



Université ABBES LAGHROUR Khenchela
Faculté des Sciences et de la Technologie
Département de Génie Industriel
جامعة عباس لغرور خنشلة
كلية العلوم والتكنولوجيا
قسم الهندسة الصناعية



Mémoire de fin de Cycle
Pour l'obtention du diplôme de Master

Filière : Télécommunications
Spécialité : Systèmes des Télécommunications

Détection et Suivi d'Objets en Mouvement dans une Séquence d'Images

Réalisé par :

- *Krim Widad*
- *Ghezal Asma*

Dirigé par : Dr. Laid Chergui

Soutenue devant le Jury :

Mr. CHERGUI Laid	M.C.B	Université de Khenchela	Rapporteur
Mr. KHEZZAR ZAKI	M.A.A	Université de Khenchela	Examineur
M^{me} FRIHA Souad	M.A.A	Université de Khenchela	Présidente

Promotion 2019/2020

Remerciements

Avant tout, nous remercions notre Dieu qui nous a donné le courage et la volonté de réaliser ce modeste travail.

Merci à nos parents pour le soutien et l'encouragement qu'ils nous ont apporté tout au long de nos travaux, nous adressons nos vifs remerciements à notre encadreur Mr. Chergui laid, pour son effort et ses conseils.

Nos remerciements vont également aux membres du jury d'avoir accepté d'évaluer nos travaux. Enfin, Sans oublier de remercier notre ami et notre collègue.

Dédicaces

Je dédie ce modeste travail à ma mère pour sa tendresse et mon père pour sa patience et encouragement et nos frères et nos sœurs, que nous aimons beaucoup.

Nos chers amis et enseignants. Tous ceux qui ont collaboré de près ou de loin à la réalisation de ce travail.

KrimWidad



Dédicace

*D'abord, je remercie mon dieu pour la force et le courage qu'il m'a donnés
pour élaborer ce modeste travail.*

Je dédie ce travail :

A mon père Lazhar et ma mère Sonia

A ma sœur Yasmine et mon frère abderrezak

A tous mes amis sans oublier la promotion télécoms

A mes professeurs

A tous ceux qui m'ont aidé durant ma vie universitaire

A ceux qui j'aime et qui m'aiment

Ghezal Asma

Liste des figures

CHAPITRE I : *Fondements de base du traitement d'images*

Figure. I.1. Histogramme d'une image sombre _____	6
Figure .I. 2. Histogramme d'une image Claire _____	6
Figure I. 3. Une image de niveau de gris et son histogramme _____	6
Figure.I.4. Principe du filtre médian _____	13
Figure.I.5. Exemple d'application du filtre médian. (a) Image Initiale. (b) Image filtrée _____	13
Figure. I. 6. Principe du filtre maximu _____	14
Figure. I. 7. Exemple d'application du filtre maximum (a) : Image Initiale(b) : Image filtrée _____	14
Figure I. 8. Principe du filtre minimum _____	15
Figure.I.9. Exemple d'une segmentation d'image, (a) : image d'origine, (b) : image segmentée _____	16
Figure.I.10. Opérateur de ROBERT (a) : Image Initiale. (b) : Image du contour _____	18
Figure.I.11. Opérateur de PREWITT, (a) : image originale, (b) : image du contour _____	19
Figure. I.12. Opérateur de SOBEL, (a) : image originale, (b) : image du contour _____	20
Figure.I.13. Opérateur de la placien, (a) : image originale, (b) : image du contou _____	.21
Figure. I .14. Représentation compacte d'une vidéo, T représente l'axe temporel _____	24
Figure.I.15. Exemple d'une séquence d'images _____	26

CHAPITRE II : *Méthodes et Techniques utilisées dans la détection D'objets en Mouvements*

Figure. II.1. Segmentation en régions par watershed _____	33
Figure.II.2. Partitions et graphes d'adjacences _____	34
Figure.II.3. Technique de différence avec image de référence, (a) l'image à l'instant t, (b) l'image de référence, (c) l'image de différence _____	35
Figure. II.4. Technique de différence sans image de référence, (a) image précédente, (b) image courante, (c) image de différence _____	37

Figure II.5. Extraction des contours en mouvement par la proposition de Haynes, (a) image précédente, (b) image courante, (c) image de différence, (d) contour d'image courante, (e) contour en mouvement _____	40
Figure. II.6. Schéma synoptique de l'opérateur mcm _____	42
Figure. II.7. Représentation schématique de l'évolution itérative du contour actif Γ vers les bords de l'objet d'intérêt dans l'image à segmenter _____	44
Figure II.8. Exemple de Seuillage d'une image. (a) image initial (b) image seuillé ____	46
Figure II. 9. Allures idéales d'un histogramme. (a) : unimodal, (b): bimodal et (c) multimodal _____	47
Figure. II.10. Décomposition d'une image couleur en composantes R, V, B ainsi que leurs histogrammes _____	53

CHAPITRE III : *Suivi d'Objets en Mouvement*

Figure.III.1. Modèle proposé pour suivi d'objet _____	61
Figure.III.2. L'ensemble des centres de gravité mesurés _____	62
Figure.III.3. Exemple de trajectoire mesurée et estimée _____	63
Figure.III.4. Fonctionnement du filtre de <i>Kalman</i> _____	66
Figure.III.5. (a)Trame à l'instant t, (b) l'objet en mouvement détecté avec son centre de gravité, (c) trajectoire mesurée sur 5 trames, (d) représentation conjointe des trajectoires mesurée et estimée, (e) représentation conjointe des trajectoires mesurée et estimée sur l'image _____	70
Figure.III.6. Gauche : image vidéo originale, droite : Le résultat du module de détection des blobs (des taches) _____	71
Figure.III.7. Module d'analyse blob _____	72
Figure. III.8. Gauche : image vidéo originale, droite : Le résultat du module d'analyse des blobs _____	73

Chapitre IV : *Implémentation Pratique et Résultats des Simulations*

Figure. IV.1. Algorithme proposé _____	75
Figure.VI. 2. L'algorithme du contour actif _____	76

Liste des figures _____

Figure.VI. 3. Image originale _____	77
Figure.VI.4. Image en niveau de gris _____	77
Figure.VI .5. Image du masque _____	78
Figure.VI.6. Image segmentée _____	78
Figure.VI.7. Image binarisée avec la méthode d'Otsu _____	82
Figure.VI.8. Résultats de la détection par segmentation de la circulation _____	83
Figure.VI.9. Représente le résultat après l'utilisation du traitement morphologique __	84
Figure.VI.10. Le processus de détection et de suivi par l'utilisation de filtre de <i>Kalman</i> _____	88
Figure.VI.11. Détection et suivi par le filtre de <i>Kalman</i> _____	90

Sommaire

Sommaire

Liste des figures

Liste des Tableaux

Introduction Général _____ 1

Chapitre I : Fondements de bases du traitement d'image

I.1.Introduction _____ 3

I.2.Notions d'image numérique _____ 3

I.2.1. Définition de l'image _____ 3

I.2.2. Définition d'une image numérique _____ 4

I.2.3. Les différents types de format d'image _____ 4

I.2.3.1. Image couleur RVB _____ 4

I.2.3.2. Image d'intensités _____ 4

I.2.3.3. Image binaire _____ 4

I.2.4. Caractéristiques d'une image numérique _____ 5

I.2.4.1. Dimension _____ 5

I.2.4.2. Résolution _____ 5

I.2.4.3. Pixel _____ 5

I.2.4.4. Histogramme _____ 6

I.2.4.5. Luminance _____ 6

I.2.4.6. Contraste _____ 6

I.2.4.7. Bruit _____ 7

I.2.4.8. Contour _____ 7

I.2.4.9. Niveau de gris _____ 7

I.3. Bruit et amélioration d'images par filtrage _____ 8

I.3.1. Le bruit _____ 8

Sommaire

I.3.2. Amélioration du rapport signal sur bruit	9
I.3.3. Amélioration par filtrage	9
I.3.3.1. La convolution	9
I.3.3.2.1. Les Filtres linéaire	10
I.3.3.2.2. Les filtres non linéaires	12
I.3.4. Avantages du filtrage	15
I.4. Segmentation d'images et détection du contour	15
I.4.1. La segmentation d'image	15
I.4.1.1. Classification des pixels	16
I.4.1.2. Segmentation par détection de contour	17
I.4.1.3. Segmentation par région	21
I.4.1.3.1. Segmentation par agrégation de pixels	21
I.4.1.3.2. Segmentation par division	21
I.4.1.3.3. Segmentation par division-fusion (Split & merge)	22
I.4.2. Binarisation	22
I.4.2.1. Opérations sur les images binaires	22
I.4.2.1.1. Dilatation	23
I.4.2.1.2. Erosion	23
I.4.2.2. Avantages de la binarisation	24
I.5. Les vidéos	24
I.5.1. Définition d'une vidéo	24
I.5.2. Représentation d'une vidéo	24
I.5.3. Définition d'un objet vidéo	25
I.5.4. Séquence d'image	25
I.5.5. Nature de la scène	26
I.5.5.1. Séquences à caméra fixe (Scène statique)	26
I.5.5.2. Séquences à caméra mobile (Scène dynamique)	26
I.5.5.3. Mono et multi-caméra	26

Sommaire

I.6. Conclusion	28
-----------------	----

Chapitre II : Méthodes et Techniques utilisées dans la détection D'objets en Mouvements

II.1. Introduction	29
II.2. Méthode de Segmentation	30
II.2.1. Le choix d'une technique de segmentation	30
II.2.2 Objectifs de la segmentation	31
II.3. Détection d'objets en mouvement par méthode de Segmentation	31
II.4. Détection par différence d'images	34
II.4.1. Approche région	35
II.4.2. Approche contour	38
II.5. Méthode MCM (la mesure des contours en mouvement)	42
II.6 Segmentation d'image par la méthode du contour actif	44
II.7. Seuil d'image global en utilisant la méthode d'Otsu	46
II.7.1. Définition du seuillage	46
II.7.2. Méthode d'Otsu	48
II.7.3. Seuils d'image multiniveaux utilisant la méthode d'Otsu	49
II.8. Seuillage d'une image couleur	51
II.9. Conclusion	56

Chapitre III : Suivi d'Objets en Mouvement

III.1. Introduction	57
III.2. Méthodes de suivi	57
III.2.1. Suivi par appariement de détections	57
III.2.2. Suivi par segmentation dynamique	58

Sommaire

III.2.3. Suivi déterministe de fenêtre englobante par détection séquentielle	58
III.3. Modèle Proposé pour suivi d'objet	58
III.4. Le filtre kalman Caractéristiques d'un bon algorithme de suivi	61
III.4.1. Définition du filtre kalman	61
III.4.2. Principe du filtre de Kalman	61
III.4.3. Algorithme de Filtre de Kalman	63
III.5. Méthode de travail du filtre dans le suivi	64
III.6. Exemple d'applications du filtre de kalman	65
III.7. Détection des Blobs	66
III.8. L'analyse de <i>Blob</i>	66
III.9. Conclusion	68

Chapitre IV : Implémentation Pratique et Résultats des Simulations

VI.1. Introduction	69
VI.2. Algorithme proposé pour la détection et le suivi d'objet dans une séquence d'images	69
VI.2.1. Segmentation d'image	70
VI.2.1.1. Segmentation d'images par la méthode du contour actif	76
IV.3. Détection d'objet par segmentation	77
VI.3.1. Segmentation de la circulation	77
VI.4. Le suivi d'objet par le filtre kalman	82
VI.4.1. Détection et suivie d'une balle par filtrage de Kalman, détection du premier plan et l'analyse de Blob	82
VI.5. Conclusion	85
Conclusion générale	86

Introduction Générale

Avec la généralisation de l'utilisation des images numériques, l'analyse du mouvement dans les vidéos s'est révélée être un outil indispensable pour des applications aussi diverses que la vidéo surveillance, la compression vidéo, l'imagerie médicale, la robotique, l'interaction homme machine, ...etc.

En effet, les zones de mouvement d'une séquence d'images correspondent souvent à des événements sur lesquels un système de vision doit se focaliser.

L'analyse du mouvement est un vaste sujet qui englobe un certain nombre de problématiques :

- La détection des objets en mouvement, c'est-à-dire la détection d'un ensemble de régions d'intérêt en mouvement dans la scène observée.
- Le suivi des objets ou de régions, dont le but est de déterminer la position de chaque objet ou région dans l'image à chaque instant.

La détection et suivi d'objets temps réel est une problématique difficile qui se pose dans un grand nombre d'applications de traitement d'images comme l'interaction homme-machine, la surveillance civile et militaire, la réalité virtuelle, l'analyse du mouvement humain ou encore la compression d'images.

Cette difficulté est accentuée dans les environnements sans contraintes où le système de suivi devra s'adapter à la variabilité importante des objets, aux variations de luminosité, aux occlusions (partielles ou totales) ainsi qu'aux problèmes de détection de mouvement.

L'exploitation des détections pour un grand nombre d'applications (surveillance, analyse de séquences sportives, compression ...) Nécessite l'ajout de cette cohérence temporelle. Nous avons donc, dans un deuxième temps, proposé une méthode de suivi.

Le problème de suivi temporel en vision numérique est de générer une inférence au sujet du mouvement perçu d'un objet dans une séquence d'images successives. Les données connues pour résoudre le problème consistent en un modèle de déplacement de l'objet à suivre et certaines mesures à partir d'une séquence d'images.

Ces mesures sont le plus souvent des points caractéristiques ou des régions, mais à peu près n'importe quoi de mesurable dans l'image peut être utilisé.

Ces mesures ne sont pas nécessairement bonnes, en effet, du bruit dans l'image peut créer de faux points caractéristiques [1].

Nous décomposons ce travail en quatre chapitres :

Chapitre 1 : Dans le premier chapitre, on a abordé certains concepts primaires concernant le traitement d'images

Chapitre 2 : Le deuxième chapitre présente la phase de détection et les méthodes utilisées de détection du mouvement.

Chapitre 3 : le troisième chapitre présente la 2ème phase : le suivi et les algorithmes utilisés de suivi.

Chapitre 4 : conclut ce mémoire en présentant l'ensemble des résultats expérimentaux obtenus pour la détection et le suivi d'objets en mouvement

Une conclusion synthétisera les contributions et les résultats obtenus et proposera des perspectives à nos travaux.

Chapitre I :
Fondements de Base
du Traitement d'Images

I.1. Introduction [2]

Depuis longtemps, le public sait qu'une image vaut mieux que mille mots, l'image constitue un moyen essentiel dans la communication homme-machine (Une vidéo vaut mille phrases).

De ce fait, le traitement d'image est devenu une discipline nécessaire, mettant en place un ensemble de techniques permettant d'extraire des informations, de modifier une image numérique (dans le but de l'améliorer), et d'automatiser son traitement.

Le domaine de traitement d'image a fortement évolué, et ses techniques sont actuellement utilisées pour résoudre une variété de problèmes pour lesquels on adapte des solutions selon la nature, les situations et les objectifs à atteindre.

Dans ce chapitre, nous abordons les notions de base sur le traitement d'images.

I.2. Notions d'image numérique

I.2.1. Définition de l'image

Une image est une représentation planaire d'une scène ou d'un objet situé en général dans un espace tridimensionnel, elle est issue du contact des rayons lumineux provenant des objets formants la scène avec un capteur (caméra, scanner, rayons X, ...).

Il ne s'agit en réalité que d'une représentation spatiale de la lumière. L'image est considérée comme un ensemble de points auquel est affectée une grandeur physique (luminance, couleur). Ces grandeurs peuvent être continues (image analogique) ou bien discrètes (images digitales).

Mathématiquement, l'image représente une fonction continue IF , appelée fonction image, de deux variables spatiales représentée par $IF(x,y)$ mesurant la nuance du niveau de gris de l'image aux coordonnées (x,y) . La fonction Image peut se représenter sous la forme suivante :

$IF: R^2 \rightarrow R$ Avec R : ensemble des réelles.

$(x, y) \rightarrow IF(x, y)$ x Et y : deux variables réelles.

I.2.2. Définition d'une image numérique[3, 4]

Une image numérique est représentée par une matrice chaque élément de la matrice est un pixel, Le pixel est l'abréviation du mot « Picture élément », représente l'intensité lumineuse comprise entre 0 et 255 c'est-à-dire 256 niveaux de gris (0=noir,1=blanc).

Une image numérique est une image échantillonnée et quantifiée.

Exemples de couleurs :

<p>(0,0,0) = Noir, (255,0,0) = Rouge, (0,255,0) = Vert, (0,0,255) = Bleu, (127,127,127) = Gris moyen, (255,255,255) = Blanc.</p>
--

I.2.3. Les différents types de format d'image [2]

I.2.3.1. Image couleur RVB

L'œil humain analyse la couleur à l'aide de trois types de cellules photo réceptrices 'les cônes'. Ces cellules sont sensibles aux basses, moyennes, ou hautes fréquences (rouge, vert, bleu). Pour représenter la couleur d'un pixel, il faut donc donner trois nombres, qui correspondent au dosage de trois couleurs de base : Rouge, Vert, Bleu.

On peut ainsi représenter une image couleur par trois matrices chacune correspondant à une couleur de base.

I.2.3.2. Image d'intensités

C'est une matrice dans laquelle chaque élément est compris entre 0 (noir) et 1 (blanc).

I.2.3.3. Image binaire

Une image binaire est une matrice rectangulaire dans l'élément valent 0 ou 1.

I.2.4. Caractéristiques d'une image numérique

L'image est un ensemble structuré d'informations caractérisé par les paramètres suivants :

I.2.4.1. Dimension

C'est la taille de l'image. Cette dernière se présente sous forme de matrice dont les éléments sont des valeurs numériques représentatives des intensités lumineuses (pixels).

Le nombre de ligne de cette matrice multipliée par le nombre de colonnes nous donne le nombre total de pixels dans une image.

I.2.4.2. Résolution

C'est la clarté ou la finesse de détails atteinte par un moniteur ou une imprimante dans la production d'images. Sur les moniteurs d'ordinateurs, la résolution est exprimée en nombre de pixels par unité de mesure (pouce ou centimètre).

On utilise aussi le mot résolution pour désigner le nombre total de pixels affichables horizontalement, ou verticalement sur un moniteur ; plus grand est ce nombre, meilleure est la résolution.

I.2.4.3. Pixel

Est une unité de surface permettant de définir la base d'une image numérique. Il matérialise un point donné (x, y) du plan de l'image. L'information

représentée par le pixel est le niveau de gris (ou la couleur) prélevée à l'emplacement correspondant dans l'image réelle.

La différence entre image monochrome et image couleur réside dans la quantité d'informations contenue dans chaque pixel, par exemple dans une image couleur (RVB : Rouge, Vert, Bleu) la valeur d'un pixel est représentée sur trois octets pour chaque couleur.

I.2.4.4. Histogramme[5]

L'histogramme des niveaux de gris ou des couleurs d'une image est une fonction qui donne la fréquence d'apparition de chaque niveau de gris (couleur) dans l'image.

Il permet de donner un grand nombre d'information sur la distribution des niveaux de gris (couleur) et de voir entre quelles bornes est repartie la majorité des niveaux de gris dans le cas d'une image trop claire ou d'une image trop foncée.

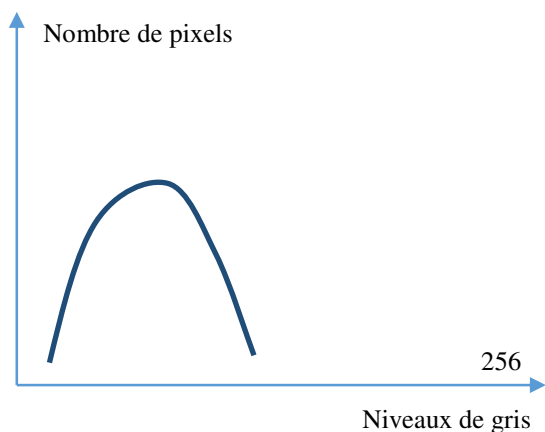


Figure.I.1. Histogramme d'une image sombre.

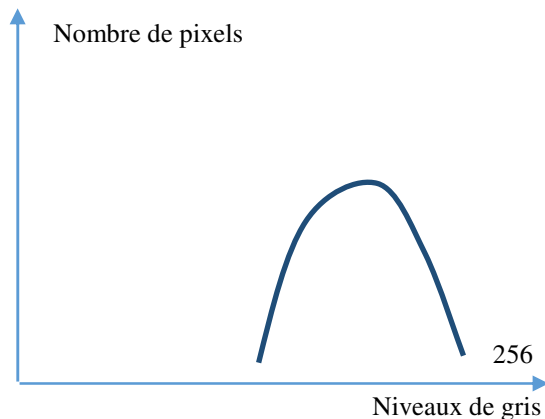


Figure.I.2. Histogramme d'une image claire.

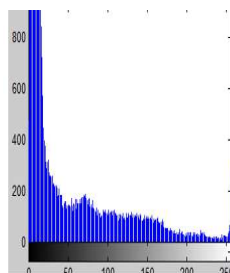


Figure .I. 3. Une image de niveau de gris et son histogramm

I.2.4.5. Luminance

C'est le degré de luminosité des points de l'image. Elle est définie aussi comme étant le quotient de l'intensité lumineuse d'une surface par l'aire apparente de cette surface, pour un observateur lointain, le mot luminance est substitué au mot brillance, qui correspond à l'éclat d'un objet.

Une bonne luminance se caractérise par :

- Des images lumineuses (brillantes) ;
- Un bon contraste (il faut éviter les images où la gamme de contraste tend vers le blanc ou le noir, ces images entraînent des pertes de détails dans les zones sombres ou lumineuses) ;
- L'absence de parasites.

I.2.4.6. Contraste

C'est l'opposition marquée entre deux régions d'une image. Une image contrastée présente une bonne dynamique de la distribution des valeurs de gris sur tout l'intervalle des valeurs possibles, avec des blancs bien clairs et des noirs profonds.

Au contraire une image peu contrastée a une faible dynamique, la plupart des pixels ayant des valeurs de gris très proches. Si L_1 et L_2 sont les degrés de luminosité respectivement de deux zones voisines A1 et A2 d'une image, le contraste est défini par le rapport :

$$C = \frac{(L_1 - L_2)}{(L_1 + L_2)} \quad (\text{I. 1})$$

I.2.4.7. Bruit

Un bruit (parasite) dans une image est considéré comme un phénomène de brusque variation de l'intensité d'un pixel par rapport à ses voisins, il provient de l'éclairage des dispositifs optiques et électroniques du capteur.

I.2.4.8. Contour

Les contours représentent la frontière entre les objets de l'image, ou la limite entre deux pixels dont les niveaux de gris représentant une différence significative.

I.2.4.9. Niveau de gris[6]

C'est la valeur d'intensité lumineuse d'un pixel. Cette valeur peut aller du noir (0) jusqu'au blanc 255 en passant par les nuances qui sont contenues dans l'intervalle (0, 255).

Elle correspond en fait à la quantité de la lumière réfléchie. Pour 8 bits, on dispose de 256 niveaux de gris. Plus le nombre de bit est grand plus les niveaux sont nombreux et plus la représentation est fidèle.

I.3. Bruit et amélioration d'images par filtrage

I.3.1. Le bruit[5]

Le bruit en image ou vidéo numériques est un défaut parasite dégradant la qualité de l'image. Dans le domaine du traitement du signal et de l'image, le bruit correspond à un phénomène aléatoire qui se surajoute à l'image idéale.

Probablement la meilleure approche pour comprendre le bruit est de réaliser que si l'on acquiert plusieurs fois l'image d'un même objet, immobile et inchangé, on n'observera pas exactement le même résultat.

La différence est liée au bruit. Exemple, en lançant de la même manière plusieurs fois un dé, on n'obtient pas le même résultat ; c'est aléatoire. Comme tout phénomène aléatoire, le bruit est caractérisé par son écart type (racine carrée de la variance), qu'on note σ .

Pour quantifier le niveau de bruit, on compare généralement l'intensité du signal avec L'écart-type du bruit. Pour déterminer le rapport signal-sur-bruit (RSB) on applique :

$$\text{RSB} = \frac{S}{\sigma} \quad (\text{I.2})$$

S :Représente l'intensité du signal et σ représente l'écart-type.

I.3.2. Amélioration du rapport signal sur bruit

La méthode la plus simple pour augmenter le rapport signal/bruit consiste à appliquer le principe des analyseurs multicanaux, c'est-à-dire effectuer plusieurs acquisitions de l'image (n par exemple) : ce qui revient à effectuer plusieurs sommations du signal.

Le bruit n'apparaissant statistiquement jamais au même endroit, sera uniformément réparti, alors que le signal, apparaissant toujours au même endroit, sera amplifié d'un facteur racine carrée de n.

I.3.3. Amélioration par filtrage

L'amélioration du rapport signal/bruit, bien qu'intéressante, est souvent insuffisante pour obtenir de bonnes images.

L'amélioration de l'image est essentiellement obtenue par ce que l'on appelle une opération de filtrage.

Le filtrage est un traitement local utilisé principalement pour réaliser une analyse spatiale de l'image, son objectif est d'accentuer les variations d'intensité de l'image, ou de détecter les contours ou de réduire les bruits existants.

Il existe un grand nombre de filtres possibles, et à quelques exceptions près, on peut les classer en deux grandes catégories :

- Les filtres linéaires et filtres non linéaires.

I.3.3.1.La convolution [7]

Pour mettre en œuvre un filtrage avec des filtres linéaires, on utilise un opérateur mathématique nommé convolution.

La convolution consiste donc à prendre un voisinage du pixel que l'on veut modifier, le plus souvent ce voisinage est compris dans une fenêtre carrée de côté de longueur impaire, de multiplier chaque valeur du voisinage par des coefficients appartenant à un masque de convolution, d'en faire la somme qui deviendra la nouvelle valeur du pixel.

Ce traitement est effectué pour chaque pixel de l'image, mis à part les bords de l'image pour lesquels, soit on met les pixels à 0, soit on modifie le masque et le diviseur.

La plus simple approche est d'appliquer l'opération seulement sur une sous image, choisie telle que pour (x, y) dans la sous image, il existe des voisins dans l'image.

Ceci donne des résultats ayant un vrai sens, mais on note néanmoins que l'image en sortie produit par l'opération est plus petite que l'image en entrée.

$$I(x, y) = I(x, y)W(x, y) = \sum_{m=-d}^d \sum_{n=-d}^d W(m, n)I(x + m, y + n) \quad (I. 3)$$

Où la taille du masque de convolution est $(2d+1) * (2d+1)$.

I.3.3.2.1. Les Filtres linéaire[8]

L'utilisation des filtres linéaires provient de l'extension des méthodes mises au point pour le traitement du signal, au traitement des images numériques.

Nous pouvons, en effet, considérer une image numérique comme un signal bidimensionnel numérisé.

Un filtre est linéaire s'il peut s'écrire de la manière suivante :

Soit F le filtre utilisé, I_1 et I_2 représentant deux images alors :

$$F * (I_1 + I_2) = F * I_1 + F * I_2 \text{ (I. 4)}$$

L'opération(*) représente le produit de convolution.

Les filtres linéaires les plus connus sont les filtres passe-bas, passe-haut. Ces filtres sont représentés sous forme de masque auquel est associé un coefficient diviseur qui permet de ramener les valeurs calculées entre 0 et 255 pour une image à 256 niveaux de gris.

a. Filtre passe-bas

L'intensité du pixel considéré est remplacée par la moyenne des pixels de son voisinage. La taille de la zone (fenêtre) entourant le pixel est un paramètre important, plus cette dimension est grande, plus sa sensibilité au bruit diminue, et le lissage devient important (le flou s'accroît).

Le filtre passe-bas laisse passer les basses fréquences (les faibles changements d'intensité de l'image) et atténue les hautes fréquences (variations rapides).

Exemple de filtre passe-bas :

1/9 *Masque du filtre

1	1	1
1	1	1
1	1	1

b. Filtre passe-haut[7]

Il est utilisé pour amplifier les détails de hautes fréquences. Il peut permettre par exemple de restaurer des images qui ont été délocalisées et d'accentuer les contours en faisant ressortir les pixels compris entre des zones homogènes.

Il met en évidence les changements rapides de l'intensité de l'image (les hautes fréquences) et laisse les zones uniformes inchangées (basses fréquences).

Exemple de filtre passe-haut :

$$\mathbf{H} = \begin{array}{|c|c|c|} \hline -1 & -1 & -1 \\ \hline -1 & 9 & -1 \\ \hline -1 & -1 & -1 \\ \hline \end{array} \text{ Masque du Filtre}$$

c. **Filtre Gaussien**[2]

L'expression gaussienne en deux dimensions est donnée par :

$$G_0(x, y) = \frac{1}{2\pi\sigma^2} * \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (I. 5)$$

L'intérêt de ce filtre est que l'on contrôle facilement le degré de filtrage à travers le paramètre σ . Le filtre gaussien donne plus d'importance aux pixels proches du pixel central, et diminue cette importance au fur et à mesure que l'on s'éloigne de celui-ci, il a les mêmes inconvénients que le filtre passe-bas, c'est-à-dire il dégrade les contours.

La discrétisation de ce filtre pour un σ égale à 0.6 donne le masque suivant :

$$(1/16) \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array} \text{ Masque du Filtre}$$

I.3.3.2.2. Les filtres non linéaires

Ils sont conçus pour régler les problèmes des filtres linéaires, surtout pour ce qui concerne la mauvaise conservation des contours. Leur principe est le même que celui des filtres linéaires, il s'agit toujours de remplacer la valeur de chaque pixel par la valeur d'une fonction calculée dans son voisinage.

La différence majeure, est que cette fonction n'est plus linéaire mais une fonction quelconque (elle peut inclure des opérateurs de comparaisons ou de classification). Les filtres non linéaires les plus connus sont :

- Filtre médian ;
- Filtre maximum ;
- Filtre minimum.

a. Le filtre Médian

Ce filtre est très utilisé pour éliminer le bruit sur une image qui peut être de différentes origines (poussières, petits nuages, baisse momentanée de l'intensité électrique sur les capteurs, ...) et qui se traduit par des taches de faible dimension dont la distribution sur l'image est aléatoire.

L'avantage de ce filtre réside dans le fait qu'il conserve les contours alors que les autres types de filtres ont tendance à les adoucir. Sur un voisinage à huit, le nouveau niveau de gris du pixel central est choisi comme étant la valeur médiane de tous les pixels de la fenêtre d'analyse, centrée sur ce dernier. Son avantage est qu'il garde la netteté des éléments qui constituent l'image sans étaler les transitions.

L'algorithme de filtre médian est le suivant :

1. Trier les valeurs par ordre croissant ;
2. Remplacer la valeur du pixel centrale par la valeur située au milieu de la triée ;
3. Répéter cette opération pour tous les pixels de l'image.

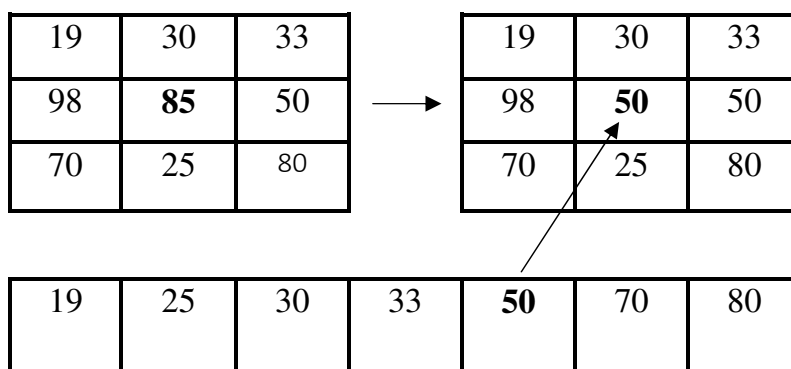


Figure. I.4. Principe du filtre médian.

Exemple :



Figure.I.5. Exemple d'application du filtre médian. (a) Image Initiale. (b) Image filtrée.

b. Filtre maximum

On applique le même traitement que celui du filtre médian mais la valeur du pixel du centre comme la montre la figure I.6 va être changée par le maximum.

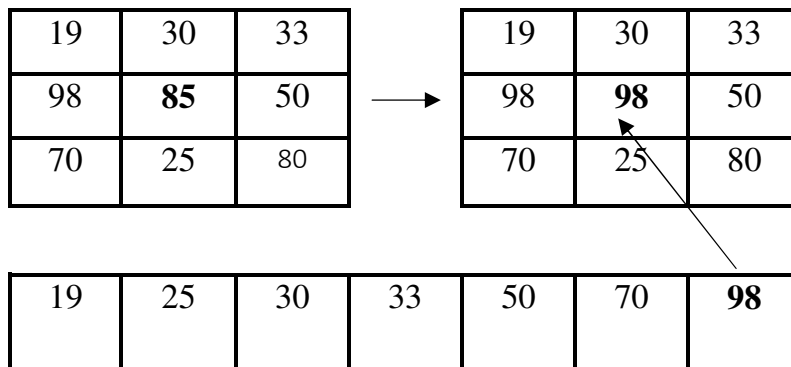


Figure. I.6. Principe du filtre maximum.

Exemple :

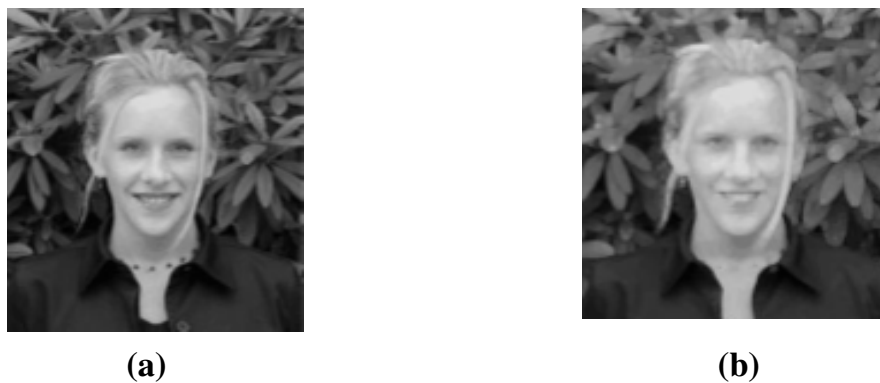


Figure.I.7. Exemple d'application du filtre maximum (a) : Image Initiale(b) : Image filtrée.

c. Filtre minimum

On applique le même traitement que celui du filtre maximum mais, cette fois, la valeur du pixel du centre comme la montre la figure va être remplacée par le minimum.

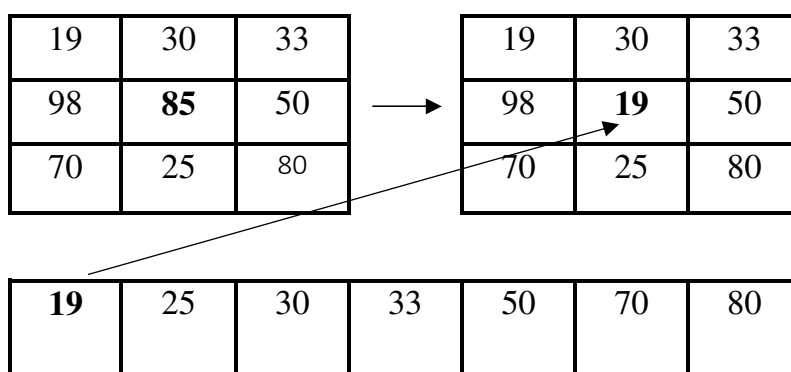


Figure. I.8. Principe du filtre minimum.

I.3.4. Avantages du filtrage [9,10]

- Réduction de bruit ;
- Accentuer les variations d'intensité de l'image ;
- Une technique importante d'amélioration de la qualité de l'image numérique ;
- Détection de contours et préservation de la forme des régions ;
- Renforcement de l'homogénéité à l'intérieur des régions.

Donc le résultat obtenu suite à un filtrage dépend énormément de la qualité du signal de l'image d'origine. Si le signal est très détérioré, le filtrage seul sera très peu efficace.

I.4. Segmentation d'images et détection du contour

I.4.1. La segmentation d'image [11]

La segmentation est une branche du traitement d'image qui s'occupe de l'analyse d'image ou l'analyse de la scène. Généralement, cette partie reçoit en

entrée une image et délivre en sortie une description de cette image (en termes de contours ou de région).

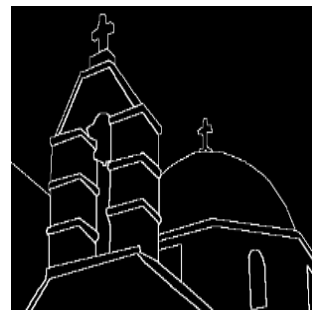
Les images sont composées de régions possédant des propriétés locales qui peuvent être la répartition des niveaux de gris.

En groupant des points qui possèdent une même propriété donnée, on obtient des régions uniformes. Cette opération s'appelle « segmentation ».

"La segmentation est un traitement de bas niveau qui consiste à créer une partition de l'image A en sous-ensembles R_i , appelés régions tels qu'aucune région ne soit vide, l'intersection entre deux régions soit vide et l'ensemble des régions recouvre toute l'image [12].



(a)



(b)

Figure. I. 9. Exemple d'une segmentation d'image, (a) : image d'origine, (b) ; image segmentée [13].

Une région est un ensemble de pixels connexes ayant des propriétés communes qui les différencient des pixels des régions voisines [12].

Il existe plusieurs méthodes pour la segmentation d'images [8] :

- Classification des pixels ;
- Segmentation par détection de contour ;
- Segmentation par les régions.

I.4.1.1. Classification des pixels

Ce type de méthode prend en considération uniquement les valeurs de niveaux de gris d'un point pour décider son appartenance à un objet ou pas. Pour pouvoir le faire, nous devons trouver le domaine de niveaux de gris qui caractérisent l'objet à segmenter.

Pour cela, une analyse de l'image et de son histogramme est nécessaire avant de faire le choix en ce qui concerne l'intervalle. Puis, une simple opération de seuillage utilisant les bornes de l'intervalle comme seuils est à effectuer.

I.4.1.2. Segmentation par détection de contour

Elle a pour but la détection de discontinuités présentes dans une image, c'est-à-dire les frontières qui existent entre les régions homogènes. Pour cela on effectue des dérivations, car les contours se coïncident avec les points forts du gradient (première dérivée) ou nuls de la dérivée seconde. Nous appellerons :

- Points de contour : tous points appartenant à la frontière d'une région ;
- Contour d'une région : la courbe fermée telle que tous les pixels de la région se trouvent dans un même côté de cette courbe.

Nous citons, dans ce qui suit, quelques opérations sur les contours :

a. Réduction des contours

Elle agit de manière que les contours mis en évidences ne fassent qu'un seul pixel d'épaisseur. Lorsqu'on effectue le gradient d'une image, il est nécessaire d'isoler les maxima locaux de l'image dérivée pour déterminer les points exacts de contours afin de réduire ceci à une courbe d'un seul pixel d'épaisseur.

Mathématiquement, une image est représentée par une fonction f :

$$f: I \in \mathbb{R}^2 \rightarrow I \in \mathbb{R} \text{ , } P(x, y) \rightarrow f(x, y) \text{ (I.6)}$$

Donc le gradient de la fonction f est donné par :

$$\text{Amplitude} = \sqrt{\left(\left(\frac{df}{dx}\right)^2 + \left(\frac{df}{dy}\right)^2\right)} \quad \text{Direction} = \left(\left(\frac{df}{dx}\right) / \left(\frac{df}{dy}\right)\right) \quad (I. 7)$$

Dans le cas discret (image digitalisée), ces dérivées sont remplacées par des différences. Par ailleurs on approxime l'amplitude en utilisant une somme ou le maximum des valeurs absolues, plutôt que la racine carrée préconisée.

a. Opérateur de ROBERT

Cet opérateur est très sensible aux micros contours, mais aussi aux bruits ; il donne des contours fins mais incomplets, il est donc appliqué aux régions à structure très fine. La qualité des contours est assez bonne.

En effet, l'opérateur de ROBERT n'est qu'une estimation des dérivées premières par les expressions et données avec la norme et l'orientation :

$$|\nabla f(x, y)| = R(x, y)|\nabla f(x, y)| = \sqrt{(\Delta_1^2 + \Delta_2^2)} \quad (I. 8)$$

Ceci revient à convoluer l'image avec les deux masques suivant :

$$H_1 = \begin{array}{|c|c|} \hline 1 & 0 \\ \hline 0 & -1 \\ \hline \end{array}$$

$$H_2 = \begin{array}{|c|c|} \hline 0 & 1 \\ \hline -1 & 0 \\ \hline \end{array}$$

Exemple :



(a)



(b)

Figure.I.10. Opérateur de ROBERT (a) : Image Initiale. (b) : Image du contour.

b. Opérateur de PREWITT

Contrairement à l'opérateur de ROBERT, cet opérateur présente l'avantage d'être moins sensible aux bruits ; il favorise les contours à structure linéaire mais la qualité des contours est meilleure.

Les dérivées directionnelles horizontale et verticale s'expriment sous la forme :

$$A_j[i,j] = h_j * A[i,j] \text{ et } A_i[i,j] = h_i * A[i,j] \text{ (I. 9)}$$

$$\text{Avec } h_j = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} \text{ et } h_i = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

Les matrices h_i et h_j , appelées aussi masques, sont les noyaux de convolution de filtres à réponse impulsionnelle finie.

Donc la norme et l'orientation du gradient sont :

$$|\nabla A[i,j]| = \sqrt{(A_j^2[i,j] + A_i^2[i,j])} \text{ et } \theta = \frac{\pi}{2} - \arctan\left(\frac{A_i}{A_j}\right) \text{ (I. 10)}$$



(a)



(b)

Figure.I.11. Opérateur de PREWITT, (a) : image originale, (b) : image du contour.

c. Opérateur de SOBEL

Très sensible aux bruits, il est utilisé pour la détection de contours. Les contours correspondent en général à des changements brusques de propriétés physiques ou géométriques de la scène.

Ce filtre comporte deux masques de convolution, un pour la détection des contours verticaux et un autre pour la détection des contours horizontaux. On obtient après convolution une image des gradients.

Cette image pourra par la suite être modifiée dans une phase de seuillage pour ne contenir que des pixels à 1 ou 0 correspondant respectivement à un point de contour ou à un non point de contour. Les dérivées directionnelles horizontale et verticale s'expriment sous la forme :

$$A_j[i, j] = h_j * A[i, j] \text{ et } A_i[i, j] = h_i * A[i, j] \text{ (I. 11)}$$

Avec
$$h_j = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \text{ et } h_i = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \text{ Exemple :}$$



(a)



(b)

Figure. I. 12. Opérateur du sobel, (a) : image originale, (b) : image du contour.

d. Opérateur du LAPLACIEN

Il approxime de façon très simple la dérivée seconde de l'image. Il est souvent utilisé pour déterminer dans quel côté se trouve un pixel au voisinage d'un

bord. En effet, la dérivée seconde est nulle sur ce bord, et prend des signes opposés d'autre part.

C'est un opérateur différentiel à coefficients constants, invariants par rotation. Cet opérateur estime que les points de contours représentent les points où la deuxième dérivée est égale à zéro. Donc l'opérateur LAPLACIEN :

$$\nabla^2 f = \frac{d^2 f}{dx^2} + \frac{d^2 f}{dy^2} \quad (I.12)$$

Peut être exprimé seulement par le masque suivant :

$$L = \begin{bmatrix} 0 & 1 & -1 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Et la détection se fait par le produit de convolution entre ce masque et l'image originale

Exemple :



Figure.I.13. Opérateur du placien , (a) : image originale, (b) : image du contour.

I.4.1.3. Segmentation par région

Il est très difficile de mettre au point un algorithme de segmentation qui fonctionne correctement dans toutes les situations comme le fait aussi bien le

système visuel humain. Nous allons dans ce qui suit exposer quelques techniques de détection des zones homogènes de l'image.

I.4.1.3.1. Segmentation par agrégation de pixels

Au départ, la méthode considère chaque pixel comme étant une région, et de regrouper ces régions ensuite selon un double critère : la similarité de niveaux de gris et la connexité.

Le critère de similarité peut être par exemple: la variance des niveaux de gris de la région R (et qui doit être inférieure à un certain seuil).

L'idée de cet algorithme est de faire croître² un pixel tant que les éléments de son voisinage vérifient le critère de similarité, sinon on passe à un nouveau pixel et on itère le processus.

I.4.1.3.2. Segmentation par division

Contrairement à la précédente, celle-ci suppose au départ que tous les pixels forment la même région. Si la région n'est pas homogène (au sens du critère de similarité) on la divise en plusieurs sous régions (généralement quatre) et on réapplique l'algorithme, sinon le processus se termine.

On peut rajouter à ceci un critère d'arrêt sur la taille des régions.

I.4.1.3.3. Segmentation par division-fusion (Split & merge)[14]

Cette méthode passe par deux étapes :

- Split : Elle divise l'image en régions (la méthode précédente) et constitue enfin un quad-tree et duquel on va déduire un graphe d'adjacence entre les régions (qui sont les feuilles de l'arbre) ;
- Merge : Elle rassemble les feuilles du quad-tree qui sont voisines (dans le graphe d'adjacence) et similaires (respectant le critère d'homogénéité) pour avoir de nouvelles régions.

I.4.2. Binarisation[8,2]

Les traitements les plus connus sur les images binaires sont ceux de la morphologie mathématique, mais leur application sur les images en niveaux de gris est également possible. Pour avoir des images binaires à partir d'une image grise, on applique les techniques de seuillage (les valeurs supérieures au seuil seront mises à un (1), et les valeurs inférieures au seuil, à zéro (0)).

I.4.2.1. Opérations sur les images binaires

Il existe des méthodes qui ont pour objectif l'amélioration des images binaires, parmi celles-ci nous pouvons citer :

I.4.2.1.1. Dilatation

Elle consiste à dilater l'image, de ce fait les points noirs isolés au milieu des parties blanches sont "mangés" par la dilatation des parties blanches. Ceci veut dire, en d'autres termes, que la dilatation sert à éliminer les points noirs de l'image.

Le principe de la dilatation est de parcourir l'image avec une fenêtre carrée de taille $N \times N$ (N étant un nombre aléatoire de pixels), et d'effectuer pour chaque pixel de l'image le OU logique des $(N - 1)$ (voisins, c'est-à-dire tous les pixels de la fenêtre sauf le pixel central).

Le résultat de cette fonction logique sera : Si le OU des voisins vaut 1, le pixel courant est forcé à 1 dans l'image résultat. Si le OU des voisins vaut 0, le pixel courant est simplement recopié de l'image initiale vers l'image résultat.

I.4.2.1.2. Erosion

Elle permet d'éliminer les points blancs de l'image ; le principe est dual à la dilatation, cette fois-ci, on effectue le ET logique de $(N - 1)$ pixels voisins ; comme résultat à cette application, l'image est érodée et tout pixel blanc isolé tend à disparaître.

❖ Opérations en cascade [15]

À partir des deux transformations de base, on peut définir deux nouvelles opérations qui sont l'ouverture et la fermeture.

- **Ouverture** : c'est une érosion suivie d'une dilatation, elle permet de supprimer certains bruits qui existent dans le fond d'image, et d'isoler certaines surfaces de l'objet.
- **Fermeture** : c'est l'opération inverse de l'ouverture, c'est-à-dire que l'on applique d'abord la dilatation ensuite une érosion, elle permet de recoller certains morceaux de surfaces de l'objet et de supprimer les trous à l'intérieur de l'objet.

I.4.2.2. Avantage de la binarisation

- La binarisation d'une image présente plusieurs avantages dont l'un des plus importants est certainement le faible espace mémoire nécessaire pour cette image binaire et la simplicité des opérations qui lui sont associés ;
- Elle peut être aussi la première étape d'isolement des objets par rapport au fond. De plus, la binarisation constitue une première méthode de la segmentation de l'image.

I.5. Les vidéos

I .5.1 Définition d'une vidéo [16]

La vidéo est une succession d'images animées défilant à une certaine cadence afin de créer une illusion de mouvement pour l'œil humain.

Elle peut être analogique (signal continu d'intensité de luminance) ou numérique (suite de trames ou images).

I .5.2 Représentation d'une vidéo [17]

Une vidéo peut être considérée comme une séquence d'images dont chacune contient une vue statique des scènes qui s'y déroulent. Ces images, ou trames, sont prises dans des intervalles réguliers et très courts (30 images par seconde).

Les trames sont des images à deux dimensions $X = (x, y)$ et l'axe temporel t constitue la troisième dimension. La figure I.14 est une illustration de la représentation compacte d'une vidéo avec une superposition des trames qui la composent.

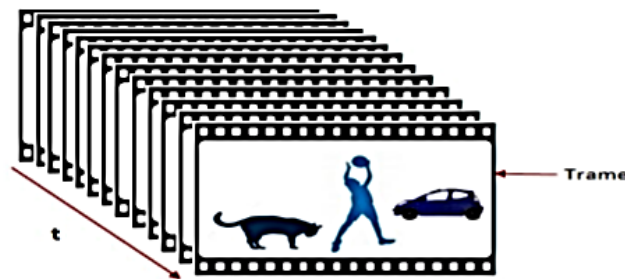


Figure. I .14. Représentation compacte d'une vidéo, T représente l'axe temporel.

Chaque trame de cette vidéo peut donc être utilisée dans les opérations de prétraitement. Ces opérations de prétraitement ont pour but de détecter les objets en mouvement dans la vidéo

I.5.3 Définition d'un objet vidéo

Les objets physiques sont les objets du monde réel qui apparaissent dans les scènes observées par les caméras.

Les objets physiques sont divisés en deux types : les objets de contexte et les objets mobiles.

a. Les objets de contexte

Sont des objets physiques qui sont habituellement statiques (par ex : les murs). Dans le cas où ils ne sont pas statiques, leurs mouvements peuvent être

prédits par les informations contextuelles (par ex : Les chaises, les portes...) sont des objets de contexte.

b. Les objets mobiles

Sont des objets physiques qui peuvent être perçus dans les scènes par leurs mouvements. Il est cependant difficile de prédire leurs mouvements (par ex : les personnes, les véhicules). Les objets dans les vidéos sont des objets mobiles.

Ils sont habituellement détectés et suivis dans plusieurs frames (Image composants la vidéo).

I.5.4. Séquence d'image [16,18]

Une séquence d'images est une succession d'images bidimensionnelles qui montre l'évolution temporelle d'une scène. La cadence est de 25 images par seconde, ce qui correspond au seuil à partir duquel l'œil humain perçoit la séquence comme un stimulus continu, grâce à la persistance rétinienne.

Par la suite, nous appellerons « trame » ou « plan » chaque image bidimensionnelle correspondant à un instant donné de la séquence.

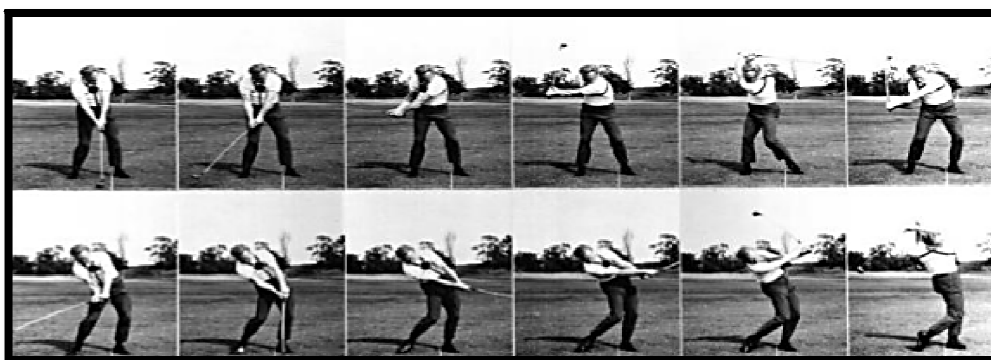


Figure.I.15. Exemple d'une séquence d'images.

I.5.5. Nature de la scène

Selon l'objet présenté et la position de la caméra, le contenu de la séquence vidéo peut changer fortement.

Dans les méthodes de détection de mouvement, les objets sont caractérisés par le fait que ce sont les seules parties de la vidéo qui ont un mouvement important.

Il s'agit d'abord de distinguer les séquences à caméra fixe et les séquences à caméra mobile.

I.5.5.1. Séquences à caméra fixe (Scène statique)

La grande majorité des systèmes telle que les systèmes de vidéosurveillance utilisent des caméras fixes par rapport à la scène filmée.

L'image issue des caméras est alors relativement stable, dans le sens où les changements temporels visibles dans l'image correspondent généralement à des objets d'intérêt.

I.5.5.2. Séquences à caméra mobile (Scène dynamique)

Dans le cas des systèmes tel que le cas d'un système embarqué dans un véhicule La mobilité de ce dernier fait perdre à la scène sa stabilité, même lorsque la caméra est fixée.

Nous ne pouvons donc pas faire l'hypothèse d'une scène statique et les variations présentes sont très difficiles à modéliser. Les variations temporelles se produisant dans une séquence vidéo.

Le mouvement dominant de la scène appliquée à tous les pixels est considéré comme étant le mouvement de la caméra et les pixels ne suivant pas ce mouvement global forme les objets en mouvement.

I.5.5.3. Mono et multi-caméra

Des méthodes de suivi existent dans le cas d'une seule caméra ou de plusieurs caméras. La différence essentielle entre ces deux méthodes provient du fait que les images prises de plusieurs caméras doivent se projeter dans le même espace, ce qui implique la calibration des caméras. Cela n'est pas nécessaire dans le cas du suivi mono caméra. La plupart des méthodes de suivi du corps en mouvement sont basées sur des séquences mono caméras.

En ce qui concerne l'avantage à utiliser plusieurs caméras est de réduire les ambiguïtés causées par les problèmes d'occultations.

I.6.Conclusion

Dans ce chapitre on a abordé certains concepts primaires concernant le traitement d'images et la transformation d'image d'un état à l'autre comme le filtrage, la segmentation...etc.Nous avons vu les notions de base de l'analyse du mouvement dans une séquence d'image. L'objectif de notre travail est la détection et le suivi d'objets en mouvement dans une séquence d'images, les prochains chapitres sont consacrés à la présentation de ce domaine.

Chapitre II :

Méthodes et Techniques
utilisées dans La détection
D'objets en Mouvements

II.1.Introduction [18,19]

La détection d'objets mobiles est une étape clé de nombreux algorithmes de vision par ordinateur, Les applications de la détection d'objets sont nombreuses. On peut notamment citer le suivi d'objets, l'analyse du comportement, la compression vidéo, la reconstruction 3D.

L'analyse du mouvement dans une séquence d'images repose soit sur la détection du mouvement soit sur la segmentation basée mouvement.

La détection de mouvement a pour but de décider quelles parties des images (pixels ou régions) correspondent à des objets mobiles.

La segmentation basée mouvement est une détection multi-label dans laquelle chaque région de l'image ayant un mouvement distinct des autres est segmentée. Ces sujets ont été et sont toujours l'objet d'un nombre important de travaux.

La diversité des recherches est liée à la difficulté de la tâche : les méthodes proposées doivent être le plus robuste possible aux fluctuations des intensités observées (en luminance, couleur ou température). Ces variations sont liées au capteur (légères vibrations, bruit électronique) ou au contenu dynamique de la scène (fond dynamique).

Dans ce chapitre, nous nous intéressons à la détection d'objets en mouvement dans une séquence d'image, par conséquent aux méthodes de séparation des objets mobiles du fond de l'image (l'arrière-plan).

Dans ce contexte, la méthode la plus utilisée est la méthode de détection des contours en mouvement.

On présente également dans ce chapitre des autres méthodes : la méthode de segmentation en se concentrant sur méthode de LPE et la segmentation par contour actif et la méthode de la différence inter images (avec et sans l'image de référence) et la méthode de mcm et la méthode d'Otsu.

II .2 Méthode de Segmentation [20,21]

Pour segmenter une image, les systèmes de vision (aussi bien informatiques qu'humains) réalisent deux tâches à la fois : la reconnaissance des objets d'intérêt et la délimitation précise de leurs contours.

Ces deux tâches sont étroitement liées et forment, et pour connaître un objet dans une image, il faut avoir défini son contour et pour définir le contour d'un objet il faut l'avoir au préalable reconnu.

La section suivante est consacrée à l'étude des techniques de segmentation d'image.

La segmentation consiste à subdiviser une scène réelle en ses parties constituantes (les objets qui composent l'image), en d'autres termes, elle cherche à extraire les primitives de surface ou de contour.

Notons que les pixels d'une primitive sont connexes et homogènes (possèdent la même propriété).

On peut dire que : La segmentation est la partition d'une image en un ensemble de régions qui ne se chevauchent pas et dont l'union est l'image entière.

Quelques règles à suivre pour obtenir une segmentation sont :

- 1) Les régions doivent être uniformes et homogènes par rapport à certaines caractéristiques (niveau de gris, écart type, gradient) ;
- 2) Leurs intérieurs doivent être simples et sans beaucoup de petits trous (des parties de région non segmentés) ;
- 3) Les régions adjacentes doivent avoir des valeurs très différentes par rapport à la caractéristique prise en compte dans la segmentation ;
- 4) Les limites de chaque région doivent être simples et spatialement précises.

II.2.1. Le choix d'une technique de segmentation

Le choix est lié à :

- La nature de l'image (éclairage, contours, texture, etc.) ;
- Aux opérations en aval de la segmentation (compression, reconnaissance des formes, Mesures, etc) ;
- Aux primitives à extraire (droites, régions, textures, etc) ;
- Aux contraintes d'exploitation (temps réel, espace mémoire, etc).

II.2.2 Objectifs de la segmentation [22]

- Fournir des régions homogènes (selon un critère donné) ;
- Localiser de manière précise les contours des régions ;
- L'étude et l'interprétation des structures anatomiques ;
- Réduction de bruit.

II.3 Détection d'objets en mouvement par méthode de Segmentation [23]

La segmentation est une étape primordiale en traitement d'images et de vidéo. La segmentation d'images consiste à déterminer les régions sémantiquement importantes, c.-à-d. les objets, en calculant les régions « homogènes » et les contours de ces régions.

La segmentation d'une image en régions consiste à rassembler les pixels ayant des propriétés communes, comme une couleur ou une texture similaire, ou un mouvement cohérent.

Tandis que la segmentation Que cela soit dans le domaine spatial, temporel ou spatio-temporel, la segmentation a pour objectif de partitionner une image ou un groupe d'images de sorte que chacune des régions ait une caractéristique qui la distingue des autres.

La segmentation spatiale divise l'image en régions homogènes en termes d'intensité. Les différentes homogénéités des régions sont distinguées par leurs frontières entourant.

SALEMBIER et ont proposé une méthode de segmentation spatiale basée sur une étape de simplification d'image utilisant les filtres morphologiques (ouverture et fermeture) et l'algorithme de ligne de partage des eaux (minima locaux).

Les différentes étapes utilisées de la segmentation spatiale :

a) Simplification d'image

Pour la segmentation spatiale, les images sont d'abord simplifiées. Des filtres morphologiques sont conçus pour lisser les images bruitées par des opérateurs d'ouverture et de fermeture utilisant un élément structurant donné.

b) Approximation du gradient [20,21]

Le gradient d'une image donne les taux de changement de niveau de gris par unité de distance dans les directions des axes de coordonnées:

Il est défini comme un vecteur caractérisé par son amplitude et sa direction, tels que :

- L'amplitude est liée à la quantité de variation locale des pixels.
- La direction du gradient est orthogonale à la frontière qui passe au point considéré

Le gradient spatial de l'image simplifiée est appliqué par l'utilisation d'un opérateur morphologique gradient qui consiste à détecter les brusques variations d'intensité lumineuse dans l'image.

c) Détection des lignes de partage des eaux (watersheds) [24,21]

Ligne de Partage des Eaux (LPE ou watershed en anglais) est l'une des techniques de segmentation les plus puissantes de la morphologie mathématique.

Bien qu'elle soit habituellement considérée comme une approche de segmentation par croissance de régions, plusieurs auteurs (Smet 1999), (Hodneland 2007),(Meyer 2012) ont souligné que cet outil de segmentation peut être attribué à la

fois aux techniques de segmentation par contours et aux techniques de segmentation par croissance de régions.

Des méthodes morphologiques comme l'algorithme de la ligne de partage des eaux LPE considèrent le gradient de l'image comme une surface topologique et partent de germes initiaux pour immerger la surface .

Ainsi, les zones où se rencontrent l'eau des différents bassins versants forment les frontières entre les régions.

Ce genre de méthode est très rapide mais fournit un nombre très grand de régions qu'il faudra par la suite fusionner pour obtenir une segmentation correcte des objets de la scène.

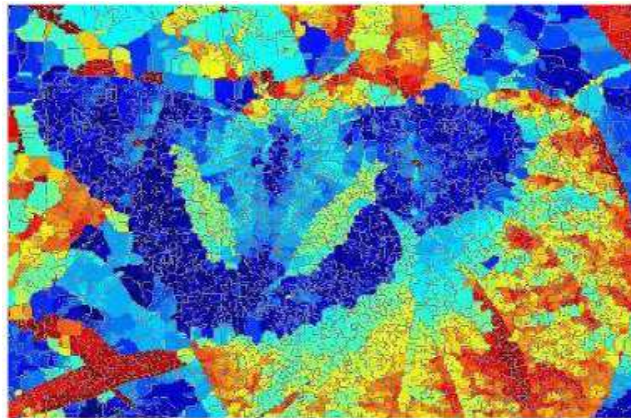


Figure. II. 1. Segmentation en région par watershed [25].

La détection des lignes de partage des eaux (LPE) est une technique classique de segmentation d'image. Cette approche considère l'image à niveaux de gris comme un relief topographique ou l'altitude étant la valeur du niveau de gris.

Ce relief comporte des minima locaux. Ce sont des zones connexes de l'image dont tous les pixels sont à même niveau de gris et dans lesquelles il est impossible.

1) **Fusion de régions**[27]

Une fusion revient à supprimer une frontière et à faire absorber une région par l'autre. Que ce soit dans les approches de fusion-division ou pour d'autres techniques, il est nécessaire de définir une mesure de similarité ou de dis similarité entre deux régions.

Le but de la fusion est de diminuer le nombre de régions en vue de certains traitements. L'objectif est d'obtenir des régions homogènes.

- **Fusion sur un graphe d'adjacence des régions [25]**

Les techniques par graphes d'adjacence sont beaucoup utilisées sur les régions sur segmentées. Lors de la fusion de deux régions, l'une est choisie arbitrairement pour "absorber" l'autre. Les données de la région absorbante sont alors "augmentées" de celles de la région absorbée.



Figure. II. 2. Partitions et graphes d'adjacences associés.

L'idée de base consiste à plonger les régions obtenues dans une structure de graphe ou une région est un nœud et un arc est une relation d'adjacence.

Puis on définit une fonction de similarité entre 2 nœuds. On trie tous les couples de nœuds adjacents dans une liste ordonnée. On regroupe les 2 candidats sur un critère pré défini. On remet à jour la liste et on itère.

II.4. Détection par différence d'images [27]

Les méthodes basées sur la différence entre deux ou trois images successives sont généralement rapides et faciles à mettre en place.

Elles s'adaptent relativement bien aux changements brutaux dans le fond et moins bien aux changements de luminosité.

La plupart des algorithmes ne se basent que sur l'intensité mais cela ne permet pas la détection d'objets ayant une intensité proche de celle des pixels du fond.

D'autres méthodes s'appuient uniquement sur le flot optique ce qui ne permet pas de différencier les objets ayant un mouvement proche de celui du fond.

Ainsi, déterminer la probabilité que chaque pixel appartienne au fond en n'utilisant que l'intensité ou le mouvement sur deux ou trois images successives ne permettra pas de détecter uniquement les objets mobiles intéressants.

Pour conclure, ces méthodes ne peuvent permettre à elles seules de détecter les objets mobiles dans les séquences que nous considérons. Néanmoins, ces techniques peuvent donner une bonne indication des pixels en mouvement et peuvent donc être utilisées comme prétraitement pour améliorer et accélérer les autres techniques de détection ou de segmentation de mouvement.

II.4.1 Approche région

La méthode de l'approche région cherche à différencier les régions en utilisant les propriétés de l'image telles que la couleur, texture, forme...etc. Nous proposons dans la suite les différentes méthodes de segmentation de type région.

a. Avec image de référence [28, 29,30]

Dans certaines applications (acquisition avec caméra fixe en particulier), il peut être possible de disposer d'une image dans laquelle seuls les éléments stationnaires sont présents.

Cette image est alors utilisée comme une image de référence $R(x, y)$ notée R . Cette image doit posséder les caractéristiques suivantes : Etre exempte de tout objet mobile. Avoir été acquise sous les mêmes conditions d'éclairiments que les images avec lesquelles elle sera comparée. L'image de différence notée D devient alors :

$$D = |I - R| \quad (II.1)$$

Ou I représente l'image à l'instant t et R est l'image de référence. Les régions en mouvement sont obtenues après seuillage de l'image différence (nécessaire à cause du bruit). Cette image de différence fait apparaître 2 types de régions :

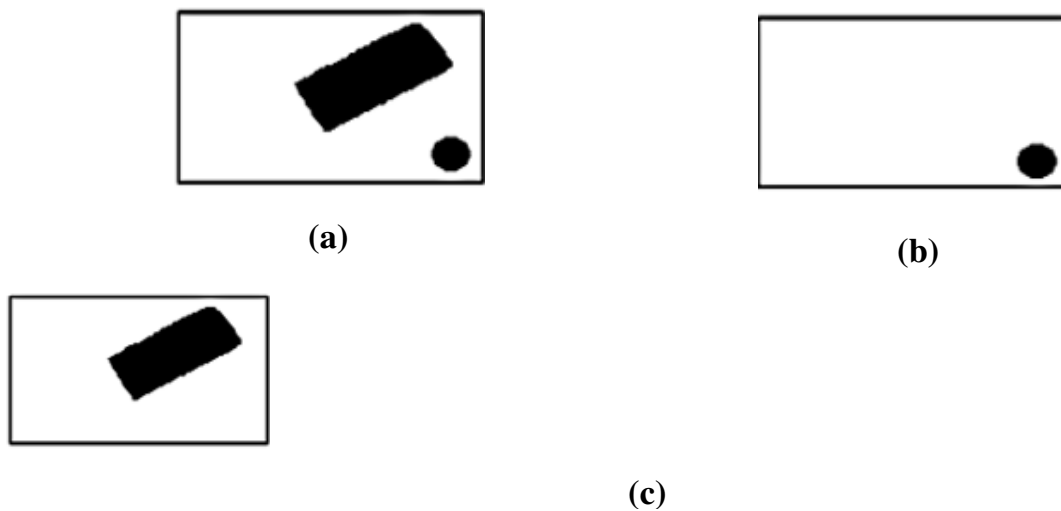


Figure.II.3. Technique de différence avec image de référence, (a) l'image à l'instant t , (b) l'image de référence, (c) l'image de différence.

F

- La première zone est constituée des points appartenant aux objets mobiles dans leur position courante.
- La seconde zone est constituée des points stationnaires présents dans les deux images $I(x, y, t)$ et $R(x, y)$.

Cette méthode est très utilisée dans le domaine de détection d'objet en mouvement mais la plus grande difficulté consiste à obtenir l'image de la référence (du fond).

Dans certains cas cette image peut être obtenue en sélectionnant manuellement une image où aucun objet n'apparaît, mais ce n'est pas toujours le cas. L'image de

référence doit en permanence être identique à l'environnement statique perçu dans chaque image de la séquence. Jain propose un algorithme qui repose sur l'analyse de l'image différence.

Makarov construit l'image de référence en se fondant sur l'analyse de la moyenne du niveau de gris d'un pixel donné sur une séquence de N images.

A chaque acquisition d'une nouvelle image I de la séquence, une nouvelle image de référence R est construite. Le niveau de gris de chaque pixel de l'image de référence est déterminé par la relation suivante :

$$R(x, y, t) = 1/N \sum_{i=t-n}^{t-1} I(x, y, t) \quad (\text{II. 2})$$

Où N : le nombre des images dans la séquence. Cette méthode nécessite une mémorisation de N images successives de la séquence, mais cette opération est assez coûteuse.

Une autre difficulté vient des changements de luminosité ; même si l'hypothèse du faible changement entre deux images successives est respectée, le changement à long terme peut être important sur une séquence très longue.

a. Sans l'image de référence

La différence temporelle s'exprime par :

$$F_D(IC, IP) = |IC - IP| \quad (\text{II. 3})$$

Où : F_D : image de différence ; IC image courante, et IP image précédente.

L'image $F_D(IC, IP)$ ainsi obtenue est nulle en tous points où le signal d'entrée I est constant. Les points où un changement temporel s'est produit sont détectés par simple seuillage de F_D .

Le seuillage est nécessaire à cause du bruit présent dans les 2 images. Le résultat du seuillage est une image binaire indiquant les zones en mouvement. Selon l'exemple de la figure (4) la différence fait apparaître quatre zones distinctes.

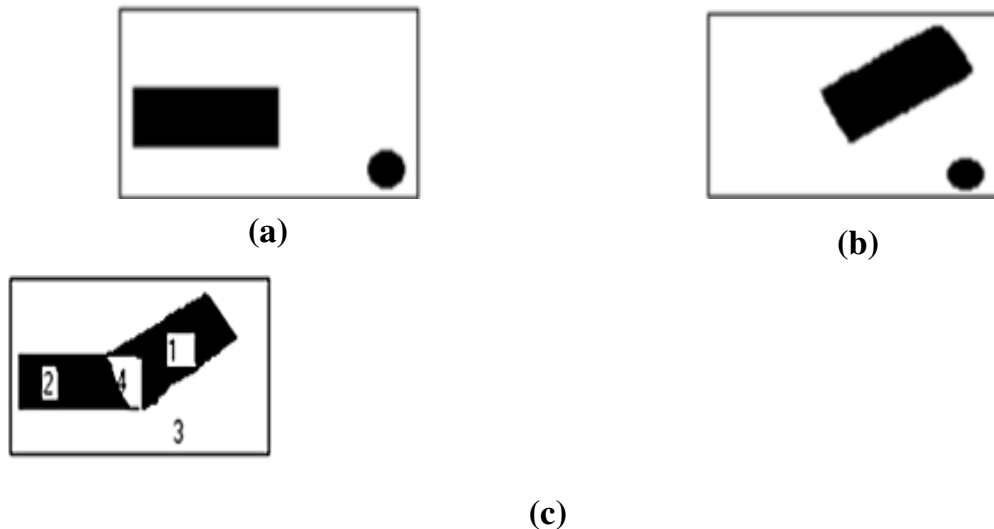


Figure. II.4. Technique de différence sans image de référence, (a) image précédente, (b) image courante, (c) image de différence.

- ❖ La zone 1 située à l'avant de l'objet dans le sens du déplacement est constituée des points qui appartenaient au fond de l'image précédente et qui sont recouverts par l'objet dans l'image courante. L'amplitude des différences de niveaux de gris y est fonction du contraste entre l'objet et le fond.
- ❖ La zone 2 située à l'arrière de l'objet est constituée des points qui appartenaient à l'objet dans l'image précédente et qui ont été découverts suite au mouvement. Ils appartiennent au fond de l'image courante. L'amplitude des différences de niveaux de gris y est fonction du contraste entre l'objet et le fond.

- ❖ La zone 3 constituée des points appartenant au fond et aux objets stationnaires dans les deux images. L'amplitude des différences de niveaux de gris y est faible.
- ❖ La zone 4 qui n'existe que lorsqu'il y a chevauchement des positions de l'objet et qui est constituée des points appartenant à l'objet en mouvement dans les deux images. L'amplitude des différences de niveaux de gris y est faible.

Ainsi, les régions 1 et 4 retracent l'objet dans l'image à l'instant t , et les régions 2 et 4 à l'instant $t - dt$, les régions 1 et 2 sont détectées comme des régions en mouvement, par contre, la région 4 l'est rarement et le problème est d'extraire cette région.

II.4.2.Approche contour[31]

Un contour est un ensemble des points d'une image numérique qui correspond à un changement brutal de l'intensité lumineuse.

Dans l'approche 'contour', on considère que les primitives à extraire sont les lignes de contrastes séparant des régions de niveaux de gris différents et relativement homogènes, ou bien des régions de texture différentes.

En pratique, il s'agit de reconnaître les zones de transition et de localiser au mieux la frontière entre les régions.

Il existe plusieurs opérateurs spécifiques tels que :

1. Proposition de HAYNES[32,28]

Il propose d'effectuer le produit simple (AND) entre l'image de différence $F_D(IC, IP)$ et l'image des contours $G(IC)$ soit :

$$F_D(IC, IP) = |IC - IP| \quad (II. 4)$$

$$CM(IC) = F_D(IC, IP).G(IC) \quad (II. 5)$$

La différence inter-image est considérée comme un détecteur de mouvement, l'opérateur

$G(IC)$ Est considéré comme un détecteur de contour, et le produit est alors un détecteur de contours en mouvement.

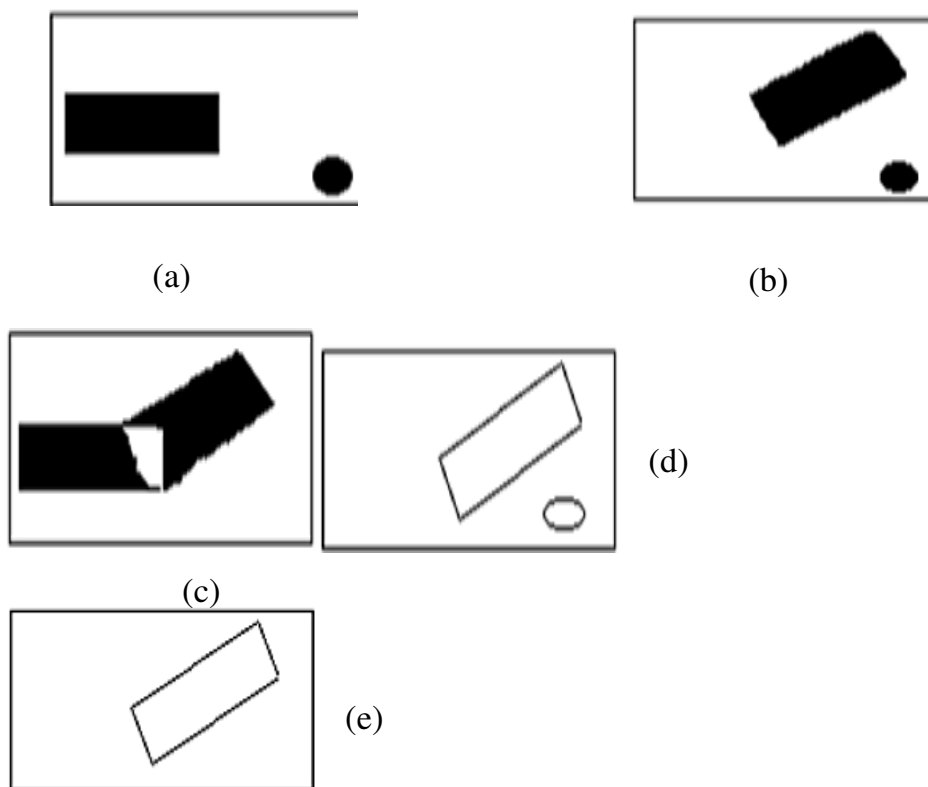


Figure II.5. Extraction des contours en mouvement par la proposition de Haynes, (a) image précédente, (b) image courante, (c) image de différence, (d) contour d'image courante, (e) contour en mouvement.

Bien que sensible au bruit, un seuillage approprié de $CM(IC)$ est néanmoins nécessaire afin de ne conserver que les points des contours en mouvement. Cette technique rencontre aussi des difficultés lorsqu'un objet se déplace parallèlement à l'un de ses contours.

Dans ce cas, malgré la présence d'un contour $G(IC)$ fort, la différence inter-images est voisine de zéro et par conséquent, $CM(IC)$ est trop faible pour permettre de détecter le contour en mouvement.

Une excellente critique de ce détecteur démontre que pour un déplacement donné, le détecteur n'est fonction que de la différence inter-images.

Lorsque celle-ci est trop faible, la réponse du détecteur est faible. Ce détecteur peut aussi répondre à des contours variant temporellement qui ne sont pas nécessairement des contours en mouvement.

Un contour variant temporellement peut être dû à un changement d'éclairage, à un contour statique caché par un objet en mouvement ou bien à un objet se déplaçant.

2. L'opérateur de Stelmaszyck

Elle est basée sur le fait que les contours en mouvement sont les seuls à être présents à la fois dans l'image des contours courants $G(IC)$ et dans l'image des contours de la différence .

$G(F_D(IC, IP))$ On obtient alors :

$$CM(IC).G(F_D(IC, IP)) \quad (II. 6)$$

Cette méthode est élégante et homogène car l'opérateur d'extraction de contours est le même pour l'image courante et pour l'image des différences.

D'autre part, le produit s'effectuant entre deux images de contours, les valeurs combinées correspondent forcément au même point de l'image courante.

Par conséquent, tous les points de contours en mouvement répondront fortement alors que pour les détecteurs de Haynes et Jain beaucoup de points de contours en mouvement sont perdus si l'amplitude de la différence est faible.

Par contre, ce détecteur pose un problème délicat lorsque les objets se déplacent sur un fond non uniforme. Pour ce cas, l'image $G(IC)$ contient les contours du fond qui étaient cachés par l'objet dans sa position précédente.

L'image $G(F_D(IC, IP))$ contient aussi ces contours puisque $F_D(IC, IP)$, contient les points du fond en ces endroits-là, par conséquent, ces points du contour du fond, communs aux deux images, apparaissent dans $CM(IC)$. Pour éviter cela, Vieren propose de mettre en jeu une troisième image.

3. L'opérateur de Vieren [33]

Vieren a développé un opérateur qui permet d'extraire les contours extérieurs d'objets en mouvement devant un fond uniforme ou non. Pour cela, trois images successives de la scène dynamique sont nécessaires :

- L'image précédente, notée P , l'image courante, notée C , l'image suivante, notée S .

Deux images différences sont calculées à partir de ces trois images : l'image différence, notée $D(P, C)$, entre l'image courante et l'image précédente, l'image différence entre l'image suivante et l'image courante, notée $D(C, S)$.

Ces différences d'images permettent d'extraire les zones de l'image affectées par le mouvement. Dans l'image courante. L'expression de $CM(C)$, l'opérateur de détection des Contours en Mouvement dans l'image Courante, est ainsi donné par : Les contours des objets en mouvement sont ensuite détectés grâce à un opérateur de type gradient. Les deux images résultantes, notées $G(D(P, C))$ et $G(D(C, S))$, contiennent plusieurs types de contours :

- Les contours extérieurs des objets en mouvement, dans les positions qu'ils occupent dans l'image courante, se retrouvent dans les images

$G(D(P, C))$ et $G(D(C, S))$;

- Les contours extérieurs des objets en mouvement dans les positions qu'ils occupent dans l'image précédente se retrouvent dans l'image $G(D(P, C))$;
- Les contours dans les positions qu'ils occupent dans l'image suivante se retrouvent dans $G(D(C, S))$;
- Les contours du fond dans des zones de l'image qui ont été recouvertes puis découvertes par les objets en mouvement peuvent être présents dans plusieurs des trois images successives ;
- On effectue la multiplication point à point des deux images $G(D(P, C))$ et $G(D(C, S))$.

Afin de ne garder parmi ces trois types de contours que ceux correspondant à la position des objets en mouvement :

$$CM(C) = G(D(P, C)) \times G(D(C, S)) \quad (\text{II.7})$$

4. L'opérateur d'Orkisz [34]

La multiplication introduisant un changement de dynamique dans les images résultats, Orkisz a préféré travailler avec un opérateur qui ne modifie pas cette dynamique.

C'est pourquoi il a choisi l'opérateur Max comme opérateur de coïncidence entre deux images. L'opérateur Max appliqué à deux images consiste à remplacer le niveau de gris de chaque pixel par le niveau de gris Maximal de ce pixel dans les deux images.

Orkisz s'est également attaché à construire un opérateur qui ne détecte pas les contours statiques présents à l'intérieur du contour extérieur des objets mobiles, contours provenant des éléments du fond qui sont découverts au cours du mouvement. Son opérateur est le suivant :

$$CM(C) = MAX[G(C), G(P), G(S)] - MAX[G(P), G(S)] \quad (II. 8)$$

Cet opérateur utilise également trois images successives de la séquence, afin de s'affranchir des non-uniformités du fond.

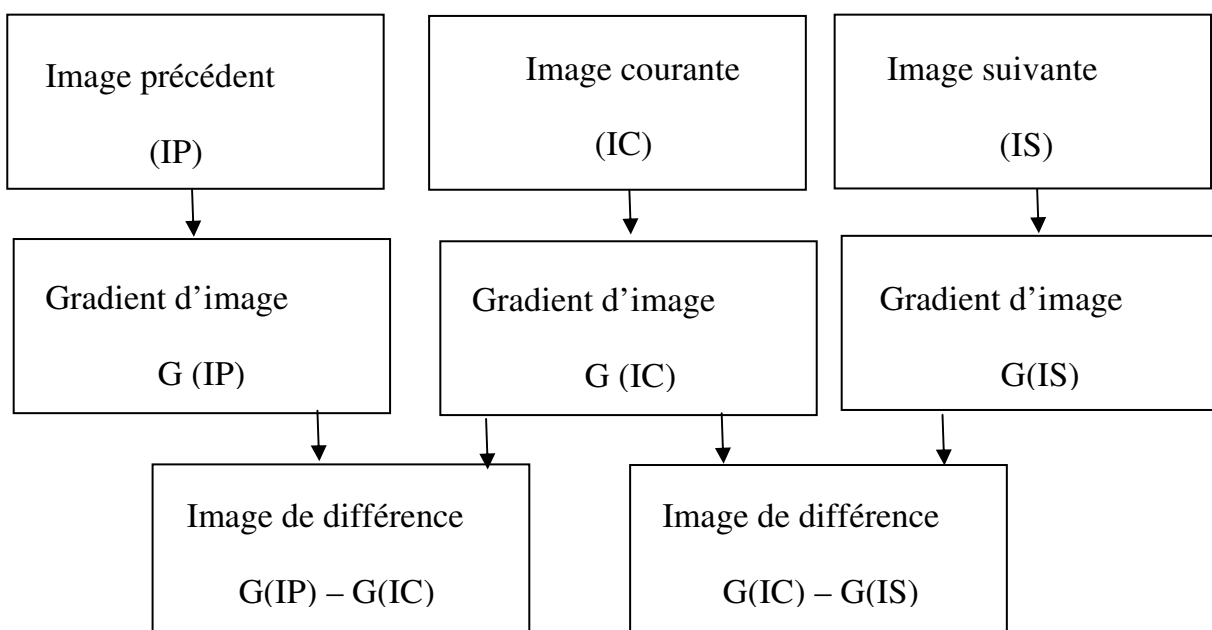
II.5. Méthode MCM (la mesure des contours en mouvement) [35, 36,37]

L'objectif de cette étape est de détecter les contours des régions en mouvement. Donc la méthode proposée prend en compte trois images successives IP, IC, IS et on calcule le module du gradient de chaque image. Ensuite, on applique symétriquement la différence absolue sur les deux couples successifs de module du gradient.

La mesure des contours en mouvement (mcm) à un temps t donné est alors définie par le minimum (AND) de ces deux différences :

$$mcm = \min(|G(IP) - G(IC)|, |G(IC) - G(IS)|) \quad (II. 9)$$

La figure (II -6) résume les différentes étapes nécessaires du calcul du mcm



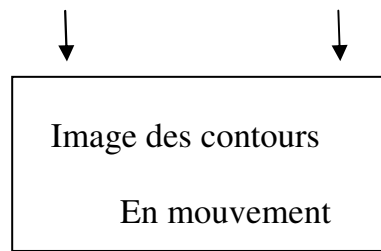


Figure. II.6. Schéma synoptique de l'opérateur mcm.

L'opérateur mcm est robuste à l'amplitude du déplacement des objets (mouvement rapide) et au bruit aléatoire présent dans l'image.

Mais l'obtention de contours dans le cas d'un mouvement de faible amplitude (mouvement lent) par le mcm sont incomplet probablement non satisfait, donc dans ce cas-là on applique la méthode de différence entre deux images suivi par un seuillage.

II.6. Segmentation d'image par la méthode du contour actif [36, 37,38]

Un contour actif (CA) est une courbe qui évolue d'une forme initiale vers les frontières de l'objet d'intérêt et L'objectif principal de cette technique est et le CA évolue ainsi jusqu'à ce qu'il atteigne les frontières de l'objet désiré,

Il est important de noter que ce type d'approche est dont le principe consiste à placer dans l'image au voisinage de la forme à détecter un contour initial(courbe) qui est ensuite déformé sous l'action de plusieurs forces :

- $E_{interne}$: Une énergie interne permettant de régulariser le contour .
- E_{image} : Une énergie potentielle liée à l'image dont l'objectif est de plaquer la courbe sur le contour à détecter. image
- $E_{externe}$:.Une énergie externe introduite par l'utilisateur pour traduire les contraintes spécifiques du problème posé.

Ces forces vont permettre au contour actif d'évoluer pour rechercher la position d'énergie minimale qui sera ainsi un compromis entre les diverses contraintes du problème.La technique du CA est devenue très populaire et est

largement utilisée dans la segmentation d'image, particulièrement bien adapté à la segmentation d'un objet dans une image, voire dans une séquence d'images.

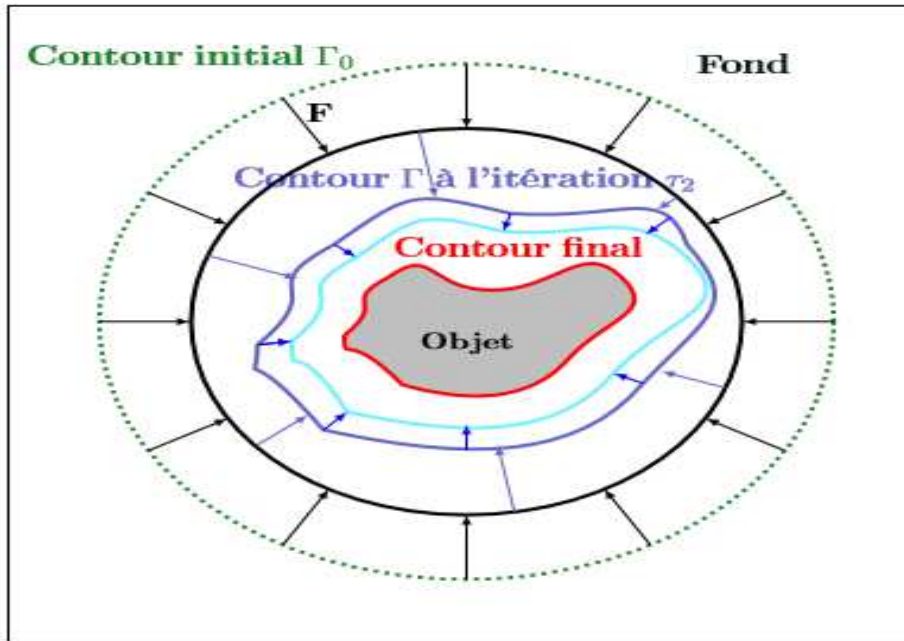


Figure. I. 7. Représentation schématique de l'évolution itérative du contour actif Γ vers les bords de l'objet d'intérêt dans l'image à segmenter.

Le principe de cette méthode est de déplacer les points pour les rapprochés des zones de fort gradient tout en conservant les caractéristiques initiales comme la courbure, changement de forme du contour ou d'autre contraintes liées au positionnement des points.

Le contour est représenté par une courbe $C = v(s, t)$, ouverte ou fermée, paramétrée par l'abscisse curviligne s tel que $s \in [0, 1]$ et t représente le temps.

Le processus de déformation est lié à la minimisation d'une fonctionnelle d'énergie, notée E . L'énergie totale du contour C s'exprime sous la forme :

$$E_{interne}(v(s)) = a(s)(dv/ds)^2 + b(s)(dv^2/ds^2)^2 \quad (II. 10)$$

Et $a(s)$: coefficient d'élasticité, $b(s)$: constante de raideur.

E Image : terme d'attache aux données qui caractérise les contours que l'on souhaite approcher. Dans le cas de la détection simple de contours, ce sont par exemple les lignes de fort gradient,

Les contours actifs, comme toutes les méthodes de détection de contour, présentent des limites :

- À l'initialisation, le contour doit être le plus proche possible de l'objet à détecter au risque de ne pas être attiré par l'objet à cause de la portée réduite du gradient de l'image ;
- Les contours actifs sont connus pour leur sensibilité à l'initialisation et leurs temps de convergence longs ;
- Les contours actifs basés contour sont efficaces mais ils souffrent d'une grande sensibilité au bruit et leur utilisation reste limitée aux images où les frontières entre les différents objets peuvent être définies par de simples gradients.
- Ainsi, leur évolution est plus délicate dans des images complexes telles que les images texturées ;
- Les contours actifs basés région possèdent une meilleure capacité de segmentation, mais celle-ci reste fortement dépendante du descripteur de régions choisi pour guider la courbe ;
- La difficulté de choisir les paramètres λ déterminant le poids de chaque énergie dans l'énergie totale ;
- Le contour actif converge vers un minimum d'énergie local ;
- Plusieurs objets ne peuvent être détectés simultanément sans utiliser de procédures spéciales.

II.7. Seuil d'image global en utilisant la méthode d'Otsu

II.7.1. Définition du seuillage [40,41]

Il s'agit d'utiliser l'histogramme de l'image dans le but de partitionner l'image en une ou plusieurs partitions (objets).

Chaque classe correspond à une gamme distincte de niveaux de gris. Ceci est réalisé en associant à chaque pic de l'histogramme une classe, le cas où la segmentation donne seulement deux classes est appelé binarisation.

La binarisation transforme l'image d'entrée en une image binaire c'est-à-dire que les pixels sont soit noir soit blanc. La difficulté majeure de cette transformation est de trouver le bon seuil.

Pour faire le seuillage d'une image, on remplace la valeur des pixels en fonction d'une valeur fixe de seuil, de la manière suivante [37] :

$$I_0(x, y) = \begin{cases} 0 & \text{si } I(x, y) \leq t, t \in [0, 255] \\ 255 & \text{sinon} \end{cases} \quad (\text{II. 11})$$

Exemple :



(a)



(b)

Figure II.8. Exemple de Seuillage d'une image. (a) image initial (b) image seuillé

Les méthodes de seuillage global reposent sur l'exploitation de l'histogramme de toute l'image. L'histogramme est une fonction discrète notée $p(i)$, qui caractérise la distribution des niveaux de gris. Il est défini par la probabilité d'apparition du niveau de gris i ($i=0, 1, \dots, L-1$) tel que :

$$P(i) = h(i)/N \quad (\text{II. 12})$$

$H(i)$ Étant le nombre de pixels ayant le niveau de gris i et N le nombre total de pixels dans l'image.

A partir de l'allure de l'histogramme, on peut déduire le nombre de classes ainsi que la position relative des seuils (Figure II.9.).

L'histogramme est unimodal, s'il est formé d'un seul pic représentant les pixels de l'objet ou ceux du fond. Il est bimodal s'il est caractérisé par deux modes séparés par une vallée, il indique l'existence d'un objet sur un fond.

L'histogramme multimodal comporte plus de deux modes séparés par des vallées, il indique la présence de plusieurs classes dans image (chaque mode correspond à une classe). Les seuils doivent être localisés dans les vallées.

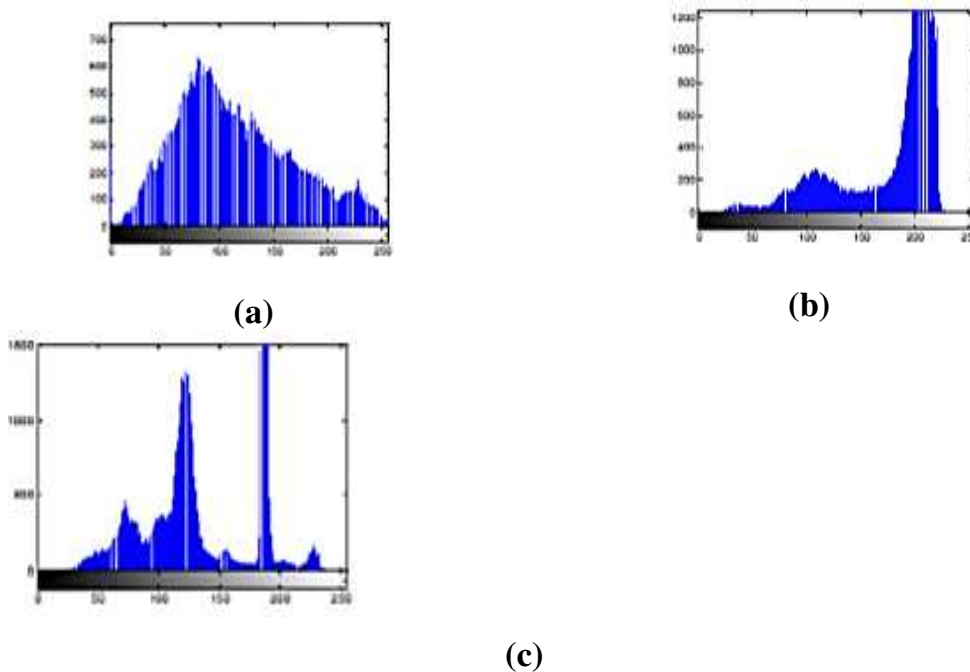


Figure II. 9. Allures idéales d'un histogramme. (a) : unimodal, (b): bimodal et (c) multimodal .

En pratique, il est rare de trouver un histogramme qui présente des modes bien distincts. En effet, la plupart des images présentent des histogrammes bruités caractérisés par des modes non discernables.

La question qui se pose donc est “comment calculer les seuils t_k ? ” Plusieurs techniques ont été proposées. La plupart d’entre elles permettent de trouver les seuils optimaux sans tenir compte d’aucune hypothèse sur la forme de l’histogramme.

Elles sont généralement basées sur l’optimisation d’une fonction objective.

La fonction objective la plus connue est celle proposée par Otsu.

II .7.2. Méthode d’Otsu

Elle est considérée comme la méthode de référence dans le domaine du seuillage d’histogrammes. Dans cette méthode, le seuillage consiste à séparer les pixels d’une image en deux classes C_1 (fond), C_2 (objet) à partir d’un seuil t .

La classe « fond » regroupe tous les pixels ayant un niveau de gris inférieur au seuil t alors que la classe « objet » contient tous les pixels de niveau de gris supérieur à

t . $C = \{0,1, \dots t\}$ et $C = \{t + 1 \dots L - 1\}$ le calcul du seuil t est effectué en maximisant la variance inter-classes ou bien en minimisant la variance intra-classes.

Soient $\sigma_W^2(t)$ la variance intra-classes, $\sigma_B^2(t)$ la variance inter-classes telles que :

$$\sigma_B^2(t) = P_1 * P_2 (\mu_2 - \mu_1)^2 \quad (\text{II. 13})$$

Et
$$\sigma_W^2(t) = \sum_{i=0}^t P_1 (i - \mu_1)^2 + \sum_{i=t+1}^{L-1} P_2 (i - \mu_2)^2 \quad (\text{II. 14})$$

Soient σ_T^2 la variance totale de l’image telle que :

$$\sigma_T^2 = \sum_{i=1}^{l-1} P_i (i - \mu)^2 \quad (\text{II. 15})$$

Elle est liée aux deux autres variances comme suit :

$$\sigma_T^2 = \sigma_B^2(t) + \sigma_W^2(t) \quad (\text{II. 16})$$

$\mu_1(t)$ et $\mu_2(t)$ représentent les niveaux de gris moyens des classes C_1 et C_2 respectivement :

$$\mu_1(t) = \sum_{i=1}^t i \cdot P_i / P_1 \text{ et } \mu_2(t) = \sum_{i=t+1}^{L-1} i \cdot P_i / P_2, \mu = \sum_{i=1}^{L-1} i \cdot P_i \quad (\text{II. 17})$$

$P_1(t)$ et $P_2(t)$ représentent respectivement les probabilités à priori des classes C_1 et C_2 tels que :

$$P_1(t) = \sum_{i=1}^t P_i \text{ et } P_2(t) = \sum_{i=t+1}^{L-1} P_i, \quad P_1(t) + P_2(t) = 1 \quad (\text{II. 18})$$

La méthode d'Otsu consiste à déterminer le seuil optimal : t^* en maximisant la variance inter-classe :

$$t^* = \text{Arg max } \sigma_B^2(t) \quad (\text{II. 19})$$

Dans le cas du multi seuillage, la méthode d'Otsu peut être étendue au calcul d'un ensemble de seuils $T = (t_1, t_2, \dots, t_{k-1})$ en maximisant la variance inter-classe :

$$T^* = \text{Arg max } J(T) \text{ ou } J(T) = \sum_{K=1}^K P_K (\mu_K - \mu)^2 \quad (\text{II. 20})$$

Avec P_K et μ_K la probabilité et la moyenne de la classe C_K tels que [40] :

$$P_K = \sum_{i=t_{k-1}}^{t_k-1} P_i \text{ et } \mu_K = \sum_{i=t_{k-1}}^{t_k-1} i P_i \quad (\text{II. 21})$$

II .7.3. Seuils d'image multiniveaux utilisant la méthode d'Otsu [39, 40,41]

La méthode traditionnelle Otsu est l'une des techniques efficaces qui fournissent des résultats de segmentation supérieurs pour une image ordinaire.

Elle s'exécute bien si l'image a une distribution bimodale de l'histogramme. De plus, s'il existe une différence distincte entre les objets de premier et de second plan, la méthode Otsu donne de meilleurs résultats, mais si les limites de l'image se chevauchent, il sera difficile d'obtenir une valeur seuil appropriée pour ces objets.

Par conséquent, le seuillage à plusieurs niveaux peut être appliqué pour segmenter l'image en question en sous-régions uniques lorsque la distribution des valeurs de gris des pixels n'est pas distincte, mais les résultats de la segmentation sont affectés par plusieurs facteurs, notamment les valeurs de seuil et le nombre de seuils.

Les techniques de seuillage peuvent être divisées en catégories à deux niveaux et à plusieurs niveaux, en fonction du nombre de segments d'image. Dans le seuillage à deux niveaux, l'image est segmentée en deux régions différentes, tandis que dans le seuillage à plusieurs niveaux, l'image est segmentée en plusieurs régions distinctes.

Par conséquent, la méthode de l'Otsu est un seuillage multi-niveaux étendu.

Le critère d'Otsu a été principalement appliqué pour le seuillage à deux niveaux et peut ensuite être étendu au seuillage à plusieurs niveaux.

1. Seuillage à deux niveaux

Dans le seuillage d'Otsu, l'image en niveaux de gris est caractérisée par des niveaux de gris $(1, 2, \dots, L)$ ayant des pixels avec des niveaux de gris provenant de l'ensemble d'entiers $\{0, 1, 2, \dots, 255\}$. dans l'approche à deux niveaux, l'image est séparée par un seuil à un niveau t en deux classes, à savoir C_0 et C_1 .

le groupe C_0 comprend les régions grises de 0 à $t-1$, et le groupe C_1 comprend les régions grises de t à $L-1$. dans ce cas, la probabilité de distribution des niveaux de gris ($X_0(t)$ et $X_1(t)$) pour C_0 et C_1 , respectivement, peut être exprimée comme suit :

$$C_0 = \frac{P_0}{X_0(t)} \cdots \cdots \frac{P_{t-1}}{X_0(t)} \text{ et } C_1 = \frac{P_t}{X_1(t)} \cdots \cdots \frac{P_{L-1}}{X_1(t)} \quad (\text{II. 22})$$

Les niveaux moyens μ_0 et μ_1 pour C_0 et C_1 peuvent être donnés par :

$$X_0(t) = \sum_{i=0}^{t-1} P_i \text{ et } X_1(t) = \sum_{i=t}^{L-1} P_i \quad (\text{II. 23})$$

$$\mu_0 = \sum_{i=0}^{t-1} \frac{iP_i}{X_0(t)} \text{ et } \mu_1 = \sum_{i=t}^{L-1} \frac{iP_i}{X_1(t)} \quad (\text{II. 24})$$

L'intensité moyenne (μ_t) de l'image entière peut être caractérisée comme :

$$\mu_T = X_0\mu_0 + X_1\mu_1 \text{ et } X_0 + X_1 = 1 \quad (\text{II. 25})$$

Pour problème de seuillage à deux niveaux, La variance entre classes d'Otsu peut être exprimée comme suit :

$$J(t)_{max} = \sigma_0 + \sigma_1 \quad (\text{II. 26})$$

Où σ_0 et σ_1 sont les écarts de pour C_0 et C_1 , respectivement.

2. Seuils à plusieurs niveaux [42,43]

Pour le seuillage multi-niveaux, supposons que des niveaux de seuil "m" (t_1, t_2, \dots, t_m) divisent l'image en m classes : C_0 avec des niveaux de gris compris entre 0 et $t_1 - 1$, C_1 avec des niveaux fermés de t_1 à $t_2 - 1, \dots$, et C_m avec des niveaux de t_m à $L - 1$. Ainsi, la fonction objectif pour le processus de segmentation peut être exprimée par :

$$J(t)_{max} = \sigma_0 + \sigma_1 + \cdots + \sigma_m, \sigma_0 = X_0(\mu_0 - \mu_t)^2 \quad (\text{II. 27})$$

$$\text{Où } \sigma_1 = X_1(\mu_1 - \mu_t)^2, \dots, \sigma_m = X_m(\mu_m - \mu_t)^2 \quad (\text{II. 28})$$

Dans le travail actuel, un algo-rithme méta-heuristique centré sur la segmentation d'images à plusieurs niveaux est abordé en utilisant la fonction d'Otsu et le processus de seuillage est exécuté avec $m = 1,2,3; 4 \text{ et } 5$ (c'est-à-dire cinq classes) .

II.8. Seuillage d'une image couleur[42,43]

Les méthodes de seuillages ont été initialement développées pour des images en niveaux de gris.

Récemment, ces méthodes ont été étendues aux images couleur. Contrairement aux images en niveaux de gris, une image couleur I est composée de N pixels ou chaque pixel est caractérisé par trois composantes au lieu d'une seule (niveau de gris).

Les composantes les plus utilisées sont le rouge (R), le vert (V) et le bleu (B). Dans ce cas on parle d'images RVB. D'autres composantes couleurs ont été proposées.

La performance d'une procédure d'une segmentation d'images dépend du choix de l'espace couleur. Plusieurs auteurs ont tenté de déterminer les espaces couleurs qui sont les mieux appropriés à leurs problèmes spécifiques de segmentation d'image couleurs.

La figure II .10 montre une image couleur décomposée en trois images ou chaque image représente l'une des trois composantes R, V et B. L'histogramme de chacune de ces images composantes présente des allures différentes.

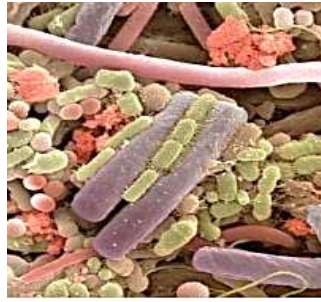
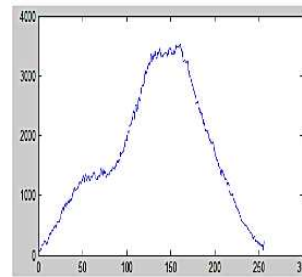
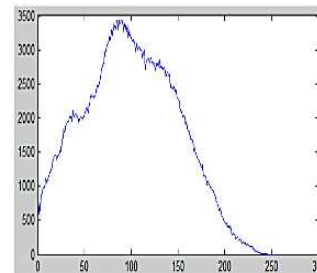


Image originale.



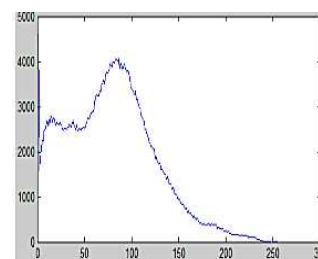
Histogramme du Composant R



Histogramme du composant V



Composant B



Histogramme du composant B

Figure. II.10.Décomposition d'une image couleur en composantes R, V, B ainsi que leurs histogrammes

La segmentation d'images couleur consiste à partitionner l'image en régions ayant des couleurs homogènes ou les pixels d'une même région partagent des propriétés colorimétriques similaires.

La segmentation d'une image couleur basée sur des procédures de classification peut être abordée de deux manières. La première consiste à regrouper les pixels en classes. La seconde consiste à détecter les modes de l'histogramme multidimensionnel

$$P(r, v, b) \text{ ou } P(r, v, b) = H(r, v, b)/N \quad (II. 29)$$

$H(r, v, b)$ étant le nombre de pixels ayant un niveau de rouge r , un niveau de vert v et un niveau de bleu avec $r, v, b = 1 \dots, L - 1$.

Cette méthode peut être complexe. Une alternative consiste à analyser les histogrammes marginaux $P(r), P(v), P(b)$ en effectuant un seuillage de chaque histogramme en K classes.

Soient $T^R = \{t_1^R, t_2^R, \dots, t_{K-1}^R\}$, $T^V = \{t_1^V, t_2^V, \dots, t_{K-1}^V\}$ et $T^B = \{t_1^B, t_2^B, \dots, t_{K-1}^B\}$ les seuils de chaque composante R, V, B .

Un pixel de coordonnées (x, y) de couleur $r(x, y), v(x, y)$ et $b(x, y)$ est affecté à la classe :

$$C_K^R \text{ si } t_k^R \leq r(x, y) \leq t_{k-1}^R \quad k = 0, 1, \dots, k - 1 \quad (II. 30)$$

$$\text{Si } t_k^V \leq v(x, y) \leq t_{k-1}^V \quad k = 0, 1, \dots, k - 1 \quad (II. 31)$$

$$C_K^B \text{ si } t_k^B \leq b(x, y) \leq t_{k-1}^B \quad k = 0, 1, \dots, k - 1 \quad (II. 32)$$

C_K^R, C_K^V et C_K^B Sont respectivement les classes des composantes rouge, vert et bleu. La construction de l'image couleur segmentée peut être obtenue par recombinaison des trois composantes segmentées.

Le nombre de classes de l'image couleur segmentée sera donc au maximum égal à $(3\bar{T} - 1)$. Dans le cas de la méthode d'Otsu, les seuils de chaque composante peuvent être déterminés indépendamment en maximisant :

$$\mathcal{J}(T^V) = \sum_{K=1}^k P_K^V (\mu_K^V - \mu^V)^2 \quad (\text{II. 33})$$

$$\mathcal{J}(T^R) = \sum_{K=1}^k P_K^R (\mu_K^R - \mu^R)^2, \quad \mathcal{J}(T^B) = \sum_{K=1}^k P_K^B (\mu_K^B - \mu^B)^2 \quad (\text{II. 34})$$

Ou d'une manière conjointe en maximisant :

$$\mathcal{J}(T^R, T^V, T^B) = \mathcal{J}(T^R) + \mathcal{J}(T^V) + \mathcal{J}(T^B) \quad (\text{II. 35})$$

Avec : $P_K^{(R,V ou B)} = \sum_{i=t_{K-1}^{(R,V ou B)}}^{t_K^{(R,V ou B)}} P_i$ la probabilité de la classe $C_K^{(R,V ou B)}$

$$\mu_K^{(R,V ou B)} = \sum_{i=t_{K-1}^{(R,V ou B)}}^{t_K^{(R,V ou B)}} i \cdot P_i \quad (\text{II. 36})$$

$\mu_K^{(R,V ou B)}$ Est la valeur moyenne rouge, vert ou bleu classe $C_K^{(R,V ou B)}$

$\mu^{(R,V ou B)}$ Le niveau moyen de toute la composante rouge, vert ou bleu .

II.9. Conclusion

Dans ce chapitre, nous avons présenté les techniques utilisées pour détecter les objets en mouvement, représentés dans la méthode de segmentation en se concentrant sur méthode de LPE, et la segmentation par contour actif et nous avons proposé une méthode de différence d'image (avec image de référence, sans l'image de référence) et utilisé la méthode mcm pour détecter les contours des régions en mouvement.

On a proposé une méthode d'Otsu pour obtenir l'image binarisée dans cette méthode le seuillage consiste à séparer les pixels d'une image en deux classes C_1 (fond) donner 0, C_2 (objet) donner 1, à partir d'un seuil t , la méthode d'Otsu est l'une des techniques efficaces qui fournissent des résultats de segmentation supérieurs pour une image ordinaire.

Chapitre III :

Suivi d'objets en mouvement

III.1. Introduction

Après avoir détecté les objets dans une séquence d'image, nous nous intéressons dans ce chapitre à la phase de suivi.

Le suivi de cibles mobiles est un sujet de recherche important dans le domaine la surveillance.

Le suivi de plusieurs objets dans une séquence d'images peut s'avérer difficile selon les scènes complexes qu'il traite, cette complexité réside là où les objets peuvent avoir des tailles différentes. Ces objets peuvent être rigides ou non-rigides et s'occulter les uns les autres de ce fait, des méthodes de suivi sont apparues.

Un bref aperçu des algorithmes de suivi sera donc présenté, ainsi que les modèles de mouvement les plus utilisés.

III.2.Méthodes de suivi [18]

Les méthodes existantes pour le suivi simultané de plusieurs objets dans une séquence d'images impliquent généralement que des caractéristiques telles que des histogrammes couleurs, des mesures de corrélation, des estimations de vitesse, ou des mesures de distance entre régions puissent être utilisées pour suivre chaque objet.

Plusieurs classifications des méthodes de suivi d'objets ont été proposées dans la littérature qui peut être divisées en trois principales catégories :

1. Suivi par appariement de détection ("detect-before-track") ;
2. Suivi par segmentation dynamique ;
3. Suivi déterministe par détection séquentielle.

III.2.1. Suivi par appariement de détections [19]

Le suivi se fait alors en assignant les observations aux pistes en cours d'estimation. Ces méthodes, dites "detect-before-track" sont très populaires en pistage sonar et radar. Elles peuvent être déterministes ou probabilistes.

a. Méthodes déterministes

Le principe des méthodes déterministes est d'associer les observations aux pistes en cours de suivi en minimisant une distance calculée sur certaines caractéristiques de l'objet.

Les caractéristiques des objets couramment utilisées sont l'apparence peuvent être des densités (histogrammes de couleur ou de contour), une carte de contours (contour ouvert ou fermé de l'objet) ou une combinaison de ces modèles.

b. Méthodes probabilistes

Les données perçues par les capteurs vidéo contiennent toujours du bruit. De plus, les mouvements d'objet peuvent subir des perturbations aléatoires, (par exemple : des manœuvres de véhicules).

Les méthodes de correspondance statistique résolvent ces problèmes de suivi en ajoutant une incertitude au modèle de l'objet et aux modèles des observations. Les méthodes de correspondance statistique utilisent l'approche de l'espace d'état pour modéliser les propriétés de l'objet telles que la position, la vitesse et l'accélération.

Lors du suivi, l'état de l'objet est estimé par un modèle dynamique de transition, mis à jour et corrigé au cours du suivi en prenant des mesures de l'image.

Parmi les méthodes d'estimation de l'état dans le contexte du suivi des points, nous citons le filtre de *Kalman* et le filtre à particules. Mais il convient de noter que ces méthodes peuvent être utilisées en général pour estimer l'état de n'importe quel système variant dans le temps.

1) Segmentation dynamique [20]

Les méthodes de suivi par segmentation dynamique sont utilisées lorsque l'on souhaite extraire la silhouette de la cible à chaque instant, et ce sans connaissance a priori sur sa forme. Ces approches reposent sur une succession de segmentations.

Elles font généralement évoluer le contour de l'objet à l'instant précédent jusqu'à sa nouvelle position à l'instant courant.

2) Suivi déterministe de fenêtre englobante par détection séquentielle [19]

Une image est une boîte (en général un rectangle mais parfois aussi une ellipse) entourant ou à l'intérieur de l'objet à suivre. Il s'agit en fait d'une petite portion de l'image. Les techniques de suivi considérées dans cette section sont basées sur la conservation de l'apparence (généralement couleur et/ou luminance) de l'objet pendant au moins deux instants consécutifs. On distingue les algorithmes faisant l'hypothèse de conservation de l'apparence localement (en chaque point de l'objet).

III.3.Modèle Proposé pour suivi d'objet

Notre modèle proposé pour le suivi d'objet dans séquence d'images se compose en trois étapes principales comme il est indiqué dans la figure ci-dessous

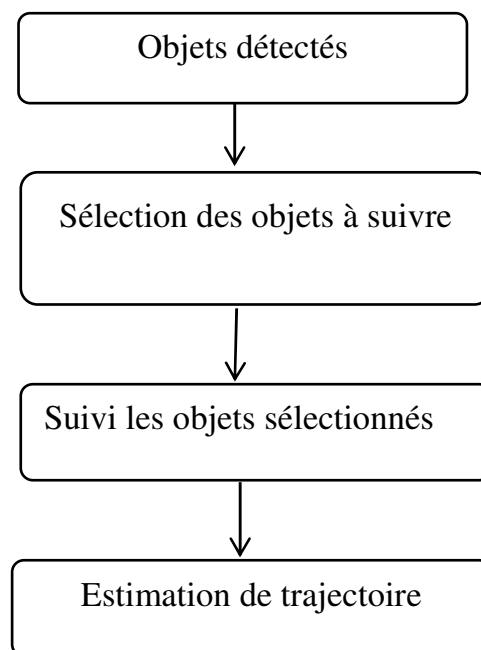


Figure.III.1. Modèle proposé pour suivi d'objet.

- **Sélection des objets à suivre**

La sélection des objets à suivre repose sur une approche utilisant une segmentation binaire et l'étiquetage des composantes connexes. Pour la détermination des composantes connexes, plusieurs algorithmes peuvent être utilisés à savoir

l'étiquetage séquentiel itératif, l'étiquetage séquentiel avec correspondance entre points, l'étiquetage par un parcours en profondeur, etc.

- **Suivi de l'objet sélectionné [44]**

Dans ce travail, nous aborderons la représentation de la forme points dans le suivi des objets en mouvement, où chaque objet est représenté par un point qui est son centre de gravité. La trajectoire d'objet en mouvement dans ce cas est l'interpolation entre les centres de gravité de l'objet détecté dans chaque trame. L'ensemble des centres de gravité mesurés permet de tracer la trajectoire de l'objet mobile.



Figure.III.2. L'ensemble des centres de gravité

- **Estimation de trajectoire [19 ,20]**

Lorsqu'on segmente un objet dans l'image, ou lorsqu'on mesure sa position il y a toujours lieu aux erreurs.

L'objectif est alors de calculer la trajectoire de l'objet en diminuant ces erreurs. Il s'agit alors de l'estimation et de la correction de la trajectoire de l'objet en mouvement.

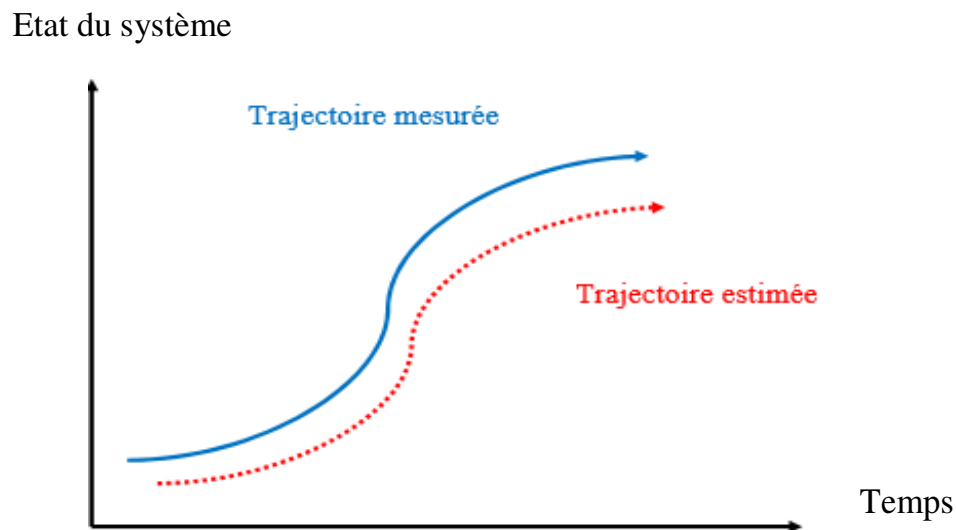


Figure.III.3. Exemple de trajectoire mesurée et estimée.

L'estimation de l'état en cours d'un système dynamique peut aussi être appelée 'Filtrage', le terme filtrage est utilisé car le processus d'obtention de la meilleure estimée à partir de données bruitées consiste à éliminer ou à filtrer ce bruit. L'état du filtre peut être composé de la position du point, ainsi que d'éventuelles informations complémentaires telles que sa vitesse ou son modèle d'intensité.

L'estimation du nouvel état est divisée en deux étapes majeures. Premièrement, une prédiction du nouvel état est obtenue à partir du modèle dynamique.

Ce modèle peut être défini a priori, ou déduit d'une étape préalable d'apprentissage.

Ensuite, la seconde étape consiste à prendre en compte une mesure de l'état à partir de la séquence d'images. Cette mesure, qui se révèle souvent bruitée ou incomplète, permet de corriger la prédiction.

Les filtres stochastiques proposent des solutions pour estimer l'état d'un système conditionnellement à l'ensemble des mesures passées.

Parmi ces filtres, on propose d'utiliser le filtre de *Kalman*.

Les deux idées qui motivent l'utilisation d'un filtre de *Kalman* sur les paramètres du mouvement sont :

- Accumuler les mesures des paramètres de mouvement d'une région au cours du temps pour affiner les estimations ;
- Obtenir une bonne prédiction du mouvement nécessaire pour prédire la segmentation avant ajustement et pour initialiser l'estimation de mouvement.

III.4. Le filtre kalman Caractéristiques d'un bon algorithme de suivi [45,21]

Il ressort des techniques de suivi présentées précédemment des points importants permettant de définir la qualité d'un algorithme de suivi.

Les caractéristiques d'un bon algorithme de suivi sont les suivantes :

- La méthode doit être capable d'initialiser automatiquement les cibles et doit gérer les arrêts et les sorties du champ de la caméra ;
- Elle doit de plus être robuste aux changements d'illumination et aux éventuels changements de topologie ;
- La topologie n'est importante que dans le cas où une segmentation de la cible est exigée ;
- Enfin, la méthode doit permettre de continuer à suivre la cible même en cas d'occultations partielles ou totales par un autre objet ou par le fond.

La première caractéristique ne peut être obtenue qu'avec l'ajout d'une méthode de détection des objets, des observations étant apportées à chaque instant ou à des instants espacés par un faible pas de temps.

Les changements d'illumination ou de photométrie ne sont bien pris en compte que par les méthodes utilisant des histogrammes ou des mélanges de gaussiennes sur l'intensité ou la couleur. Ces distributions doivent régulièrement être mises à jour.

III.4.1. Définition du filtre kalman[48,49]

Le filtre de *Kalman*, a été présenté en 1960 par '*Rudolph E. Kalman*', mathématicien américain né en 1930. Il s'agit d'un estimateur optimal de processus aléatoires.

Partant de l'hypothèse que le bruit de mesure suit une distribution gaussienne, le filtre de *Kalman* est un algorithme récursif en deux étapes, la prédiction et la

correction de l'état. L'étape de prédiction est assurée par le modèle de mouvement linéaire calculé à l'instant précédent $M_t - 1$.

L'étape de correction corrige la prédiction d'état en utilisant l'écart entre l'observation prédite (modèle) et l'observation courante.

Le filtre de *Kalman* a été largement utilisé dans les algorithmes de suivi d'objets est utilisé en trajectographie pour estimer une trajectoire comme le cas de Poursuite de cibles (avions, missiles, radars...).

Dans le processus du suivi, le filtre de *Kalman* 'regarde' un objet quand il se déplace, c'est à dire il prend les informations sur l'état de l'objet à ce moment-là. Puis il utilise ces informations pour prédire où se trouve l'objet dans la prochaine trame.

La position de l'objet est corrigée ensuite en considérant la prédiction et aussi l'observation.

III.4.2. Principe du filtre de Kalman [47]

Le Principe du filtre de *Kalman* donne l'estimation d'un processus en utilisant un système à retour sous la forme de mesures (bruités).

De cette manière, les équations du filtre de *Kalman* se regroupent en deux types d'équations :

1. Les équations de prédiction (extrapolation).
2. Les équations de mise à jour (correction).

Les équations d'extrapolation sont responsables de propager les estimées de l'état présent et les covariances d'erreur pour obtenir les estimées à priori de la prochaine étape. Alors que les équations de correction sont responsables du retour, c'est à dire, introduire une nouvelle mesure avec l'estimée à priori afin d'obtenir une estimée à posteriori améliorée.

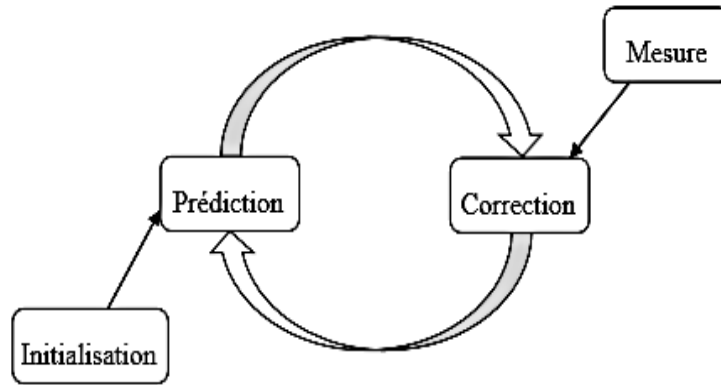


Figure.III.4. Fonctionnement du filtre de Kalman.

- **Calcul de l'estimateur de Kalman**[48]

On considère un système dynamique linéaire à temps discret défini par un vecteur d'équations aux différences, entaché d'un bruit blanc gaussien. Plus précisément, il existe un système de temps discret et son état à l'instant n est donné par le vecteur \mathbf{X}_n . L'état de la prochaine étape $n + 1$ est donné par :

$$\mathbf{X}_{n+1} = \mathbf{F}_{n+1,n}\mathbf{X}_n + \mathbf{W}_{n+1} \quad (\text{III. 1})$$

Où $\mathbf{F}_{n+1,n}$ est la matrice de transition d'état caractérisant l'évolution du système. \mathbf{W}_{n+1} Est un bruit blanc gaussien à moyenne nulle et de matrice de covariance \mathbf{Q}_{n+1} . L'équation de mesure est donnée par :

$$\mathbf{Z}_{n+1} = \mathbf{H}_{n+1}\mathbf{X}_{n+1} + \mathbf{V}_{n+1} \quad (\text{III. 2})$$

Où \mathbf{Z}_{n+1} est le vecteur de mesure. \mathbf{H}_{n+1} Est la matrice de transformation de l'état vers l'espace des mesures.

Le bruit de mesure est modélisé par un vecteur \mathbf{V}_{n+1} , sa distribution est aussi une gaussienne de moyenne nulle et de covariance \mathbf{R}_{n+1} .

Dans l'équation (III.2), le vecteur de mesure \mathbf{Z}_{n+1} ne dépend que du vecteur d'état actuel \mathbf{X}_{n+1} et le vecteur de bruit \mathbf{V}_{n+1} est indépendant du bruit \mathbf{W}_{n+1} .

Le filtre de *Kalman* calcule l'erreur d'estimation quadratique moyenne minimale de l'état \mathbf{X}_k étant donné les mesures $\mathbf{Z}_1 \dots \dots \mathbf{Z}_K$.

- **Prédiction du vecteur d'état et des mesures**

Une prédiction de l'état futur des variables que l'on cherche à estimer est tout d'abord réalisée, à partir de l'état précédent. La prédiction de l'état et sa matrice de covariance au temps t_n sont données par le modèle dynamique :

$$\hat{\mathbf{X}}_n^- = \mathbf{F}_{n,n-1} \hat{\mathbf{X}}_{n-1}^- \quad (\text{III. 3})$$

$$\mathbf{P}_n^- = \mathbf{F}_{n,n-1} \mathbf{P}_{n-1} \mathbf{F}_{n,n-1}^T \quad (\text{III. 4})$$

Où la matrice de covariance \mathbf{P} est de dimensions $\mathbf{R}^{D_x \times D_y}$ et le deuxième terme dans l'équation correspond à la covariance du vecteur de perturbation de l'état.

- **Correction de d'état[48]**

Pour la mise à jour du vecteur d'état estimé $\hat{\mathbf{x}}$ et sa covariance \mathbf{P} , il est nécessaire d'obtenir le gain du filtre \mathbf{G}_n qui dépend directement des matrices de covariance obtenues pour l'état prédit,

$$\mathbf{G}_n = \mathbf{P}_n^- \mathbf{H}_n^T [\mathbf{H}_n \mathbf{P}_n^- \mathbf{H}_n^T + \mathbf{R}]^{-1} \quad (\text{III. 5})$$

$$\hat{\mathbf{X}}_n = \hat{\mathbf{X}}_n^- + \mathbf{G}_n (\mathbf{Z}_n - \mathbf{H}_n \hat{\mathbf{X}}_n^-) \quad (\text{III. 6})$$

$$\mathbf{P}_n = (\mathbf{1} - \mathbf{G}_n \mathbf{H}_n) \mathbf{P}_n^- \quad (\text{III. 7})$$

Dans cette étape de correction, les observations de l'instant courant sont utilisées pour corriger l'état prédit dans le but d'obtenir une estimation plus précise.

Le filtre de *Kalman* est appliqué en temps réel, parce que ce filtre ne nécessite pas toutes les données passées pour produire une estimation à l'instant courant, c.-à-d. il ne nécessite donc pas de mise en mémoire et de retraitement des données.

III.4.3. Algorithme de Filtre de Kalman [48,49]

On résume les étapes de suivi par filtre de *Kalman* suivant :

1. Initialisation : $\hat{\mathbf{X}}_0$ la position initiale de l'objet.

$$\hat{\mathbf{X}}_0 = E[\mathbf{X}_0] \quad (\text{III. 8})$$

$$\mathbf{P}_0 = [(\mathbf{X}_0 - E[\mathbf{X}_0])(\mathbf{X}_0 - E[\mathbf{X}_0])^T] \quad (\text{III. 9})$$

2. Prédiction :

$$\hat{\mathbf{X}}_n^- = F_{n,n-1} \hat{\mathbf{X}}_{n-1} \quad (\text{III. 10})$$

$$\mathbf{P}_n^- = F_{n,n-1} \mathbf{P}_{n-1} F_{n,n-1}^T + \mathbf{Q}_n \quad (\text{III. 11})$$

3. Correction :

$$\mathbf{G}_n = \mathbf{P}_n^- \mathbf{H}_n^T [\mathbf{H}_n \mathbf{P}_n^- \mathbf{H}_n^T + \mathbf{R}]^{-1} \quad (\text{III. 12})$$

$$\hat{\mathbf{X}}_n = \hat{\mathbf{X}}_n^- + \mathbf{G}_n (\mathbf{Z}_n - \mathbf{H}_n \hat{\mathbf{X}}_n^-) \quad (\text{III. 13})$$

$$\mathbf{P}_n = (\mathbf{1} - \mathbf{G}_n \mathbf{H}_n) \mathbf{P}_n^- \quad (\text{III. 14})$$

Retourner à l'étape de prédiction pour l'itération suivante.

III.5. Méthode de travail du filtre dans le suivi

Dans cette partie on va suivre les objets qui sont en mouvement et associé à chaque objet une trajectoire propre lui, cette trajectoire est estimée par le filtre de *Kalman*.

Le filtre sert à prédire l'endroit de la piste (trajectoire) dans chaque image et déterminer la probabilité de chaque détection étant assignée à chaque piste.

Dans n'importe quelle image donnée, certaines détections peuvent être cédées aux pistes, tandis que les autres détections et les pistes peuvent rester non attribuées.

Les voies affectées sont mises à jour en utilisant les détections correspondantes. Les titres des objets non attribués sont marqués invisibles. Une détection non attribuée commence une nouvelle piste.

En cas d'occultation il peut prédire la position de l'objet en mouvement.

Chaque piste conserve le nombre d'images consécutives, où il est resté non attribuée. Si le nombre dépasse un seuil déterminé, l'exemple suppose que l'objet quitte le champ de vision et il supprime la piste.

III.6. Exemple d'applications du filtre de kalman [45]

La figure (5) présente un exemple d'application du filtre de *Kalman* sur la séquence scène routière.

- ❖ Etape 1 : calcul du centre de gravité de l'objet détecté. Le centre de gravité étant le point d'intersection de tous les plans qui divisent le corps en deux parties de poids égaux ;
- ❖ Etape 2 : tracer la trajectoire mesurée sur 5 trames ;
- ❖ Etape 3 : appliquer le suivi par filtre de *Kalman* avec comme composantes du vecteur d'état les coordonnées selon l'axe x et y du centre de gravité calculé. Le résultat est un tracé conjoint des trajectoires mesurée et estimée.

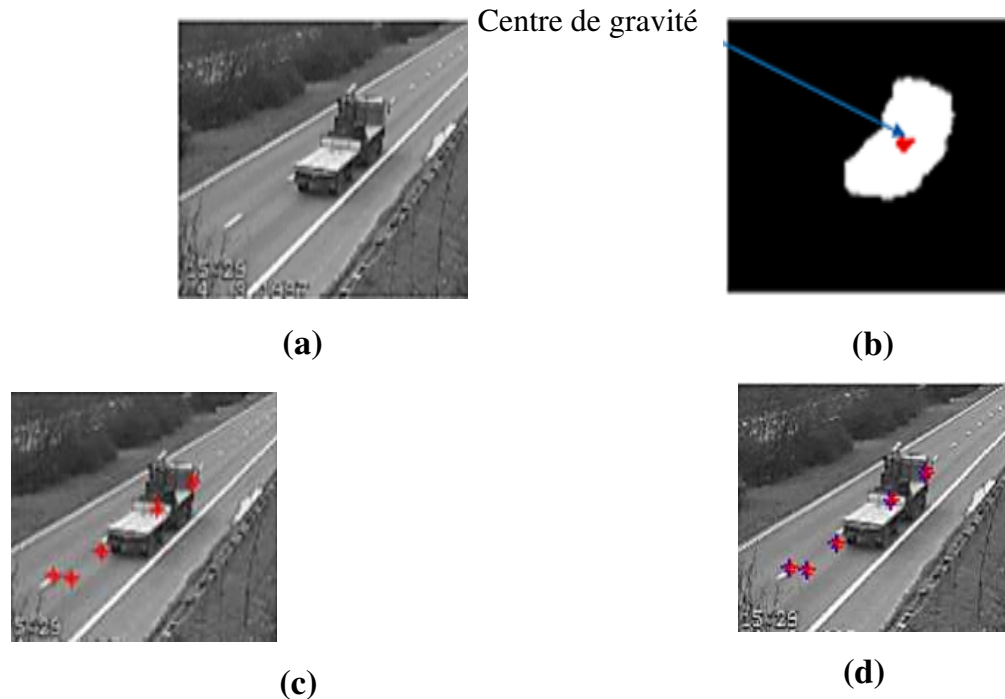


Figure.III.5. (a)Trame à l'instant t, (b) l'objet en mouvement détecté avec son centre de gravité, (c) trajectoire mesurée sur 5 trames, (d) représentation conjointe des trajectoires mesurée et estimée sur l'image.

Le filtre de *Kalman* peut aussi appliquer dans :

- La détection d'une sinusoïde noyée dans du bruit (*Kalman* étendu) ;
- L'estimation de position en navigation dans le cas continu et le cas discret ;
- L'estimation de trajectoires ou de grandeurs noyées dans du bruit ;
- La fusion de données.

III.7. Détection des Blobs [50]

Le modèle de soustraction de l'arrière-plan fournit les pixels détectés comme premier plan. Dans le modèle de détection des blobs, les pixels sont regroupés, dans l'image courante, en utilisant un algorithme de détection de contour.

Cet algorithme regroupe les pixels individuels en classes déconnectées, et trouve ensuite les contours entourant chaque classe. Chaque classe est marquée comme une blob candidate (CB).

Ces CB sont ensuite vérifiées en fonction de leur taille et les petits blobs sont retirées de l'algorithme afin de réduire les fausses détections. Le résultat de l'examen du module de détection des blobs (des taches) est illustré à la figure 6.

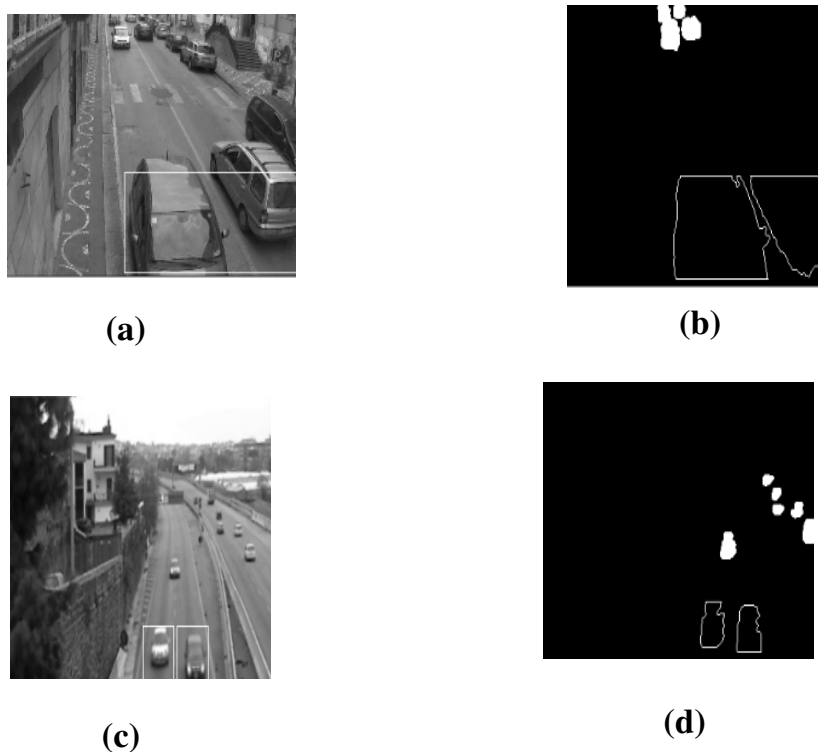


Figure.III.6.Gauche (a,c) : image vidéo originale, droite(b,d) : Le résultat du module de détection des blobs (des taches).

III.8. L'analyse de *Blob* [50,51]

Le module d'analyse des blobs est l'une des étapes les plus importantes du processus. Comme le montre la figure 6, ce module reçoit les CB avec leur position, comme entrées, et fournit le blob d'information dans la trame vidéo actuelle.

analyse Blob



Figure.III.7. Module d'analyse blob

Le module d'analyse de blob identifie quel CB dans la trame actuelle appartient au même véhicule. Par exemple :

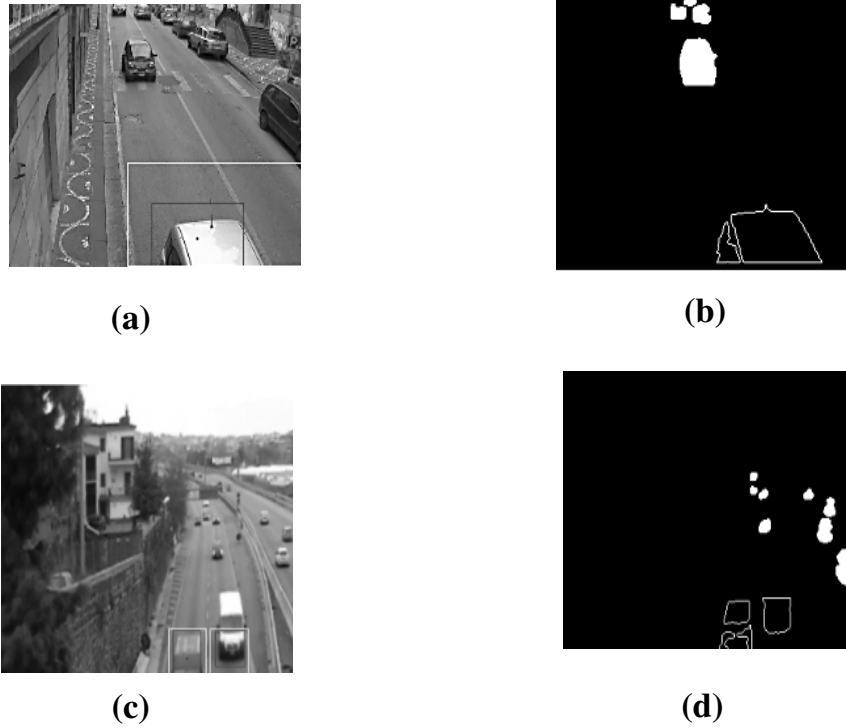


Figure. III.8. Gauche (a,c) : image vidéo originale, droite (b,d) : Le résultat du module d'analyse des blobs.

$$\rho^n = |(W_{B_c} \times h_{B_n}) - (W_{B_n} \times h_{B_c})| < \gamma \quad (\text{III. 15})$$

Où n est le numéro de la trame précédente, B_c et B_n sont respectivement le blob de la trame actuelle et de la trame précédente et δ et γ sont les seuils.

Si $dist^n$ est minimal et que les conditions $dist^n < \delta$ et $\rho^n < \delta$ sont remplies, l'objet dans la trame actuelle est considéré comme un objet dans la trame précédente et le blob qui l'identifie se voit attribuer le label de similitude.

III.9. Conclusion

Dans ce chapitre, on a présenté des algorithmes qui concernent le suivi d'objets en mouvement dans une séquence d'image. Particulièrement la méthode du filtre de *Kalman*.

Le filtre de *Kalman* est un outil performant qui permet de résoudre les problèmes d'estimation d'état instantané d'un système dynamique linéaire perturbé et noyé dans du bruit. Ceci peut être projeté au suivi d'objets en mouvement dans une scène.

Les principales caractéristiques du filtre de *Kalman* sont la prédiction de l'emplacement futur de l'objet, la réduction du bruit, la capacité de suivre un ou plusieurs objets.

Chapitre IV :
Implémentation Pratique et
Résultats des Simulations

VI.1. Introduction

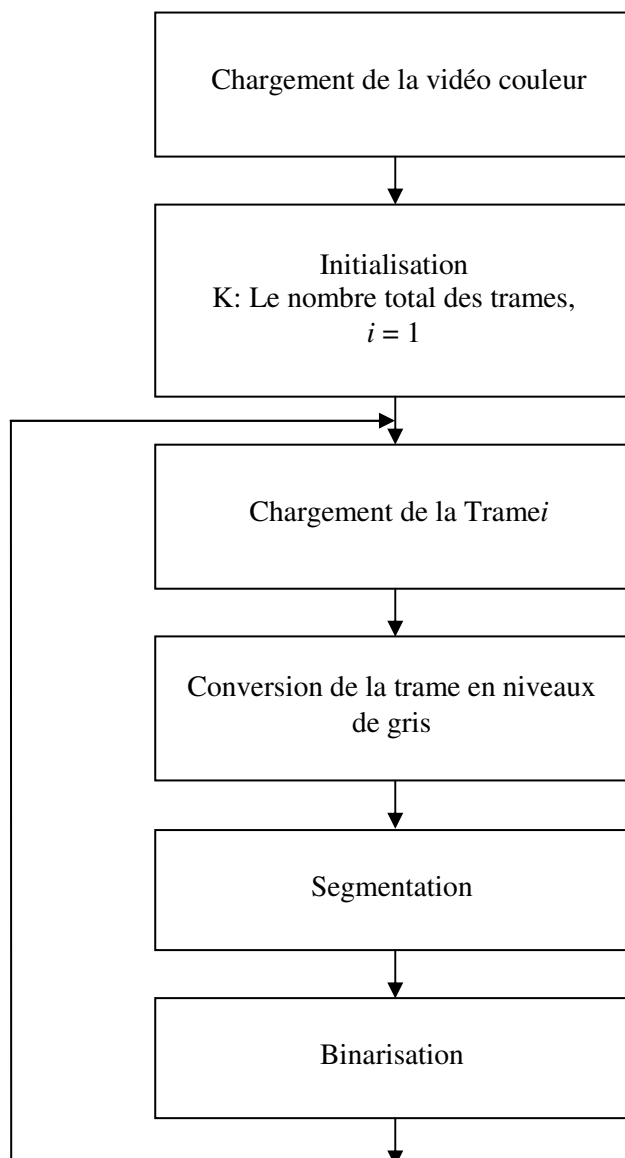
Ce chapitre présente les différentes expérimentations réalisées et les résultats des tests effectués pour la détection et le suivi d'objets en mouvement dans une séquence d'image.

Nous implémentons les différents algorithmes de la détection automatique et le suivi du mouvement des objets dans une vidéo issue d'une caméra fixe.

Dans une première partie nous présenterons les résultats de la détection du mouvement en utilisant la segmentation par la méthode d'*OTSU*.

Le suivi sera par la suite présenté dans une deuxième partie où il est effectué en utilisant le filtrage de *Kalman*.

VI.2. Algorithme proposé pour la détection et le suivi d'objet dans une séquence d'images



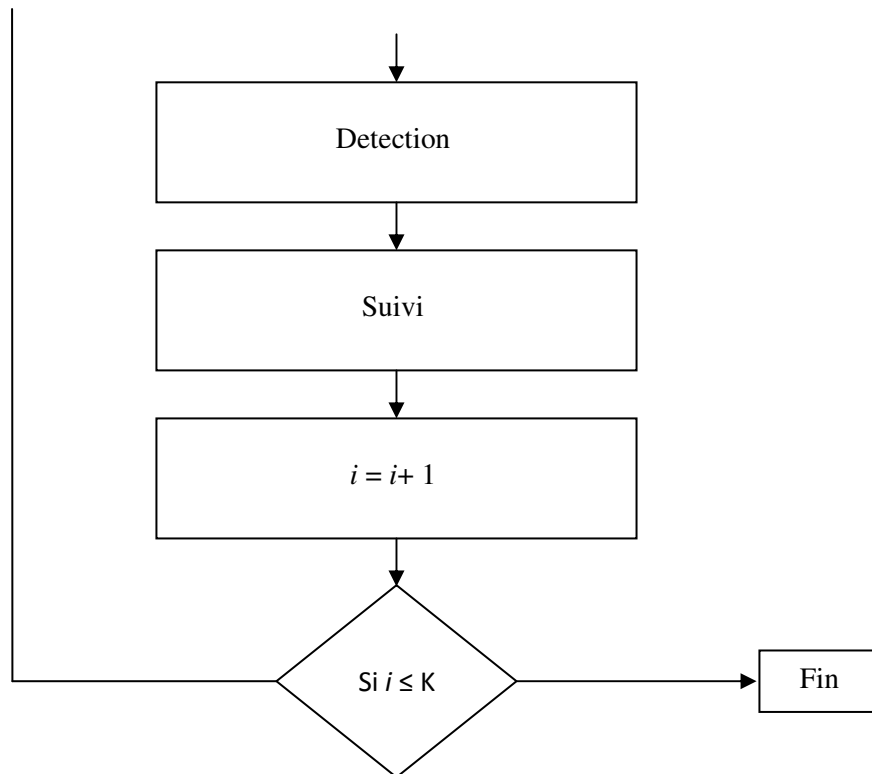


Figure.VI.1. Algorithme proposé.

VI.2.1. Segmentation d'image

Dans cette sous-section nous montrons comment une image est segmentée pour créer une image binaire résultante où autrement un masque binaire par l'utilisation des techniques de segmentation présentées dans le chapitre 2.

VI.2.1.1. Segmentation d'images par la méthode du contour actif

L'algorithme du contour actif est appliqué sur des images à niveau de gris, il est présenté par l'organigramme de la figure ci-dessous.

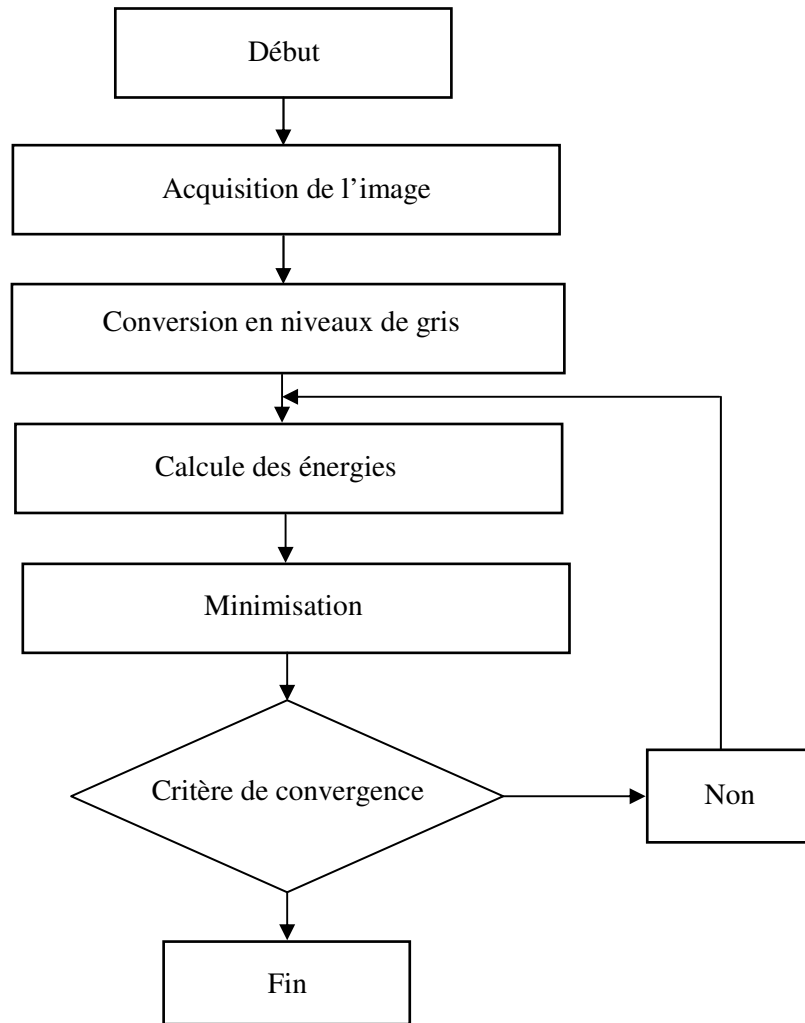


Figure.VI. 2.L'algorithme du contour actif.

a. Acquisition de l'image

```
I = imread('football.png');  
imshow(I)  
title('Image originale');
```

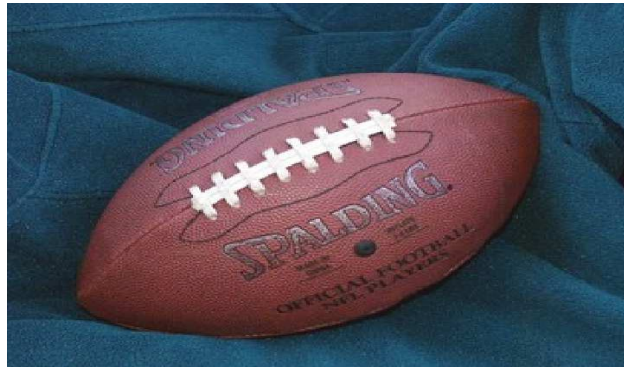


Figure IV. 3. Image couleur.

b. Conversion de l'image en niveau de gris

```
myrgb=imread('football.png');  
imshow(myrgb)  
mygray=round(mean(myrgb, 3))/255;  
figure,imshow(mygray);
```



Figure.VI.4. Image en niveau de gris.

c. Spécification du masque initial

```
mask = zeros(size(g));  
mask(25:end-25,25:end-25) = 1;  
figure, imshow(mask);  
title('Localisation du contour initial');
```



Figure.VI .5. Image du masque.

Segment the image using the default method and 300 iterations.

d. Segmentation de l'image

```
bw = activecontour(I,mask,300);  
figure;  
imshow(bw);  
title('Image segmentée');
```



Figure.VI.6. Image segmentée.

e. Seuil d'image global en utilisant la méthode d'Otsu

1. Algorithme Otsu

Pour la méthode d'Otsu, l'histogramme cherche à maximiser la variance inter-classe entre « l'objet » et « Le fond ».

a. Méthode 1

```
Lire image  
[M N] =taille d'image ;  
Level=gray thresh (image);  
BW=im2bw (image, level);
```

a.1. Programme

```
Image=imread ('Image.jpg');  
Level=gray thresh (Image);  
BW=im2bw (Image, level);  
Figure;  
Im show(BW) ;  
Title('Imagebinarisée avec la méthode 1 d'Otsu');
```

b. Méthode 2

```
Lire image  
[M N]=Taille de l'image (Image) ;  
Calcule de l'histogramme  
Histogramme(Image (i,j)+1)=Histogramme(Image(i , j)+1)+1 ;  
Probabilité de chaque Niveau de Gris  
Proba=Histogramme /NB Total ;  
Pour i=1 :255  
Calcul de la Probabilité des classes  
Proba1=Proba(1 :i) ;  
Proba2=Proba (i+1:256) ;
```

```
P1=sum(Proba1) ;
P2=sum(Proba2);
Calcul de la moyenne des classes
Moy1=sum(n1.*Proba1)/P1 ;
Moy2=sum(n1.*Proba2)/P2;
Calcul de la variance des classes
Var1= sum (((n1-Moy1). ^2).*Proba1);
Var2= sum (((n2-Moy2). ^2).*Proba2)
Variance Intra-Classe(i)=Var1+Var2 ;
Détermination du seuil
[Val, Indice]=min (Variance Intra-Classe (1 :255)) ;
level= (Indice - 1)/255
La binarisation
BW1=im2bw(Image, Level) ;
BW1=BW1*255 ;
Affiche(BW)
```

b.1. Programme

On applique ce programme pour obtenir une image binarisée

```
Image=imread ('Image.jpg ');
[Colonne Ligne] =size(Image);
NB Total=Colonne*Ligne ;
Calcul de l'histogramme
Histogramme (Image (i, j) +1) =Histogramme(Image(i,j)+1)+1 ;
Proba=Histogramme/NB Total ;
For i=1 :255
Calculi de probabilité des classes
Proba1=Proba(1 :i) ;
Proba2=Proba(i+1:256);
P1=sum(Proba1);
P2= sum (Proba2);
```

```
Calculai la moyenne des classes
n1=0 : i -1 ; n2= i : 255 ;
Moy1=sum (n1.*Proba1) /P1;
Moy2=sum (n2.*Proba2) /P2;
Calculai la variance des classes
Var1=sum (((n1-Moy1).^2).*Proba1) ;
Var2=sum (((n2-Moy2). ^2).*Proba2);
VarianceIntra-Classe(i)=Var1+Var2 ;
End
Détermination du seuil
[Val, Indice]=min (VarianceIntra -Classe (1:255)) ;
Level= (Indice - 1)/255; BW1=im2bw (Image, level);
Binarisation
BW1=im2bw (Image,level)
BW1=BW1*255 ;
Figure : Imshow (BW1) ;
Title('Image binarisée avec la méthode 2ème d'Otsu') ;
```

- On applique ce programme pour obtenir l'image binarisée.

```
Level = gray thresh (g);
BW = im2bw(g,level);
imshow(BW)
```

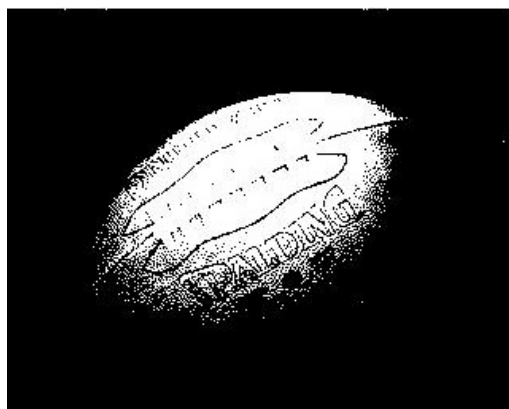


Figure.VI.7. Image binarisée avec la methode d'Otsu

IV.3. Détection d'objet par segmentation

IV.3.1. Segmentation de la circulation

Dans cette section nous allons implémenter un programme qui permet la détection d'une voiture dans la circulation en basant sur la segmentation. Le programme que nous avons implémenté permet la détection de voitures de couleurs claires.

Lorsque nous travaillons avec des données de type vidéo, il est préférable de sélectionner une trame à partir de la vidéo et de développer par la suite notre algorithme on basons sur cette dernière. L'algorithme, peut-être par la suite appliqué pour toutes les autres trames de la vidéo.

Nous examinons une trame qui inclue les voitures de couleur foncée et de couleurs claire ensemble.

Pour le cas des trames de la vidéo de la circulation qui contient plusieurs structures, il est nécessaire de simplifier l'image tant que possible avant de passer à la détection d'objet désiré.

Si nous désirons, à titre d'exemple de, détecté les voitures de couleurs claires, dans ce cas nous supprimons de l'image toutes les voitures et les objets de couleurs foncés. Pour cela, nous devons adoptés des techniques qui nous permettent de supprimer tous les objets indésirables.

La fonction MATLAB **imextendedmax** permet de supprimer les voitures de couleurs foncés. Elle permet d'avoir une image binaire dans laquelle les régions dont la valeur de l'intensité est supérieure à un seuil spécifique sont identifiées.

Les objets dans l'image avec des valeurs de pixels inférieures au seuil deviennent l'arrière-plan.

Pour éliminer les voitures de couleurs foncées, nous devons déterminer la valeur moyenne des pixels pour cet objet dans l'image.

La fonction MATLAB **rgb2gray** est utilisée pour convertir la vidéo originale du RGB aux niveaux de gris.

L'outil de région de pixel dans la fonction **implay** peut être utilisé pour voir les valeurs des pixels.

Nous spécifions la valeur moyenne des pixels ou une valeur légèrement plus haute comme un seuil dans la fonction **imextendedmax**. Dans notre cas nous spécifions la valeur à 50.

```
darkCarValue = 50;
darkCar = rgb2gray(read(trafficObj, 71));
noDarkCar = imextendedmax(darkCar, darkCarValue);
Imshow(darkCar)
Figure
Imshow(noDarkCar)
```



(a)



(b)

Figure.VI.8.Résultats de la détection par segmentation de la circulation,

(a) : l'image original, (b) : l'image traitée.

Dans l'image traitée, la plupart des voitures de couleurs foncées ont été éliminées mais beaucoup d'autres objets ont restés, notons à titre d'exemple les balises de la route.

Le traitement par le maxima régionaux ne va pas éliminer les balises car les valeurs de leurs pixels sont au-dessous du seuil.

Pour éliminer ces objets, nous utilisons la fonction morphologique **imopen** qui utilise le traitement morphologique pour éliminer les petits objets à partir de l'image binaire et préserve les petits objets.

Lorsque nous utilisons le traitement morphologique, nous devons décider sur la taille et la forme de l'élément structurant utilisé dans l'opération.

Comme les marques de voie sont des objets longs et fins, utilisez un élément structurant en forme de disque avec un rayon correspondant à la largeur des marques de voie.

Nous pouvons utiliser l'outil de région de pixels dans **implay** pour estimer la largeur de ces objets. Pour notre cas nous définissons la valeur 2.



Figure.VI.9. Représente le résultat après l'utilisation du traitement morphologique.

```
edisk = strel('disk',2);  
noSmallStructures = imopen(noDarkCar, sedisk);  
imshow(noSmallStructures)
```

Nous utilisons la fonction **regionprops** pour trouver le Cent roide des objets dans **noSmallStructures**. Ces objets doivent être seulement les voitures de couleurs claires. Nous utilisons cette information pour positionner la balise sur les voitures de couleurs claires dans la vidéo originale.

- **L'application de l'algorithme sur la vidéo**

L'application de marquage de voiture traite la vidéo une image à la fois dans une boucle. (Parce qu'une vidéo typique contient un grand nombre d'images, il faudrait beaucoup de mémoire pour lire et traiter toutes les images à la fois.) Une petite vidéo (comme celle de cet exemple) pourrait être traitée en même temps, et il existe de nombreuses fonctions qui offrent cette capacité. Pour un traitement plus rapide, pré-allouez la mémoire utilisée pour stocker la vidéo traitée.

```
n frames = get (traffic Obj, 'Number Of Frames');
I = read (traffic Obj, 1);
tagged Cars = zeros([size(I,1) size(I,2) 3 n frames], class(I));
for k = 1: n frames
    single Frame = read (traffic Obj, k);
    I = rgb2gray(single Frame);
    NoDarkCars = imextendedmax(I, darkCarValue);
    noSmallStructures = imopen(noDarkCars, sedisk);
    noSmallStructures = bwareaopen(noSmallStructures, 150);
    taggedCars(:,:,,k) = singleFrame;
    stats = regionprops(noSmallStructures, {'Centroid','Area'});
    if ~isempty([stats.Area])
        areaArray = [stats.Area];
        [junk,idx] = max(areaArray);
```

```
c = stats(idx).Centroid;
c = floor(fliplr(c));
width = 2;
row = c(1)-width:c(1)+width;
col = c(2)-width:c(2)+width;
taggedCars(row,col,1,k) = 255;
taggedCars(row,col,2,k) = 0;
taggedCars(row,col,3,k) = 0;
end
end
frameRate = get(trafficObj,'FrameRate');
implay(taggedCars,frameRate);displayEndOfDemoMessage(mfilename)
```

```
trafficObj = VideoReader('traffic.mj2')
get(trafficObj)
implay('traffic.mj2');
```

```
DarkCarValue = 50;
DarkCar = rgb2gray (read (trafficObj, 71));
NoDarkCar = imextendedmax(darkCar, darkCarValue);
imshow(darkCar)
Figure, imshow(noDarkCar)
```

```
sedisk = strel('disk',2);  
noSmallStructures = imopen(noDarkCar, sedisk);  
imshow(noSmallStructures)
```

```
N frames = get(traffic Obj, 'Number Of Frames');  
I = read(trafficObj, 1);  
taggedCars = zeros([size(I,1) size(I,2) 3 nframes], class(I));  
for k = 1 : n frames  
    single Frame = read(trafficObj, k);  
    I = rgb2gray(single Frame);  
    noDarkCars = imextendedmax(I, darkCarValue);  
    noSmallStructures = imopen(noDarkCars, sedisk);  
    noSmallStructures = bwareaopen(noSmallStructures, 150);  
    taggedCars(:,:,k) = singleFrame;  
    stats = regionprops(noSmallStructures, {'Centroid','Area'});  
    if ~isempty([stats.Area])  
        areaArray = [stats.Area]; [junk,idx] = max(areaArray);  
        c = stats(idx).Centroid; c = floor(fliplr(c));  
        width = 2;  
        row = c(1)-width:c(1)+width; col = c(2)-width:c(2)+width;  
        taggedCars(row,col,1,k) = 255;taggedCars(row,col,2,k) = 0;  
        taggedCars(row,col,3,k) = 0;  
    end  
end  
Frame Rate = get(traffic Obj,'Frame Rate');imshow(tagged Cars , frame Rate);  
display End Of Demo Message (mfilename)
```

VI.4. Le suivi d'objet par le filtre kalman

VI.4.1. Détection et suivie d'une balle par filtrage de Kalman, détection du premier plan et l'analyse de Blob

Le processus de détection et de suivi par l'utilisation de filtre de *Kalman* est décrit par l'organigramme suivant :

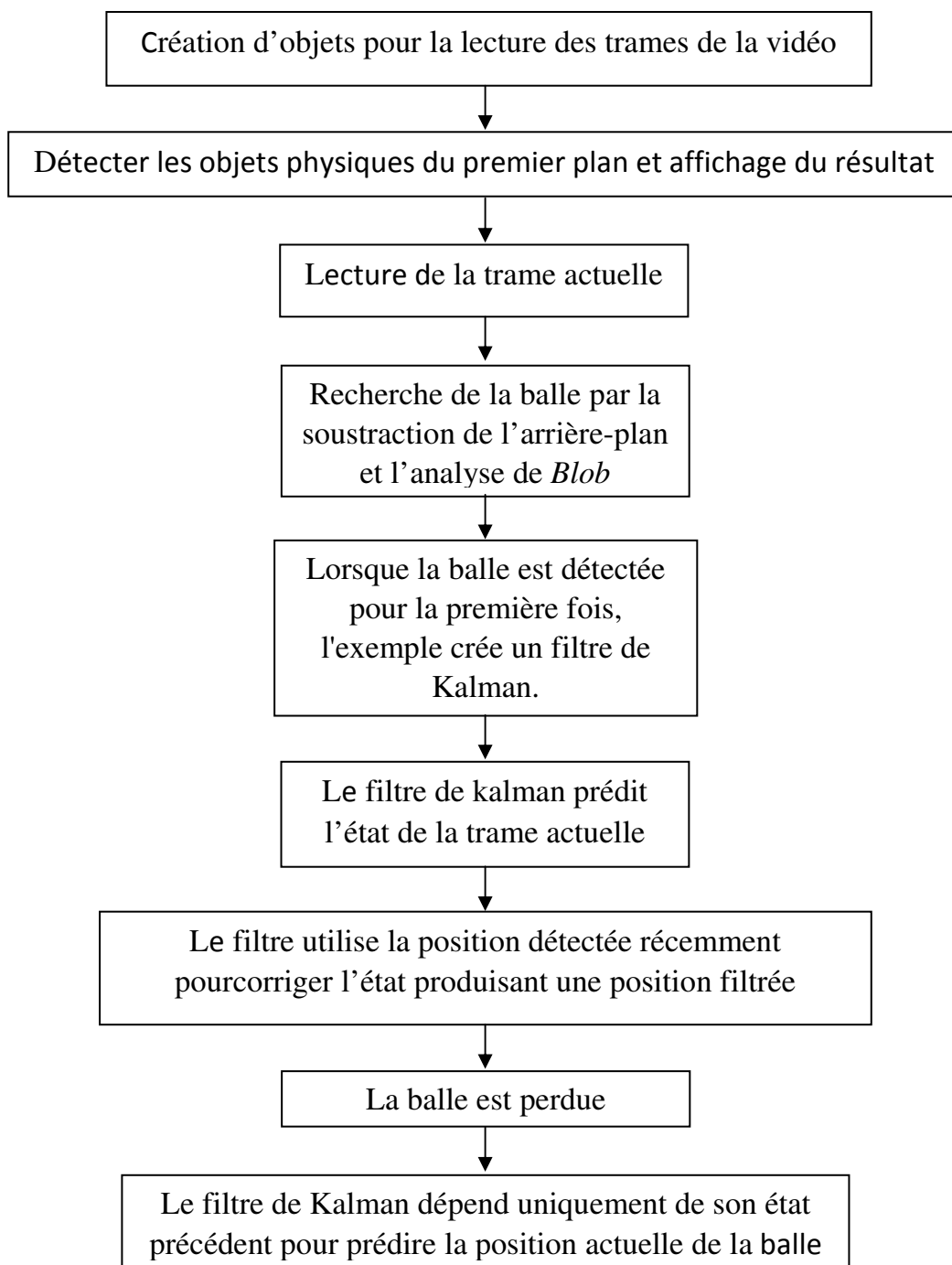
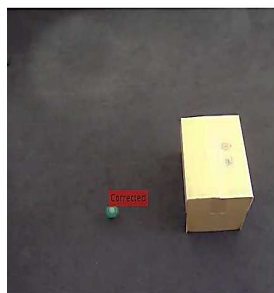


Figure.VI.10.Le processus de détection et de suivi par l'utilisation de filtre de *Kalman*.

```
clear all; clc; close all;
videoReader = VideoReader('singleball.mp4');
videoPlayer = vision.VideoPlayer('Position',[100,100,500,400]);
foregroundDetector = vision.ForegroundDetector('NumTrainingFrames',10,...
'InitialVariance',0.05);
blobAnalyzer = vision.BlobAnalysis('AreaOutputPort',false,...'MinimumBlobArea',70);
kalmanFilter = []; isTrackInitialized = false;
for k=1:45
colorImage = read(videoReader,k);
foregroundMask = step(foregroundDetector,
rgb2gray(im2single(colorImage)));
detectedLocation = step(blobAnalyzer,foregroundMask);
isObjectDetected = size(detectedLocation, 1) > 0;
if ~isTrackInitialized
if isObjectDetected
kalmanFilter = configureKalmanFilter('ConstantAcceleration',...
detectedLocation(1,:), [1 1 1]*1e5, [25, 10, 10], 25);
isTrackInitialized = true;
end
label = ""; circle = zeros(0,3);
else
if is Object Detected
predict (kalman Filter);
tracked Location = correct(kalman Filter, detected Location(1,:));
label = 'Corrected';
else
tracked Location = predict(kalman Filter);
label = 'Predicted';
end
```

```
circle = [tracked Location, 5];  
end  
  
color Image = insert Object Annotation (color Image,'circle',...  
circle, label , 'Color' , 'red');  
step (video Player, color Image);  
pause (0.1);  
end
```



(a)



(b)



(c)

Figure.VI.11. Détection et suivi par le filtre de *Kalman*.

VI.5.Conclusion

En conclusion, nous avons présenté les différents résultats que nous avons obtenus en intégrant les algorithmes de détection développés dans la première partie du mémoire suivis de ceux de poursuite de trajectoire de mouvement présentés dans la deuxième partie.

Le travail que nous avons effectué nous a permis de réaliser par des algorithmes les objectifs de notre recherche qui sont la détection, la segmentation et le suivi d'objets.

Nous suggérons de continuer la recherche dans ce sens pour améliorer encore plus les résultats surtout pour arriver à une dilatation optimale ce qui nous permet de détecter et suivre presque parfaitement.

Conclusion Générale

Dans ce travail, nous avons effectué la détection du mouvement dans une séquence d'images ; la détection et le suivi d'objets en mouvement dans des scènes simples (caméra fixe).

Dans la première partie de ce mémoire on a présenté quelques concepts et définitions de l'image et la vidéo. Dans la deuxième partie nous avons proposé les méthodes de détection d'objets, et dans la troisième partie nous avons proposé la méthode de suivi par le filtre de kalman.

La quatrième partie a présentée l'ensemble des résultats expérimentaux obtenus pour la détection et le suivi d'objets en mouvement.

Dans le cadre de ce mémoire et après une étude bibliographique sur certaines méthodes liées au contexte, on a choisi une approche de détection de mouvement et de suivi des objets rigide comme les voiture ballon de foot...etc.

Les expériences effectuée on présentée l'efficacité des méthodes utilisées.

Perspectives :

En perspectives, on propose :

- L'application des méthodes de détection et de suivi d'objets des multi- scènes (plusieurs camera).
- L'application des méthodes de détection et de suivi sur les séquences à fonds dynamiques (caméras mobiles).
- De chercher une méthode optimale pour la détection et le suivi de plusieurs objets.

Références bibliographiques

- [1] M Brougui, N Boumaraf., 'Détection et suivi d'un objet suspect dans le contexte de vidéosurveillance', Mémoire de Master, Université KasdiMerbah, Ouargla, 2015.
- [2] M Chikh Tahar., Amélioration des images par un modèle de réseau de neurones (Comparaison avec les filtres de base), Mémoire de Master en Informatique, Université Abou-BakrBelkaid Tlemcen, 2011.
- [3] Laouar N, S. Laraba., Détection d'un mouvement dans une séquence vidéo par filtres Morphologiques, mémoire de master, ENP, Alger, 2009.
- [4] Chris Solomon. Toby Breckon., Fundamentals of Digital Image Processing:
A Practical Approach with Examples in Matlab, John Wiley & Sons, 2011.
- [5] Mersaoui I, Fouiri M., Etude sur l'imagerie médicale : prétraitement, segmentation et amélioration des exploitations, Mémoire de Master, Université SAAD DAHLAB De BLIDA, 2018.
- [6] Hadjila F, Bouabdellah R., « Reconnaissance des visages par les réseaux de neurones », Mémoire de PFE, Université de Tlemcen, 2003.
- [7] Devaux F., « Filtrage d'images par réseaux de neurones », 1997.
- [8] Naciri H., Chaoui N., « Conception et Réalisation d'un système automatique d'identification des empreintes digitales », Mémoire de PFE, Université de Tlemcen, 2003.
- [9] Aek. Tahari, Détection des objets en mouvements dans une séquence d'images, Thèse De Magister, ESI (ex INI), Alger, 2002.
- [10] Pierre.Soille., Morphological Image Analysis, principes and application, 2nd edition, springer.
- [11] Milan Sonka , Vaclav Hlavac., Roger Boyle, Image Processing, Analysis, and Machine Vision, 4th Edition, 2015.
- [12] J.-P. Cocquerez , S. Philipp.,1995 , Analyse d'Images : filtrage et segmentation, Masson,

- [13] Kaiwen CHANG., Le Machine learningbased image segmentation Apprentissage artificiel pour la segmentation d'image, Thèse de Doctorat, Préparée à MINES Paris Tech université PSL, 2019.
- [14] Besançon B., « vision par ordinateur en deux et trois dimension », mémoire de master, Edition Eyrolles, 1998.
- [15] G. Burel., ' Introduction au traitement d'images', Edition Hermès Octobre 2001.
- [16] Laurent Vinet., Segmentation et Mise en correspondance de Régions de Paires d'images Stéréoscopiques, thèse de doctorat en science, Université de Paris IX Dauphine, 1991.
- [17] A. Kerioui , M. Djani., « Application des réseaux de neurones à la compression d'images », mémoire de fin d'étude d'ingénieur, Editions 1998.
- [18] A abidi,Mslimani.,suivi d'objets en mouvement dans une séquence des images ,Université Ahmed Draia –Adrar, Mémoire de Masteren informatique.
- [19] S abdelbaki ,YOUellabi ., Détection et suivi d'objets mobiles. Application dans un environnement de foule, mémoire de Master, Université EchahidHamma Lakhdar D'el oued,2015 .
- [20] Aurélie Bugeau., Détection et suivi d'objets en mouvement dans des scenes complexes, application a la surveillance des conducteurs. Thèse de Doctorat, devant l'université de rennes I, 2011.
- [21] Medjahed Fatiha., Détection et Suivi d'Objets en Mouvement Dans Une Séquence d'Images, Mémoire de Magister, Université des Sciences et de la Technologie d'Oran U. S. T. O. « Mohamed Boudiaf »,2011.
- [22] J.-P. Cocquerez , S. Philipp.,1995 , Analyse d'Images : filtrage et segmentation, Masson, DE HAUTE-ALSACE, 2002.
- [23]KaiwenCHANGLe., Machine learningbased image segmentation Apprentissage artificiel pour la segmentation d'image, Thèse de Doctorat, Préparée à MINES ParisTech université PSL,2019.

- [24] L.Lechlek, M .Ghougal.,''la segmentation d'image médicale par la méthode d'ensemble de niveaux (level sets)'' .Thésede Master, Université Larbi Ben M'hidi Oum El Bouaghi.2012.
- [25] G'urayErus., Reconnaissance d'objets cartographiquesdans les images satellitaires `a haute r'ésolution, Thèse de Doctorat, Universit'e Paris Descartesc entre universitaire des Saints-P`eres, 2008.
- [26] S Belmerabet ,NBardjak ., Segmentation d'image, Mémoire de Fin d'Etudes : MASTER, Université Larbi Ben M'hidi Oum El Bouaghi, 2017.
- [27] Youssef EL MERABET., Segmentation d'images couleur par combinaisonL PE-régions/LPE-contours et fusion de régions - Application à la segmentation de toitures à partir d'orthophoto plans, Thèse de Doctorat, u n i v e r s i t é de t e c h n o l o g i e b e l f o r t - m o n t b é l i a r d, 2013 .
- [28]Wassima Ait fares ., detection et suivi d'objets par vision fondes sur segmentation par contour actif base region, Thèse de Doctorat , l'université de toulouse ,2013.
- [29] R. Jain., << Difference and accumulative difference pictures in dynamic scene analysis>>, Image and Vision Computing,1984.
- [30] A. Makarov., << Comparison of background extraction based intrusion detection algorithms>>, Danes International Conference on Image Processing ICIP '96, 1996.
- [31] M.Sandeli., <<traitement d'images par des approches bio-inspirées>>, thèse de Magister, Université Constantine 2 ,2014.
- [32] C. VIEREN., << Segmentation de scènes dynamiques en temps réel. Application au traitement de séquences d'images pour la surveillance de carrefours routiers >>, Thèse de Docteur de l'Université, USTL Flandres Artois, 1988.
- [33] M. ORKISZ, <<Localisation d'objets mobiles dans des scènes naturelles filmées par une caméra fixe>>, Traitement du signal, 1992.
- [34] D.S. Zhang , G., Lu, << Segmentation of Moving Objects in Image Sequence: A Review, Circuits, Systems and Signal Processing>> (Special Issue on Multimedia Communication Services), 2001.
- [35] M.J. Black,P.Anandan., << The robust estimation of multiple motions>>: Parametric and piecewise-smooth flow fields, CVIU, 1996.

- [36] R. Deriche., << Fast algorithms for low-level vision>>, IEEE-PAMI, 1990.
- [37] F. CABESTAING., <<Détection de contours en mouvement dans une séquence d'images. Conception et réalisation d'un processeur câblé temps-réel >>, Thèse de Docteur de l'Université, USTL Flandres Artois, 1992.
- [38] International Journal of Advanced Science and Technology Vol. 38, January, an efficient medical Image Segmentation, Using Conventional OTSU Method
ChHima Bindu¹ and K.Satya Prasad²
¹ECE Department, QIS College of Engineering & Technology, Ongole, Andra Pradesh, India
²ECE Department, JNTU Kakinada, Kakinada, Andra Pradesh, India.
- [39] Murielgastaud., modèles de contours actifs pour la segmentation d'images et de vidéos, de doctorat, Université de nice-sophiaantipolis, 2005.
- [40] Eric HUEBER, Segmentation d'images par contour actif : implantation optique avec un corrélateur incohérent ombroscopique, thèse de doctorat..
- [41] Suresh Chandra Satapathy¹•N. Sri Madhava Raja²•V. Rajinikanth²•Amira S. Ashour³•Nilanjan Dey, ORIGINAL ARTICLE Multi-level image thresholding using Otsu and chaotic bat algorithm, Neural Comput & Applic D.
- [42] Liao PS, Chen TS, Chung PC (2001) A fast algorithm for multi-level thresholding. J Inf Sci Eng 17(5):713–727.
- [43] 'Mezianiroza ,Bounifkatia, segmentation d'image couleur par seuillage d'histogrammes, Université mouloud mammeri de Tizi–Ouzou Mémoire de Fin d'Études de MASTER ACADIMIQUE, 2017.
- [44] H. Schweitzer, J. Bell et F. Wu., Very fast template matching, Computer Vision—ECCV 2002, 2006.
- [45] H. Tao Harpre ,S. Sawhney , R. Kumar., Object tracking with Bayesian estimation of dynamic layer representations, IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI), 2002.
- [46] Hitesh A Patel¹, Darshak G Thakore., Moving Object Tracking Using Kalman Filter, IJCSMC, April 2013.
- [47] S. Medouakh., Détection et suivi d'objets, Thèse de doctorat, Université Mohamed Khider – Biskra, 2019.

- [48] Shengluan Huang., Moving Object Tracking System Based On Camshift And Kalan Filte, Computer Science Department of Xiamen University Csdxmu.
- [49] Shiuh-Ku Weng , Chung-Ming Ku, Shu-Kang tu., Video object tracking using adaptive Kalman filter, Department of Information Management, Chinese Naval Academy, Zuoying District, Kaohsuing City 813, 2006.
- [50]S. Suzuki, K. Abe., "Topological Structural Analysis of Digital BinaryImages by Border Following.", CVGIP, v.30, n.1. 1985.
- [51] G.Salv., An Automated Vehicle Counting System Based on Blob Analysis forTraffic Surveillance, Department of Economics Studies, University of Naples "Parthenope", Naples, Italy.