

n° d'ordre : 1052

50376
1993
24

50376
1993
24

THESE

présentée à

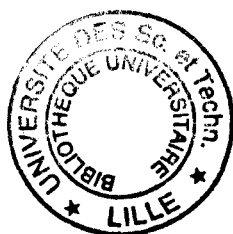
L'UNIVERSITE DES SCIENCES ET TECHNOLOGIES DE LILLE
UFR D'IEEA

en vue d'obtenir le titre de

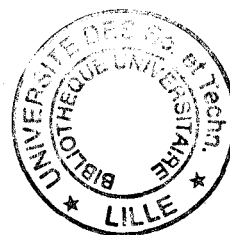
DOCTEUR de L'UNIVERSITE

Spécialité : Electronique

par



Patrick NIKYEMA



**CONTRIBUTION A L'ETUDE DE L'UTILISATION DES TECHNIQUES
MULTIRESOLUTION DANS LA COMPENSATION DE MOUVEMENT
DE SEQUENCES D'IMAGES**

soutenue le 22 Janvier 1993 devant la commission d'examen

Membres du jury : L. Raczy

M. Barlaud

H. Vu Thien

J. Froment

V. Devlaminck

J. P. Dubus

Président

Rapporteur

Rapporteur

Examineur

Examineur

Examineur

Ce travail a été réalisé au Laboratoire de Mesures Automatiques de L'Université des Sciences et Technologies de Lille.

J'exprime ma reconnaissance à Monsieur le Professeur J. P. Dubus, Directeur du Laboratoire ainsi qu'à Monsieur V. Devlaminck, Maître de Conférences à l'USTL, pour avoir assuré la direction de cette étude et pour leur aide constante et efficace à la réalisation de ce travail.

Monsieur le Professeur L. Raczky me fait l'honneur de présider ce jury, qu'il trouve ici l'expression de ma sincère gratitude.

Je voudrais formuler mes vifs remerciements à Messieurs les Professeurs M. Barlaud, de l'Université de Nice-Sophia Antipolis, et H. Vu Thien, du Conservatoire National des Arts et Métiers de Paris, pour avoir accepté de juger ce travail ainsi que pour leur présence dans ce jury en tant que rapporteurs.

Ma reconnaissance s'adresse également à Monsieur J. Froment Maître de Conférences à l'Université de Paris V qui a consenti à examiner ce mémoire, faire partie de cette commission d'examen, et qui a toujours répondu spontanément à mes sollicitations tout au long de ce travail.

Mes remerciements vont aussi à Monsieur Lannoo, Chef du Département Génie Electrique de l'IUT de Calais pour son soutien "logistique", et aussi à tout le personnel du dit établissement pour son accueil.

Que les membres du Laboratoire, A. Garbi, F. Wauquier, F. Dufresnois, S. Reboul, A. Thery, F. Wattrelot, A. Cherradi, et en particulier A. Taleb-Ahmed, soient remerciés pour l'atmosphère conviviale qu'ils y font régner.

Je remercie les membres du personnel, en particulier Mesdames Castegnier et Dekimpe, Messieurs J. Lemaire et J. P. Dehorter pour leurs contributions respectives.

Enfin, je ne saurais oublier ma Famille, mes Amis, ainsi que tous ceux que je ne peux citer ici nommément et qui m'ont soutenu pendant ces années. Qu'ils reçoivent l'expression de ma très profonde gratitude.

SOMMAIRE

II.5 - Construction d'une décomposition multirésolution invariante par changement d'échelle	45
II.6 - La pyramide de Burt et Adelson	49
II.7 - Approximation d'une fonction	55
II.8 - Définition d'une analyse multirésolution de $L^2(\mathbb{R})$	58
II.9 - Analyse multirésolution et ondelettes	60
II.9.1 - La représentation en ondelettes à une dimension	62
II.9.2 - Généralisation à la dimension n : cas des images	66
 Chapitre III : Compensation de mouvement hiérarchique par décomposition multirésolution	71
 III.1 - Introduction	71
III.2 - Principe de l'estimation de mouvement multirésolution	74
III.3 - Caractéristiques de la séquence de travail	80
III.4 - Comparaison entre monorésolution et multirésolution	81
III.5 - Problème des occlusions	86
III.6 - Intégration des pyramides de détails	89
III.7 - Problème de la délocalisation	94
III.8 - Influence de la taille des filtres	98
 Conclusion	109
 Annexe	111
 Références bibliographiques	115
 Bibliographie	123

INTRODUCTION

INTRODUCTION

L'utilisation, de plus en plus fréquente de nos jours, des signaux numériques, amène à se poser des questions sur les capacités des mémoires nécessaires pour leur archivage ou les débits requis pour leur transmission. C'est tout particulièrement le cas avec les images numériques pour lesquelles on peut citer deux chiffres à titre d'exemple. Un film d'une heure à la cadence habituelle (25 images par seconde) avec une définition comparable à celle de la télévision actuelle, nécessiterait à l'état brut, une capacité de stockage d'environ 60 giga-octets. Un autre exemple est le débit binaire requis pour effectuer la transmission d'images de télévision haute définition, qui serait de l'ordre du giga-bits par seconde, si l'on devait effectuer cette transmission sans traitement préalable.

Il faut toutefois remarquer que ces chiffres sont dûs à la fois à la complexité de l'information contenue dans ce type de signal, mais également au mode de représentation, c'est-à-dire au codage adopté. Partant de cette constatation, on peut imaginer un premier type de traitement qui consiste à transformer le mode de représentation afin d'en obtenir un plus approprié, c'est-à-dire dont l'entropie est plus faible. Ceci permet, après un codage de type entropique, d'avoir un gain de compression. Dans la mesure où la transformation est réversible, on réalisera une compression sans perte d'information.

En général, les gains de compression obtenus par de telles techniques ne sont pas très élevés. Si l'on cherche à les augmenter de façon conséquente, il est souvent nécessaire de tronquer l'information contenue dans l'image, et ceci de manière irréversible. On réalise alors une compression avec perte d'information ou irréversible. Dans la plupart des cas, ceci est réalisé en restreignant le nombre de valeurs possibles que l'on attribue au signal, c'est-à-

dire en le quantifiant. Les gains maximaux produits par de telles techniques sont intimement liés à la définition d'une information pertinente dans l'image, notion étroitement dépendante de l'application traitée.

Dans la pratique, les techniques de compression font appel en général à une mise en cascade de ces deux approches de compression que nous avons présentées. C'est en particulier le cas dans le problème de la compression des séquences d'images numériques. Le changement de représentation qui est le plus fréquemment appliqué consiste à construire un signal image, obtenu par différence des images successives de la séquence, ceci afin de profiter de la corrélation temporelle qui existe entre les pixels d'une image à l'autre dans la séquence. En réalité, afin d'exploiter pleinement cette corrélation, il convient de réaliser, préalablement à l'opération de différence, une action de compensation des mouvements entre les deux images en cours de traitement. Le gain potentiel obtenu avec un tel changement de représentation est évidemment lié à la qualité de la compensation de ces mouvements.

Notre travail s'inscrit dans le cadre de l'amélioration des estimateurs de mouvement qui permettent cette compensation. Pour ce faire, nous commençons par rappeler, dans le premier chapitre, les principales techniques de compression qui s'appliquent en traitement des images pour arriver à celles qui nous intéressent plus particulièrement dans le domaine des séquences d'images. Nous présentons les deux techniques d'estimation du mouvement appliquées à l'heure actuelle dans les problèmes de compensation. La technique, dite de block matching, que nous avons retenue, conduit tout naturellement à utiliser la représentation multirésolution des signaux afin d'augmenter la qualité des résultats obtenus.

Dans le chapitre deux, nous rappelons l'intérêt de cette représentation et décrivons sa mise en oeuvre dans le cas d'une analyse multirésolution par pyramide d'ondelettes.

Nous proposons ensuite une méthode combinant le block matching et la décomposition en ondelettes, à partir de plusieurs filtres, dans le but d'améliorer les performances du système dans le cas de séquences d'images bruitées. Des résultats sont présentés, ainsi que leur interprétation, de manière à évaluer cette technique.

CHAPITRE I :

LA COMPRESSION D'IMAGES

LA COMPRESSION D'IMAGES

I.1 - Les méthodes de base

Notre propos étant de nous intéresser à une séquence d'images numériques entrelacées noir et blanc, nous allons effectuer un aperçu historique pour faire un état de l'art en la matière.

L'idée première a été d'utiliser les propriétés aussi bien statistiques que géométriques d'une image pour la comprimer, en se ramenant au problème du codage d'une source en théorie de l'information. On considère la fonction $L(x,y,t)$ représentant la luminance de l'image comme étant une fonction aléatoire et on cherche une stratégie de codage telle que pour un débit de transmission donné on minimise la distorsion moyenne définie par [NET80a] :

$$D = E [d (L , \hat{L})] \quad (1.1)$$

où $d(L, \hat{L})$ est une mesure de la distorsion entre deux champs d'intensité L et $\hat{L}$ (correspondant à deux images) ; $\hat{L}$ étant la version codée de L . Les outils mathématiques nécessaires à la résolution de ce problème proviennent de la théorie du taux de distorsion élaborée par Shannon [SHA49],[SHA60]. Très vite, cette approche s'est trouvée confrontée à quatre problèmes principaux :

- l'absence d'un modèle statistique performant pour représenter l'image.
- la non prise en compte du processus de la vision humaine par le critère de distorsion.
- le calcul des fonctions de distorsion.
- la synthèse d'un codeur adéquat.

L'utilisation de la théorie du taux de distorsion s'étant heurté à ces écueils, on a vu se développer plusieurs autres techniques de codage, exploitant la redondance observée au niveau de l'image, pour réaliser la compression.

Dans le cas d'une séquence d'images, il existe deux types de redondance que l'on peut mettre à profit :

- la redondance intra-image ou spatiale : il existe une forte corrélation entre les pixels appartenant à un même voisinage. Autrement dit, les valeurs de pixels $I(x,y,t)$ et $I(x+dx,y+dy,t)$ sont très dépendantes les unes des autres.
- la redondance inter-image ou temporelle : lorsque l'on considère deux images consécutives de la séquence aux instants t et $t+dt$, vu la fréquence de balayage (25 images par seconde par exemple), la valeur $I(x,y,t)$ est fortement corrélée à $I(x,y,t+dt)$ compte tenu du peu de variation survenu au cours de dt .

Nous allons commencer par décrire quatre techniques différentes de base qui sont :

- la modulation par impulsions codées (MIC), ou pulse code modulation (PCM).
- le codage prédictif ou modulation par impulsions codées différentielle (MICD), ou differential pulse code modulation (DPCM).
- le codage par transformation.
- le codage par interpolation.

I.1.1 - La modulation par impulsions codées [OLI48]

Elle consiste à discrétiser un signal analogique en temps puis en amplitude ainsi que l'indique le schéma de la figure I-1.

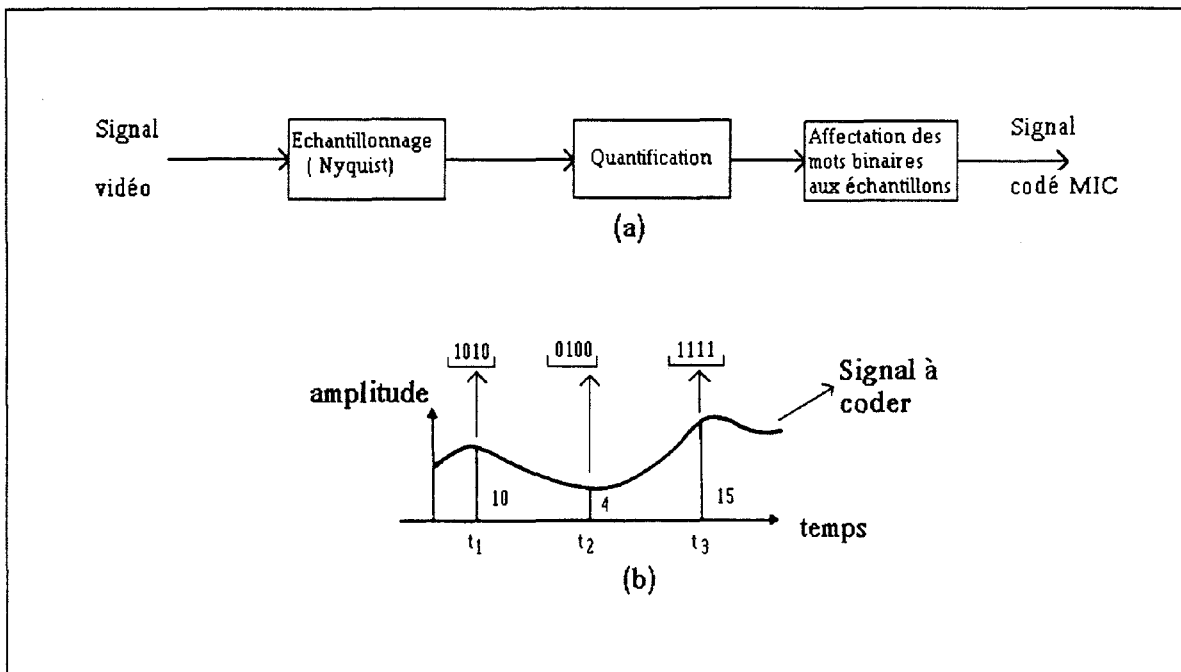


Figure I-1. a) Codeur MIC. b) Exemple de représentation binaire d'échantillons. Le niveau maximum d'amplitude est 15.

Cette méthode ne tire pas parti de la redondance du signal et n'est pas vraiment utilisée pour faire de la compression, dans la mesure où le gain de compression se situe uniquement dans le choix du pas de quantification. Ce pas sera choisi de manière à ne pas trop altérer l'image décodée.

I.1.2 - Le codage prédictif [DeJ52],[ELI55]

Le principe de base de cette technique est illustré par la figure I-2. Une estimation $\hat{e}$ de l'échantillon à coder est fournie par le prédicteur, à partir des valeurs des échantillons déjà transmis. L'erreur e résultant de la différence entre la valeur vraie et la valeur estimée est ensuite quantifiée puis codée par un codage à longueur fixe ou variable du type Huffman, et transmise dans le canal.

Après décodage des mots transmis, le récepteur reconstruit l'échantillon en additionnant la valeur estimée et l'erreur de prédiction.

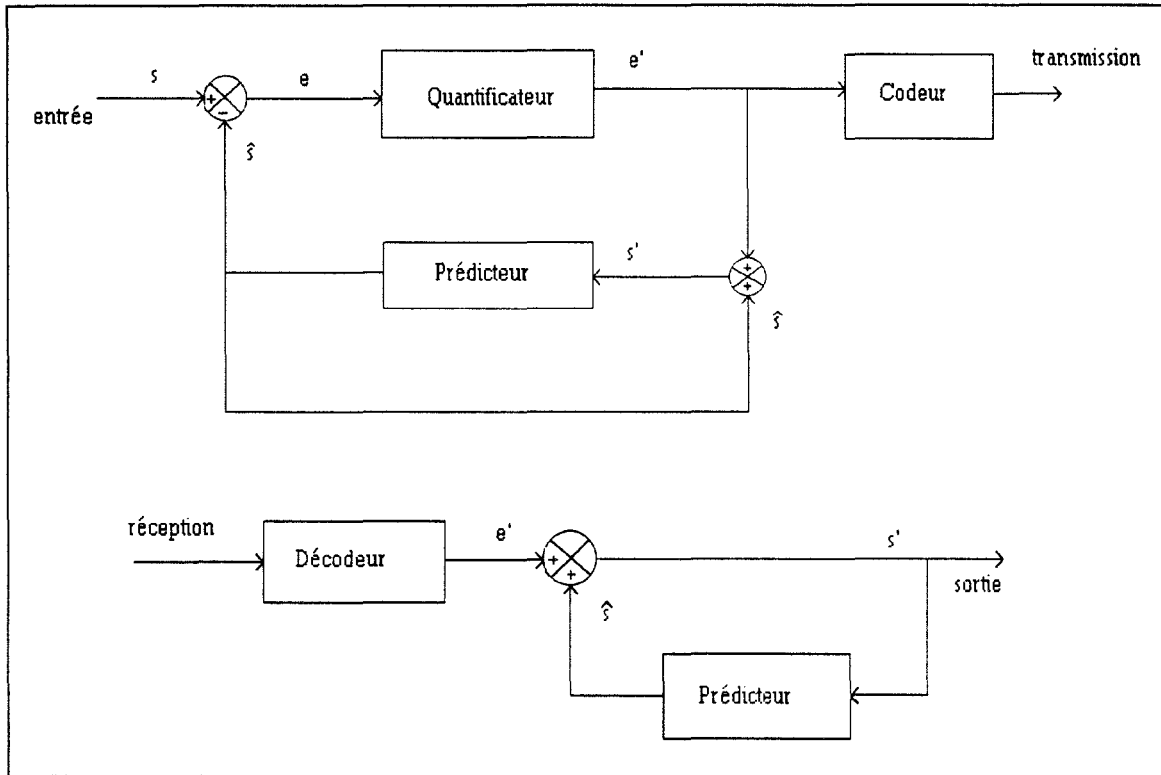


Figure I-2. Schéma bloc d'un système MICD.

Les performances de ce système seront donc fonction des trois éléments-clé que sont le prédicteur, le quantificateur et le codeur.

I.1.3 - Le codage par transformation [WIN72]

Nous avons mentionné plus haut qu'une image, par nature, possédait une certaine redondance intra-image. L'idée de base dans le codage par transformation est d'extraire au préalable cette redondance en changeant le mode de représentation de l'image, la transformation amenant à travailler dans un domaine où la séparation entre l'information utile et la redondance est

plus aisée. On découpe généralement l'image en petits blocs que l'on décompose par projection sur une base orthogonale, obtenant ainsi des composantes de l'image dans un plan dit transformé, sous forme de coefficients décorrélés. Seuls les coefficients jugés pertinents sont alors quantifiés, codés puis transmis.

A la réception, les bits reçus sont décodés, puis par application de la transformée inverse, l'image de départ est recouverte aux erreurs de quantification et de reconstruction près. Toutes ces opérations sont schématisées par la figure I-3.

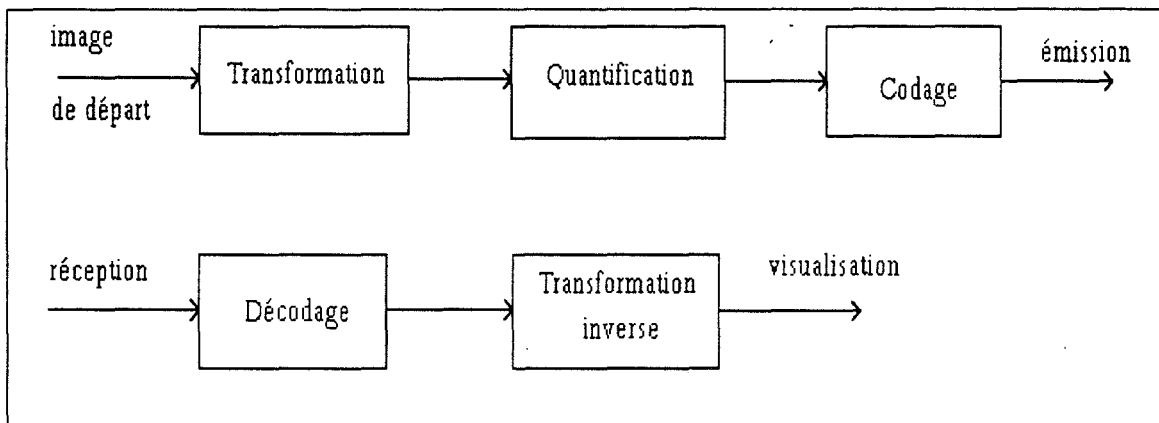


Figure I-3. Principe du codage par transformation.

Il ressort de ce principe que les paramètres qui vont déterminer l'efficacité du système de codage par transformation sont : la taille des blocs (le cas échéant), le choix de la transformation, le type de balayage utilisé pour sélectionner les coefficients à transmettre, la quantification, et le codage.

Puisque le gain de compression est essentiellement dû aux coefficients non transmis après la transformation, il est souhaitable que celle-ci concentre l'information utile de l'image en un nombre de coefficients le plus petit possible. Le corollaire de cette exigence est que la transformation puisse, à la reconstruction de l'image, minimiser l'erreur due à la perte des coefficients non transmis. Parmi les transformations existantes, celle dont la série converge le plus vite et qui répond aux critères cités, c'est la transformation de Karhunen-Loeve [HOT33],[KAR47]. Malheureusement, sa réalisation pratique

est peu aisée car il n'existe pas d'algorithme de calcul rapide. Elle sert alors le plus souvent de référence pour les performances à atteindre par les transformations d'implantation plus facile qui lui sont préférées. Une tranformation simple à mettre en oeuvre et dont le comportement est très proche de celui de Karhunen-Loeve est la DCT (Discrete Cosine Transform ou Transformation en Cosinus Discret). Un exemple de résultat obtenu par DCT est donné par la figure I-4.

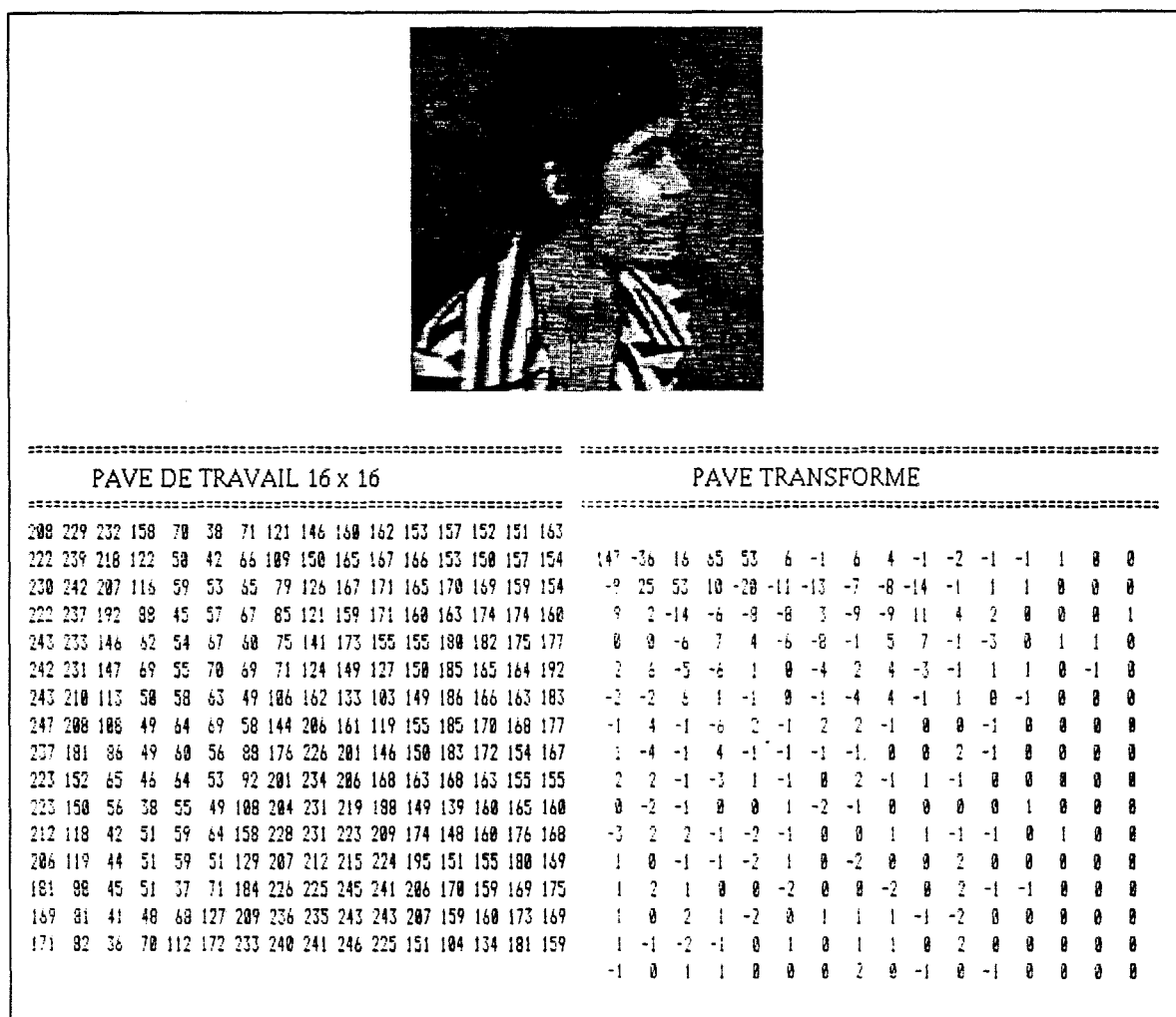


Figure I-4. Exemple d'application de la DCT sur un pavé 16 x 16.

Dans le plan transformé, les coefficients significatifs sont regroupés dans le coin supérieur gauche du bloc. Par un balayage adéquat, on peut donc limiter la zone à transmettre.

Dans la littérature, on trouve sous l'appellation de codage hybride, la combinaison de ces deux derniers types de codage [HAB74]. On effectue d'abord une transformation avant d'appliquer un codage prédictif. Le schéma de principe apparaît sur la figure I-5.

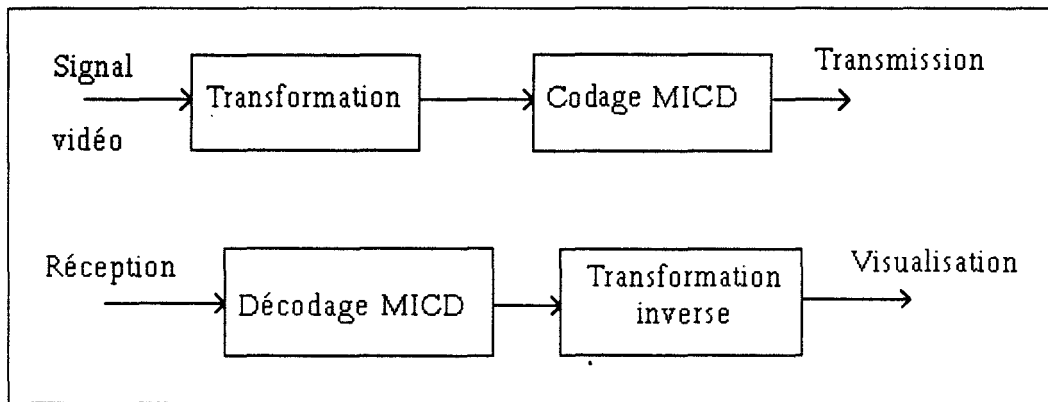


Figure I-5. Principe du codage hybride.

I.1.4 - Le codage par interpolation [HOC67],[KOR67]

Dans le codage par interpolation, un certain nombre de pixels sont transmis, les autres étant reconstitués à la réception par interpolation. La figure I-6 montre un exemple où un pixel sur deux a été transmis au cours du balayage horizontal.

Le pixel E est le résultat d'une interpolation linéaire faite à partir de A,B,C, et D. Les performances de cette méthode vont dépendre d'une part de la technique utilisée à l'émission (on peut par exemple transmettre une ligne sur deux au lieu d'un pixel sur deux comme dans notre exemple), d'autre part du type d'interpolation employé à la reconstitution.

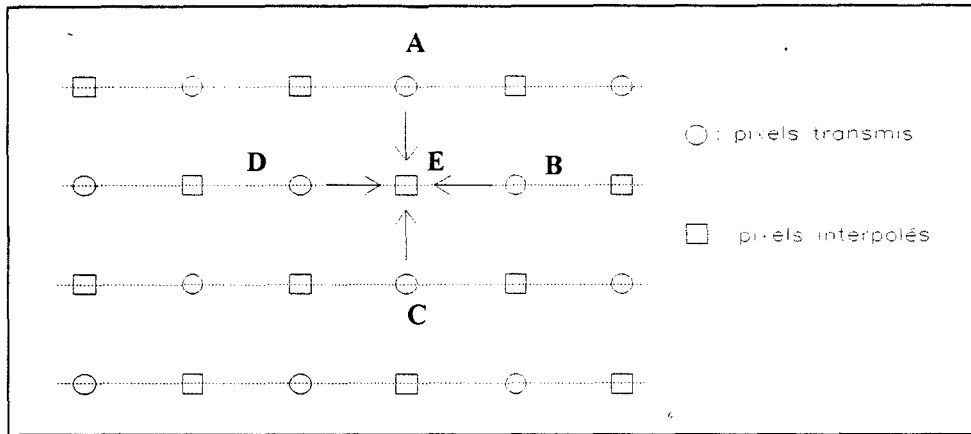


Figure I-6. Exemple de codage par interpolation. Sur chaque ligne, on choisit de ne transmettre qu'un pixel sur deux.

Toutes ces méthodes que nous venons d'énumérer sont celles qui sont apparues chronologiquement en premier et que l'on peut qualifier de techniques de première génération. Elles présentent toutes des variantes fixes ou adaptatives suivant que les paramètres déterminants de chaque méthode sont fixés ou rendus adaptatifs en fonction des caractéristiques locales des différentes zones de l'image. Ces techniques exploitent pour la plupart la redondance statistique de l'image pour opérer la compression. Cependant, il reste que le dernier récepteur pour toute image est l'œil humain. Il serait donc intéressant de tenir compte de ce que l'œil humain perçoit et comment il le perçoit, pour développer d'autres techniques.

Parallèlement à celles que nous avons citées s'est développée une méthode appelée "Synthetic High System" (SHS), prenant en compte le critère de la vision [SCH59],[SCH63],[SCH72]. A cause de sa complexité de mise en oeuvre pour l'époque, le SHS n'a pas connu l'essor qu'il méritait à ce moment-là, mais il reste l'ancêtre de plusieurs méthodes actuelles. Son principe est le suivant.

L'image de départ est divisée en deux parties. La première, obtenue par filtrage passe-bas, contient la texture. Elle peut donc, en vertu du théorème

de l'échantillonnage, être codée avec moins d'échantillons. La deuxième partie de l'image contient les contours, extraits par un opérateur de type laplacien ou gradient. On effectue ensuite un seuillage pour ne conserver que les points pertinents des contours. Les adresses et les valeurs des points conservés sont alors codées. Le schéma fonctionnel du SHS est représenté par la figure I-7.

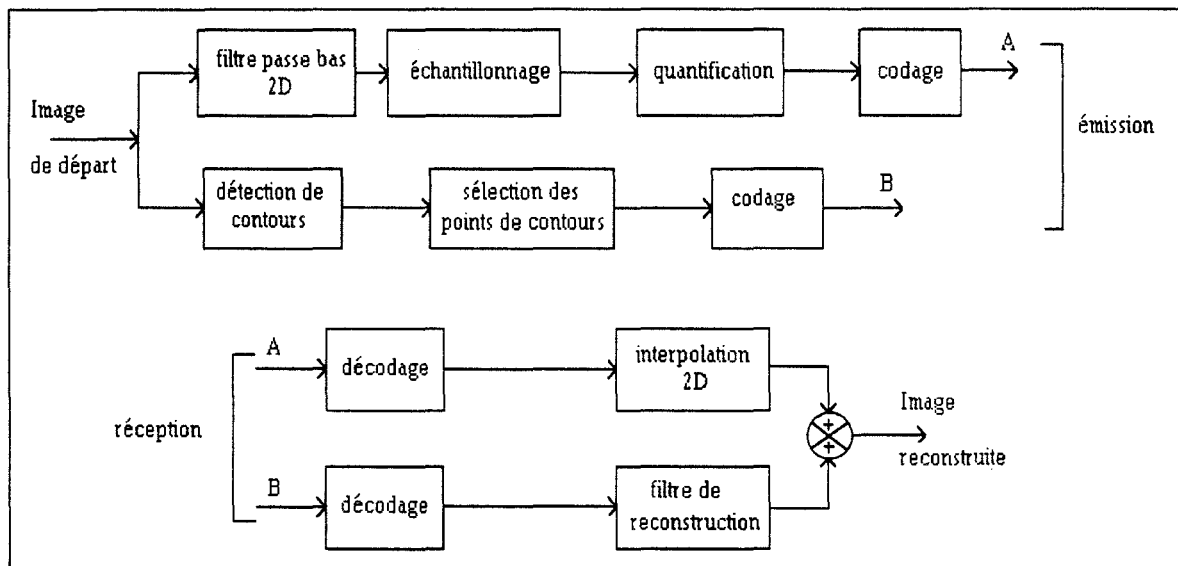


Figure I-7. Schéma fonctionnel du SHS.

Les propriétés du filtre de reconstruction sont déterminées à partir du passe-bas utilisé pour obtenir la texture de l'image. Les progrès effectués dans la connaissance du mécanisme de la vision humaine (sensibilité directionnelle des neurones, traitement séparé des contours et des textures) ainsi que l'évolution des méthodes de segmentation, ont donné naissance à des techniques de compression permettant d'améliorer les performances du SHS [KUN85].

Un autre type de codage utilisé en compression et qui tire profit du fonctionnement de l'oeil est le codage pyramidal dont nous parlerons dans le chapitre II.

Il est clair que la classification des différentes méthodes que nous venons de faire n'est pas exhaustive et aurait pu être basée sur d'autres critères de regroupement. En outre, nous avons intentionnellement omis de faire cas des méthodes inter-image qui vont constituer le thème de la partie suivante.

I.2 - La compensation de mouvement

I.2.1 - Introduction

Les images de télévision sont obtenues par le balayage d'une scène plusieurs fois par seconde, et cela, même s'il ne s'est produit aucun mouvement d'une image à l'autre. On aboutit ainsi à une importante redondance inter-image dans une séquence d'images. La manière la plus simple de diminuer cette redondance est de faire la soustraction pixel à pixel entre deux images consécutives k et $k-1$ (Figure I-8):

$$I(X,t) - I(X,t-1) \quad (I.1)$$

où $I(X,t)$ désigne l'intensité du pixel à la position $X(i,j)$, dans l'image k , à l'instant t .

Au lieu de transmettre l'image k , on n'envoie que la différence directe FD (Frame Difference) ainsi obtenue. A la réception, l'image précédente sera mise à jour par addition de cette image de différence directe FD. Cette technique a été à l'origine mise en oeuvre par Mounts [CAN71],[MOU69] et possède également des variantes fixes et adaptatives.

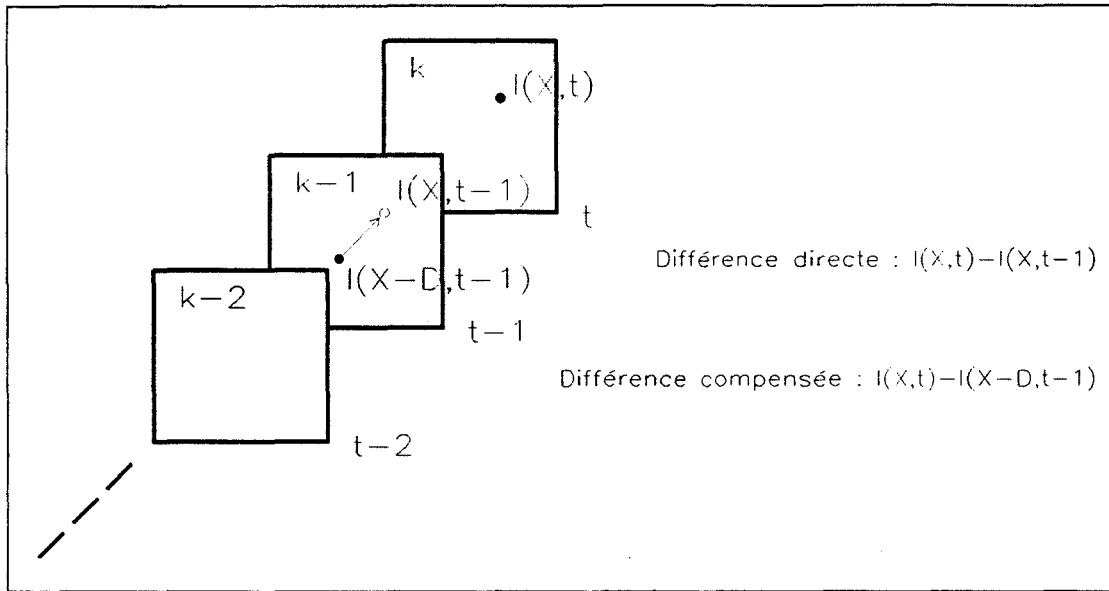


Figure I-8.

Des améliorations ont par la suite été apportées au principe de base, en prenant en compte le mouvement dans l'image [HAS72],[HAS78],[NET79]. En effet, la portion de signal correspondant aux parties statiques de l'image n'a pas besoin d'être transmise à chaque fois, si l'on dispose d'une mémoire suffisante au niveau du récepteur pour la stocker. Au lieu d'effectuer la différence directe par (I.1), on procède au préalable à la détermination du mouvement survenu entre les instants t et $t-1$. Si D représente le déplacement subi entre $t-1$ et t par le pixel à la position X de l'image courante k , sa position dans l'image précédente $k-1$ était donc $X-D$. Le champ des vecteurs de déplacement étant déterminé pour toute l'image, l'opération de compensation consiste à déplacer tous les pixels de l'image $k-1$ conformément à ce champ. On peut alors effectuer la soustraction pixel à pixel de (I.1) qui revient à faire:

$$I(X,t) - I(X-D,t-1) \quad (I.2)$$

Cette opération conduit à la différence d'image compensée ou DFD (Displaced Frame Difference). On comprend que la quantité d'information contenue dans la DFD soit plus faible que celle provenant de la différence directe. La figure I-9 montre un exemple pratique d'application de ce principe dans un schéma de codage hybride utilisant la DCT.

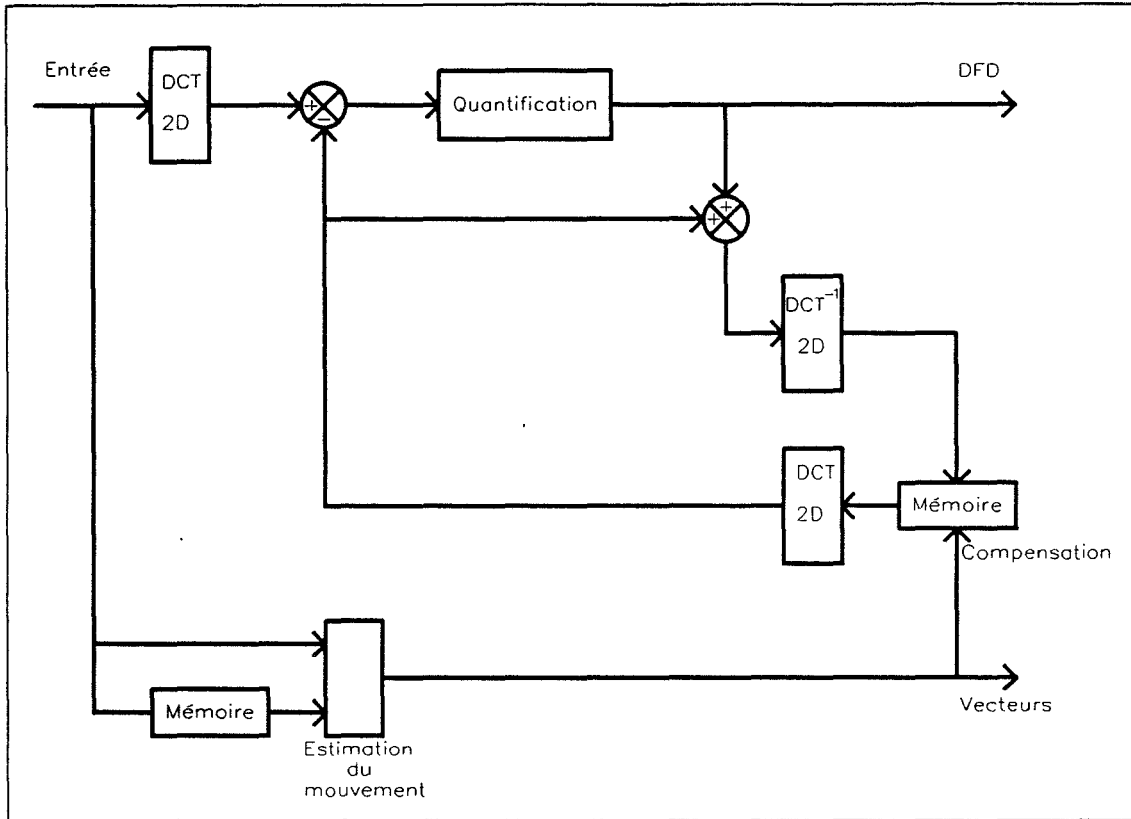


Figure I-9. Exemple de schéma de codage hybride avec compensation du mouvement.

La DFD est calculée en utilisant l'image k-1 codée, décodée et compensée. De cette façon, on se place dans le même contexte qu'à la réception où la DFD sera rajoutée à l'image précédente qui y aura été décodée puis compensée par les vecteurs reçus. La transmission des vecteurs représente tout de même une partie importante du débit total. En les codant par un codage de longueur variable [FAD91], cette partie peut représenter 35 à 50% de ce débit total.

Plusieurs méthodes d'estimation du mouvement ont été développées afin d'améliorer les performances de cette technique [MUS85]. Toutefois, il existe à l'heure actuelle deux principales techniques d'estimation de mouvement communément employées. Ces techniques sont le pel-récuratif et le block matching. Précisons qu'on fait l'hypothèse que les mouvements sont de translation. L'estimation d'autres types de mouvements, bien que possible, [HUA81], [NAG83], [ROA80] fait appel à des techniques beaucoup plus lourdes à mettre en oeuvre. De plus amples informations peuvent être trouvées dans [BOU88], [BOU89], [BOU90], [HOR81] et [LAB83].

I.2.2 - La technique du pel-récuratif

Le principe de base du pel-récuratif est le suivant. Pour un pixel donné d'intensité $I(X,t)$ appartenant à la zone de mouvement de l'image courante et animé d'un mouvement de translation, on essaie de localiser sa position dans l'image précédente en recherchant le pixel d'intensité correspondante $I(X-D,t-\tau)$. X représente la position (i,j) du pixel, D le déplacement et τ le temps séparant les deux images. Ce raisonnement suppose bien entendu que les conditions d'éclairage de l'image ne changent pas d'une image à l'autre et que par conséquent les variations d'intensité des pixels sont le seul fait du mouvement.

Les premiers algorithmes fondés sur ce principe ont vu le jour au cours des années 75 et 76 [CAF76], [LIM75]. De par leur conception, ces algorithmes ne permettaient que l'estimation D' de petits déplacements. En vue d'améliorer la précision de l'estimation et d'augmenter l'amplitude des mouvements mesurables, Netravali et Robbins [NET79] ont proposé un schéma récuratif dans lequel une meilleure estimation D'_{i+1} est obtenue à partir d'une estimation courante D'_i selon la formule de récurrence :

$$D'_{i+1} = D'_i + U_i \quad (1.2)$$

où U_i est le terme correctif à l'itération i .

Supposons qu'un pixel se soit traduit de la position $X-D'_{i-1}$ à l'instant $t-\tau$, à la position X à l'instant t , l'expression de la DFD est donnée localement par la relation :

$$DFD(X, D'_{i-1}) = I(X, t) - I(X-D'_{i-1}, t-\tau) \quad (1.3)$$

Par ailleurs, ce pixel n'ayant pas changé de valeur, on peut écrire :

$$I(X, t) = I(X-D, t-\tau) \quad (1.4)$$

En combinant les équations (1.3) et (1.4), il vient :

$$DFD (X, D'_{i-1}) = I(X-D, t-\tau) - I(X-D'_{i-1}, t-\tau) \quad (1.5)$$

On peut linéariser la DFD [HOR86] en utilisant le développement de Taylor à l'ordre un (en négligeant les infiniment petits d'ordre supérieur) de la fonction $I(X-D, t-\tau)$ en $X-D'_{i-1}$. On obtient alors :

$$DFD (X, D'_{i-1}) = -(D - D'_{i-1}) \cdot \nabla I(X-D'_{i-1}, t-\tau) \quad (1.6)$$

$\nabla I(X-D'_{i-1}, t-\tau)$ représente le gradient spatial de l'intensité dans l'image précédente.

Netravali et Robbins ont proposé, pour avoir un déplacement D'_i à partir de l'estimation précédente D'_{i-1} , un critère d'itération du type algorithme de descente du gradient, minimisant le carré de la DFD, pour lequel on aboutit à la formule suivante :

$$D'_i = D'_{i-1} - \varepsilon DFD(X, D'_{i-1}) \cdot \nabla I(X-D'_{i-1}, t-\tau) \quad (1.7)$$

où ε est une constante positive.

Pour trouver la correspondance avec le pixel recherché, on doit se déplacer dans le sens des I croissant si le terme $DFD(X, D'_{i-1})$ est négatif, et dans le sens des I décroissant si $DFD(X, D'_{i-1})$ est positif.

La figure I-10 illustre la manière dont l'algorithme approche le vrai déplacement D après plusieurs itérations.

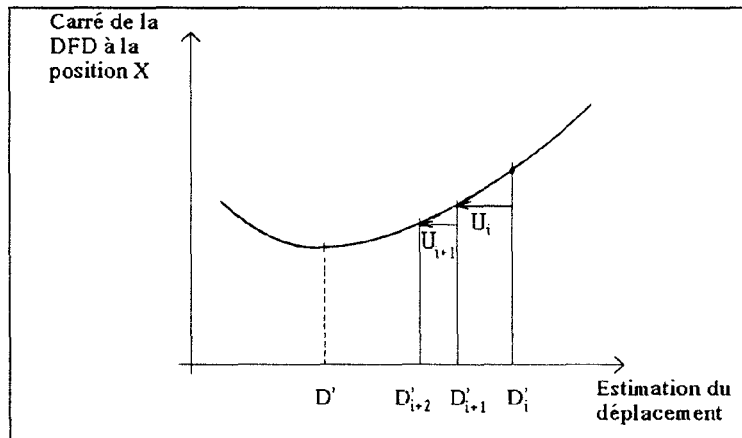


Figure I-10. Illustration du schéma proposé par Netravali et Robbins.

D'après (1.7), le terme correctif U_i , à chaque itération, est proportionnel au gradient ∇I et à la constante ε qui s'avère être un facteur de convergence dont le choix a son importance. Dans le schéma original de Netravali et Robbins, ce facteur vaut $1/1024$. Cette valeur entraîne une lenteur de l'algorithme, qui nécessite alors plusieurs itérations pour aboutir à une estimation satisfaisante du mouvement.

Compte tenu du fait qu'un terme correctif U_i important est nécessaire lorsque $|DFD|$ est grand et $|\nabla I|$ petit et qu'inversement on a besoin d'un U_i faible pour $|DFD|$ petit et $|\nabla I|$ élevé, Walker et Rao [WAL84] ont proposé d'utiliser un facteur de convergence adaptatif variant en fonction du gradient mesuré :

$$\varepsilon = \frac{1}{2} \frac{1}{|\nabla I (X-D'_{i-1}, t-\tau)|^2} \quad (1.8)$$

D'autres chercheurs, Cafforio et Rocca, avaient abouti à une expression similaire [CAF83].

Il existe encore des variantes du pel-récuratif, différant sur l'expression du facteur de convergence ε . La figure I-11 montre un schéma comparatif de l'estimation du mouvement par différents algorithmes [MUS85]. Pour effectuer ces mesures, on utilise des séquences d'images synthétiques dont on fixe l'amplitude des mouvements de translation. Dans le cas présent, on a un déplacement de 5,5 pixels d'une image à l'autre.

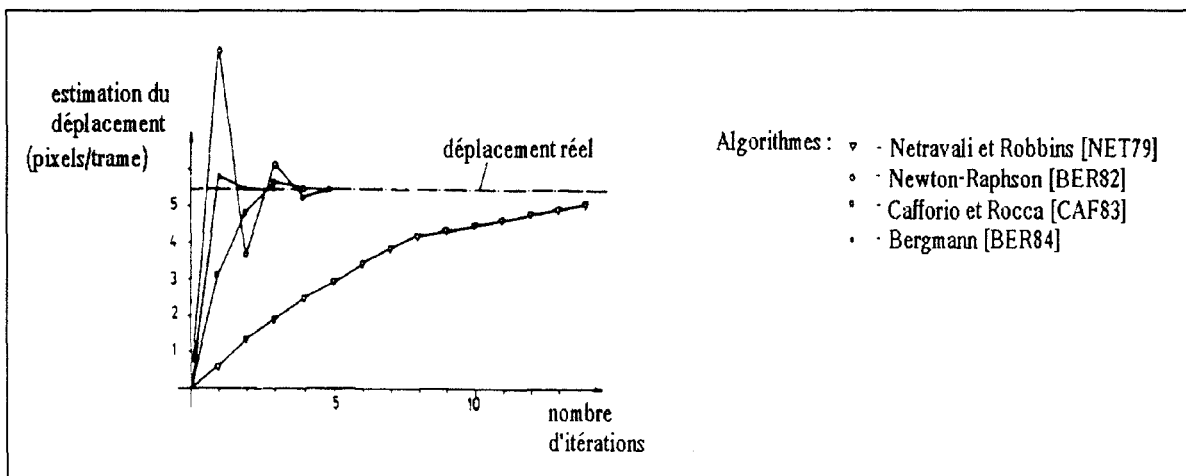


Figure I-11. Estimations comparées du déplacement pour différents algorithmes pel-récuratif.

Plus récemment, des travaux ont abouti à la mise en oeuvre d'une méthode pel-réursive basé sur un filtre de Wiener récursif [BIE87]. La philosophie de cette technique est la suivante. Dans le développement de l'équation (1.5), le reste de la formule de Taylor n'est pas négligé. Ensuite, dans l'expression obtenue, ce reste ainsi que le terme $(D-D'_{i-1})$ sont considérés comme étant des échantillons d'un processus stochastique. La DFD est alors une observation de ce processus et son calcul est fait, non plus pour le seul pixel courant, mais pour un ensemble de N pixels appartenant à son proche voisinage. Pour le cas particulier où $N=1$ (on ne prend que le pixel courant), la formule de récurrence est donnée par l'expression :

$$D'_i = D'_{i-1} - \frac{DFD(X, D'_{i-1}) \cdot \nabla I(X - D'_{i-1}, t - \tau)}{|\nabla I(X - D'_{i-1}, t - \tau)|^2 + \mu} \quad (1.9)$$

Il est intéressant de noter que les termes correctifs U_i utilisés dans les autres méthodes citées ne sont que des cas particuliers de celui-ci. On en déduit que cette technique est une généralisation des précédentes. Une étude comparative a montré son comportement supérieur du point de vue robustesse, stabilité et rapidité de convergence.

Pour être plus complet, nous mentionnerons une méthode pel-réursive abordée sous un autre angle [MOO87]. Cette technique pose comme hypothèse que l'amplitude du mouvement est constante d'une image à l'autre. Par conséquent, si $I(X, t)$ provient de $I(X - D, t - \tau)$, le vecteur mouvement de $I(X - D, t - \tau)$ servira d'initialisation au calcul de celui de $I(X, t)$. Cette méthode est connue sous le nom de PAMT (Projection Along the Movement Trajectory). Elle n'a pas montré d'amélioration sensible par rapport aux autres méthodes évoquées.

Les méthodes de compensation de mouvement utilisant le pel-récursif permettent de déterminer les vecteurs de déplacement aussi bien à l'émission qu'à la réception. De ce fait, aucun vecteur n'a besoin d'être transmis. Nous décrivons brièvement un exemple d'algorithme [BIE87], où la compensation s'effectue le long des lignes de balayage de l'image.

L'estimation trouvée pour le pixel précédent sur une ligne est utilisée pour déterminer la DFD au niveau du pixel courant et c'est cette DFD qui est transmise. Ensuite, une mise à jour de l'estimation est faite en utilisant la formule de récurrence; on obtient ainsi un nouveau vecteur. Ce vecteur sera employé pour de nouveau itérer le processus sur le pixel suivant de la ligne. L'estimation du déplacement a lieu uniquement aux emplacements où les pixels ont changé de valeur. On considère qu'un pixel a changé de valeur lorsque la DFD calculée est supérieure à un seuil S . Ce seuil est fixé à 3 dans la plupart des méthodes mais peut bien sûr avoir une valeur différente ou être adaptatif [NET80b].

On peut résumer cet algorithme comme suit :

Etape 1 : Calcul de la DFD à transmettre :

Utiliser D'_{i-1} du pixel précédent comme initialisation pour le pixel courant. Ce D'_{i-1} sert à calculer la DFD à transmettre ($DFD(X, D'_{i-1})$).

Etape 2 : Calcul du nouveau déplacement $\hat{D}_i$:

(a) - si $|DFD(X, D'_{i-1})| < S$ alors $D'_i = D'_{i-1} \Rightarrow$ pas de mise à jour parce qu'on a utilisé le bon vecteur.

(b) - sinon si $\sum_n |FD(X)| < \sum_n |DFD(X, D'_{i-1})|$ alors

$D'_i = 0 \Rightarrow$ reset.

(c) - sinon utiliser un algorithme pel-récuratif pour avoir un nouveau vecteur D'_i .

Etape 3 : Aller au prochain pixel ayant varié et retour à l'étape 1.

Pour chaque pixel changé, l'émetteur envoie son adresse ainsi que la DFD locale. Le récepteur peut alors reconstituer le pixel en additionnant la DFD à la valeur prédite. Il peut également, en cas de décrochage, recalculer un vecteur conformément à l'étape 2c. L'étape 2b correspond à une réinitialisation du système. En effet, si la différence directe calculée avec quelques points du voisinage du pixel courant est inférieure à la DFD correspondant à la même zone, il est évident qu'un mauvais vecteur a été utilisé. On choisit alors de réinitialiser l'algorithme.

Un autre algorithme peut être trouvé dans [WAL84] où l'on ne transmet plus les adresses des pixels. Mais en revanche on est obligé de transmettre alors intégralement toutes les DFD calculées.

Les performances du pel-récurif du point de vue convergence, dépendent de l'endroit où l'on débute l'estimation. En outre, des mesures récentes ont montré des inconvénients majeurs à l'utilisation de la technique du pel-récurif pour effectuer de la compensation de mouvement dans certains cas [WAN92]. Par exemple, les contours d'objets en mouvement sont mal compensés par cet algorithme. Une des causes de ce phénomène est la prise en compte du fond découvert par le bord de l'objet, dans le calcul du gradient d'un pixel se trouvant justement sur ce contour. En effet, le calcul ne devrait logiquement se faire qu'avec les points appartenant à l'objet. Cela conduit à des aberrations qui faussent totalement les vecteurs de mouvement de ce contour. La sensibilité au bruit de cet algorithme a également été mise en évidence.

En résumé, les techniques pel-récurives, bien qu'estimant le mouvement point par point, doivent être utilisées avec d'autres méthodes complémentaires afin d'aboutir à de bons résultats [BAA91a],[GAI92]. De plus, dans un schéma de compensation visant à faire de la compression, la performance recherchée pour l'estimateur de mouvement se situe surtout au niveau de la minimisation de la DFD, du moins lorsqu'il n'y a pas lieu d'effectuer une interpolation du mouvement à la réception par exemple. Ces raisons, ajoutées à une relative complexité de mise en oeuvre, font que la technique d'estimation par block matching prend souvent le pas sur le pel-récurif.

I.2.3 - La technique du block matching

Au lieu d'estimer le déplacement de manière réursive, on peut calculer la fonction d'intercorrélation normalisée définie par :

$$F(D) = \frac{R_{I_k I_{k-1}}(D)}{\sqrt{R_{I_k I_k}(0) \cdot R_{I_{k-1} I_{k-1}}(0)}} \quad (1.10)$$

Une estimation du déplacement est obtenue en trouvant le maximum (ou pic de corrélation) de cette fonction. Pour déterminer le déplacement d'un point de coordonnées (i,j), un bloc de l'image k de taille MxN pixels centré en (i,j) est corrélé avec des pixels de l'image k-1 dans une zone de recherche limitée pour obtenir la meilleure correspondance, c'est-à-dire le maximum de la fonction F définie par :

$$F(i,j) = \frac{\sum_{m=1}^M \sum_{n=1}^N I_k(m,n) I_{k-1}(m+i,n+j)}{\left[\sum_{m=1}^M \sum_{n=1}^N I_k^2(m,n) \right] \left[\sum_{m=1}^M \sum_{n=1}^N I_{k-1}^2(m+i,n+j) \right]} \quad (1.11)$$

où (i,j) désigne les coordonnées du pixel central du bloc. En supposant un déplacement maximum horizontal ou vertical de dm pixels, la zone de recherche est donnée par :

$$Z_R = (M + 2dm) \times (N + 2dm) \quad (1.12)$$

comme l'indique la figure I-12.

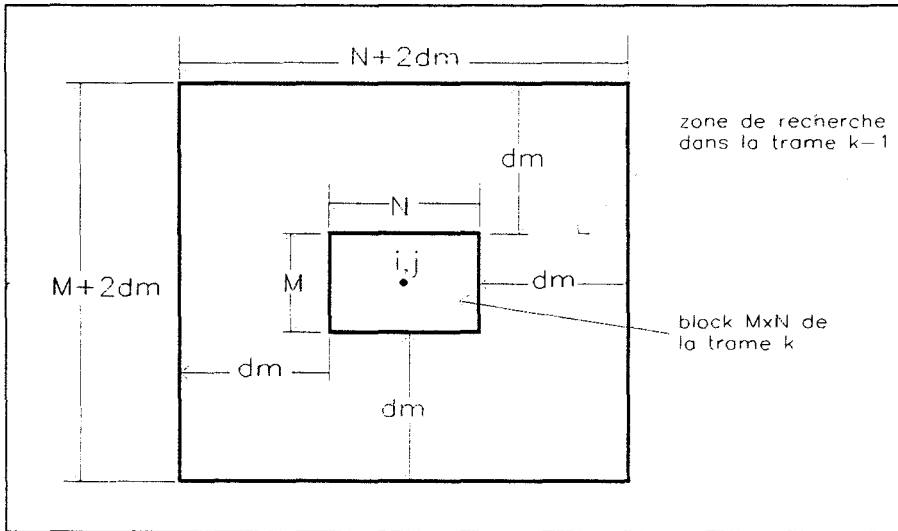


Figure I-12. Zone de recherche de la meilleure corrélation

La recherche du pic de corrélation dans cette zone nécessite $(2dm + 1)^2$ calculs de $F(i,j)$, ce qui est assez lourd en terme d'implantation.

Afin de réduire cette complexité logicielle et travailler en temps réel, l'image est divisée en blocs de taille fixe en faisant l'hypothèse que tous les pixels d'un même bloc sont animés du même mouvement. On calcule ainsi un vecteur de déplacement par bloc. Le critère de correspondance est simplifié et on cherche à minimiser, soit l'erreur quadratique moyenne ou MSE (mean square error), soit la moyenne de la différence directe ou MAD (mean absolute difference).

$$MSE(i,j) = \frac{1}{M \cdot N} \sum_{m=1}^M \sum_{n=1}^N [I_k(m,n) - I_{k-1}(m+i,n+j)]^2 \quad (1.13)$$

$$MAD(i,j) = \frac{1}{M \cdot N} \sum_{m=1}^M \sum_{n=1}^N |I_k(m,n) - I_{k-1}(m+i,n+j)| \quad (1.14)$$

La zone de recherche peut être étendue à toute l'image pour être certain de trouver la bonne correspondance (recherche exhaustive). Cependant, l'intérêt général s'est porté sur des méthodes basées sur des hypothèses permettant de limiter la zone de recherche, pour des raisons de rapidité. Cette technique d'estimation du mouvement a reçu le nom de block matching, et plusieurs variantes ont été développées.

I.2.3.a - Block matching classique

J. Jain et A. Jain en ont été les pionniers [JA181] en utilisant la MSE comme critère. Pour simplifier la procédure de recherche, ils ont proposé une méthode dite 2D-logarithmique, fondée sur l'hypothèse qu'on observe une décroissance monotone de la MSE lorsqu'on s'oriente dans la direction du

mouvement recherché. Cette direction, appelée direction de distorsion minimale, est déterminée par étapes successives. La figure I-13 illustre cette procédure de recherche sur un exemple.

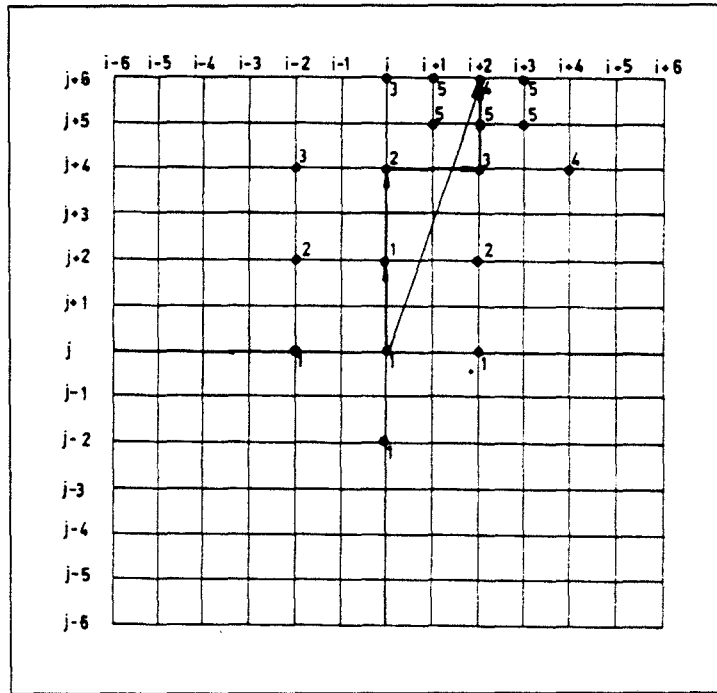


Figure I-13. Exemple de recherche 2D-logarithmique. Un numéro n indique un point de recherche à l'étape n .

A chaque étape, cinq points sont testés. La distance entre les points de recherche est réduite si le minimum se situe au milieu de la fenêtre courante ou sur le bord de la zone de recherche. Dans l'exemple de la figure, cinq étapes ont été nécessaires pour trouver le vecteur déplacement en $(i+2, j+6)$.

Une autre variante du block matching a été proposée à la même époque par d'autres chercheurs [KOG81], utilisant le MAD et une procédure de recherche appelée "menu search", très proche de la méthode précédente. Un exemple apparaît sur la figure I-14. Les numéros ont la même signification que sur la figure I-13.

Au départ, en plus du point central, huit autres points sont testés en prenant un espacement assez grossier comme le montre la figure. Dans la seconde étape, huit nouveaux points sont pris autour de celui obtenu lors de la première étape, en réduisant l'espacement entre eux. On itère le procédé jusqu'à atteindre la précision voulue. Dans cet exemple, la troisième étape a donné le vecteur final $(i+2, j+6)$.

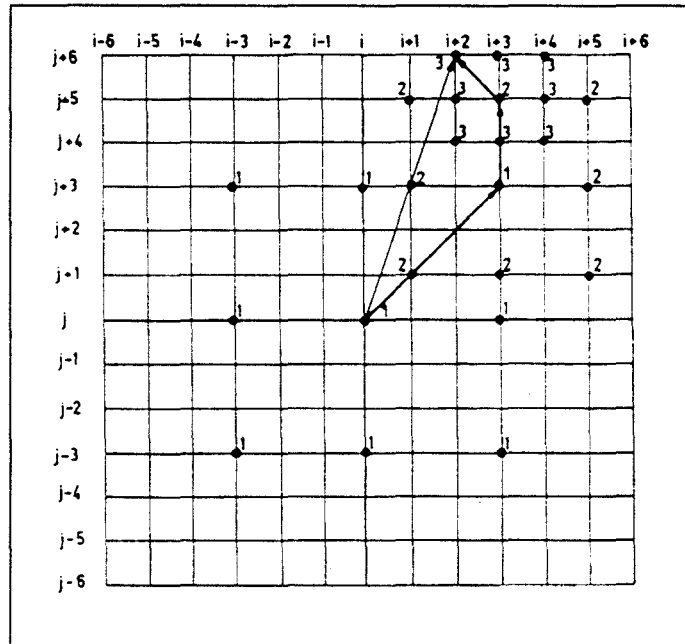


Figure I-14. Exemple de " menu search ".

La troisième variante de block matching classique que nous citerons est celle de Srinivasan et Rao [SRI84] appelée méthode conjuguée (conjugate direction search). Nous décrivons cet algorithme à partir de l'exemple de la figure I-15. La direction de distorsion minimale est recherchée en utilisant le MAD.

Dans un premier temps, le minimum dans la direction i est déterminé en calculant $MAD(i-1, j)$, $MAD(i, j)$, et $MAD(i+1, j)$. $MAD(i+1, j)$ étant le plus faible, on calcule $MAD(i+2, j)$ et de nouveau on prend la plus petite valeur entre $MAD(i, j)$, $MAD(i+1, j)$ et $MAD(i+2, j)$. En procédant de cette manière, le minimum dans la direction i est localisé lorsqu'une valeur est

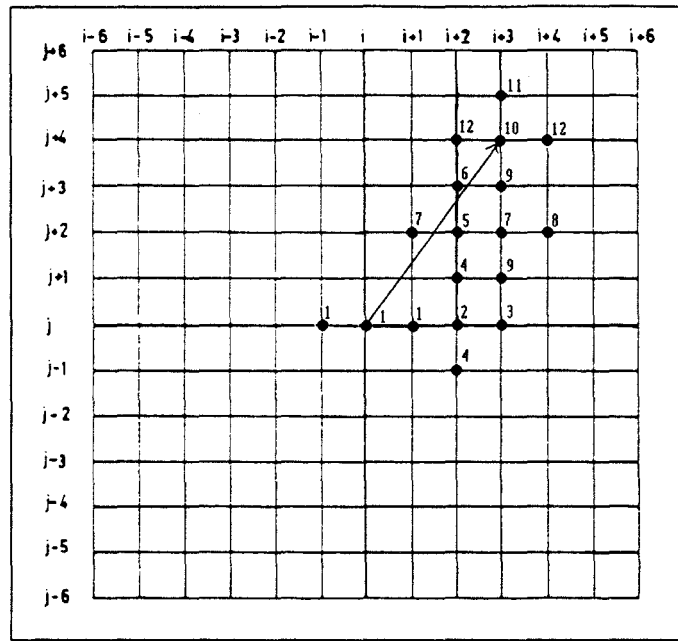


Figure I-15. Exemple de recherche conjuguée.

encadrée par deux valeurs de MAD qui lui sont supérieures. On commence ensuite la recherche dans la direction j selon le même principe, en partant du minimum trouvé précédemment, et ainsi de suite jusqu'à ce que le MAD ait un minimum dans les deux directions. Dans notre exemple, le vecteur final est $(i+3, j+4)$.

Ces différentes procédures de recherche, bien que rapides, peuvent conduire à des minima locaux (MAD ou MSE) qui ne sont pas optimaux, d'une part parce que la recherche se fait dans une zone limitée, d'autre part parce que l'algorithme de recherche peut également mener à un faux minimum local. Ces risques sont somme toute minimales, et ces techniques de recherche sont largement utilisées car compatibles avec les besoins en temps réel. Cependant, certains concepteurs, prenant en compte la simplicité de mise en oeuvre de la recherche exhaustive, ainsi que la certitude de trouver le vrai minimum, l'ont retenu pour la fabrication de circuits spécialisés [CON89], [PAR90].

I.2.3.b - Block matching utilisant des tailles variables

Dans la technique du block matching, la correspondance obtenue est optimum seulement au sens du minimum de la MSE ou du MAD et peut très bien ne pas correspondre au mouvement puisqu'il s'agit de trouver deux pavés les plus semblables possible. La fiabilité de l'estimation dépend en fait du choix de la taille des blocs. Le résultat peut être faussé si on utilise un petit bloc pour mesurer un grand déplacement. En effet, plus le bloc sera petit, plus on aura de chance de trouver un bloc correspondant dans la deuxième image (au sens du critère retenu) même s'il n'y a aucun rapport entre les deux du point de vue du mouvement. Par conséquent, l'évaluation d'un mouvement de grande amplitude nécessite l'utilisation d'un bloc suffisamment grand.

Cependant, l'estimation va s'avérer erronée si à l'intérieur de ce bloc il y a plusieurs mouvements différents en amplitude et en direction. Il est donc nécessaire, pour aboutir à une mesure plus précise, d'avoir recours à des blocs plus petits.

Afin de pouvoir prendre en compte ces deux besoins antagonistes, des méthodes utilisant des blocs de taille variable ont été proposées [BIE88], [CHA90], [KER90].

Une première méthode que l'on peut qualifier de "conditionnelle", a été développée par Chan, Yu et Constantinides [CHA90]. Dans un premier temps, une estimation du mouvement est faite avec de grands blocs. Si la compensation obtenue est satisfaisante, on s'en tient là, sinon les blocs sont subdivisés en d'autres plus petits et ainsi de suite.

Une autre technique, présentée par Bierling [BIE88], adopte une démarche plus hiérarchique. On exécute n niveaux de recherche pour obtenir un vecteur, somme de n vecteurs. Au premier niveau, une estimation est effectuée avec un grand bloc. Au niveau suivant, on réduit la taille du bloc et on se sert de l'estimation du niveau précédent comme initialisation. On calcule ainsi une estimation pour les n niveaux, et le vecteur final est la somme de ces n vecteurs.

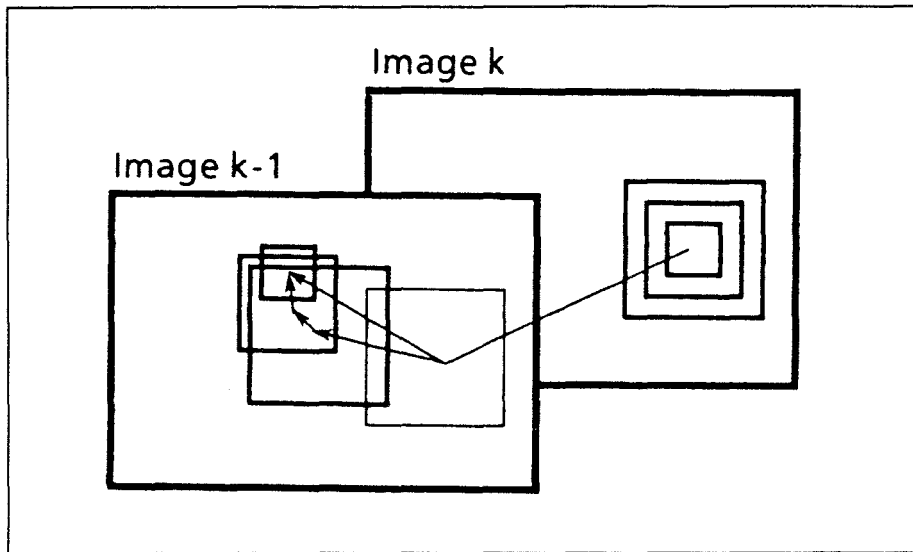


Figure I-16. Principe du block matching hiérarchique sur trois niveaux. Le vecteur final est la somme de trois vecteurs.

Le premier niveau permet d'obtenir le mouvement d'ensemble du bloc, tandis que le dernier autorise la prise en compte des mouvements de moindre importance qui y sont contenus. L'exemple de la figure I-16, avec trois niveaux hiérarchiques, explique ce principe.

Les résultats produits par cette méthode hiérarchique sont meilleurs que ceux obtenus par le block matching classique.

Il est à noter qu'intrinsèquement ces méthodes de block matching hiérarchiques font référence à une notion de résolution variable dans l'image. Il paraît donc naturel de les associer à un mode de représentation des images basé sur ce concept de résolution. C'est ce que permettent aujourd'hui les techniques de l'analyse multirésolution.

CHAPITRE II :

L'ANALYSE MULTIRESOLUTION

L'ANALYSE MULTIRÉSOLUTION

Une façon peut-être indirecte d'introduire le concept d'analyse multirésolution est de s'intéresser au fonctionnement de l'oeil humain, tout au moins dans la partie vision primaire. Cette approche permet de mettre en évidence l'intérêt du concept de multirésolution en tant qu'essai de modélisation de la perception des images. En ce sens, il s'agit bien plus que d'une simple technique de traitement.

II.1 - Le mécanisme de la vision

II.1.2 - Le système nerveux central

Le système visuel fait partie du système nerveux, qui est le réseau de communication chez l'homme, commandé par le cerveau. Les informations sont acheminées à travers ce réseau, par des cellules nerveuses appelées neurones. Un neurone est constitué d'un corps cellulaire auquel sont rattachées une fibre principale appelée axone, et des fibres secondaires, les dendrites. La figure II-1 montre des exemples de neurones [KUF76].

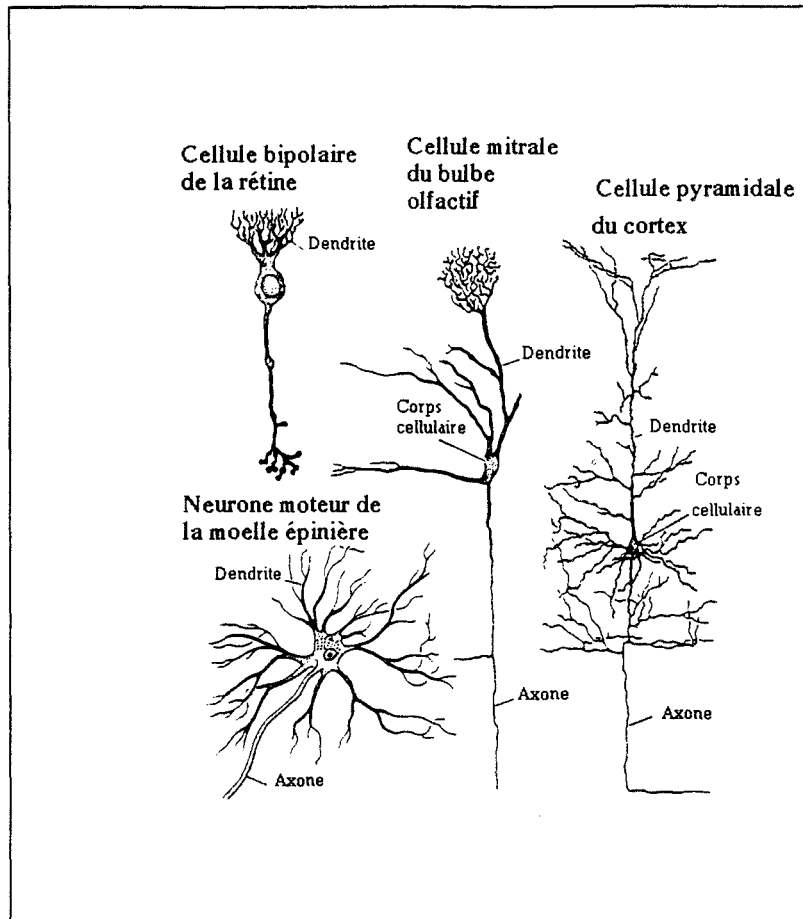


Figure II-1. Quelques exemples de neurones spécialisés.

L'information, ou influx nerveux, circule toujours des dendrites vers l'axone. La région de contact entre deux neurones est une synapse. Un neurone peut avoir deux types de comportement : excitateur ou inhibiteur. Dans le premier cas, il transmet l'influx nerveux qu'il reçoit, et dans le second, il l'inhibe. L'étude du système nerveux est facilité par les propriétés suivantes :

- Il existe uniquement deux types de signaux qui sont véhiculés : des signaux longues distances et d'autres pour les faibles parcours.

- Quelle que soit la nature de l'information (visuelle, sonore, tactile, ...), les signaux sont identiques dans presque tous les neurones. En outre, les formes de ces signaux ne changent pas d'une espèce à l'autre, ce qui permet de pouvoir prendre par exemple un chat comme cobaye.

Le cerveau reçoit donc des informations qui sont la traduction d'événements extérieurs. Au niveau d'un neurone donné, le signal reçu est un train d'impulsion dont la fréquence est proportionnelle à l'intensité de l'excitation. Le mode de communication au sein du système nerveux est par conséquent la modulation de fréquence. Le cerveau distingue deux signaux identiques qu'il reçoit, par les chemins empruntés par ceux-ci, c'est-à-dire le "câblage" des neurones. Voyons alors quel est celui du système visuel.

II.1.3 - L'œil

C'est le capteur des signaux visuels. Du point de vue de son anatomie et de son fonctionnement, on peut établir un parallèle avec un appareil photographique comme le montre la figure II-2.

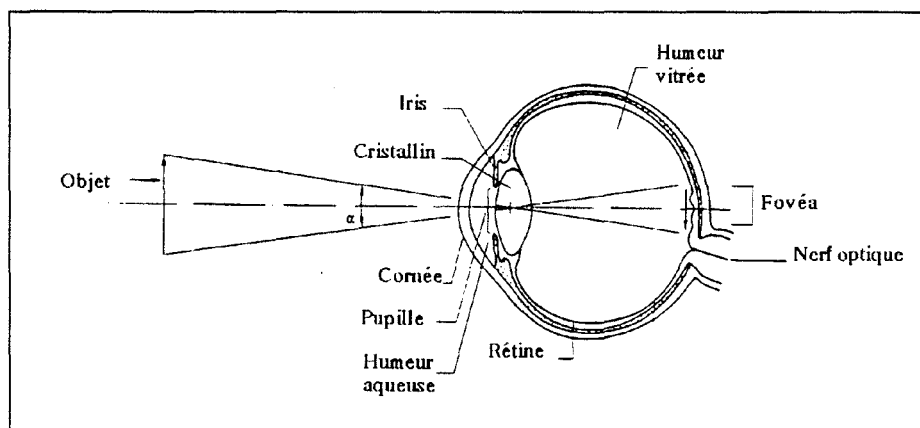


Figure II-2. Coupe schématique de l'œil.

L'objet est focalisé sur la rétine qui l'analyse et envoie l'information au cerveau via le nerf optique et tout un circuit spécifique. Après traversée de la cornée et de l'humeur aqueuse, la lumière pénètre dans la "chambre" de l'oeil par la pupille qui va jouer le rôle d'un diaphragme, réglant la quantité de lumière pour éviter tout éblouissement. L'influence de la pupille est modélisée par un filtre passe-bas dont la fréquence de coupure est inversement proportionnelle au diamètre de cette ouverture. Le cristallin, à la manière d'une lentille, focalise l'image sur la rétine, dans une zone appelée fovéa, où la vision est la plus nette. Cependant, même chez une personne ayant une vue correcte, ce cristallin présente de petites imperfections, sources du phénomène d'aberration sphérique. Ce défaut peut également être représenté par un filtrage passe bas à deux dimensions.

II.1.4 - La rétine

C'est la partie neurosensitive de l'oeil. Elle transforme la lumière reçue en un signal électrique transmis au cortex visuel par l'intermédiaire du nerf optique. La figure II-3 indique l'itinéraire suivi par l'information dans le système visuel humain.

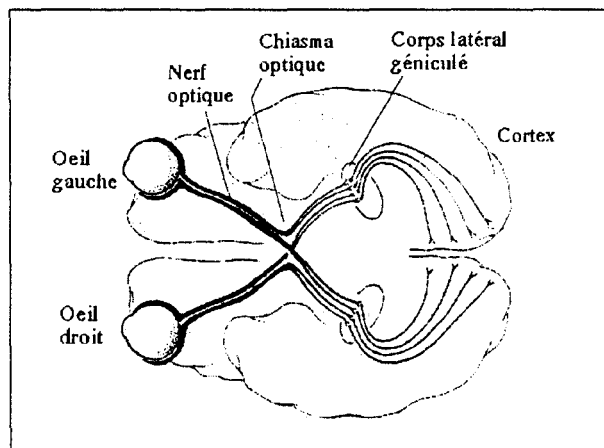


Figure II-3. Eléments du système visuel humain.

La rétine se compose de plusieurs types de cellules disposées en couches comme le montre la figure II-4 [COR70].

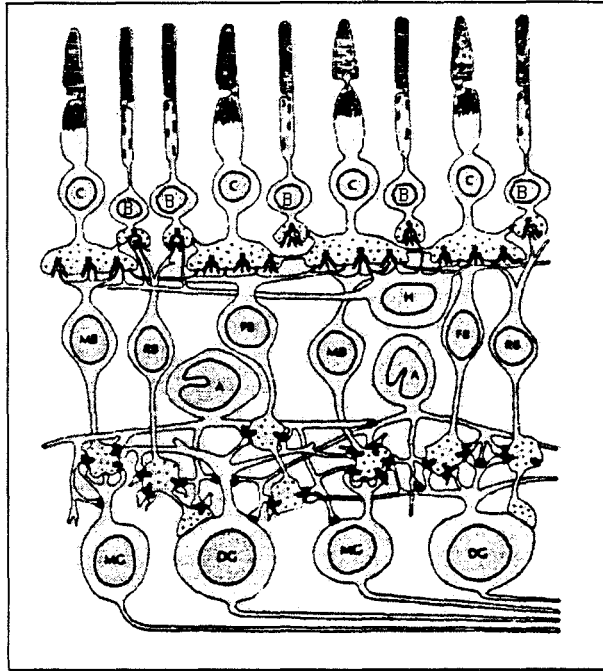


Figure II-4. Coupe de la rétine.

La couche supérieure comporte deux sortes de cellules photosensibles différentes : les cônes et les bâtonnets. Les cônes ont besoin de la lumière du jour. Ils détectent les couleurs, et permettent de distinguer les détails (photopic or bright-light vision), ceci essentiellement parce que chaque cône possède sa propre terminaison nerveuse. Par contre, plusieurs bâtonnets sont reliés à un même nerf. Ils ont par conséquent peu d'influence sur la résolution de l'oeil. Ils sont sensibles aux formes et ne nécessitent pas un éclairage important. Leur rôle sera prépondérant dans la vision nocturne. Les objets qui, par exemple, apparaissent colorés à la lumière diurne, se transforment, en vision nocturne, en masses difformes dépourvues de couleur parce que seuls les bâtonnets sont excités (scotopic or dim-light vision).

La répartition de ces deux types de cellules n'est pas uniforme sur toute la rétine. On a notamment une plus grande concentration de cônes au niveau de la fovéa. C'est la raison pour laquelle la vision est nette lorsque l'image est formée dans cette zone.

Ces cellules photosensibles sont à l'origine de la transformation de la lumière en signaux électriques. Puisqu'une image numérique est un ensemble de points lumineux, l'aptitude de l'oeil à distinguer différents niveaux d'intensité est un paramètre important. La plage de luminosité à laquelle l'oeil peut s'adapter est très grande. Du seuil de perception (seuil bas de la vision "scotopique") à celui de l'éblouissement, on a un rapport de l'ordre de 10^{10} . Les cellules photosensibles, en opérant la conversion, compriment cette plage de variation selon une loi logarithmique (loi de Weber-Fechner).

Sous les cônes et les bâtonnets, on trouve les cellules bipolaires qui elles-même sont reliées aux cellules ganglionnaires, situées encore plus en profondeur. Le faisceau formé par les axones de ces cellules ganglionnaires constitue le nerf optique.

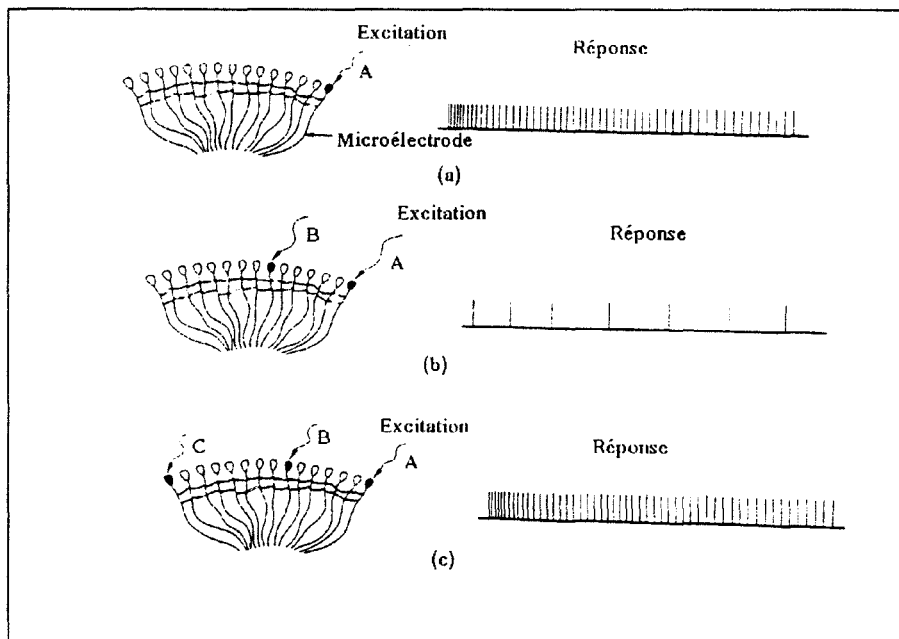


Figure II-5. Illustration du phénomène d'inhibition latérale.

Sur la figure II-4, on peut voir qu'il existe des liaisons série et parallèle, formant ainsi quelques boucles de réaction. Ces connections vont provoquer l'apparition d'un phénomène connu sous le nom d'inhibition latérale, résumé par la figure II-5.

L'excitation d'une cellule A produit un train d'impulsions dont la fréquence est proportionnelle à l'intensité du stimulus (II.5a). Si une cellule voisine B est également excitée, elle altère et peut inhiber la réponse de A (II.5b). L'action de B sur A peut se trouver annulée en excitant une cellule C suffisamment voisine de B, mais assez loin de A pour ne pas l'influencer (II.5c). En résumé, ce niveau du système visuel peut être modélisé linéairement par un filtre de bande passante limitée comme l'indique la courbe représentée sur la figure II-6 [KUN85].

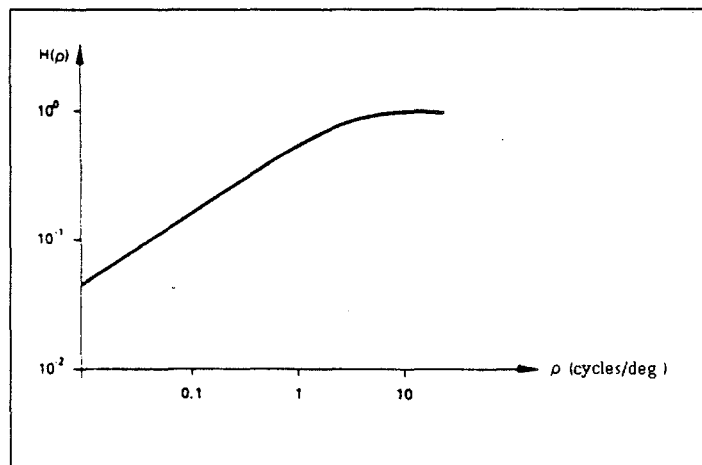


Figure II-6. Réponse (radiale) en fréquence des cellules de la rétine.

La limite de 10 cycles par degré correspond à la distance minimale en dessous de laquelle deux excitations voisines sont confondues.

II.1.5 - Le cortex cérébral

Situé à l'arrière du cerveau, c'est le revêtement superficiel des hémisphères cérébraux. Il est considéré comme étant l'instance supérieure du système nerveux central. Il est divisé en plusieurs régions spécialisées dont l'une est le point d'arrivée des informations en provenance des yeux, que nous appellerons cortex visuel.

L'analyse cellulaire du cortex visuel fait apparaître une disposition en couches hiérarchiques, d'un certain nombre de neurones différents. On distingue par ordre croissant dans la hiérarchie : les cellules simples, les cellules complexes, les cellules hypercomplexes et les cellules hypercomplexes d'ordre supérieur.

C'est à ce niveau du système visuel que l'analyse de l'image est affinée quant à la sensibilité aux positions, directions et contours.

En conclusion, les caractéristiques du système que nous venons de décrire sont résumées par la figure II-7 [KUN85].

Le premier bloc est un filtre spatial passe-bas isotrope, représentant l'aberration sphérique et l'effet de la pupille. On a ensuite la caractéristique de conversion des cellules photosensibles. Elle peut être de type logarithmique ou en x^α . Le bloc suivant est un filtre isotrope passe-haut rendant compte du phénomène d'inhibition latérale. Enfin, on trouve un banc de filtres directionnels, équivalant au traitement opéré au niveau inférieur du cortex visuel. Notons au passage que si l'on translate un peu l'image (c'est-à-dire si on l'écarte de la fovéa), les caractéristiques de ce système changent en raison de la répartition inhomogène des cellules photosensibles, ce qui va entraîner une dégradation de la résolution.

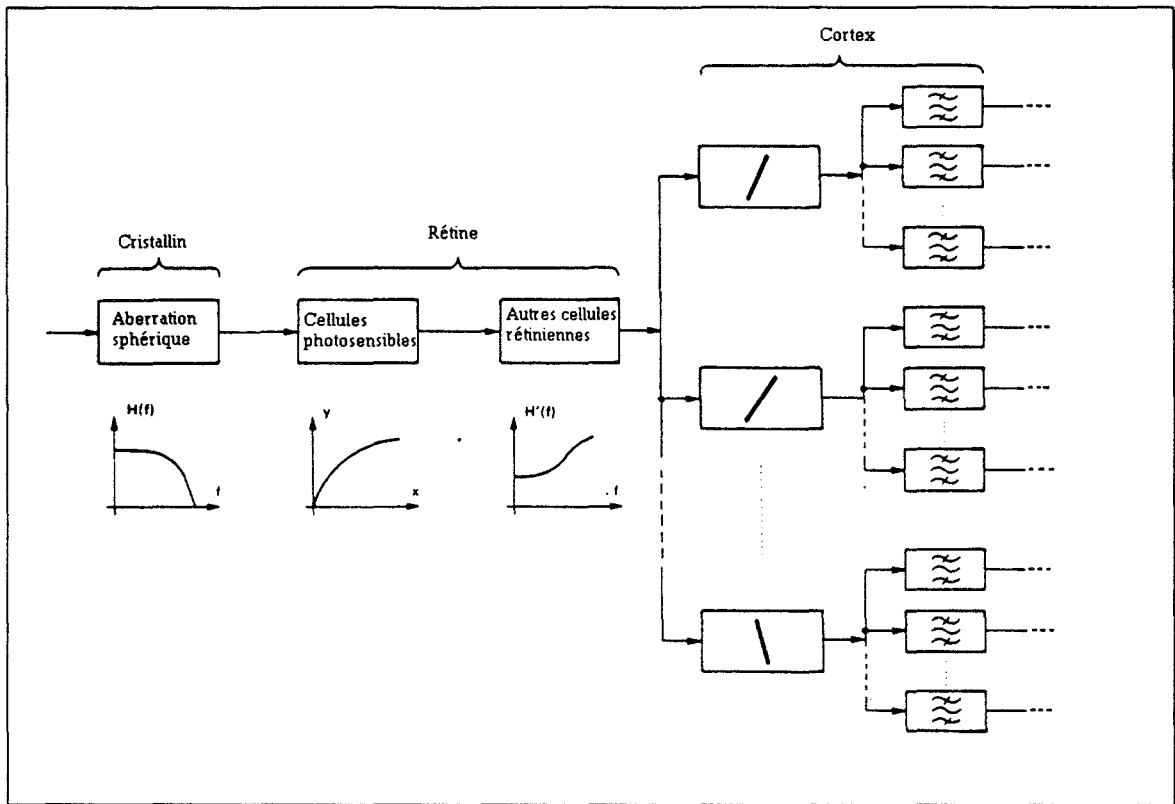


Figure II-7. Schéma bloc équivalent du système visuel humain.

Pour simplifier la représentation, le schéma de la figure II-7 peut être réduit à celui de la figure II-8 [MAL89b].

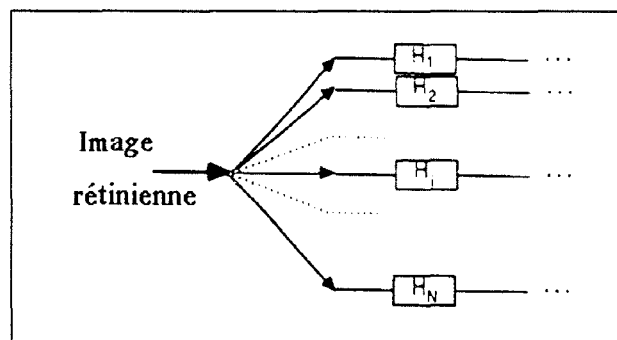


Figure II-8. Modèle multicanaux de la vision humaine.

Les filtres H_i sont des passe-bandes indépendants, ayant approximativement la même largeur de bande sur une échelle logarithmique. En effet, compte tenu de la largeur de la dynamique de luminosité perceptible dont nous avons parlé en II.4, le système visuel ne peut pas réaliser une adaptation simultanément sur toute cette gamme. Il accomplit alors une adaptation à la luminosité sur des plages plus réduites dont rendent compte ces filtres. En outre, conformément à ce que nous avons dit dans le paragraphe II.5, ces filtres sont sensibles à l'orientation spatiale.

Ce modèle, ainsi que les éléments que nous avons mentionnés dans cette étude de l'oeil, ont permis aux chercheurs en traitement d'images de développer des algorithmes s'inspirant de son mode de fonctionnement. C'est le cas de la technique du SHS que nous citons dans le chapitre I. En anticipant sur la suite du chapitre, on peut dire que la vision de l'oeil est "multirésolution", car le modèle obtenu correspond à l'approche envisagée pour l'étude des images dans l'analyse multirésolution.

La partie du système visuel dont rend compte ce modèle est communément appelée "vision bas niveau". La suite du mécanisme de la vision, matérialisée par les pointillés, est un domaine encore mal connu et toujours à l'étude par les spécialistes. Des informations complémentaires sont données dans [ALE82], [GON77], [MAR82], [NET88], [PEA75] et [ULL79].

II.2 - Notions d'échelle et de résolution

Soit une fonction $f(t)$. Un changement d'échelle d'un facteur α se traduit par la transformation :

$$f(t) \rightarrow f(\alpha t) \text{ avec } \alpha > 0$$

Si $\alpha > 1$, le signal est contracté sur l'axe des temps, et si $\alpha < 1$, il est dilaté.

La notion d'échelle est donc relative à la dimension temporelle du signal, ou spatiale dans le cas d'une image.

Un changement d'échelle dans le domaine temporel entraîne également une variation dans celui des fréquences mais en sens inverse. Si l'on note $F(f)$ la transformée de Fourier de $f(t)$ on a :

$$\text{TF} \{f(\alpha t)\} = \frac{1}{|\alpha|} \cdot F\left[\frac{f}{\alpha}\right] \quad (2.1)$$

Au vu de cette propriété, si l'on veut faire en sorte que l'énergie soit conservée au cours de la transformation, on définira plutôt un changement d'échelle par la relation suivante :

$$f(t) \rightarrow \sqrt{\alpha} \cdot f(\alpha t)$$

Quant au terme résolution, il correspond plutôt à la quantité d'information présente et est aussi lié à l'énergie du signal. Par abus de langage on dira que c'est le "contenu spectral" du signal.

Si par exemple on opère un filtrage passe-bas sur un signal, tronquant par conséquent sa bande de fréquence, on modifie alors sa résolution et aussi son énergie. L'échelle et la résolution sont donc deux notions différentes. D'ailleurs dans l'exemple ci-dessus, la résolution du signal est affectée mais pas son échelle.

Inversement, le changement d'échelle, dans le cas d'un signal continu, ne modifie pas sa résolution.

Lorsqu'on considère par contre un signal discret, ce qui est le cas d'une image numérique, la remarque précédente n'est plus vraie.

II.3 - Conséquences de la discrétisation

Rappelons que la discrétisation d'un signal de fréquence maximale F , s'effectue par un échantillonnage à la cadence F_e . La condition de Shannon, nécessaire pour pouvoir recouvrer le signal de départ, impose que F_e soit supérieur ou égal à $2F$.

Soient T la période du signal et T_e le temps séparant deux échantillons successifs. On a la relation : $T = N \cdot T_e$.

En considérant le signal sur une période on disposera de N échantillons. On peut relier l'échelle au nombre d'échantillons par unité de temps (pour un signal fonction du temps).

Envisageons le cas d'un signal $x(t)$, de fréquence maximale F , échantillonné à la fréquence limite de Shannon, dont la représentation temporelle et l'encombrement spectral de sa transformée de Fourier sont données par la figure II-3.

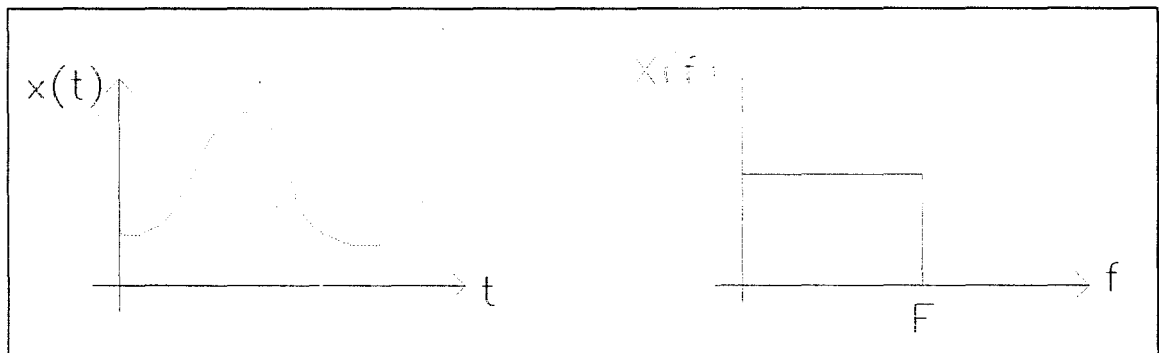


Figure II-3. Signal $x(t)$ échantillonné et encombrement spectral de sa transformée de Fourier.

En effectuant un changement d'échelle d'un facteur α ($\alpha > 1$) on obtient :

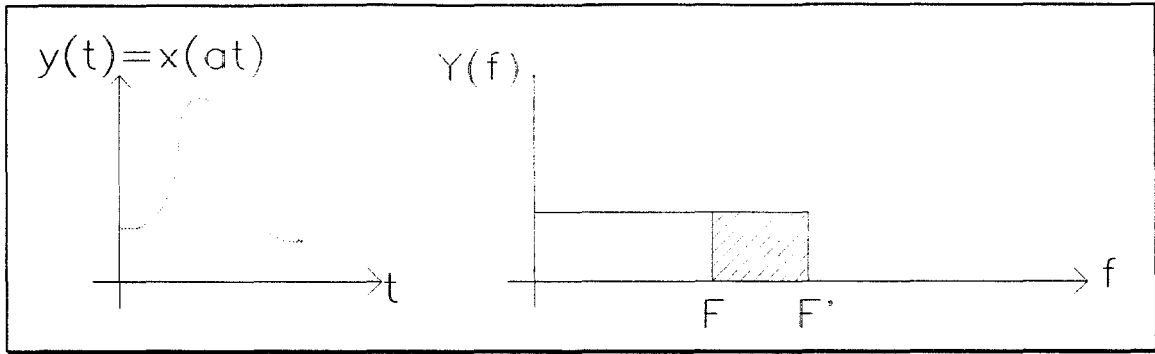


Figure II-4. Effet du changement d'échelle sur $x(t)$.

La fréquence d'échantillonnage du système étant fixée, et compte tenu de la condition de Shannon, la partie hachurée du spectre fréquentiel est perdue, d'où une perte de résolution. Nous constatons donc que dans le cas d'un signal discret, le changement d'échelle entraîne automatiquement une modification de la résolution.

Dans cet exemple, pour pouvoir opérer la transformation inverse et récupérer le signal initial, il faudrait faire suivre le changement d'échelle d'un

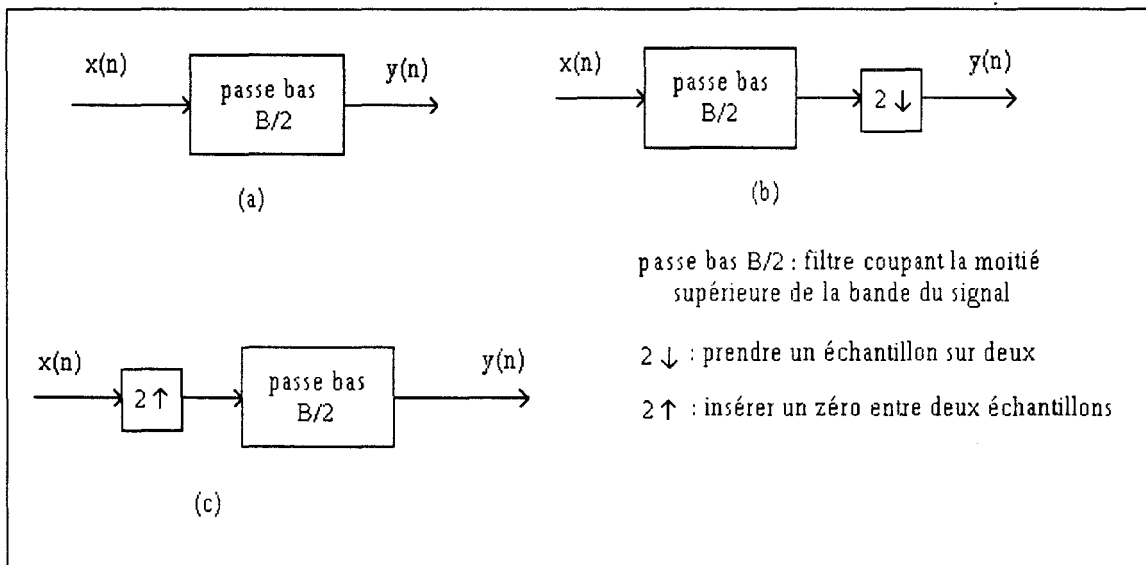


Figure II-5. Illustration du changement d'échelle et de résolution dans le cas d'un signal discret. (a) - la résolution est diminuée de moitié, l'échelle est inchangée. (b) - la résolution est diminuée de moitié, l'échelle est doublée. (c) - la résolution est inchangée, l'échelle est divisée par deux.

sous-échantillonnage à la fréquence $2F'$.

La figure II-5 illustre le changement d'échelle et de résolution dans le cas d'un signal discret [RIO91].

II.4 - Rôle de l'échelle dans la perception visuelle

L'analyse monodimensionnelle précédente se généralise bien sûr à deux dimensions pour s'appliquer aux images.

Une image est constituée par la juxtaposition de points élémentaires - les pixels - auxquels on affecte une valeur d'intensité lumineuse notée $I(x,y)$ pour le pixel de coordonnées cartésiennes x et y .

Considérons le cas d'images de texture [BUR82].

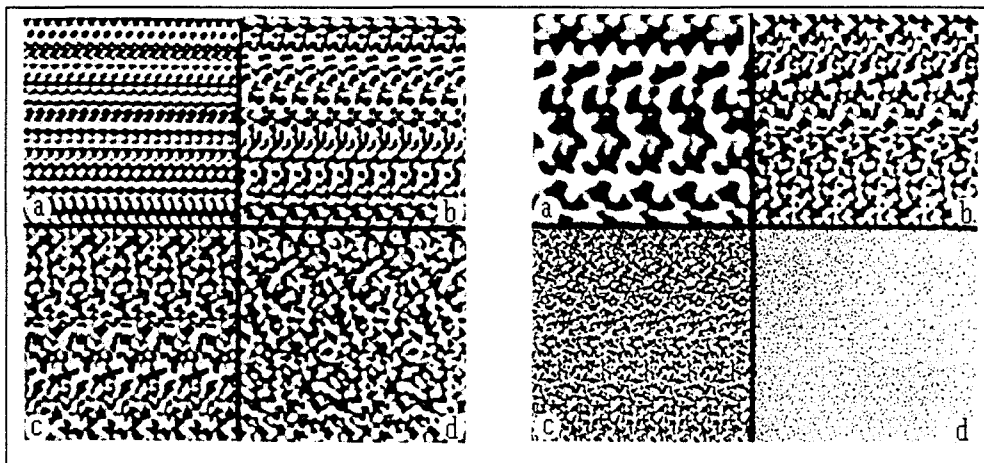


Figure II-6. Deux exemples de textures.

De telles images peuvent être analysées à deux échelles différentes, microscopique et macroscopique.

Au niveau microscopique, on considère l'image comme étant formée

de "grains" élémentaires répartis d'une certaine façon. A l'échelle macroscopique, on la caractérise plutôt par la répétition de motifs spécifiques formés par ces grains élémentaires. En d'autres termes, dans le premier cas, l'entité élémentaire est le grain, tandis que dans le second, c'est un ensemble de grains (motif).

Pour distinguer une texture d'une autre il faut donc mesurer des paramètres caractéristiques à ces deux niveaux. Cela suppose que le système puisse opérer à des échelles différentes. Comme en outre ces échelles, pour des images données, ne sont pas connues d'avance, on doit pouvoir effectuer l'étude à plusieurs échelles différentes.

Un exemple du rôle de l'échelle dans la perception d'une texture est illustrée par la figure II-7.

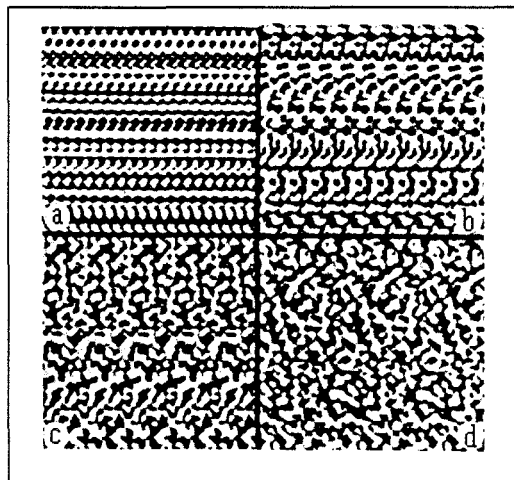


Figure II-7. Effet du doublement de la période de répétition du motif d'une texture.

De l'image (a) à l'image (d), on fait à chaque fois doubler la période de répétition λ_p du motif caractéristique, sans changer la finesse du grain élémentaire. On s'aperçoit alors qu'il existe un λ_p maximum (λ_{pmax}) au delà duquel on ne distingue plus la périodicité du motif. Ce λ_{pmax} , quantifié par Tyler [TYL77] et correspondant à une échelle donnée, peut être considéré comme un paramètre essentiel de la vision humaine, qui marque la limite de perception correcte pour une texture donnée.

Cependant, Tyler a également constaté que si un changement d'échelle s'opérait sur l'ensemble de l'image (figure II-7a), avant de dilater λ_p , λ_{pmax} changeait dans le même rapport. Autrement dit, changer la distance d'observation de la figure II-7, c'est-à-dire l'échelle de l'image sur la rétine de l'oeil, n'affecte pas l'aptitude à percevoir la périodicité de la texture. Ainsi, λ_{pmax} , plutôt que de caractériser la vision humaine, doit être considéré comme un facteur distinctif de la texture. Cela nous amène à la conclusion suivante : la perception de la texture ne dépend pas de l'échelle choisie pour la représenter, ou, ce qui revient au même, la distance d'observation (dans une limite raisonnable) n'influe pas sur la perception de la périodicité.

Ainsi énoncée, cette conséquence du changement d'échelle peut paraître banale. Cela traduit simplement le fait que la manière dont nous percevons les objets ne change pas lorsqu'on s'en rapproche ou on s'en éloigne. Néanmoins, cette caractéristique mérite d'être soulignée car elle n'est pas commune à la plupart des traitements utilisés en analyse d'images.

En outre, nous avons vu que dans le cas discret, le changement d'échelle entraînait obligatoirement un changement de résolution. Ce changement de résolution va bien sûr affecter la représentation de l'objet. Le problème posé consiste donc à trouver un algorithme d'analyse tel que la représentation de l'image ne soit pas modifiée par un changement d'échelle.

II.5 - Construction d'une décomposition multirésolution invariante par changement d'échelle

On cherche à obtenir à partir d'une image origine, une succession de "sous-images", approximations de celle de départ, à des résolutions différentes. On définit donc arbitrairement une suite de niveaux de résolution

que l'on notera $(r_j)_{j \in \mathbb{Z}}$. En se donnant un opérateur $A_{n,m}$ permettant de passer du niveau de résolution n au niveau m , on peut alors obtenir la décomposition selon l'algorithme de la figure II-8. L'image origine est considérée comme ayant la résolution maximale. Par convention ce niveau est noté r_0 , ainsi la résolution croît avec l'indice j .

Dans ce concept d'analyse, on considère qu'une image est un ensemble de détails visibles ou non selon le niveau de résolution. On définit les détails à un niveau r_j comme étant la différence entre I_j et I_{j-1} .

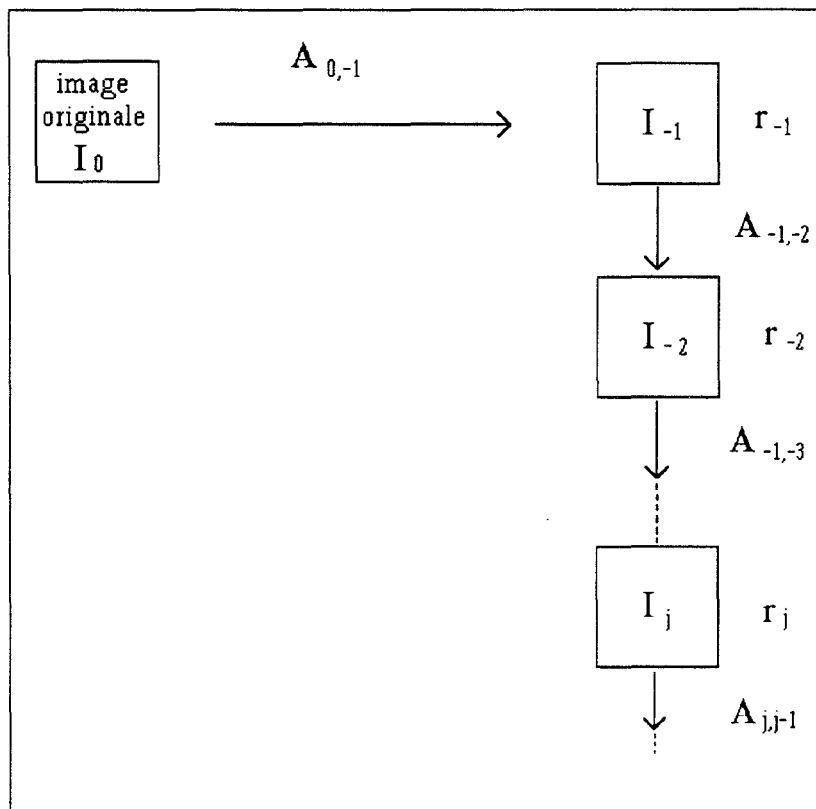


Figure II-8. Algorithme de décomposition multirésolution.

Par définition, l'ensemble des $(I_j)_{j \in \mathbb{Z}^*}$ est appelé représentation multirésolution de l'image origine.

Précisons maintenant ce qu'on entend par invariance par changement

d'échelle. Prenons une image origine quelconque et faisons lui subir un changement d'échelle avant de la décomposer. La représentation ainsi obtenue doit être semblable à celle obtenue sans changement d'échelle. Si cette identité est vérifiée, on dit que la représentation est invariante par changement d'échelle. Voyons alors comment on peut construire une telle décomposition.

En se référant à la définition de la résolution, on en déduit que pour construire les différentes approximations de l'image d'origine, l'opérateur A peut être un filtre passe-bas (PB), dont la bande passante décroît avec la résolution. Prenons comme exemple un signal monodimensionnel d'étendue spectrale initiale B_0 , et appliquons l'algorithme de la figure II-8 en représentant l'encombrement spectral du signal à chaque niveau de résolution (figure II-9).

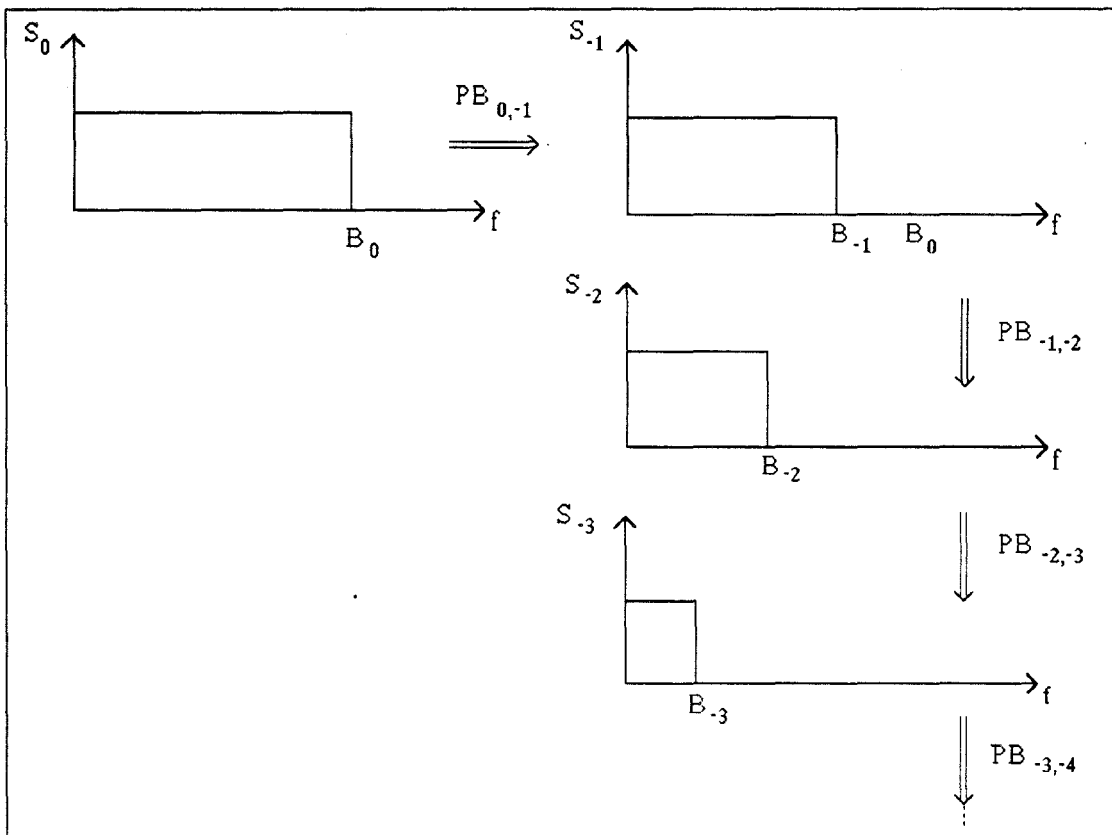


Figure II-9. Equivalence spectrale de l'algorithme de la figure II-8.

On doit alors résoudre le problème suivant : à quelle condition un changement d'échelle d'un facteur α de I_0 n'affecte pas cette décomposition? Nous allons raisonner avec l'exemple $\alpha = \frac{1}{2}$ afin de mettre en évidence le ou les critères.

Comme nous l'avons vu précédemment, un changement d'échelle d'un facteur α dans l'espace temporel entraîne une variation en $\frac{1}{\alpha}$ en fréquence. Dans notre cas, le signal temporel est dilaté d'un facteur 2, d'où une contraction dans le même rapport en fréquence.

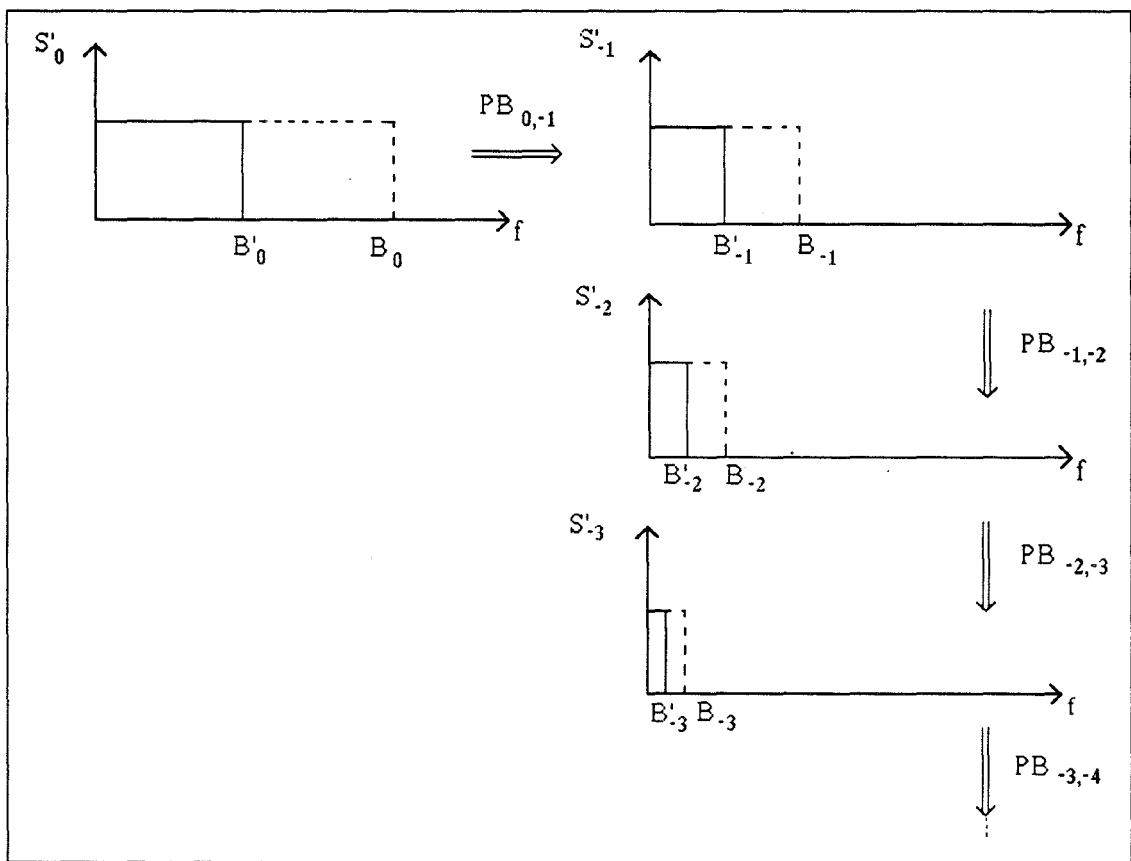


Figure II-10. Exemple de décomposition avec $\alpha = 1/2$.

Si on fait en sorte que B'_j soit égal à B_{j-1} , on constate qu'on obtient la même représentation qu'avant le changement d'échelle, à la seule différence qu'elle est décalée d'un cran sur l'échelle des résolutions.

On obtiendra ce résultat si on établit entre le facteur de changement d'échelle α et les niveaux de résolution r_j la relation suivante :

$$r_j = \alpha^j \quad (2.2)$$

En d'autres termes, la suite des niveaux de résolution $(r_j)_{j \in \mathbb{Z}^*}$ doit avoir une variation exponentielle définie par l'équation (2.2). α est alors appelé pas ou facteur de résolution. Cela implique que la bande passante du filtre passe-bas $PB_{n,m}$ soit réduite à chaque fois de $\frac{1}{\alpha}$.

II.6 - La pyramide de Burt et Adelson

En illustration du schéma de décomposition que nous avons donné, décrivons succinctement la pyramide de Burt et Adelson [BUR83a]. Selon le principe de la figure II-8, Burt et Adelson ont effectué une décomposition en prenant comme opérateur $A_{n,m}$, un filtre W de taille 5×5 de type "moyenne pondérée", et tel que les pixels du niveau j aient tous la même contribution dans le calcul des pixels du niveau $j-1$. L'image de niveau r_{j-1} sera donnée par la convolution de l'image origine avec ce filtre. De manière plus générale, l'intensité d'un pixel d'une approximation au niveau r_j est obtenue à partir du niveau précédent par la formule :

$$I_j(x,y) = \sum_{p=-2}^2 \sum_{q=-2}^2 W(p,q) I_{j-1}(2x+p, 2y+q) \quad (2.3)$$

Si l'on se contente d'une seule direction (ligne ou colonne), on aboutit à la représentation de la figure II-11.

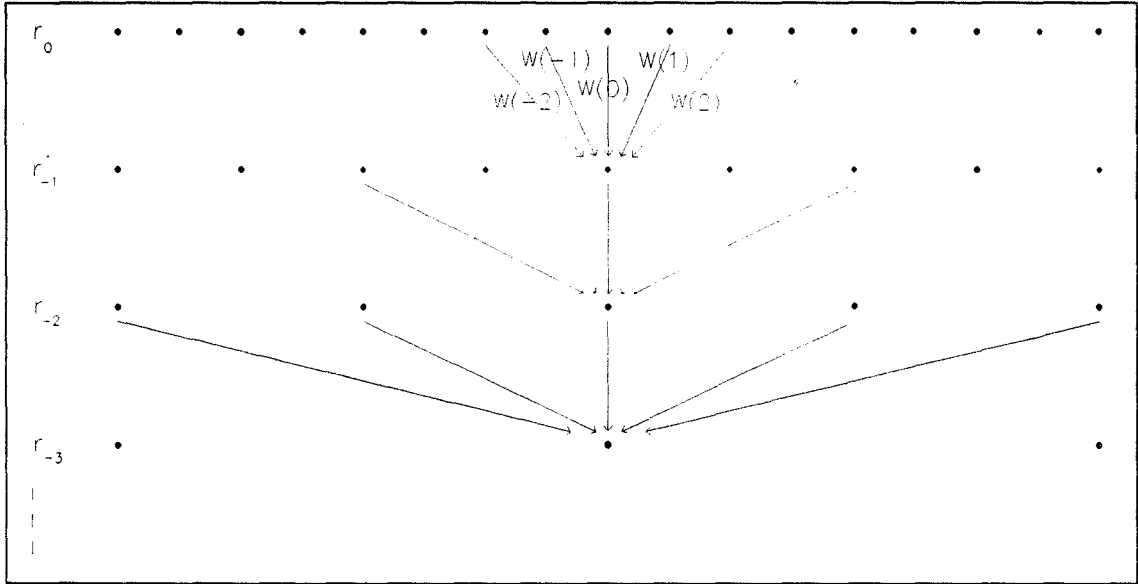


Figure II-11. Illustration à une dimension de la construction de la pyramide gaussienne.

Parmi les propriétés du filtre W figure la séparabilité. L'image de résolution inférieure est donc obtenue en appliquant cet algorithme aux lignes et aux colonnes.

On peut également s'intéresser à la manière d'obtenir l'image I_j directement à partir de I_0 . Cela revient à convoluer I_0 avec un filtre équivalent $h_j(p,q)$ qui traduit la succession de filtrages par W , on a alors :

$$I_j = h_j * I_0 \quad (2.4)$$

d'où :

$$I_j(x,y) = \sum_{p=-M_j}^{M_j} \sum_{q=-M_j}^{M_j} h_j(p,q) I_0(x2^j+p, y2^j+q) \quad (2.5)$$

La figure II-12 donne la représentation des filtres équivalents h_j (à une dimension) pour des coefficients de W donnés.

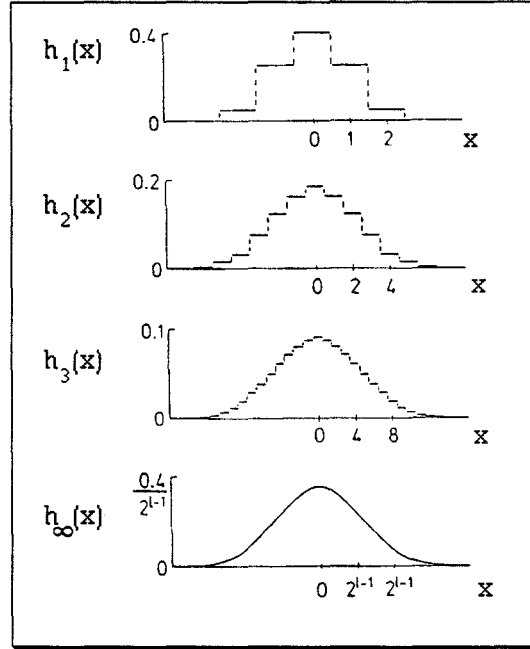


Figure II-12. Filtres équivalents pour $j = 1, 2, 3$ et l'infini de la décomposition de Burt et Adelson.

Comme on peut le constater, le filtre $h_j(x)$ correspond à une gaussienne. La largeur $2M_j$ de ce filtre, double d'un niveau à l'autre, ainsi que la distance entre deux échantillons. Ceci va correspondre, dans l'espace fréquentiel, à une division par deux de la bande passante du filtre. On réalise ainsi par cette méthode, une décomposition multirésolution de facteur de résolution $\alpha = \frac{1}{2}$ comme nous l'avons défini précédemment.

La bande spectrale étant réduite d'un facteur 2 à chaque fois, d'après le théorème de Shannon, on peut opérer un sous-échantillonnage selon le même facteur, ce qui ne conduit pas à une perte supplémentaire de

résolution. Il est donc possible de ne calculer qu'un point sur deux en passant d'un niveau au suivant.

Il en résulte une succession d'images dont la taille et la résolution vont en diminuant. Si l'on imagine ces images empilées les unes sur les autres avec comme base l'image origine, on obtient une structure pyramidale, comme le montre la figure II-13.

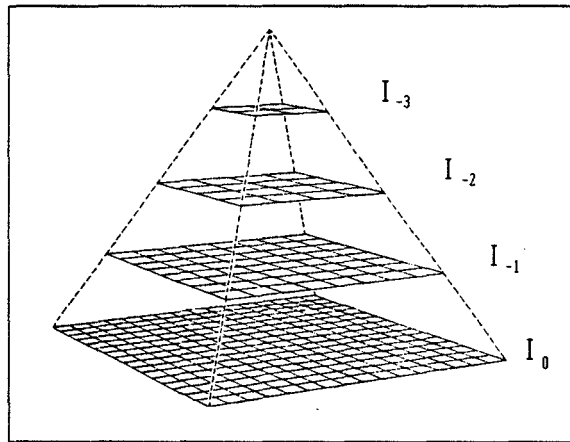


Figure II-13. Exemple de pyramide à quatre niveaux.

Pour toutes ces raisons, cette décomposition est appelée pyramide gaussienne multirésolution. On peut voir les six premiers niveaux de cette pyramide, effectués sur l'image "Lady" sur la figure II-14.



Figure II-14. Les six premiers niveaux de la pyramide gaussienne de la "Lady".

Pour bien mettre en évidence la perte de résolution entre chaque niveau, il faut garder la même échelle pour toutes les images. Par une interpolation, consistant à intercaler un zéro entre deux échantillons et à filtrer l'ensemble par un passe-bas [CRO81] les images de tailles inférieures sont remises au format de la première, et le résultat apparaît sur la figure II-15.



Figure II-15. Les quatre premiers niveaux de la pyramide gaussienne obtenus après expansion.

On observe que le filtrage passe-bas se traduit effectivement par une perte d'information qui rend l'image de plus en plus floue. Cette perte correspond à ce que nous avons défini comme étant les détails. Ces détails se retrouvent dans une seconde décomposition que Burt et Adelson ont construite à partir de la pyramide gaussienne, en faisant la soustraction entre deux images successives (figure II-16). Soit D_j , l'image de détails du niveau j , on a :

$$D_j = I_j - I_{j-1} \quad (2.6)$$



Figure II-16. Obtention des quatre premiers niveaux de la pyramide laplacienne de Burt et Adelson

Il convient de noter que les détails les plus fins se situent dans les premiers niveaux, tandis que pour faire apparaître les composantes plus grossières il faut aller plus en profondeur dans la décomposition.

De la même manière que pour la pyramide gaussienne, on peut se poser la question de savoir quelle est la nature du filtre équivalent permettant d'obtenir directement un élément de cette seconde décomposition à partir de I_0 . De (3.6) il découle que :

$$D_j = I * h_j - I * h_{j-1} = I * (h_j - h_{j-1}) \quad (3.7)$$

On peut montrer que ce filtre équivalent, différence de deux gaussiennes, est un filtre passe-bande de type laplacien. La représentation des détails a donc été appelée pyramide laplacienne.

Après avoir décomposé une image jusqu'à un certain niveau, on peut effectuer le chemin inverse et reconstruire l'image en partant du dernier niveau de la pyramide. Les étapes de la décomposition et de la reconstruction dans l'exemple de Burt et Adelson sont schématisés par la figure II-17.

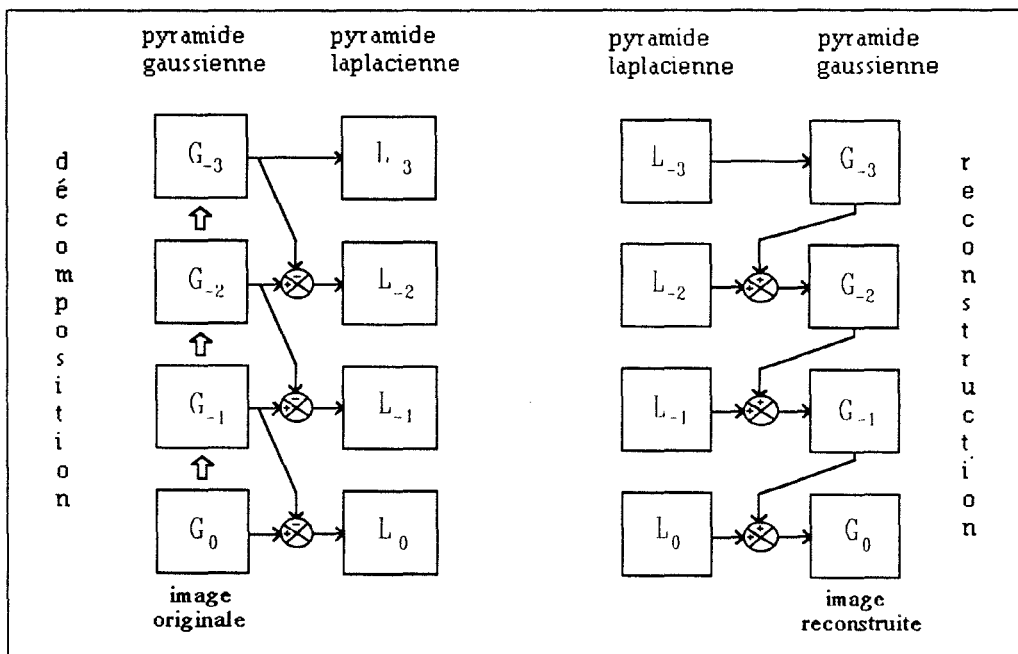


Figure II-17. Décomposition et reconstruction selon Burt et Adelson.

A ce stade de notre étude, il importe de mettre en exergue les deux remarques suivantes :

- Il est possible, grâce à ce concept, de décomposer un signal par filtrages passe-bas et passe-haut successifs en vue d'obtenir respectivement des versions approximées du signal et des détails additionnels.

- Comme on peut le constater sur la figure II-17, la pyramide laplacienne, avec à son sommet le dernier niveau de la pyramide gaussienne, permet de reconstruire l'image de départ. En d'autres termes, cet ensemble est équivalent à l'image origine. Si on dispose au départ d'une image $N \times N$ pixels, on arrive, avec la pyramide laplacienne, à un nombre total de $\frac{4}{3} N^2$ pixels. On a donc d'une part N^2 pixels pour l'image de départ, et d'autre part $\frac{4}{3} N^2$ pour sa représentation. Cette augmentation du nombre de pixels signifie que l'information est redondante dans la représentation ainsi construite.

On peut, grâce à une décomposition adéquate, supprimer cette redondance en optimisant l'approximation réalisée à chaque niveau de la décomposition. Mais pour ce faire, nous allons faire quelques rappels sur la manière d'approximer une fonction. Cela nous mènera alors à une formulation plus mathématique de l'analyse multirésolution.

II.7 - Approximation d'une fonction

Nous rappelons quelques éléments de la théorie des espaces de Hilbert [GAS90]. Ces espaces généralisent $\mathbb{R}^n$ muni de la distance euclidienne. En outre, leur caractère complet permet de développer la notion de base en

dimension infinie qui sera importante par la suite. L'exemple type d'espace de Hilbert est $L^2(I)$. I étant un intervalle borné ou non de $\mathbb{R}$, on appelle $L^2(I)$ l'espace des fonctions de I dans $\mathbb{R}$ ou $\mathbb{C}$, mesurables et de carré intégrable. On cherche alors à approcher le mieux possible une fonction de $L^2(I)$ par une combinaison linéaire de n fonctions, n pouvant être infini.

Définitions:

a - Soit E un espace vectoriel sur K ($\mathbb{C}$ ou $\mathbb{R}$). Un produit scalaire sur E est une application notée $(\cdot, \cdot)$ de $E \times E$ à valeurs dans K telle que pour tout $x, y, z \in E$ et $\alpha \in K$ on ait :

$$* (x, x) \geq 0 \text{ et } (x, x) = 0 \Rightarrow x = 0$$

$$* (x, y) = (y, x)$$

$$* (x + y, z) = (x, z) + (y, z) \text{ et } (\alpha x, y) = \alpha(x, y)$$

b - On appelle espace préhilbertien, un espace vectoriel muni d'un produit scalaire.

c - Un espace préhilbertien est un espace vectoriel normé avec la norme $\|f\| = \sqrt{(f, f)}$.

d - Lorsqu'un espace préhilbertien H muni de la norme $\|f\| = \sqrt{(f, f)}$ est complet, on dit que H est un espace de Hilbert.

e - Soit H un espace de Hilbert, on dit que x et $y \in H$ sont orthogonaux si $(x, y) = 0$. Soit S une partie de H , on appelle orthogonal de S , l'ensemble noté $S^\perp$ défini par :

$$S^\perp = \{ y \in H, (x, y) = 0 \text{ } x \in S \}.$$

f - Meilleure approximation sur un sous espace vectoriel:

Etant donné H un espace préhilbertien, V un sous espace vectoriel de H et $f \in H$, on se pose la question suivante :

$$\text{Existe-t-il } f^* \in V \text{ tel que } \|f - f^*\| = \min\{\|f - v\|, v \in V\}?$$

La réponse est donnée par le théorème (de projection orthogonale) qui est le suivant:

- Soient H un espace préhilbertien et V un sous espace vectoriel complet. Etant donné $f \in H$, il existe un et un seul $f^* \in V$ tel que :

$$\|f - f^*\| = \min\{\|f - v\|, v \in V\}.$$

Cette norme dérivant d'un produit scalaire, le calcul de la meilleure approximation en est facilité, d'où la caractérisation suivante de f^* :

- Soient H un espace préhilbertien, V un sous espace vectoriel et $f \in H$, f^* est la meilleure approximation de f sur V si et seulement si f^* vérifie :

$$(f - f^*, v) = 0 \text{ pour tout } v \in V.$$

Cette condition nécessaire et suffisante permet de calculer f^* lorsque V est un sous espace de dimension n . Si $\phi_1, \phi_2, \dots, \phi_n$ est une base de V , on écrit :

$$f^* = \sum_{k=1}^n \lambda_k \phi_k \quad (2.8)$$

Dans le cas où la base est orthogonale, l'expression de λ_k est :

$$\lambda_k = \frac{(f, \phi_k)}{(\phi_k, \phi_k)} \quad (2.9)$$

Si les ϕ_k sont orthonormés, on a $\lambda_k = (f, \phi_k)$. f^* est alors le polynôme de meilleure approximation de f en norme quadratique. Cette notion se généralise à un espace de Hilbert quelconque.

II.8 - Définition d'une analyse multirésolution de $L^2(\mathbb{R})$

$L^2(\mathbb{R})$ est l'espace vectoriel des fonctions $f(x)$ monodimensionnelles mesurables et de carrés intégrables.

L'idée est de représenter toute fonction $f(x)$ de $L^2(\mathbb{R})$ par une succession d'approximations; chaque approximation étant la projection de $f(x)$ sur un sous espace de $L^2(\mathbb{R})$. En d'autres termes, une analyse multirésolution de $L^2(\mathbb{R})$ est une suite de sous espaces vectoriels fermés emboîtés V_m de $L^2(\mathbb{R})$, $m \in \mathbb{Z}$:

$$\dots \supset V_2 \supset V_1 \supset V_0 \supset V_{-1} \supset V_{-2} \supset \dots$$

Cette suite de sous espaces vectoriels doit en outre vérifier un certain nombre de propriétés que nous énumérons ci-dessous.

Soit A_j l'opérateur qui approxime une fonction $f(x)$ de $L^2(\mathbb{R})$ à la résolution r_j . Nous allons caractériser A_j conformément aux propriétés de la décomposition dont nous avons fait état à la section II.5.

1 - A_j est un opérateur linéaire. Si $A_j f(x)$ est l'approximation à la résolution r_j , alors $A_j f(x)$ n'est pas modifié si on l'approxime de nouveau à la même résolution. Il vient que :

$$A_j \circ A_j = A_j$$

A_j est donc un opérateur de projection sur un sous espace particulier V_j de $L^2(\mathbb{R})$. Ce sous espace V_j peut être considéré comme étant l'ensemble de toutes les approximations à la résolution r_j de toute fonction $f(x)$ de $L^2(\mathbb{R})$.

2 - $A_j f(x)$ doit être la meilleure approximation de $f(x)$ dans V_j . C'est-à-dire :

$$\forall g(x) \in V_j \quad ||g(x) - f(x)|| \geq ||A_j f(x) - f(x)||$$

On en déduit que A_j est l'opérateur de projection orthogonale sur V_j .

[3] - L'approximation de $f(x)$ à la résolution r_{j+1} doit contenir toute l'information nécessaire pour calculer l'approximation à la résolution inférieure r_j , d'où :

$$\forall j \in \mathbb{Z} \quad V_j \subset V_{j+1}$$

[4] - La suite $(r_j)_{j \in \mathbb{Z}}$ des niveaux de résolution doit varier exponentiellement d'après ce que nous avons vu à la section II.5. On pose alors :

$$\forall j \in \mathbb{Z}, \quad f(x) \in V_j \quad \Leftrightarrow \quad f(2x) \in V_{j+1}$$

Ce faisant, on impose un pas de résolution $\alpha = \frac{1}{2}$.

[5] - Lorsqu'on fait tendre le niveau de résolution vers l'infini, l'approximation doit converger vers la fonction $f(x)$; inversement, lorsque la résolution décroît, $A_j f(x)$ contient de moins en moins d'information et doit tendre vers zéro. Cela se traduit respectivement par :

$$* \lim_{j \rightarrow +\infty} V_j = \bigcup_{j=-\infty}^{+\infty} V_j \quad \text{est dense dans } L^2(\mathbb{R})$$

et

$$* \lim_{j \rightarrow -\infty} V_j = \bigcap_{j=-\infty}^{+\infty} V_j = \{0\}$$

[6] - Il existe une fonction $g(x) \in V_0$ telle que $\{g(x-k)\}_{k \in \mathbb{Z}}$ soit une base de Riesz pour V_0 . On peut de ce fait construire une base orthonormée de V_0 sur laquelle on pourra projeter $f(x)$. L'énoncé de cette propriété pour $j=0$ est suffisante compte tenu de la propriété 3.

On appellera tout ensemble d'espaces vectoriels $(V_j)_{j \in \mathbb{Z}}$ vérifiant ces propriétés, une approximation multirésolution de $L^2(\mathbb{R})$.

II.9 - Analyse multirésolution et ondelettes

Nous allons maintenant présenter un résumé des travaux de S. Mallat [MAL89a] qui ont permis d'établir un lien systématique entre l'analyse multirésolution et la théorie des ondelettes.

Nous venons de voir que l'opérateur A_j est un opérateur de projection orthogonale sur l'espace vectoriel V_j . Il faut donc trouver une base orthogonale de V_j pour pouvoir caractériser A_j . Le théorème suivant nous renseigne sur l'existence de cette base :

Soit $(V_j)_{j \in \mathbb{Z}}$, une approximation multirésolution de $L^2(\mathbb{R})$. Il existe une et une seule fonction $\phi(x) \in L^2(\mathbb{R})$, appelée fonction d'échelle, telle que si on pose $\phi_j(x) = 2^j \phi(2^j x)$ avec $j \in \mathbb{Z}$, la dilatation ou la contraction de $\phi(x)$ par 2^j , alors $(\sqrt{2^j} \phi(x-2^{-j}n))_{n \in \mathbb{Z}}$ est une base orthonormale de V_j .

Autrement dit, à chaque approximation multirésolution de $L^2(\mathbb{R})$ est associée une fonction d'échelle différente. Mallat montre que pour obtenir $A_j f(x)$, décomposition de $f(x)$ sur la base donnée par le théorème, cela revient à convoluer $A_{j+1} f(x)$ avec un filtre discret H_s dont la réponse impulsionnelle est déduite du produit scalaire :

$$h(n) = \langle \phi_{-1}(u), \phi(u-n) \rangle \quad \forall n \in \mathbb{Z} \quad (2.10)$$

par la relation $h_s(n) = h(-n)$.

La transformée de Fourier de $h(n)$, $H(\omega)$, doit satisfaire les relations ci-après :

- $|H(0)|=1$ et $h(n) = O(n^{-2})$ à l'infini (a)
- $|H(\pi)|=0$ (b)
- $|H(\omega)|^2 + |H(\omega+\pi)|^2 = 1$ (c)

Les filtres vérifiant (b) sont dits conjugués [GRI85],[MIL85],[PIR84]. Ces propriétés montrent en outre que H est un passe-bas.

Inversement, on peut définir une fonction d'échelle régulière ϕ par sa transformée de Fourier $\Phi(\omega)$, à partir de la donnée d'un filtre conjugué $H(\omega)$ vérifiant en plus des propriétés (a), (b) et (c) les suivantes [ANT92] :

$$* |H(\omega)| \neq 0 \text{ pour } \omega \in [0, \pi], \quad (d)$$

$$* \prod_{p=1}^{\infty} H(2^{-p}\omega) \text{ décroît plus vite que } C(1+|\omega|)^{-\varepsilon-1/2} \text{ quand } |\omega| \rightarrow \infty \text{ pour } \varepsilon > 0 \quad (e)$$

par la relation :

$$\Phi(\omega) = \prod_{p=1}^{\infty} H(2^{-p}\omega) \quad (2.11)$$

Le coefficient de régularité est donné par la pente de la décroissance. Une condition suffisante pour obtenir la régularité est que $H(\omega)$ puisse s'écrire sous la forme [DAU88] :

$$H(\omega) = \left[\frac{1}{2} (1 + e^{i\omega}) \right]^N \left[\sum_n f(n) e^{in\omega} \right]$$

La régularité dépend alors de l'exposant N .

On passe d'un niveau d'approximation au suivant en appliquant le schéma de la figure II-18 où on convolue avec le filtre H_s puis on garde un échantillon sur deux.

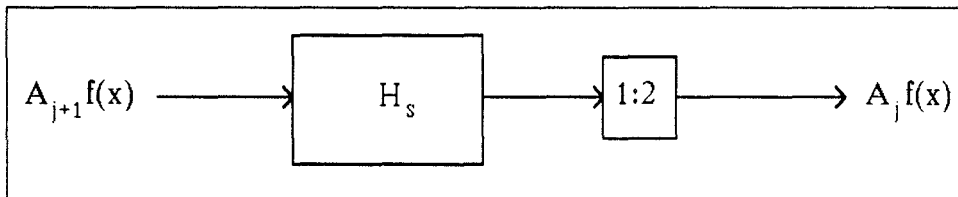


Figure II-18. Schéma de décomposition de $A_{j+1}f$ en $A_j f$.

Si on part d'un niveau de référence indicé $j=0$ et que l'on itère ce processus J fois, on obtient alors l'ensemble $(A_j f)_{-J \leq j \leq -1}$ des approximations du signal aux différentes résolutions.

II.9.1 - La représentation en ondelettes à une dimension

Nous expliquons maintenant comment on peut, toujours selon la méthode de Mallat, construire à partir de la décomposition précédente, une seconde décomposition qui sera composée des détails.

Soit donc à extraire la différence d'information entre les approximations $A_{j+1}f(x)$ et $A_j f(x)$. Ces dernières sont obtenues en projetant $f(x)$ sur respectivement V_{j+1} et V_j . On peut montrer que les détails à la résolution r_j sont donnés par la projection $D_j f(x)$ de $f(x)$ sur l'ensemble O_j , complément orthogonal à V_j dans V_{j+1} :

$$V_{j+1} = V_j \oplus O_j \quad (2.12)$$

Il suffit alors de trouver une base orthonormale de O_j dans laquelle s'effectuera la décomposition, de telle sorte qu'on ait :

$$A_{j+1}f(x) = A_j f(x) + D_j f(x) \quad (2.13)$$

Tout comme pour la fonction d'échelle, on montre qu'il existe une fonction $\psi(x)$ dont les dilatations et contractions sont définies par $\psi_j(x) = 2^{j/2} \psi(2^j x)$ et telle que $(\frac{1}{\sqrt{2^j}} \psi_j(x - 2^{-j}n))_{n \in \mathbb{Z}}$ soit une base orthonormale de

$\psi_j(x) = 2^{j/2} \psi(2^j x)$ et telle que $(\sqrt{2^{-j}} \psi_j(x-2^{-j}n))_{n \in \mathbb{Z}}$ soit une base orthonormale de O_j .

Soient $\phi(x)$ la fonction d'échelle et H le filtre conjugué associé, la fonction $\psi(x)$ est alors définie par sa transformée de Fourier :

$$\Psi(\omega) = G\left[\frac{\omega}{2}\right] \Phi\left[\frac{\omega}{2}\right] \quad \text{où } G(\omega) = e^{-i\omega} \overline{H(\omega+\pi)} \quad (2.14)$$

$\psi(x)$ est appelée ondelette orthogonale. D'après l'expression de $G(\omega)$ en fonction de $H(\omega)$, les réponses impulsionnelles de ces deux filtres sont liées par la relation :

$$g(n) = (-1)^n h(1-n) \quad (2.15)$$

G est le filtre miroir de H et est un passe-haut. Dans la littérature, G et H sont appelés filtres miroirs en quadrature ou QMF (Quadrature Mirror Filters)[EST77]. Daubechies montre qu'on peut alors, en fonction des propriétés de régularité et de nullité des moments que l'on veut attribuer à la fonction d'échelle et à l'ondelette, choisir le filtre H en conséquence[DAU88].

Les travaux de Daubechies permettent d'obtenir des filtres à support compact, résultats dont nous nous servirons dans le chapitre III. Ces filtres sont non symétriques et à phase minimale. Leur régularité croît linéairement avec la longueur spatiale et on peut obtenir des moments nuls jusqu'à un ordre n quelconque en augmentant la taille des filtres.

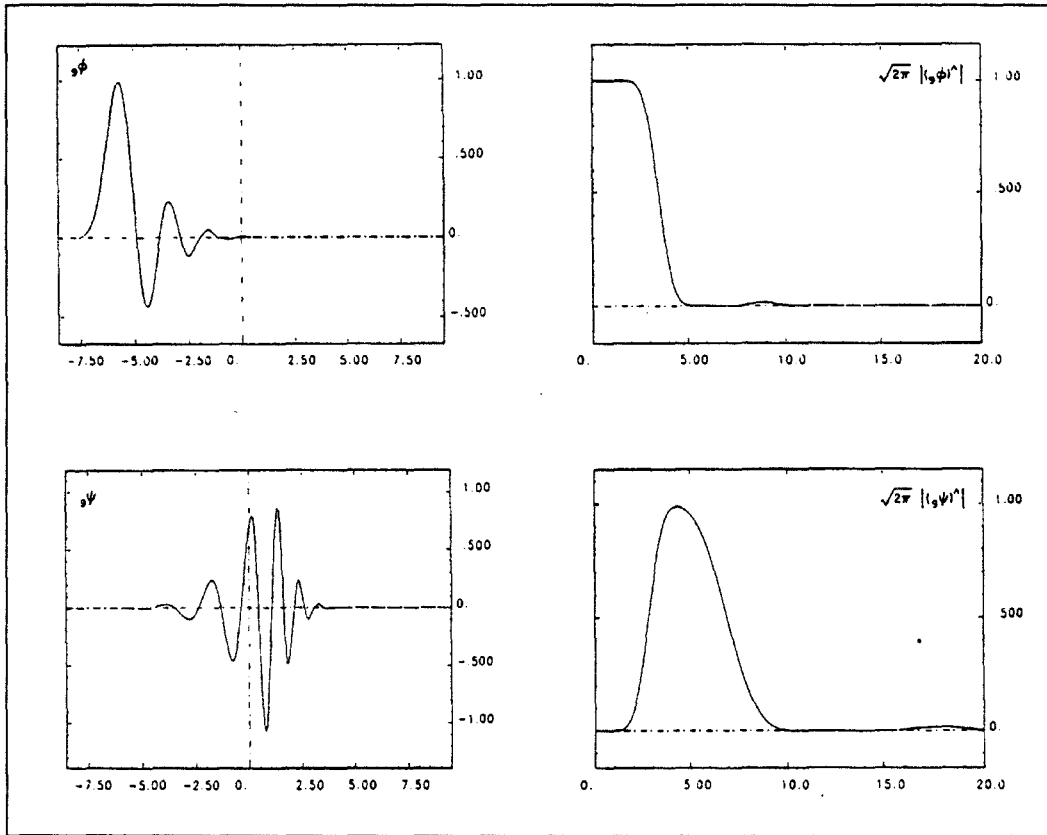


Figure II-19. Exemple de fonction d'échelle ϕ , et de son ondelette associée. Les transformées de Fourier sont également représentées. D'après [DAU88].

Un exemple de fonctions est donnée figure II-19 avec les filtres de Daubechies d'horizon 9, où on peut noter que l'ondelette $\psi(x)$ est équivalente à un filtre passe-bande. Dans notre partie pratique, un filtre de Daubechies d'horizon p est noté $F_p(n)$. Cet horizon nous donne un filtre ayant $p+1$ coefficients. Il est relié à l'indice N classique des filtres de Daubechies par la relation : $N = 2p-1$.

Mallat a montré qu'il existe également un algorithme pyramidal simple qui permet de déterminer $D_j f(x)$ en convoluant $A_{j+1} f(x)$ avec G_S (défini d'une manière similaire à H_S) et en retenant ensuite un échantillon sur deux. En considérant une profondeur de décomposition J (toujours avec $j=0$ comme niveau de départ), la cellule élémentaire de la figure II-20 va permettre alors

d'obtenir d'une part l'ensemble $(A_j f)_{-J \leq j \leq -1}$ des approximations de $f(x)$, et d'autre part l'ensemble $(D_j f)_{-J \leq j \leq -1}$ des détails aux différents niveaux de résolution. Le signal f de départ est équivalent à l'ensemble $(A_j f, (D_j f)_{-J \leq j \leq -1})$. Cet ensemble est appelé représentation en ondelettes orthogonale de f .

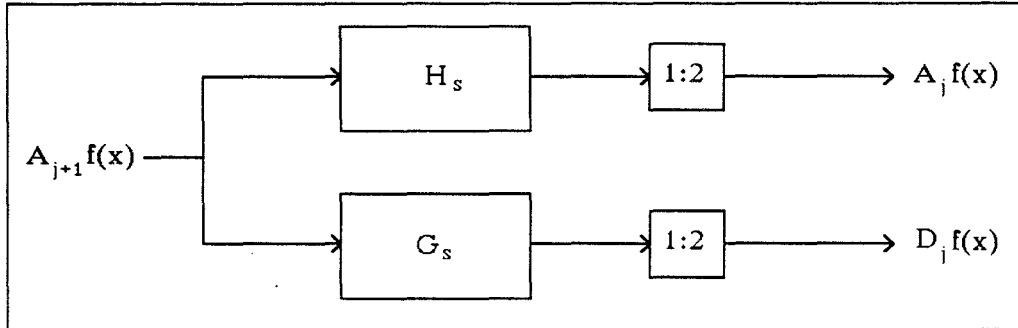


Figure II-20. Schéma de décomposition en ondelettes à une dimension. de $A_{j+1}f$.

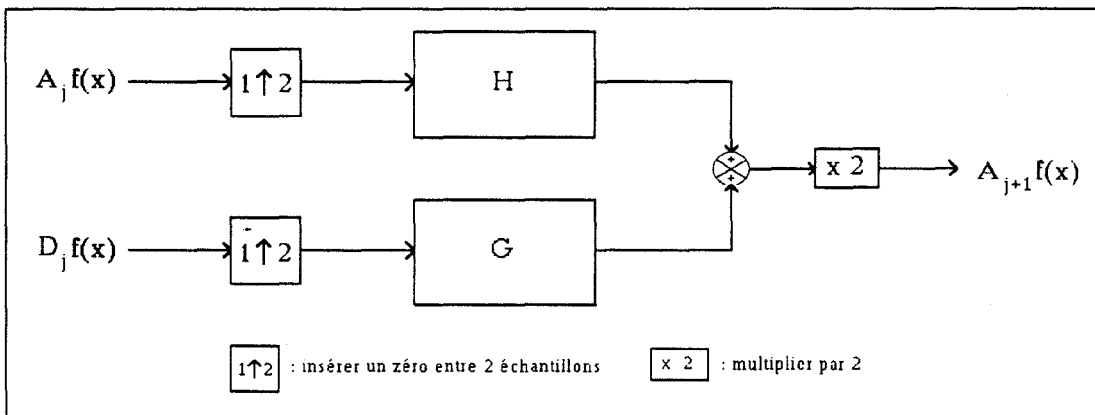


Figure II-21. Principe de la reconstruction de $A_{j+1}f$.

On peut bien sûr reconstruire le signal de départ (niveau 0) à partir de sa représentation en ondelettes. On itère pour cela l'algorithme illustré par la figure II-21, en partant du niveau $-J$.

Le rajout d'un zéro entre deux échantillons suivi du filtrage, réalise également l'interpolation pour recouvrer le format supérieur (comme dans le cas de Burt et Adelson).

II.9.2 - Généralisation à la dimension n : cas des images

Ce modèle de décomposition peut être généralisé à n dimensions. Dans le cas des images, où $n=2$, on considère alors un signal $f(x,y) \in L^2(\mathbb{R}^2)$. $L^2(\mathbb{R}^2)$ est l'espace vectoriel des fonctions $f(x,y)$ à deux dimensions, mesurables et de carré intégrable. Une approximation multirésolution de $L^2(\mathbb{R}^2)$ est une suite d'espaces vectoriels V_j vérifiant les propriétés 2 à 6 de la section II.8, étendues à deux dimensions. On se limite ici au cas des approximations multirésolutions dites séparables [MEY] où chaque espace vectoriel V_j s'écrit sous la forme d'un produit tensoriel de deux sous espaces identiques de $L^2(\mathbb{R})$:

$$V_j = V'_j \otimes V'_j \quad (2.16)$$

La suite $(V'_j)_{j \in \mathbb{Z}}$ est elle-même une approximation multirésolution de $L^2(\mathbb{R})$. Cette hypothèse se vérifie pour bon nombre d'images rencontrées en traitement d'images. Le raisonnement tenu à une dimension est alors applicable et les images de résolutions inférieures sont obtenues grâce à une fonction d'échelle $\phi(x) \in L^2(\mathbb{R}^2)$ telle que :

$$\phi(x,y) = \phi(x) \phi(y) \quad (2.17)$$

$\phi(x)$ est une fonction d'échelle de $(V'_j)_{j \in \mathbb{Z}}$ et $\{2^j \phi_j(x-2^j n) \phi_j(y-2^j m)\}_{(n,m) \in \mathbb{Z}^2}$, une base orthonormale de V_j .

La pyramide multirésolution des approximations $(A_j f)_{-1 \leq j \leq J}$, est construite par l'itération de l'algorithme de la figure II-22 qui est l'application de celui de la figure II-18 en ligne et colonne.

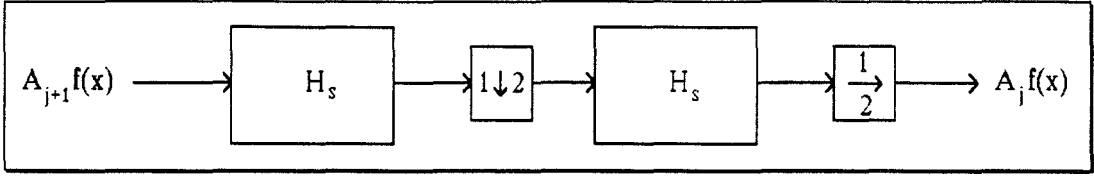


Figure II-22. Généralisation du principe de la figure II-18 dans le cas d'une image.

On filtre les lignes puis on retient une colonne sur deux; ensuite on filtre en colonnes puis on garde une ligne sur deux.

Comme dans le cas monodimensionnel, les détails sont issus de la projection sur O_j , complément orthogonal à V_j dans V_{j+1} . Soit alors $\psi(x)$, l'ondelette associée à la fonction d'échelle $\phi(x)$ de la relation (2.17). La base orthonormale de O_j est :

$$\{ 2^{-j} \psi(x-2^{-j}n, y-2^{-j}m), 2^{-j} \psi(x-2^{-j}n, y-2^{-j}m), 2^{-j} \psi(x-2^{-j}n, y-2^{-j}m) \}_{(n,m) \in \mathbb{Z}^2}$$

$$\text{avec } \psi^1(x,y)=\phi(x)\psi(y) \quad \psi^2(x,y)=\psi(x)\phi(y) \quad \psi^3(x,y)=\psi(x)\psi(y)$$

La projection de $f(x,y)$ sur cette base conduit alors à trois images de détails D_j^1f , D_j^2f et D_j^3f , correspondant aux trois ondelettes ci-dessus définies. On

montre que ces trois images caractérisent les détails dans les directions respectivement verticale, horizontale et dans les deux à la fois.

La représentation en ondelettes d'une image donnée (niveau 0) décomposée jusqu'à l'ordre J est donnée par l'ensemble :

$$\left[A_{-j}f, (D_j^1f), (D_j^2f), (D_j^3f) \right]_{-J \leq j \leq -1} \quad (2.18)$$

La cellule élémentaire permettant l'obtention de cette représentation apparaît figure II-23.

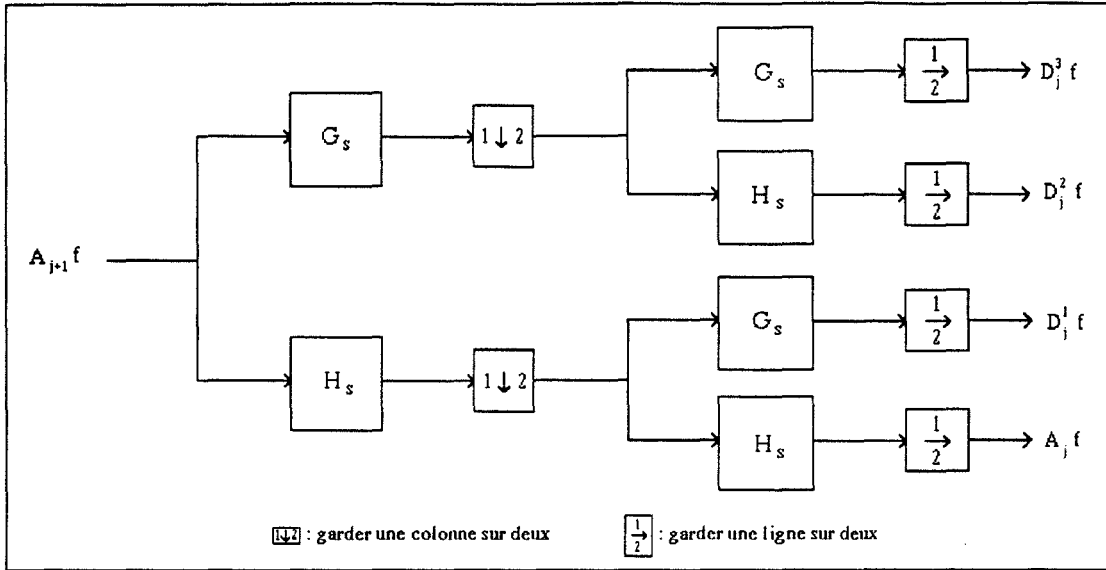


Figure II-23. Décomposition en ondelettes d'une image $A_{j+1}f$.

Cet algorithme de décomposition est l'extension de celui à une dimension, tenant compte de la propriété de séparabilité des fonctions. Les filtres utilisés sont les mêmes que ceux à une dimension.

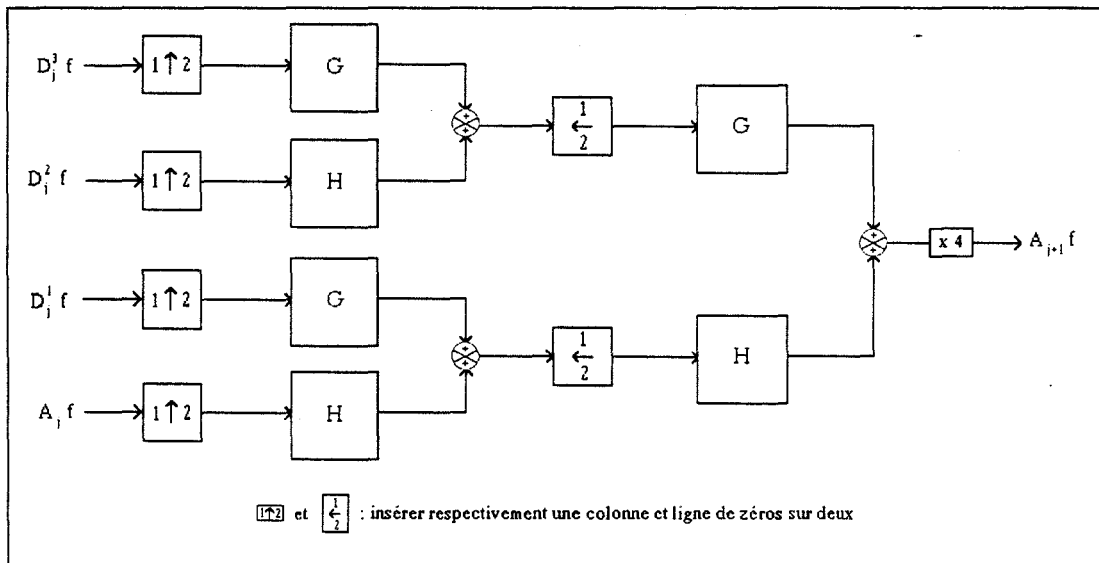


Figure II-24. Reconstruction de l'image $A_{j+1}f$ à partir des produits de décomposition.

Enfin, on peut de nouveau recouvrer l'image de départ à partir de sa représentation en ondelettes en itérant le processus de la figure II-24.

De nombreux auteurs [ANT92][COH90][FEA90][FRO90][LEM86][MAL91][VET90], ont travaillé sur les ondelettes, leurs applications et plus généralement sur les algorithmes pyramidaux et leurs applications [ADE87][WOO86][BOS92]. Une part importante de ces travaux a consisté en la détermination de nouvelles bases d'ondelettes. On peut citer tout particulièrement ceux d'I. Daubechies [DAU88], auxquels nous avons déjà fait référence, qui a construit une famille de bases d'ondelettes orthogonales à support compact, c'est-à-dire conduisant à des réalisations à l'aide de filtres à réponse impulsionnelle finie (FIR). Les coefficients normalisés à $\sqrt{2}$ de la réponse impulsionnelle des dix premiers filtres de cette famille sont donnés en annexe, ainsi que deux exemples de représentation des fonctions d'échelle et d'ondelette associées.

La possibilité de contrôler l'horizon temporel (ou spatial dans le cas des images) de ces filtres, peut être d'un intérêt non négligeable dans des applications où la notion de recouvrement entre les zones de traitement n'est pas sans importance.

C'est le cas dans les problèmes d'estimation de mouvements par block matching hiérarchique où le passage d'un niveau de résolution donné à celui de résolution inférieure par filtrage, provoque un recouvrement entre des blocs de recherche apparemment disjoints. Ce recouvrement est d'autant plus important que l'horizon du filtre est grand. Or, dans la mesure où rien n'impose au champ de vecteurs d'être continu sur toute l'image, ce recouvrement peut être à l'origine d'aberrations dans l'estimation du mouvement.

Cette raison fait qu'il apparaît intéressant d'utiliser cette famille de filtres dans le cadre de l'application des techniques de block matching sur une représentation multirésolution d'images.

Nous avons donc mené une étude afin d'étudier les interactions qui peuvent exister entre l'horizon des filtres, la taille des blocs de recherche et l'amplitude du mouvement. Nous allons nous attacher à caractériser l'influence de la taille des filtres en observant ce qui se produit lors du passage d'un niveau à un autre, dans un schéma de compensation de mouvement hiérarchique par décomposition multirésolution.

CHAPITRE III :
COMPENSATION DE MOUVEMENT
HIERARCHIQUE PAR DECOM-
POSITION MULTIRESOLUTION

COMPENSATION DE MOUVEMENT HIERARCHIQUE PAR DECOMPOSITION MULTIREOLUTION

III.1 - Introduction

Dans ce chapitre nous allons présenter les résultats de notre travail, en exposant auparavant la méthode et les moyens mis en oeuvre pour effectuer l'étude. Rappelons que le but recherché est la caractérisation de l'influence de la taille des filtres lors du passage d'un niveau de résolution au suivant. Nous avons opté pour l'emploi du block matching comme estimateur de mouvement mais il est à noter que des travaux récents ont été effectués dans le domaine de la compensation de mouvement dans un schéma multirésolution, en utilisant la technique du pel-récuratif [BAA91a] [BAR92] [GAI92].

Si l'utilisation de la représentation multirésolution semble naturelle dans le cadre de la compensation de mouvement par la technique de block matching, il n'en demeure pas moins que la mise en pratique de cette méthode pose un certain nombre de problèmes liés principalement à la façon de "passer" les vecteurs d'un niveau de résolution inférieure vers un niveau supérieur. Dans la mesure où l'hypothèse de continuité du champ de vecteurs ne peut être admise sur toute l'image, il est vain d'espérer une solution par seule interpolation linéaire. C'est en particulier le cas lorsque l'on doit traiter le problème des occlusions. Une occlusion décrit le phénomène d'apparition d'un objet caché par un autre lors du déplacement de ce dernier.

Barlaud, Blanc-Feraud et Colin [BAR92] ont proposé une méthode d'estimation du mouvement prenant en compte les discontinuités du champ de vecteurs. Cette méthode, basée sur un modèle de Markov du mouvement, fait appel à un nouveau critère de régularité qui permet de mieux contrôler les ruptures du champ de vecteurs. Elle a reçu le nom de "Time method" et a été appliquée avec succès pour le codage de séquences d'images [GAI92].

D'autres problèmes de mise en oeuvre se posent, parmi lesquels on peut citer la perte progressive de localisation des objets dans l'image, au fur et à mesure que l'on parcourt la pyramide vers les niveaux de résolution inférieure. Cet effet, combiné à la délimitation de blocs imposée par la méthode de calcul des vecteurs, ne sera pas sans influence sur la qualité du résultat obtenu.

Afin d'analyser ces différents aspects, nous avons dans un premier temps, réalisé un logiciel de décomposition multiéchelle, multirésolution. Ce logiciel, que nous avons écrit en langage C, présentent les principales caractéristiques suivantes :

☞ la décomposition et la reconstruction se font selon l'algorithme de Mallat que nous avons décrit dans le chapitre II.

☞ les filtres utilisés sont ceux construits par Daubechies [DAU88], dont les coefficients sont normalisés à 1. On peut inclure n'importe lequel de ces filtres pour effectuer la décomposition. Nous nous servons pour nos mesures, des filtres de Haar et de Daubechies d'horizon 3, 5 et 9 que nous désignerons respectivement par H_a , F_3 , F_5 et F_9 . Les coefficients de ces filtres sont listés dans le tableau III-1.

n	$H_a(n)$	$F_3(n)$	$F_5(n)$	$F_9(n)$
0	0,5	0,341506351	0,235233603	0,113209491
1	0,5	0,59150635	0,570558458	0,426971771
2		0,158493649	0,3251825	0,512163472
3		-0,091506351	-0,095467208	0,09788348
4			-0,060416104	-0,171328358
5			0,024908749	-0,022800566
6				0,054851329
7				-0,0044134
8				-0,008895934
9				-0,002358714

Tableau III-1. Coefficients des filtres utilisés.

Comme nous l'avons indiqué dans le chapitre II, ces filtres ne sont pas à phase linéaire. Ce décalage de phase peut donc entraîner des imprécisions dans la localisation des objets dans l'image. Cependant, dans la mesure où nous ne recherchons pas une estimation exacte du mouvement mais bien une similitude entre deux blocs afin de minimiser l'erreur quadratique moyenne, le critère de linéarité de la phase ne semble pas essentiel ici. Le choix de ces filtres a surtout été fait de manière à disposer d'une famille de filtres d'horizons spatiaux variables et connus.

☞ les effets de bords lors de la décomposition sont traités par symétrie miroir, bien que les filtres utilisés ne soient pas symétriques. Nous avons néanmoins préféré cette solution du fait que la périodisation de l'image aurait introduit des discontinuités aux bords, produisant des artefacts à la visualisation.

☞ il est possible de ne pas opérer de changement d'échelle lors de la décomposition. On peut de cette manière avoir une représentation multirésolution non pyramidale.

Nous avons ensuite écrit un deuxième logiciel, d'estimation du mouvement, toujours dans le même langage, répondant aux spécificités ci-après :

☞ la technique utilisée est celle du block matching.

☞ la taille des fenêtres de recherche est paramétrée.

☞ le critère de recherche minimise l'erreur quadratique moyenne (MSE).

☞ la procédure de recherche retenue est celle de la méthode conjuguée décrite dans le chapitre I.

☞ il n'est pas fait d'hypothèse particulière sur les bords dont les effets se trouvent négligés.

☞ dans le cas où l'estimation du mouvement se fait sur des images de résolution inférieure, possédant donc des valeurs non entières, les calculs sont

effectués en ayant au préalable arrondi ces coefficients aux valeurs entières et pris leurs valeurs absolues. Ainsi, tous les calculs (FD, DFD, MSE) se font sur des valeurs entières.

Notre station de travail se compose d'un PC-AT 486 équipé d'une carte d'acquisition MVP-AT + Ramagette, d'une caméra ccd et d'un moniteur de visualisation.

La séquence d'image sur laquelle nous avons travaillé a été réalisée en laboratoire.

III.2 - Principe de l'estimation de mouvement multirésolution

La première étape consiste à décomposer l'image courante et la précédente sur un nombre de niveaux donné J . On calcule ensuite le champ de vecteurs grossier au sommet des pyramides. Faisons remarquer au passage, que le calcul se faisant sur une image réduite, cela apporte un gain de temps.

On descend d'une marche dans la pyramide de l'image précédente où il s'agit alors d'estimer le champ de vecteurs de cette image de résolution supérieure en se servant du champ précédent comme initialisation. Compte tenu de ce que nous venons de dire, il faut donc :

- ⇒ multiplier le module des vecteurs pour prendre en considération le changement d'échelle de l'image.
- ⇒ affiner le champ de vecteurs pour compenser les effets de la délocalisation dûs au changement d'échelle.

⇒ affecter des vecteurs aux zones de l'image qui en sont dépourvus du fait de l'expansion.

Soit donc à déterminer le déplacement D_j d'un pixel X_j au niveau j , à partir de l'estimation D_{j-1} du niveau $j-1$. D_j sera déduit de l'équation suivante :

$$D_j(X_j) = k \cdot D_{j-1}(X_{j-1}) + dD \quad (3.1)$$

où k désigne le facteur compensateur du changement d'échelle.

Après être ainsi descendu jusqu'au niveau initial de résolution, on effectue alors le calcul de la DFD, qui doit être minimale si le champ de vecteur obtenu est optimum.

Différentes méthodes d'estimation de mouvement basées sur ce principe ont déjà été proposées [ANA84],[BUR83],[WAN92]. Elles se distinguent par la manière dont les vecteurs du niveau j sont déduits du niveau $j-1$. Cette déduction est bien sûr fonction du type de décomposition employé.

Pour notre étude, nous avons retenu une décomposition en ondelettes dyadique de Mallat, à cause de la perspective intéressante à notre avis, qu'elle présente de pouvoir utiliser les pyramides de détails pour, à chaque niveau, déterminer le terme d'ajustement dD de l'équation (3.1). Nous procédons de la manière suivante

Dans une première étape, nous décomposons deux images consécutives de la séquence notées k et $k-1$ sur un niveau de résolution. Afin de pas faire varier tous les paramètres en même temps (taille des blocs, horizon des filtres, amplitude du mouvement) et de ne caractériser que l'influence de l'horizon des filtres, nous nous limitons à l'observation des phénomènes lors du passage du niveau de résolution 0 au niveau -1. Il faut toutefois noter que dans la mesure où notre séquence ne comporte que des mouvements très lents cela n'est pas trop préjudiciable. En effet, au niveau de résolution -1 la distance entre deux pixels voisins correspond à une distance double au niveau 0. Ainsi, mesurer un déplacement d'un pixel à ce niveau correspond en réalité à une distance de ± 2 pixels. Plus les mouvements seront rapides, plus il faudra de niveaux de décomposition. Par exemple, au niveau -3 le déplacement d'un pixel

équivaldrait à une distance de ± 8 pixels. Tout cela est bien sûr fonction du type de décomposition.

On obtient ainsi quatre sous-images par image comme on peut le voir sur la figure III-1.

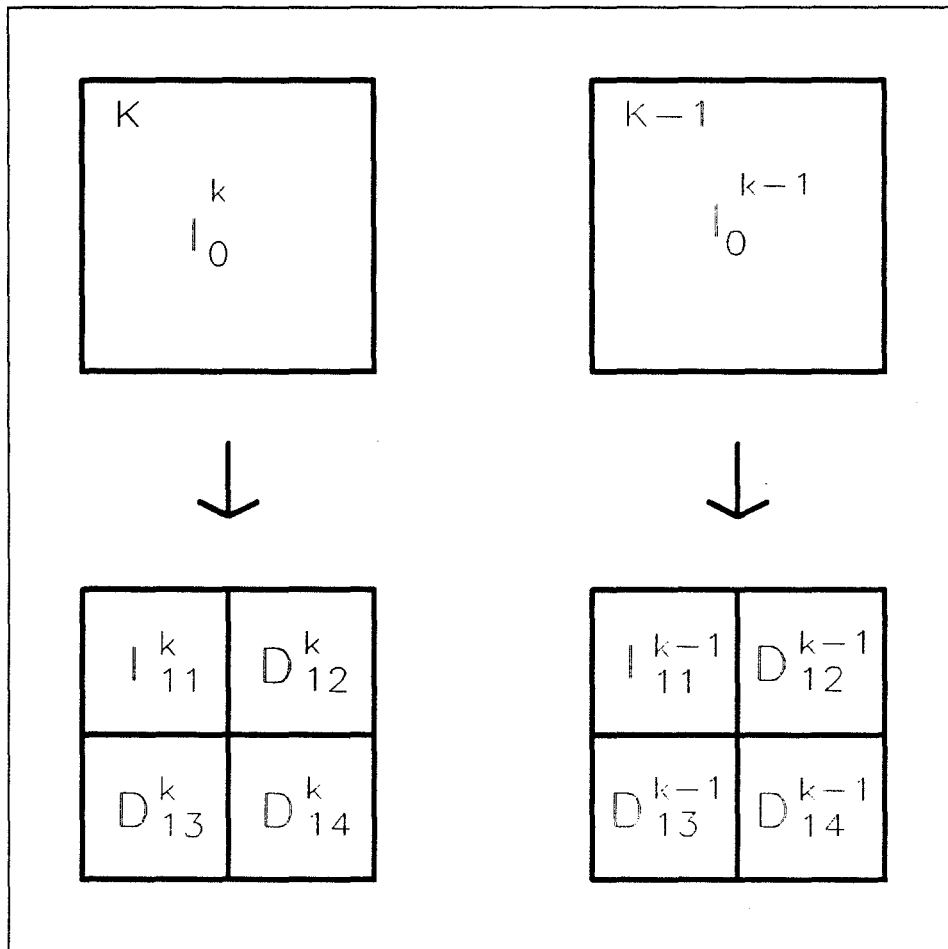


Figure III-1. Produits de la décomposition au niveau -1.

Un système de compensation complet prendra en compte l'ensemble de ces sous-images pour effectuer l'estimation. Pour l'instant cependant, nous nous limitons à l'utilisation des sous-images d'approximation I_1 . Nous opérons une recherche par block matching en utilisant des fenêtres de taille 4 pixels sur 4, qui nous donne le champ des vecteurs de déplacement D_1 . Pour interpoler ce champ au niveau supérieur, nous utilisons l'équation d'interpolation (3.1) qui s'écrit dans notre cas :

$$D_0(X_0) = 2 \cdot D_{-1}(X_{-1}) + dD \quad (3.2)$$

Il s'agit alors de trouver pour un pixel du niveau 0 à la position X_0 , le vecteur d'initialisation D_{-1} du niveau -1 qui va servir à calculer son déplacement D_0 .

Pour cela on remarque que le changement d'échelle fait qu'un bloc du niveau -1 va en engendrer quatre au niveau 0. On a donc un schéma un géniteur/quatre fils, comme l'indique la figure III-2.

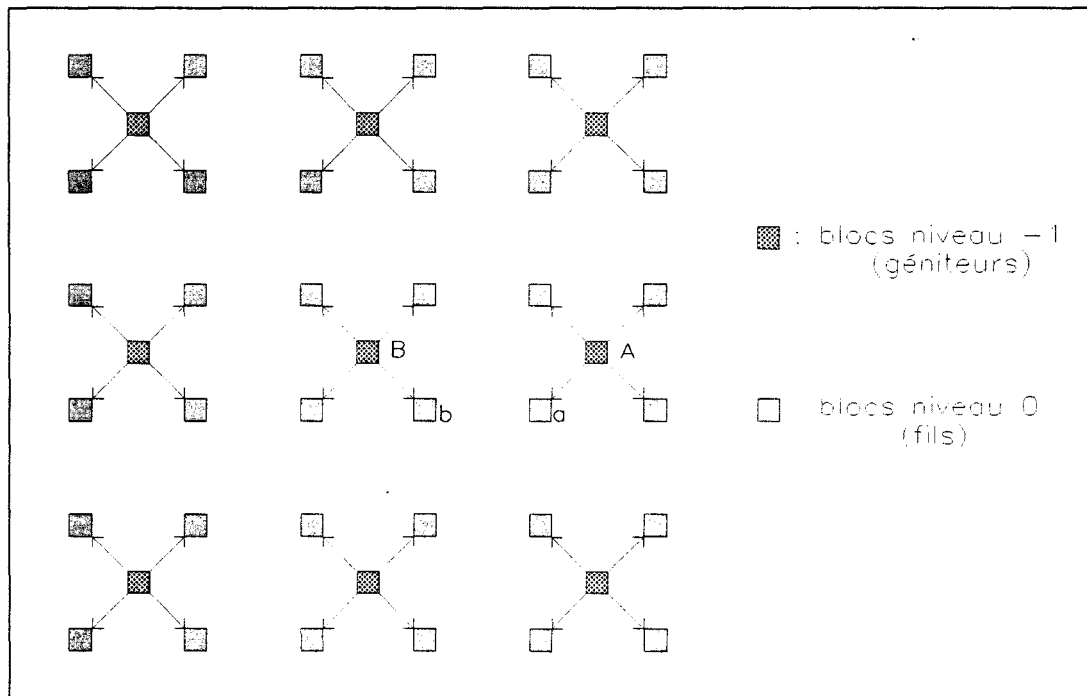


Figure III-2. Schéma 1 géniteur/4 fils.

Ce schéma nous donne la possibilité d'utiliser deux méthodes différentes d'affectation du vecteur d'initialisation.

La première consiste à affecter pour chaque bloc fils, le vecteur de son géniteur direct d'après la figure III-2, suivant le schéma 1 géniteur/4 fils. Les travaux d'Anandan [ANA84] ont montré les limites de cette approche pour les

discontinuités du champ comme c'est le cas en présence d'occlusions. Du point de vue du calcul des vecteurs de mouvement, ce phénomène est catastrophique, puisqu'il n'y a évidemment pas de relation à priori entre le déplacement des deux objets impliqués. De plus, si les blocs a et b (figure III-2) appartiennent au même objet en mouvement, il faut bien sûr qu'ils soient affectés du même vecteur $D_0(a)$, au cas où $D_{-1}(A)$ est différent de $D_{-1}(B)$.

Pour remédier à ce problème, plusieurs propositions ont été faites par différents chercheurs, parmi lesquelles on peut citer le schéma trois géniteurs/douze fils que nous avons retenu. Cela signifie que le géniteur de chaque bloc du niveau 0 sera choisi parmi trois possibles du niveau -1. Nous explicitons cette méthode à l'aide du schéma de la figure III-3.

Les trois géniteurs potentiels du bloc a sont regroupés dans la zone délimitée par les pointillés. Autrement dit, pour un bloc fils donné, on considère les trois géniteurs géométriquement les plus proches, au sens de la distance euclidienne. Si l'on prend en compte le bloc D, les résultats ne montrent pas d'amélioration sensible justifiant la complexité supplémentaire d'algorithme qui en découlerait. Ce bloc n'est donc pas retenu pour le choix des vecteurs d'initialisation.

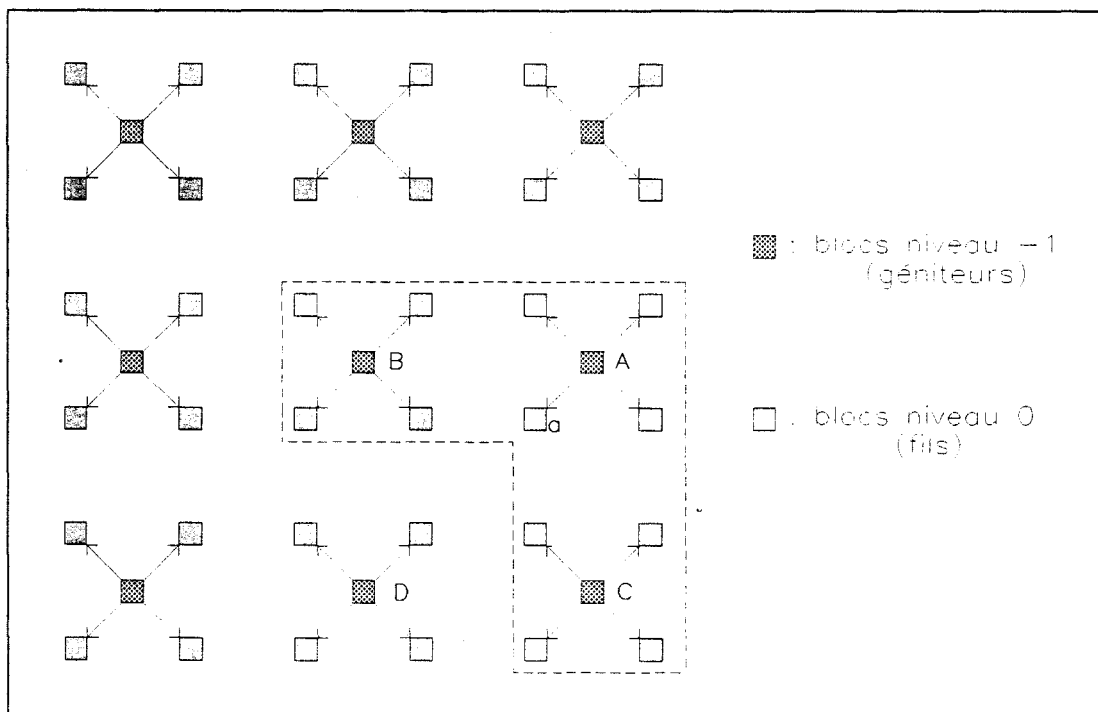


Figure III-3. Schéma 3 géniteurs/12 fils.

Soit à déterminer le mouvement du bloc "a" du niveau 0. Il faut, dans un premier temps, identifier ses trois géniteurs possibles afin d'obtenir D_{-1} . Les positions définies dans la figure III-3 nous amènent à considérer quatre cas selon l'emplacement de "a". Si l'on désigne par i et j les indices, respectivement de ligne et de colonne, des centres des blocs fils, les quatre cas envisageables sont résumés dans le tableau III-2.

j	pair	impair
i		
pair	1	2
impair	3	4

Tableau III-2. Combinaisons d'indices possibles pour la recherche des géniteurs.

Dans chaque cas, les blocs géniteurs seront ceux repérés par les couples de coordonnées (ligne,colonne) ci-après, au niveau -1.

$$1 - \left[\frac{i}{2}, \frac{j}{2} \right] \left[\frac{i}{2} - 1, \frac{j}{2} \right] \left[\frac{i}{2}, \frac{j}{2} - 1 \right] \quad (3.3)$$

$$2 - \left[\frac{i}{2}, \frac{j-1}{2} \right] \left[\frac{i}{2}, \frac{j+1}{2} \right] \left[\frac{i}{2} - 1, \frac{j-1}{2} \right] \quad (3.4)$$

$$3 - \left[\frac{i-1}{2}, \frac{j}{2} \right] \left[\frac{i+1}{2}, \frac{j}{2} \right] \left[\frac{i-1}{2}, \frac{j}{2} - 1 \right] \quad (3.5)$$

$$4 - \left[\frac{i-1}{2}, \frac{j-1}{2} \right] \left[\frac{i-1}{2}, \frac{j+1}{2} \right] \left[\frac{i+1}{2}, \frac{j-1}{2} \right] \quad (3.6)$$

On affectera alors au bloc "a" le vecteur D_{-1} du géniteur qui conduit à la plus petite MSE.

La seconde étape consiste à calculer le terme correctif dD de la formule d'interpolation. Pour ce faire, à partir de la position définie par le premier terme de (3.2), on effectue une seconde recherche par block matching 4x4. Nous avons choisi de conserver la même taille de bloc entre les différents niveaux de résolution, ce qui amène bien comme nous l'avons dit, à examiner des zones de l'image variant dans un rapport deux en dimension horizontale et verticale à chaque itération du processus.

III.3 - Caractéristiques de la séquence de travail

La séquence d'images étudiée a été réalisée en laboratoire. Nous avons fait une acquisition d'images de type visiophone, visioconférence, c'est-à-dire comportant un fond fixe et très peu de mouvements des sujets. En outre, ces mouvements sont de faible amplitude. Nous avons également pris le soin de créer une zone où l'on a deux mouvements de sens opposés pour pouvoir tester le comportement de notre logiciel sur les problèmes d'occlusion.

Cette séquence comporte dix images noir et blanc entrelacées. Les résultats que nous avons obtenus sur l'ensemble de la séquence sont présentés ici à l'aide de deux paires d'images qui en sont extraites. Il s'agit des images 1 et 2, puis 6 et 7.

Elles apparaissent sur les figures III-4b et III-5b, conformément aux dispositions des figures III-4a et III-5a respectivement. Dans le coin inférieur droit nous avons placé l'image de différence directe FD, pour laquelle nous avons calculé l'entropie H selon la formule :

$$H = - \sum_i p_i \log_2 p_i \quad (3.7)$$

où p_i désigne la probabilité d'apparition de la valeur i .

Le but recherché étant d'envoyer à la place de l'image courante brute, l'image de différence, nous indiquons également la valeur du rapport signal sur bruit défini dans [JAI81] par la relation :

$$SNR = 10 \log \frac{\text{Puissance de l'image brute}}{\text{Variance de l'image erreur}} \text{ (dB)} \quad (3.8)$$

Cette grandeur permet, mieux que le critère de l'entropie, de faire des comparaisons entre différentes séquences.

image 1	image 2
	FD H=5,446 bpp SNR=10,11dB

Figure III-4a.

image 6	image 7
	FD H=6,146 bpp SNR=5,38dB

Figure III-5a.

III.4 - Comparaison entre monorésolution et multirésolution

Pour pouvoir situer nos résultats par rapport au schéma classique en monorésolution, nous avons commencé par une première série de mesures en procédant comme suit.

Dans un premier temps, nous effectuons sur deux images k et $k-1$ à la résolution zéro, une estimation du mouvement par une technique de block matching 4×4 . Le champ de vecteurs ainsi obtenu sert à compenser directement l'image $k-1$. Ces opérations sont résumées par la figure III-6a.

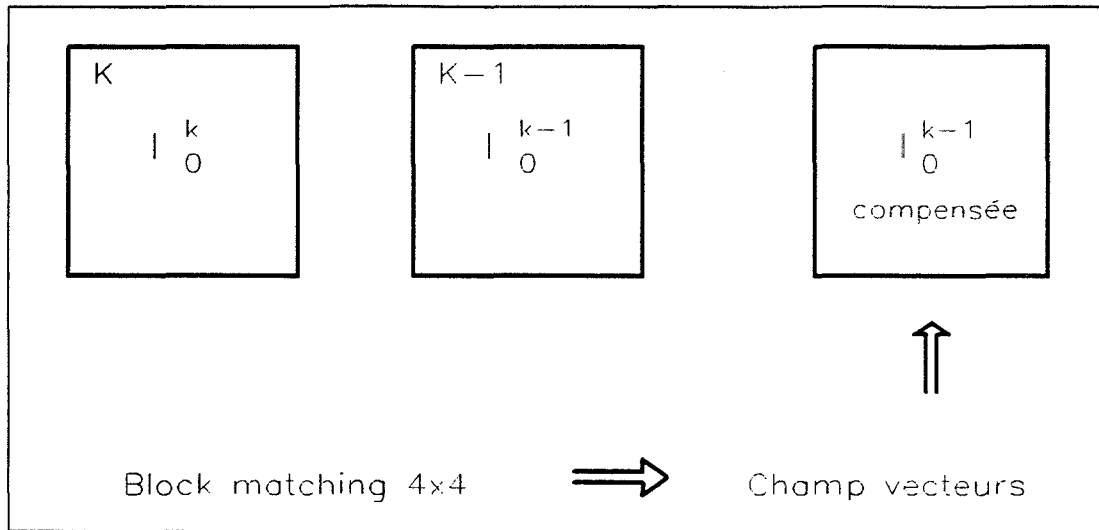


Figure III-6a.

Les résultats sont présentés sur l'exemple des images 6 et 7, sur les figures III-6b et III-6c. Nous avons disposé côte à côte les images de différence directe (FD) et de différence compensée (DFD). On constate que l'entropie passe de 6,146 bpp à 4,734 bpp.

image 6	image 7
FD	DFD
H=6,146 bpp	H=4,734 bpp
SNR=5,38dB	SNR=13,99dB

Figure III-6b.

Pour la série de mesures suivante, nous opérons selon le schéma de la figure III-7a.



Figure III-4b.



Figure III-5b.

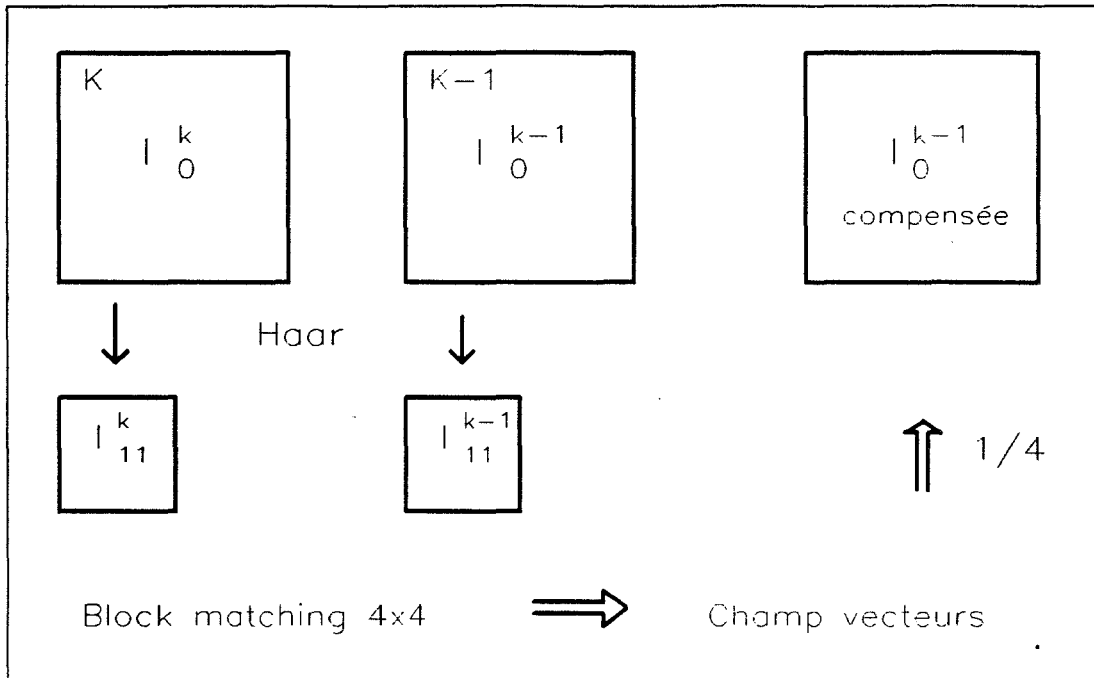


Figure III-7a.

Nous effectuons la décomposition des images k et $k-1$ au niveau -1 en utilisant le filtre de Haar (H_a). Nous calculons les vecteurs de mouvement à la résolution -1 par block matching 4×4 comme précédemment. Le champ de vecteurs du niveau 0 est obtenu du précédent par la formule (3.2), et en ayant recours au schéma 1 géniteur/4 fils. On se contente donc d'affecter le même vecteur à tous les blocs fils issus du même bloc géniteur. On aboutit alors aux images de la figure III-7c, disposées selon la figure III-7b. L'entropie de la DFD tombe cette fois-ci à 4,695 bpp.

image 6	image 7
FD	DFD
H=6,146 bpp	H=4,695 bpp
SNR=5,38dB	SNR=16,59dB

Figure III-7b.



Figure III-6c.



Figure III-7c.

Nous constatons que la diminution de l'entropie n'est pas très importante. Cependant, sur l'image de la DFD nous voyons apparaître, dans les zones correspondant aux occlusions, des pavés d'intensité relativement grande. Cela confirme le fait que le schéma d'interpolation utilisé n'est pas très efficace pour traiter les discontinuités de champ et les problèmes d'occlusion. Nous allons donc employer un autre algorithme visant à pallier ce défaut du procédé de la figure III-7a.

III.5 - Problème des occlusions

Nous modifions légèrement le schéma de la figure III-7a pour obtenir celui de la figure III-8a. La différence se situe au niveau de l'interpolation des vecteurs où le schéma 3 géniteurs/12 fils remplace celui du 1 géniteur/4 fils.

Pour chaque bloc fils nous testons les trois vecteurs d'initialisation possibles provenant des trois géniteurs potentiels. Celui qui conduit à la MSE la plus faible est alors retenu. En procédant de cette façon nous constatons que dans la plupart des cas, le vecteur choisi correspond au géniteur direct du bloc (selon le schéma 1 géniteur/4 fils), à l'exception principalement des zones d'occlusion où les vecteurs "d'appoint" sont fréquemment retenus. Il faut toutefois noter que l'importance quantitative de ces vecteurs est fonction de la taille des blocs et des positions des objets par rapport à ces blocs (exemples : bloc à cheval sur une frontière ou se situant entièrement d'un côté). Les résultats sont montrés par les figures III-8b et III-8c.

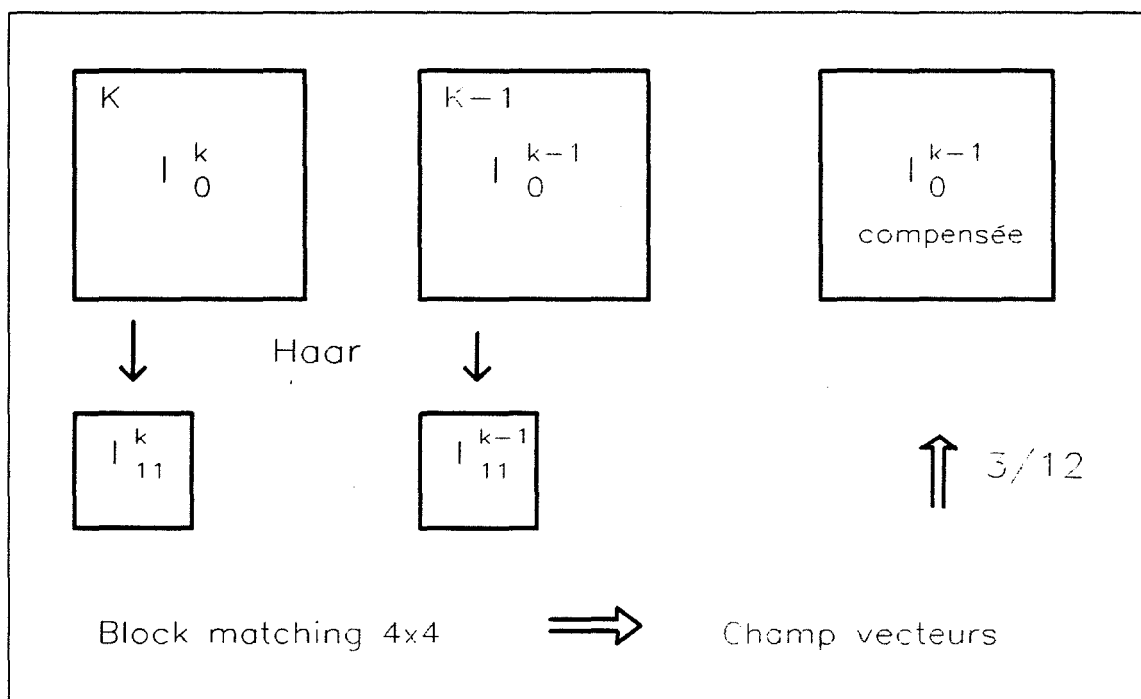


Figure III-8a.

image 6	image 7
FD	DFD
H=6,146 bpp	H=4,5103 bpp
SNR=5,38dB	SNR=17,73dB

Figure III-8b.



Figure III-8c.

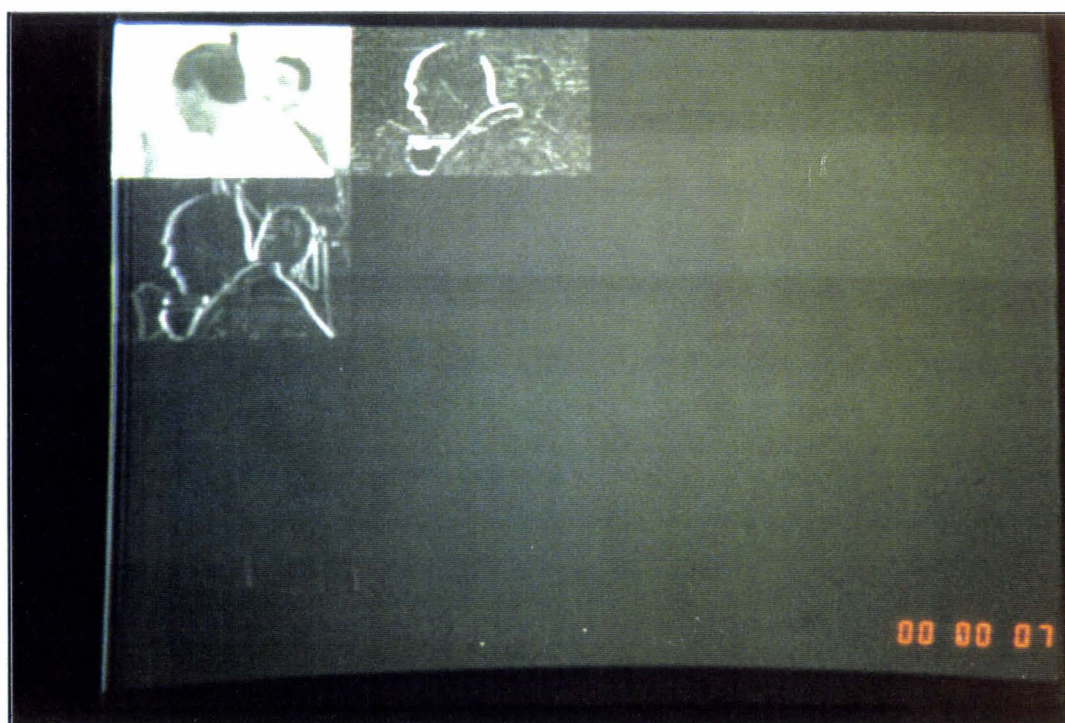


Figure III-10b.

On constate sur l'image de la DFD, que les pavés correspondant aux zones d'occlusion ont disparu. Cela entraîne une diminution de l'entropie qui passe à 4,5103 bpp.

Jusque là, nous n'avons utilisé que les sous images approximations de l'image de départ, sans tenir compte de l'existence des trois autres, composées des détails. Voyons maintenant comment on peut intégrer ces données.

III.6 - Intégration des pyramides de détails

L'application de l'algorithme de Mallat sur une image $N \times N$ pixels du niveau 0 engendre quatre sous images au niveau -1, ordonnées comme l'indiquent la figure III-9.

	N/2	N/2
N/2	image 11	image 12
N/2	image 13	image 14

Figure III-9.

L'image 11 correspond à l'approximation de l'image 0 (niveau 0) à la résolution -1, 12 aux détails horizontaux, 13 aux détails verticaux et 14 aux détails dans les deux directions. Dans les paragraphes précédents nous

n'avions travaillé qu'avec l'image 11. Ici, nous procédons selon les étapes de la figure III-10a.

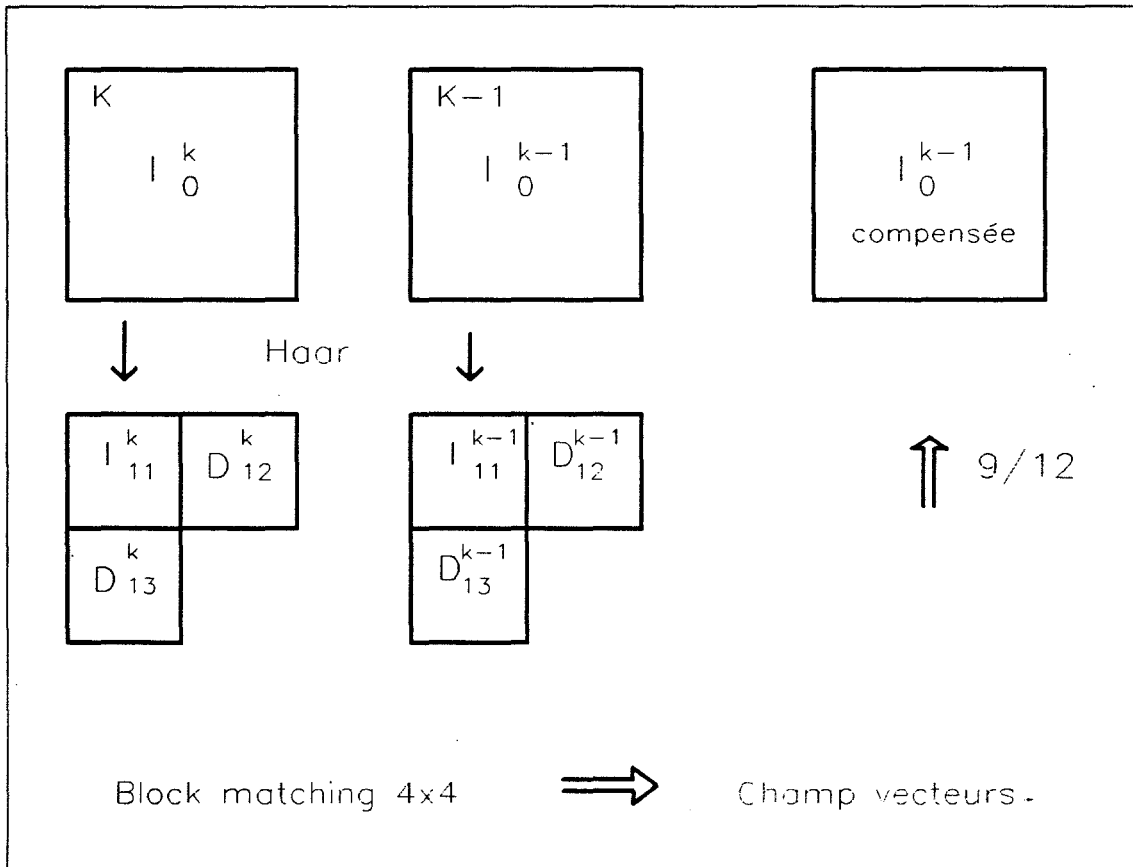


Figure III-10a.

Nous décomposons les images consécutives du niveau 0 sur le niveau -1 avec l'un des filtres, par exemple ici celui de Haar. Sur chaque couple de sous images (I_{11}^k, I_{11}^{k-1}), (D_{12}^k, D_{12}^{k-1}) et (D_{13}^k, D_{13}^{k-1}) nous effectuons une estimation du mouvement par block matching 4x4. Nous obtenons ainsi trois champs de vecteurs qui vont servir à la construction du champ au niveau 0 pour compenser I_0^{k-1} . Pour choisir le vecteur d'initialisation d'un bloc fils du niveau 0, nous utilisons le schéma 3 géniteurs/12 fils. Compte tenu du fait que nous disposons de trois champs de vecteurs, l'utilisation de ce schéma sur l'ensemble de ces champs nous donne le choix entre neuf vecteurs possibles pour un bloc fils donné. Nous testons tour à tour ces neuf vecteurs et nous

retenons celui qui conduit à la MSE la plus faible, sans autre type de contrainte supplémentaire.

Au cours de nos mesures nous avons essayé de construire le champ de vecteurs final par différentes combinaisons linéaires des trois champs, mais il s'est avéré qu'aucune de ces méthodes ne conduit à des résultats sensiblement meilleurs. De même, l'apport des vecteurs générés à l'aide des images 14 n'est pas suffisamment significatif pour que l'on prenne en compte cette source supplémentaire de vecteurs d'initialisation. C'est la raison pour laquelle nous n'utilisons que les trois autres comme l'indique la figure III-10b, qui montre les sous images utilisées, de la décomposition de l'image 6 de la séquence.

Il est à noter que les vecteurs issus de 11 sont le plus souvent retenus. Ceux provenant des images de détails le sont essentiellement dans les zones de contours d'objets en mouvement. On s'attendrait à voir les vecteurs issus de 12 être principalement choisis pour les contours horizontaux en déplacement vertical ainsi que ceux provenant de 13 pour les déplacements horizontaux des contours verticaux. Les mesures que nous avons réalisées montrent cependant que ce n'est pas toujours le cas. Il semble donc que cette déduction ne puisse pas se faire de manière aussi intuitive.

Les résultats auxquels nous aboutissons sont représentés par les figures III-10c et III-10d.

image 6	image 7
FD H=6,146 bpp SNR=5,38dB	DFD H=4,4647 bpp SNR=17,94dB

Figure III-10c.

Nous constatons que l'information contenue dans les sous images de détails contribue bien comme prévu à l'amélioration de la compensation.

Notons que, dans le cas du pel-récuratif, la prise en compte des pyramides de détails a fait l'objet de différents travaux, notamment ceux de Baaziz et Labit [BAA91b]. Ils ont d'abord testé la méthode consistant à lancer trois processus d'estimation sur chacune des sous images de détails et se sont trouvés confrontés à deux inconvénients. D'une part, l'obtention de trois champs de mouvement indépendants entre eux ne permet pas de trouver un opérateur efficace de fusion à posteriori, d'autre part on a une mauvaise performance en termes de coûts opératoires et d'amélioration des estimées. Ils utilisent alors une approche multicontrainte afin de réaliser cette intégration.

Les méthodes multicontraintes consistent à rechercher N équations indépendantes pour la résolution d'un même processus d'estimation. L'approche de Wiener, dont nous avons déjà parlé (cf. 1.2.2.), offre une solution élégante et fiable, utilisant le pel-récuratif. Les N pixels appartenant au voisinage de celui dont on veut estimer le déplacement, sont pris simultanément sur les différentes sous images d'un même niveau de résolution. La figure III-11 [BAA91b] illustre différents choix possibles de voisinage.

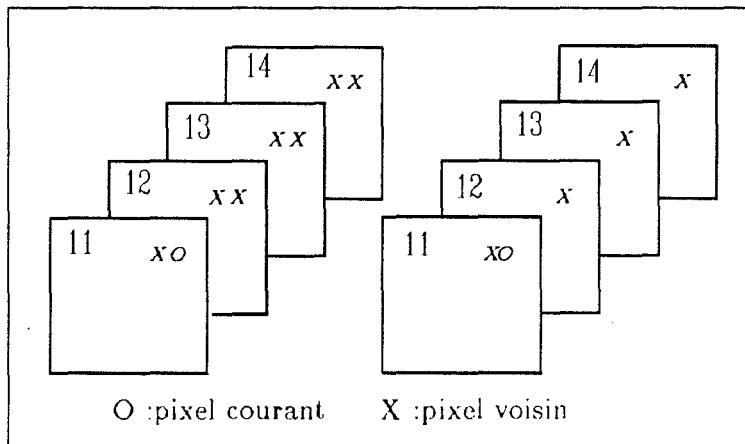


Figure III-11. Différentes configurations de voisinage multicontrainte.

Le champ de mouvement est alors estimé sur la sous image d'approximation 11. Ce champ est utilisé pour la compensation des quatre sous images, dont les DFD correspondantes sont transmises. A la réception, la reconstruction se fait naturellement par le banc de filtres d'ondelettes utilisé à l'émission pour la décomposition.

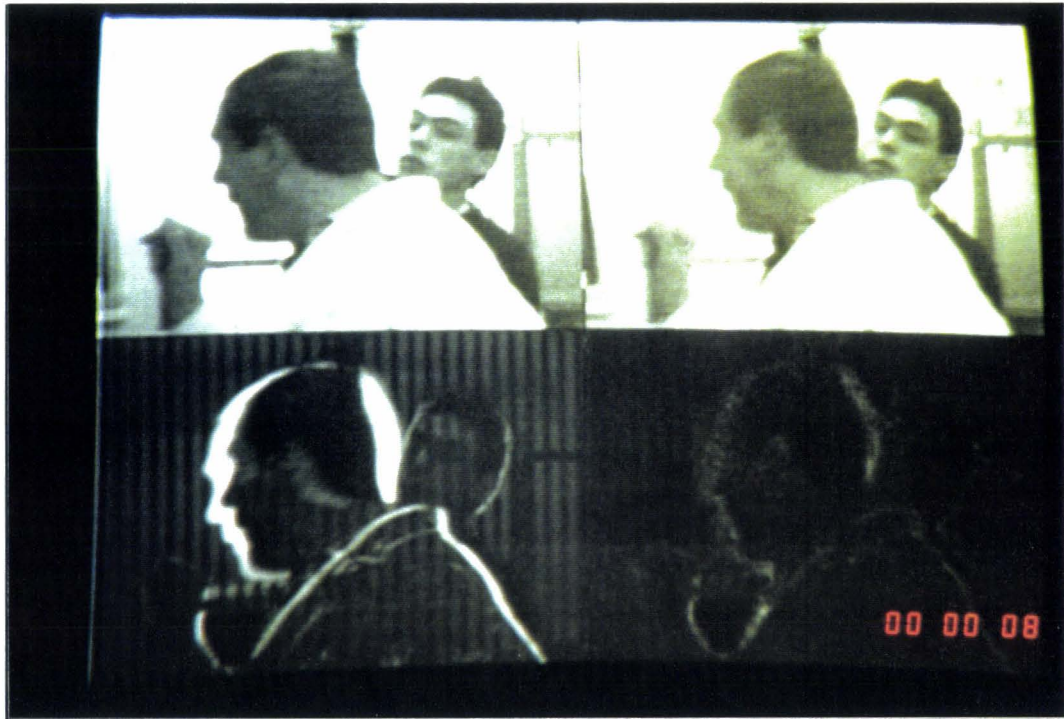


Figure III-10d.



Figure III-15b.

En introduction, nous avons fait allusion au problème de la délocalisation que nous allons aborder dans le paragraphe suivant.

III.7 - Problème de la délocalisation

Pour réaliser cette étude, nous avons appliqué le schéma de compensation sur deux séquences d'images, l'une délocalisée et l'autre non délocalisée. Les algorithmes appliqués sont les suivants.

méthode 1a : images délocalisées

Pour les images délocalisées, nous procédons selon une méthode qui consiste à décomposer dans un premier temps les images au niveau -1 (figure III-12). Ensuite on effectue sur les images I1 un block matching 4x4. Le champ de vecteurs obtenu est interpolé au niveau 0 selon le schéma 1 géniteur/4 fils.

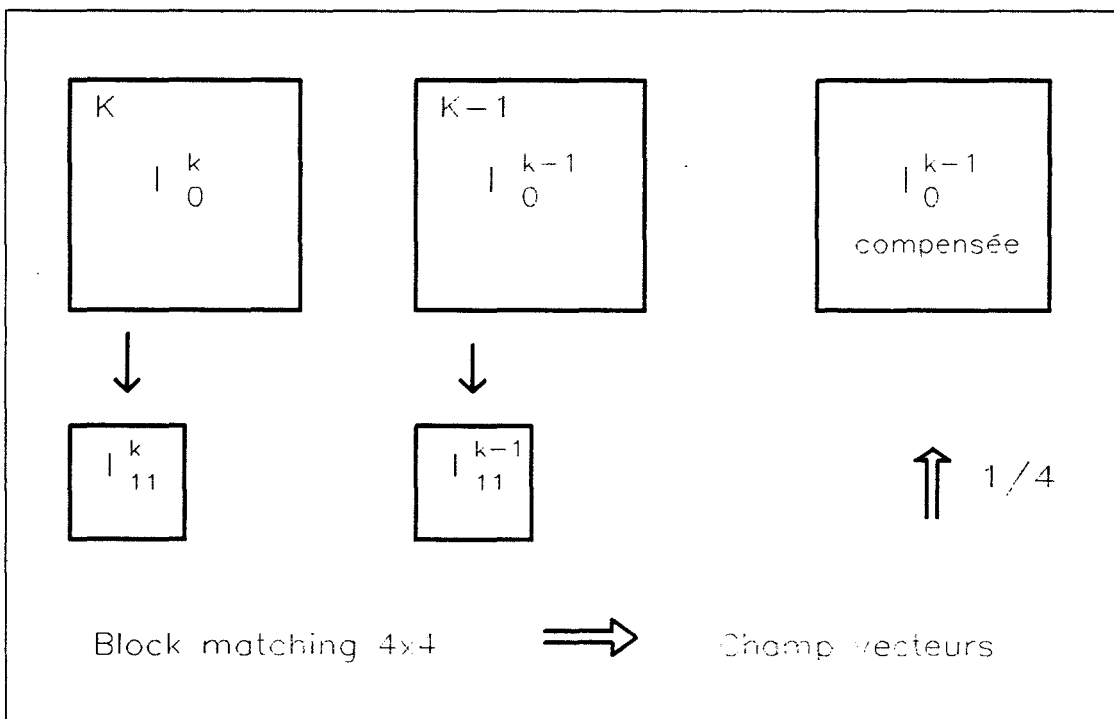


Figure III-12.

méthode 2a : images non délocalisées

Dans ce cas-ci nous décomposons au niveau -1, mais sans opérer de changement d'échelle. Nous obtenons alors des images de mêmes dimensions, mais filtrées passe-bas (figure III-13). Remarquons au passage qu'il faudrait parler plutôt de "pseudo non délocalisation", compte tenu du fait que le filtrage provoque une diminution de la localisation des contours.

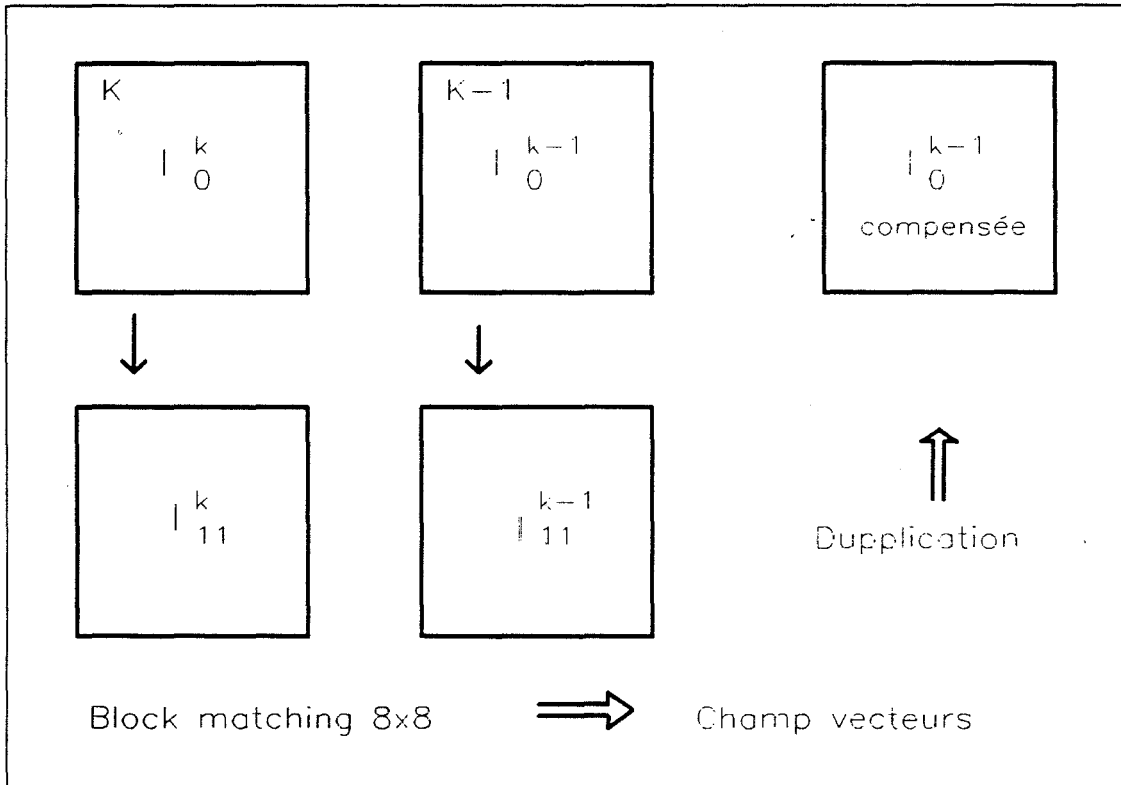


Figure III-13.

Nous opérons sur l'image 11 une recherche par block matching 8×8 pour être homogène à la même étape de la méthode 1a, en tenant compte du non changement d'échelle. Au niveau 0, les vecteurs sont ensuite dupliqués sur les blocs 4×4 issus des 8×8 , toujours dans le souci d'homogénéité.

Nous appliquons ces deux méthodes avec quatre filtres différents, à savoir H_a , F_3 , F_5 et F_9 , en mesurant à chaque fois l'entropie de la DFD obtenue. Les résultats sont rassemblés dans le tableau III-3.

Filtre	méthode 1a	méthode 2a
H _a	4,695	4,7080
F ₃	4,6978	4,746
F ₅	4,707	4,749
F ₉	4,6995	4,751

Tableau III-3. Tableau comparatif des valeurs des entropies (en bpp) de DFD produites par les méthodes 1a et 2a.

On constate qu'avec la méthode 1a, c'est-à-dire lorsqu'il y a délocalisation, l'entropie reste à peu près constante. Par contre, dans le cas de non délocalisation, cette entropie augmente avec la taille du filtre.

Nous avons voulu voir si ce comportement était en relation avec la manière dont les vecteurs étaient interpolés. Pour cela, nous avons procédé à une étude similaire à la précédente, mais en modifiant légèrement les algorithmes, ce qui nous donne :

méthode 1b : images délocalisées

Nous procédons de la même façon que dans la méthode 1a mais au lieu du schéma 1 géniteur/ 4 fils, nous utilisons celui du 3 géniteurs/12 fils.

méthode 2b : images non délocalisées

Là encore, l'algorithme ne diffère de celui de la méthode 2a que par le fait qu'on ne se contente plus de dupliquer les vecteurs. Le vecteur de chaque bloc 4x4 du niveau 0 est recherché selon le schéma 3 géniteurs/12 fils. Le tableau III-4 regroupe les valeurs d'entropie de DFD obtenues.

Filtre	méthode 1b	méthode 2b
H _a	4,5103	4,523
F ₃	4,518	4,550
F ₅	4,5146	4,541
F ₉	4,5201	4,541

Tableau III-4. Tableau comparatif des valeurs des entropies (en bpp) de DFD produites par les méthodes 1b et 2b.

On remarque une fois de plus la constance de l'entropie dans le cas d'images délocalisées, et la variation fonction de la taille du filtre dans l'autre cas. Ceci montre que la façon d'interpoler les vecteurs ne contribue pas à ce phénomène. En outre, dans les deux séries de mesures, a et b, l'entropie est toujours plus faible dans le cas de délocalisation. Ce comportement s'explique à notre avis par le fait qu'en non délocalisé, il y a plus de points à prendre en compte pour minimiser la MSE, ce qui augmente le risque de trouver un minimum local de la fonction MSE au détriment du minimum vrai de cette fonction. En élargissant la taille du filtre, on accroît encore le nombre de points de l'image de départ impliqués dans le calcul, d'où la dégradation des performances, puisque l'on augmente ainsi la zone de recouvrement entre objets affectés de mouvements différents. Ces zones, qui sont des aberrations du point de vue du mouvement, n'étant pas décimées par le sous échantillonnage comme dans le cas de délocalisation, sont sans doute à l'origine de difficultés rencontrées dans cette estimation.

Il résulte de ceci que l'on a intérêt en général à effectuer un changement d'échelle. En plus du gain en temps de calcul (image réduite), on est apparemment moins tributaire de la taille du filtre.

III.8 - Influence de la taille des filtres

Dans le but de caractériser l'influence de la taille des filtres, nous proposons un schéma utilisant trois filtres d'horizons spatiaux différents, placés en parallèle. Ces trois filtres sont H_a , F_3 , et F_5 (figure III-14).

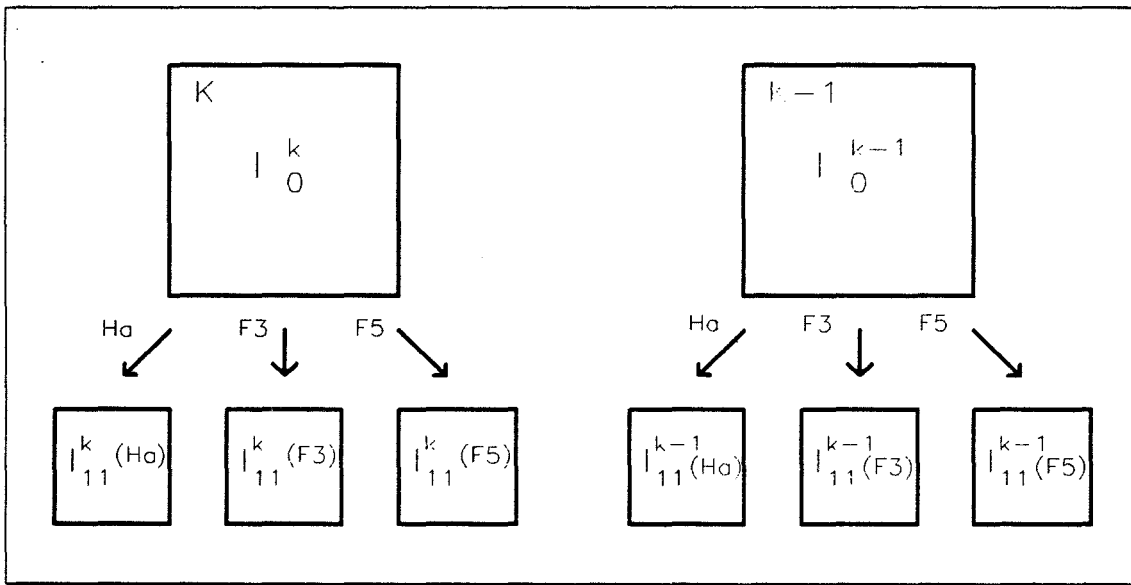


Figure III-14.

La paire d'images consécutives est décomposée sur un niveau de résolution avec chacun des filtres. Pour les raisons citées précédemment, nous nous limitons au niveau -1. On obtient donc trois sous images 11 .

Nous estimons par un block matching 4×4 les déplacements entre les sous images 11 issues du filtre de Haar, celles issues du filtre F_3 et de même pour le filtre F_5 .

Pour affecter les vecteurs d'initialisation aux blocs du niveau 0, nous utilisons le schéma 1 géniteur/4 fils en retenant le meilleur vecteur, au sens de

la plus faible erreur quadratique moyenne, issu des trois champs de vecteurs ainsi calculés.

Nous présentons les résultats de cette étude sur la paire d'images 1 et 2, affectée de cinq niveaux de bruit différents, décroissant de 1 à 5.

Les figures III-15a et III-15b nous donnent les images de différence directe, FD, entre l'image 1 et l'image 2, pour quatre des cinq points de mesure.

FD 1 H=5,446 bpp	FD 2 H=4,342 bpp
FD 3 H=4,139 bpp	FD 4 H=4,031 bpp

Figure III-15a.

Nous avons mesuré l'entropie de la DFD obtenue par chaque filtre séparément, et celle obtenue par notre méthode "multifiltres". Les résultats auxquels nous aboutissons pour le niveau de bruit le plus élevé, c'est-à-dire le niveau 1, sont reportés dans le tableau III-5, et on peut observer les images différence et leur caractéristiques sur les figures III-16a et III-16b.

Filtre	Entropie en bpp
H _a	4,790
F ₃	4,7853
F ₅	4,7974
multifiltres	4,7186

FD : H = 5,446 bpp

Tableau III-5. Résultats pour le niveau 1.

H_a $H=4,79$ bpp $SNR=16,54$ dB	F_3 $H=4,785$ bpp $SNR=16,5$ dB
F_5 $H=4,797$ bpp $SNR=16,44$ dB	multifiltres $H=4,71$ bpp $SNR=16,93$ dB

Figure III-16a.

On s'aperçoit que les pavés dûs au problème d'occlusion persistent, ce qui caractérise le fait que cette manière de procéder n'est pas équivalente à l'application du schéma 3 géniteurs/12 fils, bien que dans les deux cas nous disposions avant de calculer dD , de trois vecteurs d'initialisation pour débiter la recherche de ce terme.

Nous avons repris la même mesure en intégrant le schéma 3 géniteurs/12 fils au lieu du précédent. On aboutit alors à une entropie $H = 4,639$ bpp ($SNR = 18,09$ dB). La valeur du rapport signal sur bruit nous fait constater que l'application de cet algorithme a une efficacité comparable à la prise en compte des pyramides de détails comme nous l'avons effectué dans le paragraphe III.8, puisque dans les deux cas nous obtenons une valeur d'environ 18 dB.

Les résultats produits par les autres mesures sont rassemblés dans les tableaux III-6 à III-9.

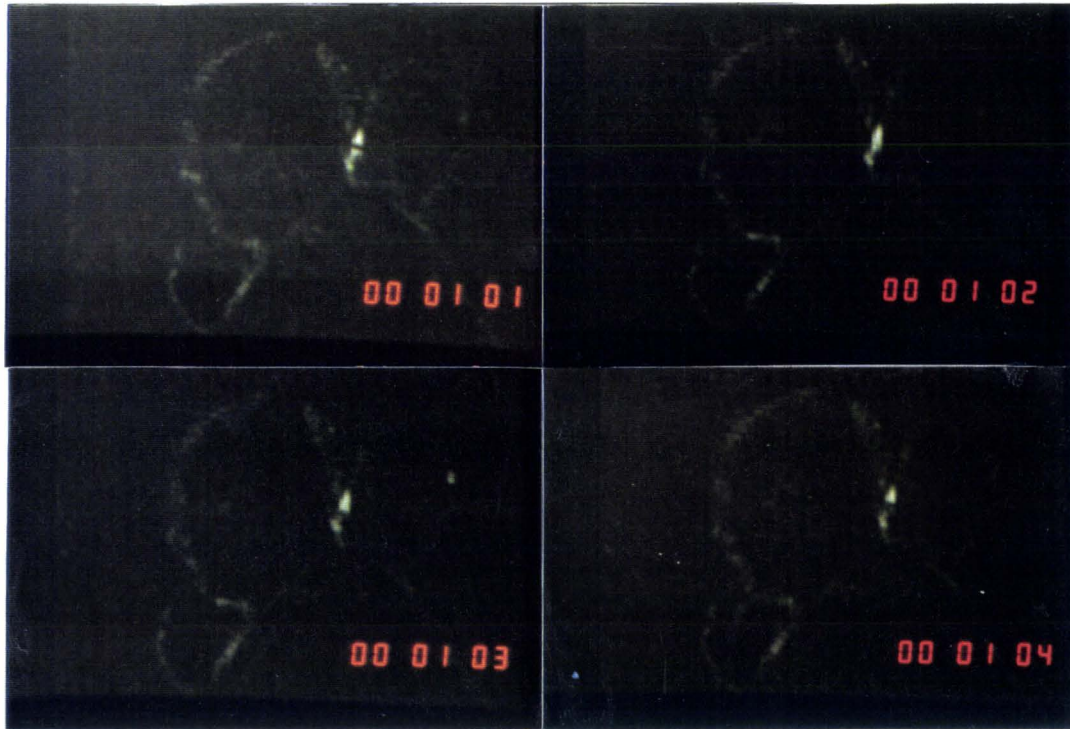


Figure III-16b.

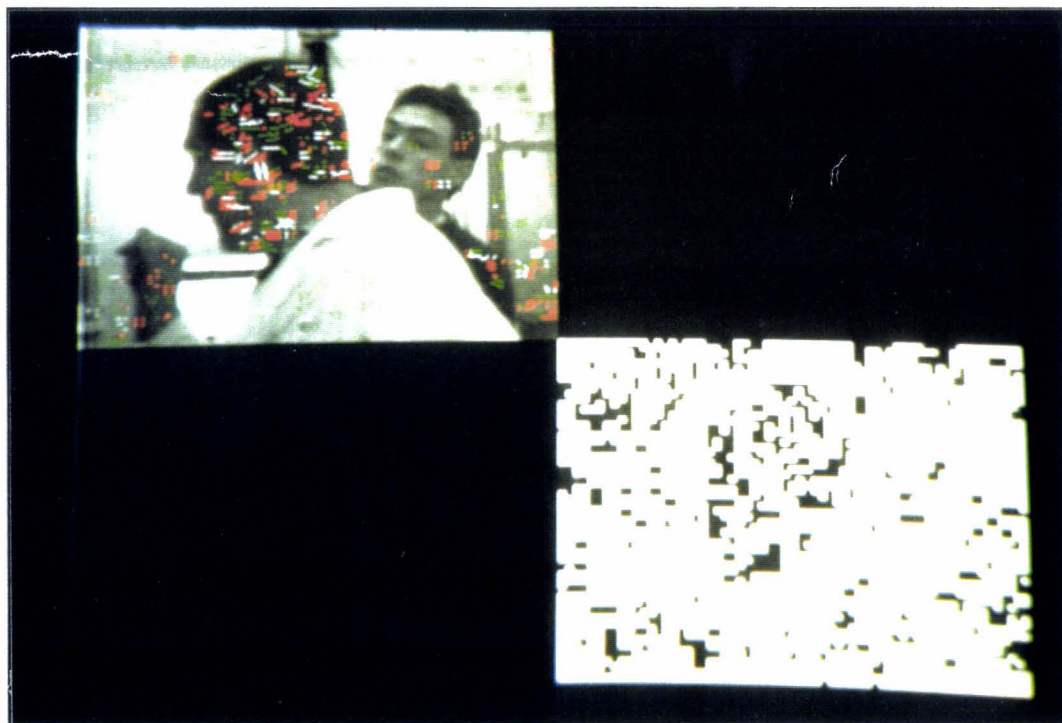


Figure III-18.

Filtre	Entropie en bpp
H _a	3,329
F ₃	3,3369
F ₅	3,3343
multifiltres	3,2950

FD : H = 4,3425 bpp

Tableau III-6. Résultats pour le niveau 2.

Filtre	Entropie en bpp
H _a	2,9852
F ₃	2,9846
F ₅	2,9828
multifiltres	2,9623

FD : H = 4,139 bpp

Tableau III-7. Résultats pour le niveau 3.

Filtre	Entropie en bpp
H _a	2,807
F ₃	2,8074
F ₅	2,8098
multifiltres	2,7787

FD : H = 4,0318 bpp

Tableau III-8. Résultats pour le niveau 4.

Filtre	Entropie en bpp
H _a	2,673
F ₃	2,6723
F ₅	2,666
multifiltres	2,6504

FD : H = 3,9578 bpp

Tableau III-9. Résultats pour le niveau 5.

Afin de présenter nos résultats sous une forme plus synthétique, nous définissons un gain G de compression de la manière suivante:

Soit H₁ l'entropie donnée par la méthode multifiltres, et H₂ l'entropie la plus faible obtenue avec l'un des filtres séparément. Le gain G est calculé par l'expression :

$$G = \frac{H_2 - H_1}{H_2} \quad (3.9)$$

Nous traçons alors l'évolution de ce gain en fonction de l'entropie de la FD entre les images 1 et 2 de départ. Nous obtenons la courbe de la figure III-17.

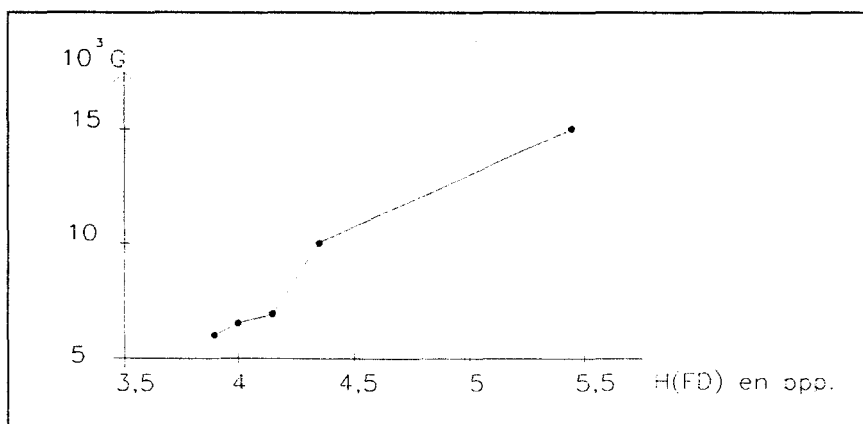


Figure III-17. Gain de compression en fonction de H(FD).

L'allure de cette courbe permet de conclure à l'intérêt de notre méthode dans le cas de séquences à haut niveau de bruit. Inversement, lorsque le niveau de bruit devient peu important, l'utilisation de cette technique vis à vis de celle d'un filtre unique ne semble plus se justifier.

Afin d'expliquer l'évolution de cette courbe en fonction du niveau de bruit, nous avons étudié la répartition de l'origine des vecteurs servant à construire le champ de vecteurs final.

Pour cela, nous avons repéré les vecteurs originaux fournis par chaque filtre (H_a , F_3 et F_5) en fonction du niveau de bruit. Un vecteur est déclaré original pour le filtre étudié s'il est retenu comme vecteur d'initialisation dans l'étape suivante de calcul, et si aucun des deux autres filtres n'a fourni le même vecteur comme candidat à l'initialisation. Ces vecteurs permettent alors de caractériser la contribution propre de chaque filtre dans la diminution de l'entropie de la DFD, vis à vis du mode de fonctionnement avec un filtre unique.

Les résultats de cette étude sont présentés, toujours sur l'exemple de la paire d'images 1 et 2 pour différents niveaux de bruit, repérés par la nomenclature niveau 1 à niveau 5.

Les figures III-18 à III-20 montrent les champs de vecteurs originaux repérés pour chaque filtre, avec la convention suivante:

Filtre H_a $\Leftrightarrow$ vecteurs de couleur rouge.

Filtre F_3 $\Leftrightarrow$ vecteurs de couleur blanche.

Filtre F_5 $\Leftrightarrow$ vecteurs de couleur verte.

Nous faisons également figurer sur ces photos, dans le coin inférieur droit, les blocs de l'image affectés par ces vecteurs originaux.

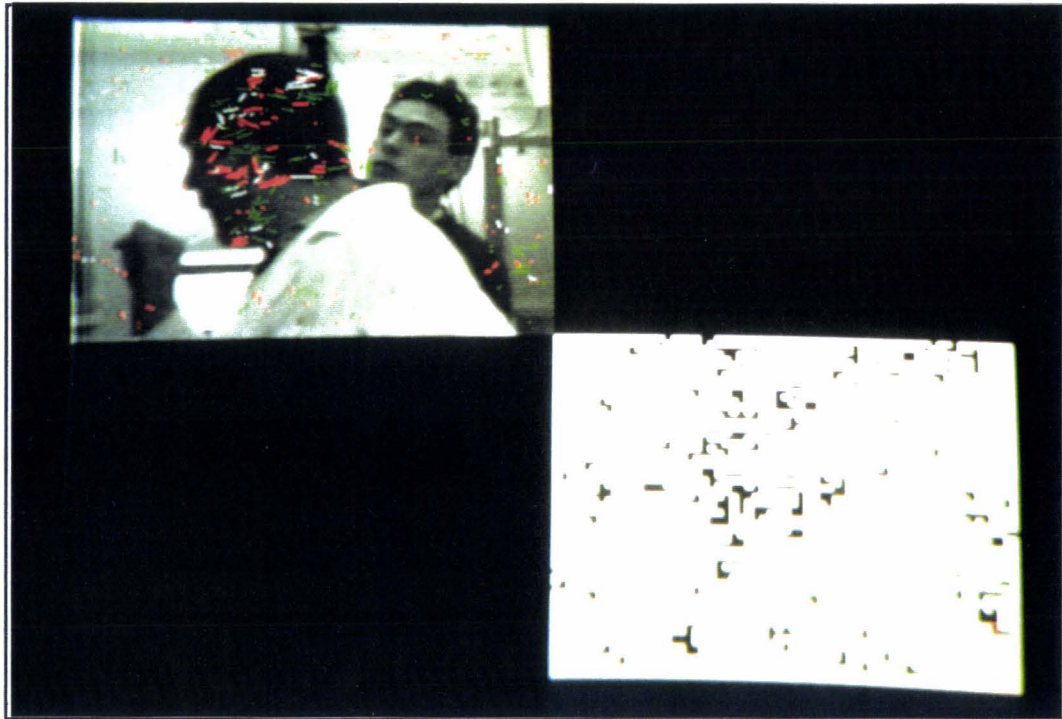


Figure III-19.



Figure III-20.

Le tableau III-10 présente la répartition quantitative de ces vecteurs originaux par filtre, ainsi que le nombre total de ces vecteurs par niveau de bruit.

	Niveau 1	Niveau 2	Niveau 3	Niveau 4	Niveau 5
H _a	333	180	152	109	81
F ₃	213	78	71	56	78
F ₅	303	169	106	121	112
Total	849	427	329	286	271

Tableau III-10. Répartition des vecteurs originaux suivant les trois filtres.

Ces mesures amènent les remarques suivantes:

Le nombre total de vecteurs originaux diminue avec le niveau de bruit. Ce résultat, au demeurant compréhensible, permet d'expliquer l'allure de la courbe de gain G, donnée à la figure III-17, puisqu'il est évident que moins notre système va générer de vecteurs originaux, et moins on aura de différence entre l'utilisation d'un filtre unique et celle de trois filtres en parallèle. C'est ce que traduit le fait que le gain tend vers zéro.

Les variations comparées des valeurs de G et du nombre total de vecteurs originaux fournis par le système, montrent que la baisse de gain constatée lorsque le niveau de bruit diminue, est moins importante que celle du nombre de vecteurs originaux (tout au moins dans le cas des séquences à niveau de bruit non négligeable). Cela tend à prouver que le gain que nous obtenons vis à vis de la méthode monofiltre, n'est pas uniquement dû au fait que l'on possède au départ du calcul trois vecteurs d'initialisation au lieu d'un seul, mais que les vecteurs ainsi générés sont pertinents.

La troisième remarque consiste à noter que ce n'est pas toujours le même filtre qui fournit le plus de vecteurs originaux : H_a pour les niveaux 1, 2 et 3; F₅ pour les niveaux 4 et 5.

Afin de préciser cette notion, nous avons effectué le même type de mesure, mais cette fois en ne conservant à chaque fois que deux filtres pris parmi les trois.

Les tableaux III-11, 12 et 13 donnent la répartition quantitative des vecteurs originaux par filtre pour les trois couples suivants : (H_a, F_3) , (H_a, F_5) et (F_3, F_5) .

	Niveau 1	Niveau 2	Niveau 3	Niveau 4	Niveau 5
H_a	456	268	221	184	162
F_3	518	244	222	162	169
Total	974	512	443	346	331

Tableau III-11. Répartition des vecteurs originaux pour H_a et F_3 .

	Niveau 1	Niveau 2	Niveau 3	Niveau 4	Niveau 5
H_a	643	346	249	208	199
F_5	600	322	258	228	206
Total	1243	668	507	436	405

Tableau III-12. Répartition des vecteurs originaux pour H_a et F_5 .

	Niveau 1	Niveau 2	Niveau 3	Niveau 4	Niveau 5
F_3	519	243	176	156	190
F_5	436	253	173	194	185
Total	955	496	349	350	375

Tableau III-13. Répartition des vecteurs originaux pour F_3 et F_5 .

La lecture de ces résultats montre qu'il n'est pas possible de dissocier vraiment l'efficacité des trois filtres en fonction du niveau de bruit. En effet, plus un filtre sera d'horizon spatial élevé, plus son action de lissage du bruit sera importante, mais plus il créera localement pour des objets affectés de mouvements différents, des zones de recouvrement qui sont sans fondement du point de vue du mouvement.

La suite de l'étude consisterait donc à essayer de dégager un critère visant à déterminer la ou les relations existant entre les paramètres dont dépendent l'efficacité d'un filtre donné à un endroit précis de l'image. Au nombre de ces paramètres locaux nous pouvons déjà citer le niveau de bruit, la présence de contours d'objets en déplacement, la taille et la position des blocs de recherche par rapport à ces contours.

CONCLUSION

CONCLUSION

L'étude que nous avons menée montre l'intérêt de l'utilisation d'une technique d'estimation de mouvement hiérarchique possédant plusieurs vecteurs d'initialisation; ceux-ci étant générés à l'aide de décomposition pyramidale (de type ondelettes) utilisant différents filtres de longueur variable.

Cette méthode permet d'obtenir un gain de compression d'autant plus intéressant que le niveau de bruit est important dans la séquence. Elle ne résout apparemment pas le problème des occlusions comme peut le faire une méthode se servant des champs de vecteurs issus des images détails de la décomposition en ondelettes. Ces méthodes sont donc complémentaires.

Si d'un point de vue technique on envisage l'emploi de cette méthode dans un schéma du type de celui illustré par la figure I-9, l'utilisation de trois filtres de décomposition au lieu d'un seul, amène une complication de l'architecture à réaliser. Il faut toutefois noter que l'on ne pénalise pas la vitesse d'exécution du système puisque les trois décompositions et les calculs des trois vecteurs d'initialisation peuvent se faire en parallèle.

On peut également citer un certain nombre de questions qui restent ouvertes.

Quelle est l'influence du groupe de filtres choisis pour réaliser les décompositions ?. Cette question qui pose le problème général du choix d'une (ou plusieurs) bases d'ondelettes pour une application donnée, ne semble pas avoir eu de réponses, tout au moins complète, à ce jour.

On remarque aussi que lorsque les filtres sont utilisés séparément, ils conduisent à une entropie sensiblement identique. Cela signifie qu'ils sont globalement équivalents et qu'il n'y en a pas un qui soit meilleur que l'autre.

Par contre, le fait que l'application de notre méthode multifiltre aboutisse à une amélioration de la compensation, révèle le fait qu'ils ne sont pas tous pris en défaut aux mêmes endroits de l'image. La suite logique de ce travail serait de tenter de répondre à la question, pourquoi tel filtre est-il efficace à tel endroit.

On pourrait également se poser la question de savoir quelle serait l'influence si l'on décomposait sur plusieurs niveaux. Notre étude nous permet de penser que l'impact de la longueur des filtres serait relativement faible, puisque plus on descend en résolution, moins notre système va générer de vecteurs originaux.

Par contre, elle ne nous permet pas de prévoir jusqu'à quel niveau il faut descendre, avant que l'influence ne se fasse plus sentir. Cependant, on peut d'ores et déjà pressentir que ce nombre de niveaux limite sera relatif à la taille de l'image de départ et de son niveau de résolution. Plus on partira d'une image de taille importante et contenant beaucoup de détails, et plus il faudra de niveaux de décomposition. On pourrait alors envisager, en conjonction avec le niveau de résolution, la donnée d'un rapport signal sur bruit limite au dessus duquel l'utilisation du système multifiltre se justifierait pleinement.

ANNEXE

ANNEXE

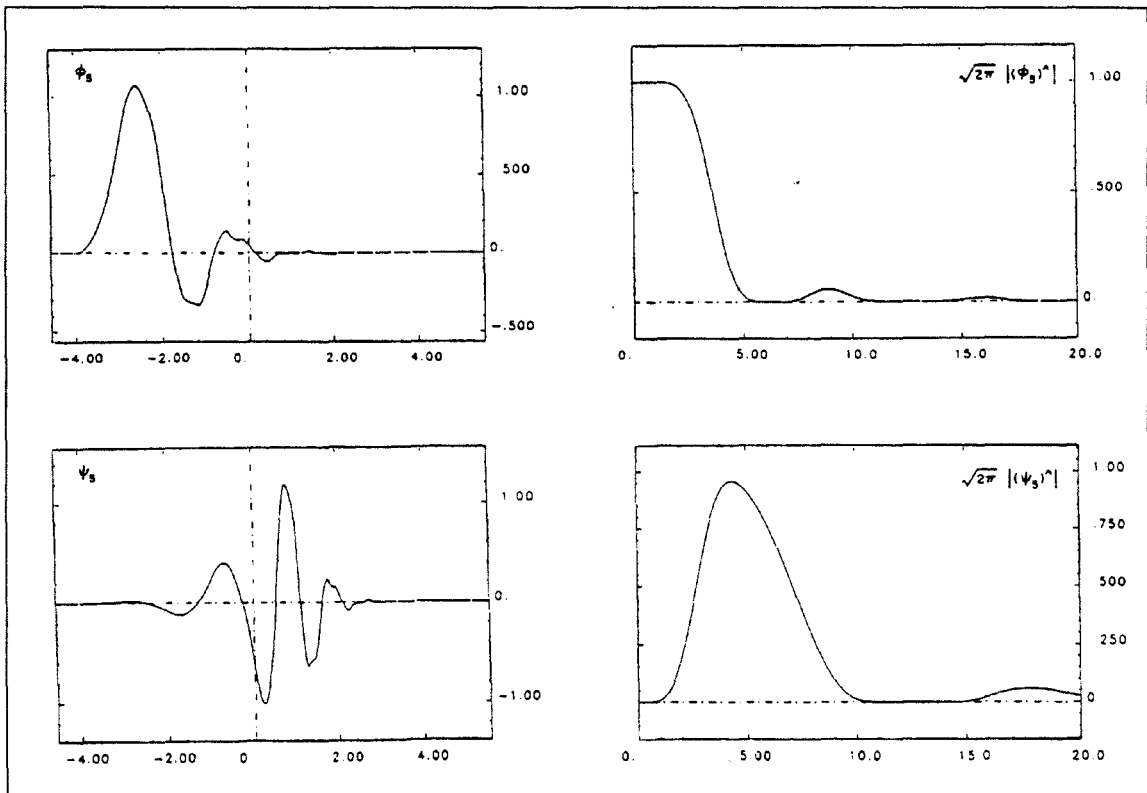
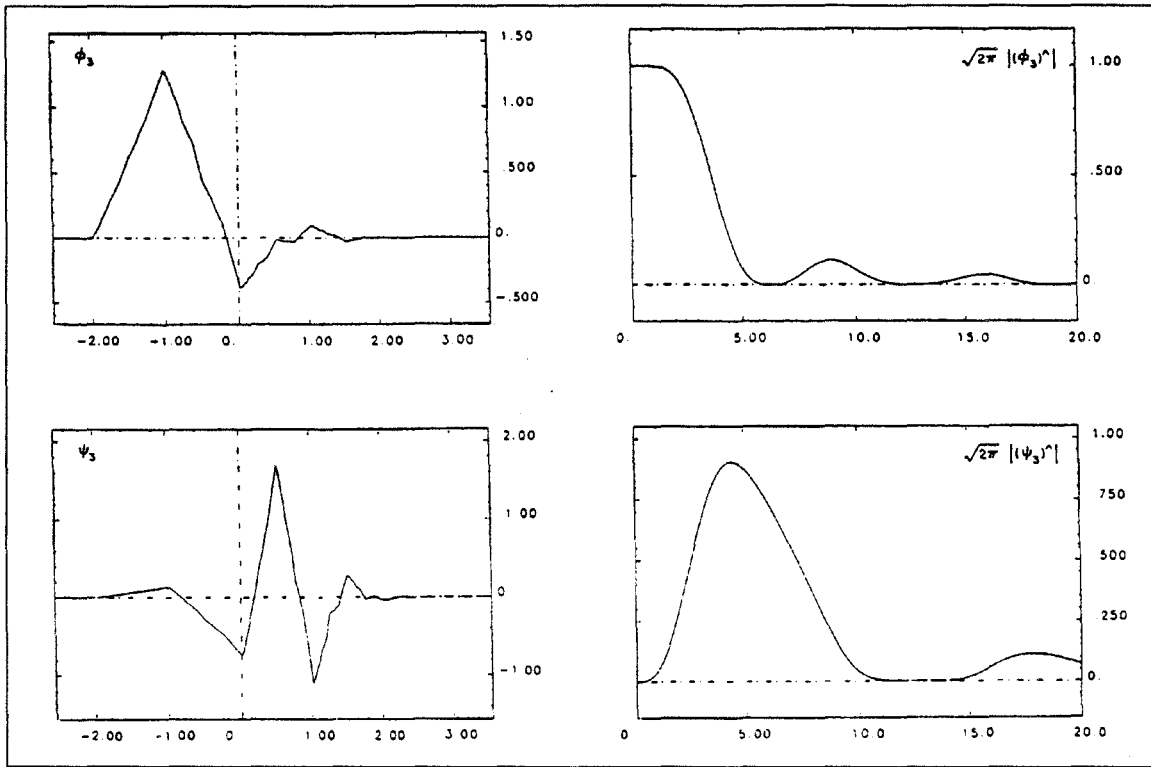
A - Coefficients des filtres de Daubechies jusqu'à l'ordre $N = 10$, normalisés à $\sqrt{2}$.

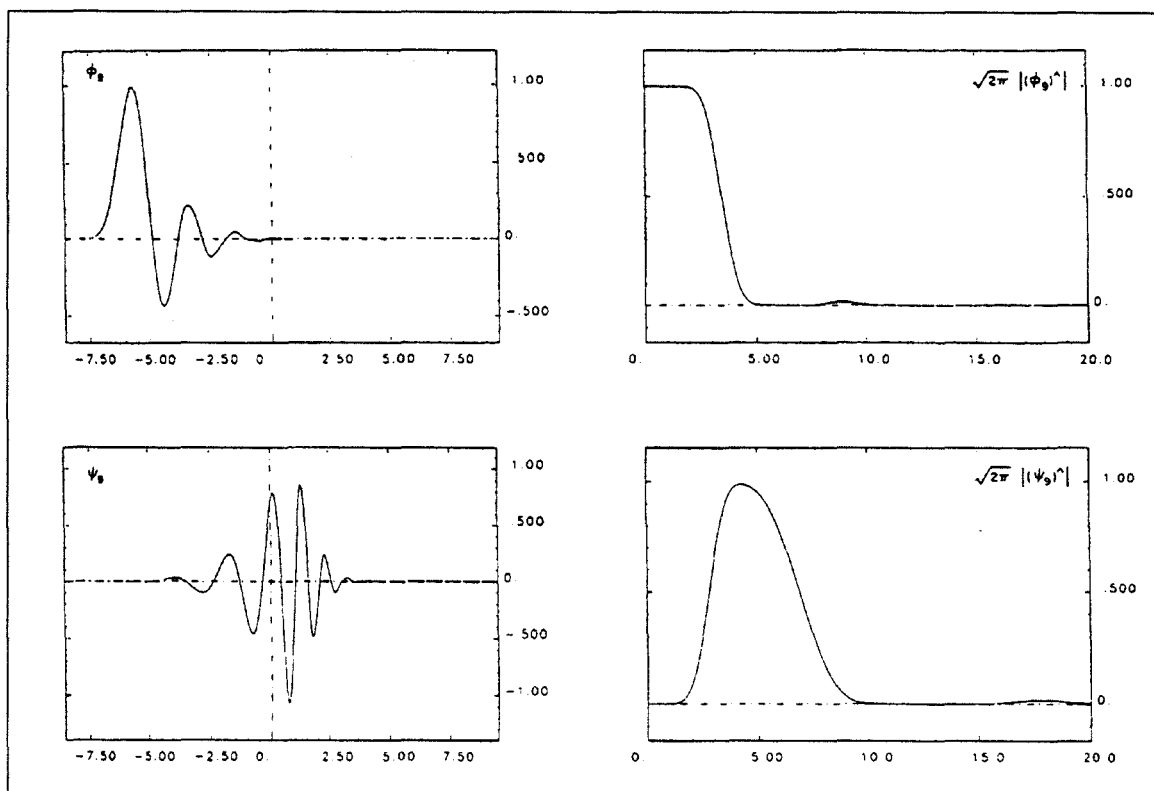
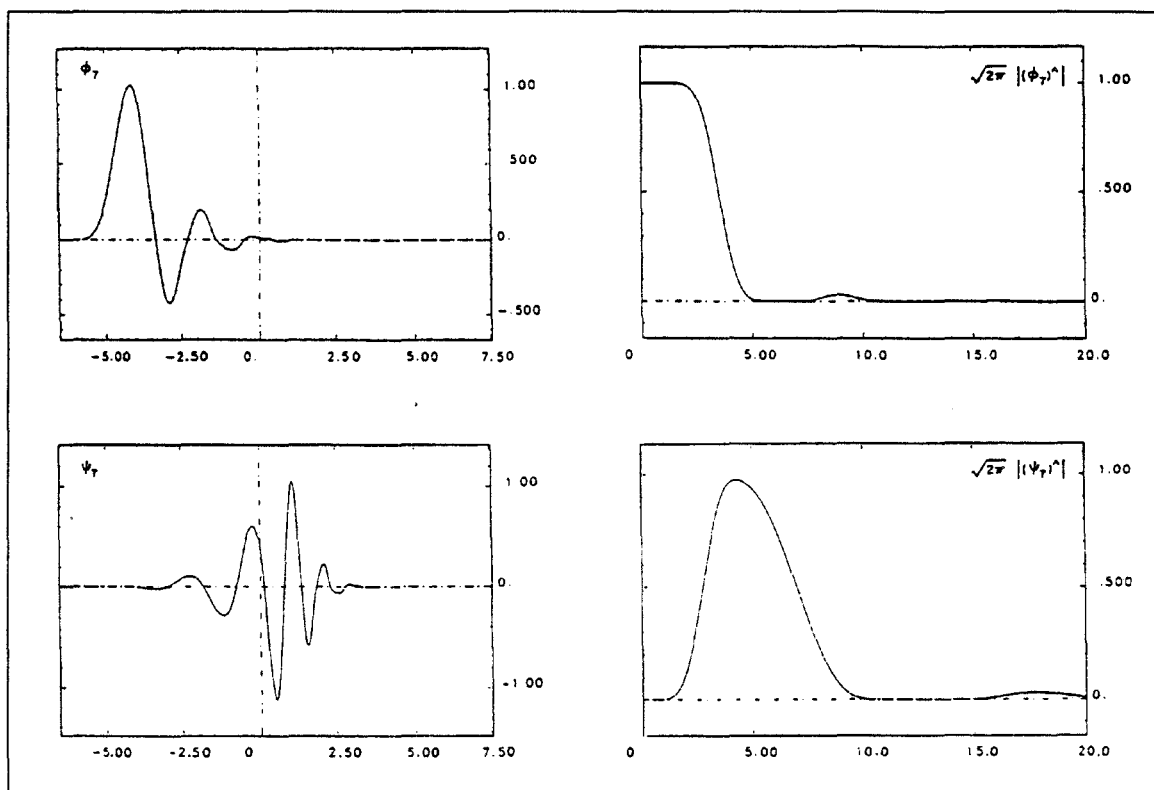
	n	$h_N(n)$
N=2	0	0,482962913145
	1	0,836516303738
	2	0,224143868042
	3	-0,129409522551
N=3	0	0,332670552950
	1	0,806891509311
	2	0,459877502118
	3	-0,135011020010
	4	-0,085441273882
	5	0,035226291882
N=4	0	0,230377813309
	1	0,714846570553
	2	0,630880767930
	3	-0,027983769417
	4	-0,187034811719
	5	0,030841381836
	6	0,032883011667
	7	-0,010597401785
N=5	0	0,160102397974
	1	0,603829269797
	2	0,724308528438
	3	0,138428145901
	4	-0,242294887066
	5	-0,032244869585
	6	0,077571493840
	7	-0,006241490213
	8	-0,012580751999
	9	0,003335725285
N=6	0	0,111540743350
	1	0,494623890398
	2	0,751133908021
	3	0,315250351709
	4	-0,226264693965
	5	-0,129766867567
	6	0,097501605587

	n	$h_N(n)$
	7	0,027522865530
	8	-0,031582039318
	9	0,000553842201
	10	0,004777257511
	11	-0,001077301085
N=7	0	0,077852054085
	1	0,396539319482
	2	0,729132090846
	3	0,469782287405
	4	-0,143906003929
	5	-0,224036184994
	6	0,071309219267
	7	0,080612609151
	8	-0,038029936938
	9	-0,016574541631
	10	0,012550998556
	11	0,000429577973
	12	-0,001801640704
	13	0,000353713800
N=8	0	0,054415842243
	1	0,312871590914
	2	0,675630736297
	3	0,585354683654
	4	-0,015829105256
	5	-0,284015542962
	6	0,000472484574
	7	0,128747426620
	8	-0,017369301002
	9	-0,044088253931
	10	0,013981027917
	11	0,008746094047
	12	-0,004870352993
	13	-0,000391740373
	14	0,000675449406
	15	-0,000117476784

	n	$h_N(n)$
N=9	0	0,038077947364
	1	0,243834674613
	2	0,0604823123690
	3	0,657288078051
	4	0,133197385825
	5	-0,293273783279
	6	-0,096840783223
	7	0,148540749338
	8	0,030725681479
	9	-0,067632829061
	10	0,000250947115
	11	0,022361662124
	12	-0,004723204758
	13	-0,004281503682
	14	0,001847646883
	15	0,000230385764
	16	-0,000251963189
	17	0,000039347320
N=10	0	0,026670057901
	1	0,188176800078
	2	0,527201188932
	3	0,688459039454
	4	0,281172343661
	5	-0,249846424327
	6	-0,195946274377
	7	0,127369340336
	8	0,093057364604
	9	-0,071394147166
	10	-0,029457536822
	11	0,033212674059
	12	0,003606553567
	13	-0,010733175483
	14	0,001395351747
	15	0,001992405295
	16	-0,000685856695
	17	-0,000116466855
	18	0,000093588670
	19	-0,000013264203

B - Allure des fonctions d'échelle et d'ondelettes construits par Daubechies (ordres 3, 5, 7 et 9)





REFERENCES

BIBLIOGRAPHIQUES

REFERENCES BIBLIOGRAPHIQUES

(Documents cités dans le corps du texte)

- [ADE87] E. H. Adelson and E. Simoncelli " Orthogonal pyramid transforms for image coding ", SPIE Vis. Com. and Im. Proc. II, vol. 845, 1987, pp. 50-58.
- [ALE82] I. Aleksander, T. J. Stonham, and R. A. Wilkie " Computer vision systems for industry ", Digital syst. indust. automat., vol. 1, n° 4, 1982, pp. 305-320.
- [ANT92] M. Antonini, M. Barlaud, P. Mathieu et I. Daubechies " Image coding using wavelet transform ", Proc. IEEE Trans. Im. Proc., vol. 1, n° 2, Apr. 1992, pp. 205-220.
- [ANA84] P. Anandan " Computing dense displacement fields with confidence measures in scenes containing occlusion ", SPIE Intelligent robots and computer vision, vol. 521, 1984, pp. 184-191.
- [BAA91a] N. Baaziz " Approches d'estimation et de compensation de mouvement multirésolutions pour le codage de séquences d'images ", Thèse, Univ. Rennes I, Oct. 1991.
- [BAA91b] N. Baaziz et C. Labit " Schémas d'estimation de mouvement multirésolution multigrille en codage de séquences d'images ", 13^{ème} colloque GRETSI, Juan-les-Pins, Sept. 91, pp. 865-868.
- [BAR92] M. Barlaud, L. Blanc-Feraud et J. M. Collin " Motion estimation involving discontinuities in a multiresolution scheme ", Proc. of SPIE Vis. Com. and Im. Proc. '92, vol. 1818, 1992, pp. 526-541.
- [BER82] H. C. Bergmann " Displacement estimation based on the correlation of image segments ", IEEE Proc. Int. Conf. on Elect. Im. Proc., York (England), July 1982, pp. 215-219.
- [BER84] H. C. Bergmann " Ein schnell konvergierendes Displacement-Schätzverfahren für die Interpolation von Fernsehbildsequenzen ", Ph. D. dissertation, Tech. Univ. of Hannover, Germany, Feb. 1984.
- [BIE87] J. Biemond, L. Looijenga, D. E. Boeke and R. H. J. M. Plompen " A pel-recursive Wiener-based displacement estimation algorithm ", Signal Processing, vol. 13, n° 4, Dec. 87, pp. 399-412.

- [BIE88] M. Bierling " Displacement estimation by hierarchical block-matching ", Proc. SPIE Visual com. and im. proc. '88, vol. 1001, 1988, pp. 942-951.
- [BOS92] F. Bosveld, R. L. Lagendijk and J. Biemond " Hierarchical coding of HDTV" Signal Processing. Image communication, vol. 4, n° 3, 1992, pp. 195-225.
- [BOU88] P. Bouthemy " Modèles et méthodes pour l'analyse du mouvement dans une séquence d'images ", Tech. et Sc. Informatiq., vol. 7, n°6, 1988, pp. 527-546.
- [BOU89] P. Bouthemy " A maximum likelihood framework for determining moving edges ", IEEE Trans. Pat. Anal. Mach. Intel., special issue on visual motion, vol. PAMI-11, n° 5, May 1989, pp. 499-511.
- [BOU90] P. Bouthemy et P. Lalande " Detection and tracking of moving objects based on a statistical regularization method in space and time", Lecture Notes in Computer Science, vol. 427, 1990, pp. 307-311.
- [BUR82] P. J. Burt " The pyramid as a structure for efficient computation", in 'Multiresolution image processing and analysis', edited by A. Rosenfeld, Springer-Verlag series information sciences, vol. 12, 1984, pp. 6-35.
- [BUR83] P. J. Burt, C. Yen and X. Xu " Multi-resolution flow-through motion analysis ", IEEE CVPR conference proceedings, June 1983, pp. 246-252.
- [BUR83b] P. J. Burt and E. H. Adelson " The laplacian pyramid as a compact image code ", IEEE Trans. on Comm., vol. COM-31, n° 4, Apr. 1983, pp. 532-540.
- [CAF76] C. Cafforio and F. Rocca " Methods for measuring small displacements of television images ", IEEE Trans. Infor. Theory, vol. IT-22, n° 5, Sept. 1976, pp. 573-579.
- [CAF83] C. Cafforio and F. Rocca " The differential method for image motion estimation ", in Image sequence processing and dynamic scene analysis, T.S. Huang, Ed., Nato ASI series, Springer, Berlin, 1983, pp. 104-124.
- [CAN71] J.C. Candy, M.A. Franke, B.G. Haskell, and F.W. Mounts " Transmitting television as clusters of frame-to-frame differences ", Bell Syst. Tech. Jour., vol. 50, n° 6, July-Aug. 1971, pp. 1889-1917.
- [CHA90] M.H. Chan, Y.B. Yu and A.G. Constantinides " Variable size block matching motion compensation with application to video coding ", Proc. IEEE, vol. 137, Part I, n° 4, Aug. 1990, pp. 205-212.

- [COH90] A. Cohen " Ondelettes, analyses multirésolutions et traitement numérique du signal ", Thèse de Doct., Paris IX, 1990.
- [CON89] V. Considine, A.S. Bhandal and J. N. Gooding " Single chip motion estimator for video codec application", Third IEE conf. image processing and its applications, Warwick University, 18-20 July 1989, pp. 285-289.
- [COR70] T. N. Cornsweet " Visual perception ", New York, Academic press, 1970, pp. 270-310.
- [CRO81] R. E. Crochiere and L. R. Rabiner " Interpolation and decimation of digital signals. A tutorial review ", Proc. IEEE, vol. 69, n° 3, March 1981, pp. 300-331.
- [DAU88] I. Daubechies " Orthonormal bases of compactly supported wavelets ", Communications on pure and applied math., vol. XLI, 1988, pp. 909-996.
- [DeJ52] F. De Jager " Delta modulation, a method of PCM transmission using a 7-unit code ", Philips Res. Rep., Dec. 1952, pp. 442-466.
- [ELI55] P. Elias " Predictive coding ", IRE Trans. Inform. Theory, vol. IT-1, Mar. 1955, pp. 16-33.
- [EST77] D. Esteban and C. Galand " Applications of quadrature mirror filters to split band voice coding schemes ", Proc. int. conf. acoust. speech signal proc., May 1977.
- [FAD91] M. H. Fadzil and T. J. Dennis " Video subband VQ coding at 64 kbit/s using short-kernel filter banks with an improved motion estimation technique ", Sig. Proc., vol. 3, 1991.
- [FEA90] J. C. Feauveau " Analyse multirésolution pour les images avec un facteur de résolution $\sqrt{2}$ ", Trait. signal, vol. 7, n° 2, 1990, pp. 117-128.
- [FRO90] J. Froment " Traitement d'images et applications de la transformée en ondelettes ", Thèse de doctorat, Univ. Paris IX Dauphine, U.F.R. Math. Déc., 1990.
- [GAI92] T. Gaidon, M. Barlaud et P. Mathieu " Image sequence coding using quincunx wavelet transform motion compensation and lattice vector quantization ", Vis. Com. and Im. Proc. '92, 18-20 Nov. 1992, Boston Ma.
- [GAS90] C. Gasquet et P. Witomski " Analyse de Fourier et applications. Filtrage, calcul numérique, ondelettes ", Masson, Paris, 1990.

- [GON77] R. C. Gonzalez and P. Wintz "Digital image processing " Addison-Wesley publishing company, Inc., Massachusetts, 1977, USA.
- [GRI85] W. Grimson " Computational experiments with a feature based stereo algorithm ", IEEE Trans. Pattern Anal. Mach. Intel., vol. PAMI-7, Jan. 1985, pp. 17-34.
- [HAB74] A. Habibi " Hybrid coding of pictorial data ", IEE Trans. Commun., vol. COM-22, May 1974, pp. 614-624.
- [HAS72] B. G. Haskell, F.W. Mounts, and J.C. Candy " Interframe coding of videotelephone pictures ", Proc. IEEE, vol. 60, n° 7, July 1972, pp. 792-800.
- [HAS78] B.G. Haskell " Frame replenishment coding of television ", a chapter in Image transmission techniques, W.K. Pratt, Ed., New York : Academic Press, 1978.
- [HOC67] D. Hochman, H. Katzman, and D.R. Weber " Application of redundancy reduction to television bandwidth compression ", Proc. IEEE, vol. 55, Mar. 1967, pp. 263-266.
- [HOR81] B. K. P. Horn et B. G. Schunck " Determining optical flow ", Artificial Intelligence, vol. 17, 1981, pp. 185-203.
- [HOR86] B. K. P. Horn " Robot Vision ", the MIT press, 1986.
- [HOT33] H. Hotelling " Analysis of complex of statistical variables into principle components ", J. Educ. Psychol., vol. 24, 1933, pp. 417-441.
- [HUA81] T. S. Huang and R. Y. Tsai " Image sequence analysis : motion estimation ", in Image sequence analysis. Berlin, Germany : Springer-Verlag, 1981, pp. 1-18.
- [HUB79] D. H. Hubel and T. N. Wiesel " Brain mechanism of vision ", Sci. Amer., vol. 241, Sept. 1979, pp. 150-162.
- [JAI81] J.R. Jain and A.K. Jain " Displacement measurement and its application in interframe image coding ", IEEE Trans. Commun., vol. COM-29, n° 12, Dec. 1981, pp. 1799-1808.
- [KAR47] H. Karhunen " Uber lineare Methoden in der Wahrscheinlichkeitsrechnung ", Ann. Acad. Sci. Fenn., Ser. A.I. 37, Helsinki, Finland, 1947.

- [KER90] M. Kerdranvat " Hierarchical motion estimation and motion information encoding ", Sig. Proc. of HDTV, II, L. Chiariglione (ed.), Elsevier Science Publishers B. V., 1990, pp. 101-108.
- [KOG81] T. Koga, K. Iinuma, A. Hirano, Y. Iijima and T. Ishiguno " Motion-compensated interframe coding for video conferencing " Proc. NTC 81, New Orleans, LA, Dec. 1981, pp. G5.3.1-G5.3.5.
- [KOR67] C.M. Kortman " Redundancy reduction. A practical method of data compression ", Proc. IEEE, vol. 55, Mar. 1967, pp. 253-263.
- [KUF76] S. W. Kuffler and J. G. Nicholls " From neuron to brain ", Sunderland, MA : Sinauer Assoc., 1976.
- [KUN85] M. Kunt, A. Ikonomopoulos and M. Kocher " Second generation image coding techniques ", Proc. of IEEE, vol. 73, n° 4, April 1985, pp. 549-574.
- [LAB83] C. Labit et A. Benveniste " Motion estimation in a sequence of television pictures ", in T. S. Huang (ed) *Image sequence analysis*, NATO-ASI Series, vol. F2, Springer-Verlag, 1983.
- [LEM86] P. G. Lemarié et Y. Meyer " Ondelettes et bases hilbertiennes ", Rev. Mat. Ibero., vol. 2, n° 1 et 2, 1986, pp. 1-18.
- [LIM75] J.O. Limb and J.A. Murphy " Measuring the speed of moving objects from television signals ", IEEE Trans. Comm., vol. COM-23, n°4, April 1975, pp. 474-478.
- [MAL89a] S. G. Mallat " A theory for multiresolution signal decomposition : the wavelet representation ", IEEE Trans. pattern anal. machine intell., vol. 11, n° 7, July 1989, pp. 674-693.
- [MAL89b] S. G. Mallat " Multifrequency channel decomposition of images, and wavelet models ", IEEE Trans. Acous. Speech Sig. Proc., vol. ASSP-37, n° 12, Dec. 1989, pp. 2091-2110.
- [MAL91] S. Mallat " Zero-crossing of a wavelet transform ", IEEE Trans. Inf. Th., vol. IT-37, n° 4, 1991, pp. 1019-1033.
- [MEY90] Y. Meyer " Ondelettes et opérateurs I. Ondelettes ", Hermann, Paris, 1990.
- [MIL85] P. C. Millar " Recursive quadrature mirror filters-Criteria, specification and design method ", IEEE Trans. Acous., Speech, Sig. Proc., vol. ASSP-33, n° 2, Apr. 1985, pp. 413-420.

- [MOO87] R. J. Moorhead, S. A. Rajala and L. W. Cook " Image sequence compression using a pel-recursive motion-compensated technique ", IEEE J. Sel. Areas Comm., vol. SAC-5, n° 7, Aug. 1987, pp. 1100-1114.
- [MOU69] F.W. Mounts " A video encoding system with conditional picture element replenishment ", Bell Syst. Tech. Journal, vol. 48, n° 7, Sept. 1969, pp. 2545-2554.
- [MUS85] H.G. Musmann, P. Pirsch and H.J. Grallert " Advances in picture coding ", Proc. IEEE, vol. 73, n° 4, April 1985, pp. 523-548.
- [NAG83] H. N. Nagel " Overview on image sequence analysis ", in Image sequence processing and dynamic scene analysis, T. S. Huang, Ed., Berlin, Germany : Springer-Verlag, 1983, pp. 2-39.
- [NET79] A.N. Netravali and J.D. Robbins " Motion-compensated television coding : part I ", Bell Syst. Tech. Jour., vol. 58, n° 3, March 1979, pp. 631-670.
- [NET80a] A.N. Netravali, J.O. Limb " Picture coding: a review ", Proc. IEEE, vol. 68, n° 3, March 1980, pp. 366-406.
- [NET80b] A.N. Netravali and J.D. Robbins " Motion compensated coding : some new results ", Bell Syst. Tech. Jour., vol. 59, n°9, Nov. 1980, pp. 1735-1745.
- [NET88] A. N. Netravali and B. G. Haskell " Digital picture : representation and compression ", AT&T Bell Laboratories Plenum Press, 1988, New York.
- [OLI48] B.M. Olivier, J.R. Pierce, C.E. Shannon " The philosophy of PCM ", Proc. IRE, vol. 36, Oct. 1948, pp. 1324-1331.
- [PAR90] I. Parke " A hardware motion compensator for a video-conferencing codec ", IEE colloquium on "Application of motion compensation", London, 8 Oct. 90.
- [PEA75] D. E. Pearson " Transmission and display of pictorial information ", New York : Wiley, 1975, pp. 38-42.
- [PIR84] G. Pirani and V. Zingarelli " Analytical formula for design of quadrature mirror filters ", IEEE Trans. Acous., Speech, Sig. Proc., vol. ASSP-32, June 1984, pp. 645-648.
- [RIO91] O. Rioul and M. Vetterli " Wavelets and signal processing ", IEEE Sig. Proc. Mag., Oct. 1991, pp. 14-38.

- [ROA80] J. W. Roach and J. K. Aggarwal " Determining the movement of objects from a sequence of images ". IEEE Trans. pattern anal. machine intell., vol. PAMI.2, n° 6, pp. 554-562.
- [SCH59] W.F. Schreiber, C.F. Knapp and N.D. Kay "Synthetic highs, an experimental TV bandwidth reduction system ", J. SMPTE, vol. 68, Aug. 1959, pp. 525-537.
- [SCH63] W. F. Schreiber " The mathematical foundation of the synthetic highs system ", MIT, RLE Quart. Progr. Rep., n° 68, Jan. 1963, p. 140.
- [SCH72] W. F. Schreiber, T.S. Huang and O.J. Tretiak " Contour coding of images ", in Picture bandwidth compression, T.S. Huang and O.J. Tretiak Eds. New York : Gordon and Breach, 1972.
- [SHA49] C. E. Shannon " The mathematical theory of communication ", Urbana, IL; Univ. Illinois Press, 1949.
- [SHA60] C. E. Shannon " Coding theorem for a discrete source with a fidelity criterion ", in Information and decision processes, R.E. Machol, Ed. New York : Mc Graw-Hill, 1960, pp. 93-126.
- [SRI84] R. Srinivasan and K.R. Rao " Predictive coding based on efficient motion estimation ", Proc. ICC '84, May 1984, pp. 521-526.
- [TYL77] C. W. Tyler and J. J. Chang " Visual echoes : the perception of repetition in quasi-random patterns ", Vision research, vol. 17, 1977, pp. 109-116.
- [ULL79] S. Ullman " The interpretation of visual motion " MIT press, Cambridge, 1979.
- [VET90] M. Vetterli and C. Herley " Wavelets and filter banks : relationships and new results ", Proc. IEEE ICASSP, Albuquerque, April 1990.
- [WAL84] D.R. Walker and K.R. Rao " Improved pel-recursive motion compensation " IEEE Trans. Comm., vol. COM-32, n° 10, Oct. 1984, pp. 1128-1134.
- [WAN92] Q. Wang and R.J. Clarke " Motion estimation and compensation for image sequence coding " Signal processing : Image communication, vol. 4, n° 2, April 1992, pp. 161-172.
- [WIN72] P.A. Wintz " Transform picture coding ", Proc. IEEE, vol. 60, n° 7, July 1972, pp. 809-820.

[WOO86] J. W. Woods " Subband coding of images ", IEEE Trans. Acous. Speech Sig. Proc., vol. ASSP-34, n° 5, Oct. 1985, pp. 1278-1288.

BIBLIOGRAPHIE

BIBLIOGRAPHIE

(Documents indiqués comme information complémentaire)

Multirésolution - Ondelettes

- [BAT87] G. Battle "A block spin construction of ondelettes. Part I : Lemarié functions" Comm. Math. Phys., vol. 110, 1987, pp. 601-615.
- [BON91] P. Bonnet et D. Remond " Une transformée en ondelettes rapide ", Trait. du Signal, vol. 8, n°3, 1991, pp. 195-207.
- [BUR81] P. J. Burt " Fast filter transforms for image processing ", Comp. Graph. Im. Proc., vol. 16, 1981, pp. 20-51.
- [COH90] A. Cohen " Ondelettes, analyse multirésolution et filtres miroirs en quadrature ", Annales Institut H. Poincaré, Analyse non linéaire, vol. 7, n° 5, 1990, pp. 439-459.
- [COH91] A. Cohen, J. Froment et J. Istas " Analyse multirésolution des signaux aléatoires ", Comptes rendus de l'Académie des sciences. Série I, Mathématiques, vol. 312, n° 8, 1991, pp. 567-570.
- [CON89] V. Considine, A.S. Bhandal and J. N. Gooding " Single chip motion estimator for video codec application ", Third IEE conf. image processing and its applications, Warwick University, 18-20 July 1989, pp. 285-289.
- [DAU88] I. Daubechies " Time-frequency localization operators : a geometric phase space approach ", IEEE Trans. Inf. Th., vol. IT-34, n° 4, 1988, pp. 605-612.
- [DAU90] I. Daubechies " The wavelet transform, time-frequency localization and signal analysis ", IEEE Trans. Inf. Th., vol. IT-36, n°5, 1990, pp. 961-1005.
- [DEV91] V. Devlaminck, P. Nikyema, J. P. Dubus " Analyse de mouvement par décomposition en ondelettes ", 13^{ème} colloque GRETSI, Juan-les-Pins, Sept. 1991, pp. 977-979.
- [DOR88] C. Dorize " Ondelettes discrètes orthogonales et signaux acoustiques ", mémoire ing. CNAM, ENST Paris, Dept. Signal, gpe. acous., 1988.

- [FEA90] J. C. Feauveau " Analyse multirésolution par ondelettes non orthogonales et bancs de filtres numériques ", Th. Doct., Univ. Paris Sud, Janv. 1990.
- [GRO84] A Grossmann and J. Morlet " Decomposition of Hardy functions into square integrable wavelets of constant shape ", Siam Jour. Math. Anal., vol. 15, n° 4, July 1984, pp. 723-736.
- [JOL90] J. M. Jolion " Analyse d'images : le modèle pyramidal ", Trait. du Signal, vol. 7, n°1, 1990, pp. 5-17.
- [KNO90] G. Knowles " VLSI architecture for the discrete wavelet transform ", Elec. letters, vol. 26, n° 15, 19th July 1990, pp. 1184-1185.
- [LEW90] A. S. Lewis and G. Knowles " Video compression using 3D wavelet transforms ", Elect. letters, vol. 26, n° 6, 15th March 1990, pp. 396-398.
- [MAL89] S. G. Mallat " Multiresolution approximations and wavelet orthonormal bases. of $L^2(\mathbb{R})$ ", Trans. Am. Math. Soc., vol. 315, n° 1, Sept. 1989, pp. 69-87.
- [MAL92] S. Mallat, W. L. Hwan " Singularity detection and processing with wavelets " IEEE Trans. Inf. Th., vol. IT-38, n° 2, 1992, pp. 617-643.
- [MAT90] P. Mathieu, M. Barlaud et M Antonini " Compression d'image par transformée en ondelettes et quantification vectorielle ", Trait. Sign., vol. 7, n°2, 1990, pp. 101-115.
- [MEE87] P. Meer, E. S. Baugher and A. Rosenfeld " Frequency domain analysis and synthesis of image pyramid generating kernels ", IEEE Trans. Pattern Anal. Mach. Intel., vol. PAMI-9, n° 4, July 1987, pp. 512-522.
- [MES91] O. Meste, H. Rix, R. Jane and P. Caminal " Détection battement par battement de potentiels tardifs dans les électrocardiogrammes à haute amplification ", Innov. Tech. Biol. Med., vol. 12, n° 5, 1991, pp. 518-531.
- [MEY85] Y. Meyer " Principe d'incertitude, bases hilbertiennes et algèbres d'opérateurs ", Bourbaki seminar, n° 662, 1985-1986.
- [MEY87] Y. Meyer, S. Jaffard et O. Rioul " L'analyse par ondelettes ", Pour la science, Sept. 1987, pp. 28-37.
- [NIK93] P. Nikyema, V. Devlaminck et J. P. Dubus " Technique multifiltre en compensation de mouvement par block matching hiérarchique ", soumis au 14^{ème} colloque du GRETSI, Juan-les-Pins, Sept. 1993.

- [PER89] V. Perrier et C. Basdevant " La décomposition en ondelettes périodiques, un outil pour l'analyse de champs inhomogènes. Théorie et algorithmes ", La Rech. Aéronautique, n° 3, Mai 1989, pp. 53-67.
- [RIO91] O. Rioul et P. Duhamel " Transformées en ondelettes discrètes et continues : comparaison et algorithmes rapides ", 13^{ème} colloque GRETSI, Juan-les-Pins, Sept. 1991, pp. 193-196.
- [SIM92] E. P. Simoncelli, W. T. Freeman, E. H. Adelson and D. J. Heeger " Shiftable multiscale transforms ", IEEE Trans. Infor. Th., vol. IT-38, n° 2, March 1992, pp. 587-607.
- [TEM90] S. Tembely " Ondelettes et analyse fonctionnelle dans les domaines lipschitziens ", Th. Doct., Univ. Paris IX (Sci.), 1990.
- [VET84] M. Vetterli " Multi-dimensional sub-band coding : some theory and algorithms ", Sign. Proc., vol. 6, n°2, Apr. 1984, pp. 97-112.

Mouvement - Compression

- [AGG86] J. K. Aggarwal and J. M. Magee " Determining motion parameters using intensity guided range sensing ", Pattern Recognition, vol. 19-2, 1986, pp. 169-180.
- [BEN92] M. Benjelloun " Analyse de scène dynamique : 1ère partie ", Etude biblio. sur le mouvement, INT Evry, 1992.
- [CAB92] F. Cabestaing " Détection de contours en mouvement dans une séquence d'images. Conception et réalisation d'un processeur câblé temps-réel ", Th. Doct., Univ. Lille I (Autom.), Jan. 1992.
- [CHO88] B. Choquet " Estimation de mouvement et segmentation spatio-temporelle en séquences d'images. Contribution à la télévision haute définition : utilisation de l'information de mouvement dans un système HDMAC ", Th. Doct., Rennes I, Déc. 1988.
- [DEN88] J. Dengler and M. Schmidt " The dynamic pyramid - a model for motion analysis with controlled continuity ", Int. Jour. Pattern Recog. Artif. Intel., vol. 2, n° 2, 1988, pp. 275-286.
- [FRA89] E. François et P. Bouthemy " Vers une interprétation qualitative de comportements cinématiques dans la scène à partir du mouvement apparent ", Rapp. de rech. I.N.R.I.A. Rennes, vol. 1081, Août 1989.
- [FRA90] E. François et P. Bouthemy " The derivation of qualitative information in motion analysis ", Lecture Notes in Computer Science, vol. 427, 1990, pp. 307-311.
- [FRA91] E. François " Interprétation qualitative du mouvement à partir d'une séquence d'images ", Th. Doct., Univ. Rennes I (Inform.), 1991.
- [JAI81] A. K. Jain " Image data compression : a review ", Proc. IEEE, vol. 69, n° 3, March 1981, pp. 349-389.
- [KER90] M. Kerdranvat " Hierarchical motion estimation and motion information encoding ", Signal processing of HDTV. II, L. Chiariglione (ed), Elsevier Science Publishers B.V., 1990, pp. 101-108.

- [LAB87] C. Labit "Séquence d'images et techniques de codage MICDA intra/inter-image ", Rapport de rech., INRIA-Rennes, n° 630, Fév. 1987.
- [LAB 89] C. Labit " La compréhension de scènes dynamiques par l'analyse de séquences d'images ", 12^{ème} coll GRETSI, Juan-les-Pins, 12-16 juin 1989, pp. 1035-1040.
- [LAL90] P. Lalande " Détection du mouvement dans une séquence d'images selon une approche Markovienne; application à la robotique sous-marine ", Th. Doct., Univ. Rennes I (Inform.), 1990.
- [PEC90] M. Pecot, P. J. Tourtier and Y. Thomas " Compatible coding of television images, part 1. Coding algorithm ", Sign. processing : Image communication 2, 1990, pp. 245-258.
- [SAY92] K. Sayood and K. Anderson " A differential lossless image compression scheme ", IEEE Trans. Sig. Proc., vol. 40, n°1, 1992, pp. 236-241.
- [SHA90] H. Shariat and K. E. Price " Motion estimation with more than two frames ", IEEE Trans. Pat. Anal. Mac. Int., vol. PAMI-12, n° 5, May 1990, pp. 417-434.
- [STU83] J. Stuller and G. Krishnamurthy " Kalman filter formulation of low-level television image motion estimation ", Computer vision, graphics and image proc. 21, 1983, pp. 169-204.
- [TAM91] A. Tamtaoui et C. Labit " Constrained disparity and motion estimators for 3DTV image sequence coding ", Signal Processing. Image communication, vol. 4, n° 1, 1991, pp. 45-54.
- [UZ92] K. M. Uz, K. Ramchandran and M. Vetterli " Combined multiresolution source coding and modulation for digital broadcast of HDTV", Sig. Proc. Im. Comm., vol. 4, 1992, pp. 283-292.
- [VAN92] L. Vandendorpe " Hierarchical transform and subband coding of video signals", Sig. Proc. Im. Comm., vol. 4, 1992, pp. 245-262.
- [VIE89] C. Vieren, J. G. Postaire, P. Bonnet et J. P. Deparis " Détection du contour extérieur d'objets en mouvement sur un fond non uniforme ", 12^{ème} colloque GRETSI, Juan-les-Pins, Juin 1989, pp. 621-624.
- [WAN89] L. Wang and M. Goldberg " Reduced-difference pyramid : a data structure for progressive image transmission ", Optical engineering, vol. 28, n° 7, July 1989, pp. 708-716.

[WAN90] Q. Wang and R. J. Clarke " Motion compensated sequence coding using image pyramids ", Elect. letters, vol. 26, n° 9, 26th Apr. 1990, pp. 575-576.

