

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

**ECOLE NATIONALE POLYTECHNIQUE**  
*Département de Génie Electrique*



**Projet de fin d'études**

**En vue de l'obtention du Diplôme  
d'Ingénieur d'Etat en Electronique**

**THEME :**

*Etude, simulation de la GMM adaptative  
pour la detection de mouvement*

***Présenté par :***

Mr. Y. HAMMOUCHE

Mr. B. YADOUGHI

***Proposé par :***

Mr. C. LARBES

Mme. S. AIT DAOUD

Promotion : Juin 2012

Ecole Nationale Polytechnique  
10, Avenue Hacén Badi, El-Harrach, Alger.

## *Remerciements*

---

*Nous exprimons notre reconnaissance à Monsieur C. LARBES d'avoir joué pleinement son rôle de promoteur en étant à nos côtés tout au long de l'étude de notre projet. Ses conseils et orientations nous ont guidés jusqu'à l'aboutissement de ce travail.*

*Nous remercions notre promotrice madame S. AIT DAOUD pour ses remarques pertinentes qui ont apporté une amélioration certaine à notre travail. Nous la remercions également d'avoir examiné pleinement notre travail et proposé des modifications constructives.*

*Nous tenons à remercier Monsieur L. ABDELOUAL, MA-A à l'Ecole Nationale Polytechnique d'Alger, d'avoir accepté de présider le jury.*

*Nos remerciements vont aussi à Monsieur M. O. TAGHI, MA-A à l'Ecole Nationale Polytechnique, d'avoir bien voulu examiner notre travail.*

*Que tous ceux et toutes celles qui ont contribué de près ou de loin à l'aboutissement de notre travail trouvent ici nos sincères remerciements et l'expression de notre reconnaissance.*

## *Dédicaces*

---

*Je dédie ce modeste travail à ma mère, mon père, mes frères et ma sœur  
qui m'ont soutenue sans relâche, dans toutes les circonstances, tout au long de  
mon parcours d'études.*

*Je dédie ce travail à toute ma famille pour leur soutien.*

*Je dédie ce travail à ma chère Amira et à tous mes ami(e)s pour leur  
soutien tout au long de mon parcours.*

*Enfin je dédie ce travail à ma chère et très spéciale promotion.*

*Y. HAMMOUCHE.*

*Je dédie ce mémoire, aux personnes qui me sont les plus chères :*

*A mes parents qui m'ont énormément soutenu dans les moments  
Les plus difficiles, partagé mes joies et mes peines, qui se sont  
Toujours sacrifiés à mes dépends.*

*A mes frères et à mes sœurs.*

*A toute ma famille.*

*A mes amis et collègues.*

*B. YADOUGHI.*

## Résumé

---

Dans ce mémoire, nous vous présenterons deux principales méthodes basées sur les GMMs : la première est celle du mélange de Gaussiennes multiples réalisée par Stauffer & Grimson et la deuxième est nommée Fuzzy GMMs qui est une amélioration de la précédente. Ces deux méthodes sont implémentées en MatLab et permettent d'obtenir de bons résultats par rapport aux autres approches classiques non statistiques. Pour cela, nous avons calculé deux taux qui sont : le FP (taux de fausses détections des objets mobiles) et FN (taux de fausses détections des zones statiques).

**Mots clés :** Vidéo - Mouvement - GMMs - FGMMs - Détection - Télésurveillance - Moyenne - Variance - MatLab.

## Abstract

---

In this memory, we will present two main methods based on GMMs: the first is the mixture of multiple Gaussian conducted by Stauffer & Grimson and the second is named Fuzzy GMMs which is an improvement from the previous one. Both methods are implemented in MatLab and can achieve good results compared to other conventional non-statistical approaches. For this, we calculated two rates are: the FP (false detection rate of mobile object) and FN (false detection rate of static areas).

**Keywords :** Video - Motion - GMMs - FGMMs - Detection - Remote Monitoring - Average - Variance - MatLab.

## ملخص

---

إن كشف الحركة مجال واسع لاختبار المشهد، يستخدم أساسا في التعرف على السلوك في (المطار، الملاعب،... الخ) و ذلك بفضل أجهزة المراقبة المرئية. الهدف من دراستنا هذه هو توظيف طريقة الخليط "الغوسيان" لاكتشاف الحركة في مجموعة صور متتالية "فيديو". تمكننا هذه الطريقة من التعرف على جميع أنواع الحركات التي يشملها المشهد: بطيئة كانت أم سريعة. كما أن هذه الطريقة تتكيف مع خلفيات الصورة، و لقد استعملت لأول مرة من طرف الباحثين "ستوفر و غريمسون" عام 1999 و طورت من طرف باحثين معاصرين آخرين. أساسها يقوم على ضبط و تصنيف "البيكسال" سواء

للمات البحث : فيديو- الحركة - GMMs - FGMMs - الكشف - المراقبة عن بعد - معدل - الفارق

.MatLab



# Table des matières

---

<b>Introduction générale</b> .....	(1)
------------------------------------	-----

## **Chapitre I** : Etat de l'art de la détection de mouvement

---

<b>I.1. Introduction</b> .....	(3)
<b>I.2. Vision par ordinateur</b> .....	(3)
<b>I.3. Analyse du mouvement</b> .....	(5)
I.3.1. Le mouvement.....	(6)
I.3.2. Problèmes liés au mouvement.....	(8)
<b>I.4. Détection de mouvement</b> .....	(8)
I.4.1. Domaines d'application de la détection de mouvement.....	(8)
I.4.2. Différentes méthodes de détection 2D et 3D.....	(11)
I.4.2.1. Introduction.....	(11)
I.4.2.2. Méthodes différentielles.....	(12)
I.4.2.2.1. Introduction.....	(12)
I.4.2.2.2. Méthode de Lucas et Kanade.....	(13)
I.4.2.2.3. Méthode de Horn et Schunck.....	(14)
I.4.2.3. Méthodes fréquentielles.....	(16)
I.4.2.3.1. Introduction.....	(16)
I.4.2.3.2. Corrélacion de phase.....	(17)
I.4.2.3.3. Transformée de Fourier-Mellin.....	(19)
I.4.2.4. Méthodes d'analyse en composantes principales (ACP).....	(20)
I.4.2.5. Méthodes morphologiques.....	(25)
I.4.2.5.1. Introduction.....	(25)
I.4.2.5.2. Moyenne récursive.....	(27)
I.4.2.5.3. Morphologie oubliieuse temporelle.....	(28)
I.4.2.5.4. Estimation - .....	(31)
<b>I.5. Conclusion</b> .....	(34)

## **Chapitre II** : Détection de mouvement par méthode de mélange de Gaussiennes

---

<b>II.1. Introduction</b> .....	(35)
---------------------------------	------

<b>II.2. Présentation du problème</b> .....	(35)
<b>II.3. Détection de mouvement par méthode de mélange de Gaussiennes</b> .....	(36)
II.3.1. Introduction.....	(36)
II.3.2. Principe du mélange de Gaussiennes.....	(37)
II.3.3. Etude de la méthode de détection par mélange de Gaussiennes.....	(39)
II.3.4. Propriétés des paramètres du mélange de Gaussiennes.....	(40)
II.3.4.1. Nombre de composantes Gaussiennes $K$ .....	(40)
II.3.4.2. Matrice de covariance $\Sigma_{i,t}$ .....	(42)
II.3.4.3. Poids $w_{i,t}$ .....	(42)
II.3.4.4. Moyenne $\mu_{i,t}$ .....	(42)
II.3.5. Processus de détection.....	(42)
II.3.5.1. Initialisation des paramètres.....	(43)
II.3.5.2. Classification des pixels.....	(44)
II.3.5.3. Mise à jour des paramètres.....	(46)
II.3.6. Discussion, et contraintes.....	(48)
II.3.6.1. Avantages.....	(48)
II.3.6.2. Inconvénients.....	(49)
<b>II.4. Améliorations apportées à la méthode de mélange de Gaussiennes</b> .....	(50)
II.4.1. Amélioration de Lee.....	(50)
II.4.2. Méthode T2-FGMM.....	(52)
II.4.2.1. Principe.....	(52)
II.4.2.2. Détection d'objets en mouvement, mise à jour des paramètres.....	(55)
<b>II.5. Conclusion</b> .....	(57)

## **Chapitre III : Réalisation et mise en œuvre, test des résultats**

---

<b>III.1. Introduction</b> .....	(58)
<b>III.2. Réalisation et mise en œuvre</b> .....	(58)
III.2.1. Environnement de travail et de développement.....	(58)
III.2.2. Architecture générale de détection.....	(60)
III.2.3. Mise en œuvre.....	(62)
III.2.3.1. Acquisition vidéo.....	(62)
III.2.3.2. Segmentation et prétraitement.....	(63)
III.2.3.3. Détection.....	(63)
III.2.3.3.1. Présentation, déclaration des données.....	(63)
III.2.3.3.2. Algorithme GMM standard.....	(64)

III.2.3.3.3. Algorithme GMM améliorée (FGMM-UM).....	(70)
III.2.3.4. Post-traitement.....	(71)
III.2.3.5. Affichage des résultats.....	(72)
<b>III.3. Tests et résultats.....</b>	<b>(72)</b>
III.3.1. Séquences utilisées.....	(72)
III.3.2. Résultats de détection.....	(73)
III.3.2.1. Résultats de détection pour une scène simple.....	(73)
III.3.2.2. Résultats de détection pour une scène complexe.....	(74)
III.3.3. Evaluation des résultats.....	(77)
III.3.3.1. Procédure à suivre.....	(77)
III.3.3.2. Résultats de l'évaluation.....	(78)
III.3.3.2.1. Résultats sans filtrage morphologique.....	(78)
III.3.3.2.2. Résultats avec filtrage morphologique.....	(79)
III.3.4. Influence des différents paramètres sur les résultats de détection.....	(81)
III.3.4.1. Influence de ALPHA sur les résultats de détection.....	(82)
III.3.4.2. Influence de K sur les résultats de détection.....	(84)
III.3.4.3. Influence du seuil T sur les résultats de détection.....	(86)
III.3.5. Estimation du fond statique.....	(88)
<b>III.4. Conclusion.....</b>	<b>(90)</b>
<b>Conclusion générale.....</b>	<b>(91)</b>

# Liste des figures

---

## Chapitre I

---

<b>Figure .I.1</b> : Processus de traitement et d'analyse d'un système visuel artificiel.....	(5)
<b>Figure .I.2</b> : Projection du mouvement réel 3D dans le plan d'image à 2D.....	(6)
<b>Figure .I.3</b> : Mouvement réel vs mouvement apparent.....	(7)
<b>Figure .I.4</b> : Congestion routière.....	(9)
<b>Figure .I.5</b> : Détection de comportements dangereux liés aux bagarres et agressions dans les métros.....	(10)
<b>Figure .I.6</b> : Image montrant l'avantage de la détection de collision pour protéger l'homme des robots dangereux.....	(10)
<b>Figure .I.7</b> : Application de la détection de mouvement pour l'évitement d'obstacles.....	(11)
<b>Figure .I.8</b> : Droite de contrainte.....	(13)
<b>Figure .I.9</b> : Méthode différentielle de Lucas et Kanade.....	(14)
<b>Figure .I.10</b> : Méthode de Horn et Schunck pour le calcul du flot optique.....	(15)
<b>Figure .I.11</b> : Représentation fréquentielle d'une image.....	(16)
<b>Figure .I.12</b> : Méthode de corrélation de phase.....	(18)
<b>Figure .I.13</b> : Changement de repère orthogonal.....	(21)
<b>Figure .I.14</b> : Comparaison des données (a) avec (b) sans valeurs centrées et réduites.....	(22)
<b>Figure .I.15</b> : La première et la dernière trame d'une séquence de 10 trames.....	(23)
<b>Figure .I.16</b> : Variance expliqué par les axes principaux.....	(23)
<b>Figure .I.17</b> : Projection de la matrice Y sur les axes principaux ordonnées selon le rang du facteur correspondant.....	(24)
<b>Figure .I.18</b> : (a) projection de Y dans le premier axe principale, (b) segmentation de l'arrière-plan obtenue à partir de (a).....	(24)
<b>Figure .I.19</b> : L'érosion temporelle.....	(26)
<b>Figure .I.20</b> : La dilatation temporelle.....	(26)

<b>Figure .I.21</b> : Le gradient temporel.....	(27)
<b>Figure .I.22</b> : Résultat de la détection par moyenne réursive ( $\alpha = \frac{1}{8}$ ).....	(28)
<b>Figure .I.23</b> : Filtrage morphologique oublieux.....	(30)
<b>Figure .I.24</b> : Application des opérateurs morphologies oublieux (en bas), comparés aux opérateurs morphologies classiques (en haut) avec ( $\alpha = \frac{1}{9}$ ).....	(30)
<b>Figure .I.25</b> : Résultat de la méthode d'estimation - .....	(32)
<b>Figure .I.26</b> : L'estimateur - en fonction du type de zone où se trouve le pixel.....	(33)

## **Chapitre II**

---

<b>Figure .II.1</b> : Mélange de gaussiennes.....	(37)
<b>Figure .II.2</b> : Comportement des pixels représenté par un mélange de distributions Gaussiennes.....	(38)
<b>Figure .II.3</b> : Influence du paramètre K sur les résultats de détection dans une scène simple.....	(40)
<b>Figure .II.4</b> : Influence du paramètre K sur les résultats de détection dans une scène complexe.....	(41)
<b>Figure .II.5</b> : fonctionnement de l'algorithme EM.....	(43)
<b>Figure .II.6</b> : Détection de pixels d'avant-plan par la méthode de mélange de Gaussiennes.....	(46)
<b>Figure .II.7</b> : Influence du paramètre sur les résultats de détection.....	(47)
<b>Figure .II.8</b> : Comparaison de vitesse de convergence entre la méthode standard et améliorée.....	(50)
<b>Figure .II.9</b> : La Gaussienne (primary MF) avec incertaine moyenne.....	(53)
<b>Figure .II.10</b> : La Gaussienne (primary MF) avec incertaine déviation standard.....	(54)
<b>Figure .II.11</b> : Comparaison des performances entre la technique standard (GMM Stauffer et Grimson) et améliorée (T2-FGMM- UV et UM).....	(57)

## **Chapitre III**

---

<b>Figure .III.1</b> : Le logiciel scientifique de calcule MatLab.....	(60)
--	------

<b>Figure .III.2</b> : L'architecture générale de détection utilisée.....	(61)
<b>Figure .III.3</b> : Résultats de détection obtenus en utilisant la méthode de détection de mouvement par mélange de Gaussiennes standard.....	(73)
<b>Figure .III.4</b> : Résultats de détection obtenus en utilisant la méthode de détection de mouvement par mélange de Gaussiennes améliorée.....	(74)
<b>Figure .III.5</b> : Résultats de détection obtenus en utilisant la méthode de détection de mouvement par mélange de Gaussiennes standard.....	(74)
<b>Figure .III.6</b> : Résultats de détection obtenus en utilisant la méthode de détection de mouvement par mélange de Gaussiennes améliorée.....	(75)
<b>Figure .III.7</b> : Création d'une image de référence pour l'évaluation des résultats de détection.....	(77)
<b>Figure .III.8</b> : Résultats de l'évaluation sans filtrage morphologique.....	(78)
<b>Figure .III.9</b> : Résultats de l'évaluation avec filtrage morphologique.....	(79)
<b>Figure .III.10</b> : Evaluation des résultats obtenue avec et sans filtrage par les deux méthodes de détection (standard et améliorée).....	(81)

# Liste des tableaux

---

## Chapitre III

---

**Tableau .III.1 :** Influence de ALPHA sur les résultats de détection pour une scène simple.....(82)

**Tableau .III.2 :** Influence de ALPHA sur les résultats de détection pour une scène complexe.....(83)

**Tableau .III.3 :** Influence de K sur les résultats de détection pour une scène simple.....(84)

**Tableau .III.4 :** Influence de K sur les résultats de détection pour une scène complexe.....(85)

**Tableau .III.5 :** Influence du seuil T sur les résultats de détection pour une scène simple.....(86)

**Tableau .III.6 :** Influence du seuil T sur les résultats de détection pour une scène complexe.....(87)

**Tableau .III.7 :** Estimation du fond statique à partir d'une séquence complexe.....(88)

# *Introduction générale*

---

La vision est sans doute notre sens le plus développé et, du point de vue de l'évolution, le plus utile. Nous nous servons quotidiennement de notre système de vision pour : nous déplacer, estimer les distances, identifier les personnes et les objets qui nous entourent, ...etc. Nous le faisons sans aucune difficulté, sans même y prêter l'attention, bien que les mécanismes mis en jeu soient assez complexes. Pour les comprendre, la communauté scientifique a cherché à reproduire ces mécanismes, en développant des systèmes de vision artificielle par ordinateur, capables d'interpréter et de voir des scènes. Autrement dit « faire voir les ordinateurs ».

Avec la généralisation de l'utilisation d'images numériques, l'analyse du mouvement dans les séquences vidéo s'est révélée être un outil indispensable pour diverses applications telles la vidéo surveillance, l'imagerie médicale, la robotique...etc. L'analyse du mouvement dans une séquence d'images est un vaste sujet qui regroupe plusieurs problématiques. On peut notamment citer l'estimation du champ de mouvement, la détection de mouvement, et la segmentation de chaque région d'une image ayant un mouvement distinct des autres.

La problématique qui se pose dans ce manuscrit est la détection de mouvement dans une séquence vidéo, qui fait appel au traitement d'images. Le traitement d'images est un domaine basé sur plusieurs disciplines : les mathématiques, les probabilités, l'informatique, ...

La détection de mouvement s'est avérée comme étant un domaine très vaste du traitement et d'analyse de la scène (ou d'images). De nos jours, un grand nombre de méthodes toutes pourvues d'inconvénients ont été développées. La méthode de « mélange de Gaussiennes » est l'une des méthodes les plus populaires car, elle peut s'adapter facilement aux scénarios les plus défavorables. Plusieurs variantes améliorées ont été développées pour répondre aux insuffisances de la modélisation standard de Stauffer et Grimson, parmi elles, on distingue celles de Lee et de Zeng.



Le but de notre travail est d'étudier la méthode de détection de mouvement par la GMM ainsi que ses variantes. Parmi elles, nous étudierons en détail, la méthode FGMM.

Ce manuscrit se décompose en trois grands chapitres :

### **Chapitre I** « Etat de l'art de la détection de mouvement »

Nous verrons d'une manière générale dans ce chapitre, les domaines d'application de la détection de mouvement, ainsi que les différents algorithmes de détection, largement utilisés. Il s'agira essentiellement des méthodes différentielles et fréquentielles, ainsi que des méthodes basées sur le fond statique...

### **Chapitre II** « Détection de mouvement par méthode de mélange de Gaussiennes »

Ce chapitre est consacré à l'étude de la méthode de détection de mouvement par mélange de Gaussiennes standard (Gaussian Mixture Model), ainsi qu'à ses différentes variantes.

### **Chapitre III** « Réalisation, tests et résultats »

Dans ce chapitre nous présenterons, les algorithmes de détection de mouvement par mélange de Gaussiennes standard, et amélioré type « Fuzzy Gaussian Mixture Model », ainsi que les résultats de détection tout en comparant les deux méthodes mises en œuvre.

# *Chapitre I*

---

*Etat de l'art de la détection de mouvement*

## **I.1. Introduction :**

Dans les organismes biologiques dotés de système visuel, la perception d'un mouvement dans une scène visuelle est un mécanisme très complexe d'un point de vue de sa compréhension. Pour comprendre ces mécanismes mis en jeu, des scientifiques ont cherché à les reproduire. Il en résulte le développement de systèmes de vision artificielle par ordinateur capable d'observer et d'interpréter des scènes visuelles.

La vision par ordinateur est une reproduction du système d'interprétation visuel humain sur une machine. Cette reproduction n'est pas aussi simple, car si l'homme sait naturellement détecter et analyser un mouvement (distinguer et séparer différents objets ou ceux en mouvement ou non), il a aussi la capacité de réagir en cas d'évènement potentiel. De ce fait, le système de vision artificielle doit fournir des connaissances nécessaires afin de permettre une interprétation non ambiguë.

A ce jour, plusieurs modèles et méthodes d'analyse du mouvement ont été mis au point. Dans ce chapitre, nous ferons un état de l'art sur : la vision par ordinateur tout en abordant la notion du mouvement dans une scène visuelle. Puis, l'application de la détection de mouvement dans différents domaines. Pour finir, les méthodes d'analyse 2D et 3D les plus utilisées dans la détection, tout en donnant les avantages et inconvénients de chacune.

## **I.2. Vision par ordinateur : [18] [19]**

### **Qu'est-ce que la vision ?**

L'espace qui nous entoure est composé d'objets et de structures 3D. Lorsqu'on demande à une personne de décrire ce qu'elle perçoit : objet, obstacle, ... elle n'éprouve aucun mal à décrire son environnement. En réalité, l'information disponible sur la rétine de ses yeux n'est ni plus ni moins qu'un ensemble de points ou pixels qui est tout simplement la projection de la réalité physique. L'objet perçu n'existe pas sur la rétine, ce ne sont que des réponses finales résultant d'un processus d'interprétation qui fait partie intégrante du système de la vision. Il est clair que le système de vision doit fournir des connaissances nécessaires à une interprétation qui est propre à chaque individu.

La vision a suscité l'intérêt de nombreux scientifiques depuis déjà très longtemps, en vue de comprendre l'anatomie et le fonctionnement du système visuel dans son ensemble.

Puis, avec la naissance de machines de calcul puissantes et très sophistiquées, de nouvelles questions se sont posées telles que :

- Est-il possible de construire un modèle de perception visuelle ? ;
- Peut-il être d'une utilité quelconque quant à la vision biologique ? ;

De ce fait, grâce aux théories et modèles élaborés et mis au point, la vision par ordinateur a pu voir le jour.

La vision par ordinateur peut s'énoncer comme étant un processus de traitement et d'analyse d'une scène (*Fig .I.I*). En concordance avec le système visuel humain, le système de vision artificielle se compose de différentes fonctions :

- l'entrée du système de vision constituée d'images, a comme fonction l'acquisition d'une séquence d'images dont l'accès peut être aléatoire ou séquentiel ;
- par la suite, le système effectue à la fois un ensemble de traitements tout en apportant un certain nombre de connaissances :
  - numérisation qui peut être précédée de corrections de bruit ;
  - pré-traitement ou traitement bas-niveau qui permet de transformer l'image au niveau du pixel (filtrage, segmentation, ...) ;
  - traitement moyen niveau qui permet de remplacer l'image 2D en blocs structurés d'information ;
- la sortie est une description de l'entrée en termes de décision (description de la scène), appelé niveau décisionnel ;

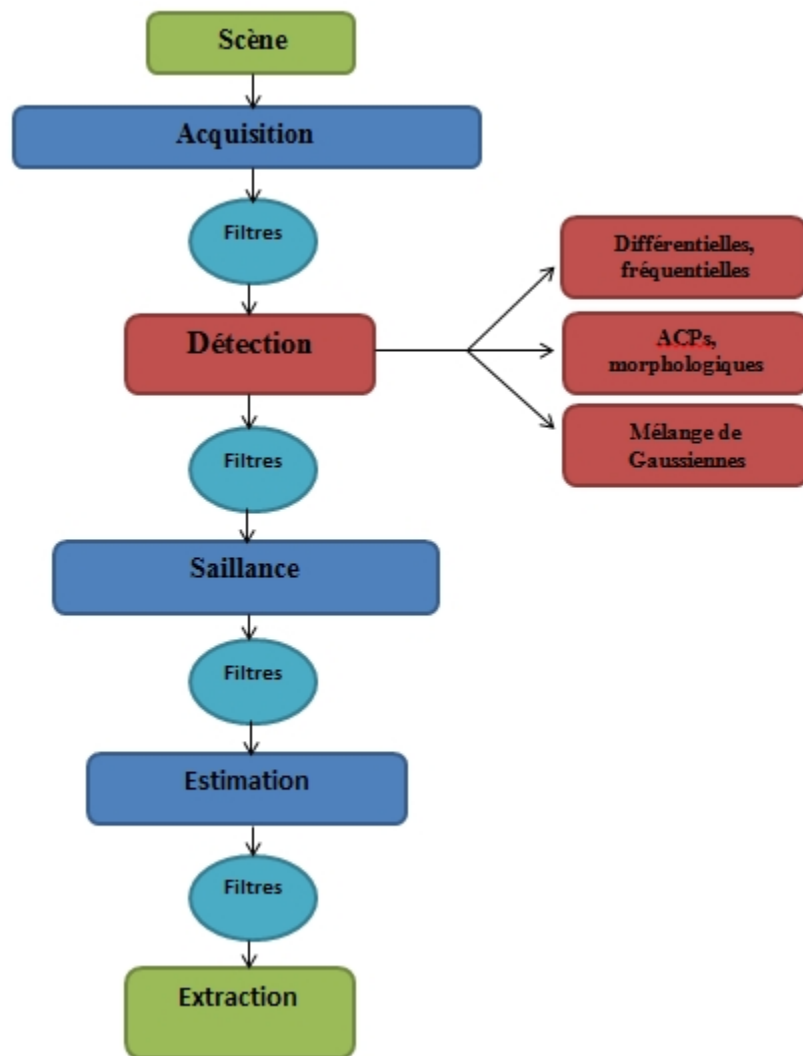


Figure .I.1 : Processus de traitement et d'analyse d'un système visuel artificiel.

Dans ce qui va suivre, on s'intéressera au processus d'analyse de mouvement (de détection de mouvement) dans une scène visuelle tout en citant les différentes méthodes, modèles mis en œuvre jusqu'à présent. Avant tout, faisant un rappel sur les notions d'analyse du mouvement.

### I.3. Analyse du mouvement :

L'analyse du mouvement fait partie intégrante du traitement d'images. C'est un domaine apparu avec l'avènement de nouvelles technologies de l'information. Elle vise à analyser des séquences d'images numériques par un ensemble de méthodes, afin d'en tirer des informations potentielles. Ces informations seront exploitées pour répondre à des problèmes dans des domaines bien précis (sécurité, médecine, robotique, ...). [12]

### I.3.1. Le mouvement :

Une image, constituée d'un ensemble de point (ou pixel) d'intensité lumineuse propre représente la projection de la réalité physique (scène réelle à 3D) dans un plan d'image à 2D. Un mouvement n'est perceptible que grâce aux variations des intensités lumineuses. L'introduction d'une séquence d'images acquise successivement à des instants séparés ajoute de l'information pour analyser la scène 3D : « le mouvement ». De ce fait, un mouvement observé (mouvement apparent) à partir d'une séquence d'images n'est qu'une projection du mouvement réel dans le plan d'image (Fig .I.2). [12] [2] [16]

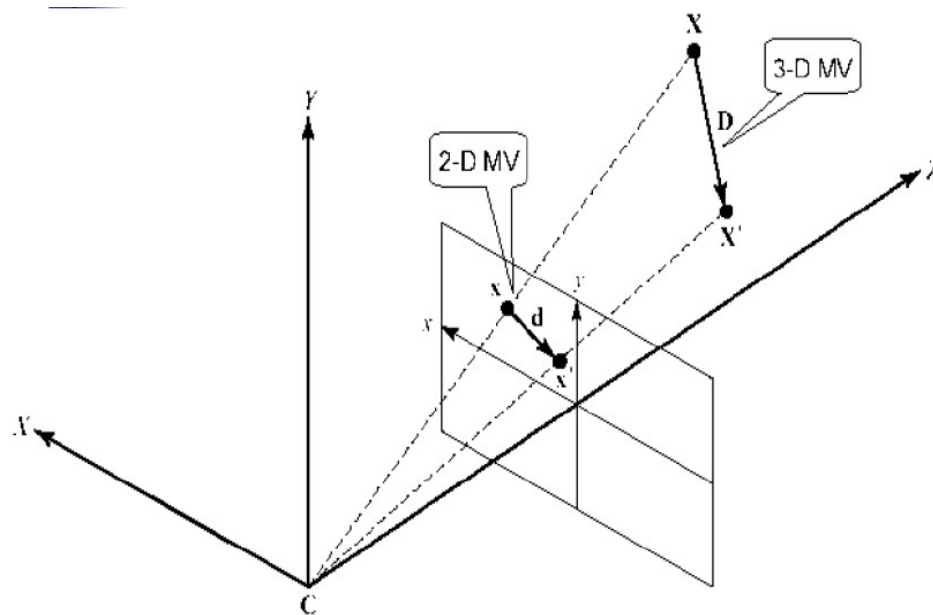


Figure .I.2 : Projection du mouvement réel 3D dans le plan d'image à 2D. [14]

#### *Qu'est-ce qu'un mouvement apparent et un mouvement réel ?*

Le mouvement réel et le mouvement apparent sont souvent différents dans une image :

- **Mouvement réel :**

Le mouvement tel qu'il est perçu par l'œil (ou caméra) dans une scène visuelle 3D est dit « mouvement réel ».

- **Mouvement apparent :**

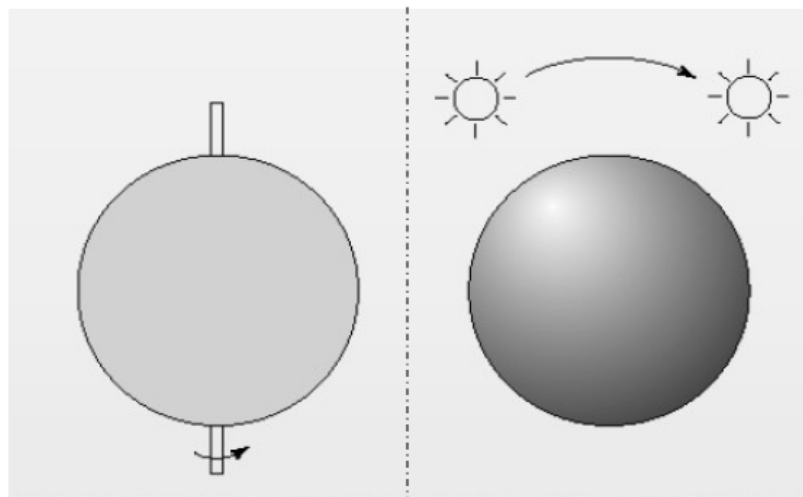
Le mouvement observé à partir des changements de distribution d'intensité lumineuse entre plusieurs images de la séquence est dit apparent (ou observé). On parlera aussi de « flot

optique » (ou champ de vitesse) et de « champ de mouvement apparent » appelé aussi mouvement projeté, représentant la projection du mouvement réel 3D dans le plan d'image 2D. [14][16]

***Quelle est la différence entre un mouvement apparent et un mouvement réel ?***

Le mouvement apparent ne correspond pas toujours au mouvement réel projeté dans le plan de l'image (Fig .I.3). En effet, la présence de variations d'intensité lumineuse non dues au mouvement entraîne un mouvement apparent différent du mouvement réel. Le mouvement apparent est donc une combinaison de deux sources de mouvement :

- le mouvement propre des objets en 3D dans la scène ;
- le mouvement de la caméra (rotation) ;



a. Sphère en rotation

b. Source lumineuse en mouvement

**Figure I.3 :** Mouvement réel vs mouvement apparent. [14]

Dans la *Figure I.3. a*, le mouvement apparent est présent alors que le mouvement réel est inexistant, due à la rotation du capteur (caméra). Dans la *Figure I.3. b*, par contre, l'existence d'une source lumineuse (qui représente un mouvement réel) fait que le mouvement apparent soit présent autant que le mouvement réel. [14] [12]

Sans oublier un troisième mouvement, aussi primordial que les deux autres : le « mouvement estimé ». On parlera alors de champ déplacement et de champ de vitesse qui ne sont que le vecteur déplacement et vecteur vitesse estimés. Rappelons qu'un vecteur déplacement estimé correspond au déplacement d'un point dans le plan d'image (traduit par

une variation « locale » d'intensité lumineuse). Le vecteur vitesse estimé correspond à la variation temporelle du déplacement par unité de temps.

Dans une séquence d'images, il n'est possible que d'estimer le mouvement à partir du champ de mouvement (déplacement ou vitesse) apparent et non le champ réel. Pour estimer le mouvement à partir du champ de mouvement apparent, il faut faire l'hypothèse que l'intensité lumineuse reste constante au cours du mouvement (hypothèse de conservation de l'intensité lumineuse). [16]

### I.3.2. Problèmes liés au mouvement :

Nous pouvons rencontrer différents problèmes liés au mouvement dans une scène visuelle, on en trouve concernant :

- **la détection de mouvement** : y a-t-il eu un mouvement dans la scène ? ;
- **l'estimation du mouvement** : problème de localisation (vitesse et position) ;
- **le suivi du mouvement** : on peut trouver un problème de correspondance entre plusieurs images pour un même mouvement ;
- **la reconnaissance du mouvement** : problème de reconnaissance du scénario correspondant au mouvement ;

## I.4. Détection de mouvement :

Après l'acquisition de données en entrée (la séquence d'images) vient la phase de détection de mouvement. La détection d'un mouvement dans une scène visuelle est parmi les phases de traitement et d'analyse de mouvement les plus importantes. Utilisée dans des champs d'activité et domaines bien précis, elle vise, grâce à des modèles et méthodes, à détecter s'il y a eu un mouvement dans la scène visuelle.

### I.4.1. Domaines d'application de la détection de mouvement :

La détection de mouvement débouche sur de nombreuses applications aussi vastes que variées. On en trouve dans celles liées à la sécurité, dont l'indexation d'images, la télésurveillance (détection d'intrusion, surveillance du trafic routier, ...). La robotique, dont la détection et l'évitement d'obstacles, ... Et aussi des applications liées à la médecine, dont l'imagerie moderne (IRM, scanners, ...).



- **La sécurité :**

Depuis quelques années, la sécurité devient un domaine d'intérêt majeur. Grâce à ces intérêts et aux avancées technologiques rapides, des applications ont été développées, mises au point et basées sur l'analyse vidéo de façon automatique, centrées sur la détection de situations anormales dans diverses activités, telles : [12]

- la surveillance du trafic routier, pour la détection de congestion (*Fig .I.4*), détection d'accidents ;



**Figure .I.4 :** Congestion routière.

Grâce à des caméras spécialisées et des capteurs de proximités placés à des endroits sensibles, on pourra à la fois surveiller et évaluer pour une meilleure fluidité du trafic routier (pour exemple, permettre un meilleur déploiement des secours en cas d'accidents graves). .

- la surveillance des lieux sensibles, et à risque majeur (centrales nucléaires, chimiques, ...), et aussi la surveillance des biens étatiques, publiques et privés (banques, habitations, ...);
- la sécurité des personnes comme : la délinquance (vols, casses, ...), détection de comportements dangereux (bagarres, agressions, ...) (*fig .I.5*), la détection des colis suspects, ...

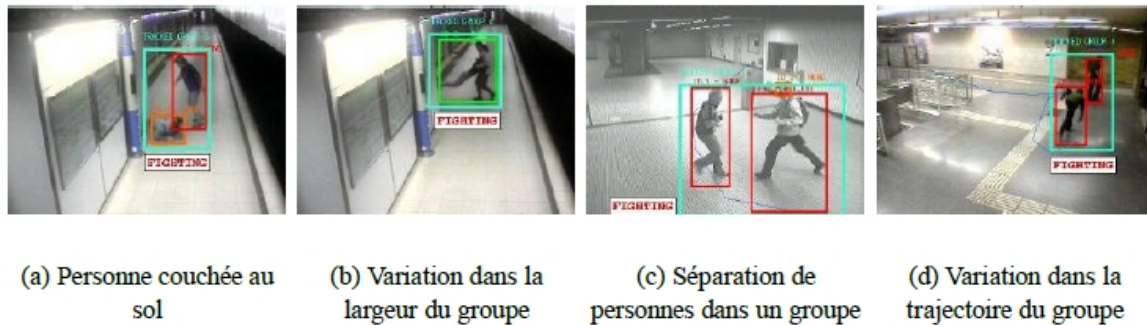


Figure I.5 : Détection de comportements dangereux liés aux bagarres et agressions dans les métros.

[16]

Chaque image de la Figure I.5 représente une scène de bagarre. Pour la reconnaissance de ces comportements dangereux, il faudra une interprétation de haut-niveau de la scène (c'est-à-dire une détection de mouvement seule ne suffit pas à reconnaître une action de bagarre ou d'agression).

L'objectif de ces applications n'est pas nécessairement lié à la sécurité, mais aussi à d'autres domaines tels que « le marketing ». On peut de plus en plus par exemple analyser le parcours type d'un client, pour optimiser la disposition de certains rayons de supermarché, ou encore compter le nombre de personnes qui fréquentent quotidiennement un centre commercial. [16]

- **La robotique :**

L'avancée technologique dans le domaine de la robotique a permis de détecter et d'éviter des obstacles, de prévenir d'éventuelles collisions de conséquence parfois dramatiques comme le montre les deux figures ci-dessous :



Figure I.6 : Image montrant l'avantage de la détection de collision pour protéger l'homme des robots dangereux. [21]

Comme le montre l'image [21], on peut observer l'évitement d'une agression humaine par un robot armé d'un couteau tranchant.



**Figure .I.7 :** Application de la détection de mouvement pour l'évitement d'obstacles. [17]

D'autres applications de détection touchent aussi tout ce qui concerne : la navigation automatique, la poursuite de cible et le guidage de missiles, ...

### - *La médecine :*

Dans le domaine médical, la détection de mouvement est utilisée dans l'imagerie médicale, scanners, ...

## **I.4.2. Différentes méthodes de détection 2D et 3D :**

### **I.4.2.1. Introduction :**

Toutes les méthodes de détection du mouvement ont deux principales contraintes : l'éclairage de la scène et le mouvement du capteur. En effet, la situation est différente selon que le capteur soit fixe, qu'il bouge lentement avec un mouvement compensé. De même, un éclairage constant, des variations d'éclairage à basse fréquence ou des changements soudains d'intensité lumineuse seront gérés différemment par un système de détection de mouvement. A ces difficultés s'ajoutent le bruit du capteur (bruit d'acquisition et de numérisation) et la gestion des zones homogènes (lorsque la différence de luminance entre deux instants est inférieure à un certain seuil). [10]

Pour cela, les méthodes de détection de mouvement doivent tenir compte des contraintes suivantes [12] :

- adaptation temporelle en fonctionnant sans intervention humaine pendant un long moment, et prise en compte des changements graduels ou soudains ;
- élimination des mouvements inintéressants tels que le bruit de la caméra et feuilles des arbres agitées par le vent, et de présenter une certaine robustesse à de petits mouvements du capteur ;
- basse consommation en terme de temps de calcul/capacité mémoire ;

Parmi les principales méthodes, on peut citer :

- les méthodes différentielles ;
- les méthodes fréquentielles ;
- les méthodes basées sur le fond statique (ACP, morphologique et GMM's) ;

#### I.4.2.2. Méthodes différentielles :

##### I.4.2.2.1. Introduction :

Les méthodes différentielles sont des méthodes basées sur les gradients spatiaux et temporels d'intensité lumineuse. Ces gradients sont approchés dans le cas discret par des différences finies. [2]

Elles sont fondées sur ce que l'on appelle l'équation de contrainte du mouvement apparent (ECMA), ou équation du fond optique. Lors d'une séquence d'images, si l'on s'appuie seulement sur l'intensité des pixels, le mouvement perçu est appelé champ de mouvement apparent ou flot optique. [2]

Soit une image  $I(x, y, t)$ , on utilisant le développement de Taylor de  $\frac{\delta I}{\delta t}$  (d'ordre un) :

$$\begin{aligned} I(x, y, t) &= I(x + V_x \Delta t, y + V_y \Delta t, t + \Delta t) \\ &= I(x, y, t) + \Delta t \left( V_x \frac{\delta I}{\delta x} + V_y \frac{\delta I}{\delta y} + \frac{\delta I}{\delta t} \right) \end{aligned}$$

Alors :  $V_x \frac{\delta I}{\delta x} + V_y \frac{\delta I}{\delta y} + \frac{\delta I}{\delta t} = 0$  ce qui se traduit par l'équation de contrainte du mouvement apparent (ECMA) :

$$I_t + V \cdot \Delta I = 0 \quad (I.1)$$

Avec :  $V = (V_x, V_y)$  : le vecteur vitesse,

$I_t$  : la dérivée partielle du signal I par rapport au temps,

$\Delta I$  : la dérivée spatiale [1] [10],

Cette équation repose sur l'hypothèse de conservation d'intensité au cours du mouvement, et un faible déplacement entre deux images consécutives de la séquence.[10]

Les changements temporels dans l'image sont équivalents (au premier ordre) au produit scalaire des changements spatiaux et de la vitesse apparente. [13]

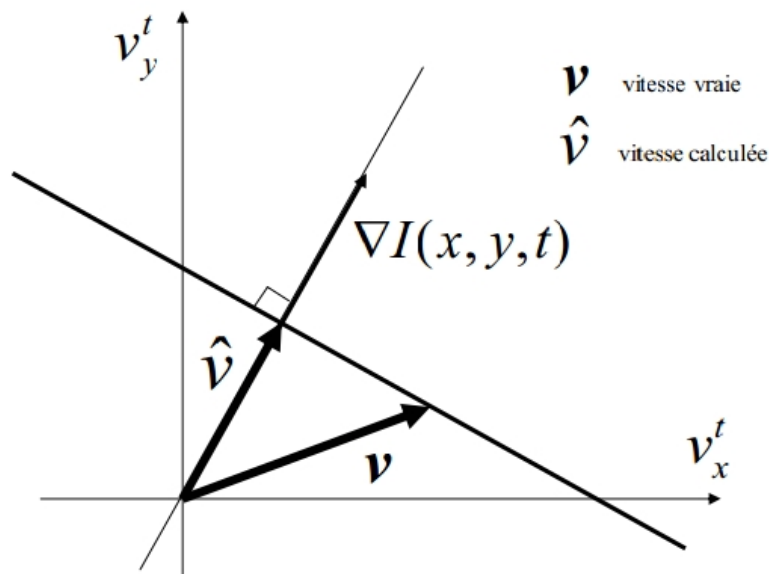


Figure I.8 : Droite de contrainte. [13]

Différents algorithmes très anciens pourtant utilisant ce principe existent. Les méthodes de Lucas et Kanade (1981) et de Horn et Schunck (1981), se différencient par la manière de formuler la contrainte du mouvement apparent. [1]

#### I.4.2.2.2. Méthode de Lucas et Kanade : [1]

L'algorithme de Lucas et Kanade (1981) suppose que le vecteur déplacement  $v$  soit constant sur un voisinage donné. La résolution de l'équation de contrainte de mouvement revient donc à résoudre sur une fenêtre spatiale  $\Omega$  :

$$\mathit{argmin}_v \sum_{x \in \Omega} \mathbf{W}(\mathbf{x}) [\nabla I_{xy}(\mathbf{x}, t) \cdot \mathbf{v} + I_t(\mathbf{x}, t)] \quad (I.2)$$

Où  $W$  est une matrice de pondération de la même taille que permettant de donner plus d'influence aux pixels proches du centre de la fenêtre. En l'exprimant de manière matricielle, cela revient en pratique à résoudre :

$$A^T W^T W A \mathbf{a} = A^T W^T W \mathbf{b} \quad (I.3)$$

Avec pour l'ensemble des N pixels  $x_i$  appartenant à ,

$$A = [\nabla I_{xy}(x_1), \dots, \nabla I_{xy}(x_N)]^T,$$

$$W = \text{diag}[W(x_1), \dots, W(x_N)],$$

$$\mathbf{b} = -[I_t(x_1), \dots, I_t(x_N)]^T$$

Certaines variantes comme celle de Simoncelli et al. (1991) analysent les valeurs propres associées à la matrice  $A^T W^T W A$ . Si les deux valeurs propres sont trop faibles, la région de l'image étudiée au travers de la fenêtre ne représente qu'une zone de basse fréquence et donc le mouvement ne peut être déterminé.

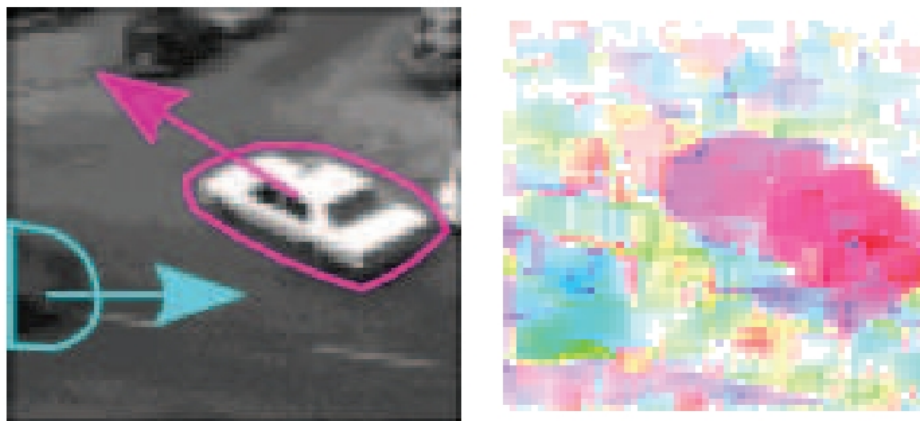


Figure I.9 : Méthode différentielle de Lucas et Kanade. [1]

#### I.4.2.2.3. Méthode de Horn et Schunck : [1]

D'après Horn et Schunck (1981) L'approche ici consiste à combiner la contrainte d'illumination et la contrainte spatiale en définissant une fonction d'énergie qui sera à minimiser. On recherche :

$$\mathop{\text{argmin}}_v \int_D (\nabla I_{xy}(\mathbf{x}, t) \cdot \mathbf{v} + I_t(\mathbf{x}, t))^2 + \alpha^2 (|\nabla \mathbf{u}|_2^2 + |\nabla \mathbf{v}|_2^2) d\mathbf{x} \quad (I.4)$$

Avec D une fenêtre d'analyse, u et v les composantes du vecteur  $\mathbf{v}$  et  $\alpha$  un paramètre de relaxation caractérisant l'influence de la contrainte spatiale qui suppose la continuité du champ de mouvement.

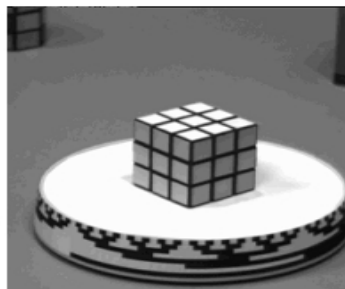
En pratique, les composantes u et v sont déterminées de manière itérative :



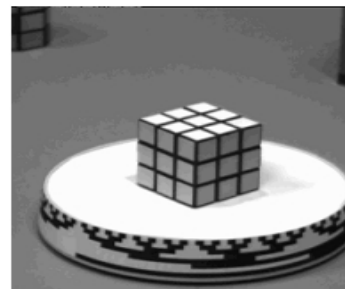
$$\mathbf{u}^{k+1} = \bar{\mathbf{u}}^k - \frac{I_x[I_x\bar{\mathbf{u}}^k + I_y\bar{\mathbf{v}}^k + I_t]}{\alpha^2 + I_x^2 + I_y^2} \quad (I.5)$$

$$\mathbf{v}^{k+1} = \bar{\mathbf{v}}^k - \frac{I_y[I_x\bar{\mathbf{u}}^k + I_y\bar{\mathbf{v}}^k + I_t]}{\alpha^2 + I_x^2 + I_y^2}$$

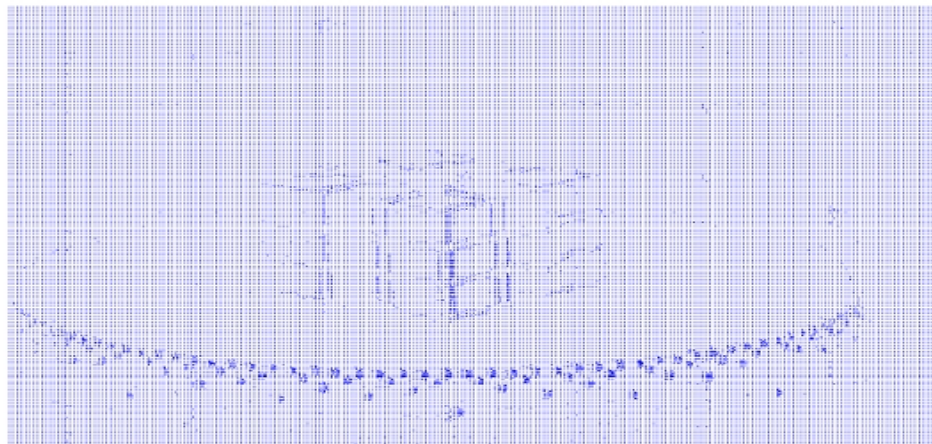
Avec  $k$  le nombre d'itérations  $\bar{\mathbf{u}}$  et  $\bar{\mathbf{v}}$  la moyenne des voisinages de  $\mathbf{u}^k$  et  $\mathbf{v}^k$ . Les vitesses initiales  $u_0$  et  $v_0$  sont fixées à zéro.



a) L'objet à l'instant  $t$



b) L'objet à l'instant  $t + 1$



c) La représentation du *flot optique*

**Figure .I.10 :** Méthode de Horn et Schunck pour le calcul du flot optique. [2]

Parmi les avantages que présentent les méthodes différentielles, nous citons la rapidité en terme de temps de calcul et la recherche de minimisation de cout. [10]

Par contre, les méthodes différentielles souffrent du bruit et de la discrétisation pixellique qui rend l'estimation des dérivées très imprécises [1]. De plus, elles sont inadaptées lorsqu'il s'agit de traiter de grands déplacements.

### I.4.2.3. Méthodes fréquentielles :

#### I.4.2.3.1. Introduction :

L'image d'un mouvement de translation d'un objet dans une scène peut être traduite par un déphasage dans le domaine fréquentiel avec l'utilisation de la Transformée de Fourier (TF).[2]

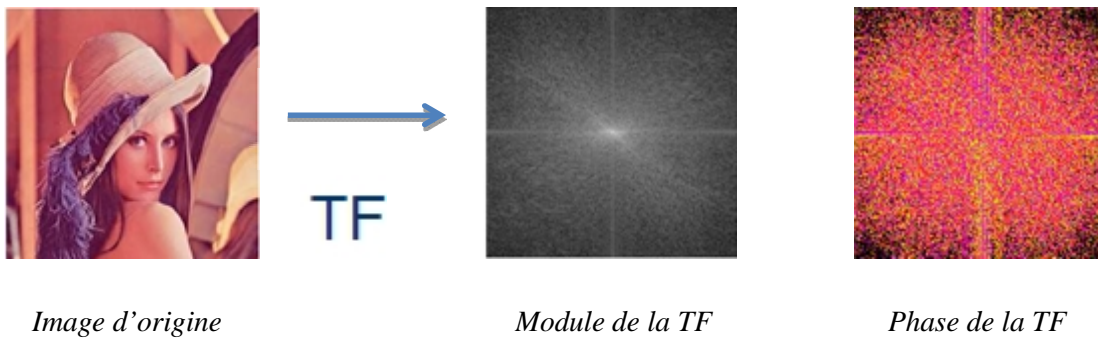


Figure .I.11 : Représentation fréquentielle d'une image. [4]

Une image, d'intensité lumineuse  $I(x,y)$ , d'une largeur  $w$ , et d'une hauteur  $h$ , peut être exprimée dans le domaine fréquentielle par l'utilisation de la transformée de Fourier comme suite :

$$F(u,v) = \sum_{x=0}^{w-1} \sum_{y=0}^{h-1} I(x,y) e^{-2j\pi(ux+vy)/wh} \quad (I.6)$$

De même, on peut exprimer  $I(x,y)$  à partir de sa transformée de Fourier en appliquant la transformée de Fourier inverse :

$$I(x,y) = \frac{1}{wh} \sum_{u=0}^{w-1} \sum_{v=0}^{h-1} F(u,v) e^{2j\pi(ux+vy)/wh} \quad (I.7)$$

D'après la propriété de translation/déphasage si  $F$  est la transformée de Fourier de  $I$ , alors la TF de  $I$  translatée de  $(-x, -y)$ , sera une fonction  $G$  telle que :

$$I(x,y) \longrightarrow F(u,v)$$

$$I(x+x_0, y+y_0) \longrightarrow G(u,v) = F(u,v) e^{2j\pi(u\delta x + v\delta y)/wh}$$

Où  $\|G(u,v)\| = \|F(u,v)\|$  et  $\varphi_G(u,v) = \varphi_F(u,v) + 2j\pi(u\delta x + v\delta y)/wh$

Le déphasage entre  $F$  et  $G$  vaut donc :



$$\varphi(\mathbf{u}, \mathbf{v}) = 2j\pi(\mathbf{u}\delta\mathbf{x} + \mathbf{v}\delta\mathbf{y})/wh \quad (\text{I.8})$$

Alors pour avoir la translation (  $x$ ,  $y$ ), il suffit de calculer le déphasage pour deux couples (u,v). Cette méthode est très sensible au bruit et aux changements d'illumination qui induisent des variations dans les basses fréquences.[2]

#### I.4.2.3.2. Corrélation de phase : [2]

La technique de corrélation de phase consiste à utiliser le spectre croisé normalisé (SPCN) pour déterminer le déplacement entre deux image consécutives par la transformée de Fourier inverse.

Soit  $I_t$  et  $I_{t+1}$  les deux images correspondantes aux instants  $t$  et  $t+1$  successivement, la fonction d'inter-corrélation entre ces deux images est :

$$c_{t,t+1}(\mathbf{x}, \mathbf{y}) = I_{t+1}(\mathbf{x} + \delta\mathbf{x}, \mathbf{y} + \delta\mathbf{y}) * I_t(\mathbf{x}, \mathbf{y})$$

$$c_{t,t+1}(\mathbf{x}, \mathbf{y}) \xrightarrow{TF} C_{t,t+1}(\mathbf{u}, \mathbf{v})$$

$$C_{t,t+1}(\mathbf{u}, \mathbf{v}) = F(\mathbf{u}, \mathbf{v}) \cdot G(\mathbf{u}, \mathbf{v})$$

Ou  $C_{t,t+1}(\mathbf{u}, \mathbf{v})$  : TF de la fonction d'inter-corrélation.

: Le produit de convolution.

• : Le produit scalaire.

Le spectre croisé normalisé (SPCN) est :

$$= \frac{F(\mathbf{u}, \mathbf{v})G(\mathbf{u}, \mathbf{v})}{\|F^*(\mathbf{u}, \mathbf{v})G(\mathbf{u}, \mathbf{v})\|}$$

Alors : 
$$= e^{2j\pi(\mathbf{u}\delta\mathbf{x} + \mathbf{v}\delta\mathbf{y})/wh} \quad (\text{I.9})$$

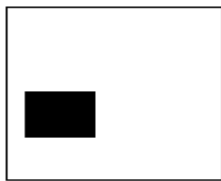
La TF inverse de est une Dirac centré au point ( $\delta\mathbf{x}$ ,  $\delta\mathbf{y}$ ) :

$$\xrightarrow{TF^{-1}} \delta(\mathbf{x} - \delta\mathbf{x}, \mathbf{y} - \delta\mathbf{y})$$

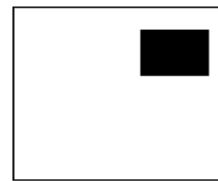
Donc, l'estimation du déplacement (  $x$ ,  $\delta\mathbf{y}$ ) par la methode de corrélation de phase consiste à :

- calculer la TF de  $I_t(x,y)$  et de  $I_{t+1}(x,y)$  correspondant aux instants  $t$  et  $t+1$  respectivement ;
- calculer la SPCN des deux transformées précédentes ;
- calculer la TF inverse de SPCN ;
- extraire le maximum de la transformée précédente ;

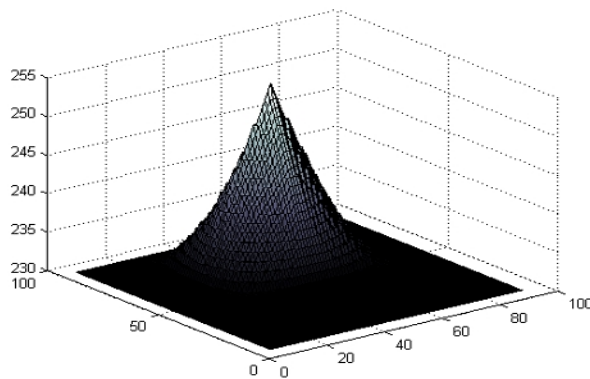
En pratique, l'utilisation de la TFD, la présence du bruit et/ou plusieurs objets en mouvement contribuent à la présence de plusieurs pics dans la fonction de corrélation de phase.



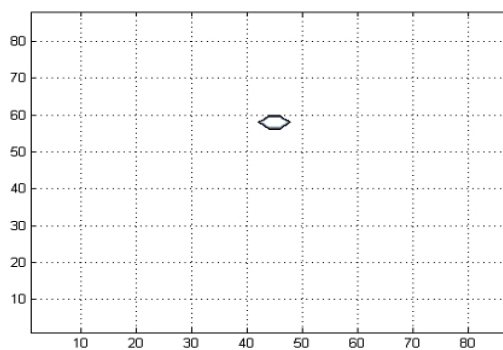
*L'objet a l'instant  $t$*



*L'objet a l'instant  $t+1$*



*La fonction de corrélation de phase*



*Les coordonnées de vecteurs déplacements*

**Figure I.12** : Méthode de corrélation de phase.

Cette méthode est avantageuse parce qu'elle est :

- capable de traiter plusieurs objets en mouvement même s'ils diffèrent en vitesses et cela se traduit par la présence de plusieurs pics dans la fonction de corrélation de phase ;
- rapide en calcul grâce à l'algorithme de calcul FFT pour le calcul de la TFD ;
- insensible aux variations de l'illumination de la scène car, l'illumination n'influe pas sur le calcul de la fonction de corrélation de phase ;

Par contre, cette méthode présente des inconvénients tels que :

- la déformation du spectre : des pics apparaîtront et déformeront le spectre si les composantes du vecteur déplacement ne correspondent pas à un multiple de la fréquence fondamentale ;
- les effets de bord : l'obtention des impulsions parfaites est conditionnée par des déplacements cycliques, ce qui n'est pas souvent le cas. Par conséquent, on aura une fausse estimation du mouvement causé par la présence d'un pic non souhaité ;

#### I.4.2.3.3. Transformée de Fourier-Mellin : [2]

La transformée de Fourier-Mellin consiste à chercher les paramètres  $(\theta, \rho)$  correspondants à une (rotation, homothétie) respectivement pour estimer le mouvement relatif, et cela grâce à la représentation log-polaire de l'espace fréquentiel par le passage :

$$(\mathbf{u}, \mathbf{v}) \longrightarrow (\theta, \log \rho)$$

Soit  $f$  une fonction bidimensionnelle, la transformée de cette fonction par une rotation d'angle  $\alpha$ , une homothétie  $\rho$ , et une translation de vecteur  $(x_0, y_0)$ , s'écrit sous la forme :

$$\mathbf{g}(\mathbf{x}, \mathbf{y}) = f(\cos \alpha \mathbf{x} + \sin \alpha \mathbf{y} - \mathbf{x}_0, -\sin \alpha \mathbf{x} + \cos \alpha \mathbf{y} - \mathbf{y}_0)$$

Notons  $G(\mathbf{u}, \mathbf{v})$  la TF de  $\mathbf{g}(\mathbf{x}, \mathbf{y})$  et  $F(\mathbf{u}, \mathbf{v})$  la TF de  $f(\mathbf{u}, \mathbf{v})$  on aura :

$$\begin{aligned} \mathbf{g}(\mathbf{u}, \mathbf{v}) &\xrightarrow{TF} \mathbf{G}(\mathbf{u}, \mathbf{v}) \\ \mathbf{f}(\mathbf{u}, \mathbf{v}) &\xrightarrow{TF} \mathbf{F}(\mathbf{u}, \mathbf{v}) \end{aligned}$$

La relation entre les modules des deux transformées est :

$$\|\mathbf{G}(\mathbf{u}, \mathbf{v})\| = \frac{1}{\sigma^2} \|\mathbf{F}\left(\frac{1}{\sigma}(\mathbf{u} \cos \alpha + \mathbf{v} \sin \alpha), \frac{1}{\sigma}(-\mathbf{u} \sin \alpha + \mathbf{v} \cos \alpha)\right)\|$$

On constate que l'amplitude de la TF dépend des paramètres de rotation et homothétie, par contre elle est invariante en fonction du vecteur de translation  $(x_0, y_0)$ .

En passant les coordonnées fréquentielles en coordonnées polaires on aura :

$$F_p(\theta, \rho) = ||F(\rho \cos\theta, \rho \sin\theta)|| ; \quad 0 \leq \theta \leq 2\pi, 0 \leq \rho$$

$$G_p(\theta, \rho) = ||G(\rho \cos\theta, \rho \sin\theta)|| ; \quad 0 \leq \theta \leq 2\pi, 0 \leq \rho$$

Après remplacement des coordonnées  $(u, v)$  dans les deux fonction F et G on tire l'équation suivante :

$$G_p(\theta, \rho) = \frac{1}{\sigma^2} F_p\left(\theta - \alpha, \frac{\rho}{\sigma}\right)$$

Finalement, en exprimant la coordonnée radiale sous forme logarithmique, en posant :

$$r = \log \rho \quad \text{et} \quad s = \log \sigma$$

On obtient :

$$G_{lp}(\theta, r) = \frac{1}{\sigma^2} F_{lp}(\theta - \alpha, r - s) \quad (I.10)$$

$$F_{lp}(\theta, r) = F_p(\theta, \rho) ;$$

$$G_{lp}(\theta, r) = G_p(\theta, \rho) ;$$

Cette méthode est employée en pratique pour estimer des transformations globales, car elle utilise la contribution de tout le spectre (ou au moins une large partie), ce qui implique une étude spatiale importante des pixels utilisés pour l'estimation de chaque transformation.

#### I.4.2.4. Méthode d'analyse en composantes principales (ACP) :

L'analyse en composantes principales (ACP), est une méthode statistique multivariable, qui permet de comprimer des données et de réduire leur dimensionnalité de sorte que l'information essentielle soit maintenue et plus facile à analyser que dans l'ensemble originale des données. Il s'agit donc d'obtenir le résumé le plus pertinent possible des données initiales.

Cette technique consiste à transformer un certain nombre de variables corrélés en un ensemble plus petit de variables non corrélées. [5]

Ainsi, elle remplace les données spectrales d'origine (les variables prédictives), fortement redondantes dans le cas de données de spectrométrie, par des variables synthétiques, les composantes principales d'allongement du nuage des facteurs. Elle consiste en une rotation de manière à placer les nouveaux axes dans la direction de plus grande dispersion du nuage de points. Le deuxième axe est orthogonal au premier et tient compte de la plus grande dispersion résiduelle. [6]

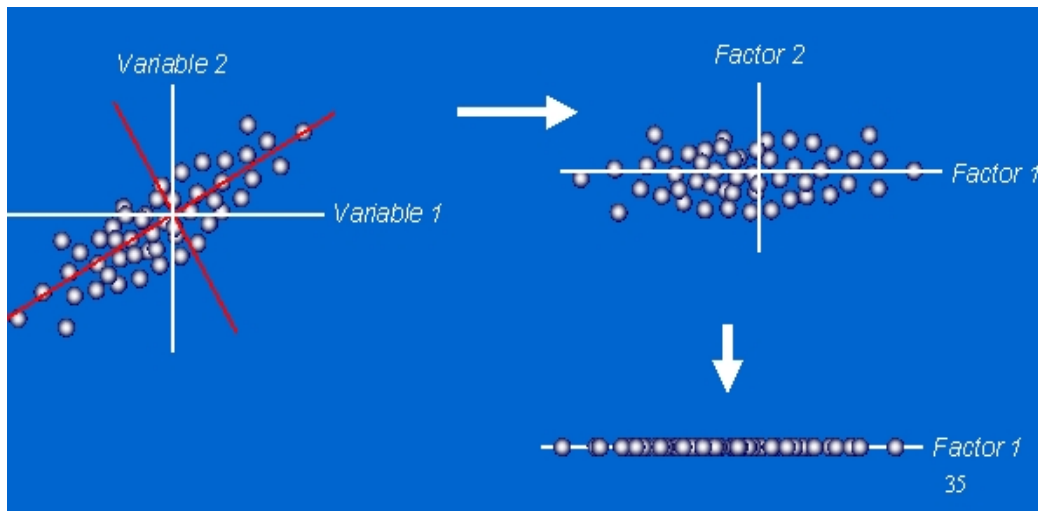


Figure I.13 : Changement de repère orthogonal. [7]

Considérant une séquence vidéo de  $p$  images, à partir de ces images on calcule la matrice variance-covariance  $C$ , le calcul de cette matrice nécessite le réarrangement des images sous forme de vecteurs-colonnes :

$$\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)^T \quad (\text{I.11})$$

$n$  : le nombre de ligne (nombre de pixels de l'image  $x$ )

La matrice des données résultante  $X_{n \times p}$  contient donc l'ensemble des caractéristiques des points à considérer (intensité lumineuse).

La variance-covariance  $C$  est donnée par :

$$\mathbf{C} = \mathbf{X}^T \cdot \mathbf{D}_p \cdot \mathbf{X} \quad (\text{I.12})$$

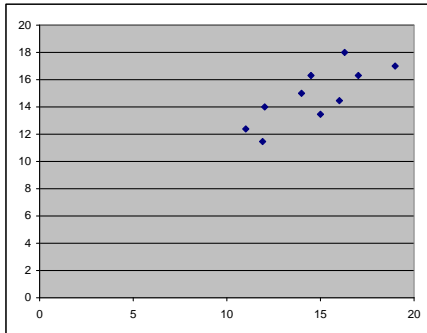
$$\mathbf{D}_p = \frac{1}{p} \mathbf{I}_p \quad (\text{I.13})$$

Où  $D_p$ : Domaine de définition des pixels

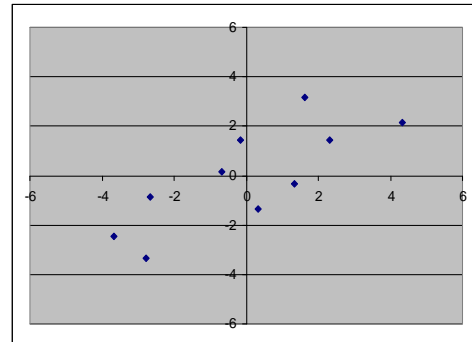
$I_p$  : Matrice identité d'ordre p

$\bar{X}$  : matrice des données centrées réduites [8]

La matrice des données introduite au calcul de la variance-covariance doit être centrée et réduite, la figure suivante permet de comparer les données sans et avec valeurs centrées et réduites :



(a) Valeurs des Données originales



(b) valeurs des données centrées et réduites

**Figure I.14** : Comparaison des données (a) avec (b) sans valeurs centrées et réduites. [7]

Après calcul de la matrice variance-covariance, cette dernière est diagonalisée pour obtenir une base de vecteurs propres, les valeurs propres associées sont disposées dans la matrice diagonale L par ordre décroissant. Pour réduire la dimensionnalité du modèle, seuls les vecteurs propres associés aux M plus grandes valeurs propres sont conservés. [9]

On peut remplir la matrice de données avec les dérivées temporelles en tout point qu'on note Y, par conséquent la dimension de l'espace sera de P-1.

### Exemple :

Considérons une séquence de 10 trames de 288 lignes par 720 colonnes dont la première et la dernière sont représentées sur la figure suivante :

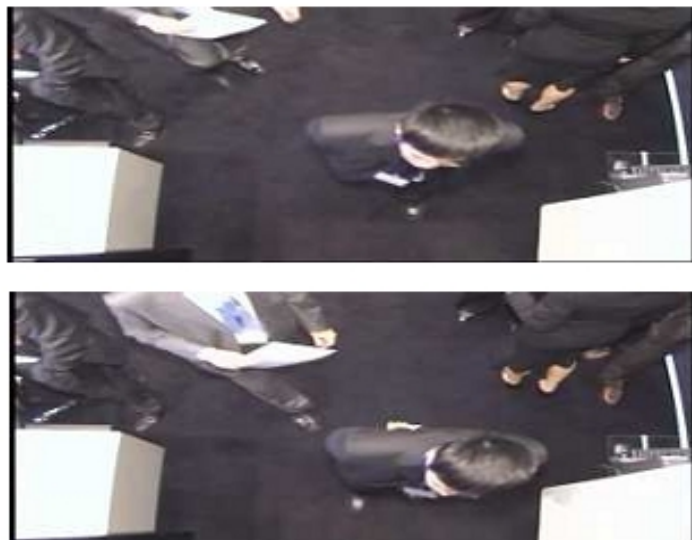


Figure .I.15 : La première et la dernière trame d'une séquence de 10 trames.

La matrice X est caractérisée par 288x720 lignes et 10 colonnes, alors que la matrice Y a 288x720 lignes et 9 colonnes.

Les zones en mouvement apparaissent clairement lorsqu'on projette la matrice Y sur les deux premiers axes principaux. La différence entre une zone statique et une zone en mouvement est accentuée sur ces axes.

On peut confirmer cette observation par l'histogramme de la variance expliqué par les facteurs selon la figure suivante :

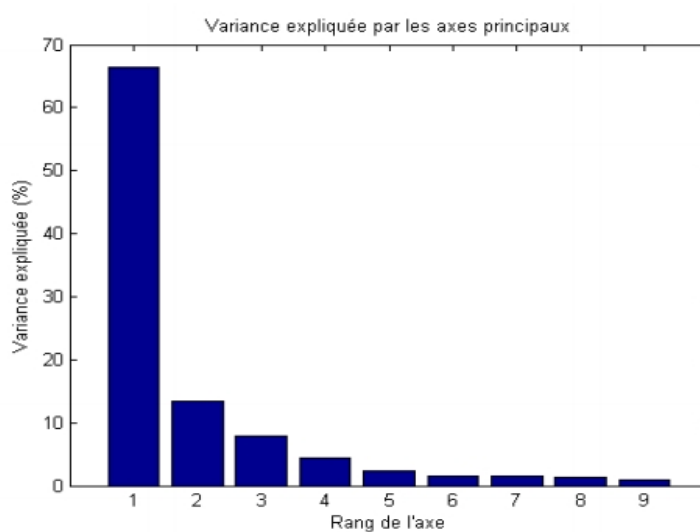
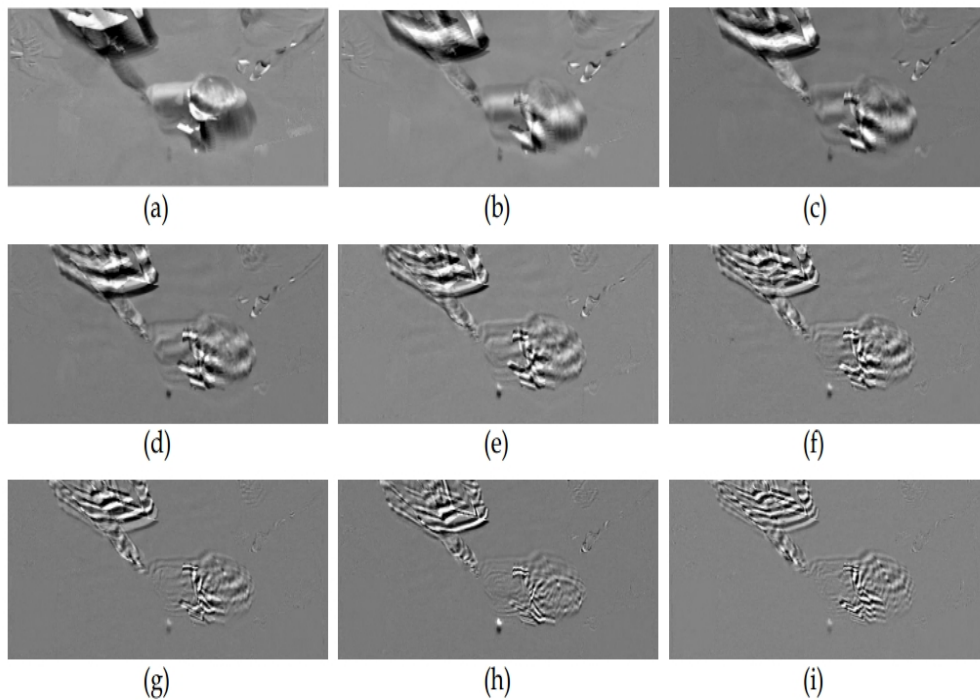


Figure .I.16 : Variance expliquée par les axes principaux.

Alors, en ne retenant que les deux premiers axes principaux, 20% de l'information initialement contenue dans notre matrice de données suffisent à préserver 80% de la variance observé.

La figure suivante représente la projection de la matrice Y sur chacun des axes principaux mises en évidence par l'ACP :



**Figure I.17 :** Projection de la matrice Y sur les axes principaux ordonnées selon le rang du facteur correspondant.

**Détection des zones de mouvement :**

A partir de la figure I.17 (a), on peut facilement détecter les mouvements au niveau des pixels, et cela en sélectionnant les pixels dont la valeur absolue est élevée (les plus sombres et les plus clairs) pour obtenir une segmentation objet mobile/arrière plan. la figure suivante représente la segmentation automatique de la projection de Y sur le premier axe principal.





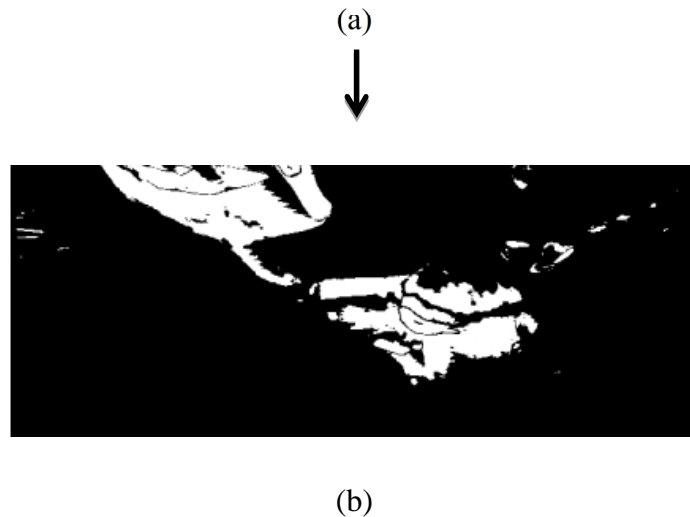


Figure .I.18 : (a) projection de Y dans le premier axe principale, (b) segmentation de l'arrière-plan obtenue à partir de (a). [9]

#### I.4.2.5. Méthode morphologique :

##### I.4.2.5.1. Introduction : [2]

La morphologie mathématique est un ensemble de méthodes d'analyse d'image, qui est utilisée pour la détection et l'estimation de mouvement. Elle consiste en l'étude des objets en fonction de leurs formes, tailles, textures, et leurs niveaux de gris ou de couleurs.

Son principe de base est de comparer l'image à analyser par rapport à un ensemble de géométrie connue appelé élément structurant. L'utilisation d'éléments structurants permet de calculer des filtres spatio-temporels intéressants, et d'intégrer des changements temporels par accumulation (lorsque l'élément structurant est allongé dans l'axe temporel), et de discriminer un déplacement donné en orientant l'élément structurant dans la direction correspondante.

L'espace-temps discret est assimilé à  $Z^3$  avec deux dimensions spatiales  $x, y$  et une dimension temporelle  $t$ . Considérons une séquence d'image  $I_t(x, y)$  où  $t$  un index de temps et  $(x, y)$  un index d'espace bidimensionnel, un élément structurant temporel  $\tau \in [t_1, t_2]$  est utilisé pour définir les filtres morphologiques temporels, avec  $t_1$  et  $t_2$  deux instants tels que  $t_1 < t_2$ , cet élément représente un intervalle temporel d'interaction.

On définit l'érosion temporelle en niveau de gris  $\epsilon(x, y, t)$  comme :

$$\epsilon(I_t(x, y, \tau)) = \min(I(x, y, t + z)); \quad z \in \tau \quad (\text{I.14})$$

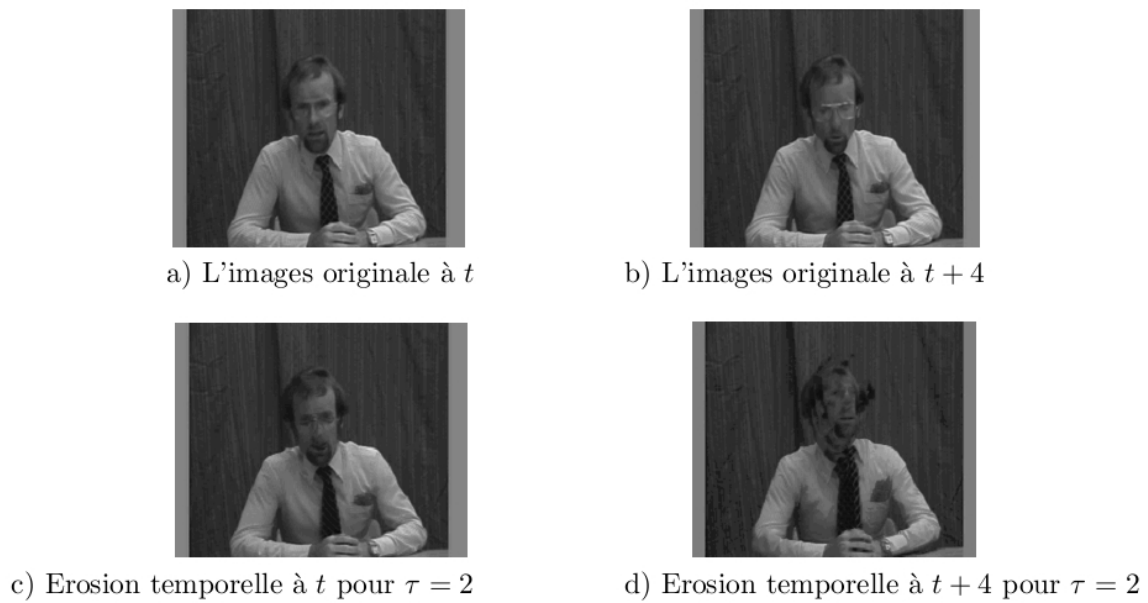


Figure .I.19 : L'érosion temporelle.

Ainsi, la dilatation temporelle en niveau de gris  $(x, y, t)$  comme :

$$(\mathbf{I}(x, y, t + z)) = \max (\mathbf{I}(x, y, t + z)); \quad z \quad (\text{I.15})$$

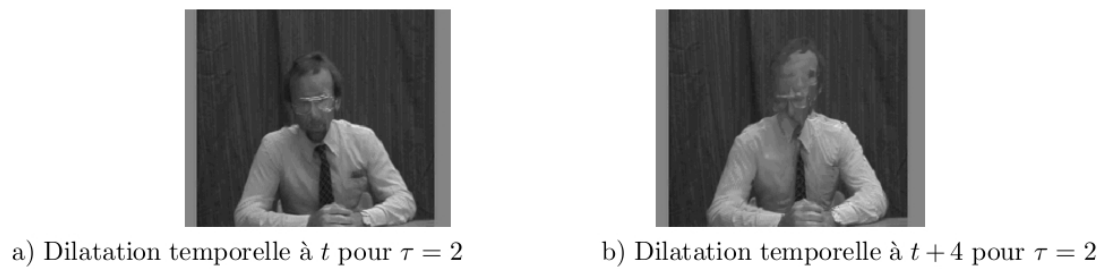
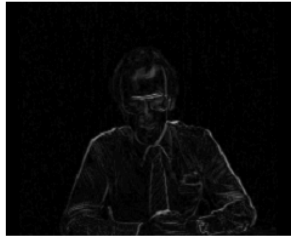


Figure .I.20 : La dilatation temporelle.

Et on définit le gradient comme:

$$(\mathbf{x}, \mathbf{y}, \mathbf{t}) = (\mathbf{x}, \mathbf{y}, \mathbf{t}) - \epsilon (\mathbf{x}, \mathbf{y}, \mathbf{t}) \quad (\text{I.16})$$



a) Gradient temporel à  $t$  pour  $\tau = 2$



c) Gradient temporel à  $t + 4$  pour  $\tau = 2$

Figure 1.21 : Le gradient temporel.

Nous allons maintenant présenter les trois méthodes de détection du mouvement (moyenne récursive, morphologie oubliuse, et moyenne - ).

#### I.4.2.5.2. Moyenne récursive :

Considérons  $I_t$  l'image acquise à l'instant  $t$ , le fond est calculé récursivement à l'aide de la formule suivante :

$$\begin{aligned} M_t &= \alpha I_t + (1 - \alpha)M_{t-1} \\ &= \alpha(I_t - M_{t-1}) + M_{t-1} \end{aligned} \quad (I.17)$$

$\alpha \in [0,1]$  une constante.

En pratique, pour éviter de multiplier les erreurs d'arrondis dues aux multiplications par  $\alpha$ , on utilise la seconde formule.

On remarque que si :

- $\alpha = 1 \longrightarrow M_t = I_t$ , d'où l'image acquise sera le fond courant.
- $\alpha = 0 \longrightarrow M_t = M_{t-1} = M_0$ , d'où le fond reste égale a son prédécesseur.

La procédure de cette technique commence par initialiser le fond à  $M_0$ , tel que pour chaque pixel de coordonnées  $(x, y)$  :

$$M_0(x, y) = I_0(x, y) \quad (I.18)$$

Puis, on détermine le fond courant à partir du fond précédent et de l'image acquise par l'expression :

$$M_t = \alpha I_t + (1 - \alpha)M_{t-1} \quad (I.19)$$

La différence est donnée par la formule :

$$\Delta_t = |I_t(x, y) - M_t(x, y)| \quad (I.20)$$

Le résultat courant de la détection est donnée par la formule suivante :

$$D_t = \delta_t(|I_t(x, y) - M_t(x, y)|) [2] \quad (I.21)$$

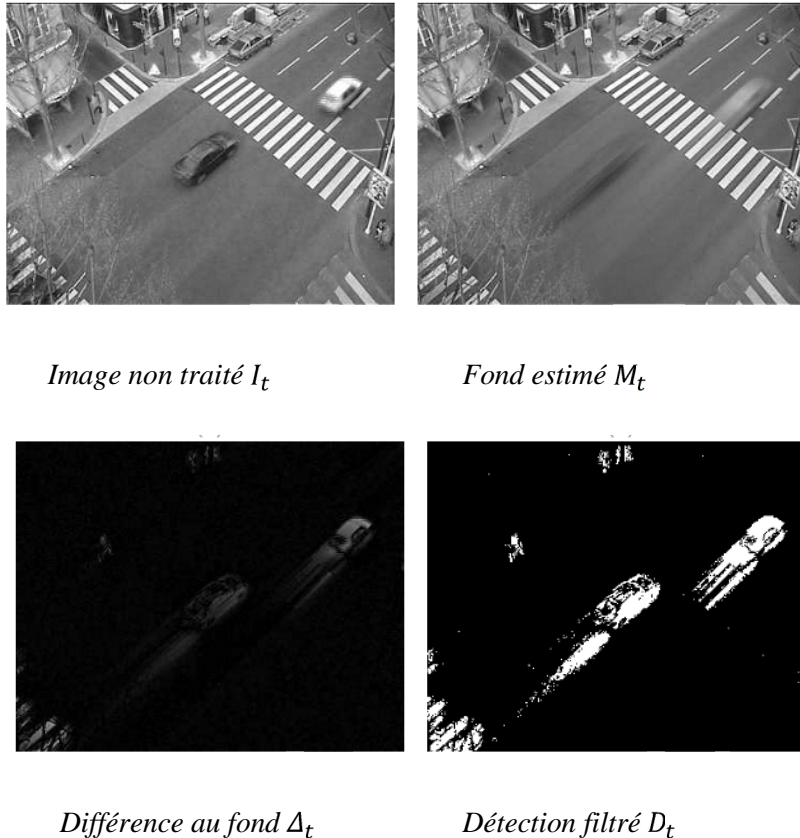


Figure I.22 : Résultat de la détection par moyenne réursive ( $\alpha = \frac{1}{8}$ ). [10]

Mais, ces filtres ont pour inconvénients d'une part la nécessité d'utiliser un buffer d'une taille très prohibitive ce qui implique une occupation exhaustive de l'espace mémoire, et d'autre part leur sensibilité aux larges variations brusques comme bruit impulsionnel. [2]

#### I.4.2.5.3. Morphologie oublieuse temporelle :

Les filtres morphologiques oublieux sont des filtres hybrides, maximum linéaire et minimum linéaire. Leur principe de base est de combiner le calcul des minimums et maximums avec un terme d'oublie  $\alpha$ .

Cette méthode permet de détecter efficacement les mouvements de faible amplitude (objets lents, petits ou faiblement contrastés) avec une relative bonne résistance au bruit grâce à la partie linéaire du filtre.

Soit  $M_0, m_0$  la dilatation morphologique oublieuse, et l'érosion morphologique initiale respectivement tel que :

$$M_0(x, y) = n_0(x, y) = I_0(x, y) \quad (I.22)$$

Pour chaque instant  $t$ , la dilatation morphologique oublieuse, et l'érosion morphologique sont :

$$M_t(x, y) = \alpha I_t(x, y) + (1 - \alpha) \max(I_t(x, y), M_{t-1}(x, y)) \quad (I.23)$$

$$n_t(x, y) = \alpha I_t(x, y) + (1 - \alpha) \min(I_t(x, y), n_{t-1}(x, y)) \quad (I.24)$$

On remarque que si :

- $\alpha = 0 \begin{cases} M_t(x, y) = \max(I_t(x, y), M_{t-1}(x, y)) \\ n_t(x, y) = \min(I_t(x, y), n_{t-1}(x, y)) \end{cases} \quad (I.25)$
- $\alpha = 1 \longrightarrow M_t(x, y) = n_t(x, y) = I_t(x, y)$

La dilatation (respectivement l'érosion) représente une maximisation (respectivement minimisation) entre l'image courante et le fond sur un intervalle  $[t - 1, t]$ . Le gradient morphologique oublieux est :

$$\Gamma_t(x, y) = M_t(x, y) - n_t(x, y) \quad [2] \quad (I.26)$$

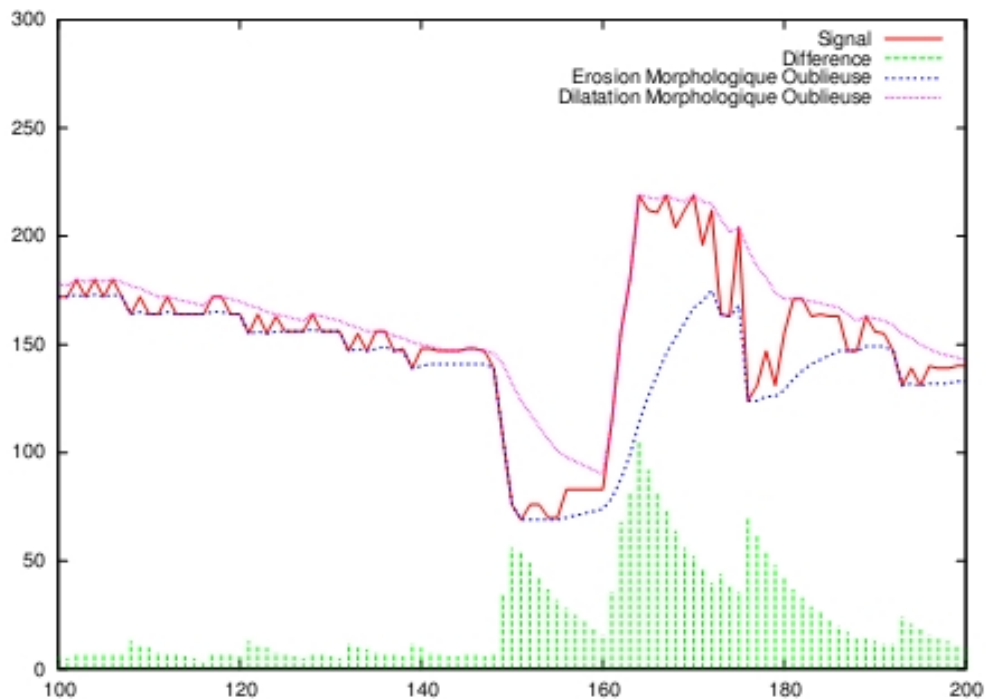


Figure I.23 : Filtrage morphologique oublieux. [11]

La courbe rouge correspond à la séquence originale, la courbe violette au maximum oublieux, la courbe pointillée bleue au minimum oublieux, et la courbe en tiret vert au gradient morphologique oublieux ( $\alpha = \frac{1}{8}$ ).

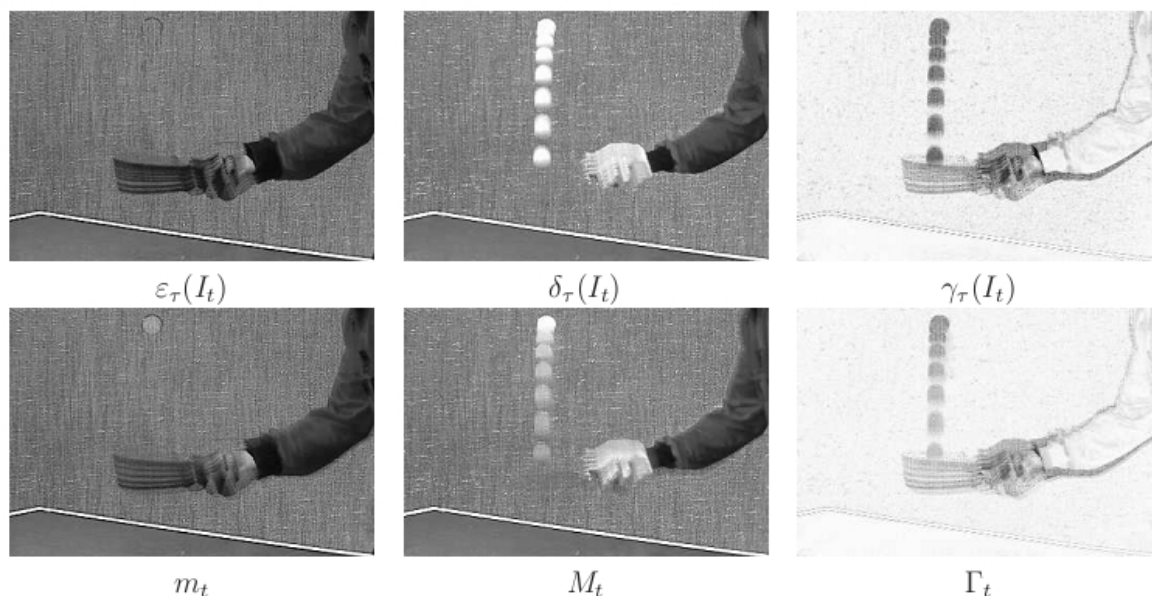


Figure I.24 : Application des opérateurs morphologies oublieux (en bas), comparés aux opérateurs morphologies classiques (en haut) avec ( $\alpha = \frac{1}{9}$ ). [10]

On remarque que la balle est mieux segmentée et sa position actuelle est mieux définie par les opérateurs de dilatation et d'érosion oubliieuse que dans les opérateurs classiques. [10]

Parmi les avantages de cette technique, nous notons : qu'elle est moins sensible aux bruits impulsionnels, et qu'elle nécessite uniquement l'utilisation d'un buffer de taille 2 images pour le calcul de la dilatation et l'érosion oubliieuses. [2]

#### I.4.2.5.4. Estimation - : [10]

Le filtre - , entièrement non-linéaire, utilise seulement des comparaisons et des incréments/décréments élémentaires.

On utilise la différence  $\Delta_t$  entre l'image originale  $I_t$  et la moyenne  $M_t$ , tel que pour chaque trame t, pour chaque pixel  $(x, y)$  :

$$\Delta_t = |I_t(x, y) - M_t(x, y)| \quad (I.27)$$

Puis, on calcule la moyenne - qui permet la mesure de la vraisemblance du mouvement :

$$\text{Initialisation pour chaque pixel } (x, y) : M_0(x, y) = I_0(x, y) \quad (I.28)$$

Pour chaque trame t, pour chaque pixel x :

$$\begin{cases} M_t(x, y) = M_{t-1}(x, y) + 1 & \text{si } \Delta_t(x, y) < 0 \\ M_t(x, y) = M_{t-1}(x, y) - 1 & \text{si } \Delta_t(x, y) > 0 \end{cases} \quad (I.29)$$

Ensuite, le calcul de la variance - définie par la moyenne de N fois les différences non nulles :

$$\text{Initialisation pour chaque pixel } (x, y) : V_0(x, y) = |\Delta_0(x, y)| \quad (I.30)$$

Pour chaque trame t, pour chaque pixel  $(x, y)$  tel que  $\Delta_t(x, y) \neq 0$  :

$$\begin{cases} V_t(x, y) = V_{t-1}(x, y) + 1 & \text{si } V_{t-1}(x, y) < N \times |\Delta_t(x, y)| \\ V_t(x, y) = V_{t-1}(x, y) - 1 & \text{si } V_{t-1}(x, y) > N \times |\Delta_t(x, y)| \end{cases} \quad (I.31)$$

Enfin, calcul de l'étiquette de mouvement par comparaison entre la différence et la variance :

Pour chaque trame t, pour chaque pixel  $(x, y)$  :



$$\begin{cases} D_t(x, y) = 0 & \text{si } V_t(x, y) < \Delta_t(x, y) \\ D_t(x, y) = 1 & \text{si } V_t(x, y) > \Delta_t(x, y) \end{cases} \quad (\text{I.32})$$

La figure suivante montre le résultat de cette méthode pour une trame prise dans une scène de trafic urbain.

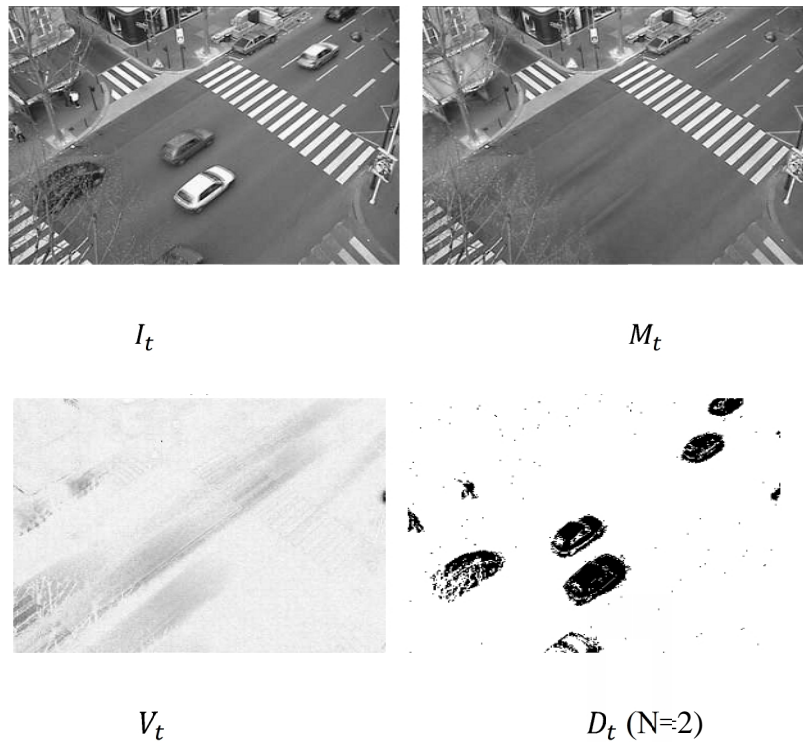
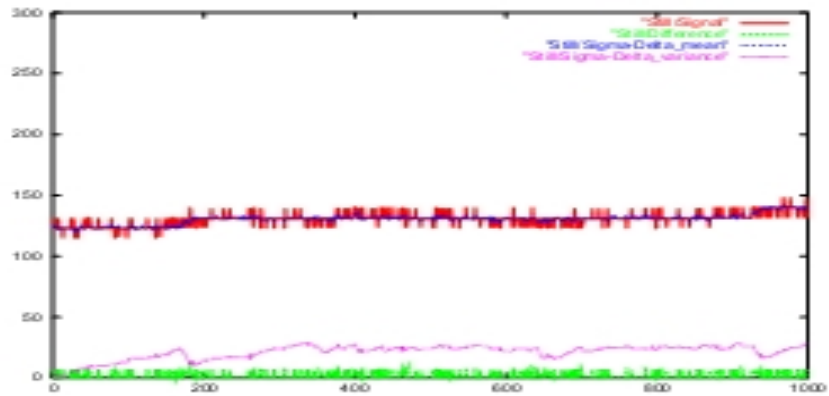


Figure I.25 : Résultat de la méthode d'estimation - .

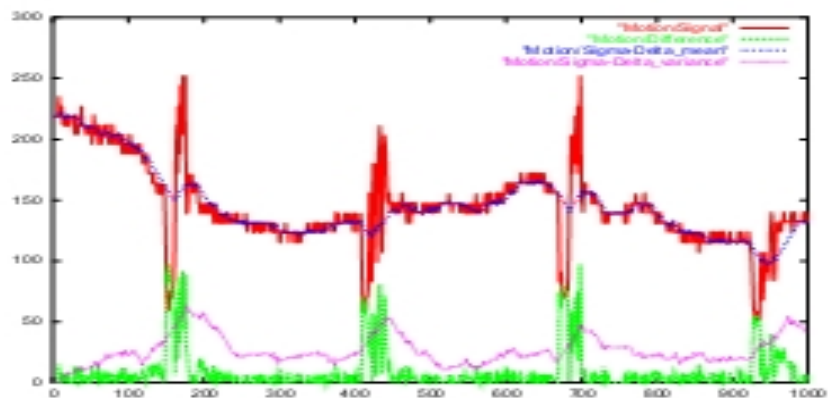
On remarque que la segmentation des objets est précise quel que soit le type de mouvement observé (gros objets en mouvement rapide (voiture) ou petits objets lents (piétons en haut)).

Un exemple illustre le comportement de trois pixels particuliers d'une séquence de 1000 images représentant une scène avec passage de véhicules, perturbé par un mouvement parasite au premier plan est représenté dans la figure suivante :

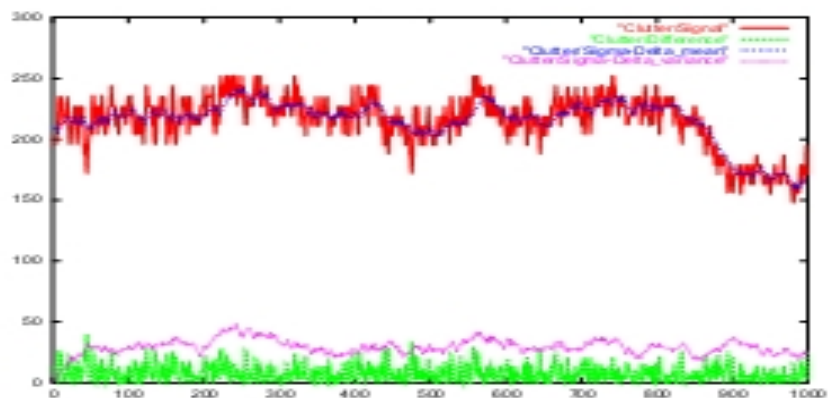




(1)



(2)



(3)

Figure I.26 : L'estimateur  $L_t$  en fonction du type de zone où se trouve le pixel.

On a représenté pour chaque pixel  $(x, y)$ , quatre graphes représentant les variations au cours du temps de :

- $I_t$  en ligne continue rouge.
- $M_t$  en ligne avec hacheurs espacées bleue.
- $\Delta_t$  en ligne hachuré verte.
- $V_t$  en ligne pointillée violette.

Le premier pixel (*Fig .I.26 (1)*) correspond à une zone du fond statique sans perturbation, le deuxième pixel (*Fig .I.26 (2)*) correspond à une zone de passage d'objet mobile, le troisième pixel (*Fig .I.26 (3)*) correspond à une zone de mouvement parasite qui est due à la nature de la scène plutôt qu'à des objets mobiles.

Parmi les avantages de cette méthode, c'est qu'elle permet une détection robuste et précise d'objets mobiles pour un cout modéré en terme de consommation de mémoire et de complexité de calcul.

Par contre elle présente comme inconvénient une limitation de capacité d'adaptation à certaines scènes complexes, comme les mouvements périodiques (exemple : une vague en arrière-plan) qui seraient interprétés par l'algorithme comme faisant partie de l'ensemble des objets mobiles au lieu du fond.

### **I.5. Conclusion :**

La détection de mouvement par les méthodes citées dans ce chapitre n'est pas vraiment évidente. En effet, elle présente plusieurs problématiques d'adaptabilité, de robustesse avec les changements d'intensité lumineuse, ... etc., qui peuvent être remédiées grâce à une autre méthode statistique basée sur le fond statique dont nous ferons une étude détaillée dans le chapitre suivant.

# *Chapitre II*

---

*Détection de mouvement par méthode de mélange de  
Gaussiennes*

## **II.1. Introduction :**

De nos jours, aucune méthode de détection de mouvement n'a fait preuve d'une totale fiabilité. La plupart d'entre elles souffrent de multiples insuffisances liées à de nombreux problèmes tels que : manque de robustesse lié aux changements de luminosité graduelle (temps du jour) ou brusque (allumer ou éteindre la lumière), les mouvements inintéressants dans la scène (les vagues, feuilles et branches d'arbres), problèmes liés aux ombres, ... etc. Tous ces problèmes pourront conduire à de fausses ou intéressantes détections, ou bien à l'impossibilité de détecter un objet en mouvement (que nous expliciterons dans ce chapitre).

Les méthodes basées sur la soustraction de fond, les plus utilisées, ont prouvé de bons résultats et ont un fort potentiel en ce qui concerne la détection. Malgré cela, ces problèmes persistaient encore une fois. Une nouvelle méthode, probabiliste, basée sur le mélange de Gaussiennes a été développée. Cette dernière a bien répondu aux critères de robustesse avec les changements de luminosité, répétitions de mouvement des objets d'une scène, mouvements lents des objets, et l'ajout (l'élimination) des objets à (de) l'arrière-plan. [23]

Nous verrons en détail dans ce chapitre, le principe de la méthode, les algorithmes utilisés pour la modélisation du fond, l'apprentissage et mise à jour des paramètres. Puis, on discutera du problème des insuffisances et des améliorations apportées à cette méthode.

## **II.2. Présentation du problème :**

Prenons le cas d'un système de surveillance comportant une caméra (statique) observant une scène. Une hypothèse généralement applicable est que les images de la scène (sans les objets en mouvement) présentent un comportement régulier qui peut être bien décrit par un modèle statistique. Si nous avons un modèle statistique de la scène alors, un objet en mouvement peut être détecté en repérant les parties de l'image qui ne correspondent pas au modèle probabiliste. Ce processus est connu sous le nom de « soustraction de fond ». [31]

Chaque pixel  $x$  de la scène a une densité de probabilité  $p(x)$  étant inconnue, et variable au cours du temps  $t$ . L'approche la plus simple est de prendre l'image de la scène sans les objets en mouvement (l'image de fond). Puis, modéliser chaque pixel de celle-ci par une Gaussienne unique de densité de probabilité :

$$p_G(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \quad (\text{II.1})$$

Ou  $\sigma^2$  : est la variance.

$\mu$  : est la moyenne.

Le modèle de fond étant modélisé. Pour détecter un mouvement dans la scène, il suffit de comparer la nouvelle image (image courante) à l'image de fond : c'est-à-dire si à un instant  $t$  la distribution probabiliste  $p(x)$  d'un pixel  $x$  ne concorde pas à la distribution probabiliste  $p_G(x)$  de l'image de fond (du même pixel  $x$ ) alors, un mouvement est détecté. [32][33]

- **Avantages, et insuffisances :**

Comme avantages :

- bonne adaptation aux faibles mouvements de la scène avec une mise à jour en utilisant un simple filtre d'adaptation [32] ;
- une seule Gaussienne adaptative par pixel aurait suffit pour modéliser la scène (image de fond) ;

La modélisation de la scène serait possible que si il n'y avait que les changements de luminosité, et un seul mode de mouvement (mouvement d'un homme) [33][12]. Dans les situations réelles ou la scène est complexe (dynamique), il y a mis à part les changements de luminosité, plusieurs modes de mouvement : mouvement des objets de fond (feuilles et branches d'arbres, les vagues, ...), mouvement lent ou rapide d'objets, ou objets de fond statiques devenant mobiles, ...etc. Cette méthode appelée aussi : méthode « uni-modale » devient alors insuffisante car, elle permet uniquement de modéliser une scène simple (statique).

Pour modéliser l'arrière-plan d'une scène dynamique, un nouveau modèle probabiliste basé sur un mélange de Gaussiennes a été développé. Il consiste à décomposer chaque pixel de l'image en plusieurs distributions (un mélange de  $K$  Gaussiennes).

### II.3. Détection de mouvement par méthode de mélange de Gaussiennes :

#### II.3.1. Introduction :

Au vu des insuffisances apportées par la méthode de détection citée précédemment. Une extension est possible. Elle consiste à classifier les pixels de l'image comme faisant

partie de l'avant-plan (objets) ou l'arrière-plan (fond) à l'aide d'un mélange de distributions Gaussiennes. Plutôt que de limiter la description à une seule distribution probabiliste centrée autour de l'unique couleur recherchée. [22]

La méthode par mélange de Gaussiennes répond bien au critère de robustesse (avec les changements de luminosité), les répétitions de mouvement des objets dans la scène, les mouvements lents des objets, et l'ajout ou l'élimination des objets à ou de l'arrière-plan. [23]

### II.3.2. Principe du mélange de Gaussiennes :

Chaque pixel d'une image est représenté par une ou plusieurs distributions probabilistes Gaussiennes. Ceci contraste avec la technique précédente (uni-modale) qui attribuait seulement une distribution pour la totalité des pixels de l'image. La figure suivante illustre un exemple de mélange de 5 Gaussiennes :

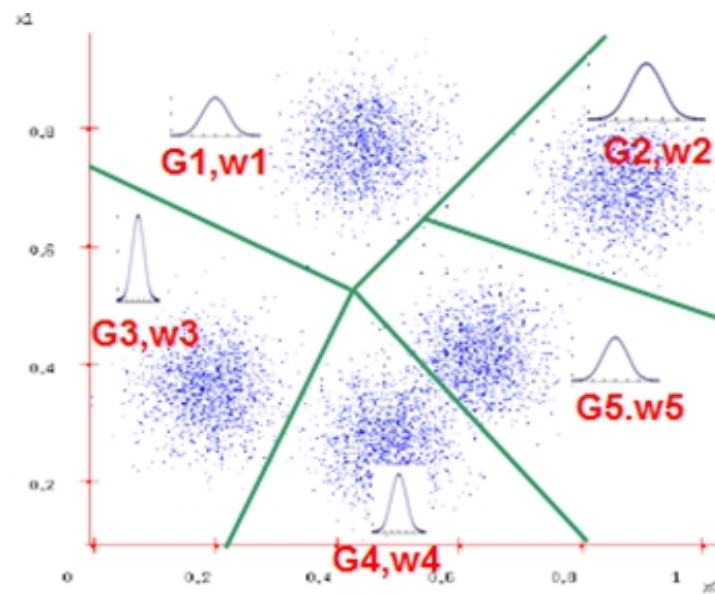


Figure. II.1 : Mélange de gaussiennes.

Les distributions probabilistes sont initialisées à partir du contenu de la première image de la séquence, en définissant le centre d'une courbe Gaussienne distincte qui s'étend le long d'une région de couleurs arbitraires à partir de la première couleur de chaque pixel. [22]

Russel et Friedman ont proposé une solution (dans le cadre de la surveillance routière) en utilisant un mélange de trois Gaussiennes. Les pixels ont été classifiés selon leurs valeurs en trois distributions indépendantes et prédéterminées : la route, l'ombre, et les véhicules. En

attribuant la composante la plus sombre à l'ombre, celle ayant la plus large variance au véhicule, et celle qui reste à la route. En comparant chaque pixel de l'image aux trois distributions classifiées précédemment, la détection d'un objet (véhicule) est possible que si le pixel étudié correspond à la Gaussienne classée « véhicules ». [12]

Cette solution est-elle aussi insuffisante, car le fond est toujours supposé uni-modal (mouvement des véhicules seulement), et malgré la présence de trois distributions, des pixels peuvent ne pas correspondre à ces trois dernières.

Stauffer et Grimson ont pris en compte le fait que certains pixels de l'arrière-plan peuvent présenter des composantes complexes par rapport à leurs couleurs. Par exemple, un pixel représentant un feu de circulation peut potentiellement prendre la couleur verte, jaune ou rouge au cours du temps. L'algorithme donc attribue aux pixels une distribution Gaussienne pour chaque région de couleur distincte tel qu'illustré dans la figure suivante : [22] [28]

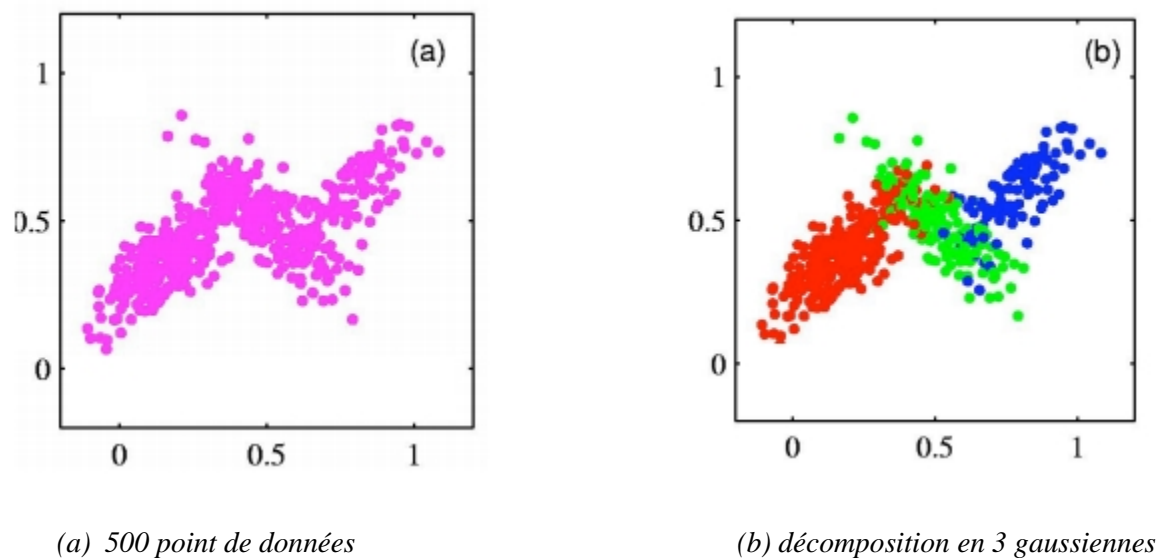


Figure. II.2 : Comportement des pixels représenté par un mélange de distributions Gaussiennes.

Le système recherche des distributions Gaussiennes qui décrivent le mieux la valeur courante des pixels pour chaque image de la séquence parmi les distributions associées à chaque pixel. Si la recherche est concluante, la distribution est raffinée afin de mieux intégrer les caractéristiques chromatiques courantes, sinon une distribution supplémentaire est ajoutée à la liste du pixel en question. [22] Encore une fois, ce modèle fait l'hypothèse que les couleurs de l'arrière-plan demeurent plus au moins constantes et que le bruit d'acquisition soit non corrélé et de faibles changements de luminosité. [22][24]

Lors de l'analyse d'une image, à chaque fois qu'une distribution probabiliste est associée à un pixel, une mesure de correspondance est attribuée à la distribution. Sous un autre angle, plus un pixel est représenté par une même distribution, plus il est probable que cette distribution probabiliste représentant une couleur spécifique soit classifiée comme faisant partie de l'arrière-plan. Les distributions associées aux objets en mouvement ont une mesure de correspondance moins élevée puisque les pixels de l'image y sont associés moins fréquemment. [22]

### II.3.3. Etude de la méthode de détection par mélange de Gaussiennes :

On représente un pixel donné  $(x_0, y_0)$  par un processus aléatoire, des séries de valeurs scalaires pour le niveau de gris et des vecteurs pour les images en couleur [12]. Soit l'historique d'un pixel donné  $(x_0, y_0)$  : [23]

$$\{X_1, \dots, X_t\} = \{I(x_0, y_0, i) : i=1, \dots, t\} \quad (\text{II.2})$$

La probabilité d'observer la valeur du pixel  $x_t$  à l'instant  $t$  est donnée par l'expression suivante :

$$P(X_t) = \sum_{i=1}^K w_{i,t} \cdot \eta(X_t, \mu_{i,t}, \Sigma_{i,t}) \quad (\text{II.3})$$

Avec comme paramètres :

$K$ : le nombre de distributions (détermine la multi-modalité, i.e. le nombre de modes de mouvement observés)

$w_{i,t}$  : poids de la  $i^{\text{ème}}$  gaussienne à l'instant  $t$ ,

$\mu_{i,t}$  : le vecteur des valeurs moyennes de la  $i^{\text{ème}}$  Gaussienne à l'instant  $t$ ,



$\Sigma_{i,t}$  : la matrice de covariance de la  $i^{\text{ème}}$  Gaussienne à l'instant  $t$ ,

$\eta$  : la fonction de densité de probabilité Gaussienne (normale) donnée par :

$$\eta(\mathbf{X}_t, \boldsymbol{\mu}_{i,t}, \Sigma_{i,t}) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} \cdot e^{-\frac{1}{2}(\mathbf{X}_t - \boldsymbol{\mu})\Sigma^{-1}(\mathbf{X}_t - \boldsymbol{\mu})} \quad (\text{II.4})$$

Où  $n$  est la dimension de  $X$ .

### II.3.4. Propriétés des paramètres du mélange de Gaussiennes :

#### II.3.4.1. Nombre de composantes Gaussiennes $K$ :

Le nombre de Gaussiennes  $K$  du mélange est choisi selon la multi-modalité de la scène, la capacité mémoire, et la puissance de calcul. Des études ont montré que par rapport aux scènes simples, le nombre de Gaussiennes  $K$  pour des scènes complexes doit être grand. [12][25]. La valeur minimale que doit prendre  $K$  est égale à deux.

Un exemple est illustré dans la figure suivante expliquant le choix du nombre de gaussiennes  $K$  pour une scène simple :

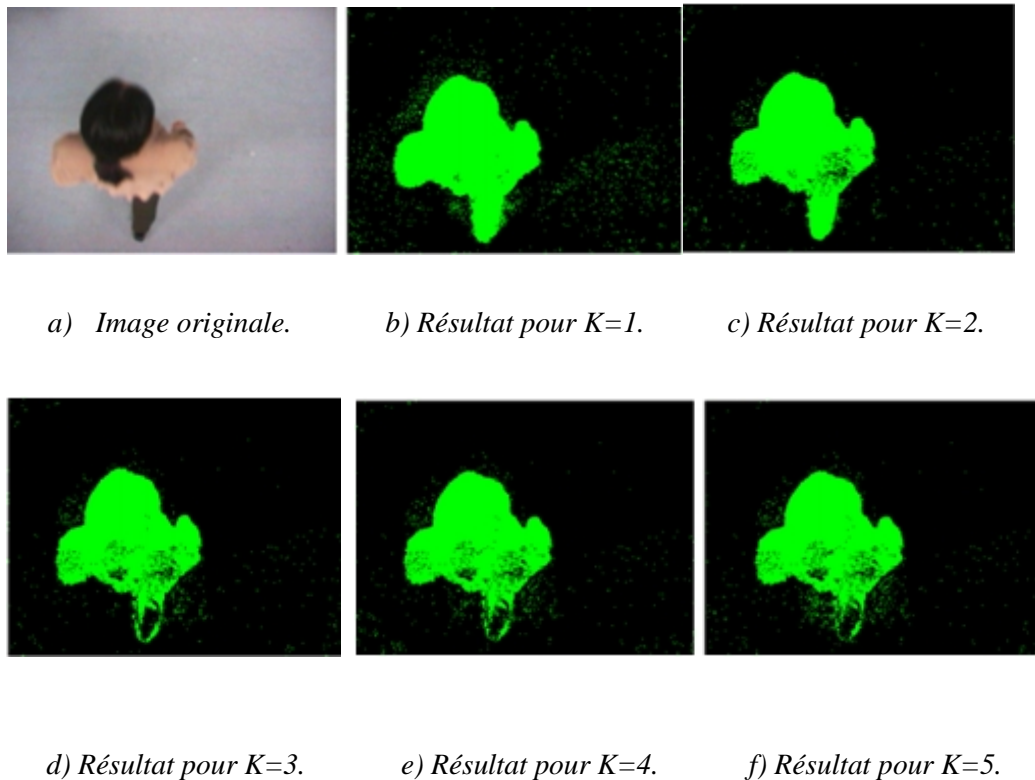
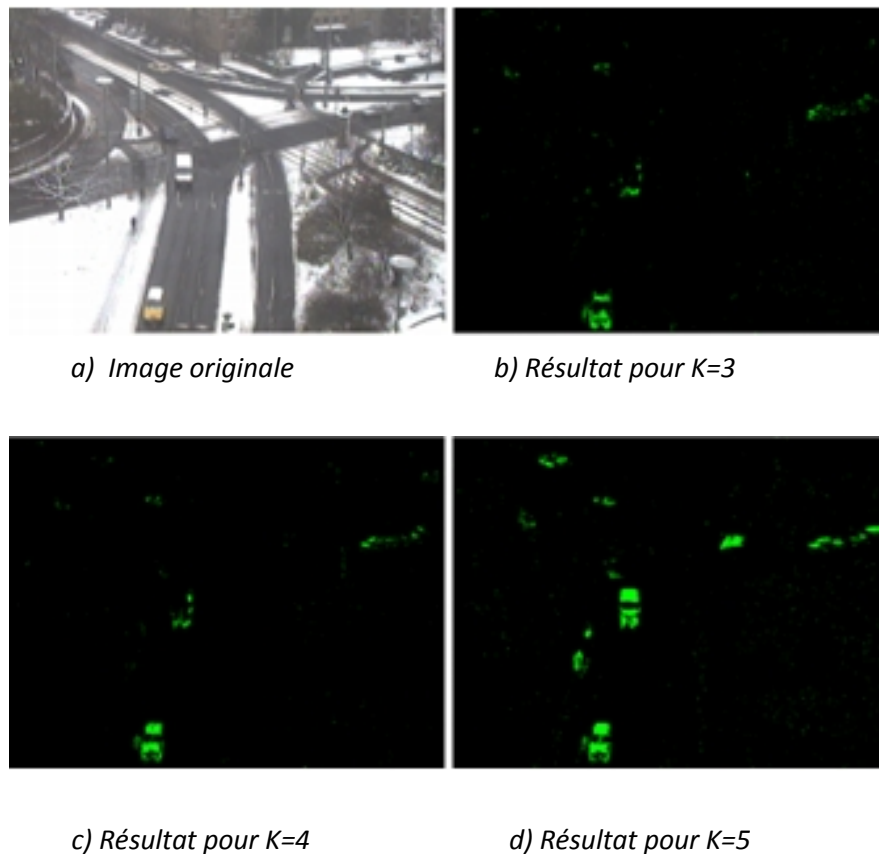


Figure. II.3 : Influence du paramètre  $K$  sur les résultats de détection dans une scène simple.

On remarque que plus le nombre de Gaussiennes  $K$  augmente plus la qualité des résultats se détériore. La détérioration est causée par l'introduction de bruits, secondaires à la mise à jour des paramètres des Gaussiennes. Ce qui confirme le choix du paramètre  $k$  petit (dans cet exemple un  $k$  égale à 1 aurait suffit) pour des scènes simples, et un  $k$  plus important pour des scènes plus complexes.

Un autre exemple est illustré dans la figure suivante qui explique le choix du nombre de Gaussiennes  $K$  pour des scènes complexes :



**Figure. II.4 :** Influence du paramètre  $K$  sur les résultats de détection dans une scène complexe.

Dans une scène complexe où il y a généralement plusieurs modes de mouvement, un choix d'un paramètre  $K$  grand est obligatoire (pour détecter un maximum de modes). Comme le montre cet exemple, on constate que plus le paramètre  $K$  augmente plus de nouveaux modes sont détectés. Dans le résultat pour  $K=5$  (montré dans d)), les régions où les gens marchent sont mieux détectées, les véhicules sont plus clairs et nets. Ce qui confirme que le choix du nombre de gaussiennes  $k$  doit être plus grand pour les scènes complexes.

### II.3.4.2. Matrice de covariance $\Sigma_{i,t}$ : [12]

L'expression de la matrice de covariance prend la forme suivante en supposant que les valeurs des pixels rouges, verts, bleus sont indépendantes et ont les mêmes variances :

$$\Sigma_{i,t} = \sigma_{i,t}^2 \mathbf{I} \quad (\text{II.5})$$

Avec:  $\mathbf{I}$  : la matrice identité de dimension (n X n).

$\sigma_{i,t}^2$  : vecteur ligne des variances de dimension (n)

On peut également écrire la matrice de covariance sous la forme :

$$\Sigma_{i,t} = \begin{pmatrix} \sigma_1^2 & \dots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \dots & \sigma_N^2 \end{pmatrix} \quad (\text{II.6})$$

Cette approximation permet d'éviter un calcul immense lors du calcul de  $\Sigma^{-1}$ .

### II.3.4.3. Poids $w_{i,t}$ : [12]

Chaque  $w_{i,t}$  est associé à la  $i^{\text{ème}}$  gaussienne à l'instant t. Elles représentent les probabilités a priori des différentes classes. Elles sont normalisées selon la relation suivante :

$$w_{i,t} = \frac{w_{i,t}}{\sum_{i=1}^K w_{i,t}} \quad (\text{II.7})$$

$$w_k \in ]0; 1[ \text{ et } \sum_{i=1}^K w_k = 1$$

### II.3.4.4. Moyenne $\mu_{i,t}$ : [12]

Elle représente la valeur moyenne de la  $i^{\text{ème}}$  Gaussienne à l'instant t.

### II.3.5. Processus de détection :

Le processus de détection de mouvement se décompose en trois étapes : l'initialisation des paramètres du mélange, la classification des pixels et enfin la mise à jour des paramètres du mélange selon la classification attribuée.

### II.3.5.1. Initialisation des paramètres :

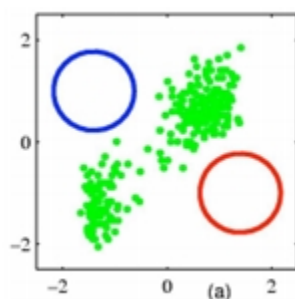
Le nombre de gaussiennes  $K$  est fixé selon Stauffer et Grimson entre 3 et 5, et le reste durant tout le processus.

Le poids  $w$ , la moyenne  $\mu$ , et l'écart type peuvent être initialisés en utilisant la méthode standard EM (Espérance-Maximisation) en maximisant la vraisemblance de manière itérative. Cette méthode est utilisée en cas d'insuffisance des informations des observations ou du manque de données.

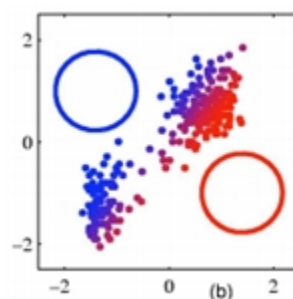
L'initialisation des paramètres (ou l'apprentissage des paramètres) par méthode EM passe par quatre étapes essentielles (figure ci-dessous) :

- étape 1 : initialisation des paramètres  $\mu$ ,  $w$ , pour chaque Gaussienne et évaluation de la valeur log-vraisemblance.
- étape 2 (l'étape-E) : elle utilise l'espérance conditionnelle pour l'estimation des paramètres.
- étape 3 : (l'étape-M) : maximisation de la fonction de vraisemblance sous l'hypothèse que les données manquantes sont connues en utilisant leurs valeurs estimées dans l'étape précédente.
- étape 4 : vérification de convergence des paramètres et log-vraisemblance. Si la convergence n'est pas vérifiée, retour à l'étape 2, et le processus continue jusqu'à la convergence. [12][27][28]

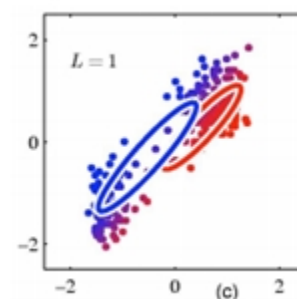
*Les points de données et le modèle de mélange initial*



*Étape initiale E*



*Après première étape M*



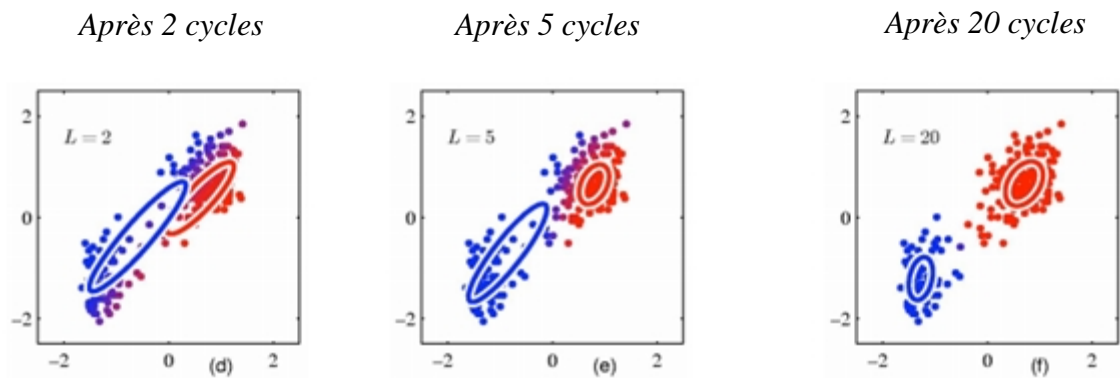


Figure. II.5 : Fonctionnement de l'algorithme EM. [28]

Sans tenir compte des discontinuités dues aux frontières des objets, deux pixels voisins analysés ont de grandes chances d'avoir des vitesses proches. Une solution est donc d'initialiser les paramètres du pixel à estimer avec les paramètres du pixel précédent. Ceci assure que dans la plupart des cas, les paramètres à estimer soient choisis à proximité des paramètres optimaux. [27]

Cette méthode présente une satisfaction automatique aux contraintes et rapidité de convergence. Par contre, l'implémentation de l'algorithme EM est coûteuse. Ainsi les variations de luminance et l'ajout (ou soustraction) des objets a/de l'arrière-plan supposent moins d'indépendance sur les observations. Pour éviter ces contraintes, la technique d'approximation K-moyennes est apparue comme extension de l'algorithme EM (Espérance-maximisation). [12][23]

### II.3.5.2. Classification des pixels :

Après initialisation des paramètres, une première détection des objets mobiles est possible, en classant les pixels comme appartenant à l'arrière-plan ou l'avant-plan (objets en mouvement).

Pour cela les K Gaussiennes du mélange sont ordonnées selon la valeur du rapport  $r_j = w_j / z_j$ . Puis, l'arrière-plan sera modélisé par les B premières distributions Gaussiennes, alors que les autres distributions représentent les objets en mouvement. Le nombre de Gaussiennes B est déterminé selon la relation suivante :

$$B = \operatorname{argmin}_b \left( \sum_{i=1}^b w_{i,t} > T \right) \quad (\text{II.8})$$

Où  $T$  est un seuil de décision entre les distributions du fond et celles des objets en mouvement. Sa valeur dépend de la complexité de la scène. Il est recommandé de choisir une petite valeur de  $T$  pour les scènes simples, et une grande valeur de  $T$  pour les scènes complexes (ayant plusieurs modes de mouvement).

Un pixel du fond correspond à un grand poids avec une faible variance du fait que le fond est plus présent que les objets en mouvements et que sa valeur est pratiquement constante. De plus, les mouvements inintéressants (mouvement de feuilles et branches d'arbre, ou autre...) seront aussi pris comme faisant partie du fond en les classant grâce à des seuils. Ainsi, le fond correspond à une ou plusieurs gaussiennes parmi les  $B$  distributions Gaussiennes de l'arrière-plan.

Les  $B$  distributions Gaussiennes du fond seront ordonnées dans une liste dont les distributions les plus probables restent au sommet tandis que les distributions les moins probables gravitent, et peuvent éventuellement être remplacées par de nouvelles distributions. [12]

A l'arrivée d'une nouvelle trame, à l'instant  $t+1$ , chaque pixel subit un test d'identification pour le classer comme appartenant à l'arrière-plan ou à un objet en mouvement, en vérifiant l'inégalité suivante :

$$SQRT ((X_{t+1} - \mu_{i,t})^T \Sigma_{i,t}^{-1} (X_{t+1} - \mu_{i,t})) > k \quad (II.9)$$

Où  $X_{t+1}$  est la valeur du pixel dans la trame à l'instant  $t+1$ , et  $k$  un seuil constant déterminé empiriquement (égale à 2,5).

Si l'inégalité précédente est vérifiée pour toutes les  $K$  distributions Gaussiennes, le pixel appartient alors au premier plan (objet en mouvement). Le graphe suivant illustre un exemple de 3 Gaussiennes où le pixel introduit ne correspond à aucune d'elles :

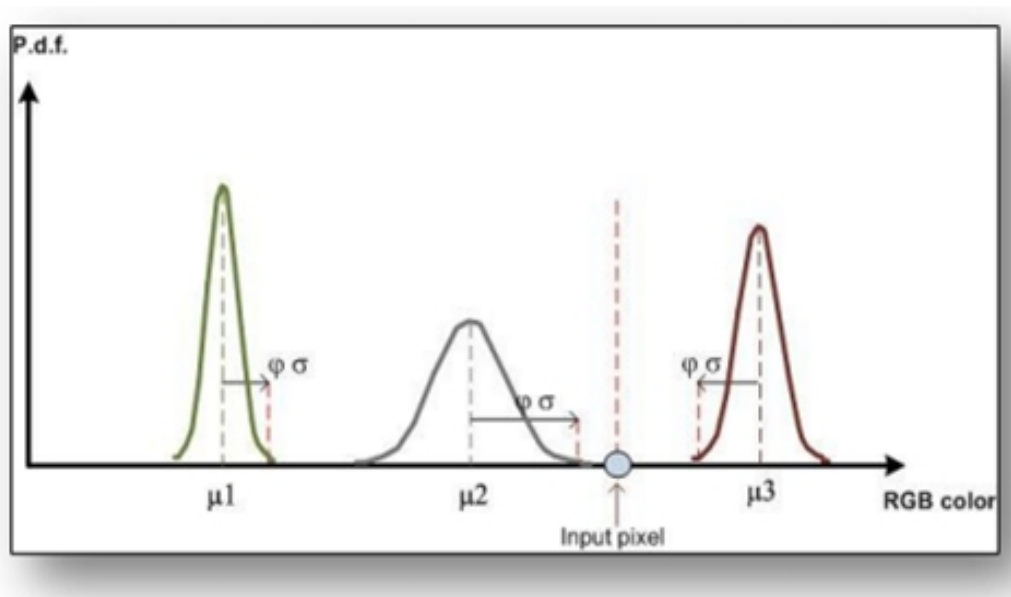


Figure II.6 : Détection de pixels d'avant-plan par la méthode de mélange de Gaussiennes

On remarque que la valeur du pixel introduite ne correspond à aucune des trois distributions existantes, ce qui implique qu'il correspond à l'un des objets en mouvement

En revanche, si l'inégalité précédente n'est pas vérifiée, alors le pixel correspond à l'une des  $K$  distributions Gaussiennes, d'où l'observation de l'un des deux cas suivants :

- Cas 1 : le pixel correspond à l'une des  $B$  distributions Gaussiennes, donc il appartient à l'arrière-plan.
- Cas 2 : le pixel correspond à l'une des autres distributions, donc il appartient au premier-plan (objet mobile).

Après avoir effectué la première détection des objets en mouvement, on passe à la dernière phase du processus : la mise à jour des paramètres. [12][26]

### II.3.5.3. Mise à jour des paramètres :

La mise à jour des paramètres nécessite un algorithme de mise à jour efficace et rapide. Bien que l'algorithme EM soit assez performant, Stauffer et Grimson se sont basés sur l'approximation K-means, dont la convergence est plus rapide.

La mise à jour des paramètres est effectuée selon qu'il y ait ou pas correspondance des pixels avec les distributions Gaussiennes :

1. S'il y a correspondance avec l'une des K Gaussiennes, la mise à jour est effectuée sur chacun des paramètres de la Gaussienne correspondante comme suit :

Le poids  $w_{i,t}$  devient :  $w_{i,t+1} = (1 - \epsilon) w_{i,t} + \epsilon w_{i,t+1}$  (II.10)

Avec  $\epsilon \in [0,1]$  un taux constant qui détermine la vitesse de changement des paramètres. Sa valeur est prédéfinie empiriquement à l'aide des séquences de test, et en fonction de la rapidité de changement du fond. Ainsi, un fond qui ne change que lentement nécessite une valeur faible de  $\epsilon$ , alors qu'un fond qui change rapidement nécessite une valeur  $\epsilon$  large.

La figure suivante illustre un exemple qui montre l'effet des différentes valeurs du taux  $\epsilon$  sur les résultats de détection :

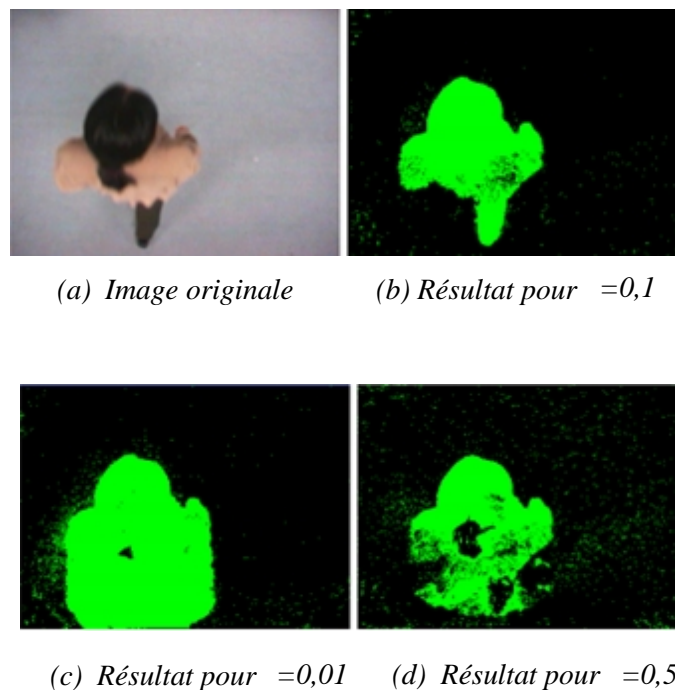


Figure. II.7 : Influence du paramètre  $\epsilon$  sur les résultats de détection

Si on observe l'image originale (a), le mouvement de la personne étant relativement simple (ni rapide ni lent), on pourra déduire que le fond varie modérément. Pour un taux  $\epsilon = 0,01$  (image (c)) on constate une déformation de la zone de détection (en vert) due à la faible fréquence de mise à jour des paramètres, et pour un taux  $\epsilon = 0,5$  une déformation est aussi constatée (image (d)) mais cette fois-ci, due à la forte fréquence de mise à jour des paramètres. Les meilleurs résultats de détection ont été obtenus pour un taux  $\epsilon = 0,1$  (image (b)) qui est due à une adaptation entre la fréquence de variation de l'arrière-plan (le fond) et la fréquence de mise à jour des paramètres.



La moyenne  $\mu_{i,t}$  et la variance  $\sigma_{i,t}^2$  deviennent :

$$\mu_{i,t+1} = (I - \alpha) \mu_{i,t} + \alpha X_{t+1} \quad (\text{II.11})$$

$$\sigma_{i,t+1}^2 = (I - \alpha) \sigma_{i,t}^2 + \alpha (X_{t+1} - \mu_{i,t+1})(X_{t+1} - \mu_{i,t+1})^T \quad (\text{II.12})$$

Avec un taux de mise à jour de la moyenne et de la variance, égale à :

$$\alpha = \frac{\sigma_b}{\sigma_b + \sigma_{i,t}} \quad (\text{II.13})$$

Pour les autres distributions dont il n'y avait pas de correspondance, la moyenne  $\mu$  et la variance  $\sigma^2$  restent inchangés, seul le poids  $w_{i,t}$  devient :

$$w_{i,t+1} = (I - \alpha) w_{i,t} \quad (\text{II.14})$$

2. S'il n'y a aucune correspondance avec les K Gaussiennes, alors la distribution la moins probable du rang K est remplacée par une autre distribution ayant comme paramètres :

$w_{i,t+1}$  = Le plus faible des poids précédents.

$$\mu_{i,t+1} = X_{t+1} \quad (\text{II.15})$$

$\sigma_{i,t+1}^2$  = La plus grande des variances initiales.

Après avoir établi l'étape de mise à jour, la prochaine détection des objets mobiles peut avoir lieu, et le cycle de détection continue. [12][26]

### II.3.6. Discussion, et contraintes :

#### II.3.6.1. Avantages :

La méthode de mélange de Gaussiennes présente plusieurs avantages :

- elle répond aux insuffisances de détection éprouvées par les méthodes citées pour les scènes complexes ayant plusieurs modes de mouvement ;
- elle présente une meilleure adaptabilité aux changements graduels de luminosité, et permet une meilleure réactivité face aux changements rapides de luminance ;

- elle s'adapte au passage du mode objet en mouvement au mode arrière-plan et vice versa. Lorsqu'un objet en mouvement s'arrête de bouger pour une certaine durée il devient part de l'arrière-plan, et s'il bouge encore après, il retourne à l'état objet mobile ;
- elle permet d'enregistrer les derniers modèles du fond. En effet, quand un objet devient part du fond, l'ancien modèle du fond n'est pas détruit et reste dans le mélange, et si l'objet bouge, elle fait appel à la distribution décrivant l'ancien fond ;
- elle présente un gain en termes de mémoire, car les paramètres du modèle peuvent être mis à jour sans avoir à sauvegarder un large buffer des trames vidéo ;

### II.3.6.2. Inconvénients :

En dépit des avantages décrits ci-dessus, cette méthode a aussi des inconvénients :

- elle se base sur le principe que toutes les distributions du fond et des objets sont Gaussiennes, alors qu'en pratique ce n'est pas toujours le cas. Il a été montré que les distributions d'une scène d'intérieur ont plus de chance d'être Laplaciennes que Gaussiennes. Ainsi, l'intensité d'un pixel n'obéit pas à une loi Gaussienne quand elle change brusquement ;
- elle est limitée par sa modalité, car elle consiste à fixer le nombre de Gaussiennes  $K$  pour tous les pixels. Alors que la multi-modalité varie temporellement et spatialement. Ajoutant que les fonds présentant des variations rapides de hautes fréquence ne peuvent pas être modélisés d'une façon précise avec seulement quelques Gaussiennes (3 à 5) ;
- elle nécessite des séries de test des trames sans objets en mouvement pour l'initialisation des paramètres des  $K$  distributions Gaussiennes, ce qui n'est pas facilement réalisable pour certaines scènes ;
- elle est très sensible aux changements soudains de luminosité. En effet, si une scène reste stationnaire pendant une longue période de temps, les variances des composantes du fond deviennent très petites, et un changement brusque de luminosité peut causer une fausse détection en considérant toute la trame comme objet mobile ;

## II.4. Améliorations apportées à la méthode de mélange de Gaussiennes :

### II.4.1. Amélioration de Lee : [12][29]

Afin d'aboutir à une solution qui combine rapidité de convergence et adaptabilité temporelle des paramètres, Lee a proposé une amélioration de l'algorithme de Stauffer et Grimson. Il a introduit une nouvelle variable  $c_k$  pour compter le nombre d'observations effectives des Gaussiennes  $G_k$ , et calculer son taux d'apprentissage  $\alpha_n$  selon l'expression suivante :

$$\alpha_n = (1 - \alpha_n) / c_k + \alpha_n \quad (\text{II.16})$$

La valeur de  $c_k$  est mise à 1 à chaque fois qu'une nouvelle Gaussienne  $G_k$  est introduite, et incrémenté à chaque fois qu'il y ait correspondance d'une valeur  $x$  à la Gaussienne  $G_k$ :

$$c_k = c_k + 1 \quad (\text{II.17})$$

A part la valeur du taux d'apprentissage qui est changée, les équations de mise à jour restent telles qu'elles sont.

Cette modification a significativement augmenté la vitesse de convergence et la précision du modèle avec aucun effet adverse. La figure suivante illustre une comparaison de calcul du log-vraisemblance entre la méthode du mélange de Gaussiennes standard et celle améliorée par Lee durant une minute ;

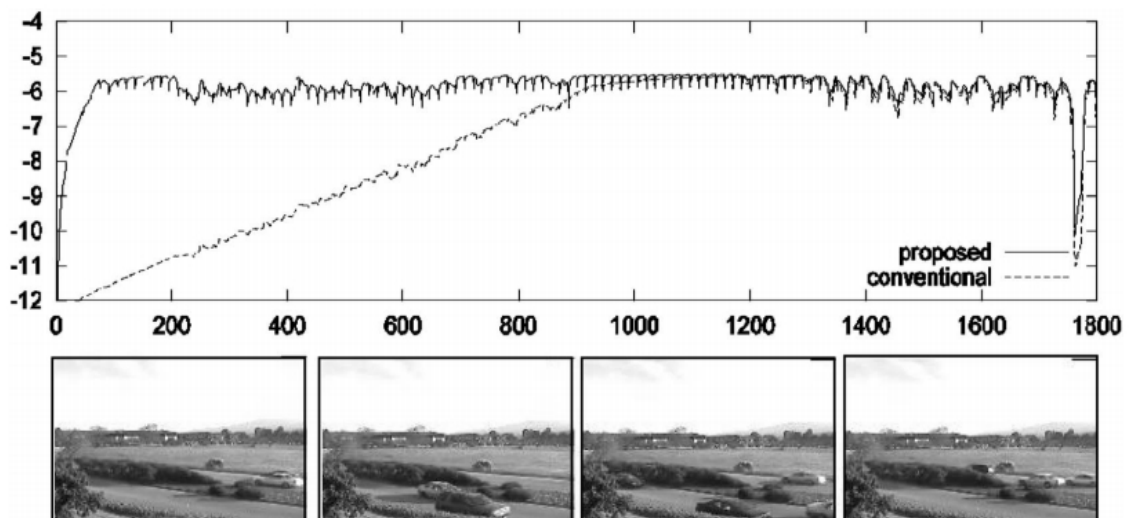


Figure II.8 : Comparaison de vitesse de convergence entre la méthode standard et améliorée.

On remarque d'après les variations du log-vraisemblance de l'image que le modèle proposé par Lee converge très rapidement (en moins de 100 trames), comparée à celle du modèle standard converge en un temps énorme (plus de 900 trames).

La détection d'objets mobiles est basée sur la probabilité conditionnelle qu'un pixel  $x$  à l'instant  $t$  appartient au  $B$  distributions Gaussiennes du fond  $P(B/x)$ . Le calcul de  $P(B/x)$  est effectué en se basant sur le théorème des probabilités totales qui consiste que pour tout évènement  $B$  d'un système complet d'évènements (soit les  $K$  composantes Gaussiennes ( $G_k$ )) de probabilités non nulles :

$$P(B/x) = \sum_1^K P(B/G_k) P(G_k/x) \quad (\text{II.18})$$

$P(G_k/x)$  est la probabilité que le pixel  $x$  appartient à l'une des distributions Gaussiennes du mélange :

$$\begin{aligned} P(G_k/x) &= \frac{P(x|G_k)P(G_k)}{P(x)} \\ &= \frac{w_k \eta_k(x, \mu_k, \sigma_k)}{\sum_1^K w_k \eta_k(x, \mu_k, \sigma_k)} \end{aligned}$$

Finalement :

$$P(B|x) = \sum_{k=1}^K P(B|G_k)P(G_k|x) = \frac{\sum_{k=1}^K P(x|G_k)P(G_k)P(B|G_k)}{\sum_{k=1}^K P(x|G_k)P(G_k)} \quad (\text{II.19})$$

Ou:  $P(x|G_k) = \eta_k(x, \mu_k, \sigma_k)$ .

$$P(G_k) = w_k.$$

Dans le modèle standard,  $P(B/x)$  est égale à 1 pour les Gaussiennes faisant partie du fond, et à 0 pour les autres Gaussiennes. Pour cette méthode proposée par Lee, une approximation de  $P(B|x)$  a été appliquée en utilisant une fonction sigmoïde sur  $w/$  :

$$\hat{P}(B|G_k) = f(w_k/\sigma_k; a, b) = 1/(1 + e^{-a \cdot w_k/\sigma_k + b}) \quad (\text{II.20})$$

Une fois  $P(x)$ , et  $P(B/x)$  estimés, les régions des objets en mouvement sont caractérisées par des pixels dont :

$$P(B/x) < 0,5 \quad (II.21)$$

#### II.4.2. Méthode T2-FGMM : [30]

Cette méthode répond aussi aux insuffisances de la modélisation par mélange de Gaussiennes standard qui initialise ses paramètres en utilisant des séquences pouvant être bruitées et/ou insuffisantes pour modéliser correctement l'arrière-plan. Cette technique consiste à introduire des descriptions d'améliorations des paramètres incertains de la GMM tout en maintenant la même procédure générale de la GMM standard.

##### II.4.2.1.Principe :

Chaque pixel est caractérisé par son intensité dans l'espace RGB. Donc, l'observation  $o$  (du pixel) est un vecteur  $X_t$  de dimension  $d=3$  dans l'espace RGB. D'où, comme on a vu précédemment, la GMM est composé de  $K$  mélanges Gaussiennes multi-variables :

$$P(o) = \sum_{i=1}^k w_i \eta(o, \mu_i, \Sigma_i) \quad (II.22)$$

Zeng et al ont proposé un type de MF (Membership Function) pour représenter les Gaussiennes multi-variables avec un incertain vecteur moyenne ou matrice de covariance seulement, et remplacer les parts correspondantes dans l'équation précédente pour construire T2 FGMM avec un incertain vecteur moyenne (T2 FGMM-UM) ou incertaine variance (T2 GMM-UV).

Alors, avec incertain vecteur moyenne, on définit la T2 FGMM-UM avec :

$$\eta(o, \tilde{\mu}, \Sigma) = \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma|^{\frac{1}{2}}} \exp\left[-\frac{1}{2} \left(\frac{o_1 - u_1}{\sigma_1}\right)^2\right] \dots \exp\left[-\frac{1}{2} \left(\frac{o_d - u_d}{\sigma_d}\right)^2\right] \quad (II.23)$$

$$\mu_1 \in [\underline{\mu}_1, \bar{\mu}_1] \dots \mu_d \in [\underline{\mu}_d, \bar{\mu}_d]$$

En remplaçant  $d$  par 3, on peut exprimer l'équation précédente sous la forme :

$$\eta(X_t, \tilde{\mu}, \Sigma) = \frac{1}{(2\pi)^{\frac{3}{2}} |\Sigma|^{\frac{1}{2}}} \exp\left(\prod -\frac{1}{2} \left(\frac{X_{t,c} - \mu_c}{\sigma_c}\right)^2\right) \quad (II.24)$$

$$\tilde{\mu}_c \in [\underline{\mu}_c, \bar{\mu}_c] \text{ and } c \in \{R, G, B\}$$

Dans le cas d'une incertaine matrice de variance, on définit la T2 FGMM-UV avec :

$$\eta(\mathbf{o}, \mu, \tilde{\Sigma}) = \frac{1}{(2\pi)^2 |\tilde{\Sigma}|^2} \exp\left[-\frac{1}{2} \left(\frac{o_1 - u_1}{\sigma_1}\right)^2\right] \dots \exp\left[-\frac{1}{2} \left(\frac{o_d - u_d}{\sigma_d}\right)^2\right] \quad (\text{II.25})$$

$$\sigma_1 \in [\underline{\sigma}_1, \bar{\sigma}_1] \dots \sigma_d \in [\underline{\sigma}_d, \bar{\sigma}_d]$$

Dont on peut l'exprimer aussi sous la forme :

$$\eta(\mathbf{X}_t, \mu, \tilde{\Sigma}) = \frac{1}{(2\pi)^{\frac{3}{2}} |\tilde{\Sigma}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2} \left(\frac{X_{t,c} - \mu_c}{\sigma_c}\right)^2\right) \quad (\text{II.26})$$

$$\tilde{\sigma}_c \in [\underline{\sigma}_c, \bar{\sigma}_c] \text{ and } c \in \{\mathbf{R}, \mathbf{G}, \mathbf{B}\}$$

$\tilde{\mu}$  et  $\tilde{\Sigma}$  représentent les incertains vecteur moyenne et matrice covariance, Zeng et al assument que la moyenne et la déviation standard varient dans des intervalles avec des possibilités uniformes,  $\mu \in [\underline{\mu}, \bar{\mu}]$  ou  $\sigma \in [\underline{\sigma}, \bar{\sigma}]$ . Chaque composante exponentielle dans les équations précédentes est la fonction la plus prioritaire du MF avec une incertaine moyenne ou déviation standard, comme illustré dans les deux figures suivantes :

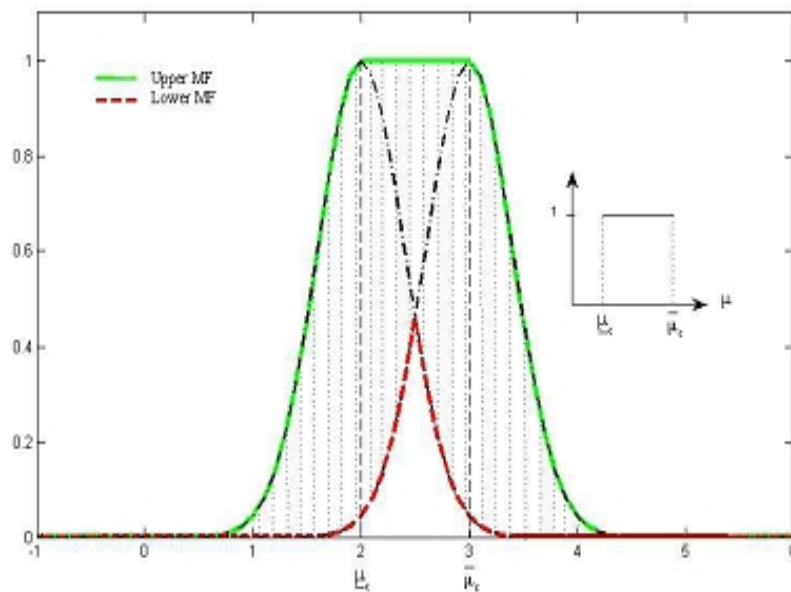


Figure. II.9 : La Gaussienne (primary MF) avec incertaine moyenne.

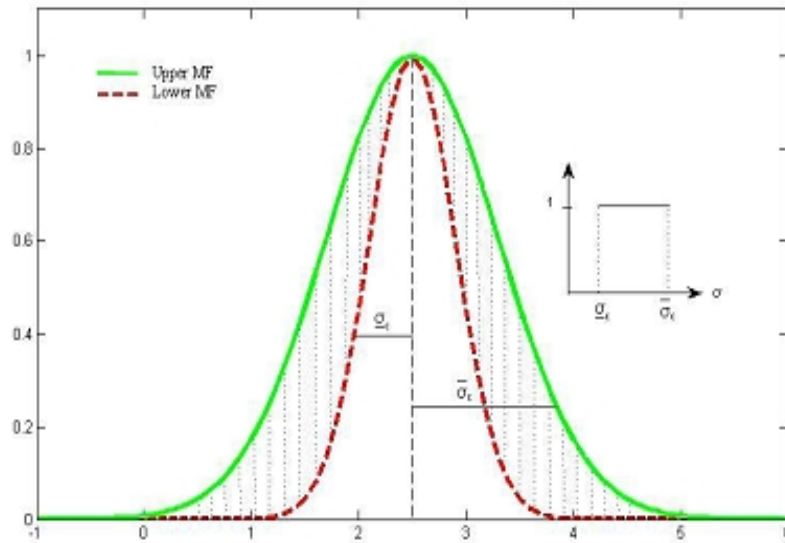


Figure. II.10 : La Gaussienne (primary MF) avec incertaine déviation standard.

Pour les deux figures, la ligne continue et la ligne discontinue représente la plus haute et la plus basse MF respectivement, et la région entre les deux représente l'intervalle des valeurs incertaines. Avec une incertaine moyenne, la plus haute MF est :

$$\begin{aligned} \bar{h}(o) &= f(o; \underline{\mu}; \sigma) & \text{if } o < \underline{\mu} \\ \bar{h}(o) &= 1 & \text{if } \underline{\mu} \leq o \leq \bar{\mu} \\ \bar{h}(o) &= f(o; \bar{\mu}; \sigma) & \text{if } o > \bar{\mu} \end{aligned} \quad (\text{II.27})$$

Avec:

$$\begin{aligned} f(o; \underline{\mu}; \sigma) &= \exp - \frac{1}{2} \left( \frac{o - \underline{\mu}}{\sigma} \right)^2 \\ f(o; \bar{\mu}; \sigma) &= \exp - \frac{1}{2} \left( \frac{o - \bar{\mu}}{\sigma} \right)^2 \end{aligned}$$

La plus basse MF est :

$$\underline{h}(o) = f(o; \bar{\mu}; \sigma) \quad \text{if } o = \frac{\mu + \bar{\mu}}{2}$$

$$\underline{h}(\mathbf{o}) = f(\mathbf{o}; \underline{\mu}; \underline{\sigma}) \quad \text{if } \mathbf{o} > \frac{\underline{\mu} + \bar{\mu}}{2}$$

En parallèle, avec une incertaine déviation standard, la plus haute MF est :

$$\underline{h}(\mathbf{o}) = f(\mathbf{o}; \underline{\mu}; \bar{\sigma})$$

Et la plus basse MF est :

$$\bar{h}(\mathbf{o}) = f(\mathbf{o}; \bar{\mu}; \underline{\sigma})$$

Les paramètres moyenne et déviation standard varient dans des intervalles correspondant selon les paramètres de contrôle  $k_m$  et  $k_v$  respectivement comme suite :

$$\begin{aligned} \underline{\mu} &= \mu - k_m \sigma, \quad \bar{\mu} = \mu + k_m \sigma \\ \underline{\sigma} &= k_v \sigma, \quad \bar{\sigma} = (1/k_v) \sigma \end{aligned} \quad (\text{II.28})$$

Zeng et al ont pris en considération qu'une Gaussienne présente 99,7% de sa probabilité dans l'intervalle  $[\mu-3, \mu+3]$  pour assumer que  $k_m \in [0, 3]$  et  $k_v \in [0.3, 1]$ , ces facteurs contrôlent aussi la région comprise entre la plus haute et la plus basse MF. Plus  $k_m$  ou  $k_v$  est grand, plus la région est large, et donc plus il y'aura de valeurs incertaines.

#### II.4.2.2. Détection d'objets en mouvement, mise à jour des paramètres :

D'une façon identique à celle de la GMM standard, la détection d'objets en mouvement consiste à classifier les pixels courants comme arrière-plan ou objet mobile en utilisant le rapport  $r_j = w_j / j$ , les B distribution Gaussiennes vérifient une certaine condition à l'intermédiaire d'un seuil T sont retenus pour l'arrière-plan :

$$B = \underset{b}{\operatorname{argmin}} ( \sum_{i=1}^b w_{i,t} > T ) \quad (\text{II.29})$$

Les autres distributions sont considérées représenter les objets mobiles. A l'arrivée d'une nouvelle trame à l'instant t+1, un test de correspondance est appliqué pour chaque pixel. Et cela selon la valeur :

$$H(\mathbf{X}_t) = | \ln(\underline{h}(\mathbf{X}_t)) - \ln(\bar{h}(\mathbf{X}_t)) | \quad (\text{II.30})$$

La Gaussienne avec la plus grand priorité MF avec une incertaine moyenne est :



$$H(X_t) = \begin{cases} \frac{2k_m |X_t - \mu|}{\sigma} & \text{if } X_t \leq \mu - k_m \sigma \text{ or } X_t \geq \mu + k_m \sigma \\ \frac{|X_t - \mu|}{2\sigma^2} + \frac{k_m |X_t - \mu|}{\sigma} + \frac{k_m^2}{2} & \text{if } \mu - k_m \sigma < X_t < \mu + k_m \sigma \end{cases} \quad (\text{II.31})$$

Et la Gaussienne avec la plus grand priorité MF avec une incertaine déviation standard est :

$$H(X_t) = \left( \frac{1}{1/k_v^2 - k_v^2} \right) \frac{|X_t - \mu|^2}{2\sigma^2} \quad (\text{II.32})$$

Avec,  $\mu$  et  $\sigma$  sont la moyenne et la déviation standard originale respectivement.

Si  $X_t$  se dévie loin du centre de distribution, il aura une large valeur de  $H(X_t)$  qui indique une grande incertitude au modèle. Donc un pixel correspond à une Gaussienne si :  $H(X_t) < k$ , ou  $k$  est une constante de seuil déterminé expérimentalement égale a 2,5.

Ainsi, la suite du processus de détection est équivalente à celle de la GMM standard, et de même pour la mise à jour des paramètres.

Les deux améliorations T2-FGMM-UM et T2-FGMM-UV peuvent être utilisées pour modéliser l'arrière-plan. La technique T2-FGMM-UM est plus robuste que celle de T2-FGMM-UV.

L'estimation des paramètres de T2-FGMM comprend trois étapes :

- étape 1 : choix du nombre de Gaussiennes  $K$  du mélange entre 3 et 5 ;
- étape 2 : estimation des paramètres du modèle de mélange de Gaussiennes en utilisant l'algorithme EM ;
- étape 3 : ajout du facteur  $k_m$  ou  $k_v$  pour aboutir à un modèle de mélange de Gaussiennes amélioré T2-FGMM-UM ou T2-FGMM-UV ;

La figure suivante illustre une comparaison entre les performances d'une GMM standard et celles de GMM améliorés T2-FGMM-UM et T2-FGMM-UV.

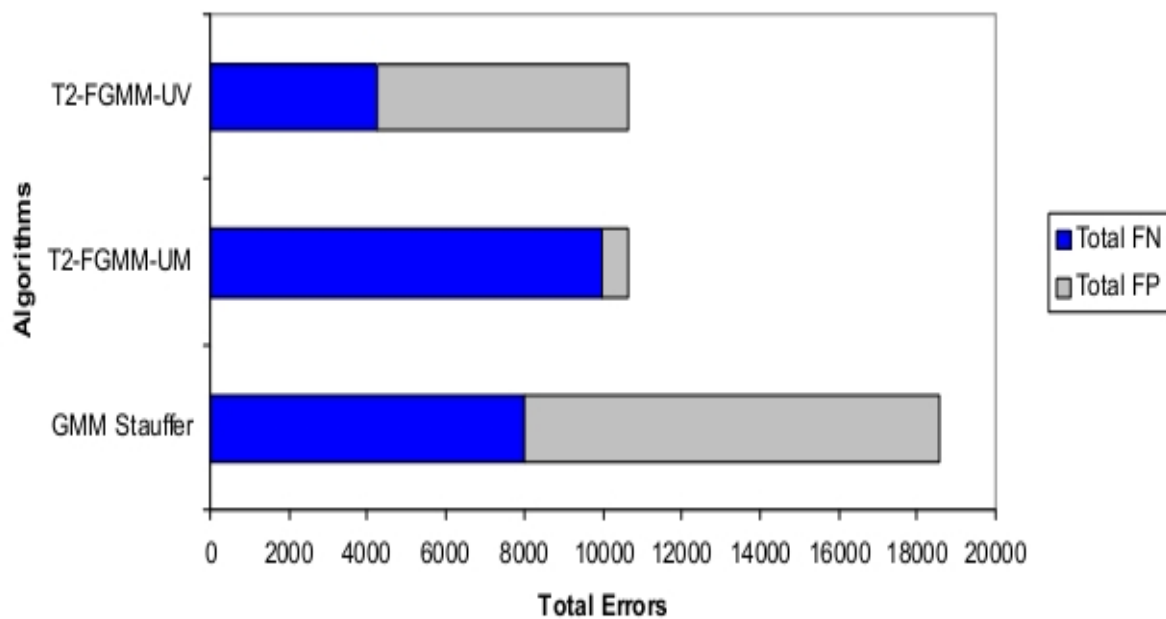


Figure. II.11 : Comparaison des performances entre la technique standard (GMM Stauffer) et améliorée (T2-FGMM- UV et UM).

On remarque que les deux techniques T2-FGMM-UM et T2-FGMM-UV présentent moins d'erreurs que celle du GMM standard. Ainsi, T2-FGMM-UM présente le moins de fausses détections positives, ce qui la rend plus efficace que T2-FGMM-UV.

## II.5. Conclusion :

Dans ce chapitre, une étude théorique a été faite sur la détection de mouvement par mélange de Gaussiennes, en commençant dans une première étape par la méthode originale, puis pour finir dans une seconde étape par développer les différentes améliorations apportées

# *Chapitre III*

---

*Réalisation, tests et résultats*

### **III.1. Introduction :**

Nous présenterons dans cette dernière partie, une architecture générale que nous détaillerons par la suite chaque module, tout en développant les différents algorithmes ainsi que l'environnement de travail et de développement utilisé (MatLab). Pour terminer, nous réaliserons une série de tests en faisant varier les différents paramètres : Alpha, K et le seuil T et nous discuterons les résultats obtenus.

### **III.2. Réalisation et mise en œuvre :**

#### **III.2.1. Environnement de travail et de développement :**

Nous avons choisi d'utiliser le logiciel MatLab comme outil de développement pour nos simulations. En effet, malgré sa lenteur, nous avons estimé que cette dernière était compensée par la facilité de programmation et d'élaboration d'algorithmes ainsi que par son aspect graphique riche.

Avant toute chose, il est nécessaire de définir quelques concepts fondamentaux que l'on va manipuler par la suite. Commençons tout d'abord par définir le concept d'image :

Une image est la représentation par ordinateur de la réalité physique, plus précisément, une représentation numérique. Dans ce contexte, on peut donc dire qu'une image numérique est un tableau (une matrice multidimensionnelle) de dimensions finies, composé de pixels. Chaque pixel est caractérisé par :

- sa position spatiale dans la matrice (ou tableau) ;
- sa valeur numérique qui représente sa valeur spectrale codée sur 8bits (comprise entre 0 et 255) ; [34]

MatLab permet de prendre facilement en main les outils de traitement d'images. C'est un logiciel de calcul scientifique permettant de développer des solutions à des problèmes techniques. Il permet de réaliser des calculs numériques, tracer des graphiques pour visualiser et analyser les données. Il dispose d'un langage et d'un environnement de programmation interactifs. MatLab est associé à des boîtes à outils appelés « Toolbox » permettant d'accéder à des fonctions spécifiques et prédisposées. [35]

En traitement d'images, les deux outils « Image acquisition toolbox » et « Image processing toolbox » fournissent toutes les fonctions nécessaires à l'acquisition de données, et

à effectuer différentes opérations de traitement d'images telles : le filtrage, la convolution, affichage et lecture d'une image ...

MatLab peut lire des images codées sur 8, 16, 24 ou 32 bits. Mais le stockage de ces données ne peut être fait qu'avec trois types de variables :

- le type uint8 (entier non signé de 8bits) de plage [0 255] ;
- le type uint16 (entier non signé de 16bits) de plage [0 65535] ;
- le type double (entier de 2bits) de plage [0 1] ;

Par défaut, les fonctions sous MatLab étant faites pour traiter des matrices de type double, il peut être nécessaire de convertir les données de la matrice image, codées en type uint8/uint16, en type double avant de les utiliser avec certaines fonctions. Dans le cas contraire, un warning sera retourné. [36]

La figure III.1 montre l'interface générale de travail utilisée par MatLab, composée :

- d'une fenêtre d'édition des commandes (Command Window) ;
- de deux fenêtres contenant un onglet de visualisation de l'espace des variables (Workspace), un onglet de visualisation des fichiers du répertoire de travail (Current Directory) et un onglet de visualisation de l'historique des commandes (Command History) ;
- d'une fenêtre d'édition avec un onglet permettant de visualiser le contenu de fichiers (Editor) et un onglet permettant de visualiser le contenu des variables (Array Editor) ;

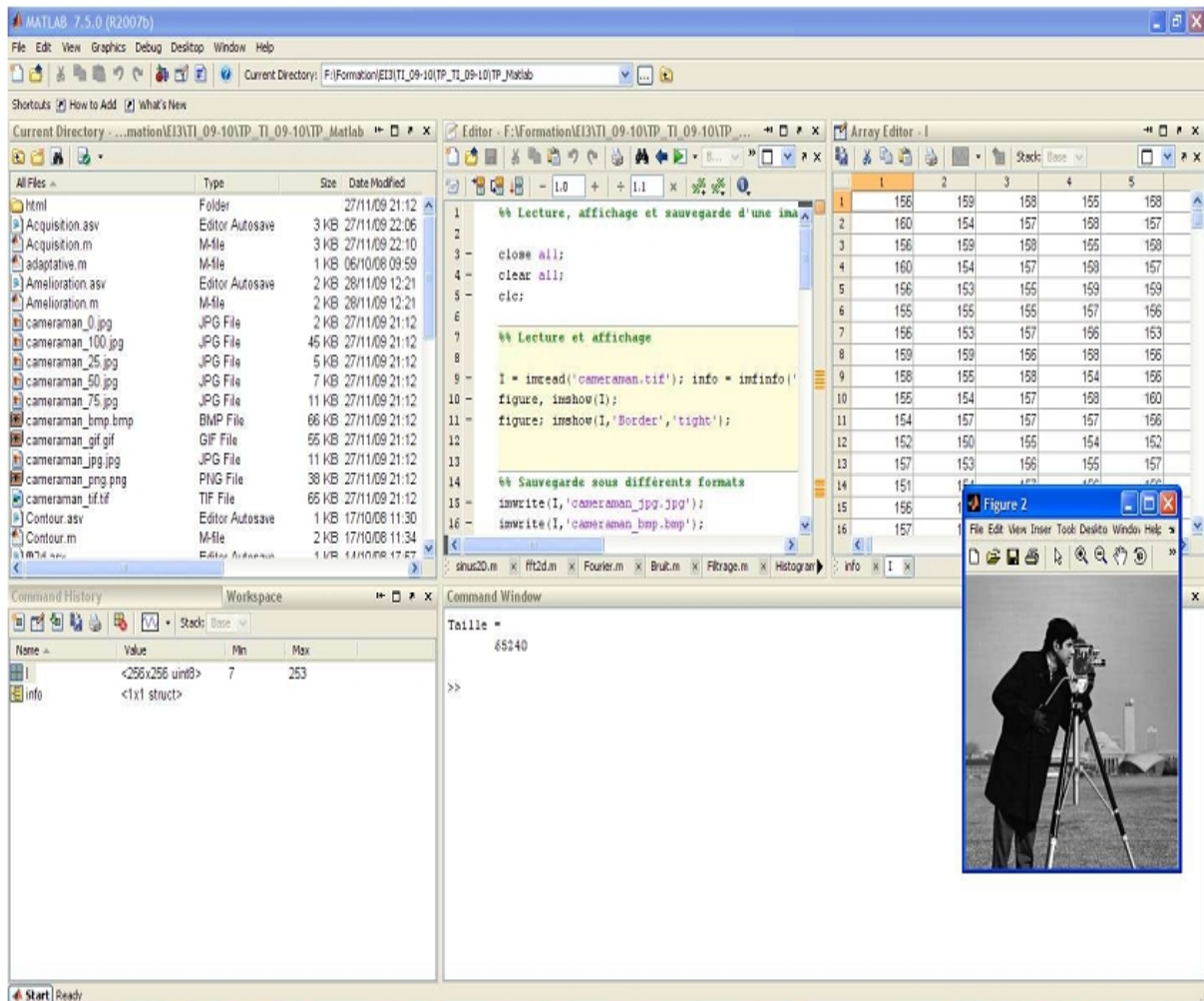


Figure III.1 : Logiciel scientifique de calcul MatLab. [35]

### III.2.2. Architecture générale de détection :

L'étape la plus cruciale de notre travail repose sur la phase de détection car, c'est une étape déterminante de la qualité des résultats. La détérioration de la qualité est causée principalement :

- soit par une mauvaise initialisation des paramètres initiaux des K distributions Gaussiennes ;
- soit par des bruits produits lors de l'acquisition de la séquence d'images ;

Nous pouvons procéder avant la phase de détection (appelée aussi phase de pré-traitement) à un filtrage Gaussien pour remédier aux bruits d'acquisition. Contrairement à ces bruits, une bonne initialisation des paramètres des K Gaussiennes peut se faire uniquement à l'aide d'un algorithme puissant « Kmeans ». [37]

L'architecture générale de détection que nous avons utilisé se décompose comme suit :

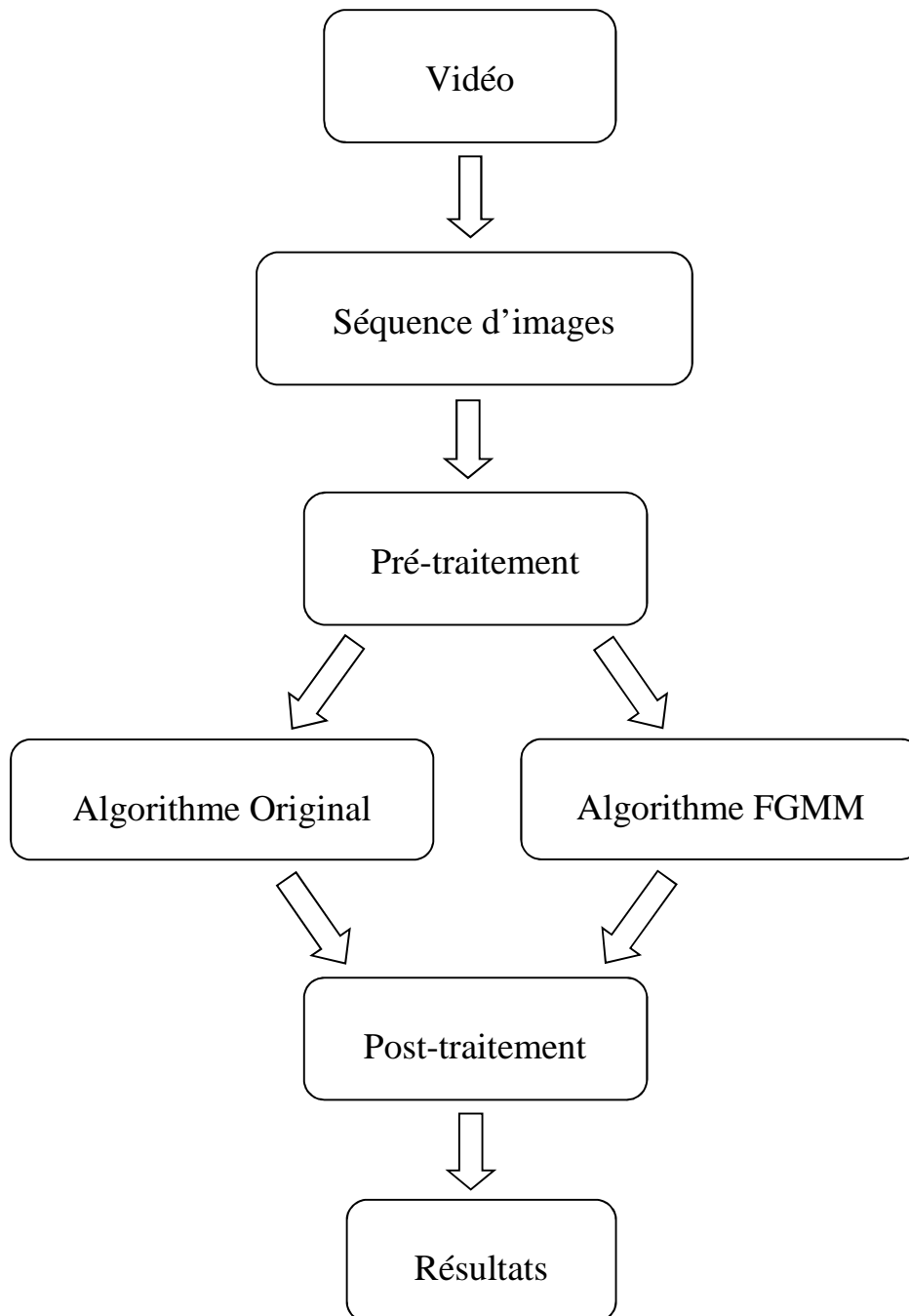


Figure .III.2 : L'architecture générale de détection utilisée.

Comme indiqué dans (Fig .III.2), notre travail est structuré en 5 étapes importantes :

- Etape d'acquisition : acquisition de la séquence vidéo, de taille réduite de préférence ;

- Etape de segmentation : segmentation de la séquence vidéo en images ;
- Etape de pré-traitement : réduction de bruits en appliquant un filtre Gaussien. Cependant, laisser les images bruitées telles quelles, permettrait de mettre les deux méthodes à rude épreuve (c'est-à-dire juger de la qualité et de la capacité de chacune à s'adapter aux différents scénarios) ;
- Etape de détection : par la méthode de détection standard (de Stauffer et Grimson) et par une amélioration de cette dernière (de Zeng) ;
- Etape de post-traitement : consiste à appliquer des filtres morphologiques pour éliminer toute fausse détection tout en raffinant les formes des objets en mouvement ;
- Etape d'affichage des résultats : résultats de détection sous forme d'images binaires (1 pour caractériser un objet mobile et 0 pour un objet de fond), et le fond estimé par une image en niveau de gris ;

### III.2.3. Mise en œuvre :

Nous illustrerons pour chaque méthode de détection, un algorithme, comprenant les différentes phases d'initialisation, classification et mise à jour des pixels d'une part, et les phases d'acquisition, filtrage (prétraitement, post-traitement) et affichage des résultats d'autre part.

#### III.2.3.1. Acquisition vidéo :

La détection de mouvement se fera sur des séquences d'images obtenues soit à partir d'une caméra vidéo fixe (une CAM) (en utilisant la toolbox « Image acquisition toolbox »), ou à partir de séquences prédisposées (MatLab, ou téléchargées).

L'acquisition d'une vidéo à travers une caméra fixe, se fera grâce à l'interface « Imaqtool », faisant comme interface (objet) entre la caméra et l'outil de développement MatLab. Avec cette interface, nous pourrions régler différents paramètres tels : la résolution, nombre de frames, espace de couleur (RGB, gris, ...), ...

Il est utile de choisir dans notre cas une séquence vidéo de faible résolution (160\*120), pour valoriser la vitesse d'exécution du programme. Les vidéos acquises seront enregistrées par la suite dans le dossier d'exécution de Matlab avec les autres séquences vidéo téléchargées.



### III.2.3.2. Segmentation et pré-traitement :

Dans cette phase, nous décomposerons la séquence vidéo en images, puis suivra la phase de pré-traitement, consistant à appliquer des filtres correcteurs de bruits. Par la suite, les images produites seront enregistrées dans un dossier appelé « Séquence\_images ». Ces images seront lues et représentées par la donnée « *Image\_data\_3D* ».

### III.2.3.3. Détection :

Avant d'entamer la présentation des deux algorithmes de détection, faisons tout d'abord une présentation des différentes données utilisées.

#### III.2.3.3.1. Présentation, déclaration des données :

Les données utilisées en traitement d'images sont généralement des tableaux à 3D ou à 4D représentant des images en niveau de gris et couleurs respectivement, des indices, des structures, ... Nous présenterons dans cette section uniquement les données jugées à la fois importantes et nécessaires à la compréhension des deux méthodes de détection mises en œuvre.

---

**Soient les données « images » suivantes :**

*Image\_data\_3D(D, C, T) : image à 3D, contenant l'intensité de chaque pixel, chaque couleur et de chaque image.*

*Image\_détection\_3D(D, C, T) : image à 3D, contenant les résultats de détection en binaire de chaque pixel, chaque couleur et de chaque image.*

*Image\_détection\_4D(Height, Width, C, T) : image à 4D, servant à l'affichage des résultats de détection en binaire.*

*Images\_fond\_3D(D, C, T) : image à 3D, contenant les résultats de l'estimation du fond pour chaque pixel, chaque couleur et pour chaque image.*

*Image\_fond\_4D(Height, Width, C, T) : image à 4D, servant à l'affichage du fond estimé.*

**Soient les paramètres et indices suivants :**

*D* : dimension de l'image (Height x Width).

*Height* : hauteur de l'image.

*Width* : largeur de l'image.

*C* : nombre de couleurs (3 couleurs en général).

*T* : nombre d'images.

*K* : nombre de Gaussiennes fixé entre 3 et 5.

*MEAN(D, C, K)* : tableau à 3D, contenant les moyennes de chaque pixel, chaque couleur et de chaque distribution Gaussienne.

*VAR(D, C, K)* : tableau à 3D, contenant les variances de chaque pixel, chaque couleur et de chaque distribution Gaussienne.

*ALPHA* : vitesse d'adaptation des poids.

*RHO* : vitesse d'adaptation des moyennes et des variances.

*Background\_seuil* : seuil pour la détermination de *B* (Gaussiennes prétendantes au fond, compris entre 1 et *K*).

*Seuil\_déviatiion\_standard* : seuil de déviation standard, une condition de correspondance du pixel *d* avec les *K* distributions Gaussiennes

---

Ces paramètres concernent à la fois la méthode standard de Stauffer et Grimson et l'améliorée de Zeng.

### **III.2.3.3.2. Algorithme GMM standard :**

---

*//////////////////// Initialisation des paramètres //////////////////////*

**Initialisation des poids :**

**Pour *d* allant de 1 à *D***

**Pour *c* allant de 1 à *C***

$WEIGHT(D, C, 1) = 1$       % Attribuer à la première Gaussienne de chaque pixel et pour chaque couleur un poids égale à 1.

**Pour k allant de 2 à K**

$WEIGHT(d, c, k) = 0$       % Mettre les poids des Gaussiennes restantes à zéro.

**Fin pour**

**Fin pour**

**Fin pour**

**Initialisation des variances :**

**Pour d allant de 1 à D**

**Pour c allant de 1 à C**

**Pour k allant de 1 à K**

$VAR(d, c, k) = VAR\_INITIALE$       % Attribuer une variance initiale (VAR\_INITIALE) à chaque pixel, pour chaque couleur et pour chaque Gaussienne.

**Fin pour**

**Fin pour**

**Fin pour**

**Initialisation des moyennes :**

**Pour d allant de 1 à D**

**Pour c allant de 1 à C**

$MEAN(d, c, 1) = Image\_data\_3D(d, c, 1)$       % Attribuer à la première Gaussienne de chaque pixel et pour chaque couleur l'intensité lumineuse de la première image comme moyenne.

**Pour k allant de 2 à K**

$MEAN(d, c, k) = \text{valeur aléatoire (comprise entre 0 et 255)}$

**Fin pour**

**Fin pour**

**Fin pour**

//////////////////// Classification des pixels //////////////////////

**Pour t allant de 1 à T**

**Pour d allant de 1 à D**

**Pour c allant de 1 à C**

**Pour k allant de 1 à K**

$Image\_data\_centrée ( ; ; k) = |Image\_data\_3D( ; ; t) - MEAN( ; ; k)|$

$Déviati\text{on\_carrée} ( ; ; k) = (Image\_data\_centrée( ; ; k) .^2) ./ VAR( ; ; k)$  % Calcul de la distance de Mahalobis

**Fin pour**

$Gaus\_identif = Zeros (D, C, K)$

Vérification de la correspondance du pixel d, couleur c avec l'une des K Gaussiennes.

$M\_data = \min (Déviati\text{on\_carrée} (d, c, :))$  % Attribuer à M\_data la plus petite déviation carrée

**Si M\_data < Seuil\_déviati\text{on\\_standard}**

- Mettre des 1 dans Gaus\_identif dans les cases qui correspondent à M\_data et qui vérifie la condition précédente. Sinon, laisser ces cases à zéro.

- Trouver les indices de Gaus\_identif qui correspondent à 1 et les mettre dans Indice\_minimum\_déviati\text{on\\_carrée}.

**Fin si**

$Gaus\_nnidentif = zeros (D, C, K)$  % Matrice qui va être remplis dans l'étape de mise à jour pour le cas de non correspondance du pixel d, couleur c avec les K distributions Gaussiennes.

$Gauss\_active = zeros (D, C, K)$

$Gaus\_active = Gaus\_identif + Gaus\_nnidentif$

Trier les matrices Gaus\_active, WEIGHT et Indice\_minimum\_déviati\text{on\\_carrée} selon le rapport WEIGHT / VAR.

Trouver les premières B\_gaussienne qui satisferont pour chaque pixel et pour chaque couleur :

$B\_gaussienne = argmin_b(\sum_{i=1}^b WEIGHT (d, c, i) > Background\_seuil).$

Marquer les b Gaussiennes correspondantes à la plus petite somme des poids comme Gaussiennes du fond.

**Pour k allant de 1 à b**

$Backgroud\_gaussienne (d, c, k) = 1$

**Fin pour**

Marquer les pixels correspondants aux objets mobiles dans une matrice binaire (1 pour caractériser un objet mobile et 0 pour un objet statique (fond)).

$Background\_gaussienne\_data(d, c, :) = Gaus\_active(d, c, :) \& Background\_gaussienne(d, c, :)$ .

$Image\_objet\_mobile\_data(d, c, :) = not(Background\_gaussienne\_data(d, c, :))$ .

Produire une image binaire (*Image\_détection\_3D*) contenant les résultats de détection de l'image *t* : 1 si le pixel correspond à un objet en mouvement sinon, 0 pour un objet appartenant au fond.

**Si Indice\_minimum\_déviatiion\_carrée (d, c) non nul**

$Image\_data\_3D(d, c, t) = Image\_objet\_mobile\_data(d, c, Indice\_minimum\_déviatiion\_carrée(d, c))$

**Fin si**

Produire une image en niveau de gris (*Image\_fond\_3D*) contenant les résultats de l'estimation du fond

**Si Indice\_minimum\_déviatiion\_carrée (d, c) et Background\_Gaussienne\_data(d, c, Indice\_minimum\_déviatiion\_carrée (d, c)) non nuls**

$Image\_fond\_3D(d, c, t) = Image\_data\_3D(d, c, t)$

**Fin si**

//////////////////// Mise à jour des paramètres //////////////////////

**Mise à jour du poids :**

$WEIGHT = (1 - ALPHA) * WEIGHT + ALPHA * Gaus\_identif$  % Mise à jour des poids dans le cas où il y a correspondance.

**Mise à jour de la moyenne et de la variance :**

**Pour d allant de 1 à D**

**Pour c allant de 1 à C**

Pour le cas où il y a correspondance du pixel *d*, couleur *c* avec l'une des *K* distributions Gaussiennes, une mise à jour de cette dernière est faite comme suit :

**Si Indice\_minimum\_déviatiion\_carree (d, c) non nul**

$MEAN(d, c, Indice\_minimum\_deviation\_carree(d, c)) = (1 - RHO) * MEAN(d, c, Indice\_minimum\_deviation\_carree(d, c)) + (RHO * Image\_data\_3D(d, c, t));$

$VAR(d, c, Indice\_minimum\_deviation\_carre(d, c)) = (1-RHO) .* VAR(d, c, Indice\_minimum\_deviation\_carre(d, c)) + (RHO .* (Image\_data\_3D(d, c, t) - MEAN(d, c, Indice\_minimum\_deviation\_carre(d, c)))) .^2;$

**Fin si**

Pour le cas où il n'y a pas correspondance du pixel  $d$ , couleur  $c$  avec l'une des  $K$  distributions Gaussiennes, la dernière Gaussienne (de poids le plus faible) sera remplacée est mise à jour comme suit :

**Si Gaus\_identif( $d, c, :$ ) nul**

$Gaus\_nnidentif(d, c, indice) = 1$

$MEAN(d, c, indice) = Image\_data\_3D(d, c, t)$  % Mettre à jour la moyenne de la nouvelle Gaussienne.

$VAR(d, c, indice) = VAR\_INITIALE$  % Mettre à jour la variance de la nouvelle Gaussienne

$WEIGHT(d, c, indice) = min(WEIGHT(d, c, :))$  % Mettre à jour le poids de la nouvelle Gaussienne en lui donnant le plus faible poids des  $K$  distributions

**Fin si**

**Fin pour**

**Fin pour**

**Fin pour**

---

### **Discussion :**

L'algorithme présenté précédemment se décompose en trois étapes essentielles : Initialisation des paramètres des Gaussiennes, classification des pixels et mise à jour des paramètres.

#### **- L'initialisation des paramètres des Gaussiennes :**

C'est l'étape la plus cruciale. En effet, une bonne initialisation permettrait d'obtenir de meilleurs résultats de détection et donc limiter le nombre de fausses détections. Cette étape consiste à définir les  $K$  distributions Gaussiennes pour chaque pixel  $d$ , couleur  $c$  selon la séquence vidéo utilisée : moyenne, variance et poids.

### **Remarque :**

Pour éviter toute confusion, la décomposition de chaque pixel en  $K$  distributions Gaussiennes ne sera pas forcément la même avec les autres pixels. En effet, des pixels peuvent appartenir au fond pendant toute la séquence alors que d'autres non (passage de voiture, piéton, ...). Le paramètre  $K$  étant fixe pour tous les pixels, seuls donc les paramètres moyenne, variance et poids varient à la fois pour chaque distribution et aussi pour chaque pixel.

#### **- Classification des pixels :**

Lors de la phase de classification des pixels, chaque pixel  $d$ , couleur  $c$  de l'image  $t$ , doit subir une série de tests :

- on vérifie s'il y'a correspondance du pixel  $d$ , couleur  $c$  avec l'une des  $K$  distributions Gaussiennes ;
- si la correspondance est vérifiée, une classification est faite selon le rapport poids/variance. Cette classification permettrait par la suite de classer le pixel comme étant un objet mobile ou statique. Puis, on détermine les  $b$  premières Gaussiennes faisant partie de l'arrière plan, et les  $K-b$  Gaussiennes restantes feront partie de l'avant plan (objet en mouvement). Finalement, on classifie le pixel appartenant à l'arrière plan ou bien à l'avant plan ;
- si la classification n'est pas vérifiée, la Gaussienne la moins probable (de faible poids) sera remplacée par une nouvelle Gaussienne ;

#### **- Mise à jour des paramètres :**

Pour qu'il y'ait adaptation du fond, les paramètres moyenne, variance et poids des  $K$  distributions Gaussiennes doivent être mis à jour à chaque itération. En effet, cette mise à jour est contrôlée grâce à deux paramètres Alpha (la vitesse d'adaptation des poids) et Rho (la vitesse d'adaptation de la moyenne et de la variance). Ces deux paramètres varient en fonction de la séquence vidéo utilisée. Dans notre travail, nous avons pris Alpha et Rho comme constante égale.

**Remarque :**

A chaque itération ( $t = t + 1$ ) une classification des pixels est faite pour chaque pixel et chaque couleur de l'image  $t + 1$ . Puis, une autre mise à jour se fera jusqu'à la fin ou l'arrêt de la séquence vidéo.

**III.2.3.3.3. Algorithme GMM améliorée (FGMM-UM) :**

La méthode améliorée « FGMM-UM » diffère de la méthode standard uniquement dans l'étape de classification des pixels, plus précisément dans l'étape de correspondance du pixel  $d$ , couleur  $c$  avec les  $K$  distributions Gaussiennes.

---

//////////////////// Initialisation des paramètres //////////////////////

**Initialisation des poids :**

Inchangée

**Initialisation des variances :**

Inchangée

**Initialisation des moyennes :**

Inchangée

//////////////////// Classification des pixels //////////////////////

**Pour  $t$  allant de 1 à  $T$**

**Pour  $d$  allant de 1 à  $D$**

**Pour  $c$  allant de 1 à  $C$**

**Pour  $k$  allant de 1 à  $K$**

**Si  $[Image\_data\_3D(d, c, t) \leq MEAN(d, c, k) - Km.*sqrt(VAR(d, c, k))]$  ou  $[Image\_data\_3D(d, c, t) \geq (MEAN(d, c, k) + Km.*sqrt(VAR(d, c, k)))]$**

$$H(d, c, k) = 2.*Km.*|Image\_data\_3D(d, c, t) - MEAN(d, c, k)|./sqrt(VAR(d, c, k))$$

**Sinon**

$$H(d, c, k) = |Image\_data\_3D(d, c, t) - MEAN(d, c, k)|./(2*VAR(d, c, k)) +$$



$Km.*|Image\_data\_3D(d, c, t) - MEAN(d, c, k)|/\sqrt{VAR(d, c, k)} + (Km.^2)/2;$

**Fin si**

**Fin pour**

$Gaus\_identif = Zeros(D, C, K)$

Vérification de la correspondance du pixel  $d$ , couleur  $c$  avec l'une des  $K$  Gaussiennes.

$M\_data = \min(H(d, c, :))$  % Attribuer à  $M\_data$  la plus petite déviation carrée

**Si  $M\_data < Seuil\_déviation\_standard$**

- Mettre des 1 dans  $Gaus\_identif$  dans les cases qui correspondent à  $M\_data$  et qui vérifie la condition précédente. Sinon, laisser ces cases à zéro.

- Trouver les indices de  $Gaus\_identif$  qui correspondent à 1 et les mettre dans  $Indice\_minimum\_déviation\_carrée$ .

**Fin si**

Le reste de l'étape classification des pixels reste inchangée.

//////////////////// Mise à jour des paramètres //////////////////////

Mise à jour du poids :

Inchangée

Mise à jour de la moyenne et de la variance :

Inchangée

**Fin pour**

**Fin pour**

**Fin pour**

---

#### III.2.3.4. Post-traitement :

L'opération de post-traitement consiste à appliquer des filtres morphologiques pour éliminer toute fausse détection causée essentiellement par les bruits lors de l'acquisition, mais aussi pour raffiner les formes des objets mobiles détectés. Nous aurons recours à deux principaux filtres : érosion, dilation. Comme nous pourrons aussi obtenir en combinant ces deux derniers filtres, deux nouveaux filtres très connus : ouverture, fermeture.

### **III.2.3.5. Affichage des résultats :**

Les résultats de détection ainsi que l'image de fond estimée seront stockés dans les fichiers « Résultat\_de\_détection » et « Image\_de\_fond » respectivement.

Pour les résultats de détection, les images stockées ne sont que des images binaires contenant des « 0 » pour représenter un objet de fond en noir et « 1 » pour représenter un objet en mouvement en blanc. Pour l'image de fond estimée, l'image stockée est en niveau de gris, montrant à chaque itération l'effacement des objets en mouvement, laissant après un certain nombre d'itérations uniquement les objets appartenant au fond.

## **III.3. Tests et résultats :**

Après avoir illustré l'algorithme du fonctionnement de la méthode de détection de mouvement par mélange de Gaussiennes standard, ainsi que celle de la méthode améliorée FGMM- UM, nous montrerons ici les résultats obtenus en comparant l'efficacité des deux méthodes en question. Puis, on étudiera l'influence des différents paramètres sur les résultats de détection. Et pour finir, nous ferons une estimation du fond à partir d'une séquence d'un trafic routier.

### **III.3.1. Séquences utilisées :**

Dans les scènes réelles, différents modes de mouvement peuvent être envisagés : mouvement rapide ou lent d'un objet, nombre d'objets mobiles, mouvement des branches d'arbres, ombre, ... etc. Pour balayer les différents modes de mouvement qui peuvent avoir lieu, nous avons utilisé plusieurs séquences d'images pour l'évaluation de notre travail. Les tests ont été effectués sur une machine Pentium Dual-Core, CPU 2,00 GHz avec 3Go de RAM (2,87 Go utilisable).

On utilisera trois séquences qui diffèrent en fonction de la complexité du mouvement :

- une séquence simple réalisée dans notre école constituée de 303 trames, de résolution 128\*96 pixels et à 8 images/sec, illustrant le déplacement de deux personnes (une par une) ;
- une séquence complexe externe constituée de 515 trames, de résolution 176\*144 pixels et à 8 images/seconde, dans laquelle plusieurs objets mobiles se présentent (déplacement de personnes et de voitures, mouvement des feuilles arbres) ;

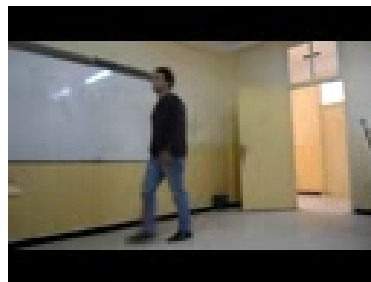
- une séquence complexe constituée de 480 trames, de résolution 128\*96 pixels et à 8 image/sec, illustrant un trafic routier. Cette dernière sera utilisée pour montrer un exemple d'un fond statique estimé ;

### III.3.2. Résultats de détection :

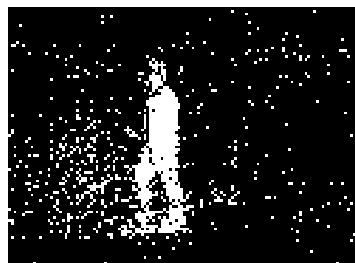
On présentera à partir des deux premières séquences d'images citées ci-dessus (une séquence simple et une autre complexe), les résultats de détection avec et sans filtrage morphologique, obtenus en utilisant les deux algorithmes de détection mis en œuvre.

#### III.3.2.1. Résultats de détection pour une scène simple :

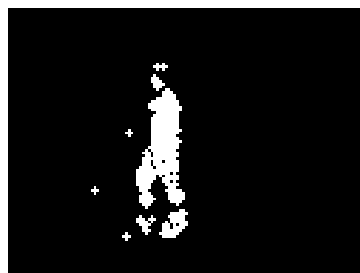
- *GMM standard* :



*Image originale. (107<sup>ème</sup> image de la séquence)*



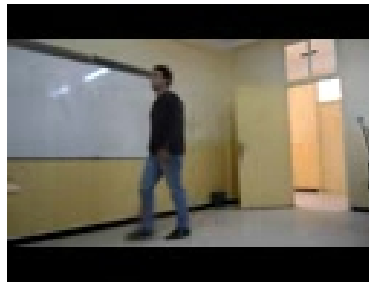
*Détection sans filtrage.*



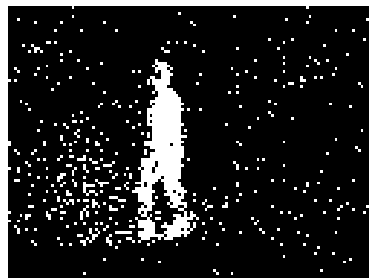
*Détection avec filtrage (d'ouverture).*

**Figure .III.3** : Résultats de détection obtenus en utilisant la méthode de détection de mouvement par mélange de Gaussiennes standard.

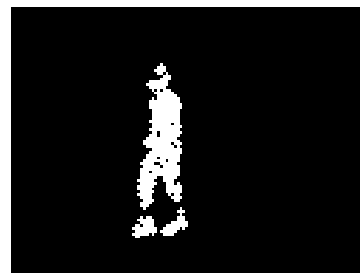
- *GMM améliorée (FGMM-UM) :*



*Image originale. (107<sup>ième</sup> image de la séquence)*



*Détection sans filtrage.*

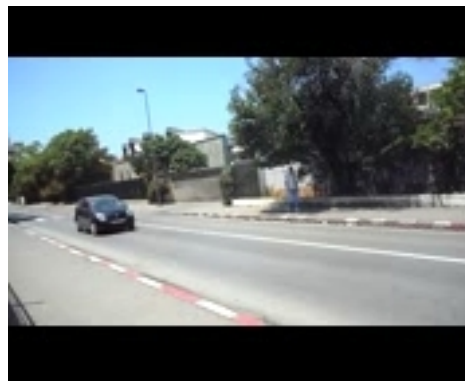


*Détection avec filtrage (d'ouverture).*

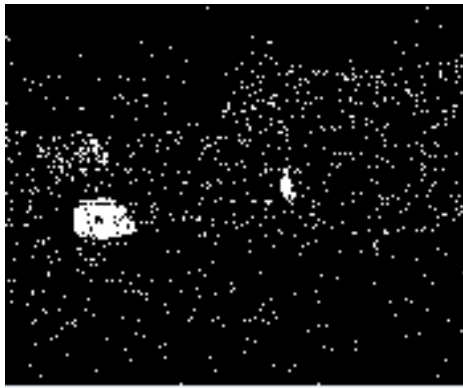
**Figure .III.4 :** Résultats de détection obtenus en utilisant la méthode de détection de mouvement par mélange de Gaussiennes améliorée.

### III.3.2.2. Résultats de détection pour une scène complexe :

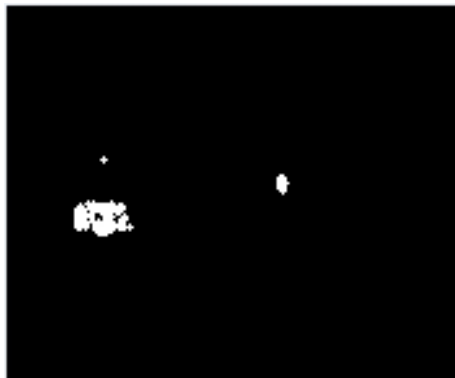
- *GMM standard :*



*Image originale. (230<sup>ième</sup> image de la séquence)*



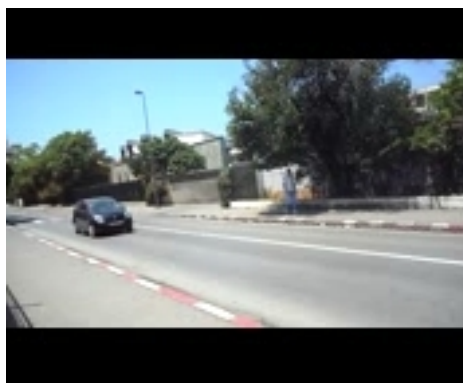
*Détection sans filtrage.*



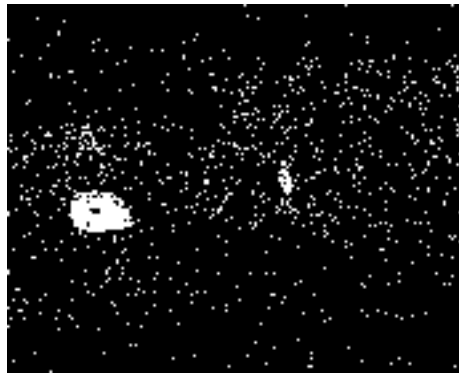
*Détection avec filtrage (d'ouverture).*

**Figure .III.5 :** Résultats de détection obtenus en utilisant la méthode de détection de mouvement par mélange de Gaussiennes standard.

- **GMM améliorée (FGMM-UM) :**



*Image originale. (230<sup>ième</sup> image de la séquence)*



*Détection sans filtrage.*



*Détection avec filtrage (d'ouverture).*

**Figure .III.6 :** Résultats de détection obtenus en utilisant la méthode de détection de mouvement par mélange de Gaussiennes améliorée.

**Discussion :**

Dans les deux cas de scènes présentés, on constate une dégradation des résultats de détection causée par les séquences d'images elles-mêmes, c'est-à-dire par des bruits lors de l'acquisition des deux séquences d'images avec l'appareil photo numérique.

La différence entre les résultats de détection par la méthode standard et améliorée n'est observable qu'après un filtrage morphologique (filtre d'ouverture). Cette opération de filtrage permettra de supprimer les grains blancs (bruit d'acquisition et aussi les mouvements de feuilles d'arbres) tout en mettant en valeur les objets en mouvement.

Dans la scène simple, on constate aussi après filtrage morphologique, une légère déformation de l'objet détecté, causée par l'utilisation d'une séquence de faible résolution (128\*98 pixels). En utilisant une séquence d'images à haute résolution, on obtiendra certes de meilleurs résultats de détection (meilleures formes d'objet, ...) mais par ailleurs, la RAM

utilisée par le système (par MatLab) augmentera, ce qui risque dans notre cas de bloquer le programme lors du traitement de la séquence d'images vu la capacité de notre machine.

### III.3.3. Evaluation des résultats :

Il va sans dire qu'aucune méthode de détection n'est parfaite. Pour évaluer les résultats de détection des deux méthodes mises en œuvre (originale et améliorée), on comparera une image de détection (avec et sans filtrage morphologique) pour chaque méthode avec une image dite de référence, en calculant le nombre de fausses détections que nous verrons dans ce qui va suivre.

#### III.3.3.1. Procédure à suivre :

La procédure pour évaluer les résultats de détection obtenus pour chaque méthode se fera en deux étapes :

- **Étape 1:** Création de l'image de référence

Avant de commencer l'évaluation, on doit tout d'abord créer une image d'une parfaite détection. Cette image dite de référence va contenir le minimum de fausses détections.

Rappelons que deux types de fausses détections peuvent avoir lieu :

- Les faux positifs ( $fp$ ) : lorsque un pixel appartenant au fond est considéré par notre programme comme faisant partie de l'avant plan.
- Les faux négatifs ( $fn$ ) : le contraire, c'est-à-dire lorsque un pixel appartenant à l'avant plan est considéré par le programme comme faisant partie du fond.

Comme le montre la figure ci-dessous, l'image de référence est tirée à partir d'un seuillage de la 107<sup>ième</sup> image de la séquence :



Image originale. (107<sup>ième</sup> image de la séquence)

Image de référence.

**Figure III.7 :** Création d'une image de référence pour l'évaluation des résultats de détection.

- **Étape 2** : Calcul de «  $fn$  » et de «  $fp$  »

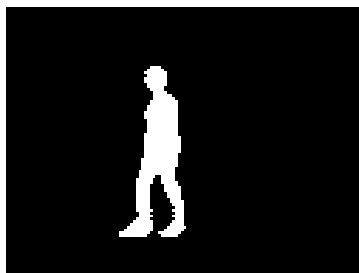
L'image de référence étant créée, on évalue maintenant le résultat de détection obtenus avec et sans filtrage morphologique et ce pour chaque méthode de détection avec l'image de référence, en calculant le nombre de fausses détections «  $fp$  » et «  $fn$  ». Puis, on calcule le taux de fausses détections «  $Taux$  », égal à :

$$Taux = \frac{fp+fn}{D} \quad [8] \quad (III.1)$$

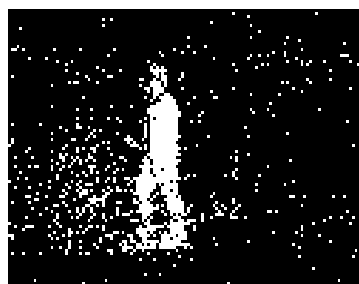
Avec D : la résolution de l'image.

### III.3.3.2. Résultats de l'évaluation :

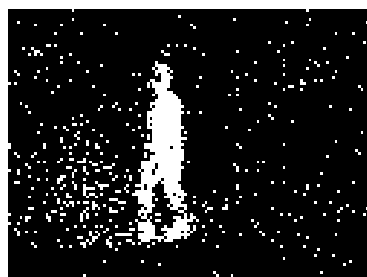
#### III.3.3.2.1. Résultats sans filtrage morphologique :



*Image de référence.*



*Image de détection sans filtrage (GMM standard)*



*Image de détection sans filtrage (FGMM - UM)*



<i>Méthode de détection</i>	<i>Faux positifs</i>	<i>Faux négatifs</i>	<i>Taux</i>
<i>GMM standard</i>	634	121	6.14 %
<i>FGMM – UM</i>	555	99	5.32%

Figure .III.8 : Résultats de l'évaluation sans filtrage morphologique.

**Discussion :**

Avec la méthode de détection standard, le taux de fausses détections est estimé à 6.14%. Par rapport aux fausses détections « faux négatifs », on constate un nombre élevé de fausses détections « faux positifs », cela étant dû au bruit introduit lors de l'acquisition de la séquence d'images (grains blancs).

Par contre, avec la méthode de détection améliorée, le taux de fausses détections est estimé à 5.32%. Comme le montre le tableau ci-dessus, on constate une diminution des fausses détections « faux positifs » par rapport à ceux de la méthode de détection standard. Donc la deuxième méthode est plus précise.

**III.3.3.2.2. Résultats avec filtrage morphologique :**



Image de référence.

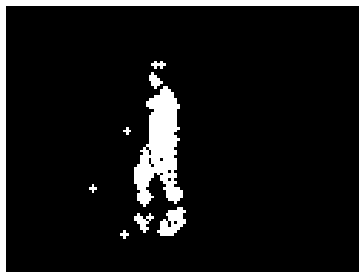


Image de détection avec filtrage (GMM standard)

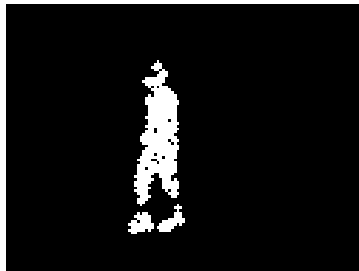


Image de détection avec filtrage (FGMM – UM)

<i>Méthode de détection</i>	<i>Faux positifs</i>	<i>Faux négatifs</i>	<i>Taux</i>
<i>GMM standard</i>	55	176	1.88 %
<i>FGMM – UM</i>	38	130	1.36%

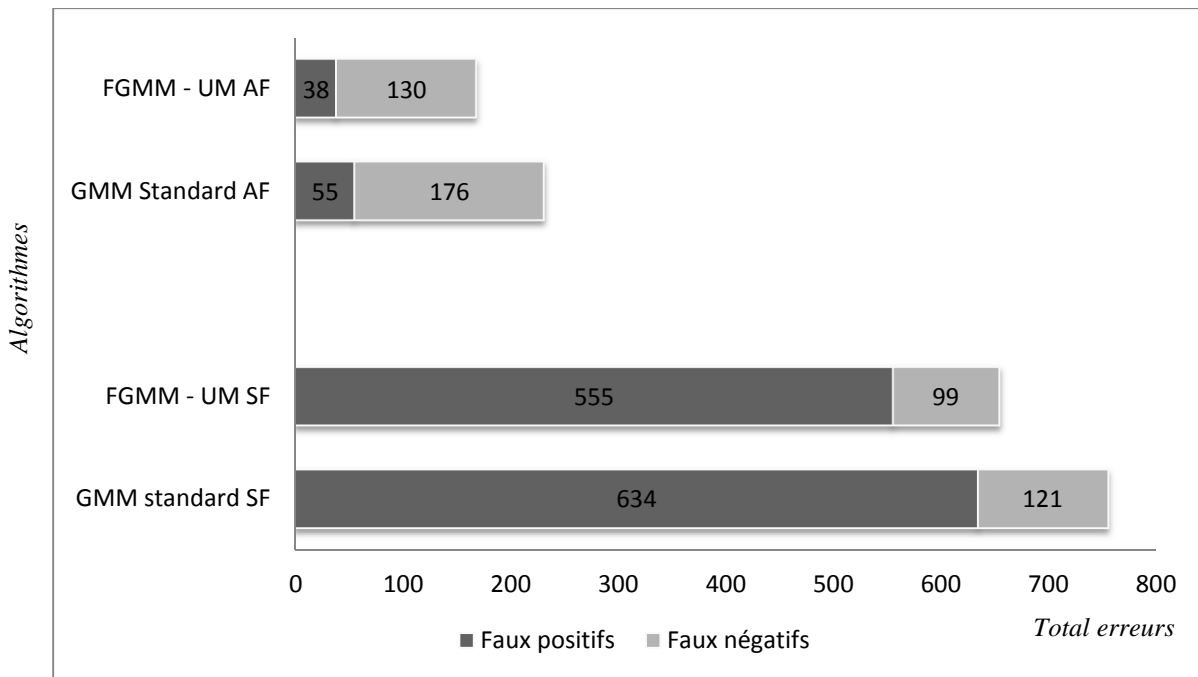
Figure .III.9 : Résultats de l'évaluation avec filtrage morphologique.

**Discussion :**

Avec la méthode de détection standard, le taux de fausses détections est estimé à 1.88%, alors qu'avec la méthode de détection améliorée, le taux de fausses détections est estimé à 1.36 %. L'application d'un filtrage morphologique (filtre d'ouverture) a permis de diminuer considérablement le taux de fausses détections, et cela en réduisant considérablement les fausses détections « faux positifs ».

On constate une augmentation des fausses détections « faux négatifs », ce qui peut s'expliquer par la déformation de la forme de l'objet détecté (grains noirs à l'intérieur de l'objet détecté).

Les résultats de l'évaluation obtenus par la méthode de détection standard et améliorée sont résumés dans la figure ci-dessous :



**Figure .III.10 :** Evaluation des résultats obtenue avec et sans filtrage par les deux méthodes de détection (standard et améliorée).

Avec :

AF : avec filtrage morphologique.

SF : sans filtrage morphologique.

### III.3.4. Influence des différents paramètres sur les résultats de détection :

Lors de l'évaluation des résultats, les paramètres K, ALPHA et RHO étaient choisis expérimentalement. Nous avons essayé de trouver les valeurs optimales pour aboutir à de meilleurs résultats de détection. On verra dans cette section l'influence des différents paramètres sur la qualité de résultats de détection. On commencera par varier le paramètre ALPHA, tout en laissant les autres paramètres inchangés. Puis de la même manière, le paramètre K. Et pour finir, le seuil T.

III.3.4.1. Influence de ALPHA sur les résultats de détection :

- Pour une scène simple :




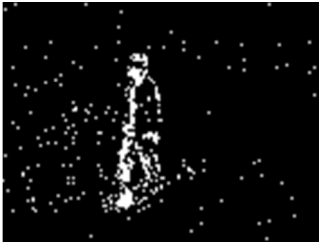

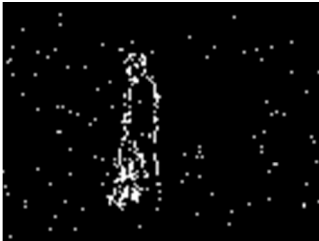

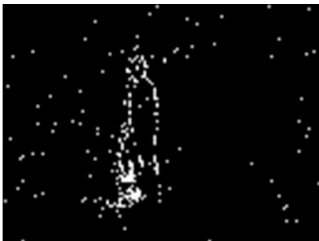
<i>Alpha</i>	<i>Image originale</i>	<i>Image de détection S.F (FGMM - UM)</i>
<i>0.05</i>		
<i>0.24</i>		
<i>0.5</i>		
<i>0.7</i>		

Tableau .III.1 : Influence de ALPHA sur les résultats de détection pour une scène simple.

- Pour une scène complexe :


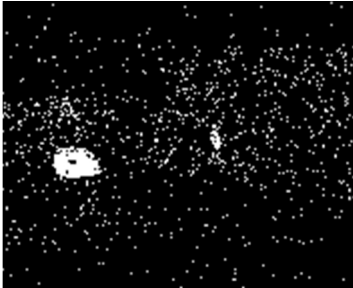

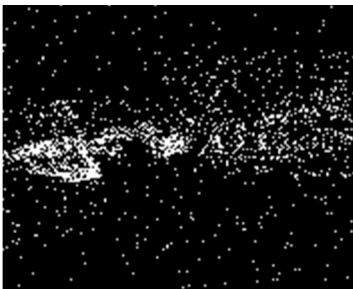

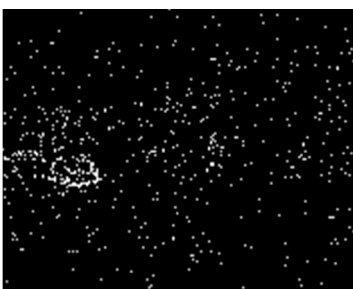

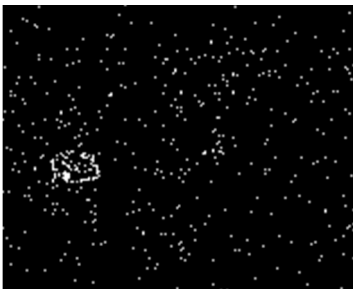
<i>Alpha</i>	<i>Image originale</i>	<i>Image de détection S.F (FGMM - UM)</i>
<i>0.01</i>		
<i>0.24</i>		
<i>0.5</i>		
<i>0.7</i>		

Tableau .III.2 : Influence de ALPHA sur les résultats de détection pour une scène complexe.





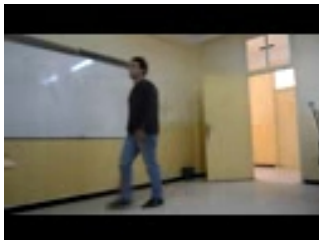

**Discussion :**

Rappelons que le paramètre ALPHA (la vitesse d'adaptation des poids) est fonction de la rapidité de changement du fond. Donc, si pour un fond qui ne change que lentement (ou rapidement respectivement), la vitesse d'adaptation des poids doit être faible (ou large respectivement).

Pour les deux scènes présentées, on constate une atténuation continue de la détection lorsque le paramètre ALPHA augmente car, la fréquence de variation du fond augmente et devient importante par rapport à celle des objets en mouvement.


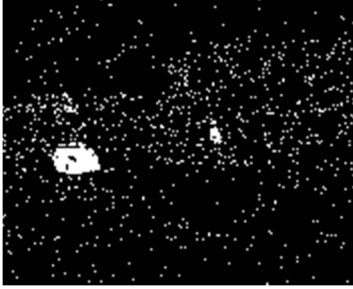

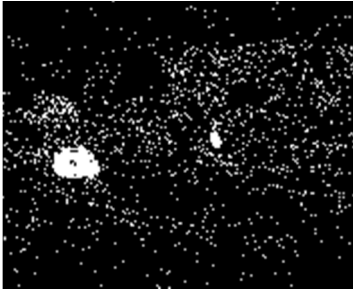

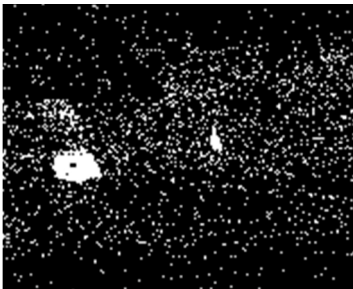
**III.3.4.2. Influence de K sur les résultats de détection :**

- *Pour une scène simple :*

<i>K</i>	<i>Image originale</i>	<i>Image de détection S.F</i> <i>(FGMM - UM)</i>
3		
4		
5		

*Tableau .III.3 : Influence de K sur les résultats de détection pour une scène simple.*

- Pour une scène complexe :

<i>K</i>	<i>Image originale</i>	<i>Image de détection S.F (FGMM - UM)</i>
3		
4		
5		

**Tableau .III.4 :** Influence de K sur les résultats de détection pour une scène complexe.

**Discussion :**

Pour la première et la deuxième séquence, les meilleurs résultats sont obtenus pour K égal à 3 car le nombre de Gaussiennes K dépend du nombre d'objets en mouvement.

Lorsque le nombre de Gaussiennes K augmente, on constate :

- une netteté des objets détectés (maximum de modes de mouvement détectés) ;
- une dégradation des résultats de détection, causée par l'introduction de bruits secondaires à la mise à jour des paramètres des Gaussiennes ;

**III.3.4.3. Influence du seuil T sur les résultats de détection :**

- *Pour une scène simple :*


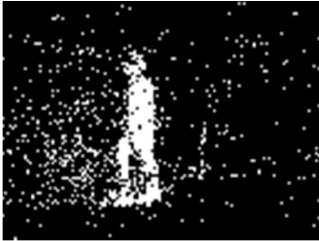

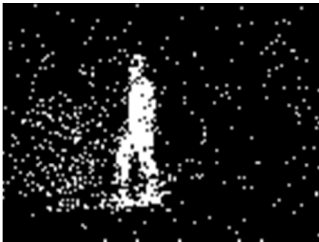

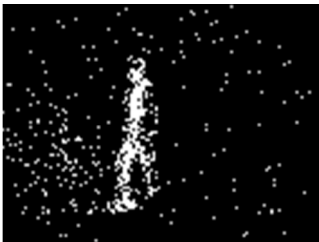


<i>Seuil T</i>	<i>Image originale</i>	<i>Image de détection S.F (FGMM - UM)</i>
<i>0.1</i>		
<i>0.4</i>		
<i>0.7</i>		
<i>1.0</i>		

Tableau .III.5 : Influence du seuil T sur les résultats de détection pour une scène simple.



- Pour une scène complexe :


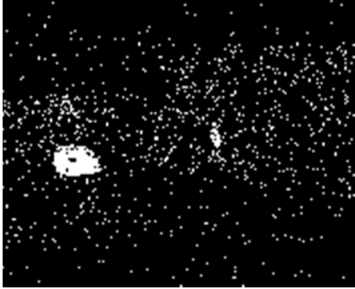

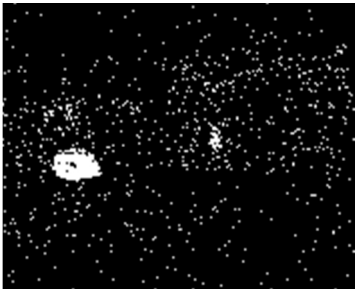




Seuil $T$	Image originale	Image de détection S.F (FGMM - UM)
0.1		
0.4		
0.7		
1.0		

Tableau .III.6 : Influence du seuil T sur les résultats de détection pour une scène complexe.

**Discussion :**





Le seuil T permet uniquement de choisir le nombre de Gaussiennes B appartenant au fond. On constate que plus le seuil T augmente (c'est-à-dire le nombre de Gaussiennes appartenant au fond augmente), la détection de mouvement s'atténue car, le nombre de Gaussiennes appartenant à l'avant plan diminue, et ce jusqu'à arrêt de détection.

**Remarque :**

Concernant les paramètres optimums utilisés pour les deux séquences d'images, le nombre de Gaussiennes K et le seuil T sont fixés à 3 et à 0.1 respectivement. Le paramètre Alpha est fixé à 0.05 pour la première séquence (la séquence simple), et fixé à 0.24 pour la deuxième séquence (la séquence complexe).

**III.3.5. Estimation du fond statique :**

Pour estimer le fond statique, nous utiliserons une séquence complexe d'un trafic routier.

<i>Trame</i>	<i>Image originale</i>	<i>Fond estimé</i>
<i>1</i>		
<i>8</i>		



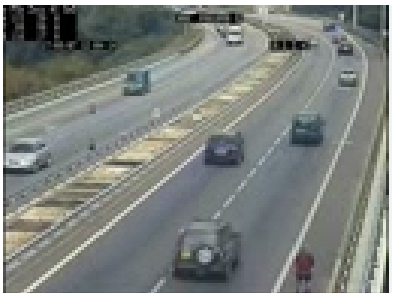



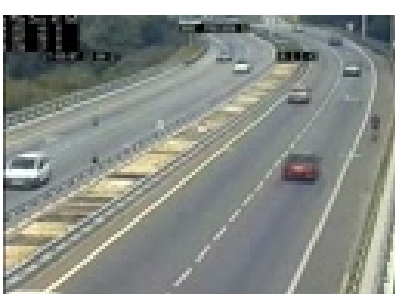

46		
93		
144		
402		

Tableau .III.7 : Estimation du fond statique à partir d'une séquence complexe.

**Discussion :**

Les paramètres utilisés pour cette séquence d'images sont fixés comme suit :

$\alpha = 0.05$ .

$\rho = 0.05$ .

$T = 0.3$ .

On constate que le fond commence à se construire à partir de la 46<sup>ième</sup> trame. La construction est peut être lente mais, elle est due à la faible valeur du paramètre « Alpha » (c'est-à-dire que dans la scène, le fond varie lentement).

Si on observe la 144<sup>ième</sup> trame, on remarque que le fond est presque totalement construit. On peut conclure alors que le fond est estimé à partir des 144 trames de la séquence d'images.

**III.4. Conclusion :**

Mis à part les bruits d'acquisitions auxquels nous pouvons remédier grâce à une opération de filtrage morphologique, les deux méthodes de détection utilisées ont donné des résultats satisfaisants. Il est clair que la méthode améliorée FGMM-UM est bien meilleure en terme de taux d'erreurs de fausses détection.

Il faut noter aussi que le choix des paramètres (K, ALPHA, RHO et seuil T) est crucial pour l'efficacité de la méthode utilisée, et ce sur la qualité des résultats de détection.

## *Conclusion et perspectives*

---

Le but de notre étude est de présenter une méthode de détection multimodale qui permet de détecter différents types de mouvements. Pour cela nous avons choisi deux méthodes basées sur les GMMs à savoir la GMM-standard et Fuzzy-GMM.

Nous avons trouvé de résultats de détection très satisfaisants, avec des taux d'erreurs estimés à 6.14% et à 5.32% en utilisant respectivement, la méthode de détection standard et améliorée. Il est clair que la méthode améliorée FGMM-UM est bien meilleure par rapport à la méthode standard, et ce en terme de fausses détections ou de précision.

D'autres améliorations existent certainement, ce qui permettrait d'optimiser la détection soit en réduisant les erreurs dues aux fausses détections, ou en améliorant le temps d'exécution, ou encore en réduisant le coût en mémoire, ...

Comme perspectives, nous proposerons deux solutions qui peuvent d'après nous contribuer à de meilleurs résultats.

Premièrement :

Quand la scène est connue au préalable, nous pouvons prédire la valeur de  $K$ . Mais si aucune donnée n'est présente, il sera préférable de calculer  $K$  automatiquement en utilisant l'algorithme K-means. [37] De plus, comme nous l'avons déjà indiqué précédemment, pour chaque pixel, le nombre de Gaussiennes  $K$  utilisé est expérimentalement fixé entre 3 et 5, ce qui ne mènerait sans doute pas à une détection optimale, car la scène peut varier en terme de modalité spatiale et temporelle. De ce fait, il serait plus efficace de choisir le nombre  $K$  instantanément selon la multi-modalité de la scène.

Deuxièmement :

Quand plusieurs mouvements à la fois lents et rapides se présentent dans la scène, prendre le paramètre ALPHA fixe pourrait engendrer de mauvais résultats de détection. Par exemple, si dans une scène, deux mouvements se présentent, le premier étant lent (un piéton)

## ***Conclusion générale***

---

et le second rapide (une voiture), si ALPHA est faible (ou large) alors nous obtiendrons contrairement à l'objet rapide (lent respectivement) une bonne détection concernant l'objet lent (rapide respectivement). De ce fait, il serait utile de choisir un ALPHA variable spatialement et temporellement, c'est-à-dire variable en fonction des mouvements présents dans la scène, et ce tout au long de la séquence. De notre point de vu, le choix de ALPHA se fera en estimant le champ de vitesse (flot optique) de tous les objets mobiles ce qui permettrait de diviser la scène en groupes (fonction de la vitesse des objets mobiles).

Enfin, nous espérons que ce modeste travail contribuera si peu soit-il à la réalisation des systèmes de télésurveillance et aidera les étudiants à mieux comprendre ce vaste domaine.

# Liste des références

---

[1] : B. LUVISON, Détection non supervisée d'évènements rares dans un flot vidéo : Application à la surveillance d'espaces publics, Thèse de doctorat, UNIVERSITÉ BLAISEPASCAL- CLERMONTII, 13 décembre 2010.

[2] : L. NIZAR & L.M. SEDDIK, Détection d'un Mouvement dans une Séquence Vidéo par Filtrage Morphologiques, Mémoire d'ingénieur en électronique ENP, 2009.

[4] : C. Fernandez – Maloigne & P. Carré, Quaternions et algèbres géométriques de nouveaux outils pour les images numériques couleur, 13 décembre 2007.

[5] : A. MOKHTARI, Diagnostic des systèmes hybrides : Développement d'une méthode associant la détection par classification et la simulation dynamique, thèse de doctorat, 23 octobre 2007.

[6] : C. Truche, Caractérisation et quantification des minéraux argileux dans sols expansifs par spectroscopie infrarouge aux échelles du laboratoire et du terrain, thèse de doctorat, Université de Toulouse, 13 décembre 2010.

[7] : Web, <http://show.docjava.com/book/cgij/code/data/lectures/cr311/pca.ppt>.

[8] : N. VERBEKE & N. VINCENT, Détection de Mouvements Cohérents dans une Séquence Vidéo, Université René Descartes, 25 Mai 2007.

[9] : N. VERBEKE, Suivi d'objets en mouvement dans une séquence vidéo, Université PARIS DESCARTES, décembre 2007.

[10] : M.J. RICHEFEU, Détection et analyse du mouvement sur système de vision à base de rétine numérique, thèse de doctorat, UNIVERSITE PARIS 6, 14 décembre 2006.

[11] : M.J. RICHEFEU & M.A. MANZANERA, Détection de mouvement par capteur intelligent, Ecole Nationale Supérieure de Techniques Avancées.

[12] : Y.S. KORI & L. HOCINI, Détection de mouvement dans une séquence vidéo par une méthode multimodale, Mémoire de fin d'études, ESI ex : INI, 2009.

[13] : Web, [http://www.ensta-paristech.fr/~manzane/Download/Tutorials/ETASM06\\_chap2.pdf](http://www.ensta-paristech.fr/~manzane/Download/Tutorials/ETASM06_chap2.pdf).

[14] : A. BAUCHER, Vision par ordinateur, IFI.

[15] : A. BASARAB , Estimation du mouvement dans des séquences d'images échographiques : application à l'élastographie de la thyroïde, thèse doctorale, 2008.

[16] : C. LEIGNEL & J.M. JOLION, Etude bibliographique suivi de personnes et trajectoire,

[17] : D. FILLIAT, Estimation du mouvement, Master Systèmes et Applications Répartis.

[18] : J. LORENCEAU, Mécanismes de la perception visuelle du mouvement.

[19] : S. BAZEILLE, Traitement d'images et vision par ordinateur.

[20] : R. HORAUD & O. MONGA, Vision par ordinateur : Outils fondamentaux, Deuxième édition, Editions Hermès.

[21] : Web, <http://www.semageek.com/la-detection-de-collision-pour-protger-l-homme-des-robots-dangereux/>.

[22] : Les technologies de l'imagerie au service de l'analyse du mouvement en pédagogie du piano, École d'ingénierie et de technologie de l'information, Université d'Ottawa.

[23] : C. STAUFFER & W.E.L. GRIMSON. Learning patterns of activity using real-time tracking, Artificial Intelligence Laboratory Massachusetts Institute of Technology Cambridge.

[24] : L. CARMINATI & J. BENOIS-PINEAU, Détection des zones de mouvement et régularisation : Application à la vidéo surveillance.

[25] : T. BOUWMANS, F. EL BAF & B. VACHON, Background Modeling using Mixture of Gaussians for Foreground Detection - A Survey, Laboratoire MIA, Université de La Rochelle.

[26] : J.G. VILA, Foreground Segmentation and Tracking based on Foreground and Background Modeling Techniques, Master Thesis, 2009.

[27] : M. PINGAULT, Estimation fréquentielle et temporelle du mouvement en transparence additive dans les séquences d'images, Thèse de doctorat, Université JOSEPH FOURIER.

[28] : Web, <http://www.cedar.buffalo.edu/~srihari/CSE574/Chap9/Ch9.2-MixturesofGaussians.pdf>.

[29] : Dar-Shyang LEE, Effective Gaussian Mixture Learning for Video Background Subtraction.

[30] : T. BOUWMANS & F. EL BAF, Modeling of Dynamic Backgrounds by Type-2 Fuzzy Gaussian Mixture Models, Paper.

[31] : Z. ZIVKOVIC, Improved Adaptive Gaussian Mixture Model for Background Subtraction, In Proc, ICPR, 2004.

[32] : S.Y. ELHABIAN, K.M. EL-SAYED & S.H. AHMED, Moving Object Detection in Spatial Domain using Background Removal Techniques, Recent Patents on Computer Science, 2008, Vol. 1, No. 1 33.

[33] : C.R. WREN, A. AZARBAYEJANI, T. DARRELL & A.P. PENTLAND, Pfinder : Real-Time Tracking of the Human Body, IEEE transactions on pattern analysis and machine intelligence, VOL. 19, NO. 7, JULY 1997.

[34] : Initiation au traitement d'images avec MATLAB, E.N.S.B.A.N.A. 2ème année, 2002/2003

[35] : N. VANDENBROUCKE, Traitement d'images sous MatLab, Ecole d'Ingénieurs de Pas-de-Calais, année 2009/2010.

[36] : J. Briot, Introduction à la gestion des images : Tour d'horizon des possibilités offertes par MATLAB, 2007.

[37] : G. H. BALL & D. J. HALL, ISODATA, A novel Method Of Data analysis And Pattern Classification, Stanford Research Institute, Menlo Park - California, April 1965.