

# TD N°2 – Réglages colorimétriques dans l'espace HSI.

Stéfane Paris and Gilles Simon

*{Stefane.Paris,Gilles.Simon}@loria.fr*

## Qu'est-ce que la couleur ?

La lumière est un signal électro-magnétique que nous traduisons en terme de couleurs grâce à nos yeux. Cette perception de la couleur étant fortement subjective, il existe plusieurs espaces couleurs ayant une modélisation différente de la couleur et des utilisations différentes. Nous proposons ici d'explorer plus particulièrement l'espace HSI (teinte (hue), saturation, intensité), très utilisé dans les logiciels de retouche d'image tels que Photoshop ou Gimp, car il permet de réaliser facilement certains réglages colorimétriques difficiles, voire impossible, à obtenir dans l'espace RGB.

## Les espaces *sRGB*, *uRGB* et *HSI*

Nous considérons dans ce TD les trois espaces de couleur suivants :

L'espace *sRGB* qui est le plus connu puisqu'utilisé par la majorité des capteurs numériques ; il s'agit en fait d'un standard qui s'est imposé (car souvent le seul proposé) construit conjointement par HP et Microsoft et qui a été approuvé par plusieurs instances internationales dont W3C, Exif, Intel, Pantone, Corel.

L'espace *unitaire uRGB* qui amène les valeurs de chaque composante de l'espace sRGB entre 0 et 1, en les divisant simplement par 255 ;

L'espace *HSI* qui transforme l'espace uRGB en teinte (hue) saturation et intensité :

$$\left\{ \begin{array}{lcl} H & = & \text{atan2}(y, x) (\in [0, 2\pi]) \\ S & = & \sqrt{x^2 + y^2} \\ I & = & \frac{uR + uG + uB}{3} \end{array} \right.$$

$$x = uR - \frac{1}{2}(uG + uB) ; y = \frac{\sqrt{3}}{2}(uG - uB)$$

La transformation inverse,  $HSI \rightarrow uRGB$ , est obtenue à l'aide des formules suivantes :

$$\left\{ \begin{array}{lcl} uR & = & I + \frac{2}{3}x \\ uG & = & I - \frac{1}{3}x + \frac{1}{\sqrt{3}}y \\ uB & = & I - \frac{1}{3}x - \frac{1}{\sqrt{3}}y \end{array} \right.$$

$$x = S \cos(H) ; y = S \sin(H)$$

# Conversion uRGB $\leftrightarrow$ HSI

**Exercice 1** Lisez l'image `shadows.jpg` et visualisez-la :

```
sRGB = imread('shadows.jpg'); figure; imshow(sRGB); drawnow();
```

La fonction `imread` retourne un tableau de taille  $M \times N$  dans le cas d'une image à niveaux de gris, ou un tableau de taille  $M \times N \times 3$  dans le cas d'une image couleur. Les coefficients du tableau sont par défaut exprimés dans l'espace sRGB, écrivez la fonction

```
uRGB = sRGB2uRGB(sRGB)
```

qui convertit une image exprimée dans l'espace sRGB en une image exprimée dans l'espace uRGB (dans la suite, nous dirons simplement "une image uRGB"). Appelez cette fonction avec l'image sRGB et affichez le résultat (`figure; imshow(uRGB); drawnow();`). À noter que la fonction `imshow` se débrouille aussi bien avec des coefficients entre 0 et 255 qu'avec des coefficients entre 0 et 1.

Écrivez la fonction

```
HSI = uRGB2HSI(uRGB)
```

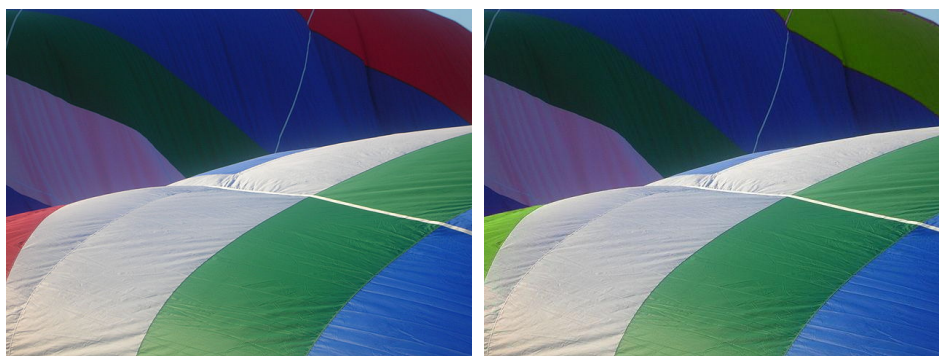
qui convertit une image uRGB en une image HSI. Appelez cette fonction avec l'image uRGB obtenue précédemment. Écrivez une fonction `compshow(im)` qui affiche séparément, en niveaux de gris, les trois composantes de l'image `im` exprimée dans un espace colorimétrique quelconque (utilisez `subplot` de manière à ce que les trois composantes soient affichées dans une même figure, sur la même ligne). Attention : les valeurs des composantes d'un espace colorimétrique quelconque ne sont pas nécessairement comprises entre 0 et 1, par exemple la valeur H de l'espace HSI est comprise entre 0 et  $2\pi$ , il faut donc utiliser le paramètre '`DisplayRange`' de la fonction `imshow` pour que toutes les valeurs possibles des différentes composantes soient représentées correctement. Affichez les trois composantes de l'image HSI obtenue, comment interprétez-vous l'apparence des différentes couches ?

Écrivez la fonction

```
uRGB = HSI2uRGB(HSI)
```

qui convertit une image HSI en une image uRGB, appelez cette fonction avec l'image HSI obtenue et vérifiez, en l'affichant, que le résultat correspond bien à l'image originale.

## Recolorisation d'une photographie



(a) Image originale.

(b) Image recolorisée.

Figure 1: Recolorisation dans l'espace HSI

**Exercice 2** Nous allons remplacer la couleur rouge du parachute par une couleur jaune-vert, sans toucher aux autres couleurs de la photographie. Pour cela :

- récupérez la teinte  $h$  de la couleur sous le pixel de coordonnées (600,100),

- remplacez par  $\pi/2$  la teinte de tous les pixels de l'image dont la teinte est à une distance inférieure à  $\pi/5$  de  $h$  (utilisez la fonction `find()` de Matlab).

Affichez le résultat, que constatez-vous ? Qu'avons-nous conservé de l'image originale dans la zone retouchée ? Aurions-nous pu obtenir un tel résultat en travaillant dans l'espace RGB ?

Remplacez les coordonnées "en dur" (600,100) par des coordonnées  $(x,y)$  obtenues en cliquant sur l'image :

» `figure; imshow(sRGB); drawnow();`

» `[x,y] = ginput(1);`

Le résultat doit apparaître immédiatement après le click, à la place de l'image originale.

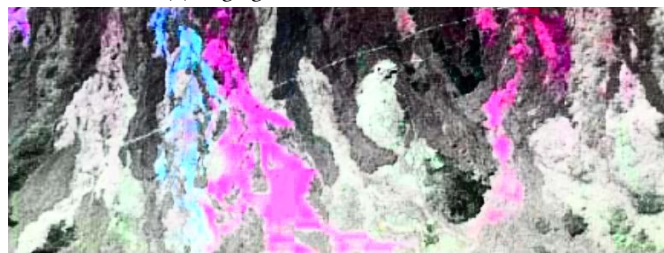
## Rehaussement des contrastes



(a) Image originale.



(b) Réglage des contrastes, couche I.



(c) Réglage des contrastes, couches I + S.

Figure 2: Rehaussement des contrastes dans l'espace HSI

Nous allons à présent rehausser les contrastes d'une photographie scientifique représentant une coulée de lave. Comme beaucoup de réglages colorimétriques, un réglage de contraste repose sur une courbe tonale, qui indique par quelles valeurs (entre 0 et 1) indiquées en ordonnée remplacer les valeurs (ramenées entre 0 et 1 et indiquées en abscisse) d'une composante donnée. Dans notre cas, nous utiliserons la courbe tonale suivante, basée sur une sigmoïde:

```
function y = tonecurve(x)
y = logsig(5*(2*x-1));
end
```

**Exercice 3** Tracez `tonecurve(X)` pour  $X=[0:0.01:1]$ . Quel effet aura l'utilisation de cette courbe tonale sur les intensités lumineuses d'une image ?

**Exercice 4** Écrivez une fonction `Y=normalise(X)` qui transforme linéairement les coefficients de la matrice  $X$  de manière à les ramener entre 0 et 1 (la valeur minimale de  $X$  devient 0 et la valeur maximale devient 1). Appliquez la courbe tonale sur la couche I normalisée de l'image `monaloo.jpg` convertie en HSI. Vérifiez que le résultat obtenu correspond à votre attente.

**Exercice 5** Appliquez la courbe tonale sur la couche **S normalisée** de l'image obtenue dans l'exercice 4. Quel est l'intérêt de ce réglage ?