# FAQ SUR LES ONDELETTES

D. LEMIRE, G. PAU

RÉSUMÉ. Ce document vise à répondre aux questions posées le plus fréquemment sur les ondelettes et les technologies similaires. Il est le compagnon idéal du forum sur les ondelettes (http ://www.ondelette.com/). Il n'est pas destiné à être un document définif, mais plutôt un document en constante révision.

## TABLE DES MATIÈRES

## 1. Au sujet de cette FAQ

**1.1. Où est-ce que je peux trouver la plus récente version de cette FAQ ?.** Le forum sur les ondelettes http ://www.ondelette.com/ contient un lien vers la plus récente version.

**1.2. Est-ce que je peux reproduire cette FAQ sur mon site ?** Absolument. La FAQ est sous GPL.

1.3. **J'ai trouvé une erreur ou une imprécision dans la FAQ, ou bien alors, j'ai une question que j'aimerais voir dans la FAQ, que dois-je faire ?** Il suffit de soumettre votre commentaire sur le forum sur les ondelettes.

1.4. **Si j'utilise cette FAQ pour un travail, un article ou un rapport, est-ce que je peux citer la FAQ ?.** Absolument. Vous pouvez citer la FAQ de cette manière : « D. Lemire et G. Pau, FAQ sur les ondelettes, http ://ondelette.com. ».

1.5. **Est-ce qu'il existe une version PDF de ce document ?** Oui.Une version PDF de la FAQ est disponible à l'adresse
http ://www.ondelette.com/forum/FAQ/French/faqondelette.pdf.

## 2. QU'EST-CE QUE LES ONDELETTES ?

2.1. **Qu'est-ce qu'une ondelette ?** Malheureusement, il n'y a pas de réponse définitive à cette question. Tout en sachant que nous n'arriverons pas à satisfaire tout le monde, essayons tout de même. Il est à noter que cette définition est rarement utilisée parce que d'une part, on ne traite pas toujours avec des fonctions et d'autre part, on ne travaille pas toujours avec des espaces de Hilbert.

Condition 1. Une base d'ondelettes doit inclure des bases de Riesz pour une famille de sous-espaces de Hilbert $W_j$ d'un espace de Hilbert $H$, on note ces bases de Riesz (3.3) $(\psi_{j,i})_i \in W_j$ et on doit pouvoir écrire $H = ... \oplus W_{-1} \oplus W_0 \oplus W_1 \oplus ....$ Par la condition de Riesz, nous avons $A_j \sum_i |c_i|^2 \leq \left\| \sum_i c_i \psi_{j,i} \right\|^2 \leq B_j \sum_i |c_i|^2$. On dira que la base est stable si $A_j$ et $B_j$ sont constants (indépendant de $j$).

Condition 2. Toutes les ondelettes doivent avoir une masse nulle $\int \psi_{j,i}(x)dx = 0$ (voir 3.11).

Condition 3. Il doit y avoir une condition d'échelle, c'est à dire qu'il existe $b > 1$ tel que $f \in W_j$ implique $f(b\cdot) \in W_{j-1}$. On choisit normalement $a = 2$ (cas dyadique). En d'autres terms, dans le cas dyadique, on exige que si $f(x) \in W_j$ alors $f(2x) \in W_{j-1}$.

La condition 1 peut être réduite à exiger que les ondelettes forment un repère (3.5).

2.2. **Qu'est-ce qu'une ondelette discrète ?** La définition est la même que pour les fonctions exception faite de la propriété d'échelle. Prenons pour exemple $\mathbb{R}^n$, alors les ondelettes doivent être des vecteurs $\psi_{j,i}$ formant des bases de Riesz et ayant une masse nulle. Pour des raisons pratiques, on requiert que les ondelettes soient locales.

Condition 1. Une base d'ondelette doit inclure des bases de Riesz pour une famille de sous-espaces $W_j$ dans $\mathbb{R}^n$, on note ces bases de Riesz (3.3) $(\psi_{j,i})_i \in W_j$ et on doit pouvoir écrire $\mathbb{R}^n = V_0 \oplus W_0 \oplus W_1 \oplus ... \oplus W_K$ où $K$ est le nombre de sous-espaces. Par la condition de Riesz, nous avons que $A_j \sum_i |c_i|^2 \leq \left\| \sum_i c_i \psi_{j,i} \right\|^2 \leq B_j \sum_i |c_i|^2$. Le sous-espace restant $V_0$ ne peut pas être représenté en utilisant les vecteurs $\psi_{j,i}$ et est parfois appelé le sous-espace d'échelle.

Condition 2. Toutes les ondelettes doivent avoir une masse nulle $\langle \psi_{j,i}, 1 \rangle = 0$ .

Condition 3. Les ondelettes doivent être locales. Définissons la taille du support d'un vecteur $v$ comment étant le plus petit entier $N$ tel qu'il existe $K$ avec la propriété que $v_{K+k} = 0$ ou est hors plage pour tous les $|k| > N$. Il doit exister $N$ et un entier $b > 1$ telle que le taille du support de $\psi_{j,i}$ est plus petite ou égale à $b^j N$ et tel que $b^K N \leq n$.

On peut réduire le première condition en remplaçant « base de Riesz » par « repère » (3.5).

2.3. **Quel est le lien entre les ondelettes discrètes et les ondelettes continues ?** Cette FAQ se préoccupe surtout des ondelettes discrètes parce que ce sont les ondelettes que l'on retrouve le plus dans les applications. Les ondelettes continues sont à distinguer des ondelettes discrètes. Il est faux de croire qu'étant donné une ondelette continue, on peut construire l'ondelettes discrète correspondante. On peut donc considérer qu'il s'agit de deux théories mathématiques différentes. Alors que les ondelettes continues sont fortement influencées par l'analyse mathématique classique (Morlet, Meyer), les ondelettes discrètes relèvent davantage de la théorie de l'approximation et des schémas de subdivision (Dubuc, Deslauriers, Daubechies).

2.4. **Qu'elles sont les ondelettes disponibles ?** On s'en doute bien, il y en a un nombre infini. Il faut aussi savoir qu'on peut construire ses ondelettes soit en utilisant des schémas de subdivision, soit par relèvement.

2.5. **Comment puis-je trouver la forme analytique d'une ondelette ?** En général, les ondelettes à support compact n'ont pas de forme analytique. En particulier, les ondelettes de Daubechies n'ont pas de forme analytique : on sait comment calculer la fonction, mais on ne peut pas l'exprimer avec une formule.

2.6. **Comment traduit-on « frame » ?** Il semblerait que la traduction appropriée soit "repère" (voir 3.5).

2.7. **Quelle est la meilleure ondelette ?** Il n'y a pas une ondelette qui soit meilleure que les autres, malheureusement. Tout dépend de votre application en fait. Dans certains cas, l'ondelette la plus simple (Haar) sera optimale. Pour d'autres applications, ce sera le pire des choix possibles. Il arrive souvent aussi en pratique que le problème de la meilleure ondelette soit un problème mal posé et sans réponse. En pratique, il semblerait que l'élément le plus important soit le nombre de moments nuls 3.11. Pour la plupart des applications, il est désirable d'avoir le plus de coefficients d'ondelettes nuls et donc plus de moments nuls implique une meilleure transformation. Cependant, les ondelettes ayant un plus grand nombre de moments ont aussi un support plus grand ce qui signifie que si votre fonction ou signal a des discontinuités brusques, elle sera plus sujettes aux phénomènes de Gibbs. En bref, si la fonction à analyser était analytique, il serait idéal d'avoir un trèès grand nombre de moments nuls, mais comme la plupart des fonctions ne sont pas analytiques, un compromis est désirable. C'est particulièrement vrai dans les applications industrielles. En imagerie, par exemple, on opte souvent pour des ondelettes ayant de 8 à 12 moments nuls.

2.8. **Est-ce que les ondelettes sont meilleures que la transformation de Fourier ?** Non, en fait, les ondelettes ont une résolution en fréquence qui est inférieure à la transformée de Fourier. Ainsi donc, si on veut caractériser la contenu en fréquence d'un signal (par exemple, trouver la fréquence exacte d'une note), la transformée par ondelette n'est pas appropriée. Une alternative est d'utiliser un algorithme plus coûteux comme le Matching Pursuit.

2.9. **Comment est-ce que les ondelettes (FWT, DWT) se comparent à la transformée de Fourier par fenêtres (Short Time Fourier Transform, STFT) ?.** Avec la STFT, on doit imposer une taille de fenêtre le plus souvent arbitrairement ce qui revient à fixer la résolution en temps et fréquence de façon unique et définitive. JPEG, par exemple, utilise des blocs de $8 \times 8$. Avec les ondelettes, on utilise un gamme variée de résolutions ce qui permet un traitement plus robuste et efficace dans de nombreuses applications. L'avantage est d'autant plus important avec les signaux très longs. En même temps, pour des signaux très longs, la transformée par ondelette risque d'être plus coûteuse en temps de calcul et la gestion de la mémoire peut être plus délicate avec certaines applications.

2.10. **Je veux faire de l'analyse fréquencielle par ondelettes, comment faire ?** La fréquence est un concept qui se définit par la transformée de Fourier. Les ondelettes fournissent une carte échelle-temps et non pas une représentation fréquentielle du signal.

2.11. **Concernant le problème de la régularité des ondelettes, est-ce que les splines ne sont pas la solution idéale ?** Les b-splines permettent d'obtenir une régularité optimale pour une certaine longueur de filtre (ou taille de support compact). Elles sont donc des ondelettes privilégiées lorsqu'il la régularité est importante. Cependant, les b-splines ne sont pas orthogonales.

## 3. Un peu de théorie solide

3.1. **Qu'est-ce que les ondelettes orthogonales, biorthogonales ?** Une transformation est orthogonale si $\langle T(x), T(y)\rangle = \langle x,y\rangle$. Une paire $(T,T')$ est biorthogonale si $\langle T'(x), T(y)\rangle = \langle x,y\rangle$, ce qui implique que $T^-1 = T'^T$.

3.2. **Pourquoi utiliser les ondelettes biorthogonales ?** Alors que les ondelettes orthogonales ont autant de moments nuls à l'analyse qu'à la synthèse, ainsi que la même régurité à l'analyse comme à la synthèse, les ondelettes biorthogonales permettent plus de liberté.

3.3. **Qu'est-ce qu'une base de Riez ?** Une base de Riesz $\{\psi_k\}_k$ est une famille de fonctions dans un espace de Hilbert telle que $A\sum_i |c_i|^2 \leq \|\sum_i c_i\psi_i\|^2 \leq B\sum_i |c_i|^2$ with $0 < A \leq B < \infty$ peu importe les $c_i$ choisis. Les ondelettes orthogonales et biorthogonales forment des bases de Riez. La grande majorité des ondelettes sont dans cette catégorie. Dans le cas d'une base orthonormale, nous avons $A = B = 1$ et $\|\sum_i c_i\psi_i\|^2 = \sum_i |c_i|^2$.

3.4. **Est-ce qu'il y a une autre définition pour les bases de Riesz ? (contribuée par F. Lenardon).** Rappelons qu'un espace de Hilbert $H$ est séparable s'il existe une suite de fonction $S = \{\psi_k\}_k$ telle que $\overline{span(S)} = H$. Soit $H$ un espace séparable de Hilbert. Alors les deux affirmations suivantes sont équivalentes :

(1) La séquence $\{\psi_k\}_k$ forme une base de Riesz pour H.
(2) Il existe un produit scalaire équivalent sur $H$, par rapport auquel la séquence $\{\psi_k\}_k$ devient une base orthogonale pour $H$.

En d'autres termes, une base de Riesz est une base orthormale à un change de produit scalaire près. Cela explique le principe derrière les ondelettes de Battle-Lemariée : comme les B-splines forment des bases de Riesz, il est possible de les orthonormaliser.

3.5. **Qu'est-ce qu'un repère ?** Un repère $\{\psi_k\}_k$ est une famille de fonctions dans un espace de Hilbert telle que $A\|f\|^2 \leq \sum_k |\langle f,\psi_k\rangle|^2 \leq B\|f\|^2$ with $0 < A \leq B < \infty$ peu importe la fonction $f$ choisie. Il est généralement accepté que les ondelettes sont nécessairement des repères.

3.6. **Qu'est-ce qu'un repère étroit (tight frame) ?** Lorsque nous avons un repère avec $A = B$, on dit que le repère est étroit. Dans le cas $A = B = 1$ et $\|\psi_k\| = 1\,\forall k$, on peut montrer que l'on a une base orthonormale. Ce qui est intéressant avec les repères étroits, c'est que l'on une correspondance précise des normes. Les bases d'ondelettes non décimées ou invariantes par translation sont souvent utilisées pour construire des repères étroits.

3.7. **Est-ce qu'un repère est une base de Riesz ?** Non, un repère n'est pas nécessairement une base de Riesz : par exemple les vecteurs $(2,1),(0,-1),(0,1)$ forment un repère dans le plan, mais pas une base de Riesz.

3.8. **Est-ce qu'un repère étroit est une base de Riesz ?** Non, un repère étroit ne forme pas nécessairement une base de Riesz. C'est toujours le cas si $A = B \neq 1$. Les vecteurs $(1)$ et $(-1)$ forme un repère (étroit) dans $\mathbb{R}$, mais il est facile de vérifier que ce n'est pas une base de Riesz. Par contre, si $A = B = 1$ et que les éléments du repère ont une norme de 1, alors il s'agit bien d'une base de Riesz, et en fait, d'une base orthonormée.

3.9. **Est-ce qu'une base de Riesz est un repère ?** Oui. On définit parfois les bases de Riesz comme étant des repères où tous les éléments sont linéairement indépendants.

3.10. **Quel est l'importance d'avoir un repère ?** Il y a deux raisons. La première est qu'un repère est toujours doté d'un repère dual c'est-à-dire qu'étant donné un repère $\psi_k$, il existe un repère dual $\widetilde{\psi_k}$ tel que $f = \sum \langle f, \psi_k \rangle \widetilde{\psi_k}$ ou $f = \sum \left\langle f, \widetilde{\psi_k} \right\rangle \psi_k$ selon le cas. La deuxième raison est que la propriété qui définit un repère nous permet d'affirmer que si deux signaux sont proches, alors leur transformation $(\langle f, \psi_k \rangle)$ sera proche, et aussi que si deux transformations sont proches, alors les signaux doivent être proches. Cette dernière propriété est essentielle parce qu'elle signifie que tant que l'on ne modifie que légèrement les coefficients d'ondelette, le signal équivalent ne sera que légèrement modifié (stabilité algorithmique).

3.11. **Qu'est-ce que le nombre de moments nuls et quel lien fait-on avec la régularité ?** On dit qu'une ondelette a $N$ moments nuls si pour $i = 0, \ldots, N-1$, $\int x^i \psi_{j,i}(x)dx = 0$. En particulier, toute ondelette se doit d'avoir au moins un moment nul.

Brièvement, on peut expliquer l'utilité pratique d'avoir un bon nombre de moments nuls, supposons qu'on prend la transformée par ondelettes de $f$ avec des ondelettes ayant un support compact et $N$ moments nuls, localement, nous supposons que $f$ a l'expansion de Taylor

$$f(x) = f(a) + f'(a)(x-a) + \ldots + f^{(N-1)}(a)(x-a)^{N-1}/(N-1)! + erreur(x)$$

Nous avons alors

$$\int f(x)\psi_{j,i}(x)dx = \int erreur(x)\psi_{j,i}(x)dx$$

où l'intégrale n'est prise que dans la voisinage de $f$ sujet à l'expansion de Taylor. On suppose ici implicitement que le support de l'ondelette est comprise dans ce voisinage, mais pour une fonction partout régulière, on peut faire l'expansion de Taylor pour n'importe quel $a$ et on peut donc centrer l'expansion dans le support de l'ondelette. Comme on peut le voir, un grand nombre de moments nuls nous permet d'affirmer que pour une fonction très régulière, les coefficients d'ondelettes seront petits. Ce détail est important pour les applications. Comme la fonction d'erreur prend localement la forme $f^{(N)}(\xi)(x-a)^N/N!$ (théorème de Taylor), on voit que plus les ondelettes auront un petit support, plus les coefficients d'ondelette seront petits. En prenant donc le ratio des coefficients d'ondelettes correspondant à des échelles différentes, on peut donc espérer mesurer la régularité de $f$.

En résumé, pour une fonction $f$ très régulière, les coefficients d'ondelette seront petits, et cela est d'autant plus vrai pour les ondelettes très localisées (échelles fines).

3.12. **Pourquoi la transformée de Fourier $F(w)$ d'une fonction $L^1$, $f(x)$, tend vers zéro quand $w$ devient grand.** Par définition, $F(w) = \int f(x)e^{-ixw}dx$. De cette équation, nous savons que $|F(w)|$ satisfait $|F(w)| \leq \int |f(x)|dx < \infty|$. Si on teste la continuité, on obtient tout d'abord

$$F(w+\varepsilon) = \int f(x)e^{-ixw}\left(1-e^{-ix\varepsilon}\right)dx.$$

Puisque $\left(1-e^{-ix\varepsilon}\right)$ sera petit pour $x$ borné, et que $\left|1-e^{-ix\varepsilon}\right| \leq 2$, il suffit donc d'avoir que $\int_N^\infty |f(x)|dx \to 0$ lorsque $N$ devient grand, mais ce résultat est certainement vrai puisque $f$ est Lebesgue...

On se résume : $F$ est bornée et continue... reste à montrer que $F(w)$ décroît lorsque $w$ augmente...ce théorème porte un nom... Riemann-Lebesgue...

Preuve du théorème (en gros) :

On sait que les fonctions de Lebesgue peuvent être générées (i.e. c'est un ensemble "dense") par les fonctions "caractéristiques", c'est-à-dire les fonctions $f_X(x) = 1$ si $x \in X$ et 0 sinon. On peut voir que la transformée de Fourier de $f_X(x)$, $\hat{f}_X(w)$, ira en $1/w$ (pour s'en convaincre, choisir $X = [0,1]$). Ensuite, il suffit de vérifier que l'ensemble des fonctions $f$ dans $L^1$ telles que $\hat{f}$ tend vers zéro à l'infini est fermé dans la norme $L^1$. Mais c'est pas bien compliqué puisque les $\hat{f}$ sont bornées par la norme $L^1$. En effet, soit $f$ une fonction telle qu'elle peut être approchée aussi près qu'on veut par des fonctions $g$ telles que leur transformée de Fourier tend vers zéro... En norme $L^1$, $|f-g|_{L^1} \to 0$, donc $|\hat{f}-\hat{g}|_{L^\infty} \to 0$. Fin de la preuve !

3.13. **Comment calculer la fréquence instantanée d'un signal non stationaire avec les ondelettes ?** La "fréquence" est un concept mal défini en dehors de Fourier, et c'est là le véritable problème. Bien qu'il puisse sembler évident qu'étant donné un signal, il doit y avoir une fréquence associée à chaque instant dans le temps, ce n'est pas vrai en pratique. Pour s'en convaincre, il suffit de considérer $f(t) = \sin t + \sin t^2$ : quelle est la fréquence de cette fonction $f(t)$ au temps $t = t_0$ ?

La vraie solution est de rechercher une représentation temps-fréquence du signal. Cela peut-être accompli avec le Matching Pursuit, les atomes de Gabor, etc. Ce ne sont plus des ondelettes cependant. Bien que les transformations demeurent linéaires, elles ne sont plus "orthogonales"... en d'autre termes, elles comportent beaucoup plus de calculs que Fourier et sont beaucoup, beaucoup plus lentes, même si leur complexité peut (parfois) être similaire.

3.14. **Est-ce qu'il existe des ondelettes respectant la causalité ?** Un filtre est causal s'il peut s'écrire $y_i = f(x_i, x_{i-1}, \ldots)$ de telle manière à ce que la valeur $y_i$ ne dépend que du passé et non de l'avenir. Ce concept n'a malheureusement du sens que pour les transformations ou filtrages dans le temps. La transformation par ondelettes étant une transformation temps-fréquence, le concept n'a plus de sens. Par exemple, il est facile de vérifier que la transformation de Haar ne correspond pas à aucun concept de causalité, par contre, on peut écrire les filtres d'échelle et d'ondelettes avec les formules $y_i = (x_i + x_{i-1})/\sqrt{2}$ et $z_i = (x_i - x_{i-1})/\sqrt{2}$ ce qui *semble* être causal. L'exemple de Haar nous ramène au problème fondamental que les coefficients d'ondelettes ont un indice de temps et un indice d'échelle, alors que la causalité n'est définie qu'entre deux séries ayant une dimension commune (le temps). Étant donné $x = (0,1)$ ou $x = (1,0)$, la transformation par ondelettes de Haar me donne $1/\sqrt{2}$ comme coefficient d'échelle et $\pm 1/\sqrt{2}$ comme coefficient d'ondelettes... est-ce causal ? La question n'a pas de sens.

## 4. LOGICIEL

4.1. **J'utilise Matlab et je veux faire des ondelettes sans payer ?** David Donoho et ses collaborateurs rendent disponible WaveLab. C'est gratuit.

4.2. **Où est-ce que je peux trouver du code Java ?** Une implémentation en Java, facile à convertir en C/C++ est disponible avec une table de filtres, et il y a aussi une classe java toute prête. On doit aussi mentionner une librairie scientifique puissante en Java JSci.

4.3. **Où est-ce que je peux trouver du code C/C++ ?** Il y a une implémentation de l'ondelette Symmlet8 en C++.

4.4. **Je veux faire de la compression d'images ou de la compression vidéo par ondelettes ! Veuillez m'envoyer du code !** Il faut savoir que ce type d'applications est sujets à des brevets et autres contraintes commerciales. Il n'est pas clair pour l'instant qu'il existe une librairie gratuite pouvant être utilisée en toute liberté. Il y a un white paper générique sur la compression d'images par ondelettes écrit pour la société CIRA.

En général, il faut avoir un solide budget de développement pour entreprendre un projet en compression d'images ou en compression vidéo surtout si l'on compte sur les applications commerciales.

4.5. **Qu'est-ce que JPEG2000 ?** Il s'agit d'une norme ISO établie par un ensemble d'industriels et de centres de recherche pour la compression d'images. La norme doit remplacer JPEG à terme. Cependant, comme j'écris ces lignes en 2002 et que JPEG2000 demeure encore virtuellement inutilisé par le grand public, on peut s'attendre à ce que le bon vieux JPEG (basé sur la transformée de Fourier) ait la vie dure !

Voici quelques liens : une page nous venant du Japon par Okumura (http ://www.matsusaka-u.ac.jp/ okumura/compression.html), http ://DataCompression.info/JPEG2000.shtml, http ://maestro.ee.unsw.edu.au/ taubman/kakadu/.

Il existe une librairie Java disponible facilement appelée JJ2000 (http ://jj2000.epfl.ch/). On peut télécharger la librairie gratuitement, mais je n'ai pas pu vérifier les droits. Il est probable que vous puissiez utiliser cette librairie à des fins académiques. De la même façon, il y a une librairie open source en C/C++ J2000(http ://www.j2000.org).

Est-ce que vous devriez utiliser JPEG 2000 ? Voici un article en anglais sur la question (http ://www.advogato.org/article/327.html). En terminant, voici un avertissement du président du comité ISO/JPEG2000sur l'aspect légal de l'utilisation de la norme JPEG2000 :

> *Particular attention is drawn to the use within the standard of selected technology or algorithms which are claimed to be protected by national and/or international patents. In the case of technology submitted for consideration by JPEG for incorporation in Part 1 of the standard, the JPEG committee believes that the individual organisations concerned will make available licences to use this intellectual property, on a royalty- and fee-free basis, under specified conditions which may apply only to conforming implementations of the standard. These conditions are available on application to the organisations concerned, which are listed in an Annex to the document.*
>
> *It is of course still possible that other organisations or individuals may claim intellectual property rights that affect implementation of the standard, and any implementors are urged to carry out their own searches and investigations in this area. The JPEG committee requests any organisations or individuals claiming (or being aware of claims) that any of*

*the committee drafts available for download here infringes other intellectual property rights to provide information and/or evidence to substantiate their claim to the JPEG Convener in the first instance.*

4.6. **Je veux faire du développement MPEG, où est-ce que je peux trouver le code ?** Les normes MPEG sont brevetés. Il faut donc normalement une entente légale avec les détenteurs des brevets pour leur utilisation. Par exemple, pour MPEG-4, les coûts d'utilisation sont determinés en partie par le temps d'utilisation du CODEC selon une facturation à la minute. Pratiquement toutes les grandes boîtes faisant de le multimédia ont des implémentations MPEG incluant, par exemple, Texas Instrument.

## 5. Applications

5.1. **Quelles sont les principales applications des ondelettes ?** On utilise surtout les ondelettes pour la compression d'images et pour le débruitage non paramétrique des signaux. Il faut noter cependant que les ondelettes sont aussi utilisées en analyse numériques pour simuler des phénomènes physiques et récemment, les ondelettes ont été utilisées avec succès pour l'accélération des bases de données.

5.2. **J'ai bien compris comment faire la transformation par ondelettes sur un signal 1D, comment faire pour transformer une image ?** La méthode la plus simple consiste à transformer les colonnes puis les rangées en faisant une transformation unidimensionnelle. Il existe cependant des ondelettes non séparables qui ne peuvent utilisé de cette façon, mais elles sont peu utilisées en pratique.

5.3. **La transformée par ondelettes ne s'applique que pour des matrices de taille $2^p$, donc pour des images de $2^p$ pixels fois $2^p$ pixels ? Alors comment faire pour des images non carrées et/ou de coté différent d'une puissance de 2 (par exemple 800*600 ou 1024*768) .** Ce n'est pas vrai que l'algorithme ne s'applique qu'aux matrices de taille $2^p$. Chaque fois que vous appliquez la transformée par ondelettes (une itération), vous divisez par deux la taille de l'image. Cependant, cela ne signifie pas que vous deviez (ou pouviez) poursuivre jusqu'à ce que vous ayiez des matrices 2 par 2. Vous pouvez très bien faire...$800\times600$ -> $400\times300$ -> $200\times150$ -> $100\times75$. Autrement, si vous devez absolument poursuivre au delà de $100\times75$, vous pouvez appliquer du remplissage pour compléter la dernière matrice à $100\times76$ par exemple.

5.4. **Qu'est-ce que le débruitage par ondelettes ?** Les ondelettes permettent un débruitage statistique non paramétrique des signaux. Un des chercheurs les plus réputés dans le domaine est David Donoho. Le principe est assez simple : on applique la transformation par ondelettes, on applique ensuite un seuillage sur les coefficients pour ne garder que les coefficients les plus significatifs. En effet, comme le bruit tend à se répartir sur tous les coefficients d'ondelettes, on sait que le ratio signal/bruit sera plus grand sur les coefficients les plus importants en amplitude et en ne gardant que ceux là, on augmente généralement le ratio signal/bruit de notre signal. (Voir les articles de Donoho et al. pour les détails.)

On peut aussi, dans une certaine mesure, faire le travail inverse et retrouver de l'information perdue par un filtre passe-bas en utilisant les ondelettes à condition d'avoir un bon modèle. Il s'agit simplement d'amplifier certains coefficients d'ondelettes (les plus significatifs) au lieu de faire un seuillage.

5.5. **L' algorithme par ondelettes prend une image $M \times N$ et va en faire un tableau de $M \times N$ coefficients, où est la compression ? Première réponse.** La transformation par ondelettes ne compresse absolument rien. La compression par ondelettes est une transformation 1-1 et pire, en pratique, c'est plutôt 4-1, car on traite des pixels 8bits en utilisant un stockage 32 bits. La compression vient plus tard, en utilisant d'autres procédés de codage d'entropie.

**Deuxième réponse.** Nulle part, pour l'instant. Par contre, si la tranformation est bien faite et si l'ondelette est bien choisie, la variance du tableau résultant de la transformation va être très faible (en gros, il y aura beaucoup de zéros, et de valeurs proches dans le tableau de sortie). La variance étant directement liée à l'entropie d'un signal, la matrice, après discrétisation, pourra être codée efficacement par un codeur entropique.

5.6. **Qu'est-ce que le codage entropique ou codage d'entropie ?** L'entropie est une mesure statistique de la quantité d'informations contenue dans un signal, introduite par Shannon. En présence d'un signal connu qui prend ses valeurs dans un ensemble fini de symboles, on peut calculer aisément cette mesure, qui s'exprime en bits/symbole.

Le codage entropique consiste à coder ces symboles, sur un nombre de bits variable, en fonction de la fréquence d'apparition de ces symboles : "plus le symbole est fréquent, plus son code sera court" : Il existe plusieurs algorithmes réalisant ce codage entropique : Huffman, Shannon-Fano, le codage artithmétique...

On montre que n'importe quel algorithme de codage entropique appliqué à un signal produira un nombre de bits moyens toujours supérieur à l'entropie de ce signal. Le codage arithmétique est un algorithme performant qui atteint asymptotiquement l'entropie (mais il est breveté par IBM).

On remarquera que la plupart des codeurs actuels (d'image, de son ou de fichiers) réalisent une transformation du signal originel, le discrétise si nécessaire et applique ensuite un codage entropique. En effet, la transformation est choisie en fonction du type de signal d'entrée et de telle manière à obtenir un signal intermédiaire qui possèdera une entropie faible.

Dans le cas de JPEG, la transformation en cosinus discret par blocs (DCT) et la quantification qui en suit, permet d'obtenir un grand nombre de zéros dans le signal intermédiaire (plus précisement, son histogramme présentera un pic très raide) qui possèdera ainsi une entropie très faible. Un codeur d'Huffman, integré à JPEG, permet alors un encodage efficace de l'image.

GZip encode à la volée les chaînes de caractères présente dans un flux de données par réference à une même chaîne précedemment observée. GZip encode alors un ensemble de couples (distance, taille de la chaîne) qui est alors encodé par deux codeurs d'Huffman. Ainsi, en présence d'un fichier présentant de nombreuses répétitions (c'est le cas des fichiers textes ou exécutables), l'entropie du signal de couples (distance, taille) sera très faible et GZip sera performant.

5.7. **L'image couleur résulte de la superposition de 3 matrices (1 codant chaque couleur primaire : rouge vert bleu). Applique-ton l'agorithme pour chaque matrice ? Première réponse.** On pourrait le faire mais en pratique, on ne le fait pas, car les résultats ne sont pas très bons (la quantité d'information compressée est triplée).

Une méthode très utilisée (dans MPEG, JPEG et JPEG-2000) consiste à utiliser une propriété de l'oeil : celui-ci perçoit de manière très sensible les variations d'intensité lumineuse mais est peu sensible aux variations de couleurs.

La méthode consiste ainsi à transformer l'espace de couleurs RGB en un autre espace nommé YUV, où Y represente la partie noire et blanc de l'image (la luminance, comme si l'image originale était convertie en "niveaux de gris"). U et V comporteront l'information de chrominance bleue et rouge, respectivement.

Cette transformation est linéaire, très simple, et permet d'obtenir à partir des trois matrices R, G et B, trois nouvelles matrices : Y, U et V. La matrice Y est alors codée comme une image en niveaux de gris classique, et les matrices U et V seront codées avec une résolution 4 fois plus faible (leur hauteur et leur largeur sont divisées par 2) ; l'oeil ne sera pas sensible à ce prétraitement destructif.

**Deuxième réponse.** Tout dépend en fait de votre choix algorithmique. Il y a des transformation scalaires et vectorielles. En effet, on peut traiter les 3 canaux comme un champ vectoriel, ou bien alors comme trois fonctions scalaires.

L'intuition nous dit que l'approche vectorielle devrait être supérieure, mais ce n'est pas évident. D'abord parce que l'on va recoder l'image, non pas en RGB, mais en YIQ ou un autre format. Dans ce nouveau format, il se trouve que l'on peut traitement chaque canal en fonction de son importance. Par exemple, on peut sous-échantillonner les canaux I et Q sans grande perte de qualité visuelle. Cette approche est courante et date des premiers jours de la télévision (couleur).

5.8. **En compression vidéo, on peut faire mieux que Fourier avec les ondelettes ?** Ce n'est pas nécessairement vrai. Il faut comprendre qu'en compression vidéo, le gros du gain provient de la compression entre les images successives. Bien que les ondelettes compressent très bien les images une à une, ce n'est pas là un gain important pour les spécialistes de la vidéo.

5.9. **Quelle est la différence entre la compression par ondelettes pour la vidéo (MPEG) et pour les images statiques (JPEG2000) ?** La compression vidéo de type MPEG compresse des images une à une tout comme la compression d'images traditionnelle (JPEG). Cependant, la compression d'images doit relever ses propres défis. En fait, de nombreux problèmes se posent en compression statique qui ne se posent que rarement en compression vidéo. D'abord, il est assez rare que l'on va exiger un compression (sans perte) en vidéo alors que c'est très courant en compression d'images statiques (pour des raisons légales surtout d'ailleurs). D'autre part, la compression d'images statiques doit souvent permettre le traitement d'images énormes (faisant 4000 par 4000 pixels par exemple) et l'accès dynamique par région (ROI - Region Of Interest). Il y a aussi parfois la nécessité d'avoir plus de 8 bits par canal, en particulier pour l'impression d'images (images en 8/10/12 bits par canal). Ainsi donc, l'algorithmie des images statiques doit pouvoir résoudre des problèmes difficiles que l'on ne rencontre pas en compression vidéo (et inversement).

5.10. **Où est-ce que je peux trouver les références concernant les formats MPEG ?.** MPEG est un consortium d'industriels et d'universitaires issu d'un comité joint de l'ISO et de l'IEC. Les normes MPEG sont la propriété de l'ISO et sont payantes. De surcroît, les algorithmes utilisés dans les normes MPEG sont souvent brevetés, et donc soumis à la propriété intellectuelle de leurs inventeurs.