

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!



Hello everyone!

This is a basic tutorial on Compositing in Blender, made with version 4.1, for beginners and focusing on **applying Ambient Occlusion** to your renderings.

Ambient Occlusion is a technique that, depending on how it is implemented, calculates or simulates the dark areas of objects which occur because ambient lighting fails to reach them. This happens in **corners, crevices, cavities, nooks, hidden areas**, and other similar situations.

It also occurs when **two objects are in contact with each other**, such as the objects resting on a desk in the scene I'm using for this tutorial. In fact, if Ambient Occlusion is not implemented, the rendering appears less realistic because the objects don't seem to be actually touching each other.

When Ambient Occlusion is implemented, **it helps to better define the shapes of objects** and adds a **sense of depth**, contributing to making the images more convincing and realistic.

In this tutorial, we will see **how to retrieve Ambient Occlusion** information, how to adjust the **size** of the ambient occlusion areas, and how to make the effect more or less **pronounced**.

But wait, there's more: at the end of the tutorial, I will also discuss what to do when, in the rendered scene, there is also **the virtual universe as background**, so stay tuned until the end!

In Blender, it's possible to implement Ambient Occlusion at **Material** level (meaning, setting it specifically for each Material available in the scene) and at **Rendering** level, using Blender's information on the position and shapes of objects within the scene. In this tutorial, we will specifically examine Ambient Occlusion at the **Rendering level, in Post-Production**.

For this tutorial, I'm using Cycles as the rendering engine, but the workflow is also valid with Eevee.

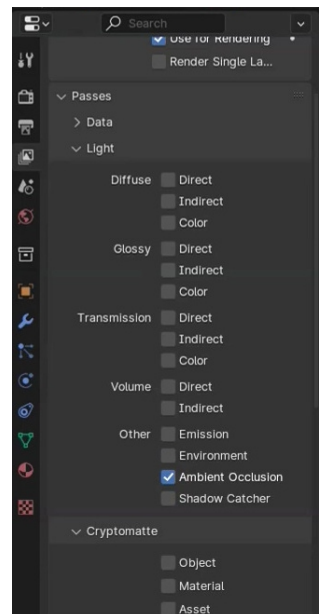
By default, when rendering a scene in Blender, **Blender does not provide Ambient Occlusion** information in the Render Layers node.

Therefore, the first thing to do is to **activate the Ambient Occlusion Render Pass** in the Light Passes section of the View Layer tab in the Properties editor.

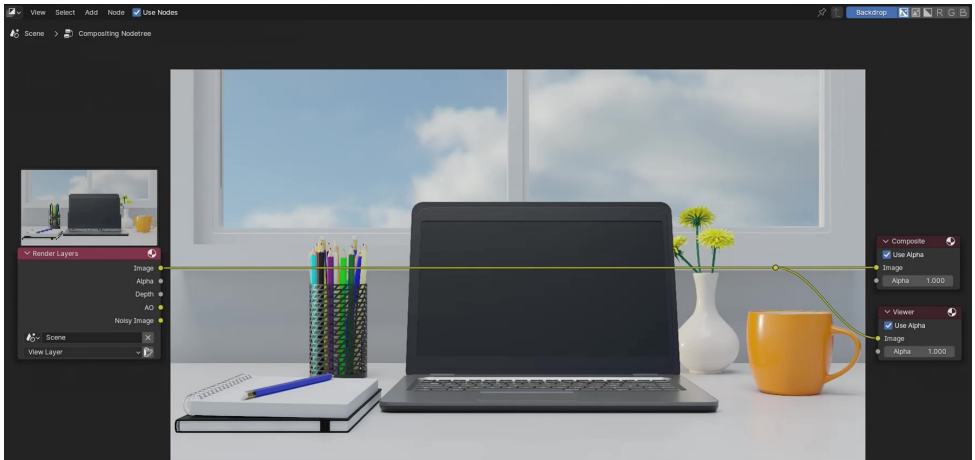
Once done, **start the rendering** and switch to a **Compositor** editor, where we can observe the changes in the background by activating the **Backdrop** function and inserting a **Viewer** node.

In my case, I'm inserting a **Reroute** node, which I'm connecting to both **Composite** and **Viewer** nodes.

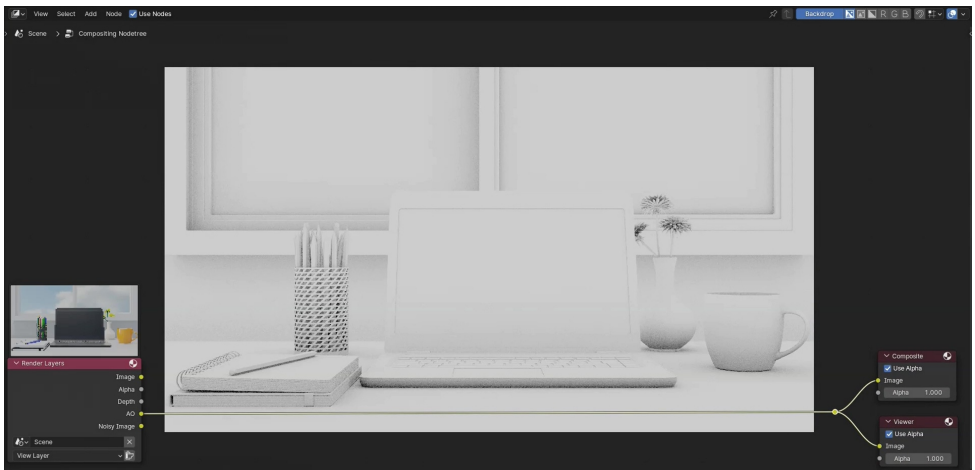
This way, the information sent to the Reroute node will be then sent to both the rendering being saved to disk and the one we are seeing as the background in the Compositor editor.



To **resize the compositing preview** in the background of the editor, press **V** or **ALT V**. To **move** that preview, use **ALT** and the **middle mouse button**.

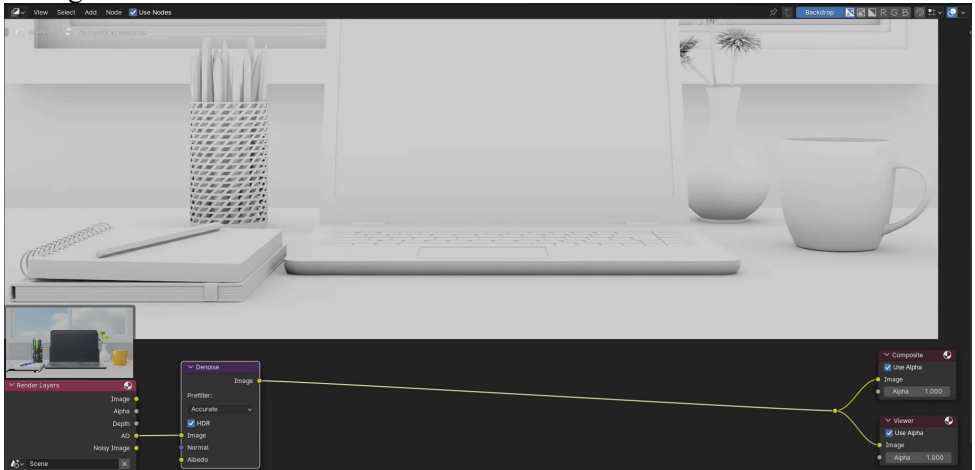


First, let's **connect the Ambient Occlusion** output of the Render Layers node **to the Reroute node**, to observe it.



Blender provides us with object occlusion information directly from the geometries available in the scene, so we notice occlusions between different objects, such as those between the cup and the desk, as well as within individual objects, as in the case of the flowers.

We also notice that **this information is very granular** or, to put it better, **noisy**... luckily, we have a node that is specifically designed to reduce noise, so let's add a **Denoise node from the Filter group** right between the Ambient Occlusion output and the Reroute node. The effects are immediately visible, even with the default settings of the Denoise node.



To add this information to the original rendering, we insert a **Mix Color node** from the Color Mix group into the setup, setting it to **Multiply** mode with a value of 1 for the Factor parameter. Next, we connect the **Image** output of the **Render Layer** node to the **first Image input** of the Multiply node.

Then, we connect **the output of the Denoise node** to the **second Image input** of the Multiply node and observe the result, which is visible by connecting the output of Multiply to the Reroute node.



Multiplying a color by a grayscale value has the effect of **darkening** that color because it basically means multiplying numbers ranging from 0 to 1. Therefore, the final result will be a value smaller than the original two numbers, resulting in a darker color.

The rendering immediately appears much more convincing than the original one. We can evaluate the effects of the Ambient Occlusion pass by **toggling the Multiply node on and off**, which we can do by selecting this node and pressing the **M key** (which is the shortcut for the **Mute** operation) on the selected node multiple times...

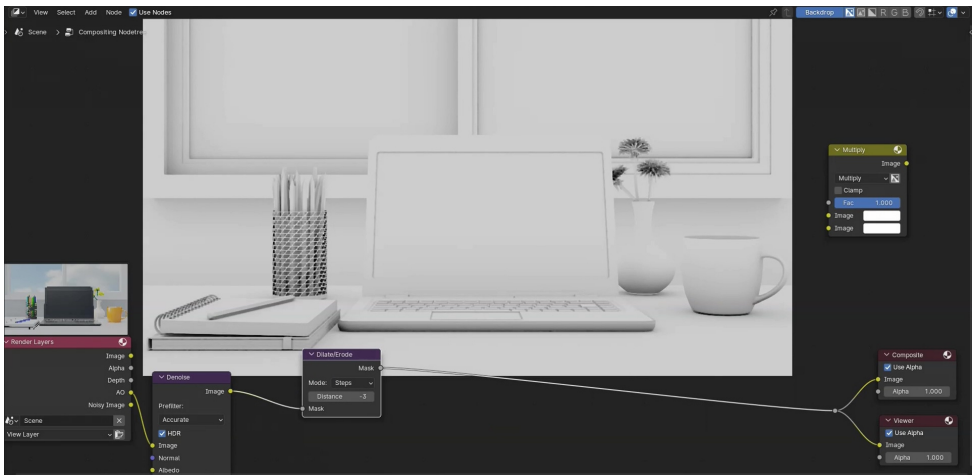
...however, we're not done yet, because we can **enlarge, shrink**, or make the information in the Ambient Occlusion channel **darker or lighter** before applying it to the image.

To better examine the effects of the operations I'm about to perform on the output of the Denoise node, **I connect this node to Reroute**.



To expand or shrink the areas generated by Ambient Occlusion, we can use a **Dilate/Erode** node from the **Filter** group. This node implements morphological operations, acting on the shapes of the dark areas.

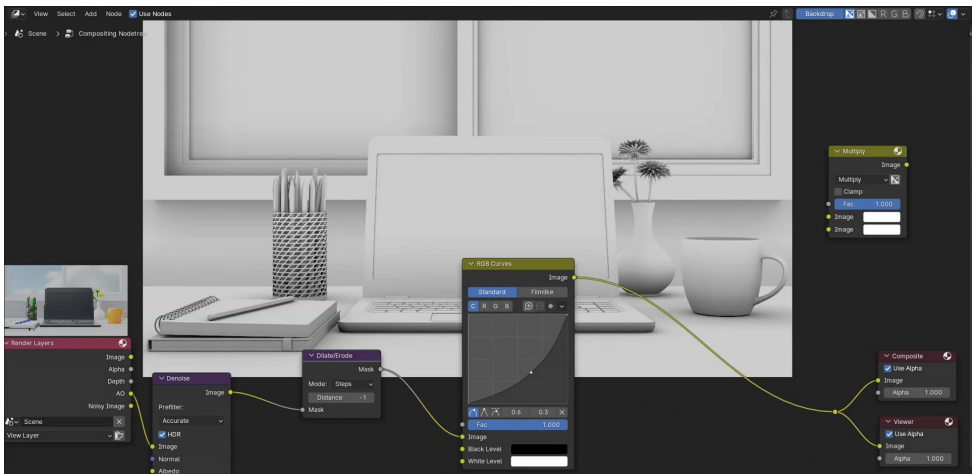
By setting a **positive** value in the Distance field, **we shrink the dark areas**, while setting a **negative** value we can **widen** them.



To make the Ambient Occlusion information **darker or brighter**, we can use a **Gamma** node, which allows us to darken or lighten the image without turning gray pixels completely black or white.

If you prefer, you can achieve a similar effect with an **RGB Curve** node by modifying the **C curve** to reproduce the Gamma curve.

This operation is very different from adding or subtracting values from the original pixels and is more suitable for this type of correction.



Now that we know the effects of both the Dilate/Erode node and the Gamma node, **let's connect the output of Gamma to the second input of Multiply.**

Then, we connect the output of **Multiply** to the **Reroute** node.

This way, we can modify the node values and observe the effects, this time on the original rendering.



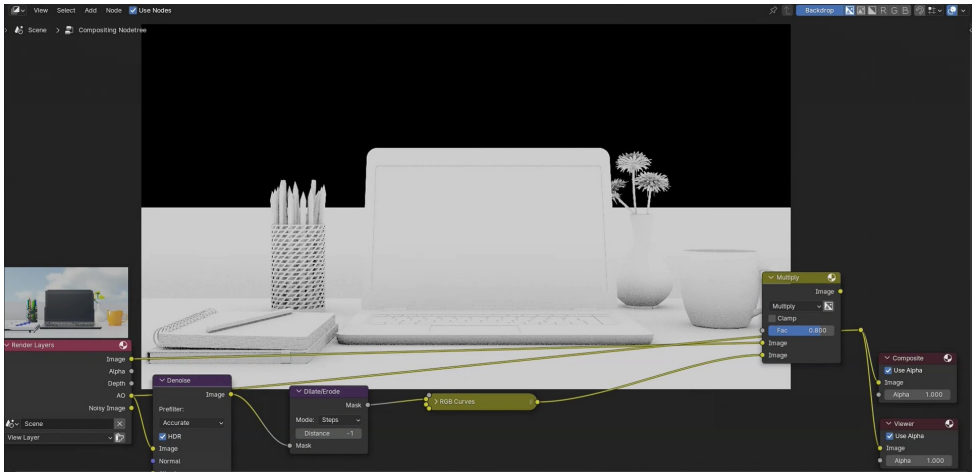
Finally, remember that the **Multiply** node also provides you with a parameter, namely **Factor**, to adjust the **intensity** of the effect applied to the original rendering. Setting this value to 0 will result in getting the original rendering only, while setting it to 1 will result in the full application of the effect, with intermediate values corresponding, of course, to proportional applications.



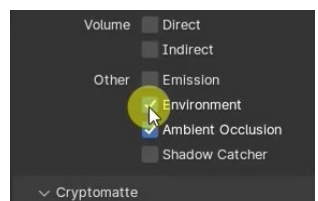
Before concluding the tutorial, one last note.

The scene I've used so far was set inside a room, and the **background of the virtual universe** didn't cause any issues because it was "filtered," so to speak, by the **Glass material** of the window.

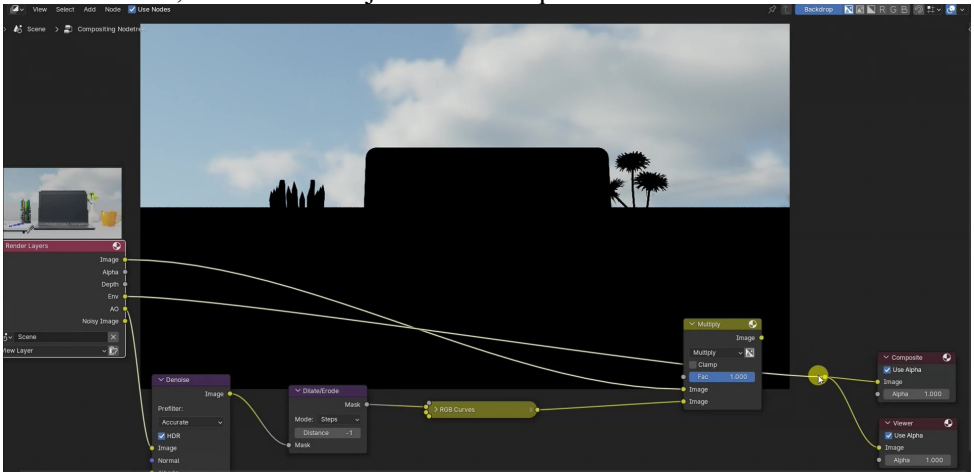
However, in outdoor scenes, the background of the virtual universe appears black in the Ambient Occlusion channel. Multiplying this information with the original image would result in the background being completely black.



To solve this issue, before rendering a scene of this kind, **activate the Environment Render Pass** too, which isolates the background of the virtual universe, representing it in color.



In this channel, all the other objects will be represented as black.



Now add a **Mix Color** node, change its mode to **Exclusion**, with a value of 1 for the Factor field, then connect the **Environment** Render Pass to the second Image input of this node. Then, connect the output of the Multiply node we used to apply **Ambient Occlusion** to the original rendering to the first Image input of the Exclusion node.



Remember: in this setup, the **Exclusion** node will work correctly only if the first image input is the product of Image by Ambient Occlusion; so, **do not connect Image directly to Exclude**.

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