

Mini Lasse9 VPN

Exercice 1

Soit une image à niveaux de gris, codée sur 8bits, représentée par la matrice suivante :

100	100	50	50	200
100	50	50	50	200
100	200	200	200	200
100	200	200	200	200

Matrice1

1. Calculer le contraste et la luminance de l'image ?
2. Quelle est la dynamique de l'image ? [50, 200]

1- La luminance = brillance = moyenne de tous les pixels de l'image

$$I_{moy} = \bar{m} = \frac{1}{NM} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} f(i,j)$$

A.N : $m = 137.5$

Le contraste = Ecart type des variations de niveaux d'intensité

$$\sigma = \sqrt{var} = \sqrt{\frac{1}{NM} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} (f(i,j) - \bar{m})^2}$$

A.N : $\text{sqrt}(var) = 64.95$

2 – La dynamique = [val_min, val_max] = [50, 200]

Chnahouwa un filtre ?

- Un filtre est une transformation mathématique (appelée produit de convolution) permettant, pour chaque pixel de la zone à laquelle il s'applique, de modifier sa valeur en fonction des valeurs des pixels avoisinants, affectées de coefficients.

On considère H le filtre de convolution défini par le noyau suivant :

1	-3	1
-3	9	-3
1	-3	1

3. Quel est le gain en continu du filtre H ? $\sum_i \sum_j h(i,j) = 1$

4. Appliquer le filtre H à l'image représentée par la matrice 1.

5. Donner le résultat de l'application du filtre Médian à la matrice 1.

3 –

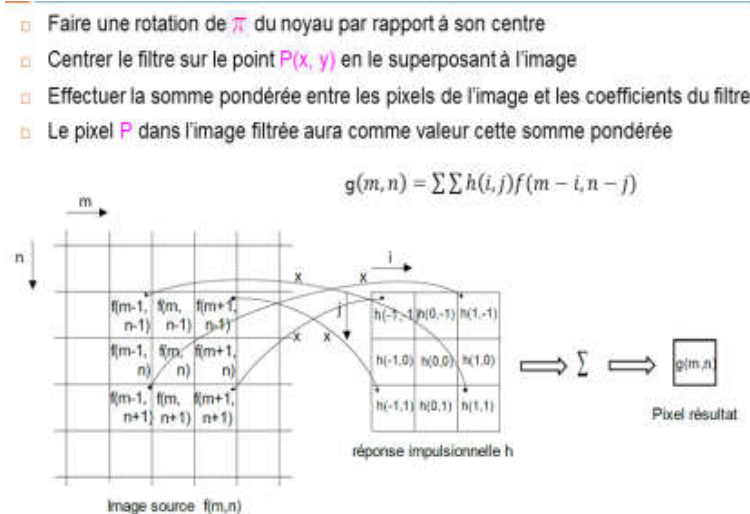
Gain en continu du filtre est $\sum_i \sum_j h(i,j)$

A.N : Gain en continu = $1 - 3 + 1 - 3 + 9 - 3 + 1 - 3 + 1 = 1$

4- Ce type de filtres est un filtre linéaire (de convolution)

Remarque : produit de convolution != produit matriciel.

Houwa sahel fl 7sab, mais khasso lkhaterr...



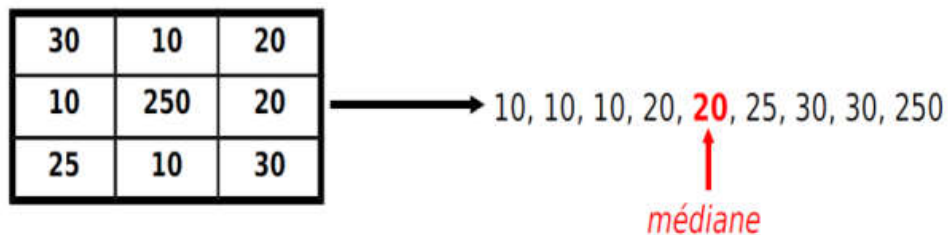
- Il faut traiter les bords de la matrice, kan3emrouhoum donc soit par des zeros : zero padding(ça facilite le 7sab)
- sinon par duplication (une valeur appartenant aux bords prend la valeur la plus proche d'elle)

100	100	100	50	50	200	200
100	100	100	50	50	200	200
100	100	50	50	50	200	200
100	100	200	200	200	200	200
100	100	200	200	200	200	200
100	100	200	200	200	200	200

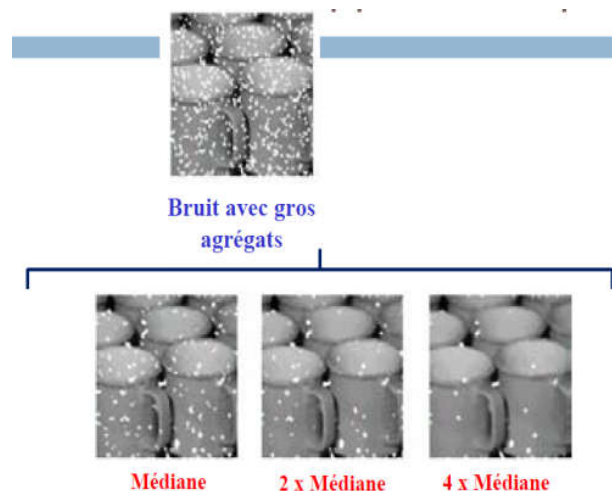
Exemple de duplication

5 – C'est un filtre non linéaire .

- ❑ Il n'est pas implémenté comme un produit de convolution
- ❑ Il est mieux que le filtre moyenne ou le filtre gaussien, pour nettoyer le bruit dans une image
- ❑ On remplace la valeur d'un pixel par la valeur médiane dans son voisinage
- ❑ Particulièrement utile pour un bruit de type Poivre et Sel



- Bruit de type Poivre : des points noirs sur l'image
- Bruit de type sel : des points blancs sur l'image :3
- Des fois il faut appliquer le filtre médiane plusieurs fois pour avoir un bon résultat.



Exercice 2 :

Nous faisons une segmentation par région de l'image de la figure 1. Le prédicat p considéré est

$$p = \text{couleur identique}$$

1. Donnez le résultat de cette segmentation par la méthode quadtree en considérant un taux de tolérance égale à 70%; à noter chaque région par un label i avec $i \in \mathbb{N}$; (ici on demande le résultat direct de la segmentation par la méthode considérée. On ne demande pas le résultat après fusion des régions).
2. Quel est le nombre de segments obtenus ? *un seul*

1 – Lli fhemt c'est qu'il faut décomposer récursivement l'image en 4 images tel que chaque image est un groupe de points vérifiant le prédicat p (couleur identique), et que l'ensemble soit identique au moins à 70%.

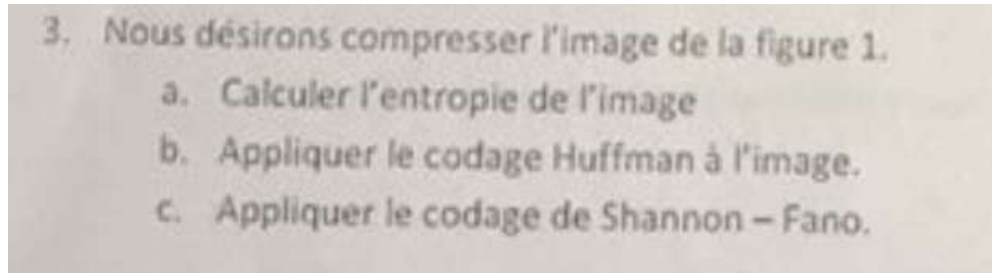
Je pense qu'il faut essayer de regrouper des groupes de tailles $n \times n$ en commençant par 16×16 w 7na nazlin b les puissances de 2. (8×8 4×4 2×2 1×1)



(<https://www.enseignement.polytechnique.fr/informatique/INF411/TD/TD4/INF411-TD4-1.php>)

2 – Remarque : La segmentation est utilisée dans la compression

- 2 segments du niveau de gris 100
- 3 segments du niveau de gris 50
- 9 segments du niveau de gris 80
- 5 segments du niveau de gris 30

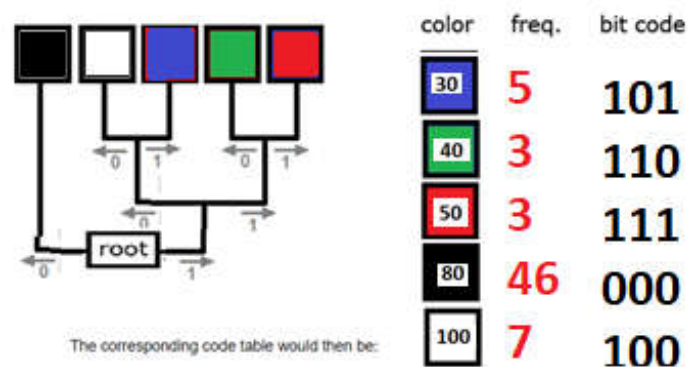


3 – a- L'entropie est définie fl cours de l'indexation comme :

Entropie	$\sum_i \sum_j c_{ij} \log(c_{ij})$
----------	-------------------------------------

b- Makaynch fl cours :/

D'après cette explication (<https://www.print-driver.com/stories/huffman-coding-jpeg>) nous devons avoir quelque chose de ce genre



c –

Couleur (niveau de gris)	Fréquence	Code
80	46	0
100	7	10
30	5	110
40	3	1110
50	3	1111

(maaaaaachi sûr mn hadchi)

- d. Donner le nombre moyen de bits nécessaires pour chaque type de codage.
- e. Donner le gain de compression de chaque technique.
- f. Donner l'erreur quadratique moyenne de chaque technique.

d- voir l'exercice suivant

e- voir l'exercice suivant

f- ??????

Exercice 3 :

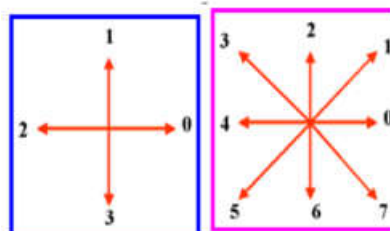
Exercice 3

Nous considérons la figure2 qui représente une image binaire. Cette image contient un objet représenté avec la couleur noire.

Figure 2 : image2

1. Donnez le nombre de composantes connexes et de trous pour l'objet de la figure2 en connexité 4 et en connexité 8.
2. Calculez le nombre d'Euler (en connexité 4 et en connexité 8) pour l'objet de la figure2.
3. En considérant une connexité 4, donnez la chaîne de Freeman utilisée pour coder le contour de la composante connexe à laquelle appartient le point A. Soit A le point de départ du codage.

1 – Pour connaître si une région est un composante connexe, il faut utiliser un des patterns



4 connexes — 8 connexes

Nombre de composantes connexes en connexité 4 : 3

Nombre de composantes connexes en connexité 8 : 1

Rappel : On note tous l'ensemble de pixels de l'arrière plan délimité par un contour interne d'une composante connexe.

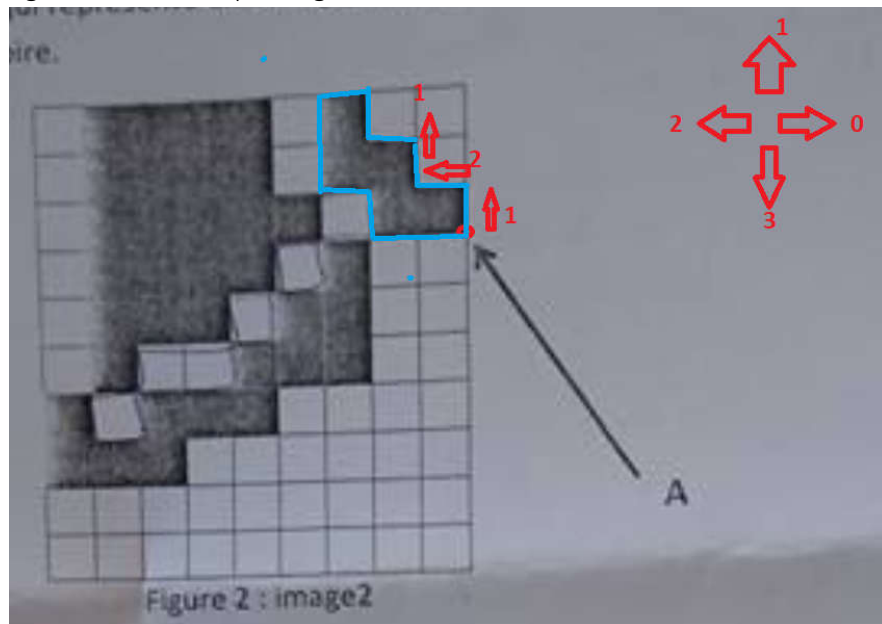
Nombre de trous en connexité 4 : 0

Nombre de trous en connexité 8 : 5

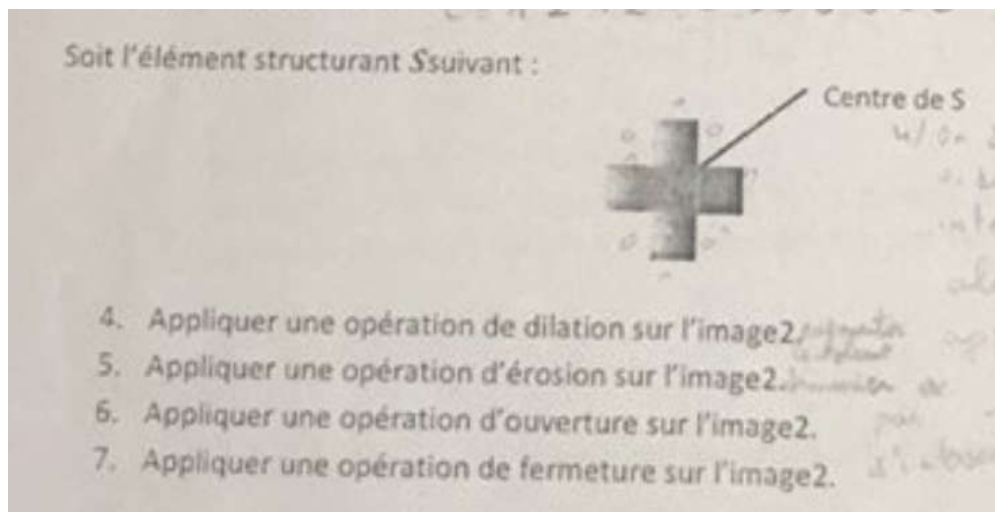
2- Rq : Le nombre d'Euler peut être négatif.

Soit C le nombre de composantes connexes et H le nombre de trous, le nombre d'Euler est donné par $E = C - H$

3 – Je pense ghir en lisant la réponse ghadi tfhem :3

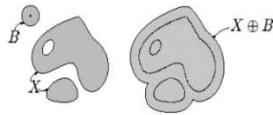


La chaîne de Freeman : 121212330300



4, 5, 6 et 7 -

□ Dilatation de X par un disque B

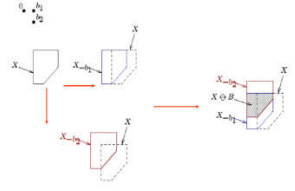


Définition : Pour un élément structurant B , la dilatacion de X par B est l'ensemble obtenu en remplaçant chaque pixel p de X par sa fenêtre B_p .

Remarque : La dilatacion permet de boucher les trous et augmenter la taille de tous les objets

Erosion morphologique

Définition : L'érosion de X par B est l'ensemble des pixels p tels que la fenêtre B_p est incluse dans X , notée $X \ominus B$

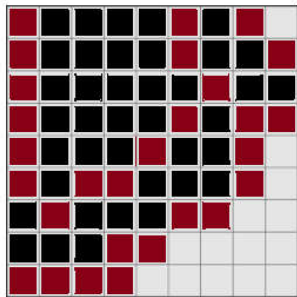


□ Définition de l'ouverture : l'ouverture de X par B est la composition de l'érosion par B suivie de la dilatacion par B :

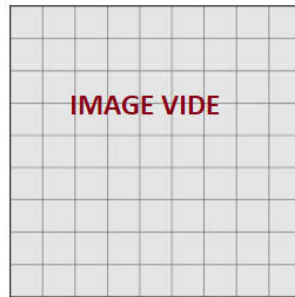
$$\text{OuvB}(X) = \text{DilB}(\text{ErosB}(X))$$

□ Définition de la fermeture : la fermeture de X par B est la composition de la dilatacion par B suivie de l'érosion par B :

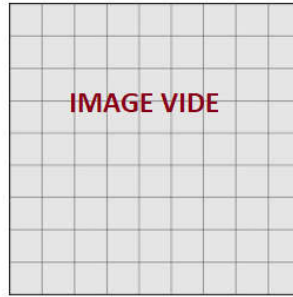
$$\text{FermB}(X) = \text{ErosB}(\text{DilB}(X))$$



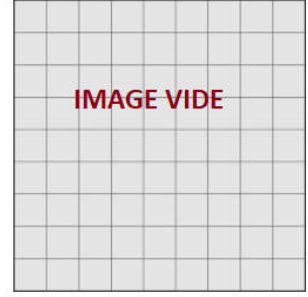
Dilatation



Erosion



Ouverture



Fermeture

(je pense que lors de la dilatacion l'image ghada tekber, donc nous allons avoir une matrice 10*11 w allaho a3lam)

Exercice 1 (TD3)

Exercice 1

Soit l'image en couleur suivante (voir figure 1), notée I . Cette image est représentée dans l'espace RGB et codée par 8 bits par canal de couleur.

100	200	200	255
10	0	200	205
100	100	200	200
100	100	150	255
100	100	50	200
100	100	100	55
100	100	100	100
100	100	100	100
100	100	100	100
10	100	100	10
100	100	100	100
100	100	100	100

Figure1 : Représentation matricielle d'une image couleur dans RGB.

1. Convertir l'image I en une image au niveau de gris. Soit I_g l'image résultante.

1 - $I_g = [[70, 100, 200, 220],$
 $[100, 100, 100, 170],$
 $[100, 100, 100, 100],$
 $[70, 100, 100, 70]]$

2. En considérant la position du point A (voir figure 4) dans l'image au niveau de gris I_g , tracer sur le même résultat de la question 1 le contour suivant en considérant une connexité 8 : C = 34460072

3. Donner le résultat du seuillage de l'image I_g selon les modes de son histogramme.

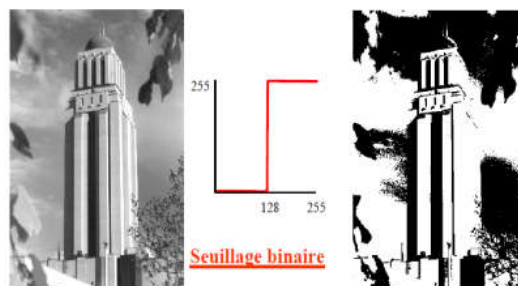
2 -

100	200	200	255
10	0	200	205
100	100	200	200
100	100	150	255
100	100	50	200
100	100	100	55
100	100	100	100
100	100	100	100
100	100	100	100
10	100	100	10
100	100	100	100
100	100	100	100

3- On calcule la fréquence des niveaux de gris et on choisi le niveau avec la plus grande fréquence (je pense)

Transformation non-linéaire d'un histogramme :
 seuillage

77

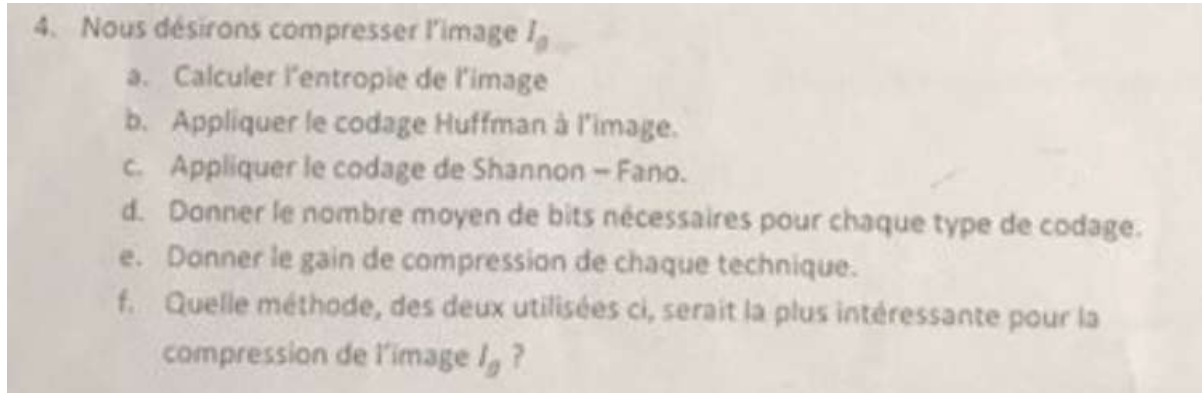


$$f(x,y) = 255 \text{ si } f(x,y) > \text{seuil}$$

$$f(x,y) = 0 \text{ sinon}$$

Nous allons donc prendre le niveau 100 comme seuil.

La matrice résultante : $I_r =$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$


4 – a/

Entropie	$\sum_i \sum_j c_{ij} \log(c_{ij})$
----------	-------------------------------------

b/ Huffman : l'arbre puis le tableau

Couleur	Code
100	0
70	100
170	101
200	110
220	111

c/ Shannon-Fano

Couleur	Fréquence	Code
100	10	0
70	3	10
170	1	110
200	1	1110
220	1	1111

d/ Nombre de bits moyen (Huffman) = $(10 \cdot 1 + 3 \cdot 2 + 1 \cdot 3 + 1 \cdot 4 + 1 \cdot 4) / 16 = 1.6875$

Nombre de bits moyen (Shannon-Fano) = $(10 \cdot 1 + 3 \cdot 3 + 1 \cdot 3 + 1 \cdot 3 + 1 \cdot 3) / 16 = 1.75$

e/ Gain de compression (Huffman) = $1 - T = 1 - (1.6875/8) = 0.7890625$

Gain de compression (Shannon-Fano) = $1 - T = (1.75/8) = 0.78125$

f/ La méthode qui propose une taille d'image minimale est la meilleure. Dans ce cas : Huffman.

Exercice 2 (TD3)

Exercice 2

Soit une image représentée par la figure 2.

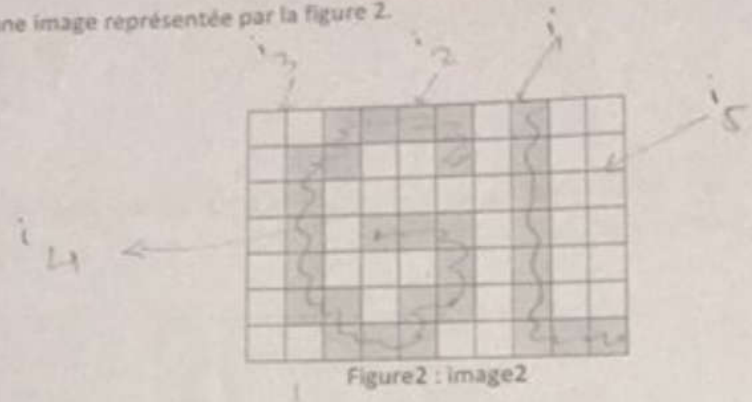


Figure 2 : image2

Nous faisons une segmentation par région de l'image de la figure 2. Le prédicat p considéré est

$p = \text{couleur identique}$

1. Donnez le résultat de cette segmentation ; à noter chaque région par un label i avec $i \in \mathbb{N}$.
2. Quel est le nombre de segments obtenus ?

5 segments

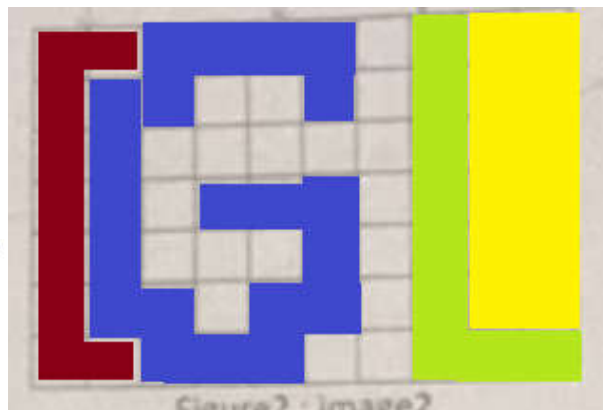
Il faut que l'union des segments donne l'image initiale.

1 – Algorithme du labyrinthe

(Pas de connexion spécifiée => connexité 4)

■ Croissance récursive : l'algorithme du labyrinthe

- ▶ On part d'un point de l'image, en le marquant comme faisant partie d'une région
- ▶ On vérifie pour chaque voisin s'il satisfait le critère d'homogénéité et s'il n'a pas déjà été visité
 - Si oui : on applique la même procédure au voisin
 - Si non, on arrête la procédure
- ▶ Cet algorithme assure de toucher tous les points d'un domaine connexe qui vérifient le critère d'homogénéité
- ▶ Méthode puissante et simple
- ▶ Méthode assez lourde en ressources machine, spécialement lorsque la récursivité est mal supportée



2 – 5 segments.

Exercice 3 (TD)

Exercice 3

Soit $H(i,j)$ le résultat de la combinaison de deux vecteurs $H1$ et $H2$ définis par :

$$H1 = [1 \ 0 \ -1]$$

$$H2 = [1 \ 1 \ 1]$$

$$H = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$

On considère l'image A donnée par :

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

Calculer $A*H$, $(A*H1)*H2^T$, $(A*H2)*H1^T$.

Quelle est votre conclusion.

$$A*H = (A*H1)*H2^T = (A*H2)*H1^T$$

Appliquer un filtre 2D à une image A , est équivalent à appliquer un filtre 1D pour chacune des dimensions de A .

Exercice 4 (TD3)

Exercice 4

Soit S un message codé sur 8 bit.

On a utilisé une méthode de compression sans perte pour coder S. On donne ici la table du codage et le message S' obtenu après compression.

Table de codage

Symbole	Code
H	101
I	110
	001
N	1001
D	10000
E	10001
P	111
A	01
T	000

S' = 101110001101110100110000100010011110100001000110001111010000100001
 H I H I N D E P A T A T I P A T A T A
 Donner le message S initial.

Exercice 4 (TD1)

1/

Exercice 4

Soit l'image couleur suivante représentée par ses composantes couleur R, G et B pour chaque pixel.

formule générale
 $I = aR + bG + cB$
 avec $a + b + c = 1$

0	128	255
0	128	255
0	128	255
255	0	0
0	255	0
0	0	255
255	255	0
255	0	255
0	255	255

On calcule la moyenne de chaque rangée
 exemple : chaque pixel par son moyen

0	128	255
85	85	85
170	170	170

1. Convertir cette image couleur en une image en niveaux de gris.
 2. Convertir l'image couleur de l'espace (R, G, B) vers l'espace couleur (C1, C2, C3).
- La matrice de passage entre (R, G, B) et (C1, C2, C3) est définie par :

$$\begin{bmatrix} 1/3 & 1/3 & 1/3 \\ \sqrt{3}/2 & -\sqrt{3}/2 & 0 \\ -1/2 & -1/2 & 1 \end{bmatrix}$$

- 1- $I_g = [[0, 128, 255],$
 $[85, 85, 85],$
 $[170, 170, 170]]$

2- Formule générale : $I = aR + bG + cB$ avec $a + b + c = 1$

$$\begin{pmatrix} C1, \\ C2, \\ C3, \end{pmatrix} = P \times \begin{pmatrix} R, \\ G, \\ B \end{pmatrix}$$

Finalement on trouve :

$$C1 = R/3 + G/3 + B/3$$

$$C2 = (\sqrt{3}/2)R - (\sqrt{3}/2)G$$

$$C3 = -R/2 - G/2 + B$$

3. On désire avoir des valeurs de $C1$, $C2$ et $C3$ comprises entre 0 et 255, et ce, quel que soit le triplet (R, G, B) considéré. Recadrer l'image obtenue afin que les valeurs de $C1$, $C2$ et $C3$ soient comprises entre 0 et 255.

3- En générale : $g(x,y) = ((G_{\max} - G_{\min})/(G - g)) * f(x,y) + (G * G_{\min} - g * G_{\max})/(G - g)$
Avec $[g, G]$ l'ancienne dynamique et $[G_{\min}, G_{\max}]$ la nouvelle dynamique.

Pour $C1$:

$$G_{\max} = 255, G_{\min} = 0, G = 255, g = -255$$

$$\text{Donc } C'1 = C1/2 + 255/2$$

Pour $C2$:

$$G_{\max} = 255, G_{\min} = 0, G = \sqrt{3} * 255, g = -\sqrt{3} * 255$$

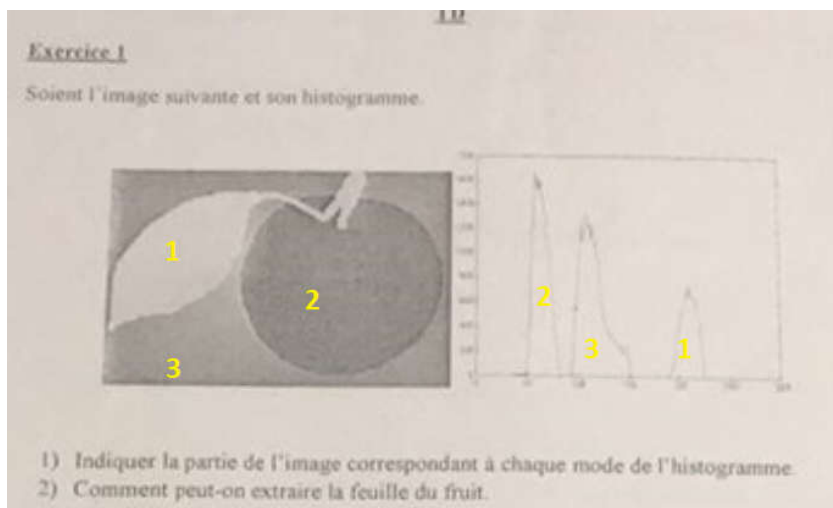
$$\text{Donc } C'2 = (\sqrt{3}/6) * C2 + 255/2$$

Pour $C3$:

$$G_{\max} = 255, G_{\min} = 0, G = 255, g = -255$$

$$\text{Donc } C'3 = C3/2 + 255/2$$

Exercice 1 (TD)



2/ Il suffit d'utiliser un filtre passe bas.

* Aperture = 2,54 cm

* Poids (octets) = Nbre total pixels * codage couleurs

* Résolution (ppi) = $\frac{\text{Définition}}{\text{Dimension Réelle}}$ (octets)

* [Définition (L x E)] = taille de l'image
= nbre total pixels.

* Codage couleurs = Profondeur (octets) * Base (RGB = 3)

* Résolution de la caméra = Def de photo.

* Luminance = moyenne

* Contraste = écart-type

* Amélioration de l'image: qualité:

1- Normalisation de l'histogramme.
[Contraste n'affecte pas le résultat]

Etapes: ① Calcul de l'histogramme: $h(i)$

② Normalisation de l'hist: $h_n(i) = \frac{h(i)}{N_p}$

③ Hist. cumulatif normalisé: $C(i) = \sum_{j=0}^{N_p} h_n(j)$

④ Trans. des niveaux de gris:

$$a) f'(x,y) = C(f(x,y)) * (2^L - 1)$$

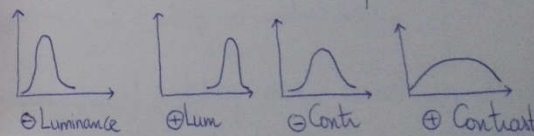
(avec image codée sur L bits)

b) Extension linéaire de la dynamique

• Dynamique: nbre de vl de gris possibles (n bits)
= [val_min; val_max]

$$f'(x,y) = \frac{G_{max} - G_{min}}{G - g} * f(x,y) + \frac{G * G_{min} - g * G_{max}}{G - g}$$

(avec [G,g] l'ancienne dynamique)



• Négatif: $f'(x,y) = (2^L - 1) - f(x,y)$

* Diminution du contraste => photo condensée.

* Contraste fort => photo étirée

* Les pixels blancs deviennent blancs => décalage vers droit.

* ~ ~ ~ nombres deviennent noirs => ~ ~ ~ gauche

* Pour débiter une image => utiliser un filtre.

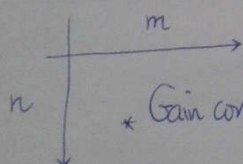
* Filtrage linéaire (convolution):

1- Rotation de π % non centre.

2- Centrer le filtre sur $P(x,y)$

3- Effectuer la somme pondérée

$$g(x,y) = \left[\sum \sum h(i,j) * f(x-i, y-j) \right] \% (2^L - 1)$$



$$* \text{Gain continu en filtre} = \sum \sum h(i,j)$$

* Effets de bord: * zero-padding (bandes noires)

* Duplication (clumping)