

Imager Denoiser Oidn

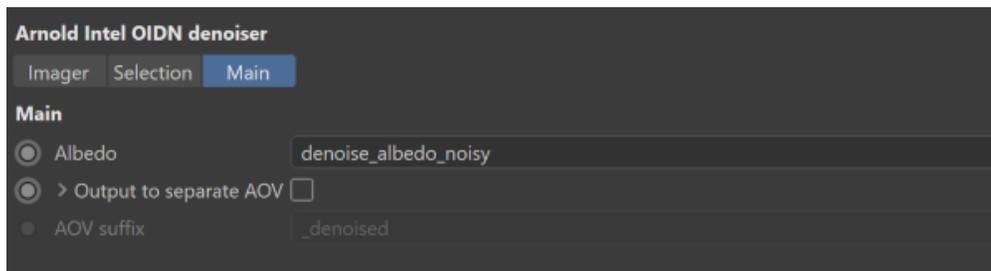


Left: without denoising. Right: using `imager_denoiser_oidn`.

The *Oidn* denoiser (based on Intel's [Open Image Denoise](#) technology) is available as a post-processing effect. It is integrated into Arnold for use with IPR as an *imager* (so that you get a very quickly denoised image as you're moving the camera and making other adjustments).



- Arnold will automatically force a `box_filter` on outputs with filters with known issues when either of the denoisers are used. This is because the *Oidn denoiser* does not perform well with filters that span across multiple pixels.
- Denoisers won't work as well after *imagers* have been applied to them. Therefore denoisers should be applied **before** any other post-processing *imagers* (top of imagers list).
- The *Oidn denoiser* only works on full frames rather than buckets.



Albedo

Optional albedo feature AOV is used to improve denoising quality (`denoise_albedo_noisy` is the default). Setting to an empty string "" will disable the use of feature AOVs, which might give better results when denoising AOVs for which no albedo exists.



denoise_albedo_noisy (default)



Without denoise_albedo_noisy

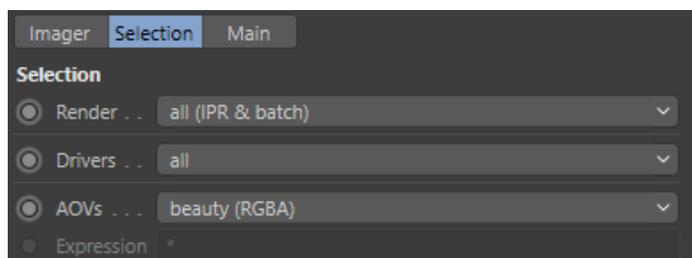
Output to separate AOV

When set, creates a new additional output with the input name plus the suffix in which the denoised result will be written into. For example, when denoising the "RGBA" AOV with the *output_suffix* set to "_denoised", the denoised result will be written into a new "RGBA_denoised" output. It only works with multi-layer drivers.



Note that this feature is available only in multi-layer Arnold drivers (e.g. EXR) and when rendering to the Picture Viewer, but not supported by single-layer Arnold drivers (e.g. jpeg, png, etc).

Selection



Render

Specifies if an imager is effective only in the IPR, only in batch (e.g. Render to Picture Viewer, Commandline render, Team Render) or both. For example, you can add imagers only to the IPR for look development, but turned off in the final render. Or vice versa, an imager may be slow or not required in the IPR, but turned on in the final render.

Drivers

Specifies if an imager is assigned to all drivers or just the selected ones.

AOVs

Specifies which AOVs are affected by the imager.

- **all**: All AOVs are affected.
- **beauty**: Imager is visible only in the beauty.
- **expression**: Imager is applied to AOVs selected by this expression. For example, to select the `diffuse_direct` and `diffuse_indirect` AOVs, write `diffuse_direct or diffuse_indirect`. Or write `diffuse_*` to select all AOVs start with 'diffuse_' prefix.

Expression

Selects layers processed by this imager with a glob or operator expression. An imager can use a selection to determine what nodes will be selected to be processed by the imager at render time. The selection is an expression that can consist of multiple selections concatenated by:

- `or` (union)
- `and` (intersection)

- **not** (negation)
- **and not** (exclusion)
- **()** for nested scoping

Each selection uses [glob patterns](#) and [regular expressions](#) and a node will be processed by the imager if the expression matches the node name. By default, glob matching is used unless the selection is in a regex quote, i.e. `r'<my_regex>'`. For example:

```
specular or diffuse  
  
not r'sss_(direct|indirect)'  
  
r'color_(mask1|mask2)' or r'mask[34]'
```