

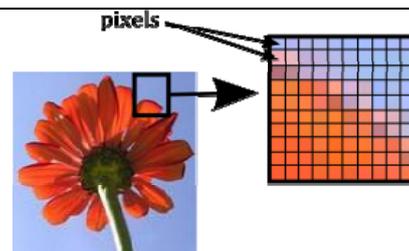
AE - l'image numérique

Doc 1 : pixel, définition, résolution d'une image.

Une image numérique est constituée d'un ensemble discret de points appelés **PIXELS** (contraction de **PICTure ELements**, le plus petit élément d'une image). Chaque pixel possède une couleur.

On appelle **définition** d'une image le nombre total de pixels composant l'image: c'est à dire le nombre de pixels en largeur (x) multiplié par le nombre de pixels en hauteur (y). Ex : Image de 640*480

La **résolution** d'une image numérique définit le degré de détail de l'image. Elle se mesure en pixel par pouce (ppp) ou « dot per inch (dpi) » en anglais (1 pouce = 2,54cm). Plus la résolution est importante, meilleur est la restitution de l'objet photographié.



Doc 2 : codage en informatique :

Le **binaire** est un système de numération utilisant la base 2. On nomme couramment bit (de l'anglais **binary** digit, soit « chiffre binaire ») les chiffres de la numération binaire positionnelle. Un bit peut prendre deux valeurs, notées par convention 0 et 1.

Le **système hexadécimal** est un système de numération positionnel en base 16. Il utilise ainsi 16 symboles, les chiffres arabes pour les dix premiers chiffres et les lettres A à F pour les six suivants.

L'ordinateur traite de informations en binaire ou en hexadécimal.

Tous les caractères d'un clavier, toutes les informations contenues dans une image sont codés en binaire ou en hexadécimal.

On trouve les correspondances dans des tables (ex : table ASCII).

exemple : la lettre "j" s'écrit en binaire: 01101010 et en hexadécimal : 6A

Un mot binaire constitué de 8bits se nomme : un octet (byte en Anglais).

Doc 3 : codage d'une image :

Chaque pixel d'une image est codé par un nombre binaire constitué de bits.

Avec 1 bit : on peut écrire 2 nombres binaires différents : 0 et 1

Avec 2 bits : on peut écrire 4 nombres binaires différents. 00 ; 01 ; 10 ; 11

Avec 3 bits : on peut écrire 2³ nombres binaires différents. 000 ; 001 ; 010 ; 111
ect...

Généralement le codage se fait sur 8 bits soit un octet.

Les nombres binaires permettent de coder la couleur des pixels de l'image.

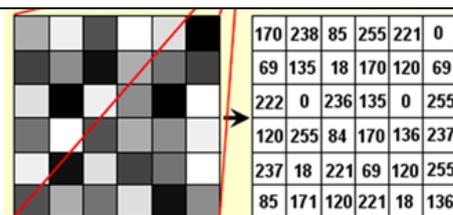
ex : Une image en Noir et Blanc peut-être codée sur 1 bit car nous n'avons besoin que de 2 nombres binaires pour coder ses pixels : Noir = 0 et Blanc = 1.

Doc 4 : image en niveau de gris et image en couleurs

Chaque pixel d'une image en niveaux de gris est codée sur 8bits (un octet) et ne peut prendre que des teintes plus ou moins grises entre le noir et le blanc. Pour une image en **niveaux de gris**, un seul nombre est requis par pixel, les sous-pixels recevant la même information.

Chaque pixel d'une image en couleur RVB24bits est constitué de 3 sous pixels (Rouge, Vert, Bleu) codé chacun sur 8bits (un octet) en binaire ou en hexadécimal. Pour chaque pixel de l'image, un triplet de nombres donne le niveau d'intensité lumineuse des trois sous-pixels **rouge vert bleu** formant le pixel.

La couleur du pixel sera le résultat de la synthèse additive de ces 3 couleurs.



170	238	85	255	221	0
69	135	18	170	120	69
222	0	236	135	0	255
120	255	84	170	136	237
237	18	221	69	120	255
85	171	120	221	18	136



254	247	205	254	247	205	254	247	205
251	244	200	251	244	200	251	244	200
251	245	197	251	245	197	251	245	197
255	250	202	255	250	202	255	250	202
255	250	202	255	250	202	255	250	202
249	243	195	249	243	195	249	243	195

Doc 5 : numération : Passage d'un système de numération à un autre :

<http://sebastienquillon.com/test/javascript/convertisseur.html>

Doc 6 : Définitions, résolution, couleurs et taille d'une image numérique : <http://physique.ostralo.net/images/>

Travail 1: caractéristiques d'une image

Image Noir et Blanc

- Déterminer la définition de cette image.
C'est du 8x8 pixels. Soit au total 64 pixels.
- Combien de couleurs possède cette image ?
Il y a 2 couleurs.
- En déduire combien de bits permettent de coder les couleurs de l'image.
1 bit (qui permet de différencier $2^1 = 2$ valeurs différentes) est suffisant pour coder les couleurs. Par exemple, 1 pour le blanc, 0 pour le noir.

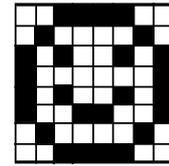
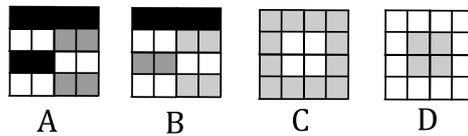


Image en niveaux de gris

- Combien de nuances de gris peuvent prendre les pixels d'une image codées sur 8 bits en niveaux de gris ?
Avec 8 bits on peut écrire $2^8 = 256$ nombre binaires différents et donc coder 256 dégradés de gris différents.
- Dans ce type de codage, le blanc a pour valeur 255 et le noir 0. Plus un gris sera foncé, plus la valeur associée sera faible.
- Associer à chaque image A, B, C, D un tableau exprimé en décimal suivant:



$$\mathbf{1} \begin{pmatrix} 255 & 255 & 255 & 255 \\ 255 & 127 & 127 & 255 \\ 255 & 127 & 127 & 255 \\ 255 & 255 & 255 & 255 \end{pmatrix} \quad \mathbf{2} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 255 & 255 & 63 & 63 \\ 0 & 0 & 255 & 255 \\ 255 & 255 & 63 & 63 \end{pmatrix} \quad \mathbf{3} \begin{pmatrix} 127 & 127 & 127 & 127 \\ 127 & 255 & 255 & 127 \\ 127 & 255 & 255 & 127 \\ 127 & 127 & 127 & 127 \end{pmatrix} \quad \mathbf{4} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 255 & 255 & 127 & 127 \\ 63 & 63 & 255 & 255 \\ 255 & 255 & 127 & 127 \end{pmatrix}$$

FF est le blanc, 00 le noir. Plus le nombre en decimal est faible, plus le gris est foncé. donc :

La A contient une ligne de noir (00 00 ...) et du gris foncé (63) alors que la B contient du gris clair (127)

En résumé : A=2 B=4 C=3 D=1

Image couleur RVB

- Soit une image prise par un appareil photo numérique avec une définition de 12Mpx et codée en RVB24bits
Combien de nuances de couleurs peut contenir cette image ?
Avec 24 bits, on peut coder $2^{24} = 16,7$ Million de valeurs différentes. Donc les pixels de cette image peuvent avoir 16,7 millions de couleurs différentes

- Déterminer la taille en octet du fichier de l'image obtenue dans un format non compressé.

1 pixel est codé sur 24 bits (8bits par sous-pixel et chaque sous-pixel correspond à une couleur R ou V ou B), donc sachant que 8bits représente 1 octet, il faut 3 octets par pixel donc l'image faisant $12 \cdot 10^6$ pixels elle occupera une place de $12 \cdot 10^6 \times 3$ octets soit 36Mo.

Travail 2: Vérifications expérimentales

Pour une image en Niveaux de Gris

- Que représentent les nombres du tableau obtenu à droite de l'image en Niveau de gris de l'annexe?
- les nombres correspondent à la valeur en décimal du codage sur 8bits des différents niveaux de gris de chaque pixel
- En quel système de numération sont-ils écrits? en décimal (base 10)
- Par quel nombre est représenté le blanc ? 255 le noir ? 0
- A quels nombres binaires sur 8bits cela correspond ? $255_{(10)} = 11111111_{(2)}$ et $0_{(10)} = 00000000_{(2)}$

Pour une image en couleurs RVB :

- Que représentent les nombres du tableau obtenu à droite de l'image en couleur de l'annexe?
pour vous aider à répondre utiliser le site <https://imagecolorpicker.com/fr/> (cliquer sur des zones de l'image et vérifier les nombres obtenus sur le site un peu plus bas sur la page).
- Chaque pixel est représenté par 3 nombres, 1 pour chaque sous-pixel. Chacun de ces nombres correspond à la valeur en décimal du codage sur 8bits des couleurs Rouge, Vert, Bleu de chaque

sous-pixel.

- En quel système de numération sont-ils écrits? **en décimal (base 10)**

Dans les logiciels de dessin les tableaux représentant les couleurs des pixels sont écrits en hexadécimal.

Utiliser le site suivant : <http://www.proftnj.com/RGB3.htm>

pour observer le codage des couleurs en hexadécimal des pixels d'une image

- A quel nombre hexadécimal correspond le noir ? **00 00 00** le blanc? **FF FF FF**
- Quel est l'intérêt de cette écriture par rapport au système binaire ?

le système hexadécimal occupe moins d'espace dans l'ordinateur car un octet est réduit à 2 caractères au lieu de 8.

Travail 3 : Formats d'images

Télécharger l'image paysage.bmp depuis [l'ENT](#)

https://leon-blum.mon-ent-occitanie.fr/lectureFichiergw.do?ID_FICHER=1515363744607

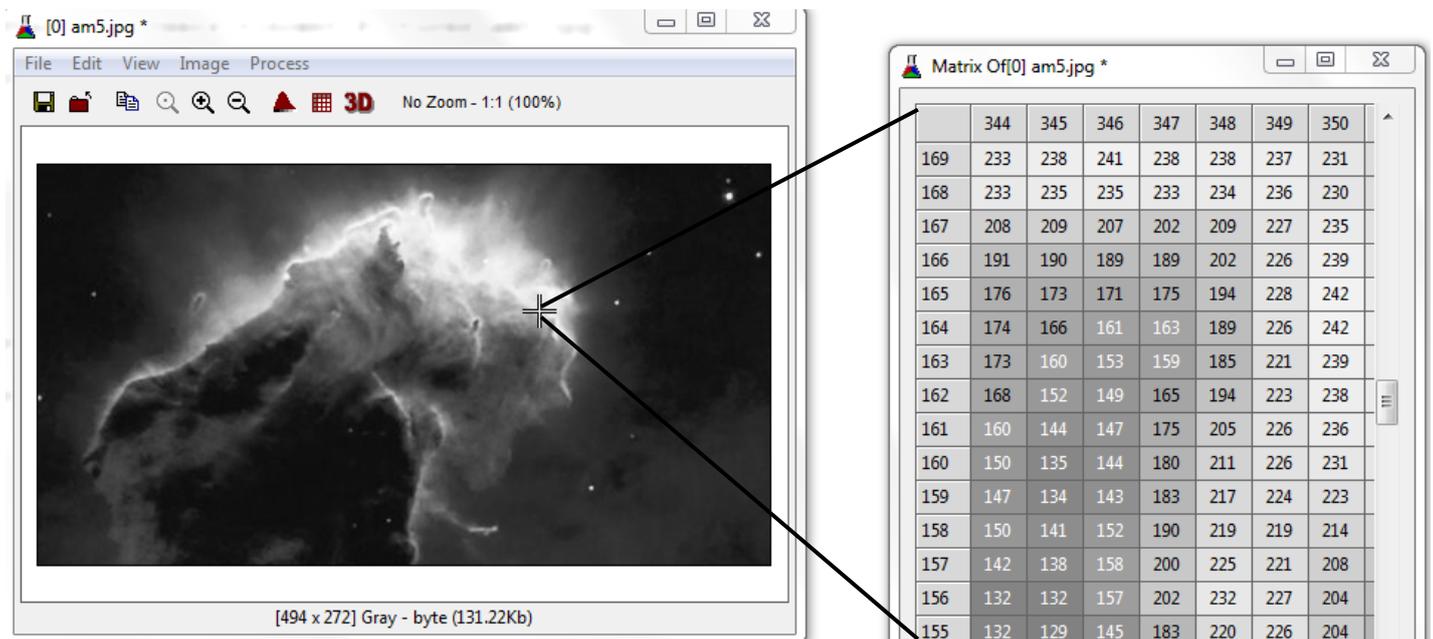
Ouvrir les propriétés de l'image avec un logiciel d'image (paint, photofiltre, gimp...) (menu fichier/propriétés image)

- Déterminer la définition de l'image et sa taille en octet. **520x447px**
- Vérifier par un calcul la taille de cette image. **520x447=232440px soit 232440x3 octets soit environ 700ko** **taille réelle dans les propriétés 681 ko.**

Sauvegarder cette image en changeant son format en jpeg (fichier /enregistrer-sous puis changer le "type").

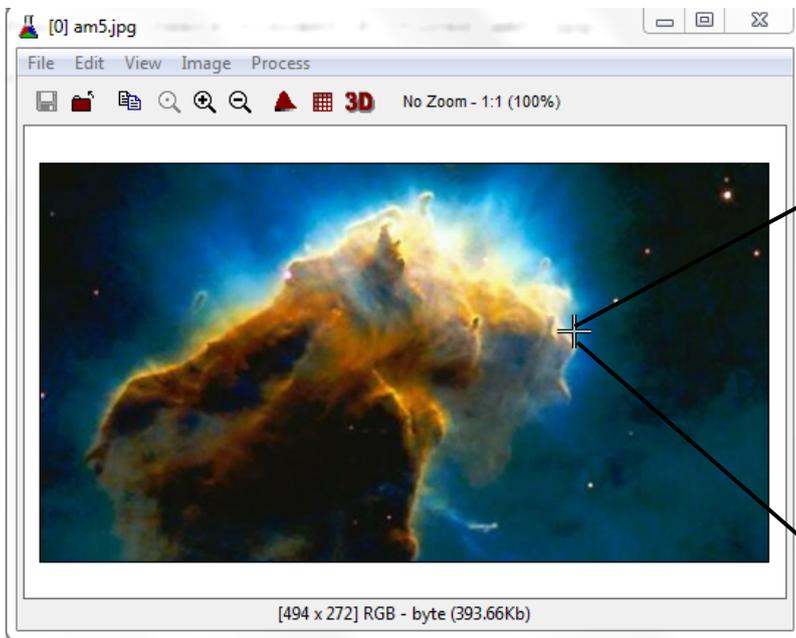
- Déterminer la nouvelle taille en octet de l'image. **dans les propriétés 127ko**
- Pourquoi utilise-t-on le format jpeg ? Quel est l'inconvénient de ce changement de format?
- **jpeg est un format compressé qui permet de gagner en taille de fichier mais il y a des pertes d'informations et de qualité**

Annexe



The image shows a screenshot of an image viewer window titled "[0] am5.jpg *". The main window displays a grayscale image of a landscape with a bright, glowing area. Below the image, it shows the dimensions and format: "[494 x 272] Gray - byte (131.22Kb)". To the right, a smaller window titled "Matrix Of[0] am5.jpg *" displays a grid of numerical values representing the pixel intensity matrix. The matrix is 17 rows by 8 columns, with values ranging from 129 to 344. A black arrow points from a specific location in the image to the corresponding row and column in the matrix.

	344	345	346	347	348	349	350
169	233	238	241	238	238	237	231
168	233	235	235	233	234	236	230
167	208	209	207	202	209	227	235
166	191	190	189	189	202	226	239
165	176	173	171	175	194	228	242
164	174	166	161	163	189	226	242
163	173	160	153	159	185	221	239
162	168	152	149	165	194	223	238
161	160	144	147	175	205	226	236
160	150	135	144	180	211	226	231
159	147	134	143	183	217	224	223
158	150	141	152	190	219	219	214
157	142	138	158	200	225	221	208
156	132	132	157	202	232	227	204
155	132	129	145	183	220	226	204



Matrix of [0] am5.jpg

	360	361	362	363
172	125 189 201	115 183 202	101 179 202	85 175 202
171	116 185 200	105 180 199	91 173 197	76 167 196
170	110 183 200	99 177 199	82 168 195	65 160 192
169	99 177 199	87 171 197	70 161 192	52 150 187
168	86 170 196	75 164 196	56 153 188	38 141 182
167	75 161 194	63 157 193	47 148 190	33 140 186
166	68 160 197	57 156 197	44 149 194	31 141 190
165	62 158 196	52 155 198	37 148 194	26 140 192
164	61 160 199	46 155 196	31 146 191	20 138 188
163	65 168 203	47 159 197	27 146 188	14 136 183
162	70 175 207	48 165 200	25 149 187	11 137 178
161	70 178 207	48 167 199	24 151 186	11 139 178
160	68 176 204	46 165 195	23 150 183	12 140 177
159	96 179 197	61 168 200	27 154 199	8 145 199
158	107 184 200	69 173 202	33 156 198	10 144 197