

Systemes Multimédia

Crédit : 4,5 Coefficient de la Matière : 2

Enseignant : Dr. Abdelli Email : Abdelli@hotmail.com

Objectifs de l'enseignement

*Donner à l'étudiant les fondements de base de toutes les technologies du multimédia.
L'ensemble des cours théoriques et pratiques permettront à l'étudiant de produire une
application multimédia intégrant l'image, la vidéo, le son et le texte.*

Mode d'évaluation: *Examen théorique et Examen pratique*

<https://sites.google.com/site/abdelkrimabdelli/teaching>

Programme

Chapitre I : Introduction à l'image : Concepts généraux

- Codage Informatique des couleurs : Représentation de couleur : Systèmes RVB, CMJN, TSL, YUV
- Affichage et acquisition des couleurs :
- Introduction aux Ecrans couleurs et aux équipements de capture d'image: Fonctionnement des Equipements E/S d'image (Ecrans, Scanners, Appareil Photo).

Chapitre 2 : L'imagerie numérique

- Définitions
- Formats d'images (Vectorielle et Matricielle).
- Méthodes de représentation d'images (matricielle).
- Méthodes de compression d'images.
- La méthode de compression JPEG.

Pratique : Création et traitement d'images : WINDOWS PAINT, ACTIF GIF, ADOBE PHOTOSHOP. (**TEST TP**)

Chapitre 3 : Le son en multimédia

- Bases théoriques et technologiques de l'audio
- Méthodes de compression du signal audio
 - compressions sans perte
 - compression avec perte
- Codage du son
 - le codage PCM
 - le codage MPEG
- La méthode de compression MP 1 2 et 3

Pratique : Création et édition sonore avec : Magnetophone
Windows, AUDACITY. (**Test TP**)

Chapitre 4 : La vidéo en multimédia

- Principes de représentation de la vidéo.
- Différents formats de la vidéo
- Numérisation du signal vidéo
- Formats vidéo numériques
- Compression du signal vidéo
 - Sans perte.
 - Avec perte.
- Format MPEG vidéo 1 2, 3 et 4.

Pratique : Création et édition vidéo avec : MOVIE MAKER
ADOBE PREMIER.

Chapitre 5 : Standards et édition de documents multimédia

- Les standards d'indexation MPEG7, MPEG21.
- Le langage d'intégration multimédias synchronisés (SMIL)
- Edition de présentations multimédia SMIL

Pratique : édition d'une présentation Multimédia (Travail pratique optionnel à rendre en fin d'année).

Chapitre 0 :

Introduction aux systèmes multimédia.

0.1. Qu'est ce que le Multimédia ?

- Le mot **multimédia** est apparu vers la fin des années 1980, lorsque les **CD-ROM** se sont développés. Il désignait alors les applications qui, grâce à la *capacité de stockage* du CD et aux *capacités de traitement l'ordinateur*, pouvaient *générer, utiliser ou piloter* différents médias simultanément : *musique, son, image, vidéo*.
- Aujourd'hui on utilise le mot multimédia pour désigner *toute application utilisant ou servant à travailler sur au moins un média spécifique*.

0.1. Qu'est ce que le Multimédia ?

- On définit le multimédia aussi comme l'industrie du « *contenu numérique* » (*digital content*) en anglais.
- Les appareils électriques, électroniques, informatiques portent encore parfois le terme multimédia car ils permettent la visualisation, ou la diffusion d'un document multimédia.
- Il faut faire la différence entre le *contenant* : ordinateurs, matériels, câbles..., et le *contenu* : l'information délivrée, transmise, diffusée.

0.1. Qu'est ce que le Multimédia ?

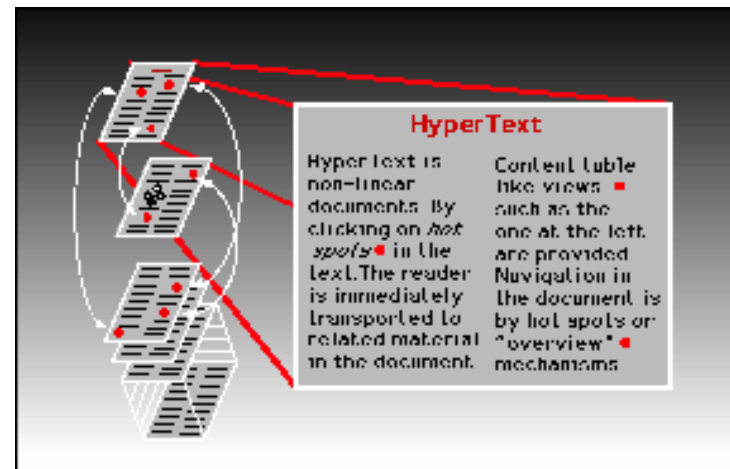
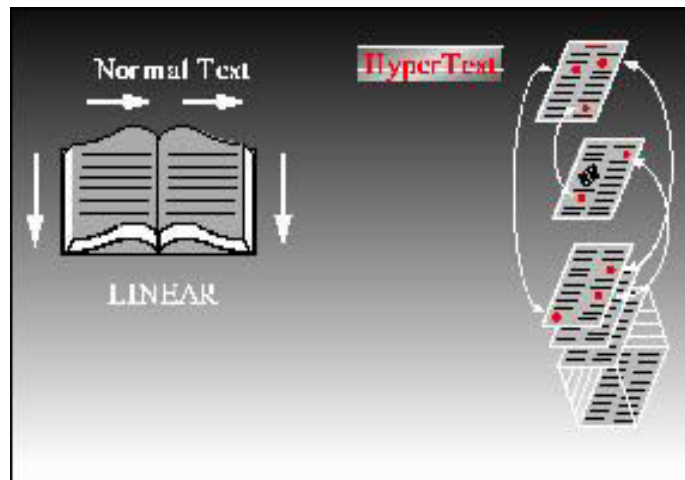
- Un *document multimédia* se caractérise par le regroupement sur un même support d'au moins deux des éléments suivants : *texte, son, image fixe, image animée* sous forme numérique.

Ces éléments sont accessibles via un programme informatique (**logiciel**) autorisant un degré plus ou moins élevé *d'interactivité* entre l'utilisateur et les éléments précités.

Par *interactivité* nous entendons les possibilités de *navigation, d'hypernavigation* (par le biais d'hyperliens), de recherche d'information (libre ou guidée) et d'aide en ligne.

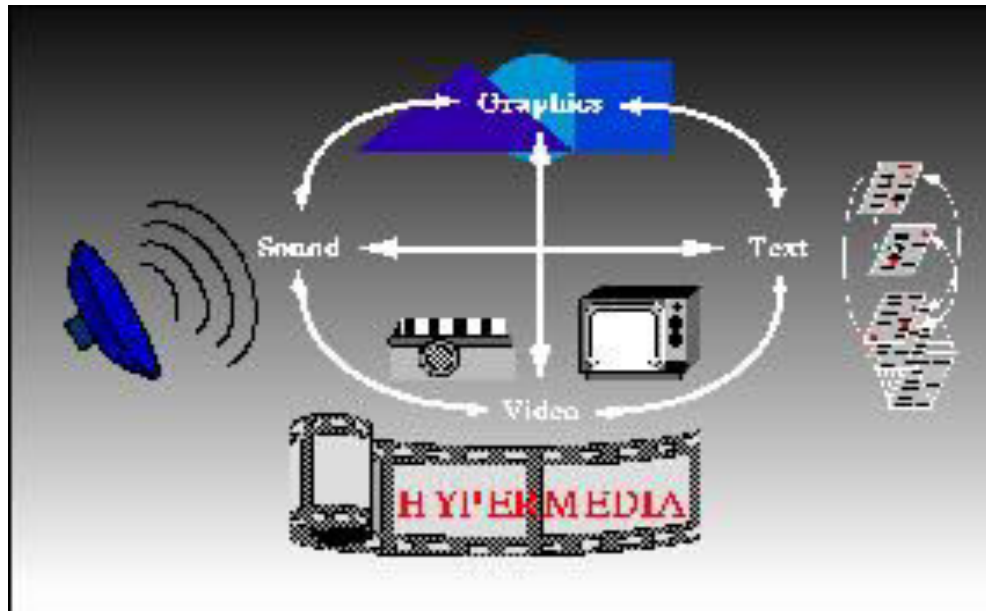
0.2. HyperText et HyperMedia

L'*hypertexte* est un texte qui contient des liens vers d'autres textes. Le terme a été inventé par Ted Nelson vers 1965. L'hypertexte est donc généralement non-linéaire (comme indiqué ci-dessous).



0.2. HyperText et HyperMedia

- **Définition du Hypermédia** : Hypermédia n'est pas basé que sur du texte. Il peut inclure d'autres médias, par exemple, graphiques, images, et surtout les **médias continus** - Son et vidéo. Le *World Wide Web* (WWW) est le meilleur exemple d'application hypermedia.



0.3. Caractéristiques des systèmes Multimédia

Un système multimédia possède quatre caractéristiques fondamentales :

1. Les systèmes multimédias doivent être *contrôlés par ordinateur*.
2. Les systèmes multimédias doivent être *intégrés*.
3. L'information doit être représentée *numériquement*.
4. L'interface finale présentant les médias est généralement *interactive*.

0.4. Défis pour les Systèmes Multimédia

La dissémination du contenu multimédia sur un réseau informatique rend ces *applications distribuées*. Un système Multimédia doit délivrer une variété de médias en simultané. Plusieurs sortes de problèmes peuvent apparaître :

- Des *relations temporelles* entre plusieurs médias peuvent exister : Séquençage dans médias dans un ordre préétabli. Synchronisation - inter-média (lors de la lecture d'une vidéo sonore par exemple).
- Comment spécifier et sauvegarder des scénarios temporels et comment les interpréter et comment remédier aux erreurs de dé-séquençage (temps de latence du réseau).

0.4. Défis pour les Systèmes Multimédia

- **La numérisation des contenus multimédia** : Scannerisation (graphiques, images fixes), échantillonnage (audio/vidéo).
- Le contenu de l'audio/vidéo étant volumineux- par conséquent, le stockage, le transfert (bande passante) et les frais généraux de traitement sont élevés. Les techniques *de compression* de données deviennent un impératif.
- **MBone (Autoroutes du multimedia)** : Multicast Backbone: Equivalent of conventional TV and Radio on the Internet. (Wimax, Optical Network,...)
- Commutation et Routages, Protocoles, Application, Codage/Compression, bases de données multimédia, *indexation multimédia*, ..etc.
- Algorithmes de correction d'erreurs et de calcul du temps élastique.

0.5. Exigences des systèmes Multimédia

- **Grande Puissance de traitement** : afin de faire face au traitement en temps réel de la masse importante de données des médias. Ceci implique des équipements appropriés.
- **Format pour la délivrance et du stockage du contenu multimédia**: afin de délivrer le contenu multimédia en temps réel de le compresser et de le décompresser-- *e.g.* Video/Audio Streaming. Nécessite des équipements et des logiciels s'exécutant en temps réel (encodeur, codecs, lecteurs).
- **Système E/S performant** : E/S vers le système des fichiers doit être efficace et rapide. Doit garantir un enregistrement en temps réel avec lecture des données.

0.5. Exigences des systèmes Multimédia

- Pour permettre un accès performant et rapide aux systèmes des fichiers. Nécessite un transfert direct sur disque, Ordonnancement des tâches en temps réel, un système d'interruption flexible, I/O streaming *etc.*
- **Enregistrement et mémoire** : Unités de stockage importantes (de 50 -100 Gb voire plus) avec mémoire vive importante (50 - 100 Mb voire plus). Aussi, un système de Caches hiérarchiques est souvent requis pour une gestion efficace.
- **Système Réseau** : Système Client-Server, protocoles adaptés.
- **Logiciels** : Outils nécessaires au traitement et à la délivrance des medias.

0.6. Matériel nécessaire pour les systèmes Multimédia

Equipements de Capture: Vidéo Camera, Vidéo enregistreur, Audio Microphone, claviers, souris, traceur de graphiques, équipement d'acquisition 3D, capteurs tactiles, Equipement d'échantillonnage et de compression.

Equipements de Stockage : Disques durs, CD-ROMs, Jaz/Zip drives, DVD, *etc*

Réseaux de Communication: Réseaux physiques (Ethernet, Token Ring, FDDI, ATM) ; Réseaux applicatifs (Intranets, Internet).

Interfaces de sortie : CD-Quality speakers, HDTV, SVGA, Hi-Res monitors, imprimantes couleurs *etc.*

Equipement de traitement Multimedia Desktop machines, Workstations.

0.7 Quelques applications multimédias

- **Animation** : dessin animé, image de synthèse ;
- **Conception assistée par ordinateur (CAO)** ;
- **Développement Web** ; Dispositifs immersifs ;
- **Enseignement assisté par ordinateur (EAO)** : enseignement à distance, didacticiel ;
- **Graphisme** : infographie, dessin bitmap, dessin vectoriel ;
- **Jeu vidéo** : jeu d'aventure, jeu de stratégie, jeu d'action, etc. ;
- **Publication assistée par ordinateur (PAO)** : Mise en page, photocomposition ;
- **Musique assistée par ordinateur (MAO)** : synthèse sonore, échantillonneur, MIDI ;
- **Montage assisté par ordinateur (son et/ou vidéo)** : vidéo ; VoIP ;
- **Visio-conférence** : GnomeMeeting, NetMeeting.
- **Réalité virtuelle**

I.6 Quelques applications multimédias

- **Les métiers du multimédia** : Journaliste, Graphiste, Webdesigner, Animateur3D, Monteur vidéo, Infographiste, Publiciste, Monteur vidéo films, Art numérique, Développement, Concepteur de site internet, Administrateur Base de données Multimédia.

Chapitre I : Introduction à l'image : Concepts généraux

Email : Abdelli@hotmail.com

Site Web :

<https://sites.google.com/site/abdelkrimabdelli/teaching>

I.1. Codage informatique des couleurs

Le codage d'une **couleur** peut se faire sur **32 bits**, dont **24 bits** sont utilisés pour coder la couleur, les 8 bits restants étant :

- soit inutilisés ;
- soit, à coder une information de transparence dite *alpha channel*. À travers ce pixel de l'image "passera" en partie la couleur d'un pixel d'une autre image placée dans la même fenêtre, mais « derrière » la première image (technique dite *alpha blending* en anglais).

Les *24 bits d'une couleur* se décomposent en 3 fois 8 bits :

Généralement on utilise le codage **RVB** (*Rouge, Vert, Bleu*) (ou **RGB** de l'anglais *Red, Green Blue*).

- 8 bits sont consacrés à la teinte primaire **rouge** ;
- 8 bits sont consacrés à la teinte primaire **vert** ;
- 8 bits sont consacrés à la teinte primaire **bleu**.

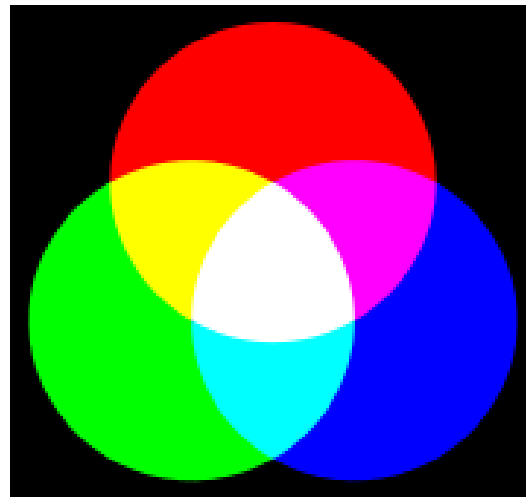
I.1. Codage informatique des couleurs

Le codage **RVB** :

- Ces trois couleurs sont les couleurs primaires en **synthèse additive**. Elles correspondent en fait à peu près aux trois longueurs d'ondes auxquelles répondent les trois types de cônes de l'œil humain. L'addition des trois donne du **blanc** pour l'œil humain. Elles sont utilisées en **éclairage** (*lorsque le support d'affichage est en noir : écran*), afin d'obtenir toutes les couleurs visibles par l'homme.
- Ce système est utilisé et mis en œuvre dans de nombreux périphériques numériques : en entrée acquisition (scanner-couleurs, appareil photo numérique, caméscope...) comme en sortie (écran en couleurs, photocopieuse-couleurs...).

I.1. Codage informatique des couleurs


- Couplées deux à deux ces couleurs donnent les couleurs cyan, magenta et jaune secondaires en synthèse additive et primaires.



Rouge, vert et bleu s'additionnant aux intersections

I.1. Codage informatique des couleurs

Une séquence de 8 bits permet de coder un nombre entier compris entre 0 et $V_{max} = 255$: en effet, 2^8 vaut 256 . Par conséquent, la valeur de la composante rouge d'un pixel peut être représentée selon 256 niveaux différents (allant du 0 , absence de rouge, à 255 , rouge d'intensité maximum). Et il en est de même pour les 2 autres composantes primaires, le vert et le bleu. *Au fait, l'intensité dénote la luminosité de la couleur.*

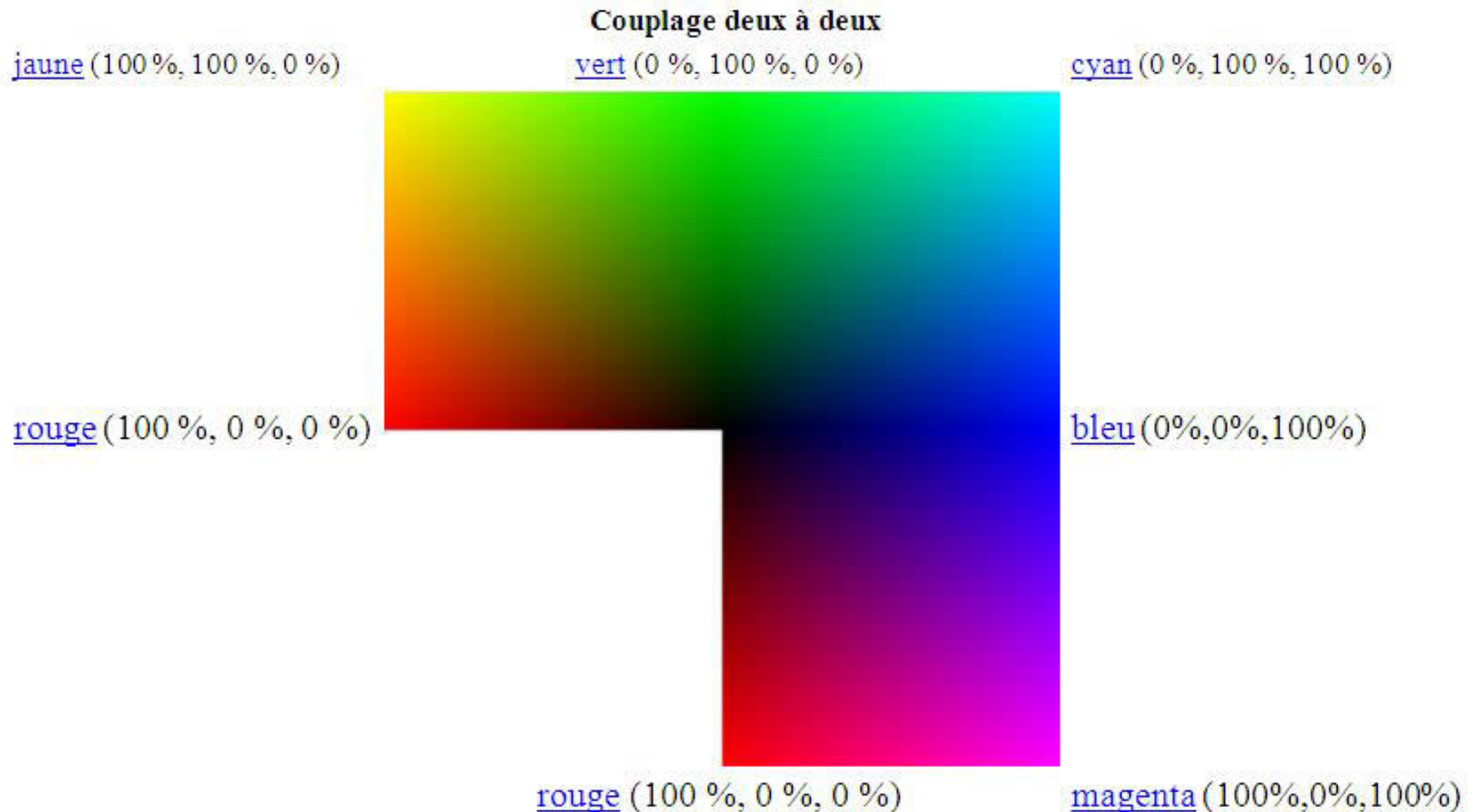
Exemple : Le carré ci-contre  a une couleur dont les caractéristiques **RVB** sont les suivantes :

- composante rouge : 251 , **11111011** ;
- composante verte : 208 , soit **11010000** ;
- composante bleue : 151 , soit **10010111**.

Le codage binaire sur 24 bits de cette couleur est donc le suivant :
111110111101000010010111.

I.1. Codage informatique des couleurs

Couplage deux à deux des couleurs RVB



I.1. Codage informatique des couleurs

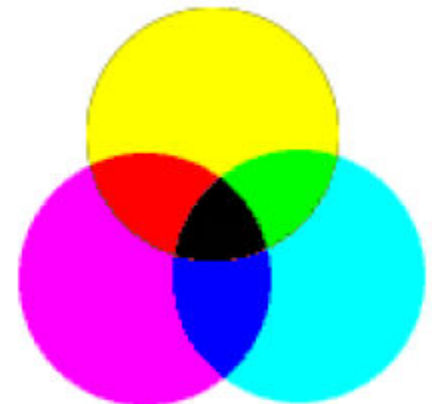
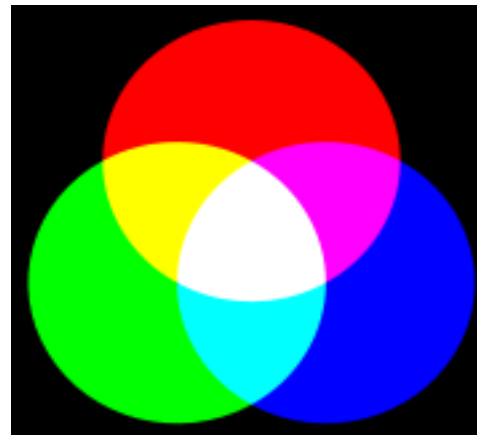
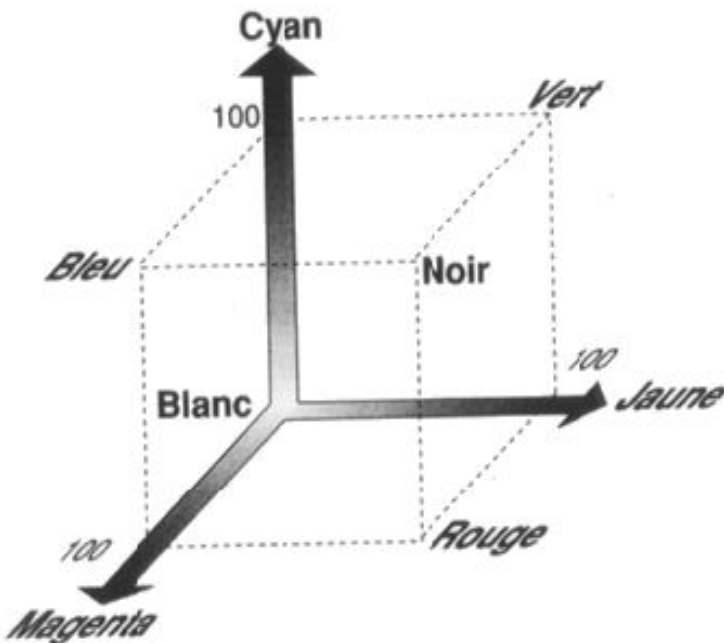
Le système **CMJN** (CMYK en anglais) :

En imprimerie, le système **RVB** est mal adapté car le support est rarement sombre mais plutôt claire. On utilise alors une synthèse de couleurs soustractive avec les primaires (*Cyan*, *Magenta*, *Jaune*) auquel on ajoute le **Noir** car la composition des 3 primaires ne donne pas en général un noir satisfaisant. On travaille alors en *quadrichromie*. Il est aussi plus économique à l'impression de n'appliquer qu'une couleur (pour le noir) au lieu de trois.

Le codage de base est le plus souvent en **32 bits**. Sur chaque couche le pixel est défini par un pourcentage de la couleur correspondante.

I.1. Codage informatique des couleurs

- **Synthèse soustractive** : Le support est de couleur **blanche** et contient donc du **rouge**, du **vert** et du **bleu**. Le **Cyan** absorbe le **rouge**, le **Magenta** absorbe le **vert** et le **Jaune** absorbe le **bleu**. Le mélange des trois couleurs avec du **blanc** absorbe tout et produit du **noir**.



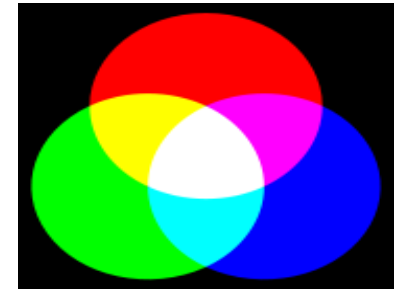
I.1. Codage informatique des couleurs

Synthèse additive :

rouge+**vert**= *Jaune* ; **rouge** +**bleu** = *Magenta* ; **vert**+ **bleu**=*Cyan*

rouge+**vert**+ **bleu**= BLANC ;

absence de couleurs = NOIR



Synthèse Soustractive :

BLANC-**rouge**= *Cyan*;

BLANC- **bleu** = *Jaune*;

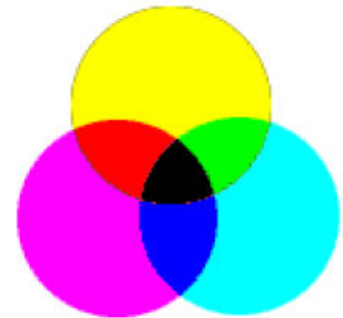
BLANC-**vert** = *Magenta*

Cyan+ *Magenta* = BLANC-**rouge** -**vert**= **bleu** ;

Jaune + *Magenta* = BLANC- **bleu** -**vert**=**rouge**;

Jaune + *Cyan* = BLANC- **bleu** -**rouge** =**vert**

Jaune + *Cyan*+ *Magenta* = BLANC- **bleu** -**rouge** -**vert**= Noir



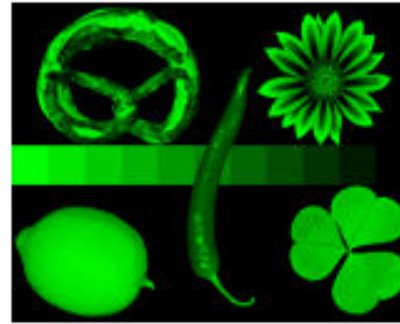
I.1. Codage informatique des couleurs



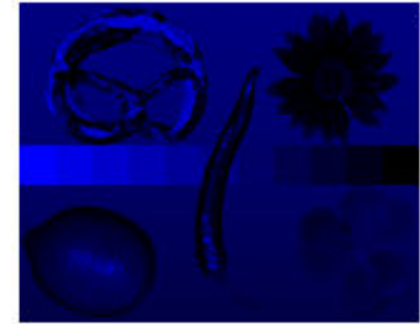
Original Image



Red Color Component



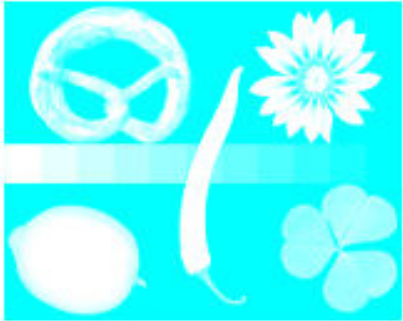
Green Color Component



Blue Color Component



Original Image



Cyan Color Component



Magenta Color Component



Yellow Color Component

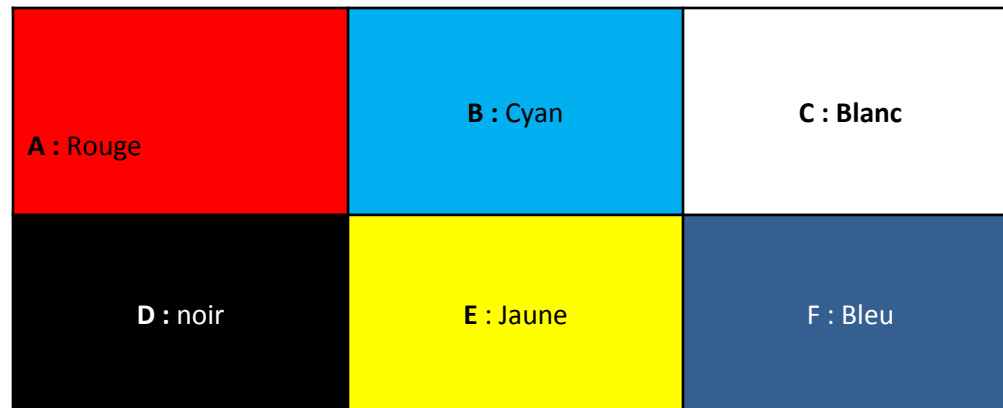
www.equasys.de

www.equasys.de

I.1. Codage informatique des couleurs

Exercice :

Soit l'Image suivante composée de six zones (a,b,c,d,e f) colorées chacune avec une teinte unie. Donnez les composantes Rouge, Vert, Bleu, magenta, cyan et jaune pour cette image.



I.1. Codage informatique des couleurs

Solution : Système RVB

A : Rouge	B : Cyan	C : Blanc
D : noir	E : Jaune	F : Bleu

A :	B	C :
D :	E :	F

A	B :	C :
D :	E :	F :

A :	B :	C :
D :	E : J	F :

I.1. Codage informatique des couleurs

Solution : Système CMJ

A : Rouge	B : Cyan	C : Blanc
D : noir	E : Jaune	F : Bleu

A :	B	C :
D :	E :	F

A :	B :	C :
D :	E :	F :

A :	B :	C :
D :	E :	F :

I.1. Codage informatique des couleurs

Le codage TSL

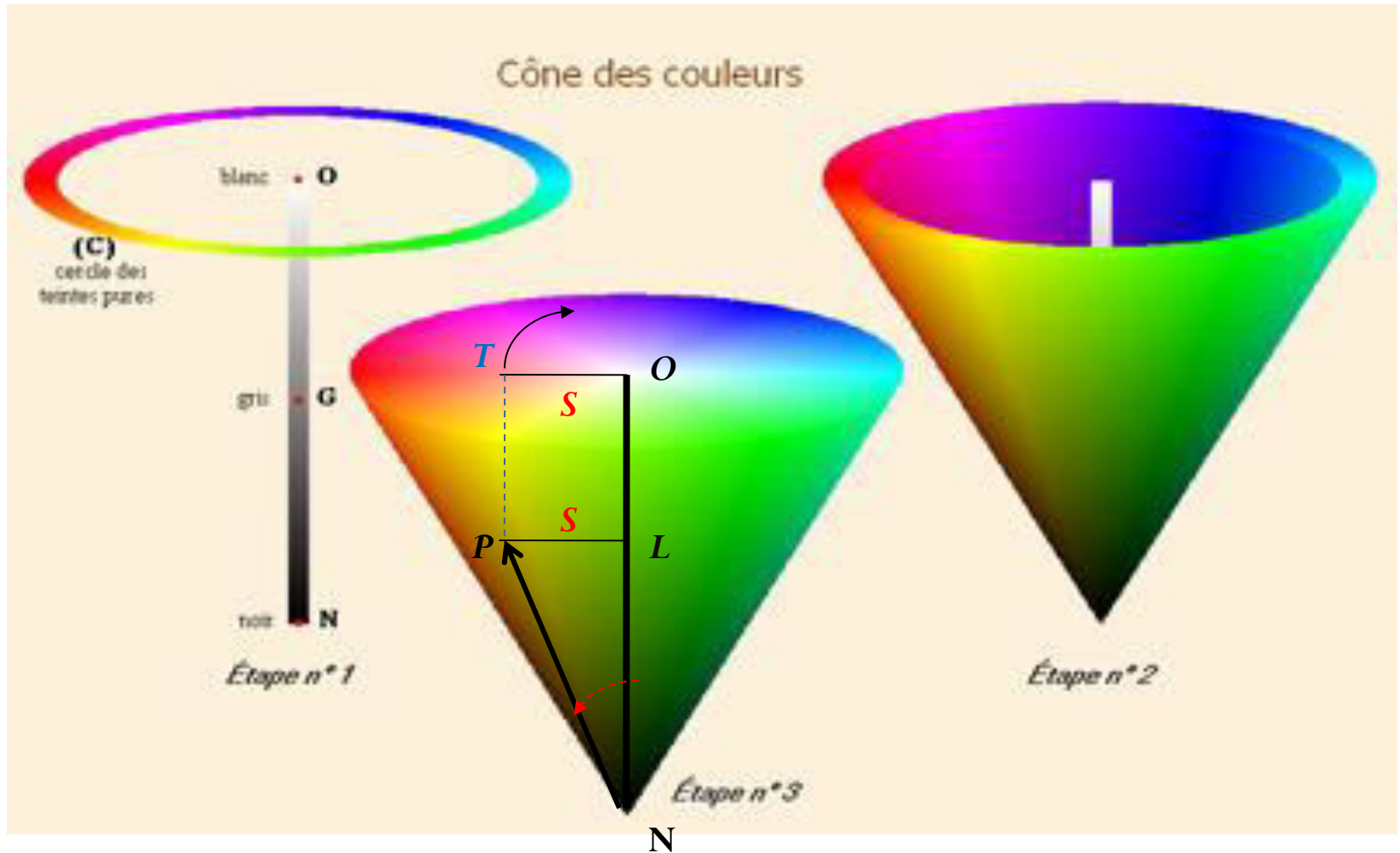
TSL pour (*Teinte, Saturation, Luminance*) ou **HSL** (*Hue, Saturation, Luminance*). Chaque élément est codé sur un intervalle. De manière théorique la teinte varie (*de 0 à 360 °*), la saturation (*de 0 à 100%*) et la luminance (*de 0 à 100 %*). En pratique ces 3 paramètres sont mises à l'échelle d'un octet de [0,255] pour gain mémoire et facilité d'utilisation.

 Teinte : **23** ; Saturation : **220** ; Luminance : **189**.

- Le codage TSL, destiné aux opérateurs humains, est adapté à la caractéristique de leurs rétines : une personne entraînée peut d'ailleurs donner avec une approximation satisfaisante les valeurs **TSL** d'une couleur qu'on lui présente. enfin, le langage **TSL** de définition des couleurs permet de définir facilement certains des dégradés de teinte que la représentation **RVB** ne permet pas de définir aussi facilement.

I.1. Codage informatique des couleurs

Le cône des couleurs :



I.1. Codage informatique des couleurs

Le cône des couleurs :

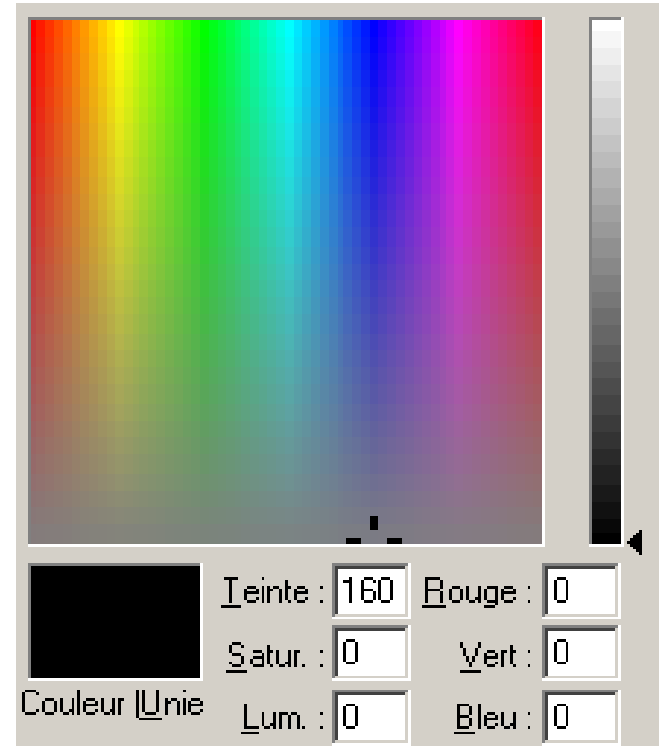
Représentation de l'espace *tri-dimensionnel* des couleurs sous forme de cône : l'axe vertical du cône correspond aux teintes grises (*saturation S nulle, et luminance L variant depuis une valeur nulle en N et une valeur maximale en O*) ; un point P quelconque situé à l'intérieur du cône correspond à une couleur dont les composantes T, S, L peuvent être définies ainsi :

- la teinte T varie linéairement avec la valeur de l'angle orienté du vecteur OH dans le plan du cercle C , le point H étant la projection de P sur ce plan,
- la saturation S est proportionnelle à l'angle (NO, NP) ,
- la luminance L dépend de la valeur conventionnelle attribuée à la luminance des couleurs complètement saturées situées au bord du cercle C .

I.1. Codage informatique des couleurs

Sélection des couleurs : Un outil de sélection de couleur comporte en général au minimum 4 parties :

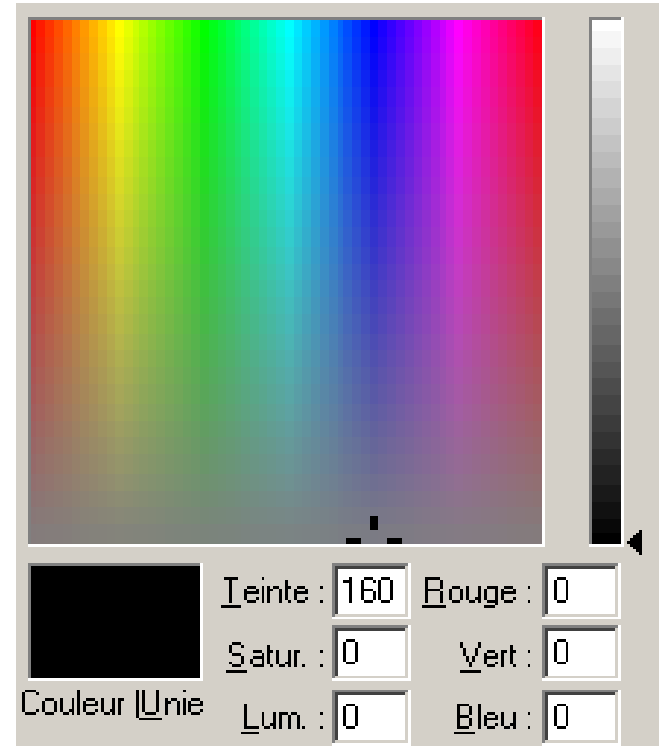
- 2 parties visuelles de choix dont l'une est un carré et l'autre un rectangle étroit dressé sur son petit côté,
- 1 partie visuelle d'affichage (petit rectangle rempli de la couleur choisie), et enfin
- 1 partie purement numérique donnant à la fois les composantes TSL et RVB de la couleur choisie par l'opérateur.



I.1. Codage informatique des couleurs

Sélection des couleurs : Dans tous les cas, la définition d'une couleur par ses composantes **TSL** exige un triple choix qui doit nécessairement être réalisé par l'opérateur en 2 temps : Le carré de gauche représente le choix des 2 composantes **T** et **S** alors que le rectangle de droite représente le choix de la composante **L**. Deux méthodes s'offrent à lui :

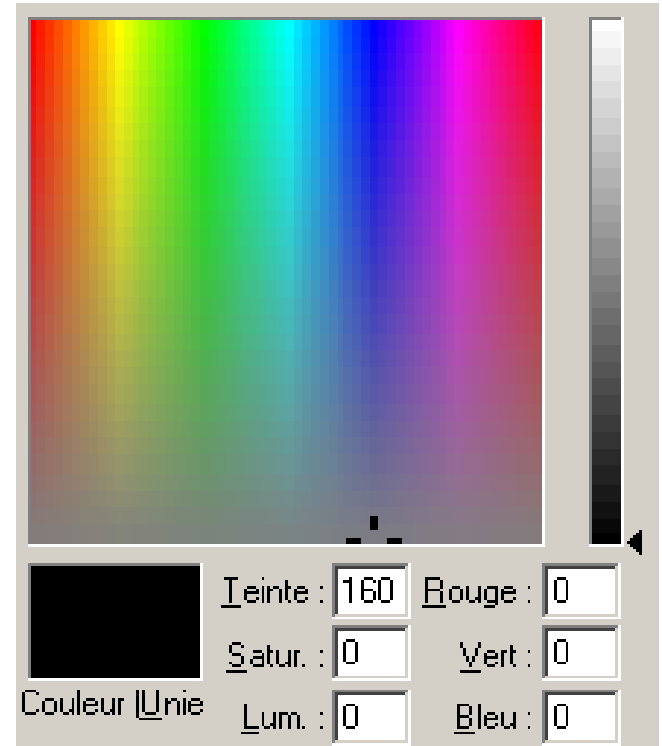
1. il peut d'abord choisir dans l'espace à 2 dimensions du carré un point de couleur (ce qui définit les composantes **T** et **S** de la couleur recherchée), puis choisir ensuite dans le rectangle le niveau de la composante **L** de la couleur choisie.
2. L'opérateur peut aussi procéder en sens inverse : choisir d'abord une luminance **L** dans le rectangle de choix, puis choisir dans le carré de choix les composantes **T** et **S**.



I.1. Codage informatique des couleurs

Sélection des couleurs :

Habituellement, l'opérateur ne parvient pas du premier coup à choisir la *bonne* couleur et, quel que soit le sélecteur dont il dispose (Microsoft ou PhotoShop ou autre), il utilise donc en général successivement et en alternance, un clic dans le rectangle et un clic dans le carré. Quant aux valeurs numériques (**TSL** ou **RVB**), elles sont mises à jour comme il convient à l'occasion de chacun des clics de choix. L'opérateur peut aussi forcer telle ou telle valeur de ces composantes en les saisissant directement.



I.1. Codage informatique des couleurs

Exercice :

Soit une Image composée de six zones (a,b,c,d,e f) dont les composantes **RVB** sont données comme suit :

1. Reconstituer les couleurs de l'image d'origine.
2. Donner ses composantes en modèle **CMJ**.

Rouge			Vert			Bleu		
A : Rouge	B : noir	C : rouge	A : noir	B : Vert	C : vert	A : noir	B : Bleu	C : Bleu
D : Noir	E : Rouge	F : noir	D : noir	E : Vert	F : noir	D : noir	E : noir	F : Bleu

I.1. Codage informatique des couleurs

Solution : reconstitution de l'image

Rouge

A : Rouge	B : noir	C : rouge
D : Noir	E : Rouge	F : noir

Vert

A : noir	B : Vert	C : vert
D : noir	E : Vert	F : noir

Bleu

A : noir	B : Bleu	C : Bleu
D : noir	E : noir	F : Bleu

A : Rouge	B : Cyan	C : Blanc
D : noir	E : Jaune	F : Bleu

I.1. Codage informatique des couleurs

Solution : représentation en *CMJ*

A : Rouge	B : Cyan	C : Blanc
D : noir	E : Jaune	F : Bleu

A :	B	C :
D :	E :	F

A :	B :	C :
D :	E :	F :

A :	B :	C :
D :	E :	F :

I.2.Affichage et acquisition des couleurs : Introduction aux Ecrans couleurs et aux équipements de capture.

Email : Abdelli@hotmail.com

Site Web :

<https://sites.google.com/site/abdelkrimabdelli/teaching>

I.2.1 Introduction aux écrans d'ordinateur

On appelle **écran** (ou *moniteur*) le périphérique d'affichage de l'ordinateur. On distingue habituellement deux familles d'écrans :

- Les **écrans à tube cathodique** (notés **CRT** pour *Cathod Ray Tube*), équipant anciennement la majorité des ordinateurs de bureau. Il s'agit de moniteurs volumineux et lourds, possédant une consommation électrique élevée.
- Les **écrans plats** équipant la totalité des ordinateurs portables, les assistants personnels (**PDA**), les appareils photo numérique, ainsi qu'un nombre de plus en plus grand d'ordinateurs de bureau. Il s'agit d'écrans peu encombrants en profondeur (d'où leur nom), légers et possédant une faible consommation électrique.

I.2.1. Introduction aux écrans d'ordinateur

Caractéristiques techniques

Les moniteurs sont souvent caractérisés par les données suivantes :

1. *La définition* : L'écran est décomposée en un nombre de points lumineux. Chaque point affiche une *couleur*. Un point de couleur est obtenu par composition de trois faisceaux lumineux *Rouge vert et bleu* émanant de trois *luminophores* reconstituant le codage RVB du *pixel à afficher*.

La *définition native* est le nombre matériel de *points lumineux* que l'écran peut afficher, ce nombre de points est généralement compris entre *640×480* (*640 points en longueur, 480 points en largeur*) et *2048×1536*, mais des résolutions supérieures sont techniquement possibles.

I.2.1. Introduction aux écrans d'ordinateur

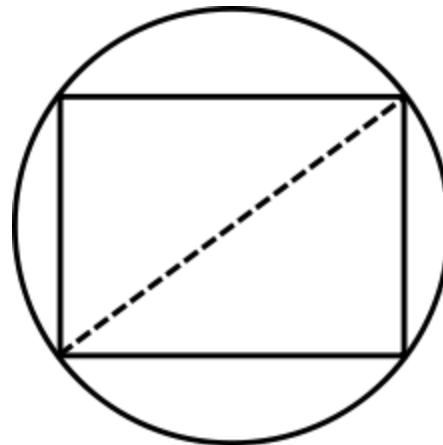
La définition : L'image affichée par l'écran peut avoir une définition inférieure à sa *définition native*. Dans ce cas la couleur de chaque *pixel* de l'image affichée par l'écran peut être rendu par plusieurs point lumineux. Cas du zoom par exemple, ou de la modification de la *définition logique de l'écran*.

Le processus d'adaptation de la *définition de l'image* à afficher par rapport à la *définition native* de l'écran est réalisée par la carte graphique, et appelée « *Scalage* » : *Associer à chaque pixel de l'image à afficher les points lumineux matériels de l'écran qui sont associés au rendu de sa couleur.*

I.2.1. Introduction aux écrans d'ordinateur

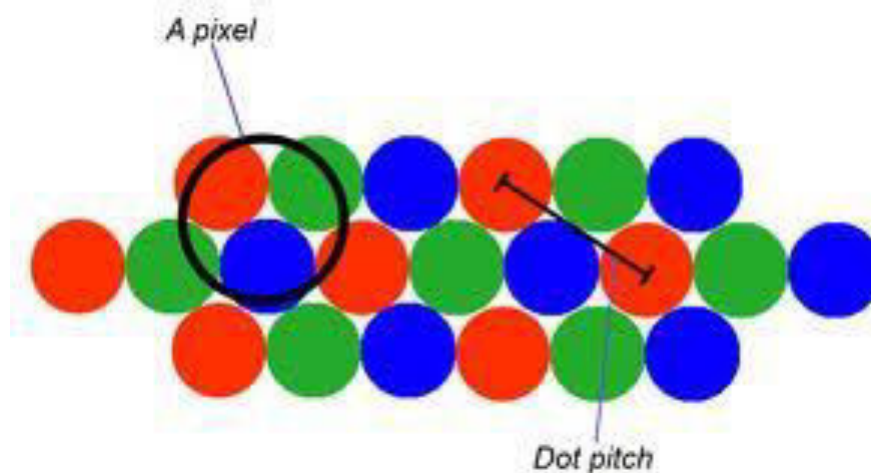
1. **La taille** : Elle se calcule en mesurant la diagonale de l'écran et est exprimée *en pouces* (un pouce équivaut à 2,54 cm). Il faut veiller à ne pas confondre la *définition* de l'écran et sa *taille*. En effet un écran d'une taille donnée peut afficher différentes définitions, :

- **14 pouces**, soit une diagonale de **36 cm** environ ;
- **15 pouces**, soit une diagonale de **38 cm** environ ;
- **17 pouces**, soit une diagonale de **43 cm** environ ;
- **19 pouces**, soit une diagonale de **48 cm** environ ;
- **21 pouces**, soit une diagonale de **53 cm** environ ;



I.2.1 Introduction aux écrans d'ordinateur

3. *Le pas de masque* (en anglais *dot pitch*): C'est la distance qui sépare deux *luminophores* de même couleur ; plus celle-ci est petite plus l'image est précise. Ainsi un pas de masque inférieur ou égal à *0,25 mm* procurera un bon confort d'utilisation, tandis que les écrans possédant des pas de masque supérieurs ou égaux à *0,28 mm* seront à proscrire.



I.2.1 Introduction aux écrans d'ordinateur

3. *La résolution*: Elle détermine le nombre de points lumineux par unité de surface (*pixels par pouce linéaire* (en anglais **DPI: Dots Per Inch**, traduisez *points par pouce*). Une résolution de **300 dpi** signifie **300 colonnes** et **300 rangées** de points sur un **pouce carré** ce qui donnerait donc **90 000** points sur un **pouce carré**. La résolution de référence de **72 dpi** nous donne un point mesurant **1/72** (un pouce divisé par 72) soit **0.353mm**, correspondant à un **Point Pica** (*unité typographique anglo-saxonne*). Les termes « **résolution** » et « **définition** » sont souvent confondus à tort.

I.2.1 Introduction aux écrans d'ordinateur

5. *La fréquence de balayage vertical* (*refresh rate* en anglais) : Elle représente le *nombre d'images* qui sont affichées par seconde, ou plus exactement le nombre de rafraîchissement de l'image par seconde. On l'appelle ainsi également *taux de rafraîchissement*, elle est exprimée en *Hertz*. Plus cette valeur est élevée meilleur est le confort visuel (on ne voit pas l'image scintiller), il faut donc qu'elle soit bien supérieure à *67 Hz* (limite inférieure à partir de laquelle l'oeil voit véritablement l'image "*clignoter*"). La plupart des personnes ne perçoivent plus de scintillement (en anglais *flicker*) à partir de *70 Hz*, ainsi une valeur supérieure ou égale à *75 Hz* conviendra généralement.
6. *La luminance* : Exprimée en *candelas* par mètre carré (*Cd/m²*), elle permet de définir la « *luminosité* » de l'écran. L'ordre de grandeur de la luminance est d'environ *250 cd/m²*.

I.2.1 Introduction aux écrans d'ordinateur

7. *Le temps de réponse* : Défini par la norme internationale ISO 13406-2, il correspond à *la durée nécessaire* afin de faire passer un point du blanc au noir, puis de nouveau au blanc (*éteindre puis allumer le point*). *Le temps de réponse* (défini en millisecondes), doit être le plus petit possible (pragmatiquement, inférieur à *25 ms*).

Rapport avec le taux de rafraichissement :
 $1000 \text{ ms} / 25 = 40 \text{ images à la seconde} = 40 \text{ hertz}$

7. *L'angle de vision vertical et horizontal* : Exprimée en degrés, il permet de définir l'angle à partir duquel la vision devient difficile lorsque l'on n'est plus face à l'écran.
8. *La consommation électrique* : est mesurée en watts.

I.2.1 Introduction aux écrans d'ordinateur

Les modes graphiques : On appelle *mode graphique* le mode d'affichage des informations à l'écran, en terme de définition et de nombre de couleurs. Il représente ainsi la capacité d'une carte graphique à gérer des détails ou celle d'un écran.

- **MDA** La mode **MDA** (*Monochrome Display Adapter*), apparu en 1981, est le mode d'affichage des écrans monochromes fournissant un affichage en mode texte de *80 colonnes par 25 lignes*. Ce mode permettait d'afficher uniquement des caractères ASCII.
- **CGA** Le mode **CGA** (*color graphic adapter*) est apparu en 1981 peu après le mode *MDA* avec l'arrivée du PC (*personal computer*). Ce mode graphique permettait : un affichage en mode texte amélioré, permettant d'afficher les caractères avec *4 couleurs* avec une résolution de *320 pixels par 200* (notée 320×200).

I.2.1 Introduction aux écrans d'ordinateur

- **EGA** Le mode **EGA** (*Enhanced Graphic Adapter*) est apparu au début des années 1985. Il permettait d'afficher **16 couleurs** avec une résolution de **640 par 350** pixels, soit des graphismes beaucoup plus fins qu'en mode **CGA**.
- **VGA** Le mode **VGA** (*Video graphics Array*) a vu le jour en 1987. Il permet une résolution de **720×400** en mode texte et une résolution de **640 par 480** en mode graphique **16 couleurs**. Il permet également d'afficher **256** couleurs avec une définition de **320×200**. Le VGA est rapidement devenu le mode d'affichage minimum pour les ordinateurs de type PC.
- **XGA** En 1990, IBM a introduit le **XGA** (*eXtended Graphics Array*). La version 2 de ce mode d'affichage, baptisé *XGA-2* offrait une résolution de **800×600** en 16 millions de couleurs et de **1024×768** en **65536** couleurs.

I.2.1. Introduction aux écrans d'ordinateur

- **SVGA** Le mode **SVGA** (*Super Video Graphics Array*) est un mode graphique permettant d'afficher 256 couleurs à des résolutions de **640×200**, **640×350** et **640×480**. Le SVGA permet également d'afficher des définitions supérieures telles que le **800×600** ou le **1024×768** en affichant moins de couleurs.
- **SXGA** Le standard **SXGA** (*Super eXtended Graphics Array*) défini par le consortium VESA fait référence à une résolution de **1280×1024** en **16 millions** de couleurs. Ce mode a pour caractéristique d'avoir un rapport de **5/4** contrairement aux autres modes (VGA, SVGA, XGA, UXGA) qui utilisent le **4/3**.
- **UXGA** Le mode **UXGA** (*Ultra eXtended Graphics Array*) définit une résolution de **1600×1200** avec **16 millions de couleurs**.

I.2.1. Introduction aux écrans d'ordinateur

- **WSXGA+** Le mode **WSXGA+** (*Wide Super eXtended Graphics Array+*) définit une résolution de **1680×1050** avec 16 millions de couleurs.
- **WUXGA** Le mode **WUXGA** (*Wide Ultra eXtended Graphics Array*) définit une résolution de **1920 × 1200** avec 16 millions de couleurs.
- **QXGA** Le mode **QXGA** (*Quad eXtended Graphics Array*) définit une résolution de **2048×1536** avec 16 millions de couleurs.
- **QSXGA** Le mode **QSXGA** (*Quad Super eXtended Graphics Array*) définit une résolution de **2560 × 2048** avec 16 millions de couleurs.
- **QUXGA** Le mode **QUXGA** (*Quad Ultra eXtended Graphics Array*) définit une résolution de **3200×2400** avec 16 millions de couleurs.

1.2.1. Introduction aux écrans d'ordinateur

Le tableau suivant récapitule les différentes résolutions, ainsi que le ratio correspondant :

Format d'affichage	Résolution horizontale	Résolution verticale	Nombre de pixels	Ratio
VGA	640	480	307 200	1
SVGA	800	600	480 000	1,56
XGA	1024	768	786 432	2,56
SXGA	1280	1024	1 310 720	4,27
SXGA+	1400	1050	1 470 000	4,78
SXGA+	1280	1024	1 310 720	4,27
UXGA	1600	1200	1 920 00	6,25
QXGA	2048	1536	3 145 728	10,2
QSXGA	2560	2048	<u>5,242,800</u>	17,1
QUXGA	3200	2400	<u>7,680,000</u>	25

I.2.1. Introduction aux écrans d'ordinateur

Par ailleurs, en 1993, un consortium de fabricants de matériel informatique (*VESA*, *Video Electronics Standards Association*) créa la norme *DPMS* (*Display Power Management Signalling*) proposant 4 modes de fonctionnement pour les appareils s'y conformant :

1. En marche.
 2. En veille (standby), avec une consommation inférieure à 25W.
 3. En suspension, avec une consommation inférieure à 8W. Dans ce mode le canon à électrons est éteint, ce qui implique un délai de remise en route plus grand qu'en veille.
- En arrêt.

I.2.2. EXERCICES

Exercice 1: L'écran d'un portable est un écran XGA (1024×768) de 14,1'. Quelle est la taille de l'écran ($L \times H$) en cm ? Quel est son *dot pitch* (*distance entre deux points* de même couleur de l'écran) ? Quelle est la résolution de l'écran ?

Solution : 14,1' représente la diagonale de l'écran. Comme **1 pouce = 2.54 cm** environ, la diagonale de l'écran mesure **35,81 cm**. Il faut ensuite écrire le théorème de Pythagore : En notant **L**, la largeur de l'écran, la hauteur **H** est égale **768×L / 1024** soit encore $\frac{3}{4} L$.

Le théorème s'écrit ici : $L^2 + (3L/4)^2 = (14,1)^2$

d'où **L=11,28 pouces** pour la largeur et donc **8,46 pouces** pour la hauteur.

La taille de l'écran est donc : **28.65cm × 21.48cm**.

Comme l'écran accepte d'afficher **1024** par **768 points** (ou plutôt, triplets de photophores **RVB**),

Le dotpitch vaut **11.28 / 1024** (ou **8.46/768**) = **0.011'** (soit environ 0.28 mm) et la résolution : **1/0.011 = 91 dpi**. environ.

I.2.2. EXERCICES

Exercice 2 : Soit un écran cathodique XGA dont les dimensions Largeur \times Hauteur sont égales à **42.98 cm \times 32.22cm**

- Donner la taille de l'écran.
- Si la valeur du Dot-Pich est égale à **0,42 mm**, quelle serait la définition de cet écran.
- Exprimer la résolution de cet écran en DPI.
- Donner le poids de l'image affichée lorsqu'elle est encodée en RVB.

Solution : Hauteur \times Largeur sont égales à **32.22cm \times 42.98 cm.**

taille de l'écran : On applique la formule de pythagore :

$$\text{Taille}^2 = (32,22)^2 + (42,88)^2 \Rightarrow 1038,13 + 1838,6944 \Rightarrow \text{taille} = 53.65 \text{ cm} = \mathbf{21 \text{ pouces.}}$$

Dot-Pich = 0,42 mm, donc

$$\text{la définition} = (42.98/0,042) \times (32.22/0,042) = \mathbf{768 \times 1024.}$$

$$\text{résolution de cet écran en DPI} = 2,54 / 0,042 = \mathbf{61.}$$

$$\text{Taille Image} = \text{Définition} \times \text{Profondeur} = (768 \times 1024) \times 3 \text{ octets} = \mathbf{2\,359\,296 \text{ octets.}}$$

I.2.2. EXERCICES

Exercice 3: Soit un écran de taille 21 pouces, trouvez sa hauteur et sa largeur (en cm) si on utilise le format matériel 4/3. Quelle est la définition de cet écran, si une image écran en RVB occupe une taille de **1 440 000** octets. Donnez la résolution de cet écran.

Solution :

Taille = 21 pouces Largeur = 4/3 Hauteur

$$\begin{aligned} 21 \times 21 &= 16/9 (\text{hauteur})^2 + (\text{hauteur})^2 \Rightarrow \text{hauteur} \\ &= 12,6 \text{ pouces} = \mathbf{32 \text{ cm}} \text{ et Largeur} = \mathbf{42,67 \text{ cm}.} \end{aligned}$$

$$\begin{aligned} \text{Définition} \times \text{Profondeur} &= \mathbf{1\ 440\ 000} \text{ octets} \Rightarrow \text{Définition} = \mathbf{1\ 440\ 000 / 3} \\ &= \mathbf{48\ 000} = H \times (H \times 4/3) = \mathbf{600 \times 800}. \end{aligned}$$

$$\text{La résolution} = 600 / \text{Hauteur} = 800 / \text{Largeur} = \mathbf{46,61 \text{ DPI}.}$$

I.2.Affichage et acquisition des couleurs : Introduction aux Ecrans couleurs et aux équipements de capture.

Email : Abdelli@hotmail.com

Site Web :

<https://sites.google.com/site/abdelkrimabdelli/teaching>

I.2.3. Le moniteur à tube Cathodique

Un moniteur **CRT** est un tube en verre sous vide dans lequel un *canon à électrons* émet un flux d'électrons dirigés par un champ électrique vers un écran couvert de petits éléments phosphorescents. Le canon à électrons est constitué d'une **cathode**, c'est-à-dire une électrode métallique chargée négativement, d'une ou plusieurs **anodes** (électrodes chargées positivement).

Un champ magnétique est chargé de dévier les électrons de gauche à droite et de bas en haut. Il est créé grâce à deux bobines **X** et **Y** sous tension (appelées **défecteurs**) servant respectivement à dévier le flux horizontalement et verticalement.

I.2.3. Le moniteur à tube Cathodique

L'écran est recouvert d'une fine couche d'éléments phosphorescents, appelés *luminophores*, émettant de la lumière par excitation lorsque les électrons viennent les heurter, ce qui constitue *un point lumineux*.

Un *moniteur noir et blanc* permet d'afficher des dégradés de couleur (niveaux de gris) en variant l'intensité du rayon.

Pour les moniteurs couleur, trois faisceaux d'électrons (correspondant à *trois cathodes*) viennent chacun heurter un point d'une couleur spécifique : *un rouge, un vert et un bleu*.

Les luminophores *bleus* sont réalisés à base de *sulfure de zinc*, les *verts* en *sulfure de zinc et de cadmium*.

I.2.3. Le moniteur à tube Cathodique

- Les *rouges* enfin sont plus difficile à réaliser, et sont faits à partir d'un mélange *d'yttrium et europium*, ou bien d'*oxyde de gadolinium*. Cependant ces luminophores sont si proches les uns des autres que l'oeil n'a pas un pouvoir séparateur assez fort pour les distinguer: il voit une couleur composée de ces trois couleurs.

De plus, pour éviter des phénomènes de bavure (un électron destiné à frapper un luminophore *vert* percutant le *bleu*) une grille métallique appelée *masque* est placée devant la couche de luminophores afin de guider les flux d'électrons.

I.2.3. Le moniteur à tube Cathodique

1. Le moniteur à tube cathodique

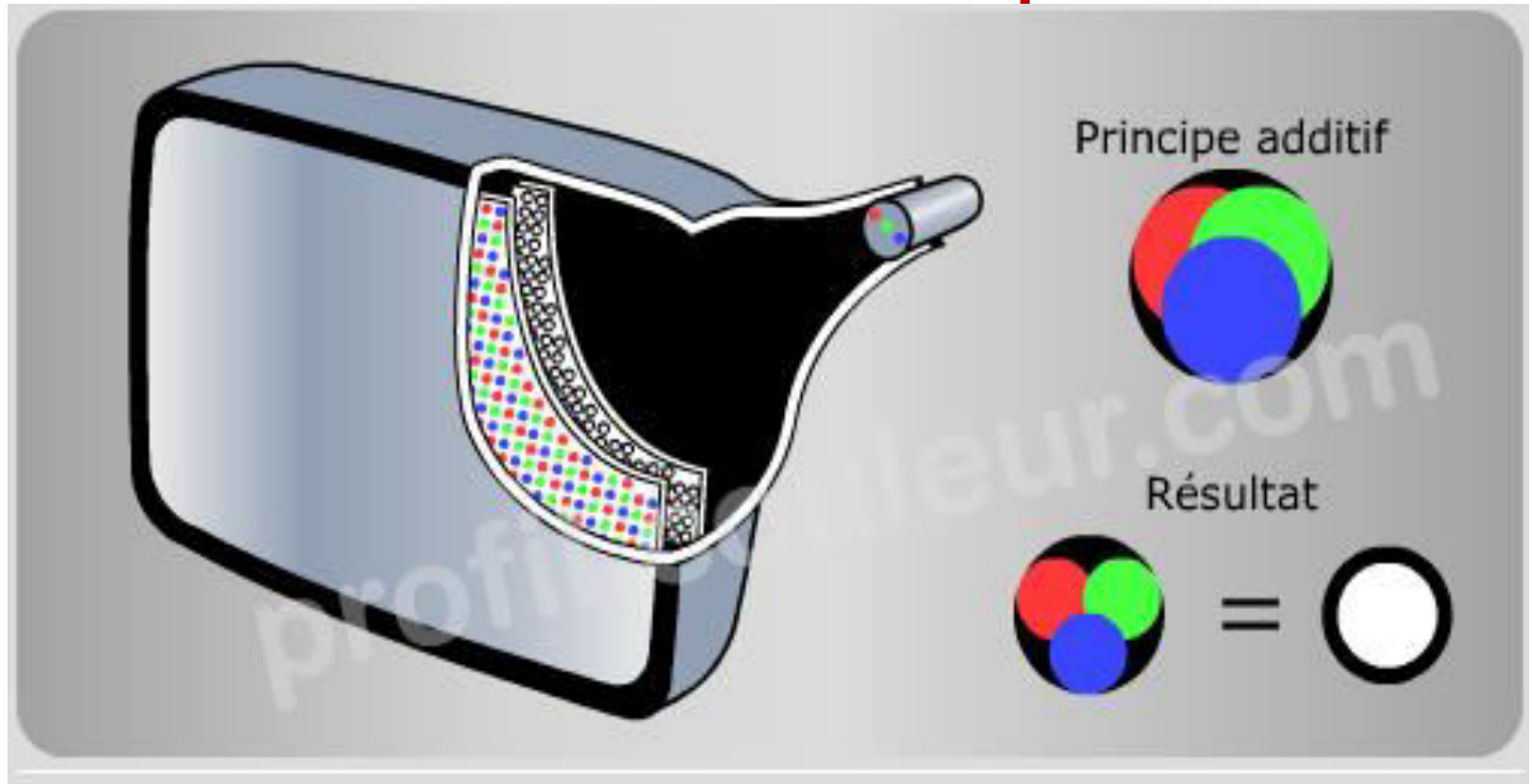
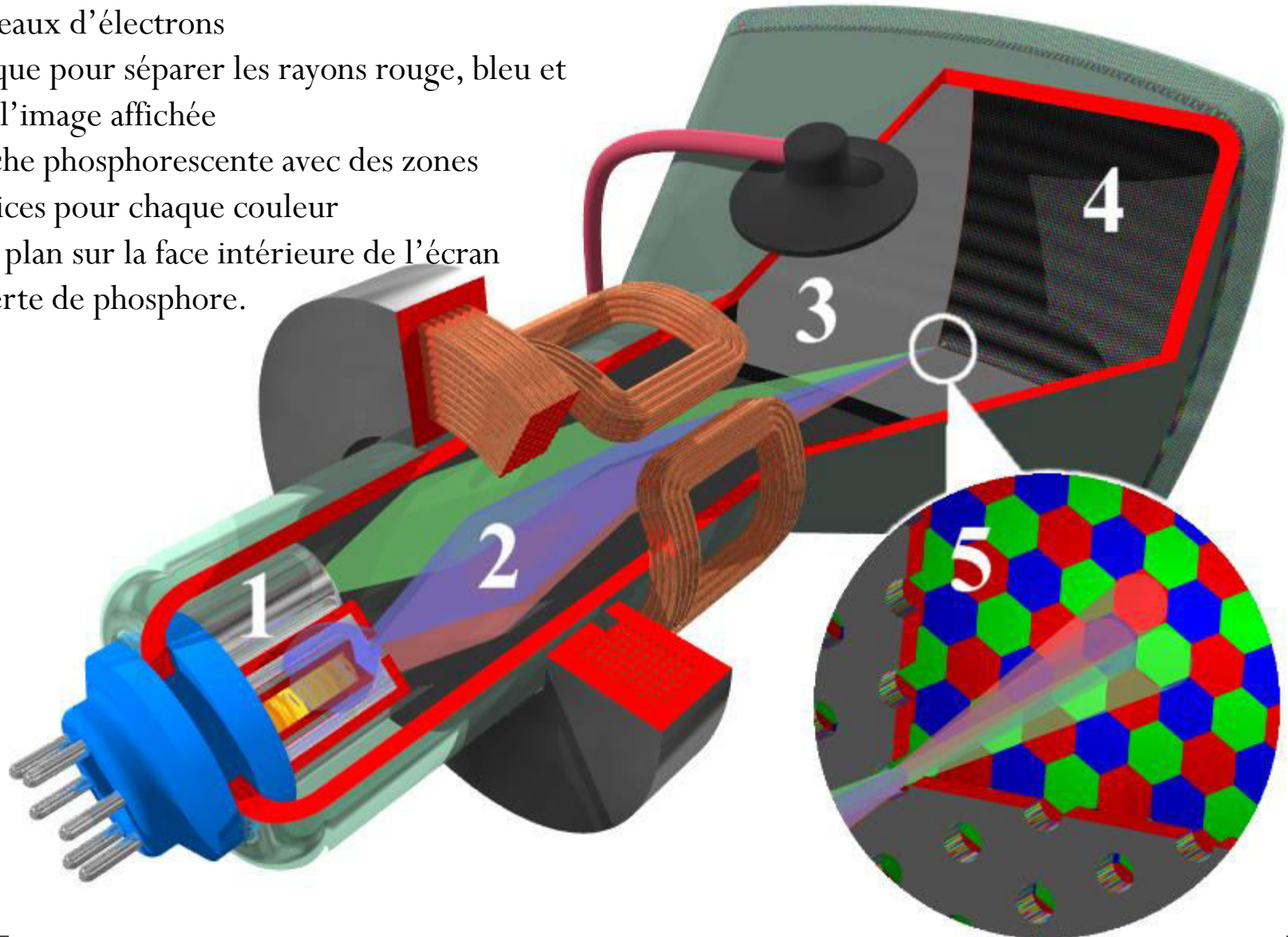


Fig. 2. Les anciens moniteurs CRT utilisent le même principe RVB. Ici, ce sont des faisceaux d'électrons qui viennent bombarder la face intérieure de l'écran qui est tapissée d'une mosaïque de pastilles phosphorescentes rouge, vert et bleu.

I.2.3. Le moniteur à tube Cathodique

- 1 : canons à électrons
- 2 : faisceaux d'électrons
- 3 : masque pour séparer les rayons rouge, bleu et vert de l'image affichée
- 4 : couche phosphorescente avec des zones réceptrices pour chaque couleur
- 5 : gros plan sur la face intérieure de l'écran recouverte de phosphore.



I.2.3. Le moniteur à tube Cathodique

[VIDEO](#)

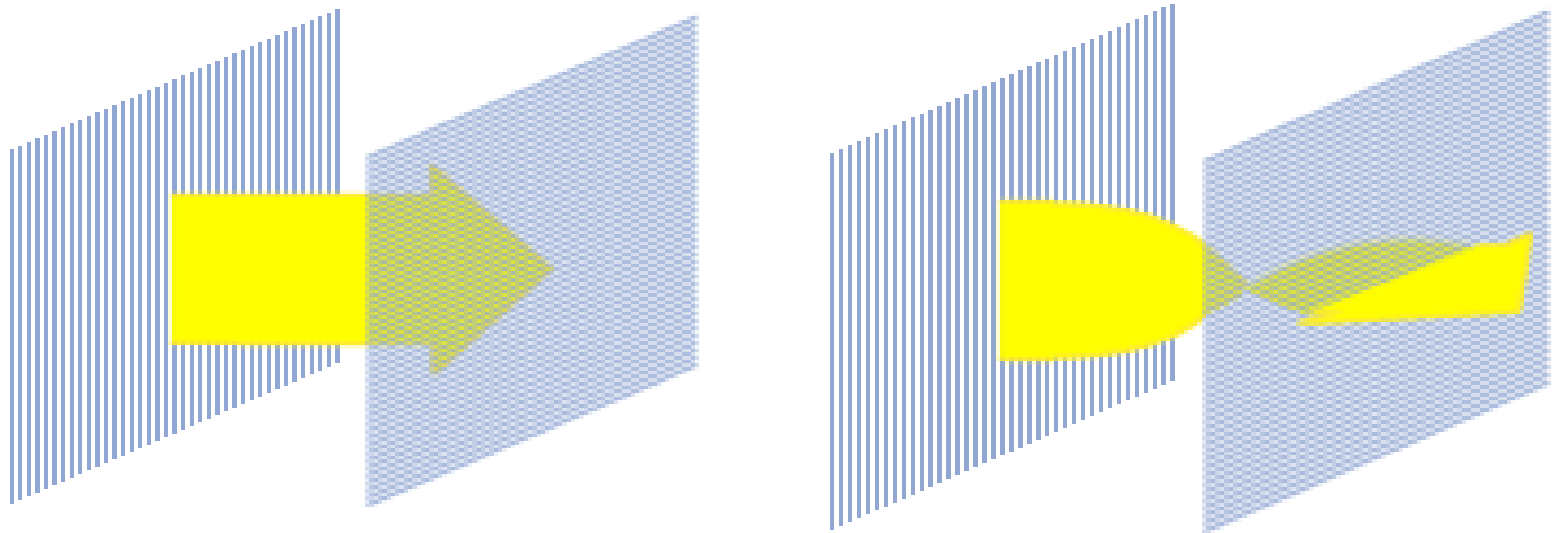
I.2.4. Le moniteur à écran plat

Les **moniteurs à écran plat** (notés parfois **FPD** pour *Flat panel display*) se généralisent de plus en plus dans la mesure où leur facteur d'encombrement et leur poids sont très inférieurs à ceux des écrans **CRT** traditionnels. De plus, les technologies utilisées dans les écrans plats sont moins consommatrices d'énergie (consommation inférieure à 10W contre 100W pour les écrans CRT) et n'émettent pas de rayonnement électromagnétique.



I.2.4. Le moniteur à écran plat

La technologie **LCD** : La technologie **LCD** (*Liquid Crystal Display*) est basée sur un écran composé de deux plaques parallèles rainurées transparentes, orientées à 90° , entre lesquelles est coincée une fine couche de liquide contenant des molécules (**cristaux liquides**) qui ont la propriété de s'orienter lorsqu'elles sont soumises à du courant électrique.



I.2.4. Le moniteur à écran plat

La technologie LCD :

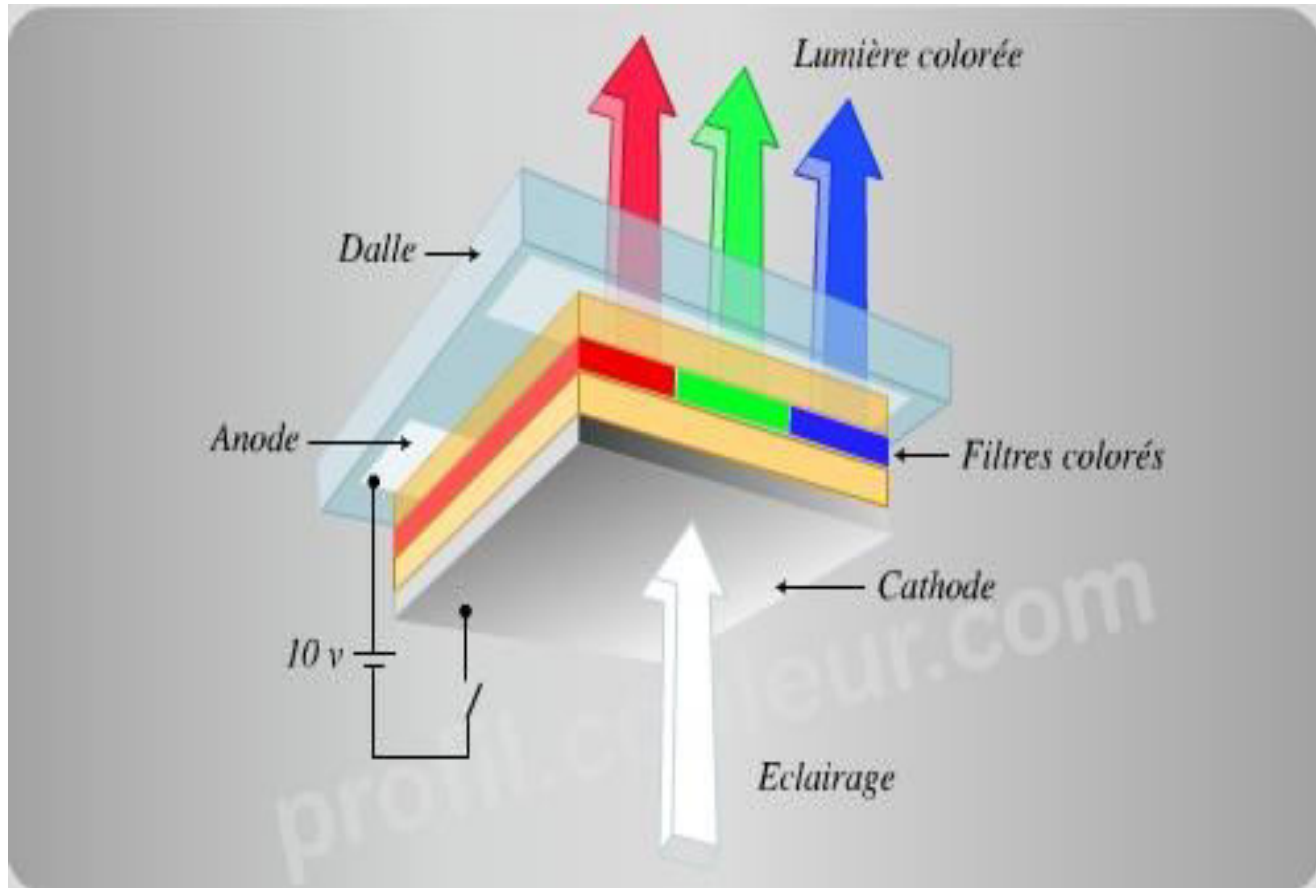
La dalle est éclairée par l'arrière avec une lumière polarisée et lorsque les sillons changent d'orientation, la lumière passe au travers. Sous l'effet d'une tension de commande, les cristaux vont progressivement s'orienter dans le sens du champ électrique et la lumière sera bloquée par le deuxième polariseur. Chaque pixel de l'image est constitué d'une cellule de ce type devant laquelle est placé un filtre rouge, vert, ou bleu.

<https://www.youtube.com/watch?v=1iL7AbZvrmU>

<https://www.youtube.com/watch?v=Ldf-WPekVyK>

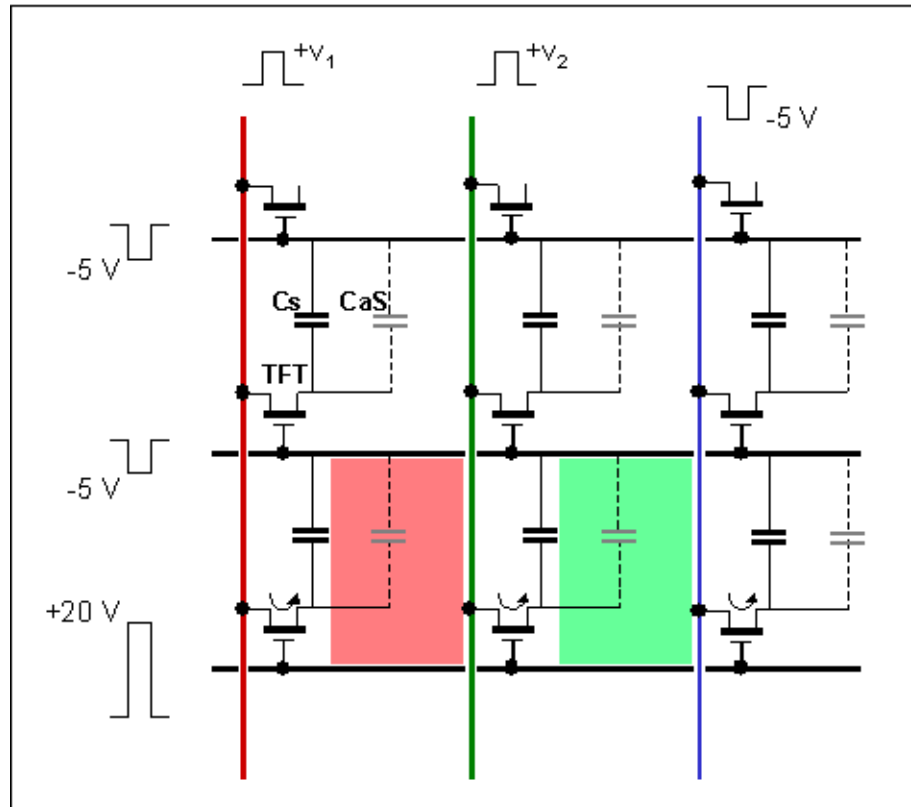
I.2.4. Le moniteur à écran plat

La technologie LCD :



I.2.4. Le moniteur à écran plat

- Schéma électrique équivalent ; le pixel inférieur est affiché jaune.



I.2.4. Le moniteur à écran plat

2- La technologie PLASMA :

La technologie **plasma** (**PDP**, *Plasma Display Panel*) est basée sur une émission de lumière grâce à l'excitation d'un gaz. Le gaz utilisé dans les écrans plasma est un mélange d'**Argon** (90%) et de **Xénon** (10%).

Ce mélange de gaz est inerte et inoffensif. Pour qu'il émette de la lumière on lui applique un courant électrique qui le transforme en **plasma**, un fluide ionisé dont les atomes qui ont perdu un ou plusieurs de leurs électrons et ne sont plus électriquement neutres, alors que les électrons ainsi libérés forment un nuage autour. Le gaz est contenu dans **des cellules**, correspondant aux luminophores.

I.2.4. Le moniteur à écran plat

2- La technologie PLASMA :

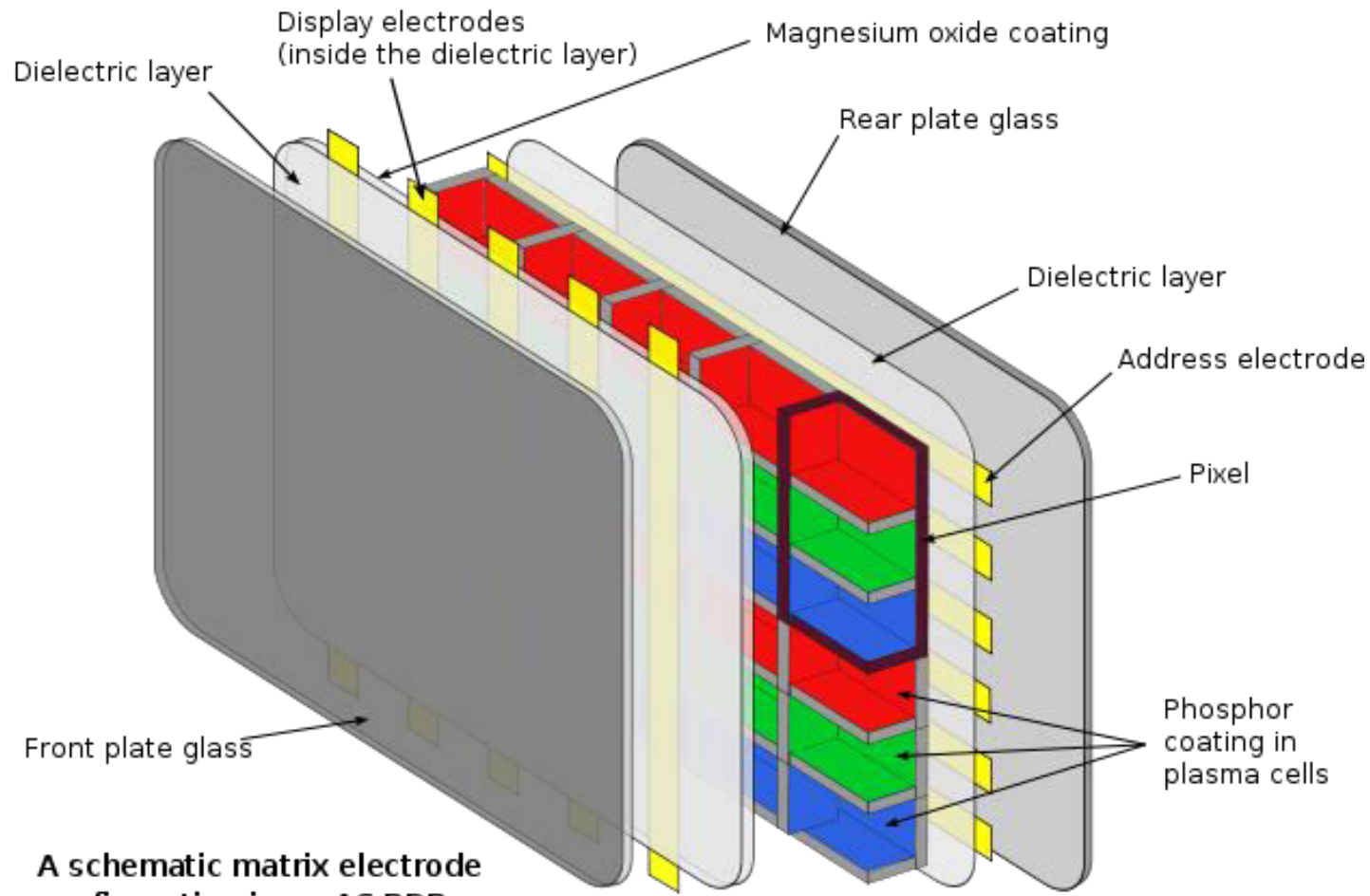
- Chaque cellule est adressée par une *électrode ligne* et une *électrode colonne* ; en modulant la tension appliquée entre les électrodes et la fréquence de l'excitation, il est possible de définir l'intensité lumineuse (en pratique on utilise jusqu'à **256 valeurs**).

Pour comprendre fonctionnement des écran plasma :

- <https://www.youtube.com/watch?v=210oGQY7Vt4>

I.2.4. Le moniteur à écran plat

2- La technologie PLASMA :



A schematic matrix electrode configuration in an AC PDP

I.2.4. Le moniteur à écran plat

2 -La technologie PLASMA :

Le gaz ainsi excité produit un rayonnement lumineux ultraviolet (donc invisible pour l'œil humain. Grâce à des luminophores respectivement *bleus*, *verts* et *rouges* répartis sur les cellules le rayonnement lumineux ultraviolet est converti en lumière visible, ce qui permet d'obtenir des pixels (composés de 3 cellules) de 16 millions de couleurs ($256 \times 256 \times 256$).

La technologie plasma permet d'obtenir des écrans de grande dimension avec de très bonnes valeurs de contrastes mais le prix d'un écran plasma reste élevé. De plus la consommation électrique est plus de **30 fois** supérieure à celle d'un écran LCD.

I.2.4. Le moniteur à écran plat

3 -La technologie LED :

Le **LED** (*Light Emitting Diode*), est un composant électronique, couramment connu sous la dénomination de *diode électroluminescente*. La principale fonction du LED reste l'éclairage.

Les téléviseurs LCD et LED partagent une grande partie de leurs composants. Le principe de fonctionnement basé sur le cristal liquide reste dans les deux cas identique. Les écrans plats LCD classiques fonctionnent avec un **système de rétro-éclairage** composé de **néons** (généralement **4 tubes** situés à l'arrière de la dalle LCD). Cette technologie a pour inconvénient une répartition parfois peu homogène de l'éclairage à l'arrière de la dalle LCD. Les écrans dits « LED » utilisent une technologie de rétro-éclairage complètement différente. Les **tubes néons** (CCFL) sont remplacés par plusieurs centaines de LEDs, disposées à l'arrière de la dalle LCD.

I.2.4. Le moniteur à écran plat

3 -La technologie LED :

- La technologie LCD ne produit pas de vrais noirs, mais seulement des gris très foncés (puisque les tubes sont allumés en permanence). Plus grave, les néons ne permettent pas aux écrans LCD de reproduire la totalité de l'espace colorimétrique (le Gamut). Ce qui se traduit généralement par des images aux couleurs plus ternes. En outre, ces tubes consomment une quantité substantielle d'énergie électrique limitant d'autant l'autonomie des ordinateurs portables et augmentant d'autant la dissipation thermique. Mais plus gênant encore, leur durée de vie est par nature limitée (de 30 000 à 50 000 heures), ce qui conditionne celle des écrans LCD.

I.2.4. Le moniteur à écran plat

3 -La technologie LED :

- Autant de faiblesses dont ne souffre pas la technologie de rétroéclairage par Led. Et pour cause : il s'agit cette fois d'utiliser, à la place des tubes, une grande quantité de minuscules diodes électroluminescentes, de trois couleurs (une rouge, une verte et une bleue, afin de créer un blanc composite) ou tout simplement blanches, espacées de quelques millimètres et réparties à l'arrière de la dalle à cristaux liquides, sur toute sa superficie. Il est dès lors possible de n'éclairer que certaines diodes pour illuminer une zone précise, d'où un contraste plus marqué.
- La matrice de LEDs composée de petites sources lumineuses permet de limiter l'effet de contagion de la lumière sur les pixels avoisinant. Et cette qualité peut encore être amplifiée numériquement par un procédé qui détecte les zones sombres de l'image et diminue l'intensité du rétroéclairage dans ces zones. On augmente ainsi notablement la profondeur du noir.

I.2.4. Le moniteur à écran plat

3 -La technologie LED :

LCD vs LED :

1. La luminance est aussi bien supérieure : elle atteint 500 cd/m^2 avec les Led de dernière Génération contre environ 350 cd/m^2 en technologie LCD. Mais plus que tout, le recours aux diodes offre aux écrans LCD qui les exploitent une prise en charge de l'intégralité de l'espace colorimétrique, d'où des couleurs plus éclatantes.
2. Plus économes en énergie. - consommation d'énergie jusqu'à 40 % inférieure aux écrans LCD classiques
3. Les diodes consomment également moins d'énergie et se contentent pour fonctionner d'une tension d'alimentation de l'ordre de $3,5 \text{ V}$ contre $400 \text{ à } 500 \text{ V}$ pour les tubes CCFL.
4. Elles offrent également une tolérance accrue au niveau de leur température de fonctionnement, qui s'étend de $-20 \text{ à } +70 \text{ }^\circ\text{C}$, contre $0 \text{ à } +50 \text{ }^\circ\text{C}$ avec les tubes. Et, finalement, leur durée de vie atteint allégrement les $100\,000 \text{ heures}$, soit 34 ans à raison d'une utilisation quotidienne de 8 heures

I.2.4. Le moniteur à écran plat

3 -La technologie LED :

On distingue aujourd'hui deux types de diodes électroluminescentes utilisées dans les écrans :

- **LED RGB** : La première famille de diodes est dite classique puisqu'elle se décline sous les couleurs primaires de l'image: rouge vert bleue,
- **LED Edge** : Les diodes LED de couleurs blanches sont installées sur tout le bord de la dalle LCD. Des **réflecteurs de lumière** permettent de diffuser la lumière des LED sur la totalité de la **dalle de l'écran** LCD. Outre les avantages du rétroéclairage LED, ce système permet de **diminuer l'espace nécessaire** à l'arrière de la dalle (3 cm).

I.2.4. Le moniteur à écran plat

3 -La technologie LED :

Les différentes technologies LED

On distingue également plusieurs types de TV à LED, à savoir les

- **Full LED ou Direct LED** Certains téléviseurs utilisent une répartition des diodes LED de type *Full LED* ou *Direct LED*. Cela signifie que les diodes LED utilisées pour le rétroéclairage tapissent l'arrière de la dalle LCD, couvrant ainsi la totalité de l'écran. Le rétro-éclairage est encore plus **homogène** que le *EDGE LED*, mais l'épaisseur de l'écran est plus importante. Dans certains téléviseurs, le rétroéclairage est assuré par **trois LED** (RVB : rouge, verte, et bleu). Ce système permet un meilleur ajustement de la colorimétrie.

I.2.4. Le moniteur à écran plat

3 -La technologie LED :

Les différentes technologies LED

- **Full LED + Local Dimming** La technologie LED est utilisée de manière optimale quand chaque diode LED peut être allumée et éteinte de manière **intelligente**, c'est à dire uniquement quand la luminosité de la scène ou du détail de la scène le nécessite. Cette technologie d'éclairage **dynamique LED** est appelée "**local dimming**". L'allumage des zones est dynamique et les noirs sont beaucoup plus profonds que sur les écrans LCD classiques. En terme de rendu et de qualité d'image, cette technologie est très proche des écrans plasma.
- **Edge LED + Local Dimming** Ce type d'écran combine les avantages du *local dimming* en termes de **qualité d'image**, et la finesse de l'écran (*diodes LED* présentes sur les bords de la dalle LCD).
- Un écran d'entrée de gamme pourra se contenter de 40 LED alors que le haut de gamme demandera 2000 LED ou plus.

I.2.4. Le moniteur à écran plat

3 -La technologie LED :

<https://www.youtube.com/watch?v=cavRBCJCCLA>



I.2.4. Le moniteur à écran plat

3 -La technologie LED :

- Par rapport aux TV LED Local Dimming, les TV Full LED disposent d'un excellent rapport de contraste mais ont pour principale faiblesse une épaisseur d'écran assez importante. Quant aux TV LED Local Dimming, elles consomment moins d'énergie et disposent d'un contraste plus dynamique ainsi que d'une profondeur des plus fine, mais accusent des halos de lumières indésirables au niveau de certaines images. La technologie LED a permis de rendre l'expression écran plat encore plus significative en raison de la finesse des TV à LED. Elle permet également une réduction considérable de la consommation d'énergie des foyers. Les prix de ses TV écologiques varient généralement selon les dimensions, les fonctionnalités et les technologies utilisées.

I.2.4. Le moniteur à écran plat

Comparaison

CRT

Points forts:

- Une excellente couleur, une large gamme.
- Affichage native dans presque n'importe quelle résolution et fréquence de rafraîchissement.
- Temps de réponse sous le milliseconde.
- **Angle de vision** excellent.

Points faibles:

- Grande taille et poids, surtout pour les grands écrans (de 20" pèse environ 22 kg)
- Consommation d'énergie importante
- Distorsion géométrique dans des tubes cathodiques plats.
- CRT âgées sont sujets au burn-in.
- Scintillement qui se produit à de faibles taux de rafraîchissement

I.2.4. Le moniteur à écran plat

LCD

Points forts:

- Très compact et léger.
- Faible consommation d'énergie.
- Pas de distorsion géométrique
- Robuste.
- Peu ou pas de scintillement en fonction de la technologie de rétro-éclairage

Points faibles:

- Angle de vue, limité provoquant la variation de la saturation, le contraste et la luminosité de la couleur lors des variations des posture.
- Rétro-éclairage inégale dans certains moniteurs, causant une distorsion de luminosité, surtout vers les bords.
- Temps de réponse lent ; LCD modernes ont des temps de réponse de **8ms** ou moins.
- Une seule résolution native. Afficher les autres résolutions nécessite un [scaler vidéo](#), ce qui dégrade la qualité d'image en résolution réduite.

LCD sont incapables de représenter les ***vraies couleurs***.

Pixels morts sont possibles lors de la fabrication

I.2.4. Le moniteur à écran plat

Plasma

Points forts:

- Compact et léger.
- Une excellente couleur.
- Bon Temps de réponse.
- Absence de distorsion de la saturation, du contraste ou de luminosité de la couleur.
- Angle de vision Excellent.
- Très évolutive, avec moins de gain de poids par augmentation de la taille (de moins de 30 pouces à 150 pouces).

Points faibles:

- Grand *terrain de pixel* ce qui signifie soit faible résolution ou un grand écran.
- Scintillement perceptible lorsqu'on visionne de près.
- Haute *température de fonctionnement* et de consommation d'électricité
- A une seule *définition native*. Afficher les autres résolutions nécessite un [scaler vidéo](#), ce qui dégrade la qualité d'image en résolution réduite.

I.2.4. Le moniteur à écran plat

Ecran LED :

Points forts:

- Epaisseur réduite : Les écrans LED autorisent une diminution notable de l'épaisseur de l'écran.
- Les ampoules LED consomment moins que les tubes fluorescents.
- Un meilleur contraste. Grâce à leur petite taille, les sources d'éclairage LED sont moins sensibles à la diffusion.
- La matrice de minuscules ampoules LED garantit une meilleure homogénéité de la luminosité sur la dalle. Si cette amélioration n'est pas primordiale sur un téléviseur, c'est un plus incontestable sur un écran d'ordinateur surtout dans une utilisation graphique professionnelle.
- Capacité à reproduire des couleurs très saturées : Le gamut ou espace colorimétrique restreint des écrans LCD est élargie grâce à la technologie LED.

Points faibles:

- Angle de vue, limité
- Temps de réponse moyen ;
- Une seule résolution native.

I.2.5. Introduction aux Capteurs photographiques

- Un *capteur photographique* est un composant électronique *photosensible* servant à convertir un *rayonnement électromagnétique* en un *signal électrique analogique*. Ce signal peut ensuite être amplifié, puis numérisé par un convertisseur analogique-numérique et enfin traité pour obtenir une image numérique. Le *capteur* est donc le composant de base des appareils photo numériques, l'équivalent du film en *photographie argentique*.
- Le capteur photographique met à profit l'effet photoélectrique, qui permet aux *photons* incidents d'arracher des *électrons* à chaque élément actif (*photosite*) d'une matrice de capteurs élémentaires constitués de *photodiodes*.

I.2.5. Introduction aux Capteurs photographiques

- Deux grandes familles de capteurs sont disponibles : les **CCD** et les **CMOS**. Les **CCD** sont surtout utilisés dans les appareils *compacts* et de plus en plus délaissés dans les reflex. Les appareils *reflex* quant à eux, utilisent majoritairement des capteurs **CMOS**.



1.2.5. Introduction aux Capteurs photographiques

1. Un capteur CCD.

Le **CCD** (*Charge-Coupled Device*, ou *dispositif à transfert de charge*) est le plus simple à fabriquer, a une bonne sensibilité, mais, du fait de son principe, le *transfert de charge est relativement lent*. Inventé par George E. Smith et Willard Boyle dans les Laboratoires Bell en 1969 (cette invention leur rapportera le Prix Nobel de physique en 2009), il a rapidement été adopté pour des applications de pointe (imagerie astronomique) puis popularisé sur les caméras et appareils photo.



1.2.5. Introduction aux Capteurs photographiques

L'organisation topologique des *photosites* permet de définir deux types de capteurs d'images **CCD** :

1. les capteurs d'images **CCD linéaires** ;
2. les capteurs d'images **CCD matriciels**.

Les *capteurs linéaires* sont notamment utilisées pour le contrôle de qualité des produits plats en défilement continu, de produits de grandes dimensions ou de produits possédant un axe de révolution. Ce type de caméra permet d'obtenir une *représentation bidimensionnelle* de la surface des produits à inspecter grâce à leurs défilements. La seconde dimension de l'image peut-être obtenue soit en déplaçant le produit à inspecter devant la caméra, soit en déplaçant la caméra devant le produit comme c'est le cas pour les images issues des *scanners ou des photocopieuses*.

1.2.5. Introduction aux Capteurs photographiques

- Le principe de base de l'acquisition consiste en la conversion des rayons lumineux réfléchis par la scène à observer en un signal vidéo à l'aide *d'un capteur de forme linéaire* parfois désigné sous le terme de *barrette* dans la littérature. La disposition linéaire des photo-capteurs ne permet que l'observation d'une ligne de la scène restituée sous la forme d'une *image-ligne* contrairement aux *caméras matricielles* qui permettent l'acquisition d'images matricielles. La figure suivante montre un exemple de ce type de capteur et met en évidence sa forme linéaire.



I.2.5. Introduction aux Capteurs photographiques

Architecture d'un capteur *CCD* :

Un *CCD* transforme les photons lumineux qu'il reçoit, en **électrons** maintenus à chaque *photosite*.

À l'aide de ces circuits, l'image est acquise en deux temps :

- **un temps d'intégration** ;
- **un temps de transfert**.

Durant le *temps d'intégration*, la lumière incidente d'énergie est absorbée par les *photosites* puis transformée en paquets de charges électriques proportionnelles à l'intensité lumineuse incidente. Les paquets de charge sont alors transférés vers les *registres analogiques* via une porte de transfert et décalés en série vers la sortie du *CCD*.

1.2.5. Introduction aux Capteurs photographiques

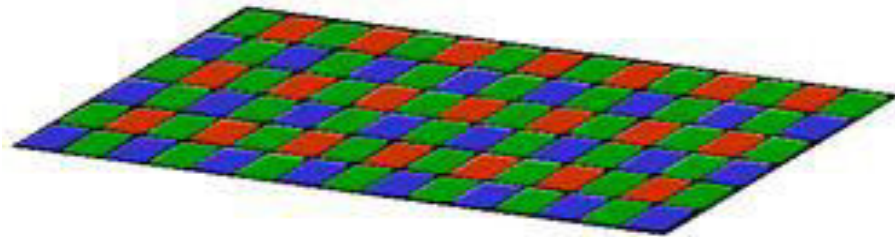
Technologies utilisées pour l'obtention de la Couleur :

Les caméras couleur et l'oeil humain se basent sur le principe de la théorie *trichrome* des récepteurs visuels. Ce principe met en évidence que n'importe quelle couleur peut être reproduite par un mélange ou *synthèse additive* de *rouge*, de *vert* et de *bleu* appelées *couleurs primaires*. Les capteurs étant sensibles sur un domaine *plus grand que le domaine du visible*, il suffit de placer un filtre coloré devant le *photo-récepteur* pour que celui-ci ne soit sensible que sur un intervalle de longueurs d'onde déterminées. Aussi, pour que l'image acquise soit proche de l'image que l'oeil perçoit, les filtres utilisés sont de couleur *rouge*, *vert* et *bleu*. *L'acquisition d'images numériques couleur utilise donc trois capteurs de sensibilités différentes pour reconstituer la couleur d'un stimulus par synthèse additive.*

1.2.5. Introduction aux Capteurs photographiques

Filtre de Bayer *RGB* : Pour ce faire, on utilise le filtre, dit *de Bayer*, constitué de cellules colorées des couleurs primaires, chaque *photosite* du capteur ne voit qu'une seule couleur : **rouge**, **vert** ou **bleu**. Sur chaque groupe de quatre photo-sites on trouve un pour le **bleu**, un pour le **rouge** et **deux** pour le **vert** ; cette répartition correspond à la sensibilité de notre vision.

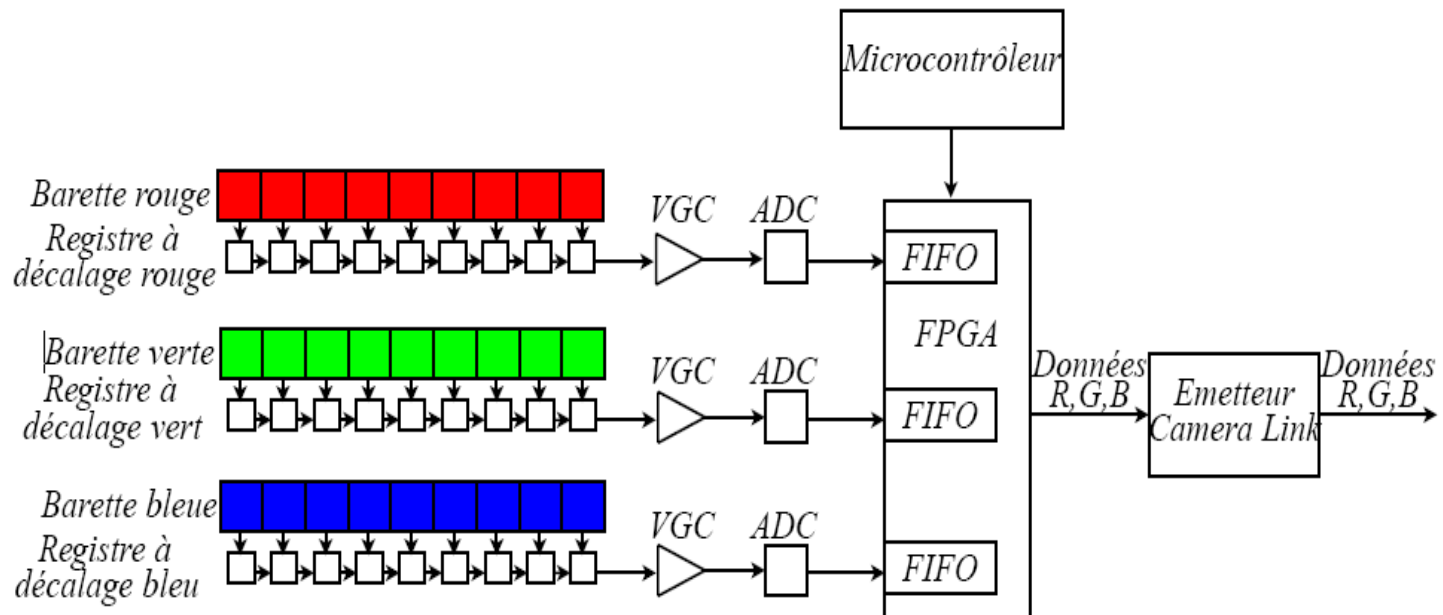
- C'est le logiciel du *photoscope* qui va recréer les couleurs, en tenant compte des courbes de réponse spectrale pour un résultat final en *trichromie* ; un des problèmes est de limiter le bruit électronique qui se traduit par des effets de *moiré* sur les zones de faible lumière par de judicieux compromis lors du traitement d'image.



I.2.5. Introduction aux Capteurs photographiques

Architecture d'un capteur linéaire :

Les niveaux de tension des *photo-récepteurs* transmis aux registres à décalage peuvent être amplifiés par un **VGC** (*Variable Gain Control*) et sont convertis en valeurs numériques grâce à un dispositif **ADC** (*Analogic to Digital Converter*).



I.2.5. Introduction aux Capteurs photographiques

La *numérisation* transforme la tension en une valeur codée sur *8 bits* que nous appellerons *niveau d'une composante*. Les niveaux des composantes proviennent d'un des capteurs couleur *rouge R*, *vert V* ou *bleu B*. Les niveaux des composantes *R*, *V* et *B* sont ensuite stockées dans des *pires FIFO* (First In First Out) différentes pour chaque capteur couleur. En fonction des paramètres de la caméra, un *micro-contrôleur* permet de sélectionner les valeurs appropriées des composantes *R*, *V* et *B* dans les *pires FIFO* suivant leur ordre d'arrivée dans les *pires*. Lorsque les informations des composantes *R*, *V* et *B* sont disponibles dans les *pires FIFO*, celles-ci sont envoyées à la carte d'acquisition. Cette information envoyée à la carte correspond aux niveaux de couleur *R*, *V* et *B* d'un point image aussi appelé *pixel* d'une image-ligne.

1.2.5. Introduction aux Capteurs photographiques

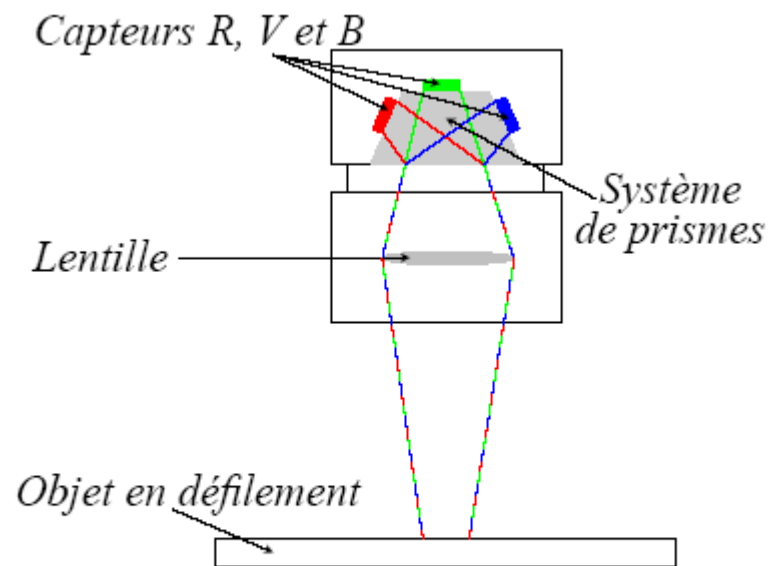
2. Les capteurs CMOS

Les capteurs **CMOS** (*Complementary metal oxide semi-conductor*) sont apparus dans les années 1980, à la suite des **matrices de photodiodes**, comme résultat de l'intégration de cellules composées d'une **photodiode** et d'une logique d'amplification puis d'obturation. Ils sont plus complexes à fabriquer mais sont produits selon des techniques classiques de micro-électronique et de ce fait peuvent avoir des dimensions importantes (jusqu'à **24 mégapixels**). Ils sont généralement utilisés pour les **capteurs des appareils reflex**.

De la même façon que beaucoup de **CCD**, les capteurs **CMOS** pour image couleur sont associés à un filtre coloré et un réseau de lentilles, encore plus nécessaire vu la faible surface relative de la photodiode, seule zone sensible.

I.2.5. Introduction aux Capteurs photographiques

- Une des méthode consiste à utiliser un système de prismes optiques permettant de projeter une même image sur trois capteurs optiques différents recouverts par des filtres colorés. Grâce à ce système de prismes, un même élément de la scène observé est projeté sur le même photo-élément des trois capteurs R , V et B comme indiqué sur la figure.



(a) Ligne de visée d'une caméra linéaire à prisme.

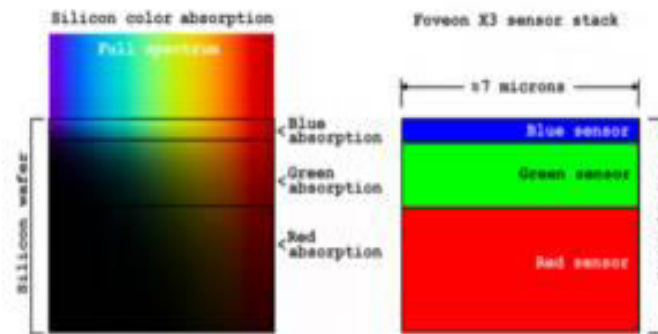
1.2.5. Introduction aux Capteurs photographiques

Les capteurs CMOS Foveon

Le capteur X3 a été développé par la société américaine Foveon, rachetée en 2008 par Sigma, qui bénéficie depuis d'un droit d'exploitation exclusif. Contrairement à un *photosite* de capteur **CCD** qui capture seulement une couleur primaire (rouge, vert ou bleu), un photo-site de capteur X3 recueille une composante RVB. Ceci nécessite donc beaucoup moins d'électronique de calcul, puisque la couleur est directement obtenue sur le *photosite* et plus après traitement électronique des couleurs de quatre photosites. C'est un avantage en termes de coût de fabrication, mais aussi en termes de qualité. En effet, l'absence de calculs et d'interpolations permet d'espérer des images plus « propres », et permettrait aussi un rythme de prises de vues plus rapide en mode rafale.

1.2.5. Introduction aux Capteurs photographiques

Les capteurs Foveon



Le capteur X3 met à profit le fait que les grandes longueurs d'onde de la lumière pénètrent plus profondément dans le silicium. Ce capteur permet la capture des trois couleurs rouge, vert et bleu par un seul *photosite*, au moyen de *trois couches de silicium* recouvertes de photosites et disposées en sandwich et filtrées chacune par un *filtre bleu, vert ou rouge* ; Chacune des couches de photo-récepteurs est précisément espacée relativement aux longueurs d'onde bleue, verte et rouge de la lumière visible. Pour simplifier nous pourrions dire qu'en recevant un rayon incident, la couche superficielle du silicium arrête le bleu, que la couche médiane arrête le vert et enfin que le rouge est stoppé par la couche inférieure.

1.2.5. Introduction aux Capteurs photographiques

Les performances des capteurs

La résolution maximale d'un capteur est fonction du nombre de photo-sites qui permettra d'obtenir autant de pixels grâce à une *interpolation* astucieuse. Le **CMOS** est plus fiable que le **CCD**. L'efficacité quantique du capteur est définie par le rapport électrons produits/photons incidents (ce qui est un point commun avec le principe de base de la photographie argentique). Elle est surtout fonction de la taille de la partie active de chaque photo-site (c'est-à-dire la surface de capture des photons).

Chapitre II : : L'imagerie numérique

Email : Abdelli@hotmail.com

Site Web :

<https://sites.google.com/site/abdelkrimabdelli/teaching>

II.1. Représentation d'une Image

Définition

Une image est une surface 2D, dont chaque point de la surface absorbe la lumière selon une longueur d'onde particulière

Image Numérique

Une image numérique est une image disposant un nombre fini de points codant l'information lumineuse dans une gamme finie de longueurs d'onde

- Point: **pixel** ou picture element
- Définition: dimension de l'image en pixels ($h \times v$)
- Résolution: nombre de pixels sur une unité de surface: (DPI, PPC)
- Profondeur: nombre de valeurs possibles de la gamme.

Types

- Noir et blanc: 1 bit par pixel.
- Gris: 8 à 12 bits par pixel.
- Couleurs: 24 à 36 bits par pixel.

II.1. Représentation d'une Image

Transparence de l'image

Possibilité de voir à travers l'image des éléments graphiques situés derrière celle-ci. Elle peut se définir de 2 façons différentes :

1. Dans le cas d'une *image indexée* : une couleur est définie comme transparente dans la palette de couleurs
2. **Alpha-blending** : à chaque pixel de l'image est ajouté un octet définissant son niveau de transparence (de *0 à 255*).

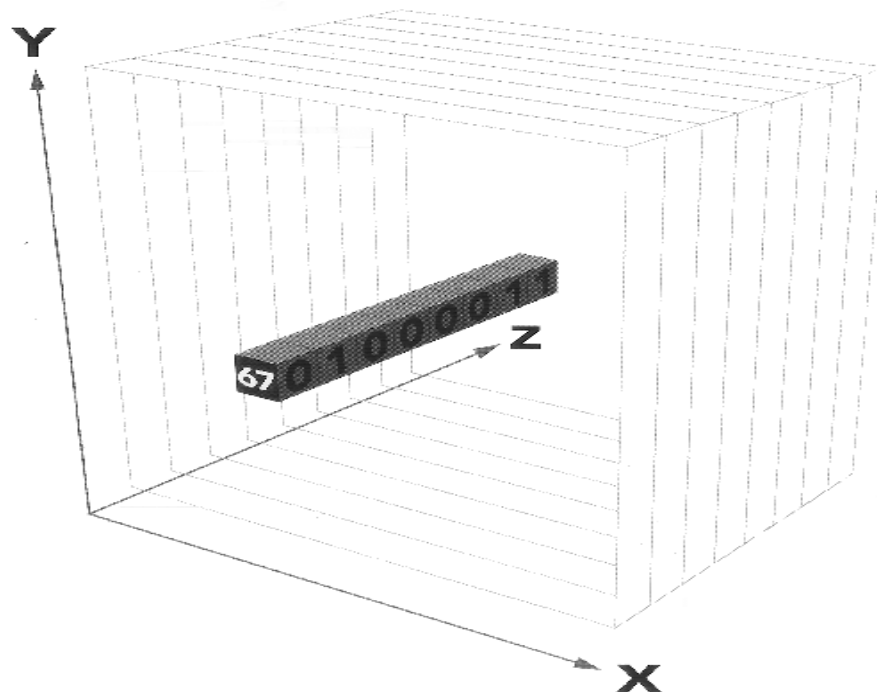
II.1. Représentation d'une Image

Une image se compose d'une grille de **pixels** (*picture elements*), tous de même taille, ayant chacun sa propre couleur. La superficie de l'image est donc sa **définition**. Le nombre de couleurs différentes contenues dans le fichier d'une image est déterminé par sa **profondeur** (résolution tonale), soit le nombre de bits alloués par pixel. Plus on alloue de bits par pixel et plus on a de couleurs (ou teintes de gris) dans l'image.

Bits par pixel	Nombre max de couleurs	Type d'image
1	2	Monochrome
4	16	Image 16 couleurs
8	256	Image standard VGA.
16	65'536	
24	16'777'216	Image RVB

II.1. Représentation d'une Image

- Si **X** et **Y** sont les coordonnées de l'écran (ou de l'image) on peut imaginer l'axe **Z** correspondant au nombre de bits alloués à chaque pixel. Dans l'exemple suivant, le pixel mis en valeur possède la valeur binaire **01000011** correspondant à la couleur **67**. Il est donc codé sur **8 bits** et peut prendre les valeurs **0** à **255**.



II.1. Représentation d'une Image

Résolution	Définition	Pixels	Taille (24 bits)
Basse	640 × 480	307 Kp	922 Ko
	800 × 600	480 Kp	1,4 Mo
Moyenne	1280 × 1024	1,3 Mp	3,9 Mo
	1600 × 1200	1,9 Mp	5,8 Mo
	2048 × 1536	3,1 Mp	9,4 Mo
Haute	3264 × 2448	8,0 Mp	24,0 Mo
	4368 × 2912	12,7 Mp	38,1 Mo

100 photos 8 Mp avec profondeur 32 bits: **3.2 Go**

Taille

La taille d'un fichier image non compressé est donnée par:

$$T = D \times P$$

- D : définition de l'image
- T : profondeur de l'image

II.1. Représentation d'une Image

Remarque : Il s'agit de faire la différence entre ce qu'on voit à l'écran, *l'image affichée*, et le *fichier de l'image*. L'image peut ainsi être décrite dans le fichier avec une *définition* en millions de couleurs tout en étant affichée à l'écran en **256** couleurs ou moins. Tout dépend de la machine (la mémoire de la carte graphique, sa carte graphique, son écran), de la façon dont on l'utilise (sa configuration) et du logiciel utilisé (certains peuvent afficher correctement en **256 couleurs** une image de milliers de couleurs à l'aide de la technique du *dithering*).

II.1. Représentation d'une Image

Les modes d'affichages et les cartes graphiques: La *définition* et le nombre de couleurs dépendent essentiellement de la quantité de mémoire vidéo (**VRAM**) présente sur la carte d'affichage. Celles-ci sont généralement équipées de **1 Mo** à **128Mo**. Une quantité inférieure à **1 Mo** est insuffisante. Avec **1 Mo** on devra se contenter de **640×480** en **16 millions** de couleurs, **800×600** et **1024×468** en **256** couleurs. **2 Mo** permettront d'afficher en **16 millions** de couleurs en **800×600** et **1280×1024** en **256** couleurs.

Résolution & mémoire	512 Ko	1 Mo	2 Mo	4 Mo	8 Mo
640 X 480	8 bits	24 bits	24 bits	24 bits	24 bits
800 X 600	8 bits	16 bits	24 bits	24 bits	24 bits
1024 X 768		8 bits	16 bits	24 bits	24 bits
1280 X 1024			8 bits	24 bits	24 bits
1600 X 1200			8 bits	16 bits	24 bits

II.2. Les Types d'Images

Il existe deux méthodes fondamentalement différentes de représentation et de codage des images : la **méthode *Bitmap*** et la **méthode *vectorielle***. Un fichier graphique peut appartenir à l'une ou l'autre de ces méthodes, ou être une combinaison des deux.

1. **BiTmap** : il lui correspond des logiciels de type *paint* (le programme **Paint** des accessoires de Windows, **Mac Paint**, **Painter**, **PSP4**, **Photoshop**, etc).
2. **Vectorielle** : il lui correspond des logiciels de type *Draw* (**MacDraw Pro**, **Illustrator**, **Corel Draw**, etc).

D'autres logiciels permettent de travailler dans les deux modes, tels **Canevas** (Deneba) et **SuperPaint** (Aldus). Enfin, les logiciels vectoriels offrent souvent la possibilité de convertir, de "retracer", les images bitmap en images vectorielles (**Corel Trace** par exemple) et inversement de transformer une image vectorielle en une image bitmap. Le menu Bitmap de **Corel 7** permet cette dernière opération.

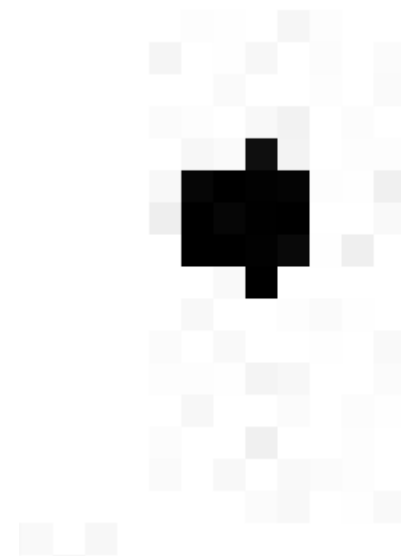
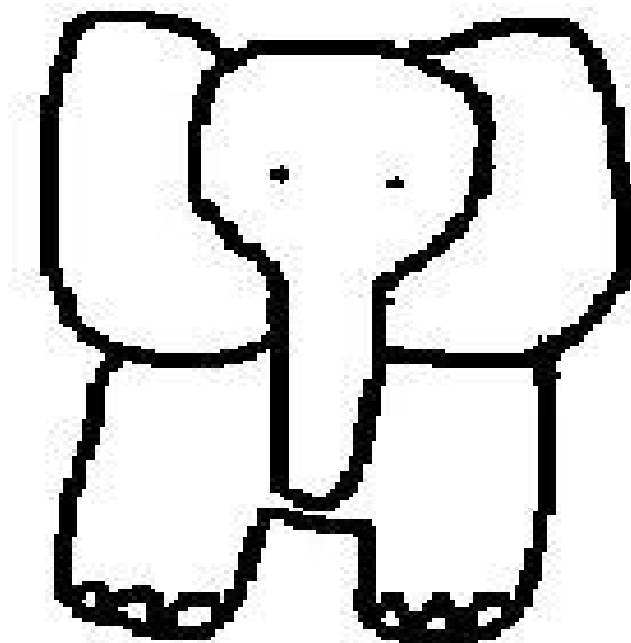
II.2. Les Types d'Images

Les images de type *bitmap*

Les images affichées sur un écran sont des images matricielles, *appelées Bitmap*. Une image bitmap est composée en mode point appelé **pixel** et codé dans la mémoire vidéo de l'ordinateur par leurs coordonnées spatiales et leur couleur.

Il s'agit donc d'une représentation graphique définie par l'ensemble des points qui la composent. Dans le cas d'une photographie, l'opération de *digitalisation*, *d'acquisition* ou encore de *numérisation*, consiste à transformer des tons et des couleurs continus - en nombre infini- en un nombre fini de points élémentaires.

II.2. Les Types d'Images



Zoom de l'oeil

II.2. Les Types d'Images

Avantages images *bitmap*

- Le mode de codage des images bitmap (24 bits, codage RVB) les rend adaptées au fonctionnement des principaux périphériques, notamment les contrôleurs d'écran "**true colors**".
- Elles conviennent fort bien aux images complexes, principalement d'origine analogique.
- Elles se laissent manipuler et traiter par des opérations techniques "**naturelles**" pour un graphiste qui retrouve des outils et les manipulations très proches de ceux qui caractérisent son métier et sa pratique professionnelle de type analogique.

II.2. Les Types d'Images

Avantages images *bitmap*

- Elles permettent des opérations comme l'*anti-aliasing* (suppression du crénelage produit à la rencontre de deux couleurs éloignées), le rehaussement des contours, les modifications locales de l'image (contraste, colorimétrie, effets, filtres, etc.).
- Le mode de codage point par point étant relativement universel, une fois cette opération effectuée, le transcodage demande des calculs répétitifs mais relativement simples : la compatibilité est aisée entre les différents formats.
- La structure du fichier est telle que des dégradations minimales des données, mais non pas de l'entête, ne le rendent pas nécessairement inutilisable.

II.2. Les Types d'Images

Inconvénients des images *bitmap*

- Les images bitmap ont une **résolution fixe** : aussi la qualité maximale sur un périphérique d'affichage ou d'impression donné rend-elle nécessaire de travailler, dans la majorité des cas, dans la résolution native de ce périphérique. Concrètement cela veut dire qu'une résolution de type écran donnera d'assez mauvais résultats sur un imageur photographique. Les images bitmap sont donc dépendantes du périphérique.
- Elles supportent mal les opérations de redimensionnement, réduction ou agrandissement. Après une *réduction* de taille, l'image réduite présentera souvent des effets d'escaliers plus marqués que ceux de l'image source avec perte d'information.
- Les images bitmap sont "*lourdes*": les fichiers, lorsque l'on traite des images en haute définition, ont des tailles qui varient entre 10 et 30 Mo par image.... Elles sont donc encombrantes, difficiles à faire passer sur le réseau, etc.

I.2. Les Types d'Images

Les images de type *vectoriel*

La *description vectorielle* d'une image consiste en une description géométrique: le document numérisé prend donc la forme d'une suite de formules mathématiques décrivant les formes élémentaires constituant l'image (carrés, rectangles, ellipses, cercles, courbes, etc.). Chaque forme élémentaire constitue un objet et se voit assigné un certain nombre d'attributs tels que la couleur, la transparence, l'épaisseur du trait, le type de trait (pointillé, etc.)

II.2. Les Types d'Images

Les images de type *vectoriel*

Le stockage d'une image de type vectoriel est donc très différent de celui d'une image de type bitmap: il consiste en la mémorisation de la représentation des coordonnées des points caractéristiques des formes qui constituent l'image. Il s'agit dès lors d'une représentation relative que l'on peut qualifier de "*symbolique*".

Caractéristiques du mode Vectoriel : Petite taille des fichiers. On ne parle plus de pixels mais *d'objet*. Permet de manipuler indépendamment les objets dessinés. Respect des proportions des objets. Très gourmand en ressources machine (calcul). Un dessin vectoriel exploite au maximum les capacités de l'imprimante.

II.2. Les Types d'Images

Avantages des Images vectorielles

- L'image numérique doit être calculée avant de pouvoir être affichée par le périphérique (opération qui porte le nom de **rastérisation**). Cette opération peut être faite pour n'importe quelle résolution du périphérique: *l'image vectorielle est réellement indépendante du périphérique*.
- Toutes les modifications spatiales de l'image (réduction, agrandissement, translation, rotation, etc.) sont aisées et n'occasionnent aucune perte d'information. Il suffit en effet de modifier les coordonnées des points de contrôle qui définissent l'objet.
- L'image vectorielle est particulièrement adaptée aux représentations schématiques et stylisés constituées de formes géométriques, uniformément remplies par des à-plats de couleur ou des motifs (Ex: **Logos**).
- Un fichier vectoriel est bien *plus compact* qu'un fichier en mode point. Sa taille varie selon la complexité de l'image, mais pas en fonction de la résolution demandée. Un fichier de **2 Mo** peut représenter une image déjà fort complexe.

II.2. Les Types d'Images

Inconvénients des images vectorielles

- Une image vectorielle ne peut coder une image analogique telle qu'une image photographique.
- Le travail sur des objets graphiques isolés qu'il faut ensuite associer ou grouper est peu familier au graphiste ou à l'illustrateur. Il faut donc une certaine habitude et un apprentissage de nouvelles procédures de création.
- Certaines manipulations telles que les modifications de couleurs sont difficiles sur une zone d'un objet, sur un objet simple ou sur un groupe d'objets.
- Un fichier vectoriel est plus fragile qu'un fichier bitmap. La moindre dégradation de l'information est souvent irréparable.
- Chaque format de fichier vectoriel possédant ses propres attributs, la compatibilité entre les formats est difficile.

II.3. Les formats de fichiers images

- **BMP (BitMaP)** : Le format **BMP** est le format par défaut du logiciel Windows. C'est un format matriciel. Les images sont souvent non compressées. L'image peut être en *monochrome* (1 bit par pixel), en **16** couleurs (4 bits/pixel), en **256** couleurs (8 bits/pixel) ou en **16** millions de couleurs (8 bits x 3 couleurs = 24 bits/pixel). *Pour les 3 premiers cas, le fichier comporte une table des couleurs de l'image et ensuite chaque pixel est représenté par le numéro de la couleur dans cette table (index).* Pour le cas **16** millions de couleurs, chaque pixel est directement représenté par 3 octets (24 bits) correspondants aux couleurs fondamentales RVB. **Ex** : Une copie d'écran est au format BMP. Une compression peut être utilisée, elle est de type RLE (non destructive). Le domaine de prédilection du format BMP est la bureautique sous Windows.

Avantages : qualité, et rapidité sous Windows.

Inconvénients : inconnu hors Windows, compression peu efficace.

II.3. Les formats de fichiers images

- **PICT** : Le format **PICT** est un format vectoriel interne au fonctionnement du Macintosh. C'est le format utilisé par le Presse-Papier du Macintosh. Il peut contenir des éléments graphiques ou des images numérisées. Ce format a quelques problèmes au niveau de la gestion de la séparation des couleurs, étape nécessaire à l'impression couleur. Pour des illustrations en noir et blanc ou en tons de gris, ce format peut satisfaire la majorité des besoins. Cependant, il est à proscrire dans le cas des impressions couleurs. Ce format permet un gain appréciable en espace disque et est reconnu par un grand nombre de logiciels autant sur environnement Macintosh que sur Windows. À ce titre, c'est le format de travail utilisé par Word pour Macintosh et son utilitaire **MSGraph**.

II.3. Les formats de fichiers images

- **PS (PostScript)** : Avec la majorité des applications d'aujourd'hui, autant les logiciels de mise en pages, de traitement de textes et autres, il est possible d'exporter un document en format **PS (PostScript)** lequel pourra être acheminé vers un périphérique d'impression. Ce format est également une façon sûre de rendre disponible un document seulement *pour impression sans droit de modification*. Il s'agit toutefois d'un format très lourd à éviter lorsqu'il doit être transféré par Internet sur des liens à basse vitesse.
- **DXF** : Le format **DXF** est un format vectoriel créé par la compagnie **AutoDesk** pour son logiciel de **CAO AUTOCAD**. Bien qu'étant un format très répandu dans le monde de la conception et du dessin assisté par ordinateur, le format **DXF** est très peu répandu en d'autres domaines.
- **PCX** : Le format **PCX** est utilisé par le logiciel **Paintbrush** sous Windows. C'est un format matriciel.

I.3. Les formats de fichiers images

- **EPS (Encapsulated PostScript)** : Un document en format **EPS** vectoriel est un fichier en langage PostScript décrivant le contenu d'une image ou d'une page complète (textes, filets, images importées). Contrairement aux fichiers PostScript (format .PS) qui ne peuvent être destinés qu'à l'impression, le fichier **EPS** peut être visualisé et importé dans bon nombre de logiciels de dessins, de **PAO** et de mise en pages. En fait, chaque fichier **EPS** contient une représentation matricielle en basse résolution en **TIFF** ou en **PICT** pour visualiser l'illustration à l'écran.
- **Le format EPS matriciel** : n'est pas très différent du **EPS** vectoriel. En fait seules les données contenues dans le fichier sont différentes. Ainsi un logiciel de retouche de photos tel que Photoshop permet l'importation, la modification et l'exportation de fichiers en format **EPS**. Il faut user de prudence lors d'une séparation en quadrichromie d'un fichier **EPS**. À noter que le format **EPS** matriciel est sensible à la résolution de l'image.

II.3. Les formats de fichiers images

- **Le format PNG :** (**PNG** : *Portable Network Graphic*) Il s'agit du « 3ème » format Internet, c'est-à-dire le 3ème format reconnu par les navigateurs. Un fichier **PNG** peut contenir des images **RVB**, en niveaux de gris, en **256** couleurs indexées, ou bitmap. Un niveau de transparence peut être enregistré sous la forme d'une **couche alpha**. **PNG** utilise une compression sans perte, et permet un affichage progressif. Ce format est donc un des rares formats, avec le **GIF** et le **JPEG**, à être reconnu par les navigateurs Internet. Il est pourtant moins utilisé car il ne présente pas d'avantages décisifs par rapport aux deux formats précédents, en taille de fichiers.

En revanche, comparé au format **GIF**, pour qui une licence doit être contractée (à cause de l'algorithme **LZW** payant), l'algorithme de compression du **PNG**, proche du **LZ77** est du domaine public.

Les navigateurs ne l'interprètent pas toujours bien, et ne tirent pas encore profit de toutes ses possibilités (couche alpha, gamma, etc.). De plus les différents logiciels d'optimisation d'images pour le web ne l'utilisent pas de la même façon : seul **Fireworks**, dont il est le format natif, semble en tirer pleinement profit et le compresse mieux qu'un **GIF**.

II.3. Les formats de fichiers images

- **TIFF (*Tagged Image File Format*)** : Il s'agit d'un format bitmap créé en 1988 par Microsoft et Aldus (société rachetée par Adobe). Le format **TIFF** est un format matriciel conçu à l'origine pour n'accepter que les images en **RVB**. Le format **TIFF** offre l'avantage d'occuper moins d'espace disque, grâce à son propre algorithme de compression appelé **LZW**. Ainsi un fichier image en format **EPS** occupe 25% moins d'espace après une conversion appropriée en format **TIFF**. Actuellement, c'est le choix à privilégier en **PAO** mais c'est aussi un choix à éviter pour le Web puisqu'aucun navigateur Web ne peut le lire directement.

C'est un format permettant le codage d'images de définition jusqu'à 65536 x 65536 pixels. Un fichier **TIFF** peut contenir des images codées selon les modes colorimétriques suivants : [RVB](#), [CMJN](#), couleurs indexées, niveaux de gris.

II.3. Les formats de fichiers images

- **Photo CD** : Créé par la compagnie KODAK, ce format est utilisé par les laboratoires de photographies afin de transférer des négatifs de photos sous une forme numérique, généralement sur un CD-ROM. Cela vous permet d'utiliser vos propres photos sans recourir à l'utilisation d'un scanner.
- **WPG** : Le format **WPG** est un format utilisé par les logiciels de la gamme de WordPerfect (WordPerfect, DrawPerfect, WP Presentations et autres) sous DOS, Windows ou Macintosh. C'est un format vectoriel qui donne un résultat acceptable lors de l'impression, mais qui doit surtout être utilisé en tant que format de travail. D'autant plus que ce n'est pas un format qui est reconnu par tous les logiciels.

II.3. Les formats de fichiers images

- **GIF (*Graphical Interchange Format*)** : Format de fichier créé en 1987 par la société CompuServe. Le format **GIF** est un format qui a ouvert la voie à l'image sur le **World Wide Web**. C'est un format de compression qui n'accepte que les images en couleurs indexés codé sur 8 bits, soit en **256** couleurs.

Les images en milliers de couleurs doivent d'abord être converties en **256** couleurs avant d'être exportées en format **GIF**. Le format **GIF** ne peut toutefois pas être utilisé par les logiciels de mise en pages et de **PAO**. C'est un format qui perd beaucoup de son marché suite à une bataille juridique concernant les droits d'utilisation sur Internet. Il permet une compression automatique des documents; les fichiers sont alors relativement petits ; la profondeur maximisant encore plus la compression et réduisant la taille des fichiers. Il permet aussi une facilité de décompression et d'affichage sur des ordinateurs de plateformes différentes.

II.3. Les formats de fichiers images

GIF utilise une compression systématique (non destructrice) [LZW](#). C'est un format qui permet un mode de transparence, l'affichage progressif et la possibilité de créer des animations. Le domaine d'utilisation concerne Internet, les logos et graphiques simples

La limitation du format GIF tient dans le passage obligé par l'utilisation d'une palette (*256 couleurs indexées au maximum*). Cette limitation du nombre de couleurs peut être considérée comme une méthode de compression (destructive) pour les images comportant initialement plus de couleurs. Le résultat est visuellement mauvais pour les images complexes (photographies), même si la palette peut être adaptative à l'image et des couleurs intermédiaires peuvent être simulées par la technique du tramage.

La compression de type **LZW** est plus efficace qu'une compression **RLE** mais est sensible au bruit dans les images et est entreprise ligne par ligne: la taille résultante peut varier en fonction de l'orientation d'une image rectangulaire.

II.3. Les formats de fichiers images

- **JPEG (*Joint Photographic Experts Group*)** : Les images **JPEG** sont des images de **24 bits**. C'est-à dire qu'elles peuvent afficher un spectre de 16 millions de couleurs. C'est la meilleure qualité d'images disponible. Par contre, si la carte graphique de votre ordinateur est ajustée à **256** couleurs, les images **JPEG** auront meilleure allure que les images en format "**GIF** ». Il est sans doute le mode compression le plus efficace qui soit, avec un bon compromis entre gain d'espace disque, temps de compression/décompression et qualité des images. Ainsi une image brute de **2 Mo** n'occupera après conversion en **JPEG** que **130 à 400 Ko** selon la qualité d'image voulue.

II.3. Les formats de fichiers images

L'inconvénient d'une image **JPEG**, est qu'elle ne peut être importée directement dans un logiciel de mise en page. Il faut d'abord la convertir vers une image matricielle.

Le format **JPEG** est un des formats les plus utilisés dans le monde du **WWW** et lu par les principaux outils de navigation.

JPEG est le résultat d'une méthode de compression destructive, au taux de compression réglable. Les dimensions maximales sont là aussi 65536 x 65536 pixels.

Un fichier **JPEG** peut contenir une image en modes ***RVB***, ***CJMN***, ou niveaux de gris.

Un fichier **JPEG** de base ne peut pas contenir de *calques* ni de *couches alpha*.

Animation et Compression des Images

Email : Abdelli@hotmail.com

Site Web :

<https://sites.google.com/site/abdelkrimabdelli/teaching>

II.5. Animation d' Images

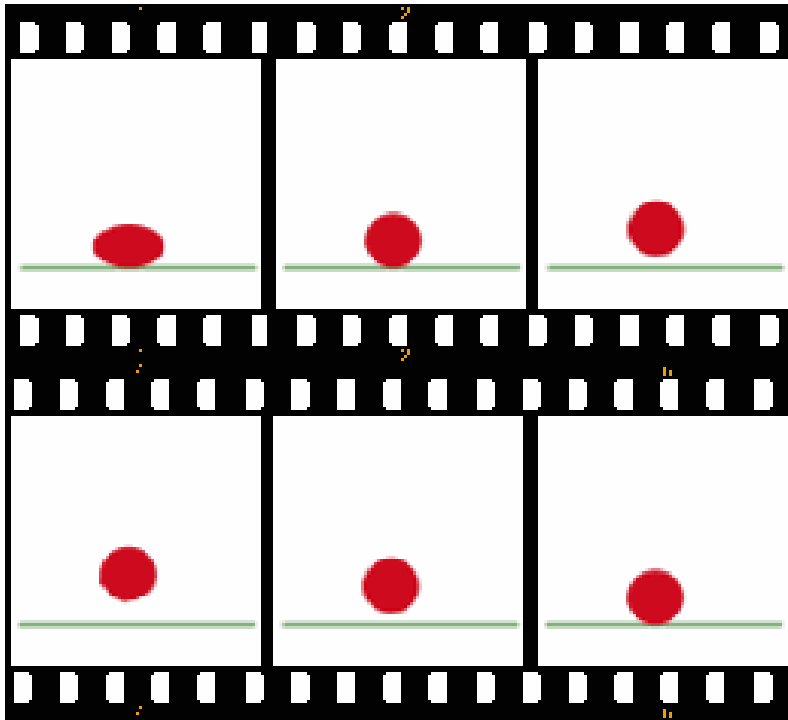
Plusieurs techniques sont utilisées :

1. **Animations par affichages successifs** : Elle consiste à afficher successivement plusieurs images présentant les différentes phases d'un mouvement. Le nombre d'images qu'il faut afficher en une seconde pour obtenir une animation fluide dépend de la vitesse et du type de mouvement. Si le mouvement est lent, on peut se contenter d'un nombre d'images faible. Si le mouvement est rapide, il ne faudra pas descendre en dessous de *16 images seconde*. Il n'est pas envisageable d'afficher de longues séquences animées composées d'images de la taille de l'écran en **16 millions de couleurs** à moins de disposer d'un équipement professionnel.

II.5. Animation d' Images

Afin d'optimiser une animation, trois options sont possibles :

- Réduire la taille des images.
- Réduire le nombre d'images par seconde .
- Réduire le nombre de couleurs.



II.5. Animation d' Images

2. Animations par déplacement d'image :

Elle consiste à afficher une image fixe appelée **sprite** à différentes positions successives de l'écran. Ces positions sont calculées en fonction d'une trajectoire définie préalablement.

La vitesse de déplacement peut être fixe ou variable. Chaque fois que le programme déplace une image, il doit d'abord l'effacer avec le fond avant de l'afficher à sa nouvelle position. Tout déplacement d'image consiste en fait à déplacer trois images : une image du fond, un masque de la taille de l'image sur lequel s'ajoute l'image.

II.5. Animation d' Images

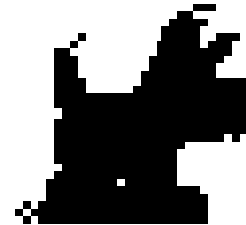
- Les images déplacées sont toujours de forme rectangulaire, ce qui peut poser un problème. En effet, il est rare que les objets représentés soient rectangulaires. Si l'image déplacée a un fond identique à celui du décor (une couleur unie) alors la solution consiste à déplacer un morceau de fond en même temps. Si le fond n'est pas uni alors l'image déplacée doit avoir un fond transparent, et donc posséder une couleur déclarée comme transparente.

II.5. Animation d' Images

Nous avons une image à l'écran (le fond) sur laquelle nous voulons voir se déplacer un chien (le sprite) :



Sprite



Masque

II.5. Animation d' Images

3. Animations par déplacement d'images animées :

Il est tout à fait possible d'afficher de la même façon une image animée. Il suffit pour cela de créer les différentes images correspondant aux phases de l'animation. Pour chaque image, on crée également un masque qui permette de voir le fond sur lequel se déplace l'animation. On affiche alors successivement les images selon la trajectoire du mouvement choisi.

Remarque : Lorsque l'affichage est effectué en couleurs indexées, c'est à dire à l'aide d'une palette de 256 couleurs, il est impératif que tous les éléments utilisent la même palette.

II.5. Compression des Images

La *compression d'image* est une application de la compression de données sur des images numériques. Cette compression a pour utilité de réduire la redondance des données d'une image Bitmap afin de pouvoir l'emmagasiner sans occuper beaucoup d'espace ou la transmettre rapidement.

La compression d'image peut être effectuée *avec perte* de données ou *sans perte*. La compression *sans perte* est parfois préférée pour des images artificielles telles que les schémas, les dessins techniques, les icônes ou les bandes dessinées. C'est à cause du fait que les méthodes de compression avec perte, lorsqu'en particulier elles sont utilisées à des bas *débits binaires*, présentent des défauts.

II.5. Compression des Images

Des méthodes de compression sans perte peuvent également être préférées pour garder une grande précision, tel que pour des balayages médicaux, ou des numérisations d'images destinées à l'archivage. Les méthodes avec perte sont particulièrement appropriées aux images normales telles que des photos dans les applications où une perte mineure de fidélité (parfois imperceptible) est acceptable pour réaliser une réduction substantielle du débit binaire.

Les méthodes les plus importantes de compression d'image sans perte sont :

1. la méthode du codage des répétitions.
2. le codage entropique ; tel que le **codage de Huffman**.
3. les algorithmes à dictionnaire adaptable tels que **LZW**.

II.5. Compression des Images

Les méthodes les plus importantes de compression avec perte sont :

1. la réduction de l'*espace des couleurs* aux couleurs les plus fréquentes dans une image. Les couleurs choisies sont indiquées dans la palette de couleur dans l'en-tête de l'image compressée.
2. le sous-échantillonnage de la chrominance utilisé dans **NTSC**.
3. le codage par transformation. C'est généralement la méthode la plus utilisée. La *transformée en cosinus discrète (JPEG)* et la transformation par ondelettes sont les transformations les plus populaires (*JPEG 2000*).
4. la compression fractale.

II.6. Algorithmes de compression sans pertes

Cette technique **Algorithme RLE "Run Length Encoding"** est utilisée par de nombreux formats d'images (**BMP, PCX, TIFF**), et consiste à repérer et à éliminer la redondance des données. Une image contient souvent des surfaces de couleurs identiques, il est donc plus efficace de coder cette couleur et sa répétition que de coder unitairement chaque point. Par exemple la séquence:

"**AAAAHHHHHHHHHHHHHHHH**" est remplacée par le nombre de répétitions suivi de l'élément répété "**5A14H**". Ce codage est intéressant pour des données comportant peu de valeurs différentes et de longues séquences. Par contre il est moins intéressant pour des images photographiques avec une profondeur supérieure à 256 couleurs.

II.6. Algorithmes de compression sans pertes

Algorithme de Huffman

Le codage **Huffman** (Huffman 1952) est élaboré par [David Albert Huffman](#), lors de sa thèse de doctorat au [MIT](#) publiée en 1952. Un *code binaire* dont la longueur est variable est assigné à chaque **valeur** selon le schéma suivant : « à une valeur fréquemment rencontrée on affecte un code de longueur courte » . Une table de conversion recueille les valeurs et le code qui leur est ainsi affecté. Cette table sera passée comme paramètre à tout programme de décodage avant l'envoi des données elles-mêmes.

II.6. Algorithmes de compression sans pertes

Par cette méthode, le ratio de compression peut atteindre jusqu'à **8:1**. La qualité de la compression dépendra de l'algorithme statistique qui établira la conversion. Dans la pratique, ce type de compression est peu efficace pour les fichiers où l'on rencontre de longues chaînes de valeurs différentes et uniques ; car ce genre de fichiers, une fois codés avec cette méthode, deviennent plus longs que les originaux ! Pour cette raison, le codage **Huffman** est réalisé par deux passages. Le premier passage crée *l'algorithme statistique*, le second code *les données*. Un dernier point à relever est la lenteur de cette méthode due au travail intense que le décodage impose au processeur.

II.6. Algorithmes de compression sans pertes

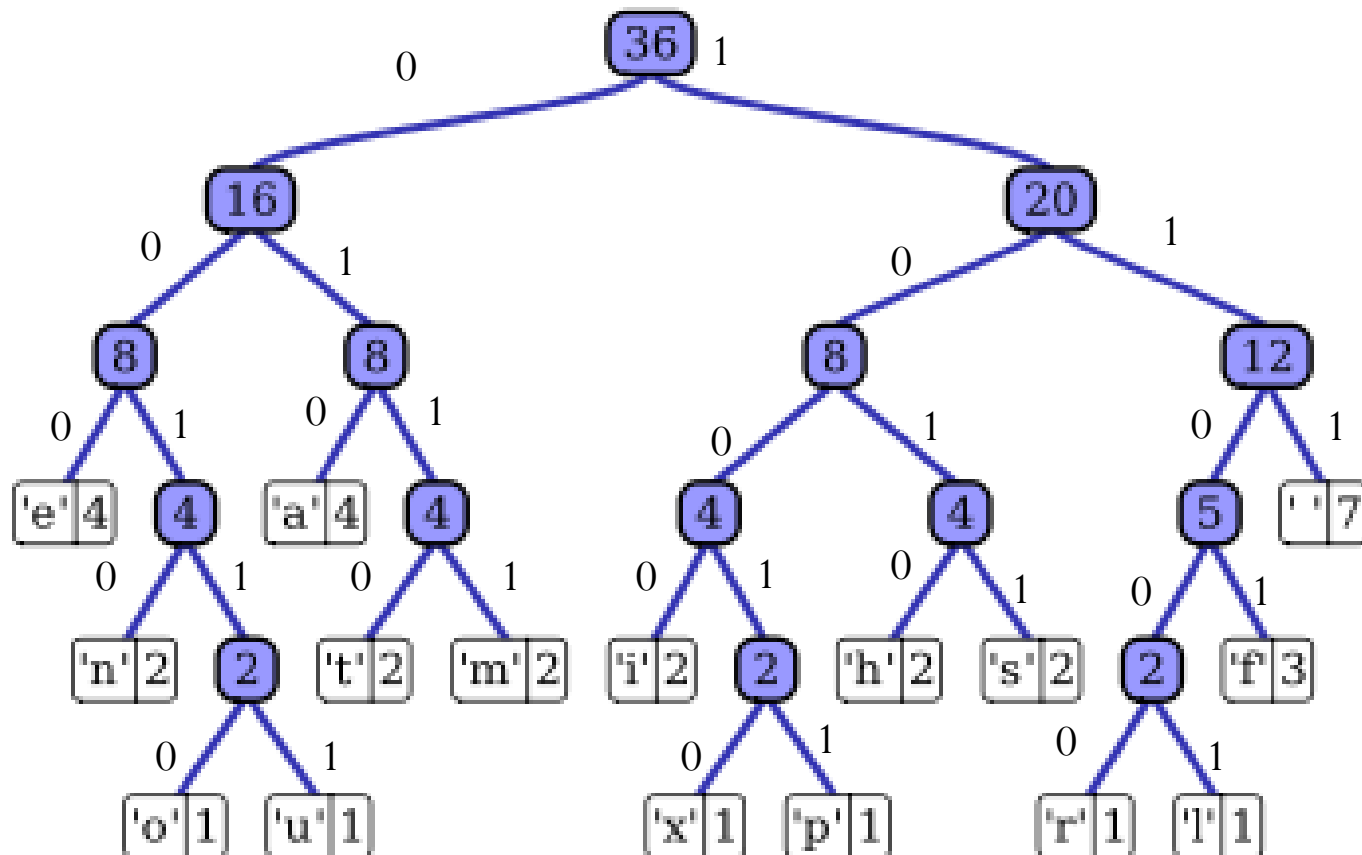
Le principe du codage de **Huffman** repose sur la création d'un **arbre binaire**. Chaque caractère (valeur) constitue une des feuilles de l'arbre à laquelle on associe un *poids* valant son *nombre d'occurrences*.

Puis l'arbre est créé suivant un principe simple : on associe à chaque fois les deux nœuds de plus faibles poids pour donner un nœud dont le poids équivaut à la **somme des poids** de ses fils jusqu'à n'en avoir plus qu'un, la *racine*. On associe ensuite par exemple le code **0** à la branche de gauche et le code **1** à la branche de droite.

II.6. Algorithmes de compression sans pertes

Arbre de Huffman

Un arbre binaire construit sur la phrase "this is an example of a huffman tree"



II.6. Algorithmes de compression sans pertes

Pour obtenir le *code binaire* de chaque caractère, on remonte l'arbre à partir de la racine jusqu'aux feuilles en rajoutant à chaque fois au code un **0** ou un **1** selon la branche suivie.

Pour coder '**this**', nous obtenons donc en binaire : **0110 1010 1000, 1011** soit 16 bits au lieu de 32 (4 caractères x 8 bits par caractère).

Cependant, il faut considérer l'espace de la table de correspondance. Dans notre cas elle occupera $16 \times (1+1) = 32$ octets maximum.

II.6. Algorithmes de compression sans pertes

Utilisation de Algorithme de Huffman

- Le codage de **Huffman** ne se base que sur la fréquence relative des symboles d'entrée (suites de bits) sans distinction pour leur provenance (images, vidéos, sons, etc.). C'est pourquoi, il est utilisé en général à un second niveau de compression, i.e, une fois la redondance propre au média mise en évidence par d'autres algorithmes. On pense en particulier à la compression JPEG pour les image, MPEG pour les vidéos et MP3 pour le son, qui peuvent retirer les éléments superflus imperceptibles pour les humains. On parle alors de compression avec perte.
- D'autres algorithmes de compression, dits sans perte, tels que ceux utilisés pour la compression de fichiers, utilisent également **Huffman** pour comprimer le dictionnaire résultant. Par exemple, LZH (Lha) et deflate (ZIP, gzip) combinent un algorithme de compression par dictionnaire (dit de Lempel-Ziv) et un codage de Huffman.

II.1. Algorithmes de compression sans pertes

Algorithme **LZW** pour *Lempel-Ziv-Welch*)

C'est un algorithme de compression de données sans perte. Il s'agit d'une amélioration de l'algorithme **LZ78** inventé par Abraham Lempel et Jacob Ziv en 1978. **LZW** fut créé en 1984 par Terry Welch, d'où son nom. **LZW** avait été breveté par la société Unisys. Il a été utilisé dans les modems (norme **V42 bis**) et est encore utilisé dans les formats d'image numérique **GIF** ou **TIFF** et les fichiers audio **MOD**.

Les ratios de compression obtenus avec cette méthode varient entre **1:1** et **3:1**, bien que certaines images fortement sous-séquencées puissent atteindre jusqu'à **10 : 1**.

II.6. Algorithmes de compression sans pertes

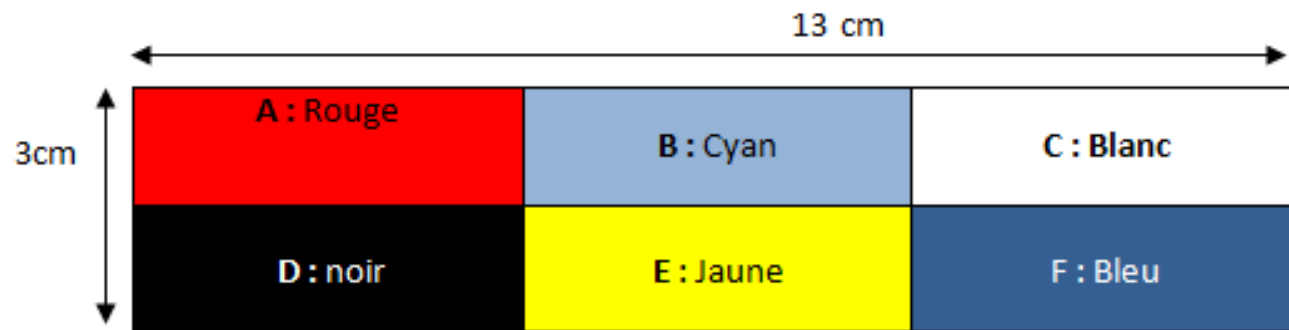
- **Algorithme** LZW pour *Lempel-Ziv-Welch*)

Elle construit dynamiquement un dictionnaire, initialisé avec les **256** valeurs de la table **ASCII**, en examinant les données : les chaînes de caractères rencontrées sont placées une par une dans le dictionnaire. Lorsqu'une chaîne est déjà présente dans le dictionnaire, son code de fréquence d'utilisation est incrémenté. Les chaînes de caractères ayant des codes de fréquence élevés sont remplacées par un "**mot**" ayant un nombre de caractères le plus petit possible et le code de correspondance est inscrit dans le dictionnaire. On obtient ainsi une information encodée et son dictionnaire. La compression sera d'autant plus efficace que le nombre de répétitions sera élevé.

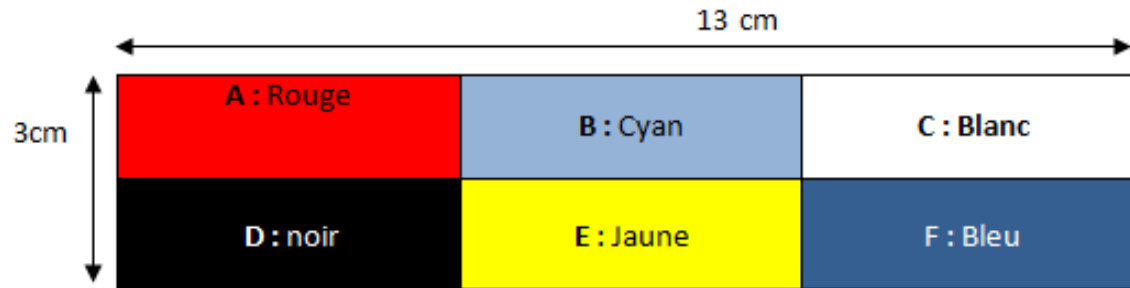
II.6. Algorithmes de compression sans pertes

EXERCICE : Soit l'Image suivante composée de six zones (a,b,c,d,e,f) colorées chacune avec une teinte unie. Supposons que la résolution de l'image précédente est de (50 Pixels par Cm) donnez son poids dans les cas d'un Codage RVB.

- Si un algorithme de compression de type RLE est utilisé, quel serait approximativement son poids dans le cas RVB; expliquez la démarche.



II.6. Algorithmes de compression sans pertes



Résolution = 50 Pixels par cm \rightarrow Définition = $(13 \times 50) \times (3 \times 50)$

- **Codage RVB**

Poids = Définition \times Nombre d'octet par pixel
 $= (13 \times 50) \times (3 \times 50) \times 3 = \mathbf{292\ 500\ octets}$

- L'algorithme **RLE** compresse en éliminant les répétitions des codes consécutifs en explorant l'image dans le sens horizontal. Chaque ligne de codes de Pixels contigus de même couleur sont réduits par le code de la couleur suivi du nombre de répétitions. Donc chaque bloc (A,B, C ,D ,E, F) sera réduit à deux colonnes de codes ; on obtient :
- Poids RLE = Nbre de blocs \times (hauteur du bloc \times Résolution \times (3 +1))
- Poids RLE = $6 \times (1,5 \times 50 \times 4\ octet) = \mathbf{1800\ octets}$

II.6. Algorithmes de compression sans pertes

- Appliquez l'algorithme de Huffman sur l'image précédente, après compression RLE.

Poids RLE = $6 \times (1,5 \times 50 \times 4 \text{ octet}) = \mathbf{1800 \text{ octets}}$

Nous avons six codes possibles : quatre codes vont être codés sur **3 bit** et les deux restants sur **2 bits**.

La taille de la table =

$$(24 + 3) \text{ bits} \times 6 = 162 \text{ bits}$$

$$\begin{aligned} \text{Poids Huffman} = & 4 \times (1,5 \times 50 \times 3 \text{ bit}) + \\ & 2 \times (1,5 \times 50 \times 2 \text{ bit}) + \text{taille table} \end{aligned}$$

$$\text{Poids Huffman} = 900 \text{ bits} + 300 \text{ bits} + 162 \text{ bits} = \mathbf{1362 \text{ octets}}$$

II.7. Algorithmes de compression avec pertes

Une compression avec perte ne s'applique qu'aux données « perceptuelles », en général *sonores ou visuelles, qui peuvent subir une modification, parfois importante, sans que cela ne soit (ou peu) perceptible par un humain*. La perte d'information est **irréversible (non conservative)** : il est impossible de retrouver les données d'origine après une telle compression.

Les **compressions d'image** s'appuient ainsi sur certaines *caractéristiques de la vision humaine* : notre oeil perçoit en effet plus facilement les grandes masses que les détails et notre rétine est davantage *sensible aux variations de lumière que de couleur*. Ces compressions, en vue du gain de place, vont donc éliminer des données qui altèrent davantage les teintes que la luminosité.

On distingue plusieurs types de compressions d'image avec perte :

II.7. Algorithmes de compression avec pertes

- ❑ **La compression DCT** de type **JPEG** (JFIF, TIF-JPEG) : pour images en 24 bits, excellente compression mais effet de mosaïque pouvant être visible, taux contrôlable par facteur de qualité, relativement rapide.
- ❑ **La compression fractale** (**FIF** : Fractal Image Format) : pour images de 8 à 24 bits, taille ajustable sans effet de pixellisation, mais présentant un effet de flou aux plus forts taux de compression, très lent en compression, décompression rapide et progressive.
- ❑ **La compression par ondelettes** (**WI** : Wavelet Image, JPEG2000) : excellente qualité, réglable, flou à fort taux de compression, pour images de 8 à 24 bits, rapide, taux contrôlable par facteur de qualité.

II.8. JPEG (Joint Photographic Experts Group)

Sous cette appellation se cachent plusieurs techniques de compression dont le principe est le suivant : peu important les données originales, on ne garde *que celles dont on ne peut se passer*, les autres sont perdues. Ce principe est applicable pour l'imagerie commerciale (dont la télévision et les applications multimédia).

Basée sur *la compression par transformations fréquentielles*. Le taux de compression d'une image **JPEG est réglable**. Plus ce taux est élevé, plus le fichier résultant est léger, et plus l'image est dégradée. Cette méthode de compression est beaucoup plus efficace sur les images photographiques (comportant de nombreux pixels de couleurs différentes) et non sur des images géométriques (à la différence de la compression **LZW**) car sur ces dernières, les différences de nuances dues à la compression sont très visibles.

II.8. JPEG (Joint Photographic Experts Group)

Principe de Compression :

Cet algorithme est de type “**Lossy**” car il y a perte d'information : en effet, cette technique profite des imperfections de la perception de l'oeil humain.

1. L'oeil perçoit mieux les contrastes sur les faibles que sur *les fortes intensités*; cette propriété a des conséquences sur le nombre de niveaux qui servent à coder l'image; celui-ci pourra être plus faible pour les faibles intensités. **Ex** : Sur une zone de l'image fortement éclaircie, les valeurs du signal vidéo pourront être codées sur 5 bits (32 niveaux) au lieu de 8 bits (256 niveaux).

II.8. JPEG (Joint Photographic Experts Group)

Principe de Compression :

2. L'oeil est plus sensible à l'intensité lumineuse qu'à la couleur. Ce qui va permettre de numériser la composante couleur de l'image (chrominance), avec moins d'échantillons sans dégradation apparente. Rapport de 1 à 2 par rapport à la luminance.
3. L'oeil enfin, agit comme *un intégrateur sur les hautes fréquences* spatiales des images. Ainsi, on peut directement éliminer certaines informations correspondant *aux hautes fréquences* de l'image, ce qui a pour effet de créer des portions d'images plus uniformes, et donc favoriser les opérations d'élimination de redondance. Par exemple, une zone comportant une alternance très serrée de raies noires et blanches peut être représentée par un nombre unique correspondant au gris.

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG

L'image est décomposée en blocs de 8×8 pixels codés en *Chrominance-luminance* auxquels on applique une transformation en *cosinus discret (DCT)* qui permet de *repérer le niveau des détails*. Un processus de quantification permet ensuite de supprimer les détails non significatifs.

Un codage *RLE* réduit les zéros consécutifs puis un codage de **Huffman** assure une compression de l'ensemble. Le taux de compression pourra être choisi selon la définition désirée, il est élevé car il s'agit d'une compression avec pertes.

11.8. JPEG (Joint Photographic Experts Group)

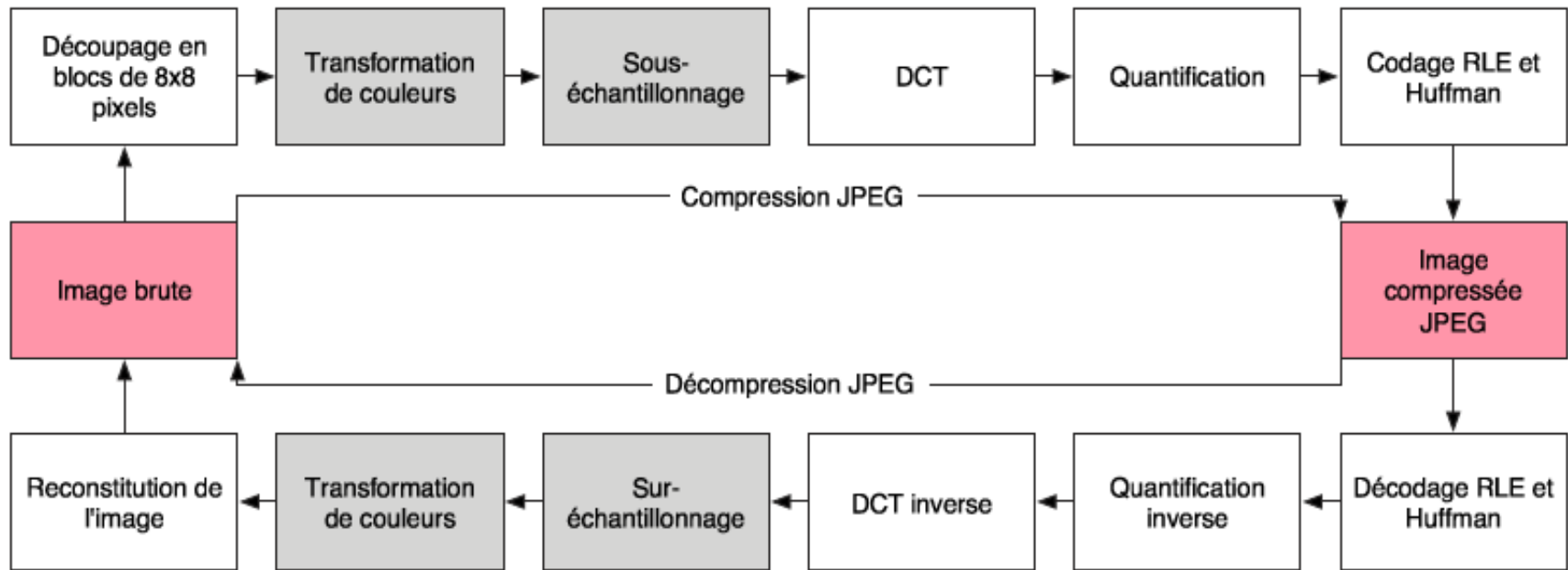


FIG. 1 – Etapes de la transformation JPEG

Attention, quand on suit l'algorithme du bas, on n'aboutit pas à l'image brute, mais à une image *restituée*, qui n'est en aucun cas l'image brute. Entre l'image brute et l'image restituée, il y a des écarts, qui seront d'autant plus importants que l'opération de quantification a été forte.

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 1 : Segmentation

Découpage en blocs : Le format JPEG, comme le font généralement les algorithmes de compression à perte, commence par découper l'image en blocs carrés de 64 (8×8).



Resolution 720x572 pixels



Block at 8x8 pixels

40	38	45	40	43	54	60	58
39	36	44	32	47	59	77	85
50	40	25	54	66	50	33	32
57	38	38	66	47	11	2	5
59	36	47	62	24	2	9	11
58	41	55	53	6	4	10	1
58	33	57	39	3	5	4	2
64	44	54	35	3	7	3	3

Color value matrix

Donnée

Une image segmentée en blocs 8×8 pixels

Objectif

Obtenir une nouvelle représentation de chaque bloc:

- contenant la même information
- l'information est concentrée sur peu d'éléments

II.8. JPEG (Joint Photographic Experts Group)

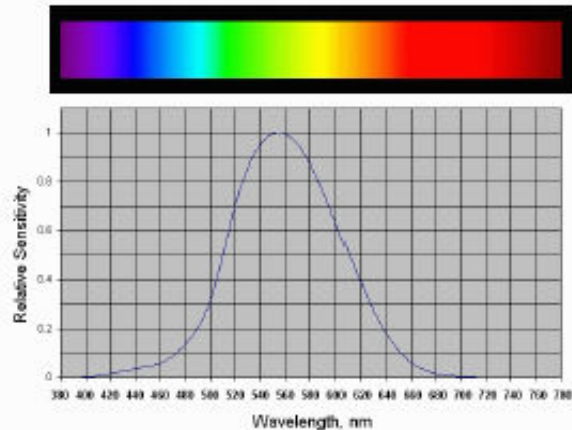
Algorithme JPEG : ETAPE 2 : Transformation des couleurs.

JPEG est capable de coder les couleurs sous n'importe quel format, toutefois les meilleurs taux de compression sont obtenus avec des codages de couleur de type luminance/chrominance tels que **YUV**, **YCbCr** car l'œil est assez sensible à la luminance mais peu à la chrominance.

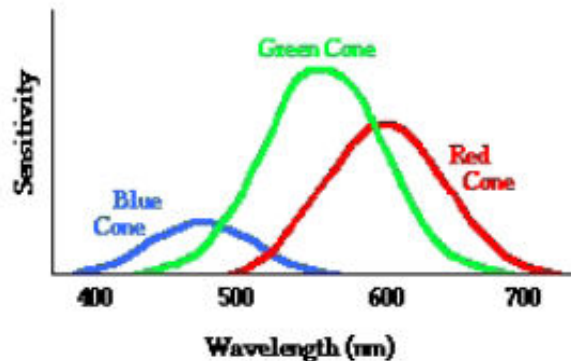
Globalement, la luminance est définie sur chaque pixel tandis que la chrominance est la valeur moyenne d'un bloc de 2×1 pixels voire 2×2 pixels.

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 3 : Sous-échantillonnage



Response of Human Eye Versus Wavelength
(Data from the 1988 C.I.E. Photopic Luminous Efficiency Function)



Processus

Construction additive des couleurs par trois types de cones (R,G,B).

Gamme Visuelle

de 400 nm (violet) à 700 nm (rouge)

Sensibilité

- Vision maximale dans le vert
- Vision minimale dans les bleus et rouges
- Sensibilité très différente selon les personnes

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 3 : Sous-échantillonnage

La façon la plus simple d'exploiter la faible sensibilité de l'œil à la chrominance est simplement de sous-échantillonner les signaux de chrominance. Généralement, on utilise un sous-échantillonnage de type **2h1v** ou **2h2v**. Dans le premier cas (le plus utilisé), on a un sous-échantillonnage **2:1** horizontalement et **1:1** verticalement ; dans le deuxième cas, on a un sous-échantillonnage **2:1** horizontalement et verticalement. Ces sous-échantillonnages sont utilisés pour les chrominances, pour la luminance on n'utilise jamais de sous-échantillonnage.

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 4 : La transformée DCT

(*Discrete Cosine Transform*, en français transformée en cosinus discrète), est une transformation numérique qui est appliquée à chaque bloc. Cette transformée est une variante de la transformée de Fourier. Cette méthode permet de décrire chaque bloc en une carte de fréquences et en amplitudes plutôt qu'en pixels et couleurs. La valeur d'une fréquence reflète l'importance et la rapidité d'un changement.

À chaque bloc de pixels $N \times N$ sont ainsi associées $N \times N$ fréquences. Le calcul d'une **DCT** est complexe. C'est l'étape qui coûte le plus de temps et de ressources dans la compression et la décompression **JPEG**, mais c'est peut-être la plus importante car elle nous permet de séparer les basses fréquences et les hautes fréquences présentes dans l'image.

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 4 : La transformée DCT

En particulier, le coefficient d'ordre **(0,0)**, appelé aussi coefficient **DC**, représente *la composante continue*, c'est-à-dire la **luminance moyenne du bloc**. Plus l'ordre du coefficient augmente, plus il exprime une variation *spatiale de haute fréquence*, dans le sens horizontal ou vertical de l'image. Il est évident que, pour la plupart des *blocs*, tant la luminance que la chrominance seront plus ou moins constantes, ce qui se traduira par : - des coefficients de la **DCT** de valeur importante, pour les ordres faibles (basses fréquences), - des coefficients peu significatifs pour les ordres élevés. En sorte, On traduit le bloc de pixels en **64** coefficients d'éloignement par rapport à la moyenne

Note : la **DCT** est une opération réversible, si tous les coefficients sont conservés

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 4 : La transformée DCT

La transformée **DCT** s'exprime mathématiquement par :

$$\text{DCT}(i,j) = \frac{2}{N} C(i) C(j) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \text{pixel}(x,y) \cos \left[\frac{(2x+1)i\pi}{2N} \right] \cos \left[\frac{(2y+1)j\pi}{2N} \right]$$

Et la transformée **DCT** inverse s'exprime par :

$$\text{pixel}(x,y) = \frac{2}{N} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} C(i) C(j) \text{DCT}(i,j) \cos \left[\frac{(2x+1)i\pi}{2N} \right] \cos \left[\frac{(2y+1)j\pi}{2N} \right]$$

Dans les deux cas, la constante **C** vaut :

$$C(x) = \begin{cases} \frac{1}{\sqrt{2}} & \text{pour } x = 0 \\ 1 & \text{pour } x > 0 \end{cases}$$

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 4 : La transformée DCT

Principe :

Matrice A

$$\begin{pmatrix} Y(0,0) & \dots & Y(0,7) \\ \dots & & \dots \\ Y(7,0) & \dots & Y(7,7) \end{pmatrix}$$

On considère la matrice P formée par les éléments P_{ij} :

$$p_{ij} = \alpha_j \sqrt{\frac{2}{N}} \cos\left(\frac{(2i+1)j\pi}{2N}\right)$$

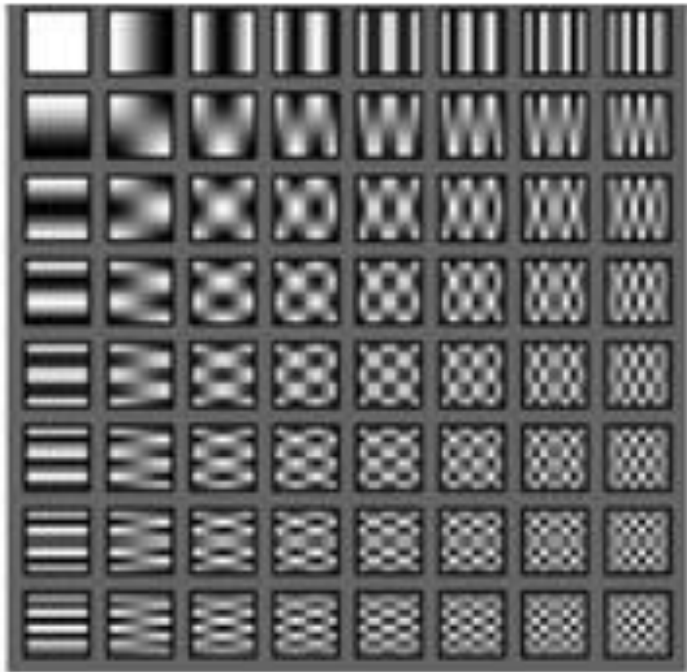
donnant une matrice de transformation P :

$$\begin{pmatrix} \frac{1}{\sqrt{8}} \frac{1}{2} \cos(\frac{\pi}{16}) & \frac{1}{2} \cos(\frac{2\pi}{16}) & \dots & \frac{1}{2} \cos(\frac{7\pi}{16}) \\ \frac{1}{\sqrt{8}} \frac{1}{2} \cos(\frac{3\pi}{16}) & \frac{1}{2} \cos(\frac{3*2\pi}{16}) & \dots & \frac{1}{2} \cos(\frac{3*7\pi}{16}) \\ \dots & & & \\ \frac{1}{\sqrt{8}} \frac{1}{2} \cos(\frac{15\pi}{16}) & \dots & & \frac{1}{2} \cos(\frac{15*7\pi}{16}) \end{pmatrix}$$

La transformation se fait par : $F = {}^t P \times A \times P$

II.8. JPEG ([Joint Photographic Experts Group](#))

Algorithme JPEG : ETAPE 4 : La transformée DCT



Calcul

Compilation de la DCT

- Construction des **matrices de base**
- Utilisation d'un opérateur de masquage pour obtenir les coefficients

Bloc de coefficients

- Coefficients les plus importants en haut à gauche (faibles fréquences)
- Les coefficients diminuent vers la droite et vers le bas

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 4 : La transformée DCT

Matrice (bloc de pixels) de base :

$$f = \begin{bmatrix} 139 & 144 & 149 & 153 & 155 & 155 & 155 & 155 \\ 144 & 151 & 153 & 156 & 159 & 156 & 156 & 156 \\ 150 & 155 & 160 & 163 & 158 & 156 & 156 & 156 \\ 159 & 161 & 162 & 160 & 160 & 159 & 159 & 159 \\ 159 & 160 & 161 & 162 & 162 & 155 & 155 & 155 \\ 161 & 161 & 161 & 161 & 160 & 157 & 157 & 157 \\ 162 & 162 & 161 & 163 & 162 & 157 & 157 & 157 \\ 162 & 162 & 161 & 161 & 163 & 158 & 158 & 158 \end{bmatrix}$$

Équation 4 : Matrice d'origine.

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 4 : La transformée DCT

En effectuant la transformée DCT on obtient la matrice des fréquences suivante :

$$F = \begin{bmatrix} 1260 & -1 & -12 & -5 & 2 & -2 & -3 & 1 \\ -23 & -17 & -6 & -3 & -3 & 0 & 0 & -1 \\ -11 & -9 & -2 & 2 & 0 & -1 & -1 & 0 \\ -7 & -2 & 0 & 1 & 1 & 0 & 0 & 0 \\ -1 & -1 & 1 & 2 & 0 & -1 & 1 & 1 \\ 2 & 0 & 2 & 0 & -1 & 1 & 1 & -1 \\ -1 & 0 & 0 & -1 & 0 & 2 & 1 & -1 \\ -3 & 2 & -4 & -2 & 2 & 1 & -1 & 0 \end{bmatrix}$$

Équation 5 : Matrice transformée DCT.

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 4 : La transformée DCT

En effectuant la transformée DCT on obtient la matrice des fréquences suivante :

$$F = \begin{bmatrix} 1260 & -1 & -12 & -5 & 2 & -2 & -3 & 1 \\ -23 & -17 & -6 & -3 & -3 & 0 & 0 & -1 \\ -11 & -9 & -2 & 2 & 0 & -1 & -1 & 0 \\ -7 & -2 & 0 & 1 & 1 & 0 & 0 & 0 \\ -1 & -1 & 1 & 2 & 0 & -1 & 1 & 1 \\ 2 & 0 & 2 & 0 & -1 & 1 & 1 & -1 \\ -1 & 0 & 0 & -1 & 0 & 2 & 1 & -1 \\ -3 & 2 & -4 & -2 & 2 & 1 & -1 & 0 \end{bmatrix}$$

Équation 5 : Matrice transformée DCT.

II.8. JPEG (Joint Photographic Experts Group)

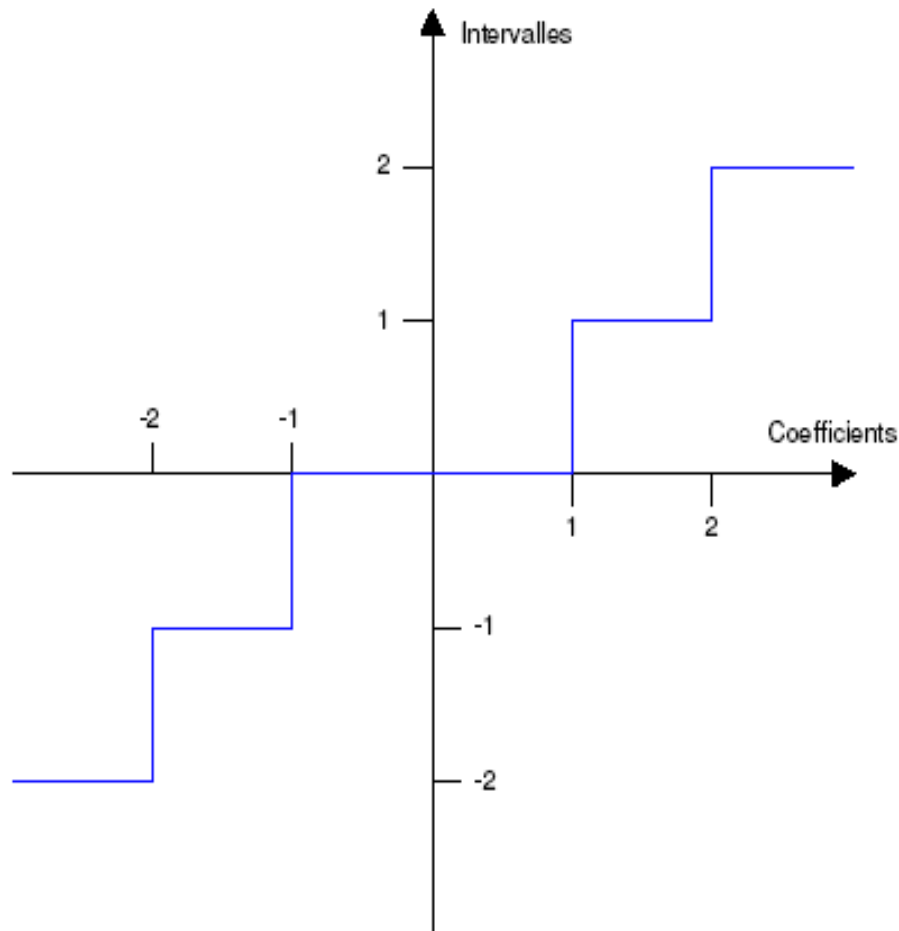
Algorithme JPEG : ETAPE 5 : La quantification

La **quantification** est l'étape dans laquelle on perd réellement des informations (et donc de la qualité visuelle), mais c'est celle qui fait gagner beaucoup de place (contrairement à la **DCT**, qui ne compresse pas). La **DCT** a retourné, pour chaque bloc, une matrice de **8×8** nombres (fréquences). La quantification consiste à diviser cette matrice par une autre, appelée matrice de quantification, et qui contient **8×8** coefficients savamment choisis par le codeur.

Le but est ici d'atténuer les hautes fréquences (*les valeurs de coefficients faibles*), c'est-à-dire celles auxquelles l'œil humain est très peu sensible. Ces fréquences ont des amplitudes faibles, et elles sont encore plus atténuées par la quantification (les coefficients sont même ramenés à **0**).

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 5 : La quantification



Objectif

Coder les coefficients sur un minimum de valeurs possibles

Principe

- Choisir un ensemble de I intervalles
- Associer à chaque coefficient son intervalle correspondant
- Coder les intervalles sur $\log_2(I)$ bits.

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 5 : La quantification

La quantification ramène beaucoup de coefficients à **0** (surtout en bas à droite dans la matrice, là où sont les hautes fréquences). Seules quelques informations essentielles (coin en haut à gauche) sont gardées pour représenter le bloc.

L'intérêt est qu'au moment de coder le résultat dans le fichier, la longue suite de zéros nécessitera très peu de place. Mais si la quantification est trop forte (= taux de compression trop élevé), il y aura trop peu de coefficients non nuls pour représenter fidèlement le bloc ; dès lors, à l'écran la division en blocs devient visible, et l'image apparaît « pixellisée ».

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 5 : La quantification

Matrice quantifiée : $F^*(i,j) = \lceil F(i,j) / Q(i,j) \rceil$

Considérons la matrice de quantification suivante :

$$Q = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

11.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 5 : La quantification

Ce qui donne comme matrice des fréquences quantifiée :

Matrice quantifiée.

$$F = \begin{bmatrix} 1260 & -1 & -12 & -5 & 2 & -2 & -3 & 1 \\ -23 & -17 & -6 & -3 & -3 & 0 & 0 & -1 \\ -11 & -9 & -2 & 2 & 0 & -1 & -1 & 0 \\ -7 & -2 & 0 & 1 & 1 & 0 & 0 & 0 \\ -1 & -1 & 1 & 2 & 0 & -1 & 1 & 1 \\ 2 & 0 & 2 & 0 & -1 & 1 & 1 & -1 \\ -1 & 0 & 0 & -1 & 0 & 2 & 1 & -1 \\ -3 & 2 & -4 & -2 & 2 & 1 & -1 & 0 \end{bmatrix}$$

[illegible]

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 5 : La quantification

La matrice de quantification peut être donnée différemment

Soit la matrice de quantification Q définie pour chacun des blocs de l'image par :

$$Q(i, j) = 1 + (1 + i + j)F_q$$

- $Q(i, j)$: valeur du pas de quantification dans un bloc ($i = 0 \dots N - 1$ et $j = 0 \dots N - 1$).
- F_q : facteur de qualité de la quantification.

II.8. JPEG (*Joint Photographic Experts Group*)

Algorithme JPEG : ETAPE 5 : La quantification



Une photo de fleur compressée en JPEG, avec des compressions de plus en plus fortes, de gauche à droite.

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 6 : Codage, compression RLE et Huffman

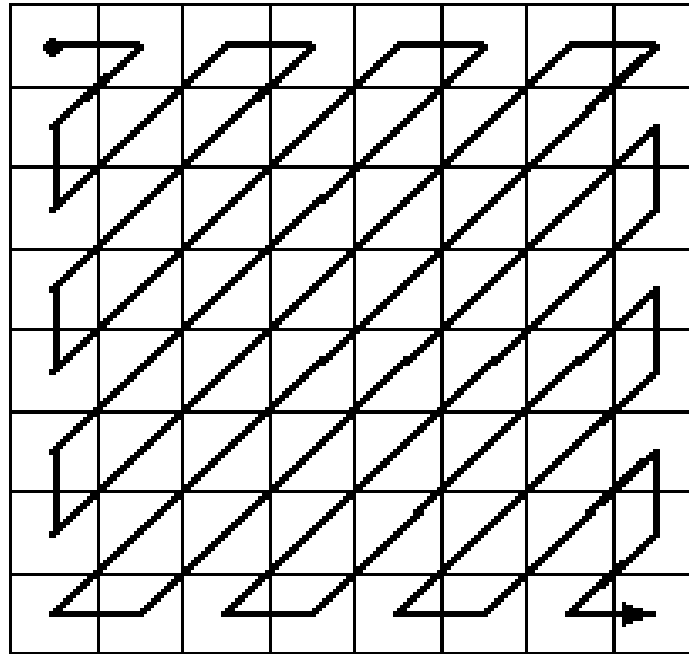
Viennent enfin deux encodages **sans perte**, qui réduisent encore le *bitrate* :

- Un codage **RLE** qui code plutôt les séquences de zéros;
- Un codage de type **Huffman**, qui accorde aux symboles les plus fréquents les codes les plus courts.

II.8. JPEG (*Joint Photographic Experts Group*)

Algorithme JPEG : ETAPE 6 : Codage, compression RLE et Huffman

Le codage s'effectue en zigzag comme le montre la figure suivante et se termine par un caractère de fin :



Ordre de codage défini par la norme JPEG.

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 6 : Codage, compression RLE et Huffman

Lorsqu'on regarde l'exemple présenté à la section précédente, la matrice quantifiée possède des **0** et **-1** non en séquence. Le codage de Huffman a pour but d'attribuer un code de petite taille à une valeur fréquente dans la matrice. Il permet une compression statistique des données.

Ce codage nous délivre **deux tables** (quantification et huffman) qui sont enregistrées dans le fichier final.

II.8. JPEG (Joint Photographic Experts Group)

Algorithme JPEG : ETAPE 6 : Codage, compression RLE et Huffman

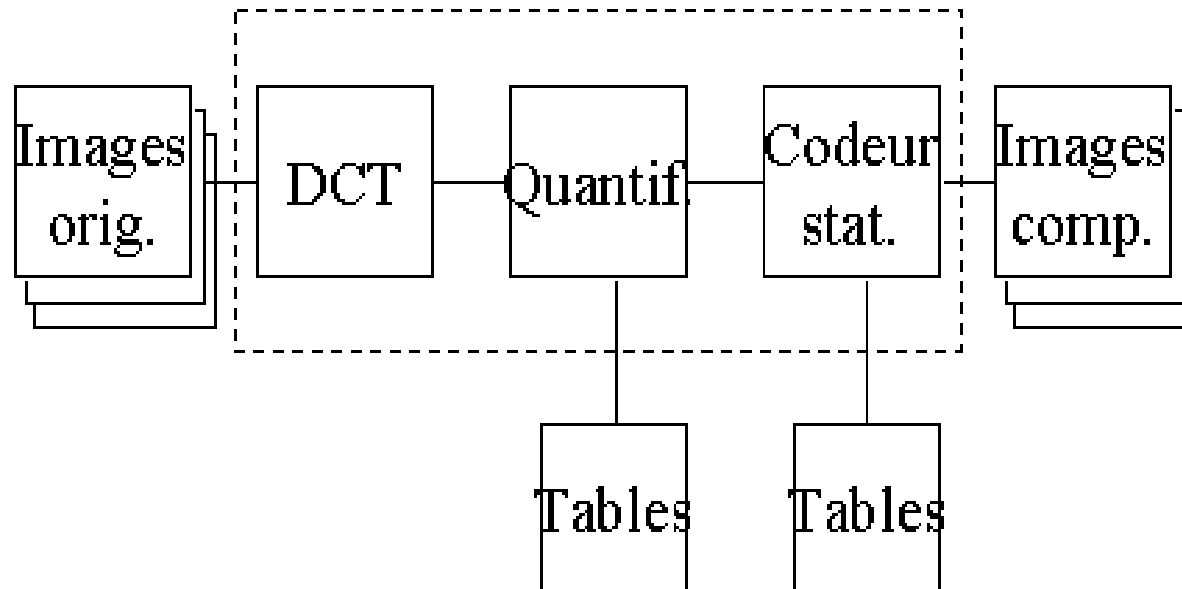


Schéma de codage simplifié.

II.8. JPEG (Joint Photographic Experts Group)

Décompression JPEG

Les étapes de la décompression s'effectuent dans l'ordre inverse de la compression suivant les méthodes définies précédemment.

$$f = \begin{bmatrix} 139 & 144 & 149 & 153 & 155 & 155 & 155 & 155 \\ 144 & 151 & 153 & 156 & 159 & 156 & 156 & 156 \\ 150 & 155 & 160 & 163 & 158 & 156 & 156 & 156 \\ 159 & 161 & 162 & 160 & 160 & 159 & 159 & 159 \\ 159 & 160 & 161 & 162 & 162 & 155 & 155 & 155 \\ 161 & 161 & 161 & 161 & 160 & 157 & 157 & 157 \\ 162 & 162 & 161 & 163 & 162 & 157 & 157 & 157 \\ 162 & 162 & 161 & 161 & 163 & 158 & 158 & 158 \end{bmatrix}$$

$$f = \begin{bmatrix} 144 & 146 & 149 & 152 & 154 & 156 & 156 & 156 \\ 148 & 150 & 152 & 154 & 156 & 156 & 156 & 156 \\ 155 & 156 & 157 & 158 & 158 & 157 & 156 & 155 \\ 160 & 161 & 161 & 162 & 161 & 159 & 157 & 155 \\ 163 & 163 & 164 & 164 & 162 & 160 & 158 & 156 \\ 163 & 163 & 164 & 164 & 162 & 160 & 158 & 157 \\ 160 & 161 & 162 & 162 & 162 & 161 & 159 & 158 \\ 158 & 159 & 161 & 161 & 162 & 161 & 159 & 158 \end{bmatrix}$$

Résultat de la décompression.

II.8. JPEG (Joint Photographic Experts Group)

Décompression JPEG

Ainsi que la matrice d'erreur :

$$e = \begin{bmatrix} -5 & -2 & 0 & 1 & 1 & -1 & -1 & -1 \\ -4 & 1 & 1 & 2 & 3 & 0 & 0 & 0 \\ -5 & -1 & 3 & 5 & 0 & -1 & 0 & 1 \\ -1 & 0 & 1 & -2 & -1 & 0 & 2 & 4 \\ -1 & 0 & 1 & -2 & -1 & 0 & 2 & 4 \\ -2 & -2 & -3 & -3 & -2 & -3 & -1 & 0 \\ 2 & 1 & -1 & 1 & 0 & -4 & -2 & -1 \\ 4 & 3 & 0 & 0 & 1 & -3 & -1 & 0 \end{bmatrix}$$

L'erreur est majorée par 5 et a une moyenne de **1,6** pour un coefficient de **150** ; ce qui nous donne un pourcentage d'erreur d'environ **1 %**, sur **64** valeurs.