République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Dr Yahia Farès de Médéa
Faculté des Sciences et de la Technologie
Département de Génie Électrique et de l'Informatique



Mémoire

Présenté en vue d'obtention du diplôme de Magister

Spécialité : Électronique

Option : Instrumentation et Microélectronique

Présenté par :

Mr : Ben Salah Hocine

Thème

Poursuite d'objets en mouvements par filtrage particulaire

Soutenue publiquement le 27 Mai 2014

Jury

Pr. Salam ABUDURA	Professeur, U. Médéa	Président
Dr. Noureddine BATEL	MCA, U. Médéa	Examinateur
Dr. Mohamed OULDZMIRLI	MCA, U. Médéa	Examinateur
Pr. Abderrezak GUESSOUM	Professeur, U. Blida	Encadreur

2013-2014

Remerciements

À l'issue de cette fin d'étude, j'adresse mes sincères remerciements premièrement à dieu tout puissant qui m'a donné la santé, la volonté et la patience de vouloir terminer et réussir ce travail.

Ensuite, je tiens à adresser mes plus vifs remerciements à mon promoteur Mr. ***Guessoum. A****, professeur à l'université de Blida pour m'avoir encadré, suivi et encouragé.*

Je remercie également, tous les membres du jury de bien vouloir juger ce travail. Ainsi que tous mes enseignants du Département d'Electronique de l'université de Médéa.

*Je n'exclue pas de ces remerciements toutes les personnes qui m'ont aidé de près ou de loin dans la réalisation de ce mémoire, notamment monsieur****:*** ***Belhout.K****, pour ses conseils appréciables et ses critiques constructives.*

Les discussions, les remarques et les commentaires de mes collègues ; monsieur ***Mansouri.S****, monsieur* ***Mahmoudi.A****, et madame* ***Kerrach.S*** *ont été sources d'idées et ont contribué au développement et à l'amélioration de cette étude.*

Je veux également et particulièrement remercier mes parents, pour leur soutien inconditionnel tout au long de mes longues années d'études.

Enfin, je remercie tous mes frères et sœurs, ainsi que ma chère épouse pour leur soutien moral.

Poursuite d’objets en mouvements par filtrages particulaire

ملخص:

هذه المذكرة تخص الخوارزميات المتعلقة بمتابعة الآهداف. اهتمامنا يتمحور حول الأهداف المتحركة وبالأخص تقدير حركات هذه الآهداف اللازمة للتطبيق الكافي لحالة المتابعة. نوعية المتابعة تكون أحسن بكثير بفضل التقديرات المتينة للحركة. من أجل انجاز التقدير, اخترنا تقنيات الترشيح الإحصائي الذي يتعلق بمرشح كالمان في الحالة الخطية القوصية أو بطرق تعاقبية لمنتي كارلو في الحالة الغير الخطية. تمثيل الحالة مكيف دائما بدلالة الملاحضات الدائمة وهذا من أجل تمثيل أحسن لديناميكيات النظام. في هذه المذكرة , عملنا يهدف الى مقارنة نتائج المعطيات بواسطة مرشح كالمان ومرشح منتي كارلو في اِطار متابعة الهدف المتحرك.

الطرق المقدمة في هذه المذكرة تم تأكيدها بالتمثيل على الحاسوب.

كلمات مفتاحية : مرشح إحصائي, مرشح كالمان , طرق لمنتي كارلو, مرشح منتي كارلو التعاقبي, متابعة الهدف المتحرك.

Résumé:

Le travail de ce mémoire concerne plus particulièrement les algorithmes de poursuite de cible. Nous nous sommes intéressés au cas d’objets mobiles, et particulièrement à l’estimation du mouvement de ces objets, nécessaire pour une exécution satisfaisante des taches de poursuites. La qualité de poursuite est grandement améliorée grâce à une estimation robuste du mouvement. Pour réaliser l’estimation, nous avons choisi les techniques de filtrage stochastique. Il s’agit de filtre de Kalman dans le cas linéaire gaussien, ou des méthodes séquentielles de Monte Carlo dans le cas non linéaire. La représentation d’état est adaptée en permanence en fonction des observations courantes pour représenter au mieux la dynamique du système. Dans ce mémoire, notre travail consiste à faire de la comparaison des résultats donnés par le filtre de Kalman et le filtre particulaire dans le cadre de la poursuite d’objet en mouvement.

Les méthodes présentées dans ce mémoire ont été validées en simulation.

Mots clés: Filtrage stochastique, filtre de Kalman, Méthode de Monte Carlo, Filtre particulaire, poursuite d’objet en mouvement.

Summary:

The work of this memory more particularly relates to the algorithms of tracking target. We were interested in the mobile cases of objects, and particularly in the estimate of the movement of these objects, necessary for a satisfactory execution of the spots of tracking. The quality of tracking is largely improved thanks to a robust estimate of the movement. To carry out the estimate, we chose the techniques of stochastic filtering. It is about Kalman filter in the Gaussian linear case or of the sequential methods of Monte Carlo in the nonlinear case. The representation of state is adapted permanently according to the current observations to represent the dynamics of the system as well as possible. In this memory, our work consists in making comparison of the results given by the Kalman filter and the particle filter within the framework of the moving object tracking.

The methods presented in this memory were validated in simulation.

Key words: Stochastic filtering, Kalman filter, Monte Carlo method, Particle Filter, tracking of moving object.

SOMMAIRE

Introduction Générale 1

CHAPITRE I
Généralités sur l'analyse d'image

I.1 Introduction 4
I.2 Image numérique 5
I.3 Bruit 5
I.4 Filtrage 6
I.5 Segmentation d'image 10
I.6 Détection de contour 10
I.6.1 Technique du gradient 11
I.7 Segmentation par morphologie mathématique 13
I.8 Seuillage 15
I.9 Définition d'une vidéo 16
I.10 Résolution temporelle d'une vidéo 16
I.11 Séquence d'image 16
I.12 Analyse du mouvement dans une séquence d'images 17
I.12.1 Détection de mouvement 18
I.12.2 Définition d'un objet vidéo 20
I.12.3 Suivi d'objets dans la vidéo 20
I.13 Conclusion 20

CHAPITRE II
Détection d'objets en mouvement

II.1 Introduction 21
II.2 Détection d'objets en mouvement par la méthode de Segmentation 21
II.3 Détection par différence d'images 26
II.3.1 Approche région 26
II.3.2 Approche contour 28
II .4 Détection des contours en mouvement par (mcm) 32
II.5 Conclusion 34

CHAPITRE III
Suivi d'objets en mouvement

III.1 Introduction 35
III.2 Algorithmes de suivi 35
III.3 Classification des algorithmes de suivi 35
III.3.1 Les algorithmes orientés mouvement 35
III.3.2 Les algorithmes orientés modèle 36
III.4 Structure général des algorithmes de suivi 37
III.4.1 Phase de prédiction 37
III.4.2 Phase d'ajustement 38
III.4.3 Phase d'estimation 38
III.5 Suivi de trajectoire d'objet en mouvement 39
III.6 Modélisation des trajectoires d'une cible 41
III.6.1 Modè1e d'une droite (Trajectoire rectiligne) 41
III.6.2 Modèle d'un cercle (Trajectoire circulaire) 45
III.7 Applications du suivi 49
III.8 Conclusion 50

CHAPITRE IV
Le filtrage particulaire

IV.1 Introduction 51
IV.2 Le problème du filtrage 51
IV.2.1 Le filtrage optimal 52
IV.3 Méthodes de résolution 53
IV.3.1 Méthodes Analytiques 53
IV.3.1.1 Filtre de Kalman-Bucy (KF) 53
IV.3.1.2 Filtre de Kalman Etendu (EKF) 55
IV.3.1.3 Filtre de Kalman sans biais (UKF) 55
IV.3.2 Méthodes Numériques 56
IV.4 Méthodes de Monte Carlo pour le filtrage non linéaire 57
VI.4.1 Echantillonnage Monte Carlo standard 58
IV.4.1.1 Echantillonnage Monte Carlo pondéré 59
IV.4.1.2 Génération de variables aléatoires par la méthode d'acceptation/rejet 59
IV.4.2 Filtrage particulaire 60
IV.4.2.1 L'algorithme du Filtrage Particulaire 60

IV.4.2.2 Choix de la densité d'importance 63
IV.4.2.3 Dégénérescence du filtre particulaire 64
IV.4.2.4 Ré-échantillonnage des particules 65
IV.4.2.5 L'indicateur de Rééchantillonnage 67
IV.4.2.6 Résumé de l'algorithme du filtrage particulaire (S.I.R) 68
IV.5 conclusion 69

CHAPITRE V
Les Résultats de Simulation

V.1 Introduction 70
V.2 Simulink 70
V.3 Conception et simulation vidéo et les systèmes de traitement de l'image 71
V.4 Le système de suivi 71
V.4.1 Prétraitement 72
V.4.1.2 La conversion des couleurs 73
V.4.1.1 Affichage de la fréquence trame 73
V.4.1.3 La soustraction de l'arrière-plan 73
V.5 Modèle exemplaire de suivi de personnes 74
V.6 Les résultats de suivi des personnes 78
V.6.1 Interprétations des résultats de suivi des personnes 84
V.6.2 Comparaison entre le filtre Kalman et le filtre particulaire 85
V.6.2.1 Le filtre Kalman 85
V.6.2.2 Le filtre particulaire 87
V.7 Conclusion 88
Conclusion générale 90
Bibliographie 92
Annexe 96

LISTE DES FIGURES

CHAPITRE I
Généralités sur l'analyse d'image

Figure I.1 : Application du filtre moyenneur, (a) : image d'origine, (b) : filtre 3×3, (c) : filtre 9×9 8

Figure I.2 : Application du filtre median, (a) : image d'origine, (b) : image bruitée (poivre et sel), (c) : filtre 3×3 9

Figure I.3 : Exemple d'une segmentation d'image, (a) : image d'origine, (b) : image segmentée 10

Figure I.4 : Exemple d'application du filtre Robert, (a) : image d'origine, (b) : contour de Robert. 12

Figure I.5 : Exemple d'application du filtre Sobel, (a): image d'origine, (b): contour de Sobel 12

Figure I.6 : Exemple d'application de la dilation, (a) : image binaire, (b) : image dilatée 13

Figure I.7: Exemple d'application de l'érosion, (a) : image binaire, (b) : image érodée 14

Figure I.8 : Opération de seuillage. *(a)* : une image originale en 256 niveaux de gris. 15

Figure I.9 : Exemple d'une séquence d'images 17

Figure I.10 : Exemple de vidéosurveillance, (a) : les caméras de surveillance, (b) : dans salle de contrôle : les hommes sont aidés par la machine dans la tâche de détection 18

CHAPITRE I
Détection d'objets en mouvement

Figure II.1 : Schéma synoptique de la segmentation spatiale 22

Figure II.2 : Les minima locaux m_i et les bassins de captures BC_i associés 23

Figure II.3 : Partitions et graphes d'adjacences associés 25

Figure II.4 : Exemples de fusion de deux régions à partir d'un GAR de la partition 25

Figure II.5 : Technique de différence avec image de référence, (a) l'image à l'instant t, (b) l'image de référence, (c) l'image de différence 26

Figure II.6 : Technique de différence sans image de référence, (a) image précédente, (b) image courante, (c) image de différence 28

Figure II.7 : 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 29

Figure II.8 : Extraction des contours en mouvement par la proposition de Vieren 31

Figure II.9 : Schéma synoptique de l'opérateur mcm 33

CHAPITRE III
Suivi d'objets en mouvement

Figure III.1 : Exemple de trajectoire mesurée et estimée 40
Figure III.2 : Modélisation d'une droite 41
Figure III.3 : Modélisation d'un cercle 45

CHAPITRE IV
Le filtrage particulaire

Figure IV.1 : Mesure discrète pondérée 61
Figure IV.2 : Ré-échantillonnage Multinomial 66

CHAPITRE V
Les Résultats de Simulation

Figure.V.1 : Le système de suivi 72
Figure.V.2 : Modèle de suivi des personnes 74
Figure V.3 : Estimation de l'arrière-plan 75
Figure V.4 : Sous-système de segmentation 76
Figure V.5 : Sous-système de détection 76
Figure V.6 : Sous-système de suivi utilisant le filtre de Kalman 77
Figure V.7: Sous- système de suivi utilisant le filtre de particulaire 78
Figure.V.8 : Suivi d'une seule personne pour la première vidéo 79
Figure.V.9 : Suivi d'une seule personne pour la deuxième vidéo 79
Figure.V.10 : Suivi d'une seule personne pour la troisième vidéo 80
Figure.V.11 : Suivi de deux personnes pour la première vidéo 81
Figure.V.12 : Suivi de deux personnes pour la deuxième vidéo 82
Figure V.13 : Suivi de deux personnes pour la troisième vidéo 82
Figure.V.14 : Suivi de plusieurs personnes 83
Figure.V.15 : Suivi de deux personnes 84
Figure.V.16 : Plot positions des boites entourantes au cours du temps 85
Figure.V.17 : Suivi de deux personnes par filtre Kalman 86
Figure.V.18 : Plot positions des boîtes entourantes au cours du temps par filtre Kalman 86
Figure.V.19 : Suivi de deux personnes par filtre Particulaire 87
Figure.V.20 : Plot positions des boîtes entourantes au cours du temps par filtre particulaire 88

Liste des Notations générales

$x \sim f$: x est distribué suivant la densité f.

$f \propto g$: f est proportionnel à g.

$det\ P$: Déterminant de la matrice P.

k : Le temps discrétisé.

$x_{0:k} = (x_0, x_1, \dots, x_k)^T$: Trajectoire du processus.

$y_{1:k} = (y_1, \dots, y_k)^T$: Vecteur de mesures jusqu'à l'instant k.

$p_k = p(x_k|y_{1:k})$: Densité conditionnelle.

$p_{k|k-1} = p(x_k|y_{1:k-1})$: Densité prédite.

$< f, g > = \int f(x)g(x)dx$.

$\bar{x} = \sum_{i=1}^{N} \omega^i x^i$, $cov(x^i|\omega^i) = \sum_{i=1}^{N} \omega^i (x^i - \bar{x})(x^i - \bar{x})^T$.

N : Le nombre de particules.

ω^i: Poids des particules x^i.

$d\mathbb{E}_p[.]$: L'espérance par rapport à la densité p.

var_p: Variance par rapport à la densité p.

$P_{k|k-1}$: Matrice de covariance de l'état prédit $\hat{x}_{k|k-1}$.

P_k : Matrice de covariance de l'état corrigé $\hat{x}_k$.

Q_k : Matrice de covariance du bruit dynamique.

R_k : Matrice de covariance du bruit de mesure.

$\emptyset(m|P)$: Densité Gaussienne de moyenne m et de matrice de covariance P.

$u(a, b)$: Loi de probabilité uniforme sur l'intervalle $[a, b]$.

h : Facteur de dilatation du noyau.

$q(x)$: Densité d'importance.

$(K * p)(x) = \int K(x-u)p(u)du$: produit de convolution entre les densités K et p.

$\Gamma(x) = \int_0^{+\infty} e^{-t}t^{x-1}\,dt$: La fonction gamma.

$\widetilde{\omega}_k^i$: Poids des particules x_k^i normalisés.

Ent : Indicateur de ré-échantillonnage entropique.

Th : Seuil de ré-échantillonnage entropique.

Si P est une matrice de covariance : $P \geq 0$; P est semi-définie positive.

α : L'angle que fait la droite avec l'axe des abscisses.

s : La distance parcourue le long de la droite.

v : La vitesse.

a : L'accélération.

X_O : L'abscisse du centre du cercle.

Y_0 : L'ordonnée du centre du cercle.

R_0 : Rayon du cercle.

Abréviations

FP : Filtre Particulaire.

FNL : Filtrage Non Linéaire.

MC : Monte Carlo.

UKF : Unscented Kalman Filter.

EKF : Extended Kalman Filter.

SIR : Sampling Importance Resampling.

RPF : Regularized Particle Filter.

iid : indépendantes identiquement distribuées.

Les autres abréviations sont explicitées dans le texte.

Introduction Générale

Dans tout système biologique de vision, la capacité d'analyse des images est très développée parce qu'elle s'avère être d'une importance capitale dans de nombreuses situations, comme par exemple la chasse ou la communication. Dans le cas de la vision humaine, le traitement du signal effectué à la fois par les cellules nerveuses de la rétine et par le reste du cerveau permet d'extraire de l'image reconstituée sur la rétine une quantité d'information gigantesque, traitée en temps réel. Lorsque l'on veut suivre un objet, le cerveau effectue différentes tâches : la reconnaissance de l'objet en question, l'estimation de son mouvement .... L'idée de la vision par ordinateur est de construire un système capable de reproduire de telles opérations ; pour cela, on procède par étapes en construisant différents modules pour estimer le mouvement d'objets connus ou inconnus, reconnaître des objets, les suivre dans une séquence d'image….

L'idée de base d'un module de suivi est de pouvoir suivre un objet (pas forcément connu au départ) le plus longtemps possible. Mais que signifie exactement l'expression « suivre un objet »? Intuitivement, suivre un objet dans une séquence d'images, c'est être capable de localiser cet objet dans chaque image, la caméra et/ou l'objet pouvant être en mouvement. Mais le suivi ne se limite pas à cette tâche de localisation : on doit aussi caractériser l'objet d'intérêt. La caractérisation de l'objet dépend de l'application visée, et peut être une information de taille, d'orientation, de couleur ou de forme. Dans certaines applications, il peut être utile de connaître les déformations possibles de l'objet. L'ensemble de ces informations constituent l'état de l'objet d'intérêt à un instant donné. Selon l'application on va choisir de suivre des points d'intérêt, des primitives géométriques (lignes, courbes, . . .), ou plutôt une forme ou un contour (dans le cas du suivi d'un visage par exemple). L'objectif du suivi est donc d'inférer un certain nombre de paramètres relatifs à l'état de l'objet d'intérêt afin de pouvoir le localiser et le caractériser dans chacune des images de la séquence.

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. 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.

Le filtre de Kalman est un outil d'estimation classique qui permet de déterminer récursivement l'état d'un système dynamique à partir d'une mesure. Cette méthode est utilisée pour la navigation de robots ou pour l'aide à la conduite par exemple.

L'algorithme de base met en jeu un modèle qui décrit l'évolution temporelle de l'état du système et un modèle qui relie cet état à la mesure que l'on peut en faire. Le filtrage se décompose en trois étapes :

1. L'étape de prédiction, qui utilise le modèle d'état pour estimer la valeur prévue du paramètre, ainsi que la variance (ou matrice de covariance) associée.
2. L'étape d'observation, qui recueille une mesure.
3. L'étape d'estimation, qui pondère la mesure effectuée et la prédiction pour mettre à jour la variable d'état et la variance associée.

L'utilisation de ce filtre se limite aux cas où les modèles d'état et les mesures sont linéaires, et où les modèles de bruit sont additifs gaussiens. En effet, ces conditions remplies, la loi a posteriori présente l'avantage d'être une gaussienne et peut être entièrement définie par sa moyenne et sa covariance.

Dans le cas où les modèles d'état ne sont plus linéaires ou que les modèles de bruit ne sont plus gaussiens, il existe des solutions au problème : ce sont des extensions du filtre de Kalman. Parmi les plus courantes, on trouve le filtre de Kalman linéarisé et le filtre de Kalman étendu.

Par contre, lorsque les modèles d'état ne sont plus linéaires et que les modèles de bruit ne sont plus gaussiens, il faut se placer dans un cadre bayésien et utiliser des méthodes probabilistes pour résoudre le problème.

Les méthodes probabilistes fournissent alors un cadre permettant d'estimer la loi a posteriori de manière simple. Les méthodes séquentielles de Monte Carlo proposent une approximation de cette loi par une somme de lois de Dirac centrées en des points appelés particules, et pondérées par la vraisemblance des mesures conditionnellement à ces points. Ces méthodes reposent sur le principe de Monte Carlo qui consiste à, simuler un grand nombre de réalisations d'une variable aléatoire pour en approcher sa loi. Le concept est apparu au milieu du vingtième siècle mais n'a été mis en œuvre que beaucoup plus tard dans le cadre du problème de l'estimation séquentielle, car il a fallu attendre que la puissance de calcul des ordinateurs soit suffisante. L'objectif de ce mémoire est d'étudier et d'implémenter une méthode de suivi d'objets dans une séquence vidéo, basée sur les méthodes récentes d'estimation par filtrage particulaire.

Ce mémoire est organisé en cinq chapitres selon le plan suivant :

Dans le chapitre1, on présente des généralités sur l'image numérique et l'analyse de mouvement dans une séquence d'images.
Dans chapitre 2, on présente un état de l'art des méthodes de détection et de segmentation du mouvement.
Dans le chapitre 3, on traite de l'estimation du mouvement servant comme complément à la détection ; un bref état de l'art des méthodes de suivi de mouvement est fourni, et nous décrivons les différents modèles de trajectoires des systèmes dynamiques non linéaires. La majorité des trajectoires ne sont que des successions de trajectoires élémentaires canoniques telles que des segments de droites, des arcs de cercles, ...

Dans le chapitre 4, on décrit un état d'art des différentes méthodes analytiques (filtre de Kalman —Bucy. filtre de Kalman étendu. . ..) et numériques classiquement utilisées dans le filtrage. Nous montrons les limites de ces méthodes de résolution, et on décrit le principe fondamental des méthodes de Monte Carlo et on compare certain de ces méthodes appliquées dans le filtrage non linéaire. La liste de ces méthodes n'est pas exhaustive. Nous montrons certaines faiblesses des méthodes particulaires classiques.

Dans le dernier chapitre, nous abordons une implémentation d'un modèle choisi du suivi de personnes, dans lequel on a remplacé le filtre de Kalman par le filtre particulaire. Ceci nous permet de les comparer par rapport à leur efficacité.

Une conclusion générale et le dégagement de plusieurs perspectives à ces travaux clôt ce document.

Chapitre I:

Généralités sur l'analyse d'image

I.1 Introduction

Le XXI*e* siècle est celui de l'image numérique, vecteur d'information pour la presse, les artistes, le monde médical mais aussi celui de l'industrie. Le traitement d'images est une science récente qui a pour but d'offrir aux spécialistes de différents domaines, comme au grand public, des outils de manipulation de ces données digitales issues du monde réel.

L'histoire commence dans les années cinquante avec les rayons X. Puis vint l'étude de méthodes d'extraction automatique d'informations avec de nombreuses applications en reconnaissance de caractères et en contrôle qualité. **[1]**

Dans les années quatre-vingt, les efforts se poursuivirent avec l'introduction des systèmes experts qui devaient remplacer l'homme dans ses différents diagnostics. Enfin dans le courant des années quatre-vingt et quatre-vingt-dix, le traitement d'images passe du 2 D au *n* D grâce aux nouvelles possibilités technologiques. Les supports de stockage, les processeurs permettent désormais de traiter des informations multidimensionnelles de couleur et de mouvement.

Puis dans les années deux mille, l'intérêt se porte de plus en plus sur les modèles perceptifs afin d'optimiser les modèles usuels. Par ailleurs, plutôt que de développer des algorithmes utilisables dans toutes les situations, des méthodes dédiées sont développées, en intégrant de la connaissance à priori sur les contenus et surtout sur la finalité de l'exploitation des données.

Le traitement d'images est donc une science en pleine évolution. Nous allons tenter de donner les bases de cette science qui a pour objectifs principaux :

- de saisir une scène réelle pour constituer un fichier informatique, c'est l'étape d'acquisition.
- de mettre en évidence les informations intéressantes : c'est l'étape de traitement dans laquelle on distinguera tout particulièrement les étapes de débruitage puis la segmentation.
- d'interpréter ces informations pour décider d'un diagnostic ou d'une action à engager : c'est l'étape d'analyse.
- sans oublier les étapes de stockage des informations, à différentes étapes de la chaîne, et de visualisation et/ou d'impression des résultats.

I.2 Image numérique

Une image numérique est une matrice de pixels repérés par leur coordonnées (x, y). S'il s'agit d'une image couleur, un pixel est codé par 3 composantes (r, g, b) (chacune comprise au sens large entre 0 et 255), représentant respectivement les doses de rouge, vert et bleu qui caractérisent la couleur du pixel.

S'il s'agit d'une image en niveau de gris, il est codé par 1 composante comprise au sens large entre 0 et 255, représentant la luminosité du pixel.

I.3 Bruit

En traitement d'image, toute brusque fluctuation d'un pixel par rapport à ses voisins étant considéré comme un bruit provoquant ainsi la dégradation de la qualité d'image que ce soit de luminance ou de contraste. Ses sources les plus fréquentes sont :

a. Bruit lié aux dispositifs d'acquisition

La caméra, les câbles et tous autres blocs de connexion engendrent un bruit qui a un effet multiplicatif, pour cela, il serait plus judicieux de mettre les sources qui produisent plus de bruit dans les étages inférieurs pour diminuer leur contribution générale au bruit général.

b. Bruit lié à la scène

Il se trouve parfois que la scène soit polluée de poussière, ou couverte de nuages, brouillard ou buée qui dégrade la qualité des images. Cependant, là encore, la connaissance a priori du phénomène perturbateur permet d'envisager une modélisation et donc une correction systématique. Ce type de bruit a un effet de bruit additif.

c. Bruit lié à la transmodulation

Les images qui sont capturées à partir d'une séquence vidéo qui regroupe le son et l'image dans un même signal sont atteintes par un bruit du à la transmodulation qui se produit du fait qu'on utilise une double modulation : modulation audio et vidéo. Il arrive, ainsi, des cas où des composantes du signal audio bruitent notre signal vidéo.

I.4 Filtrage

I.4.1 Notion du filtre

Un filtre est souvent une transformation mathématique (appelée produit de convolution) permettant pour chaque pixel de la zone à laquelle il s'applique de modifier sa valeur en fonction des valeurs des pixels avoisinants, affectées de coefficients. Le filtre est représenté par un tableau (matrice), caractérisé par ses dimensions et ses coefficients, dont le centre correspond au pixel concerné. Les coefficients du tableau déterminent les propriétés du filtre.

I.4.2 Principe du filtrage

Le filtrage consiste à modifier la valeur des pixels de tout ou partie d'une image numérique, généralement dans le but d'améliorer son aspect lorsque l'image ne peut pas être exploitée directement. En pratique, il s'agit de créer une nouvelle image en se servant des valeurs des pixels de l'image d'origine entièrement ou partiellement :

a. Filtrage global

Où chaque pixel de la nouvelle image est calculé en prenant en compte la totalité des pixels de l'image de départ.

b. Filtrage local

Où chaque pixel de la nouvelle image est calculé en prenant en compte seulement un voisinage du pixel correspondant dans l'image d'origine. Il est d'usage de choisir un voisinage carré et symétrique autour du pixel considéré. Ces voisinages sont donc assimilables à des tableaux à deux dimensions (matrices) de taille impaire.

N'entrent pas dans la catégorie du filtrage toutes les transformations de l'image d'origine : zoom, découpage, projections.

I.4.3 Types de filtrage

On distingue généralement les types de filtres suivants :

a. Les filtres passe-bas

Consistant à atténuer les composantes de l'image ayant une fréquence hauteCe type de filtrage est généralement utilisé pour atténuer le bruit de l'image, c'est la raison pour laquelle on parle habituellement de "lissage", "débruitage" ou encore "filtre antibruit". Les coefficients du masque sont positifs et leur somme est égale à 1.

b. Les filtres passe-haut

À l'inverse des passe-bas, atténuent les composantes de basse fréquence de l'image et permettent notamment d'accentuer les détails et le contraste, et donc les différences entre les pixels voisins. C'est la raison pour laquelle le terme de "filtre d'accentuation" est parfois utilisé. Les coefficients du masque peuvent être de signe quelconque mais leur somme est égale à 0.

I.4.4 Techniques du filtrage

Il en existe plusieurs, chaque filtre cherche à atténuer un type de défaut bien précis. Il n'y a pas de filtre universel capable de corriger tous les défauts. Il faut choisir les bons filtres suivant les défauts que nous désirons corriger.

a. Filtres linéaires

Le filtre local est dit linéaire si la valeur du nouveau pixel est une combinaison linéaire des valeurs des pixels du voisinage y compris le pixel en question.

- **Filtre moyenneur**

Cette technique est la plus directe pour adoucir une image trop heurtée dans ses contours.

Le niveau de gris de chaque pixel est remplacé par la moyenne des niveaux de gris de ses huit voisins. Ce filtrage pourra se faire à l'aide du filtre :

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

Le facteur 1/9 sert à normaliser le filtre de manière que celui-ci n'influe pas sur l'intensité globale de l'image.

Cette technique a pour défaut de créer un flou sur les bords des objets.

Pour une image d'entrée I_e on a :

Is image de sortie tel que :

$$I_s = 1/9\{I_e(x-1,y-1) + I_e(x-1,y) + I_e(x-1,y+1) + I_e(x,y-1) + I_e(x,y) + I_e(x,y+1) + I_e(x+1,y-1) + I_e(x+1,y) + I_e(x+1,y+1)\}.$$

$$= 1/9 \sum\sum H(i+1,j+1).I_e(x+i,y+j). \quad \text{(I.1)}$$

et $i \in [-1,+1]$; $j \in [-1,+1]$.

De manière plus générale, pour un filtre H de taille $n+1$ x $n+1$ et la somme des coefficients vaut K (facteur de normalisation) :

$$I_s = 1/k \sum\sum H(i+n/2\ , j+n/2).\, I_e\ (x+i, y+j). \quad \text{(I.2)}$$

Avec n : pair, $i \in [-n/2, +n/2]$; $j \in [-n/2, +n/2]$.

La formule précédente n'est autre que le produit de convolution discrète.



Figure I.1 Application du filtre moyenneur, (a) : image d'origine, (b): filtre 3×3, (c) : filtre9×9

b. Filtres non linéaires

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

- **Filtre median**

Consiste à remplacer l'intensité de chaque pixel par la valeur médiane de ses voisins choisis à l'avance. C'est un filtre passe bas qui élimine le bruit de type neige, mais il entraîne cependant une petite perte de résolution, son principe est :

Pour chaque pixel on classe les pixels voisins du pixel courant, par valeurs croissantes, on prend la valeur médiane des pixels classés et on l'affecte au pixel courant.

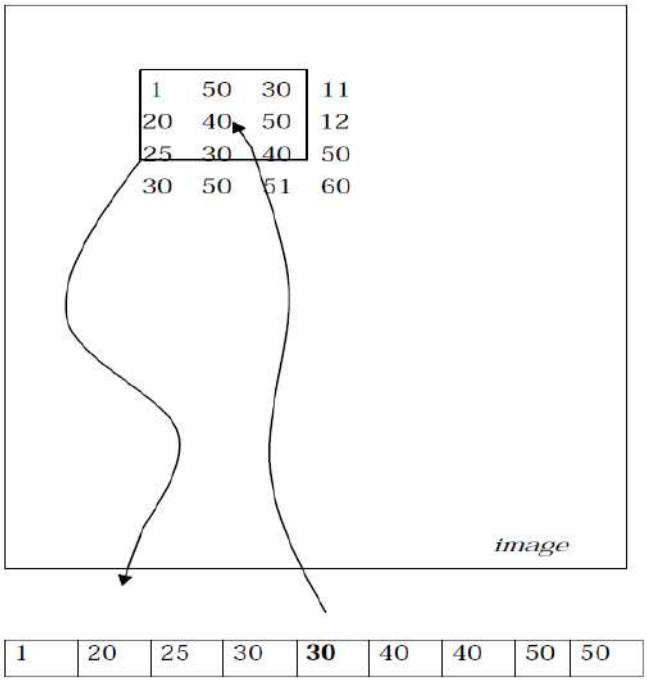




Figure I.2 Application du filtre median, (a) : image d'origine, (b) : image bruitée (poivre et sel), (c) : filtre 3×3.

I.4.5 Avantages du filtrage

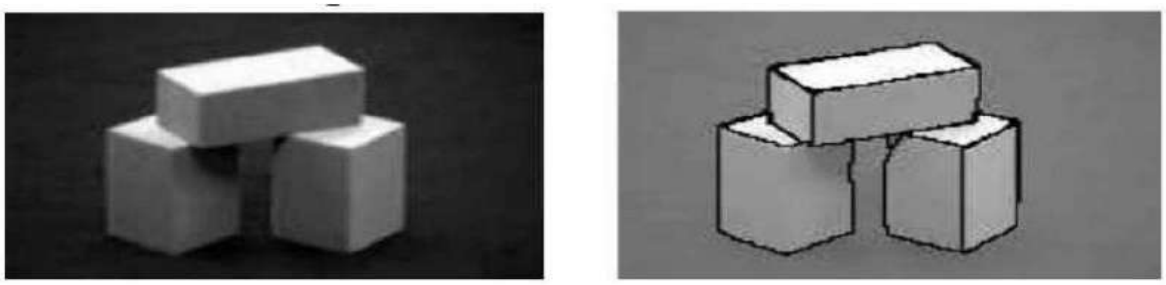
On citera comme avantage :

- Accentuer les variations d'intensité de l'image : renforcer les différences entre les pixels appartenant à des régions adjacentes (rehaussement de contraste).
- Détection de contours et préservation de la forme des régions.
- Renforcement de l'homogénéité à l'intérieur des régions (réduction de bruit).

Cependant, 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. Dans ce genre de situation le filtrage est juste une première étape dans un traitement plus complexe qui fera intervenir des informations sémantiques sur l'image.

I.5 Segmentation d'image

Les images sont composées de régions possédant des propriétés locales qui peuvent être la répartition des niveaux de gris. En regroupant des points de l'image qui possèdent une même propriété donnée, on obtient des régions uniformes. Cette opération s'appelle «segmentation».



(a) (b)

Figure I.3 Exemple d'une segmentation d'image, (a) : image d'origine, (b) : image segmentée.

Une autre méthode de segmentation basée sur les discontinuités des niveaux de gris, celles-ci indiquent les frontières entre les régions de l'image. La variation brusque des niveaux de gris s'appelle «contour» (un contour correspond à une forte variation de la fonction intensité).

I.6 Détection de contour

La détection de contours est très utilisée comme étape de prétraitement pour la détection d'objets, pour trouver les limites de régions. En effet, un objet peut être localisé à partir de l'ensemble des pixels de son contour. De plus, trouver cet ensemble permet d'obtenir une information sur la forme de l'objet. Du point de vue théorique, un contour est défini par un changement marqué de l'intensité d'un pixel à l'autre. Comme en chaque point, le contour est considéré comme perpendiculaire à la direction du gradient, ce dernier est utilisé pour la détection : selon les modèles classiques de traitement d'image, en tout point de l'image les dérivées partielles sont estimées. Ceux qui fournissent les réponses les plus fortes correspondent à des pixels de contour.

En pratique, le calcul de gradient est effectué par la convolution de filtres linéaires. Ces filtres peuvent estimer les dérivées premières, les dérivées secondes, etc., et se focaliser sur des contours de directions différentes. Il existe ainsi de nombreux filtres, qui ont des fondements mathématiques différents. Cependant, quelle que soit leur justification théorique,

leur objectif reste le même : mettre en évidence les pixels qui ont une valeur très différente de leurs voisins. Nous ne présentons ici, à titre d'exemple, que le filtre gradient et les opérateurs de Robert et Prewitt parmi les plus couramment employés.

I.6.1 Technique du Gradient

Soit la fonction image $F(x,y)$, le gradient de F en un point (x,y) est un vecteur bidimensionnel qui s'écrit :

$$\vec{G}=\begin{bmatrix} \frac{\partial F}{\partial X} \\ \frac{\partial F}{\partial Y} \end{bmatrix} \tag{I.3}$$

Le gradient $\vec{G}$ est calculé en chaque point (x,y) de l'image, son orientation va dans le sens de la variation maximale de *F* et sa norme est généralement donnée par :

$$|\vec{G}| = \left(\left(\frac{\partial F}{\partial X}\right)^2 + \left(\frac{\partial F}{\partial X}\right)^2\right)^{1/2} \tag{I.4}$$

Localement, la norme du gradient est d'autant plus élevée que les courbes de niveau sont rapprochées et de valeurs différentes.

La direction de $\vec{G}$ par rapport à l'axe des X est donnée par l'angle :

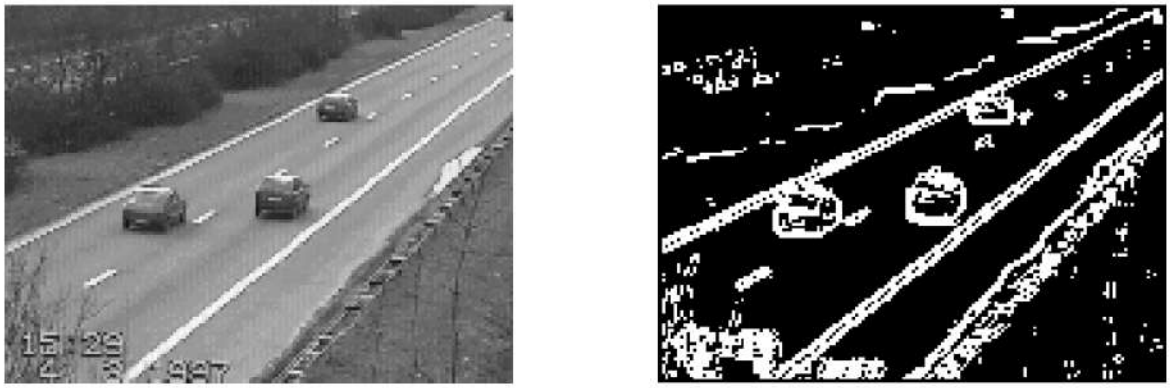
$$\vartheta = arctg\left(\frac{\partial F/\partial Y}{\partial F/\partial X}\right) \tag{I.5}$$

Le calcul du gradient est utile pour mettre en évidence les contours. Le point (x,y) est un contour si la norme est supérieure à un seuil.

- **Opérateurs de Robert**

$$G_X = \begin{matrix} -1 & 0 \\ 0 & 1 \end{matrix} \qquad G_Y = \begin{matrix} 0 & -1 \\ 1 & 0 \end{matrix}$$

On fait la convolution de ces masques avec l'image originale et on trouve les résultats suivants.



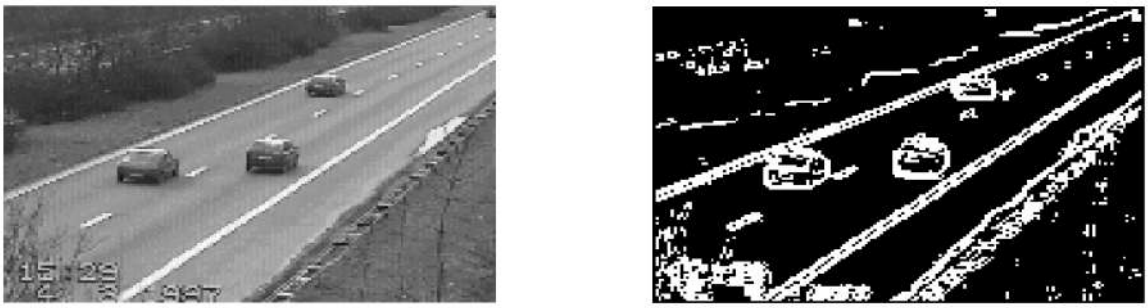
(a) (b)

Figure I.4Exemple d'application du filtre Robert, (a) : image d'origine, (b): contour de Robert

- **Opérateurs de Sobel et de Prewitt**

$$G_X = \begin{matrix} 1 & 0 & -1 \\ c & 0 & -c \\ 1 & 0 & -1 \end{matrix} \qquad G_Y = \begin{matrix} 1 & c & 1 \\ 0 & 0 & 0 \\ -1 & -c & -1 \end{matrix}$$

c = 2 pour *Sobel*, **c**=1 pour *Prewitt*.



(a) (b)

Figure I.5 Exemple d'application du filtre Sobel, (a) : image d'origine, (b) : contour de Sobel.

Les opérateurs de *Sobel* et *Prewitt* sont fréquemment utilisées, pour leur simplicité et leur faible coût d'évaluation, ils sont très performants dans les scènes de faible bruit et texture.

I.7 Segmentation par morphologie mathématique

Le principe de la morphologie mathématique repose sur le principe suivant:
On considère un élément structurant à centre G, cet élément structurant et formé d'un ensemble de points forme une forme géométrique. L'élément structurant constitue le concept de base de la morphologie mathématique. Il est défini par un ensemble de points constituant une forme géométrique plane. Le centre G est l'élément principal qu'on pose sur le pixel pour faire appliquer la morphologie binaire. Donc la segmentation par morphologie mathématique consiste à reconstruire l'objet à segmenter (isolé) par l'élément structurant.

a. Dilatation

On fait passer le centre de l'élément structurant sur tous les pixels de l'image et seuls les pixels pour lesquels l'intersection entre l'élément structurant et la zone à dilater seront gardé pour construire la zone dilatée.
Elle consiste à éliminer les points noirs isolés au milieu des parties blanches.
Soit E^N un espace euclidien à N dimension et soit I et S deux sous-ensembles de E^N, la dilatation de I par S est définie à partir de l'addition de Minkowski par :

$$I \oplus S = \{ c \in E^N / c = i + s , i \in I , s \in S \} \quad \text{(I.6)}$$



(a) (b)

Figure I.6 Exemple d'application de la dilation, (a) : image binaire, (b) : image dilatée.

b. Erosion

Dans l'opération de construction on pose le centre G de l'élément structurant sur les pixels (appartenant à la région à segmenter) et on vérifie si tout l'élément structurant est

inclus totalement dans l'objet à segmenter. L'opération d'érosion consiste à prendre que les pixels pour lesquels l'élément structurant est inclus dans l'objet.

Dans l'opération d'érosion les points blancs isolés disparaissent. Soit E^N un espace euclidien à N dimension et soit I et S deux sous-ensembles de E^N, l'érosion de I par S est définie à partir de la soustraction de Minkowski par :

$$I\Theta S = \{ c \in E^N / i = c + s , i \in I , \forall s \in S \} \quad \text{(I.7)}$$



(a) (b)

Figure I.7 Exemple d'application de l'érosion, (a) : image binaire, (b) : image érodée.

c. Ouverture

L'ouverture de I par S notée IoS est le résultat d'une érosion de I suivie d'une dilatation de l'ensemble érodé par le même élément structurant.

$$IoS = (I\Theta S) \oplus S \quad \text{(I.8)}$$

L'ouverture adoucit les contours et élimine les pics aigus.

d. Fermeture

C'est l'opération duale de l'ouverture, notée I•S, c'est le résultat d'une dilatation suivie d'une érosion en utilisant le même élément structurant.

$$I \bullet S = (I \oplus S) \Theta S \quad \text{(I.9)}$$

La fermeture fusionne les coupures étroites, élimine les petits trous, et comble les vides sur les contours.

I.8 Seuillage

L'opération non linéaire dite de seuillage simple consiste à mettre à zéro tous les pixels ayant un niveau de gris inférieur à une certaine valeur (appelée seuil, en anglais *threshold*) et à la valeur maximale (souvent la valeur 1) les pixels ayant une valeur supérieure.

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

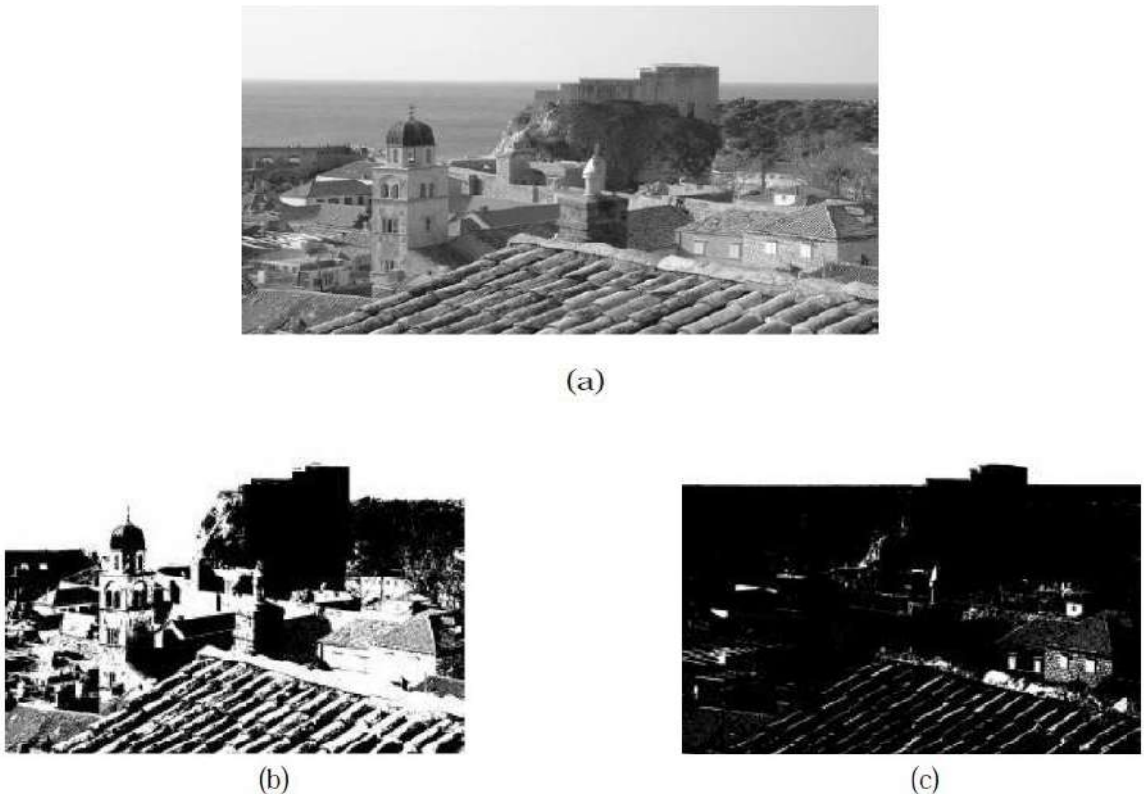


Figure I.8 Opération de seuillage. (a) : une image originale en 256 niveaux de gris.

Cette convention de seuillage s'appelle seuillage supérieur (*threshold above*), ainsi, on pourra définir le seuillage inverse dit inférieur (*threshold below*). Il est encore possible de définir deux valeurs de seuil, Respectivement borne inférieure et borne supérieure, afin de mettre à la valeur maximale les pixels ayant une valeur comprise entre les bornes et à zéro l'ensemble des autres valeurs. Il s'agit d'un seuillage intérieur (*threshold inside*), et par opposition, on définit le seuillage extérieur (*threshold outside*) qui est son inverse.

I.9 Définition d'une vidéo

La vidéo est une succession d'images animées défilant à une certaine cadence afin de créer une illusion de mouvement pour l'oeil humain. Elle peut être analogique (signal continu d'intensité de luminance) ou numérique (suite de trames ou images).

On peut distinguer deux types de vidéo:

- **la vidéo entrelacée**: où chaque image est formée de deux champs entrelacés. Le premier champ contenant uniquement les lignes impaires de l'image, le second champ, quant à lui, contenant les lignes paires (les signaux de télévision nord-américain et européen sont entrelacés).
- **la vidéo progressive:** dans une vidéo progressive, contrairement au mode entrelacé, toutes les lignes d'une image sont lues en une passe (vidéo numérique).

I.10 Résolution temporelle d'une vidéo

La résolution temporelle d'une vidéo est définie par le nombre d'images défilant par seconde.

Afin d'éviter les désagréments dus aux papillotements, et prenant en considération les spécificités de l'œil humain qui garde une image environ 10 micro secondes au niveau de sa rétine ; les images doivent défiler à une certaine cadence:

- environ 24 images/s pour un film de cinéma.
- 25 images/s pour la télévision européenne.
- environ 30 images/s pour la télévision nord-américaine et japonaise.

I.11 Séquence d'image

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'oeil 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.

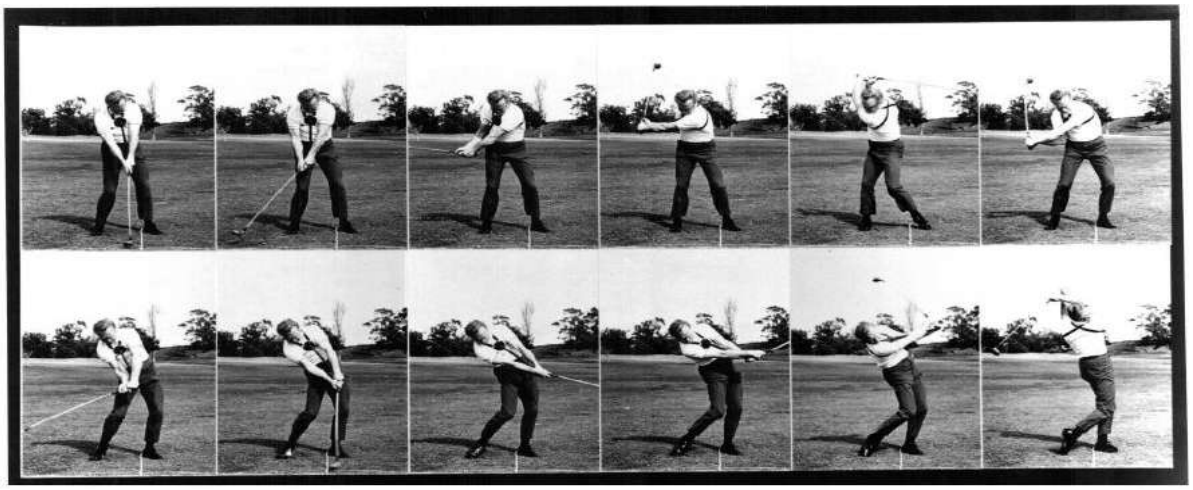


Figure I.9 Exemple d'une séquence d'images.

La considération de séquences d'images numériques ordonnées dans le temps introduit un champ spécifique d'investigation en analyse d'image, à savoir l'extraction d'informations de mouvement et plus généralement d'informations spatio-temporelles. Ce type d'études a émergé dans les années 70 à travers trois applications principales : le codage d'images de télévision et notamment la conception de codeurs avec compensation de mouvement, l'imagerie météorologique avec l'estimation de champs de vent via des traceurs nuageux, et le suivi de cibles mobiles dans un contexte militaire.

I.12 Analyse du mouvement dans une séquence d'images

L'analyse des séquences vidéo est l'un des sujets de recherche les plus dynamiques. Peut-être que ceci découle de la richesse en informations que présente une scène ou encore de la variété des applications (médicales, sécuritaires, militaires, ...etc.). Les spécialistes du domaine ont tenté, et tentent encore d'imiter via le système caméra/calculateur le comportement humain en termes de vision. Pour y parvenir différents axes ont été développés. Ainsi apparaît le terme de " groupement spatio-temporel " **[2]**.

C'est le terme qui englobe à la fois l'identification des objets et la détection du mouvement.



(a) (b)

Figure I.10 Exemple de vidéosurveillance, (a) : les caméras de surveillance, (b) : dans salle de contrôle : les hommes sont aidés par la machine dans la tâche de détection.

L'analyse de mouvement dans une séquence d'images doit être posée comme un problème d'estimation et de segmentation, puisqu'il s'agit d'appréhender des informations partielles observables et discontinues. C'est un problème particulièrement difficile, mais dont la résolution est cruciale pour la plupart des taches en analyse de scène dynamique. Ce problème peut se présenter en fait sous plusieurs variantes, suivant que l'objectif prioritaire se trouve être l'obtention d'une mesure dense ou paramétrique du mouvement, d'une partition de l'image en régions, ou l'extraction d'entités pertinentes L'analyse du mouvement dans une séquence d'images repose soit sur la détection du mouvement soit sur la segmentation basée mouvement soit sur la poursuite de mouvement.

I.12.1 Détection de mouvement

La détection du mouvement dans une séquence d'images consiste à distinguer les zones fixes et mobiles d'une scène. Comme le mouvement d'objets induit des différences temporelles entre images, sa détection s'appuie sur l'étude des variations temporelles de la fonction de luminance : si le capteur est fixe et que l'éclairement de la scène varie peu, alors nous supposons que toute variation temporelle de l'intensité est liée au mouvement d'un objet ou à du bruit

.

- **Détection de mouvement (Motion détection)**

Ce terme est le plus générique, il indique uniquement que l'on parle d'une méthode qui a pour objet de trouver en quels points de l'image un mouvement a eu lieu. Un algorithme ayant cet objectif fournit en sortie une variable quantitative (quantité de mouvement) ou qualitative

(booléenne) pour tout pixel de chaque image d'entrée. Toutes les méthodes présentées ci-après rentrent dans cette catégorie.

- **Estimation du mouvement (Motion estimation)**

Cette notion inclut la précédente en y ajoutant la contrainte que le résultat fourni doit être quantitatif. Ce terme est surtout utilisé par les auteurs travaillant sur l'estimation du flot optique ; dans ce cas, le résultat de l'estimation est un champ de vecteurs à deux dimensions représentant la projection sur le plan de l'image du mouvement réel tridimensionnel ayant lieu dans la scène. Pour un état de l'art détaillé et une étude comparative des méthodes classiques de calcul du flot optique, voir **[3]**.

- **Modélisation de l'arrière-plan (Background Modeling)**

Cette catégorie regroupe toutes les méthodes de détection de mouvement qui consistent à créer un modèle de l'arrière-plan de la scène filmée (sans aucun objet mobile). Ce modèle peut être une image créée à partir des pixels observés à différents instants de la séquence vidéo **[4]**.

- **Soustraction de l'arrière-plan (Background substraction)**

La soustraction de l'arrière-plan est l'opération qui suit logiquement la modélisation de l'arrière-plan afin d'obtenir une détection de mouvement. Si le modèle de l'arrière-plan est une image, une différence en valeur absolue entre ce modèle et l'image courante est effectuée afin d'obtenir une détection de mouvement.

- **Segmentation de (par le) mouvement (Motion [-based] segmentation)**

Cette tâche va au-delà de la détection de mouvement puisqu'il s'agit de segmenter chaque image en régions qui présentent une homogénéité du mouvement apparent. Cette opération est généralement réalisée à partir d'une estimation du flot optique ou des dérivées spatiotemporelles de l'intensité lumineuse.

La détection du mouvement contient les méthodes de différences d'images (approche directe), de corrélation ou de recherche dans l'espace des paramètres.

Les méthodes de bas niveau exploitent la comparaison pixel à pixel, ou petite région à petite région entre deux images consécutives d'une séquence. Elles permettent de déterminer les régions de variations de l'image dans le temps. Elles nécessitent soit une caméra fixe, soit un recalage préalable dans le cas d'un observateur mobile, afin de détecter uniquement les zones

de mouvement dans la scène. Elles sont généralement limitées aux mouvements d'objets rigides et au cas de petits déplacements.

I.12.2 Définition d'un objet vidéo

Définir un objet vidéo est une chose difficile. Intuitivement, un objet vidéo est défini par une forme, une texture et un mouvement (rigide ou non rigide). Cependant, la notion d'objet vidéo est beaucoup plus descriptive qu'une simple région, un objet vidéo peut par exemple être un modèle 3D.

La notion d'objet ne fait pas forcément référence à un objet du monde réel. En effet, dans le domaine de la vidéo, un objet n'est pas nécessairement un objet d'une scène 3D mais plutôt le résultat de l'analyse de la projection d'un monde 3D sur un plan. Ainsi, un objet vidéo est défini comme une région de la vidéo conforme à un modèle. On peut par exemple avoir pour modèle :

- un modèle de mouvement.
- un modèle d'objet physique.

La notion d'objets est alors subjectivement définie par rapport aux modèles utilisés. On constate que pour pouvoir trouver des objets vidéo, il est nécessaire de proposer des modèles auxquels ces objets répondront **[5]**.

I.12.3 Suivi d'objets dans la vidéo

Le suivi d'objets quelconques dans une séquence vidéo réelle est une tâche très délicate. Surtout quand ces objets sont non-rigides, le fond de la scène n'est pas fixe et dans le cas de plusieurs objets mobiles dans la même scène. Lorsqu'il s'agit de suivre des objets particuliers comme le visage, la main, le bras. La tâche est plus simple car le modèle et les contraintes de variation de ces objets sont connus a priori. D'autres critères de type géométrique et/ ou statique peuvent aussi être introduits dans l'identification de ces objets.

I.13 Conclusion

Dans ce chapitre nous avons abordé certaines définitions primaires concernant le traitement et la transformation d'image d'un état à l'autre comme le filtrage, la segmentation…etc. et nous avons vu les notions de base de l'analyse du mouvement dans une séquence d'image.

Chapitre II :

Détection d'objets en mouvement

II 1.Introduction

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...etc.

Les techniques de détection du mouvement peuvent être considérées comme un prétraitement permettant de réduire la quantité d'informations à analyser. Un objet en mouvement est détecté si sa position change relativement à celle d'un ensemble d'objets statiques ou s'il est localisé dans une image acquise à l'instant t de la séquence à une position différente de celle qu'il occupait dans l'image précédente. Le mouvement d'un objet dans la séquence d'images peut donc être mis en évidence en comparant le contenu d'au moins deux images issues de la séquence.

II 2. Détection d'objets en mouvement par méthode de 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. De nombreuses techniques de segmentation ont été proposées. Ces techniques ont toutes en commun la minimisation d'une fonctionnelle d'énergie. Ainsi quel que soit l'algorithme utilisé, on peut s'attendre à obtenir sensiblement les mêmes résultats. Le choix d'un algorithme est alors fonction de ses avantages et de ses inconvénients (sensibilité par rapport à l'initialisation, vitesse d'exécution, complexité, convergence ...).

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 al **[6]** 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). Cette méthode a montré des problèmes de sur-segmentations d'une même région. Afin de remédier à ce problème une étape de fusionnement des régions est alors proposée. Le schéma synoptique de la figure (II.1) résume les différentes étapes de la segmentation spatiale qui seront détaillées par la suite.

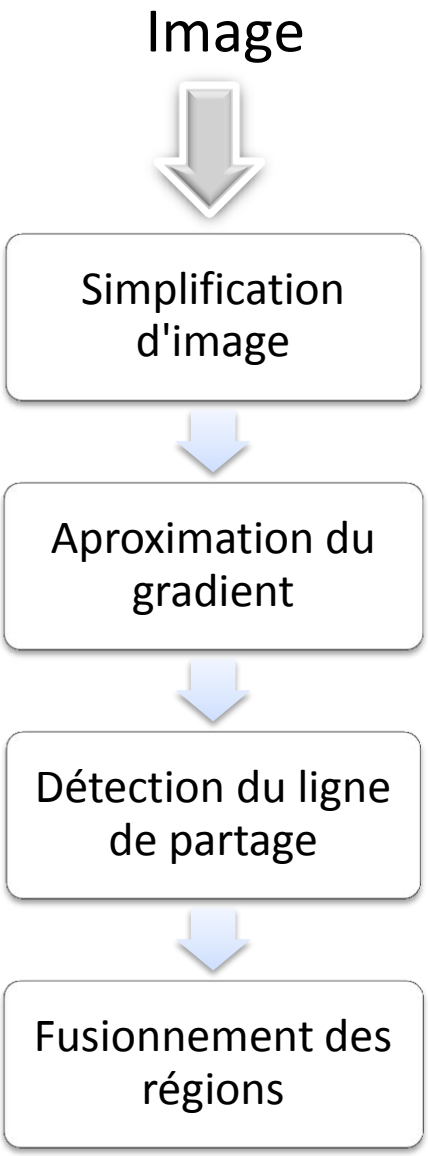


Figure. II.1 Schéma synoptique de la segmentation spatiale

a. Simplification d'image

Pour la segmentation spatiale, les images sont d'abord simplifiées. Des filtres morphologiques **[7]** sont conçus pour lisser les images bruitées par des opérateurs d'ouverture et de fermeture utilisant un élément structurant donné. On peut noter qu'afin de simplifier l'image originale l'opération de l'ouverture enlève les composants clairs et l'opération de fermeture élimine les composants foncés.

b. Approximation du gradient

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)

La détection des lignes de partage des eaux (LPE) est une technique classique de segmentation d'image **[8]**. 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 d'atteindre un pixel d'altitude inférieure sans avoir une montée.

Considérons maintenant le problème hydraulique suivant : Supposant qu'une goutte d'eau tombe sur le relief, elle va s'écouler, le long de celui-ci jusqu'à atteindre un minimum local.

On appelle bassin de capture, l'ensemble des pixels tel que si une goutte d'eau y tombait, elle se dirigera vers un même minimum local. Ceci est illustré sur la figure (II.2) ; les parties en gras sont les minimum locaux mi les lignes en pointillées (lignes de partage des eaux) délimitent les bassins de captures BCi

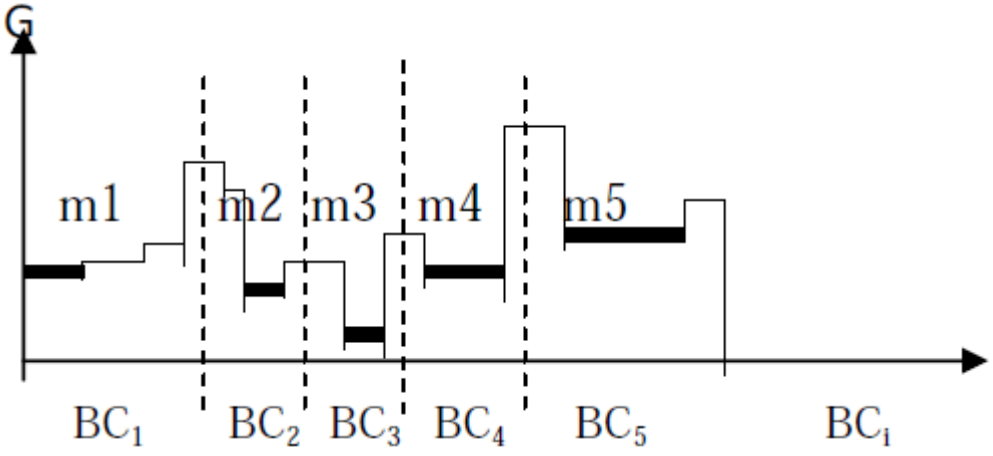


Figure. II.2 Les minima locaux mi et les bassins de captures BCi associés.

Les bassins de captures sont connexes et forment une partition de l'image. La frontière de ces bassins est appelée ligne de partage des eaux (watersheds). Cette frontière est souvent définie à un pixel près.

Pour segmenter une image, on utilise en règle générale les watersheds sur l'image gradient ou sur toute autre fonction d'énergie. Par exemple les variations d'intensités ou de texture localisée se traduisent par les maxima locaux du gradient, des crêtes en termes de relief. La valeur des maxima du gradient est faible dans des parties homogènes de l'image. Ces parties correspondent à des minima locaux.

L'algorithme de ligne de partage des eaux LPE est extrêmement sensible au gradient bruité qui rapporte beaucoup de bassins de capture, ayant pour résultat une sur-segmentation. Pour cela un algorithme de fusionnement des régions (Region merging) basé sur l'homogénéité d'intensité devrait être introduit pour résoudre le problème de sur segmentation.

d. Fusion des régions

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. Cela revient à modéliser le problème par une fonctionnelle d'énergie qu'il faut minimiser. Le choix de fusionner deux régions peut alors être fonction de la variation d'énergie avant - après fusion ou fonction de la similarité entre deux régions. Il y a donc une notion de seuil qui intervient pour stopper les fusions. Différents seuils peuvent être utilisés comme le seuil sur la variation d'énergie, celui sur la similarité ou celui sur le nombre total de fusions.

❖ **Intérêt de la fusion**

- 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 (GAR)**

Les techniques par graphes d'adjacence **[9]** sont beaucoup utilisées sur les régions sur segmentées. Ces sur segmentations sont par exemple le résultat d'un prédicat trop sévère dans la phase de segmentation ou d'un algorithme très sensible aux variations locales comme la technique des lignes de partage des eaux.

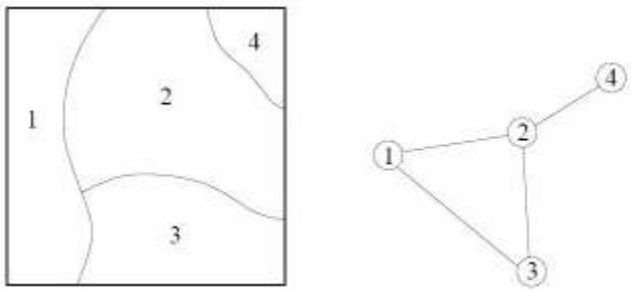


Figure II.3 Partitions et graphes d'adjacences associés.

L'idée de base consiste à plonger les régions obtenues dans une structure de graphe où 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.

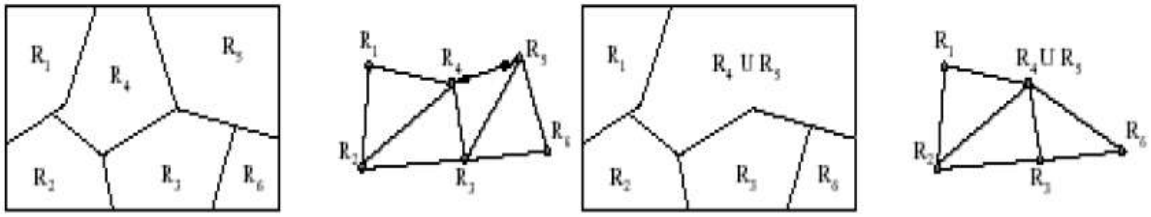


Figure II.4 Exemples de fusion de deux régions à partir d'un GAR de la partition.

Dans le domaine spatio-temporel, la fusion sur un GAR est empirique et requiert de calcul lourd. Afin de remédier à ce problème on procède à la détection de mouvement par la différence inter images puis on applique un nouveau schéma de détection des contours en mouvement.

II.3 Détection par différence d'images

II.3.1 Approche région

a. Avec image de référence

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'éclairements que les images avec lesquelles elle sera comparée.

L'image de différence notée *D* devient alors :

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

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 :

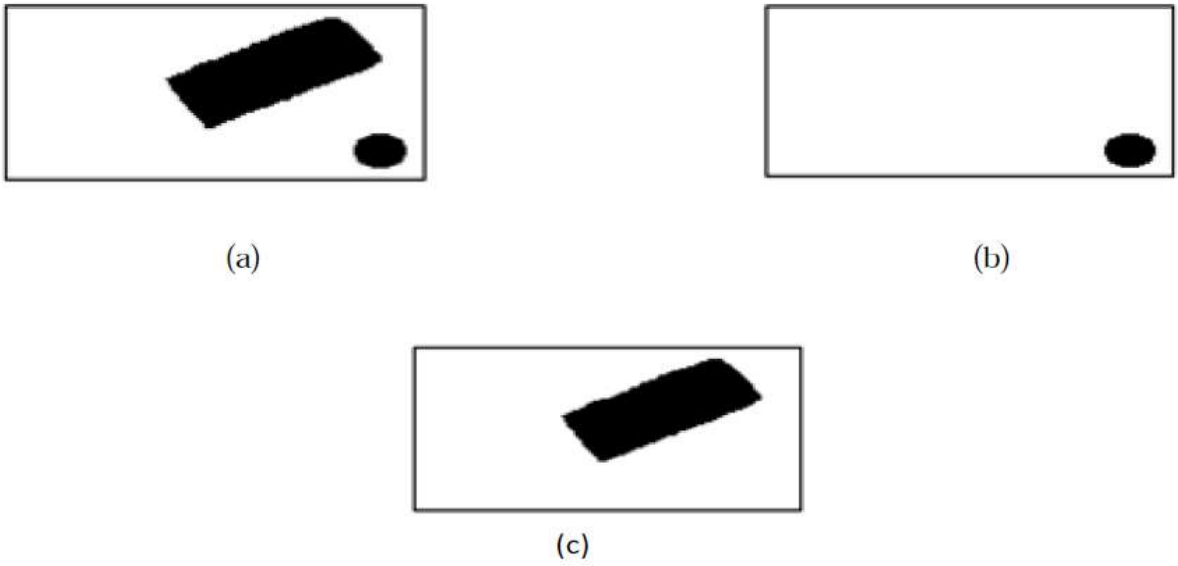


Figure. II.5 Technique de différence avec image de référence, (a) l'image à l'instant t, (b) l'image de référence, (c) l'image de différence.

- 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 ou aucun objet n'apparait, mais ce n'est pas toujours le cas. La construction d'une image de référence est donc une question délicate abordée dans la littérature par de nombreux auteurs. L'image de référence doit en permanence être identique à l'environnement statique perçu dans chaque

image de la séquence. Dans **[10]**, Jain propose un algorithme qui repose sur l'analyse de l'image différence. Dans **[11]**, 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 :

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

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

b. Sans l'image de référence :

La différence temporelle s'exprime par :

$$F_D\,(I_C, I_P) \;=\; |\, I_C - I_P \,| \qquad \text{(II.3)}$$

Ou : F_D **:** image de différence

Et I_C image courante, et I_P image précédente.

L'image $F_D\,(I_C, I_P)$ 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 (II.6) la différence fait apparaître quatre zones distinctes.

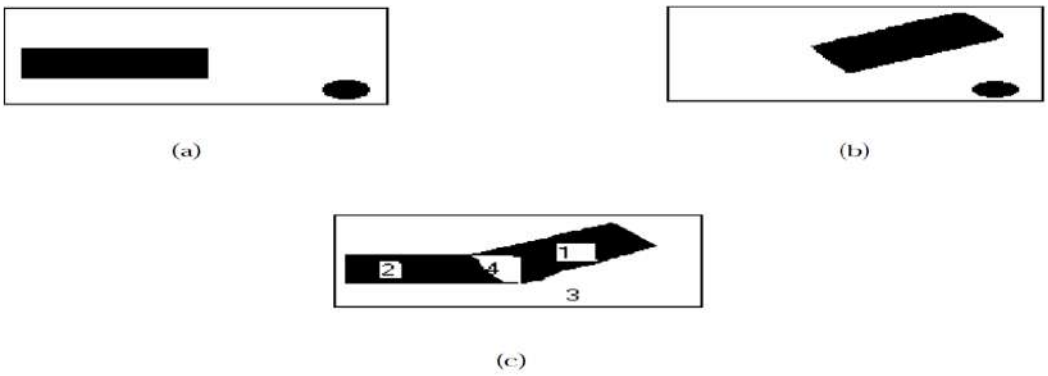


Figure. II.6 Technique de différence sans image de référence, (a) image précédente, (b) image courante, (c) image de différence.

- 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.3.2 Approche contour

Jusqu'à maintenant, la détection des changements temporels a été abordée sous l'aspect extraction de région en mouvement. On propose dans ce paragraphe une approche légèrement différente, qui consiste à extraire les contours en mouvement.

Nous noterons $G\ (I)$ l'image des contours (gradients) de l'image I, et $CM\ (I)$ l'image des contours en mouvement de l'image I.

On distingue plusieurs variantes de la méthode :

a. Proposition de HAYNES [12]

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

$$F_D\ (I_C, I_P)\ =\ |\ I_C\ -\ I_P\ | \qquad \text{(II.4)}$$

$$CM(I_C)\ = F_D\ (I_C, I_P).\, G(I_C). \qquad \text{(II.5)}$$

La différence inter-image est considérée comme un détecteur de mouvement, l'opérateur $G(I_C)$ est considéré comme un détecteur de contour, et le produit est alors un détecteur de contours en mouvement.

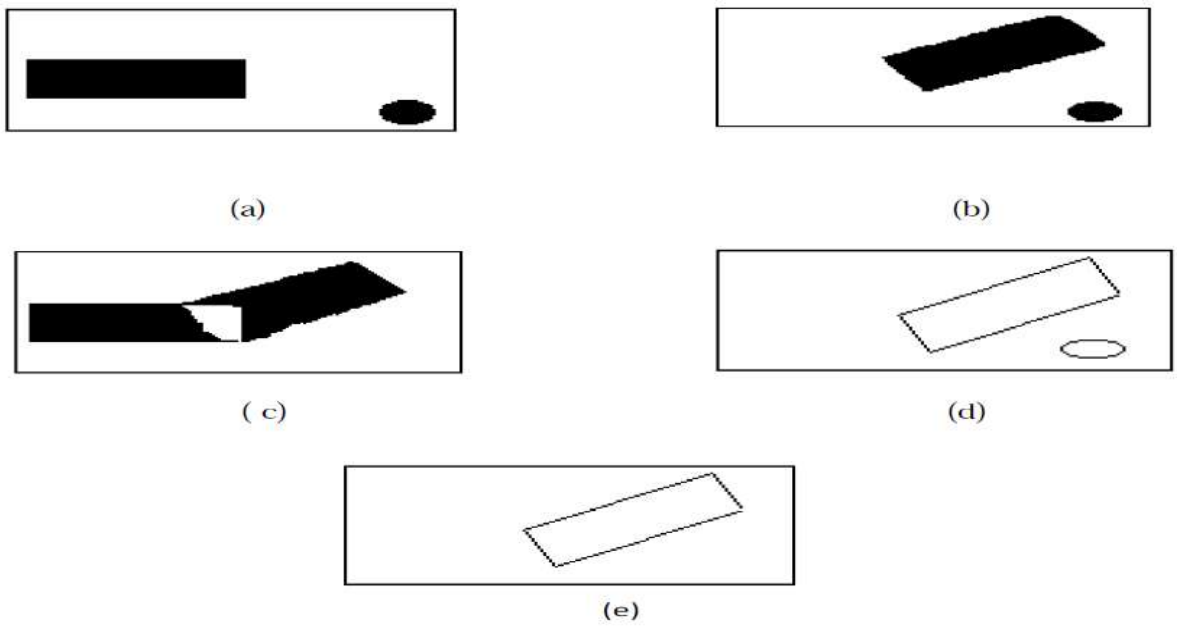


Figure II.7 Extraction des contours en mouvement par la proposition de Haynes, (a) image précédente, (b) image courante, (c) image de différence, (d) contour d'image courante, (e) contour en mouvement.

Bien que sensible au bruit, un seuillage approprié de $CM(I_C)$ 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 ($G(I_C)$ fort), la différence inter-images est voisine de

zéro et par conséquent, $CM(I_C)$ 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.

b. Proposition de STELMASZYK [13]

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 $G(I_C)$ et dans l'image des contours de la différence $G\ (F_D\ (I_C, I_P))$.

On obtient alors :

$$CM(I_C)\ =\ G(I_C).G\ (F_D(I_C, I_P\ )). \tag{II.6}$$

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 $G(I_C)$ contient les contours du fond qui étaient cachés par l'objet dans sa position précédente. L'image $G\ (F_D\ (I_C, I_P))$ contient aussi ces contours puisque $F_D\ (I_C, I_P)$ contient les points du fond en ces endroits-là, par conséquent, ces points du contour du fond, communs aux deux images, apparaissent dans $CM(I_C)$.

Pour éviter cela, Vieren propose de mettre en jeu une troisième image.

c. Proposition de VIEREN [14]

Le détecteur proposé est le suivant :

$$CM(I_C)\ =\ G\ (FD(I_C, I_P)).G\ (FD(I_C, I_S)) \tag{II.7}$$

Ou I_S représente l'image suivante :

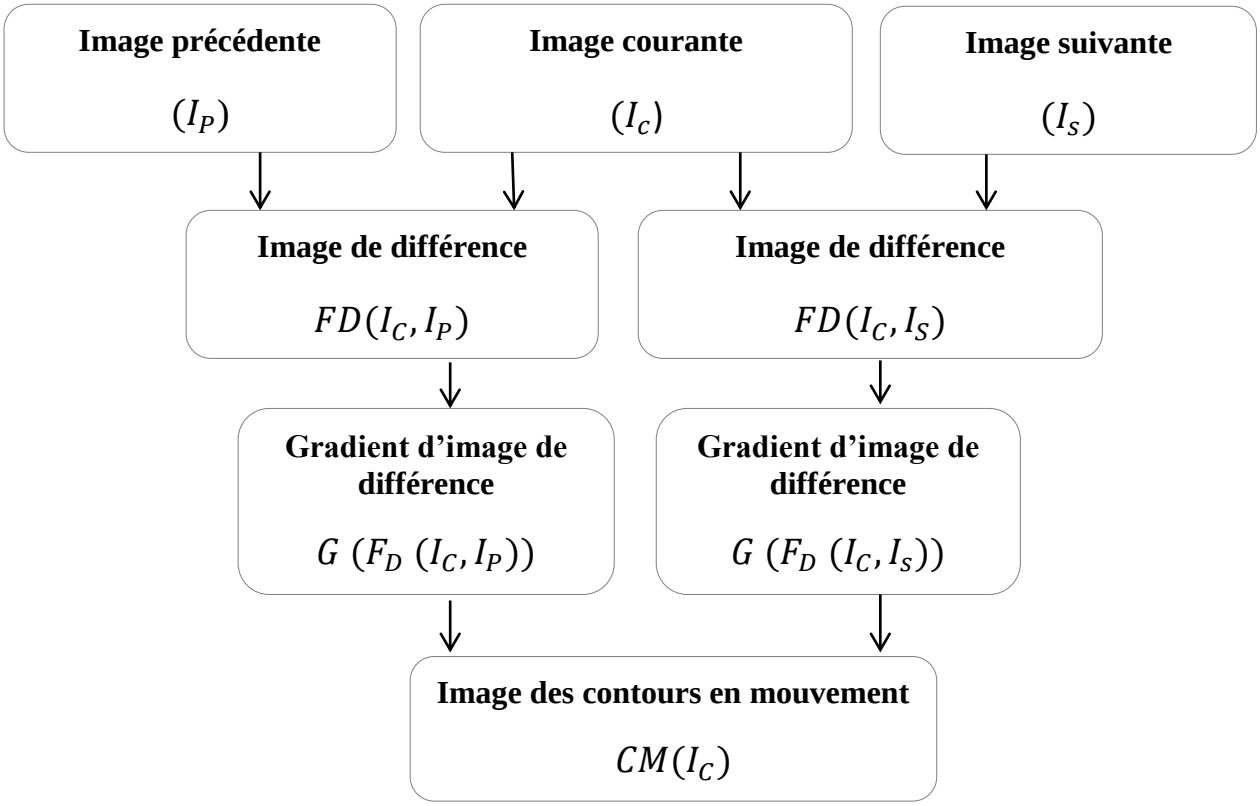


Figure. II.8 Extraction des contours en mouvement par la proposition de Vieren.

La suppression des fausses détections est assurée, car il ne peut pas y avoir de zone de l'objet commune aux images I_P et I_S qui soit extérieure aux contours de l'objet dans sa position courante.

Néanmoins, l'inconvénient majeur de toutes les méthodes ci-dessus provient du fait qu'elles mettent toutes en jeu une multiplication d'un gradient qui ne peut fonctionner correctement que dans les parties bien contrastées. La multiplication par une petite valeur de gradient affaiblit le résultat.

A partir de ces considérations, Orkisz propose de comparer les images des gradients.

d. Proposition d'ORKISZ [15]

La soustraction du gradient de l'image précédente à celui de l'image courante élimine les contours stationnaires et affecte peu les contours des objets en mouvement dans l'image courante. Par conséquent, pour les contours stationnaires, le résultat de cette soustraction est

différent des valeurs du gradient de l'image courante, par contre, il est très proche pour les contours en mouvement.

Pour détecter les objets en mouvement, il faut donc chercher les endroits où le gradient de l'image est suffisamment grand (présence de contours) et à peu près égal à la différence des gradients des images courante et précédente.

On compare donc :

$$G(I_C) \text{ à } |G(I_C) - G(I_P)| \quad \text{(II.8)}$$

Partant du même principe de cette proposition (éviter le produit gradient), une autre méthode basée sur la mesure des contours en mouvement (mcm) qui est proposé.

II .4 Détection des contours en mouvement par (mcm)

L'objectif de cette étape est de détecter les contours des régions en mouvement. Donc la méthode proposée **[16]**, prend en compte trois images successives I_P, I_C, I_S 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.

$$mcm = min\,(|G(I_P) - G(I_C)|, |G(I_C) - G(I_S)|) \quad \text{(II.9)}$$

La figure II.9 résume les différentes étapes nécessaires du calcul du mcm :

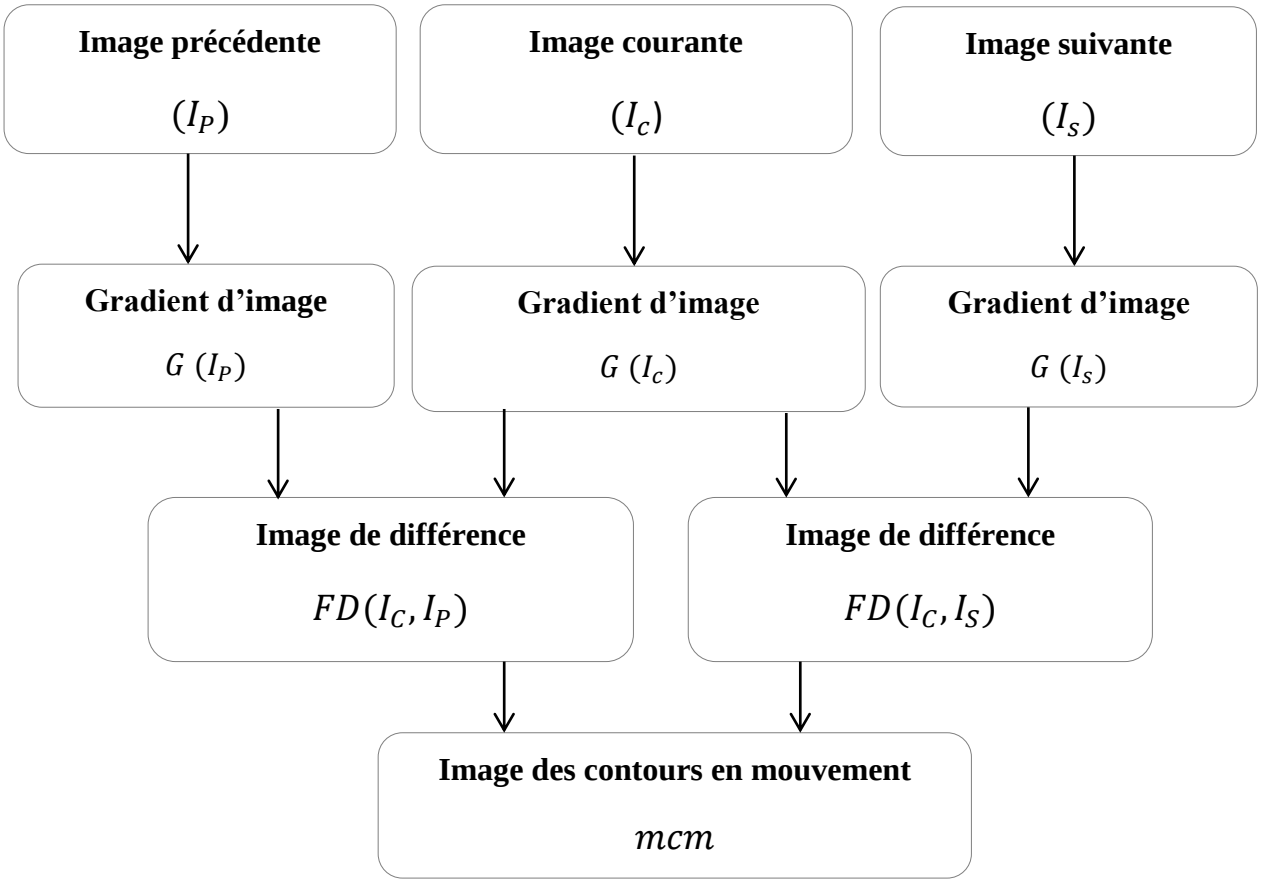


Figure II.9 Schéma synoptique de l'opérateur mcm.

Les contours des objets en mouvement sont ainsi extraits à l'aide d'un opérateur basé sur la différenciation des gradients spatiaux de trois images successives. Les contours forts de chaque image sont obtenus par le gradient.

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 satisfaite, donc dans ce cas-là on applique la méthode de différence entre deux images suivi par un seuillage.

Une étape supplémentaire est alors nécessaire pour agglomérer les différentes zones issues de la détection des changements et pour filtrer les détections parasites. Des traitements de type dilatation, érosion sont appliqués à la sortie des détecteurs des changements temporels. Ils permettent d'une part de nettoyer les images résultats et d'autre part, ils peuvent être utiles à la reconstruction des masques des objets en mouvement. Ces concepts font partis des techniques de morphologie mathématique.

II.5 Conclusion

Dans ce chapitre, nous avons présenté les techniques de détection des objets en mouvement. Nous différencions les méthodes de détection basées sur la segmentation par LPE et les méthodes de la différence inter-images, celles reposant sur une image de référence et celles sans image de référence. Dans ce contexte, on a proposé une méthode de différence entre deux images successives suivie par un seuillage et certaines techniques de morphologie mathématique lorsque le mouvement est faible (lent). Dans le deuxième cas si le mouvement est rapide, on opère une méthode pour extraire les contours des objets en mouvement basée sur la différenciation des gradients spatiaux de trois images successives.

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.

La problématique que nous proposons consiste à suivre un objet de forme et de texture quelconques soumis à un mouvement quelconque pouvant être de forte amplitude, dans des séquences d'images en 256 niveaux de gris. Un bref aperçu des algorithmes de suivi sera donc présenté, ainsi que les modèles de mouvement les plus utilisés.

III.2 Algorithmes de suivi

Le suivi d'objets dans une séquence d'images est un problème classique et largement étudié dans le domaine de l'analyse du mouvement, du fait de ses nombreuses applications parmi lesquelles on peut noter la surveillance vidéo, la robotique, la compression de vidéos, l'étude de la déformation d'objets, . . . Ainsi, de nombreux algorithmes de suivi ont été proposés ces dernières années. Nous allons, dans les sections suivantes, présenter une classification possible de ces algorithmes, ainsi que leur structure générale.

III.3 Classification des algorithmes de suivi

Les algorithmes de suivi peuvent globalement être classés en deux catégories :
Les algorithmes orientés mouvement, et ceux orientés modèle

III.3.1 Les algorithmes orientés mouvement

S'appuient sur des méthodes de regroupement de régions homogènes au sens du mouvement. Ils ont l'avantage d'être rapides, cependant, les régions suivies n'ont en général pas de signification physique globale puisqu'elles ont été obtenues à partir d'un critère d'homogénéité et non d'une représentation d'objet. Le principe de ces algorithmes est de partir d'une segmentation spatio-temporelle initiale **[17].** Cette dernière peut être obtenue de différentes façons : à partir de pixels individuels (techniques Markoviennes fournissant un champ dense **[18]**) de petits blocs de pixels (approche MPEG), d'une grille arbitraire, de l'image entière, ou encore d'une segmentation spatiale. Les paramètres de mouvement sont ensuite estimés à partir de cette segmentation, puis des fusions ou des divisions de régions sont réalisées selon un critère d'uniformité du mouvement. Cette segmentation est ensuite suivie au cours de la séquence. Ce type d'algorithme est utilisé pour la compression, le codage, l'interpolation et l'indexation de vidéos et la détection et le suivi de fouillis radar.

III.3.2 Les algorithmes orientés modèle

Contrairement aux précédents, ces algorithmes se basent sur le suivi des régions correspondant aux projections 2D des objets auxquels on s'intéresse, en les représentant soit par des modèles déformables 3D **[19]**, soit par des modèles déformables 2D **[20]**. Les régions suivies sont donc plus significatives, mais la complexité calculatoire est plus importante du fait de la nécessité de déterminer d'éventuelles rotations, translations, homothéties ou autres déformations.

Dans les deux cas, le suivi peut être réalisé par des méthodes basées soit sur les frontières, soit sur les régions. Il est à noter que les premières recherches dans le domaine s'appuyaient sur le suivi de primitives simples telles que des segments de contours ou des points de référence dans les objets, Cependant, ces techniques étaient peu fiables du fait de données trop éparses, et sensibles aux occlusions. C'est pourquoi les recherches se sont orientées vers des primitives plus complexes et plus globales telles que les contours et les régions.

III.3.2.1 Les méthodes basées sur les frontières

Elles reposent sur l'information apportée par le contour de l'objet à suivre. Leur avantage est d'être indépendant du type de mouvement ainsi que de l'objet. De plus, elles sont efficaces pour extraire les contours de l'objet avec précision. Elles sont donc particulièrement bien adaptées au suivi d'objets déformables. Ce sont en général des modèles de contours déformables tels que les contours actifs (snakes), les contours actifs géodésiques ou encore des modèles statistiques qui sont utilisés. Les principaux inconvénients de ces modèles sont leur instabilité due à la prise en compte d'une information locale, ainsi que leur sensibilité à l'initialisation.

En effet, le principe de ces algorithmes est de partir d'un contour initial et de converger de façon itérative vers le contour recherché, par minimisation d'une fonction d'énergie. Ils sont donc performants dans le cas de mouvements lents, mais mal adaptés aux déplacements importants.

III.3.2.2 Les méthodes basées sur les régions

Utilisent, quant à elles, l'information apportée par la région entière, comme par exemple des propriétés de texture ou de mouvement. Bien qu'ayant des difficultés à suivre avec précision les contours, ces méthodes sont plus robustes puisqu'elles utilisent une information

plus globale et qu'elles sont moins dépendantes de l'initialisation. Elles sont ainsi capables de gérer des déplacements importants. De tels modèles ont été développés dans **[21]** en utilisant des modèles de régions déformables géodésiques, et dans **[22]** en utilisant des contours actifs basés région. De plus, en associant au suivi de région une segmentation de régions basée mouvement **[23]**, il est même possible de s'affranchir de toute initialisation puisque cette segmentation détecte les objets en mouvement dans l'image.

Ces deux types de méthodes possèdent donc des qualités complémentaires. En les combinant, certains algorithmes permettent un suivi optimisé **[24]**.

III.4 Structure général des algorithmes de suivi

Le suivi des primitives présentées précédemment d'un instant *t* vers un instant *t + 1*, est généralement composé d'un minimum de trois étapes, après avoir calculé une segmentation spatiale ou spatio-temporelle initiale, selon le type d'algorithme de suivi considéré. Ces trois étapes sont les suivantes :

- Prédiction du mouvement des primitives de l'instant *t* à l'instant *t + 1*
- Ajustement de la segmentation prédite
- Estimation du mouvement des régions suivies de *t* vers *t + 1*

Nous allons brièvement considérer chacune de ces trois étapes

III.4.1 Phase de prédiction

Elle a pour objectif de relier la segmentation de l'image à l'instant *t* à celle de l'image à l'instant *t + 1*. Quatre modèles de prédiction peuvent être envisagés **[25]**:

a. Prédiction de mouvement nul

On considère que la primitive ne se déplace pas entre t et $t + 1$. Sa position prédite est donc identique à celle qu'elle avait à l'instant précédent.

b. Prédiction à court-terme simple

On prend uniquement en compte l'information du mouvement entre les deux images précédentes. Le terme court-terme vient du fait que dans ce cas on n'utilise aucun modèle temporel d'évolution.

c. Prédiction à court-terme lissée

On utilise alors un modèle d'évolution, tel que le filtre de Kalman. Le mouvement considéré est celui fourni par les équations de filtrage (ou lissage) de ce filtre.

d. Prédiction à long-terme

Dans ce cas, on prend en compte l'information du mouvement entre toutes les images précédentes. Les valeurs prédites sont alors issues de l'équation de prédiction du filtre de Kalman.

III.4.2 Phase d'ajustement

Une fois la phase de prédiction réalisée, nous disposons d'une segmentation prédite de l'image à l'instant $t + 1$. Cependant, cette dernière n'est en général pas parfaite. Il est donc nécessaire d'ajuster les frontières des régions de cette segmentation sur les frontières réelles.
C'est le rôle de la phase d'ajustement. Selon le type de primitives suivies, deux types de méthodes d'ajustement peuvent être envisagées :

a. Ajustement basé contour

Ces méthodes utilisent des modèles de contours déformables afin d'ajuster les frontières des régions prédites, en se basant uniquement sur des informations spatiales, telles que le gradient spatial, dans l'image courante.

b. Ajustement basé région

Cette approche peut s'appuyer, par exemple, sur une modélisation markovienne des zones d'occlusion **[26]**, ou encore sur des modèles de régions déformables **[27]**. Elle peut prendre en considération à la fois des informations spatiales dans l'image courante (gradient spatial, texture,...), et des informations de mouvement entre les images courante et précédente.

III.4.3 Phase d'estimation

Elle est réalisée après la phase d'ajustement et permet de déterminer quel a été le mouvement effectif des régions suivies entre les images aux instants t et $t + 1$. Il existe trois grands types de méthodes d'estimation du mouvement **[28]**:

a. Les méthodes par mise en correspondance

Elles cherchent à déterminer des similarités entre des points, des contours ou des régions, présents dans deux images successives. Pour cela, une fonction de corrélation doit être maximisée. Ce sont en général des régions constituées de petits blocs carrés qui sont considérées. Seules les translations sont tolérées, car ce modèle suppose que tous les pixels appartenant à une même région subissent un déplacement identique. Les normes MPEG, par exemple, ont recours à ces méthodes.

b. Les méthodes par transformées

Elles utilisent les effets du mouvement dans l'image sur les composantes ou les caractéristiques fréquentielles de transformées telles que la transformée de Fourier ou encore la décomposition en séries de Gabor. En effet, une translation spatiale, par exemple, modifie la phase de la transformée de Fourier de l'image.

c. Les méthodes différentielles

Elles reposent sur l'hypothèse d'invariance de la luminance des points au cours du temps, lors de leur déplacement. L'estimation est alors réalisée par la minimisation d'une fonction d'énergie mesurant la différence de luminance d'un point d'une image et son projeté dans l'image suivante.

III.5 Suivi de trajectoire d'objet en mouvement

Dans le suivi de vidéo, un objet peut être de nature quelconque. Tous les scénarios sont possibles à imaginer : un bateau en pleine mer, un joueur dans un stade avec son ballon, un véhicule en autoroute, un avion dans le ciel, une pièce à usiner dans une chaine de fabrication,… etc. Selon la nature de l'objet et l'objectif de l'application à réaliser, de différentes représentations de la forme (points, formes géométriques de bases, modèle articulé, silhouette,… etc.) et d'apparence (le modèle d'apparence active (AAM),… etc.) de l'objet sont utilisées permettant, ainsi, plusieurs algorithmes de suivi d'objet en mouvement.

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.1).

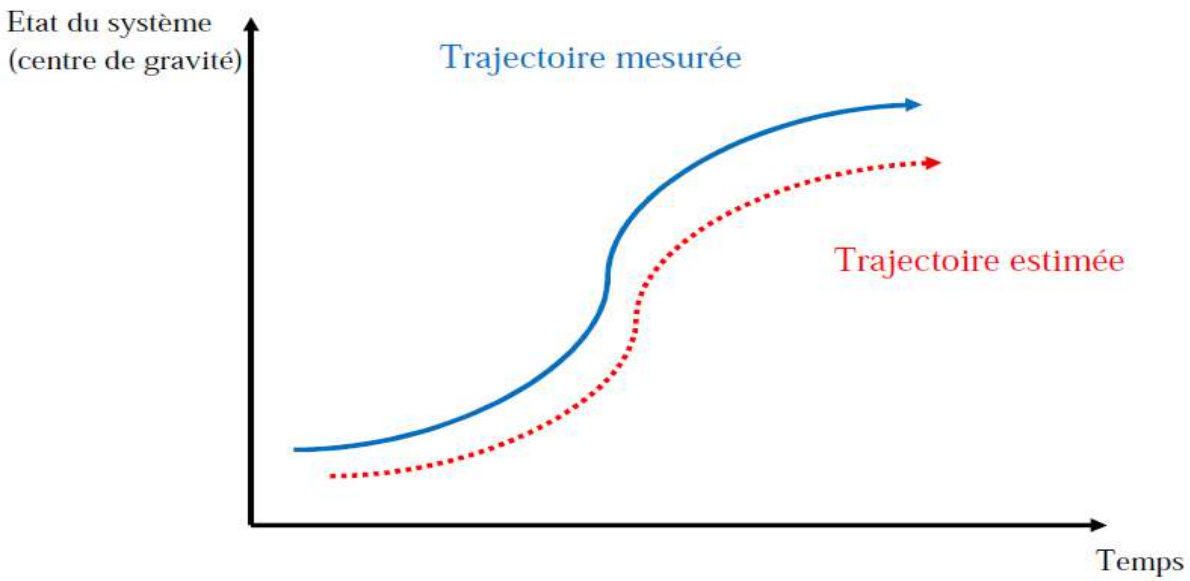


Figure III.1 Exemple de trajectoire mesurée et estimée.

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.

Les deux idées qui motivent l'utilisation des filtres stochastiques 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.6 Modélisation des trajectoires d'une cible

Pour modéliser la trajectoire d'une cible. nous pouvons affirmer que quel que soit la complexité de cette dernière, elle pourra toujours être représentée par une succession de trajectoires élémentaires canoniques tels que des segments de droites, des arcs de cercles, des arcs elliptiques, ... A partir de là, nous pouvons dire que pour déterminer la trajectoire suivie par une cible donnée durant un temps t il suffit de déterminer les paramètres des différentes trajectoires canoniques suivies par la cible durant ce temps.

Dans notre cas, nous allons restreindre les trajectoires canoniques choisies au segment de droites et à l'arc de cercle.

III.6.1 Modèle d'une droite (Trajectoire rectiligne) [29]

Nous modéliserons une droite de la manière suivante :

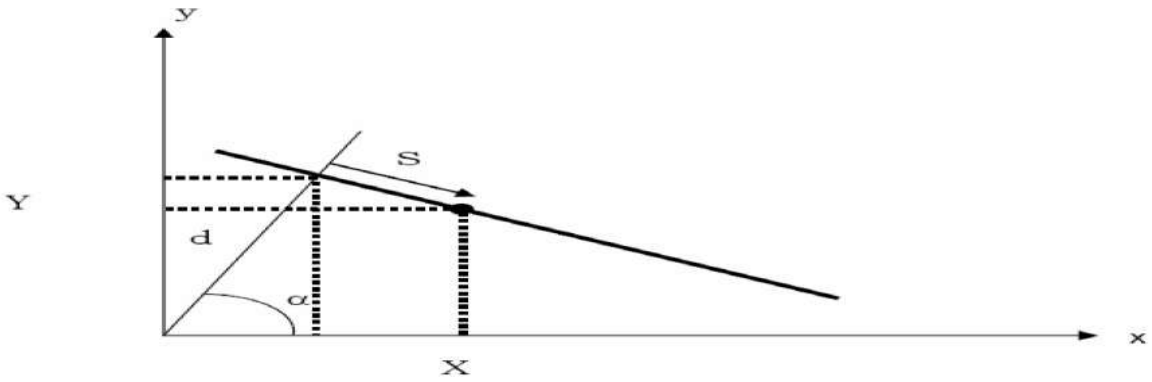


Figure III-2 Modélisation d'une droite.

La représentation ci-dessus permet de définir toute droite (mouvement rectiligne) par ses coordonnées plucklériennes d et α, où

- d est la distance perpendiculaire à la droite passant par l'origine des coordonnées.
- α est l'angle que fait l'axe des abscisses avec la droite reliant l'origine à sa projection orthogonale sur la droite considérée.

D'autre part, on représente le mouvement sur une droite quelconque par les grandeurs suivantes :

- La distance parcourue s.
- La vitesse de déplacement $v = \dot{s}$.

- L'accélération $a = \ddot{s}$.

Cette représentation permet un découplage entre le mouvement et la trajectoire suivie.

III.6.1.1 Mouvement rectiligne à vitesse constante

L'état de base d'un mouvement rectiligne à vitesse constante est :

$$\begin{cases} s = (t_k - t_{k-1})v + s \\ v = \frac{ds}{dt} = \dot{s} \end{cases} \quad \text{(III.1)}$$

Où: $(t_k - t_{k-1})$ est la différence de temps.

En temps discret, ces équations deviennent :

$$\begin{cases} s(k) = (t_k - t_{k-1})v(k-1) + s(k-1) \\ v(k) = v(k-1) \end{cases} \quad \text{(III.2)}$$

Posons $\begin{cases} x_1(k) = s(k) \\ x_2(k) = v(k) \end{cases}$

On obtient :

$$x(k) = \begin{cases} x_1(k-1) + Tx_2(k-1) \\ x_2(k) = x_2(k-1) \end{cases} \quad \text{(III.3)}$$

Avec $T = t_k - t_{k-1}$

L'équation du vecteur des paramètres est :

$$\theta(k) = \begin{cases} \alpha \\ d \end{cases} \quad \text{(III.4)}$$

Posons $\begin{cases} x_3(k) = \alpha \\ x_4(k) = d \end{cases}$

On obtient :

$$\theta(k) = \begin{cases} x_3(k) = x_3(k-1) \\ x_4(k) = x_4(k-1) \end{cases} \quad \text{(III.5)}$$

Parce qu'ils sont invariant dans le temps.

Donc l'état augmenté sera :

$$x(k) = \begin{bmatrix} x(k) \\ \theta(k) \end{bmatrix} = \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{bmatrix} \quad \text{(III.6)}$$

$$x(k) = \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{bmatrix} = \begin{bmatrix} x_1(k-1) + Tx_2(k-1) \\ x_2(k-1) \\ x_3(k-1) \\ x_4(k-1) \end{bmatrix} + V\ (k-1) \quad \text{(III.7)}$$

Avec *V(k)* : bruit d'état.

$$x(k) = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x(k-1) + V(k-1) \quad \text{(III.8)}$$

$x(k)$ est sous la forme *x(k+1) = Fx(k) + V(k).*

Ou *F* est la matrice de transition d'état :

$F \triangleq \exp(AT)$, avec $\dot{x}(t) = Ax(t) + \tilde{V}(t)$

La fonction de mesure est une fonction non linéaire reliant l'état de base avec les paramètres inconnus du mouvement rectiligne (droite) selon les deux équations suivantes :

$$\begin{cases} X = d\cos(\alpha) + s\sin(\alpha) \\ Y = d\sin(\alpha) - s\cos(\alpha) \end{cases} \quad \text{(III.9)}$$

Qui sont la position *X* et *Y* de l'objet poursuivi.

Ces équations peuvent être écrites en fonction de variables d'état comme suit :

$$y(k) = \begin{cases} X(k) = x_4(k)\cos\big(x_3(k)\big) + x_1(k)\sin\big(x_3(k)\big) \\ Y(k) = x_4(k)\sin\big(x_3(k)\big) - x_1(k)\cos\big(x_3(k)\big) \end{cases} \quad \text{(III.10)}$$

Donc *y(k)* est de la forme suivant : *y(k) = H(x(k)) + w(k).*

Avec *w(k)* : bruit de mesure.

La linéarisation de *h(x)* est réalisée en considérant une petite variation dans l'état :

$$H_x = \begin{bmatrix} \frac{\partial X}{\partial S} & \frac{\partial X}{\partial \dot{S}} & \frac{\partial X}{\partial \alpha} & \frac{\partial X}{\partial d} \\ \frac{\partial Y}{\partial S} & \frac{\partial Y}{\partial \dot{S}} & \frac{\partial Y}{\partial \alpha} & \frac{\partial Y}{\partial d} \end{bmatrix} \quad \text{(III.11)}$$

Dans ce cas on aura

$$H_x = \begin{bmatrix} \sin(\alpha) & 0 & -d\,sin\,(\alpha) + \mathrm{S}\cos(\alpha) & \cos(\alpha) \\ -\cos(\alpha) & 0 & d\cos(\alpha) + S\sin(\alpha) & \sin(\alpha) \end{bmatrix} \quad \text{(III.12)}$$

III.6.1.2 Mouvement rectiligne à accélération constante

L'état de base d'un mouvement rectiligne à accélération constante est :

$$\begin{cases} s(k) = s(k-1) + Tv(k-1) + \frac{1}{2}T^2 a(k-1) \\ v(k) = v(k-1) + Ta(k-1) \\ a(k) = a(k-1) \end{cases} \quad \text{(III.13)}$$

L'équation du vecteur des paramètres reste la même que le mouvement rectiligne à vitesse constante.

$$\text{Posons} \quad \begin{cases} x_1(k) = s(k) \\ x_2(k) = v(k) \\ x_3(k) = a(k) \\ x_4(k) = \alpha \\ x_5(k) = d \end{cases} \quad \text{(III.14)}$$

L'état augmenté est donc :

$$x(k) = \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \\ x_5(k) \end{bmatrix} = \begin{bmatrix} x_1(k-1) + Tx_2(k-1) + 0.5T^2 x_3(k-1) \\ x_2(k-1) + Tx_3(k-1) \\ x_3(k-1) \\ x_4(k-1) \\ x_5(k-1) \end{bmatrix} + V(k-1) \quad \text{(III.15)}$$

$$x(k) = \begin{bmatrix} 1 & T & 0.5T^2 & 0 & 0 \\ 0 & 1 & T & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} x(k-1) + V(k-1) \quad \text{(III.16)}$$

La fonction de mesure reste la même que le mouvement précédent et la seule différence se trouve dans l'ordre des variable d'état.

$$y(k) = \begin{cases} X(k) = x_5(k)\cos\big(x_4(k)\big) + x_1(k)\sin\big(x_4(k)\big) \\ Y(k) = x_5(k)\sin\big(x_4(k)\big) - x_1(k)\ \cos\big(x_4(k)\big) \end{cases} \quad \text{(III.17)}$$

Donc *y(k)* est de la forme suivant $y(k) = H(x(k)) + w(k)$

Avec *w(k)* : bruit de mesure.

La matrice de mesure linéarisée H_x se déduit par dérivation de *h(x).*

$$H_x = \begin{bmatrix} \sin(\alpha) & 0 & -d\ sin\,(\alpha) + S\cos(\alpha) & \cos(\alpha) \\ -\cos(\alpha) & 0 & d\cos(\alpha) + S\sin(\alpha) & \sin(\alpha) \end{bmatrix} \quad \text{(III.18)}$$

III.6.2 Modèle d'un cercle (Trajectoire circulaire) [29]

Comme pour le cas d'un mouvement rectiligne, les mouvements circulaires peuvent être représentés en fonction des coordonnées puicklériennes R_0, X_0 et Y_0.

Où: R_0 est le rayon du cercle.

(X_0, Y_0) est le centre de ce cercle.

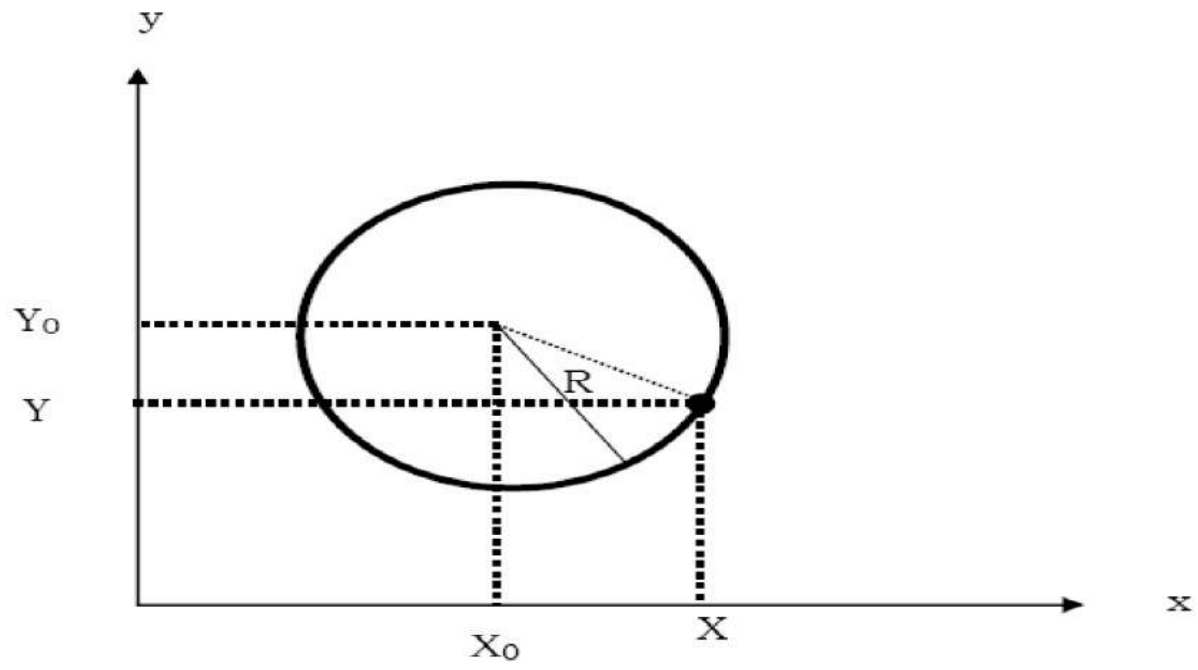


Figure III-3 Modélisation d'un cercle.

III.6.2.1 Mouvement circulaire à vitesse constante

L'état de base d'un mouvement circulaire à vitesse constante est identique à celui du mouvement rectiligne à vitesse constante, c'est-à-dire :

$$x(k)= \begin{cases} x_1(k) = x_1(k-1) + Tx_2(k-1) \\ x_2(k) = x_2(k-1) \end{cases} \quad \text{(III.19)}$$

L'équation de vecteur des paramètres est

$$\theta(k) = \begin{bmatrix} R_0 \\ X_0 \\ Y_0 \end{bmatrix} \quad \text{(III.20)}$$

Posons
$$\begin{cases} x_3(k) = R_0 \\ x_4(k) = X_0 \\ x_5(k) = Y_0 \end{cases} \quad \text{(III.21)}$$

L'état augmenté sera:

$$x(k) = \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \\ x_5(k) \end{bmatrix} = \begin{bmatrix} x_1(k-1) + Tx_2(k-1) \\ x_2(k-1) \\ x_3(k-1) \\ x_4(k-1) \\ x_5(k-1) \end{bmatrix} + V(k-1) \quad \text{(III.22)}$$

$$x(k) = \begin{bmatrix} 1 & T & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} x(k-1) + V(k-1) \quad \text{(III.23)}$$

$x(k)$ est sous la forme *x(k) = Fx(k-1) + V(k-1)*

Avec *V(k-1)* : bruit d'état.

Et la matrice de transition sera :

$$F = \begin{bmatrix} 1 & T & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Les équations de mesure qui décrivent la position de l'objet poursuivi sont :

$$\begin{cases} X = X_0 + R_0 \cos\left(\frac{s}{R_0}\right) \\ Y = Y_0 - R_0 sin\left(\frac{s}{R_0}\right) \end{cases} \quad \text{(III.24)}$$

Ces équations peuvent être en fonction des variables d'état comme suit :

$$y(k) = \begin{cases} X(k) = x_4(k) + x_3(k) cos\left(\frac{x_1(k)}{x_3(k)}\right) \\ Y(k) = x_5(k) - x_3(k) sin\left(\frac{x_1(k)}{x_3(k)}\right) \end{cases} \quad \text{(III.25)}$$

y(k) a la forme suivante *y(k)=H(x(k)) + w(k).*

Avec *w(k)* : bruit de mesure.

La matrice de mesure linéarisée se déduit par dérivation de *h(x)*.

$$H_x = \begin{bmatrix} \frac{\partial X}{\partial S} & \frac{\partial X}{\partial \dot{S}} & \frac{\partial X}{\partial \mathrm{R}} & \frac{\partial X}{\partial X_0} & \frac{\partial X}{\partial Y_0} \\ \frac{\partial Y}{\partial S} & \frac{\partial Y}{\partial \dot{S}} & \frac{\partial Y}{\partial \mathrm{R}} & \frac{\partial Y}{\partial X_0} & \frac{\partial Y}{\partial Y_0} \end{bmatrix} \quad \text{(III.26)}$$

Nous aurons alors

$$H_x = \begin{bmatrix} -\sin\left(\frac{S}{R}\right) & 0 & cos\left(\frac{S}{R}\right) - \frac{S}{R}\sin\left(\frac{S}{R}\right) & 1 & 0 \\ \cos\left(\frac{S}{R}\right) & 0 & -\sin\left(\frac{S}{R}\right) - \frac{S}{R}\cos\left(\frac{S}{R}\right) & 0 & 1 \end{bmatrix} \quad \text{(III.27)}$$

III.6.2.2 Mouvement circulaire à accélération constante

L'état de base d'un mouvement circulaire à accélération constante est identique de celui du mouvement rectiligne à accélération constante.

$$\begin{cases} s(k) = s(k-1) + Tv(k-1) + \frac{1}{2}T^2 a(k-1) \\ v(k) = v(k-1) + Ta(k-1) \\ a(k) = a(k-1) \end{cases} \quad \text{(III.28)}$$

Le vecteur des paramètres est le même que celui du mouvement circulaire à vitesse constance.

$$\theta(k) = \begin{bmatrix} R_0 \\ X_0 \\ Y_0 \end{bmatrix} \quad \text{(III.29)}$$

Et posons $\begin{cases} x_1(k) = s(k) \\ x_2(k) = v(k) \\ x_3(k) = a(k) \\ x_4(k) = R_0 \\ x_5(k) = X_0 \\ x_6(k) = Y_0 \end{cases}$ (III.30)

L'état augmenté sera :

$$x(k) = \begin{bmatrix} x(k) \\ \theta(k) \end{bmatrix} = \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \\ x_5(k) \\ x_6(k) \end{bmatrix} \quad \text{(III.31)}$$

$$x(k) = \begin{bmatrix} 1 & T & 0.5T^2 & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} x(k-1) + V(k-1) \quad \text{(III.32)}$$

x(k) a la forme suivante *x(k) = F x(k-1)+V(k-1)*

Les équations de mesures sont les mêmes que celles du mouvement circulaire à vitesse constante et la seule différence se trouve dans l'ordre des variables d'état.

$$y(k) = \begin{cases} X(k) = x_5(k) + x_4(k) cos\left(\frac{x_1(k)}{x_4(k)}\right) \\ Y(k) = x_6(k) + x_4(k) sin\left(\frac{x_1(k)}{x_4(k)}\right) \end{cases} \quad \text{(III.33)}$$

$y(k)$ est de la forme suivant : *y(k)=H(x(k)) + w(k)*

Avec *w(k)* : bruit de mesure.

Par conséquent la matrice H_x s'écrit de la façon suivante :

$$H_x = \begin{bmatrix} -\sin\left(\frac{S}{R}\right) & 0 & 0 & cos\left(\frac{S}{R}\right) - \frac{S}{R}\sin\left(\frac{S}{R}\right) & 1 & 0 \\ \cos\left(\frac{S}{R}\right) & 0 & 0 & -\sin\left(\frac{S}{R}\right) - \frac{S}{R}\cos\left(\frac{S}{R}\right) & 0 & 1 \end{bmatrix} \quad \text{(III.34)}$$

$V(k)$ et $w(k)$ Représentent les bruits du système et d'observation respectivement. Les variances des composantes du vecteur d'état qui représentent le mouvement sont prises de manière à satisfaire les conditions sur l'accélération maximale pour le mouvement à vitesse constante et l'incrément sur l'accélération maximale pour le mouvement à accélération constante.

III.7 Applications du suivi

Le suivi d'objets dans une séquence d'images à beaucoup d'applications possibles : en télédétection, le suivi de l'évolution de la couverture au sol permet d'assurer une bonne gestion des ressources et de planifier les activités. Dans le domaine de la sécurité ou de la surveillance, la vidéo est un moyen efficace de détecter des mouvements, de compter et/ou de suivre des personnes, d'identifier des personnes suspectes,… Dans le domaine robotique, les applications sont différentes : la vidéo est utilisée pour l'évitement d'obstacles, ou Lors de la découverte d'un environnement inconnu, pour la construction d'une carte. Dans le domaine automobile, le suivi sert également à l'élaboration d'algorithmes d'évitement d'obstacles, mais aussi à assister le conducteur dans sa conduite (voire rendre la conduite totalement automatique). Le suivi est également utilisé dans le domaine médical pour faire de la télé opération. Enfin, 1a Réalité Virtuelle permet à l'utilisateur d'interagir en temps réel avec sa machine, pour des applications de jeux vidéo bien sûr, mais aussi pour des applications industrielles comme par exemple pour pouvoir examiner un lieu sans pour autant être présent sur place, ou encore pour remplacer à moindre coût de nombreux prototypes physiques (qui peuvent alors être testés immédiatement, modifiés si besoin et retestés indéfiniment).

Selon le type d'application, le degré de précision du suivi est plus ou moins grand, et un type de représentation bien choisi permet souvent d'améliorer la qualité du suivi pour obtenir la précision souhaitée en rapport avec l'application.

III.8 Conclusion

Dans ce chapitre, on a présenté des algorithmes qui concernent le suivi d'objets en mouvement dans une séquence d'image.

La génération des trajectoires de la cible est basée sur les quatre modèles d'état cité ci-dessus, qui sont : mouvement rectiligne à vitesse constante, mouvement rectiligne à accélération constante, mouvement circulaire à vitesse constante et mouvement circulaire à accélération constante.

Chapitre IV :

Le filtrage particulaire

IV.1 Introduction

Dans ce chapitre, nous commençons par présenter deux grandes classes de méthodes: les méthodes analytiques, comme le célèbre filtre de Kalman et les méthodes numériques comme les méthodes de maillage. Nous montrons ainsi les limites et faiblesses de ces méthodes.

On se propose également d'introduire les bases d'une technique d'approximation dite particulaire. Cette technique fait appel à la méthode de Monte Carlo que nous présentons dans un premier temps. Après avoir présenté le formalisme et le principe des méthodes de Monte Carlo, nous abordons les difficultés liées à ce type d'estimateur ainsi que les techniques permettant de pallier ces problèmes. Ensuite, nous décrivons l'algorithme générique de filtrage particulaire qui peut en être déduit.

IV.2 Le problème du filtrage

Le problème du filtrage consiste à déterminer des estimateurs des variables d'un système dynamique (variables d'états) sujet à des perturbations et observées partiellement. La modélisation de l'évolution dans le temps des états $(x_t)t \geq 0$ du système considéré s'écrit alors sous la forme d'une partie d'évolution déterministe et d'une partie stochastique,

$$x_t = x_0 + \int_0^t f(s, x_s, \omega_s)ds \qquad \text{(IV.1)}$$

Où x_0 est de loi donnée, f est une fonction déterministe caractérisant la dynamique et $(w_t)t \geq 0$ est un processus de Wiener standard modélisant les perturbations aléatoires de la dynamique et notre méconnaissance du modèle. Le processus x_t, (IV.1) est un processus Markovien, on désigne par Q_k son noyau de transition. Dans le but de déterminer l'état du système, on est amené à construire une équation d'observation qui relie à des instants k l'observation y_k à l'état courant x_k. Ces mesures sont entachées d'erreurs dues à l'imperfection du capteur de mesure. La suite des observations $(y_k)k \geq 0$ est modélisée par l'équation suivante :

$$y_k = h_k(x_k) + v_k \qquad \text{(IV.2)}$$

Où h_k est la fonction d'observation connue et $(v_k)k \geq 0$ une suite de variables aléatoires (v. a.) de statistique connue modélisant l'imperfection des observations. On cherche à déterminer l'état du système x_k à partir d'observations bruitées $(y_1, \ldots, y_n)$. Suivant la valeur de n, on distingue :

- si $n > k$, il s'agit d'un problème de lissage,
- si $n = k$, il s'agit d'un problème de filtrage,

- si $n < k$, il s'agit d'un problème de prédiction,

Les hypothèses classiques du filtrage non-linéaire sont :

-les observations $\{ y_k, k \geq 1\}$ sont mutuellement indépendantes, conditionnellement à $\{ x_k, k \geq 0 \}$.

-le processus d'état x_k est Markovien et le bruit de mesure v_k est une séquence de variables aléatoires indépendantes identiquement distribuées (iid), indépendant du processus x_k.

- x_0, l'état initial de densité p_0 est supposé de densité connue et indépendant de v_k et de w_k.

On cherche à minimiser la variance de l'erreur de filtrage. Ainsi, si $\hat{x}_k$ désigne un estimateur de l'état x_k, celui-ci sera optimal s'il minimise l'erreur moyenne quadratique :

$$E[\|x_k - \hat{x}_k\|^2]$$

Partant de l'hypothèse que seules informations dont on dispose sur le système sont les mesures $y_{1:k} = y_1, \ldots . y_k$, l'espérance conditionnelle donnée par :

$$\hat{x}_k = \mathbb{E}[x_k | y_1, \ldots, y_k] \quad \text{(IV. 3)}$$

Donne l'estimateur non biaisé de variance minimale. Dans le cas où la loi conditionnelle de x_k sachant les observations $(y_1,\ldots,y_k)$ est multimodale, il peut être souhaitable de prendre comme estimateur le maximum a posteriori,

$$\hat{x}_k = ArgMax_{x_k}[p(x_k | y_1, \ldots, y_k)] \quad \text{(IV.4)}$$

IV.2.1 Le filtrage optimal

Le filtrage optimal consiste à calculer la densité conditionnelle p_k de l'état x_k, sachant les observations $y_{1:k}= (y_1,\ldots,y_k)$ jusqu'à l'instant courant k,

$$p_k(dx) = \mathbb{P}[x_k \in dx | y_{1:k}].$$

La suite des lois de probabilités conditionnelles $(p_k)_{k\geq 0}$, est appelée filtre optimal. En pratique, on souhaite obtenir un algorithme récursif de p_k. Le calcul de p_k ne doit être fonction que de la dernière observation y_k et de la loi conditionnelle précédente p_{k-1}. Le calcul effectif du filtre optimal est un problème trés délicat, par contre son formalisme mathématique est aisé. Le passage entre p_{k-1} à pk fait intervenir la loi conditionnelle de $p_{k|k-1}$ de l'état x_k sachant les observations $y_{1:k-1} = (y_1, \ldots . , y_{k-1})$ jusqu'à l'instant précédent $k - 1$,

$$p_{k|k-1}(dx) = \mathbb{P}[x_k \in dx | y_{1:k-1}]$$

La suite des lois de probabilité $\left(p_{k|k-1}\right)_{k\geq 0}$, est appelée filtre de prédiction. L'évolution du filtre optimal se décompose en deux étapes essentielles.

- **Prédiction :**

L'étape de prédiction du filtre optimal utilise la connaissance a priori du système, à travers le noyau de transition Q_k, en réalisant la transition de p_{k-1} à $p_{k|k-1}$ de la manière suivante,

$$p_{k|k-1}(dx) = \int_{\mathbb{R}^d} p_{k-1}(dx')Q_k(x', dx) \triangleq (Q_k p_{k-1})(dx) \quad \text{(IV.5)}$$

où $\mathbb{R}^d$ désigne l'espace d'état.

- **Correction :**

La deuxième étape utilise l'observation y_k à travers la densité conditionnelle de y_k sachant x_k Cette densité notée $g_k(x) = p(y_k|x_k)$ est appelée vraisemblance, elle permet de corriger la densité prédite $p_{k|k-1}$, par application de la formule de Bayes.

$$p_k(dx) = \frac{g_k(x)p_{k|k-1}(dx)}{\int_{\mathbb{R}^d} g_k(x')p_{k|k-1}(dx')} \quad \text{(IV.6)}$$

L'application de ces étapes (IV.5) et (IV.6) donnent la solution théorique du filtrage non linéaire optimal. Son calcul effectif exige le calcul d'intégrales multidimensionnelles. Malheureusement, dès que l'équation d'état et/ou l'équation d'observation est non-linéaire, ces intégrales ne se calculent pas analytiquement. On a alors recours à des approximations soit par des méthodes analytiques (par exemple le filtre de Kalman étendu qui considère le modèle linéarisé), soit par des techniques numériques (méthodes de maillage).

IV.3 Méthodes de résolution

IV.3.1Méthodes Analytiques

IV.3.1.1Filtre de Kalman-Bucy (KF)

Le filtre de Kalman est un outil essentiel pour les ingénieurs pour résoudre les problèmes de filtrage. Version moderne et probabiliste de la méthode des moindres carrés, le filtre de Kalman concerne essentiellement les systèmes linéaires. Le filtre fut développé par Kalman **[30]** en 1960 pour le cas discret et repris en 1961 par Kalman et Bucy **[31]** pour le cas continu. On considère que le modèle est linéaire, avec un bruit additif Gaussien et l'observation est une fonction linéaire de l'état entachée d'un bruit additif Gaussien,

$$\begin{cases} x_k = F_k x_{k-1} + B_k + W_k \\ y_k = H_k x_k + D_k + V_k \end{cases}$$

Où :

- F_k de dimension $(d \times d), B_k\ (d \times 1)$, $H_k\ (m \times d)$ et $D_k\ (m \times 1)$ sont des matrices déterministes et connues.

- les bruits d'état et d'observation W_k et V_k à valeurs dans $\mathbb{R}^d$ et $\mathbb{R}^m$ sont des bruits blancs Gaussiens de matrices de covariance respectives Q_k et R_k . Ces bruits sont mutuellement indépendants et indépendants de la condition initiale x_0 .

- la loi initiale x_0 est Gaussienne de moyenne $\mathbb{E}[x_0] = \overline{x_0}$ et de matrice de covariance P_0.

Et dans ce cas, on montre que le couple $(x_k, y_k)_{k\geq 1}$ est Gaussien et que les lois conditionnelles p_k sont elles aussi Gaussiennes. Le problème est de dimension finie. Deux paramètres suffisent à caractériser la densité p_k : sa moyenne $\hat{x}_k$ et sa matrice de covariance P_k. Ces paramètres sont calculés de manière récursive à partir des équations (VI.5) et (VI.6) ces paramètres sont calculés de manière récursive suivant les étapes de prédiction et de correction du filtre de Kalman.

- **Equations de prédiction :**

$$\begin{cases} \hat{x}_{k/k-1} = F_k \hat{x}_{k-1} + B_k \\ P_{k/k-1} = F_k P_{k-1} F_k^T + Q_k \end{cases} \quad \text{(IV.7)}$$

- **Equation de correction :**

$$\begin{cases} \hat{x}_k = \hat{x}_{k/k-1} + K_k\left[y_k - \left(H_k \hat{x}_{k/k-1} + D_k\right)\right] \\ P_k = [I - K_k H_k] P_{k/k-1} \\ K_k = P_{k/k-1} H_k^T \left[\left[H_k P_{k/k-1} H_k^T + R_k\right]\right]^{-1} \end{cases} \quad \text{(IV.8)}$$

K_k est le gain de Kalman. On remarque que les matrices de covariance $(P_k)_{k\geq 0}$, $(P_{k|k-1})_{k\geq 0}$ et le gain $(K_k)_{k\geq 0}$ ne dépendent pas des observations $(y_k)_{k\geq 1}$. On peut donc les pré-calculer afin de diminuer la quantité de calculs à effectuer en temps réel. On peut voir le filtre de Kalman de trois manières différentes :

- le filtre de Kalman est solution récursive du maximum de vraisemblance (moindres carrés récursifs).

- le filtre de Kalman résout les équations bayésiennes.

- le filtre de Kalman est la projection du processus d'état sur l'espace des mesures de base orthonormale constituée par innovations [y_k - (H_k |$\hat{x}_{k|k-1}$ + D_k)] démontré par Anderson et Moore **[32]**.

Il est intéressant de noter que, dans le cas de bruits non Gaussiens, et de modèles linéaires, le filtre de Kalman fournit l'estimée linéaire de l'état du système à variance minimale. Dans le cas où les coefficients dépendent des observations passées, le modèle est

dit "conditionnellement Gaussien". Le processus (x_k, y_k) n'est pas un processus Gaussien mais la loi conditionnelle p_k reste Gaussienne et le filtre de Kalman fonctionne toujours **[33]**.

IV.3.1.2 Filtre de Kalman Etendu (EKF)

Cette fois ci, le système dynamique est non linéaire avec des bruits blancs Gaussiens additifs, indépendants entre eux et indépendants de la loi initiale.

$$\begin{cases} x_k = f_k(x_{k-1}) + W_k \\ y_k = h_k(x_k) + V_k \end{cases}$$

Si les fonctions f_k et h_k sont dérivables, on peut linéariser le système autour de l'état prédit $\hat{x}_{k|k-1}$ et autour de l'état courant $\hat{x}_k$.

$f_k(x) \approx F_k x + C_k$, avec $F_k = \nabla f_k(\hat{x}_{k-1})$, et $C_k = f_k(\hat{x}_{k-1}) - F_k \hat{x}_{k-1}, h_k(x) \approx H_k x + D_k$, avec $H_k = \nabla h_k(\hat{x}_k)$, et $D_k = h_k(\hat{x}_{k|k-1}) - H_k \hat{x}_{k|k-1}$.

Le modèle linéarisé est alors le suivant :

$$\begin{cases} x_k = F_k x_{k-1} + C_k + W_k \\ y_k = H_k x_k + D_k + V_k \end{cases}$$

Dans ce cas les matrices de covariance et de gain ne peuvent plus être calculés hors ligne car H_k et F_{k+1} sont fonction de l'état prédit, et de l'état courant (VI.7) et (VI.8).
Simple à mettre en oeuvre, rapide en temps de calcul, le filtre de Kalman est un outil complet, il est utilisé couramment par les ingénieurs dans le domaine du filtrage. Des travaux récents de Picard et Getout-Petit **[34, 35]** montrent que l'utilisation de l'EKF dans le cas d'un modèle état / observation en temps continu est justifiée lorsque le système est observable et lorsque le bruit d'observation et l'erreur initiale sont faibles. Cependant, l'EKF ne peut pas s'appliquer lorsque la densité conditionnelle est multimodale.

IV.3.1.3 Filtre de Kalman sans biais (UKF)

Le filtre de Kalman sans biais, appelé dans la littérature « Unscented Kalman Filter (UKF) » de Julier et Uhlmann **[36]** est une autre extension du filtre de Kalman aux systèmes non-linéaires.

Comme pour l'EKF la densité conditionnelle est supposée être une gaussienne qui sera maintenant caractérisée par un ensemble de $(2d + 1)$ points soigneusement choisis (sigma points). Ils sont alors propagés grâce à la vraie équation dynamique non-linéaire du système. Ils permettent d'évaluer précisément la moyenne et la matrice de covariance de l'état prédit (avec une précision d'ordre 3 de Taylor). Contrairement à l'EKF, l'algorithme de l'UKF

n'utilise pas explicitement le calcul du gradient des équations du système. Les approximations sont basées sur les interpolations des équations du système avec les sigma-points.

L'algorithme de l'UKF est le suivant :

- **Initialisation** : la première étape de l'UKF consiste à générer un nuage de points répartis dans un volume paramétré par $\overline{x_0}$ et P_0 et qui constitue une approximation discrète de la loi $N(\overline{x_0},\ P_0)$.
- **Prédiction** : à l'instant $t = k - 1$, chaque point du nuage est prédit selon la loi d'évolution non linéaire du système. Les estimées prédites $\hat{x}_{1:k|0:k-1}$ et $P_{1:k|0:k-1}$ sont alors calculées en fonction de l'échantillon de la loi $P(x_{1:k}|y_{0:k-1})$ que constituent les points du nuage prédit. A chaque point du nuage est de plus associée une mesure prédite dont on calcule également la moyenne. $\hat{y}_{k|k-1}$.
- **Mise à jour (correction) :** les estimées prédites $\hat{x}_{1:k|0:k-1}$ et $P_{1:k|0:k-1}$ sont mise a joins proportionnellement à l'innovation $y_k - \hat{y}_{k|k-1}$ et à la matrice de covariance des mesures prédites.

Remarque :

L'UKF est plus robuste aux non-linéarités que l'EKF, et il ne demande aucune linéarisation de modèle, ce qui ne demande aucun calcul explicite de gradient. De plus, le coût de calcul de l'UKF est quasiment identique à celui de l'EKF. Dans les applications où la dynamique est fortement non-linéaire, ce filtre donne de meilleurs résultats que l'EKF.

Les filtres analytiques présentés ci-dessus ne sont pas adaptés lorsque la densité conditionnelle est multimodale.

IV.3.2 Méthodes Numériques

Considérons un système dynamique à temps continu d'état $x_t \in \mathbb{R}^d$ et observé par la variable $y_t \in \mathbb{R}^m$ dont les équations d'évolution sont définies par :

$$\begin{cases} dx_t = a(x_t)dt + \sigma(x_t)dW_t \\ dy_t = h(x_t)dt + dV_t \end{cases} \quad \text{(IV.9)}$$

Où dW_t et dV_t sont deux processus de Wiener indépendants. Parmi les approches utilisées pour traiter le problème du filtrage non linéaire, l'approche numérique consiste à voir le filtre optimal p_t comme la solution d'une équation aux dérivées partielles stochastique, l'équation de Zakai (sous forme continue) **[37]**.

$$dp_t = L^T p_t dt + p_t h^T R^{-1} dy_t \quad \text{(IV. 10)}$$

avec L est un opérateur associé au processus x_t, t ≥ 0

$$L = \frac{1}{2}\sum_{i,j=1}^{m} a_{i,j}(.)\frac{\partial^2}{\partial x_i \partial x_j} + \sum_{i,j=1}^{m} f_i(.)\frac{\partial}{\partial x_i} \quad \text{(IV.11)}$$

avec $a = (a_{i,j}) = \sigma\sigma^T$.

On discrétise l'équation de Zakai par des méthodes de maillage de l'espace d'état (schéma d'Euler). Il existe deux types de méthodes de maillage : méthode de maillage à pas fixe qui a l'inconvénient de mailler des zones où la densité de probabilité prend des valeurs négligeables. Pour cette méthode, la taille de la maille sera prise identique pour des zones de forte probabilité. Les méthodes de maillage à pas variable développées par Cai, LeGland et Zhang **[38]**, sont plus intéressantes car elles permettent de raffiner le pas de maillage dans les zones de forte probabilité. Pour de faibles dimensions de l'espace d'état ($d \leq 3$), ces méthodes donnent de bons résultats pour un temps de calcul acceptable. Mais au-delà de la dimension 3 ces méthodes sont inutilisables car, pour une précision donnée, le temps de calcul croît exponentiellement avec d.

En conclusion : le filtre de Kalman et ses extensions sont performants si les hypothèses sur les densités mises en jeu sont vérifiées en pratique. Dans des cas plus généraux, les filtres par mélange de lois gaussiennes ou les méthodes numériques par maillage d'état sont applicables, mais sont difficilement utilisables à cause de leur coût de calcul élevé et de la difficulté à les mettre en œuvre. Plus récemment, de nouvelles méthodes ont été développées pour résoudre le problème du filtrage dans un cas général. Nous allons présenter ces méthodes dans la section suivante.

IV.4 Méthodes de Monte Carlo pour le filtrage non linéaire

Lorsque les modèles d'état ne sont plus linéaires et que les modèles de bruit ne sont plus gaussiens, ce qui est souvent le cas en réalité, les calculs sont beaucoup plus complexes, et dès que la dimension de l'espace d'état est supérieure à quatre, les approximations ne peuvent plus être calculées.

Les méthodes probabilistes fournissent alors un cadre permettant d'estimer la loi a posteriori de manière simple. Les méthodes séquentielles de Monte Carlo proposent une

approximation de cette loi par une somme de lois de Dirac centrées en des points appelés particules, et pondérées par la vraisemblance des mesures conditionnellement à ces points. Ces méthodes reposent sur le principe de Monte Carlo qui consiste à, simuler un grand nombre de réalisations d'une variable aléatoire pour en approcher sa loi. Le concept est apparu au milieu du vingtième siècle mais n'a été mis en œuvre que beaucoup plus tard dans le cadre du problème de l'estimation séquentielle, car il a fallu attendre que la puissance de calcul des ordinateurs soit suffisante.

IV.4.1 Echantillonnage Monte Carlo standard

Soit X une variable aléatoire dans $\mathbb{R}^d$ distribuée selon la loi de densité de probabilité p et $\left(X^i\right)_{i=1,\ldots..,N}$ des variables aléatoires indépendantes sur $\mathbb{R}^d$ de même loi que X (iid). Pour toute fonction φ bornée de $\mathbb{R}^d \to \mathbb{R}$; L'espérance de φ (X) vaut

$$\mathbb{E}[\varphi(X)] = \int \varphi(x)p(x)dx \tag{IV.12}$$

La loi des grands nombres assure que, presque sûrement (p.s), la moyenne empirique tend vers l'espérance.

$$\bar{\varphi}^N(X) = \frac{1}{N}\sum_{i=1}^{N} \varphi\left(X^i\right) \xrightarrow[N\to+\infty]{} \mathbb{E}[\varphi(X)] \tag{IV.13}$$

La variance de $\bar{\varphi}^N$vaut,

$$var\left(\bar{\varphi}^N(X)\right) = \frac{\sigma^2}{N} \tag{IV.14}$$

avec

$$\sigma^2 = \int(\varphi(x) - \mathbb{E}[\varphi(X)])^2 p(x)dx$$

Le théorème central limite nous donne la loi de l'erreur de $(\varphi(X) - E[\varphi(X)])$ quand N tend vers l'infini.

$$\frac{\sqrt{N}}{\sigma}(\varphi(x) - \mathbb{E}[\varphi(X)]) \xrightarrow[N\to+\infty]{} N(0,1) \tag{IV.15}$$

Avec $N(0,1)$ est la loi normale centrée réduite. Ceci montre que l'erreur moyenne d'une quadrature Monte Carlo est de l'ordre de $\sigma/\sqrt{N}$. Ainsi l'erreur de l'approximation Monte Carlo est peu dépendante de la dimension de l'état. Un autre avantage de cette méthode est qu'elle ne dépend pas de la régularité de la fonction φ à intégrer, pourvu que φ soit de carré intégrable. Ces méthodes sont basées sur la simulation de variables aléatoires pour calculer

approximativement des intégrales de grande dimension. Elles tirent leur justification de la loi des grands nombres qui permet d'approcher une mesure de probabilité par la mesure empirique calculée à l'aide d'un échantillon.

IV.4.1.1 Echantillonnage Monte Carlo pondéré

L'échantillonnage Monte Carlo pondéré peut être vu comme une généralisation de la méthode d'échantillonnage Monte Carlo standard. Considérons p une loi de densité difficile à simuler. Supposons que l'on sache simuler selon la loi de densité q, et que la densité p puisse se mettre sous la forme :

$$p(x) = c\, g(x) q(x)$$

Où c est une constante de normalisation, et où g est une fonction positive bornée que l'on sait évaluer en tout point. Soit $(x^1, \ldots, x^N)$, un échantillon issu de la densité q. Par la loi des grands nombres, on approche p par:

$$p(x) \approx \frac{c}{N} \sum_{i=1}^{N} g(x^i)\, \delta(x = x^i) \qquad \text{(IV.16)}$$

q est appelé la probabilité d'importance en anglais proposal density. La constante c se calcule elle aussi par approximation Monte Carlo

$$c = \frac{1}{< g, q >} \approx \frac{1}{\frac{1}{N} \sum_{i=1}^{N} g(x^i)}$$

Finalement, la densité approchée de p vaut :

$$p = \sum_{i=1}^{N} \omega^i \delta(x^i), \qquad \omega^i = \frac{g(x^i)}{\sum_{j=1}^{N} g(x^j)} \qquad \text{(IV.17)}$$

Les pondérations ω^i sont appelés les poids.

IV.4.1.2 Génération de variables aléatoires par la méthode d'acceptation/rejet

Cette méthode consiste à générer un échantillon de $v.a.$ qui suit exactement la loi p en acceptant ou rejetant les $v.a.$ générées suivant q. Posons $\gamma = sup_x g(x) < \infty$. On applique l'algorithme suivant :

(1) Générer $x^i \sim q$ et u uniforme dans [0; 1].

(2) Si $u \leq \frac{g(x^i)}{\gamma}$ (test de rejection) alors on retient la variable aléatoire $x^j = x^i$, sinon retourner en (1).

En filtrage particulaire, cette méthode a l'inconvénient d'être très coûteuse en temps de calcul, puisqu'on risque de rejeter beaucoup de variables aléatoires. En effet, la probabilité d'acceptation vaut :

$$P_a = \frac{<g,q>}{\gamma} \quad \text{(IV.18)}$$

Si il y a peu de cohérence entre g et q($< g; q >$ petit), P_a sera faible. En revanche, l'échantillon suit exactement la loi p.

IV.4.2 Filtrage particulaire

Considérons un système Markovien discret non linéaire suivant. Par souci de simplicité, les bruits sont pris additifs :

$$\begin{cases} x_k = F_k(x_{k-1}) + W_k \\ y_k = H_k(x_k) + V_k \end{cases}$$

Où x_k est le vecteur d'état du système de dimension d, y_k est le vecteur d'observation de dimension m, F_k et H_k sont deux fonctions continues différentiables de $\mathbb{R}^d$ vers $\mathbb{R}^d$ et de $\mathbb{R}^d$ vers $\mathbb{R}^m$ respectivement. $W_k$et V_k sont des vecteurs de bruits (pas nécessairement gaussiens). Le Filtrage Particulaire appelé aussi « bootstrap filter » ou « Monte Carlo filter » est une méthode numérique permettant d'approcher la distribution de probabilité conditionnelle de l'état sachant les observations, au moyen de la distribution empirique d'un système de particules. Les particules se déplacent selon des réalisations indépendantes à partir de l'équation d'état. Elles sont corrigées ou pondérées en fonction de leur cohérence avec les observations (quantifiée par la fonction de vraisemblance) (IV.13). Les conditions de base nécessaires à l'implémentation de tout algorithme particulaire sont :

- savoir générer suivant la loi de transition de l'état $p(x_k|x_{k-1})$.
- savoir évaluer la fonction de vraisemblance $p(y_k|x_k)$ en tout point de l'espace d'état.

IV.4.2.1 L'algorithme du Filtrage Particulaire

On peut présenter le filtrage particulaire classique de 2 façons différentes.

-Première approche du filtrage particulaire : La première façon est basée sur l'approximation d'une densité par une combinaison convexe de Dirac (voir figure (IV.1)). On suppose qu'à l'instant $k-1$, on a une approximation de la distribution conditionnelle $p(x_{k-1}|y_{1:k-1})$ de l'état x_{k-1} sachant les mesures $y_{1:k-1} = y_1, \dots\dots, y_{k-1}$ de la forme

$$p(x_{k-1}|y_{1:k-1}) \approx \sum_{i=1}^{N} \omega_{k-1}^{i} \delta\left(x_{k-1} = x_{k-1}^{i}\right)$$

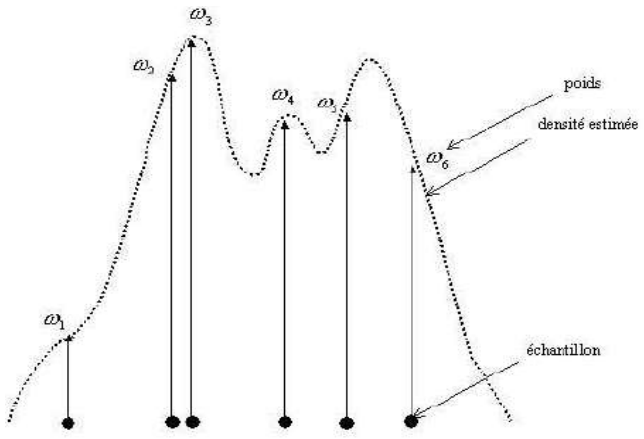


Figure. IV.1 Mesure discrète pondérée.

Les ω_{k-1}^{i} sont des poids positifs de somme 1, $x_{k-1}^{1},\ldots, x_{k-1}^{N}$ sont des vecteurs de $\mathbb{R}^{d}$ appelé « particules ».

-L'étape de prédiction (IV.2.1) : Considère $p(x_k, x_{k-1}|y_{1:k-1})$ comme la densité conjointe (conditionnelle) alors on a

$$p(x_k|y_{1:k-1}) = \int p(x_k, x_{k-1}|y_{1:k-1}) dx_{k-1} \quad \text{(IV.19)}$$

Par définition de la probabilité conditionnelle, on a

$$p(x_k, x_{k-1}|y_{1:k-1}) = p(x_k|x_{k-1},\, y_{1:k-1}) p(x_{k-1}|y_{1:k-1}) \quad \text{(IV.20)}$$

Par hypothèse, on a

$$p(x_k|x_{k-1},\, y_{1:k-1}) = p(x_k|x_{k-1})$$

D'où

$$p(x_k|y_{1:k-1}) = \int p(x_k|x_{k-1}) p(x_{k-1}|y_{1:k-1}) dx_{k-1} \quad \text{(IV.21)}$$

Si on approche $p(x_{k-1}|y_{1:k-1}) \approx \sum_{i=1}^{N} \omega_{k-1}^{i} \delta\left(x_{k-1} = x_{k-1}^{i}\right)$, on a alors

$$p(x_k|y_{1:k-1}) \approx \sum_{i=1}^{N} \omega_{k-1}^{i} p\left(x_k|x_{k-1}^{i}\right) \quad \text{(IV.22)}$$

La loi de densité prédite $p(x_k|y_{1:k-1})$ s'approche par

$$p(x_k|y_{1:k-1}) = \sum_{i=1}^{N} \omega_{k|k-1}^{i} \delta\left(x_k = x_{k|k-1}^{i}\right) \quad \text{(IV.23)}$$

Avec $x_{k|k-1}^{i}$ qui sont obtenues par des réalisations indépendantes de la loi de transition $p(x_k|x_{k|k-1}^{i})$ et les poids $\omega_{k|k-1}^{i}$ ne changent pas $\omega_{k|k-1}^{i} = \omega_{k-1}^{i}$.

-L'étape de correction (IV.2.1) : à l'étape de correction, on passe de la loi de densité prédite $p(x_k|y_{1:k-1})$ à la loi de densité conditionnelle $p(x_k|y_{1:k})$ grâce à la vraisemblance $g(y_k - H_k(x_{k|k-1}^{i}))$ donnée par le modèle du bruit de mesure. La loi de densité conditionnelle est alors approchée par une autre distribution de Dirac de support $x_k^i = x_{k|k-1}^{i}$

$$p(x_k|y_{1:k}) \approx \sum_{i=1}^{N} \omega_k^i \, \delta\left(x_k = x_k^i\right) \quad \text{(IV.24)}$$

et de poids

$$\omega_k^i = \frac{\omega_{k\backslash k-1}^{i} g\left(y_k - H_k\left(x_{k|k-1}^{i}\right)\right)}{\sum_{j=1}^{N} \omega_{k\backslash k-1}^{j} g\left(y_k - H_k\left(x_{k|k-1}^{j}\right)\right)} \quad \text{(IV.25)}$$

- Seconde approche du filtrage particulaire : La seconde façon de présenter le filtrage particulaire est basée sur l'échantillonnage pondéré séquentiel (IV.4.1.1). On notera $x_{0:k} = (x_0, \ldots, x_k)^T$, la trajectoire de l'état de l'instant initial x_0 à l'instant final x_k. Soit à calculer

$$\mathbb{E}[f(x_{0:k}|y_{1:k})] = \int f(x_{0:k}) \, p(x_{0:k}|y_{1:k}) dx_{0:k}$$

$$= \int f(x_{0:k}) \frac{p(x_{0:k}|y_{1:k})}{q(x_{0:k}|y_{1:k})} q(x_{0:k}|y_{1:k}) dx_{0:k}$$

D'après la méthode d'échantillonnage pondéré séquentielle

$$\omega_k = \frac{p(x_{0:k}|y_{1:k})}{q(x_{0:k}|y_{1:k})} \quad \text{(IV.26)}$$

On a alors

$$\mathbb{E}[f(x_{0:k}|y_{1:k})] \approx \sum_{i=1}^{N} f\left(x_{0:k}^{i}\right) \omega\left(x_{0:k}^{i}\right) / \sum_{j=1}^{N} \omega\left(x_{0:k}^{i}\right) \quad \text{(IV.27)}$$

On a

$$p(x_{0:k}|y_{1:k}) = \frac{p(x_{0:k-1}|y_{1:k-1}) p(x_k|x_{0:k-1}, y_{1:k-1}) p(y_k|x_k)}{p(y_k|y_{1:k-1})}$$

vu le caractère Markovien de x_k, on peut écrire $p(x_k|x_{0:k-1}, y_{1:k-1}) p(y_k|x_k) = p(x_k|x_{k-1})$, d'où

$$p(x_{0:k}|y_{1:k}) = \frac{p(x_{0:k-1}|y_{1:k-1})p(x_k|x_{k-1})p(y_k|x_k)}{p(y_k|y_{1:k-1})} \quad \text{(IV.28)}$$

Pour obtenir un algorithme récursif, la fonction d'importance q doit vérifier la relation suivante

$$q(x_{0:k}|y_{1:k}) = q(x_{0:k-1}|y_{1:k-1})q(x_k|x_{0:k-1}, y_{1:k}) \quad \text{(IV.29)}$$

Si on remplace la relation (IV.26), par (IV.28) et (IV.29), on a alors la relation récurrente suivante

$$\omega_k \propto \omega_{k-1} \frac{p(x_k|x_{0:k-1})p(y_k|x_k)}{q(x_k|x_{0:k-1},y_{1:k})} \quad \text{(IV.30)}$$

Qui réactualise les poids ω_k en fonction de $q(x_k|x_{0:k-1}, y_{1:k})$. Si g désigne la densité de la loi du bruit de mesure, alors $p(y_k|x_k) = g(y_k - h_k(x_k|x_{k-1}))$, et on a

$$\omega_k^i \propto \omega_{k-1}^i \frac{g(y_k - H_k(x_{k|k-1}^i))p\left(x_k^i|x_{k-1}^i\right)}{q(x_k^i|x_{1:k-1}^i,y_{1:k})} \quad \text{(IV.31)}$$

L'estimée de l'état $\hat{x}_k$ est alors égale à

$$\hat{x}_k \approx \sum_{i=1}^N \omega_k^i x_k^i \quad \text{(IV.32)}$$

et la matrice de covariance est estimée par

$$P_k \approx \sum_{i=1}^N \omega_k^i \left(x_k^i - \hat{x}_k\right)\left(x_k^i - \hat{x}_k\right)^T \quad \text{(IV.33)}$$

IV.4.2.2 Choix de la densité d'importance

Le choix de la loi de densité d'importance $q(x_k|x_{0:k-1}, y_{1:k})$ conditionne le bon fonctionnement de l'algorithme particulaire Doucet **[39]**. En effet, la méthode d'échantillonnage pondéré repose sur la génération d'échantillons de particules à partir de la densité d'importance, qui va permettre d'évaluer les poids d'importance. Elle doit être choisie de préférence à minimiser la variance de ces poids et doit satisfaire les conditions suivantes :

- le support de la densité d'importance doit recouvrir le support de la loi de densité conditionnelle.
- la densité d'importance doit tenir compte de l'observation récente.

Pour des raisons de coût de calcul, il n'est pas possible en général de construire une telle fonction d'importance. Cependant, on peut prendre la densité d'importance $q(x_k|x_{0:k-1}, y_{1:k})$ égale à une loi de densité Gaussienne $\emptyset(\hat{x}_k|P_k)$, tel que la moyenne $\hat{x}_k$ et la matrice de

covariance P_k sont calculées par les équations de Kalman en linéarisant les équations d'état et de mesure autour de $x_{k|k-1}$et de x_{k-1} respectivement. Un autre choix très courant est de prendre la densité a priori $q(x_k|x_{0:k-1}, y_{1:k}) = p(x_k|x_{k-1})$ la récente observation n'est pas prise en compte équation (IV.25). Ce choix facilite l'implémentation de l'algorithme. Malheureusement, la fonction d'importance optimale dépend de la densité recherchée :

Supposons que l'on génère un échantillon $(x_1,\ldots,x_N)$ qui suit une loi $q(.)$ quelconque ne s'annulant pas sur le support de p. On a

$$\mathbb{E}_p\big(f(X)\big) = \int f(x)p(x)dx = \int f(x)\frac{p(x)}{q(x)}q(x)dx = \mathbb{E}_q\left[f(x)\frac{p(x)}{q(x)}\right] \qquad \text{(IV.34)}$$

Où f est une fonction à valeurs dans $\mathbb{R}$.

On associe à chaque particule x^ile poids ω^i (ici les poids ω^i sont aléatoires et non normalisés) défini par,

$$\omega^i = \frac{1}{N}\frac{p(x^i)}{q(x^i)}$$

L'estimateur de μ,

$$\bar{\mu}^N \simeq \frac{1}{N}\sum_{i=1}^{N}\frac{p(x^i)}{q(x^i)}f(x^i) = \sum_{i=1}^{N}\omega^i f(x^i) \xrightarrow[N\to+\infty]{} \mathbb{E}_q\left[f(X)\frac{p(X)}{q(X)}\right] = \mu \qquad \text{(IV.35)}$$

A pour variance,

$$var_q\{\bar{\mu}^N\} = \frac{1}{N}\left[\int f^2(x)\frac{p^2(x)}{q^2(x)}dx - \left(\int f(x)p(x)dx\right)^2\right] = \frac{1}{N}\left[\int f^2(x)\frac{p^2(x)}{q^2(x)}dx - \mu^2\right] \qquad \text{(IV.36)}$$

On montre que cette variance est minimale pour

$$q(x) = \frac{|f(x)|p(x)}{\int |f(x)|p(x)dx} \qquad \text{(IV.37)}$$

Cette variance est nulle si $f(.)$ est positive.

IV.4.2.3 Dégénérescence du filtre particulaire

D'après ((IV.26), on a l'augmentation de la dispersion des poids a des effets néfastes sur la qualité de la précision de l'estimée **[39]**, et provoque à long terme la divergence du filtre.

En pratique, si q n'est pas assez proche de p, les poids auront tendance à prendre des valeurs disparates. En effet les poids normalisés décroissent vers 0 pour toutes les trajectoires simulées sauf une qui prend tout le poids et ceci à cause du caractère multiplicatif de la formule récursive des poids ((IV.25). Ce phénomène est appelé « dégénérescence des

pondérations » et provoque la divergence du filtre. Afin de remédier à ce problème, plusieurs méthodes ont été proposées dans le but de régulariser les poids. La première méthode donne naissance au filtre de Monte Carlo à facteur d'oubli qui propose l'introduction d'un facteur d'oubli au niveau des poids. Ceci permet de diminuer l'influence des mesures passées **[40]**, et donc de diminuer la variance des poids. En effet, le filtre à oubli pondère plus fortement les observations récentes. Les poids sont mis à jour de la manière suivante,

$$\omega_k^i \propto p\left(y_k \middle| x_k^i\right)\left(\omega_{k|k-1}^i\right)^{\alpha}, i = 1, \dots, N$$

Où α est un coefficient compris entre 0 et 1. Les mesures éloignées dans le passé $(k - m)$ contribuent peu à l'estimation courante, les poids $(\omega_{k-m}^i)^{\alpha^m}$ tendant vers 1 quand m est grand. Cette méthode permet d'éviter de cumuler les erreurs depuis le premier instant de mesure. Cependant, cette technique empêche de tirer parti pleinement de toutes les observations. Une autre approche pour éviter la dégénérescence des poids est la méthode de ré-échantillonnage.

Cette méthode consiste à favoriser les particules de forte vraisemblance et à éliminer les autres.

IV.4.2.4 Ré-échantillonnage des particules

Comme on l'a vu dans le paragraphe (VI.4.2.3), si le système de particules pondérées $[(x^1, \omega^1), \dots\dots, (x^N; \omega^N)]$ est tel que les poids ont une grande variance, on va générer un autre système de particules de même taille en favorisant les particules de grands poids. Celles-ci seront dupliquées et les autres éliminées. On obtient alors un nouvel échantillon de particules de même poids. Nous allons décrire brièvement, les principales méthodes de sélection des particules utilisées dans les filtres « bootstrap » comme le résumé Doucet **[41]**.

- Ré-échantillonnage Multinomial :

Cette technique est la plus utilisée, chaque particule est sélectionnée en fonction de l'importance de son poids $\widetilde{\omega}_k^i$.

La première étape de la procédure consiste à construire la fonction de répartition des particules du nuage. En abscisse, figure l'index de la particule et, en ordonnée, son poids cumulé à celui de toutes les particules précédentes, voir figure IV.2.

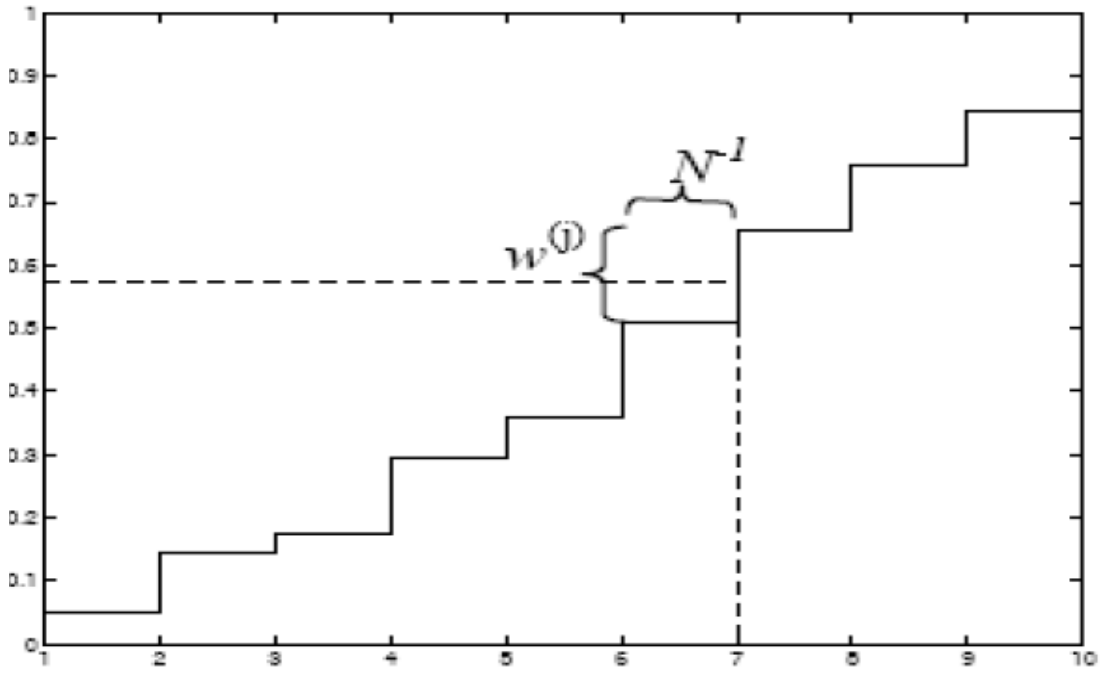


Figure.IV.2 Ré-échantillonnage Multinomial.

L'étape suivante procède au tirage successif de l'index de N particules. Pour ce faire, on tire N nombres appartenant à l'intervalle [0,1] et distribués selon une loi uniforme. Supposons que la situation représentée sur la figure IV.2 correspond au ré-échantillonnage de la particule d'index i. En effectuant un tirage aléatoire, selon la loi uniforme sur [0,1], le nombre extrait est environ 0.57, ce dernier est ensuite projeté sur la fonction de répartition, l'intersection donne l'index j (ici 7) d'une particule qui est conservée comme nouvel échantillon d'indice i. Son poids est réinitialisé à la valeur $1/N$. La probabilité de tirer une particule quelconque à chacun des tirages correspond à son poids normalisé $\tilde{\omega}_k^j$. Il apparaît clairement que l'effectif de la descendance d'une particule donnée (après ré-échantillonnage) est proportionnel au poids normalisé de la particule. Globalement les effectifs N_k^i des particules, à l'issue de ré-échantillonnage, correspondent à une réalisation de la loi multinomiale des paramètres N et $\tilde{\omega}_k^i$ avec $i = L.\ldots.N$. La variance de l'effectif de la descendance d'une particule i est donnée par la relation classique de la variance d'une loi binomiale de paramètres N et $\tilde{\omega}_k^i$.

$$Var\left(N_k^i\right) = N\tilde{\omega}_k^i(1 - \tilde{\omega}_k^i) \tag{IV.38}$$

Cependant, il est possible de réduire la variance par d'autres schémas de sélection.

-Le ré-échantillonnage résiduel :

Cette technique est une amélioration du tirage multinomial, dans le but de diminuer la variance $Var(N_k^i)$ des particules dupliquées. On crée $\lfloor N\omega^i \rfloor$copies de la particule x^i où $\lfloor . \rfloor$ dénote la partie entière. Pour garder un nombre constant de particules, il reste à choisir $\bar{N}$ Particules, Dont :

$$\bar{N} = \sum_{i=1}^{N} (N\omega^i - \lfloor N\omega^i \rfloor) \quad \text{(IV.39)}$$

Pour ce faire on peut utiliser l'échantillonnage résiduel qui consiste à générer un $\bar{N}$ échantillon parmi (x^1,……, x^N), (ré-échantillonnage multinomial) pondérées maintenant par les poids résiduels $\omega'^i = (N\omega^i - \lfloor N\omega^i \rfloor)/\bar{N}$.

-Le ré-échantillonnage déterministe :

Cette méthode a été proposée par Kitagawa **[42]**, on ne génère pas des variables uniformes ordonnées comme les méthodes précédentes mais plutôt un échantillon de variable déterministe $u'_i = (i-m)/N$, avec $m \in [0,1]$. Nous avons alors $\mathbb{E}[N_k^i] \neq N\omega^i$ en revanche la variance $var(N_k^i)$ est très petite.

IV.4.2.5 L'indicateur de ré-échantillonnage

Le ré-échantillonnage n'est pas effectué à chaque cycle de calcul, car il entraine des erreurs de Monte Carlo. On choisit le moment propice pour le ré-échantillonnage des particules, c'est-à dire avant la dispersion des poids. Plusieurs méthodes ont été proposées dans la littérature, on citera 2 critères parmi les critères les plus utilisés. Le premier critère est de Kong et Liu **[43]**, il est basé sur la variance des pondérations. Il introduit le paramètre

$$N_{eff} = \frac{1}{\sum_{i=1}^{N} \omega_k^{i2}} \quad \text{(IV.40)}$$

On montre que la variance de l'approximation S.I.R est proportionnelle à la variance des poids **[43]**. L'utilisateur se fixe un seuil N_{Th} suivant l'application. Le ré-échantillonnage est exécuté si $N_{eff} < N_{Th}$. Le second critère de ré-échantillonnage a été proposé par Pham **[44]**. Le critère est basé sur le calcul de l'entropie des pondérations

$$Ent = \log(N) + \sum_{i=1}^{N} \omega_k^i \log \omega_k^i \quad \text{(IV.41)}$$

On a toujours

$$0 < Ent < \log(N)$$

Le critère est nul si tous les poids sont égaux à $1/N$ et vaut $\log(N)$ si l'un des poids vaut 1. Le ré-échantillonnage n'est enclenché que si $Ent > Th$, avec Th fixé par l'utilisateur

en fonction de l'application. En particulier, il dépend de l'amplitude de la variation du signal de mesure en fonction de l'état. Les filtres particulaires utilisant le ré-échantillonnage s'appellent les filtres S.I.R (Sampling Importance Resampling) **[45]**.

Nous présentons ci-dessous l'algorithme du filtrage particulaire S.I.R ou la fonction d'importance est égale à la densité prédite.

IV.4.2.6 Résumé de l'algorithme du filtrage particulaire (S.I.R)

(1) l'initialisation :

On tire les particules $x_{0|0}^1, \ldots\ldots, x_{0|0}^N$ selon une loi Gaussienne connue a priori de moyenne le vecteur d'état initial et de matrice de covariance $P_{0|0}$.

(2) l'étape de prédiction :

Chaque particule x_{k-1}^i est propagée selon l'équation d'évolution du système :

$x_{k|k-1}^i = F_k(x_{k-1}^i) + W_k^i$, ou les W_k^i, sont un échantillon (iid) du bruit de dynamique.

(3) l'étape de correction :

À l'instant k, les poids de chaque particule $\omega_{k|k-1}^i$ sont réactualisés en ω_k^i via la formule :

$$\omega_k^i = \frac{\omega_{k|k-1}^i g\left(y_k - H_k\left(x_{k|k-1}^i\right)\right)}{\sum_{j=1}^N \omega_{k|k-1}^j g\left(y_k - H_k\left(x_{k|k-1}^j\right)\right)} \qquad \text{(IV.42)}$$

(4) l'étape du ré-échantillonnage :

Si le test du ré-échantillonnage est positif, on fait un tirage multinomial des particules $x_{k|k-1}^i, \ldots\ldots, x_{k|k-1}^N$ selon leur poids $\omega_k^1, \ldots, \omega_k^N$. Les poids ω_k^i sont alors réinitialisés à $\frac{1}{N}$.

L'estimée de l'état est donnée par $x_k^i \approx \sum_{i=1}^N \omega_k^i x_k^i$ qui est la moyenne du nuage de particule. La matrice de covariance de l'état est estimée $P_k \approx \sum_{i=1}^N \omega_k^i (x_k^i - \hat{x}_k)\,(x_k^i - \hat{x}_k)^T$.

Cependant, le filtre particulaire présente des comportements instables dus particulièrement aux cas de bruit de dynamique ou de mesure faible. En effet, après chaque ré-échantillonnage, des particules peuvent occuper la même position, si le bruit de dynamique est faible. Les particules n'occupent alors qu'une petite zone restreinte de l'espace d'état. Ainsi, les particules perdent leur capacité d'exploration de l'espace d'état, ceci provoque la divergence du filtre.

Dans le but de robustifier le filtre particulaire, un nouveau filtre particulaire a été proposé, appelé filtre particulaire régularisé (RPF).

IV.5 Conclusion

Dans ce chapitre, nous avons décrit les deux principales méthodes utilisées dans le filtrage qui sont les méthodes analytiques et numériques. Dans les méthodes analytiques, le filtre de Kalman étendu est le filtre par excellence des ingénieurs, il donne de très bons résultats dans le cadre des systèmes faiblement non-linéaires. Mais lorsque les systèmes sont fortement non linéaires, les performances du filtre de Kalman étendu sont mauvaises. Quant aux méthodes numériques, elles fournissent des résultats d'une bonne précision au problème du filtrage non linéaire. Cependant, elles restent limitées à des espaces de dimension au plus égale à 3. Au-delà de cette dimension, le temps de calcul augmente d'une façon exponentielle avec la dimension de l'espace, ce qui les rend impraticables en temps réel. C'est pour cela qu'une nouvelle catégorie de méthodes a été introduite : les méthodes de Monte Carlo. Elles sont mieux adaptées au problème du filtrage non linéaire car elles sont peu sensibles à la dimension de l'espace d'état. Nous appliquerons ce type de méthodes, appelées filtrage particulaire au problème du recalage de navigation inertielle.

De plus, nous avons présenté les méthodes particulaires classiques. Dans le cas général, les méthodes particulaires donnent une bonne approximation du filtre optimal. Cependant, dans le cas de bruits de dynamique faible, le filtre particulaire classique peut diverger. Dans le but d'assurer la diversité du système de particules au cours du temps, une étape de régularisation a été introduite, ce qui robustifie le filtre particulaire. Ces filtres particulaires sont appelés filtres particulaires régularisés (RPF) et sont basés sur la théorie de l'estimation par noyaux.

Chapitre V:

Les Résultats de Simulation

V.1 Introduction

Le suivi d'objet est utilisé pour décrire le processus d'enregistrement de mouvement et de traduire ce mouvement sur un modèle numérique. Simulink avec la vidéo et l'image blockset permettent d'effectuer des simulations rapides en temps réel de la vidéo intégrée, la vision et systèmes d'imagerie **[46]**. Il peut créer des spécifications exécutables pour communiquer le système aux équipes de conception en aval et de fournir une référence pour une vérification complète tout au long du processus de conception. La quantité de travail ne varie pas en fonction de la complexité ou de la performance avec le même degré que par l'utilisation de techniques traditionnelles. L'objectif de ce chapitre est de mettre en œuvre une méthodologie efficace pour suivre l'objet en mouvement présent à l'intérieur des vidéos mobiles. Utilisant le blockset, le système est réalisé en utilisant les étapes de traitement d'image telles que le traitement vidéo, l'affichage de l'image, la soustraction du fond, détection de bord, segmentation et le suivi **[47]**. Tout d'abord, les vidéos sont séparées sous forme de trames et le procédé de prétraitement est utilisé pour la conversion de couleurs pour soustraire les objets de premier plan à partir de l'arrière-plan, et la soustraction du fond est utilisée pour trouver la variation totale soudaine de l'intensité ou de la vidéo. La détection des contours est réalisée en tant que travail intermédiaire avant la segmentation par l'extraction de l'objet à partir du fond. Enfin le suivi sera effectué. L'objectif principal de la poursuite d'objet est de détecter et de suivre l'objet en mouvement à travers la séquence vidéo. La plupart des techniques de traitement d'image couvertes dans ce chapitre sont liées aux résultats des expérimentations de suivi des objets.

V.2 Simulink

Simulink est une extension graphique de MATLAB pour la modélisation et la simulation de systèmes. Il prend en charge les systèmes linéaires et non linéaires, sur le modèle en temps continu, temps échantillonné, ou un mélange des deux.

Dans Simulink, les systèmes sont dessinés sur l'écran sous forme de diagrammes de blocs. De nombreux éléments de diagrammes sont disponibles, telles que les fonctions de transfert, les jonctions de sommation, etc, ainsi que l'entrée virtuelle et les périphériques de sortie tels que les générateurs de fonctions et oscilloscopes. Simulink est intégré avec MATLAB et les données peuvent être facilement transférées entre les programmes.

V.3 La Conception, la simulation vidéo, et les systèmes de traitement d'image

Vidéo et Image Processing Blockset fournit des algorithmes et des outils pour la conception et la simulation de traitement vidéo, traitement d'image, et les systèmes de vision par ordinateur. Vous pouvez traiter les données vidéo et image pour résoudre des problèmes tels que le bruit, faible contraste, et le flou optique. Vous pouvez ensuite effectuer des tâches telles que l'analyse de mouvement, détection et suivi d'un objet, la stabilisation vidéo, et l'estimation de la disparité pour la vision stéréo. La plupart des algorithmes et des outils sont disponibles comme Objets Système (pour une utilisation dans MATLAB) et comme Blocs (pour utilisation dans Simulink).

Vidéo et Image Processing Blockset peut lire et écrire des fichiers multimédias dans un large éventail de formats, tels que AVI, MPEG et WMA. Vous pouvez diffuser de la vidéo à partir de sources ou MMS via Internet ou un réseau local. Vous pouvez acquérir la vidéo directement à partir de caméras Web, cartes d'acquisition vidéo, et d'autres appareils d'imagerie utilisant Image Acquisition Toolbox.

Le blockset inclut un visionneur vidéo flexible avec beaucoup de caractéristiques. Vous pouvez :

. Voir tout signal vidéo au sein de votre simulation.

. Utilisez plusieurs visionneurs vidéo en même temps.

. Démarrer, arrêter, mettre en pause, et exécuter pas à pas une trame à la fois.

. Zoomer et faire la panoramique pour regarder de plus près lorsque la simulation est en cours.

V.4 Le système de suivi

Ce diagramme résume les différentes étapes du système de suivi:

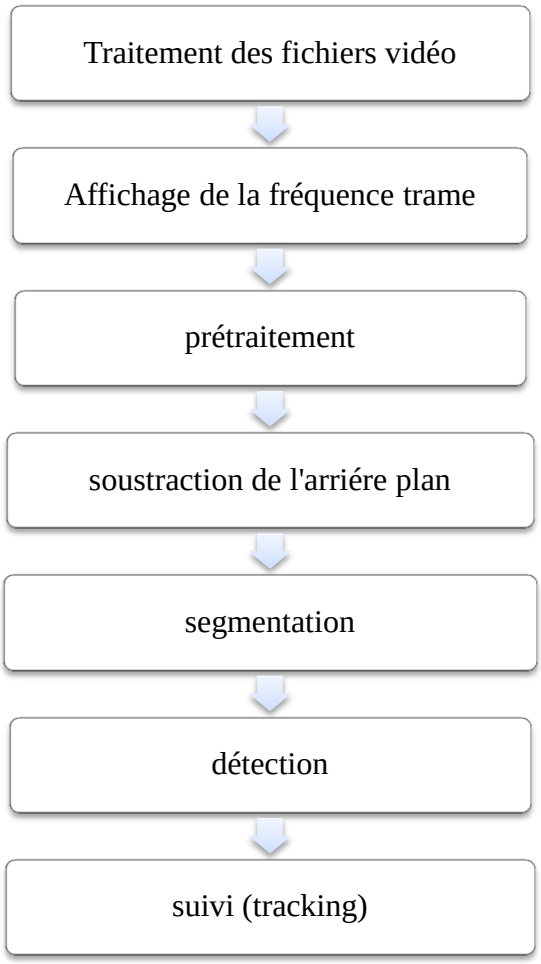


Figure.V.1 Le système de suivi.

V.4.1 Prétraitement

Le prétraitement est principalement utilisé pour améliorer le contraste de l'image, la suppression du bruit et pour isoler les objets d'intérêt dans l'image. Le prétraitement est une forme de traitement de signal pour lequel la sortie est une image ou une vidéo, la sortie peut être soit une image ou un ensemble de caractéristiques ou des paramètres liés à l'image ou à la vidéo afin d'améliorer ou de modifier une certaine qualité de l'entrée **[48]**. Le prétraitement permet d'améliorer la vidéo ou l'image pour augmenter les chances de succès d'autres processus.

V.4.1.1 Affichage de la fréquence trame

Ce processus permet de calculer et afficher la vitesse de trame du signal d'entrée.

Il permet également de calculer le paramètre du taux d'affichage pour contrôler la fréquence d'affichage. Lorsque ce paramètre est supérieur à 1, le bloc affiche le taux moyen de mise à jour pour le nombre spécifié de trames vidéo. Par exemple, si vous entrez 10, le bloc calcule la quantité de temps qu'il faut pour le modèle de passer 10 trames vidéo au bloc. Il divise ce temps par 10 et affiche ce taux de trame vidéo moyenne sur le bloc.

Ce bloc est utilisé pour vérifier le taux d'image vidéo de la simulation pour contrôler le nombre spécifié d'images vidéo.

V.4.1.2 La conversion des couleurs

Conversion de l'espace couleur est la représentation d'une couleur d'une base à une autre. L'objectif est de faire de l'image traduite paraitre aussi proche que possible des conversions originales des couleurs. Une couleur peut être représentée par teinte, la saturation et de la valeur des composants. Il fournit un ensemble de routines pour la conversion entre RGB et d'autres espaces couleur. Les fonctions de traitement d'image elles-mêmes supposent toutes les données de couleur en RGB et traitent une image qui utilise un espace de couleur différent en la convertissant d'abord en RGB, puis ensuite en la reconvertissant dans l'espace de couleur d'origine. Le bloc de conversion d'espace couleur permet de convertir les données de couleur de l'espace de couleur R'G'B' vers l'espace de couleur Y'CbCr et de l'espace de couleur Y'CbCr à l'espace de couleur du R'G'B'. Ce bloc peut également être utilisé pour convertir de l'espace colorimétrique R'G'B' à l'intensité.

V.4.1.3 La soustraction de l'arrière-plan

La soustraction de l'arrière-plan comme son nom l'indique est le processus de séparation des objets d'avant-plan à partir de l'arrière-plan dans une séquence de trames vidéo.

Soustraction de fond consiste à comparer l'image observée avec une estimation de l'image, si elle ne contenait pas d'objet d'intérêt. Dans ce processus de soustraction de fond, le

modèle Simulink est utilisé pour spécifier le bloc modifié des paramètres (edit parameter) pour estimer l'arrière-plan.

V.5 Modèle exemplaire de suivi des personnes

La figure suivante montre le modèle exemplaire de suivi des personnes :

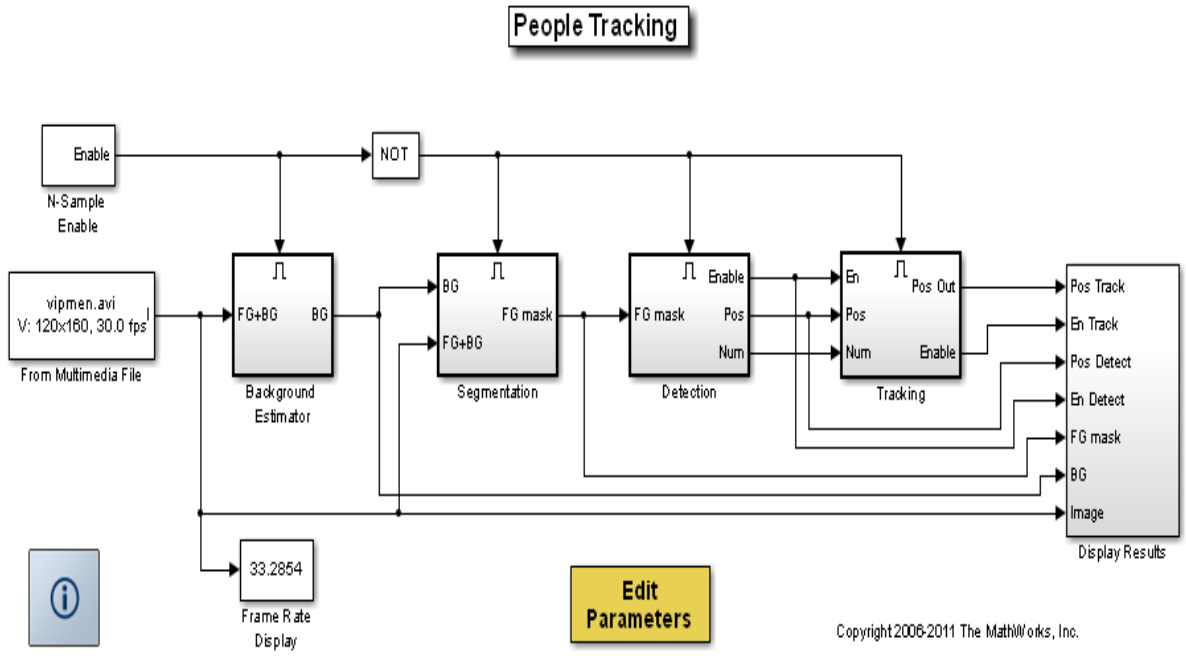


Figure.V.2 Modèle de suivi des personnes.

V.5.1 À propos de la vidéo

La vidéo ci-dessus a été créée à l'aide du bloc écriture de fichier AVI. Lors de l'exécution de ce modèle Simulink, vous pouvez voir la vidéo obtenue en utilisant un visualiseur vidéo ou Au bloc de l'affichage vidéo. Vous pouvez connecter ces visionneurs à tout signal dans le modèle, ajuster les paramètres du modèle, et voir les résultats en temps réel.

V.5.2 Edit Paramètres

Si vous souhaitez substituer vos propres vidéos dans cette démo, vous pouvez modifier les paramètres de la démo pour adapter les valeurs de luminance, la taille de la cible et mouvement dans votre flux vidéo.

V.5.3 Estimation de l'arrière-plan

Dans le sous-système de l'estimateur de l'arrière-plan, l'exemple utilise les premières images de la vidéo pour estimer l'image d'arrière-plan. Dans le sous-système Tracker, l'exemple suivant soustrait l'arrière-plan de chaque image vidéo pour produire des images d'avant-plan.

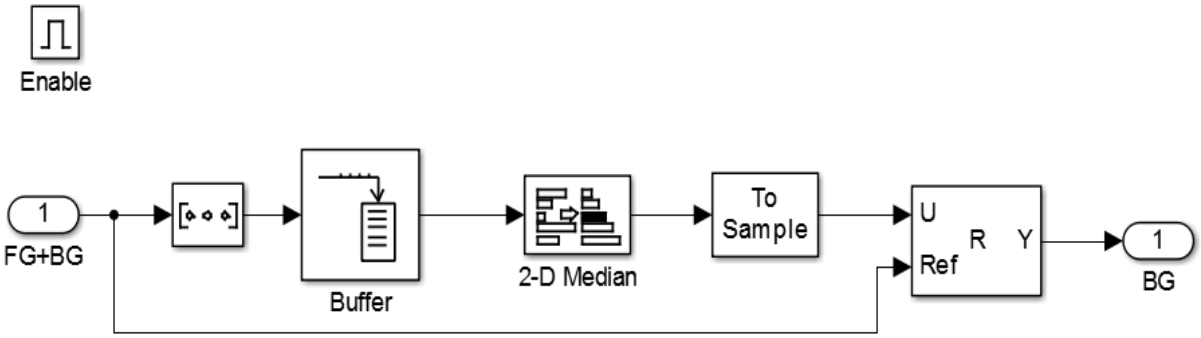


Figure.V.3 Estimation de l'arrière-plan.

V.5.4 Sous-système de segmentation

Dans le sous-système de la Segmentation, le bloc de 'Autothreshold' utilise la différence de valeurs des pixels entre l'image d'entrée normalisé et l'image d'arrière-plan pour déterminer quels pixels correspondent à des objets en mouvement dans la scène.

Autothreshold :

Convertit automatiquement une image intensité vers une image binaire. Ce bloc utilise la méthode d'Otsu, qui détermine le seuil en fractionnant l'histogramme de l'image d'entrée telle que la variance pour chacun des groupes de pixel est réduit au minimum. Éventuellement, le bloc peut produire une métrique qui indique l'efficacité de seuillage de l'image d'entrée. La limite inférieure de la métrique (zéro) est accessible seulement par les images ayant un seul niveau de gris, et la limite supérieure (un) est accessible seulement par les images à deux valeurs.

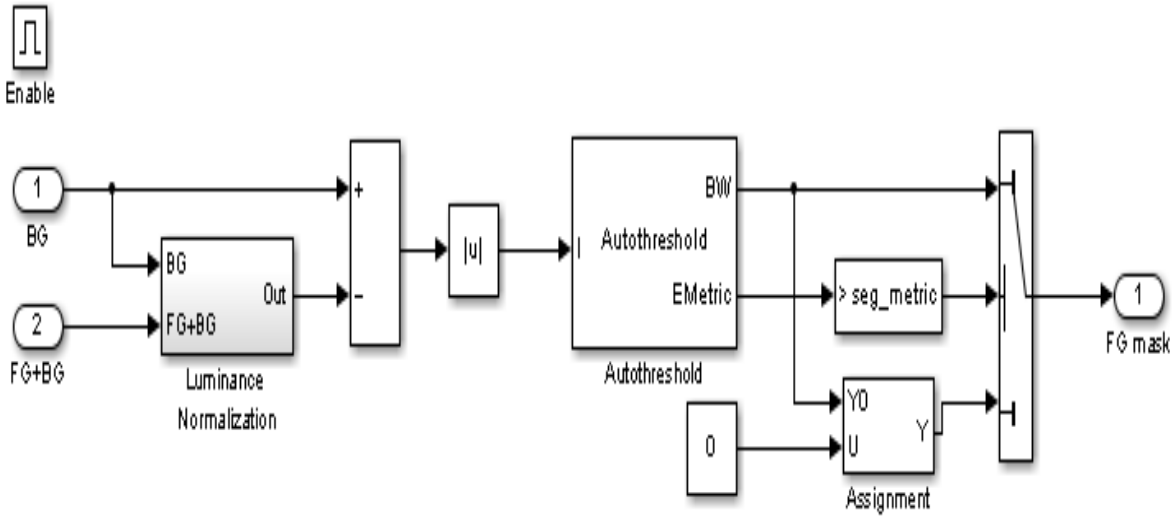


Figure.V.4 Sous-système de segmentation.

V.5.5 Sous-système de détection

Dans le sous-système de détection, le bloc Fermer fusionne les pixels d'objet qui sont proches les uns des autres pour créer des taches (Blobs). Par exemple, les pixels qui représentent une partie du corps d'une personne sont regroupés ensemble. Ensuite, le bloc Analyse Blob calcule les boîtes entourantes de ces taches. Dans la dernière étape, le sous-système de détection fusionne les boîtes entourantes individuelles de sorte que chaque personne est entourée par une seule boite entourante.

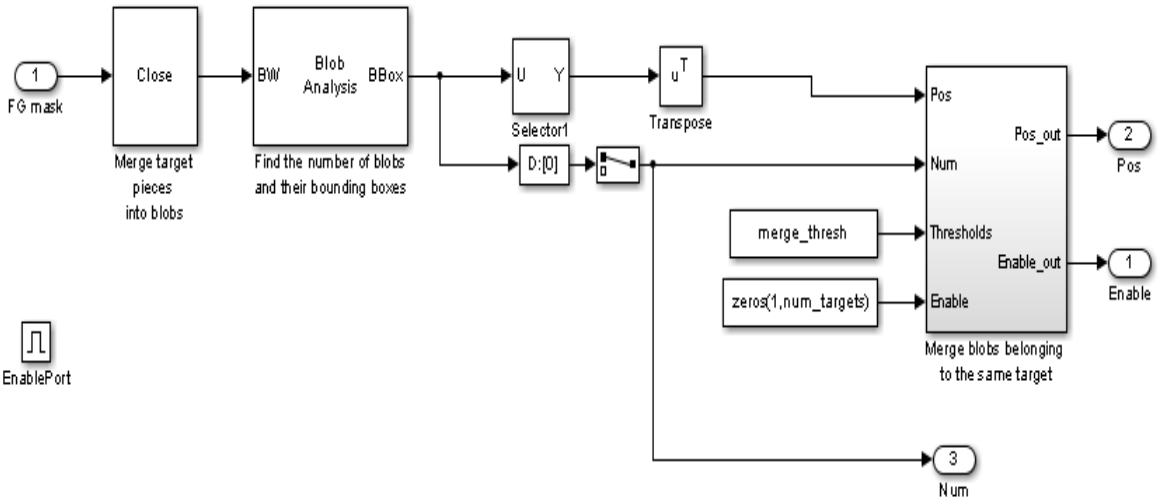


Figure.V.5 Sous-système de détection.

V.5.6 Sous-système de suivi

Dans ce sous- système, on remplace le bloc du filtre de Kalman (Figure.V.6) par celui du filtre particulaire (Figure.V.7) ; ce dernier utilise les emplacements des boîtes entourantes détectés dans les images précédentes pour prédire leurs nouveaux emplacements dans la trame courante.

Le filtre particulaire est conçu pour le suivi. Vous pouvez l'utiliser pour prédire le futur emplacement d'un objet physique, pour réduire le bruit dans l'emplacement détecté, ou pour suivre plusieurs objets physiques associés avec leurs titres correspondants. Un objet de filtre particulaire peut être configuré pour chaque objet physique pour le suivi d'objets multiples.

L'algorithme de filtre de particulaire comprend deux étapes, de prédiction et de correction (également connu sous le nom de l'étape de mise à jour). La première étape utilise les états antérieurs pour prédire l'état actuel. La deuxième étape utilise la mesure courante, tels que l'emplacement de l'objet, afin de corriger l'état. Le filtre particulaire est à temps discret, l'état du système-espace est décrit par une équation non linéaire.

Pour déterminer l'emplacement des personnes spécifiques d'une image à l'autre, l'exemple compare les emplacements prévus des boîtes entourantes avec les emplacements détectés. Cela permet par exemple d'attribuer une couleur unique à chaque personne. L'exemple utilise également le bloc de filtre particulaire afin de réduire l'effet du bruit dans la détection de l'emplacement de la boîte entourante.

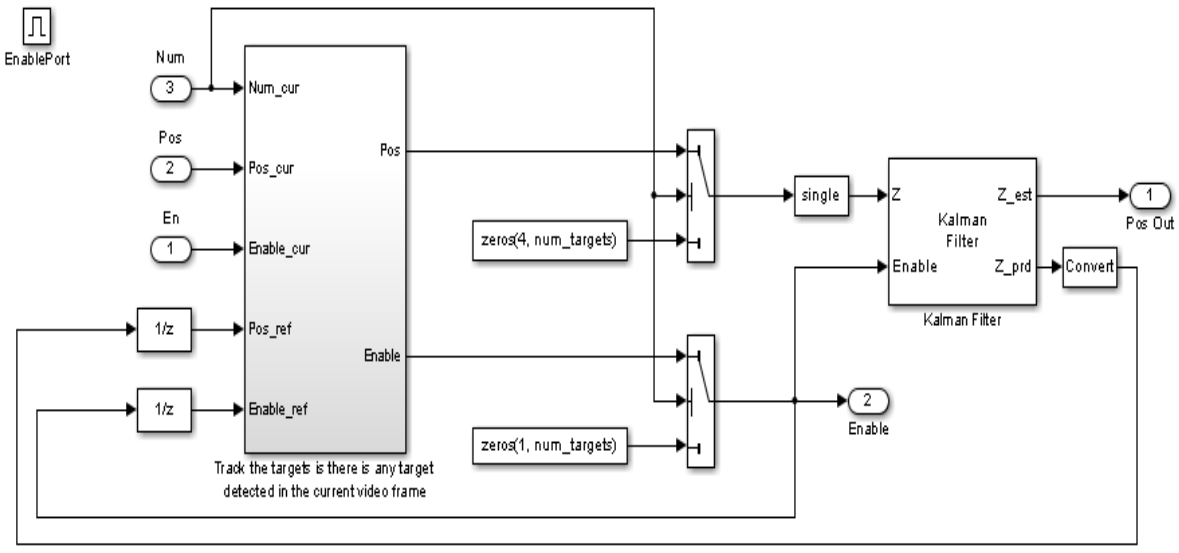


Figure.V.6 Sous-système de suivi utilisant le filtre de kalman.

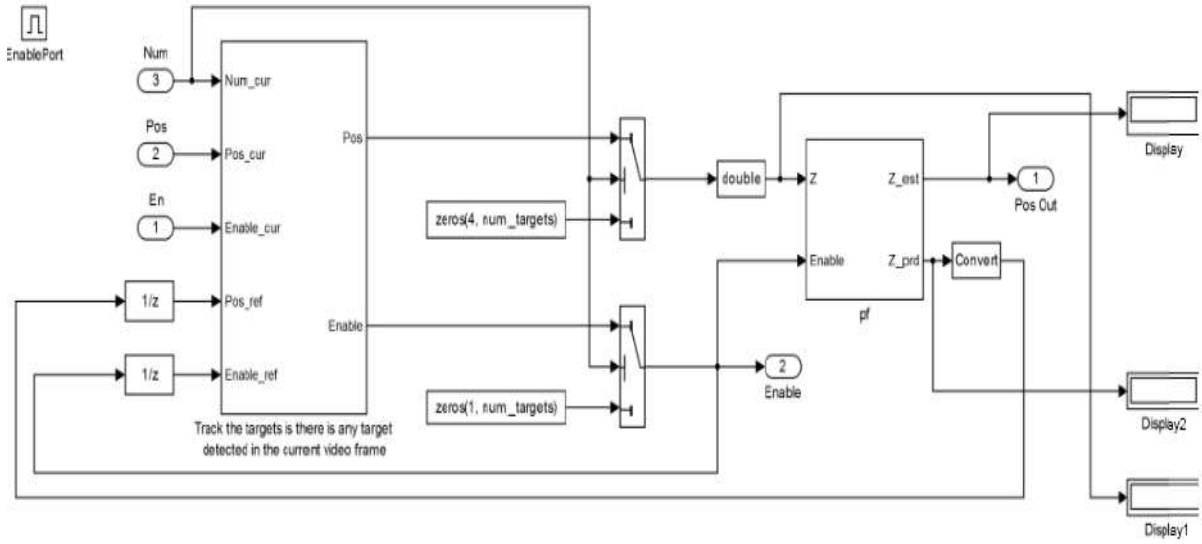


Figure.V.7 Sous-système de suivi utilisant le filtre particulaire.

V.6 Les résultats de suivi des personnes

- ***Suivi d'une seule personne***

On obtient les résultats suivants :

- **pour la première vidéo :**





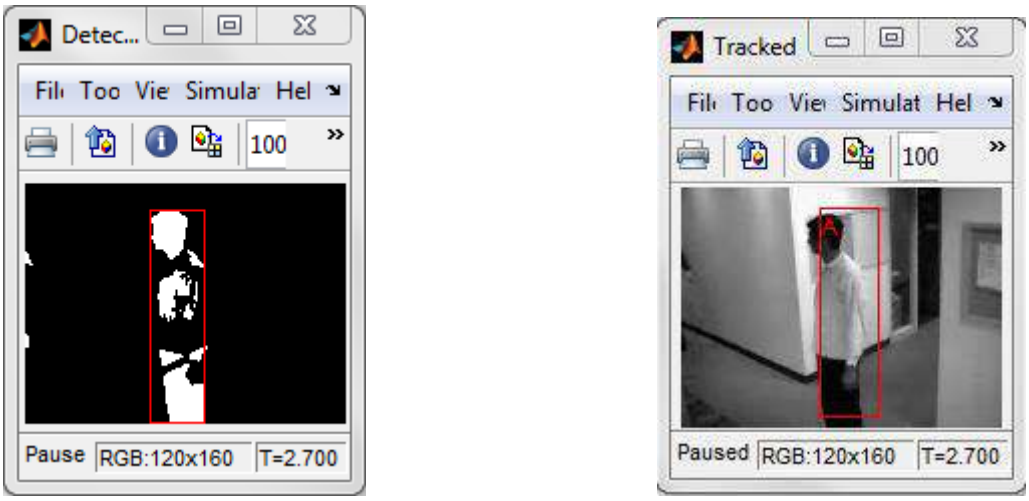


Figure.V.8 Suivi d'une seule personne pour la première vidéo.

- **Pour la deuxième vidéo :**

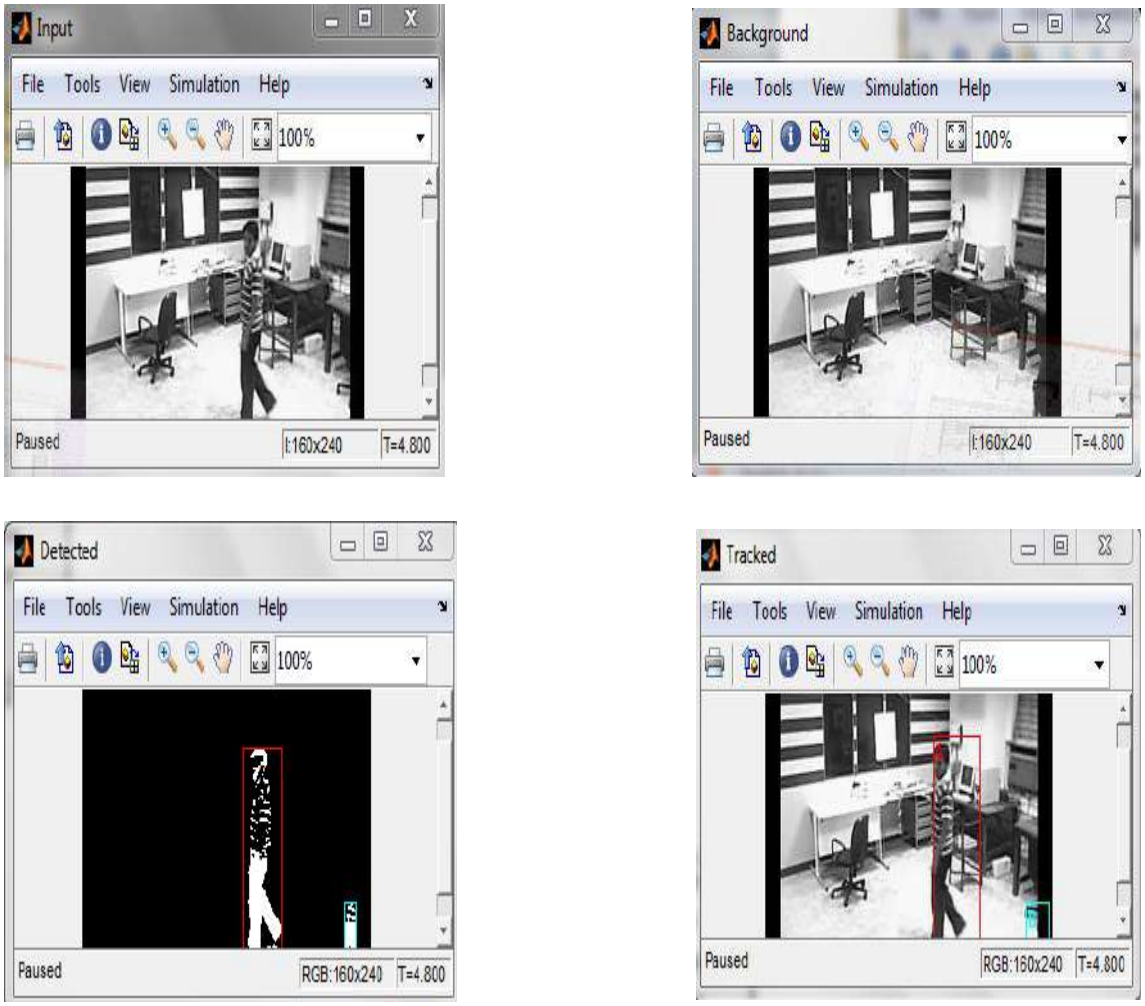


Figure.V.9 Suivi d'une seule personne pour la deuxième vidéo.

- **Pour la troisième vidéo :**



Figure.V.10 Suivi d'une seule personne pour la troisième vidéo.

Les résultats obtenus sont satisfaisants puisque la boîte entourante obtenue suit bien la personne en mouvement.

- ***Suivi de deux personnes***

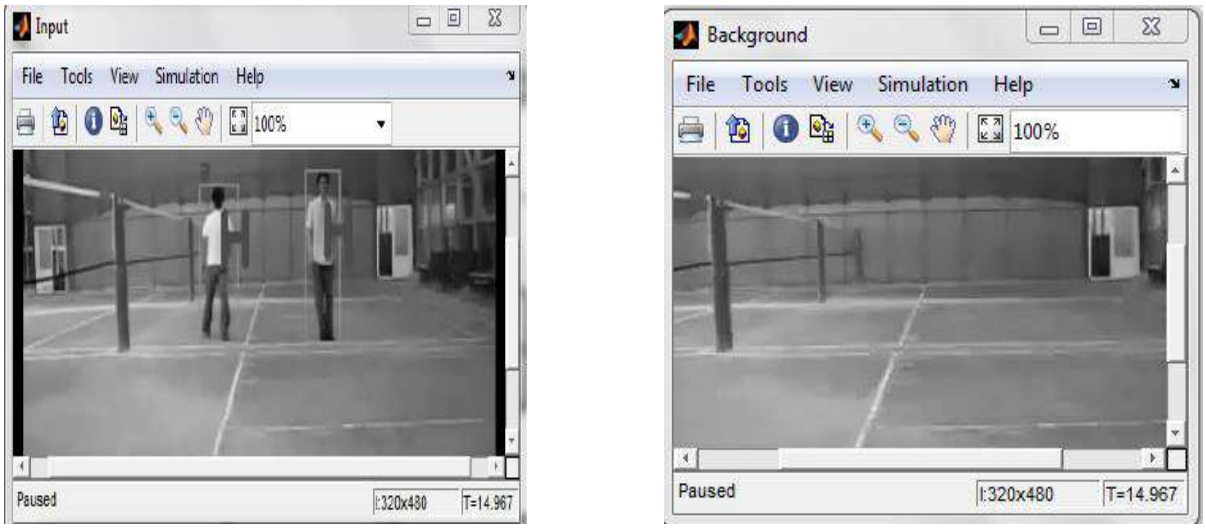
On obtient les résultats suivants :

- **pour la première vidéo :**



Figure.V.11 Suivi de deux personnes pour la première vidéo.

- **Pour la deuxième vidéo :**



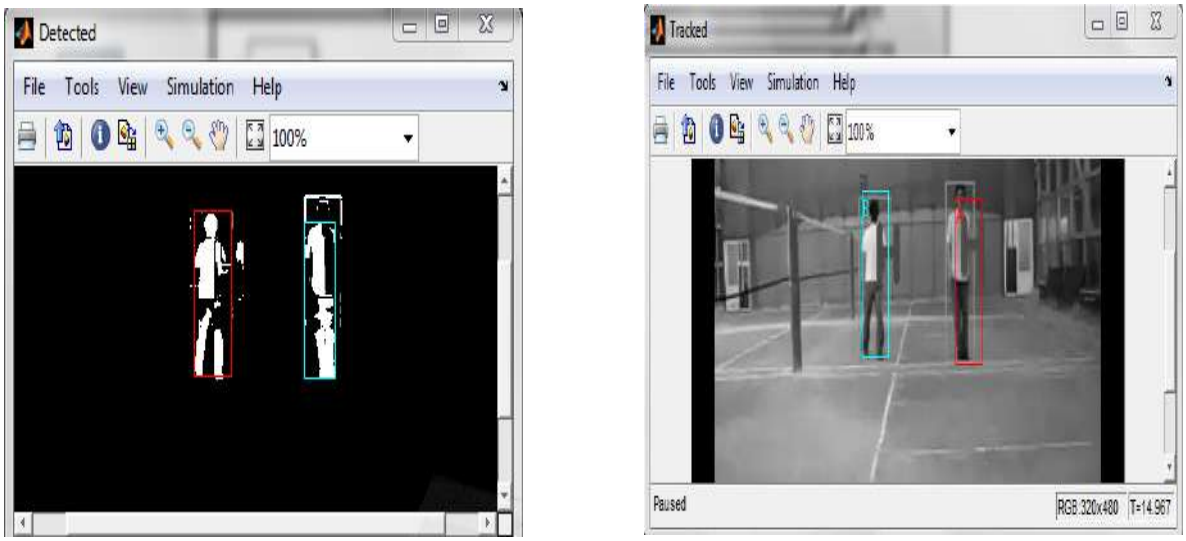


Figure.V.12 Suivi de deux personnes pour la deuxième vidéo.

- **Pour la troisième vidéo :**



Figure.V.13 Suivi de deux personnes pour la troisième vidéo.

Les résultats obtenus sont satisfaisants puisque les boîtes entourantes obtenues suivent bien les personnes en mouvement.

- ***Suivi de plusieurs personnes***

On obtient ce résultat :



Figure.V.14 Suivi de plusieurs personnes.

Les résultats obtenus sont satisfaisants puisque les boîtes entourantes obtenues suivent bien les personnes en mouvement.

V.6.1 Interprétations des résultats de suivi des personnes

Dans la fenêtre détectée, les personnes dans la scène sont entourées par des boîtes entourantes. L'exemple assigne à chaque boîte entourante une couleur basée sur l'ordre à partir duquel chaque personne est détectée. Par exemple, la première personne détectée a une boîte entourante rouge et la deuxième personne détectée a une boîte verte. La couleur de ces boîtes change parce que les personnes dans la scène ne sont pas suivies.

Dans la fenêtre de suivi, chaque personne a une couleur unique de la boîte entrourante pour la durée de la vidéo.

On peut double-cliquer sur la case ' Edit Parameters box ' et sélectionner Plot positions des boîtes entourantes au cours du temps.

Ensuite, exécutez l'exemple à nouveau,

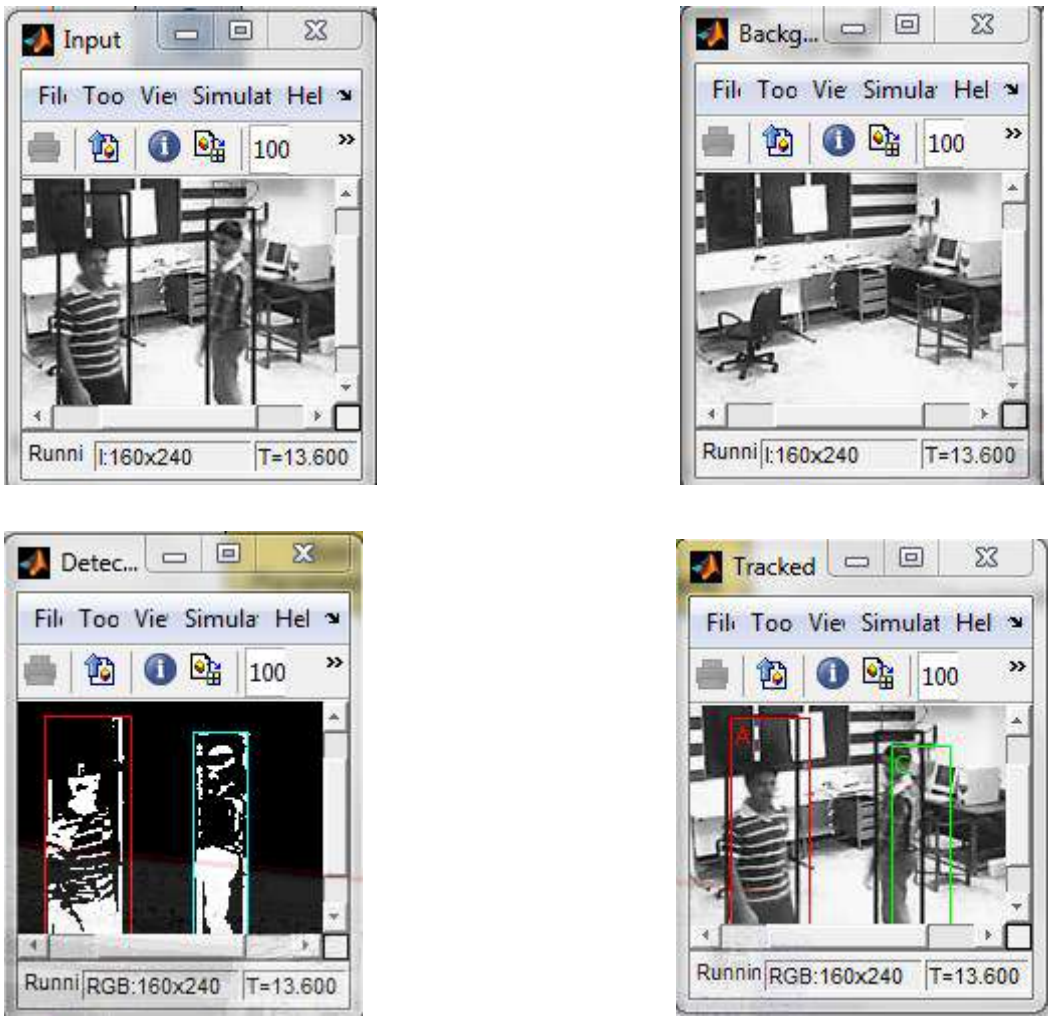


Figure.V.15 Suivi de deux personnes.

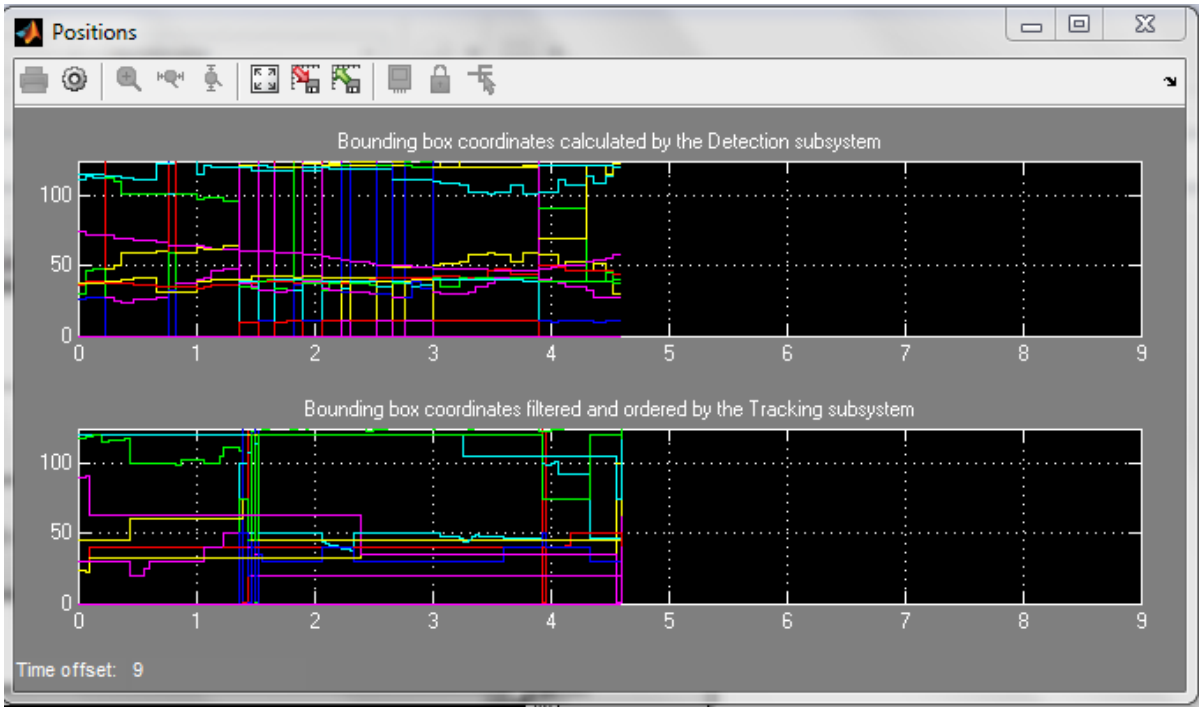


Figure.V.16 Plot positions des boîtes entourantes au cours du temps.

Dans la fenêtre de positions, l'exemple trace les coordonnées des boîtes entourantes au fil du temps. Les coordonnées de chaque boîte sont définies par la rangée et la colonne de l'emplacement de son coin supérieur-gauche aussi bien que par sa largeur et sa hauteur. En conséquence, chaque personne dans la vidéo correspond à quatre lignes dans le dessin.

Parce que le bloc de Filtre particulaire réduit le bruit, les positions des boîtes entourantes calculées par le sous-système de suivi ont des trajectoires plus lisses que ceux calculés par le sous-système de Détection.

V.6.2 Comparaison entre le filtre Kalman et le filtre particulaire

V.6.2.1 Le filtre Kalman

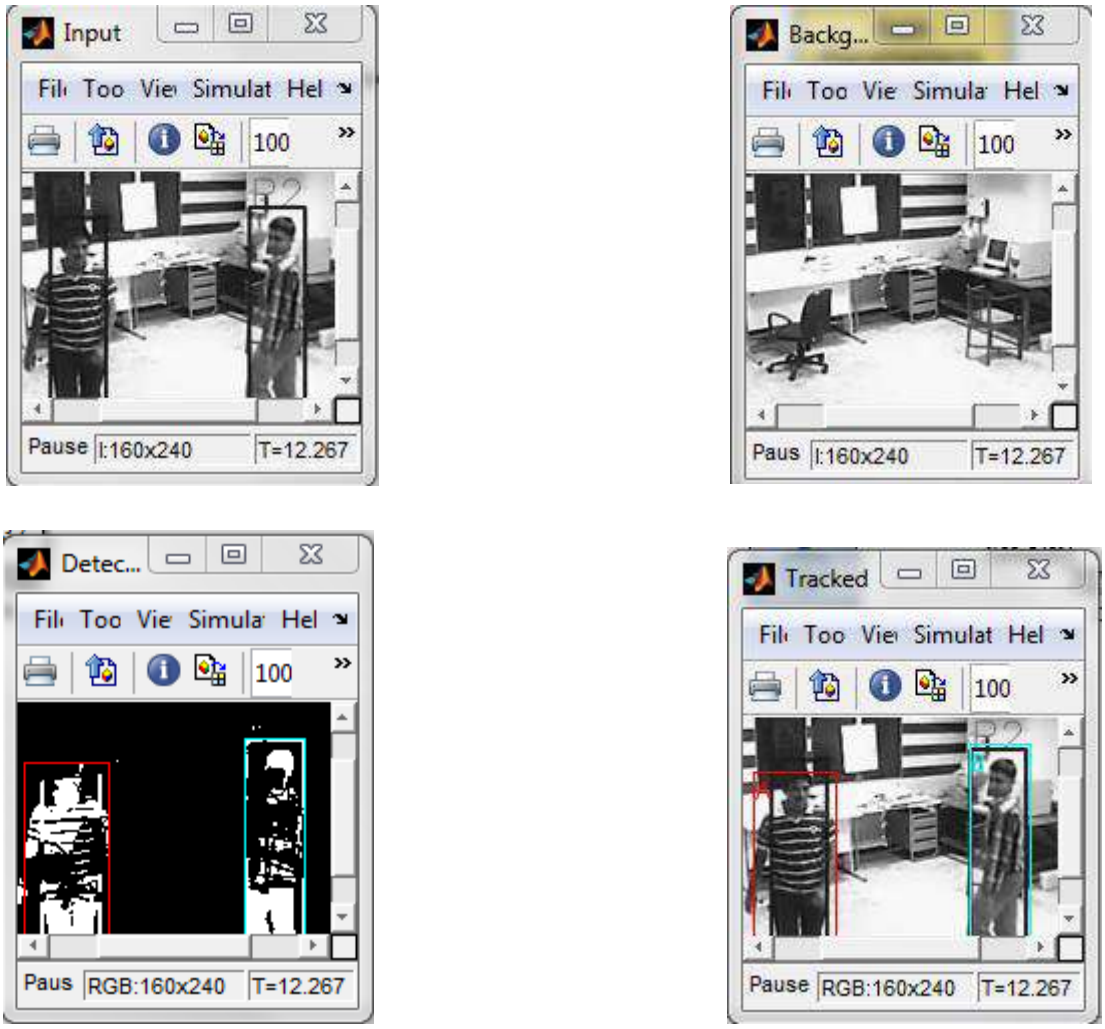


Figure.V.17 Suivi de deux personnes par filtre Kalman.

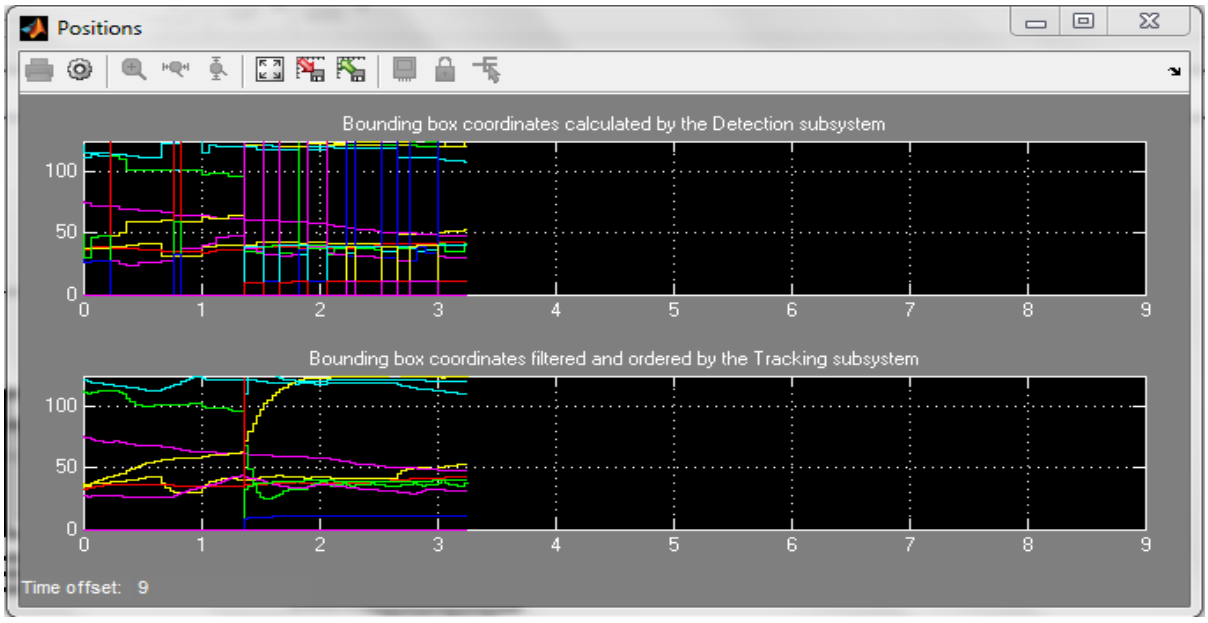


Figure.V.18 Plot positions des boîtes entourantes au cours du temps par filtre Kalman.

V.6.2.2 Filtre particulaire



Figure.V.19 Suivi de deux personnes par filtre particulaire.

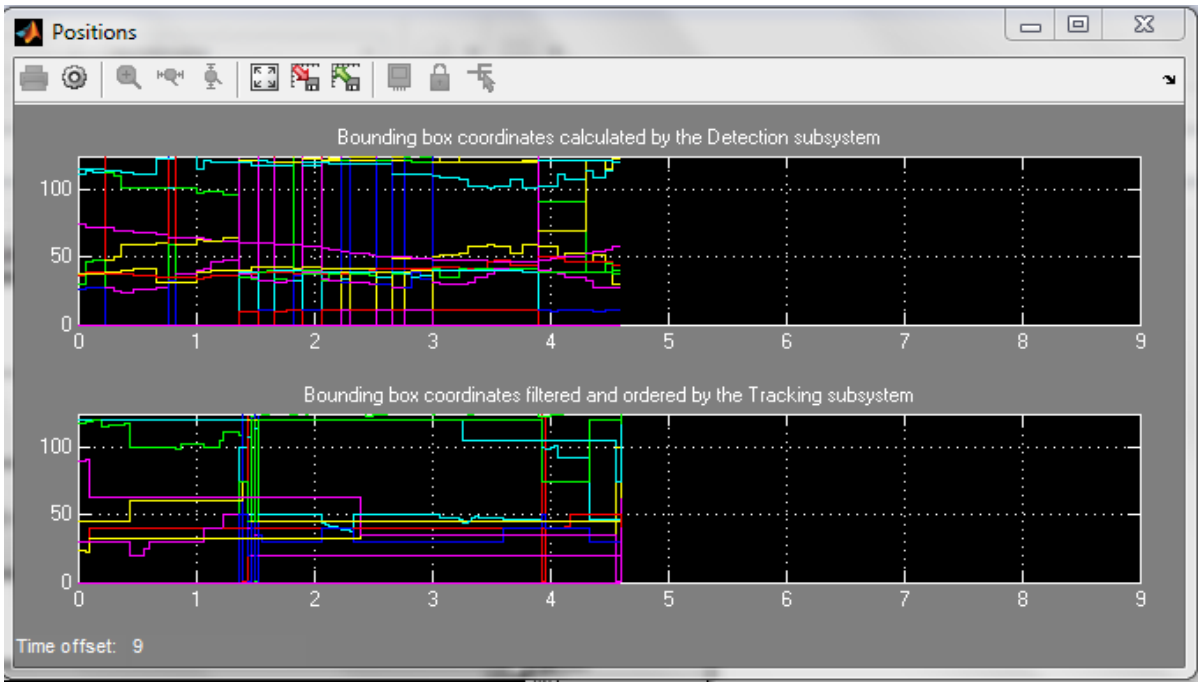


Figure.V.20 Plot positions des boîtes entourantes au cours du temps par filtre particulaire.

- ✓ Le temps de calcul du filtre particulaire est beaucoup plus grand que celui du filtre Kalman.
- ✓ Les positions de boîte entourantes calculées par le sous-système de suivi du filtre particulaire ont des trajectoires plus lisses que ceux calculés par le sous-système de suivi qui utilise le filtre de Kalman.

V.7 Conclusion

Cet exemple montre comment détecter et suivre les personnes dans une séquence vidéo avec un arrière-plan en utilisant le processus suivant :

1) Utiliser les premières images de la vidéo pour estimer l'image d'arrière-plan.

2) Séparez les pixels qui représentent les gens des pixels qui représentent l'arrière-plan.

3) Grouper les pixels qui représentent les personnes ensemble et calculer les boîtes entourantes pour chaque personne.

4) Correspondre les personnes dans la trame active avec ceux de la trame précédente en comparant les boîtes entourantes entre les trames.

Pour déterminer l'emplacement des personnes spécifiques d'une image à l'autre, l'exemple compare les emplacements prévus des boîtes entourantes avec les emplacements détectés. Cela permet à l'exemple d'attribuer une couleur unique à chaque personne. L'exemple utilise également le bloc de filtre de particulaire afin de réduire l'effet du bruit dans la détection de l'emplacement de la boîte entourante. Les positions de boîte entourantes calculées par le sous-système de suivi ont des trajectoires plus lisses que ceux calculés par le sous-système de Détection.

Le temps de calcul du filtre particulaire est beaucoup plus grand que celui du filtre Kalman.

Conclusion Générale

Conclusion générale

L'objectif du suivi d'objets est d'associer les objets cibles dans les images vidéo consécutives. Il est très utile pour l'entrainement militaire, le sport, et les applications médicales et pour la validation de la vision par ordinateur et de la robotique. Le suivi dans les vidéos et images de l'open source peut être accompli et il peut donc être facilement mis en œuvre et son utilisation est plus large. L'implémentation de cette méthodologie utilisant le blockset de Simulink est très utile pour la sécurité et la surveillance vidéo. Ce système fournit également des services supplémentaires, tels que des informations sur l'emplacement et l'identité des objets à différents points dans le temps et constitue la base pour détecter les mouvements d'objets insolites. Le suivi de la position réelle des objets se fait par le suivi de son état en utilisant le filtrage de la région et ceci utilise les informations du blob en cours et l'état des objets précédents pour créer une estimation du nouvel état des objets.

Dans ce mémoire, nous avons décrit les deux principales méthodes utilisées dans le filtrage qui sont les méthodes analytiques et numériques. Dans les méthodes analytiques, le filtre de Kalman étendu est le filtre par excellence des ingénieurs, il donne de très bons résultats dans le cadre des systèmes faiblement non-linéaires. Mais lorsque les systèmes sont fortement non linéaires, les performances du filtre de Kalman étendu sont mauvaises. Quant aux méthodes numériques, elles fournissent des résultats d'une bonne précision au problème du filtrage non linéaire. Cependant, elles restent limitées à des espaces de dimension au plus égale à 3. Au-delà de cette dimension, le temps de calcul augmente d'une façon exponentielle avec la dimension de l'espace, ce qui les rend impraticables en temps réel. C'est pour cela qu'une nouvelle catégorie de méthodes a été introduite : les méthodes de Monte Carlo. Elles sont mieux adaptées au problème du filtrage non linéaire car elles sont peu sensibles à la dimension de l'espace d'état. Nous avons appliqué ce type de méthodes, appelées filtrage particulaire.

L'objet de filtre particulaire est conçu pour le suivi. Vous pouvez l'utiliser pour prédire le futur emplacement d'un objet physique, pour réduire le bruit dans l'emplacement détecté, ou pour détecter plusieurs objets physiques associés avec leurs propriétés correspondantes. Un objet de filtre particulaire peut être configuré pour chaque objet physique pour le suivi d'objets multiples.

L'algorithme de filtre de particulaire comprend deux étapes, de prédiction et de correction (également connu sous le nom de l'étape de mise à jour). La première étape utilise les états antérieurs afin de prédire l'état actuel. La deuxième étape utilise la mesure courante, tel que l'emplacement de l'objet, afin de corriger l'état.

Dans ce mémoire, nous avons implémenté sur Matlab le filtre à particule pour le suivi d'objets dans une séance vidéo. Les résultats obtenus sont très satisfaisants et sont meilleurs que ceux obtenus par filtre de Kalman. Cependant, la complexité de calcul est très élevée et comme perspective, on peut citer le développement de méthodes moins complexes et donc plus rapides et moins gourmande en énergie et en composants (lors d'une implémentation hardware sur FPGA par exemple).

Bibliographie

[1] F. Berry, Technologies des capteurs d'images couleurs, In Imagerie Numérique CouleurGFINC, G. F, editor, EHINC'05, Lille, 2005.

[2] C. Cafforioet F. Rocca, Methods for measuring small displacements of television images, IEEE-Trans. Information Theory, IT-22, n° 5, p. 573-579, sept. 1976.

[3] J. L. Barron, D. J. Fleet, et Beauchemin, S. S, Performance of optical ow techniques, Int. Journal of Computer Vision, 12(1):43 -77, 1994.

[4] T. Yang, S. Z. Li, Q. Pan, et J, Li, Real-time and accurate segmentation of moving objects in dynamic scene, In Proc. ACM 2nd Int. Workshop on Video Surveillance & Sensor Networks, pages 136-143, New York (NY), USA, VSSN 2004.

[5] M. Chaumont, représentation en objets vidéo pour un codage progressif et concurrentiel des séquences d'images, thèse doctorat, université de rennes 1, novembre 2000.

[6] P. Salembier and M. Pardas, Hierarchical morphological segmentation for image sequence coding, IEEE Trans. Image Processing, vol. 3, no.5, pp. 639–651, 1994.

[7] Vincent Agnus, Segmentation spatio-temporelle de séquence d'image par des opérateurs des morphologies mathématique, thèse doctorat. Université Louis Pasteur de Strasbourg-Ecole Doctorale SPI. pp 22, 2001.

[8] S. Castan, J. Zhao, J. Shen, Une famille de Détection de Contour Basée sur Filtre Exponentiel Optimal, 7ème Edition 1986.

[9] Christophe Fiorio, Analyse d'image, Split and Merge, Laboratoire d'Informatique, de Robotique et Microélectronique de Montpellier LIRMM France Novembre 2005.

[10] R. Jain, Différence and accumulative difference pictures in dynamic scene analysis, Image and Vision Computing, 2(2):99–108, may 1984.

[11] A. Makarov, Comparison of background extraction based intrusion detection algorithms, Dans International Conference on Image Processing ICIP '96, pages 521–524, 1996.

[12] S. Haynes, Detection of moving edges, Computer Graphics and Image Processing, 21, 1983.

[13] P. Stelmaszyk. Analyse de scènes dynamiques par recherche des contours en mouvement. PhD thesis, Université des Sciences and Technologies de Lille, Novembre 1985.

[14] C. Vieren, Segmentation de scènes dynamiques en temps réel. Application au traitement de séquences d'images pour la surveillance de carrefours routiers, PhD thesis, Université des Sciences and Technologies de Lille, Avril 1988.

[15] M. Orkisz, ocalisation d'objets mobiles dans les scènes naturelles filmées par une caméra fixe, Traitement du Signal, 9(4) :325–346, 1992.

[16] D.S. Zhang et G. Lu, Segmentation of Moving Objects in Image Sequence: A Review, Circuits, Systems and Signal Processing (Special Issue on Multimedia Communication Services), volume 20(2), p 143-183, 2001.

[17] E. François, Interprétation qualitative du mouvement à partir d'une séquence d'images, Thèse de Doctorat de l'Université de Rennes I, Juin 1991.

[18] M. Gelgon, Segmentation spatio-temporelle et suivi dans une séquence d'images : application à la structuration et à l'indexation de vidéo, Thèse de doctorat, Université de Rennes 1, 1998.

[19] D.G. Lowe, Robust model-based motion tracking through the intégration of search and estimation, International Journal of Computer Vision, 8(2):113–122, 1992.

[20] B. Bascle, P. Bouthemy, R. Deriche, et F. Meyer, Suivi de primitives complexes sur une séquence d'images, Rapport de recherche 2428, INRIA-Sophia Antipolis, Décembre 1994. 22 pages.

[21] S. Jehan-Besson, Modèles de contours actifs basés régions pour la segmentation d'images et de vidéos, Thèse de Doctorat de l'Université de Nice - Sophia Antipolis, Janvier 2003.

[22] F.G. Meyer et P. Bouthemy, Region-based tracking using affine motion models in long image sequences, CVGIP : Image Understanding, 60(2) :119–140, September 1994.

[23] B. Bascle et R. Deriche, Region tracking through image sequences, Rapport de recherche 2439, INRIA-Sophia Antipolis, 21 pages, Décembre 1994.

[24] V. Garcia-Garduno et C. Labit, Suivi de segmentation spatiotemporelle pour la compression de séquences animées, Quatorzième Colloque GRETSI, Juan-les-Pins, Septembre 1993.

[25] V. Garcia-Garduno, Une approche de compression orientée-objets par suivi de segmentation basée mouvement pour le codage de séquences d'images numériques, Thèse de doctorat, Université de Rennes 1, 1995.

[26] A. Yilmaz, X. Li et M. Shah, Object tracking: a survey, ACM New York, NY, USA, 2006.

[27] C.J. Veenman, M.J.T. Reinders et E. Backer**,** Resolving Motion correspondence for Densely Moving Points, IEEE Transactions on Pattern, 2001.

[28] S. Birchfield, Elliptical Head Tracking Using Intensity Gradients and Color Histograms, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Santa Barbara, California, pages 232-237, June 1998.

[29] Mohand Said Djouadi, 'Poursuite de Cibles Manoeuvrants par vision Active. ' Thèse de Doctorat, Ecole Nationale Polytechnique d'Alger, Décembre 2005.

[30] Kalman, R. E., "A new approach to linear filtering and prediction problems," vol. 82, pp. 34-45, 1960.

[31] Kalman, R. E. and Bucy, R., "A new approach to linear filtering and prediction problems theory," vol. 83, pp. 95-108, 1961.

[32] Anderson, B. D. O. and Moore, J. B., Optimal Filtering. Providence, RI: Pentice-Hall, 1979.

[33] Liptser, R. S. and Shiryayev, A. N., Statistics of random processes. Springer, 1977.

[34] Picard, J., Asymptotic study of estimation problems with small observation noise. In stochastic modelling and Filtering, 91 of Lecture Notes in Control and Information Sciences. Springer, 1987.

[35] Getout-Petit, A., "Approximate filter for the conditional law of partially observed process in nonlinear filtering," SIAM Journal on Control and Optimization, vol. 36, no. 4, pp. 1423-1447, 1998.

[36] Julier, S. J. and Uhlmann, J. K., ,"A new extention of the Kalman filter to nonlinear systems," in The 11 th International Symposium on Aerospace/Defence Sensing, Simulation and Controls, Vol. Multi Sensor Fusion, Tracking and Resource Management II, (Orlando, Florida), 1997.

[37] Jazwinski, A. H., Stochastic Processes and Filtering Theory. New-York: Academic Press, 1970.

[38] Cai, Z., LeGland, F., and Zhang, H., "An adaptative local grid refinement method for nonlinear filtering," Tech. Rep. Technical report 2679, INRIA, 1995.

[39] Doucet, A., Godsill, S. J., and Andrieu, C., "On sequential Monte Carlo sampling methods for Bayesian filtering," Statistics and Computing, vol. 10, no. 3, pp. 197-208, 2000.

[40] Oudjane, N., Stabilité et Approximations Particulaires en Filtrage non linéaire Application au pistage. Ph. D. Thesis, L'Université de Rennes 1, France, Dec. 2000.

[41] Doucet, A., "On sequential simulation based methods for Bayesian filtering," Tech. Rep. Technical report CUFD/F-INFENG/TR.310, Signal Processing group, Department of Engineering, University of Cambridge, 1998.

[42] Kitagawa, G., "Monte Carlo filter and smoother for non-Gaussian nonlinear state space models," vol. 5, no. 1, pp. 1{25, 1996.

[43] Kong, A., Liu, J. S., and Wong, W. H., "Sequential imputations and Bayesian missing data problems," vol. 89.

[44] Pham, D. T., "Stochastic methods for sequential data assimilation in strongly nonlinear systems," Monthly Weather Review, vol. 129, no. 5, pp. 1194{1207, 2001.

[45] Gordon, N. J., Salmond, D. J., and Smith, A. F. M., "Novel approach to nonlinear/non Gaussian Bayesian state estimation," IEE Proceedings, vol. 2, no. 140, pp. 107-113, 1993.

[46] G. Karlsson and M. Vetterli, “Three-dimensional subband coding of video,” in Proc. Int. Conf. Acoustics, Speech, Signal Processing, vol. 2, Apr. 1988.

[47] Zhou, Q.; Aggarwal, J. K.; “Tracking and classifying moving objects from video”, Proc of 2nd IEEE Intl Workshop on Performance Evaluation of Tracking and Surveillance (PETS’2001), Kauai, Hawaii, USA (December 2001).

[48] Collins, R. T.;Lipton, A. J.; Kanade, T.; Fujiyoshi, H.; Duggins, D.; Tsin YTolliver,D.; Enomoto, N. and Hasegawa, O.; “A system for video surveillance and monitoring”, Technical Report CMU-RI-TR -00-12, CMU, 2000.

ANNEXE

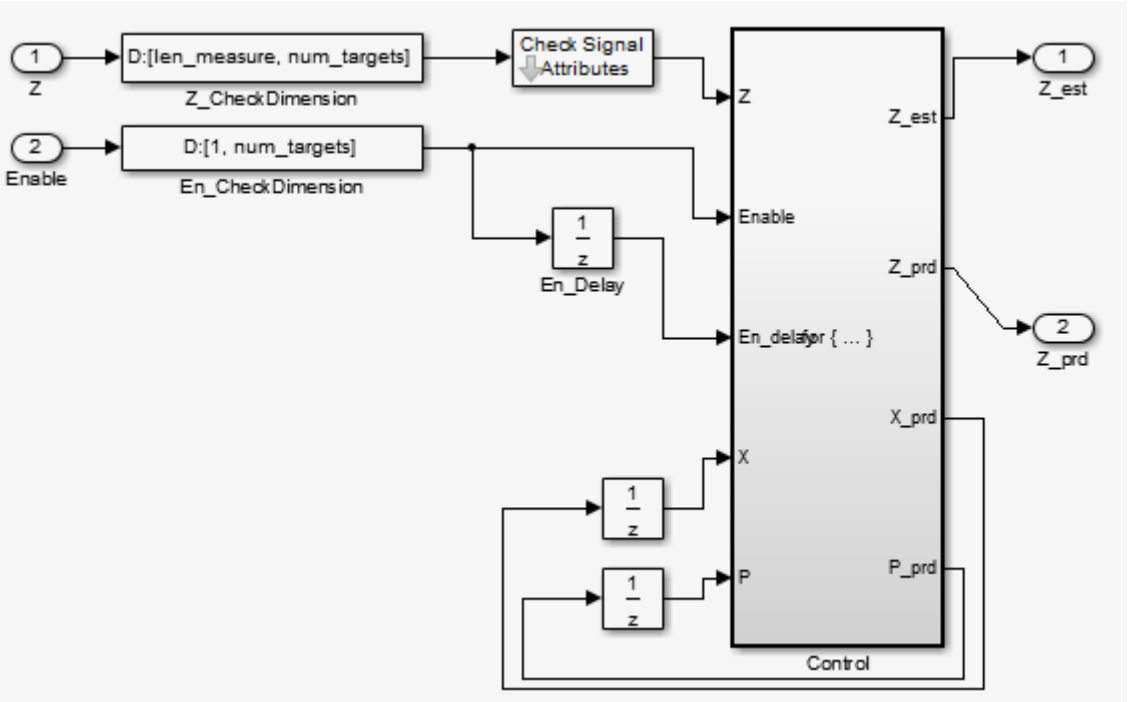


Figure.1 Modèle Simulink de filtre Kalman.

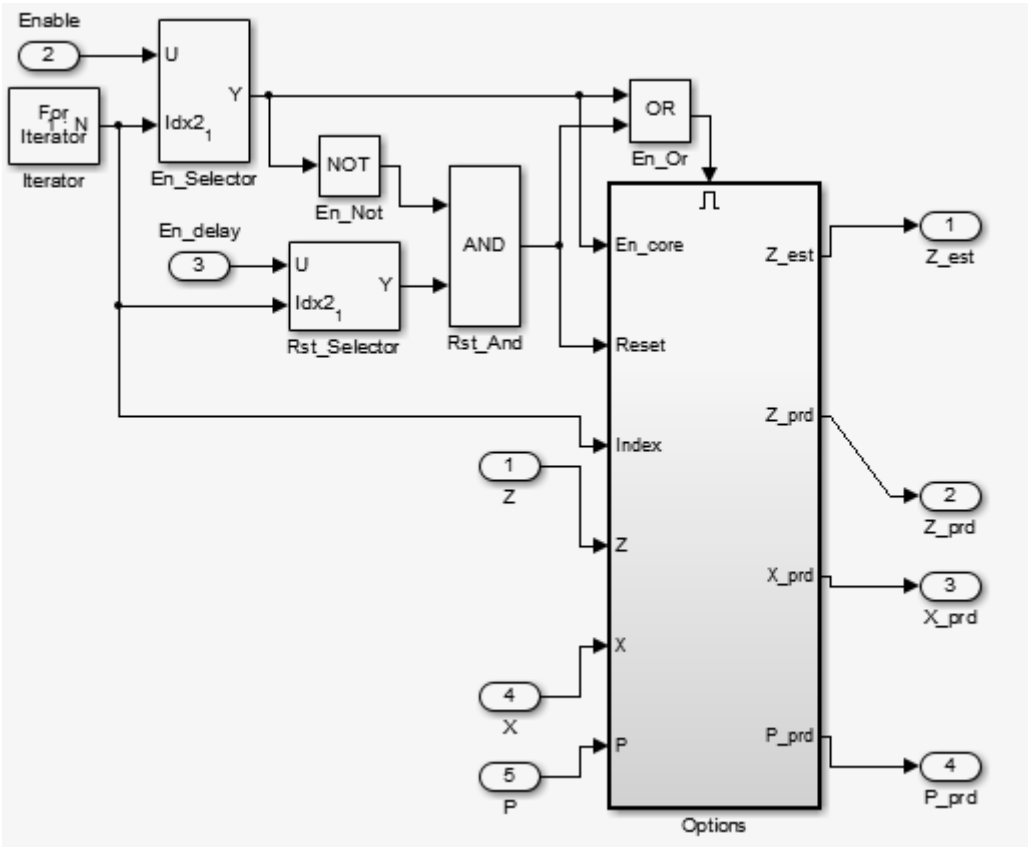


Figure.2 Modèle Simulink pour le bloc du contrôle de filtre Kalman.

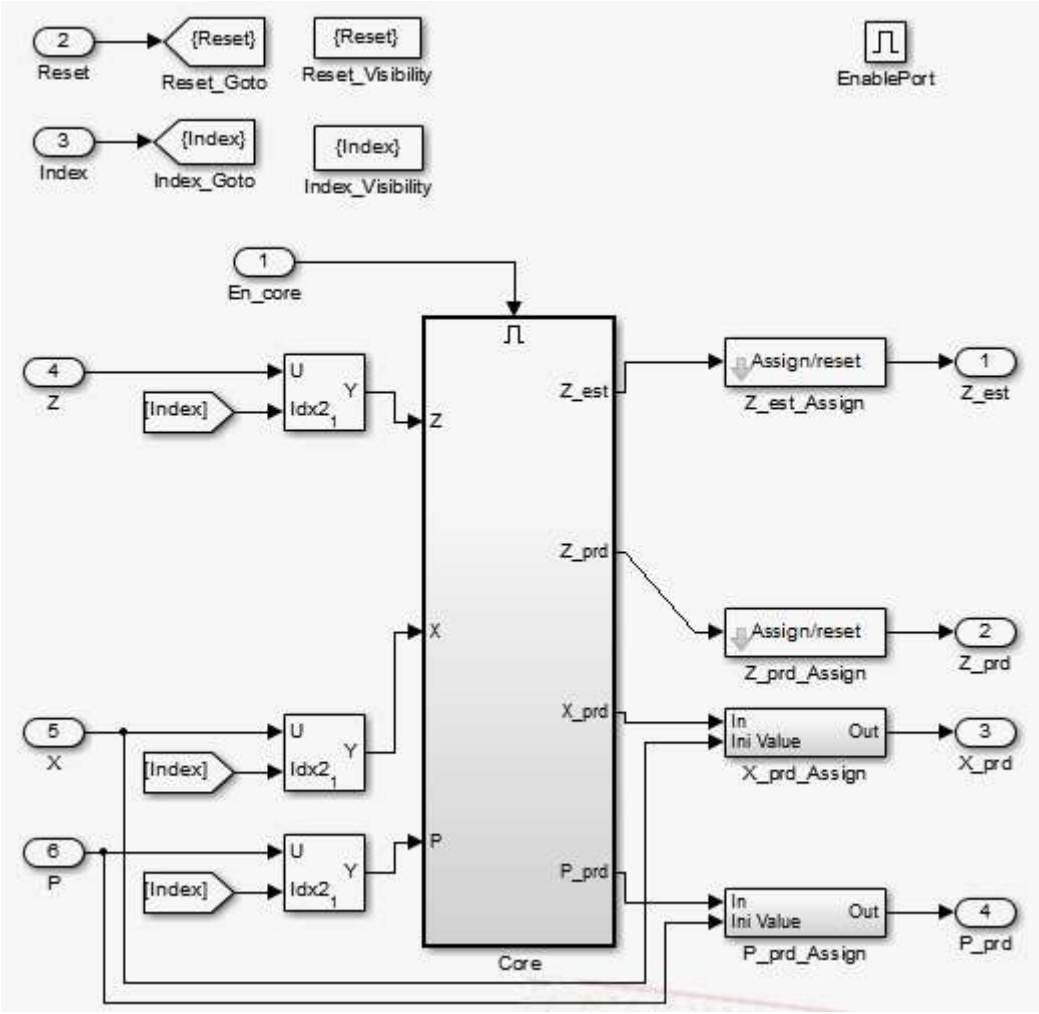


Figure.3 Modèle Simulink pour le bloc des options de filtre Kalman.

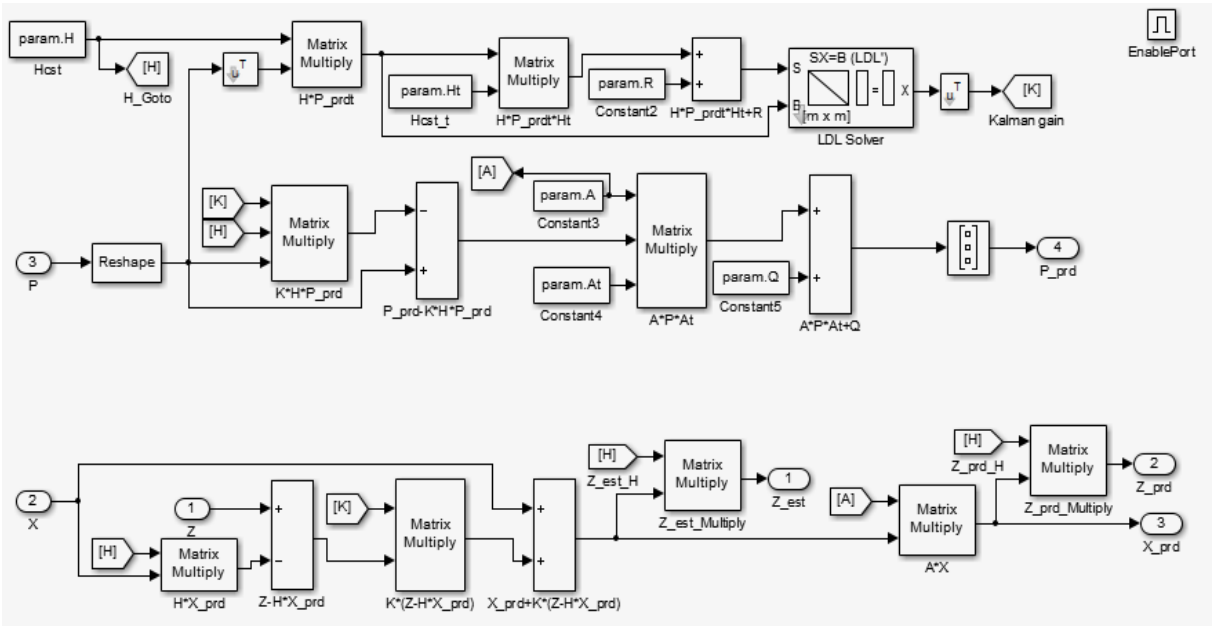


Figure.4 Modèle Simulink pour le bloc du noyau de filtre Kalman.

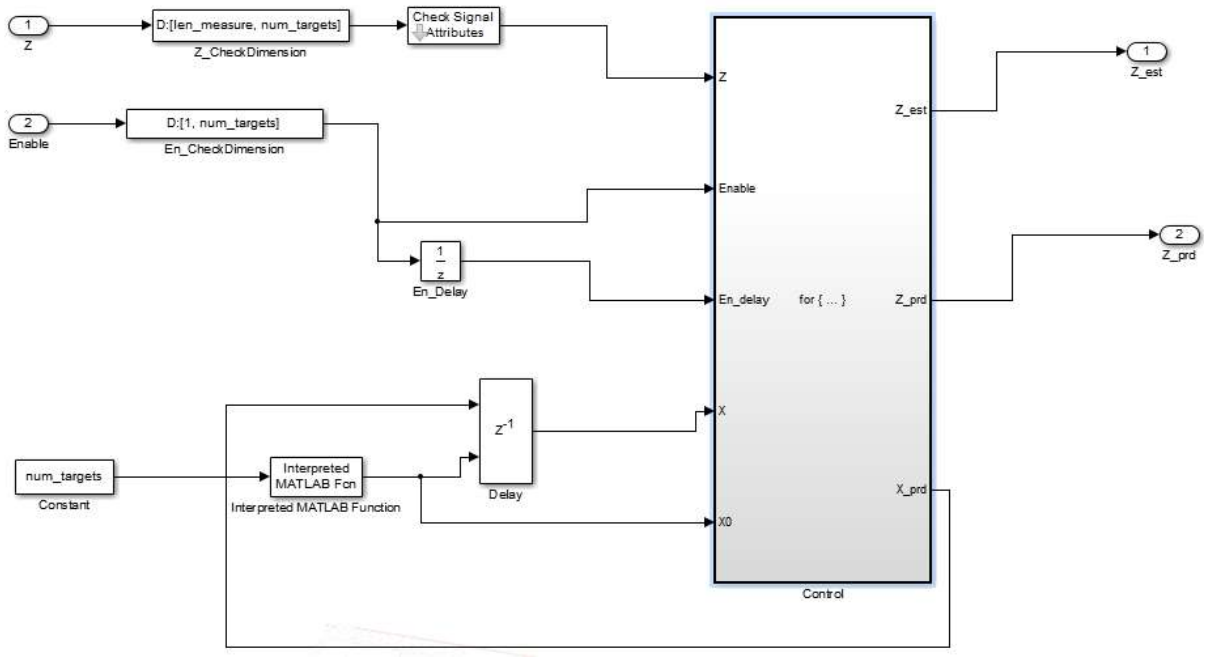


Figure.5 Modèle Simulink de filtre particulaire.

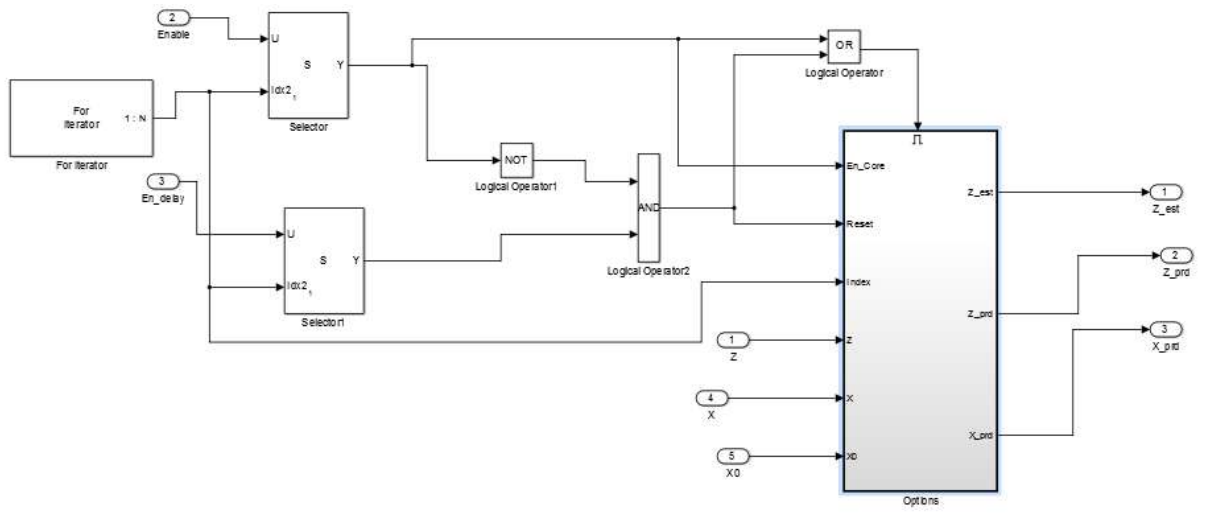


Figure.6 Modèle Simulink pour le bloc du contrôle de filtre particulaire.

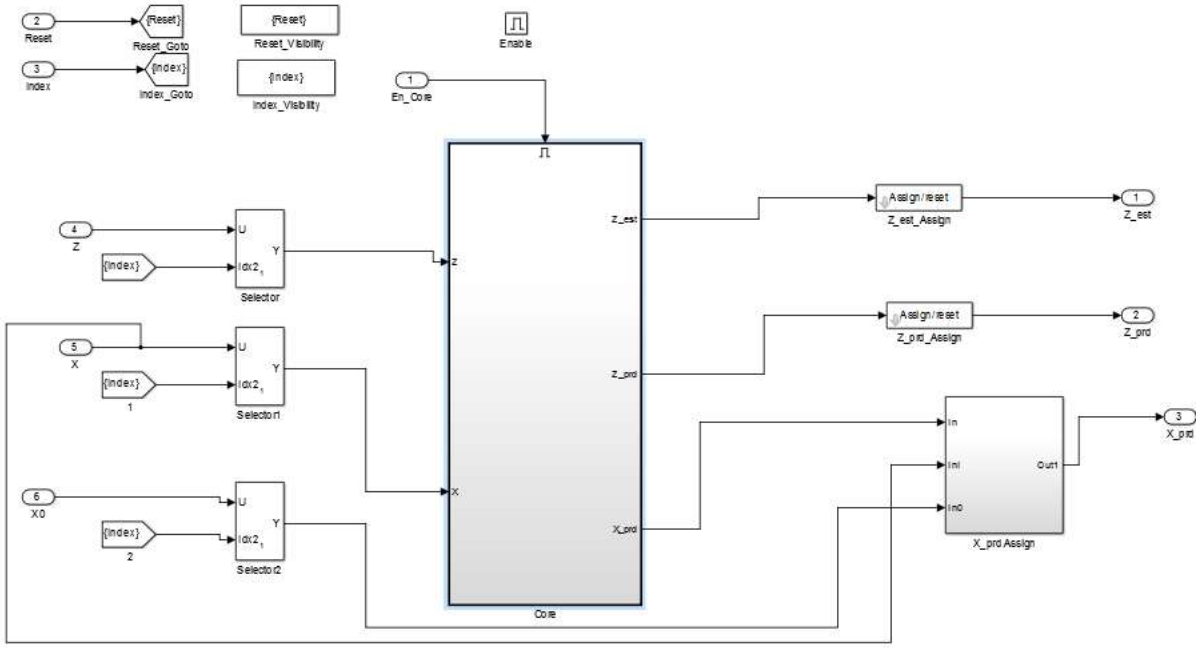


Figure.7 Modèle Simulink pour le bloc des options de filtre particulaire.

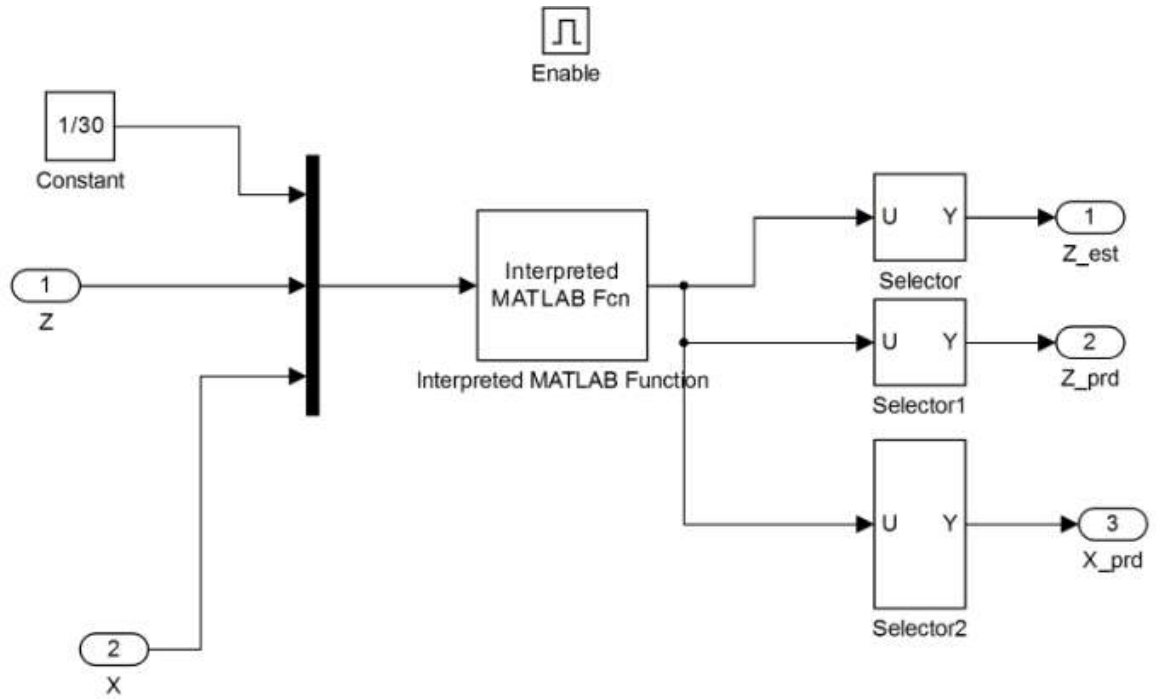


Figure.8 Modèle Simulink pour le bloc du noyau de filtre particulaire.