

Représentations parcimonieuses par un dictionnaire à structure adaptative : application au codage d'images satellitaires

Jérémy AGHAEI MAZAHERI, Christine GUILLEMOT, Claude LABIT

INRIA Rennes

Campus universitaire de Beaulieu, 35042 Rennes Cedex, France *

jeremy.agmaei_mazaheri@inria.fr, christine.guillemot@inria.fr
claude.labit@inria.fr

Résumé – Dans un contexte de codage d'images satellitaires, une structure de dictionnaire adaptative est apprise afin de représenter les données de façon parcimonieuse tout en optimisant le compromis entre l'erreur d'approximation et le coût de codage. Cette nouvelle structure de dictionnaire, dite en "cerf-volant", s'adapte aux données d'apprentissage et peut être considérée comme une évolution d'une structure en arbre. Une méthode de sélection des atomes au sein de cette structure, appelée Sélection Adaptative des Atomes par Niveau (SAAN), est également présentée. Elle permet de sélectionner un ou plusieurs atomes par niveau selon un critère de qualité de la représentation. Un gain en débit-distorsion est observé pour la structure en "cerf-volant" par rapport aux dictionnaires dits "plats" (K-SVD, DCT) et par rapport à une structure arborescente, dont les niveaux sont de plus en plus vides et donc de moins en moins efficaces.

Abstract – In a context of satellite image coding, an adaptive dictionary structure is learned to represent the data as sparse decompositions while optimizing the compromise between error of approximation and coding cost. This novel dictionary structure, structured as a "kite", adapts itself to the learning data and can be considered as an evolution of a tree structure. A method to select the atoms among this structure, called Adaptive Sparse Coding (AdSC), is also presented. It selects one or more atoms per level on a criteria of quality of representation. A rate-distortion gain is observed for the "kite" structure compared to the "flat" dictionaries (K-SVD, DCT) and to the tree structure, whose levels are more and more empty and so less and less efficient.

1 Introduction

Représenter un signal $y \in \mathbb{R}^n$ de façon parcimonieuse consiste à décomposer ce dernier en une combinaison linéaire de quelques colonnes, appelées atomes, d'un dictionnaire $D \in \mathbb{R}^{n \times K}$ de K atomes. Le signal peut ainsi être approximé par $y \approx Dx$, où $x \in \mathbb{R}^K$ est parcimonieux, i.e. contenant seulement quelques coefficients non nuls. Le dictionnaire D peut être soit prédéfini, soit appris sur une base d'apprentissage.

Utiliser les représentations parcimonieuses dans un contexte de compression revient à coder des paires indice-coefficient et a été abordé dans [1, 2, 3], mais avec des méthodes d'apprentissage différentes. Les auteurs dans [1, 2] utilisent un dictionnaire "plat" appris via l'algorithme K-SVD, tandis que dans [3], un dictionnaire structuré en arbre est utilisé. L'idée d'utiliser un dictionnaire structuré adapté pour la compression a été explorée avec la notion d'ITD (Iteration-Tuned Dictionary) [4], où un dictionnaire différent est utilisé pour chaque itération de l'algorithme d'approximation de type poursuite, tel que OMP [5]. Une application simple de cette idée est une structure de dictionnaire arborescente et scalable, telle qu'elle est proposée dans [3] sous le nom Tree K-SVD. De haut en bas, chaque niveau correspond alors à une itération de l'algorithme de poursuite. Chaque dictionnaire à un niveau est appris

via l'algorithme K-SVD sur des résidus du niveau supérieur et donc adapté à une certaine classe de résidus. Mais lorsque le nombre de niveaux de l'arbre augmente, le nombre cumulé d'atomes appris augmente rapidement, ce qui peut conduire à un problème d'"overfitting". De plus, les dictionnaires deviennent de plus en plus incomplets, perdant ainsi en efficacité. Les résidus ne sont quant à eux plus structurés et s'apparentent davantage à du bruit. Il est donc nécessaire d'apprendre une structure qui s'adapte aux données d'entrée tout en limitant le nombre d'atomes afin d'éviter l'"overfitting".

C'est pourquoi nous proposons une structure dite en "cerf-volant", où l'arbre est refermé à un certain niveau pour n'apprendre ensuite qu'un seul dictionnaire par niveau, plus général et complet. Cette structure permet d'une part l'apprentissage de davantage de niveaux que la structure en arbre tout en gardant une taille de dictionnaire raisonnable, et d'autre part l'apprentissage de davantage d'atomes qu'un dictionnaire "plat" avec un coût de codage des indices équivalent.

Cette structure en "cerf-volant" est comparée au dictionnaire appris avec K-SVD et au dictionnaire prédéfini DCT, ainsi qu'à la structure en arbre Tree K-SVD, et les surpasse en matière de performances débit-distorsion.

La section 2 présente la structure adaptative en "cerf-volant" après avoir rappelé les concepts de la structure arborescente Tree K-SVD. La méthode de Sélection Adaptative des Atomes

*Ces travaux ont été soutenus par EADS Astrium.

par niveau (SAAN) au sein de la structure est expliquée en section 3. La structure adaptative est comparée aux autres structures en section 4. Enfin, nous concluons cet article et évoquons des perspectives dans la section 5.

2 De la structure en arbre à la structure en "cerf-volant"

2.1 La structure en arbre classique

La structure en arbre possède 1 dictionnaire au premier niveau, K au second, K^2 au troisième, etc. (Fig.1). Nous choisissons pour cette étude la méthode d'apprentissage de dictionnaires en arbre nommée Tree K-SVD [3]. Chaque niveau est appris sur les résidus du niveau supérieur et chaque dictionnaire est adapté à une classe de résidus. Chaque dictionnaire au sein de la structure est appris via l'algorithme K-SVD [1], avec une parcimonie fixée à un atome. L'unique dictionnaire au premier niveau est appris sur l'ensemble des données d'apprentissage. Puis ces données sont approximées par un atome de ce dictionnaire grâce à l'algorithme OMP (Orthogonal Matching Pursuit) [5]. Des résidus r_2 sont ainsi calculés puis partitionnés en K résidus, selon l'atome utilisé lors de l'approximation par le dictionnaire au niveau 1. Chaque partition de résidus $r_{2,k}$, $k \in [1, \dots, K]$, permet alors d'apprendre un dictionnaire du second niveau. La même procédure est ensuite appliquée pour les niveaux suivants. Un dictionnaire peut ainsi être incomplet voire vide si aucun vecteur d'apprentissage n'utilise l'atome correspondant au niveau supérieur. Le dictionnaire incomplet correspond alors à une simple copie avec normalisation des résidus et l'apprentissage de cette branche est arrêté.

Composée de petits dictionnaires, cette structure permet un codage efficace des indices des atomes sélectionnés pour la représentation ainsi qu'une recherche rapide de ceux-ci lors de la décomposition d'un signal (du même ordre qu'avec un dictionnaire "plat" de K atomes). Elle est de plus scalable car, apprise pour un nombre de niveaux donné, elle peut être utilisée pour toute approximation avec un nombre d'atomes inférieur ou égal au nombre de niveaux.

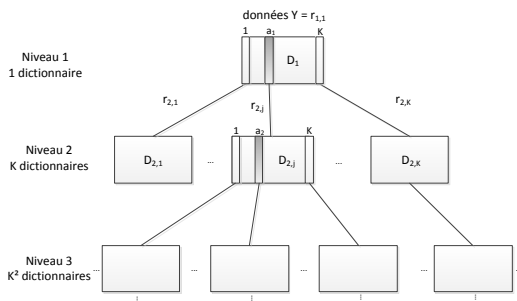


FIGURE 1 – Structure arborescente : Tree K-SVD

2.2 La structure en "cerf-volant" proposée

La structure en "cerf-volant" peut être vue comme une évolution de la structure en arbre. Elle est identique à un arbre sur les premiers niveaux mais à partir d'un niveau donné, il n'y a plus qu'un dictionnaire par niveau pour former la traîne du "cerf-volant" (Fig.2). Elle présente les mêmes avantages que

la structure arborescente. Son intérêt vient du fait de refermer l'arbre. En effet, développer l'arbre trop profondément provoque une forte augmentation du nombre d'atomes conduisant à un "overfitting", les niveaux plus profonds de l'arbre étant trop spécifiques aux données d'apprentissage. En outre, plus le niveau de l'arbre est profond et plus les dictionnaires sont incomplets voire vides et donc de moins en moins performants. C'est pourquoi pour le "cerf-volant", l'arbre est refermé sur un critère de taux de remplissage des dictionnaires du niveau, avant d'atteindre un niveau trop vide et donc trop peu efficace, pour n'apprendre ensuite qu'un seul dictionnaire par niveau, plus généraliste mais complet.

Lors de l'apprentissage, la structure en "cerf-volant" s'adapte aux données d'apprentissage. Si des dictionnaires sont incomplets dans la partie en arbre, l'apprentissage de la branche est arrêté et le dictionnaire a comme fils le premier dictionnaire de la traîne du "cerf-volant". Lorsque l'arbre a été développé jusqu'au niveau souhaité, les branches des dictionnaires complets sont arrêtées à leur tour. Les résidus de ces dictionnaires complets sont fusionnés pour apprendre un unique dictionnaire au niveau suivant, qui constitue le premier dictionnaire de la traîne. Dès lors, un seul dictionnaire par niveau est appris sur les résidus du niveau supérieur.

Une fois la structure en "cerf-volant" apprise, elle est utilisée pour approximer un signal comme une combinaison linéaire et parcimonieuse d'atomes de cette structure. Un atome est sélectionné par niveau du "cerf-volant", jusqu'à la parcimonie souhaitée, à l'aide de l'algorithme OMP [5]. Au sein de la partie en arbre, le choix du dictionnaire où chercher l'atome à un niveau donné est indiqué par l'indice de l'atome choisi au niveau supérieur.

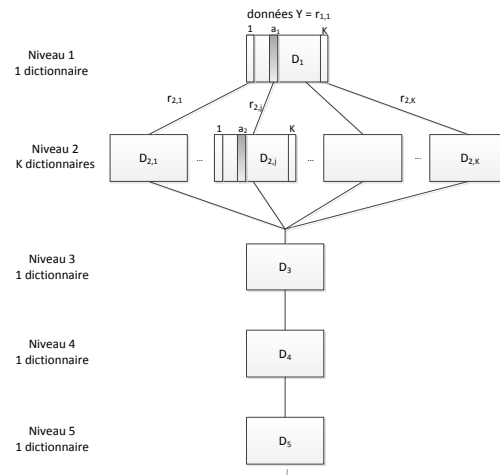


FIGURE 2 – Structure en "cerf-volant" (exemple de fermeture au niveau 3)

3 Sélection Adaptative des Atomes par Niveau (SAAN)

La sélection adaptative des atomes par niveau est une méthode de sélection des atomes initialement présentée dans [3]

pour la structure en arbre, mais qui peut être adaptée pour fonctionner de la même façon au sein de la structure en "cerf-volant". Cette méthode est une alternative à la méthode précédemment présentée consistant à sélectionner un atome par niveau. Elle permet de sélectionner plus d'un atome par niveau, ce qui permet d'utiliser davantage d'atomes dans la représentation à partir de la même structure de dictionnaire. A chaque niveau, une fois qu'un premier atome a été sélectionné, cette méthode offre le choix soit de rester au même niveau pour y sélectionner un atome supplémentaire au sein du même dictionnaire, soit de descendre au niveau suivant. Pour réaliser ce choix, les deux atomes les plus corrélés au vecteur de résidus à approximer, un au niveau courant et un au niveau suivant, sont déterminés. L'atome minimisant l'énergie des résidus est ensuite conservé dans la représentation. De cette façon, le nombre d'atomes sélectionnés par niveau s'adapte automatiquement afin de minimiser la distorsion.

4 Expérimentations sur des images satellitaires

Les différentes structures de dictionnaires sont apprises sur une base d'apprentissage de 654 688 vecteurs provenant de blocs 8x8 de 13 images satellitaires représentant des scènes variées (ville, campagne, mer, desert, ...). L'image New York (Fig.3) est utilisée pour les tests. C'est une image de 2400x2400 pixels, comportant aussi bien des zones uniformes que des zones texturées, ce qui représente 90 000 vecteurs tests. Chaque dictionnaire au sein des différentes structures est initialisé par le dictionnaire DCT. Les dictionnaires de référence sont le dictionnaire appris par l'algorithme K-SVD [1] en 50 itérations, ainsi que le dictionnaire prédéfini DCT, ajouté à la comparaison afin de montrer le gain en qualité apporté par l'apprentissage. Pour Tree K-SVD et la structure en "cerf-volant", 50 itérations du K-SVD sont réalisées pour apprendre le dictionnaire au premier niveau, puis 10 itérations sont suffisantes pour apprendre les autres niveaux. Dans un contexte de compression, nous avons choisi de nous placer dans un cas où le codage des indices est équivalent pour toutes les structures. C'est pourquoi les dictionnaires de référence "plats" ont une taille de K atomes, tandis que chaque dictionnaire des structures en arbre et en "cerf-volant" possède K atomes.

4.1 Analyse en fonction de la parcimonie

Afin de comparer la structure en "cerf-volant" avec les dictionnaires de référence et la structure en arbre, l'image test est reconstruite pour différentes parcimonies (Fig. 4 et 5). Le "cerf-volant" est ici refermé à partir du troisième niveau, car le troisième niveau aurait été trop incomplet s'il avait été développé.

On note tout d'abord que le dictionnaire pré-défini DCT est nettement en-dessous en terme de qualité de reconstruction en comparaison des dictionnaires appris. La structure en arbre Tree K-SVD est au début performante mais stagne lorsque les atomes sont sélectionnés plus profondément dans l'arbre. Étant donné que les branches de l'arbre sont arrêtées lorsque qu'un dictionnaire est incomplet, peu de branches atteignent 10 niveaux et

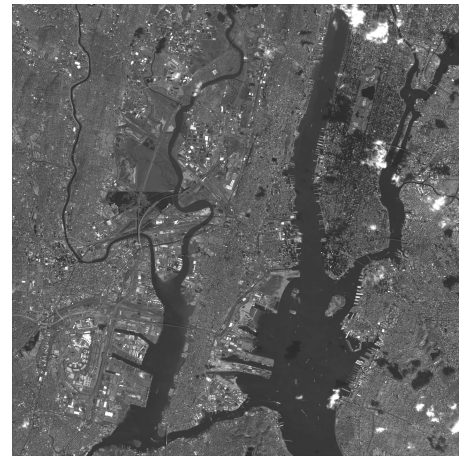


FIGURE 3 – Image test : New York (2400x2400)

en sélectionnant un seul atome par niveau, on ne peut souvent pas atteindre le niveau de parcimonie souhaité. Le fait de pouvoir sélectionner plusieurs atomes par niveau grâce à la méthode SAAN (Sélection Adaptative des Atomes par Niveau) permet de sélectionner le nombre d'atomes souhaité mais on se retrouve souvent bloqué sur un dictionnaire sans fils et y sélectionner trop d'atomes affecte la qualité de la décomposition. C'est pour cela que les performances en PSNR de K-SVD finissent par rejoindre et dépasser celles de Tree K-SVD SAAN lorsque davantage d'atomes sont sélectionnés.

La structure en "cerf-volant" permet de compenser les faiblesses de la structure en arbre en refermant l'arbre avant d'atteindre un niveau trop vide et en le prolongeant par une traîne de dictionnaires. Ainsi, les résultats de la structure en "cerf-volant" sont similaires à ceux de la structure en arbre au début puis les dépassent nettement lorsque le nombre d'atomes dans la représentation augmente (+0.88dB par rapport à K-SVD, +1.51dB par rapport à Tree K-SVD SAAN, pour 10 atomes et $K=64$ (Fig.4)). Le fait de s'autoriser à sélectionner plus d'un atome par niveau au sein de la structure en "cerf-volant" apporte une légère amélioration des résultats (+0.28dB).

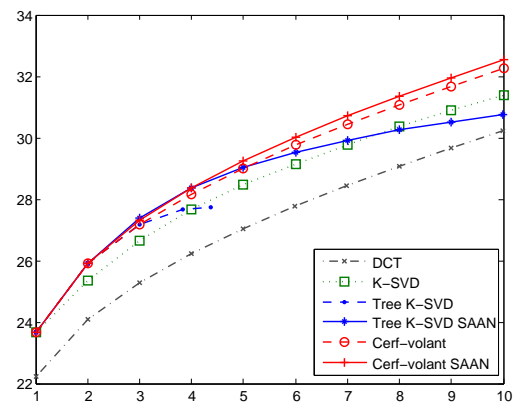


FIGURE 4 – Résultats PSNR-parcimonie ($K=64$)

4.2 Codage des images satellitaires

Dans un contexte de compression, les indices des atomes choisis ainsi que les coefficients associés doivent être codés

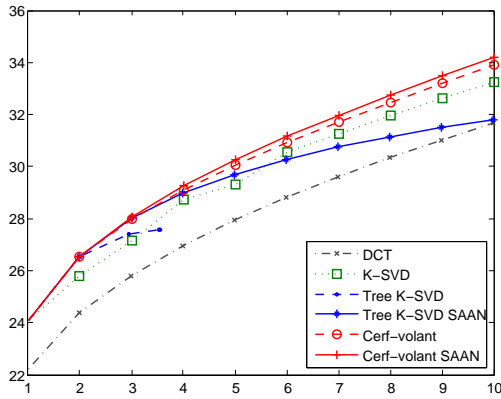


FIGURE 5 – Résultats PSNR-parcimonie (K=256)

pour chaque bloc de l'image. Les coefficients sont d'abord quantifiés par un quantificateur uniforme à zone morte. Les indices de quantification sont alors placés sous forme d'une séquence, un code "End Of Block" séparant chaque bloc. La séquence ainsi formée est ensuite encodée à l'aide d'un codeur entropique de Huffman comme cela est fait dans le codeur JPEG, et avec la même table. Enfin, le débit des indices des atomes est ajouté. Pour chaque indice a_i , le débit associé est $R(a_i) = \log_2(K)$.

En faisant varier le pas de quantification, on obtient les différentes courbes PSNR-débit (Fig. 6 et 7). Le quantificateur joue alors le rôle de régulateur du nombre d'atomes utilisés pour chaque bloc. En effet, la zone morte est fonction du pas de quantification et tout coefficient inférieur à la moitié du pas est mis à zéro. Ainsi, selon le pas de quantification et la valeur des coefficients, plus ou moins d'indices et de coefficients sont codés par bloc, en fixant la limite à 10.

La structure en "cerf-volant" donne de meilleurs résultats que les dictionnaires de référence "plats" (+0.86dB par rapport à K-SVD à 2bpp pour K=64) et que la structure en arbre Tree K-SVD (+1.69dB). L'utilisation de la sélection adaptative en parcimonie des atomes par niveau améliore légèrement les résultats à haut débit (+0.30dB à 2bpp), en particulier pour les dictionnaires de 64 atomes (K=64) et malgré un surplus de codage pour indiquer le changement de niveau ou non de 1 bit pour chaque atome.

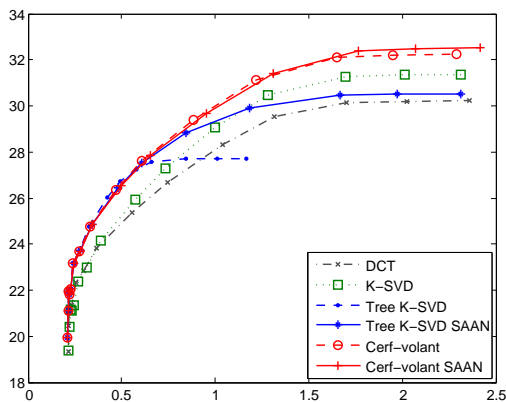


FIGURE 6 – Résultats PSNR-débit (K=64)

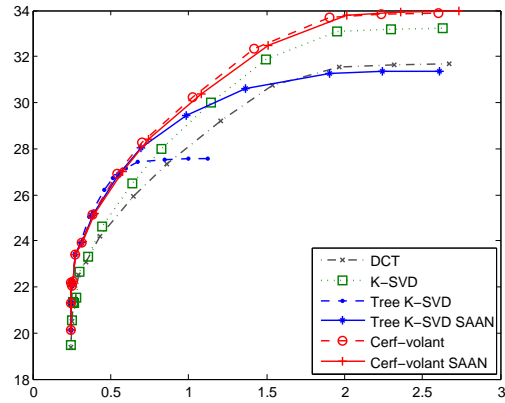


FIGURE 7 – Résultats PSNR-débit (K=256)

5 Conclusion

Nous avons présenté dans cette étude une nouvelle structure de dictionnaire dite en "cerf-volant" permettant de corriger les faiblesses de la structure arborescente. Utilisée dans le cadre des représentations parcimonieuses, notamment avec la méthode de sélection adaptative des atomes par niveau (SAAN), cette structure offre de meilleures performances débit-distorsion que les dictionnaires "plats" K-SVD et DCT et que la structure en arbre Tree K-SVD pour la compression d'images satellitaires. De plus, elle offre comme la structure arborescente les propriétés de codage efficace des indices, de recherche rapide des atomes et de scalabilité en parcimonie.

Nous souhaitons à l'avenir faire évoluer cette structure afin de la rendre totalement adaptative en supprimant la contrainte de fermeture de toutes les branches à un niveau donné. De plus, il serait intéressant d'intégrer cette structure au sein du standard HEVC (High Efficiency Video Coding) en remplaçant la transformée afin de tirer profit de l'efficacité des prédictions et du codage entropique de HEVC.

Références

- [1] M. Aharon, M. Elad et A. Bruckstein. *K-SVD : an algorithm for designing overcomplete dictionaries for sparse representation*. IEEE Trans. on Signal Process., vol.54, no.11, pp.4311-4322, 2006. *K-SVD matlab toolbox* : <http://www.cs.technion.ac.il/~elad/software>
- [2] O. Bryt and M. Elad. *Compression of facial images using the K-SVD algorithm*. Journal of Visual Commun. and Image Represent., vol. 19, no. 4, pp. 270-282, 2008.
- [3] J. Aghaei Mazaheri, C. Guillemot, C. Labit. *Learning a tree-structured dictionary for efficient image representation with adaptive sparse coding*. ICASSP, 2013.
- [4] J. Zepeda, C. Guillemot et E. Kijak. *The iteration-tuned dictionary for sparse representations*. Proc. of the IEEE Int. Workshop on MMSP, 2010.
- [5] Y. C. Pati, R. Rezaifar et P. S. Krishnaprasad. *Orthogonal matching pursuit : Recursive function approximation with applications to wavelet decomposition*. Conference Record of the 27th Asilomar Conf. on Signals, Syst. and Comput., 1993.