

G'MIC 3.4.0 : Le traitement d'images à la fleur de l'âge

Posté par [David Tschumperlé \(site web personnel\)](#) le 17 juin 2024 à 23:57. Édité par **4 personnes**. Modéré par [Pierre Jarillon](#). [Licence CC By-SA](#).
Étiquettes : aucune + [Étiqueter](#)

Une nouvelle version **3.4.0** de [G'MIC](#) (*GREYC's Magic for Image Computing*) vient de sortir !

À cette occasion, nous vous proposons une description des fonctionnalités récentes ajoutées à ce [cadriciel](#)^w libre pour [le traitement des images numériques](#)^w, faisant suite à notre [précédente dépêche](#) sur ce sujet (publiée en mai 2023).



N. D. A. : Cliquez sur les images pour en obtenir une version en pleine résolution, ou une vidéo correspondante lorsque les images contiennent l'icône



Sommaire

- [1. G'MIC en quelques mots](#)
- [2. Quoi de neuf dans cette version 3.4.0 ?](#)
 - [G'MIC Online \(G'MICol\)](#)
 - [Greffon G'MIC-Qt](#)
 - [Outil CLI `gmic`](#)
 - [Autres améliorations](#)
- [3. Détails des nouveaux filtres](#)
 - [3.1. Retouche photographique](#)
 - [3.2. Filtres de déformations](#)
 - [3.3. Filtres de dégradations](#)
 - [3.4. Filtres de rendu et de textures](#)

- [3.5. Effets artistiques](#)

- [4. Un logiciel aux utilisations variées](#)
- [5. Conclusion](#)

1. G'MIC en quelques mots

[G'MIC](#) est un cadriciel (*framework*) libre pour la manipulation et le traitement des [images numériques](#)^W, développé au sein de l'équipe [IMAGE](#) du laboratoire de recherche [GREYC](#) de Caen ([UMR CNRS](#)^W 6072).

Il définit différentes interfaces utilisateur permettant l'application d'algorithmes variés sur des images et des signaux, variés eux aussi. L'élément moteur du projet est un interpréteur d'un langage de script, le « [langage G'MIC](#) », élaboré spécifiquement pour faciliter le prototypage et l'implémentation de nouveaux algorithmes et opérateurs de traitement d'images. Les utilisateurs peuvent ainsi appliquer des opérateurs parmi plusieurs centaines déjà implémentés, ou écrire leurs propres pipelines de traitement et les rendre accessibles dans les différentes interfaces utilisateur du projet. C'est donc, par essence, un cadriciel ouvert, extensible et en évolution constante.

Les interfaces de G'MIC les plus populaires sont : [gmic](#), outil en ligne de commande (complément indispensable à [ImageMagick](#) ou [GraphicsMagick](#) pour traiter/générer/analyser des images à partir du *shell*), le service Web [G'MIC Online](#), et surtout, le greffon [G'MIC-Qt](#), utilisable dans de nombreux logiciels d'édition d'images numériques tels que [GIMP](#), [Krita](#), [DigiKam](#), [Paint.net](#), [Adobe Photoshop](#)^W, [Affinity Photo](#)^W... Ce greffon propose aujourd'hui plus de **620 filtres** divers, pour élargir les possibilités de ces logiciels de manipulation d'images.

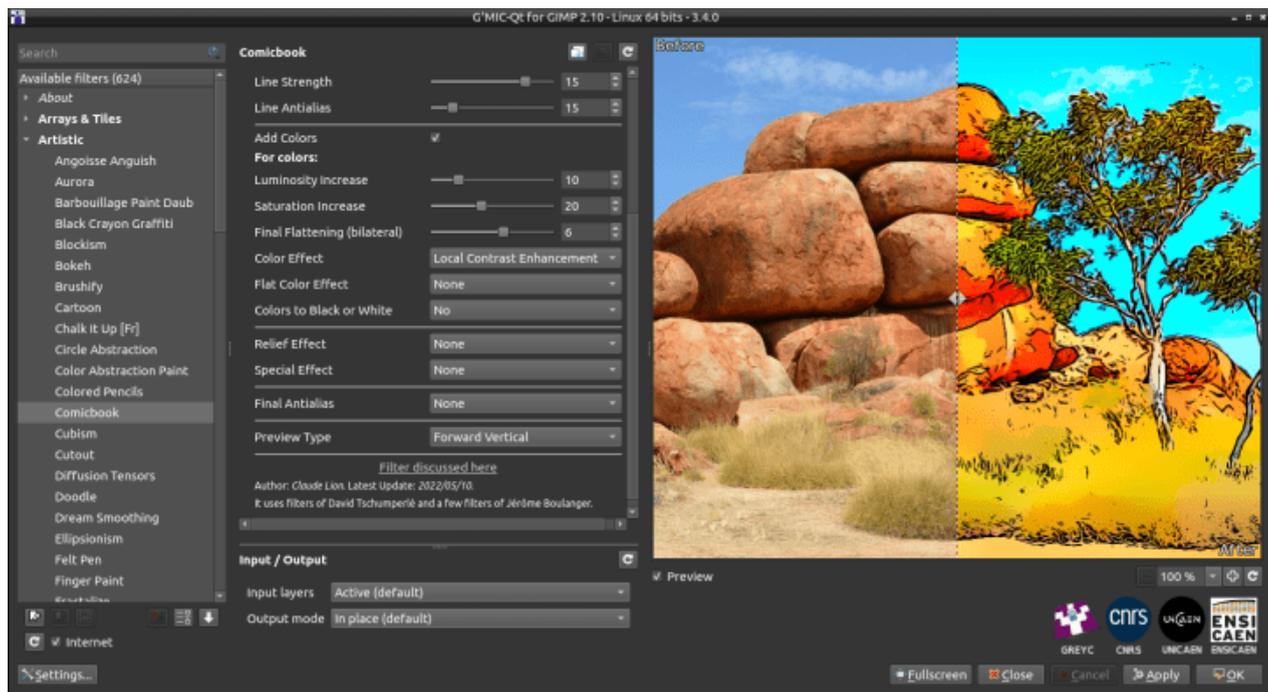


Fig. 1.1. Aperçu du greffon G'MIC-Qt en version **3.4.0**, ici lancé depuis GIMP 2.10, avec le filtre « [Comicbook](#) » sélectionné.

2. Quoi de neuf dans cette version 3.4.0 ?

La version **3.4.0** de G'MIC_ se focalise sur la *stabilité* et le support *long-terme* : Après plus de 15 ans de développement continu du projet, les concepts et codes qui constituent G'MIC ont eu le temps d'être éprouvés, et nous voulons maintenant assurer la stabilisation des *API* des différents composants du projet : l'*API* de la [libgmic](#) (pour intégrer les fonctionnalités de G'MIC dans un code C ou C++), celle du greffon [G'MIC-Qt](#) (pour implémenter une version du greffon au sein

d'un nouveau logiciel hôte), celle des fonctions de base définissant la [bibliothèque standard de G'MIC \(stdLib \)](#), ainsi que la syntaxe du [langage G'MIC](#) lui-même.

Ainsi, toutes les versions de cette branche 3.4.x seront consacrées à la correction de bugs et le développement de fonctionnalités ne nécessitant pas de modification de ces fondations du projet (par exemple, le développement de nouveaux filtres ou d'opérateurs de traitement d'images), avec le souhait d'assurer au maximum une rétro-compatibilité entre ces versions. En ce sens, on considère que la version 3.4.0 représente un jalon dans l'histoire du projet.

En ce qui concerne les améliorations générales apportées aux différentes interfaces utilisateurs, on peut citer principalement :

G'MIC Online (G'MICol)

Ce service, qui permet d'appliquer les filtres de G'MIC sur des images, directement à partir d'un navigateur web, existe depuis plusieurs années. En 2024, les membres du service [DDA](#) (Développement et Déploiement d'Applications) du laboratoire GREYC ont initié une refonte totale de ce service, avec à la clé un nouveau look, une amélioration nette de l'interface utilisateur (*widgets* plus intuitifs, amélioration de la fenêtre de prévisualisation, apparition de thèmes clair/sombre, gestion des filtres favoris...) et une automatisation des mises à jour de G'MIC et des filtres associés. La liste des améliorations réalisées serait trop longue à énumérer, et une figure valant mille mots, voici à quoi ressemble aujourd'hui la nouvelle version de ce service web :

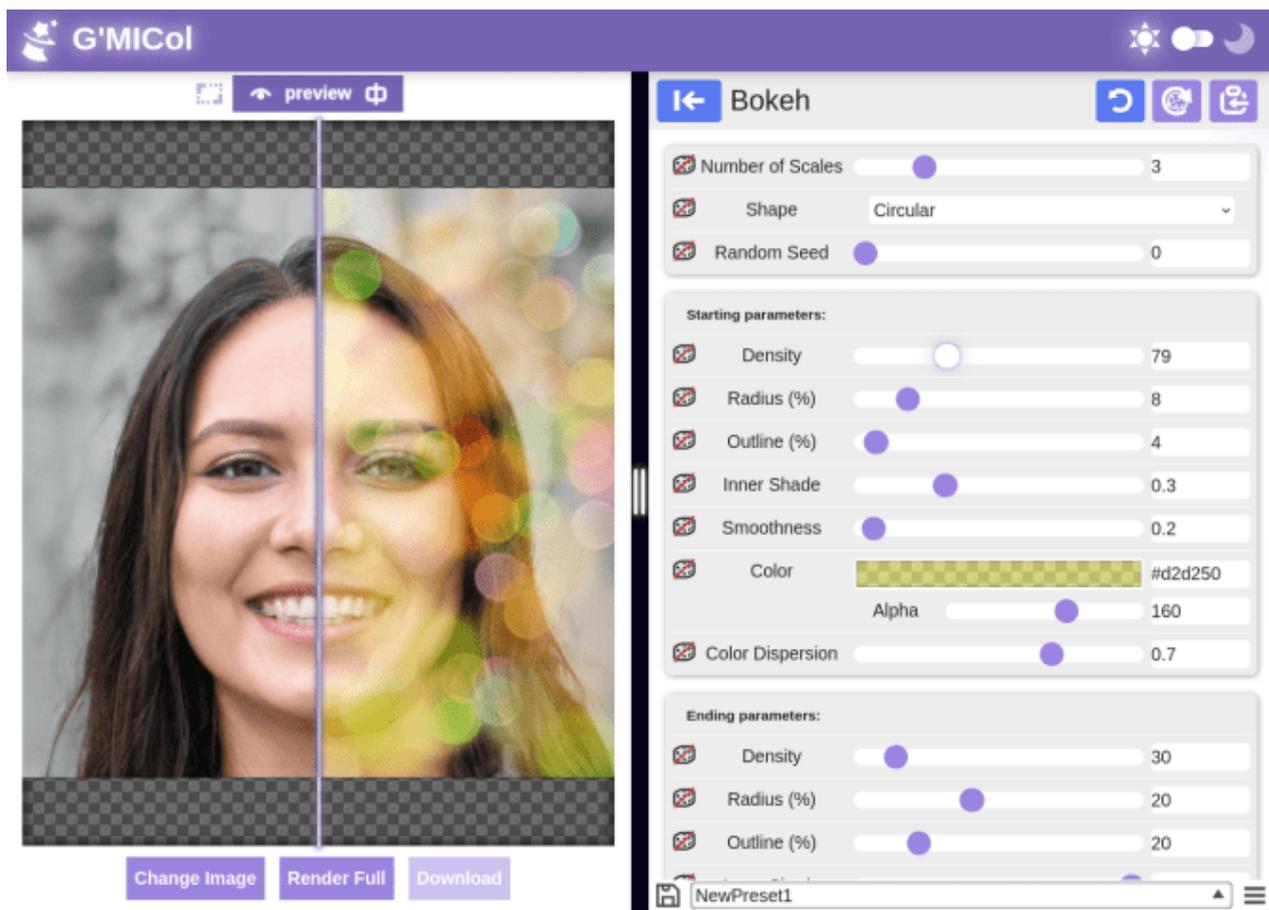


Fig.2.1. Aperçu du service web G'MIC Online, remis au goût du jour à l'occasion de la sortie de la version 3.4.0 de G'MIC (ici, avec le thème clair).

Si vous souhaitez l'essayer, foncez voir [G'MIC Online](#), ça sera pour nous l'occasion d'avoir un test grandeur nature 🙌 ! Et n'hésitez pas à nous faire remonter des soucis éventuels. Dans tous les cas, un grand bravo 🙌 à nos collègues de l'équipe DDA pour ce travail important de refonte !

Greffon *G'MIC-Qt*

En l'espace d'une année, ce sont plus de **40 nouveaux filtres** de traitement d'images qui ont fait leur apparition dans le greffon. La plupart seront détaillés dans la suite de cette dépêche. Mentionnons également l'apparition dans l'interface, d'un nouveau bouton « **Paramètres aléatoires** », qui assigne des valeurs aléatoires aux paramètres d'un filtre sélectionné, ce qui permet d'obtenir un aperçu rapide de la variété de rendus possibles pour un filtre donné.

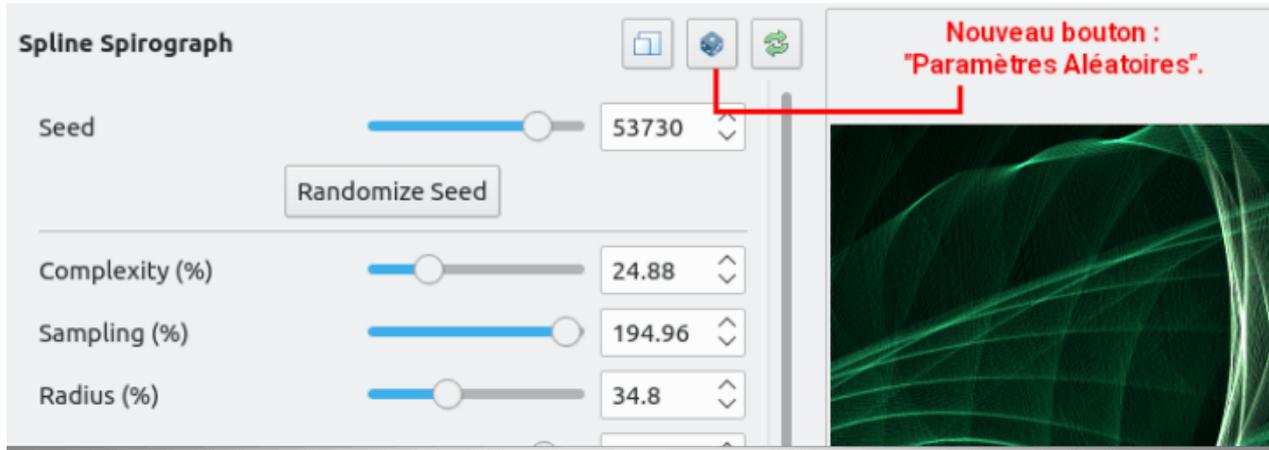


Fig.2.2. Greffon *G'MIC-Qt* : Le nouveau bouton « Paramètres aléatoires » assigne des valeurs aléatoires aux paramètres du filtre sélectionné.

Outil CLI `gmic`

L'ensemble des filtres réalisés par la communauté des développeurs est maintenant inclus dans l'exécutable dédié à la ligne de commande, ce qui fait qu'il n'y a plus de différences concernant le nombre de filtres disponibles par défaut entre le greffon *G'MIC-Qt* et l'outil CLI `gmic` (il fallait auparavant forcer la mise à jour des filtres avec `$ gmic update`). Aujourd'hui, `gmic` dispose d'un arsenal de plus de **4000 fonctions** pour traiter vos images en ligne de commande. Notons que des efforts particuliers ont été faits pour améliorer le visualiseur intégré d'images (commande `display`). On peut dorénavant visualiser plus aisément des collections d'images 2D, des images volumiques 3D ou même des maillages 3D, dans une unique interface, comme illustré sur la figure ci-dessous :



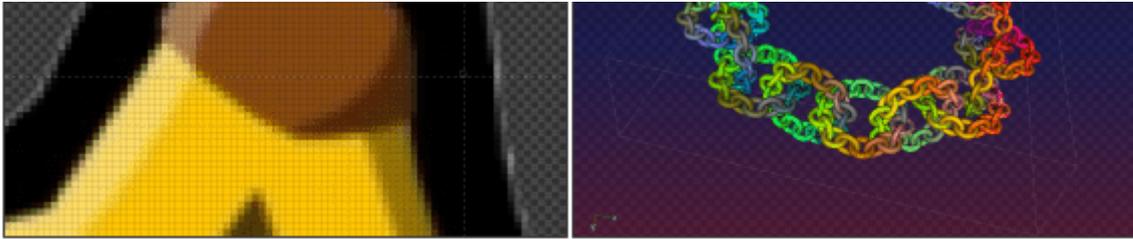


Fig.2.3. Outil CLI `gmic` : Le visualiseur d'images de G'MIC a été entièrement ré-implémenté et permet de visualiser des types d'images différents au sein d'une même interface.

C'est également le cas pour le visualiseur de fonctions mathématiques ou de signaux 1D (commande `plot`), qui a été refait à neuf :

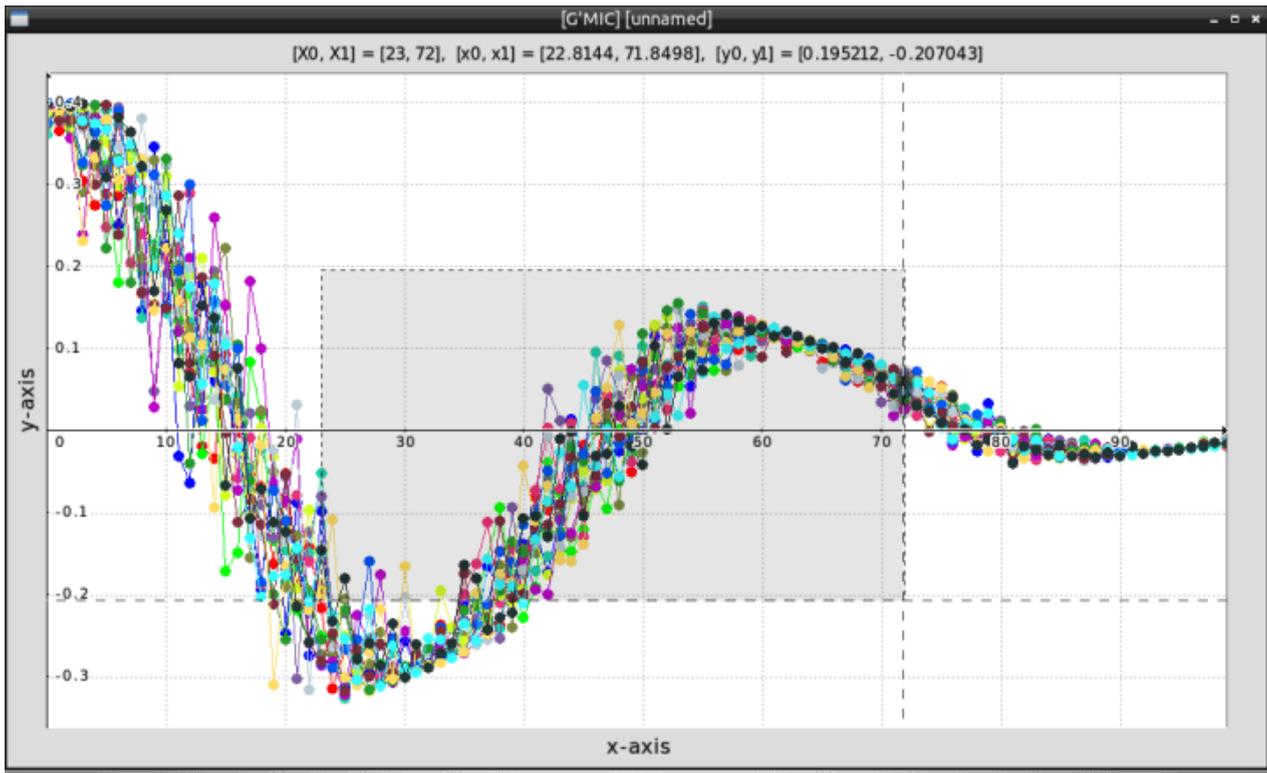


Fig.2.4. Commande `plot` : Le visualiseur de fonctions/signaux 1D a été entièrement repensé.

Tout ceci fait de `gmic` un outil bien pratique, même si on ne se limite qu'à la visualisation d'images en ligne de commande.

Autres améliorations

- L'évaluateur intégré d'expressions mathématiques a encore bien évolué : c'est l'une des briques essentielle du langage G'MIC, puisqu'il est en charge d'évaluer le résultat des opérations mathématiques dont on a besoin dans les scripts (autant dire qu'en traitement d'images, on fait tout le temps des calculs). Cet évaluateur s'est enrichi de nombreuses fonctions, en particulier pour le calcul et le traitement de [vecteurs^W](#), de [matrices^W](#), et la manipulation native de [tas^W](#).
- De nombreuses fonctions natives (C++) de la bibliothèque ont pu être ré-écrites intégralement en langage G'MIC. Ceci va faciliter l'évolution et la maintenance de ces fonctionnalités, puisque leur modification/amélioration future ne nécessitera plus de mise à jour obligatoire des binaires du projet.
- Les commandes de chargement/sauvegarde de maillages 3D en format `.obj` (*Wavefront*) ont été améliorées, et de nouvelles commandes de traitement/création de maillages 3D ont vu le jour (par exemple pour visualiser les vecteurs normaux aux sommets d'un

maillage, comme illustré sur la vidéo ci-dessous). Et, non, ceci n'est pas une pomme pourrie :

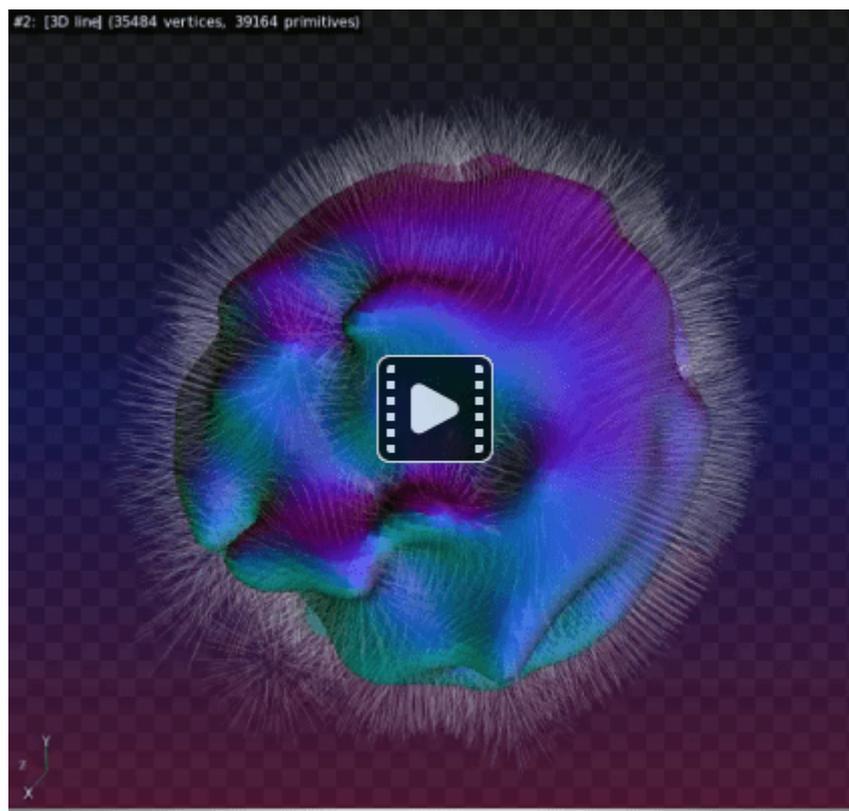


Fig.2.5. La gestion des maillages 3D au sein de G'MIC a été enrichie de nouvelles commandes.

Voilà pour ce qui concerne les améliorations générales des différentes interfaces proposées par le cadriciel.

Passons maintenant au détail des nouveaux filtres et traitements d'images apparus dans G'MIC ces douze derniers mois.

3. Détails des nouveaux filtres

Dans cette (longue) section, nous catégorisons et décrivons les filtres récents, par type d'utilisation : [Retouche photographique](#), [Filtres de déformations](#), [Filtres de dégradations](#), [Filtres de rendu et de textures](#), et enfin [Effets artistiques](#).

3.1. Retouche photographique

Trois filtres intéressants ont fait leur apparition dans le greffon G'MIC-Qt pour aider les photographes à retoucher leurs clichés numériques.

Tout d'abord, le filtre **Colors / Mixer [Generic]**, un filtre de mixage de canaux couleur qui offre la possibilité de choisir parmi pas moins de 16 espaces/représentations de couleurs différents pour le mixage ([CMY^w](#), [CMYK^w](#), [HCY](#), [HSI](#), [HSL^w](#), [HSV^w](#), [Jzazbz](#), [Lab^w](#), [Lch](#), [OKlab](#), [RGB^w](#), [RYB^w](#), [XYZ^w](#), [YCbCr^w](#), [YIQ^w](#) et [YUV^w](#)). Cela constitue une bonne alternative aux outils traditionnels de rehaussement de contrastes ou de couleurs, pour retoucher les photographies qui pourraient être un peu palôtes.



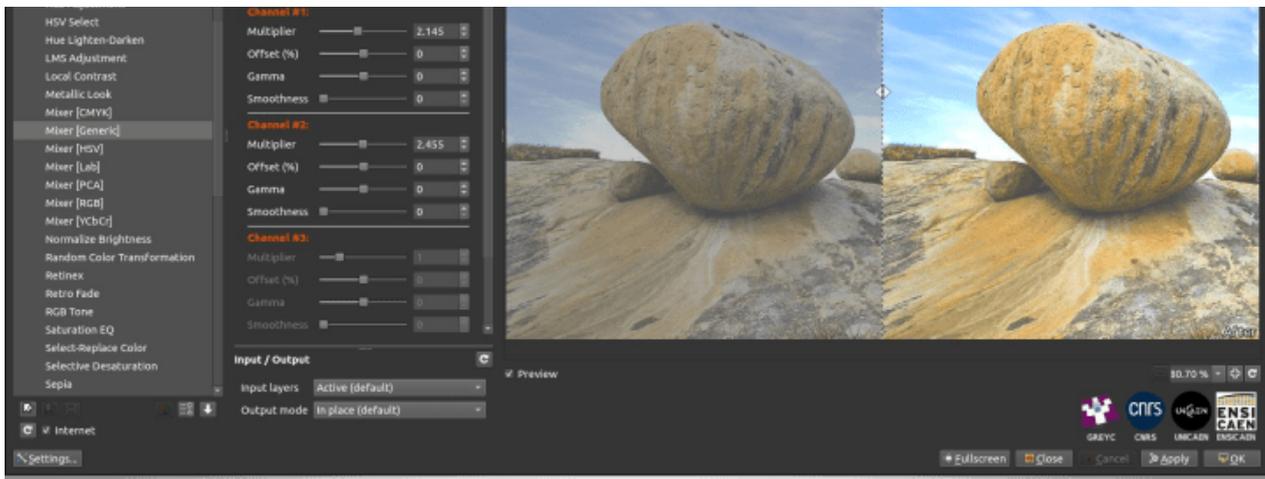


Fig.3.1.1. Le filtre **Colors / Mixer [Generic]** vient enrichir l'arsenal de filtres déjà disponibles pour la retouche des contrastes et des couleurs.

Parlons également du filtre **Details / Sharpen [Alpha]**, qui comme son nom l'indique, permet de rehausser les détails fins dans des photographies. Il est basé sur une technique originale de décomposition pyramidale de l'image relativement à un opérateur d'[Alpha blending](#)^W. Cette technique cherche en particulier à minimiser l'apparition de « halos » près des contours, qui est un artefact classique rencontré avec les filtres usuels de rehaussement de détails.

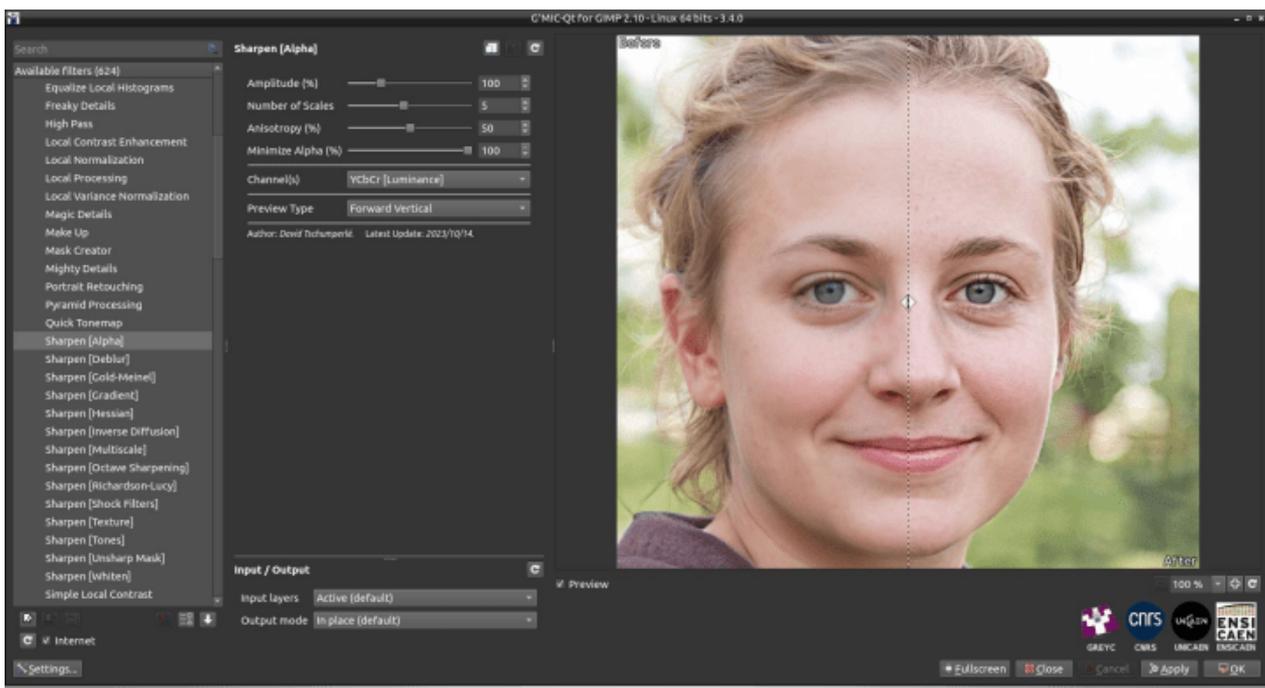


Fig.3.1.2. Le filtre **Details / Sharpen [Alpha]** permet de rehausser assez finement les détails dans les photographies, en minimisant l'apparition de « halos ».

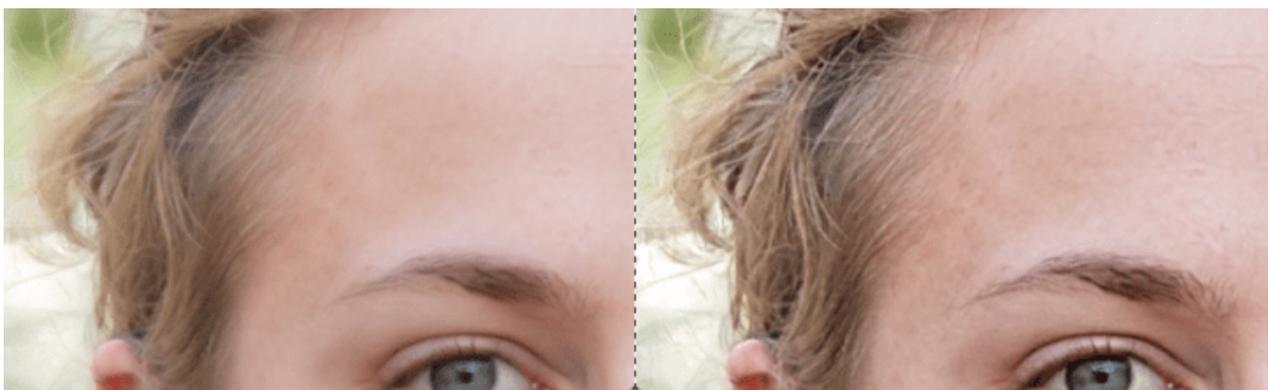




Fig.3.1.3. Détails « Avant/Après » illustrant l'intérêt du filtre **Details / Sharpen [Alpha]**.

Enfin, le filtre **Layers / Spatial Blend Multi-Layers** permet de fusionner plusieurs prises d'un même point de vue en une seule image, avec la possibilité de générer un gradient linéaire spatial entre ces différentes vues. On peut par exemple photographier une même scène à plusieurs moments de la journée (et de la nuit), et utiliser ce filtre de fusion spatiale pour créer une image telle que celle illustrée ci-dessous, en un ou deux clics de souris :



Fig.3.1.4. Le filtre **Layers / Spatial Blend Multi-Layers** permet de fusionner plusieurs photographies avec un gradient spatial linéaire, réglable par l'utilisateur.

(**Crédits** : les images utilisées dans la figure ci-dessus proviennent de la vidéo « [Stunning New York City skyline timelapse: Day to night](#) » de la chaîne Youtube « [Rumble Viral](#) »).

La vidéo suivante illustre le processus complet, en utilisant le greffon *G´MIC-Qt* sous *GIMP 2.10* :





3.2. Filtres de déformations

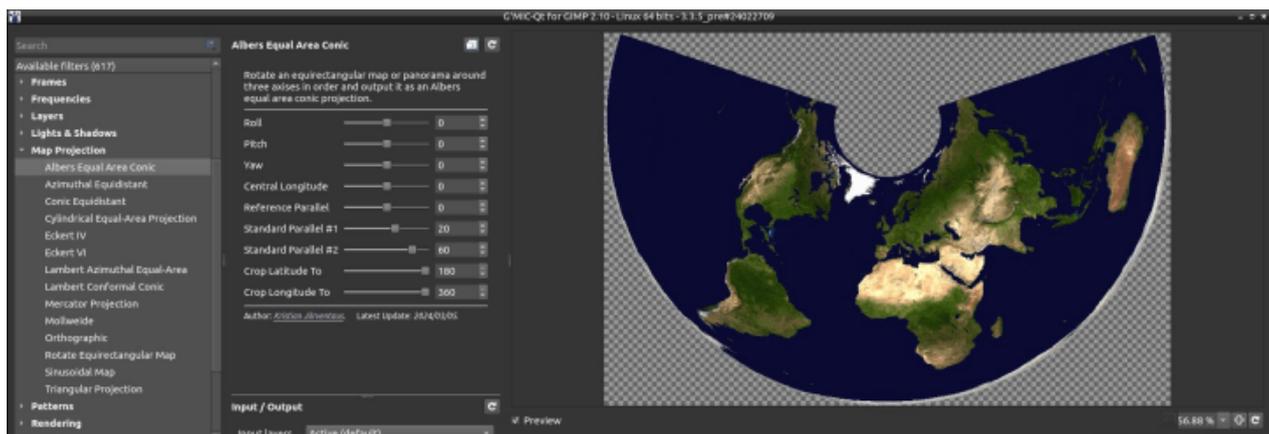
Passons maintenant à un ensemble de nouveaux effets disponibles pour déformer vos images un peu dans tous les sens.

Avec tout d'abord un premier filtre **Déformations / Distort [RBF]**, qui déforme une image à partir de points clés définis par l'utilisateur et d'une fonction d'interpolation de type *RBF* ([Fonction de base radiale](#)^W) dont la formule mathématique est spécifiable par l'utilisateur.



Fig.3.2.1. Le filtre **Déformations / Distort [RBF]** permet d'appliquer des déformations variées, basées sur des *RBFs*. Ici, en spécifiant la fonction de base radiale $\phi(r) = \log(0.1+r)$.

Mentionnons ensuite l'apparition d'une nouvelle catégorie **Map Projection**, qui contient 14 filtres différents dédiés à la transformation de cartes initialement sous la forme de [projection cylindrique équidistante](#)^W (cartes équirectangulaires), pour les convertir sous la forme d'autres types de projections. Cette série de filtre est une contribution de *Kristian Järventaus*, un membre du [Cartographers Guild forum](#), forum spécialisé dans la cartographie.



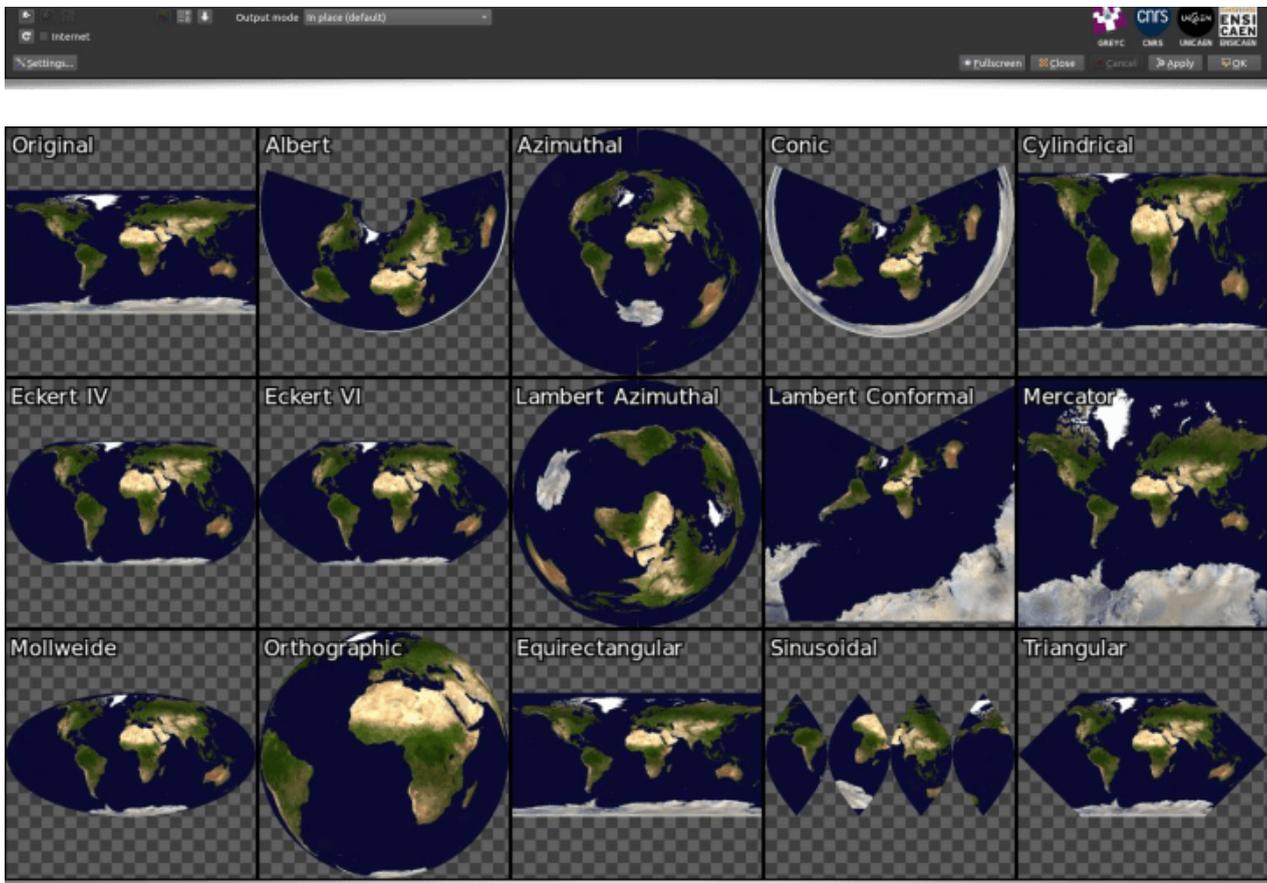
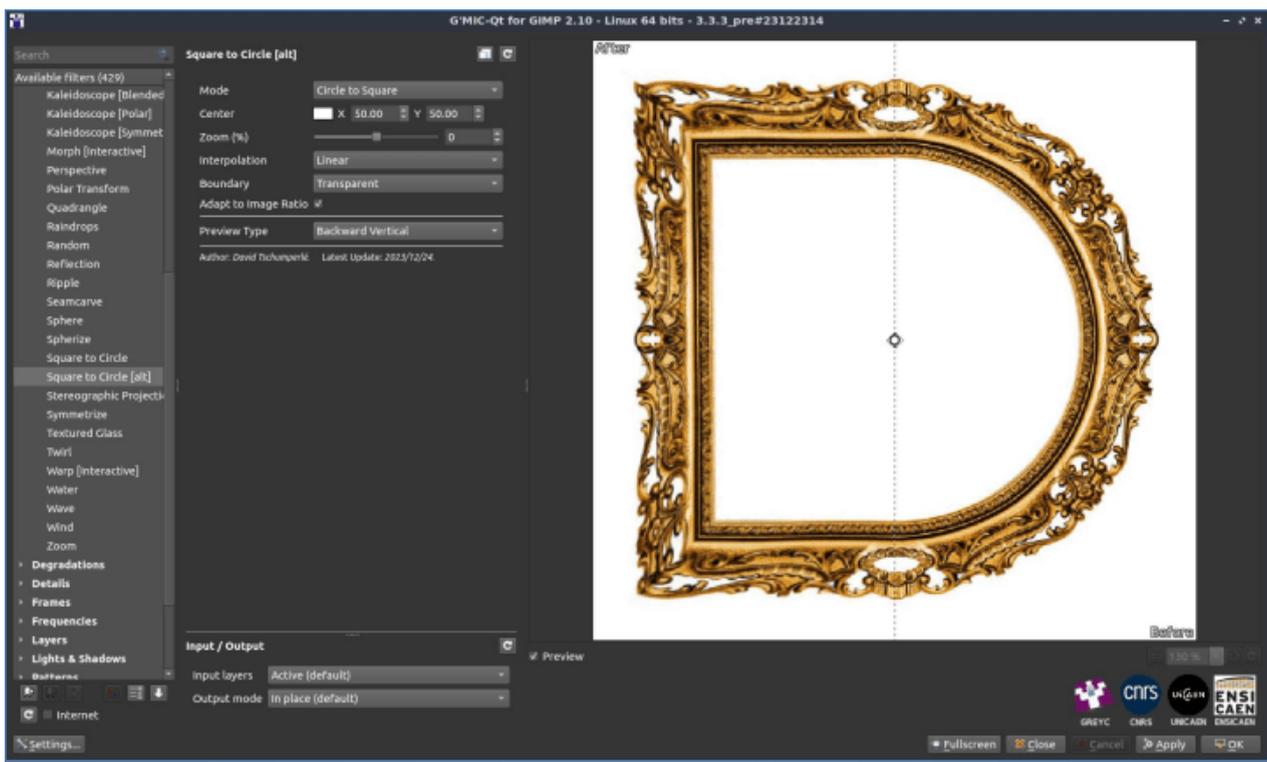


Fig.3.2.2. La nouvelle catégorie de filtres **Map Projection** propose plusieurs algorithmes de projection de cartes géographiques.

Notons également l'arrivée du filtre **Deformations / Square to Circle [alt]**, dont le but est de transformer tout rectangle (ou carré) centré dans une image, en une ellipse (ou un cercle), et vice-versa. Un filtre certes très spécialisé, avec évidemment peu d'applications évidentes au premier abord, mais le jour où on en a besoin, on est bien content de l'avoir sous la main ! Nous l'avons par exemple utilisé ci-dessous pour transformer un cadre de peinture initialement rond, en un cadre carré :



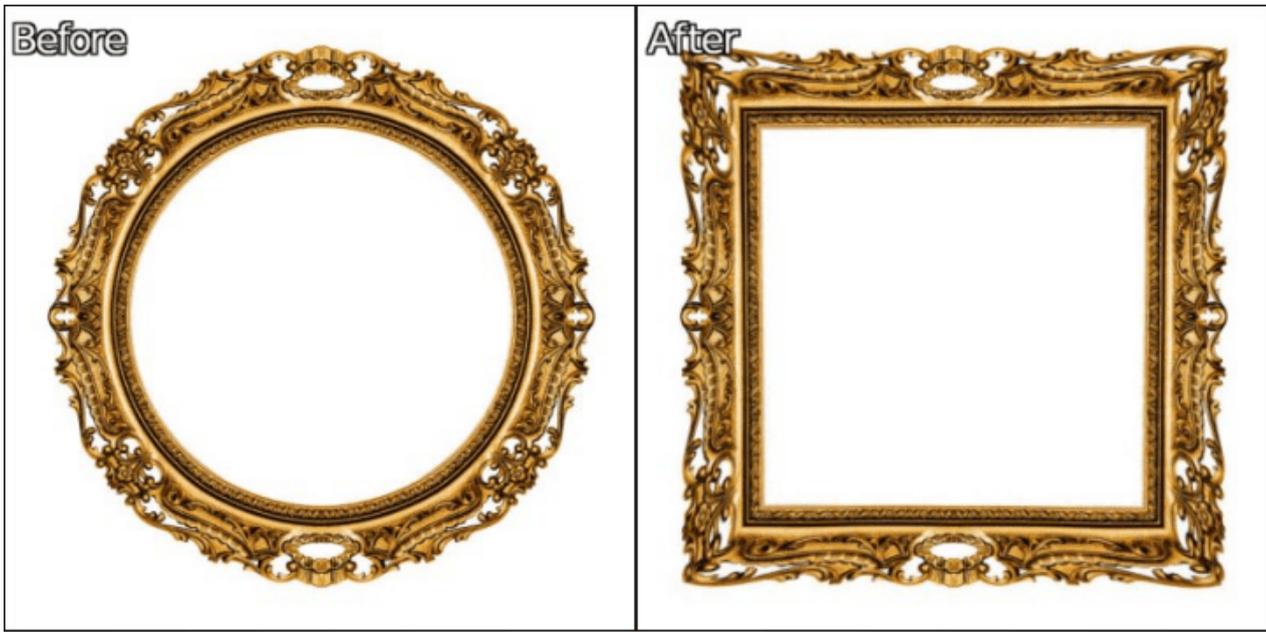


Fig.3.2.3. Le filtre **Deformations / Square to Circle [alt]** permet de convertir des objets carrés ou rectangulaires en objets ronds ou elliptiques, et inversement.

Enfin, terminons cette revue des nouveaux filtres de déformations d'images, avec le filtre **Deformations / Poincaré Disk**. Celui-ci permet d'une part de générer des disques de Poincaré^w, une famille de figures géométriques basées sur des géométries hyperboliques, comme illustrée sur la figure suivante :

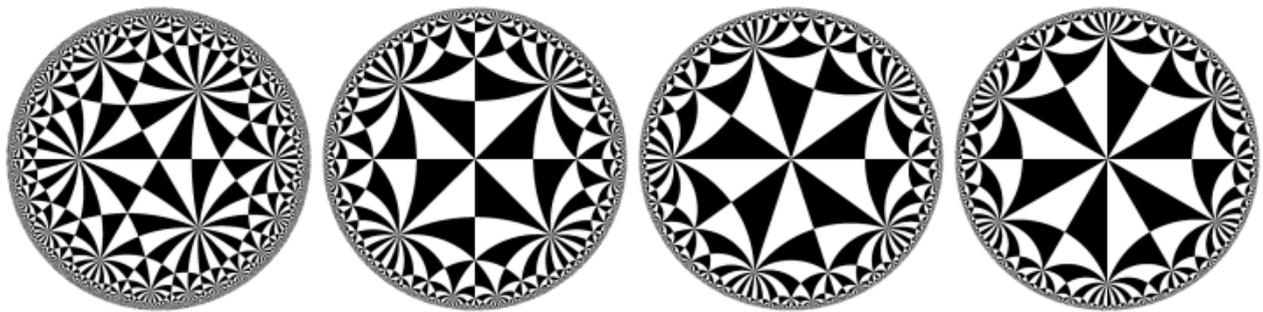


Fig.3.2.4. Quelques exemples de disques de Poincaré, générés par le filtre **Deformations / Poincaré Disk**.

Mais là où ça devient intéressant, c'est que ce filtre permet aussi de déformer des images en les projetant sur ces géométries si particulières :



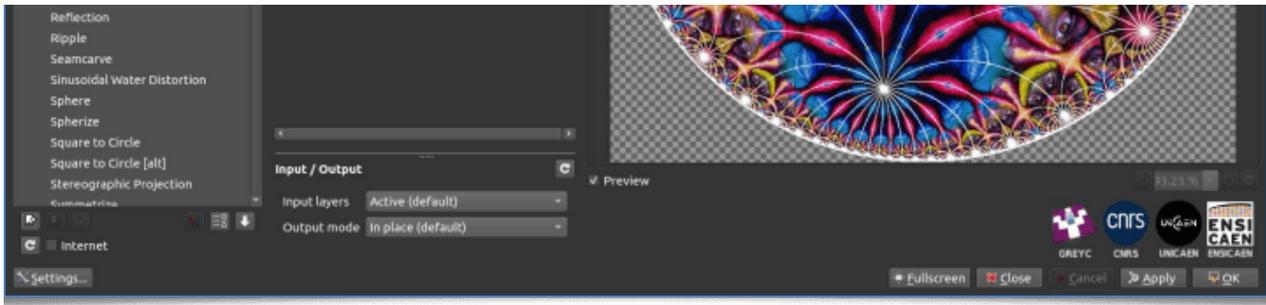
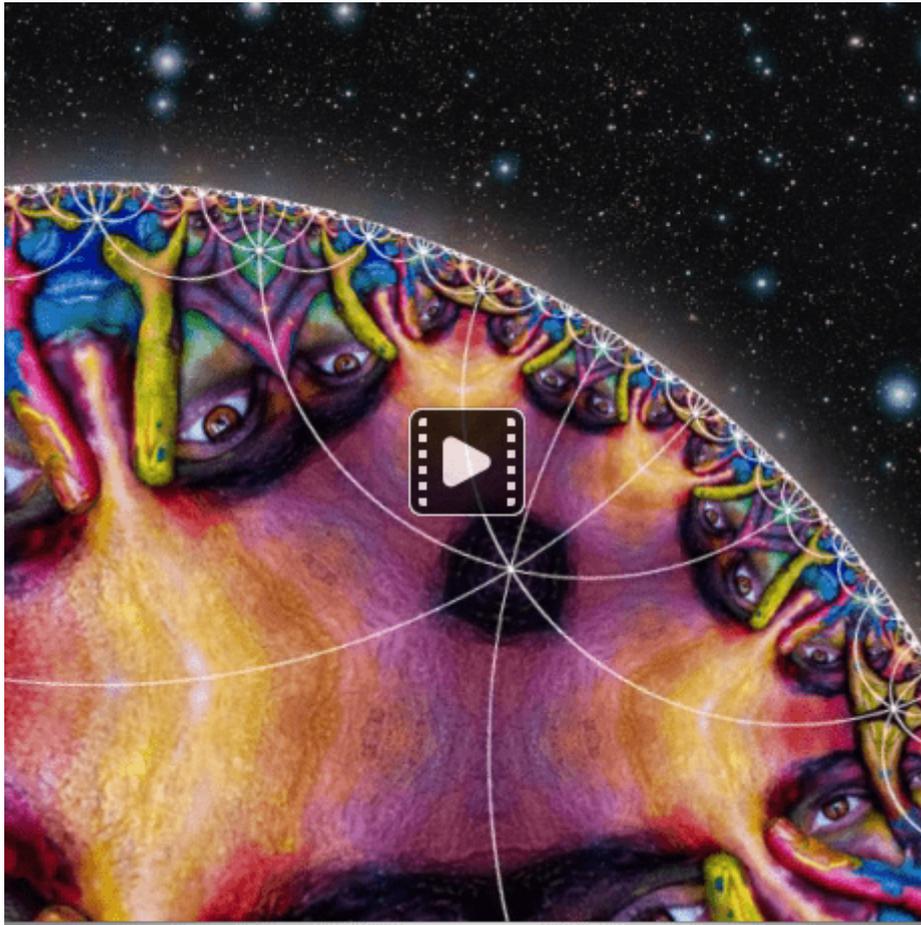


Fig.3.2.5. Projection d'une image sur un disque de Poincaré.

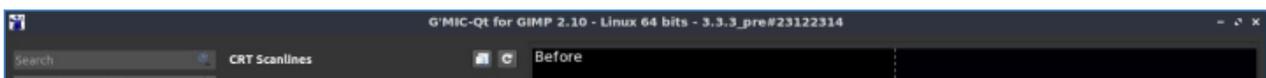
Par exemple, nous avons utilisé ce filtre (avec quelques modifications maison) pour générer la courte animation suivante, « *Survol de la planète Poincaré* », qui illustre les étonnantes propriétés fractales de ces bizarreries géométriques :



3.3. Filtres de dégradations

Il arrive que l'on cherche volontairement à *dégrader* des images, soit pour simuler une altération réelle (par exemple un flou de bougé, du bruit de capteurs...), soit dans la recherche d'un effet purement esthétique ([Glitch Art](#)^W). Dans ces buts, les nouveaux effets suivants ont été ajoutés à G'MIC :

- Les filtres **Rendering / CRT Scanlines** et **Degradations / CRT Phosphors** cherchent à imiter la restitution d'images sur des [écrans à tube cathodique \(CRT\)](#)^W, en simulant deux effets caractéristiques de ces types d'afficheurs, à savoir [l'effet Scanline](#)^W et l'affichage par [phosphores](#)^W. Ces deux filtres ont été réalisés en collaboration avec [Romain Hérauld](#), nouveau contributeur (qui a rejoint le GREYC récemment).



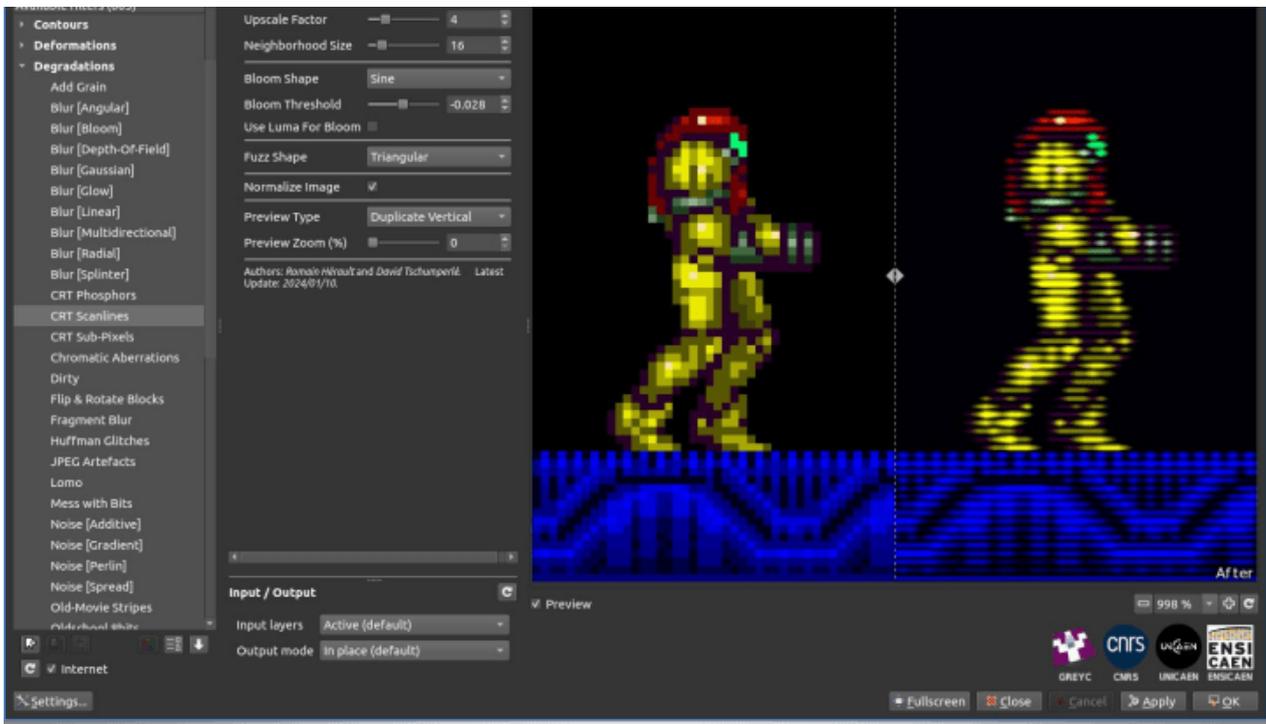


Fig.3.3.1. Le filtre **Rendering / CRT Scanlines** imite l'effet « Scanline » typique des afficheurs CRT.

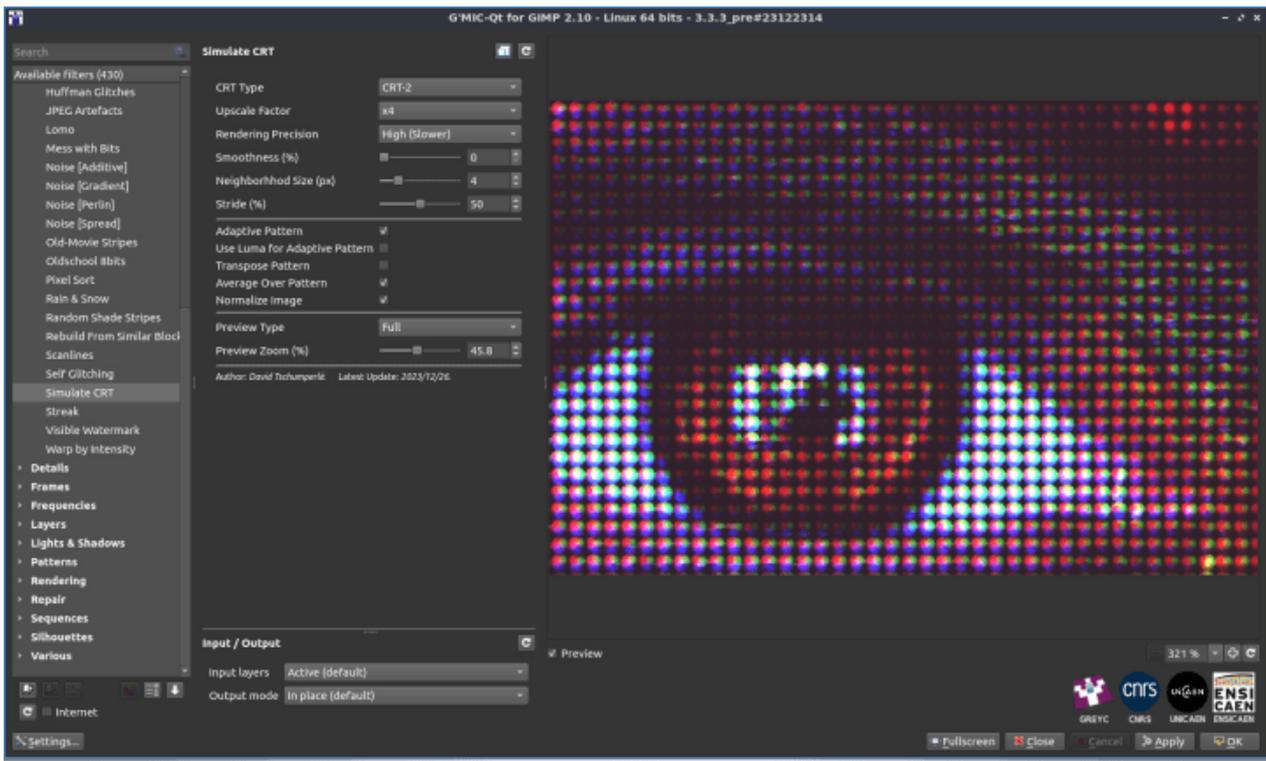


Fig.3.3.2. Le filtre **Degradations / CRT Phosphors** quant à lui simule la technique d'affichage par phosphore lumineux des afficheurs CRT.

- Le filtre **Degradations / Blur [Motion]** permet, lui, de synthétiser un flou de bougé, avec des trajectoires de type *splines* possiblement complexes, réglables par l'utilisateur directement en modifiant des points clés dans la fenêtre de prévisualisation de G'MIC-Qt, comme illustré sur la figure ci-dessous :



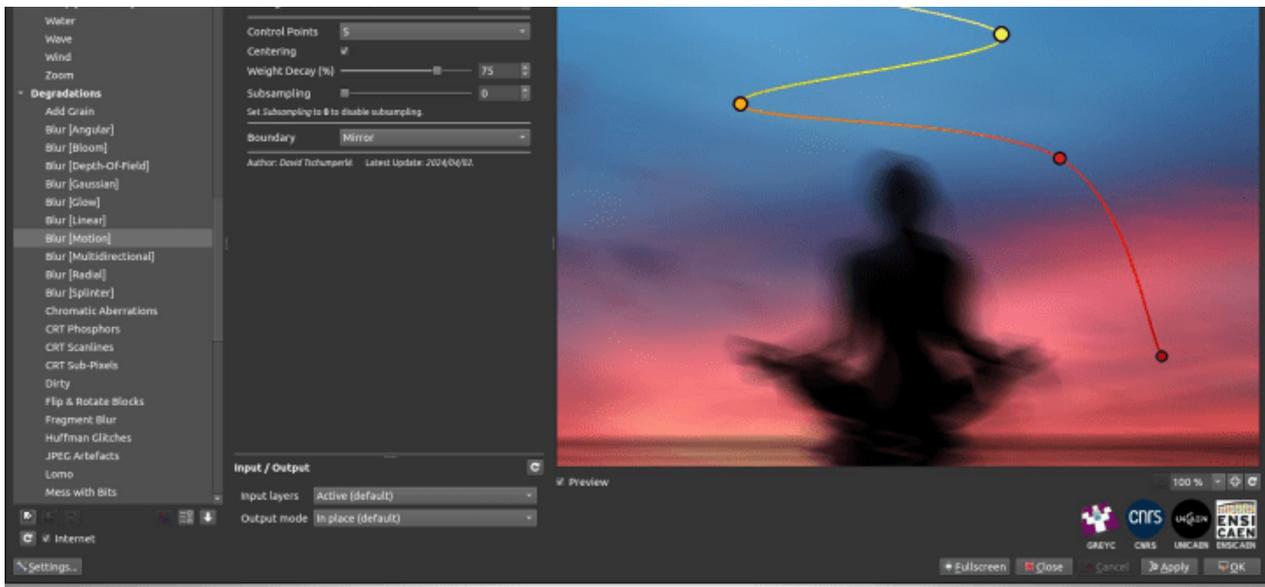


Fig.3.3.3. Le filtre **Degradations / Blur [Motion]** simule un flou de bougé.

- Le filtre **Degradations / Sloppy Mess** est quant à lui plutôt destiné aux amateurs de **Glitch Art**. C'est l'un des premiers filtres d'un nouveau contributeur, [Prawnsushi](#), qui s'est récemment intéressé au langage *G'MIC* pour la création de filtres. C'est un effet un peu long à calculer, mais qui possède de nombreux paramètres et une grande palette de rendus différents.

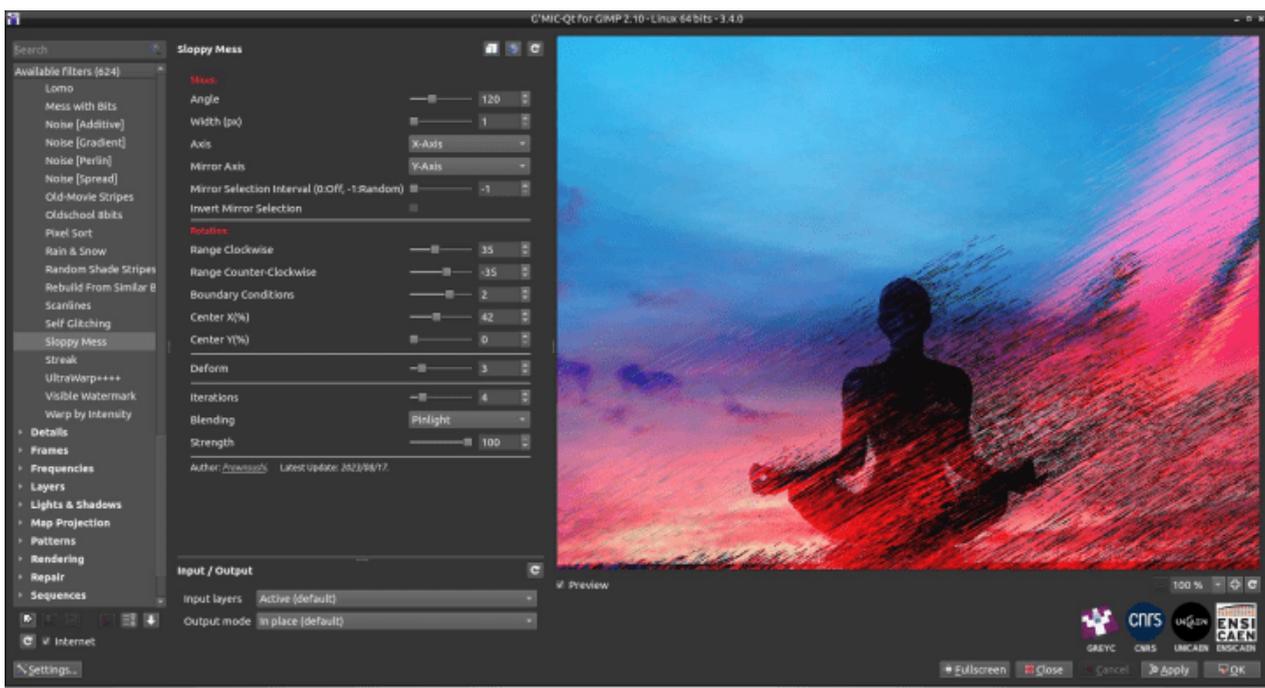


Fig.3.3.4. Le filtre **Degradations / Sloppy Mess** crée volontairement des artefacts numériques artistiques sur vos images.

3.4. Filtres de rendu et de textures

Les filtres de rendu ont cette particularité de ne pas nécessiter d'images d'entrée pour fonctionner, puisqu'ils créent de nouvelles images (ou de nouveaux calques) en partant de zéro, par **génération procédurale**^W.

- Le filtre **Rendering / Underwoods**, un deuxième filtre réalisé par [Prawnsushi](#), n'est rien de moins qu'un **générateur de sous-bois** ! On pourra nous dire que ça ne sert pas forcé-

ment à grand-chose, et évidemment ce n'est pas un filtre qui va être indispensable aux traiteurs d'images, photographes ou illustrateurs pour leur travail de tous les jours.

Mais en quelque sorte, ce filtre illustre pourtant toute la philosophie du projet *G'MIC* : Produire un logiciel qui favorise la **créativité algorithmique**, facilite l'implémentation de **toutes sortes de filtres** (utiles ou moins utiles), et permet le **libre partage** avec des utilisateurs. En pratique, ce filtre a le mérite d'exister, chacun peut l'essayer, et il n'utilise que peu de ressources mémoire (quelques dizaines d'octets, grâce à la compacité du langage *G'MIC* par construction). Et en plus, les résultats sont plutôt cool ! Et soyez assurés qu'il servira inmanquablement, un jour ou l'autre, à un artiste voulant synthétiser une image de sous-bois en deux clics de souris !

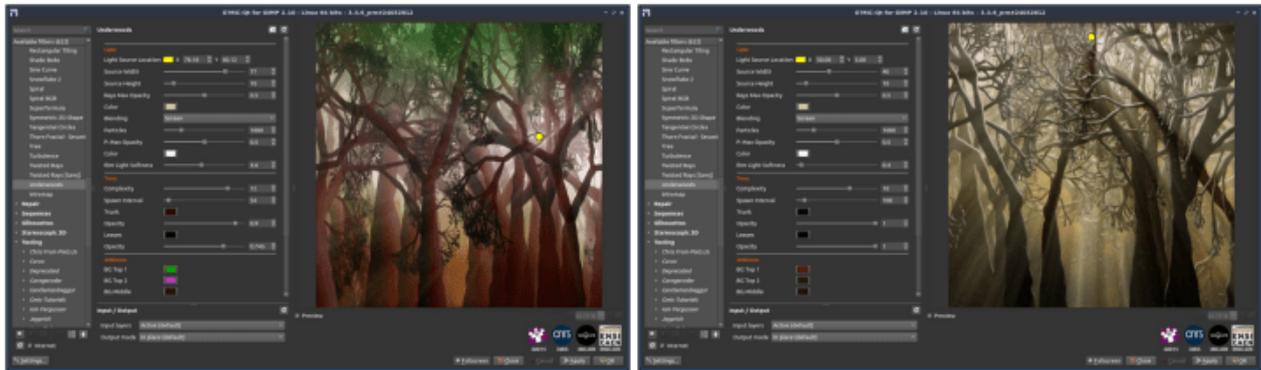


Fig.3.4.2. Exemples de rendus d'images de sous-bois par le filtre **Rendering / Underwoods**.

- Le filtre **Patterns / Reaction-Diffusion** synthétise des textures du type de celles qu'on rencontre dans la modélisation des systèmes de [Réaction-Diffusion^W](#).



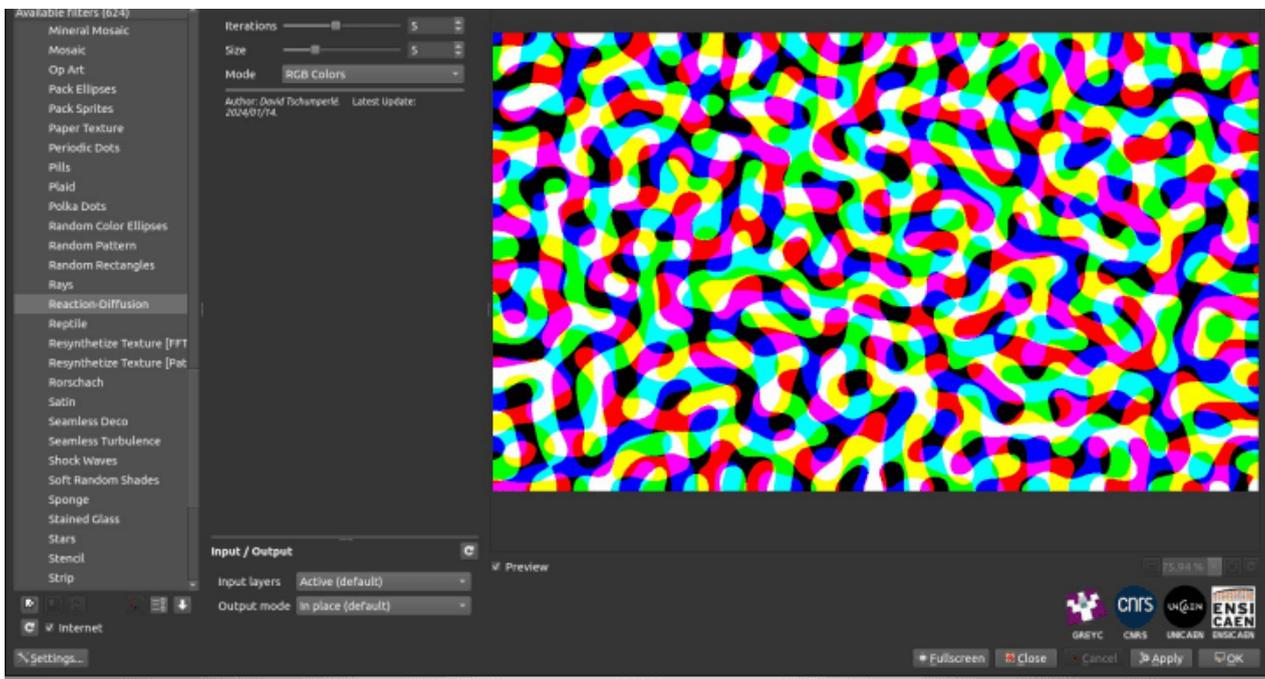


Fig.3.4.3. Le filtre **Patterns / Reaction-Diffusion** dans le greffon G'MIC-Qt.

Là encore, l'intérêt de ce type de filtres peut paraître assez limité. Pourtant c'est l'exemple même de texture qui peut servir de base à la création d'œuvres d'art génératif. Prenez une texture créée par ce filtre, appliquez-lui quelques-uns des autres effets disponibles dans G'MIC (au hasard, le filtre **Deformations / Drop Water**), et vous voilà prêt à produire des animations amusantes, telles que celle ci-dessous :



- Le filtre **Rendering / Spline Spirograph** s'inspire du jeu du [Spirographe](#) pour le processus de génération de courbes paramétriques, créant des effets de texture et de couleurs intrigants. La présence du bouton **Paramètres aléatoires** dans le greffon G'MIC-Qt est ici plus que bienvenue, pour obtenir rapidement un panorama varié des résultats possibles de ce filtre !

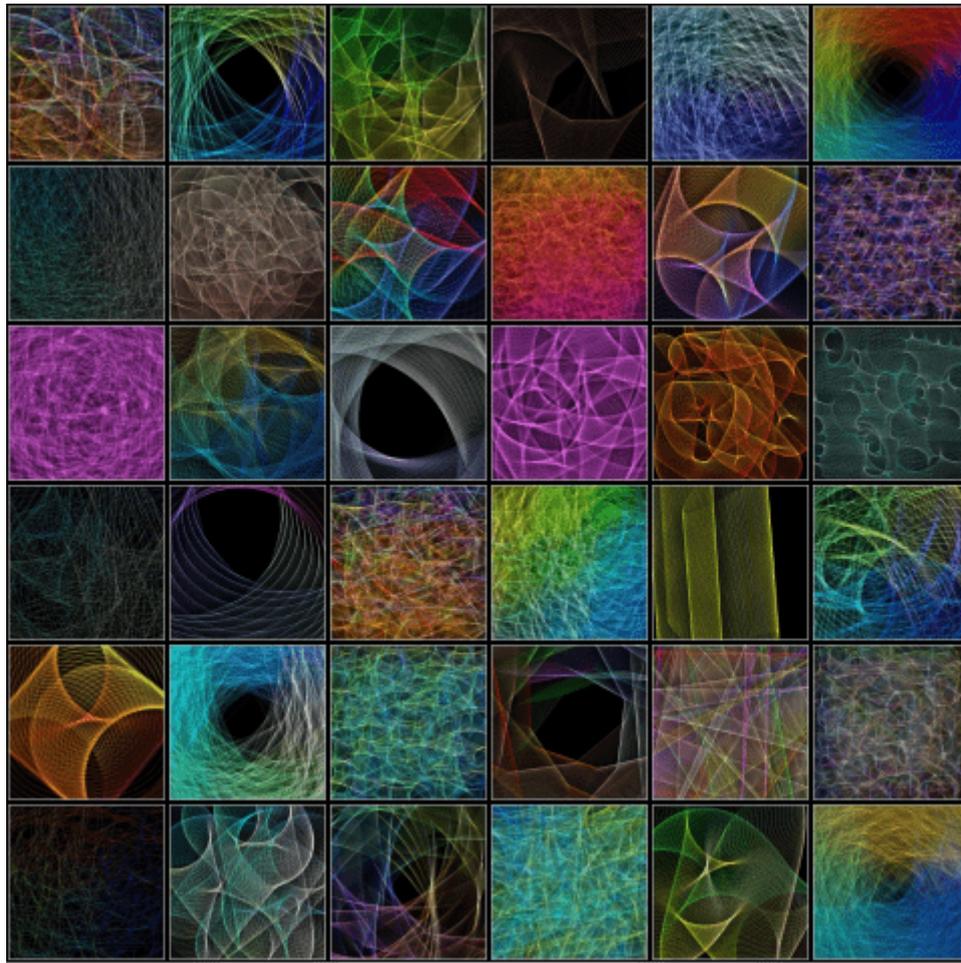
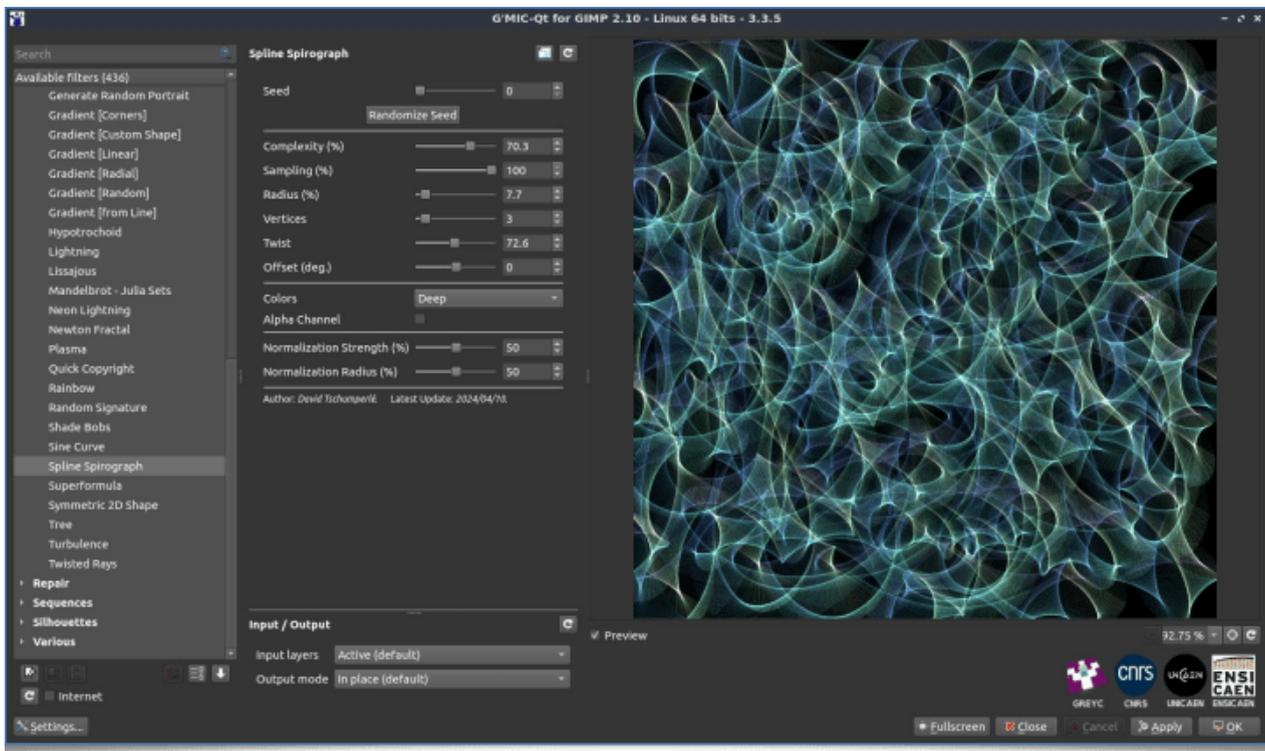
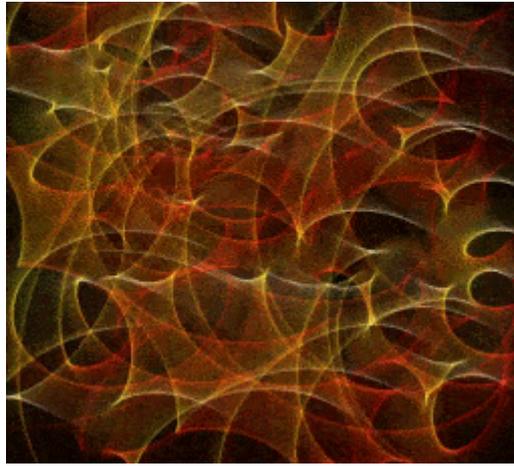


Fig.3.4.1. Le filtre **Rendering / Spline Spirograph** et quelques exemples de rendus possibles.

Ce filtre propose également un mode de sortie animé, produisant plusieurs calques consécutifs qui représentent différentes *frames* d'une animation, ce qui permet de synthétiser de courtes vidéos comme celle-ci :





- Le filtre **Rendering / ABN Filigrees** sait lui aussi tracer des courbes paramétriques intéressantes, en s'inspirant cette fois des filigranes que l'on trouve sur les certificats d'actions ou de papier-monnaie. Ce filtre rend hommage à l'*American Bank Note Company*, un graveur de papier financier ayant été actif à Wall Street du milieu du XIX^e siècle jusqu'à il y a une soixantaine d'années.

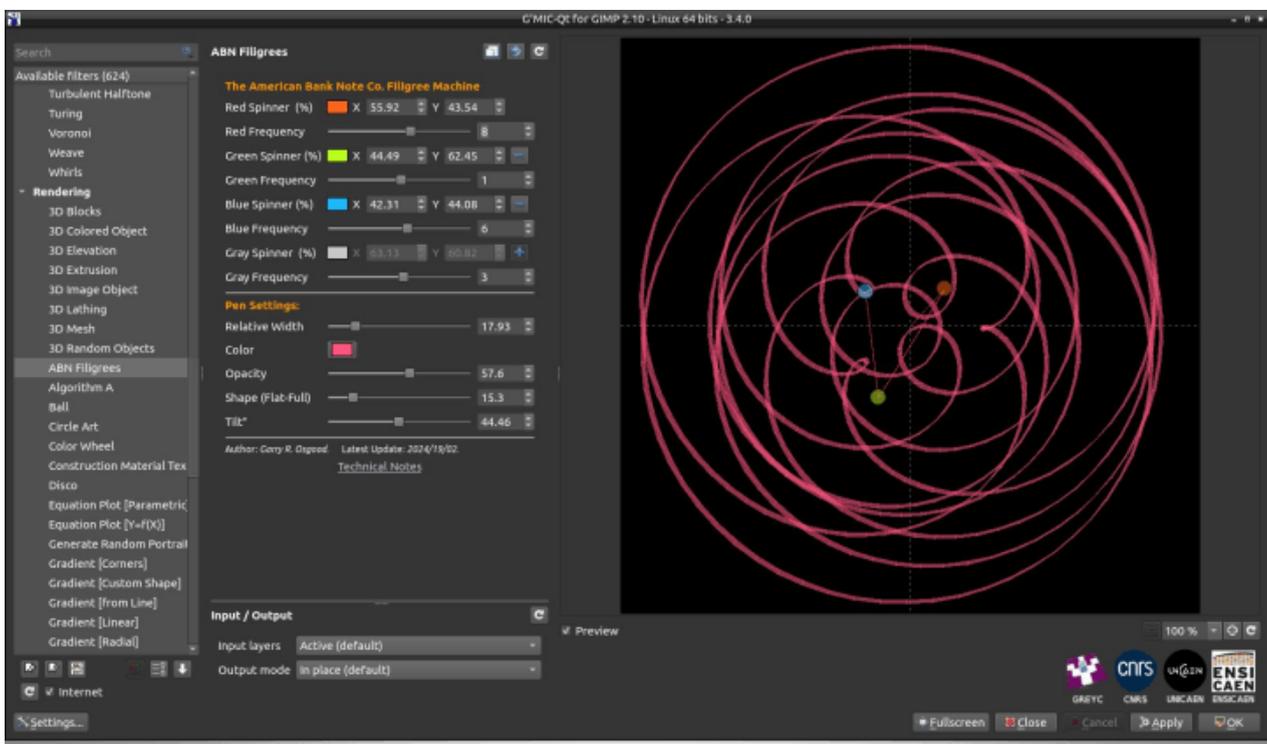


Fig.3.4.4. Le filtre **Rendering / ABN Filigrees** en action.

Ce filtre a été réalisé par [Garry Osgood](#), contributeur de longue date et rédacteur [des pages de tutoriels](#) du projet. Garry a détaillé tous les aspects techniques de ce filtre-là en particulier, [sur cette page](#).

- Le filtre **Rendering / Random Signature** est lui aussi un générateur de courbes amusant : il se propose de dessiner des signatures aléatoires, comme l'illustre la vidéo ci-dessous :





Fig.3.4.4. Le filtre **Rendering / Random Signature** permet de créer des signatures aléatoires.

Là encore, un filtre dont l'utilité ne peut pas faire débat : « C'est complètement inutile, donc rigoureusement indispensable ! »^W.

- Et pour finir cette section, citons le filtre **Rendering / Twisted Rays** qui génère, comme son nom l'indique, un effet de rayons torsadés. Là encore, l'utilisation de ce filtre restera probablement assez confidentielle. Mais qui sait si dans les milliers d'utilisateurs de *G'MIC*, il n'y a pas justement quelqu'un qui cherchait à réaliser cet effet en particulier ? Et bien maintenant, il peut ! 😊

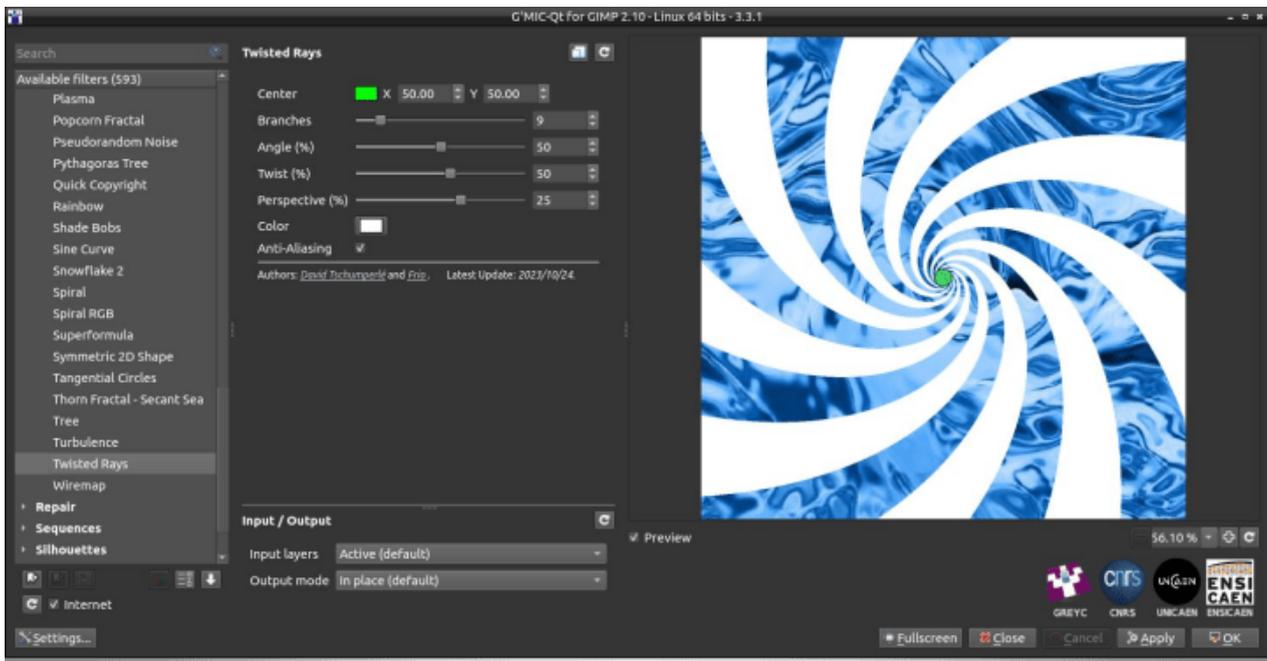
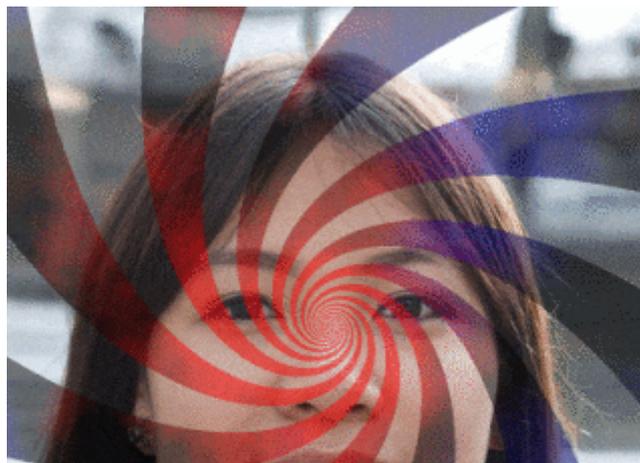


Fig.3.4.5. Le filtre **Rendering / Twisted Rays** crée un tourbillon de rayons sur vos images.

Ce filtre peut par exemple être utilisé pour créer ce genre d'effets animés psychédéliques :





3.5. Effets artistiques

Et pour finir avec cette énumération des nouveaux filtres de *G'MIC*, voici quelques filtres en vrac, que l'on va regrouper sous la dénomination « *Artistique* » (c'est généralement le terme employé dans les logiciels de traitement d'images, pour qualifier des filtres qu'on ne sait pas trop catégoriser...).

- Le filtre **Artistic / Stringify** décompose une image d'entrée en *blobs* de couleurs quantifiés, et relie les points (sous-échantillonnés) localisés sur les contours de ces *blobs* par des segments de couleur. Là encore, le rendu peut faire penser aux courbes que l'on obtient avec le jeu du Spirographe.

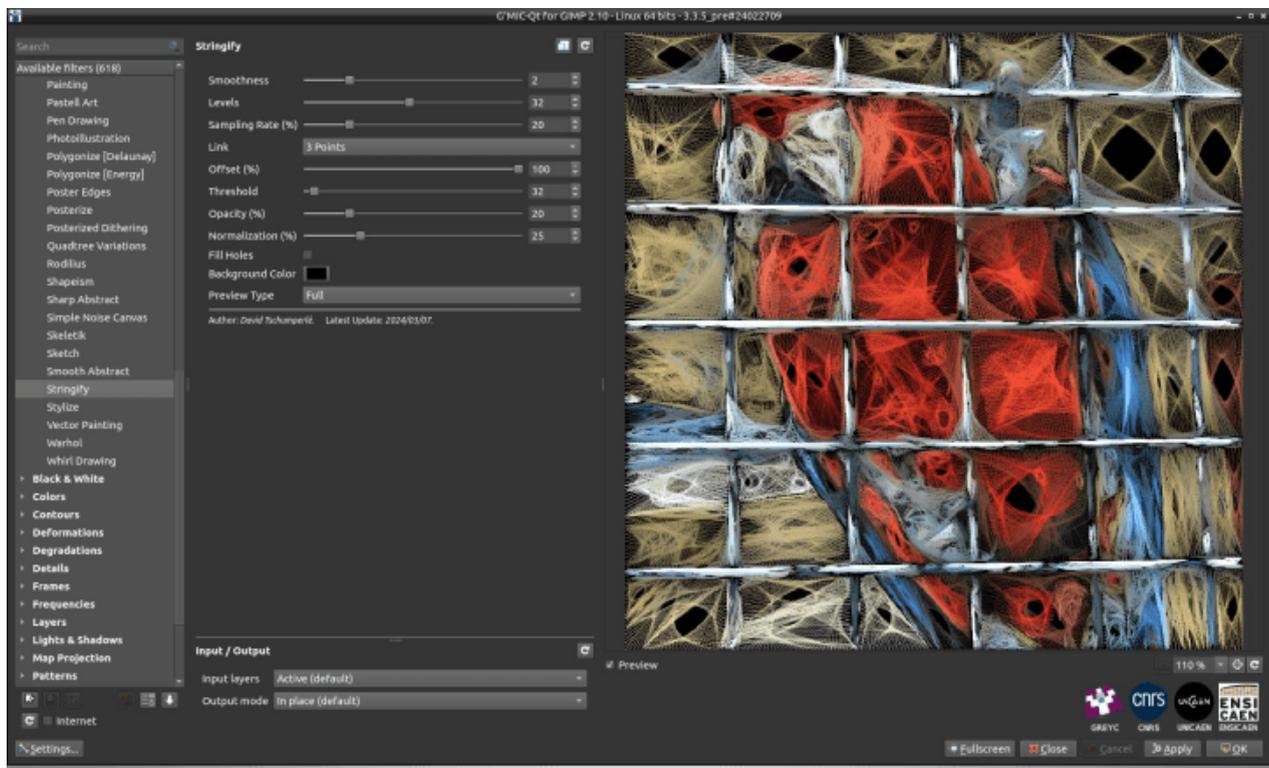
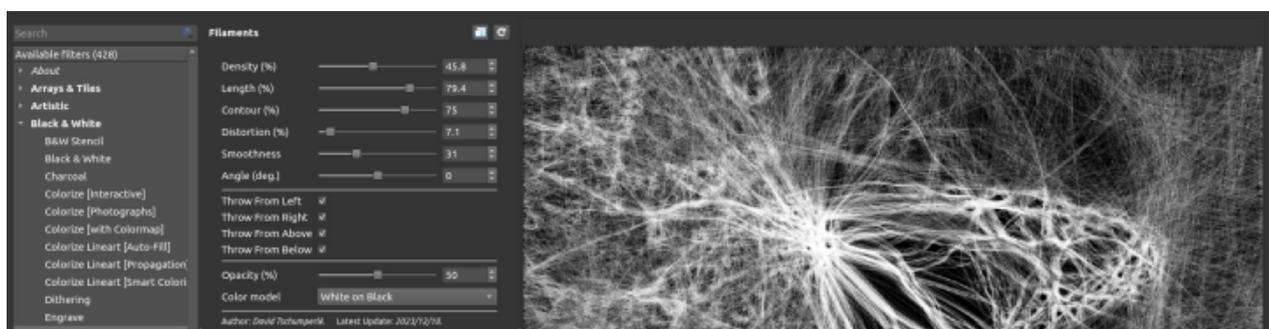


Fig.3.5.1. Le filtre **Artistic / Stringify** crée des abstractions d'images à partir de segments de couleur.

- Le filtre **Black & White / Filaments** est lui, basé sur le lancer de milliers de particules à partir d'un ou plusieurs bords de l'image, avec des trajectoires qui sont déformées par la géométrie des contours de l'image. Le tracé de ces milliers de trajectoires avec des couleurs semi-transparentes permet de produire ce genre d'images :



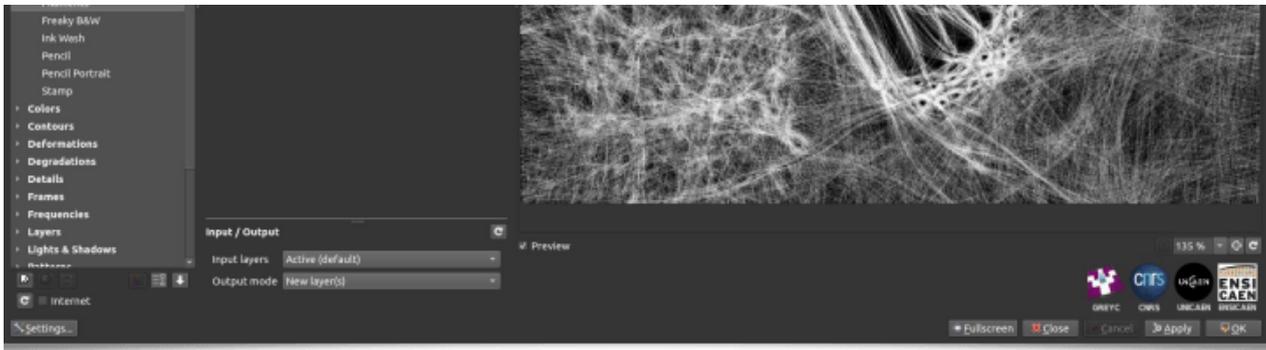


Fig.3.5.2. Le filtre **Black & White / Filaments** transforme vos images en ensembles de filaments qui se déforment.

- Le filtre **Arrays & Tiles / Loose Photos** simule un effet de lancer de photos en vrac sur une table, de telle manière que le contenu des photos recrée une image globale, spécifiée comme entrée du filtre. Beaucoup de paramètres sont réglables (densité, taille et ratio des photographies, paramètres d'ombrage, etc.), afin de laisser à l'utilisateur une grande liberté sur le choix du rendu final.

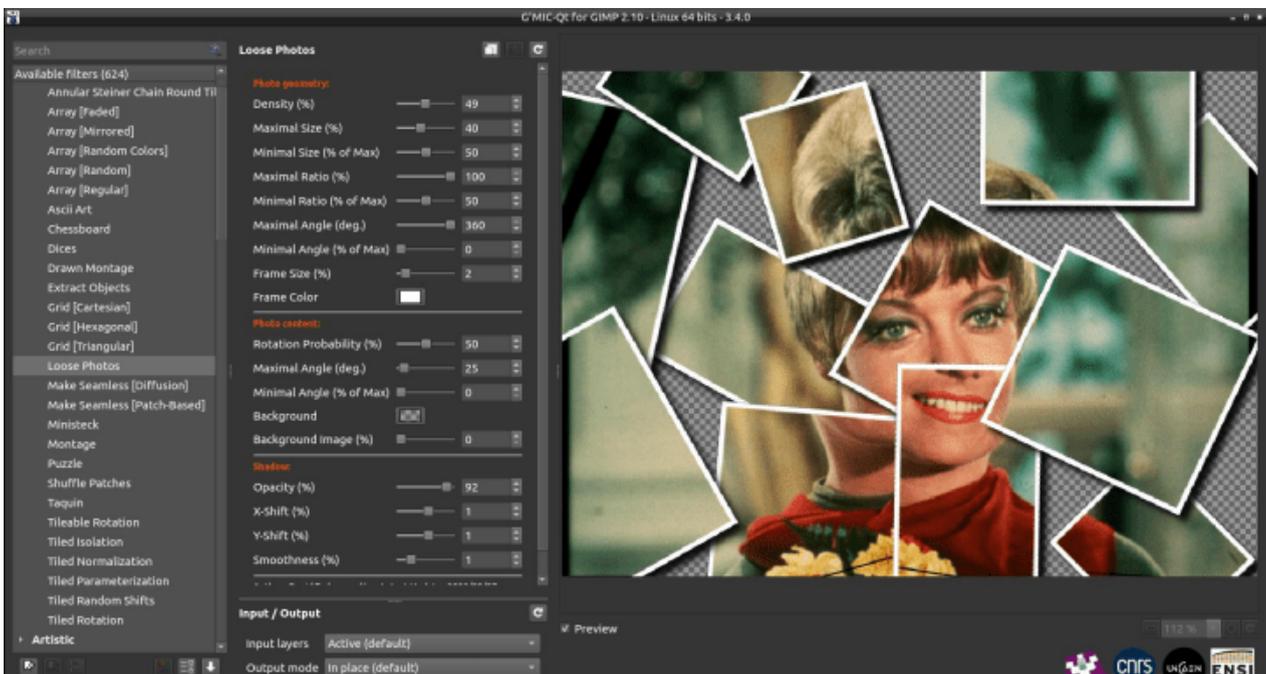




Fig.3.5.3. Exemple de rendu par le filtre **Arrays & Tiles / Loose Photos**.

- Le filtre **Rendering / Quick Copyright**, qui existait déjà dans les versions précédentes, a été ré-implémenté à partir de zéro. Il s'agit d'un filtre facilitant l'insertion d'un texte de *copyright* (ou d'une signature texte) sur une image numérique. Il possède maintenant de nombreux paramètres permettant de régler finement la position du texte, sa taille, sa fonte, etc. Combiné avec les possibilités d'automatisation du langage G'MIC, c'est un filtre avantageux pour insérer facilement du texte, de manière normalisée, sur des milliers de photographies.

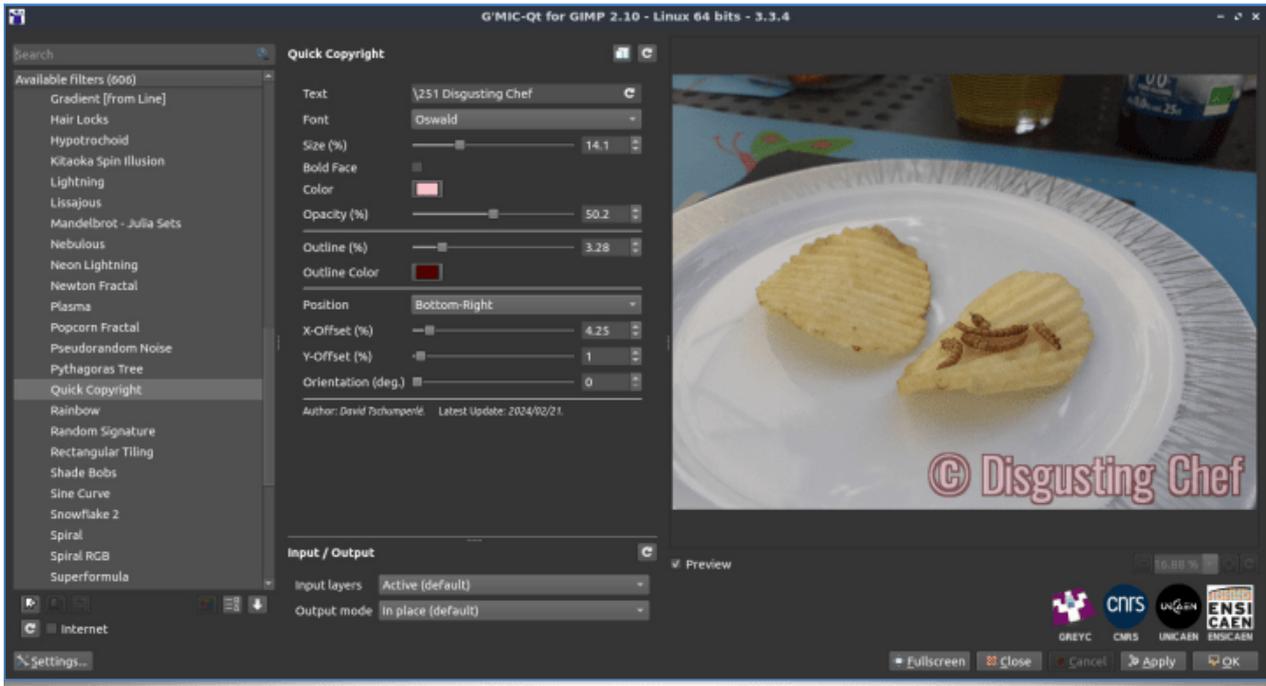
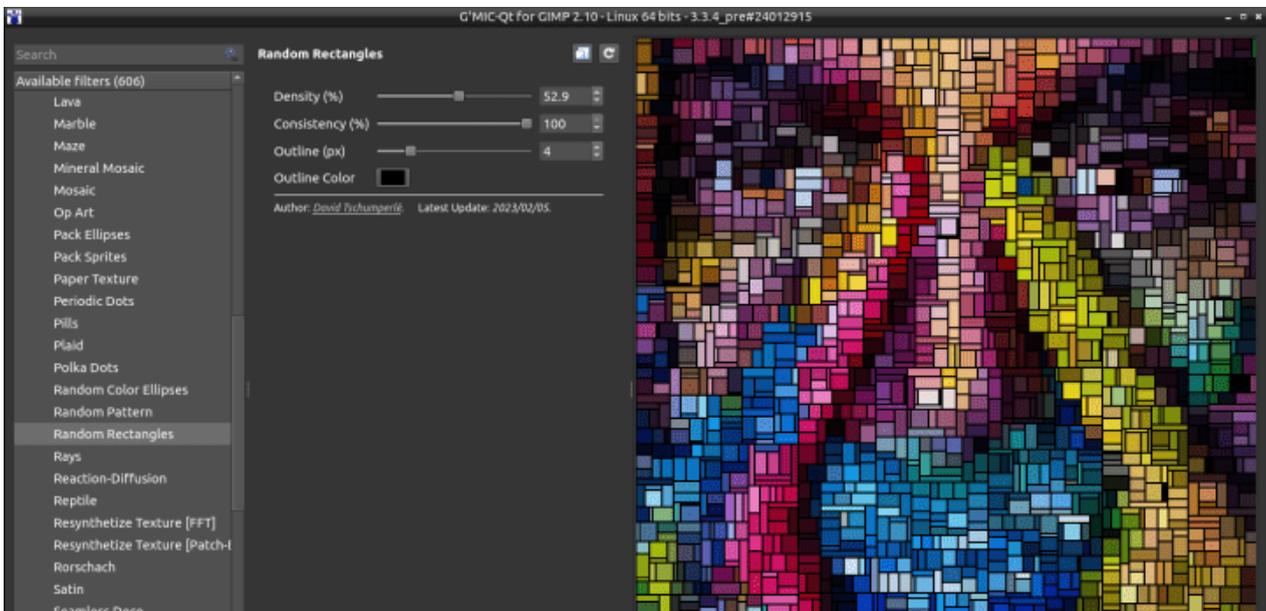


Fig.3.5.4. Le filtre **Rendering / Quick Copyright** en action pour l'insertion de texte de *copyright* sur une image.

- Et enfin, le filtre **Patterns / Random Rectangles** transforme une image en une partition aléatoire de rectangles colorés, pour donner un effet d'abstraction d'image, s'inspirant vaguement du style de certains tableaux de [Piet Mondrian](#)^W.



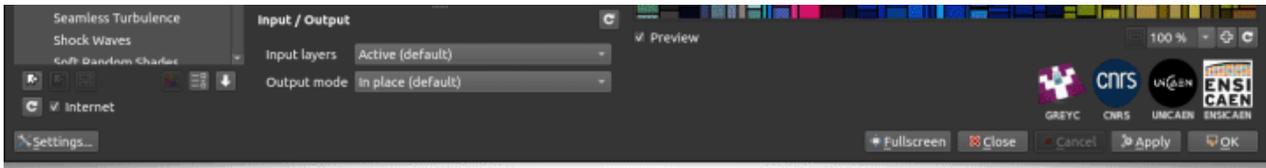
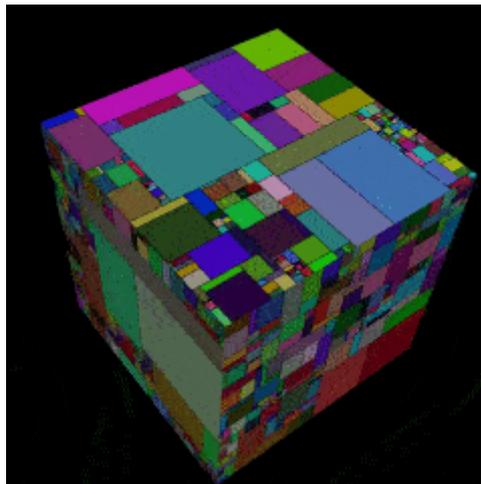


Fig.3.5.5. Le filtre **Patterns / Random Rectangles** crée une partition colorée de rectangles aléatoirement placés sur l'image.

Là encore, on peut imaginer utiliser ce filtre pour générer des textures qui pourront par exemple servir à décorer des objets, comme cela a été fait dans l'animation ci-dessous (en partant d'une image de bruit pur) :



4. Un logiciel aux utilisations variées

Comme nous avons essayé de vous le montrer, G'MIC possède de multiples facettes, avec une vaste palette d'applications. La communauté des utilisateurs ne se limite d'ailleurs pas aux artistes numériques, mais comporte également des chercheurs, des programmeurs, des algorithmiciens... Nous montrons ci-dessous quelques autres utilisations variées du logiciel.

- **Débruitage d'images du JWST :**

En octobre 2023, nous avons appris par l'intermédiaire de [M.J. McCaughrean](#), chercheur senior à l'[ESA \(European Space Agency\)](#) ^W (parti à la retraite en 2024), que G'MIC était utilisé par certaines personnes au sein de l'ESA pour traiter des images provenant du [James Webb Space Telescope](#) ^W (JWST), en particulier pour l'atténuation du bruit fréquentiel apparaissant sur certaines images acquises par le télescope (via le filtre **Repair / Banding Denoise** notamment).

G'MIC a par exemple été utilisé (entre autres logiciels) pour la réalisation de l'image de couverture du magazine [Nature, vol. 622, issue 7981](#) du 5 octobre 2023, comme confirmé par [M.J. McCaughrean](#), et indiqué d'ailleurs dans [la publication associée](#), dont il est le co-auteur.



Article

Outflows from the youngest stars are mostly molecular

<https://doi.org/10.1038/s41586-023-06551-1>

Received: 3 June 2023

Accepted: 17 August 2023

T. P. Ray^{1,2*}, M. J. McCaughrean², A. Caratti o Garatti¹, P. J. Kavanagh^{1,3}, K. Justtanon⁴, E. F. van Dishoeck⁵, M. Reitsma^{2,7}, H. Beuther⁶, L. Francia², C. Gieser⁸, P. Klaassen¹⁰, G. Perotti⁸, L. Tychoniec⁹, M. van Gelder¹, L. Colina², Th. R. Greve¹⁰, M. Güdel^{11,12}, Th. Henning⁸, P. O. Lagage¹³, G. Östlin¹⁴, B. Vandenbussche¹⁵, C. Waelkens¹⁶ & G. Wright¹⁷

reverted to row-by-row correction only. We reprocessed these new calibrated slope files through Image2Pipeline and Image3Pipeline to reproduce the mosaics, this time with significantly less banding. Additional band denoising was done on the final images using G'MIC-Qt in GIMP before combining the images to make a colour composite.

The second issue encountered was slight subpixel offsets in the relative astrometry between the filter mosaics. To account for these, we

Extended Data Fig. 2 shows a plot of measured tangential velocity versus distance from the source for comparison with Fig. 5. We note here the apparent slowdown in motion to the northwest which may be related to its environment. The presence of an east-west density asymmetry, with denser gas to the west, is supported by mm-band observations. In particular a high-density filament is seen to the west, that is, redshifted side of HH 211-mm, in H¹³CO⁺ which is also observed in NH₃

Fig.4.1. Utilisation de G'MIC dans le cadre du débruitage d'images de la proto-étoile Herbig-Haro 211, acquises par le JWST.

Pour nous les développeurs, ça a vraiment été une bonne surprise d'apprendre ça, et de nous rendre compte que G'MIC était utilisé dans le domaine de l'astrophysique.

- **Code créatif :**

G'MIC se révèle être un allié précieux lorsque l'on cherche à faire du [code créatif^w](#), de par sa capacité à générer/manipuler facilement des images, et de par sa bibliothèque standard d'opérateurs fournie. Durant cette année écoulée, nous nous sommes amusés à explorer ses possibilités pour la création algorithmique d'images et d'animations, dont voici un petit florilège :

Commençons tout d'abord par cette petite animation de flocons de neige tourbillonnants ([code source](#), 30 lignes) :



Fig.4.2. Animation de flocons de neige, générée par un script G'MIC.

Puis enchaînons avec cette variante amusante du jeu de [Pierre-Feuille-Ciseaux^w](#), où l'on fait d'abord jouer consécutivement chaque pixel d'une image (dont les valeurs initialement aléatoires représentent soit 0 : une pierre, 1 : une feuille, ou 2 : des ciseaux) avec ses 8 voisins, en gardant à chaque fois l'élément qui a été gagnant le plus souvent. Dans un deuxième temps, on stylise la séquence de ces différentes images de labels, là encore, avec le filtre **Drop Water**, pour aboutir à l'animation suivante, que l'on pourrait intituler « La soupe de l'enfer » ([code source](#), 30 lignes),





Fig.4.3. Animation du jeu de Pierre-Feuille-Ciseaux, où tous les pixels d'une image jouent simultanément.

Et enfin, (re-)plongeons dans le monde merveilleux des [fractales de Mandelbrot^w](#), tout d'abord avec ce rendu de l'ensemble de Mandelbrot utilisant la technique du [Orbit Trap^w](#) permettant de faire un rendu en y plaquant une image couleur :



Fig.4.4. Rendu de l'ensemble de Mandelbrot par la méthode du Orbit Trap.

Puis, avec un autre type de rendu, connu sous le nom de [Buddhabrot^w](#), mais ici en considérant des séries complexes de type $z_{n+1} = z_n^p + c$, et en faisant varier linéairement l'exposant réel p entre 0 et 6 (plutôt qu'en gardant $p=2$ comme pour l'ensemble de Mandelbrot classique), pour générer chaque image de l'animation ci-dessous (code source et détails [sur cette page](#)) :



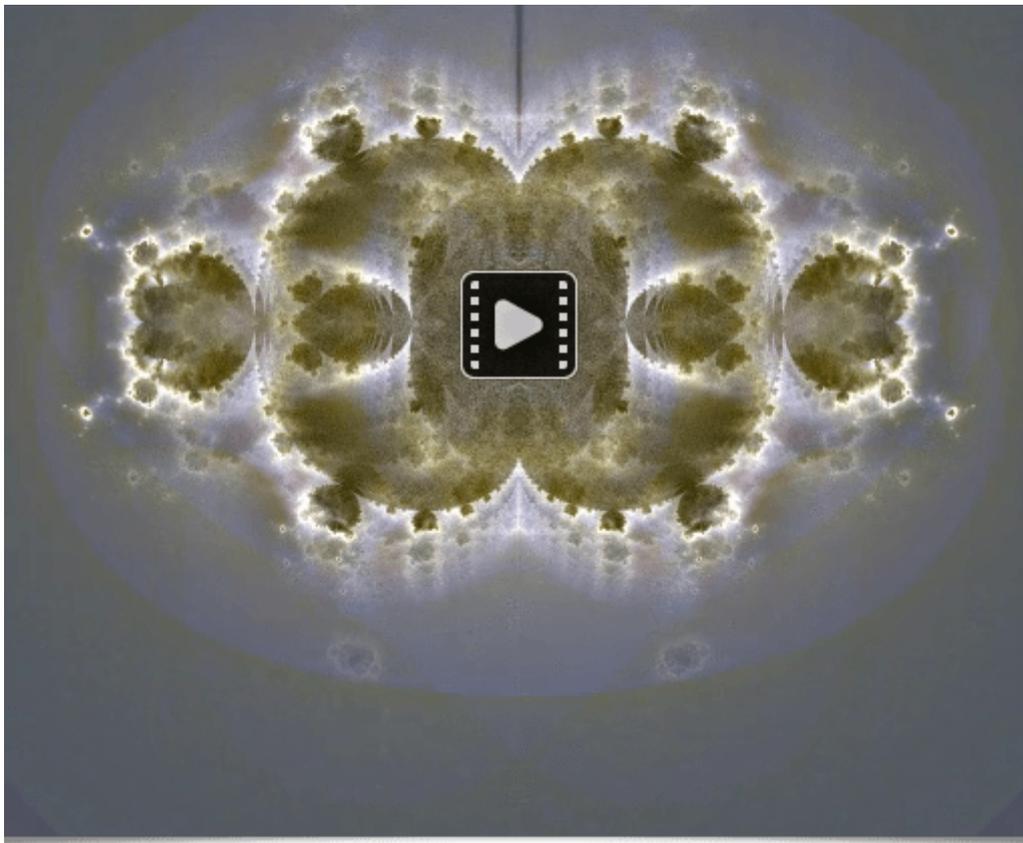


Fig.4.5. Variations fractales autour du Buddhabrot.

À noter que pour générer ces images du *Buddhabrot* modifié à bonne résolution, cela requiert déjà un temps de calcul important (quelques minutes par image). Je trouve cette animation intrigante : il est assez facile d'y retrouver/halluciner des formes familières quand on regarde en détail certaines *frames* de l'animation, un peu comme quand on regarde les nuages dans le ciel (si [vous croyez apercevoir](#) dans cette animation, une tête d'ours, un vieux monsieur assis, une silhouette de personnage, une tête de dragon... alors vous n'êtes pas les seuls à être un peu rêveurs ☺).

- **Conversion d'images pour le *String Art* :**

Le [String Art](#)^W est une forme de création artistique caractérisée par l'utilisation de fils de couleurs reliés entre des points (le plus souvent des clous) pour former un motif ou reproduire une photographie (classiquement des portraits). Dans un but d'expérimentation, nous avons écrit un script *G'MIC* qui cherche à transformer une image d'entrée en niveaux de gris choisie par l'utilisateur, sous la forme d'une série d'instructions à suivre pour relier des clous numérotés avec un fil monochrome, afin de reproduire l'image le plus fidèlement possible (en considérant les contraintes propres à ce mode de création) :



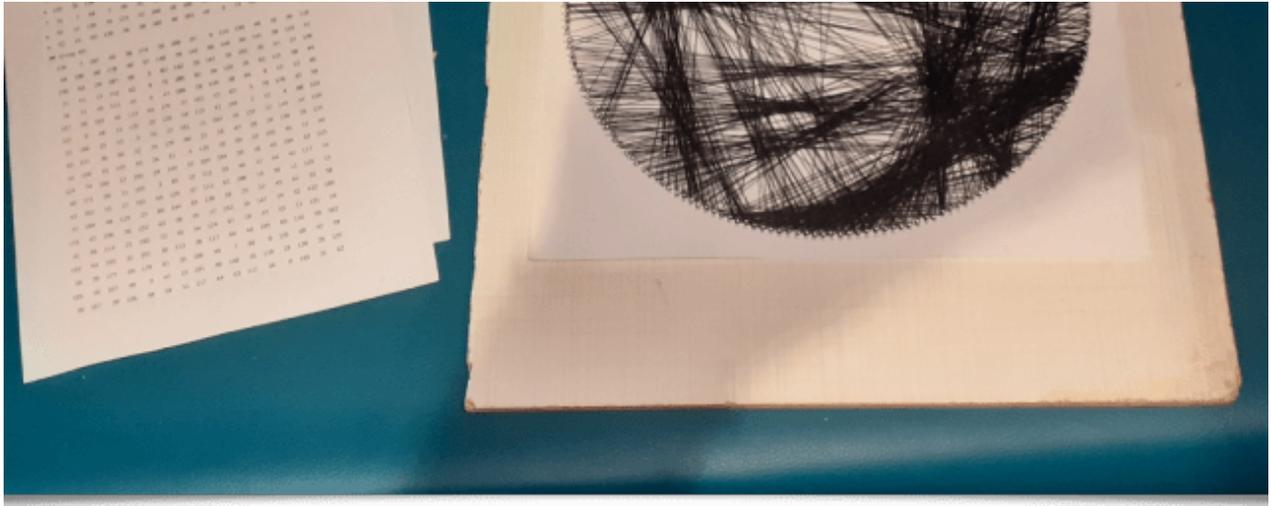


Fig.4.6. Tentative plus ou moins réussie d'utilisation de G'MIC pour le String Art.

Saurez-vous reconnaître qui est censé être représenté sur la photo ci-dessous (il n'y a rien de spécial à gagner...) ? Une tentative pas forcément couronnée de succès, qui mériterait d'être approfondie (notamment en gérant des fils de différentes couleurs), mais qui illustre cependant les possibilités multiples d'expérimentations que permet le cadriciel G'MIC !

- **Autres liens connexes :**

Pour finir, citons ces quelques liens en vrac, qui permettent d'apprécier la polyvalence de G'MIC :

1. **Stéganographie** : Ou comment cacher des données dans une image de bruit ? La commande `rand` est capable de générer des [valeurs aléatoires à densité \$w\$](#) , c'est-à-dire dont la densité de probabilité est une fonction spécifiée par l'utilisateur. Cela a par exemple comme application de cacher des images de taille 256x256 dans des histogrammes d'images de bruit à 16bits/canal, comme [détaillé sur cette page](#).
2. **Galleries artistiques** : [Ivelieu](#) et [Gannjondal](#) sont deux artistes fréquentant le site *Deviant Art* qui utilisent parfois G'MIC dans leur processus de création artistique. Allez faire un tour sur leurs galleries respectives, ça vaut le coup d'œil !

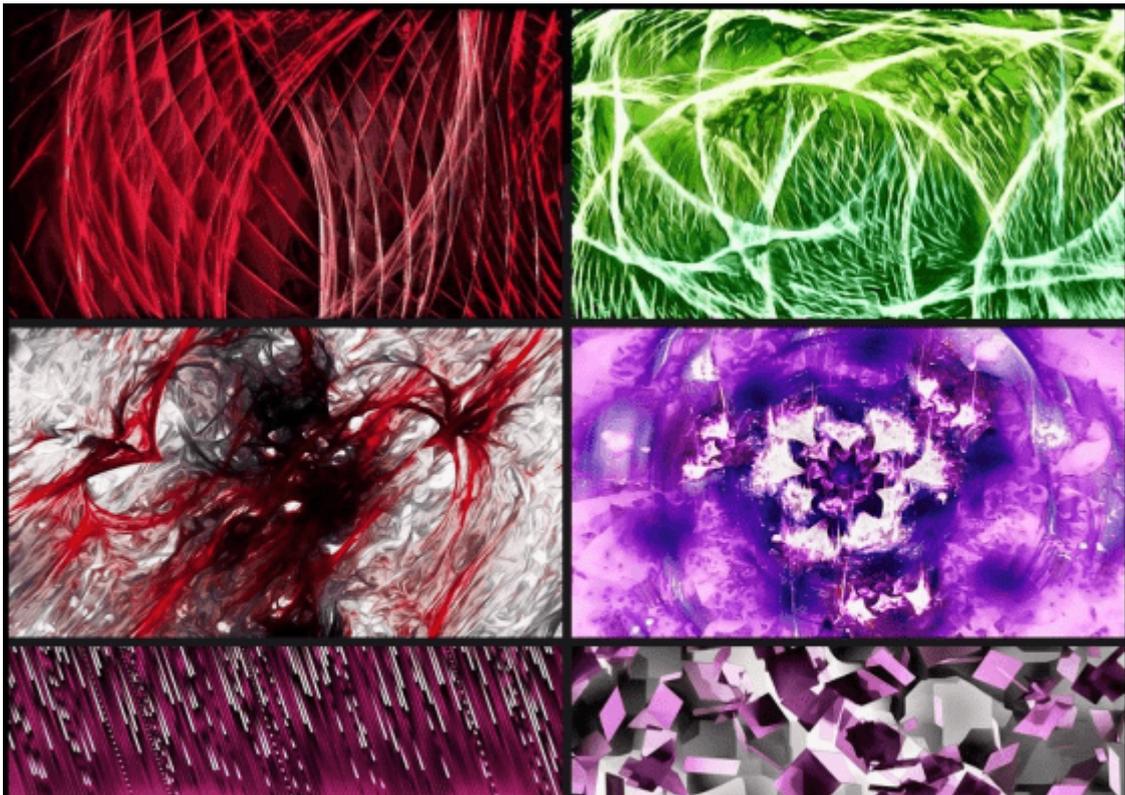




Fig.4.7. Quelques réalisations de Ivelieu / Devian Art.



Fig.4.8. Quelques réalisations de Gannjondal / Devian Art.

1. Nous avons appris que la prochaine version du programme de gestion de photos *digiKam* 8.4.0 **inclura un outil de traitement G'MIC** au sein du *Batch Queue Manager*, ce qui permettra aux utilisateurs de *digiKam* d'intégrer les filtres *G'MIC* dans les flux de travail de post-traitement.
2. **Thiojoe** est un Youtubeur produisant des vidéos autour de la technologie, et programmeur à ses heures. Il a commencé à développer un outil libre basé sur *G'MIC* pour générer facilement des animations, outil que vous pouvez **retrouver ici**. Ne fonctionne que sous *Windows* pour le moment, mais nous garderons un œil dessus.
3. Enfin, mentionnons la chaîne *Youtube* de **JustCallMeInsane**, illustratrice numérique utilisant *Krita*. Elle a récemment réalisé une série de vidéos qui explore les différentes catégories de filtres du greffon *G'MIC-Qt* pour *Krita*. On espère en passant que l'équipe de développement de *Krita*, qui gère sa propre version du greffon, pourra le mettre à jour assez rapidement.

5. Conclusion

On pourrait résumer cette dépêche en disant que le projet *G'MIC* « continue son petit bonhomme de chemin ».

G'MIC reste bien sûr un projet modeste, développé et maintenu par une petite équipe (de passionnés), mais avec un nombre d'utilisateurs croissant et des retours d'utilisation de plus en plus variés. Ce cadriciel a la chance de pouvoir être développé au sein du *GREYC*, un laboratoire de recherche publique assurant au projet un contexte de développement favorable, d'être soutenu par les tutelles du laboratoire (l'institut « Sciences Informatiques » du *CNRS*, l'Université de Caen et l'*ENSICAEN*) et même encouragé par la direction dudit laboratoire.

Et même si le plus gros du développement sur ce projet a probablement déjà été réalisé (plus de 15 ans de développement quand même), nous sommes confiants dans le fait que nous allons continuer à faire évoluer *G'MIC*, en tout cas, tant qu'on verra qu'il peut s'avérer utile aux traiteurs et traiteuses d'images de tout poil ! Et nous en reparlerons certainement dans une prochaine dépêche.

Aller plus loin