



Guide de la démo « Illusions graphiques »

Espace LOGIN – espace dédié aux sciences du numérique

08-07-21

Inria

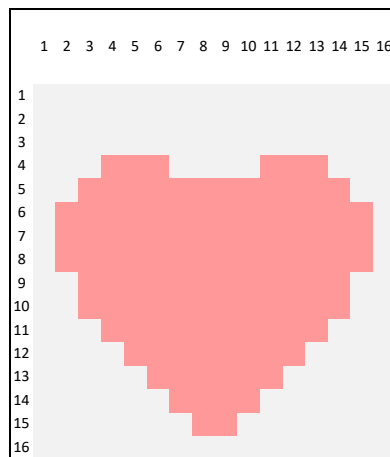
SOMMAIRE

1. Images et algorithmique	3
1.1. COMMENT UN ORDINATEUR LIT LES COULEURS ET COMMENT SONT-ELLES CODEES ?	3
1.2. COMMENT UN ALOGORITHME PEUT-IL CHANGER LES COULEURS D'UNE IMAGE ?	4
	5
1.3. ENJEUX DES RECHERCHES DE L'EQUIPE MAVERICK	6
2. Illusions graphiques	7
2.1. LE DEMONSTRATEUR	7
2.2. MODIFICATION DU STYLE GRAPHIQUE	7
2.3. MODIFICATION DES COULEURS	9
2.4. CHANGEMENT DE FORMES ET CARICATURES	13
<i>2.4.1. Modifier l'impression de relief</i>	
3. Conclusion	17
4. Sources et liens utiles	18
<i>4.1.1. Mieux comprendre le pixel</i>	
<i>4.1.2. Un peu de thèses...</i>	
<i>4.1.3. Un peu de programmation...</i>	

1. Images et algorithmique

1.1. COMMENT UN ORDINATEUR LIT LES COULEURS ET COMMENT SONT-ELLES CODEES ?

Un ordinateur ne voit pas les images. Comme tout ce qu'il manipule, il ne les comprend que comme des fichiers, de très longues suites de 1 et de 0. Une image est stockée sous la forme d'un texte, ou plus précisément d'une série de chiffres stockés dans un grand tableau.



Sur cette image, on reconnaît assez facilement un cœur. Mais un ordinateur ne peut pas reconnaître une forme de lui-même. Pour lui, une image n'est qu'un tableau rempli de valeurs. Chaque case de ce tableau contient un code couleur, qui définit la couleur de ce point précis de l'image.

Dans un ordinateur, la couleur est représentée sous la forme d'un code, trois valeurs assemblées : une valeur pour le Rouge, une valeur pour le Vert, et une valeur pour le Bleu. Chacune de ces valeurs va de 0 à 255, 0 étant une absence totale de cette couleur, et 255 étant son maximum.

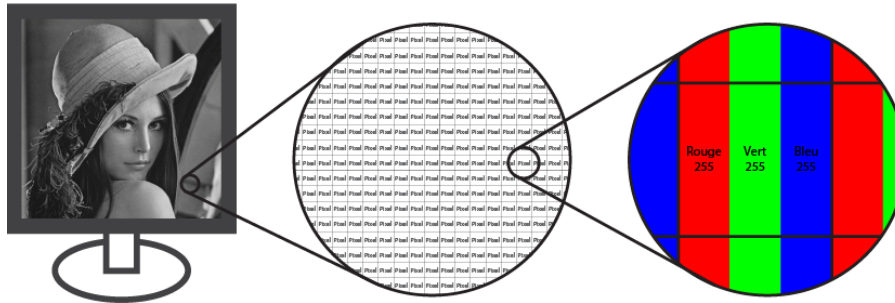
L'image n'est ainsi qu'un tableau, dans lequel on retrouve les deux couleurs de notre image :

[Red=0, Green=0, Blue=0] = Blanc

[R 255, G 0, B 0] = Rouge

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
2	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
3	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
4	00	00	00	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255
5	00	00	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255
6	00	00	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255
7	00	00	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255
8	00	00	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255
9	00	00	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255
10	00	00	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255
11	00	00	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255
12	00	00	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255
13	00	00	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255
14	00	00	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255
15	00	00	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255	R255
16	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

L'ordinateur envoie ces codes à l'écran auquel il est relié et qui va afficher la couleur demandée sur chacun de ses pixels. Pour l'écran LCD dont la technologie n'est pas capable d'afficher plusieurs couleurs à partir du même composant, chaque pixel est subdivisé en trois composants, chacun émettant l'une des trois couleurs primaires.



1.2. COMMENT UN ALGORITHME PEUT-IL CHANGER LES COULEURS D'UNE IMAGE ?

Demander à un ordinateur de modifier une image, c'est fondamentalement lui demander de modifier les valeurs de tous ou une partie de ces pixels. Par exemple, changeons la couleur de ce cœur pour le rendre jaune.

Le jaune correspond à un mélange de rouge et de vert. Je dois donc demander à mon ordinateur d'augmenter la valeur Green de chacun des pixels de mon cœur. Pour ce résultat :
 [Red=0, Green=0, Blue=0] = Blanc

[R 255, G 255, B 0] = Jaune

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0
2	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0
3	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0
4	R0 G0 B0	R0 G0 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0
5	R0 G0 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0
6	R0 G0 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0
7	R0 G0 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0
8	R0 G0 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0
9	R0 G0 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0
10	R0 G0 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0
11	R0 G0 B0	R0 G0 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0
12	R0 G0 B0	R0 G0 B0	R0 G0 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0
13	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0
14	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0
15	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0	R255 G255 B0
16	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0	R0 G0 B0

Un ordinateur étant très efficace pour des tâches mathématiques répétitives, on peut assez facilement créer des dégradés de couleur en modifiant par étape la valeur de chaque cellule. Pour créer un changement de couleur, on peut modifier chaque ligne du tableau l'une après l'autre et augmenter progressivement la valeur Green :

Pour chaque ligne du tableau :

Sélectionner tous les pixels du cœur :

Et augmenter un peu la valeur Green

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO
2	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO
3	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO
4	RO GO BO	RO GO BO	RO GO BO	255 255 255	255 255 255	255 255 255	RO GO BO	RO GO BO	RO GO BO	RO GO BO	255 255 255	255 255 255	255 255 255	RO GO BO	RO GO BO	RO GO BO
5	RO GO BO	RO GO BO	250 250 250	250 250 250	250 250 250	250 250 250	250 250 250	250 250 250	250 250 250	250 250 250	250 250 250	250 250 250	250 250 250	RO GO BO	RO GO BO	RO GO BO
6	RO GO BO	225 225 225	225 225 225	225 225 225	225 225 225	225 225 225	225 225 225	225 225 225	225 225 225	225 225 225	225 225 225	225 225 225	225 225 225	225 225 225	RO GO BO	RO GO BO
7	RO GO BO	200 200 200	200 200 200	200 200 200	200 200 200	200 200 200	200 200 200	200 200 200	200 200 200	200 200 200	200 200 200	200 200 200	200 200 200	200 200 200	RO GO BO	RO GO BO
8	RO GO BO	175 175 175	175 175 175	175 175 175	175 175 175	175 175 175	175 175 175	175 175 175	175 175 175	175 175 175	175 175 175	175 175 175	175 175 175	175 175 175	RO GO BO	RO GO BO
9	RO GO BO	RO GO BO	150 150 150	150 150 150	150 150 150	150 150 150	150 150 150	150 150 150	150 150 150	150 150 150	150 150 150	150 150 150	150 150 150	RO GO BO	RO GO BO	RO GO BO
10	RO GO BO	RO GO BO	125 125 125	125 125 125	125 125 125	125 125 125	125 125 125	125 125 125	125 125 125	125 125 125	125 125 125	125 125 125	125 125 125	RO GO BO	RO GO BO	RO GO BO
11	RO GO BO	RO GO BO	RO GO BO	100 100 100	100 100 100	100 100 100	100 100 100	100 100 100	100 100 100	100 100 100	100 100 100	100 100 100	100 100 100	RO GO BO	RO GO BO	RO GO BO
12	RO GO BO	RO GO BO	RO GO BO	RO GO BO	75 75 75	75 75 75	75 75 75	75 75 75	75 75 75	75 75 75	75 75 75	75 75 75	RO GO BO	RO GO BO	RO GO BO	RO GO BO
13	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	50 50 50	50 50 50	50 50 50	50 50 50	50 50 50	50 50 50	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO
14	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	25 25 25	25 25 25	25 25 25	25 25 25	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO
15	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	0 0 0	0 0 0	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO
16	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO	RO GO BO

On progresse ainsi peu à peu du rouge vers le jaune. Les étapes intermédiaires nous montrent un aperçu de la diversité des couleurs possibles.

255 valeurs possibles pour trois variables, cela signifie 255x255x255 possibilités ; Soit plus de 16 millions de couleurs disponibles.

L'algorithme qui me permet ici de générer ce dégradé est extrêmement simple. Mais on peut travailler à des modifications plus subtiles ou plus complexes selon le même principe. Un algorithme n'est alors qu'une suite d'instructions modifiant les valeurs de tous les pixels ou seulement d'une partie des pixels de l'image. On peut changer sa couleur, augmenter ou réduire son contraste.

Ce sont ces algorithmes que l'on retrouve dans les logiciels de traitement d'image.

1.3. ENJEUX DES RECHERCHES DE L'ÉQUIPE MAVERICK

Une partie de l'équipe de recherche MAVERICK du centre INRIA de Grenoble est de créer des algorithmes de ce type et faire en sorte qu'ils soient les plus efficaces, réalistes et utiles à un objectif précis.

L'équipe Maverick travaille sur les images et les vidéos numériques et sur leur qualité de transmetteur d'un message. Les chercheurs partent du principe qu'une image, quel que soit son but (Représentation de données, Narration, Simulation, Esthétique, Visualisation Scientifique, etc.) va être vue par des humains et donc poser le même problème de recherche c'est-à-dire s'assurer que le message et les caractéristiques importantes de l'image soient perçues.

Il s'agit d'étudier notre rapport à une image et d'en déduire des caractéristiques fondamentales sur notre perception :

- Comment fait-on la différence entre une image calculée par ordinateur et une image filmée ?
- Comment reconnaît-on le style de la peinture utilisée dans un tableau ?
- Qu'est-ce qui attire notre regard en premier dans une image ?

2. Illusions graphiques

2.1. LE DEMONSTRATEUR

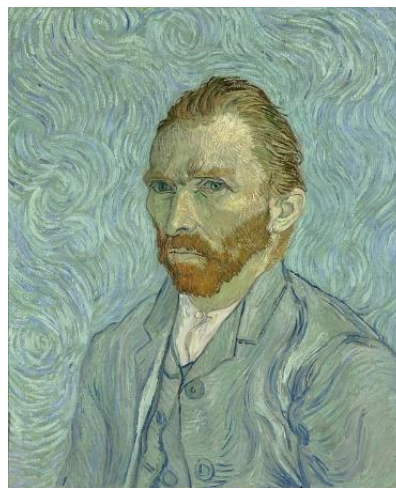
Le démonstrateur *Illusions Graphiques* vous donne l'occasion d'utiliser des algorithmes de modification d'image numérique c'est-à-dire des façons de modifier les valeurs de tous ou partie des pixels d'une image.

Contrairement à notre cœur qui était très petit (16 pixels de large par 16 pixels de haut), nous travaillons sur notre propre image, prise à l'instant, avec une résolution de 1080x1080 pixels. Pour l'ordinateur, ce ne sont que deux tableaux, l'un étant beaucoup plus grand que l'autre.

Les algorithmes appliqués sont également beaucoup plus complexes...

2.2. MODIFICATION DU STYLE GRAPHIQUE

Lorsque nous regardons une image, nous sommes capables assez facilement de deviner son origine. On peut distinguer une photographie d'une peinture, d'un dessin, d'une image calculée par ordinateur, etc.



Nous arrivons à distinguer la source de ces images par expérience car notre cerveau reconnaît des caractéristiques propres à cette technique. Nous voyons immédiatement le grain du papier d'un dessin, l'absence de reflets complexes d'une image calculée par ordinateur, le photoréalisme d'une photographie, etc.

En identifiant ces indices visuels, il est possible de faire des images virtuelles de plus en plus réalistes qui copient les caractéristiques et les indices des images réelles. A partir des éléments caractéristiques de l'aquarelle, le démonstrateur *Illusions Graphiques* peut ainsi transformer une image pour nous faire croire que c'est une aquarelle.

Il a fallu pour cela identifier les caractères reconnaissables de l'aquarelle et en faire une liste de traitement à appliquer. Il faut penser à la diffusion de l'eau sur le support, la granularité et les imperfections du papier, les traits de brouillon visible sous le dessin, etc. A chaque caractéristique correspond un traitement de l'image et donc un algorithme à créer avec un but spécifique pour passer de la photo d'une montagne à l'aquarelle d'une montagne.

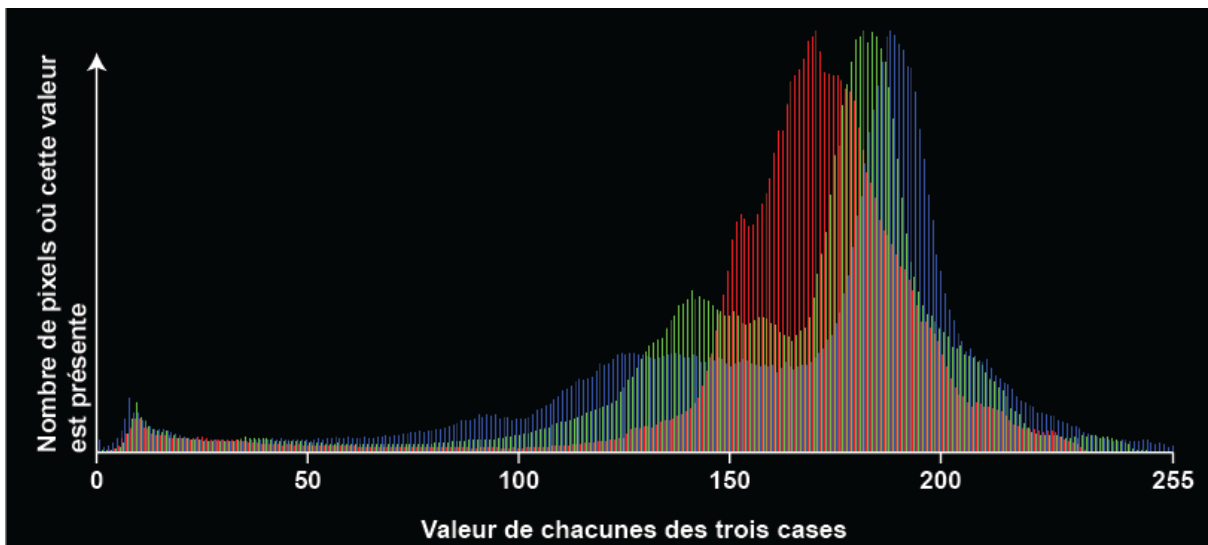
Vous pouvez retrouver dans « Liens utiles », la recette de l'aquarelle en une version simplifiée à réaliser chez soi :



©http://maverick.inria.fr/Membres/Adrien.Bousseau/watercolor_tutorial/tutorial1.pdf

2.3. MODIFICATION DES COULEURS

	Valeur Red	Valeur Green	Valeur Blue
Pixel 1	155	160	163
Pixel 2	156	161	164
Pixel 3	158	163	164
...			
Pixel 786 432	146	151	147



© [Color histogram - Wikipedia](#)

Faire l'histogramme d'une image, c'est montrer de quelles couleurs l'image est principalement constituée. Ainsi, l'histogramme de notre chat nous montre qu'une majorité de pixels ont des valeurs situées au-dessus de 150. On peut en déduire, sans même voir l'image, qu'elle est lumineuse, proche du blanc, car la plupart des valeurs de ses pixels en sont proches.

Cet histogramme représente l'ambiance de l'image.

L'histogramme d'une image est très utile lorsque l'on cherche à la modifier, pour l'éclaircir ou l'assombrir, pour augmenter ou réduire son contraste, pour changer l'impression globale qu'elle renvoie. Voyons quelques exemples à partir d'une autre image :

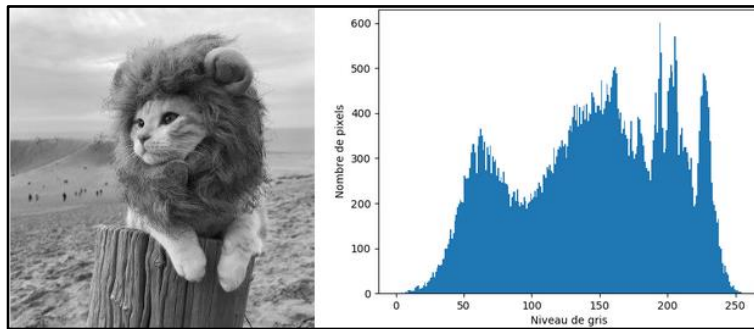


Image équilibrée : l'histogramme occupe la majorité des niveaux de gris possibles

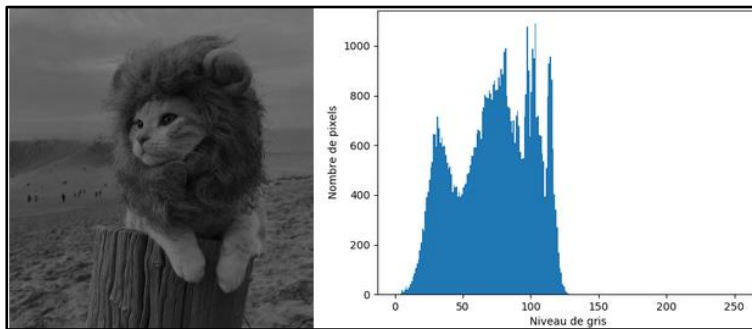


Image sous exposée (pas assez de lumière, trop sombre) : l'histogramme n'occupe qu'une partie des couleurs possibles, les plus sombres

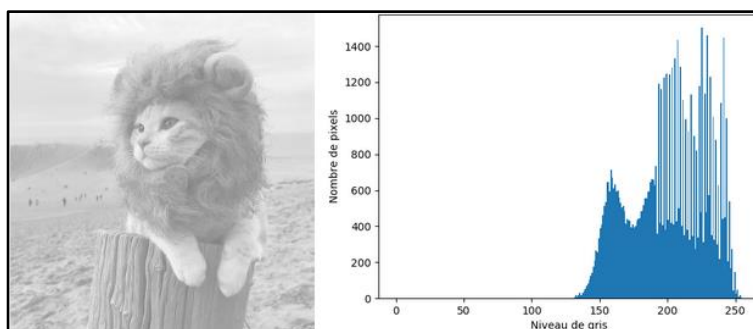


Image surexposée (trop de lumière, pas assez de noir) : l'histogramme n'occupe qu'une partie des couleurs possibles, les plus claires

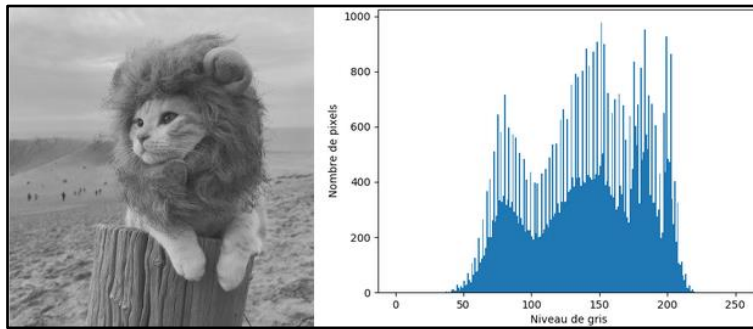


Image sans contraste (terne) : l’histogramme n’occupe que le centre, il y a peu de différences entre les couleurs de l’image

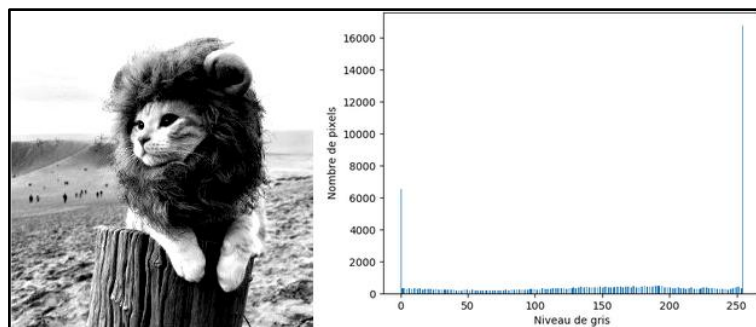


Image trop contrastée (agressive) : l’histogramme est réparti équitablement, toutes ls couleurs sont également utilisées

© [Comprenez comment un ordinateur voit une image - Classez et segmentez des données visuelles - OpenClassrooms](#)

Dans Photomaton, nous pouvons utiliser ces histogrammes pour appliquer les couleurs d’une image à une autre image. Une technique de transformation, le Color Transfer de Reinhard et collègues, a été développé à l’université de Utah dans les années 2000. Elle permet d’appliquer l’histogramme colorimétrique d’une image à une autre image, sans en modifier les formes.

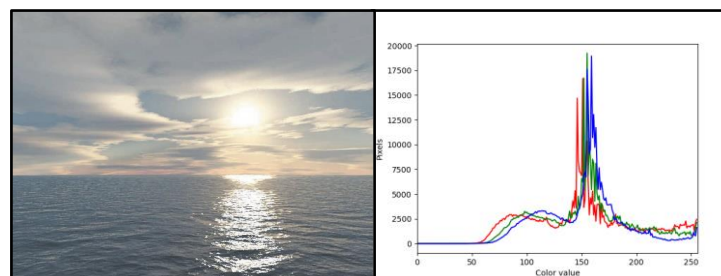
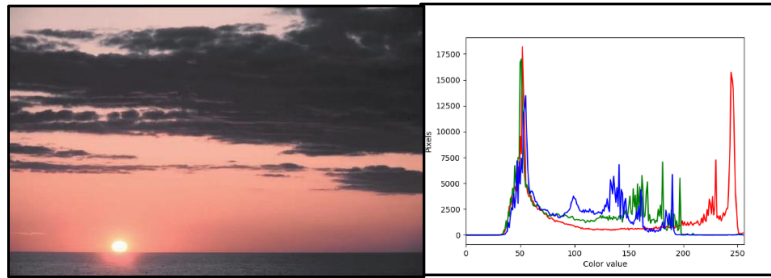
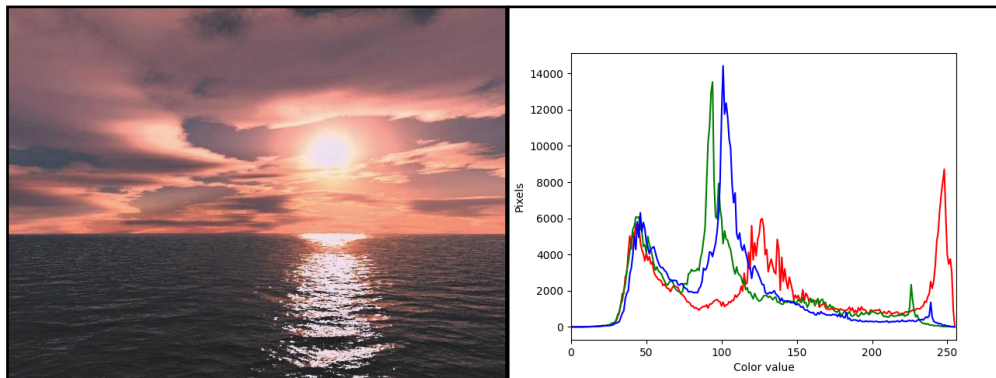


Image initiale : coucher de soleil en 3D



Histogramme source (vraie photo)

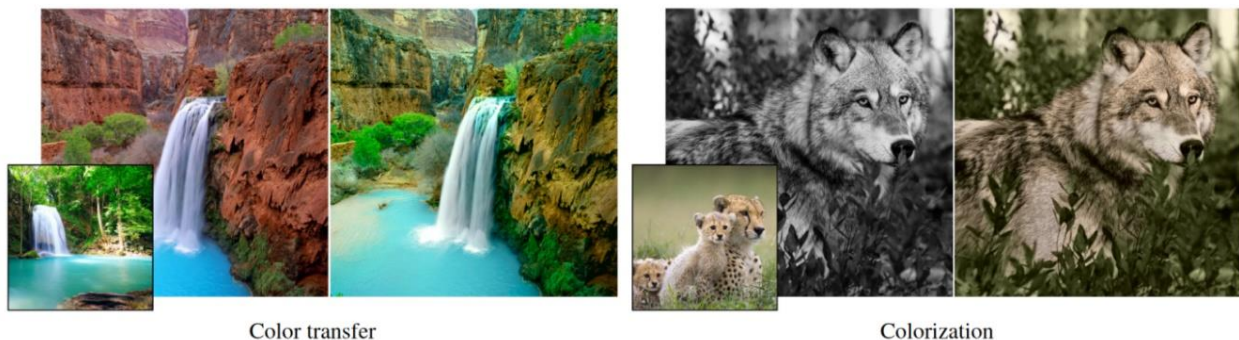


Coucher de soleil réaliste en 3D

©q5rein.lo (erikreinhard.com)

Cette technique permet de modifier de façon plus organique l'ambiance souhaitée. Exemple : une scène de coucher de soleil générée par ordinateur dont on rend la lumière beaucoup plus réaliste à l'aide de l'ambiance d'une vraie photo.

L'équipe de recherche MAVERICK essaye de développer cette technique en déterminant plus précisément les cibles du transfert de couleur. Plutôt que d'appliquer une transformation d'histogramme uniforme à l'image, on peut essayer de reconnaître des zones (poils, nuages, etc) similaires entre les deux images afin d'opérer un transfert plus localisé et plus précis. A suivre donc...



© Benoit Arbelot, Romain Vergne, Thomas Hurtut, Joëlle Thollot. Automatic Texture Guided Color Transfer and Colorization. Expressive 2016, May 2016, Lisbonne, Portugal. pp.21-32, ff10.2312/exp.20161060ff. fffhal-01305596

2.4. CHANGEMENT DE FORMES ET CARICATURES

L'œil humain regarde mais ne voit pas

Contrairement à une caméra numérique, notre vision humaine ne nous permet pas de voir le monde comme un assemblage de pixels. Nous percevons le monde qui nous entoure par nos yeux mais notre compréhension de ces images découle aussi de nos expériences passées et de notre connaissance du monde visible.

Voilà pourquoi produire informatiquement des images qui nous semblent réalistes est si difficile : notre perception visuelle est bien plus complexe qu'un simple panneau de couleur en deux dimensions.



© Romain Vergne, Pascal Barla, Georges-Pierre Bonneau, Roland Fleming. *Flow-Guided Warping for Image-Based Shape Manipulation*. *ACM Transactions on Graphics, Association for Computing Machinery*, 2016, 34 (4), pp.Article No. 93. [ff10.1145/2897824.2925937ff](https://doi.org/10.1145/2897824.2925937ff). [ffhal-01307571f](https://arxiv.org/abs/1601.01307)

Ainsi nous ne percevons pas l'image de ce lion comme un tableau de valeurs Rouges, Vertes et Bleues. Nous savons immédiatement que les surfaces blanches ne sont pas une couleur, mais un reflet lumineux. Notre expérience nous apprend instantanément que c'est une image en relief, sculptée, et non pas peinte, dans une matière qui réfléchit la lumière.

Une supposition scientifique peut être établie : la forme de ces surfaces blanches joue un rôle dans notre compréhension des images. En les modifiant par le biais d'un algorithme spécifique, il est possible de modifier notre compréhension de l'image.

2.4.1. Modifier l'impression de relief

Les chercheurs de l'équipe MAVERICK ont donc conçu un algorithme en deux étapes.

Tout d'abord, en utilisant des données statistiques, l'algorithme détecte ces formes blanches que notre œil reconnaît comme issues d'une image en relief. Cette forme, comme d'autres indices visuels, est un marqueur de relief.

Ensuite, l'algorithme peut modifier ces marqueurs pour les rendre plus larges ou au contraire plus étroits. La modification de ces marqueurs modifie notre perception de l'image et nous fait croire que son relief est plus marqué ou au contraire plus doux.

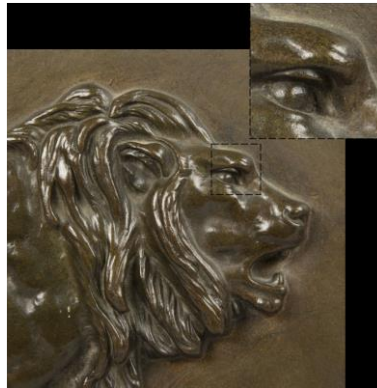
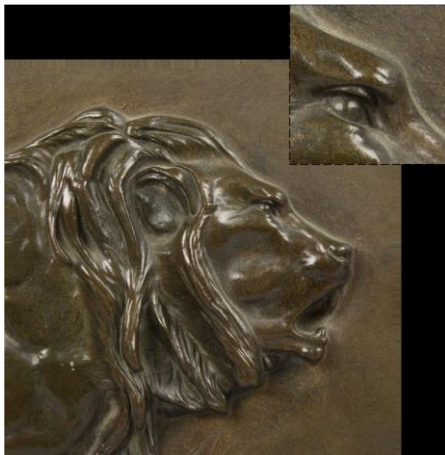


Image originale



Marqueurs rétrécis

On remarque que les poils de la crinière semblent plus fins et la joue beaucoup plus large car relativement plane

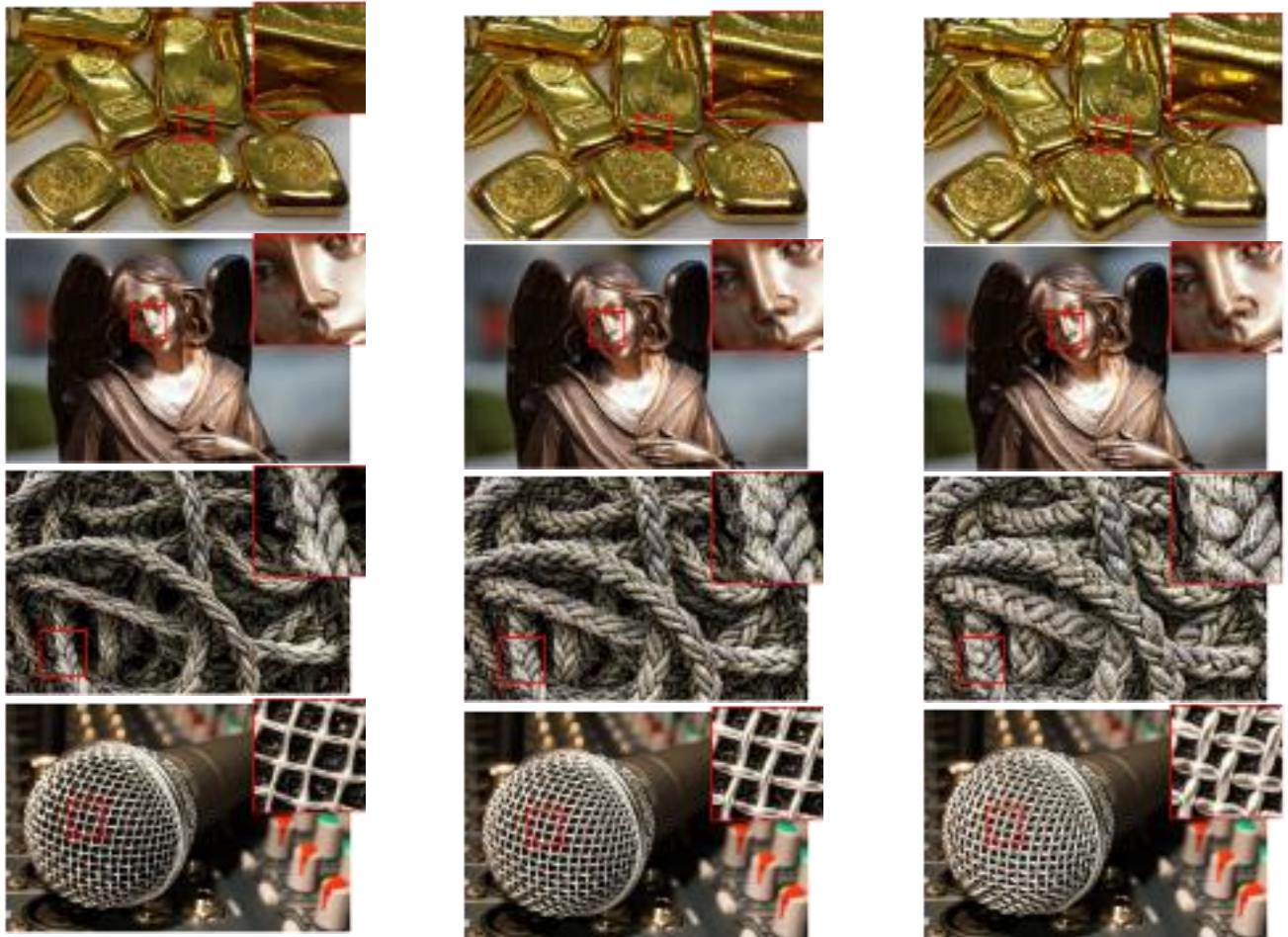


Marqueurs élargis

Ici, la crinière prend presque toute la place, la joue disparaît quasiment sous la masse de poils qui sont en relief

© Romain Vergne, Pascal Barla, Georges-Pierre Bonneau, Roland Fleming. Flow-Guided Warping for Image-Based Shape Manipulation. ACM Transactions on Graphics, Association for Computing Machinery, 2016, 34 (4), pp.Article No. 93. ff10.1145/2897824.2925937ff. fhal-01307571f

Mais l'algorithme ne s'intéresse pas qu'aux surfaces réfléchissantes. Il détecte des formes particulières qui sont très souvent le signe d'une surface en relief. On peut donc appliquer l'algorithme à de nombreuses images et étudier la façon dont nous comprenons la modification de ces formes :



Marqueurs rétrécis

Image originale

Marqueurs agrandis

© Romain Vergne, Pascal Barla, Georges-Pierre Bonneau, Roland Fleming. *Flow-Guided Warping for Image-Based Shape Manipulation*. *ACM Transactions on Graphics, Association for Computing Machinery*, 2016, 34 (4), pp.Article No. 93.

Nous avons un algorithme qui détecte les reliefs sur une image à partir de certaines caractéristiques de celle-ci. Il est ensuite capable d'accentuer ou de réduire ces reliefs, de les rendre plus pointus ou de les arrondir. C'est comme cela que nous en arrivons à Photomaton et aux caricatures.

En effet, une caricature est l'accentuation des traits principaux du visage, donc du relief de celui-ci. C'est pourquoi, quand l'algorithme est associé à un visage, il produira des caricatures. En fait, il détecte les lignes principales de nos traits, et les modifie, pour en faire des images qui nous paraissent caricaturales.



© Romain Vergne, Pascal Barla, Georges-Pierre Bonneau, Roland Fleming. *Flow-Guided Warping for Image-Based Shape Manipulation*. *ACM Transactions on Graphics, Association for Computing Machinery*, 2016, 34 (4), pp.Article No. 93. 01307571f

3. Conclusion

Les trois possibilités du démonstrateur Photomaton reflètent trois approches différentes de la manipulation d'image : modification de style graphique, modification de couleur ou modification de forme.

Le traitement d'image est donc avant tout un travail d'illusionnistes ; on utilise les biais de notre cerveau pour donner une impression, faire ressentir une émotion ou faire croire à un style particulier.

Malgré des problématiques et des techniques différentes, on distingue de nombreux points communs à ces approches, notamment le fait qu'il ne s'agit pas que d'algorithmes et de calculs mathématiques. Notre perception des images et les mécanismes qui la régissent sont également au cœur de ces recherches.

Les applications de ces recherches vont bien plus loin qu'un simple attrait esthétique, tant les images sont aujourd'hui présentes dans nos vies. Comprendre comment nous regardons une image nous permet d'en faire un outil plus précis pour représenter plus d'informations. On en fait des cartes plus utiles, des plans plus complets, des représentations de données abstraites (température, consommation électrique, etc.) plus efficaces. On trouve donc des applications concrètes de ces recherches en cartographie, en archéologie, en architectures, plus généralement dans tous les domaines où l'image est un véritable vecteur d'information.

4. Sources et liens utiles

4.1.1. Mieux comprendre le pixel

- Page Wikipédia de Pixel : [Pixel - Wikipedia](#)
- Page Wikipédia : [Color histogram - Wikipedia](#)

4.1.2. Un peu de thèses...

- Article Automatic Texture Guided Color Transfer and Colorization de Benoit Arbelot, Romain Vergne, Thomas Hurtut, Joëlle Thollot: [Automatic Texture Guided Color Transfer and Colorization \(archives-ouvertes.fr\)](#)
- Article Color transfer between images de Erik Reinhard, Michael Ashikhmin, Bruce Gooch, and Peter Shirley, University of Utah: [g5rein.lo \(erikreinhard.com\)](#)
- Article Flow-Guided Warping for Image-Based Shape Manipulation de Romain Vergne, Pascal Barla, Georges-Pierre Bonneau, Roland Fleming : [Flow-Guided Warping for Image-Based Shape Manipulation \(inria.fr\)](#)
- Water color tutorial d'Adrien Bousseau : [Microsoft Word - watercolor_tutorial.doc \(inria.fr\)](#)

4.1.3. Un peu de programmation...

- Cours gratuit sur le site Openclassrooms : [Comprenez comment un ordinateur voit une image - Classez et segmentez des données visuelles - OpenClassrooms](#)
- [Creating Histograms – Image Processing with Python \(datacarpentry.org\)](#)
- [GitHub - chia56028/Color-Transfer-between-Images: This is the open-source implement the paper "Color Transfer between Images" by Erik Reinhard, Michael Ashikhmin, Bruce Gooch and Peter Shirley.](#)