

Ms050/13A

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

ECOLE NATIONALE POLYTECHNIQUE

Département d'Électronique



المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

En vue de l'obtention du diplôme de Master 2 en Electronique

Thème :

**Étude des différentes Techniques
d'estimation de mouvement**

Réalisé par :

Zine elbadine DAHMANE

Soutenu devant le jury composé de:

Pr. M.MEHENNI

Pr. R.SADOUN

Pr. R.AKSAS

Mr. L.ABDELOUEL

Mr. M.TAGHI

Président

Examineur

Examineur

promoteur

co-promoteur

Promotion : juin 2013

Remerciements

Au terme de ce projet de fin d'études réalisé à l'Ecole Nationale Polytechnique d'Alger, je remercie les membres du jury qui ont accepté de juger ce travail.

Je tien aussi à exprimer ma profonde gratitude à mon encadrant Mr. L.ABDELOUEL, qui grâce à sa disponibilité son aide et ces rigoureux conseils, ma permis de mener à bien ce projet.

Mes remerciements s'adressent également à Mr M.TAGHI pour son aide et sa disponibilité.

Je remercie également ma familles qui ma soutenus tout au long de mon cursus, mes amis qui nous ont encouragés et à tous ceux qui nous ont aidés, de près ou de loin, à accomplir ce projet.

هذا العمل الماستر يأتي في إطار تنمية تطبيق محسن لتقدير الحركة. في الجزء الأول قدمت مفاهيم أساسية للحركة والبعض منها حول تقديرها. في الجزء الثاني تطرقت إلى مختلف تقنيات تقدير الحركة من أجل إيجاد من تقدم أحسن النتائج. أخيرا عينت المعايير التي تحسن هاته التقنيات. الكلمات الرئيسية: تقدير الحركة, مفاهيم الحركة, شعاع التنقل.

Résumé

Ce travail de master s'inscrit dans le cadre du développement d'une application optimisée d'estimation de mouvement. Dans une première partie, j'ai présenté les notions essentielles du mouvement et quelques une de son estimation. Dans la deuxième partie, j'ai abordé les différentes techniques de l'estimation du mouvement afin de pouvoir trouver celle qui donne les meilleurs résultats. Pour finir, j'ai déterminé les critères par lesquelles on pourrait optimiser ces techniques.

Mots-clefs : estimation de mouvement, notions du mouvement, vecteur de déplacement

Abstract

This work of master lies within the scope of the development of an optimized application to estimate the movement. In a first part, I presented the essential notions of the movement and some of his estimations. In the second part, I studied the various techniques of the estimation of movement in order to be able to find that which gives the best results. To finish, I determined the criteria by which one could optimize these techniques.

Key words: estimation of movement, notions of the movement, array moving.



Projet de Fin d'Études réalisé au Laboratoire des

Dispositifs de Communications et de Conversions Photovoltaïques

École Nationale Polytechnique

10 avenue Hacène BADI – El Harrach

BP182–16200 Alger

Algérie

Tél : (+213) 21 52 53 01/03

Fax : (+213) 21 52 29 73

Web : <http://www.enp.edu.dz/>

Table de Matières

Remerciements	2
Résumé	3
Liste des figures	6
Introduction général	7
Chapitre 1 Notions et généralité sur le mouvement	8
Introduction.....	8
1.1 Mouvement réel, mouvement apparent et mouvement estimé.....	9
1.2 Champ de mouvement réel et champ de mouvement apparent	9
1.3 Champ de mouvement apparent et champ de mouvement estimé.....	10
1.4 Estimation de mouvement	11
1.4.1 Problème d'occultation.....	13
1.4.2 Modèles du champ de vecteurs mouvement.....	14
Chapitre 2 Technique d'estimation de mouvement	17
Introduction.....	17
2.1 Estimation du mouvement	17
2.2 Méthodes différentielles (basées gradient).....	17
2.3 Méthodes fréquentielles	20
2.4 méthodes de mise en correspondance de blocs (Block Matching).....	21
2.4.1 Critères de mise en correspondance	22
2.4.2 Dimension optimale du bloc et de la fenêtre de recherche	23
2.4.3 Algorithmes de parcours du voisinage.....	23
Conclusion générale.....	29
References Bibliographies.....	30

Liste des figures

Fig.1.1 : Illustration des mouvements réel et apparent, dans un système optique de prise de vues	10
Fig.1.2 : Estimation directe et inverse du vecteur déplacement	12
Fig.1.3 : Illustration du concept de fond couvert et découvert	14
Fig.1.4 : Modèle de translation du mouvement	14
Fig.1.5 : Modèle de translation, avec des blocs: (a) disjoints; (b) chevauchants	16
Fig.2.1 : Descente de gradient	19
Fig.2.2 : Mise en correspondance de blocs	21
Fig.2.3 Recherche exhaustive	24
Fig.2.4 : Recherche en 3SS	25
Fig.2.5 : Recherche en 4SS	25
Fig.2.6 : recherche logarithmique à 2 dimensions	27
Fig.2.7 : Recherche orthogonale	27

Introduction général

L'estimation de mouvement est une opération clé pour la compression vidéo. Cette opération contribue pour une grande partie à l'efficacité de la compression en éliminant les redondances temporelles. Cette estimation représente la partie la plus critique dans un codeur vidéo et peut atteindre (60%-70%) en termes de complexité et temps de calcul. D'ailleurs la technique d'estimation n'est pas définie par la norme et est laissée au concepteur de choisir la technique d'estimation qui lui convient au mieux. Une estimation de faible qualité est souvent source de problèmes importants comme par exemple des effets de blocs visuellement gênants. Ainsi, depuis plus de vingt ans, son étude reste un sujet de recherche important. Les applications de l'estimation de mouvement sont surtout la réduction de la redondance temporelle pour la compression et l'analyse de scène. Pour cela, beaucoup de chercheurs ont proposé différentes fonctions dans le but de formaliser un bon critère d'estimation de mouvement.

Chapitre 1 Notions et généralité sur le mouvement

Introduction

La détection et l'estimation du mouvement dans des séquences temporelles d'images bidimensionnelles (2D) et tridimensionnelles (3D) est un des problèmes fondamentaux en traitement et en analyse d'images. Ce domaine de recherche récent est à l'origine de nombreux problèmes ouverts pour lesquels il n'existe aucune solution satisfaisante et générale [1].

L'œil humain perçoit et traite en continu le flux de données contenu dans une séquence temporelle d'images. De nombreuses études sur le mouvement [2] ont mis en évidence l'existence de structures spécialisées dans la détection du mouvement dans les yeux de chats et de grenouilles. Il semble que de telles structures existent aussi chez l'homme [3].

On distingue chez l'homme deux types de vision pour percevoir le mouvement:

- ❖ une vision focalisée, au centre de l'image, qui analyse et estime le mouvement;
- ❖ une vision périphérique, qui détecte seulement le mouvement dans le reste du champ visuel.

La capacité à discerner les objets et en particulier leur mouvement, et à naviguer dans l'espace tridimensionnel 3D est une caractéristique presque universelle dans le monde animal. L'intégration d'un tel système de vision dans une machine ou un robot est un problème non résolu. Même s'il est relativement facile d'assembler un système de vision artificielle, il s'est avéré très difficile d'obtenir une capacité visuelle sur une machine, même avec un degré très limité et une variété très large de détecteurs de mouvement et de systèmes visuels.

On note deux grands groupes de chercheurs en vision. Un premier groupe étudie la vision humaine et animale pour comprendre le fonctionnement du système biologique, ses limitations et sa diversité. Les chercheurs qui composent ce groupe sont des neurophysiologistes, physiciens et psycho-physiciens, qui ont prouvé l'existence de certaines cellules corticales dédiées à la détection et à l'estimation du mouvement [3]. Le deuxième groupe inclut des spécialistes en informatique et des ingénieurs qui étudient la vision artificielle, afin de développer des systèmes de vision capables de reconnaître et poursuivre des objets, estimer leur vitesse et leur trajectoire. Les connaissances et les résultats obtenus par les neurophysiologistes [4], les physiciens et les psycho-physiciens [5] aident à la conception des systèmes de vision artificielle faits par les ingénieurs et les spécialistes en informatique. Dans le même temps, les résultats issus de la vision artificielle offrent un cadre conceptuel et technique pour la modélisation de la vision biologique. Les résultats obtenus en combinant systèmes d'acquisition d'images et systèmes avancés de traitement de l'information en sont un exemple (réseaux neuronaux) [6]. La confrontation entre les concepts développés par la vision artificielle et une meilleure connaissance des mécanismes de la vision biologique sont source de progrès mutuels.

Les sources potentielles d'images perçues ou acquises puis traitées par les systèmes visuels biologiques et artificiels, sont nombreuses et les applications variées. A titre d'exemple nous mentionnons:

- ❖ les images aériennes avec la détection et le suivi des masses nuageuses ou l'étude de l'évolution des cultures agricoles.
- ❖ les scènes de télévision numérique avec la compression d'images utilisant l'information de mouvement.
- ❖ la poursuite de cibles pour les applications militaires ou la régulation du trafic automobile.
- ❖ la biologie avec la cinétique des cellules, ou la médecine avec l'analyse du mouvement du cœur.
- ❖ la robotique avec l'inspection visuelle d'une scène animée.

Ces exemples mettent en évidence la diversité des applications dans lesquelles l'estimation du mouvement à partir de séquences temporelles d'images est d'une importance majeure.

1.1 Mouvement réel, mouvement apparent et mouvement estimé

Les images représentent souvent la projection de scènes réelles 3D. C'est pourquoi le mouvement observé (ou mouvement apparent) dans une séquence temporelle d'images représente le plus souvent la projection du mouvement 3D dans le plan image. Une image peut également représenter une coupe d'un objet 3D. Dans ce cas, le champ de mouvement observé à l'intérieur de cette coupe pourra représenter à la fois un déplacement dans le plan image et également la projection sur ce plan d'un mouvement 3D. Le but de l'estimation de mouvement est d'estimer le champ de mouvement 2D ou 3D à partir d'une séquence temporelle d'images 2D ou 3D dont le contenu varie en fonction du temps. On doit différencier:

- ❖ le mouvement réel;
- ❖ le mouvement apparent (ou observé);
- ❖ le mouvement estimé.

1.2 Champ de mouvement réel et champ de mouvement apparent

Le mouvement (ou déplacement) réel anime la scène réelle, dans l'espace réel 3D. Ce mouvement réel est observé soit par l'œil humain soit par un système de prise de vue. Le second fournit un enregistrement du mouvement observé ou apparent sous la forme d'une séquence temporelle d'images. Souvent, le champ de mouvement réel n'est pas directement mesurable [7] dans une séquence d'images 2D ou 3D si l'on s'appuie seulement sur l'intensité des pixels. Dans ce cas, on observe en fait des changements de la distribution spatiale d'intensité lumineuse. Le mouvement ainsi perçu est appelé champ de "mouvement apparent" ou "flux optique" qui, en général, est différent du champ réel de mouvement.

Le champ de mouvement ou de déplacement apparent représente, en général, la projection (orthographique ou projective) du mouvement réel 3D dans le plan image. C'est pourquoi, le champ de mouvement apparent s'appelle aussi "mouvement projeté" [3]. Il représente une approximation du mouvement réel (figure 1.1).

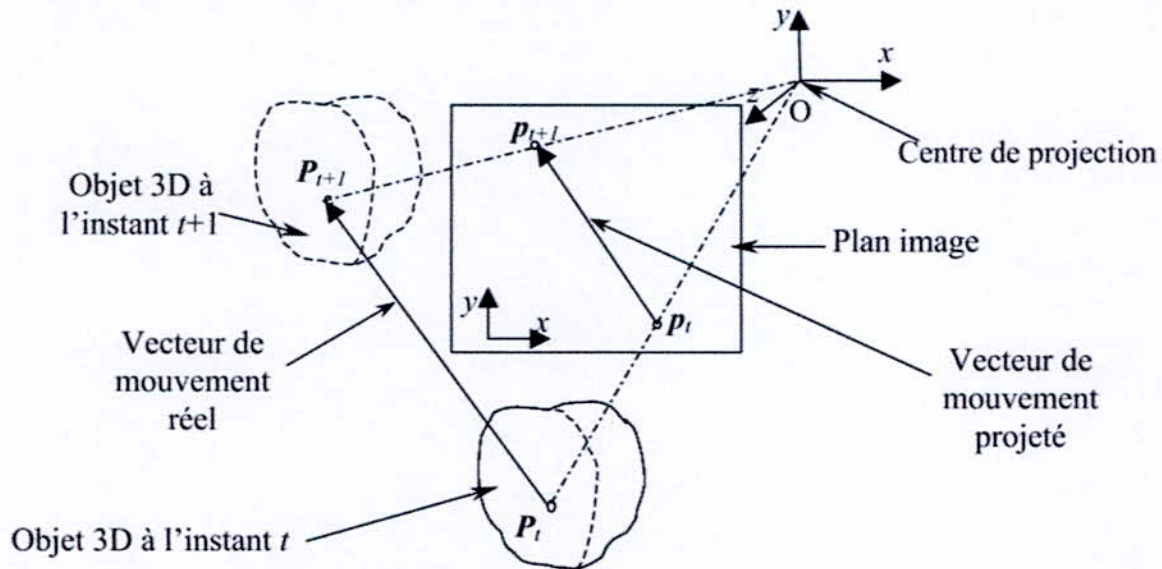


Fig. 1.1 : Illustration des mouvements réel et apparent, dans un système optique de prise de vues [3].

On suppose que le point P_t d'un objet réel 3D, à l'instant t , devient le point P_{t+1} à l'instant $t+1$. La projection perspective des point P_t et P_{t+1} dans le plan image est notée p_t et p_{t+1} , respectivement avec $I_t(x, y)$ et $I_{t+1}(x, y)$ leur intensité. Sur la figure 1.1 on note que le déplacement apparent $\overline{p_t p_{t+1}}$ correspond à la projection perspective du mouvement réel $\overline{P_t P_{t+1}}$.

1.3 Champ de mouvement apparent et champ de mouvement estimé

Le vecteur déplacement estimé $d(p) = (d_x(p), d_y(p))$ correspondant au déplacement du point $p = (x, y)$ du plan image, est déterminé à partir du champ de mouvement apparent, c'est-à-dire des variations locales d'intensité lumineuse $I_t(x, y)$ entre les instants t et $t + \Delta t$, où Δt est la distance en temps inter-images, dans le cas continu. Le vecteur de vitesse estimé $v = (v_x(p), v_y(p))$ est défini comme la variation temporelle du déplacement par unité de temps. $(v_x, v_y) = (d_x/dt, d_y/dt)$. Ceci explique pourquoi une séquence temporelle d'images ne permet que d'estimer le champ de mouvement (déplacement ou vitesse) apparent observable dans la séquence, et non le champ de vitesse réel. On nomme champ de déplacement (respectivement de vitesse), le champ de vecteurs déplacement (respectivement de vitesse) estimé avec une méthode quelconque.

1.4 Estimation de mouvement

Une des hypothèses qui doit être faite pour estimer le mouvement à partir du champ de mouvement apparent est que l'intensité image reste constante au cours du mouvement ou qu'elle varie d'une manière prédictible d'une image à l'autre [8].

L'hypothèse de conservation de l'intensité lumineuse en chaque point le long de la trajectoire du mouvement peut s'exprimer par l'équation DFD des différences entre les images déplacées (en anglais DFD=Displaced Frame Différence), c'est-à-dire entre les images aux instants t et $t + \Delta t$, avec $\Delta t = \pm 1$:

$$DFD = I(x + d_x, y + d_y, t + \Delta t) - I(x, y, t) = 0 \quad (1.1)$$

Où $I(x, y, t)$ est l'intensité du point image $p = (x, y)$ à l'instant t , et $d(p) = (d_x(p), d_y(p))$ est le vecteur déplacement correspondant, entre les instants t et $t + \Delta t$.

On observe que:

- ❖ si les composantes de \mathbf{d} ne sont pas entières, le calcul de la DFD en chaque pixel, nécessite une étape d'interpolation;
- ❖ si les composantes de \mathbf{d} correspondent à la valeur réelle du déplacement, alors l'erreur d'estimation, donc la DFD, est nulle en chaque pixel.

L'estimation du mouvement réel à partir du mouvement apparent peut être abordée de deux manières différentes:

a) l'estimation des vecteurs déplacements dans le plan image:

$$d(x, y, t) = (d_x(x, y, t), d_y(x, y, t)) \quad (1.2)$$

estimés entre les images à t et $t+1$;

b) l'estimation des vectrices vitesses:

$$v(x, y, t) = (v_x(x, y, t), v_y(x, y, t)) \quad (1.3)$$

Le vecteur déplacement (respectivement vitesse) estimés peut varier en espace et en temps.

a) Estimation des vecteurs déplacements: L'estimation peut être vue comme un problème d'estimation de mouvement direct ou inverse ("avant" ou "arrière"), selon que l'estimation est réalisée entre les instants t et $t+1$ ou entre les instants t et $t-1$ (figure 1.2).

Dans le cas de l'estimation directe (avant) du mouvement, le problème se pose de la manière suivante: si on connaît les échantillons spatio-temporels $I_t(x, y)$ et $I_{t+1}(x, y)$, qui sont liés par la relation (hypothèse de conservation de l'intensité):

$$I_t(x, y) = I_{t+1}(x + d_x(x, y, t), y + d_y(x, y, t)) \quad (1.4)$$

On doit trouver le vecteur de déplacement:

$$d_t(x, y) = (d_{xd}(x, y, t), d_{yd}(x, y, t)) \quad (1.5)$$

Dans le cas de l'estimation inverse (arrière) du mouvement où les vecteurs de déplacement sont définis entre les instants t et M , on a la relation:

$$I_t(x, y) = I_{t-1}(x - d_{xi}(x, y, t), y - d_{yi}(x, y, t)) \quad (1.6)$$

En estimation de mouvement on utilise généralement l'estimation inverse (arrière). L'estimation avec compensation directe (avant) du mouvement est classiquement utilisée dans la compression prédictive de séquences d'images. Les valeurs prises par les déplacements d_x, d_y sont souvent réelles, ce qui nécessite une étape d'interpolation pour l'estimation du mouvement.

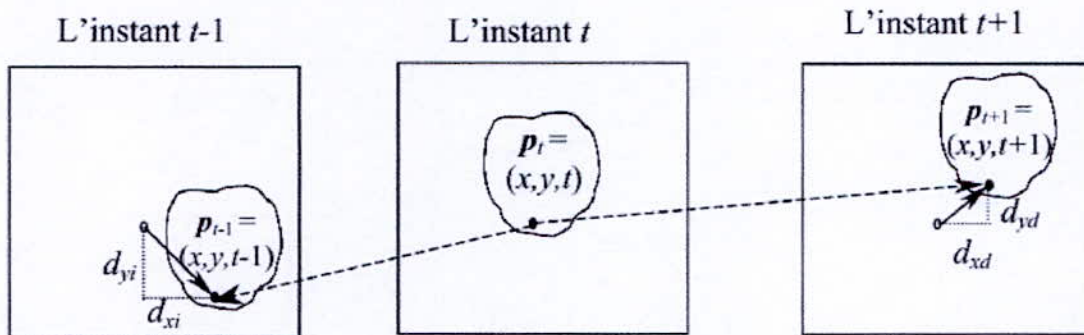


Fig.1.2 : Estimation directe et inverse des vecteurs déplacement [8].

Dans le cas de l'estimation directe on a note: $d_d(x, y) = (d_{xd}(x, y, t), d_{yd}(x, y, t))$

Dans le cas de l'estimation inverse on a note : $d_i(x, y) = (d_{xi}(x, y, t), d_{yi}(x, y, t))$

b) Estimation des vecteurs vitesse: étant donnés les échantillons $I_t(x, y)$, on doit déterminer les vecteurs vitesse $v(x, y, t)$. On observe que si la vitesse reste constante dans l'intervalle Δt entre deux images et si Δt est petit, alors la vitesse estimée peut être assimilée au déplacement:

$$v(x, y, t) = \frac{d_t(x, y)}{\Delta t} \quad (1.7)$$

Pour un mouvement accéléré, on doit prendre en compte plus de deux images afin d'estimer correctement le champ de vitesse.

En conclusion, le champ de mouvement estimé peut être caractérisé par le champ de vecteurs vitesse ou par le champ de vecteurs déplacement (ou de correspondance). Le champ de vitesse estimé et le champ de déplacement sont identiques quand l'échantillonnage temporel de la séquence est constant. L'estimation du mouvement par la perspective de l'estimation du champ de vecteurs vitesse ou par l'estimation de vecteurs déplacement (ou de correspondance) représente donc deux approches équivalentes, quand l'échantillonnage temporel de la séquence est constant ou quand on connaît le pas d'échantillonnage temporel de la séquence. L'estimation du mouvement par la perspective de l'estimation du champ de vecteurs vitesse ou par l'estimation de vecteurs déplacement (ou de correspondance) représente donc deux approches équivalentes, quand on connaît le pas d'échantillonnage temporel de la séquence. Dans la suite on s'intéressera à l'estimation du champ de vecteurs déplacement.

L'estimation de mouvement est très sensible au bruit présent dans les images qui peut être interprété comme étant le résultat d'un mouvement dans la scène réelle.

L'estimation du mouvement est un problème mal-posé qui nécessite l'ajout de contraintes.

Un problème est dit mal-posé s'il n'a pas une solution unique et/ou la solution n'est pas unique:

- ❖ le problème d'existence de la solution peut être illustré dans le cas "d'occultation" ou de mouvement 3D multiples projetés;

1.4.1 Problème d'occultation

L'occultation concerne le recouvrement ou non-recouvrement d'une surface (d'habitude le fond d'une image), dû à la translation ou à la rotation d'un objet dans le champ visuel.

Le concept de fond couvert ou découvert est illustré sur la (figure 1.3) Sur cette dernière, l'objet représenté en pointillé effectue un mouvement de translation dans la direction x , entre les instants t et $t + l$, correspondant aux images I_t et I_{t+1} de la séquence. La région représentée en pointillé dans l'image I_t représente la région du fond qui sera couverte par l'objet dans l'image I_{t+1} . Par conséquent, pour les pixels de cette région il ne sera pas possible de déterminer un correspondant dans l'image I_{t+1} .

La région représentée en pointillés dans l'image I_{t+1} représente la région du fond qui sera découverte par l'objet en mouvement. Par conséquent, les pixels de cette région n'auront aucun correspondant dans l'image I_t .

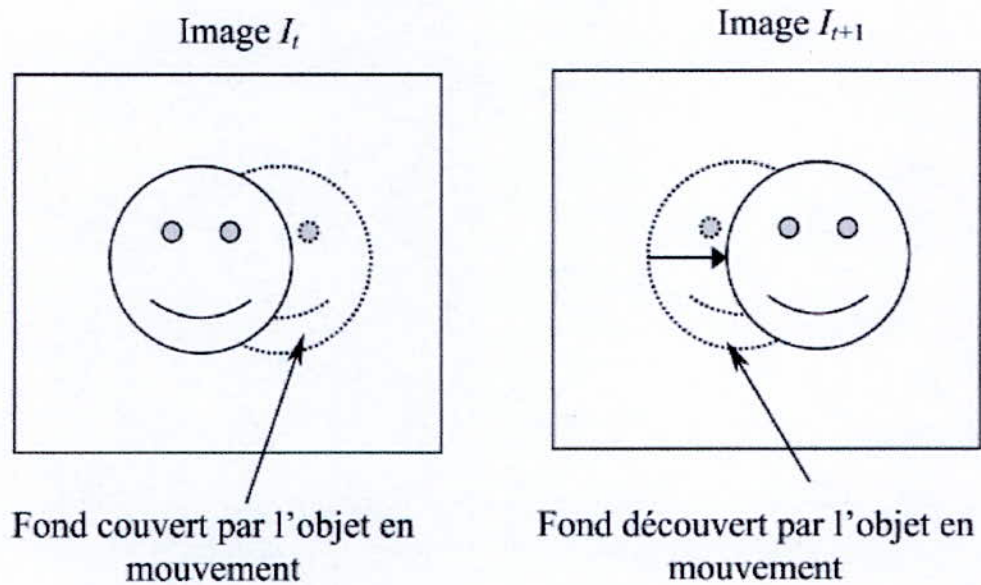


Fig.1.3 : Illustration du concept de fond couvert et découvert [8].

1.4.2 Modèles du champ de vecteurs mouvement

A cause du caractère sous-determiné de l'estimation de mouvement (problème d'ouverture), il est nécessaire d'introduire des modèles des contraintes supplémentaires concernant le champ de mouvement [9] ou directement des modèles (paramétriques ou non paramétriques) de ce champ.

1.4.2.1 Modèle de translation

Le modèle de mouvement le plus utilisé dans les méthodes d'estimation du mouvement est le modèle de mouvement constant ou modèle de translation, qui suppose que tous les pixels du bloc effectuent le même déplacement d (figure 1.4) [10].

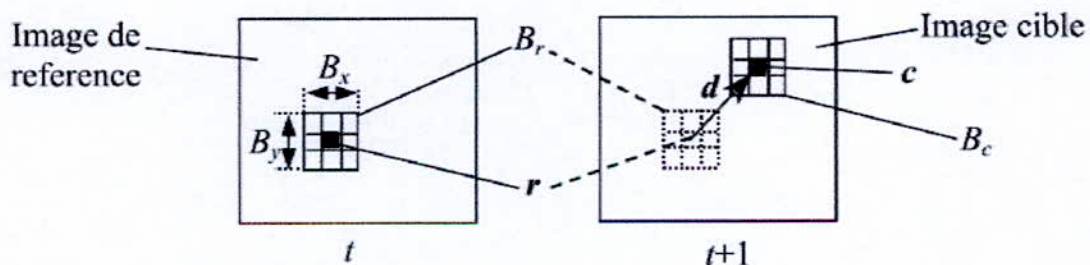


Fig.1.4 : Modèle de translation du mouvement[10].

Ainsi, un bloc B_r de pixels de l'image de référence, de dimensions $B_x * B_y$, centré sur le pixel (r) de coordonnées (x,y) dans l'image de référence à l'instant t , sera mis en correspondance avec le bloc B_c centré en pixel c , dans l'image cible à l'instant $t+1$, ce qui s'écrit:

$$B_r(x, y, t) = B_c(x + d_x, y + d_y, t + 1) \quad (1.8)$$

où $d = (d_x, d_y)$ est le vecteur déplacement. Dans ce modèle, les blocs peuvent être disjoints (le cas le plus fréquent en pratique) ou chevauchants (figure 1.5). Dans le cas du modèle de translation avec des blocs disjoints, on peut attribuer un seul vecteur au bloc entier. Ainsi, la compensation de mouvement peut être réalisée en copiant l'information de gris ou de couleur, pixel par pixel, de l'image de référence dans l'image cible. Dans le cas du modèle de translation avec blocs chevauchants, on doit estimer le vecteur moyen de mouvement dans la région de superposition [10]

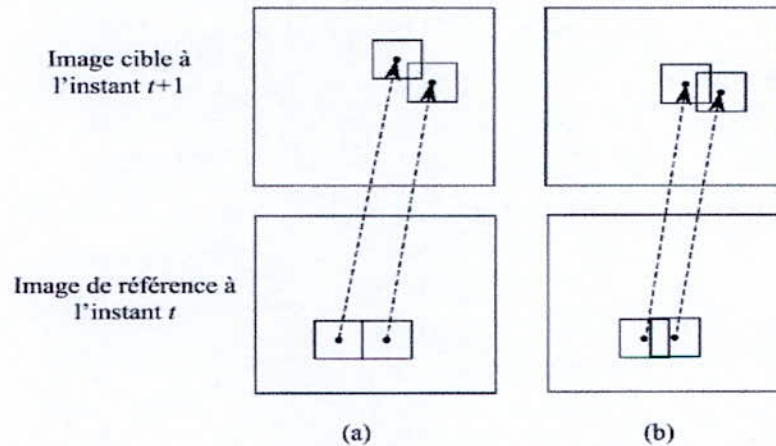


Fig.1.5 : Modèle de translation, avec des blocs: (a) disjoints; (b) chevauchants[10].

Les avantages de l'estimation de mouvement utilisant un modèle de translation du mouvement sont:

- ❖ la compression obtenue en attribuant un seul vecteur de mouvement par bloc;
- ❖ la facilité d'implantation hardware.

Un des inconvénients de l'estimation de mouvement utilisant un modèle de translation du mouvement est constitué par les artefacts de type bloc, dus à un sous-échantillonnage du champ de mouvement.

1.4.2.2 Méthodes non paramétriques

L'avantage des méthodes non paramétriques est qu'elles peuvent être utilisées pour régulariser des mouvements complexes.

Les principales approches non paramétriques sont :

- a) **Les méthodes différentielles.** Les méthodes différentielles d'estimation du mouvement [8] s'appuient sur l'estimation des gradients spatio-temporels de l'intensité en chaque pixel. Dans le cas des images monochromatiques, les méthodes différentielles avec une contrainte de lissage spatio-temporel font l'hypothèse que le vecteur déplacement varie lentement au voisinage du pixel. Dans le cas des images couleurs, cette contrainte peut être imposée séparément pour chaque couleur, ce qui peut contraindre le vecteur déplacement dans 3 directions différentes [11]. En général, une contrainte supplémentaire est nécessaire pour obtenir des résultats satisfaisants. Dans le cas de superposition des objets (occultations), une contrainte globale appliquée à l'image conduit à une estimation inexacte du mouvement.
- b) **Les méthodes de mise en correspondance.** Dans ces méthodes on suppose que l'image est divisée en plusieurs régions de mouvement. Il existe des méthodes de mise en correspondance dans le plan image [9] ou dans le plan transformé.
- c) **Les méthodes récursives.** Les méthodes récursives sont basées sur la correction d'une prédiction (ou d'un estimé) du vecteur déplacement. Ainsi, l'estimée ou la prédiction peut être considérée comme étant le vecteur déplacement d'un pixel voisin ou une combinaison linéaire des déplacements des pixels dans un voisinage du pixel courant. La correction de la prédiction est faite en minimisant le gradient de l'image de différence entre les images déplacées DFD. Le pas de la prédiction est considéré, en général, comme une contrainte de lissage. Une extension de cette méthode dans le cas des méthodes de mise en correspondance de blocs, a comme résultat les méthodes d'estimation de type Wiener [1].

Chapitre 2 Technique d'estimation de mouvement

Introduction

Dans Ce chapitre je présente les techniques d'estimation de mouvement, qui cherchent à évaluer les mouvements entre deux images. L'information de mouvement peut être utilisée pour la conversion de standard, le dés-entrelacement, la vision informatique, ou encore la compression vidéo de bloc. Je commence par décrire brièvement l'ensemble de méthodes, dites classiques, d'estimation du mouvement. Ainsi que les différents algorithmes utilisés dans l'estimation de mouvement.

2.1 Estimation du mouvement

L'estimation du mouvement dans des séquences temporelles d'images bidimensionnelles est un des problèmes fondamentaux en traitement d'images. Les applications sont nombreuses et parmi les domaines d'application concernés nous pouvons mentionner : la compression d'images utilisant l'information du mouvement, la robotique, la météorologie avec le suivi de masses nuageuses, la médecine avec l'estimation du mouvement d'organes mobiles (comme le cœur ou les poumons), etc.

Les images représentent très souvent la projection de scènes réelles 3D. Pour cette raison, nous pouvons identifier trois types de mouvement : le mouvement réel, le mouvement apparent et le mouvement estimé. Le mouvement apparent, observé à partir des changements de la distribution spatiale d'intensité lumineuse, est aussi appelé le flux optique. Ce mouvement apparent est très souvent différent du mouvement réel et représente en général la projection du mouvement réel 3D dans le plan image.

Afin d'obtenir le mouvement estimé, plusieurs types de méthodes existent. Nous les avons classées dans ce mémoire en trois catégories : méthodes différentielles, méthodes de mise en correspondance de blocs et méthodes statistiques.

2.2 Méthodes différentielles (basées gradient)

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

Soient $x = (x_1, x_2)$ un point de l'image, $I_n(x)$ son intensité lumineuse pour l'image n et d le déplacement du point x entre les images $n-1$ et n . La différence inter-image, ou DFD (Displaced Frame Difference) est définie par :

$$DFD(x, d) = I_{n-1}(x + d) - I_n(x) \quad (2.1)$$

En théorie, DFD atteint la valeur 0 lorsque d est égal au déplacement relatif pratique, d prend des valeurs fractionnaires, nécessitant une interpolation et générant des erreurs. De plus il existe du bruit dans l'image, des variations d'illumination et des occlusions. Le but est donc de trouver le déplacement d qui minimise la DFD. Les techniques d'estimation de mouvement basées gradient s'appuient sur le réarrangement de l'équation (2.1) afin d'avoir accès au déplacement d itérativement, grâce à un développement limite en série de Taylor au premier ordre en d_i (vecteur déplacement à l'itération i) :

$$I_{n-1}(x + d_{i+1}) = I_{n-1}(x + d_i) + (d_{i+1} - d_i)^T \nabla I_{n-1}(x + d_i) + e_{n-1}(x) \quad (2.2)$$

Où ∇ est l'opérateur gradient bidimensionnel, T est l'opérateur transposé, et $e_{n-1}(x)$ représente les termes de plus hauts ordres négligés. On obtient donc d'après

(2.1) et (2.2) :

$$DFD(x, d_{i+1}) = DFD(x, d_i) + (d_{i+1} - d_i)^T \nabla I_{n-1}(x + d_i) \quad (2.3)$$

Comme il n'existe en général pas de solution analytique, on a recouru à des méthodes récursives [12, 13, 14, 15] où le déplacement est estimé itérativement grâce à la méthode de la plus grande pente [15]. Le principe est de considérer que l'opposé du gradient d'une fonction pointe dans la direction où la fonction décroît le plus rapidement (figure 2.1). On peut donc écrire :

$$d_{i+1} = d_i - \epsilon \cdot DFD(x, d_i) \cdot \nabla_{d_i} DFD(x, d_i) = d_i - \epsilon \cdot DFD(x, d_i) \cdot \nabla I_{n-1}(x - d_i) \quad (2.4)$$

Ou plus simplement :

$$d_{i+1} = d_i - \epsilon \cdot \text{sign}(DFD(x, d_i)) \cdot \text{sign}(\nabla I_{n-1}(x - d_i)) \quad (2.5)$$

Où ϵ est un terme de mise à jour. Le choix de ϵ requiert un compromis : si une grande valeur est choisie, la convergence est rapide mais oscille. A l'inverse, si est faible, la convergence est plus précise mais aussi plus lente. Ce terme peut être constant [15] ou adaptatif [13], [16].

Les algorithmes utilisés dans [12], [14] sont basés sur la technique de Wiener. Le mouvement d est calculé itérativement, en utilisant une approche de pseudo inverse de matrice. Cette méthode obtient les meilleurs résultats au prix de calculs coûteux.

Les méthodes pel-récurrentes basées sur le gradient conduisent à un champ de vecteur dense. C'est à dire qu'un vecteur est calculé par pixel. Elles sont donc très sensibles au bruit puisque la DFD n'est calculée que sur un pixel. Afin d'augmenter la robustesse, plusieurs pixels voisins peuvent être pris en compte. Comme les méthodes sont basées sur des développements limités, ces méthodes ne sont pas adaptées à l'estimation de grands déplacements. Une approche hiérarchique peut améliorer les résultats grâce à une représentation multi-résolution des images.

Le champ de vecteur résultant est adapté à des traitements pixel par pixel, tel que du filtrage ou de la conversion de standard.

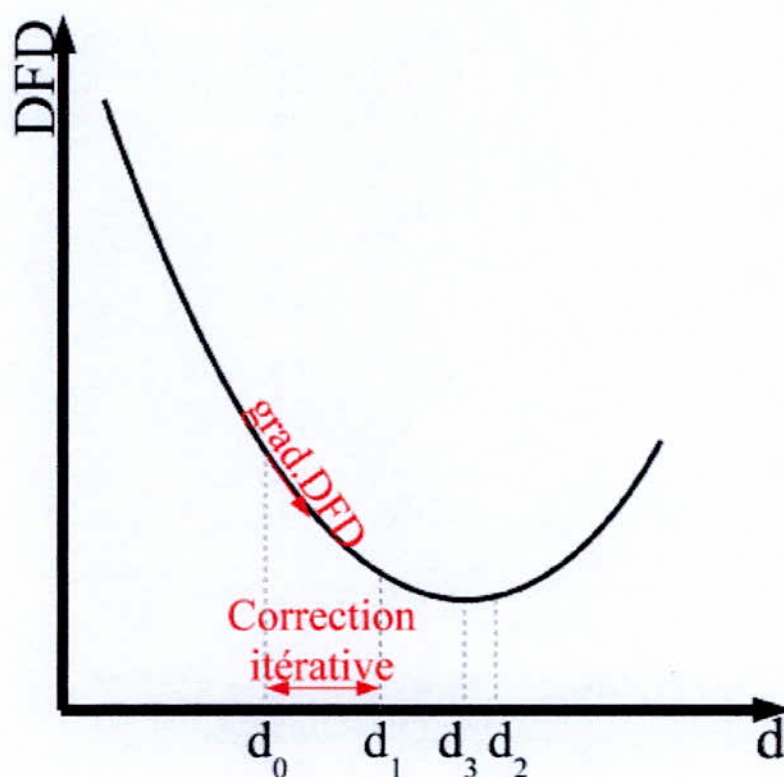


Fig.2.1 : Descente de gradient[14]

2.3 Méthodes fréquentielles

Ces méthodes d'estimation de mouvement sont basées sur des transformations qui permettent de prendre en compte les propriétés fréquentielles du mouvement à savoir qu'une translation dans le domaine de l'image se caractérise par un déphasage dans le domaine fréquentiel [17, 18, 19]. Ces méthodes basées sur une "mesure" du mouvement sont robustes au bruit et donnent le mouvement réel.

Considérons une image en espace et en temps $I(x, y, t)$ et sa transformée de Fourier $\hat{I}(fx, fy, ft)$. Soit r_x et r_y respectivement les mouvements horizontaux et verticaux. La transformée de Fourier de l'image en mouvement est :

$$TF(I(x - r_x t, y - r_y t, t)) = \hat{I}(u, v, w + r_x u + r_y v) \quad (2.6)$$

Les fréquences spatiales sont inchangées, mais toutes les fréquences temporelles sont translatées par le produit de la vitesse et des fréquences spatiales. Il s'agit d'identifier dans l'espace des fréquences un plan de vitesse d'équation, afin de retrouver les composantes r_x et r_y .

$$w + r_x u + r_y v = 0 \quad (2.7)$$

L'information de mouvement est en général extraite par des filtres orientés en espace et en temps (du type Gabor 3D ou filtre large-bande). La contrainte imposée pour éviter le problème d'ouverture est spatio-temporelle : le mouvement à déterminer est supposé constant à la fois sur le support spatial et temporel des filtres orientés. Le résultat obtenu est lissé en espace et en temps, ce qui peut poser des problèmes pour des séquences d'images où les actions sont rapides et saccadées.

On distingue deux approches : les méthodes basées sur l'énergie [20, 21, 22] et les méthodes exploitant la phase du signal [23]. Ces dernières permettent d'obtenir de bons résultats, la phase étant peu sensible aux variations d'illumination.

La taille de la transformée contraint le déplacement maximal détectable. Les avantages de cette méthode est qu'elle est peu sensible au bruit, aux variations d'intensité lumineuse (éclair, explosion) et que les vecteurs résultants sont précis (précision subpixel). Cette technique est intéressante pour du recalage d'image par exemple, mais implique beaucoup de calculs.

2.4 méthodes de mise en correspondance de blocs (Block Matching)

Les méthodes de mise en correspondance de blocs sont les plus utilisées, grâce à leur simplicité d'implantation software et hardware [24], [25], [26]. Ce type de méthode s'est imposé dans les standards de compression d'images, comme H.261 ou MPEG-1, MPEG-2 et MPEG-4.

Les méthodes de mise en correspondance par blocs sont basées sur l'hypothèse selon laquelle l'intensité lumineuse des pixels est constante ou faiblement variable le long des trajectoires de mouvement. Pour chaque point de l'image courante à l'instant « t » on cherche le point d'intensité la plus proche dans l'image référence (instant t-1). Pour éviter les correspondances incohérentes dans l'image référence, on limite la recherche à une fenêtre (ou une zone) de recherche $\pm P$. Ainsi, le déplacement d'un pixel de coordonnées (x, y) , du bloc B de dimension $N * M$, de l'image courante, est déterminé par la position du bloc de même dimension, le plus proche au sens d'un critère de similarité, dans l'image référence (instants t-1), figure 2.2. La recherche est limitée à une fenêtre de dimensions [27]: $F = (2 \cdot P_{x_max} + N) * (2 \cdot P_{y_max} + M)$, où P_{x_max} (resp. P_{y_max}) est la valeur maximale du déplacement à estimer dans la direction x (resp. y), dans notre cas je prend $P_{x_max} = P_{y_max} = P$.

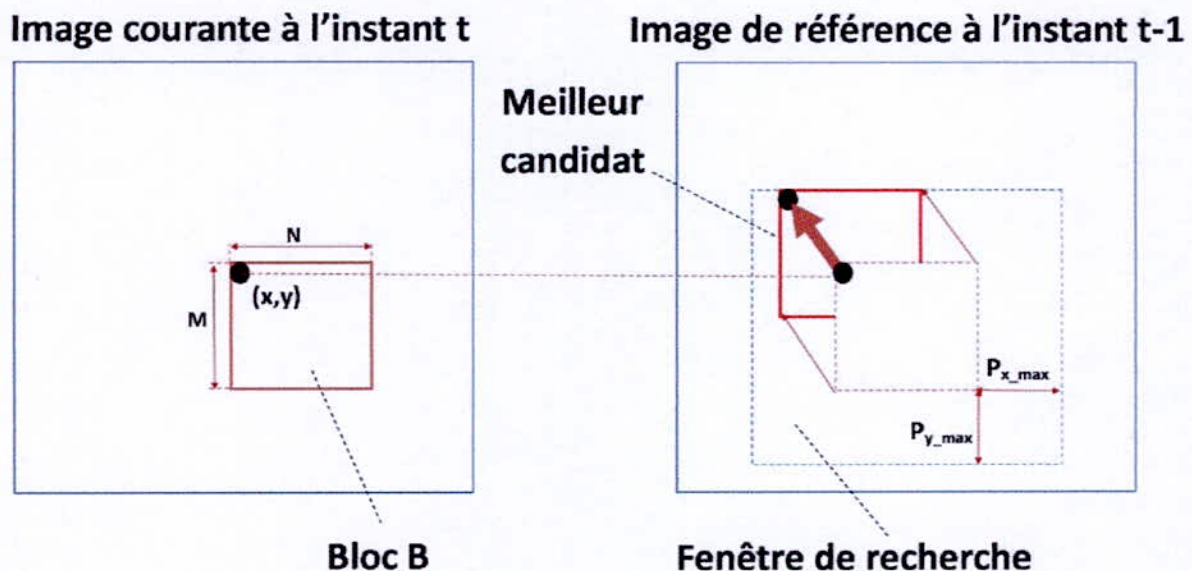


Fig.2.2 : Mise en correspondance de blocs

La ressemblance entre deux blocs est calculée sur l'ensemble du bloc. Ceci sous-entend une hypothèse supplémentaire selon laquelle tous les pixels d'un bloc effectuent le même mouvement [28]. On considère en général, un modèle de translation (figure 2.2) mais on peut considérer aussi des modèles plus complexes (rotation ,translation et rotation). Les méthodes de mise en correspondance de blocs peuvent être classées en fonction :

- ❖ du critère de mise en correspondance ;
- ❖ de la mise en œuvre nécessitant d'optimiser deux éléments: la dimension du bloc et celle de la fenêtre de recherche;
- ❖ de la stratégie de balayage (ou de recherche) de la fenêtre de recherche

2.4.1 Critères de mise en correspondance

Comme nous l'avons montré sur le schéma expliquant le principe des méthodes de BM, un critère de ressemblance ou de disparité est utilisé afin de trouver le meilleur bloc candidat dans une zone de recherche.

Nous donnons les expressions pour un bloc B de taille $N * M$, repérée sur le pixel de coordonnées $P = (x, y)$ dans l'image courante. Le décalage entre un bloc candidat et le bloc B de référence est noté $d = (i, j)$ et est borné par la taille de la zone de recherche. La position du meilleur bloc candidat est notée (\hat{i}, \hat{j}) .

Mesures de disparité - le meilleur candidat correspond au minimum de ces fonctions

Somme des différences absolues (SAD) [28].

$$\left. \begin{aligned} SAD(p, d) &= \sum_{k=1}^N \sum_{l=1}^M |B_2(k, l) - B_1(k + i, l + j)| \\ (\hat{i}, \hat{j}) &= \arg \min (SAD(p, d)) \end{aligned} \right\} \quad (2.8)$$

Somme des carrés des différences (SSD) [28].

$$\left. \begin{aligned} SSD(p, d) &= \sum_{k=1}^N \sum_{l=1}^M (B_2(k, l) - B_1(k + i, l + j))^2 \\ (\hat{i}, \hat{j}) &= \arg \min (SSD(p, d)) \end{aligned} \right\} \quad (2.9)$$

Mesures de ressemblance - le meilleur candidat correspond au maximum de ces fonctions

Coefficient d'inter-corrélation (CC) [28].

$$\left. \begin{aligned} CC(p, d) &= \sum_{k=1}^N \sum_{l=1}^M B_2(k, l) * B_1(k + i, l + j) \\ (\hat{i}, \hat{j}) &= \arg \max (CC(p, d)) \end{aligned} \right\} \quad (2.10)$$

2.4.2 Dimension optimale du bloc et de la fenêtre de recherche

La stratégie de recherche et les dimensions de la fenêtre de recherche et du bloc influencent le coût de calcul. La détermination de leurs dimensions optimales respectives nécessite toujours un compromis. Une grande fenêtre de recherche implique des calculs longs et un risque élevé de confusion du bloc recherché avec un bloc semblable. Mais plus la taille de la fenêtre de recherche décroît, plus le déplacement maximal estimé diminue. De la même manière, quand les blocs sont grands, le coût de calcul est grand, la résolution spatiale est faible et un bloc peut contenir des pixels appartenant à des objets différents. Inversement, un bloc trop petit peut ne pas contenir suffisamment d'information discriminante. Ce dernier problème apparaît, en particulier, dans les zones homogènes où la mise en correspondance n'est pas fiable, pouvant devenir aléatoire.

En plus des problèmes déjà mentionnés (temps de calcul, résolution spatiale et fiabilité de la mise en correspondance) on peut rappeler les limites de l'hypothèse selon laquelle tous les pixels d'un bloc ont le même vecteur déplacement. La méthode suppose des mouvements de translation ou localement assimilables (figure 2.2), mais cette hypothèse n'est pas respectée, par exemple dans le cas d'une rotation ou d'une homothétie rapide par rapport à la fréquence temporelle d'échantillonnage. Elle n'est pas respectée non plus à la frontière aux bords des objets animés de mouvement différent. Pour résoudre ce type de problème il est nécessaire de choisir un modèle de mouvement plus dans lequel la mise en correspondance s'effectue sur des blocs de formes arbitraires [29], [30], [31] ou sur des contours [32], [33], [34].

2.4.3 Algorithmes de parcours du voisinage

Rappelons qu'étant donné un bloc dans l'image courante, l'objectif est de trouver le bloc dans l'image précédente ou suivante (selon le type de prédiction utilisé) qui lui correspond au mieux selon des critères précédemment définis.

Il existe de nombreuses méthodes de block-matching cherchant à optimiser l'efficacité et la rapidité de l'algorithme [35].

2.4.3.1 Algorithme de recherche exhaustive (Full Search Algorithm)

Cet algorithme est le plus simple mais aussi le plus coûteux en calcul puisque toutes les comparaisons possibles entre blocs sont réalisées. Le coût en calcul de cet algorithme est fonction de la taille de l'espace de recherche (fenêtre de recherche). Le bloc retourné par l'algorithme sera celui qui minimise le critère de comparaison (SAD). C'est l'algorithme le plus simple, et n'effectue aucune minimisation. On teste tous les blocs de la zone de recherche. Dans la figure suivante on à la fenêtre de recherche est ici définie par un déplacement $[3,3]$; $[3,3]$ c'est-à-dire que le déplacement maximal autorisé est de trois dans toutes les directions. En effet, un déplacement maximal petit pénalise les grands déplacements et augmente naturellement l'erreur de prédiction dans le cadre d'une vidéo présentant de tels déplacements. Cependant, peu de blocs seront testés et l'algorithme sera alors rapide. A l'inverse, un grand déplacement autorisé permet de donner plus de liberté aux vecteurs de mouvement mais augmente de manière quadratique le nombre de blocs testés. Il y a donc un compromis entre qualité et rapidité de l'algorithme. Il est évident que cet algorithme soit d'une haute précision, mais ses besoins insupportables en temps de calcul défavorisent son adaptation [35].

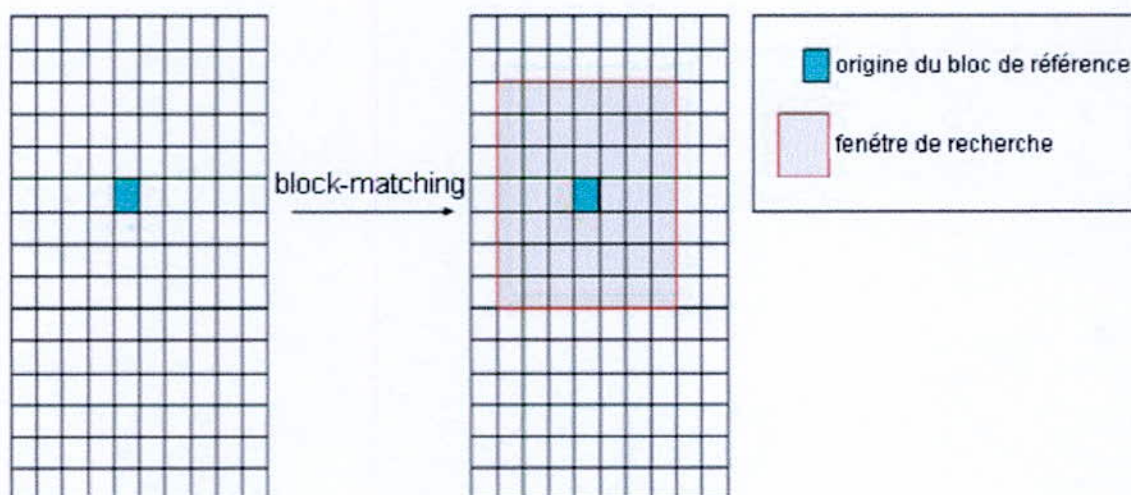


Fig.2.3 Recherche exhaustive[35].

2.4.3.2 Recherche en trois pas (3SS ou TSS : Three Step Search)

L'algorithme 3SS, proposé en 1981 par Koga, est basé sur une approche par raffinements successifs avec un pas à décroissance logarithmique. A chaque itération le sous-ensemble choisi est un "carré" de pas égal au pas précédent divisé par 2. Le pas initial est $[d/2]$. La figure 2.4 illustre la recherche en 3 pas pour des vecteurs mouvements d'amplitude ± 7 pixels [35].

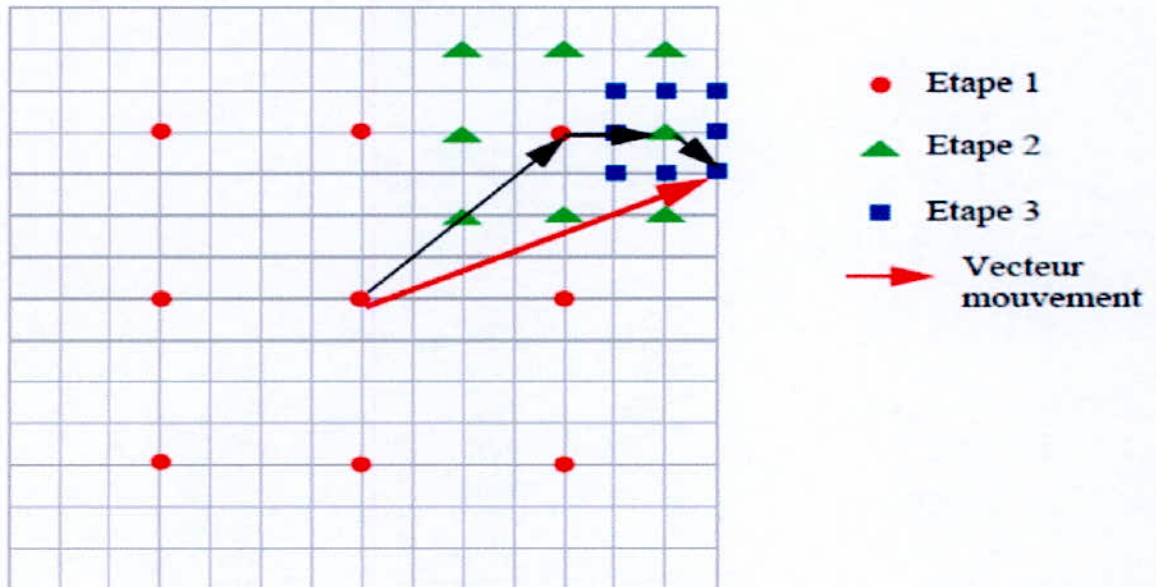


Fig.2.4 : Recherche en 3SS[35].

2.4.3.3 Recherche en quatre pas (4SS ou FSS : Four Step Search) :

L'algorithme 4SS utilise une zone de recherche avec 9 points de pixels sur une fenêtre de 5*5 dans la première étape.

Le centre de la fenêtre de recherche est alors déplacé au point de SAD. La taille de la fenêtre de recherche des deux prochaines étapes dépend de remplacement du point minimum SAD, s'il se trouve au centre de la zone de recherche, alors on passe à la dernière étape (étape 4) avec 3*3 fenêtre de recherche. Sinon, elle est mise à jour en 5X5 pour l'étape 2 ou l'étape 3

Dans la dernière étape, elle est réduite à la taille 3X3 pixels et la recherche s'arrête [35].

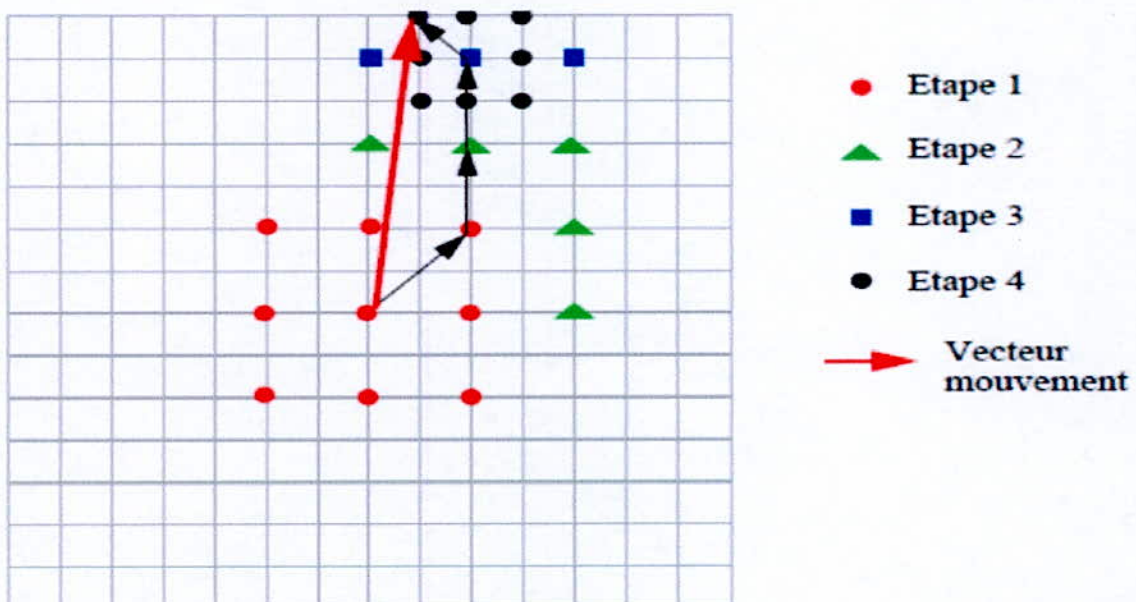


Fig.2.5 : Recherche en 4SS[35].

L'algorithme 4SS se résume comme suit (Figure 2.5):

Étape 1: Un point du minimum SAD est trouvé d'un modèle de 9 points de contrôle sur une fenêtre du 5x5 située au centre de la zone de recherche suivant les indications de (Figure 2.5)

Si le point du minimum SAD est trouvé au centre de la fenêtre de recherche, passez à l'étape 4; autrement passez à l'étape 2.

Étape 2 : La stratégie de recherche est la même que l'étape 1, mais finalement, on passe à l'étape 3.

Étape 3 : La stratégie de recherche est la même que l'étape 1, mais finalement, on passe à l'étape 4.

Étape 4 : La fenêtre de recherche est réduite à 3x3 comme indiqué dans (la Figure 2.5) et la direction du vecteur global de mouvement est considérée comme le point du minimum SAD parmi ces 9 points de la recherche.

Nous pouvons constater qu'il est possible de passer directement à l'étape finale d'une fenêtre 3x3 si le minimum SAD point est situé au centre de la fenêtre de recherche. Sur la base de ces quatre étapes modèle de recherche, nous pouvons couvrir l'ensemble des déplacements 15x15 mêmes fenêtres que de petites fenêtres de recherche 5X5 et 3X3 sont utilisées. Il ya chevauchement des points de contrôle sur la 5x5 fenêtres de recherche dans l'étape 2 et l'étape 3, donc le nombre total de points de contrôle est varié à partir de $(9 + 8) = 17$ à $(9 + 5 + 5 + 8) = 27$. Le pire cas de calcul de l'exigence 4SS est de 27 blocs.

2.4.3.4 Recherche logarithmique à deux dimensions

Proposé par Jain en 1981, cet algorithme utilise comme sous-ensemble de recherche le "plus". Le pas initial est $[d/4]$. Ce pas est divisé par 2 lorsque le minimum de distorsion courant est au centre du "plus" ou lorsque le bord de la fenêtre de recherche a été atteint. Pour la dernière itération (pas égal à 1), le sous-ensemble est un "carré". La figure 2.6 illustre une réalisation [35].

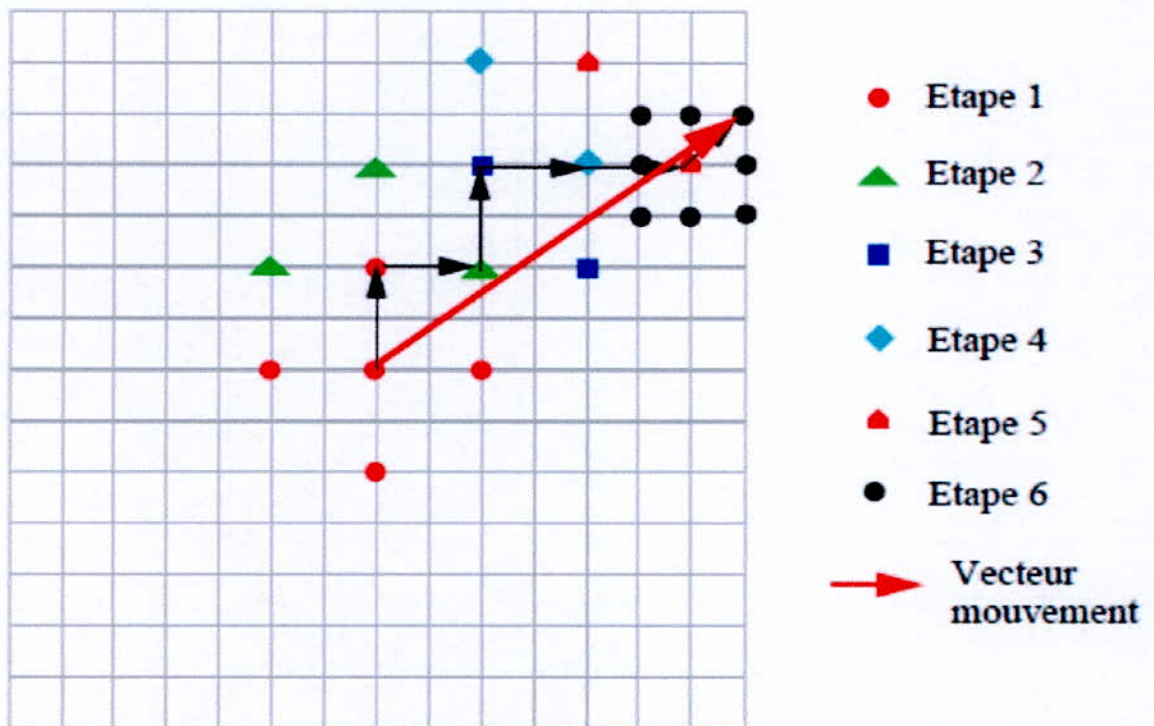


Fig.2.6 : recherche logarithmique à 2 dimensions[35].

2.4.3.5 Recherche orthogonale :

Puri en 1987 propose un algorithme qui procède par bonds successifs horizontaux puis verticaux. Les positions testées sont distantes initialement de $[d/2]$, le pas est divisé par 2 à chaque couple d'itérations (horizontale et verticale) [35].

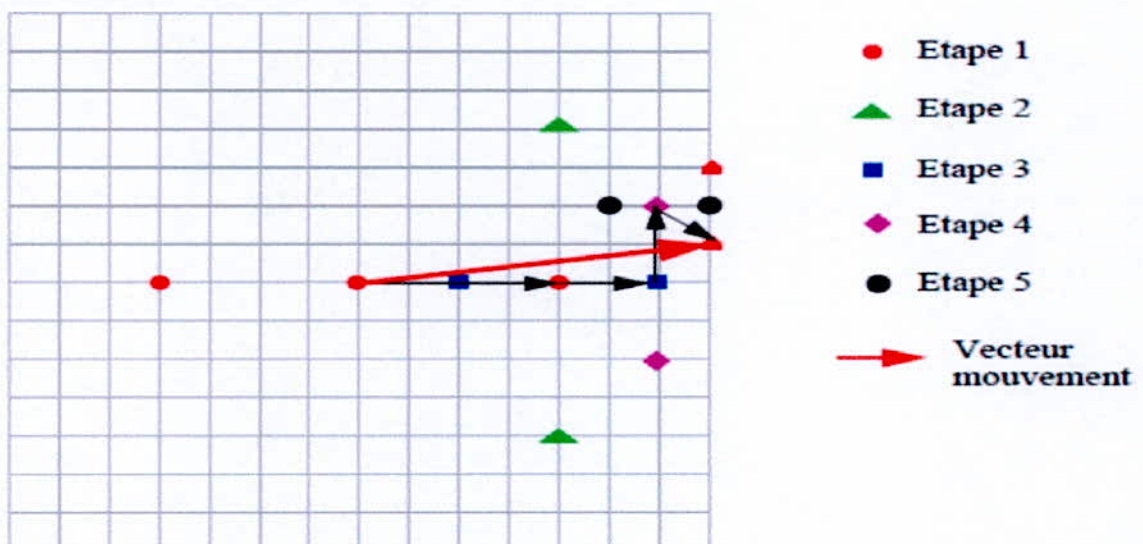


Fig.2.7 : Recherche orthogonale[35].

Pour palier au grand nombre d'opérations nécessaires et à une amplitude P bornée, plusieurs techniques peuvent être utilisées :

- ❖ Stopper les calculs le plus tôt possible (éviter les opérations inutiles).

Le calcul peut être arrêté dès que les valeurs déjà accumulées dépassent le minimum courant. Ainsi les boucles i et j peuvent être arrêtées prématurément si $SAD(j, i) > \text{minimum}$. Comme la SAD peut augmenter très vite lorsque l'on s'éloigne de l'optimum, de nombreux calculs sont évités (élimination de candidats).

- ❖ Limiter la recherche à un nombre restreint de candidats.

La recherche n'est alors plus exhaustive, l'idée générale est de réduire le nombre de candidats en supposant que la SAD varie de façon monotone dans la fenêtre de recherche. Il est alors possible de suivre une direction de descente privilégiée (algorithmes prédictifs et récursifs),

- ❖ Simplifier les calculs effectués.

Il est possible de simplifier le critère d'erreur afin de réduire les besoins lors de l'implantation. Par exemple le nombre de bits peut être réduit pour abaisser le coût d'une implantation câblée ou bien le nombre de pixels pris en compte dans le critère d'erreur afin de réduire le nombre de calculs.

Conclusion générale

Dans ce master, j'ai présenté plusieurs techniques d'estimation de mouvement. L'estimation peut se faire selon deux approches ;

1. Calculer le mouvement apparent (déplacement/vitesse instantanée) de chaque pixel ce qui conduit à un flux optique dense.
2. Apparier certaines structures spatiales (mise en correspondance) pour chaque couple d'images.

Les méthodes basées sur le gradient sont très précises, mais ne permettent d'estimer que de faibles déplacements. Les techniques fréquentielles sont robustes, mais nécessitent une étape supplémentaire de mise en correspondance. Les techniques permettant la mise en correspondance de blocs restent la solution appropriée en termes de complexité, temps calcul, cohérence des résultats pour une application de compression vidéo.

References Bibliographies

- [1] J. Konrad - "Motion detection and estimation" - Image Processing HandBook, Networking and Multimedia, 2000, pag. 207-227;
- [2] J.K. Aggarwall, Q.Cai - "Human motion analysis: a review"-Proceedings of the IEEE Non-Rigid and Articulated Motion Workshop, 1997, Piscataway, USA, p. 1-13;
- [3] J.K. Aggarwall, N. Nandhakumar - "On the computation of motion from sequences of images - A review" - Proceedings of IEEE, 1988, Vol.76, No.8, p. 917-935;
- [4] T.S. Huang, R.Y. Tsai - "Image sequences analysis: Motion estimation" -In: T.S. Huang, editor "Image sequence analysis", Springer-Verlag, Berlin, 1981, p. 1-18;
- [5] W. Martin, J.K. Aggarwall - "Motion understanding: Robot and human vision" - Boston: Kluwer Academic Publishers, 1988, 387 pages;
- [6] L.O. Chua, L. Yang - "Cellular neural networks: Theory and applications" - IEEE Trans, on Circuits and Systems, (CAS), 1988, Vol.35, p. 1257-1290;
- [7] F. Dufaux, F. Moscheni - "Motion estimation techniques for digital TV: A review and a new contribution" - Proceedings of IEEE, 1995, Vol.83, No.6, p. 858-876;
- [8] B.K.P. Horn, B.G. Schunck "Determining Optical Flow" - Artificial Intelligence, 1981, Vol.17, p. 185-203;
- [9] C. Stiller, J. Konrad - "On models, criteria and search strategies for motion estimation in image sequences" - IEEE Signal Processing Magazine, 1998, p. 1-41;
- [10] W. B. Thomson, K. M. Mutch, V. A. Berzins - "Dynamic occlusion analysis in optical flow fields" - IEEE Trans, on PAMI, 1985, Vol.7, No.4, p. 374-383;
- [11] N. Ohta - "Optical flow detection by color images" - NEC Res. and Dev., 1990, No.97, p. 78-84;
- [12] : J. Biemond, L. Looijenga, D. E. Boekee, and R. H. J. M. Plompen. A pel-recursive wiener-based displacement estimation algorithm. Signal Process, 13(4) :399-412, 1987.
- [13] : C. Cafforio and F. Korea. The differential method for image motion estimation. Image sequence processing and dynamic scene analysis, pages 104-124. 1983.
- [14] : F. Kelly and A. Kokaram. Graphics hardware for gradient based motion estimation. IS and T/SPIE Electronic Imaging - Embedded Processors for Multimedia and Communications, 2004.
- [15] : A.N. Netravali and J.D. Robbins. Motion-compensated television coding : Part i. Bell Syst. Technical Journal, 58 :631-670, mar 1979.

- [16]: D. R. WALKER and K. R. RAO. Improved pel-recnrsive motion compensation. IEEE Transactions on Communications, 10 :1128—1134, October 1984.
- [17]: E. D. Castro and C. Morandi. Registration of translated and rotated images using finite fourier transforms. In IEEE Transactions on Pattern Analysis and Machine Intelligence, volume PAMI9 (5), pages 700-703. Sept 1987.
- [18]: R. Storey. HDTV Motion Adaptive Bandwidth Reduction using DATV. Technical Report BBC RD 1986/5, BBC Research Department, 1986.
- [19]: G.A. Thomas. Television motion measurement for DATV and other applications. Technical He-port BBC HI) 1987/11. BBC Research Department, 11 1987.
- [20] D.J. Hecger. Optical flow using spatiotemporal filters. International Journal of Computer Vision, pages 297-302, 1988.
- [21] A. Spinei, D. Pellerin. and J. Héroult. Spatiotemporal energy-based method for velocity estimation. Signal Processing, 65(6) :347-362, 1998.
- [22] A.B. Torralba and J. Héroult. An efficient neuromorphic analog network for motion estimation. IEEE Transactions on Circuits and Systems-I, 46(2), February 1999.
- [23] D. Fleet and A. Jepson. Computation of component image velocity from local phase information. International Journal of Computer Vision, 5 :77-104, 1990.
- [24]H. Nagel, W. Enkelmann, An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1986, Vol. 8, n° 1, p. 565-593.
- [25] J. L. Barron, D. J. Fleet, Beauchemin, Performance of optical flow techniques, International Journal of Computer Vision, 1992, Vol. 12, n° 1, p. 43-77.
- [26] L. Jacobson and H. Wechler, « Derivation of optical flow using spatiotemporal frequency approach ». CVGIP, 38, 1987, pp. 29-65.
- [27] J. Weng. « A theory of image matching », Proc. 3rd International Conference on Computer Vision, Osaka, Japon, 1990, pp. 200-209.
- [28] F.H. Adelson and J.R. Bergen, « Spatiotemporal energy models for the perception of motion ». Journal of the Optical Society of America A., vol. 2, n°2, 1985, pp. 284-299.
- [29]D.J. Hecger. Optical flow using spatiotemporal filters. International Journal of Computer Vision, pages 297-302, 1988.
- [30]D. Fleet and A. Jepson. Computation of component image velocity from local phase information. International Journal of Computer Vision, 5 :77-104, 1990.
- [31] : Rapport de Stage : Estimation de mouvement subpixelique par blocs adaptée a la couleur avec modèle de mouvement, 14/12/2004.

[32] M. Gallant, G. Cote, F. Kossentni, « An Efficient Computation-Constrained Block-Based Motion Estimation Algorithm for Low Bit Rate Video Coding », IEEE Trans, on Image Processing, Vol 8, No. 12, pp. 1816-1823, Dec. 1999.

[33] Implantation optimisé d'estimateurs de mouvement pour la compression video sous plates-formes hétérogène multi-composant, 6/12/2007.

[34] P. Brigger, J. Hoeg, M. Unser - "B-Splines snakes: A flexible tool for parametric contour detection" - IEEE Trans, on Image Processing, 2000, Vol.9, No.9, p. 1484-1496;

[35] : Sébastien ROUX, Thèse pour obtenir le grade de docteur de l'Inpg, sujet : adéquation algorithme-architecture pour le traitement multimedia embarqué.