

COMPRESSION D'IMAGES PAR "MATCHING PURSUIT"

Amina SERIR & Boualem SANSAL & Abdelatif SEGHIER

Laboratoire de traitement d'images, Institut d'Electronique, U.S.T.H.B
BP 32 El-Alia , 16011 Alger. (Algérie).

RÉSUMÉ

Nous présentons une extension au cas 2-D, de l'algorithme de décomposition adaptative multi-échelle "Matching pursuit" de S. Mallat. En utilisant les atomes espace- fréquence de Gabor, et un critère d'arrêt basé sur la norme de l'image résiduelle, un taux de compression de l'ordre de 47.31 est alors obtenu.

ABSTRACT

This paper investigates the possibility of an extension at 2-D case of Mallat's algorithm called " Matching Pursuit ". It consists on a signal multiscale adaptatif decomposition on waveforms or atoms. Using Gabor's space-frequency atoms and a test on the resisual image norm, an image compression rate of 47.31 is also reached.

I. INTRODUCTION

Les représentations multi-résolution forment à présent un vaste ensemble de méthodes éprouvées, auxquelles les différents domaines d'applications du traitement du signal font de plus en plus naturellement appel [5,8]. Néanmoins, les décompositions linéaires temps/fréquence en choisissant une base unique (Fourier, ondelettes,...) [8] ne sont pas flexibles pour mettre clairement en évidence toutes les structures locales du signal.

C'est pourquoi plusieurs techniques très différentes ont été proposées récemment pour fournir des représentations plus précises et plus flexibles. Une approche particulière proposée par S.Mallat [6] consiste en la décomposition du signal en formes d'ondes appelées atomes temps/fréquence dont les propriétés sont adaptées aux structures locales du signal. La sélection de ces atomes se fait par un algorithme connu sous le nom de "Matching Pursuit" [3].

La compression de signaux unidimensionnels est une application directe de l'algorithme étudié. Dans le but d'obtenir un résultat équivalent pour la compression d'images 2-D, une stratégie d'extension de l'algorithme au cas bi-dimensionnel est alors élaborée.

Nous commencerons par rappeler les concepts de base de l'algorithme "matching pursuit", avant d'aborder l'extension au cas 2-D. Nous présenterons en dernier, les différents types de codages utilisés pour optimiser le taux de compression.

II. MATCHING PURSUIT

Après avoir établi un dictionnaire d'atomes temps/fréquence, le matching pursuit offre une transformation temps/fréquence adaptative qui décompose toute fonction $f(t) \in L^2(R)$ en une somme d'atomes temps/ fréquence complexes les plus adaptés aux structures locales du signal résiduel.



II.1 Dictionnaire temps/fréquence

Cet ensemble est constitué d'atomes générés par dilatation, translation et modulation d'une fonction fenêtre unique $g(t) \in L^2(\mathbb{R})$.

La fonction $g(t)$ est continue, différentiable, $\|g\| = 1$, l'intégrale de $g(t)$ est non nulle et $g(0) \neq 0$.

Pour toute échelle $s > 0$, fréquence ξ et translation u , on note $\gamma = (s, u, \xi)$ et on définit :

$$g_\gamma(t) = \frac{1}{\sqrt{s}} g\left(\frac{t-u}{s}\right) e^{i\xi t} \quad (1.1)$$

La famille $D = (g_\gamma(t))_{\gamma \in \Gamma}$ $\Gamma \in \mathbb{R}^+ \times \mathbb{R}^2$

est extrêmement redondante. Pour représenter d'une manière précise toute fonction $f(t)$ nous devons sélectionner un sous-ensemble d'atomes appropriés $(g_{\gamma_n}(t))_{n \in \mathbb{N}}$ avec $\gamma_n = (s_n, u_n, \xi_n)$. Et $f(t)$ peut alors s'écrire :

$$f(t) = \sum_{n=-\infty}^{+\infty} a_n g_{\gamma_n}(t) \quad (1.2)$$

II.2 Algorithme du "Matching pursuit"

Le but de cet algorithme est d'établir une décomposition linéaire d'un signal $f(t)$ sur un ensemble de vecteurs sélectionnés dans le dictionnaire D , afin de faire ressortir les structures internes du signal. Ceci est faisable par des approximations successives de $f(t)$ par des projections orthogonale sur les éléments de D .

Soit $g_{\gamma_0} \in D$, le signal $f(t)$ peut être alors décomposé en :

$$f = \langle f, g_{\gamma_0} \rangle g_{\gamma_0} + R_f \quad (1.3)$$

R_f étant le signal résiduel.

Le principal intérêt de cette méthode est d'obtenir un nombre limité d'approximations successives du signal original. Par conséquent, cela revient à minimiser à chaque étape le signal résiduel. Si on considère la norme de l'équation (1.3) :

$$\|f\| = \left| \langle f, g_{\gamma_0} \rangle \right|^2 + \|R_f\|^2 \quad (1.4)$$

on remarquera que pour minimiser la norme du signal résiduel, il faut alors trouver un atome $g_{\gamma_0} \in D$ qui maximise en valeur absolue le produit scalaire.

L'algorithme du matching pursuit décompose le signal après m itérations en une somme de ses projections et du dernier signal résiduel.

$$f = \sum_{i=0}^{m-1} \langle R_f^i, g_{\gamma_i} \rangle g_{\gamma_i} + R_f^m \quad (1.5)$$

L'équation (1.5) montre que l'on peut reconstituer le signal connaissant à chaque étape de la décomposition les paramètres des atomes sélectionnés $\gamma_n = (s_n, u_n, \xi_n)$ et les projections associées des signaux résiduels.

Le nombre d'itérations dépend du taux de décroissance de la norme du signal résiduel.

III. EXTENSION AU CAS 2-D

III.1 Dictionnaire Espace/Fréquence 2-D

Cet ensemble est constitué d'atomes générés par dilatation, translation et modulation d'une fonction fenêtre unique $g(x, y) \in L^2(\mathbb{R}^2)$, pour faciliter l'implémentation de l'algorithme, on choisit des atomes à variables séparables :

$$g(x, y) = g(x) * g(y)$$

La fonction $g(x, y)$ est continue, différentiable, $\|g(x, y)\| = 1$, l'intégrale de $g(x, y)$ est non nulle et $g(0, 0) \neq 0$.

Pour toute échelle ($s_x > 0, s_y > 0$) fréquence $\xi = (\xi_x, \xi_y)$ et translation (u_x, u_y) , on note $\gamma = (s_x, s_y, u_x, u_y, \xi_x, \xi_y)$ et on définit :

$$g_\gamma(x, y) = \frac{1}{\sqrt{s_x} \sqrt{s_y}} g\left(\frac{x-u_x}{s_x}\right) g\left(\frac{y-u_y}{s_y}\right) e^{i(\xi_x x + \xi_y y)} \quad (2.1)$$

$$g_\gamma(x, y) = g_{\gamma_x}(x) g_{\gamma_y}(y)$$

$$\text{avec } \gamma_x = (s_x, u_x, \xi_x) \text{ et } \gamma_y = (s_y, u_y, \xi_y)$$

s_x, u_x et ξ_x étant respectivement l'échelle, la translation et la fréquence suivant la direction x . La famille $D = (g_\gamma(t))_{\gamma \in \Gamma}$ $\Gamma \in \mathbb{R}^{+2} \times \mathbb{R}^4$ est extrêmement redondante.

III.2 Algorithme du "Matching pursuit" dans le cas 2-D

Notre principal but est de réduire la quantité d'information nécessaire pour représenter correctement toutes les structures locales de l'image numérique $I(n, m)$. Par conséquent, nous proposons d'établir une décomposition linéaire adaptative de l'image par projections orthogonales sur un ensemble de vecteurs sélectionnés dans le dictionnaire D_2 .

Soit $g_{\gamma_0} \in D_2$, l'image $I(n, m)$ peut être alors décomposée en :

$$I(n, m) = \left\langle I(n, m), g_{\gamma_0}(n, m) \right\rangle g_{\gamma_0}(n, m) + R_I(n, m) \quad (2.2)$$

RI étant l'image résiduelle.

Pour les mêmes raisons que pour le cas unidimensionnel, on cherchera l'atome $g_{\gamma_0} \in D2$ qui maximise la valeur absolue du produit scalaire. Sachant que l'atome $g_{\gamma_0} \in D2$ est considéré à variables séparables, le produit scalaire s'écrit alors :

$$\langle I(n,m), g_{\gamma_0}(n,m) \rangle = \sum_n \sum_m I(n,m) \overline{g_{\gamma_0}(n)} g_{\gamma_0}(m) \quad (2.3)$$

Pour reconstruire l'image originale, l'équation (2.4) montre qu'il est nécessaire de stocker à chaque itération les 6 paramètres des atomes sélectionnés et les projections orthogonales associées.

$$I(n,m) = \sum_{m=0}^{p-1} \langle RI^m(n,m), g_{\gamma_0}(n,m) \rangle g_{\gamma_0}(n,m) + RI^p(n,m) \quad (2.4)$$

III.3 Implémentation et atomes de Gabor

Il est évident que pour une implémentation pratique de l'algorithme, on ne peut considérer tous les atomes espace-fréquence du dictionnaire redondant. Il est alors plus judicieux de choisir un sous-dictionnaire.

On utilise les atomes de Gabor pour la génération du dictionnaire espace-fréquence en raison de leurs propriétés particulières (modélisation des mécanismes biologiques de la vision, caractéristiques temps/fréquence, la séparabilité des variables et la simplicité d'implémentation). La fenêtre gaussienne unidimensionnelle à partir de laquelle on construit notre dictionnaire D2 s'écrit :

$$gf1(x) = 2^{1/4} e^{-\pi x^2} \quad (2.5)$$

Nous supposons que l'image est réelle de N lignes et N colonnes, pour avoir à une échelle quelconque, un signal discret et périodique, la fonction fenêtre est uniformément échantillonnée et périodisée sur les N points :

$$gf1_s(n) = \frac{K_s}{\sqrt{s}} \sum_{p=-\infty}^{+\infty} gf1\left(\frac{n-pN}{s}\right) \quad (2.6)$$

L'atome bidimensionnel s'écrira alors :

$$gf2(x) = 2^{1/4} e^{-\pi(x^2)} e^{-\pi(y^2)} = g(x) * g(y) \quad (2.7)$$

$$gf2_s(n,m) = \frac{K_{sx} K_{sy}}{\sqrt{s_x} \sqrt{s_y}} \sum_{p=-\infty}^{+\infty} g\left(\frac{n-pN}{s_x}\right) * g\left(\frac{m-pN}{s_y}\right) \quad (2.8)$$

Les constantes K_{sx} et K_{sy} normalisent la norme discrète $g_s(n,m)$. Pour tout entier $0 \leq ux < N$, $0 \leq uy < N$ et $0 \leq kx < N$, $0 \leq ky < N$,

$$\text{on note } \gamma = (sx, sy, ux, uy, \frac{2\pi kx}{8 * N}, \frac{2\pi ky}{8 * N})$$

et on définit l'atome de Gabor discret :

$$g_s(n,m) = g_{sx}(n-ux) g_{sy}(m-uy) e^{i(2\pi kx/8N)} e^{i(2\pi ky/8N)} \quad (2.9)$$

Pour favoriser la séparabilité des variables, on considère que :

$$\begin{aligned} \|g_s(n,m)\| &= 1 \quad \text{si } sx = sy & (2.10) \\ \|g_s(n,m)\| &< 1 \quad \text{lorsque } sx \neq sy \end{aligned}$$

A partir de ce type de normalisation, on déduit des atomes unidimensionnels applicables au cas 2-D.

Pour restreindre le nombre d'échelles utilisé on considère que :

$$sx = 2^{jx} \text{ et } sy = 2^{jy}, \quad jx \text{ et } jy = 1, \dots, 4, 5$$

(valeur déduite pratiquement).

Les translations ux et uy varient entre 0 et N-1, tandis que pour les modulations en fréquence, on

$$\text{pose que } \xi_x = \frac{2\pi kx}{8N} \quad \text{et} \quad \xi_y = \frac{2\pi ky}{8N}$$

avec kx et $ky = 0, 1, \dots, 7$.

Un exemple d'atome 2-D est illustré par la figure 1.

L'application de cet algorithme sur une image du type femme de taille NxN octets convertit l'information à stocker en 6*p paramètres d'atomes (p étant le nombre d'itérations), p produits internes associés et l'image résiduelle RI_p. Il est évident que le taux de compression dépend du nombre d'itérations qui est lié au taux de décroissance de la norme de l'image résiduelle.

En pratique, nous avons remarqué que la norme de l'image résiduelle décroît exponentiellement, et une norme très réduite n'est obtenue qu'après un nombre important d'itérations. Pour pallier à ce problème, nous avons choisi comme critère d'arrêt non pas une norme négligeable mais une variation négligeable de la norme de l'image résiduelle.



$$\frac{\|RI^i - RI^{i-1}\|}{\|RI^i\|} < \varepsilon \quad (2.11)$$

On remarquera qu'à ce stade de la décomposition l'image résiduelle comporte essentiellement les informations hautes fréquences (contours des objets image fig 2.b).

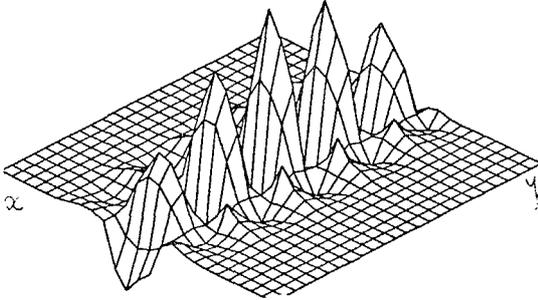


Figure 1 : Atome 2-D avec $j_x = 2$ et $j_y = 4$,
 $k_x = 1$ $k_y = 2$.

III.4 Codage et résultats

Les informations à stocker sont de trois types:

- * L'image résiduelle constituée d'informations contours,
- * les paramètres des atomes sélectionnés,
- * les projections orthogonales associées,

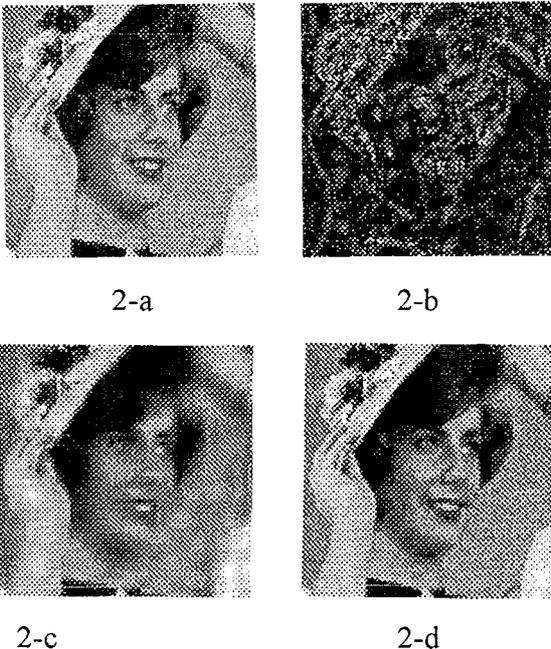


Figure 2: a) Image originale
b) image résidu contrastée
c) image reconstruite sans l'image résiduelle
d) image reconstruite à partir des fichiers codés.

Du fait de la nature différente des informations, nous prévoyons 3 types de codeurs :

- * Un codage par quantification vectorielle de l'image résiduelle [2,4].
- * Un codage de Huffman des paramètres [1],
- * Un codage prédictif de la différence des projections entre deux itérations successives.

La combinaison des trois codeurs offre une bonne qualité de l'image reconstruite pour un taux de compression de 47.31. Les résultats sont illustrés par la figure 2.

IV. CONCLUSION

Nous avons présenté une extension 2-D d'un algorithme de représentation très flexible puisqu'il n'existe aucune limitation dans le choix du dictionnaire. Ce choix a été optimisé pour minimiser le stockage d'information. Les résultats obtenus sont meilleures que ceux obtenus par la décomposition sur des bases orthogonales [7]. L'optimisation du temps de calcul pour la sélection de l'atome le plus adapté reste un problème ouvert.

V. REFERENCES

- [1] R. G. Gallager, "Variations on a theme by Huffman", IEEE Trans. on Information theory, vol. IT-24, N° 6, November 1978.
- [2] R. Gray, "Vector quantization," IEEE ASSP Mas, Apr 1984.
- [3] P.J. Huber, "Projection Pursuit", Ann. Statist., vol. 13, n°2, pp. 435-475
- [4] Y. Linde, A Buzo and R.M. Gray, "An algorithm for vector quantizer design", IEEE trans. Comm. vol COM-28 pp 84-95 Jan 1980.
- [5] Y. Meyer, "Ondelettes et Opérateurs". Paris Hermann, 1990.
- [6] S. Mallat & Z. Zhang, "Matching Pursuits with time- frequency dictionaries", IEEE Trans. on Signal Processing, vol 41, N° 12, December 1993.
- [7] A. Serir & B. Sansal, "Satellite images data compression using multiresolution analysis based on optimal edge detection", RSSS'94 July San Diego U.S.A
- [8] O. Rioul & M. Vetterli, "Wavelets ans signal Processing", IEEE Signal Processing Magazine, October 1991.