



NUKE SURVIVAL TOOLKIT

TONY LYONS | 2020

NUKE SURVIVAL TOOLKIT DOCUMENTATION

Release v1.1.0

Tony Lyons | 2020

About

The Nuke Survival Toolkit is a portable tool menu for the Foundry's Nuke with a hand-picked selection of nuke gizmos collected from all over the web, organized into 1 easy-to-install toolbar.

Installation

Here's how to install and use the Nuke Survival Toolkit:

- 1.) Download the .zip folder from the Nuke Survival Toolkit github website.

https://github.com/CreativeLyons/NukeSurvivalToolkit_publicRelease/releases/latest

This github will have all of the up to date changes, bug fixes, tweaks, additions, etc. So feel free to watch or star the github, and check back regularly if you'd like to stay up to date.

- 2.) Copy or move the NukeSurvivalToolkit Folder either in your User/.nuke/ folder for personal use, or for use in a pipeline or to share with multiple artists, place the folder in any shared and accessible network folder.

- 3.) Open your init.py file in your /.nuke/ folder into any text editor (or create a new init.py in your User/.nuke/ directory if one doesn't already exist)

- 4.) Copy the following code into your init.py file:

```
nuke.pluginAddPath("Your/NukeSurvivalToolkit/FolderPath/Here")
```

- 5.) Copy the filepath location of where you placed your NukeSurvivalToolkit. Replace the Your/NukeSurvivalToolkit/FolderPath/Here text with the NukeSurvivalToolkit filepath location, making sure to keep quotation marks around the filepath.

- 6.) Save your init.py file, and restart your Nuke session



- 7.) That's it! Congrats, you will now see a little red multi-tool in your nuke toolbar.

Technical Details

There are a few things about this menu that try and make it both easy and safe to use.

- 1.) In the main folder there is an init.py file that is used to recursively add any folder downstream of the main folder to the nuke plugin path list. This is very useful for folder organization or for quickly adding a “pre-built” toolkit/menu by another artist.

You can ignore any folder by adding an “_” underscore in front of the folder name and the init.py will skip over it while loading. This can be useful for quickly debugging if an error is coming from a specific folder. There is also a “whitelist” and a “blacklist” in the init.py and you may add any folder to that list for it to skip or load.

- 2.) The menu.py in the main folder is primarily building almost the entire toolkit menu. You will find it organized into sections: Draw, Time, Color, Filter, etc. The tools will show up in the order that you designate them in this menu.

Because the init.py is loading downstream folders, any menu.py files located in another folder, will be loaded along with the main menu.py. This is happening with “Expression Nodes”, “Hagbarth Tools” and “Xavier Martin’s X_Tools” toolset menus. It was sometimes easier to group these tools together by artist and add them to Draw/Expression AG, Draw/Hagbarth Tools, or Filter/X_Tools XM. These submenus are being added by the menu.py files in their artist’s respective folder.

- 3.) Nuke does not like to load multiple gizmo files with the same name. Because the Nuke Survival Toolkit may be added into company pipelines that already have many gizmo’s being loaded in, I have given all .gizmo files their own prefix “NST_”. This means all files should have a unique name to any file that would be already installed. For example, if there was an `iBlur.gizmo` installed, the one in Nuke Survival Toolkit is named `NST_iBlur.gizmo`, so there should be no conflicts. In the main menu.py at the top, there is a variable that you can replace if you choose to find/replace the “NST_” prefix to a custom one for all the gizmos. You could do this with a renaming software or via the terminal for all gizmos with the “NST_” prefix. If you change “NST_” to “WOW_” for example, just enter “WOW_” in this variable. This might help if two different Nuke Survival Toolkits are being loaded at once, to keep them unique.

- 4.) All gizmo's are stored as .gizmo files on the folder system, but are all actually loaded into nuke as Groups, with no link back to the gizmo filepath. This is a strange bug / feature / work around that sort of tricks nuke into thinking you have loaded a gizmo, but actually have loaded a group. There are a few advantages to this method:
 - a.) Nuke will automatically open the properties panel of the tool, unlike if you `nuke.nodePaste()` a .nk file
 - b.) Nuke actually stores the defaults of the gizmo in memory, during that specific nuke session. This means you will be able to ctrl + click on knobs and reset them to their intended default settings. This unfortunately goes away once you close and re-open the script, as nuke will just consider the nodes a normal group and will not know what the defaults are.
 - c.) Groups are generally easier to debug and enter inside to see what is going on.
 - d.) This will help with render farms or other users opening scripts that would normally be sourcing the gizmos from wherever you have placed the Nuke Survival Toolkit. Sometimes render farms or other users cannot access your local directories, which might cause errors when other artists or render farms are trying to open the script, since they may not be loading the NukeSurvivalToolkit. Making sure the tools are Groups will mean the tools exist in your nuke script and will never be unlinked/unsourced when someone else is opening the nuke script.

If you prefer to use gizmos instead of groups, you simply have to open the gizmo in a text editor and change where it says "Group" at the top of each .gizmo file, and replace it with "Gizmo". It is case sensitive, so make sure you capitalize `Gizmo` or `Group`.

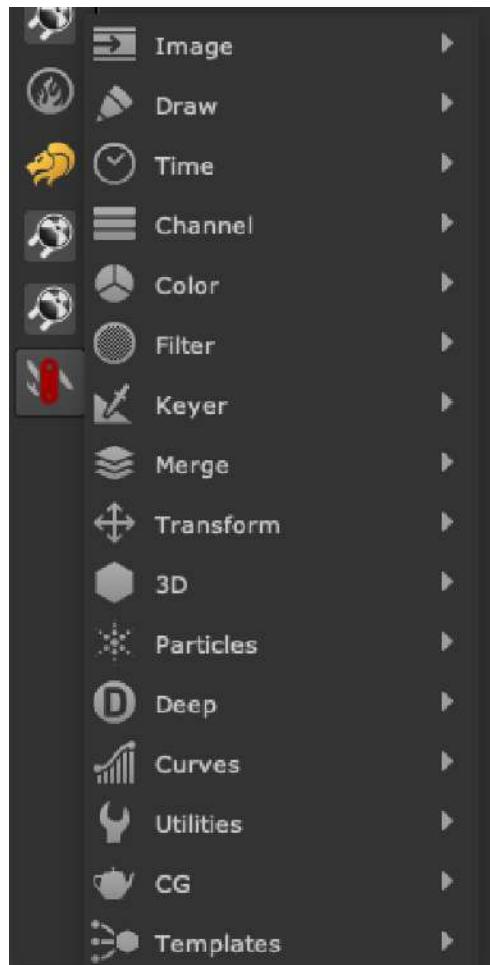
- 5.) Removed all x and y node graph positions from the gizmos, (xpos and ypos). If you leave these in; when you have a node selected and create a gizmo, instead of spawning under the node, it can fly to the part of the node graph where the x and y positions were stored at.
- 6.) Removed all Nuke Version lines from the gizmos to avoid annoying errors about different versions. Most of these tools were tested using Nuke 11.3v4, but that does not mean they require that version. Some gizmos were created for different versions, so please use the links provided to see what versions the tools are compatible with if something is not working.

- 7.) Tried consolidating the types of channels the gizmos might be bringing into your scripts by making sure they are using the same types of channel names. For example, all Position World pass channels will come in as P.red, P.green, P.blue, P.alpha, and all Normals World pass channels will come in as N.red, N.green, N.blue, N.alpha. There are a few exceptions where some tools are using unique channel names, but for the most part they are always using .red, .green, .blue, .alpha, .u, or .v at the end of the channels. Most channel/layer names are kept as the original tool had them. For example apChroma, hag_pos, despill, etc.
- 8.) Added an Author Tag to the end of all Gizmos in the menu. NKPD just stands for Nukedpedia, where I did not make a custom tag if there weren't many tools from this author. These might help in 2 ways:
 - 1.) To filter for certain tools if you want to search by all of Adrian Pueyo's AP tools or Mark Joey Tang's MJT tools using nuke's tab search. Will also help you identify who made what, and make it easier to find in the Tool Documentation
 - 2.) To help identify that this gizmo is from the Nuke Survival Toolkit, in case there are duplicate tools in the pipeline loaded with the same name.
- 9.) Dealing with Hard Coded filepaths on Gizmo Creation
 - a.) There is a function, filepathCreateNode(), stored in the NST_helper.py file, that first detects if the Group/Gizmo being created has a Read, DeepRead, ReadGeo, Camera, Axis. Then, if the file knob in the node contains the string <<<replace>>> in the filepath, this will be replaced by the location where the NukeSurvivalToolkit is stored.
 - b.) This means for templates, example scripts, and occasional gizmos that require image files, They will be created with hardcoded links pointing to images in the Nuke Survival Toolkit.
 - c.) This was necessary because if I manually hardcoded the filepath, it will error because it does not know where your NST image is. If you use a live variable, similar to [root.name] to try and point to the NST, it will work for you and anyone with the same NST installed, but not if you try and render on a renderfarm without the NST installed or pass the script to the artist without the NST installed, as nuke won't find the variable and won't know where to point to. Replacing the variable and hard coding the filepath on creation is the best way to make sure the tool to work with anyone opening the script, as long as the Nuke Survival Toolkit does not move locations, or the image file is not moved, deleted, renamed, etc

Menus

The tool menu's categorization is laid out in a bit of a mix between Nuke's original toolbar organization, and Nukedpedia's gizmo categories. This should be helpful and intuitive when browsing for certain types of tools, or to quickly find the tool you are looking for if you forget the name. Some of these menus have sub-menus such as `Filter/Glows/` for further groupings to reduce the overall list size of each menu.

Nuke Survival Toolkit Menu Bar:



Nukedpedia's gizmo Categories

Main Repository Page

Gizmos [RSS](#)

Share useful gizmos with the Nuke community.

Deep	(0/15)
Image	(0/74)
Particles	(0/8)
Draw	(0/84)
Time	(0/18)
Channel	(0/45)
Colo(u)r	(0/63)
Filter	(0/161)
Keyer	(0/67)
Merge	(0/22)
Transform	(0/86)
3D	(0/81)
Stereo	(0/16)
MetaData	(0/2)
Other	(0/133)

Tool Index

About	2
Installation	2
Technical Details	3
Menus	6
Tool Index	7
1. Image	15
LabelFromRead TL	15
2. Draw	16
Expression Nodes AG Menu	16
Hagbarth Tools MHD Menu	17
Silk MHD	17
Gradient Editor MHD	18
STiCKiT MHD	19
WaveletBlur MHD	20
GradMagic TL	21
NoiseAdvanced TL	22
RadialAdvanced TL	23
WaterLens MJT	24
VoronoiGradient NKPD	25
CellNoise NKPD	27
LineTool NKPD	28
PlotScanline NKPD	29
SliceTool FR	30
DasGrain FH	31
LumaGrain NKPD	32
GrainAdvanced SPIN	33
PerspectiveGuide NKPD	34
SpotFlare MHD	35
AutoFlare NKPD	36
FlareSuperStar NKPD	37
3. Time	38
apLoop AP	38

Looper NKPD	39
FrameMedian MHD	40
TimeMachine NKPD	41
FrameFiller MJT	42
4. Channel	43
BinaryAlpha TL	43
ChannelCombiner TL	44
ChannelControl TL	45
ChannelCreator TL	46
InjectMatteChannel TL	47
StreamCart MJT	48
RenameChannels AG	50
5. Color	51
BlacksMatch TL	51
ColorCopy TL	52
Contrast TL	53
GradeLayerPass TL	54
HighlightSuppress TL	55
ShadowMult TL	56
WhiteBalance TL	57
WhiteSoftClip TL	58
apColorSampler AP	59
apVignette AP	60
GammaPlus MJT	61
Suppress_RGBCMY SPIN	62
HSL_Tool NKPD	63
BiasedSaturation NKPD	64
6.) Filter	65
X Tools XM	65
X Aton Volumetrics XM	65
X_Denoise XM	66
X_Sharpen XM	67
X_Soften XM	67
X_Distort XM	68
X_Tesla XM	69
GloWS Menu	70
apGlow AP	70
ExponGlow TL	71

Glow_Exponential SPIN	72
Optical Glow BM	73
Blurs Menu	74
ExponBlurSimple TL	74
Directional Blur TL	75
IBlur NKPD	76
Edges Menu	77
apEdgePush AP	77
EdgeDetectAlias TL	78
AntiAliasFilter AG	79
ErodeSmooth TL	80
Edge_RimLight AG	81
EdgeDetectPRO AG	82
Erode Fine SPIN	83
Edge_Expand SPIN	84
Edge RB	85
ColourSmear NKPD	88
KillOutline NKPD	89
EdgeFromAlpha FR	90
VectorExtendEdge NKPD	91
FractalBlur NKPD	92
Distortions Menu	93
Glass FR	93
HeatWave NKPD	94
BeautifulSkin TL	95
BlacksExpon TL	96
Halation TL	97
Highpass TL	98
LightWrapPro TL	99
bm_Lightwrap BM	100
iConvolve AP	101
ConvolutionMatrix AG	102
apChroma Tools AP	103
apChroma AP	104
apChromaTransform AP	105
apChromaBlur AP	106
apChromaUnpremult AP	107
apChromaPremult AP	107
apChromaMerge AP	108

Chromatik SPIN	108
CatsEyeDefocus NKPD	110
DefocusSwirlyBokeh NKPD	111
deHaze NKPD	111
DeflickerVelocity NKPD	113
FillSampler NKPD	116
MECfiller NKPD	117
7.) Keyer	118
apDespill AP	119
SpillCorrect SPIN	120
DespillToColor NKPD	121
AdditiveKeyerPro TL	121
apeScreenClean AP	122
apScreenGrow AP	124
LumaKeyer DR	125
KeyChew RK	126
8.) Merge	127
ContactSheetAuto TL	127
KeymixBBox TL	127
MergeAtmos TL	128
MergeBlend TL	129
MergeAll AP	131
9.) Transform	131
Math Tools MGA-EL	132
InvertAxis MT	133
ZeroAxis MT	133
InvertMatrix4 MT	133
ProductMatrix4 MT	133
RotateMatrix4 MT	133
ScaleMatrix4 MT	134
TransformMatrix4 MT	134
TranslateMatrix4 MT	134
TransposeMatrix4 MT	134
CrossProductVector2 MT	135
DotProductVector2 MT	135
MagnitudeVector2 MT	135
NormalizeVector2 MT	135
RotateVector2 MT	135

TransformVector2 MT	135
CrossProductVector3 MT	136
DotProductVector3 MT	136
MagnitudeVector3 MT	136
MultiplyVector3Matrix3 MT	137
NormalizeVector3 MT	137
RotateVector3 MT	137
TransformVector3 MT	137
GenerateMatrix4 MT	137
GenerateSTMap MT	138
LumaToVector3 MT	138
STMapToVector2 MT	138
Vector2ToSTMap MT	139
Vector3ToMatrix4 MT	139
vector3DMathExpression EL	140
Vectors_Direction EL	140
Vectors_to_Degrees EL	140
VectorTracker NKPD	141
AutoCropTool TL	142
BBoxToFormat TL	143
ImagePlane3D	144
Matrix4x4_Inverse TL	147
Matrix4x4_Math TL	148
MirrorBorder TL	149
TransformCutOut TL	150
IMorph TL	151
RP_Reformat MJT	151
InverseMatrix MJT	154
CardToTrack AK	155
CProject AK	155
TProject AK	156
TransformMatrix AG	157
CornerPin2D_Matrix AG	158
IIDistort EL	159
CameraShake BM	161
MorphDissolve SPIN	162
ITransform FR	162
RotoCentroid NKPD	164
STMapInverse NKPD	165

Transform_Mix NKPD	166
PlanarProjection NKPD	167
Reconcile3DFast DR	167
10.) 3D	169
aPCard AP	169
DummyCam AP	169
mScatterGeo MJT	170
Origami MJT	171
RayDeepAO MJT	177
SceneDepthCalculator MJT	179
SSMesh MJT	181
Unify3DCoordinate MJT	181
UVEditor MJT	182
Distance3D NKPD	187
DistanceBetween_CS NKPD	189
Lightning3D EL	190
GeoToPoints MHD	191
Noise3DTexture NKPD	192
11.) Particles	192
WaterSchmutz DR	193
Sparky NKPD	193
ParticleLights MHD	196
ParticleKiller NKPD	197
12.) Deep	197
Deep2VP Suite MJT	198
Deep2VPosition MJT	199
DVPTolImage MJT	201
DVPort MJT	201
DVPmatte MJT	202
DVPattern MJT	206
DVProjection MJT	206
DVPsetLight MJT	210
DVPfresnel MJT	211
DVPrelight MJT	213
DVPscene MJT	214
DeepToPosition TL	217
DeepRecolorMatte TL	218
DeepMerge_Advanced BM	219

DeepCropSoft NKPD	221
DeepKeyMix NKPD	222
DeepHoldoutSmoothener NKPD	223
DeepCopyBBox NKPD	224
DeepBoolean MJT	225
DeepFromPosition MJT	228
DeepSampleCount MJT	230
DeepSer MJT	230
13.) CG	234
UV_Mapper TL	234
PNZ Suite MJT	237
ConvertPNZ MJT	237
P2N MJT	238
P2Z MJT	239
Z2N MJT	240
Z2P MJT	241
Pos Toolkit MJT	242
PosMatte MJT	243
PosPattern MJT	244
PosProjection MJT	245
Noise3D SPIN	246
Noise4D MHD	247
Relight_Simple SPIN	249
Reproject3D SPIN	249
C44Kernal AP	250
apDirLight AP	252
apFresnel AP	252
CameraNormals NKPD	254
NormalsRotate NKPD	255
EnvReflect_BB NKPD	255
Relight_BB NKPD	256
N_Reflection KNPD	257
SimpleSSS MHD	259
aPmatte AP	259
P_Project NKPD	262
GlueP LS	262
P_Ramp NKPD	263
P_NoiseAdvanced NKPD	264

14.) Curves	265
WaveMachine FL	266
WaveMaker FL	267
WaveCustom FL	267
WaveGrade FL	268
WaveRetime FL	268
WaveMerge FL	269
Randomizer TL	270
AnimationCurve AG	271
CurveRemapper BM	281
NoiseGen BM	281
15.) Utilities	282
GUI_Switch TL	283
NAN_INF_Killer TL	283
apViewerBlocker AP	285
Python_and_TCL AG	285
RotoQC NKPD	288
bm_MatteCheck BM	289
ViewerRender MJT	290
NukeZ MJT	290
Pyclopedia MJT	292
16.) Templates	293
Advanced Keying Template TL	294
Advanced Keying Template Stamps TL	295
STMap Keyer Setup EL	295
Gizmo Demo Scripts	296
WaterLens Demo MJT	297
SSMesh Demo MJT	297
UV Editor Demo MJT	297
Sparky Demo NKPD	297
Particle Lights Demo MHD	297
Special Thanks	298
Contact	299

1. Image

LabelFromRead TL



Author: Tony Lyons

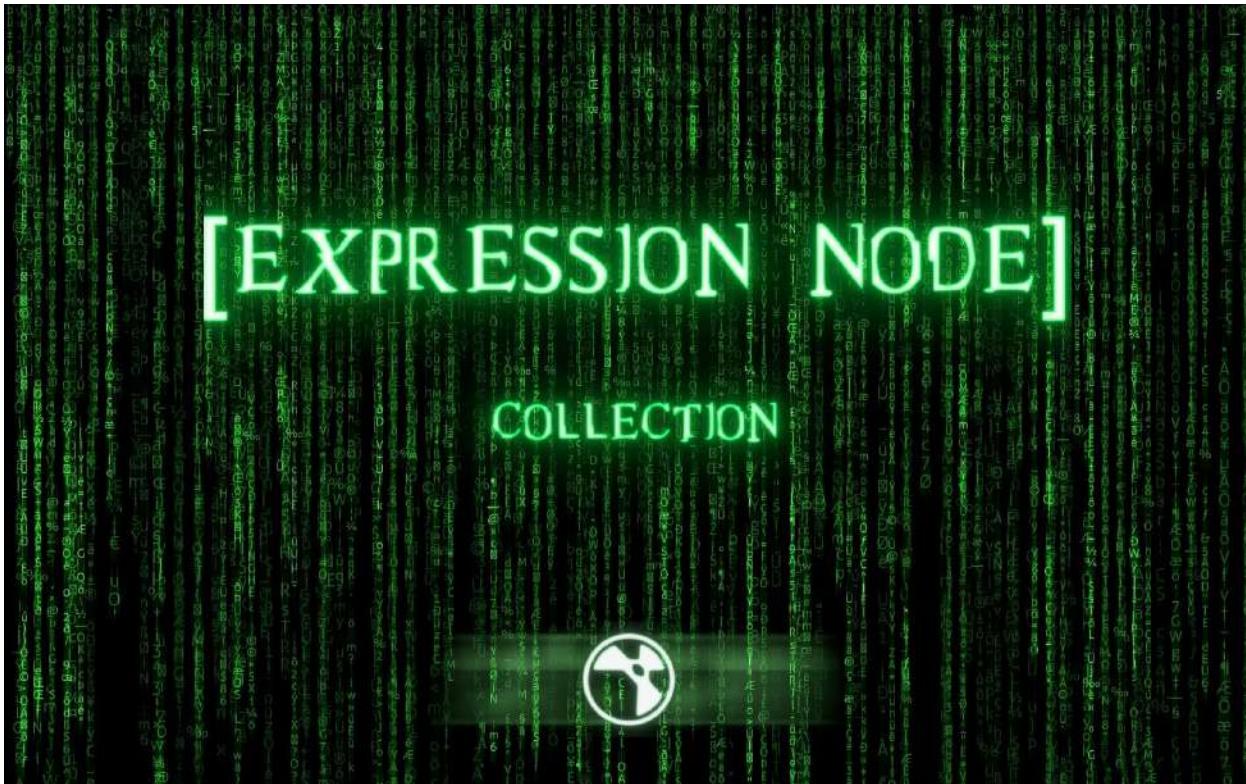
Nukepedia Link: <http://www.nukepedia.com/gizmos/image/labelfromread>

Displays Filepath of topmost read node over the image. Useful when viewing sequence contact sheets / matching shots.

2. Draw

Expression Nodes AG Menu

Author: Andrea Geremia - www.andreageremia.it/tutorial.html



Full tool details: http://www.andreageremia.it/tutorial_expression_node.html

Nukepedia link: <http://www.nukepedia.com/gizmos/other/expression-node-collection-for-nuke>

Quick preview of some of the tools: <https://vimeo.com/364508565>

Various premade expressions. Separated into 6 categories. Please go to the first link above for full details on Andrea Geremia's main website.

- 1. CREATIONS**
- 2. ALPHA**
- 3. PIXEL**
- 4. KEYING and DESPILL**
- 5. TRANSFORM**
- 6. 3D and DEEP**

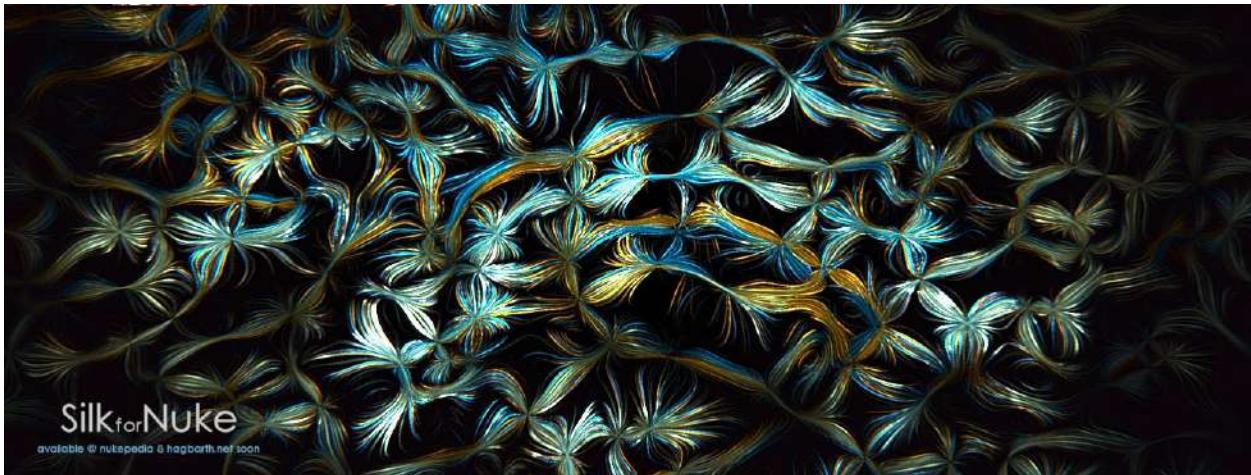
Hagbarth Tools MHD Menu

Author: Mads Hagbarth Damsbo - <https://hagbarth.net/blog/>

A collection of some of Mads Hagbarth Damsbo's best tools

Silk MHD

Author: Mads Hagbarth Damsbo - <https://hagbarth.net/blog/>



Main description website: <https://hagbarth.net/project/silk/>

Nukepedia download: <https://www.nukepedia.com/gizmos/filter/silk>

Silk is a creative 2d processing effect that takes your footage and turns it into laser spaghetti.

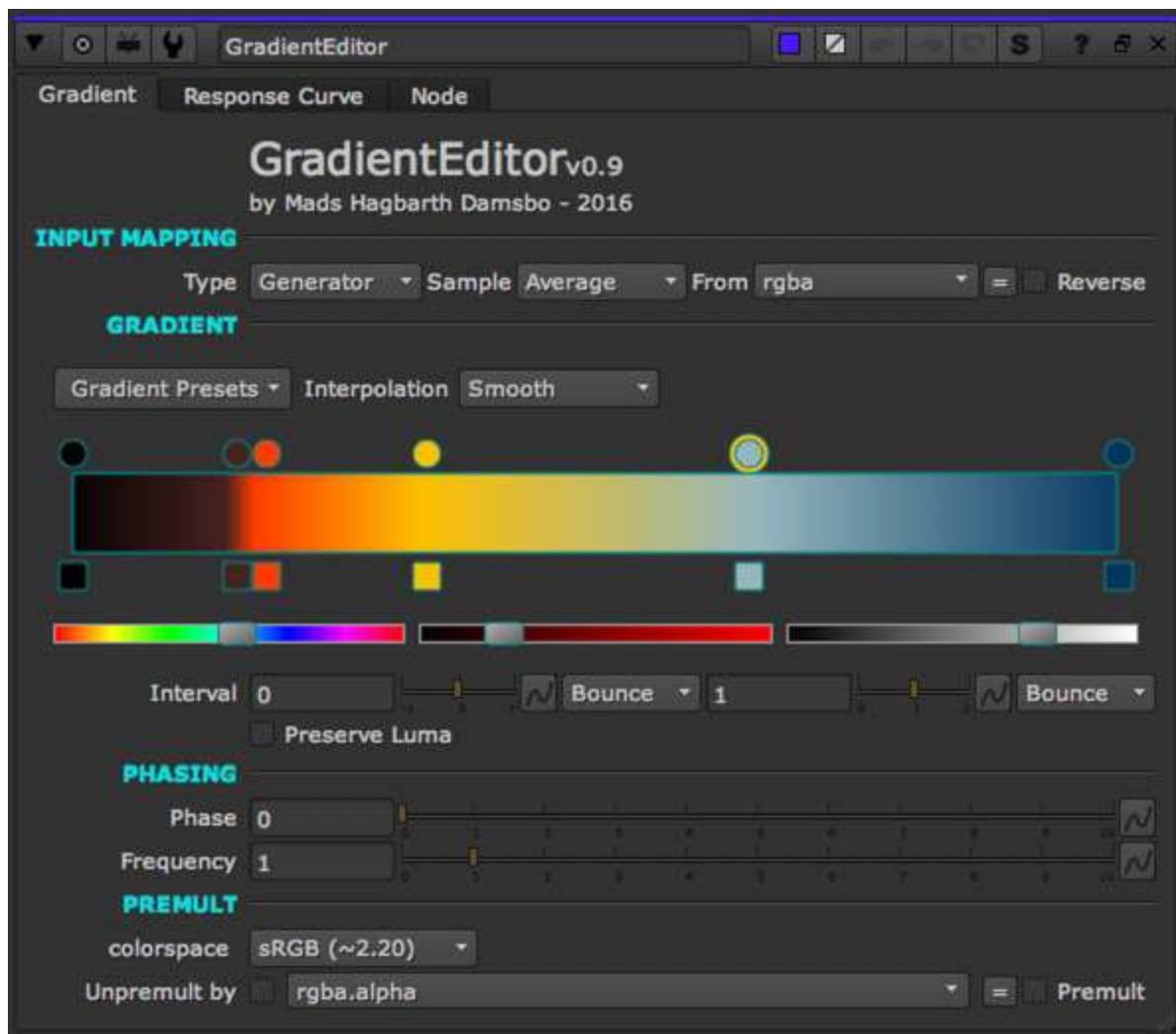
Video demo: <https://vimeo.com/195532256>

Demo: <https://vimeo.com/156336299>

Introduction Video: <https://vimeo.com/195883171>

Gradient Editor MHD

Author: Mads Hagbarth Damsbo - <https://hagbarth.net/blog/>



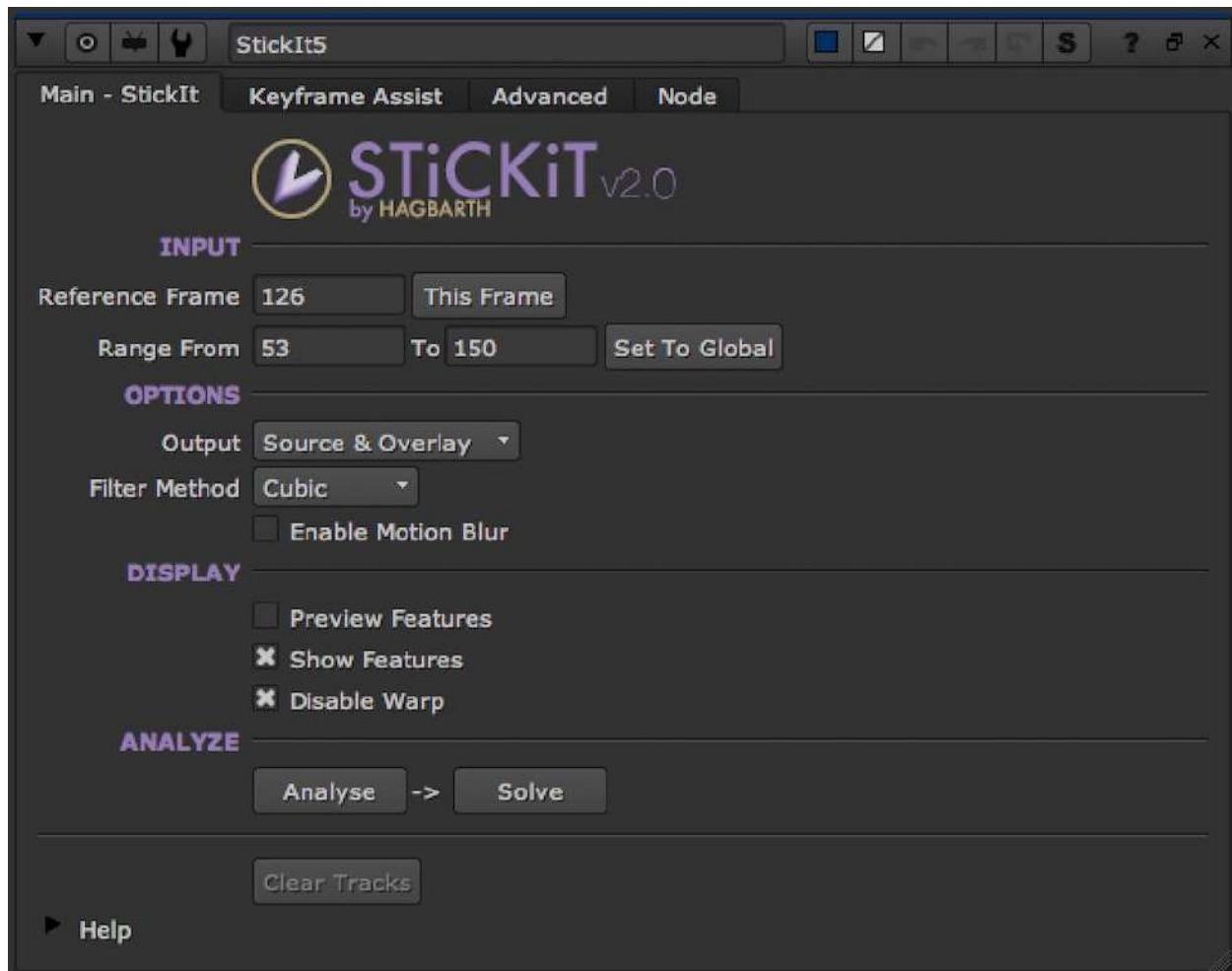
Nukepedia link: <http://www.nukepedia.com/gizmos/draw/gradient-editor>

Preview: <https://vimeo.com/223874378>

This is a simple little visual gradient editor for Nuke

STiCKiT MHD

Author: Mads Hagbarth Damsbo - <https://hagbarth.net/blog/>



Detailed blog post: <https://hagbarth.net/stickit-digital-makeup-gizmo-for-nuke/>

Nukpedia link: <http://www.nukpedia.com/toolsets/transform/stickit-alpha>

Video preview: <https://vimeo.com/94563838>

StickIt V2 is a 2D Warp Match-Moving tool, for matchmoving on (from a 2D perspective) non-ridgid surfaces.

WaveletBlur MHD

Author: Mads Hagbarth Damsbo - <https://hagbarth.net/blog/>



Nukepedia Link: <http://www.nukepedia.com/gizmos/filter/wavelet-blur>

Video preview: <https://vimeo.com/212641249>

This tool allows you to pick a specific range of frequencies to blur in an image. Helpful for doing beauty and other work where preserving original image detail is important. Using a BlinkScript powered bilateral filter, this tool also allows you to preserve edges of your footage, while still having good render times.

GradMagic TL

Author: Tony Lyons - www.CompositingMentor.com



Nukepedia link: <http://www.nukepedia.com/gizmos/draw/gradmagic>

A live sampling 4 point gradient tool with ability to bake colors.

Demo video:

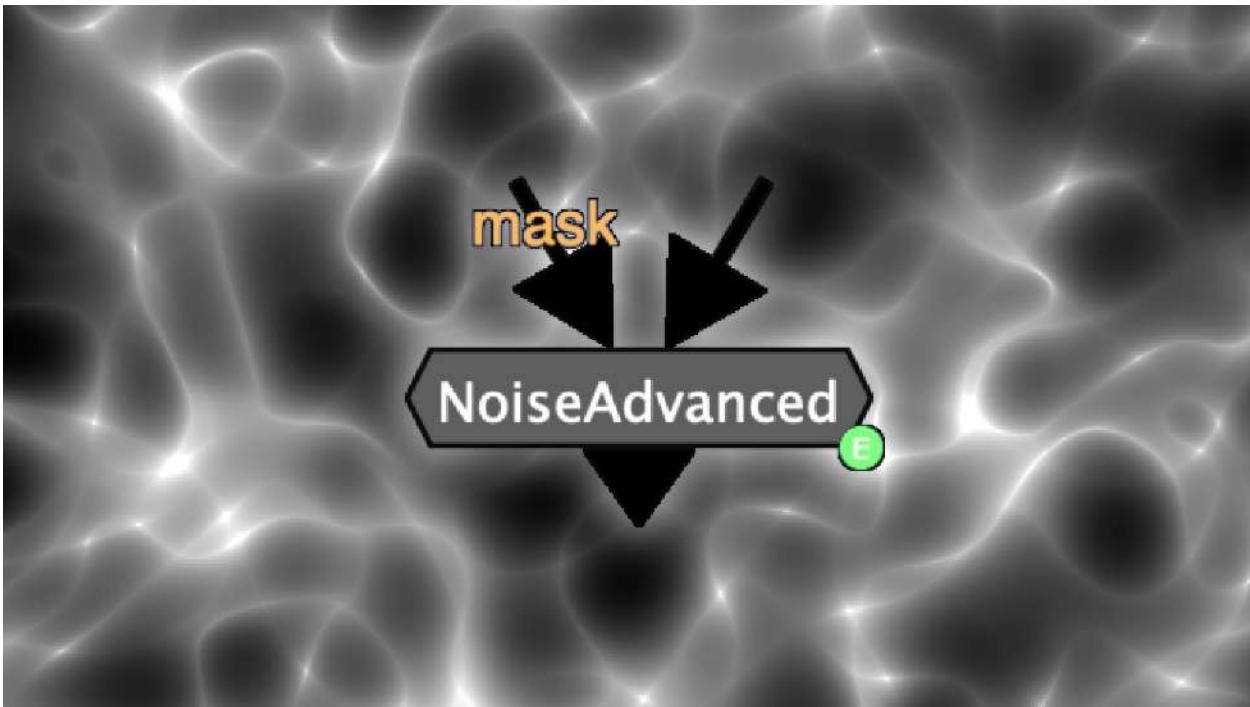
<https://youtu.be/oqe8jMR0LRw>

Or here on vimeo:

<https://vimeo.com/341514150>

NoiseAdvanced TL

Author: Tony Lyons - www.CompositingMentor.com



<http://www.nukepedia.com/gizmos/draw/noiseadvanced>

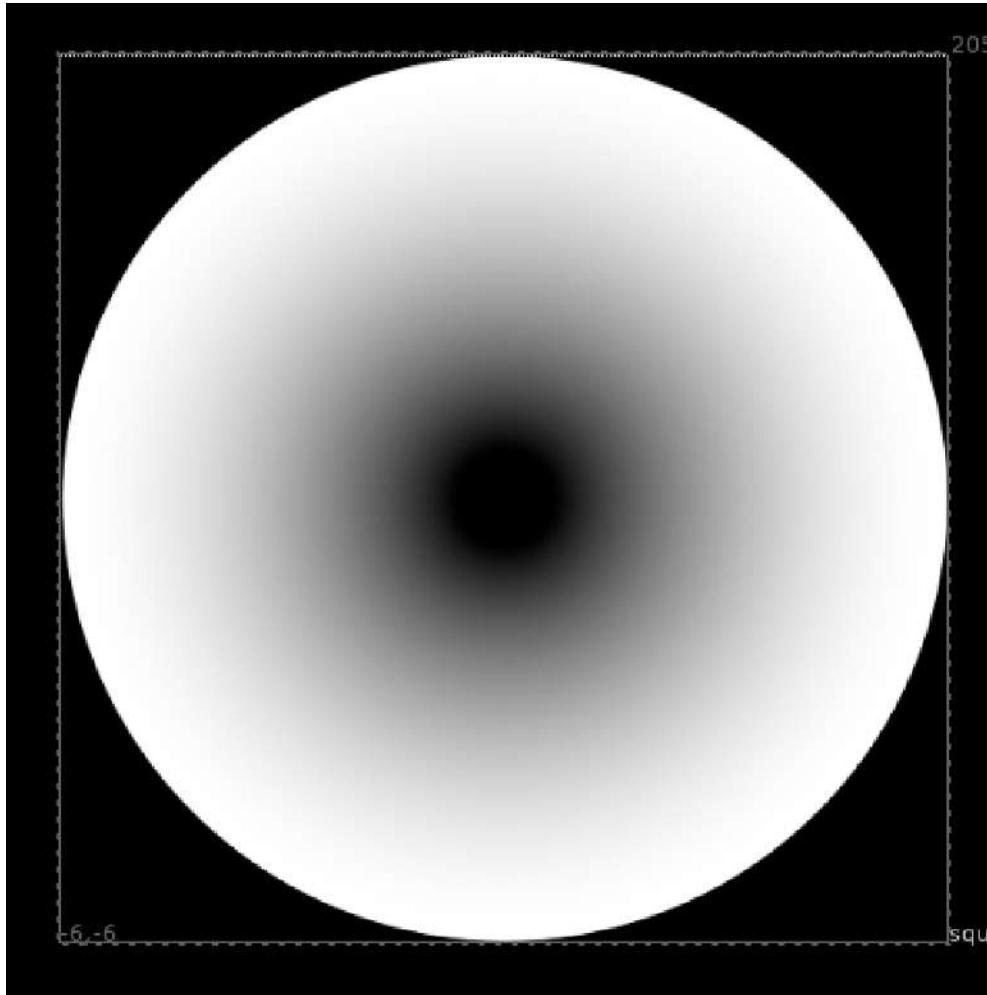
Noise with user friendly animation sliders and overscan.

Demo:

<https://youtu.be/EsHDBGonwEs>

RadialAdvanced TL

Author: Tony Lyons - www.CompositingMentor.com



A radial tool that creates a circle and ramped falloff to create a “ring” effect. Easy animation settings. Useful for shockwaves or other lookDev tasks.

WaterLens MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab

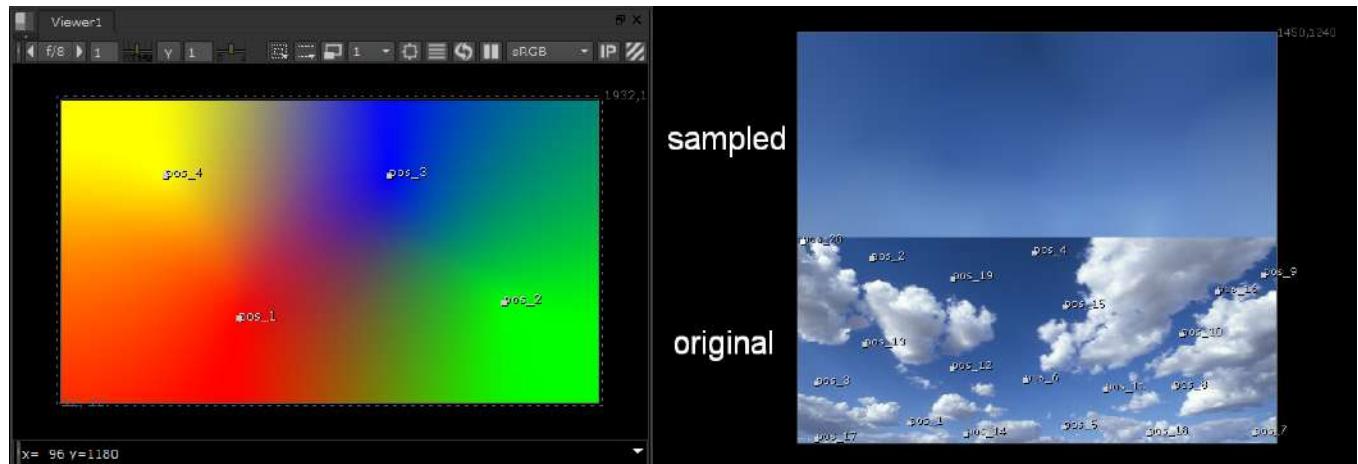


Nukepedia Link: http://www.nukepedia.com/gizmos/draw/waterlens_11

Create water droplet on lens.

VoronoiGradient NKPD

Author: Nikolai Wüstemann - www.wuestemann.net



Nukepedia Link: <http://www.nukepedia.com/gizmos/colour/voronoi-gradient/>

Nuke implementation for 2D Gradients. Create an arbitrary number of color-samples in 2D and produce a smooth, natural interpolation over the entire image.

The Gizmo uses Natural Neighbor Interpolation to calculate the pixels inbetween samples, using Blinkscripts.

You can also output the underlying Voronoi Diagram or play with the smoothness value to control the amount of the softening (0 = Voronoi Diagram, 1 = Accurate Natural Neighbor Interpolation).

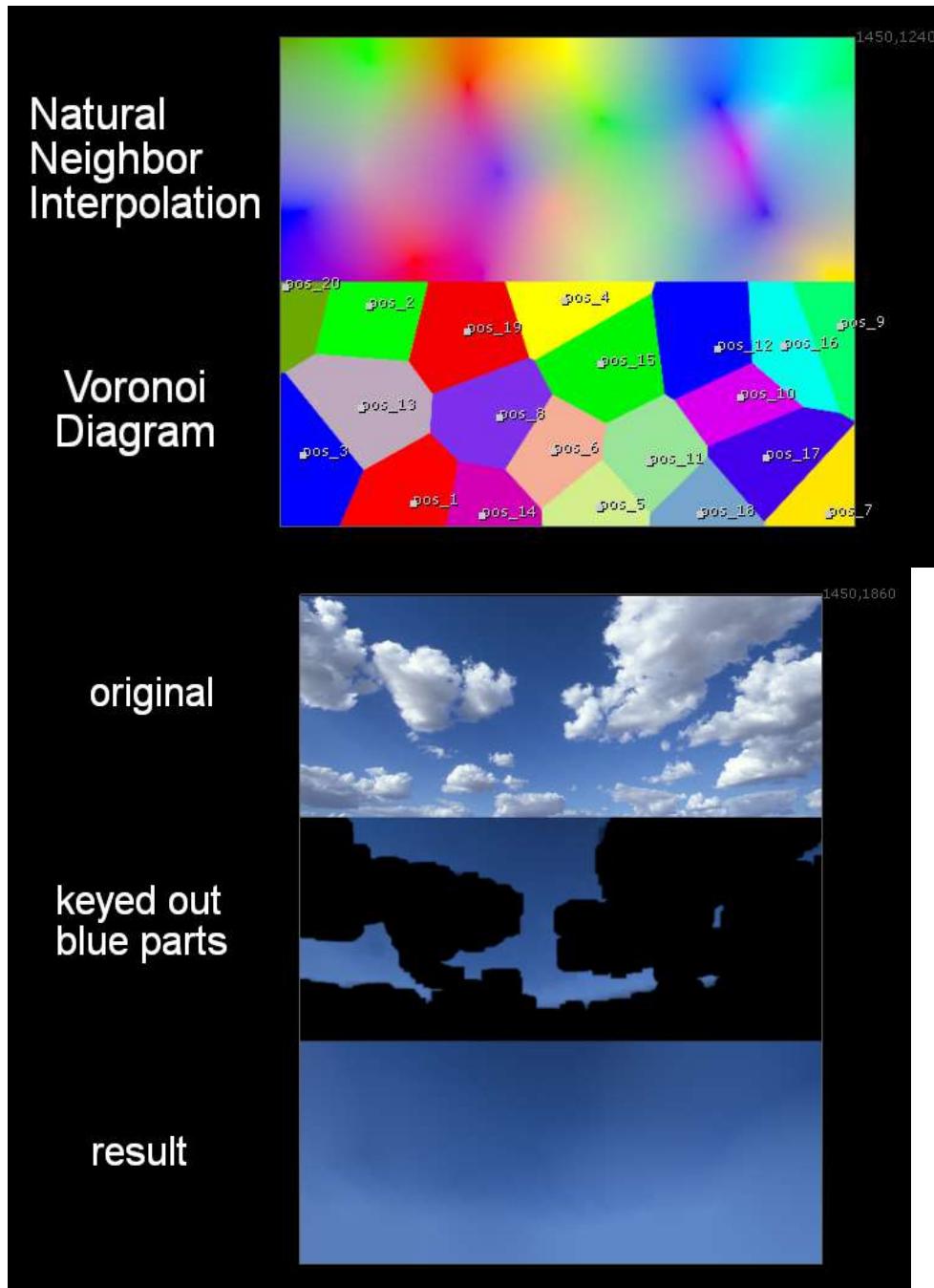
Another important function is the ability to sample input colors, instead of defining them yourself. Setting the Type to 'Sample' uses all created points to sample the input colors at given positions. Furthermore you can use the 'Fill' Type to interpolate missing information in any image. A premultiplied input is required for this.

Changing the Colorspace will change the color falloff. This can be used to achieve the best artistic result. Setting the Colorspace to HSV for example, will interpolate the colors over the spectrum.

There are several tricks and hacks used in this Gizmo to make it work, so please report any bugs you find, I am sure there still are some.

The user knobs and the inside of the gizmo are well documented to help with understanding the concept.

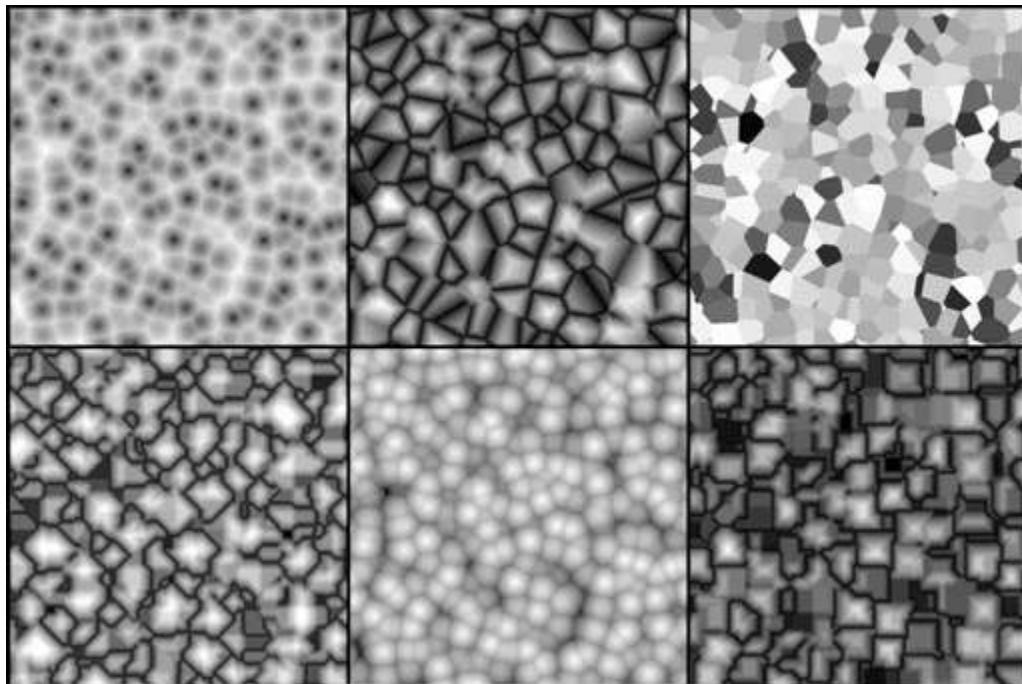
(The algorithm implemented is not the elegant geometric process, but a simple brute-force method, which was easy to implement. This however makes the tool super slow and you might wanna use the speed optimization control to make it a little bit faster at the cost of some quality. That's why I would still consider the whole thing a proof of concept. Although, with my update to v2.7 I'd say we are production ready now :))



CellNoise NKPD

Author: Matthew Shaw

Website: www.gizmosandgames.com



Nukepedia: <http://www.nukepedia.com/blink/image/cell-noise>

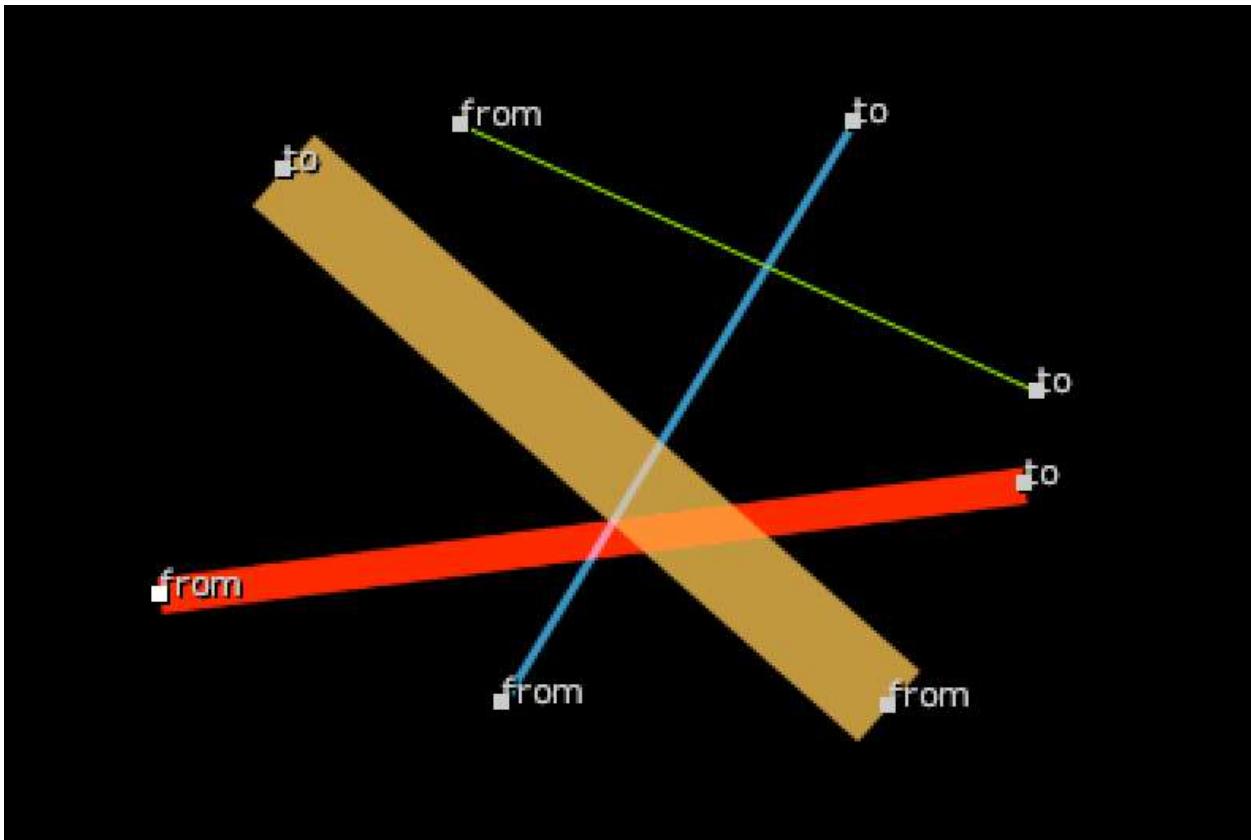
6 cellular noise types :

Worley, Voronoi, Manhattan, Chebyshev, Euclidian, and Worley Inverse.

Uses the same transformation controls as standard nuke noise.

LineTool NKPD

Author: Fredrik Brännbacka



Nukepedia Link: <http://www.nukepedia.com/gizmos/draw/mcp-line>

Line drawing gizmo. Use it to draw lines on an input or use it as an input draw node.
Subsample or don't if you are drawing vertical or horizontal lines.

PlotScanline NKPD

Author: Theodore Groembroome



www.euqahuba.com | theo@euqahuba.com | 2010

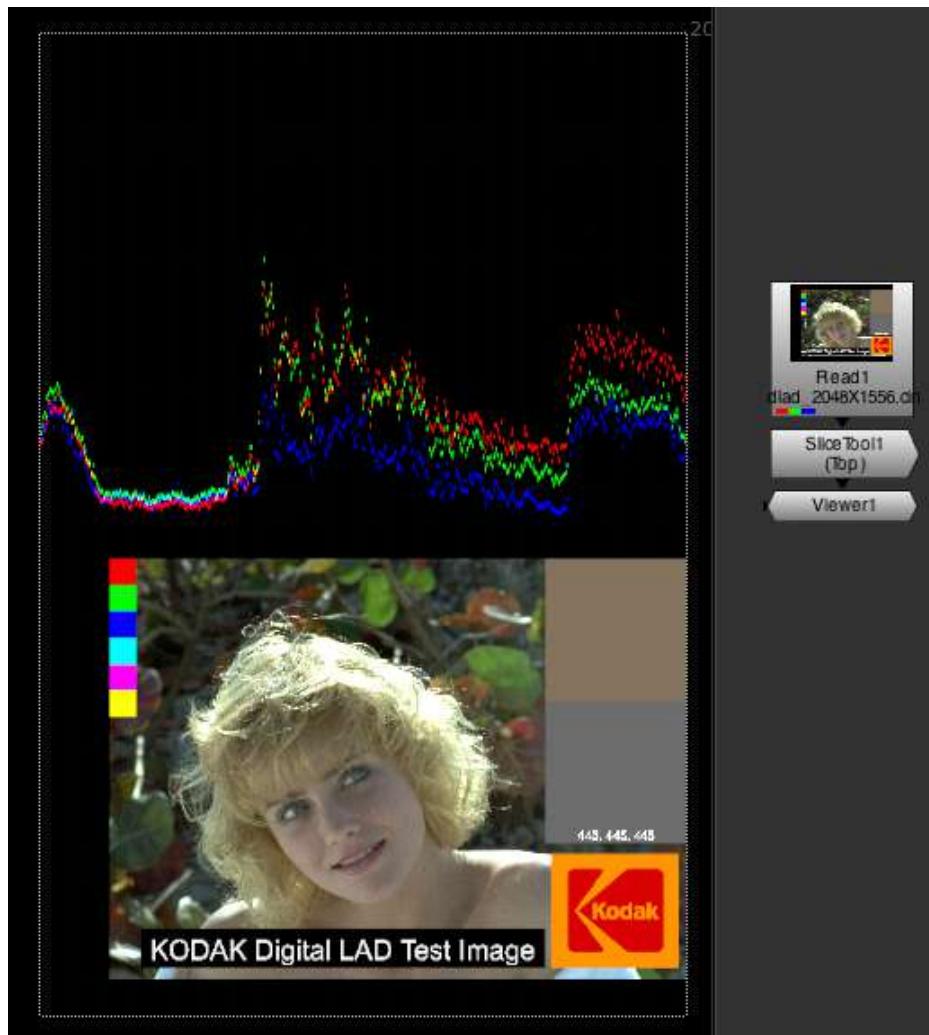
Website Download: <https://euqahuba.com/blog/?p=121>

Slice and plot scanlines in Nuke!

Set up point 1 and point 2 and calculate alone whole line from edge of frame to edge of frame or calculate only the area between the 2 points.

SliceTool FR

Author: Frank Rueter - www.ohufx.com

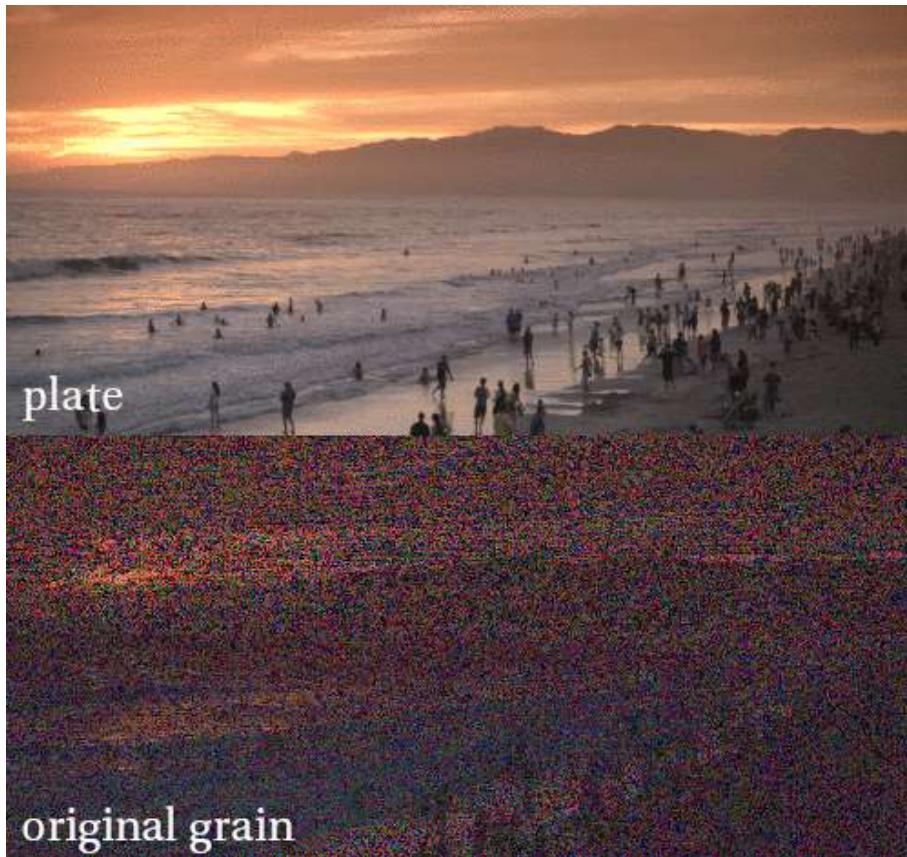


analyze an arbitrary slice of an image. Place start and end position on the incoming image to plot a scan line to represent an arbitrary slice.

Thanks to Ben Pierre for a little magic under the hood.

DasGrain FH

Author: Fabian Holtz



Nukepedia Link: <http://www.nukepedia.com/gizmos/other/dasgrain>

Semiautomatic regraining tool.

DasGrain takes the pain out of the re-graining process by automatically adapting the plate grain to your comp.

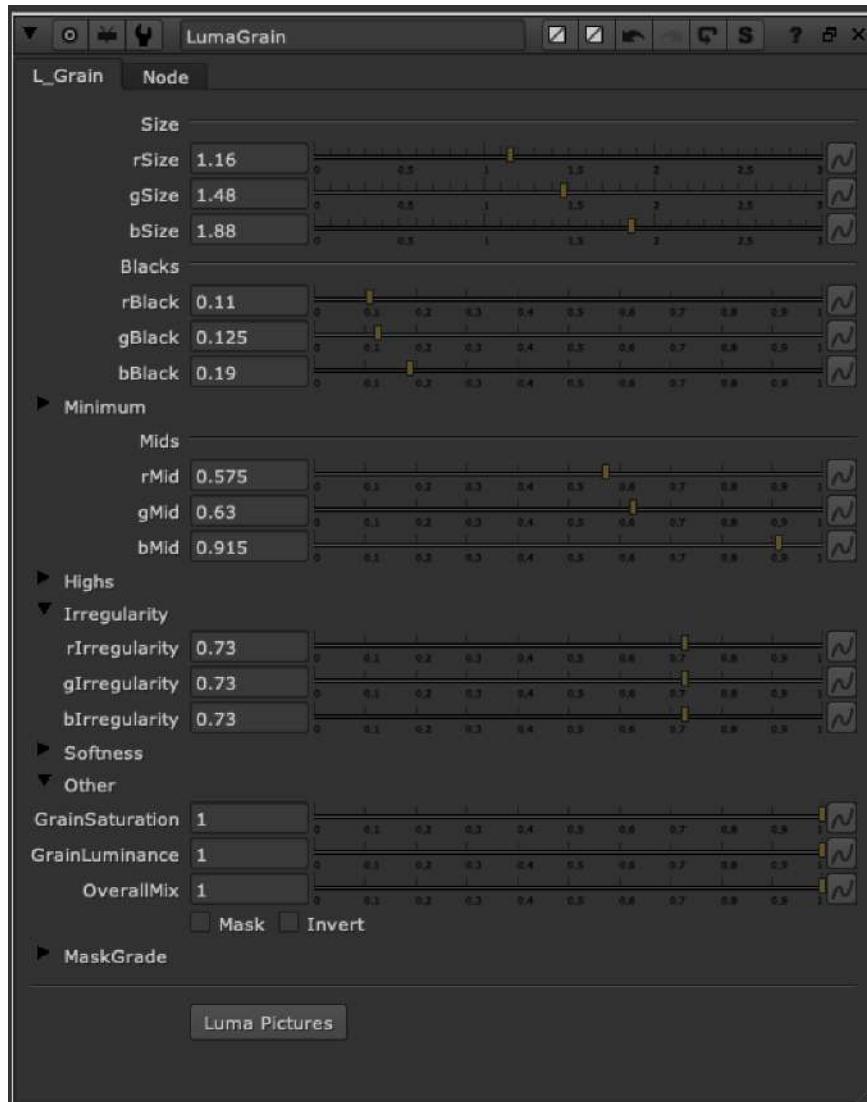
For areas where the original grain doesn't work, you can generate a new grain that is based on the original grain.

Having a good degrain is crucial for this tool!

Amazing Video demo here: <https://vimeo.com/284820390>

LumaGrain NKPD

Author: Luma Pictures

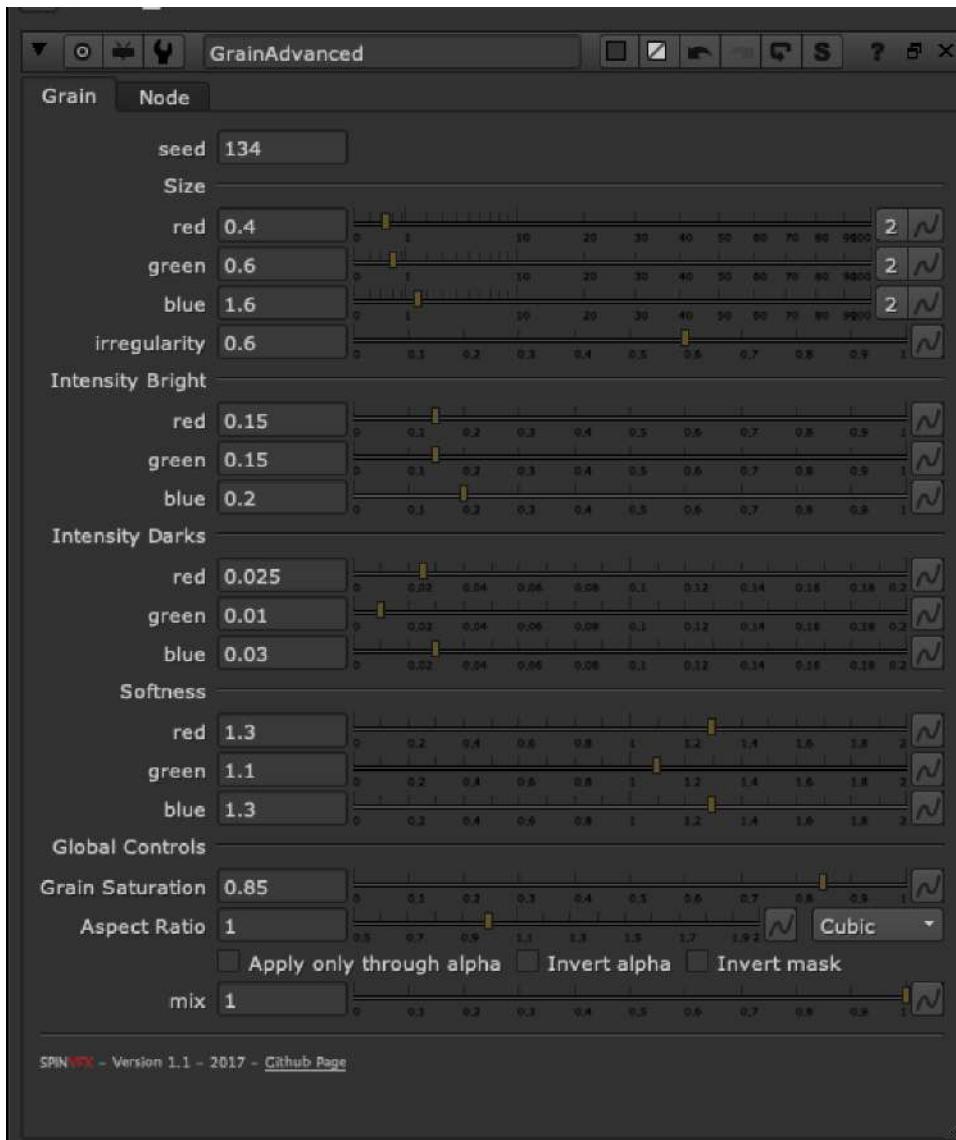


Nukepedia link: http://www.nukepedia.com/gizmos/draw/l_grain

Added functionality to Nuke's default grain node.

GrainAdvanced SPIN

Author: Spin FX



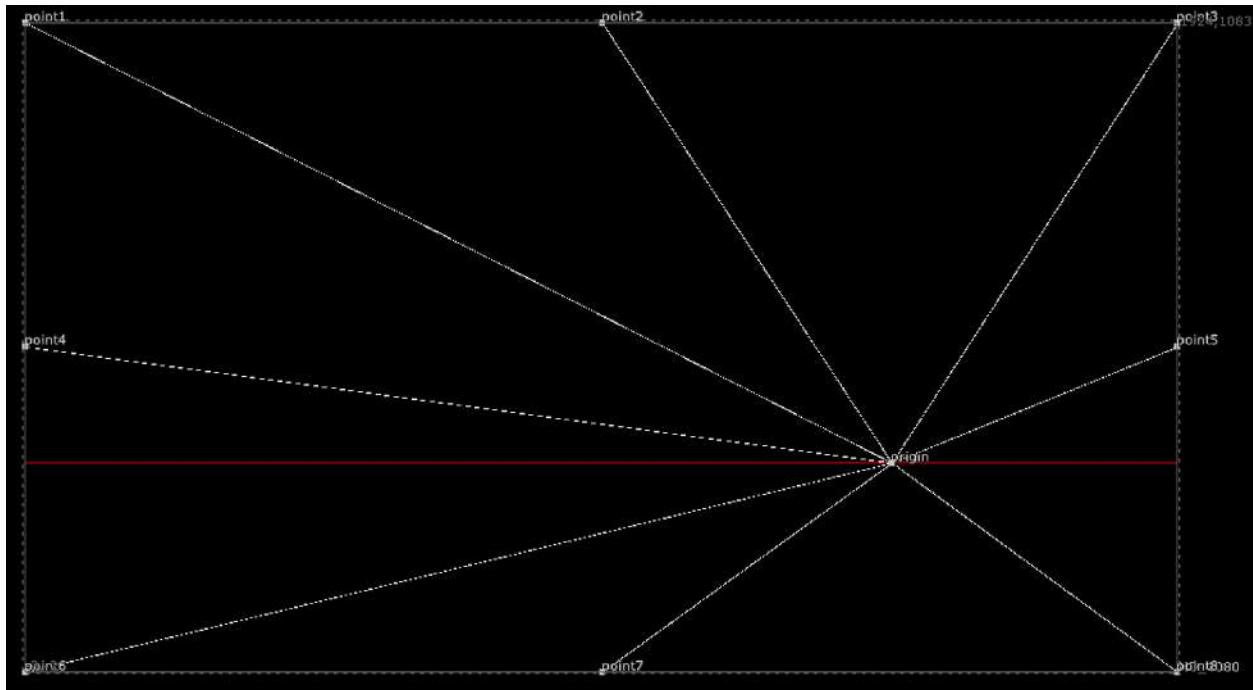
Nukepedia link: http://www.nukepedia.com/gizmos/other/spin_nuke_gizmos-1

Github link: https://github.com/SpinVFX/spin_nuke_gizmos

Adds synthetic grain. The defaults are setup to resemble an HD Alexa plate's grain. You can adjust the sliders to match a sample grain.

PerspectiveGuide NKPD

Author: Peter Farkas - Baseblack (London) Ltd.



Nukepedia link: <http://www.nukepedia.com/gizmos/other/perspective-guide-110>

Simple perspective guide.

SpotFlare MHD

Author: Mads Hagbarth Damsbo - <https://hagbarth.net/blog/>



Website: <https://hagbarth.net/project/spotflare/>

Nukepedia link: <http://www.nukepedia.com/gizmos/draw/spot-flare>

Spotflare is a procedural flare generator, that generates a general radial light but also allow for light shimmer and "cone" masking.

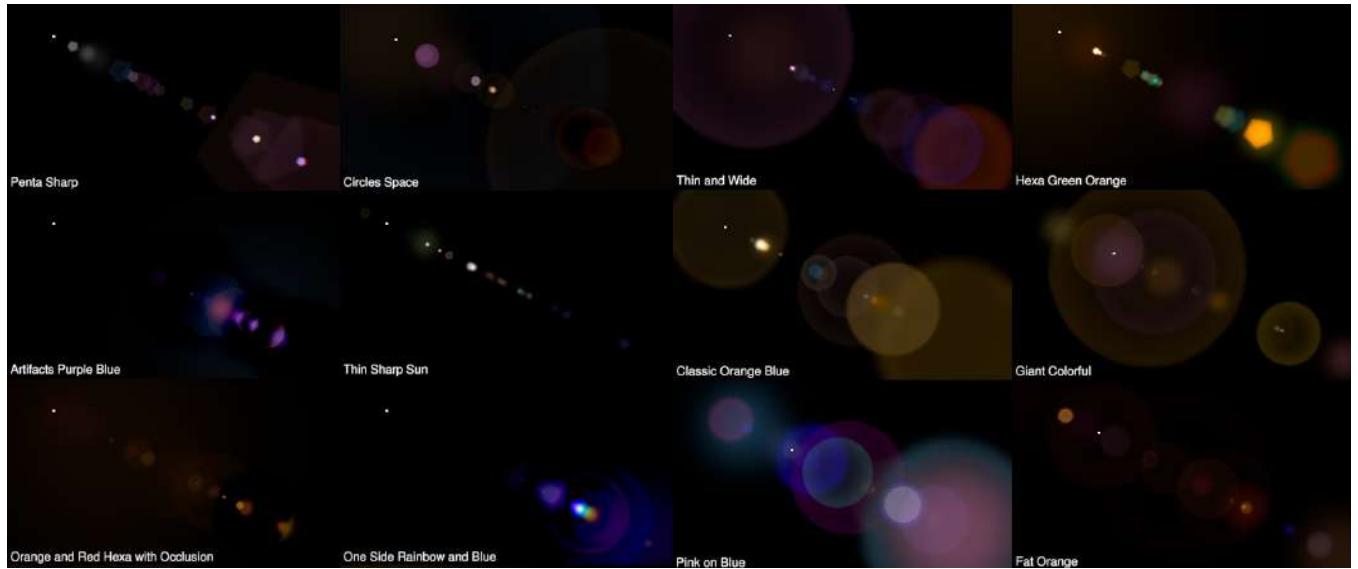
At the core of spotflare is a radial light, that is generated from the inverse square of the distance to the center. Unlike a linear falloff, a inverse square falloff gives a very realistic light look.

If you look at the difference between a gaussian and a gamma adjusted inverse square profile you can clearly see the effect.

Demo: <https://vimeo.com/116440577>

AutoFlare NKPD

Author: Vincent Wauters - www.vincentwauters.com



Nukepedia Link: <http://www.nukepedia.com/gizmos/filter/autoflare>

Detailed description on AutoFlare:

<http://vincentwauters.com/programming/autoflare20-for-nuke>

This is an automatic lens flare filter based on image content and values.

It is using simple expressions and convolution filters to create lens flares. Since it is based on image content, there are no position parameters, and therefore no need to track the hotspots in the image.

FlareSuperStar NKPD

Author: Lukas Fabian



Nukepedia link: <http://www.nukepedia.com/gizmos/draw/flaresuperstar>

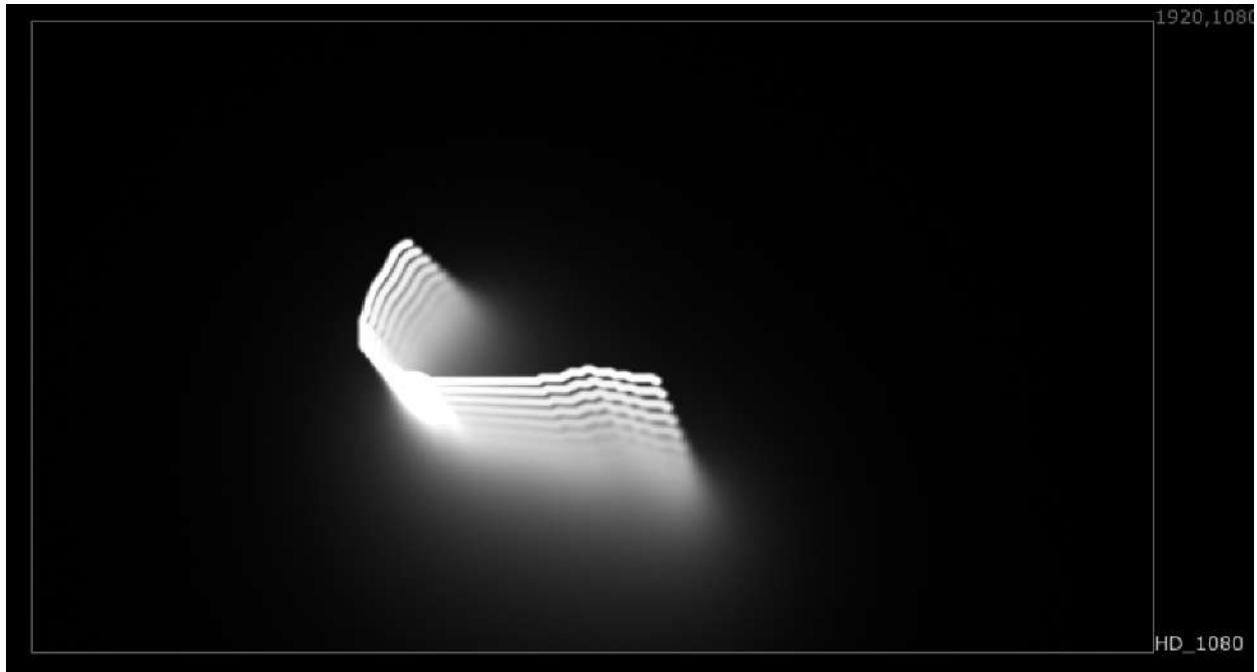
Main features

- easily create star flares with separate controls over rays, glare and glow
- automatic animation on the glare either by shimmer (over time) and/or by changing the position of the flare
- rays are very customizable and have controls for adjusting thickness and angle + spread (or shrink) from a specific distance
- position the flare either manual or let it spawn automatically from the highlights of an input image with convolve mode

3. Time

apLoop AP

Author: Adrian Pueyo - <http://www.adrianpueyo.com/>



Quick tool to simulate a loop effect while affecting the gain, blur and transformations on each "iteration".

Feel free to play with it and see its applications. Some of them: create an exponential (or normal) glow in seconds, an expo blur, a grid or mosaic (adding this gizmo twice), godrays, directional blurs, etc.

Looper NKPD

Author: Damian Binder

Original:



Morph:



Nukepedia Link: <http://www.nukepedia.com/gizmos/time/looper>

Looper is a tool that allows you to create seamless loops of playing clips.

Looping is not only a useful way to lengthen clips that are too short to be used in a comp but it is also a great way to save rendering times since only a specific frame range needs to be rendered.

Looper has two looping methods: Dissolve and Morph

Dissolve: Uses a simple Dissolve node to dissolve the input clip between an offset version of the same clip. The offset amount is determined by the number of frames looped. Works great when looping clips with constant/similar movement like rain or snow stock footage.

Morph: Similar to the Dissolve method but instead uses a Kronos node to calculate motion vectors to then generate a morphed loop. Best used with clips containing complex but well defined movement.

Other Features:

- FgMatte input: An optional matte of the foreground can be used which may improve Kronos's motion estimation. (Highly recommended)
- Vector generator settings like Vector Detail and Strength are accessible.
- Avoid Clip End knob: Negatively offsets the input clip to avoid looping unexisting frames before first frame.
- A dynamic text visually indicates the total number of frames being looped.

FrameMedian MHD

Author: Mads Hagbarth Damsbo - <https://hagbarth.net/blog/>



Nukepedia link: <http://www.nukepedia.com/blink/time/framedmedian>

Website: <https://hagbarth.net/1054/>

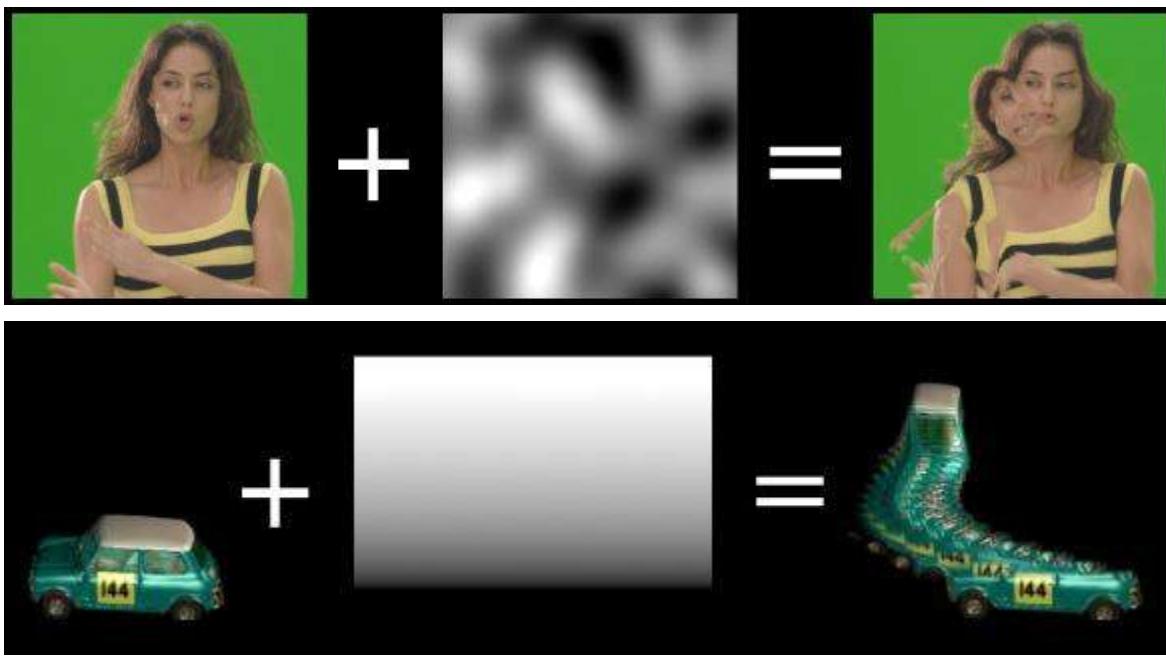
FrameMedian is a temporal median toolset that calculates a median from a range of frames. Unlike the TemporalMedian tool that samples 3 frames, the FrameMedian can sample up to 20 frames.

What is it for?

The tool is generally used for creating cleanplates from super very busy shots.

TimeMachine NKPD

Author: Ivan Busquets



Nukepedia link: <http://www.nukepedia.com/gizmos/time/timemachine>

Does a per-pixel time offset on the image, based on a secondary mask input.

This is sort of a copy of Houdini's Time Machine. Takes an image sequence plus a mask, then offsets the image in time based on the input mask as follows:

- Pixels with a mask value of 1 will be offset by the number of frames set in the "frames" knob.
- Mask values of 0 return the image at the current frame.
- Values between 0-1 will return an interpolated offset.
- Mask gets clamped to 0-1, so values <0 and >1 are not accounted for.

Very simple as it stands, but could be a good example of rebuilding a gizmo's internals using callbacks.

FrameFiller MJT

Author: Mark Joey Tang - blog: www.facebook.com/MJTLab



Nukepedia link: <https://www.nukepedia.com/gizmos/time/framefiller>

Github link: https://github.com/xmjtx/MJTLab/tree/main/gizmo_library/Time/FrameFiller_v10

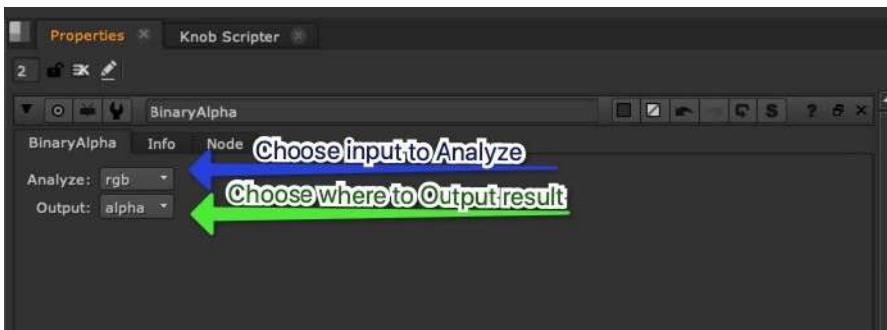
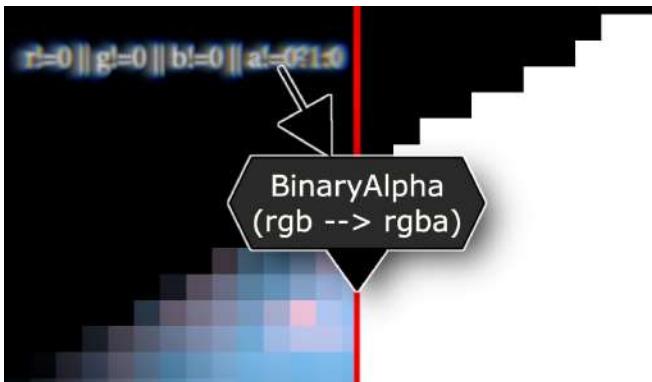
Video Demo: <https://youtu.be/VZSk7j1zOCU>

Using frame interpolation to fill up missing frames. Support multiple frames at once, support Regularized motion estimation, no NukeX Required. Motion blur, sub-frames bias adjustable.

4. Channel

BinaryAlpha TL

Author: Tony Lyons - www.CompositingMentor.com



Nukpedia Link: <http://www.nukpedia.com/gizmos/channel/binaryalpha>

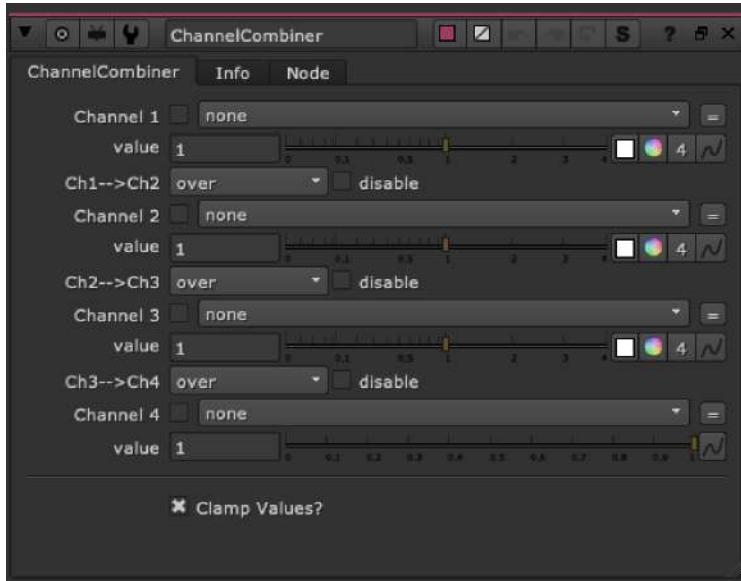
Analyzes a choice of the RGB, RGBA, or Alpha input and outputs an Alpha Channel (or RGBA) that is Binary, 0 or 1. Any Pixels that are not 0 will be turned into 1 (negative numbers also), and 0 will remain 0. This is perfect for those blur + unpremult tricks or if you need a quick matte for finding any rgb color above or below 0, in a CG render passes for example.

Tcl expression:

$r!=0 \parallel g!=0 \parallel b!=0 \parallel a = 0 ? 1 : 0$

ChannelCombiner TL

Author: Tony Lyons - www.CompositingMentor.com

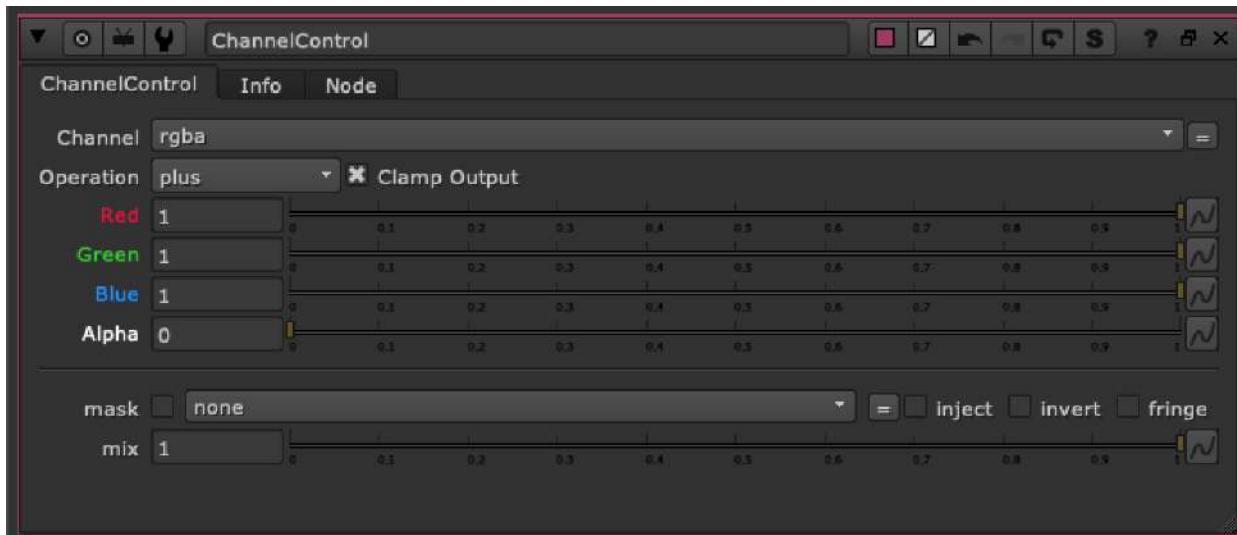


Quickly combine 4 channels with an operation between each of them. Works best with ID mattes or roto's that are injected into a single stream. The channels dropdown lists every channel in the stream, so the best workflow would be to have a “mattes Stream” or “ID stream” with all the matte/roto/ID channels copied into an empty stream (no other channels)

Then you can use this node to make quick combinations:
Helmet.red plus Visor.red minus Antenna.red, for example

ChannelControl TL

Author: Tony Lyons - www.CompositingMentor.com

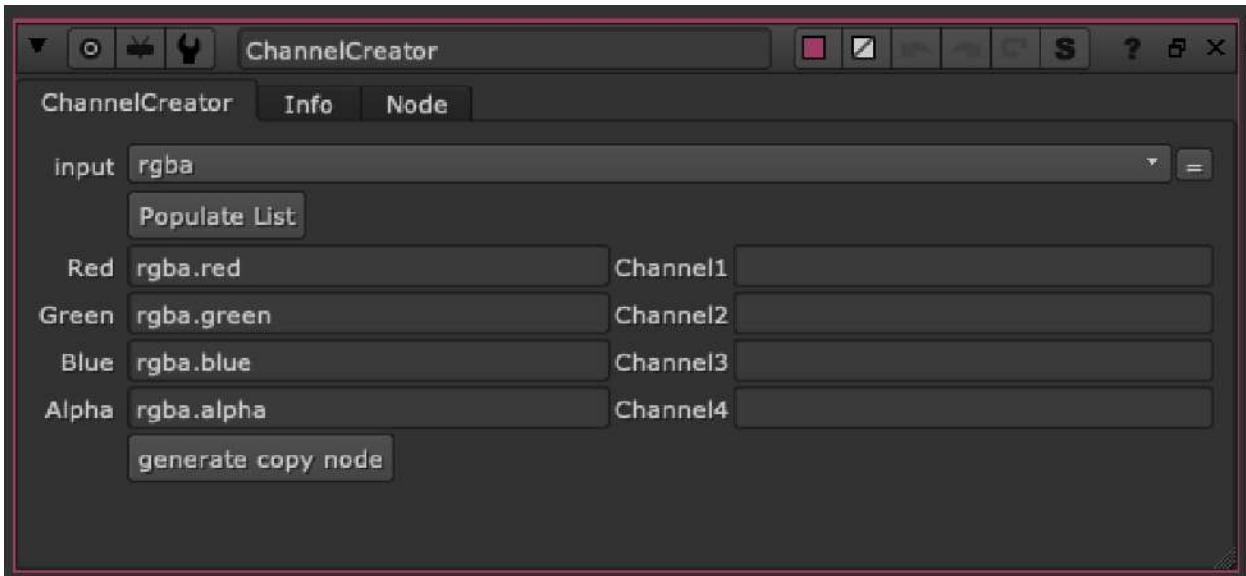


Mix the ratio of Red, Green, Blue, Alpha Channels and choose a Merge operation.
Result is a black and white matte output into RGBA. Mask and mix options available.

ChannelCreator TL

Author: Tony Lyons - www.CompositingMentor.com

python help by Carlo Cherisier



This tool is meant to be used with ID's or mattes that are in the RGBA layer. Create new names for the channel you wish to copy over and this node will create a copy node that converts rgba.red, rgba.green, rgba.blue, and rgba.alpha over to new channels. Just enter a name in the right side column and the suffix will be <string>.red

Use this to quickly transfer and rename channels into a new “channel” stream in nuke that are uniquely named and can be identified and pulled out later to be used when compositing. Good for roto shapes or ID's when CG compositing.

Example:

```
rgba.red --> hat      =  hat.red
rgba.green --> glasses =  glasses.red
rgba.blue  --> shoes   =  shoes.red
rgba.alpha --> jacket  =  jacket.red
```

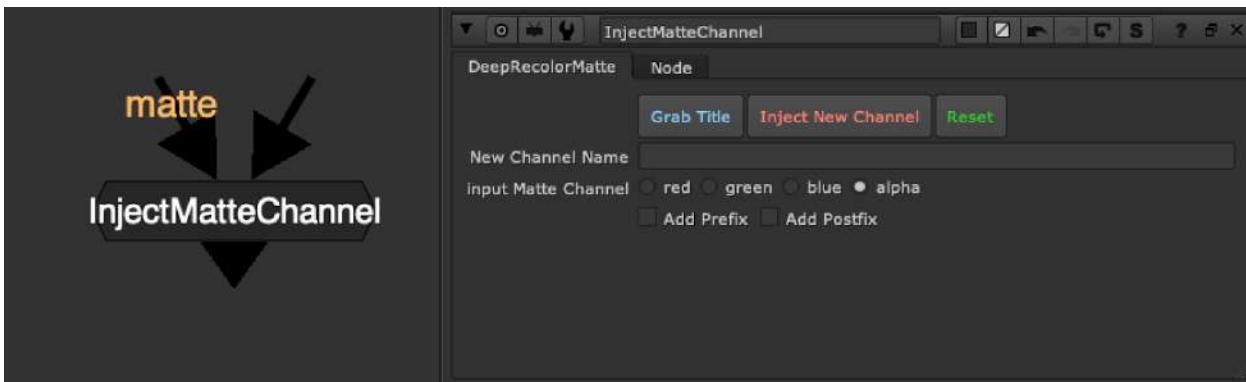
Generate copy node will generate a copy node as follows:

```
rgba.red --> hat.red
rgba.green --> glasses.red
rgba.blue  --> shoes.red
rgba.alpha --> jacket.red
```

Any fields left blank will not copy over in the generated copy node.

InjectMatteChannel TL

Author: Tony Lyons - www.CompositingMentor.com



Takes a matte input and injects it into the stream as a new channel, using the “New Channel Name text input” and adding ‘.red’ to the end, creating a new <channelName>.red channel.

A couple of buttons for ease of use, but ‘grab title’ requires adrian pueyo’s stamps installed to work properly because it is running a function from his code.

Grab Title: Will try and grab either the nearest stamp name, from the matte stream, or else a read node name if it finds a read node. Hopefully if you are using a pipeline, and stamps, and have configured stamps to default to part of the name of a read node to use as the name (you can adjust this in the stamps_config.py file) then it will run the same function to find the ‘right’ name to use and fill in the “New Channel Name” text input.

Inject New Channel: This will generate a new channel based off the new channel name input text and the add prefix, add post fix text inputs. It will copy the selected channel from ‘Input Matte Channel’ into the stream

Reset: Set node to default (does not remove the channel from the script)

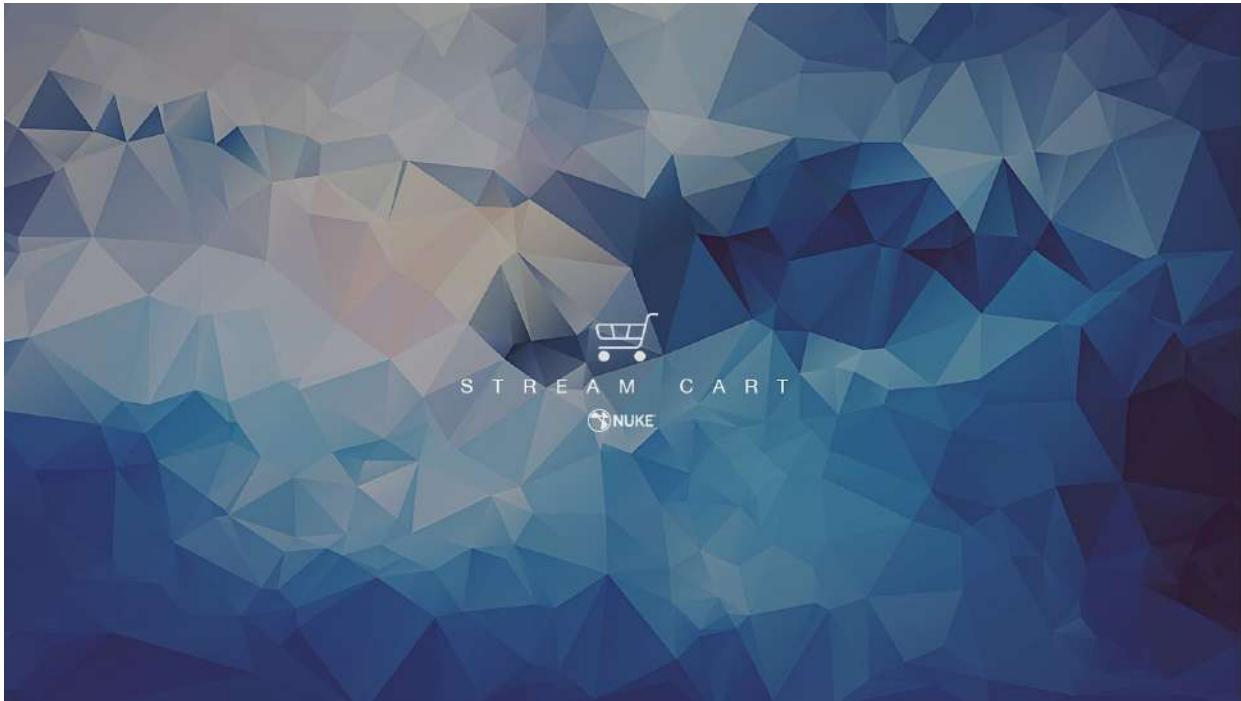
New Channel Name: Manually enter the new channel name, or try the Grab Title to autofill

Input Matte Channel: Choose which channel from the matte input with be copied as new channel

Add Prefix, Add Postfix : click to reveal additional input text fields, where you can add a prefix, which will appear before the new channel name, with an underscore. ie. “ID_” or a postfix, which will add text after your new channel name, with an underscore. ie. “_Dlmatte”

StreamCart MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Demo video : <https://vimeo.com/367649727>

Nukepedia link: <https://www.nukepedia.com/gizmos/channel/streamcart>

Download Mark Joey Tang's entire Toolset: <http://bit.ly/menupy>

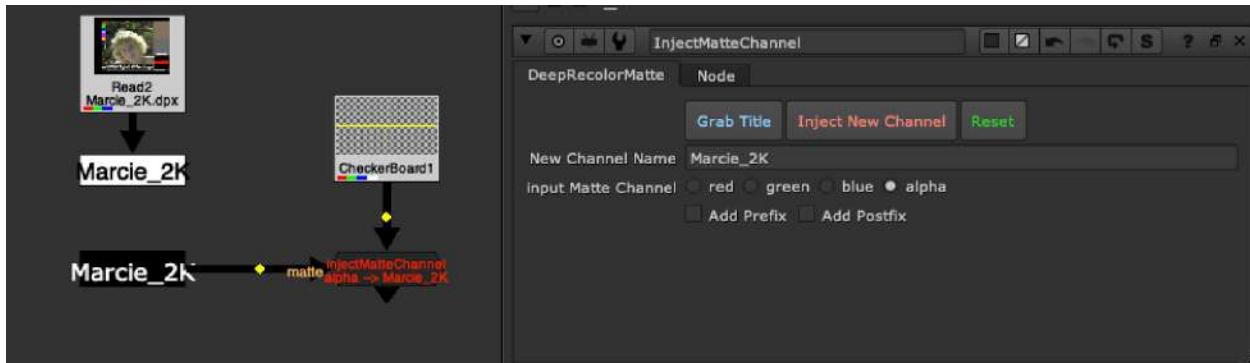
Select channel or geo and quickly shuffle it.

How to use (channel shuffle) :

- connect streamCart node to anywhere of the tree.
- click 'get channels / geo'. It will scan through available channels from upstream
- click the channels to create shuffle node

How to use (Geo select) :

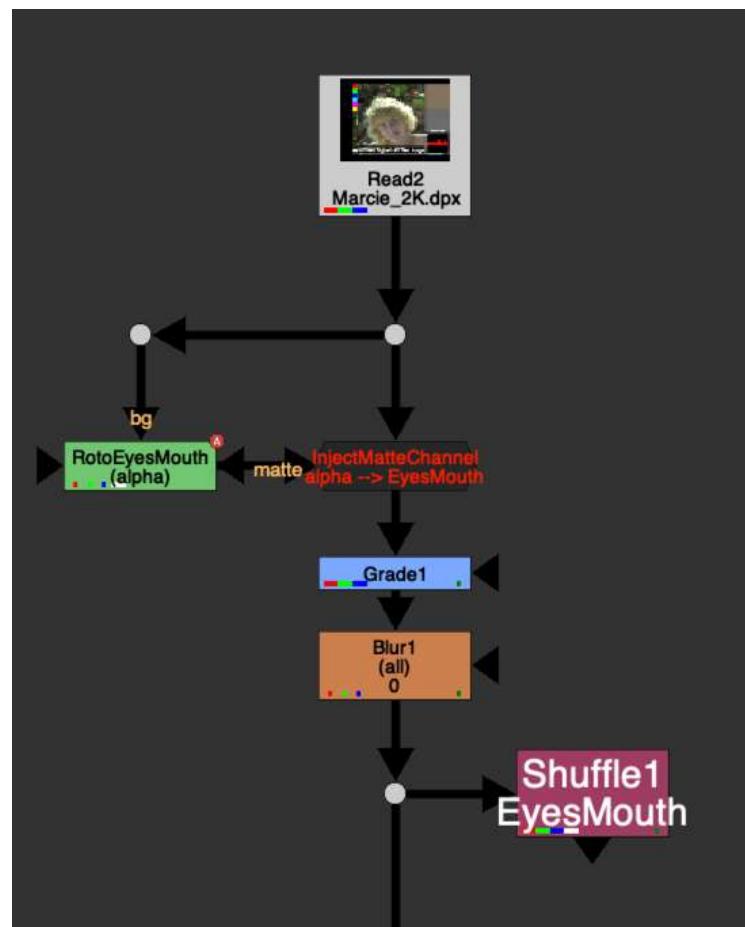
- connect streamCart node directly to 'ReadGeo'
- click 'get channels / geo'. It will scan through available geo mesh from upstream 'scene_view'
- select all the individual objects
- click 'ReadGeo Checkout' to create a new ReadGeo with the selection.



Above is an example of InjectMatteChannel ‘Grab Title’ grabbing the name of the incoming stamp. This can speed up your workflow quite a bit if you are used to using stamps to move around AOVs and mattes, and parts of your script with an alpha that you might want to use as a matte

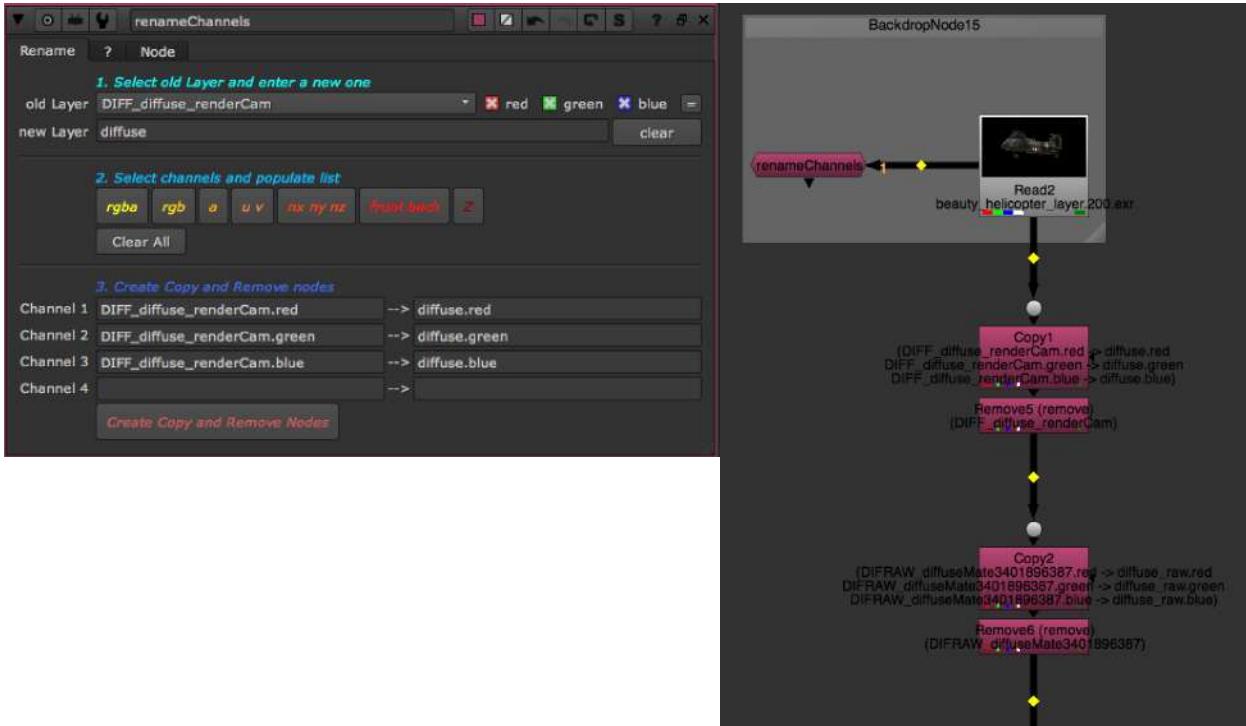
To the Right is a small example of how you might use a channel workflow. I have roto’ed the eyes and mouth of Marcie, and used InjectMatteChannel to copy the alpha into a new channel ‘EyesMouth’.

This channel now exists in the stream as another ‘layer’ you can see it in your viewer channels, and later on you can shuffle out the channel and use it for masks, grain matte, DI mattes, etc



RenameChannels AG

Author: Andrea Geremia - www.andreageremia.it/tutorial.html



With this Gizmo you can rename Channels and Layers through Copy and Remove Nodes. In this way you have more control with the possibility to cancel the operation.

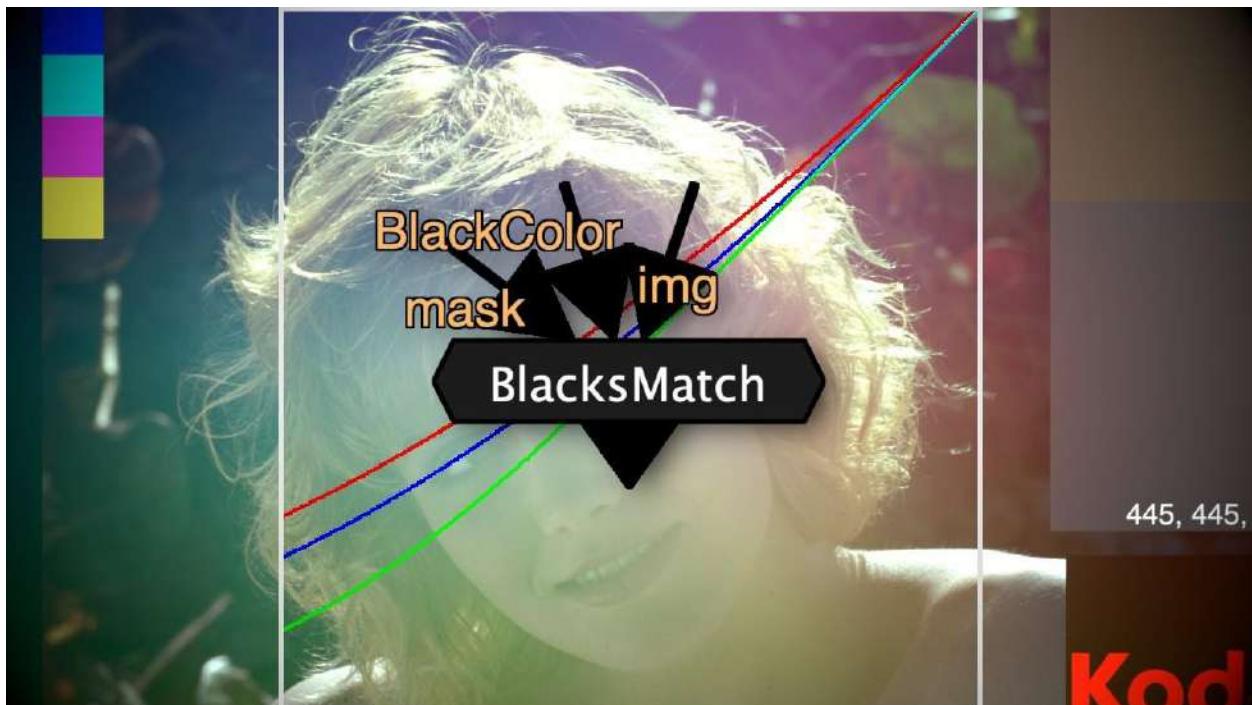
Instruction:

1. Connect renameChannels node to your script.
2. Select the oldLayer and insert the name of the newLayer. Basically you want to rename the old name with the new one.
3. Select the Channels you want to create. They depend from the oldLayer
4. Click on button "Create Copy and Remove Nodes"
5. Connect the nodes created to your script.

5. Color

BlacksMatch TL

Author: Tony Lyons - www.CompositingMentor.com



Full Tool Description Breakdown:

<https://compositingmentor.com/2019/06/30/blacksmatch/>

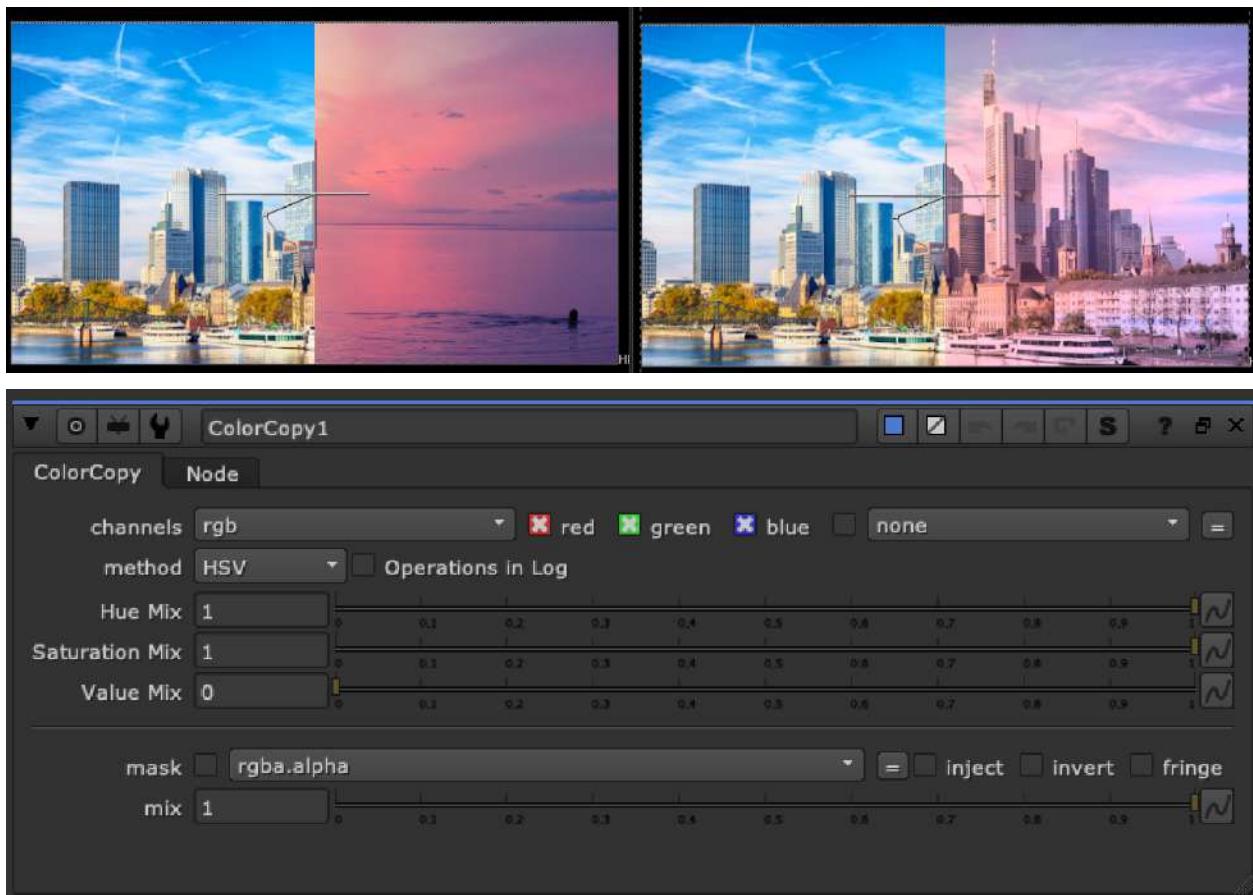
Nukepedia link: http://www.nukepedia.com/gizmos/colour/blacksmatch_20

Full tutorial/ Demo: <https://www.youtube.com/watch?v=Kw3bcsmkGuk>

This tool recreates a toe operation that's able to input an external image as it's black point and has controls for the multiply (the amount above the blackpoint that the operation stops affecting the midtones and highlights), and a control for the Gamma, or falloff, which is the bottom part of the curve and how it's blending with the blackpoint. You can toggle a preview overlay of a plot scanline and see how your blackpoint is affecting the rest of your image.

ColorCopy TL

Author: Tony Lyons - www.CompositingMentor.com



ColorCopy converts A and B images into HSV or HSL colorspace and mixes hue, saturation, and luminance (value) from image A to image B.

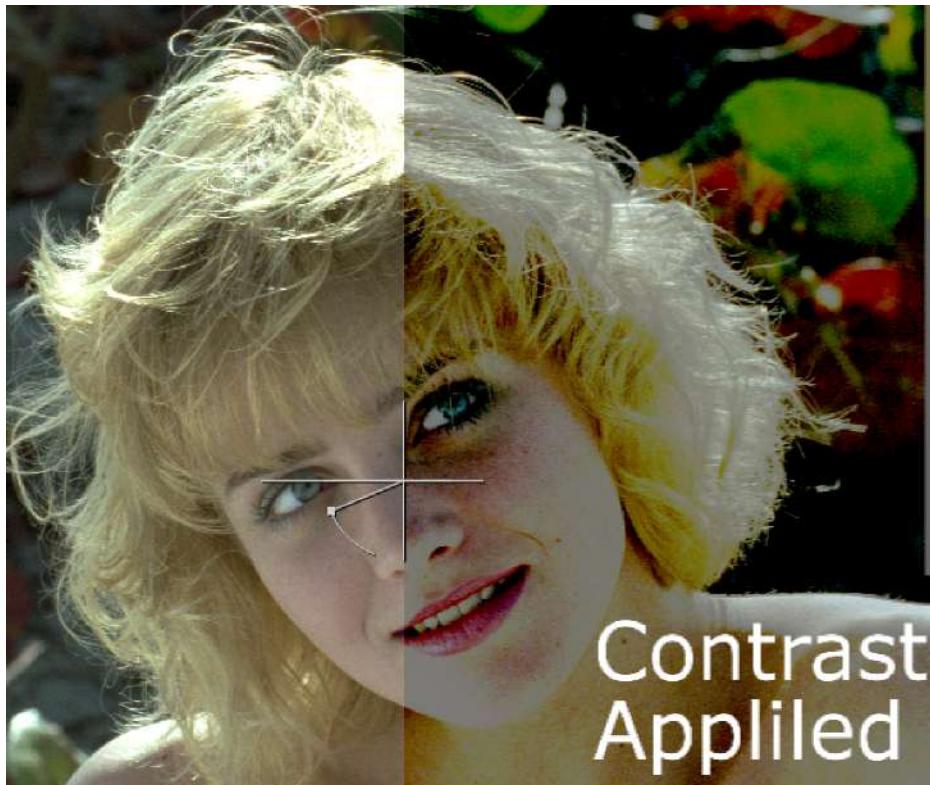
The other modes are a way of separating color and luminance from the image by dividing the original image by a desaturated version. The other methods are the methods found in the saturation node: rec709, average, maximum, etc.

When you are on HSV or HSL you have control over hue and saturation separately, and in the rest of the methods have just a single 'color mix' control (along with the luminance mix).

Can toggle Operations to be done in Log space.

Contrast TL

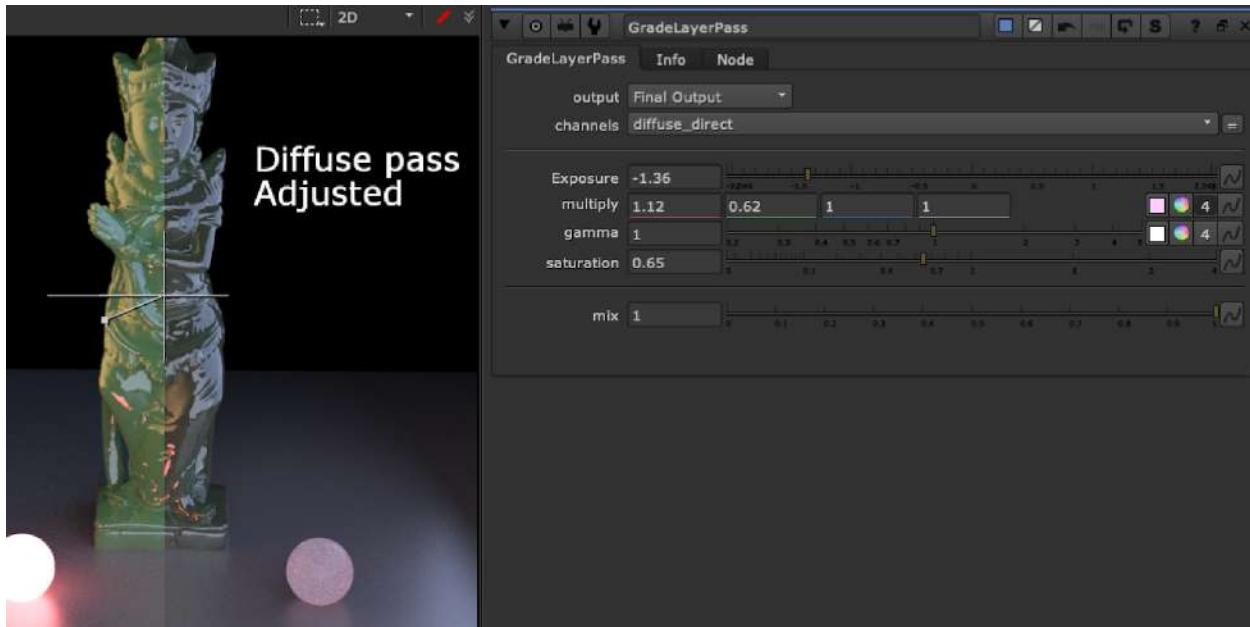
Author: Tony Lyons - www.CompositingMentor.com



Simple contrast tool with a pivot point controlling areas about and below the pivot separately.

GradeLayerPass TL

Author: Tony Lyons - www.CompositingMentor.com



Useful for grading CG AOVs. Choose AOV from the channels dropdown that you wish to grade. I've chosen the most common grade adjustments (also useful from a lighting artist point of view that translates well back to lighting application).

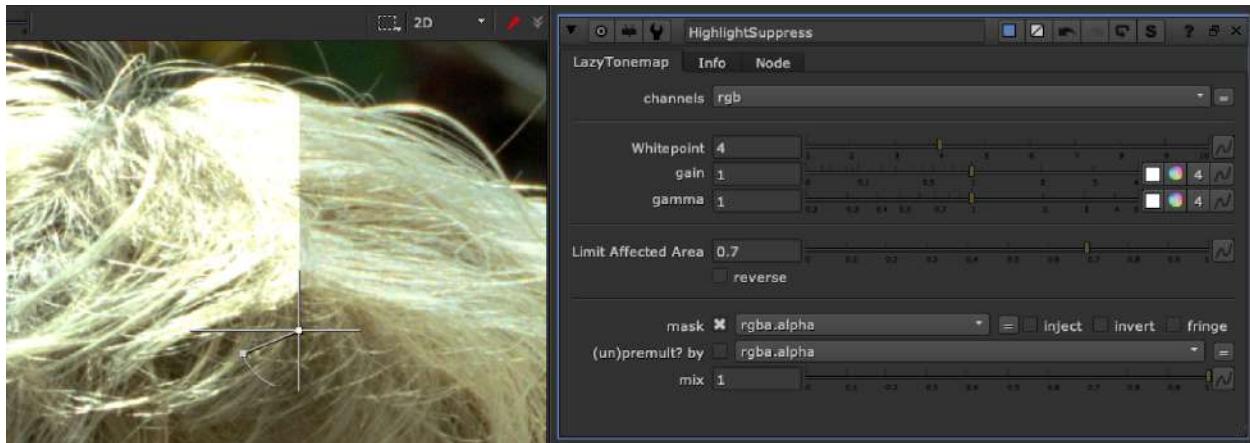
Exposure - for luminance, Multiply - for color, Gamma - midtones, Saturation.

This tool will minus the AOV layer from the beauty, makes adjustments to the AOV layer, and plus the changed layer back. It also injects the changed AOV layer back into the stream so if you shuffle it out afterwards it will reflect the changes made to the AOV layer.

HighlightSuppress TL

Author: iiro Harra - Originally Lazy_Tonemap

Modified and renamed by: Tony Lyons - www.CompositingMentor.com



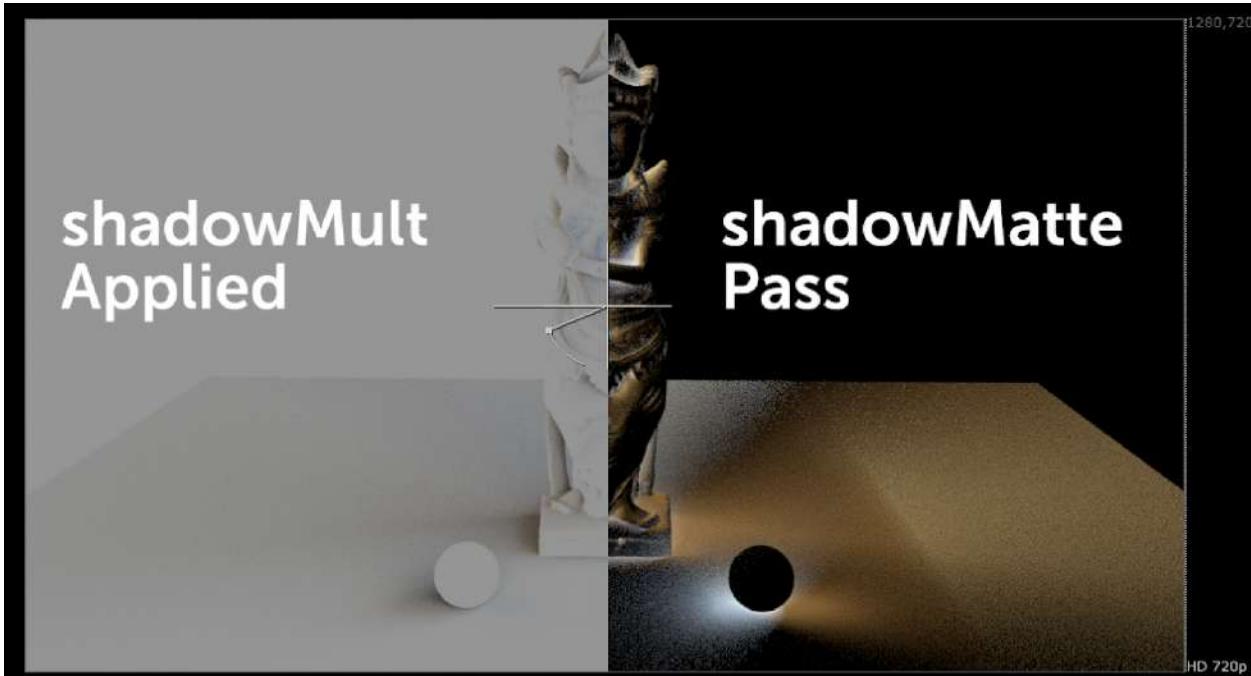
Original Nukpedia Link: http://www.nukpedia.com/gizmos/colour/lazy_tonemap

Uses expression to try and suppress highlights above 1 to a more aesthetic fall off, retaining color information. When the whitepoint is set to 1, the image will be the same as the original, higher the whitepoint, the stronger the effect. Gain and gamma can be used to compensate for the slight decontrasting that occurs on the image.

Limit Affected Area is a Luminance key on the original image. Setting to 1 will gradually ramp up the effect towards (and above) a value of 1 and helps preserve lows and midtones.

ShadowMult TL

Author: Tony Lyons - www.CompositingMentor.com



Arnold rendering and other renderers enable you to output a layer called “shadowMatte”. This can be a bit of a mysterious pass to figure out. Most artists just desaturate the image, shuffle to alpha and use it as a mask to multiply the plate or CG down to 0.

There is in fact color information in the shadowMatte pass. Each channel, red, green, blue, needs to be used as a mask to multiply the corresponding channel.

shadowMatte.red - used as mask to multiply red to 0

shadowMatte.green - used as mask to multiply green to 0

shadowMatte.blue - used as mask to multiply blue to 0

I made this tool to automatically apply this method quickly and effortlessly. Simple additional controls for multiply (in case you want to change color) and gamma.

WhiteBalance TL

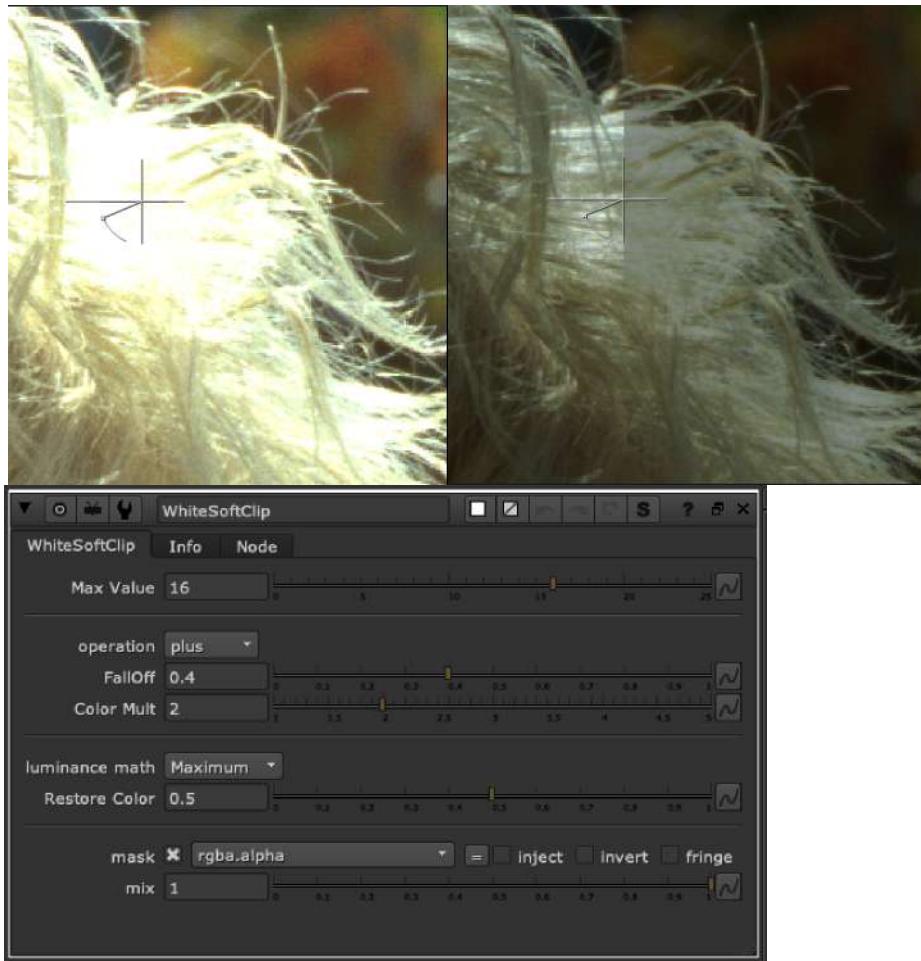
Author: Tony Lyons - www.CompositingMentor.com



Sample the 'white' or neutral area of the plate you wish to white balance. This tool preserves the overall luminance of the image. There is also a reverse option. Can be used to balance plates before greenscreen/bluescreen keying, and then reverse back after the keying/despill process to preserve original colors.

WhiteSoftClip TL

Author: Tony Lyons - www.CompositingMentor.com



This tool's aim is to better approach the softClip tool. There are many times where you want to set a max value amount for the shot, 16, 25, 50, whichever. Unfortunately, the softClip tool in nuke tends to clamp all channels at the top amount equally, which seems to break the ratio between the channels and lose the color of the highlights.

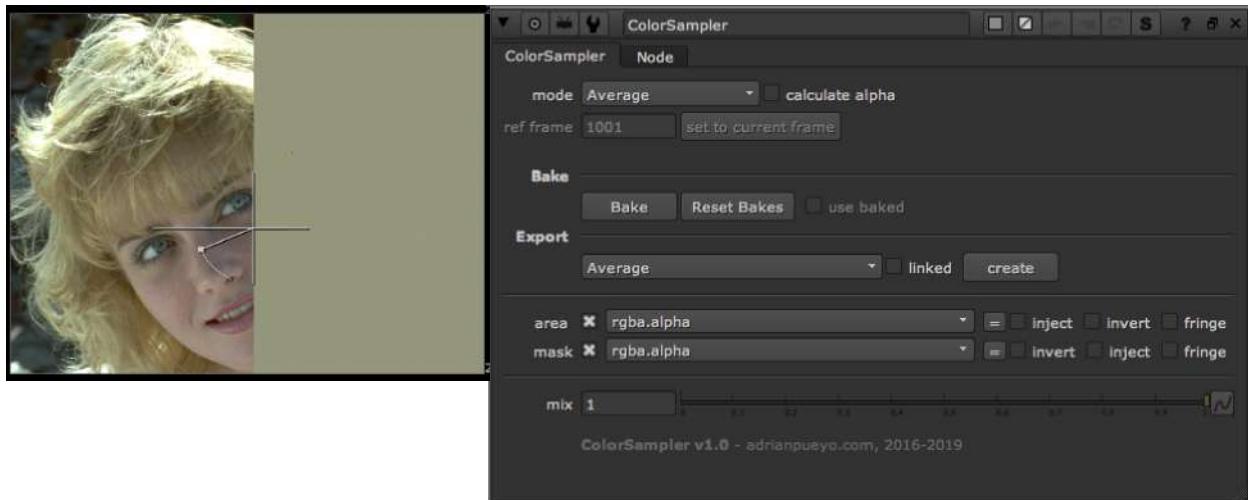
Set the max value you'd like to max your highlights out at.

Adjust the FallOff and Color Mult to adjust where the whiteclipping begins in the highlights.

The Restore Color slider restores more of the color by pushing the ratio between the colors a bit farther apart and maintaining the original colors. Default is set to .5

apColorSampler AP

Author: Adrian Pueyo - <http://www.adrianpueyo.com/>



Nukepedia Link: <http://www.nukepedia.com/blink/colour/colorampler>

Tutorial/Overview video: <https://vimeo.com/adrianpueyo/colorsampler>

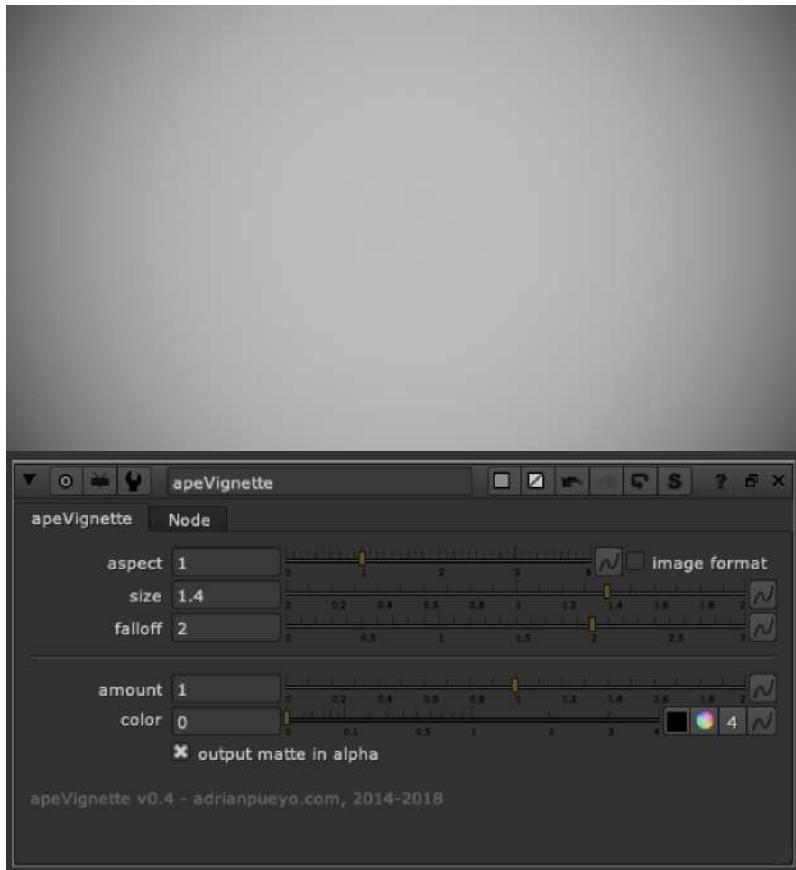
apColorSampler is a tool that calculates the average color of a target input (or the src image if there's no target input), weighted through the area input (or the whole frame if there's no area input). It can also calculate the maximum or minimum value over the area. Additionally, you can directly remove color flickering to an image, or apply it from a target.

You can think of ColorSampler as a live version of CurveTool with some additional features using the power of Blinkscript, where instead of being limited to a rectangle you can plug a roto to use for the sampling area... or a key... :)

Bake options available for framerange

apVignette AP

Author: Adrian Pueyo - <http://www.adrianpueyo.com/>



Simple and lightweight vignetting gizmo with controls for size, falloff, color
Option for outputting matte in the alpha channel.

GammaPlus MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Nukepedia Link: <http://www.nukepedia.com/gizmos/colour/gammaplus>

Demo Video: <https://youtu.be/bhjwnHXHPxQ>

Provide more options on Gamma math. Make contrast more easier in color integration. Break the boundary of default Gamma method.

Algorithm of Color Integration in Nuke :

[Nuke Math] Color Grade

<https://www.facebook.com/media/set/?vanity=MJTlab&set=a.665976340650857>

[Nuke Math] ColorCorrect - contrast

<https://www.facebook.com/media/set/?vanity=MJTlab&set=a.678965762685248>

[Nuke Math] Keyer - Luminance & Saturation

<https://www.facebook.com/media/set/?vanity=MJTlab&set=a.684167245498433>

[Nuke Math] HSV & sRGB

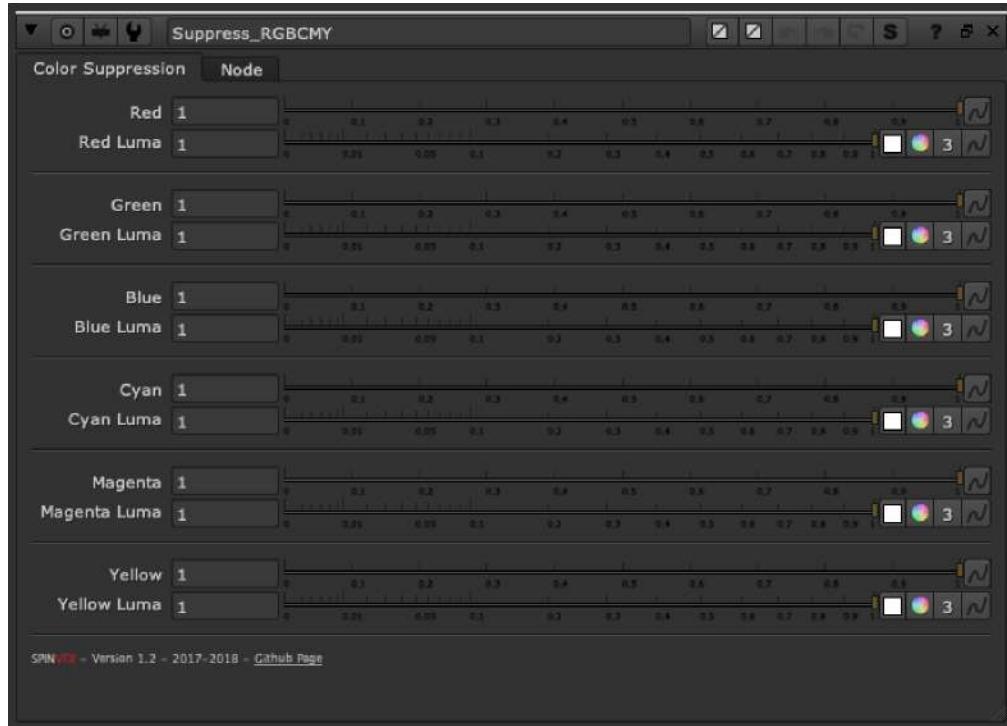
<https://www.facebook.com/media/set/?vanity=MJTlab&set=a.690958631485961>

[Nuke Math] Saturation in Nuke

<https://www.facebook.com/media/set?vanity=MJTlab&set=a.709545469627277>

Suppress_RGBCMY SPIN

Author: Spin VFX



Spin tools github: https://github.com/SpinVFX/spin_nuke_gizmos

Nukepedia link: http://www.nukepedia.com/gizmos/other/spin_nuke_gizmos-1

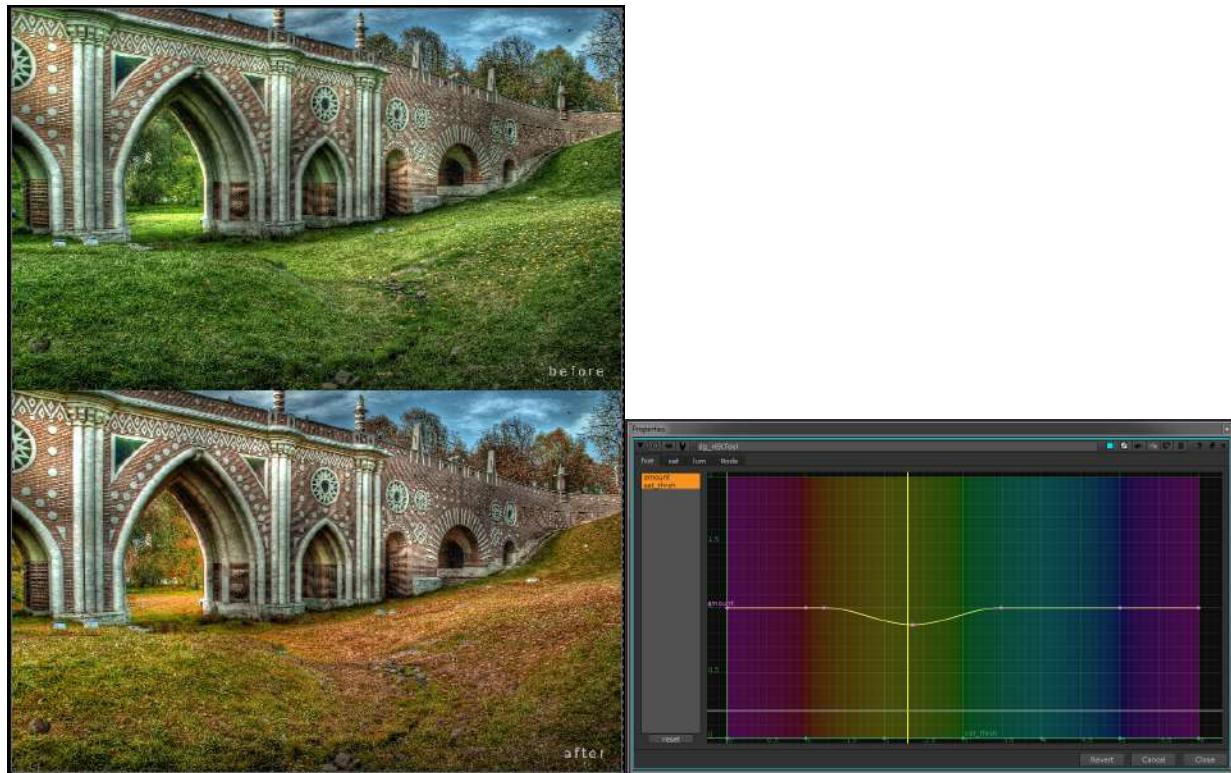
Vimeo Demo: <https://vimeo.com/381270956/f7399b6e1d> @ 9:15

Suppress (or boost) specific colors: Red, Green, Blue, Cyan, Magenta or Yellow.

HSL_Tool NKPD

Author: Den Gheiko

Website: www.gheiko.com



Nukepedia: http://www.nukepedia.com/gizmos/colour/dg_hsltool

A kind of DaVinci Resolve 'Hue Vs Hue', 'Hue Vs Sat' and 'Hue Vs Lum' color correction tool

Curve based adjusting hue, sat and value in a specific hue range

Adjust the curves, by default, the difference matte from the original image is stored in alpha, so can be used as a subtle keyer.

BiasedSaturation NKPD

Created by: Paul Raeburn

adjusted by: Tony Lyons - www.CompositingMentor.com



Nukepedia link: <http://www.nukepedia.com/gizmos/colour/biasedsaturation>

Simple tool for changing the saturation but toward a picked colour.

**Changes:

- Added channels dropdown and restore luminance slider.
- Default set to a bluish color and saturation and mix sliders set to 0.5

6.) Filter

X Tools XM

Author: Xavier Martin - <http://www.xaviermartinvfx.com/articles/>

These tools were made by Xavier Martin. I decided to place them all in 1 folder because I felt they are great and unique tools.

X Aton Volumetrics XM

Author: Xavier Martin - <http://www.xaviermartinvfx.com/articles/>



Nukepedia Link: http://www.nukepedia.com/blink/filter/x_aton

Website Documentation: http://www.xaviermartinvfx.com/x_aton/

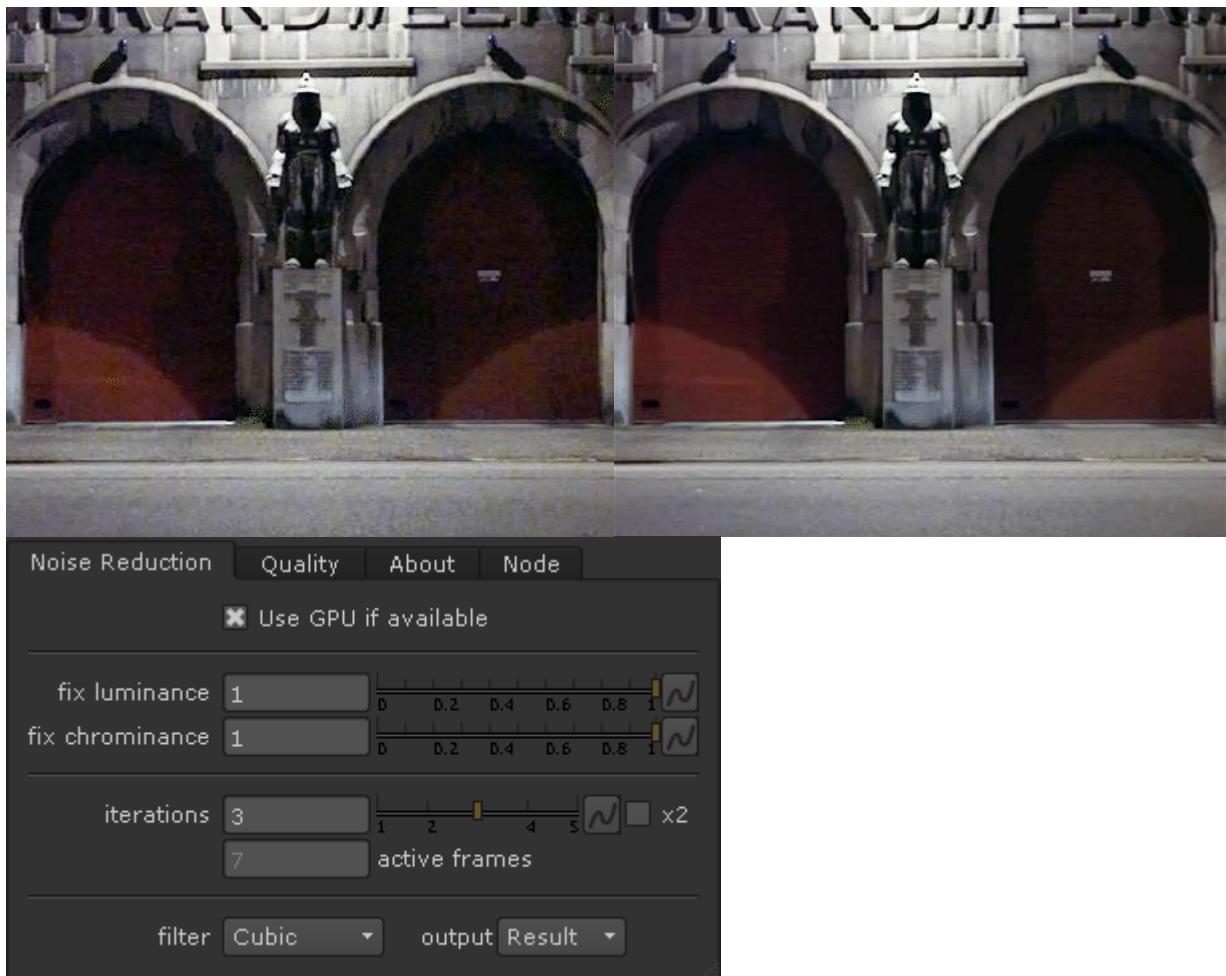
This tool is an advanced variation on the God Rays node. It creates volumetric lighting effects with more realism. It supports soft and diffused light rays created by area lights and volumetric lights. It supports Nuke 3D Cameras, Point Lights and Axis to have a better control of the direction of the illumination.

It supports Nuke 3D Cameras, Point Lights and Axis to have a better control of the direction of the illumination. The tool can also be used as a simple screen-space extruder, to create apparent 3D texts or extruded shapes. Now you can do WordArt from the 90s inside of Nuke!

All of this using the power of the GPU to speed up the calculations.

X_Denoise XM

Author: Xavier Martin - <http://www.xavermartinvfx.com/articles/>



Nukepedia link: http://www.nukepedia.com/gizmos/filter/x_denoise

Website Documentation: http://www.xavermartinvfx.com/x_denoise/

X_Denoise is a noise reduction gizmo that can be used to repair damaged or compressed footage. It does the same function of the Nuke Denoise, but using a different algorithm that can sometimes provide better results.

While most de-noisers try to work out which small pixels are susceptible to be noisy, the X_Denoise averages different frames in order to smooth the noise, making it invisible to the eye. The gizmo offers multiple settings to take control over how many frames are being used and how much detail is preserved.

X Sharpen XM

Author: Xavier Martin - <http://www.xaviermartinvfx.com/articles/>



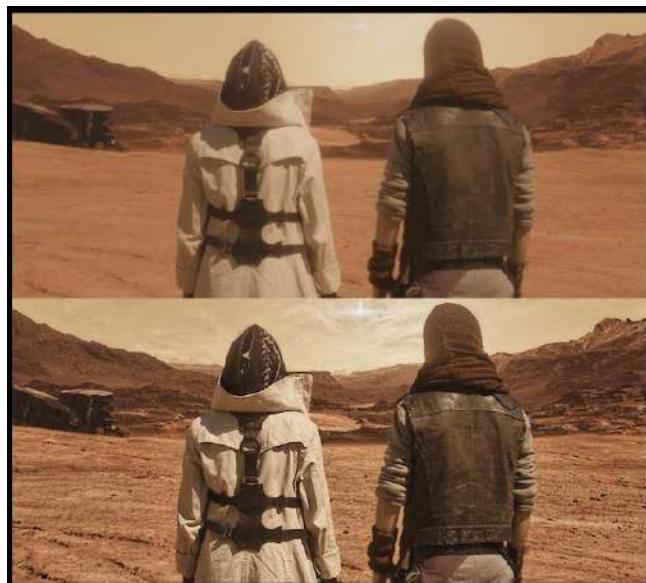
Nukepedia Link: http://www.nukepedia.com/blink/filter/x_sharpen-and-x_soften/

Advanced Sharpening and Softening filter.

This GPU accelerated filter preserves colours, avoid negative lobes and creates minimal ringing and banding on edges.

X Soften XM

Author: Xavier Martin - <http://www.xaviermartinvfx.com/articles/>

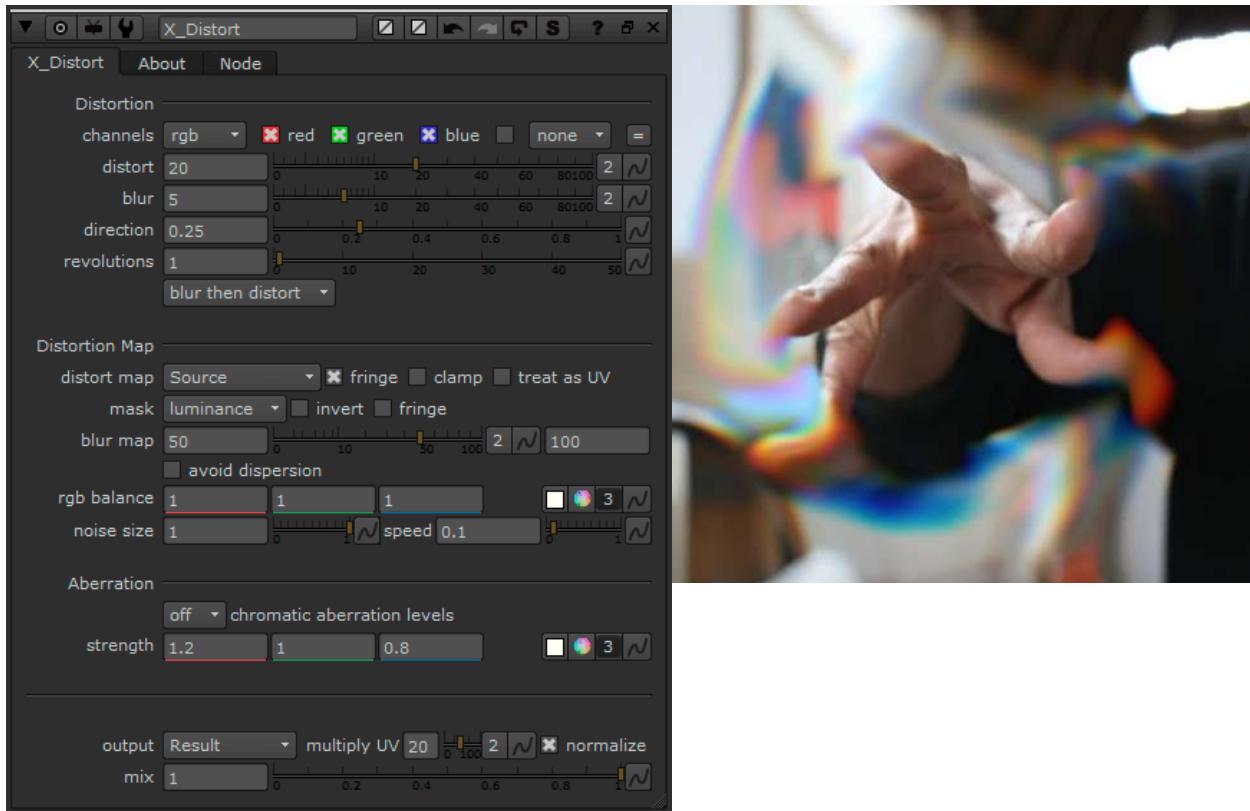


Nukepedia Link: http://www.nukepedia.com/blink/filter/x_sharpen-and-x_soften/

The softening filter is similar to a bilateral filter but much much faster

X_Distort XM

Author: Xavier Martin - <http://www.xavermartinvfx.com/articles/>



Nukepedia Link: http://www.nukepedia.com/gizmos/transform/x_distort

Website Documentation: http://www.xavermartinvfx.com/x_distort/

This gizmo allows you to distort images with control and flexibility. It is more customizable and easier to use than Nuke's IDistort. You don't need to copy any channels and you have many other controls to play with. You can blur the parts of the image which are being distorted to get a smoother result.

You can distort an image using its own channels, using another image or using an automatic noise. You can choose the detail of the deformation.

You can distort each color per separate, creating a realistic chromatic aberration effect. You can decide the quality of the effect in order to speed up render times.

X_Tesla XM

Author: Xavier Martin - <http://www.xaviermartinvfx.com/articles/>



Nukepedia Link: http://www.nukepedia.com/gizmos/draw/x_tesla

Website Documentation: http://www.xaviermartinvfx.com/x_tesla/

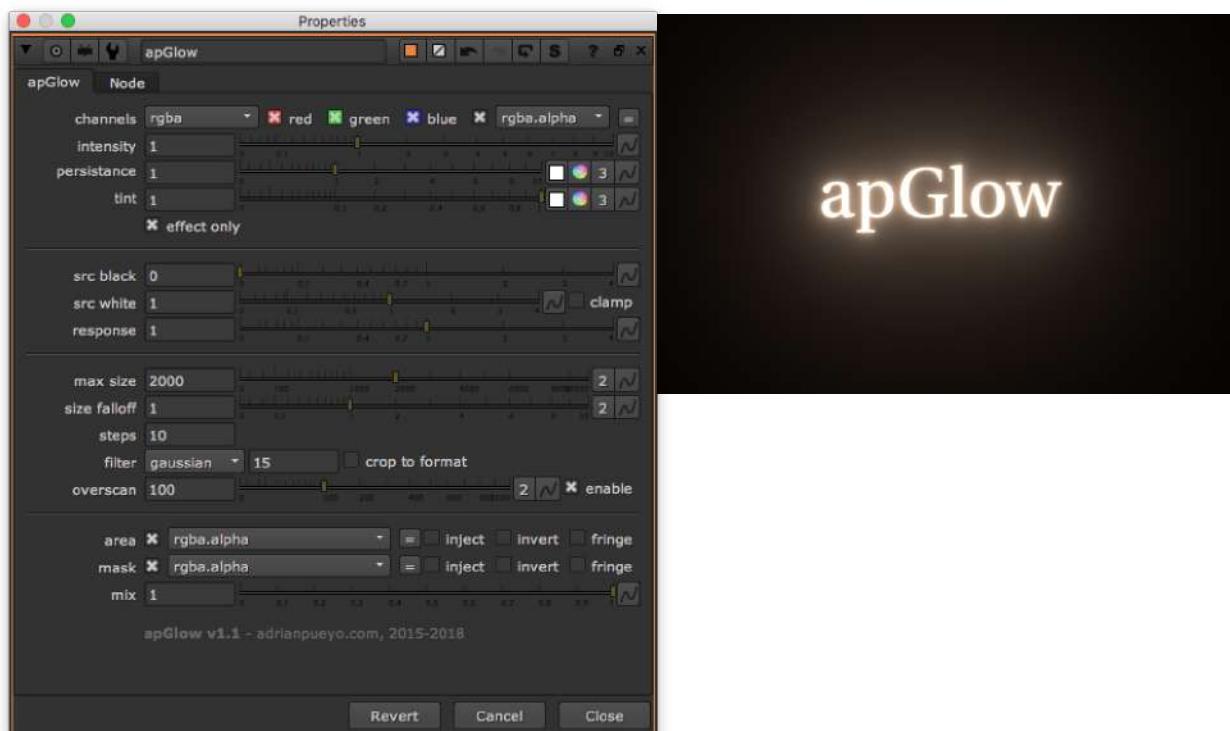
With this Gizmo you will be able to create lightning and electricity effects. Animated electric arcs will be procedurally created between two points.

The gizmo includes some realistic render option such as the temperature based chromatic aberration and glow, an advanced soften filter, an easy to use 2 colour system. Or you can just disable everything with a simple check box, that's also OK.

GloWS Menu

apGlow AP

Author: Adrian Pueyo - <http://www.adrianpueyo.com/>

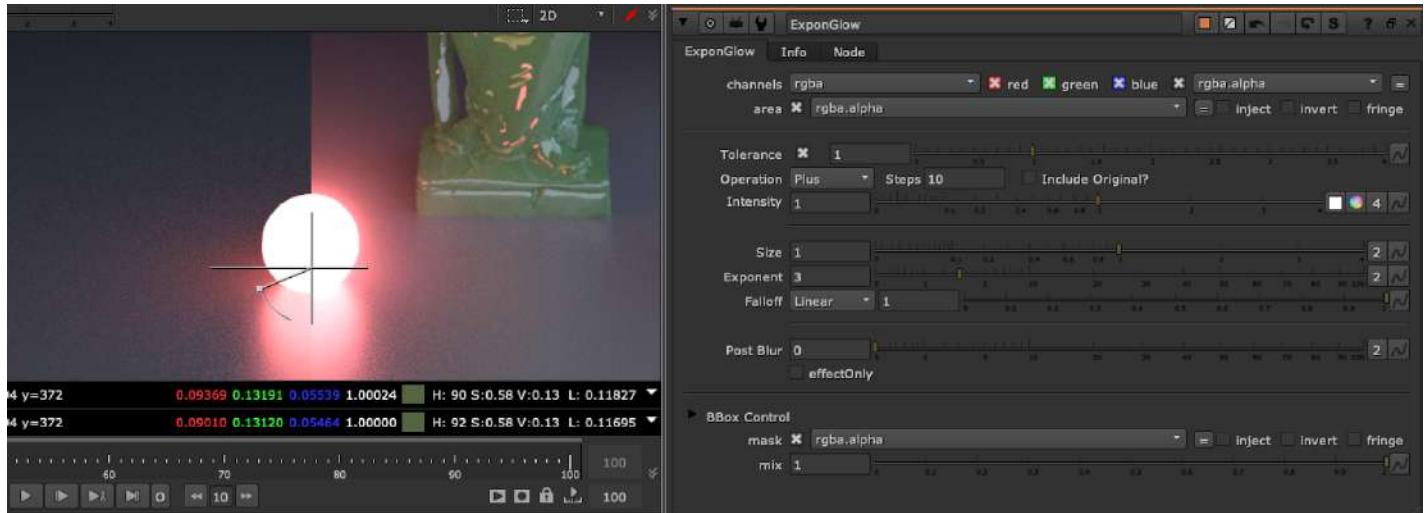


Nukepedia link: <http://www.nukepedia.com/gizmos/filter/apglow>

apGlow is a lightweight exponential glow effect with advanced control over the steps, falloff, distribution, color and other useful parameters.

ExponGlow TL

Author: Tony Lyons - www.CompositingMentor.com



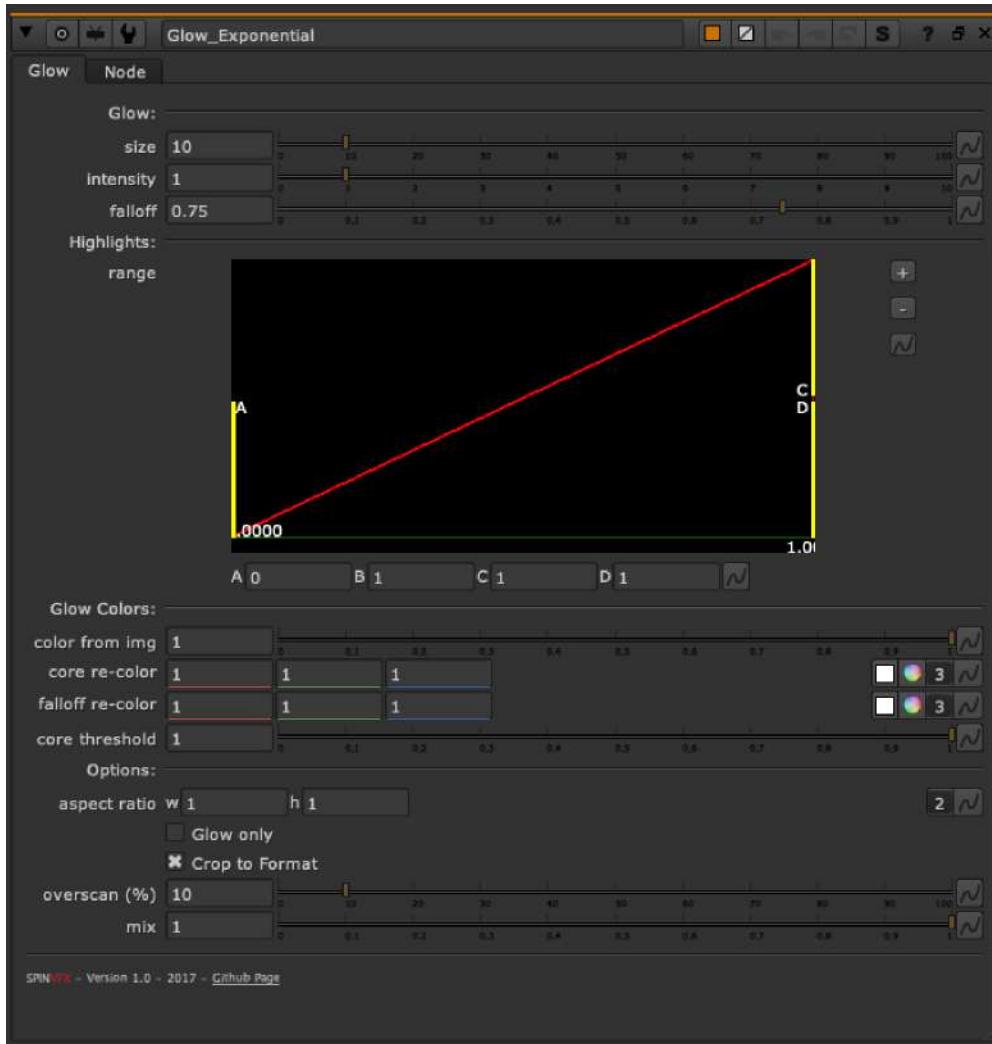
There are many Glow tools out there, Here are some of the features that makes this one unique:

- 1.) Iterable Blur steps, adds more or less blurs as you change the steps number
- 2.) Uses percentage blurs, meaning the blur ratio scales along with your format, so when changing from a 2K plate to a 4K plate, the Glow should look the same
- 3.) Different types of Merge operations to choose from:
Screen, Plus, Over, Hypot, Average, Max, Min (Can include original image in Merge)
- 4.) Different type of falloff to choose from, similar to ramp or roto falloff:
Linear, pLinear, Smooth, Smooth0, Smooth1
- 5.) Tolerance option (luma key on input) and Area mask input, so you can use mattes/rotos to isolate which part of the image glows
- 6.) BBox optimization, which has a safe BBox mode that will stop the bbox from growing 10% beyond the format size (or input BBox size, whichever is bigger). You can adjust this amount, or change to pixels instead of percent. And final BBox adjustment, so you can further grow/shrink the final bbox (in percent or pixels).

The focus was on an exponential Glow that has a lot of control and lookDev options in the type of falloff, size, and amount, while still paying attention to Bounding Box size and calculation time.

Glow_Exponential SPIN

Author: SPIN FX



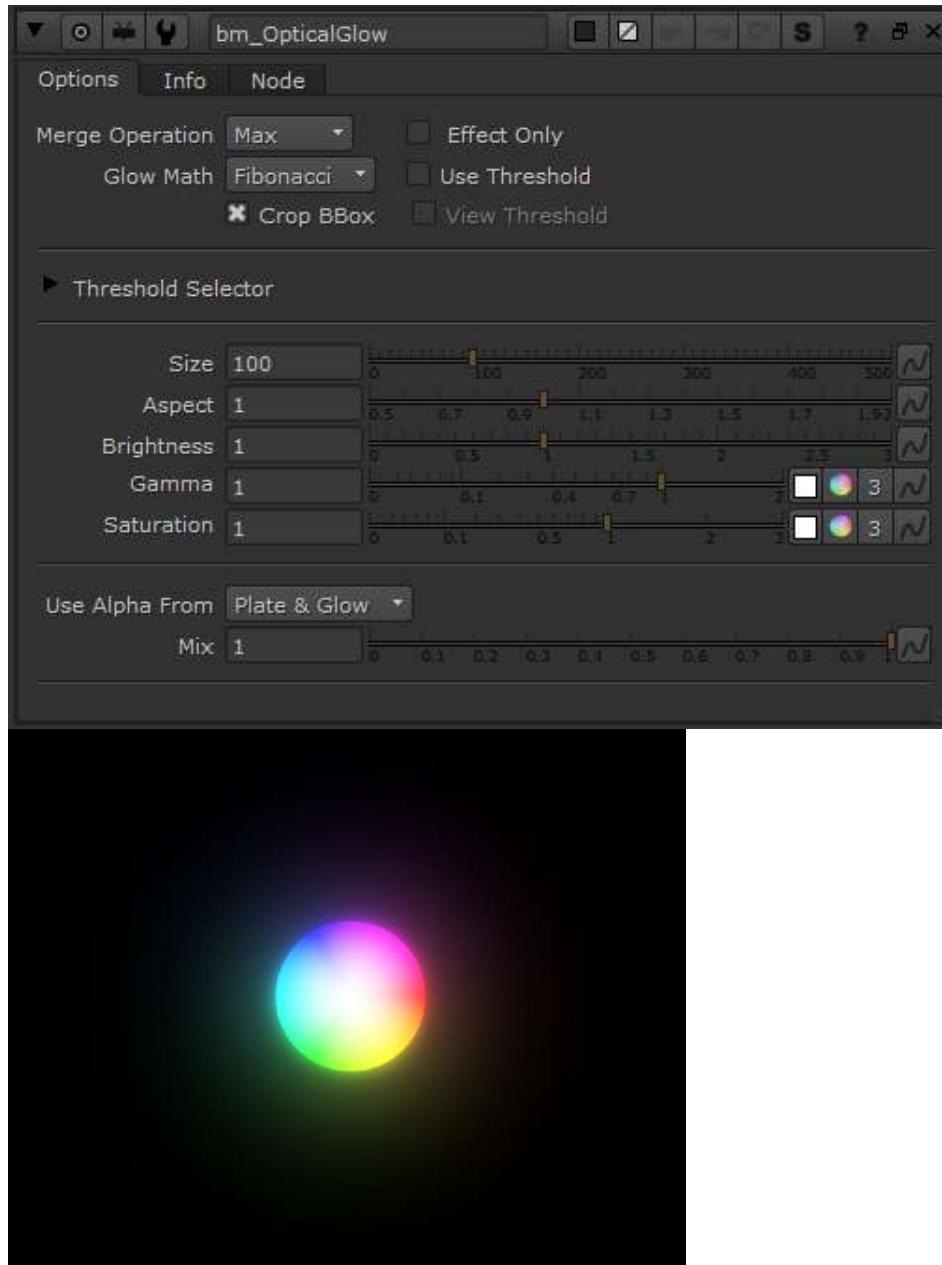
Github link: https://github.com/SpinVFX/spin_nuke_gizmos

Vimeo link: <https://vimeo.com/381270956/f7399b6e1d> @ 23:30

Exponential Glow node, with options to recolor and adjust falloff.

Optical Glow BM

Author: Ben McEwan - <https://benmcewan.com/blog/>



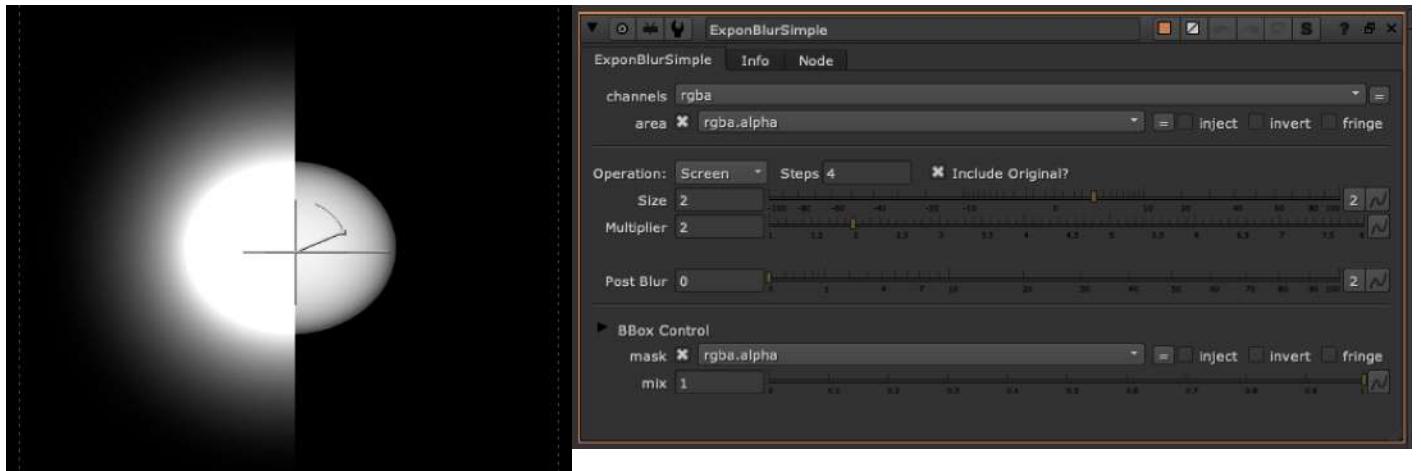
Link to Ben's Website: <https://benmcewan.com/nukeTools.html>

Adds exponentially-increasing blurs together to produce a more optically-correct, natural glow.

Blurs Menu

ExponBlurSimple TL

Author: Tony Lyons - www.CompositingMentor.com



Simple exponential blur iterating steps feature. Most often used with rotos/Mattes
Based on Luma Pictures ExponBlur gizmo.

http://www.nukedatabase.com/gizmos/filter/l_exponblur

Set your blur size and multiplier (exponent), and steps (iterations, # of blurs)

Different types of Merge operations to choose from:

Screen, Plus, Over, Hypot, Average, Max, Min (Can include original image in Merge)

Has a clamp and post blur options.

If you go negative with the size, the matet will blur inwards (invert, blur, invert back). This can be used for softly eroding/blurring mattes and alpha edges

BBox optimization, which has a safe BBox mode that will stop the bbox from growing 10% beyond the format size (or input BBox size, whichever is bigger). You can adjust this amount, or change to pixels instead of percent. And final BBox adjustment, so you can further grow/shrink the final bbox (in percent or pixels).

Directional Blur TL

Author: Tony Lyons - www.CompositingMentor.com



Nukepedia Link: <http://www.nukepedia.com/gizmos/filter/directionalblur>

Select the rotation angle and size of the blur. Choose between blur and defocus. Has a perpendicular blur that blurs in the perpendicular direction to the angle chosen.

Some helpful options for managing your BBox.

Has channels, mask, mix, etc

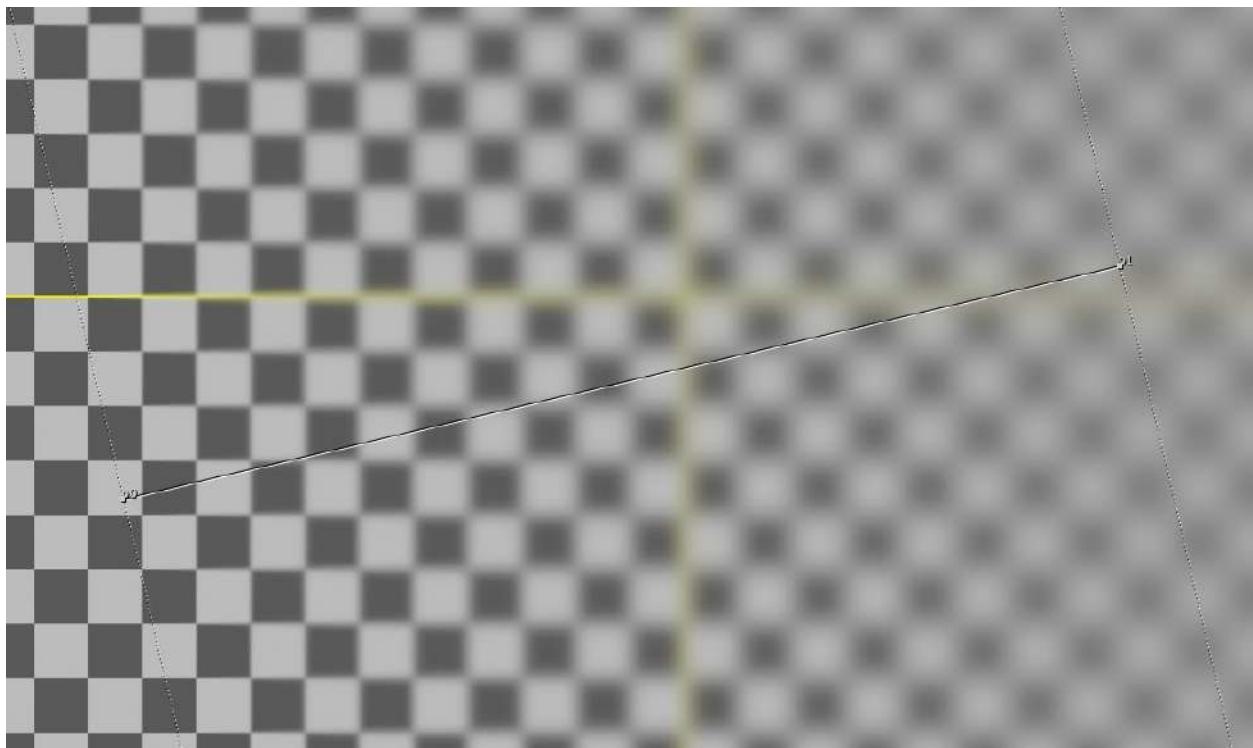
View the Demo here on youtube:

<https://youtu.be/BrioyN9YMA8>

IBlur NKPD

Author: moritz eiche

Modified by Tony Lyons www.CompositingMentor.com



Nukepedia Link: <http://www.nukepedia.com/gizmos/filter/iblur>

With this gizmo you get a smoothly ramped blur on your image, based on the matte input. It works like 'iBlur'/'iDefocus' in Shake.

It's faster and easier than 'ZBlur', also you can choose between 'blur' and 'defocus'.

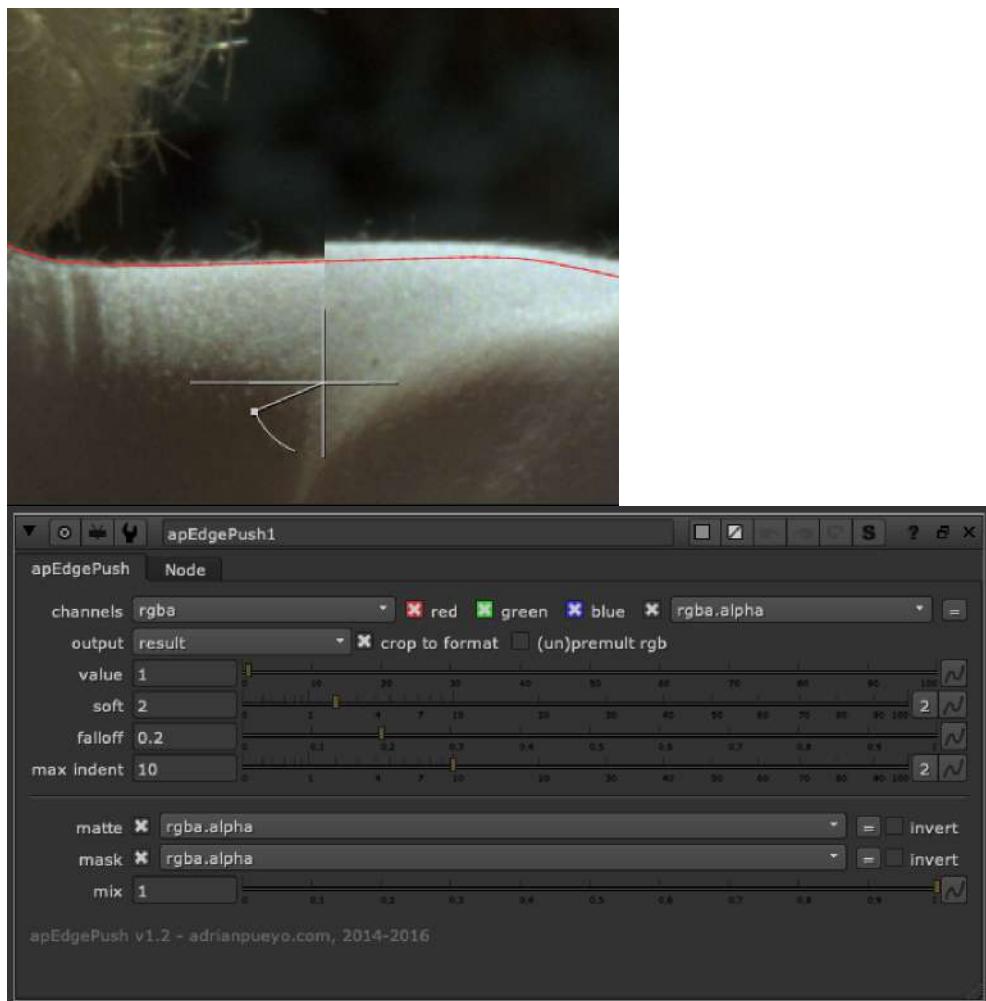
Update by Tony Lyons:

Allowed for x,y individual size options. Added some better BBox management.

Edges Menu

apEdgePush AP

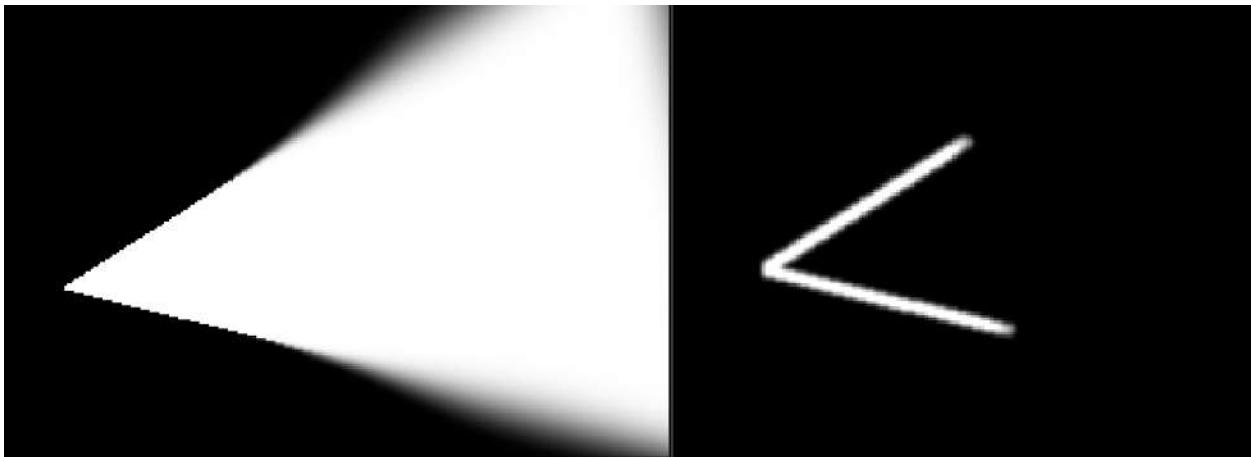
Author: Adrian Pueyo - <http://www.adrianpueyo.com/>



apEdgePush is a vector distort edge warp that warps the edge to try and get rid of fringe colors. By default looks for alpha in the img input but you can plug in a custom matte to the matte input. Switch channels to rgb and warp before the premult in order to warp the image “within” the premultiplied alpha region.

EdgeDetectAlias TL

Author: Tony Lyons - www.CompositingMentor.com



Analyzes your alpha for aliased edges and gives you an edge detect for aliased areas. You can then use an iBlur or blur with mask to introduce blurs to help smooth out edges. Good for CG or DMP or deep combines where you get some harsh edges.

AntiAliasFilter AG

Author: Andrea Geremia - www.andreageremia.it/tutorial.html



Nukepedia link: <http://www.nukepedia.com/gizmos/filter/antialiasing-filter>

Increase Antialiasing filter

Decrease Antialiasing

Instructions:

Select channels. You can just apply to the edge in your alpha channel or also to the RGB

Decide the operation (increase or reduce)

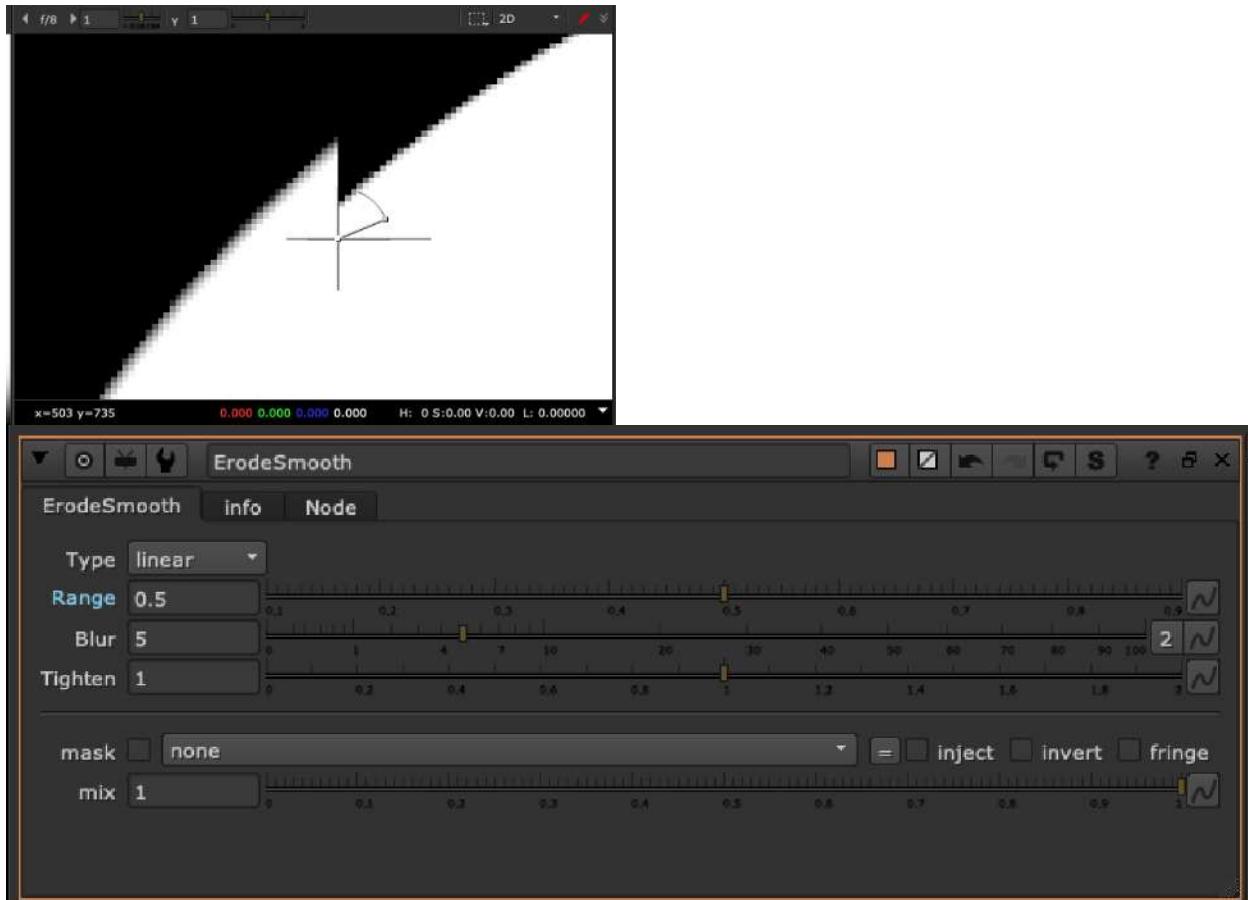
Move slider to select size of the filter

Use gamma for a better tuning

This gizmo uses a 3x3 Matrix

ErodeSmooth TL

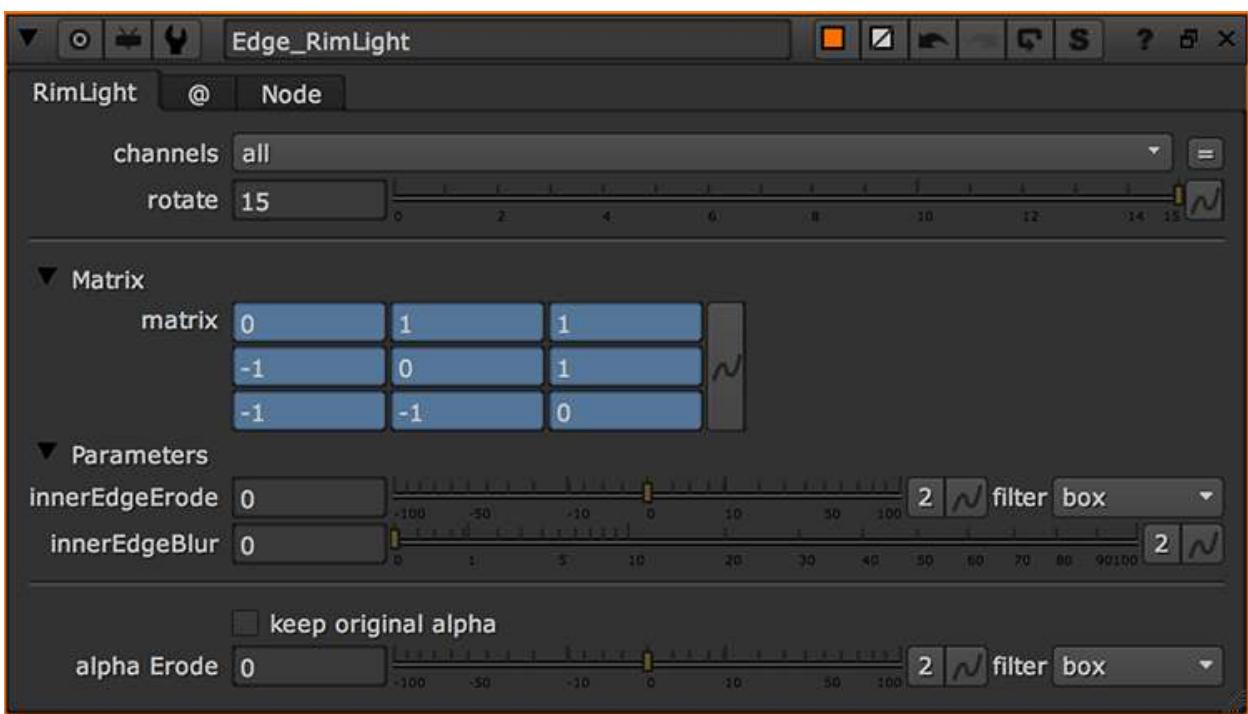
Author: Tony Lyons - www.CompositingMentor.com



This erode uses a method of blurring the alpha, then uses colorlookup node to tighten the edge back down. The range slides along the full area of the blurred region. Best when used with tight rotos / edges without much blur or falloff.

Edge_RimLight AG

Author: Andrea Geremia - www.andreageremia.it/tutorial.html



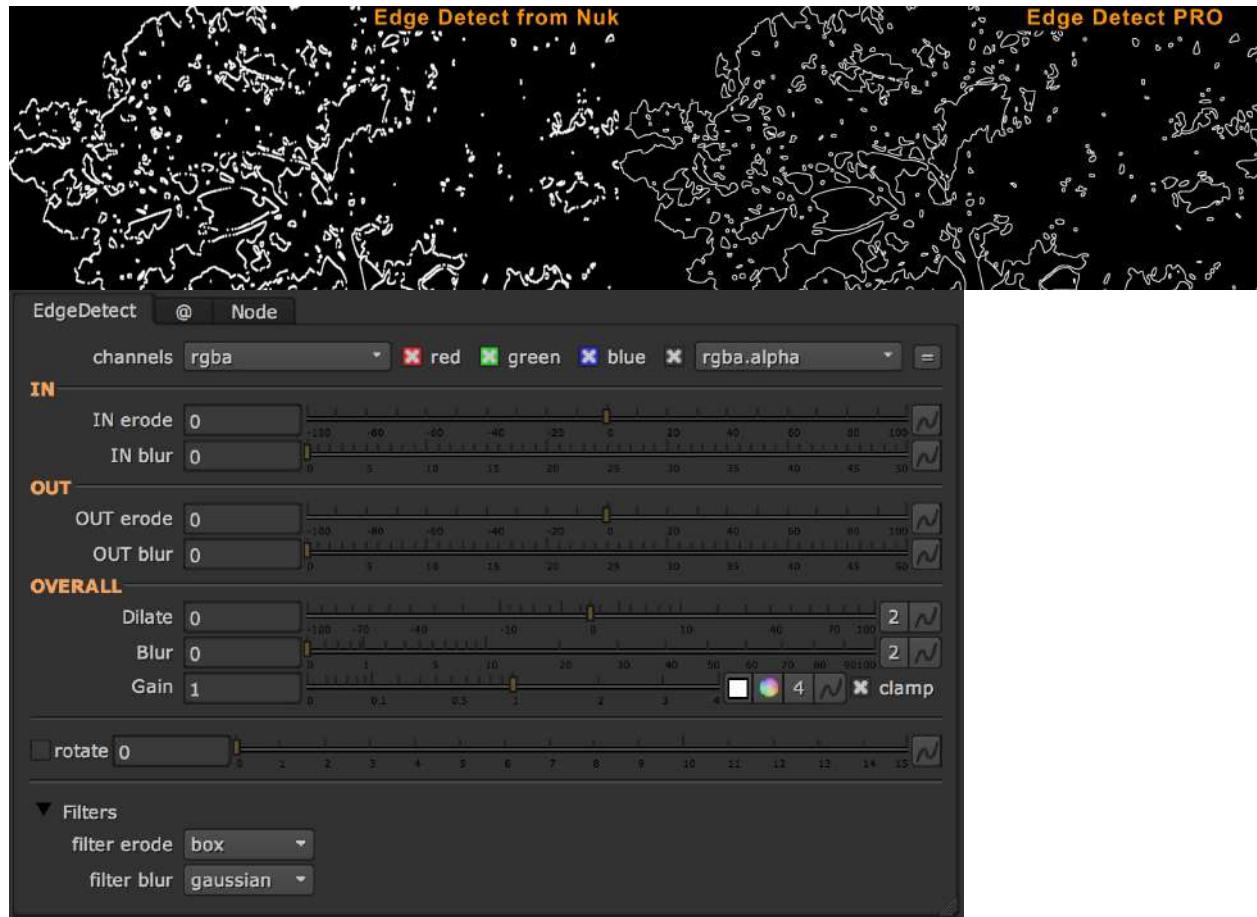
Nukepedia Link: <http://www.nukepedia.com/gizmos/filter/edge-rim-light>

With this tool you can create a quick mask for your Rim Light. Move the slider rotate and that's it!

Use the Parameters to modify the size and the softness of the edge.

EdgeDetectPRO AG

Author: Andea Geremia



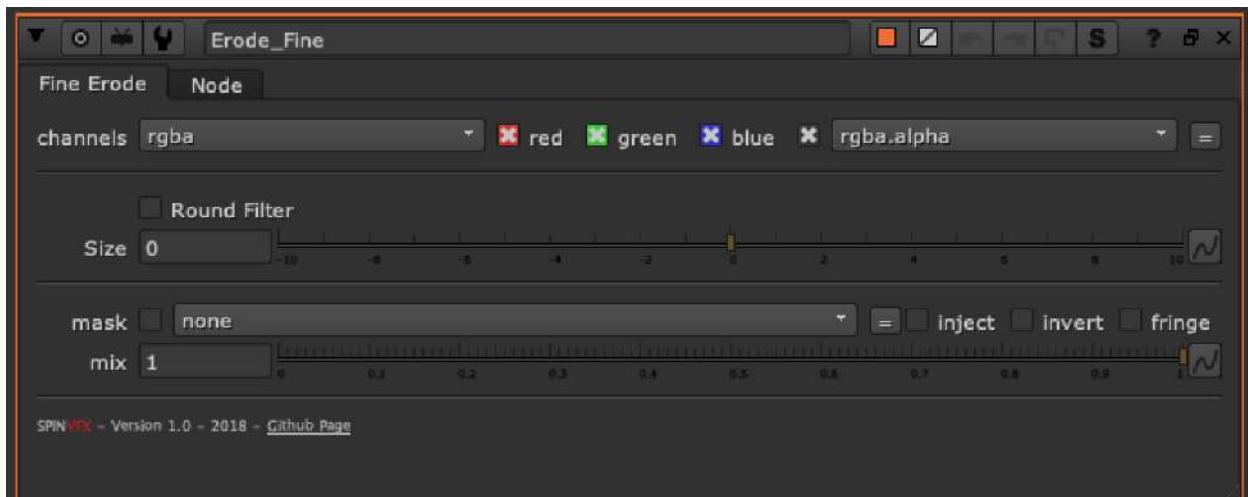
Nukepedia Link: <http://www.nukepedia.com/gizmos/filter/edgedetect-pro>

If you want to go deeper with this topic, visit this tutorial:
http://www.andreageremia.it/tutorial_edge_rim_light.html

EdgeDetect PRO is an evolution of the classic node from Nuke. Here you have more options to get a better result with thinner edge reveal.

Erode Fine SPIN

Author: Spin FX



Erode an image with fine controls, as opposed to Nuke's default erode node which can only erode full pixels.

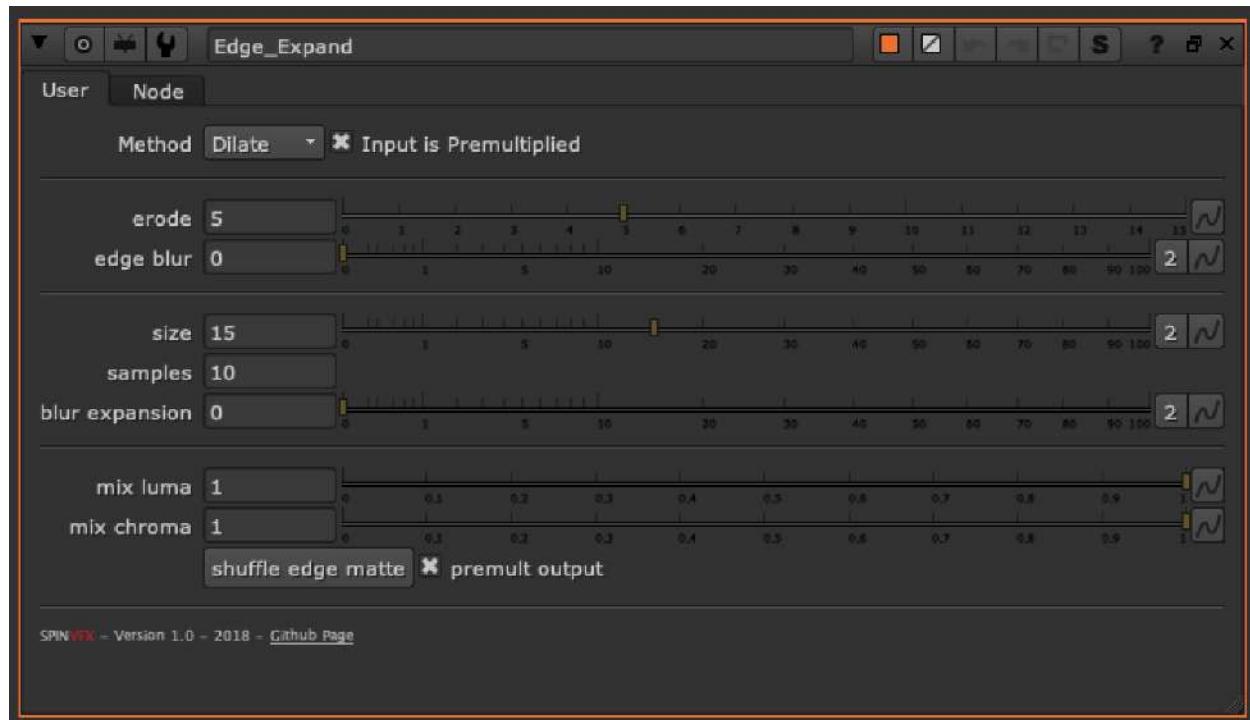
Nukepedia Link: http://www.nukepedia.com/gizmos/other/spin_nuke_gizmos-1

Github Link: https://github.com/SpinVFX/spin_nuke_gizmos

Demo video: <https://vimeo.com/381270956/f7399b6e1d> @ 31:08

Edge_Expand SPIN

Author: Spin FX



Nukepedia Link: http://www.nukepedia.com/gizmos/other/spin_nuke_gizmos-1

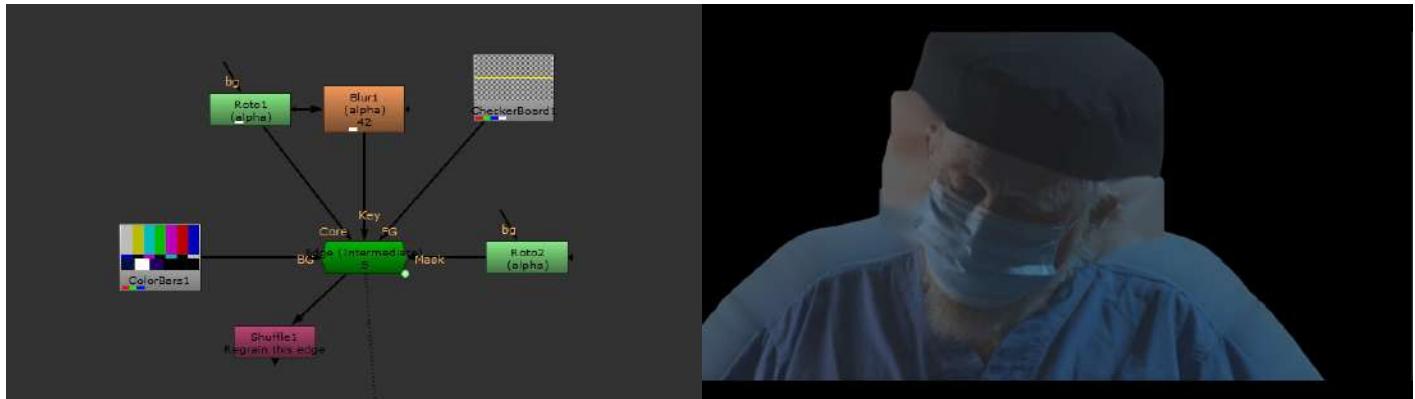
Github Link: https://github.com/SpinVFX/spin_nuke_gizmos

Demo video: <https://vimeo.com/381270956/f7399b6e1d> @ 24:55

Expand edges to fix fringing on keys.

Edge RB

Author: Rob Bannister - <http://www.bannisterpost.com/blog/>



Nukepedia Link: <http://www.nukepedia.com/gizmos/filter/edge>

More information can be found on Rob's site at <http://www.bannisterpost.com> and you can view the video tutorials his my channel here <https://youtu.be/E3MV282dEbs> and the new updated video here https://youtu.be/QXOMe_7fAmw

Edge allows you to extend the proper color out over transparent edges or areas with a lot of motionblur without eroding away the nice detail. This edge extend has some more advanced features like splitting the core matte and final key which allows you to work on motion blur without affecting small details like hair or a swinging rope for example. There is also more control over how you blend the core detail to the extended edge color. This was shown to me years ago by an old comp sup I had and is the same as the edge extend by Frank Reuter. This is my version of it that has evolved a little bit over the years.

INPUTS

There are multiple inputs to Edge but you can just plug in your FG with alpha if you want to use it simplified. Separating Core and Softer keys or using roto with and without motionblur has advantages in certain situations. ie: If you have a thin object this will help maintain solid alpha where you need it.

FG – Foreground plate with Spill suppression applied

Core – This is like a core matte, it helps identify the leading edge and where the edge tool should start from. It is important to add this core so that small details are not lost like a thin rope or hair.

Final Key – This is your Key with fine hair details and transparent motion blurred edges. For Roto you want to have motion blur turned on.

BG – This is where your Background goes if you are using the node as Final Result

Mask – Currently this only works if you are using Edge in Final result.

PROPERTIES – EXTEND

The first setting you will see in the properties panel after channels is the **Operation** selector. Here you can choose if you want to use the Edge Tool to merge together your foreground and background giving you the Final Result or if you would like the Intermediate Result which is unpremultiplied and not merged over the background. The Intermediate Result can be helpful when tweaking parameters like the slice start and size. Next to the operation drop down is a check box that allows you to preview your edge matte that you can use via the channels created in the edge node or by creating a shuffle using the button at the bottom.

extend chroma only – This selection will extend only Chroma information by swapping back in the original luminance information.

exp – The exponential check box will add a variable to the duplicated blurs that increases as they move outwards from the original edge. This might help if you have a large distance to extend.

premultiplied – Select this if your footage is already premultiplied to remove dark edges in the extension.

clamp alpha – Select this if your alpha has values below 0 or above 1.

slice start – Determines where you would like to start bleeding out the color from the inside of your key. It does this with a series of unpremult, blur, premult operations instead of a standard erode.

slice iterations and slice width – Sets how large the edge with the color bleeding will be from the edge grow start or core matte and how many times this will be duplicated. The smaller the width the more detail is preserved. This should be just big enough to reach the edge of the largest areas of motionblur.

edge blend – Helps to soften the transition from the original to the new edge. It tends to erode the bleeding color back inwards if you use high values.

edge blur – This helps soften the transition bringing back original detail. This can sometimes bring back some lighting information instead of having it completely replaced.

edge smooth – This one will help with extremely large motionblur and completely takes away detail. This should not be used in most cases but if you do use it sparingly.

create edge matte – This will generate a shuffle node that isolates the areas of extension which you can use to regrain your edges.

Properties - Extend

For all of the edge blending parameters you should use very low values like 1.6, 1.2 etc

PROPERTIES – BLENDING

This tab will help you blend your foreground edges with the background color. It is based off the edge extension in the first tab but you can modify the edge properties to get your desired look.

In the Edge Blending tab you can enable this feature and preview the edge you are using. This helps if you want to modify the edge used for blending with your BG.

operation – This is the merge operation you want to use for blending. It is set to average by default but you can use anything you like ie: plus, max, min.

expand edge – You can expand or contract the edge created in the extend tab.

soften edge – This blur will help soften the transition of the inside blending edge.

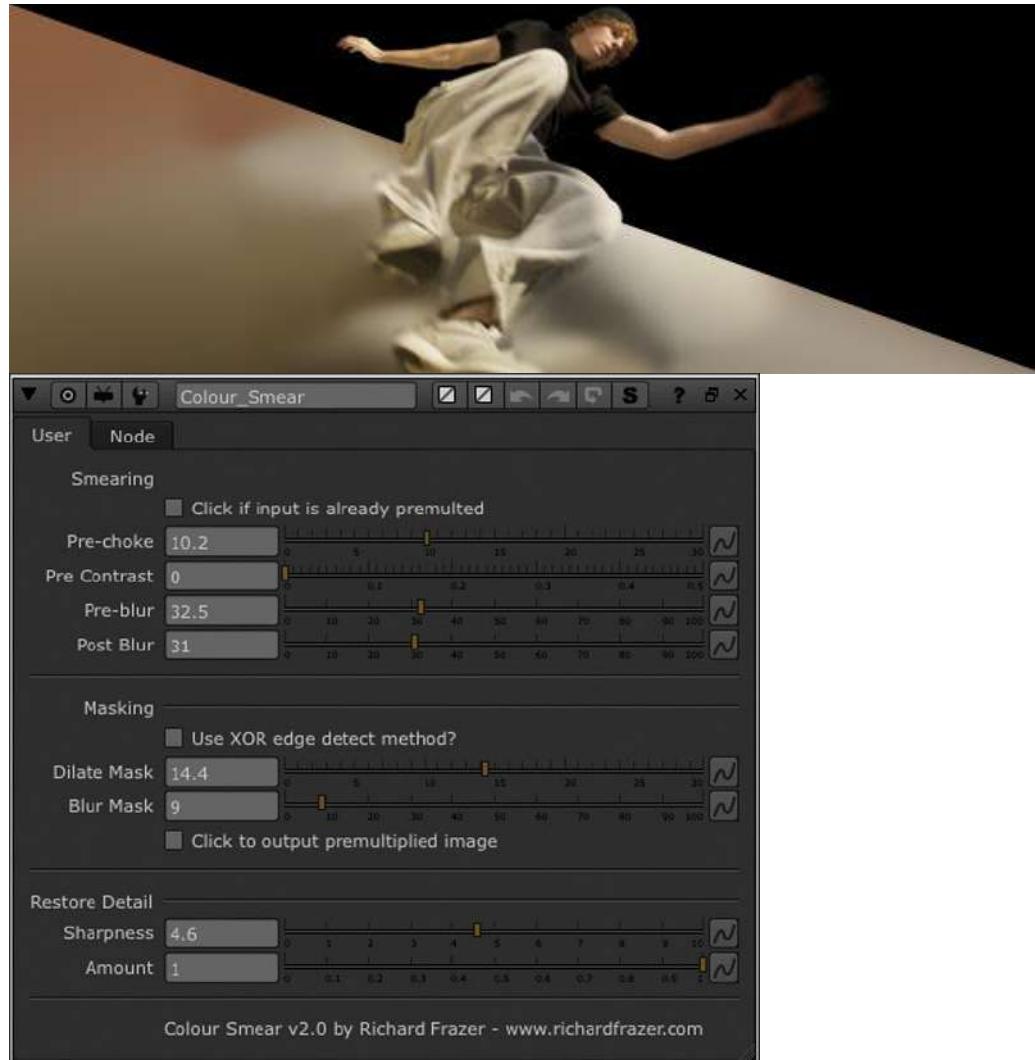
mix fg luma – At a value of 0 this will only blend the chroma information but you can mix this back to include luminance info as well.

adjust bg color – Here you can modify the color and saturation of the bg being used for blending. This will only affect the blending, not the actual bg.

mask bg luma – This section will allow you to preview and isolate the bright areas of the bg to wrap around your foreground. Think of this as a way to have lights wrap around your foreground in car comps.

ColourSmear NKPD

Author: Richard Frazer



Github link:

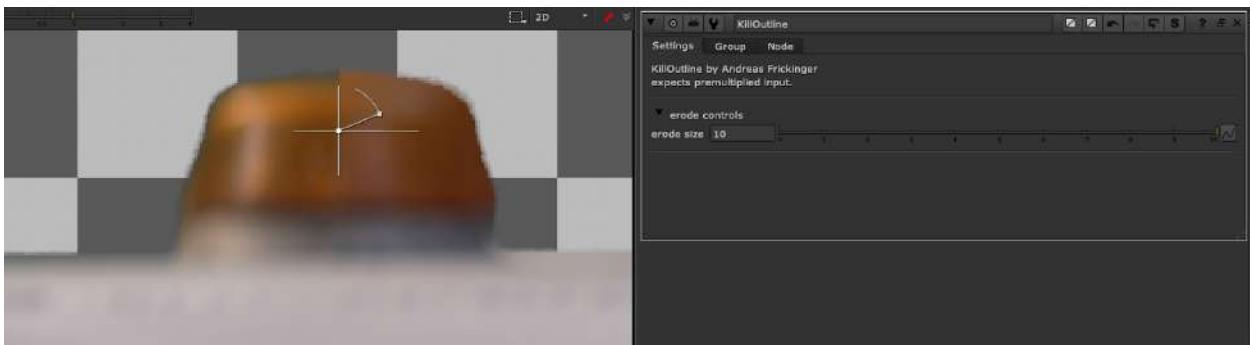
<https://github.com/RichFrazer/colour-smear-for-Nuke/blob/master/colour-smear.nk>

Smear out the edge colour of your A plate to create better soft edges. Works by blurring and un-premultiplying your image.

I have seen different artists do similar techniques in a number of ways, but this is my take on it (the EdgeExtend gizmo is a simpler version – this one combines the edge erosion aspect and produces softer results). This technique is not application specific, but I am demonstrating it here in Nuke (although I doubt you could do this in After Effects as you need to be able to manually control pre-multiplication).

KillOutline NKPD

Author: Andreas Frickinger

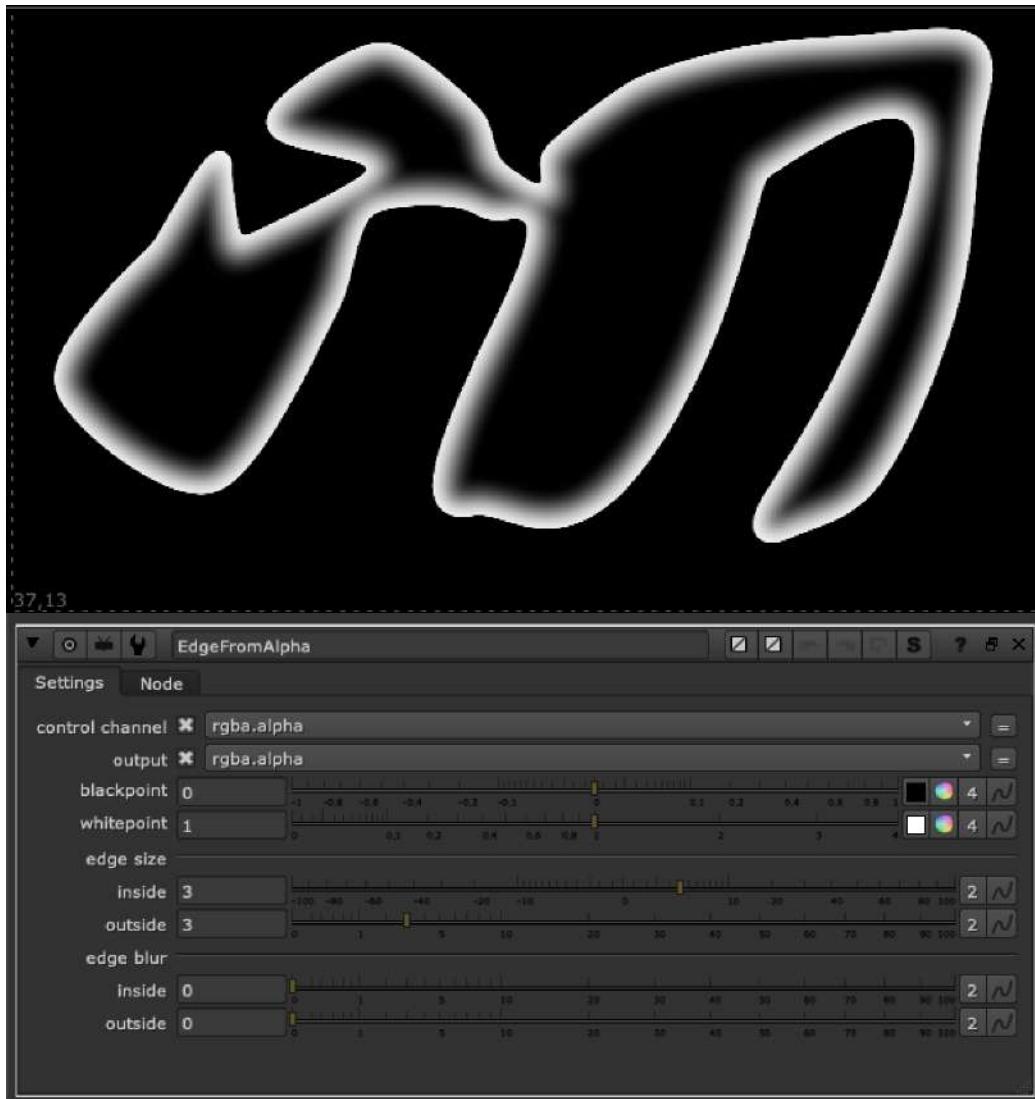


Nukepedia link: <http://www.nukepedia.com/gizmos/keyer/killoutline/>

Erodes/Expands rgb edges of keyed image image to get rid of unwanted outlines. Includes fine tuning for edge treatment. Based on Frank Rueter's Edge Extend.

EdgeFromAlpha FR

Author: Frank Rueter - www.ohufx.com



Nukepedia link: <http://www.nukepedia.com/gizmos/filter/edgefromalpha/>

This tool is an edge detect on the alpha channel that has separate adjustable erode and blur controls for both inside and outside of the matte.

VectorExtendEdge NKPD

Author: Michael Garrett



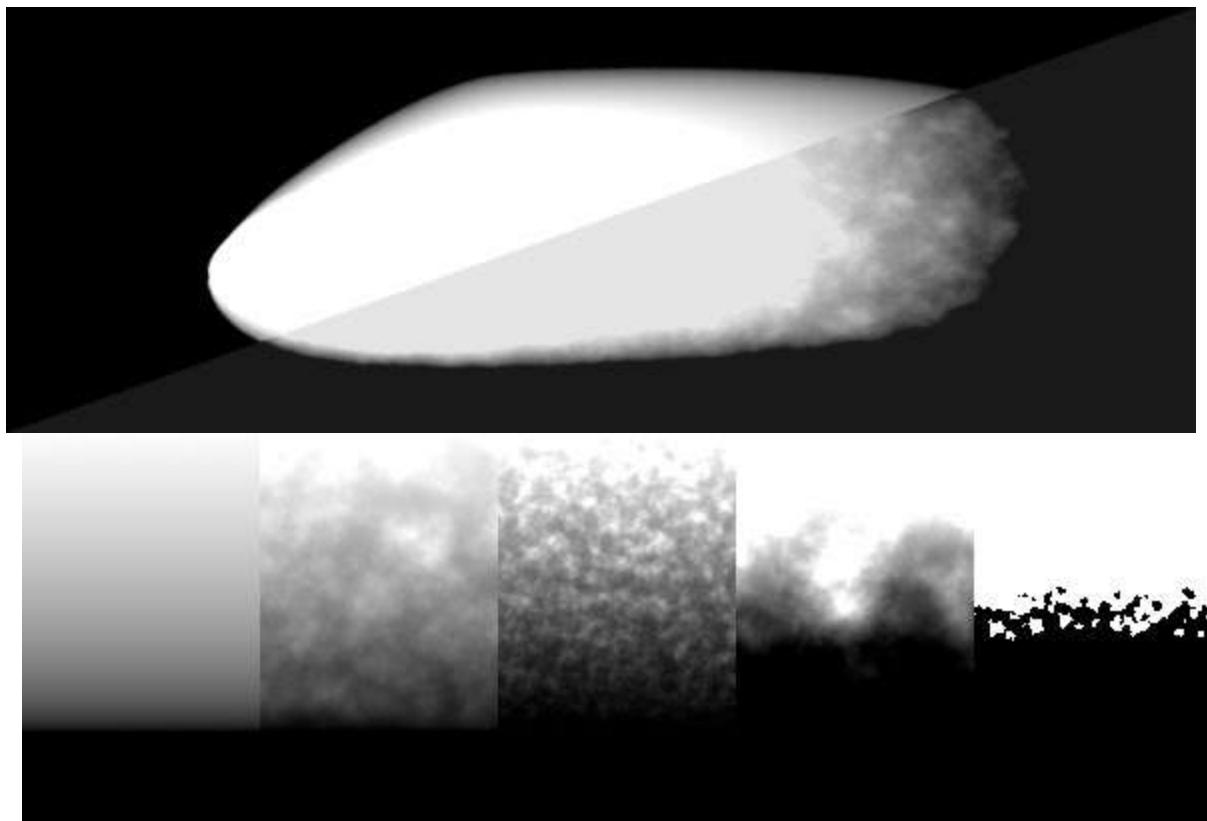
Nukepedia link: <http://www.nukepedia.com/gizmos/filter/vectorextendededge/>

Pushes rgb pixels outwards using vectors generated perpendicular to a control matte edge. It's similar to Frank Reuter's EdgeExtend gizmo but instead of recursively blurring and unpremuling, it recursively generates vectors based on the input control matte and uses VectorBlur to push the rgb pixels outwards.

To get the best results, you need to input a matte that conforms to the pixels you want to extend.

FractalBlur NKPD

Author: Richard Frazer



Tool link: <https://richardfrazer.com/tools-tutorials/fractal-blur-for-nuke/>

Github Link: <https://github.com/RichFrazer/fractal-blur/blob/master/fractal-blur.nk>

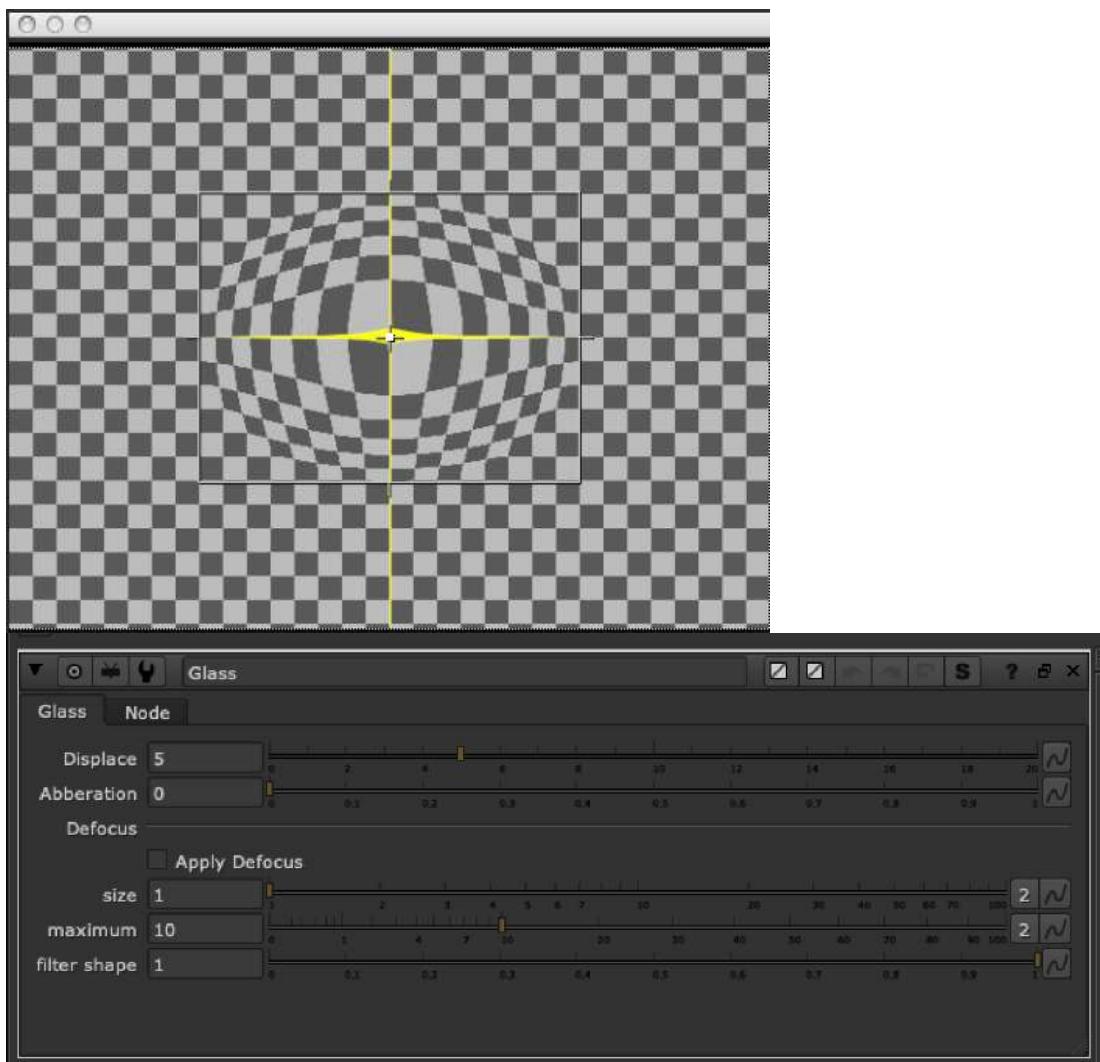
It's essentially just a blur combined with a noise filter so that the softened image does not have smooth gradients. It really helps to hide soft-edge mattes where the combined images have a lot of texture. I was working on Where the Wild Things Are at the time and every plate almost entirely consisted of heavy natural texture (forests, trees, fur etc.) and it became essential to use the fractalBlur on every single mask. Since then I frequently require this plugin and have not found an equivalent in Nuke, so thought I'd put one together.

Distortions Menu

This menu focuses on tools that distort, warp, haze, etc

Glass FR

Author: Frank Rueter - www.ohufx.com

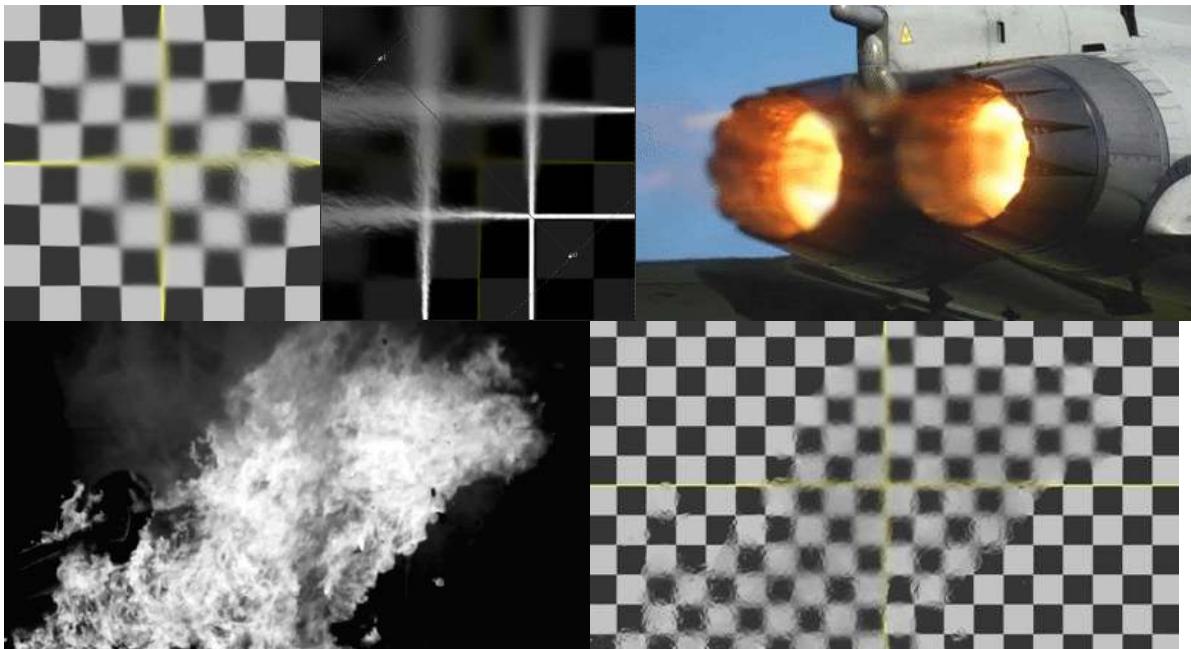


Nukedpedia link: <http://www.nukedpedia.com/gizmos/transform/glass>

This gizmo uses IDistort to create a simple glass light effect based on a control mask.

HeatWave NKPD

Author: Damian Binder



Nukepedia link: <http://www.nukepedia.com/gizmos/filter/realheatdistortion>

HeatWave is a gizmo created by Damian Binder that simulates realistic heat distortion you often see around fire or other sources of heat.

Main Features:

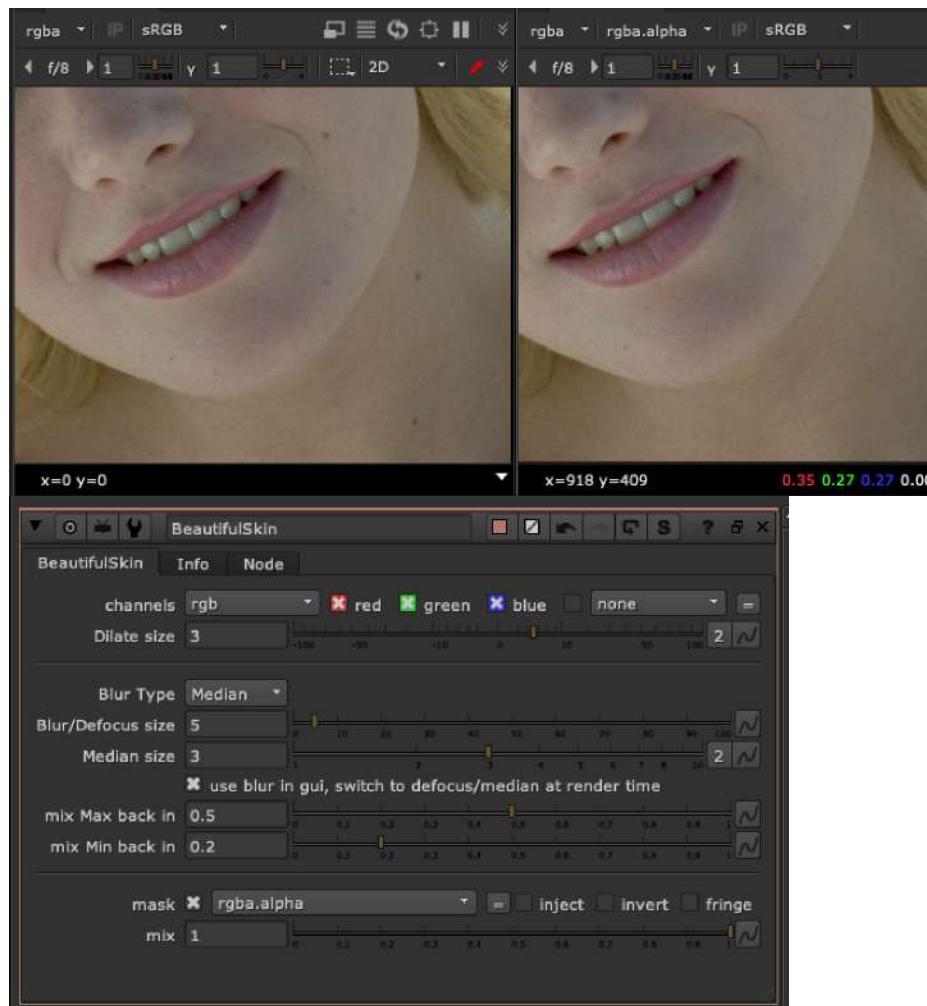
- Realistic heat distortion and haze simulation.
- Smart on-screen wind direction controller.
- Smooth transitions between distorted and non-distorted areas.
- Tracking data for moving shots can be added.
- Custom distortion maps can be used (Real fire elements or cg simulations).

Other features:

- Faster rendering: When using the Mask or Custom input, only the inside of the bounding box will be calculated.
- Distortion maps can be looped. This helps when using short fire or cg elements. It can also save rendering times (Limit the number of frames the noise node has to calculate).
- Works with anamorphic resolutions.
- Some effects like Smoke and Chromatic Aberration can be added.

BeautifulSkin TL

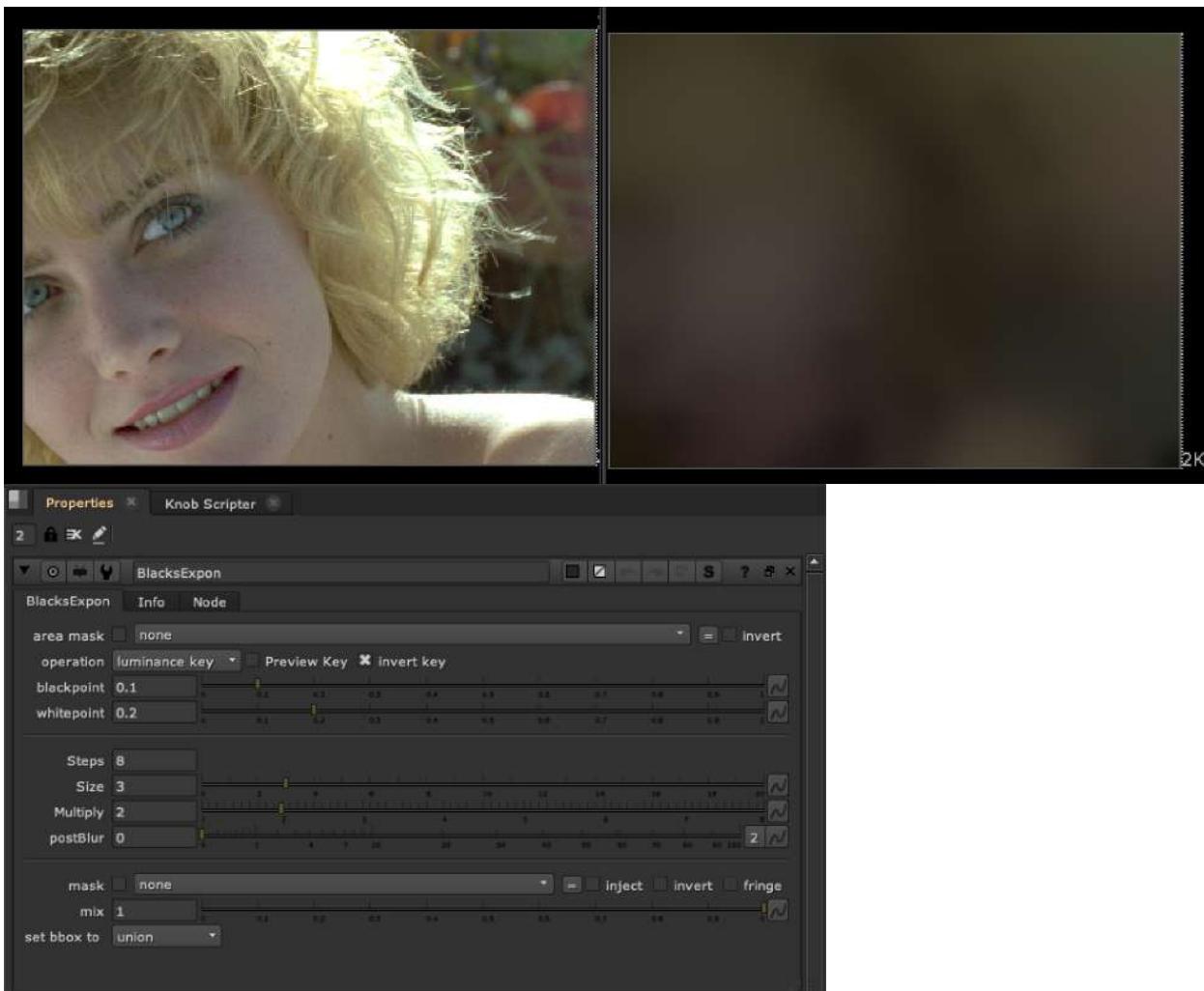
Author: Tony Lyons - www.CompositingMentor.com



Simple tool that uses erode dilate, blur, defocus, median with a mask to paint out moles, artifacts, markings, etc while maintaining a soft smooth appearance. Ability to bring back min/max values with separate sliders. Best when used with mask.

BlacksExpon TL

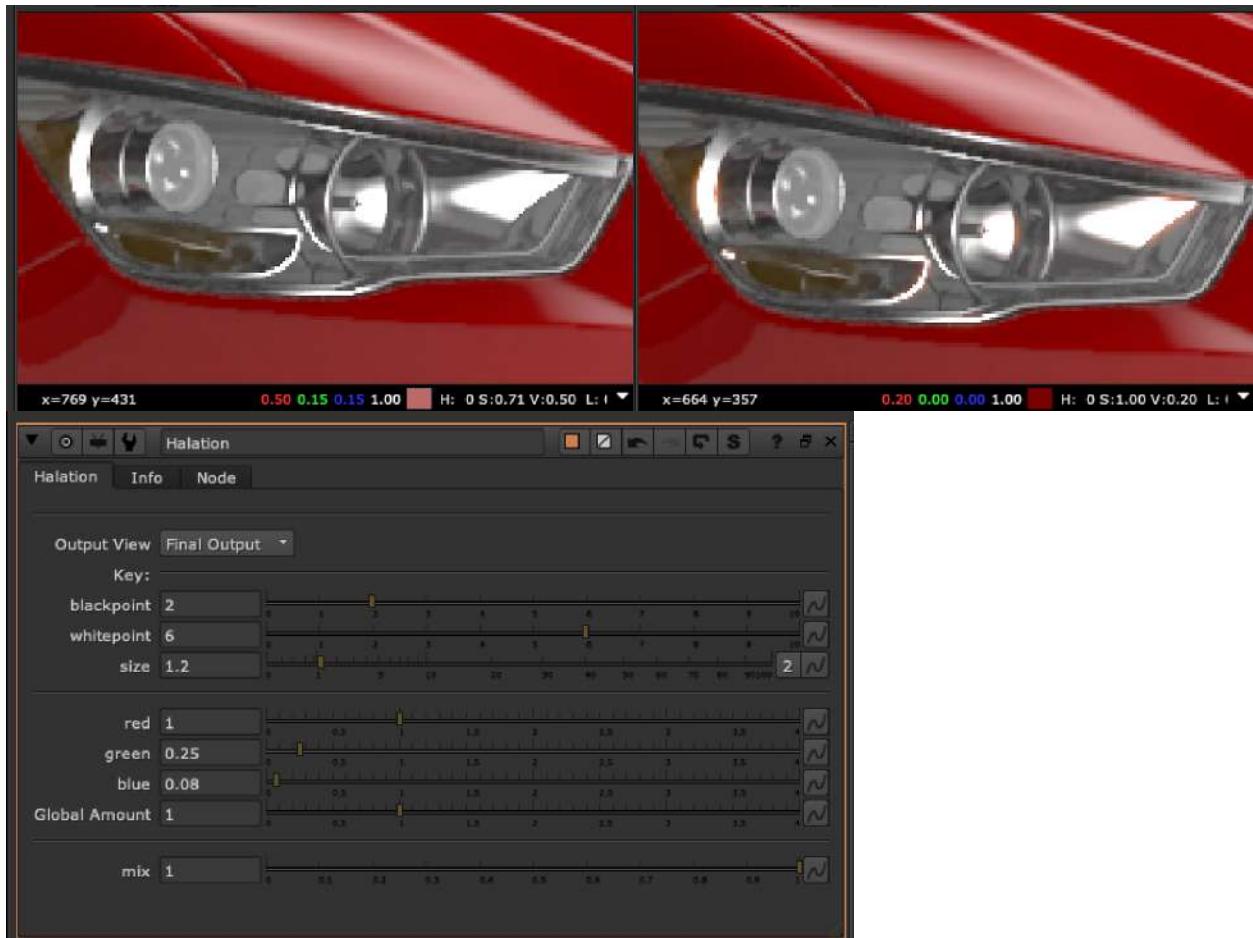
Author: Tony Lyons - www.CompositingMentor.com



This tool Exponentially blurs the lows of the plate with a merge(Min) operation. Basically trying to find the low colors and spread them out using blur/unpremult technique. This can be quite handle if used with the blacksMatch tool color input. If you have a plate with dynamic lighting, it can be a handy way to get an animated black color for free to match your CG renders too. Can also help with prep tasks if you need to paint out highlights and replacing with a “base” color.

Halation TL

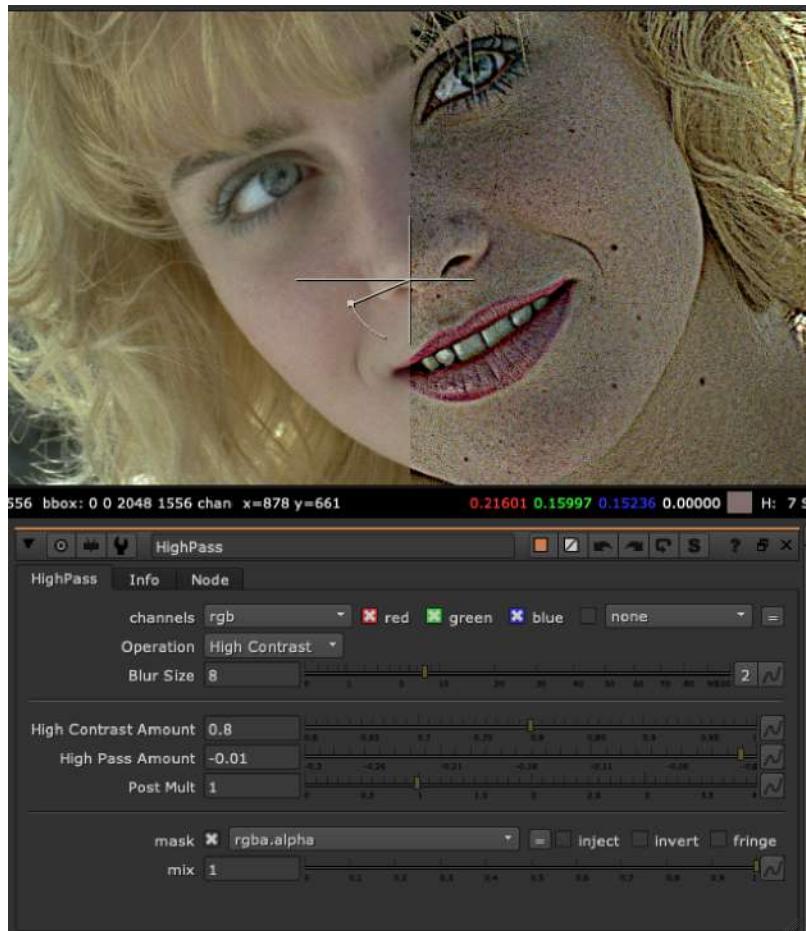
Author: Tony Lyons - www.CompositingMentor.com



Simple tool to introduce a little halation effect. Adjusts amount of r,g,b channels individually with an overall blur. Blackpoint and whitepoint sliders are the low and high of a luminance key threshold for where the effect starts. You can output final, effect only, and luma key.

Highpass TL

Author: Tony Lyons - www.CompositingMentor.com



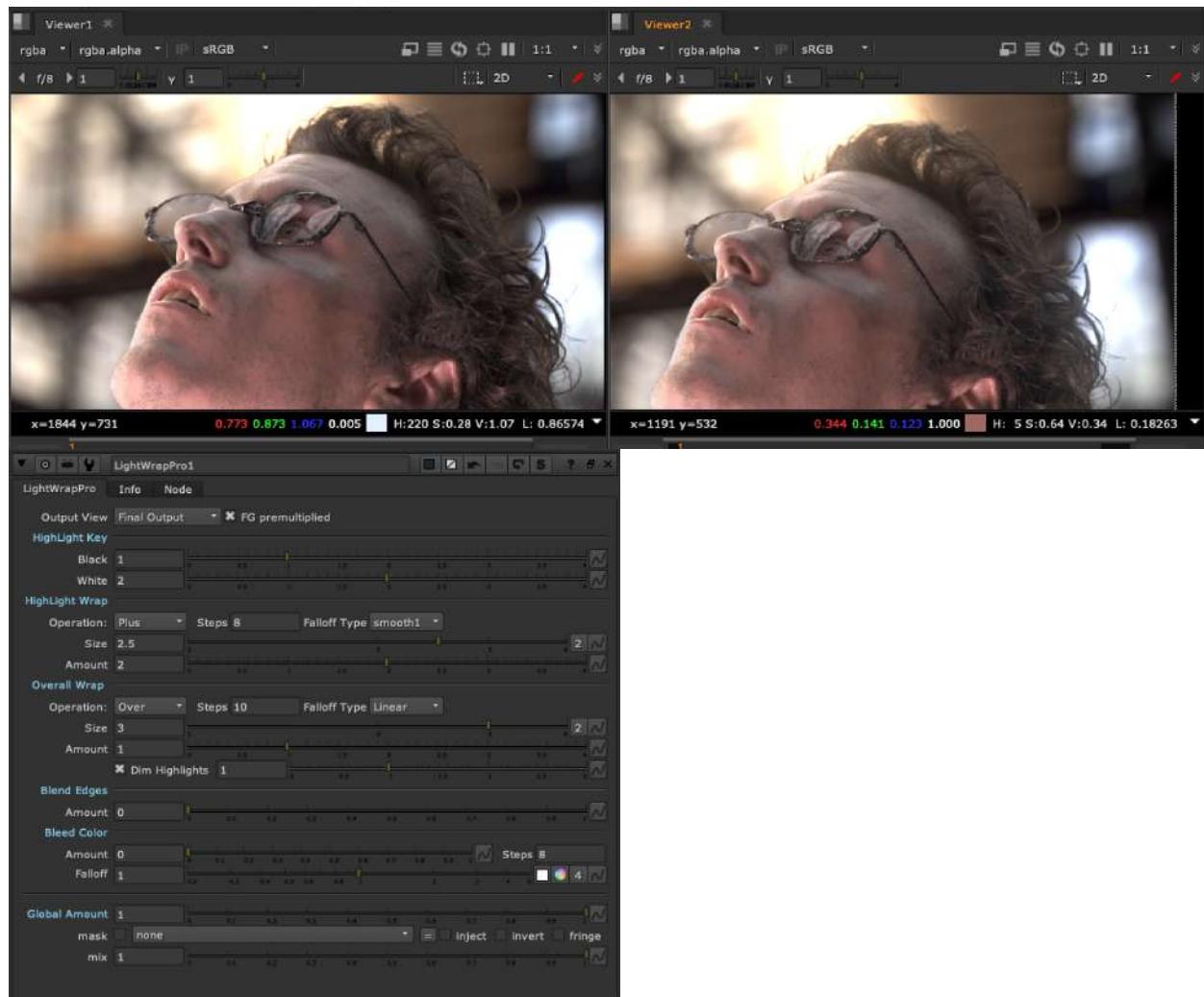
Gives you the difference between a blurred input and the original input. making small details quite noticeable.

The 2 main uses are:

- 1.) to aid 2d tracking
- 2.) to apply a different type of sharpen filter to an image.

LightWrapPro TL

Author: Tony Lyons - www.CompositingMentor.com



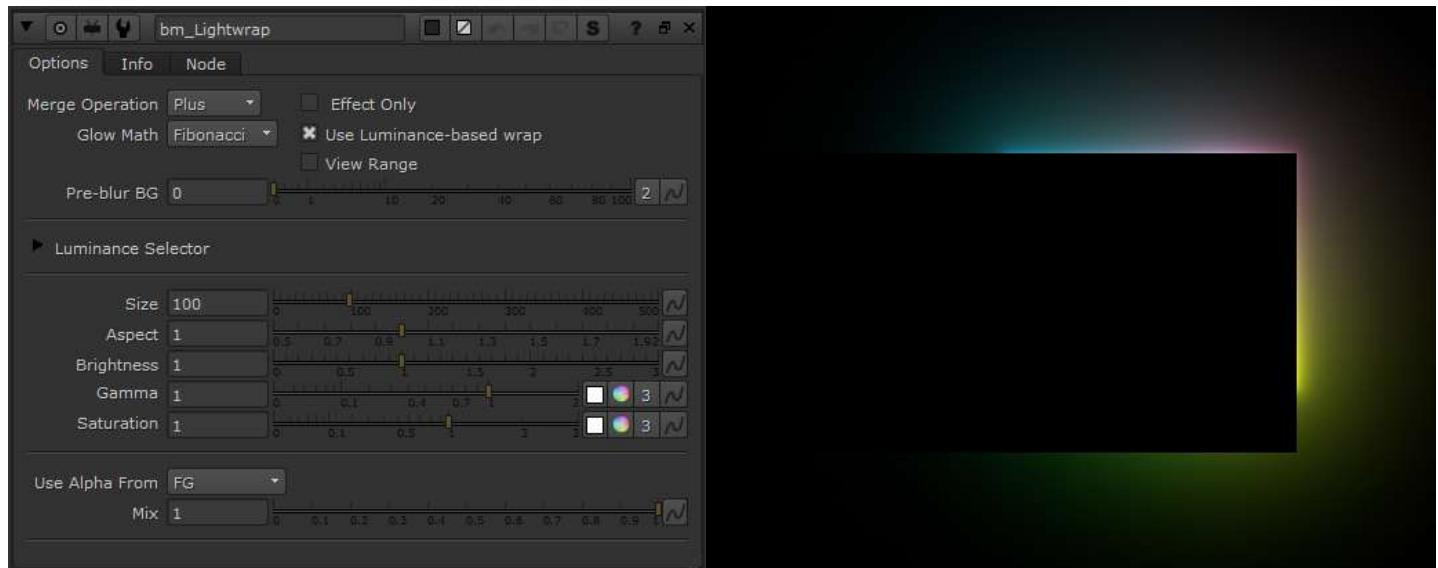
This lightwrap tool evolved from luma pictures Fuse node, but as a stand alone lightwrap. The features that set this tool apart: Exponential blurring with adjustable steps, a Highlight Wrap and Overall Wrap, blend edges and bleed color options (idea from luma's fuse). Multiple output views to help you work step by step with the workflow of the tool.

Steps:

- 1.) View/Adjust Highlight key, this is a key of the brightest areas of the plate to wrap
- 2.) View/Adjust Highlight Wrap, make adjustments for a tight wrap for highlight areas
- 3.) View/Adjust Overall Wrap, make adjustments for larger, dimmer overall lightwrap
- 4.) Good rule of thumb is when you think your lightwrap looks good, reduce the global amount down to $\frac{1}{3}$

bm_Lightwrap BM

Author: Ben McEwan - <https://benmcewan.com/nukeTools.html>



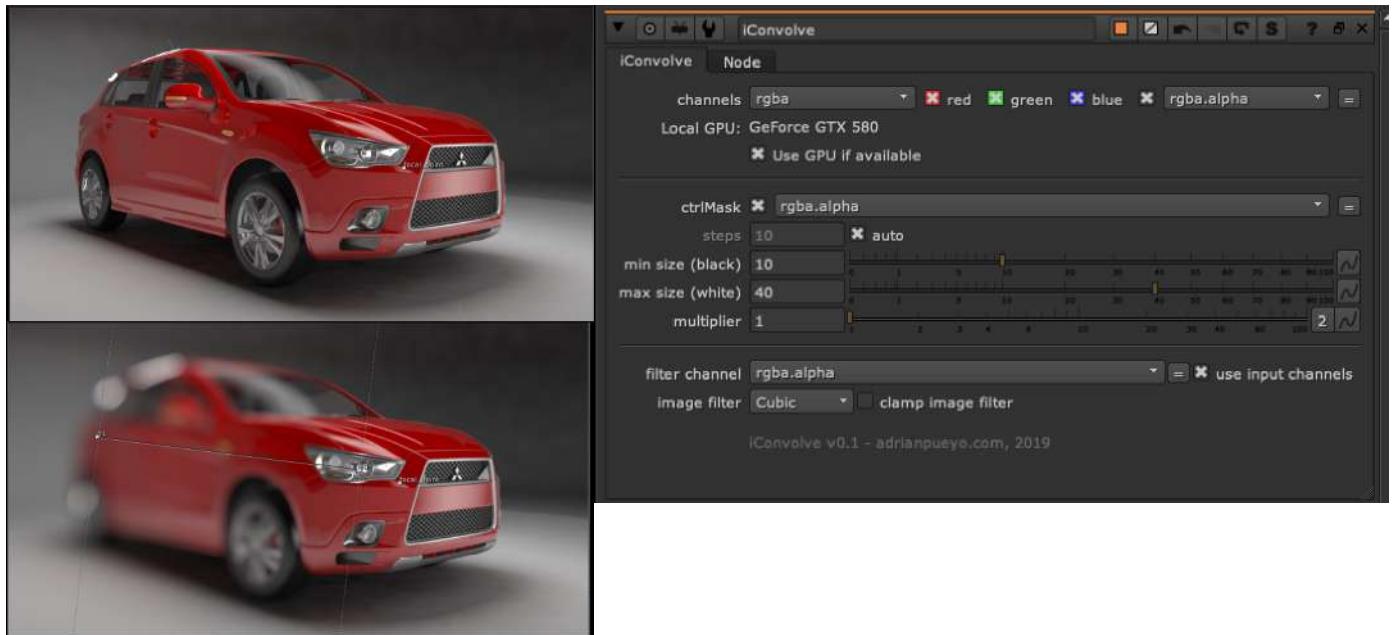
Nukepedia link: http://www.nukepedia.com/gizmos/filter/bm_lightwrap

Github link: https://github.com/BenMcEwan/nuke_public

Like bm_OpticalGlow, this adds exponentially-increasing blurs together to produce a more optically-correct, natural lightwrap.

iConvolve AP

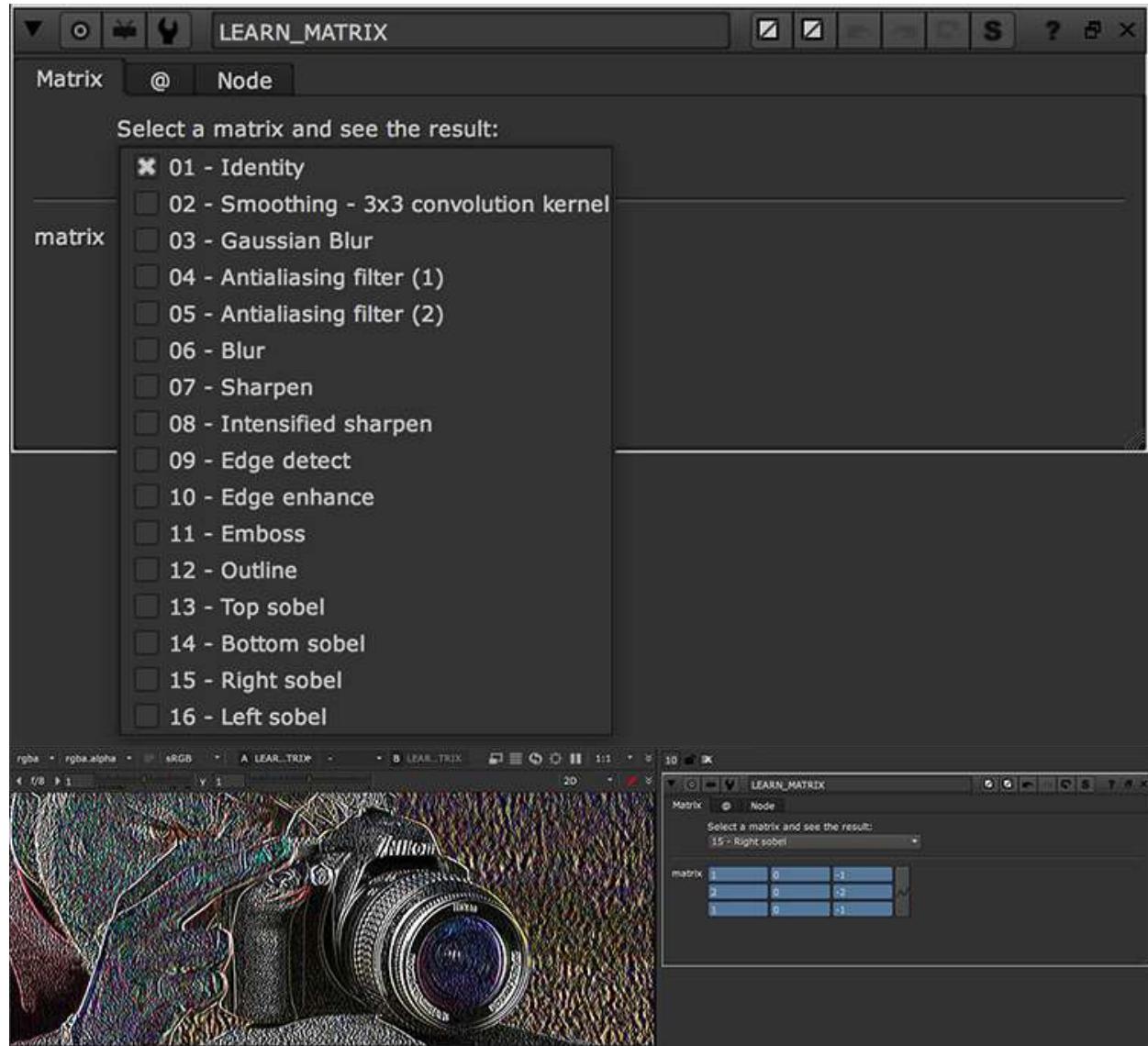
Author: Adrian Pueyo - <http://www.adrianpueyo.com/>



Similar to the iBlur, but with a convolve (defocus)! Uses a control mask and a custom filter/kernel to create a convolve effect with a falloff. Ramps off from 0-1, 0 will have the min size convolve, 1 will have the max size convolve.

ConvolutionMatrix AG

Author: Andrea Geremia - www.andreageremia.it/tutorial.html



Nukepedia Link: http://www.nukepedia.com/gizmos/filter/learn_matrix

Website: http://www.andreageremia.it/tutorial_matrix.html

Apply a preset Matrix Filter 3x3 to your image

apChroma Tools AP

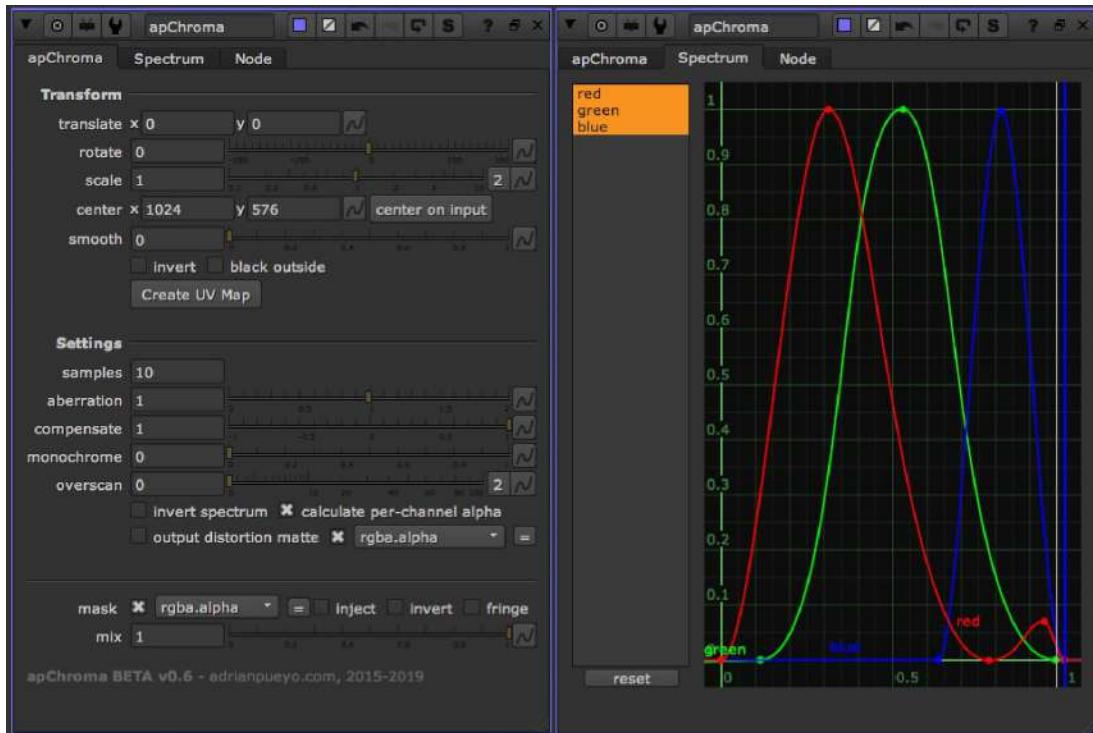
Author: Adrian Pueyo - <http://www.adrianpueyo.com/>



The following tools are apChroma tools meant to work with each other in a chromatic aberration workflow. From applying different types of chromatic aberration, blur, transforms, per channel (and could apply to all channels in the stream) to merging per channel so the aberration is applied correctly with the background.

apChroma AP

Author: Adrian Pueyo - <http://www.adrianpueyo.com/>



Nukepedia link: <http://www.nukepedia.com/gizmos/filter/apchroma>

apChroma is an advanced chromatic aberration and drift gizmo, that works through a sub-frame blend of different values on an STMap and Transform, while creating a user-defined color spectrum.

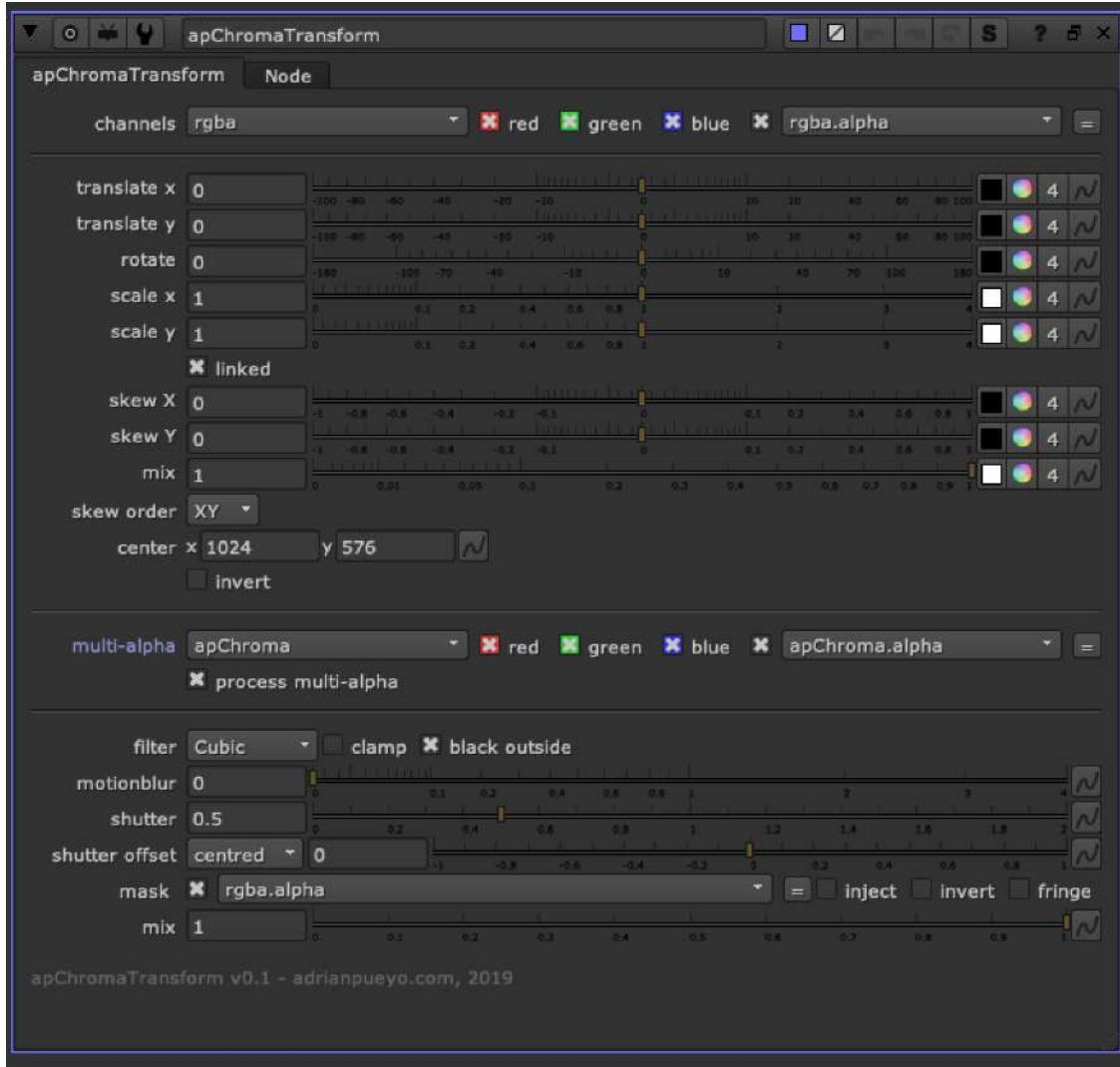
apChroma can calculate a multi-channel alpha for correct merging of the result onto a plate, and the included apChromaMerge node will perform the multi-alpha merge operation.

Demo video on vimeo: <https://vimeo.com/344248811>

apChroma demo on VFXforFilmmakers channel: <https://youtu.be/K28VNUVseTY?t=1764>

apChromaTransform AP

Author: Adrian Pueyo



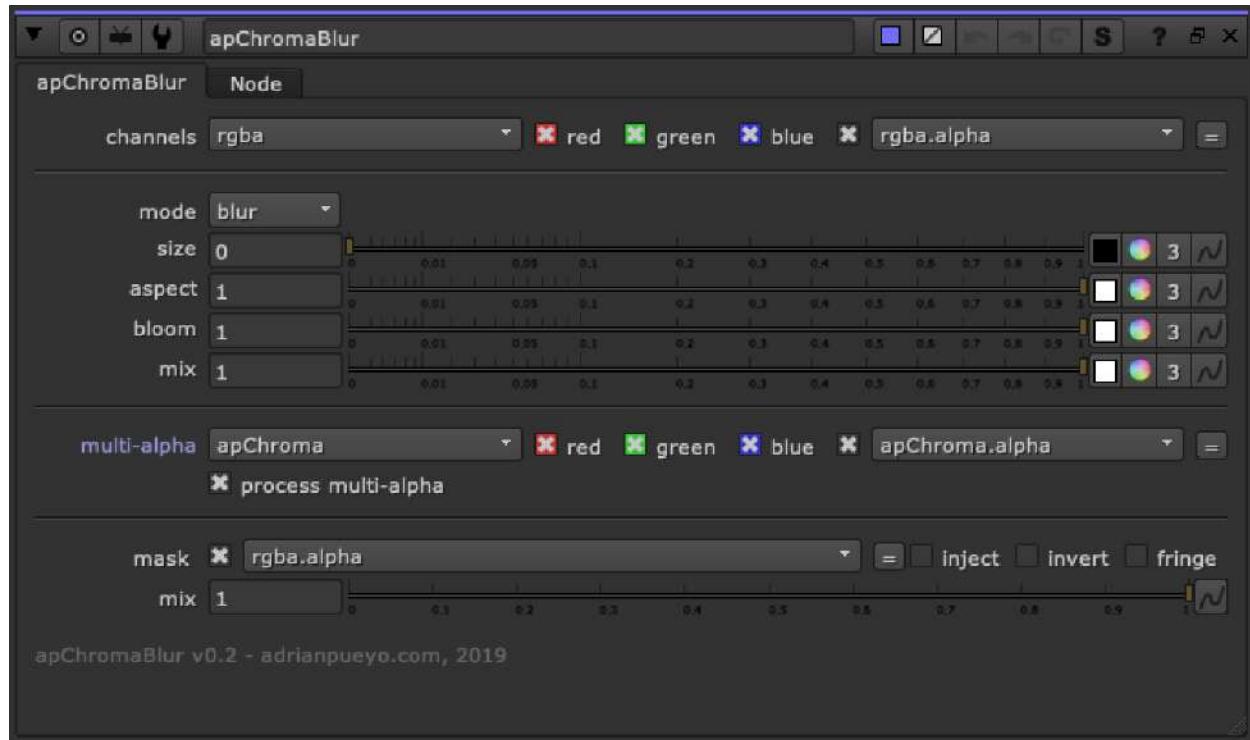
This tool allows you to transform red, green, and blue channels separately, and for all Layers (channels) in the stream.

Each transform knob can be broken up to r,g,b,a channels and individually manipulated.

To be used with the apChroma multi-channel alpha workflow so that apChromaMerge can be used.

apChromaBlur AP

Author: Adrian Pueyo - <http://www.adrianpueyo.com/>



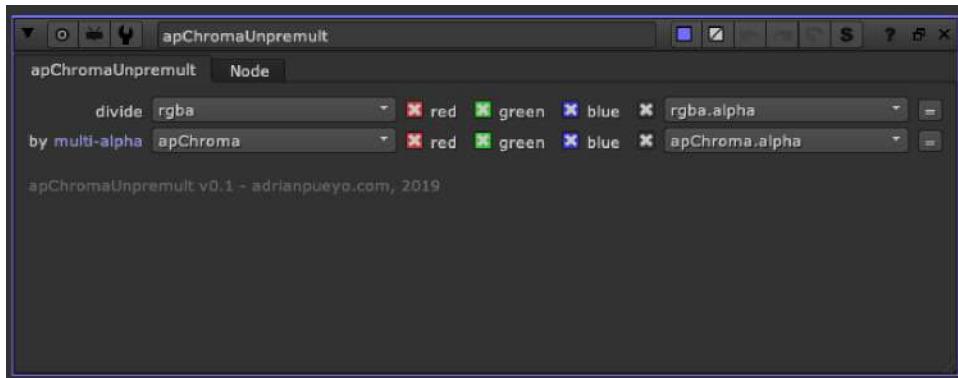
This tool allows you to blur or defocus red, green, and blue channels separately, and for all Layers (channels) in the stream.

Each knob can be broken up to r, g, b channels and individually manipulated.

To be used with the apChroma multi-channel alpha workflow so that apChromaMerge can be used.

apChromaUnpremult AP

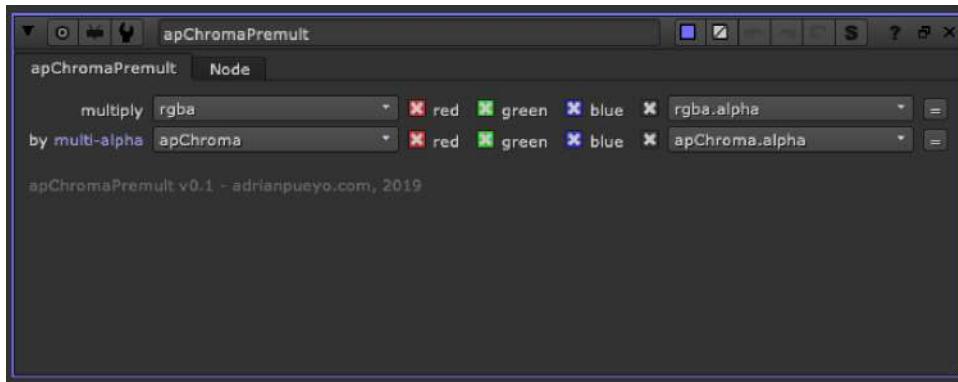
Author: Adrian Pueyo - <http://www.adrianpueyo.com/>



There are some rare cases where you will need to unpremult and premult all layers(channels) by the apChroma multi-alpha layer. Whether it's for color corrections, lightwrap, or something else, this node gives you the option to successfully unpremult and premult during the apChroma workflow.

apChromaPremult AP

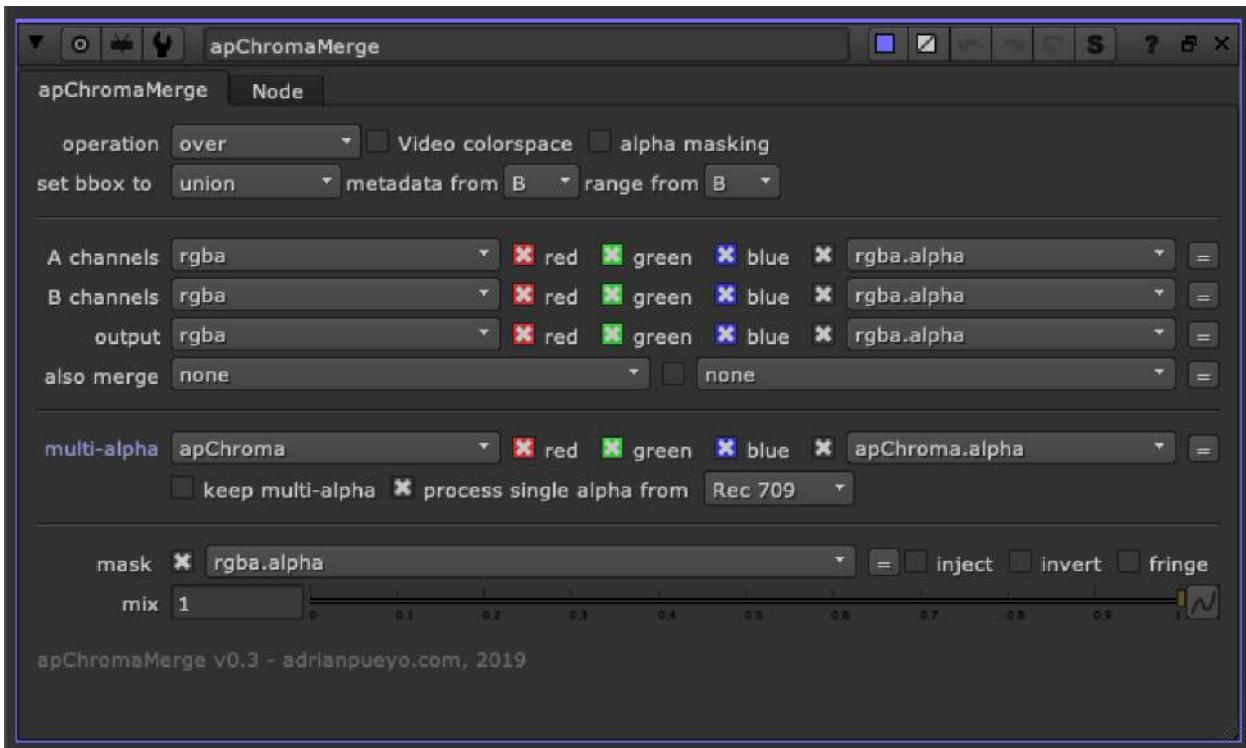
Author: Adrian Pueyo - <http://www.adrianpueyo.com/>



There are some rare cases where you will need to unpremult and premult all layers(channels) by the apChroma multi-alpha layer. Whether it's for color corrections, lightwrap, or something else, this node gives you the option to successfully unpremult and premult during the apChroma workflow.

apChromaMerge AP

Author: Adrian Pueyo



apChromaMerge is the final step in the apChroma Workflow. When using the apChroma toolset, a new layer, apChroma will be in the channels stream. This layer(channel) is storing each 'alpha' to be used per channel.

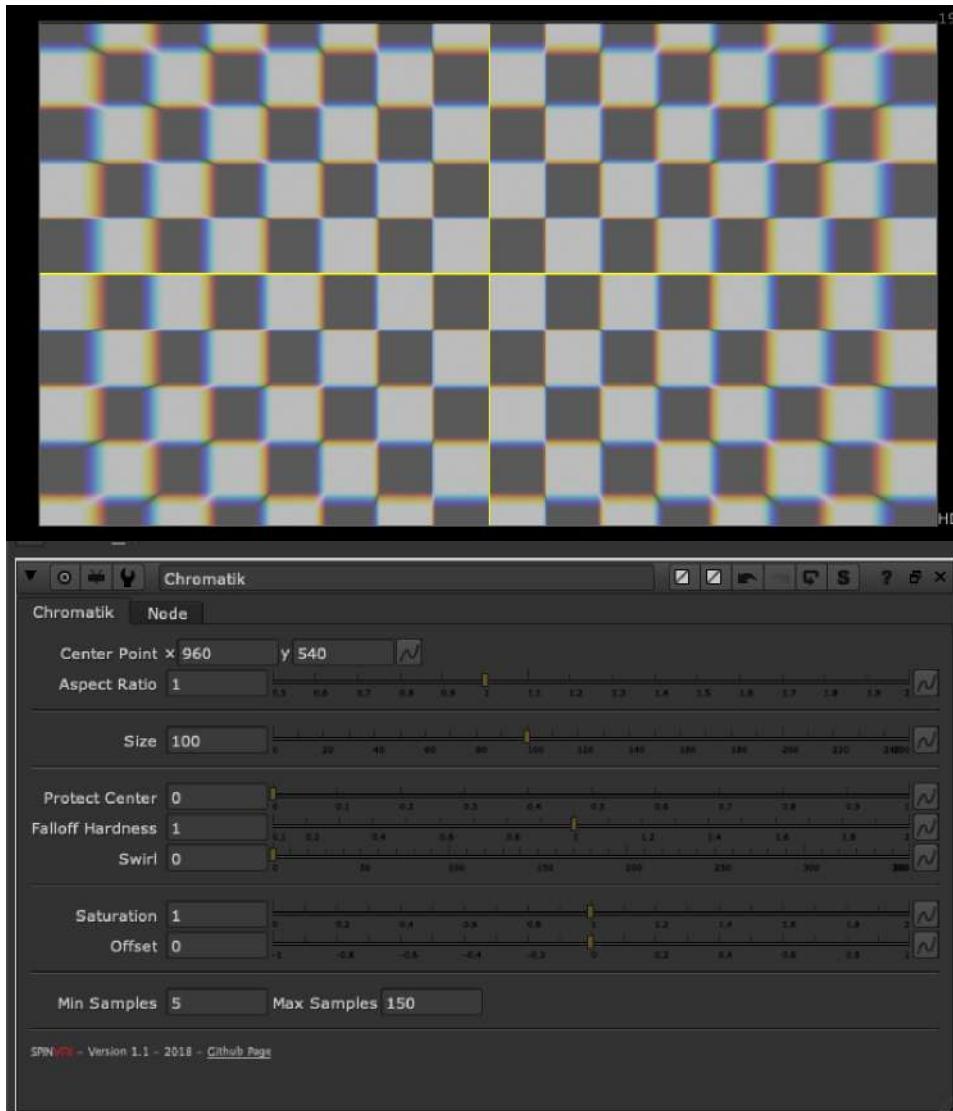
Since there is separation between the channels, whether they are transformed or blurred differently, they also need individual alphas to properly merge them over the background image.

You will find the normal options of a merge, with the option to:

- 1.) Keep the mult-alpha apChroma layer and pass it onto the this merge stream
- 2.) Since there are 3 alphas being used to merge each channel (red, green, blue) it's difficult to know which alpha to use for the final alpha to be passed onto this stream, representing the A inputs final alpha. By default, process single alpha from Rec 709 is checked on, meaning all 3 channels will be desaturated with the Rec 709 algorithm to produce a new greyscale alpha off the (luminance) and this will represent the new alpha channel to be passed onto the B stream after the merge.

Chromatik SPIN

Author: SPIN FX



Nukepedia Link: http://www.nukepedia.com/gizmos/other/spin_nuke_gizmos-1

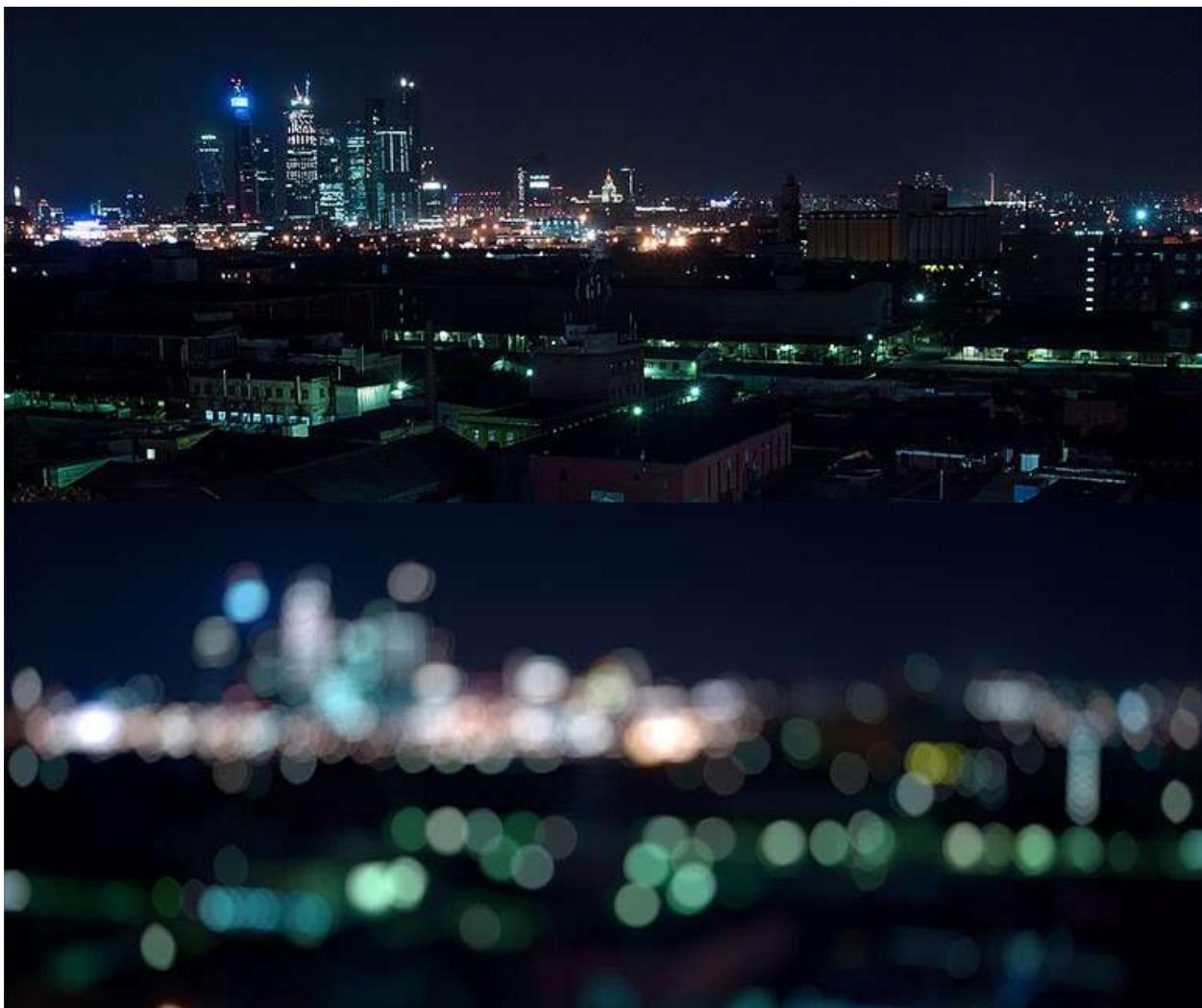
Github Link: https://github.com/SpinVFX/spin_nuke_gizmos

Demo video: <https://vimeo.com/381270956/f7399b6e1d> @ 16:45

Chromatic aberration node using a spectral wavelength gradient.

CatsEyeDefocus NKPD

Author: Alexander Kulikov

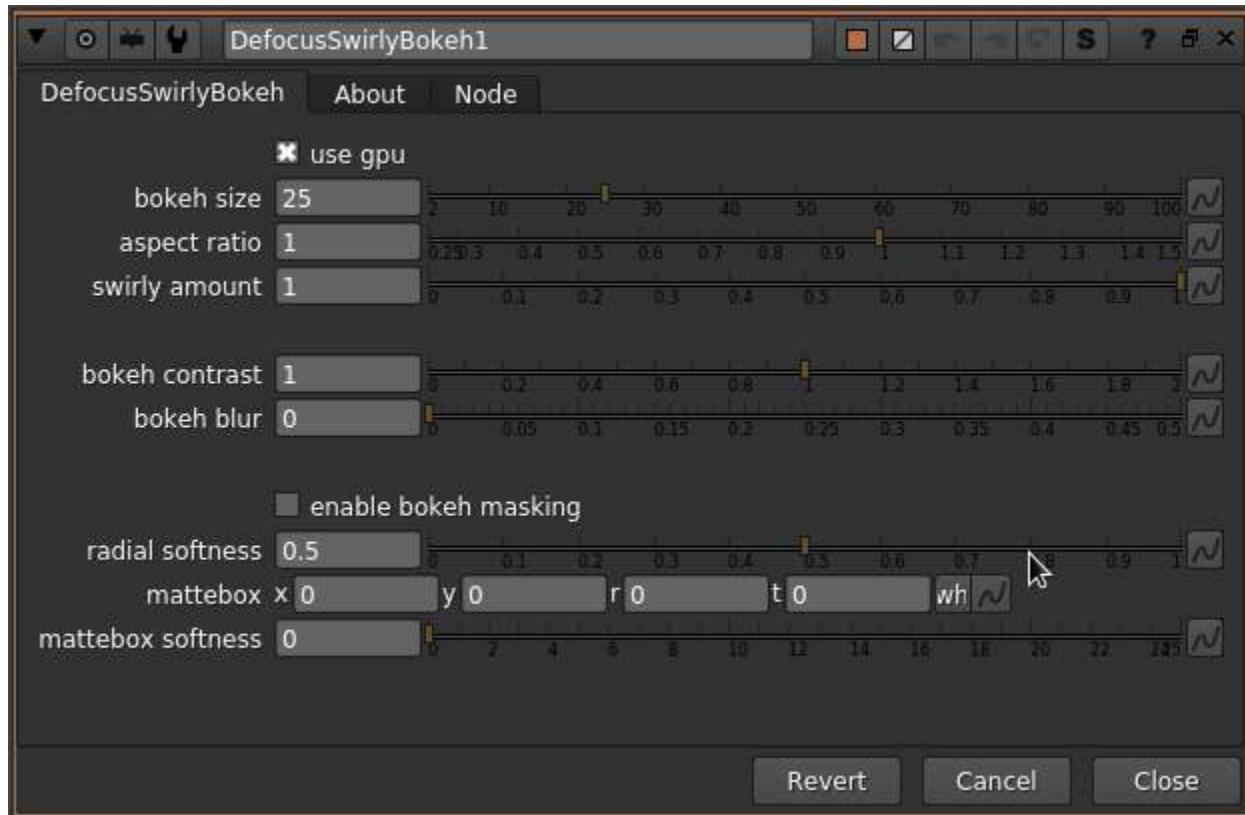


CatsEyeDefocus is a convolution filter which simulates swirly bokeh.

This lens abnormality is also called cat's eye effect and it is noticeable when an aperture goes wide. The shape of the bokeh progressively narrows from the image center towards the edges and starts to resemble a cat's eye.

DefocusSwirlyBokeh NKPD

Author: Jed Smith



Github link: <https://gist.github.com/jedypod/5d35858d488df478aaf2f2e8f3f7875a>

Creates Swirly Bokeh or Cat's Eye Bokeh shapes on the edges of frame. Does not perform depth-varying defocus. Needs a good GPU to run fast.

Based on Alexander Koolikov's CatsEyeDefocus.

deHaze NKPD

Author: Lucas Pfaff



deHaze is build after the great dehazing-tutorial made by Mads hagbarth Damsbo. Mads concludes the tutorial with 'So you can just package this up and make a cool tool and put it Nukepedia', so I gave it a try :)

I added some functionality to shuffle the affected areas into the alpha channel, as well as colour correct the footage based on this matte. Results vary highly with the given shot, it always needs a bit of fiddling around. I highly recommend watching the tutorial to understand the underlying principle.

Needs Nuke 12 for the C_Bilateral node.

DeflickerVelocity NKPD

Author: Julien Vanhoenacker



Deflickering

Part of the CG artist's job is to balance rendering time and quality. Low rendering quality often resulting in aliasing, or flickering, specially on raytrace renderers. To find the right balance means to test render with different settings in order to find the one that render the fastest while still being acceptable in terms of aliasing and flickering. But this is time consuming and

the render time necessary to achieve this result might also be quite high. It is therefore interesting to find tricks to improve quality with equal or even lower render times. It is possible to use built-in denoisers, however this becomes useless if the flickering happens on the edges of the objects, or if it is on wide GI artefacts (as in lightcache or irradiance GI), and tends to make the image blurry. The technique we present here is deflickering based on previous and next frames.

Basic Deflickering

Consider this, a static or very slow camera movement, and slow or no animations in the scene. This is a scenario where flickering will be most obvious as very little is changing in the scene. This is also a scenario where deflickering is the easiest to achieve. Offset time by -1 frame to get the previous frame, merge it at 50% on top of your current frame, Offset time by +1 frame to get the next frame, merge it at 33% on top of your previous result and voila! Your compositing is now averaging previous, current and next frame and as a result, flickering, aliasing and noise are decreased by 3. And because the camera and animation are not or slowly moving, the averaging is close to invisible, apart in the overall gained stability in the picture.

The Deflickering is specially useful when dealing with sharp self-illuminating object that create complex GI solutions...

Masking

Now a situation where nothing is moving in the scene is uncommon. In the case where something is bouncing around, it should definitely not be averaged, otherwise this will result in an ugly 3 frame long stepped motion blur that nobody wants. The good news is that if the object is actually moving quite fast while the rest of the scene is quiet, the flickering will most likely not be noticeable on this object, since its position, angle, lighting will be changing every frame. It is therefore ok to not deflicker it, while still deflickering the surroundings. In order to do that we can just mask it out of the deflicker.

Now nobody likes to animate roto shapes, so a quick tip to automatically mask the fast moving subject is to use the velocity pass that you have cleverly rendered beforehand. The velocity pass is actually a colored representation of the speed of each pixels, so all we have to do is to key in static (grey) pixels, key out fast (colorful) pixels and use the resulting alpha as a mask for the deflickering, the tolerance of the keying will dictate how much deflickering you want.

Deflickering with Velocity pass

Now in some cases, things are not that easy. For example a not so slow but smooth camera pan or track, or an object slowly shifting in the opposite direction will result in quite visible flickering, and would give ugly results with basic deflickering. Here we want to use the

aforementioned velocity, but not for masking anymore. The velocity channel stores in the red and green channel the velocity of a pixel as a 2D vector. Following the same principle as before, we want to offset time to get the previous frame, and use the velocity pass with a displacement node to distort this frame 'forward' and align it with the current frame. Realign the next frame 'backward' and blend the whole thing together and you get a result that is deflickered, with movement. Now this technique is quite powerful and can save your life in a tight deadline. Of course there are still limitations for super fast moving objects and very separate foreground and backgrounds, but mixing this with the previous mask technique, you can deal with pretty much anything...

Extra Tip 1:

For those who use Vray, it is important in the DMC settings to uncheck "Time independent", to enable a noise that changes every frame, otherwise averaging will still display the same amount of noise.

Extra Tip 2:

Also for VRay users, you will notice if you separate your renders into elements, that flickering is mostly in the GI pass, and sometime in reflection or speculars... The deflickering can be applied only on these passes, leaving the other passes intact...

Extra Tip 3:

You can use multiple Deflickering nodes one after another to average 5, 7, 9 or even 11 frames together, depending on your type of animation/camera move...

FillSampler NKPD

Author: Mads Hagbarth Damsbo



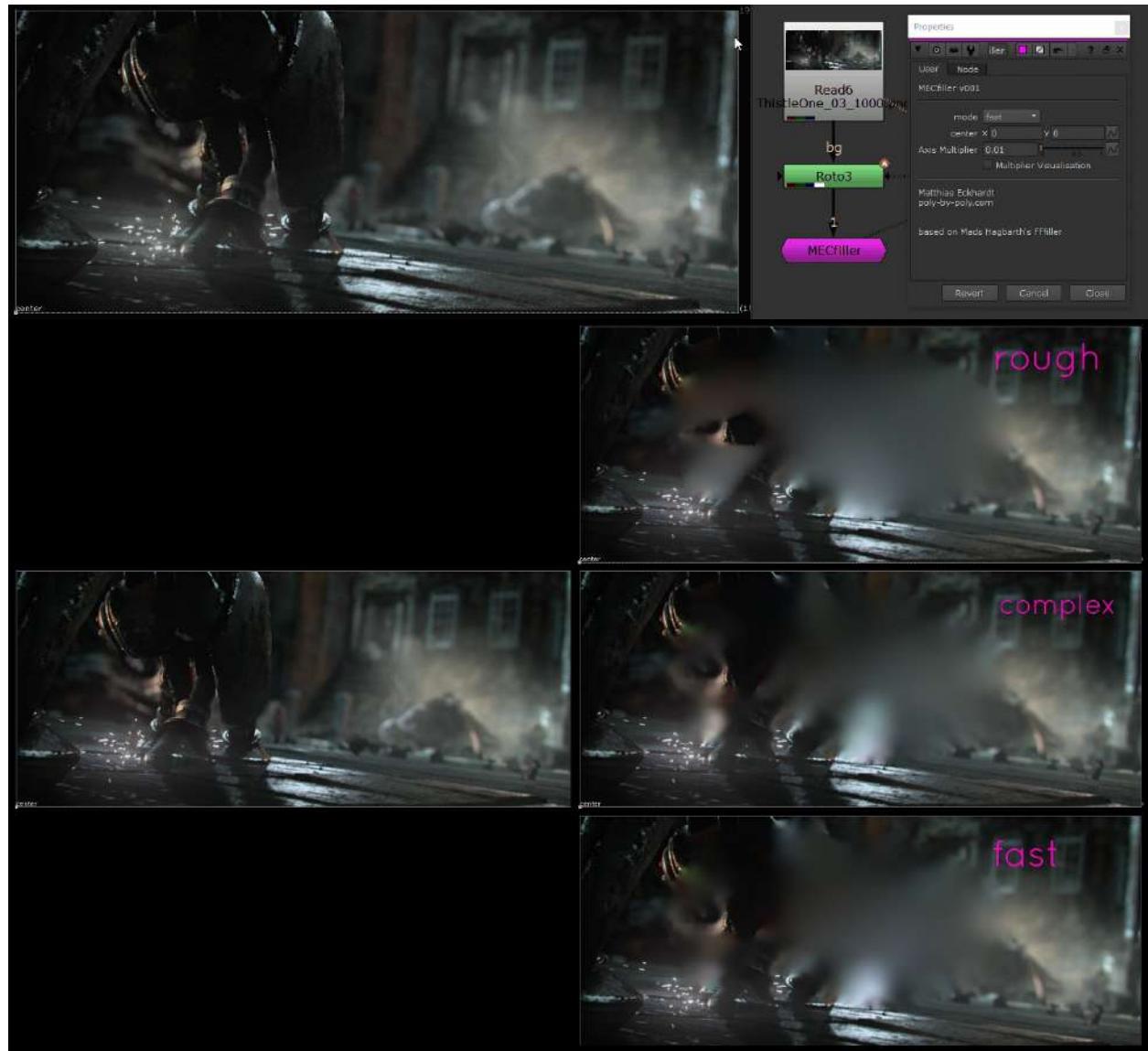
Website with more Details: <https://hagbarth.net/pixel-filling-methods/>

Download Link: http://www.hagbarth.net/nuke/FFfiller_v01.nk

Great tool from Mads Hagbarth that fills or extends edges or holes in the plate.

MECfiller NKPD

Author: Matthias Eckhardt



Nukepedia link: <http://www.nukepedia.com/gizmos/draw/mecfiller>

Modes:

- complex: 6 directional filtering with center/axis control settings
- fast: approach like 'complex', but with no center/axis control
- rough: multidirection estimation with center/axis control

center:

- if both values are on 0, the filtering is using the overall center of the image to direct the expansion towards.
- if the center gets placed, its redirecting the filtering distribution to this new middlepoint.

axis multiplier:

- specifies the amount of horizontal/vertical influence for the direction of the filling that should be used.

Axis Explanation:



In general the tool splits up the picture in quarters and applies a downsizing with a custom filter based on the direction from the specified center. Red for example filters more from the top left while yellow filters more information of the picture from the bottom up. If the axis multiplier is very low it only takes a very small amount of vertical/horizontal direction into account. If the axis multiplier is higher it filters more directly from the top/bottom or left/right which creates a more pointed fill estimation. This value generally can be adjusted to try to find a nicer overall filling.

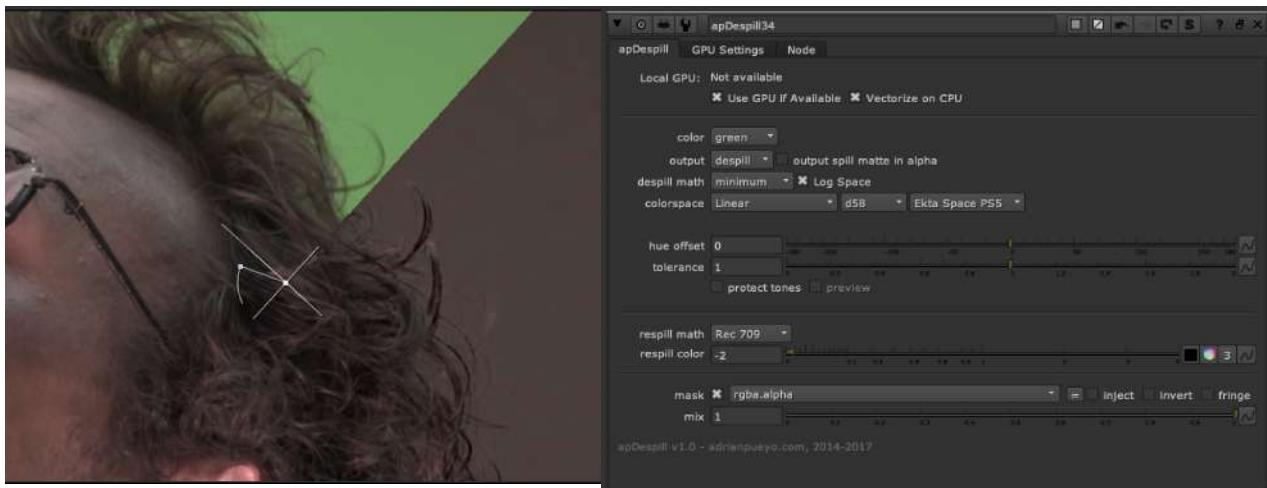
This hole approach is based on Mads Hagbarth's FFfiller tool. He wrote an article about it that can be found on his homepage - check it out if you want your mind to be blown:

<https://hagbarth.net/pixel-filling-methods/>

7.) Keyer

apDespill AP

Author: Adrian Pueyo - <http://www.adrianpueyo.com/>



Website download: <http://www.adrianpueyo.com>

Vimeo demo: <https://vimeo.com/208432912>

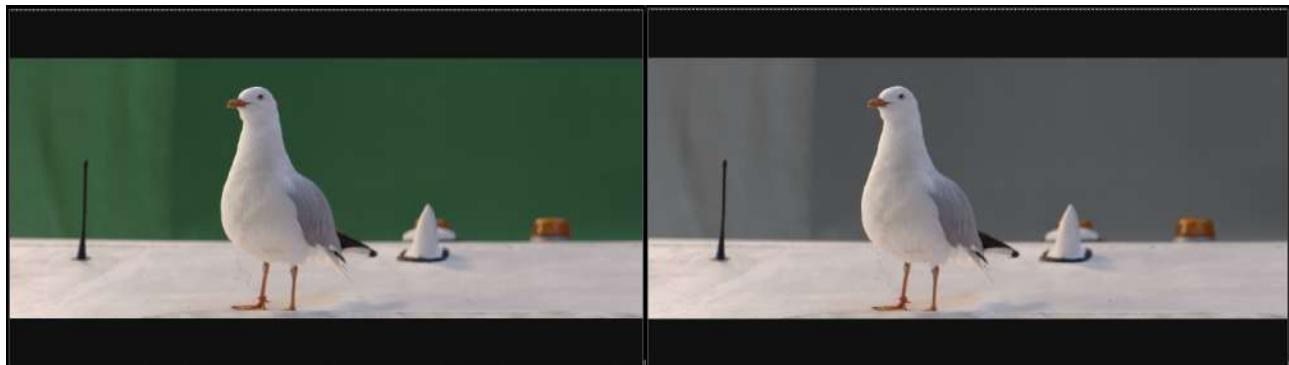
Nukepedia Link: <http://www.nukepedia.com/blink/colour/apdespill>

Nuke gizmo made to despill an image, with options to despill any hue (and not only the primaries), select different algorithms for de/re-spilling, and protect tones from the despill. All coded in Blinkscript.

- choose a custom color
- Despill math
- output a despillMatte
- swap colorspaces before/after
- despill in log Space
- adjust hue before/after
- adjust tolerance
- Protect tones for extra level of precision
- adjust respill color
- mask/mix

SpillCorrect SPIN

Author: SPIN FX



Github link:

https://github.com/SpinVFX/spin_nuke_gizmos/blob/master/gizmos/spin_tools/Keying/Spill_Correct2.gizmo

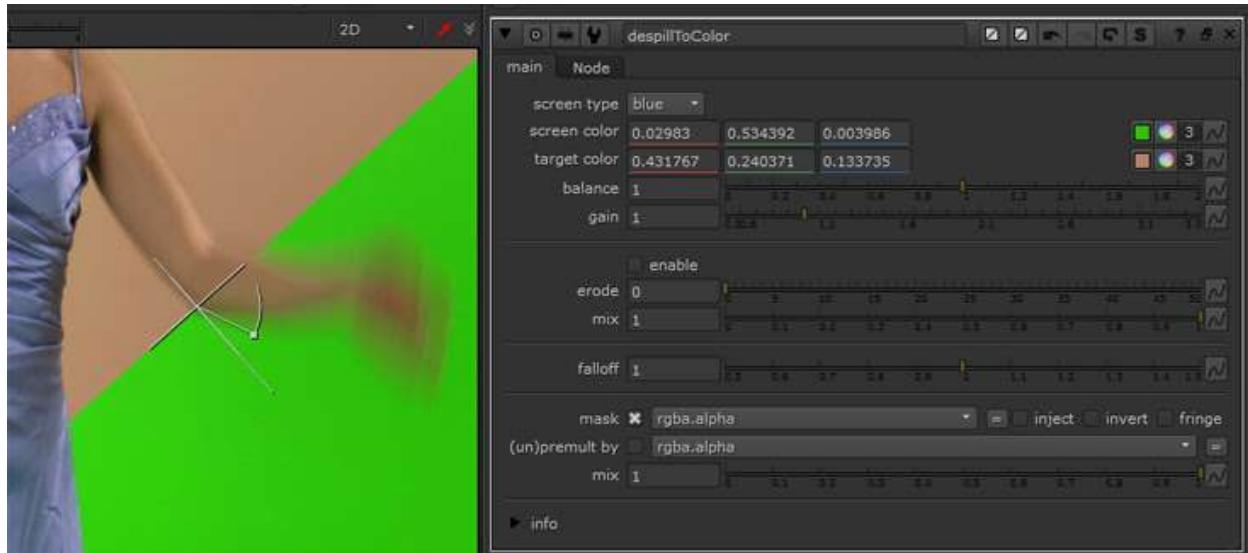
Vimeo demo: <https://vimeo.com/381270956/f7399b6e1d> @ 33:07

Use this tool to "despill" or mute colors introduced from Red/Green/Blue screens. Can replace the spill with a chosen color.

DespillToColor NKPD

Author: Johannes Masanz

Authors website: www.johannesmasanz.com



Nukpedia link: <http://www.nukpedia.com/toolsets/colour/despill-to-color>

Demo video: www.vimeo.com/despillToColor

If you are trying to fix an object's edge, by bringing back its original color, then this tool is perfect for you! DespillToColor is designed to work best with a specific user defined screen color and therefore a good local solution for certain areas of your plate.

- pick a blue- / greenscreen color - set or pick a target color.
- 'balance' controls how much red tones are being affected by the despill operation opposed to blue and green tones.
- Increasing the 'gain' will affect a wider range of colors, like the finetune knob in DespillMadness.
- 'erode' and 'falloff' sometimes work well for increasing the effect in defocussed areas. I usually go for ramping up the 'gain' OR the 'erode', rarely both.

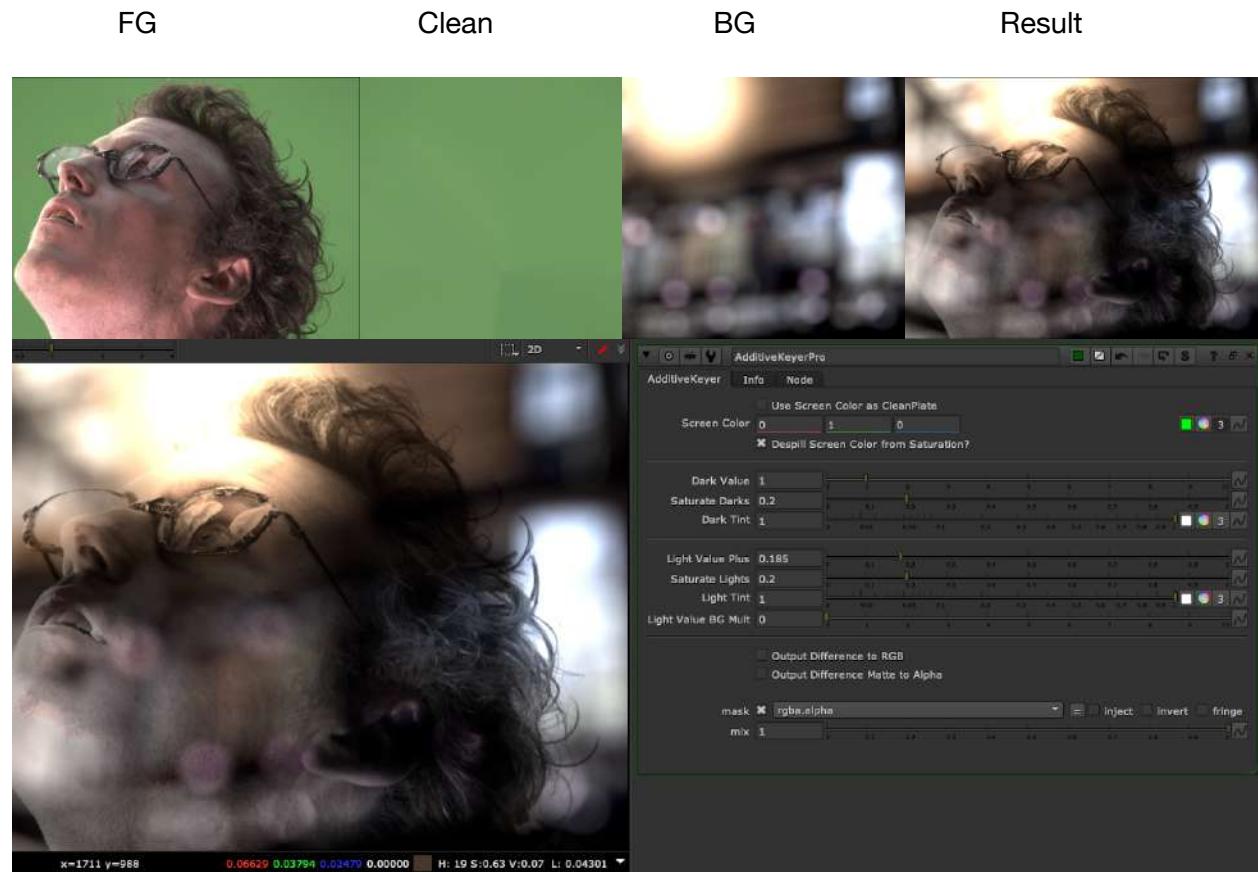
Example pics: (For testing I recommend blurring these jpegs a little to get rid of compression artefacts.)

<http://johannesmasanz.com/files/despillToColor/greenscreen.jpg>

<http://johannesmasanz.com/files/despillToColor/bluescreen.jpg>

AdditiveKeyerPro TL

Author: Tony Lyons - www.CompositingMentor.com



Additive Keyer pro does an additive key, which finds the highs and lows from the difference of a cleanplate and the original greenscreen plate and lightens / darkens the BG image. This process is handy for capturing subtle details in edges, such as hair and motionblur.

There are many additivekeyers, but this one has some unique features:

- 1.) The light values have a plus and BG mult slider, giving you more control over highlights
- 2.) The dark values uses a divide/multiply technique on the bg, preventing negative values
- 3.) Option to keep some saturation from the original colors, with a screen color picker to help remove green/blue spill from the edges if you choose to keep saturation.
- 4.) Options to output the difference to RGB, and difference matte to alpha, to use other comp techniques

apeScreenClean AP

Author: Adrian Pueyo



Website download: <http://www.adrianpueyo.com>

Clean up an irregular green or blue screen using your plate and a clean plate. apScreenClean is my own flavor of clean screen tool (ScreenCorrect, PxF_ScreenClean, ...) offering extended functionality.

apScreenGrow AP

Author: Adrian Pueyo - <http://www.adrianpueyo.com/>



Erode the color of a screen to fill the insides of your subject, useful to generate cleanplates in a controlled way.

LumaKeyer DR

Author: Derek Rein - <http://derekvfx.ca/nuke/>
adjusted by Tony Lyons www.CompositingMentor.com



Original Tool by Derek Rein can be found on his github page:
<https://github.com/DerekRein/.nuke/blob/master/ToolSets/lumaKeyer.nk>

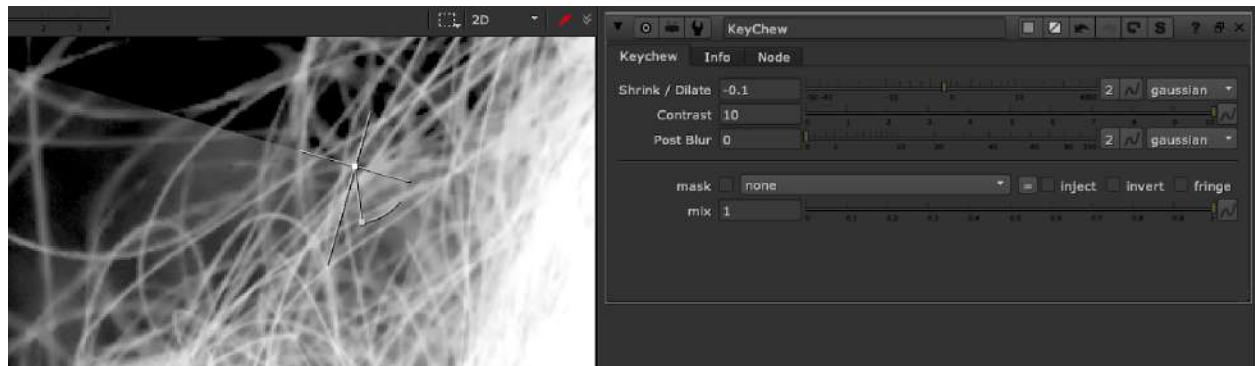
Derek's tool is a simple slider controlled luminance keyer.

Addition features added

- 1.) output options of the result to alpha or rgba
- 2.) smoothing settings which mimic the smoothstep options found in a ramp, smooth, smooth0, and smooth1, using colorLookup curves
- 3.) mask and mix options

KeyChew RK

Author: Rafal Kaniewski - <http://movingimagearts.com>



Download from: <http://movingimagearts.com/nuke-tools-london/>

Or on github: <https://raw.githubusercontent.com/openNuke/toolset/master/keychew.nk>

Dilates and erodes a matte.

Good for subtly eroding in keyed edges.

Can go negative and positive which erodes in and out.

Equipped with mask and mix.

8.) Merge

ContactSheetAuto TL

Author: Tony Lyons www.CompositingMentor.com (based on Ben McEwan's blog post)



Nukepedia link: <http://www.nukepedia.com/gizmos/merge/contactsheetauto>

Full Credit goes to Ben McEwan and his very detailed blog post about powering up your contact sheets. <https://benmcewan.com/blog/2018/08/26/power-up-your-contact-sheets/>

The python script to change your knob defaults on the normal contactSheet node in your menu.py file is already here, posted by Ben:

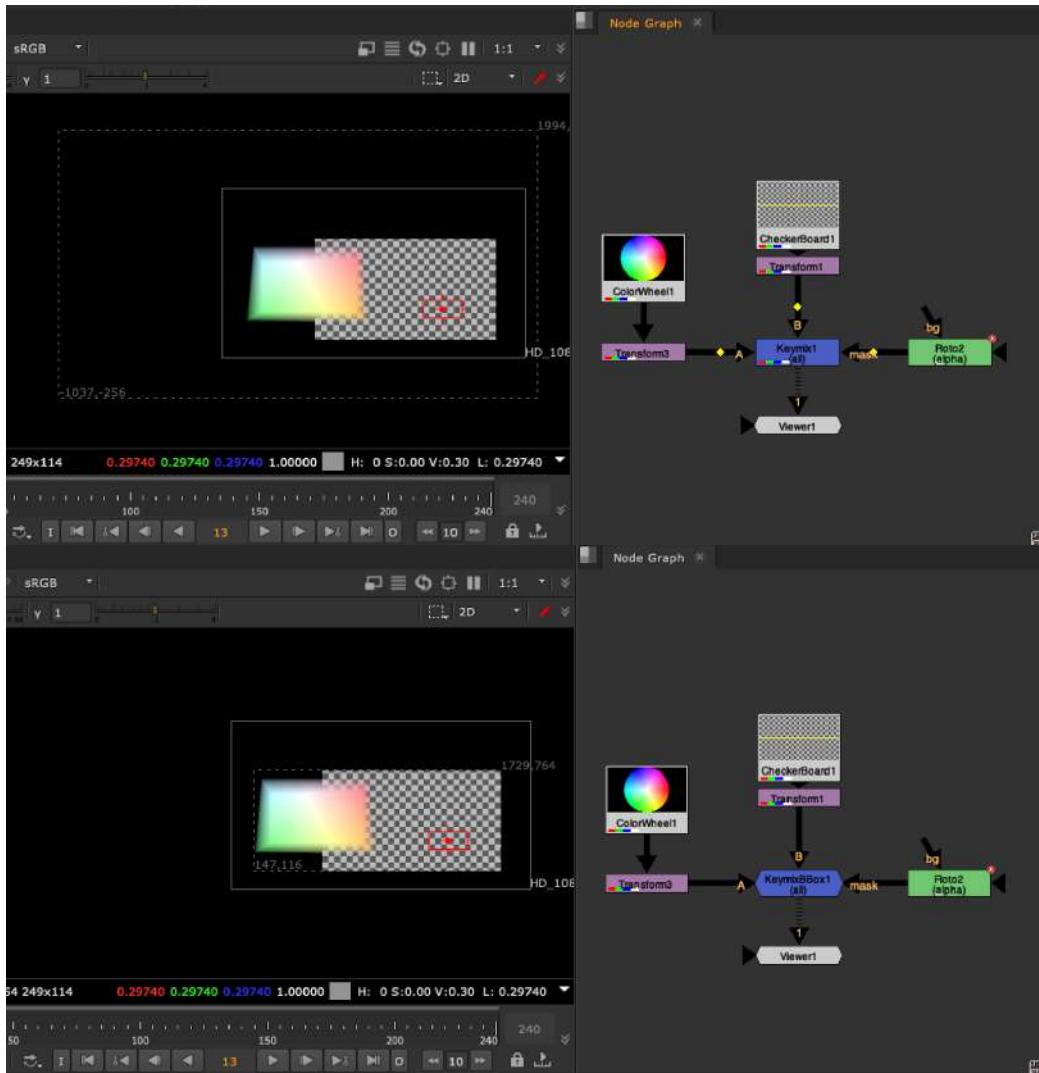
<http://www.nukepedia.com/python/misc/autocontactsheet>

This one is just for people who want to download the expressioned node and add it to their toolsets and not mess with any menu.py files.

Demo video: <https://youtu.be/dqzzT169GAc>

KeymixBBox TL

Author: Tony Lyons - www.CompositingMentor.com

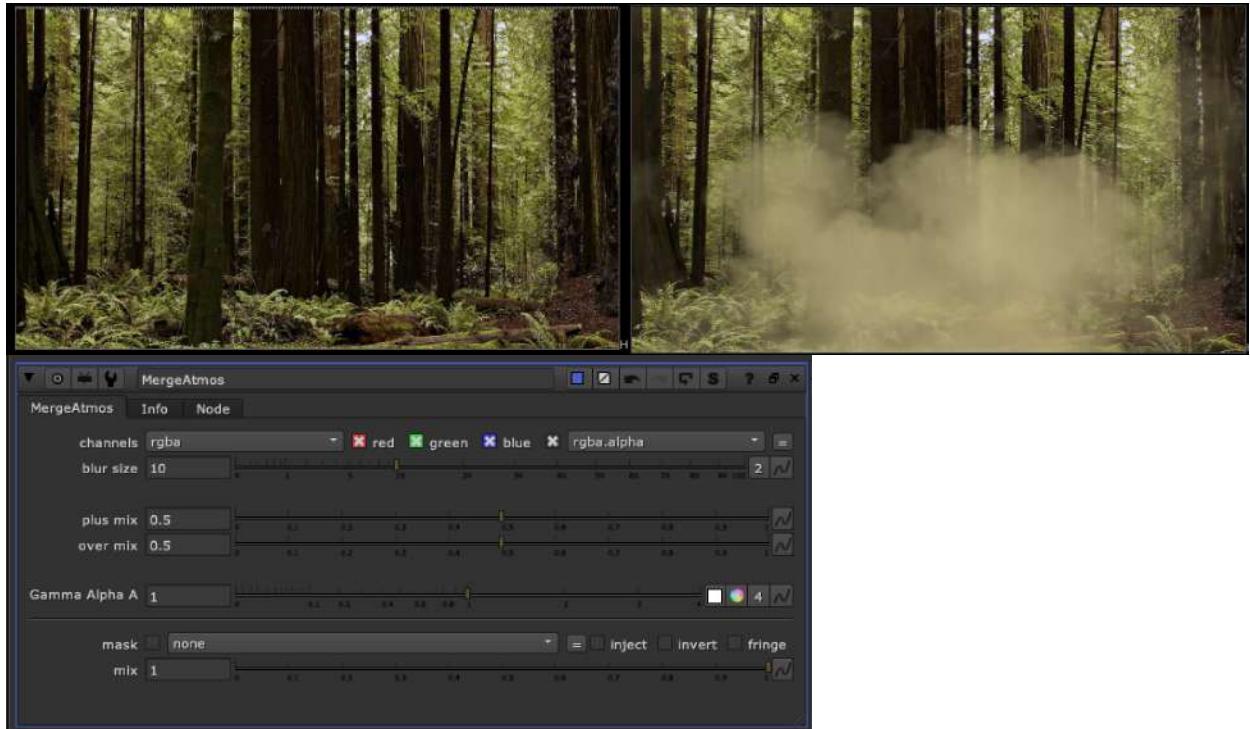


Same functionality as normal keymix, but with slightly better BBox management.

Nuke's keymix node takes into account the bbox's of the A, B, and mask inputs. In most cases you want the mask bbox to be ignored and that the maximum BBox result would be max bbox of A or B, and then an intersection of both A and B when the mask bbox is smaller.

MergeAtmos TL

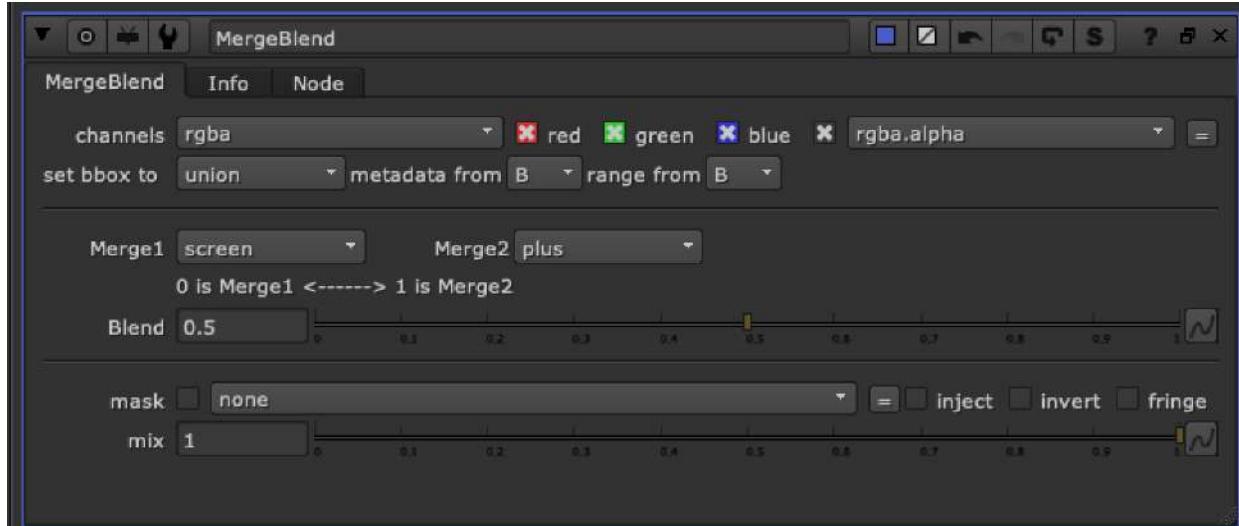
Author: Tony Lyons - www.CompositingMentor.com



MergeAtmos is a merge for smoke, dust, and/or atmospheric effects. It has mixes of a merge(plus) and merge(over) exposed so you can find the right balance. The alpha of the smoke element is also driving a blur node that is simulating a bit of a diffusion effect.

MergeBlend TL

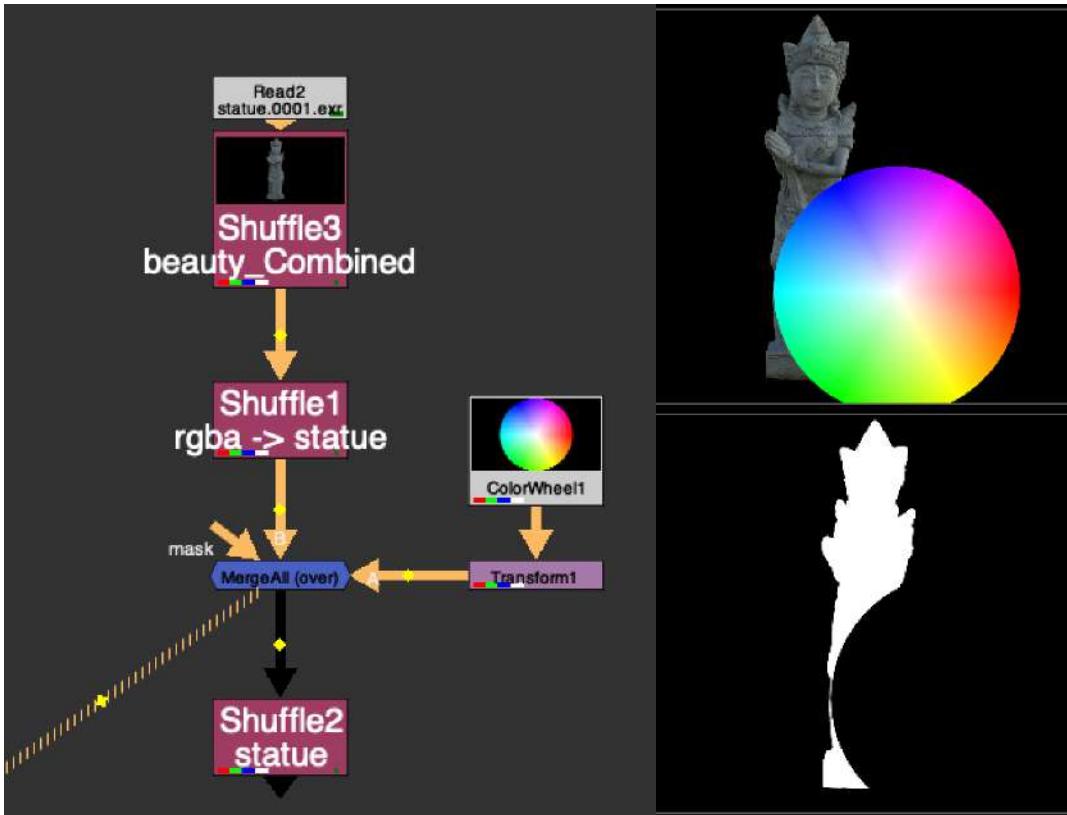
Author: Tony Lyons - www.compositingMentor.com



Select 2 different Merge Operations and Blend between the 2 results

MergeAll AP

Author: Adrian Pueyo - <http://www.adrianpueyo.com/>



Many times when we are using an extra channel in the stream to pull out later to either use as a mask, DI matte, grain matte, etc. we have to manage that channel separately.

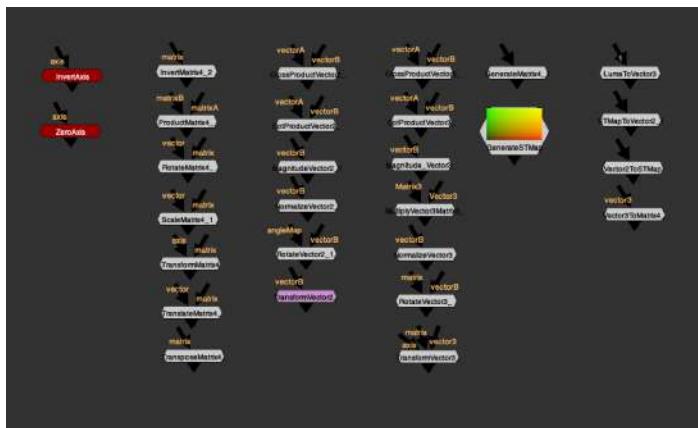
In this example, we have a statue matte channel that we want to pull out later, and we've over'ed a colorwheel. Normally you might have to add a channelMerge and stencil the colorWheel from the statue matte channel separately. MergeAll can save you this step. It basically adds all the channels from A stream to B stream and vice versa, ensuring that before merging, both streams contain the same exact channels. Because the colorWheel in this case now contains a "statue channel" which is black, when you mergeAll, it will merge "black" where the colorWheel is in the statue channel. So when you pull out the statue channel layer, it will appear as though the colorWheel is stenciled from the statue alpha.

This is sort of what you would expect to happen when you "merge All channels" in a merge node, but because sometimes the 2 streams don't have exactly the same channels, sometimes they get ignored. This helps solve that.

9.) Transform

Math Tools MGA-EL

Authors: Mathieu Goulet-Aubin & Erwan Leroy - <http://erwanleroy.com/blog/>



Math Tools combines Mathieu Goulet-Aubin & Erwan Leroy's vector tools into 1 main menu

Original nukedpedia links:

Vector matrix toolset: <http://www.nukedpedia.com/toolsets/transform/vector-matrix-toolset>

Vector Tools: <http://www.nukedpedia.com/toolsets/other/vectortools>

Collaboration Github link: <https://github.com/mapoga/nuke-vector-matrix>

Resources to learn about Vectors and Matrices:

Most tools in this toolset are mathematical tools and require some basic knowledge about Vectors and Matrices for optimal use.

[Math is Fun: Scalar, Vector, Matrix](#)

[Wikipedia: Transformation Matrices](#)

[Nukedpedia: Python Vector and Matrix Math](#)

[Nukedpedia: The Matrix Knob](#)

Erwan Leroy's tool description:

<http://erwanleroy.com/vector-tools-for-nuke-tutorials-and-math/>

<http://erwanleroy.com/vector-tools-for-nuke-tutorials-and-math-part2/>

Introduction:

The toolset is separated into 2 categories. One to operate on vectors3 and one to operate on 4x4 transformation matrices. Every pixel can be worked on independently because each can have its own vector and matrix data.

Math Menu:

Axis Menu:

InvertAxis MT

Inverts an input Axis

ZeroAxis MT

Inverts an input Axis at a specified frame

Matrix4 Menu:

Matrix data is expected to be in the layers; matrix0, matrix1, matrix2 and matrix3.

Layers are automatically created by the nodes.

The transformation matrix is stored as followed:

LayerName: red[0], green[1] ,blue[2], alpha[3]

matrix0: [0, 1, 2, 3,

matrix1: 0, 1, 2, 3,

matrix2: 0, 1, 2, 3,

matrix3: 0, 1, 2, 3]

InvertMatrix4 MT

Invert a pixel based Matrix4 (Defined as layers matrix0, matrix1, matrix2 and matrix3)

Returns the inverted input 4x4 matrix.

input: matrix

output: matrix

ProductMatrix4 MT

Multiply two pixel based Matrix4 (Defined as layers matrix0, matrix1, matrix2 and matrix3)

Returns the dot product of matrixB * matrixA.

The matrix order is very important as switching A and B will give different results.

inputs: matrixB, matrixA output: matrix

RotateMatrix4 MT

Returns the input 4x4 matrix rotated by the input vector.

The rotation unit knob specify the rotation unit(degrees or radians) of the input vector.

The rotation order knob specify the order in which the axes are rotated

inputs: matrix, vector

output: matrix

ScaleMatrix4 MT

Scale a matrix4 using a control channel (rgb from vector input) for which each channel is considered as a scalar for x, y and z

Returns the input 4x4 matrix scaled by the input vector.

inputs: matrix, vector

output: matrix

TransformMatrix4 MT

Returns the input 4x4 matrix transformed by the the node's knobs.

If no matrix is given, the transformations will be made on a identity matrix.

input: matrix

output: matrix

TranslateMatrix4 MT

Translate a matrix4 using a control channel (rgb) for which each channel is considered as a scalar for x, y and z

Returns the input 4x4 matrix translated by the input vector.

inputs: matrix, vector

output: matrix

TransposeMatrix4 MT

Transpose a pixel based Matrix4 (Defined as layers matrix0, matrix1, matrix2 and matrix3)

Returns the input 4x4 matrix transposed.

Values of the matrix are mirrored diagonally.

inputs: matrix

output: matrix

Vector2 Menu: (for 2D vectors)

Vector data is expected to be in the layer rgb.

(r, g) = (x, y)

Operations like vector addition, subtraction, multiplication and division can be done with a standard merge or merge expression node.

CrossProductVector2 MT

Calculates the cross product of 2 Vector2 inputs.

Returns the cross product between 2 vectors.

input: vectorA, vectorB

output: vector

DotProductVector2 MT

Calculates the dot product of 2 Vector2 inputs.

Returns the dot product between 2 vectors.

The resulted output is repeated to fill up the output vector.

inputs: vectorA, vectorB

output: vector

MagnitudeVector2 MT

Calculate the magnitude (scalar) of an input Vector2.

Returns the magnitude of a vector. In other words, the vector's length.

The resulting output is repeated to fill up the output vector.

input: vector

output: vector

NormalizeVector2 MT

Normalize the magnitude of a Vector2 (to be of magnitude 1)

Returns the normalized vector.

Scaling the vector so that its length equal 1.

input: vector

output: vector

RotateVector2 MT

Rotate a 2D vector on the same 2D plane.

A utility to rotate 2D vectors such as motion vectors, and flip them if necessary

TransformVector2 MT

Transforms an image assuming it is a motion vector in RGBA.

Compared to a regular transform, this will edit the pixel colors to compensate for vector direction and magnitude.

Warning: This node breaks concatenation.

Like Nuke's default transform, but will rotate vectors and scale them accordingly.

Vector3 Menu: (for 3D vectors)

Vector data is expected to be in the layer rgb.

(r, g, b) = (x, y, z)

Operations like vector addition, subtraction, multiplication and division can be done with a standard merge or merge expression node.

CrossProductVector3 MT

Calculates the cross product of 2 Vector3 inputs.

Returns the cross product between 2 vectors.

input: vectorA, vectorB

output: vector

DotProductVector3 MT

Calculates the dot product of 2 Vector3 inputs.

The resulting output is repeated to fill up the output vector.

inputs: vectorA, vectorB

output: vector

MagnitudeVector3 MT

Calculate the magnitude (scalar) of an input Vector3.

Returns the magnitude of a vector. In other words, the vector's length.

The resulting output is repeated to fill up the output vector.

input: vector

output: vector

MultiplyVector3Matrix3 MT

Multiply (transform) a Vector3 by a Matrix3. This is the equivalent of applying Rotation/Scale/Skew from a Matrix to the vector.

A Matrix4 can be used, but the last row/column will be ignored.

NormalizeVector3 MT

Normalize the magnitude of a Vector3 (to be of magnitude 1)

Returns the normalized vector.

Scaling the vector so that its length equal 1.

input: vector

output: vector

RotateVector3 MT

Rotate a Vector3 in 3 dimensions.

A utility to rotate vectors such as motion vectors, and flip them if necessary.

TransformVector3 MT

Transform a Vector3 in 3 dimensions.

Generate Menu:

GenerateMatrix4 MT

Generate a Matrix4 based on a Matrix Knob. (Defaults to an identity matrix)

Matrix data is expected to be in the layers; matrix0, matrix1, matrix2 and matrix3.

Layers are automatically created by the nodes.

The transformation matrix is stored as followed:

LayerName: red[0], green[1] ,blue[2], alpha[3]

matrix0: [0, 1, 2, 3,

matrix1: 0, 1, 2, 3,

matrix2: 0, 1, 2, 3,

matrix3: 0, 1, 2, 3]

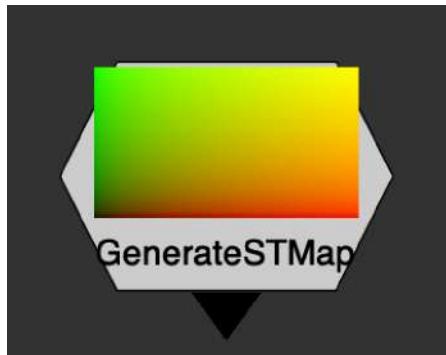
Returns an 4x4 identity matrix.

The input bg is used to give a format to the matrix.

input: bg

output: matrix

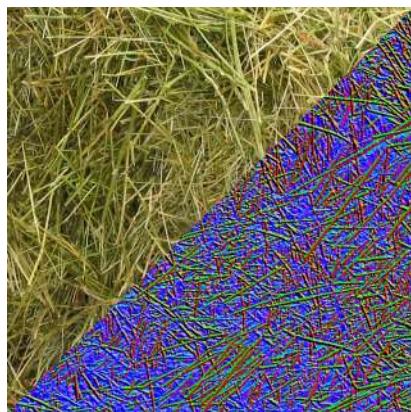
GenerateSTMap MT



Generates a default UV map at any resolution. Can be used to run through a lens distortion in another software or any sort of distortion that can then be re-applied with an STMap node.
Options for generating overscan STMap data

Convert Menu:

LumaToVector3 MT



Performs a Sobel filter on the Luminance channel of an image to extract an approximation of a Normal map. For a mathematical conversion of a displacement map to normals, do not use Details separation. Converts any image to normals, using its Luma Channel. Provides most accurate results used on displacement maps or Zdepth passes.

STMapToVector2 MT

Transforms a distorted UV Map to Motion Vectors corresponding to the distortion.
Converts a distorted UV map to motion vectors.

Vector2ToSTMap MT

Converts 2D Vectors to an UV map.

Vector3ToMatrix4 MT

Converts 3D vectors to a 4x4 Matrix
Matrix data to be in the layers; matrix0, matrix1, matrix2 and matrix3.
Layers are automatically created by the nodes.
The transformation matrix is stored as followed:
LayerName: red[0], green[1] ,blue[2], alpha[3]

matrix0: [0, 1, 2, 3,
matrix1: 0, 1, 2, 3,
matrix2: 0, 1, 2, 3,
matrix3: 0, 1, 2, 3]

Returns an 4x4 identity matrix.

vector3DMathExpression EL

Author: Erwan Leroy - <http://erwanleroy.com/blog/>

A NoOp node with expressions to calculate a 3D vector between 2 3D points, as well as its magnitude.

<http://erwanleroy.com/vector-tools-for-nuke-tutorials-and-math/>

<http://www.nukepedia.com/toolsets/other/vectortools>

Vectors_Direction EL

Author: Erwan Leroy - <http://erwanleroy.com/blog/>

An utility to rotate 2D vectors such as motion vectors, and flip them if necessary.

angleRad = radians(parent.rotation)

$r = r * \cos(\text{angleRad}) - g * \sin(\text{angleRad})$

$g = r * \sin(\text{angleRad}) + g * \cos(\text{angleRad})$

<http://erwanleroy.com/vector-tools-for-nuke-tutorials-and-math/>

<http://www.nukepedia.com/toolsets/other/vectortools>

Vectors_to_Degrees EL

Author: Erwan Leroy - <http://erwanleroy.com/blog/>

Gives you an angle value (0-360) for a given channel. Can be used to generate Anisotropy maps from vectors for CG

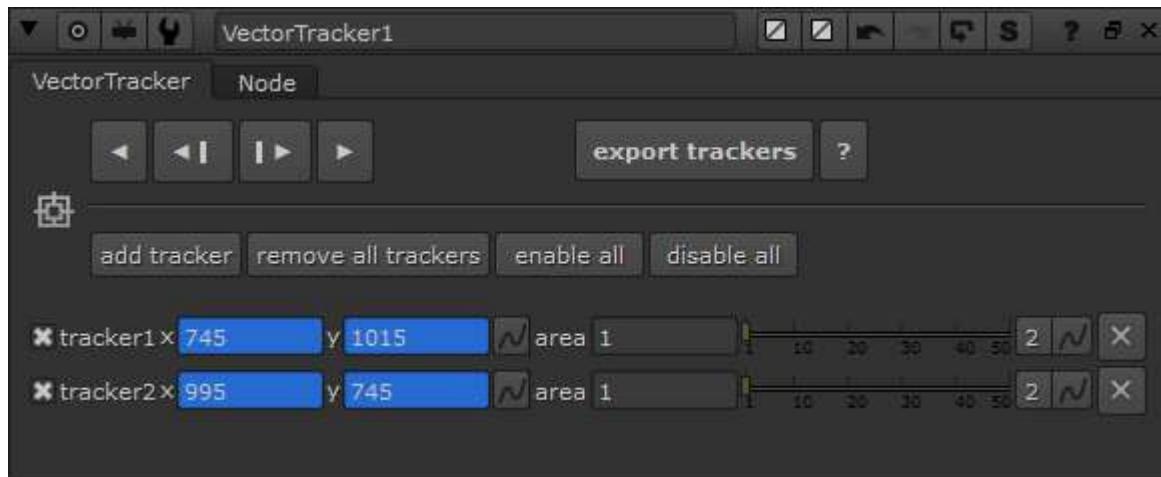
<http://erwanleroy.com/vector-tools-for-nuke-tutorials-and-math/>

<http://www.nukepedia.com/toolsets/other/vectortools>

VectorTracker NKPD

Author: Jorrit Schulte

Author's website: <http://www.jorritschulte.com/nuke-tools/>



This is a tracker gizmo that uses vector information rather than image data to track points. This is useful for certain tracking jobs that are otherwise hard to accomplish. Think of shots that constantly shift focus, or tracks on objects that deform.

This gizmo works with both classic nuke vectors or smartvector data.

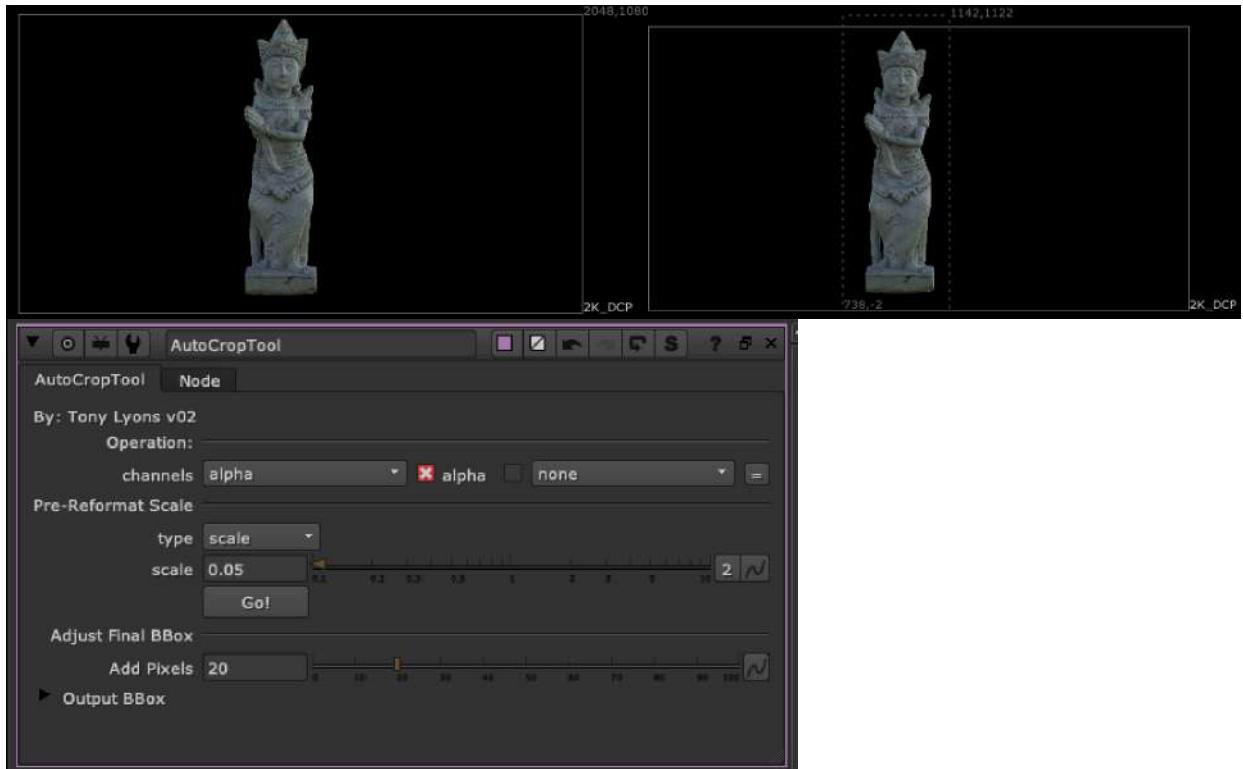
If you use smartVectors live, make sure to add a VectorToMotion node after the smartVector and plug the vector input into this to register.

Simply render out either classic nuke vectors or smartvectors, and use this tool to generate tracker points. You can try using a VectorGenerator node to generate live vectors instead, but due to the way nuke and python sample image data this won't always work. The points are exported to a regular Tracker node.

It is looking for motion, forward, backward, backward channels, or a vectorGenerator / smartVector (plus VectorToMotion) nodes for the vector input.

AutoCropTool TL

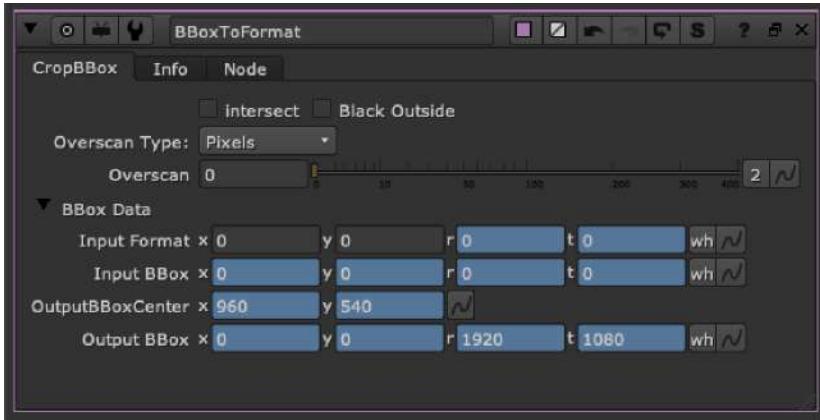
Author: Tony Lyons - www.CompositingMentor.com



AutoCropTool runs a curvetool node autocrop process on a scaled down single channel version of the input image. Use this to generate a quick bounding box on CG renders or other elements without a defined bounding box. Keeping your bounding box tight and isolated to only the important part of the frame will save processing time in nuke.

BBoxToFormat TL

Author: Tony Lyons - www.CompositingMentor.com



BBoxToFormat sets your bbox exactly to the input format.

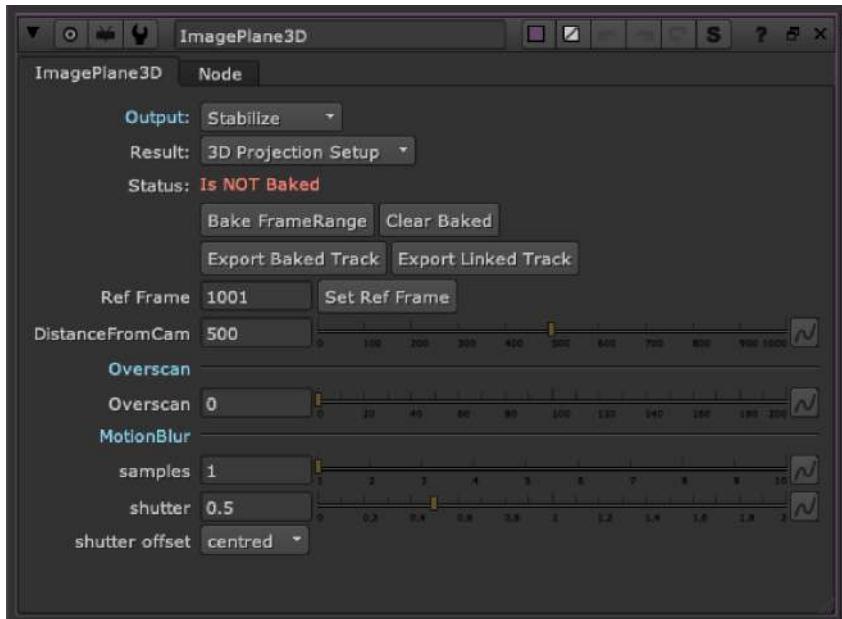
You also have options for either a percentage overscan or pixel based overscan. This is a good way to manage your bbox while keeping some extra overscan for distortion or cameraShake, etc when you are using CG with overscan or additional elements.

The intersect option will be the intersection of the input format with the input bbox.

Exposed data for input format, input bbox, output BBox Center, and Output BBox in a group tab because sometimes it's useful to reference.

ImagePlane3D

Author: Tony Lyons



Based off the great imagePlane tool by Frank Rueter on nukepedia:

<http://www.nukepedia.com/gizmos/transform/imageplane>

The original imagePlane node seems to have a problem with stabilizing the image, and then reverting the stabilized image back to the original position by matchmoving. Unfortunately, not being about to reliably restore the stabilized image back to the original position meant your rotos and prep work or whatever positioning you were doing was not guaranteed to matchmove back into the original footage. This tool aims to solve this problem and offering a few extra features.

Output: Choose between **stabilize** and **matchmove**

Result:

3D Projection Setup - this is using 3d nodes, cards and camera projections to guarantee no funkiness and that the stabilize/matchmove process will be reversible. Unfortunately the 3d projection setup will be the slowest option, but it will be a clean preview

card3D setup - this is using the same method as the original imagePlane node, for some reason, this method creates the problem of stabilizing and reversing the movement when matchmoving. It is however very fast because there is no 3D scanline render and calculates as a 2d cornerpin. This will either give you identical or very similar results to the 3D projection setup but is prone to this “unreversible” result. Good mode finding the stabilized distance quickly.

Live (Reconcile 3D) - This is by far the fastest calculation for this stabilization process and is recommended for quick previews and finding the stabilization distance. However when using nuke’s reconcile 3D node with live points, it gets very buggy. Some frames are black, other frames ‘explode’ and seem to lose the 3d points for reference. Proxy mode or lower resolution preview will be buggy or not work. In theory it should be the exact same result as the 3d Projection setup, so the stabilization/matchmove workflow is in tact, but it’s very buggy, especially when rendering. Do not leave on this live mode, purely use it as a quick preview to find the stabilization distance and ensure accuracy.

Baked (Cornerpin) - When you bake the framerange, it will switch to this mode in order to save calculation time, and to eliminate the bugginess of the stabilization/matchmove workflow from the “live” previews

Status: will tell you the status of the node, whether it is Live or baked (and what framerange)

Ref Frame: This frame is the stabilization pivot frame of the stabilization/matchmove and will be unaffected / have zero transformations

Distance from Cam: This number is the distance from camera in nuke units to place the stabilization card. Sometimes you roughly know the distance from the camera that the object you are trying to stabilize is, and other times you can just eye ball distance by moving away from your reference frame in the timeline and changing the distance until object remains in the same position in the screen as it was in the reference frame. Using the color picker to ‘save’ a position can sometimes help you line up the object position over multiple frames.

Bake FrameRange: This will create a baked cornerpin based off the ref frame and distance from camera. It’s recommended to run this once you are happy with your stabilization result because it will not only be less prone to nuke’s bugginess from the other live setups but it will also be a lot faster since it is baked and not calculating anything live with expressions or 3D setups. Warning: this may take awhile as it is running a reconcile3D for each of the 4 points over the desired framerange. Large frameranges can take some time, and unfortunately there is no feedback on how long it is taking, so nuke may appear to freeze. Just be patient and it will work, normally it is a very fast process.

Clear baked: Clears the keyframes and reverts the node back to live. Sets to default

Export Baked Track Export Linked Track: These 2 buttons are acting like the export cornerpin button in nuke's 2d Tracker node. They will export a cornerpin node that is stabilized or matchmoved (depending on the output you have selected). This way you can use the cornerpin elsewhere in your script and won't have to duplicate this 'heavy' node. Please note, expression linked cornerpin, while convenient, will be slower in nuke to calculate than a baked result, so choose wisely.

There are some additional settings i have added to the cornerPin node for convenience of use for you.

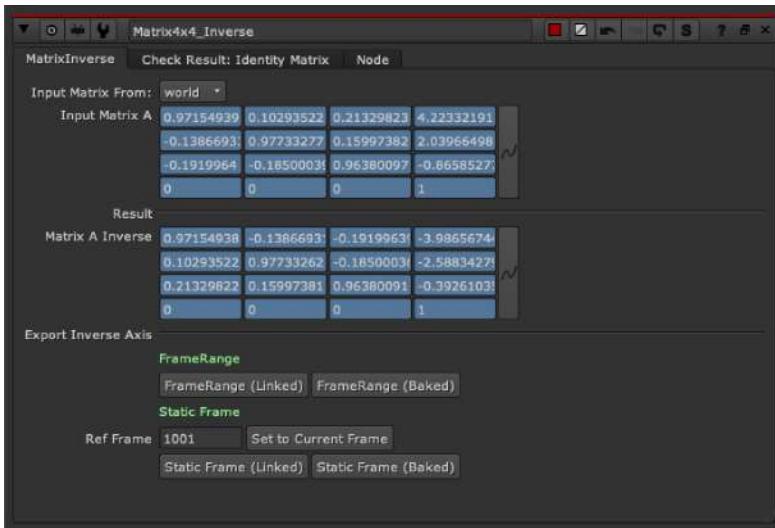
There are controls for settings and changing the reference frame that you want. And also 2 buttons to switch the cornerpin between stabilize and matchmove.

Overscan: A simple setting to manage your bbox and set an overscan allowance (in pixels). This will be important when distorting and undistorting your image. Once you bake the cornerpin this shouldn't matter too much, mostly for the 3d projection setup scanline render overscan

Motion Blur: Ability to add motionblur for your matchmove result based on either samples on the scanline render or motion blur knob on the card3D or cornerpin node. Depends on the 'result' mode you choose

Matrix4x4_Inverse TL

Author: Tony Lyons - www.CompositingMentor.com



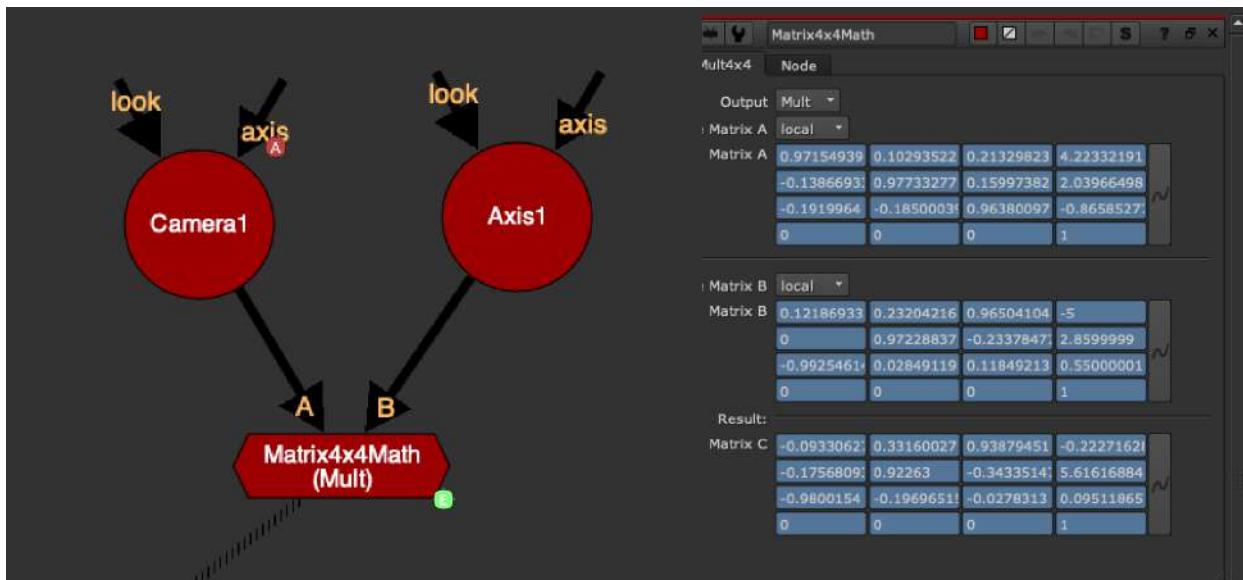
Matrix4x4_Inverse is a node that takes a node with a 3d transformation matrix, such as cameras, cards, axis, transformGeo, etc, and produces the inverse matrix. This inverse matrix can be used to return the 3D object to the origin (aka to the identity matrix). Sounds really complicated but an easy way to comprehend it is this node stabilizes and returns the 3d object to the origin, where you can then transform it to a new position. Like a 3D ‘Stabilize’ and then ‘matchmove’ technique.

Plug the node into the 3d node or chain of axis’ or cameras with transformGeo, etc. Be sure to choose between the input matrix being either the world matrix or local matrix. Local matrix is only taking into consideration the transformations of that specific input node. World matrix takes into consideration the actual position of the object in 3D space, or the ‘totality’ of all concatenated 3d transformations (real position).

You can choose to export an Axis node with the inverse matrix either linked or baked and either over a framerange or on a single reference frame.

Matrix4x4_Math TL

Author: Tony Lyons - www.CompositingMentor.com



Matrix4x4_Math does some basic matrix math between matrix A and matrix B.

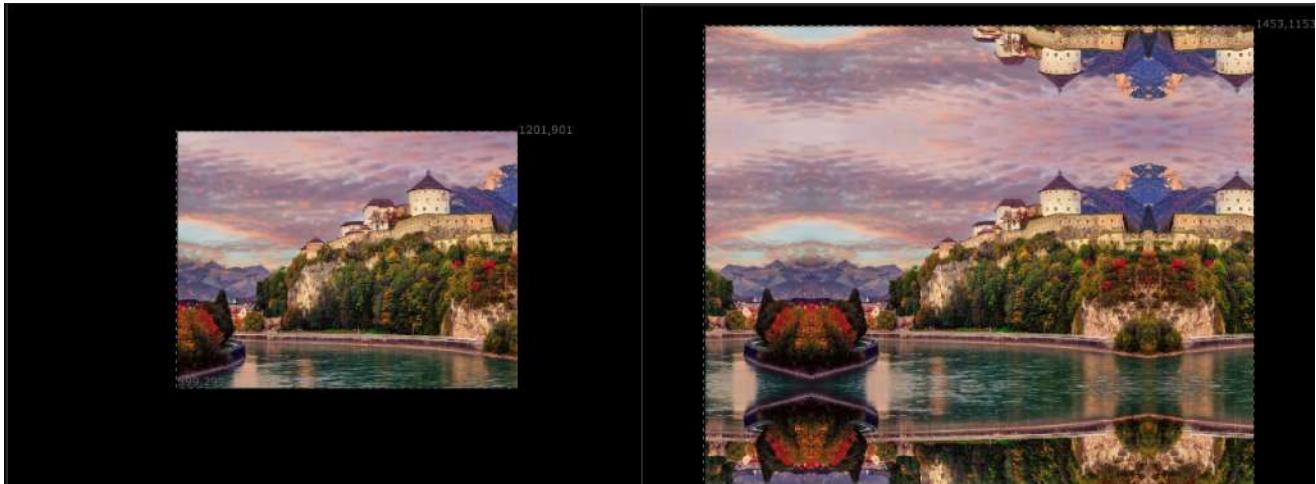
Operations are: Add, Subtract, Mult

Choose between local and world matrices of the inputs.

You can expression link the resulting matrix (matrix C) to other nodes' local matrix

MirrorBorder TL

Author: Tony Lyons - www.CompositingMentor.com



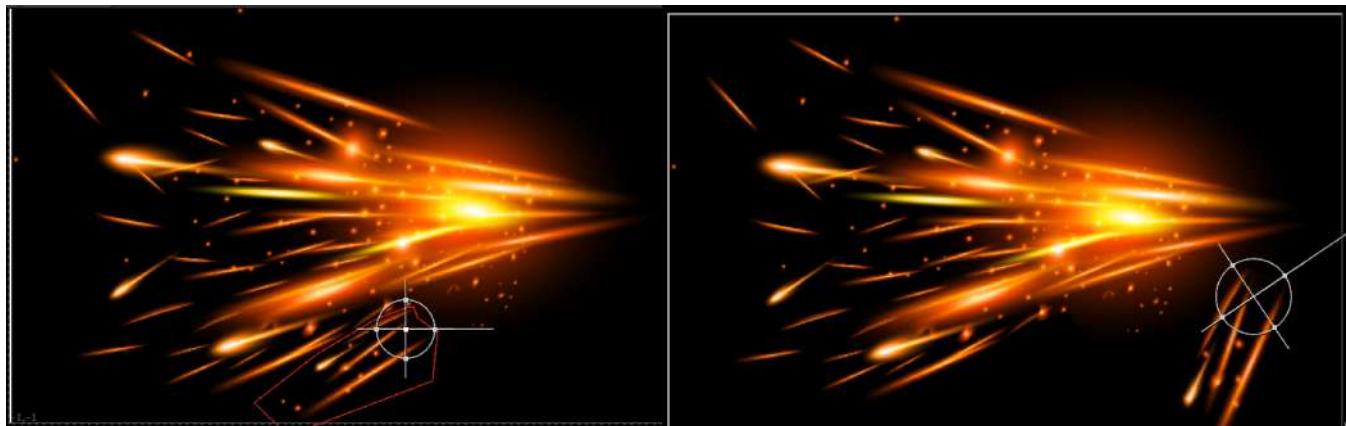
MirrorBorder mimics Adobe AfterEffect's Motion Tile effect. It will tile and mirror the frame around the border of the input format, mostly to be used to produce extra edge pixel details for camera shake. This can avoid either a black edge or stretchy pixels around the edge of frame when adding camera shake.

The tile region is either the input format or input bbox. Tile amount is expansion amount in pixels.

Choke edges refers to cropping the edges in a bit before mirroring them. Left, Right, Top, Bottom choke sliders available.

TransformCutOut TL

Author: Tony Lyons - www.CompositingMentor.com



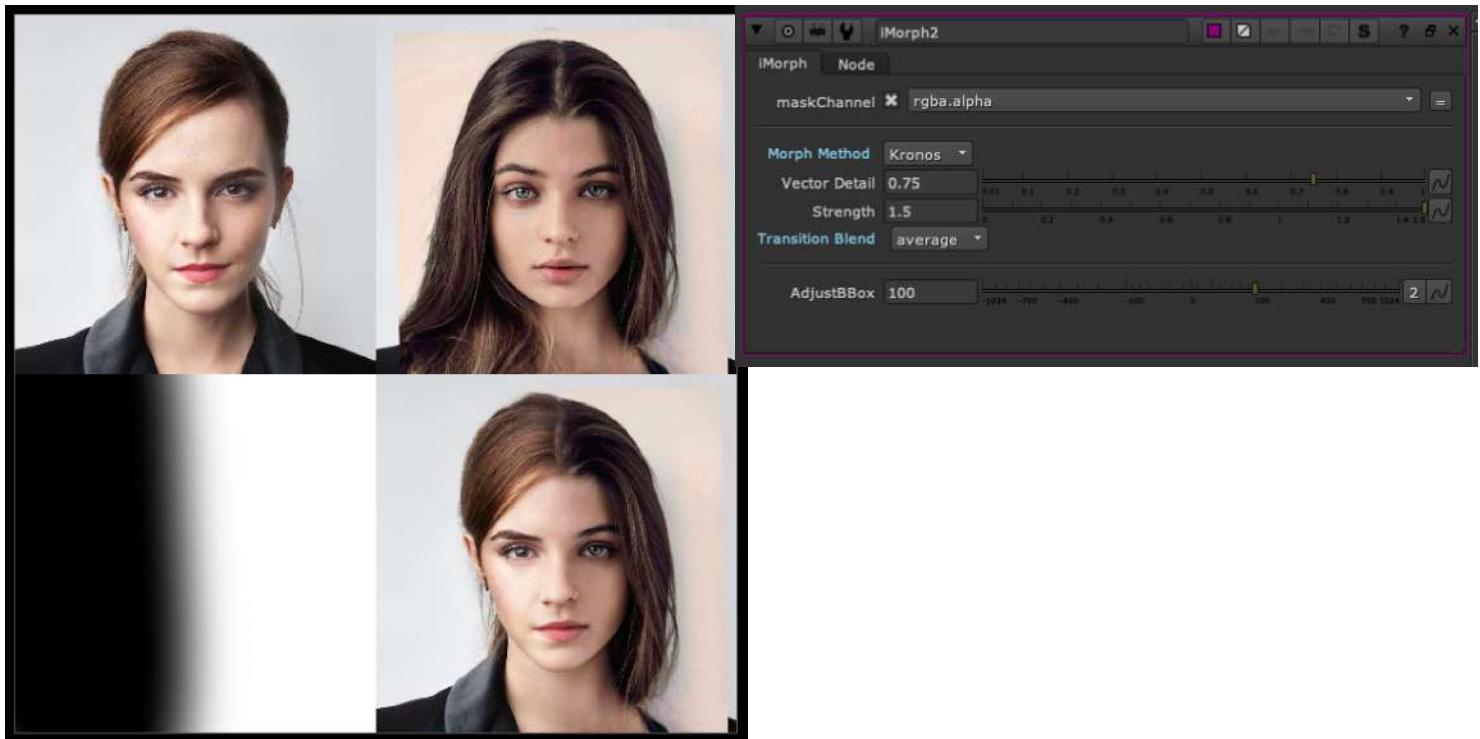
TransformCutOut takes the masked area of the image and transforms it and overs it (disjoint-over) back onto the image. Useful for masking an object and placing it somewhere else.

This is different to the transformMasked node which essentially moves the image around 'inside' of the masked area. This node will cut out the masked area, and over it back after the transformation, leaving a hole where the original mask was.

Buttons to set center pivot to either the center of the input format or the center of the mask bounding box, which is meant to compliment rotoshapes that have their own BBox that you will likely use as the mask input. So this option will snap the center pivot to the center of the rotoshape/mask input.

IMorph TL

Author: Tony Lyons - www.CompositingMentor.com and Adrian Pueyo www.AdrianPueyo.com



IMorph is a spin-off of MorphDissolve from Erwan Leroy and SPIN FX gizmo set. It's also inspired by the TimeMachine gizmo by ivan busquets. Here are the links to those nodes:

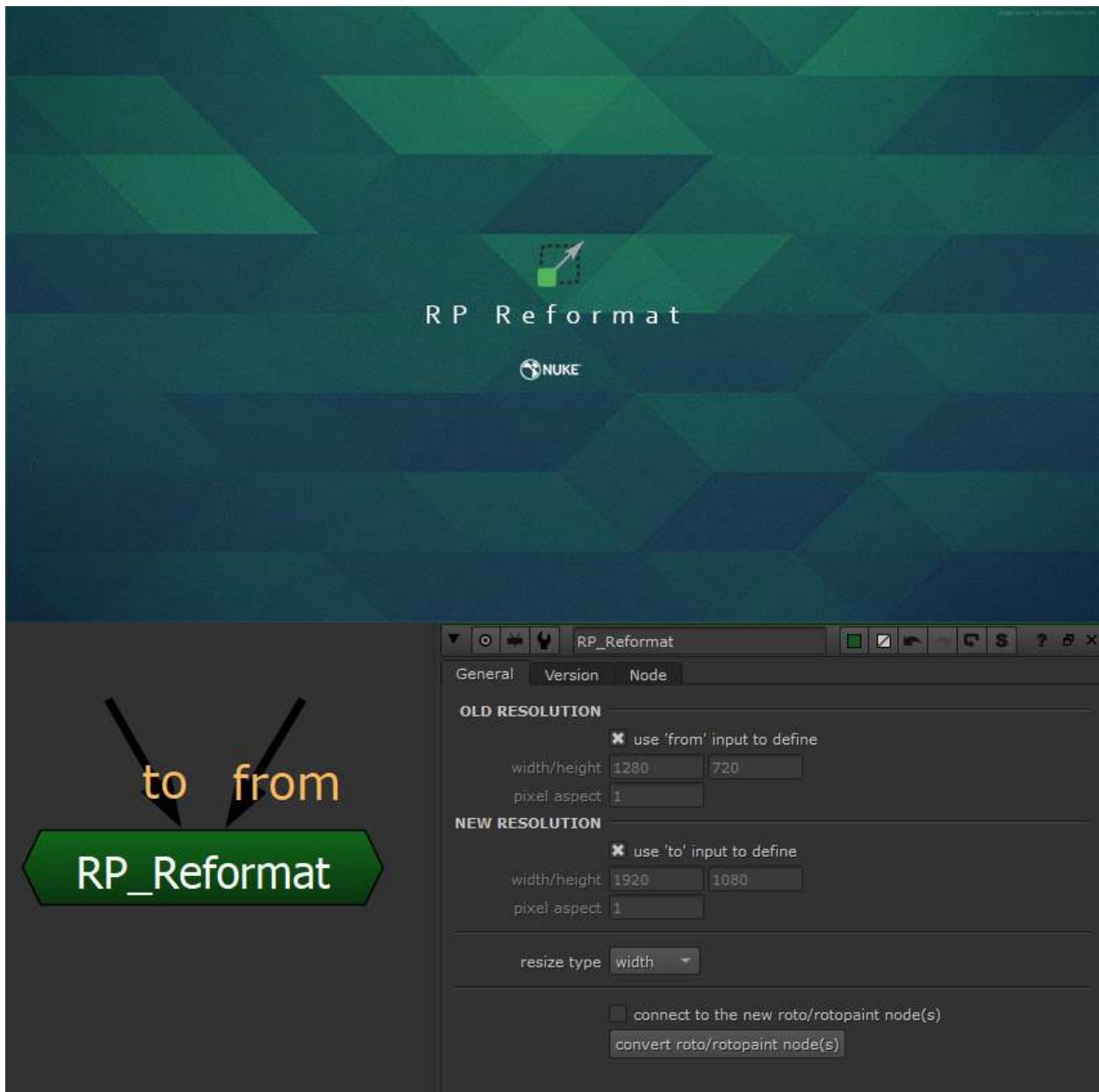
http://www.nukepedia.com/gizmos/transform/morph_dissolve

<http://www.nukepedia.com/gizmos/time/timemachine>

MorphDissolve uses 2 images and has a slider from 0-1 to determine how much morphing there is. IMorph, similar to IBlur or ITransform, uses a Mask input to determine where image A is shown and where image B is shown and then morph-dissolves in steps from A to B in between, which produces really smooth transitions from 1 image into another.

RP_Reformat MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Nukepedia link: <http://www.nukepedia.com/gizmos/draw/rp-reformat>

FaceBook Link: <https://m.facebook.com/MJTlab/posts/628051737776651>

Download Mark Joey Tang's entire Toolset: <http://bit.ly/menupy>

Video about this tool : <https://youtu.be/vGZ6kNnOcTs>

Reformat Roto & RotoPaint node's vector data without resolution issue. Keep the same result on any paint strokes. Support all kinds of splines, brushes and aspect ratio reformat.

How to use:

Fill in the old resolution of the Roto/RotoPaint was done before.

Fill in the new resolution.

Selected which type of resize to process (This depends on how's the plate resize)

Select all the Roto/RotoPaint node(s) *Support mult-select

Click 'convert roto/rotopaint node(s)' . Then the new generated node will be placed next to the original node(s)

* Entire process will NOT modify the original node

** The resize data will replaced on individual shape elements :

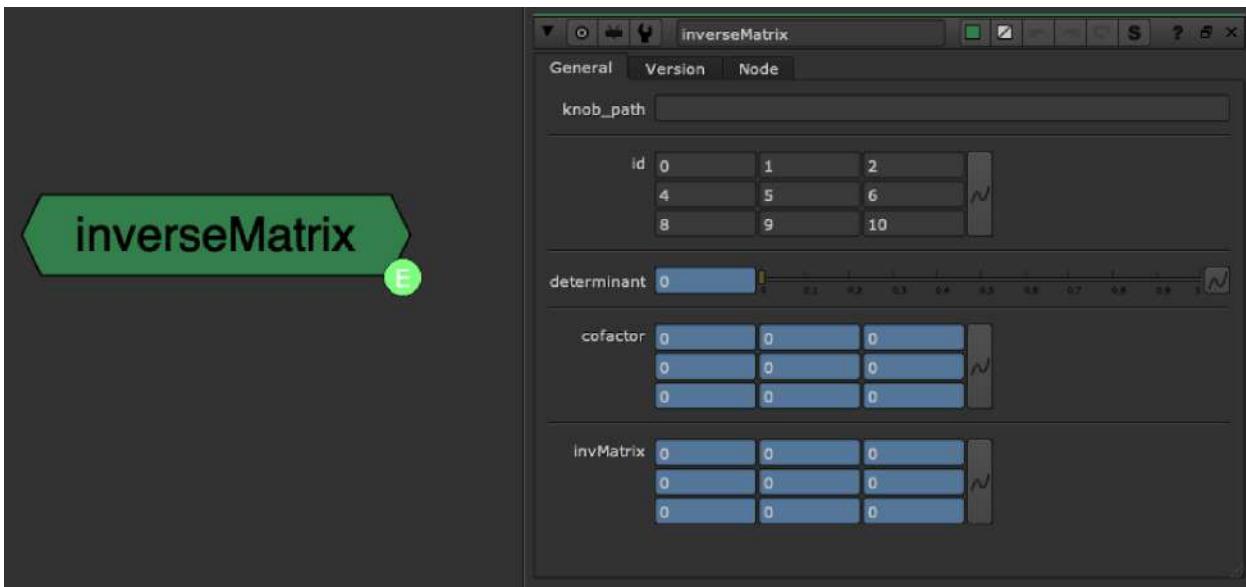
Spline : translate, scale, center & feather

Stroke : translate, scale, center, source translate, brush size, brush space & effect

*** Resize process will not touch any data on Layer.

InverseMatrix MJT

Author: Mark Joey Tang - www.facebook.com/MJTLab



Download Mark Joey Tang's entire Toolset: <http://bit.ly/menupy>

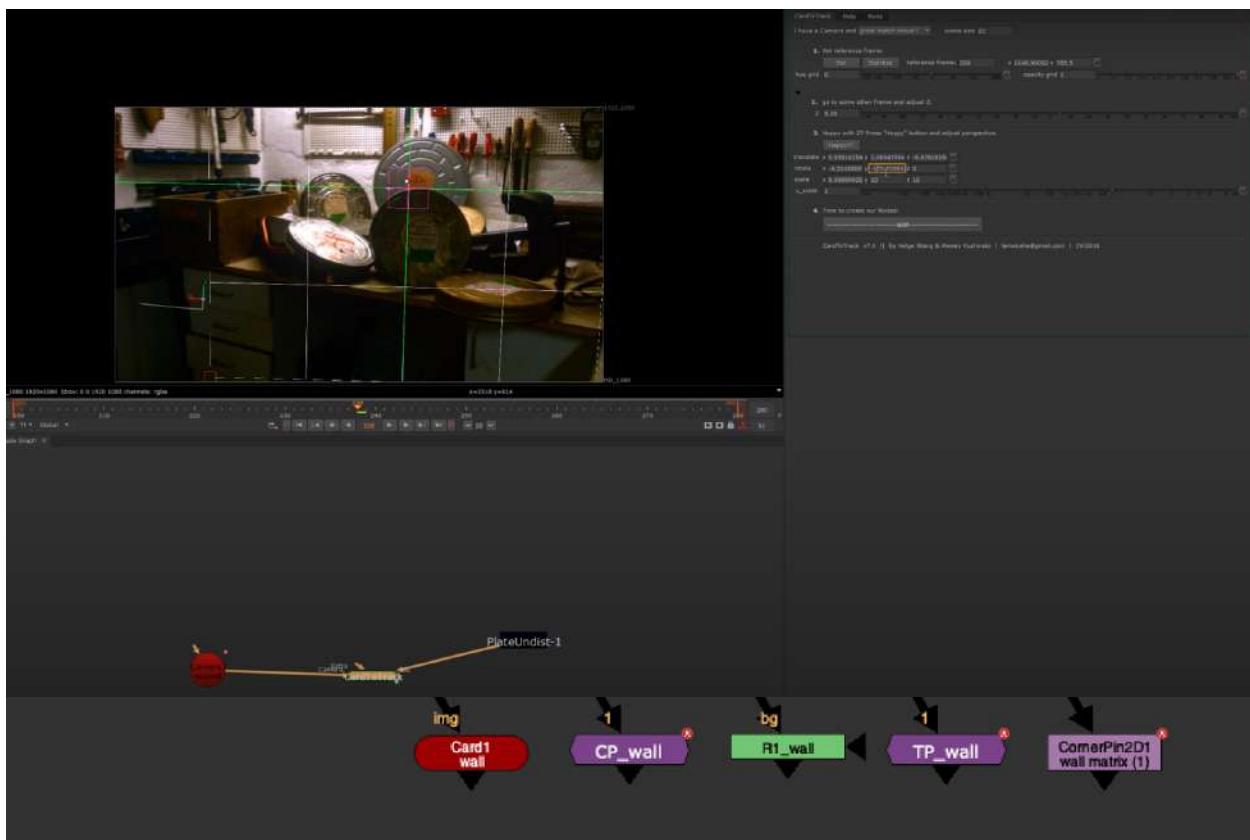
Inverse Matrix provide the calculate output of matrix inverse, which is c44matrix and colorMatrix doesn't show the result. It handy for using this for deep expression which is heavily use in deepExpression.

How to use :

- fill in knob_path of the node with matrix. e.g. camera1.world_matrix
- id is reassign 4x4 matrix to 3x3 matrix. Default value already good for rotation and scale inverse
- don't have to do anything for the rest of the knob. Inverse matrix result will be shown in invMatrix knob

CardToTrack AK

Author: Alexey Kuchinski & Helge Stang



Nukepedia link: <http://www.nukepedia.com/python/3d/cardtotrack>

Video Demo: https://youtu.be/N_M2IJWpe4

Finding position in 3D space based on Geometry, World position pass, Deep or manual method. Reconcile 3D position to CornerPin, Transform and Roto nodes.

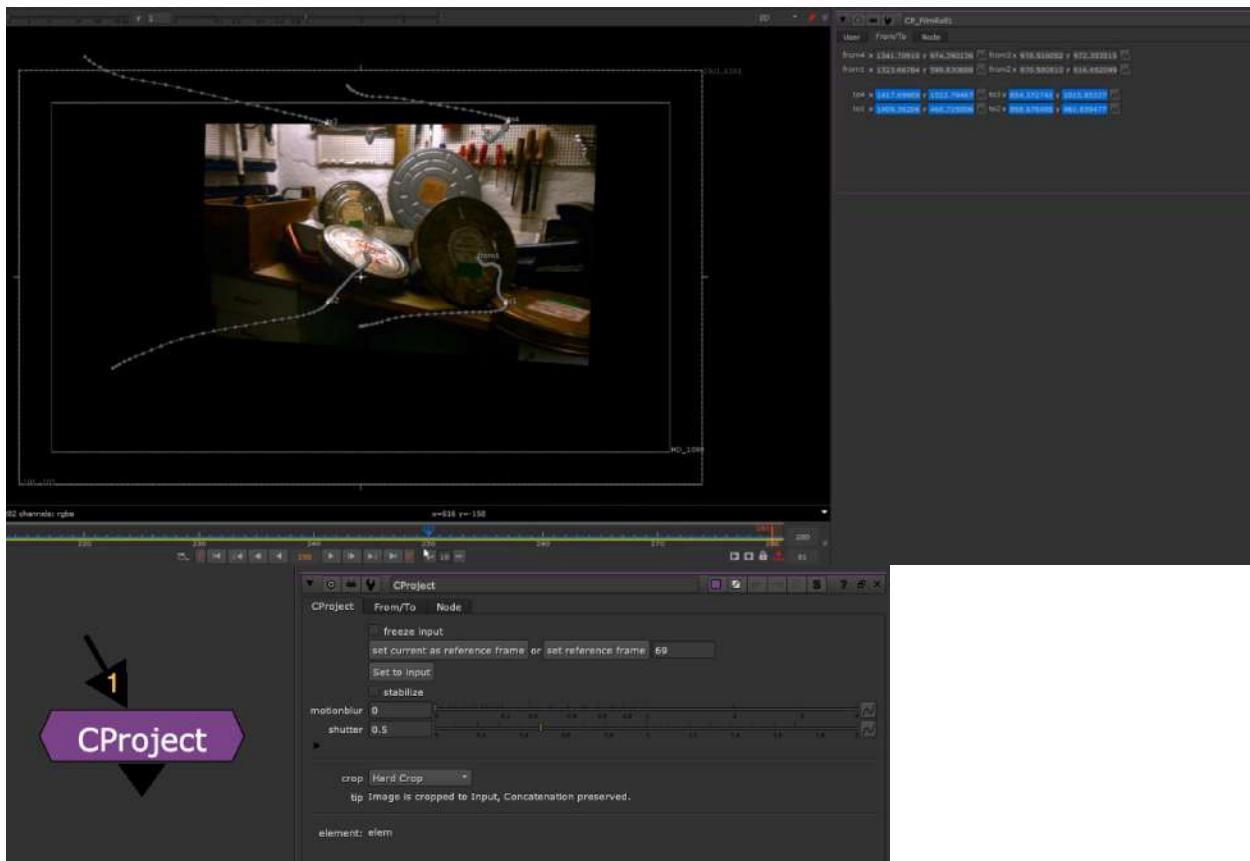
*Calculation from an Axis: Many times we provided with an animated Axis, in previous version i was copying values from an Axis to the card and calculating, the problem is that if by default an axis had an unfortunate rotation like 90 degrees to the camera it was impossible to adjust it. With the Axis option you can connect a few Axes together and Card to track will calculate the animation from the world matrix of the Axis you connected to with CardToTrack.(Supports animated Axes)

*Calculation from Deep: Connect your Deep footage to the Extra input. (The position sampled on the reference frame and can not be animated)

*UI changes: Knob Groups are presented only if they are relevant.

CProject AK

Author: Alexey Kuchinski



Nukepedia link: <http://www.nukepedia.com/python/3d/cardtotrack>

Video Demo: https://youtu.be/N_M2lJWpe4

CornerPin matchmove/stabilize meant to be used with CardToTrack tool.

Allows you to set frame, switch between stabilize and matchmove, add motion blur, and 3 different BBox management settings:

Hard Crop - (reformat node - concatenation is preserved)

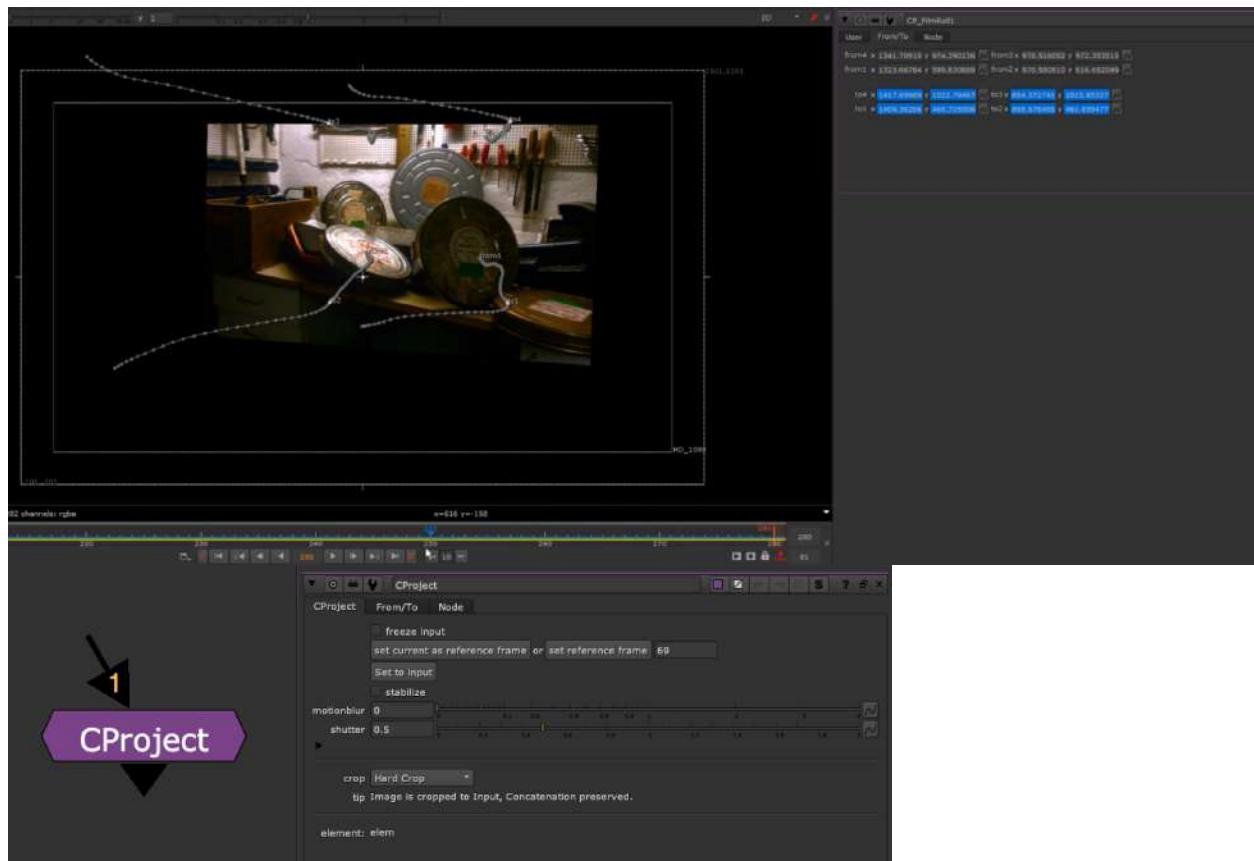
Adjustable crop - breaks concatenation but allows for adjustable bbox

No Crop - no crop applied at all, concatenation preserved but bbox can get quite big

Set to input: This will distort the image with the corner pin to fit the format, it is the same as projecting the image on a card and rendering in UV space

TProject AK

Author: Alexey Kuchinski



Nukepedia link: <http://www.nukepedia.com/python/3d/cardtotrack>

Video Demo: https://youtu.be/N_M2lJWpe4

TProject is similar to CProject but uses a Transform instead of a CornerPin to do a basic single point track.

Allows you to set frame, switch between stabilize and matchmove, add motion blur, and 3 different BBox management settings:

Hard Crop - (reformat node - concatenation is preserved)

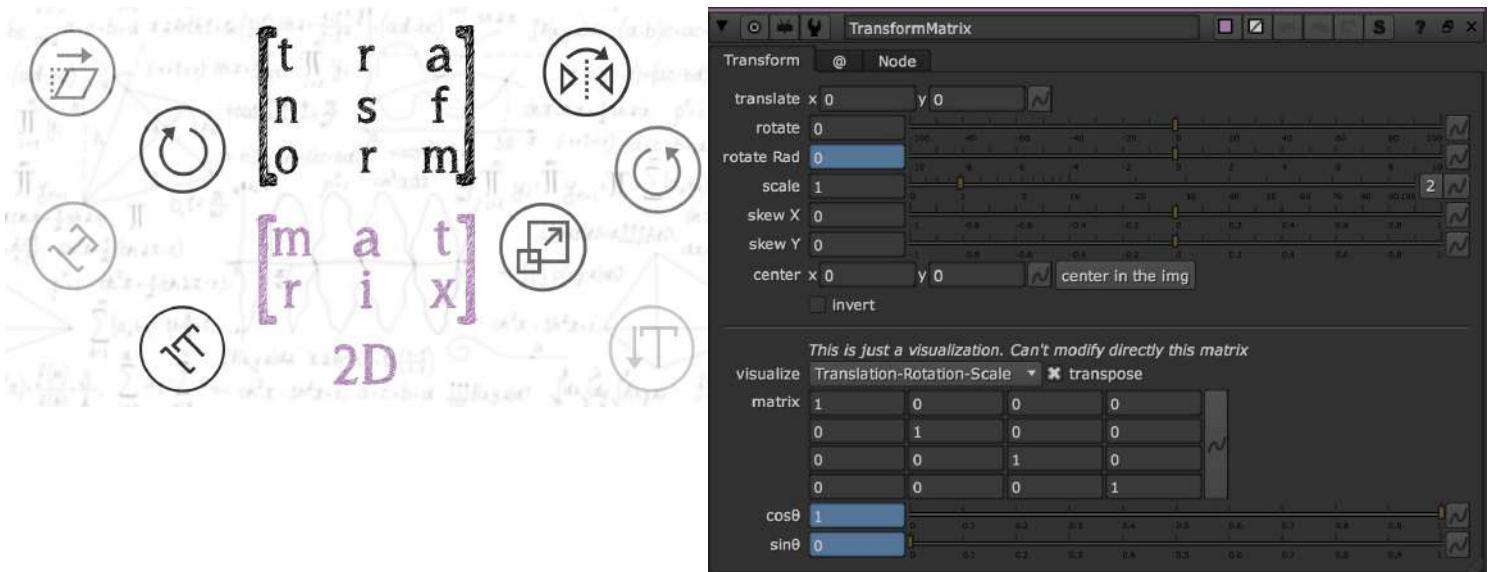
Adjustable crop - breaks concatenation but allows for adjustable bbox

No Crop - no crop applied at all, concatenation preserved but bbox can get quite big

Set to input: This will distort the image with the corner pin to fit the format, it is the same as projecting the image on a card and rendering in UV space

TransformMatrix AG

Author: Andrea Geremia - www.andreageremia.it/tutorial.html



Nukpedia link: <http://www.nukpedia.com/gizmos/transform/transform-matrix>

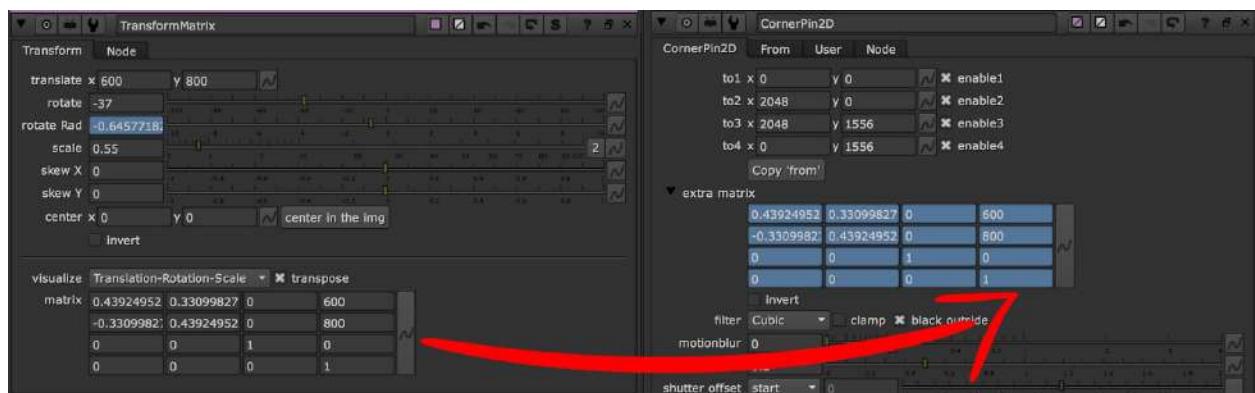
Andrea's Transform Matrix tutorial: http://www.andreageremia.it/tutorial_matrix_transform.html

Classic Transform node with a Matrix 4x4

Modified the Transform node of Nuke, adding a Matrix 4x4.

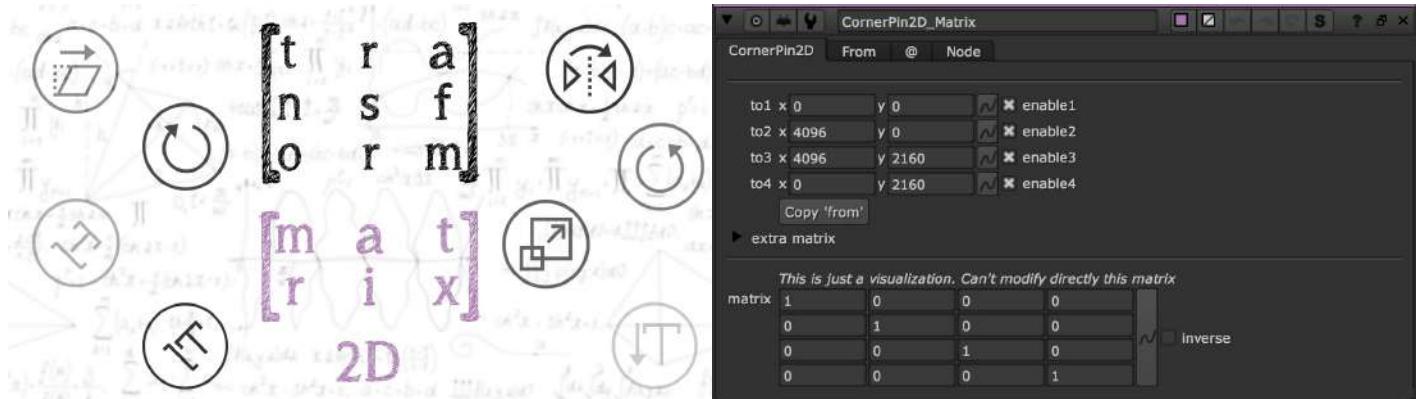
Now you can visualize the 2D Transformation with a Matrix.

This Matrix is good because you can understand how it works, but at the same time copy/paste the matrix to the extra_matrix knob in the CornerPin (for example).



CornerPin2D_Matrix AG

Author: Andrea Geremia - www.andreageremia.it/tutorial.html

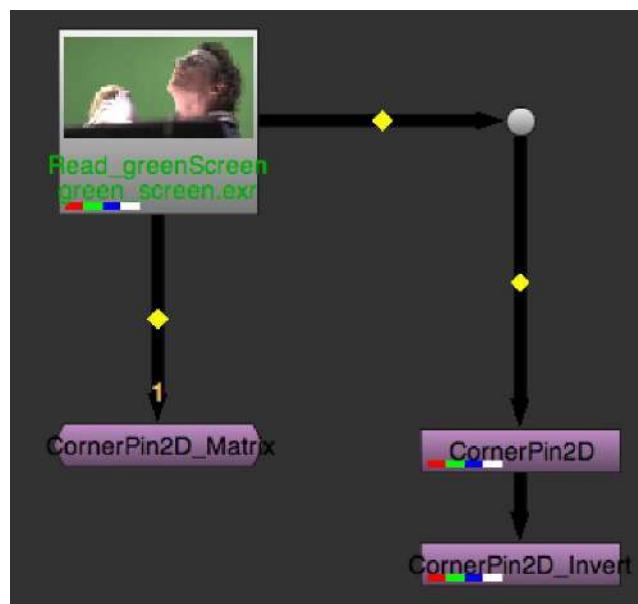


Nukepedia link: <http://www.nukepedia.com/gizmos/transform/cornerpin-matrix>

Andrea's Transform Matrix tutorial: http://www.andreageremia.it/tutorial_matrix_transform.html

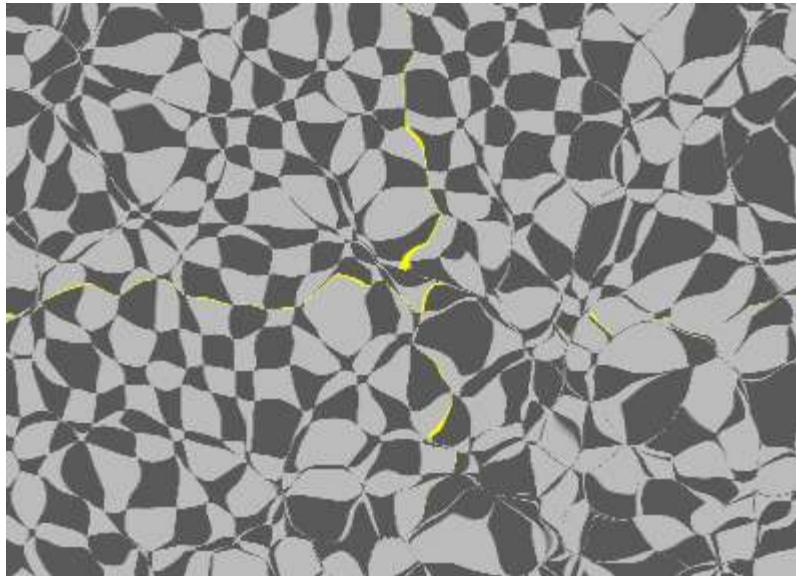
Get the classic CornerPin node in a Matrix 4x4

There is a checkbox useful to invert the Matrix. In this case we can copy the matrix in another CornerPin and the copy the inverse matrix in another CornerPin. The final result will be the original picture



IIDistort EL

Author: Erwan Leroy <http://www.erwanleroy.com>



Nukepedia link: <http://www.nukepedia.com/blink/transform/iidistort>

Recursive IDistort, produces similar results as Substance Designer's Vector Warp/Morph.

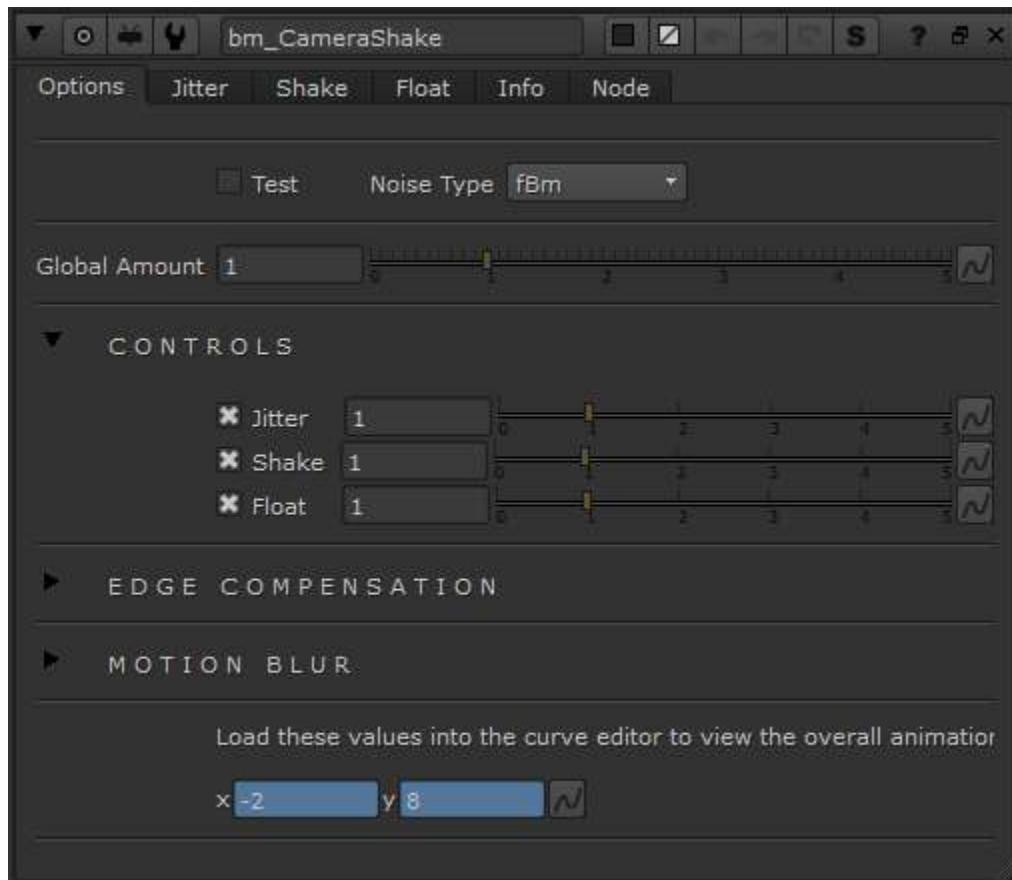
Built with Blinkscript, this is a very easy kernel to breakdown for beginners. The code is almost identical to Mads Hagbarth Damsbo's example code in his Foundry talk
<https://www.youtube.com/watch?v=p3Lv7ThKbUk>

I find the distortions created from it sometime smore interesting than from a regular IDistort.
the tool will be included with the Nuke Vector Matrix toolset available here:

<https://github.com/mapoga/nuke-vector-matrix>

CameraShake BM

Author: Ben McEwan - <https://benmcewan.com/blog/>



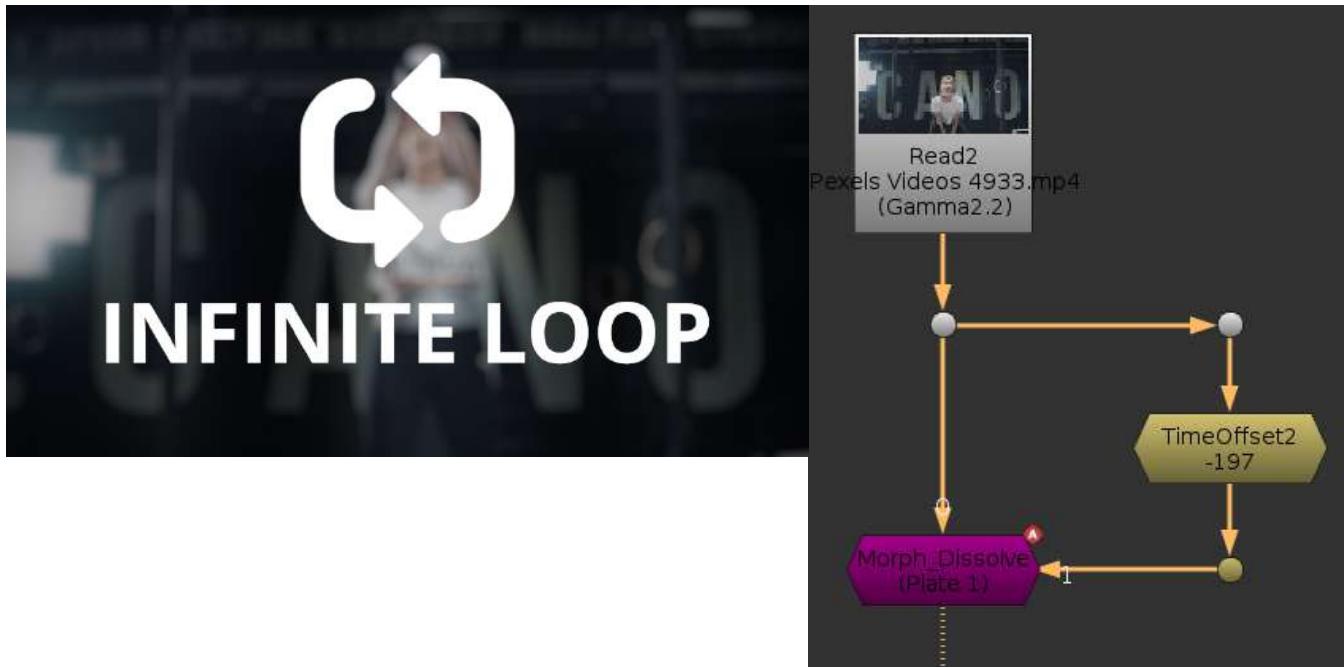
Nukepedia link: http://www.nukepedia.com/gizmos/transform/bm_camerashake/

Ben's Website: <http://benmcewan.com/nukeTools.html>

A replacement for Nuke's default camera shake node -- offers more control over 3 different frequencies of camera shake, and also shakes the centre-point, giving more detail to sub-frame motionblur. Also has options for how to deal with edge-of-frame pixels, so pushing in isn't always your best option anymore!

MorphDissolve SPIN

Author: SPIN FX and Erwan Leroy - <http://erwanleroy.com/blog/>



Nukepedia Link: http://www.nukepedia.com/gizmos/transform/morph_dissolve

Github Link:

https://github.com/SpinVFX/spin_nuke_gizmos/blob/master/gizmos/spin_tools/Comp/Morph_Dissolve.gizmo

Erwan's Write up: http://erwanleroy.com/morph_dissolve-gizmo-for-nuke/

Allows to morph between two moving plates automatically, or can be used to improve manual Morphs. Inspired by Avid Fluid Morph or Adobe's Morph Cut.

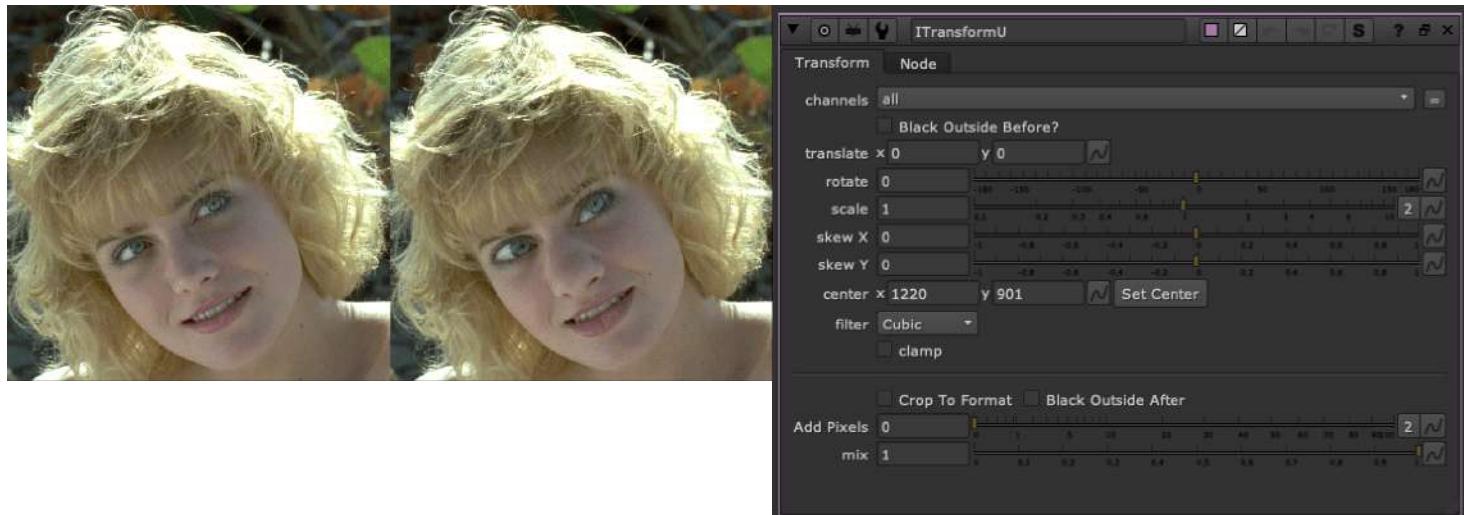
Inspired by Avid Fluid Morph or Adobe's Morph Cut.

Will work best on visually similar plates or for invisible jump cuts. The more different the two plate to morph, the more artefacts will be present.

Can be used to improve manual Morphs (splineWarp or Gridwarp) by feeding the distorted A in one input and the distorted B in the other input. The Morph_Dissolve will look for the small details you may have missed or ignored with your manual morph.

ITransform FR

Author: Frank Rueter



Updated version of Frank Rueter's Itranform tool on nukpedia, link here:

<http://www.nukpedia.com/gizmos/transform/ittransform/>

mask based warper with transform controls

Updates include:

Channels: defaults to all channels but you can select channel to warp

Set Center Button: Click to set to the center of the root.format or the input.format

Black Outside Before/After: Click to apply a black outside before and/or after the warp, this can eliminate unwanted stretching edge pixels because of bounding box issues.

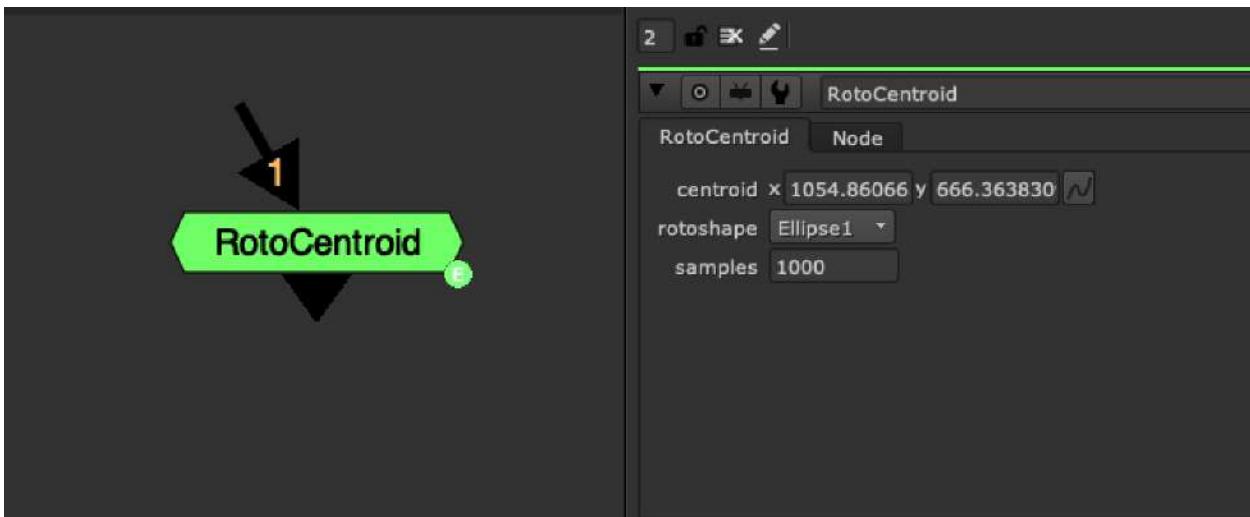
Crop To Format and Add Pixels: More options for BBox management

Mix: Using a transformMasked node instead of a transform, so the node is able to mix the warp effect

Otherwise the node reacts the same way as the original ITransform node.

RotoCentroid NKPD

Author: Alister Chowdhury - <http://alisterchowdhury.co.uk/?page=vfx>



Nukepedia Link: <http://www.nukepedia.com/gizmos/transform/roto-centroid>

Demo Video: https://www.youtube.com/watch?v=McdI_qb3ycE&feature=youtu.be

It tries to find out the centroid of a given rotoshape (based upon the area, NOT bounding box or average point values).

It does this using an algorithm I put together in python for finding the centroid of 2d shapes based upon their point locations.

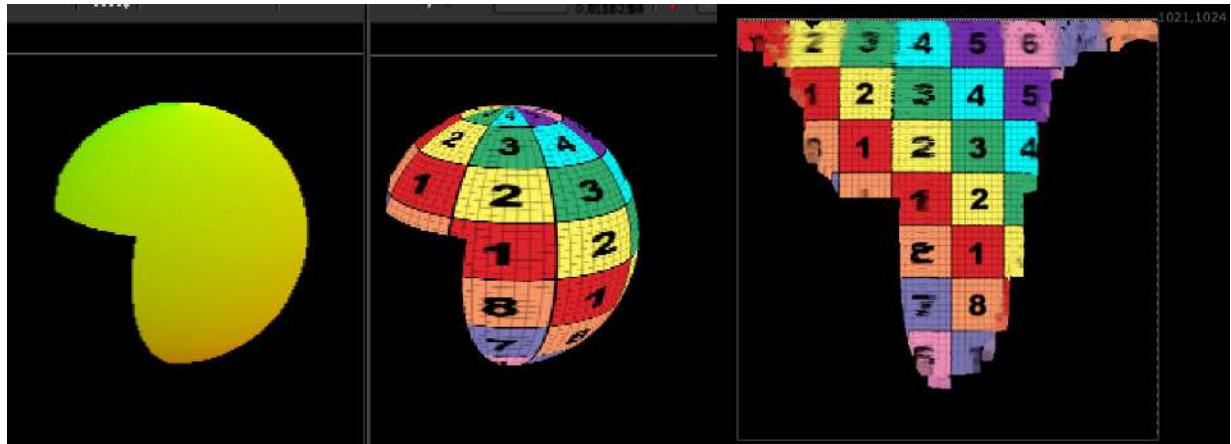
This allows you to effectively find the center of mass of a Beziers etc.

This differs from the center of the bounding box, or average point values in that it changes depending on how the shape is distributed, so it is unbiased on the amount of points you're using.

I made this because Houdini can do this, so I felt Nuke should be able to as well.

STMapInverse NKPD

Author: Luca Mignardi - www.lookinvfx.com



Nukepedia Link: http://www.nukepedia.com/gizmos/transform/in_inversestmap

A simple and fast solution to Unwarp to UV space your CG render

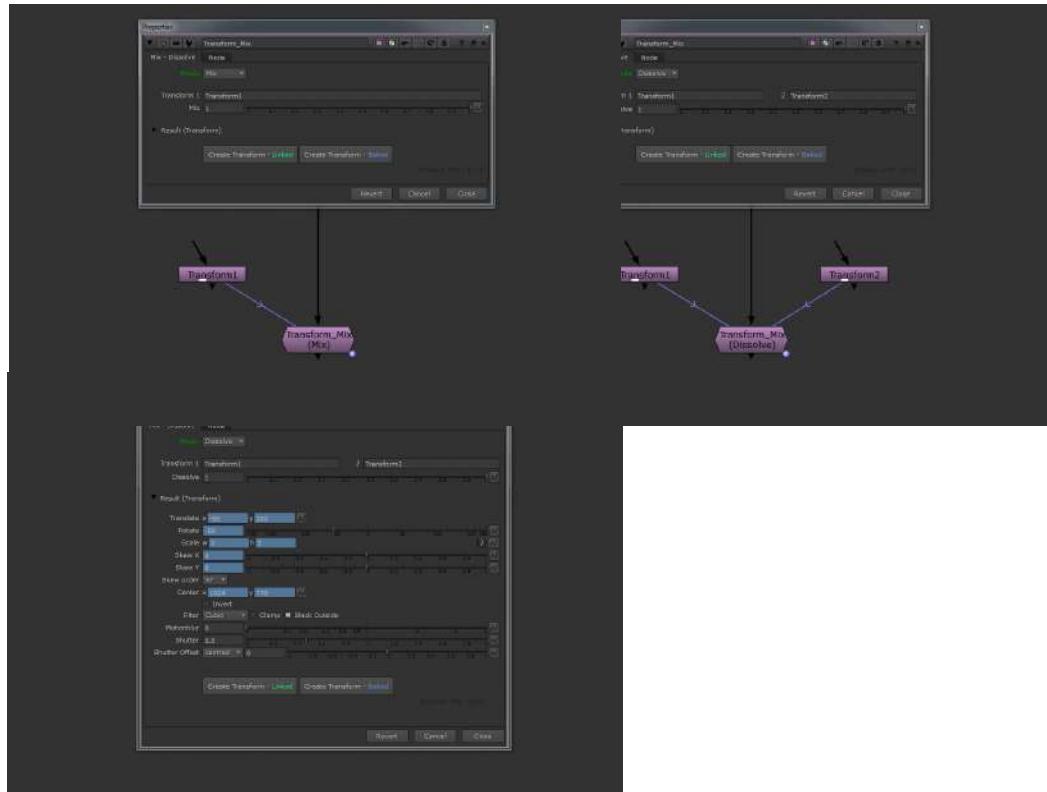
The Gizmo use a simple PositionToPoint node to redistribute each pixel in his UV space position.

Works as the nuke default STMap but does exactly the opposite. Can be used in many different ways.

Very useful especially for reference to know where to draw a roto in uv space to be then distorted via STMap to mask a particular area of your CG render.

Transform_Mix NKPD

Author: Franklin Toussaint



Tool Source Page: http://franklinvfx.com/transform_mix-3/

This tool has two functions:

Mix mode – To decrease the resulting value of an other Transform node.

Dissolve mode – To mix two Transform nodes together whis a slider.

Mode: To choose between “Mix” mode or “Dissolve” mode

Transform 1: To create a link (enter the name of the Transform reference node)

Transform 2: If on “Dissolve” mode enter the name of the second Transform reference node

Mix: 0 = all the knobs go to the default value of a native Transform node

1 = the value of all the knobs becomes the value of the linked node (Transform 1)

Dissolve: 0 = the value of all the knobs becomes the value of the linked node (Transform 1)

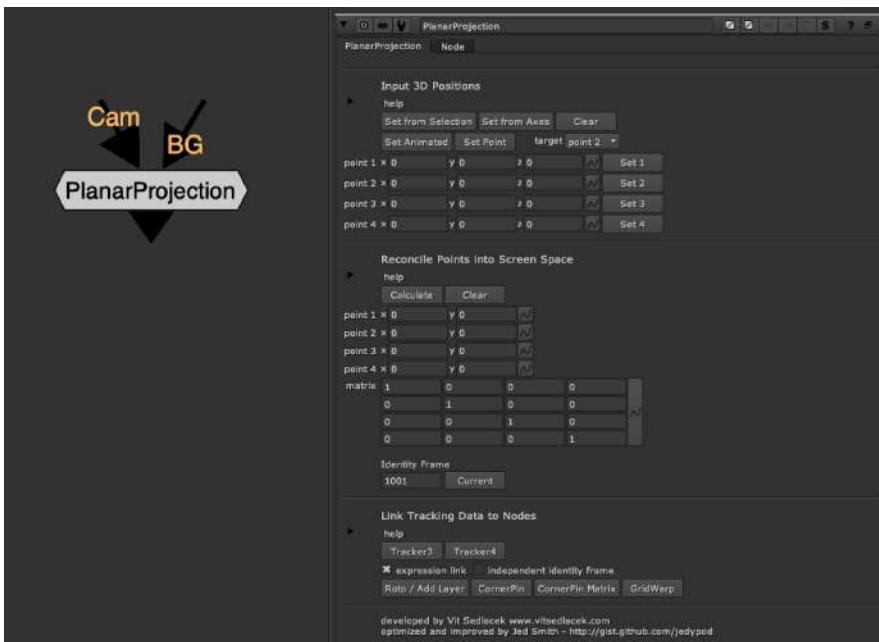
1 = the value of all the knobs becomes the value of the linked node (Transform 2)

Create Transform: To create a new Transform node with the current transform value (linked or baked)

PlanarProjection NKPD

Author: Vit Sedlacek www.vitsedlacek.com

optimized and improved by Jed Smith - <http://gist.github.com/jedypod>



Nukepedia link: <http://www.nukepedia.com/gizmos/3d/vplanarprojection>

Updated tool Github link: <https://gist.github.com/jedypod/98dc18acd8008e7e5cbe>

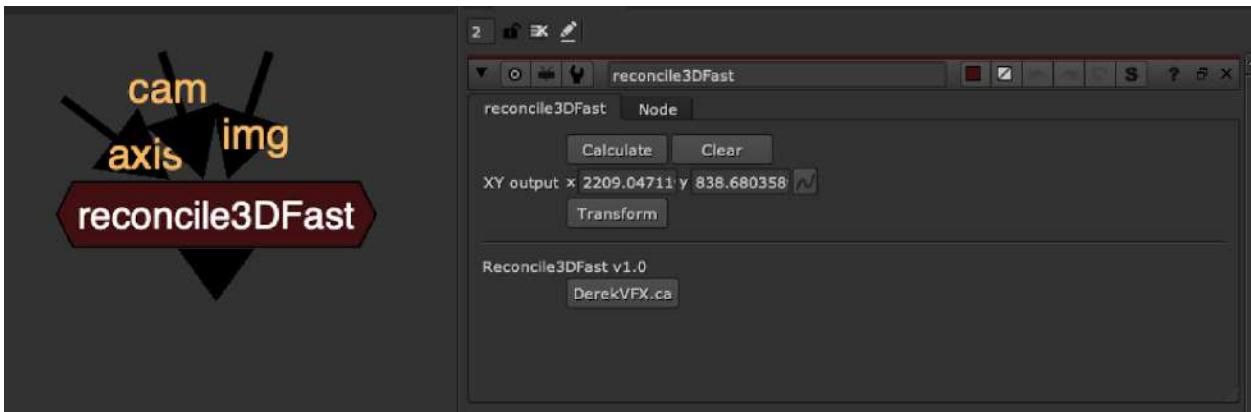
Smart, better and faster Reconcile3D node. PlanarProjection Planar Projection Generates 2D coordinates for points in 3D space. Works on 4 points at once, is instantaneous to calculate, and generates a 4x4 transform matrix for use in rotos. This gizmo was initially developed as a tool for fixing drift in complex matchmove shots in post, rather than spending time to solve unsolvable shots, but during the time it served as a perfect tool for rotoscoping or cornerpinning (instead of using 3d cards).

Workflow:

- 1) connect the camera
- 2) select the points in 3d viewport (you can use vertex selection mode, or move handles manually)
- 3) generate 2D projection points
- 4) generate matrix
- 5) use inbuild functions to generate Rotopaint/SplineWarp or GridWarp node, which is tracked with generated points

Reconcile3DFast DR

Author: Derek Rein - <https://derekvfx.ca/nuke/>



Download from github:

<https://raw.githubusercontent.com/DerekRein/DerekVFX/master/ToolSets/reconcile3DFast.nk>

This gizmo is taking some of the python script from PlanarProjection NKPD node by:

Vit Sedlacek www.vitsedlacek.com optimized and improved by Jed Smith -

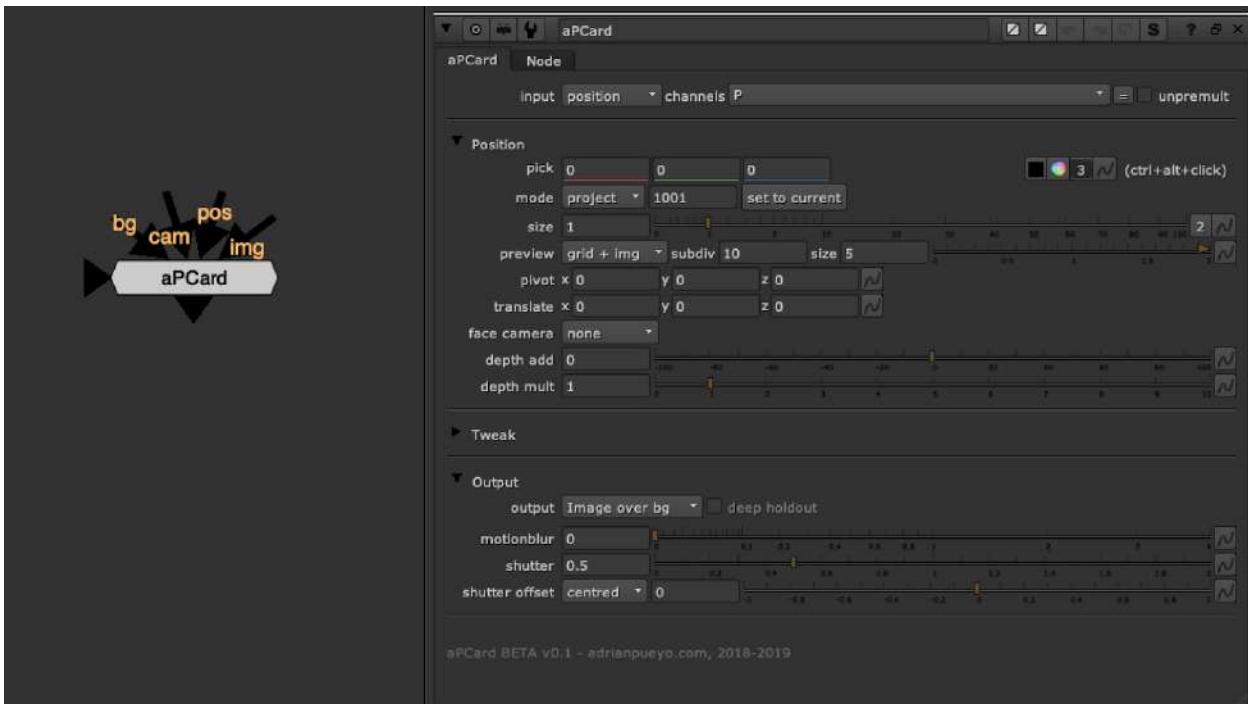
<http://gist.github.com/jedypod>

The planar projection tool is amazing and robust, but this one is simplified for just 1 axis, 2d point result and it is easy, fast, and simple.

10.) 3D

aPCard AP

Author: Adrian Pueyo - www.adrianpueyo.com



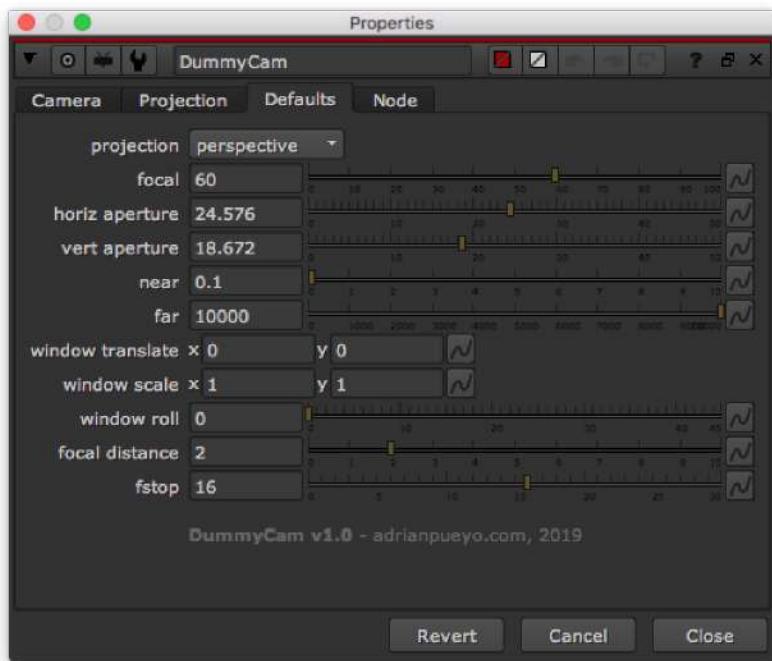
aPCard helps you quickly place a card using CG render passes. Use either: Position Pass, Depth pass, Deep data, or geometry. Ctrl+alt click to sample image data in order to place the 3d card. From there you can set a reference frame, face the card to the camera (on ref frame or permanently) and you can choose the mode of either projection or just place the card (so the texture will be in card UV space).

When you are previewing the card placement, it provides a handy grid for quick placement and size/orientation checking.

You can also use additional deep data to do a deep holdout of the card, in case it needs to go behind/between some 3d objects

DummyCam AP

Author: Adrian Pueyo - www.adrianpueyo.com



Nukepedia link: <http://www.nukepedia.com/gizmos/3d/dummymcam>

DummyCam is a Camera that apart from the matrices also grabs all the 'Projection' values from the upstream camera that it's connected to. Has a 'Defaults' tab with the knob values to use if not there's no Camera connected. It can also be used inside of groups, as many levels deep as you want. All using live and super fast TCL.

DummyCam is mainly meant for making gizmos that require a Camera input, as there is currently no other way to look for the Projection values upstream, and it is usually solved with ugly callbacks, with a Python button, or looking for the topnode, which only works as long as you don't plug an axis to the top camera, etc. Simply put it inside the Group, plugged to the cam Input node, and voilà.

Special thanks to Ernest Dios, we decyphered the undocumented parts of the code together.

mScatterGeo MJT

Author: Mark Joey Tang - blog: www.facebook.com/MJTLab



Nukepedia link: <http://www.nukepedia.com/gizmos/3d/mscattergeo>

Download Mark Joey Tang's entire Toolset: <http://bit.ly/menupy>

mScatterGeo supports scattering on geo's vertices, faces and density map in nuke. 2 rotation types available, 'look At' and 'normals'. Variation of size, rotation, texture and instance geo can be set.

v1.0 Demo :

https://youtu.be/_v-6UR_Mkj4

v1.1 Extract Axis Demo :

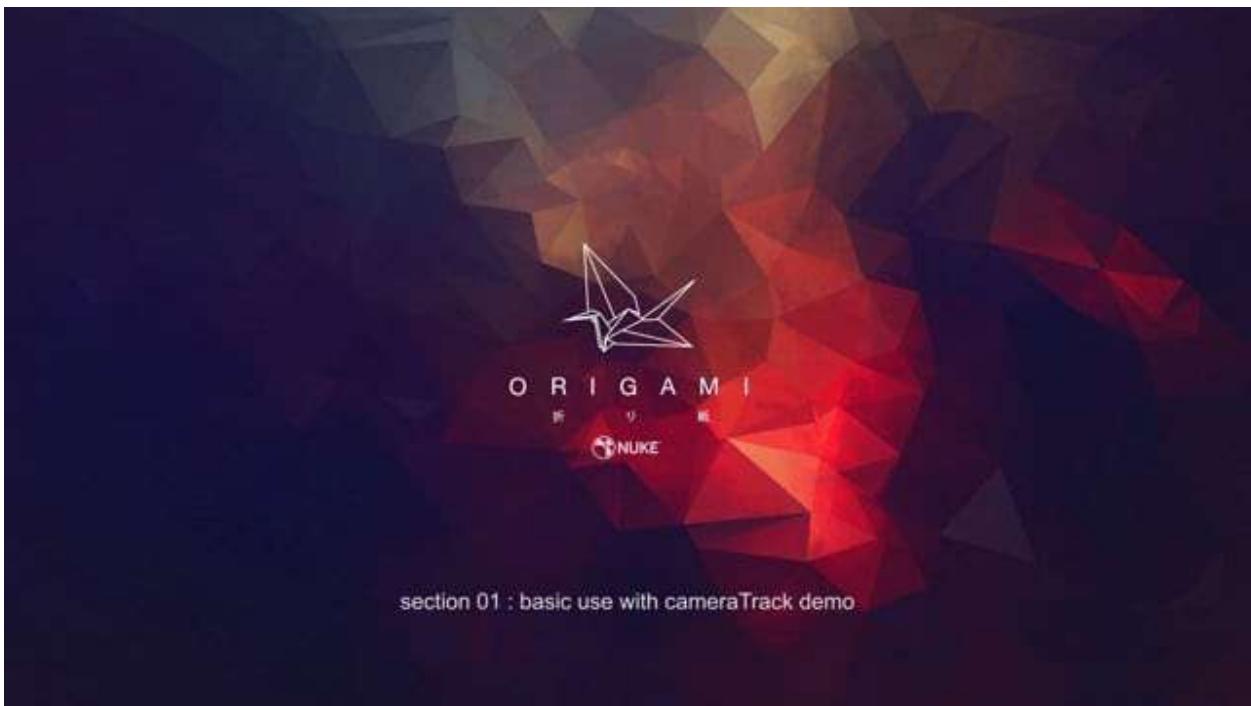
<https://youtu.be/CpcWB1oXdtk>

v1.2 Density Map Demo :

<https://youtu.be/EWijN05bsS8>

Origami MJT

Author: Mark Joey Tang - blog: www.facebook.com/MJTLab



Nukepedia link: <http://www.nukepedia.com/gizmos/3d/origami>

Download Mark Joey Tang's entire Toolset: <http://bit.ly/menupy>

Demo video (new version) :

<https://www.facebook.com/MJTLab/videos/527711171052558/>

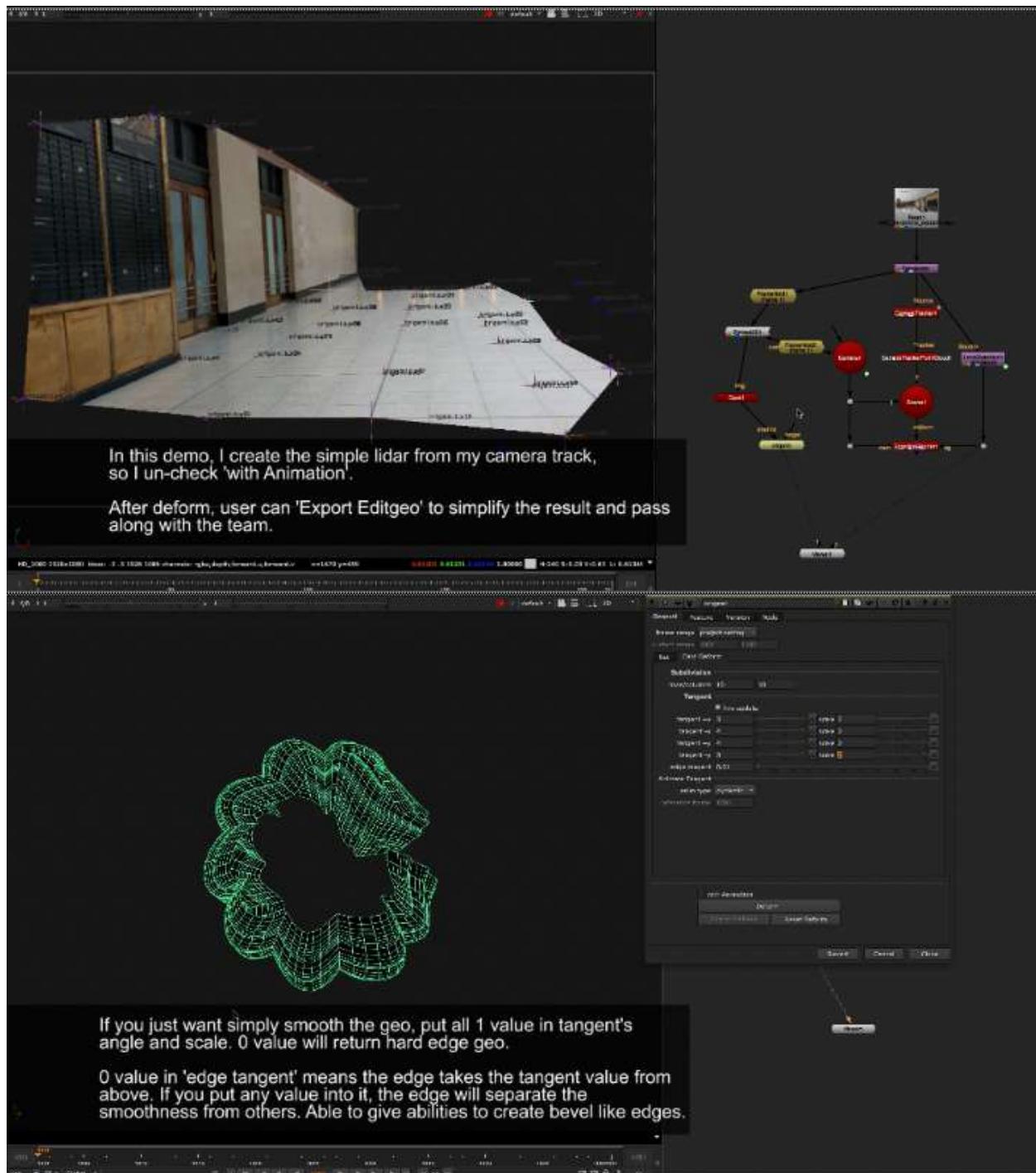
<https://vimeo.com/318138533>

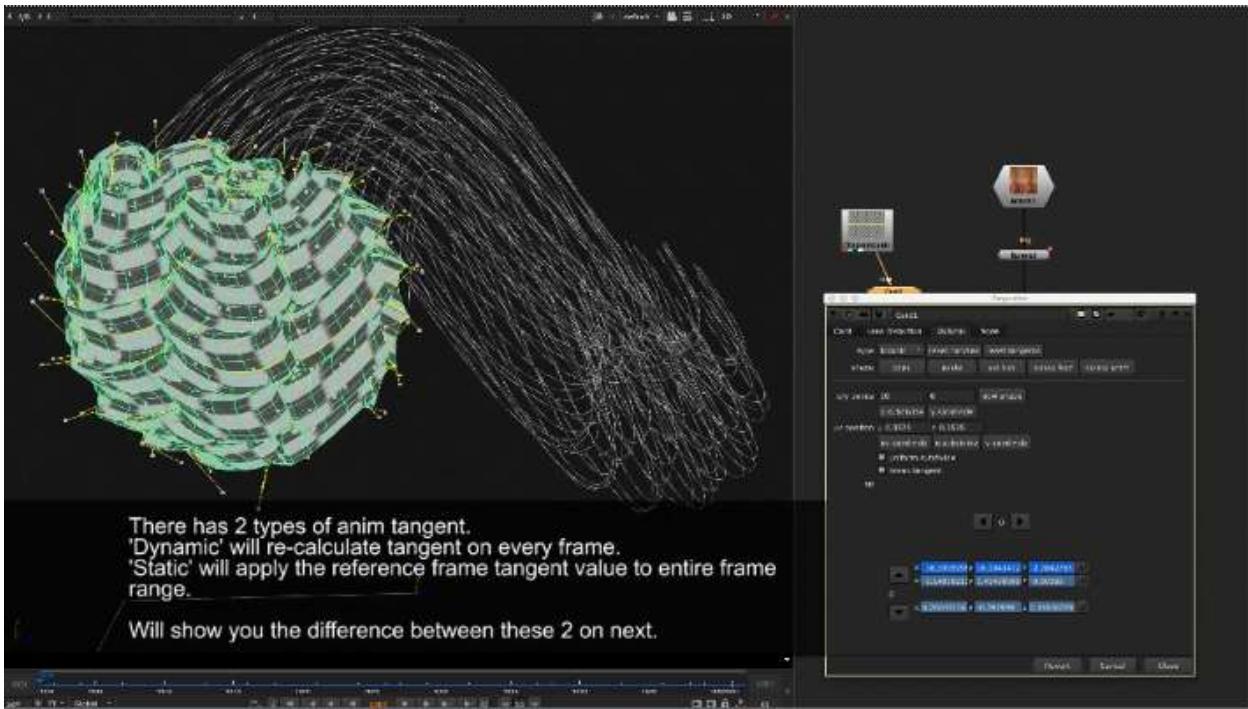
Initial intention of 'Origami' is a *just for fun* tool. Since the tool involved many setup. Then I start to build it for practical usage. It helps to build geo patch for scatter objects or form a new UV, re-build a messy wireframe photoscan geo, clone a high-end geo to low-end geo for patch or 3D reference, and also able to create interact animation. v1.2 added tangent for geo smoothness.

Here is one of the example case of using origami :

create fake new UV for texturing in nuke

https://www.facebook.com/pg/MJTIlab/photos/?tab=album&album_id=415866105661883



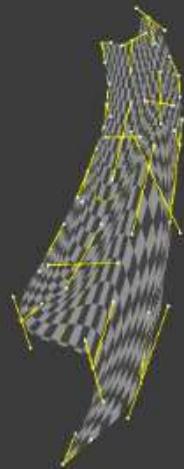


Anim type of tangent comparison

Each type of tangent anim can apply in different cases. Depends on the situation.

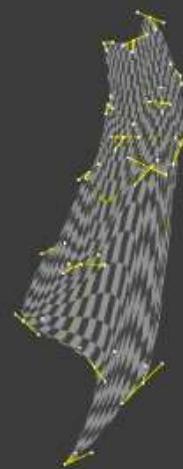
Dynamic

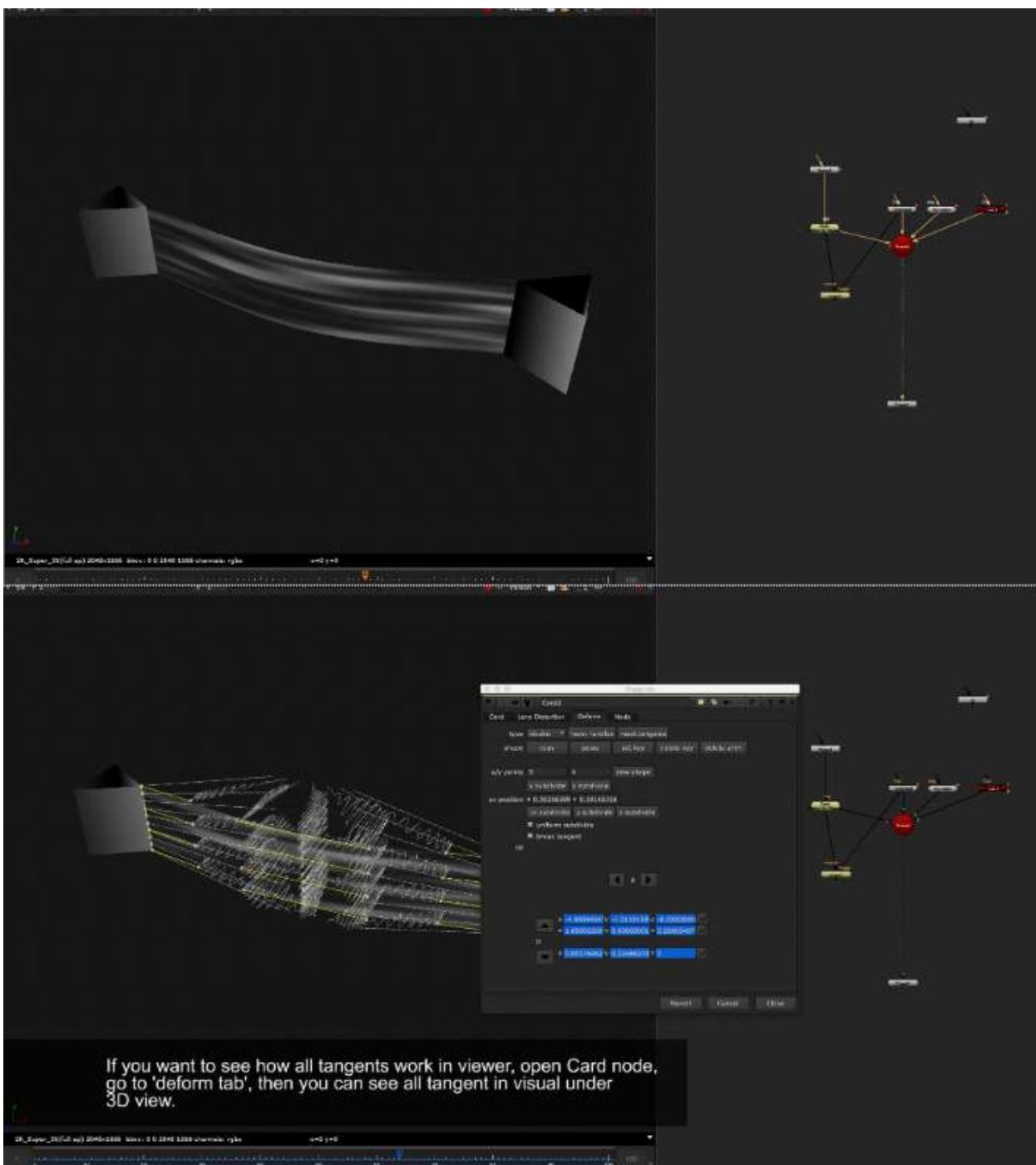
Calculate tangent value
on every frame according
position change on each
points.

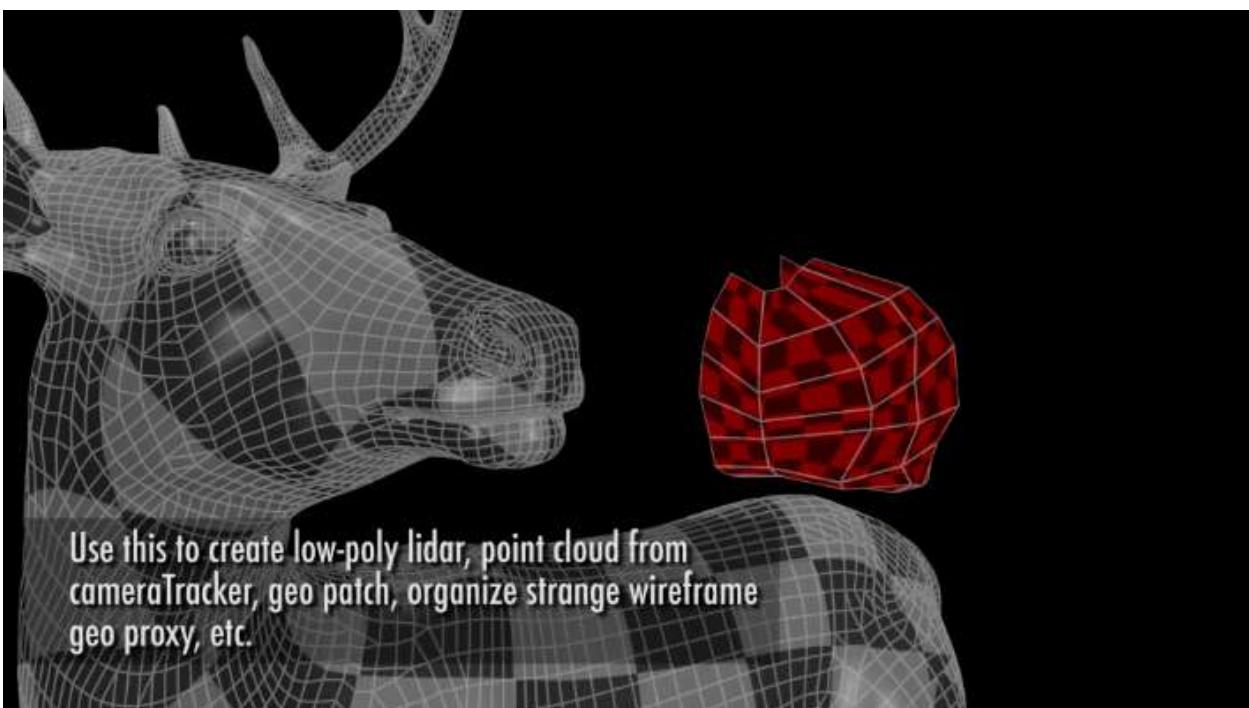


Static

Apply the tangent value
from reference frame to
entire frame range.







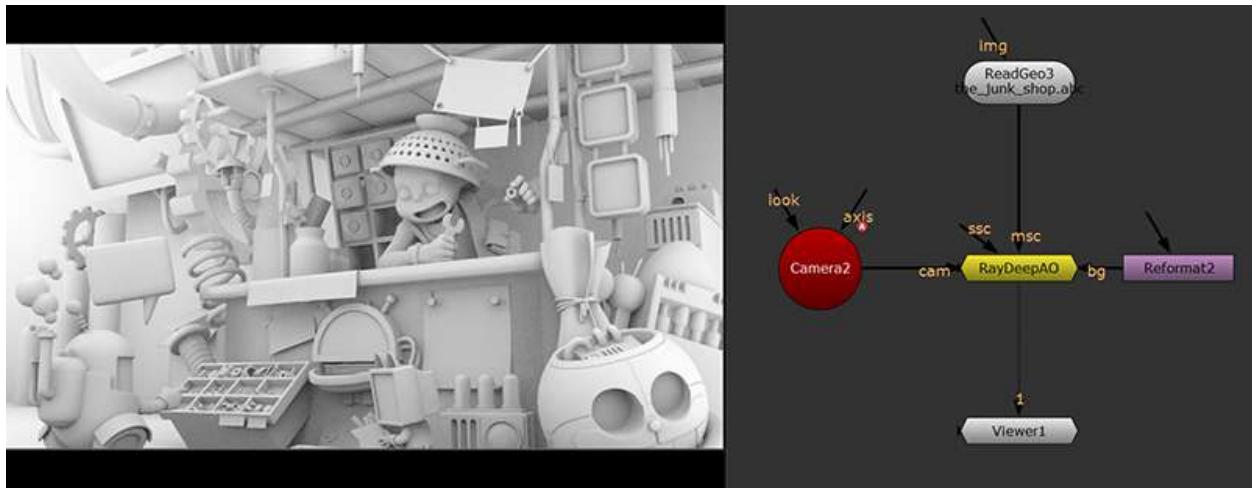
Use this to create low-poly lidar, point cloud from
cameraTracker, geo patch, organize strange wireframe
geo proxy, etc.

Saliva demo
50% speed playback



RayDeepAO MJT

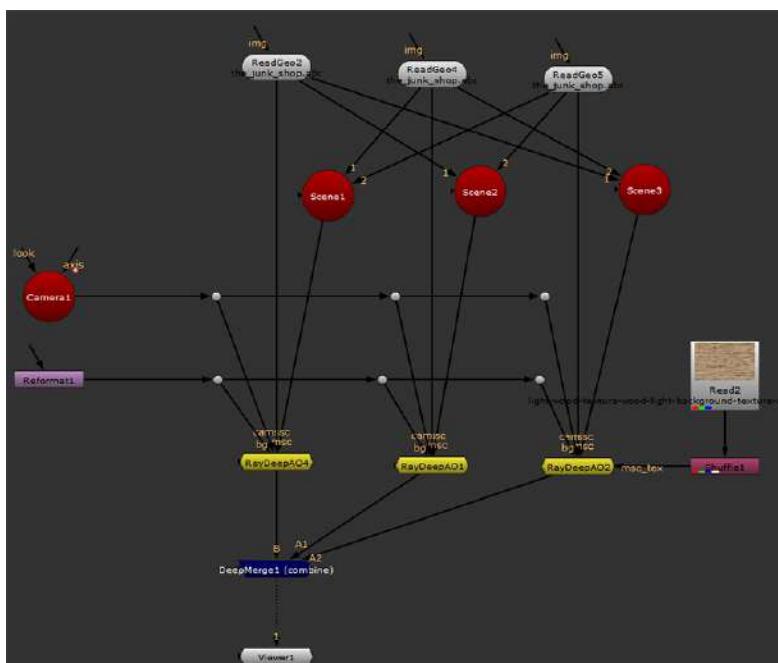
Author: Mark Joey Tang - blog: www.facebook.com/MJTLab

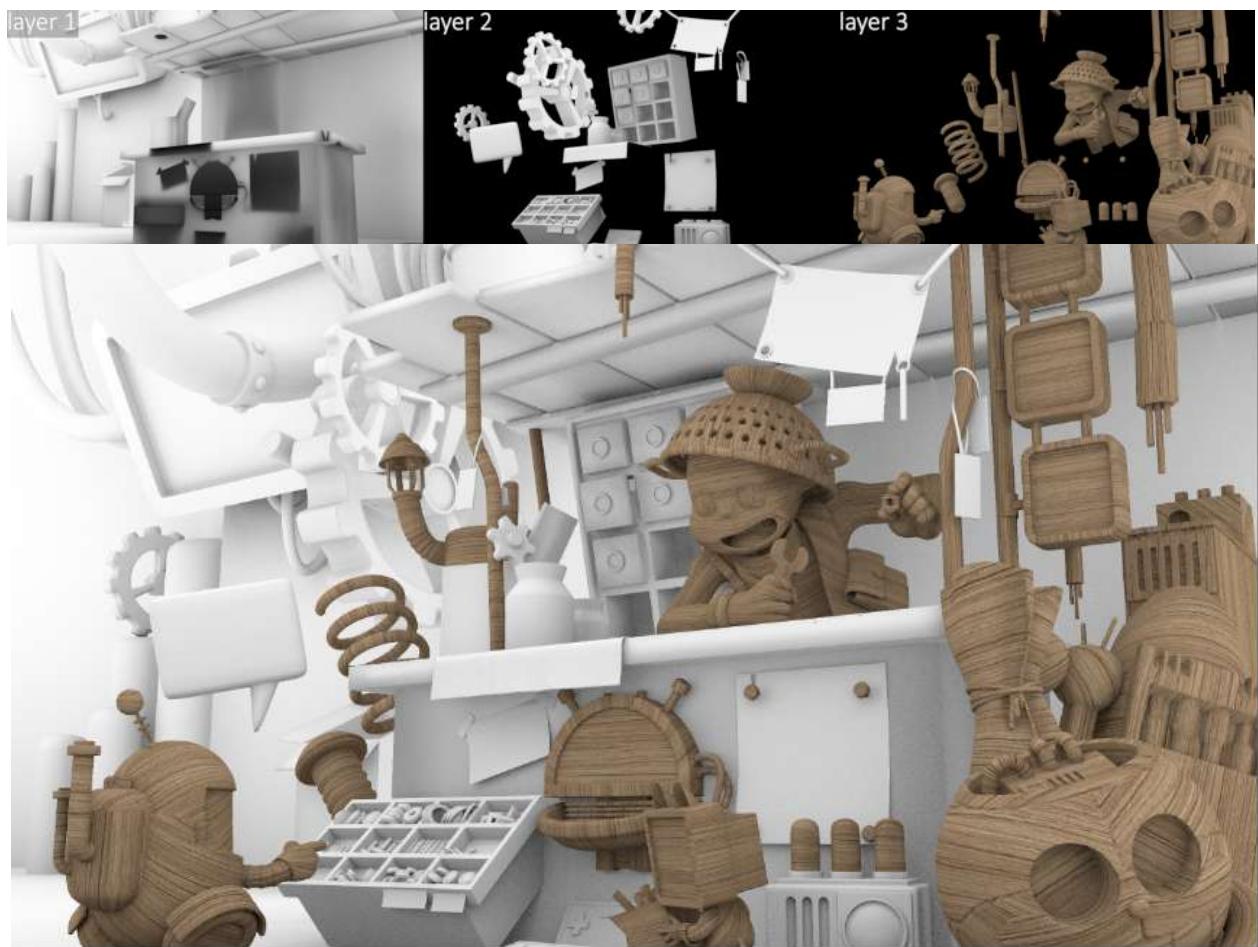


Nukepedia link: <http://www.nukepedia.com/gizmos/3d/raydeepao>

Download Mark Joey Tang's entire Toolset: <http://bit.ly/menupy>

A setup to render Ambient Occlusion from Geo to deep format, able to set non-renderable objects.





Create all in layers, so the main scene can interact with other geos but not holdout by them.

inputs :

cam: connect with camera node

bg: define the output resolution

msc: stand for 'main scene', which is the primary geo(s) to render AO. Input can be a single geo or a scene node connected to multiple geos.

ssc: stand for 'sub scene', which is setup non-renderable geo(s) but they interact with the primary geo(s). Input can be a single geo or a scene node connected to multiple geos.

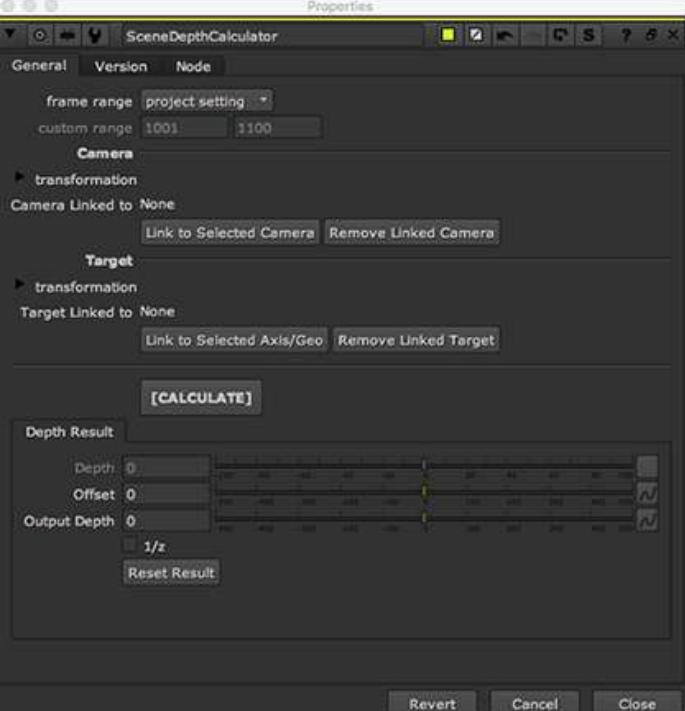
msc_tex: stand for 'main scene texture'. This input will be used when 'use texture' checked. texture required in UV space.

output :

Output is in deep format. If output vector is checked, position and normal data is also in deep format.

SceneDepthCalculator Mjt

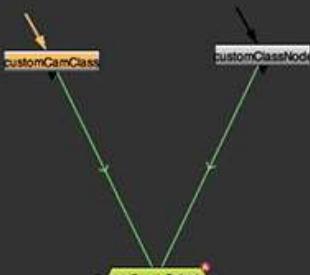
Author: Mark Joey Tang - blog: www.facebook.com/MJTLab



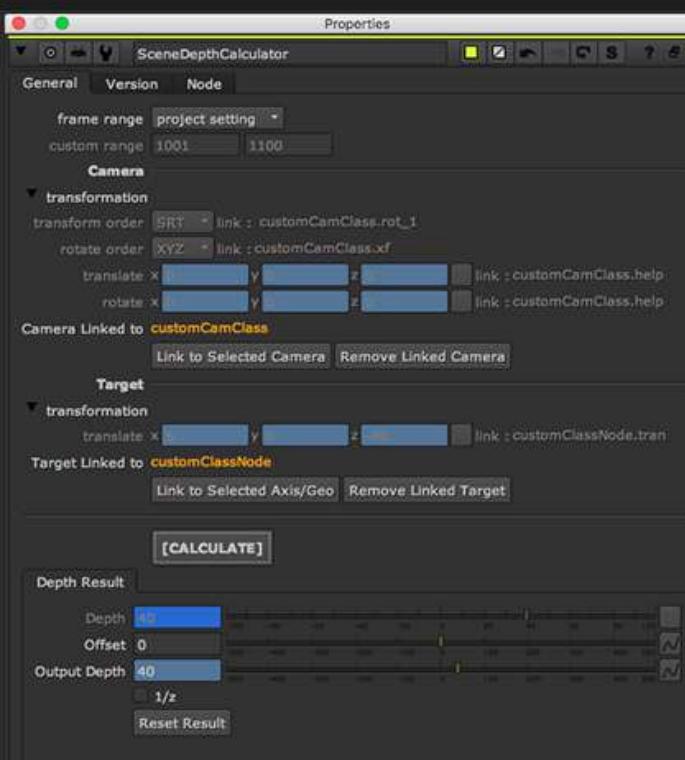
SceneDepthCalculator 1.7

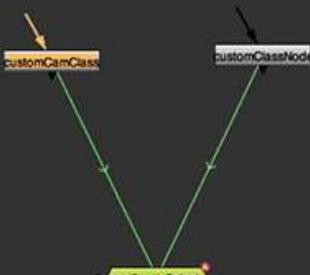
1.7

- removed node class restriction
 - * Because some companies are using their own camera/axis class nodes. So it supported any type of nodes
- can use any nodes' knob value
 - * if detected missing default knob to link, user can set another knob manually or even use the custom data in 'NoOp'.
- show linked expression of camera and target
- frame range added 'current frame' option. For 1 frame calculation.



```
graph TD; customCamClass --> SceneDepthCalculator; customClassNameode --> SceneDepthCalculator;
```





```
graph TD; customCamClass --> SceneDepthCalculator; customClassNameode --> SceneDepthCalculator;
```

Nukepedia link: <http://www.nukepedia.com/gizmos/3d/scenedepthcalculator>

Download Mark Joey Tang's entire Toolset: <http://bit.ly/menupy>

Demo video :

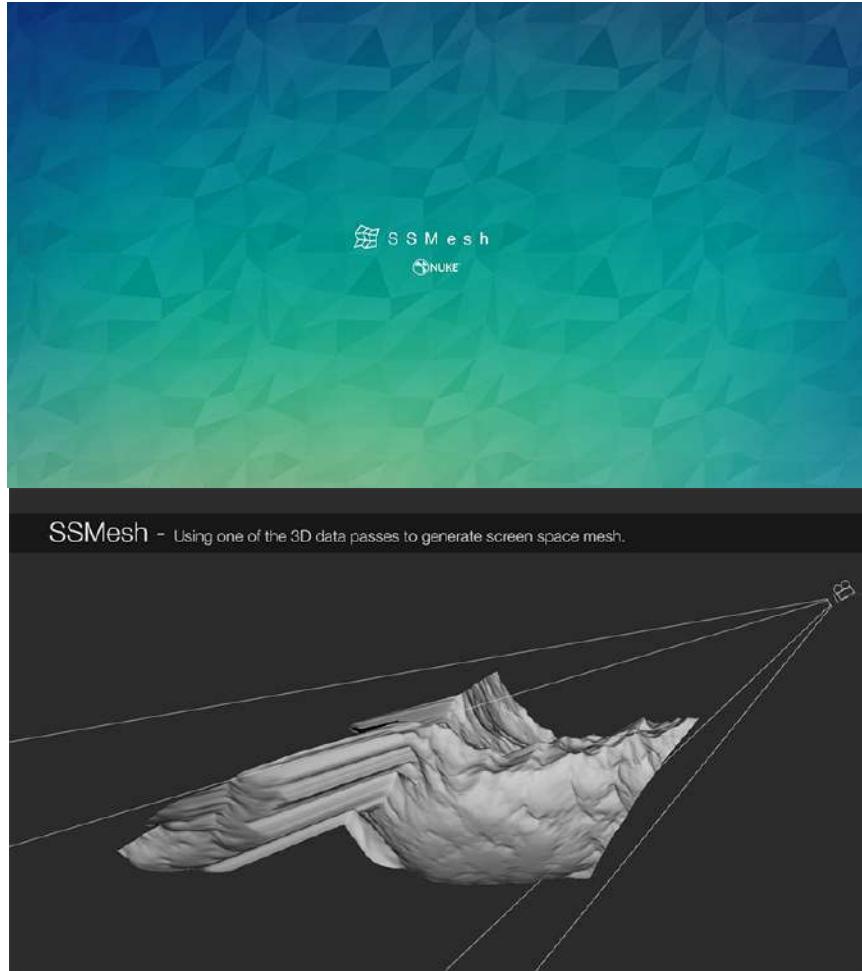
<https://vimeo.com/309033461>

<https://www.facebook.com/MJTLab/videos/2330503703848374/>

SceneDepthCalculator for calculate the depth value from camera to object. It useful for any depth tool, mostly common for rack defocus. I am also using this to drive all 2D nodes while setup template. Use lidar and camera to calculate the shot scale, and use the scale ratio to drive all 2D nodes in template base on the key shot. So it will get the consistent look from the key shot.

SSMesh MJT

Author: Mark Joey Tang - blog: www.facebook.com/MJTLab



Nukepedia link: <https://www.nukepedia.com/gizmos/3d/ssmesh>

Download Mark Joey Tang's entire Toolset: <http://bit.ly/menupy>

Using Position, depth or deep data to convert screen space mesh. Since pointcloud is pixel base 3D coordinate data, and it changes on every frame. So SSMesh helps to convert those data to vector data and process any vector tools.

Demo v1.2 (added focus region demo) :

<https://www.facebook.com/watch/?v=493031774607125>

<https://vimeo.com/356083546>

Unify3DCoordinate MJT

Author: Mark Joey Tang - blog: www.facebook.com/MJTLab



Nukepedia link: <http://www.nukepedia.com/gizmos/3d/unify3dcoordinate>

Download Mark Joey Tang's entire Toolset: <http://bit.ly/menupy>

Unify all the 3D transformation nodes into one single coordinate value. Snap vertex supported animation, shifted pivot, and able to export translation and rotation.

Solve complicated 3D transformation setup and tricky snap vertex, return in one single coordinate value.

video about the latest tool can be find here :

<https://vimeo.com/308083719>

<https://www.facebook.com/MJTLab/videos/591505891309655/>

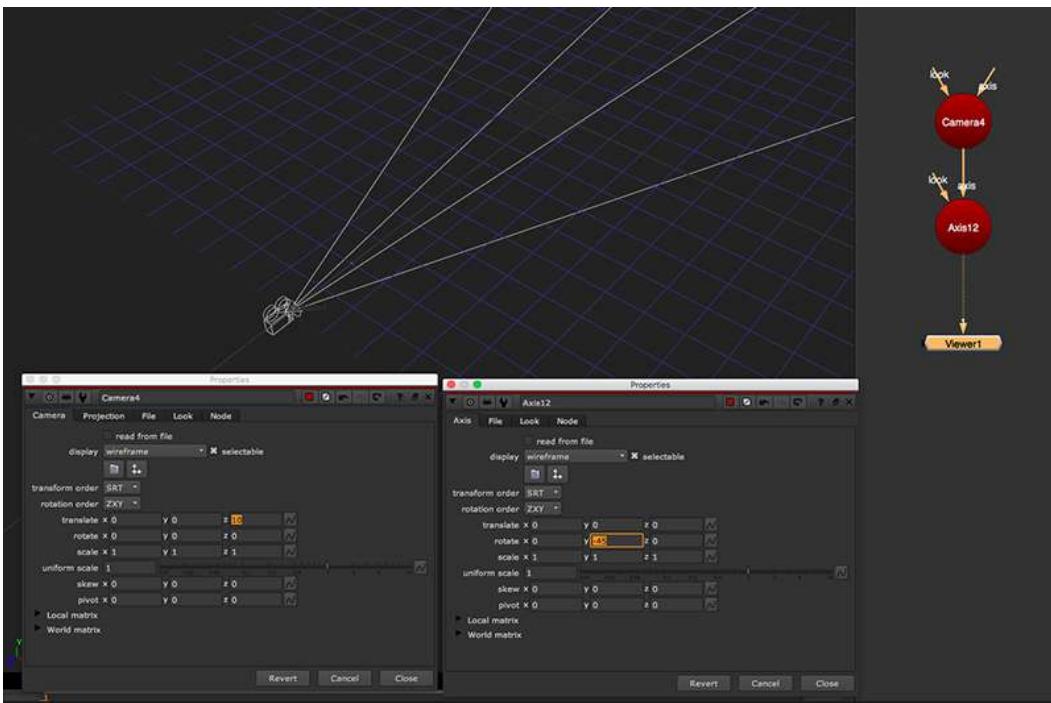
video content :

Part01 - introduction & camera demo

Part02 - geo demo

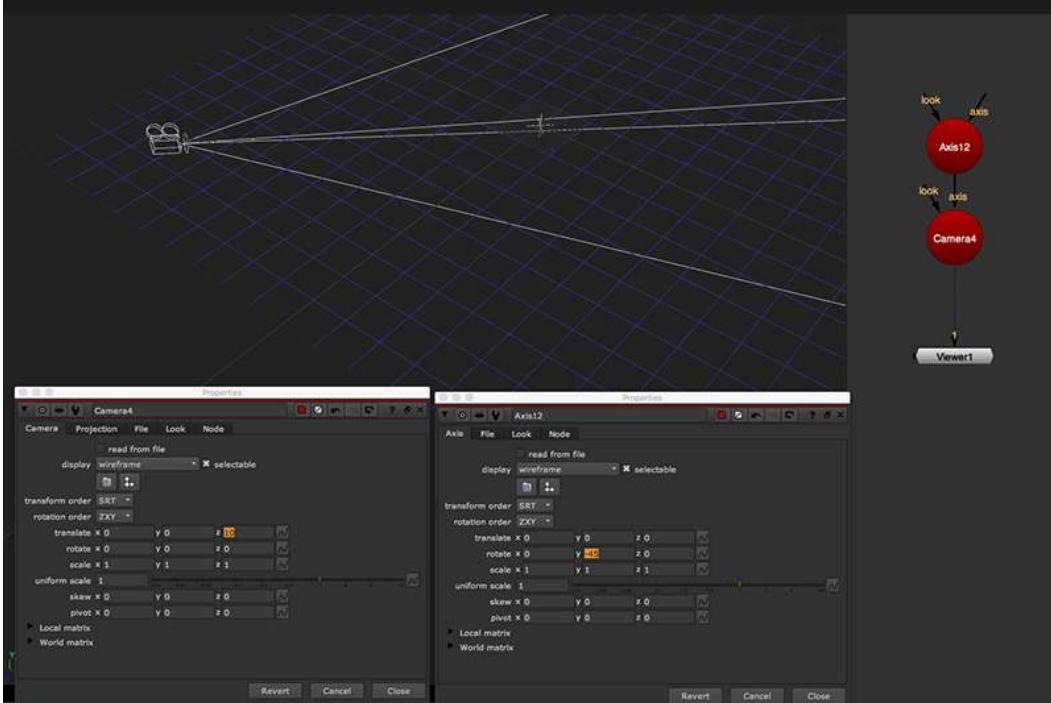
Part03 - light demo

Part04 - snap vertex on readGeo demo



What is non-reverse node in Unify3DCoordinate ?

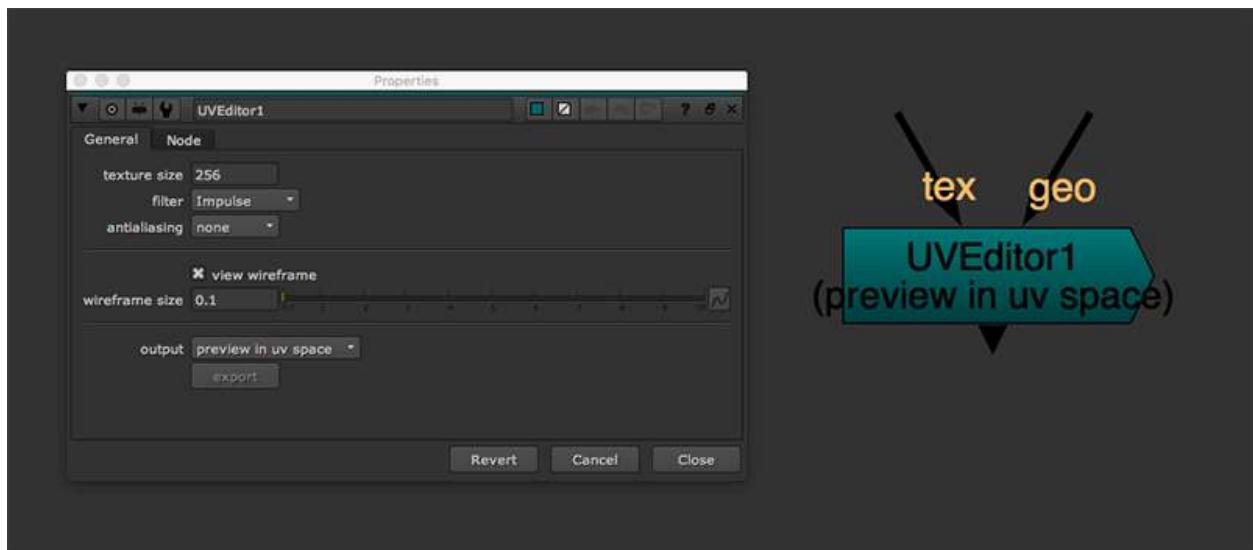
An example above, when camera connected to Axis, but the rotation in Axis doesn't apply on the camera position.



When swap the Axis to Camera's upstream, the rotation applied to the camera. Then the Camera Class (Camera2) should be on 'filterNonReverse' list for Unify3DCoordinate to process.

UVEditor MJT

Author: Mark Joey Tang - blog: www.facebook.com/MJTLab



Nukepedia link: <http://www.nukepedia.com/gizmos/3d/uveditor>

Download Mark Joey Tang's entire Toolset: <http://bit.ly/menupy>

UVEditor is simplify the steps of create uv tile or udim setup in nuke, provide the visual of uv tile in uv space of the geo, export udim texture, extract mattepainting projection in udim, paint fix the texture and map it back to the geo.

For what is uv tile / udim and how to get it works in nuke. Please check here :

https://www.facebook.com/pg/MJTLab/photos/?tab=album&album_id=348648709050290

Download zip included this tool in gizmo and group format, UVEditor_demo.nk, demo geo (please re-link the geo file).

texture size

- Define the texture resolution in here.

filter

- uv output has to go through scanlineRender, this filter is same as all the transformation pixel filter

antialiasing

- uv output has to go through scanlineRender, this antialiasing is same as antialiasing setting as scanlineRender

show tile number

- show the number of tile, helps for indicate udim frame.

view wireframe

- Enable this wireframe option will preview uv in wireframe instead of texture. Give user an idea of how the look of geo in uv space.

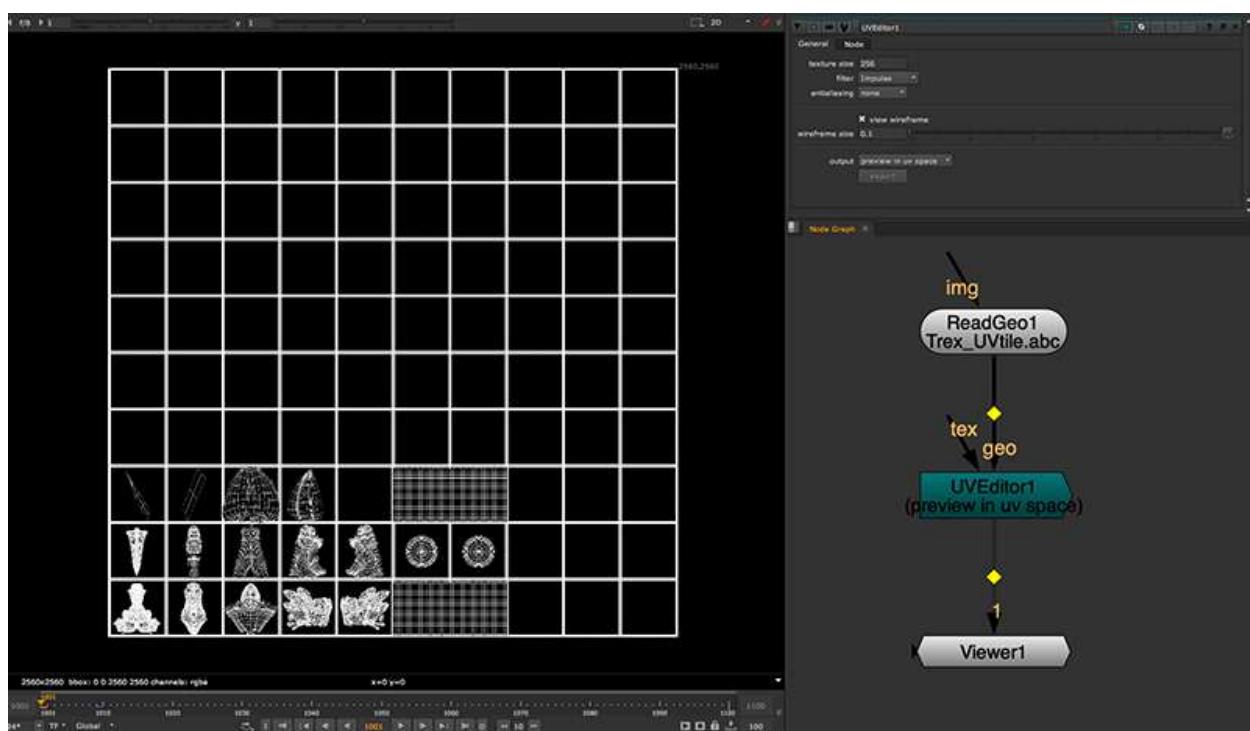
wireframe size

- The thickness of wireframe to show. Only enable when view wireframe is checked.

output

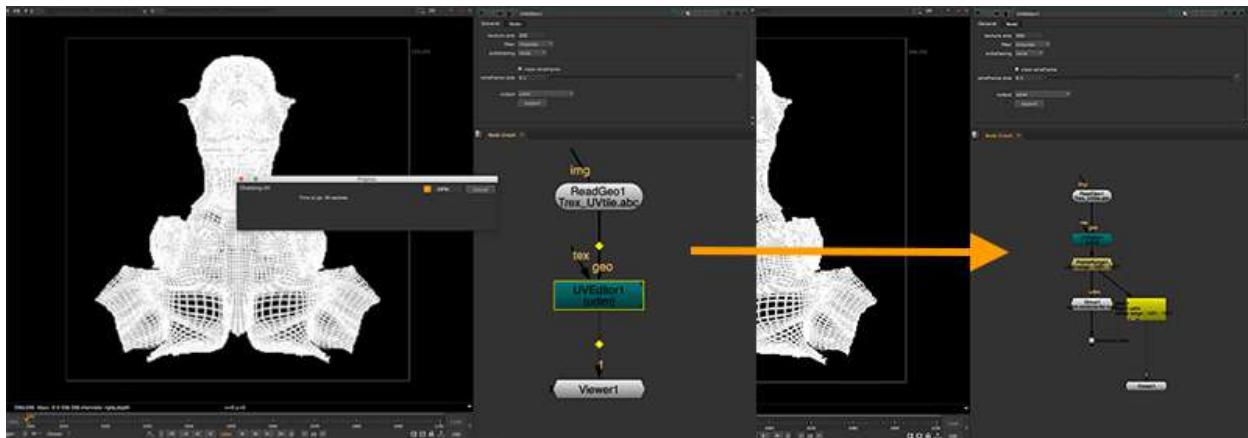
- preview in uv space

This will displace uv tile in uv space. For preview purpose.



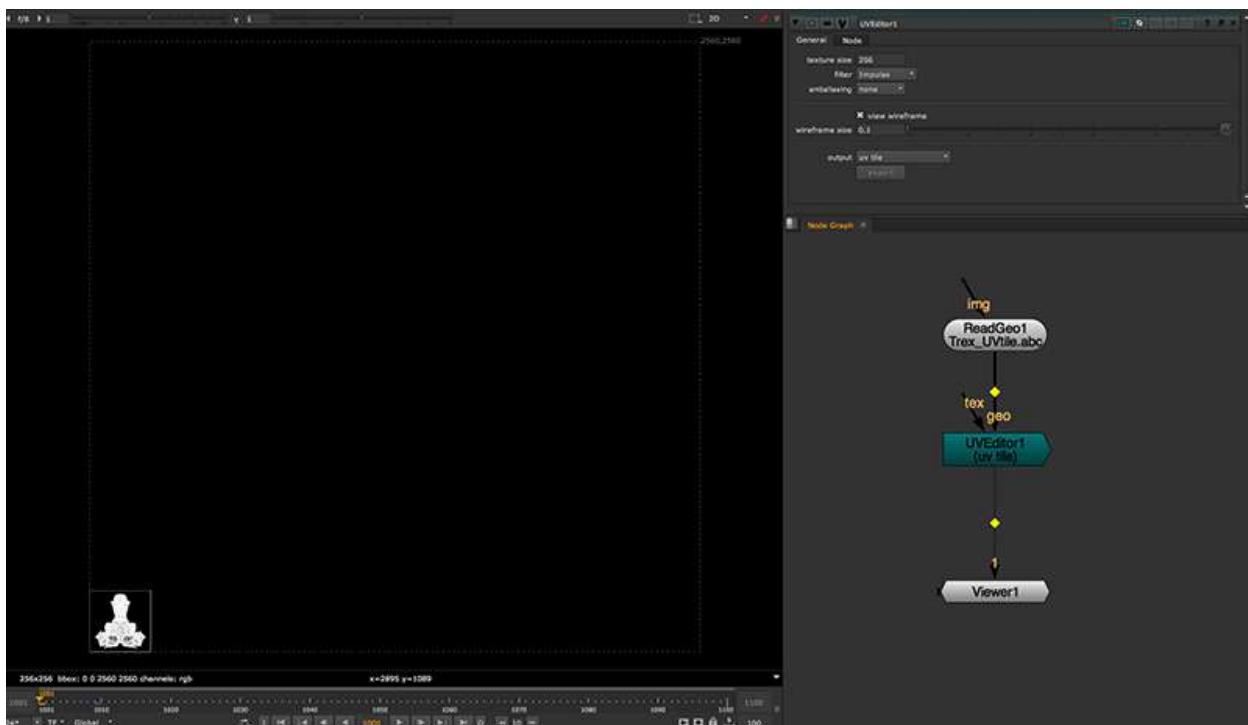
-udim

This will separate uv tile in udim format (base on frame number start from 1001). When selected udim as output, 'export' button will be able. That will scan through all available udim and return the frame range of the udim. It will generate a group with all udim combined for nuke to work with, and also a write node to show where should render if udim need to be export as texture sequence.



- uvtile

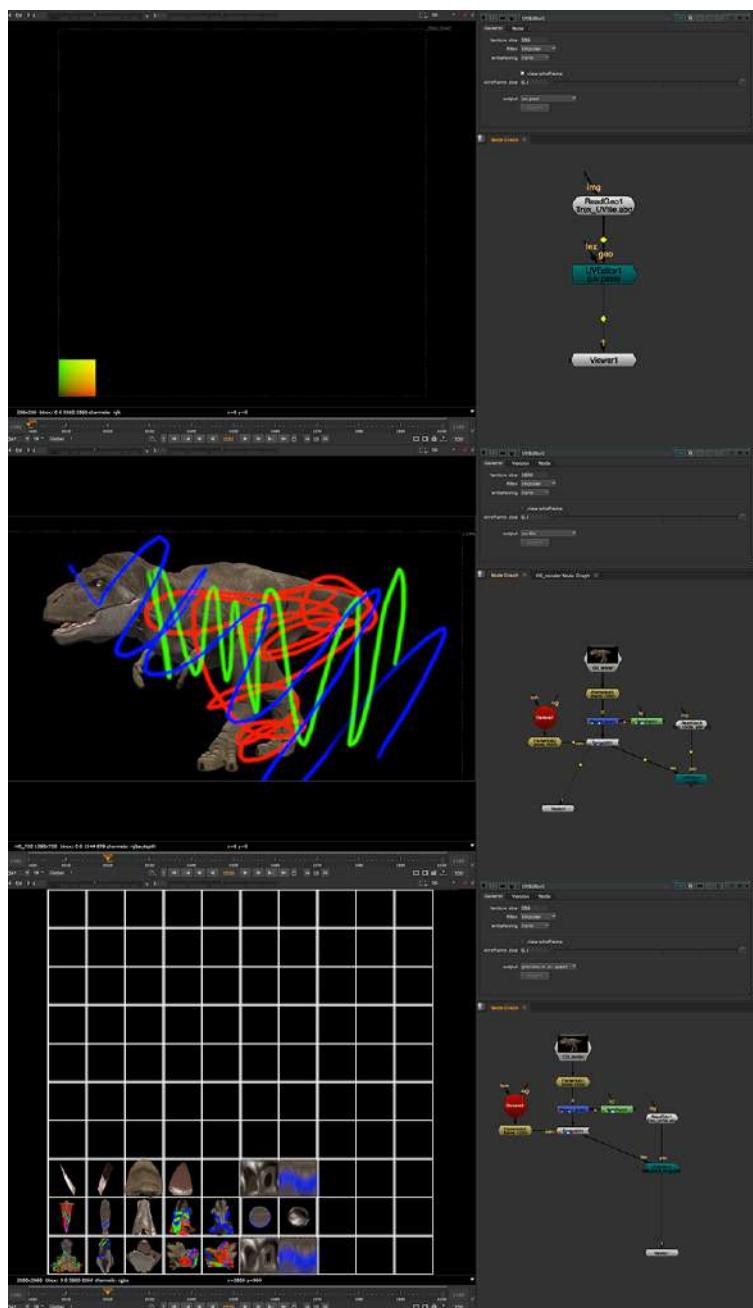
Output the uv in tile format, can map to the geo directly without any process. If user work with



uvtile for texture modification, user need to work with overscan size manually.

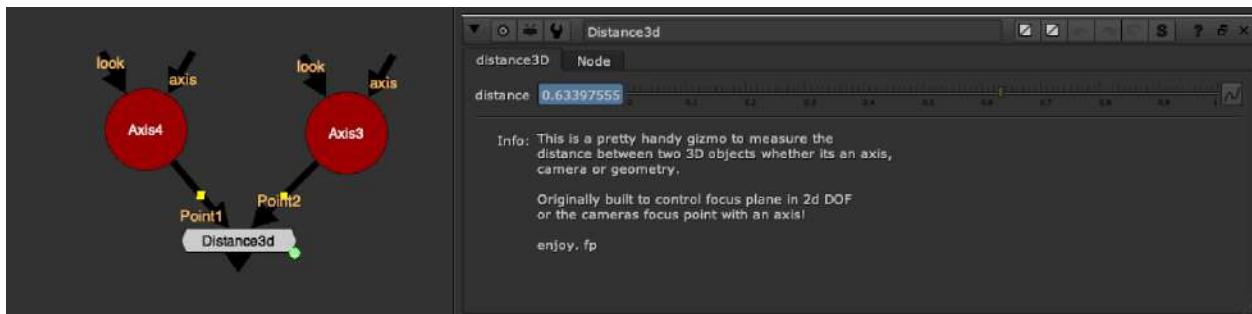
- uv pass

Output uv data as texture, same as 3D software will provide in render. Since the tool is working n 10x10 uv tile, so the uv pass will also support 10.0x10.0 uv data. Then user can use this to work with smap for texture mapping.



Distance3D NKPD

Author: Falko Paepel



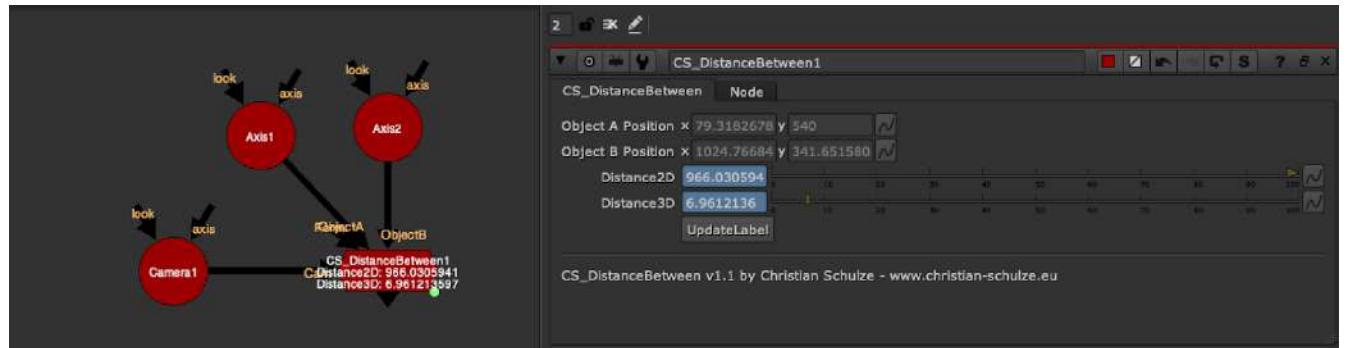
Nukepedia: http://www.nukepedia.com/gizmos/3d/distance3d_v02_fp

This is a pretty handy gizmo to measure the distance between two 3D objects whether its an axis, camera or geometry.

Originally built to control focus plane in 2d DOF or the cameras focus point with an axis! enjoy.

DistanceBetween_CS NKPD

Author: Christian Kauppert



Github link: <https://gist.github.com/kpprt/a86e5c8a3a181c89e24cc45281c6d1d3>

Christian's git gists: <https://gist.github.com/kpprt>

Calculate the distance between two 3D objects (Camera, Axis) in Nuke. 3D distance needs the two objects, 2D distance also needs a Camera input and an optional format when different from the root format.

Lightning3D EL

Author: Erwan Leroy - <http://erwanleroy.com/blog/>



Making 3D Lightning in Nuke using Blinkscript.

Needs to be used with the HigX particle renderer by Mads Hagbarth

3D lighting, similar to the X_tesla node, settings for look and animation of 3d Lighting.

Blog Website: <http://erwanleroy.com/making-3d-lightning-in-nuke-using-blinkscript/>

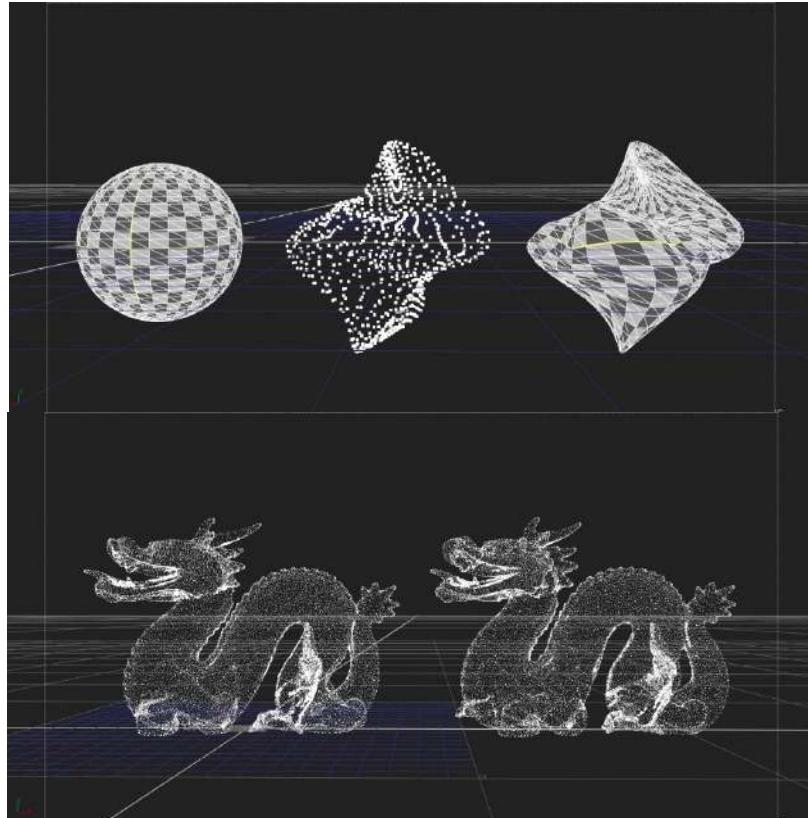
Github Link:

https://github.com/herronelou/nuke_stuff/blob/master/toolsets/blinkscript/lightning_generator_nk

Demo Video: <https://vimeo.com/387061845>

GeoToPoints MHD

Author: Mads Hagbarth Damsbo - <https://hagbarth.net/blog/>



Blog write up: <https://hagbarth.net/major-bug-in-nukes-particle-system/>

Link to tool: <http://www.hagbarth.net/nuke/GeoToPoints.nk>

Creates a point cloud based on input Geo vertices

Noise3DTexture NKPD

Author: Ben Sumner



Nukepedia link: http://www.nukepedia.com/gizmos/3d_bs_noise3d

This Gizmo outputs either the input geometry with a 3D world position noise applied as a texture, or it can output the texture itself in the UV space of the model.

Your model must have UVs!

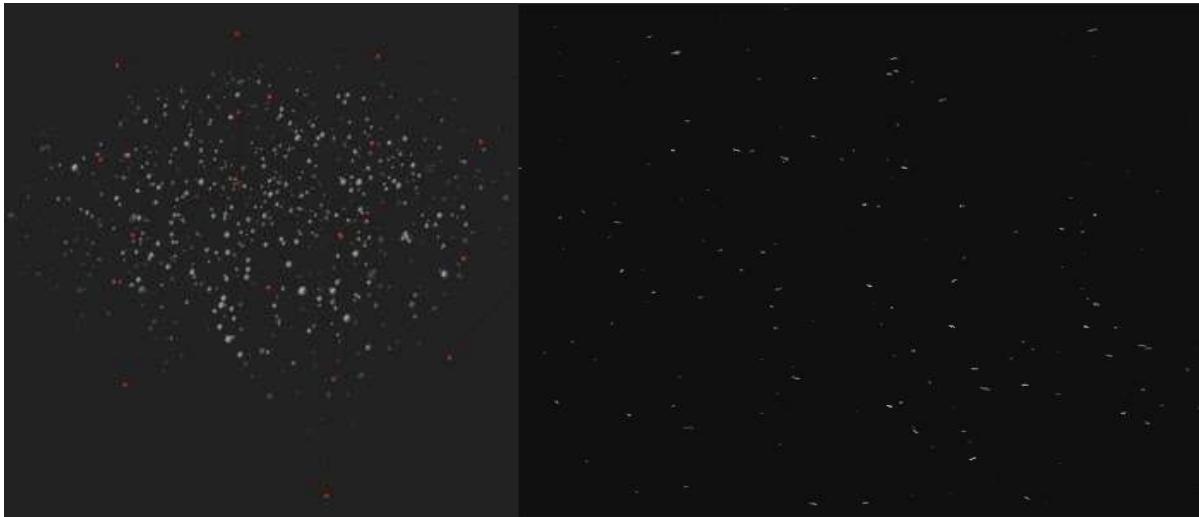
You can output noise or textured model, either with the originally applied alpha, solid white alpha, or with the generated noise in the alpha channel.

This node is a good way to get seamless noise textures on your 3D models, or to export a texture to be used for seamless displacement.

11.) Particles

WaterSchmutz DR

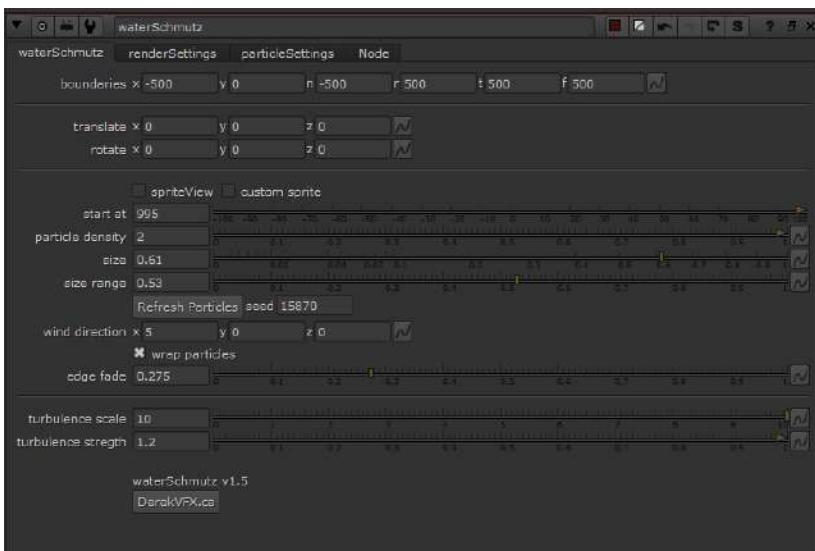
Author: Derek Rein - <http://derekvfx.ca/nuke/>



githubLink: <https://raw.githubusercontent.com/DerekRein/.nuke/master/ToolSets/schmutz.nk>

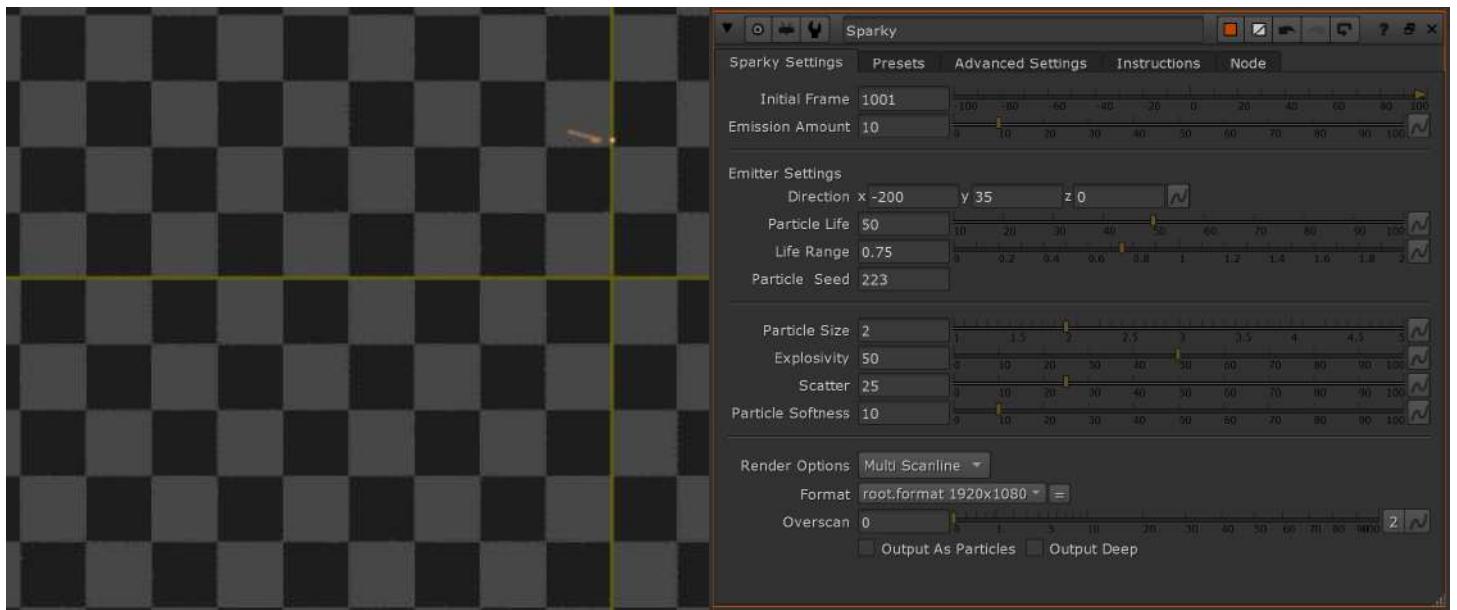
WaterSchmutz is a quick and easy particle box to create floating (or static) particles with some built-in variation settings such as size and color.

Plug in a camera and an optional customSprite input.



Sparky NKPD

Author: Dimitri Breidenbach



Nukepedia link: http://www.nukepedia.com/gizmos/particles/db_sparky

Easy to use particle setup to create sparks. Comes with a few animation presets.

Sparky is a pretty clean and simple setup to add sparks in your shot. It is a particle setup that is rendered through a ScanlineRender. The idea is that you can use it like a simple pre-rendered 2D Element coming from your favorite library, but, the main difference being that you can rotate the sparks, and give them the exact orientation you need.

This tool is delivered with an example .nk scene to see both 2D and 3D workflows.

-> Axis and Cam input : While the main goal is to be a 2D pre-rendered option, you can also select to use a 3D camera to track properly with your shot. In this case, you'll have to plug your Camera and an Axis to move the sparks in the 3D space.

--> Presets : While the look of the particles is very important, the emission rate is definitely a huge part of selling the sparks look. That is why, I decided to add 6 presets that can be loaded from the second tab. These presets will ask you to provide an initial frame and will then apply an animation to the 'Emission Amount' setting.

Warning: Presets dont work in Nuke Non-Commercial due to some Python Limitation

- Single Hit Heavy: Heavy hit with a double pop.
- Single Hit Light : Light pop of sparks.
- Welding : Expression that will mimic a natural welding feeling.
- Wavy : Constant sim with some strong variation in birth rate.
- Constant : Softer variation than the Wavy preset.
- Loopy : Loops the same pop animation of Sparks.

--> Output As Particles : This will allow you to plug the particles into a Scene node. It will therefore not render any 2D preview anymore. If Sparky doesn't render anything, verify that you didn't check this box by error.

--> Advanced Settings : If you wish to have a wider spread of sparks, or if you decide do animate the Axis to which the sparks are attached, you can access to a few settings that may interest you here.

Video example of Sparky

<https://vimeo.com/420973211>

The node is deep compatible if you keep it in MultiScanline mode (the default one), but if you want to be 200% sure you are in the right setting, you can check the 'Output Deep' option.

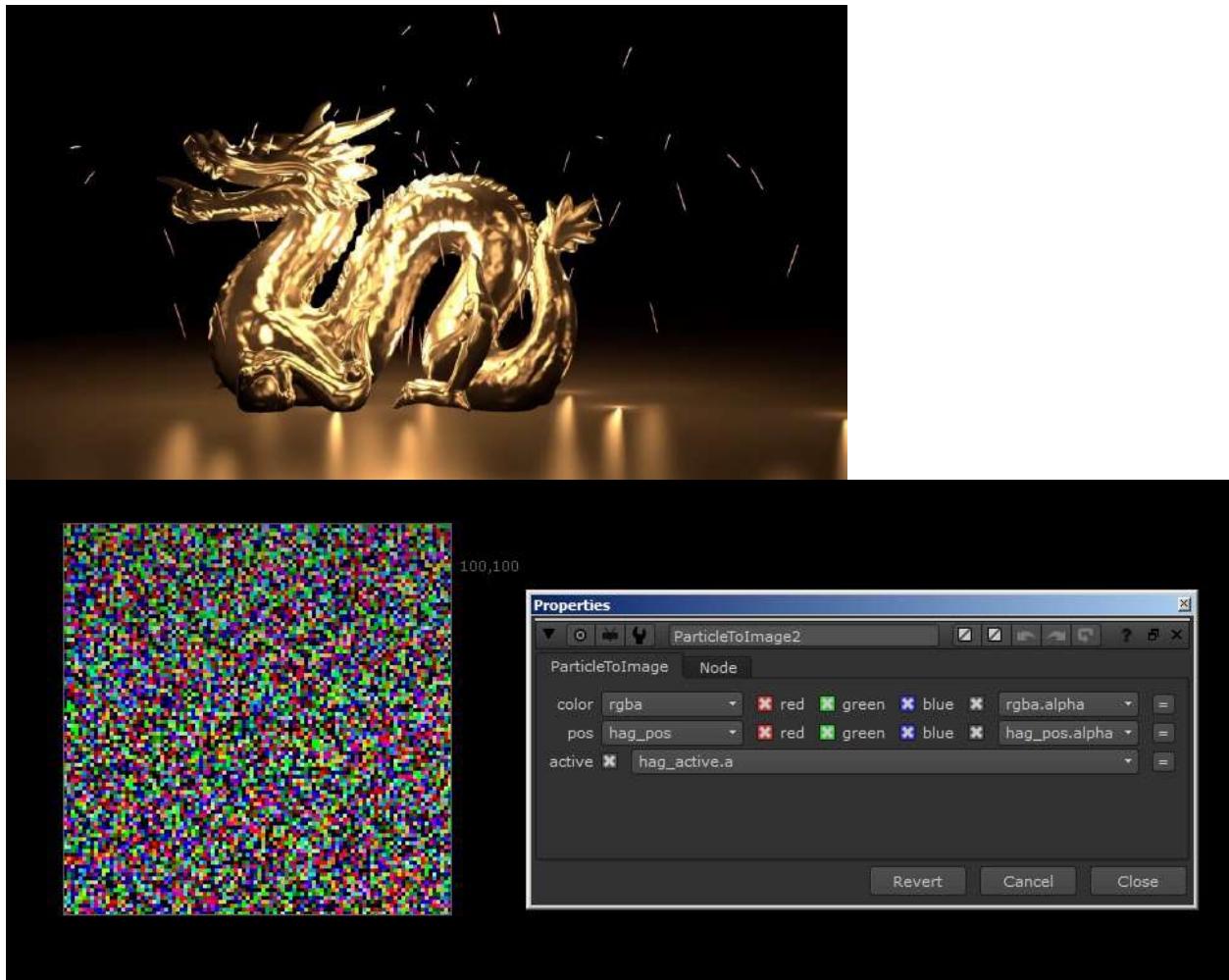
While this tool will never replace the real thing, it will definitely be more than enough when chucked in some background scene, or with some DoF or motionblur. Hopefully you will find some use for it and maybe even learn a little bit more about particles.

For any bug or suggestion, dont hesitate to contact me.

LinkedIn : <https://www.linkedin.com/in/breidenbachdimitri>

ParticleLights MHD

Author: Mads Hagbarth Damsbo - <https://hagbarth.net/blog/>



Nukepedia link: <http://www.nukepedia.com/blink/particles/particle-lights>

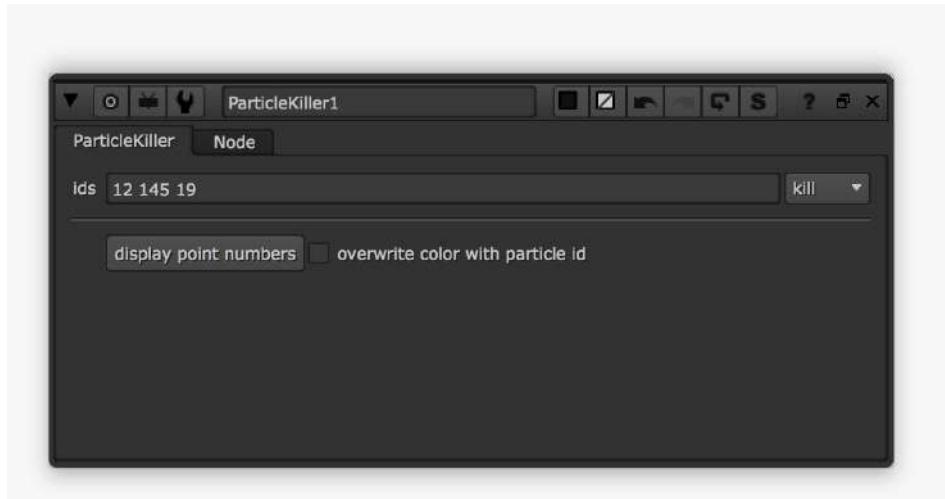
Mads' blog post: <https://hagbarth.net/project/particle-lights/>

Vimeo Demo: <https://vimeo.com/96521727>

Particle Lights is a custom relight node that, enables particles to cast lights into a scene.
Using the new node "ParticleToImage" introduced in Nuke8 to relight a scene using particles.

ParticleKiller NKPD

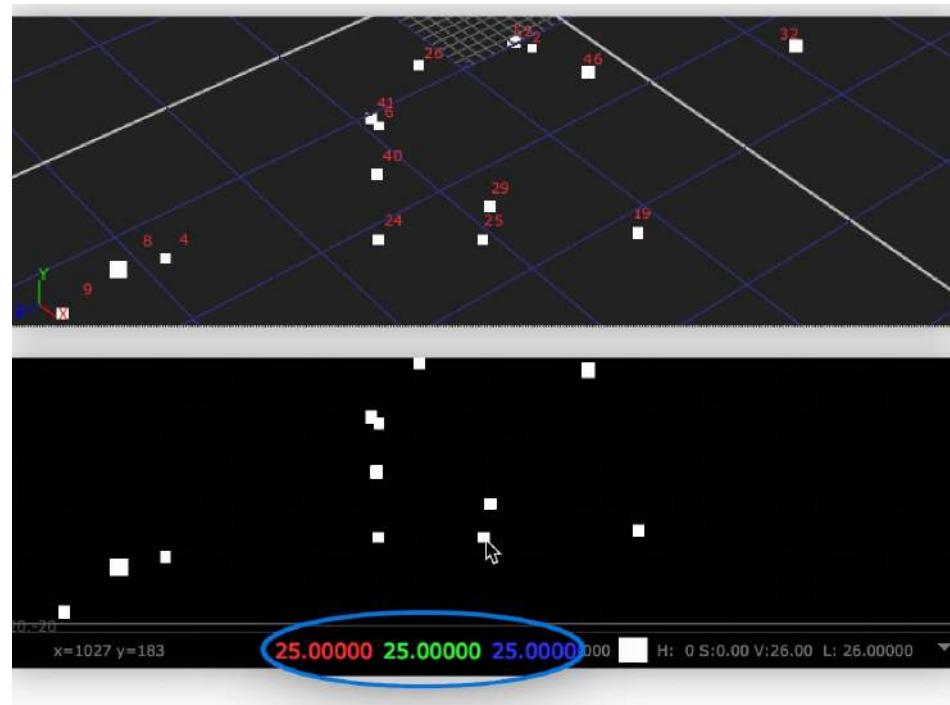
Author: Wouter Gilsing - www.woutergilsing.com



Nukepedia link: <http://www.nukepedia.com/gizmos/particles/particlekiller>

A tool to kill specific particles based on their id.

Options to make identifying individual particles easier:



12.) Deep

Deep2VP Suite MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Nukepedia link: <http://www.nukepedia.com/gizmos/deep/deep2vp>

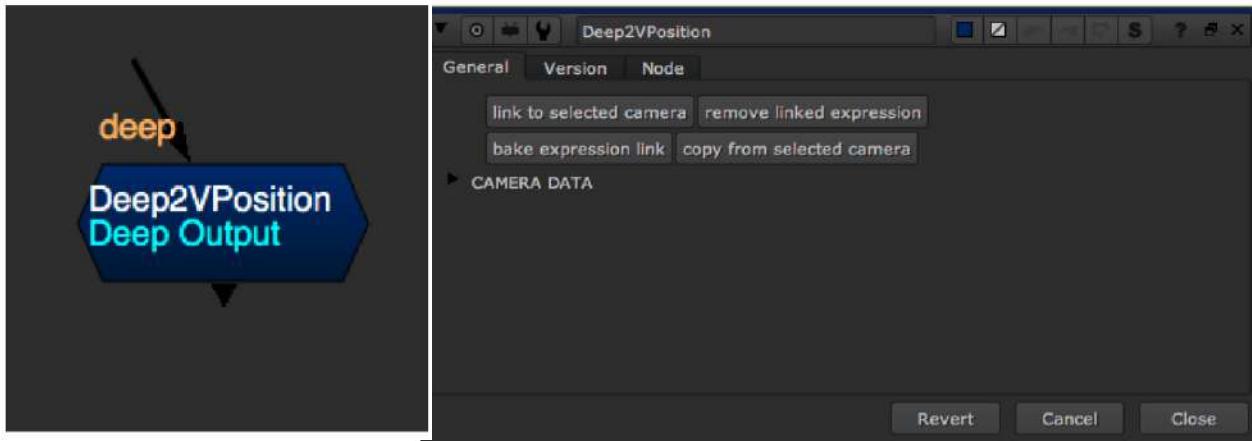
Download Mark Joey Tang's entire Toolset: <http://bit.ly/menupy>

3.5 Showcase : <https://vimeo.com/379527067>

2.0 Demo : <https://vimeo.com/338501133>

Deep2VP suite is a toolset to convert deep data to world space position, and process with all the possibilities of this deepPosition data.

Deep2VPosition MJT



Deep2VPosition is a main node in this entire toolset. It converts deep data to 3D position data and process all the nodes in this toolset. So this node needs to go before any nodes. Once gather the camera data, and it will generate position data under a new channel, called 'deepPosition'. After this channel had been created, then you don't have to add this node anymore in downstream.

'deepPosition' also able to use on any position tools in 2D. Use 'DVPTolImage' converts to this channel to 2D can fix all the semi-transparent pixels and pixel filter.

Linked to selected camera:

All required camera data will be linked to this node by expression. Some companies have their own camera node with a different node class or different knob name, so this button is not limited to specify node class. When default camera knob name cannot find in selected node, it will prompt up a window to let user selected relevant knobs. You can find the world space position data under 'deepPosition' channel after this node. It supported volumetric deep data. If want to bring the position data to 2D in downstream. Please only use DVPTolImage which comes along with this toolset.

Removed linked expression:

Removed linked expression in this node. All the values remain unchanged, but no animation will be baked.

Baked expression link:

Will bake all the animated value from expression, no more connection with any nodes.

Copy from selected camera:

Select a camera and this will copy all the animated values to this node without any connection with any nodes.

User can also copy or linked the camera data manually to the knobs in ‘CAMERA DATA’ group.

This node will also create another channel called ‘deepRawColor’ . Casual user doesn’t need to understand what

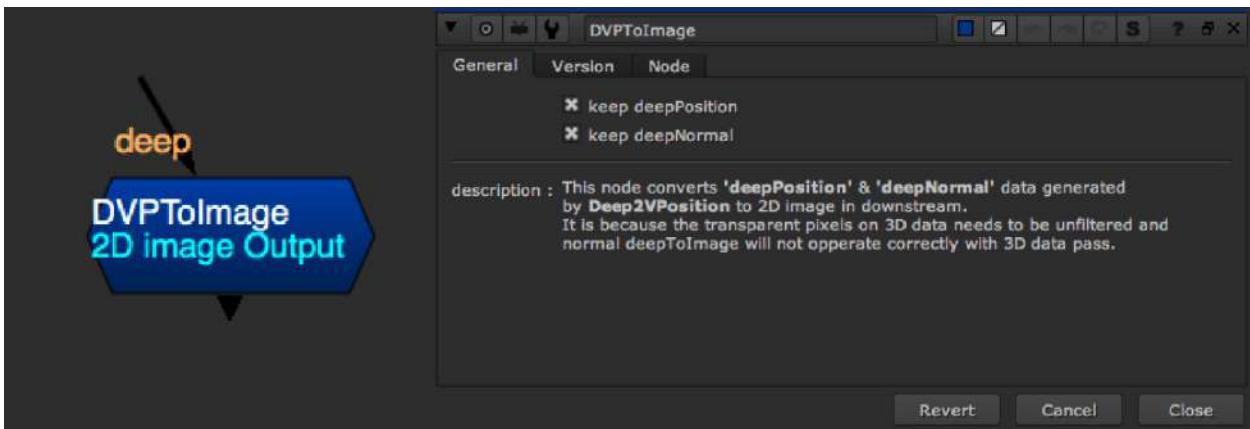
this channel does. This channel will be removed when you use ‘DVPTolImage’ to convert 2D image. It is only for

operation in this toolset. Check out Notes section if you want to understand how the tools use this channel.

* This node supported animated or non-animated re-rack camera since 3.0

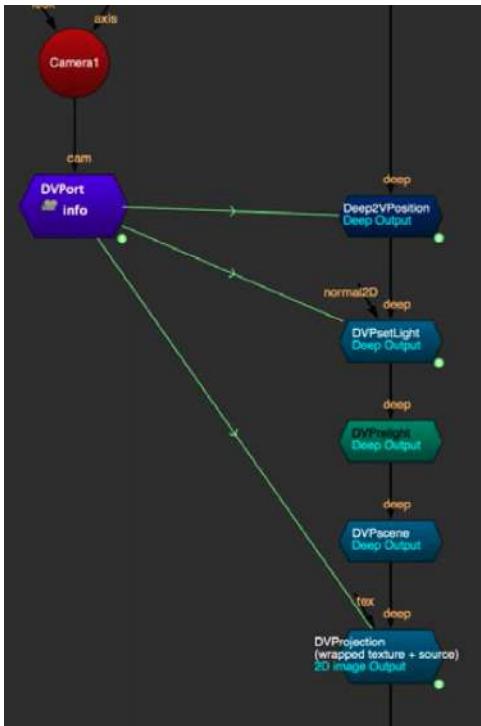
** This node supported pixel aspect in project setting since 3.3 (esp for anamorphic project)

DVPTolImage MJT



This node helps to output 'deepPosition' and 'deepNormal' channel to 2D. Due to 3D data supposed to be non-filtered data, all transparent, semi-transparent and filtered pixel will be fixed to raw non-filtered 3D data. That is regular 'DeepToImage' cannot do beside deep.front and depth.z. If want to keep deepPosition and deepNormal in 2D, required this node to process.

DVPort MJT

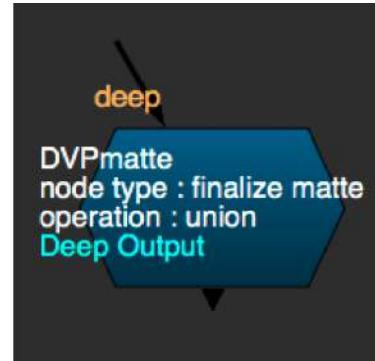


This node will centralize the camera data in one place. Due to the limitation in TCL and speed in python, I choose expression link to camera in this toolset. But one of the requests is get camera input for the tool instead of link with expression. The request concern is when the camera node changed, all the Deep2VP nodes need to re-link again. So the purpose of this node is centralize all the camera data in this node, and set link to this node instead. So when the camera change, just connect the new camera to this node.
The camera can stack with axis nodes, the toolset will only gather world matrix for any camera transformation (beside DVProjection). It doesn't expect any nodes between camera and this node. It's because it is using 'input.xxxxx' TCL to fill up the camera information.
* I don't use [topnode] because concern axis stack.

'KnobChanged' on parent node will slow down the nuke script.

DVPmatte MJT

Position matte is very common in CG compositing. DVPmatte is same as position matte but with deep in real 3D space. It is taking advantage of using deep to get real z matte and separate non-filtered deep samples and pixel filter in alpha. In 3.5, this tool can combine with multiple matte with different operation, so it can generate more shape than just sphere, cube and cylinder shape.



This tool has 4 different ways to sample the position where you want to apply. 2D sample, transform preview geo in 3D, link selected axis and import chan file.

Matte shape:

Select the 3D shape of the matte. 3D shape can be preview under 3D view when the panel of this node is active.

Sphere : only support uniform falloff.

Cube : support separate 3-axis falloff.

Cylinder : only support uniform falloff.

Invert matte:

Invert the matte of the shape. Same as deepHoldout but with falloff support (soften matte).

Node type:

finalize matte: Premult all the mattes in the 'DVPmatte' stack to get the final matte.

multi matte: Keep the matte and pass along to downstream for multi mattes purpose. It will only premult RGB for 2D preview.

Options:

premult RGBA: Premult RGBA in volumetric deep. It good to carry for deep comp in down stream. May find some edges has stronger alpha in 2D view. That is because of the samples overlap in front and back, but the data per sample are accurate.

black & white matte: This will fix the samples overlap look, but not good for deep comp.

Recommend convert it back to 2D and shuffle the black & white to alpha for 2D comp purpose.

Toggle 2D sample:

By default, this option is off. Turn this on, then you can sample the image in 2D view. When you are done, remember to turn this off again. Otherwise it will sample the value in every frame.

Sample position:

This knob is disable by default. Active when 'Toggle 2D sample' is on.

Transformation:

Then this node is active (shown in panel), user switch to 3D view to preview the matte in 3D space. Snap the pointcloud or move the preview geo manually.

Linked to selected axis/geo:

Use this function to link the transformation of 3D node to this node.

Remove linked expression:

Removed linked expression in this node. All the values remain unchanged, but no animation will be baked.

Baked expression link:

Baked out the existing expression link back to this node itself. After that, the linked node is not required for this node anymore. Baked value is base on the frame range in project setting of the nuke script.

Copy from selected axis/geo:

Select the axis/geo, then this will process 'link to selected axis/geo' and 'bake expression link' in once.

Pointcloud:

When this node panel is active, pointcloud will be shown under 3D view.

None: will not show the pointcloud in 3D

raw pointcloud: show the pointcloud from input data, without any effects from this node.

with matte applied: This can show the instant result. Since it feedback in real time, it might slow down the progress in 3D space.

Cube 3D falloff:

This will only applied when matte shape selected as cube.

Falloff type:

6 different falloff types

Uniform falloff:

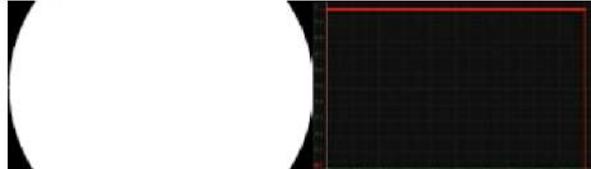
This is a global falloff value. Support for any shape of matte.

Exponential:

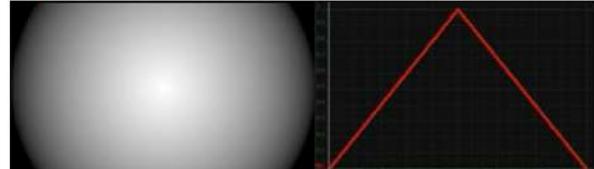
This knob will enable when exponential had been selected in falloff type.

Fall off types:

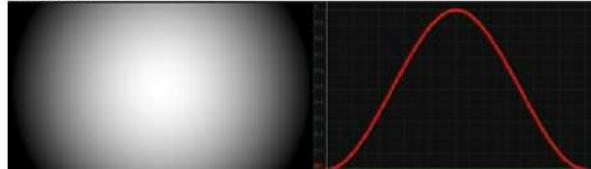
None



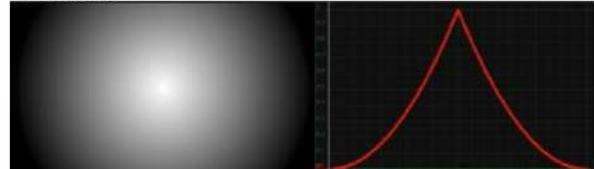
Linear



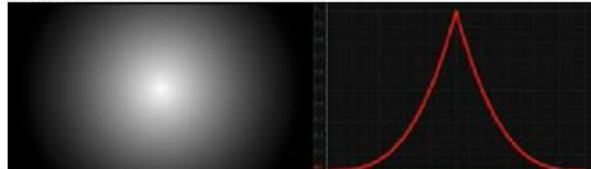
Smooth



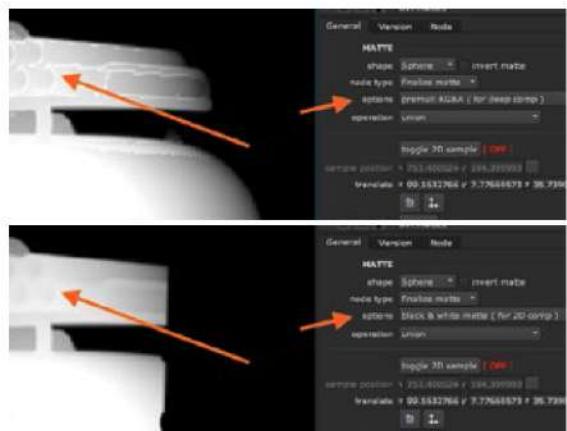
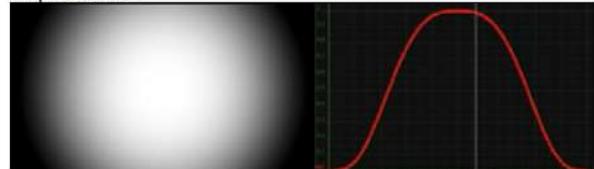
Quadratic



Cubic

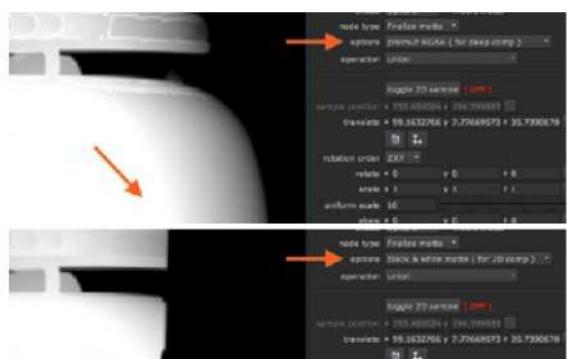


Exponential

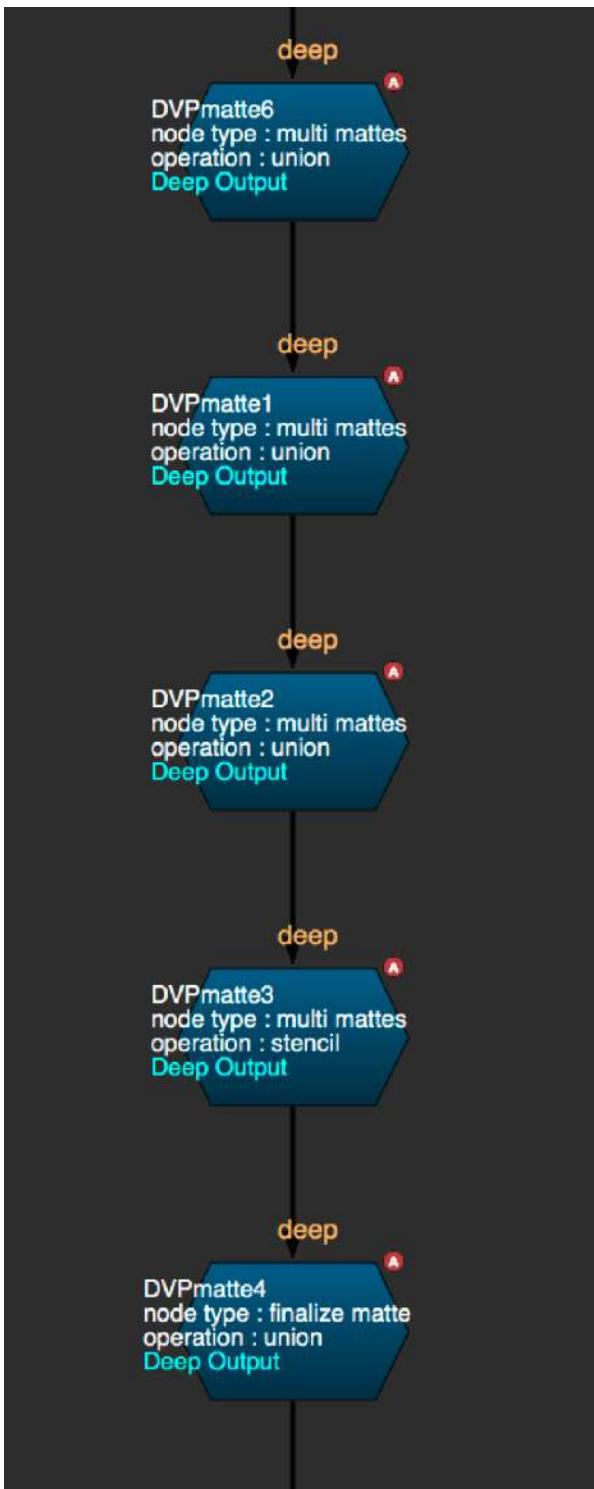


Output options:

Matte on deep, you might see this kind of edges on semi-transparent pixel. That's because of multi sample layers in the same screen space coordinate, but it does make sense and correct in 3D space. So if compositing in deep, it all works fine. If you want to output the matte in 2D, it might give you edge issue. So 'DVPmatte' options can give you a black & white matte that can be used in 2D comp.



But black & white matte will make you lose the possibilities of matte in z axis. Be consider what the needs in your comp.



Mutli-matte workflow

Multi matte workflow needs to set all the node type to multi matte, but the last one set to finalize matte. The first multi matte node's operation must be 'union'.

DVPattern MJT

Position noise is one of the common nodes in CG compositing. DVPattern is also doing the same thing in deep, has 7 different patterns.

They are: fBm, turbulence, noise, random, stripes, ripple and rays.

Pointcloud:

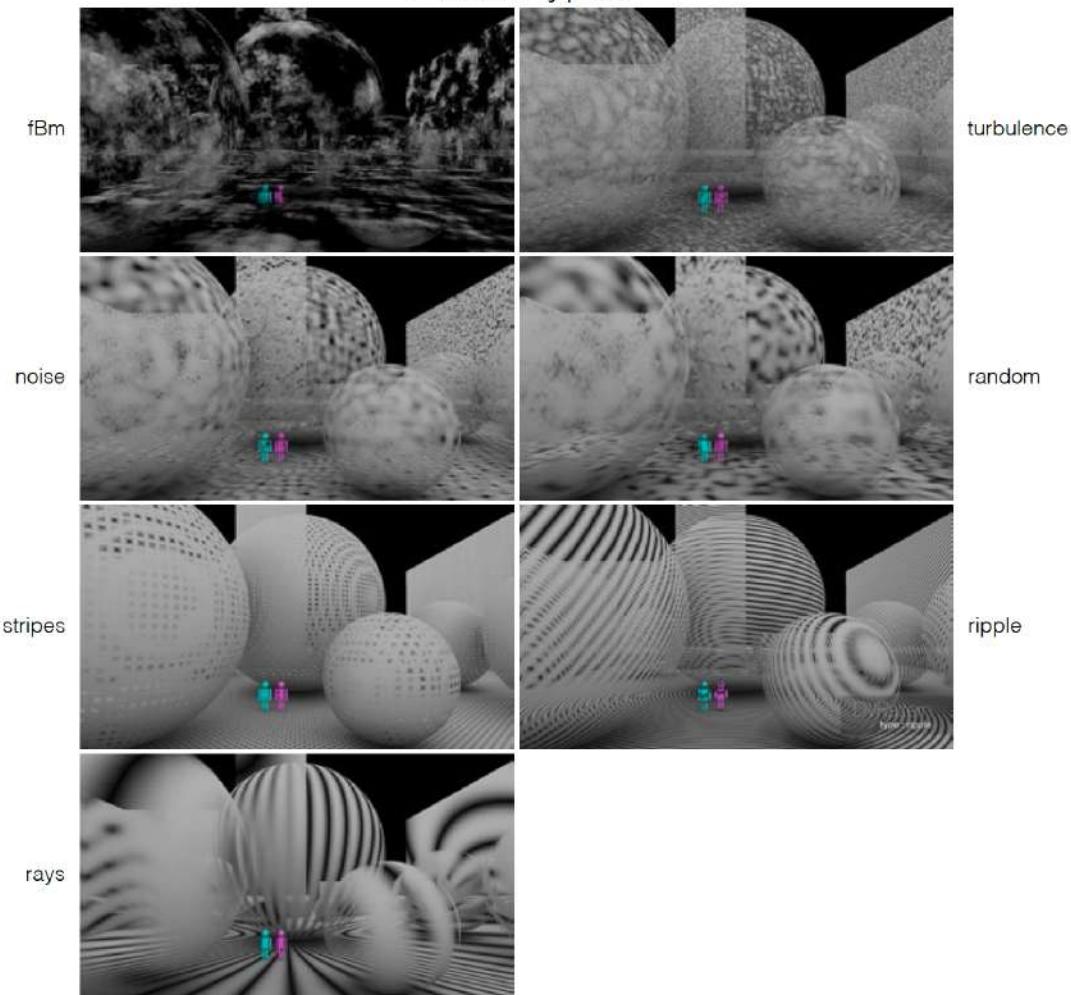
When this node panel is active, pointcloud will be shown under 3D view.

none: will not show the pointcloud in 3D

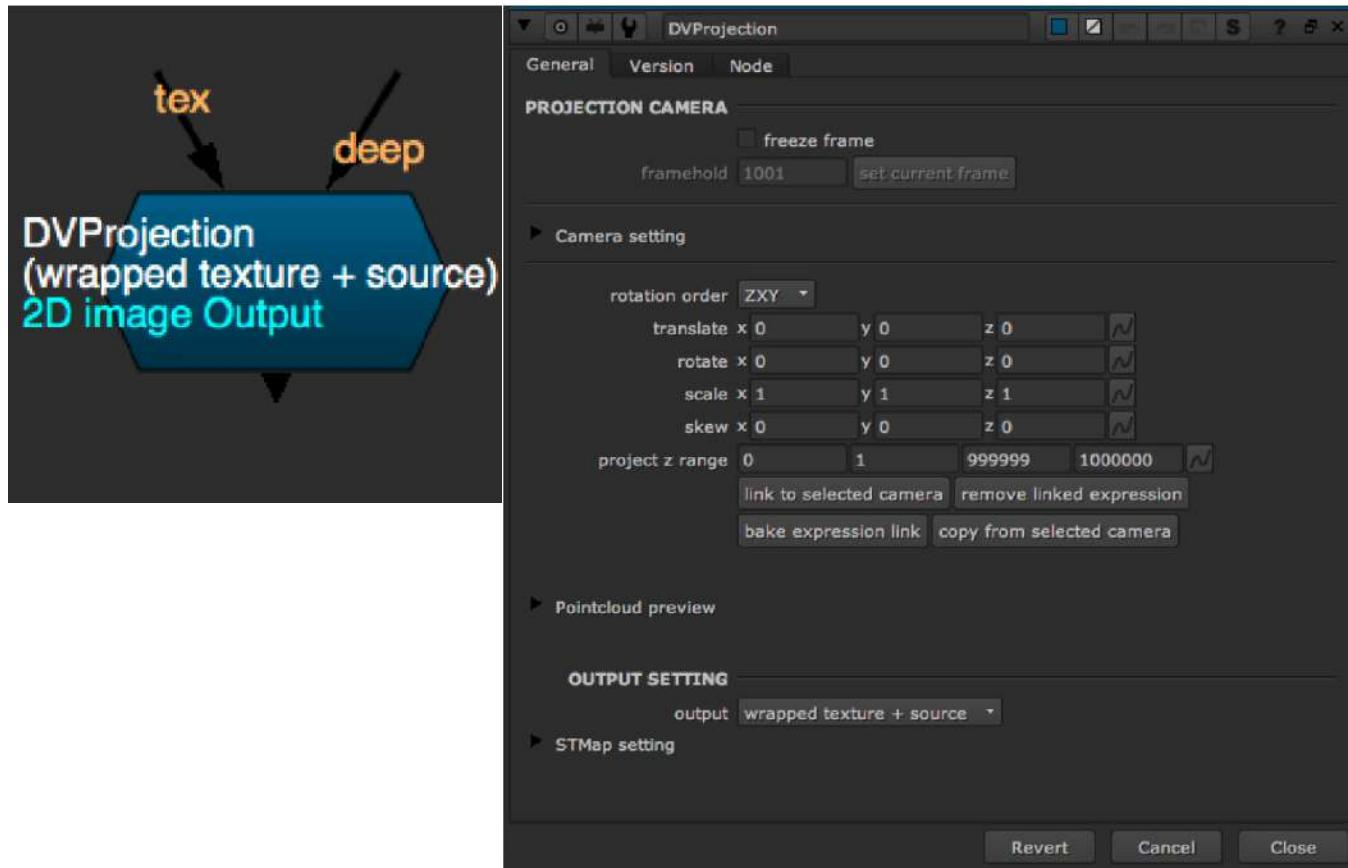
raw pointcloud: show the pointcloud from input data, without any effects from this node

with pattern applied: This can show the instant result. Since it feedback in real time, it might slow down the progress in 3D space.

Pattern types



DVProjection MJT



This node can process projection on deep. It can be used with the current shot camera, or place the camera manually. Output of this node is 2D image, so it doesn't need 'DeepToImage' or 'DVPToImage' in downstream. On the other hand, the deep data will be lost after this node.

Settings:

Freeze frame:

Check this box to enable framehold function for animated camera

Framehold:

Input the frame number and freeze the projection camera's animation

Set current frame:

Automatically set frame frame into framehold knob.

Project z range:

Set the projection distance from the projection camera (in term of distance)

First value : start ramp in value

Second value : end of ramp in value

Third value : start ramp out value

Forth value : end of ramp out value

Link to selected camera:

Selected any Camera nodes and this button will link the position to the projection camera.

Removed linked expression:

Removed linked expression in this node. All the values remain unchanged, but no animation will be baked.

Baked expression link:

Will baked all the animated value from expression, no more connection with any nodes.

Copy from selected camera:

Select a camera and this will copy all the animated values to this node without any connection with any nodes.

Pointcloud:

When this node panel is active, pointcloud will be shown under 3D view.

none: will not show the pointcloud in 3D

raw pointcloud: show the pointcloud from input data, without any effects from this node

with project image: This can show the instant result. Since it feedback in real time, it might slow down the progress in 3D space.

Output

Support 3 different outputs

Wrapped texture + source: which is a projected texture composite with the input.

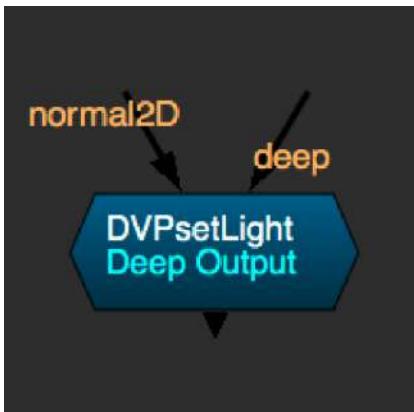
Wrapped texture: which is only projected texture

UV: which is a uv map, can use this with STMap in downstream.

DVP Relight setup

Let's introduce a basic setup for relight before I walkthrough the nodes in toolset. For relight in deep, I breakdown 2-3 nodes in the setup for multi lights supported, raw light pass output and light with beauty output. DVPsetLight always be the first node for relight preparation, then add DVPrelight nodes as many as you need. If you need raw light only, you can convert it to 2D image directly. If you are lighting on beauty and keep the deep comp, connect DVPscene at the end.

DVPsetLight MJT



This node needs to go before DVPrelight. Setup how the normal data use in deep for relight. This node will provide 4 options to pipe in normal data into deep stream.

Generate normal by estimate from position:

If no any normal data provide, this could be one of the options. The result might not perfect, might have some edges problem on individual objects. If deep data from scanlineRender, geo required high level subdivision.

Required camera data for this option.

Input 2D normal (world / camera space):

If have normal data in 2D data, connect the pass to this node. The resolution of normal2D input must be the same as the deep resolution. Camera space normal required camera data, but not for the world space.

Deep normal:

Use deep normal data from deep input. Type in the channel contain normal data in deep.

For example : channel name is ‘Nworld.red’ , ‘Nworld.green’ , ‘Nworld.blue’. Type in ‘Nworld’ for the channel name.

* When using scanlineRender for deep output, set output vector normal, then you can have normal data in deep format.

Link to selected camera:

Selected any Camera nodes and this button will link the position to the projection camera.

Removed linked expression:

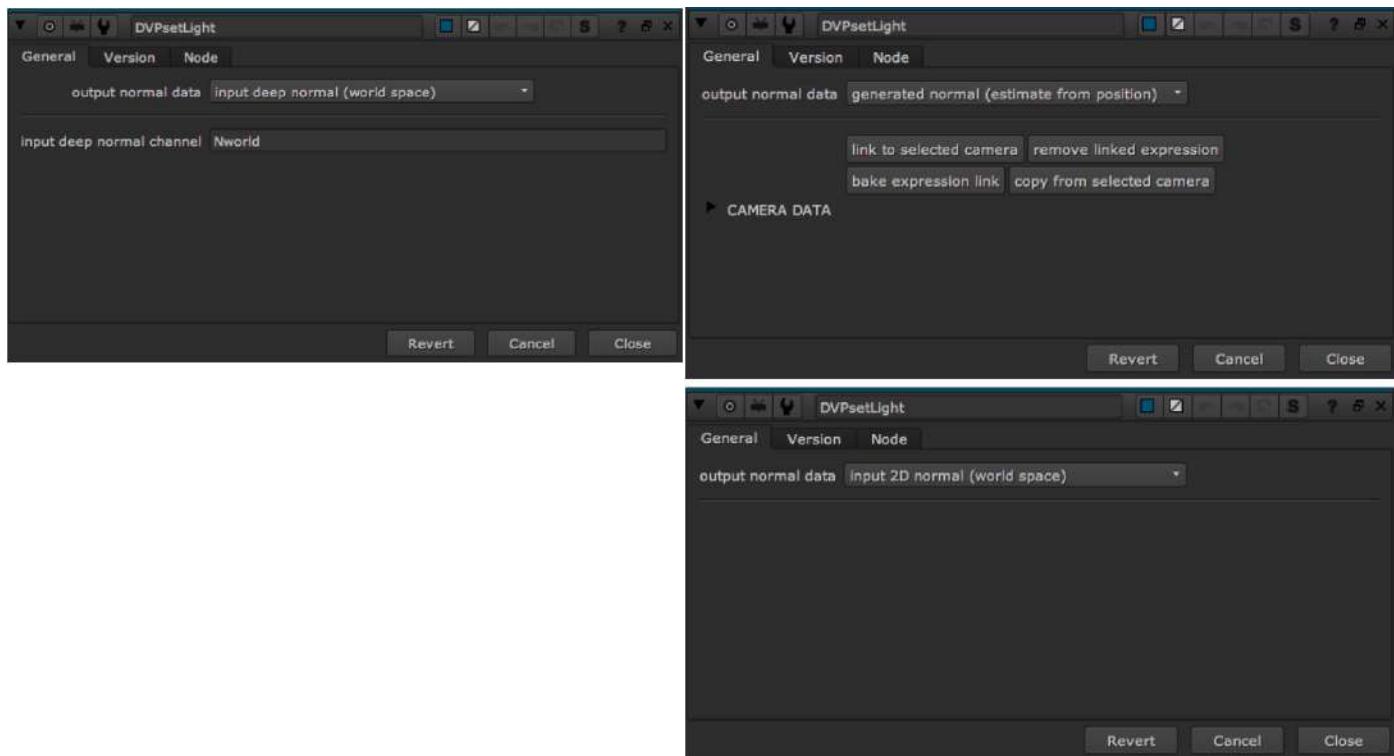
Removed linked expression in this node. All the values remain unchanged, but no animation will be baked.

Baked expression link:

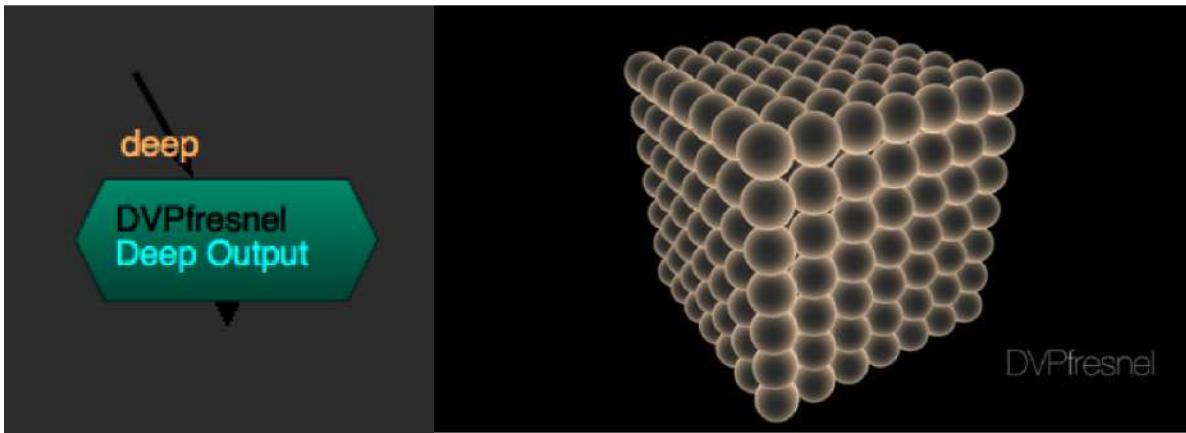
Will baked all the animated value from expression, no more connection with any nodes.

Copy from selected camera:

Select a camera and this will copy all the animated values to this node without any connection with any nodes.

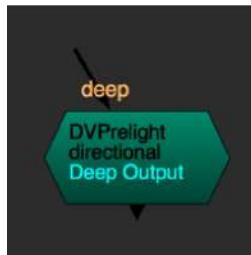


DVPfresnel MJT



DVPfresnel is part of the lighting node in this toolset. It is because it required normal data. It created fresnel (aka facing ratio) to every objects in the scene. This node can stack with multiple 'DVPrelight' nodes together. The node itself output raw light data in deep format. Connect to 'DVPscene' to get this raw fresnel composite with beauty.

DVPrelight MJT



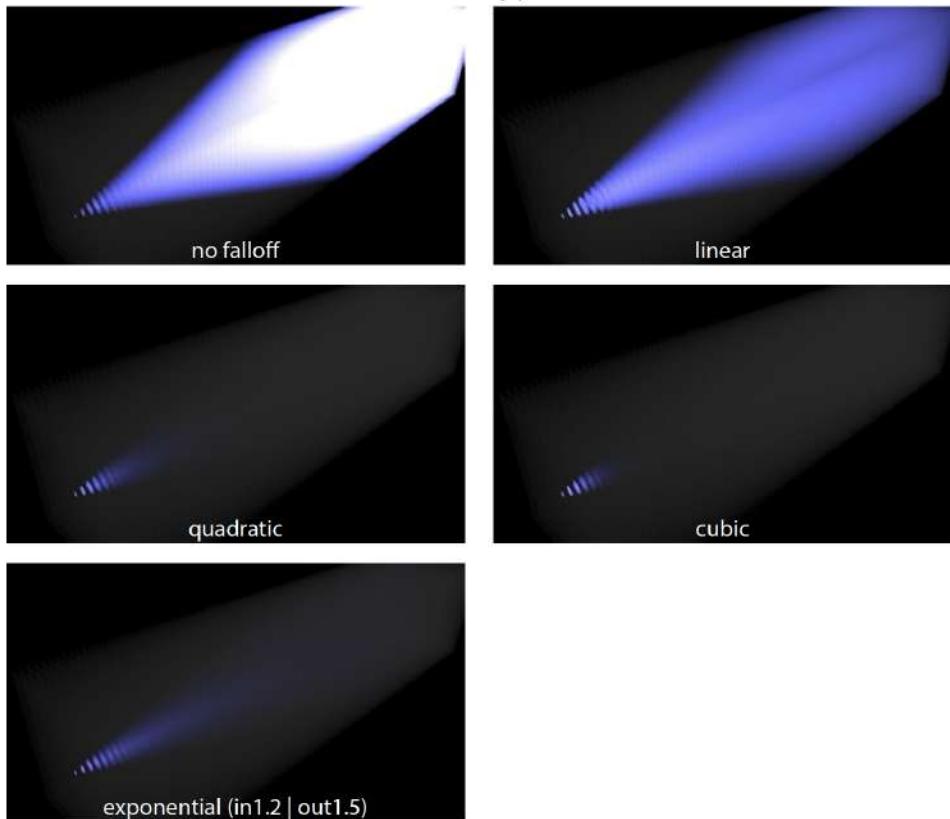
'DVPrelight' node has 3 types of light as default nuke light node. It can stack with multiple nodes together and will output raw light in deep format. Connect 'DVPscene' to composite with beauty. The setting in this node is almost same as nuke basic light node, but with extra exponential falloff type, to get a flexible result.

Pointcloud:

When this node panel is active, pointcloud will be shown under 3D view.

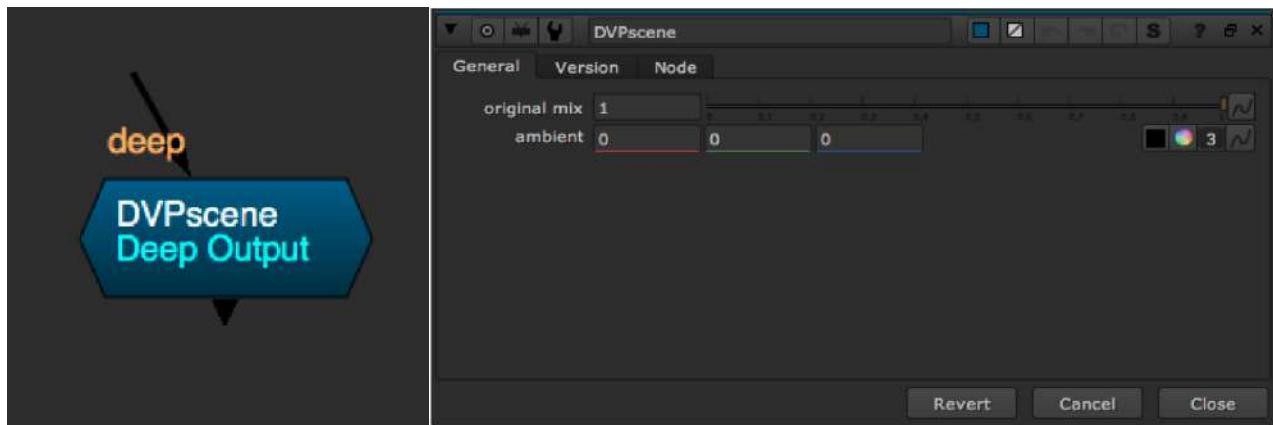
None	will not show the pointcloud in 3D
raw pointcloud	show the pointcloud from input data, without any effects from this node
with light applied	This can show the instant result. Since it feeds back in real time, it might

Falloff types



slow down the progress in 3D space.

DVPscene MJT



This node will composite the relight process in deep format.

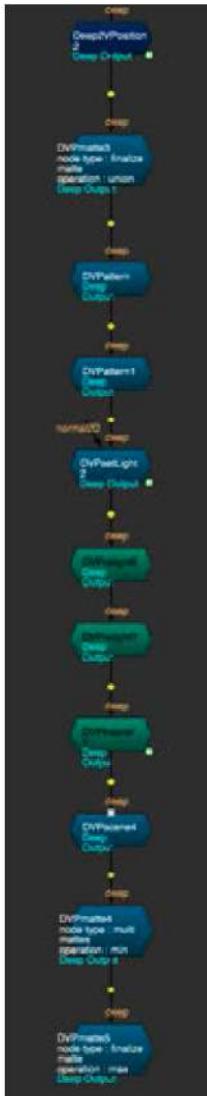
Original mix:

Combine the original color in diffuse.

Ambient:

Add ambient color on top of diffuse, before the rawlight

Nodes:



Stack DVP nodes:

All the nodes in the toolset can stack together, only green nodes need to connect between ‘DVPsetLight’ and ‘DVPscene’. Blue nodes cannot put inside the lighting setup, otherwise will break the setup.

DeepRawColor:

‘Deep2VPosition’ and ‘DVPsetLight’ will generate a channel called ‘deepRawColor’. The channel helps to pass the data from upstream to process stack operations. Regular user might not need to understand how this process. If anyone want to troubleshoot the setup, the following is the Information.

In ‘DVPmatte’, current matte processed under deepPosition.alpha, and then deepRawColor.alpha stored the matte from upstream plus current matte. When selected ‘multi mattes’, all the matte process will store under deepRawColor.alpha, but will not applied on alpha because of the premult process in deep. When selected ‘finalize matte’, the deepRawColor.alpha will process to RGBA and removed the deep samples when alpha ≤ 0.0 . And removed samples cannot restore anymore in downstream.

In Deep2VP lighting setup, because requested to have raw light output and concern multiple light apply, so ‘DVPsetLight’ will shuffle RGB to deepRawColor.RGB and put RGB to black.

In ‘DVPrelight’ and ‘DVPfresnel’, light matte process under deepNormal.alpha and apply the matte to RGB for the raw light. ‘deepRawColor’ remain untouched until ‘DVPscene’ node connected.

Current light will combine with input RGB to output the new raw light. That's why 'DVPsetLight' set RGB to black.

In ‘DVPscene’, it will bring back `deepRawColor.RGB` and process with RGB (raw light) to return

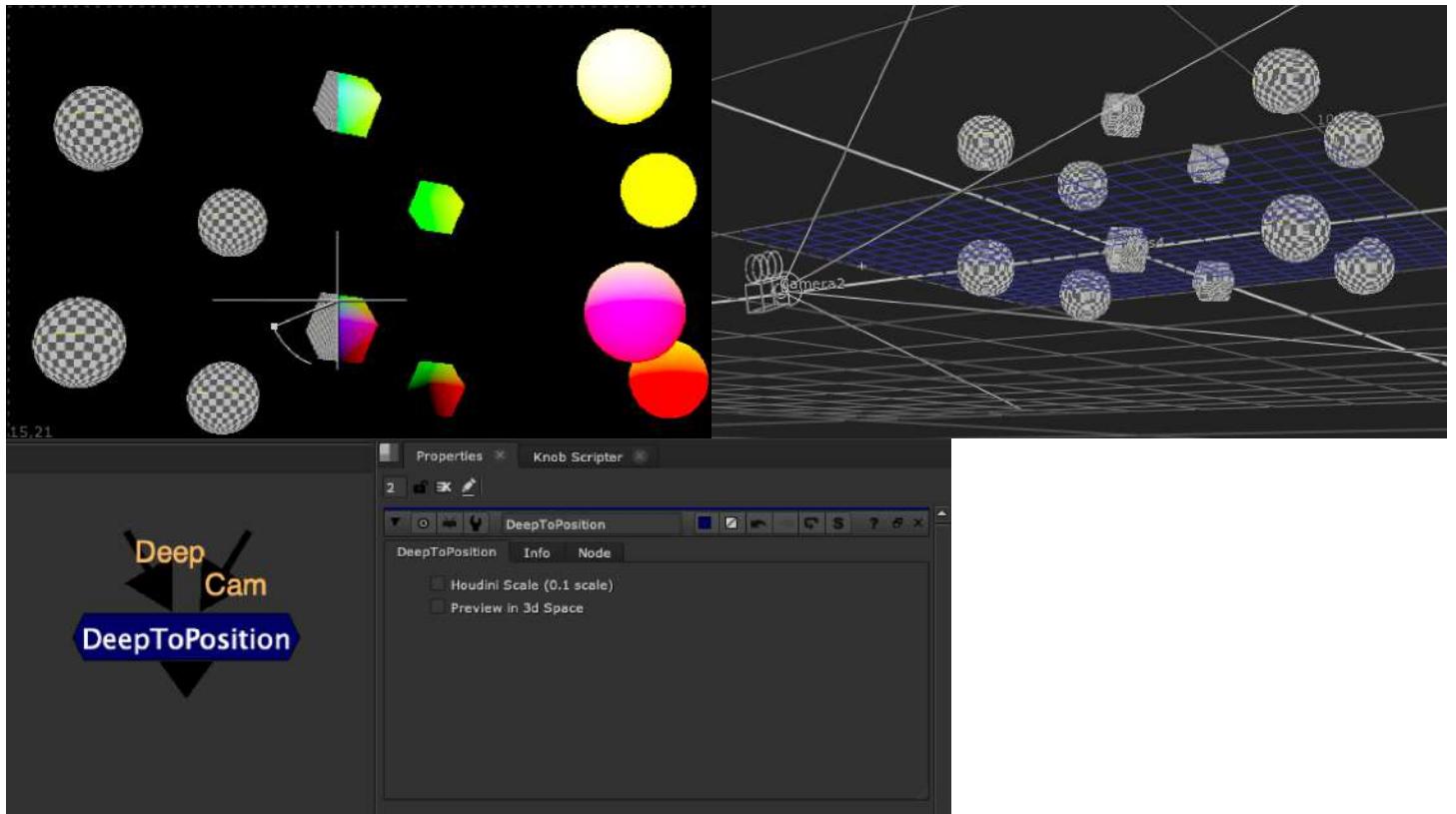
the color with new lighting. Like how diffuse color combine with raw light in AOV. The algorithm

is :

(input deep color + ambient color) * raw light color

DeepToPosition TL

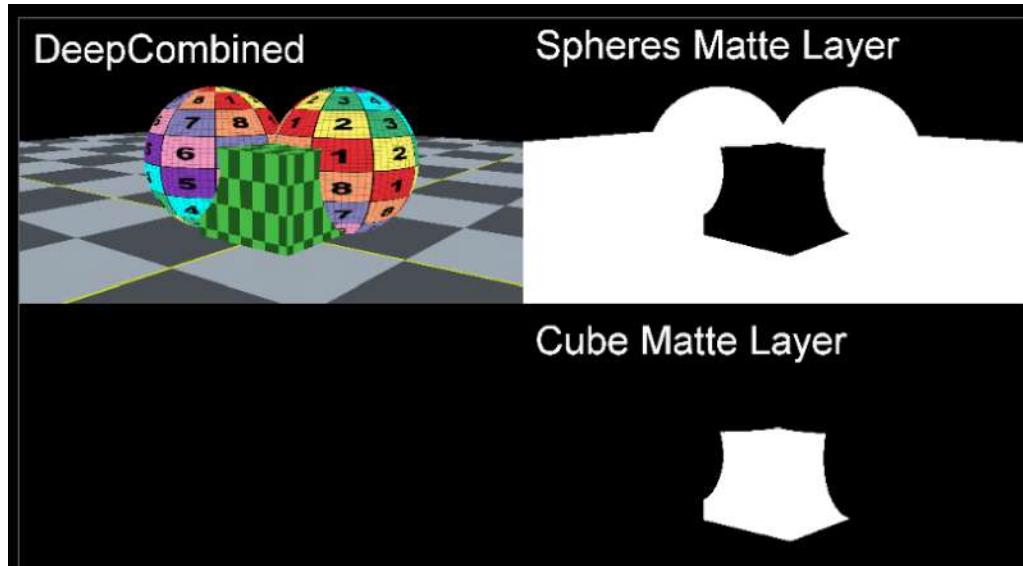
Author: Tony Lyons - www.CompositingMentor.com



Plug in your deep render and your camera and this node will convert the deep into a WorldPosition pass. There was an option when using Houdini renders that the scale was off so there is a houdini scale compensation checkbox
There is also a preview in 3d space option for seeing where your deep exists in nuke's 3d viewer.

DeepRecolorMatte TL

Author: Tony Lyons - www.CompositingMentor.com

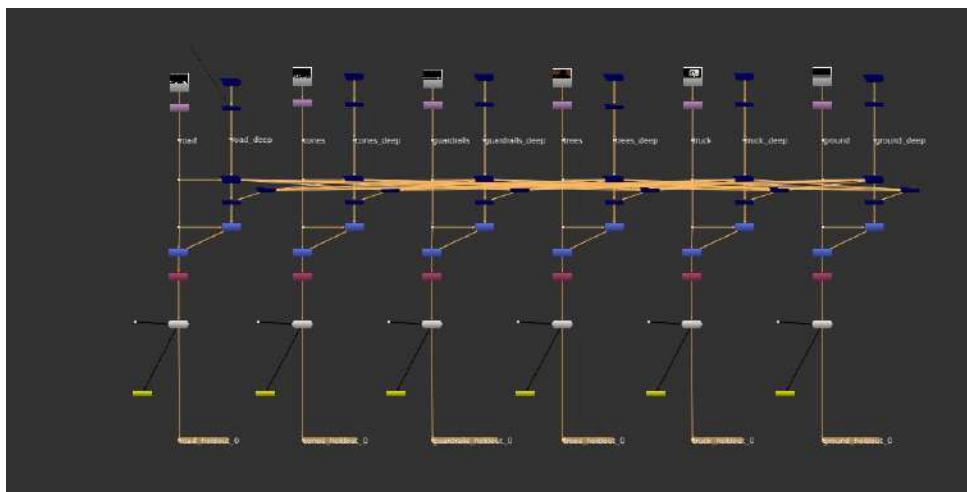


There is a popular deep workflow used in production where you deep combine many elements together, and have to make deepHoldouts for each layer, usually pre-rendering out the alpha channel for each individual layer, to use later, in order to reduce heaviness.

A really great article is written about this method from Boris MC, which can be found here:
<http://boris-mc.com/?p=2700>

There is also a nice video explaining the workflow here:
<https://vimeo.com/429161580>

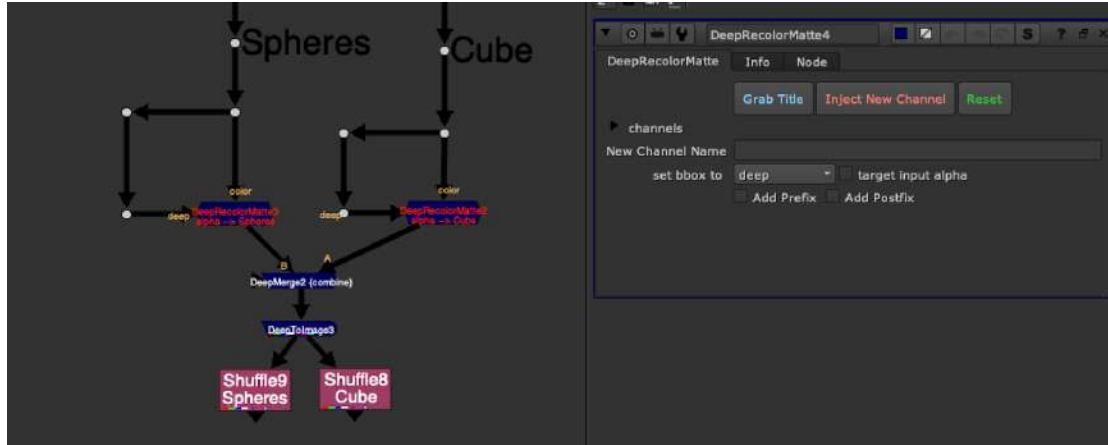
This workflow can often get chaotic quite quickly with lots of objects holding out each other:



This tool aims to speed up and simplify this workflow even further. It creates a new layer with the alpha channel of the element and a custom name.

Using additional channels and deepMerging all of the elements together into 1 big scene, you can later pull out the mattes from each individual element, giving you the matte of that element, held out by all other elements in the scene. This is a custom holdout matte for that object.

The goal would be to prerender just 1 render with all channels, and simply shuffle out the deep holdout mattes for any element you need.



Grab Title button is using functions from Adrian Pueyo's stamps plugin to try and predict the layername from the deep input. It tries to grab the name from the nearest stamp, or the nearest read node filePathName, in the same way that stamps does, so if you are using stamps, this should work for you, and if not, it's ok, you can just enter the name manually.

After you've auto-grabbed the layer name, or entered a manual layer name in the New Channel Name field, you can choose to add a prefix or postfix to the layername. Check the box and enter a prefix or postfix. For example, you can add ID to the prefix and the layername will be "ID_newLayer" or "matte" to the postfix, and the layername will be "newLayer_matte". No need to include the underscores, this tool will add the underscores between the prefix / layername / postfix automatically.

Once you are ready, click Inject New Channel, and the node's text will turn red and label will change to let you know that you have injected a new Channel. All channels are just a single .red channel, to save time during the deep merge process, since each layer and each channel must be calculated, you'll want as few remaining as possible. The tool only saves RGBA, and the new channel and removes all other channels to save calculation time.

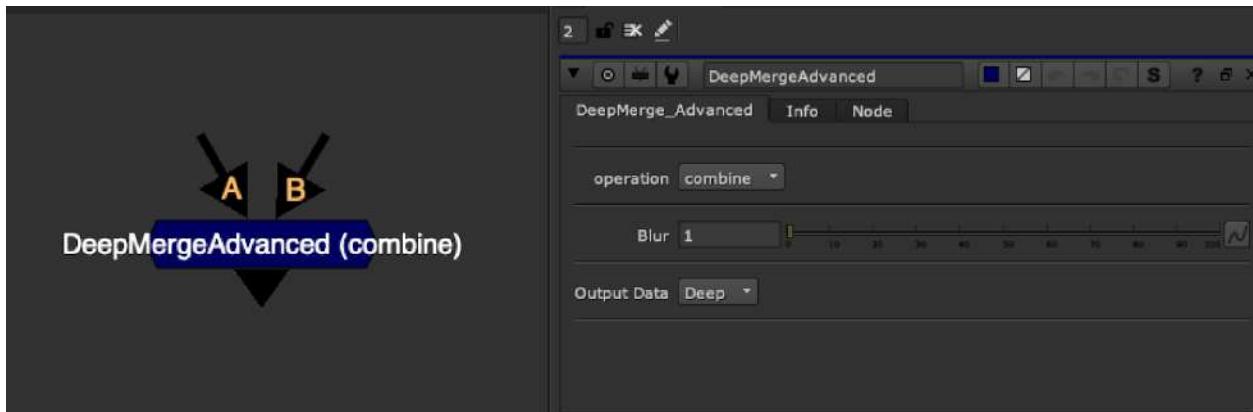
Click Reset to restore default state of the node

Turn on Target Input Alpha if your alpha from your color input is different than the alpha from your deep input. If you've roto'ed something out for example

Remove Deep channel, Prerender the Deep Tree after DeepToImage, Shuffle out the mattes

DeepMerge_Advanced BM

Author: Ben McEwan - <https://benmcewan.com/nukeTools.html>

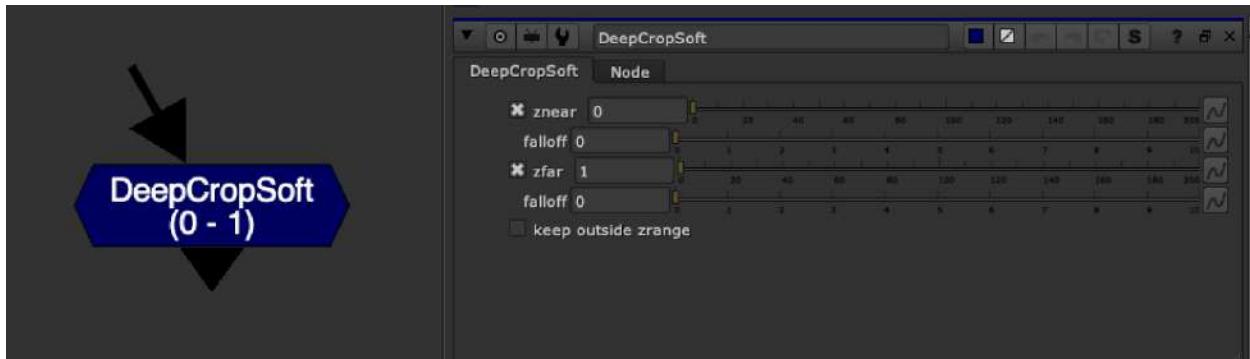


Download from github: https://github.com/BenMcEwan/nuke_public/tree/master/gizmos

- Merges all channels from A and B in deep.
- Fixes issue with DeepMerge's holdout operation which exists in Nuke 11. Choose between a Deep or 2D output.
- Soften/ Blur slider which helps blend the 2 deep renders using a DeepExpression and deep.front and deep.back

DeepCropSoft NKPD

Author: Wouter Gilsing - www.woutergilsing.com



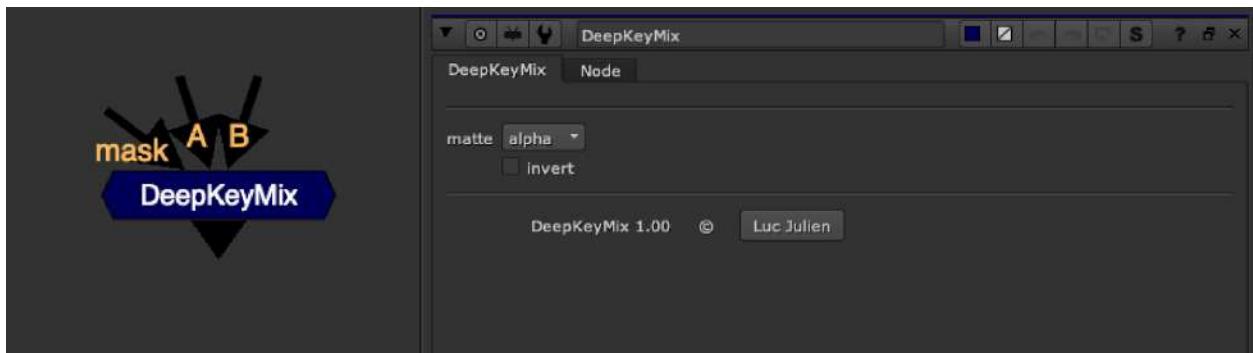
Nukepedia link: <http://www.nukepedia.com/gizmos/deep/deepcropsoft>

A version of the DeepCrop node that allows the user to set a falloff for a soft transition. Can be used to gradually fade off anything that reaches a specific distance from the camera (like anything super close). This can for example be useful to prevent visible 'popping' when a camera moves through geometry.

**Minor tweaking of the original knob positions to make it slightly easier to use.

DeepKeyMix NKPD

Author: Luc Julien



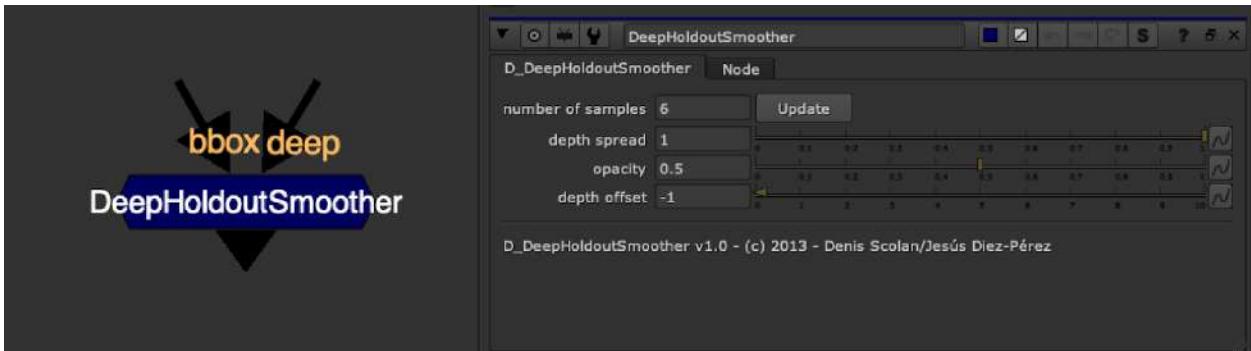
<http://www.nukepedia.com/gizmos/deep/deepkeymix>

Same basic function as the original keymix but for deep input.

It enable you to copies deep channels from A to B only where the mask input is non-zero. The mask input use standard channels.

DeepHoldoutSmoothen NKPD

Author: Denis Scolan + Jesús Diez-Pérez



Nukepedia link: http://www.nukepedia.com/gizmos/deep/d_deepholdoutsmoother

Smoothing the harsh holdout intersection that occurs when the holdout itself doesn't have enough samples.

I've made a gizmo to smooth the harsh holdout intersection that occurs when the holdout itself doesn't have enough samples (coming from a standard Z pass for instance).

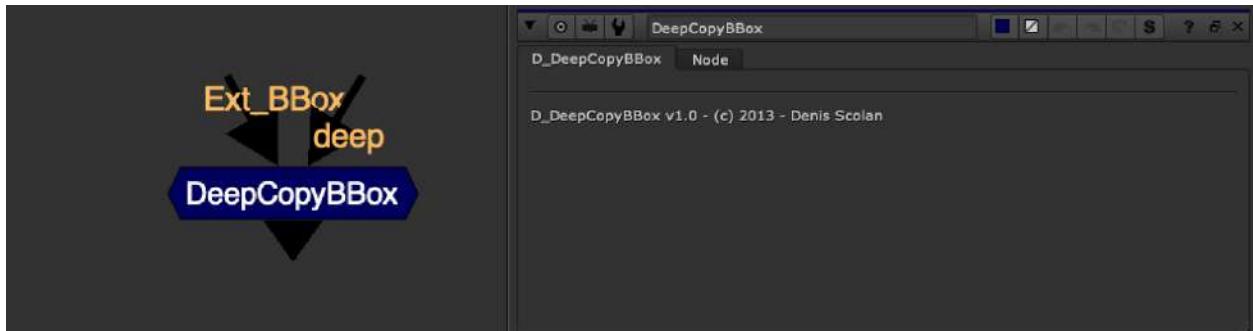
The D_DeepHoldoutSmoothen is meant to be inserted in your tree on the holdout stream and before the DeepHoldout/DeepMerge(holdout mode) node.

Keep in mind that the holdout will expand slightly and also the more you increase the samples the heavier your comp will get.

Thank you to Jesús Diez-Pérez for the Python script.

DeepCopyBBox NKPD

Author: Denis Scolan



Nukepedia link: http://www.nukepedia.com/gizmos/deep/d_deepcopybbox

CopyBBox for deep image stream

this gizmo helps you carry the BBox of your 2D stream onto your deep image stream.

It's useful for reintroducing the BBox after a DeepMerge in 'holdout' mode.

Down side : you can only copy the BBox from a 2D stream, it doesn't work with a deep stream yet.

DeepBoolean MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Nukepedia link: <http://www.nukepedia.com/gizmos/deep/deepboolean>

Download Mark Joey Tang's entire Toolset: <http://bit.ly/menupy>

Demo video: <https://vimeo.com/354811205>

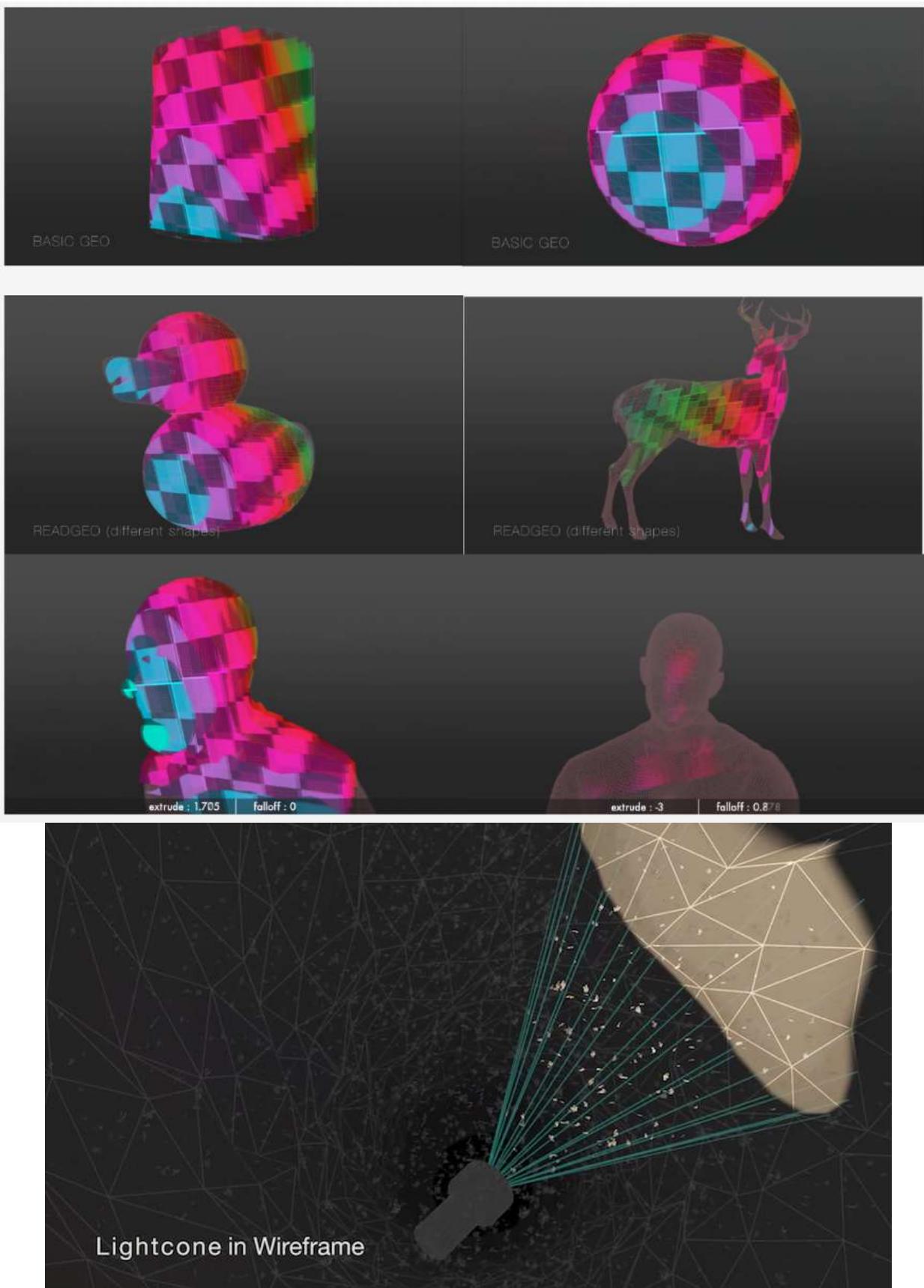
Demo Video: <https://vimeo.com/322695922>

DeepBoolean works like maya's boolean's tool, but for deep in nuke. It works like deepHoldout but more function than that. Intersect mode to keep the deep data inside the geo matte. Subtract mode to holdout the deep data outside of the geo matte. Both support geo matte extrude (erode the terms in nuke) and falloff in 3D space.

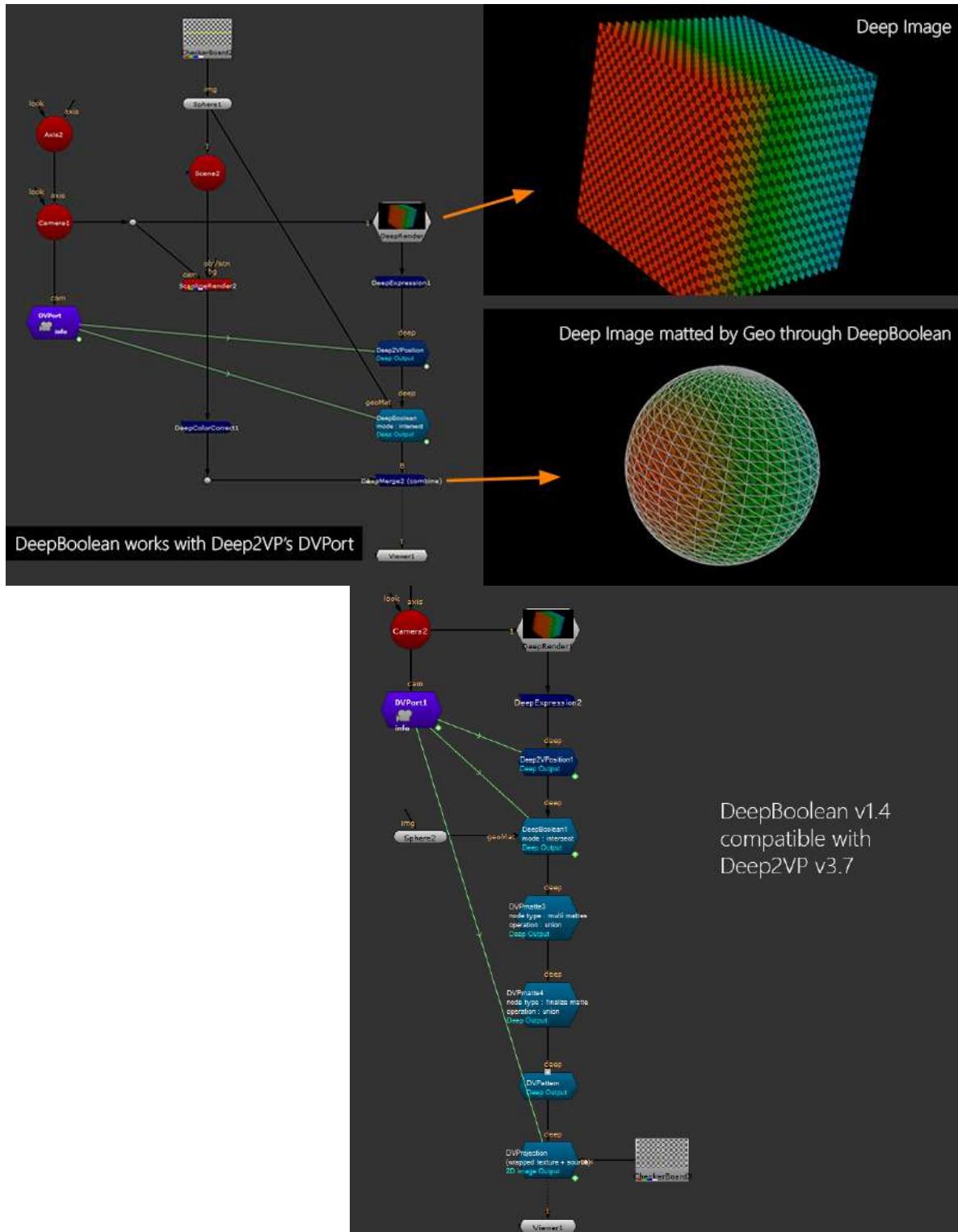
Any geo needs to be close face because this tool is camera space Normal deep matte.

It works with any kind of readgeo or basic geo in nuke (cube might has a bit strange behaviour on extrude because of separate faces).

*Beware: before Nuke11, DeepExpression has some strange behavior, thanks Foundry to fix most of them in Nuke11. So this tool is only work and tested in Nuke11. If you are using lower version, this tool will not work probably.

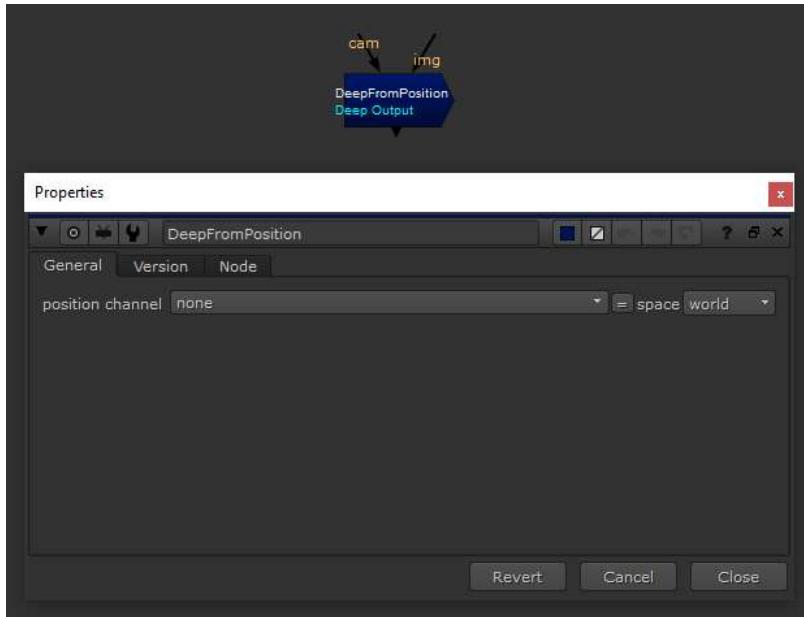


Works with Deep2VP v3.7+ , can link the camera data through DVPort and stack with DVPortmatte.



DeepFromPosition MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Nukepedia link: <https://www.nukepedia.com/gizmos/deep/deepfromposition>

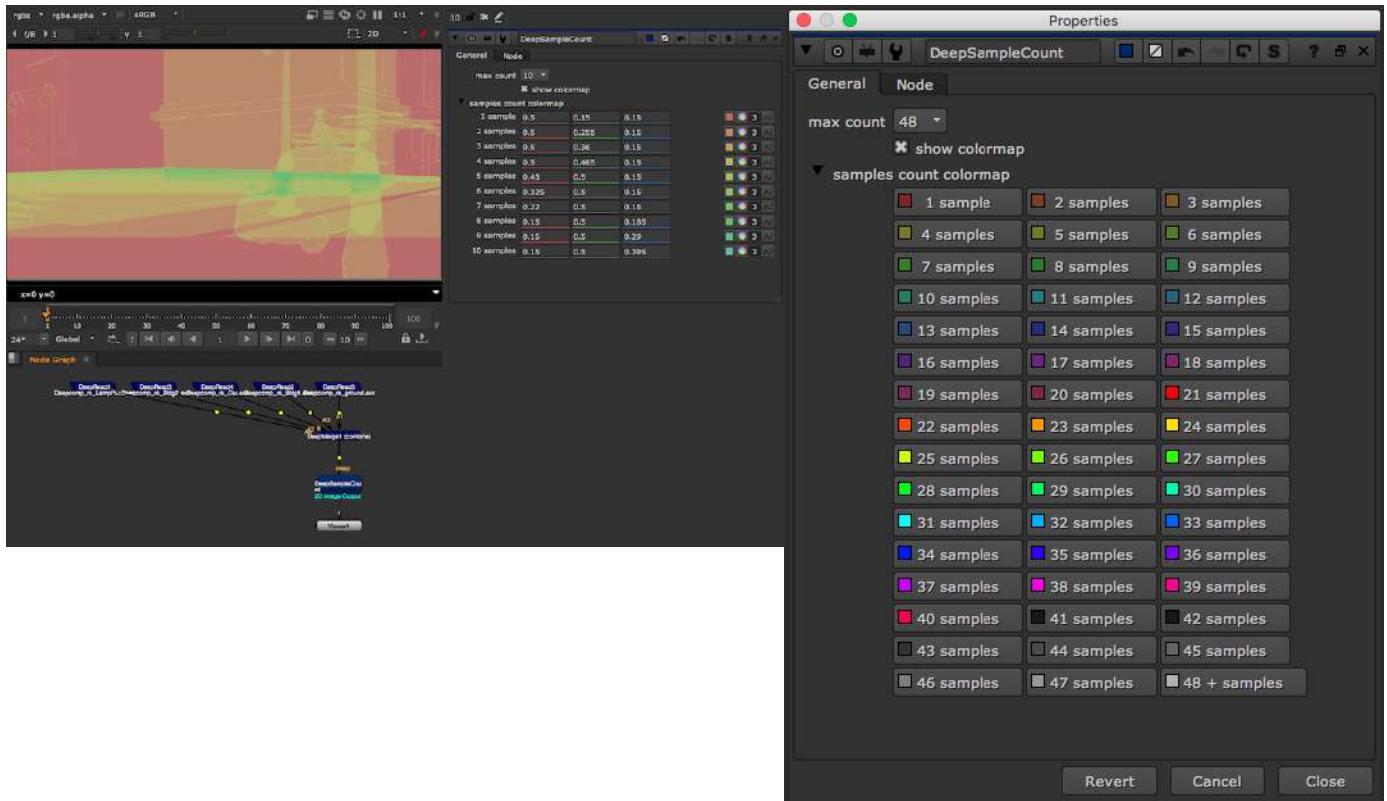
Convert position data to deep.

How to use :

- connect comp tree
- connect camera
- select the channel and what space of your position data
- then downstream will deep

DeepSampleCount MJT

Author: Mark Joey Tang - www.facebook.com/MJTLab



Nukepedia link: http://www.nukepedia.com/gizmos深深/deepsamplecount_v11

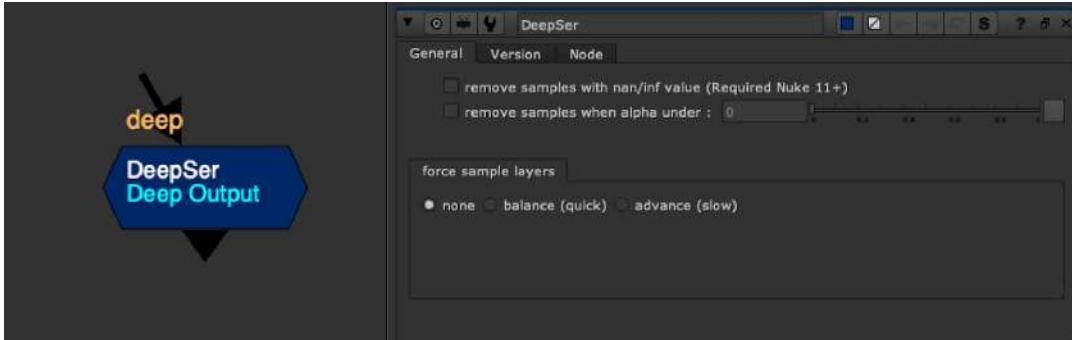
Convert the amount of samples per pixel into colormap visual. Help for troubleshoot, investigate the ground of slow deep tree, or tool development. The setup run by some math & TCL expression, able to run in realtime and fast feedback.

It will created a channel called 'deepSample.count', it stores the total samples per pixel.

*support to detect 48 samples in pixel on v1.2+

DeepSer MJT

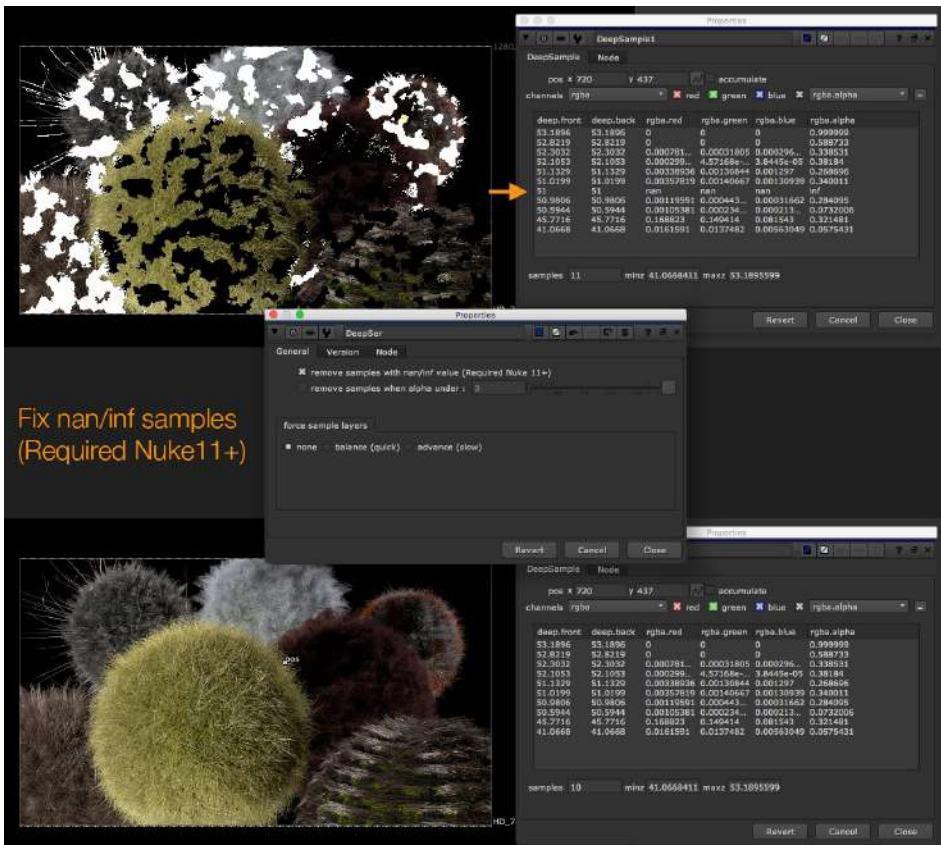
Author: Mark Joey Tang - www.facebook.com/MJTlab



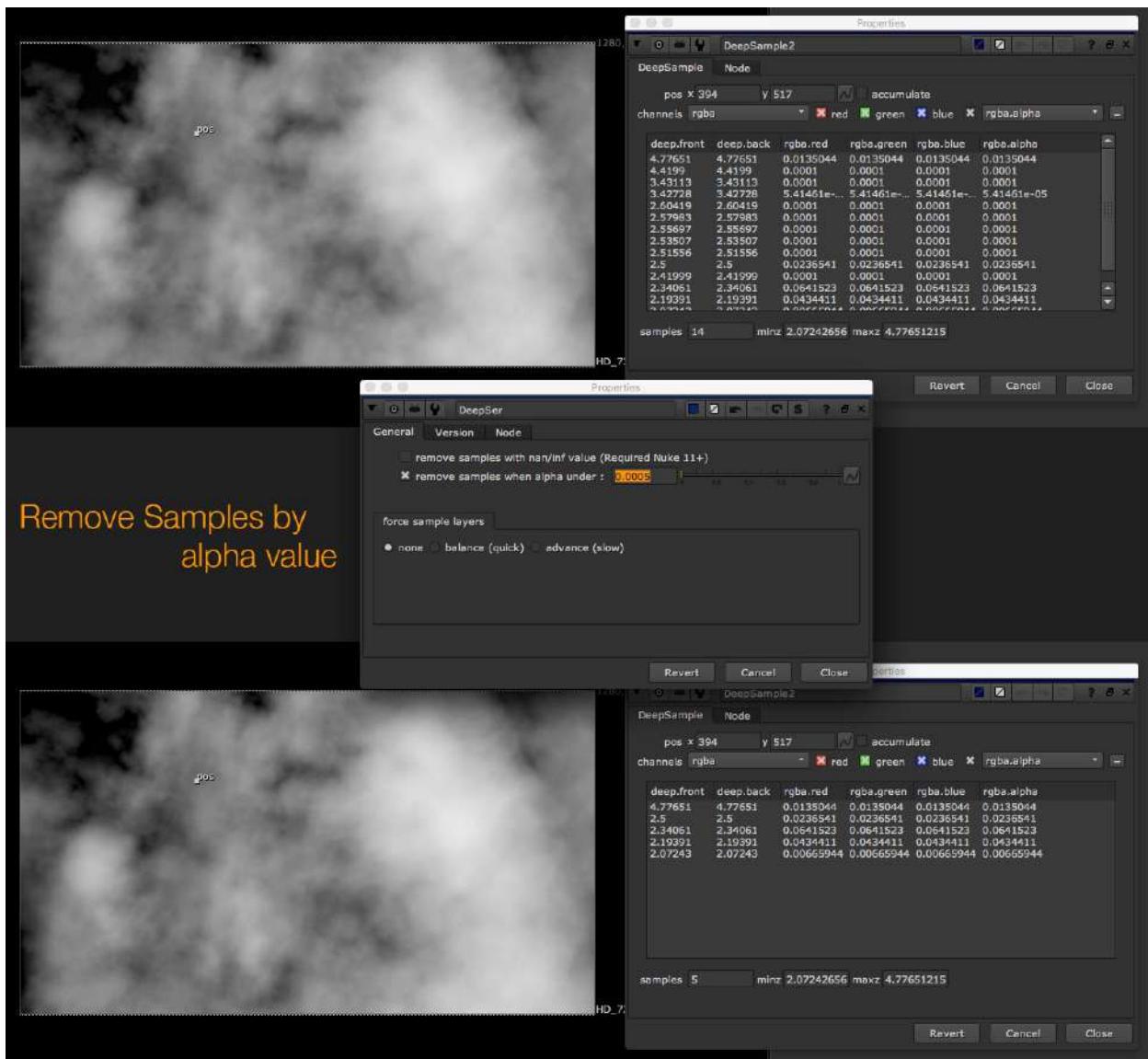
Nukepedia link: <http://www.nukepedia.com/gizmos/deep/deepser>

A tool to manage/remove/optimize deep samples.

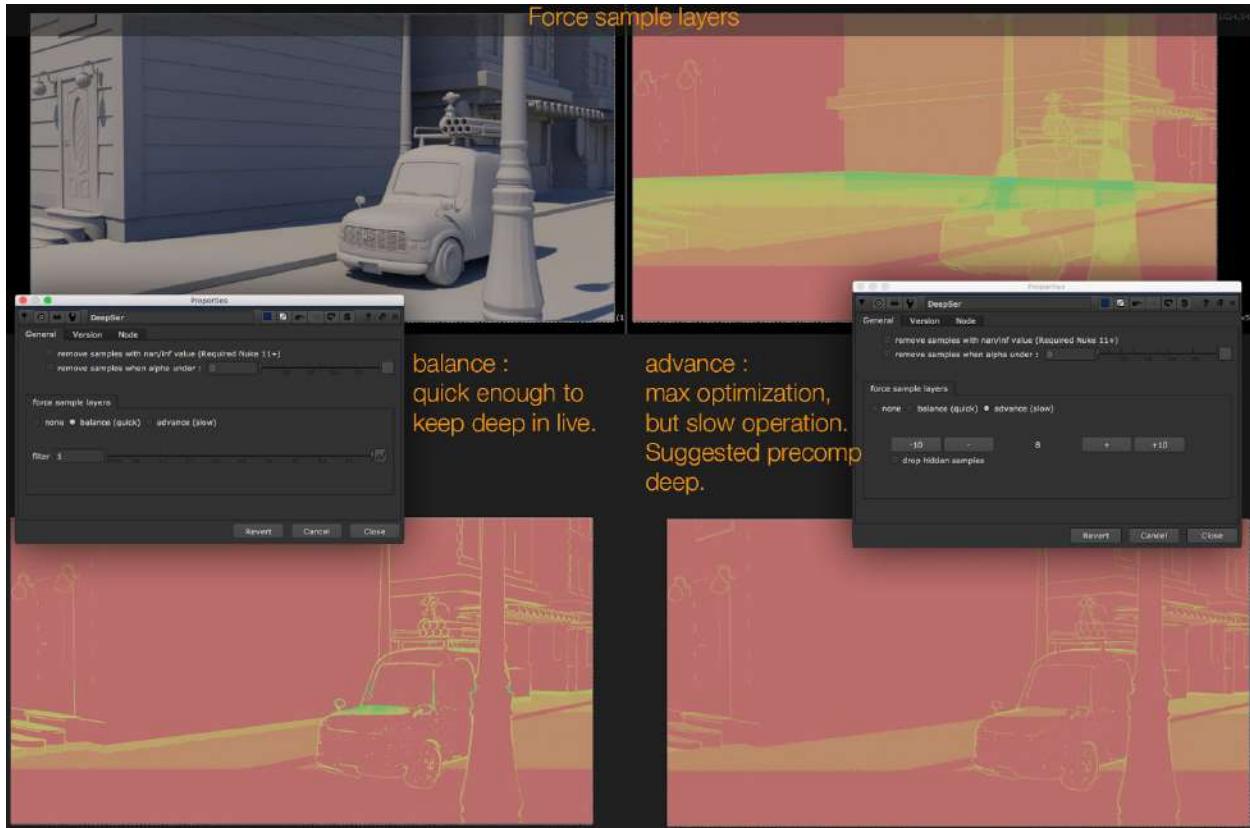
Remove nan/inf samples



Remove samples by alpha value



Remove samples by layer in pixel

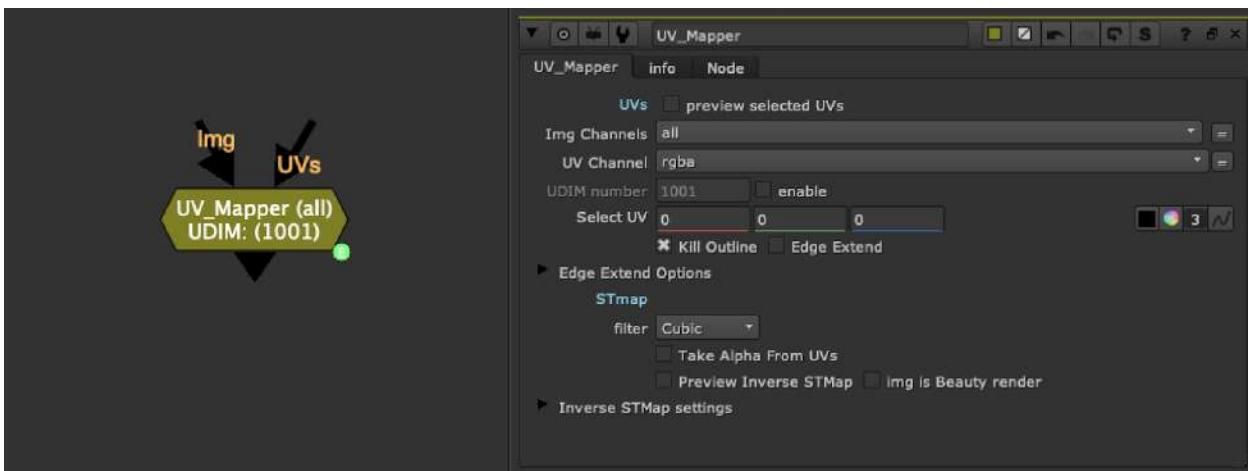


*deep sample colormap generated by DeepSampleCount

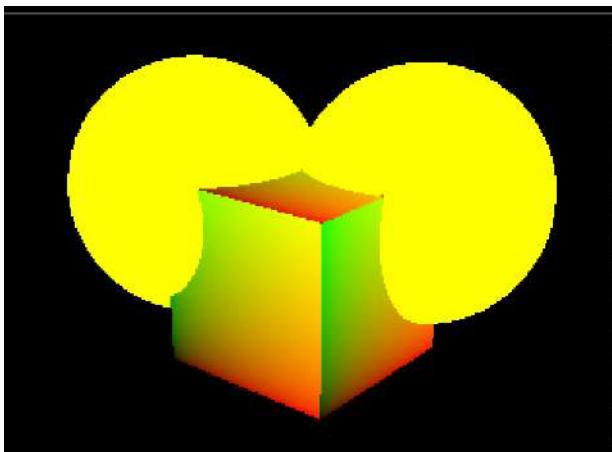
13.) CG

UV_Mapper TL

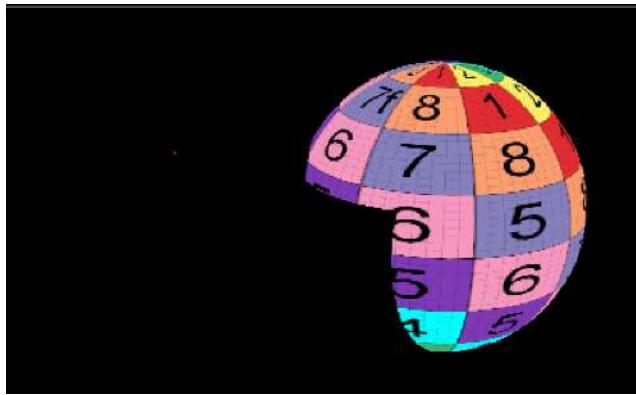
Author: Tony Lyons - www.CompositingMentor.com



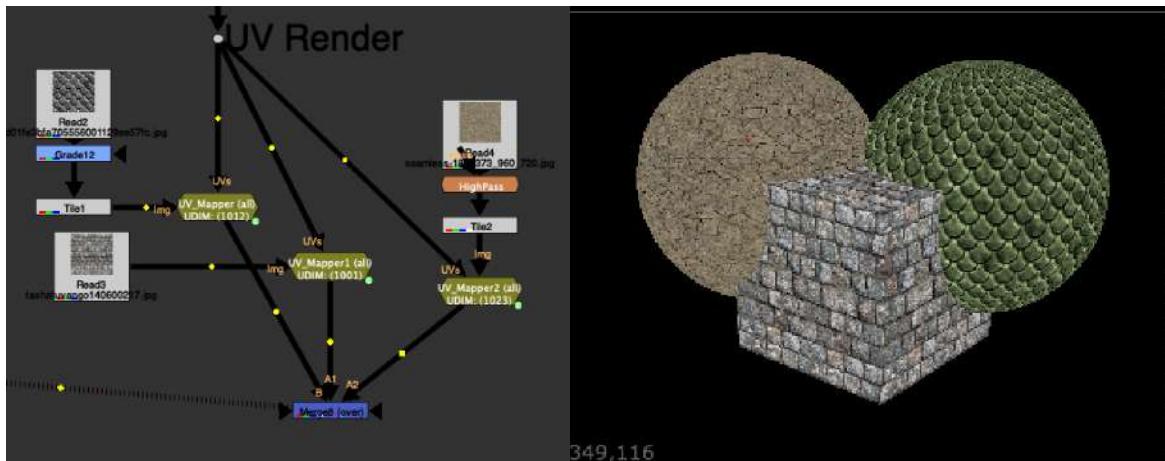
UV_Mapper is for mapping textures on a UV pass with multiple UDIM in the render. Sometimes for creature or other assets, the CG department breaks up the texture passes into multiple UDIMs. This means there are multiple UV sections in the render. Frequently they render this UV pass with each UDIM offset by 1. So you would have 0.5, 1.5, 2.5 for UDIM 1001, 1002, 1003 for example. Here is a preview of what that pass might look like using nuke nodes:



Plug the your UV pass render into the UV_mapper in the UVs input. Use the select UV color picker to isolate which of the UDIM UVs you'd like to use. By default, when where it no image plugged into the img input, I have a UV grid preview as a placeholder for you to see which UDIM you have selected. Selecting around the UV pass you will have this texture jumping from UDIM to UDIM. You should see something like this:



Once you plug an image into the img input, it will replace this placeholder. You can create a couple UV_Mappers and grab a few textures, and with a simply setup you can retexture parts of your image:



Sometimes anti-aliased UV passes can give unwanted results when blending with another UDIM or around the edges when there is filtering. I've included some basic kill outline (erodes in a bit), and edge extend option to help push some UV color value there to reduce artifacting.

Take alpha from UVs: means instead of using the CG render alpha, use the alpha from the input image, that's remapped in UV space.

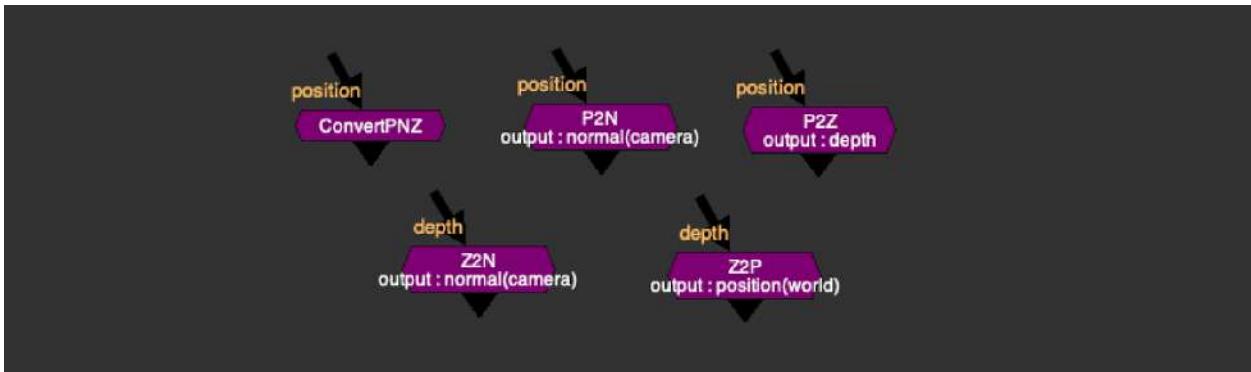
Sometimes it's useful to know where the UV's are corresponding to in UV space, thanks to Luca Mignardi and his inverseSTMap gizmo, I included the option for seeing what part of UVspace the texture occupies when "unwrapped"



This is just a crude visual to help with roto's or paints or alignments. There is of course some situations where you want to know where the beauty render lies in UV space in the selected UDIM. So there is a checkbox called 'Img is beauty render'. First select the UDIM you want, and plug the image into the beauty render and select both preview STMap and Img is beauty and it will isolate the area of the beauty render where the UDIM you have selected is, and unwrap this in UV space, so you can roughly see where your beauty render lies in UV space according to the UDIM you have chosen.

PNZ Suite MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Nukepedia link: <http://www.nukepedia.com/blink/draw/positiontonormal>

Convert 3D data passes. Included P2N (PositionToNormal), P2Z (PositionToDepth), Z2N (DepthToNormal), Z2P (DepthToNormal) and ConvertPNZ (self conversion, such as space convert, fresnel, invert depth).

I have done PositionToNormal during Deep2VP v3.5. I create this standalone version for 2D comp and added more different conversion for 2.5D comp.

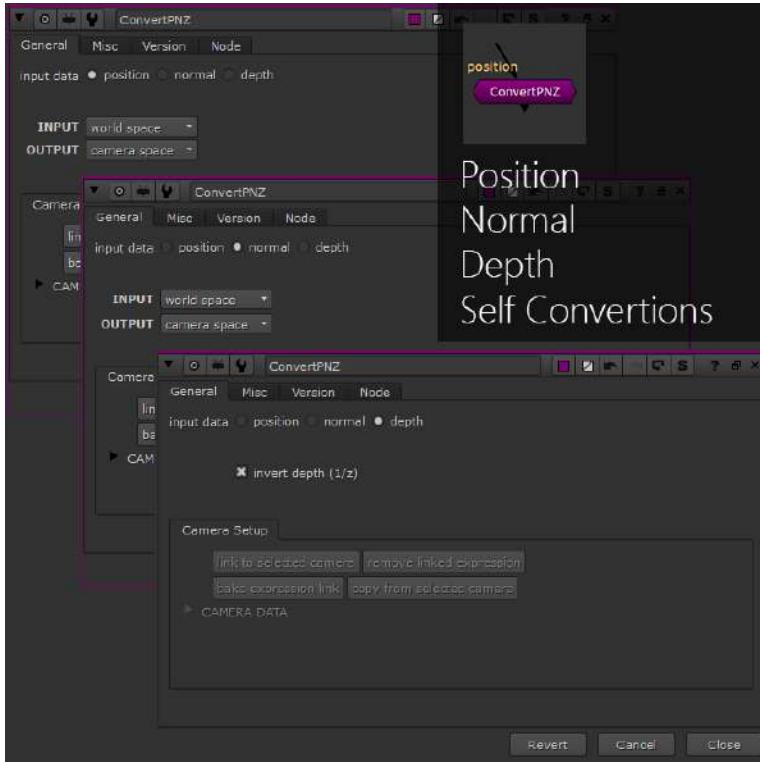
Convert Position/Depth to Normal. It might has a little bit artifact on edges. Try to adjust 'Depth Threshold' for better result. It might not able to get decent result on thin object, such as hair/fur. Depends on the needs.

If you use it for relight, try to light it with this pass first, those artifact might not an issue.

Some space swap require camera data.

ConvertPNZ MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Convert PNZ is for self conversion cases

Select the input data: Position, Normal or Depth

With Position selected:

INPUT: world coordinate space or camera space

OUTPUT: world space or camera space conversion.

With Normal selected:

INPUT: world coordinate space or camera space

OUTPUT: world space or camera space conversion. Or fresnel pass

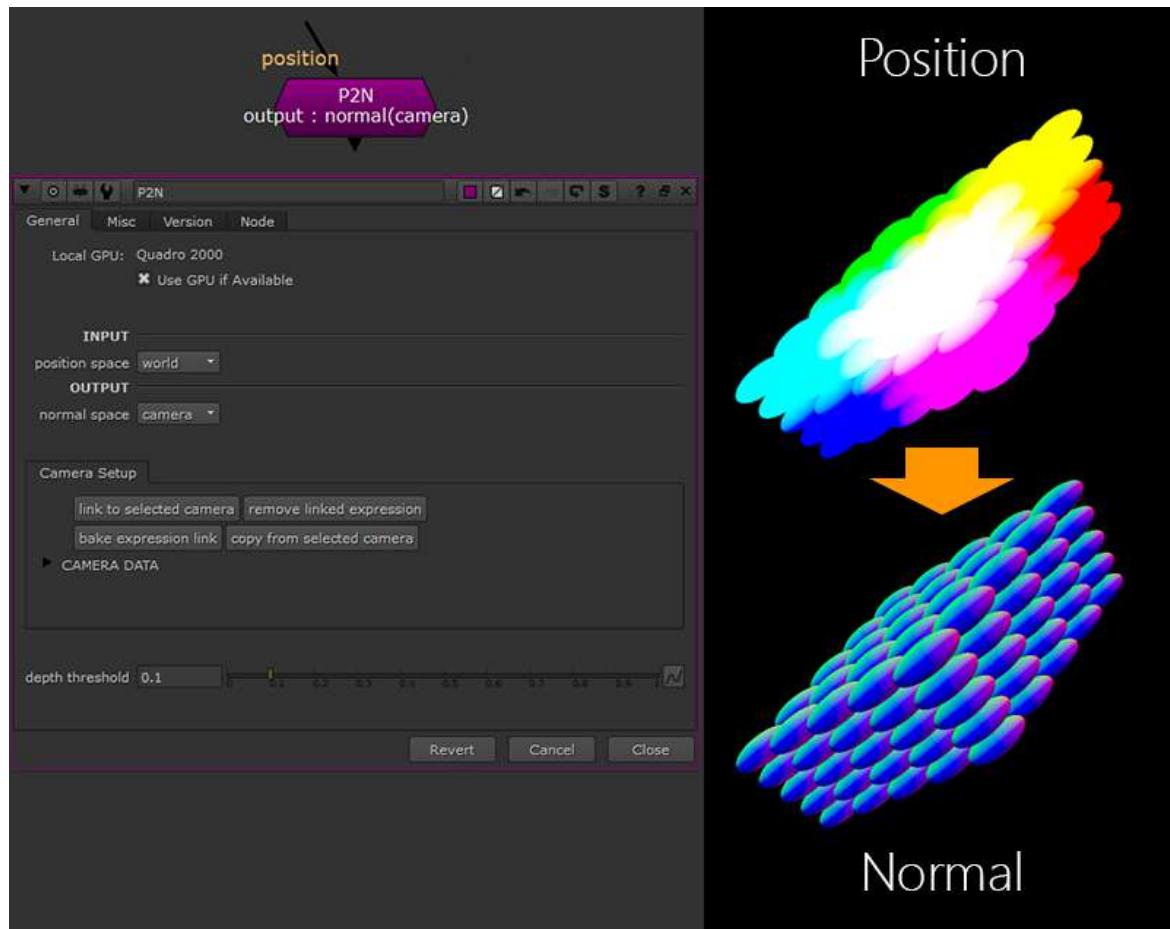
With Depth selected:

You can choose to invert your depth pass with $1/z$ to switch between normalized depth and real units from the renderer

Camera space conversion will require camera data which you can link, bake link, or copy animation from a selected camera. Python buttons to help with linking.

P2N MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Position To Normal

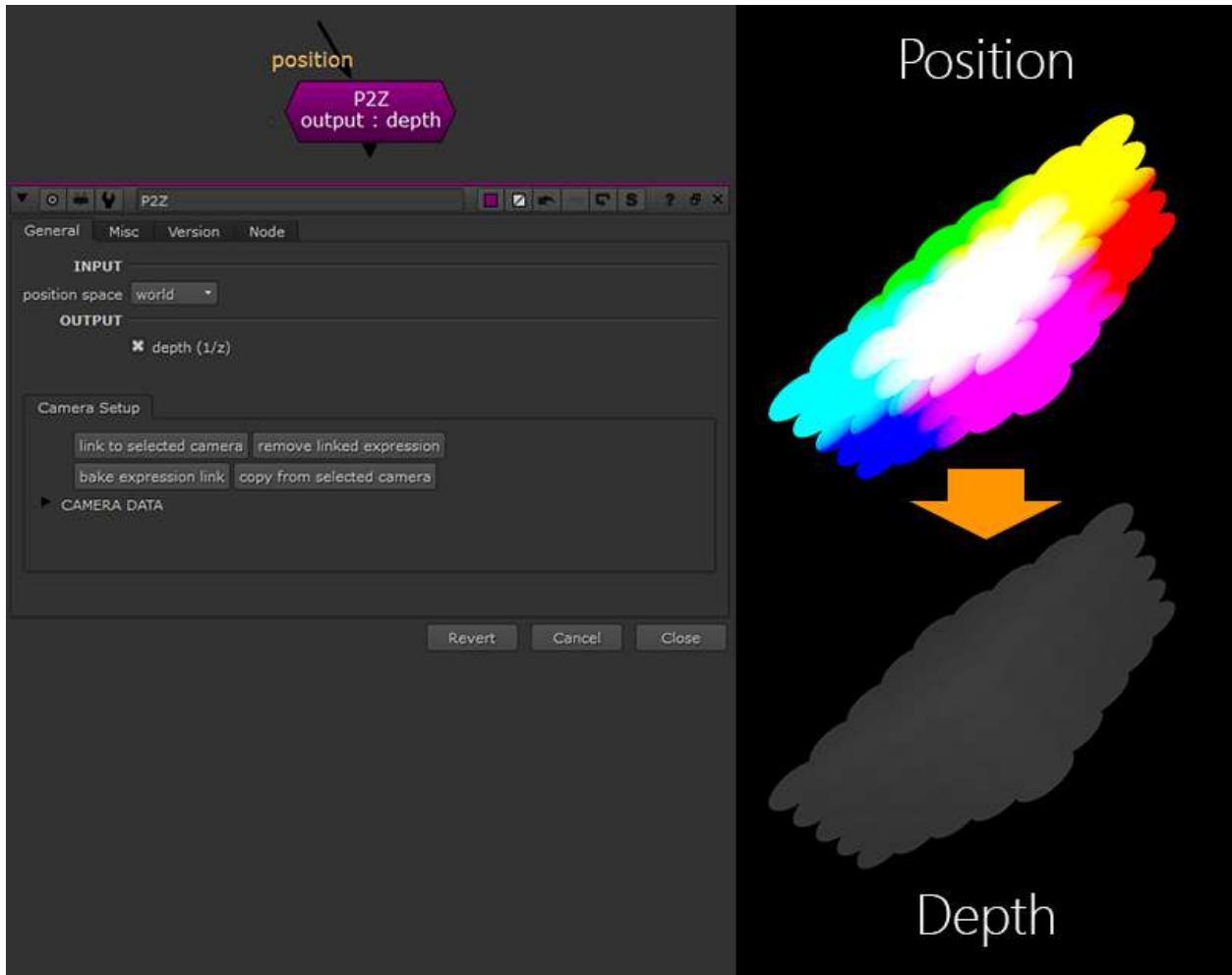
INPUT: World or Camera Space Position

OUTPUT: World or Camera Space Normals

Camera space conversion will require camera data which you can link, bake link, or copy animation from a selected camera. Python buttons to help with linking.

P2Z MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Position To Depth

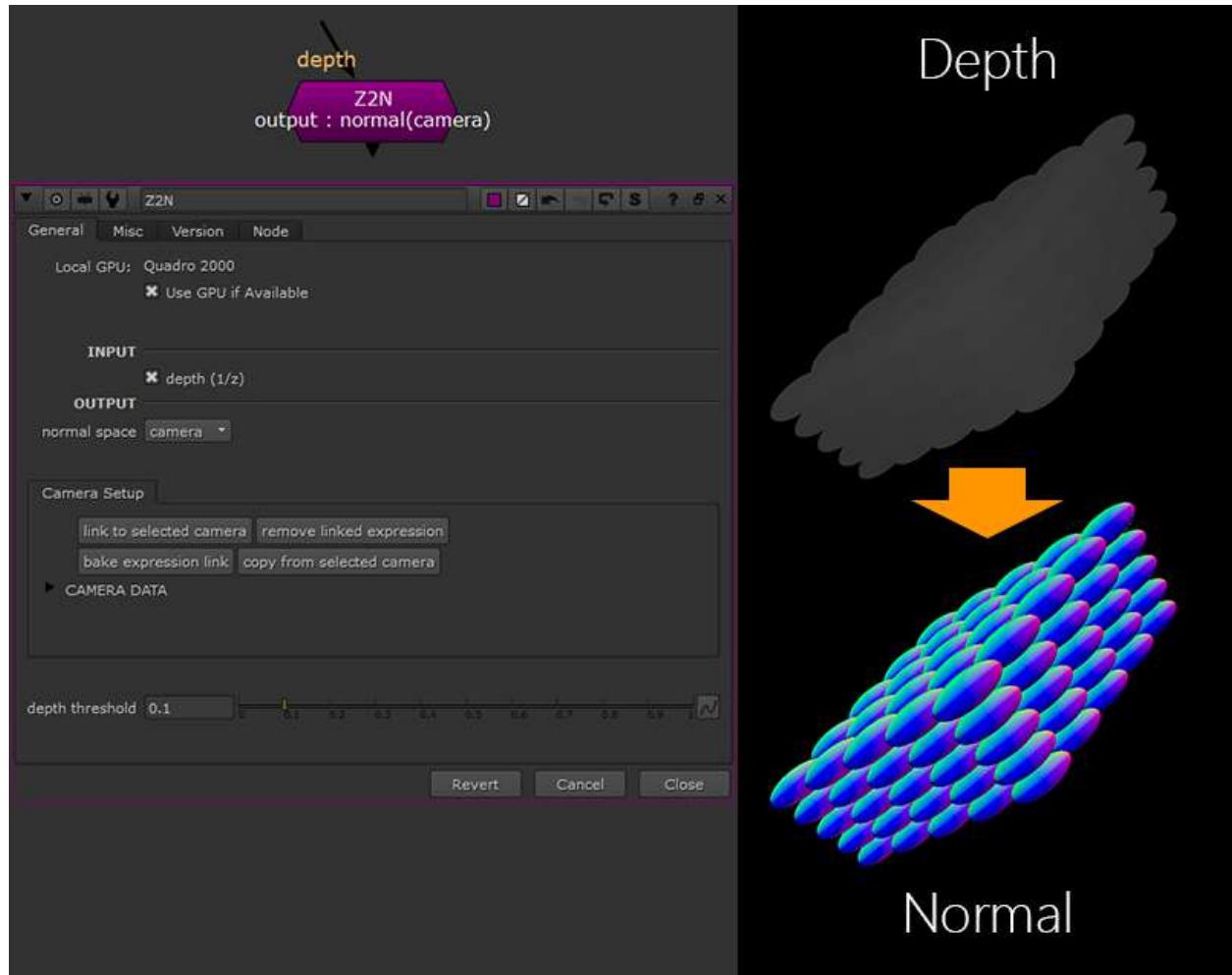
INPUT: World or Camera Space Position

OUTPUT: Real unit depth or normalized depth $1/z$

Camera space conversion will require camera data which you can link, bake link, or copy animation from a selected camera. Python buttons to help with linking.

Z2N MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Depth To Normals

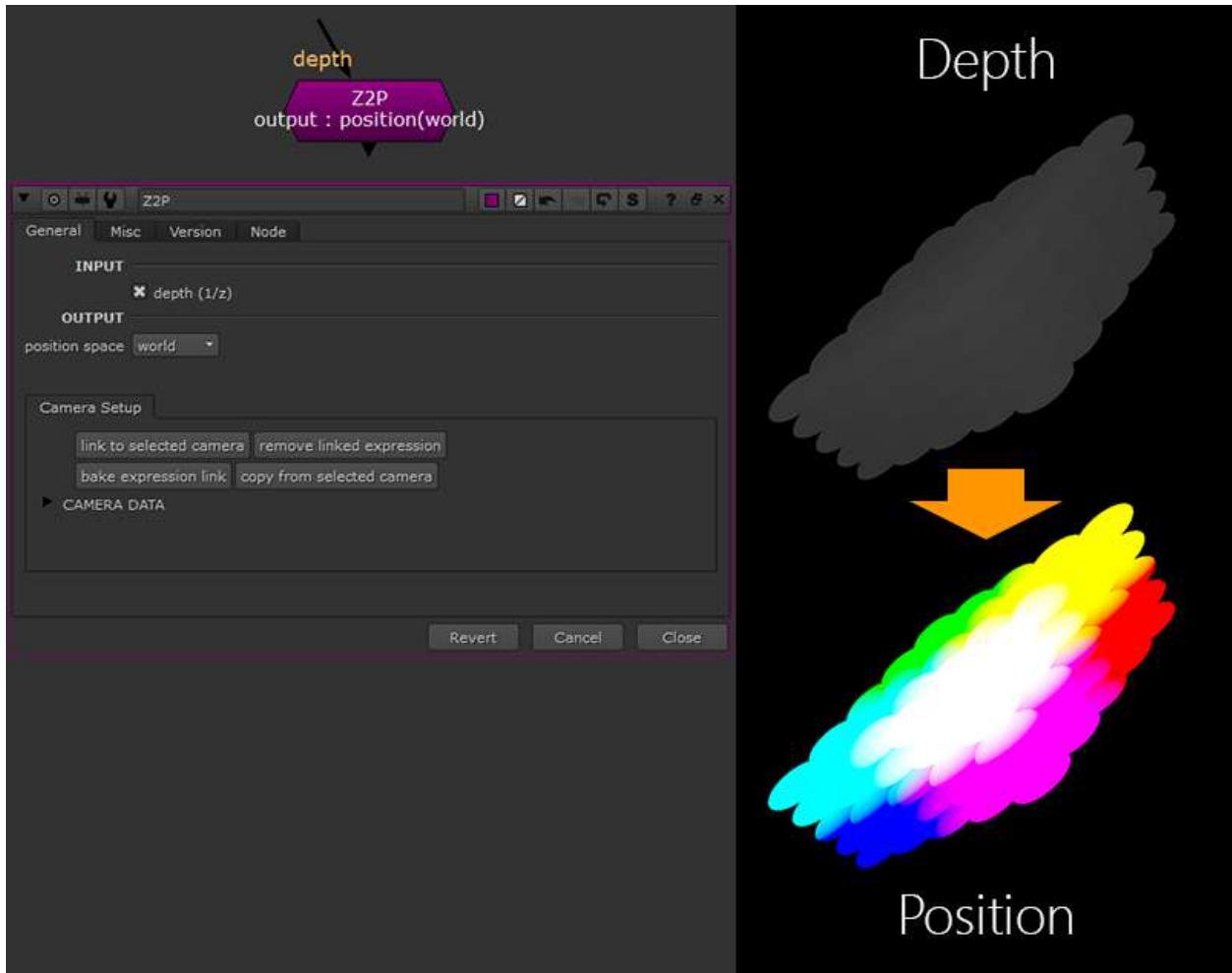
INPUT: Real unit or normalized 1/z depth

OUTPUT: World space or camera space normals

Camera space conversion will require camera data which you can link, bake link, or copy animation from a selected camera. Python buttons to help with linking.

Z2P MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Depth To Position

INPUT: Real unit or normalized 1/z depth

OUTPUT: World space or camera space Position

Camera space conversion will require camera data which you can link, bake link, or copy animation from a selected camera. Python buttons to help with linking.

Pos Toolkit MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Nukepedia Link: <http://www.nukepedia.com/toolsets/3d/pos-toolkit>

Demo/ Tutorial video: <https://vimeo.com/391732849>

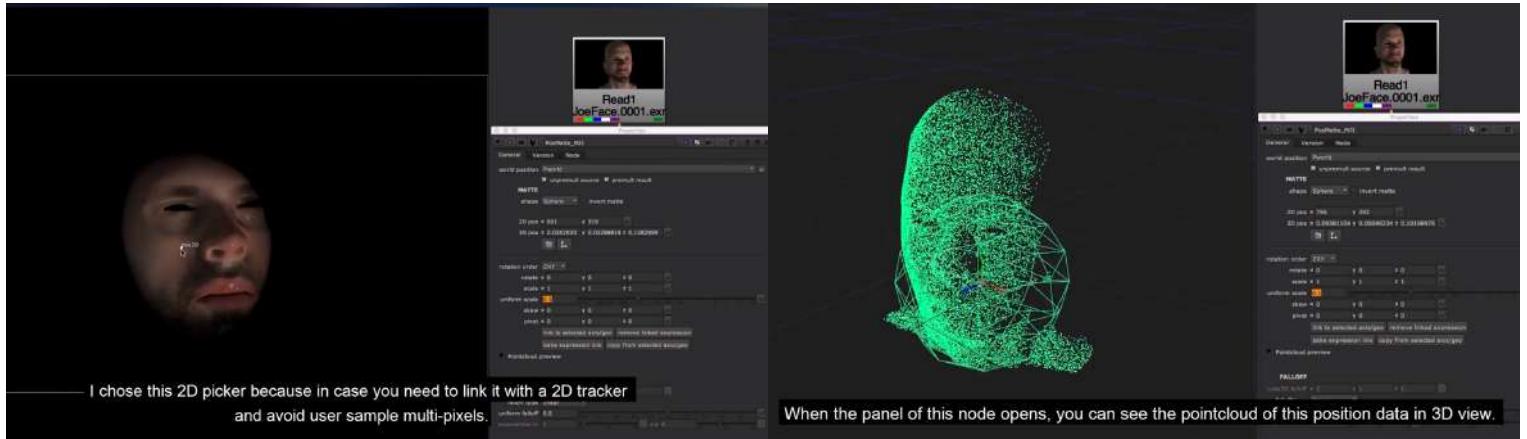
This position toolkit is based on Deep2VP to create and work in 2D compositing instead of Deep. The toolset has PosMatte, PosPattern and PosProjection, the function and workflow is the same as DVPMatte, DVPattern and DVProjection in Deep2VP.

Input of all these nodes must contain world space position data and alpha. If you need to convert position camera space, can use PNZ Suite to do the job.

<http://www.nukepedia.com/blink/draw/positiononnormal>

PosMatte MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Creates 2.5D matte from position data. In this node, it has 3 matte shape and 5 different ways to place the matte.

Choose your World Position channel you are sampling from

Choose either a 2d position (picks from 2d screen space), or a 3D position.

Choose shape type: Sphere, Cube, Cylinder (choice to invert the matte)

Link to a selected axis or copy / bake animations from a selected 3d Axis. Make tweaks to 3d position with these settings

Pointcloud preview:

Choose between no point cloud, raw (full) point cloud, or with matte applied, which the pointcloud will only appear within the 3d shape applied. Additional point size and amount settings in this section.

Falloff type - choose between:

none

linear

smooth

quadratic

cubic

Exponential - additional exp in / out settings

Fall off global controls in this section

PosPattern MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Can create 7 different patterns through position data.
fBM, turbulence, noise, random, stripes, ripple, and rays

Each have different parameters exposed when selected for adjustments
Snap menu available to snap to points and vertices

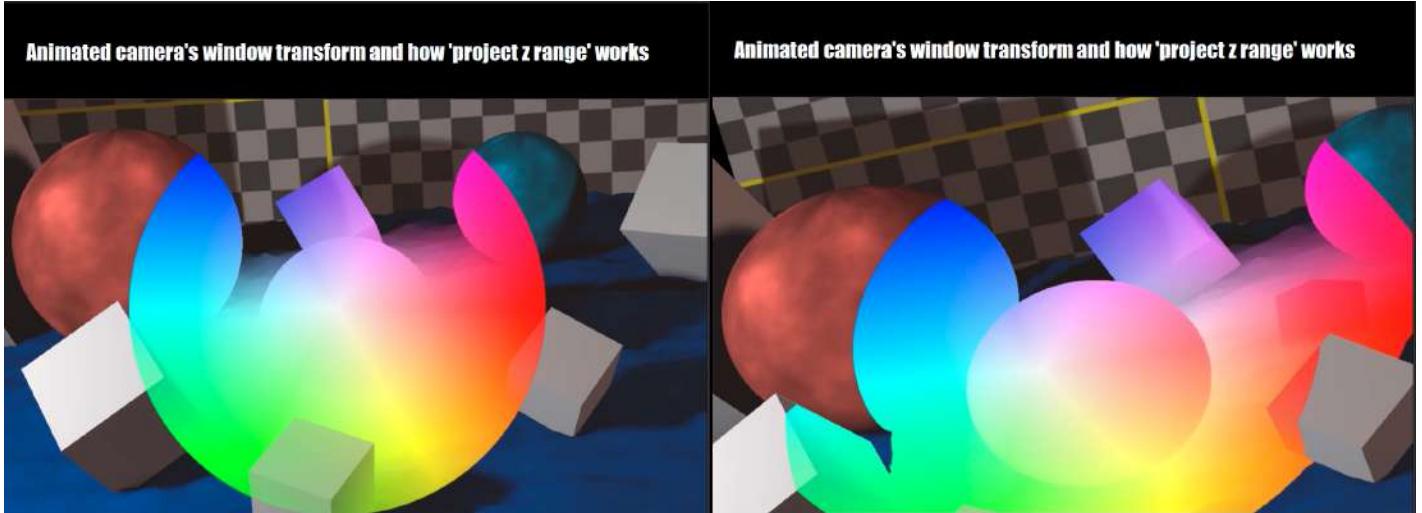
3D translation/scale options available

Pointcloud preview:

Choose between no point cloud, raw (full) point cloud, or with pattern applied. Additional point size and amount settings in this section.

PosProjection MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Can do camera projection with position data. This node requires a camera, texture is optional (it is because it also can output UV maps). Camera can be a custom one, but usually for CG patch fix will use the shot camera to avoid line up perspective manually. It's because the node will calculate FOV of the camera for projection.

Select your WorldPosition channel

Optional freeze frame settings

Project Z range, similar to deepSoft Clip, you can set up where the texture begins and ends in Z space from the camera, along with a Z fade (similar to keyer node)

Link, bake, or copy camera settings from a selected camera

Pointcloud preview:

Choose between no point cloud, raw (full) point cloud, or with project image(slow), Additional point size and amount settings in this section.

Outputs:

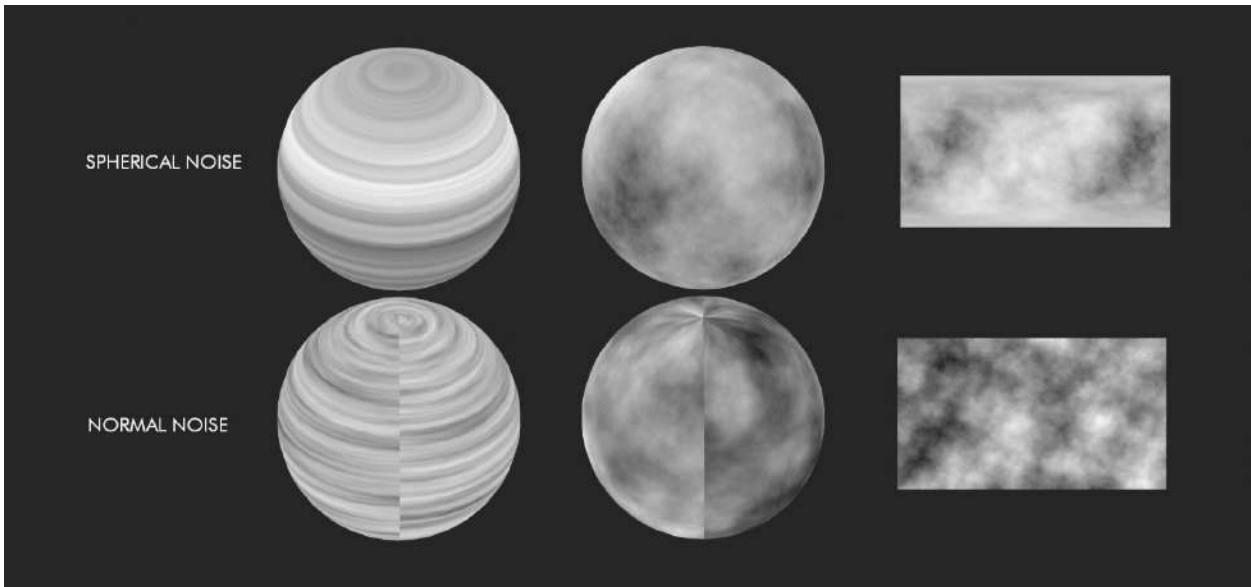
wrapped texture + source: which is a projected texture composite with the input.

wrapped texture: which is only projected texture

uv: which is a uv map, can use this with STMap in downstream. STMap settings available

Noise3D SPIN

Author: SPIN FX - Erwan Leroy - <http://erwanleroy.com/blog/>



Spin gizmos on github: https://github.com/SpinVFX/spin_nuke_gizmos

Nukepedia link: http://www.nukepedia.com/gizmos/other/spin_nuke_gizmos-1

Spin tools 2.0: <http://erwanleroy.com/spin-vfx-nuke-gizmos-2-0-released/>

Vimeo Demo @ 00:50 : <https://vimeo.com/381270956/f7399b6e1d>

Generate Noise in 3D space based on Position passes. It includes pre-made Position passes for some 3D primitives, or can use a custom Position pass. Uses a 4D noise internally so that the 4th dimension can be used to add a 'boiling' effect.

Choose between fBm and Turbulence.

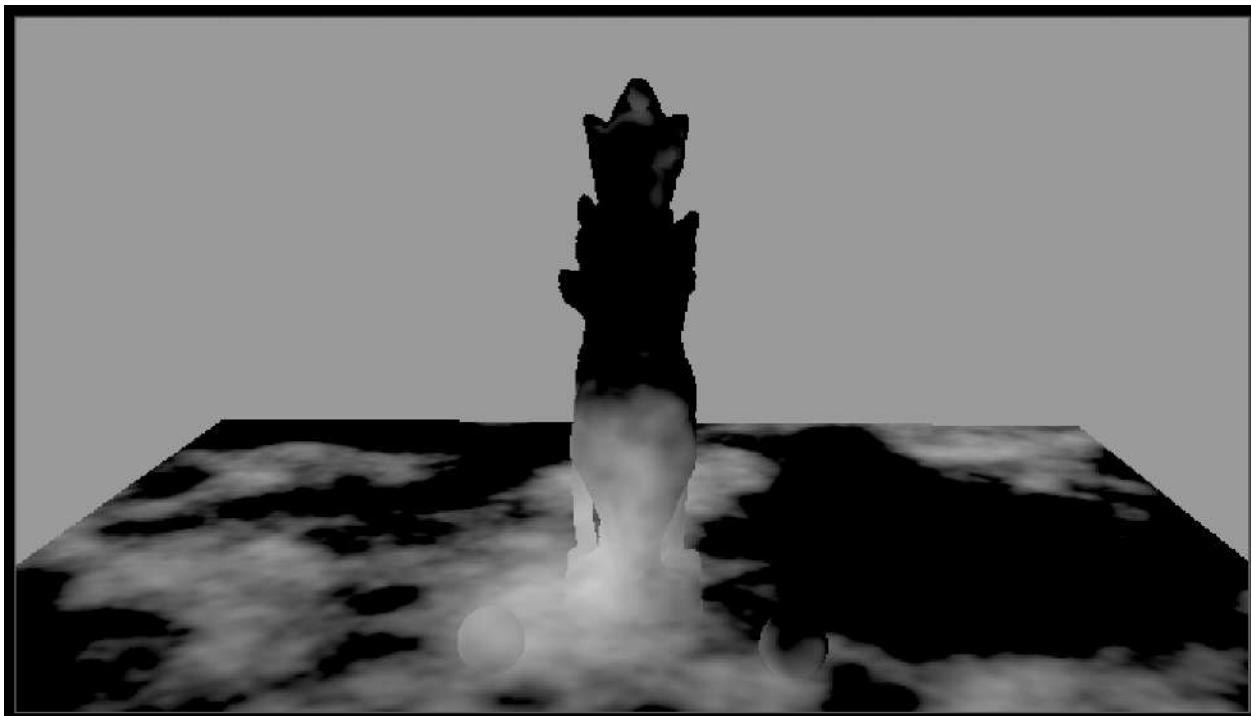
Multiple mapping options:

Card (Flat), Spherical, Cylindrical (Tile X), Cylinder 4D (Tile X and Y), Custom P Input

The Custom P Input lets you input the shape of your choice. Using a pWorld pass as custom P will generate a 3D noise as seen from the same camera as used to render the pworld. Ensure an Alpha channel is present for Custom P.

Noise4D MHD

Author: Mads Hagbarth Damsbo - <https://hagbarth.net/blog/>



Nukepedia link: <http://www.nukepedia.com/blink/draw/4d-noise>

This is a port of the 4D simplex noise found at
<https://github.com/Draradech/csworldgen/blob/master/simplexnoise.cpp>

Blink script - 4D Noise Generator (Based on image values)

It uses the image values of the input image to generate the noise. It is not fast, but it does the job quite well.

Red, Green, Blue image (Pworld or Pref or vectors) X, Y, Z position

Alpha Channel is the evolution (4th dimension), change alpha to change the seed/evolution.

Relight_Simple SPIN

Author: SPIN FX - Erwan Leroy - <http://erwanleroy.com/blog/>



Spin gizmos on github: https://github.com/SpinVFX/spin_nuke_gizmos

Nukepedia link: http://www.nukepedia.com/gizmos/other/spin_nuke_gizmos-1

Spin tools 2.0: <http://erwanleroy.com/spin-vfx-nuke-gizmos-2-0-released/>

Vimeo Demo @ 03:28 : <https://vimeo.com/381270956/f7399b6e1d>

Simplified relight node. Only requires a normal map to get started.

This node will not reproduce accurate lighting, as it does not take into account the actual 3D world space, but instead considers the image in its own local space.

Plug in Normals input to a Normals Pass that is shuffled into rgba

Choose a 2d position to orient the relighting angle

Light Z attempts to set an distance of the picker, forward or backwards in 'Z space'

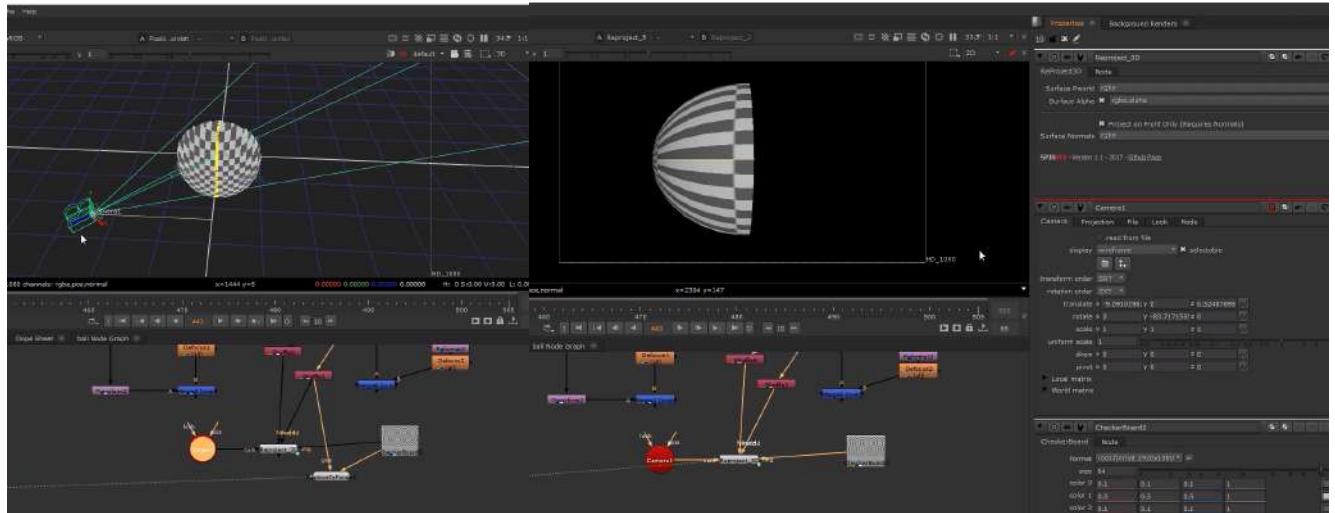
Choose light color

Option to use an albedo pass (aka a texture or color pass) as an input

Specularity and shininess options

Reproject3D SPIN

Author: SPIN FX - Erwan Leroy - <http://erwanleroy.com/blog/>



Spin gizmos on github: https://github.com/SpinVFX/spin_nuke_gizmos
Nukepedia link: http://www.nukpedia.com/gizmos/other/spin_nuke_gizmos-1
Spin tools 2.0: <http://erwanleroy.com/spin-vfx-nuke-gizmos-2-0-released/>
Vimeo Demo @ 04:45 : <https://vimeo.com/381270956/f7399b6e1d>

This gizmo does camera projection using a render point position pass (in world space) and a 3D camera to remap all the layers in the input image.

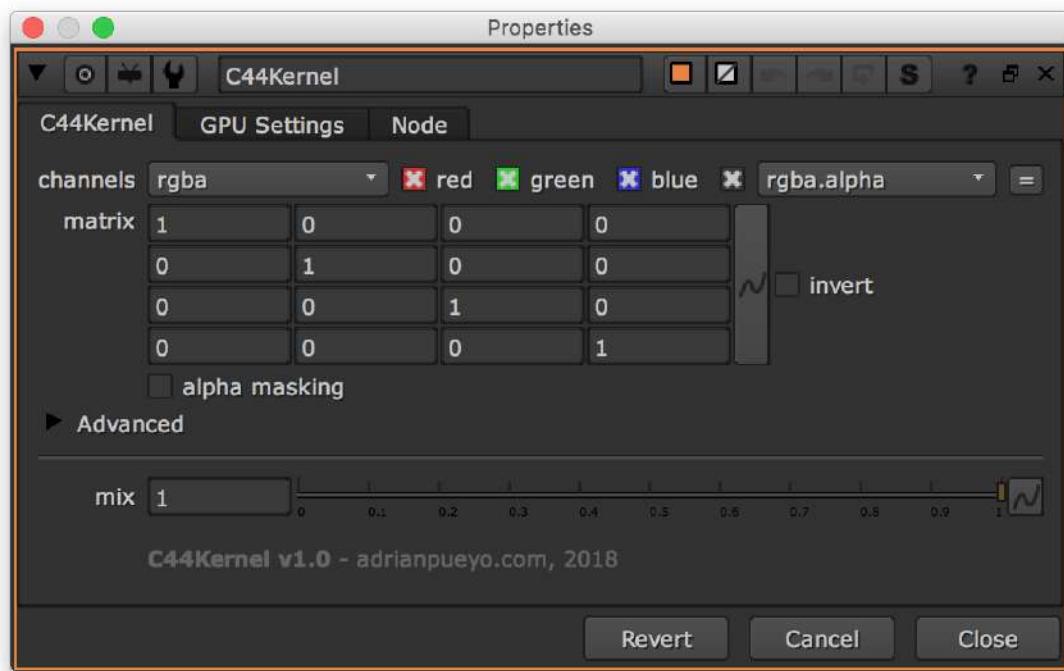
NOTE: The projection works best with unmatted input images or at least unpremuted matting with some coverage, then all masking occurs within the gizmo. It can mask the projected image in the following ways:

Input alpha from the 3D render.

Projecting on surfaces facing camera, using normal (N) in world space.

C44Kernel AP

Author: Adrian Pueyo - <http://www.adrianpueyo.com/>



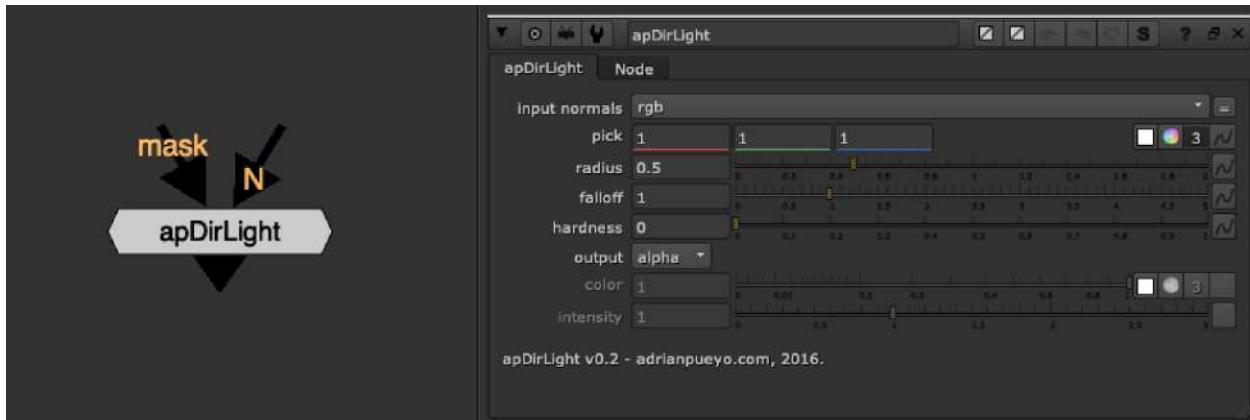
Nukepedia link: <http://www.nukepedia.com/blink/colour/c44kernel>

Multiply the rgb or rgba colors by an arbitrary 4x4 Matrix. Useful for transforming vector passes like Position or Normals. You can also plug in an Axis or Camera node into the axis input, to apply its transformations.

C44Kernel is a simpler, Blinkscript-based alternative to the C44Matrix node by Ivan Busquets, which is incredibly useful but has the compatibility limitations of a plugin.

apDirLight AP

Author: Adrian Pueyo - www.adrianpueyo.com



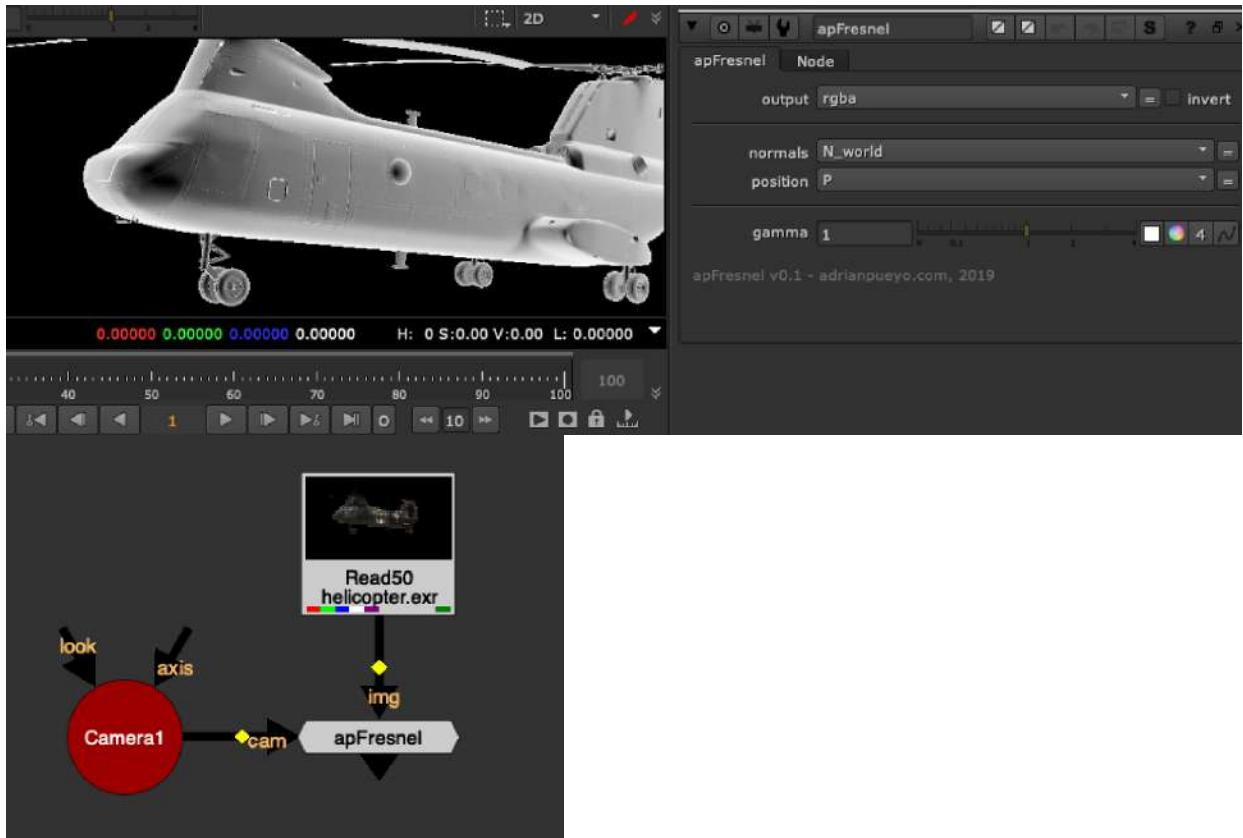
Simulate a simple directional/infinite light through a normal pass, by picking a Normals pass color and then tweaking the radius, falloff and hardness of the light.

Demo Video: <https://vimeo.com/157309298>



apFresnel AP

Author: Adrian Pueyo - www.adrianpueyo.com



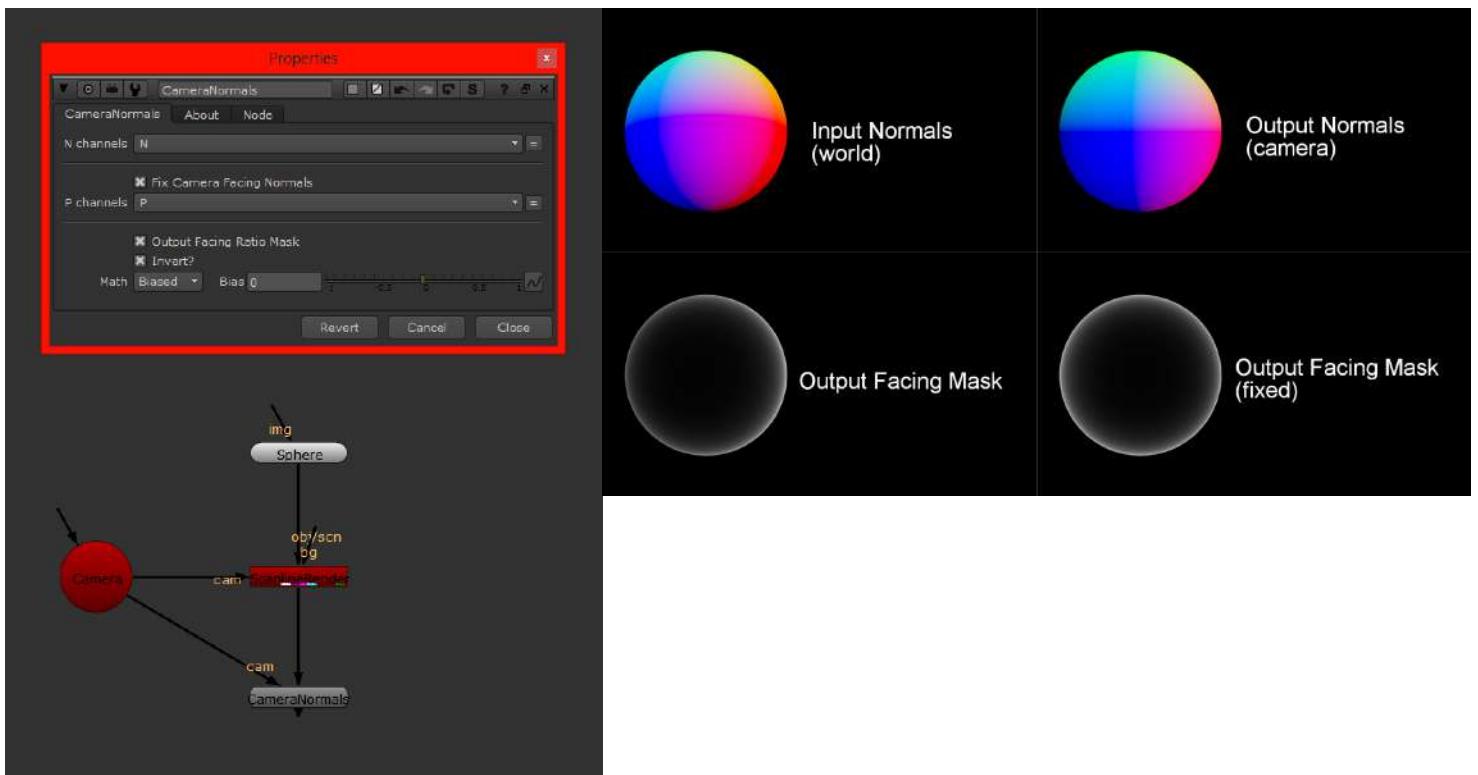
Convert your Normals Worldspace pass into a Camera Space Fresnel pass.

Plug in your render camera, pick the normals and position channels from the input.

Can adjust the gamma of the Fresnel (Facing ratio)

CameraNormals NKPD

Author: Nikolai Wüstemann



Nukepedia link: <http://www.nukepedia.com/gizmos/channel/cameranormals>

The Gizmo lets you convert your CG worldspace normals to camera space normals.

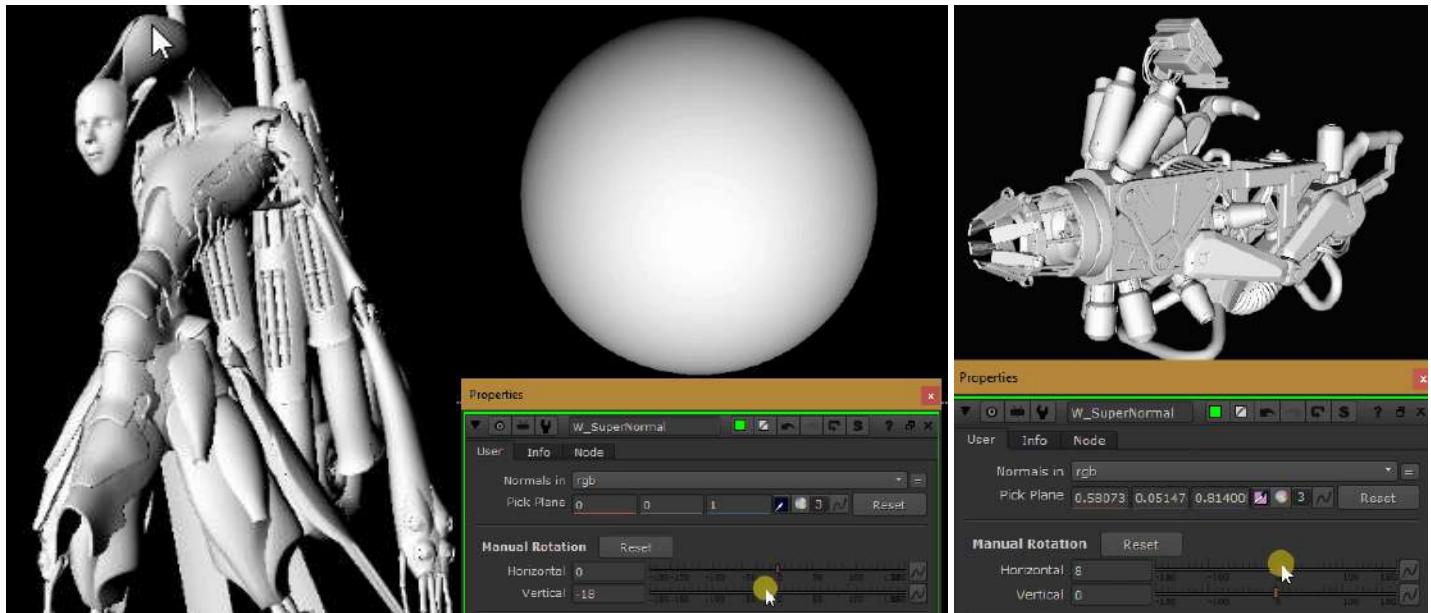
As the blue component of camera space normals is often used as a facing ratio, you also have the option to directly output the mask.

In addition you have some control about the facing mask falloff (Simple or Biased), that is driven by an exponential expression.

Modified with Adrian Pueyo's Dummy Camera gizmo

NormalsRotate NKPD

Author: Wes Heo - gizmo named W_SuperNormal, renamed for clarity



Nukepedia link: http://www.nukepedia.com/gizmos/3d/w_supernormal

There are two ways to select the surface angle:

1. P_Matte has been the inspiration for this tool to do with normals what it does with position data, and as such there is the colorpick method like P_Matte
2. Horizontal and Vertical Rotations sliders

Instructions:

1. Select the layer containing normals in the dropdown menu.
2. Enable color picker and pick the point where you want the matte to be white.
(I look at the alpha output, hold ctrl+alt and "glide" over the surfaces.)
3. You can also manually rotate the matte. When you colorpick a new point, it is recommended that you reset the manual rotation values to 0.

Matte output:

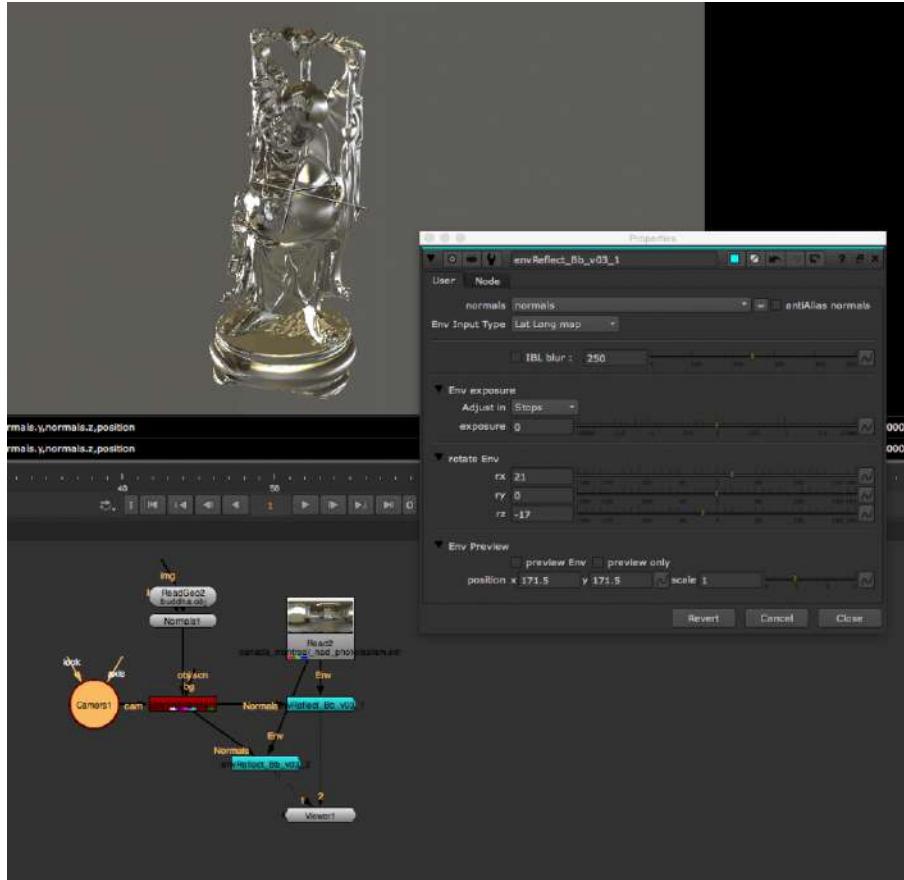
invert output, mask by alpha, unpremult

Exponential falloff controls and post grade.

Mask and mix controls added.

EnvReflect_BB NKPD

Author: Bastian Brenot - <http://www.bastienbrenot.com/nuke-tools/>



Nukepedia link: http://www.nukepedia.com/gizmos/other/envreflect_bb_v03

Fake Environment reflections on normals pass

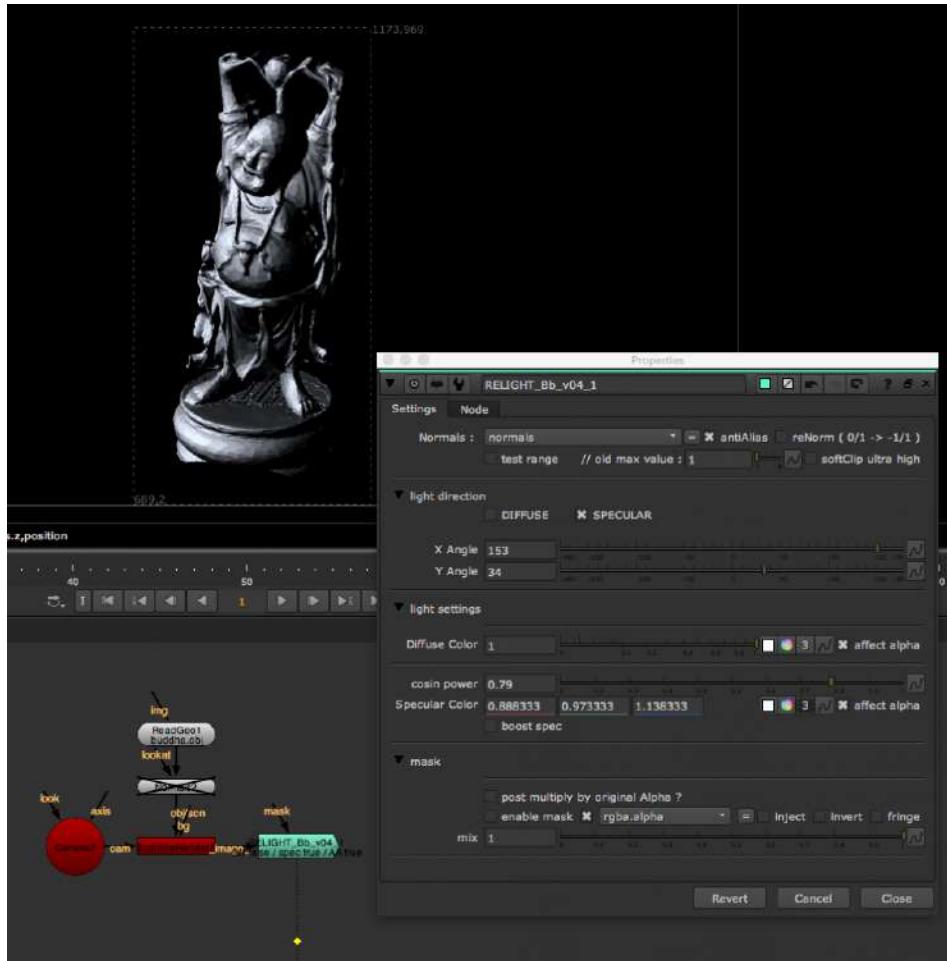
Feed it with a normals pass through the 'image' input, and select the right channel for it.
(works better with cameraSpace Normals)

Choose between: Mirror Ball, Latlong Map, Cube, Angular map 180, angular map 360, sphere and 180 degree fish eye lens

- Option to Blur the IBL
- Antialias normals to get rid of flickering
- option to rotate the environment map in x,y,z
- Use Env Preview to preview the Envball with transformations

Relight_BB NKPD

Author: Bastian Brenot - <http://www.bastienbrenot.com/nuke-tools/>



Nukepedia link: http://www.nukepedia.com/gizmos/other/relight_bb_v04

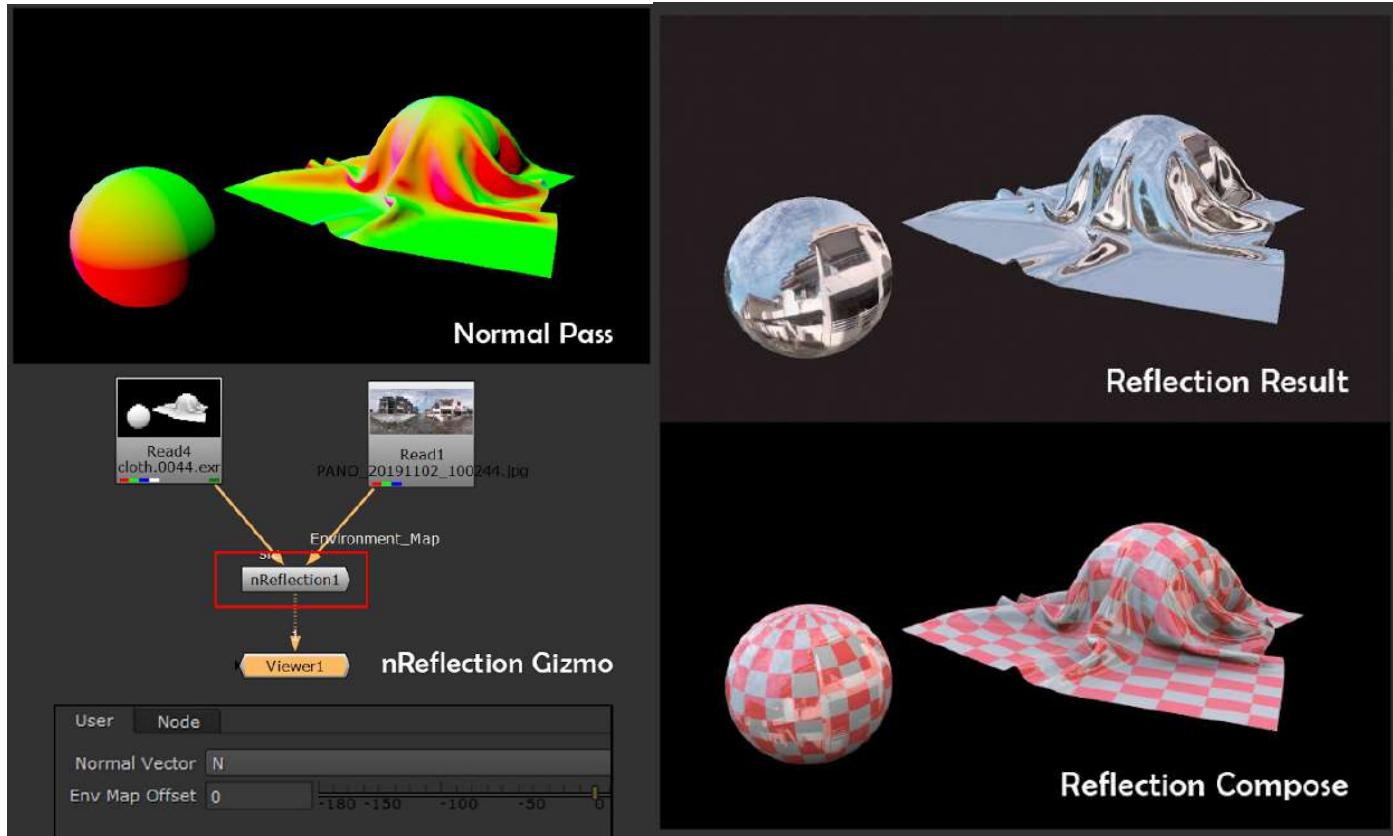
A 2D based Relighting tool to help relight 3d renders with help of a normals pass.
All you need is to feed it with a normals pass through the Normals input

Use the 'mask' input to isolate a specific object in your scene.

- Antialias normals to get rid of flickering
- use ReNorm if your normals are coded from 0 to 1 instead of -1 to 1

N_Reflection KNPD

Author: Chetal Gazdar



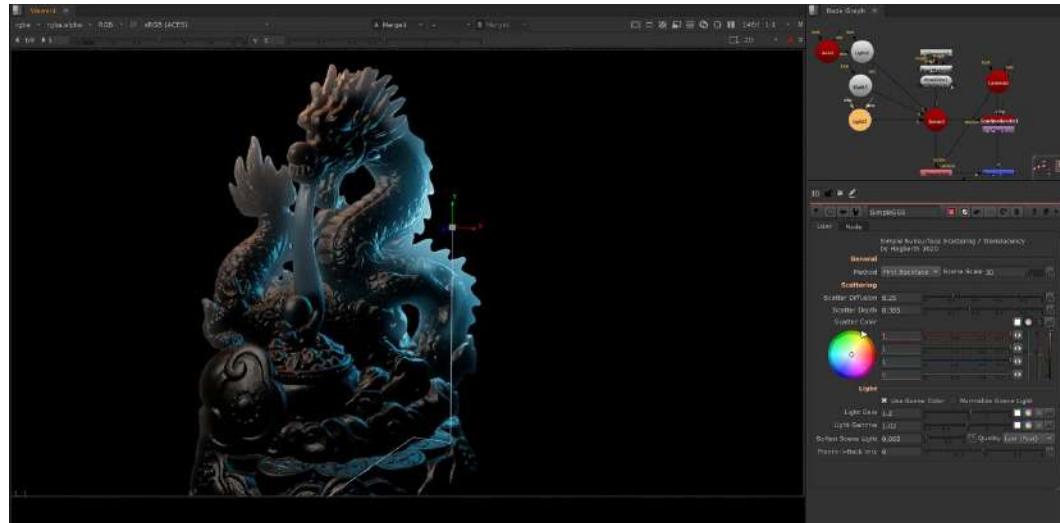
Nukepedia link: <https://www.nukepedia.com/gizmos/transform/nreflection>

Through this Gizmo you can create fake reflection in post using only Normal pass.

Plug in your Normals pass, select normals channel, and rotate the environment map. LatLong HDRI's work best, but any image will work

SimpleSSS MHD

Author: Mads Hagbarth Damsbo - <https://hagbarth.net/blog/>



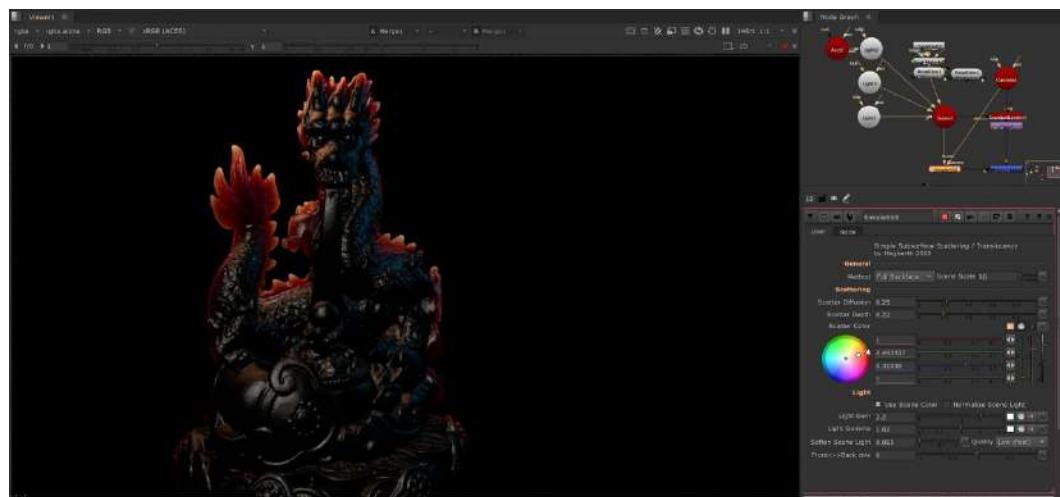
Nukepedia link: <http://www.nukepedia.com/gizmos/3d/simple-subsurface-scattering>

The Comp Lair episode where we discussed the effect: www.youtube.com/watch?v=qW7FrnTl8Jo

Original article: www.hagbarth.net/translucency-shader-in-nuke/

Demo Video: <https://vimeo.com/462218833>

SimpleSSS tool lets you create a simple, stylized faux sss/translucency effect on your nuke Geo. Simply apply this node, input your model and a camera, and plus the output of the node over your normal scanline render.





aPmatte AP

Author: Adrian Pueyo - www.adrianpueyo.com



Nukepedia link: <http://www.nukepedia.com/blink/keyer/apmatte>

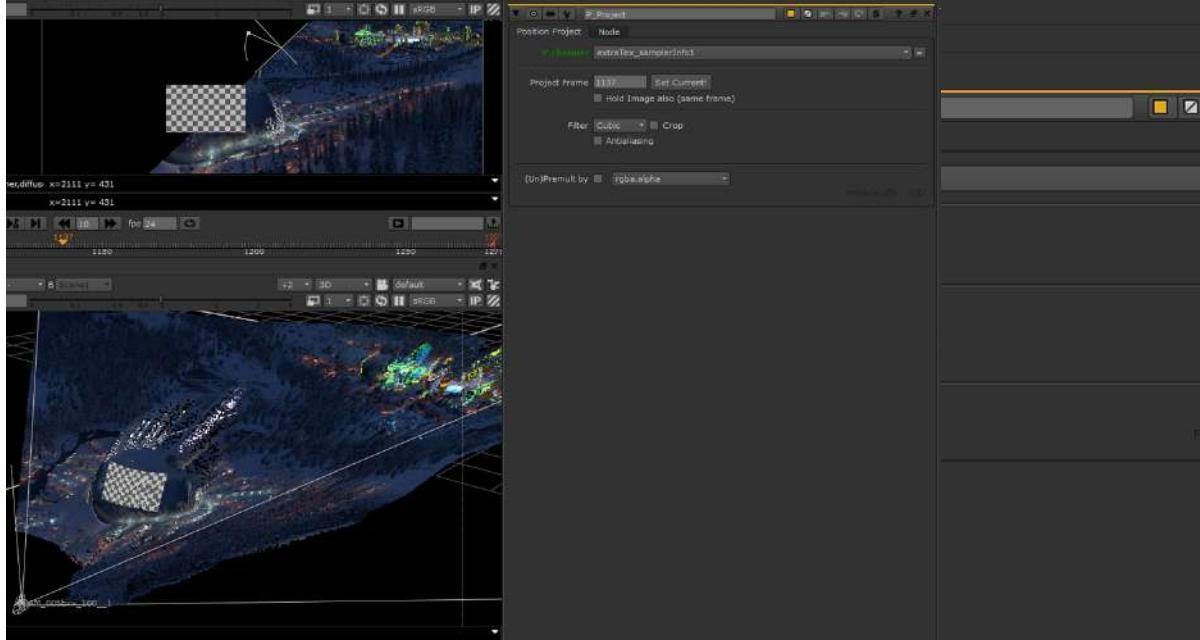
Tutorial/Overview video: <https://vimeo.com/adrianpueyo/apmattev2>

Blinkscript-based Nuke gizmo for generating mattes or 4D noise from a Position pass.

v2.0 includes some new shapes and modes, a 3D-space view of the matte and point cloud, and accepts an Axis or Camera input. Also, lets you preview the p-matte over an image input and merge the alpha with the incoming one through different operations.

P_Project NKPD

Author: Franklin Toussaint - <http://franklinvfx.com/tools-2/>



Website download: http://franklinvfx.com/pos_project/

Github Link: https://github.com/franklinvfx/Position_Project-Tool-for-Nuke

Camera projection using the render of “World Position” and camera.

The most important advantage of this one is that it's a very light tool (you don't need to import in nuke a heavy mesh).

So it's fast and easy to use, but still has to have the position pass and the 3D camera

Tool details:

P Channel: To select the Position Pass channel.

Project Frame: Select the reference frame (hold the camera who do the projection).

Hold Image: To frame hold the input image as the same frame as the camera.

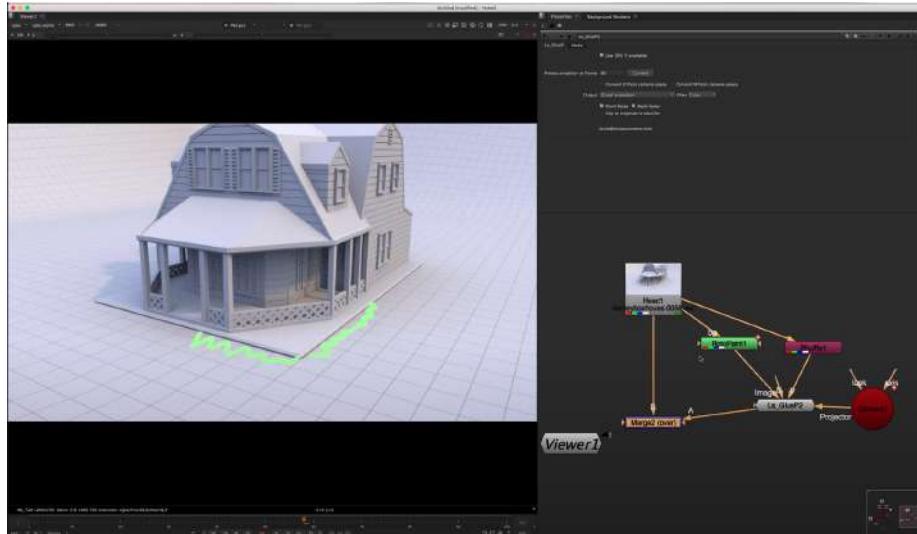
Crop: To project or not outside of the format area.

Antialiasing: To reduce the edges aliasing problems.

(Un)Premult: To keep nice shapes on edges.

GlueP LS

Author: Lewis Saunders - <http://lewissaunders.com/>



Github Link to file: <https://github.com/lcrs/blinks/>

Video Demo of the GlueP: <https://www.youtube.com/watch?v=jakExzK9MMw>

GlueP - glues an image to a position pass, by projecting on it, Using BlinkScript.

Inputs:

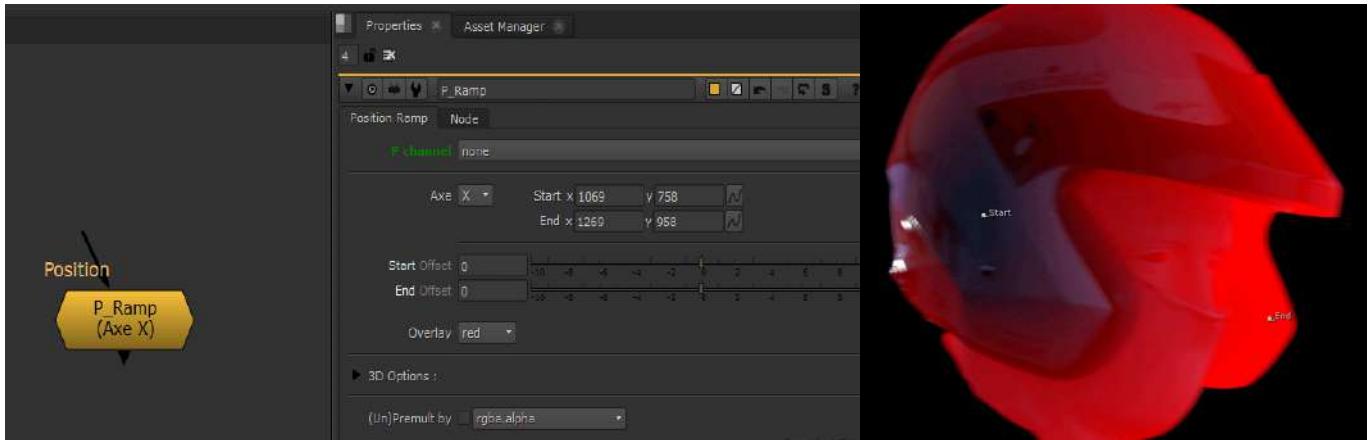
- Image to project
- P, the position pass
- N, an optional normals pass for front/back facing selection
- Projector, the Camera to project from
- CGCamera, optionally the original camera from the CG render

The second camera input can be used to convert P and N from camera space to world space - many renderers output camera space P and N by default, but we need world space to do this. Even better is a "world position at frame 1" or "P0" or "Rest position" pass, which sticks properly to the surface of animated meshes.

A good way to check that a position pass is suitable to project on is to plug it into a PositionToPoints node along with the beauty pass and check in the 3D view that the points lie within the camera frustum and are laid out in space like the 3D scene was.
It works best with 32-bit float P and N passes, rather than 16-bit half.

P_Ramp NKPD

Author: Franklin Toussaint - <http://franklinvfx.com/tools-2/>



Website download: http://franklinvfx.com/pos_ramp/

Github Link: https://github.com/franklinvfx/Position_Ramp-Tool-for-Nuke

To quickly create a ramp using the data contained in the render of "World Position" or "Ref Position".

This node has several advantages. The position of the ramp is not set by a "color picker" but with a 2D Position knob which executes callbacks. So it is possible to precisely place the ramp regardless of the channel viewed. With the "Overlay" mode, the tool is visually even more convenient. Visualization in 3D space is also a real advantage in a certain situation, as well as the integration of the "Unpremult" and "Premult" which makes it possible to obtain clean edges.

Tool details:

P Channel: To select the Position Pass channel.

Axe: To choose the ramp direction (x,y or z).

Start & End: To adjust the position of the two extremes of the ramp.

Offset (Start & End): To precisely adjust the position between the two extremes with a slider.

Overlay: To show the ramp on overlay (looking the rgb) and to change the color (none, red, green, blue or black)

3D Options:

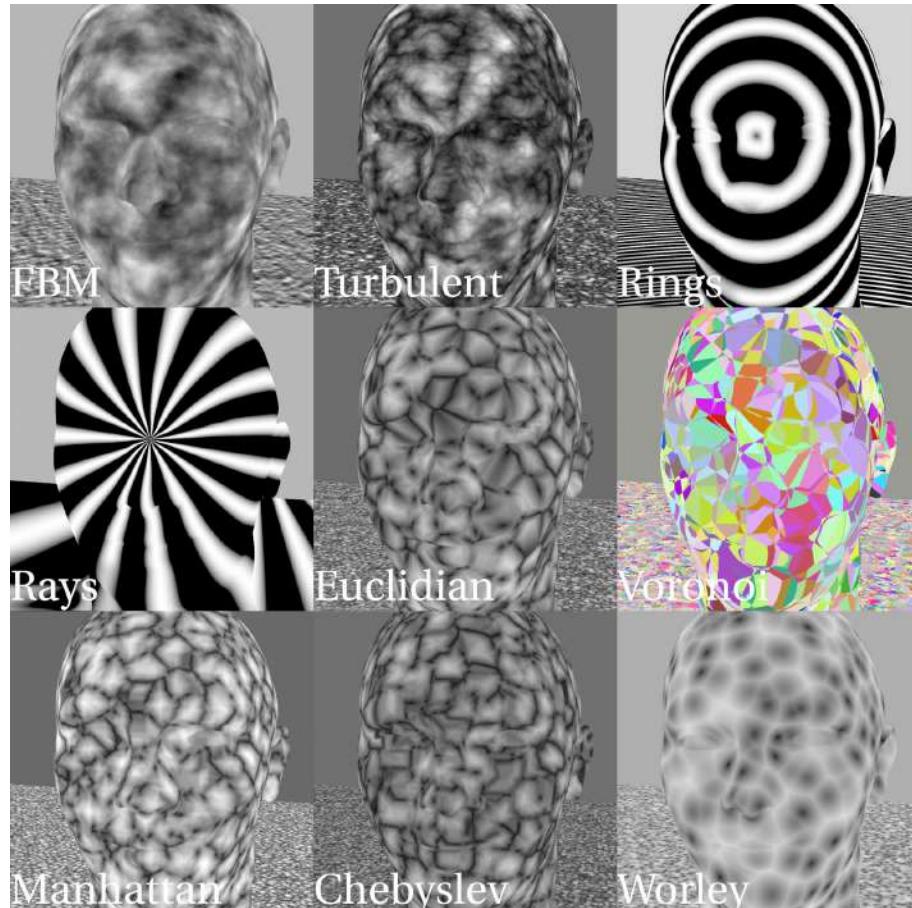
– To visualize in 3D the input image and the ramp. This uses a "position to point" node and it gives the possibility to adjust the "Point Detail" and the "Point Size".

– To adjust the size of 3D points representing the position of the two extremes of the ramp.

(Un)Premult: To keep nice shapes on edges.

P_NoiseAdvanced NKPD

Author: Riley Gray



Nukepedia link: http://www.nukepedia.com/blink/keyer/p_noise_advanced

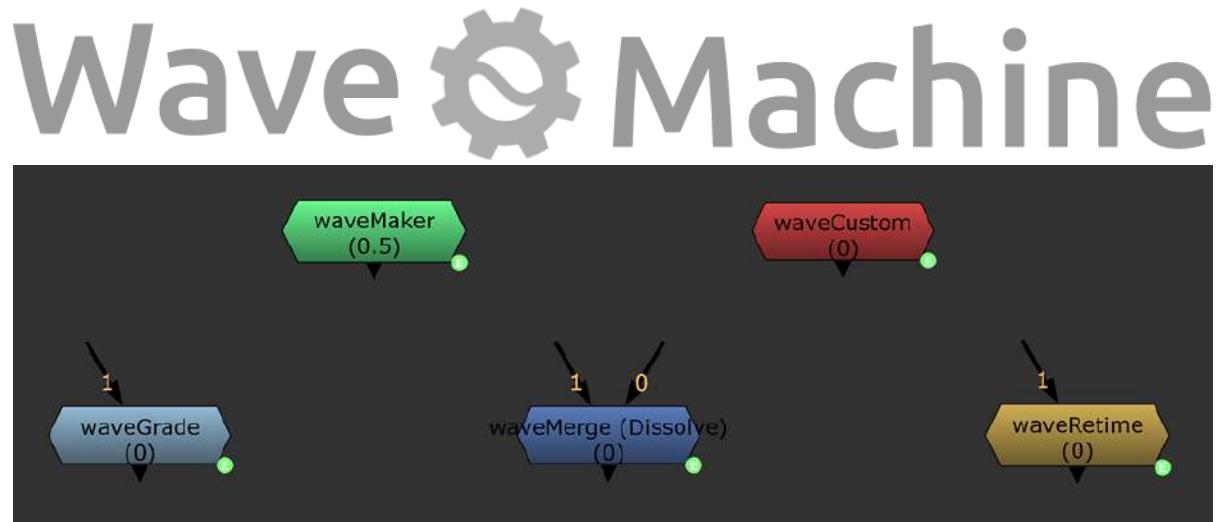
P_Noise_Advanced is a tool that allows you to apply various 3d noises and fractals to a rendered position pass such as world position or rest position. The various noise types are shown above and each has their own knobs and controls specific to each one.

A 2d position knob can be used to sample a center point in real time and Adrian Pueyo's 'c44kernal' allows you to transform the noise however you wish in 3d space. The Blink written noise types (Euclidian, Voronoi, Manhattan, Chebyslev and Worley) are based on matthew Shaw's 'Cell Noise' and just adapted to work on a position pass; click here to see his tool. The other noise types are simply written in tcl using nukes expression node.

14.) Curves

WaveMachine FL

Author: Fynn Laue - www.fynnlau.com



Nukepedia Link: <https://www.nukepedia.com/toolsets/other/wavemachine>

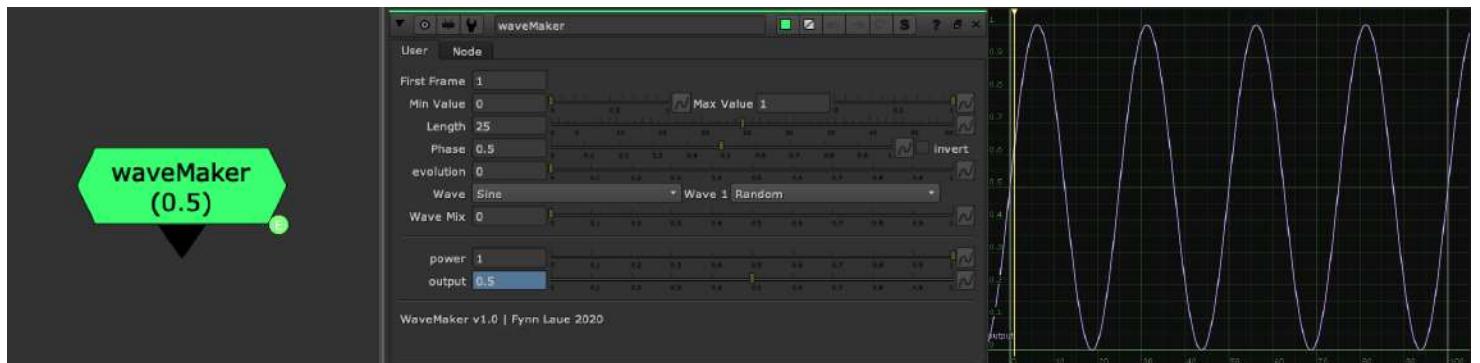
Demo Video: <https://vimeo.com/431245732>

Node Based Animation - Generate and modify animation curves with this toolset collection

WaveMaker FL

Author: Fynn Laue - www.fynnlau.com

Generate standard waves such as sine, triangle, square, random and more



First Frame: is the time pivot point for length(frequency)

Length: Higher values are slower. Values below 2 may produce undesirable results when using even waves like sine

Phase: Not counted in frames but in cycles: 1 'phase' is half a cycle (or evolution) of the wave.

Evolution: evolution for the random curve

Wave mix: Blend between 2 types of curves, 0 is wave0, 1 is wave1

Power: power is the final mix for the curve

Output: use with other wave machine nodes or to drive other nodes' animation

Wave types:

Sine

Triangle

Square

Bounce

Random

Sawtooth/Sawtooth

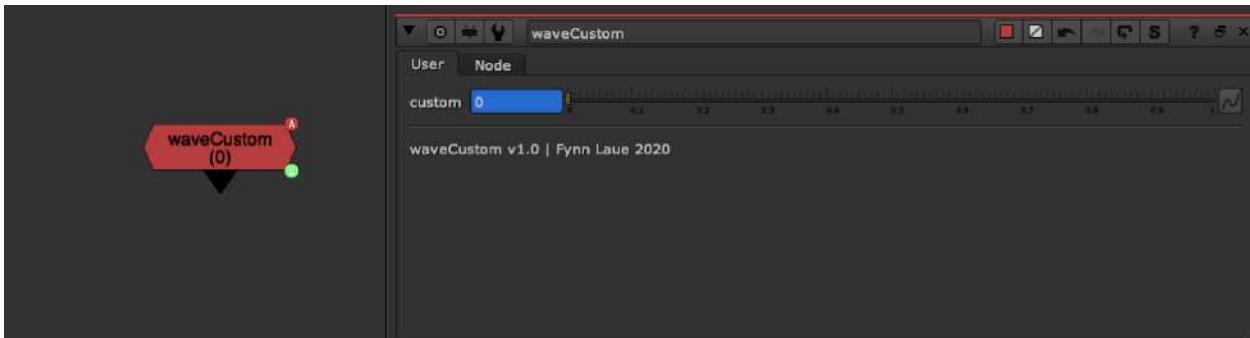
Sawtooth/Sawtooth (Parabolic)

Sawtooth/Sawtooth (Parabolic Reversed)

Sawtooth/Sawtooth (Exponential)

WaveCustom FL

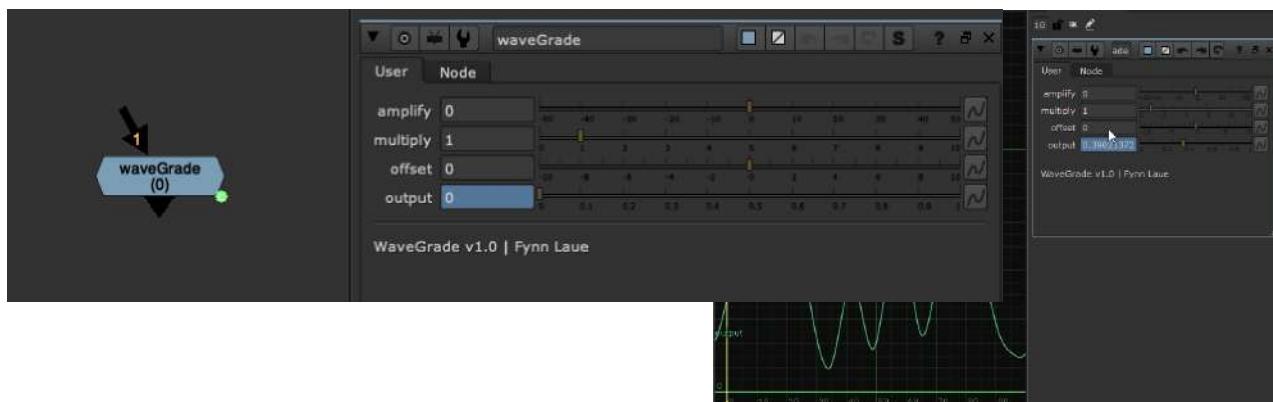
Author: Fynn Laue - www.fynnlau.com



Create a custom animation curve or expression link an external curve into this custom knob.

WaveGrade FL

Author: Fynn Laue - www.fynnlau.com



Grade animations. This way you can easily modify an existing animation or remap it for different ranges.

Amplify:

Multiply the average input value with this and add the input curve on top:
 $((inp1 - avgcurve) * amplify) + inp1$

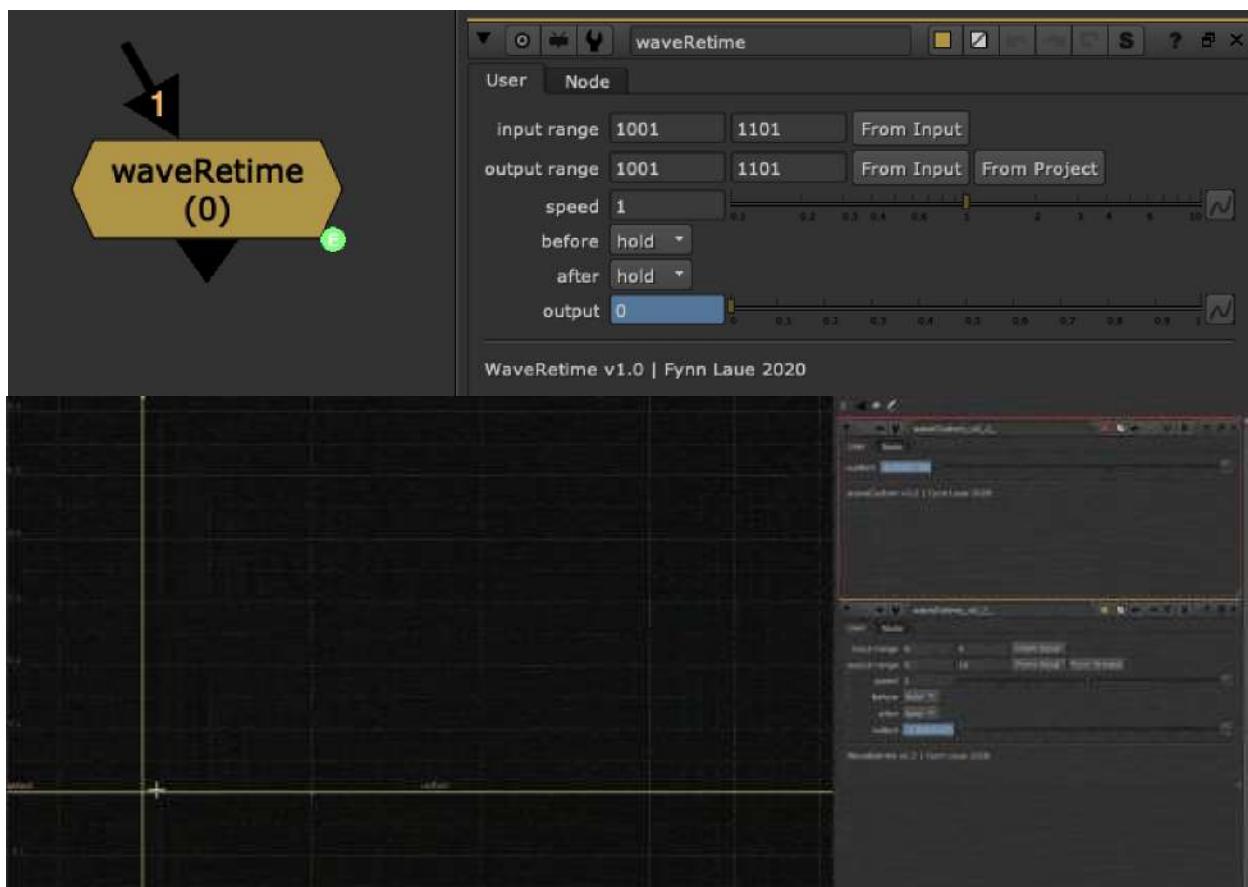
Multiply: Multiplies the input curve

Offset: adds an amount to the input curve

Plug this into the wave machine node stack to further manipulate the curve

WaveRetime FL

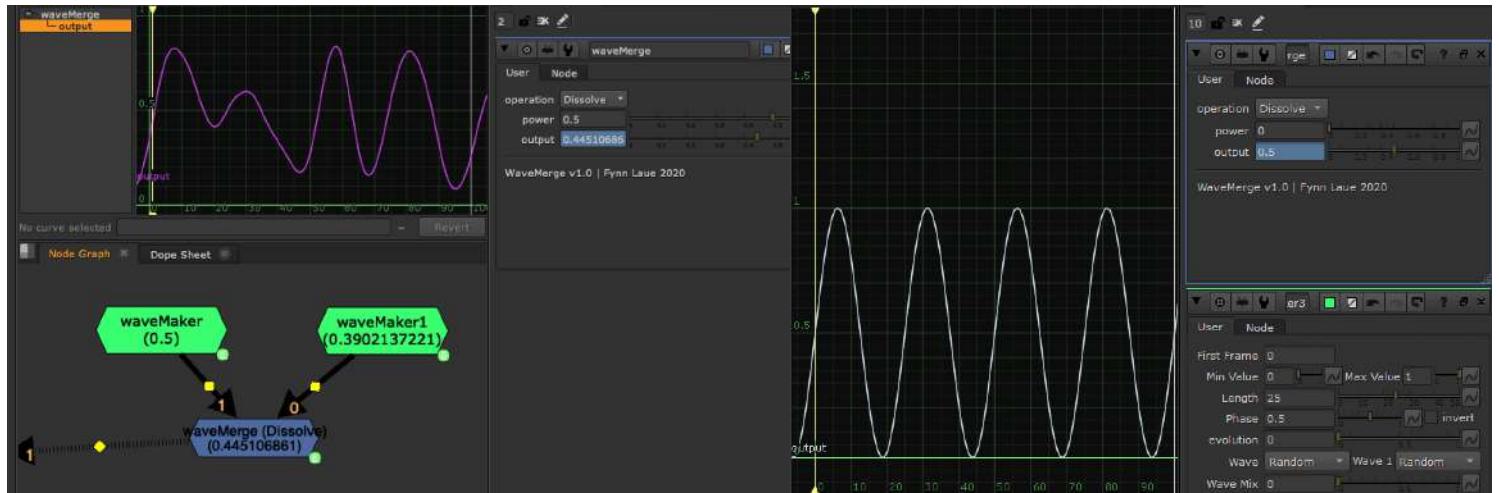
Author: Fynn Laue - www.fynnlau.com



Loop and retime input animation curves from wave machine nodes

WaveMerge FL

Author: Fynn Laue - www.fynnlau.com



waveMerge combines 2 different wave machine curves and mixes them with a few different merge operations:

Operations:

Add	$(\text{input0} + \text{input1} * \text{power})$
Dissolve	$((\text{input0} * (1 - \text{power})) + (\text{input1} * \text{power}))$
Max	$(\max(\text{input0}, \text{input1} * \text{power} + (1 - \text{power}) * \text{input0}))$
Min	$(\min(\text{input0}, \text{input1} * \text{power} + \text{input0} * (1 - \text{power})))$
Multiply	$(\text{input0} * (\text{input1} * \text{power}))$
Minus	$(\text{input0} - (\text{input1} * \text{power}))$
Union	$((\text{input0} * (1 - \text{power})) + ((\text{input0} + \text{input1}) - (\text{input0} * \text{input1})) * \text{power})$

Power:

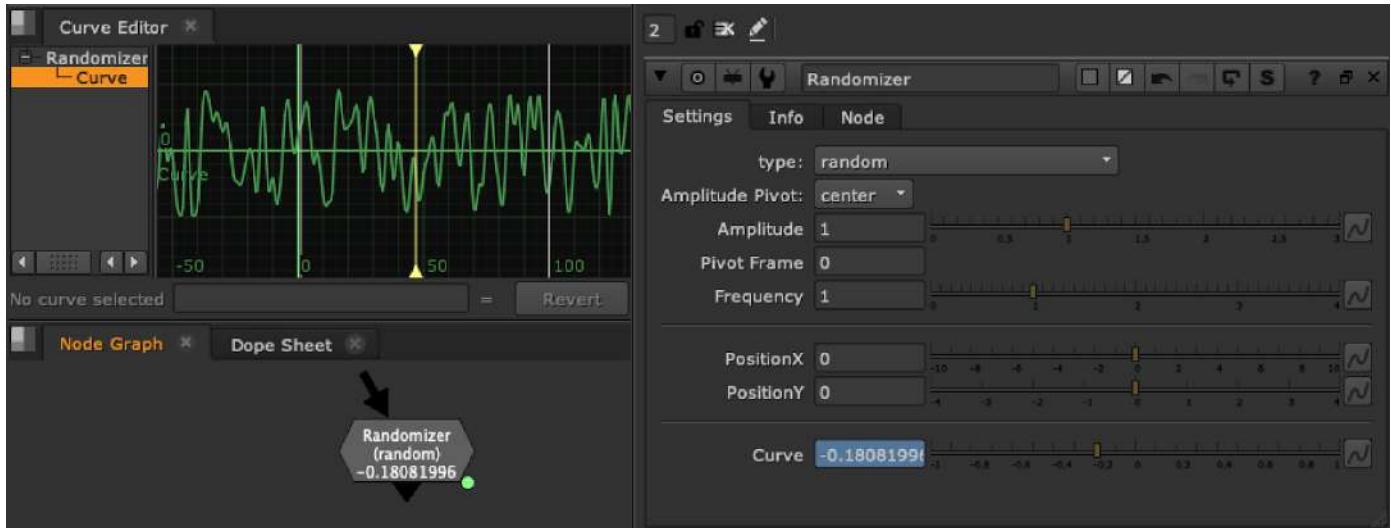
This is your mix slider for the operation

0 = curve from input 0

1 = curve from input 1

Randomizer TL

Author: Tony Lyons - www.CompositingMentor.com



Randomizer is aimed at being a simple curve manipulation tool. It is meant to be used with the curve editor and to have terminology relating to graphs

Pick between the following curve types:

random, noise, sine, triangle, square, bounce, sawtooth, sawtooth (parabolic), sawtooth (parabolic reversed), sawtooth (exponential), blip, sine blip

Amplitude scales the curve in the Y axis. You can set the pivot of the scale point to either center, min, or max, depending on how you want your curve to scale.

Frequency scales the curve in the X axis, and pivot frame serves as the pivot point for the X axis scale, so if you set to frame 1050 and scale the frequency, frame 1050 will remain the same value and scale outward from there.

Position X and position Y are simple controls to move the curve up and down and left and right on the curve editor.

You can sort of 'stack' or drive multiple curves by expression linking another curve/randomizer into the Position Y, amplitude, or frequency knobs, or by manually animating them.

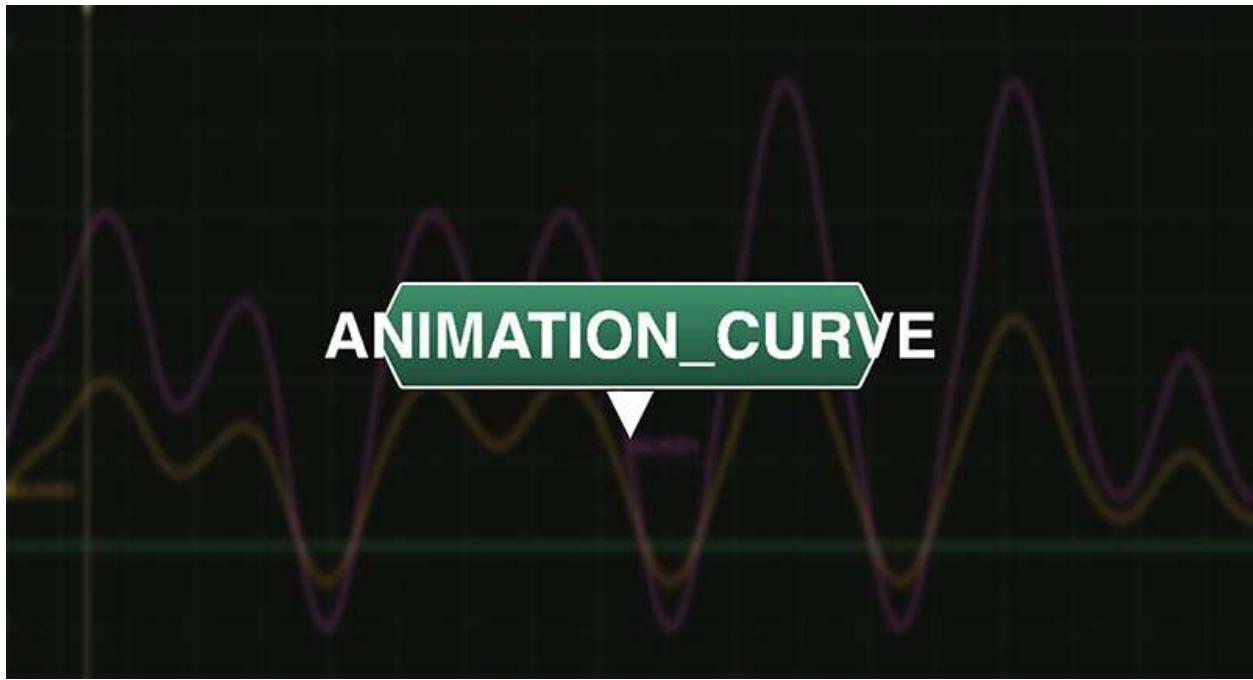
You can achieve some pretty dynamic results.

Squareify option makes random and noise steppy, and included a random seed button

This tool can also be used with the Wave Machine toolset

AnimationCurve AG

Author: Andrea Geremia - <http://www.andreageremia.it/tutorial.html>



Nukepedia link: http://www.nukepedia.com/gizmos/other/animation_curve

Very detailed tutorial: www.andreageremia.it/tutorial_animation_nuke.html

Generate or modify animation curves

All the functions in this Gizmo:

Wave Generator

New Range

Smooth Curves

Modify Curves

Fade

Reference Frame

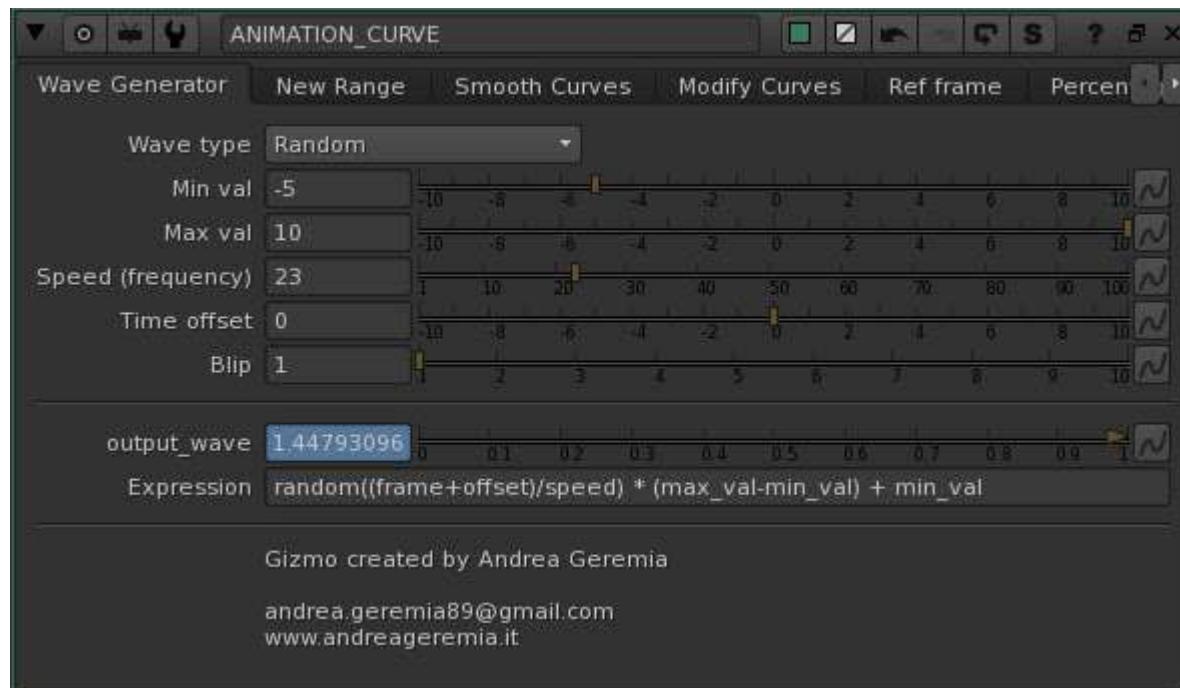
Percentage

Average

1. Wave Generator

In this first Tab you can generate a Wave with different options. For of all, select the Type. Please, check all the available types.

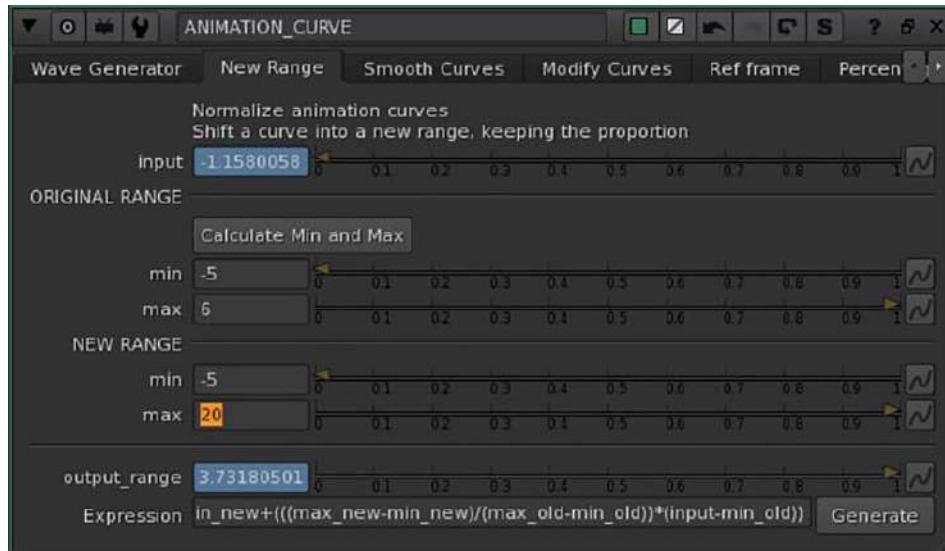
- Noise
- Random
- Sine
- Sine Blip
- Triangle
- Square
- Bounce
- Blip
- Saw Tooth
- Saw Parabolic
- Saw Parabolic Reverse
- Saw Exponential



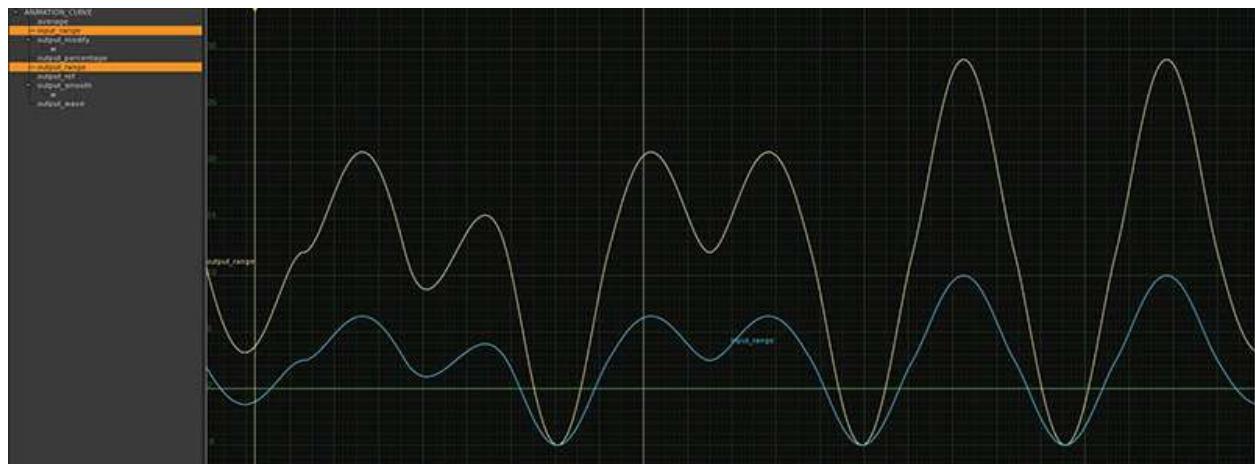
2. New Range

Here you can change the range of an Animation Curve. For example, if min and max values are -5 and 6, you can project all the curve in the new frame range -5 and 20.

Drag and Drop your curve in the knob Input and use the button to find out the min and max value, then insert your new values and check in the Curve Editor your new curve.

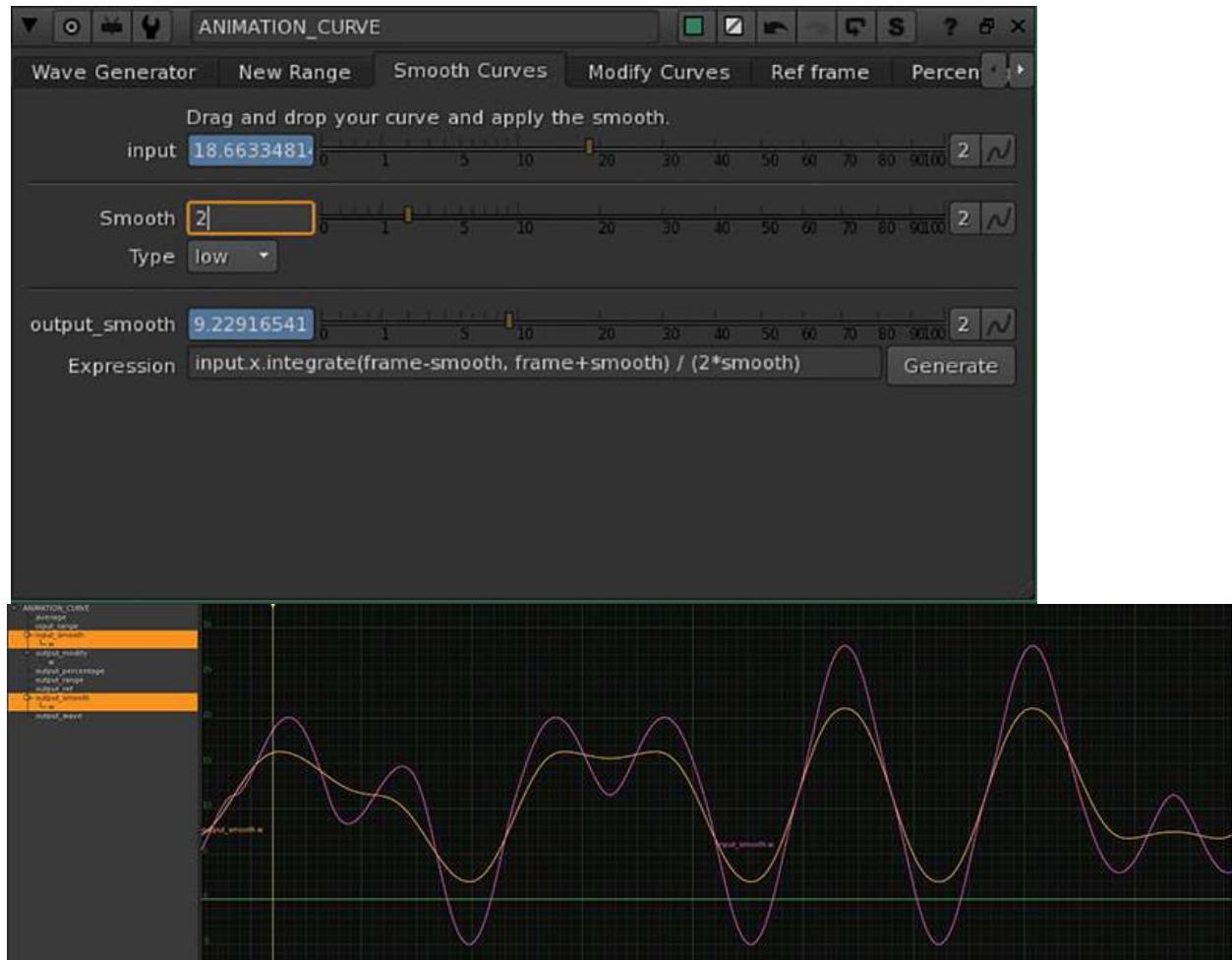


In blue, you can see the original curve and in yellow the new one



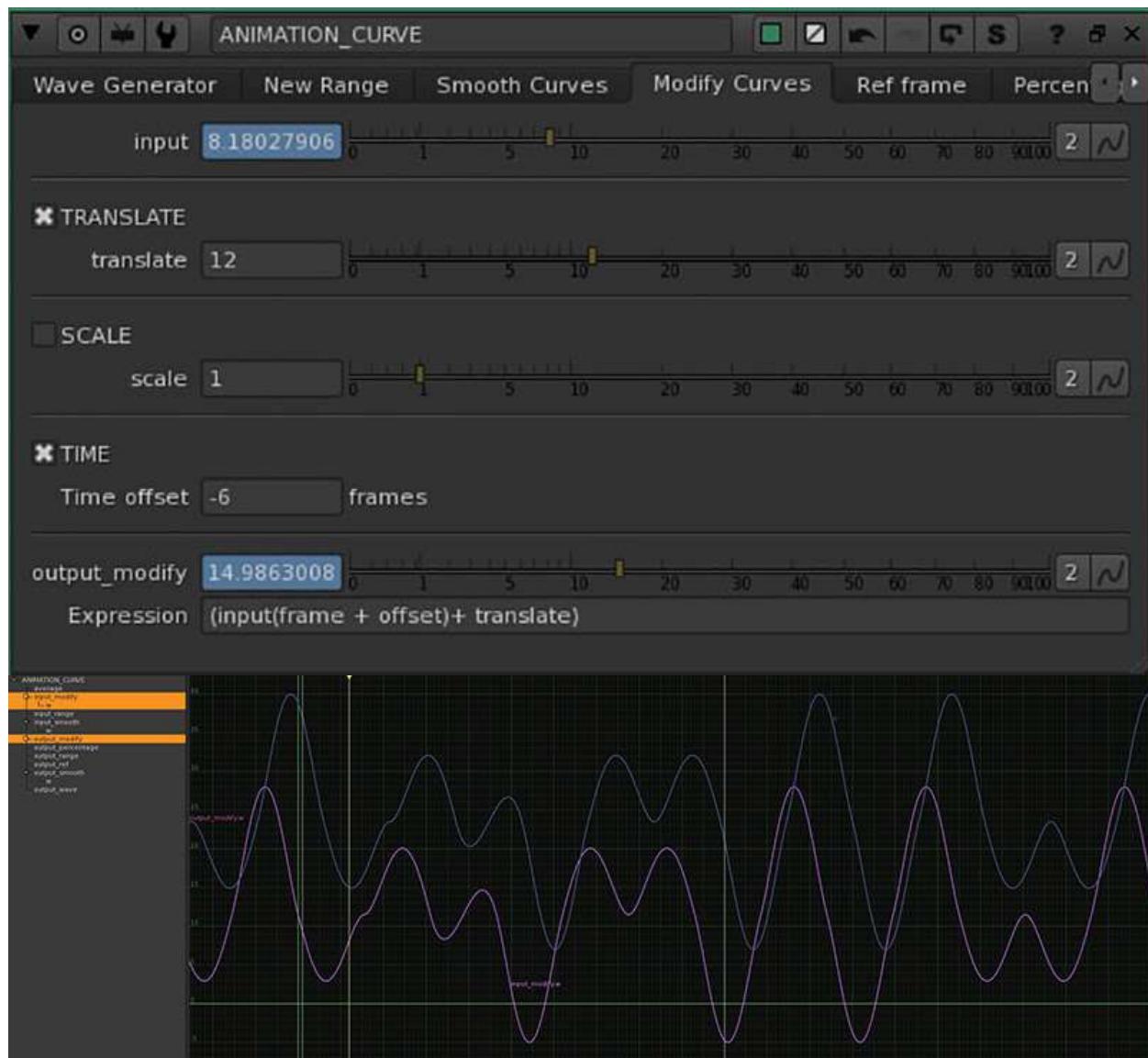
3. Smooth Curves

Smooth the curve with this tool. Insert your curve in the knob Input, select the Type (High or Low) and the power of the smooth



4. Modify Curves

With this tool you can modify curves with: Translate, Scale and Time Offset. With the checkbox you can activate or not the modifiers.



5. Fade

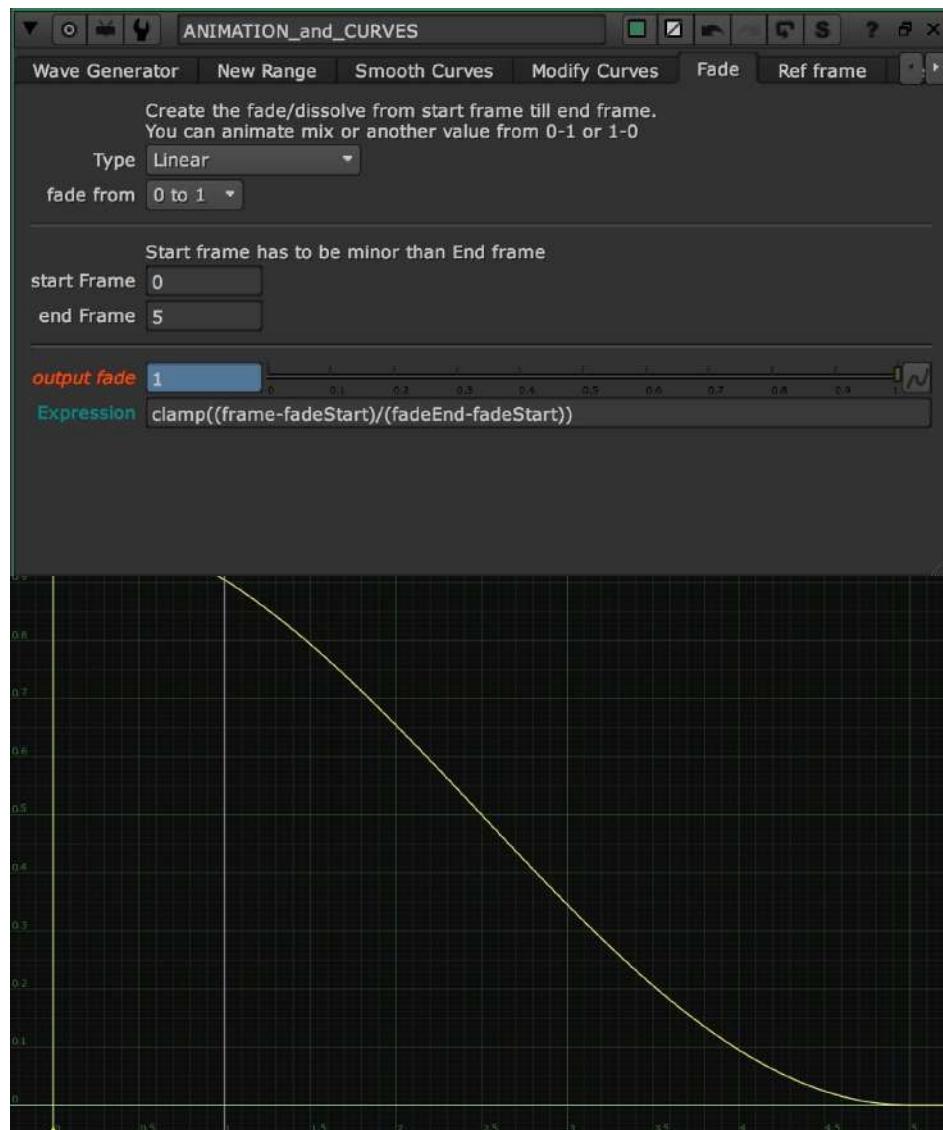
Create the fade/dissolve from start frame till end frame. Animate mix or another knob from 0-1 or 1-0.

Select between:

- Linear
- Slow-in Slow-out
- Slow-in Linear-out
- Linear-in Slow-out

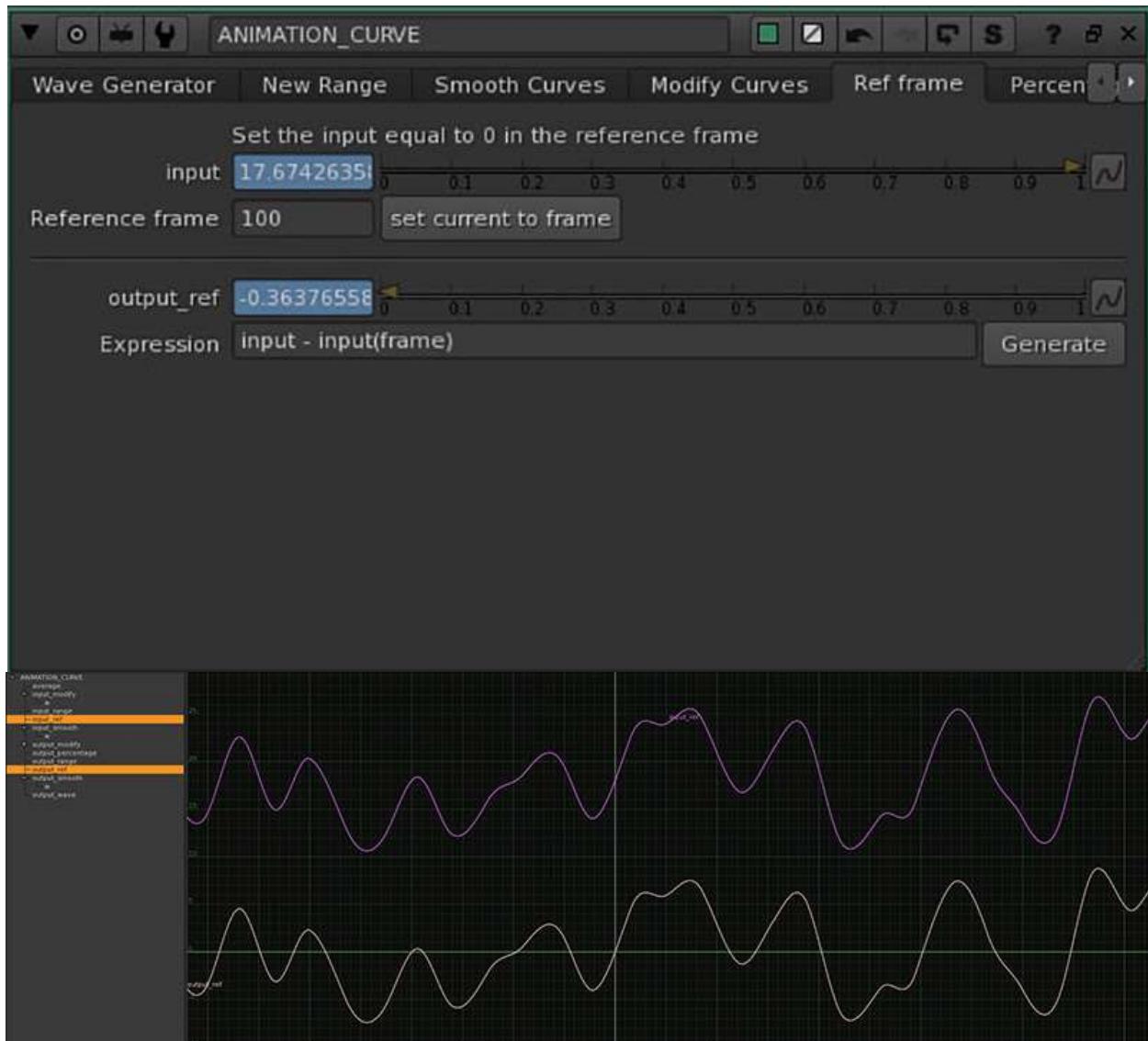
From David Ozols' tutorial

<HTTPS://DJOZOLS.COM/2018/03/01/SIMPLE-FADE-EXPRESSIONS-IN-NUKE/>



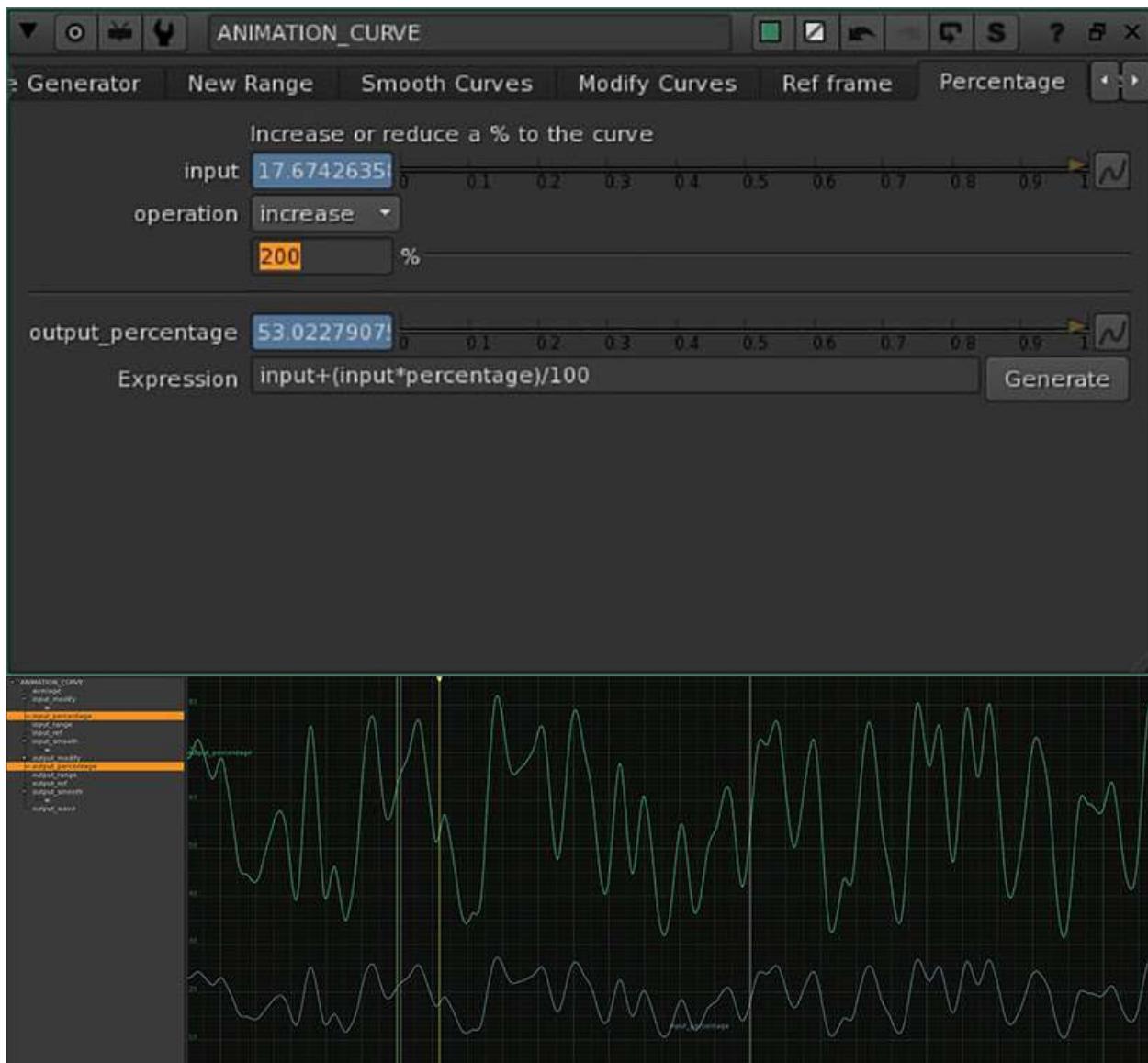
6. Reference Frame

Here set a reference frame. What does it mean? It means that your curve will be set to 0 in that frame. Basically it will be translated to 0 in the Reference Frame.



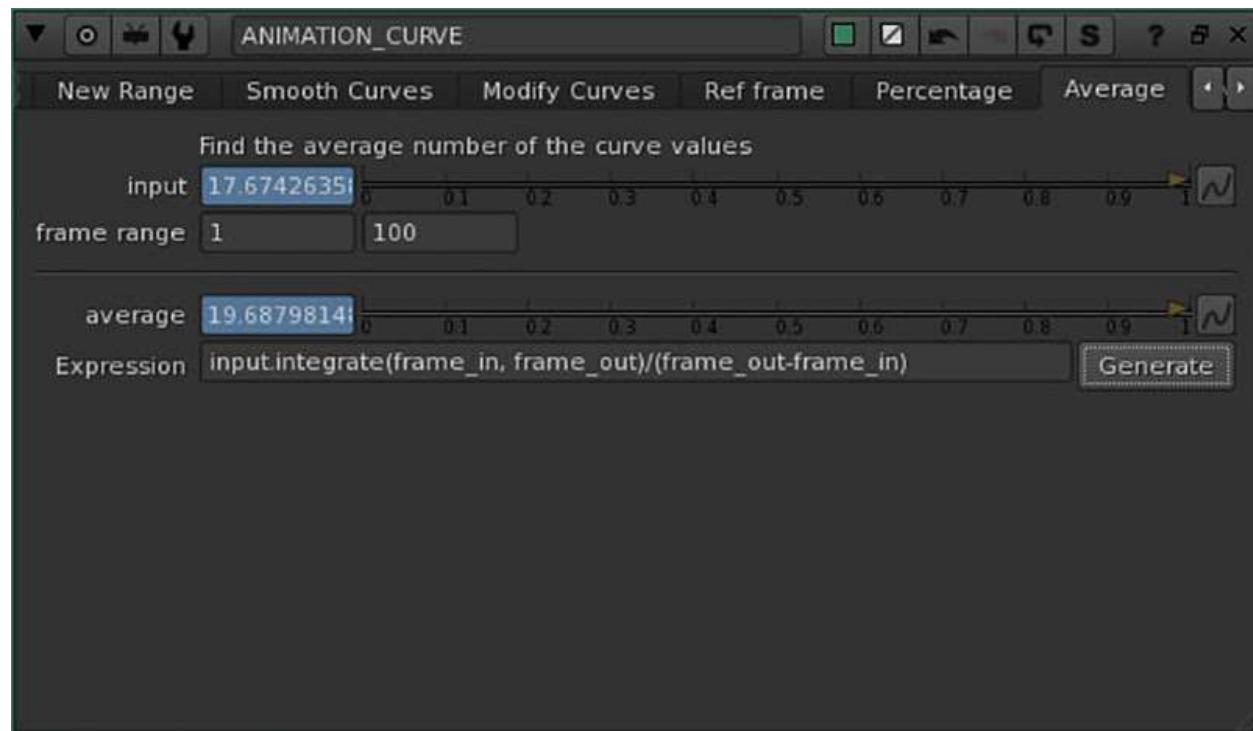
7. Percentage

Increase or Decrease the curve for a X percentage.



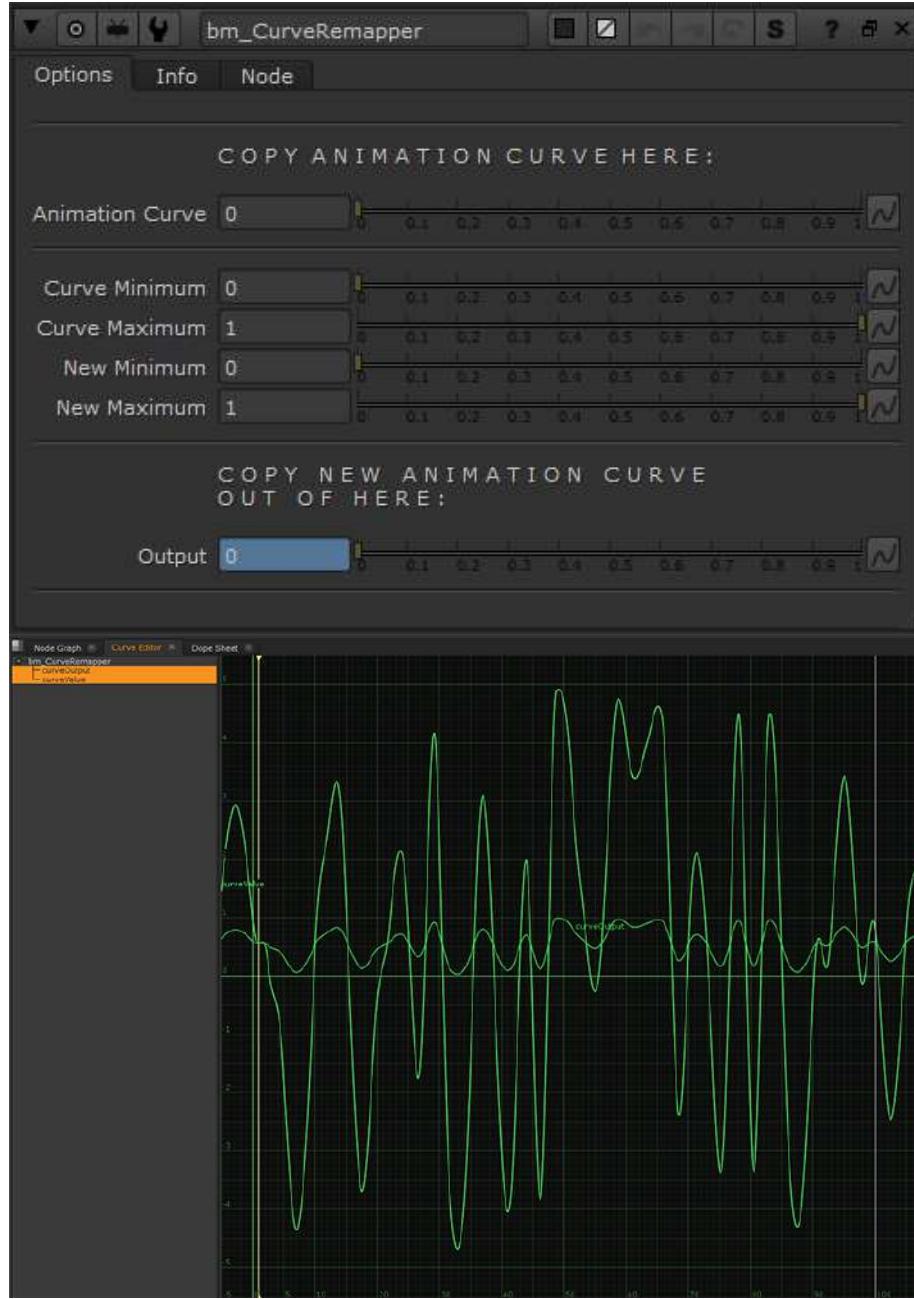
8. Average

This is a single value for the entire curve. It will return the average value for the curve.



CurveRemapper BM

Author: Ben McEwan - <https://benmcewan.com/blog/>

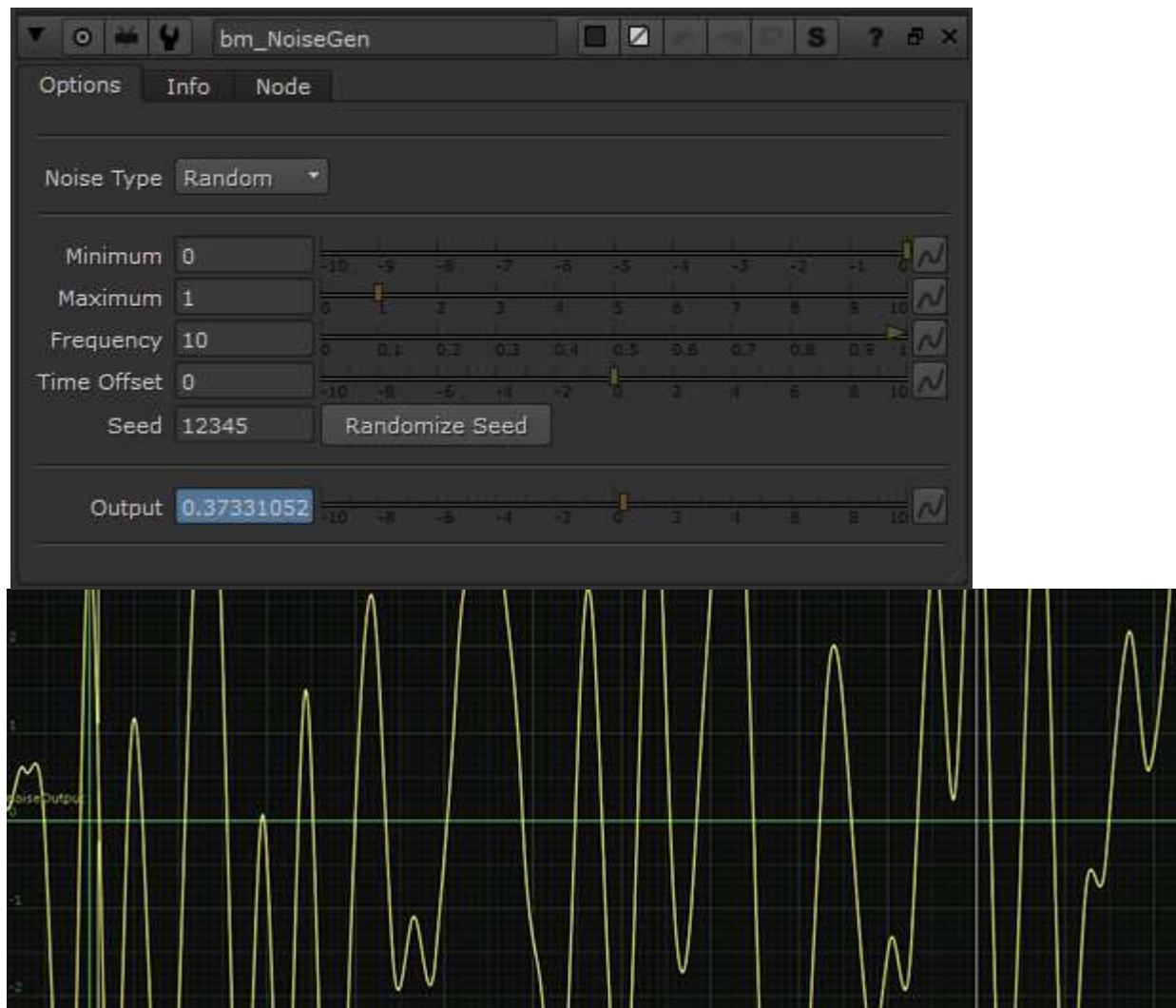


Nukpedia Link: http://www.nukpedia.com/gizmos/other/bm_curveremapper

Useful for remapping arbitrary animation curves, such as those from the CurveTool. Input the animation's min & max values, then set new min & max values.

NoiseGen BM

Author: Ben McEwan - <https://benmcewan.com/blog/>



Download tool: <https://benmcewan.com/nukeTools.html>

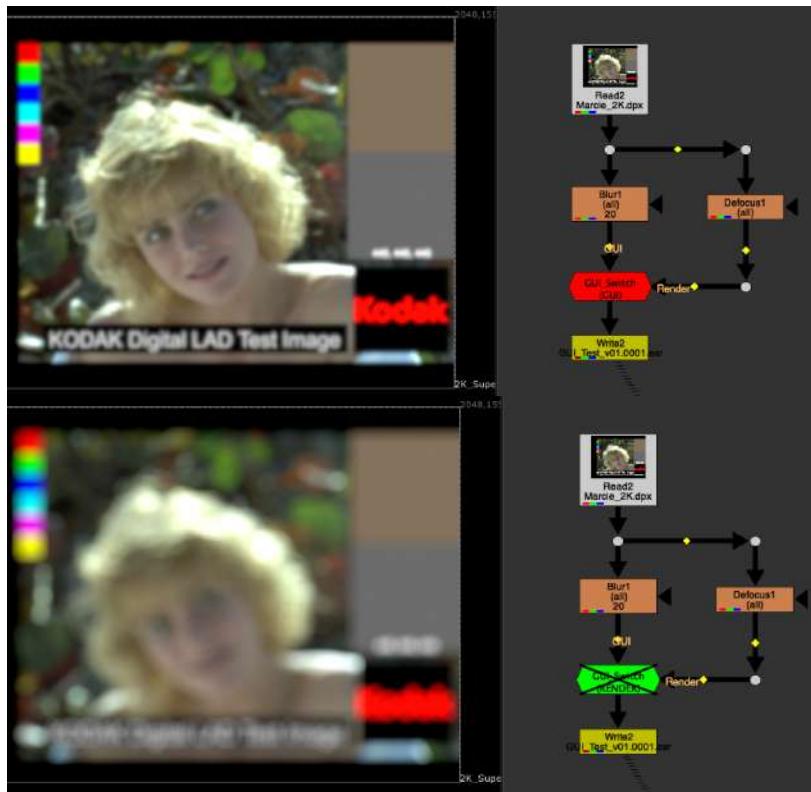
Generates a random noise curve based on a minimum, maximum & frequency value.

Modified so the output curve will work with Wave Machine toolset

15.) Utilities

GUI_Switch TL

Author: Tony Lyons - www.CompositingMentor.com



There were some issues with the \$gui expression that everyone knows that when you render a frame locally instead of on a render farm, it did not register and would not switch over.

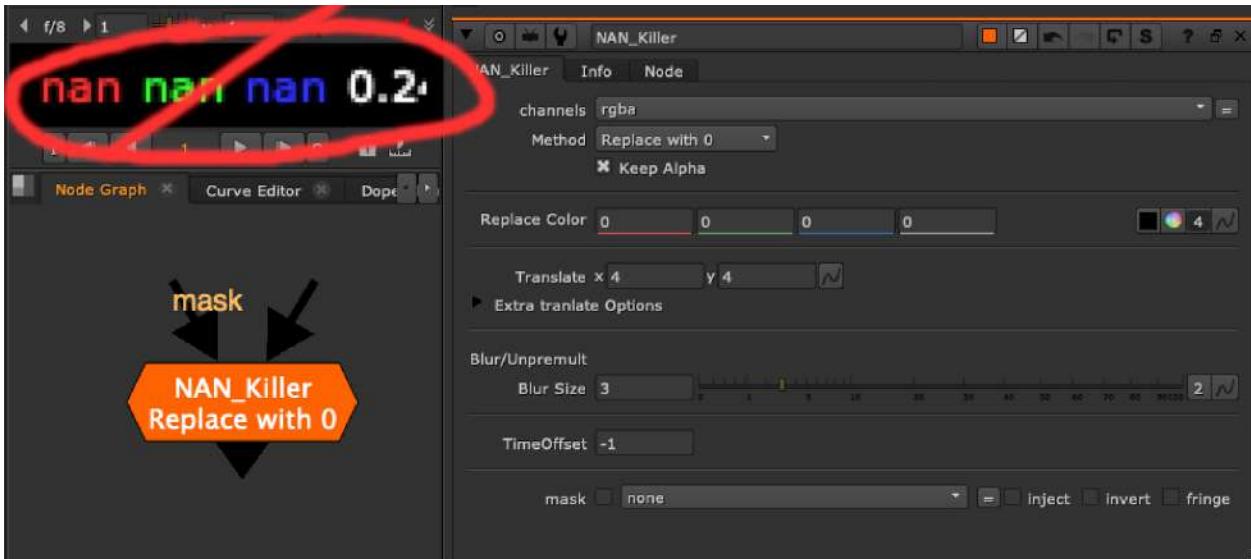
This tool is using a lesser known python expression called `nuke.executing()` and seems to solve this problem. Inside the gizmo is just a switch node with the expression on the 'which' knob. 2 inputs, GUI and Render. Plug GUI into the node you want to view live in the nuke script while you are working, and Render input plug into whatever you want to switch to when you are rendering / executing something. Usually this is for speedy work environments and switching to higher settings / samples on render time

For visuals, when the node is on it's Red and says GUI, and if you disable the node, it will turn Green and say RENDER. Hopefully this will let you conveniently preview the Render or GUI

Node has no settings, just a description. Just plug in and disable/enable!

NAN_INF_Killer TL

Author: Tony Lyons - www.CompositingMentor.com



Kills NAN and INF pixels using a variety of replacement methods:

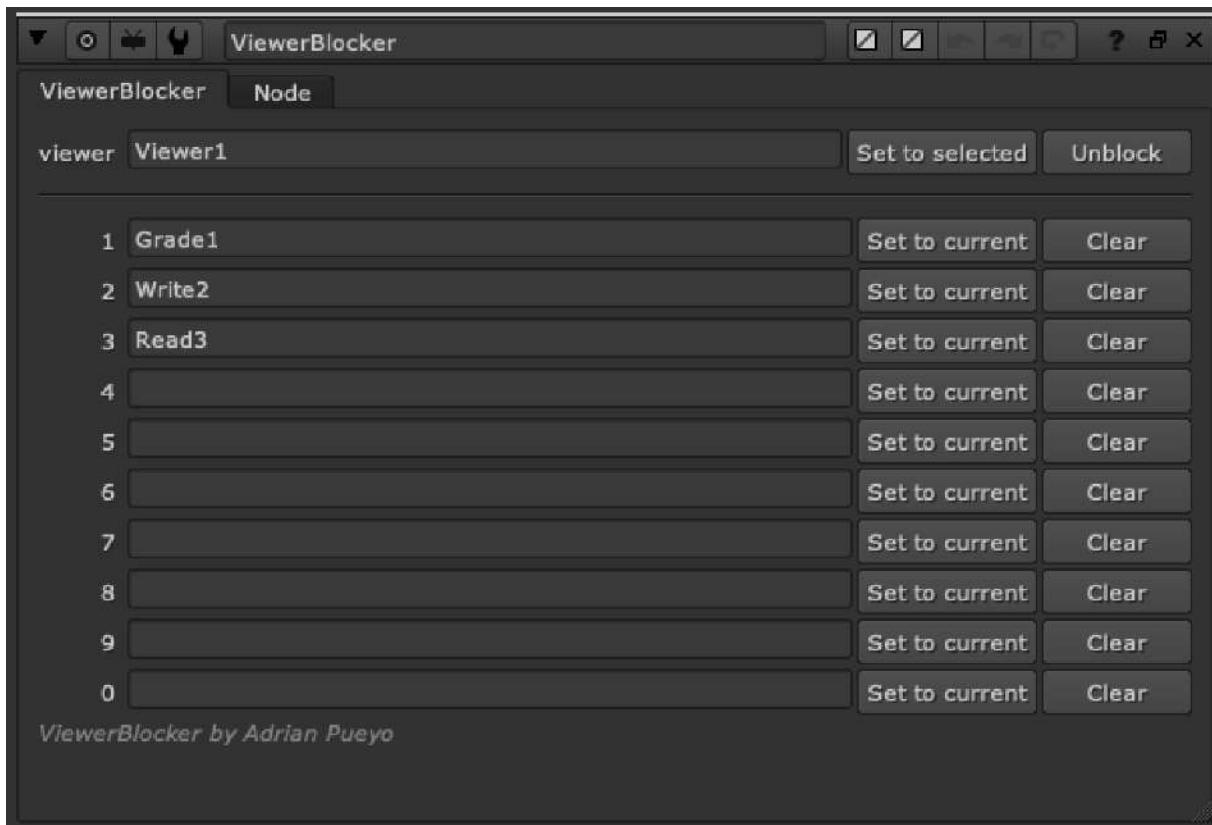
- Replace with 0
- Replace with Custom Color
- Clone Over
- Blur/Unpremult
- Time

Keep alpha will keep the original alpha if the alpha channel does not have NAN or INF in it.
Separates RGB from Alpha in expression

Exposed settings for replace color, transform (for Clone over), blur/unpremult, time offset

apViewerBlocker AP

Author: Adrian Pueyo - <http://www.adrianpueyo.com/>



Locks the Viewers input to specific nodes, so you can reference the same views when you are using the numeric hotkeys without worrying about resetting or accidentally switching the viewer inputs. Good for referencing certain images, like Final image set to 9, reference image set to 8, plate set to 0, etc

Select a viewer you wish to Block / Lock and click set to Selected on the viewer box

View an image that you wish to save/ lock to hotkey and ‘set to current’ on one of the numbers 1,2,3,4,5,6,7,8,9,0. This will enter the nodes name in the text input and lock that viewer input to that node so when you use the number hotkey it always views this node.

Clear any fields to reset them.

Python_and_TCL AG

Author: Andrea Geremia - www.andreageremia.it/tutorial.html

The screenshot shows the Python_and_TCL AG Gizmo interface. At the top, there is a large Python logo. Below it, the code editor displays two files: `Python_and_TCL.py` and `Python_and_TCL.tcl`. The `Python_and_TCL.py` file contains Python code for interacting with Nuke nodes and their connections. The `Python_and_TCL.tcl` file contains TCL code for similar operations. Below the code editor is a tab bar with "Read Me", "Node", and "operation". The "operation" tab is selected, showing "00. Python & TCL Overview" and a "Create Sticky Note" button. Under the "Read Me" tab, there is a "Python" section titled "PYTHON OVERVIEW" containing basic print statements and variable assignments. Under the "TCL" tab, there is a "TCL OVERVIEW" section with examples of reading node values. The "Website" tab lists several URLs related to Python and TCL documentation.

```
1 #ARDI DKB8
2
3 # List all knobs for selected node
4 #return knobNodeList()
5
6 # List all knobs for specific node
7 #return knobNodeList(node)
8
9 # Get value from specific node, method 01
10 #value = knobNodeList('knob1').knob('Color').getValue()
11
12 #Gett value from specific node, method 02
13 #value = node['knob1'].knob('Color').getValue()
14 #print node['knob1'].value()
15
16 # Get value from selected node
17 #value = nuke.selectedNode().knob('file').getValue()
18
19 #Get default value
20 #value = nuke.nodes('Blur1').knob('size').defaultValue()
21
22 # Get value from specific node and evaluate it (good for expressions)
23 #value = nuke.nodes('Knob1').knob('file').evaluate()
24
25 #-----#
26 #----CONNECT NODES
27 #-----#
28
29 #Get the expression of a knob
30 #if nuke.nodes('Blur1').knob('size').defaulExpression():
31 #    print nuke.nodes('Blur1').knob('size').defaulExpression()
32
33 #Gett value X from individually
34 #nuke.nodes('Tzschroff1').knob('x').defaultValue()
35 #nuke.nodes('Tzschroff1').knob('y').defaultValue()
36
37 #Gett value from the weight and width from a selected node
38 #nuke.selectedNode('knob1').weight()
39 #nuke.selectedNode('knob1').width()
40
41 #CHOOSE FROM BULDOWN CHOICE
42
43 #Get value from Bulldown Choice (name)
44 #value = nuke.nodes('Blur1').knob('file').value()
45
46 #Get index from Bulldown Choice (name)
47 #int(nuke.nodes('Blur1').knob('file').getVal())
48 #int(nuke.nodes('Blur1').knob('file').getVal())
49
50 #Check the name of a knob will be needed later
51 #node = nuke.selectedNode()
52 #node['knob1'].name()
53
54 #Get a script for every selected node
55 #script = nuke.selectedNodes() # The list of
```

```
1 import os
2 from PyQt import QtGui, QtCore
3 from Qt import QC
4 from Qt import QCImage
5 from Qt import QCImage
6 from Qt import QCImage
7
8 #load the file for the interface
9 file_interface = os.path.join("script/gizmos/PythonAndTcl", "file.py")
10
11 #-----#
12 #----CREATE LIST NODES
13 #Populate the list nodes
14 #for n in nuke.allNodes():
15 #    if n.Class == "Node":
16 #        nodeClass = n.Class
17 #        nodes.append(node)
18
19 #root the list
20 #nodes()
21
22 #-----#
23 #----LIST ALL SELECTED NODES
24 #nukecripts.create_selection_recursion()
25
26 #-----#
27 #----SELECT NODE
28 #-----#
29 #Select node
30 #nuke.nodes('Knob1').knob('size')
31 #nuke.nodes('Knob1').knob('size').setSelected(True)
32 #nuke.nodes('Knob1').knob('size').setSelected(False)
33 #nuke.nodes('Knob1').knob('size').setSelected(True)
34 #nuke.nodes('Knob1').knob('size').setSelected(False)
35
36 #Manipulate the Node Graph. Icon on the Selected Node
37 nuke.iconify( [ n.xpos(), n.ypos() ] )
38
39 #-----#
40 #-----#
41 #Select node "Grade"
42
43 #-----#
44 class MyWindow(QDialog):
45     def __init__(self, parent=None):
46         super(MyWindow, self).__init__(parent)
47         self.resize(400, 300)
48         self.setWindowTitle("Python And TCL Reference")
49         self.setWindowModality(Qt.WindowModal)
50
51         self.loadUI()
52
53     def loadUI(self):
54         self.layout = QVBoxLayout()
55
56         self.setLayout(self.layout)
```

Read Me Node operation 00. Python & TCL Overview Create Sticky Note

Python PYTHON OVERVIEW

```
#-----
#PRINT and VARIABLES
print 'Hello World'

name = 'Andrea'
print 'Hello' + name

#-----
```

TCL TCL OVERVIEW

```
Getting a knob's value of a specific node:

#First frame of current read/write:
[value Read1.first]

#Getting a knob's value of current node:
[value this.first_frame]
```

Website <https://www.tutorialspoint.com/python/index.htm>
<https://learn.foundry.com/nuke/developers/63/pythondvguide/basic.html>
<https://learn.foundry.com/nuke/developers/63/pythondvguide/index.html>

<http://www.nukepedia.com/tcl/tcl-reference>

Nukepedia link: <http://www.nukepedia.com/gizmos/other/python-and-tcl-tips-and-tricks>

Huge collection of Python and TCL snippets to use inside Nuke

In this Gizmo you will find out some tips and tricks that I've wrote and picked up from other websites about Python and TCL.

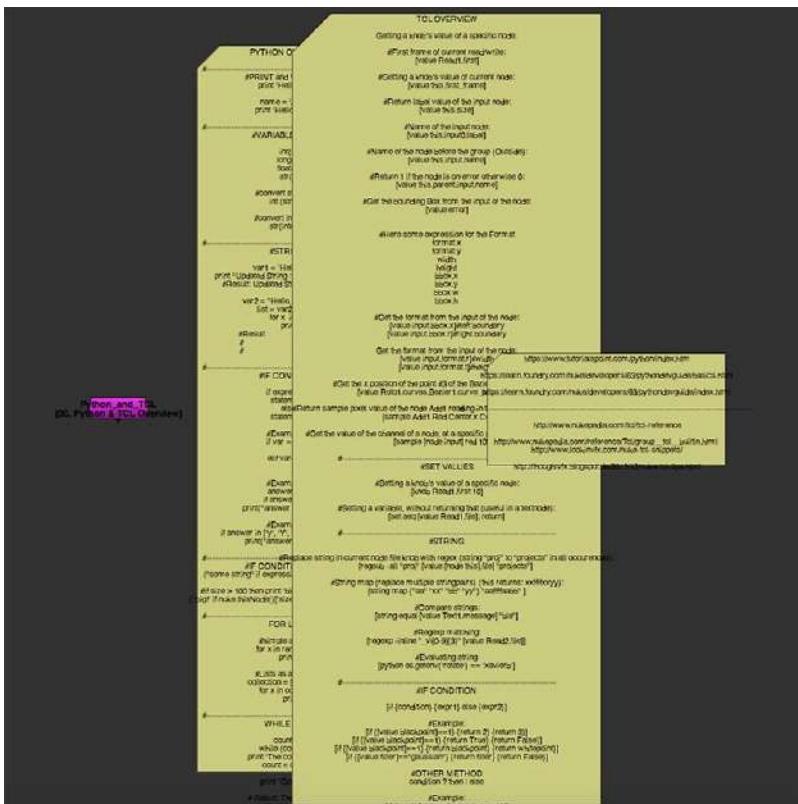
Check out the complete guide even here:

http://www.andreageremia.it/tutorial_python_tcl.html

It's a Guide by categories to read directly in Nuke, really useful for all the Nuke Artists.

Sections of the Guide:

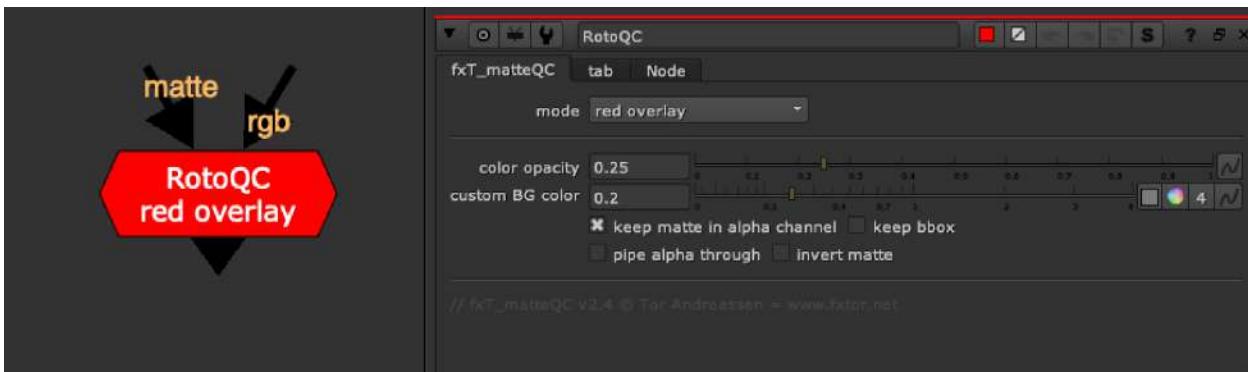
- 00. PYTHON AND TCL OVERVIEW
 - 01. CREATE NODE
 - 02. SELECT NODE
 - 03. CONNECT NODES
 - 04. READ FROM A KNOB
 - 05. WRITE INTO A KNOB
 - 06. CREATE A NEW KNOB
 - 07. ANIMATION AND CURVE
 - 08. EXPRESSIONS
 - 09. MATH FUNCTIONS AND WAVE GENERATOR
 - 10. FUNCTIONS DEF()
 - 11. CALLBACKS
 - 12. CUSTOMS PANELS
 - 13. TRICKS



RotoQC NKPD

Author: Tor Andreassen - <http://www.fxtor.net/nuke.html>

Originally named fxT_matteQC



Nukepedia Link: <http://www.nukepedia.com/gizmos/other/matteoverlay>

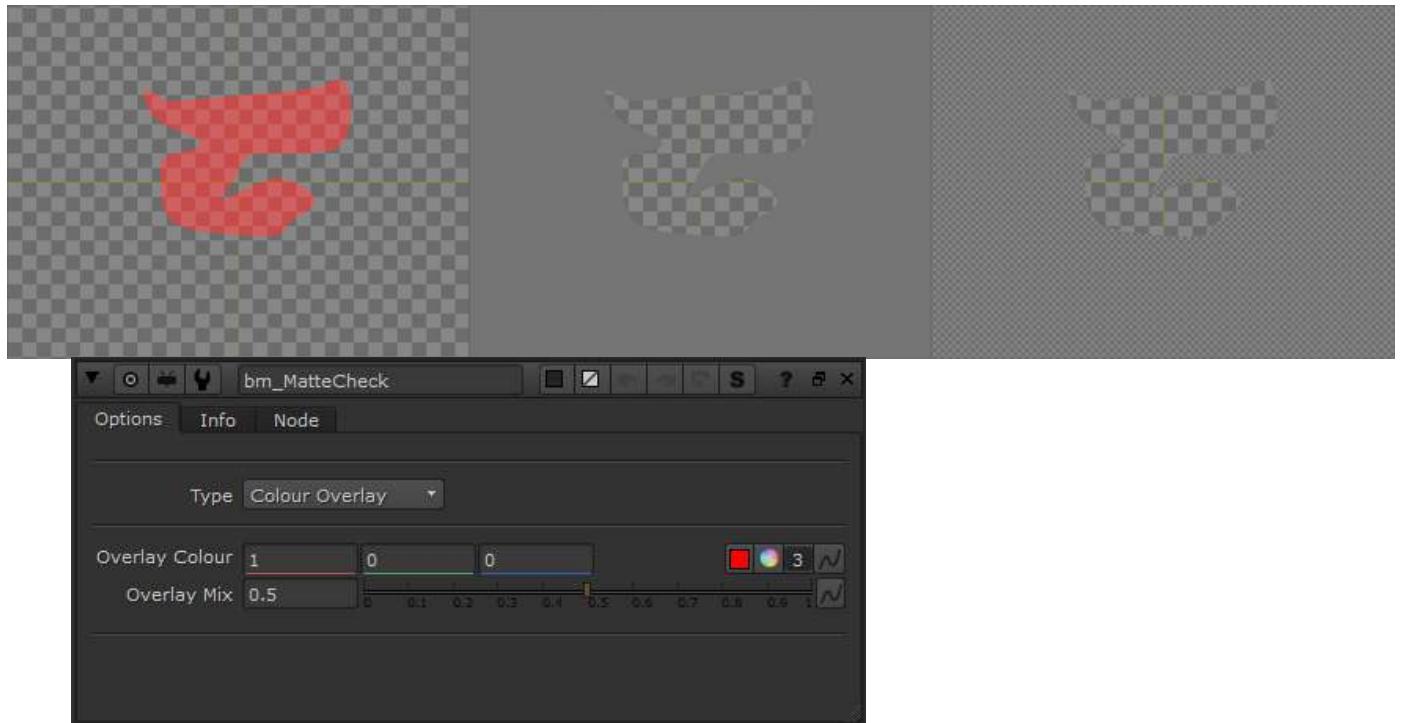
Tool intended to check roto matte's accuracy.

This is a QC tool for checking roto-mattes. It works with all formats (format is grabbed from the input), meaning it will work with all aspect ratios, including anamorphic plates.

The node has these QC options: overlay of red, green, or blue, Stencil from plate, premult against BG (grey, checkerboard, or custom color). It's set up in a way so all the user needs to do is pick the QC mode from a dropdown menu, and the rest is handled by the gizmo, including alpha handling. For color overlay the user can control the opacity with a slider. The node is stackable and alpha will carry through so you can have as many of these nodes you want and the alpha will still be correct.

bm_MatteCheck BM

Author: Ben McEwan - <https://benmcewan.com/nukeTools.html>

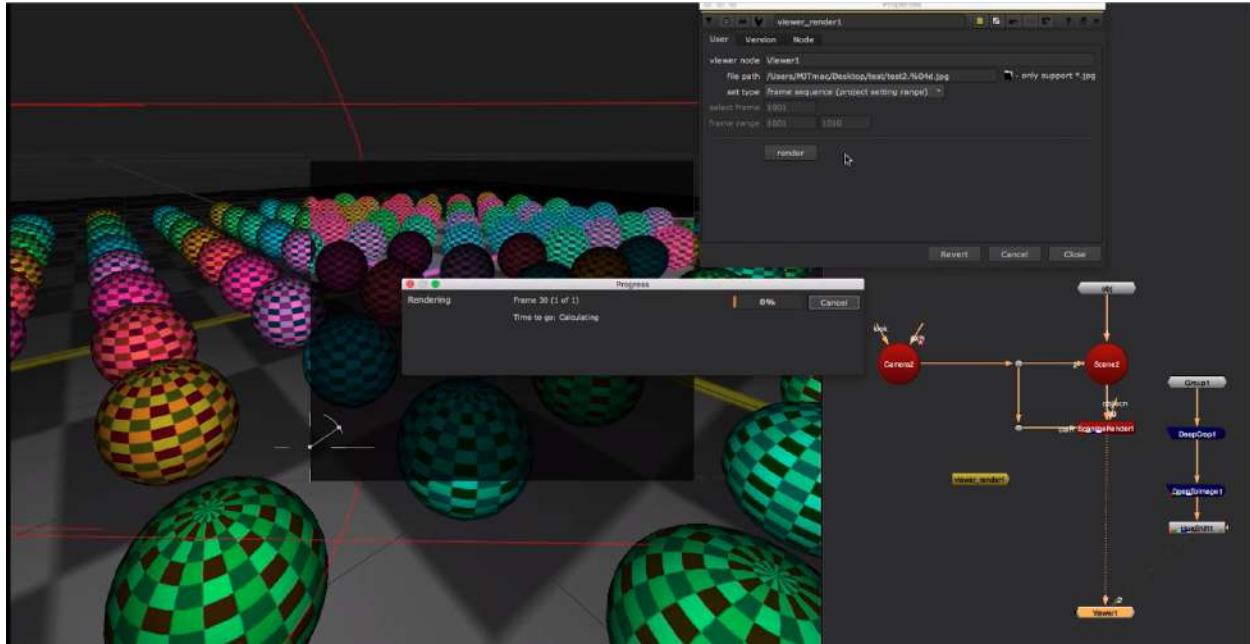


Tool Download page: <https://benmcewan.com/nukeTools.html>

Simple gizmo to help QC roto and keys, by overlaying a transparent colour, viewing a premultiplied image over grey or a checkerboard (for light and dark values).

ViewerRender MJT

Author: Mark Joey Tang - www.facebook.com/MJTLab



Nukepedia link: <http://www.nukepedia.com/gizmos/other/viewer-render>

Download Mark Joey Tang's entire Toolset: <http://bit.ly/menupy>

Viewer Render is a tool that can render anything shown in viewer, even with UI, like playblast in maya but in nuke. Support 2D and 3D, anything visible in the viewer. It's good to show a witness camera, while making tutorial video, show the problem in visual and bring it up for discussion.

If you run into any problems during render, please make sure the display card driver is up to date.

Demo video :

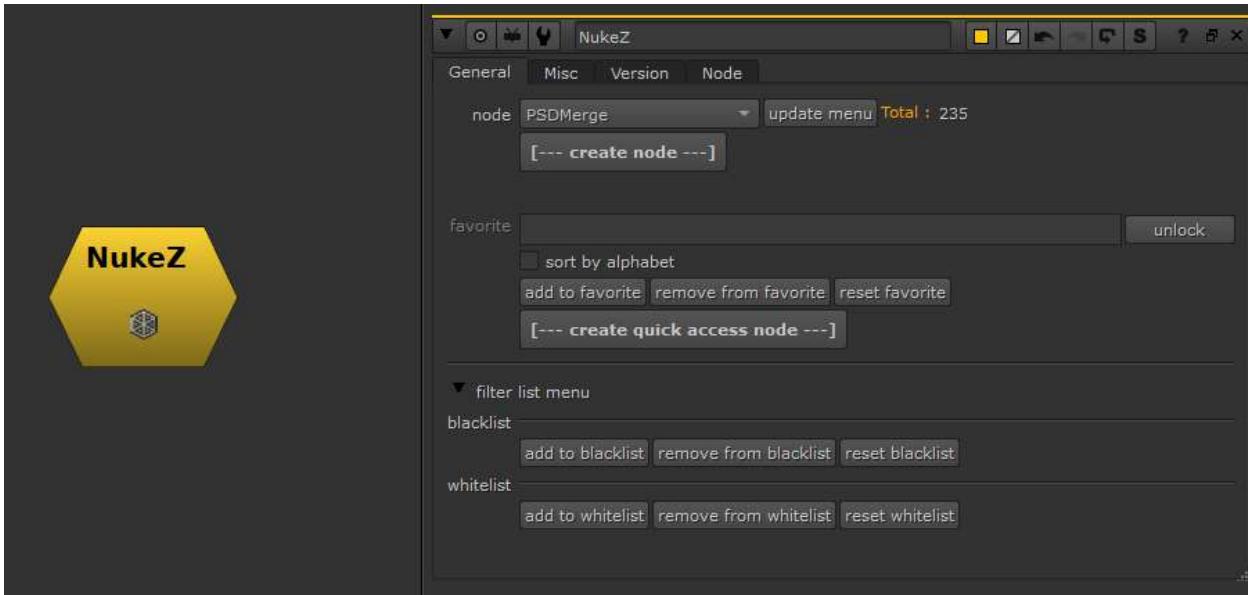
<https://www.facebook.com/MJTLab/videos/766383973741245/>

<https://vimeo.com/315590788>

Enter Viewer node name to render and set render path and frameRange and click render

NukeZ MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Download Mark Joey Tang's entire Toolset: <http://bit.ly/menupy>

NukeZ can access hidden / old version nodes quickly without memorize the class name.
Create personal favorite node set (including those hidden / old version nodes)

Click 'update menu' for your first run. It's because different versions of nuke has different node set.

e.g. ParticleBlinkscriptRender will give you error if pull it out on Nuke12.0 below.

After update the menu, you can simply select any nodes in the menu, and click 'create node' to pull it out.

You can also 'add to favorite' . Once you are done with the favorite list, click 'create quick access node', then it will create a node with all your favorite nodes in a single panel.

After that, you don't need NukeZ anymore, you can just save 'NukeZfav_xx' into your toolset for your personal use.

The tool actually detect all the file in plugin folder, there has a lot of node already exist in nuke's UI menu, so I created a filter function. The duplicated class will not show up in this tool. I only keep those with multiple versions and hidden ones.

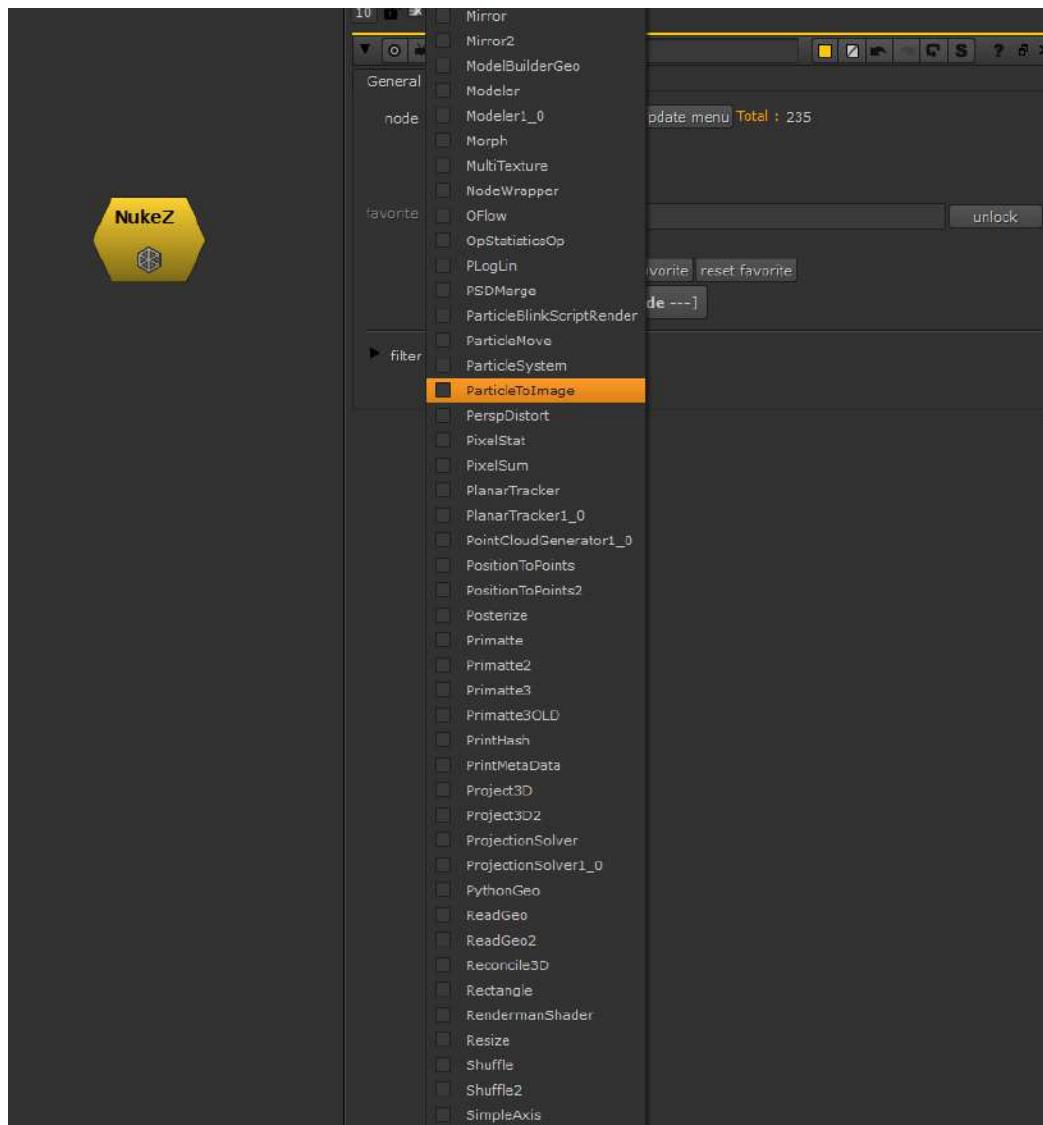
e.g.

you can find those 4 different versions of 'Tracking' (many people like to stick with the old one), 2 versions of lensdistortion (I also prefer the old one).

hidden node like 'ParticleToImage' (you can check the document from Mads website)

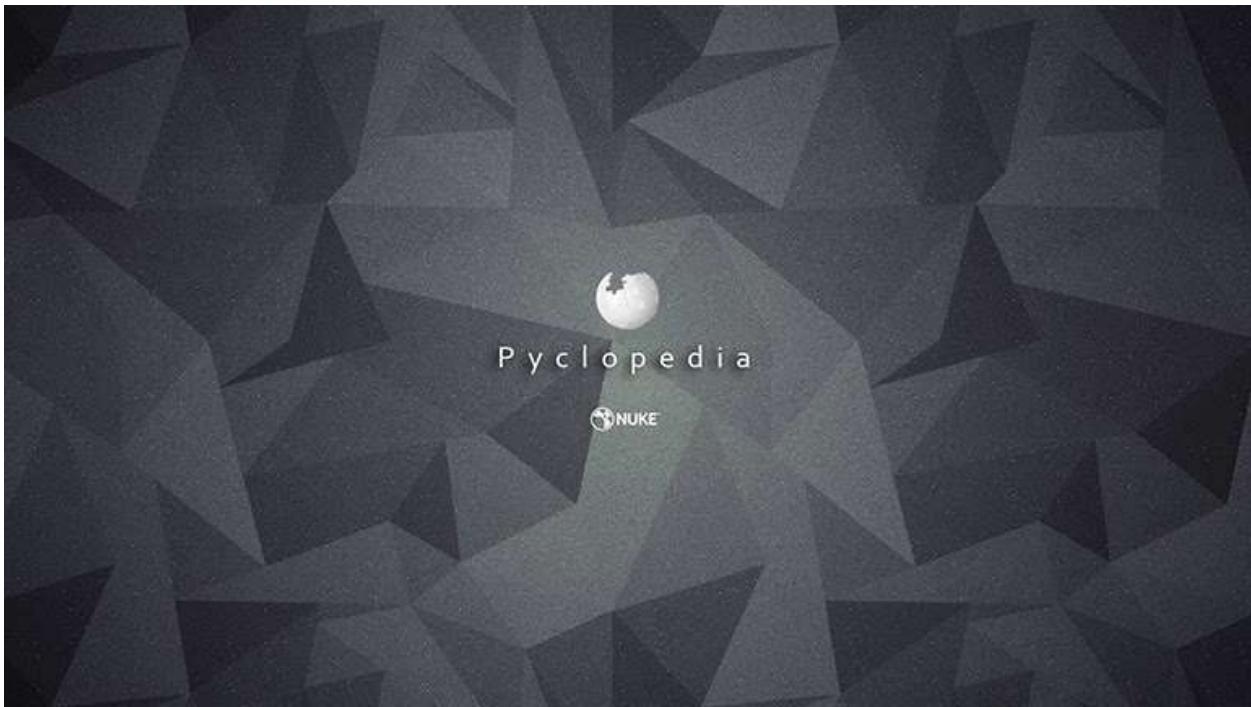
You can edit those filter manually and detect file type.

Whitelist is a list that doesn't find in the folder, so type in the node class manually to make it appear in the menu.



Pyclopedia MJT

Author: Mark Joey Tang - www.facebook.com/MJTlab



Nukepedia Link: http://www.nukepedia.com/gizmos/other/pyclopedia_v10

Download Mark Joey Tang's entire Toolset: <http://bit.ly/menupy>

Video :

<https://youtu.be/icLr4uo6F-c>

Check available python API and documents in Nuke with clear interface.

type in the command line you want to check.

click 'get next level' to see what command available.

click 'get details' to see what is the command doing, what argument it takes, how to use, etc.

In command line, can type in any level to get quickly search.

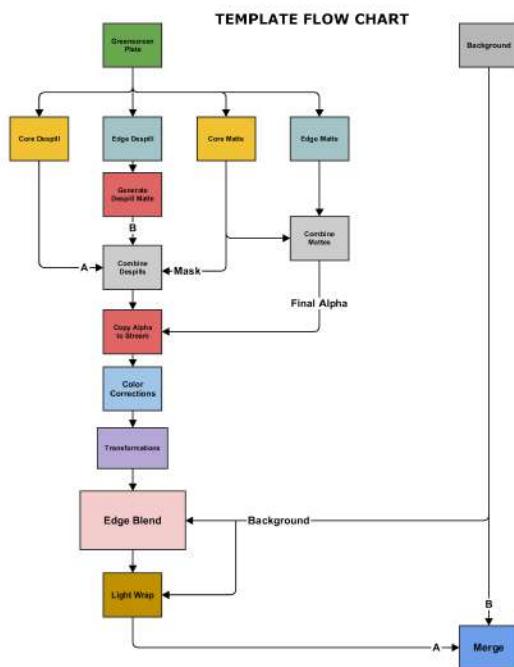
It also can check python from import any module.

In Pyclopedia, you can also point to a node and check what attributes can use.

16.) Templates

Advanced Keying Template TL

Author: Tony Lyons - www.CompositingMentor.com



Nukepedia Link: http://www.nukepedia.com/gizmos/keyer/advanced_keying_template_pack

Website Write-up:

<https://compositingmentor.com/2015/12/18/advanced-keying-breakdown-4-1-template/>

Video Tutorial on Template: <https://www.youtube.com/watch?v=BKcKpPFVmCk>

For in depth detail about how to use this template and different techniques and methods to look for in each section, please visit the full Keying Tutorial Playlist:

<https://www.youtube.com/playlist?list=PLt2Nu4KGXJ2iXe7s-ydCQ9u1tTzzApmjX>

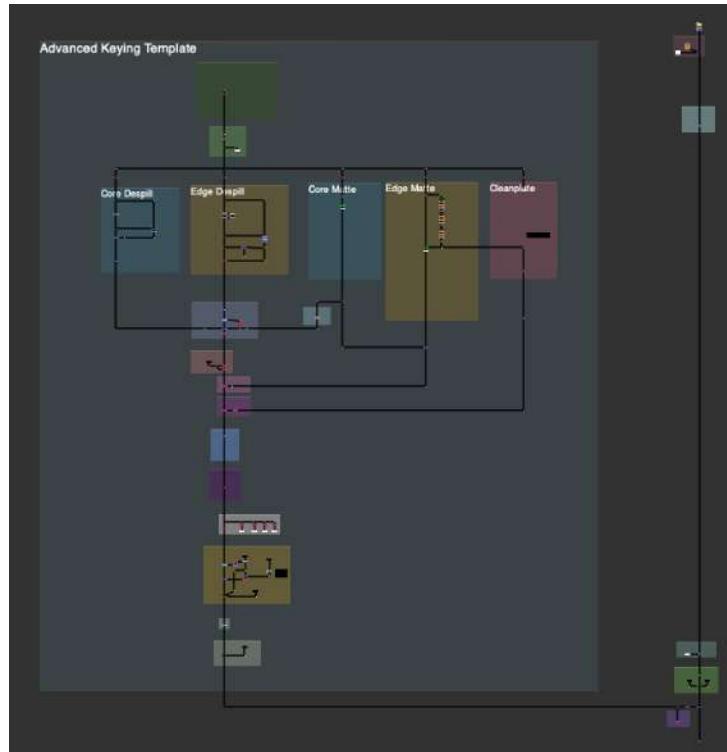
There are also more details on the website series:

<https://compositingmentor.com/category/advanced-keying-breakdown/>

Broken down into 5 sections, Core Despill, Edge Despill, Core Key, Edge Key, and the Background Blend. Organized to help with complex shots with transformations and color corrections to the Greenscreen element.

Advanced Keying Template Stamps TL

Author: Tony Lyons - www.CompositingMentor.com



This is an updated Advanced Keying Template using Adrian Pueyo's Stamps. You can download his stamps plugin here: <https://www.nukepedia.com/gizmos/other/stamps>

Mainly just changed the organization of the script while keeping true to the original template structure. There is a cleanplate section, which can be used later on with the additive keying section.

All passes are piped in before the Color Corrections and Transformations section, and piped out and converted to stamps after the transformations section, to be used elsewhere.

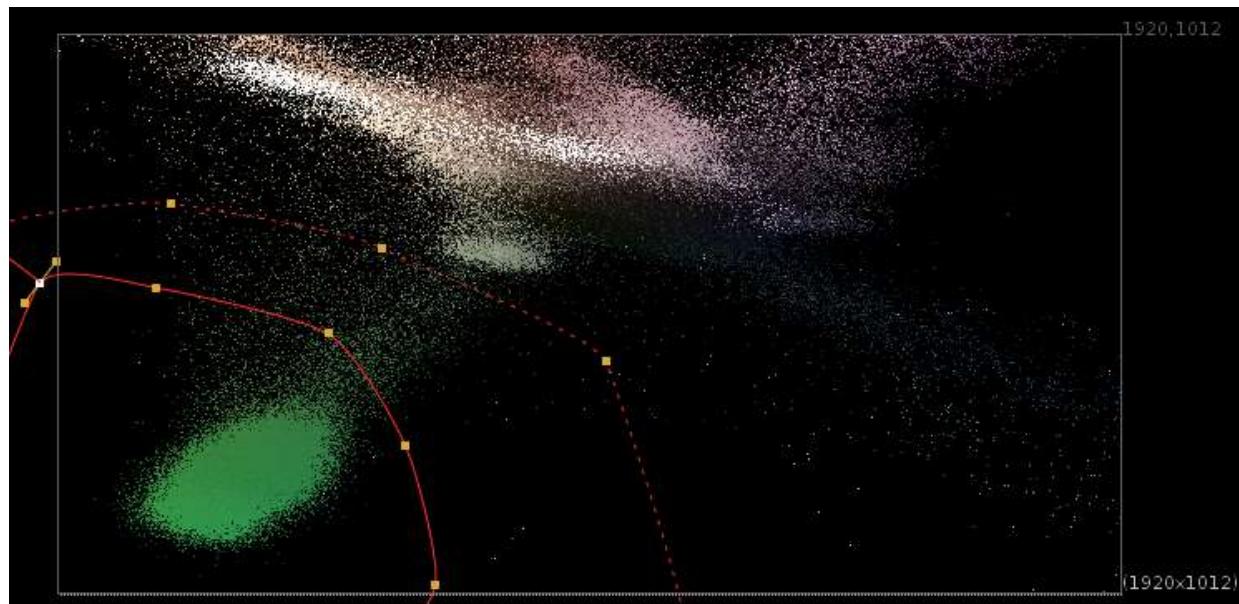
There is an updated, although experimental way of creating the cleanplate from ibk_Colour node. Feel free to use this or any other unpremult/blur or IBK stack, or edge extend method, or plug in your cleanplate if you have one. Feature was replaced by the exponBlurSimple node

There is a lightwrap section after the premult as well as an additive keying section near the merge.

If you don't like stamps, feel free to adjust/ add dots and pipelines and configure in whichever way you like to help your workflow

STMap Keyer Setup EL

Author: Erwan Leroy - <http://erwanleroy.com/blog/>



Great Write up about this cool STMap keying technique by Erwan Leroy:
<http://erwanleroy.com/make-a-custom-advanced-keyer-using-stmap/>

The idea of this method is to represent the image as a 2D point cloud (vs 1D points or 3D points above) and use some sort of shape to define which area should be transparent or solid, as well as softness. This is convenient because we have some great tools to define areas in 2d: Roto and Rotopaint, as well as a great tool to do 2d mapping: STMap.

Since we're in 2D, we can now use 2 channels as our 2 axes. Which 2 you would like to use is sort of up to you, and based on your specific footage. I will be using a plate from the Open Source project Tears of Steel, which you've probably seem many times, and can download here: https://media.xiph.org/tearsofsteel/tearsofsteel-footage-exr/02_3c/linear_hd/ For chroma keying, two channels that make sense for this approach are the cb and cr channels of the Ycbcr colorspace, though Luma and Hue could work, or Hue and Saturation, etc..

The basic implementation is rather simple: Set 2 channels as the STMap input of an STMap node, and a roto as the source.

Gizmo Demo Scripts

There are Demo Scripts graciously donated by the Authors to help demo their tools.
The following tools have Scripts in this menu, links to the tools in this document below:

[WaterLens Demo MJT](#)

[SSMesh Demo MJT](#)

[UV Editor Demo MJT](#)

[Sparky Demo NKPD](#)

[Particle Lights Demo MHD](#)

Special Thanks

I'd like to take this opportunity to thank the entire Nuke Compositing Community, and all of the various artists who have made and shared a gizmo online, and donated a piece of their knowledge, experience, and hard work to making this community better.

[Nukepedia](#) is an amazing website that many of us take for granted just how powerful it is for all of us to be able to gather, share, explore, and create new tools and techniques. Thank you to the founders of Nukepedia for providing over 10 years of gizmos, tutorials, and scripts for all of us to play with.

This Toolkit is in no way trying to "steal the thunder of", undermine, or "re-distribute" the various gizmo's through another website. I have done my best to document , link to, and credit every single tool and author to the best of my knowledge, and fully encourage anyone who wants to know more to explore their websites, and click on the links in the tools to check for updates and more details and support the authors of the tools. My goal was to simply bring the tools together in a single organized menu, so that anyone could download, install, and access a library of tools at their fingertips. Full credits belong to the authors; this toolkit would not exist without their contribution.

Also, this toolkit is just a small collection of the thousands of tools available on nukepedia and online. It is not a reflection of "the best" tools. The tools selected on this menu are subjectively picked by myself because I found them useful and unique. There are many tools that I either missed, overlooked, or simply tried to condense the categories, but are no doubt just as valuable or more valuable to another artist out there. I encourage you to explore more tools and not to take this as an "end all be all". Some tools will fit your workflow better than someone else's workflow, and it's up to you to find your "go-to" tools for a compositing situation. You may consider this a "rough guide" or good starting point in your search. And many in here you may not personally find a use for.

In particular, I'd like to personally thank a few of my favorite contributors:

Frank Rueter
Erwan Leroy
Xavier Martin
Mads Hagbarth Damsbo
Mark Joey Tang
Wouter Gilsing
Ben McEwan
SPIN VFX
Andrea Geremia
Alexey Kuchinski
Derek Rein
Fynn Laue

And many others for all of their awesome work.

And a very special thank you **Adrian Pueyo**, for helping come up with the vision for this Toolkit with me back in 2019, for allowing me to post all of his amazing tools in this toolkit, for teaching, inspiring, and encouraging me to get better at coding and gizmo creation to the point where I am confidently able to deliver to everyone the Nuke Survival Toolkit.

You all drive me to be a better artist, and I am very honored to be part of this community with you. Thank you.

Credit for Swiss Army Knife Icon: <https://icons8.com/icon/2532/swiss-army-knife>

Contact

If you are having any trouble with the Nuke Survival Toolkit, or would like me to remove a certain tool, or think I need to update or include a particular tool, please feel free to contact me via email.

You can reach me by filling out a brief contact form on my contacts page, found here:
<https://www.creativelyons.com/contact>

Please include “Nuke Survival Toolkit” or “NST” in the subject field to help me prioritize the mail in my inbox. I appreciate your feedback.

If you’d like to view my portfolio website, please visit:
www.CreativeLyons.com

For more tutorials and tools, please visit my compositing blog:
www.CompositingMentor.com

Thank you, and please enjoy the Nuke Survival Toolkit.

Tony Lyons | 09/2020