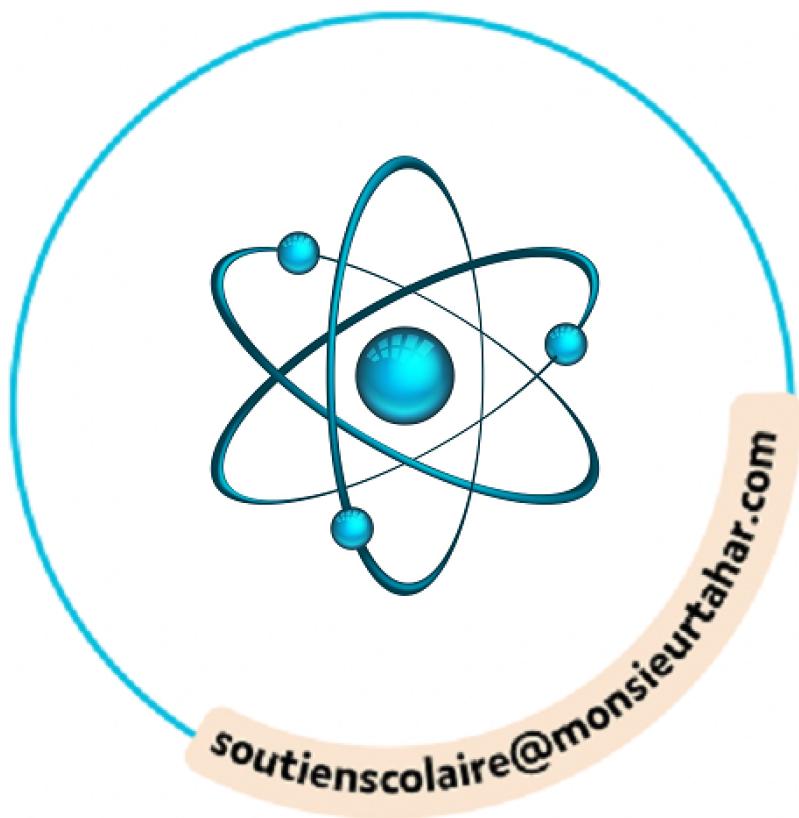


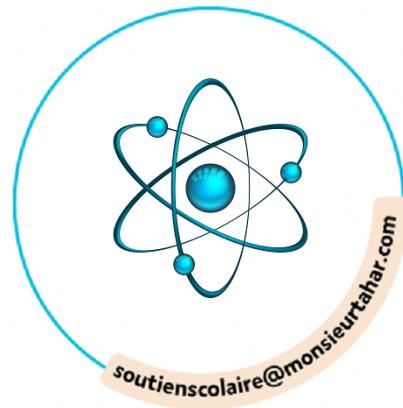
# SNT



## CHAPITRE 7

# Exercices

## Autoévaluation



### 1. Résolution et définition

Lorsque l'on multiplie par 2 la résolution d'une photographie, sa définition est multipliée par :

Lorsque l'on multiplie par 2 la résolution d'une photographie, sa définition est multipliée par 4. En effet, on augmente par 2 le nombre de pixels selon la hauteur et selon la largeur. Cela se traduit donc par une augmentation totale du nombre de pixels par  $2^2 = 4$ .

### 2. Codage des couleurs

#### 1. Rappeler les codes RVB du cyan, du jaune et du magenta.

Les codes RVB sont les suivants :

- code RVB du cyan : (0, 255, 255) ;
- code RVB du jaune : (255, 255, 0) ;
- code RVB du magenta : (255, 0, 255).

#### 2. En se rendant sur [LLS.fr/SNT2EXRVB](http://LLS.fr/SNT2EXRVB), trouver la couleur dont le code RVB est (255, 96, 208).

On trouve que la couleur codée par (R,V,B) = (255,96,208) est le rose.

### 3. Taille d'une image

On considère une image ayant une résolution de 64 ppp et une définition de 4 096 × 4 096 pixels. Déterminer sa taille.

On sait qu'il y a 64 pixels par pouce et que la largeur et la hauteur valent 4 096 pixels.

$$r = \frac{n_l}{l}$$

$$l = \frac{n_l}{r}$$

$$AN : l = \frac{4096}{64} = 64, = 160 \text{ cm.}$$

### 4. Conservation des proportions

#### 1. Déterminer la hauteur de l'image finale dans l'espace disponible si les proportions sont respectées.

L'image étant en format paysage avec une largeur initiale  $l$  de 22 cm et finale  $l'$  de 5 cm, on a donc le coefficient de réduction suivant :

$$k = \frac{l}{l'} \\ \text{AN : } k = \frac{22}{5} = 4,4$$

Par conséquent, pour trouver la hauteur  $h'$  correspondante, on a donc :

$$k = \frac{h}{h'} \\ h' = \frac{h}{k} \\ \text{AN : } h' = \frac{11}{4,4} = 2,5 \text{ cm.}$$

## 2. Sachant que le nombre de pixels est conservé, en déduire de combien augmente ou diminue la résolution.

Le nombre de pixels étant conservé, noté par exemple  $n_h$  pour le nombre de pixels selon la hauteur, on a :

$$r = \frac{n_h}{h} \\ r' = \frac{n_h}{h'}$$

En combinant les deux relations, il vient :

$$h' \times r' = h \times r \\ r' = \frac{h}{h'} \times r \\ \text{AN : } r' = 4,4r$$

La résolution a été multipliée par 4,4.

## 5. Pixel

Combien de photosites au minimum sont nécessaires pour coder correctement la couleur d'un pixel ?

Pour coder correctement la couleur d'un pixel, il ne suffit que de trois photosites. Néanmoins, par des soucis de praticité en termes de maillage des filtres sur la matrice, il est courant d'avoir quatre photosites utilisés pour coder les trois composantes de couleur. Comme le vert est la couleur pour laquelle l'œil humain est le plus sensible, doubler le nombre de photosites sensibles à cette couleur permet d'améliorer le rendu global.

## 6. Vrai/Faux

### 1. Tous les appareils photographiques fonctionnent de la même manière.

Faux. Les appareils photographiques argentiques diffèrent des appareils photographiques numériques, notamment par la numérisation de données, en encore la quantification à l'échelle des photosites du rayonnement lumineux mesuré.

## **2. Il existe plusieurs moyens de stabiliser une photographie avec un appareil numérique.**

Vrai. On peut citer par exemple la stabilisation mécanique qui consiste à atténuer les mouvements brusques du capteur photographique lors de la capture, ou encore une multiplication de la prise de photographies qui permet, après sélection, de ne conserver que les photographies qui ne sont pas floues.

## **7. Frise chronologique**

**Classer par ordre chronologique les éléments suivants.**

- 1 : Invention de la photographie argentique (1826)
- 3 : Invention de la photographie numérique (1975)
- 4 : Lancement du logiciel Photoshop (1990)
- 2 : Première photo en couleurs (1861)

## **8. Définitions**

**Relier chaque valeur à la grandeur qui lui est associée.**

- 5" x 5" • • Taille  
10 ppp • • Résolution  
50 x 50 pixels • • Définition  
12,5 x 12,5 cm • • Taille

## **9. Qui suis-je ?**

### **1. Je suis une grandeur caractérisant le nombre de pixels par unité de longueur dans une image.**

Je suis la résolution.

### **2. Je suis un ensemble de données caractérisant une photographie, et donnant des informations sur l'appareil qui l'a prise.**

Je suis les métadonnées EXIF.

# Application

## 10. Données

### 1. Donner le nom des données contenues dans le fichier et ne servant pas à coder l'image.

Les données autres que celles liées à la position et à la couleur des pixels sont les métadonnées EXIF.

### 2. Donner deux exemples d'informations pouvant être apportées par ces données.

Elles indiquent un ensemble de caractéristiques liées à l'image : résolution, définition, codage des couleurs, lieu et date de prise de vue, etc.

## 11. HDR

### ◆ Donner un autre exemple de méthode d'amélioration de la qualité d'une photographie.

Avec des systèmes de stabilisation numérique pour compenser des petits mouvements (vibrations des mains, etc.) ou jusqu'au HDR qui permet de d'obtenir une image finale avec plus de détails grâce à la capture de plusieurs images au même moment (avec une tonalité de lumière, une ombre et des durées d'exposition différentes), les algorithmes permettent d'obtenir des rendus de meilleure qualité.

On peut également citer plusieurs algorithmes comme le *focus-peaking* (scintillement des contours nets), le focus stacking pour étendre la netteté avec plusieurs mises au point successives, la balance des blancs, la réduction du bruit et l'amélioration de la netteté, etc.

## 12. Mont-Blanc vu en grand

### ◆ Calculer, pour un codage RVB 24 bits (3 octets par pixel), la mémoire occupée par l'image. Exprimer le résultat en téraoctets (To) (1 To = $10^{12}$ o).

Comme il y a 365 milliards de pixels et que chaque pixel a besoin de 24 bits, on a donc  $365 \times 10^9 \times 24 = 8,76 \times 10^{12}$  bits.

Mais, comme 24 bits correspondent à 3 o, on a donc

$$\frac{8,76 \times 10^{12} \times 3}{24} = 1,10 \times 10^{12} \text{ o} = 1,10 \text{ To.}$$

## 13. Stabilisation d'image

### ◆ Expliquer brièvement le principe de la stabilisation mécanique d'un appareil photographique.

Lors d'une prise de vue, certains mouvements intempestifs de l'appareil et du photographe peuvent provoquer une déformation de la photographie et générer un flou global.

Les ingénieurs ont mis au point un système de réponse mécanique à ces mouvements.

À partir de capteurs, les mouvements de l'appareil sont détectés ; la réponse du système consiste à déplacer légèrement le capteur photosensible de façon à conserver une prise de vue stable. Ce déplacement est réalisé grâce à des aimants contrôlés par un algorithme qui tient compte des mesures fournies par les détecteurs.

## 14. Métadonnées EXIF

### 1. Dans les métadonnées ci-dessous, relever la méthode de codage des couleurs.

Il s'agit du codage RVB (ligne suivante : *Photometric interpretation {0x0106} = RGB (2)*).

### 2. À partir de la résolution et de la définition données, calculer la taille de l'image en pouces.

Les métadonnées EXIF indiquent une image avec une définition de 2 136 x 3 216 pixels et une résolution de 300 ppp. On a donc :

$$r = \frac{n_h}{h}$$

$$h = \frac{n_h}{r}$$

$$\text{AN : } h = \frac{2136}{300} = 7,12, \text{''} = 18,1 \text{ cm}$$

$$r = \frac{n_l}{l}$$

$$l = \frac{n_l}{r}$$

$$\text{AN : } l = \frac{3216}{300} = 10,7, \text{''} = 27,2 \text{ cm}$$

## 15. Vrai/Faux

### 1. Toutes les images utilisent le codage RVB pour les couleurs.

Faux. On peut citer d'autres codages comme le codage TSV par exemple.

### 2. La résolution d'une image est généralement exprimée en pixels par pied (ppp).

Faux. C'est en pixels par pouce.

## 16. Manipulation

### 1. Il est possible de modifier les métadonnées EXIF d'une image.

Vrai. Les métadonnées EXIF sont facilement modifiables à partir d'un logiciel de traitement d'images, comme Gimp par exemple.

### 2. Les métadonnées EXIF sont un moyen fiable de connaître l'origine d'une image.

Faux. Les métadonnées EXIF peuvent être modifiées, elles ne peuvent donc pas être une source d'information fiable pour trouver l'origine d'une photographie.

## 17. Retouche d'image

### ◆ Les logiciels de retouche d'image :

Les logiciels de retouche d'image permettent de modifier complètement une image et de lui faire porter un message totalement différent du message initial.

## 18. Traitement d'images

### ◆ Compléter le code Python ci-dessous pour obtenir l'algorithme de transformation d'une image en niveau de gris.

```
from PIL import Image
import numpy as np
import matplotlib.pyplot as plt

im = np.array(Image.open("photo.jpg"))

rouge = im[:, :, 0]
vert = im[:, :, 1]
bleu = im[:, :, 2]

gris = round((rouge+vert+bleu)/3)

im[:, :, 0] = gris
im[:, :, 1] = gris
im[:, :, 2] = gris

plt.imshow(im)
plt.show()
```

Remarque : la fonction `round()` renvoie le nombre entier le plus proche du nombre spécifié, par exemple `round(5.3)` donne 5.

## 19. Vrai/Faux

### 1. Les métadonnées EXIF peuvent indiquer qui a pris la photographie.

Vrai. Lorsque les appareils photographiques numériques ont le nom de leur propriétaire enregistré, ils peuvent l'associer aux métadonnées EXIF des photographies prises.

### 2. Les métadonnées EXIF peuvent indiquer où a été prise la photographie.

Vrai. Les coordonnées GPS font partie des métadonnées EXIF éditées, à condition que l'appareil photographique ait la possibilité de se repérer (les *smartphones* par exemple, lorsque la localisation est activée).

## 20. Deepfakes

### ◆ ➡ À l'aide d'Internet, expliquer brièvement le danger créé par les *deep fakes*.

En décembre 2017, sur la plateforme *Reddit*, un utilisateur du pseudonyme « *Deepfakes* » a publié des vidéos de célébrités qui se sont révélées fausses par la suite. Il a généré ces

fausses images en utilisant un algorithme de dernière génération en matière d'intelligence artificielle : l'apprentissage en profondeur ou *deep learning*.

Les dérives du *deep fake* sont nombreuses. Comme expliqué plus haut, des vidéos de célébrités se sont révélées fausses par la suite, mais comme le montre l'image ci-dessous (un *deep fake* de l'ancien président Obama comparé à une vraie photographie), par exemple, il est extrêmement difficile de trouver l'image originale.



## 21. Matrice de Bayer

### 1. Donner le nom des éléments du capteur représentés par un carré gris.

Les éléments en gris sont les photosites.

### 2. Préciser combien d'éléments sont nécessaires au minimum pour coder correctement un pixel.

Il faut au minimum 3 photosites pour coder correctement un pixel (un photosite par couleur) ; néanmoins, pour pallier la forte sensibilité de l'oeil humain à la couleur verte, et pour tenir compte de la géométrie des capteurs, on utilise plutôt quatre photosites pour coder correctement un pixel (un pour le bleu, un pour le rouge et deux pour le vert).

## 22. Codage des couleurs

### ◆ Relier les codes RVB suivants à la couleur à laquelle ils correspondent.

- (127,0,255) • • Violet
- (255,255,0) • • Jaune
- (253,108,158) • • Rose
- (75,75,75) • • Gris

## Pratique

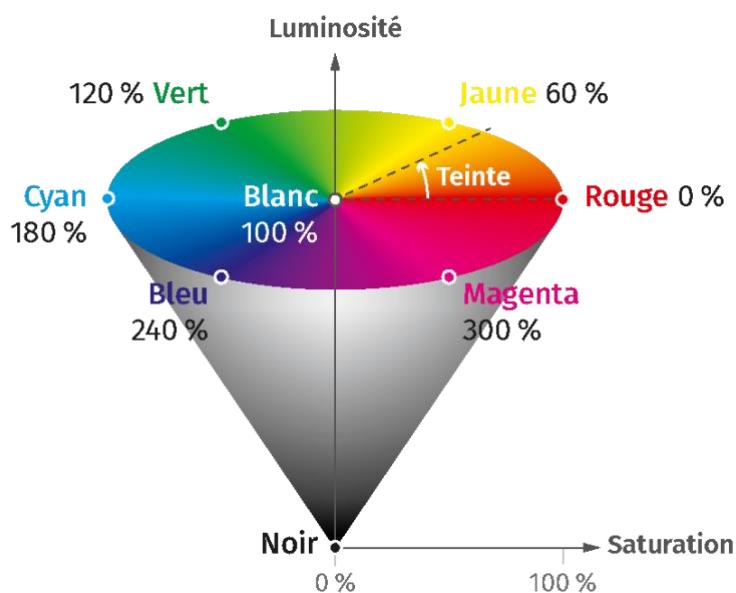
### 23. Algorithme de traitement des images

Correction :

#### 1. Identifier le type de codage des couleurs utilisé dans l'algorithme.

Il s'agit du codage TSV (en anglais *HSV*) où les pixels sont décomposés en 3 composantes, la Teinte, la Saturation et la Valeur (en anglais : *Hue, Saturation, Value*) :

- la teinte est décrite par un angle qui correspond à la position de la couleur sur le cercle chromatique : 0° rouge, 60° jaune, 120° vert, 180° cyan, 240° bleu, 300° magenta, 360° rouge à nouveau ;
- la saturation décrit la quantité de gris dans une couleur particulière, de 0 % à 100 % ;
- la valeur (ou luminosité) fonctionne en conjonction avec la saturation et décrit la luminosité ou l'intensité de la couleur, de 0 % (complètement noir) à 100 % (le plus brillant).



#### 2. Identifier la valeur modifiée par cet algorithme.

On modifie la valeur  $h$  de chaque pixel en lui ajoutant 180°. Tout se passe dans la ligne de code 9 : `new2.putpixel((i, j), (h+180, s, v))`. On ajoute une valeur de 180° à toutes les valeurs de teinte des pixels.

## Parcours pix

### Correction :

#### **Niveau 1. Quels sont les codages RVB des couleurs rouge, vert et bleu ?**

Pour les couleurs rouge, vert et bleu, il faut utiliser les codes suivants :

- rouge (255, 0, 0) ;
- vert (0, 255, 0) ;
- bleu (0, 0, 255).

#### **Niveau 2. Quel est le code couleur RVB du noir ? Du blanc ? D'un gris ?**

Le code RVB du noir est (0, 0, 0) et celui du blanc est (255, 255, 255). Pour un gris, il suffit que les trois composantes aient la même valeur, par exemple (100, 100, 100).

#### **Niveau 3. Un développeur Web souhaite utiliser la couleur disponible sur [LLS.fr/SNT2EXP92](http://LLS.fr/SNT2EXP92) pour le fond de la page d'accueil d'un site. Quel code couleur devra-t-il renseigner dans le fichier CSS de la page Web ?**

En accédant au lien, on peut déterminer la couleur de l'encart, à savoir celle d'un rose-violet (227, 0, 123). Pour l'utiliser dans le cadre d'une mise en forme d'une page Web, on peut renseigner le code suivant à l'élément que l'on souhaite colorisé : **rgb(227, 0, 123)**.