

G'MIC

X

Python

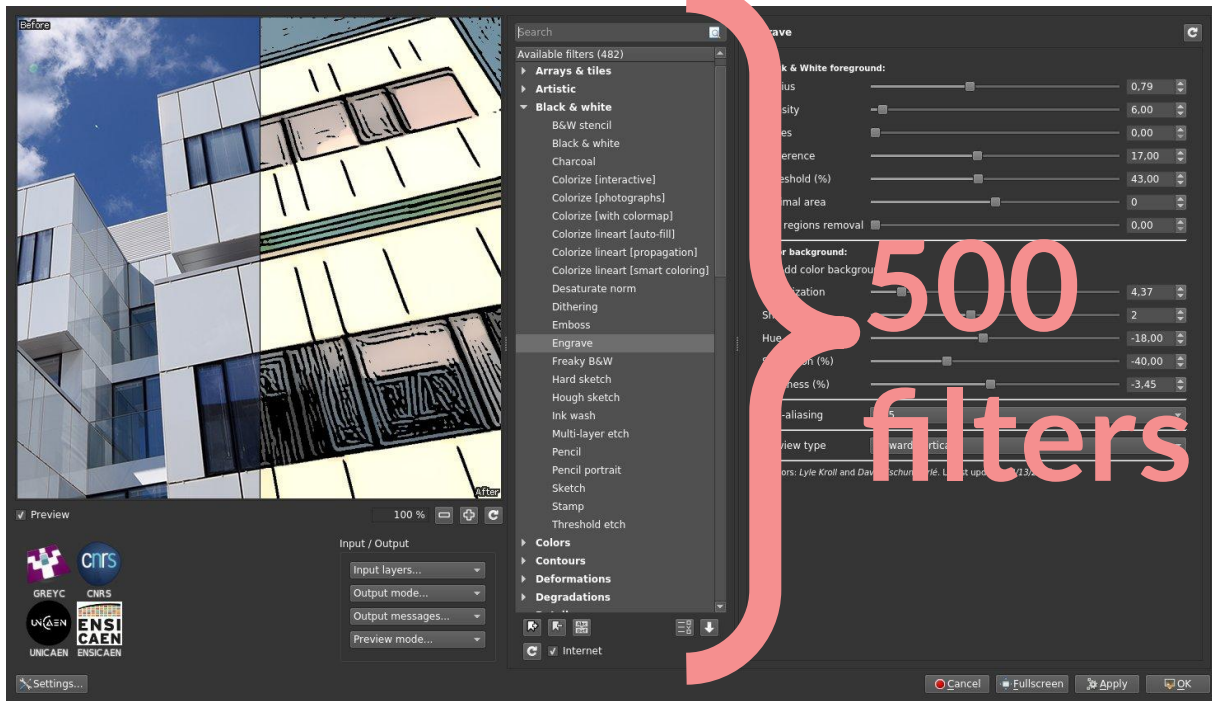
X

Numpy

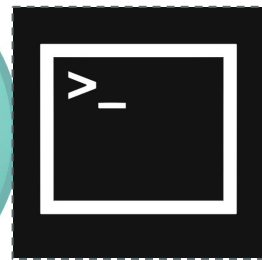
X

Blender

G'MIC in 30 seconds



500 filters



GOOD NEWS

FUNDING FOR 1 PYTHON DEVELOPER FOR 1 YEAR for G'MIC



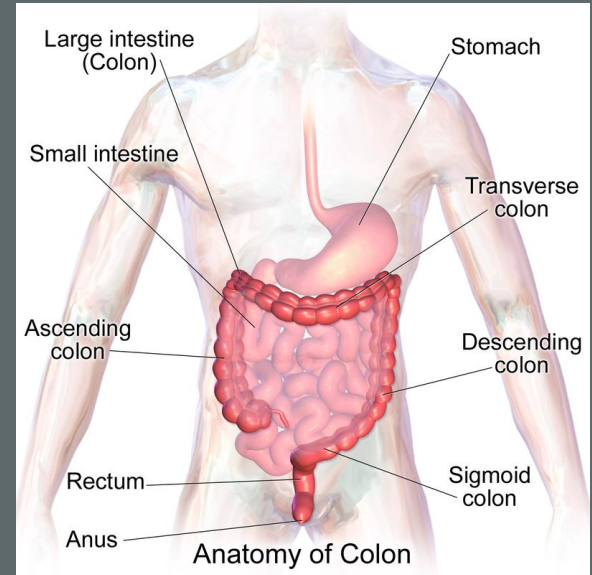
I - Into the guts of gmic-py



X

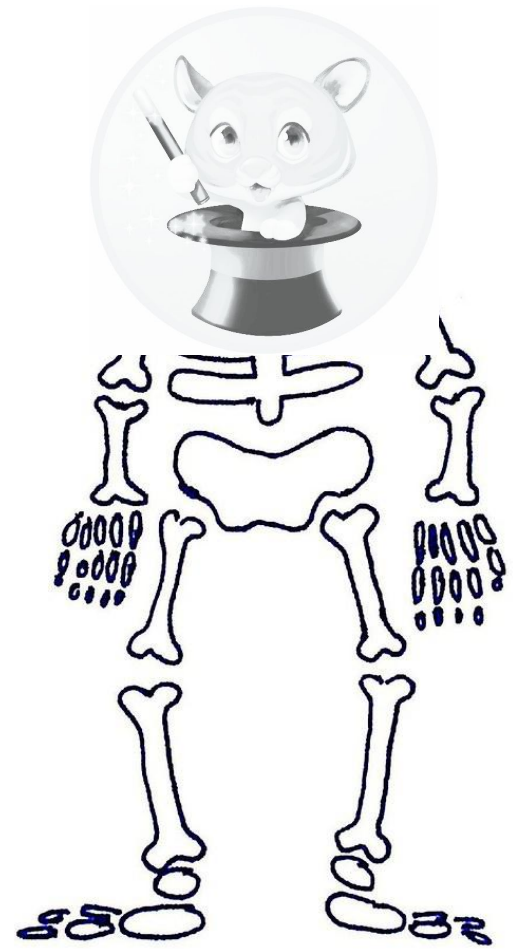


X



G'MIC IS

- 1 PACKAGE
- YOUR IMAGES
- 1 LANGUAGE
- 1 INTERPRETER



G'MIC C/API Python binding

G'MIC C++

```
wget gmic29x.tar.gz ; make
```

```
#include <gmic.h>
```

```
image = new gmic_img();
```

```
images = new gmic_list(...);
```

```
gmic(images, "blur")
```

G'MIC Python

```
pip install gmic
```

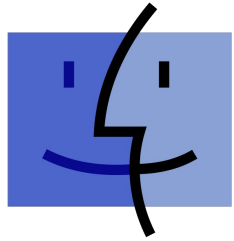
```
import gmic
```

```
i = gmic.GmicImage()
```

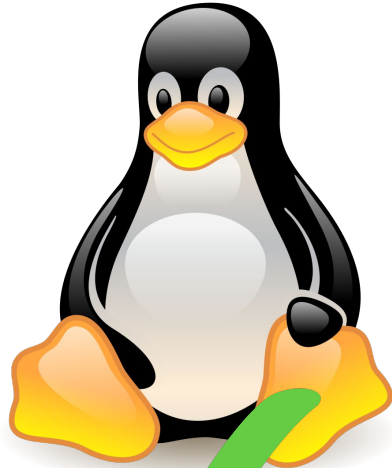
```
l = [i, i, i]
```

```
gmic.Gmic().run("blur", l)
```

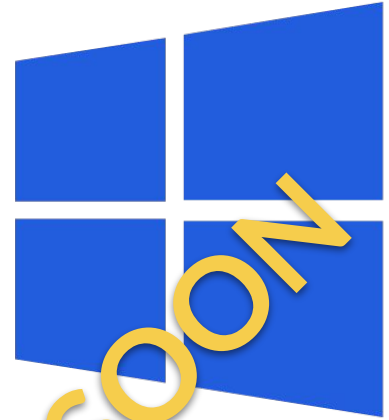
OS support (64bits mostly) for Python 3.5+



Mac OS



colab



Windows

With:
brew install libomp

FFTW OpenMP

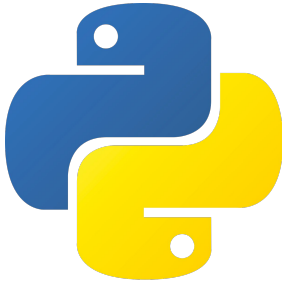
curl



gmic-py demo



- level 1

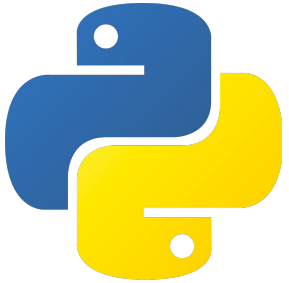


Installing, reading and writing images with 1 command, 1 filter

gmic-py demo



- level 2



Real-life example - processing.js game map



gmic-py demo

- level 3



Massive headless testing of

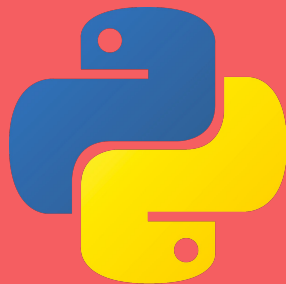
G'MIC filters using filters290.json



GitHub

ALPHA
SUPPORT

II - G'MIC x Numpy





G'MIC +NUMPY MUST-KNOWS FOR DATA-SCIENTISTS

Grace Hopper & UNIVAC
CC Unknown (Smithsonian Institution)

1. From gmic-py 2.9.1:
builtin numpy support

Install numpy yourself



2. GmicImage ↔ numpy.ndarray

```
GmicImage(numpy_array)
```

```
GmicImage.from_numpy_array()
```

```
GmicImage.to_numpy_array()
```

3. G'MIC works in float32 deinterleaved

Use GmicImage (de)interleave flags
Use ndarray.astype()

RGBRGBRGB



RRRGGGBBB

1-64bit bool-int-float

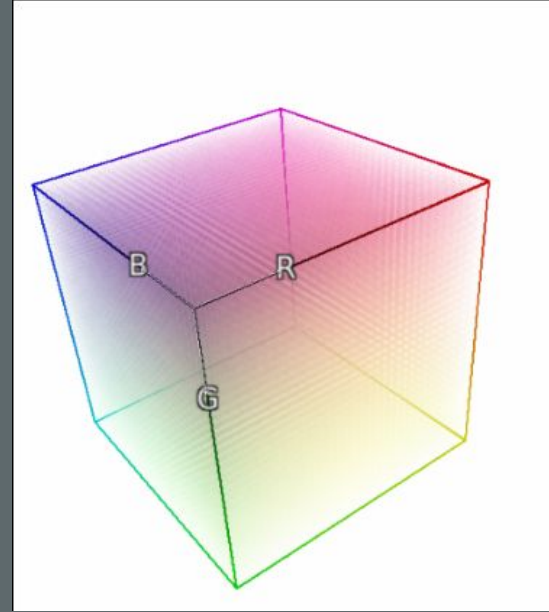


float32

4. G'MIC images are
1D, 2D, 3D

X

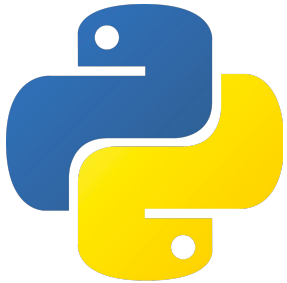
~2 billion channels



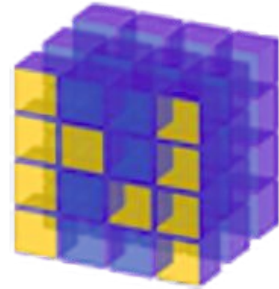
numpy demo



Using Pillow input & output with deinterleaving



pillow



III - G'MIC

X



Anthology of G'MIC and Blender



2017 - *Blender-2-G'MIC*

By StarfallRobles

Blender 2.76 - no patch required

2016 - VSE filters (gmic-cli)

G'MIC GTK widget (gmic-cli)

G'MIC Qt widget


Cross-platform

Demonstration of Blender with G'MIC-Qt on Linux

Blender* [/home/voyim/MESMER/Mesmer6.blend]

File Render Window Help Video Editing gmicWorld Blender Render v2.78 | Verts:0 | Faces:0 | Tris:0 | Objects:0/2 | Lamps:0/1 | Mem:66.06M

G'MIC-Qt - Linux 64 bits - 2.0.3_pre#170803



Search

Available filters (485)

- Blockism
- Bokeh
- Brushify
- Cartoon
- Chalk it up
- Circle abstraction
- Color abstraction paint
- Colored pencils**
- Cubism
- Cutout
- Diffusion tensors
- Dream smoothing
- Ellipsionism
- Felt pen
- Finger paint
- Fractalize

Colored pencils

Size: 1.30

Amplitude: 50.00

Quantize colors: 20

Color smoothness: 2.00

Mixer mode: Grain Merge

Color intensity: 1.00

Preview type: Full

Author: PhotoComiX. Latest update : 2010/29/12.

Preview 100%

Input / Output

- All
- In place (default)
- Verbose (log file)
- Preview mode...

Settings...

Blender Cancel Fullscreen Apply OK

2017-*Blender-custom-nodes G'MIC*

By bitsawer

Blender 2.79 - C++ PATCHED

Compositing nodes - 1 G'MIC node

Based on libcgmic

Windows x64 build only

Blender compositor with a custom G'MIC node

v2.79 | Vets: 7,966 | Faces: 7,878 | Tris: 15,756 | Objects: 1/4 | Lamps: 1/1 | Mem: 64.92M | Lamp

The compositor setup includes a 'Render Layers' node connected to an 'Image' node. The 'Image' node is connected to a 'G'MIC' node. The 'G'MIC' node has the following parameters: Quality: 1.000, -water 30,1.5, Normalize (checked), and five \$arg nodes (1.000). The 'G'MIC' node is connected to a 'Composite' node with 'Use Alpha' checked and 'Z' set to 1.000.



Example G'MIC effects:

- cartoon ,
- water 30,1.5
- pencilbw 0.3
- sketchbw ,
- fx_painting 3,1.5,2
- erode 10 --erode 3 --equalize 200 -smooth 80,0,1,1,3 -n 0,1 -negate
- rodilius ,
- rodilius 12,10,5,10
- drawing 200
- taquin 5
- normalize_local 80,6,2
- to_rgba -rotate_tiles 15,5,\$arg1
- transform_polar R-r

Properties panel: Name: Image.001, Label: Image, Use Alpha checked, Image: Image, Alpha: Alpha, Z: 1.000.

Render Presets: Resolution: X: 1280 px, Y: 720 px, Frame Range: Start Frame: 1, End Frame: 250, Frame Step: 1, Aspect Ratio: X: 1.000, Y: 1.000, Frame Rate: 24 fps, Time Remapping: Bar, Cro, 100% x 100%.

Hopes for G'MIC and Blender

The background of the image is a vibrant blue sky filled with numerous white, fluffy clouds of varying sizes and densities. The clouds are scattered across the frame, creating a bright and airy atmosphere. The text is overlaid on the left side of the image.

2020 - *gmic-blender*

WORK IN
PROGRESS

By GREYC IMAGE / myselfhimself

Blender 2.8x - no patch

Stand-alone add-on

Embeds gmic-py - `import gmic`

500 compositing nodes?

Linux (MacOS, Windows soon)

blender demo

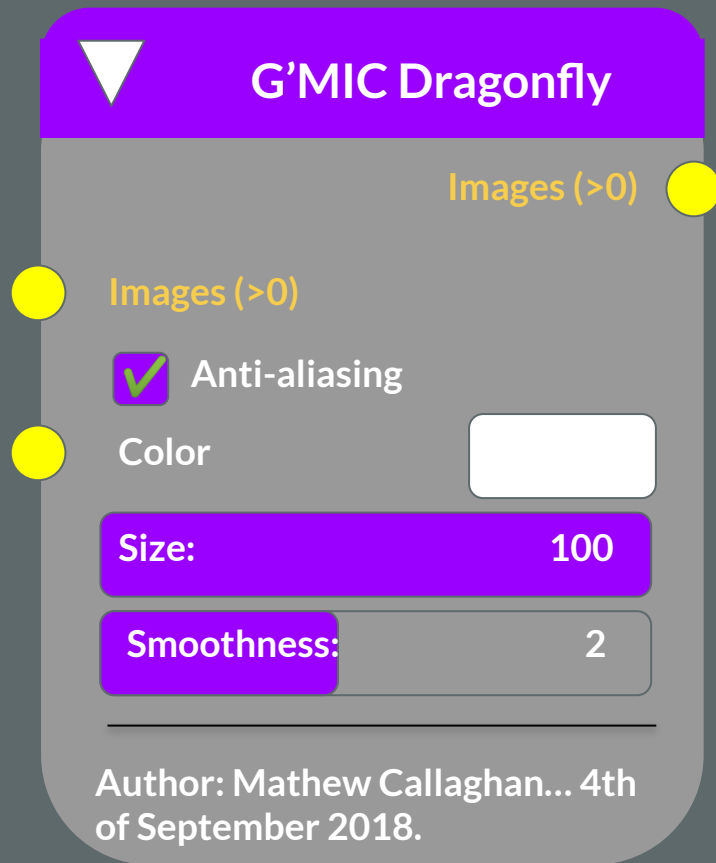
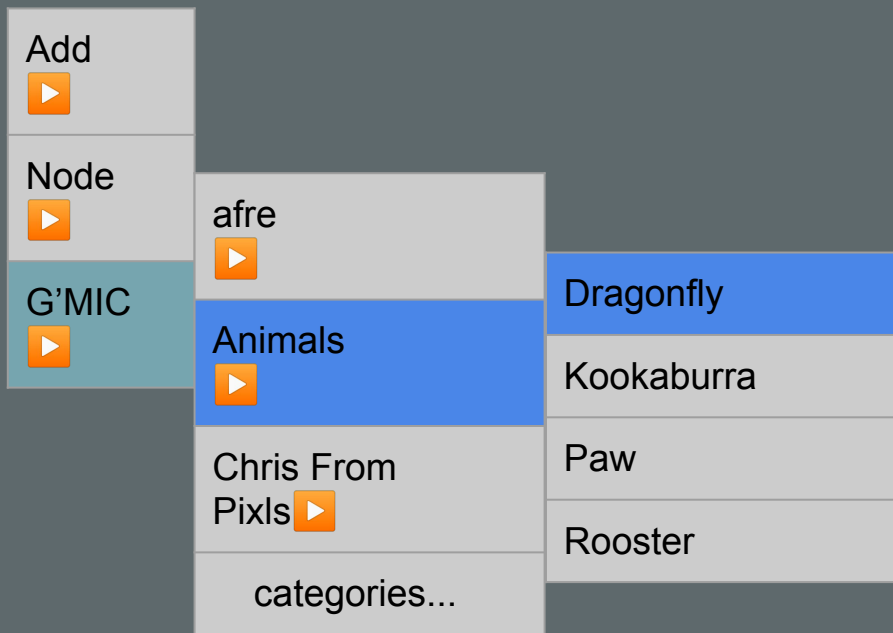


Gmic-blender add-on + using gmic-py in the Text editor



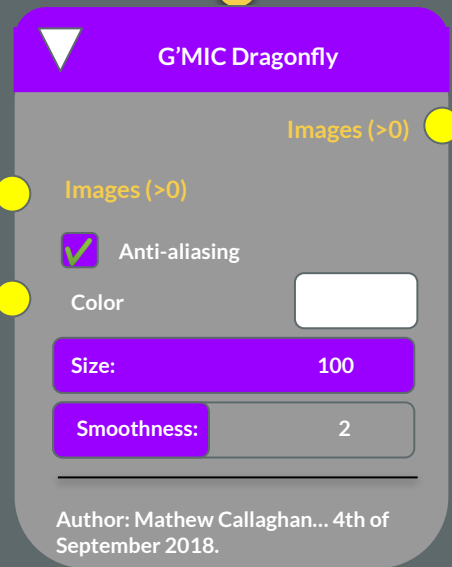
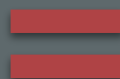
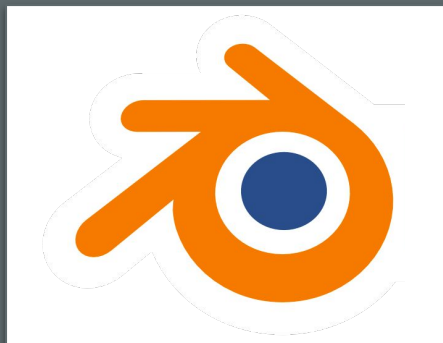
Compositing nodes UI wish

500 nodes in subcategories



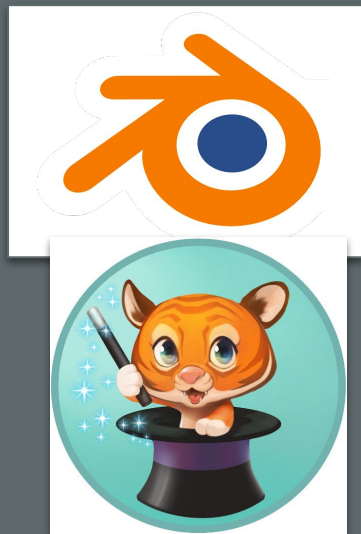
gmic-blender architecture (A)

NO
BLENDER
C++ PATCH



gmic-blender architecture (B)

EMBED
LIBGMIC INTO
BLENDER

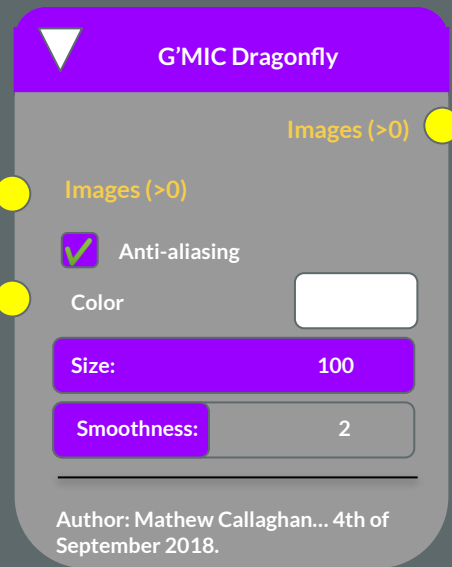


Bundled
ADD-ON

G'MIC
Python UI
generation
only



Freestyle-like?



DREP ENSICAEN / Normandie Incubation, R
 olein, Winkelcentrum Boven 't IJ, Amsterdam,
 Car (OSRM) Go
 Reverse Directions

Directions

Distance: 678km. Time: 7:24.

1. Start on Rue Claude Bloch 50m
2. At roundabout take 2nd exit onto unnamed road 40m
3. Exit roundabout onto unnamed road 10m
4. Continue on Rue Claude Bloch 140m
5. Continue on Rue Professeur Joseph Rousselot 200m
6. At roundabout take 1st exit onto Avenue du Professeur André Maurice 10m
7. Exit roundabout onto Avenue du Professeur André Maurice 140m
8. Turn left onto Avenue de la Côte de Nacre 300m
9. At roundabout take 3rd exit onto N 814 200m



Thank you!

gmic.eu



myselfhimself