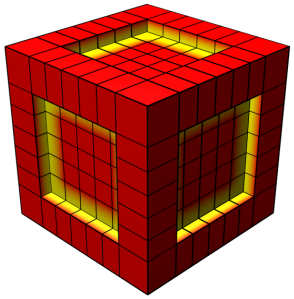
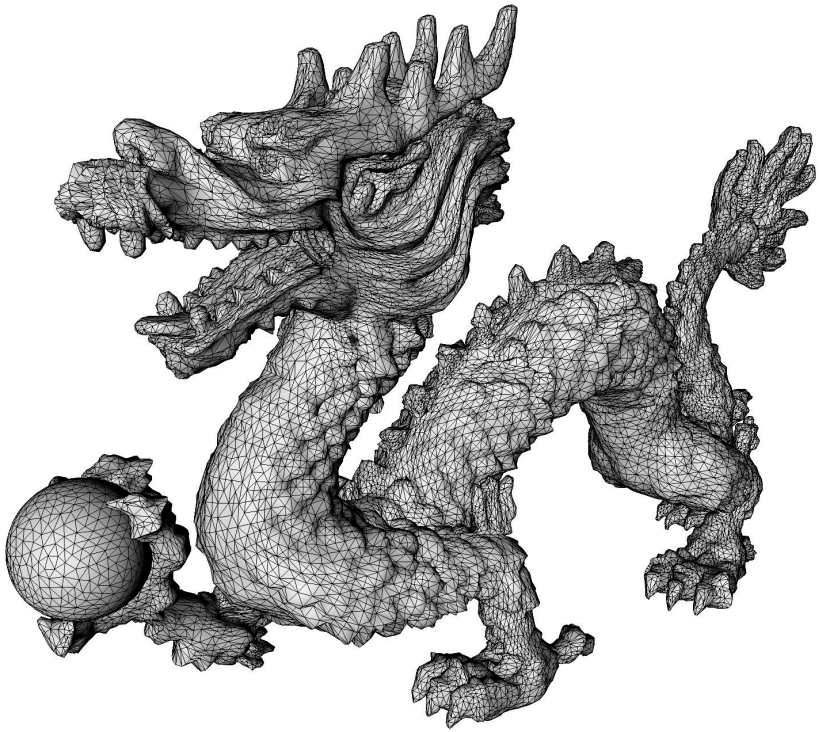
VertexAO



VertexAO is a tag that will create vertex maps with Ambient Occlusion/Proximity values, in order to be able to speedup render or create special effects base on geometry indentations, bumps, crevices, nooks and crannies.

This tag, of course, will only work on polygonal editable objects since Vertex Maps are only valid for those types of objects.

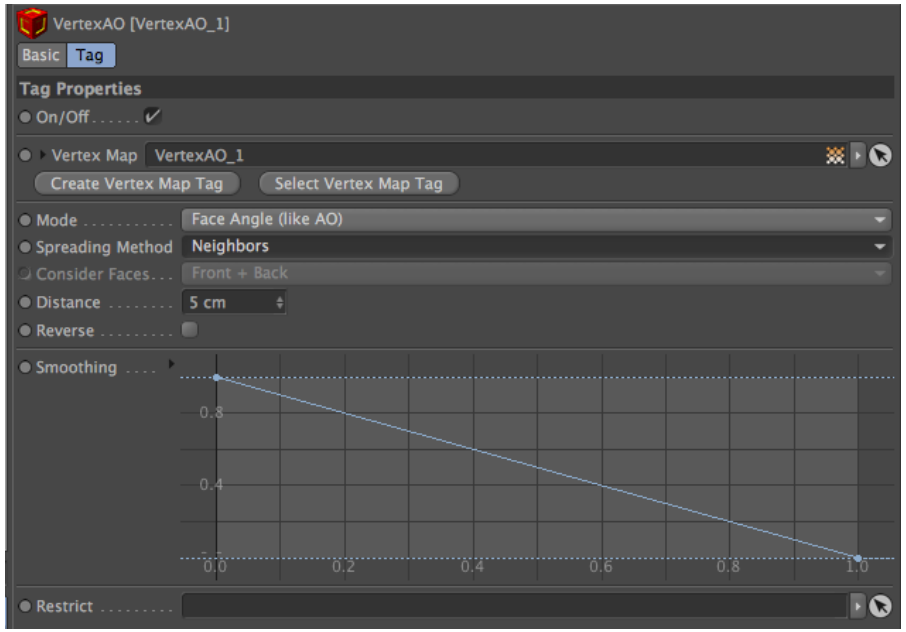
Throughout this manual, the sample mesh that will be used with be this chinese dragon:



If a mesh with many, many polygons is used, expect some extra calculation time.

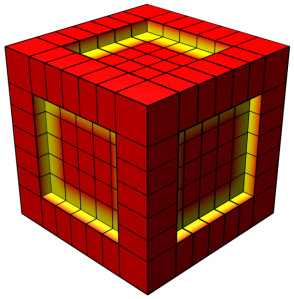
Calculations are executed inside a parallel thread, so every calculation can be interrupted by pressing Esc or changing a parameter. A progress bar is displayed at the status bar when **VertexAO** is performing calculations.
Since all calculations are very complex, if the user interrupts them, sometimes **VertexAO** may not realize that it should restart calculations.
If that occurs, simply turn **VertexAO** off and on again.

Simply add a **VertexAO** tag to your object and you will get this set of parameters in the Attribute Manager.



The **On/Off** switch is useful in two situations:

- Turning off and on will force a new calculation if, for some reason, it got interrupted.
- Once a Vertex Map is set and the result is as expected, turning VertexAO off will speedup things, as new calculations will be be performed.



Inside the **Vertex Map** field should be placed a Vertex Map tag where the result of the calculations will be stored.
If the object still has no Vertex Map tags, a new one can be created by clicking the **Create Vertex Map Tag** button.
To see the result of the calculations applied to the Vertex Map tag, the Vertex Map tag must be selected.
The problem with this is that, if the Vertex Map tag is selected, the Attribute Manager shows the parameters of the Vertex Map tag, instead of the parameters of the **VertexAO** tag.
To solve this, do the following:

- Select the **VertexAO** tag.
- Click the **Lock** icon in the Attribute Manager. 
- Click the Vertex Map tag or click the **Select Vertex Map Tag** button in the **VertexAO** tag.

VertexAO has two calculation modes. The **Face Angle (like AO)** takes into consideration the angles between faces. By default, it will turn on the Vertex Map on vertexes that are part of edges that are shared by faces that have an angle of less than 180° between them.

The **Facing Proximity (like SSS)** mode takes into account the distance between the centres of the faces and the direction of their normals.
This will be explained better later on.

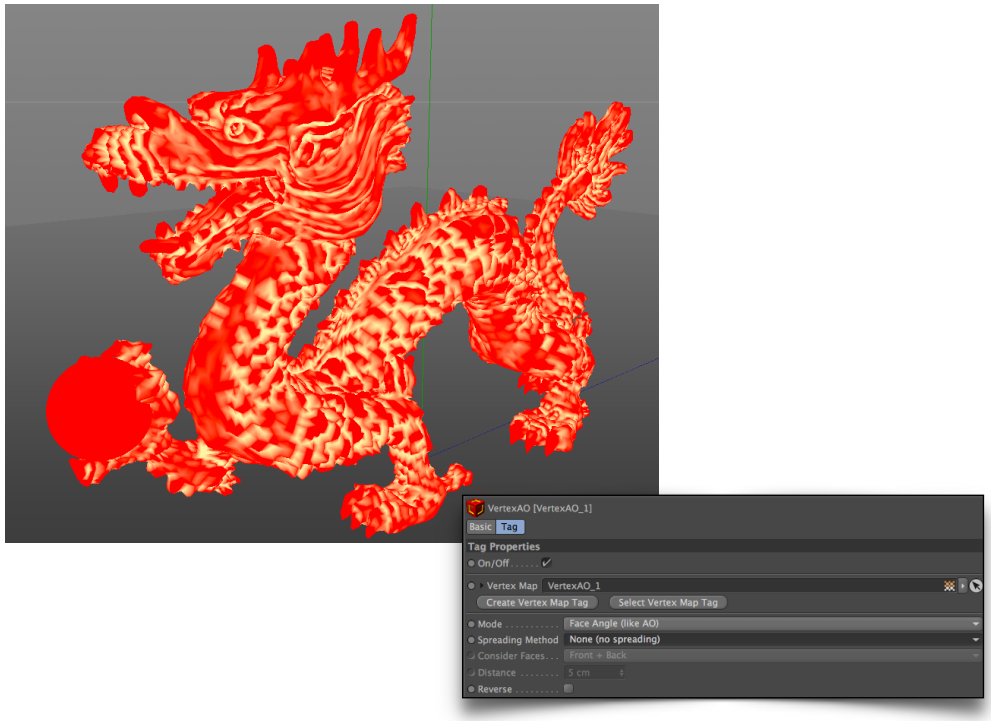
When the mode is set to **Face Angle (like AO)**, it is possible to select a **Spreading Method**. This will increase the range of “darkening” after the initial evaluation of angles is calculated and, of course, increase a little bit the time that it takes to calculate the values of the Vertex Map.

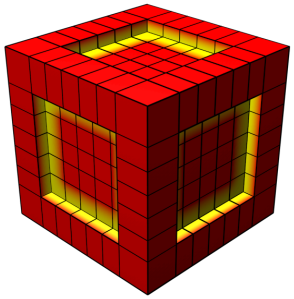
Spreading Method can be one of three options:

Neighbours - After a vertex gets “darkened”, the neighbouring vertexes, within a specific radius are ”darkened” also. The radius is set by the **Distance** parameter.

Global - After a vertex gets “darkened”, all the vertexes, within a specific radius are ”darkened” also. The radius is set by the **Distance** parameter.

None (No Spreading) - Only the vertexes belonging to edges that are common to faces that have less than 180° between them are “darkened”.

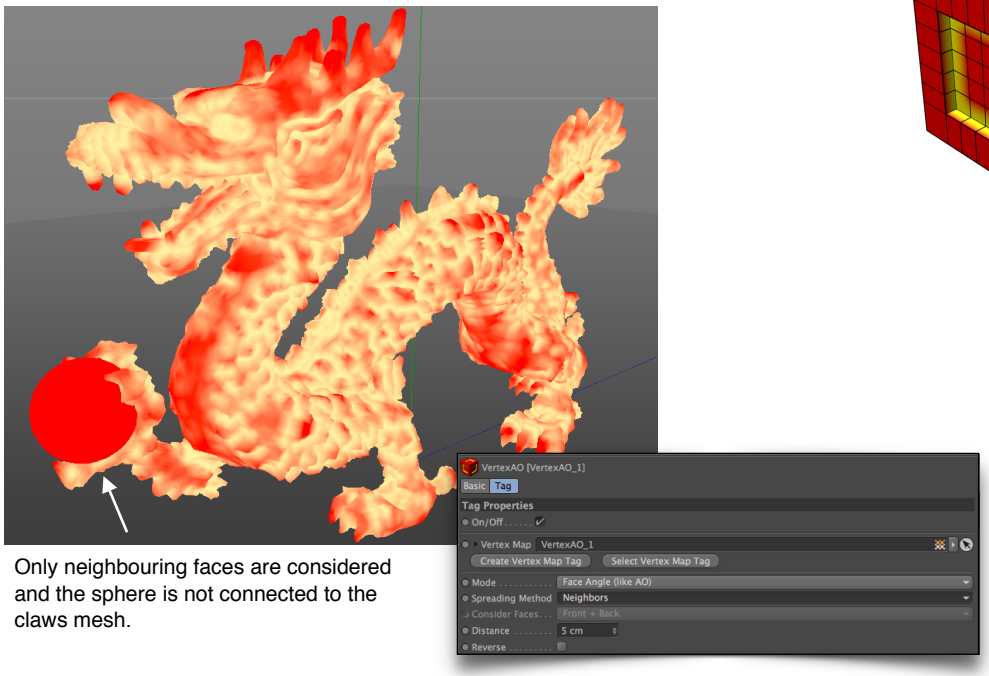




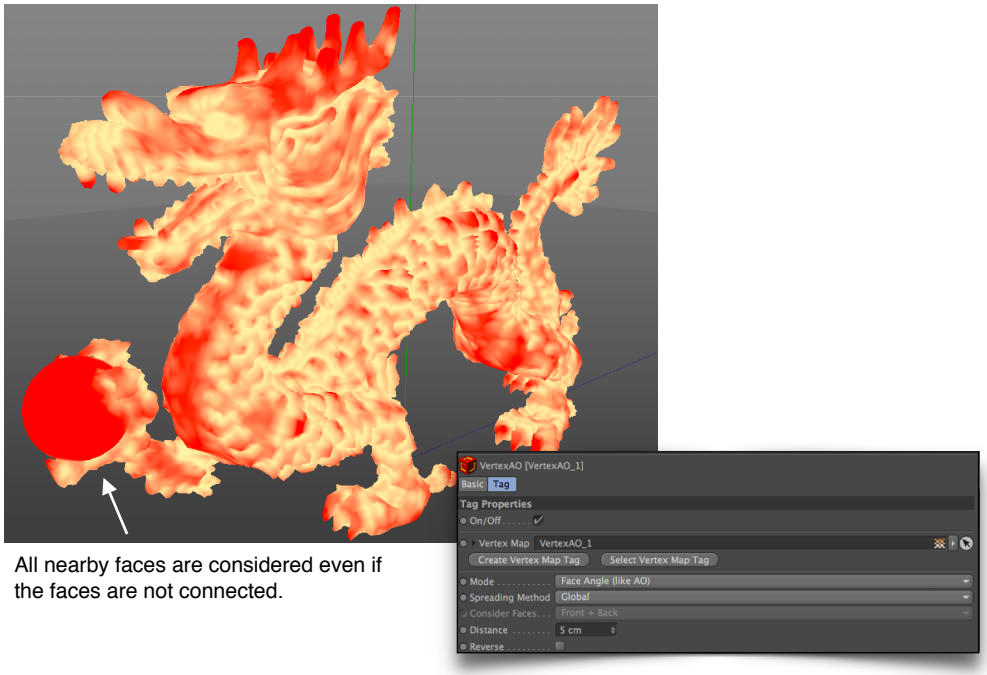
If the **Mode** is set to **Face Angle (Like AO)**, it is also possible to revert the way the angle is considered. If the **Revert** option is on, instead of consider only the angles that are less than 180°, it will only consider angles that are more than 180°.

Also available if the **Mode** is set to **Face Angle (Like AO)** is a **Smoothing** spline. Adjusting this spline will allow for the adjustment of the Brightness/Contrast of the gradient of the spreading. The default is a linear ramp from full 100% to 0%.

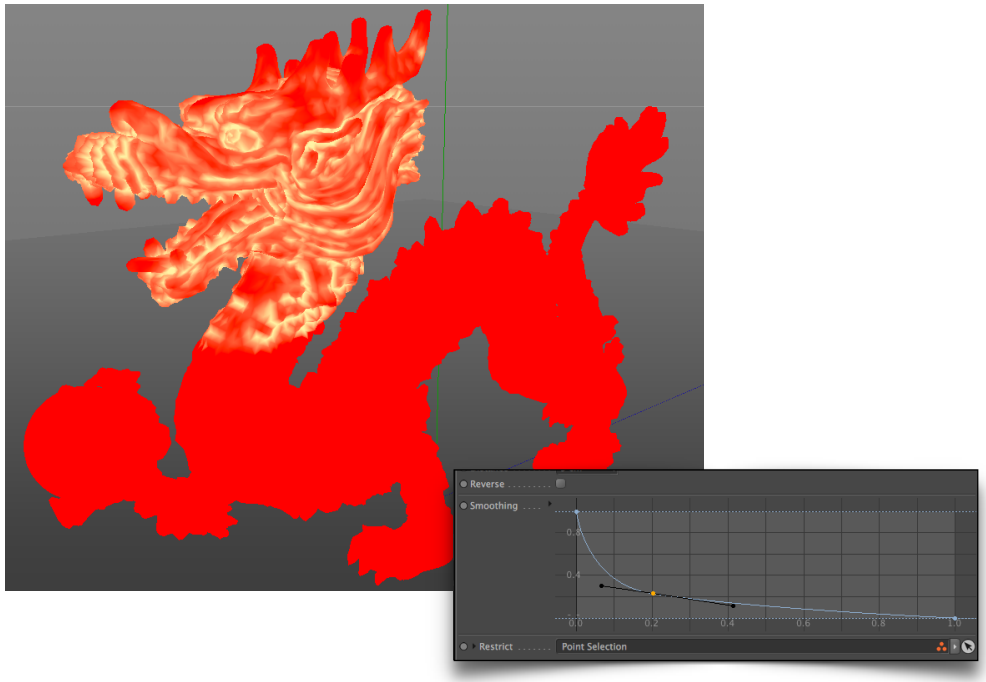
Finally, the **Restrict** field allows a Point Selection tag to be dragged in. If it contains a Point Selection tag, only the points defined in that Point Selection tag will be considered, speeding up the calculation considerably.



Only neighbouring faces are considered and the sphere is not connected to the claws mesh.



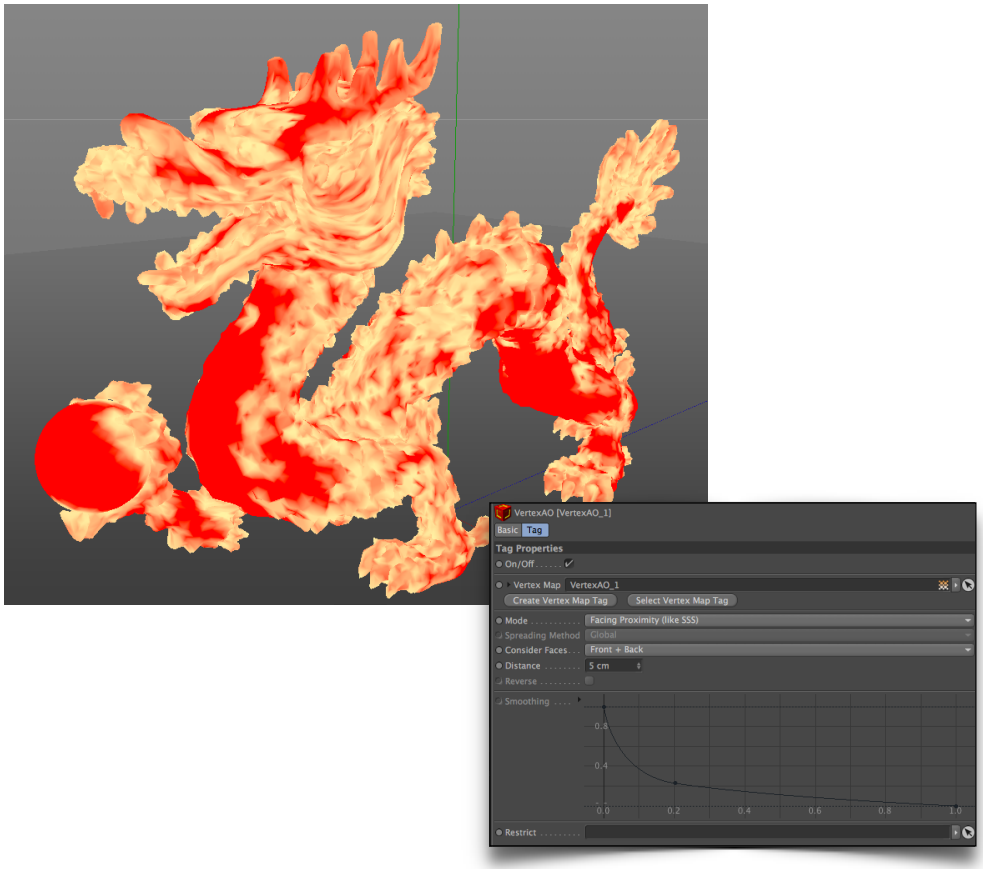
All nearby faces are considered even if the faces are not connected.



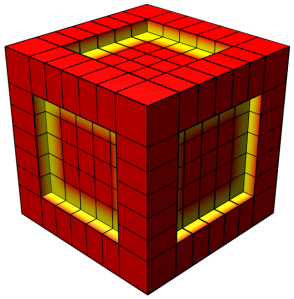
Also, using a Point Selection tag, several locations of the mesh can be calculated using different options. For that, several **VertexAO** tags can be used and the result combined with a **VertexMapMix** tag.

When the mode is set to **Facing Proximity (like SSS)**, it is not possible to select a **Spreading Method**. However, the **Distance** parameter is still considered.

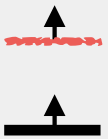
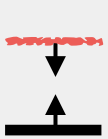
Reverse and the **Smoothing** spline are also turned off.



The results of the **Facing Proximity (Like SSS)** method are coarser than the ones from **Face Angle (like AO)**.
It is advisable to use a **VertexMapMix** tag to soften then result, if required.



Using this method (**Facing Proximity (Like SSS)**), will make available a new set of options to define how faces are considered for inclusion in the calculation (**Consider Faces**).

Front	Back	Front+Back	Same Direction +	Same Direction -
				

Usually, the best method is the **Front+Back** (it is the default).
But, for some particular shapes, experimentation with the other method could show some nice results. Test it out.

For better results, it is advisable that all normals are coherent and aligned as expected (facing outwards).