

CODAGE HYBRIDE D'IMAGES NUMERIQUES -
HYBRID CODING OF DIGITAL IMAGES

D. THOREAU - J. PAILLE

LABORATOIRE DE TRAITEMENT DU SIGNAL NUMERIQUE ENSP D.U. St Jérôme 13397 MARSEILLE Cedex 13

RESUME

Dans cet article est présentée une technique d'encodage hybride par bloc, utilisant la transformée de Hadamard sur 16 points le long des lignes et le DPCM sur les colonnes, pour des images numériques de 512x512 pixels. Un code de Huffman a été employé pour coder les erreurs de quantification

L'étude principale a consisté à analyser l'influence, d'une part du nombre de composantes de Hadamard conservé, et d'autre part de la valeur du facteur de normalisation de ces composantes avant quantification sur les performances de cet algorithme d'encodage. Des résultats portant sur le taux de compression et le rapport signal à bruit sont présentés à partir de différents essais effectués sur une image numérique.

SUMMARY

This paper presents hybrid encoding technic of digital images using the Hadamard transform on one-dimensional arrays (1x16 elements) along the rows and, differential pulse code modulation (DPCM) in the vertical direction.

The quantization strategy consists in a normalization of the coefficients by the variance and the utilization of one and the same quantizer. Code words of unequal length are generated from the quantizer output levels by a Huffman code.

The main study was to analyse, on the one hand, the influence of the number of components used in the Hadamard spectrum and, on the other hand, the influence of the normalization factor of these components before the quantization, on the performances of this encoding algorithm.

Experiments were performed on 512x512 pixels image, and results about compression ratio and signal to noise ratio are presented.



1 - ENCODAGE HYBRIDE

L'encodage hybride, principalement dû aux travaux de HABIBI [1], [2], et WINTZ [3], permet d'allier le concept d'encodage par transformation à une dimension et le système DPCM. Un tel procédé assure essentiellement les mêmes performances que l'encodage par transformation à deux dimensions, mais est plus facile à mettre en oeuvre. Cet encodage hybride utilise la transformation de Hadamard dans la direction horizontale et le DPCM dans la direction verticale.

Le but de la transformation est de réduire la corrélation dans la direction horizontale. Cependant, comme les points suivant la direction verticale sont généralement corrélés, nous pouvons raisonnablement penser que les coefficients de même indice produits par la transformation de Hadamard seront également corrélés. Cette corrélation entre les coefficients dans le sens vertical peut être diminuée en utilisant un codage DPCM sur les colonnes de coefficients de la transformée. Puisque les coefficients issus de la transformée ont en général des variances différentes, nous allons utiliser une méthode de quantification par blocs. Afin d'obtenir un taux de compression encore meilleur, le code utilisé dans cette méthode sera un code de Huffman [4], qui est le code optimal de la distribution des niveaux de sortie du DPCM. Le but de cette étude est d'analyser d'une part l'influence du nombre de composantes de hadamard conservées, et d'autre part la technique de quantification sur les performances de cet algorithme.

2 - CODAGE PAR TRANSFORMATION DE HADAMARD

Chaque ligne de l'image 512x512 points est décomposée en sous-blocs de 1x16 éléments dans le sens horizontal, formant ainsi une image composée de 32 bandes de 512x16 pixels dans le sens vertical. Ensuite chaque bloc est transformé à l'aide de la transformation de Hadamard suivant la relation (1). Le but de cette transformation est de réduire la corrélation entre pixels, et cela dans l'objectif d'améliorer l'efficacité de l'encodage en traitant les coefficients transformés indépendamment les uns des autres.

$$H(u) = \frac{1}{N} \sum_{x=0}^{15} f(x) (-1)^{\sum_{i=0}^x b_i(x) p_i(u)} \quad (1)$$

La transformée inverse étant donnée par la relation suivante :

$$f(x) = \sum_{u=0}^{15} H(u) (-1)^{\sum_{i=0}^u b_i(x) p_i(u)} \quad (2)$$

Les coefficients de la transformée sont rangés dans l'ordre croissant du nombre de changements de signe du noyau de calcul de la transformée de Hadamard c'est à dire un ordre équivalent à celui des fréquences dans le plan de Fourier [5].

3 - QUANTIFICATION DES COEFFICIENTS

Chaque composante $H(u)$ issue de la transformation de Hadamard est une combinaison linéaire des 16 pixels du sous réseau d'entrée, et, bien que les pixels d'entrée puissent s'étendre seulement sur 64 niveaux de gris, les valeurs de sortie peuvent varier de 1024 à -1024. Il serait inefficace, en conséquence, d'utiliser le même quantificateur pour tous les coefficients. En d'autres termes, si les niveaux de sortie du quantificateur sont ajustés pour conserver la plage des coefficients ayant la plus grande variance, alors les coefficients de variances les plus petites tombent dans la plage la plus faible, avec pour résultat que la plupart des niveaux du quantificateur ne seront pas utilisés. Pour remédier à un tel problème, une technique que nous allons utiliser consiste à normaliser chaque coefficient de la transformée par rapport à l'écart-type de la colonne de coefficients, de telle sorte que dans le plan intermédiaire d'une bande de l'image nous ayons :

$$G(u,y) = \begin{cases} H(u,y)/\sigma_u & \text{si } \sigma_u > 1 \\ H(u,y) & \text{si } \sigma_u \leq 1 \end{cases} \quad (3)$$

où $u = 0, 1, 2, \dots, 15$
et $y = 0, 1, 2, \dots, 511$
avec σ_u tel que :

$$\sigma_u^2 = \frac{1}{512} \sum_{y=0}^{511} (H(u,y) - \bar{H}_u)^2 / 512 \quad (4)$$

$$\bar{H}_u = \frac{1}{512} \sum_{y=0}^{511} H(u,y) / 512 \quad (5)$$

La relation (3) se décompose ainsi : dans une des 32 bandes de l'image, $H(u,y)$ est le coefficient de la transformée de Hadamard à une dimension, d'indice u de l'un des 512 sous blocs d'indice y , composant cette bande, σ_u est issu du calcul de l'écart-type de l'une des 16 colonnes d'indice u , composée de 512 coefficients de même indice. Enfin, $G(u,y)$ est le nouveau coefficient normalisé prêt à subir le codage.

4 - CODAGE

Vu du codeur DPCM chaque bloc à coder se présente sous la forme d'une colonne de 512 éléments dont la base de prédiction est $G(u,0)$. Une fois que σ_u et $G(u,0)$ sont transmis, l'erreur de prédiction $e_{u,1}$ est calculée, comme va nous le montrer l'exemple suivant dans le cas du codage de la première colonne des coefficients d'ordre 0 d'une bande :

- 1) transmission de σ_0 et de $G(0,0)$,
- 2) $\hat{G}(0,1) = G(0,0)$
 $e(0,1) = G(0,1) - \hat{G}(0,1) \longrightarrow e^q(0,1) \longrightarrow \text{Codage}$
- 3) $\hat{G}(0,2) = G(0,1) + e^q(0,1)$
 $e(0,2) = G(0,2) - \hat{G}(0,2) \longrightarrow e^q(0,2) \longrightarrow \text{Codage}$
- ⋮
- 512) $\hat{G}(0,511) = G(0,510) + e^q(0,510)$
 $e(0,511) = G(0,511) - \hat{G}(0,511) \longrightarrow e^q(0,511) \longrightarrow \text{Codage}$

Ici l'estimateur est d'ordre 0, c'est à dire que la valeur à estimer est supposée égale à la valeur précédente reconstituée.

Quantification des erreurs

dans cette application la caractéristique de quantification est linéaire avec saturation à la valeur 3, comme le représente la Figure 1.

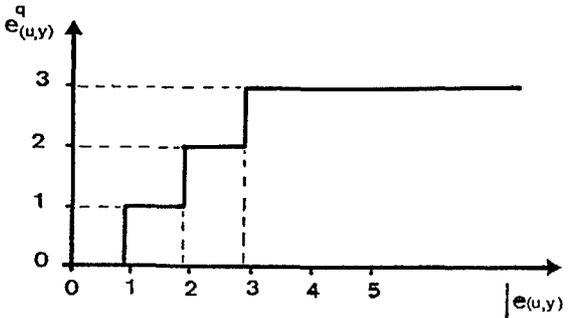


Figure 1. Courbe de quantification

Format de codage

Les résultats d'une telle opération dans le cas du codage des 16 coefficients sur une image se présente de telle sorte que nous avons :

- a) les 512 écarts-types, en entier, des 16 coefficients de chacune des 32 bandes,
- b) les 7 mots de code et le nombre de bits par mot de code,
- c) la base de prédiction et les 511 erreurs de prédiction quantifiées et codées, et cela à raison de 512 fois, soit autant

qu'il y a de colonnes.

5 - RESULTATS

Quand nous parlons de faire varier le facteur de normalisation des composantes, il s'agit en fait d'attribuer respectivement à F_n (facteur de normalisation) les valeurs $\sigma/16, \sigma/8, \sigma/4, \sigma/2, \sigma$ et 2σ , où σ est l'écart-type d'une colonne de 512 coefficients.

Ainsi, en faisant varier F_n , nous avons relevé dans le tableau 1 la répartition des niveaux de quantification des erreurs de prédiction. Comme on pouvait s'y attendre, lorsque F_n est élevé, soit égal à 2σ ou σ , la quantification se fait de façon grossière et augmente par la même occasion le coefficient de corrélation entre les coefficients de même indice, ce qui a pour effet de concentrer les erreurs de prédiction vers 0. En revanche, au fur et à mesure que F_n décroît de $\sigma/2$ à $\sigma/16$, notamment lorsque $F_n = \sigma/16$, les niveaux -3 et $+3$ deviennent très fréquents ; cela est un signe de saturation du quantificateur, c'est à dire qu'il suit mal la dynamique de l'entrée. Inversement, lorsque F_n est important, tout se passe comme si les intervalles d'entrée du quantificateur étaient trop grands par rapport à la majorité des valeurs des erreurs de prédiction. Ces deux inconvénients, dus aux valeurs extrêmes de F_n , sont sanctionnés par l'obtention d'un rapport signal à bruit relativement faible comme nous le verrons ultérieurement.

	$\sigma/16$	$\sigma/8$	$\sigma/4$	$\sigma/2$	σ	2σ
P_{-3}	0,2	0,16	0,101	0,042	0,003	0,001
P_{-2}	0,089	0,089	0,08	0,049	0,013	0,0015
P_{-1}	0,123	0,137	0,153	0,149	0,1	0,035
P_0	0,176	0,228	0,332	0,520	0,768	0,925
P_{+1}	0,123	0,137	0,153	0,149	0,1	0,035
P_{+2}	0,089	0,089	0,08	0,049	0,013	0,0015
P_{+3}	0,2	0,16	0,101	0,042	0,003	0,001
Entropie	2,73	2,74	2,61	2,12	1,17	0,49

Tableau 1. Répartition des niveaux de sortie du quantificateur et de l'entropie en fonction des valeurs du facteur de normalisation, pour 16 composantes.

Le code de Huffman permet de s'adapter aux différentes distributions du tableau 1 en engendrant les mots de code représentés dans le tableau 2. On



note que plus la répartition est concentrée sur la valeur 0, moins il faut de bits pour coder le 0, et plus le taux de compression augmente. Cependant, en comparant l'entropie (tableau 1) et le nombre moyen de bits alloués (tableau 2) à chaque point codé de chacune des distributions relatives aux différents F_n , on s'aperçoit que de $\sigma/16$ à σ , le nombre moyen de bits est convenable. En revanche, pour $F_n = 2\sigma$, ce nombre moyen est de 1,19 alors que l'entropie est de 0,49. Une telle différence s'explique par le fait qu'on ne puisse attribuer moins d'un bit à un mot de code.

	$\sigma/16$	$\sigma/8$	$\sigma/4$	$\sigma/2$	σ	2σ
-3	10	000	101	1110	101010	10110
-2	0000	110	110	1100	1011	10100
-1	010	010	011	101	11	11
0	001	10	00	0	0	0
+1	011	011	010	100	100	100
+2	0001	111	111	1101	10100	10101
+3	11	001	100	1111	101011	10111
Nb	2,84	2,34	2,73	2,20	1,48	1,19

Tableau 2. Evolution du nombre de bits/point (Nb) et des mots de code en fonction des valeurs de normalisation, pour 16 composantes.

Après examen des images restituées figures 2 à 6 et au vu de leur rapport signal à bruit, il est manifeste que des valeurs de facteur de normalisation telles que σ et 2σ sont à exclure. De plus, bien que l'image de la figure 3 représente le meilleur rapport signal à bruit, 23.2 dB, celle-ci semble floue. Cet état de fait confirme nettement ce que nous disions auparavant, à savoir que pour $F_n = \sigma/16$ et $F_n = \sigma/8$, le quantificateur des erreurs de prédiction suit difficilement la dynamique de l'entrée, de telle sorte que le signal à coder subit un lissage. Par conséquent, attribuer à F_n la valeur $\sigma/4$ semble être le meilleur compromis, mais il est bien évident qu'une observation attentive de cette image figure 4 révèle des défauts de restitution.



Figure 2. Image originale

Figure 3. $F_n = \sigma/8$,
 $\tau = 2,8$, $S/N = 23,2$ dBFigure 4. $F_n = \sigma/4$,
 $\tau = 2,9$, $S/N = 22,8$ dBFigure 5. $F_n = \sigma/2$,
 $\tau = 3,6$, $S/N = 17,3$ dBFigure 6. $F_n = \sigma$
 $\tau = 5,3$, $S/N = 12$ dB

La dernière exploitation de cette technique de codage, tout en faisant varier le facteur de normalisation, a été de diminuer progressivement le nombre de composantes de la THD à coder de 16 à 4. Tous ces résultats sont visibles dans le tableau 3.

	$\sigma/16$	$\sigma/8$	$\sigma/4$	$\sigma/2$	σ	Nombre de composantes
S/n	18,6	23,2	22,8	17,3	12	16
τ	2,8	2,8	2,9	3,6	5,3	
S/n	18,5	22,7	21,8	17,4	12	12
τ	3,8	3,8	4,1	5	7,1	
S/n	18,4	21,7	22,2	17,3	12	8
τ	5,7	5,7	6,8	9,7	12,5	
S/n	17,8	21,1	20,6	16,8	11,8	4
τ	11,4	12,2	16,5	21,3	24,4	

Tableau 3. Variation du rapport signal à bruit S/n en dB et du taux de compression τ , en fonction des valeurs du facteur de normalisation et du nombre de composantes conservées.

De ces différentes valeurs sont extraites les caractéristiques suivantes :

a) en figure 7 l'évolution du rapport signal à bruit S/N en fonction des valeurs de F_n avec pour paramètre le nombre de composantes codées. Cette courbe nous permet de voir que le rapport signal à bruit passe par un maximum qui, quel que soit le nombre de composantes conservées, se situe entre $\sigma/8$ et $\sigma/4$;

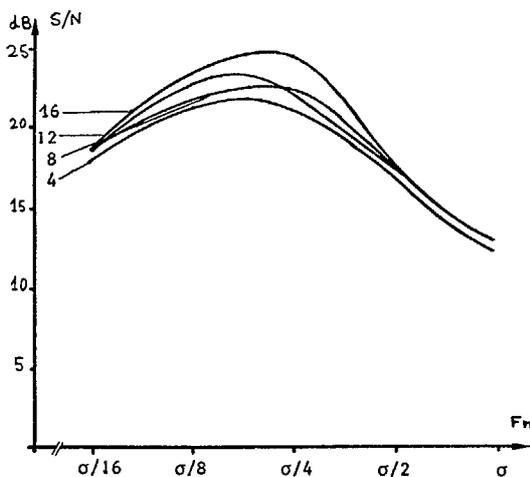


Figure 7. Rapport signal à bruit (S/N) en fonction du facteur de normalisation (F_n) avec pour paramètre le nombre de composantes de la THD conservées.

b) en figure 8 l'évolution du taux de compression en fonction des valeurs de F_n avec pour paramètre le nombre de composantes codées. Cette caractéristique nous confirme que le taux de compression croît avec la valeur de F_n , et inversement, avec le nombre de composantes conservées, la non linéarité de ces courbes étant essentiellement due à l'utilisation d'un code à longueur variable.

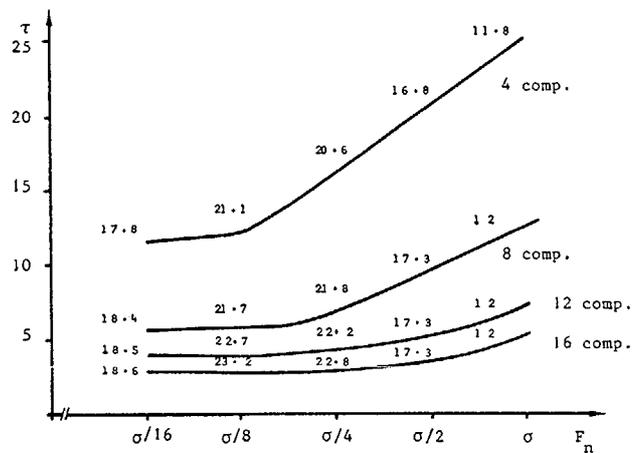


Figure 8. Evolution du taux de compression en fonction du facteur de normalisation avec en paramètres :

- le rapport signal à bruit en dB,
- le nombre de composantes conservées.

BIBLIOGRAPHIE

- [1] HABIBI A. and WINTS P. A. "Hybrid coding by linear transformations and block quantization", 1971, IEEE Trans Comm, Tech, Vol. COM-19, P. 50-62.
- [2] HABIBI A. "Comparison of N^{th} order DPCM Encoder with linear Transformation and Block Quantization Technics".
- [3] HABIBI A. and WINTZ P.A. "Hybrid Coding of Pictorial data" IEEE Trans Comm Tech, Vol. COMM 22, N° 5, P. 614-624, 1974.
- [4] HUFFMAN D. A. "A method for the Construction of Minimum redundancy codes", Proc IRE, Vol. 40, N° 10, P. 1098-1101, 1952.
- [5] ANDREWS M.C. "Computers techniques in Image Processing", Academic press, 1970.