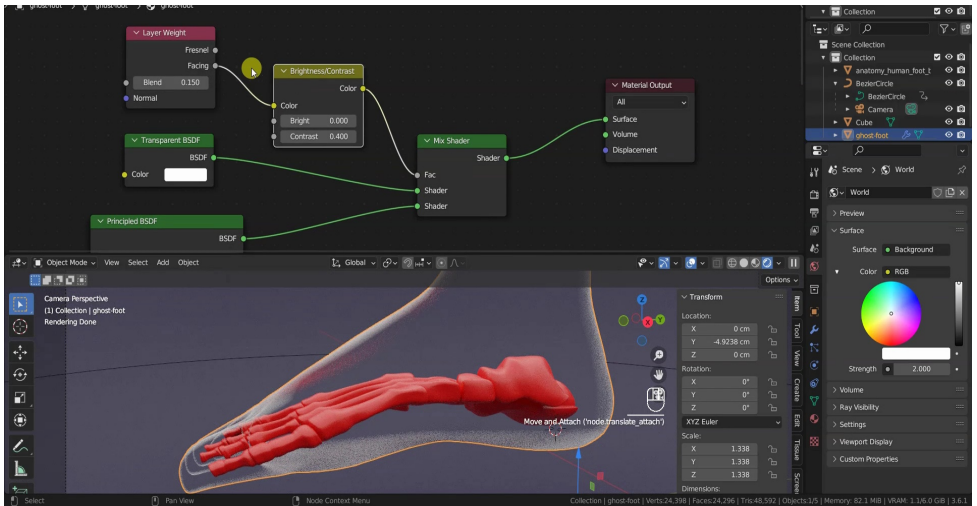


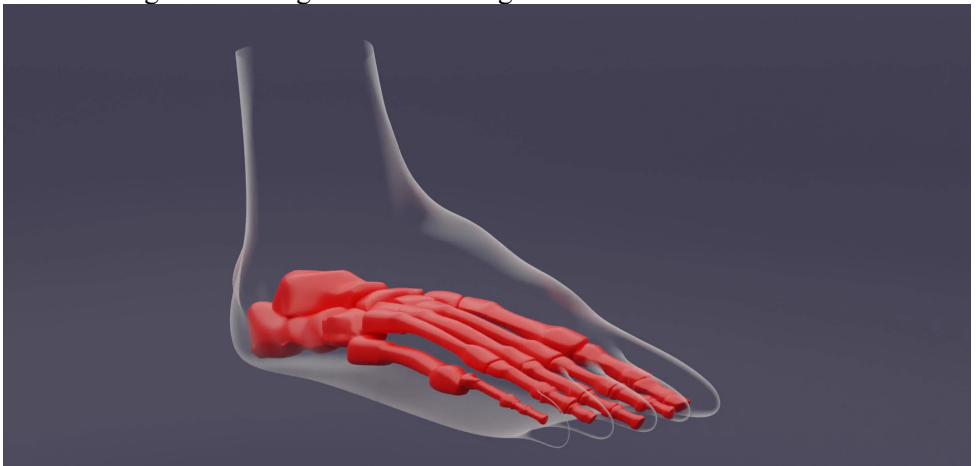
This PDF is available for free; feel free to share it!

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



Hello everyone!

In this short tutorial, we'll see how to make the faces of an object **transparent** based on their **orientation relative to the camera**, achieving the effect visible in the rendering I'm showing on the screen right now.

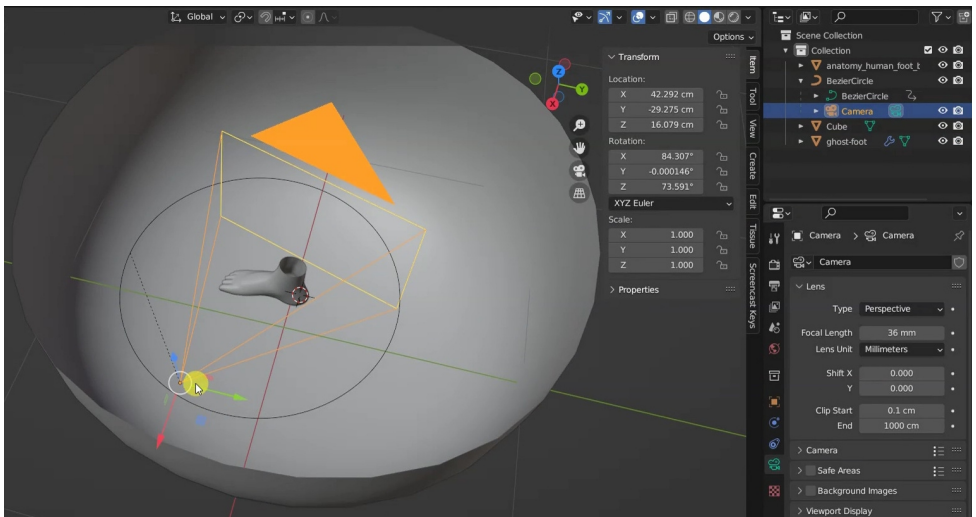


This is therefore an effect that **highlights only the silhouette** of an object, allowing its shape to be glimpsed while also showing other objects inside it.

This is a basic tutorial and was made using **Blender 3.6**, specifically with the **Cycles** rendering engine; however, the nodes used have been available in several versions of the software and, presumably, will continue to be available in the future.

In the scene I am using for this example, there are two objects placed inside a container, providing a backdrop to the objects.

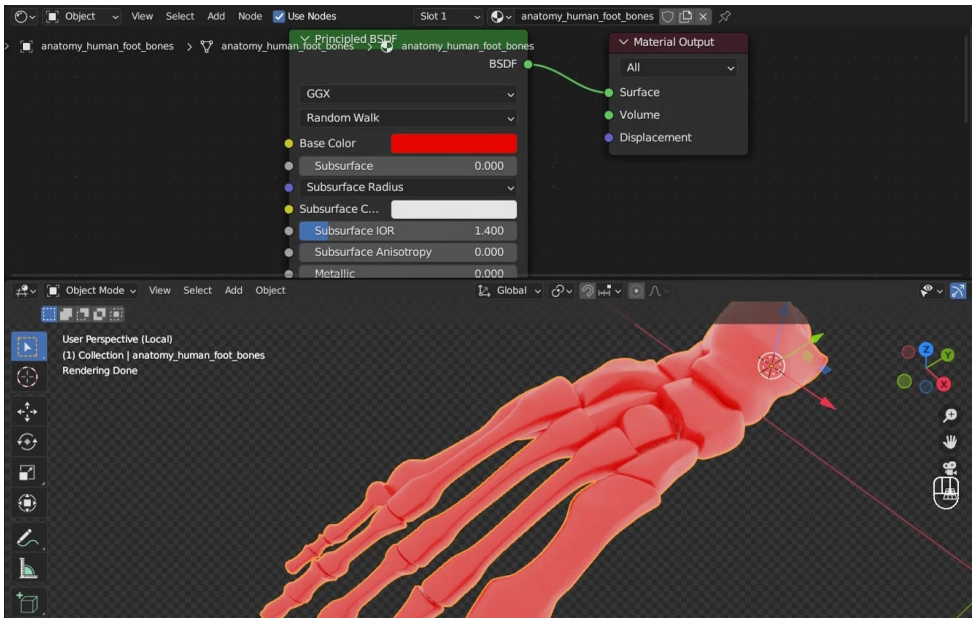
There's also a camera and a Bezier circle that I used to give movement to the camera, which rotates around the objects.



The lighting is entirely provided by World Background.

The two objects used for the example are a foot and its bones, with the latter grouped into a single object.

For the **bones**, I provided a **very simple material**, made up of just the **Principled** node with a red color, **Emission** intensity set to 1, and a very **dark red** shade for the Emission Color field.



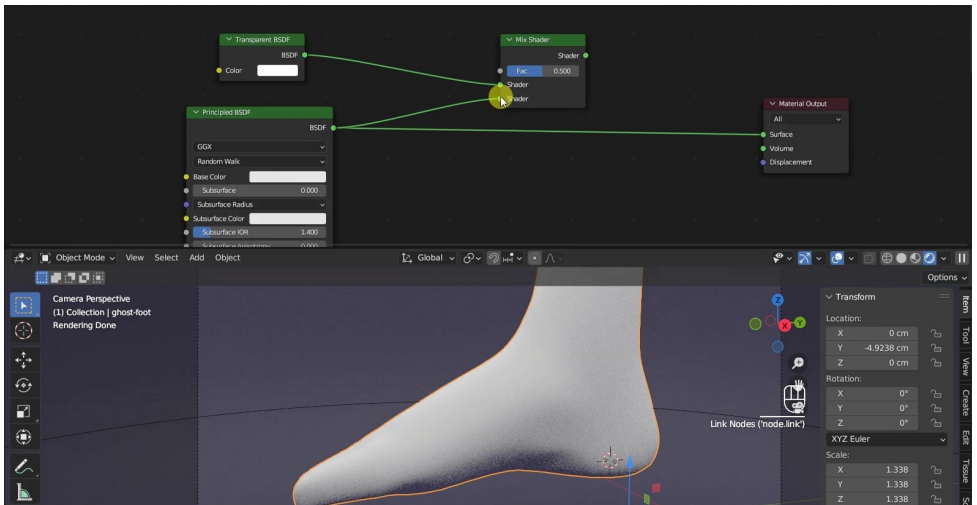
However, the heart of the tutorial is, of course, **the material of the foot**, which encloses the bones.

We start, in this case as well, with a simple **Principled** node, giving it the color we want to provide to the silhouette, for example, pure white.

To achieve the transparent part, we obviously need a **Transparent** node, again in pure white, so we add this node to the scheme.

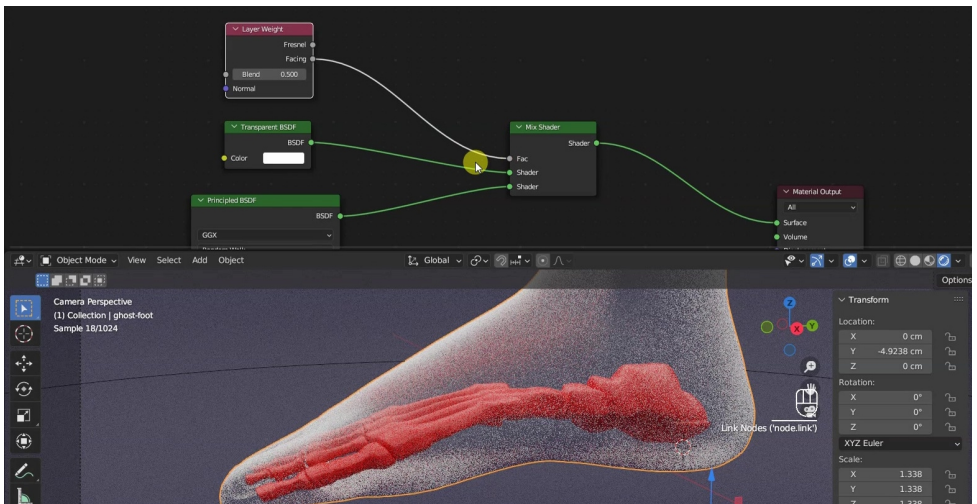
To perform the mix, we add a **Mix Shader** node, making sure to connect the **Transparent** node to the **first** Image input of the node.

We then connect **Principled** to the **second** Image input of the Mix node.

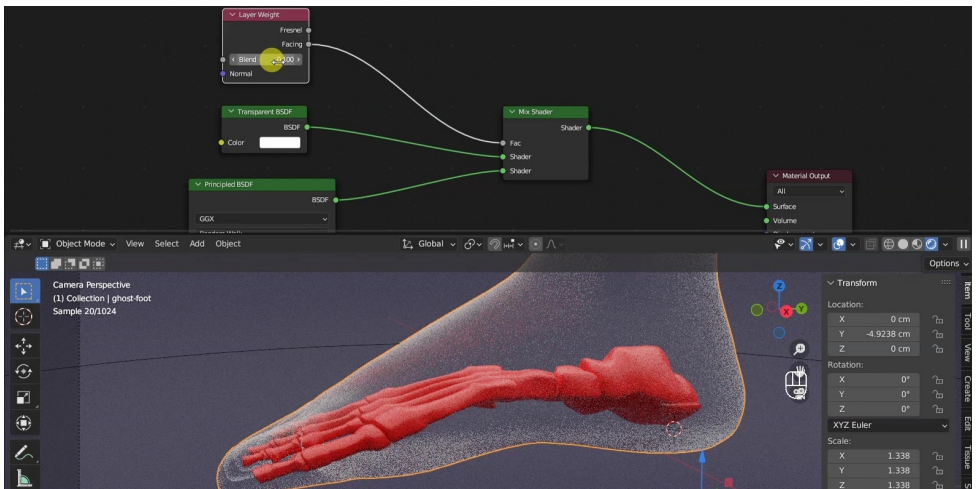


In this way, when the value used for **Factor** is equal to **0**, **Transparent** will be used; on the other hand, when that value is equal to **1**, **Principled** will be used.

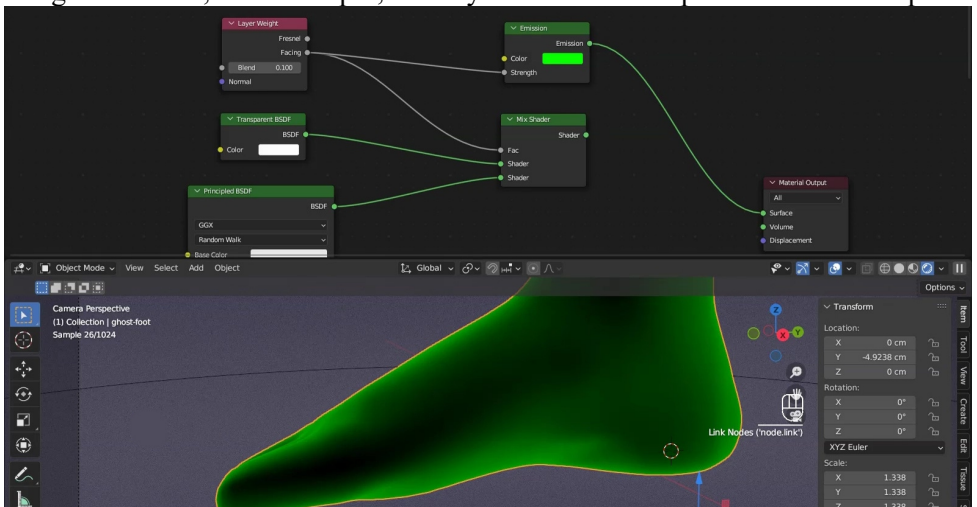
The node that can provide us these values in the desired manner (that is: 0 when a face points towards the camera and subsequently other values for other orientations), is the **Input Layer Weight** node, which we can then add to our scheme. We are particularly interested in the **Facing** output of this node, which we can connect to the **Factor** input of the **Mix Shader** node.



Once the connection is made, we can **adjust the Blend parameter** value of the **Layer Weight** node, which slightly shifts the output value, so it's a good idea to use a low value, such as 0.1; however, this solution provides **too gradual a transition** between the various faces, as can be seen when observing a rendering preview in the 3D Viewport.

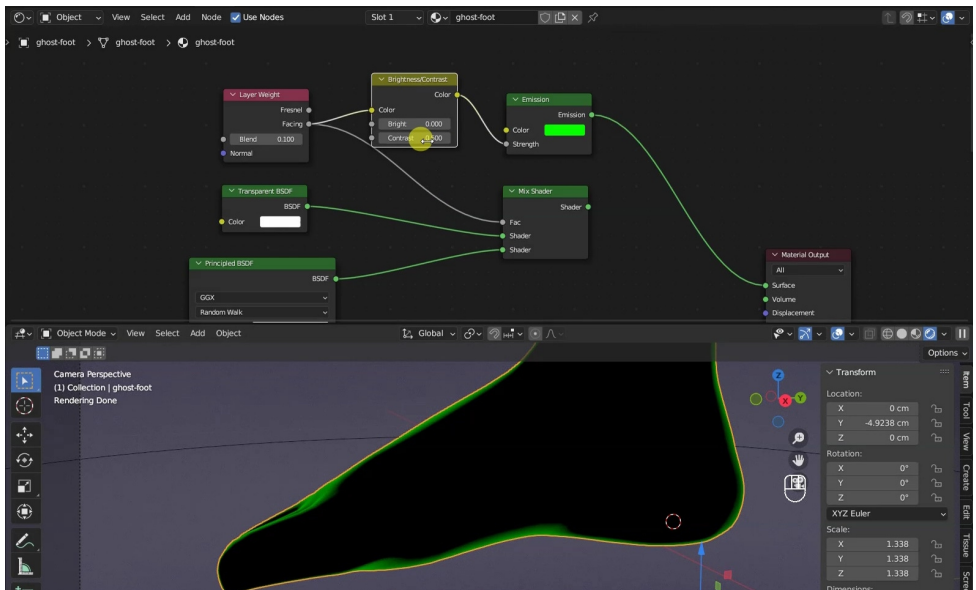


To better observe both the problem and the solution, I add an **Emission Shader** node with a green color, connecting to it the Facing value input from the Layer Weight node and and, for the output, directly to the Surface input of Material Output.



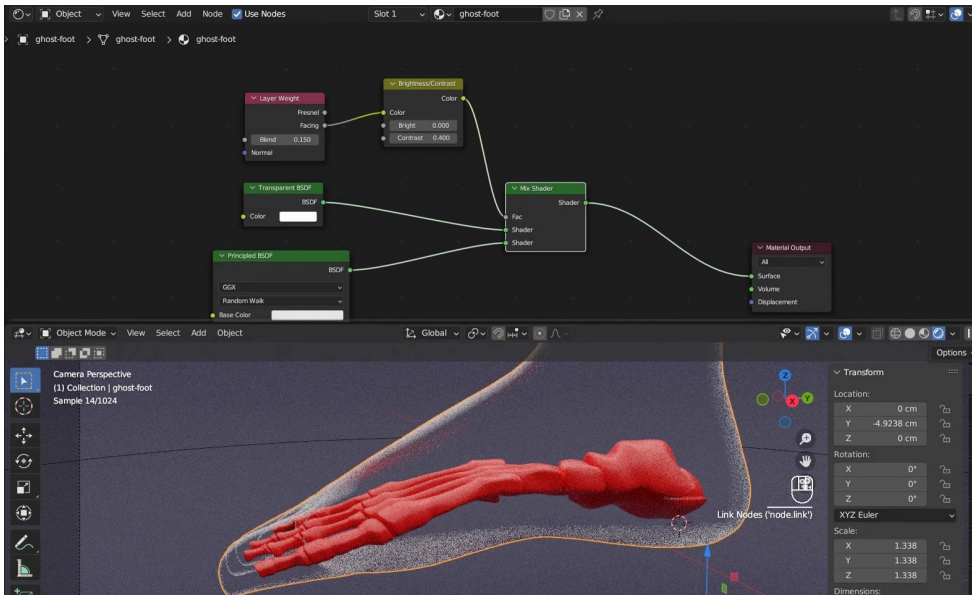
We would like a less gradual, more distinct separation between the green and black areas.

An operation of this type is performed by the **contrast** control, so we can add a **Color Bright Contrast** node right between the Layer Weight and Emission nodes, and then adjust the Contrast field value as desired until achieving a satisfactory result.



Once a good result is achieved, remove the Emission node and connect the Color output from Bright Contrast to the Factor input of Mix Shader.

Finally, connect the Shader output from Mix Shader to the Surface input of the Material Output node, make any other necessary adjustments, and then proceed with rendering!



I hope this short tutorial was helpful to you! See you soon!