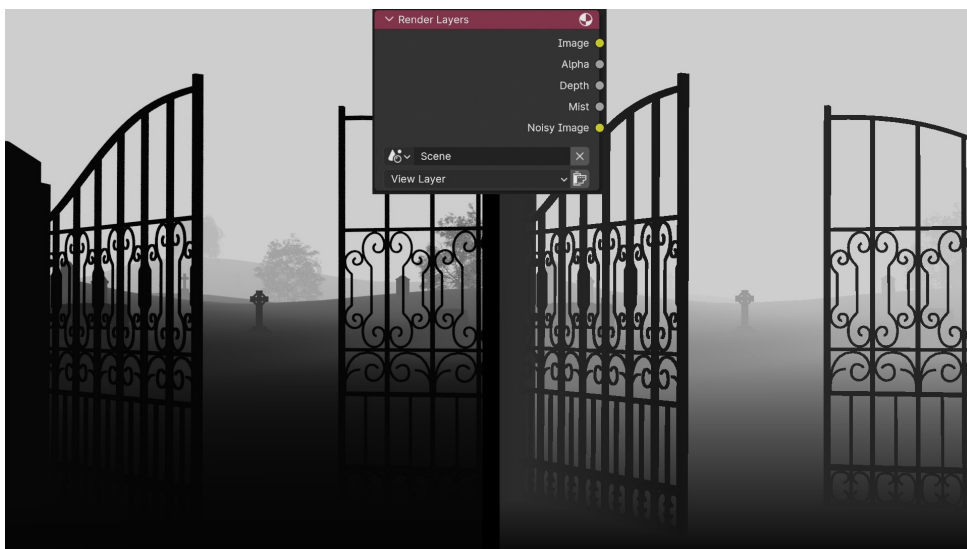


This PDF is available for free; feel free to share it!

[Subscribe to my YT channel](#) & turn on notifications to get updates on new videos!



INTRODUCTION

Hello everyone!

In this tutorial, we will examine the features, settings, common points, and differences between **Depth** and **Mist**: two Compositing Render Passes in Blender that provide us with **information about the distance of objects** or the depth of the scene in post-production.

Both can be used to obtain images that appear **black at points closest** to the camera and **white at points farthest away**, allowing us to create **effects** either by directly blending these images with the rendering or by using them as **masks** to create other effects, as I have shown, for example, in my series of videotutorials on Atmospheric Perspective.

However, there are some differences, not only in how **the range of values** is calculated in the virtual scene, but also in other characteristics of the resulting

images, which can make them more or less suitable for certain uses. To try to explain these differences better, throughout the tutorial, I will use a couple of 3D scenes and show you the differences in the results by changing some settings each time.

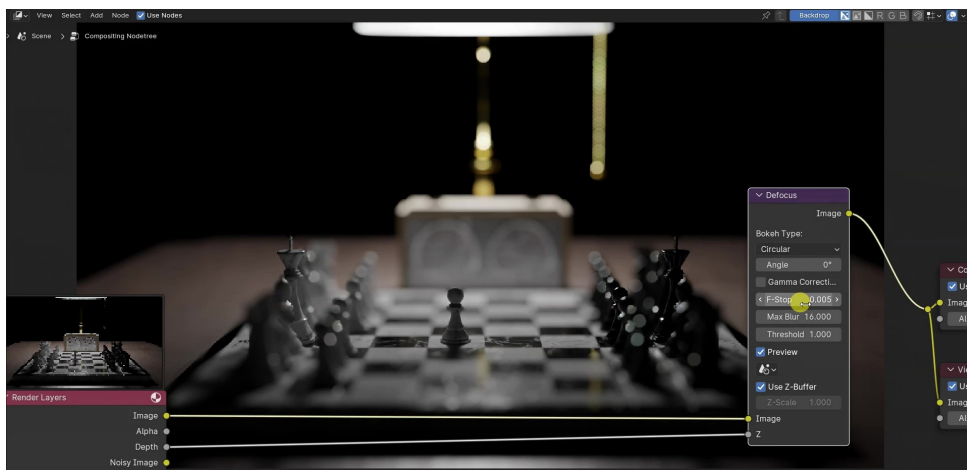
THE BOTTOM LINE

In short, the difference between Depth and Mist is that **Depth returns the actual distance values** of objects from the virtual camera, while **Mist introduces fog starting from a certain distance** from the camera and increasing it along a certain depth, beyond which everything becomes white.

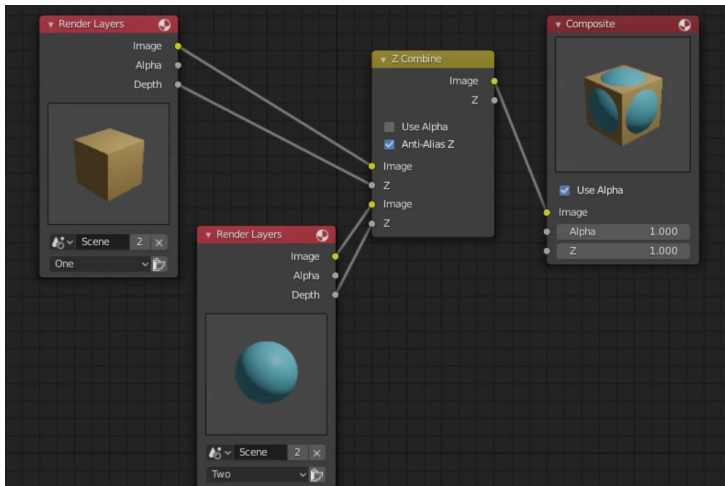
Z-DEPTH

Therefore, **Depth** is more suitable for applications where it is necessary to use the **true distance values**, such as with Defocus and Z-Combine nodes.

Defocus allows you to focus on elements at a certain distance from the camera, blurring others (thus implementing depth of field).

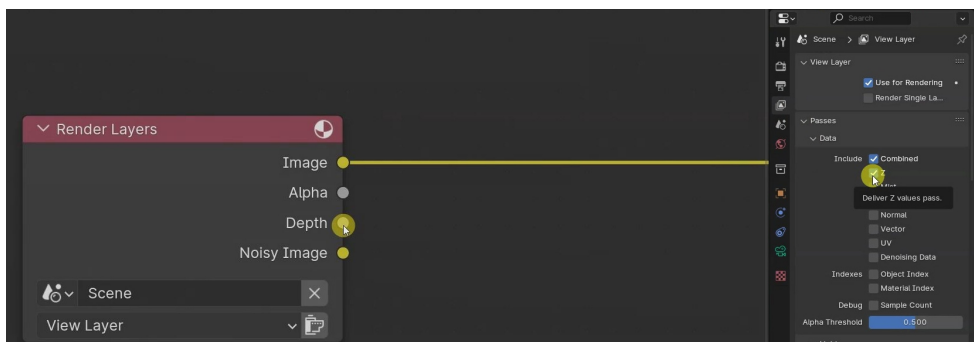


Z-Combine, on the other hand, allows you to **blend two scenes** by determining which pixels to show and which not to show based on the distance values from the camera, stored in the Z or Depth fields of each scene or image.

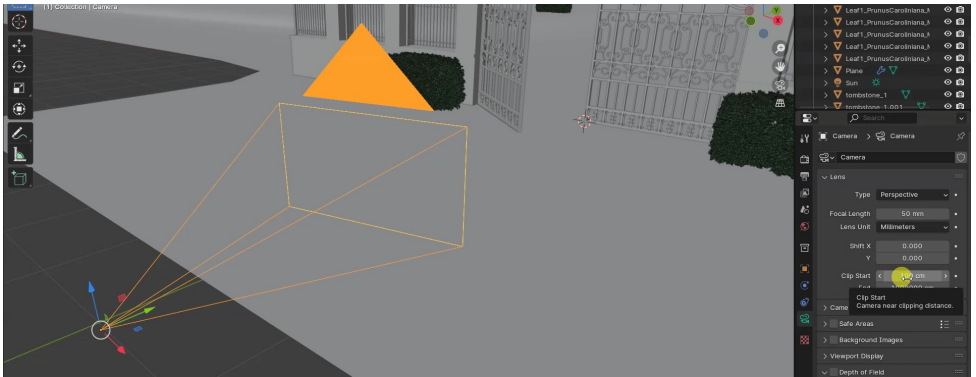


However, this doesn't mean that Depth cannot be used in other cases: I myself used it in the series on Atmospheric Perspective, but there **I had to normalize the information returned by Depth**; we will soon see what normalizing Depth means and what consequences it can have on the data.

The Depth Render Pass is selected by default in the Passes section of the **View Layers** tab in the Properties editor; in that tab, this Render Pass is called Z, while in the Compositing Render Layers node, it is called **Depth**.



The minimum distance value returned by Depth is given by the **Clip Start** value of the Camera object because all objects placed closer to the camera than Clip Start are not rendered, so - if the output of Depth is not normalized - that is the minimum value that Depth can return.



The **maximum distance** value returned by Depth is given instead by the **Clip End value** of the Camera object because all objects placed beyond that value are not rendered. The **background** of the virtual universe is considered at an **infinite** distance, so it will have the maximum value in the Depth mask.

You can **display the Clip Start and Clip End points** in the 3D scene by selecting the virtual camera and activating the **Limits** option in the Viewport Display section of the Data tab in the Properties editor.

This way, it will be easier to set the Clip End value appropriately.



MIST

Mist, on the other hand, returns an image that becomes **progressively whiter** as the distance from the Camera increases. However, in this case, the starting point of the fog effect is **not given by Clip Start**, but by a value that we can specify as desired, called **Mist Start**.

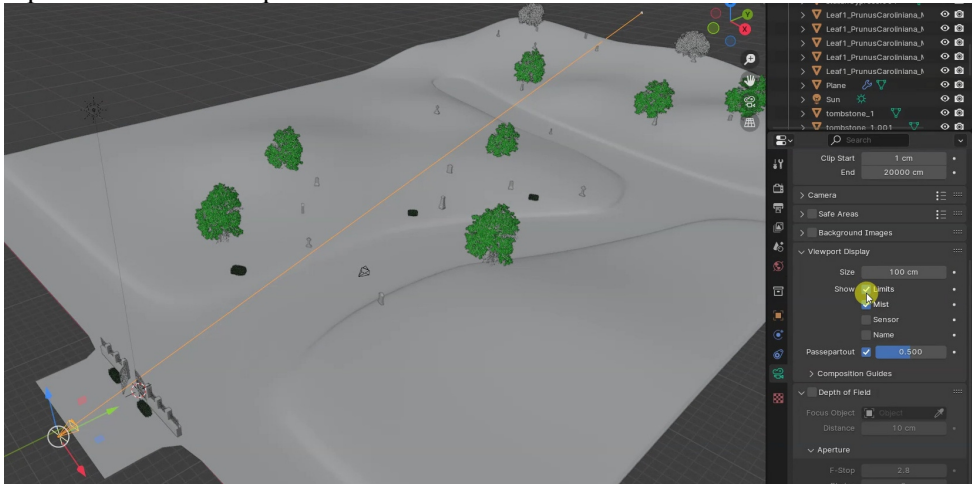
Unlike Depth, which is active by default, **the Mist Render Pass needs to be explicitly activated** in the Passes section. This operation should be done **before starting the rendering**; otherwise, the information will not be processed, and you will need to start the rendering again to use it.

After activating the Mist Render Pass, **a new section called Mist** will appear in the **World** tab, where we can set Mist Start and other **parameters**.



There's another difference between Mist and Depth: **Mist doesn't use Clip End** to determine the distance beyond which all objects render as white. Instead, it uses a parameter called **Mist Depth**. So, all objects placed at a distance of **Start + Depth** will be rendered as white.

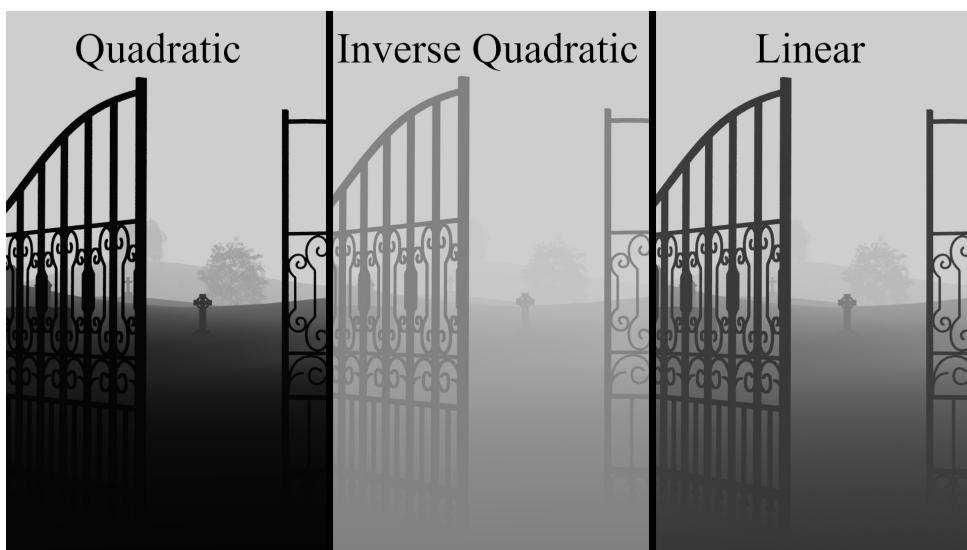
You can also **display Mist Start and Mist Depth in the 3D Viewport** by simply activating the Mist checkbox in the **Viewport Display** section of the Data tab. If you've also activated the Limits option, it's advisable to deactivate it because their representations overlap.



Before rendering again, I'm setting the Mist Start and Depth values so that the effect **overlaps** with the Clip Start and Clip End values.

In the Mist section of the World tab, there's another parameter: **Falloff**, which indicates how to implement the fog over distance. The default mode is **Quadratic**, but there are also options like **Inverse Quadratic** and **Linear**.

Quadratic implements the same transition as lighting, providing a smoother transition from 0 to 1. **Linear** implements a slightly sharper transition than Quadratic. Finally, **Inverse Quadratic** transitions very quickly from 0 to 1, showing dense fog even for foreground elements.



Z-Depth doesn't have this parameter because it's not an effect to be implemented following a formula; rather, it returns the true distance values of objects... one could say **it's always linear**, but it doesn't make much sense to make this clarification.

As I mentioned earlier, **Z-Depth** is therefore more suitable for obtaining information about **true distances**, while **Mist** is more suitable **for creating the effect** from which it takes its name, because it allows us to set the characteristics as desired.

By **rendering now** and examining the passes in a **Compositor** window, we would get an interesting result from the **Mist** output, while the mask coming from **Depth** will be, at least apparently, unusable.



NORMALIZING DEPTH

Let's now see what "**normalizing Depth**" means and why the **Normalize** node is often used in various tutorials. As mentioned earlier, the values returned by Z-Depth are not in the range from 0 to 1 but actually indicate **distances from the virtual Camera**. Therefore, it can be useful if, for example, you want to introduce **an effect exactly at a certain distance from the camera**.



The stored numbers can be **very big**, so the Z-Depth field stores the data using more memory than the 8 or 16 bits of most image files like JPG or PNG; I'm almost sure it actually uses **32 bits**.

Moreover, **in the Backdrop**, all values greater than 1 are **truncated (or "clamped")** to that value, and the corresponding pixels are displayed as white. However, the **Z-Depth values** are given by Clip End, which is generally **much greater than 1**.

For these reasons, the returned values are **not displayed correctly** in the Backdrop of the Compositing editor because in that tab, all values greater than 1 are rendered as **white**. Therefore, in most cases, the returned images appear very bright and lacking in details...

... in fact, **in the scene I used**, the Clip Start value is 1 and Clip End is 20000, so the Depth Pass appears completely white!

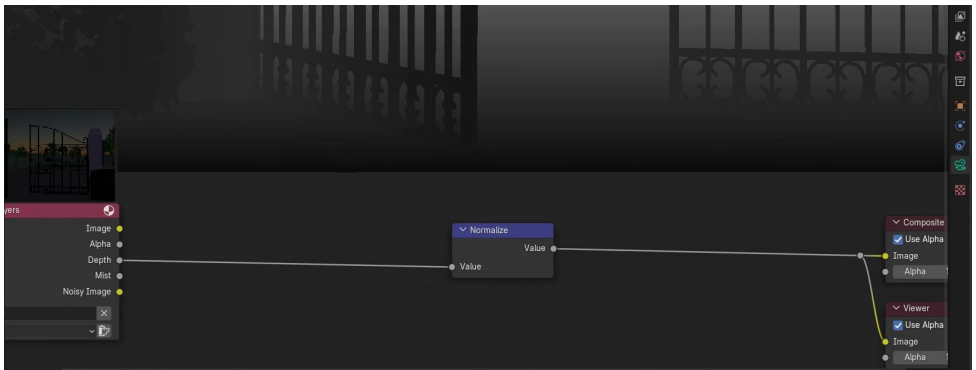
For this reason, the **Normalize node** from the **Utilities** group is often used to map the values to the range **from 0 to 1**. However, on this point, we need to pause because there is an **important clarification** to make.



The Normalize node doesn't map the values between Clip Start and Clip End to the range from 0 to 1. Instead, it considers the minimum and maximum **values of the pixels present in the image it processes**.

However, **this isn't the image we see in the Backdrop** because, as mentioned earlier, the Depth channel stores much more data than what is visible.

That's why, if we connect a Normalize node between the Depth output and the Viewer, **the elements at the bottom of the rendering appear black**: because those are the points closest to the virtual camera, which therefore have the darkest values in the Depth mask, and thus will be **rendered as black in the normalized version**, even though they are clearly not at Clip Start distance from the Camera!



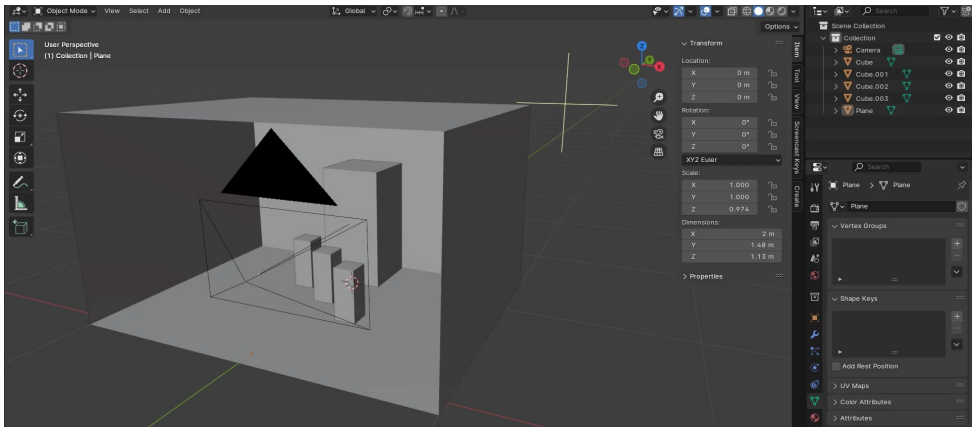
Similar considerations can be made for objects farther away and for pixels that will appear **white in the normalized version** of the Depth mask.

As further evidence that Normalize acts on the minimum and maximum values it finds in the image provided to it, look at what happens if I also Normalize the information coming from Mist: the pixels at the bottom, which are the darkest, are remapped to 0 and, therefore, rendered as black, while the pixels of the farthest elements were already white and thus remain unchanged.

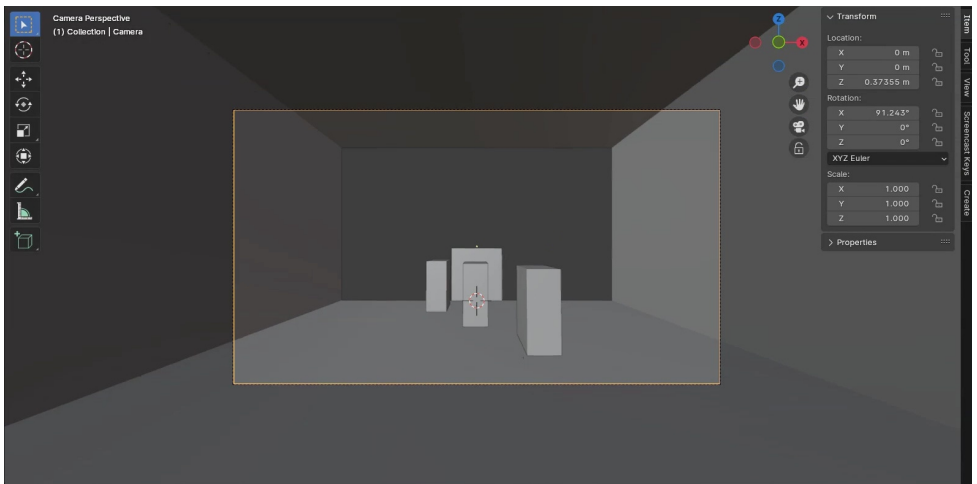
ANOTHER EXAMPLE

I'll show you **another example** to better understand this concept. In the scene I'm showing you now, the camera and some objects are positioned inside a **room**.

The Clipping End value of the camera is set in such a way as to render the front face of the farthest box placed in the room, but **not the back wall** of the room.



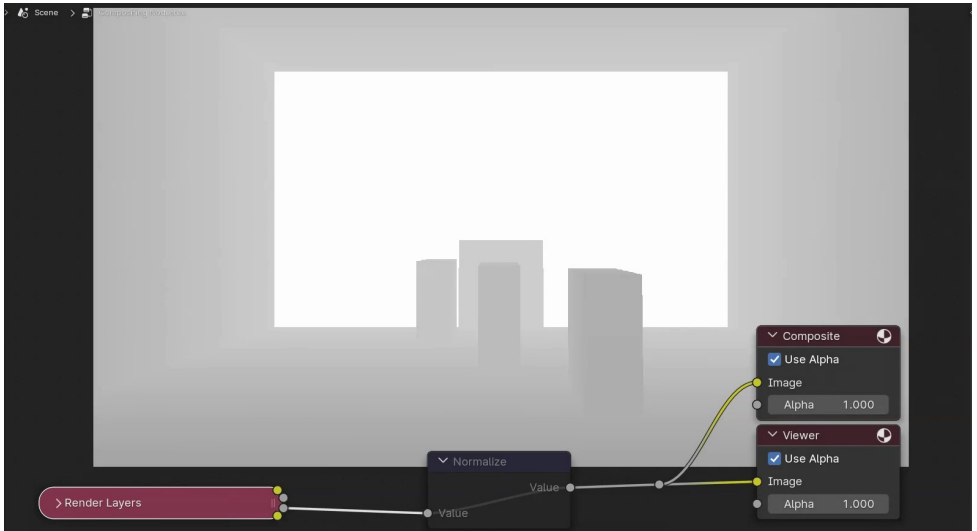
In fact, **not even the farthest box** is entirely visible, but this is due to the perspective effect of the framing.



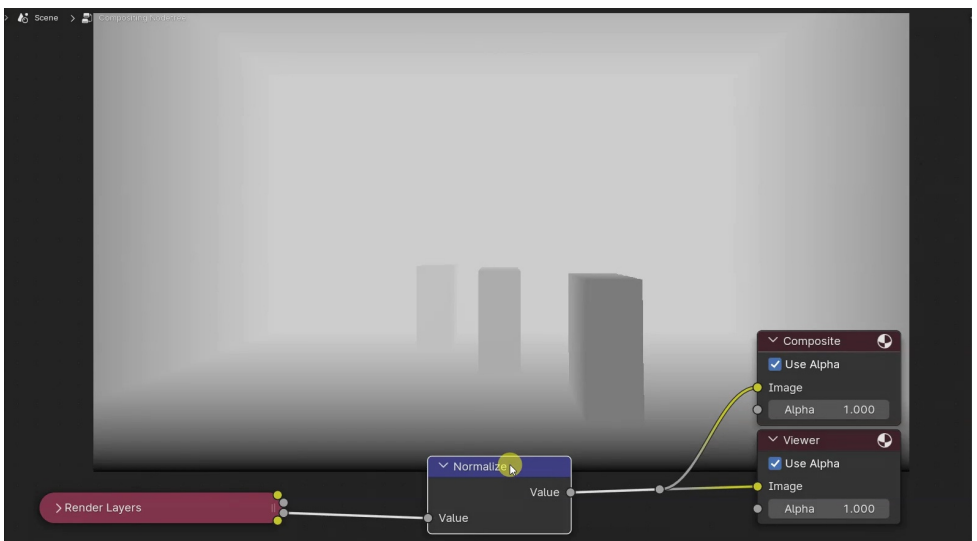
There are **no objects placed near the Clip Start**, which is set at 10 centimeters from the camera: even the walls of the room at the edges of the frame are at a greater distance than that value from the camera.

If I start rendering now and examine the **not normalized Depth Render Pass**, the **back wall is completely white** because, in fact, the Depth Pass doesn't see beyond the farthest box and renders everything else as white.

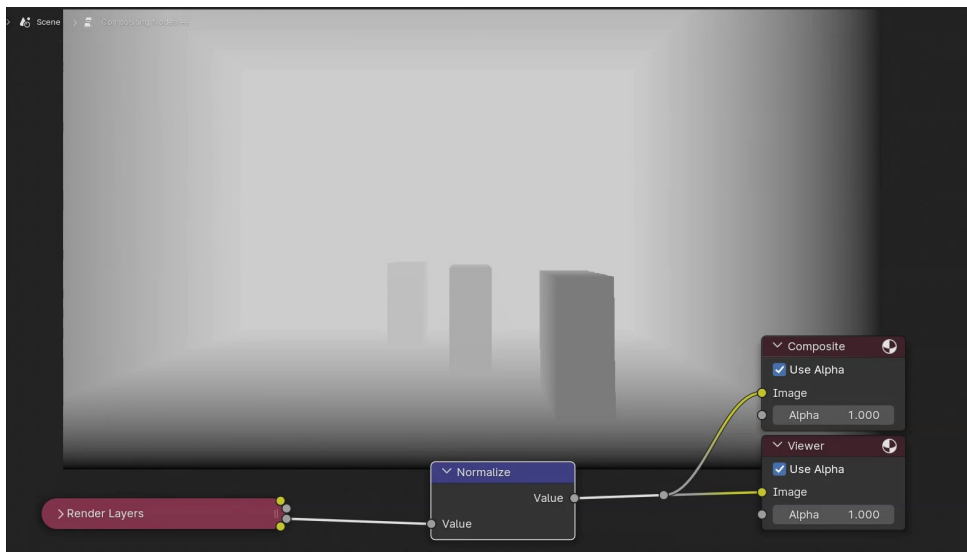
However, there are **no black areas** because, as mentioned earlier, there are no objects placed at Clip Start.



But if we use a **Normalize node**, a **black band** is introduced at the bottom, which we also saw in the rendering of the scene used previously: that part of the floor is evidently **the closest to the camera**.



We can verify what was said by **moving one of the side walls inward**, so that they are closer to the camera, and then **start rendering again**.



It's important to understand **how the Depth mask is altered by Normalize**.

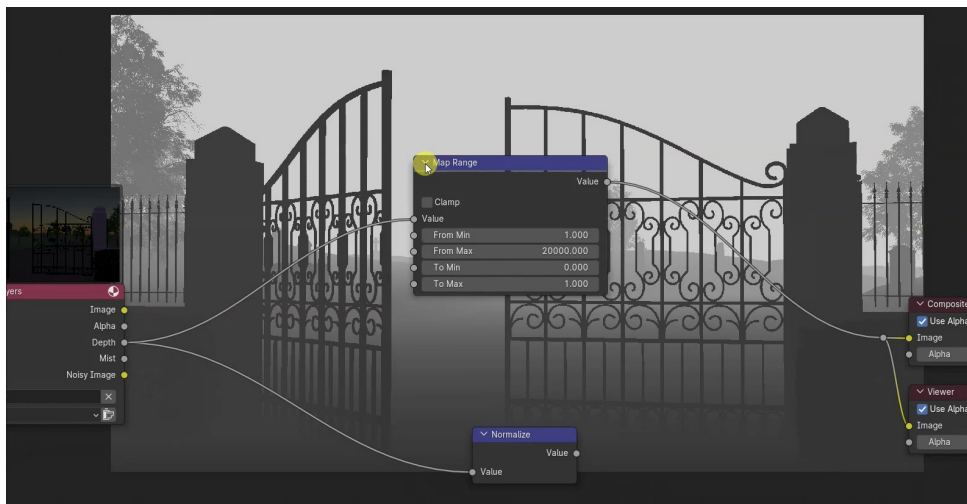
This node shouldn't be used with Mist, and its use with Depth should also be decided judiciously: it's useful for creating masks for special effects, such as the aforementioned Atmospheric Perspective, while in other cases, it should not be used.

HOW TO MAP THE Z-DEPTH MASK - THE RIGHT WAY

So, you might be wondering if there is **a way to map** the Clip Start and the Clip End values in the range **from 0 to 1**, to see the Depth mask in the Backdrop...

... well, here it is: **add a Map Range node** from the **Utilities** group in the Compositor, then make the **From Mix** and **From Max** values of the node equal to the **Clip Start and Clip End** values of the Camera.

You'll get a mask similar to the Mist pass, but of course you'll lost the real depth values.

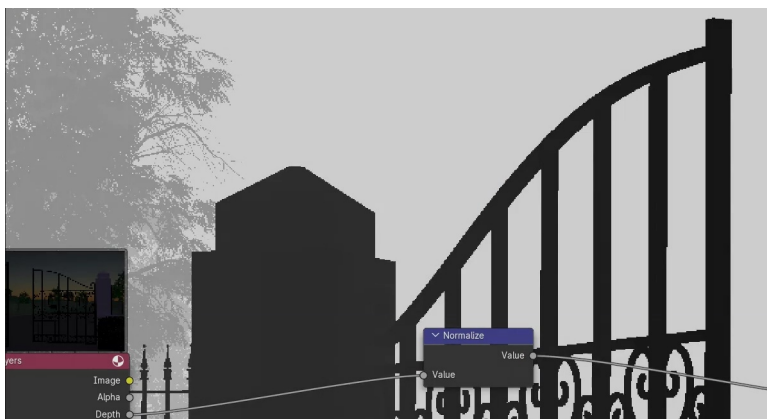


SAMPLES & ANTI-ALIASING

There's one last important difference to underline before closing this tutorial: **the number of samples** used to calculate the distance values of objects in the scene.

Mist, in fact, also implements **Anti-Aliasing**, which is not calculated by Z-Depth.

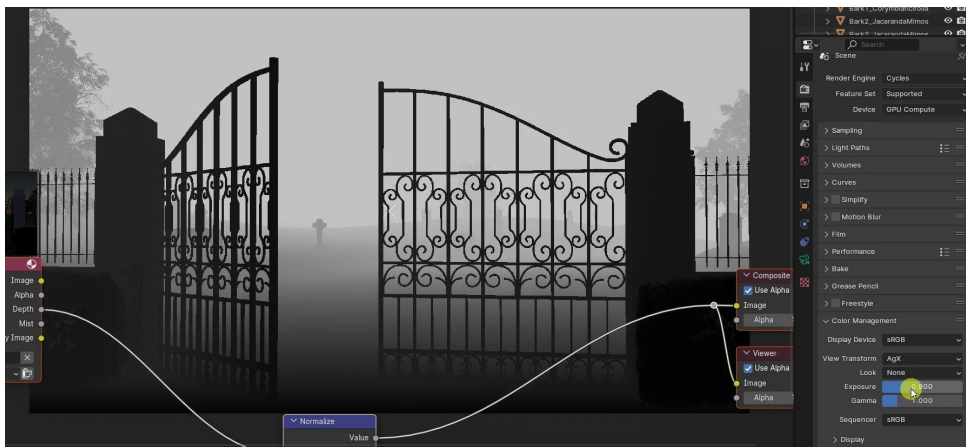
The Z-Depth value, in particular, uses only one sample, so it can be used without problems **when effects** like Motion Blur or Depth of Field **are not used**, i.e., in cases where blending the depth values is not necessary. When it is necessary to blend the depth values, using Mist is more appropriate.



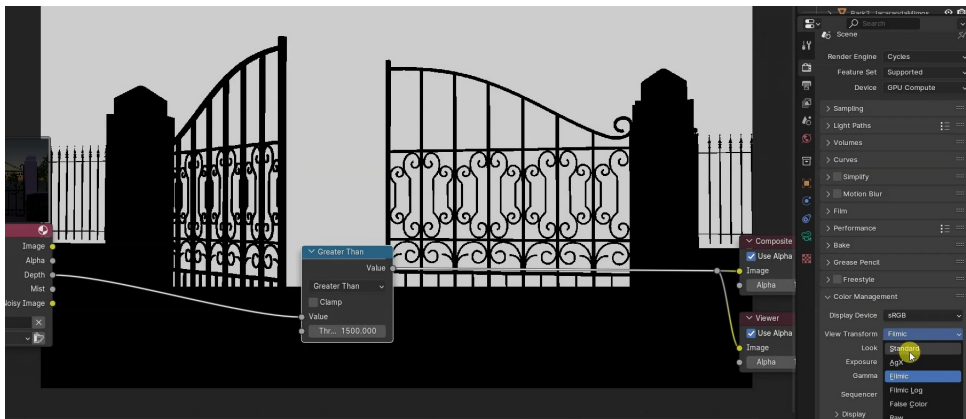
COLOR MANAGEMENT VIEW TRANSFORM & CONCLUSION

Finally, a consideration about the **Color Management** settings of the rendering and **how** the Z-Depth and Mist masks **are displayed** in the Backdrop.

Depending on the mode chosen for the **View Transform** parameter, the results shown on the screen for both Mist and Depth, whether normalized or not, can vary greatly.



However, these transformations should only affect **how the data is displayed in the Backdrop**, so they shouldn't alter their **intrinsic values**.



Well, that's all for this tutorial! I hope it was helpful! See you soon!