



JR Macros: Blender AgX Transforms

# Introduction

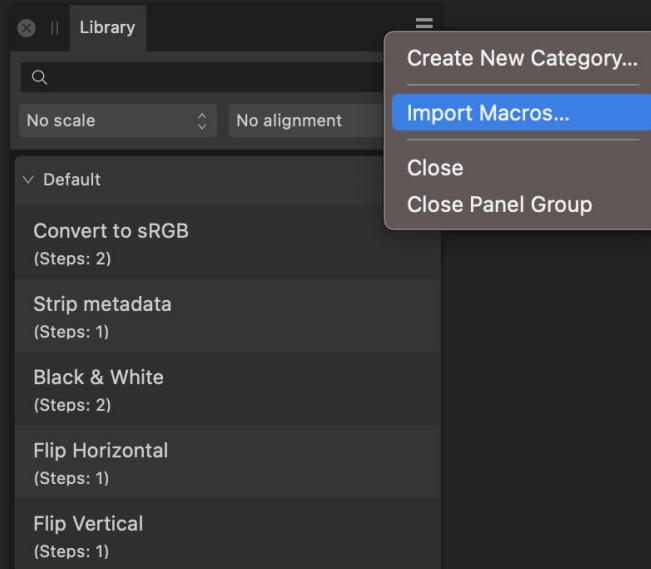
This macro pack is designed to emulate Troy Sobotka's *AgX* view transform, which provides an organic-looking tonal compression that makes renders instantly usable when saved out of Blender as non-linear gamma encoded images.

If you wish to work with OpenEXR or Radiance HDR exports, you may find it difficult to replicate the *AgX* transform because Blender writes out linear unbounded float values to these document formats.

Affinity Photo supports OpenColorIO, so you can of course load the *AgX* configuration into the app and use its view transforms. This does not write the transforms into the document colour values, however, so it cannot be used to export to a non-linear format such as 8-bit JPEG, 16-bit TIFF etc.

There are workarounds to “bake” the *AgX* transform into the colour values, but it can be a complex procedure. It also requires having the *AgX* OpenColorIO configuration installed within Affinity Photo.

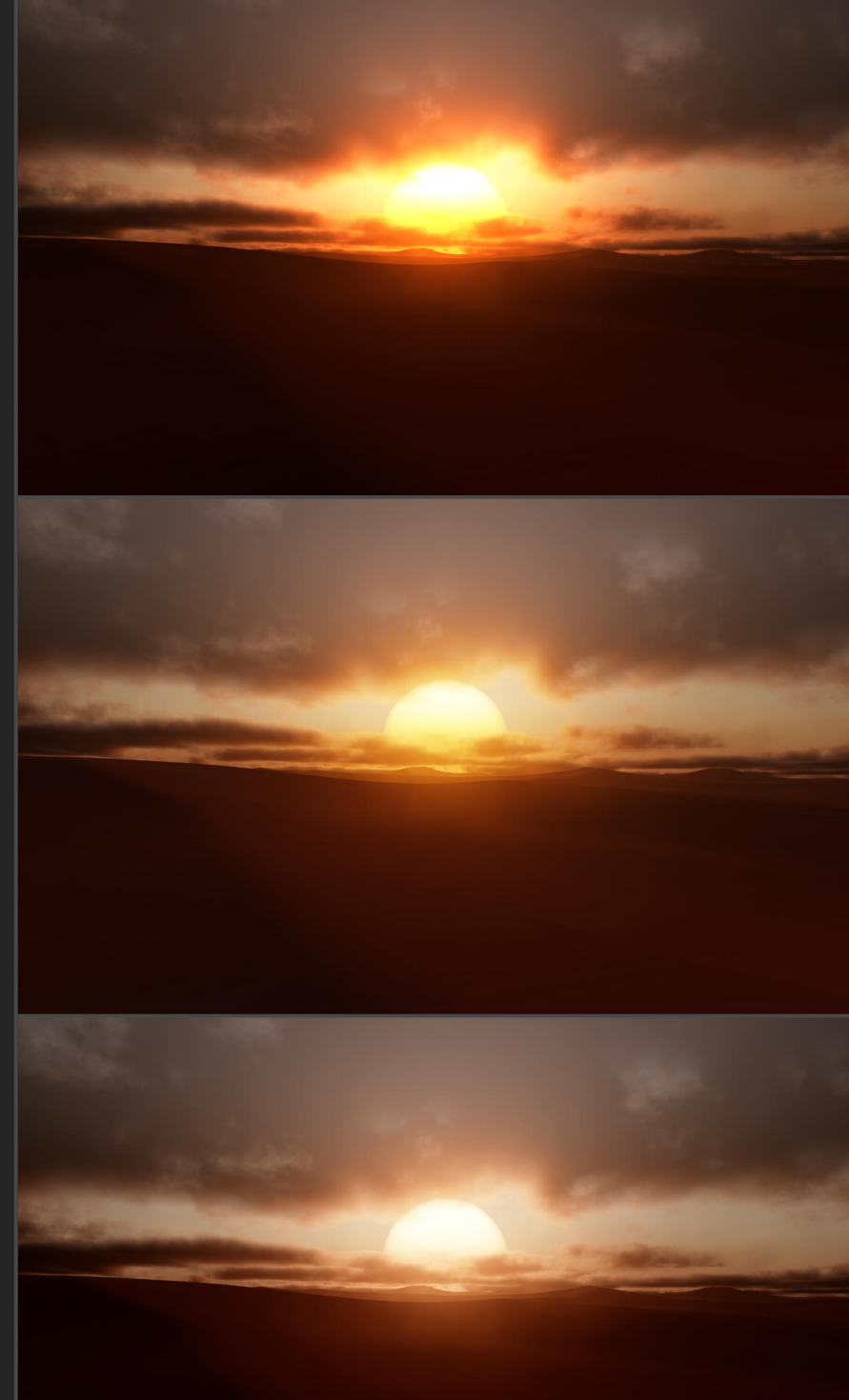
Instead, these macros offer a single-click solution to get exactly the look you want, with *no dependency on OpenColorIO* at all. Even with a fresh install of Affinity Photo, just open your EXR/HDR render, find the macro library, then click to add the *AgX* transform.



# Installation

1. Extract the *.afmacros* file to a directory of your choice.
2. In Affinity Photo, you will need to expose the **Library** panel. In V1, go to **View>Studio>Library**, or in V2 go to **Window>Library**.
3. Click the small icon at the top right of the **Library** panel and choose **Import Macros**.
4. Navigate to the directory containing the *.afmacros* file and select it, then click **Open** (or double click the file).
5. The **Library** panel will then be populated with the macros from that category. If you are installing any other macro packs, repeat the process for those categories.

Tip: you can also drag-drop the *.afmacros* file onto a blank area of the app and it will immediately import and be shown on the Library panel. You can bulk import multiple *.afmacros* files this way.



Above: linear render with no transform. Middle: Blender Filmic Medium look applied. Below: *AgX* Base look applied.



# Further Instructions

**Please note:** when applying a transform macro for the first time after starting Affinity Photo, you may experience a delay before the result appears. This is normal behaviour, as a 1D shaper LUT is being loaded.

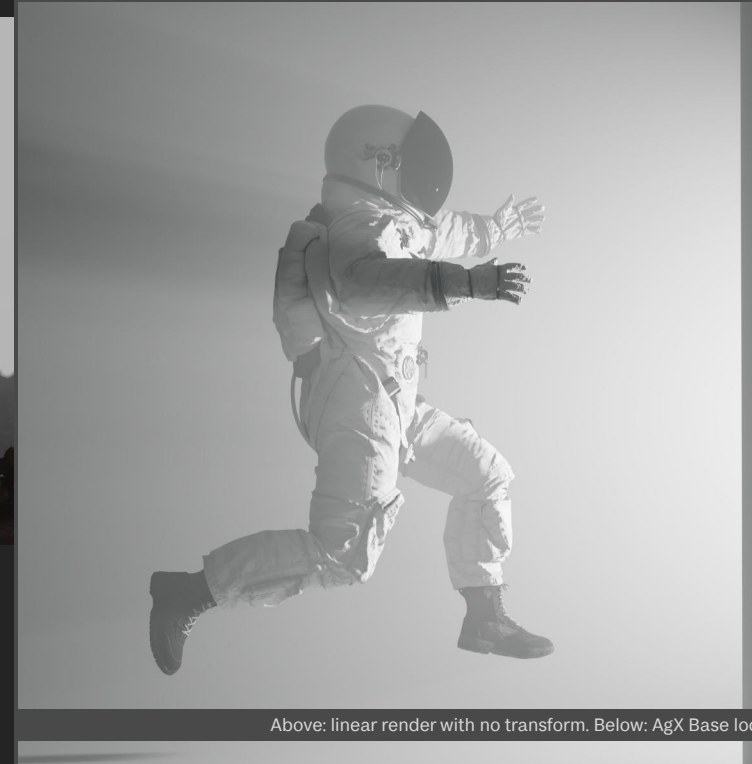
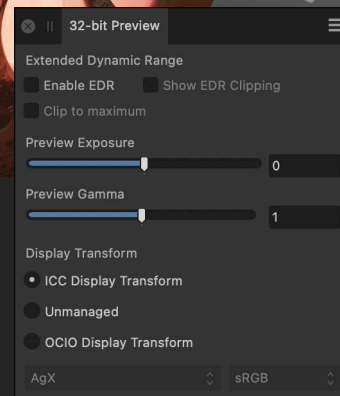
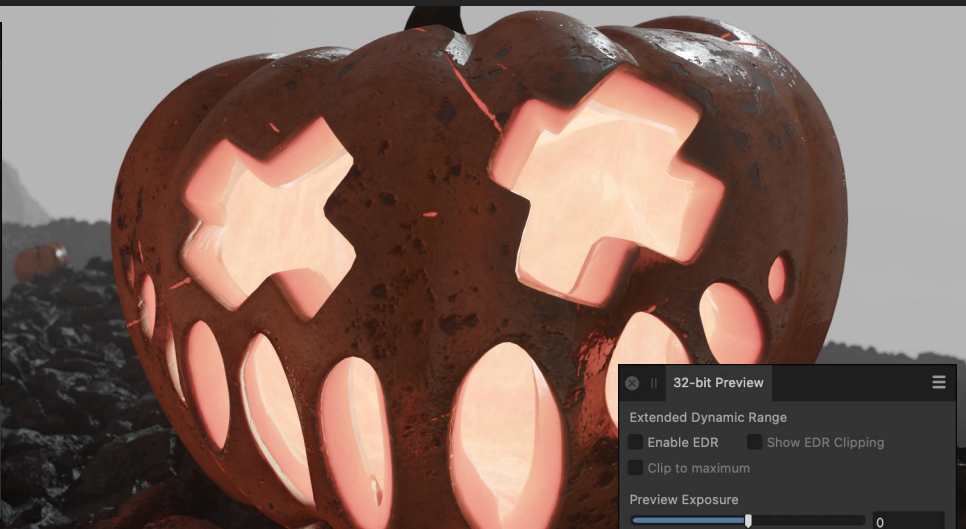
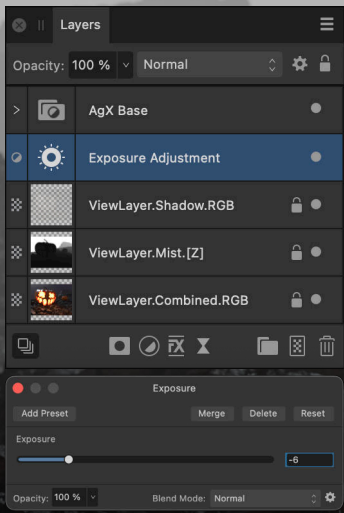
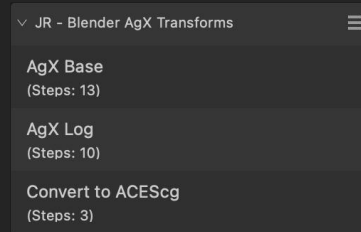
When applying these macros, the order of the layer stack is very important.

See the UI screenshot for guidance: you will want to add your AgX transform *above* the render layers but *beneath* any additional adjustments, filters or other layer work you might be applying.

Also, you may need to adjust the overall exposure of your render if you have changed Blender's Exposure slider—some plugins, such as **True Sky** and **Physical Starlight** and **Atmosphere** may do this automatically. The exposure may be reduced by up to -6 stops, but this change is not reflected in the linear values of an EXR or HDR document when saved.

To emulate this, simply add an **Exposure** adjustment (Layer>New Adjustment Layer>Exposure) and place it *above* the render layers but *beneath* the AgX transform group. On the Exposure adjustment dialog, change the slider to match the value used in Blender (e.g. -6 or -3).

If you prefer to do your own contrast grading, there is an AgX Log macro which only applies the log shaping and final gamma transform.



Finally, be aware that if you already have OpenColorIO configured within Affinity Photo, it will use the **OCIO Display Transform** method by default when you load an OpenEXR document. This must be changed back to ICC Display Transform to ensure consistency with the result you will get when exporting to 8/16-bit formats. You can access this option on the **32-bit Preview Panel** (View>Studio>32-bit Preview).

Above: linear render with no transform. Below: AgX Base look applied.

# Credits

All 3D renders and editing by James Ritson (<http://www.jamesritson.co.uk>)

Implementation for Affinity Photo derived from Troy Sobotka's AgX OpenColorIO configuration (<https://github.com/sobotka/AgX>).

