



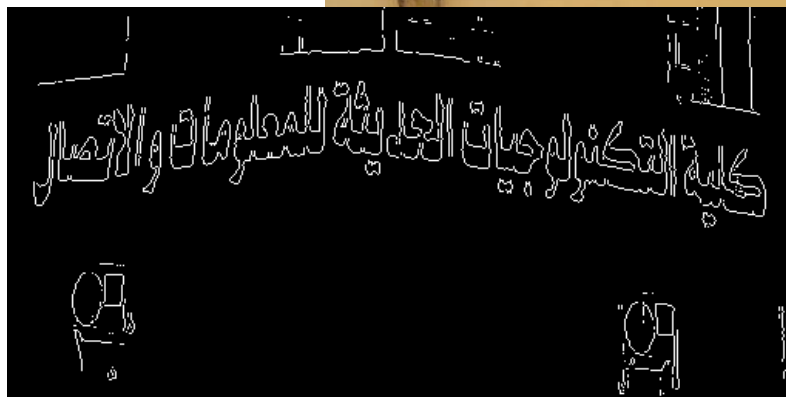
UNIVERSITE KASDI MERBAH D'OUARGLA
**Faculté des Nouvelles Technologies de l'Information
et de la communication**
Département d'Electronique et Des Télécommunications



Cours de traitement d'images

Conformément au programme
Master 1 : Systèmes de Télécommunication

Dr CHARIF Fella



E-mail : cherif.fe@univ-ouargla.dz

Septembre 2017

Semestre : 2
Unité d'enseignement : UEM 1.2
Matière 4 : Traitement d'images
VHS : 37h30 (Cours: 1h30, TP : 1h00)
Crédits : 3
Coefficient : 2

Objectifs de l'enseignement :

Comprendre les concepts de la capture et la numérisation des images. Connaitre les différents paramètres et formats d'images numériques. Maîtriser les fondements de base de l'analyse d'images. Apprendre à utiliser les outils préliminaires en traitement numérique d'images de bas niveau avec une introduction aux traitements de haut niveau.

Connaissances préalables recommandées :

Traitement du signal.

Contenu de la matière :

Chapitre 1. Perception de la couleur (2 Semaines)

- Colorimétrie. Lumière et couleur dans la perception humaine
- Systèmes de représentation de la couleur : rgb, xyz, yuv, hsv, yiq
- Formats couleur et stratégies de traitement de l'image couleur

Chapitre 2. Capteurs d'images et dispositifs d'acquisition numérique (2 Semaines)

- Schéma de principe d'une chaîne de traitement d'images.
- Principe des capteurs CCD et CMOS
- Spécifications des capteurs couleur
- Numérisation d'une image
- Notions de définition, résolution et quantification d'une image numérique (taille, dpi, ppi, bpp ...etc)
- Exemples de formats d'images numériques (BMP, TIFF, JPG, GIF et PNG)

Chapitre 3. Traitements de bases sur l'image (3 Semaines)

- Notion d'histogramme et de contraste
- Correction de la dynamique de l'image par les transformations affines sur l'histogramme
- Egalisation d'histogramme et correction gamma
- Opérations logiques et arithmétiques sur les images

Chapitre 4. Filtrage numérique des images (2 Semaines)

- Filtrage spatial et Convolution 2D : notion de masque (moyenneur, gaussien, binomial ...etc)
- Lissage linéaire puis non linéaire de l'image (médian ...etc)
- Filtrage fréquentiel : (FFT 2D et propriété de séparabilité, filtre passe-bas, passe-haut ...etc)

Chapitre 5. Détection de contours (3 Semaines)

- Objectifs et généralités
- Types de contours
- Dérivées 1^{ère} : masque de convolution (Opérateurs de gradient : masque de Roberts, Prewitt, Sobel ...etc)
- Dérivées 2^{ème} d'une image (Opérateurs Laplacien, Filtre de Marr-Hildreth)
- Opérateurs Laplacien vs Opérateur de gradient (sensibilité aux bruits, localisation ...etc)
- Filtre optimal (critères d'optimalité, Canny et Derriche ...etc)

Chapitre 6. Segmentation et classification (3 Semaines)

- Principe et différentes approches de segmentation (par seuillage, par régions, approche de la classification ...etc)
- Seuillage d'images : seuillage global, seuillage locale, seuillage par détection de Vallées, seuillage dynamique, seuillage par minimisation de variance, méthodes de classification bayésienne ...etc
- Opérations morphologiques (dilatation, érosion, ouverture, fermetureetc)
- Extraction de paramètres et classification d'objets (distance Euclidienne, Kppv ...etc)

Avant-propos

Ce cours sur le '*Traitement d'images*' s'adresse aux étudiants du master système de télécommunications et tous ceux qui ont à pratiquer cette discipline. Il présente les principales techniques rencontrées dans le traitement d'images. Différents concepts y sont abordés : l'histogramme, le filtrage, la détection de contours, les opérateurs morphologiques et la recherche de primitives pour la classification d'objets. Le cours s'appuie sur de nombreux exemples illustratifs permettant une meilleure compréhension du sujet.

Le cours est organisé en six chapitres:

- ❖ Perception de la couleur.
- ❖ Capteurs d'images et dispositifs d'acquisition numérique
- ❖ Traitements de bases sur l'image
- ❖ Filtrage numérique des images
- ❖ Détection de contours
- ❖ Segmentation et classification

Le contenu de ce polycopié est conforme au programme du module *Traitement d'images* établi par le ministère de l'enseignement supérieur et de la recherche scientifique.

Sommaire

Chapitre 1 Perception de la couleur

1.1. Colorimétrie.	1
1.2. Lumière et couleur dans la perception humaine	1
1.2.1. Lumière	2
1.2.2. Objet	2
1.2.3. L'œil humain	3
1.3. Synthèse de la couleur	3
1.3.1. Synthèse additive	3
1.3.2. Synthèse soustractive	3
1.4. Systèmes de représentation de la couleur	4
1.4.1. Espace RVB / RGB	4
1.4.2. Espace HLS / HSV	5
1.4.3. Espace XYZ	6
1.4.4. Espace YUV	6
1.4.5. Espace CMJN / CMYN	7
1.5. Stratégies de traitement de l'image couleur	8
1.5.1. Stratégie marginale	8
1.5.2. Stratégie vectorielle	8

Chapitre 2 Capteurs d'images et dispositifs d'acquisition numérique

2.1. Schéma de principe d'une chaîne de traitement d'images	9
2.1.1. Acquisition d'une image	9
2.1.2. Prétraitements	9
2.1.3. Traitement d'images numériques	10
2.2. Principe des capteurs CCD et CMOS	10
2.2.1. Caméras CCD	10
2.2.2. Caméras CMOS	10
2.3. Spécifications des capteurs couleur	10
2.4. Numérisation d'une image	11
2.4.1. Échantillonnage spatial	11
2.4.2. Quantification	11
2.4.3. Définition de l'image	12
2.4.4. Résolution spatiale et résolution tonale d'une image	13
2.5. Notion de définition et résolution	14
2.5.1. Notion de définition	14
2.5.2. Résolution	14
2.6. Types d'images	15
2.6.1. Types d'images selon la couleur	15
2.6.2. Types d'images selon la dimension	16
2.7. Exemples de formats d'images numériques	16
2.8. Types des traitements d'images	17
2.8.1. Traitement ponctuel	17
2.8.2. Traitement local	17
2.8.3. Traitement global	17

Chapitre 3 Traitements de bases sur l'image

3.1. Notion d'histogramme et de contraste	18
3.1.1. Histogramme	18
3.1.2. Histogramme normalisé	20
3.1.3. Histogramme cumulé	20
3.1.4. Informations issues de l'histogramme	21
3.2. Correction de la dynamique de l'image par les transformations affines sur l'histogramme	22
3.2.1. Etirement d'histogramme	22
3.2.2. Masquage des zones	23
3.2.3. Extraction d'une fenêtre d'intensité	24
3.2.4. Dilatation locale	24
3.3. Egalisation d'histogramme et correction gamma	25
3.3.1. Egalisation d'histogramme	25
3.3.2. Correction gamma	28
3.4. Opérations arithmétiques et logiques sur les images	28
3.4.1. Opérations arithmétiques	28
3.4.2. Opérateurs logiques	30

Chapitre 4 Filtrage numérique des images

4.1. Introduction sur le filtrage numérique	31
4.2. Filtrage spatial et convolution 2D	31
4.2.1. Voisinage	32
4.2.2. Produit de convolution	32
4.2.3. Traitement des points de bordure	34
4.2.4. Séparabilité d'un filtre de convolution	35
4.2.5. Classification des filtres	35
4.3. Filtres de Lissage	36
4.3.1. Filtres linéaires	36
4.3.2. Filtres non linéaires	38
4.4. Filtrage fréquentiel (spectral)	40
4.4.1. Représentation fréquentielle d'une image	40
4.4.2. Transformée de Fourier 2D	40
4.4.3. Interprétation de la Transformée 2D	41
4.4.4. Classification de filtrage fréquentiel	42

Chapitre 5 Détection de contours

5.1. Objectifs et généralités	45
5.1.1. Contour	45
5.1.2. Détection de contours	45
5.2. Différents types de contour	45
5.3. Filtres différentiels	46
5.3.1. Opérateurs de gradient par masques	47
5.3.2. Filtre Laplacien	49
5.4. Opérateurs du premier ordre (Approche gradient)	49
5.4.1. Calcul de la norme du gradient	50
5.4.2. Seuillage	52

5.5. Opérateurs de second ordre	54
5.5.1. Opérateurs Laplacien	54
5.5.2. Filtre de Marr-Hildreth	54
5.6. Opérateurs Laplacien vs Opérateur de gradient	55
5.7. Filtre optimal	55
5.7.1. Critères d'optimalité	55
5.7.2. Filtre de Canny	55
5.7.3. Filtre de Deriche	56

Chapitre 6 Segmentation et classification

6.1. Principe de segmentation	58
6.2. Techniques de segmentation	58
6.2.1. Approche contour	58
6.2.2. Approche région	58
6.2.3. Segmentation d'images par classification	59
6.2.4. Segmentation par seuillage	59
6.3. Seuillage d'images	60
6.3.1. Seuillage globale	60
6.3.2. Seuillage local	61
6.3.3. Seuillage par détection de vallées	61
6.3.4. Seuillage par minimisation de variance	62
6.4. Morphologie mathématique	62
6.4.1. Élément structurant	62
6.4.2. Operations morphologiques de base	63
6.5. Extraction de paramètres et classification d'objet	65
6.5.1. Extraction des paramètres	65
6.5.2. Méthodes de classification	68
Bibliographie	69

Chapitre 1

Perception de la couleur

1.1. Colorimétrie

La colorimétrie est la science qui permet de déterminer les propriétés d'un objet en fonction de sa couleur. C'est un ensemble des techniques qui permettent la définition, la mesure, la comparaison et d'une manière générale, l'étude des couleurs.

Plusieurs paramètres pour la couleur :

- ❖ **Teinte** (Hue) : déterminée par la longueur d'onde dominante (rouge, jaune, pourpre, ...).
- ❖ **Saturation** : proportion de couleur chromatiquement pure contenue dans la couleur.
- ❖ **Luminance** (Lightness) ou clarté : intensité de la lumière que la couleur émet ou réfléchit.

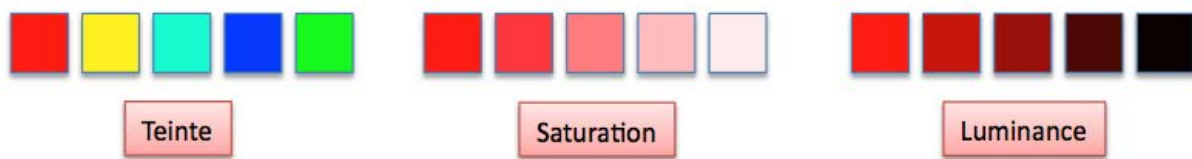


Figure 1.1. Paramètres de la couleur

1.2. Lumière et couleur dans la perception humaine

La notion de couleur est subjective, elle résulte de la perception de la lumière des différentes longueurs d'onde qui constituent la lumière visible par le système visuel humain.

La conception de la couleur découle de la notion basée sur un triplet formé de :

1. La source lumineuse qui éclaire l'objet.
2. L'objet que l'on observe.
3. l'œil et le cerveau de l'observateur.

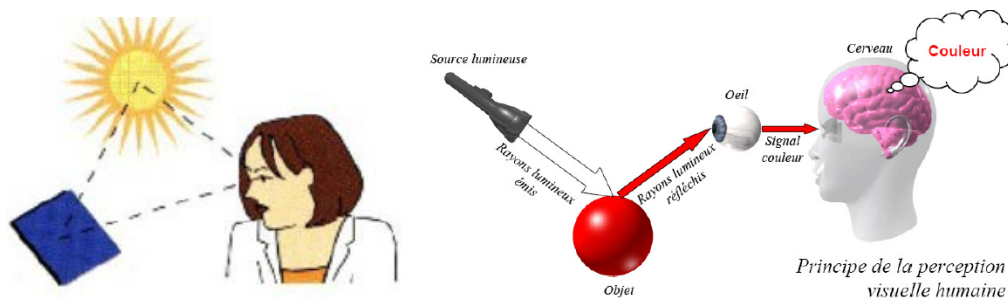


Figure 1.2. Principe de la perception humaine

Puisque c'est le flux lumineux (la lumière) qui active l'œil, il constitue le premier élément fondamental pour la vision des couleurs. L'œil en est le second. Un troisième élément intervient : c'est l'objet qui réfléchit plus ou moins l'énergie provenant de la source (on ne voit pas les objets mais la lumière qu'ils *réflètent* ou *transmettent*).

1.2.1. Lumière

Les longueurs d'onde du spectre visible s'étendent de 380 à 780 nm.

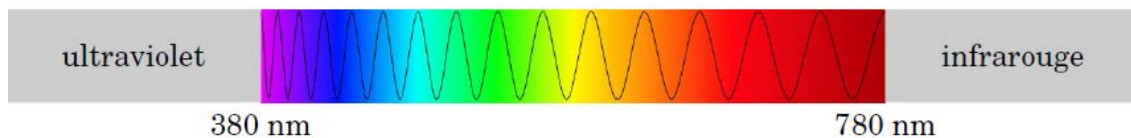


Figure 1.3. Longueur d'onde du spectre visible

Ce spectre s'étale du violet à partir de 380 nm à 450 nm, au bleu de 450 à 490 nm, au vert de 500 nm à 570 nm, au jaune de 570 nm à 590 nm, à l'orange de 590 nm à 650 nm, au rouge de 650 nm à 780 nm.

Pour caractériser une lumière monochromatique, qui peut être détectée par l'œil humain, on doit connaître sa longueur d'onde λ et sa luminance L .

1.2.2. L'objet

La lumière qui éclaire un objet sera modifiée de multiples façons et dans des directions diverses. La résultante de cette distribution de lumière nous donne l'impression visuelle que nous avons de ce produit.

L'absorption de la lumière par l'échantillon

La transmission : la lumière traverse l'échantillon

La réflexion : la lumière est réfléchiée sur la surface de l'échantillon dans une direction qui est telle que l'angle de réflexion est égale à l'angle d'incidence.(réflexion spéculaire)

La dispersion : la lumière se réfléchit sur la surface dans toutes les directions.

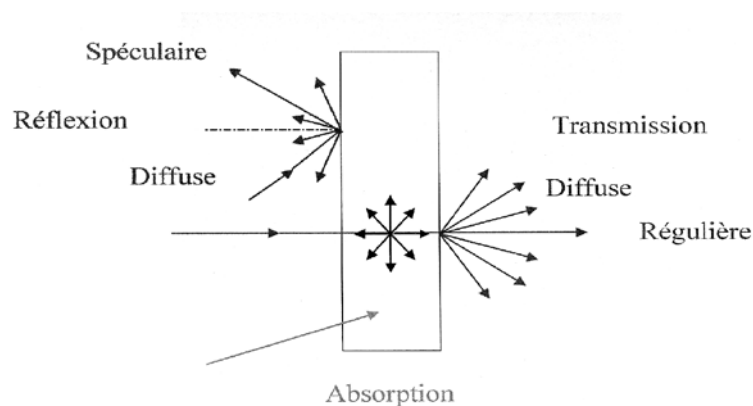


Figure 1.4. Dispersion de la lumière à travers un objet

Ainsi, un corps qui absorbe toutes les radiations nous apparaît noir et un corps qui les réfléchit toutes (et donc qui n'en absorbe aucune) nous paraît blanc.

Les radiations qui ne sont pas absorbées sont réfléchies ou transmissent par l'objet et donc visibles par les observateurs.

1.2.3. L'œil humain

La perception visuelle est le résultat de l'interprétation faite par le cerveau, de la lumière perçue par les yeux, lumière modifiée et transmise par les objets ou directement émises par les sources de lumière. L'œil n'est pas sensible de la même façon à toutes les radiations du spectre visible.

1.3. Synthèse de la couleur

La synthèse de la couleur consiste à reproduire l'ensemble des couleurs visibles à partir d'un petit nombre de couleurs, appelées couleurs primaires. On distingue deux types de synthèse, la synthèse additive et la synthèse soustractive.

1.3.1. Synthèse additive

La synthèse additive est la construction des couleurs par addition de sources lumineuses comme les télévisions, les écrans...

Les 3 couleurs primaires de la synthèse additive sont le rouge (R), le vert (V) et le bleu (B). L'absence de lumière ($R=V=B=0$) donne le noir.

La somme des 3 couleurs primaires $R+V+B$ donne le blanc. Les couleurs secondaires sont définies par l'addition de 2 couleurs primaires:

- Rouge + Vert = Jaune
- Vert + Bleu = Cyan
- Bleu + Rouge = Magenta

Les autres couleurs sont obtenues en faisant varier les intensités respectives des 3 primaires.

1.3.2. Synthèse soustractive

La synthèse soustractive est la construction des couleurs à partir de pigments colorés comme la peinture, les imprimantes. Plus on ajoute de pigments colorés, plus la lumière est absorbée et la couleur obtenue est sombre. Les 3 couleurs primaires de la synthèse soustractive sont le Cyan (C), le Magenta (M) et le Jaune (J). Chacune absorbe une des couleurs primaires de la lumière :

- Le Cyan absorbe le Rouge
- Le Magenta absorbe le Vert
- Le Jaune absorbe le Bleu
- L'absence de pigment donne le blanc.

- La somme des 3 primaires C+M+J donne le noir.

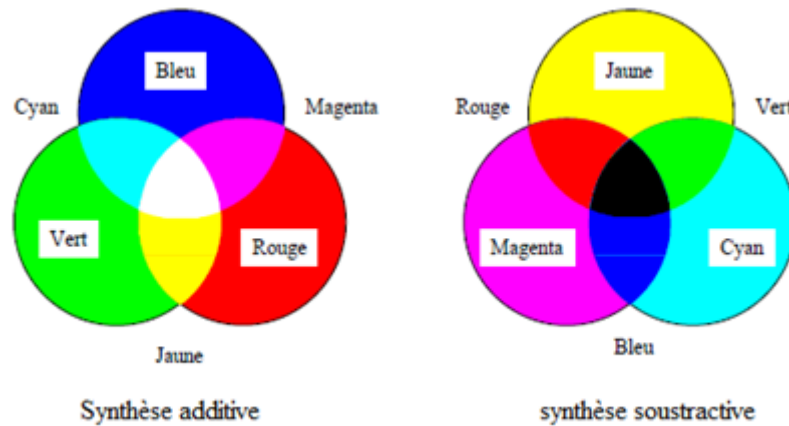


Figure 1.5. Synthèse de la couleur

1.4. Systèmes de représentation de la couleur

Les systèmes de représentation des couleurs ont pour objectif de représenter la couleur perçue par un ensemble de points dans un espace à 3 dimensions.

Il existe plusieurs systèmes de représentation des couleurs ou espaces de couleur dont les principaux d'entre eux sont :

1.4.1. Espace RVB / RGB

Espace le plus couramment utilisé. L'espace RVB (Rouge Vert Bleu) ou RGB (Red Green Blue) est basé sur la synthèse additive. La télévision, les caméras et les moniteurs des ordinateurs utilisent ce système lors de l'affichage et de l'acquisition des images. Chaque couleur est représentée par 3 valeurs qui sont les quantités de rouge, de vert et de bleu qu'elle contient. Les valeurs sont comprises entre 0 et 1 (valeurs normalisées), ou 0 et 255 lors du codage des couleurs sur 8 bits. Par exemple : (0,0,0) = noir, (1,1,1) = blanc, (1,0,0) = rouge, (0,1,0) = vert, (0,0,1) = bleu, (0,1,1) = cyan, (1,0,1) = magenta, (1,1,0) = jaune.

On représente souvent l'espace RVB comme un cube de dimensions 1x1x1, placé à l'origine d'un repère à trois dimensions dont les axes indiquent la quantité de Rouge, de Vert et de Bleu. Chaque couleur correspond à un point C de coordonnées (Rc, Vc, Bc) situé dans le cube. Les couleurs Rouge, Vert, Bleu, Cyan, Magenta, Jaune, Noir et Blanc sont situées aux sommets du cube. Le Noir (0,0,0) est situé à l'origine du repère.

Les couleurs primaires sont situées sur les 3 axes à la distance 1. La diagonale principale du cube est une droite tracée entre l'origine (0,0,0) et le point (1,1,1). Les couleurs situées sur cette ligne ont des quantités égales de Rouge, de Vert, et de Bleu. Ce sont donc les différents niveaux de gris, variant du Noir en (0,0,0) au Blanc en (1,1,1). Difficile de se faire une idée de la luminosité dans ce mode colorimétrique :

$$L = 0.3R + 0.59G + 0.11B$$

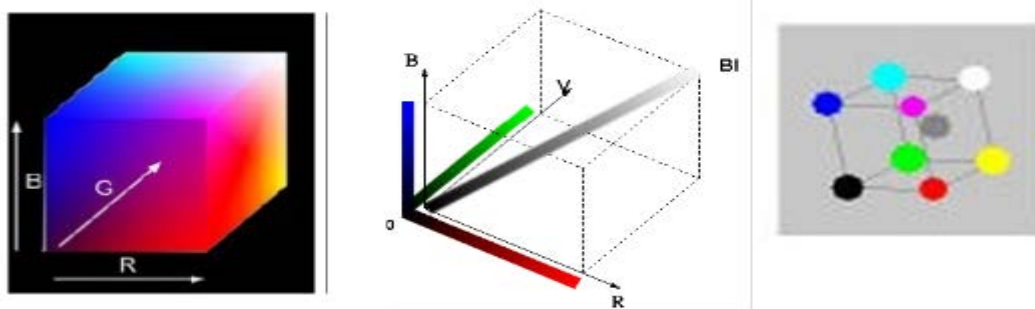
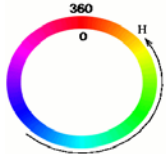
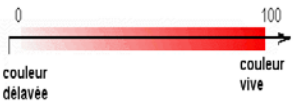



Figure 1.6. Espace RVB

1.4.2. Espace HLS / HSV

TLS (Teinte, Luminance, Saturation) ou HSV (Hue, Saturation, Value) ou HLS (Hue, Luminance, Saturation) espace qui permet de décomposer une couleur en trois composantes plus intuitives qui sont la teinte (Hue), la saturation et la luminance(Value). Il se base sur la décomposition de la couleur en une couleur pure du spectre de l'arc en ciel à laquelle est ajoutée un gris (addition de R,V,B à parts égales).

La teinte indique la couleur pure qui est prépondérante dans la couleur. Les teintes peuvent-être représentées par un angle sur un cercle.	
La saturation correspond à la proportion de couleur pure. Elle varie entre 0 et 1. S=0 : gris S=1 : couleur pure Plus la saturation est faible plus la couleur est délavée. Plus la saturation est élevée, plus la couleur est vive.	
La luminance caractérise l'intensité totale du point lumineux coloré.	

L'ensemble des couleurs représentables dans l'espace HLS est délimité par deux cônes ayant la même base. Pour une luminance donnée les couleurs sont situées sur un disque.

Si on fixe une teinte, et qu'on fait varier la saturation, on se déplace sur un rayon du disque. La surface du cône contient l'ensemble des couleurs pures. L'axe du cône contient les gris du noir au blanc.

Les formules de passage entre l'espace HLS et l'espace RVB :

$$L = \frac{1}{3}(R + G + B)$$

$$S = 1 - \frac{3}{(R + G + B)} \min(R, G, B)$$

$$H = \arccos\left(\frac{0.5(R - G) + 0.5(R - B)}{\sqrt{(R - G)^2 + (R - B)(G - B)}}\right)$$

$$H = 360^\circ - H \quad \text{si } B > G$$

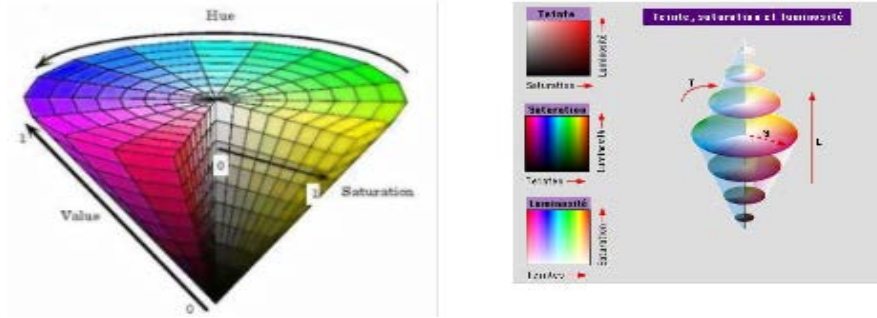


Figure 1.7. Espace HSV

1.4.3. Espace XYZ

C'est un espace qui permet de représenter toutes les couleurs et qui distingue la luminance de la chrominance. Il est défini à partir d'une transformation linéaire sur l'espace (RGB) telle que toutes les couleurs du spectre visible soient contenues dans le triangle (XYZ) par :

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 2.76 & 1.75 & 1.13 \\ 1 & 4.59 & 0.06 \\ 0 & 0.05 & 5.95 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Toutes les couleurs visibles peuvent s'exprimer comme l'addition de 3 composantes positives X, Y, et Z.

- ✓ Y ne contient que l'information de luminance perçue : on additionne R, V, B avec des proportions de 30%, 59%, 11% qui tiennent compte de la sensibilité de l'œil, bien plus importante pour le vert que pour le rouge et le bleu.
- ✓ Les gris correspondent aux points tels que X=Y=Z.

1.4.4. Espace YUV

Espace qui distingue la luminance de la chrominance. Ce système est principalement utilisé pour la transmission des signaux vidéos. Son principe est basé sur l'utilisation d'une composante de luminance Y, et 2 composantes de chrominance (U, V) correspondant aux composantes bleue et rouge dans les coordonnées chromatiques réduites.

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.2989 & 0.5866 & 0.1145 \\ -0.1687 & -0.3312 & 0.5 \\ 0.5 & -0.4183 & -0.0817 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Dans l'espace YUV, les informations de luminance et de chrominance sont séparées.

Y porte l'information de luminance. Y est issue du système XYZ.

U et V (ou Cb, Cr) portent les informations de chrominance.

$$U = Cr = R - Y$$

$$V = Cb = B - Y$$

- ✓ Permettre l'envoi d'un même signal pour les télévisions N&B et couleur. Y peut être directement affichée sur un poste noir et blanc.
- ✓ Diminuer la quantité d'information à transmettre : l'œil n'étant pas sensible à toutes les nuances de couleur, et surtout les nuances de bleu, les composantes U et V peuvent être codées sur moins de bits.

1.4.5. Espace CMJN / CMYN

L'espace CMJN (Cyan Magenta Jaune Noir) / CMYK (Cyan Magenta Yellow Black) est basé sur la synthèse soustractive des couleurs. Il est utilisé spécifiquement dans l'imprimerie. Pour chaque couleur, on indique la quantité d'encre Cyan, Magenta, Jaune et Noir permettant de la reproduire.

On peut simplement passer de l'espace RVB à l'espace CMJN :

$$C = 1 - R, \quad M = 1 - G, \quad J = 1 - B$$

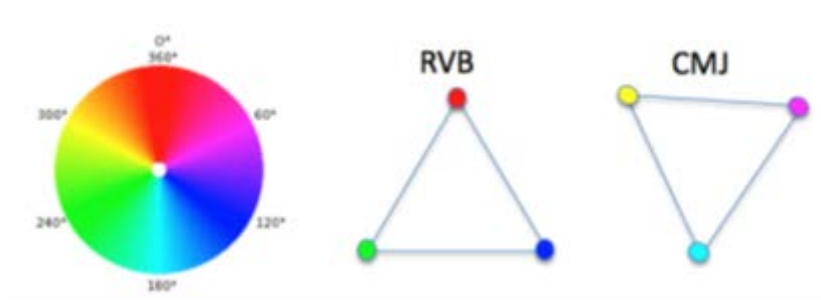


Figure 1.8. Espace CMJ

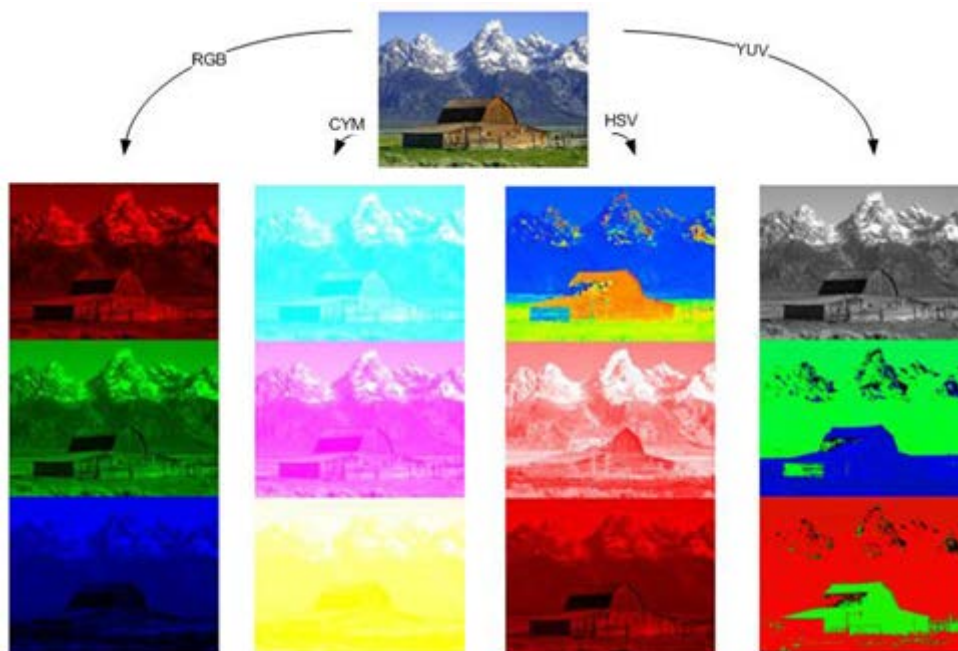


Figure 1.9. Une image et sa représentation dans différents espaces colorimétriques.

1.5.Stratégies de traitement de l'image couleur

On peut distinguer deux grands types de stratégies dans le traitement d'images couleurs :

- les stratégies marginales (ou scalaires) où le contexte multi-composante ne sera pas exploité.
- les stratégies vectorielles, élaborées en tenant compte de la nature multi-composantes des images couleur.

1.5.1. Stratégie marginale

Cette stratégie consiste à traiter chaque composante séparément en utilisant des traitements mono-composantes. L'intérêt de cette approche est de n'utiliser que des traitements scalaires. Toutes les méthodes définies en imagerie en niveaux de gris sont alors directement exploitables.

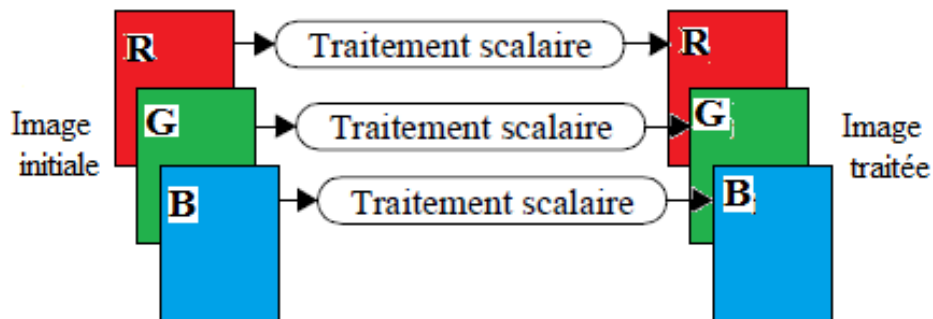


Figure 1.10. Stratégie marginale

1.5.2. Stratégie vectorielle

L'alternative à la stratégie marginale est la stratégie vectorielle qui traite de manière globale l'ensemble des composantes. L'intérêt de cette approche se situe également dans le fait qu'elle ne nécessite qu'un seul traitement, quel que soit le nombre de composantes. Cet avantage doit être pondéré par une complexité accrue de ce seul traitement.

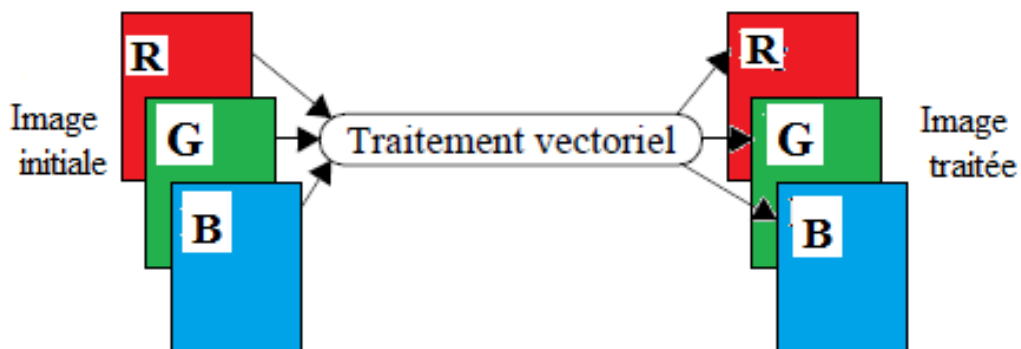


Figure 1.11. Stratégie vectorielle

Chapitre 2

Capteurs d'images et dispositifs d'acquisition numérique

2.1. Schéma de principe d'une chaîne de traitement d'images

Un système de traitement numérique d'images est composé de :

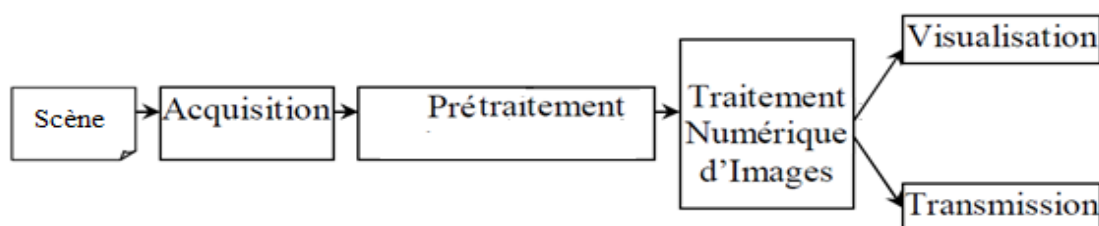


Figure 2.1. Schéma d'un système de traitement d'images

2.1.1. Acquisition d'une image

Pour pouvoir manipuler une image sur un système informatique, il est avant tout nécessaire de lui faire subir une transformation qui la rendra lisible et manipulable par ce système. Le passage de cet objet externe (l'image d'origine) à sa représentation interne (dans l'unité de traitement) se fait grâce à une procédure de numérisation. Ces systèmes de saisie, dénommés optiques, peuvent être classés en deux catégories principales : les caméras numériques, et les scanners. La vitesse *d'acquisition* caractérise le nombre d'images formées par unité de temps.

2.1.2. Prétraitements

Ils préparent l'image pour son analyse ultérieure. Il s'agit souvent d'obtenir l'image théorique que l'on aurait dû acquérir en l'absence de toute dégradation. Ainsi, ils peuvent par exemple corriger :

- ✓ Bruits liés aux conditions de prise de vues : Ce sont des événements vérifiant les conditions d'acquisition du signal : le bougé, problèmes liés à l'éclairage de la scène observée...
- ✓ Bruits liés au capteur : Capteur mal réglé, capteur de mauvaise qualité (distorsion de la gamme des niveaux de gris ou en flou)...

- ✓ Bruits liés à l'échantillonnage : Ces bruits reflètent essentiellement des problèmes de quantification.
- ✓ Bruits liés à la nature de la scène : Nuage sur les images satellitaires, poussières dans les scènes industrielles, brouillard pour les scènes routières...

2.1.3. Traitement d'images numériques

Le traitement d'images numériques est l'ensemble des techniques permettant de modifier une image numérique pour *l'améliorer* ou en *extraire des informations*.

Les différents niveaux de traitement d'image :

1. Traitement de bas niveau : filtrage (linéaire ou non linéaire) et l'amélioration de contraste.
2. Traitement de haut niveau : reconnaissance des formes, reconstruction 3D, calculs de mouvements.

2.2.Principe des capteurs CCD et CMOS

Le rôle du capteur d'image est de transformer l'énergie lumineuse de chaque point de l'original en un signal électrique. La réalisation des capteurs est très variable suivant les performances recherchées; aux plus elles sont élevées, au plus le capteur est complexe et difficile de mise en œuvre.

2.2.1. Caméras CCD (Charged-coupled devices)

Ce sont des Caméra matricielle qui fournissent une image en 2 dimensions de la scène analysée. Elles sont constituées de capteurs solides CCD. Le capteur CCD est une matrice de cellules photosensibles, chaque cellule accumule des charges en fonction de la lumière incidente qu'elle reçoit. Les charges sont ensuite transportées par colonnes sur le capteur. Les caméras CCD sont caractérisées par:

- signal vidéo plus linéaire par rapport à l'intensité lumineuse.
- résolution importante.
- très populaire.

2.2.2. Caméras CMOS (Complementary Metal Oxide Semiconductor)

Le principe est le même que pour les capteurs CCD. Il s'agit d'une matrice de cellules photosensibles. La différence principale est que chaque cellule peut être adressée individuellement, et l'information transmise est la quantité de lumière instantanée et non accumulée. Ces caméras restent moins précises que les CCDs, mais par contre elles sont moins chères que les CCDs.

2.3. Spécifications des capteurs couleur

- Placer un filtre devant le capteur et effectuer trois acquisitions successives (lent).
- Utiliser trois CCDs (cher).

- Utiliser deux CCDs, un pour la chrominance avec un filtre et un autre pour la luminance.



Capteur CCD

Caméra matricielle

Système tri-CCD

Figure 2.2. Capteur CCD , caméra matricielle et système tri-CCD

2.4. Numérisation d'une image

La numérisation des images est une opération qui sert à transformer une image continue en une image discrète. Cette opération s'effectue en deux étapes :

2.4.1. Echantillonnage spatial

Consiste à représenter l'image par un nombre fini de points. Il peut être vu comme une superposition à l'image une grille régulière formée de petits éléments, chacun d'eux va présenter un point de l'image discrète appelé pixel (PICTure ELement). L'échantillonnage spatial détermine la taille de chaque point élémentaire de l'image (pixel). Cette taille est fonction de la résolution du capteur.

2.4.2. Quantification

Consiste à décomposer l'éclairement du pixel en un nombre fini de valeurs appelées niveaux de gris (**niveaux de quantification**). Dans la plupart du temps l'intensité des pixels allant du blanc jusqu'au noir, prend des valeurs sur l'intervalle [0 255].

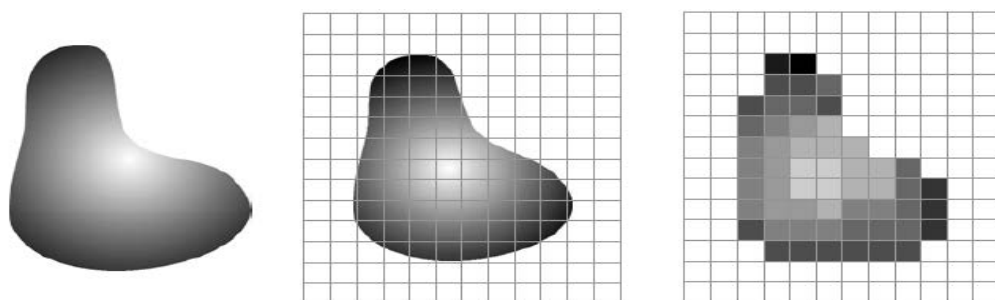


Image réelle

application de la grille d'échantillonnage

image échantillonnée

Figure 2.3. Principe de la quantification

Le résultat de l'échantillonnage et la quantification est une matrice dont ces éléments sont des nombres réels.

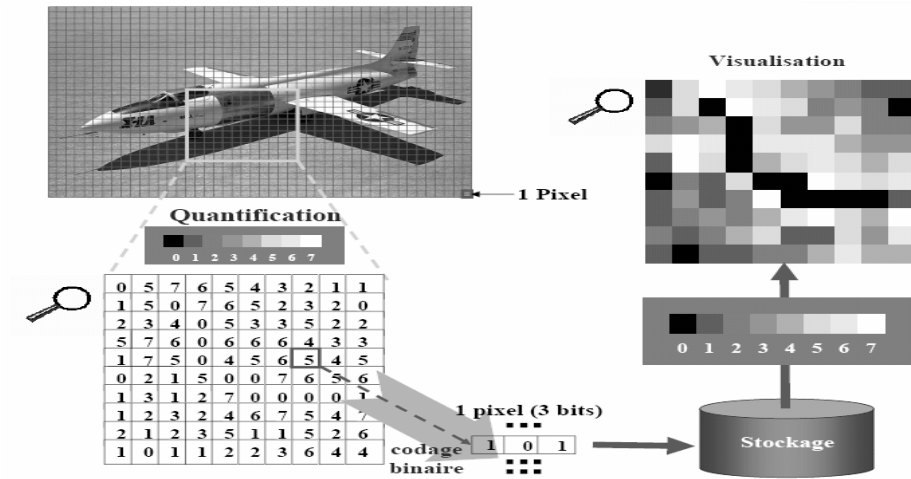


Figure 2.4. Quantification et codage d'une image

2.4.3. Définition de l'image

Une image est une reproduction exacte d'un être ou d'une chose. C'est une forme discrète d'un phénomène continu obtenue après discrétisation. Le plus souvent, cette forme est bidimensionnelle. En traitement de signal, une image est définie comme une fonction discrète $f(x,y)$ à deux dimensions donnant l'intensité lumineuse [intensity, value, luminosity] en chaque points (pixels) de coordonnées spatiales (x,y) .

$$f: M \times N \rightarrow R$$

$$(x,y) \rightarrow f(x,y)$$

Cette image sera représentée par une matrice de dimension $M \times N$ dont les éléments représentent les niveaux de gris, M nombre de lignes et N nombre des colonnes.

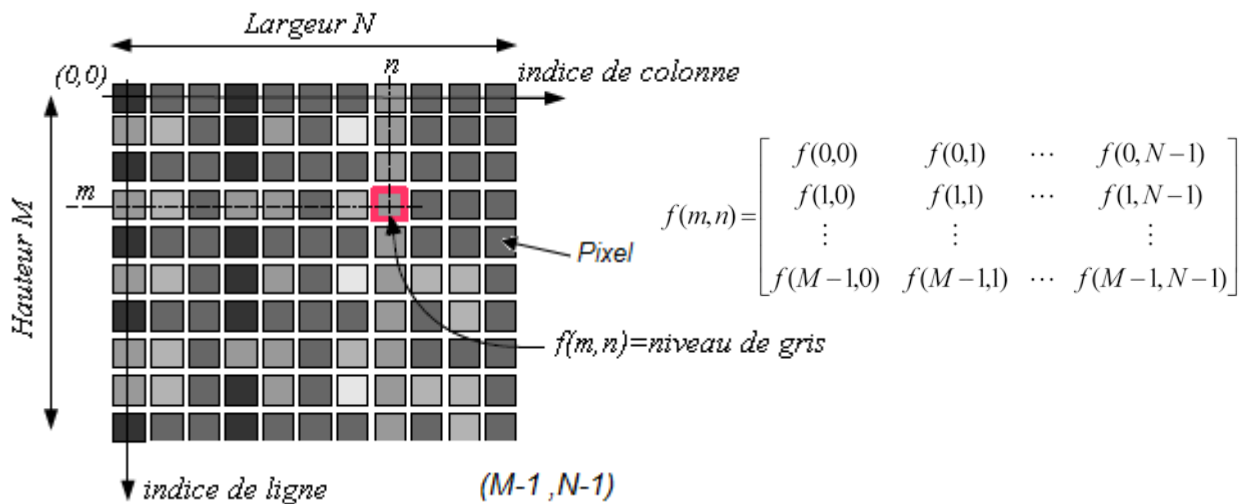


Figure 2.5. Représentation mathématique de l'image

Pixel : Une image est constituée d'un ensemble de points appelés pixels (pixel est une abréviation de PICture ELe ment). Le pixel représente ainsi le plus petit élément constitutif

d'une image numérique. L'ensemble de ces pixels est contenu dans un tableau à deux dimensions constituant l'image.

Niveau de gris : C'est la valeur de l'intensité lumineuse en un point de l'image.

Dynamique de l'image : C'est l'étendue des teintes de gris ou des couleurs que peut prendre chaque pixel.

Taille du fichier image : C'est le nombre de pixels multiplié par le nombre d'octets nécessaires au codage d'un seul pixel. Les images brutes codent en général chaque pixel sur un octet, Par contre les images de très haute qualité sont codées sur 3, 4, voire 6 octets par pixel. Ces fichiers sont donc très volumineux et subissent une compression pour le stockage et la transmission.

Pour une image de taille $M \times N$, où chaque pixel est codé sur k bits, l'espace mémoire nécessaire pour le stockage de cette image est donné par $b = N \cdot M \cdot k$

Pour les images carrées ($N = M$) : $b = N^2 \cdot k$

Le nombre de niveaux de gris est un nombre entier puissance de 2 : $L = 2^k$

Si $k = 8$ ceci donne un niveau de gris qui s'étend entre 0 et 255 dont les valeurs limites représentent le noir et le blanc respectivement, et les valeurs intermédiaires représentent les nuances de gris.

Exemple : Pour une image de taille 512×512 avec 256 niveaux gris ($k = 8$) nous avons besoin de 2.097.152 bits ou 262.144 octets. C'est pourquoi nous essayons souvent de réduire k et N en évitant une perte significative de la qualité de l'image.

2.4.4. Résolution spatiale et résolution tonale d'une image

La résolution d'image s'intéresse à la résolution spatiale et la résolution en niveaux de gris. La résolution spatiale exprime combien de détail nous pouvons voir dans une image, elle dépend de la taille de l'image et le nombre de pixels. En gardant k constant et en diminuant N conduit à un effet de crénelage (damier), les contours des zones homogènes sont crénelés dès que la forme du pixel devient visible.

Pour la résolution en niveaux de gris (résolution tonale), elle consiste à la perception du plus petit changement du niveau de gris. Une réduction de la résolution tonale entraîne une perte d'information dans les images en conduisant à de faux contours : les zones de faible variation d'intensité apparaissent comme des plages homogènes juxtaposées.

L'expérience montre que plus une image est riche en informations (plus détaillée), l'augmentation de la résolution ne donne pas un plus pour la qualité de l'image.

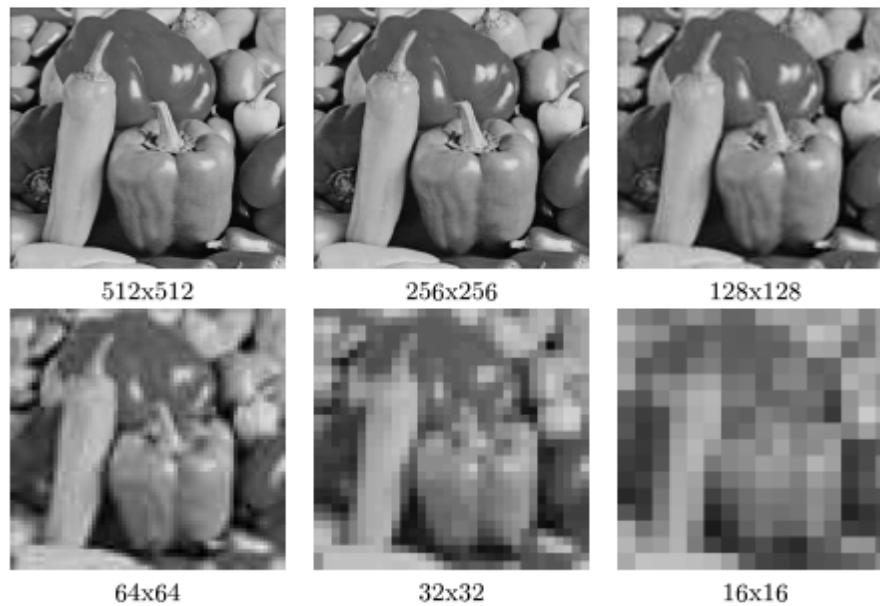


Figure 2.6. Résolution spatiale

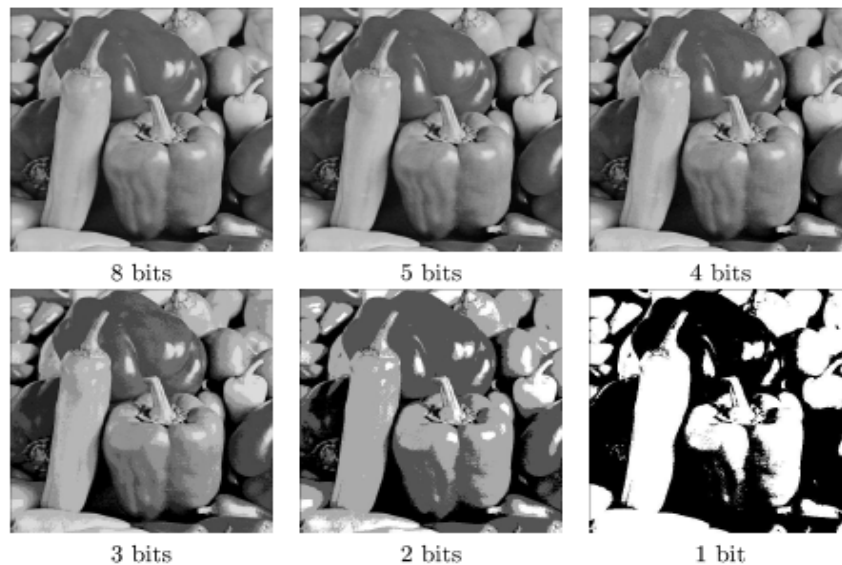


Figure 2.7. Résolution tonale

2.5. Notion de définition et résolution

2.5.1. Notion de définition

On appelle définition le nombre de points (pixels) constituant une image ; c'est le nombre de colonnes de l'image que multiplie son nombre de lignes. Une image possédant 100 colonnes et 110 lignes aura une définition de 100x110.

2.5.2. Résolution

C'est le nombre de points contenu dans une longueur donnée (en pouce). Elle est exprimée en points par pouce (PPP, en anglais: DPI pour Dots Per Inch). Un pouce mesure 2.54 cm, c'est une unité de mesure britannique.

La *résolution* permet ainsi d'établir le rapport entre la *définition en pixels d'une image* et la *dimension réelle* de sa représentation sur un support physique (affichage écran, impression papier...).

$$\text{Résolution} = \text{définition (largeur)} / \text{dimension (largeur)}$$

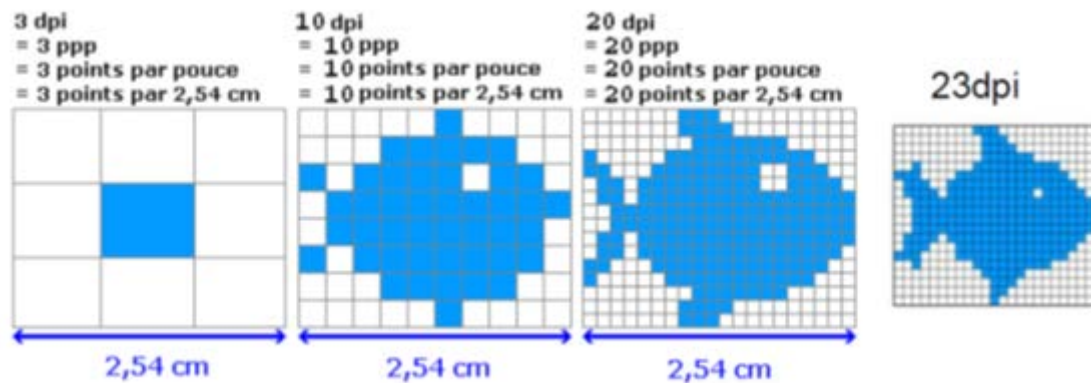


Figure 2.8 Définition d'une image

Exemple : la résolution d'une image de 300 pixels de large mesurant 2 pouces de côté est de $300 / 2 = 150\text{dpi}$

2.6. Types d'images

2.6.1. Types d'images selon la couleur

1. Images binaire : Une image binaire est une matrice rectangulaire dont les éléments valent 0 ou 1. Lorsque l'on visualise une telle image, les zéros sont représentés par des noirs et les uns par des blancs.

2. Image d'intensité : Une image d'intensité est une matrice dont laquelle chaque élément est un réel compris entre zéros et un. On parle aussi d'image en niveaux de gris car les valeurs comprises entre 0 et 1 représentent les différents niveaux de gris.

3. Image couleur RGB : Pour représenter la couleur d'un pixel, il faut donner trois nombre qui correspondent au dosage des trois couleurs de base : rouge, vert et bleu (RGB). On peut ainsi représenter une image couleur par trois matrices, chaque matrice correspondant à une couleur de base.



Image binaire



Image d'intensité



Image RGB

Figure 2.9. Types d'images

2.6.2. Types d'images selon la dimension

1. Bidimensionnels (2D) : Il s'agit dans ce cas d'images. La variable est une variable d'espace représentant les deux coordonnées (x, y) d'un point du plan de l'image.

2. Tridimensionnels (3D) : Il peut s'agir, soit d'images 3D (dans l'espace), soit d'une séquence d'images 2D dans le temps (vidéo). Dans le premier cas, La variable est une variable d'espace représentant les trois coordonnées d'un point (x, y, z) de l'image. Dans le second cas il s'agit des deux coordonnées x, y dans le plan et du temps t .

3. Quadridimensionnels (4D) : C'est le cas, par exemple, d'images 3D (volumes) évoluant dans le temps.

Définition d'une séquence vidéo 2D : Une séquence vidéo est une succession d'images ou trames acquises successivement à des instants séparés par un intervalle de temps t assez petit. Elle peut être caractérisée par trois principaux paramètres : sa résolution en luminance, sa résolution spatiale et sa résolution temporelle. La résolution en luminance détermine le nombre de nuances ou de couleurs possibles pour un pixel. Celle-ci est généralement de 8 bits pour les niveaux de gris et de 24 bits pour les séquences en couleurs.

2.7. Exemples de formats d'images numériques

Les images numériques sont créées par de nombreux capteurs avec des formats différents. Le tableau suivant présente quelques formats d'images numériques.

Format	Extension	Type de fichier
BMP	.bmp	utilisé par windows (bitmap) .Sans compression. Format non supporté pour la publication sur internet.
TIFF (<i>Tagged Image File Format</i>)	.tif ou .tiff	Ancien format de fichier graphique bitmap très utilisé. Diverses compressions.
GIF (<i>Graphics Interchange Format</i>)	.gif	Format de fichier graphique bitmap. Images jusqu'à 16000 x 16000 pixels codées en 256 couleurs. Compression LZW (<i>Lempel-Ziv-Welch</i>).
JPEG (<i>Joint Photographic Experts Group</i>)	.jpg	Compression JPEG. Très utilisé pour la publication sur internet
JPEG 2000	.jp2	Le plus connu et le plus employé pour ses modes de compression variés.
PNG (<i>Portable Network Graphics</i>)	.png	Récent. Accélère et améliore l'affichage. Code jusqu'à 24 bits. Compression sans perte RLE (<i>Run Length Encoding</i>).

Remarque : Le terme bitmap fait référence aux formats de fichier qui stockent les pixels sous forme de tableau de points, avec une gestion des couleurs soit en couleur vraie soit par une palette indexée.

2.8. Types des traitements d'images

On désigne l'ensemble des opérations sur les images numériques, qui transforment une image en une autre image. Il existe différents types de transformations d'images :

2.8.1. Traitement ponctuel

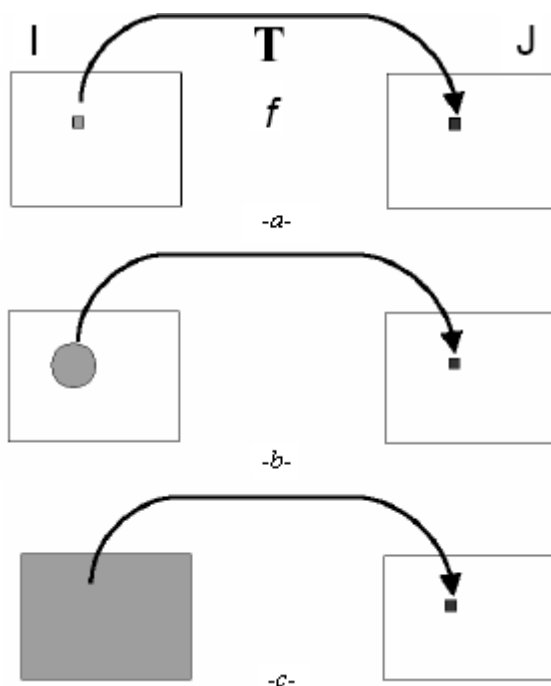
En chaque point on applique une formule de modification du niveau de gris indépendamment de la localisation géométrique de ce pixel dans l'image (sa position), c.-à-d. le nouveau niveau du gris dépend seulement de l'ancien niveau de gris du pixel considéré.

2.8.2. Traitement local

Le nouveau niveau de gris en un point est fonction de l'ancien niveau de gris du pixel considéré et de ses voisins.

2.8.3. Traitement global

Le nouveau niveau de gris en un point est fonction de l'ancien niveau de gris de tous les pixels de l'image.



-a-Transformations ponctuelles (point à point) :

$J(i_0, j_0)$ est obtenu en transformant $I(i_0, j_0)$:
 $J(i_0, j_0) = f[I(i_0, j_0)]$.

Exemple : Opération sur les histogrammes

-b-Transformations locales:

$J(i_0, j_0)$ est obtenu à partir des éléments de I autour de la position (i_0, j_0) : $J(i_0, j_0) = f[I(V)]$
 avec V : voisinage de (i_0, j_0) .

Exemple : Filtres,..

-c-Transformations globales: $J(i_0, j_0)$ est obtenu à partir de tout I : $J(i_0, j_0) = f[I(i, j)]$

Exemple : Transformée de fourrier,..

Figure 2.10. Différents types de traitement, a : ponctuel, b : local, c : global

Chapitre 3

Traitements de bases sur l'image

3.1. Notion d'histogramme et de contraste

3.1.1. Histogramme

L'histogramme est un graphique qui correspond à l'évolution du nombre des pixels d'une image possédant un même niveau de gris. En pratique, pour le calcul d'un histogramme, on donne un nombre de niveau de gris, et pour chaque niveau, on compte le nombre de pixels de l'image correspondant à ce niveau.

$$h(r_k) = n_k, \quad k = 0, \dots, L-1$$

r_k : $k^{\text{ième}}$ niveau de gris ; n_k : Nombre de pixels ayant le niveau de gris r_k ; L : Nombre de niveaux de gris.

Pour des images en niveaux de gris l'histogramme indique pour chaque valeur entre le noir (0) et le blanc (255), combien il y a de pixels de cette valeur dans l'image; en abscisse (axe x) : le niveau de gris (de 0 à 255); en ordonnée (axe y) : le nombre de pixels.

Les pixels sombres apparaissent à gauche de l'histogramme, les pixels clairs à droite de l'histogramme et les pixels gris au centre de l'histogramme.

Exemple 1 : Compter le nombre de pixels pour chaque niveau de gris, à l'aide de la matrice des valeurs de luminance.



Figure 3.1 Image synthétique et son histogramme

- L'image comporte 3 niveaux de gris différents : 0, 1 et 2.
- Les niveaux 0, 1 et 2 sont respectivement représentés par 24, 12 et 28 pixels \Rightarrow représentation de cette population de pixels sur l'histogramme.

Exemple 2 : étant donné une image $I(5 \times 5)$:

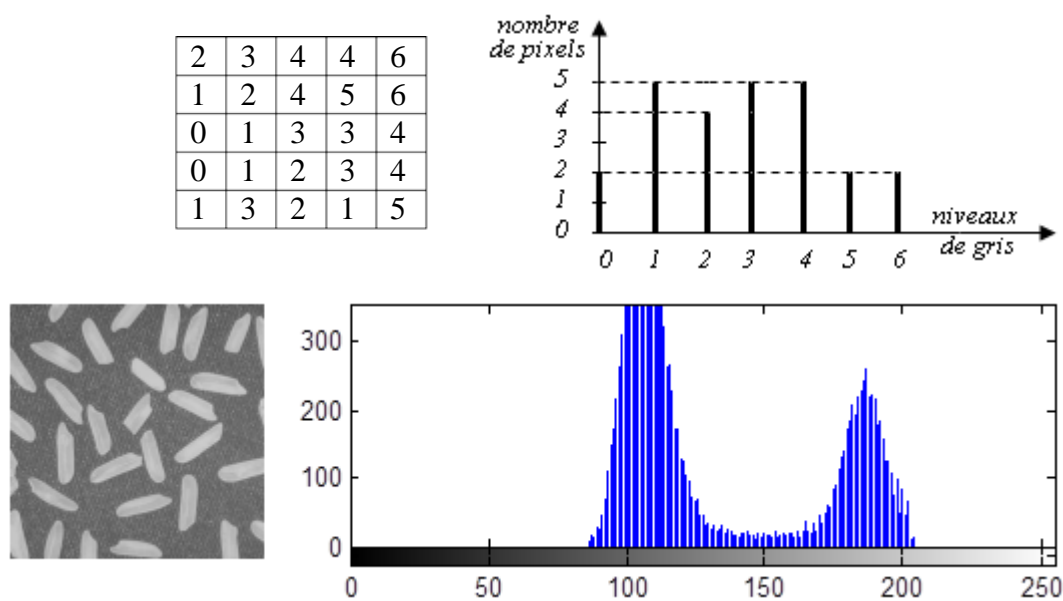


Figure 3.2. Image réelle et son histogramme

Remarque : L'histogramme donne le contenu global de l'image, un histogramme déporté vers les niveaux noirs nous informe qu'il s'agit d'une image sombre alors qu'un histogramme déporté vers les niveaux blancs est celui d'une image bien éclairée. Pour l'image ci-dessous, on remarque l'existence de deux zones distinctes, l'une représente les grains de riz, l'autre représente la texture.

Cependant, il ne donne aucune information sur la répartition des niveaux de gris. Les deux images ci-dessous sont différentes mais ayant le même histogramme.

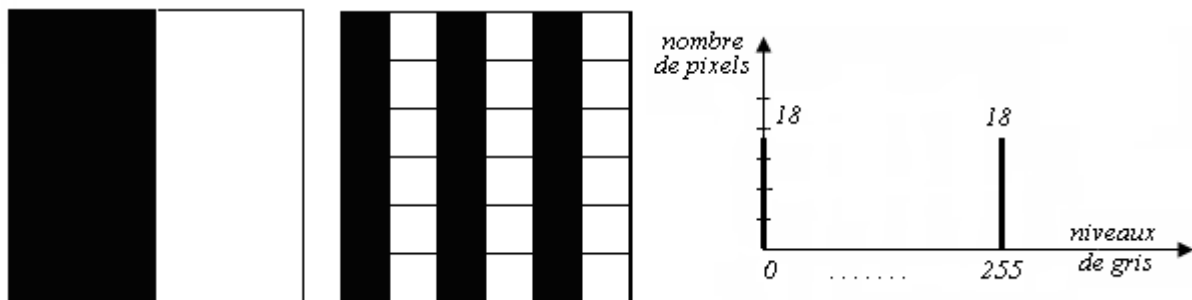


Figure 3.3. Deux images différentes ayant le même histogramme

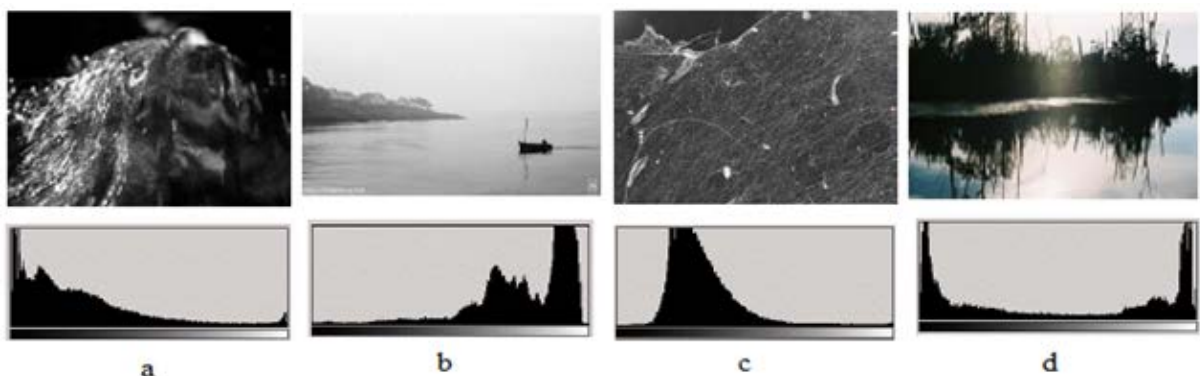


Figure 3.4. Histogramme de quelques images

a : Image sombre : beaucoup de pixels pour les faibles valeurs de niveau de gris

b : Image claire : beaucoup de pixels pour les valeurs claires de niveau de gris

c : Image peu contrastée : la plupart des pixels pour les valeurs moyennes de niveaux de gris

d : Image presque binaire : pixels concentrés vers le noir et le blanc

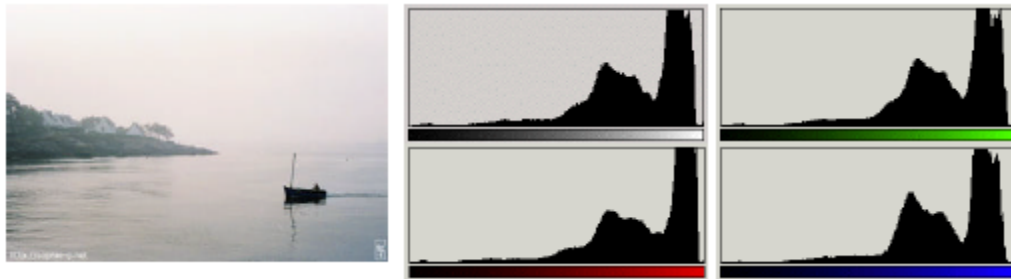


Figure 3.5. Histogrammes d'une image couleur

Image RGB claire : l'image contient beaucoup de gris et les histogrammes sont similaires

3.1.2. Histogramme normalisé

La probabilité d'apparition d'un niveau de gris r_k dans l'image est :

$$h_n(r_k) = \frac{h(r_k)}{N} \in [0,1], \quad k = 0, \dots, L-1$$

3.1.3. Histogramme cumulé

Il représente la distribution cumulée des intensités des pixels d'une image :

$$h_C(r_k) = \sum_{i=0}^k h(r_i)$$

Il est défini de façon récursive par :
$$\begin{cases} h_C(r_0) = h(r_0) \\ h_C(r_k) = h_C(r_{k-1}) + h(r_k) \end{cases} \quad k = 0, \dots, L-1$$

Exemple 1 :

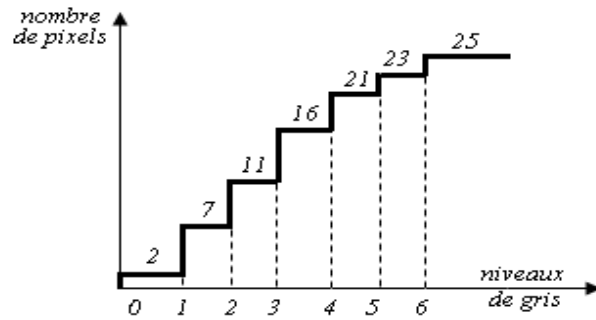


Figure 3.6. Image synthétique et son histogramme cumulé

- Chaque bâton cumule le nombre de pixels des niveaux de gris concerné et des niveaux de gris inférieurs : les niveaux 0, 1, 2 sont donc représentés respectivement par 24, 36 et 64 pixels.

Exemple 2 : Informations issues de l'histogramme

2	3	4	4	6
1	2	4	5	6
0	1	3	3	4
0	1	2	3	4
1	3	2	1	5



3.1.4. Informations issues de l'histogramme

1. **Luminance ou brillance d'une image** = moyenne $\mu = \frac{1}{N} \sum_{n=1}^N I(i, j)$ des niveaux de gris.

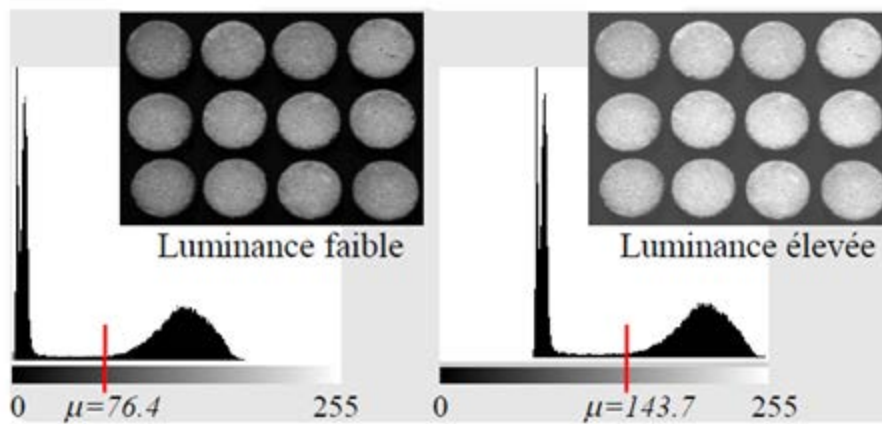


Figure 3.7. Lien histogramme/luminance

2. **Contraste** : Le contraste peut être défini de plusieurs façons :

- Variance des niveaux de gris (N nombre de pixels dans l'image) :

$$\sigma^2 = \frac{1}{N} \sum_{n=1}^N (I(i, j) - \mu)^2$$

- Variation entre niveaux de gris max et min : $\frac{\max[I(i, j)] - \min[I(i, j)]}{\max[I(i, j)] + \min[I(i, j)]}$

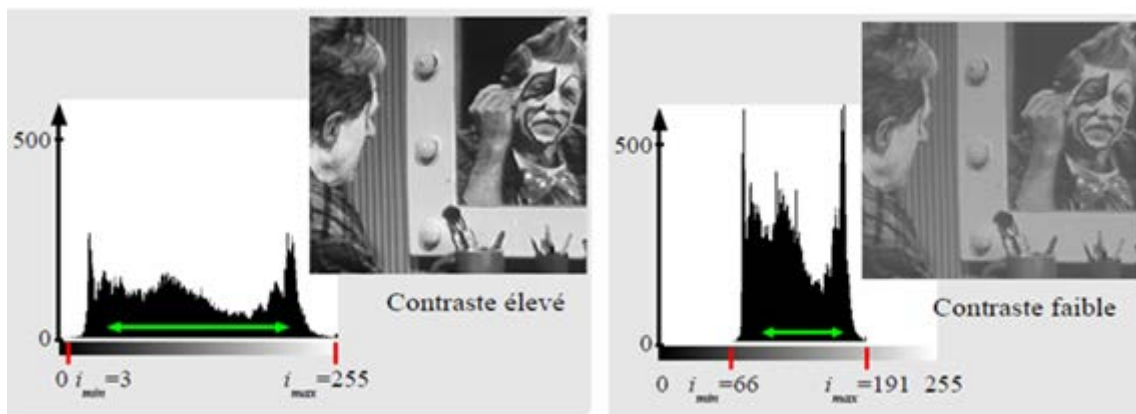


Figure 3.8. Lien histogramme/contraste

Le contraste est d'autant plus élevé que l'histogramme est « étalé » horizontalement.

3. Dynamique de l'image : nombre de niveaux $[I_{min}, I_{max}]$.

3.2. Correction de la dynamique de l'image par les transformations affines sur l'histogramme

3.2.1. Étirement d'histogramme

L'étirement d'histogramme est une technique qui modifie la répartition des niveaux de gris dans une image afin d'en améliorer le contraste, pour cela, il convient d'augmenter l'intervalle des niveaux de gris sur l'histogramme de l'image d'entrée.

Si l'histogramme de l'image originale commence de Ng_{min} et se termine par Ng_{max} , on peut avoir une image dont le niveau de gris s'étale de Ng'_{min} à Ng'_{max} par la relation :

$$Ng' = \frac{Ng'_{max} - Ng'_{min}}{Ng_{max} - Ng_{min}}(Ng - Ng_{min}) + Ng'_{min}$$

Dans le cas où $Ng'_{max} = 255$, $Ng'_{min} = 0$, $Ng' = \frac{255}{Ng_{max} - Ng_{min}}(Ng - Ng_{min})$

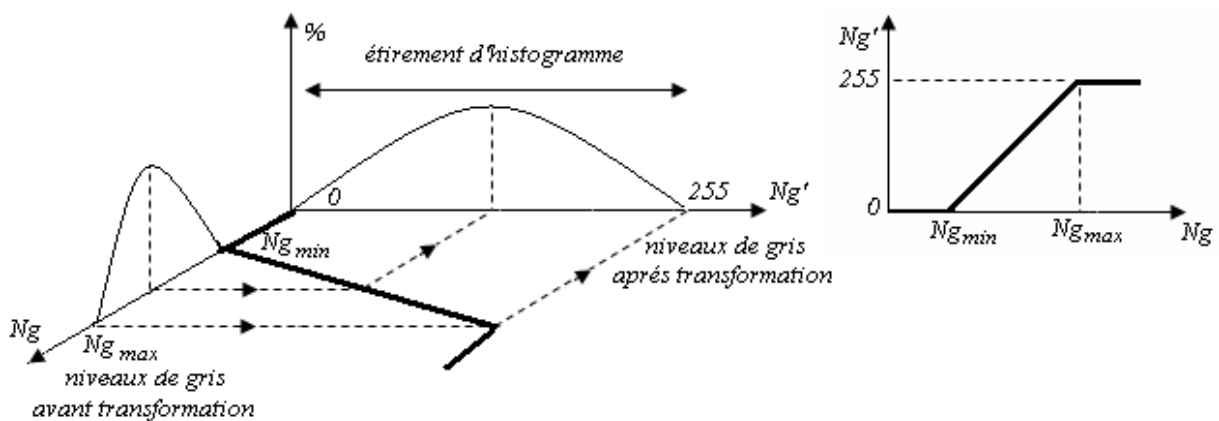


Figure 3.9. Principe de l'étirement d'histogramme

Exemple 1: Soit une image dont l'histogramme est visualisé ci-dessous. Ses valeurs en niveaux de gris sont comprises entre 10 et 80. L'image résultante est très foncée, afin d'améliorer l'image trouver une méthode permettant de recadrer (étaler) l'histogramme entre "0" et "255" sans changer la valeur des pics.

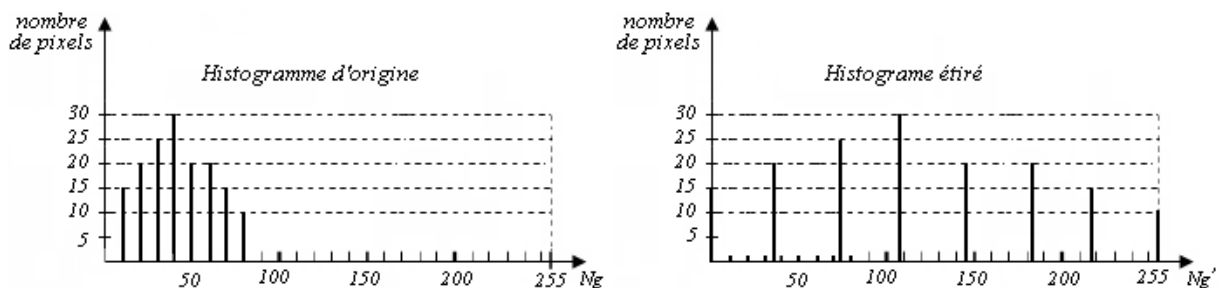


Figure 3.10. Histogramme et son étirement

En appliquant la formule précédente, l'histogramme de départ est réparti sur l'ensemble des valeurs de niveaux de gris. Le premier terme permet de décaler l'histogramme sur la gauche et le deuxième terme permet d'étaler l'histogramme sur tous les niveaux.

Ng	10	20	30	40	50	60	70	80
Ng'	0	36	72	109	145	182	218	255
N. pixels	15	20	25	30	20	20	15	10

Exemple 2 :

Dans l'exemple ci-dessous, l'image 'rice' a un niveau de gris qui s'étale de 44 à 206. en faisant un étirement d'histogramme de 0 à 255, on obtient une image plus contrastée (qualité visuelle meilleure)

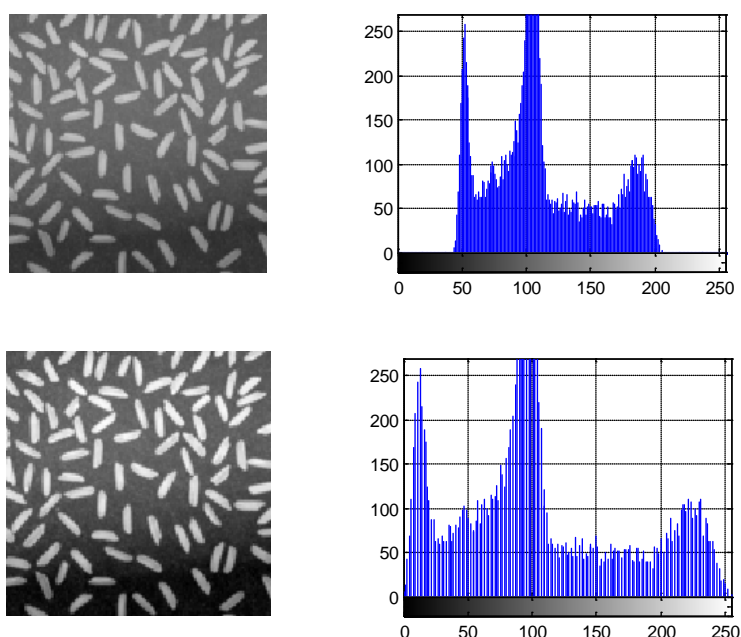
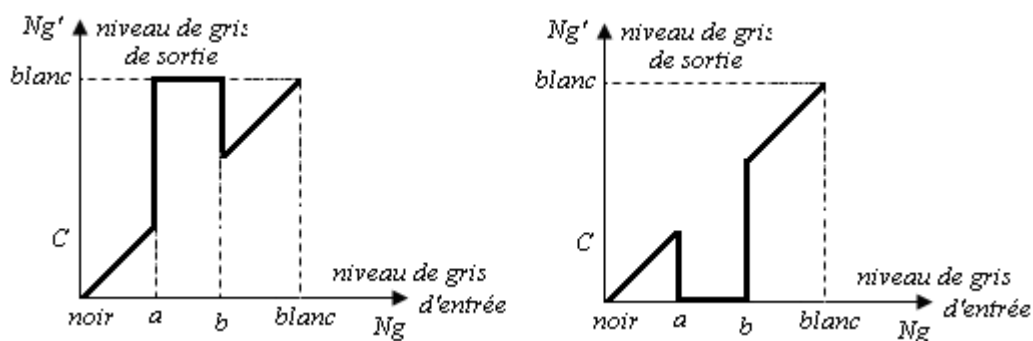


Figure 3.11. Etirement d'histogramme d'une image

3.2.2. Masquage des zones

Le masquage sert à remplacer un intervalle de niveau de gris par du blanc ou du noir.



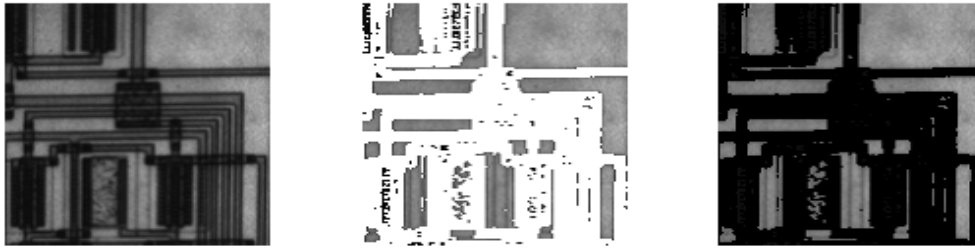


Figure 3.12. Masquage des zones

3.2.3. Extraction d'une fenêtre d'intensité

Avec la transformation décrite ci-dessous, la nouvelle image ne visualise que les pixels dont le niveau d'intensité appartient à l'intervalle $[a, b]$. Sous réserve d'une connaissance a priori de la distribution des niveaux de gris des objets de l'image originale, cette technique permet une segmentation d'objets particuliers de l'image.

$$Ng' = \begin{cases} Ng & a \leq Ng \leq b \\ 0 & b < Ng < a \end{cases}$$



Figure 3.13. Seuillage avec fenêtre d'intensité entre 100 et 200

3.2.4. Dilatation locale

Ce type de traitement permet d'accentuer le contraste dans une plage précise de niveau.

$$Ng' = \begin{cases} \frac{Ng_1'}{Ng_1} Ng & 0 \leq Ng \leq Ng_1 \\ \frac{(Ng_{\max} - Ng_1')Ng + Ng_{\max}(Ng_1' - Ng_1)}{Ng_{\max} - Ng_1} & Ng_1 \leq Ng \leq Ng_{\max} \end{cases}$$

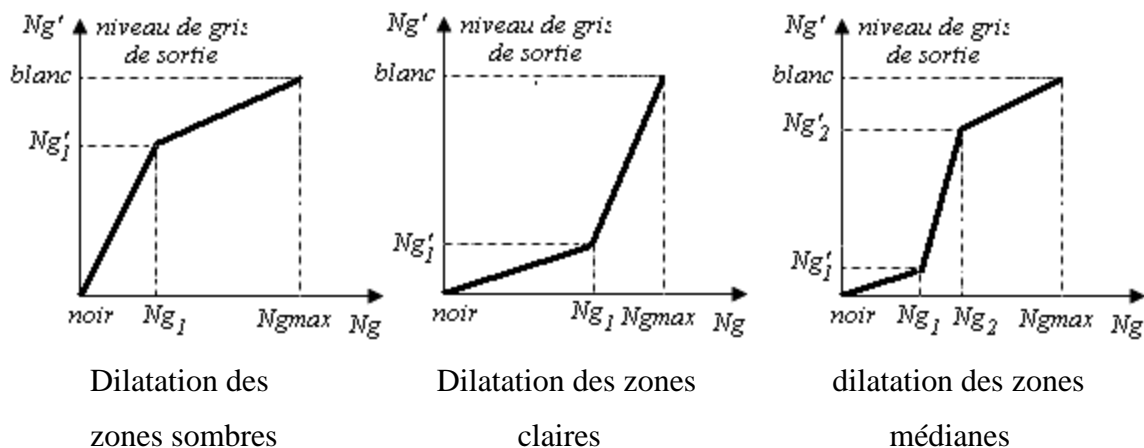
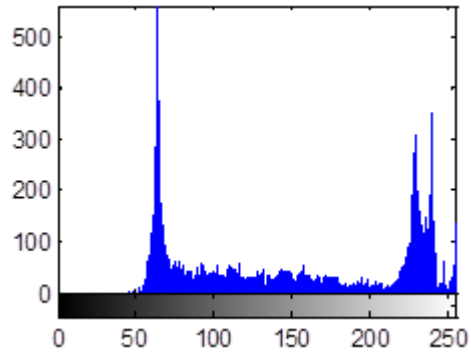




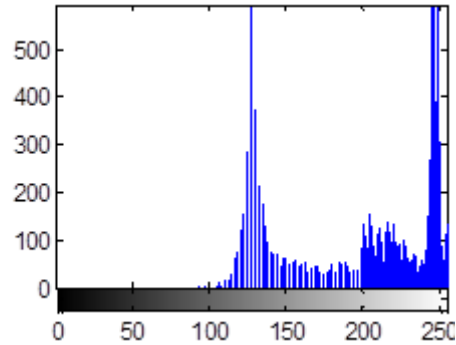
Image originale



Histogramme



Dilatation de la dynamique des zones sombres $Ng_1 = 100$, $Ng'_1 = 200$



Dilatation de la dynamique des zones claires $Ng_1 = 180$, $Ng'_1 = 100$

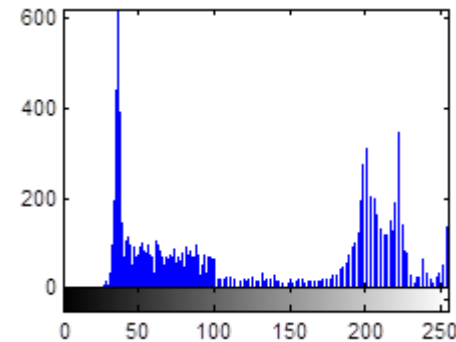


Figure 3.14. Exemples de dilatation de la dynamique

3.3. Egalisation d'histogramme et correction gamma

3.3.1. Egalisation d'histogramme

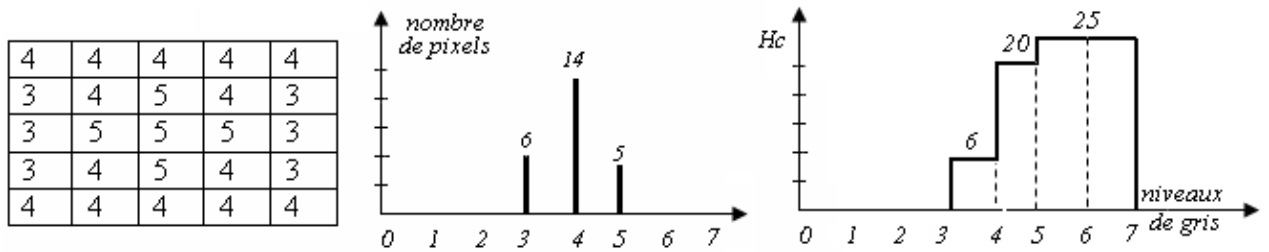
L'égalisation d'histogramme comprend un étirement d'histogramme avec une répartition uniforme des niveaux de gris. Chaque niveau de gris est représenté par un nombre constant de pixel. (Équilibrer le mieux possible la distribution des pixels).

$$f'(x, y) = Ng'_{\max} \frac{Hc[f(x, y)]}{N}$$

N : Nombre de pixels de l'image ; $Hc[\cdot]$: Histogramme cumulé.

$f(x, y), f'(x, y)$: Niveau de gris du pixel de coordonnées x, y de l'image originale et celle transformée.

Exemple 1 : Soit une image $f(5 \times 5)$ et son histogramme associé, trouver l'image obtenue par une égalisation d'histogramme.



$$Ng'_{\max} = 7, N = 25$$

$$f(x, y) = 3 \Rightarrow f'(x, y) = \frac{7 \times 6}{25} = 1.68 \approx 1 ; f(x, y) = 4 \Rightarrow f'(x, y) = \frac{7 \times 20}{25} = 5.6 \approx 5$$

$$f(x, y) = 5 \Rightarrow f'(x, y) = \frac{7 \times 25}{25} = 7$$

Exemple 2 : Effet de l'égalisation d'histogramme

Cet exemple d'égalisation permet visuellement de rehausser les contrastes de l'image. L'histogramme obtenu après égalisation s'étale bien sur toute l'échelle des niveaux de gris avec un espacement accru. Les données discrètes des niveaux de gris ne permettent pas d'obtenir un histogramme rigoureusement plat.

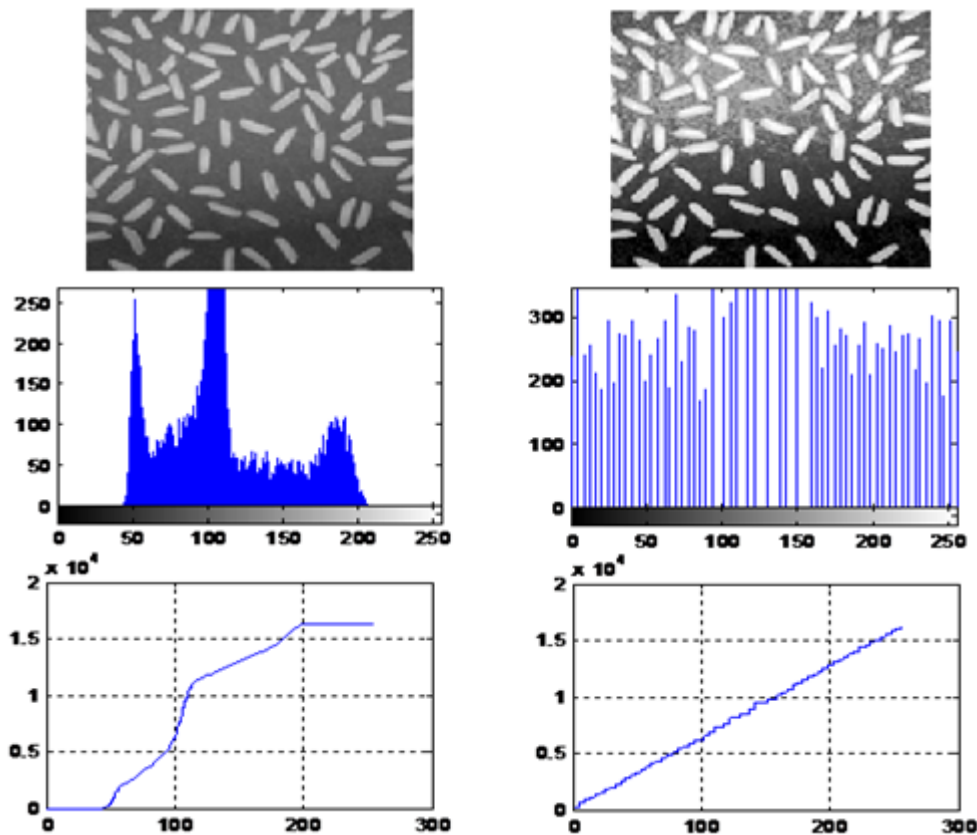


Figure 3.15. Image originale, Histogramme et Histogramme cumulé
Image égalisée, Histogramme égalisé et Histogramme cumulé

L'égalisation d'histogramme est une transformation des niveaux de gris dont le principe est d'équilibrer le mieux possible la distribution des pixels dans la dynamique (Idéalement, on cherche à obtenir un histogramme plat). Le résultat est une augmentation globale du contraste dans l'image.

Avantage

Le rehaussement du contraste est plus marqué avec l'égalisation d'histogramme autorisant la détection de structures situées dans l'ombre. En fait, tout niveau de gris fortement représenté est étiré à l'inverse tout niveau de gris faiblement représenté est fusionné avec d'autres niveaux proches.

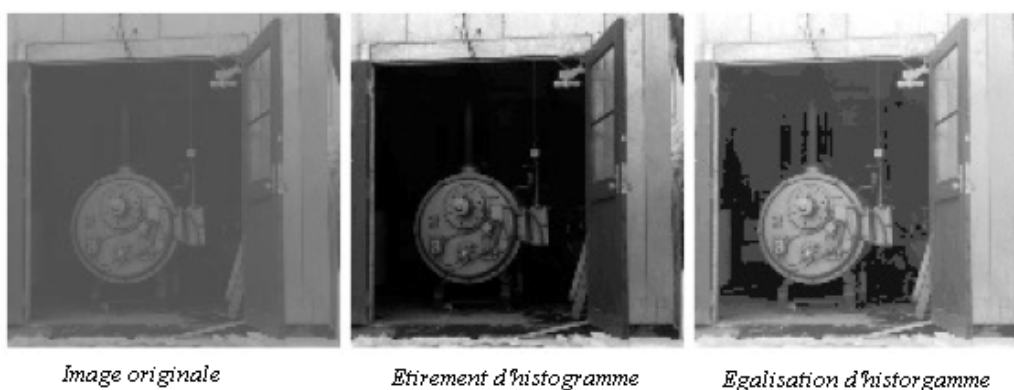


Figure 3.16. Avantages de l'égalisation d'histogramme

Attention si l'image est déjà bien contrastée une égalisation peut dégrader l'image !

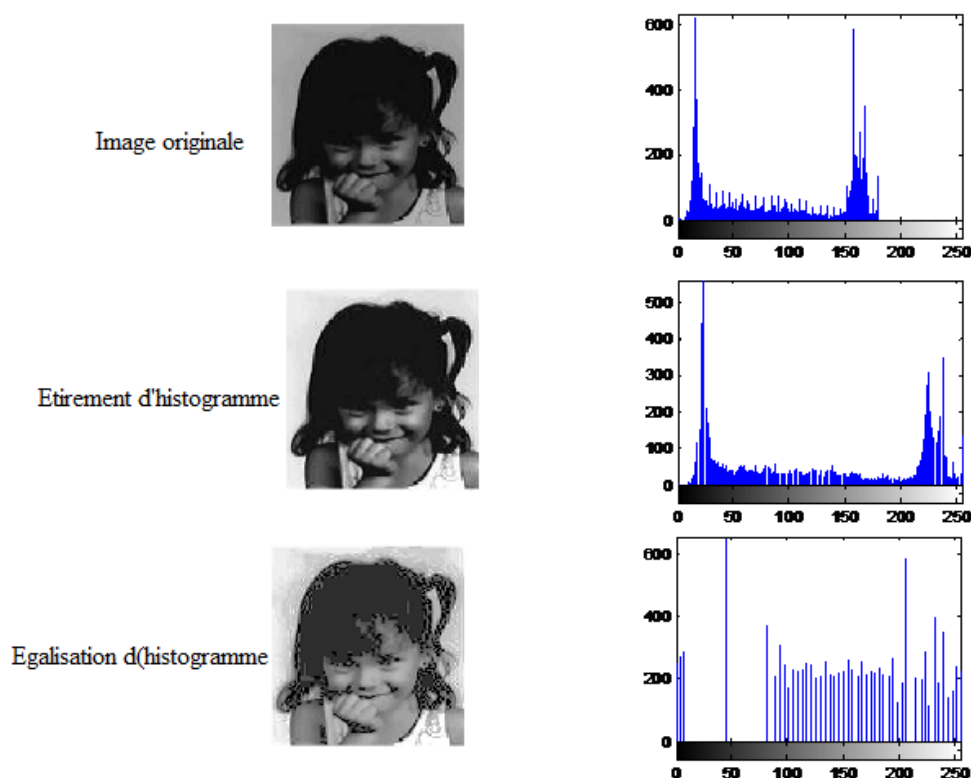


Figure 3.17. Inconvénient de l'égalisation d'histogramme

3.3.2. Correction gamma

C'est une transformation non linéaire donnée par :

$$Ng'(x, y) = 255 \left(\frac{Ng(x, y)}{255} \right)^{1/\gamma}$$

Si $\gamma < 1$ éclaircit principalement les parties foncées

Si $\gamma > 1$ assombrit principalement les parties claires

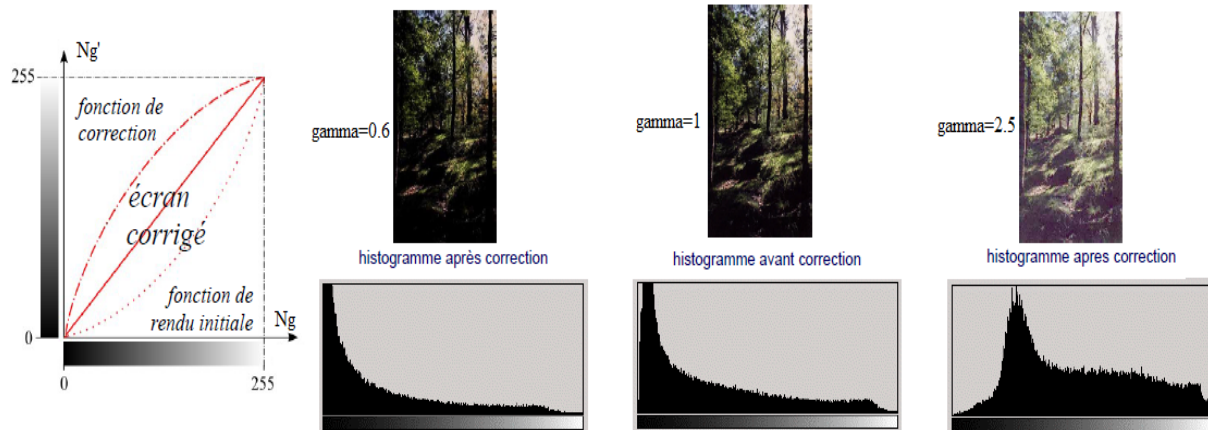


Figure 3.18. Exemple de correction gamma

3.4. Opérations arithmétiques et logiques sur les images

3.4.1. Opérations arithmétiques

Copie : l'image de sortie est une copie de l'image d'entrée.

Inversion : L'image résultante d'une opération d'inversion est une image dont le noir devient blanc et le blanc devient noir. Pour une image de 256 niveaux de gris : $Ng' = 255 - Ng$

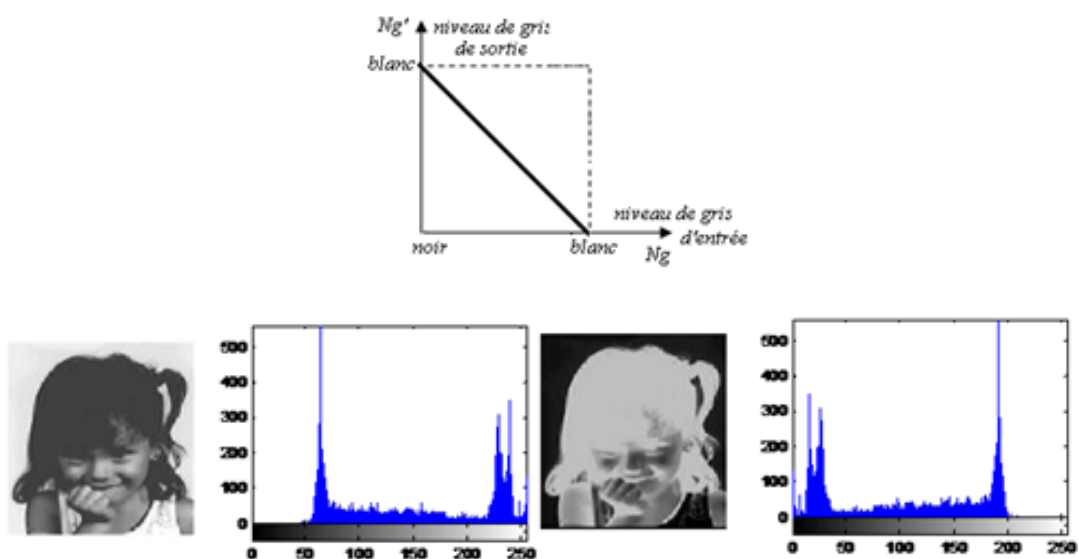


Figure 3.19. Inversion d'images

Addition : L'addition est une opération qui sert à augmenter la brillance de l'image. Elle se fait par l'addition d'une constante aux niveaux de gris de tous les pixels de l'image.

Si $Ng(x,y)$ est le niveau de gris du pixel (x,y) de l'image originale, le niveau de gris du même pixel (x,y) de l'image résultante par l'addition est : $Ng'(x,y) = Ng(x,y) + C$

L'image de la figure ci-dessous a un niveau de gris de 0 à 180 et subit une augmentation de brillance par l'addition de 75 à tous les pixels qui est équivalent à une translation de l'histogramme de 75.

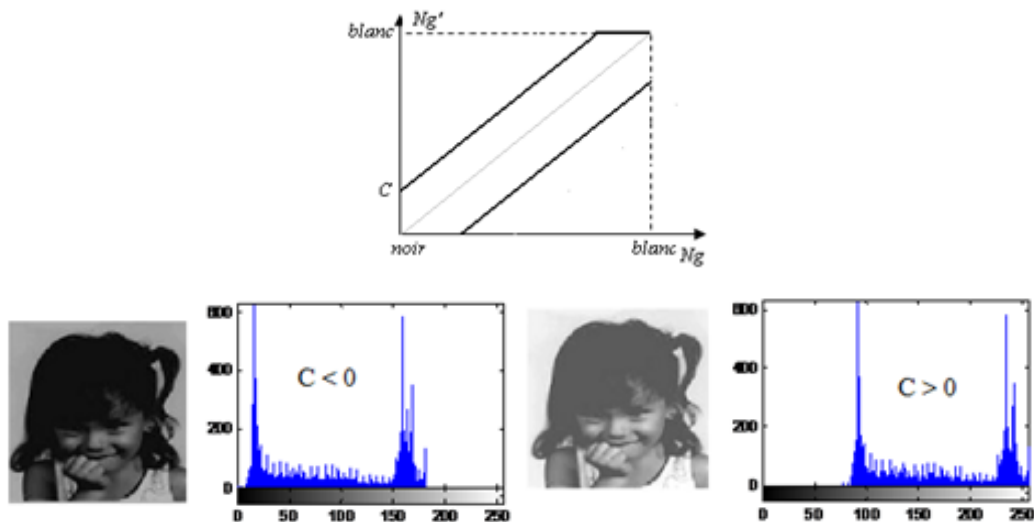


Figure 3.20. Exemple d'addition

Multiplication : Par la multiplication des niveaux de gris par une constante, l'image originale subit un éclairage ou un assombrissement suivant la valeur de la constante :

$$Ng'(x,y) = \alpha \cdot Ng(x,y)$$

$\alpha > 1$: Éclairage , $\alpha = 1$: Copie , $0 < \alpha < 1$ Assombrissement

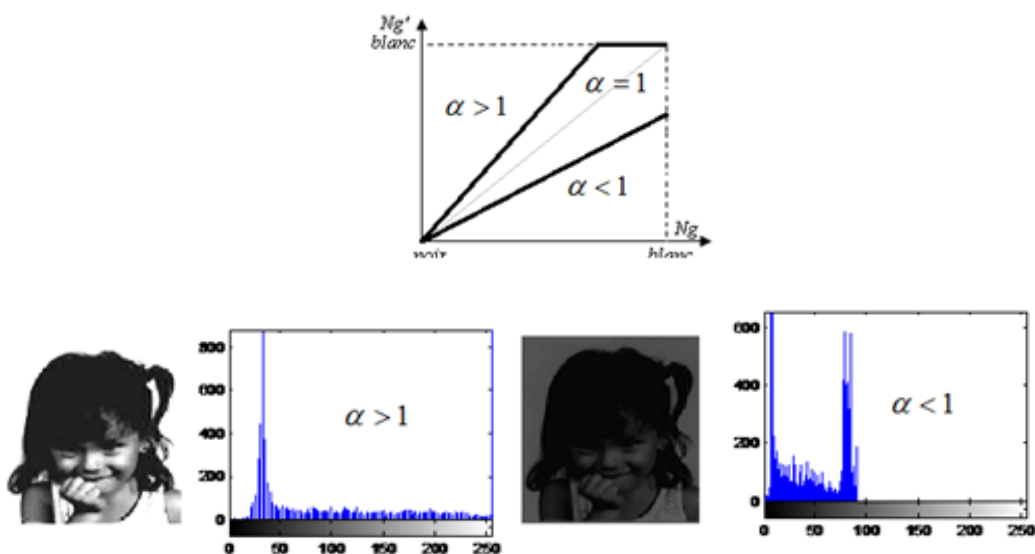


Figure 3.21. Exemple de multiplication

Opérateurs arithmétiques sur deux images: Ils créent une image résultat $P'(x, y)$ à partir d'une ou plusieurs images $P1(x, y), P2(x, y)$, etc. où x et y sont les numéros des lignes et colonnes.

- Addition : $P'(x, y) = P1(x, y) + P2(x, y)$
- Soustraction : $P'(x, y) = P1(x, y) - P2(x, y)$
- Multiplication : $P'(x, y) = P1(x, y) \cdot P2(x, y)$
- Division : $P'(x, y) = P1(x, y) / P2(x, y)$
- Minimum / maximum : $P'(x, y) = \text{Min}[P1(x, y), P2(x, y)]$
- Combinaison linéaire : $P'(x, y) = C1 \cdot P1(x, y) + C2 \cdot P2(x, y)$ où $C1$ et $C2$ sont des constantes.
- Valeur absolue : $P'(x, y) = |P1(x, y)|$

3.4.2. Opérateurs logiques

Les opérateurs logiques sont appliqués aux images binaires, c'est à dire des images dont les pixels ne peuvent valoir que la valeur "vrai" ou la valeur "faux".

Non logique : "Not" ; Addition logique : "And" ; Ou logique : "Or" ; Ou exclusif logique : "Xor" ; Différence logique : "logical-sub" ; Equivalence logique : "n xo"

A	B	Not A	A and B	A or B	A xor B	A-B	A n xo B
1	1	0	1	1	0	0	1
1	0	0	0	1	1	1	0
0	1	1	0	1	1	0	0
0	0	1	0	0	0	0	1


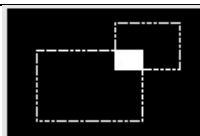

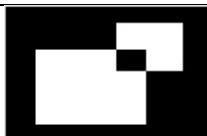
Images binaires	ET logique	OU logique	OU exclusif
			

Figure 3.22. Operations logiques sur deux images

Chapitre 4

Filtrage numérique des images

4.1 Introduction sur le filtrage numérique

Le filtrage est une opération qui a pour but d'extraire une information ou d'améliorer l'aspect de l'image, par exemple en éliminant un bruit ou en améliorant les contours d'une image floue. Le traitement local d'une image utilise la notion de filtre (filtrage spatial). Ces filtres peuvent s'écrire sous la forme d'un produit de convolution entre une matrice (ou noyau de convolution) et d'une image. Malheureusement, en appliquant de gros masques de convolution, on peut constater que le temps de calcul devient très important. Pour cela, on peut appliquer les masques dans l'espace des fonctions de Fourier (filtrage spectral) qui rendront les calculs beaucoup plus rapides car il existe un algorithme de calcul de la transformée très rapide (FFT).

Les sources de bruit :

- ✓ Bruits liés aux conditions de prise de vues : Ce sont des événements vérifiant les conditions d'acquisition du signal : le bougé, problèmes liés à l'éclairage de la scène observée...
- ✓ Bruits liés au capteur : Capteur mal réglé, capteur de mauvaise qualité (distorsion de la gamme des niveaux de gris ou en flou)...
- ✓ Bruits liés à l'échantillonnage : Ces bruits reflètent essentiellement des problèmes de quantification (CCD).
- ✓ Bruits liés à la nature de la scène : Nuage sur les images satellitaires, poussières dans les scènes industrielles, brouillard pour les scènes routières...

4.2 Filtrage spatial et convolution 2D

Dans le traitement local le nouveau niveau de gris en un point est fonction de l'ancien niveau de gris du pixel considéré et de ses voisins. L'opération de voisinage fait appel à un domaine associé à un point $p(x, y)$ limité par un contour de taille fixe appelé voisinage de p (neighbourhood en anglais). Le résultat (niveau de gris) est calculé en considérant l'ensemble des niveaux de gris inclus dans le voisinage; ce résultat est affecté au point de coordonnées (x, y) de l'image résultat.

Le calcul de l'image résultat consiste donc à prendre chacun des points de l'image source, à leur associer le voisinage et à effectuer le calcul. Cette méthode de calcul est donc beaucoup plus coûteuse en temps de calcul que l'opération ponctuelle.

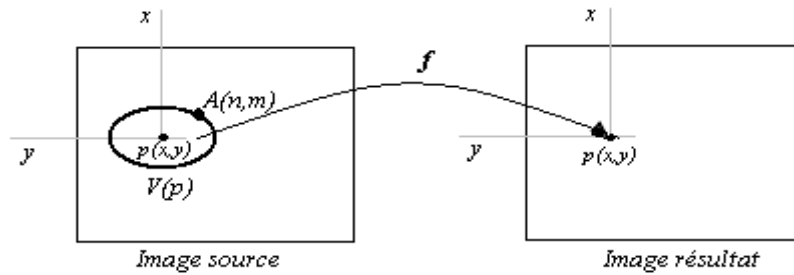


Figure 4.1. Principe de l'opération de voisinage

4.2.1 Voisinage

Un pixel $I(x, y)$ possède quatre voisins horizontaux et verticaux qui forment ce qu'on appelle le 4-voisinage de $I(x, y)$. Si on considère un pixel comme un carré élémentaire, les pixels présentant un côté commun avec $I(x, y)$ sont appelés les 4-voisins de (x, y) . Le pixel $I(x, y)$ possède aussi quatre voisins diagonaux, ce sont les pixels qui ont un sommet commun avec (x, y) . L'ensemble des huit voisins du pixel (x, y) représentent ce qu'on appelle les 8-voisins (8-voisinage). Ces huit voisins forment la fenêtre 3x3 du pixel $I(x, y)$. On dit aussi que ces pixels sont 8-adjacents de (x, y) .

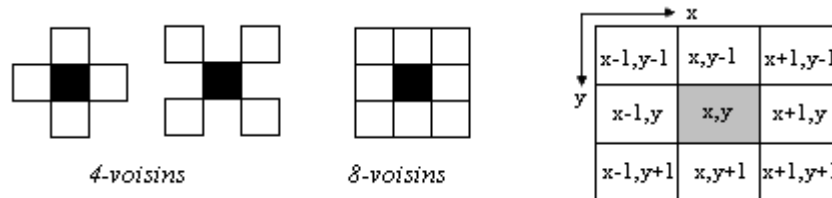


Figure 4.2. Exemples de voisinage

4.2.2 Produit de convolution

Le produit de convolution est une opération très courante en traitement d'images. Il sert d'opération de voisinage linéaire et invariante par translation. Cette opération fait donc appel aux outils disponibles pour respecter la convolution:

- multiplication du niveau de gris d'un point par une constante.
- addition des niveaux de gris des points d'une image; dans le cas continu, cette addition des niveaux se traduit sous forme d'une intégrale de surface.

1. Définition de la convolution dans l'espace continu

Soit F une image une image continue caractérisée par un niveau de gris f et G l'image résultat de niveau de gris g . Le produit de convolution $g = f * h$ est défini par :

$$g(x, y) = f * h \Big|_{x,y} = \iint_{(V)} f(x-u, y-v) h(u, v) du dv$$

h est la réponse impulsionnelle de l'opération de convolution.

2. Définition de la convolution dans l'espace discret

Soit F une image discrète caractérisée par un niveau de gris f et G l'image résultat de niveau de gris g . La convolution discrète est un calcul linéaire de voisinage:

$$g(m, n) = f * h \Big|_{m,n} = \sum_{i=-k_1}^{k_1} \sum_{j=-k_2}^{k_2} f(m-i, n-j) h(i, j), \quad k_1 = \frac{K-1}{2}, k_2 = \frac{L-1}{2}$$

où le domaine de définition de la réponse impulsionnelle $h(i, j)$ est restreint à une matrice de dimension (K, L) . Exemple pour $K=L=3, k_1=k_2=1$. La convolution de l'image f par un filtre de réponse impulsionnelle h représenté par un noyau fini (éventuellement tronqué) peut être calculée directement par balayage des pixels de f et calcul de la somme des valeurs des voisins de chaque pixel pondérées par les valeurs du noyau de convolution. Le mécanisme de la convolution discrète est facilement visualisable.

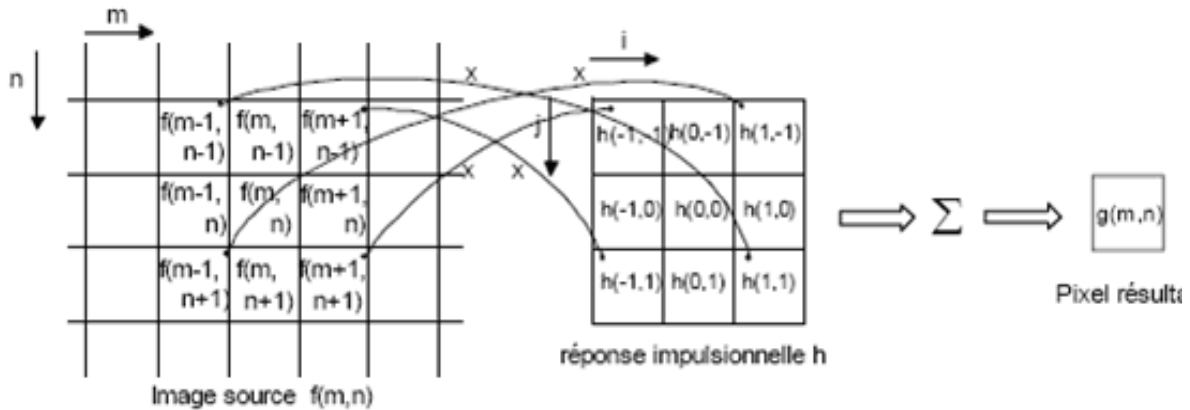


Figure 4.3. Principe de calcul de la convolution discrète à partir de la réponse impulsionnelle

On notera que le calcul met en correspondance les points de f avec les points symétriques de h . Cette remarque conduit à définir la notion de masque de convolution à partir d'une symétrie centrale appliquée à h .

$$\tilde{h}(m, n) = h(-m, -n)$$

Remarquer que le voisinage de calcul de f est strictement celui du masque.

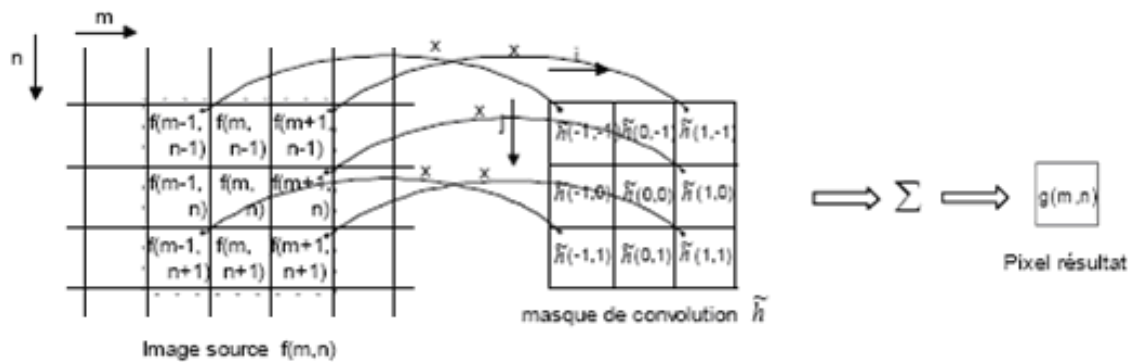


Figure 4.4. Principe de calcul de la convolution discrète par application d'un masque.

Le plus souvent, les opérateurs discrets seront donnés sous forme de masque.

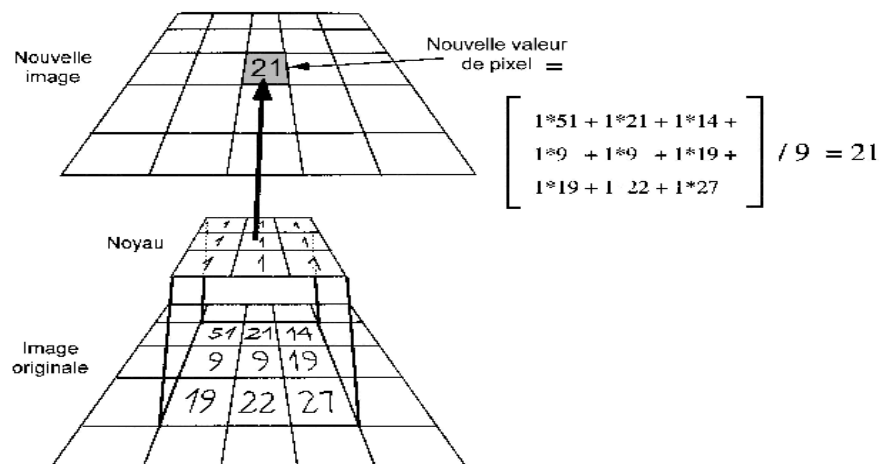


Figure 4.5. Exemple de calcul de la convolution par application d'un masque .

4.2.3 Traitement des points de bordure

Il y a plusieurs façons de gérer les points de bordure. Nous ne présenterons que les 3 méthodes suivantes :

- 1. La méthode des zéros :** Dans cette approche, on considère que tous les points qui ne font pas partie de l'image ont le niveau de gris 0. On remplit donc de zéros les points extérieurs voisins des points de bordure, avant de faire la convolution. Le nombre de points extérieurs à considérer est fonction de la taille du masque de convolution. La figure suivante montre un exemple pour un masque de taille 3 x 3.
- 2. La méthode symétrique :** Dans cette approche, on attribue aux points extérieurs les niveaux de gris de leurs symétriques se trouvant dans l'image.
- 3. La méthode de la symétrie circulaire :** Dans cette approche, on agit comme si l'image se refermait sur elle-même, et les points extérieurs sont vus comme s'ils se retrouvaient de l'autre côté de l'image. Il convient de noter que cette approche n'est pas réaliste sur le plan pratique, puisque la nature des pixels se trouvant sur un bord de la scène peut être complètement différente de celle des pixels se trouvant à l'autre extrémité.

4.2.4 Séparabilité d'un filtre de convolution

Un filtre de convolution est dit séparable si $h(x, y) = h_x(x) \cdot h_y(y)$

Cette structure de filtre simplifie grandement les calculs. Le premier calcul conduit à faire une convolution h_y selon les colonnes puis à faire une convolution h_x selon les lignes.

La décomposition du filtre h en deux sous filtres h_x et h_y formant des noyaux à 1 dimension permet de ramener un traitement bidimensionnel à un traitement selon x suivi d'un traitement selon y .

$$g(m, n) = \sum_{i=-k_1}^{k_1} \sum_{j=-k_2}^{k_2} f(m-i, n-j) h(i, j) = \sum_{i=-k_1}^{k_1} h_x(i) \sum_{j=-k_2}^{k_2} h_y(j) f(m-i, n-j)$$

Pratiquement, le filtre séparable est obtenu comme suit :

$$\begin{bmatrix} a & b & c \end{bmatrix} \otimes \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} a\alpha & b\alpha & c\alpha \\ a\beta & b\beta & c\beta \\ a\gamma & b\gamma & c\gamma \end{bmatrix} = \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} \times \begin{bmatrix} a & b & c \end{bmatrix} . \quad \begin{bmatrix} 1 & 2 & 1 \end{bmatrix} \otimes \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

\otimes : produit tensoriel

4.2.5 Classification des filtres

La plupart des opérations que l'on peut réaliser sur une image utilisent la notion de filtre. Ils peuvent servir à corriger une image, à détecter des contours ou encore à améliorer la qualité visuelle de l'image. Dans le filtrage global, chaque pixel de la nouvelle image est calculé en prenant en compte la totalité des pixels de l'image de départ en utilisant des transformations du domaine spatial au domaine fréquentiel.

Dans le filtrage local, chaque pixel de la nouvelle image est calculé en prenant en compte seulement un voisinage du pixel correspondant dans l'image d'origine. Chaque filtre cherche à atténuer un type de défaut bien précis. Il n'y a pas de filtre universel capable de corriger tous les défauts, il faut choisir les bons filtres suivant les défauts que vous désirez corriger. Ces filtres peuvent être classés suivant leurs nature (linéaire - non linéaire) et leur fonction (lissage – différentiation)

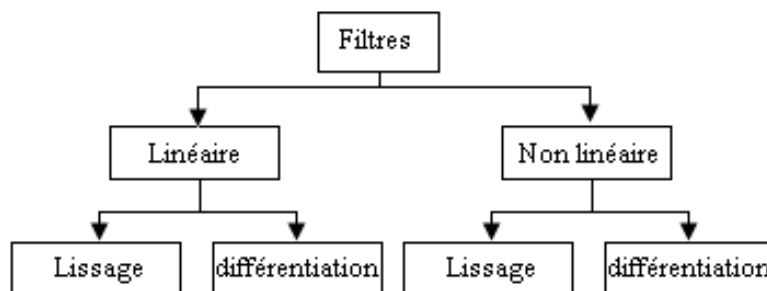


Figure 4.6.Classification des filtres

4.3 Filtres de Lissage

L'objectif du filtrage de lissage est de réduire les variations d'intensité au sein de chaque région de l'image tout en respectant l'intégrité des scènes : les transitions entre régions homogènes, les éléments significatifs de l'image doivent être préservés au mieux. Différentes méthodes de filtrage ont été développées suivant le type et l'intensité du bruit, ou les applications auxquelles on destine l'image. Les premières et les plus simples de ces méthodes sont basées sur le filtrage linéaire de lissage qui consiste à mélanger les niveaux de gris dans un voisinage, donc d'engendrer du flou sur l'image.

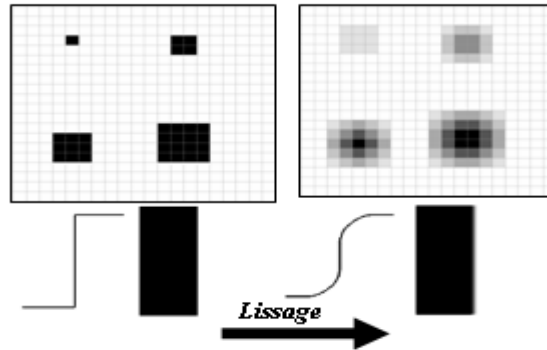


Figure 4.7. Lissage idéal / Lissage réel

Un masque de lissage doit obligatoirement avoir des coefficients non négatifs et leur somme doit être égale à 1. En générale, les coefficients d'un masque de lissage peuvent être donnés manuellement, ou par une fonction mathématique.

4.3.1 Filtres linéaires

Le filtre local est dit linéaire si la valeur du nouveau pixel est une combinaison linéaire des valeurs des pixels du voisinage. Cette combinaison linéaire est usuellement représentée par un masque.

Filtre moyennneur : $\frac{1}{9} \cdot \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix},$

$$\frac{1}{25} \cdot \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Filtre moyennneur pondéré : $\frac{1}{16} \cdot \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}, \quad \frac{1}{8} \cdot \begin{bmatrix} 0 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$

- *Diffuse le bruit (atténuation).*
- *Perte des fortes transitions (élimine les hautes fréquences).*
- *Plus la taille du filtre est grand, plus le lissage sera important, et plus l'image filtrée perd les détails de l'image originale.*

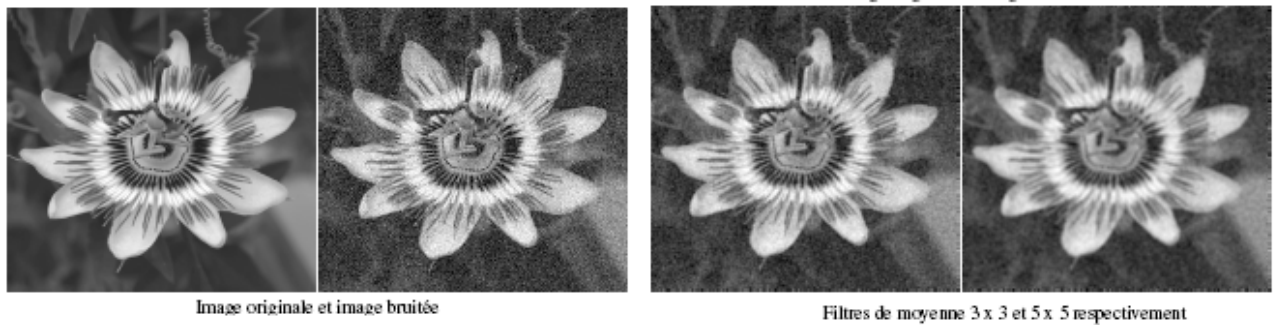


Figure 4.8. Effet de la taille du filtre moyennneur

Filtre gaussien : Le noyau gaussien centré et d'écart-type σ est défini par :

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{\left(\frac{-(x^2+y^2)}{2\sigma^2}\right)}$$

Si par exemple $\sigma = 0,8$, on a le filtre 3x3 suivant :
environ

$$\begin{bmatrix} G(-1, -1) & G(0, -1) & G(1, -1) \\ G(-1, 0) & G(0, 0) & G(1, 0) \\ G(-1, 1) & G(0, 1) & G(1, 1) \end{bmatrix} \simeq \frac{1}{16} \cdot \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

$$\frac{1}{300} \cdot \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 18 & 30 & 18 & 4 \\ 6 & 30 & 48 & 30 & 6 \\ 4 & 18 & 30 & 18 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$$

- **Accorde plus d'importance au pixel central**
- **Filtre séparable:** Économie de calcul, exemple: filtre 5x5 **nécessite** 5x5 multiplications + 24 additions par pixel, alors un filtre séparé nécessite 5x2 multiplications + 4x2 additions par pixel.
- **Atténuation du bruit**

L'effet de ce filtre sur l'image est assez similaire au filtre moyenne, mais la moyenne est pondérée en ce sens où les pixels près du centre ont un effet ou un "poids" plus important que ceux qui sont situés plus loin.

En général un filtre Gaussien avec un sigma < 1 est utilisé pour réduire le bruit, et si sigma > 1 , le filtre sert à engendrer une image floue pour simuler la defocalisation. Il faut noter que plus le sigma est grand, plus la cloche Gaussienne est large et plus le flou appliqué à l'image sera marqué.

Filtre binomial : Les coefficients de ce filtre sont obtenus par le binôme de Newton. Un filtre 1D binomial d'ordre 4 est un filtre donné par le vecteur

$$v = \frac{1}{16} [1 \ 4 \ 6 \ 4 \ 1]. \text{ Un filtre 2D binomial d'ordre 4 est le filtre}$$

séparable donné par $v^T v$:

$$\frac{1}{256} \cdot \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$$

Les filtres de lissage linéaires montrent des limites dans deux aspects :

- *Le bruit impulsionnel*
- *Le respect des frontières.*

Ces limitations ont conduit au développement des filtres non linéaires.

4.3.2 Filtres non linéaires

Si le filtre ne peut pas être exprimé par une combinaison linéaire, il est appelé " non linéaire ". Les filtres non linéaires sont plus complexes à mettre en œuvre que les filtres linéaires. Cependant les résultats obtenus avec les filtres non linéaires sont très souvent de meilleure qualité que ceux obtenus par les filtres linéaires.

Filtre médian : Le filtre médian est utilisé pour atténuer des pixels isolés, d'une valeur très différente de leur entourage.

Principe : le filtrage médian procède tout d'abord par un tri des valeurs de niveau de gris du voisinage suivi d'une sélection de l'élément milieu du tri, c-à-d remplacer la valeur du pixel central par la valeur médiane de la répartition (luminances triées dans l'ordre croissant) des niveaux de gris des pixels situés à l'intérieur de cette fenêtre.

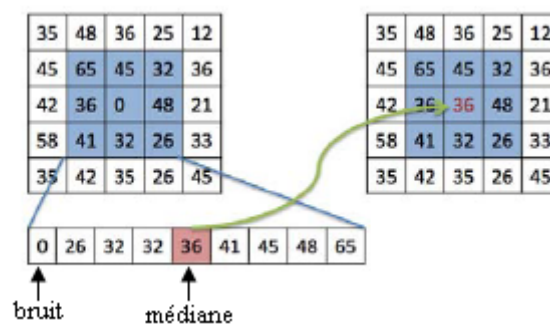


Figure 4.9. Principe du filtre median

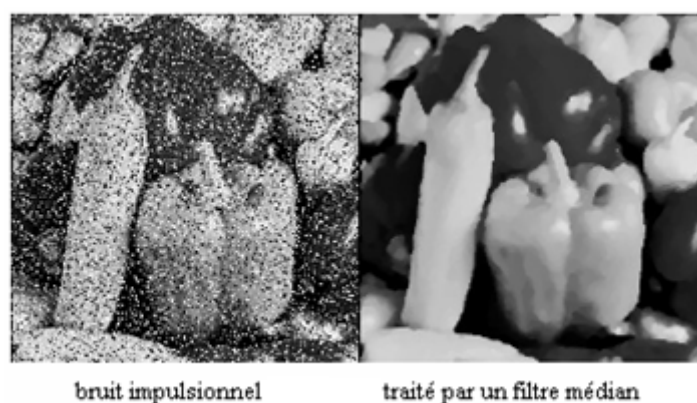


Figure 4.10. Exemple de filtrage par le filtre median

- *Très adapté au bruit type "poivre et sel" (faux "blanc" et "noir" dans l'image)*
- *Préserve les contours*
- *Réduit le bruit additif uniforme ou gaussien (lissage de l'image)*

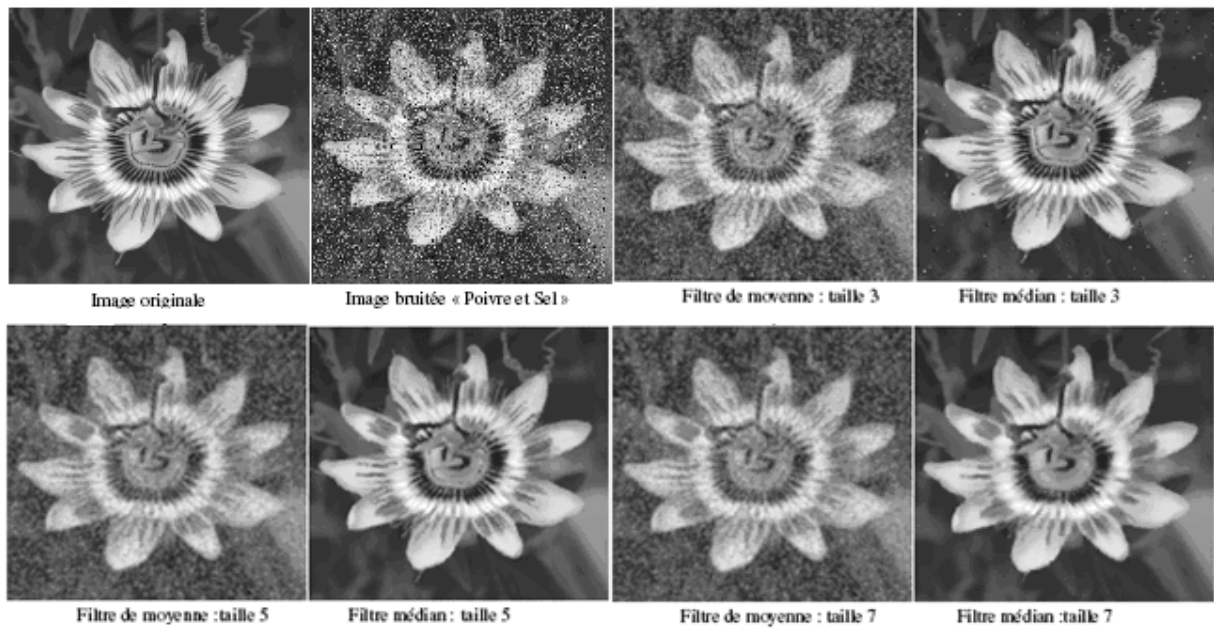


Figure 4.11. Comparaison filtres de moyenne et filtres médians

Filtre de Nagao : Le filtre de Nagao examine la fenêtre 5x5 centrée sur chaque pixel. 9 domaines sont définis dans cette fenêtre. On calcule pour chaque domaine D_i la moyenne μ_i et la variance $\nu_i = \sum_{k=1}^9 (\mu_i - x_k)^2$. Le résultat de l'opérateur est la moyenne du domaine qui présente la plus faible variance.



Figure 4.12. Les 9 fenêtres du filtre de Nagao

Propriétés :

- *complexité plus grande qu'un filtre linéaire,*
- *conserve la dynamique des luminances le long des contours,*
- *respecte mieux les angles des formes.*

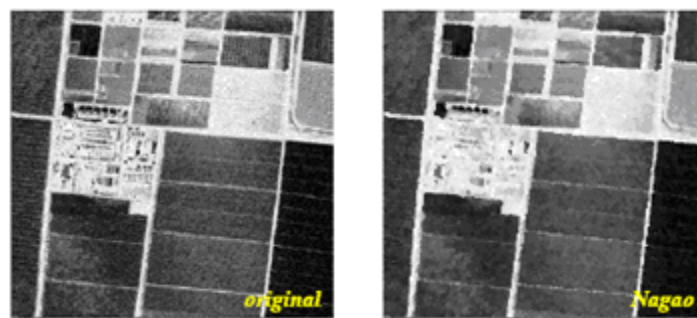


Figure 4.13. Exemple de filtrage par le filtre de NAGAO

4.4 Filtrage fréquentiel (spectral)

Dans le domaine spatial, le filtrage se fait par convolution, alors dans le domaine spectral (ou fréquentiel), il se fait par multiplication (ou masquage de l'image).

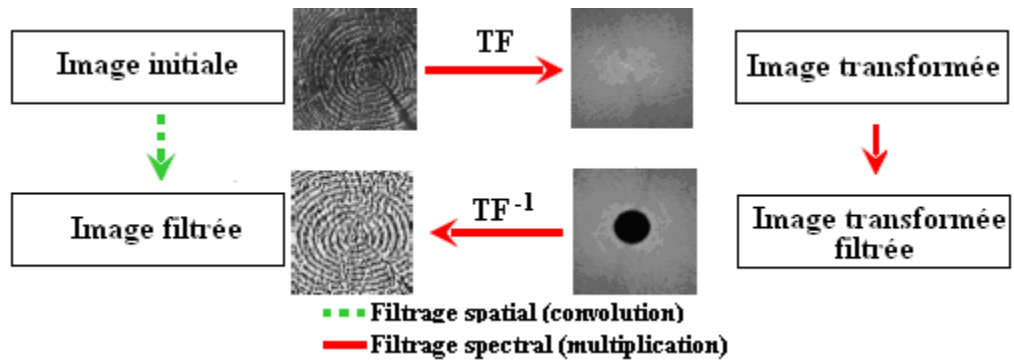


Figure 4.14. Filtrage spatial / fréquentiel

4.4.1 Représentation fréquentielle d'une image

La fréquence dans une image représente la variation de l'intensité des pixels de l'image, les basses fréquences (correspondent à des changements d'intensité lents) représentent les régions homogènes et floues, tandis que les hautes fréquences (correspondent à des changements d'intensité rapides) représentent les contours et les changements brusques d'intensité.

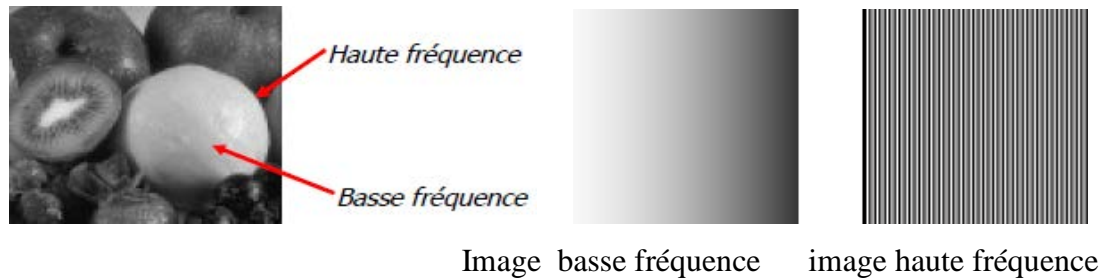


Figure 4.15. Contenu fréquentiel d'une image

4.4.2 Transformée de Fourier 2D

La transformée de Fourier permet la décomposition d'un signal image f en combinaison linéaire de sinusoides complexes, dont les coefficients $F[u, v]$ dit coefficients de Fourier, fournissent des informations sur les fréquences (u, v) et permettent des manipulations dans le domaine fréquentiel.

La transformée de Fourier de l'image $f(x, y)$, de largeur N et de hauteur M est donnée par :

$$F(u, v) = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} f(x, y) e^{-i2\pi(\frac{ux}{N} + \frac{vy}{M})}$$

La Transformée de Fourier se représente dans un espace fréquentiel bidimensionnel. Etant donné que la transformée est une grandeur complexe, sa représentation graphique se fait soit par le module et la phase, soit par la partie réelle et imaginaire de la transformée de Fourier.

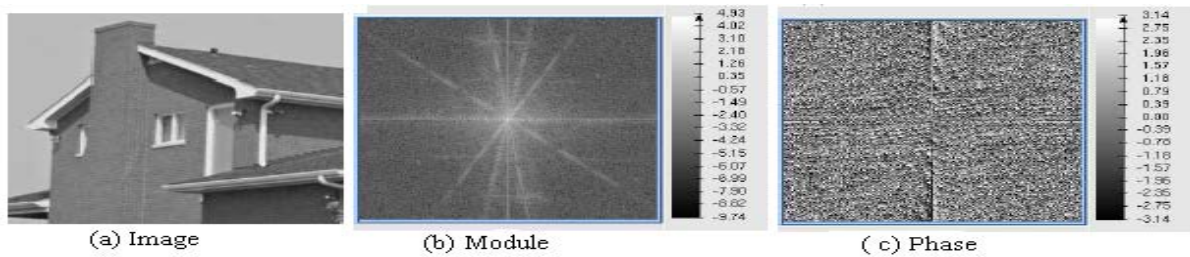


Figure 4.16. Spectre d'amplitude et de phase d'une image

Les valeurs de l'amplitude en général présentent une très grande dynamique (les basses fréquences sont plus importantes que les hautes fréquences), le spectre d'amplitude représente le logarithme du module de la transformée de Fourier.



Figure 4.17. Visualisation du spectre d'image

La transformée inverse du domaine fréquentiel au domaine spatial est donnée par :

$$f(x, y) = \frac{1}{MN} \sum_{u=0}^{N-1} \sum_{v=0}^{M-1} F(u, v) e^{i2\pi(\frac{ux}{N} + \frac{vy}{M})}$$

4.4.3 Interprétation de la Transformée 2D

- Hautes fréquences : loin du centre de la TF
- Basses fréquences : proche du centre de la TF
- Composante continue : centre de l'image (fréquence zéro = moyenne de l'image)

Exemples de Transformées de Fourier

1. image formée par un carré blanc
2. Image formée par une sinusoïde suivant la direction horizontale et verticale
3. Rotation d'une image formée de sinusoïdes dans la direction horizontale engendre une rotation de la TF (même angle).
4. L'image de gauche possède des lignes horizontales/verticales qu'on retrouve dans sa transformée, l'image de droite possède des lignes dans toutes les directions qu'on retrouve aussi dans sa transformée

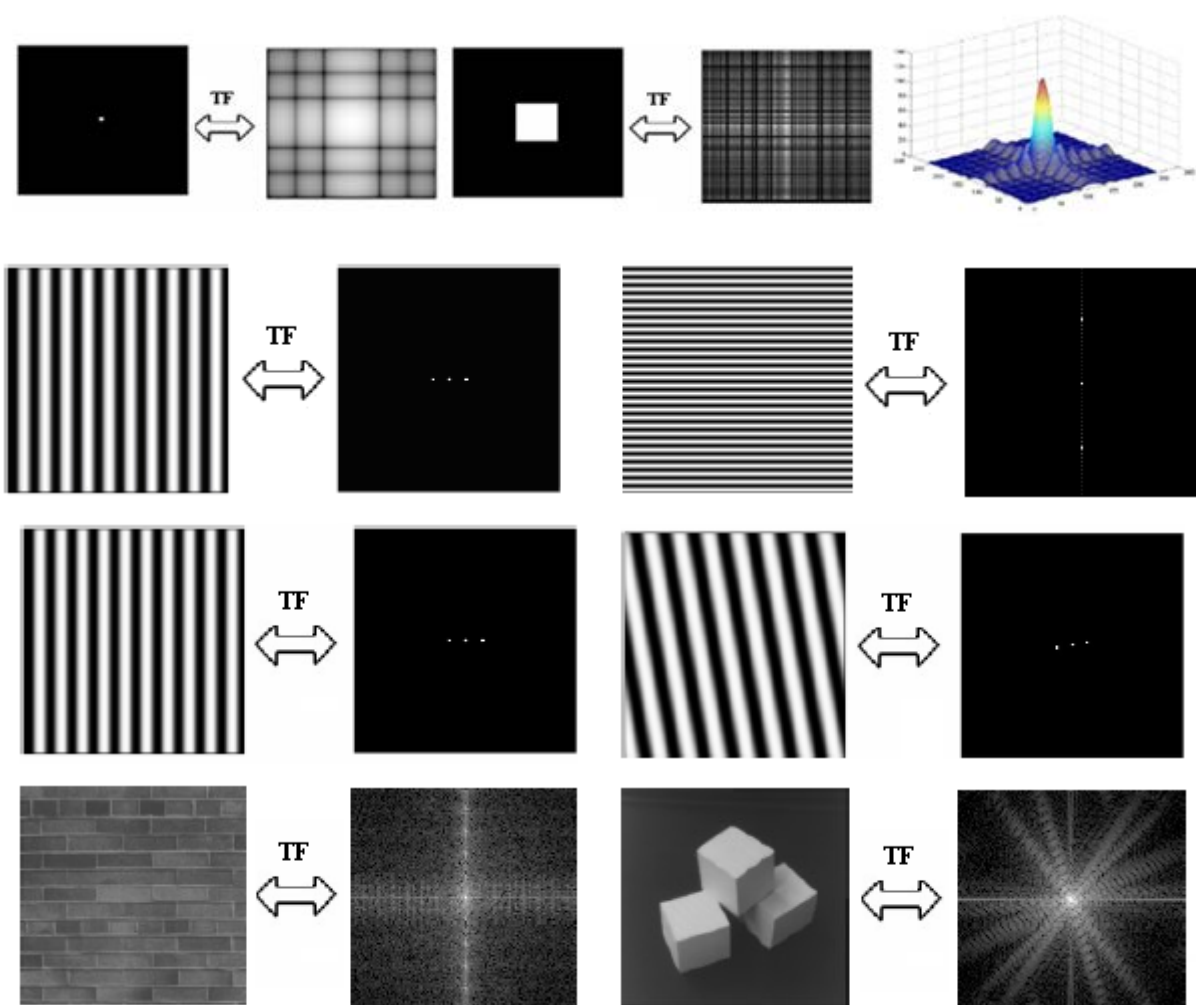


Figure 4.18. Exemples de Transformées de Fourier

Transformée de Fourier d'une convolution : La transformée de Fourier d'une convolution de deux signaux bidimensionnels est un produit des transformées de Fourier de ces deux signaux.

4.4.4 Classification de filtrage fréquentiel

Il est possible d'extraire la composante fréquentielle de l'image avec des filtres passe-haut, passe-bas ou passe bande. Un filtrage passe-haut laisse passer les hautes fréquences et atténue les basses fréquences (il a pour effet de faire apparaître les détails de l'image). Un filtrage passe-bas laisse passer les basses fréquences et atténue les hautes fréquences (il a pour effet de faire disparaître les détails de l'image).

Etant donné la propriété de linéarité de la transformée de Fourier, l'image peut être décomposée comme $IMG=BF+HF$.

a- Filtrage passe-bas

Le filtrage passe bas est la multiplication dans le domaine fréquentiel par une fonction porte dont la fonction de transfert est de la forme :

$$H(u,v) = \begin{cases} 1 & \text{si } \sqrt{u^2 + v^2} \leq D_0 \\ 0 & \text{sinon} \end{cases}$$

Avec D_0 : La distance de coupure par rapport à l'origine.

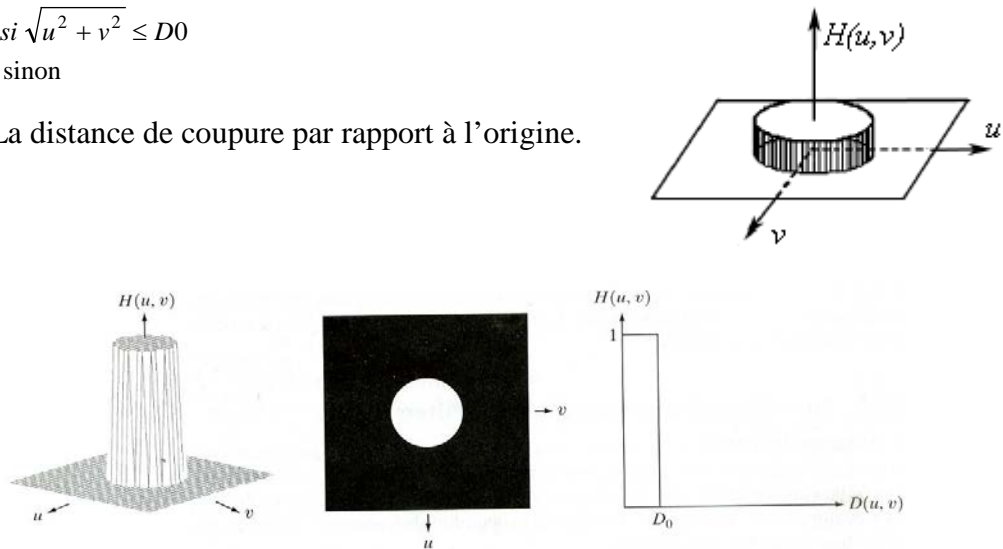


Figure 4.19. Représentation 3D du filtre idéal passe bas ; représentation 2D dans le plan (u, v) , et représentation 2D dans le plan (H,D) .



Figure 4.20. Image, son spectre et les trois images obtenues par des filtres passe bas de rayon 15,30 et 80

b- Filtres passe-haut

Le passage d'un filtre passe bas à un filtre passe haut s'obtient par la relation suivante :

$$H_{ph}(u,v) = 1 - H_{pb}(u,v)$$

Où $H_{ph}(u,v)$: Le filtre passe haut; $H_{pb}(u,v)$: Le filtre passe bas

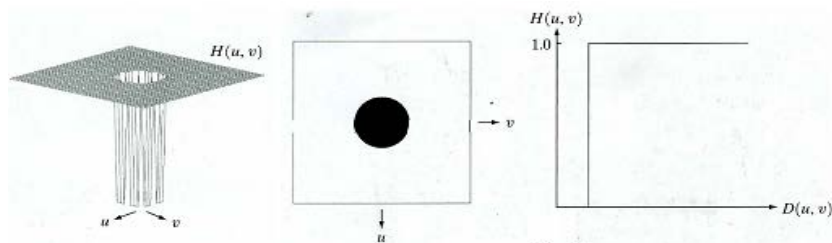


Figure 4.21. Représentation 3D du filtre idéal passe haut ; représentation 2D dans le plan (u, v) , et représentation 2D dans le plan (H,D) .

Le filtre passe-haut idéal est donné par :

$$H(u,v) = \begin{cases} 1 & \text{si } \sqrt{u^2 + v^2} \geq D_0 \\ 0 & \text{sinon} \end{cases}$$

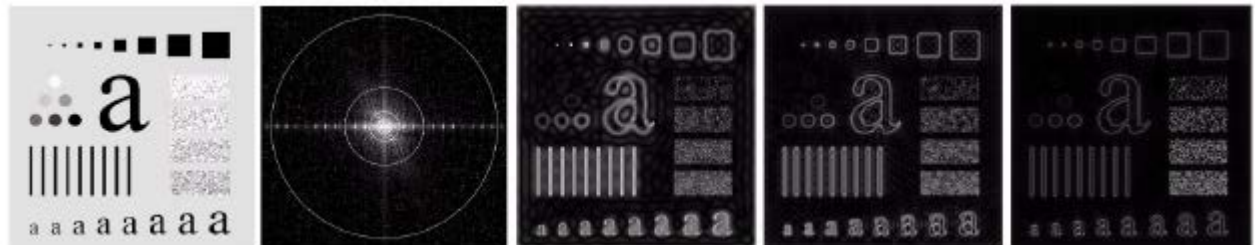
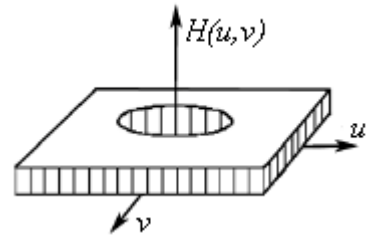


Figure 4.22. Image, son spectre et les trois images obtenues par des filtres passe haut de rayon 15,30 et 80

Chapitre 5

Détection de contours

5.1.Objectifs et généralités

La détection de contour est une étape préliminaire à de nombreuses applications de l'analyse d'images. Les contours dans une image proviennent de la discontinuité de la fonction d'intensité.

5.1.1. Contour

Par définition, un contour est la frontière qui sépare deux objets dans une image [4-5].

5.1.2. Détection de contours

La détection de contours est une technique qui consiste à transformer l'image en un ensemble de courbes formant les frontières significatives de l'image. Le principe de la détection des contours dans une image est de détecter et de représenter les changements (brusques) d'intensité [4-5].

5.2.Différents types de contour

Les contours sont caractérisés par des discontinuités de la fonction d'intensité .La figure 5.1 présente quelques types de contours : marche d'escalier, rampe et toit. Le plus utilisé est celui en marche d'escalier [4-5] .

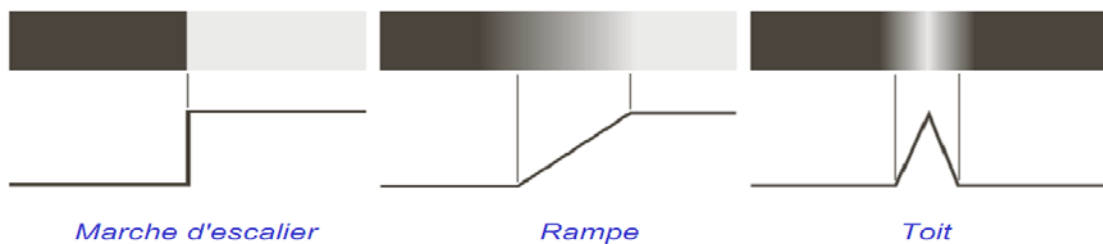


Figure 5.1. Différents types de contours : marche d'escalier, rampe et toit.

Le principe de la détection de contours repose donc sur l'étude des dérivées de la fonction d'intensité dans l'image, mais la difficulté réside dans la présence de bruit dans les images. La figure 5.2.présente la fonction d'intensité au voisinage d'un contour en marche d'escalier et ses dérivées première et seconde. Ainsi que la figure 5.3 présente différents types de contours sans bruit (figure5.3.a), contour avec un peu de bruit (figure 5.3.b) et contour au beaucoup de bruit (figure 5.3.c).

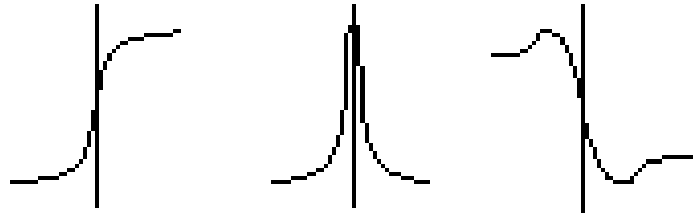


Figure 5.2. Fonction d'intensité au voisinage d'un contour en marche et ses dérivées première et seconde.

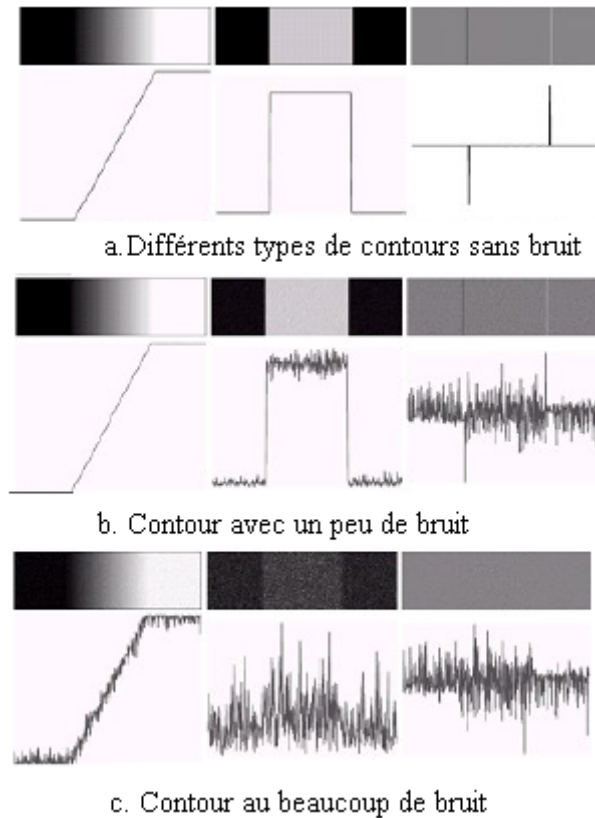


Figure 5.3. Effet du bruit sur le contour

5.3. Filtres différentiels

Les variations d'intensité constituent une source primordiale d'informations utilisée dans presque toutes les fonctions d'analyse d'images. Le calcul différentiel permet d'étudier le comportement local. En considérant l'image dans un repère orthogonal (Oxy) tel que (Ox) désigne l'axe horizontal et (Oy) l'axe vertical, le Gradient de l'image en tout point de

coordonnées (x, y) est désigné par : $\vec{\nabla}f = \left(\frac{\partial f}{\partial x} \quad \frac{\partial f}{\partial y} \right)^T$

Le module du gradient permet de quantifier l'importance du contour mis en évidence, c'est-à-

dire l'amplitude du saut d'intensité relevé dans l'image : $\|\vec{\nabla}f\| = \sqrt{\left(\frac{\partial f}{\partial x} \right)^2 + \left(\frac{\partial f}{\partial y} \right)^2}$

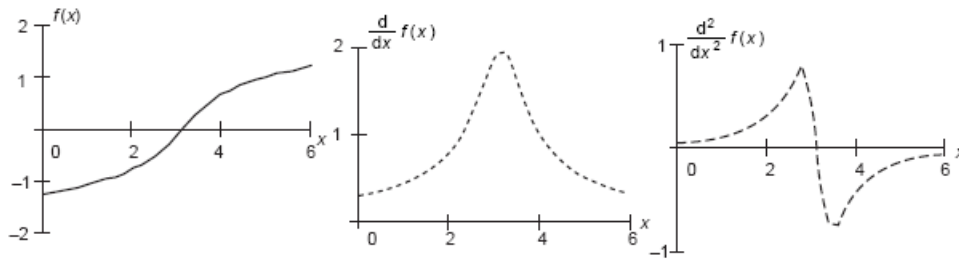
La direction du gradient permet de déterminer l'arête présente dans l'image. En effet, la

direction du gradient est orthogonale à celle du contour : $\alpha_0 = \arctan\left(\frac{\partial f / \partial y}{\partial f / \partial x}\right)$

Les dérivées secondes qui s'expriment par la fonction matricielle le Hessian :

$$H = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix}$$

Le Laplacien est le trace du Hessian : $\Delta f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$



5.3.1. Opérateurs de gradient par masques

Pour chaque opérateur, deux masques sont utilisés de façon à déterminer le gradient de l'image dans deux directions orthogonales.

Simple différences finies

$$\text{Approximation en } x : \frac{\partial f}{\partial x} = \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x, y) - f(x, y)}{\Delta x} = \lim_{\Delta x \rightarrow 0} \frac{f(x, y) - f(x - \Delta x, y)}{\Delta x}$$

Pour $\Delta x = 1$, le masque de convolution pour avoir le gradient selon x est : $[1 \ -1]$ ou $[-1 \ 1]$

De la même manière, le masque de convolution pour avoir le gradient selon y est : $[1 \ -1]^T$ ou $[-1 \ 1]^T$

Développement de la formule de Lagrange (au 1^{er} ordre) :

$$f(x) \approx \frac{f(x+1) - f(x-1)}{2}, \quad \frac{\partial f}{\partial x} = \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x, y) - f(x - \Delta x, y)}{2\Delta x}$$

Pour $\Delta x = 1$, le masque de convolution pour avoir le gradient selon x est : $[1 \ 0 \ -1]$

De la même manière, le masque de convolution pour avoir le gradient selon y est : $[1 \ 0 \ -1]^T$.

Exemple : Calculer pour le pixel central la norme du gradient et sa direction ($[1 \ 0 \ -1]$ et $[1 \ 0 \ -1]^T$).

On trouve, pour le pixel central $G_x = -5$ et $G_y = 5$.

Donc, $|G| = 10$ et $\theta = \arctan(-1) = -\pi/4$

$$\begin{array}{c} \nearrow x \\ y \searrow \end{array} \begin{bmatrix} 3 & 2 & 1 \\ 7 & 5 & 1 \\ 8 & 7 & 3 \end{bmatrix}$$

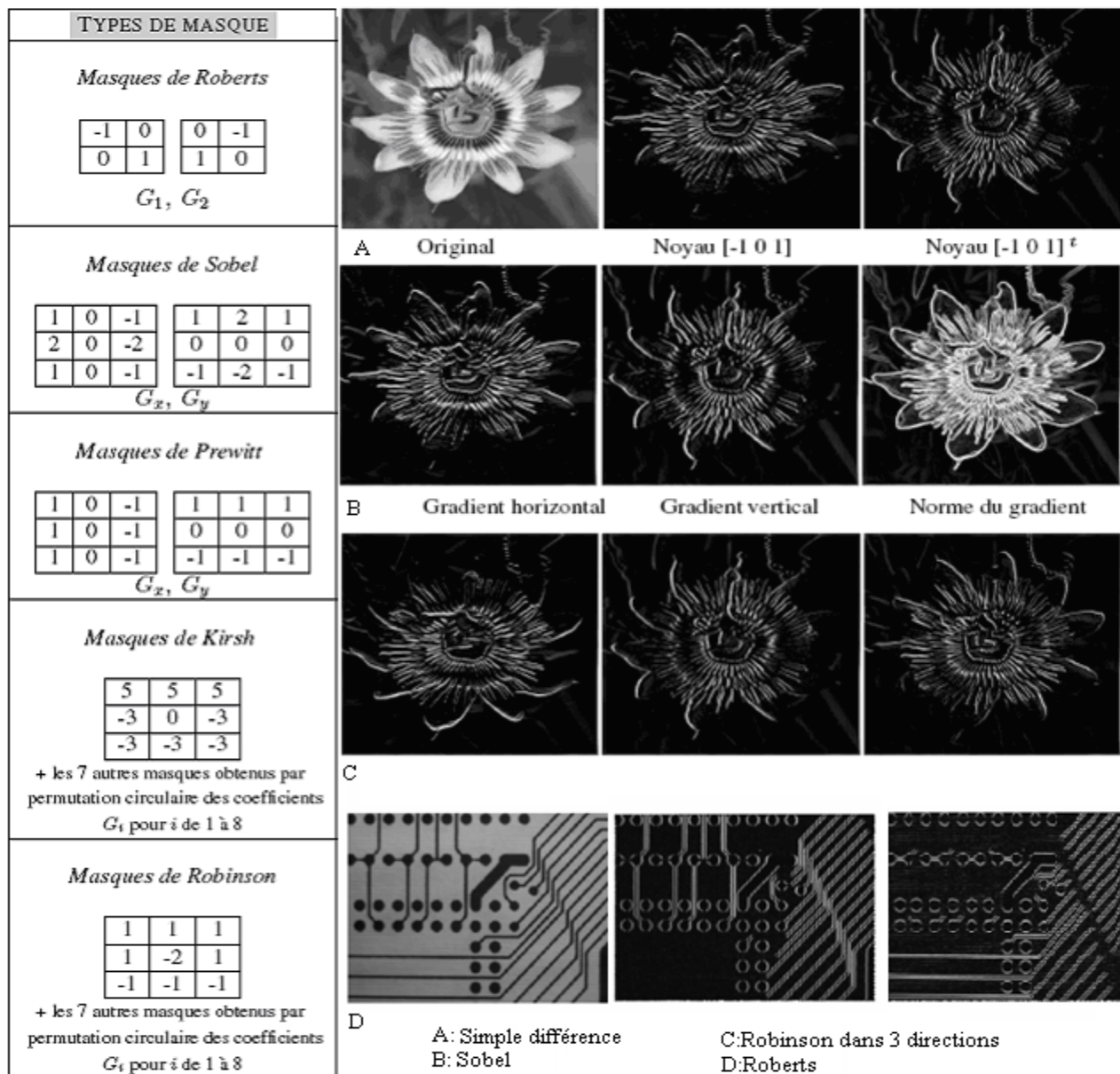


Figure 5.4. Application du masque gradient

Opérateur gradient boussole

Les opérateurs dits boussole mesurent le gradient dans des directions sélectionnées. L'image est successivement filtrée par un ensemble de masques $m_k(i,j)$ dont chacun représente une approximation discrète d'un contour idéal dans une orientation spécifique (voir figure ci-après). Le résultat du filtrage de l'image $f(i,j)$ avec le $k^{\text{ième}}$ masque est $g_k(i,j)$.

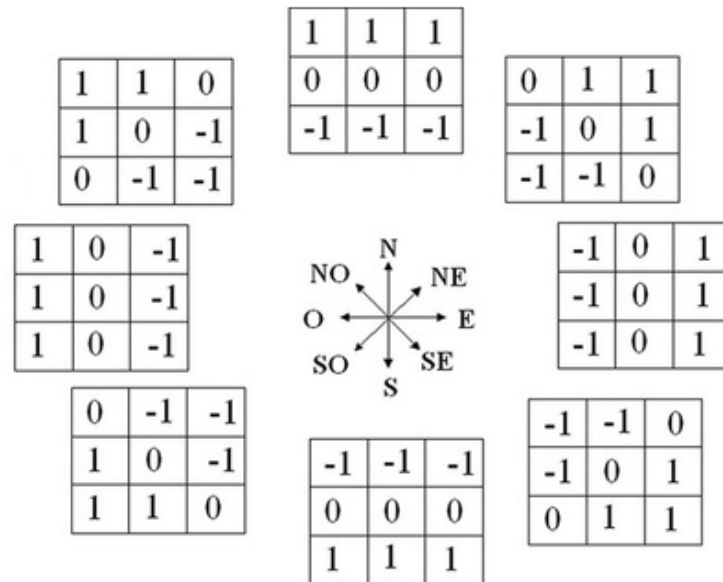


Figure 5.5. Masque du gradient boussole

5.3.2. Filtre Laplacien

De la même façon, l'approximation par différences finies la plus simple de la dérivée seconde est la convolution par le noyau $\begin{bmatrix} 1 & -2 & 1 \end{bmatrix}$ pour l'approximation de $\frac{\partial^2 f}{\partial x^2}$ et $\begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix}$ pour

l'approximation de $\frac{\partial^2 f}{\partial y^2}$

Le laplacien $\Delta f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$ donc être approché par l'un des opérateurs linéaires suivants :

<i>Laplacien discret - 4</i>	<i>Laplacien discret - 8</i>	<i>Laplacien de Robinson</i>																											
<table border="1"> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>-4</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> </table>	0	1	0	1	-4	1	0	1	0	<table border="1"> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>-8</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	1	1	1	1	-8	1	1	1	1	<table border="1"> <tr><td>1</td><td>-2</td><td>1</td></tr> <tr><td>-2</td><td>4</td><td>-2</td></tr> <tr><td>1</td><td>-2</td><td>1</td></tr> </table>	1	-2	1	-2	4	-2	1	-2	1
0	1	0																											
1	-4	1																											
0	1	0																											
1	1	1																											
1	-8	1																											
1	1	1																											
1	-2	1																											
-2	4	-2																											
1	-2	1																											

5.4. Opérateurs du premier ordre (Approche gradient)

Un très grand nombre d'opérateurs de détection de contours est disponible dans la littérature, chacun étant conçu pour être sensible à certains types de contours. On se limite uniquement dans cette partie aux techniques différentielles où l'estimation des dérivées partielles du premier ordre par les différences finies sera considérée. Cette approche se décompose en deux étapes ; calculer la norme du gradient en tous point de l'image, puis sélectionner les pixels à l'aide d'un seuil fixé a priori pour la norme du gradient. La figure 5.6 présente un exemple de détecteur de contours par gradient.

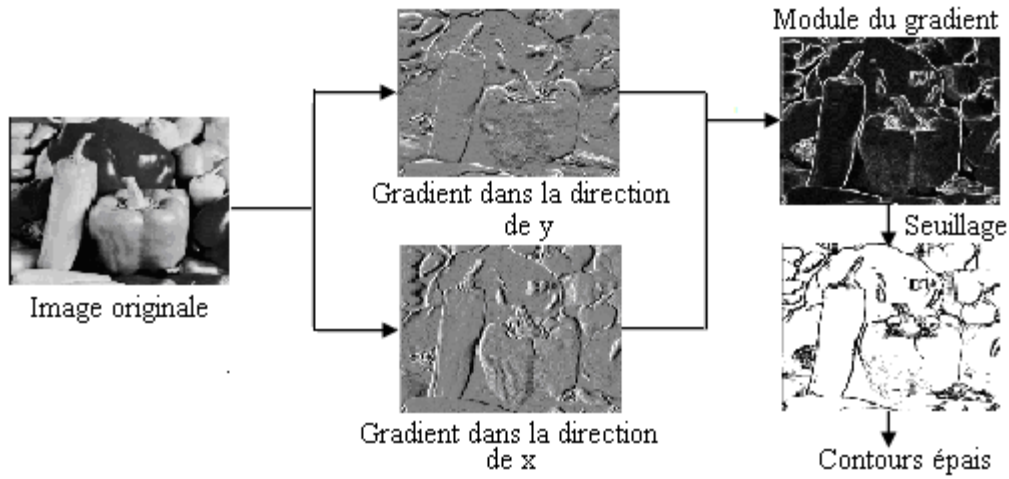


Figure 5.6. Schéma de détecteur de contours par gradient.

Le calcul de dérivée nécessite un pré filtrage des images. Le but est de diminuer le bruit tout en préservant les contours:

- Filtrage linéaire pour les bruits de moyenne nulle, par exemple pour un bruit blanc Gaussien on utilise un filtre Gaussien.
- Filtrage non linéaire pour les bruits impulsionnels comme le filtre médian.

5.4.1. Calcul de la norme du gradient

Pour calculer le gradient en chaque point de l'image, on utilise un masque de petite dimension avec lequel l'image considérée sera convoluée. Nous nous sommes limitées uniquement à la description des masques de gradient de Roberts, de Prewitt et de Sobel. Pour chaque opérateur deux masques sont définis de façon à pouvoir estimer le gradient de l'image dans les deux directions orthogonales. Dans le domaine discret, nous pouvons approximer les dérivées partielles premières I_x et I_y de l'image I en chaque point (x, y) par la façon la plus simple:

$$\begin{cases} \frac{\partial I(x, y)}{\partial x} = I_x(x, y) = G_x(x, y) = (I * h_x)(x, y) \\ \frac{\partial I(x, y)}{\partial y} = I_y(x, y) = G_y(x, y) = (I * h_y)(x, y) \end{cases}$$

où $*$ représente le symbole de convolution. Cette approximation revient à convoluer l'image par deux masques h_x et h_y qui représentent respectivement une estimation du gradient suivant l'axe horizontal et vertical.

La norme du gradient G en chaque point (x, y) peut être obtenue de trois manières :

$$G(x, y) = \sqrt{G_x(x, y)^2 + G_y(x, y)^2}$$

$$G(x, y) = |G_x(x, y)| + |G_y(x, y)|$$

$$G(x, y) = \max(|G_x(x, y)|, |G_y(x, y)|)$$

Les masques de gradient sont appliqués sur des fenêtres de 2 x 2 pixels ou 3 x 3 pixels. On note que la somme des coefficients de ces filtres est nulle, et que les coefficients sont anti-symétriques. Les filtres les plus utilisés sont :

1. Filtre dérivateur Roberts : Les filtres de Roberts sont les suivants:

$$\text{Filtre diagonal : } h_y = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

$$\text{Filtre anti-diagonal : } h_x = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

2. Filtre dérivateur Prewitt : Ce filtre comporte deux masques de convolution, un pour la détection des contours verticaux h_x ou h_y et un autre pour la détection des contours horizontaux h_y ou h_h .

$$\text{Filtre horizontal : } h_h = \frac{1}{3} \cdot \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} = \frac{1}{3} \cdot \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$$

$$\text{Filtre vertical : } h_v = \frac{1}{3} \cdot \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} = \frac{1}{3} \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -1 \end{bmatrix}$$

Les filtres de Prewitt sont séparables. La décomposition de h_v fait apparaître clairement :

- Une fonction de lissage dans une direction, lissage associé au vecteur $[1 \ 1 \ 1]/\sqrt{3}$. Ce lissage rend ces masques un peu moins sensibles au bruit que le précédent.
- Une fonction de dérivation dans l'autre direction, dérivation associée au vecteur $[1 \ 0 \ -1]/\sqrt{3}$.

3. Filtre dérivateur de Sobel : Ce filtre comporte deux masques de convolution, un pour la détection des contours verticaux h_v et un autre pour la détection des contours horizontaux h_h .

$$\text{Filtre horizontal : } h_h = \frac{1}{4} \cdot \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} = \frac{1}{4} \cdot \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} [1 \ 2 \ 1]$$

$$\text{Filtre vertical : } h_v = \frac{1}{4} \cdot \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} = \frac{1}{4} \cdot \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} [1 \ 0 \ -1]$$

Les filtres de Sobel sont séparables. On peut voir qu'ils effectuent une dérivation associée au vecteur $[1 \ 0 \ -1]/2$ dans une direction et un lissage associé au vecteur $[1 \ 2 \ 1]/2$ dans la direction orthogonale.



Figure 5.7. Exemples de détections de contours

5.4.2. Seuillage

Les **post-traitements** commencent le plus souvent par l'élimination des points de contour trop faibles par un seuillage. Différentes méthodes de seuillages existent, parmi lesquelles on cite les seuillages globales. Ces méthodes globales de seuillage peuvent être classées en deux catégories:

1. Seuillage globales simple : La première consiste à fixer une valeur de seuil S entre 0 et 1 d'une façon un peu arbitraire. Au préalable, il est nécessaire de normaliser l'image du module du gradient afin que tous les pixels se trouvent également dans l'intervalle $[0, 1]$. Dans ce cas, si la valeur du module du gradient en un pixel (x, y) dépasse le seuil fixé S alors la valeur résultante du pixel est égale à 1. Dans le cas contraire, cette valeur sera fixée à 0, c-à-d:

$$G(x, y) = \begin{cases} 1 & \text{si } G(x, y) \geq S \\ 0 & \text{si } G(x, y) < S \end{cases}$$

Ainsi le résultat du seuillage est une image binaire contenant des pixels noirs et blancs, c'est la raison pour laquelle le terme de binarisation est parfois employé. Le seuillage permet de mettre en évidence des formes ou des objets dans une image. Cette méthode est simple à

appliquer sauf qu'elle est très dépendante du choix du seuil. La figure 5.8. présente une image en 256 niveaux de gris et les résultats d'une opération de seuillage avec les valeurs respectives de seuil de 0.1 et 0.3

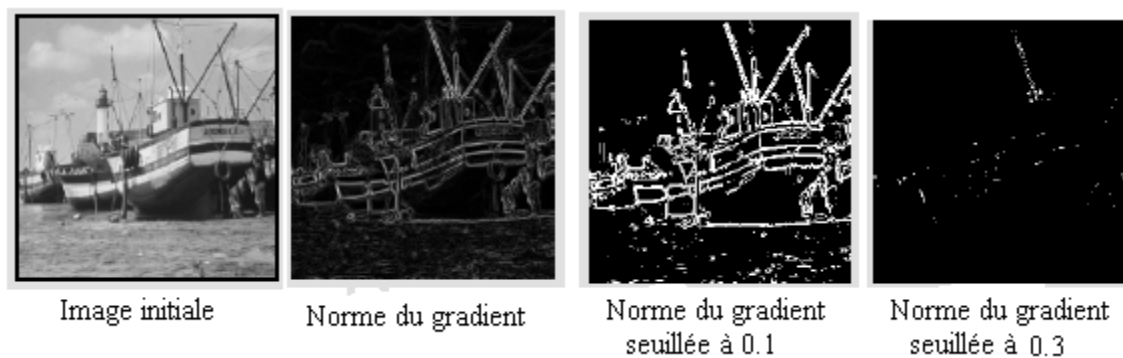


Figure 5.8. Exemple de résultat d'un seuillage global

2. Seuillage global par histogramme : Pour pallier ce problème, un seuillage global par histogramme peut alors être effectué. L'idée est de tracer tout d'abord l'histogramme du module de gradient. Cet histogramme consiste à représenter le nombre de pixel pour chaque intensité lumineuse relative à l'image gradient. Ensuite, une détermination de l'histogramme cumulé du module du gradient permet alors le choix d'un pourcentage de pixels. Ce pourcentage permet alors de choisir la valeur du seuil S à utiliser. Même si cette méthode est une aide au choix du seuil, elle n'est pas forcément pertinente puisqu'elle impose une seule valeur de seuil à appliquer à toute l'image. La figure 5.9. Présente un exemple de résultat d'un seuillage global par histogramme.

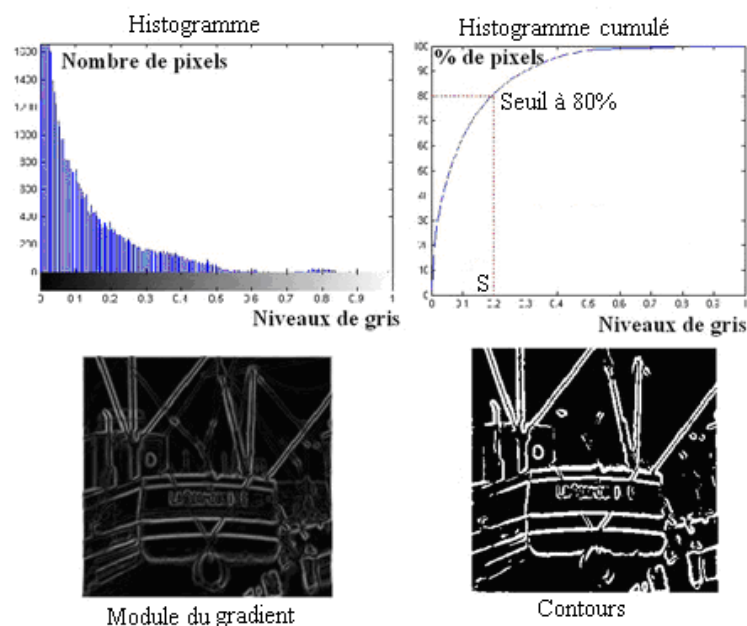


Figure 5.9. Exemple de résultat d'un seuillage global par histogramme.

5.5. Opérateurs de second ordre

5.5.1. Opérateurs Laplacien

Le laplacien est la dérivée seconde de l'intensité de l'image f qui est donné par

$$\Delta f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

Une approximation numérique du Laplacien pour un voisinage de 3x3 peut se faire par l'utilisation des masques suivants :

<i>Laplacien discret - 4</i>	<i>Laplacien discret - 8</i>	<i>Laplacien de Robinson</i>																											
<table border="1"> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>-4</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> </table>	0	1	0	1	-4	1	0	1	0	<table border="1"> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>-8</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	1	1	1	1	-8	1	1	1	1	<table border="1"> <tr><td>1</td><td>-2</td><td>1</td></tr> <tr><td>-2</td><td>4</td><td>-2</td></tr> <tr><td>1</td><td>-2</td><td>1</td></tr> </table>	1	-2	1	-2	4	-2	1	-2	1
0	1	0																											
1	-4	1																											
0	1	0																											
1	1	1																											
1	-8	1																											
1	1	1																											
1	-2	1																											
-2	4	-2																											
1	-2	1																											

Les points de contour sont caractérisés par des passages par zéro du laplacien. L'image de contours est obtenue par :

1. Détection des passages par zéros du Laplacien.
2. Seuillage des passages par zéros de fortes amplitudes .

l'estimation de la dérivée seconde étant très sensible aux bruits, il convient de filtrer l'image par un filtre Gaussien avant d'en mesurer le laplacien.

5.5.2. Filtre de Marr-Hildreth

Le filtre de Marr et Hildreth [1] consiste en une opération de filtrage préalable de l'image par l'opérateur isotrope Gaussien $G(x,y)$ suivi par une localisation des passages par zéro du Laplacien.

Cela conduit au filtrage suivant :

$$G * \left(\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \right) = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) [G * f] = \left(\frac{\partial^2 G}{\partial x^2} + \frac{\partial^2 G}{\partial y^2} \right) * f = LoG_{\sigma} * f$$

Où $G(x, y) = \frac{1}{2\pi\sigma^2} e^{\left(\frac{-(x^2+y^2)}{2\sigma^2}\right)}$ est un filtre passe-bas de type gaussien de variance , car le lissage gaussien réalise le meilleur compromis entre détection et localisation.

Le filtre obtenu par convolution avec le laplacien d'une gaussienne est connu sous le nom de LOG(Laplacian of Gaussian).

Donc l'image de contours est obtenue par filtrage de l'image par la dérivée seconde du filtre gaussien , puis détection des zéros de la fonction ainsi obtenue.Cette méthode perd la propriété de fermeture présentée par les lignes de passages par 0 du Laplacien.

5.6.Opérateurs Laplacien vs Opérateur de gradient

Le gradient est un opérateur qui permet une bonne localisation en fournissant des contours fins, cependant il est sensible au bruit d'où un lissage préliminaire est nécessaire ce qui engendre des contours non fermés. Pour le Laplacien même s'il fournit des lignes fermées, il est très sensible au bruit, d'où la nécessité d'un lissage fort ce qui peut affecter la localisation des contours.

5.7.Filtre optimal

Pour améliorer la qualité des méthodes dérivatives et pallier aux problèmes de précision et de localisation des, autre type d'approche de détection de contour repose sur la définition de critères d'optimalité de la détection de contours; ces critères débouchant sur des filtres de lissage optimaux.

5.7.1. Critères d'optimalité

Les critères d'optimalités de Canny sont :

Bonne détection : le contour doit être détecté, il faut minimiser les fausses réponses

Bonne localisation : le contour doit être localisé avec précision, il s'agit de minimiser la distance entre les points détectés et le vrai contour.

Réponse unique : Ce critère correspond à la minimisation du nombre de réponse à un seul contour.

Plusieurs filtres optimaux sont apparus, on trouve les filtre de Canny , de Deriche et celui de Shen et Castan.

5.7.2. Filtre de Canny

L'algorithme de Canny, est conçu pour être optimal suivant les trois critères cités auparavant, la mise en œuvre de ce filtre est la suivante :

1. Appliquer un filtre Gaussien sur l'image pour réduire le bruit de l'image originale avant d'en détecter les contours.
2. Calculer le gradient de suivant les directions x et y en se basant sur le filtre de Sobel.

La valeur du gradient en un point est approximée par la formule : $G = |G_x| + |G_y|$

3. Calculer les directions du gradient dans l'image $\theta = \arctan(G_y/G_x)$ et arrondir les directions par multiples de 45°

4. Supprimer des non-maxima : Si la norme du gradient en un pixel (x, y) est inférieure à la norme du gradient d'un de ses deux voisins le long de la direction du gradient, alors mettre la norme pour le pixel (x, y) à zéro.
5. Seuillage des contours (Utiliser deux seuils : un seuil haut (Sh) et un seuil bas (Sb))

Pour chaque point, si l'intensité de son gradient est :

- Inférieur au seuil bas Sb , alors le pixel est mis à zéro (noncontour)
- Supérieur au seuil haut Sh , alors le pixel est contour
- Entre le seuil bas et le seuil haut, le point est accepté s'il est connecté à un point déjà accepté.

La taille du filtre de réduction du bruit ainsi que le choix du seuil détermineront l'efficacité de l'algorithme et le temps de calcul nécessaire.

5.7.3. Filtre de Deriche

Le filtre de Deriche est un opérateur optimal sous forme de filtre à réponse impulsionnelle infinie (RII), il est exprimé sous la forme suivante :

$$h(x) = c(\alpha |x| + 1) e^{-\alpha|x|}$$

avec : $c = \frac{(1 - e^{-\alpha})^2}{1 + 2\alpha e^{-\alpha} - e^{-2\alpha}}$

et le filtre de dérivation s'écrit : $h'(x) = c'x e^{-\alpha|x|}$

avec : $c' = -\frac{(1 - e^{-\alpha})^2}{e^{-\alpha}}$

Où α est un paramètre d'échelle . En effet plus α est petit plus le lissage est fort, plus α est grand, plus le lissage est faible.

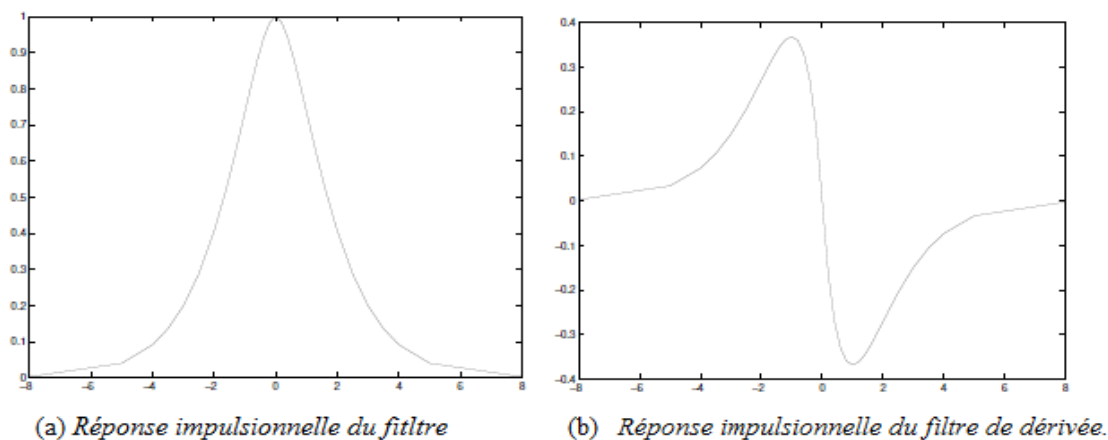


Figure 5.10. Réponse impulsionnelle du filtre de Deriche

Le schéma en blocs du gradient de Deriche est présenté en figure.5.11. I étant l'image initiale, h_x et h_y les réponses impulsionnelles du filtre h selon les directions x et y .

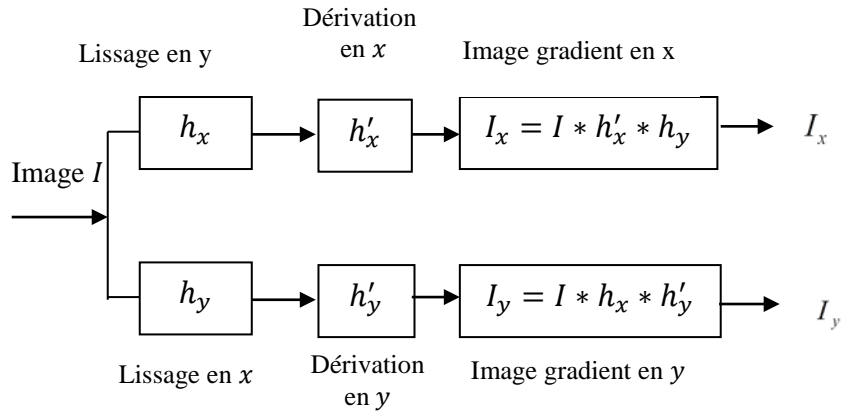


Figure 5.11. Processeur du gradient de Deriche.

La figure 5.12. représente les résultats de l'estimation des dérivées suivant x et y par le filtre de Deriche. Les dérivées estimées se trouvent moins bruitées et plus lisses.



Figure.5.12. Gradient de Deriche suivant x (gauche) et suivant y (droite)



Figure 5.13. Seuillages des extrema locaux pour différentes valeurs du paramètre du filtre de Deriche : (a) $\alpha = 0.5$., (b) $\alpha = 1.5$.

Chapitre 6

Segmentation et classification

6.1. Principe de segmentation

La segmentation d'images est l'opération la plus importante dans un système de traitement d'images, car elle est située à l'articulation entre le traitement et l'analyse des images. La segmentation consiste à créer une partition d'une image A en sous-ensembles R_i appelés régions tel que est de partitionner une image en plusieurs régions

$$\forall i \ R_i \neq \emptyset$$

$$\forall i, j \ R_i \cap R_j = \emptyset$$

$$A = \cup_i R_i$$

Une région est un ensemble convexe de points images ayant les propriétés communes (intensité, texture, ...).

6.2. Techniques de segmentation

On peut distinguer quatre grandes approches de segmentation : par seuillage, par régions, par contour (frontière) et approche de la classification.

6.2.1. Approche contour

Il s'agit dans cette approche de rechercher les contours qui délimitent les régions homogènes de l'image. Un contour est un ensemble de pixels formant une frontière entre deux ou plusieurs régions voisines. Cependant, cette méthode ne conduit pas directement à une segmentation de l'image car les contours obtenus sont rarement convexes, il faut donc procéder à une fermeture des contours si on souhaite une partition complète de l'image.

Il existe plusieurs méthodes de segmentation basées sur l'approche contour qu'on peut regrouper en trois catégories : les méthodes dérivatives, par filtrage optimal et les contours actifs.

6.2.2. Approche région

Contrairement à la segmentation d'images par l'approche contour, la segmentation par région consiste à décomposer l'image en régions selon des critères d'homogénéité.

On distingue quatre types de méthodes :

✓ Croissance des régions

Appelé aussi méthode d'agrégation du pixel. C'est une méthode ascendante, elle consiste à choisir initialement un ensemble de pixels comme des noyaux des régions à partir desquels

la croissance des régions s'effectuera. Chaque région doit respecter un critère d'homogénéité comme la variance $var(R)$ du niveau de gris de l'image associée au point de la région R est inférieure à un seuil.

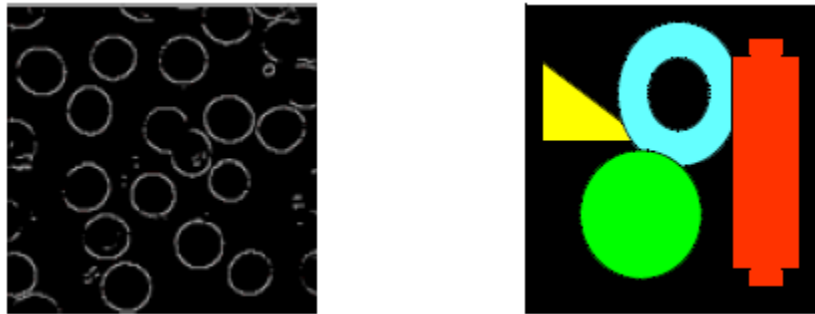


Figure 6.1. Segmentation par (a) approches contours (b) approches régions

✓ Division des régions

Division des régions ce sont des méthodes descendantes consistent à diviser l'image ou une partie de l'image en région en utilisant des partitions élémentaires connues comme le maillage ou l'arbre quaternaire (quadtree). le maillage peut être régulier ou irrégulier de type rectangulaire, triangulaire ou polygone quelconque. La division d'une région R en sous régions se fait si R ne remplit pas la contrainte d'homogénéité fixée. La décision peut aboutir à un nombre important de régions, la solution consiste à fusionner les régions adjacentes similaires. La figure () montre le principe de la méthode.

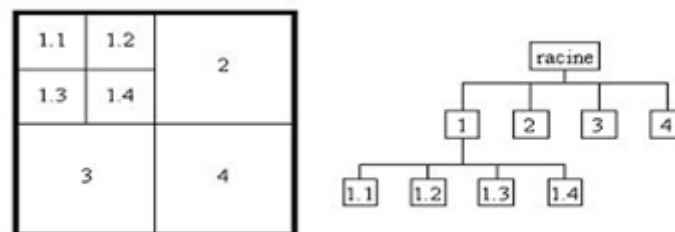


Figure 6.2. Principe de la segmentation par division des régions

6.2.3. Segmentation d'images par classification

Cette technique consiste à regrouper des pixels ayant des vecteurs caractéristiques aussi similaires que possible sans la considération de la disposition spatiale des pixels. De ce fait, les pixels appartenant à une classe peuvent former plusieurs régions non-adjacentes dans l'image mais ayant les mêmes propriétés. Plusieurs algorithmes sont utilisés dans la segmentation par classification, on cite entre autres l'algorithme K-means.

6.2.4. Segmentation par seuillage

La segmentation par seuillage permet de répartir les pixels en classes en fonction de leurs niveaux de gris. Les classes sont alors déterminées par des seuils.

Soit $\{0,1,2, \dots L - 1\}$ l'ensemble des niveaux de gris de l'image I , la segmentation par seuillage consiste à répartir les pixels en K classes ($C_1, C_2, \dots C_K$) à partir d'un ensemble de seuils $T = (t_1, t_2, \dots t_{K-1})$. Un pixel de niveau de gris $I(x, y)$ est affecté à la classe C_k si $t_k \leq I(x, y) < t_{k+1}$ avec $k = 0, 1, 2, \dots K - 1$ et $t_0 = 0$ et $t_K = L - 1$.

Si le nombre de seuils est supérieur à 1, on parle de multi-seuillage ou multi-level thresholding.

6.3. Seuillage d'images

Le seuillage (Thresholding en Anglais) permet d'extraire des objets de leurs fonds en fonction d'un seuil. Le problème majeur dans le seuillage réside dans la détermination du seuil. Plusieurs méthodes ont été proposées pour le calcul d'un seuil puis étendues au problème du multi-seuillage.

Parmi les méthodes de seuillage on cite :

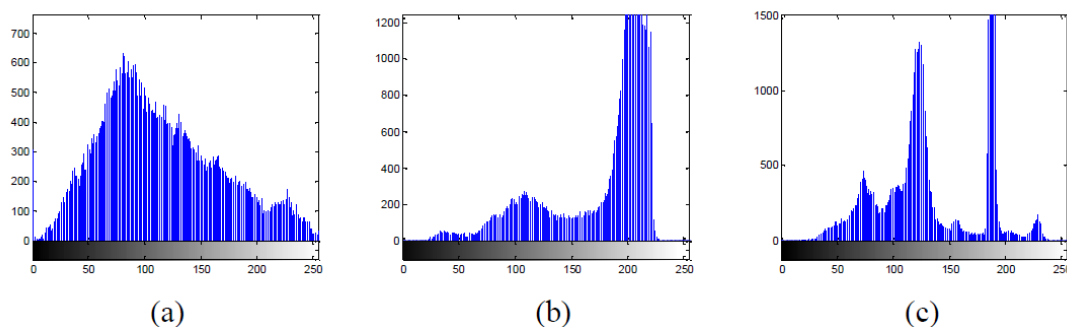
6.3.1. Seuillage globale

Pour les méthodes de seuillage globale on définit un même seuil pour tous les pixels de l'image sur l'exploitation de l'histogramme de toute l'image.

A partir de l'allure de l'histogramme, on peut déduire le nombre de classes ainsi que la position relative des seuils.

- Histogramme uni-modal → formé d'un seul pic → Objet ou fond
- Histogramme bimodal → caractérisé par deux modes séparés par une vallée → Objet sur un fond
- Histogramme multi-modal → comporte plus de deux modes séparés par des vallées → plusieurs classes

Les seuils doivent être localisés dans les vallées.



Histogramme unimodal

Histogramme bimodal

Histogramme multimodal

Figure 6.3. Différents types d'histogramme

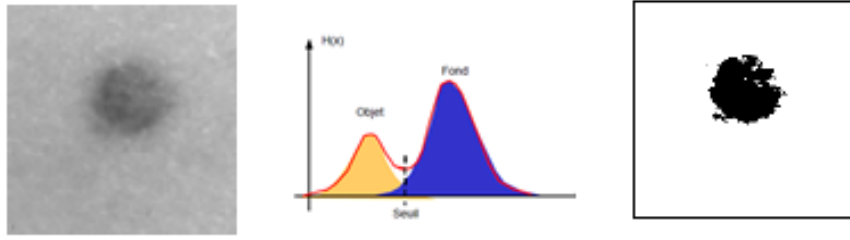


Figure 6.4. Exemple de seuillage en 2 classes par histogramme

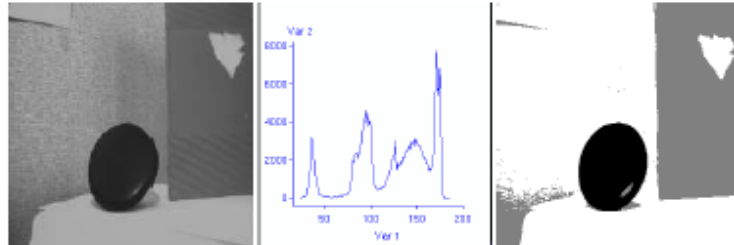


Figure 6.5. Exemple de seuillage en 4 classes par histogramme

6.3.2. Seuillage local

Dans le seuillage locale, la valeur des pixels voisins est considérée pour le calcul des seuils. La méthode consiste à diviser l'image en blocs non chevauchants, puis pour chaque bloc un seuil est déterminé selon l'histogramme local i.e. du bloc en question. Une autre approche consiste à déterminer un seuil pour chaque bloc en fonction des pixels voisins. Pour chaque pixel centré $I(x, y)$, le seuil $t(x, y)$ est évalué en fonction de la moyenne et l'écart-type des niveaux de gris des pixels de la fenêtre :

$$t(x, y) = \mu(x, y) + \alpha\sigma(x, y)$$

avec α un parametre réel fixe.

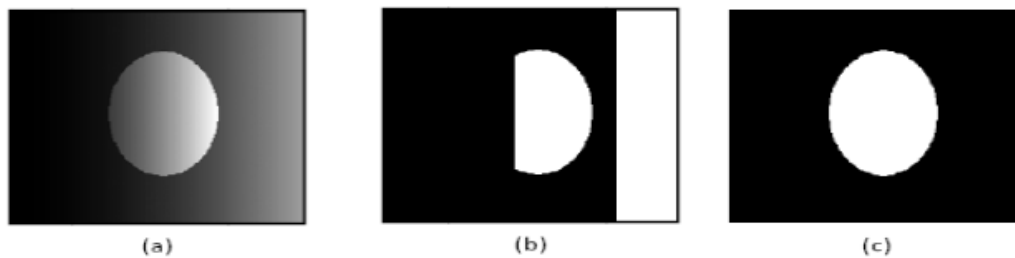


Figure 6.6. Illustration du seuillage local (a) image originale , (b) image segmentée avec un seuil global , (c) image segmentée avec un seuil local.

6.3.3. Seuillage par détection de vallées

On suppose que chaque classe correspond à une plage de niveaux de gris. La position des minima de l'histogramme H permet de fixer les $(m - 1)$ seuils nécessaires pour séparer les m classes. les seuils t_i sont obtenus par :

$$H(t_i) = \text{Min} [H(k)] \text{ pour } k \in]m_i, m_{i+1}[$$

où m_i et m_{i+1} sont les valeurs moyennes (ou les modes) de l'intensité lumineuse dans les classes C_i et C_{i+1} .

Malgré le développement de techniques robustes visant à faciliter la détection des vallées, cette méthode, bien que simple, est très peu appliquée car les histogrammes traités sont le plus souvent bruités et uni-modaux.

6.3.4. Seuillage par minimisation de variance

Méthode de Fisher : L'objectif de la méthode est de trouver le seuil t qui minimise la somme des moments d'ordre 2 de deux classes (meilleur seuillage dans cette approche correspond à une minimisation de la variance intra-classe).

Soit $H(x)$ l'histogramme de l'image, la variance d'une classe i est donné par :

$$var_i(s) = \sum_{x \in C_i} (x - G_i)^2 H(x)$$

où $G_i(s)$ est le centre de gravité de la classe i qui est donné par :

$$G_i(s) = \frac{\sum_{x \in C_i} x H(x)}{H(x)}$$

Le problème revient à trouver le seuil S qui minimise la somme des variances :

$$S_{opt} = \min_s (var_1(s) + var_2(s))$$

Plus récemment, Otsu a proposé de réaliser une maximisation de la variance inter-classe.

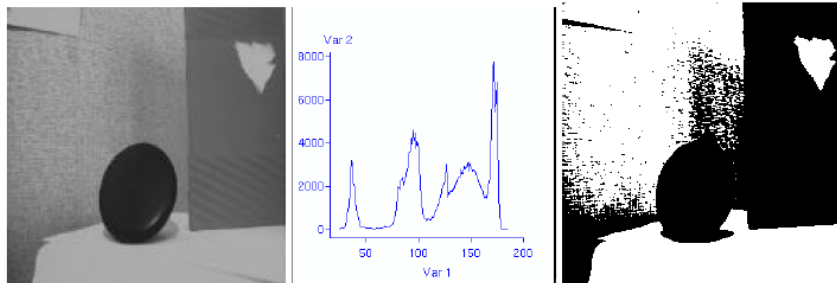


Figure 6.7. Exemple de seuillage par minimisation de variance

6.4. Morphologie mathématique

La morphologie mathématique est un outil qui permet d'explorer la structure géométrique des objets dans une image en utilisant la théorie des ensembles. De ce fait les opérateurs de la mise en œuvre sont des opérateurs ensemblistes.

6.4.1. Élément structurant

L'élément structurant est un ensemble ou surface de géométrie connue. Il est caractérisé par sa forme et sa taille et son origine. Pour la forme on peut avoir un diamant, un disque, une ligne ...

Si A est l'objet à traiter et B l'élément structurant, la morphologie mathématique consiste à déplacer B sur A comme un opérateur de convolution. A chaque position de B, des opérations logiques sont effectuées entre A et B

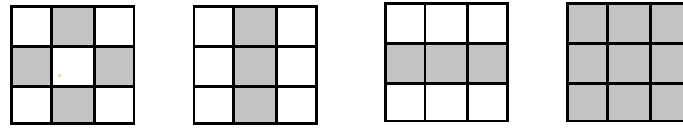


Figure 6.8. Exemples d'éléments structurants

6.4.2. Operations morphologiques de base

Dilatation

La dilatation sert à grandir les objets dans une image binaire. Donc elle permet d'ajouter à l'objet des pixels qui lui sont proche dans des directions qui dépendent de l'élément structurant. Ce dernier parcourt l'objet en gardant son centre à l'intérieur de l'objet.

Soient A et B deux ensembles, on définit la dilatation de A par B, notée $A \oplus B$, par :

$$c = A \oplus B = \{x/A \cap B_x \neq \emptyset\}$$

L'exemple suivant permet de montrer un exemple de dilatation.

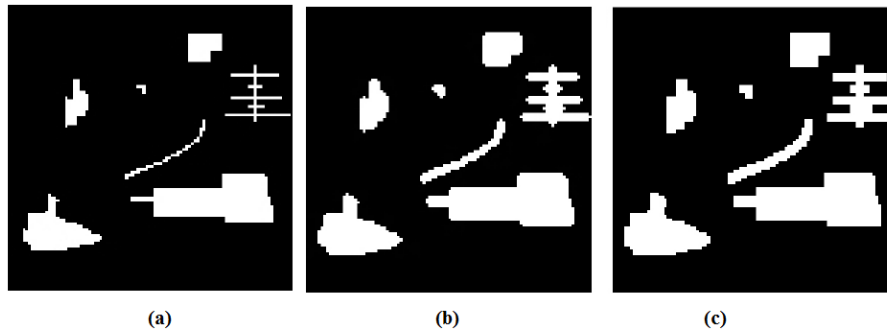


Figure 6.9. Image originale en (a), Images dilatées en 4-connextité (b) et en 8-connextité (c)

Erosion

L'érosion sert à rétrécir les objets dans une image. Lorsque l'élément structurant parcourt l'objet, l'ensemble des pixels non recouvert par l'élément structurant forment l'ensemble érodé.

Soient A et B deux ensembles, on définit l'érosion de A par B, notée $A \ominus B$, par :

$$c = A \ominus B = \{x/B_x \subset A\}$$

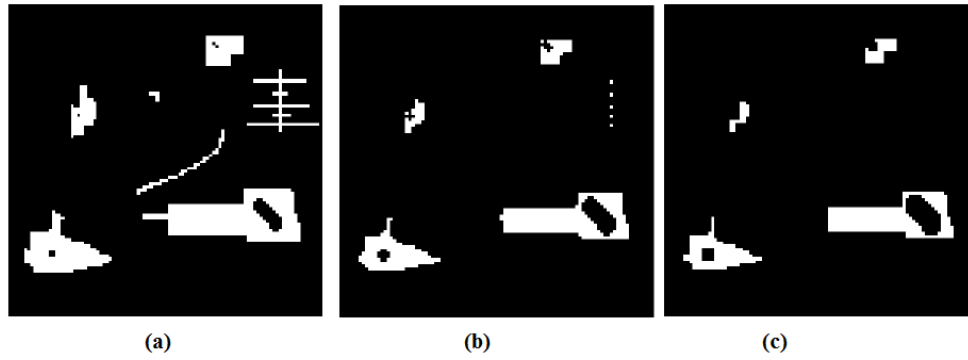


Figure 6.10. Image originale en (a), Images érodées en 4-connectivité (b) et en 8-connectivité (c)

Fermeture

C'est une dilatation suivie d'une érosion avec le même élément structurant.

La fermeture de l'ensemble A par l'élément structurant B, notée $A \bullet B$ est définie par :

$$c = A \bullet B = (A \oplus B) \ominus B$$

La fermeture permet le recouvrement des objets proches et la suppression des trous.

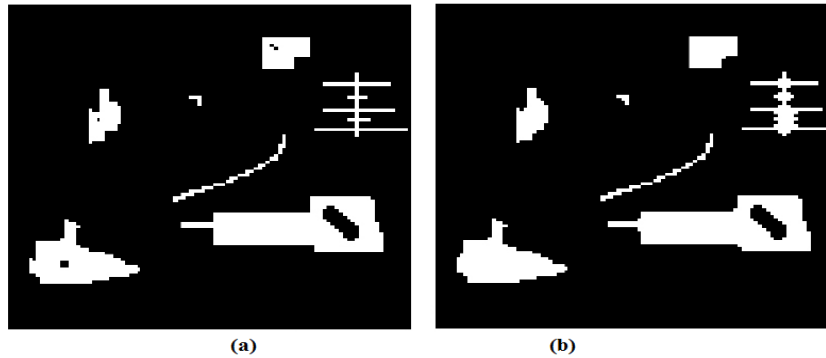


Figure 6.11. (a) Image originale ; (b) Résultat d'une fermeture

Ouverture

L'ouverture est une érosion suivie d'une dilatation avec le même élément structurant.

L'ouverture de l'ensemble A par l'élément structurant B, notée $A \circ B$ est définie par :

$$c = A \circ B = (A \ominus B) \oplus B$$

L'ouverture lisse les contours, supprime les petits îlots.

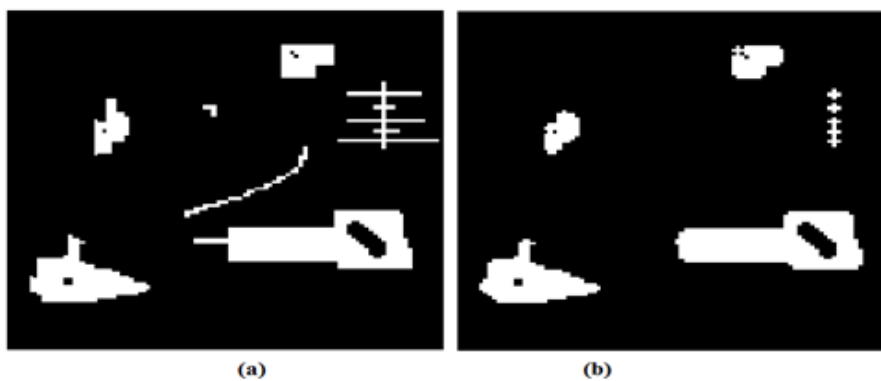


Figure 6.12. (a) Image originale ; (b) Résultat d'une ouverture

6.5. Extraction de paramètres et classification d'objets

6.4.1 Extraction des paramètres

L'extraction de paramètres caractéristiques -en anglais, « feature extraction »- est une étape de grande importance. Si elle est mal conçue, il sera difficile, voire impossible, d'effectuer une classification efficace.

L'extraction des caractéristiques consiste à l'obtention de l'information la plus pertinente d'une forme pour la présenter au système de reconnaissance. Les caractéristiques ou primitives d'une forme constituent une représentation réduite de la forme d'où une perte irréversible d'information s'impose.

Nous disposons d'une image binaire contenant des objets ou régions avec des contours ou frontières parfaitement définis (localisés, lissés, fermés), l'objectif est d'extraire des paramètres de forme afin de caractériser chacun des objets ou région. Parmi les paramètres caractéristiques, on cite les moments géométriques et les moments de Zernike.

Les moments géométriques

La méthode des moments géométriques, comme toutes les méthodes statistiques, permet d'extraire des paramètres propres à la forme. Hu a élaboré un théorème énonçant qu'un ensemble infini de moments calculés à partir d'une image permet de la décrire uniquement et complètement. Il a aussi créé une version de ces moments invariants en translation, en rotation et en homothétie.

Les moments d'ordre $p + q$ d'une fonction $f(x, y)$ sont définis par :

$$m_{pq} = \iint_{-\infty}^{+\infty} x^p y^q f(x, y) dx dy$$

où p et q sont l'ordre du moment par rapport à x et y respectivement. Dans le cas d'une image binaire où :

$$f(x, y) = \begin{cases} 1 & (x, y) \in \text{objet} \\ 0 & \text{sinon} \end{cases}$$

L'équation du moment devient :

$$m_{pq} = \sum_x \sum_y x^p y^q$$

Pour rendre les moments invariants par rapport à la translation, l'image doit être centrée par rapport à son centre de gravité. Le moment central résultant aura la forme :

$$\mu_{pq} = \sum_x \sum_y (x - x_c)^p (y - y_c)^q$$

où x_c et y_c sont les coordonnées du centre de gravité de l'image de l'objet données par :

$$x_c = \frac{m_{10}}{m_{00}}, \quad y_c = \frac{m_{01}}{m_{00}}$$

$$\mu_{00} = m_{00}$$

$$\mu_{10} = \mu_{01} = 0$$

$$\mu_{11} = m_{11} - (m_{00}x_c y_c)$$

$$\mu_{20} = m_{20} - (m_{00}x_c^2)$$

$$\mu_{02} = m_{02} - (m_{00}y_c^2)$$

$$\mu_{21} = m_{21} - (m_{20}y_c) - (2m_{11}x_c) - (2m_{00}x_c^2 y_c)$$

$$\mu_{12} = m_{12} - (m_{02}x_c) - (2m_{11}y_c) - (2m_{00}y_c^2 x)$$

$$\mu_{30} = m_{30} - (3m_{20}x_c) + (2m_{00}x_c^3)$$

$$\mu_{03} = m_{03} - (3m_{20}y_c) + (2m_{00}y_c^3)$$

Les moment centrés et normalisés η_{pq} sont obtenus en divisant chaque moment centré μ_{pq} par γ :

$$n_{pq} = \frac{\mu_{pq}}{\mu_{00}^\gamma}, \quad \gamma = \frac{p+q}{2} + 1$$

Ces moments sont caractérisés par une invraiance par rapport à la translation et le changement d'echelle.

$$\emptyset_1 = \eta_{20} + \eta_{02}$$

$$\emptyset_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$

$$\emptyset_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$$

$$\emptyset_4 = (\eta_{20} - \eta_{02})^2 + (\eta_{21} - \eta_{03})^2$$

$$\emptyset_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} - \eta_{03})^2] + (3\eta_{21} - \eta_{03}) + (3\eta_{30} - 3\eta_{12}) + (\eta_{21} - \eta_{03})(3(\eta_{30} + \eta_{12})^2 - (\eta_{21} - \eta_{03})^2)$$

$$\emptyset_6 = (\eta_{20} - \eta_{02})(\eta_{20} - \eta_{02})^2 - (3\eta_{21} - \eta_{03})^2 + \eta_{11}(\eta_{03} - \eta_{12})(\eta_{21} - \eta_{03})$$

$$\emptyset_7 = (\eta_{21} - \eta_{03})(\eta_{30} - 3\eta_{12})[(\eta_{30} + \eta_{12})^2 - (3\eta_{21} - \eta_{03})^2] - (\eta_{30} + 3\eta_{12})(\eta_{21} - \eta_{03})(3\eta_{21} - \eta_{03})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} - \eta_{03})^2]$$

Le calcul de ces momoents se fait comme suit, après la segmentation de l'image ,on calcule ces moments m_{pq} , puis les moments centraux μ_{pq} et enfin les moments invariants \emptyset_k .

Les moments de Zernike

Les moments de Zernike sont basés sur le principe des polynômes orthogonaux. De ce fait, la reconstruction (en général approchée) de la forme à partir de ces moments est possible. Un ensemble relativement petit de moments de Zernike peut caractériser la forme globale d'un objet, Les moments d'ordre bas représentent la forme globale d'un objet et celles d'ordre plus

supérieur le détail de cet objet. Plus on fait appel à des moments de Zernike d'ordre plus élevé, plus est la précision de reconstruction de l'image de l'objet. Les moments de Zernike sont qualifiés par leur invariance par translation, par changement d'échelle et par rotation.

Les polynômes de Zernike notés par $V_{nm}(x, y)$ sont donnés par :

$$V_{nm}(x, y) = V_{nm}(\rho, \theta) = R_{nm}(\rho)e^{jm\theta}$$

où n entier positif ou nul, m entier positif ou négatif selon n ; ρ longueur du vecteur de l'origine au pixel (x, y) et θ l'angle entre le vecteur ρ et l'axe des abscisses dans le sens des aiguilles d'une montre

R_{nm} est un polynôme radiale donné par :

$$R_{nm}(\rho) = \sum_{s=0}^{n-|m|/2} (-1)^s \frac{(n-s)!}{s! \left(\frac{n+|m|}{2} - s\right)! \left(\frac{n-|m|}{2} - s\right)!} \rho^{n-2s}$$

Les moments de Zernike sont les projections de l'image sur la base des fonctions orthogonales $V_{nm}(\rho, \theta)$

Les moments de Zernike d'ordre n avec une répétition m pour une fonction image $f(x, y)$ qui s'annule en dehors du cercle unitaire ont pour expression :

$$A_{nm} = \frac{n+1}{\pi} \iint_{x^2+y^2 \leq 1} f(x, y) V_{nm}^*(\rho, \theta) dx dy$$

Pour une image numérique :

$$A_{nm} = \frac{n+1}{\pi} \sum_x \sum_y f(x, y) V_{nm}^*(\rho, \theta)$$

Pour calculer les moments de Zernike d'une image donnée, le centre de l'image est pris comme origine

Pour ramener l'image à l'intérieur du cercle unitaire, on réalise une normalisation des rayons de tous les pixels de l'image : Partant du centre de gravité, on balaye tous les pixels de l'image pour chercher le plus grand rayon, celui-ci est considéré comme unitaire.

Tous les autres rayons sont divisés par ce rayon maximum pour inclure les pixels correspondants dans le cercle unité.

Remarques

Les paramètres caractéristiques devront être :

Discriminatifs : les paramètres caractéristiques doivent être significativement différents pour des formes appartenant à des classes différentes.

Fiables : pour des formes d'une même classe, les paramètres caractéristiques doivent être similaires.

De nombre réduit : le coût, la complexité et les exigences en temps de calcul d'un système de reconnaissance de formes croissent avec le nombre de paramètres utilisés.

6.4.2 Méthodes de classification

L'objectif des méthodes de classification s'est de classer des individus en exploitant leur similarité mesurée par les paramètres caractéristiques. Les individus peuvent être :

Les pixels : les attributs sont ponctuels ou calculés dans un voisinage centré sur le pixel.

Les sous-images : l'image est découpée en fenêtres. Chaque fenêtre est classée en utilisant les attributs calculés sur la totalité de la fenêtre

Les régions : obtenues par une méthode de segmentation.

La classification sert à interpréter les formes issues de la segmentation (reconnaissance de formes).

A chaque individu, on associe un vecteur de paramètres caractéristiques. Ce vecteur représente l'individu dans l'espace des attributs.

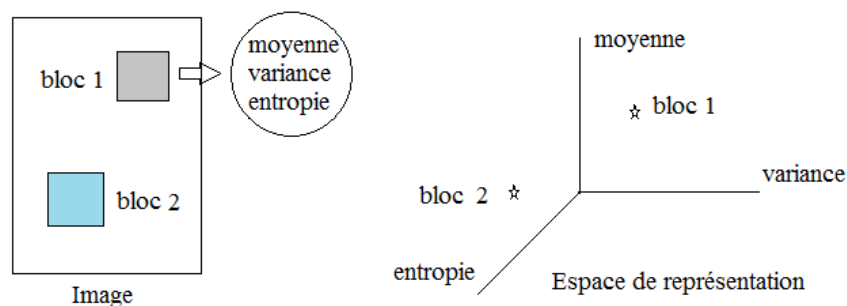


Figure 6.13. Espaces de représentation des individus

Pour la classification, il existe deux approches :

a- Classification non supervisée

C'est le partitionnement de l'ensemble d'individus en fonction des critères de proximité des vecteurs d'attributs dans l'espace de représentation. Cette méthode ne demande pas des connaissances à priori sur l'image.

b- Classification supervisée

Dans ce cas la classe d'un certain nombre d'individu est connu apriori, cette connaissance permet de construire un partitionnement de l'espace de représentation.

Les individus de cette classe inconnue sont classés en fonction de leur position dans l'espace de représentation grâce à ce découpage. Dans ce cas il est nécessaire d'avoir un ensemble d'individu dont la classe d'appartenance est connue, c'est l'ensemble d'apprentissage. Parmi les méthodes de la classification supervisée on cite la méthode des K plus proches voisins (kppv).

Bibliographie

- [1].G.Blanchet ,M.Charbit , Signaux et images sous Matlab. HEES Science Europe Ltd, Paris , 2001.
- [2].G. Burel, Introduction au traitement d'images : simulation sous Matlab . Edition Paris HERMES, 2001.
- [3].JP .Cocquerez ,Analyse d'images : filtrage et segmentation – JP – Masson 1995
- [4].R. Horaud ,O. Monga. Vision par ordinateur : outils fondamentaux. Deuxième édition. Editions Hermès
- [5].R.Gonzalez , R.Woods et S. Eddins. Digital Image Processing Using Matlab ,Gatesmark publishing ,2008