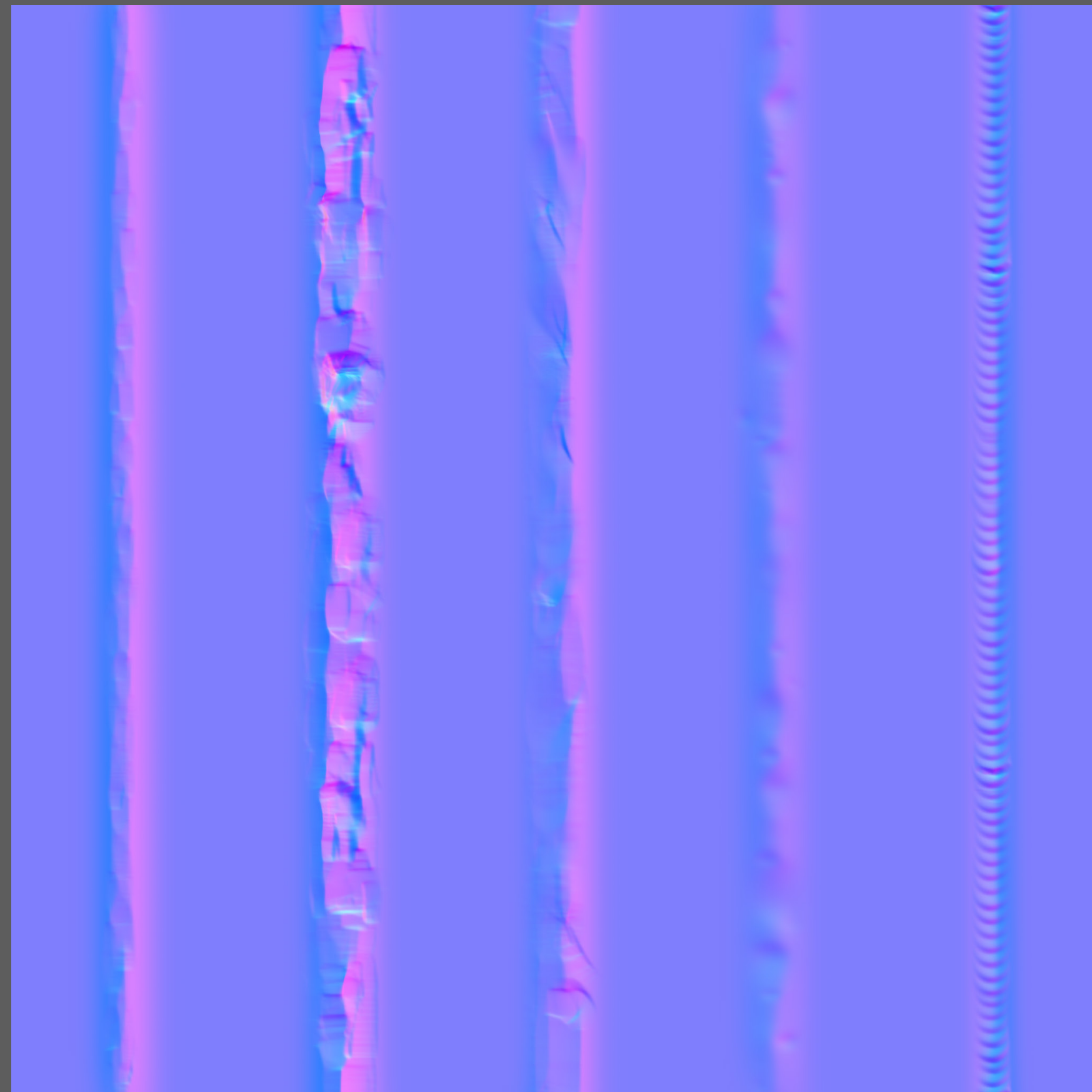
Now we need to adjust the texture a bit. Since we baked from a 3d model on a smooth mesh, we need to flatten a bit the area between the different corner dents so that when we applied the normal decal, the transition will be more seamless



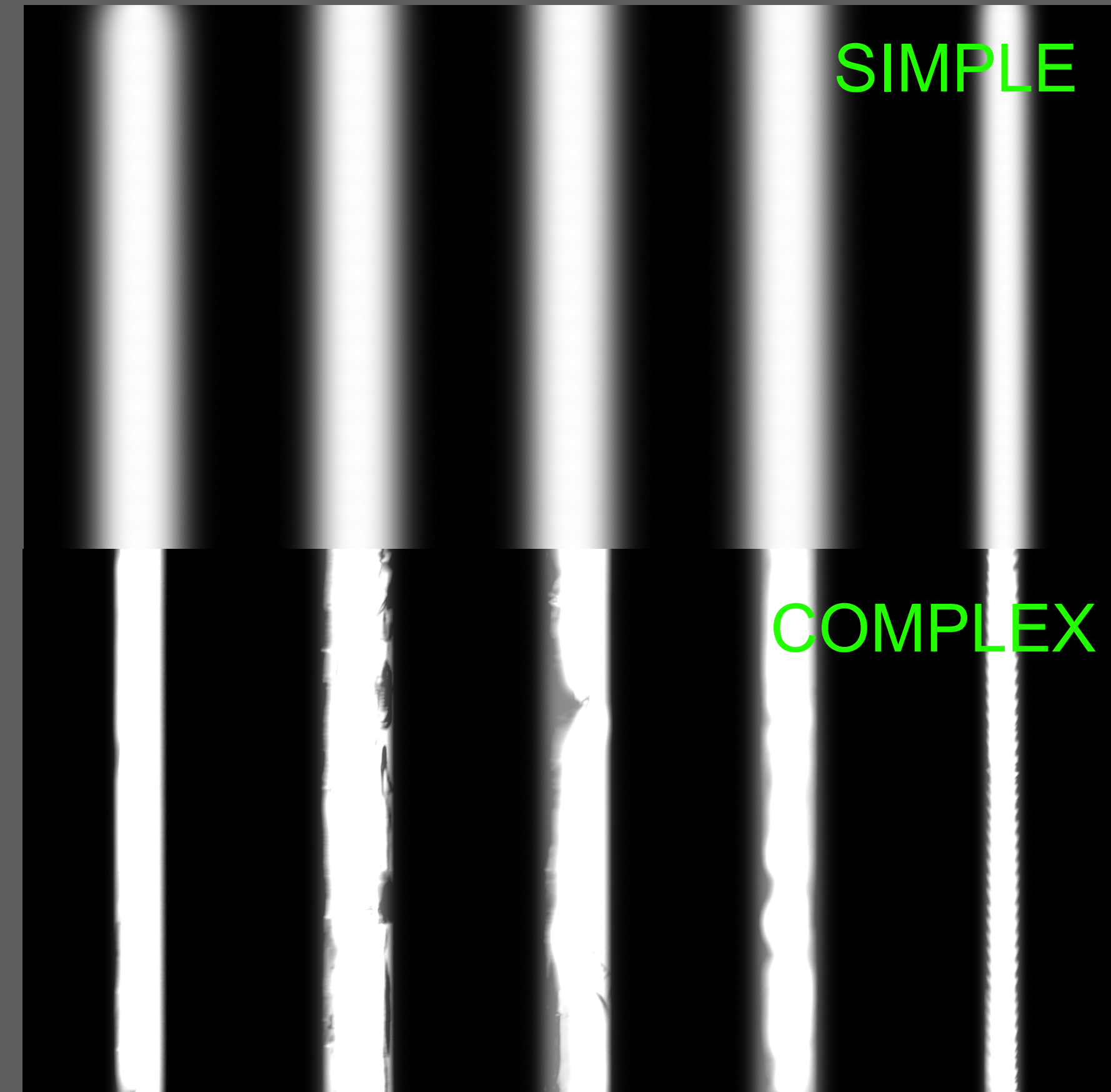
To get this result open it with photoshop (or similar) and create a new layer. Then with a soft brush make some strokes in the gaps. Use the colour for the flat normal map.



HEX: 7E7DFD



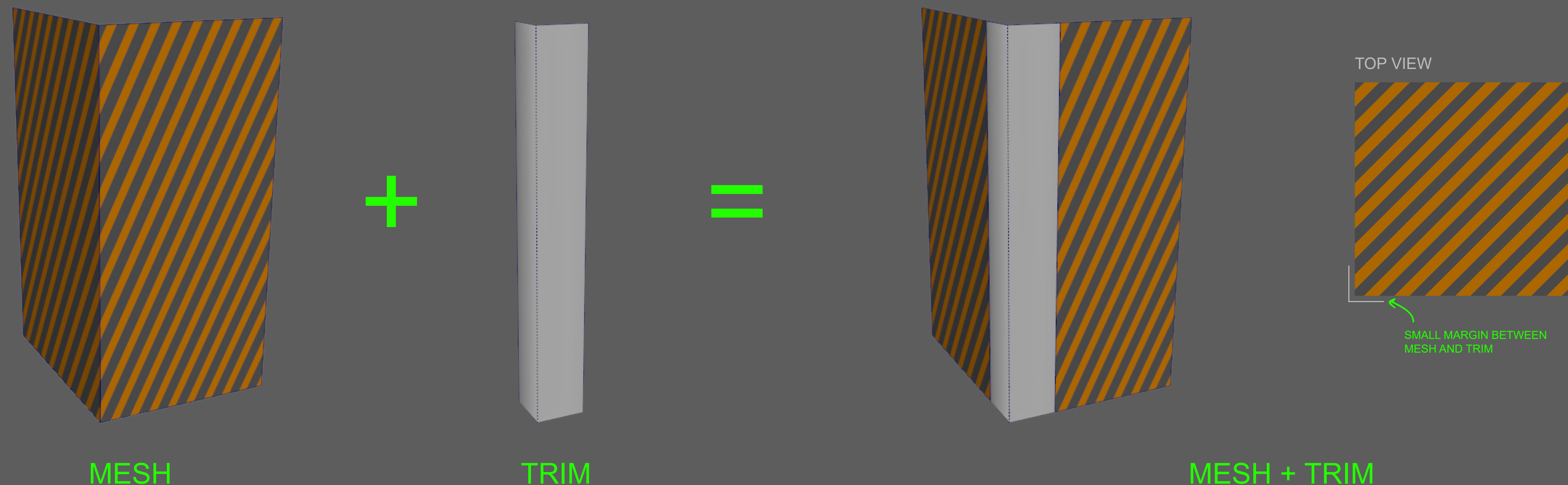
+



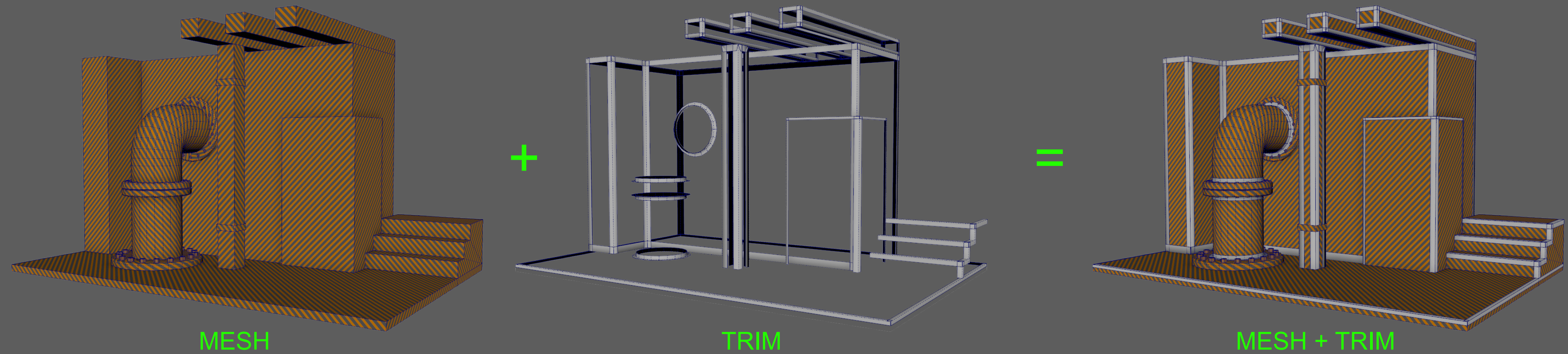
Last thing we need to create an alpha texture to mask out the part of the normal we don't need it. You can also create a more complex alpha if you see some problems in the way the normal decal is blending with the other normals. Both should work nicely, the complex one is more accurate but at the same time if not correctly done could bring some other problems. My advice is to start with the simple one, and if the edge decal normal map is not blending nicely, You can try to make a more precise mask. You create them with the brush tool in photoshop, and the channels of the normal map can help you to extrapolate some shapes

HOW TO ADD NORMAL EDGE DECAL ON YOUR MODEL

What you need to create is just a corner mesh which I'm going to call trim mesh from now on. The faces need to be **slightly offset** from the original mesh in order to avoid visual issues. The width of the faces can vary based on the scale of the edge normal decal wanted. **The uvs** of the corner mesh need to be **mapped correctly** on top of one of the slots you have. The mesh needs to have **one smoothing** group (in other words the edge needs to be a soften edge)

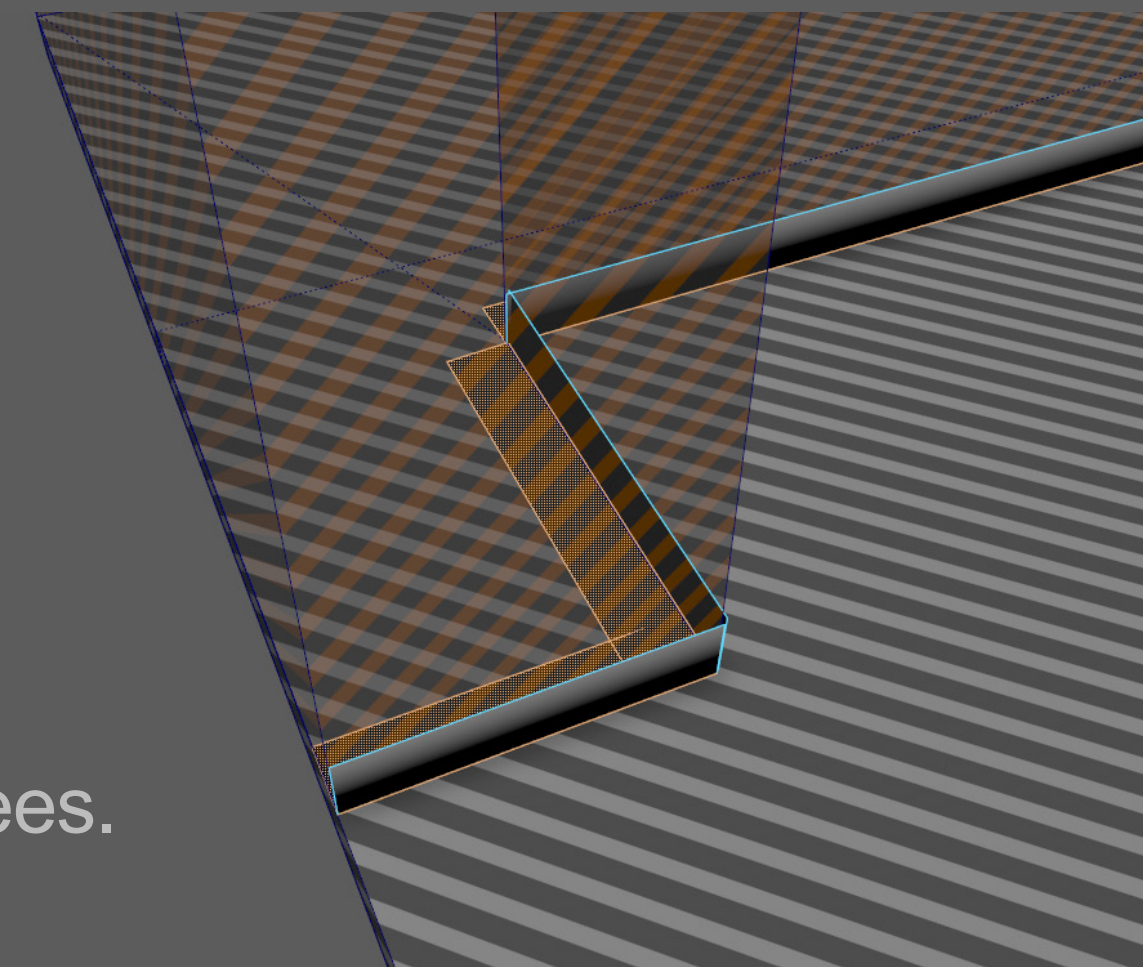


Let's have a look at something more complex.

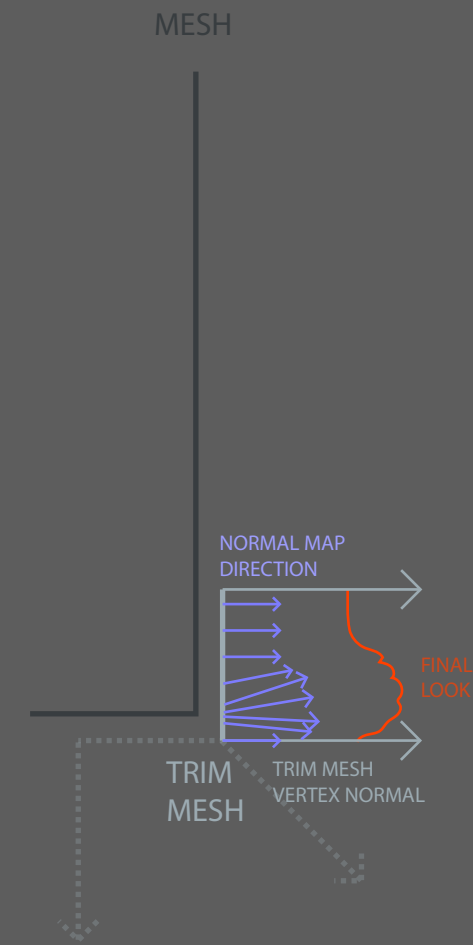
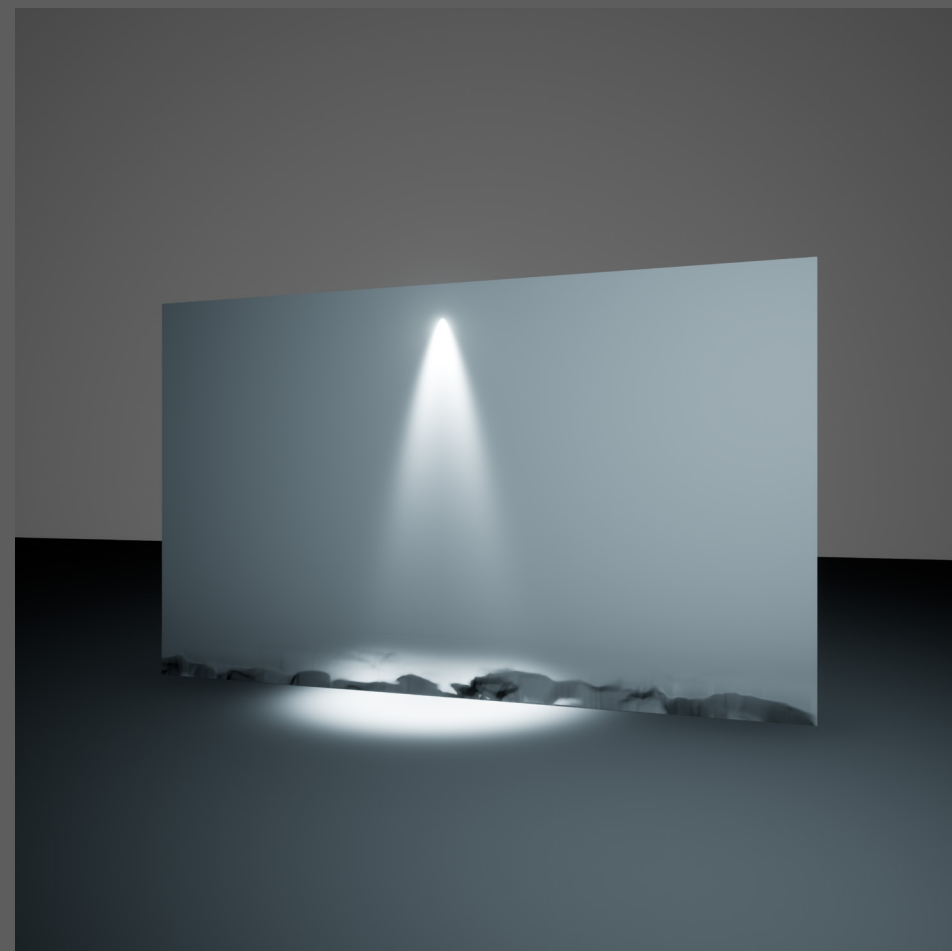


EDGE DECAL ON OBJECT BASES

There are some cases where you want to have a normal edge decal on the base of an object to get a visually smoother transition between the vertical and horizontal surfaces. In my specific example (image on the right) I want some details on the “concrete slabs”, but I want to keep the floor untouched. It is important that you keep those hidden faces (the selected ones). Because they allow the edge of the corner to be soft which is required to visualise the normal information correctly. Alternatively, you can rotate the vertex normal by 45 degrees. See next page for a more in-depth explanation, and some other alternatives.



“ISSUE”

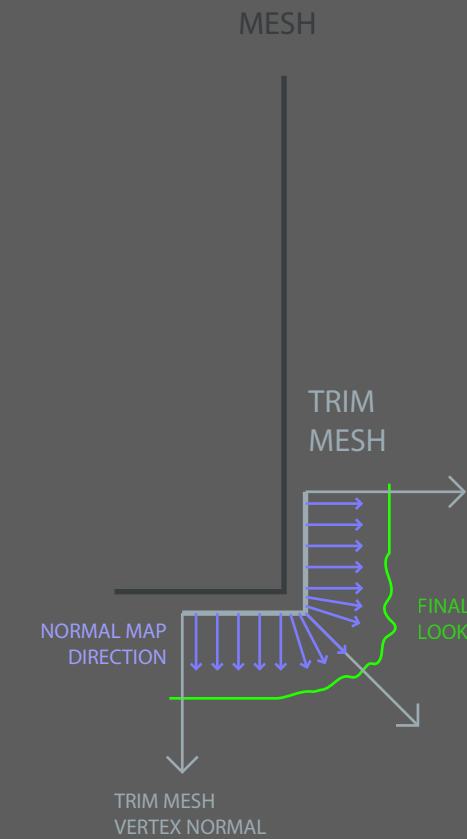
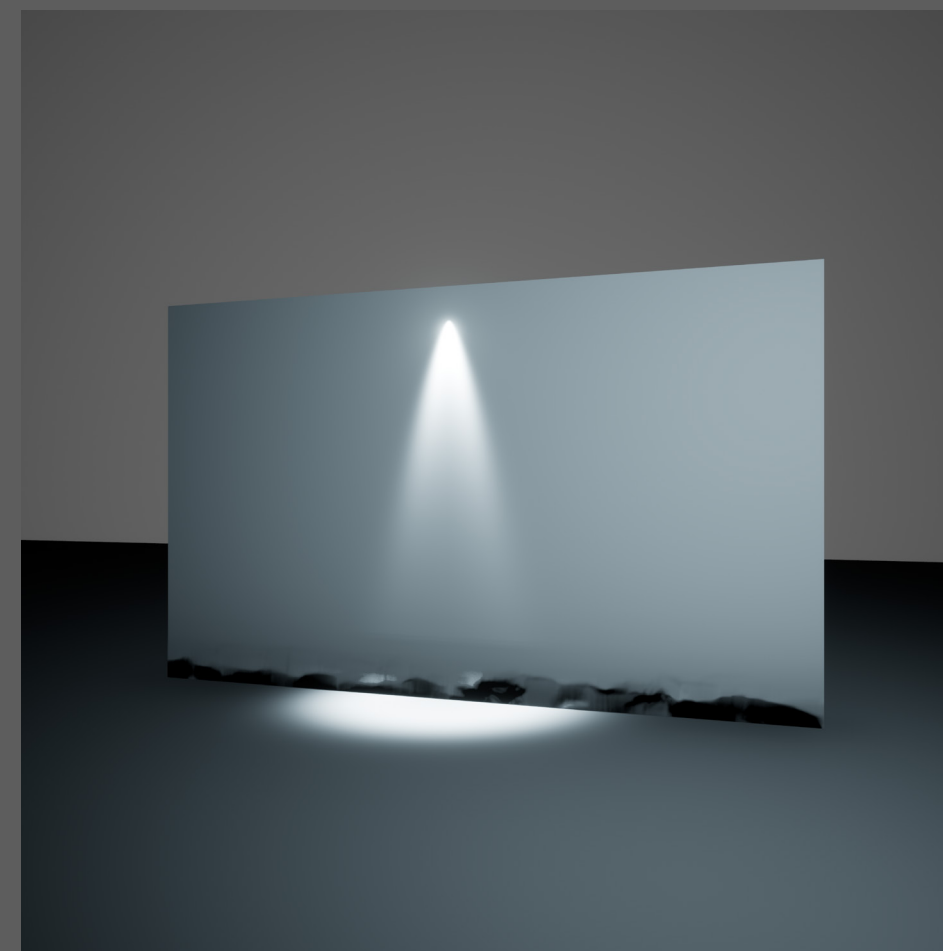


As you can see by deleting the hidden face of the trim at the base, you are (by default) straightening the vertex normal, which means that the normal map (that has been created along the average normal of the corner of those two faces) will perform incorrectly, and it will lean out trying to bend around a missing old average vertex normal.

You can purposely remove the face if you are trying to achieve something like the example above :)

SOLUTIONS

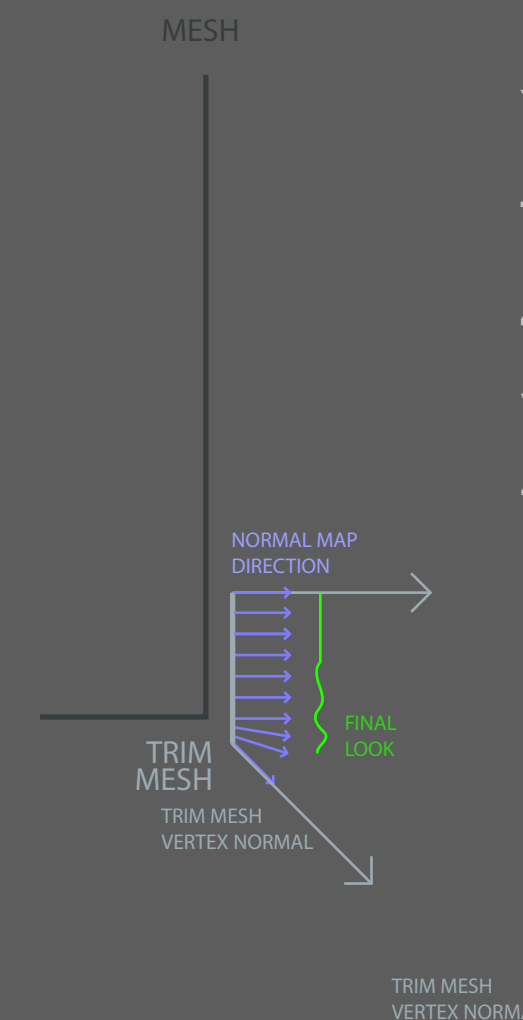
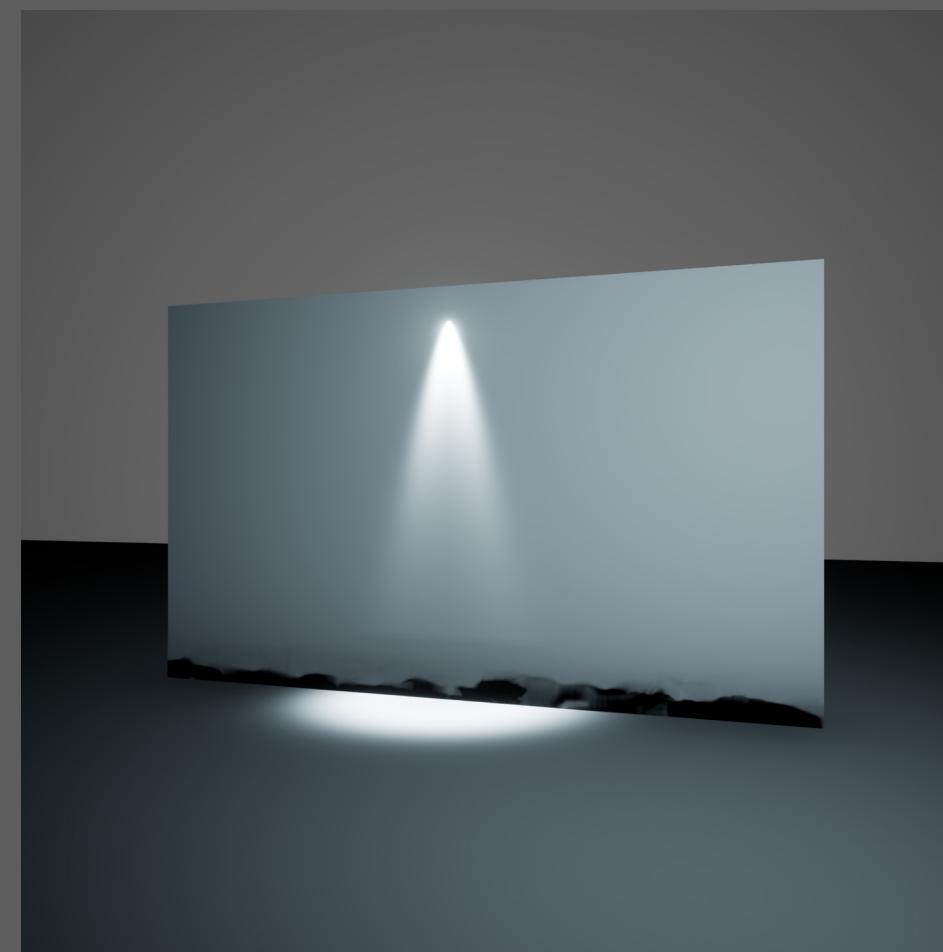
A



Even if you don't see it, you keep the hidden face.

PRO: Quick
CONS Extra faces

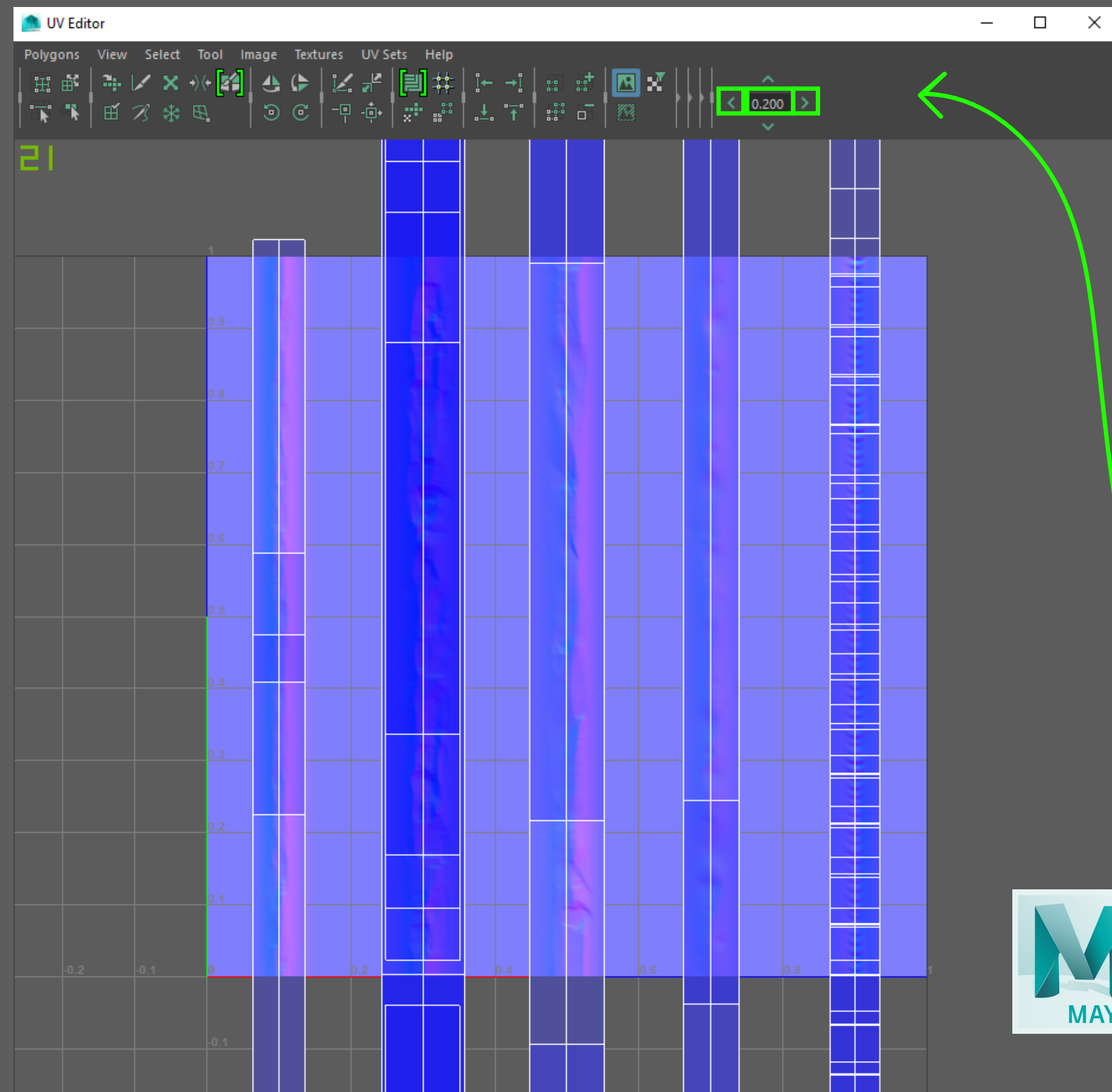
B



You delete the hidden face, and you rotate by 45 degrees the lowest vertex normal towards the bottom.

PRO: Single face
CON: Slow

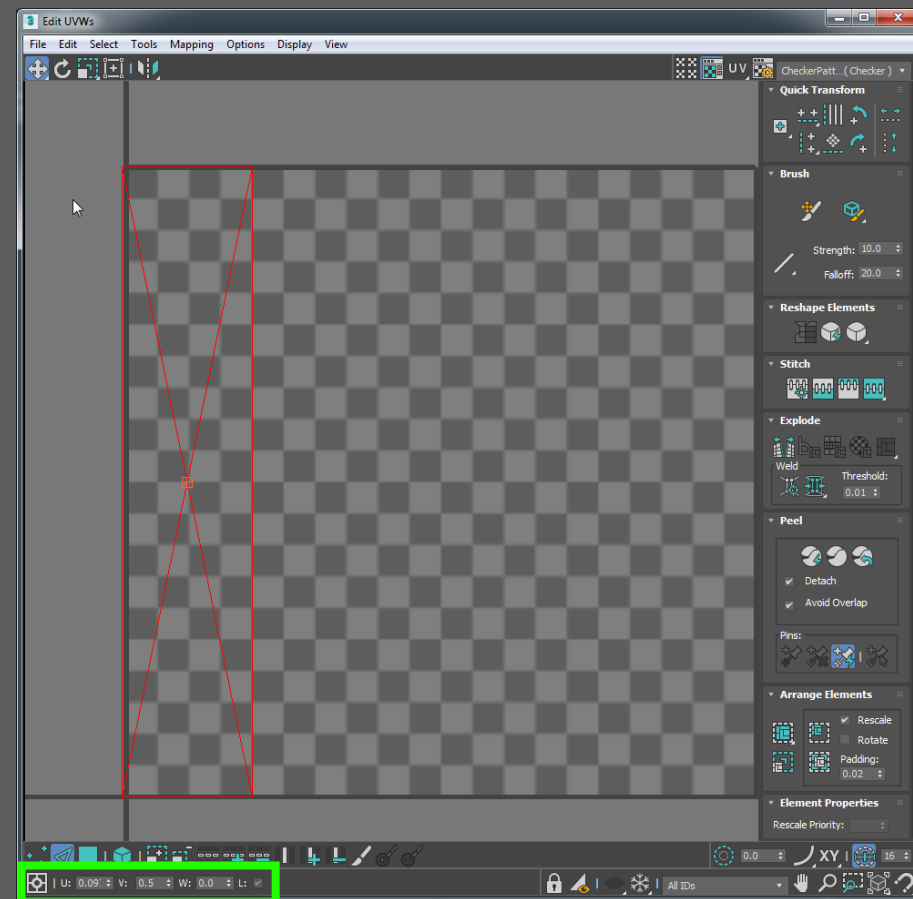
UVs wise for the trims you are going to get something like the following image. You want to offset some UV islands in order to map the trim on a different part of the texture so that you can achieve more variation.



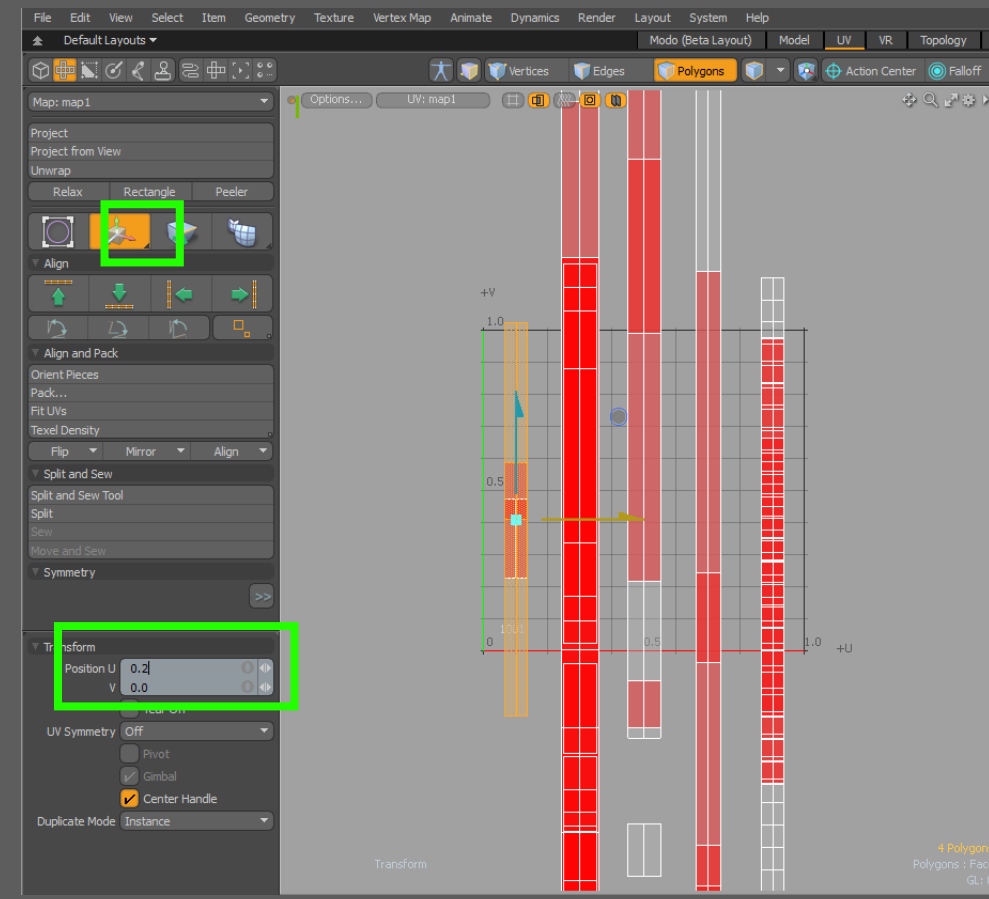
As you can see the grid is really helping us snapping the islands in the right slot and making sure that the decals are mapped correctly on the edge.

Also since we snapped on the grid in the UV editor (with Maya), we can set the offset tool to 0.2 (in my case), and every time I'll press the left and right arrows I will offset the selected UV to the previous or next slot allowing you to quickly trying them out

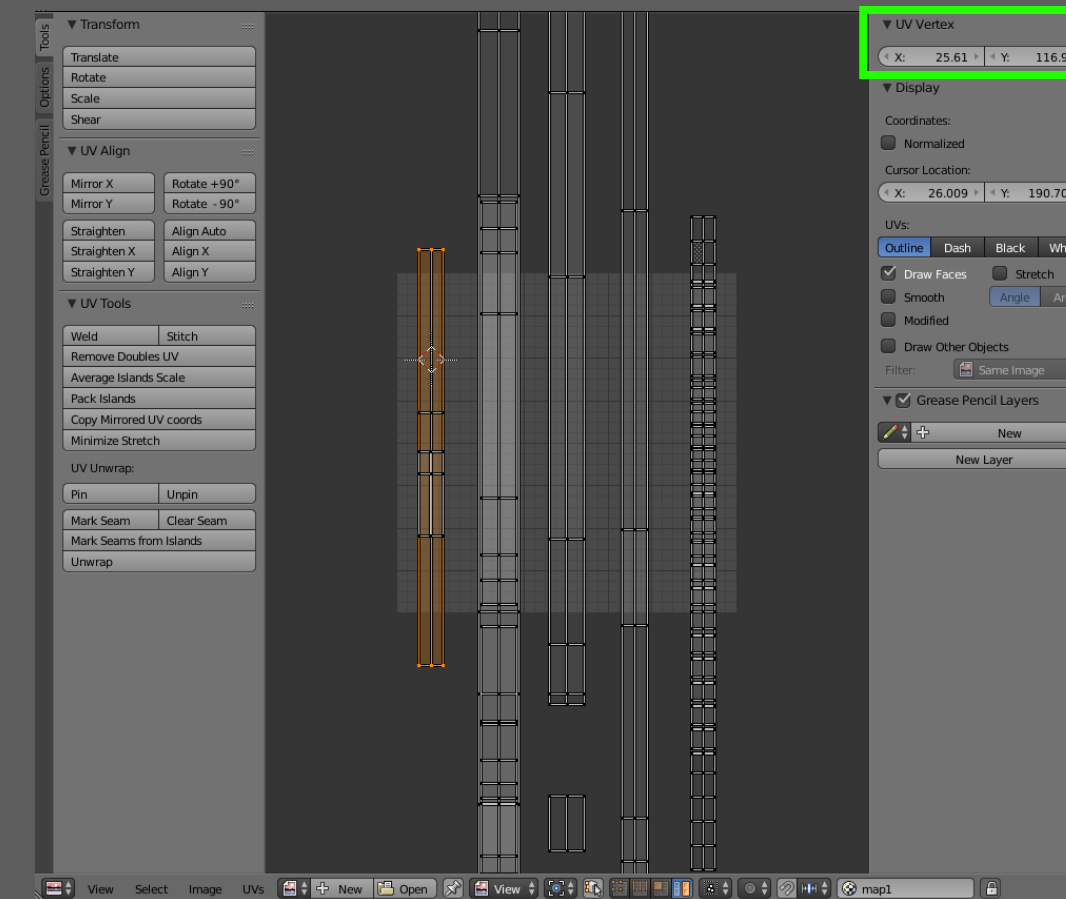
Not only Maya has the feature to offset uvs, here the spae option on other softwares



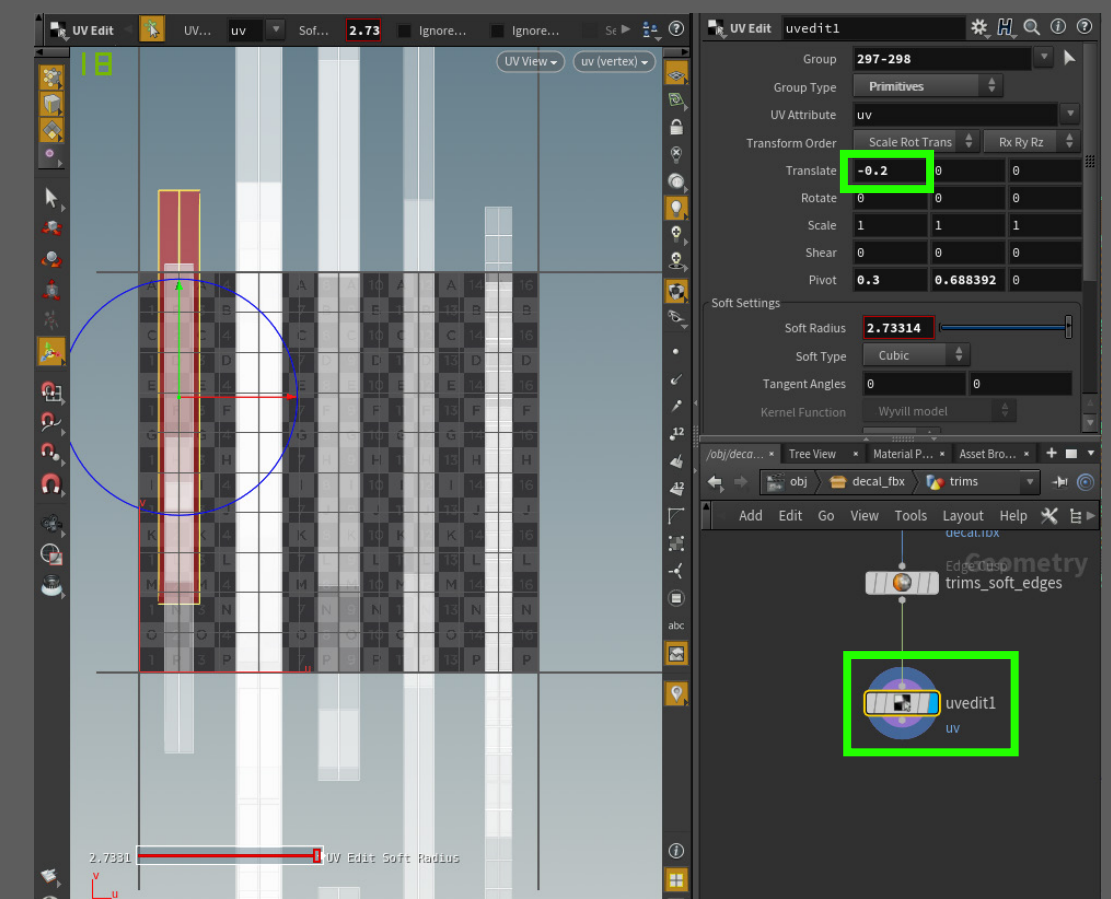
ABSOLUTE
U: 0.09 V: 0.5 W: 0.0 L:
RELATIVE
U: 0.0 V: 0.0 W: 0.0 L:



In Modo in the uv editor, you select the trim uv you want to move, select Transform tool and in the Transform properties you have the U and V value. You can offset by adding or subtracting 0.2 (in my case)



In Blender after open the uv editor press "N" to get the right bar with the properties. The values are absolute so you need to add or subtract the offset you want. In my case 50 units

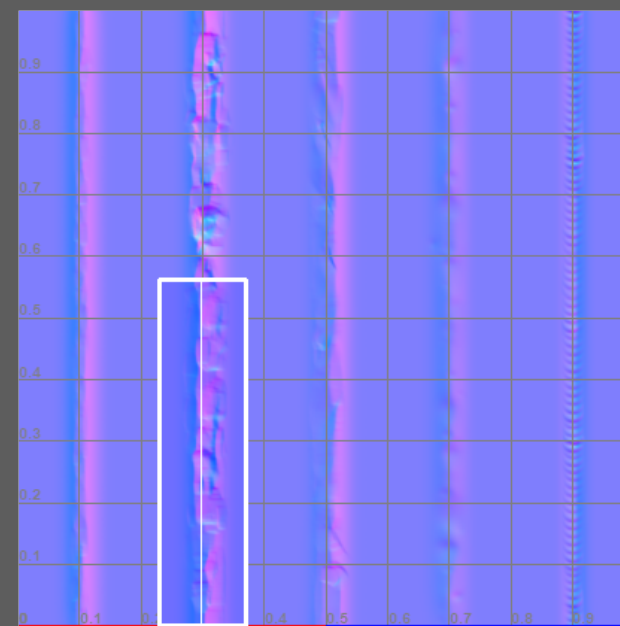


In Houdini you create an uvedit (or when you are making the uv), you open the uv editor (space + 5) and select the uv then add or subtract 0.2 (in my case) in the transform properties

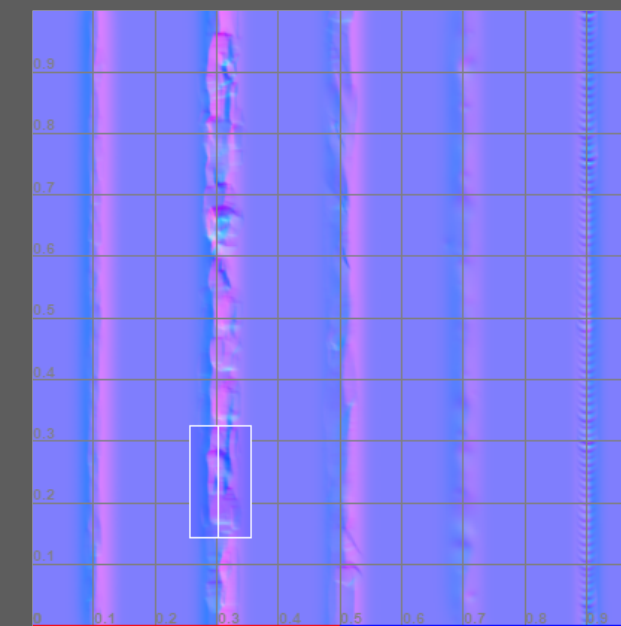
SOME WAYS TO CHANGE THE SCALE OF THE DECAL DETAILS

The first thing first, we should take into consideration one important thing: We should **ALWAYS REDUCE THE PART OF THE MESH THAT IS GOING TO BE TRANSPARENT** based on the Alpha we generated earlier. For two reasons: First because the amount of transparent area we are rendering: less transparent is cheaper. Second reason is that by reducing the unnecessary mesh area, we are reducing the size of the trim mesh, which will reduce the draw distance, that means the trim mesh is rendered only up-close. The following are some of the available options you have if you want to scale details, choose wisely :)

YOUR INITIAL DECAL



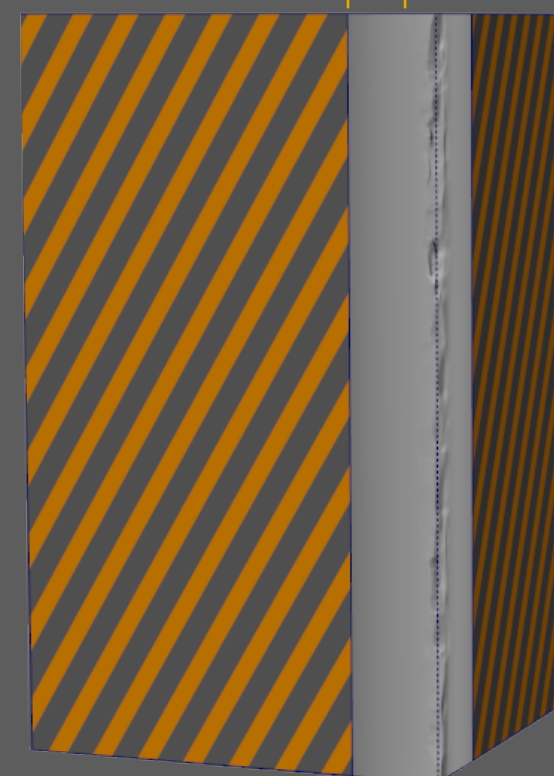
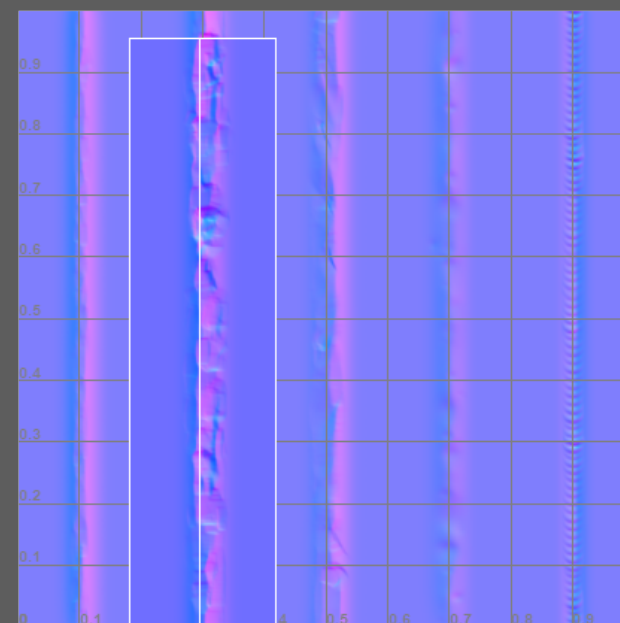
MESH SCALED UP



A

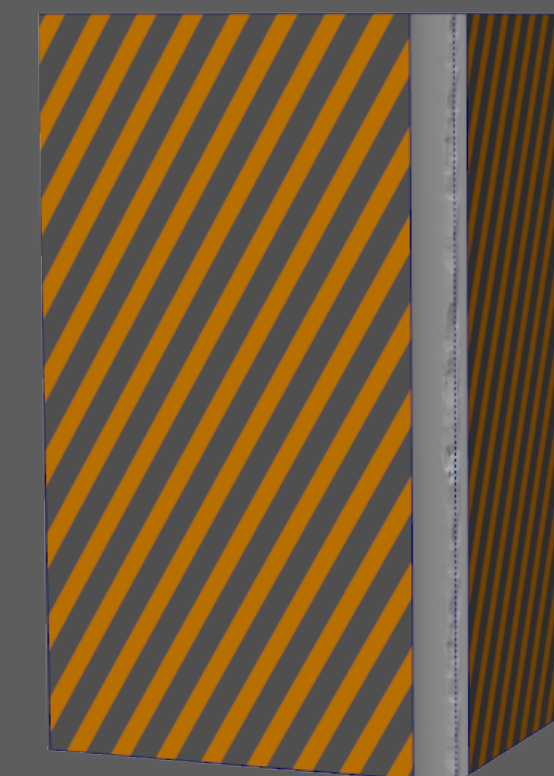
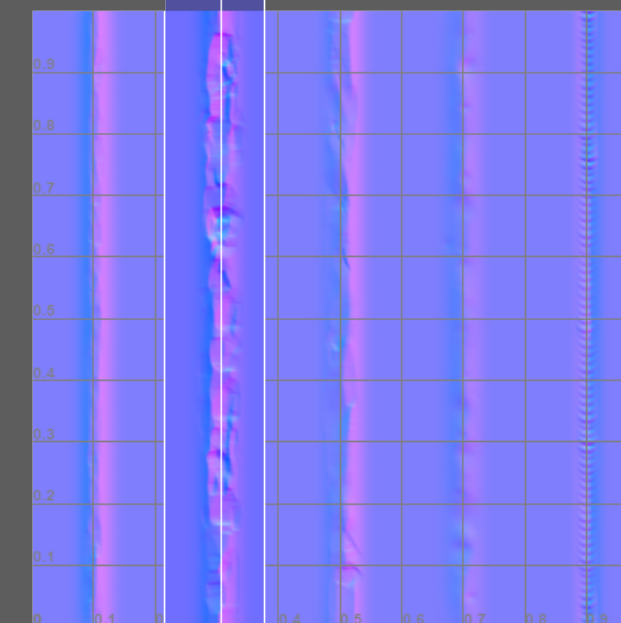
UV SCALED

however we would need to cut a bit of mesh in order to reduce the transparent area



B

MESH SCALED DOWN + UNFOLD



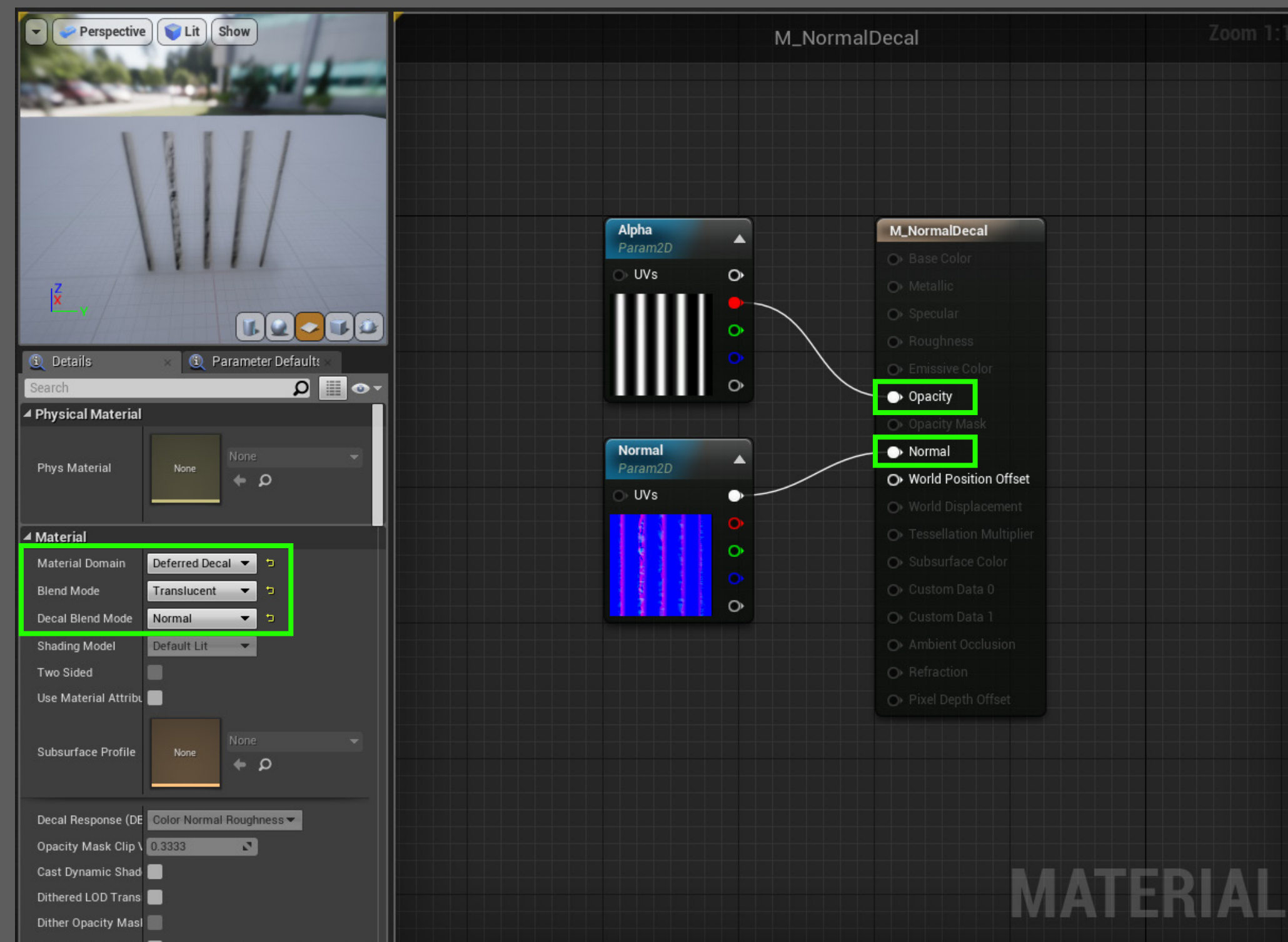
C



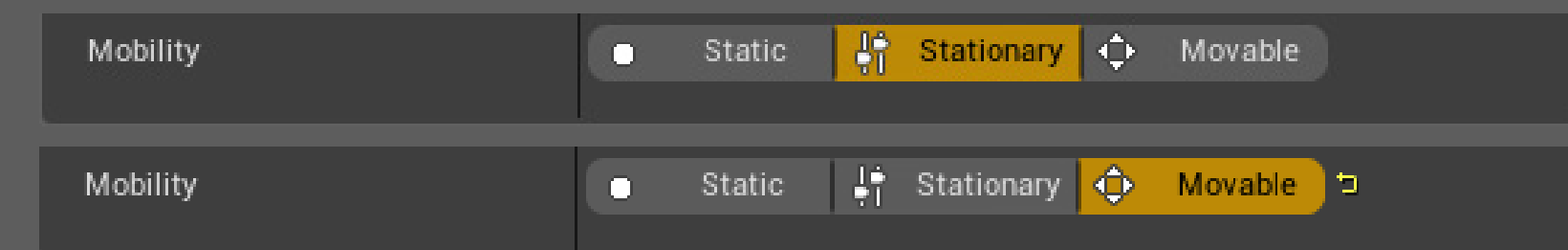
NORMAL DECAL IN UNREAL (SIMPLE)

Creating the material for the normal decal is incredibly easy in Unreal Engine. Just import the two texture you create (Normal map and alpha) in the material editor connect them as shown in the image and set the properties of the Material. Then apply the material to the trim mesh slot.

NOTE: YOU CAN LEARN MORE ABOUT DECAL BLEND MODE IN UNREAL ON THEIR DOCUMENTATION: [HTTPS://GOO.GL/ZVTEPW](https://goo.gl/ZVTEPW)



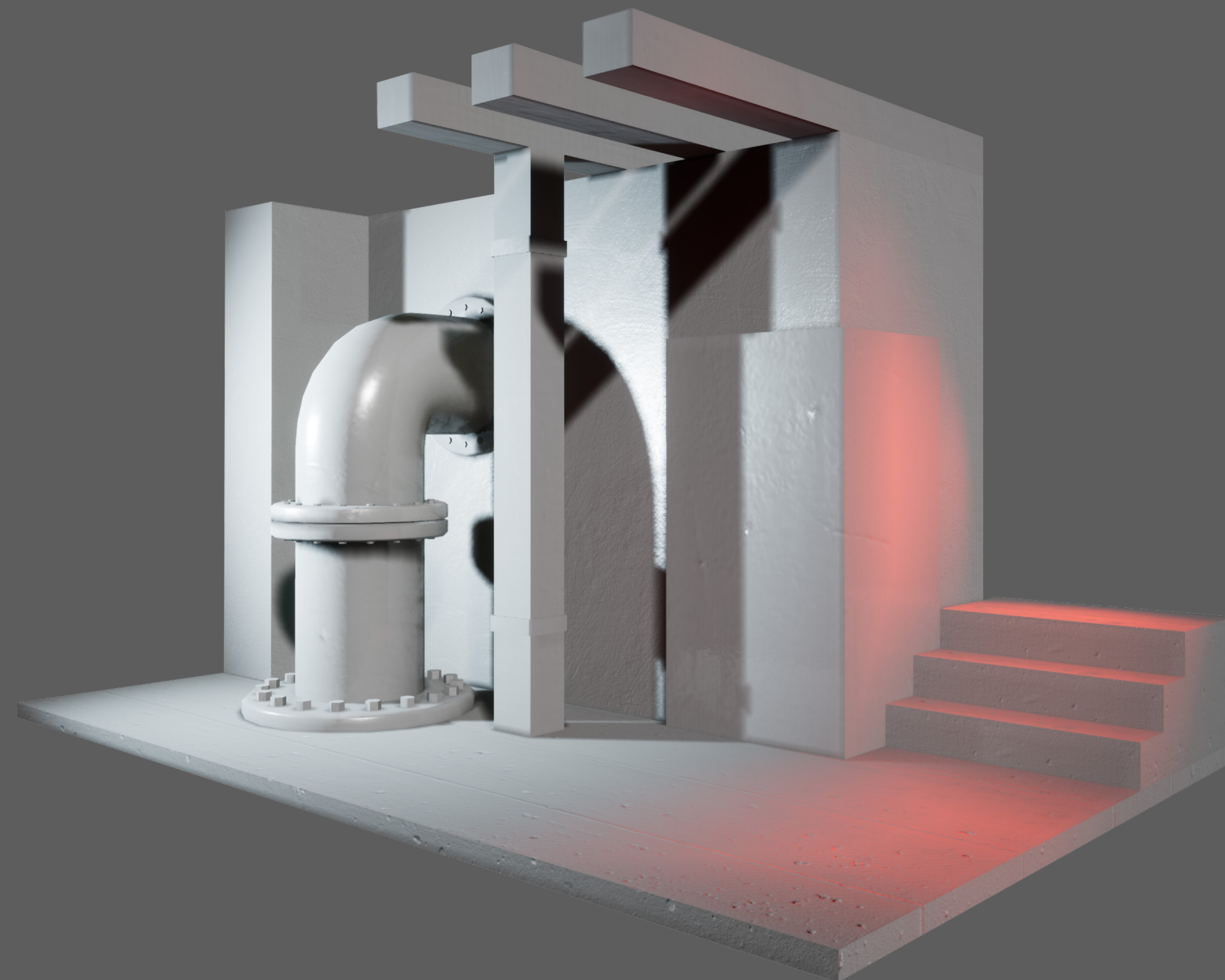
The normal edge decals only work with lights set as movable or stationary.



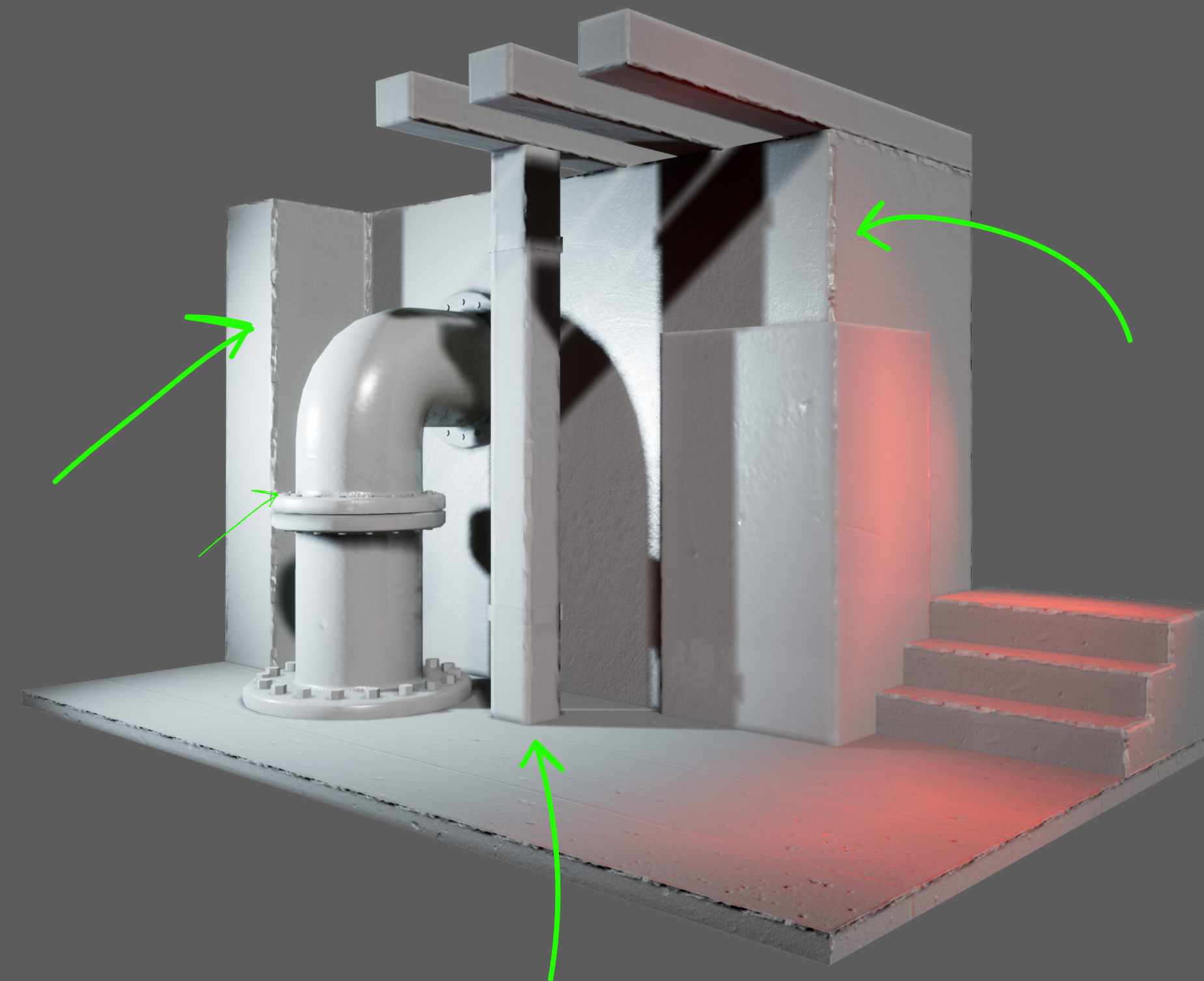
In very short, the reason why is not working with the static lights is that the normal maps are used for faking the lighting of bumps and destructions by making the rays bouncing in different directions from the actual geometry. If you bake the lighting (static) you are baking the light as a map onto the meshes which means that you are not going to have rays which means that the normal map can't work properly in the way I'm explaining with this tutorial

Here a comparason in unreal engine. I applied some tileable material as well to simulate more the resoult you will get. Normal edge decal sill work even if the surface is completly flat

MESH ONLY



MESH + NORMAL EDGE DECAL





NORMAL DECAL IN UNITY

```
1 Shader "Decal/NormalDecal"
2 {
3   Properties
4   {
5     _AlphaTex("Alpha", 2D) = "white" {}
6     _NormalTex("Normal", 2D) = "bump" {}
7   }
8   SubShader
9   {
10    Name "DEFERRED"
11    Tags {
12      "LightMode" = "Deferred"
13      "Queue" = "Geometry+10"
14    }
15    LOD 100
16
17    Pass
18    {
19      Offset -1, -1
20      zwrite off
21      CGPROGRAM
22      #pragma vertex vert
23      #pragma fragment frag
24
25
26      #include "UnityCG.cginc"
27
28      struct appdata
29      {
30        float4 vertex : POSITION;
31        float2 uv : TEXCOORD0;
32        float3 normal : NORMAL;
33        float4 tangent : TANGENT;
34      };
35
```

With this tutorial, you also get the shader you need to use to get the deferred normal decal functionality.

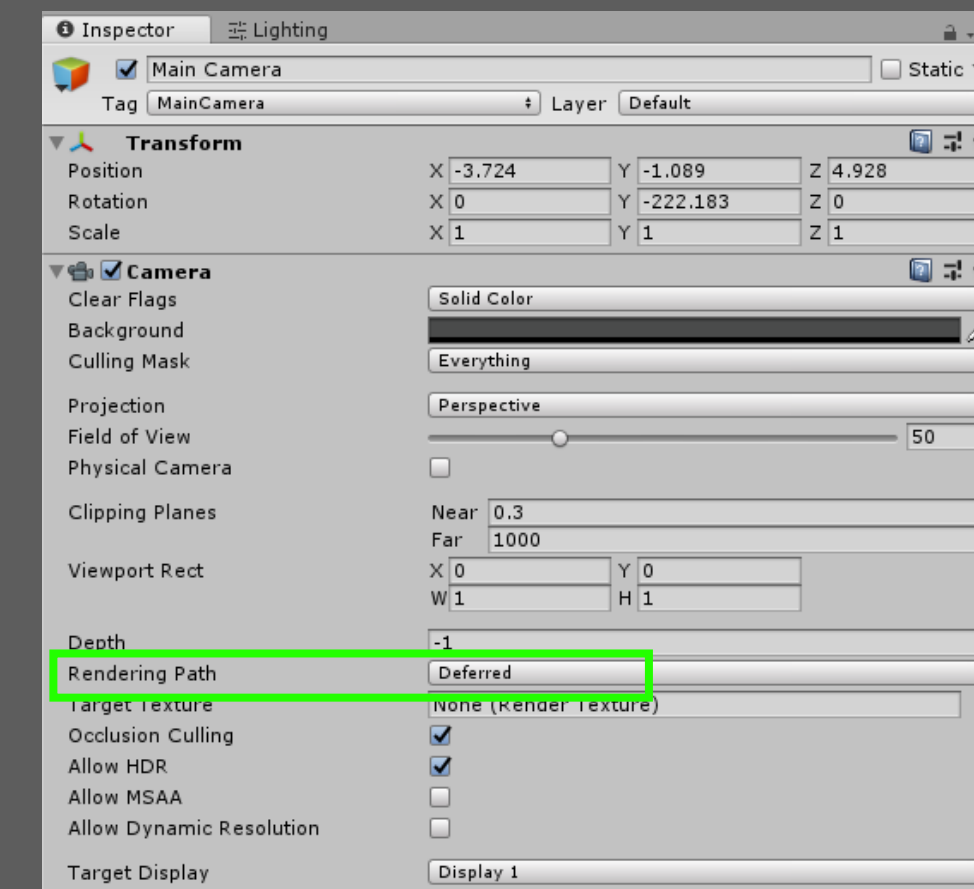
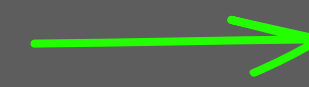
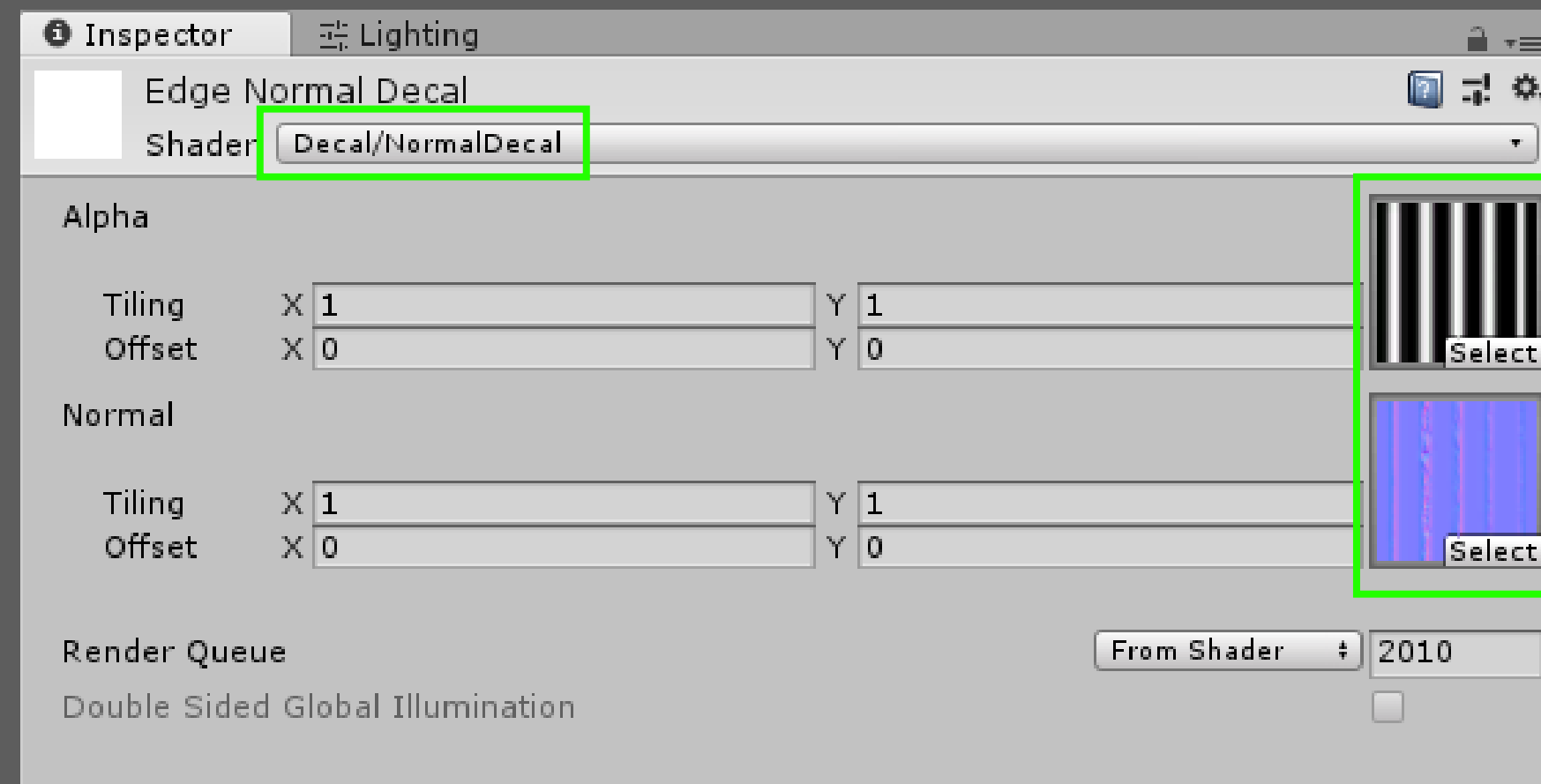
In your project, if nicely organised you should have a folder called "shader". Drag and drop the shader file in that folder. or in case you don't have the shader directory, cut it somewhere inside the project

* if you are reading this from another source go to leonano.com and you will find the link for download the original file.

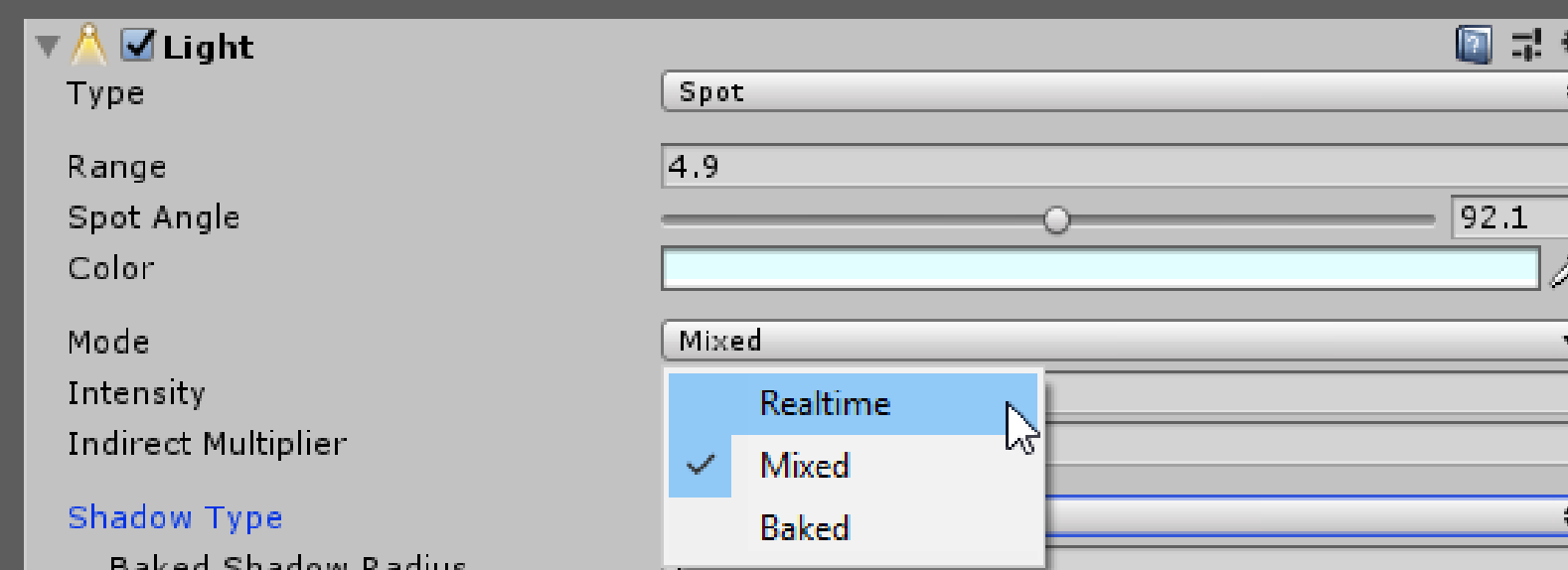
The Shader is made by: Dickie McCarthy ([@dickiemccarthy](https://twitter.com/dickiemccarthy))

Once you have the shader, Create a new Material and apply the new shader. Then assign the texture you create earlier (Normal map and alpha)

In order to see the normal decals, you will have to set your camera or project. **“Rendering Path” as Deferred**



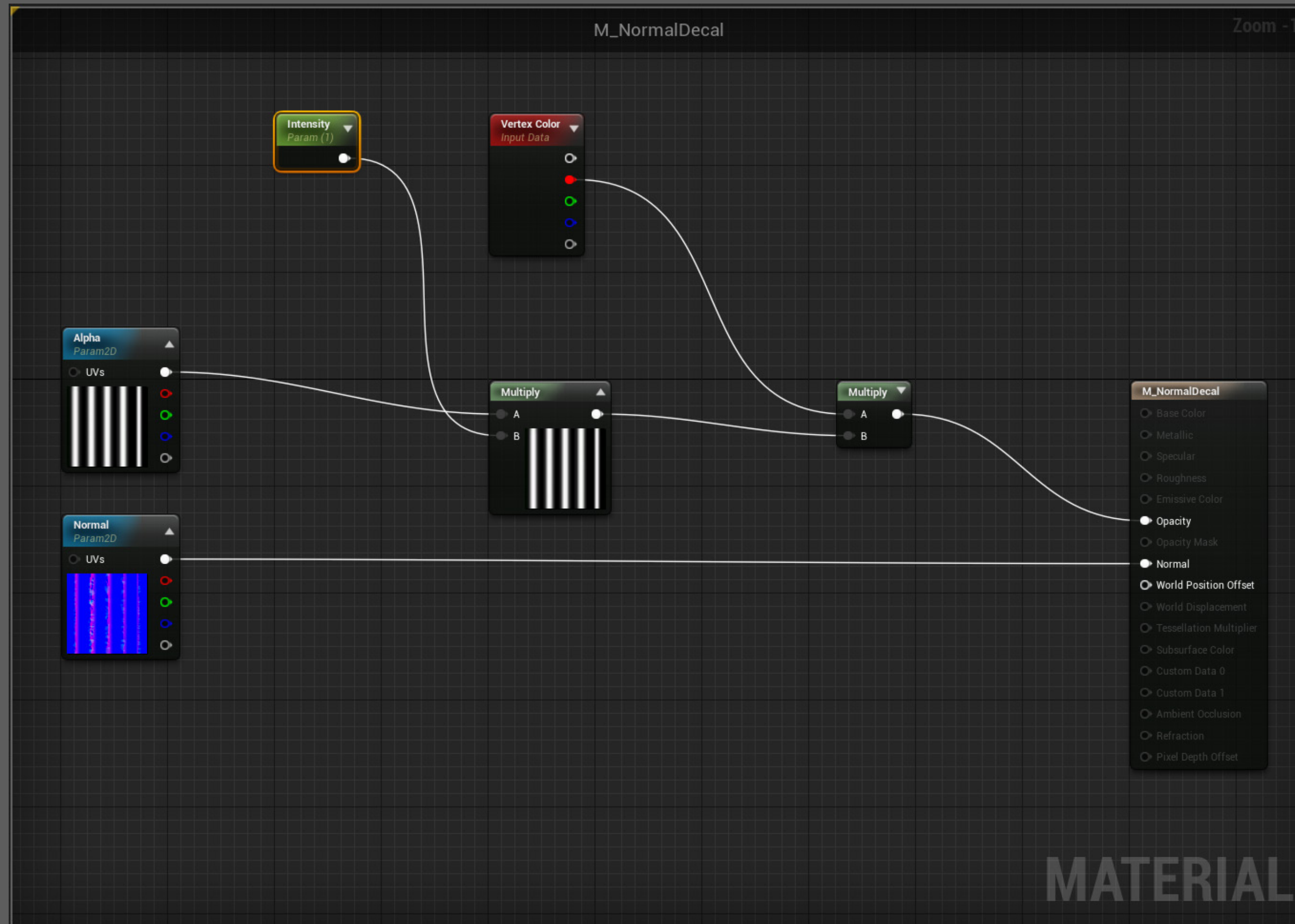
The normal edge decals only work with lights set as Real-time or Mixed.



(This is a copy paste of the Unreal section if you read that you can skip this one) In very short, the reason why is not working with the static lights is that the normal maps are used for faking the lighting of bumps and dents by making the rays bouncing in a different direction from the actual geometry. If you bake the lighting (baked) you are baking the light as a map onto the meshes which means that you are not going to have rays which means that the normal map can't work properly



EDGE NORMAL DECAL + VERTEX PAINTING IN UNREAL

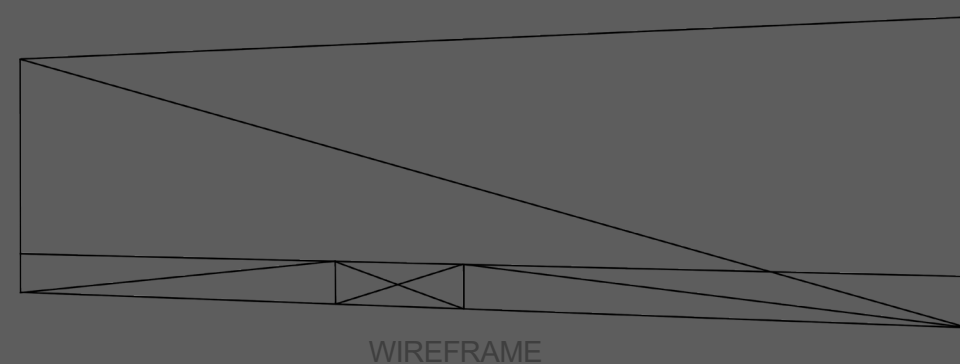
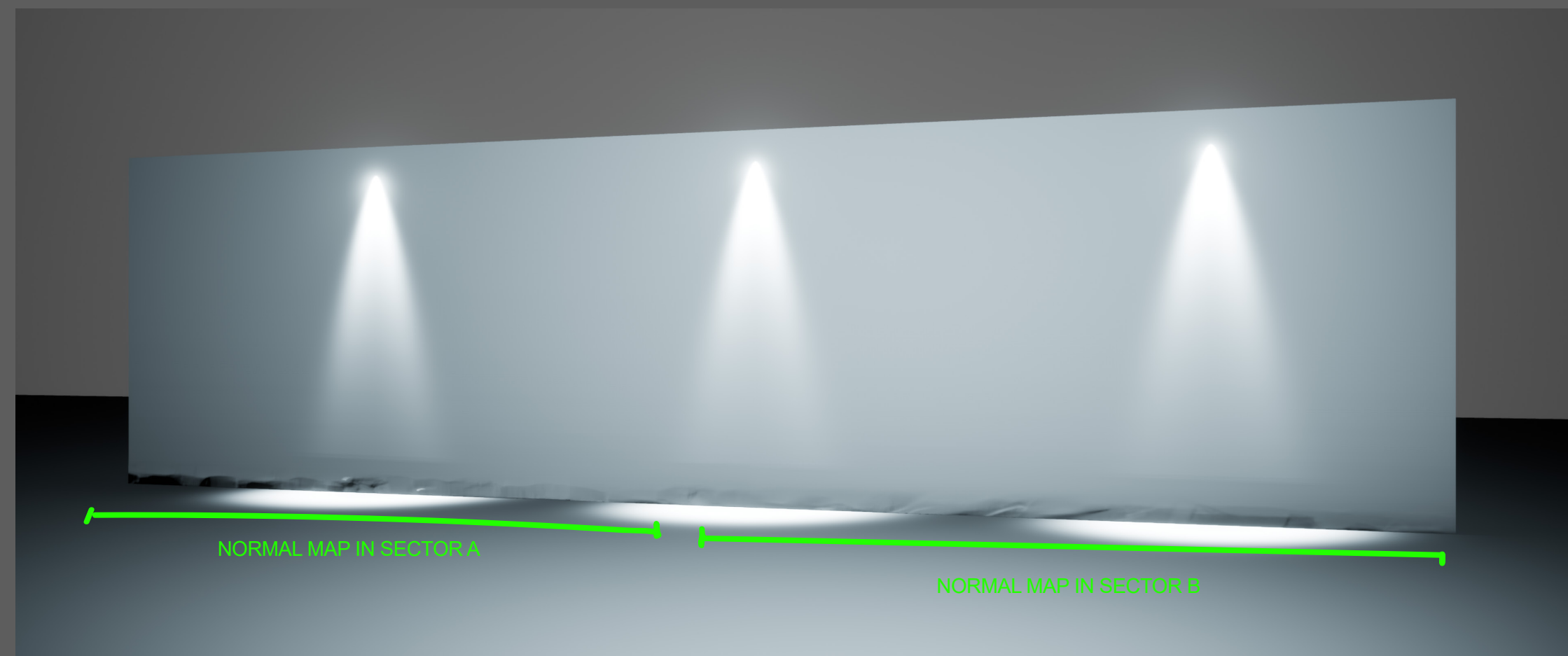


Another technology worth mentioning that you can use to improve the edge decal is vertex colour. The vertex colour will give us an extra four values you can use combined with your master material, almost for free. In the example on the left I'm using the RED channel to mask out the decal. (I've also exposed a global float to control visibility) There are plenty of ideas with the vertex colour, for example, you can control the normal intensity with a channel, you can offset the texture to a different slot, etc. etc.

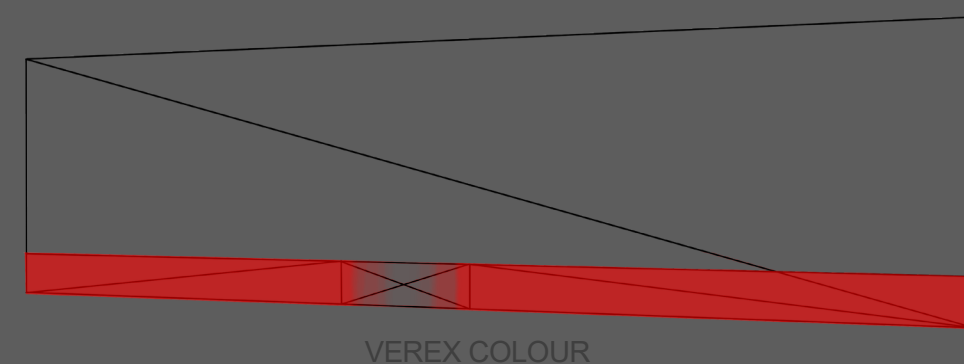
All good but it comes with a price when you use vertex colour you might face some LOD issue since this workflow relay on vertex density which is something that you want to reduce in your LODS.

You can add colours to your vertex in Unreal Engine with the paint tool as well as in your 3d software (Maya, Max, Modo etc)

BLENDING USAGE



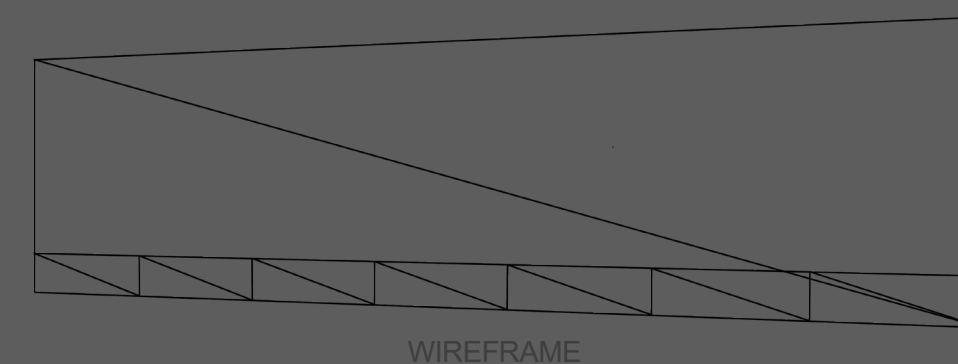
WIREFRAME



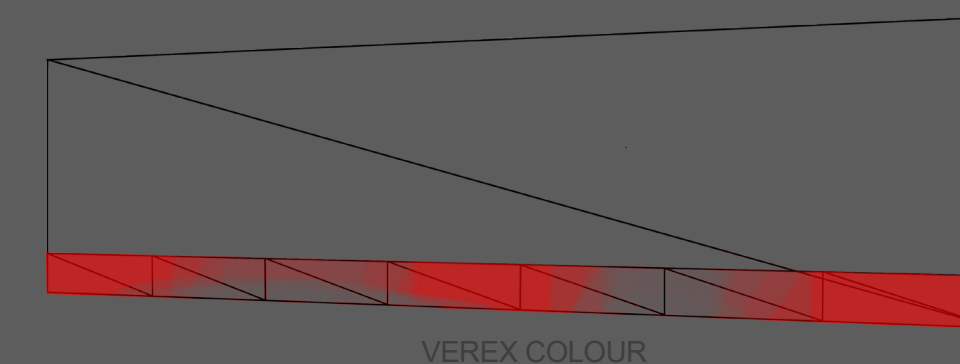
VEREX COLOUR

In this usage, you can use vertex colours as a way to blend between two different normal edge decals. For example, you can have soft concrete dents combining with a destroyed concrete. In this example, I'm overlapping the two decals in the center part, and I'm erasing the red colour from the last two vertexes that are overlapping. In this case, you shouldn't have significant LOD popping issues since you have two different meshes made of very few faces.

MASKING USAGE



WIREFRAME



VEREX COLOUR

In this case, I'm using the vertex colours to mask the normal edge decals to break that repetitiveness and add some variation. As you can see from the wireframe, I added some vertex in order to have more control when using the vertex colour. The disadvantage or proper you might face with this technique is the LOD popping out. If you generate LOD automatically, Unreal, Sympligon or whatever tool you are using they will remove vertexes which means that you are going to lose density of your mesh in the generated LOD leading to popping.

PRO AND CONS

ADD DETAILS IN A NON-DESTRUCTIVE WAY

BREAK THE HARSH MESH EDGES

FAIRLY EASY TO CREATE

FEWER UNIQUE TEXTURE
FOR CORNER DENTS

THE TRIM CREATION PROCESS CAN
BE EASILY AUTOMATED

DEFERRED DECALS USAGE

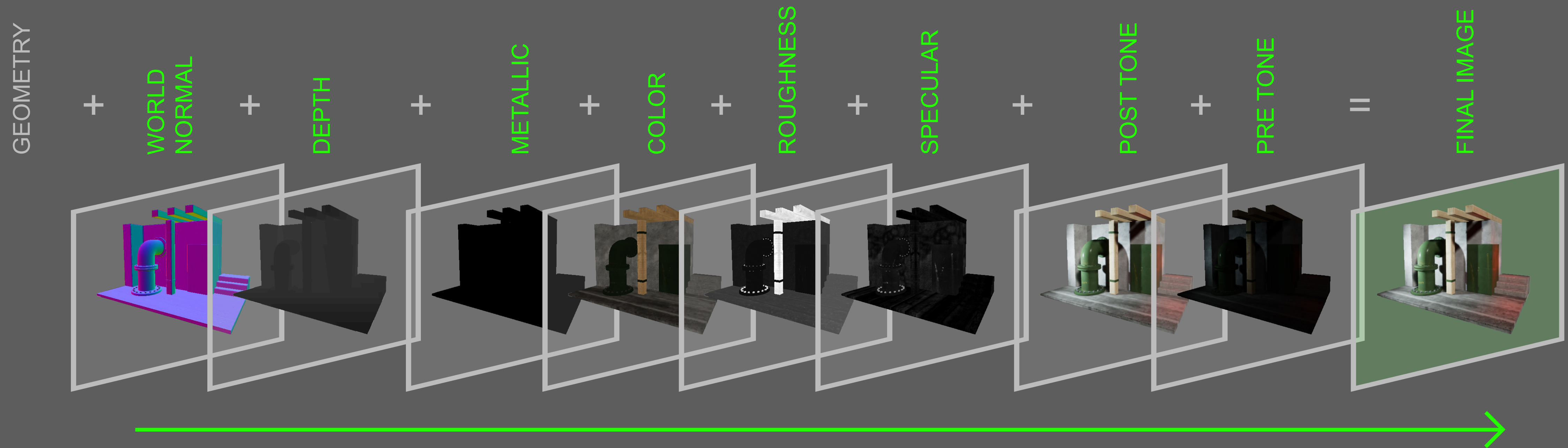
EXTRA MATERIAL SLOT

NOT WORKING WITH STATIC LIGHTING

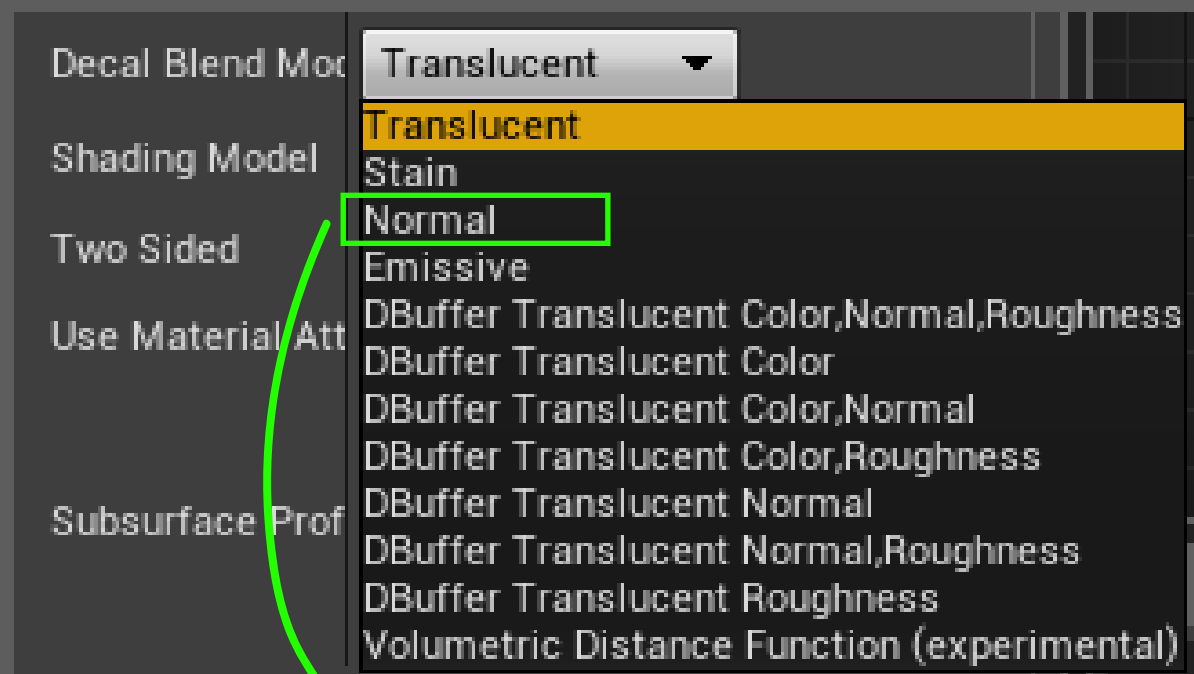
ISSUES WITH REALTIME GEOMETRICAL
DESTRUCTION

IT GIVES NO SILHOUETTE TO THE EDGE

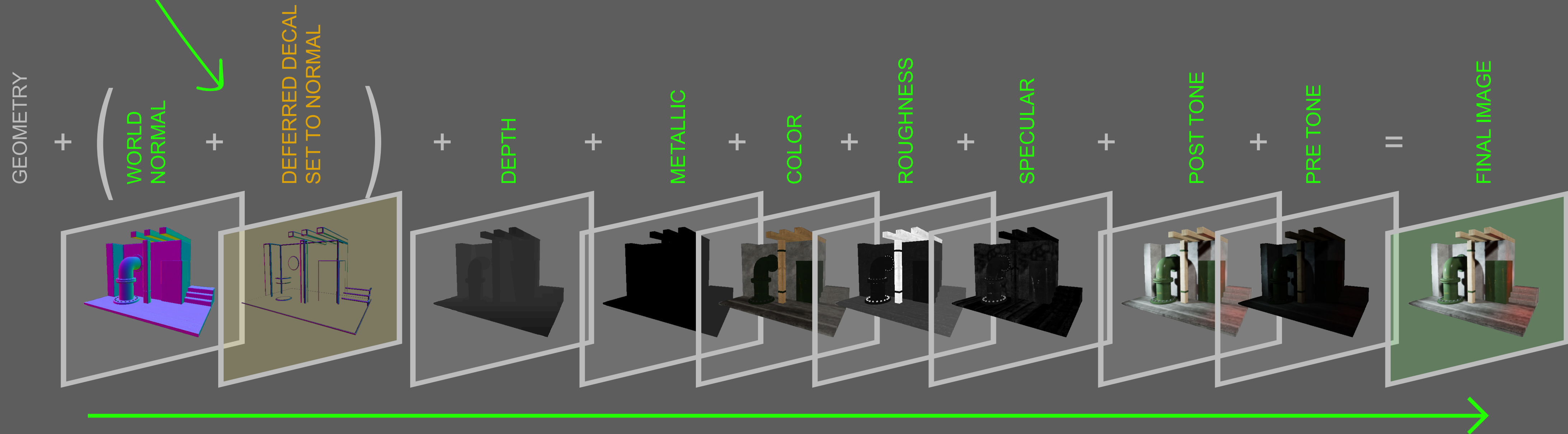
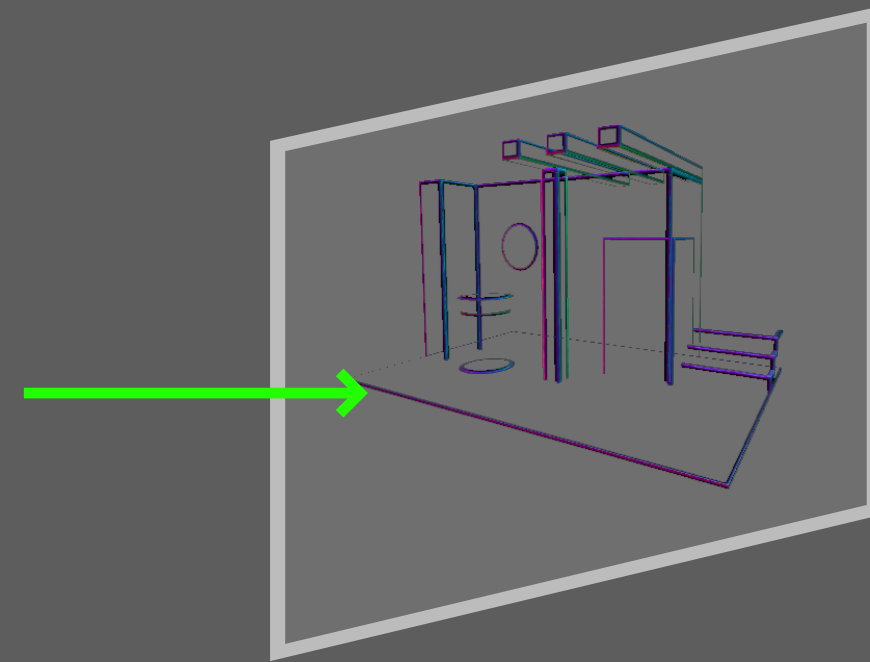
UNDERSTAND WHAT YOU ARE DOING



In a very simplified way, showing only a few of the passes, this is what happens in the buffer every time you're rendering a frame with the Deferred Shading pipeline. Through shaders we can access all of these layers and override or change properties for each of them



When we create the shader (in Unreal), we select Deferred Decal as “Material Domain”. Basically we are saying to the shader to go affect one or more of the passes previously shown. How do we choose which pass we want to affect? When selecting Deferred Decal, you are going to get “Decal Blend Mode” enabled. From there you can pick the different way the decal will blend, in our case we have “**NORMAL**” WHICH MEANS THAT WILL ONLY AFFECT THE **NORMAL OF THE OBJECT**. If we want some colour variation (for example dry blood decals, or paint marks), you may choose Stain, and if the full material is needed you just leave it on “Translucent” (for example better paint decals with rough variation).



Everything come with a cost. As you can tell in this “flow chart”, we added an extra pass that the buffer needs to go through every frame. Ideally, it is not a big deal. The problems start when you overlap many deferred decals on top of each other which is very likely to happen since they are so coool :) so be careful!

WRAPPING UP

Now that you learned how this technology works and how cool the normal edge decals are. Try to think of other ways you can use this workflow for different stuff.

For example, imagine if you want to add a patch with a tiny drop on a tiled floor. Or you want to enrich your storytelling with some scratches on a wall. Or even add some footprints marks on a muddy floor.

All of these examples are achievable with the same tools I showed you in this tutorial. But I want you to stretch your mind and get to the solution by yourself so that by trying you will master this workflow. If you really can't think of anything feel free to drop me a message tho :)

LINKS AND REFERENCE

<https://gfycat.com/InconsequentialObeseClam#Inspiration>

https://www.reddit.com/r/Unity3D/comments/7ertg2/shader_which_only_writes_to_normal_buffer_for/

<https://geofflester.wordpress.com/tag/substance-painter-2/>

<https://geofflester.files.wordpress.com/2016/10/substancepainterdecalscene.png>

<https://docs.cryengine.com/plugins/servlet/mobile#content/view/1310733>

<https://simonschreibt.de/gat/fallout-3-edges/>

<https://polycount.com/discussion/155894/decal-technique-from-star-citizen/p3>

<https://www.youtube.com/watch?v=iNdZ2LKih9I>

<http://www.richterdesigns.co.uk/mobilesite/scripting.html>

http://www.richterdesigns.co.uk/graphics/decal_generator.png

<https://answers.unrealengine.com/questions/75508/is-it-possible-to-use-custom-geometry-decals.html>

https://docs.unrealengine.com/en-us/Resources/ContentExamples/Decals/1_2

<https://pastebin.com/g5RYj45H>

<https://forum.unity.com/threads/how-do-i-write-a-normal-decal-shader-using-a-newly-added-unity-5-2-finalgbuffer-modifier.356644/>

https://en.wikipedia.org/wiki/Deferred_shading

<https://learnopengl.com/Advanced-Lighting/Deferred-Shading>

<http://blog.digital-horror.com/introduction-to-deferred-rendering/>

<https://simonschreibt.de/gat/renderhell-book1/>

IF YOU LIKED THIS TUTORIAL

If you liked this tutorial please push the “thump up” and let me know in the comment on Artstation what you think was good, bad or unclear, as well as if you have any suggestion feedback or ideas for the next ones, feel free to spread it around with your friend or colleague by sharing this link:

<https://goo.gl/zMhqUK>

Follow my artstation page and you should get notified when a new tutorial is out

OTHER TUTORIALS:

