

## Chapitre 7 - La photographie numérique

### I - Introduction : un peu d'histoire !

#### Activité 1

Réaliser, à l'aide de l'outil situé à l'adresse [http ://www.frisechronos.fr/dojomain.htm](http://www.frisechronos.fr/dojomain.htm) , une frise chronologique comportant les dates clés suivantes sur la photographie numérique.

- 1826 - naissance de la photographie argentique
- 1900 - photographie en couleurs
- 1950 - généralisation du format 24x36 et de la visée reflex
- 1969 - premier capteur CCD
- 1975 - premier appareil numérique
- 1990 - premier logiciel de retouche photo (Photoshop 1.0)
- 2007 - apparition du smartphone et de son appareil photo intégré

Tous les événements situés sur la frise seront illustrés par une image et un commentaire.

#### Activité 2

Associer chacun des inventeurs suivants à son invention en lien avec la photographie.

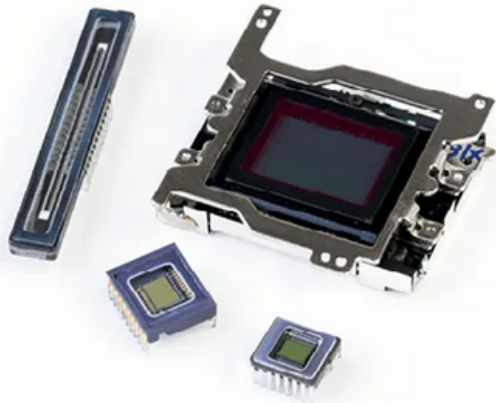
Inventeurs	Inventions
Louis Daguerre ●	● La couleur trichromique
William Henry Fox Talbot ●	● Premier appareil photo numérique
Louis Ducot du Hauron ●	● L'autochrome
George Eastman ●	● L'héliographie
Steven Sasson ●	● Le daguerréotype
Nicéphore Niepce ●	● Le calotype (négatif papier)
Louis Lumière ●	● Premier appareil photo portable Kodak

## II - La capture des images numériques

### Le principe de la capture

La structure optique d'un appareil photographique est la même pour tous les appareils. L'objectif est composé d'un ensemble de **lentilles focalisant la lumière** afin de former une image au niveau de l'élément de capture.

Dans un appareil numérique, l'élément de capture est un capteur photographique qui **convertit le signal lumineux en signal électrique**.



#### Capteurs CCD et CMOS

*Successeurs des capteurs CCD (Charge Coupled Device), les capteurs CMOS (Complementary metal oxide semi-conductor) sont aujourd'hui les capteurs photosensibles les plus présents dans les appareils photographiques numériques.*

### La structure d'un capteur

Les capteurs ne permettent pas d'effectuer à eux seuls une différenciation entre les couleurs. Afin d'obtenir une information concernant les couleurs, un filtre coloré est superposé au capteur.

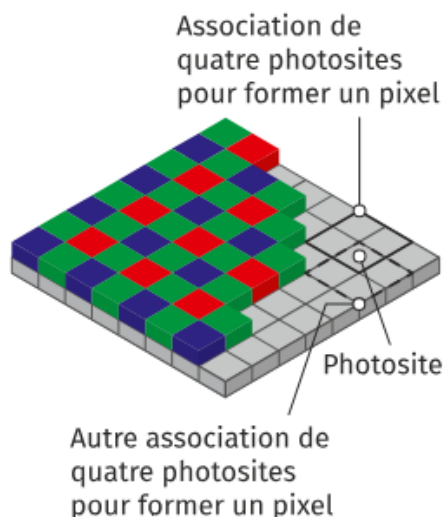
Chaque élément unitaire du capteur, appelé **photosite**, recueille alors une information concernant une couleur : **le rouge, le vert ou le bleu**.

#### Définition 1

Un **photosite** est une cellule de capteur photographique sensible à une partie spécifique du rayonnement lumineux, émettant en réponse un signal électrique.

En d'autres termes : Un photosite est un composant microscopique d'un capteur **sensible à la couleur de la lumière reçue**.

Dans notre oeil, les cônes sont des cellules qui jouent le même rôle que les photosites.



#### Matrice de Bayer

Maillage de photosites spécifiquement sensibles aux trois couleurs (rouge, vert et bleu).

### III - Les images numériques

#### Définition 2

Un **pixel** est un composant élémentaire d'une image numérique associé à des coordonnées et à des composantes de couleur.

En d'autres termes : C'est le **plus petit élément coloré d'une image**.

En peinture, les pixels seraient comparables aux touches de couleurs des peintillistes.

#### Définition d'une image

#### Définition 3

Les images numériques ne sont pas continues, elles se décomposent en **pixels**.

La **définition** indique le nombre de pixels d'une image.

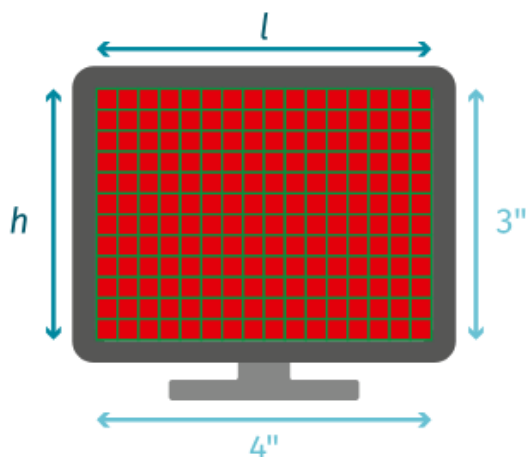
La définition est souvent présentée sous la forme  $l \times h$  où  $l$  est le nombre de pixels dans le sens de la largeur et  $h$  le nombre de pixels dans le sens de la hauteur.

#### Résolution

#### Définition 4

La **résolution** d'une image correspond au nombre de pixels par unité de longueur d'une image, le plus souvent exprimé en **pixel par pouce (ppp) (dpi pour dots per inch en anglais)**.

La résolution est importante lorsque l'on regarde l'image sur un écran ou lorsque l'on effectue une impression, c'est elle qui définit **la qualité de l'image**.



**Définition :**  $l \times h = 16 \times 12$  pixels

**Taille :** 4 "  $\times$  3 "

**Résolution :** 4 ppp



Crédits : PPstock/Shutterstock

Différentes résolutions pour une même photographie

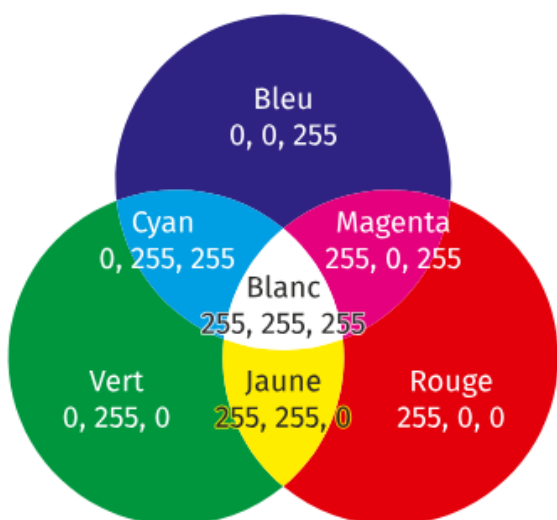
## Le codage des couleurs

Chaque pixel d'une image numérique contient l'information sur sa couleur. Le codage le plus répandu est le codage **RVB** (ou **RGB** en anglais).

La couleur d'un pixel est alors codé par trois nombres représentant le rouge, le vert et le bleu. On appelle profondeur de couleur le nombre de bits utilisés pour coder les couleurs des pixels.

**Exemple** : Codage RGB avec une profondeur de couleur de 8 : La couleur d'un pixel est définie par 3 nombres de 8 bits.

- Chaque nombre définit une quantité de rouge, de vert ou de bleu.
- La quantité d'une couleur varie de 0 à 255 (valeur maximale que l'on peut coder sur 8 bits).
- la couleur RGB (255, 255, 255) indique une quantité de rouge de 255, une quantité de vert de 255 et une quantité de bleu de 255. Ce mélange correspond à la couleur "blanc".
- la couleur RGB (0, 0, 0) indique une quantité de rouge de 0, une quantité de vert de 0 et une quantité de bleu de 0. Ce mélange correspond à la couleur "noir".
- la couleur RGB (255, 0, 0) indique une quantité de rouge de 255, une quantité de vert de 0 et une quantité de bleu de 0. Ce mélange correspond à la couleur "rouge".



Crédits : [lelivrescolaire.fr](http://lelivrescolaire.fr)

Codage RVB pour la couleur dorée (255, 215, 0)

Couleur	Code décimal	Code binaire	Code hexadécimal
R	255	1111 1111	FF
V	215	1101 0111	D7
B	0	0000 0000	00

## Les formats de fichier

De très nombreux formats de fichiers existent pour les images numériques. Certains d'entre eux sont spécifiques à la photographie comme les formats **.raw**, **.tiff** ou **.jpg**. Ils se distinguent les uns des autres par **les algorithmes de compression** qu'ils utilisent pour diminuer le volume de données des images.

## Les métadonnées EXIF

Lors de la création de l'image, l'appareil photographique joint au fichier un certain nombre de données qui ne décrivent pas des pixels. Ces données sont appelées métadonnées EXIF (Exchangeable image file format) et indiquent un ensemble de caractéristiques liées à l'image : résolution, définition, codage des couleurs, lieu et date de prise de vue, etc.

### Images

ID de l'image	
Dimensions	4096 x 3072
Largeur	4096 pixels
Hauteur	3072 pixels
Résolution horizontale	72 ppp
Résolution verticale	72 ppp
Profondeur de couleur	24
Compression	
Unité de résolution	2
Représentation des couleurs	sRGB
Bits compressés/pixel	

### Appareil photo

Marque appareil photo	motorola
Modèle d'appareil photo	moto g(7) power
Focale	F/2

Crédits : [lelivrescolaire.fr](http://lelivrescolaire.fr)

## IV - Le traitement des images numériques

### Les traitements dans l'appareil

Une fois les données collectées lors de la capture de l'image, l'appareil photographique réalise différents traitements automatiques. Certains correspondent aux réglages de l'appareil, d'autres à des améliorations nécessitant l'intervention d'algorithmes de traitement d'image.

### Les retouches par logiciel

Les images peuvent aussi être modifiées après leur obtention par l'appareil photographique. Ces modifications peuvent avoir plusieurs objectifs : artistique (les filtres Instagram), esthétique (lorsqu'on réalise une retouche sur le visage ou le corps), humoristique et parfois calomnieux (les deepfakes). Aujourd'hui, il est facile de modifier des images pour obtenir des images « choc ». Il convient d'être prudent vis-à-vis de telles images et de s'interroger sur leur authenticité.



*Crédits : Creative Travel Projects/Shutterstock*

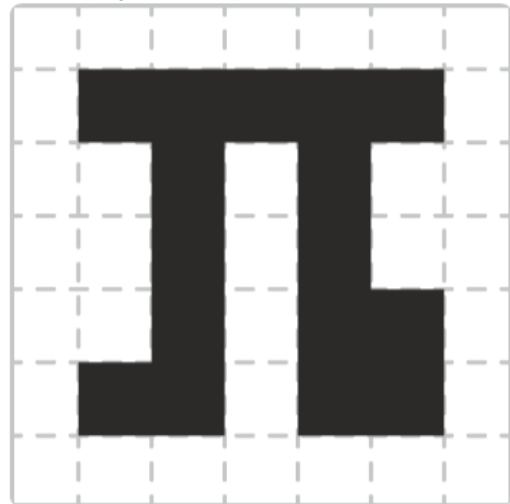
Exemple de photographie avant et après retouche

## V - Exercices

### Exercice 1

On considère l'image ci-contre codée en binaire (0 pour le blanc, 1 pour le noir).

1. Déterminer le nombre de pixel de l'image.
2. Déterminer la définition de l'image.
3. L'image est un carré dont chaque côté mesure 35 pouces. Déterminer la résolution de l'image en dpi.

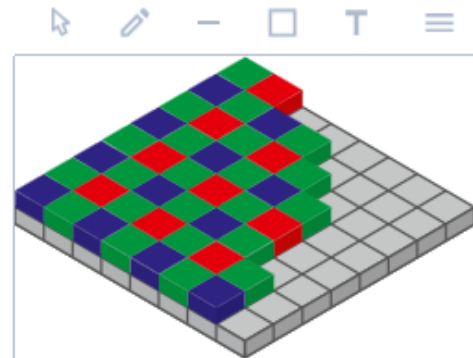


Crédits : Lelivrescolaire.fr

### Exercice 2

Un capteur photographique surmonté de sa matrice de Bayer est représenté ci-contre.

1. Donner le nom des éléments du capteur représentés par un carré gris.
2. Quelle est la fonctionnalité de ces éléments ?
3. Préciser combien d'éléments sont nécessaires au minimum pour coder correctement un pixel.



### Exercice 3

Un logiciel de traitement des images ou certains systèmes d'exploitation permettent d'accéder à l'ensemble des données contenues dans un fichier image.

1. Donner le nom des données contenues dans le fichier et ne servant pas à coder l'image.
2. Donner deux exemples d'informations pouvant être apportées par ces données.
3. Préciser les risques auxquels on s'expose en laissant ces données dans le fichier lors d'un partage sur les réseaux sociaux

### Exercice 4

On rappelle que le codage des couleurs suit le système RVB où chaque pixel est alors codé par trois nombres (allant de 0 à 255) représentant le rouge, le vert et le bleu.

1. Rappeler les codes RVB du cyan et du jaune.
2. En utilisant la page web "Couleurs RBG" (disponible dans la section "liens utiles" du site Web SNT), trouver la couleur dont le code RVB est (255, 96, 208).



## Exercice 5

On considère les métadonnées EXIF d'une photographie.

1. Relever les coordonnées du lieu où a été prise cette photographie.
2. Le jour et l'heure de la prise font-ils partie des données ? Si oui, les indiquer.
3. Identifier l'appareil avec lequel la photographie a été prise.

Global Positioning System	
GPS Altitude	31.9 m
GPS Latitude	6deg 14' 7.620"
GPS Longitude	106deg 49' 30.210"
Image Information	
Date and Time	2018:08:24 15:47:27
Manufacturer	Apple
Model	iPhone 6s
Photograph Information	
Aperture	F2.2
Exposure Bias	0 EV
Exposure Mode	Auto
Exposure Program	Auto
Exposure Time	1/874 s
Flash	No, auto
FNumber	F2.2
Focal Length	4.2 mm
ISO Speed Ratings	25
Metering Mode	Multi-segment
Shutter speed	1/874 s
White Balance	Auto

Crédits : Vulphere/Wikimedia

## Exercice 6

Un magazine souhaite imprimer une photographie de feu d'artifice du nouvel an disponible sous plusieurs définitions. La largeur de la photographie dans le magazine est de 5 cm (soit 2 pouces). La résolution minimale pour une impression de qualité est de 400 ppp.



Crédits : totojang1977/Shutterstock

	Nombre de pixels de l'image	Nombre de pixels en hauteur
Photo 1	518 400	960
Photo 2	360 000	450
Photo 3	230 400	360

### Caractéristiques des images disponibles.

1. Donner, pour chaque photographie, le nombre de pixels en largeur. Écrire alors la définition de chaque image en format *largeur* × *hauteur*.
2. Calculer la résolution de chaque photographie imprimée au format du magazine. En déduire la ou les photographies pouvant convenir.

## Exercice 7 TP python : pacman

L'objectif de ce TP est de créer, à l'aide de Python, une image de définition  $16 \times 16$  de type retro gaming.  
Nous allons, étape par étape, créer une image de **pacman**.



- **Etape 1 : Création d'une image  $16 \times 16$  vide, c'est à dire, composée de 256 pixels blancs.**

```
from PIL import Image
nom_image = "pacman.png"
img = Image.new("RGB", (16, 16), (255,255,255))
img.save(nom_image, "PNG")
img.show("pacman")
```

Explications :

- Ligne 1 : chargement de la librairie pour le traitement d'image avec Python
- Ligne 2 : création d'une variable contenant le nom du fichier de l'image qui sera créée.
- Ligne 3 : création d'une image de 256 ( $16 \times 16$ ) pixels (pour l'instant tous de couleur blanc).
- Ligne 4 : sauvegarde de l'image au format PNG
- Ligne 5 : affichage de l'image

Voici comment python représente votre image :  
Il s'agit d'une grille de pixels avec 16 lignes et 16 colonnes, numérotées de 0 à 15.  
Chaque case représente un pixel.

**Recopiez maintenant le code python dans l'éditeur "Basthon" et exécutez le programme.**

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0																
1																
2																
3																
4																
5																
6																
7																
8																
9																
10																
11																
12																
13																
14																
15																

- **Etape 2 : Remplissage de la première ligne.**

L'objectif est maintenant de remplir la première ligne de la grille. Pour cela nous allons modifier le code précédent en ajoutant 3 nouvelles lignes :

```
from PIL import Image
nom_image = "pacman.png"
img = Image.new("RGB", (16, 16), (255,255,255))
#ligne 1
for i in range(7,11):
    img.putpixel((i,0), (255,255,0))
img.save(nom_image, "PNG")
img.show("pacman")
```



Explications :

- Ligne 4 : Commentaire indiquant que l'on crée la ligne
- Ligne 5 : boucle "for" utilisant une variable *i* représentant le numéro de la colonne du pixel que nous allons modifier. L'instruction *range(7,11)* indique que nous allons mettre à jour les pixels de la colonne 7 à 10 (attention c'est écrit 11, mais il faut interpréter  $11 - 1 = 10$ ).
- Ligne 6 : *img.putpixel((i,0),(255,255,0))*, cette instruction modifie la couleur du pixel de la colonne *i* de la ligne 1. La couleur (255,255,0) correspond au jaune.
- **Etape 3 : Remplissage des autres lignes**

```
from PIL import Image
nom_image = "pacman.png"
img = Image.new("RGB", (16, 16), (255,255,255))
#ligne 1
for i in range(7,11):
    img.putpixel((i,0), (255,255,0))

#ligne 2
for i in range(6,12):
    img.putpixel((i,1), (255,255,0))

#ligne 3
for i in range(5,13):
    img.putpixel((i,2), (255,255,0))

#ligne 4
for i in range(4,14):
    img.putpixel((i,3), (255,255,0))

#ligne 5
for i in range(3,13):
    img.putpixel((i,4), (255,255,0))

#ligne 6
for i in range(2,12):
    img.putpixel((i,5), (255,255,0))

#ligne 7
for i in range(2,11):
    img.putpixel((i,6), (255,255,0))

#ligne 8
for i in range(1,10):
    img.putpixel((i,7), (255,255,0))

#####
###  LIGNE 9 - 15 A COMPLETER  ###
#####

img.save(nom_image, "PNG")
img.show("pacman")
```

**Recoiez le code précédent en n'oubliant pas de compléter le code pour afficher les lignes 9 à 15 de l'image**

## • Etape 4 : L'oeil de Pacman !

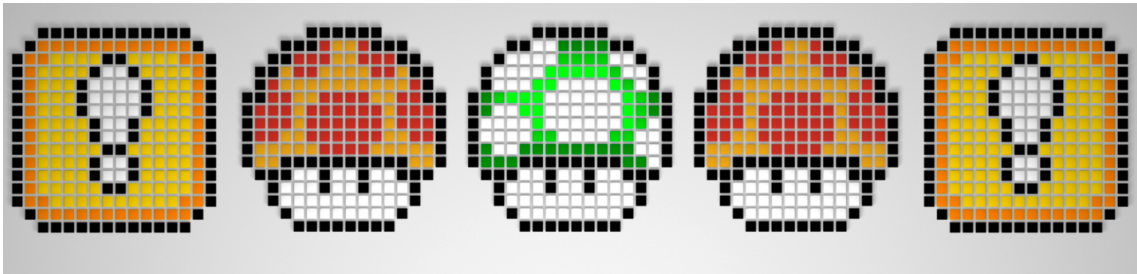
```
# l'oeil de pacman
img.putpixel((7,3),(0,0,0))
img.putpixel((7,4),(0,0,0))
img.putpixel((8,3),(0,0,0))
img.putpixel((8,4),(0,0,0))
```

Recopiez le code précédent à la fin de votre programme (avant le *img.show*). Ce code python modifie en noir la couleur des pixels (7,3), (7,4), (8,3) et (8,4).

### Exercice 8 TP python - Concours image retrogaming - EVALUÉ

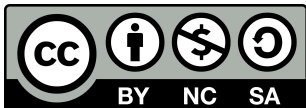
L'objectif de cette exercice est de créer une image retrogaming  $16 \times 16$  sur le même principe que l'exercice précédent.

1. Tu choisiras l'image de ton choix. Voici quelques exemples :



2. Tu enregistreras ton image au format PNG.
3. Tu enregistreras ton programem Python.
4. Tu déposeras ces deux fichiers sur EcoleDirecte.

A l'issue de cet exercice, un concours sera organisé au sein de la classe pour définir la plus belle image retrogaming !



Sources : [lelivrescolaire.fr](http://lelivrescolaire.fr)