

Compression d'images à l'aide d'un codage Hybride Huffman et Shanon-Fano

[Compression of imagery to help of a coding Hybrid Huffman and Shanon-Fano]

Hassane ROUKHE¹, Abdenbi BOUZID², and Ahmed ROUKHE²

¹Department Techniques et ingénierie,
Institut Supérieur des Métiers d'Audiovisuel et du Cinéma, ISMAC,
Rabat, Maroc

²Department de physique,
Université Moulay Ismaïl, Faculté des sciences Meknès,
Meknès, Maroc

Copyright © 2015 ISSR Journals. This is an open access article distributed under the **Creative Commons Attribution License**, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

ABSTRACT: The very great volume of information and data in a digital image can cause practical problems. Transmitting an image from one computer to another and/or archiving are very expensive due to the abundance of data representing the image in the form of bits. We present in this article a compression method, which takes in account the coding hybrid of Huffman and Shanon Fano, which has been applied to imagery and data.

We've determined the method limit by considering different forms of images histogram. The results showed that the method is efficient when the number of bit by pixel is of the same order of magnitude as entropy.

KEYWORDS: Coding Hybrid, Coding of Huffman, Coding of Shanon Fano, entropy, redundancy of codes, compression, decode.

RESUME: Un grand volume d'informations et de données dans une image numérique peut entraîner des problèmes pratiques. La transmission d'une image d'un ordinateur à un autre et/ou l'archivage sont très coûteux en raison de l'abondance des données représentant l'image sous forme de bits. Nous présentons dans cet article une méthode de compression qui prend en considération le codage hybride de Huffman et de Shanon-Fano, qui a été appliquée à des images et des données.

Nous avons déterminé la limite de la méthode en considérant des images à différentes formes d'histogramme. Les résultats ont montré que la méthode est efficace dans le cas où le nombre de bits par pixel est du même ordre de grandeur que l'entropie.

MOTS-CLEFS: Codage Hybrid, Codage de Huffman, Codage de Shanon Fano, entropie, redondance de codes, compression, décodage.

1 INTRODUCTION

L'utilisation de données sous leurs formes numériques ne serait pas possible aujourd'hui sans la compression préalable de celles-ci. Et ceci pour plusieurs raisons [1],[2] :

- Les capacités de stockages des utilisateurs, même si elles ne cessent d'augmenter, ne sont pas infinies.
- La durée des transmissions de ces données numériques est conditionnée par le débit du réseau qui est utilisé et qui est parfois relativement faible.

La compression des données numériques permet donc de diminuer la taille de stockages et de rendre possible leurs transports sur des réseaux de communications (Internet, GSM, câble, TV satellite).

Dans ce papier nous présentons un nouvel algorithme à partir des algorithmes de codage tel que Huffman et Shanon-fano [3][4].

Dans un premier temps nous allons présenter quelques définitions de l'entropie [5],[6],[7] d'une source d'information et les propriétés et caractéristiques d'une entropie..

Puis nous représentons les algorithmes de codage de Huffman et Shanon fano, nous illustrons les résultats obtenus par ces algorithmes du même message.

Et après nous présentons notre approche d'un algorithme Hybrid ainsi que les résultats obtenus pour les comparés et les évalués aux résultats précédents.

Une fois cette nouvelle approche vérifie les critères d'une entropie et d'un code [8], nous proposons une application sur les images comme cas d'un message d'une longueur assez élevée d'où nous allons faire une comparaison entre un codage avec la méthode hybride et la méthode Huffman et/ou Jpeg [9].

Cette méthode nous permet de bien évaluer les résultats dans les cas d'un message de longueur réduite et dans le cas où le message d'une longueur assez élevée.

2 ENTROPIE D'UNE SOURCE D'INFORMATION

2.1 DEFINITION 1: NOTION D'ENTROPIE D'UNE SOURCE SIMPLE OU SANS MEMOIRE

Soit une source S définie par son alphabet $\{s_1, s_2, \dots, s_N\}$ de symboles et ses caractéristiques d'émission régies par une loi de probabilité P :

$\{P(s_1), P(s_2), \dots, P(s_N)\}$. Une source sera dite simple (ou sans mémoire) si les symboles émis par la source S sont indépendants et de même loi. Une suite de N symboles émis aux instants 1, 2,N par S suit donc une loi de probabilité : $P(s_1, s_2, \dots, s_N) = p(s_1) p(s_2) \dots P(s_N)$.

L'entropie d'ordre zéro H(S) d'une source simple S, de loi de probabilité P, est définie par l'expression:

$$H(S) = \sum_{i=1}^{N-1} -p(s_i) \log_2 p(s_i)$$

2.2 REMARQUE

L'entropie dans ce cas a la propriété suivante: H(S) est maximale si tous les symboles $\{s_1, s_2, \dots, s_N\}$ de la source S sont équiprobables. Dans ce cas, l'entropie est égale à l'information associée à chaque message pris individuellement.

$$\forall i \in \{1, 2, \dots, N\}, p(s_i) = \frac{1}{N} \Leftrightarrow I(S) = \log_2 p(N)$$

2.3 CALCUL DE L'ENTROPIE D'UNE SOURCE BINAIRE

Considérons le cas d'une source binaire S dont l'alphabet est $\{0,1\}$ tel que $P(1) = p$ et donc $P(0)=1-p$ avec $0 < p \leq 1$.

$$H(S) = -p \log_2(p) - (1-p) \log_2(1-p) = f(p)$$

la fonction f(p) est symétrique par rapport à $p=0.5$ et est maximale.

2.4 DEFINITION 2: ENTROPIE D'ORDRE K

On suppose une source à amplitude discrète avec mémoire, Il existe alors des dépendances statistiques entre les échantillons successifs de la source.

Soit $s = s(n) = \{s_n, s_{n+1}, \dots, s_{n+K-1}\}^T$ un vecteur constitué de K échantillons successifs de la source. Ce vecteur est caractérisé par sa probabilité conjointe $PS(s)$ qui est indépendante du temps si on considère une source stationnaire, s'est

une réalisation du vecteur aléatoire $S = \{S_n, S_{n+1}, \dots, S_{n+K-1}\}$. On désigne par entropie d'ordre k ou entropie conjointe des vecteurs aléatoires l'expression:

$$H_K(S) = \frac{1}{K} E(-\log_2 P_s(S)) = \frac{-1}{K} \sum_s \dots \sum_s p_s(s) \log_2 p_s(s)$$

3 DEFINITIONS ET RÉSULTATS FONDAMENTAUX

3.1 DEFINITION 3: RÉGULARITÉ

Un code est dit régulier si tous les mots de code mi sont distincts. Tous les codes doivent au moins être réguliers pour permettre un décodage univoque.

3.2 DEFINITION 4: DÉCHIFFRABILITÉ

Un code régulier est déchiffrable si pour toute suite m_1, m_2, \dots, m_n de mots de code il est possible de distinguer les m_i sans ambiguïté et reconstruire ainsi les symboles si correspondants.

Pour les codes de longueurs variables, cette propriété est satisfaite pour les codes dits sans préfixes, obtenus en évitant qu'un mot du code ne soit identique au début d'un autre.

3.3 DEFINITION 5: CODES INSTANTANÉS

On dit d'un code qu'il est à décodage instantané s'il est possible de décoder les mots de code dès lors que tous les symboles qui en font partie ont été reçus.

3.4 DEFINITION 6: EXTENSION D'ORDRE N D'UN CODE

L'extension d'ordre n d'un code, est le code formé par les séquences de n mots du code initial.

3.5 CONDITION NECESSAIRE ET SUFFISANTE DE DECHIFFRABILITE

Pour qu'un code soit déchiffrable il faut et il suffit que toutes ses extensions soient régulières.

L'inégalité de Kraft constitue un résultat fondamental en théorie des codes. Elle fournit une condition nécessaire et suffisante d'existence de codes déchiffrables et instantanés, exprimée en fonction de la longueur des mots de code.

3.6 INÉGALITÉ DE KRAFT

Soient l_1, l_2, \dots, l_N des longueurs de mots candidates pour coder une source N-aire dans un alphabet binaire. Alors l'inégalité de Kraft :

$$\sum_{i=1}^N \frac{1}{2^{l_i}} \leq 1$$

Est une condition nécessaire et suffisante d'existence de codes déchiffrables et instantanés respectant ces longueurs de mots.

3.7 THÉORÈME DE SHANNON

Limite inférieure de la longueur moyenne d'un code.

Soit une source stationnaire sans mémoire, la longueur moyenne des mots codés m_i est limitée par la valeur de l'entropie de la source S.

$$\bar{L} = \sum_{i=1}^N l_i p(s_i) \geq \sum_{i=1}^N p(s_i) \log_2 (p(s_i)).$$

Borne supérieure de la limite supérieure (code de Shannon) il est également possible de trouver un code déchiffrable tel que:

$$H(S) \leq \bar{I} \leq H(S) + 1$$

4 LES ALGORITHMES DE CODAGE STATISTIQUES

4.1 MÉTHODE 1: ALGORITHME DE SHANNON-FANO

L'algorithme comporte les étapes suivantes:

- 1) Les probabilités d'apparition de chaque symbole sont placées dans un tableau trié par ordre décroissant de probabilités.
- 2) Le tableau est coupé en deux groupes de symboles S_0 et S_1 dont la somme des probabilités de chaque groupe avoisine 0,5.
- 3) Le groupe S_0 est codé par un "0" et S_1 par "1".
- 4) Si un groupe S_i n'a qu'un seul élément, c'est une feuille terminale, sinon la procédure reprend récursivement à l'étape 2 sur le groupe S_i .

4.2 APPLICATION DE L'ALGORITHME DE SHANNON-FANO

Pour illustrer cet algorithme, nous allons coder la phrase suivante:

" INTERNATIONAL CONFERENCE ON OPERATOR THEORY "

{ I N T E R A O L S p a c e C F P H Y }

Char	A	C	E	F	H	I	L	N	O	P	R	T	Y	Spa.
Fréq.	3	2	6	1	1	2	1	6	6	1	5	4	1	4

Etape 1 :

On commence par le calcul des probabilités à partir des fréquences d'apparition on obtient le tableau suivant:

Char	A	C	E	F	H	I	L	N	O	P	R	T	Y	Spa
Fréq.	3	2	6	1	1	2	1	6	6	1	5	4	1	4
Prob.(%)	7	5	14	2	2	5	2	14	14	2	12	9	2	9

Après le calcul des probabilités on classe les symboles dans un ordre décroissant comme suit:

Char	E	N	O	R	T	Spa	A	C	I	F	H	L	P	Y
Fréq.	6	6	6	5	4	4	3	2	2	1	1	1	1	1

Application successive des étapes 2,3 et 4, nous obtenons le résultat suivant:

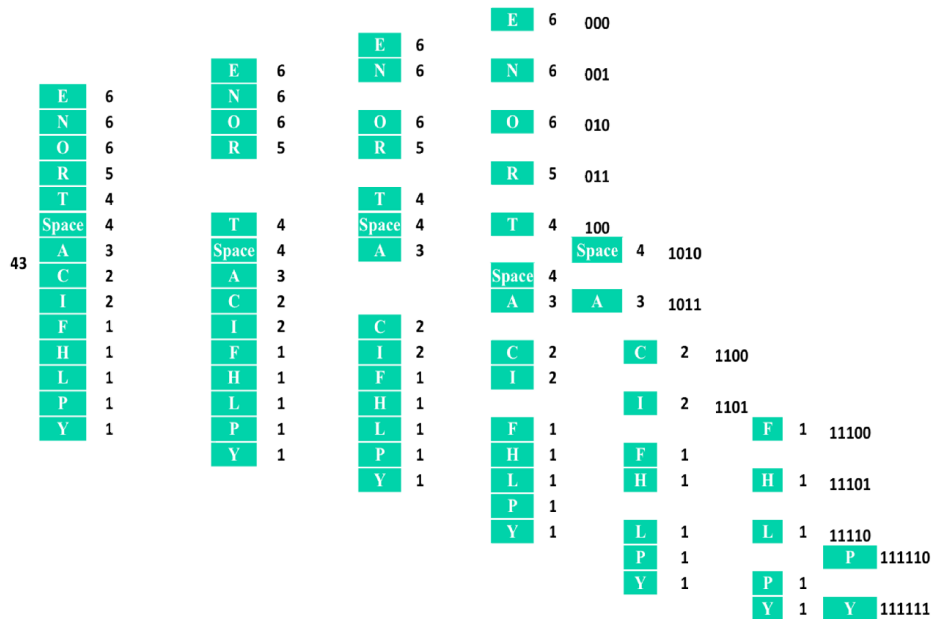


Fig. 1. Algorithme de shannon-Fano

Le résultat de codage est illustrer dans le tableau suivant où l est la longueur de code obtenu.

TABLE 1. Résultats des codes obtenus par Shannon-Fano

Char	Code	l
A	1011	4
C	1100	4
E	000	3
F	11100	5
H	11101	5
I	1101	4
L	11110	5
N	001	3
O	010	3
P	111110	6
R	011	3
T	100	3
Y	111111	6
Space	1010	4

4.3 MÉTHODE 2: ALGORITHME DE HUFFMAN

Comme pour le codage de shannon-Fano, les probabilités d'apparition des symboles sont placées dans un tableau trié par ordre décroissant de probabilités.

L'algorithme de Huffman est implémenté suivant une structure d'arbre.

Le principe de cet algorithme consiste à regrouper les deux symboles de probabilité la plus faible pour en faire un nouveau symbole dont la probabilité est la somme des probabilités de ces deux symboles. On itère cette opération et à chaque étape le nombre de symboles diminue. On construit de cette manière un arbre dont les feuilles sont les symboles à coder et les embranchements les codages intermédiaires.

4.4 APPLICATION DE L'ALGORITHME DE HUFFMAN

Considérons à nouveau le même exemple précédent, Il s'agit de la phrase:

"INTERNATIONAL CONFERENCE ON OPERATOR THEORY ".

{ I N T E R A O L S p a c e C F P H Y }

Char	A	C	E	F	H	I	L	N	O	P	R	T	Y	Spa.
Fréq.	3	2	6	1	1	2	1	6	6	1	5	4	1	4

Etape 1 :

On commence par le calcul des probabilités à partir des fréquences d' apparition et on obtient le tableau suivant:

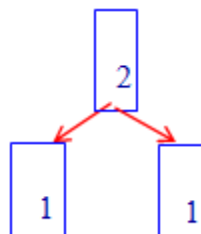
Char	A	C	E	F	H	I	L	N	O	P	R	T	Y	Spa
Fréq.	3	2	6	1	1	2	1	6	6	1	5	4	1	4
Prob.(%)	7	5	14	2	2	5	2	14	14	2	12	9	2	9

Après le calcul des probabilités on classe les symboles dans un ordre décroissant comme suit:

Char	E	N	O	R	T	Spa	A	C	I	F	H	L	P	Y
Fréq.	6	6	6	5	4	4	3	2	2	1	1	1	1	1

Application du principe de l'algorithme de Huffman nous permet d'obtenir le résultat suivant:

Pour une première étape nous arrivons à regroupe les deux premières symboles à faible probabilité:



Par la suite nous réitérons cette opération d'une façon successive et nous obtenons l'arbre à la figure 2.

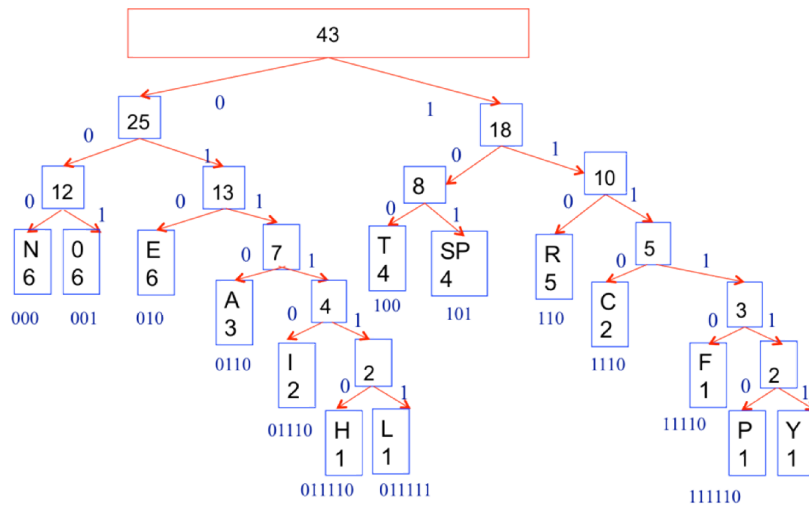


Fig. 2. Arbre de Huffman

Le résultat de codage est illustrer dans le tableau suivant où l est la longueur de code obtenu.

TABLE 2. Résultats des codes obtenus par Huffman

Char	Code	l
A	0110	4
C	1110	4
E	010	3
F	11110	5
H	011110	6
I	01110	5
L	011111	6
N	000	3
O	001	3
P	111110	6
R	110	3
T	100	3
Y	111111	6
Space	101	3

5 ALGORITHME DE CODAGE HYBRIDE

Notre apport et de réaliser un codage hybride entre Huffman et Shannon-Fano, qui se base sur les deux approches des algorithmes précédents.

5.1 MÉTHODE DE L'ALGORITHME HYBRIDE

Comme pour le codage de shannon-fano, les probabilités d'apparition des symboles sont placées dans un tableau trié par ordre décroissant de probabilités. Notre Algorithme hybride est implémenté suivant la structure qui combine entre l'algorithme de Huffman et Shannon fano selon la structure d'arbre.

Le principe de l'algorithme comporte les étapes suivantes:

- 1) Les probabilités d'apparition de chaque symbole sont placées dans un tableau trié par ordre décroissant de probabilités.
- 2) Pour l'itération i:

Si i est impaire : Le tableau est coupé en deux groupes de symboles S_0 et S_1 dont la somme des probabilités de chaque groupe avoisine 0,5.

Si i est paire: On regroupe les deux symboles de probabilités la plus faible pour en faire un nouveau symbole dont la probabilité est la somme des probabilités de ces deux symboles.

3) Le groupe S_0 est codé par un "0" et S_1 par "1".

4) Si un groupe S_i n'a qu'un seul élément, c'est une feuille terminale, sinon la procédure reprend récursivement à l'étape 2 sur le groupe S_i .

5.2 APPLICATION DE L'ALGORITHME HYBRIDE

Pour illustrer cet algorithme, nous allons coder la même phrase des exemples précédents:

" INTERNATIONAL CONFERENCE ON OPERATOR THEORY "

{ I N T E R A O L Space C F P H Y }

Char	A	C	E	F	H	I	L	N	O	P	R	T	Y	Spa.
Fréq.	3	2	6	1	1	2	1	6	6	1	5	4	1	4

Etape 1 :

On commence par le calcul des probabilités à partir des fréquences d'apparition et on obtient le tableau suivant:

Char	A	C	E	F	H	I	L	N	O	P	R	T	Y	Spa
Fréq.	3	2	6	1	1	2	1	6	6	1	5	4	1	4
Prob.(%)	7	5	14	2	2	5	2	14	14	2	12	9	2	9

Après le calcul des probabilités on classe les symboles dans un ordre décroissant comme suit:

Char	E	N	O	R	T	Spa	A	C	I	F	H	L	P	Y
Fréq.	6	6	6	5	4	4	3	2	2	1	1	1	1	1

Après l'application du principe de l'algorithme Hybride on réitère les algorithmes de Shannon Fano et Huffman selon la parité de l'itération et nous obtenons le schéma au fig 3 suivant:

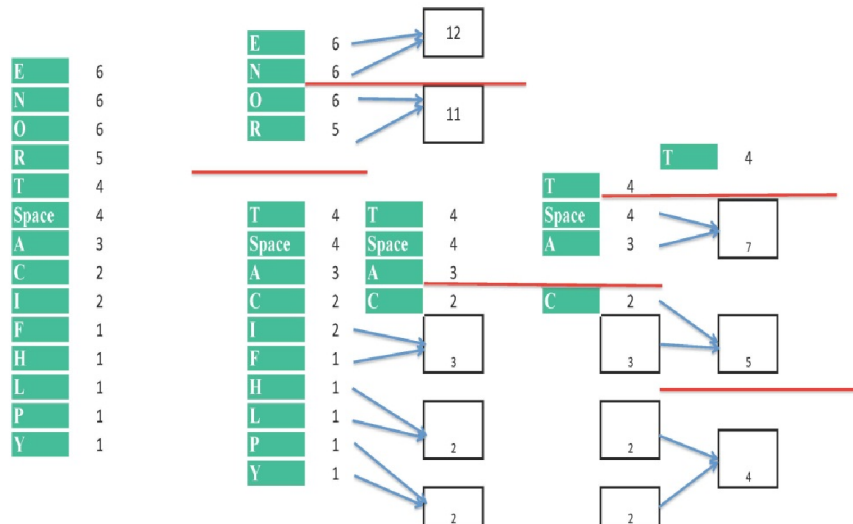


Fig. 3. Schéma de l'algorithme Hybride

5.3 RÉSULTAT DE CODAGE HYBRIDE

Le résultat de codage est illustrer dans le tableau suivant où l est la longueur de code obtenu.

TABLE 3. Résultats des codes obtenus par Huffman

Char	Code	l
A	0110	4
C	1110	4
E	010	3
F	11110	5
H	011110	6
I	01110	5
L	011111	6
N	000	3
O	001	3
P	111110	6
R	110	3
T	100	3
Y	111111	6
Space	101	3

5.4 PERFORMANCE DES ALGORITHMES

Pour mieux analyser le résultat obtenu par le codage hybride, nous allons faire appel à l'entropie utilisée est le calcul de la moyenne par symbole:

$$H = - \sum pi * \log_2(pi)$$

$$= -(0,07 \log_2(0,07) + 0,05 \log_2(0,05) + \dots + 0,09 \log_2(0,09)) = 3,45$$

Et on vérifie également l'inégalité de Kraft d'où:

$$\sum_{i=1}^N \frac{1}{2^{l_i}} = 1 \leq 1$$

Donc les codes obtenus par notre algorithme sont déchiffrables et instantanés.

TABLE 4. Tableau comparatif des résultats des codes obtenus

Char	A	C	E	F	H	I	L	N	O	P	R	T	Y	Space
Fréq.	3	2	6	1	1	2	1	6	6	1	5	4	1	4
Prob.	0,07	0,05	0,14	0,02	0,02	0,05	0,02	0,14	0,14	0,02	0,12	0,09	0,02	0,09
pi log2(pi)	-0,27	-0,22	-0,40	-0,11	-0,11	-0,22	-0,11	-0,40	-0,40	-0,11	-0,37	-0,31	-0,11	-0,31
H	3,45													
Source														
l	4	4	3	5	6	5	6	3	3	6	3	3	6	3
moy(l)=	0,28	0,2	0,42	0,1	0,12	0,25	0,12	0,42	0,42	0,12	0,36	0,27	0,12	0,27
3,47	12	8	18	5	6	10	6	18	18	6	15	12	6	12
Huffman	152													
l	4	4	3	5	5	4	5	3	3	6	3	3	6	4
moy(l)=	0,28	0,2	0,42	0,1	0,1	0,2	0,1	0,42	0,42	0,12	0,36	0,27	0,12	0,36
3,47	12	8	18	5	5	8	5	18	18	6	15	12	6	16
Shannon-fano	152													
l	4	4	3	5	5	5	5	3	3	5	3	3	5	4
moy(l)=	0,28	0,2	0,42	0,1	0,1	0,25	0,1	0,42	0,42	0,1	0,36	0,27	0,1	0,36
3,48	12	8	18	5	5	10	5	18	18	5	15	12	5	16
Hybride	152													

D'après les résultats obtenus par les trois méthodes on peut illustré que l'algorithme hybride donne les mêmes résultats que ceux de Huffman et Shannon-Fano, pour des messages S de taille réduite.

S="INTERNATIONAL CONFERENCE ON OPERATOR THEORY "

Débit initiale $= I(S) = 43 * 8 = 344$.

Code : S' est composés des mots de codes alors :

Débit final $= I(S_H') = f_1 * l_1 + f_2 * l_2 + \dots + f_N * l_N = 152$

Débit final $= I(S_{SF}') = f_1 * l_1 + f_2 * l_2 + \dots + f_N * l_N = 152$

Débit final $= I(S_{Hyb}') = f_1 * l_1 + f_2 * l_2 + \dots + f_N * l_N = 152$

Donc les résultats obtenus sont importants dans le cas des messages de petite taille, la question qui se pose est ce que l'algorithme de codage hybride reste performant dans le cas des messages de taille importante, et est-ce que l'algorithme hybride est adapté à la compression des images

6 APPLICATION DE CODAGE HYBRIDE A LA COMPRESSION DES IMAGES

On peut remarquer que l'hypothèse d'une source sans mémoire n'est pas respectée, une première redondance existe et il faut la diminuer. Il est possible de s'approcher de l'hypothèse d'une source sans mémoire, si sur chaque ligne d'une image, on calcule simplement l'erreur entre le niveau de gris du pixel courant $I(i,j)$ et celui du précédent sur la ligne $I(i-1,j)$: $e(i,j) = I(i,j) - I(i-1,j)$.

6.1 TRANSFORMATION DCT

On peut s'en convaincre par les histogrammes de l'image de la fig. 4. et de l'image différence.

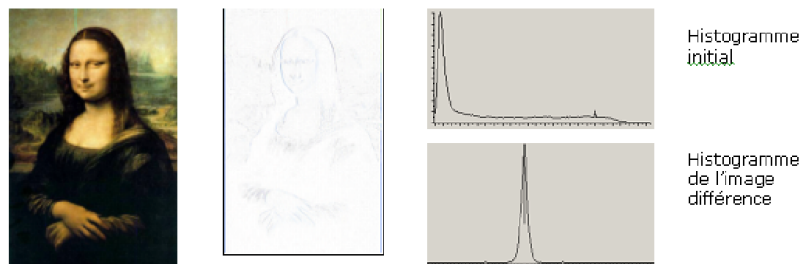


Fig. 4. L'image I, l'image différence et leurs histogrammes

Dans la pratique, les algorithmes de codage entropique [10] ne sont pas utilisés directement sur les valeurs des pixels de l'image, mais sur des grandeurs obtenues après transformation des valeurs des pixels (comme la DCT par exemple) fig.5.

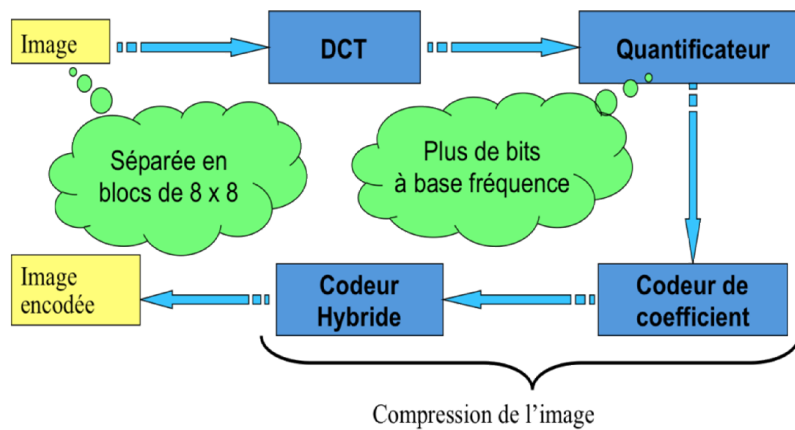


Fig. 5. Schéma de traitement – compression de l'image

6.2 LE CODAGE ENTROPIQUE

Après la transformation DCT [11], nous aurons besoin à un codage des coefficients pour la mise en ordre des composantes de l'image, on utilisant le balayage zigzag des pixel comme ci-dessous dans la fig. 6.

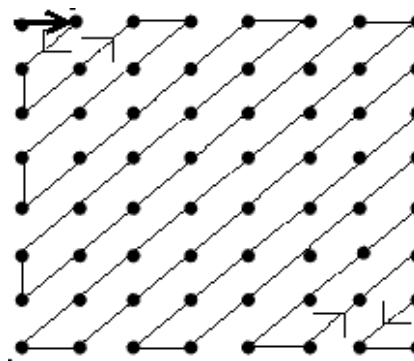


Fig. 6. Balayage des pixels de l'image

Puis nous allons arriver à la phase d'application de l'algorithme de codage qui nous a permis d'avoir une image encodée. Et pour retourner à la phase initiale nous appliquons le schéma de décompression comme suit.

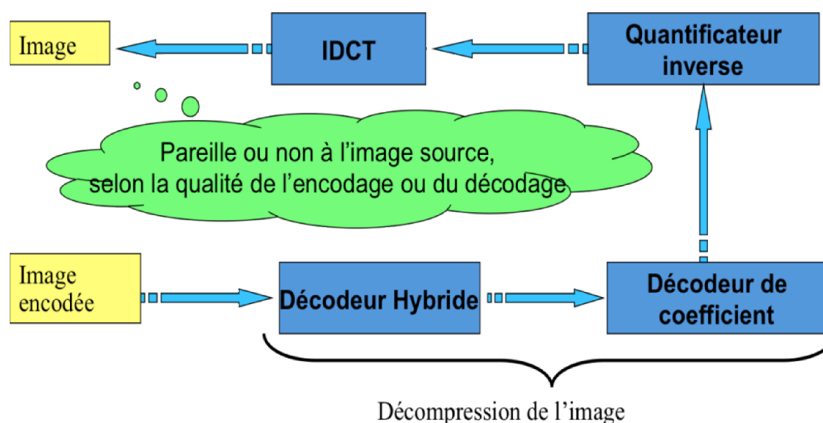


Fig. 7. Schéma de traitement – décompression de l'image

6.3 LE CODAGE HYBRIDE CONTRE HUFFMAN

La méthode de codage hybride a été appliquée avec succès sur des images qui nous permettent d'obtenir des messages à longue taille, ce résultat nous permet de généraliser l'application de l'approche hybride et qui nous a permis d'obtenir un résultat important dans la compression des images fig. 8.



Fig. 8. Compression Hybride vs Jpeg

7 CONCLUSION

Aussi, les résultats exploratoires obtenus dans ce papier mériteraient d'être approfondis par des expérimentations plus rigoureuses. Tout d'abord, on aurait besoin d'augmenter l'envergure des tests d'appréciation visuelle des images en ajoutant des observateurs cobayes et en augmentant le nombre des images à classer dans une échelle qualitative.

Aussi, les conditions des expériences (distance entre l'observateur et l'image, luminosité ambiante, calibration des couleurs de l'écran, temps d'observation des images) gagneraient à être contrôlées et décrites, l'analyse des résultats devrait se faire avec des outils statistiques plus sophistiqués que le dénombrement, la moyenne et l'écart-type.

Il serait très intéressant de comparer la performance de la compression fractale avec les deux autres algorithmes de compression d'images. On se doute que cet autre algorithme fera moins bien que Huffman, mais on ne sait pas s'il peut battre JPEG et JPEG 2000. En fait, sa performance relative devrait dépendre du taux de compression choisi.

REMERCIEMENTS

Ce travail a été réalisé dans le cadre de la collaboration entre le département Techniques et ingénierie à l'ISMAR Rabat et le Laboratoire de Traitement d'image, Optronique et Vision Artificielle à la faculté des sciences de l'Université Moulay Ismaïl Meknès.

REFERENCES

- [1] T. Ramstad, S. O. Aase & J. H. Husoy, *Subband compression of images*, Elsevier Science Inc. 1995.
- [2] D.S. Taubman, M.W. Marcellin, *JPEG 2000: Image Compression Fundamentals*, 2001.
- [3] D.A. Huffman, *A method for the construction of minimum-redundancy codes*, 1952.
- [4] I.H. Witten, R.M. Neal & J.G. Cleary, *Arithmetic coding for data compression*, 1987.
- [5] Jean yves BOUDEC, Patrick THIRAN, Rüdiger URBANK, *Sciences de l'information*, Novembre 2014.
- [6] Robert Mc Eliece, *The Theory of Information and Coding*, Addison Wesley, 1977.
- [7] Thomas Cover, Joy Thomas, *Elements of Information Theory*, J. Wiley & sons, 1991.
- [8] M Bigerelle, A Iost, *Relations entre l'entropie physique, le codage de l'information et l'énergie de simulation*, *Revue canadienne de physique*, 85(12): 1381-1394, 10.1139/p07-136, 2007.
- [9] R.J. Clarke, *Transform coding of images*, Academic Press 1985.
- [10] J.W. Woods, *Subband Image Coding*, 1990.
- [11] Z. CVETKOVIC, M. V. POPOVIC, *New fast recursive algorithms for computation of discrete cosine and sine transforms*, *IEEE Transactions on signal processing*, vol. 40, no8, pp. 2083-2086 (6 ref.), 1992.