

White paper sur la compression d'images par la *Fast Wavelet Transform*

Daniel Lemire[†], Ph.D. et Alain Béliveau[‡] Ph.D.

Ce rapport résume la technologie de compression d'images par ondelettes développées au sein de la société de Compression d'Images en réseaux et Automatismes (CIRA).

I. INTRODUCTION

La théorie des ondelettes est encore très récente. On estime généralement que l'ingénieur français Morlet en est l'inventeur (début des années 1980). Plus tard, un mathématicien français, Meyer, fut le premier à donner des assises mathématiques solides à la théorie naissante (1986). Cependant, l'histoire retiendra que la mathématicienne américaine Daubechies fut la première à fournir des ondelettes *of industrial strength* (1992).

Daubechies a raconté qu'elle imagina ses ondelettes alors qu'elle était l'invitée des professeurs Deslauriers et Dubuc, de l'Université de Montréal, tous deux spécialistes de l'interpolation itérative qui est une des bases de la théorie moderne des ondelettes. Comme c'était l'hiver à Montréal et qu'il faisait très froid, elle demeura prisonnière de sa chambre d'hôtel durant tout son séjour, forcée à travailler...

L'une des premières applications vraiment excitante des ondelettes est le résultat d'une étude commandée par le FBI concernant la compression des empreintes digitales. En effet, aux États-Unis, 40 000 empreintes digitales doivent être traitées, classées et comparées chaque jour. Outre le FBI, l'ensemble des corps policiers des États-Unis doit gérer de multiples bases de données dont les coûts d'exploitation sont relativement importants. On avait d'abord songé à la compression des données à l'aide du standard JPEG (*Joint Picture Expert Group*) qui est basé sur une DCT (*Discrete Cosine Transform*) par blocs de 8 par 8 (voir les figures 1 et 2). Dans le cas du FBI, les images sont monochromes (256 teintes), sont toujours de la même taille et se ressemblent fortement. Dans la figure 1, on peut très bien voir les blocs de 8 par 8 sur l'image compressée à gauche.



Figure 1. Empreinte digitale compressée avec JPEG

Alors que JPEG réussit très bien la compression pour de faibles ratios de l'ordre de 5:1 à 10:1, les résultats sont souvent inacceptables à des ratios de 15:1 à 20:1. Ce n'est souvent pas le cas avec les ondelettes. Les ondelettes sont maintenant utilisées tous les jours par le FBI et de nombreux corps policiers pour traiter des milliers d'empreintes.

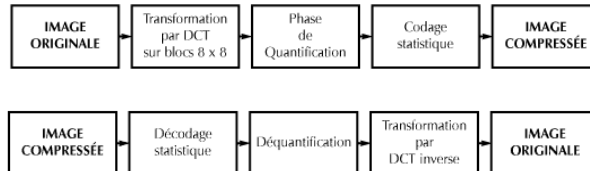


Figure 2. Algorithme de compression JPEG

Un des avantages certains de la compression par ondelettes est qu'il s'agit d'une approche globale et non *par blocs* : les effets de *mosaïque* sont donc rares, contrairement à JPEG.



Figure 3. Comparaison qualitative entre JPEG (à gauche) et la technologie CIRA (à droite).

[†] Institut de génie biomédical, Université de Montréal, Montréal, Canada, Tél. (514) 524-3517, Télécopieur (514) 524-0141, courriel Daniel.Lemire@Videotron.ca

[‡] Département de Mathématiques Appliquées, École Polytechnique de Montréal, Montréal, Canada, Tél. (514) 340-4711 poste 4292, Télécopieur (514) 340-4463, email Beliveau@MathAppl.PolyMtl.Ca

II. SCHÉMA PROPOSÉ

À l'origine, les chercheurs travaillant pour le FBI avaient imaginé l'utilisation des algorithmes d'optimisation dans le genre *Matching Pursuit* (Stéphane Mallat) ou *Best Basis* (David Donoho). Cependant, ce type d'algorithme est plus lent que la *Fast Wavelet Transform* sur lequel il repose et les gains peuvent parfois être faibles. Le modèle finalement adopté, au grand désarroi de certains chercheurs, est néanmoins astucieux. Le FBI s'est rendu compte que les algorithmes d'optimisation en arrivaient toujours ou presque à faire les mêmes choix. À partir de cette constatation, il fut possible d'obtenir un algorithme qui était à la fois très rapide et aussi optimisé. Ce phénomène s'explique tout simplement par le fait que les empreintes digitales sont assez semblables en général.

CIRA doit cependant utiliser un algorithme plus flexible qui n'est pas nécessairement optimal pour un type d'images en particulier, mais qui offre de bonnes performances en général. En bref, il s'agit de fournir une alternative à JPEG qui soit nettement meilleure dans certaines conditions notamment lorsqu'il s'agit de compresser agressivement des images de bonne taille.

Une des préoccupations particulières à CIRA est la rapidité de l'algorithme de décompression. De plus, comme il s'agit de traiter des images provenant du format JPEG, l'algorithme de compression doit fonctionner avec un codage à 24 bits (auss appelé *true color*).

A. Approche par canaux

Au lieu de considérer une image comme étant un bloc de pixels, il est utile d'utiliser une représentation en n canaux (voir figure 4). La représentation en n canaux sans doute la plus connue en informatique est le codage RGB (*Red-Green-Blue*). Outre les considérations mathématiques, il y a trois raisons qui favorisent l'utilisation d'une représentation en n canaux :

1. les fonctions utilisées sont les mêmes pendant le traitement d'une image en tons de gris (1 seul canal nécessaire) et les images couleurs (3 canaux);
2. la prise en compte du mécanisme de perception humain (voir ci-après);
3. la possibilité de programmer l'affichage de l'image d'une façon progressive, un canal à la fois.

Le deuxième point doit être expliqué un peu plus longuement. L'œil humain distingue moins facilement les variations chromatiques (Cb/Cr ou I/Q), tout en étant plus sensible à la luminance Y. C'est pourquoi les chercheurs recommandent l'utilisation des modèles YCbCr ou YIQ (par opposition à RGB). On obtient de meilleurs taux de compression en utilisant des canaux moins riches pour les

composantes chromatiques (Cb/Cr ou I/Q). Dans le domaine de la télévision, on utilise cette même technique pour les caméras numériques : les composantes Cb et Cr (ou I et Q) sont échantillonnées quatre fois moins que la composante Y.

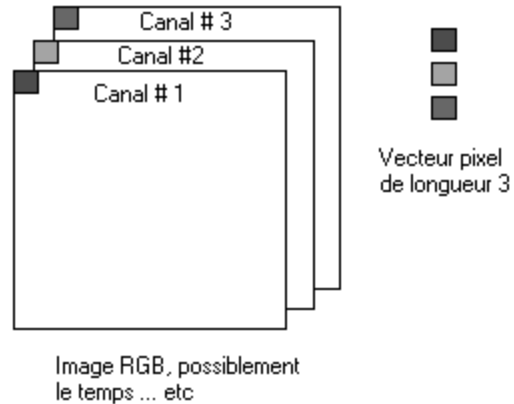


Figure 4. Représentation d'une image par canaux

B. Les ondelettes

On applique sur chaque canal la transformation par ondelettes rapide (*Fast Wavelet Transform*). Ces dernières années, l'expérimentation semble indiquer que deux types d'ondelettes sont optimales pour le traitement des images : Cohen-Daubechies-Feauveau et Daubechies symétrique (symmlet). Toutes ces ondelettes sont (presque) symétriques. Pour les besoins de ce projet, les ondelettes choisies par le FBI, les ondelettes symétriques de Daubechies, seront adoptées parce que le contenu en information n'est jamais déplacé dans les blocs de coefficients d'ondelettes pendant le codage (voir figure 5).

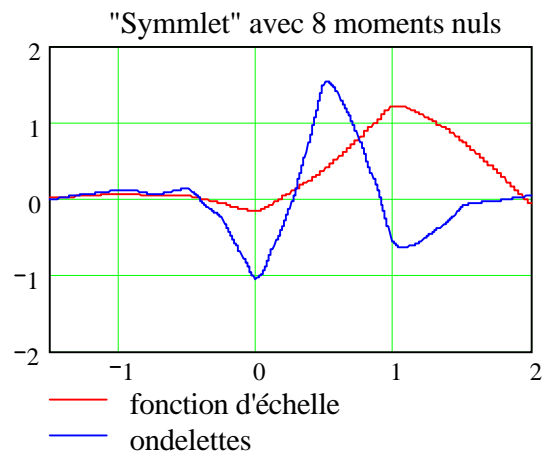


Figure 5. Ondelettes choisies

C. Zerotree

Les coefficients d'ondelettes ne sont plus représentés par des octets (valeurs de 0 à 255), contrairement au codage original de l'image par canaux, mais, dans notre cas, par des nombres à virgule flottante. Il n'y a donc encore aucune compression, bien au contraire, puisque les nombres à virgule flottante occupent plus d'espace en mémoire. Cependant, sans entrer dans les détails, il est permis dans cette nouvelle représentation de mettre un grand nombre de coefficients à zéro (en appliquant un seuil selon une certaine tolérance) sans beaucoup changer l'image. On peut ensuite noter de façon économique la position de ces zéros à l'aide d'un *zerotree* qui profite essentiellement du fait que si, dans une certaine région de l'image, il n'y a aucune *discontinuité*, tous les coefficients d'ondelettes correspondant seront nuls (ou presque). Cet ensemble de coefficients forme ce que l'on appelle un arbre de zéros.

D. Quantization

Par la suite, avant d'enregistrer cette représentation compressée, on doit convertir les coefficients à virgule flottante en des coefficients en nombre entier pour fins de stockage (*quantization*). Comme l'œil perçoit les détails selon une échelle logarithmique, la marge d'erreur est alors très mince pour l'approximation des coefficients de faible valeur, mais on pourra se permettre une marge d'erreur plus importante pour les coefficients de grandes valeurs. Cette conversion en nombres entiers peut être faite selon un mode *multi-canaux* ou canal par canal.

E. Codage d'entropie

Une fois la conversion en nombres entiers terminée, il faut procéder à un codage de l'entropie. En effet, tout comme le code Morse profite du fait que la lettre e est plus souvent utilisée en anglais et la représente de façon plus économique que les autres (par un seul trait), on peut compresser considérablement l'image en utilisant des techniques probabilistes, comme le *Huffman coding* ou LZ qui demeurent les techniques de compression les plus rapides pour les données brutes. De manière à aider davantage le codage d'entropie, on utilise un nombre de bits différent selon l'échelle en accordant plus d'importance (de bits) aux coefficients correspondants aux grandes échelles et moins aux petites échelles.

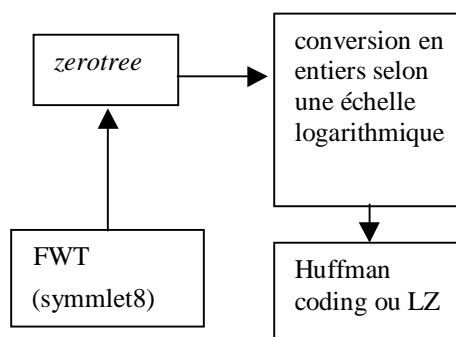


Figure 6. Schéma de compression proposé

III. RÉSULTATS

Nous avons testé l'ensemble de cet algorithme à l'aide d'un prototype écrit en Java et puis ensuite avec une version écrite en C/C++ (Microsoft Visual C++).

La version C/C++ comprend deux DLL qui vont respectivement du format RGB au format CIRA COD (compressé par ondelettes) et du format CIRA COD au format RGB. À partir de ces DLL, il est possible d'écrire des programmes compressant pratiquement tous les formats d'images possible. À titre d'illustration, des exécutables compressant des fichiers JPEG ont été utilisés sur diverses images JPEG.

Au niveau des temps de compression et décompression, le prototype Java compressait et décompressait une image 512 par 512 pixels en environ 4 s. La version C/C++ fait le même travail en environ 1 s. Une réécriture de la version Java avec les nouvelles techniques va permettre d'approcher les temps obtenus avec la version C/C++.

Les résultats qualitatifs sont présentés dans les figures 7, 8 et 9. On constate qu'en plus d'offrir un taux de compression environ 6 fois supérieur à JPEG, le format CIRA COD permet de plus d'obtenir une qualité d'image qui est exceptionnelle.



Figure 7. Image originale (261 Ko)

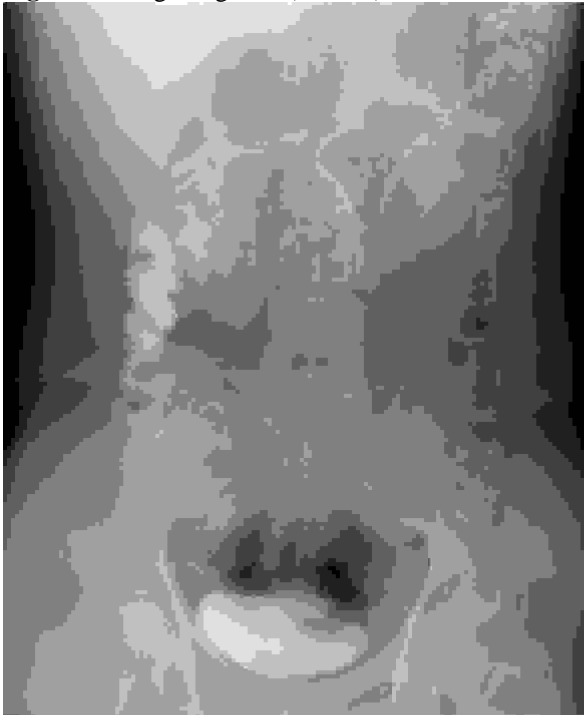


Figure 8. Image compressée JPEG (28 Ko)



Figure 9. Image compressée par CIRA (5 Ko)

IV. OPTIONS

Ce programme permet par la suite l'adjonction d'options, notamment :

1. ajout de filtres spéciaux sur les coefficients d'ondelettes, notamment pour rendre la lecture des mammographies plus facile;
2. possibilité de compléter l'image en fournissant l'information manquante;
3. affichage progressif de l'image;
4. affichage d'une section donnée de l'image;
5. Segmentation rapide des images à partir du format COD (détection des muscles, os, etc.);
6. interface avec le XML pour le contrôle des points précédents, soit de 2 à 5, à partir d'un navigateur;
7. la mise à niveau selon les normes émergentes (norme JPEG2000, etc.);
8. mise à jour de l'algorithme selon les progrès dans la théorie du codage d'entropie;

9. portage vers d'autres plates-formes (PowerPC, RISC, Java, puces DSP, etc.).

© Société de Compression d'Images en Réseaux et
Automatismes (CIRA), 1999
Reproduction autorisée pour la presse avec mention
des auteurs et des coordonnées de la société et
fourniture d'un justificatif.

Société de Compression d'Images en Réseaux et
Automatismes (CIRA)
30, allée du Vert Galant - 78240 CHAMBOURCY
Tél 01 39 30 42 00
Fax 01 39 30 42 09