

Compression d'image

par transformée en ondelette

et quantification vectorielle

Image coding using wavelet transform

and vector quantization



P. MATHIEU

LASSY Equipe de I3S CNRS, Université de NICE, SOPHIA ANTIPOLIS, Bât. 4, S.P.I.
rue Albert Einstein, 06560 VALBONNE (FRANCE)
et GDR TRAITEMENT DU SIGNAL ET IMAGE CNRS.

P. Mathieu, est né à Alger le 10 mai 1956. Il a reçu le diplôme d'ingénieur ENSEEIHT et une thèse de Docteur-ingénieur à l'INP Toulouse. Il est actuellement Maître de Conférences au Laboratoire de Signaux et Systèmes de l'Université de Nice-Sophia Antipolis. Ses domaines de recherche s'inscrivent en Traitement de Signal bidimensionnel. Il s'intéresse plus particulièrement dans le cadre du traitement d'image, à l'analyse multirésolution par Ondelettes orthogonales et biorthogonales, au codage d'image, à la restauration d'image et à la détection de contours.



M. BARLAUD

LASSY Equipe de I3S CNRS, Université de NICE, SOPHIA ANTIPOLIS, Bât. 4, S.P.I.
rue Albert Einstein, 06560 VALBONNE (FRANCE)
et GDR TRAITEMENT DU SIGNAL ET IMAGE CNRS.

M. Barlaud, né en France le 24 novembre 1945. Il a obtenu le grade docteur d'état de l'Université Paris XII. Il est actuellement Professeur au Laboratoire Signaux et Systèmes de l'Université de Nice-Sophia Antipolis. Ses axes de recherches concernent le traitement du signal. Il a particulièrement étudié la modélisation, l'identification et le filtrage de signaux non-stationnaires. Actuellement ses travaux portent sur le traitement des signaux bidimensionnels. Mise en œuvre d'analyse multirésolution par ondelettes biorthogonales. Quantification vectorielle multirésolution. Applications à la compression d'images, la restauration d'images, la détection de contours et la segmentation d'images.



M. ANTONINI

LASSY Equipe de I3S CNRS, Université de NICE, SOPHIA ANTIPOLIS, Bât. 4, S.P.I.
rue Albert Einstein, 06560 VALBONNE (FRANCE)
et GDR TRAITEMENT DU SIGNAL ET IMAGE CNRS.

M. Antonini, est né en France le 29 août 1965. Il a reçu le diplôme de DEA en Automatique et Traitement du Signal en 1988 à l'Université de Nice-Sophia Antipolis. Depuis 1988 il travaille sur sa thèse de doctorat au laboratoire de Signaux et Systèmes de l'Université de Nice-Sophia Antipolis. Ses domaines de recherche concernent le Traitement de Signal bidimensionnel et plus particulièrement, l'analyse multirésolution d'images par ondelettes orthogonales et biorthogonales et, le codage d'image.

RÉSUMÉ

Dans cet article est proposée une nouvelle méthode de compression numérique d'image.

Dans une première étape l'image est décomposée en sous-images au moyen d'une transformée en ondelettes. Les ondelettes choisies sont à la fois bien localisées dans l'espace et en fréquence, et privilégient les directions horizontale et verticale, ce qui permet de respecter les caractéristiques de la vision humaine. La décomposition est effectuée par un algorithme rapide à structure pyramidale.

Ensuite les coefficients d'ondelette sont codés par une méthode de quantification vectorielle.

Un codebook est élaboré pour chaque résolution et direction privilégiée à partir d'une séquence d'apprentissage en utilisant un critère quadratique. Ainsi un vecteur à coder est classifié (direction, résolution) puis la seule sous-classe appropriée est scrutée.

Finalement l'allocation de bit de chaque classe est ajustée de façon à préserver les formes et contours.

MOTS-CLÉS : Ondelette, algorithme pyramidal multiéchelle, quantification vectorielle.

ABSTRACT

The purpose of this paper is to propose a new scheme for image compression.

First, we use a wavelet transform in order to obtain a set of orthonormal subclasses of images. The wavelet functions are well localized both in the space and frequency domains. The original image is decomposed on this orthonormal basis with a pyramidal algorithm architecture. This decomposition privileges horizontal, vertical and diagonal orientations. Since wavelet decomposition is based on multiresolution and privileged directions, it tries to match human vision.

The wavelets coefficients of each classes are then vector quantized. A separate optimal codebook is designed for each given resolution and direction using a training sequence and a MSE distortion measurement.

Then the input vector is classified (resolution and direction) and only the appropriate subclass of the codebook is then checked using the usual MSE. Thus the computational complexity is reduced.

Finally bit allocation of each class is adjusted in order to preserve edges and shapes.

KEY WORDS : Wavelet, multiscale pyramidal algorithm, vector quantization.

1. Introduction

Dans de nombreux domaines l'image numérisée remplace les images analogiques classiques telles que les photographies, radiographies, etc... Le très grand nombre de données brutes nécessaires pour la décrire rend la durée de transmission souvent trop longue et le stockage trop coûteux. Il est donc nécessaire de compresser l'information contenue dans l'image en en extrayant l'information visuelle qui seule sera codée : On réalise ainsi une réduction très sensible du volume de données.

La compression de l'information peut être obtenue par une transformation des données en les projetant sur une base de fonctions orthogonales et réaliser ensuite un codage de cette transformée.

Du fait de la nature du signal d'image et du fonctionnement de la vision humaine, cette transformation doit admettre la non-stationnarité et doit être à la fois « bien » localisée en fréquence et dans l'espace. Pour éviter une redondance néfaste à la compression, elle doit être orthogonale et pour éviter des temps de calculs prohibitifs, il doit exister un algorithme rapide. La transformée en ondelette définie par Y. Meyer et J. Lemarié remplit toutes ces conditions [18].

Le codage se fait au niveau des coefficients de la transformée de manière adaptée. On localise la distorsion en tenant compte d'un facteur perceptuel. La quantification vectorielle a été retenue pour notre application car c'est une méthode plus efficace que la quantification scalaire [29].

Nous proposons donc une nouvelle méthode de codage associant une transformée en ondelette à un algorithme de quantification vectorielle.

2. Transformée en ondelette

2.1. INTRODUCTION

L'intérêt que présentent de nombreuses transformées pour comprimer l'information est de projeter le signal sur une base de fonctions orthogonales, c'est-à-dire de distribuer l'énergie de ce signal sur des composantes décorrélatées entre elles. Il existe de nombreuses transformées orthogonales dont les propriétés diffèrent.

On peut citer la transformée de Fourier (DFT), la transformée en Cosinus (DCT), en Sinus (DST) et celle de Karhunen-Lève (KL) qui sont les plus

connues et les plus utilisées, ainsi que les transformées de Haar et de Hadamard.

La transformée de Karhunen-Løve est la transformée optimale au sens où elle condense le mieux l'énergie sur ses composantes [16]. L'absence d'algorithme rapide lui fait souvent préférer la DCT qui dans de nombreux cas donne des résultats très comparables [2]. Cependant l'absence de propriété de convolution rigoureuse [26] pousse à l'utilisation de la DFT dans sa version rapide FFT. Ces transformées localisent bien l'énergie en fréquence mais sont tout à fait délocalisées en temps : elles n'admettent pas la non-stationnarité.

A l'opposé, les transformées de Haar et de Hadamard présentent une bonne localisation temporelle mais pas fréquentielle [1].

La transformée de Wigner-Ville qui n'est pas une transformée orthogonale, permet de prendre en compte la non-stationnarité du signal. elle présente cependant plusieurs inconvénients majeurs : sa bilinéarité rend son inversion délicate [27] et comme il ne s'agit pas d'une projection sur une base de fonctions, le nombre de points de la transformée évolue comme N^2 si N est le nombre de points du signal.

La transformée en ondelette définie par Y. Meyer et J. Lemarié est une transformation orthogonale qui admet la non-stationnarité, elle est « bien » localisée en fréquence et en temps et peut être obtenue par un algorithme rapide. Du fait de toutes ces propriétés nous avons retenu cette transformation, que nous décrivons dans le paragraphe suivant.

2.2. DÉFINITION DES ONDELETTES

Les ondelettes sont des fonctions générées à partir d'une fonction ψ par translations et dilatations. Grossmann et Morlet ont introduit [15] cette fonction ψ qui, dilatée d'un facteur d'échelle a et translatée d'un temps b permet d'analyser un signal, de le manipuler puis de le resynthétiser

$$(1) \quad \psi_{a,b}(t) = a^{-1/2} \psi \left(\frac{t-b}{a} \right) \quad a > 0.$$

Ici, nous considérons que t est une variable monodimensionnelle.

L'ondelette mère, ainsi que sa transformée de Fourier, doivent être bien localisées. De plus, la condition $\int \psi(x) dx = 0$ doit être satisfaite. Cette condition entraîne que l'ondelette est oscillante (au moins quelques oscillations).

L'idée principale de la transformée en ondelette est de représenter une fonction f quelconque comme une superposition d'ondelettes. De plus, cette transformée contracte et dilate l'axe des temps ce qui permet d'introduire la notion d'échelle ou de résolution.

Y. Meyer a montré [22] qu'il existait des fonctions ψ telles que pour $a = 2^j$ et $b = k 2^j$ les $\psi_{j,k}(t)$ constituent une base orthogonale de $L^2(\mathbb{R})$

$$(2) \quad \psi_{j,k}(t) = 2^{-j/2} \psi(2^{-j} t - k) \quad (j, k) \in \mathbb{Z}^2.$$

C'est un sous-ensemble de ces fonctions que nous utilisons dans l'article.

Les coefficients d'ondelette sont alors calculés par :

$$(3) \quad c_{j,k} = \langle f, \psi_{j,k} \rangle = \int \psi_{j,k}(x) f(x) dx$$

$$\text{et} \quad f = \sum c_{j,k} \psi_{j,k}.$$

De façon à introduire la notion de multirésolution, on définit aussi une fonction d'échelle φ . Comme pour la fonction ψ on a :

$$(4) \quad \varphi_{j,k}(t) = 2^{-j/2} \varphi(2^{-j} t - k) \quad (j, k) \in \mathbb{Z}^2.$$

Ces fonctions $\varphi_{j,k}$ forment une famille orthogonale pour j fixé. Elles possèdent la propriété suivante :

$$(5) \quad \varphi(t) = \sum_n \alpha_n \varphi(2t - n)$$

ce qui entraîne [10] que l'ondelette associée suit la relation

$$(6) \quad \psi(t) = \sum_n (-1)^n \alpha_{n+1} \varphi(2t + n).$$

On a donc, en projetant sur cette famille de fonctions $\varphi_{j,k}$ une approximation du signal à la résolution 2^j . La perte d'information, lors du passage d'une résolution 2^{j-1} à une résolution 2^j plus grossière, est alors décrite par les coefficients $c_{j,k} = \langle f, \psi_{j,k} \rangle$ définis précédemment.

Des exemples de fonction d'échelle φ et d'ondelette ψ (ondelette de Meyer) sont présentés figures 1 et 2.

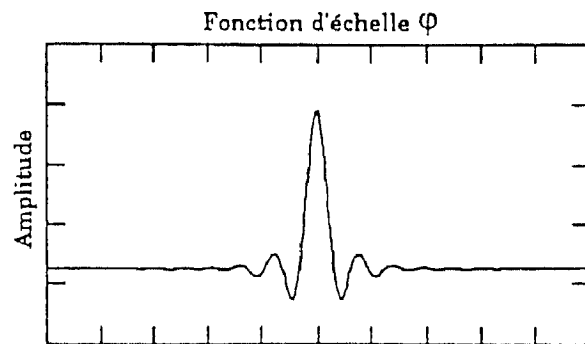


Fig. 1.

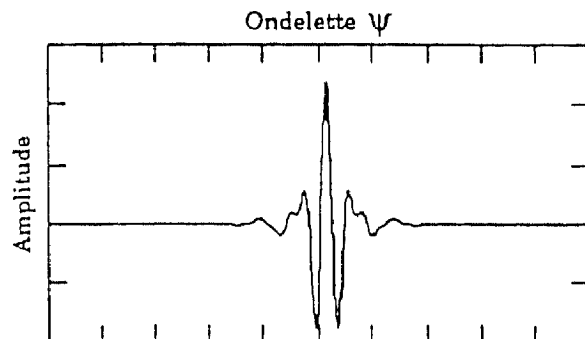


Fig. 2.

La mise en œuvre algorithmique de l'analyse multirésolution est présentée au paragraphe 2.3 suivant.

2.3. MISE EN ŒUVRE ALGORITHMIQUE SUR DES SIGNAUX ÉCHANTILLONNÉS

Les signaux d'images que nous utilisons en pratique sont des signaux échantillonnés. Les échantillons de ces signaux d'origine sont considérés comme les coefficients d'approximation $s_{0,n}$ de plus grande résolution ; cette résolution est déterminée par la fréquence d'échantillonnage des signaux.

Soient h_n et g_n des filtres tels que

$$h_n = 2^{1/2} \int \varphi(x-n) \varphi(2x) dx$$

$$\text{et } g_l = (-1)^l h_{-l+1}.$$

L'algorithme de décomposition multirésolution d'un signal est alors décrit formule (7) et figure 3 (pour de plus amples informations cf. [20])

$$(7) \quad c_{j,n} = \sum_k g_{2n-k} s_{j-1,k}$$

$$s_{j,n} = \sum_k h_{2n-k} s_{j-1,k}.$$

I. Daubechies a montré que les coefficients h_n correspondent aux coefficients α_n de la formule (5) [10].

La condition nécessaire et suffisante pour avoir une base orthogonale d'ondelettes est alors :

$$(8) \quad |H(\omega)|^2 + |H(\omega + \pi)|^2 = 2$$

$$G(\omega) = e^{-i\omega} \bar{H}(\omega + \pi).$$

La valeur 2 est due au coefficient de normalisation et H et G sont les transformées de Fourier respectives des filtres h et g . Ces 2 filtres orthogonaux H et G, de somme constante, décomposent le signal en deux sous-bandes. Ceci correspond bien aux CQF introduits par Smith et Barnwell [25].

Ces filtres, utilisés aussi pour la synthèse du signal, donnent alors une reconstruction exacte :

$$(9) \quad s_{j-1,k} = \sum_n [h_{2n-k} s_{j,n} + g_{2n-k} c_{j,n}].$$

Cependant, la décomposition sur une base d'ondelettes orthogonales permet, par rapport aux filtres CQF à reconstruction exacte, de transcrire le signal à décomposer comme une superposition d'ondelettes relativement douces. Ceci se traduit par la possibilité de choisir la régularité de l'ondelette. En effet, cette condition de régularité n'est pas satisfaite habituellement par les filtres à reconstruction exacte utilisés dans les algorithmes de codage en sous-bande.

Se référant aux travaux de Smith et Barnwell et I. Daubechies, il n'existe pas de filtre symétrique (à phase linéaire), à support compact, permettant la reconstruction exacte en conservant la condition d'orthogonalité. La mise en œuvre de filtres CQF courts à phase non linéaire pour le codage d'images conduit à de mauvais résultats (importance de la phase en traitement d'image). Les seules solutions acceptables en codage d'image sont, soit une solution approchée à filtres symétriques courts par filtres QMF (cf. travaux de Johnston 1980, Woods 1986, Biemond 1986), soit un filtre CQF de longueur infinie à reconstruction exacte (Ondelette de Meyer).

C'est cette deuxième solution, c'est-à-dire l'utilisation de l'ondelette de Meyer, que nous avons retenu.

2.4. PROBLÈMES DE BORDS

Le filtrage numérique pose toujours des problèmes de bord dus à la longueur spatiale du filtre.

La convolution que l'on est amené à effectuer ici est délicate du fait de la longueur de l'ondelette dont le support n'est pas compact et de la volonté de ne pas perdre d'information sur les bords de l'image.

Compte tenu du grand nombre de convolutions à effectuer nous avons choisi, pour réduire la charge de calcul, la méthode qui consiste à effectuer le produit des transformées de Fourier. Cela présente

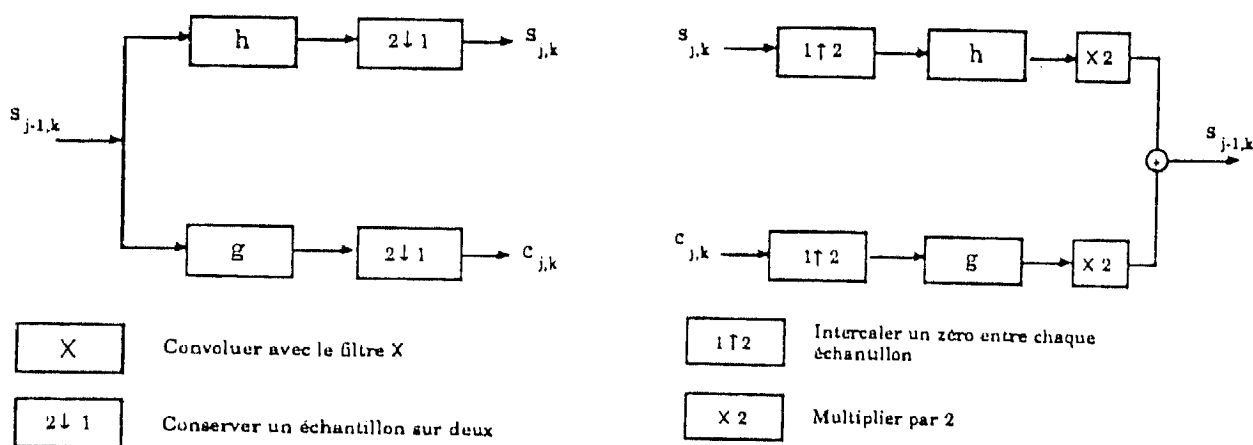


Fig. 3.

encore deux avantages : G et H sont définis en fréquence et sont périodiques par construction ; en effet ces filtres correspondent aux transformées de Fourier de g et h .

L'algorithme de FFT suppose la périodicité à la fois en espace et en fréquence, il génère donc des *signaux périodiques*.

L'image, lors de la transformation par FFT, est supposée périodique, ce qui n'est bien sûr pas le cas général.

Considérons par exemple une ligne de l'image ; la valeur de l'intensité des pixels est notée x_i où i représente l'indice de colonne :

$$x_1, x_2, x_3, \dots, x_{N-2}, x_{N-1}, x_N.$$

Il existe en général une discontinuité si en considérant la périodicité, on fait suivre x_N , de x_1, x_2, x_3, \dots . Cette discontinuité introduit de nombreuses fréquences parasites, lors de la FFT, qui perturbent l'algorithme de reconstruction [5].

Une solution que nous avons déjà préconisée [4], [5], [6] est de symétriser la ligne de la manière suivante :

$$x_1, x_2, x_3, \dots, x_{N-1}, x_N, z, x_N, x_{N-1}, \dots, x_3, x_2.$$

Cette nouvelle ligne est obtenue par recopie des valeurs de pixels de x_2 à x_N . La valeur du point z supplémentaire est interpolée, par exemple de manière polynomiale. La périodisation de cette ligne n'introduit plus de discontinuité sur le signal.

On obtient ainsi une ligne de longueur $(2N)$. Du fait de la propriété de symétrie qui a été imposée, la transformée de Fourier de la ligne est symétrique et réelle, ce qui n'augmente pas le nombre de données à traiter.

Ce prétraitement est effectué sur toutes les lignes et les colonnes lorsque cela est nécessaire.

2.5. APPLICATION DES ONDELETTES À L'ANALYSE D'IMAGE. EXTENSION AU CAS 2D

Il existe de nombreuses extensions de la transformée en ondelette 1D à plusieurs dimensions. Dans notre application, nous avons utilisé une transformée en ondelette qui privilégie les directions horizontales et verticales [20].

Dans un schéma d'analyse 2D par ondelette on introduit, comme dans le cas 1D, une fonction d'échelle $\Phi(x, y)$ séparable telle que :

$$(10) \quad \Phi(x, y) = \varphi(x) \varphi(y)$$

où $\varphi(x)$ est une fonction d'échelle 1D.

Soit $\psi(x)$ l'ondelette mono-dimensionnelle associée à la fonction d'échelle $\varphi(x)$. Alors, les trois ondelettes 2D sont définies de la manière suivante :

$$(11) \quad \begin{aligned} \Psi^H(x, y) &= \varphi(x) \psi(y) \\ \Psi^V(x, y) &= \psi(x) \varphi(y) \\ \Psi^D(x, y) &= \psi(x) \psi(y). \end{aligned}$$

La figure 4 représente un étage d'un schéma de décomposition pyramidale multiéchelle d'une image : les coefficients d'ondelette de l'image sont calculés, comme dans le cas 1D (§ 2.3 et § 2.4), à l'aide d'un algorithme de codage en sous-bande. Les filtres h et g sont des filtres mono-dimensionnels. Cette décomposition fournit des sous-images à différents niveaux de résolution et pour différentes orientations de contours (cf. *fig. 5*). Le schéma de reconstruction de l'image est présenté figure 6.

La transformation en ondelette d'une image permet d'obtenir la décomposition de celle-ci sur des sous-espaces orthogonaux ; chacune des sous-images obtenues est ainsi constituée par des coefficients

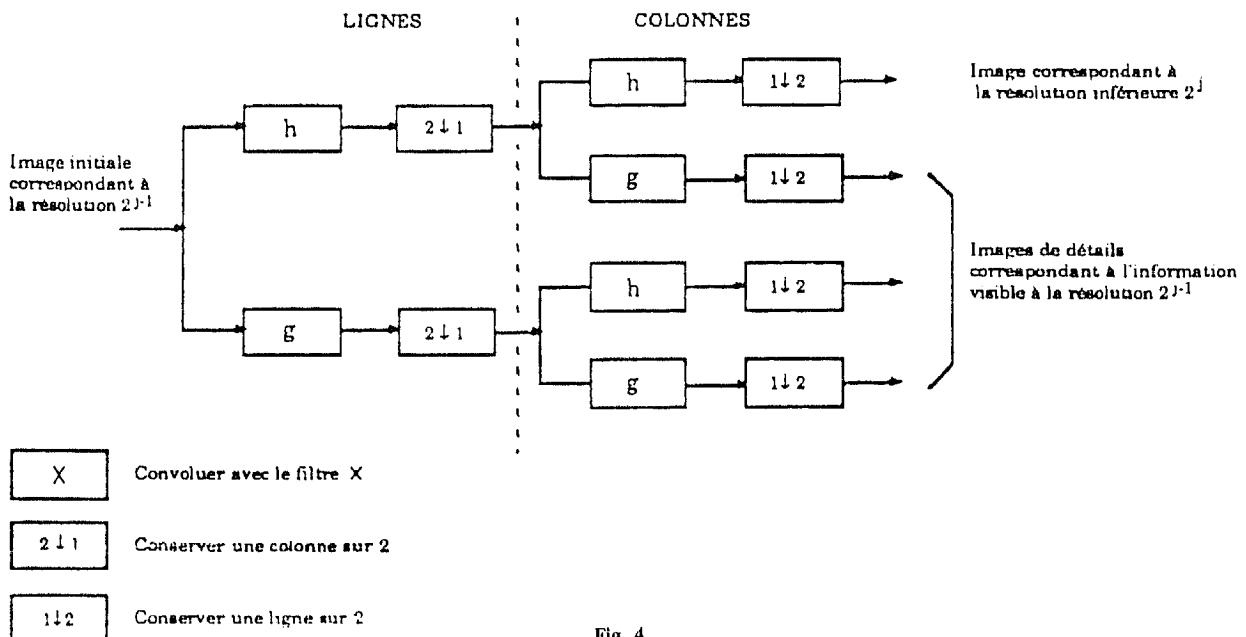


Fig. 4.

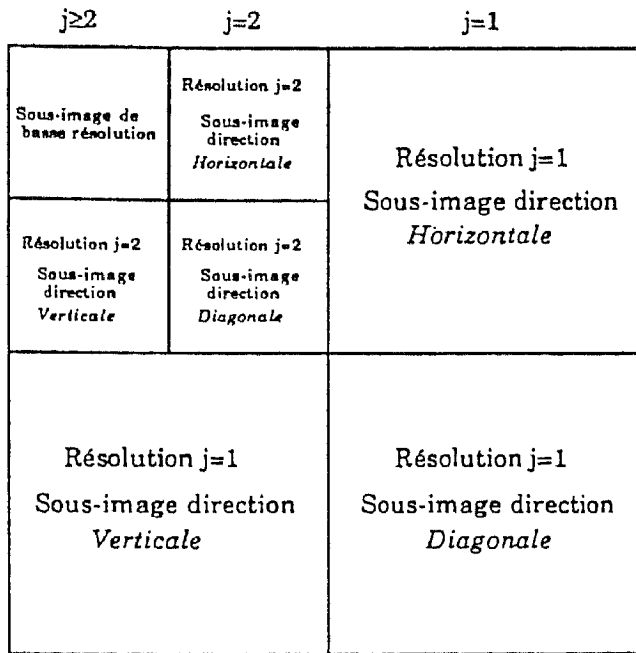


Fig. 5.

d'ondelette. Cette représentation est intéressante car elle révèle de l'information sur les contours, les textures orientées qui ne sont pas facilement accessibles dans l'image d'origine.

Lors de la compression, ce sont les coefficients d'ondelette, correspondant à chaque sous-image, qui sont codés par la méthode décrite dans le paragraphe suivant.

3. Codage des coefficients d'ondelette par quantification vectorielle

Pour coder les coefficients d'ondelette plusieurs techniques de compression peuvent être envisagées, techniques utilisant la quantification scalaire ou vectorielle.

Si l'on se réfère à la théorie sur le taux de distorsion introduite par Shannon, une meilleure performance est toujours réalisable en théorie si l'on code des vecteurs plutôt que des scalaires. Aussi, nous avons adopté pour notre application, la quantification vectorielle des coefficients d'ondelette.

3.1. PRINCIPE DE LA QUANTIFICATION VECTORIELLE

La quantification vectorielle (VQ), développée récemment par Gersho et Gray (1980) [13] [14], s'est avérée particulièrement performante pour la compression numérique des images. Son principe est de coder une séquence d'échantillons (vecteur) au lieu de coder chacun de ces échantillons individuellement. Le codage s'effectue en approximant la séquence à coder par un vecteur appartenant à un catalogue de formes usuellement désigné dans la littérature par le terme anglais codebook.

Le codebook utilisé est créé, et optimisé, par l'algorithme de classification de Linde, Buzo et Gray connu sous le nom de LBG [19]. Cet algorithme a pour but d'effectuer une opération de classification (création d'une partition) sur une séquence d'apprentissage constituée par des vecteurs appartenant à différentes images; il converge de manière itérative vers un codebook localement optimal. Chacun des vecteurs du codebook est indexé; lors du codage, c'est l'index du vecteur du codebook le

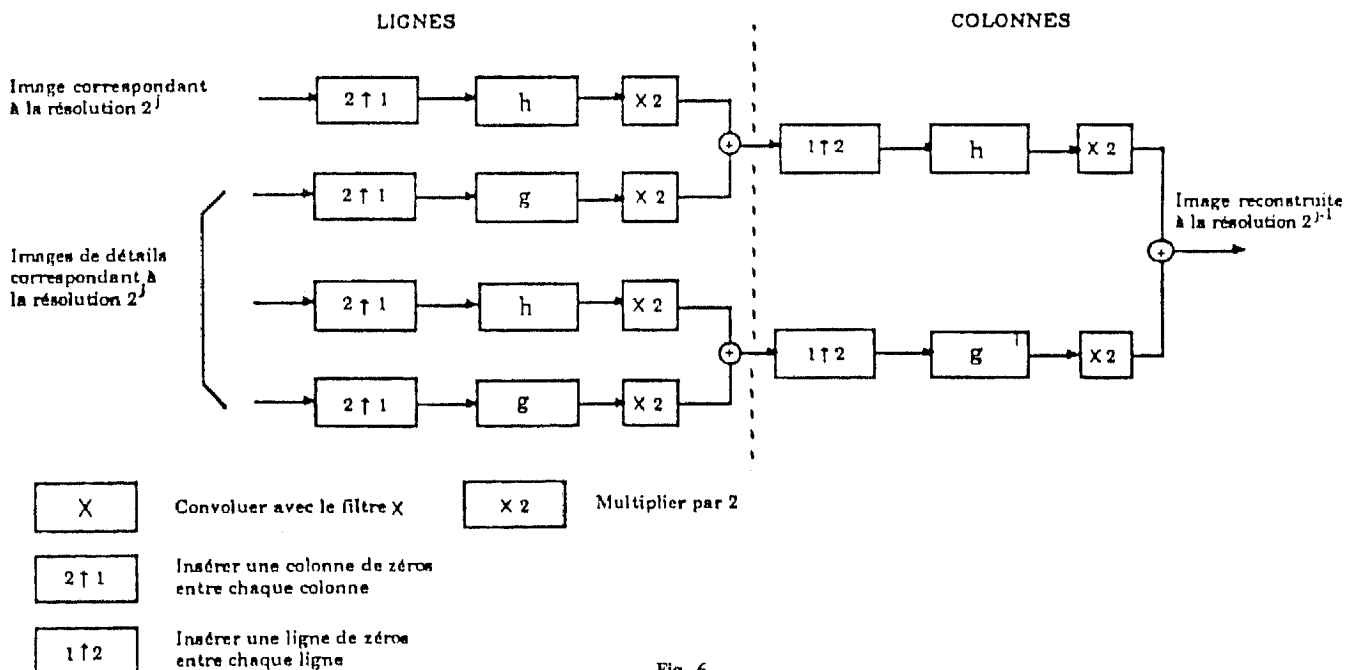


Fig. 6.

plus « proche » (au sens d'un certain critère) de la séquence d'échantillons à coder, qui servira à la représenter. Outre cet index, l'écart type de la séquence à coder est pris en compte. L'index donnera de l'information au niveau forme, et l'écart type au niveau énergie. Il est évident que pour reconstituer la séquence, le décodeur doit être muni du même codebook que le codeur.

L'index et l'écart type étant codés sur peu de bits par rapport au nombre de bits de codage d'un échantillon, il en résulte une compression au niveau des bits transmis ou stockés. Le facteur de compression Γ est alors :

$$(12) \quad \Gamma = (\log_2 I)/R$$

avec :

$$(13) \quad R = (\log_2 V + \log_2 E)/k \text{ bits/pixel (bpp)}$$

où :

- k = dimension des vecteurs
- I = nombre de niveaux de gris de l'image (en général $I = 256$ niveaux)
- V = nombre de vecteurs dans le codebook
- E = nombre maximal de niveaux considérés pour l'écart type

Le critère utilisé lors de la classification et du codage est la mesure du carré de l'erreur (norme Euclidienne au carrée) qui est la plus utilisée pour le codage des formes de signaux :

$$(14) \quad d(X, Y) = \|X - Y\|^2 = \sum_{j=1}^k (X(j) - Y(j))^2.$$

Pour comparer la quantification vectorielle avec la quantification scalaire (SQ), une étude est faite, dans le paragraphe suivant, sur les histogrammes normalisés (fonctions de densité de probabilité) des sous-images obtenues par décomposition en ondelette.

3.2. RÉPARTITION DES COEFFICIENTS D'ONDELETTE. COMPARAISON DES PERFORMANCES DE LA VQ PAR RAPPORT A LA SQ

3.2.1. Répartition des coefficients d'ondelette

Un des principaux avantages de la transformation en ondelette est qu'elle décompose une image de résolution donnée en sous-images décorréliées ; chacune de ces sous-images (résolution-direction) est alors codée séparément de manière plus fidèle que si l'on effectuait un codage brutal sur toute l'image. En effet, le type de codeur utilisé pour un niveau de résolution et une direction donnés peut être déterminé par les statistiques de la sous-image correspondante, c'est-à-dire par sa fonction de densité de probabilité (PDF).

Une PDF typique d'une sous-image est présentée figure 8, en correspondance avec la PDF de l'image d'origine Lena figure 7. Elle correspond à l'indice de résolution $j = 3$ et à la direction diagonale.

Comparativement avec la PDF multimodale de l'image d'origine ces PDF sont unimodales et très

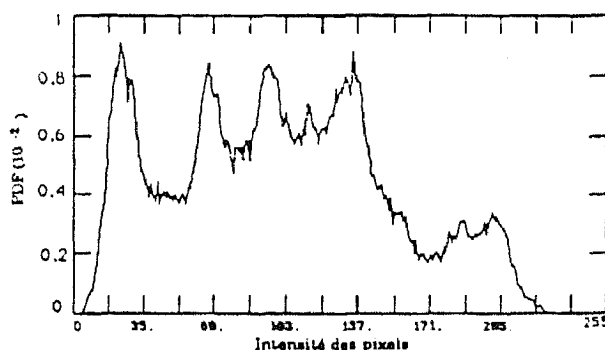


Fig. 7. — PDF originale (Lena).

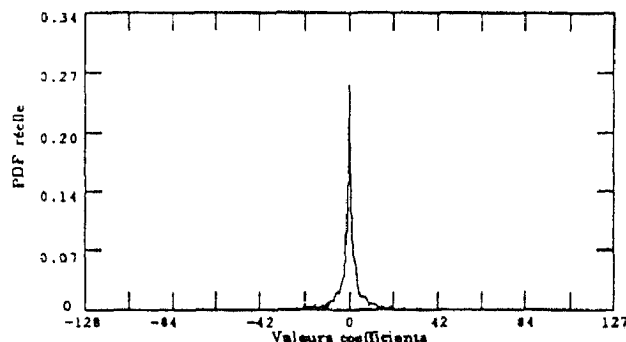


Fig. 8. — $j = 3, d = D$.

pointues (variance faible) et ont une moyenne nulle : ceci montre que dans chaque sous-image il y a un grand nombre de coefficients d'ondelette de faible valeur.

La figure 9 montre la comparaison entre une PDF réelle et son approximation par une loi de Laplace. La loi de Laplace est donnée par la formule (15) :

$$(15) \quad P_{j,d}(x) = \frac{\lambda_{j,d}}{2} e^{-\lambda_{j,d} |x|}$$

$\lambda_{j,d}$ = paramètre de la loi pour la sous-image d'indice de résolution j correspondant à la direction d , calculé par rapport à la valeur maximale de la PDF.

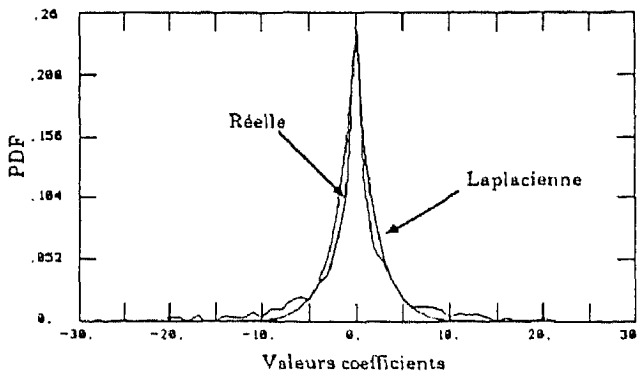


Fig. 9. — Comparaison : $j = 3, d = D, \lambda_{3,D} = 0,51$.

3.2.2. Performances de la VQ par rapport à la SQ

Dans ce paragraphe, nous justifions le choix de notre méthode de codage en calculant le gain en distorsion obtenu lorsque l'on fait de la quantification vectorielle dans une sous-image plutôt que de la quantification scalaire.

Soit $D_{j,d}(R_{j,d})$ la distorsion moyenne (erreur quadratique moyenne) de codage dans la sous-image (j,d) pour un taux $R_{j,d}$ de bits par pixel :

$$(16) \quad D_{i,j}(R_{j,d}) = E(|x - q(x)|^c) \quad c = 2$$

pour tous les coefficients x qui appartiennent à la sous-image, $q(x)$ étant la quantification de x .

La distorsion totale moyenne sur l'image, pour un taux total R_T de bits par pixel est alors définie par :

$$(17) \quad D_T(R_T) = (1/2^{2J}) \times D_J^{SQ}(R_J) + \sum_{j=1}^J \left[\sum_{d \in \{H, V, D\}} (1/2^{2J}) \times D_{j,d}(R_{j,d}) \right]$$

avec $D_J^{SQ}(R_J)$ correspondant à la distorsion dans la sous-image de plus basse résolution 2^J .

Il faut cependant remarquer que la formule (17) n'est valable que pour $c = 2$ car elle est une conséquence de l'orthogonalité des ondelettes $\Psi(x, y)$ [20].

La minimisation de cette distorsion $D_T(R_T)$ conduit à appliquer un bruit de quantification uniforme sur toute l'image donnant un bon rapport signal à bruit (défini formule (25)), mais une image de qualité visuelle moins bonne que si l'on effectue du « noise shaping » pour un même taux R_T de bits par pixel [9]. Le « noise shaping » consiste en effet à répartir le bruit de quantification en fonction des sous-images (j, d) codées, conduisant à minimiser chaque distorsion $D_{j,d}(R_{j,d})$.

Ceci nous incite à calculer le gain en distorsion $G_{j,d}^{VQ}$ pour la sous-image (j, d) codée. Ce gain correspond au rapport entre la distorsion obtenue par un quantificateur scalaire et celle obtenue par un quantificateur vectoriel pour un même taux $R_{j,d}$. Cette quantité est exprimée par :

$$(18) \quad G_{j,d}^{VQ} = \frac{D_{j,d}^{SQ}(R_{j,d})}{D_{j,d}^{VQ}(R_{j,d})}$$

Si l'on se réfère aux travaux d'Algazi [3], la distorsion fournie par le quantificateur scalaire dans la sous-image (j, d) pour un taux $R_{j,d}$ de bits par pixel est donnée par la formule :

$$(19) \quad D_{j,d}^{SQ} = \frac{2^{-c}}{c+1} 2^{-cR_{j,d}} \left\{ \int \{P_{j,d}(x)\}^{1/(c+1)} dx \right\}^{(c+1)}$$

avec $c \geq 1$.

De même, pour un quantificateur vectoriel, la distorsion correspondant à des vecteurs de dimension k , pour un taux $R_{j,d}$ de bits par pixel ($2^{kR_{j,d}}$ vecteurs

dans le codebook), est calculée par la formule de Zador [31], [8], [12] :

$$(20) \quad D_{j,d}^{VQ} = A(k, c) 2^{-cR_{j,d}} \times \left\{ \int \{P_{j,d}(x)\}^{k/(c+k)} dx \right\}^{(c+k)/k}$$

avec $c \geq 1$.

La constante $A(k, c)$ est fonction de la dimension k des vecteurs et de la valeur de c ($c = 2$ car nous avons choisi comme mesure de distorsion l'erreur quadratique moyenne). Pour $c = 2$, les valeurs de $A(k, 2)$ que nous utilisons ont été calculées et tabulées par Conway et Sloane [8].

Le gain défini en (18) peut alors être exprimé par :

$$(21) \quad G_{j,d}^{VQ} = \frac{2^{-c}}{(c+1) A(k, c)} \times \frac{\left\{ \int \{P_{j,d}(x)\}^{1/(c+1)} dx \right\}^{(c+1)}}{\left\{ \int \{P_{j,d}(x)\}^{k/(c+k)} dx \right\}^{(c+k)/k}}$$

Nous pouvons remarquer que $G_{j,d}^{VQ}$ ne dépend pas du taux $R_{j,d}$ affecté à la sous-image (j, d) . Cependant, compte tenu des approximations faites pour le calcul des formules de distorsion (19) et (20), la formule du gain $G_{j,d}^{VQ}$ s'applique seulement lorsque ce taux $R_{j,d}$ est grand (faible bruit de surcharge) et ne donne donc qu'une indication asymptotique.

Nous avons évalué ce gain $G_{j,d}^{VQ}$ dans le cas particulier d'une PDF correspondant à la loi de Laplace (15). La formule de ce gain est la suivante :

$$(22) \quad G_{j,d}^{VQ} = \frac{2^{-c/k} (c+1)^c}{A(k, c) \left(\frac{c}{k} + 1 \right)^{(c+k)/k}} \lambda_{j,d}^{-c(k-1)/k}$$

$G_{j,d}^{VQ}$ dépend à la fois du paramètre $\lambda_{j,d}$ de la PDF et de la dimension k des vecteurs du quantificateur vectoriel. La figure 10 montre l'évolution du gain théorique $G_{j,d}^{VQ}$, exprimé en dB, en fonction de

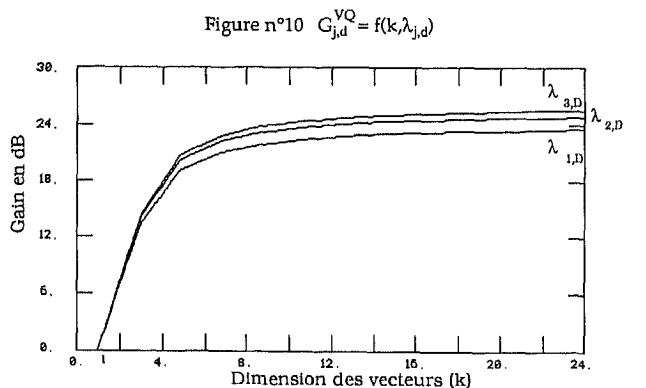


Fig. 10. — $G_{j,d}^{VQ} = f(k, \lambda_{j,d})$.

k pour différentes sous-images (courbes paramétrées par les valeurs de $\lambda_{j,d}$). Les valeurs de $A(k, 2)$ utilisées correspondent aux limites supérieures de l'erreur quadratique de codage [8] : Ceci nous donne une indication sur le gain théorique minimum $G_{j,d}^{VQ}$ que l'on peut atteindre.

3.3. GÉNÉRATION D'UN CODEBOOK MULTIRÉSOLUTION

Nous avons montré dans le paragraphe précédent l'intérêt que présente la VQ. Toutefois, des problèmes importants sont rencontrés en quantification vectorielle d'image :

- Impossibilité d'obtenir un codebook universel (efficace pour chaque image à coder) ;
- Hautes fréquences lissées par l'algorithme LBG (perte en résolution) ;
- Compromis : faible distorsion et compression élevée (coût en calculs) ;
- Difficultés de tenir compte des propriétés du système visuel humain [24], [21].

Une solution à ces difficultés est apportée par la transformée en ondelette, c'est-à-dire par la multirésolution.

En effet, la décomposition en ondelette d'une image permet d'élaborer un codebook de vecteurs bidimensionnels pour chaque niveau de résolution et de direction privilégiée (H, V, D). Chacun de ces codebooks est créé par l'algorithme LBG :

- La séquence d'apprentissage utilisée est constituée par des vecteurs qui appartiennent à différentes images correspondant au niveau de résolution et à la direction considérés ;
- Le codebook initial de l'algorithme est créé par splitting [14] du centroid (centre de gravité) de cette séquence d'apprentissage.

On peut alors constituer un codebook général par assemblage de ces différents codebooks ainsi obtenus. Chacun des codebooks est de faible distorsion et contient peu de mots ; il en résulte un gain évident au niveau recherche du meilleur vecteur de codage ainsi que de la qualité de l'image codée puisque seul le codebook concerné (résolution-direction) est scruté.

Ce codebook général se présente de la manière suivante (fig. 11) :

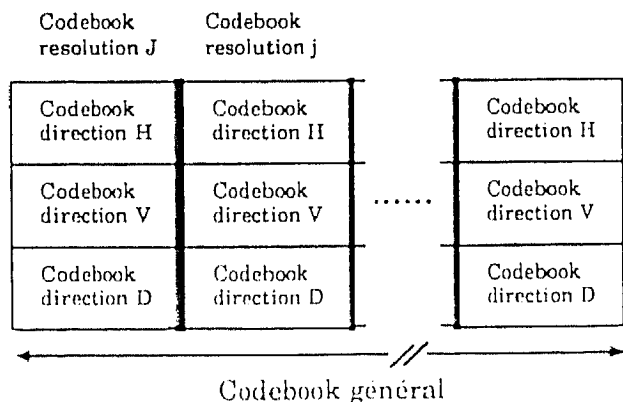


Fig. 11.

Chaque codebook correspondant à un niveau de résolution et à une direction donnés contient les formes de coefficients d'ondelette les plus représentatives au sens du critère défini par l'équation (14) ci-dessus.

L'aspect spatio-fréquentiel de l'image étant pris en compte lors de la décomposition en ondelette, le classement et la recherche lors du codage d'un vecteur appartenant à un sous-espace, peuvent en effet être effectués à l'aide d'un critère simple de type moindres carrés. On peut donc s'affranchir de l'utilisation de mesures de distorsion de types moindres carrés pondérés ou autres mesures faisant intervenir des facteurs perceptuels. En effet, ces algorithmes s'avèrent trop coûteux en temps calcul.

3.4. ALLOCATION ADAPTÉE DU NOMBRE DE BITS DE CODAGE

La multirésolution exploite les effets de masquage de l'œil [21] et donc, nous permet d'affiner et de choisir le type de codage en fonction du niveau de résolution (« noise shaping »). Le nombre de bits de codage d'un vecteur du codebook, ainsi que sa dimension, sont alors fonction de la résolution codée :

- Peu de bits sont nécessaires pour la haute fréquence (indice de résolution $j = 1$, contours très fins et en général moins importants d'un point de vue perceptuel que les basses et moyennes fréquences) ;
- Nombre de bits croissant avec j , donc avec la diminution de la résolution. Un quantificateur scalaire est utilisé pour la plus basse résolution.

Le facteur de compression total sur l'image est alors donné par la formule :

$$(23) \quad \Gamma_T = (\log_2 I) / R_T$$

avec :

$$(24) \quad R_T = R_j^{SQ} / 2^{2J} + \sum_{j=1}^J \sum_{d=(H,V,D)} (R_{j,d} / 2^{2j})$$

- d correspond aux différentes directions codées par niveau de résolution (H, V et D).
- $R_{j,d}$ est défini comme R (Cf. formule (13)) mais pour chaque indice de résolution j et direction d .
- R_j^{SQ} est le nombre de bits de codage utilisés par le quantificateur scalaire pour les indices de résolution $j \geq J$ (faibles résolutions).

4. Résultats expérimentaux

Les images que nous utilisons sont des images noir et blanc échantillonnées, de dimension (M, N) (M = 256 lignes et N = 256 colonnes) dont l'intensité de chaque pixel est codée sur 256 niveaux de gris (8 bits/pixel). Elles sont présentées figures 12 et 13.

Pour évaluer numériquement les performances du codeur, nous avons calculé le rapport signal à bruit,



Fig. 12. — LENA.

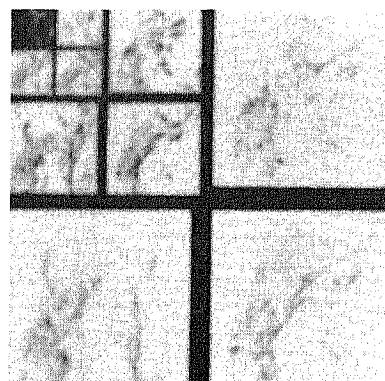


Fig. 14. — Décomposition de LENA.



Fig. 13. — CAMERAMAN.

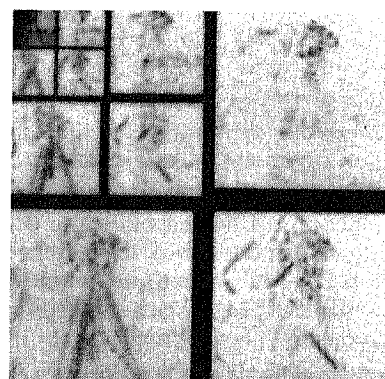


Fig. 15. — Décomposition de CAMERAMAN.

PSNR (Peak SNR), entre l'image d'origine $x(m, n)$ et l'image codée $\hat{x}(m, n)$ par la formule [17] :

$$(25) \text{PSNR}_{\text{dB}} = 10 \log_{10} \frac{255^2}{E[(x(m, n) - \hat{x}(m, n))^2]}$$

Lorsque l'on calcule ce SNR entre deux sous-images (j, d), il s'exprime par la formule (26) car le nombre de niveaux de gris est alors inférieur à 256.

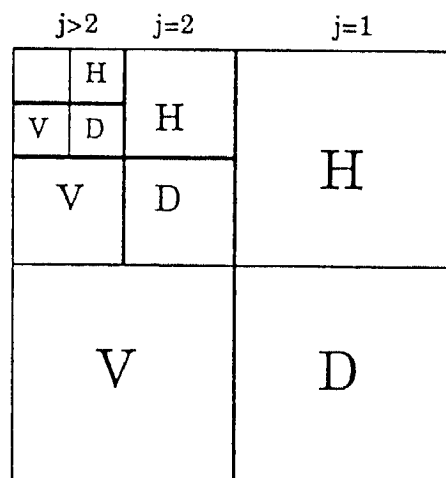
$$(26) \text{SNR}_{\text{dB}} = 10 \log_{10} \frac{E[x(m, n)^2]}{E[(x(m, n) - \hat{x}(m, n))^2]}$$

Dans la suite, les taux $R_{j,d}$ de bits par pixel ont été déterminés en fonction des sous-images (j, d) codées : Importance de la résolution considérée par rapport à la perception visuelle humaine (« noise shaping », cf. § 3.3 et 3.4).

4.1. EXEMPLE DE DÉCOMPOSITION EN ONDELETTE

La décomposition en ondelette des images à coder est effectuée par la méthode décrite au paragraphe 2.5. A titre d'exemple, les décompositions de l'image LENA (« The lady with hat ») et de l'image CAMERAMAN sont présentées figures 14 et 15 à différents niveaux de résolution : $j = 1$, $j = 2$ et $j > 2$ (valeur absolue et inverse video). La disposi-

tion des sous-images est faite selon le schéma suivant :



4.2. QUANTIFICATEUR VECTORIEL

Le quantificateur vectoriel que nous avons utilisé est un quantificateur « Full search », c'est-à-dire que lors du codage, tous les vecteurs du codebook, correspondant à la résolution et à la direction

codées, sont scrutés. Le critère de sélection (distorsion) utilisé est la mesure du carré de l'erreur défini formule (14).

Ces codebooks sont créés par la méthode LBG décrite aux paragraphes 3.1 et 3.3 pour chaque niveau de résolution ; ils correspondent aux directions horizontale (H), verticale (V) et diagonale (D) (cf. fig. 11). Les séquences d'apprentissage utilisées pour la création de ces codebooks sont constituées chacune par les trois images : Lena, Cameraman et l'image du Bateau aux résolutions correspondantes.

Nous avons utilisé dans nos essais expérimentaux deux types de quantificateur vectoriel.

Dans le but de vérifier la formule théorique du gain $G_{j,d}^{VQ}$, c'est-à-dire de se placer dans le cas développé par Zador [31], nous avons d'abord effectué des tests de codage avec un quantificateur vectoriel travaillant sur des vecteurs non normalisés : La variance n'est pas codée séparément de la forme. Les codebooks de formes utilisés, pour l'indice de résolution $j = 2$, contiennent 256 vecteurs de dimension (2×2) , dont l'index est codé sur 8 bits, ce qui correspond à des taux $R_{2,d}$ égaux à 2 bits par pixel ($\Gamma_{2,d} = 4$, cf. § 4.3, cas n° 1).

Par souci de gain de temps calcul, nous avons testé une deuxième méthode (cas n° 2), présentée au paragraphe 3.1, qui considère à la fois la forme et la variance des vecteurs. Chacun des codebooks, pour l'indice de résolution $j = 2$, contient 64 vecteurs bidimensionnels (2×2) : L'index de ces vecteurs est donc codé sur 6 bits. Le nombre maximal de niveaux de l'écart-type des vecteurs est pris égal à 64 ($E = 64$), ce qui nécessite aussi 6 bits de codage, 12 bits de codage sont donc affectés à un vecteur. Les taux $R_{2,d}$ sont alors égaux à 3 bits par pixel, ce qui correspond à des compressions $\Gamma_{2,d}$ égales à 2,7. Les taux de compression donnés dans cet article ne tiennent pas compte de l'entropie du signal. Par exemple, pour la résolution correspondant à $j = 2$, 8 bits ont été utilisés pour coder l'index d'un vecteur des codebooks (256 vecteurs dans les codebooks pour cette résolution). Or, l'entropie (définie par Shannon) des codebooks, pour l'image Lena montrée dans cet article et pour $j = 2$, est typiquement de 5,3 bits par vecteur. Un codage entropique augmenterait alors d'environ 20 % le taux de compression sur l'image totale, ce qui ferait par exemple pour l'image Lena, un taux de 0,69 bit par pixel au lieu de 0,87 bit par pixel.

4.3. COMPARAISON QUANTIFICATION VECTORIELLE PAR RAPPORT A LA QUANTIFICATION SCALAIRE

4.3.1. Quantification vectorielle de la forme des vecteurs uniquement

Nous avons comparé la quantification vectorielle avec la quantification scalaire dans les sous-images d'indice de résolution $j = 2$ et $j = 3$. Les résultats pour la direction $d = D$ et pour des vecteurs de dimension k égale à 4 sont présentés dans le tableau 1.

Le calcul de Zador donnant la formule de distorsion $D_{j,d}^{VQ}(20)$ a été effectué pour un nombre de vecteurs

Tableau 1

$G_{j,d}^{VQ}$ (dB)	Gains théoriques	Gains expérimentaux	
		Cas n°1 VQ Forme	Cas n°2 VQ forme + σ
$j=2$ $d=D$	20,25 dB	10,24 dB	6,75 dB
$j=3$ $d=D$	20,83 dB	13,28 dB	6,08 dB

dans le codebook égal à $2^{kR_{j,d}}$. Ceci correspond aux valeurs de gain en distorsion du cas n° 1 présentées tableau 1. Le cas n° 2 correspond à un codage séparé de la forme et de la variance pour un nombre de vecteurs dans le codebook égal à $2^{kR'_{j,d}/2}$ ($kR'_{j,d}/2$ bits sont affectés au codage de la variance).

Ces résultats nous montrent que les quantificateurs vectoriels retenus ne sont pas optimaux et que les valeurs théoriques de $G_{j,d}^{VQ}$ estimées dans le paragraphe 3.2.2 sont trop optimistes. Ceci est dû d'une part au fait que nous utilisons des faibles valeurs de $R_{j,d}$, que les codebooks ne sont pas optimaux et d'autre part que nous approximons les PDF par des lois de Laplace. Toutefois, dans les deux cas, les résultats obtenus par la VQ sont meilleurs que ceux obtenus par la SQ, aussi bien sur le plan numérique que visuel comme nous le montrent les images des figures 16 à 20. Le gain maximal en PSNR (défini formule (25)) que nous avons obtenu par VQ (par rapport à la SQ) est environ de 4 dB sur l'image totale.

Cependant, nous pouvons remarquer que le gain théorique est mieux approché dans le cas n° 1, c'est-à-dire lorsque l'on ne code pas séparément la variance et la forme, mais au prix d'un fort coût en calcul puisque beaucoup de vecteurs sont scrutés lors du codage (256 au lieu de 64). Une image codée par un tel quantificateur est présentée figure 16 pour un taux R_T égal à 0,87 bit par pixel.



Fig. 16. — $R_T = 0,87$ bpp PSNR = 30,31 dB.

L'affectation des taux $R_{j,d}$ ($d = H, V, D$) dans les différentes sous-images est la suivante :

- $R_2^{SQ} = 8$ bits/pixel → Quantificateur Scalaire
- $R_{2,d} = 2$ bits/pixel → Quantificateur Vectoriel
- $R_{1,d} = 0$ bit/pixel

4.3.2. *Quantification vectorielle de la forme des vecteurs avec codage séparé de leur variance*

Nous présentons dans cette partie de l'article et, dans le paragraphe 4.4, des résultats obtenus avec un quantificateur vectoriel considérant la variance des vecteurs. Nous avons retenu dans un premier temps ce quantificateur sous-optimal car il permet un codage rapide des images (peu de vecteurs (64) sont scrutés lors du codage) bien que le PSNR obtenu par cette méthode reste inférieur à celui obtenu lors d'un simple codage de formes, avec un taux plus faible de bits par vecteurs.



Fig. 17. — Vectoriel. $R_T = 1,06$ bpp PSNR = 30,14 dB.



Fig. 18. — Scalaire. $R_T = 1,06$ bpp PSNR = 27,32 dB.

Les images figures 17 et 18 sont des images codées avec un taux $R_T = 1,06$ bits/pixel. L'affectation des taux $R_{j,d}$ ($d = H, V, D$) dans les différentes sous-images est la suivante :

$R_2^{SQ} = 8$ bits/pixel → Quantificateur Scalaire

$R_{2,d} = 3$ bits/pixel → Quantificateur:

- Vectoriel figure 17
- Scalaire figure 18

$R_{1,d} = 0$ bit/pixel

De même, les images figures 19 et 20 sont codées avec un taux $R_T = 1,23$ bits/pixel. L'affectation des taux $R_{j,d}$ ($d = H, V, D$) est la suivante :

$R_3^{SQ} = 8$ bits/pixel → Quantificateur Scalaire

$R_{3,d} = 3$ bits/pixel → Quantificateur:

- Vectoriel figure 19
- Scalaire figure 20

$R_{2,d} = 3$ bits/pixel → Quantificateur Vectoriel



Fig. 19. — Vectoriel. $R_T = 1,23$ bpp PSNR = 30,57 dB.



Fig. 20. — Scalaire. $R_T = 1,23$ bpp PSNR = 26,04 dB.

$R_{1,d} = 0,54$ bit/pixel : compte tenu de la perception visuelle humaine, la sous-image correspondant à cette résolution d'indice $j = 1$ et à la direction Diagonale est supprimée, les deux autres sous-images (direction Horizontale et Verticale) sont seuillées par rapport à leur variance minimale. Seuls les pixels supérieurs à ce seuil sont codés en amplitude (6 bits) et en position (14 bits).

4.4. QUELQUES RÉSULTATS D'IMAGES COMPRESSÉES

Des résultats obtenus sur les images LENA et CAMERAMAN, pour différents taux de compression, sont présentés dans ce paragraphe.

L'affectation des taux $R_{j,d}$ est effectuée comme dans le paragraphe 4.3, c'est-à-dire de manière subjective.

Les figures 21, 22 et 23 correspondent respectivement à l'image LENA codée avec des taux R_T égaux à 0,83 bit par pixel (résolution d'indice $j = 1$ supprimée), 0,97 bit par pixel (résolution d'indice $j = 1$ supprimée) et 1,37 bits par pixel (résolution d'indice $j = 1$ seuillée codée, $R_{1,d} = 0,54$ bit/pixel).

La figure 24 correspond à l'image du Cameraman codée avec un taux R_T égal à 1,64 bits par pixel (résolution d'indice $j = 1$ seuillée codée, $R_{1,d} = 0,9$ bit/pixel).



Fig. 21. — $R_T = 0,83$ bpp PSNR = 28,79 dB.



Fig. 22. — $R_T = 0,97$ bpp PSNR = 29,86 dB.



Fig. 23. — $R_T = 1,37$ bpp PSNR = 32,31 dB.



Fig. 24. — $R_T = 1,64$ bpp PSNR = 26,61 dB.

Nous pouvons remarquer que les dégradations dues au codage sont visibles sur les contours de haute résolution ($j = 1, j = 2$) car c'est dans les hautes fréquences que nous avons effectué les plus fortes compressions. Ces erreurs sont dans certains cas imperceptibles par l'œil humain, ceci est un des avantages apporté par l'utilisation de la décomposition en ondelette.

4.5. PERFORMANCES DU CODEBOOK

Un problème important en quantification vectorielle est l'universalité du codebook. C'est-à-dire, la possi-



Fig. 25. — Image du CCETT. 256×256 pixels codés sur 8 bits.



Fig. 26. — Image codée. $R_T = 0,87$ bpp PSNR = 31 dB.

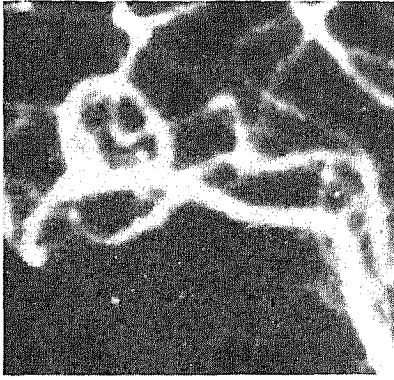


Fig. 27.— ANGIOGRAPHIE. 256 x 256 pixels codés sur 8 bits.



Fig. 28. — Image figure 28 filtrée par un FILTRE MEDIAN 3 x 3 ⇒ diminution du bruit impulsif.



Fig. 29. — Image figure 28 codée $R_T = 0,87$ bpp PSNR = 36,68 dB.

bilité de coder plusieurs types différents d'images, n'appartenant pas à la séquence d'apprentissage, avec le même codebook.

Nous proposons dans cette partie quelques résultats de codage vectoriel d'images de types variés, n'appartenant pas à la séquence d'apprentissage. Le quantificateur vectoriel utilisé n'effectue qu'un simple codage de forme (cf. § 4.3.1).

5. Conclusion et perspectives

Dans cet article, nous avons décrit un nouveau schéma de codage d'images qui combine la transformée en ondelette et la quantification vectorielle. La transformée en ondelette utilisée ici nous a permis de prendre en compte l'effet de masquage de l'œil, ce qui nous a conduit à de premiers résultats très encourageants. En effet, la méthode proposée permet pour des taux de compression importants de conserver une bonne qualité visuelle grâce à une procédure d'allocation de bits dans les sous-images (« noise shaping »). Les effets de blocs qui apparaissent généralement lorsque l'on effectue de la VQ spatiale (problèmes de moyenne) ne se produisent pas ici. La transformée en ondelette concentre la moyenne dans les basses résolutions, ce qui permet de construire des quantificateurs vectoriels pour des sous-images haute résolution de moyenne nulle.

Enfin, la complexité du codeur est cependant très faible puisque les codebooks utilisés sont de petite dimension, $R \leq 12$ bits par vecteur qui peuvent être répartis entre index et variance pour diminuer la charge de calculs.

Le choix de l'ondelette est crucial. C'est à l'heure actuelle un problème ouvert qui fait l'objet de recherches s'appuyant sur les critères suivants : Support de l'ondelette compact (faible coût calcul), filtre symétrique, adaptation au signal d'image (PDF), adaptation au système visuel humain...

On s'efforcera alors de trouver le meilleur compromis.

Enfin, l'autre axe de recherche actuellement développé est la prise en compte de la corrélation « visuelle » des coefficients d'ondelette entre deux niveaux de résolution.

Manuscrit reçu le 21 mai 1989.

Bibliographie

- [1] E. H. ADELSON, E. SIMONCELLI, R. HINGORANI, Orthogonal Pyramid Transforms for Image Coding, *SPIE Visual Commun. and Image Proc. II* (1987), vol. 845, pp. 50-58.
- [2] N. AHMED, T. NATARAJAN and K. R. RAO, Discrete cosine transform, *IEEE Trans. Comput.*, vol. C-23, pp. 90-93, 1974.
- [3] V. R. ALGAZI, Useful Approximation to optimum Quantization, *IEEE Trans. commun.*, vol. COM-14, No. 3, pp. 297-301, June 1966.
- [4] M. BARLAUD, L. BLANC-FÉRAUD, P. MATHIEU, J. MENEZ, M. ANTONINI, *2D Linear Predictive Image Coding With Vector Quantization*, EUSIPCO, 88.
- [5] L. BLANC-FÉRAUD, M. BARLAUD, P. MATHIEU, Amélioration de la restauration d'images floues par un filtrage de Kalman utilisant une image miroir, *Traitement du Signal*, vol. 5 (1988) pp. 249-261.
- [6] L. BLANC-FÉRAUD, M. BARLAUD, P. MATHIEU, Ringing Reduction In Images Restoration using mirror images and adaptive Kalman Filtering, ICASSP, 88.
- [7] A. CALDERON, Intermediate spaces and interpolation, the complex method, *Studia Math.*, 24, pp. 113-190 (1964).

- [8] J. H. CONWAY and N. J. A. SLOANE, A Lower Bound on the Average Error of Vector Quantizers, *IEEE Trans. on Inform. Theory*, vol. IT-31, No. 1, pp. 106-109, January 1985.
- [9] V. CUPERMAN, On Adaptive Vector Transform Quantization for Speech Coding, *IEEE Trans. on Comm.*, vol. 37, No. 3, March 1989.
- [10] I. DAUBECHIES, Orthonormal Bases of Compactly Supported Wavelets, *Communications on Pure and Applied Mathematics*, Vol. XLI pp. 909-996, 1988.
- [11] D. ESTEBAN, C. GALAND, Application of Quadrature Mirror Filters to split Band Voice Coding Systems in *Proc. Int. Conf. Acoust. Speech Signal Processing*, Washington, DC, April 1979.
- [12] A. GERSHO, Asymptotically Optimal Block Quantization, *IEEE Trans. on Inform. Theory*, vol. IT-25, No. 4, July 1979.
- [13] A. GERSHO, On the structure of Vector Quantizers, *IEEE Trans. on Inform. Theory*, vol. IT-28, No. 2, March 1982.
- [14] R. M. GRAY, Vector Quantization, *IEEE ASSP Mag.*, pp. 4-29, April 1984.
- [15] A. GROSSMANN, J. MORLET, Decomposition of Hardy functions into square integrable wavelets of constant shape, *SIAM J. Math.*, vol. 15, pp. 723-736, 1984.
- [16] A. K. JAIN, A Sinusoidal Family of Unitary Transforms, *IEEE Trans. Pattern analysis and machine intelligence*, vol. PAMI-1, 1979.
- [17] A. K. JAIN, Image Data Compression: A Review, *IEEE*, vol. 69, No. 3, March 1981.
- [18] P. G. LEMARIÉ and Y. MEYER, Ondelettes et bases hilbertiennes, *Revista Mathematica Ibero-America*, vol. 2, 1986.
- [19] Y. LINDE, A. BUZO, R. M. GRAY, An Algorithm for Vector Quantizer Design, *IEEE Trans. Commun.*, vol. COM-28, pp. 84-95, January 1980.
- [20] S. G. MALLAT, A Theory for Multiresolution Signal Decomposition: the Wavelet Representation, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 11, n° 7, July 1989.
- [21] D. MARR, *Vision*, W. H. Freeman and Company, New York, 1982.
- [22] Y. MEYER, *Principe d'incertitude, bases hilbertiennes et algèbre d'opérateurs*, Séminaire Bourbaki n° 662, 1985-86.
- [23] N. M. NASRABADI, R. A. KING, Image Coding Using Vector Quantization: A Review, *IEEE Trans. on Comm.*, vol. 36, No. 8, August 1988.
- [24] W. K. PRATT, *Digital Image Processing*, J. Wiley, New York, 1978.
- [25] M. J. SMITH and D. P. BARNWELL, Exact reconstruction for tree-structured subband coders. *IEEE Trans. on ASSP* 34, 434-441, 1986.
- [26] P. STROBACH, H. HÖLZIMMER, M. BUCHNER, A Design Technique for Generalized Zonal Masks for DCT-Coefficients, *Digital Signal Processing*, pp. 448-452 North-Holland, 1987.
- [27] B. V. K. VIJAYA KUMAR, C. P. NEUMAN, K. J. DE VOS, Discrete Wigner Synthesis, *Signal Processing*, 11, pp. 277-304, 1987.
- [28] P. H. WESTERINK, J. BIEMOND, D. E. BOEKEE, Evaluation of Image Subband Coding Schemes, *Signal Processing IV: Theories and Applications*, Euraspip, 1988.
- [29] P. H. WESTERINK, D. E. BOEKEE, J. BIEMOND, J. W. WOODS, Subband Coding of Image Using Vector Quantization, *IEEE Trans. Commun.*, vol. 36, pp. 713-719, 1988.
- [30] J. W. WOODS, S. D. O'NEIL, Subband Coding of Images, *IEEE Trans. on ASSP*, vol. 34, NO. 5, October 1986.
- [31] P. ZADOR, Asymptotic quantization error of continuous signals and their quantization dimension, *IEEE Trans. On Inform. Theory*, vol. IT-28, pp. 139-149, 1982.