

Bancs de filtres M -bandes et filtrage spatial

Les transformées en bancs de filtres M -bandes sont obtenues par extension du nombre de bandes mises en jeu dans un banc de filtres dyadique. De part leurs performances de décorrélation, leur flexibilité et leur grande sélectivité fréquentielle, certaines d'entre elles comme la DCT ou les transformées orthogonales à recouvrement sont très utilisées en compression d'image fixe. Nous nous intéressons dans ce chapitre à l'étude de leurs propriétés, à leur mise en œuvre dans le cadre de notre schéma de codage vidéo et nous présenterons alors un moyen d'étendre leurs propriétés de scalabilité.

Après avoir rappelé la définition des transformées M -bandes et énoncé quelques unes de leurs propriétés dans la section 5.1, nous décrivons les transformées de cette famille les plus communément utilisées en codage d'image fixe et en codage vidéo. Nous verrons alors comment leurs propriétés de sélectivité fréquentielle et leur généralité peuvent être mises à profit dans le cadre de notre schéma de codage vidéo.

Tout au long du chapitre précédent, nous avons étudié comment les images d'une séquence vidéo sont transformées par le filtre temporel afin de tirer parti de leur redondance temporelle. Les sous-bandes temporelles résultantes sont alors décomposées spatialement pour exploiter la redondance spatiale présente dans ces images. Nous nous intéressons tout d'abord dans la section 5.2 à l'étude des caractéristiques spatiales et fréquentielles de ces sous-bandes temporelles. L'utilisation d'un schéma générique de construction de transformées M -bandes nous permet alors de spécifier une transformée 4-bandes adaptée à la décorrélation des sous-bandes temporelles de détail. Nous mettons ensuite en œuvre cette transformée au sein des codecs MC-EZBC et Vidwav et observons l'efficacité de codage qu'elle offre. D'autres transformées M -bandes sont aussi utilisées et comparées à la transformée 9/7 dyadique.

En dépit de leurs avantages, les transformées M -bandes semblent cependant ne pas posséder des propriétés de scalabilité suffisamment fines pour permettre d'offrir des changements de résolutions de rapports variés. En effet, dans un banc de filtres M -bandes classique, la sous-bande résultant du filtrage passe-bas peut servir d'approximation de l'image originale à une résolution réduite d'un facteur M , offrant ainsi seulement une scalabilité d'ordre M . Afin de pallier ce problème, nous présentons enfin dans la section 5.3 comment le banc de filtres de synthèse peut être modifié de façon à donner aux transformées M -bandes des propriétés de scalabilité étendues.

5.1 Bancs de filtres M -bandes ; rappels

Nous rappelons dans cette section la définition du banc de filtres M -bandes, étudié en détail par Vaidyanathan [155] et énonçons plusieurs de ses propriétés. Des exemples de transformées M -bandes communément utilisées en codage d'image sont alors présentés

et nous décrivons les relations qu'entretiennent les bancs de filtres M -bandes avec les bases d'ondelettes M -bandes.

5.1.1 Définition

La transformée en banc de filtres M -bandes est obtenue par extension du nombre de bandes mises en jeu dans un banc de filtres dyadique. De part son nombre accru de sous-bandes, elle est alors plus flexible que ce dernier et possède une meilleure sélectivité fréquentielle. Tout comme le banc d'analyse et de synthèse dyadique, il est possible d'accoler un banc d'analyse et de synthèse M -bandes de manière à créer un schéma de compression où le banc d'analyse sera utilisé lors de l'encodage. Un tel banc d'analyse-synthèse M -bandes est illustré par la Fig. 5.1.

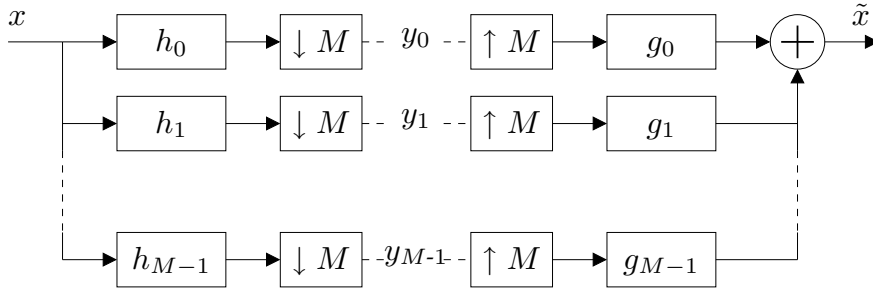


FIG. 5.1 – Banc de filtres d'analyse-synthèse M -bandes.

Dans le banc d'analyse, le signal d'entrée x est tout d'abord filtré par un jeu de M filtres d'analyse de réponses impulsionnelles $\{h_k\}_{0 \leq k < M}$. Chaque sous-bande résultante est alors sous-échantillonnée d'un facteur M par l'opérateur $[\downarrow M]$. Cet opérateur est défini dans le cas général par :

$$u = [\downarrow M]v \Leftrightarrow \forall n \in \mathbb{Z}, u[n] = v[nM] \quad (5.1)$$

On obtient alors les sous-bandes $\{y_k\}_{0 \leq k < M}$, résultant de la décomposition du signal x par le banc d'analyse. Tout comme dans le cas du banc de filtres dyadique, ces sous-bandes sont destinées à être quantifiées et codées par un codeur entropique.

La reconstruction du signal est effectuée par le banc de synthèse. Les sous-bandes $\{y_k\}_{0 \leq k < M}$ sont tout d'abord suréchantillonnées d'un facteur M par l'opérateur $[\uparrow M]$. Ce dernier est défini dans le cas général par :

$$u = [\uparrow M]v \Leftrightarrow \forall n \in \mathbb{Z}, u[n] = \begin{cases} v[n/M] & \text{si } n \text{ est divisible par } M \\ 0 & \text{sinon} \end{cases} \quad (5.2)$$

Les sous-bandes suréchantillonnées sont alors filtrées par le jeu de filtres de synthèse de réponses impulsionnelles $\{g_k\}_{0 \leq k < M}$. La sommation des signaux résultants du filtrage permet alors de reconstruire le signal \tilde{x} .

Les conditions de reconstruction parfaites permettant d'assurer $\tilde{x} = x$ ont été énoncées par Vaidyanathan [155]. Elles sont équivalentes à une condition de biorthogonalité sur les familles $\{h_k[n - Mt]\}_{t \in \mathbb{Z}}$ et $\{g_k[n - Mt]\}_{t \in \mathbb{Z}}$, et s'écrivent dans le domaine temporel sous la forme :

$$\forall k, \forall i, j \quad \sum_{n \in \mathbb{Z}} h_i(n) g_j(n - Mk) = \delta_k \delta_{i-j} \quad (5.3)$$

où δ est l'opérateur de Kronecker. Ces propriétés peuvent aussi s'énoncer dans le domaine fréquentiel par les relations :

$$\forall f, 0 \leq i, j < M, \quad \sum_{k=0}^{M-1} \widehat{h}_i\left(\frac{f+k}{M}\right) \widehat{g}_j^*\left(\frac{f+k}{M}\right) = M\delta_{i-j} \quad (5.4)$$

Du fait du nombre accru de sous-bandes mises en jeu dans le banc de filtres, les équations de reconstruction parfaite laissent davantage de liberté pour le choix des filtres $\{g_k\}_{0 \leq k < M}$ et $\{h_k\}_{0 \leq k < M}$, comparé au cas dyadique. C'est la raison de la flexibilité accrue des transformées M -bandes.

On dira que le banc de filtres M -bandes est orthogonal si pour tout $0 \leq k < M$, $g_k[n] = h_k[-n]$. En effet, pour tout $n \in \mathbb{Z}$, les familles $\{h_k[n - jM]\}_{j \in \mathbb{Z}}$ et $\{g_k[n - jM]\}_{j \in \mathbb{Z}}$ forment alors une base orthogonale et ceci implique l'orthogonalité de la transformée M -bandes. C'est donc une isométrie qui préserve la norme ℓ_2 et donc l'énergie d'un signal, assurant ainsi $\sum_n x(n)^2 = \sum_j \sum_n y_j(n)^2$.

On remarquera de plus que la décomposition en paquets d'ondelettes dyadiques en utilisant une base de décomposition uniforme, où toutes les sous-bandes sont de même taille, est une transformée M -bandes où M est une puissance de deux.

Ondelettes M -bandes

Nous avons vu dans la section 1.2.2 les liens étroits qu'entretiennent les bancs de filtres dyadiques et l'analyse multirésolution par ondelettes. Il est en effet possible de montrer que toute transformée en ondelettes discrètes de support compact peut être mise sous forme de banc de filtres.

Tout comme les bases d'ondelettes dyadiques rappelées en section 1.2.1, il est possible de construire des bases d'ondelettes M -bandes [135] en utilisant *plusieurs* espaces de détail $\{\mathbf{W}_j^p\}_{1 \leq p < M}$ au niveau j , pour représenter l'information de détail perdue entre deux niveaux de résolutions. Un espace de détail \mathbf{W}_j^p est alors engendré par les translatées de la p -ième ondelette mère ψ^p dilatée au niveau j et est donc défini par :

$$\mathbf{W}_j^p = \left\{ t \mapsto \frac{1}{M^{j/2}} \psi^p\left(\frac{t}{M^j} - k\right) \right\}_{k \in \mathbb{Z}} \quad (5.5)$$

Lors de l'analyse multirésolution d'un signal, la différence entre l'approximation sur \mathbf{V}_j et celle sur \mathbf{V}_{j+1} peut alors être représentée par un ensemble de fonctions appartenant aux espaces $\{\mathbf{W}_j^p\}_{1 \leq p < M}$. Il est alors possible de définir l'espace \mathbf{V}_j par sommation directe d'espaces vectoriels selon la relation :

$$\mathbf{V}_j = \mathbf{V}_{j+1} \oplus \left[\bigoplus_{p=1}^{M-1} \mathbf{W}_{j+1}^p \right] \quad (5.6)$$

Cependant, nous ne nous attarderons pas dans la suite du document sur les ondelettes M -bandes et nos constructions de transformées M -bandes reposeront uniquement sur la formulation en banc de filtres présentée ci-dessus.

5.1.2 Transformées en blocs

Les transformées en blocs peuvent être vues comme des transformées M -bandes dont les filtres $\{h_k\}_{0 \leq k < M}$ et $\{g_k\}_{0 \leq k < M}$ ont une réponse impulsionnelle de taille $L = M$

échantillons. Nous rappelons les propriétés de quelques transformées en blocs très utilisées en compression d'image et en codage vidéo.

Transformée en cosinus discrète - DCT

La transformée en cosinus discrète ou DCT (*Discrete Cosine Transform*) a été popularisée par Rao [11] et est couramment utilisée en compression d'image et en codage vidéo : elle est en effet à la base des codecs JPEG et MPEG. C'est une transformée orthogonale qui correspond à des filtres à phase linéaire et qui peut s'interpréter comme la transformée de Fourier discrète d'un signal fini prolongé symétriquement sur ses bords. La DCT est aussi une transformée M -bandes dont les filtres $\{h_k\}_{0 \leq k < M}$ de longueur $L = M$ échantillons sont donnés par :

$$\forall 0 \leq n < M, \quad h_k(n) = \lambda_k \sqrt{\frac{2}{M}} \cos \left[\frac{k\pi}{M} \left(n + \frac{1}{2} \right) \right] \quad \text{avec } \lambda_k = \begin{cases} 1/\sqrt{2} & \text{si } k=0 \\ 1 & \text{sinon} \end{cases}$$

La formulation en bancs de filtres n'est certainement pas la façon la plus efficace pour implémenter le calcul de la DCT et il existe de nombreux algorithmes rapides [36] visant à alléger la complexité de sa mise en œuvre. La forme 2D séparable de la DCT à 8-bandes est très utilisée en codage spatial pour décomposer les images car on peut montrer que son efficacité de décorrélacion est proche de celle de la KLT. C'est pourquoi la DCT est utilisée dans les normes JPEG, MPEG-1, MPEG-2, MPEG-4 Partie 2 et dans la série des codecs H.261 et H.263 pour décomposer les images en blocs avant quantification et codage.

En codage d'images, un des inconvénients majeurs de la DCT réside dans sa propension à créer des artefacts de type blocs lorsque elle est utilisée à bas débit. En effet, la DCT est une transformée locale qui n'opère que sur le bloc courant de taille $M \times M$ pixels, indépendamment des blocs voisins. Lors de la perte d'informations par quantification, chaque bloc est alors décodé indépendamment des autres et des discontinuités peuvent apparaître à leurs frontières. Les transformées orthogonales à recouvrement ont été introduites pour pallier à cet inconvénient et sont abordées dans la suite du document.

Transformée DCT entière et transformée d'Hadamard

La DCT est une transformation dont les coefficients des filtres d'analyse et de synthèse sont des nombres réels. Cependant, une implémentation logicielle ou matérielle de la DCT utilise nécessairement une approximation à précision finie de ces coefficients. Lors du déploiement des normes JPEG et MPEG, on a observé que ces approximations étaient la cause de dérives importantes entre différentes implémentations de codecs d'images ou vidéo utilisant la DCT. Ces dérives sont ainsi dues à des architectures logicielles et matérielles différentes qui utilisent des opérateurs de précision inégale.

Afin d'éviter ces problèmes de dérives, la norme H.264 a opté pour l'utilisation de transformées spatiales *entières* afin d'obtenir des résultats déterministes, indépendants d'une architecture matérielle ou d'une factorisation particulière de la transformée. La norme H.264 [118] utilise ainsi deux transformées 4-bandes spatiales : une transformée DCT entière et une transformée de Hadamard.

La transformée DCT entière utilisée résulte simplement de l'approximation entière de la DCT classique sur 4 bandes. Elle est utilisée sous forme séparable 2D pour décorrélacion spatialement les blocs issues des résidus de la prédiction temporelle du codec

H.264. Cette transformée s'écrit classiquement sous la forme matricielle séparable $y = H_{\text{dcte}} x H_{\text{dcte}}^T$, où x est un bloc de 4×4 pixels et y les coefficients transformés. Vu comme une transformée 4-bandes, les coefficients de ses filtres d'analyse sont donnés par les lignes de la matrice H_{dcte} suivante :

$$\forall 0 \leq n < 4, \quad h_k(n) = H_{\text{dcte}}(n, k) \quad \text{avec} \quad H_{\text{dcte}} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix}$$

La transformée de Hadamard est une autre transformée orthogonale utilisée par la norme H.264 pour transformer à nouveaux les coefficients de la composante basse fréquence issue de la transformée DCT entière. Elle s'écrit en utilisant la forme concise de cette dernière au moyen de la matrice H_{had} :

$$\forall 0 \leq n < 4, \quad h_k(n) = H_{\text{had}}(n, k) \quad \text{avec} \quad H_{\text{had}} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}$$

On notera l'existence de récentes alternatives à l'approximation de la DCT entière utilisée dans H.264, autorisant cependant une implémentation n'ayant pas recours à une arithmétique à virgule flottante. Ces travaux concernent la BinDCT de Tran [147], basée sur une factorisation directe en treillis de la DCT et d'autres travaux, dont la DCT entière de Chen [37]. Le sujet est cependant vaste et nous ne n'y attarderons pas.

5.1.3 Transformées à recouvrement

En dépit de leur efficacité de décorrélation et de leur simplicité d'utilisation, les transformées en blocs ont l'inconvénient de générer des artefacts aux bords des blocs très visibles lors de la quantification à bas débit. Afin d'amoindrir ces effets, plusieurs auteurs dont Malvar [82, 85] et Meyer [79] ont les premiers préconisé l'utilisation de transformées M -bandes à *recouvrement*, dont les supports des filtres ont une taille L supérieure au nombre de bandes M .

LOT, LBT et GenLOT

La transformée orthogonale à recouvrement, dénommée LOT (*Lapped Orthogonal Transform*), a été introduite et popularisée par Malvar [85]. C'est une modification de la DCT qui permet, contrairement à cette dernière, de décomposer un signal en blocs en utilisant le voisinage du bloc courant. Elle permet ainsi d'amoindrir nettement les problèmes d'artefacts de blocs très visibles lors de l'utilisation de la DCT avec une quantification à bas débit. La LOT est une transformée orthogonale, à phase linéaire dont la formulation classique est basée sur une modification du graphe de traitement de la DCT et est illustrée dans [85]. Elle peut cependant être représentée par un banc de filtres M -bandes dont les coefficients des filtres d'analyse $\{h_k\}_{0 \leq k < M}$ de support $L = 2M$ sont donnés dans [79].

La relaxation de la condition d'orthogonalité $g_k[n] = h_k[-n]$ a permis à Malvar [83] de construire une version biorthogonale de la LOT, la LBT (*Lapped Biorthogonal Transform*) ou transformée biorthogonale à recouvrement. La LBT est classiquement définie par une modification du graphe de traitement de la LOT, où les coefficients de la bande y_1 sont

multipliés par $\sqrt{2}$ à l'analyse et par $1/\sqrt{2}$ à la synthèse. Cette subtile modification permet à la LBT de projeter un signal sur une base dont les fonctions décroissent vers zéro en leurs bords. Cette propriété constitue une amélioration notable sur la LOT, dont les fonctions de base décroissent vers des valeurs *proches* de zéro. La LBT possède ainsi un meilleur gain de codage que la LOT ou la DCT et conduit expérimentalement à un gain significatif de l'efficacité de codage par rapport à ces dernières.

Des travaux ultérieurs dus à Nguyen, de Queiroz et Rao [132] ont permis la généralisation de la LOT et de la LBT en montrant que ces transformées sont des cas spéciaux de la GenLOT [45] (*GENeralized LOT*). Cette dernière étend la largeur de chevauchement des transformées à recouvrement en mettant en jeu des filtres d'analyse de taille KM .

MLT et MBLT

En parallèle des transformées LOT et LBT basées sur la DCT, Malvar a introduit une autre transformée à recouvrement : la transformée modulée à recouvrement [82], aussi nommée MLT (*Modulated Lapped Transform*). C'est une transformée orthogonale, basée sur la modulation d'un filtre prototype cosinusoidal à réponse finie, dont les filtres d'analyse $\{h_k\}_{0 \leq k < M}$ de support $L = 2M$ sont définis par :

$$\forall 0 \leq n < 2M, \quad h_k(n) = w(n) \sqrt{\frac{2}{M}} \cos \left[\left(n + \frac{M+1}{2} \right) \left(k + \frac{1}{2} \right) \frac{\pi}{M} \right]$$

$$\text{avec } w(n) = -\sin \left[\left(n + \frac{1}{2} \right) \frac{\pi}{2M} \right]$$

La MLT n'est pas une transformée à phase linéaire. Elle possède cependant une bonne sélectivité fréquentielle où les lobes secondaires sont peu importants, comme illustré sur la Fig. 5.2, où sont tracées les réponses fréquentielles des filtres $\{h_k\}_{0 \leq k < 4}$ intervenant dans la MLT à $M = 4$ bandes.

La relaxation de la contrainte portant sur la fenêtre w permet d'introduire une version biorthogonale de la MLT, la MLBT [83] (*Modulated Lapped Biorthogonal Transform*) ou transformée modulée biorthogonale à recouvrement. En partant du même filtre prototype de la MLT, mais modulé par des fenêtres d'analyse w_a et de synthèse w_s différentes, on définit la MLBT par :

$$\forall 0 \leq n < 2M, \quad h_k(n) = w_a(n) \sqrt{\frac{2}{M}} \cos \left[\left(n + \frac{M+1}{2} \right) \left(k + \frac{1}{2} \right) \frac{\pi}{M} \right]$$

$$\forall 0 \leq n < 2M, \quad g_k(n) = w_s(n) \sqrt{\frac{2}{M}} \cos \left[\left(n + \frac{M+1}{2} \right) \left(k + \frac{1}{2} \right) \frac{\pi}{M} \right]$$

où les fenêtres symétriques d'analyse $w_a(n) = w_a(2M - n - 1)$ et de synthèse $w_s(n) = w_s(2M - n - 1)$ sont définies par :

$$\forall 0 \leq n < M, \quad w_s(n) = \frac{1 - \cos \left[\left(\frac{n+1}{2M} \right)^\alpha \pi \right] + \beta}{2 + \beta}$$

$$\forall 0 \leq n < M, \quad w_a(n) = \frac{w_s(n)}{w_s^2(n)^2 + w_s^2(n+M)}$$

Les paramètres α et β permettent de modifier les fenêtres w_a et w_s de la MLBT et d’agir sur ses propriétés. La MLBT possède une meilleure sélectivité fréquentielle que la MLT. Ceci s’observe sur la Fig. 5.2 où sont tracées les réponses fréquentielles des filtres 4-bandes d’analyse de la MLT et de la MLBT, paramétrées avec $\alpha = 0.85$ et $\beta = 0.0$

La MLT et la MLBT sont des transformées principalement utilisées en codage audio car leur non-linéarité de phase est gênante en codage d’image. Cependant, leur grande sélectivité fréquentielle en fait des transformées candidates intéressantes pour le codage des sous-bandes issues de la décomposition temporelle d’une séquence vidéo.

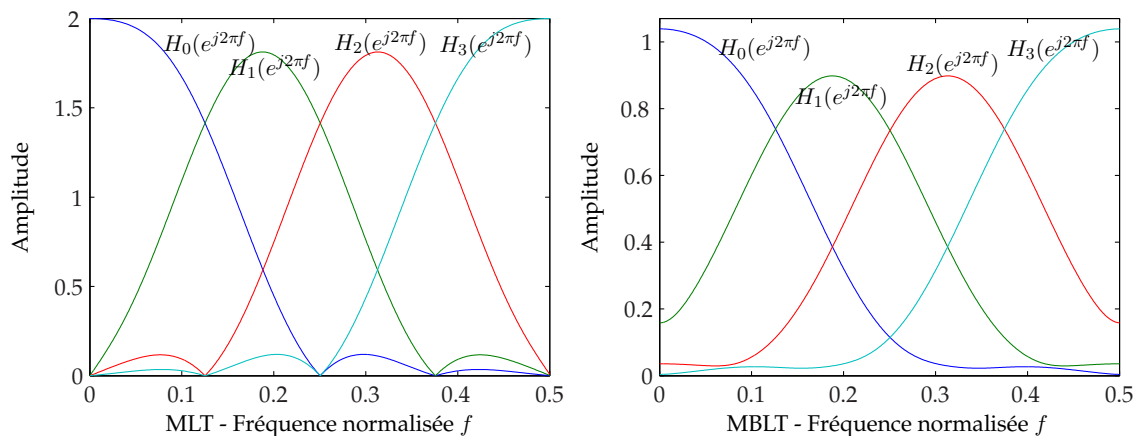


FIG. 5.2 – Réponses fréquentielles des filtres d’analyse $\{h_k\}_{0 \leq k < 4}$ de la MLT 4-bandes (gauche) et de la MLBT 4-bandes (droite).

5.2 Codage spatial par bancs de filtres M-bandes

Dans le cadre de notre schéma de codage vidéo, nous nous intéressons dans un premier temps en section 5.2.1 à l’étude des différents types de sous-bandes issues du filtrage temporel. L’utilisation d’un cadre générique de construction de transformées M -bandes nous permet alors dans la section 5.2.2 de spécifier une transformée 4-bandes adaptée à la décorrélation des sous-bandes temporelles de détail. Enfin, des simulations expérimentales nous permettent de montrer le gain objectif en efficacité de codage apporté par cette transformée. Les résultats de ces travaux ont conduit à la publication d’un article de conférence [101], contribuant à la discussion sur un thème peu abordé dans la littérature.

Nous présentons alors en section 5.2.3 des résultats plus complets sur l’utilisation des transformées M -bandes comme alternative au filtrage 9/7 pour la décomposition des sous-bandes temporelles de détail. La transformée 4-bandes FB1 construite précédemment et les transformées 8-bandes LOT, LBT, MLT et MBLT sont alors mises en œuvre au sein du codec Vidwav afin de conduire les simulations. Les conclusions de cette étude publiée dans [100] sont cependant mitigées et nous tenterons d’en expliquer les raisons.

5.2.1 Caractéristiques des sous-bandes temporelles

Dans le cadre de notre schéma de codage vidéo, les sous-bandes issues de la décomposition temporelle d’une séquence vidéo sont des images présentant un aspect visuel très différent en fonction de leur type. À titre d’exemple, nous avons illustré en Fig. 5.3 les

sous-bandes issues de la décomposition temporelle d'un extrait de la séquence *Foreman*. Commentons tout d'abord l'aspect visuel de ces images.

Les sous-bandes d'approximation, provenant du filtrage temporel passe-bas, ont un aspect très similaire à une image naturelle. Comme vu dans la section 4.2.1 du chapitre précédent, elles peuvent toutefois comporter quelques artefacts fantômes et des discontinuités spatiales, provenant de la pseudo-inversion des champs de mouvement effectuée durant l'étape de mise à jour temporelle.

Au contraire, les sous-bandes de détail, provenant du filtrage temporel passe-haut, ne présentent pas du tout un comportement d'image naturelle : ces images sont de moyenne nulle, possèdent de nombreux "contours" et de larges zones de texture à hautes fréquences, baignant dans un bruit assez perceptible. Ces trames ressemblent beaucoup à des images de gradient ou à une image de résidu de prédiction temporelle de type DFD, intervenant dans les schémas de codage vidéo hybride décrits en section 2.1.



FIG. 5.3 – Sous-bandes issues de la décomposition temporelle 5/3 sur 4 niveaux de la séquence *Foreman* CIF. De gauche à droite et de haut en bas, on observe successivement une sous-bande de détail du premier, deuxième et troisième niveau, suivie d'une sous-bande d'approximation du quatrième niveau temporel.

L'étude des caractéristiques spatiales des sous-bandes temporelles peut être précisée dans le domaine fréquentiel. À cette fin, nous avons présenté dans les Tabs. 5.1 et 5.2 la répartition de la densité spectrale de puissance des sous-bandes d'approximation et de

détail, issues de la décomposition temporelle 5/3 des séquences *Mobile* et *Foreman*.

f_h/f_v	$[0, \frac{1}{8})$	$[\frac{1}{8}, \frac{1}{4})$	$[\frac{1}{4}, \frac{3}{8})$	$[\frac{3}{8}, \frac{1}{2}]$	f_h/f_v	$[0, \frac{1}{8})$	$[\frac{1}{8}, \frac{1}{4})$	$[\frac{1}{4}, \frac{3}{8})$	$[\frac{3}{8}, \frac{1}{2}]$
$[0, \frac{1}{8})$	97.99%	0.40%	0.19%	0.05%	$[0, \frac{1}{8})$	99.58%	0.07%	0.02%	0.01%
$[\frac{1}{8}, \frac{1}{4})$	0.66%	0.23%	0.12%	0.02%	$[\frac{1}{8}, \frac{1}{4})$	0.18%	0.01%	0.0%	0.0%
$[\frac{1}{4}, \frac{3}{8})$	0.18%	0.07%	0.04%	0.01%	$[\frac{1}{4}, \frac{3}{8})$	0.07%	0.01%	0.0%	0.0%
$[\frac{3}{8}, \frac{1}{2}]$	0.03%	0.01%	0.0%	0.0%	$[\frac{3}{8}, \frac{1}{2}]$	0.04%	0.01%	0.0%	0.0%

TAB. 5.1 – Répartition moyenne de la densité spectrale d'énergie observée sur les sous-bandes d'approximation issues du 4^{ème} niveau de la décomposition temporelle des séquences *Mobile* (gauche) et *Foreman* (droite), selon les bandes fréquentielles normalisées horizontale f_h et verticale f_v .

f_h/f_v	$[0, \frac{1}{8})$	$[\frac{1}{8}, \frac{1}{4})$	$[\frac{1}{4}, \frac{3}{8})$	$[\frac{3}{8}, \frac{1}{2}]$	f_h/f_v	$[0, \frac{1}{8})$	$[\frac{1}{8}, \frac{1}{4})$	$[\frac{1}{4}, \frac{3}{8})$	$[\frac{3}{8}, \frac{1}{2}]$
$[0, \frac{1}{8})$	18.80%	10%	7.13%	3.27%	$[0, \frac{1}{8})$	39.55%	10.29%	4.27%	1.96%
$[\frac{1}{8}, \frac{1}{4})$	14.91%	9.20%	6.65%	2.50%	$[\frac{1}{8}, \frac{1}{4})$	13.89%	4.06%	1.74%	0.52%
$[\frac{1}{4}, \frac{3}{8})$	9.71%	6.09%	4.12%	1.21%	$[\frac{1}{4}, \frac{3}{8})$	7.88%	2.73%	1.18%	0.3%
$[\frac{3}{8}, \frac{1}{2}]$	3.30%	1.75%	1.05%	0.29%	$[\frac{3}{8}, \frac{1}{2}]$	7.33%	2.88%	1.17%	0.25%

TAB. 5.2 – Répartition moyenne de la densité spectrale d'énergie observée sur les sous-bandes de détail issues du 4^{ème} niveau de la décomposition temporelle des séquences *Mobile* (gauche) et *Foreman* (droite), selon les bandes fréquentielles normalisées horizontale f_h et verticale f_v .

On aperçoit clairement la forte dissimilarité des caractéristiques spectrales des deux types de sous-bandes temporelles. Dans les images d'approximation, plus de 98 % de l'énergie de l'image est contenu dans la bande 2D de fréquence normalisée passe-bas inférieure à 1/8. Il n'y a donc quasiment pas de moyennes et hautes fréquences dans les sous-bandes d'approximation. Le cas des sous-bandes temporelles de détail est très différent. La portion des basses fréquences inférieures à 1/8 représente seulement 18.80 % de l'énergie totale sur *Mobile* et 40 % dans le cas de *Foreman*. Le spectre 2D est ainsi nettement plus riche en moyennes et hautes fréquences, horizontales comme verticales. Cependant, il est loin d'être uniformément réparti et les hautes fréquences diagonales sont par exemple rares. Nous ne sommes donc pas en présence d'un bruit blanc et pouvons ainsi espérer trouver une transformée adaptée à la décorrélation de ce type d'image.

Bien que ces deux types de sous-bandes temporelles ne partagent pas du tout les mêmes caractéristiques visuelles et fréquentielles, la majorité des schémas de codage vidéo $t + 2D$ [16, 26, 127, 164] utilisent la transformation en ondelettes biorthogonale 9/7 pour décomposer indifféremment les sous-bandes d'approximation et de détail. Ce choix se relève certainement justifié dans le cas des images d'approximation, tant elles ressemblent à des images naturelles et du fait que la transformation 9/7 soit adaptée [18, 159] à la décomposition de ce type d'images. Cependant, son utilisation pour le filtrage spatial des sous-bandes temporelles de détail n'est motivée par aucun argument.

Ces remarques motivent ainsi notre approche décrite dans la section suivante, consistant en la construction d'une transformée 4-bandes dont les caractéristiques de régularité et de sélectivité fréquentielle sont adaptées à la décomposition des sous-bandes de détail.

5.2.2 Construction d'un banc de filtres 4-bandes adapté

Nous souhaitons construire un banc de filtres M -bandes adapté à la décomposition spatiale des sous-bandes temporelles de détail. Cependant et avant d'imposer des conditions sur le banc de filtres, comment construit-on une transformée M -bandes ayant des propriétés précises ? La tâche n'est pas aisée car les équations de reconstruction parfaite (5.3) sont difficiles à satisfaire. Afin d'y parvenir simplement, nous nous proposons d'utiliser un algorithme de construction de transformées M -bandes offrant suffisamment de degrés de liberté pour nous permettre par la suite d'imposer des conditions supplémentaires de régularité.

Algorithme d'Alkin et Caglar

Alkin et Caglar [12] ont proposé un algorithme général de construction de bancs de filtres M -bandes orthogonaux, à phase linéaire et à reconstruction parfaite. Cet algorithme offre de plus suffisamment de degrés de liberté pour pouvoir ajouter des conditions supplémentaires sur le banc de filtres, de façon à l'adapter à la décorrélation d'un signal spécifique. Le cadre de construction impose que M soit une puissance de 2 et que les filtres $\{h_k\}$ soient tous d'une même taille L , multiple de $2M$; on notera ainsi $L = 2KM$.

Le principe est le suivant : étant donné un filtre prototype passe-bas h_0 symétrique par rapport à $-1/2$ et dont les M -translatées forment une famille orthogonale, comme spécifié par la relation (5.7), on obtient les coefficients des $M - 1$ autres filtres par permutation des coefficients du filtre h_0 .

$$\forall k \in \mathbb{Z} \quad \sum_n h_0(n)h_0(n - Mk) = \delta_k \quad (5.7)$$

Le filtre prototype passe-bas h_0 caractérise donc entièrement le banc de filtres. Les autres filtres $\{h_k\}_{1 \leq k < M}$ sont obtenus à partir de h_0 et de la relation $h_k = B_k h_0$ où B_k est une matrice de permutation calculée grâce aux relations suivantes et au moyen du produit tensoriel \otimes :

$$B_k = \prod_{i=1}^{\log_2 M} A_i^{r_{ki}}$$

$$A_i = P_{N/2^i} \otimes J_{2^i}$$

$$P_K = \{p_{jk}\}_{K \times K} \text{ avec } p_{jk} = (-1)^j \delta_{j-k} \quad (\text{matrice identité alternée})$$

$$J_K = \{a_{jk}\}_{K \times K} \text{ avec } a_{jk} = \delta_{K+1-j-k} \quad (\text{matrice contre-identité})$$

où r_{ki} prend ses valeurs dans $\{0, 1\}$ et représente le i -ième bit de poids faible de la représentation binaire de l'entier k .

L'algorithme d'Alkin et Caglar permet ainsi de construire une transformée M -bandes à reconstruction parfaite à partir du seul filtre h_0 . Celui-ci doit cependant être symétrique et vérifier les conditions d'orthogonalité (5.7). Ainsi, pour une transformée M -bandes dont les filtres ont une taille $L = 2KM$, on dispose de $d = L/2 - L/M = K(M - 2)$ degrés de liberté pour choisir le filtre prototype passe-bas h_0 .

Construction du filtre prototype h_0 dans le cas 4-bandes

Considérons dans un premier temps le cas $M = 4$ bandes car c'est le cas le plus simple non-dyadique autorisé par le schéma de construction d'Alkin et Caglar. De même, nous

choisissons une longueur de filtre $L = 8$ échantillons, correspondant au cas le plus simple $K = 1$. Cette longueur est proche de celle de l'ondelette 9/7 d'analyse et paraît raisonnable pour assurer une décorrélation efficace des sous-bandes temporelles de détail.

Dans le cas $M = 4$ bandes, les coefficients des filtres h_1 , h_2 et h_3 sont alors obtenus à partir du filtre prototype passe-bas symétrique h_0 et des permutations suivantes :

$$\forall -4 \leq n \leq 3 \quad \begin{cases} h_1(n) &= (-1)^{\lfloor n/2 \rfloor} h_0(n + (-1)^n) \\ h_2(n) &= (-1)^n h_1(n) \\ h_3(n) &= (-1)^n h_0(n) \end{cases} \quad (5.8)$$

Conformément à la section précédente, nous disposons donc de $d = 2$ degrés de liberté pour choisir le filtre h_0 . Nous décidons de les utiliser pour construire un banc de filtres régulier avec suffisamment de moments nuls, lui permettant ainsi d'assurer une bonne approximation polynomiale. À cette fin, nous imposons des conditions pour maximiser le nombre de moments nuls des ondelettes ψ_1 , ψ_2 et ψ_3 sous-jacentes au banc de filtres et respectivement associées aux filtres h_1 , h_2 et h_3 .

Tout d'abord, le filtre prototype h_0 de taille $L = 8$ doit être symétrique et satisfaire les conditions d'orthogonalité exprimées par les relations (5.7). Ces hypothèses peuvent être traduites par les équations suivantes :

$$\begin{cases} h_0(0)^2 + h_0(1)^2 + h_0(2)^2 + h_0(3)^2 &= 1/2 \\ h_0(0)h_0(3) + h_0(1)h_0(2) &= 0 \end{cases} \quad (5.9)$$

Imposons désormais les moments nuls, grâce à une propriété liant une ondelette à son filtre miroir conjugué associé. Une ondelette ψ possède ainsi p moments nuls si et seulement si la transformée en Z de son filtre miroir $H(z)$ possède un zéro de multiplicité p en $z = 1$. L'antisymétrie des filtres h_1 et h_3 par rapport à $-1/2$ implique que $H_1(1) = H_3(1) = 0$ et confère donc à leur ondelettes respectives ψ_1 et ψ_3 au moins un moment nul.

Nous décidons d'utiliser un degré de liberté pour imposer un moment nul sur ψ_2 . Cette condition revient donc à poser une contrainte d'égalité sur h_2 afin de vérifier $H_2(1) = 0$. Grâce aux relations de permutations (5.8), cette condition s'exprime alors sous forme de contrainte sur h_0 par :

$$h_0(0) + h_0(3) = h_0(1) + h_0(2) \quad (5.10)$$

Afin de minimiser les coefficients d'ondelettes situés dans la dernière bande et de permettre ainsi au codeur à arbre de zéros sous-jacent de mieux traiter les coefficients des bandes précédentes, nous imposons à ψ_3 de posséder au moins deux moments nuls. Ceci équivaut à exiger que la dérivée de la transformée en Z de h_3 vérifie $H'_3(1) = 0$, conduisant à $\sum_k k h_3(k) = 0$. De plus, due à l'antisymétrie de h_3 , cette condition se simplifie par $\sum_{k \geq 0} (2k + 1) h_3(k) = 0$, imposant à son tour une nouvelle contrainte sur h_0 :

$$-7h_0(3) + 5h_0(2) - 3h_0(1) + h_0(0) = 0 \quad (5.11)$$

Il est utile de rappeler que dû à la symétrie de h_2 par rapport à $-1/2$, on peut montrer que $H_2(1) = 0$ implique $H'_2(1) = 0$ et qu'ainsi ψ_2 possède deux moments nuls. De même et à cause de l'antisymétrie de h_3 , $H'_3(1) = 0$ implique que $H''_3(1) = 0$, donnant alors trois moments nuls à ψ_3 . Au final, les ondelettes ψ_1 , ψ_2 et ψ_3 possèdent ainsi respectivement 1, 2 et 3 moments nuls.

La réunion des conditions (5.9), (5.10) et (5.11) amène alors au système d'équations suivant, qui lie les coefficients de h_0 :

$$\begin{cases} h_0(0)^2 + h_0(1)^2 + h_0(2)^2 + h_0(3)^2 = 1/2 \\ h_0(0)h_0(3) + h_0(1)h_0(2) = 0 \\ h_0(0) + h_0(3) - h_0(1) - h_0(2) = 0 \\ -7h_0(3) + 5h_0(2) - 3h_0(1) + h_0(0) = 0 \end{cases}$$

Ce système non-linéaire est résoluble et possède une unique solution :

$$\begin{cases} h_0(0) = (5 + \sqrt{15})/16 \\ h_0(1) = (3 + \sqrt{15})/16 \\ h_0(2) = (5 - \sqrt{15})/16 \\ h_0(3) = (3 - \sqrt{15})/16. \end{cases} \quad (5.12)$$

Le filtre prototype symétrique h_0 dont les coefficients sont donnés par la relation (5.12) permet ainsi de construire un banc de filtres 4-bandes au moyen des équations de permutation (5.8). Par construction, ce banc à reconstruction parfaite est orthogonal, à phase linéaire et ses ondelettes ψ_1 , ψ_2 et ψ_3 sous-jacentes possèdent respectivement 1, 2 et 3 moments nuls. Il sera nommé FB1 dans la suite du document. On remarquera que dans le cas dyadique, il n'est pas possible de construire un banc de filtres simultanément orthogonal et à phase linéaire (mis à part le cas trivial de Haar) : ceci justifie ainsi la flexibilité accrue des transformées M -bandes.

Il est bien sûr possible de fixer d'autres conditions sur le banc de filtres. À titre expérimental, nous pouvons ainsi imposer à ψ_1 , ψ_2 et ψ_3 de posséder respectivement 3, 2 et 1 moments nuls. En suivant le même raisonnement que précédemment, on aboutit à un banc de filtres 4-bandes que nous nommerons FB2 et dont les coefficients du filtre prototype passe-bas h_0 sont donnés par la relation (5.13) suivante :

$$\begin{cases} h_0(0) = (3 + \sqrt{3})/8 \\ h_0(1) = (3 - \sqrt{3})/8 \\ h_0(2) = (1 + \sqrt{3})/8 \\ h_0(3) = (1 - \sqrt{3})/8. \end{cases} \quad (5.13)$$

On remarquera que les exemples de bancs de filtres donnés par Alkin [12] pour illustrer son algorithme n'ont pas été obtenus en imposant ces conditions formelles. Pour calculer les coefficients du filtre prototype h_0 , l'auteur utilise au contraire une stratégie d'optimisation numérique ayant pour but de maximiser le gain de codage du banc de filtres. Il obtient alors des résultats numériques dont les valeurs approchées ne satisfont l'équation (5.10) qu'à 10^{-3} près. Il en résulte une transformée qui, en présence d'un signal constant, donne des coefficients non strictement égaux à 0, mais plus proches de 10^{-3} . Cela est probablement dû au fait que le calcul des coefficients par la stratégie d'optimisation numérique ait été mené avec une précision machine de 10^{-7} . Dans le contexte d'un codeur emboîté à arbre de zéros, ces coefficients non strictement nuls sont susceptibles de dégrader l'efficacité du codage.

Nous présentons dans le Tab. 5.3 les valeurs numériques des filtres prototypes h_0 obtenues pour les bancs de filtres FB1 et FB2 et les comparons avec celles calculées par Alkin pour le banc de filtres 4-bandes à 8 échantillons. On observe la forte similitude des coefficients de FB1 avec ceux d'Alkin. De plus, afin d'étudier quelques propriétés des bancs FB1 et FB2, nous avons représenté sur la Fig. 5.4 les fonctions de transfert de leurs filtres.

Les fonctions de transfert du banc d'Alkin, très proches de celles de FB1, n'ont pas été représentées.

n	$h_0(n)$ FB1	$h_0(n)$ FB2	$h_0(n)$ Alkin
0	0.554561459	0.591506350	0.567030813
1	0.429561459	0.158493649	0.406151488
2	0.070438540	0.341506350	0.094517754
3	-0.054561459	-0.091506350	-0.067700953

TAB. 5.3 – Comparaison des valeurs numériques des coefficients de h_0 .

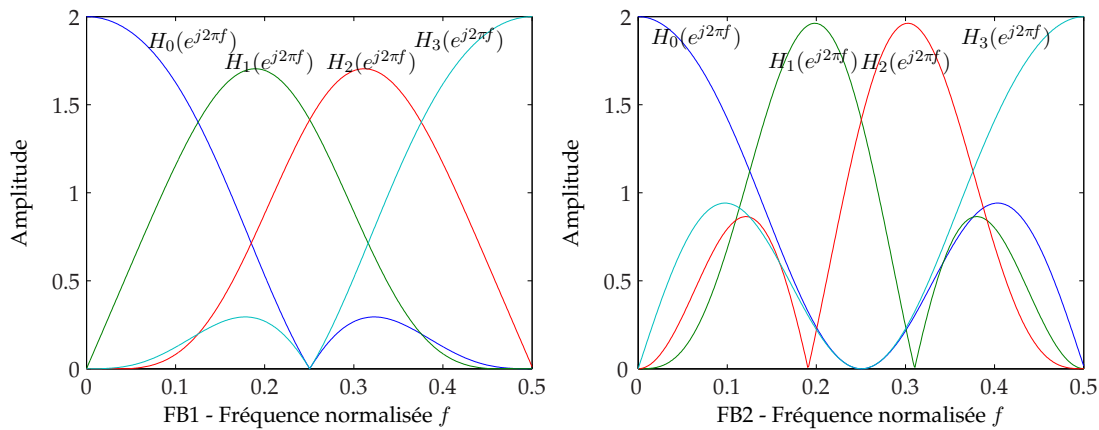


FIG. 5.4 – Réponses fréquentielles des filtres d'analyse $\{h_k\}_{0 \leq k < 4}$ des bancs de filtres 4-bandes FB1 (gauche) et FB2 (droite).

On observe la bonne sélectivité fréquentielle offerte par le banc de filtres FB1. Au contraire, les réponses fréquentielles des filtres du banc FB2 semblent posséder des lobes secondaires importants, pouvant nuire à l'efficacité de la décorrélation. Enfin, pour illustrer le lien existant entre le banc de filtres 4-bandes FB1 et ses ondelettes sous-jacentes, nous avons tracé sur la Fig. 5.5 la fonction d'échelle ϕ et les ondelettes ψ_1 , ψ_2 et ψ_3 associées respectivement aux filtres h_0 , h_1 , h_2 et h_3 du banc FB1.

Résultats expérimentaux

Afin d'évaluer le gain objectif en efficacité de codage apporté par les transformées FB1 et FB2 utilisées pour la décomposition spatiale des sous-bandes temporelles de détail, nous avons réalisé des simulations de codage vidéo en utilisant notre prototype basé sur le codec MC-EZBC. Lors de nos expérimentations, nous avons considéré les séquences CIF *Mobile* et *City* à 30 Hz, choisies pour la grande diversité de mouvement et de textures qu'elles offrent. Ces dernières ont alors été décomposées sur 5 niveaux temporels par la transformée 5/3 uniforme, décrite dans la section 4.2. Les sous-bandes temporelles d'approximation résultantes ont été décomposées spatialement sur 5 niveaux avec la transformée 9/7. Enfin, les sous-bandes temporelles de détail et d'approximation ont été encodées avec l'algorithme EZBC qui a du être adapté pour fonctionner avec des pyramides spatiales 4-bandes. La modification a simplement consisté en la réorganisation de la pyramide afin qu'elle ressemble à une pyramide spatiale dyadique classique.

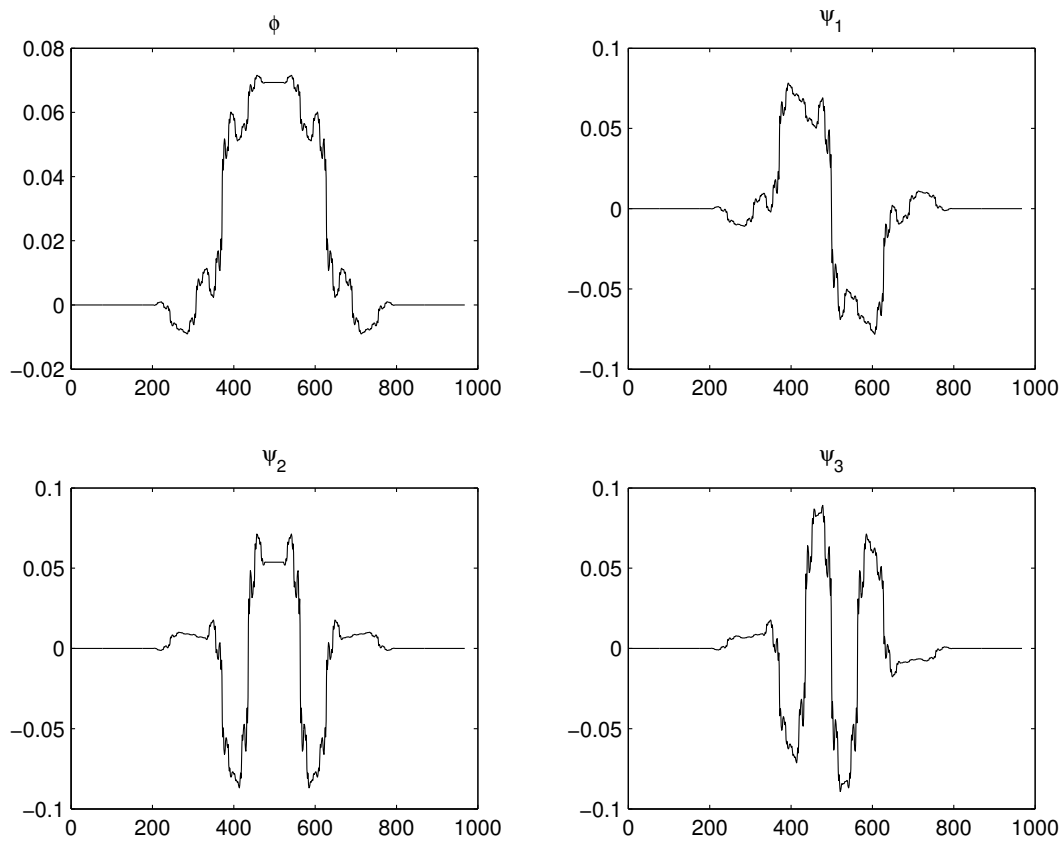


FIG. 5.5 – Fonction d'échelle et ondelettes associées au banc de filtres 4-bandes FB1.

Nous présentons dans les Tabs. 5.4 et 5.5 l'efficacité de codage obtenue en utilisant les transformées 4-bandes FB1 et FB2 pour décomposer les sous-bandes temporelles de détail sur un niveau spatial. Afin de comparer ces résultats, nous avons de plus effectué des simulations de codage en utilisant les transformées dyadiques 9/7 et 5/3 pour décomposer ces mêmes sous-bandes temporelles sur deux niveaux spatiaux. L'ensemble des résultats est exprimé sous forme de Y-PSNR moyen calculé sur l'ensemble des images décodées.

YSNR (en dB)	384 kbs	512 kbs	768 kbs	1024 kbs
4-band FB1	27.93	29.95	32.16	33.65
4-band FB2	27.80	29.70	31.70	33.05
9/7	27.71	29.76	32.00	33.47
5/3	27.40	29.52	31.60	33.05

TAB. 5.4 – Comparaison débit-distorsion de plusieurs transformées spatiales sur la séquence *Mobile* CIF à 30 Hz.

Nous observons que l'utilisation du banc de filtres FB1 pour décomposer les sous-bandes temporelles de détail apporte un gain d'environ 0.2 dB à tous les débits, comparé à la transformée 9/7 et un gain d'environ 0.5 dB lors de la comparaison avec la transformée 5/3. Nous remarquons aussi les performances médiocres obtenues avec le banc

YSNR (en dB)	384 kbs	512 kbs	768 kbs	1024 kbs
4-band FB1	33.61	35.45	37.65	39.26
4-band FB2	33.44	35.10	37.08	38.59
9/7	33.40	35.23	37.47	39.09
5/3	33.16	34.92	37.06	38.73

TAB. 5.5 – Comparaison débit-distorsion de plusieurs transformées spatiales sur la séquence *City* CIF à 30 Hz.

de filtres FB2, qui montre une efficacité inférieure à celle de la transformée 9/7. Cette contre-performance n'est pas étonnante et est certainement due à la mauvaise sélectivité fréquentielle du banc de filtres FB2.

Les gains en efficacité de codage observés lors de l'utilisation du banc de filtres FB1 pour filtrer spatialement les sous-bandes de détail, bien que modérés, sont cependant encourageants. Ils justifient en effet la démarche suivie, à savoir la construction d'une transformée M -bandes en imposant des conditions de régularité sur ses filtres sous-jacents. Il serait alors intéressant de continuer ces travaux en utilisant des bancs de filtres plus longs et sur un nombre supérieur de bandes. Cependant, l'architecture du codec spatial EZBC ne s'y prête pas car ce dernier ne gère que les pyramides spatiales dyadiques. Nous nous proposons dans la section suivante d'utiliser le codec Vidwav pour continuer notre étude sur les transformées M -bandes utilisées pour la décomposition spatiale des sous-bandes temporelles de détail.

5.2.3 Étude de différents bancs de filtres

Le codec vidéo MPEG Vidwav, issu des travaux de Song, Wu, Xiong et Xu [133, 164, 165] sur l'algorithme 3D-ESCOT, est un schéma de codage vidéo générique et efficace décrit dans la section 2.2.5. Il offre des performances équivalentes au schéma de codage MC-EZBC en résolution nominale mais donne de meilleurs résultats en scalabilité spatiale. De plus, il est nettement plus flexible que le codec MC-EZBC et gère correctement les décompositions spatiales en paquets d'ondelettes. Dans le contexte du processus de normalisation MPEG sur le codage vidéo scalable SVC, le codec Vidwav a été mis tardivement à la disposition des membres MPEG en Mai 2005. De part son efficacité de codage et sa souplesse d'utilisation, il nous a alors semblé intéressant d'utiliser ce codec afin de poursuivre nos investigations sur les décompositions M -bandes des sous-bandes temporelles de détail.

On remarquera les travaux similaires sur les paquets d'ondelettes proposés par Cheng [162] pour décomposer les résidus de prédiction temporelle DFD dans le cadre d'un codec vidéo hybride MPEG-2. Bien qu'intéressants, les résultats ne sont cependant pas entièrement satisfaisants et nécessitent l'utilisation d'algorithmes complexes de recherche de la meilleure base. Dans la continuation de ces travaux, Trocan [149] a présenté dans le contexte du codec Vidwav, des résultats nettement plus encourageants sur l'utilisation des décompositions en paquets d'ondelettes des sous-bandes temporelles de détail.

Nous avons ainsi conduit plusieurs simulations de codage vidéo en utilisant le codec Vidwav. Les séquences ont été encodées dans le mode $t + 2D$, sans utilisation du module *Inband* $2D + t + 2D$ ni du module *Base layer*. L'estimation de mouvement est basée sur des blocs de tailles variant de 4×4 à 16×16 pixels en utilisant des champs estimés au $1/4$ de pixel près. La décomposition temporelle des séquences a été réalisée au moyen

du filtre 5/3 sur 4 niveaux où chaque bloc est filtré selon son mode de prédiction le plus favorable, tel que décrit dans la section 2.2.5. Les sous-bandes temporelles sont codées au moyen de l'algorithme 3D-ESCOT.

L'algorithme 3D-ESCOT est capable de gérer les décompositions spatiales dyadiques en paquets d'ondelettes. Or, il est clair que la structure de décomposition d'une transformée M -bandes peut être exprimée sous forme d'une base dyadique en paquets d'ondelettes si M est une puissance de deux. Il suffit ainsi de décomposer uniformément toutes les sous-bandes d'approximation et de détail de manière à obtenir une base de décomposition où toutes les sous-bandes sont de même taille. Le codec Vidwav peut donc coder les coefficients issus d'une transformée M -bandes si M est une puissance de 2.

Dans nos expérimentations, les sous-bandes temporelles d'approximation sont décomposées spatialement sur 5 niveaux par la transformée dyadique 9/7. Afin d'évaluer l'efficacité des transformées M -bandes, les sous-bandes de détail sont décomposées avec la transformée dyadique 9/7 sur deux niveaux spatiaux et sur un seul niveau avec les transformées 4-bandes MLT, MBLT et FB1 et les transformées 8-bandes LOT et LBT. L'expérimentation supplémentaire MLT2 a été conduite avec la transformée MLT sur deux niveaux spatiaux. L'implémentation de la LBT et de la LOT est due à Malvar [81, 84].

Nous comparons dans les Tabs. 5.6, 5.7 et 5.8 l'efficacité de codage observée à plusieurs débits avec les différentes transformées spatiales proposées pour la décomposition des sous-bandes temporelles de détail. L'efficacité de codage est exprimée en terme de PNSR moyen calculé sur la composante de luminance Y de l'ensemble des images décodées. De plus, nous avons inséré à des fins de comparaison les résultats de codage obtenus avec le codec MPEG/ITU SVC, en cours de normalisation dans sa version JSVM-1.

YSNR (en dB)	192 kbs	256 kbs	384 kbs
2-band 9/7	26.04	27.47	29.20
4-band FB1	26.06	27.46	29.17
4-band MLT	26.12	27.51	29.20
4-band MLT2	25.77	27.15	28.75
4-band MLBT	25.92	27.17	28.66
8-band LOT	26.02	27.36	29.03
8-band LBT	25.83	27.18	28.82
JSVM-1	26.11	27.25	29.16

TAB. 5.6 – Comparaison débit-distorsion de différentes transformées spatiales utilisées pour la décomposition des sous-bandes de détail de la séquence *Mobile* CIF à 30 Hz.

Nous observons tout d'abord que la MLT et la transformée dyadique 9/7 donnent les meilleurs résultats globaux dans ces expérimentations et pour tous les débits. La transformée MLT est la plus efficace sur les séquences *Mobile* et *Bus* tandis que la performance du filtre 9/7 s'illustre sur la séquence *Foreman*. Ceci peut être expliqué par les mouvements complexes rotatoires observés sur cette séquence ; ceux-ci gênent la prédiction temporelle assurée par le filtre 5/3 et induisent la création de zones mal-prédites au sein des sous-bandes temporelles de détail. Ces zones possèdent des caractéristiques proches d'une image naturelle, comme illustré par la Fig. 5.3 et justifié dans le Tab. 5.2 par la présence de basses fréquences. Dû à son nombre de moments nuls élevés, la transformée 9/7 possède alors de meilleures propriétés d'approximation sur ces zones, expliquant ainsi le

YSNR (en dB)	128 kbs	160 kbs	192 kbs	256 kbs
2-band 9/7	31.47	32.46	33.22	34.33
4-band FB1	31.23	32.22	32.97	34.07
4-band MLT	31.24	32.23	32.99	34.12
4-band MLT2	31.06	31.99	32.73	33.79
4-band MBLT	30.97	31.89	32.68	33.85
8-band LOT	31.43	32.38	33.13	34.22
8-band LBT	31.31	32.03	33.03	34.14
JSVM-1	30.92	31.96	32.70	34.20

TAB. 5.7 – Comparaison débit-distorsion de différentes transformées spatiales utilisées pour la décomposition des sous-bandes de détail de la séquence *Foreman* CIF à 30 Hz.

YSNR (en dB)	256 kbs	320 kbs	384 kbs	512 kbs
2-band 9/7	27.41	28.45	29.21	30.50
4-band FB1	27.41	28.44	29.22	30.53
4-band MLT	27.43	28.47	29.26	30.57
4-band MLT2	27.20	28.15	28.81	30.10
4-band MLBT	27.14	28.12	28.76	30.02
8-band LOT	27.40	28.42	29.18	30.45
8-band LBT	27.24	28.26	29.06	30.28
JSVM-1	27.11	27.84	28.90	30.09

TAB. 5.8 – Comparaison débit-distorsion de différentes transformées spatiales utilisées pour la décomposition des sous-bandes de détail de la séquence *Bus* CIF à 30 Hz.

gain observé par rapport à la transformée MLT.

On notera aussi les bonnes performances globales obtenues par la transformée à recouvrement LOT. Sachant que les algorithmes rapides implémentant cette transformée sont basés sur la DCT et que ces derniers se retrouvent aisément dans des équipement matériels destinés au traitement vidéo, la LOT constitue une bonne alternative entre la performance qu'elle offre et la complexité qu'elle nécessite. Dans le cadre de son codec PTC [84], Malvar a de plus montré que le calcul de la LOT possède une complexité inférieure à celle de la transformée 9/7 d'environ 15 %.

Durant ces expérimentations, la transformée FB1 donne cependant des résultats inférieurs à ceux observés avec la MLT ou la transformée 9/7. Ces observations ne confirment donc pas les conclusions tirées dans la section précédente où, dans le cadre du codec vidéo MC-EZBC, la transformée FB1 surpassait la transformée 9/7 d'environ 0.3 dB. Ceci peut s'expliquer par la prédiction temporelle avec sélection de blocs opérée par le codec Vidwaw. Ce type de prédiction diminue en effet le coût des blocs mais au prix d'une augmentation du nombre d'artefacts dans les sous-bandes temporelles de détail, qui ne sont pas traités efficacement par la transformée FB1.

Les transformées biorthogonales MLBT et LBT ne se sont pas distinguées par de bons résultats dans ces simulations. En dépit de sa meilleure sélectivité fréquentielle, la MLBT possède en effet un gain de codage inférieur à la MLT. Il semble en fait que la biortho-

gonalité de ces transformées nuit à leur efficacité de décorrélation, laissant penser que l'algorithme d'allocation débit-distorsion intégré dans le codec 3D-ESCOT nécessite la présence d'une transformée orthogonale.

Enfin, on observe les mauvais résultats obtenus avec l'expérimentation MLT2, basée sur la décomposition des sous-bandes temporelles de détail par la transformée MLT sur deux niveaux spatiaux. Comparé à la transformée MLT sur un seul niveau spatial, on observe ainsi que l'ajout de ce niveau fait chuter l'efficacité de codage. La raison pourrait consister en la présence des 31 sous-bandes spatiales mises en jeu dans l'expérimentation MLT2, ajoutant une surcharge importante au bitstream qui n'est pas négligeable dans la gamme de bas débits sur lesquels les simulations sont faites.

5.3 Scalabilité fractionnaire

Malgré leur flexibilité et leur grande sélectivité fréquentielle, les transformées M -bandes semblent cependant ne pas posséder de propriétés de scalabilité aussi fines que les transformées dyadiques. En effet, dans le banc de filtres M -bandes classique, seule la sous-bande résultant du filtrage passe-bas peut servir d'approximation de l'image originale, conduisant ainsi à un facteur de scalabilité d'ordre M .

Avant d'aborder ce problème, nous détaillons tout d'abord dans la section 5.3.1 la motivation et les raisons sous-jacentes à l'utilité de la scalabilité fine, notamment dans le domaine spatial. Nous aborderons de plus les stratégies actuellement utilisées pour pallier à l'absence d'une telle scalabilité. Nous rappellerons enfin dans cette section les liens étroits qu'entretient la notion de scalabilité avec les méthodes classiques utilisées pour changer la résolution d'une image.

Nous présentons et démontrons alors dans la section 5.3.2 une propriété étonnante, capable de donner à un banc de filtres M -bandes la faculté de pouvoir reconstruire l'image de départ réduite d'un facteur rationnel quelconque M/P . Cette propriété, nommée scalabilité fractionnaire ou scalabilité rationnelle, est obtenue par une simple modification du banc de synthèse, sans changement du banc d'analyse. Elle permet ainsi la construction de schémas de décodage par sous-bandes offrant des facteurs de scalabilité rationnels. Elle étend ainsi grandement la gamme de facteurs de scalabilité que l'on peut obtenir en codage par sous-bandes, comparée aux seuls facteurs dyadiques offerts par les transformées en ondelettes classiques.

Par rapport à une stratégie simple qui consisterait à reconstruire entièrement une image puis à la redimensionner à une résolution réduite M/P , notre technique permet de reconstruire *directement* l'image à la résolution réduite. Nous calculons dans la section 5.3.3 la complexité théorique des deux stratégies et montrerons alors la supériorité de notre approche.

Enfin, nous montrons en section 5.3.4 plusieurs résultats expérimentaux pour confirmer notre méthode. Nous présentons tout d'abord quelques images obtenues en modifiant un banc de synthèse M -bandes et évaluons la vitesse de reconstruction. Nous expérimentons alors divers filtres de rééchantillonnage pour évaluer leur influence sur la qualité de l'image reconstruite et donnons enfin de nombreux résultats d'efficacité de codage en utilisant la propriété de scalabilité rationnelle.

Ces travaux ont fait l'objet de la publication d'un article de revue [103] qui dans un premier temps propose et démontre la technique proposée, puis décrit la complexité théorique de notre approche. Les résultats expérimentaux ont quant à eux fait l'objet de la

soumission d'un article de conférence [102].

5.3.1 Motivation

Le codage d'image par transformée en ondelettes n'offre qu'une scalabilité de type dyadique où la sous-bande d'approximation peut être vue comme la version réduite d'un facteur 2 de l'image originale. Les décompositions successives de la sous-bande d'approximation permettent alors d'obtenir d'autres facteurs mais tous sont des puissances de deux et aucun autre facteur ne peut être obtenu directement. Enfin, les transformées M -bandes étudiées dans ce chapitre offrent une scalabilité encore plus grossière où tous les facteurs sont des puissances de M .

Divers articles [50, 86] soulignent pourtant la nécessité de pouvoir gérer une large gamme de facteurs de scalabilité et, si possible, de pouvoir tenir compte des facteurs rationnels. Prenons l'exemple des nouveaux formats HDTV 720p et 1080p, dont les résolutions sont liées par un facteur $2/3$. En utilisant une transformée en ondelettes classique, il n'est pas possible de construire un flux scalable capable d'être reconstruit à ces deux résolutions. Il n'est pas non plus possible de créer un flux capable de représenter deux formats n'ayant pas le même facteur de forme (*aspect ratio*) : c'est le cas des formats $16/9$ et $4/3$. La même situation se présente si on désire diffuser un contenu visuel scalable destiné à des téléphones mobiles qui ne possèdent pas les mêmes tailles d'écran.

Comment obtenir alors un schéma de codage capable d'offrir des rapports rationnels et non-dyadiques ? Bien qu'il n'existe pas de transformées capables d'offrir naturellement de tels rapports, il est possible d'utiliser des stratégies de type Simulcast ou des schémas de codage prédictif en couches pour contourner cette limitation. C'est en suivant cette approche que Marquant [86] a ainsi proposé d'utiliser des facteurs de scalabilité rationnels lors de la construction des différentes couches spatiales utilisées dans le codec SVC.

Une autre classe de stratégies consiste en la reconstruction totale de l'image, suivie d'une étape de redimensionnement afin que la taille de l'image décodée soit adaptée à la résolution du terminal récepteur. C'est cependant une approche coûteuse en terme de complexité car l'image complète doit être décodée à pleine résolution puis rééchantillonnée. On remarquera que d'autres techniques existent comme dans [96] où les auteurs proposent une stratégie de redimensionnement d'image dans le domaine transformé DCT en utilisant un module externe de transcodage. Cette approche ne nécessite pas la modification du parc de terminaux récepteurs mais est très coûteuse en termes de calculs car elle nécessite le décodage et le réencodage des images.

Notre approche

Nous proposons dans la section 5.3.2 une solution générique permettant de pallier à ces problèmes. Dans le contexte d'un banc de filtres M -bandes, cette solution consiste à modifier le banc de synthèse, de façon à ce que dernier puisse reconstruire une image d'une résolution réduite d'un facteur rationnel quelconque M/P . Nous montrons de plus que l'image reconstruite n'est pas quelconque ; elle est identique à celle qu'on aurait obtenue par redimensionnement de l'image originale au moyen d'un opérateur donné. Nous concluons enfin en montrant que cette modification permet de donner au banc de filtres M -bandes la faculté de pouvoir gérer les facteurs de scalabilité rationnels.

Les travaux de Gopinah [55] et Kovačević [70] traitent de bancs de filtres à reconstruction parfaite et à facteurs d'échantillonnage rationnels. Les auteurs abordent ici le

problème de la construction de bancs de filtres capable d'analyser un signal sur une division non-uniforme du spectre, où chaque branche du banc de filtres d'analyse possède un facteur d'échantillonnage rationnel. Sur la base de ces travaux, Vaidyanathan [35] a alors proposé un cadre unificateur, amenant à la construction de banc de filtres multidimensionnels rationnels. D'autres travaux [19, 22] ont fait le lien avec l'analyse multidimensionnelle rationnelle. Cependant, tous ces travaux abordent la construction de bancs de filtres où le banc d'analyse *et* le banc de synthèse sont simultanément spécifiés de façon à fournir *un seul* facteur de scalabilité rationnel. Notre approche est différente dans la mesure où nous proposons à partir d'un banc d'analyse M -bandes quelconque, de modifier son banc de synthèse associé de façon à fournir un facteur de scalabilité rationnel *quelconque* M/P , où M et P sont des entiers avec $P \leq M$.

On remarquera enfin les articles [90, 91] décrivant une méthode de codage d'image permettant l'obtention de facteurs rationnels de la forme $N/2^D$. Elle repose sur l'utilisation d'une décomposition en paquets d'ondelettes sur une base uniforme et consiste en une reconstruction suivie d'une étape de redimensionnement. Notre approche permet cependant l'obtention d'une gamme de facteurs plus étendue et est moins complexe car elle ne nécessite pas d'étape supplémentaire de redimensionnement. Enfin, une solution basée sur une DCT 8-bandes est proposée dans [68] : elle consiste à n'utiliser que P sous-bandes sur les 8 disponibles lors de la reconstruction et à modifier le facteur de suréchantillonnage par P . Cette solution est cependant restreinte à une transformation de type DCT et est susceptible de créer des artefacts de Gibbs lors de la reconstruction. Nous verrons dans la suite que nos travaux sont une généralisation de cette solution.

5.3.2 Modification du banc de synthèse

Dans le contexte d'un banc de filtres M -bandes, nous décrivons dans cette section comment modifier le banc de synthèse, afin que ce dernier puisse reconstruire une image d'une résolution réduite d'un facteur rationnel quelconque M/P . Cependant, la discussion sur le redimensionnement d'images par un facteur rationnel nécessite préalablement que l'on définisse clairement cette notion.

Redimensionnement et filtre de rééchantillonnage

Comment redimensionner une image d'un facteur M/P ? La question peut paraître anodine mais le sujet est pourtant vaste : il existe ainsi une littérature abondante sur le sujet [116, 160]. Cependant, la technique habituelle est linéaire et consiste à rééchantillonner l'image sur ses deux dimensions. Dans la suite, nous noterons alors indifféremment $[\downarrow \frac{M}{P}]$ l'opérateur de rééchantillonnage d'un facteur M/P et l'opérateur de redimensionnement de facteur M/P .

L'opérateur de rééchantillonnage $[\downarrow \frac{M}{P}]$ est défini comme la mise en cascade d'un opérateur de sur-échantillonnage de facteur P , d'un filtre de rééchantillonnage w et d'un opérateur de sous-échantillonnage de facteur M . Il permet de réduire la résolution d'un signal ou d'une image d'un facteur M/P et est illustré en Fig. 5.6.

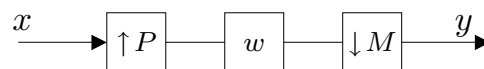


FIG. 5.6 – Opérateur de rééchantillonnage $[\downarrow \frac{M}{P}]$.

L'utilisation d'un filtre de rééchantillonnage w est rendue nécessaire pour éviter les effets de crénelage (*aliasing*) créés par le sous-échantillonnage $[\downarrow M]$. Son choix résulte d'un compromis entre la taille de son support, sa ressemblance avec un filtre passe-bas idéal de fréquence normalisée de coupure P/M et son aptitude à ne pas générer d'artefacts de type *ringing* en cas de quantification à bas débit.

Les filtres w classiquement utilisés en redimensionnement d'image sont le filtre échelon, qui permet une interpolation de type plus proche voisin, le filtre triangle, qui permet une interpolation de type linéaire, le filtre bicubique [65] et le filtre de Lanczos. On remarquera les travaux récents de Seidner [128] sur la construction de filtres de rééchantillonnage optimaux dans le domaine polyphase. Cependant, le choix et la construction d'un filtre w optimal dépasse largement les prétentions de ce document.

Nous nous limiterons dans ce document à l'utilisation du filtre échelon w_e et du filtre triangle w_t . Le filtre échelon w_e de taille P , permet une interpolation de type plus proche voisin et possède la réponse impulsionnelle suivante :

$$w_e[k] = \begin{cases} 1, & \text{si } 0 \leq k < P \\ 0, & \text{sinon} \end{cases} \quad (5.14)$$

Enfin, le filtre triangle w_t de taille $2P - 1$, permet une interpolation de type linéaire et sa réponse impulsionnelle est définie par :

$$w_t[k] = \begin{cases} 1 - |k/P|, & \text{si } -P < k < P \\ 0, & \text{sinon} \end{cases} \quad (5.15)$$

Modification proposée

Au moyen de l'opérateur de redimensionnement $[\downarrow \frac{M}{P}]$ précédemment défini, nous pouvons alors énoncer la proposition suivante :

Proposition 1 Soit un banc de filtres M -bandes à reconstruction parfaite. Dans le banc de synthèse, il suffit de remplacer les sur-échantillonneurs de facteur M par des sur-échantillonneurs de facteur P et d'utiliser des filtres de synthèse rééchantillonnés par un facteur M/P $\{\tilde{g}_k = [\downarrow \frac{M}{P}]g_k\}$ à la place des $\{g_k\}$ pour pouvoir reconstruire un signal de sortie qui est exactement la version rééchantillonnée d'un facteur M/P du signal d'entrée : $\tilde{y} = [\downarrow \frac{M}{P}]x$. Cette proposition est illustrée par la Fig. 5.7.

Donnons une preuve de cette proposition. Dans un premier temps, nous calculons la transformée de Fourier du signal $z = [\downarrow \frac{M}{P}]x$ et dans un second temps, celle du signal \tilde{y} . Nous montrons alors l'égalité des deux expressions.

Preuve. Calculons tout d'abord la transformée de Fourier de $z = [\downarrow \frac{M}{P}]x$. Au vu de la définition de l'opérateur de rééchantillonnage M/P , elle s'exprime par :

$$\hat{z}(f) = \frac{1}{M} \sum_{i=0}^{M-1} \hat{w}\left(\frac{f+i}{M}\right) \hat{x}\left(\frac{Pf+Pi}{M}\right)$$

De plus, sachant qu'il existe un entier unique k strictement inférieur à M tel que $Pi = Mq + k$ et à cause de la 1-périodicité de \hat{x} qui implique que $\hat{x}(f+q) = \hat{x}(f)$, nous pouvons alors réécrire $\hat{z}(f)$ par :

$$\hat{z}(f) = \frac{1}{M} \sum_{k=0}^{M-1} \alpha_k(f) \hat{x}\left(\frac{Pf+k}{M}\right) \quad \text{avec} \quad \alpha_k(f) = \sum_{\substack{0 \leq i < M \\ Pi \equiv k[M]}} \hat{w}\left(\frac{f+i}{M}\right) \quad (5.16)$$

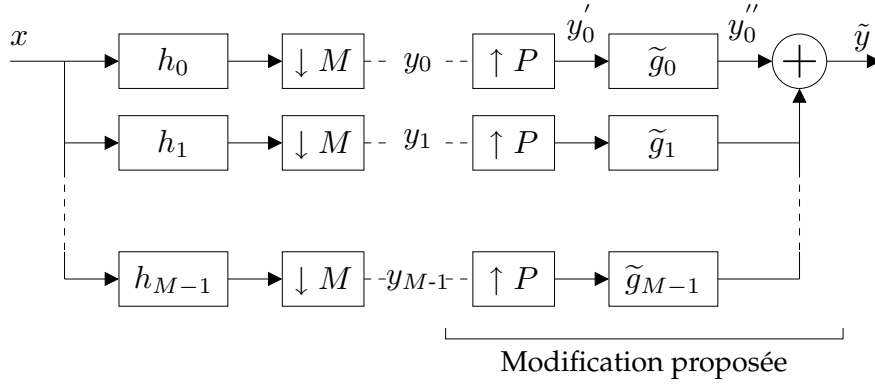


FIG. 5.7 – Banc de filtres M -bandes d'analyse et de synthèse. Ce dernier est en mesure de fournir une image réduite d'un facteur M/P .

Calculons désormais la transformée de Fourier du signal \tilde{y} . Considérons une sous-bande y_i issue du banc de filtres d'analyse. Sa transformée de Fourier $\hat{y}_i(f)$ peut s'exprimer par :

$$\hat{y}_i(f) = \frac{1}{M} \sum_{k=0}^{M-1} \hat{x}\left(\frac{f+k}{M}\right) \hat{h}_i\left(\frac{f+k}{M}\right)$$

Après sur-échantillonnage d'un facteur P , nous obtenons alors :

$$\hat{y}'_i(f) = \frac{1}{M} \sum_{k=0}^{M-1} \hat{x}\left(\frac{Pf+k}{M}\right) \hat{h}_i\left(\frac{Pf+k}{M}\right)$$

Suite au filtrage par les filtres de synthèses modifiés $\tilde{g}_i = [\downarrow \frac{M}{P}]g_i$, on a :

$$\hat{y}''_i(f) = \frac{1}{M^2} \left[\sum_{k=0}^{M-1} \hat{x}\left(\frac{Pf+k}{M}\right) \hat{h}_i\left(\frac{Pf+k}{M}\right) \right] \left[\sum_{j=0}^{M-1} \alpha_j(f) \hat{g}_i\left(\frac{Pf+j}{M}\right) \right]$$

Après permutation des sommes, la transformée de Fourier du signal \tilde{y} vaut donc :

$$\hat{\tilde{y}}(f) = \sum_{i=0}^{M-1} \hat{y}''_i(f) = \frac{1}{M^2} \sum_{k=0}^{M-1} \hat{x}\left(\frac{Pf+k}{M}\right) \sum_{j=0}^{M-1} \left[\alpha_j(f) \sum_{i=0}^{M-1} \hat{h}_i\left(\frac{Pf+k}{M}\right) \hat{g}_i\left(\frac{Pf+j}{M}\right) \right] \quad (5.17)$$

Or, la condition de reconstruction parfaite (5.4) des filtres M -bandes s'exprime par :

$$\forall f, 0 \leq i, j < M, \quad \sum_{k=0}^{M-1} \hat{h}_i\left(\frac{f+k}{M}\right) \hat{g}_j^*\left(\frac{f+k}{M}\right) = M\delta_{i-j}$$

En permutant les termes de sommation, cette condition peut se réécrire :

$$\forall f, 0 \leq l, k < M, \quad \sum_{i=0}^{M-1} \hat{h}_i\left(\frac{f+k}{M}\right) \hat{g}_i^*\left(\frac{f+l}{M}\right) = M\delta_{l-k} \quad (5.18)$$

L'injection de la condition (5.18) dans l'équation (5.17) nous permet alors d'obtenir l'expression finale de $\hat{\tilde{y}}(f)$:

$$\hat{\tilde{y}}(f) = \frac{1}{M} \sum_{k=0}^{M-1} \alpha_k(f) \hat{x}\left(\frac{Pf+k}{M}\right) \quad (5.19)$$

Les expressions (5.16) et (5.19) sont identiques. Nous avons donc $\tilde{y} = [\downarrow \frac{M}{P}]x$. ■

L'image reconstruite par le banc de synthèse modifié est donc l'image originale mais redimensionnée d'un facteur M/P par l'opérateur $z = [\downarrow \frac{M}{P}]x$. Il est ainsi clair que cette modification permet de donner aux bancs de filtres M -bandes une propriété de scalabilité rationnelle d'un facteur variable M/P , pour tout entier P inférieur à M et sans modification du banc d'analyse.

On notera de plus qu'il est possible de reconstruire un signal de taille réduite en utilisant seulement les Q premières sous-bandes $\{y_i\}_{0 \leq i < Q}$ fournies par le banc d'analyse. Dans ce cas, le signal reconstruit P_{V_Q} et l'erreur de reconstruction s'écrivent :

$$P_{V_Q}(x) = \sum_{j=0}^{Q-1} \sum_k g_j(n - Mk)y_j(k)$$

$$x - P_{V_Q}(x) = \sum_{j=Q}^{M-1} \sum_k g_j(n - Mk)y_j(k)$$

P_{V_Q} est donc un projecteur *oblique* du signal d'entrée x sur l'espace vectoriel $A = \text{Vect}\{(g_j(n - Mk))_{n \in \mathbb{Z}}, j \in \{0, \dots, Q-1\}, k \in \mathbb{Z}\}$. Plus précisément, $x - P_{V_Q}(x)$ est orthogonal à $B = \text{Vect}\{(h_j(n - Mk))_{n \in \mathbb{Z}}, j \in \{0, \dots, Q-1\}, k \in \mathbb{Z}\}$ puisque $\langle x - P_{V_Q}(x), h_j(n - Mk) \rangle = 0, \forall j \in \{0, \dots, Q-1\}$. En développant des arguments similaires à ceux utilisés pour montrer l'équation (5.19), nous pouvons déduire que le signal reconstruit à partir des Q premières sous-bandes et en utilisant le banc de synthèse modifié s'écrit alors :

$$\tilde{z}_Q = [\downarrow \frac{M}{P}]P_{V_Q}(x).$$

5.3.3 Complexité théorique

Nous nous proposons dans cette section de calculer la complexité théorique de notre structure avec modification du banc de synthèse et de la comparer à celle d'une structure naïve qui consisterait à reconstruire entièrement le signal puis à le redimensionner. Une telle structure est illustrée par la Fig. 5.8.

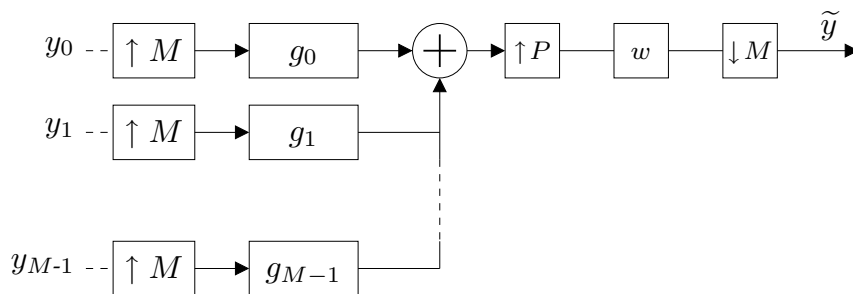


FIG. 5.8 – Reconstruction à pleine résolution par un banc de synthèse M -bandes avec une étape de redimensionnement.

La représentation polyphase [155] permet une implémentation efficace des structures présentées. Tout d'abord, un algorithme rapide pour réaliser l'opération de rééchantillonnage décrite par l'opérateur $[\downarrow \frac{M}{P}]$ en section 5.3.2 possède une complexité de nW/M

multiplications, où n est la taille du signal d'entrée et W la longueur de la réponse impulsionnelle du filtre w . De plus, les opérateurs d'interpolation intervenant dans le banc de synthèse M -bandes consistent au sur-échantillonnage d'un signal d'un facteur M suivi d'une convolution par un filtre d'une réponse impulsionnelle de longueur N . Une implémentation efficace d'un tel opérateur nécessite alors nN multiplications. Si Q sous-bandes sont considérées pour la reconstruction, le banc de synthèse M -bandes possède ainsi une complexité de nQN multiplications.

Nous sommes alors en mesure de calculer les complexités théoriques des deux structures de reconstruction, exprimées en nombre de multiplications pour Q sous-bandes de taille n . Les résultats sont présentés sur le Tab. 5.9. On observe ainsi la réduction théorique de la complexité apportée par notre structure.

Structure	Complexité (en multiplications)
Proposée	$\frac{n}{M} \left(NQ \frac{P}{M} + W \frac{Q}{M} \right)$
Avec post-redimensionnement	$\frac{n}{M} (NQ + W)$

TAB. 5.9 – Comparaison de la complexité des deux structures.

5.3.4 Résultats expérimentaux

Afin d'illustrer l'intérêt de notre proposition, nous avons conduit plusieurs expérimentations en utilisant des bancs de filtres M -bandes variés, des paramètres P et Q différents et plusieurs filtres de rééchantillonnage w . Dans un premier temps, nous visualisons l'influence des paramètres P , Q et du filtre de rééchantillonnage w . Nous mesurons alors la durée de reconstruction réelle accomplie par notre structure et la comparons à une structure avec post-redimensionnement afin de justifier nos résultats de complexité théorique. Nous présentons enfin des résultats concernant l'efficacité de codage en utilisant plusieurs bancs de filtres et en faisant varier les paramètres P et Q .

Scalabilité rationnelle

Nous nous proposons dans cette section d'utiliser le banc de filtres 8-bandes à 16 échantillons proposé par Alkin [12] qui est bien adapté à la compression d'image et au codage vidéo. En appliquant la modification du banc de synthèse telle que décrite en section 5.3.2 et en utilisant le filtre de rééchantillonnage linéaire w_t , nous pouvons reconstruire une image d'une résolution réduite d'un facteur $8/P$ à partir de Q sous-bandes. Aucun codage n'est effectué dans cette expérimentation. La Fig. 5.9 illustre les résultats obtenus sur *Lena* en faisant varier le paramètre P de 1 à 8 et en utilisant un nombre de sous-bandes Q égal à P . On observe ainsi la finesse de la scalabilité offerte par notre méthode.

Dans une expérimentation sans codage, le nombre de sous-bandes Q permet d'influer sur la qualité de la reconstruction. La Fig. 5.10 montre les résultats obtenus sur *Barbara* en utilisant un paramètre $P = 5$ fixe et un nombre de sous-bandes Q variant de 1 à 8. On observe l'augmentation très rapide de la qualité de l'image en fonction de Q et ceci est très lié à la sélectivité fréquentielle du banc de filtres. Dans la sous-section suivante consacrée au codage, nous verrons l'utilité qu'offre la possibilité de reconstruire l'image en utilisant un nombre réduit de sous-bandes.

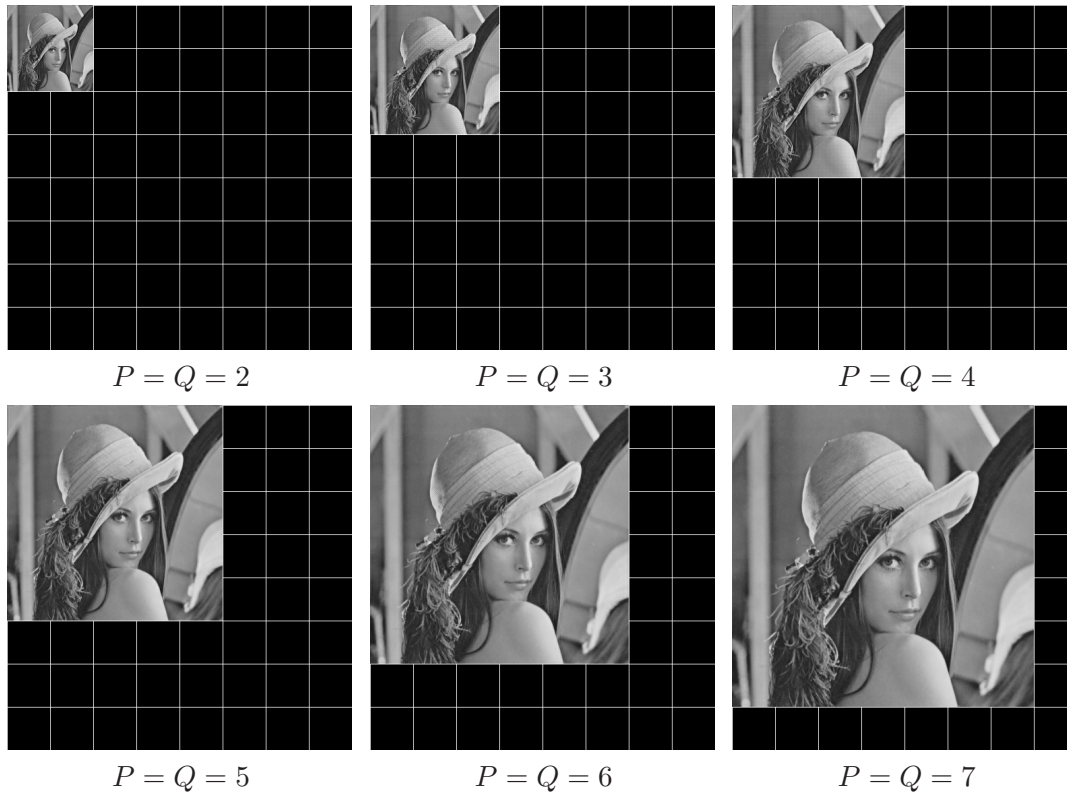


FIG. 5.9 – Illustration de différents facteurs de scalabilité M/P obtenus par la méthode proposée sur l'image *Lena* avec $M = 8$, $P = \{2, 3, 4, 5, 6, 7\}$ et $Q = P$.

Influence du filtre de rééchantillonnage

Quelle est l'influence du filtre de rééchantillonnage w sur les images reconstruites ? Afin de répondre à cette question, nous avons conduit sur *Barbara* des expérimentations avec $P = 5$, un nombre variable de sous-bandes Q et en comparant les filtres de rééchantillonnage échelon w_e et triangle w_t . Les résultats sont présentés en Fig. 5.11. On observe la meilleure qualité visuelle des images obtenues avec le filtre triangle w_t , dû à ses propriétés d'interpolation linéaire. Son utilisation conduit ainsi à l'obtention d'images lisses et dépourvues d'artefacts de crénelage, très visibles au contraire dans le cas $Q = 8$, w_e .

Durée de reconstruction

Afin d'illustrer les résultats de complexité théorique, nous nous plaçons encore dans un scénario sans codage, muni de la transformée 8-bandes d'Alkin. Afin de reconstruire une image réduite d'un facteur $8/P$ à partir de Q sous-bandes, nous considérons tout d'abord notre structure avec modification du banc de filtres de synthèse. Nous considérons alors une deuxième structure qui consiste naïvement en la reconstruction du signal à pleine résolution, suivie d'une étape de rééchantillonnage. En réutilisant les paramètres mis en œuvre dans les Fig. 5.9 et 5.10 et le filtre de rééchantillonnage w_t , nous présentons dans les Tabs. 5.10 et 5.11 les durées de reconstruction obtenues avec les deux structures sur un Pentium IV cadencé à 2.8 GHz. On observe clairement la supériorité de notre approche, montrant des durées de reconstruction jusqu'à 3 fois inférieures à la structure naïve.

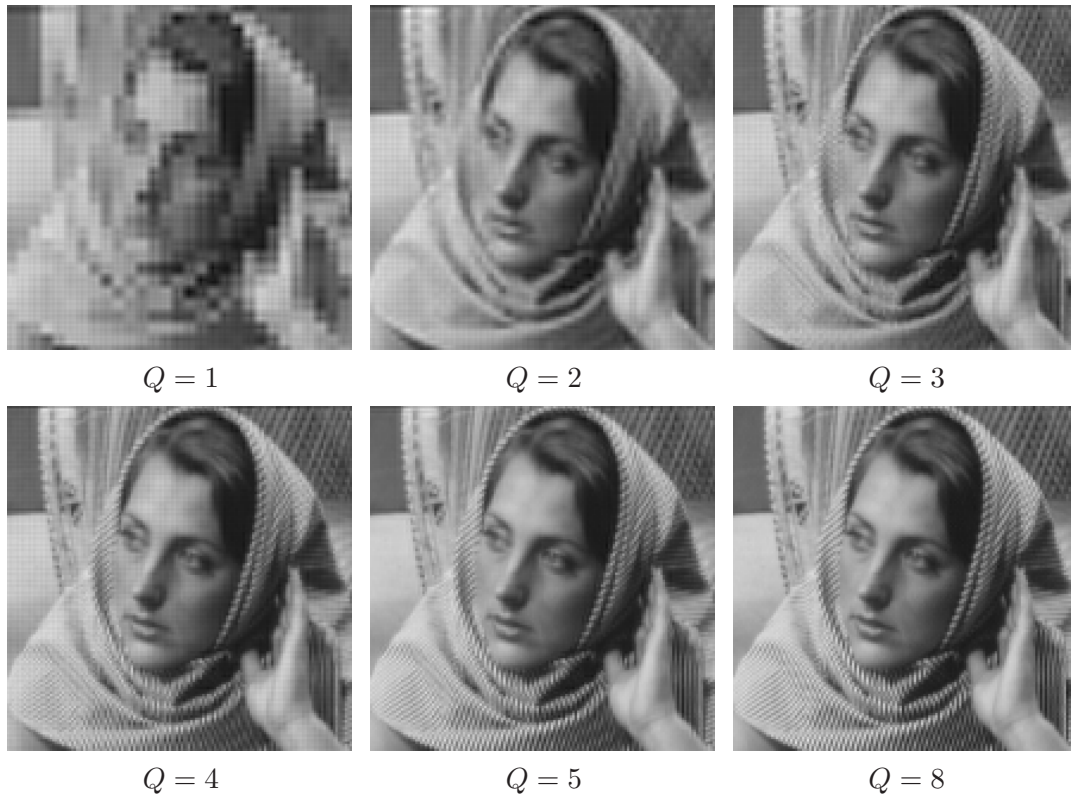


FIG. 5.10 – Illustration de différentes reconstructions en utilisant un nombre variable de sous-bandes Q sur l'image *Barbara* avec $M = 8$, $P = 5$ et $Q = \{1, 2, 3, 4, 5, 8\}$.

Durée (en ms)	$P = 2$	$P = 3$	$P = 4$	$P = 5$	$P = 6$	$P = 7$	$P = 8$
Structure proposée	62.3	77.8	102.3	126.8	162.3	207.4	263.5
Avec rééchantillonnage	168.1	199.8	247.8	283.8	320.5	373.5	262.9

TAB. 5.10 – Durées mesurées pour reconstruire *Lena* au moyen des deux structures présentées et avec les paramètres utilisés dans la Fig. 5.9, $M = 8$ et $Q = P$.

Durée (en ms)	$Q = 1$	$Q = 2$	$Q = 3$	$Q = 4$	$Q = 5$	$Q = 6$	$Q = 8$
Structure proposée	60.5	78.7	95.8	108.1	126.0	143.7	180.5
Avec rééchantillonnage	186.4	214.7	241.3	258.2	284.6	311.9	369.4

TAB. 5.11 – Durées mesurées pour reconstruire *Barbara* au moyen des deux structures présentées et avec les paramètres utilisés dans la Fig. 5.10, $M = 8$, $P = 5$ et plusieurs valeurs de Q .

Efficacité de codage avec reconstruction totale

Notre méthode permet de donner la propriété de scalabilité rationnelle à un banc de filtres M -bandes. Cette propriété peut s'avérer très utile en codage d'image où l'on souhaite créer un train binaire, scalable, unique et capable d'être décodé dans de nombreuses résolutions. Cependant, il n'est pas souhaitable que la propriété de scalabilité rationnelle

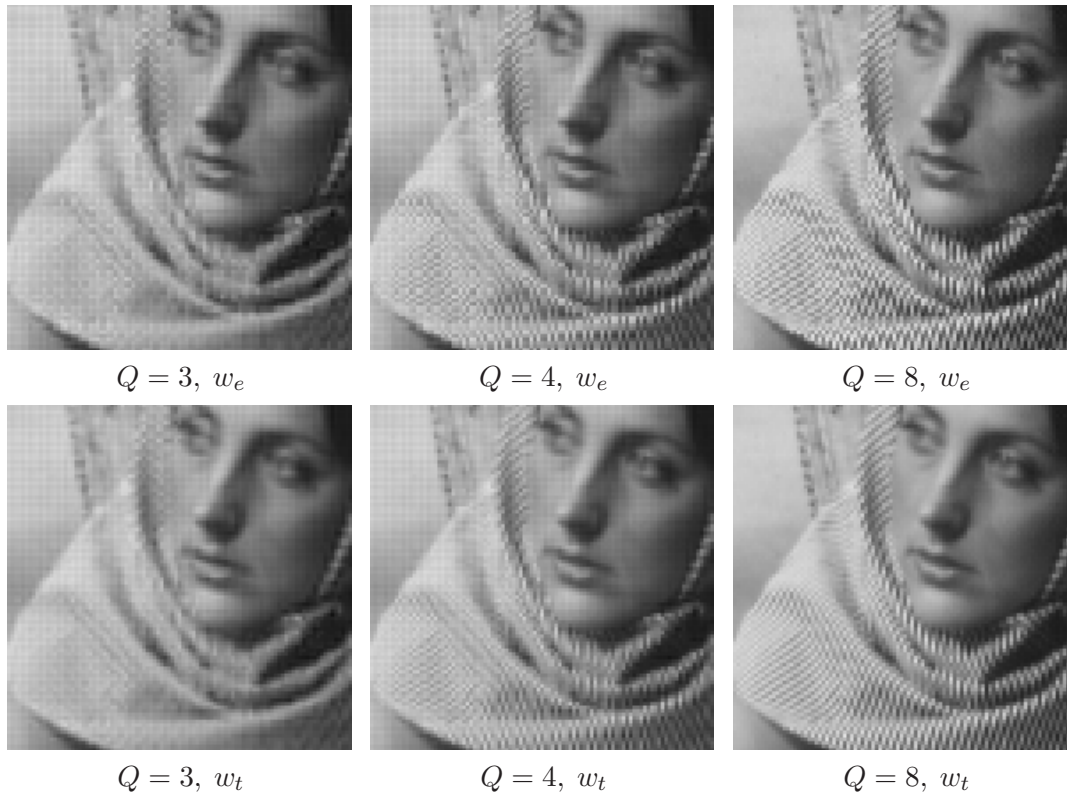


FIG. 5.11 – Influence du filtre de rééchantillonnage w_e et w_t , illustrée par différentes reconstructions en utilisant un nombre de variable de sous-bandes Q sur l'image *Barbara* avec $M = 8, P = 5$ et $Q = \{3, 4, 8\}$.

nuise à l'efficacité de codage. Afin de répondre à cette question, nous avons souhaité tester dans cette section l'efficacité de codage de plusieurs transformées M -bandes en les comparant à la transformée dyadique 9/7.

Nous avons considéré dans nos expérimentations les transformées 8-bandes suivantes : transformée d'Alkin [12] à 16 échantillons, DCT, LOT, LBT et une transformée en paquets d'ondelettes 9/7 avec une base uniforme. Nous avons de plus testé la transformée dyadique 9/7 dont l'efficacité a été prouvée par le succès du codec JPEG-2000. Les images sont décomposées sur un niveau spatial avec les transformées 8-bandes et sur trois niveaux avec la transformée dyadique, de façon à obtenir une sous-bande spatiale d'approximation de même taille. Les structures de décomposition utilisées pour les transformées 8-bandes et pour la transformée dyadique sont illustrées en Fig. 5.12, où Q et Q' dénotent respectivement le nombre de sous-bandes utilisées pour la reconstruction dans le cas 8-bandes et dans le cas dyadique. Les expérimentations ont été conduites sur le codec JPEG-2000 VM 8.0, en utilisant l'implémentation logicielle de Malvar [81] de la DCT, la LOT et de la LBT. Le rééchantillonnage des filtres est fait au moyen du filtre triangle w_t , jouant le rôle d'interpolateur linéaire.

Nous souhaitons dans un premier temps évaluer l'efficacité de codage offerte par les transformées M -bandes avec reconstruction totale, sans modification du banc de synthèse ($P = 8$) et en utilisant toutes les sous-bandes lors de la reconstruction ($Q = 8$). Les Tabs. 5.12 et 5.13 présentent les résultats obtenus en comparant plusieurs transformées 8-bandes et la transformée dyadique 9/7.

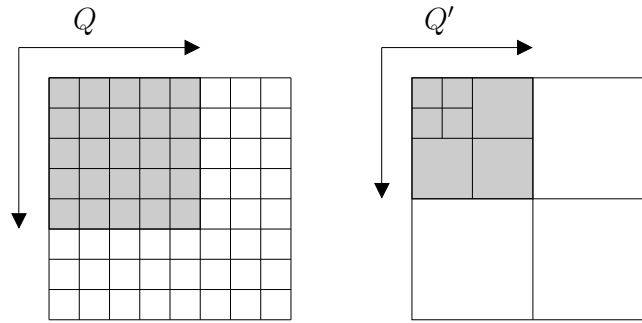


FIG. 5.12 – Bases de décomposition utilisées pour les transformées 8-bandes (gauche) et pour la transformée dyadique 9/7 (droite). Q et Q' dénotent le nombre de sous-bandes utilisées pour la reconstruction.

Débit (bpp)	0.1	0.2	0.4	1.0
9/7-dyadique	24.47	27.02	30.61	37.02
Alkin	24.29	26.63	29.79	35.25
LOT	25.00	27.84	31.34	37.23
LBT	25.27	28.17	31.76	37.70
DCT	23.92	26.44	29.78	36.00
9/7-paquets	25.27	28.00	31.38	37.43

TAB. 5.12 – Comparaison débit-PSNR en dB de la transformée dyadique 9/7 et de plusieurs transformées 8-bandes en reconstruisant *Barbara* à pleine résolution ($P = Q = 8$).

Débit (bpp)	0.1	0.2	0.4	1.0
9/7-dyadique	29.62	32.74	36.05	40.36
Alkin	28.31	31.09	34.06	38.79
LOT	29.06	32.05	35.27	39.47
LBT	29.48	32.63	35.71	39.82
DCT	28.20	31.29	34.73	39.27
9/7-paquets	29.72	32.78	35.80	39.85

TAB. 5.13 – Comparaison débit-PSNR en dB de la transformée dyadique 9/7 et de plusieurs transformées 8-bandes en reconstruisant *Lena* à pleine résolution ($P = Q = 8$).

On observe que la transformée LBT offre les meilleurs résultats sur *Barbara* tandis que la transformée dyadique 9/7 semble plus efficace sur *Lena*. Il apparaît enfin que ces deux transformées ont la meilleure efficacité globale sur l'ensemble des deux images. Ces résultats sont confirmés sur la Fig. 5.13, où l'on observe la meilleure qualité des images reconstruites avec la transformée dyadique 9/7 et la LBT. Elles surpassent notamment la DCT, qui présente de sévères artefacts de type bloc. En accord avec le codec PTC [84] issu des travaux de Malvar sur la compression d'image par codage emboîté des coefficients LBT, ces résultats montrent que la transformée LBT donne de bons résultats et reste compétitive en comparaison avec la transformée dyadique 9/7.

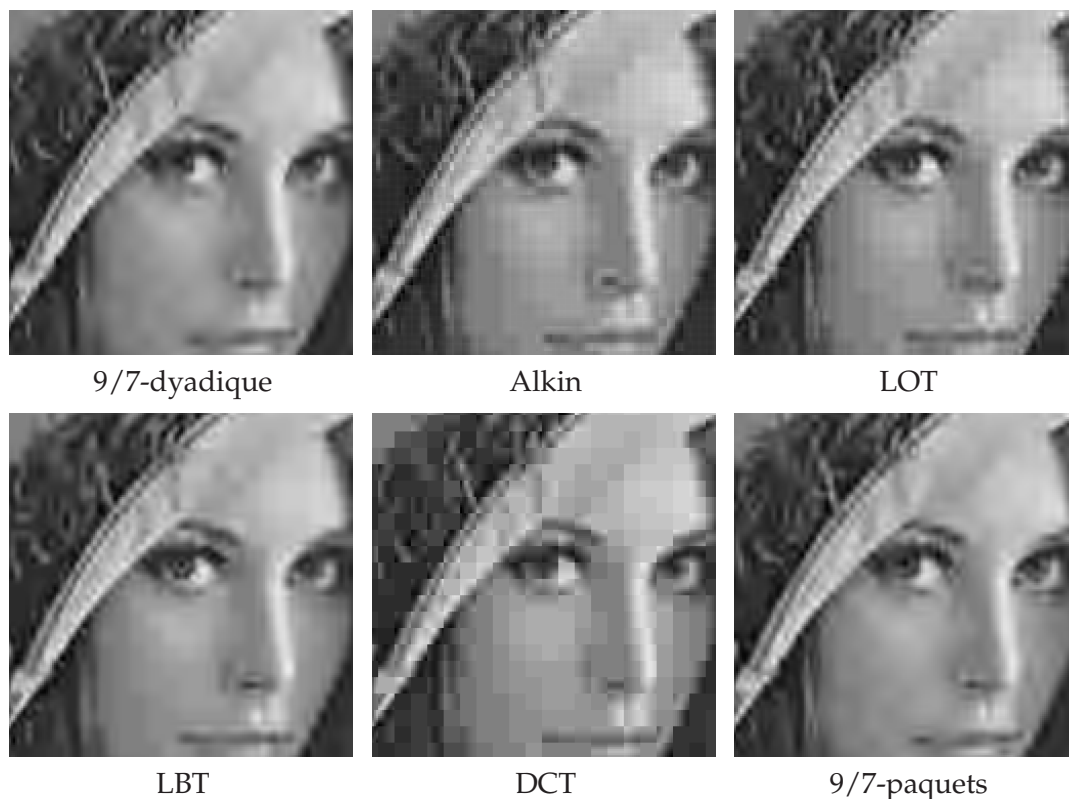


FIG. 5.13 – Zooms sur des images reconstruites à pleine résolution ($P = Q = 8$) de *Lena* avec différentes transformées, à un débit de 0.1 bpp.

Efficacité de codage avec scalabilité rationnelle

Nous souhaitons désormais évaluer l'efficacité de codage offerte par les transformées M -bandes en utilisant la scalabilité rationnelle dans le scénario suivant. Supposons un réseau où un serveur de contenu diffuse une image compressée à destination d'un parc de récepteurs possédant des écrans de tailles différentes. Certains terminaux sont des PDA capables d'afficher des images 512 pixels de largeur tandis que d'autres sont des smartphones limités à 320 ou 192 pixels. Enfin, les récepteurs de type téléphone portables possèdent des écrans de seulement 128 pixels de large. Ces contraintes exigent alors la construction d'un train binaire scalable capable de tenir compte des facteurs de scalabilité rationnels $8/8$, $8/5$, $8/3$ et $8/2$. Nous allons mettre en place deux stratégies pour répondre aux exigences de ce scénario. La première est basée sur une transformée dyadique suivie d'une étape de redimensionnement tandis que la seconde repose sur une transformée M -bandes munie de la scalabilité rationnelle.

Comme vu précédemment, un schéma de codage basé sur une transformée en ondelettes dyadique ne peut pas offrir de manière native des facteurs de scalabilité non dyadiques. De plus, la stratégie Simulcast étant trop coûteuse et le schéma de codage prédictif en couches n'étant pas assez souple, il nous faut utiliser une technique avec post-redimensionnement. Le terminal récepteur est alors capable de recevoir Q' sous-bandes compressées, de reconstruire l'image à pleine résolution, de la redimensionner d'un facteur $8/P$ et l'adapter ainsi à la taille de son écran. Notons qu'à cause des contraintes de la pyramide dyadique, Q' est nécessairement une puissance de deux.

Au contraire, un schéma de codage reposant sur une transformée M -bandes et utilisant notre propriété de scalabilité rationnelle est capable de fournir directement les facteurs de scalabilité exigés par le scénario. Il est alors aisé de créer un train binaire scalable capable d'être décodé aux résolutions citées précédemment. On notera que les décompositions M -bandes offrent une granularité plus fine sur le choix du nombre de bandes : Q peut ainsi être choisi de façon quelconque entre 1 et M .

Nous comparons tout d'abord dans les Tabs. 5.14 et 5.15 l'efficacité de codage des transformées 8-bandes et de la transformée dyadique 9/7 obtenues sur les images *Barbara* et *Lena*, en utilisant $P = Q = 5$. Cette expérimentation est équivalente au scénario proposé où le terminal récepteur ne peut recevoir que $Q = 5$ bandes du train binaire et possède une résolution réduite d'un facteur 8/5 par rapport à la taille de l'image originale. De plus, à cause des restrictions incombant aux décompositions dyadiques, Q' a du être abaissé à 4.

Débit (bpp)	0.1	0.2	0.4	1.0
9/7-dyadique	25.44	26.71	28.10	28.88
Alkin	25.50	27.42	29.08	30.47
LOT	26.18	28.51	30.72	32.43
LBT	26.51	28.79	30.90	32.48
DCT	25.10	27.24	29.77	32.18
9/7-paquets	25.62	27.19	28.28	28.93

TAB. 5.14 – Comparaison débit-PSNR en dB de la transformée dyadique 9/7 et de plusieurs transformées 8-bandes en reconstruisant *Barbara* à la résolution 8/5 ($P = Q = 5$).

Débit (bpp)	0.1	0.2	0.4	1.0
9/7-dyadique	30.89	34.23	36.76	38.08
Alkin	29.42	32.35	34.65	36.40
LOT	30.34	33.97	37.45	40.51
LBT	30.72	34.41	38.04	41.14
DCT	29.15	32.82	36.83	40.54
9/7-paquets	31.01	34.24	36.56	38.18

TAB. 5.15 – Comparaison débit-PSNR en dB de la transformée dyadique 9/7 et de plusieurs transformées 8-bandes en reconstruisant *Lena* à la résolution 8/5 ($P = Q = 5$).

Nous observons que les transformées 8-bandes surpassent nettement la transformée dyadique 9/7. Les résultats obtenus avec la LBT montrent en particulier un gain allant jusqu'à 3 dB. Ceci peut-être expliqué par le fait que la structure pyramidale de la transformée 9/7 dyadique impose à cette dernière de n'utiliser que $Q' = 4$ bandes alors que les transformées 8-bandes disposent de $Q = 5$ bandes. Le gain observé pourrait ainsi être expliqué par le manque de flexibilité de la structure de la décomposition dyadique.

Nous avons cependant voulu vérifier cette hypothèse en relançant les mêmes simulations avec $P = 5$ mais en utilisant le même nombre de bandes $Q = Q' = 4$. Les résultats obtenus sur *Lena* sont présentés dans le Tab. 5.16. On observe encore la supériorité de l'efficacité de codage des transformées LBT et en paquets d'ondelettes 9/7, comparée à

la transformée dyadique 9/7. Cependant, le gain obtenu est inférieur à celui observé lors de la simulation précédente et peut s'expliquer par la meilleure sélectivité fréquentielle des transformées 8-bandes. En effet, Li avait déjà constaté [75] la mauvaise sélectivité fréquentielle du filtre passe-haut dyadique 9/7 en observant une fuite importante d'énergie dans sa bande de coupure. Cette caractéristique peut alors conduire à une augmentation de l'amplitude des coefficients d'ondelettes, réduisant ainsi l'efficacité globale de codage.

Débit (bpp)	0.1	0.2	0.4	1.0
9/7-dyadique	30.89	34.23	36.76	38.08
Alkin	29.34	31.88	33.54	34.46
LOT	30.34	33.83	36.52	38.28
LBT	30.72	34.19	37.03	38.60
DCT	29.14	32.64	35.75	37.54
9/7-paquets	31.01	34.24	36.55	38.09

TAB. 5.16 – Comparaison débit-PSNR en dB de la transformée dyadique 9/7 et de plusieurs transformées 8-bandes en reconstruisant *Lena* à la résolution 8/5 ($P = 5, Q = 4$).

Nous comparons désormais l'efficacité de codage obtenue sur *Lena* avec un coefficient de réduction de 8/3, avec $P = 3$ et $Q = 4$. Les résultats sont présentés dans le Tab. 5.17. On observe encore la supériorité de l'efficacité de codage des transformées 8-bandes LBT et 9/7-paquets, comparée à la transformée dyadique 9/7.

Débit (bpp)	0.1	0.2	0.4	1.0
9/7-dyadique	30.88	34.14	36.74	38.09
Alkin	29.01	31.52	33.11	33.98
LOT	30.15	33.55	36.22	37.96
LBT	30.65	34.04	36.83	38.36
DCT	28.70	32.14	35.29	37.24
9/7-paquets	30.98	34.18	36.53	38.12

TAB. 5.17 – Comparaison débit-PSNR en dB de la transformée dyadique 9/7 et de plusieurs transformées 8-bandes en reconstruisant *Lena* à la résolution 8/3 ($P = 3, Q = 4$).

La Fig. 5.14 montre les images obtenues lors de la reconstruction avec les différentes transformées en utilisant $P = Q = 3$ et un débit de 0.4 bpp. Les résultats sont comparés avec une version de référence non-codée et obtenue par redimensionnement de l'image originale d'un facteur 8/3, en utilisant le filtre d'interpolation linéaire w_t . Nous observons clairement la meilleure qualité de l'image reconstruite avec la LBT, comparée à celle obtenue avec la transformée dyadique 9/7.

Un autre exemple illustrant la mauvaise sélectivité fréquentielle des filtres mis en jeu dans la transformée 9/7 est présenté en Fig. 5.15. On remarquera sur l'image obtenue avec le filtre 9/7 les artefacts de crénelage très visibles sur le foulard porté par Barbara. De tels artefacts ne sont pas présents en utilisant la transformée LBT.



FIG. 5.14 – Zooms sur des images reconstruites   la r solution r duite de $8/3$ ($P = Q = 3$) de *Lena*   0.4 bpp.

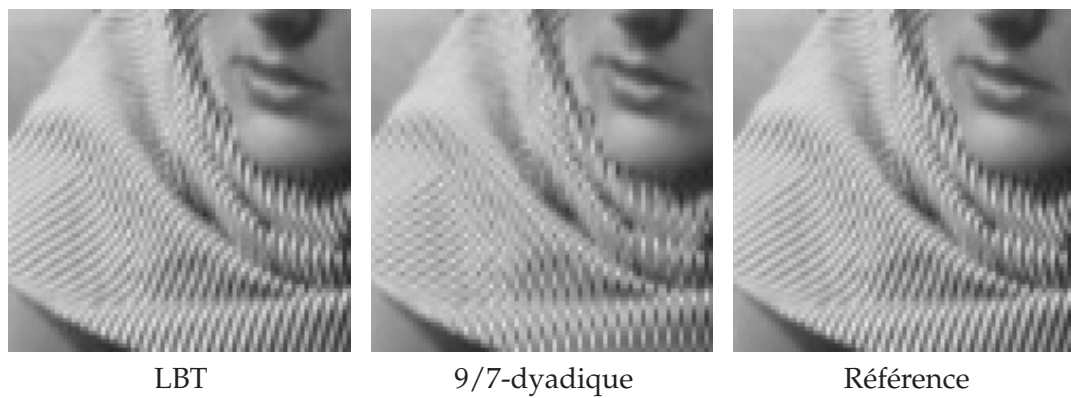


FIG. 5.15 – Zooms sur des images reconstruites   la r solution r duite de $8/6$ ($P = Q = 6$) de *Barbara*   0.4 bpp.

5.4 Conclusion

Le principe du sch ma de codage vid o $t+2D$ repose sur l'utilisation d'un filtre temporel compens  en mouvement, pour tirer b n fice de la redondance temporelle des images, suivie d'une d composition spatiale des sous-bandes r sultantes afin d'exploiter leur redondance spatiale. L'utilisation de la transform e biorthogonale 9/7 est justifi e pour la transformation des sous-bandes temporelles d'approximation, ressemblant   des images naturelles, mais ne l'est pas dans le cas des sous-bandes de d tail qui comportent de larges zones de texture   hautes fr quences.

De part leur grande s lectivit  fr quentielle et leur flexibilit , les bancs de filtres M -bandes sont des candidats id aux pour d composer ces sous-bandes temporelles de d tail. Dans un premier temps, nous avons construit un banc de filtres 4-bandes sym trique, orthogonal et disposant d'un nombre suffisant de moments nuls pour lui permettre d'assurer une bonne approximation polynomiale. Sa mise en place au sein du codec MC-EZBC a conduit   un gain en PSNR moyen d'environ 0.3 dB par rapport   la transform e 9/7. Nous avons alors proc d    une  tude plus compl te avec d'autres transform es M -bandes couramment utilis es en compression d'image (LOT, LBT, MLT) : leur utilisation au sein du codec Vidwav a cependant conduit   un gain en PSNR plus modeste.

En dépit de leurs qualités, les bancs de filtres M -bandes ne possèdent cependant pas de propriétés de scalabilité aussi fines que leurs homologues dyadiques et ne peuvent fournir seulement que des facteurs de scalabilité d'ordre M . Afin de pallier à cet inconvénient, nous avons montré qu'une simple modification du banc de filtres de synthèse permet d'obtenir un facteur de scalabilité rationnel quelconque P/M , où P est un entier inférieur à M . Cette propriété autorise ainsi la construction de schémas de décodage dotés de scalabilité fractionnaire et permet de disposer d'une vaste gamme de facteurs de scalabilité, pour n'importe quelle transformée M -bandes et sans nécessiter un changement du banc d'analyse. Nous avons alors montré la réduction en complexité offerte par cette approche, comparée à une stratégie qui consiste à reconstruire entièrement le signal puis à le redimensionner. Enfin, nous avons montré l'intérêt de cette propriété lors d'un scénario de diffusion de contenu à destination d'un parc de récepteurs possédant des écrans de tailles différentes, En effet, l'utilisation de la transformée LBT 8-bandes et de la scalabilité fractionnaire dans ce scénario permet de reconstruire directement l'image dans une résolution réduite, tout en offrant une meilleure efficacité de codage que l'utilisation de la transformée 9/7 couplée à un opérateur de redimensionnement.
