



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



N° Série :.....

Mémoire de fin d'étude

Présenté pour l'obtention du diplôme de Master

Filière : Télécommunications

Spécialité : Systèmes des Télécommunications

THEME

Détection et poursuite d'un objet coloré sur une séquence vidéo

Réalisé par : TAGHRAOUT Hamza

Encadreur : Mme. FRIHA Souad

Devant Le Jury :

Président:

Dr. HASSAD Mourad

Examineur:

Dr. KHALFAOUI Mehjou

Promotion 2021/2022

بِسْمِ
الرَّحْمَنِ
الرَّحِيمِ

Remerciements

الحمد لله أولاً و الحمد لله آخراً و الحمد لله دائماً و أبداً

*Je voudrais exprimer ma gratitude à mon encadreur Mme
FRIHA Souad qui m'a guidé à travers ce travail,*

*Je tiens également à exprimer mes sincères remerciements
aux membres du jury,*

*Je remercie profondément tous les enseignants qui ont
contribué à la réalisation de ce modeste travail, ma famille en
particulier mon père, mon frère et mes proches Bouchra et
Abdenour, collègues et amis,*

*Je voudrais enfin remercier tous ceux qui ont participé de
loin ou de près avec une idée ou une suggestion, ou avec un
moment de soutien moral.*

*Enfin et surtout, je veux me remercier d'avoir cru en moi, je
veux me remercier pour tout ce travail acharné, je veux me
remercier de ne jamais avoir abandonné...*

Merci.

DÉDICACE

Je dédie ce modeste travail

À mes très chers parents

À mon frère Tarek et mes petites sœurs

À mes cousins Bouchra et Abdenour

À toute ma famille et mes amis

*À mes collègues, spécialement Rahila, Nabil , Marwa , Farah
Hadil et Rihana ...*

À Salsabil pour avoir coopéré avec moi

À tous ceux qui m'ont encouragé et soutenu...

Hamza

Résumé :

Un objet a de nombreuses caractéristiques générales qui le décrivent dans son ensemble telles que la couleur et la forme. Ces discriminateurs peuvent être utilisés pour détecter et suivre un objet dans une série d'images.

Ce mémoire présente une technique de détection et de suivi d'objets en fonction respectivement de sa couleur et de son mouvement. Cette méthode est utile lorsque l'objet à détecter est d'une couleur spécifique et que cette couleur est différente de la couleur de fond.

Le système interactif de programmation scientifique MATLAB est utilisé pour programmer l'algorithme. La détection et le suivi des objets en mouvement sont exécutés en temps réel dans différentes images de la vidéo en fonction de leurs couleurs et leurs mouvements. La méthode a prouvé son efficacité dans la plupart des cas.

Le développement de solutions fiables de traitement de séquences vidéo ouvre de nouveaux horizons pour de nombreuses applications telles que les systèmes de transport intelligents, la vidéosurveillance et la robotique.

Mots clé : Détection, suivi, objet coloré, seuillage, image RGB.

Abstract :

An object has many general characteristics such as color and shape, which describe the object as a whole. These discriminators can be used to detect and track an object in a series of images.

This thesis introduces a technique for detecting and tracking objects using the color and the motion features. This method is useful when the object to be detected is of a specific color and this color is different from the background color.

We worked on an algorithm developed on the interactive scientific programming system MATLAB to track moving objects in real time in different frames of the video using color and motion function, and it has proven to be effective in most cases.

The development of reliable solutions for processing video sequences opens new horizons for many applications such as intelligent transport systems, video surveillance and robotics.

Key words : Detection, tracking, colored object, thresholding, RGB.image

ملخص:

للأجسام العديد من الخصائص العامة مثل اللون والشكل والتي تصف الجسم ككل.

يمكن استخدام هذه المميزات لاكتشاف وتتبع جسم ما في سلسلة من الصور. يقدم هذا البحث تقنية للكشف عن الاجسام وتعقبها باستخدام ميزتي اللون والحركة . هذه الطريقة مفيدة عندما يكون الجسم المراد اكتشافه بلون معين ويختلف لونه هذا عن لون الخلفية.

قمنا بالعمل على خوارزمية تم تطويرها على برنامج ماتلاب لتتبع الأجسام المتحركة في الوقت الحقيقي في صور مختلفة من الفيديو باستخدام ميزة الألوان والحركة ، وقد أثبتت فعاليتها في معظم الحالات.

ان تطوير حلول موثوقة لمعالجة الفيديوهات يفتح آفاقًا جديدة للعديد من التطبيقات مثل أنظمة النقل الذكية والمراقبة بالفيديو والروبوتات .

الكلمات المفتاحية: الكشف، التتبع، كائن ملون، عتبة، صور RGB.

SOMMAIRE

Remercîment	
Dédicace	
Résumé	
Liste des figures	
Liste des abréviations	
Introduction	1

CHAPITRE I : GÉNÉRALITÉS SUR LE TRAITEMENT D'IMAGE

I.1 Introduction	4
I.2 Définition d'une image	4
I.3 L'image numérique	4
I.4 Caractéristiques d'une image numérique	5
I.4.1 Pixel	5
I.4.2 La résolution	6
I.4.3 Dimension	7
I.4.4 La texture	7
I.4.5 Bruit	7
I.4.6 La luminance	8
I.4.7 Contours et textures	8
I.4.8 Histogramme	8
I.4.9 Le contraste	9
I.5 Types d'images	10
I.5.1 Images binaires (en noir et blanc)	10
I.5.2 Images à niveaux de gris (Monochromes)	10
I.5.3 Images couleurs (polychrome)	11
I.6 Qualité de l'image numérique	12
I.7 Images bitmap et images vectorielles	12

I.8 Système de traitement d'image	12
I.9 traitement numérique des images	13
I.9.1 Filtrage Numérique	13
I.9.1.1 Filtres linéaires	13
I.9.1.2 Filtres non linéaire	15
I.9.2 Avantages du filtrage	17
I.9.3 Segmentation d'image	17
I.9.4 Détection de contour	17
I.9.5 Domaines d'application	18
I.10 Seuillage d'une Image	18
I.11 La vidéo	19
I.12 Définition	19
I.13 Résolution temporelle d'une vidéo	19
I.14 Séquence d'image	20
I.15 Types de vidéo	20
I.15.1 La vidéo analogique	20
I.15.2 La vidéo numérique	20
I.16 Les fondamentaux physiques et techniques	21
I.16.1 La vision	21
I.16.2 La couleur	21
I.16.3 Le signal	22
I.17 Les caractéristiques d'un signal vidéo	23
I.18 Objet vidéo	23
I.18.1 Définition	23
I.18.2 Suivi d'objets dans la vidéo	24

I.19 Conclusion	25
-----------------------	----

CHAPITRE II : DÉTECTION ET SUIVI D'OBJETS

II.1 Introduction	27
II.2 Objet	27
II.3 Modélisation d'un objet	27
II.3.1 Caractéristiques d'un objet	27
II.3.2 Représentation d'un objet	28
II.3.2.1 Modélisation par points	28
II.3.2.2 Modélisation par formes géométriques	29
II.3.2.3 Modélisation par silhouette et contour	29
II.3.2.4 Modélisation par des modèles de formes articulées	30
II.3.2.5 Modèles squelettiques	30
II.4 La détection de l'objet	31
II.4.1 Méthode de Détection d'objets en mouvement	31
II.4.1.1 La différence temporelle des images (méthode de mouvement)	31
II.4.1.2 Double de différence temporelle et caractère de contour	32
II.4.1.3 La soustraction de l'image de fond (méthode de différence)	34
II.4.1.4 Méthode de détection des objets selon la Couleur	35
II.4.1.5 Les détections utilisant la notion de cohérence	37
II.5 Le suivi des objets	37
II.5.1 But du suivi.....	37
II.5.2 Méthodes de suivi	37
II.5.2.1 Suivi par appariement de détections.....	38
II.5.2.2 Le filtre de Kalman	39
II.5.2.3 Segmentation dynamique	39

II.5.2.4 Suivi déterministe de fenêtre englobante par détection séquentielle.....	40
II.6 Caractéristiques d'un bon algorithme de suivi	41
II.7 Conclusion	42

CHAPITRE III: Résultats et commentaires

III.1 Introduction	44
III.2 Environnement du travail	44
III.2.1 Environnement matériel ou le hardware	44
III.2.2 Environnement immatériel ou le software	44
III.3 MATLAB	45
III.4 L'interface graphique (GUI)	47
III.5 Schéma résumant les étapes des algorithmes appliqués.....	49
III.6 Principes de base de l'utilisation d'une caméra à l'aide de MATLAB.....	50
III.7 Implémentation de l'algorithme de détection et de suivi des couleurs RVB-B.....	52
III.8 Première partie	52
III.8.1 Les seuils de couleurs.....	52
III.9 Le cycle de traitement	53
III.10 Résultats pour la détection des couleurs	57
III.10.1 Détection d'objets rouges.....	57
III.10.2 Détection d'objets verts.....	58
III.10.3 Détection d'objets bleus	59
III.10.4 Détection d'objets blancs	60
III.11 Résultats de suivi.....	61
III.11.1 Suivi d'un objet rouge	61
III.11.2 Suivi d'un objet vert.....	62
III.11.3 Suivi d'un objet bleu	63
III.11.4 Suivi d'un objet blanc	63
III.11.5 Suivi multi objet coloré	65

III.12 Méthode utilisée pour le suivi	66
III.13 Les paramètres que nous avons utilisés	66
III.14 Discuter et interpréter les résultats	66
III.15 Conclusion.....	67
Conclusion générale	69
Bibliographie.....	

LISTES DES FIGURES

CHAPITRE I

Figure I.1: Représentation d'image numérique.....	5
Figure I.2 : groupe de pixel formant la lettre « a »	5
Figure I.3 : Une image en niveaux de gris avec Sous image de taille 5×5 représenter l'intensité lumineuse par chaque pixel	6
Figure I.4 : la relation entre la densité de pixels et sa résolution	6
Figure I.5 : la taille de l'image.	7
Figure I.6 : la définition de l'image	7
Figure I.7 : Image sans bruit.....	7
Figure I.8 : Image avec bruit.	7
Figure I.9.: Contour d'une image	8
Figure I.10 : Image avec son histogramme.	9
Figure I.11 : Images avec contraste différent	9
Figure I.12 : Image binaire.	10
Figure I.13 : Image Monochrome	11
Figure I.14 : Image polychrome (65536 couleurs).	11
Figure I.15 : Composition d'un système de traitement numérique	12
Figure I.16 : Principe du filtre médian.	15
Figure I.17 : Principe du filtre maximum.	16
Figure I.18 : Principe du filtre minimum.	16
Figure I.19 : Exemple d'une segmentation d'image, (a) : image d'origine, (b) : image segmentée.....	17
Figure I.20 : Une image aux niveaux de gris avant et après le seuillage	21
Figure I.21 : Exemple d'une séquence d'images	20
Figure I.22 : codage de couleur en RVB.....	21
Figure I.23 : Schéma de principe d'un Capture Tri-CCD (charged coupled device)	22

CHAPITRE II

Figure II.1 : Points de contrôle.....	28
Figure II.2 : Boîtes englobantes	29
Figure II.3 : Modèle silhouette.....	29
Figure II.4 : Modèle d'apparence articulé.....	30
Figure II.5 : Modèle squelette	30
Figure II.6 : Un exemple de la différence temporelle [6]. (a) une scène simple avec deux objets, (b) les régions rouges sont la différence entre deux images consécutives	31
Figure II.7 : Détection de mouvement par différence temporel.....	34
Figure II.8 : Détections basées sur la modélisation du fond	34
Figure II.9 : Exemple1, pixels de couleur rouge reconnus dans les particules et reconnaissance de l'objet	36
Figure II.10 : Exemple 2, reconnaissance d'un objet à l'aide de la couleur après modification d'un paramètre.....	36
Figure II.11 : Contour actif	40
Figure II.12 : Suivi de distribution	41

CHAPITRE III

Figure III.1 : le matériel utilisé dans ce travail	45
Figure III.2 : Comment entrer dans l'environnement de développement 'guide'	46
Figure III.3 : l'interface graphique de guide	47
Figure III.4 : L'interface graphique de détection et suivi d'objet colorés	48
Figure III.5 : Les étapes de la détection.	49
Figure III.6 : L'aperçu de la caméra	51
Figure III.7 : la photo capturée (snapshot)	52
Figure III.8 : image originale RGB	54
Figure III.9 : niveau de gris de l'image originale	54
Figure III.10 : Extraction de la composante rouge	55
Figure III.11 : Extraction de la composante verte	55

Figure III.12 : Extraction de la composante bleue	55
Figure III.13 : image (a) avant le filtrage médian et l'image (b) après le filtrage.....	56
Figure III.14 : Seuillage pour la couleur rouge	56
Figure III.15 : Détection d'un seul objet rouge	57
Figure III.16 : Détection de multi objets rouges	57
Figure III.17 : Détection d'un seul objet vert	58
Figure III.18 : Détection de multi objets verts	58
Figure III.19 : Détection d'un seul objet bleu.....	59
Figure III.20 : Détection de multi objets bleus.....	59
Figure III.21 : Détection d'un seul objet blanc	60
Figure III.22 : Détection de multi objets blancs.....	60
Figure III.23 : Suivi d'un Poivron rouge dans une séquence vidéo	61
Figure III.24 : Suivi d'un ciseau vert dans une séquence vidéo.....	62
Figure III.25 : Suivi une carte bleue dans une séquence vidéo	63
Figure III.26: Suivi une boîte de chargeur blanche dans une séquence vidéo	64
Figure III.27 : Suivi multi objets (verts) dans une séquence vidéo.....	65

Liste des abréviations

Pixel	<u>P</u>i ctu<u>r</u>e <u>E</u>L ement
PPI	P ixel P er I nch
RGB	<u>R</u>eed <u>G</u>reen <u>B</u>lue
RVB	<u>R</u>ouge <u>V</u>ert <u>B</u>leu
YCbCr	Luminance (Y) , Chrominance (Red-Yellow), Chrominance (Blue-Yellow)
3D	Images A <u>T</u>rois <u>D</u>imensions.
CCD	<u>C</u>harged <u>C</u>oupled <u>D</u>evice
SSD	<u>S</u>um of <u>S</u>quare <u>D</u>ifferences
MATLAB	<u>M</u>atrix-<u>L</u>aboratory

Introduction Générale

Introduction générale

Une image vaut mille mots, en raison de la grande quantité d'informations qu'elle peut contenir, Actuellement les images numériques touchent de nombreux domaines de notre vie, tels que la médecine, la météo, les communications, la cartographie, la géologie...etc., et avec le développement des ordinateurs et la généralisation des images numériques, de nombreuses techniques de traitement et d'analyse ont été proposées, consistant en des transformations mathématiques sur les images afin d'en améliorer la qualité ou d'en extraire des informations, à l'instar de la découverte d'objets en mouvement et de leur suivi dans une séquence d'images ou de vidéos.

La détection et le suivi d'objet dans les images et les vidéos est indispensable pour des applications nombreuses et variées, telles que la vidéosurveillance, la compression vidéo, l'imagerie médicale, la robotique, l'interaction homme-machine, l'analyse de séquences mathématiques...etc. En effet, souvent les zones de mouvement correspondent aux zones sur lesquelles se concentre un système de vision.

La détection et le suivi d'objets sont des technologies de vision par ordinateur dont les algorithmes reposent généralement en partie sur l'apprentissage automatique ou l'apprentissage détaillé pour produire des résultats utiles, et l'analyse de mouvement est un vaste sujet qui comprend un certain nombre de problèmes. On peut citer notamment :

- La détection de mouvement : qui consiste à nommer chaque pixel de l'image selon qu'il correspond ou non à une zone en mouvement de la scène.
- Détection d'objets en mouvement : c'est-à-dire la détection d'un groupe de zones d'intérêt en mouvement dans la scène tridimensionnelle observée.
- Segmentation de scène basée sur le mouvement : chaque zone de l'image est détectée et segmentée avec un mouvement distinct des autres zones.

Introduction générale

- Estimation de mouvement : qui consiste à estimer, à partir d'une série d'images, le mouvement apparent d'objets composant une scène en trois dimensions.
- Suivi des primitives ou régions : son but est de déterminer la position de chaque primitive ou région dans l'image à chaque instant.
- Reconnaître et modéliser des activités ou des gestes.

Les trois premiers problèmes (détection de mouvement, détection d'objets en mouvement et segmentation basée sur le mouvement) sont, en général, la première étape pour les outils de vision artificielle par ordinateur. Le but de ces outils peut être de détecter, détecter et identifier des objets, ou détecter et suivre des objets, par exemple pour analyser le comportement ou la trajectoire de tels objets.

D'autres enjeux ne sont pas moins importants et nécessitent la mise en place de méthodes simples et efficaces. Tous ces sujets font l'objet d'un grand nombre de travaux, mais il n'existe actuellement aucun algorithme performant adapté à toute situation.

L'objectif du travail présenté dans ce mémoire est de concevoir un code sous **MATLAB** pour détecter et suivre des objets colorés dans l'image. C'est pourquoi nous avons construit notre mémoire autour de trois chapitres :

Chapitre Un : Dans ce chapitre nous présentons des généralités sur le traitement d'images et une étude générale des caractéristiques de base des vidéos.

Chapitre Deux : Nous discutons des méthodes de détection d'objets, de leurs avantages et inconvénients, ainsi que des différentes techniques de suivi.

Chapitre Trois : Dans le dernier chapitre, nous présentons tous les résultats expérimentaux qui ont été obtenus pour la détection et le suivi d'objets colorés via la méthode choisie.

CHAPITRE I

GÉNÉRALITÉS SUR LE TRAITEMENT D'IMAGE

I.1 Introduction :

les images sont l'un des moyens les plus importants que les gens utilisent pour percevoir ce qui les entoure et communiquer entre eux, comme le dit le romancier russe **Ivan Tourgueniev** ; « une seule image peut montrer ce qu'un écrivain ne pourrait pas dire en 100 pages ».

Le traitement d'images est l'une des branches de l'informatique et est représenté par un ensemble d'algorithmes (méthodes et techniques mathématiques) qui améliorent la qualité des images et les rendent plus pures afin d'en extraire diverses informations, découvrir les objets qu'elles contiennent.. . etc.

Dans ce chapitre, nous discutons d'un ensemble de concepts de base et d'informations générales concernant le domaine du traitement d'images numériques, telles que la définition de l'image, les types d'images, leurs propriétés,...

I.2 Définition d'une image:

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

Elle peut être décrite sous la forme d'une fonction $I(x, y)$ analogique continue, définie dans un domaine borné, tel que x et y sont les coordonnées spatiales d'un point de l'image et I est une fonction d'intensité lumineuse et de couleur. Sous cet aspect, l'image est inexploitable par la machine, ce qui nécessite sa numérisation. [1]

I.3 L'image numérique :

Contrairement aux images obtenues à l'aide d'un appareil photo, ou dessinées sur du papier, les images manipulées par un ordinateur sont numériques (représentées par une série de bits).

L'image numérique est l'image dont la surface est divisée en éléments de tailles fixes appelés cellules ou pixels, ayant chacun comme caractéristique un niveau de gris ou de couleurs prélevé à l'emplacement correspondant dans l'image réelle, ou calculé à partir d'une description interne de la scène à représenter [2].

La numérisation d'une image est la conversion de celle-ci de son état analogique en une image numérique représentée par une matrice bidimensionnelle de valeurs numériques $f(x,y)$ où :

x, y : coordonnées cartésiennes d'un point de l'image,

$f(x, y)$: niveau de gris en ce point.

Pour des raisons de commodité de représentation pour l'affichage et l'adressage, les données images sont généralement rangées sous formes de tableau I de n lignes et p colonnes. Chaque élément I(x, y) représente un pixel de l'image et à sa valeur est associé un niveau de gris codé sur m bits (2^m niveaux de gris ; 0 = noir ; 2^m-1 = blanc). La valeur en chaque point exprime la mesure d'intensité lumineuse perçue par le capteur.

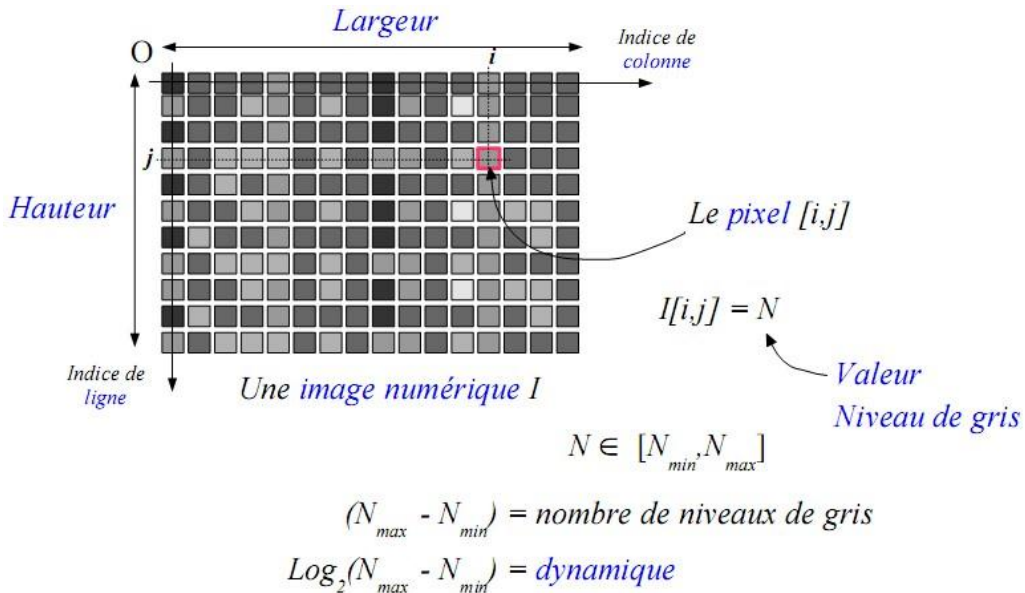


Figure I.1: Représentation d'image numérique

I.4 Caractéristiques d'une image numérique:

Comme nous l'avons vu, l'image est un ensemble structuré d'informations parmi ses caractéristiques nous pouvons citer les paramètres suivants:

I.4.1 Pixel :

Contraction de l'expression anglaise " Picture Elements ": éléments d'image, le pixel est le plus petit point de l'image, c'est une valeur numérique représentative des intensités lumineuses.

Si le bit est la plus petite unité d'information que peut traiter un ordinateur, le pixel est le plus petit élément que peuvent manipuler les matériels et logiciels sur l'image (cf. Figure I.2 et Figure I.3) [3].

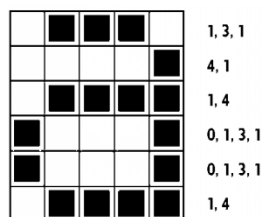


Figure I.2 : groupe de pixel formant la lettre « a »



Figure I.3 : Une image en niveaux de gris avec Sous image de taille 5×5 représentant l'intensité lumineuse par chaque pixel

I.4.2 La résolution :

La résolution d'une image correspond au niveau de détail qui va être représenté sur cette image. C'est le nombre de pixels par unité de longueur dans l'image à numériser. Elle est en dpi (dots per inch) ou en ppp (points par pouce) (cf. Figure 1.4).

Plus le nombre de pixels est élevé par unité de longueur de l'image à numériser, plus la quantité d'information qui décrit l'image est importante et plus la résolution est élevée. [4]

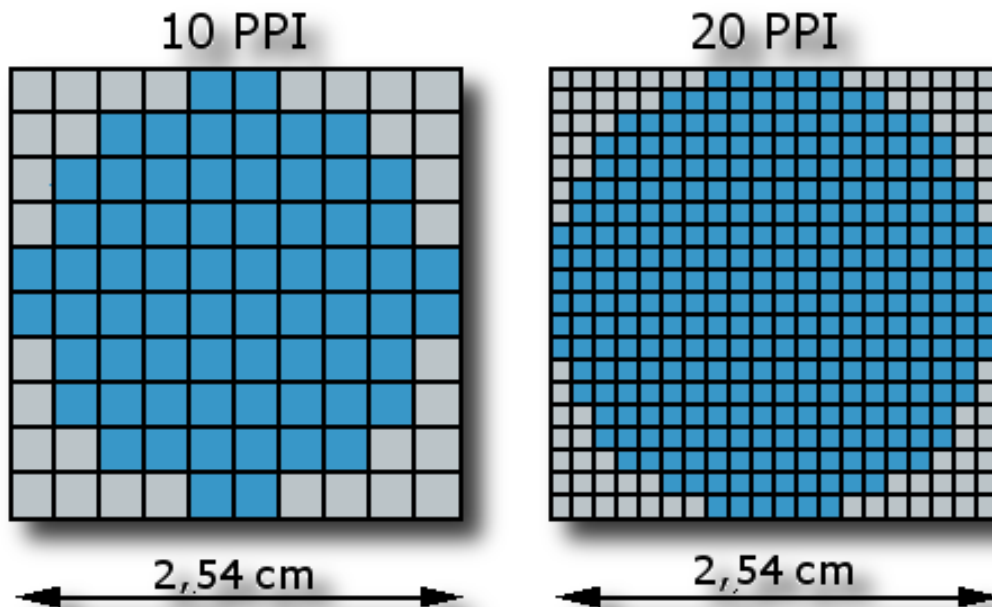


Figure I.4 la relation entre la densité de pixels et sa résolution.

I.4.3 Dimension :

C'est la taille de l'image. Cette dernière se présente sous forme de matrice dont les éléments sont des valeurs numériques représentatives des intensités lumineuses (pixels). Le nombre de lignes de cette matrice multiplié par le nombre de colonnes nous donne le nombre total de pixels dans une image (cf. Figure 1.5). [5]

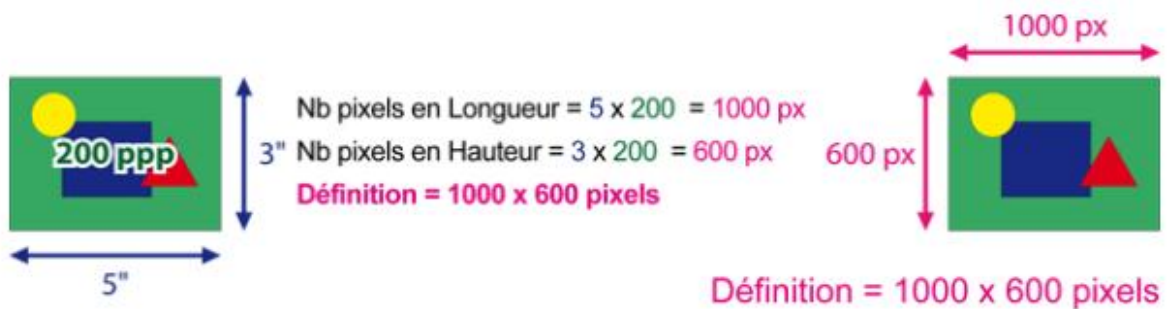


Figure I.5 : La taille de l'image.

Figure I.6 : La définition de l'image

I.4.4 La texture :

Une texture est une région dans une image numérique qui a des caractéristiques homogènes. Ces caractéristiques sont par exemple un motif basique qui se répète. La texture est composée de Texel.

Un Texel est un pixel unique dans une carte de texture, qui est une image qui est utilisée (en tout ou en partie) comme image présentée sur la surface d'un polygone dans une image rendue en 3D. Il ne faut pas le confondre avec le pixel qui est l'unité d'espace de l'écran.

I.4.5 Bruit :

Un bruit (parasite) dans une image est considéré comme un phénomène de brusque variation de l'intensité d'un pixel par rapport à ses voisins, il provient de l'éclairage des dispositifs optiques et électroniques du capteur.[2]



Figure I.7 : Image sans bruit .

Figure I.8 : Image avec bruit.

I.4.6 La luminance :

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

Une bonne luminance se caractérise par :

- Des images lumineuses (brillantes);
- Un bon contraste : il faut éviter les images où la gamme de contraste tend vers le blanc ou le noir; ces images entraînent des pertes de détails dans les zones sombres ou lumineuses.
- L'absence de parasites. [6]

I.4.7 Contours et textures :

Les contours représentent la frontière entre les objets de l'image, ou la limite entre deux pixels dont les niveaux de gris représentent une différence significative. Les textures décrivent la structure de ceux-ci. L'extraction de contour consiste à identifier dans l'image les points qui séparent deux textures différentes.[6]



Figure I 9.: Contour d'une image.

I.4.8 Histogramme :

L'histogramme des niveaux de gris ou des couleurs d'une image est une fonction qui donne la fréquence d'apparition de chaque niveau de gris (couleur) dans l'image. Pour diminuer l'erreur de quantification, pour comparer deux images obtenues sous des éclairages différents, ou encore pour mesurer certaines propriétés sur une image. [6],[2] Il permet de donner un grand nombre d'information sur la distribution des niveaux de gris (couleur) et de voir entre quelles bornes est répartie la majorité des niveaux de gris (couleur) dans les cas d'une image trop claire ou d'une image trop foncée.

Il peut être utilisé pour améliorer la qualité d'une image (Rehaussement d'image) en introduisant quelques modifications, pour pouvoir extraire les informations utiles de celle-ci. La figure (1-8) montre une image avec son histogramme.

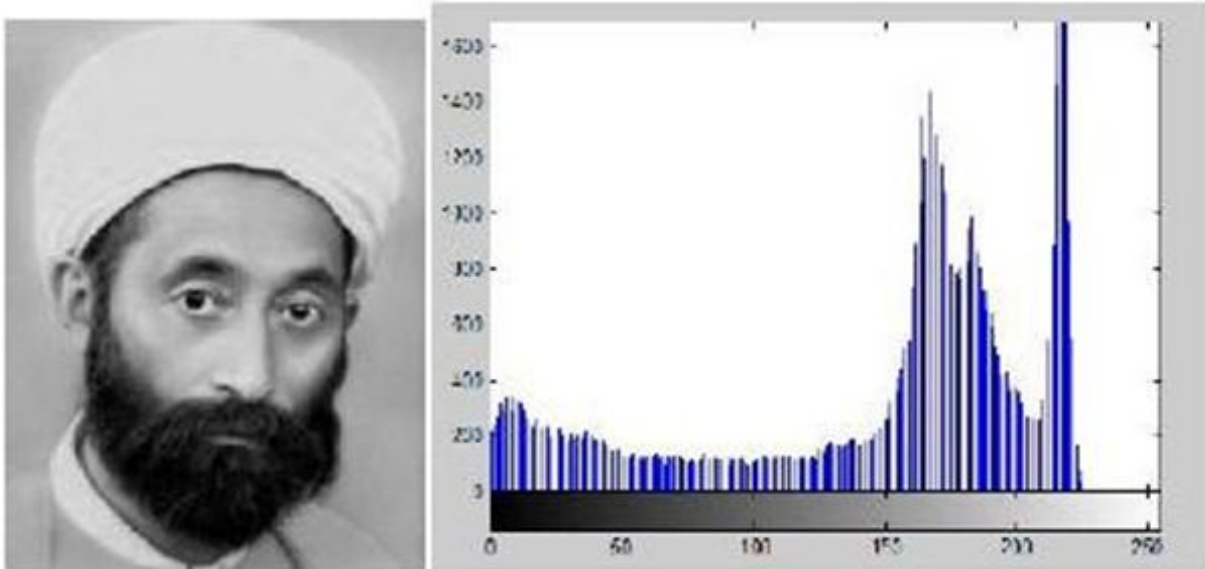


Figure I.10 : Image avec son histogramme.

I.4.9 Le contraste :

C'est l'opposition marquée entre deux régions d'une image, plus précisément entre les régions sombres et les régions claires de cette image. Le contraste est défini en fonction des luminances de deux zones d'images.

Si $L1$ et $L2$ sont les degrés de luminosité respectivement de deux zones voisines $A1$ et $A2$ d'une image, le contraste C est défini par le rapport : [7]

$$C = \frac{L1 - L2}{L1 + L2}$$



a) Image mal contrastée



b) Image bien contrastée

Figure I.11 : Images avec contraste différent

I.5 Types d'images :

On distingue trois types d'images :

- Binaire : 2 couleurs (arrière-plan et avant-plan).
- Monochrome : variations d'une même teinte.
- Polychrome : " vraies " couleurs. [9]

I.5.1 Images binaires (en noir et blanc) :

Les images binaires sont les plus simples. Bichromes (dont les pixels ne peuvent avoir que les valeurs 0 et 1) .Le 0 correspond à un pixel noir et 1 à un pixel blanc. Le niveau de gris est donc codé sur un seul bit.



Figure I.12 : Image binaire.

I.5.2 Images à niveaux de gris (Monochromes) :

Le niveau de gris est la valeur de l'intensité lumineuse en un point. La couleur du pixel peut prendre des valeurs allant du noir au blanc en passant par un nombre fini de niveaux intermédiaires. Donc pour représenter les images à niveaux de gris, on peut attribuer à chaque pixel de l'image une valeur correspondant à la quantité de lumière renvoyée. Cette valeur peut être comprise par exemple entre 0 et 255. Chaque pixel n'est donc plus représenté par 1 bit, mais par 1 octet. Pour cela, il faut que le matériel utilisé pour afficher l'image, soit capable de produire les différents niveaux de gris correspondant.

Le nombre de niveaux de gris dépend du nombre de bits utilisés pour décrire la " couleur " de chaque pixel de l'image. Plus ce nombre est important, plus les niveaux possibles sont nombreux.[9]



Figure I.13 : Image Monochrome

I.5.3 Images couleurs (polychrome) :

Elle est obtenue par la combinaison de trois couleurs dites primaires : rouge, vert et bleu (RVB). Chaque couleur est codée comme une image à niveaux de gris, avec des valeurs allant de 0 à 255. Pour $R=V=B=0$ nous aurons un noir pur, et pour $R=V=B=255$ nous aurons un blanc pur. La représentation des images couleurs se fait donc soit par une image dont la valeur du pixel est une combinaison linéaire des valeurs des trois composantes couleurs, soit par trois images distinctes représentant chacune une composante couleur, on distingue généralement deux types d'images: [10]

- Images 24 bits ;
- Images à palettes ;



Figure I.14 : Image polychrome (65536 couleurs).

I.6 Qualité de l'image numérique :

Elle dépend, d'une part, de la qualité des images d'origine et, d'autre part, des moyens mis en œuvre pour convertir un signal analogique en signal numérique. Elle dépend aussi de :

- La qualité des périphériques de numérisation de l'image, du nombre de niveaux de gris ou de couleurs enregistrées, etc.
- La qualité de l'affichage à l'écran : définition de l'écran, nombre de teintes disponibles. [11]

Les critères d'appréciation de la qualité d'une image, tels que cités succinctement ci-dessus, dépendent largement de la structure même de l'image réaliste ou conceptuelle et de son mode de représentation (bitmap ou vectorielle). [9]

I.7 Images bitmap et images vectorielles :

Les images appartiennent à deux grandes familles : bitmap (image-bit) et vectorielle. Alors qu'une image vectorielle est décrite à l'aide de courbes et d'équations mathématiques, une image bitmap est constituée de pixels et se réduit donc à une matrice de points. Si les images vectorielles peuvent être manipulées avec beaucoup de facilité, les modifications de taille, par exemple, apportées à une image bitmap ne sont pas sans incidence.

I.8 Système de traitement d'image :

Un système de traitement d'image est généralement composé des unités suivantes (cf. Figure 1.15):

- Un système d'acquisition et de numérisation qui permet d'effectuer l'échantillonnage et la quantification d'une image.
- Une mémoire de masse pour stocker les images numérisées.
- Un système de visualisation.
- Une unité centrale permettant d'effectuer les différentes opérations de traitement d'images. [12]

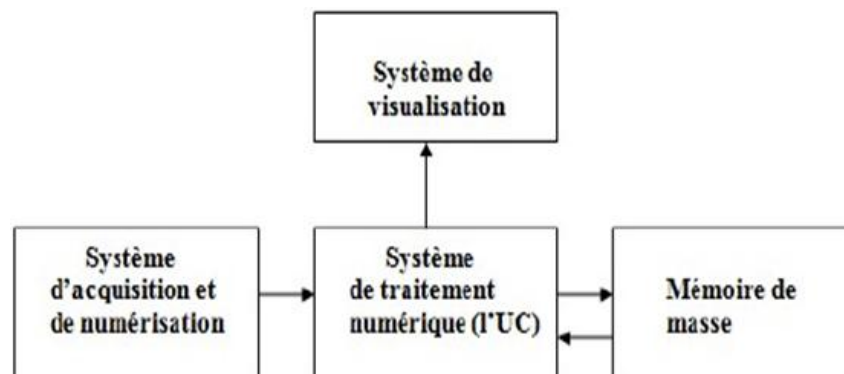


Figure I.15 : Composition d'un système de traitement numérique.

Acquisition et numérisation :

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

Visualisation :

Tout système de traitement d'image est doté d'un dispositif de visualisation qui permet l'affichage des images.

L'utilisation de différents types de reconstituteurs permet de transformer le signal numérique qu'est la matrice image en un signal analogique visible par l'œil de l'observateur. Pour cela, différents types de supports peuvent être employés : moniteur vidéo, clichés photographiques, impression sur papier. Dans tous les cas et pour chaque échantillon de l'image numérique, on recrée un nouvel élément d'image ou un nouveau pixel dont on choisit la forme de façon à reconstituer une image analogique qui soit la plus proche possible de l'image avant numérisation compte tenu des erreurs introduites lors de l'acquisition, de la numérisation et de la transmission .[11]

I.9 Traitement numérique des images :

Les techniques de traitement sont destinées à l'exploitation des informations contenues dans les images, ceci dans le but d'améliorer la qualité des images et de les rendre plus facilement interprétables, en d'autres termes elles permettent d'augmenter la qualité visuelle de l'image

I.9.1 Filtrage Numérique :

Les images numériques telles qu'elles sont acquises, sont très souvent inexploitable pour le traitement d'images. Elles contiennent des signaux bruités. Pour remédier à cela, différents prétraitements pour l'amélioration ou la correction sont effectués. [10]

On peut scinder les filtres en deux grandes catégories :

I.9.1.1 Filtres linéaires :

Les filtres linéaires transforment un ensemble de données d'entrée en un ensemble de données de sortie par une convolution bidimensionnelle qui est une opération mathématique.

Ils permettent de supprimer le bruit dans l'image. Chaque filtre a une taille $N \times N$ avec N impair. [10]

Les filtres linéaires les plus connus sont les filtres passe-bas, passe-haut.

A. Filtre passe-bas (lissage) :

Ce filtre n'affecte pas les composantes de basse fréquence dans les données d'une image, mais doit atténuer les composantes de haute fréquence. L'opération de lissage est souvent utilisée pour atténuer le bruit et les irrégularités de l'image. Elle peut être répétée plusieurs fois, ce qui crée un effet de flou. En pratique, il faut choisir un compromis entre l'atténuation du bruit et la conservation des détails et contours significatifs. [12]

$$\begin{array}{c}
 \begin{array}{ccc}
 1 & 1 & 1 \\
 1 & 1 & 1 \\
 1 & 1 & 1
 \end{array} \\
 \text{Masque de filtre}
 \end{array}$$

$1/9^*$

B. Filtre Passe-haut (Accentuation) :

Le renforcement des contours et leur extraction s'obtiennent dans le domaine fréquentiel par l'application d'un filtre passe-haut. Le filtre digital passe-haut a les caractéristiques inverses du filtre passe-bas. Ce filtre n'affecte pas les composantes de haute fréquence d'un signal, mais doit atténuer les composantes de basse fréquence. [12]

$$\begin{array}{c}
 \begin{array}{ccc}
 -1 & -1 & -1 \\
 -1 & 9 & -1 \\
 -1 & -1 & -1
 \end{array} \\
 \text{Masque de filtre}
 \end{array}$$

$H=$

I.9.1.2 Filtres non linéaire :

Ils sont conçus pour régler les problèmes des filtres linéaires, Leur principe est le même que celui des filtres linéaires, il s'agit toujours de remplacer la valeur de chaque pixel par la valeur d'une fonction calculée dans son voisinage. La différence majeure, est que cette fonction n'est plus linéaire mais une fonction quelconque (elle peut inclure des opérateurs de comparaisons ou de classification). Les filtres non linéaires les plus connus sont : [12]

a. Filtre médian

Ce filtre est très utilisé pour éliminer le bruit sur une image qui peut être de différentes origines (poussières, petits nuages, baisse momentanée de l'intensité électrique sur les capteurs, ...). L'avantage de ce filtre réside dans le fait qu'il conserve les contours alors que les autres types de filtres ont tendance à les adoucir. [12]

L'algorithme de filtre médian est le suivant (cf. Figure 1.16) :

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

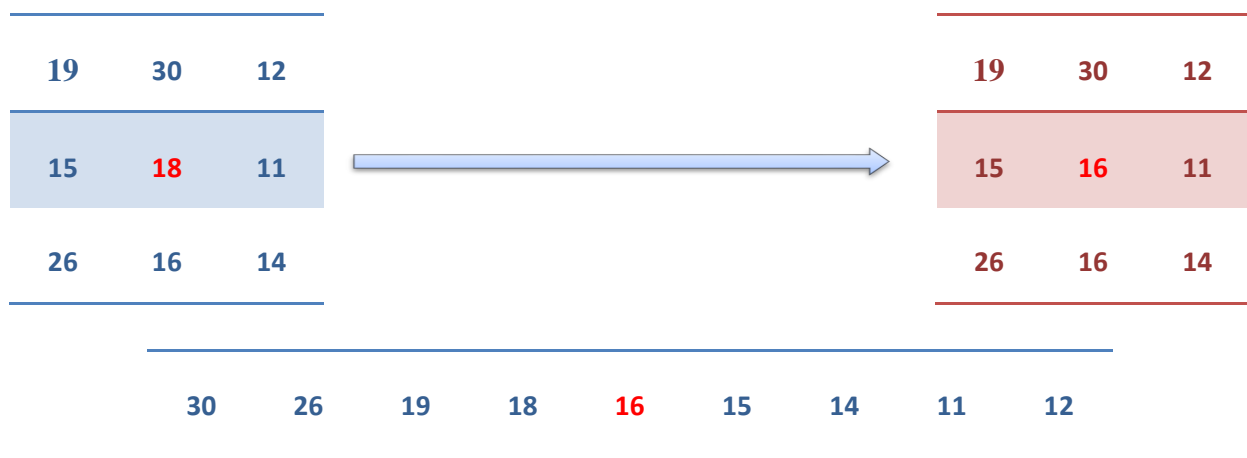


Figure I.16 : Principe du filtre médian.

b. Filtre maximum

On applique le même traitement que celui du filtre médian mais la valeur du pixel du centre comme la montre la figure I.17, va être changée par le maximum.

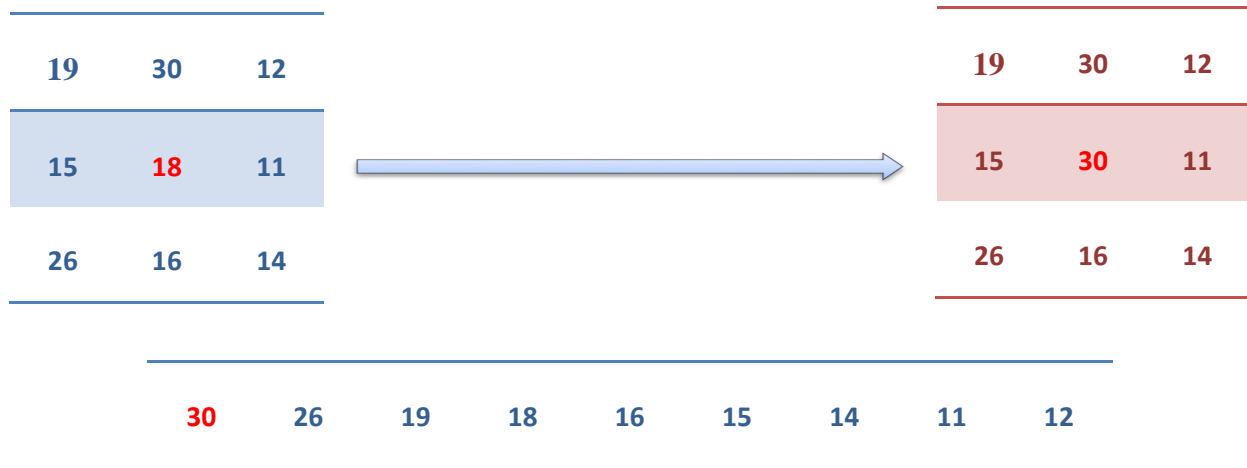


Figure I.17 : Principe du filtre maximum.

c. Filtre minimum

On applique le même traitement que celui du filtre maximum mais, cette fois, la valeur du pixel du centre comme la montre la figure va être remplacée par le minimum (cf. Figure 1.18). [12]

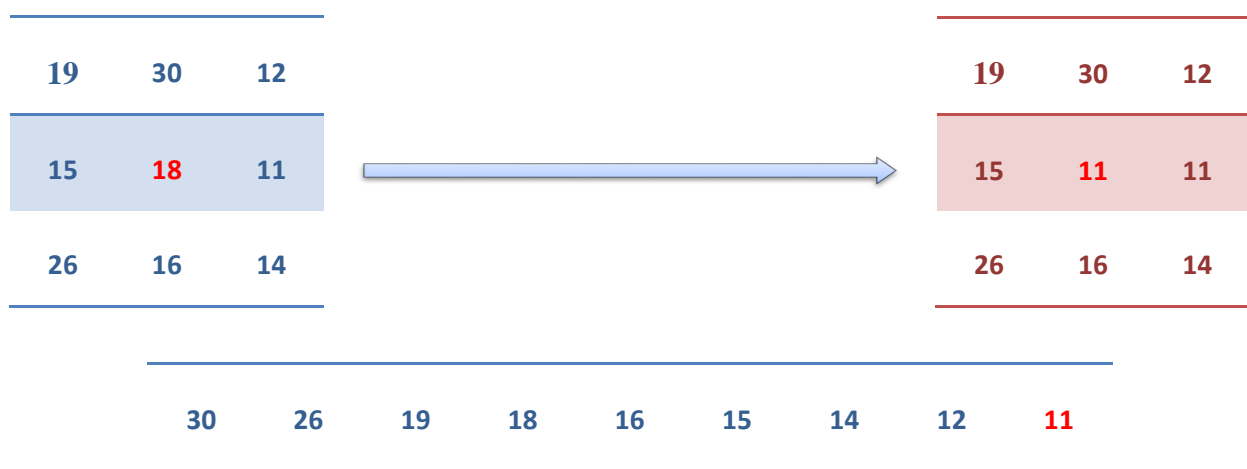


Figure I.18 : Principe du filtre minimum.

I.9.2 Avantages du filtrage :

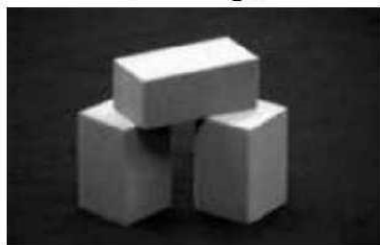
On citera comme avantage :

- Accentuer les variations d'intensité de l'image : renforcer les différences entre les pixels appartenant à des régions adjacentes (rehaussement de contraste).
- Détection de contours et préservation de la forme des régions.
- Renforcement de l'homogénéité à l'intérieur des régions (réduction de bruit). [13][14]

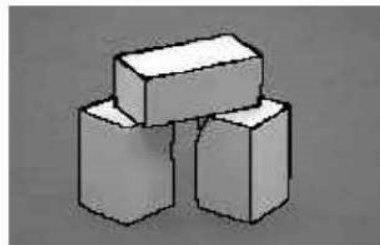
Cependant, Le résultat obtenu suite à un filtrage dépend énormément de la qualité du signal de l'image d'origine. Si le signal est très détérioré, le filtrage seul sera très peu efficace. Dans ce genre de situation le filtrage est juste une première étape dans un traitement plus complexe qui fera intervenir des informations sémantiques sur l'image.

I.9.3 Segmentation d'image :

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



(a)



(b)

Figure I.19 : Exemple d'une segmentation d'image, (a) : image d'origine, (b) : image segmentée.

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

I.9.4 Détection de contour :

La détection de contours est très utilisée comme étape de prétraitement pour la détection d'objets, pour trouver les limites de régions. En effet, un objet peut être localisé à partir de l'ensemble des pixels de son contour.

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

En pratique, le calcul de gradient est effectué par la convolution de filtres linéaires. Ces filtres peuvent estimer les dérivées premières, les dérivées secondes, etc., et se focaliser sur des contours de directions différentes. Il existe ainsi de nombreux filtres, qui ont des fondements mathématiques différents. Cependant, quelle que soit leur justification théorique, leur objectif reste le même : mettre en évidence les pixels qui ont une valeur très différente de leurs voisins.

I.9.5 Domaines d'application :

Le traitement d'images possède l'aspect multidisciplinaire. On trouve ses applications dans des domaines très variés tels que les télécommunications (T.V., vidéo, publicité,...), la médecine (radiographie, ultrasons,...), biologie, astronomie, géologie, l'industrie (robotique, sécurité), la météorologie, l'architecture, l'imprimerie, l'armement (application militaire).

De nouvelles applications pratiques sont possibles aujourd'hui et touchent tous les domaines d'activités, tels que : métiers du spectacle, de la radio, créations artistiques,... [11].

Dans le présent travail nous avons dû passer par un seuillage de l'image traitée, aussi nous explicitons ci-après le principe de cette méthode de prétraitement incontournable.

I.10 Seuillage d'une Image :

Le seuillage d'image est la méthode la plus simple de segmentation d'image. À partir d'une image en niveau de gris, le seuillage d'image peut être utilisé pour créer une image comportant uniquement deux valeurs, noir ou blanc (monochrome).

Le seuillage d'image remplace un à un les pixels d'une image à l'aide d'une valeur seuil fixée (par exemple 123). Ainsi, si un pixel a une valeur supérieure au seuil (par exemple 150), il prendra la valeur 255 (blanc), et si sa valeur est inférieure (par exemple 100), il prendra la valeur 0 (noir).[S1]

L'opération du seuillage a pour objectifs de :

- **Réduire la taille de l'image:** les pixels d'une image binaire ne peuvent prendre que deux valeurs (soit 0 ou 1). Le codage des pixels est « effectué sur 1 bits contrairement

aux images usuelles sur 8 bits. » On constate un rapport 8 concernant l'allocation mémoire des deux images

- **Faciliter les traitements** : il est beaucoup plus simple de traiter les images binaires contrairement aux images multi-niveaux (segmentation, labellisation, classification, etc.). Il permet de réduire l'espace des paramètres de l'image à traiter[15]

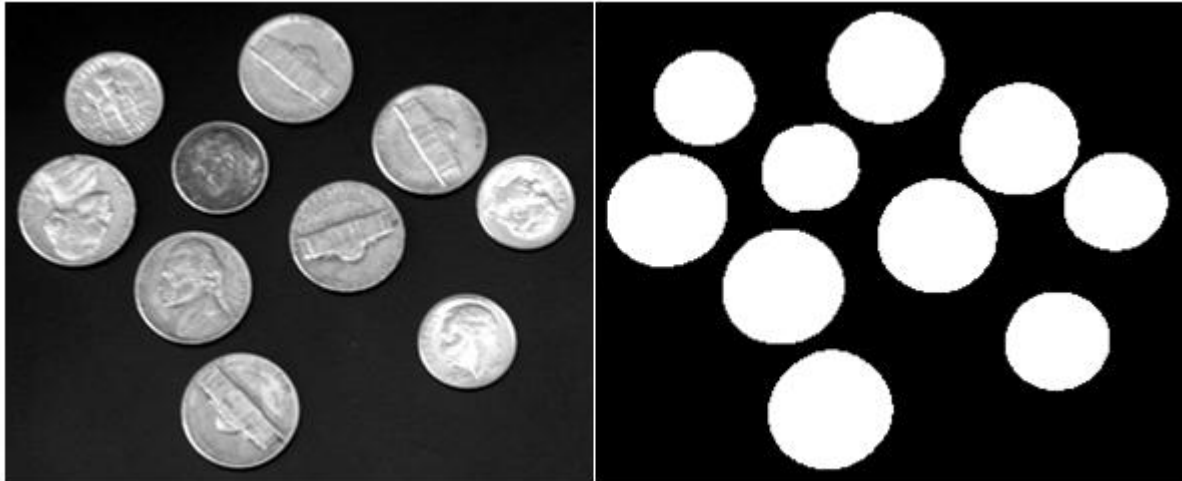


Figure I.20 : Une image aux niveaux de gris avant et après le seuillage

I.11 La vidéo :

I.12 Définition :

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

Elle peut être analogique (signal continu d'intensité de luminance) ou numérique (suite de trames ou images). [16]

I.13 Résolution temporelle d'une vidéo :

La résolution temporelle d'une vidéo est définie par le nombre d'images défilant par seconde. Afin d'éviter les désagréments dus aux papillotements, et prenant en considération les spécificités de l'œil humain qui garde une image environ 10 micro secondes au niveau de sa rétine ; les images doivent défiler à une certaine cadence:

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

I.14 Séquence d'image :

Une séquence d'images est une succession d'images bidimensionnelles qui montre l'évolution temporelle d'une scène. La cadence est de 25 images par seconde, ce qui correspond au seuil à partir duquel l'œil humain perçoit la séquence comme un stimulus continu, grâce à la persistance rétinienne. Par la suite, nous appellerons « trame » ou « plan » chaque image bidimensionnelle correspondant à un instant donné de la séquence.

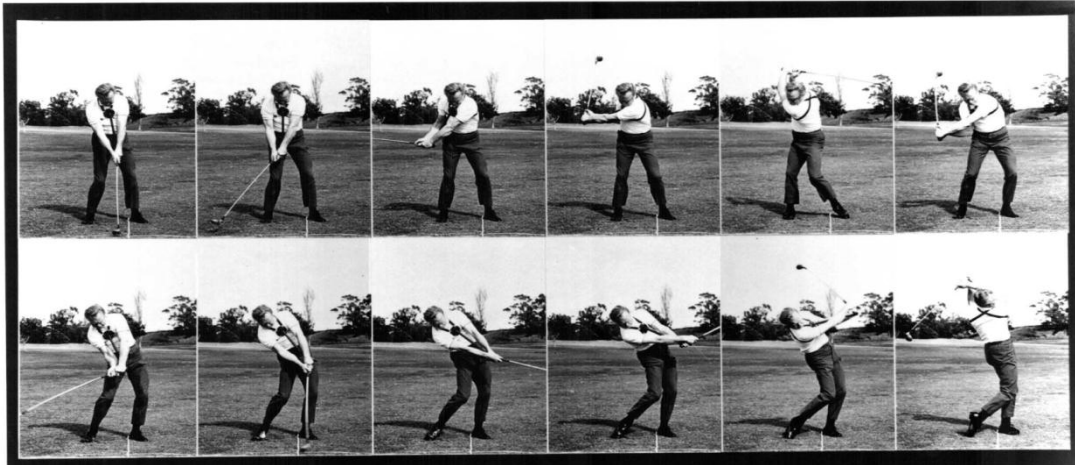


Figure I.21 : Exemple d'une séquence d'images.

I.15 Types de vidéo :

On distingue deux grandes familles de systèmes vidéo : les systèmes vidéo analogiques et les systèmes vidéo numériques.

I.15.1 La vidéo analogique :

La caméra balaye l'image bidimensionnelle qu'elle a devant elle par un faisceau d'électrons qui se déplace très rapidement de gauche à droite et plus lentement de haut en bas et produit une tension en fonction du temps. Elle enregistre ainsi l'intensité lumineuse, et à la fin du balayage, on a alors une trame. Le faisceau revient à l'origine pour recommencer. Le récepteur va recevoir cette intensité en fonction du temps, et pour reconstruire l'image, va répéter le processus de balayage.[17]

I.15.2 La vidéo numérique :

Le processus de captation de l'image vidéo en mode numérique est essentiellement le même que pour la vidéo analogique. Un système optique sépare la lumière en trois composants. À la sortie toutefois, plus d'enregistrement d'un signal électrique mais plutôt l'enregistrement d'une valeur numérique définie pour chacune des trois couleurs de chacun des pixels (RVB).

Le processus de transformation d'un signal électrique en valeurs numériques s'appelle la numérisation ou l'échantillonnage. Deux paramètres servent à échantillonner un signal électrique : la fréquence d'échantillonnage et la quantification de l'échantillonnage ou le nombre de bits utilisés pour le codage. [S2]

I.16 Les fondamentaux physiques et techniques :

I.16.1 La vision :

Dans le phénomène de la vision humaine, un flux lumineux composé de photons frappe l'œil. Dans l'œil, les bâtonnets vont réagir à l'intensité de la lumière (la luminance ou Y) et les cônes vont réagir à la couleur (la chrominance ou C).

Le cerveau effectue la synthèse de ces informations pour composer une image. L'œil humain est davantage sensible à l'intensité lumineuse (Y) qu'à la couleur (C). [S2]

I.16.2 La couleur :

La vidéo utilise ce qui est identifié comme la synthèse additive de la couleur. Dans le système colorimétrique additif, les trois couleurs primaires sont le rouge, le vert et le bleu (RVB ou RGB dans la terminologie anglophone). En combinant ces trois couleurs primaires, il est possible de reproduire tout le spectre visible par l'œil humain. L'écran de visualisation vidéo sera donc composé d'une série de triades rouge-vert-bleu. L'activation de l'ensemble de ces triades formera l'image [S2]

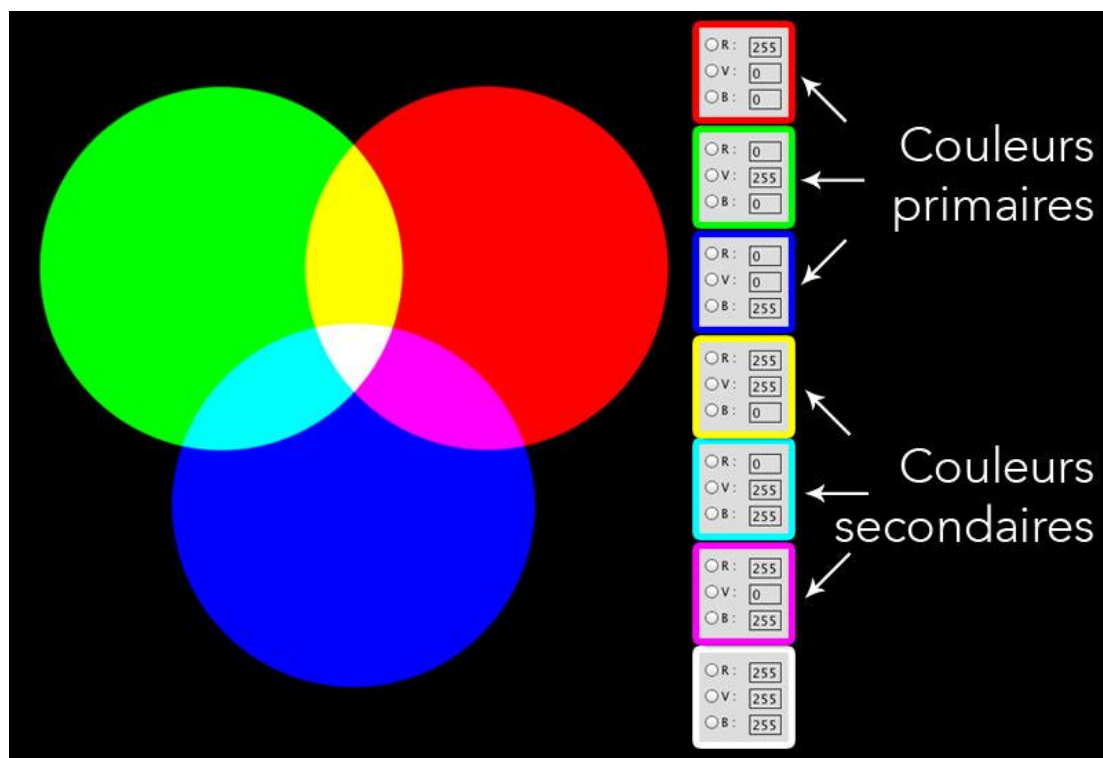


Figure I.22 : codage de couleur en RVB

I.16.3 Le signal :

En vidéo, c'est la caméra qui transforme l'information lumineuse (photons) en signal électrique (électrons) , en vidéo analogique, l'intensité de ce signal électrique varie de façon continue. Le processus de transformation de l'information lumineuse en signal électrique est le suivant :

- L'élément optique de la caméra, l'objectif, sépare la lumière en trois composants le rouge, le vert, le bleu. Cette opération est réalisée en faisant passer le flux lumineux par une succession de filtres dichroïques réfléchissant certaines couleurs et en laissant passer d'autres.
- En vidéo professionnelle, les trois images sont projetée sur trois capteurs photo sensibles distincts formés chacun de centaines de milliers de points généralement entre 400 000 et 700 000. Ces capteurs sont nommés CCD (charged coupled device) ou dispositifs à transfert de charge. Les caméras domestiques ne sont généralement équipées que d'un seul capteur CCD. [S2]



Figure I.23 : Schéma de principe d'un Capture Tri-CCD (charged coupled device) [S3]

- Pour chacun des points de chacun des capteurs, l'énergie lumineuse sera transformée en énergie électrique.

Ainsi, à la sortie des capteurs, trois signaux électriques d'intensité variable, un signal pour chacune des trois composantes. La lumière blanche est formée par la somme des trois composantes **RVB**.

Toutefois, elle n'est pas composée des trois couleurs primaires en quantité égale. Les propositions sont les suivantes :

$$100\% \text{ Y} = 29.9\%(\text{R}) + 58,7\%(\text{V}) + 11,4\%(\text{B})$$

- Une information supplémentaire devra être ajoutée à ces signaux. Comme nous l'avons dit, une valeur doit être établie pour chacun des points des capteurs, c'est cette valeur pour chacun des points de chacun des pixels qui permettra de recomposer l'image sur un écran de visualisation, chacun des points de la surface d'affichage pouvant alors prendre la même valeur qu'au moment de la capture par la caméra. La restitution de l'information pour chacun des pixels se fait par balayage ligne par ligne de la surface de l'écran de reproduction, les valeurs enregistrées pour chacun des pixels au moment de la capture étant transférées aux pixels de la surface de reproduction. Pour synchroniser ces balayages, un signal de fin de ligne est ajouté pour chacune des lignes. Un signal de fin d'image est également inséré. [S2]

I.17 Les caractéristiques d'un signal vidéo :

Chaque fichier vidéo a des attributs qui décrivent ce qui constitue le signal vidéo. Ces caractéristiques comprennent :

- + **Châssis de taille** : Il s'agit de la dimension de pixel de l'image
- + **Le ratio d'aspect** : C'est le rapport entre la largeur et la hauteur
- + **Vitesse de défilement**: C'est la vitesse à laquelle les images sont capturées et destinés à la lecture.
- + **Débit**: Le taux de débit ou de données est la quantité de données utilisées pour décrire la partie audio ou vidéo du fichier. Il est généralement mesuré en unités par seconde et peut être en kilo-octets, méga-octets ou giga-octets par seconde. En général, plus la vitesse de transmission, meilleure est la qualité.
- + **Le taux d'échantillonnage audio**: C'est à quelle fréquence le signal audio est échantillonné lors de la conversion d'une source analogique à un fichier numérique. [18]

I.18 objet vidéo :

I.18.1 Définition :

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

La notion d'objet ne fait pas forcément référence à un objet du monde réel. En effet, dans le domaine de la vidéo, un objet n'est pas nécessairement un objet d'une scène **3D** mais plutôt le résultat de l'analyse de la projection d'un monde **3D** sur un plan. Ainsi,

un objet vidéo est défini comme une région de la vidéo conforme à un modèle. On peut par exemple avoir pour modèle :

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

La notion d'objets est alors subjectivement définie par rapport aux modèles utilisés, On constate que pour pouvoir trouver des objets vidéo, il est nécessaire de proposer des modèles auxquels ces objets répondront [19].

I.18.2 Suivi d'objets dans la vidéo :

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

I.19 Conclusion :

J'ai voulu que ce chapitre soit une introduction générale ou une brève introduction aux concepts liés au domaine du traitement d'images et vidéos. Pour cela j'y ai abordé les principales définitions de ce domaine relatives aux caractéristiques de l'image numérique ainsi que les méthodes de traitement qui nous permettent d'améliorer sa qualité telle que la suppression de bruit par divers filtres, sans compromettre l'image d'origine, modifier son contenu ou en perdre des informations utiles. Une extension aux vidéos a été jugée nécessaire.

Nous pouvons maintenant discuter des différentes techniques de détection d'objets dans les photos et les vidéos dans le chapitre suivant.

CHAPITRE II

DÉTECTION ET SUIVI D'OBJETS

II.1 Introduction :

La détection d'objets est l'une des principales problématiques du traitement d'images. C'est une étape préliminaire de nombreuses applications, tels que le suivi d'objets, l'analyse comportementale, la compression vidéo, la reconstruction 3D, etc.

Il est évident que ce n'est qu'après avoir détecté les objets dans la séquence d'image que nous pouvons suivre le mouvement de ces derniers à l'aide des algorithmes de suivi.

Dans ce chapitre, nous discutons des modèles et des propriétés des objets et des méthodes de détection et de suivi de ces derniers.

II.2 Objet :

Les objets physiques sont les objets du monde réel qui apparaissent dans les scènes observées par les caméras. [20]

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

- **Les objets de contexte :**

Sont des objets physiques qui sont habituellement statiques (e.g. les murs). Dans le cas où ils ne sont pas statiques, leurs mouvements peuvent être prédits par les informations contextuelles. Par exemple les chaises, les portes sont des objets de contexte. [20]

- **Les objets mobiles :**

Sont des objets physiques qui peuvent être perçus dans les scènes par leurs mouvements [20].

Il est cependant difficile de prédire leurs mouvements par exemple les personnes, les véhicules.

II.3 Modélisation d'un objet [21] :

II.3.1 Caractéristiques d'un objet :

Dans le domaine du mouvement d'un objet dans une séquence, un objet peut avoir différentes caractéristique, il peut être :

- **Constant ou variable :** La forme, le mouvement, les couleurs ou les textures d'un objet varient ou non au cours du temps. Ces considérations jouent en faveur de l'adoption ou non d'un modèle de l'objet pour en améliorer le suivi.

- **Rigide ou non rigide** : pour la forme nous parlerons d'objet rigide (voiture) ou non-rigide (corps humain) c.à.d.la distance entre les points de l'objet est constante ou variable au cours du temps.
- **Unique ou multiple** : Dans le cadre d'un suivi labellisé où chaque objet porte une étiquette et n'en change pas durant l'opération de son suivi, une distinction entre chaque objet est nécessaire c.à.d. que chaque objet doit être unique. L'unicité repose, là aussi, sur un modèle. Cependant celui-ci n'est pas nécessairement connu au préalable. Cependant une connaissance des critères potentiellement discriminants permet une amélioration de la charge de calcul. La description de chaque objet selon des critères pertinents assure une labellisation fiable.

II.3.2 Représentation d'un objet :

Un objet, dans un scénario de suivi, est une entité indépendante pourvue de son identité spatiale (sa forme, son contour,...etc.) dans un environnement particulier. Cependant, on ne dispose pas toujours de toutes les informations qui caractérisent l'objet à suivre, mais on utilise un modèle de représentation de son état à surveiller. Une modélisation d'objets par catégories peut être présentée.

II.3.2.1 Modélisation par points :

L'objet à suivre est représenté soit par un point qui représente le centre de gravité de l'objet concerné, soit par un ensemble de points (cf. Figure II.1). En général cette façon de faire est efficace pour le suivi d'objets de petites tailles ou régions [22].

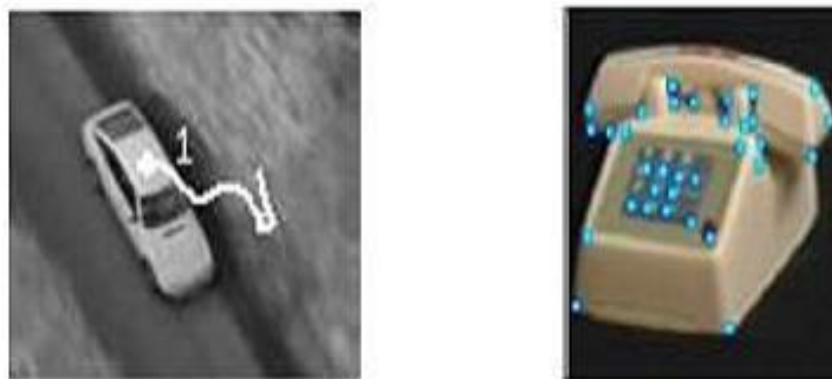


Figure II.1 : Points de contrôle

II.3.2.2 Modélisation par formes géométriques :

La forme d'un objet est représentée par un rectangle ou (boîtes englobantes) ou une ellipse (cf. Figure II.2). Le mouvement d'objet défini par cette représentation est souvent modélisé par une translation, projection et autre transformation, cependant ces formes géométriques sont plus appropriées à représenter les objets rigides simples, elles sont également employées pour suivre les objets souples.



Figure II.2 : Boîtes englobantes

II.3.2.3 Modélisation par silhouette et contour :

Par définition, un contour est la frontière qui sépare deux objets dans une image. La région à l'intérieur du contour s'appelle la silhouette de l'objet. La représentation par le contour et la silhouette est appropriée pour représenter des objets non rigides et les modèles déformables complexes (cf. Figure II.3).



Figure II.3 : Modèle silhouette

II.3.2.4 Modélisation par des modèles de formes articulées :

Les objets articulés sont composés de parties du corps qui sont liés avec des joints. Par exemple le corps humain est un objet articulé avec torse, jambes, mains, tête et pieds reliés par des joints. On peut modéliser ces parties en utilisant des cylindres ou des ellipses, les rapports entre eux sont régis par des modèles de cinématique de mouvement, par exemple l'angle commun etc (cf. Figure II.4).

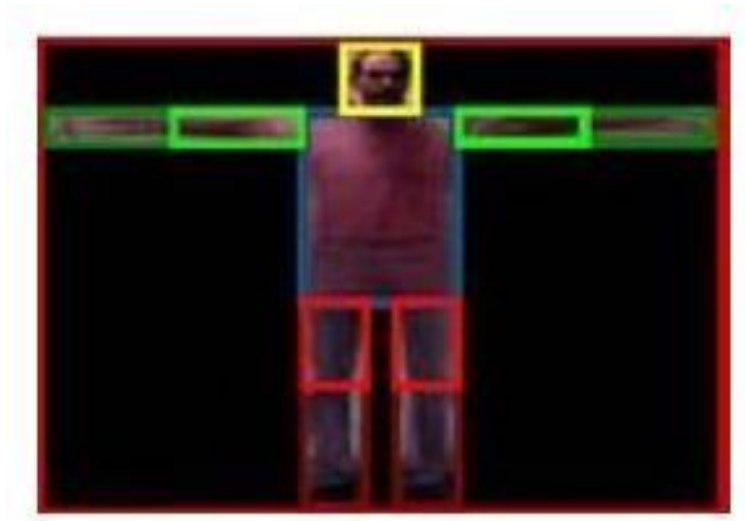


Figure II.4 : Modèle d'apparence articulé

II.3.2.5 Modèles squelettiques :

Le squelette d'objet peut être extrait en appliquant la transformation de l'axe médiane à la silhouette d'objet (cf. Figure II.5). Ce modèle est généralement utilisé pour la reconnaissance de l'objet et du suivi des modèles cinématiques complexes et la reconnaissance de mouvement.

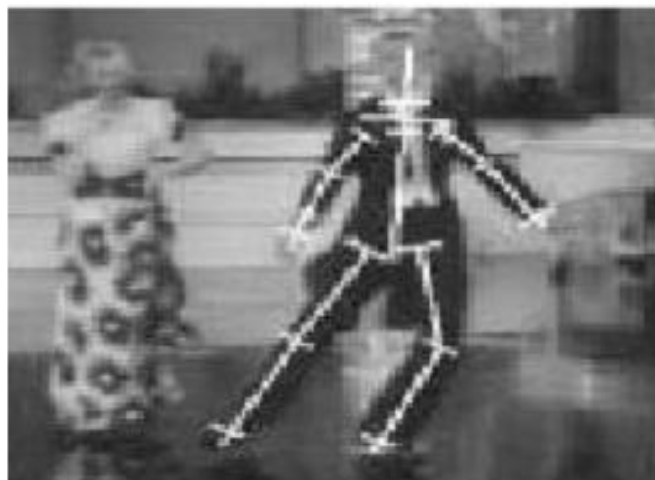


Figure II.5 : Modèle squelette

II.4 La détection de l'objet :

La détection de l'objet comprend la détection et la reconnaissance de formes dans le cadre d'une séquence vidéo. Un mécanisme de détection d'objet est nécessaire dans n'importe quel procédé de suivi soit dans chaque trame ou lorsque l'objet apparaît en premier dans la vidéo.

II.4.1 Méthode de Détection d'objets en mouvement :

Pour pouvoir suivre les mouvements dans la vidéo, la première étape est de détecter les mouvements, cette étape joue un rôle très important dans l'analyse vidéo.

Dans cette partie, on va passer en revue cinq méthodes de détection de mouvement.

II.4.1.1 La différence temporelle des images (méthode de mouvement) :

La différence temporelle détecte la région de mouvement grâce à la différence de pixel par pixel de deux trames consécutives dans un flux vidéo (cf. Figure II.6) [23][24].

Cette méthode adapte le changement de la scène, mais elle a peu d'efficacité parce que dans une durée du temps Δt , on peut détecter seulement une partie d'objet, par exemple : la main, la tête etc. mais aussi la région d'où l'objet s'est déplacé est de même détectée.

Dans ce cas-là, c'est très difficile d'extraire les propriétés de l'objet mobile (la taille, la position, la vitesse etc.) et à suivre son mouvement. [25]

Une amélioration de cette méthode consiste à extraire trois images de différence au lieu de deux permettant de décider si un pixel a bougé.

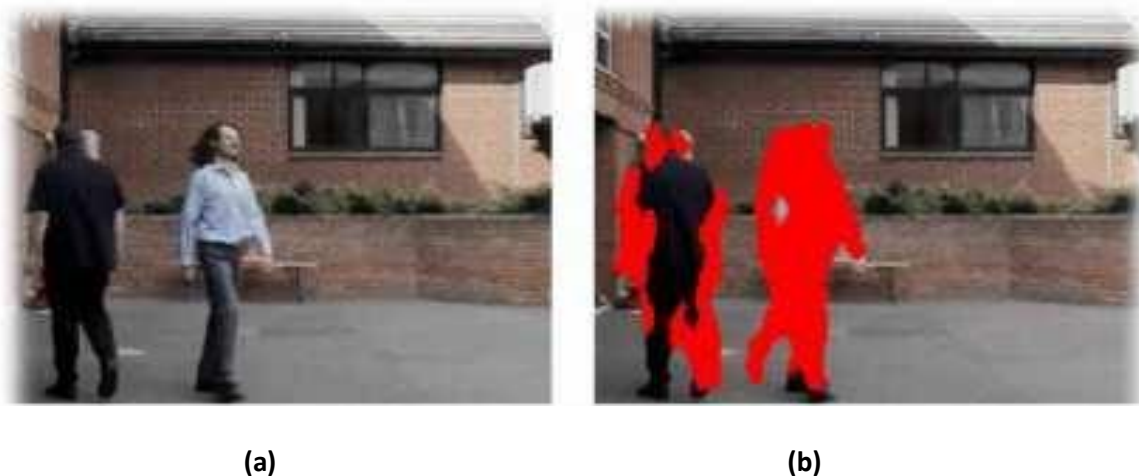


Figure II.6 : Un exemple de la différence temporelle [25]. **(a)** une scène simple avec deux objets, **(b)** les régions rouges sont la différence entre deux images consécutives.

L'idée principale de cette méthode est la suivante :

Soit I_t l'image à l'instant t et I_{t-1} l'image à l'instant $t-1$. L'objet du mouvement se compose les pixels qui satisfont l'équation suivante :

$$\max (| I_t(x, y) * c - I_{t-1}(x, y) * c |, c=(R,G,B)) \geq \text{seuil} \quad (1)$$

On peut améliorer cette méthode de la façon suivante :

Au lieu de soustraire l'image à l'instant $t-1$, on soustrait la moyenne de N dernières images [24], Soit I_{tm} la moyenne de N dernières images à l'instant t . L'objet du mouvement se compose les pixels qui satisfont l'équation suivante :

$$\max (| I_t(x, y) * c - I_{tm}(x, y) * c |, c=(R,G,B)) \geq \text{seuil} \quad (2)$$

La moyenne de N images à l'instant $t+1$ est mise à jour :

$$I_{t+1}(x, y) * c = \alpha I_t(x, y) * c + (1-\alpha) I_{tm}(x, y) * c, c=(R, G, B) \quad (3)$$

Où $\alpha \in (0,1)$ est une constante et est décidé par la pratique.

Avantages

- Adaptation au changement de la scène.
- Détection de la région en mouvement.

Inconvénients

- Ne permet pas de détecter le mouvement dans les zones uniformes intérieures à l'objet
- Ne fonctionne pas dans plusieurs cas, pour différentes raisons, telles que : la présence de bruit du capteur et les changements de luminosité de la scène qui modifient les intensités des pixels.

II.4.1.2 Double de différence temporelle et caractère de contour :

Dans cette méthode, on utilise aussi la différence de pixel par pixel des trames consécutives dans un flux vidéo comme la 1^{ère} méthode. Mais on va utiliser trois trames consécutives. [26] Cette façon de faire nous donne de meilleur résultat que celui de la 1^{ère} méthode, de plus elle s'adapte aussi au changement de la scène.

L'idée principale de cette méthode est la suivante:

Soit I l'image à l'instant t , I l'image à l'instant $t-1$ et I l'image à l'instant $t-2$.

L'objet du mouvement se compose les pixels qui satisfont l'équation suivante :

$$I_1(x, y) = \max (| I_t(x, y) * c - I_{t-1}(x, y) * c |, c=(R,G,B)) \geq \text{seuil} \quad (4)$$

$$I_2(x, y) = \max (| I_{t-1}(x, y) * c - I_{t-2}(x, y) * c |, c=(R,G,B)) >= \text{seuil} \quad (5)$$

$$I_{\text{résultat}}(x, y) = I_1(x, y) \cup I_2(x, y). \quad (6)$$

Nous extrayons la région des objets mouvants par la double méthode de différence. Dans le cas où l'objet se déplace lentement ou il qu'il n'y a qu'une partie de l'objet qui se déplace, nous ne pouvons pas obtenir complètement la forme de l'objet. Un objet peut être divisé en plusieurs régions. Nous utilisons donc le caractère de contour pour combiner ces régions.

Après avoir masqué l'image à l'instant t-1 avec la région mouvante obtenue à partir de la double méthode de différence, le contour est calculée dans la région où le mouvement est produit.

Le contour est calculé à partir de l'image F qui est l'image à l'instant t-1 masquée avec $R = I_1(x, y) \cup I_2(x, y)$. Le contour est représenté par :

$$G = \begin{bmatrix} F_x^2 & F_x F_y \\ F_x F_y & F_y^2 \end{bmatrix} \quad (7)$$

En utilisant un caractère de gradient avec l'intensité de l'image masquée F. En chaque pixel que le mouvement est détecté par l'intermédiaire de la double méthode de différence dans l'armature courante, le coefficient est calculé. C'est-à-dire, les pixels commandés par le caractère maximal de contour sont choisis en tant que points de caractère de contour parce que les valeurs propres minimum plus grandes sont les caractères plus forts de contour. [25]

Avantages :

- Le résultat de cette méthode est meilleur par rapport la méthode précédant.
- Détecte la région de mouvement.
- Utilise trois trames.

Inconvénient :

- Si l'objet ne se déplace pas pendant quelques secondes, cette manière ne peut pas détecter le mouvement.

La Figure (II.7) représente la possibilité de détecter le mouvement d'une voiture zn détectant le changement de position des pixels sélectionnés par rapport à l'arrière-plan dans une certaine période de temps

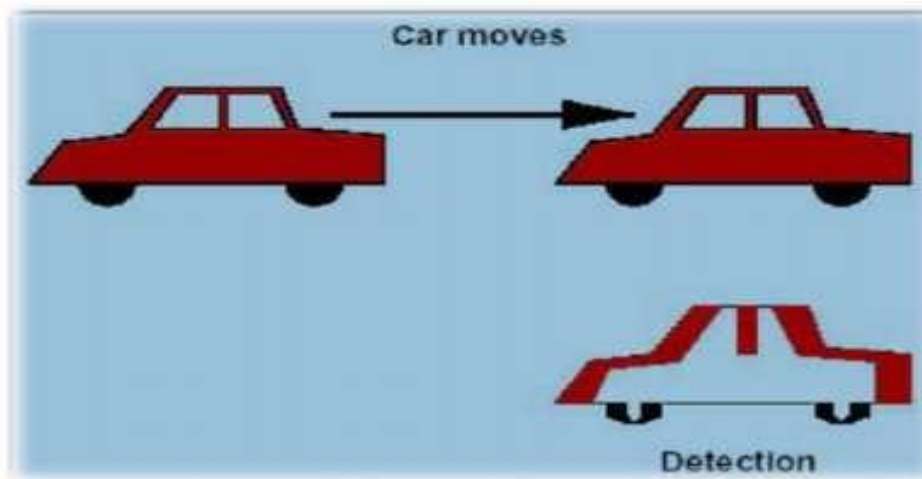


Figure II.7 : Détection de mouvement par différence temporelle [27]

II.4.1.3 La soustraction de l'image de fond (méthode de différence) :

La troisième méthode utilise l'image d'arrière-plan, elle est très populaire en détection de mouvement, et de nombreuses applications l'utilisent, elle dépend de la modélisation de fond et de cette façon elle détecte la zone de mouvement en soustrayant l'image courante pixel par pixel de l'image de fond (cf. Figure II.8).

Il y a plusieurs conditions nécessaires pour utiliser ces techniques, tout d'abord la caméra doit rester fixe et des parties du fond doivent rester bloquées par des objets en mouvement temporairement minoritaires, de plus il est préférable d'avoir au début de la séquence un certain nombre de des images sans objets en mouvement afin qu'on puisse reconnaître correctement le fond.

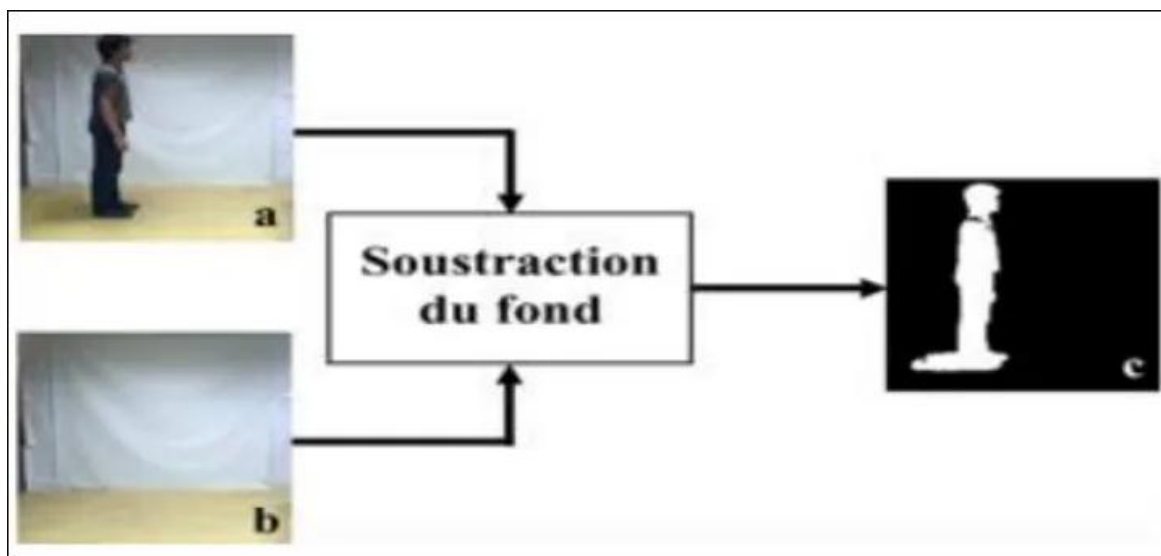


Figure II.8 : Détections basées sur la modélisation du fond

Soit I_t l'image à l'instant t . B_t est l'image de fond à l'instant t . L'objet du mouvement se compose des pixels qui satisfont l'équation suivante :

$$\max (|I_t(x, y)_c - B_t(x, y)_c|, c = (R, G, B)) > = S(x, y) \quad (8)$$

Où $S_t(x, y)$ est le seuil de la position (x, y) à l'instant t . [25]

✚ Création d'image de fond :

Il existe trois manières pour créer l'image de fond :

- Le calcul de moyenne de N premières images.
- Choisir l'image de fond qui a peu de changement ou qu'il n'y a pas de mouvement pendant une durée longue.
- Choisir la première image dans le flux vidéo. [S4]

Les Avantages :

- Cette méthode est très populaire.
- Utilisée par plusieurs applications.
- Détecte l'objet complètement.

Inconvénient :

- Cette technique se limite aux caméras en position fixe.

II.4.1.4 Méthode de détection des objets selon la Couleur :

La quatrième méthode est basée sur la couleur d'objets détectés. On procède à la lecture par pixel de sa valeur en rouge vert et bleu. Puis, on teste si la valeur du rouge, du vert et du bleu est supérieure ou inférieure au seuil imposé suivant la couleur à reconnaître. [28]

❖ Test

Condition de supériorité ou d'infériorité sur chaque couleur.

❖ Paramètres

Les paramètres principaux sont les différents seuils des couleurs (RGB).

Dans l'exemple utilisé, les conditions sont $R > 120$, $G < 100$, $B < 100$. [S5]

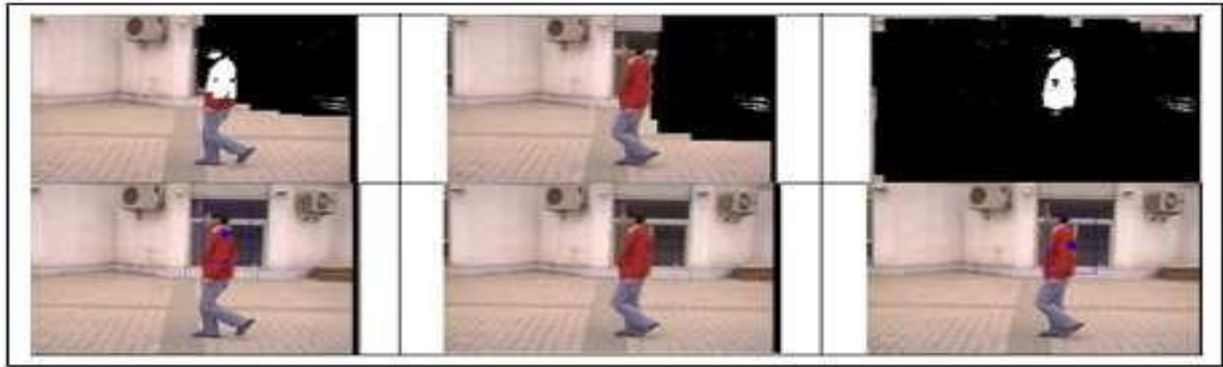


Figure II.9 : Exemple d'erreur de détection de la couleur rouge produit par un seuil mal choisi. [S5]

Dans les figures II.9 et II.10 l'image de dessous représente l'image source et l'image du dessus le résultat de la détection.

Sur les images de la figure II.9 une erreur apparait, l'objet n'est pas reconnu sur le frame du milieu, cette erreur est dû au seuil appliqué a la moyenne de la particule, en effet si la moyenne est trop faible alors elle est considérée comme nulle.

Le même exemple est repris avec l'ajout de ce seuil donne le résultat de la figure II.10. [S5]

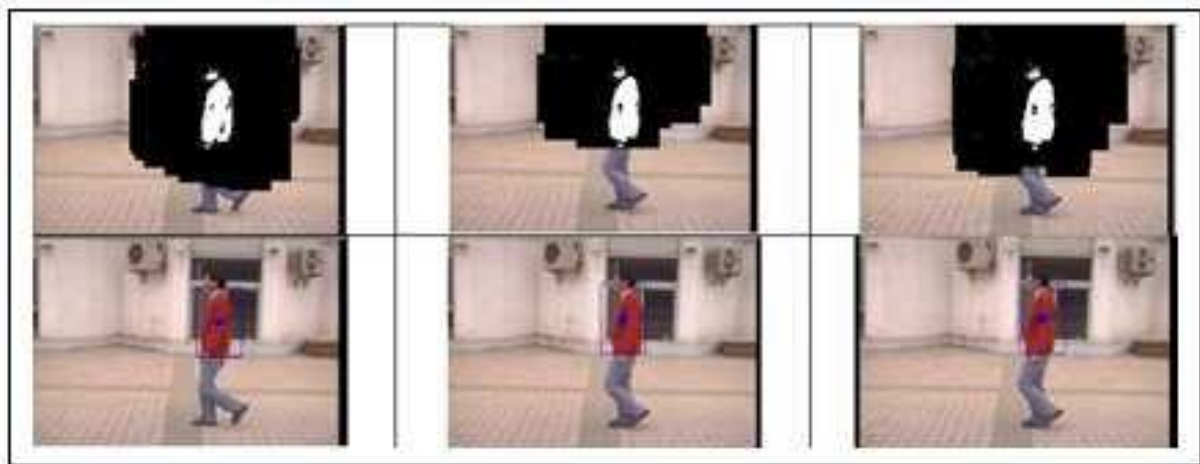


Figure II.10 Bonne détection de la couleur rouge après modification du seuil. [S5]

Avantages :

- C'est une méthode assez simple à mettre en œuvre, il suffit de reconnaître une couleur qui ressort par rapport au reste de l'image.

Inconvénients :

- Il existe beaucoup de nuances de couleur (256^3) il faut alors avoir une grande base de données puis choisir la couleur ou définir a chaque fois la couleur souhaitée ce qui sous entend que ces valeurs soit connues.

- Si un autre objet possède la même couleur on ne peut les différencier, sauf s'il y a une grande différence de taille entre les deux objets et que l'objet à reconnaître est le plus grand.
- Si notre objet change de couleur au cours de la vidéo, la couleur à reconnaître devra également être modifiée.

II.4.1.5 Les détections utilisant la notion de cohérence :

Une dernière approche consiste à définir un objet mobile comme une région ayant un mouvement cohérent. Dans ce contexte, un mouvement cohérent peut être défini comme un mouvement ayant de grande chance de provenir d'une "cible" classique (personne, véhicule). Cette définition a été utilisée pour la segmentation de mouvement en ajoutant l'hypothèse suivante: un objet avec un mouvement cohérent se déplace dans une direction approximativement constante pendant une courte période (en pratique quelques images).

Les modèles de mouvement dérivés du flot optique sont utilisés comme primitives pour la détection ou le suivi. Le flot optique a pour rôle de décrire le mouvement cohérent des points entre des images successives, des mesures de flot optique cohérentes en direction sont accumulées pendant quelques pas de temps. Une estimation du déplacement de chaque pixel à travers une séquence d'images est ainsi obtenue. Il est alors possible de distinguer les objets mobiles, qui se déplacent avec une direction constante, des mouvements parasites. La différence et l'avantage ici est que seuls les objets cohérents sont détectés.

Cela permet l'application directe d'autres traitements tels que l'identification ou le suivi.

II.5 Le suivi des objets :

II.5.1 But du suivi [28] :

Le suivi de plusieurs objets dans une séquence d'images peut s'avérer difficile selon les scènes complexes qu'il traite, cette complexité réside là où les objets peuvent avoir des tailles différentes. Ces objets peuvent être rigides ou non-rigides et s'occulter les uns les autres de ce fait, des méthodes de suivi sont apparues, le but de ces méthodes est d'estimer au fil du temps les paramètres d'une cible (ou plusieurs) présente dans le champ de vision de la caméra et initialement détectée par un moyen quelconque. Les paramètres peuvent être divers : position dans l'image, à laquelle peuvent s'ajouter la taille et l'orientation apparente, l'attitude, l'apparence, etc.

II.5.2 Méthodes de suivi :

Les méthodes existantes pour le suivi simultané de plusieurs objets dans une séquence d'images impliquent généralement que des caractéristiques telles que des histogrammes

couleurs, des mesures de corrélation, des estimations de vitesse, ou des mesures de distance entre régions puissent être utilisées pour suivre chaque objet.

Plusieurs classifications des méthodes de suivi d'objets ont été proposées dans la littérature [28] [29] qui peuvent être divisées en trois principales catégories:

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

II.5.2.1 Suivi par appariement de détections [29] :

Le suivi se fait alors en assignant les observations aux pistes en cours d'estimation. Ces méthodes, dites "detect-before-track" sont très populaires en pistage sonar et radar. Elles peuvent être déterministes ou probabilistes. On est confronté dans les deux cas à un problème combinatoire d'association.

❖ Méthodes déterministes :

Le principe des méthodes déterministes est d'associer les observations aux pistes en cours de suivi en minimisant une distance calculée sur certaines caractéristiques de l'objet. Les caractéristiques des objets couramment utilisées sont les apparences qui peuvent être des densités (histogrammes de couleur ou de contour), une carte de contours (contour ouvert ou fermé de l'objet) ou une combinaison de ces modèles.

❖ Méthodes probabilistes :

Les observations obtenues par un algorithme de détection sont très souvent corrompues par du bruit. De plus, le mouvement ou l'apparence d'un objet peut légèrement varier entre deux images consécutives. Les méthodes probabilistes permettent de gérer ces variations en ajoutant une incertitude au modèle de l'objet et aux modèles des observations. Le suivi d'une seule cible est alors obtenu par des méthodes de filtrage (**filtres de Kalman, filtrage particulaire**). Le suivi de plusieurs objets peut lui aussi se faire avec ces méthodes de filtrage mais une étape préalable d'association de l'objet avec l'observation la plus probable doit être ajoutée.

II.5.2.2 Le filtre de Kalman :

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

Il permet l'estimation de variables d'état d'un système (position/vitesse, par exemple, pour des systèmes mécaniques ou encore tension/intensité, par exemple, pour des systèmes électriques...) et il fournit aussi la variance de l'erreur de l'estimation.

Son appellation « filtre » vient du fait qu'il permet, entre autre, d'estimer un signal bruité (bruit blanc), en séparant le signal de son bruit.

Le filtre de **Kalman** s'applique aux signaux non stationnaires, il résout le problème de filtrage linéaire et se présente sous la forme d'un ensemble d'équations différentielles ou récurrentes plus faciles à résoudre sur calculateur numérique en présence de condition initiales et d'entrées déterministes.

Ce filtre est utilisé en trajectographie pour estimer une trajectoire comme le cas de Poursuite de cibles (avions, missiles, radars...) en navigation pour localiser un engin, ou encore en balistique pour estimer des conditions initiales inconnues.

Il est également employé lorsque l'on veut coupler des informations venant de sources différentes afin de retrouver, par exemple, une trajectoire.

Le Filtre de **Kalman** est une approche statistique, d'assimilation de données, dont le principe est de corriger la trajectoire du modèle en combinant les observations avec l'information fournie par le modèle de façon à minimiser l'erreur entre l'état vrai et l'état filtré.

II.5.2.3 Segmentation dynamique [28] :

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

Ces approches reposent sur une succession de segmentations.

Elles font généralement évoluer le contour de l'objet à l'instant précédent jusqu'à sa nouvelle position à l'instant courant (cf. Figure II.11).

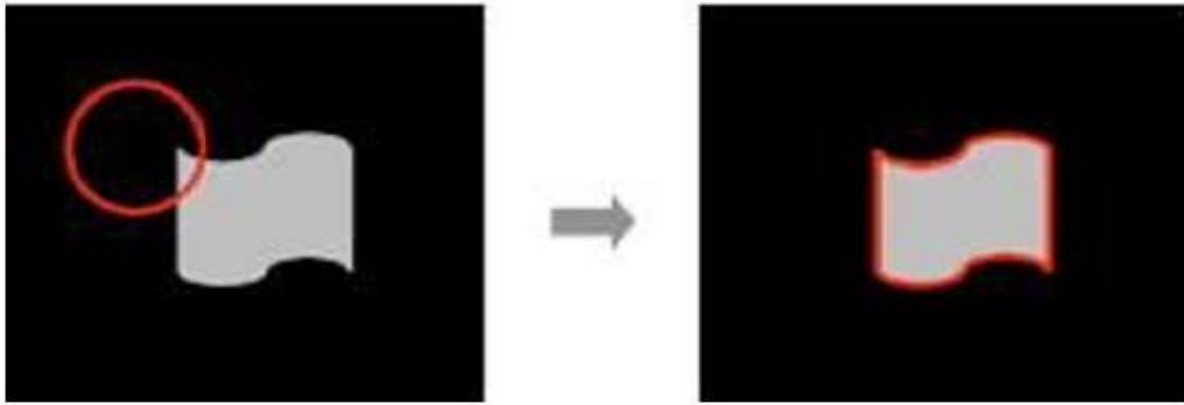


Figure II.11 : Contour actif

II.5.2.4 Suivi déterministe de fenêtre englobante par détection séquentielle [29] :

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

1. Suivi différentiel d'image :

Cette approche consiste à chercher la portion d'image la plus proche dans l'image courante de celle à l'instant précédent. La recherche se fait souvent autour de la position de l'objet à l'instant précédent. Ces approches sont appelées "block matching" ou "template matching". Elles font appel à une mesure de similarité telle que la corrélation ou la SSD ("sum of square differences"). Les intensités et les couleurs étant sensibles aux changements d'illumination, il est aussi possible d'exploiter les gradients de l'image. Le coût de calcul de ce type de méthodes peut être élevé. Il dépend de la taille du voisinage dans lequel l'image est recherchée.

2. Suivi de distributions :

L'utilisation de distributions dans la recherche de la cible à l'instant courant se base alors sur des distributions (histogrammes ou mélanges de gaussiennes) des couleurs dans une région de géométrie simple. La méthode la plus utilisée dans cette catégorie est le suivi par mean shift. La distribution est un histogramme de couleur (cf. Figure II.12).

L'algorithme consiste alors à déplacer une fenêtre d'analyse (noyau) de manière à trouver l'histogramme contenu dans la fenêtre qui coïncide le mieux avec l'histogramme de référence.

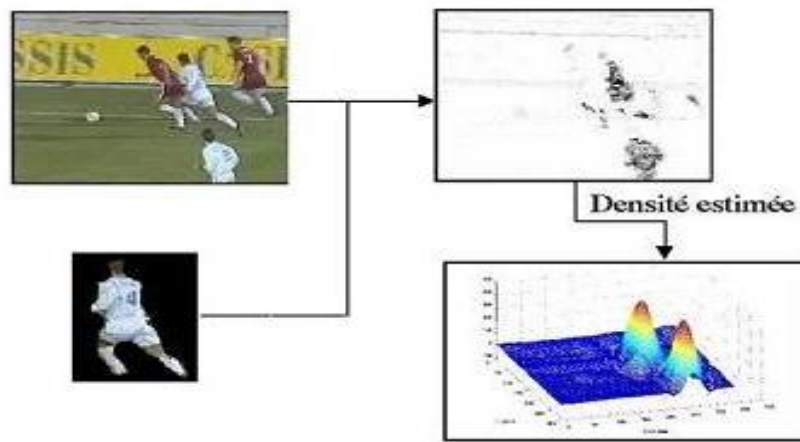


Figure II.12 : Suivi de distribution

II.6 Caractéristiques d'un bon algorithme de suivi [28] :

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

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

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

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

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

Enfin, la gestion des occultations n'est bien considérée qu'avec l'utilisation d'observations ou en ajoutant une loi dynamique au processus de suivi,

la difficulté étant alors de déterminer cette loi. La plupart du temps les objets peuvent avoir un mouvement quelconque qu'aucune loi ne peut caractériser.

II.7 Conclusion :

Les différents algorithmes étudiés présentent des avantages et des inconvénients pour différentes situations.

Pour notre cas, étant en phase d'initiation et pour des contraintes de temps, nous avons choisi de détecter et de suivre des objets à couleur unique. Ceci fera l'objet du chapitre suivant.

Chapitre III

Résultats et commentaires

III.1 Introduction :

Après avoir abordé dans les deux chapitres précédents les connaissances théoriques des bases du traitement d'images et de vidéos et diverses techniques de détection et de suivi d'objets en mouvement dans la vidéo, je vise maintenant dans ce chapitre à appliquer l'une de ces techniques (détection et suivi d'objets selon leur couleur en temps réel) en utilisant les algorithmes du programme MATLAB, En raison de la facilité de programmation qu'il offre grâce à ses commandes, et il contient des boîtes à outils spécialisées qui facilitent le traitement des images, en plus de la possibilité de développer une interface graphique simplifiée GUI (Graphical User Interface) à l'aide de l'outil « GUIDE ».

Dans la première partie, je présenterai un aperçu du matériel et des logiciels utilisés, puis j'aborderai les résultats de la détection et du suivi et terminerai par une conclusion en guise de synthèse.

III.2 Environnement du travail :

III.2.1 Environnement matériel ou le hardware :

UN laptop :

- ❖ Manufacturer: SAMSUNG.
- ❖ Model: NP300E5X.
- ❖ Processor : Intel(R) Core (TM) i3-2310M CPU @ 2.10 GHZ, (4CPUs), ~ 2.1GHz
- ❖ RAM : 4096MB
- ❖ Disque Dur: 500 GO.

Webcam intégrée:

- ❖ Modèle : SC-03FFL11739P
- ❖ Résolution : 160x120. Pixels
- ❖ Tout objet coloré en Rouge, Vert, Bleu et Blanc (jouet, cartes, boites...etc.)

III.2.2 Environnement immatériel ou le software :

- ❖ System d 'exploitation : Windows 8 professional N 32-bit (6.2, version 9200).
- ❖ Fabricant du system : SAMSUNG ELECTRONICS CO.,LTD.
- ❖ Carte graphique : intel® HD Graphics 3000.
- ❖ Logiciel : MATLAB R2013a

J'ai choisi d'utiliser MATLAB en raison de mon expérience de l'utiliser dans de nombreuses applications au cours de ma formation universitaire.

III.3 MATLAB :

Comme nous le savons, il s'agit d'un programme produit par la société MATWORKS, ciblant les applications d'ingénierie et mathématiques, permettant à son utilisateur de manipuler mathématiquement des matrices en plus de représenter graphiquement des fonctions mathématique, d'implémenter divers algorithmes, de créer des interfaces graphiques et même de communiquer avec des programmes écrits dans d'autres langages comme C++, java...etc.

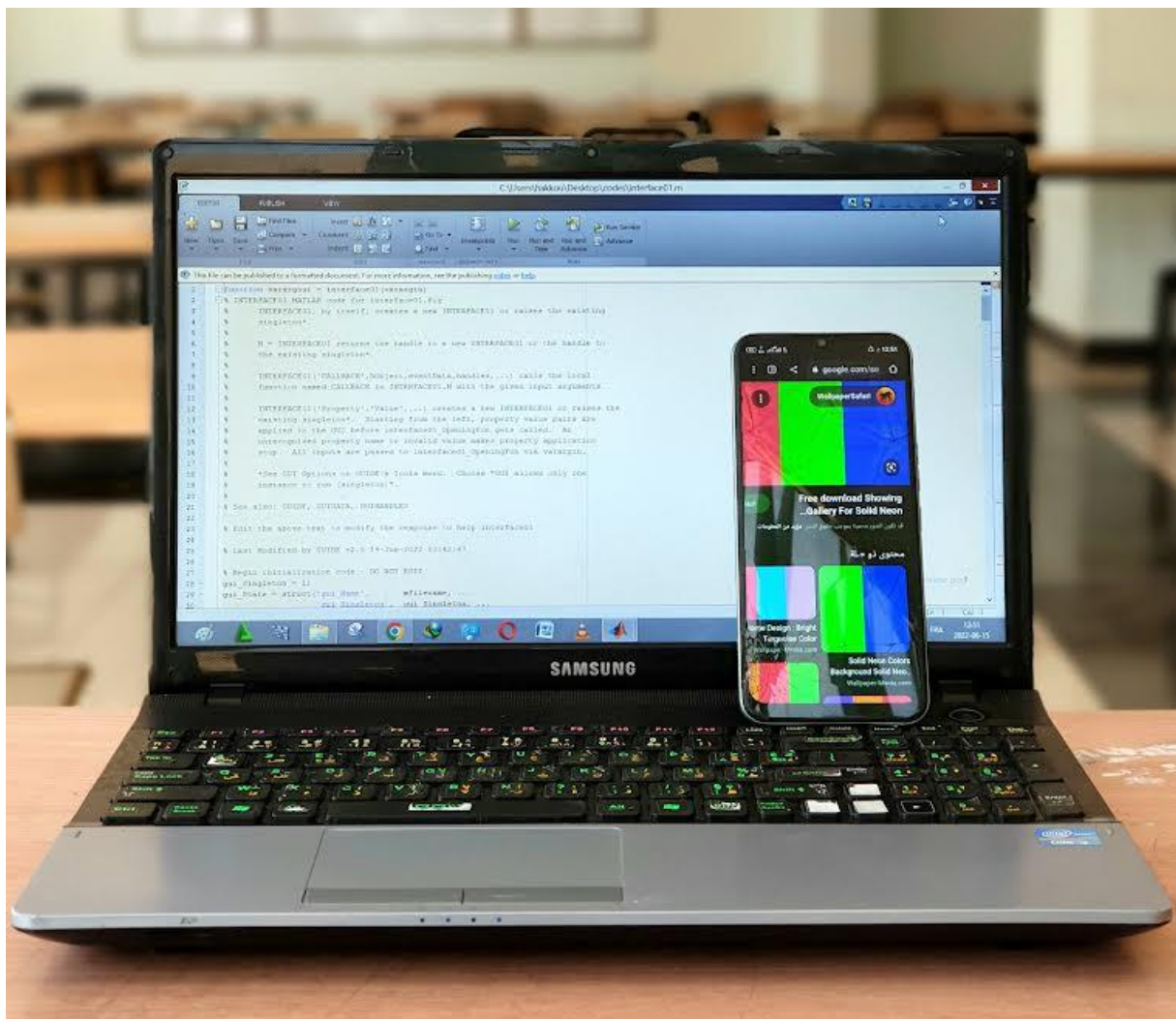


Figure III.1 : le matériel utilisé dans ce travail

III.4 L'interface graphique (GUI) :

Je voulais utiliser une interface graphique simple pour exécuter le programme en quelques clics sans avoir à entrer le code principal chaque fois que nous devons l'exécuter.

La commande **guide** sur MATLAB fournit un environnement simple pour développer des interfaces utilisateur graphiques car elle fournit des outils dédiés à la conception d'interfaces et à leur liaison avec le code de programmation, et nous pouvons facilement modifier les propriétés de ces outils. Par exemple, la fonction **callback** nous permet de programmer chaque bouton individuellement pour effectuer une action spécifique, après avoir terminé le travail sur l'interface Le **guide** génère automatiquement le code MATLAB.

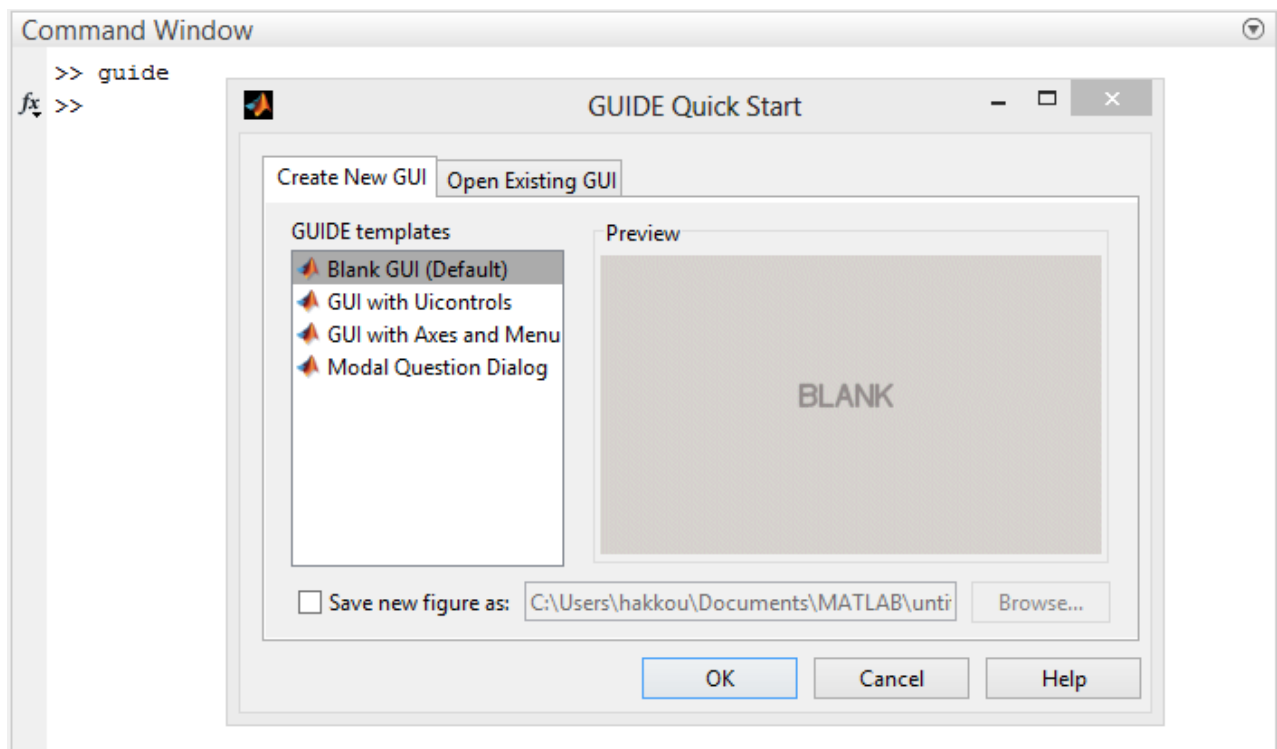


Figure III.2 : Comment entrer dans l'environnement de développement 'guide'

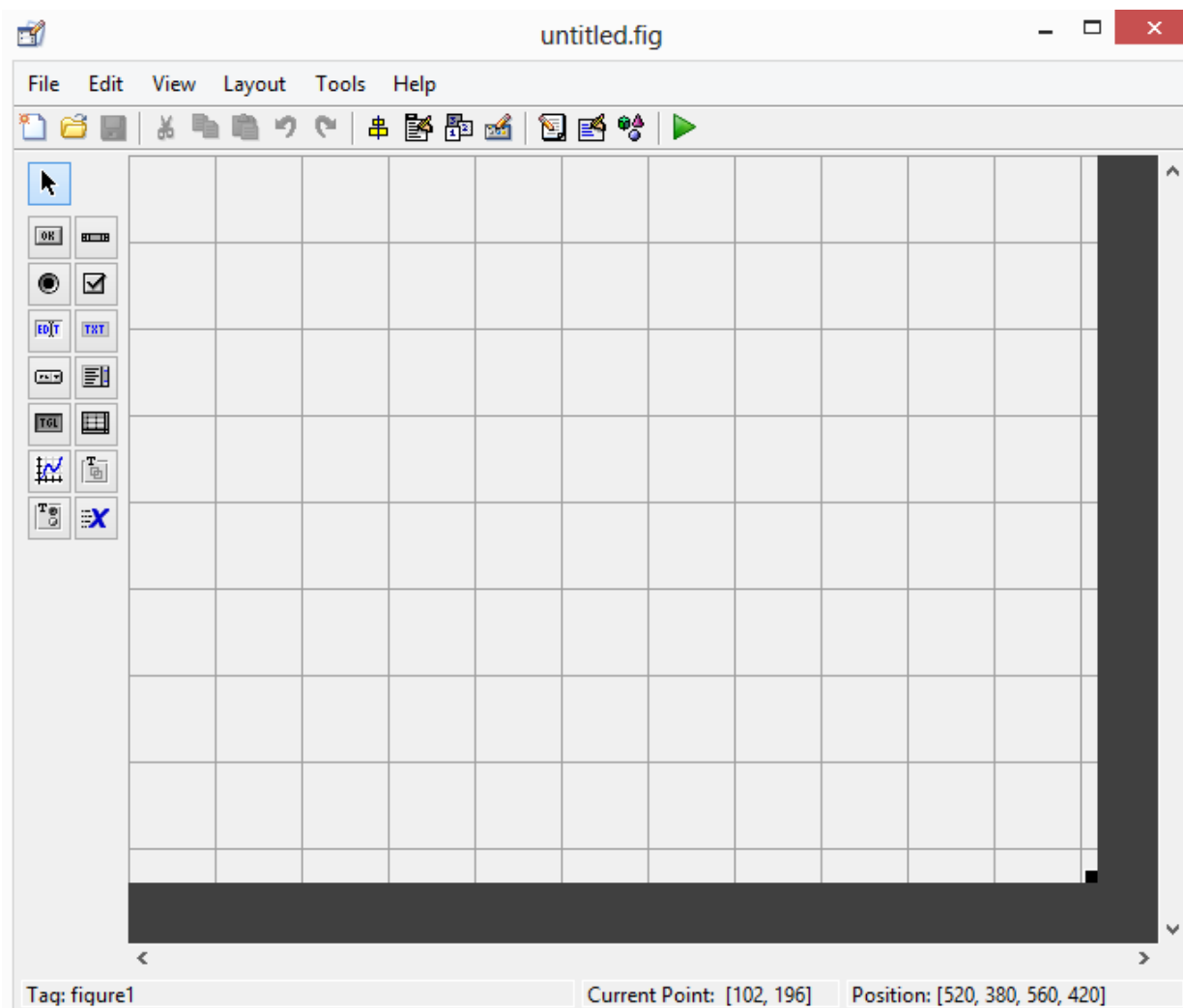


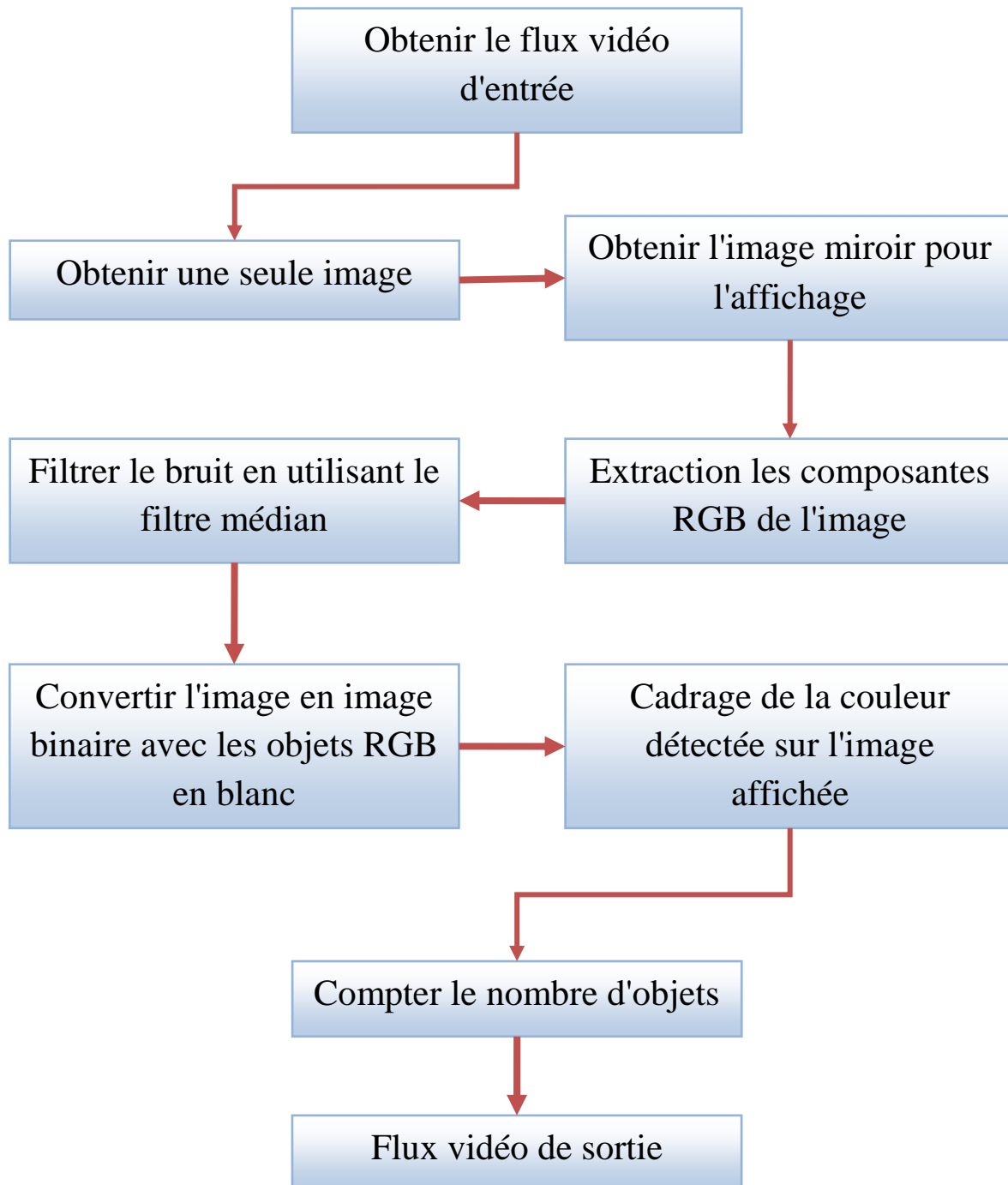
Figure III.3 : l'interface graphique de **guide**

La figure suivante (III.3) représente ma propre conception d'une interface simple pour exécuter un programme qui détecte et suit les objets colorés en rouge, vert, bleu et blanc



Figure III.4 : L'interface graphique de détection et suivi d'objet colorés

L'organigramme suivant (Figure III.5) résume les étapes de travail des algorithmes appliqués dans ce travail

III.5 Schéma résumant les étapes des algorithmes appliqués :**Figure III.5 :** Les étapes de la détection.

III.6 Principes de base de l'utilisation d'une caméra à l'aide de MATLAB :

- ✚ Pour extraire des informations sur l'entrée vidéo dans l'ordinateur via MATLAB, il nous suffit d'utiliser la commande '`info=imaqhwinfo('winvideo')`'.

Concernant notre ordinateur nous obtenons les informations suivantes :

```
info =
  AdaptorDllName: [1x81 char]
  AdaptorDllVersion: '4.5 (R2013a)'
  AdaptorName: 'winvideo'
  DeviceIDs: {[1]}
  DeviceInfo: [1x1 struct]
```

Ceci veut dire qu'on a une seule caméra sur notre ordinateur '`{[1]}`' sous le nom d'adaptateur '`winvideo`'.

- ✚ Et pour en savoir plus sur la caméra connectée à l'ordinateur, nous utilisons la commande : '`info.DeviceInfo(1)`'

Nous obtenons :

```
ans =
  DefaultFormat: 'YUY2_160x120'
  DeviceFileSupported: 0
  DeviceName: 'WebCam SC-03FFL11739P'
  DeviceID: 1
  VideoInputConstructor: 'videoinput('winvideo', 1)'
  VideoDeviceConstructor: 'imaq.VideoDevice('winvideo', 1)'
  SupportedFormats: {1x5 cell}
```

On Remarque que nous obtenons des informations plus détaillées sur la caméra, telles que son format par défaut '`YUY2_160x120`', le nom et le type de l'appareil '`WebCam SC-03FFL11739P`', ainsi que son ID sur MATLAB '`1`' et les formats pris en charge `{1x5 cell}`.

Pour utiliser cette caméra dans le programme, nous devons écrire la commande :

```
'videoinput('winvideo', 1)'
```

- ✚ Nous utilisons la commande '`info.DeviceInfo.SupportedFormats`' pour connaître tous les formats pris en charge sur la caméra.

```
ans =
```

Columns 1 through 4

```
'YUY2_160x120' 'YUY2_176x144' 'YUY2_320x240' 'YUY2_352x288'
```

Column 5

```
'YUY2_640x480'
```

Nous utiliserons cette format ' **YUY2_640x480** ' dans notre travail .

✚ Une fois que nous connaissons toutes ces informations, nous pouvons maintenant utiliser facilement la caméra via MATLAB grâce à ces commandes :

```
vid=videoinput('winvideo', 1)
```

```
preview(vid)
```

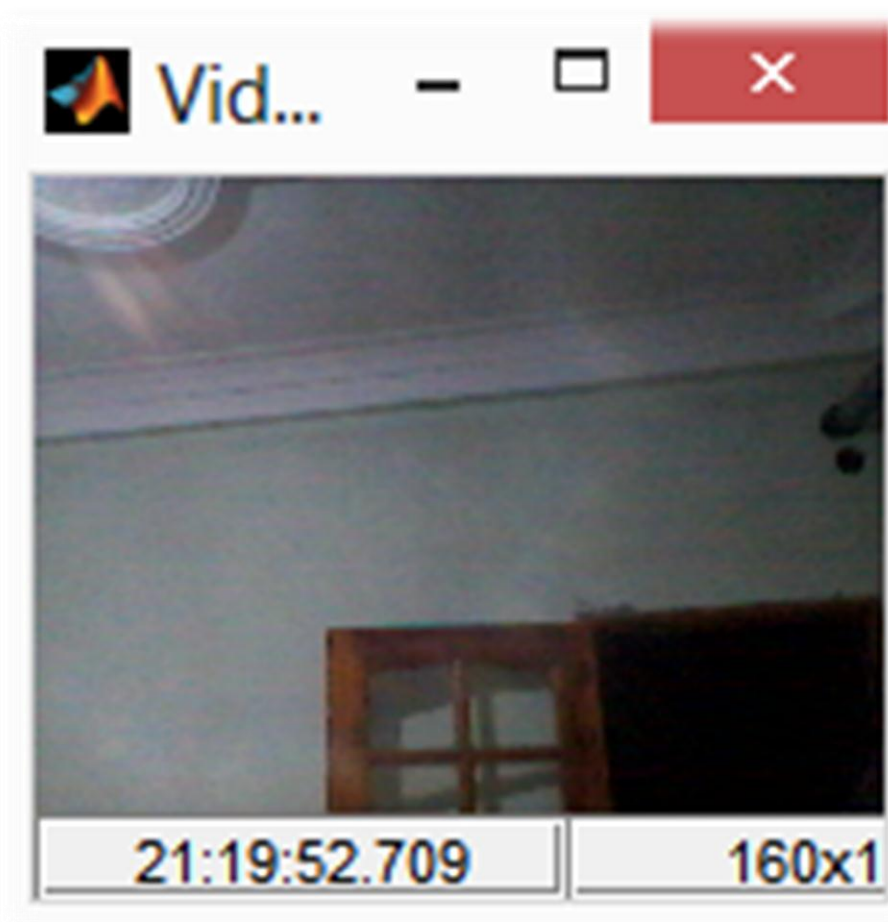


Figure III.6 : L'aperçu de la caméra

Et on peut aussi prendre une photo (cf. Figure II.7) en tapant :

```
s=getsnapshot(vid);
```

```
image(s)
```

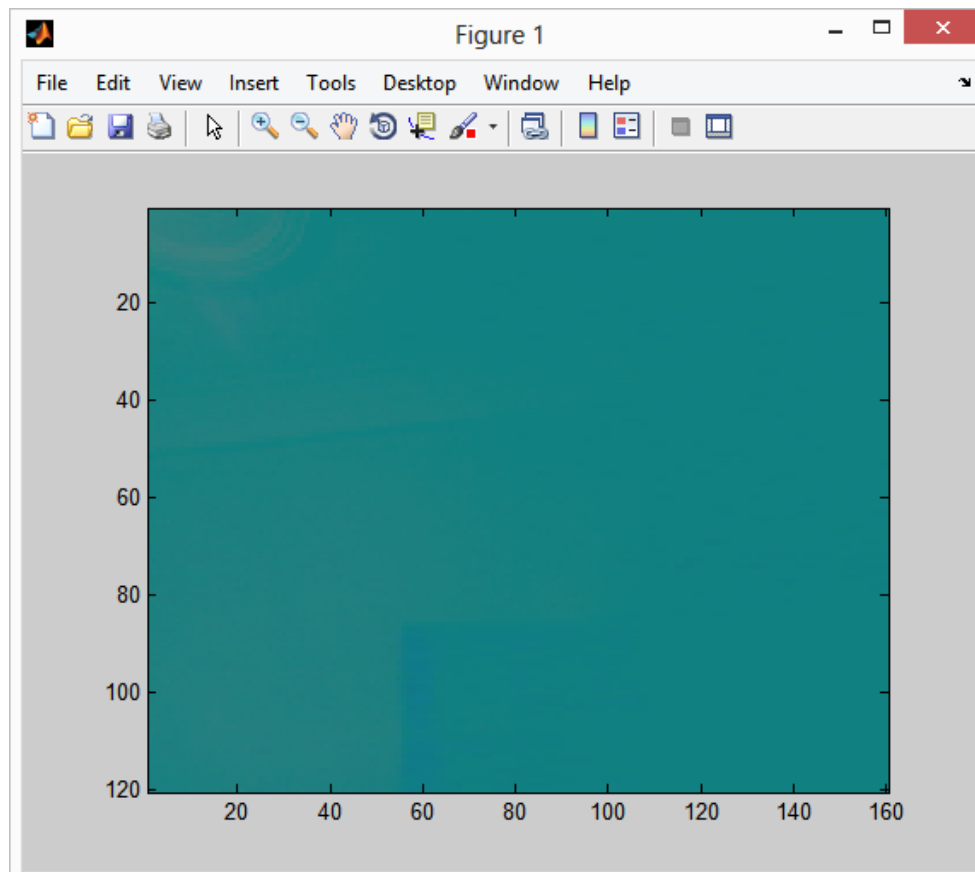


Figure III.7 : la photo capturée (snapshot)

III.7 Implémentation de l'algorithme de détection et de suivi des couleurs RVB-B : [30]

L'algorithme est divisé en deux parties, une partie secondaire (contient : seuil de couleur, fonctionnement de la caméra, etc.) et une partie majeure, qui est la boucle de traitement.

Nous expliquons les commandes les plus importantes dans ces deux parties :

III.8 Première partie :

La première étape :

Déterminer la couleur que nous voulons détecter et suivre en spécifiant le seuil de couleur.

III.8.1 Les seuils de couleurs :

- **Rouge** [1 0 0] = 0.25
- **Vert** [0 1 0] = 0.05
- **Bleu** [0 0 1] = 0.15
- **Blanc** [1 1 1] = 0.75

La deuxième étapes :

Obtention du flux vidéo d'entrée RGB avec des dimensions de 640 x 480

```
vidDevice = imaq.VideoDevice ('winvideo', 1, 'YUY2_640x480',  
...  
'ROI', [1 1 640 480], ...  
'ReturnedColorSpace', 'rgb');
```

Nous passons directement à la partie la plus importante,

III.9 Le cycle de traitement :**Tout d'abord :**

Nous spécifions le nombre d'images à enregistrer (la durée de fonctionnement de la caméra).

Plus le nombre d'images est élevé, plus la caméra fonctionnera longtemps

Par exemple : (**nFrame < 400**)

La deuxième étape :

Nous ne prenons qu'une seule image par la commande

```
rgbFrame = step(vidDevice);
```

Troisième étape :

Nous obtenons une image miroir de l'affichage, cette fonctionnalité nous permet d'afficher les choses telles qu'elles sont en réalité sans permutation dans la direction (de droite à gauche)

```
rgbFrame = flipdim(rgbFrame,2);
```

Quatrième étapes :

Nous obtenons la composante de couleur à détecter à partir de l'image à l'aide de deux commandes, **rgb2gray** et **imsubtract**, où cette dernière calcule la différence entre l'image principale et l'image niveaux de gris (cf. Figures II.8 à II.12)).

```
diffFrame = imsubtract(rgbFrame(:,:,1), rgb2gray(rgbFrame));
```



Figure III.8 : image originale RGB



Figure III.9 : niveau de gris de l'image originale



Figure III.10 : Extraction de la composante rouge



Figure III.11 : Extraction de la composante verte

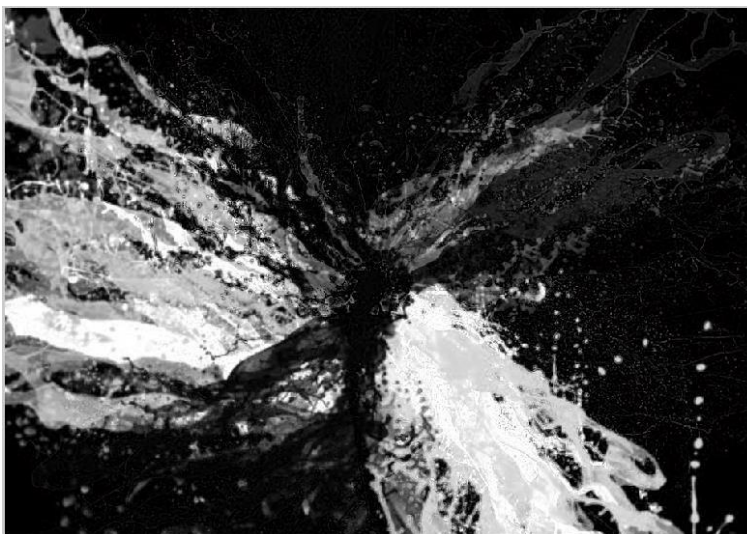


Figure III.12 : Extraction de la composante bleue

Cinquième étape :

Filtrage du bruit à l'aide d'un filtre moyen (filtrage médian 2D)

```
diffFrame = medfilt2(diffFrame, [3 3]);
```

Le filtrage médian est une opération non linéaire souvent utilisée dans le traitement d'image pour réduire le bruit "sel et poivre".

Un filtre médian est plus efficace que la convolution lorsque l'objectif est simultanément de réduire le bruit et de préserver les contours (cf. Figure II.13).

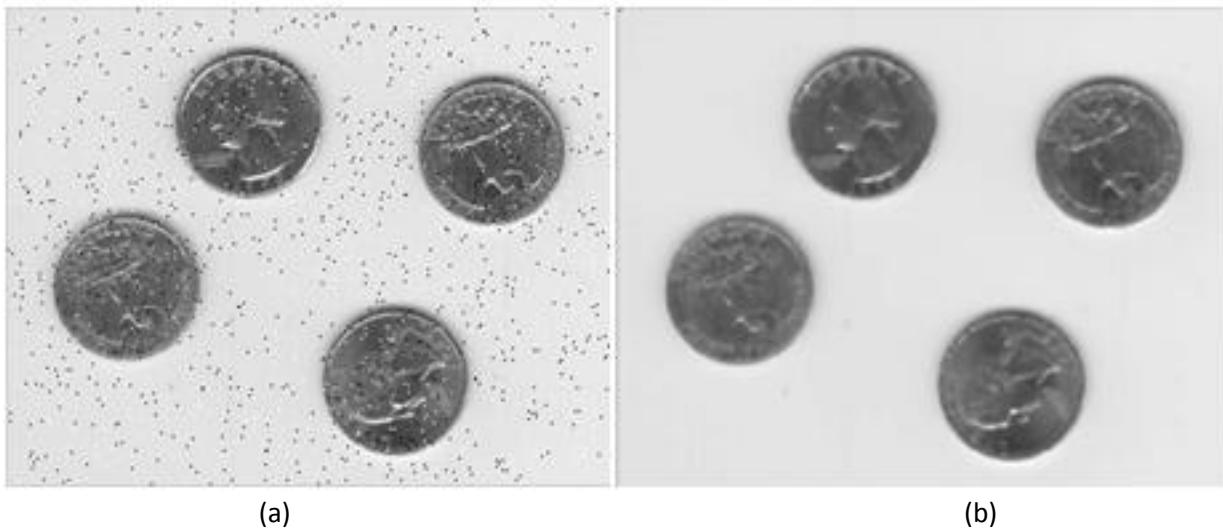


Figure III.13 : image (a) avant le filtrage médian et l'image (b) après le filtrage

La dernière étape :

Convertir l'image en une image binaire avec les objets de la couleur ciblée en blanc, ou ce qu'on appelle le Seuillage (cf. Figure II.14).

```
Par exemple : binFrame = im2bw(diffFrame, redThresh);
```

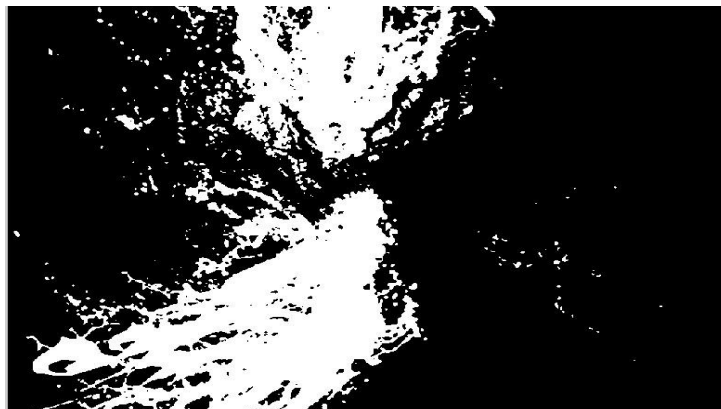


Figure III.14 : Seuillage pour la couleur rouge

III.10 Résultats pour la détection des couleurs :

Dans cette partie, nous présentons les résultats de la détection d'objets colorés (Rouge, Vert, bleu et blanc), que nous avons obtenus à l'aide des algorithmes de détection décrits précédemment, en utilisant l'interface graphique conçue à cet effet.

III.10.1 Détection d'objets rouges :

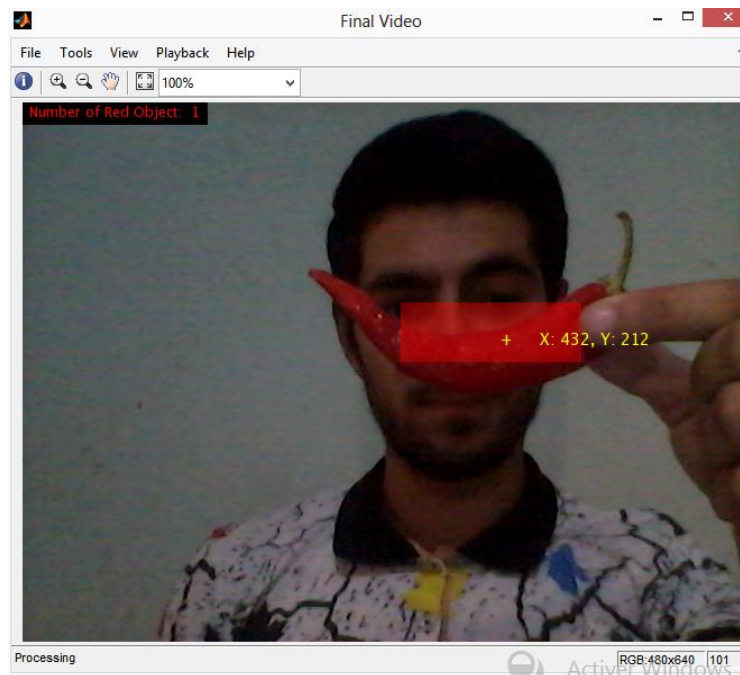


Figure III.15 : Détection d'un seul objet rouge

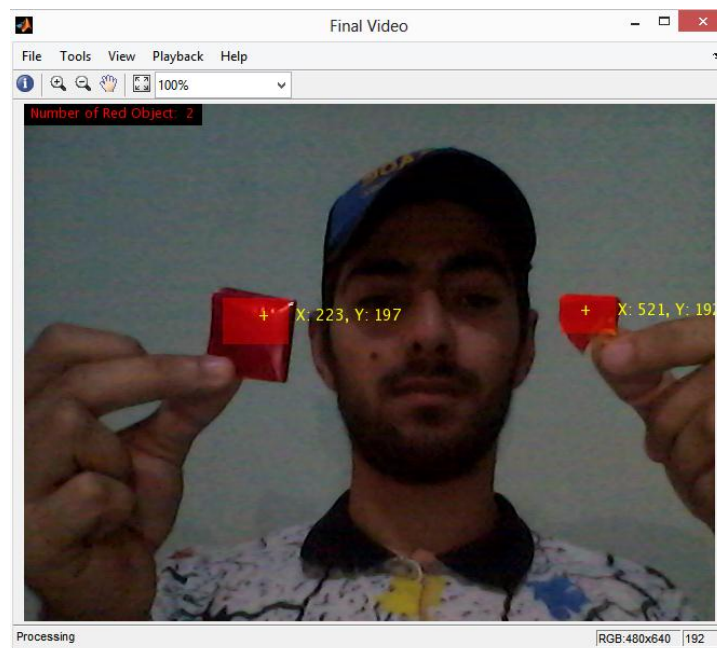


Figure III.16 : Détection de multi objets rouges

III.10.2 Détection d'objets verts :

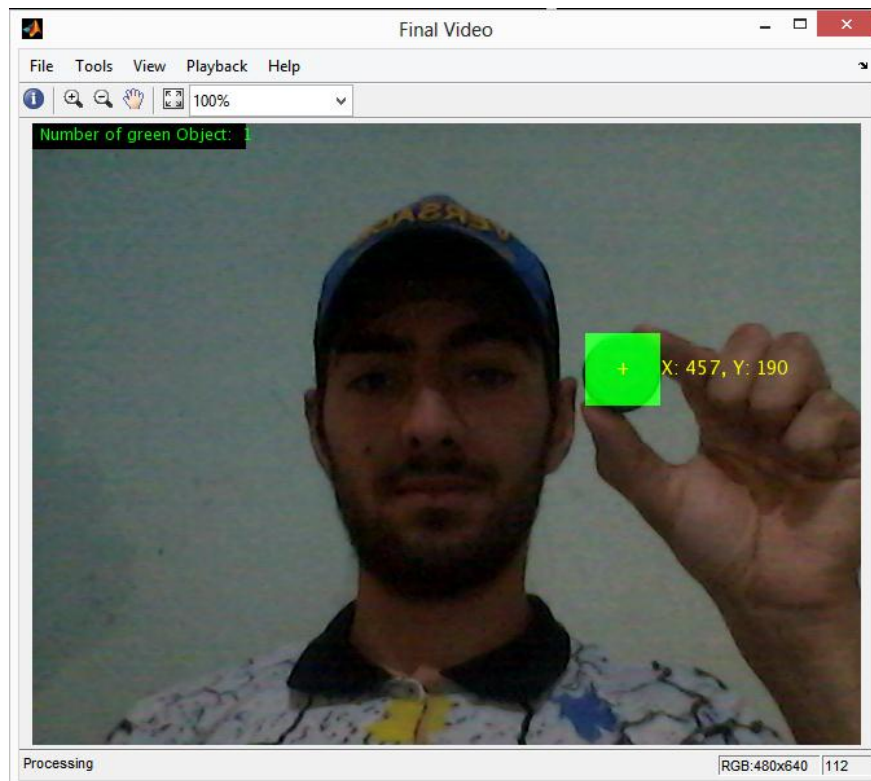


Figure III.17 : Détection d'un seul objet vert

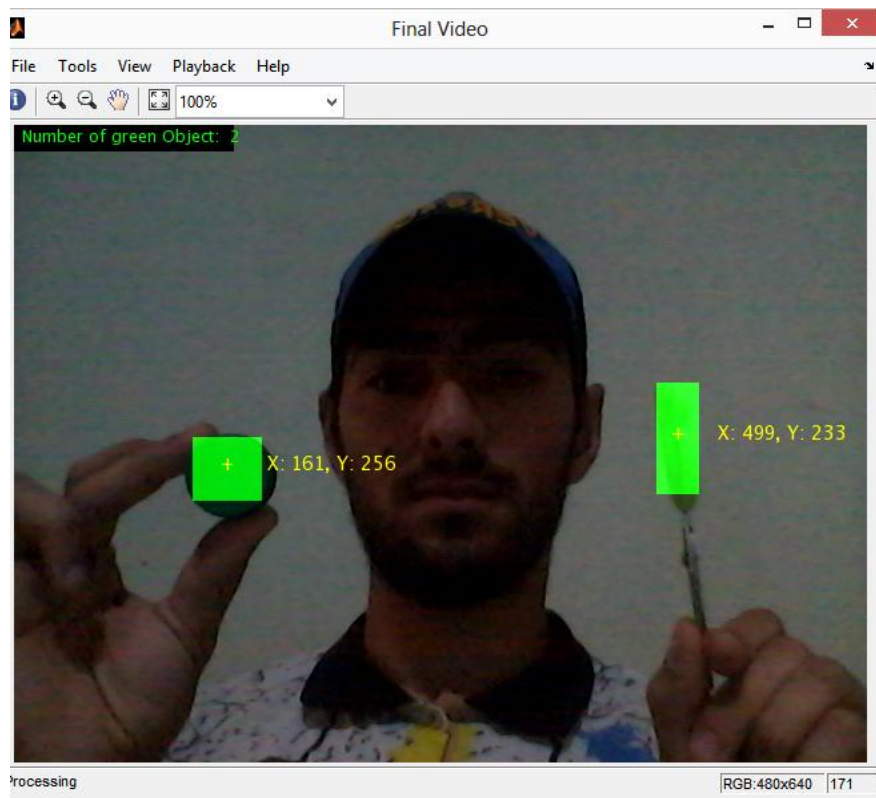


Figure III.18 : Détection de multi objets verts

III.10.3 Détection d'objets bleus :

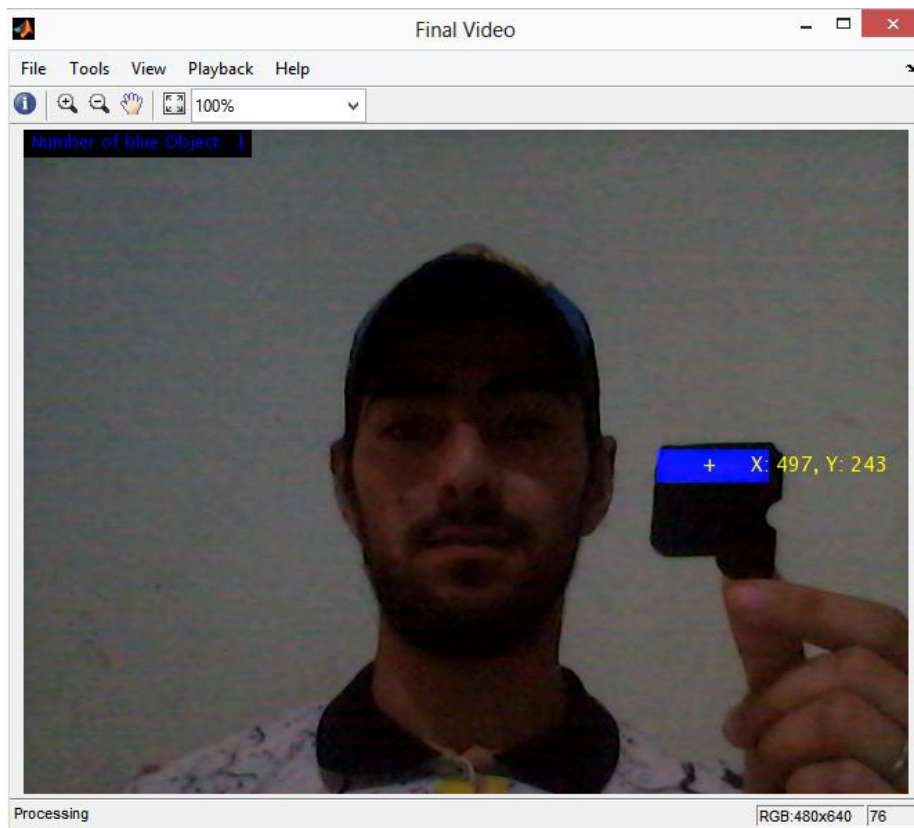


Figure III.19 : Détection d'un seul objet bleu

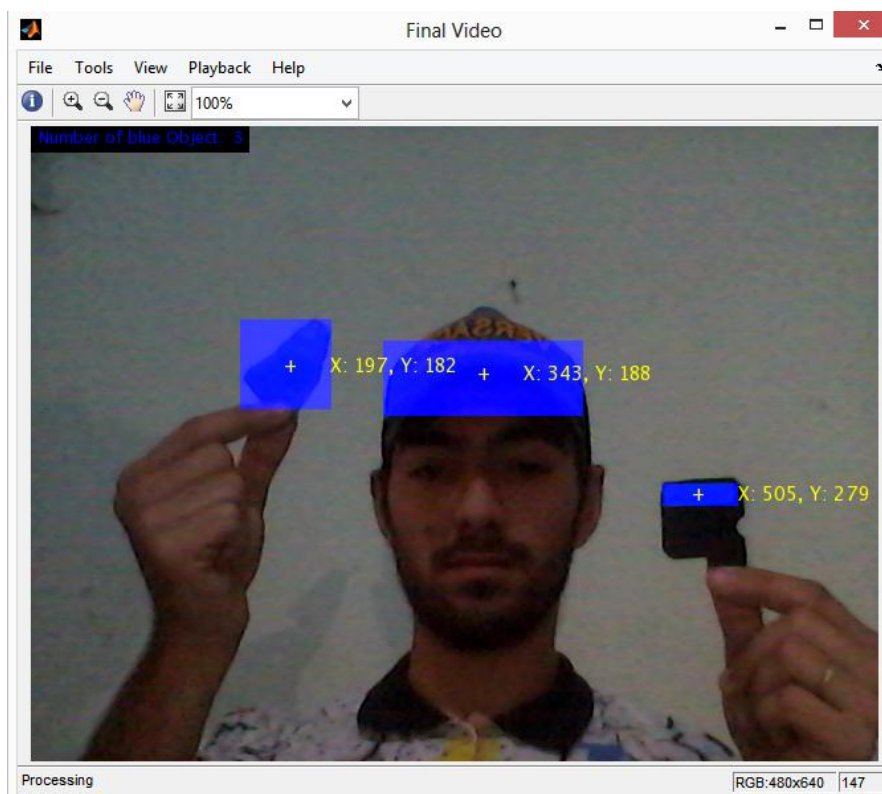


Figure III.20 : Détection de multi objets bleus

III.10.4 Détection d'objets blancs :

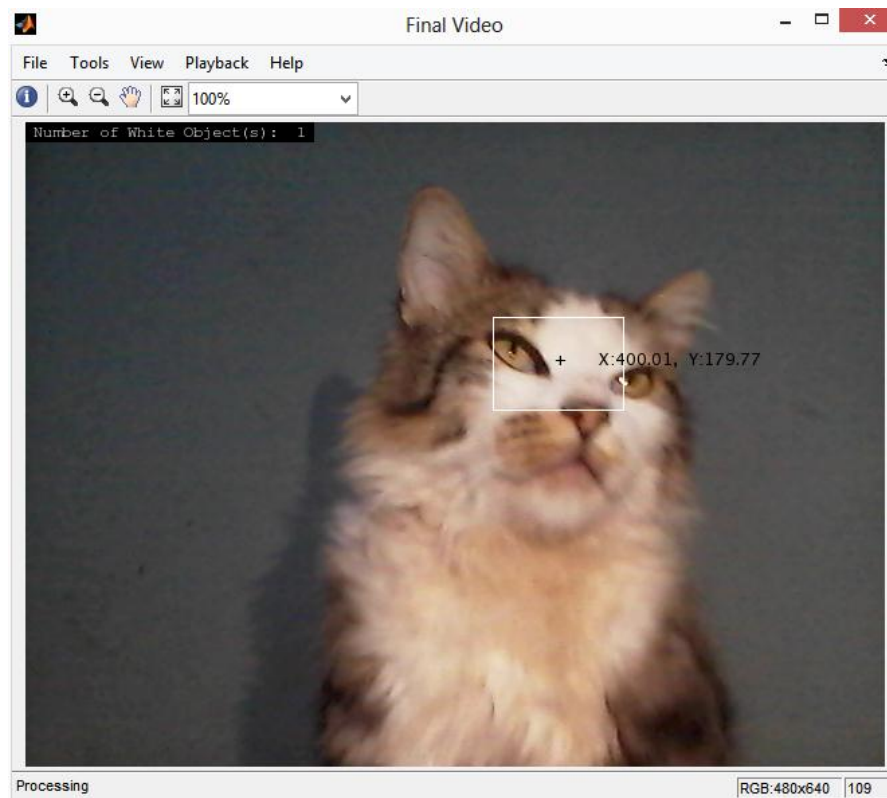


Figure III.21 : Détection d'un seul objet blanc

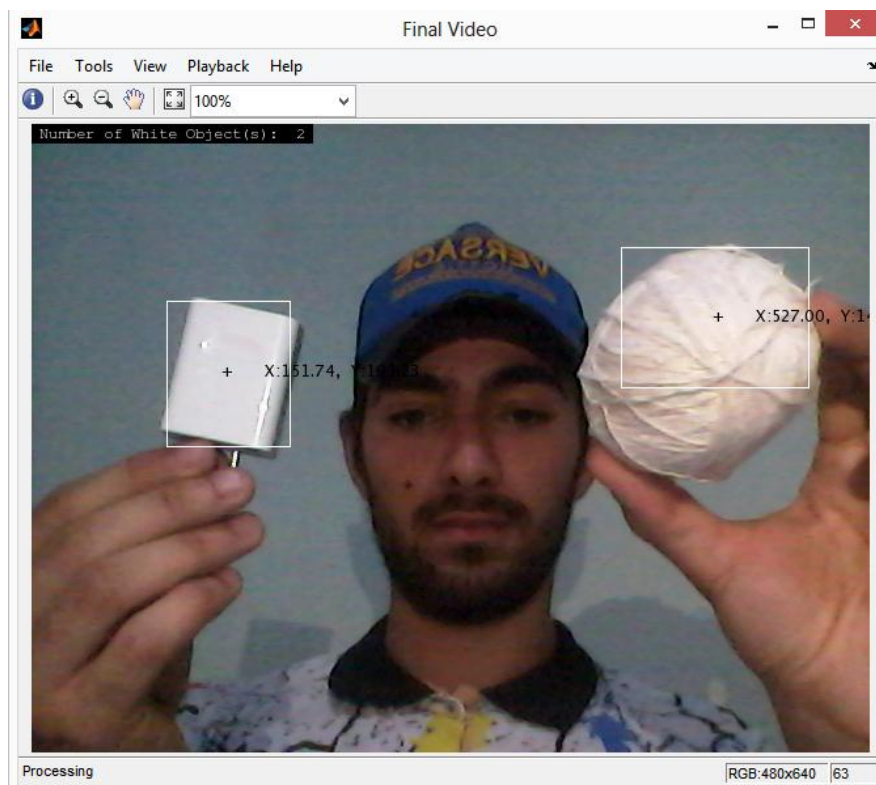
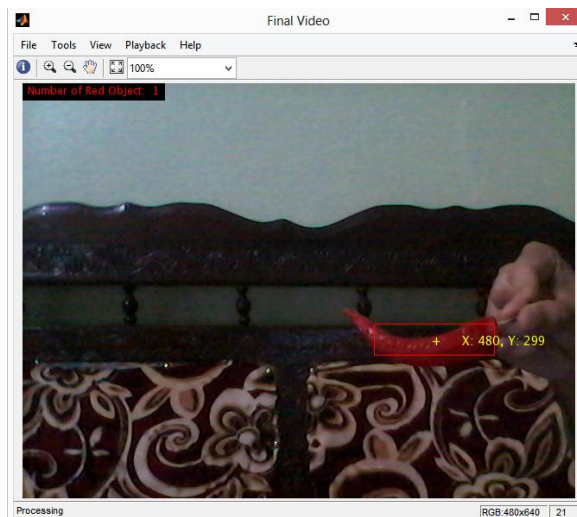


Figure III.22 : Détection de multi objets blancs

III.11 Résultats de suivi : [30]

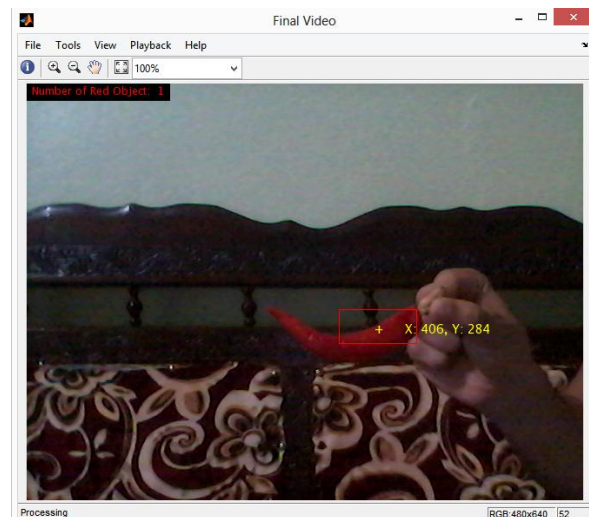
Pour le suivi nous utilisons la fonction `vision.BlobAnalysis` de Matlab est utilisée pour obtenir les centroides des régions colorés ciblées et les rectangles qui les entourent. Cette fonction permet de faire une détection de taches (Blob détection). En vision par ordinateur, les méthodes de détection de taches visent à détecter dans une image numérique des régions dont les propriétés, telles que la luminosité ou la couleur, diffèrent des régions environnantes.

III.11.1 Suivi d'un objet rouge :



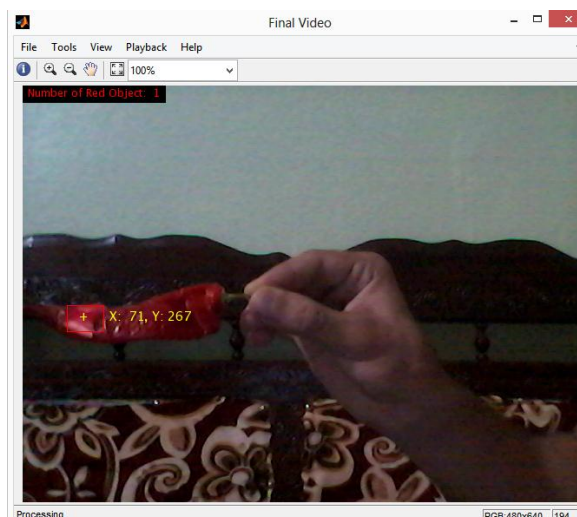
L'instant T1 = 21s

Les coordonnées : X :480 , Y :299



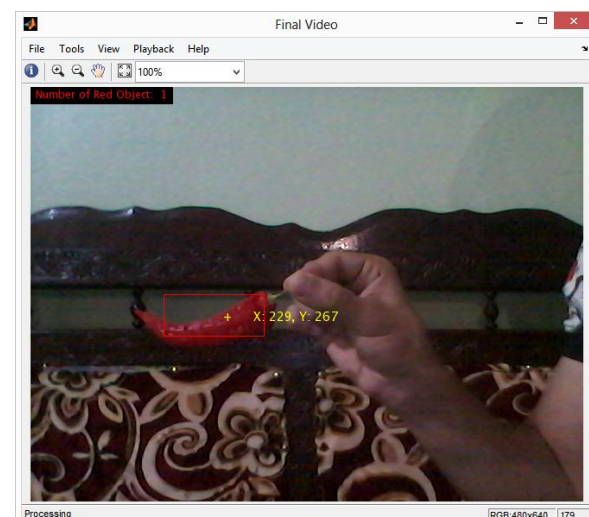
L'instant T2 = 52s

Les coordonnées : X :406 , Y :284



L'instant T4 = 194s

Les coordonnées : X :71 , Y :267

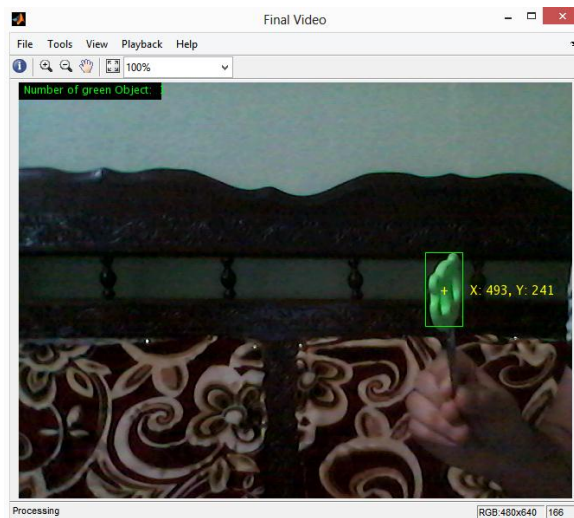


L'instant T3 = 179s

Les coordonnées : X :229 , Y :267

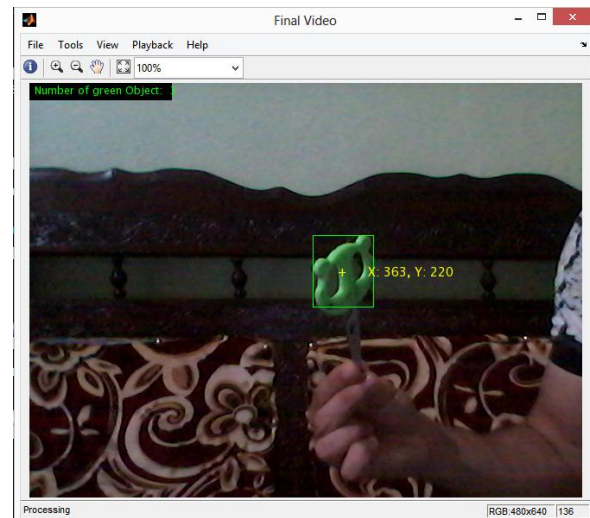
Figure III.23 : Suivi d'un Poivron rouge dans une séquence vidéo

III.11.2 Suivi d'un objet vert :



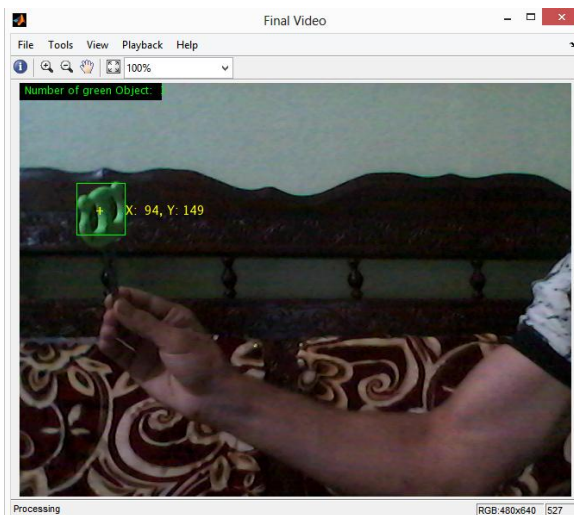
L'instant T1 = 166s

Les coordonnées : X :493 , Y :241



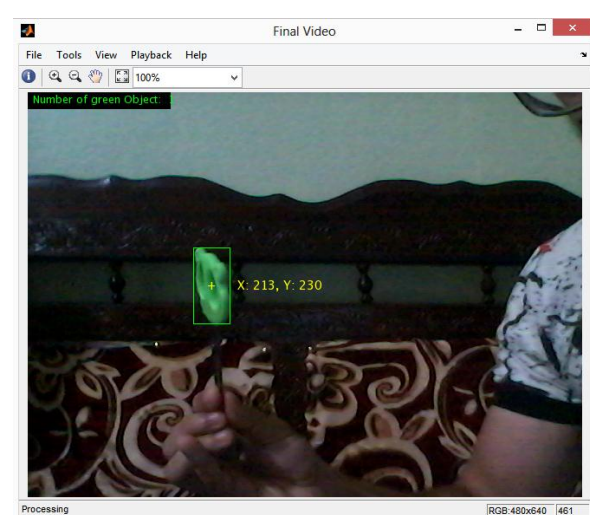
L'instant T2 = 196s

Les coordonnées : X :363 , Y :220



L'instant T4 = 527s

Les coordonnées : X :94 , Y :149

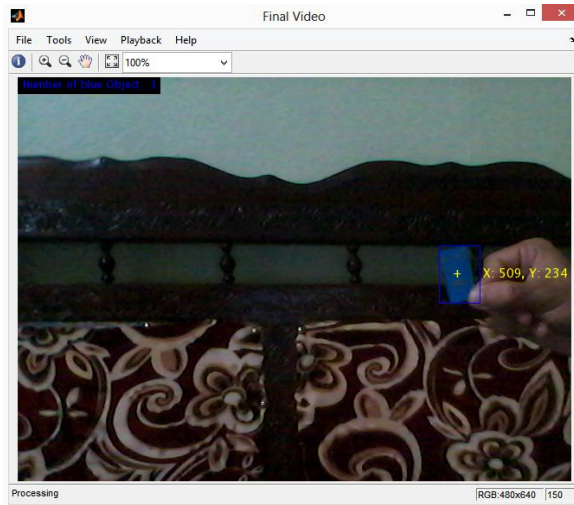


L'instant T3 = 461s

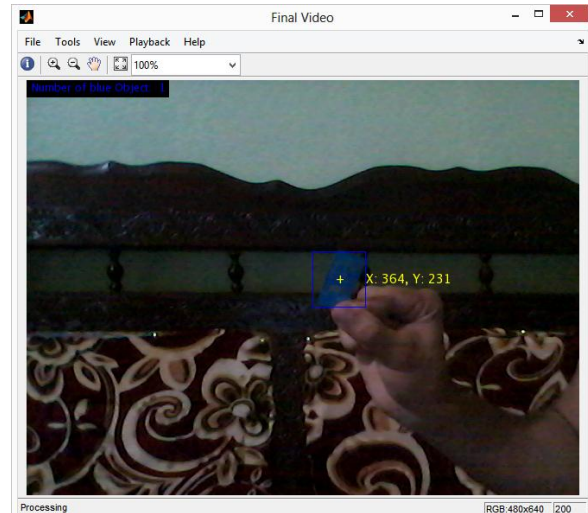
Les coordonnées : X :213 , Y :230

Figure III.24 : Suivi d'un ciseau vert dans une séquence vidéo

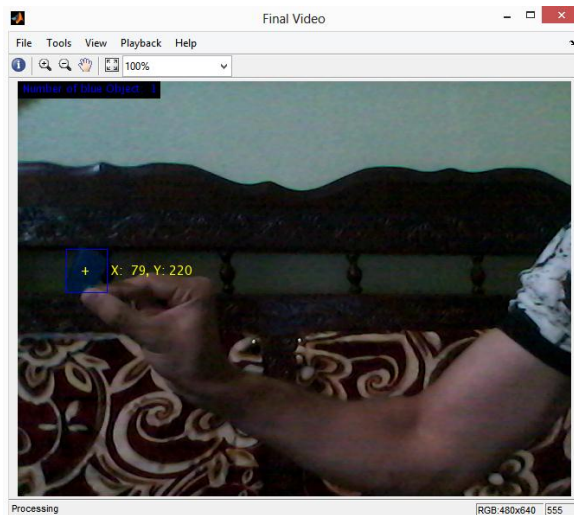
III.11.3 Suivi d'un objet bleu :



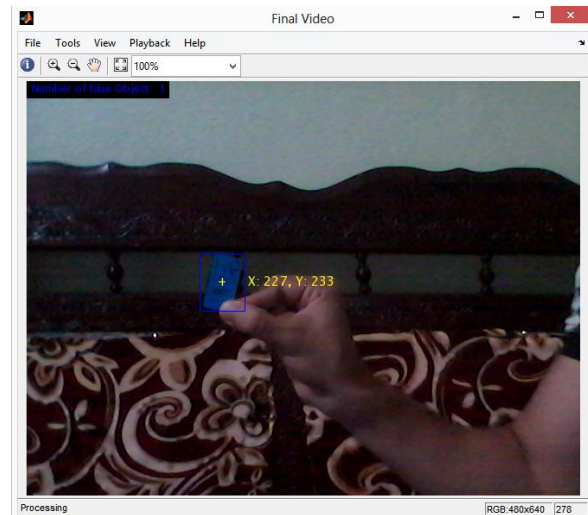
L'instant T1 = 150s
Les coordonnées : X :509 , Y :234



L'instant T2 = 200s
Les coordonnées : X :364 , Y :231



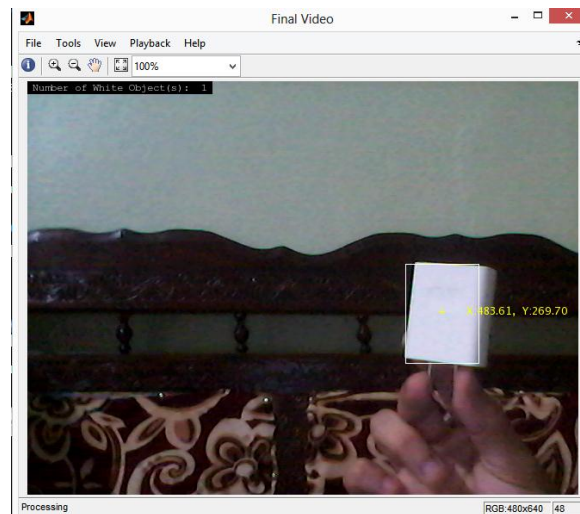
L'instant T4 = 555s
Les coordonnées : X :79 , Y :220



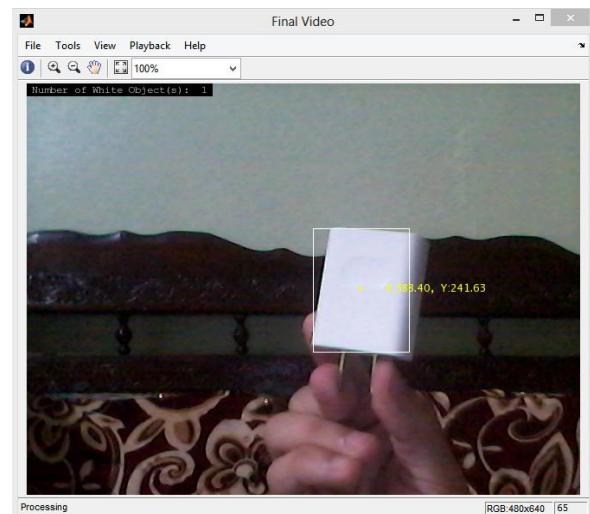
L'instant T3 = 278s
Les coordonnées : X :227 , Y :233

Figure III.25 : Suivi une carte bleue dans une séquence vidéo

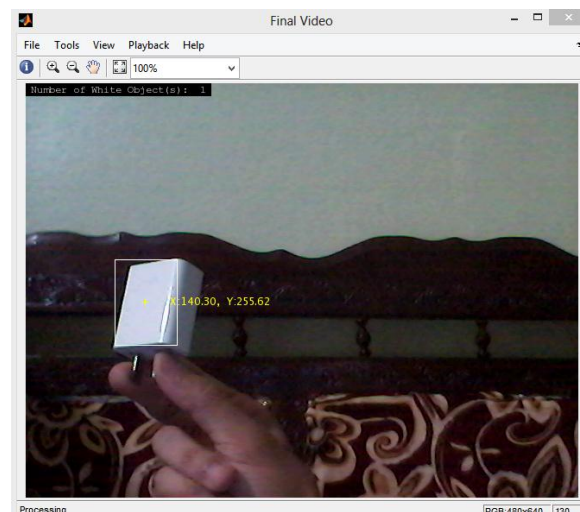
III.11.4 Suivi d'un objet blanc :



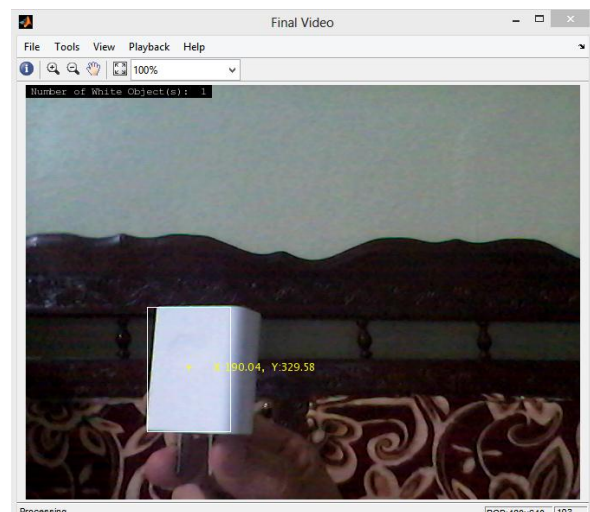
L'instant T1 = 48s
Les coordonnées : X :483 , Y :269



L'instant T2 = 65s
Les coordonnées : X :388 , Y :241



L'instant T4 = 130s
Les coordonnées : X :140 , Y :255

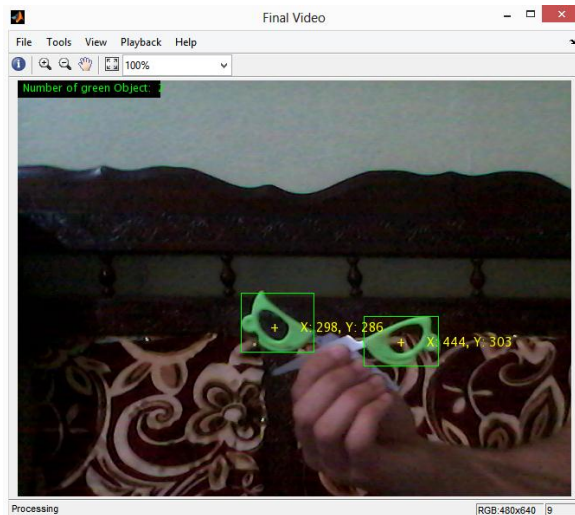


L'instant T3 = 103s
Les coordonnées : X :190 , Y :329

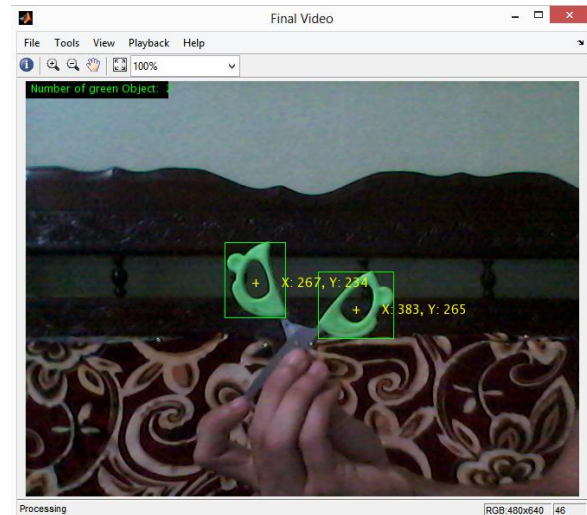
Figure III.26 : Suivi une boîte de chargeur blanche dans une séquence vidéo

III.11.5 Suivi multi objet coloré :

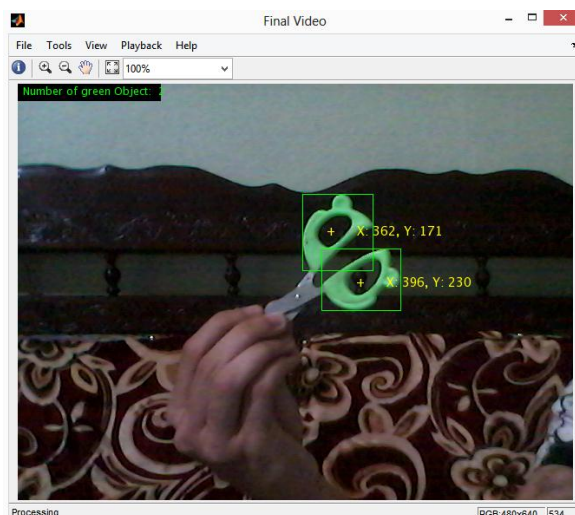
Nous avons choisi la couleur verte pour mener les tests de suivi.



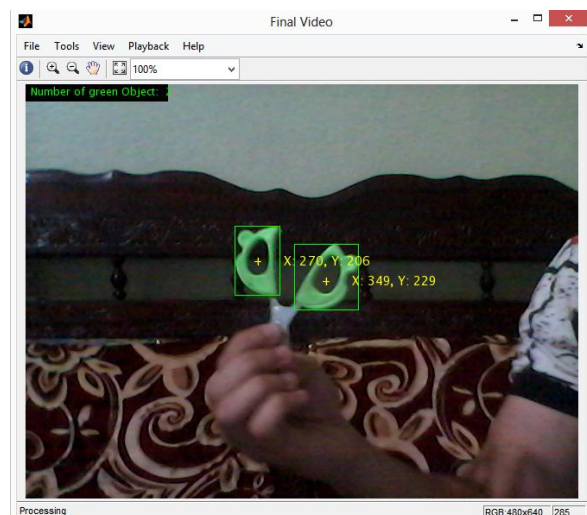
L'instant T1 = 9s
X1 : 444 , Y1 : 303 , X2 :298 , Y2 :286



L'instant T2 = 46s
X1 : 383 , Y1 : 265 , X2 :267 , Y2 :234



L'instant T4 = 534s
X1 : 396 , Y1 : 230 , X2 : 362 , Y2 :171



L'instant T3 = 285s
X1 : 449 , Y1 : 229 , X2 :270 , Y2 :206

Figure III.27 : Suivi multi objets (verts) dans une séquence vidéo

III.12 Méthode utilisée pour le suivi :

Une analyse des taches (Blob Analysis) est appliquée par la fonction **vision.BlobAnalysis** . Cette dernière retourne un objet résultant d'une analyse des taches qui permet de calculer les statistiques des régions connectées dans une image binaire.

```
hblob = vision.BlobAnalysis('AreaOutputPort', false, ...  
    'CentroidOutputPort', true, ...  
    'BoundingBoxOutputPort', true', ...  
    'MinimumBlobArea', 400, ...  
    'MaximumCount', 50);
```

III.13 Les paramètres que nous avons utilisés :

CentroidOutputPort : Retourne les coordonnées du **centroïde** du blob

Bounding BoxOutputPort : Retourne les coordonnées du rectangle conteneur de l'objet

MinimumBlobArea = 400 surface minimale du blob en pixels

MaximumCount = 50 nombre maximal des régions connectées dans chaque image d'entrée

Précision = nombre des vrais positifs / (nombre des vrais positifs + nombre des faux positifs) = 0.571

III.14 Discuter et interpréter les résultats :

Les résultats obtenus dans ce travail prouvent l'efficacité de la méthode appliquée pour détecter les objets colorés dans les couleurs primaires 'rouge, vert, bleu et blanc' séparément, respectivement et ce en temps réel.

Iles tests ont montré une bonne précision en ce qui concerne le suivi, qu'il s'agisse d'un seul objet ou de plusieurs à la fois.

Nous avons choisi les seuils appropriés pour chaque couleur, et malgré ça nous avons parfois remarqué des erreurs de détection, notamment la couleur bleue dues à plusieurs facteurs, à savoir une mauvaise qualité de l'appareil photo, qui provoque un peu de flou, qui conduit à une erreur de calcul de la vraie couleur des pixels, nous avons utilisé un filtre **moyen** pour réduire un peu ce problème.

La qualité de l'éclairage est également un facteur très important pour le bon fonctionnement de cette méthode, car une lumière excessive provoque une réflexion blanche sur les surfaces lisses, tandis qu'un mauvais éclairage crée plus de bruit dans la scène et l'apparition de zones noires.

III.15 Conclusion :

Dans ce chapitre, nous avons fourni une brève explication de certaines commandes et algorithmes sur la base desquels fonctionne l'application de détection et de suivi des objets colorés (RVB.B), qui a prouvé son efficacité dans la plupart des cas, et cela est démontré par les résultats auxquels nous sommes parvenus.

Nous avons également construit et présenté une interface graphique simple pour faciliter l'utilisation sans avoir besoin de passer par les différentes étapes de traitement.

Et au final, on peut considérer les résultats obtenus comme bons dans l'ensemble et comme un témoignage de la qualité de la méthode utilisée pour la détection et le suivi des couleurs.

Conclusion général

Conclusion général

Dans ce mémoire, j'ai essayé d'appliquer les connaissances que j'ai acquises au cours de mon parcours universitaire, et ceci dans le but de résoudre deux problèmes principaux, à savoir la détection et le suivi d'objets dans une scène vidéo en temps réel (caméra fixe, arrière-plan fixe, objet en mouvement).

Dans ce travail, sont abordées les bases de traitements d'images et de vidéos, en plus des différentes méthodes de détection et de suivi d'un ou de plusieurs objets en mouvement.

Toutes les méthodes citées ont montré des avantages et des inconvénients. Certaines sont faciles à comprendre et à utiliser avec des résultats limités, d'autres sont plus performantes mais plus complexes. Chacun des algorithmes étudiés est plus efficace que d'autres algorithmes dans des cas spécifiques en raison de sa simplicité ou de sa précision, c'est-à-dire qu'ils partagent des compromis.

J'ai choisi de travailler sur les objets en fonction de leur couleur. Les algorithmes de reconnaissance des couleurs sont simples et pratiques, mais ils sont limités car ils reconnaissent des objets d'une seule couleur.

J'ai utilisé MATLAB pour programmer les codes de détection et de suivi car il est relativement facile à utiliser et il possède une bibliothèque riche de fonctions puissantes, notamment pour le traitement d'images et de vidéos.

La méthode adoptée permet d'utiliser les couleurs primaires rouge, vert et bleu pour détecter correctement la région désirée et ainsi permettre un suivi précis des séquences en temps réel.

Dans cette étude, nous avons confirmé que la couleur est efficace pour détecter et suivre les objets en mouvement, et cela semble clair à travers les résultats obtenus.

Les points forts de cette méthode résident dans :

- sa simplicité par rapport aux autres méthodes,
- la possibilité de l'appliquer à tout type d'image et de vidéo indépendamment de la nature de l'objet suivi.

Il est également intéressant de mener une étude comparative entre différentes méthodes de détection d'objets en mouvement pour voir leurs performances réelles dans le domaine de la vision par ordinateur.

BIBLIOGRAPHIE

Bibliographie

- [1] M. André, « Introduction aux techniques de traitement d'images », Eyrolles, 1987.
- [2] R.C. GONZALES et P. WINTZ, « Digital Image Processing », Addison Wesley », 1997
- [3] M. HADALLAH, « Codage des images fixes par méthodes des hybride basée sur la QV et les approximations
- [4] M. KUNT, « Traitement numérique des images », vol.2, 1993
- [5] MICROSOFT, « Encyclopédie Encarta », 2005.
- [6] A. d'HARDANCOURT, « Fou du multimédia » Sybex 1995
- [7] SAMAMBA Tony, Reconnaissance des formes comme outil d'aide aux traitements d'image. Cas des empreintes digitales, mémoire ISP/Bukavu, 2005-2006.
- [8] MOISE MWEZE, Compression Des Images, TFC/ISP Bukavu, 2003-2004
- [9] M Sandeli, traitement d'images par des approches bio-inspirées application à la segmentation d'images, université constantine 2. 2014.
- [10] C.Houassine, segmentation d'images par une approche biomimétique hybride. université universite m'hamed bougara- boumerdes. 2012.
- [11] K. Aounallah, les approches de segmentation d'image par coopération régions-contours, 2010.
- [12] M .T.Chikh , amélioration des images par un modèle de réseau de neurones (comparaison avec les filtres de base), université abou-bakr belkaid tlemcen.2011
- [13] Aek . Tahari, Détection des objets en mouvements dans une séquence d'images, Thèse de Magister, ESI (ex INI), Alger, 2002.
- [14] N. Guealia, L. Hadji, Suivi d'objet en mouvement, Projet de fin d'étude, ESI (ex INI), Alger, 2003.
- [15] Seuillage d'une Image avec MATLAB. FPGA | Arduino | Matlab | Cours @ www.electronique-mixte.fr
- [16] MEDJAHED Fatiha, " Détection et Suivi d'Objets en Mouvement Dans Une Séquence d'Images ", Université Mohamed Boudiaf Oran, 2011.
- [17] Ahmed Ben Atitallah, "Etude et Implantation d'Algorithmes de Compression d'Images dans un Environnement Mixte Matériel et Logiciel", Université Bordeaux 1, Juillet 2007.

Bibliographie

- [18] David Salomon, "Data-compression-the-complete-reference ", 2007.
- [19] M. Chaumont, représentation en objets vidéo pour un codage progressif et concurrentiel des séquences d'images, thèse doctorat, université de rennes 1, novembre 2000.
- [20] J.MARIE. La liaison automatique des plusieurs images perçues sur un scanner ISP (Institut Supérieur Pédagogique de Bukavu) - licencié en pédagogie; Option : Informatique de Gestion 2008.P2 .P4.P5.
- [21] H.YEDJOUR. Détection de contours et suivi d'objet dans une séquence d'images par les réseaux de neurones impulsionnels Université Mohamed Boudiaf (USTO) ORAN Thèse Magister 2010.P13,14,16,17,18,19,20,21,22 ;23,P24.
- [22] J.PIATER , P. GABRIEL, J. VERLY and A. GENON. The state of the art in multiple object tracking under occlusion in video sequences. Advanced Concepts for Intelligent Vision Systems, 2003, P 166,173.
- [23] Cao Tien Dung, " La vidéos surveillance ", Institut de la Francophonie pour l'Informatique, Janvier 2007.
- [24] HORAUD, Radu, MONGA, Olivier, "Vision par ordinateur", 1995.
- [25] Centre de recherche informatique de Montréal, "La vidéosurveillance intelligente", Avril 2009.
- [26] Sung Wook Seol, Jee Hye Jang, Hyo Sung Kim, Chul Hun Lee, and Ki Gon Nam "Detection and Tracking System of Moving Objects Using Double Difference based Motion Estimation", 2003.
- [27] R. Collins, A. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N.Enomoto, O. Hasegawa, P. Burt et L. Wixson, "A System for Video Surveillance and Monitoring", Robotics Institute, May 2000.
- [28] H.YEDJOUR. Détection de contours et suivi d'objet dans une séquence d'images par les réseaux de neurones impulsionnels Université Mohamed Boudiaf (USTO) ORAN Thèse Magister 2010.P13,14,P16,17,18,19,20,21,22 ;23,P24.
- [29] A . BUGEAU .Détection et suivi d'objets en mouvement dans des scènes complexes, application a la surveillance des conducteurs, Thèse de Docteur Université Rennes 1, 2007.P23 ,24.
- [30] ARINDAM Bose, white Object Detection and Tracking, 1.05, How to detect and track white objects in Live Video

Bibliographie

LES SITES :

[S1] <http://perso.univ-lemans.fr/~berger/CoursStereoVision/co/Seuillage.html>

[S2] <http://gillesboulet.ca/textes/VideoNumerique.pdf>.

[S3] <http://www.tdechardin.org/web/page.php?id=1350>.

[S4] <http://docslide.fr/download/link/27714afb8a63911930a39ea51e575c0e-pfe-oussama>

[S5] <http://www.vision.uji.es/~montoliu/docs/pfm/AnneClaireMaheo.pdf>

الحمد لله
الذي
بنعمته تتم
الصالحات