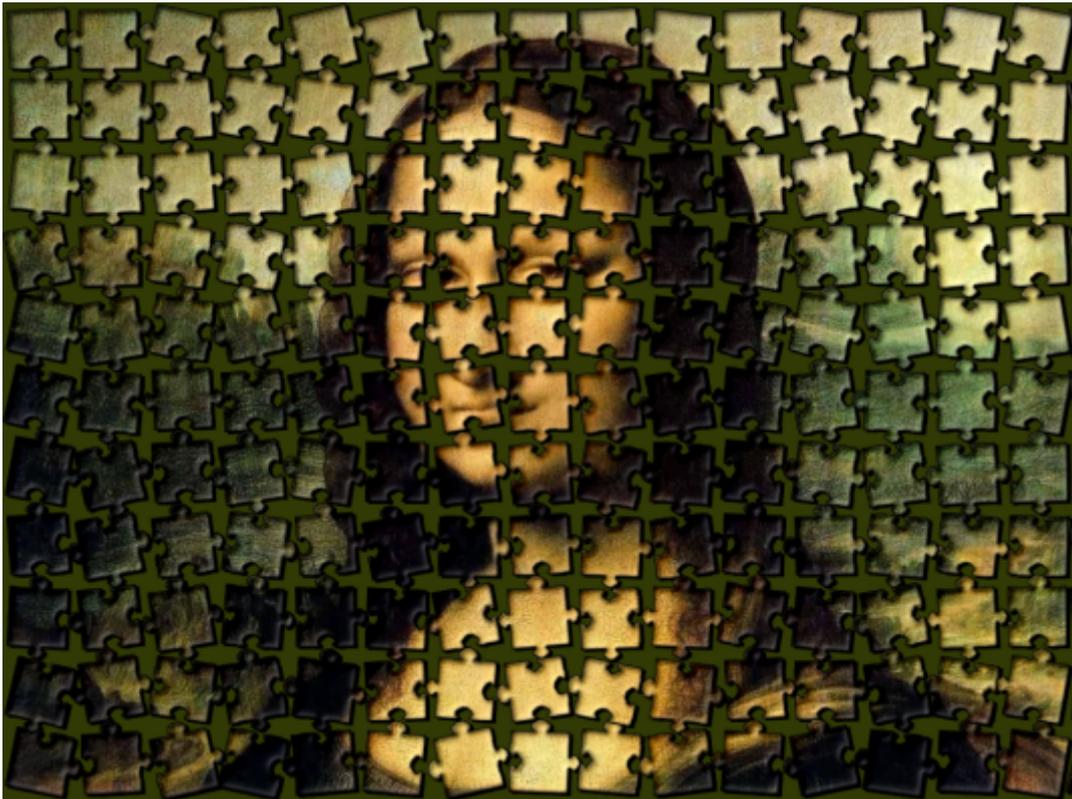


G'MIC 1.5.8.3 : Quelques avancées supplémentaires pour le traitement d'image libre

Posté par [David Tschumperlé \(site web personnel\)](#) le 12/02/14 à 11:58. Édité par 4 contributeurs. Modéré par [patrick_g](#). [Licence CC By-SA](#).

Étiquettes : [photo](#) , [gimp](#) , [g'mic](#) , [traitement_d'images](#) + [Étiqueter](#)

Je vais vous donner (encore!) quelques nouvelles du projet G'MIC, un framework libre pour le traitement d'image, que nous développons dans l'équipe IMAGE du laboratoire GREYC (UMR CNRS 6072), et qui est distribué sous licence libre CECILL. À l'occasion de la sortie de la dernière version (estampillée 1.5.8.3), je vous propose un récapitulatif des nouvelles fonctionnalités introduites depuis l'arrivée de la branche 1.5.8.x (c'est-à-dire juste quelque temps après la publication de mon [dernier journal](#) sur le sujet, fin octobre).



NdM : merci à David Tschumperlé pour son journal.

Sommaire

- - [G'MIC : Kesako ?](#)
 - [Les nouveaux effets disponibles.](#)
 - [Effet : Arrays & tiles / Puzzle](#)
 - [Effet : Artistic / Hope poster](#)
 - [Effet : Artistic / Polygonize](#)
 - [Effet : Frames / Frame \[blur\]](#)
 - [Effet : Repair / Inpaint \[patch-based\]](#)
 - [Effet : Rendering / Equation plot \[parametric\]](#)
 - [Effet : Rendering / Cupid](#)
 - [Effet : Rendering / 3d blocks](#)
 - [Effets : Repair / Smooth \[skin\] , Colors / Detect skin , Details / Split details et Details / Details equalizer](#)
 - [Effet : Deformation / Textured glass](#)
 - [Effet : Patterns / Op art](#)

- [Améliorations diverses, sous le capot](#)
- [Perspectives](#)

G'MIC : Kesako ?



Le projet libre [G'MIC](#) (acronyme de *GREYC's Magic for Image Computing*) propose différentes interfaces utilisateurs pour la manipulation d'images génériques (2d/3d, multi-spectrales,...), dont un [plug-in pour GIMP](#), un web-service [G'MIC Online](#) (qui a été [présenté sur linuxfr](#) en 2012), un outil en ligne de commande (`gmic`), ou encore une bibliothèque C++ (`libgmic`), et une interface graphique en Qt pour la manipulation d'images provenant de la webcam (`ZArt`).

Le [plug-in pour GIMP](#) étant aujourd'hui l'interface de G'MIC la plus utilisée, j'insisterais ci-dessous sur les fonctionnalités ajoutées pour cette interface. Mais bien sûr, gardez en mémoire que vous pouvez réaliser exactement les mêmes opérations en utilisant les autres interfaces disponibles (notamment, vous pouvez tout automatiser dans des scripts shell, ou disposer de tous ces filtres dans vos propres programmes si vous le souhaitez !).

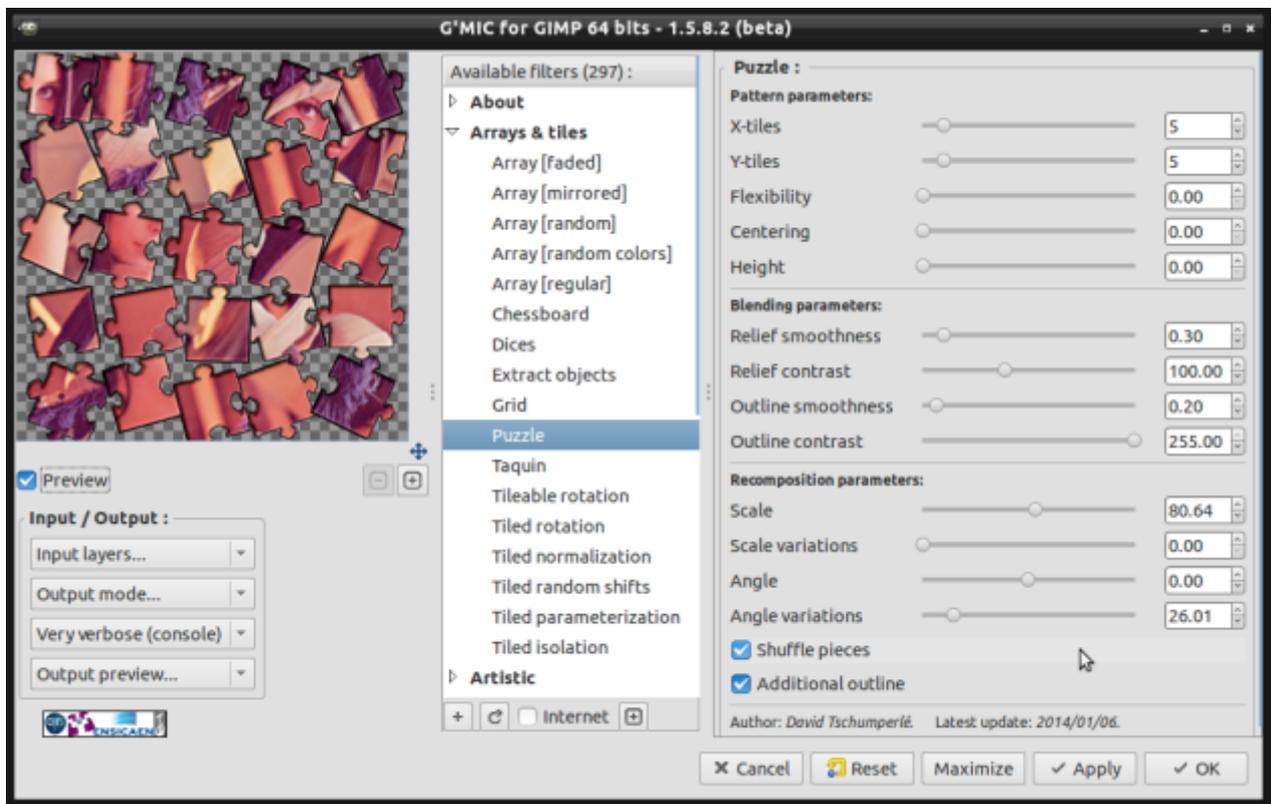
Les nouveaux effets disponibles.

Aujourd'hui, l'installation du greffon G'MIC pour GIMP et de la dernière mise à jour des filtres vous permet de disposer d'un total de **612** effets à appliquer sur vos images. Même si beaucoup de ces filtres sont encore en phase de *test* (rangés dans la section *Testing/*), il y a quand même de quoi s'amuser pendant quelque temps ! Voici un (long) résumé des effets ajoutés récemment. Vais-je battre le record du journal contenant le plus d'images ? Suspens !

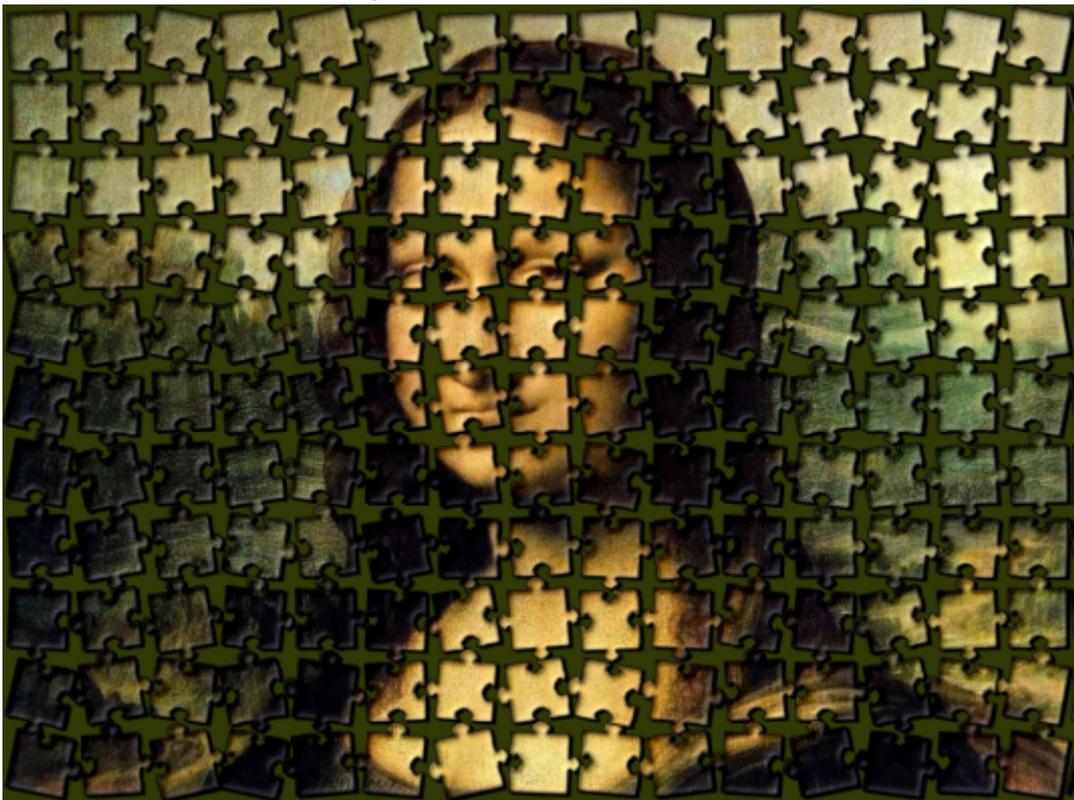
Effet : `Arrays & tiles / Puzzle`

Comme son nom l'indique, ce filtre convertit une image en puzzle.

"Mais, mais, mais, il y a déjà un effet de ce type par défaut dans GIMP !" vous entend-je crier au loin. C'est vrai (effet par défaut disponible dans `Filters / Render / Pattern / Jig-saw`), mais le filtre G'MIC va plus loin : il décompose *réellement* l'image en différentes pièces de puzzle. Il ne se contente pas d'ajouter un motif au dessus de l'image qui fait croire à un puzzle "tout fait". Cela nous permet en particulier de pouvoir appliquer des rotations ou du redimensionnement sur chaque pièce (avec un peu d'aléatoire si besoin est), et éventuellement de mélanger les pièces. On peut même faire en sorte que chaque pièce se place indépendamment sur un nouveau calque.



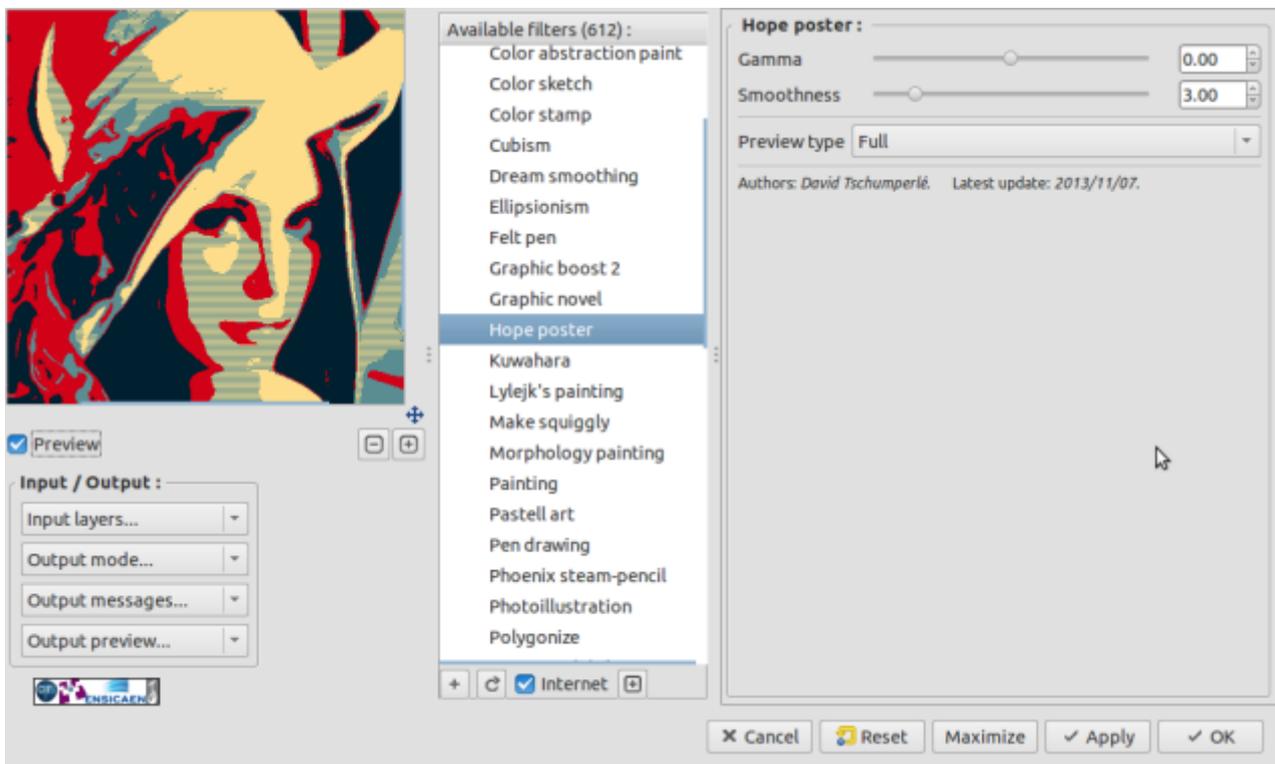
Du coup, on peut très rapidement faire ce genre de choses :



Sympa non ?

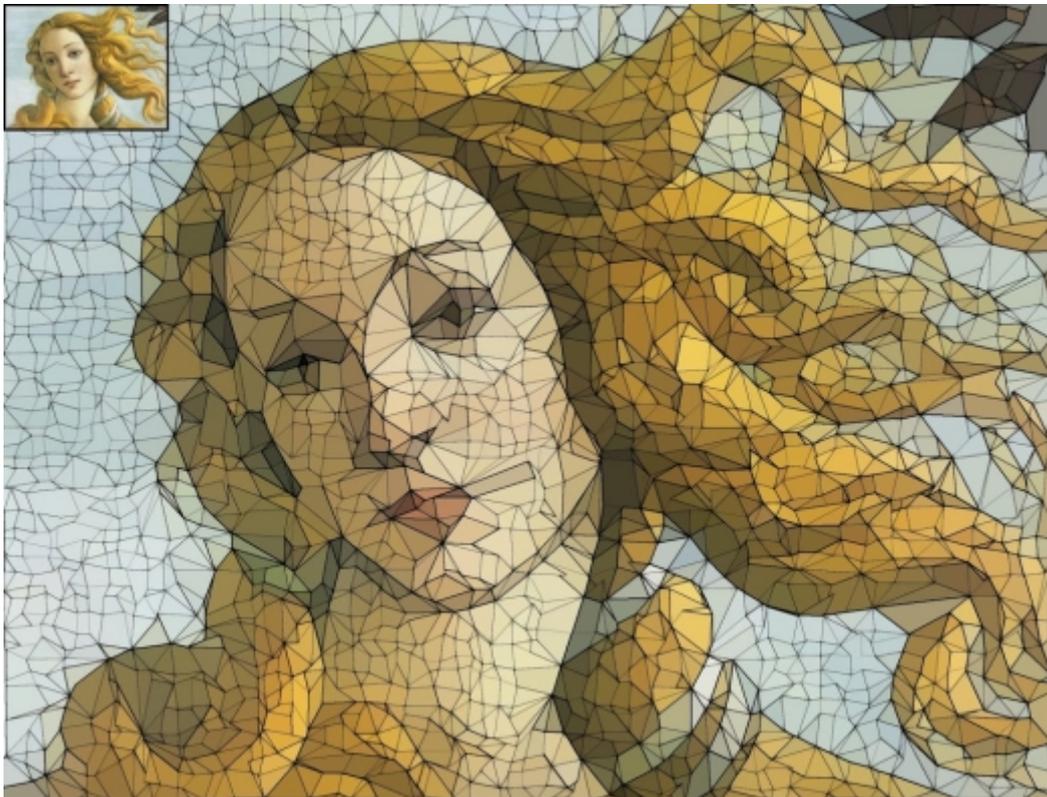
Effet :

Pas grand chose à dire, c'est un filtre qui tente de transformer votre image en un poster de campagne façon Barack Obama :)



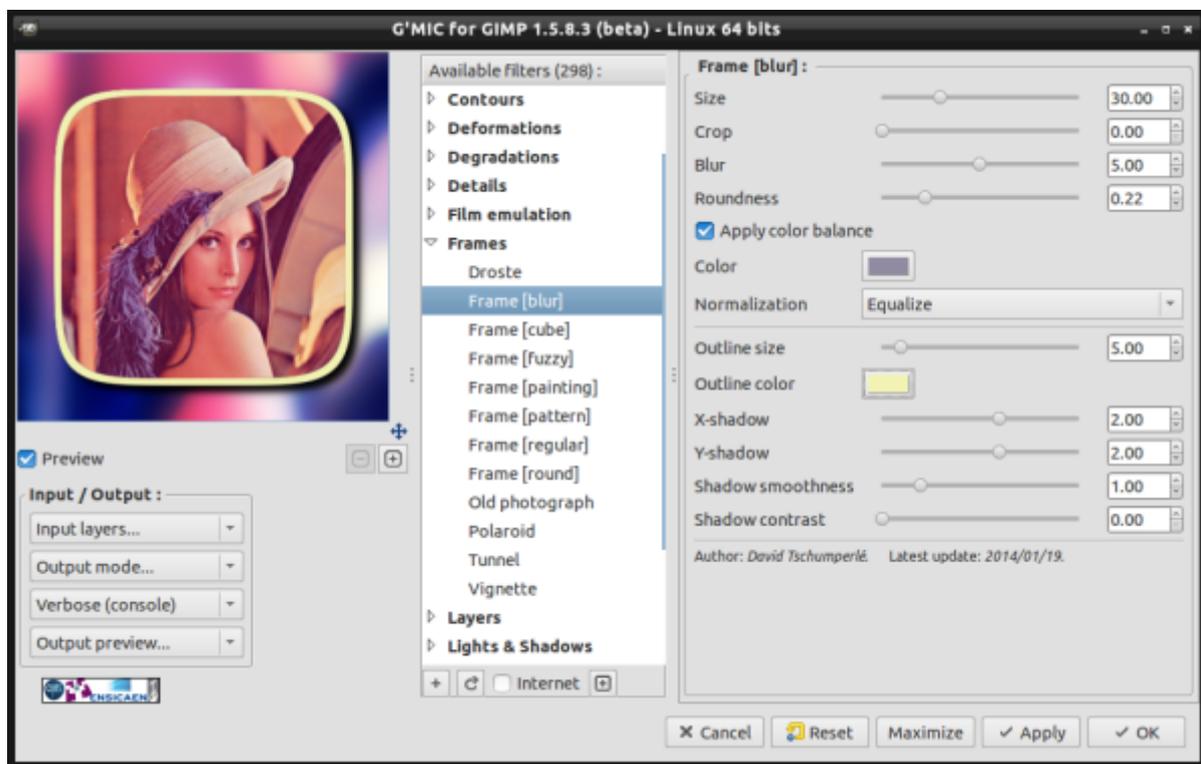
Effet : Artistic / Polygonize

Voilà un filtre amusant, qui essaye de transformer votre image en un ensemble de polygones de couleurs constantes, qui suivent plus ou moins les contours. Peut-être intéressant pour les créateurs de vitraux d'églises ! En pratique, on part d'un ensemble de points aléatoires connectés et on déplace ces points de façon à ce qu'ils collent aux contours présents dans l'image. Rien de bien compliqué, mais avec les bons paramètres, on peut vraiment générer de jolies choses (et éviter de faire ça à la main, tant qu'à faire).



Effet : Frames / Frame [blur]

Un petit filtre tout bête qui ajoute un cadre à une image, en prenant une version lissée de l'image comme référence pour remplir l'intérieur du cadre, avec éventuellement l'ajout d'un petit bandeau de couleur ainsi que de l'ombrage. Ici c'est pareil, ça fait gagner pas mal de temps pour un résultat rapide et sympa.



Effet :

Nous avons également beaucoup retravaillé sur l'algorithme d'_Inpainting_', qui permet de reconstruire de manière automatique des parties manquantes dans les images. J'avais déjà présenté cet algorithme sur Linuxfr, lors de [la sortie de G'MIC 1.5.5.1](#), mais les améliorations que nous avons récemment apportées à cet algorithme méritent qu'on en reparle un peu.

Notons d'abord que quelques résultats de ces travaux, réalisés en partie par [Maxime Daisy](#) et [Pierre Buysens](#) (respectivement doctorant et post-doc dans l'équipe IMAGE du [GREYC](#)) ont été publiés et présentés récemment à [SIGGRAPH Asia 2013](#) à Hong-Kong (voir [cette page de démo](#)), et les autres améliorations apportées plus récemment (notamment sur le temps de calcul mais aussi sur la qualité de la reconstruction) sont en cours de publication.

Patrick David (photographe amateur américain, proche de l'équipe de développement de G'MIC) en a d'ailleurs assez bien parlé sur un [article de son blog](#). De son point de vue, l'algorithme actuel d'inpainting disponible dans G'MIC est maintenant plus performant que celui du greffon [Resynthesizer](#) classiquement utilisé jusqu'à maintenant pour enlever des objets gênants dans des images sous GIMP.

Nos toutes dernières expériences montrent que nous rivalisons largement avec le fameux algorithme d'inpainting dit du *Content Aware Filling* disponible sur Photoshop, qui fait des choses équivalentes. Un exemple ci-dessous, où l'algorithme d'inpainting de G'MIC permet de reconstruire toute la partie masquée en rouge.

Masked

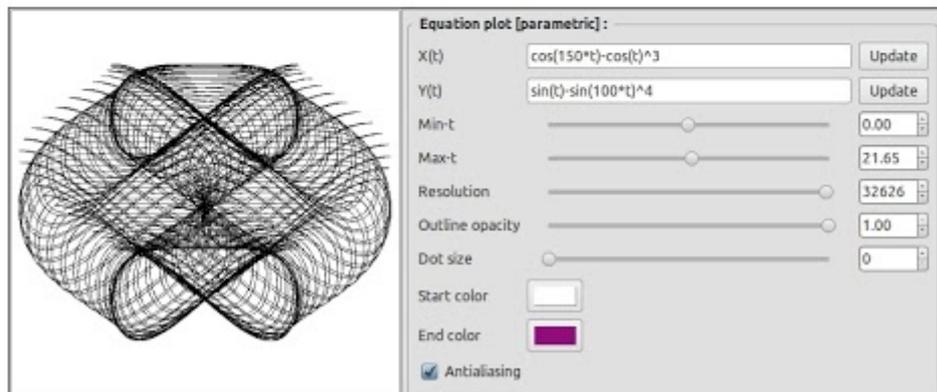
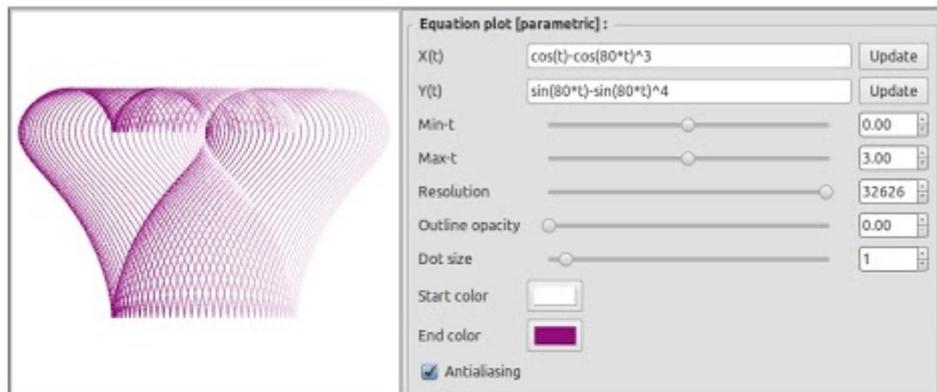
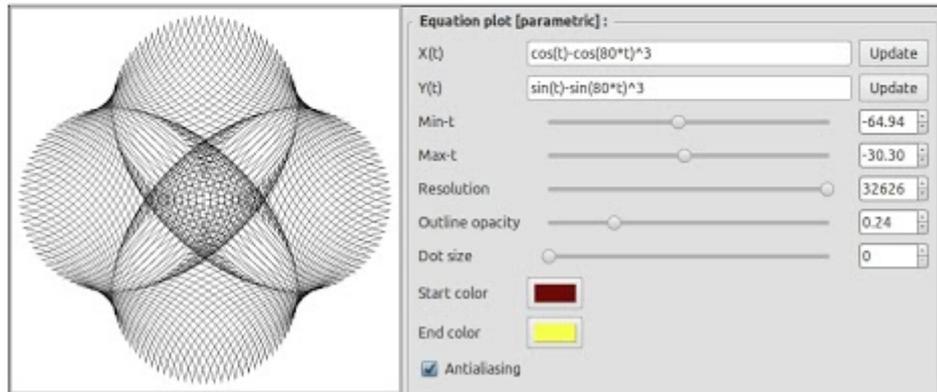
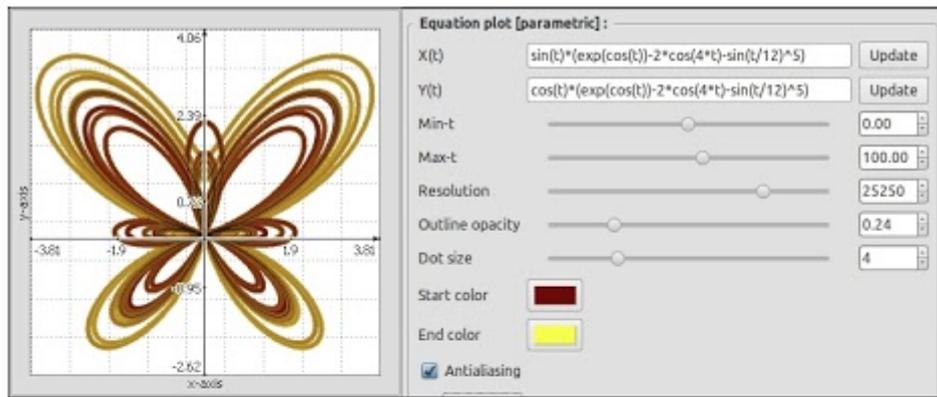


Inpainted



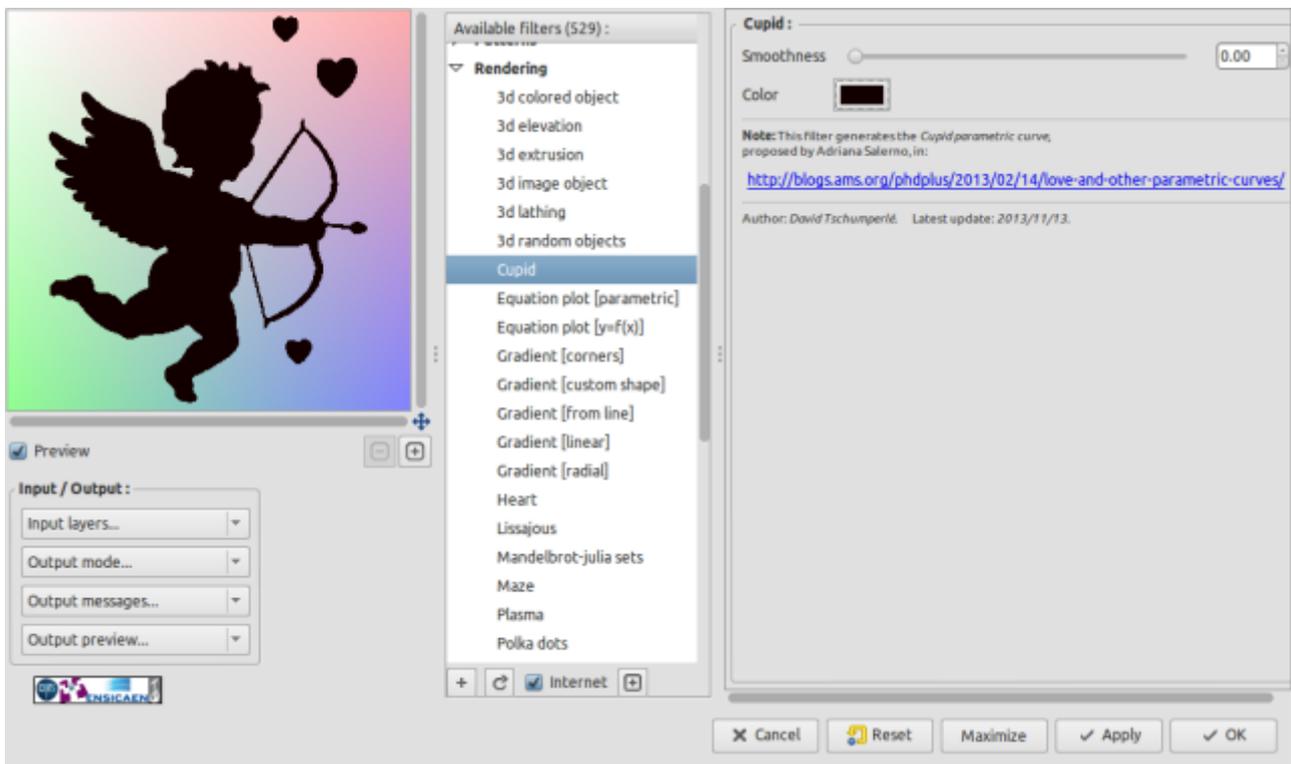
Effet : `Rendering / Equation plot [parametric]`

Vous rêvez de tracer de jolies courbes en restant sous GIMP ? Alors ce petit filtre est fait pour vous ! Il permet de tracer des équations paramétriques 2d dont vous donnez la formule. Voici quelques exemples :



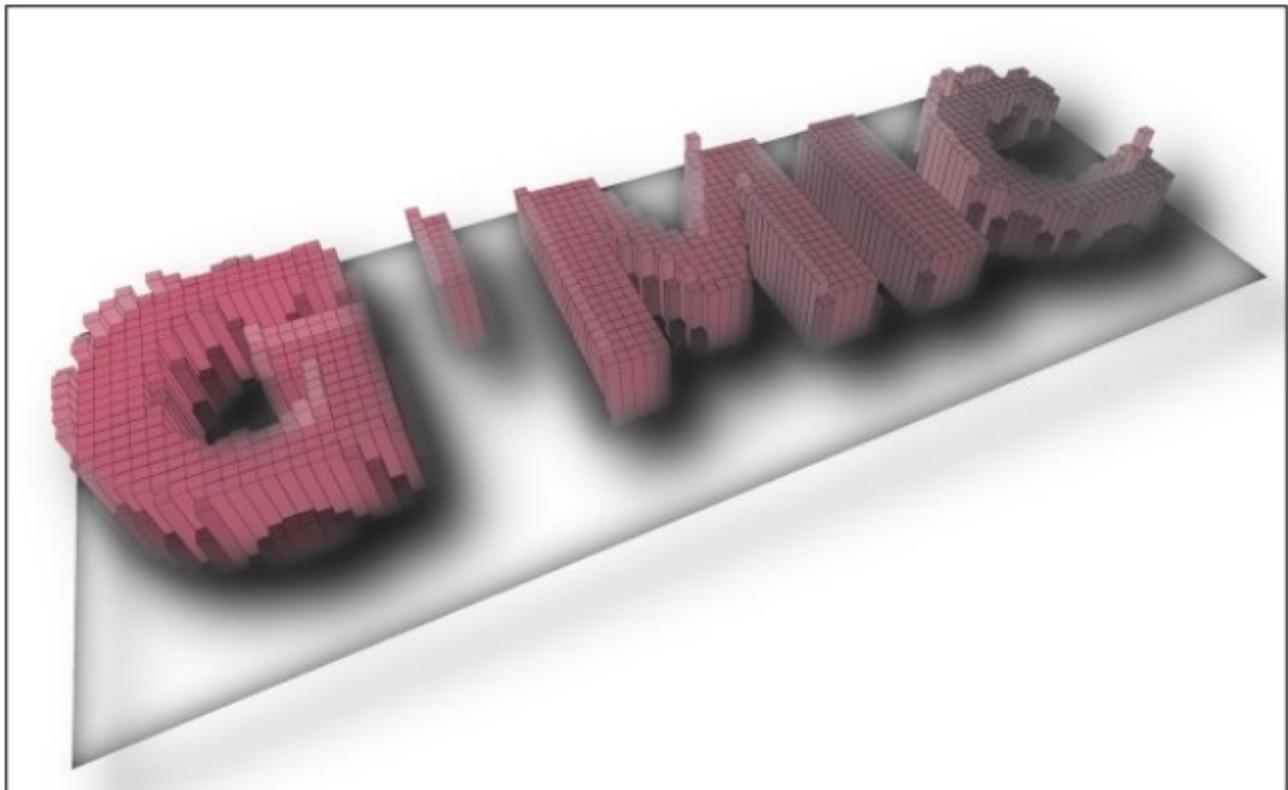
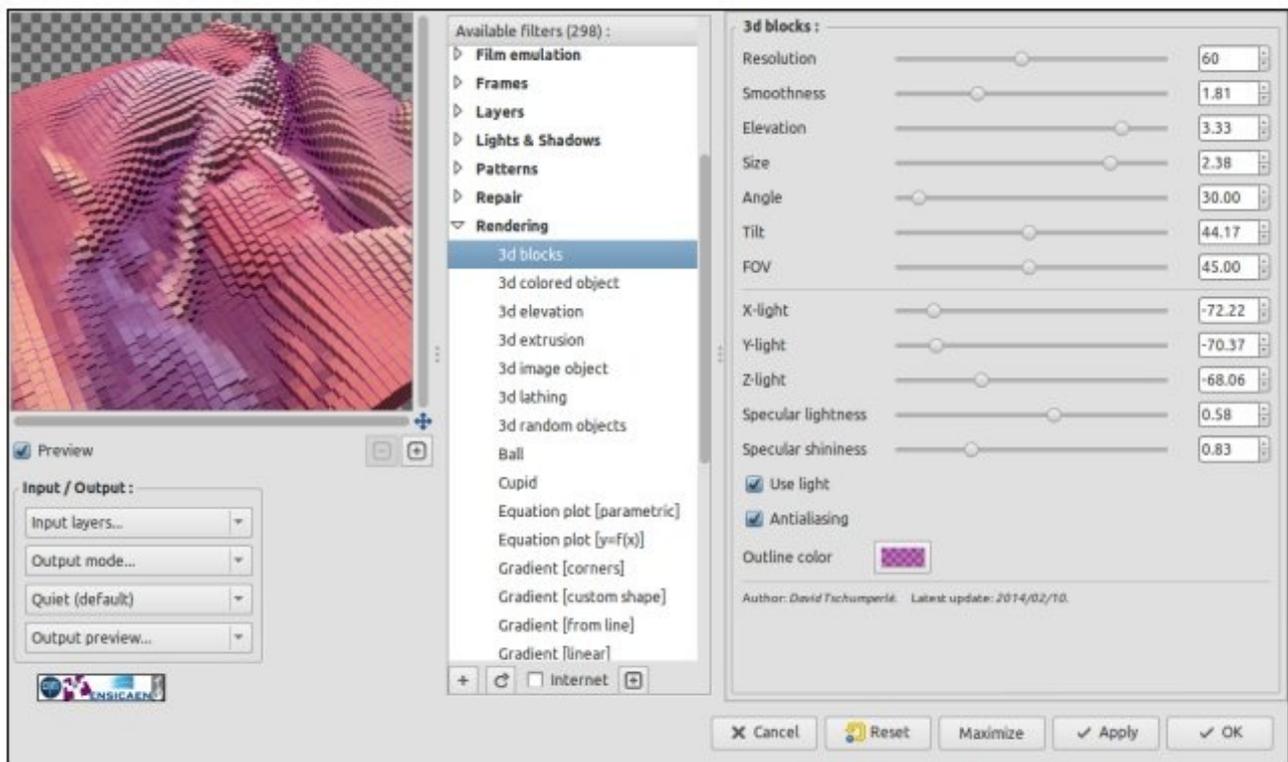
Effet : Rendering / Cupid

Et justement, une équation paramétrique remarquable (par sa longueur et par l'originalité de son tracé) est la fameuse [Cupid curve](#) (allez voir la formule, ça vaut le coup d'oeil!). Et pour vous éviter de vous retaper l'équation à la main, on vous a fait un filtre exprès. C'est pas gentil ça ? Oui ça sert à rien, et alors ! (enfin si, ça sert *in fine* à tester l'évaluateur d'équation intégré à G'MIC!).



Effet : Rendering / 3d blocks

Voilà un effet 3d sympa, qui transforme votre image en un champ de parallélépipèdes plus ou moins hauts, suivant la valeur de la luminance des pixels de votre image. Un filtre pour les amateurs de LEGO ?



Effets : `Repair / Smooth [skin]` , `Colors / Detect skin` , `Details / Split details` et `Details / Details equalizer` .

Ces 4 filtres ont été ajoutés presque simultanément. Le but initial était de pouvoir *lisser* la peau lors de la retouche de portraits, mais on peut aussi les utiliser dans d'autres cas bien entendu.

Le principe de base pour la retouche de peau sur des images de portraits repose sur deux fonctionnalités principales :

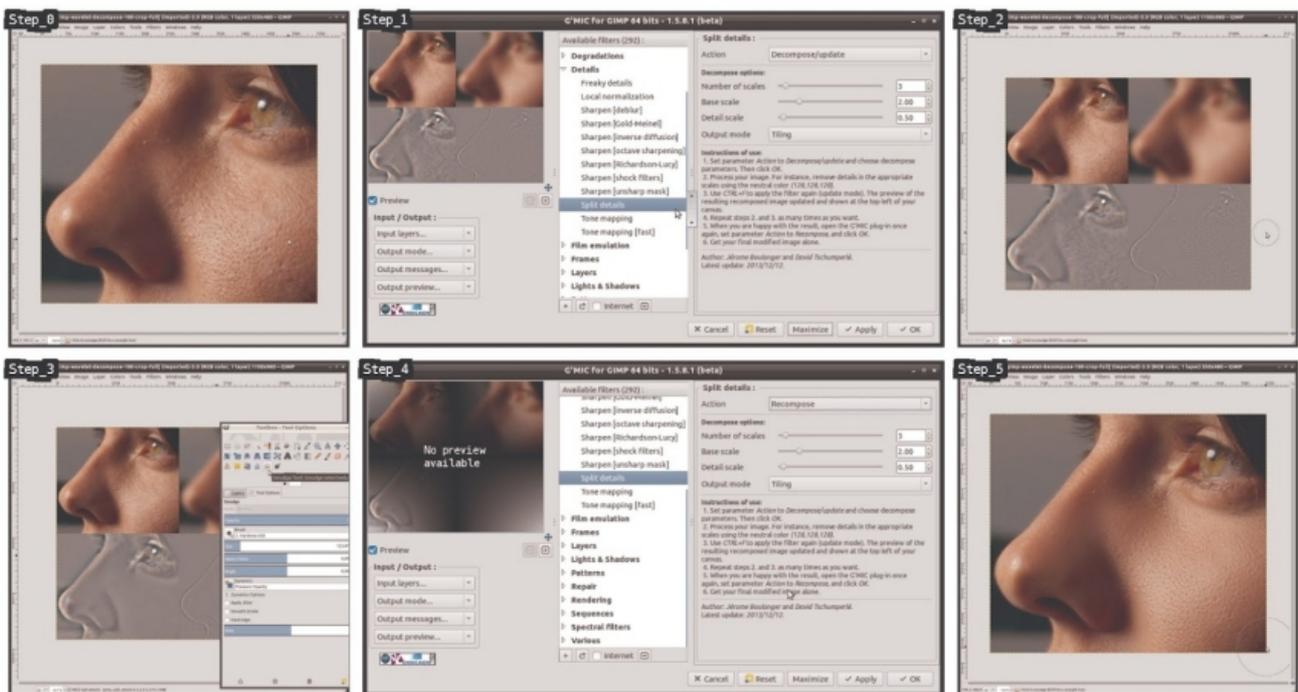
1. **La détection de peau**, pour générer un masque qui va permettre de ne pas traiter les autres parties d'une photographie (la création de ce masque est le but principal du filtre `Colors / Detect skin`). Cette étape se fait de manière assez rudimentaire, en calculant une carte de probabilité d'appartenance des pixels à la

couleur 'peau', définis comme les couleurs dont les chromaticités sont proches de chromaticités de référence (obtenus par apprentissage sur une base de données de pixels labellisés à la main). C'est une technique assez classique pour segmenter la peau dans les photos (et que j'ai vue aussi par ailleurs pour détecter des images à caractère pornographique!). La qualité du résultat dépend bien sûr pas mal de l'environnement dans lequel a été prise la photo (intérieur / extérieur), du type d'éclairage lors de la prise de vue, etc, mais en général c'est satisfaisant. Bien sûr, dans les cas où ça ne fonctionne pas, le filtre propose un mode 'manuel' qui permet à l'utilisateur de faire lui-même son apprentissage sur un échantillon de peau qu'il détermine 'à la main'.

2. Le lissage de la peau : cette partie doit se faire avec assez de précaution pour ne pas faire ressembler le modèle de votre portrait à un mannequin en plastique. Classiquement, les artistes qui font ça de manière propre décomposent d'abord l'image en plusieurs échelles de détails, et enlèvent des détails correspondants à une échelle moyenne (ni trop fine, ni trop grossière), de façon à garder les détails fins qui correspondent grosso-modo à la texture naturelle de la peau (pores).

On a donc ajouté toute une panoplie de filtres permettant de décomposer une image suivant plusieurs niveaux de détails, de pouvoir modifier ces détails indépendamment (soit manuellement, soit automatiquement), et de recomposer l'image finale à partir de chacun des niveaux de détails modifiés.

Bref, pour la retouche de peau, on peut maintenant faire des choses assez sympas dans G'MIC, en rendant le processus plus ou moins automatisé en fonction de son niveau d'expérience. Quelques exemples valent mieux que de longs discours (j'ai volontairement exagéré le lissage pour le rendre visible sur des petites images, mais bien sûr tout cela est réglable).

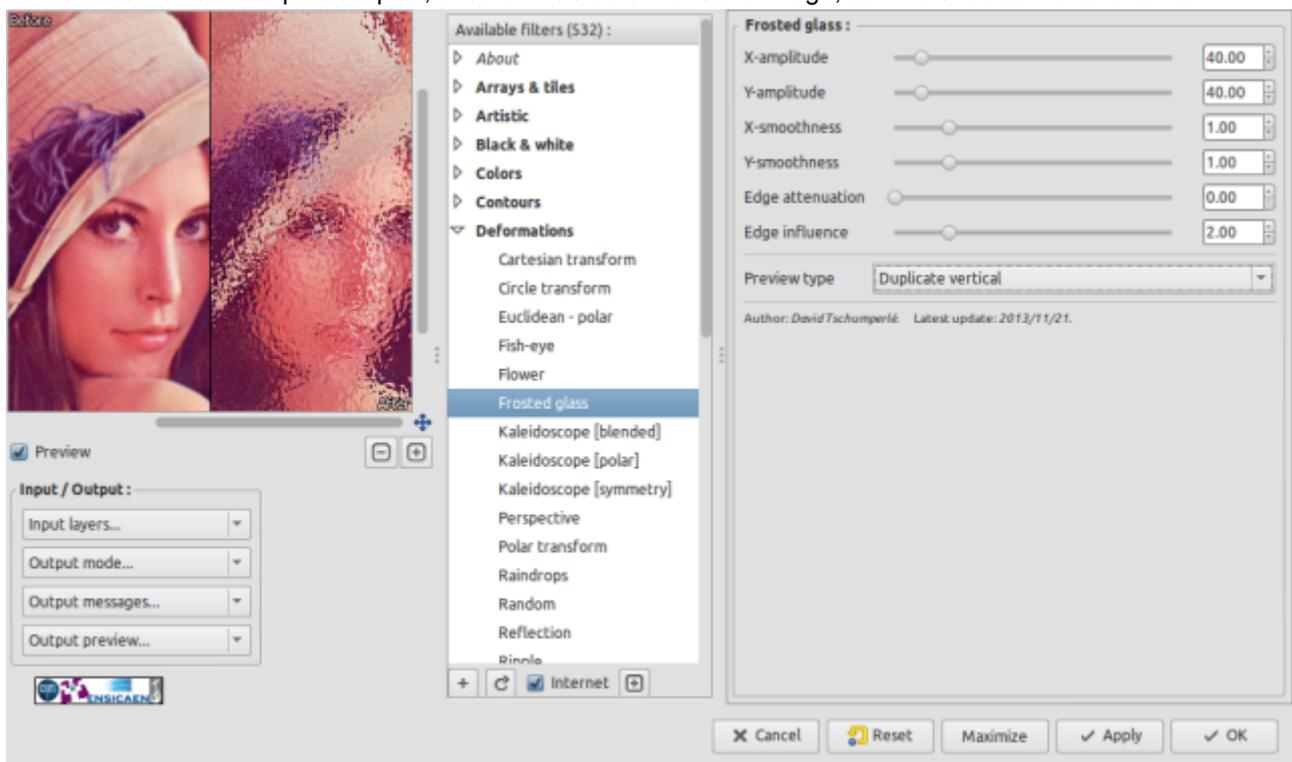


En poussant un peu la bête, on peut gagner facilement 10 ans ! (bon j'avoue j'ai un peu trop forcé sur les paramètres).



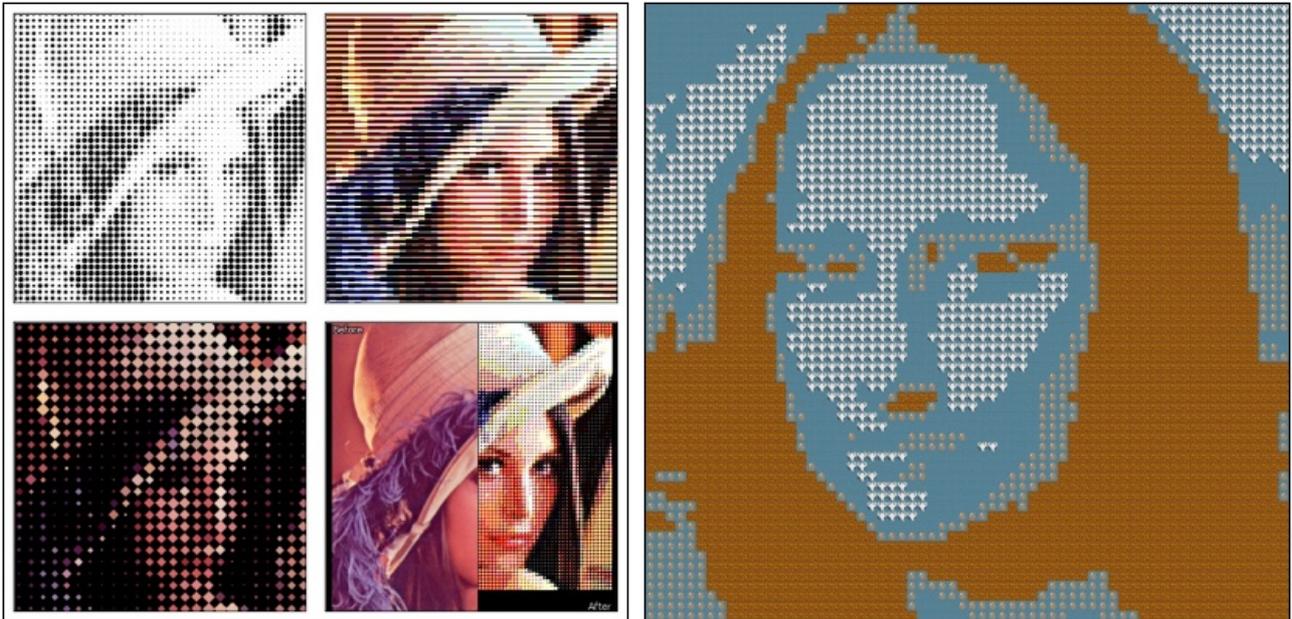
Effet : Deformation / Textured glass

On revient à des choses plus simples, avec un filtre déformant une image, comme avec du verre 'brouillé'.



Effet : Patterns / Op art

Ca c'est un filtre amusant et pouvant générer des effets très différents : il permet de remplacer chaque pixel de votre image, par une imagette quelconque. L'utilisateur peut spécifier les motifs d'entrée qu'il veut utiliser, ou choisir dans une liste de motifs prédéfinis. L'image ci-dessous illustre les deux façons de faire, avec des motifs prédéfinis (à gauche) ou définis par l'utilisateur (à droite, avec les sprites du jeu *Gauntlet*).



Améliorations diverses, sous le capot

Bien sûr tous ces filtres correspondent à la partie 'émergée' de l'iceberg. Un gros travail a été aussi réalisé sur d'autres améliorations moins visibles, l'optimisation de l'interpréteur de script interne à G'MIC, une meilleure stabilité (notamment pour les machines tournant sous MacOSX, merci valgrind!), etc...

Quelques démos supplémentaires ont été ajoutées, et vous pouvez les apercevoir en lançant la commande

```
$ gmic -demo
```

Perspectives

Bref, ces **4 derniers mois** passés ont été plutôt intenses pour l'amélioration du projet G'MIC, et on va essayer de maintenir le cap pour les prochains mois. Le projet est vraiment actif, et la vitesse avec laquelle les nouvelles fonctionnalités apparaissent est vraiment bonne pour le moment.

A noter que nous présenterons G'MIC à la conférence [Libre Graphics Meeting 2014](#), à Leipzig, avec une retrospective des 2 dernières années écoulées. Avec 20 minutes prévues pour la présentation, autant dire que nous allons manquer un peu de temps :)

Pour en finir avec ce long journal (le record a-t-il été battu ?), je voudrais signaler qu'on a organisé il y a quelques temps un challenge spécial G'MIC, uniquement pour le fun.

Ce modeste challenge intitulé [From scratch challenge](#) consistait à proposer des images dont le contenu avait été généré uniquement par l'application de filtres G'MIC (à partir d'une image blanche ou noire en entrée). C'est plutôt intéressant de voir la variété des résultats proposés (59 images ont été soumises pour ce challenge), et de se dire que quand même, on peut en faire des choses avec G'MIC !

Toutes les images de ce challenge [sont disponibles ici](#).

Merci de votre attention, et à dans quelques mois pour la suite de nos aventures.

Aller plus loin

-  [Journal à l'origine de la dépêche](#) (225 clics)
-  [Images du From scratch challenge](#) (98 clics)
-  [Libre Graphics Meeting 2014](#) (40 clics)
-  [G'MIC online](#) (495 clics)
-  [Le site de G'MIC](#) (861 clics)
-  [Dépêche de mars 2013](#) (48 clics)