

*Université Kasdi Merbah – Ouargla*  
*Faculté des Nouvelles Technologies de l'Information et de la*  
*Télécommunication*  
*Département d'Electronique et de Télécommunication*



**MASTER ACADEMIQUE**  
**Domaine : SCIENCES ET TECHNOLOGIES**  
**Filière : ELECTRONIQUE**  
**Spécialité : AUTOMATIQUE**  
**Thème :**

---

## ***Détection et localisation des cercles***

---

**Présenté par :**

**GOUARAH Djmouai.**

**BOUAFFANE Salah-Eddine.**

**Devant le jury composé de :**

M.	SAMAI Djamel	MC (B)	Président	UKM Ouargla
M.	DJELLAL Adel	MA (A)	Encadreur/rapporteur	UKM Ouargla
M.	MERAOUMIA Abdallah	MC (B)	Examineur	UKM Ouargla
M.	SAIGAA Mohamed	MA (A)	Examineur	UKM Ouargla

***Année universitaire : 2015/2016***

# REMERCIEMENTS

*Tout d'abord nous remercions le bon Dieu tout puissant de la bonne santé, de la volonté et de la patience qu'il nous a accordé tout au long de nos études. Nous exprimons nos plus vifs remerciements à notre encadreur : **Mr. Adel Djellal** qui a orienté et assidûment dirigé notre travail. Nous présentons aussi nos sincères remerciements à tous les enseignants de département d'électronique et de télécommunications.*

*Nous remercions les membres du jury qui ont bien voulu accepter d'examiner ce travail.*

## Dédicace

*Après avoir remercié vivement Allah, le tout puissant de nous avoir donné l'opportunité de penser et d'écrire, nous dédions spécialement ce travail à tous nos parents.*

- *A nos frères et sœurs*
- *A nos ami(e)s et collègues de toute promotion*

*Également, nous vous dédions ce travail.*

## Table des matières

Table des matières .....	1
Liste des Figures .....	1
Table des Tableaux.....	1
Introduction Générale.....	1
Chapitre 1. Caractéristiques et formats d'images .....	2
1.1. Introduction .....	2
1.2. Définition de l'image : .....	2
1.3. Codage : .....	2
1.4. Types d'images : .....	2
1.4.1. Image binaire : .....	3
1.4.2. Image en niveau de gris.....	3
1.4.3. Image couleur (RGB) : .....	4
1.5. Résolution.....	6
1.6. Caractéristiques de l'image : .....	7
1.6.1. Pixel.....	7
1.6.2. Poids de l'image.....	7
1.6.3. Transparence : .....	7
1.6.4. Luminance : .....	7
1.6.5. Contraste : .....	7
1.6.6. Histogramme : .....	7
1.6.7. Homogénéité : .....	8
1.6.8. Connexité : .....	8
1.6.9. Région.....	9
1.7. Les différents formats d'images : .....	9
1.7.1. Format vectorielle : .....	9
1.7.2. Format matricielle : .....	10
1.7.3. Les avantages de l'image vectorisée : .....	10
1.8. Les techniques fondamentales en traitement d'images : .....	10
1.8.1. Acquisition de l'image : .....	11
1.8.2. Le rehaussement de l'image : .....	12
1.8.3. morphologie : .....	12
1.8.4. segmentation : .....	12
1.8.5. représentation et description : .....	13
1.8.6. reconnaissance : .....	13

1.8.7.	restauration d'image :	13
1.8.8.	Comprissions :	13
1.9.	Conclusion	14
<b>Chapitre 2. L'image RGB et le prétraitement d'image</b>		15
2.1.	Introduction	15
2.2.	Définition et résolution	15
2.2.1.	Courbe tonale	15
2.2.2.	Changer la luminosité :	15
2.2.3.	Augmenter le contraste	17
2.3.	Quelques traitements classiques	17
2.3.1.	Négatif	17
2.3.2.	Rouge	18
2.3.3.	Niveaux de gris	18
2.3.4.	Seuillage	19
2.3.5.	Sépia	19
2.3.6.	Lissage	19
2.4.	Prétraitement des images	20
2.4.1.	Lissage local	20
2.4.2.	Amélioration d'images	22
2.5.	Conclusion	24
<b>Chapitre 3. La détection des cercles et des ellipses dans les images.</b>		25
3.1.	Introduction	25
3.2.	Cercle Hough Transformation de Hough d'un cercle (CHT) ..	25
3.2.1.	Limites	26
3.2.2.	Avantage	26
3.2.3.	Inconvénients	26
3.2.4.	Représentation géométrique d'un cercle et une ellipse :	26
3.3.	Les ellipses	28
3.4.	Les différentes techniques de détection de contour :	29
3.4.1.	Filtre de Prewitt :	29
3.4.2.	Filtre de Sobel :	29
3.4.3.	Filtre de Canny :	29
3.4.4.	Filtre de robert :	30
3.5.	Résultat :	30
3.5.1.	Explication du la première méthode :	30
3.5.2.	Explication de deuxième méthode	30

<b>3.5.3. Comparaison les résultats de détection du cercle selon le paramètre choisis .....</b>	<b>33</b>
<b>3.6. Discussion .....</b>	<b>36</b>
<b>3.7. Conclusion.....</b>	<b>36</b>
<b>Conclusion Générale .....</b>	<b>37</b>
<b>Bibliographie.....</b>	<b>38</b>

## Liste des Figures

Figure 1. Pixel de l'image de coordonnées (i, j).....	2
Figure 2. Image en noir et blanc.....	3
Figure 3. Image en niveaux de gris.....	4
Figure 4. Image en couleurs 8 bits et la palette de couleurs 8 bits.....	4
Figure 5. Palette 24 bits Exemple de couleur en 24 bits.....	5
Figure 6. illustre la façon dont la teinte, la saturation et la valeur sont définis.....	5
Figure 7. Image en couleurs 24 bits.....	6
Figure 8. Image en niveau de gris et son histogramme.....	8
Figure 9. Voisinages.....	9
Figure 10. Correction géométrique.....	9
Figure 11. Traitements fondamentaux en traitement d'images.....	11
Figure 12. Chaîne d'acquisition d'image numérique.....	12
Figure 13. Schéma de principe d'un numériseur d'image optique.....	12
Figure 14. courbe tonale.....	15
Figure 15. Augmentation de la luminosité.....	16
Figure 16. rejoinde les axes tangentiellment.....	16
Figure 17. Diminuer la luminosité.....	17
Figure 18. Augmentation du contraste.....	17
Figure 19. le négatif d'un image.....	18
Figure 20. Le niveau du rouge d'une image.....	18
Figure 21. Le niveau de gris d'une image.....	18
Figure 22. Le seuillage d'une image.....	19
Figure 23. Sépia d'une image.....	19
Figure 24. Lissage d'une image.....	20
Figure 25. L'analyse d'images est une chaîne de traitement de l'information.....	20
Figure 26. voisinage du pixel x ,y.....	21
Figure 27. Cercle géométrie.....	26
Figure 28. Ellipse géométrie.....	27
Figure 29. Deuxième géométrie de l'ellipse.....	28
Figure 30. Pièces de 100 dinars.....	31
Figure 31. Pinions.....	31
Figure 32. Cannelles.....	32
Figure 33. Piston.....	32
Figure 34. Joint de culasse.....	32
Figure 35. Roulement.....	33
Figure 36. hough_test2.....	33
Figure 37. Quatre différents arrière plants.....	34
Figure 38. Exemple d'utilisation des techniques de détection de contours.....	34

## Table des Tableaux

Tableau 1. Tableau du codage d'une image en couleurs 24 bits .....	5
Tableau 2. Comparaison entre les deux méthodes.....	31
Tableau 3. Comparaison entre les quatre filtres.....	33
Tableau 4. Comparaison entre les quatre épaisseurs. ....	35
Tableau 5. Comparaison entre les 5 pas d'angle utilisés .....	35

## Liste des Abréviations

RGB : red green blue

HSV : Hue Saturation valeur

RVB : rouge vert bleu

CHT : Cercle Hough Transformation

## Introduction Générale

La vision par ordinateur représente une partie très dynamique dans l'informatique et la science cognitive de la machine. Suite à une explosion d'intérêts durant les années 1970s, les années 1980s et 1990s ont reconnu une maturation du domaine et l'apparition de plusieurs domaines d'applications : contrôle de qualité, diagnostic technique, guidance des véhicules, imagerie médicale, et vidéo surveillance. Ce progrès peut être observé dans le nombre incrémental des logiciels et matériels dans le marché, ainsi que le nombre colossal des cours sur le traitement numérique de l'image et la vision de la machine qui sont présentés sur un nombre infini des universités à travers le monde.

La reconnaissance de formes offre un outil très intéressant et d'actualité pour les domaines de la robotique mobile et de bras manipulateurs, ainsi que le contrôle de qualité et la détection de pièces sur la chaîne de production. Ces domaines offrent l'avantage que les formes à détecter sont prédéfinies (cercles, carrés, ellipses, etc.), mais souffrent d'une contrainte indispensable : vu que la reconnaissance de formes se fait en temps réel, le traitement d'image doit être fait en minimisant le temps de calculs pour réduire les délais. Pour le cas de la robotique mobile.

Ce travail sur la reconnaissance d'une forme prédéfini sur un fond aléatoire, ce qui est le cas général de la reconnaissance de forme nécessaire pour les robots pour la localisation ainsi que la détection de pièces sur la chaîne de production industriel. Ce travail consiste à utiliser une technique de reconnaissance de formes générale qui peut être utilisée sur des formes prédéfinies, il suffit d'avoir une fonction en  $x$  et  $y$  du contour de la forme à détecter. Cette technique a été présentée par Paul Hough en 1962, et a été breveté par IBM. Elle consiste à parcourir une image pour trouver la probabilité d'existence de la forme à détecter sur chaque pixel de cette image. Ce principe pose le problème qu'il nécessite un temps important pour traiter chaque image, ce qui pose un grand problème pour l'utilisation de cette technique en temps réel.

Ce travail peut être décomposé comme suit :

- Un premier chapitre donnant une idée générale sur l'imagerie numérique.
- Le second chapitre explique les traitements nécessaires sur l'image numérique représentée en format RGB.
- Le dernier chapitre consiste à présenter les étapes nécessaires à l'application de la transformation de Hough ainsi que les résultats d'applications.



## Chapitre 1. Caractéristiques et formats d'images

### 1.1. Introduction

Dans ce chapitre, nous allons présenter les différentes phases de la formation d'image ainsi que sa définition, son interprétation par la machine, ces différents types, ces caractéristiques, de ces formats.

### 1.2. Définition de l'image :

L'image est une représentation d'une personne ou d'un objet par la peinture, le dessin, la photographie, le film, etc. C'est aussi un ensemble structuré d'informations, qui après affichage sur l'écran, ont une signification pour l'œil humain [1].

Elle peut être décrite sous la forme d'une fonction  $I(x, y)$  de brillance analogique continue, définie dans un domaine borné, tel que  $x$  et  $y$  sont les coordonnées spatiales d'un point de l'image et  $I$  est une fonction d'intensité lumineuse et de couleur. Sous cet aspect, l'image est inexploitable par la machine, ce qui nécessite sa numérisation [2].

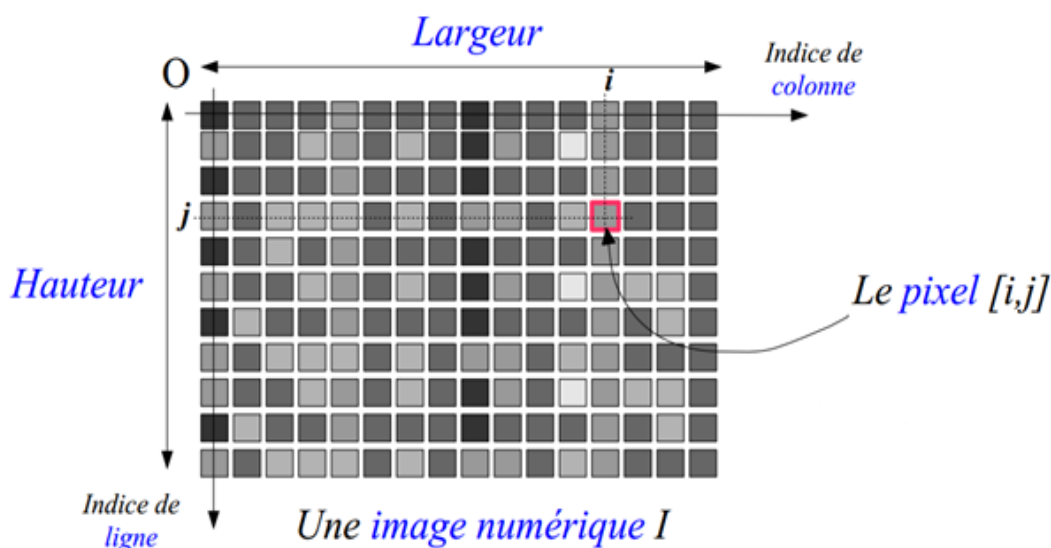


Figure 1. Pixel de l'image de coordonnées  $(i, j)$

L'image numérique, désignée aussi par le terme scène, possède un repère comme indiqué en figure 1, il est différent de celui d'une fonction mathématique. Elle a une hauteur ( $H$ ) et une largeur ( $W$ ).

### 1.3. Codage :

L'information est codée en binaire. Le support évolue mais le principe est toujours le même un même élément peut se trouver dans 2 états différents stables. Il constitue une mémoire élémentaire ou bit [3].

### 1.4. Types d'images :

En traitement d'image nous aurons à faire à trois types d'images [4] :

### 1.4.1. Image binaire :

Chaque pixel est soit blanc soit noir. Puisqu'il y a uniquement deux valeurs pour chaque pixel, une seule bite est utilisée pour le coder.

#### Codage d'une image en noir et blanc

Pour ce type de codage, chaque pixel est soit noir, soit blanc. Il faut un bit pour coder un pixel (0 pour noir, 1 pour blanc). L'image de 10000 pixels codée occupe donc 10000 bits en mémoire.



Figure 2. Image en noir et blanc

Ce type de codage peut convenir pour un plan ou un texte mais on voit ses limites lorsqu'il s'agit d'une photographie.

### 1.4.2. Image en niveau de gris

Chaque pixel est un niveau de gris, allant de 0 (noir) à 255 (blanc). Cet intervalle de valeur signifie que chaque pixel est codé sur huit bits (un octet). 256 niveaux de gris suffisent pour représenter les différents pixels

#### Codage d'une image en niveaux de gris

En général on code chaque pixel sur 8 bits = 1 octet. On a alors 256 possibilités (on dit 256 niveaux de gris). L'image de 10 000 pixels codée occupe alors 10 000 octets en mémoire.



Figure 3. Image en niveaux de gris

### 1.4.3. Image couleur (RGB) :

Chaque pixel possède une couleur décrite par la quantité de rouge (R), vert (G) et bleu (B). Chacune de ces trois composantes est codée sur l'intervalle  $[0, 255]$ , ce qui donne  $256^3 = 16\,777\,216$  couleurs possibles. Il faut 24 bits pour coder un pixel.

#### ➤ Codage d'une image en couleurs 8 bits

Dans ce cas on attache une palette de 256 couleurs à l'image. Ces 256 couleurs sont choisies parmi les 16 millions de couleurs de la palette RVB. Pour chaque image le programme recherche les 256 couleurs les plus pertinentes. Chaque code (de 0 à 255) désigne une couleur. L'image occupe 3 fois moins de place en mémoire qu'avec un codage 24 bits. L'image est moins nuancée : sa qualité est bonne mais moindre.

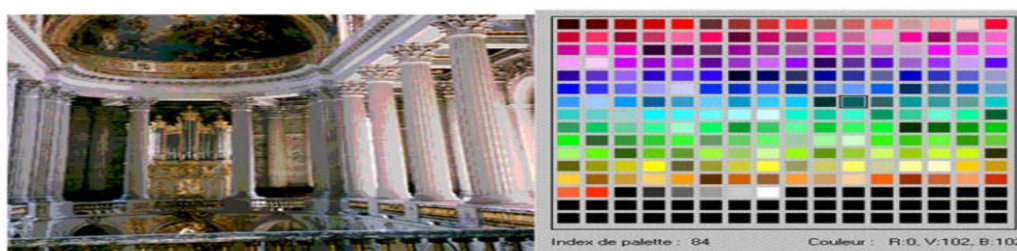


Figure 4. Image en couleurs 8 bits et la palette de couleurs 8 bits

Image couleur 8 bits et sa palette. Remarquer la couleur 84 et sa correspondance en RVB.

#### ➤ Codage d'une image en couleurs 24 bits

Il existe plusieurs modes de codage de la couleur. Le plus utilisé est le codage Rouge, Vert, Bleu (RVB). Chaque couleur est codée sur 1 octet = 8 bits. Chaque pixel sur 3 octets c'est à dire 24 bits : le rouge de 0 à 255, le vert de 0 à 255, le Bleu de 0 à 255. Le principe repose sur la synthèse additive des couleurs : on peut obtenir une couleur quelconque par addition de ces 3 couleurs primaires en proportions convenables. On obtient ainsi  $256 \times 256 \times 256 = 16\,777\,216$  (plus de 16 millions de couleurs différentes).

Tableau 1. Tableau du codage d'une image en couleurs 24 bits

couleur	noir	Bleu pale	Vert pale	rose	bleu	vert	rouge	blanc
R	0	120	120	150	120	120	255	255
V	0	120	150	120	120	255	120	255
B	0	150	120	120	255	120	120	255

L'image de 10000 pixels ainsi codée occupe  $10000 \times 3 = 30000$  octets.

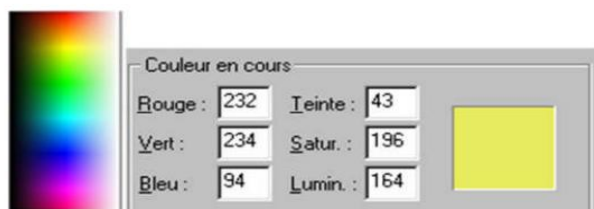


Figure 5. Palette 24 bits Exemple de couleur en 24 bits

### ➤ Conversion de RGB à HSV :

La vision des couleurs peut être traitée en utilisant l'espace de couleur RVB ou l'espace de couleur HSV. L'espace de couleur RVB décrit les couleurs en fonction de la quantité de rouge, vert et bleu présente. L'espace de couleur HSV décrit les couleurs en fonction de la teinte, la saturation et la valeur. Dans les situations où la description de la couleur joue un rôle essentiel, le modèle de couleur HSV est souvent préféré par rapport au modèle RVB.

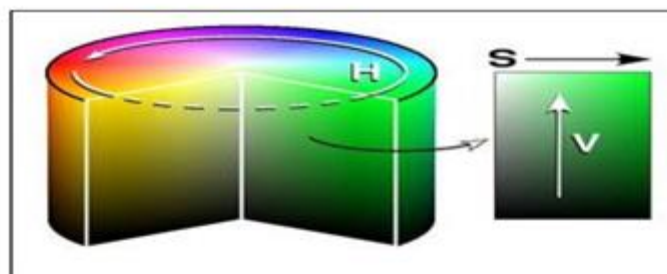


Figure 6. illustre la façon dont la teinte, la saturation et la valeur sont définis.

- Hue représente le type de couleur. Elle peut être décrite en termes d'un angle au-dessus de cercle. Bien qu'un cercle contienne 360 degrés de rotation, la valeur de teinte est normalisée à une plage de 0 à 255, 0 étant rouge.
- Saturation représente la vitalité de la couleur. Sa valeur varie de 0 à 255.
- La valeur représente la luminosité de la couleur. Elle varie de 0 à 255, 0 étant complètement sombre et 255 étant entièrement lumineux.
- Blanc a une valeur HSV de 0-255, 0-255, 255. Noir a une valeur HSV de 0-255, 0-255,0. La teinte et niveau de saturation ne fait pas de différence lorsque la valeur est au max ou min niveau d'intensité. La caméra couleur, sur le robot, utilise le modèle RVB pour déterminer la couleur. Une fois que l'appareil à lire ces valeurs, elles sont converties en valeurs HSV.

Pour déterminer l'emplacement d'un objet, couleur spécifique pour laquelle le robot est à la recherche. Les pixels sont contrôlés individuellement pour déterminer si elles correspondent à un seuil de couleur prédéterminée.

La technique de Conversion RVB à HSV :

Étant donné trois R, G et B (chacun entre 0 et 255), vous pouvez d'abord définir m et M avec les relations

$$M = \max \{R, G, B\}.$$

$$m = \min \{R, G, B\}.$$

Et puis V et S sont définis par les équations :

$$V = M / 255$$

$$\begin{cases} S = \frac{1-m}{M} & , \text{ si } M > 0 \\ S = 0 & , \text{ si } M = 0 \end{cases}$$

la teinte H est définie par les équations

$$\begin{cases} H = \cos^{-1}[(R - \frac{1}{2}G - \frac{1}{2}B) / \sqrt{R^2 + G^2 + B^2 - RG - RB - GB}] & , \text{ si } G \geq B \\ H = 360 - \cos^{-1} [(R - \frac{1}{2}G - \frac{1}{2}B) / \sqrt{R^2 + G^2 + B^2 - RG - RB - GB}] & , \text{ si } B > G \end{cases}$$

Cosinus inverse est calculé en degrés.



Figure 7. Image en couleurs 24 bits

### 1.5. Résolution

Une image est divisée en points ou pixels. Considérons une image de 10 cm sur 10 cm avec une résolution très faible de 10 pixels par cm. Elle est codée sur  $100 \times 100 = 10000$  pixels. Avec une résolution convenable de 100 pixels par cm, elle serait codée sur  $1000 \times 1000 = 1000000$  pixels = 1 M pixels. Le symbole M signifiant million.

## **1.6. Caractéristiques de l'image :**

### **1.6.1. Pixel**

Une image est constituée d'un ensemble de points appelés pixels. Le pixel (Picture élément) représente ainsi le plus petit élément constitutif d'une image numérique. La quantité d'information que véhicule chaque pixel donne des nuances entre images monochromes et images couleurs. Pour les images 3D le «pixel» est alors appelé un voxel, et représente un volume élémentaire. Des exemples d'images de ce type se rencontrent dans les images médicales. Les images tomographiques axiales sont ainsi des images construites à partir de plusieurs radiographies faites sous des angles de vue différents [3].

### **1.6.2. Poids de l'image**

C'est la taille de l'image. Etant donné que cette dernière est représentée sous forme d'une matrice dont les valeurs représentent l'intensité (pixels), le nombre de colonne (n) multiplié par le nombre de ligne (m) donne le nombre total de pixels dans l'image. Pour une image de 640x480 en couleur : Nombre de pixel =  $640 \times 480 = 307200$ , Poids de chaque pixel = 3 octets  
Le poids de l'image =  $307200 \times 3 = 921600$  octets = 900 Ko.

### **1.6.3. Transparence :**

La transparence est une caractéristique définissant le niveau d'opacité des éléments de l'image, c'est la possibilité de voir à travers l'image des éléments graphiques situés derrière celle-ci [1].

### **1.6.4. Luminance :**

C'est le degré de luminosité des points de l'image. Elle est définie ainsi comme étant le quotient de l'intensité lumineuse d'une surface par l'aire apparente de cette surface [1].

### **1.6.5. Contraste :**

Est une propriété intrinsèque d'une image qui désigne et quantifie la différence entre les parties claires et foncées d'une image (elle différencie les couleurs claires des couleurs foncées). En photographie on le définit le contraste comme la différence entre la densité la plus forte et la plus faible d'une image. Le contrôle du contraste est un élément important de la pratique photographique.

### **1.6.6. Histogramme :**

Un histogramme est un graphique statistique permettant de représenter la distribution des intensités des pixels d'une image. Il fournit diverses informations comme les statistiques d'ordre (moyenne, variance, ...), l'entropie, et peut permettre d'isoler des objets [1].

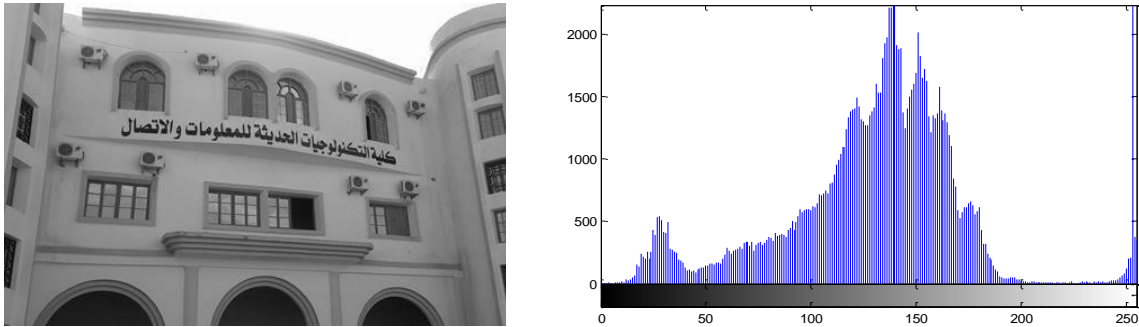


Figure 8. Image en niveau de gris et son histogramme

L'histogramme de la figure 8 noté ici  $H(x)$ , est le nombre de pixels dont le niveau de gris est égal à  $x$ . l'histogramme cumulatif normalisé est calculé comme suit :

$$HC(x) = \sum_{i=0}^x \frac{H(x)}{n*m}$$

$HC(x)$  est le taux de pixels dont le niveau de gris est inférieur à  $x$ . Nous présentons dans la suite quelques traitements d'analyse effectués uniquement à partir de l'histogramme.

- (1) Normalisation : exploiter toute la dynamique de codage.
- (2) Egalisation : équilibrer la dynamique de codage et augmenter le contraste.
- (3) Segmentation : simplifier l'image en regroupant les pixels selon leurs valeurs.

### 1.6.7. Homogénéité :

L'homogénéité est une information locale et correspond au caractère uniforme d'une région. Une région dans une image est dite homogène si elle regroupe un ensemble de pixels qui possèdent des caractéristiques similaires ou uniformes. Ces caractéristiques peuvent être par exemple la variance du niveau de gris, la couleur, ...

### 1.6.8. Connexité :

En traitant une image, on est souvent amené à se déplacer dans celle-ci. Un déplacement doit souvent obéir à des règles de voisinage, on utilise généralement deux types de voisinage : le voisinage à 4-connexité (4 pixels voisins, figure 9 (a)) et le voisinage à 8-connexité (8 pixels voisins, figure 9 (b)).



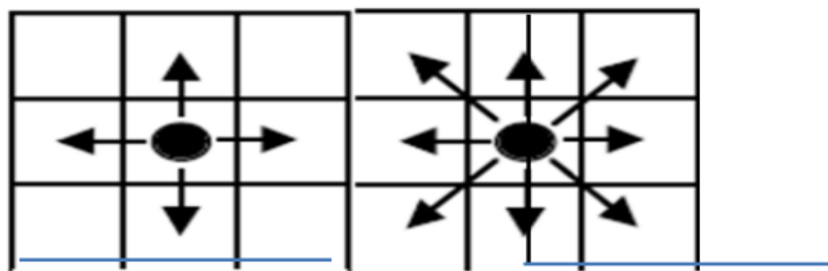


Figure 9. Voisinages

### ➤ Opérateurs de voisinage avec ré-échantillonnage spatial :

Les dimensions des images résultat et de départ peuvent différer. Ceci survient dans le cas d'un ré-échantillonnage spatial de l'image de départ destiné à changer de repère spatial. Ceci est réalisé avec des transformations géométriques (*e.g.*, enchaînement de translations, rotations et homothéties).

Ainsi, la Figure 10 illustre la transformation d'une image de 4 lignes x 4 colonnes en une image de 7 colonnes x 5 lignes. Le compte numérique de tout pixel de l'image résultat résulte d'une convolution de l'image de départ par un opérateur (*e.g.*, méthode du plus proche voisin, bilinéaire, cubique, etc.).

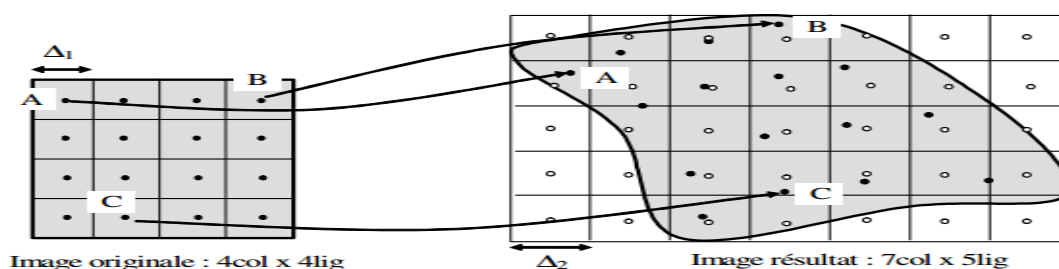


Figure 10. Correction géométrique.

L'image originale est transformée en une nouvelle image (pixels  $\Delta_2$  de taille  $\Delta_2$ ) de plus grande dimension. La transformation géométrique a été définie à partir de points A, B et C qui ont été repérés dans les repères de départ et d'arrivée. Ces points sont appelés "points de contrôle".

### 1.6.9. Région

Une région est un ensemble de pixels connexes et homogènes. Un pixel n'appartient à une région donnée que s'il vérifie les caractéristiques de celle-ci (intensité moyenne, centre de gravité,...). Une région est toujours limitée par un contour.

### 1.7. Les différents formats d'images :

On peut classer les images en deux formats :

#### 1.7.1. Format vectorielle :

Dans une image vectorielle les données sont représentées par des formes géométriques simples qui sont décrites d'un point de vue mathématique. Par exemple, un cercle est décrit



par une information du type (cercle, position du centre, rayon). Ces images sont essentiellement utilisées pour réaliser des schémas ou des plans [4].

Voici les formats usuels :

- Le format **svg** est un format ouvert d'image vectorielle.
- Le format **odg** est le format ouvert utilisé dans la suite bureautique libre d'OpenOffice.org.
- Les formats **wmf** et **emf** sont des formats de Microsoft supportant le dessin vectoriel.

### 1.7.2. Format matricielle :

Une image matricielle est formée d'un tableau de points ou pixels. Plus la densité des points sont élevée, plus le nombre d'informations est grand et plus la résolution de l'image est élevée. Corrélativement la place occupée en mémoire et la durée de traitement seront d'autant plus grandes. Les images vues sur un écran de télévision ou une photographie sont des images matricielles. On obtient également des images matricielles à l'aide d'un appareil photo numérique, d'une caméra vidéo numérique ou d'un scanner. Parmi ces formats on peut citer [4] :

**BMP (BitMap) :** Le format BMP est le format par défaut du logiciel Windows. C'est un format matriciel. Les images ne sont pas compressées. Son logiciel d'origine.

**GIF (Graphical Interchange Format) :** Le format GIF est un format qui a ouvert la voie à l'image sur le World Wide Web. C'est un format de compression qui n'accepte que les images en couleurs indexés codé sur 8 bits, C'est un format qui perd beaucoup de son marché suite à une bataille juridique concernant les droits d'utilisation sur Internet.

**JPEG (Joint Photographique Experts Group) :** Les images JPEG sont des images de 24 bits. C'est-à dire qu'elles peuvent afficher un spectre de 16 millions de couleurs. C'est la meilleure qualité d'images disponible.

**PPM (Potable PixMap):** Le format PPM concerne les images couleurs. Chaque pixel a pour valeur un triple (R, G, B) composé d'une composante rouge, verte et bleue.

### La différence entre une image matricielle et vectorielle

La principale différence entre ces deux formats est qu'une image vectorielle peut être agrandie sans perdre sa qualité alors qu'une image matricielle perd en netteté à l'agrandissement.

### 1.7.3. Les avantages de l'image vectorisée :

L'avantage de la vectorisation est de pouvoir agrandir ou réduire une image à volonté sans qu'elle ne perde en qualité. Les lignes vectorielles qui composent l'image étant créées par des formules mathématiques, ces dernières sont recalculées et réadaptées à chaque changement de taille. Cette technique permet de garantir à 100 % la qualité de l'image.

## 1.8. Les techniques fondamentales en traitement d'images :

Une panoplie de traitements peut être appliquée à l'image numérique, la figure suivante résume l'ensemble de ces traitements.

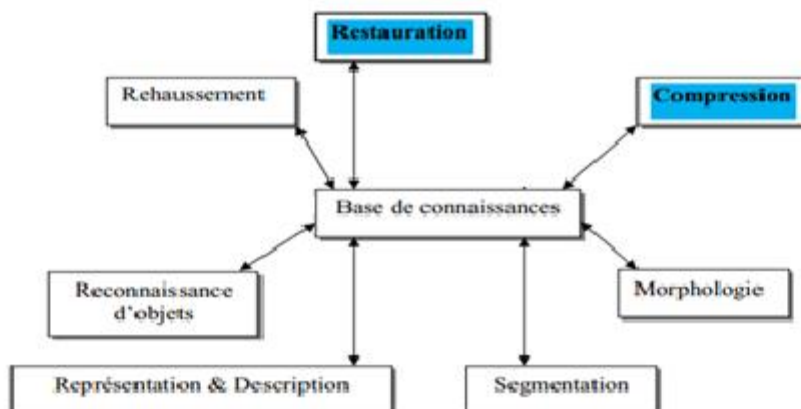


Figure 11. Traitements fondamentaux en traitement d'images

### 1.8.1. Acquisition de l'image :

L'acquisition d'images est une mesure spatiale d'une interaction entre une onde et de la matière. L'onde est émise par une source et reçue par un capteur. La matière occupe de l'espace et possède une masse. Elle a pour objet de passer de la scène physique à une forme numérique observée. Divers types de capteurs permettent d'acquérir des images raster numériques. Les plus usuels sont :

- les appareils photo et caméras numériques monocanal et multi-canal. L'acquisition d'images dans les domaines spectraux du bleu, du vert et du rouge permet de reconstituer la vision humaine.
- les scanners médicaux. Ils donnent des images 3-D sous la forme de séries d'images 2-D.
- les scanners et micro densitomètres. Ils numérisent des images ou négatifs sous forme analogique.
- les microscopes. En modifiant leur focale, ils permettent d'observer la nature 3-D des objets étudiés.
- les capteurs de télédétection. Systèmes rigides (*e.g.* barrette de 6000 CCD pour SPOT) ou à balayage (*e.g.* Thématique Mapper), avec un nombre très variable de bandes spectrales. Dans le domaine optique, de l'ultraviolet à l'infrarouge, l'acquisition d'image numérique (Figure 12, Figure 13) revient à numériser une image analogique de la scène observée. Quelle que soit la technologie (CCD, scanner, appareil photo numérique, etc.), elle requiert 3 étapes :

i) création de l'image optique de la scène dans le plan image du système optique qui agit comme un filtre passe bas de fréquence de coupure  $f_c$  (pas de fréquence  $f > f_c$  dans l'image optique).

ii) saisie de l'image optique (2 opérations). (1) Un spot / ouverture de taille  $a$  balaye l'image optique. (2) Durant un temps  $T$ , un élément photo sensible (photo diode / CCD ponctuel, linéaire ou surfacique, souvent avec des registres de décalage CCD) capte l'énergie lumineuse et la transforme en signal électrique. Le spot induit un filtrage passe bas de fréquence de coupure  $1/a$ . L'échantillonnage peut entraîner une perte de résolution spatiale et un repliement de spectre.

iii) conversion "analogique - numérique" de l'image saisie. Un numériseur convertit la tension électrique (U: Volt) de chaque pixel en compte numérique (CN) codé sur un nombre prédéfini de bits. L'image est ensuite stockée sur un support physique (CD Rom, disque dur,...) pour être ultérieurement visualisée et analysée, en vue d'une éventuelle prise de décision.

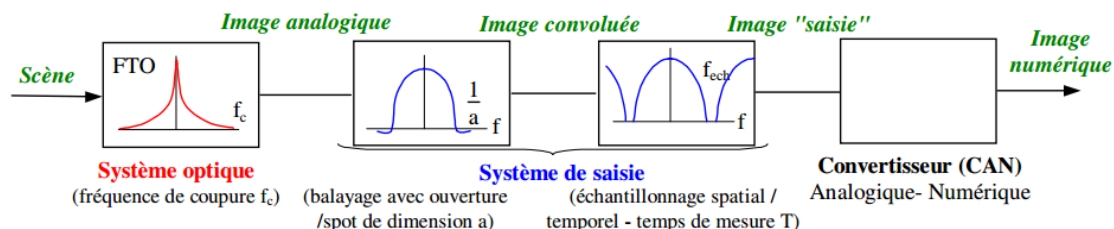


Figure 12. Chaîne d'acquisition d'image numérique

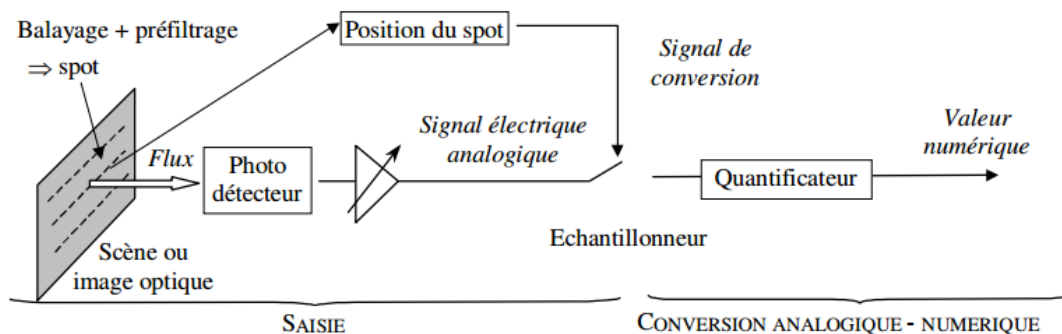


Figure 13. Schéma de principe d'un numériseur d'image optique.

### 1.8.2. Le rehaussement de l'image :

Le mécanisme de formation des images est loin d'être parfait donc présence de différentes formes de bruit, améliorer le contraste d'où l'objectif est de :

- Rehausser le niveau de gris
- Accentuer les caractéristiques

### 1.8.3. morphologie :

C'est un outil permettant d'extraire des composantes d'une image pour décrire et représenter différentes formes.

### 1.8.4. segmentation :

C'est une procédure permettant de partitionner l'image en ses constituants ou objets. La segmentation automatique est la tâche la plus difficile en traitement d'images. Plus la segmentation est meilleure plus l'étape de reconnaissance d'objets est réussie.

### **1.8.5. représentation et description :**

La représentation est l'étape qui vient juste après la segmentation. Le résultat de la segmentation est un ensemble de pixels relatifs à une région. Ces données doivent être converties en une forme traitable par un ordinateur. On peut soit représenter la région ou sa frontière. La description est l'extraction d'attribut permettant de distinguer une classe d'objets d'une autre classe.

### **1.8.6. reconnaissance :**

Est le traitement qui affecte une étiquette (exemple : route, voiture, ...) à un objet en se basant sur ses descripteurs. La base de connaissance contient la connaissance du domaine du problème en cours du traitement. Dans son aspect le plus simple, elle peut consister en coordonnées de l'objet à traiter, ceci permet de réduire l'espace de recherche. Comme elle peut être complexe contenant toutes les défaillances que peut présenter un produit manufacturé [5].

### **1.8.7. restauration d'image :**

Les images subissent des dégradations, ces dégradations sont dues, d'une part au Appareilles d'acquisition et d'autre part aux conditions de prise de vue... Son but est d'améliorer la qualité d'une image, atténuer, supprimer les dégradations La restauration se fait par des technique comme noir et blanc, négative, rotation, etc.

### **1.8.8. Comprissions :**

La compression d'image est une application de la compression de données sur des images numériques. Cette compression a pour utilité de réduire la redondance des données d'une image afin de pouvoir l'emmagasiner sans occuper beaucoup d'espace ou la transmettre rapidement. La compression d'image peut être effectuée avec perte de données ou sans perte [2].

#### **➤ Compression sans perte :**

Appelée aussi compression non destructrice, la qualité de l'image après décompression est la même que celle de l'image originale, le taux de compression de ce type est limité. Ce type de compression on le trouve beaucoup dans le domaine où la précision est majeure comme l'image médicale (IRM par ex.) ou la télédétection (imagerie satellite par ex.).

#### **➤ Compression avec perte :**

C'est une compression destructrice, elle permet de sacrifier certains détails de l'image non récupérable en décompression au profit de réduction de poids. Cette dégradation peut être contrôlée selon la qualité qu'on veut obtenir en fonction du taux de compression choisie.

Ce type de compression on le trouve généralement dans le domaine normal pratique là où la réduction du poids de l'images est très important, comme le domaine multimédia par exemple (web, photographie) où la fidélité envers l'image original n'est pas très importante et le taux de compression sera plus grand que celui d'une compression sans perte du fait qu'on est juste limité par la qualité qu'on souhaite obtenir.

### **1.9. Conclusion**

Dans ce chapitre, nous avons cité les différents traitements que peut subir une image et les termes généraux qui sont très intéressent pour l'étude d'une image. Dans la suite nous présentons l'image RGB et le prétraitement d'image.

## Chapitre 2. L'image RGB et le prétraitement d'image

### 2.1. Introduction

Ce chapitre consiste à expliquer en détails le format d'image RGB ainsi que les prétraitements possibles sur l'image numérique pour améliorer les caractéristiques d'une image afin d'appliquer tout ce que nous devons à l'exploitation.

### 2.2. Définition et résolution

La définition d'une image matricielle est donnée par le nombre de points la composant. En image numérique, cela correspond au nombre de pixels qui composent l'image en hauteur (axe vertical) et en largeur (axe horizontal) : 200 pixels x 450 pixels par exemple. La résolution d'une image matricielle est donnée par un nombre de pixels par unité de longueur. Ce paramètre est défini lors de la numérisation (passage de l'image sous forme binaire), et dépend principalement des caractéristiques du matériel utilisé lors de la numérisation. Plus le nombre de pixels par unité de longueur de la structure à numériser est élevé, plus la quantité d'information qui décrit cette structure est importante et plus la résolution est élevée.

#### 2.2.1. Courbe tonale [16]

Retoucher une image revient à modifier les valeurs de certains pixels. On peut le faire localement (à un endroit bien précis de l'image) ou globalement. Dans ce dernier cas, on utilise un outil appelé « courbe tonale », qui ressemble au dessin ci-contre. Sur l'abscisse, on lit les valeurs originales des pixels et sur l'ordonnée les valeurs après modifications. Sur le graphique ci-contre, tous les pixels de valeurs 100 prendront la valeur 200. Ils vont donc s'éclaircir. La diagonale grise est la courbe où il n'y a aucune modification. En fait, il y a trois courbes tonales : une pour le rouge, une pour le vert et une pour le bleu. On les modifie souvent simultanément de la même façon, mais on peut aussi les modifier séparément.

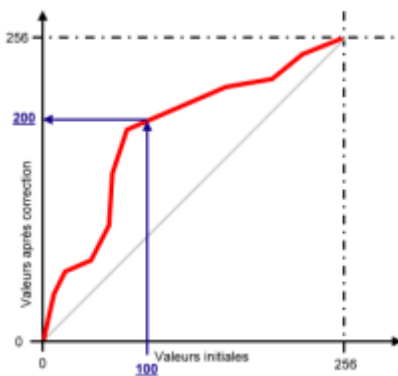
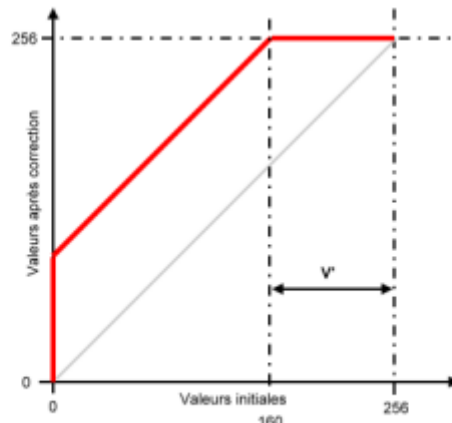


Figure 14. courbe tonale.

#### 2.2.2. Changer la luminosité :

Pour augmenter la luminosité, il suffit d'ajouter une valeur fixe à tous les niveaux



*Pour une valeur de + 96, tous les points de l'espace  $V'$  seront blancs.*

Figure 15. Augmentation de la luminosité.

**Première conséquence** : les points les plus noirs auront une valeur égale à 96 et il n'existera plus aucun point entre 0 et 95.

**Deuxième conséquence** : les points ayant une valeur supérieure à 160 deviendront des points parfaitement blancs, puisque la valeur maximale possible est 255. Il y a donc perte d'informations.

Pour éviter ces pertes d'informations, il faut que la courbe tonale rejoigne les axes tangentiellement, comme dans l'exemple ci-contre. Ainsi, aucun point ne débordera des valeurs limites minimale (0) ou maximale (255). Il sera en particulier possible de revenir en arrière.

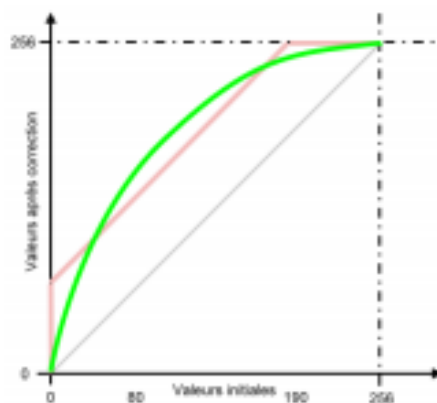
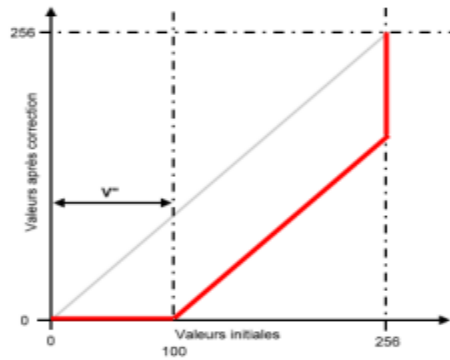


Figure 16. rejoint les axes tangentiellement

Pour diminuer la luminosité il faudra au contraire soustraire une valeur fixe à tous les niveaux.



*Pour une valeur de -100, tous les points de l'espace  $V''$  seront noirs.*

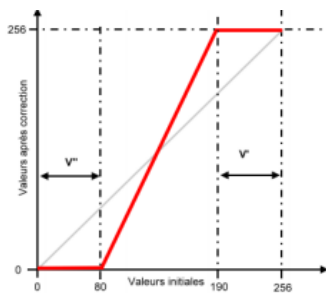
Figure 17. Diminuer la luminosité

**Première conséquence :** les points les plus blancs auront une valeur égale à 156 et il n'existera plus aucun point entre 157 et 255.

**Deuxième conséquence :** les points ayant une valeur comprise entre 0 et 100 deviendront noirs, puisque la valeur minimale possible. Il y aura donc là aussi perte d'informations.

### 2.2.3. Augmenter le contraste

Pour rendre une image plus contrastée, il faut assombrir les points foncés et les points clairs, par exemple comme dans la figure ci-dessous :



*Les points de l'espace  $V''$  seront noirs et ceux de l'espace  $V'$  blancs.*



Figure 18. Augmentation du contraste

## 2.3. Quelques traitements classiques [17]

### 2.3.1. Négatif

Obtenir le négatif d'une image est très simple, toutes les composantes  $x$  de tous les pixels de l'image sont remplacées par  $255-x$ .





Figure 19. le négatif d'un image

### 2.3.2. Rouge

Chaque pixel de l'image est une combinaison de rouge, de vert et de bleu. En assignant la valeur 0 aux composantes verte et bleue, on obtient l'image de droite.

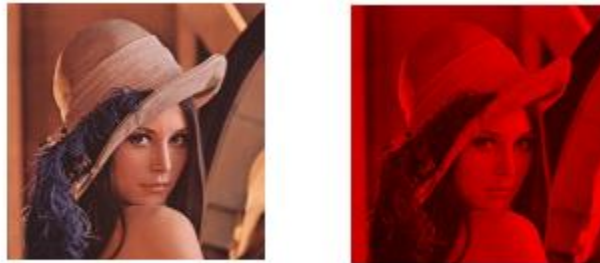


Figure 20. Le niveau du rouge d'une image

### 2.3.3. Niveaux de gris

Dans une image en niveaux de gris, chaque pixel est noir, blanc, ou a un niveau de gris entre les deux. Cela signifie que les trois composantes ont la même valeur. L'œil est plus sensible à certaines couleurs qu'à d'autres. Le vert (pur), par exemple, paraît plus clair que le bleu (pur). Pour tenir compte de cette sensibilité dans la transformation d'une image couleur en une image en niveaux de gris, on ne prend généralement pas la moyenne arithmétique des intensités de couleurs fondamentale, mais une moyenne pondérée.

La formule mathématique standard donnant le niveau de gris en fonction des trois Composantes est :

$$\text{gris} = \text{int}(\text{round}(0.299 \cdot \text{rouge} + 0.587 \cdot \text{vert} + 0.114 \cdot \text{bleu}))$$



Figure 21. Le niveau de gris d'une image

### 2.3.4. Seuillage

Le seuillage d'image est la méthode la plus simple de segmentation d'image. À partir d'une image en niveau de gris, le seuillage d'image peut être utilisé pour créer une image comportant uniquement deux valeurs, noir ou blanc (monochrome). On remplace un à un les pixels d'une image par rapport à une valeur seuil fixée (par exemple 123). Ainsi, si un pixel a une valeur supérieure au seuil (par exemple 150), il prendra la valeur 255 (blanc), et si sa valeur est inférieure (par exemple 100), il prendra la valeur 0 (noir).



Figure 22. Le seuillage d'une image

### 2.3.5. Sépia

En photographie, le sépia est une qualité de tirage qui ressemble au noir et blanc, mais avec des variations de brun, et non de gris. La couleur sépia dans le système RVB est  $S(94, 38, 18)$ . Dans la transformation d'une image couleur en une image en nuances de sépia, on tient compte d'un seuil ( $0 < \text{seuil} < 255$ ), qui sépare le sépia assombri et le sépia éclairci. La transformation se fait alors pixel par pixel en deux temps. Pour chaque pixel, On calcule d'abord un niveau de gris qui est la moyenne  $m$  des intensités de rouge, vert et bleu. Puis, si le gris obtenu est foncé ( $m < \text{seuil}$ ), on le remplace par une couleur du segment  $NS$ , couleur d'autant plus proche du noir que  $m$  est petit. Si le gris est plus clair ( $m > \text{seuil}$ ), il est remplacé par une couleur du segment  $SB$ , couleur d'autant plus proche du blanc  $B$  que  $m$  est grand.



Figure 23. Sépia d'une image.

### 2.3.6. Lissage

Filtrage est une transformation mathématique (produit de convolution) permettant de modifier la valeur d'un pixel en fonction des valeurs des pixels avoisinants, affectées de coefficients. Les calculs sont faits pour chacune des trois composantes de couleur.



Figure 24. Lissage d'une image

## 2.4. Prétraitement des images

Le traitement d'une image a pour but d'extraire l'information qu'elle contient pour résoudre un problème en éliminant l'information inutile. Le traitement s'effectue en plusieurs étapes:

- Prétraitement pour améliorer la qualité des images (Filtrage)
- Traitement pour simplifier l'image (binarisation, opérateurs morphologiques)
- Décomposition de l'image pour extraire les caractéristiques (Contour, segmentation)

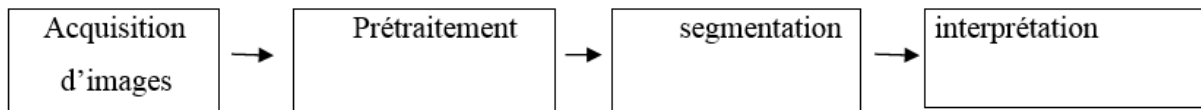


Figure 25. L'analyse d'images est une chaîne de traitement de l'information

Le prétraitement regroupe l'ensemble des processus visant à améliorer les caractéristiques d'une image.

- Le lissage local : il s'agit de réduire le bruit, ou les petites variations, présent dans une image. L'intensité d'un pixel est transformée en fonction des intensités sur un petit voisinage du pixel.
- L'amélioration d'images consiste à modifier les caractéristiques visuelles de l'image (contraste, ...) pour faciliter son interprétation par l'œil humain.
- La restauration d'images a pour but de supprimer les dégradations subies par une image à l'aide de connaissances a priori sur ces dégradations.

### 2.4.1. Lissage local

Le lissage local (ou filtrage) consiste à réduire le bruit présent dans une image en étudiant, pour chaque pixel, les valeurs d'intensité sur son voisinage.

#### ➤ Moyennage

Une première classe d'approche est basée sur la redondance d'informations. La nouvelle valeur d'un pixel est calculée par moyennage des valeurs sur un voisinage.

Cette opération linéaire peut être vue comme la convolution discrète de l'image par un masque.

$$I'(i,j) = \sum_{(m,n) \in V} h(m,n) (i-m, j-n)$$

$$\sum_{(m,n) \in V} h(m,n) = 1$$

ou  $I$  est l'intensité de l'image originale,  $I'$  est l'intensité de l'image filtrée,  $V$  est le voisinage utilisé et  $h$  est le masque de convolution.

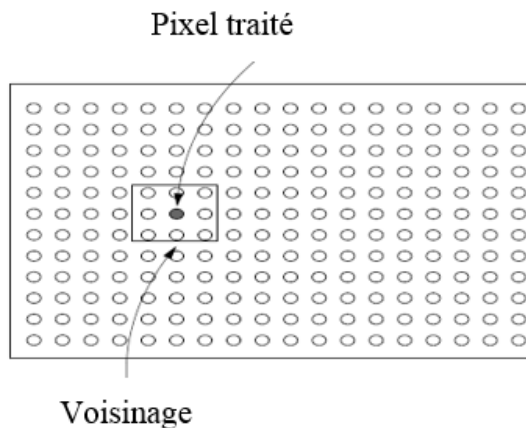


Figure 26. voisinage du pixel  $x, y$ .

Le moyennage sur un voisinage 3x3 :

$$H = 1/9 \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

+ Le moyennage est un filtre passe-bas.

+ Rend l'image floue, en particulier les contours.

+ Élimine les dégradations locales de faibles dimensions. Valide lorsque les objets présents dans l'image sont de dimensions supérieures aux dégradations.

Une amélioration du filtre moyenne consiste à jouer sur les valeurs des coefficients

du masque :

$$H1 = 1/10 \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

C'est le cas des filtres binomiaux pour lesquels les valeurs des coefficients sont générées par le triangle de Pascal :

$$H2 = 1/16 \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

### ➤ Filtres médians

Les filtres de moyennage ont tendance à rendre l'image floue et donc à perdre de l'information sur les contours caractérisés par des fortes variations d'intensité. Pour diminuer cet effet, on ne moyenne plus sur le voisinage mais on prend la valeur médiane sur ce voisinage. C'est le filtre médian.

Exemple pour un voisinage 3x3 :

$$I = 1/16 \begin{bmatrix} 2 & 12 & 12 \\ 2 & 12 & 60 \\ 2 & 2 & 12 \end{bmatrix}$$

La valeur médiane est ici 12.

- + Filtre non-linéaire.
- + Élimine le bruit impulsionnel.
- + Préserve l'information de contour et peut être appliqué itérativement.
- + Élimine les contours très fins. Un voisinage adapté permet de limiter cet effet.

### ➤ Autres filtres

Toujours dans l'objectif d'améliorer les caractéristiques du filtre et en particulier la préservation des contours dans l'image, on peut utiliser des filtres plus évolués.

Il en existe un certain nombre. Nous donnons ici l'exemple d'un filtre faisant intervenir des variations d'intensité.

Soit  $d(i, j, k, l)$  la variations d'intensité entre le pixel  $(i, j)$  et  $(k, l)$  :

$$d(i, j, k, l) = \begin{cases} |I(i, j) - I(k, l)| & \text{si } I(i, j) \neq I(k, l), \\ 1/2 & \text{si non} \end{cases} \quad \begin{cases} i - 1 \leq k \leq i + 1 \\ j - 1 \leq l \leq j + 1 \end{cases}$$

alors les coefficients du masque de convolution sont déterminés par :

$$h(m, n) = \frac{(1/d(i, j, i+m, j+n))}{\sum_{(m, n) \in V} 1/d(i, j, i+m, j+n)}$$

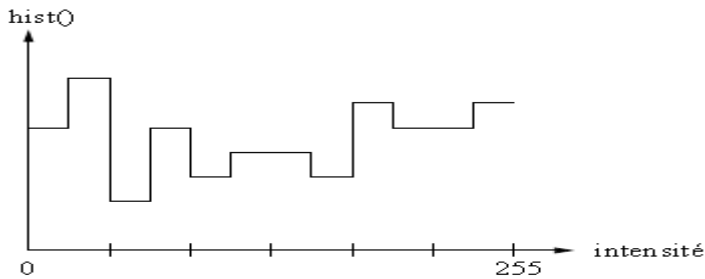
- + La distance est plus importante sur un contour que dans une région homogène.
- + Les points fortement bruités ont peu d'influence ( $h(m, n)$  petit).
- + Si le pixel est sur un contour, les pixels voisins de même région auront un poids important alors que les pixels voisins de région différente auront peu de poids. Le contour est de cette manière préservé.

### 2.4.2. Amélioration d'images

L'amélioration d'images consiste modifier les caractéristiques visuelles de l'image de manière à en faciliter son interprétation par l'œil humain. Il peut s'agir de rehausser les contrastes, d'accentuer certaines intensités pour mettre en valeur une région, ... Les histogrammes sont fréquemment utilisés pour effectuer ce type d'opérations.

### ➤ Les histogrammes

L'histogramme d'une image est la fonction qui associe à une valeur d'intensité  $i$  le nombre de pixels dans l'image ayant cette valeur.



+ Pour une image couleur, il y a un histogramme par composante.

+ L'histogramme peut être normalisé pour donner une estimation de la densité de probabilité des pixels :

$$P(i) = \text{hist}(i) / \sum_j \text{hist}(j),$$

$$\sum_i p(i) = 1$$

+ Un histogramme peut avoir un pic (uni modal), deux pics (bi modal) ou plusieurs pics (multi modal).

#### ➤ **Modifications d'histogrammes**

Pour modifier les caractéristiques de l'image (accentuer les contrastes en général), une approche générale consiste à appliquer une fonction qui associe à chaque valeur d'intensité dans l'image une nouvelle valeur. Cette fonction va modifier l'histogramme de l'image.

Soit  $i$ ,  $i < \text{Max}$  les valeurs d'intensité de l'image traitée, on considère alors les transformations du type :

$i' = T(i)$ , qui donne une nouvelle valeur d'intensité  $i'$  pour chaque valeur  $i$  de l'image. On suppose que la fonction  $T()$  est telle que :

- $T(i)$  est monotone (souvent croissante) sur l'intervalle. Cette condition assurant que l'ordre des intensités est préservé après transformation.
- $0 \leq T(i) \leq \text{Max}$  pour  $0 \leq i \leq \text{Max}$  qui garantit que la nouvelle image est cohérente avec les niveaux d'intensité autorisés.
- La transformation inverse satisfait les deux conditions précédentes.

#### ➤ **Transformations linéaires**

Les transformations linéaires d'histogrammes sont nombreuses et variées. Elles permettent d'accentuer une zone d'intensité ou de modifier la répartition des valeurs d'intensité. Un exemple de transformation linéaire est le recadrage. Le recadrage consiste à modifier l'intervalle des valeurs d'intensité (la dynamique) de façon à obtenir pour l'image améliorée un intervalle de valeurs maximal.

Si  $[i1, i2]$  est l'intervalle de l'image traitée, alors la transformation équivalente s'écrit simplement :  $T(i) = \text{Max} (i - i1) / (i2 - i1)$ .

### ➤ Transformations non-linéaires

Une modification d'histogramme très répandue pour augmenter le contraste de manière automatique est la linéarisation d'histogramme. Le principe est de transformer l'image de manière à obtenir un histogramme plat, soit une distribution uniforme des intensités. Cela revient à maximiser l'entropie de l'image et donc à obtenir théoriquement une image présentant une information maximale.

Supposons tout d'abord que l'on soit dans le cas continu et que la transformation  $T$  soit strictement croissante. Alors :

$$p'(i') = p(i) \cdot di/di', \text{ avec } i' = T(i). \text{ D'où : } di' = p(i) di,$$

Soit :

$$i' = T(i) = \int_0^i p(w)dw$$

Dans le cas discret, la transformation s'écrit :

$$T(i) = \sum_{j=0}^i \text{hist}(j) / \sum_{j=0}^{\text{MaxInt}} \text{hist}(j),$$

+ A noter que dans le cas discret, et du fait que l'histogramme soit une approximation d'une fonction de densité de probabilité, l'histogramme résultant est très rarement parfaitement plat.

+ Pour une image majoritairement claire la linéarisation va augmenter la dynamique de la partie sombre de l'histogramme au détriment de la partie claire.

+ Permet de faire des comparaisons d'images sur une même base.

+ L'opération peut s'effectuer par régions : linéarisation adaptative.

### 2.5. Conclusion

Dans ce chapitre nous avons représenté quelques traitement important sur les images couleur (RGB) et les opérations nécessaires de leur prétraitement, Dans la suite nous présentons les différentes techniques utilisées pour la détection des cercles et des ellipses dans une image avec l'implémentation de la transforme de hough.

## Chapitre 3. La détection des cercles et des ellipses dans les images.

### 3.1. Introduction

Ce chapitre consiste à expliquer la transformation de Hough ainsi que les exemples d'utilisation de cette technique sur plusieurs cas d'images réelles.

Dans la détection d'éléments spécifiques, il existe des algorithmes qui, pour identifier ces formes élémentaires, entreprennent de suivre les contours pour finalement les lier par des critères plus ou moins complexes afin de remonter jusqu'aux formes recherchées. Une autre approche de ce problème serait de tenter d'accumuler des évidences sur l'existence d'une forme particulière telle qu'une droite, un cercle ou une ellipse. C'est cette démarche qui a été adoptée dans la transformée de Hough.

#### La transformée de Hough : [18]

La transformée de Hough a été développée par Paul Hough en 1962, et a été brevetée par IBM. Dans les dernières décennies, la transformée de Hough est devenue un outil standard dans le domaine de la vision artificielle. Elle permet la détection de droites, de cercles ou d'ellipses de façon traditionnelle. Elle peut aussi être étendue à des cas de description d'objets plus complexes. La transformée de Hough permet d'avoir une bonne robustesse contre les discontinuités et les éléments manquants d'une structure. Le principe général de la transformée de Hough est d'établir une projection entre l'espace de l'image et un espace de paramètres représentatif de la forme recherchée. Nous verrons par la suite comment adapter ce modèle à la détection des ellipses et des cercles.

### 3.2. Cercle Hough Transformation de Hough d'un cercle (CHT) [19]

La transformation de Hough peut être utilisée pour déterminer les paramètres d'un cercle quand un certain nombre de points qui tombent sur le périmètre sont connus. Un cercle de rayon  $R$  et le centre  $(a, b)$  peut être décrit avec les équations paramétriques

$$x = a + R \cos(\theta)$$

$$y = b + R \sin(\theta)$$

Lorsque les balayages de l'angle à travers la gamme complète de 360 degrés aux points  $(x, y)$  tracer le périmètre d'un cercle. Si une image contient de nombreux points, dont certains tombent sur des périmètres de cercles, puis le travail du programme de recherche est de trouver des triplets paramètres  $(a, b, R)$  pour décrire chaque cercle. Le fait que l'espace de paramètres est en 3D permet une mise en œuvre directe de la technique plus coûteuse de Hough dans la mémoire de l'ordinateur et du temps. Rechercher avec  $R$  fixe. Si les cercles dans une image sont de rayon connu  $R$ , puis la recherche peut être réduite à 2D. L'objectif est de trouver le  $(a, b)$  les coordonnées des centres.  $x = a + R \cos(\theta)$ ,  $y = b + R \sin(\theta)$  Le lieu de  $(a, b)$  des points dans l'espace des paramètres chute sur un cercle de rayon  $R$  centré sur  $(x, y)$ . Le véritable point central sera commun à tous les milieux de paramètres, et peut être trouvé avec un tableau accumulateur Hough.



### 3.2.1. Limites [20] [21]

#### Dépendance de la qualité des données:

La performance du HT est fortement dépendante des résultats du détecteur de bord. L'entrée image doit être soigneusement choisie pour une plus grande détection de bord. En utilisant le HT sur les images bruitées réduit l'efficacité de l'algorithme en raison de quelques points de bordure étant oubliées parce qu'elles ne sont pas définies comme bords.

#### Objet taille / position contraintes:

CHT dépend d'une valeur prédéfinie du rayon des cercles. Un programme qui dépend entièrement sur les entrées de l'utilisateur n'est pas pratique pour un usage général. La position de l'objet par rapport à l'image joue un rôle important sur la façon précise qu'il peut être détecté. Par exemple, l'exécution du CHT sur un deux images des mêmes pièces de monnaie peut détenir des résultats différents en fonction de la distance entre les pièces de monnaie et de la caméra au moment de la photo a été prise.

### 3.2.2. Avantage

- Conceptuellement simple.
- La mise en œuvre facile.
- Poignées manquantes et occlus des données très gracieusement.
- Peut être adapté à de nombreux types de formes, pas seulement les lignes.

### 3.2.3. Inconvénients

- Complexe pour les objets avec plan informatique de nombreux paramètres.
- Cherche un seul type d'objet.
- Peut être "dupé" par des "lignes apparentes".
- La longueur et la position d'un segment de ligne ne peut pas être déterminée.
- Segments de ligne colinéaires ne peuvent pas être séparés

### 3.2.4. Représentation géométrique d'un cercle et une ellipse :

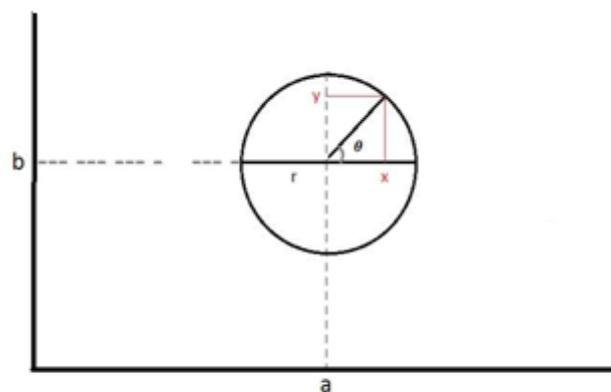


Figure 27. Cercle géométrie.

$$\text{Avec } \begin{cases} x = a + r \cdot \cos(\theta) \\ y = b + r \cdot \sin(\theta) \end{cases}$$

L'équation du cercle est donnée par la formule suivant :

$$(x - a)^2 + (y - b)^2 = r^2$$

Avec : a et b : centre de cercle et r : le rayon de cercle

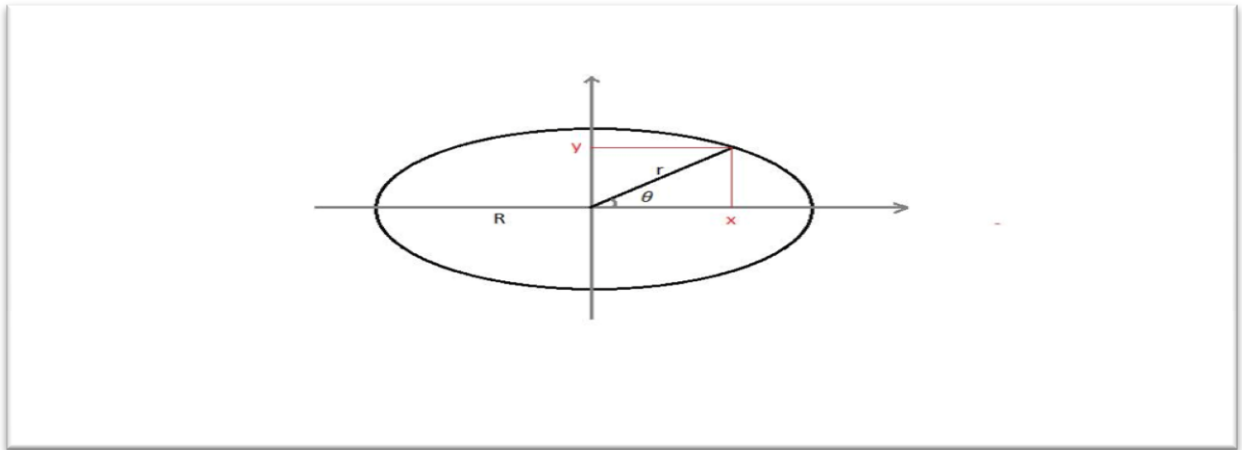


Figure 28. Ellipse géométrie

$$\text{Avec } \begin{cases} x = a + r \cdot \cos(\theta) \\ y = b + r \cdot \sin(\theta) \end{cases}$$

L'équation d'ellipse donne pare le formule suivant :

$$\alpha(x - a)^2 + \beta(y - b)^2 = R^2$$

$$\begin{cases} x = a + r * \cos(\theta) \\ y = b + r * \sin(\theta) \end{cases}$$

Pour trouver la relation entre r, R,  $\theta$ ,  $\alpha$  et  $\beta$  remplacer se dernier dans l'équation d'ellipse on obtient :

$$\alpha(a + r * \cos(\theta) - a)^2 + \beta(b + r * \sin(\theta) - b)^2 = R^2$$

$$\alpha(r * \cos(\theta))^2 + \beta(r * \sin(\theta))^2 = R^2$$

$$r^2(\alpha * \cos^2(\theta) + \beta * \sin^2(\theta)) = R^2$$

$$r^2 = \frac{R^2}{\alpha * \cos(\theta)^2 + \beta * \sin(\theta)^2}$$

Pour  $\alpha = \beta = 1$

$R=r$  donc on peut déduire que le cercle est un cas particulier dans l'ellipse.

a et b : centre de cercle

et r : le rayon de cercle

### 3.3. Les ellipses

Pour chaque paire de pixels  $(x_1, y_1)$  et  $(x_2, y_2)$ , nous supposons qu'ils sont deux sommets sur l'axe principal d'une ellipse. Nous pouvons également supposer qu'ils sont deux sommets sur le petit axe d'une ellipse, mais il augmentera le temps de calcul parce que nous avons besoin de vérifier les pixels dans une gamme plus large de l'image. Ensuite, nous pouvons calculer quatre paramètres pour l'ellipse supposée comme suit:

$$x_0 = (x_1 + x_2) / 2 \quad (1)$$

$$y_0 = (y_1 + y_2) / 2 \quad (2)$$

$$a = [(x_2 - x_1)^2 + (y_2 - y_1)^2]^{1/2} / 2 \quad (3)$$

$$\alpha = \text{atan} [(y_2 - y_1) / (x_2 - x_1)], \quad (4)$$

Où  $(x_0, y_0)$  est le centre de l'ellipse supposée,  $a$  est la demi-longueur du grand axe et  $\alpha$  est l'orientation de l'ellipse.

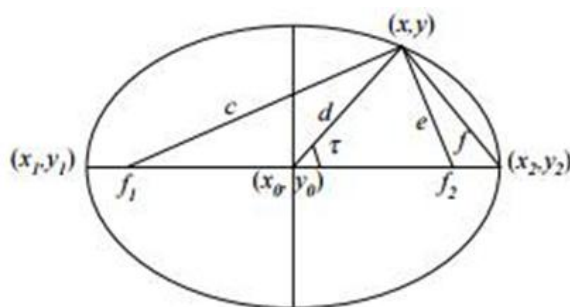


Figure 29. Deuxième géométrie de l'ellipse

La figure 29 montre la géométrie ellipse.  $f_1$  et  $f_2$  sont des foyers de l'ellipse et  $(x, y)$  est le troisième point utilisé pour calculer le cinquième paramètre. La distance entre  $(x, y)$  et  $(x_0, y_0)$  doit être inférieure à la distance entre  $(x_1, y_1)$  et  $(x_0, y_0)$  ou entre  $(x_2, y_2)$  et  $(x_0, y_0)$ . Donc, la demi-longueur de l'axe mineur peut être estimée par l'équation suivante :

$$b^2 = (a^2 d^2 \sin^2 \tau) / (a^2 - d^2 \cos^2 \tau) \quad (5)$$

Où  $\cos(\tau)$  est :

$$\cos(\tau) = (a^2 + d^2 - f^2) / (2ad) \quad (6)$$

Et  $d$  est la distance entre  $(x, y)$  et  $(x_0, y_0)$ . Par conséquent, en utilisant les équations (1) - (6), il est possible de calculer l'ensemble des cinq paramètres d'une ellipse. Étant donné que nous ne devons voter sur la demi-longueur de l'axe mineur, nous pouvons utiliser un tableau accumulateur à une dimension. Si les votes atteignent un seuil, une ellipse est trouvée et la sortie, nous les paramètres de cette ellipse détectée et supprimer tous les pixels sur cette ellipse de l'image. Après cette paire de pixels est vérifiée, nous avons clairement le tableau accumulateur et aller à La paire suivant.

### 3.4. Les différentes techniques de détection de contour :

#### 3.4.1. Filtre de Prewitt :

La matrice qui correspond au filtrage horizontal, faisant ressortir essentiellement les contours verticaux, selon l'opérateur de Prewitt, s'écrit  $h_x = [-1 \ 0 \ 1]$  tandis que la matrice verticale  $h_y$  est sa transposée. Les deux convolutions avec le tableau de valeurs initiales créent deux tableaux  $G_x$  et  $G_y$  à l'origine du tableau  $G$  sur lequel on peut localiser les maximums.

#### 3.4.2. Filtre de Sobel :

Le principe de ce filtre est que l'opérateur calcule le gradient de l'intensité de chaque pixel. Ceci indique la direction de la plus forte variation du clair au sombre, ainsi que le taux de changement dans cette direction. On connaît alors les points de changement soudain de luminosité, correspondant probablement à des bords, ainsi que l'orientation de ces bords. L'opérateur utilise des matrices de convolution. La matrice (généralement de taille  $3 \times 3$ ) subit une convolution avec l'image pour calculer des approximations des dérivées horizontale et verticale. Soit  $A$  l'image source, et deux images qui en chaque point contiennent des approximations respectivement de la dérivée horizontale et verticale de chaque point. Ces images sont calculées comme suit :

$$G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} * A \qquad G_y = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} * A$$

En chaque point, les approximations des gradients horizontaux et verticaux peuvent être combinées comme suit pour obtenir une approximation de la norme du gradient : On implémente alors le script suivant :

$$G = \sqrt{G_x^2 + G_y^2}$$

On peut également calculer la direction du gradient comme suit :

$$\varphi = \tan^{-1} \frac{G_y}{G_x}$$

Où, par exemple, vaut 0 pour un contour vertical plus foncé à gauche.

#### 3.4.3. Filtre de Canny :

La méthode Canny trouve bords en recherchant des maxima locaux du gradient de  $I$ . Le gradient est calculé en utilisant le dérivé d'un filtre gaussien. Le procédé utilise deux seuils, pour détecter des bords forts et faibles, et comprend les bords faibles de la sortie que si elles sont reliées à des bords forts. Cette méthode est donc moins susceptible que les autres à se laisser bernier par le bruit, et plus susceptibles de détecter les vrais bords faibles.

### 3.4.4. Filtre de robert :

Les filtres de Roberts sont une approche discrète de la dérivée de pas 1 d'une fonction : le gradient de cette fonction. Si  $I(x,y)$  représente un pixel dans une image, alors les amplitudes des gradients en  $x$  et en  $y$  peuvent s'écrire respectivement:

$$\begin{cases} G_x = I(x + 1, y) - I(x, y) \\ G_y = I(x, y + 1) - I(x, y) \end{cases}$$

Cela revient à convoluer l'image avec les deux filtres  $R_x = [-1 \ 1]$  et  $R_y = \text{transposa}([-1 \ 1])$ . L'amplitude du gradient peut être alors calculée de plusieurs façons:

$$G_{1(x,y)} = \sqrt{G_x^2 + G_y^2}$$

$$G_{2(x,y)} = \text{Max}(\text{abs}(G_x) + \text{abs}(G_y))$$

$$G_{3(x,y)} = \text{abs}(G_x) + \text{abs}(G_y)$$

Et la direction du gradient est donnée par :

$$\varphi = \tan^{-1} \frac{G_x}{G_y}$$

Or le bruit peut aussi être une brusque variation locale des niveaux de gris : ces filtres sont donc très sensibles au bruit car ils accentuent, par dérivation, le bruit présent dans l'image. De plus, ces filtres donneront un contour épais si celui-ci est un contour de type "rampe".

## 3.5. Résultat :

### 3.5.1. Explication de la première méthode :

Pour la détection de cercles d'un rayon donné, on commence par parcourir tous les pixels de l'image de contours. Quand un pixel appartient à un contour, nous parcourons l'image afin de connaître le rayon du cercle du centre courant. Si le rayon correspondant au rayon recherché, on incrémente la valeur de coordonnées du centre du cercle.

Ensuite, on parcourt le cercle dont leur valeur appartient au seuil fixé et ainsi que toutes les pixels de l'image. Si les coordonnées du pixel courant correspondent à l'équation de cercles courante, c'est que le pixel appartient au cercle, on le met donc à blanc. Cela permet donc de dessiner les cercles de centre  $a$  et  $b$  et du rayon recherché.

### 3.5.2. Explication de deuxième méthode

Cette deuxième méthode consiste à, pour chaque pixel  $(a,b)$ , cumuler le niveau du gris du contour en faisant la rotation du point sur un centre  $(a,b)$  et pour une distance constante  $r$  qui est le rayon du cercle à détecter.

### les détails de l'ordinateur utilisé dans l'application :

Edition Windows : Windows familles unilingue

© 2015 Microsoft Corporation. Tous droits réservés.

Système processeur : Intel(R) core(TM) i5-3230M CPU @ 2.60GHz

Mémoire installée (RAM) : 4,00 Go

Type du système : système d'exploitation 64 bit, processeur x64

Tableau 2. Comparaison entre les deux méthodes

Les figures	les rayons	la résolution	le temps (sec) la méthode 01	le temps (sec) la méthode 02
Pièces de 100 dinars	$r=33$	480×270	11.828	1.535
Les Pinions	$r_1=71$	480×270	19.500	0.785
	$r_2=62$	480×270	18.146	0.949
Les canettes	$r=19$	480×360	6.742	2.630
Le piston	$r_1=29$	480×360	11.843	2.355
	$r_2=12$	480×360	3.654	2.827
Les joints de culasse	$r=21$	480×360	7.722	2.524
Les roulements	$r=25$	480×360	9.734	2.480
hough_test2	$r=64$	480×360	30.761	1.475

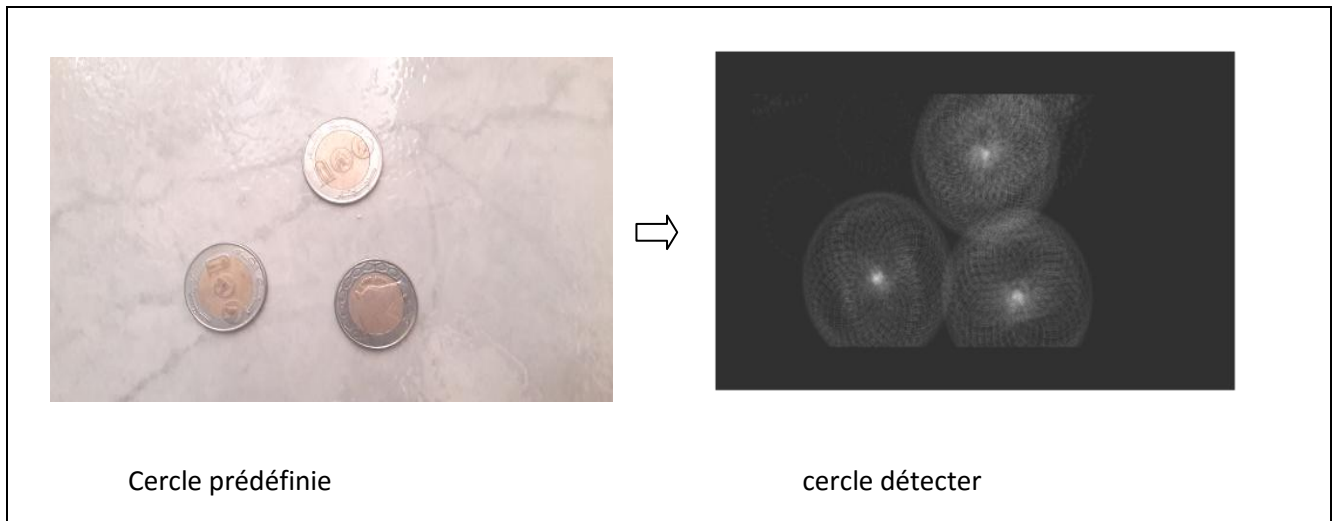


Figure 30. Pièces de 100 dinars

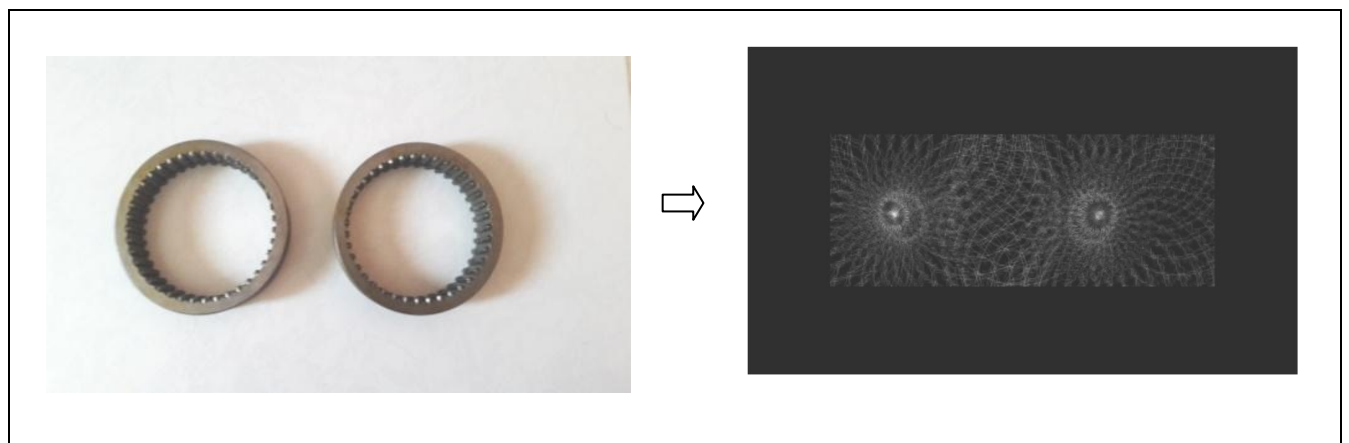


Figure 31. Pinions

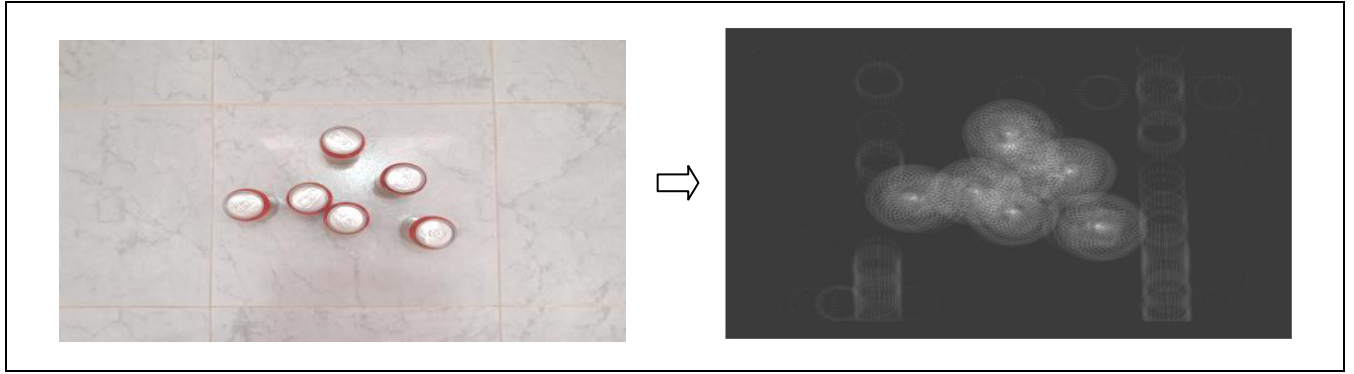


Figure 32. Cigarettes

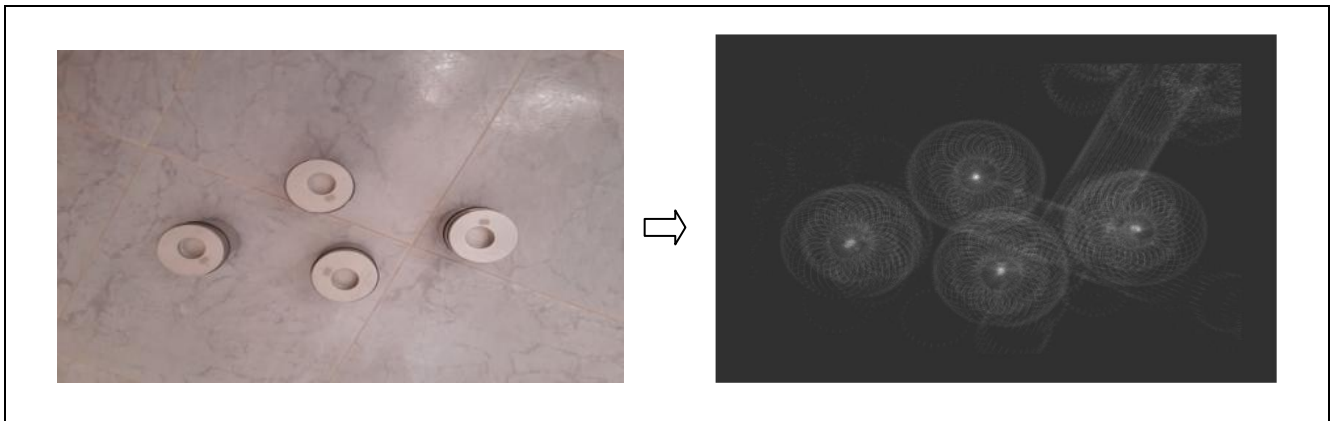


Figure 33. Piston

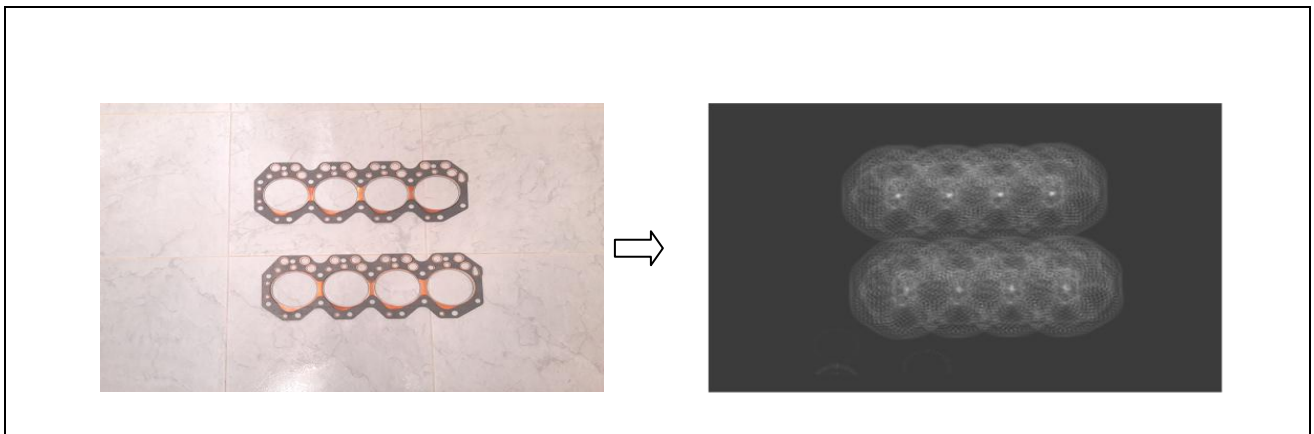


Figure 34. Joint de culasse

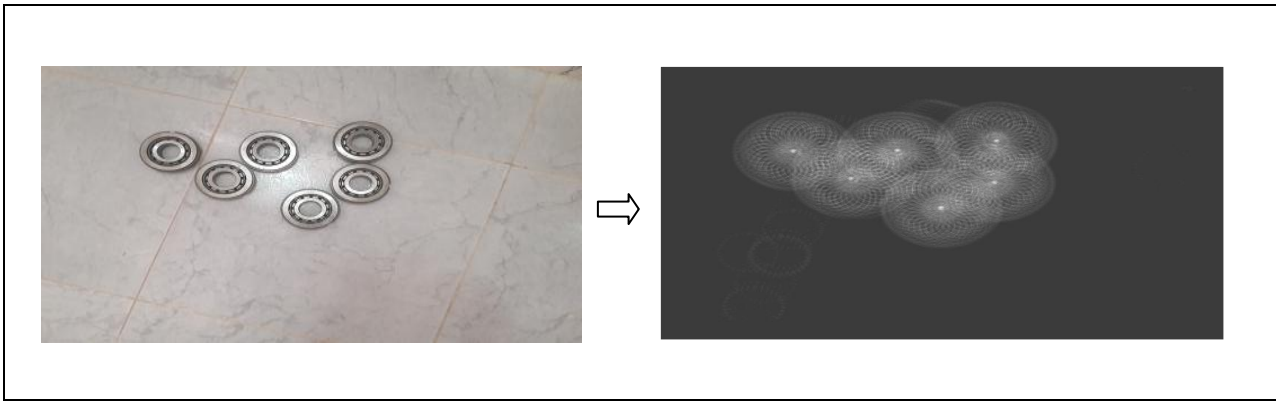


Figure 35. Roulement

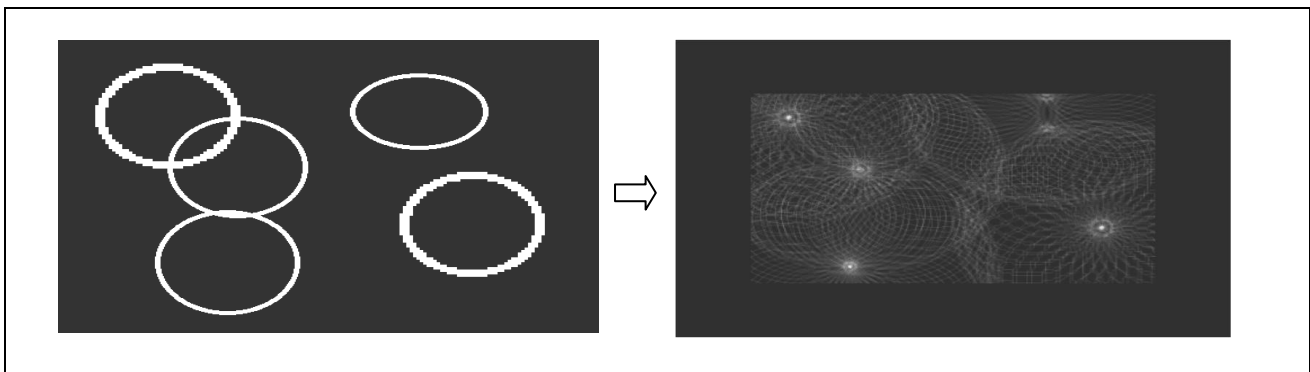


Figure 36. hough\_test2

### 3.5.3. Comparaison les résultats de détection du cercle selon le paramètre choisis

#### ➤ Comparaison selon le filtre utiliser

Tableau 3. Comparaison entre les quatre filtres

filtre figures	Sobel		Canny		Prewitt		Robert	
	le temps (sec)	le taux de réussite	temps (sec)	taux de réussite	temps (sec)	taux de réussite	temps (sec)	taux de réussite
(a)	2.718	10/10	2.732	6/10	2.735	10/10	2.706	9/10
(b)	3.085	10/10	3.129	0/10	3.095	10/10	3.089	10/10
(c)	2.624	9/10	2.640	9/10	2.636	9/10	2.626	8/10
(d)	2.439	9/10	2.467	8/9	2.429	9/10	2.429	9/10



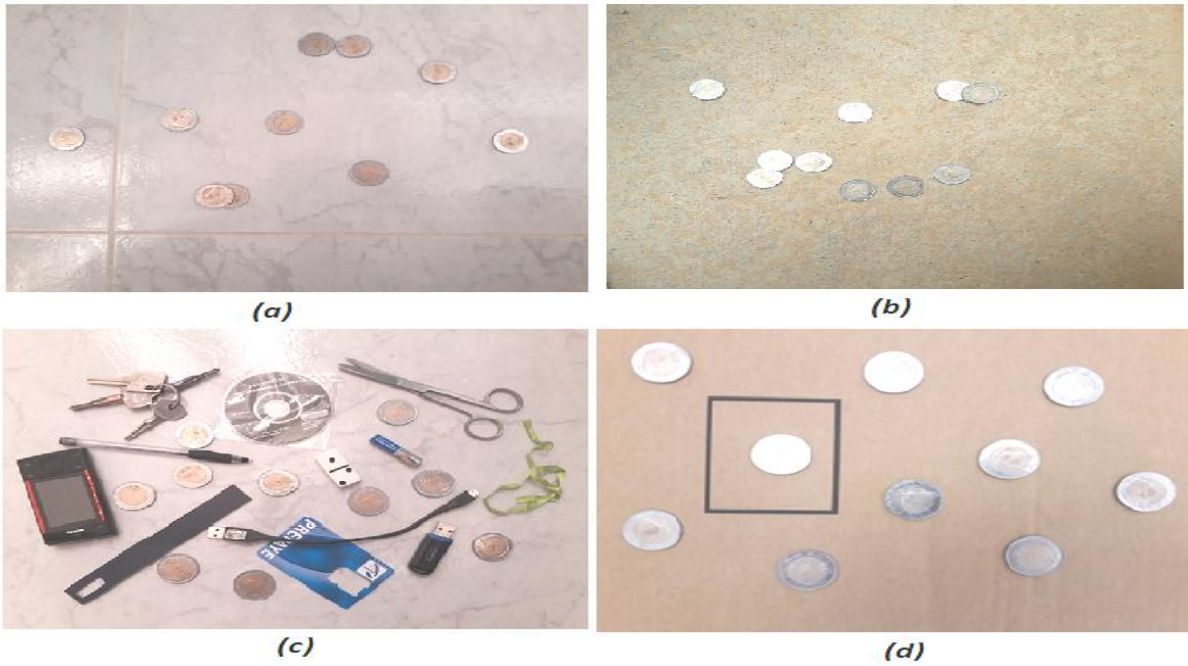


Figure 37. Quatre différents arrière plans.

➤ Explication avec un exemple (10 Dinars)

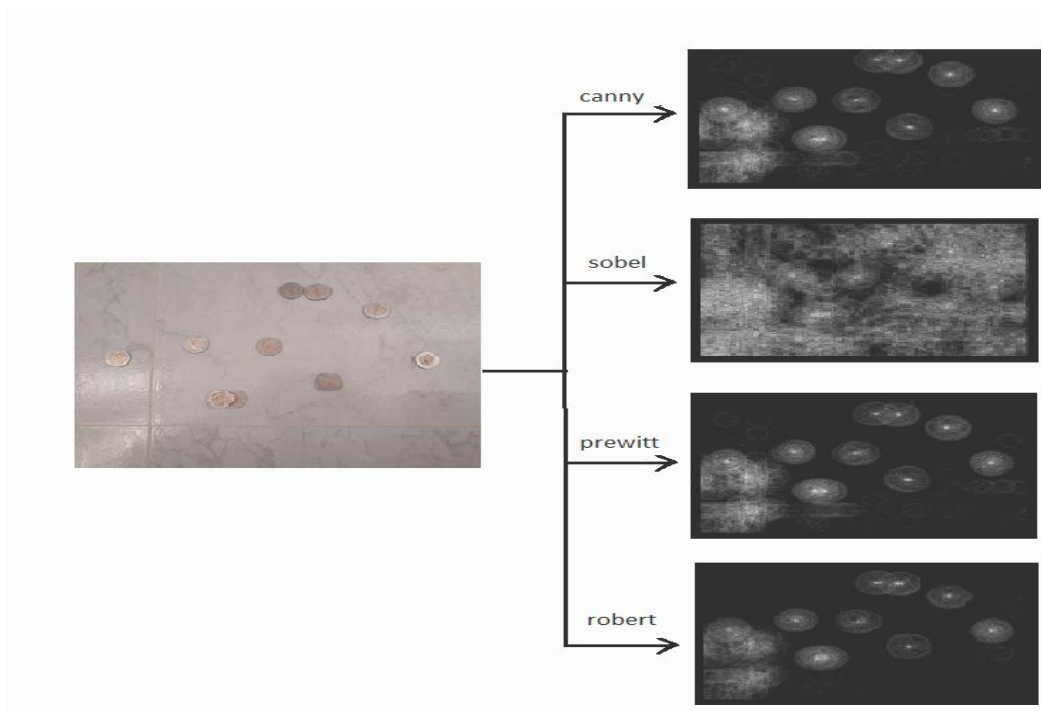


Figure 38. Exemple d'utilisation des techniques de détection de contours

➤ **Comparaison selon l'épaisseur « d » utilisé**

Tableau 4. Comparaison entre les quatre épaisseurs.

Le paramètre	le temps (sec)	le taux de réussite
d1=1	2.712	10/10
d2=2	3.913	10/10
d3=3	4.857	9/10
d4=4	5.887	7/10

➤ **Comparaison selon l'angle "θ" utilisé.**

Tableau 5. Comparaison entre les 5 pas d'angle utilisés

Le paramètre	le temps (sec)	le taux de réussite
θ=0.05	10.583	10/10
θ=0.1	5.311	10/10
θ=0.2	2.736	10/10
θ=0.3	1.820	9/10
θ=0.5	1.320	9/10

Les étapes de détection de cercle :

- Lire l'image.
- Convertir l'image couleur à aux niveaux de gris.
- Détecter les contours d'image
- Donne la valeur de rayon du cercle réel.
- Donne l'intervalle « d » épaisseur de cercle.
- Initialiser l'accumulateur à zéro.
- définir la variation de centre du cercle.
  - Définir l'équation du cercle par la relation entre son rayon et centre.
    - Si la différence entre le rayon réel et le rayon calculé est inférieur ou égale à « d ».
    - Alors accumuler le niveau du gris dans l'accumulateur.
- L'affichage de l'image qui contient les cercles détectés.

### 3.6. Discussion

Il est observé que la deuxième technique de transformation souffre de temps de calcul très grand, ce qui est optimisé pour la première technique en gardant un taux de réussite invariant.

Le pas d'angle joue un rôle très important sur le temps de calcul, mais en augmentant le pas, le temps de calcul a diminué jusqu'à 10% malgré que le taux de réussite diminue aussi de 10%.

L'intervalle de cumul du rayon du cercle ( $d$ ) pose un problème dans le cas d'augmentation inutile, cet intervalle peut être augmenté seulement si les formes à détecter sont des cercles déformés, ce qui n'est le cas pour nos exemples, alors l'intervalle a diminué le taux de réussite ainsi qu'il a augmenté le temps de calcul d'une façon importante.

Pour la technique de détection de contour, deux méthodes ont donné des résultats intéressants pour le cas de fond unis (Sobel et Prewitt), mais pour le cas d'un fond aléatoire (ce qui est le cas réel) la méthode de Canny a donné un prétraitement très distinguant facilitant la tâche à la transformation de Hough.

### 3.7. Conclusion

Ce chapitre présente la technique de détection de formes prédéfinies et son application sur le cas du cercle. Le changement de paramètre de détection ont fait des changements apparentes sur son rendement, ce qui doit être pris en compte pour l'implémentation réel sur un traitement en temps réel.

## Conclusion Générale

La méthode de détection du cercle est capable de détecter des formes prédéfinis sur des fonds aléatoires. Pour chaque pixel de l'image, le cumul de la valeur du contour est calculé. Cette technique a donné un taux de réussite (le nombre de cercles détectés) élevé mais souffre principalement du temps de calcul important, ce qui est un problème pour l'implémentation en temps réel.

Dans ce mémoire les paramètres de la transformation de Hough ont été variés pour améliorer les caractéristiques de cette technique. A savoir, le pas de parcours de l'angle et l'intervalle de diamètre.

Cette étude a donné des résultats très intéressants pour le cas de cercle, vu qu'il a seulement le centre comme paramètre, mais peut être étendue aux autres formes comme l'ellipse et le carré mais un nouveau problème sera posé qui est l'angle d'orientation de la forme.

Les problèmes survenus pendant l'étude sont :

- Le temps de calcul important qui pose un grand problème pour l'utilisation temps réel.
- La sensibilité à l'orientation de la forme à détecter ce qui pose un problème pour le temps de calcul, vu qu'il faut parcourir l'image et faire varier l'angle de rotation.
- La sensibilité à la technique de détection de contour.

## Bibliographie

- [1] S. Bres, J.M. Jolion, F. Lebourgeois « *Traitement et analyse des images numériques* » HermesLavoisier 2003
- [2] E. LePennec, «Compression d'image,» 2006.
- [3] «Images numériques Caractéristiques, formats et destinations,» 2010.
- [4] «Images Numériques & Biologie».
- [5] A. Gagalowic, «de cours dispensés à l'ESIEA,» 2002.
- [6] J.-P. Delahaye, «Mona Lisa au photomaton,» 2013.
- [7] D. Müller, «Cacher un texte dans une image».
- [8] A. S. Aguado, M. E. Montiel et M. S. Nixon, «On using directional information for parameter space decomposition in ellipse detection,» Pattern Recognition, 1996.
- [9] E. R. Davies, «Finding ellipses using the generalized Hough transform,» Pattern Recognition, 1989.
- [10] C. Ho et C. L., «A fast ellipse/circle detector using geometric symmetry,» Pattern Recognition, 1995.
- [11] J. Qiang et R. M. Haralick, «A Statistically Efficient Method for Ellipse Detection,» IEEE, 1999.
- [12] Y. Lei et K. C. Wong, «Ellipse detection based on symmetry,» Pattern Recognition, 1999.
- [13] T. S. et F. Matsumoto, «Detection of ellipses by a modified Hough transform,» IEEE, 1978.
- [14] P. Yin et C. L., «New method for ellipse detection by means for symmetry,» Journal of Electronic Imaging, 1994.

- [15] R. K. K. Yip, P. K. S. Yip et D. N. K. Leung, «Modification of Hough transform for circles and ellipses detection using a 2-dimensional array,» Pattern Recognition, 1992.
- [16] Delahaye Jean-Paul, « Mona Lisa au photomaton » — *Images des Mathématiques*, CNRS, 2013
- [17] Müller Didier, « Cacher un texte dans une image »,
- [18] Just Kjeldgaard Pedersen, Simon. “Circular Hough Transform.” Aalborg University, Vision, Graphics, and Interactive Systems. November 2007.  
<[http://www.cvmt.dk/education/teaching/e07/MED3/IP/Simon\\_Pedersen\\_CircularHoughTransform.pdf](http://www.cvmt.dk/education/teaching/e07/MED3/IP/Simon_Pedersen_CircularHoughTransform.pdf)>  
(11/15/2010)
- [19] Epitropou, Victor. “Enhanced (Circular) Hough Transform”. DUTH Circle Detection. 2005-2006  
<<http://www.papamarkos.gr/uploaded-files/Papaparkos/demos/GHT/Help/EHT.htm>>  
(11/15/2010)
- [20] MathWorks. “Hough Transform.” 2010.
- [21] Kolas . “Hough transform.” Wikipedia.  
< [http://en.wikipedia.org/wiki/File:Hough\\_transform\\_diagram.png](http://en.wikipedia.org/wiki/File:Hough_transform_diagram.png)> (11/18/2010)