Vers une nouvelle génération de schémas de lifting basés sur les réseaux de neurones et application à la compression d'images

T. Dardouri[1], M. Kaaniche[2,3], A. Benazza-Benyahia[4], J.-C. Pesquet[3], G. Dauphin[2]
[1] Novelis, R&D laboratory, 75012, Paris, France
[2] Université Sorbonne Paris Nord, L2TI, UR 3043, F-93430, Villetaneuse, France.
[3] Centre de Vision Numérique, Univ. Paris-Saclay, CentraleSupélec, INRIA, 91190 Gif-sur-Yvette, France
[4] Université de Carthage, SUP'COM, LR11TIC01, COSIM Lab., 2083, El Ghazala, Tunisie
tdardouri@novelis.io, {mounir.kaaniche, gabriel.dauphin}@univ-paris13.fr
benazza.amel@supcom.rnu.tn, jean-christophe.pesquet@centralesupelec.fr

Résumé

Les schémas de lifting ont connu un grand succès en analyse et traitement d'images, et plus particulièrement, en compression d'images. Dans ce contexte, l'optimisation des opérateurs de lifting (i.e., les opérateurs de prédiction et de mise à jour) joue un rôle crucial dans la conception de nouveaux schémas de codage efficaces et adaptés aux images d'entrée. À cet égard, nous proposons, dans cet article, d'explorer le potentiel des réseaux neuronaux dans le contexte de structures de lifting 2D non séparables. Contrairement aux travaux précédents où différents modèles de réseaux neuronaux sont utilisés pour toutes les étapes de prédiction et de mise à jour, notre conception repose sur un nouveau modèle de réseau de neurones convolutif multi-tâches qui prend en compte les similitudes entre deux étapes de prédiction. Les simulations effectuées sur deux bases d'images usuelles montrent l'intérêt de l'architecture proposée pour la compression d'images.

Mots clefs

Transformées en ondelettes, schémas de lifting adaptatifs, optimisation, réseaux de neurones, compression d'images.

1 Introduction

Les ondelettes ont suscité beaucoup d'intérêt dans la comunauté de traitement du signal et d'images grâce à leurs bonnes propriétés de scalabilité en qualité et en résolution ainsi que leur capacité d'offrir une analyse multi-échelle des données. Par exemple, elles ont été largement adoptées dans diverses tâches de traitement [1, 2, 3] de différents types de contenus multimédias tels que les images 2D et 3D, la vidéo, l'audio, etc [4, 5].

Pour produire les coefficients d'ondelettes, le schéma de lifting (*Lifting Scheme* (LS)) s'est avéré être un outil efficace permettant une mise en oeuvre rapide et une reconstruction parfaite du signal d'entrée. Ainsi, un schéma de lifting conventionnel repose sur une étape de prédiction suivie d'une étape de mise à jour permettant de générer respectivement les coefficients de détails et d'approximation. Alors que la norme de codage d'images JPEG2000 utilise certains filtres prédéfinis avec des poids fixes, de nombreux efforts ont été déployés pour mieux adapter ces filtres au contenu des données d'entrée et améliorer l'efficacité des codeurs basés sur le schéma de lifting. Pour cela, différentes techniques d'optimisation ont été développées pour la conception des opérateurs (ou filtres) de prédiction et de mise à jour. La plupart de ces techniques ont été consacrées au filtre de prédiction, qui est souvent optimisé en minimisant un certain critère défini sur les coefficients de détail. Les critères utilisés comprennent les normes ℓ_2 [6] et ℓ_1[7] ainsi que l'entropie [8, 9]. Cependant, l'optimisation de l'opérateur de mise à jour est plus difficile, et a été peu explorée dans la littérature [6, 10, 11].

En plus de ces approches d'optimisation traditionnelles, certaines méthodes mettant en jeu les réseaux neuronaux ont été récemment proposées. En effet, les tâches de prédiction et de mise à jour ont été réalisées à l'aide de réseaux de neurones convolutifs (CNN) [12, 13] et de réseaux de neurones entièrement connectés (FCNN) [14, 15]. De tels schémas peuvent être considérés comme une première catégorie de méthodes de compression d'images basées sur l'apprentissage profond, une autre catégorie de méthodes, inspirées par les auto-encodeurs, a également été présentée dans la littérature. L'architecture commune à la plupart de ces méthodes comprend trois modules : (1) transformée d'analyse non linéaire, (2) quantification et codage entropique, et (3) transformée de synthèse non linéaire [16, 17, 18]. Notons que d'autres méthodes de codage prédictif (intra) ont également été proposées [19, 20].

Motivé par les nombreux avantages des représentations issues des schémas de lifting et les résultats prometteurs obtenus par les modèles FCNN [14, 15], l'objectif de cet article est d'exploiter davantage les réseaux neuronaux dans les systèmes de codage d'images basés sur les schémas de lifting.

Le reste de l'article est organisé comme suit. La section 2 rappelle le concept des schémas de lifting basés sur les réseaux de neurones. Ensuite, la section 3 décrit l'architecture proposée. Enfin, les résultats expérimentaux sont pré-

sentés dans la section 4 et des conclusions sont tirées dans la section 5.

2 Travaux connexes

Récemment, nous avons proposé de nouvelles structures de lifting reposant sur les réseaux de neurones [14, 15]. Pour cela, une structure de lifting 2D non séparable, qui présente l'avantage de réduire le nombre d'étapes de lifting par rapport à la décomposition séparable, a été adoptée. Cette structure de lifting est composée de trois étapes de prédiction suivi d'une étape de mise à jour [11]. Plus précisément, la structure 2D, illustrée dans la Fig. 1, consiste à décomposer une image d'entrée $X_j(m,n)$ en quatre composantes polyphases : $\mathbf{X}_{0,j}(m,n) = \mathbf{X}_j(2m,2n)$, $\mathbf{X}_{1,j}(m,n) = \mathbf{X}_j(2m,2n+1)$, $\mathbf{X}_{2,j}(m,n) = \mathbf{X}_j(2m+1,2n)$ et $\mathbf{X}_{3,j}(m,n) = \mathbf{X}_j(2m+1,2n+1)$. Ensuite, trois étapes de prédiction sont appliquées sur $\mathbf{X}_{3,j}(m,n)$, $\mathbf{X}_{2,j}(m,n)$ et $\mathbf{X}_{1,j}(m,n)$ afin de générer respectivement les coefficients de détails diagonaux $\mathbf{X}_{j+1}^{(HH)}(m,n)$, verticaux $\mathbf{X}_{j+1}^{(LH)}(m,n)$ et horizontaux $\mathbf{X}_{j+1}^{(HL)}(m,n)$. Enfin, une étape de mise à jour est appliquée à $\mathbf{X}_{0,j}(m,n)$ afin de produire les coefficients d'approximation $\mathbf{X}_{j+1}(m,n)$. Les étapes de prédiction et de mise à jour sont effectuées en utilisant quatre modèles FCNN désignés par $f_j^{(o)}$, avec $o \in \{HH, LH, HL, LL\}$.

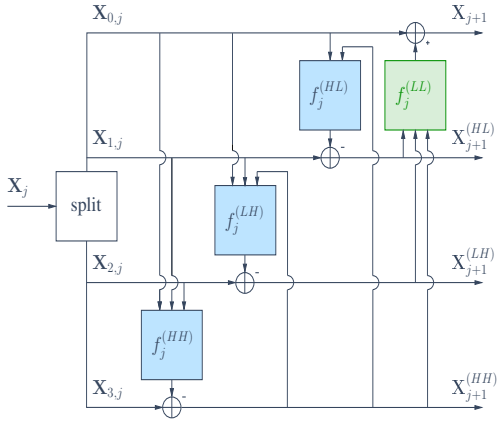

FIGURE 1 – *Structure d'analyse du FCNN-LS [14].*

Notons ici que les différents modèles FCNN (de prédiction et de mise à jour) peuvent être appris séparément à chaque niveau de résolution j [14]. Il est également possible d'optimiser conjointement les différents réseaux FCNN [15].

3 Nouvelle architecture basée sur un réseau CNN multi-tâches

3.1 Motivation

Une limitation majeure du modèle FCNN utilisé est qu'il ne tient pas compte des fortes corrélations locales dans l'image d'entrée. Pour surmonter ce problème et améliorer davantage les performances de prédiction, nous proposons d'abord de recourir à un réseau de neurones convolutionnels (CNN). De plus, dans les approches précédemment proposées, quatre modèles $f_j^{(o)}$ sont utilisés pour générer la sous-bande d'approximation ainsi que les trois sous-bandes de détails. Cependant, la Fig. 1 montre qu'une fois les coefficients de détails diagonaux sont générés, les deux étapes de prédiction suivantes peuvent être effectuées simultanément pour produire les coefficients de détails verticaux et horizontaux. Il est important de souligner que ces deux étapes de prédiction sont assez similaires et partagent certains entrées. Par conséquent, il devient plus intéressant de concevoir un nouveau modèle CNN multi-tâches (désigné dans la suite par MT-CNN) pour réaliser conjointement ces deux étapes de prédiction.

3.2 Modèles et méthodes d'apprentissage

La structure d'analyse de l'architecture de lifting, illustrée dans la Fig. 2, est composée des trois modèles CNN suivants. Le premier modèle, noté $C_j^{(HH)}$, correspond à la première étape de prédiction qui vise à générer les coefficients de détails diagonaux $\mathbf{X}_{j+1}^{(HH)}$. Le deuxième, désigné par $C_j^{(HL,LH)}$, effectue simultanément les deux tâches de prédiction permettant de générer les coefficients de détails verticaux $\mathbf{X}_{j+1}^{(LH)}$ et horizontaux $\mathbf{X}_{j+1}^{(HL)}$. Enfin, le dernier modèle, désigné par $C_j^{(LL)}$, concerne l'étape de mise à jour pour produire les coefficients d'approximation $\mathbf{X}_{j+1}$.

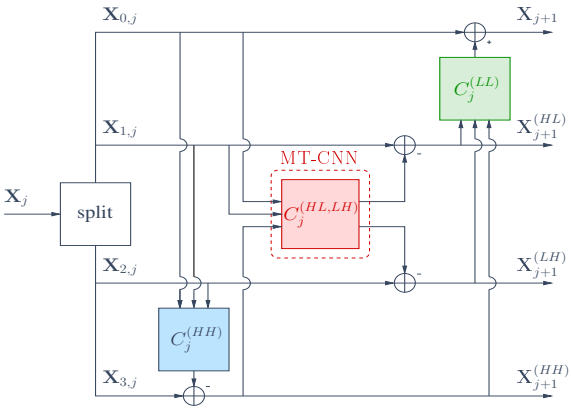

FIGURE 2 – *Structure d'analyse de l'architecture de lifting à base d'un réseau CNN multi-tâches.*

Les modèles CNN impliqués et leurs stratégies d'apprentissage sont décrits par la suite.

Étape de prédiction diagonale basée sur un CNN : De manière similaire à l'étape de prédiction basée sur le FCNN, la première étape de prédiction reposant sur le CNN vise à obtenir les coefficients de détails diagonaux en calculant la différence entre les composantes polyphases originales et celles prédites. La structure de l'architecture CNN retenue comprend cinq couches de convolution utilisant respectivement les nombres de canaux suivants : 32, 16, 16, 32 et 1. La taille des noyaux de la première couche est de 7×7, tandis que celles associées aux couches suivantes sont de 3×3. Nous utilisons également la fonction d'activation Gaussian Error Linear Unit (GELU). Le

modèle CNN retenu dépend d'un vecteur de paramètres $\mathbf{\Theta}_j^{(HH)}$ qui est appris en minimisant un critère d'erreur quadratique moyenne.

Étapes de prédiction horizontale et verticale basées sur un CNN multi-tâches : Une fois les coefficients de détails diagonaux générés, on peut procéder aux deuxième et troisième étapes de prédiction pour produire simultanément les coefficients de détails verticaux et horizontaux. En raison de la similitude entre ces deux étapes, un nouveau modèle CNN multi-tâches est proposé pour ces étapes de prédiction. Le modèle MT-CNN proposé, noté par $C_j^{(HL,LH)}$ dans la Fig. 2, est construit en utilisant le schéma de partage des paramètres [21] comme indiqué dans la Fig. 3.

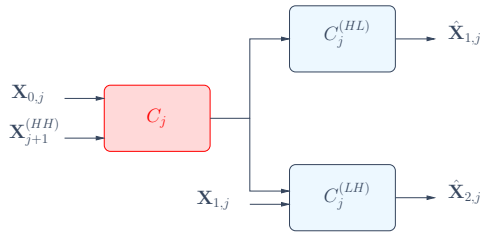

FIGURE 3 – *Schéma de partage des paramètres pour le modèle MT-CNN proposé* $C_j^{(HL,LH)}$.

Plus précisément, le modèle MT-CNN se compose d'un modèle CNN partagé suivi de deux modèles spécifiques à chaque tâche. En fait, dans une structure NSLS typique (comme illustré dans la Fig. 1), le calcul des coefficients de détail horizontaux et verticaux nécessite deux signaux de référence communs $\mathbf{X}_{0,j}$ et $\mathbf{X}_{j+1}^{(HH)}$. Ces deux canaux constitueront d'abord les entrées du modèle partagé noté C_j. Ensuite, la sortie de C_j sert de référence dans les deux modèles CNN spécifiques à chaque tâche illustrés dans les branches supérieure et inférieure du réseau, et désignés respectivement par $C_j^{(HL)}$ et $C_j^{(LH)}$.

Sur la Fig. 1, en plus des deux canaux d'entrée communs utilisés par le modèle CNN partagé, la génération des coefficients de détail verticaux $\mathbf{X}_{j+1}^{(LH)}$ utilise un troisième signal de référence correspondant à $\mathbf{X}_{1,j}$. Pour cette raison, un canal supplémentaire $\mathbf{X}_{1,j}$ a été inclus en tant qu'entrée du modèle $C_j^{(LH)}$. Enfin, les couches de sortie des deux modèles spécifiques à chaque tâche $C_j^{(HL)}$ et $C_j^{(LH)}$ permettent de générer les sous-bandes de détails horizontaux $\mathbf{X}_{j+1}^{(HL)}$ et verticaux $\mathbf{X}_{j+1}^{(LH)}$. Pour apprendre le modèle conjoint $C_j^{(HL,LH)}$, une approche d'apprentissage multi-tâches est adoptée. En effet, le modèle est appris en optimisant la somme des fonctions de coût spécifiques à chaque tâche, reposant sur un critère d'erreur quadratique moyenne.

Étape de mise à jour basée sur un CNN : Après les étapes de prédiction, une étape de mise à jour basée sur un réseau CNN est finalement effectuée pour calculer les coefficients d'approximation $\mathbf{X}_{j+1}$. En effet, les sous-bandes de détails générées constitueront les trois canaux d'entrée du modèle CNN de mise à jour $C_j^{(LL)}$. Son canal de sortie sera utilisé pour produire les coefficients d'approximation $\mathbf{X}_{j+1}$. Il convient de noter ici que la structure utilisée pour $C_j^{(LL)}$ est similaire à celle de $C_j^{(HH)}$. Par conséquent, elle est entraînée de la même manière.

4 Résultats expérimentaux

L'architecture proposée a été entraînée en utilisant la base de données Flickr composée de 8,000 images de différentes tailles [1]. Deux bases de données de test ont été considérées : Kodak [2] (composée de 24 images de taille 768×512) et Tecnick [3] [22] (où 30 images, de taille 1200×1200, ont été sélectionnées). Notre approche a été comparée à JPEG2000 ainsi que d'autres méthodes d'état de l'art basées sur les réseaux de neurones. Plus précisément, les tableaux 1 et 2 montrent les gains de notre méthode par rapport à CNN-LS[12] et FCNN-LS [14], en terme de métrique de Bjøntegaard. Notons ici que la qualité de reconstruction est évaluée en utilisant la métrique perceptuelle PieAPP [23], qui s'est avérée plus pertinente que les métriques traditionnelles telles que le PSNR et le SSIM. Les résultats obtenus à bas et moyens débits (aux points $\{0.07, 0.1, 0.15, 0.2\}$ et $\{0.2, 0.25, 0.3, 0.4\}$ bpp, respectivement) montrent des gains significatifs de l'architecture proposée, en terme de réduction de débit, par rapport à FCNN-LS [14] and CNN-LS [12].

	gain de débit (in %)		**gain de PieAPP**	
Bases	**bas**	**moyens**	**bas**	**moyens**
Kodak	-13.80	-8.74	-0.11	-0.06
Tecnick	-19.87	-14.41	-0.14	-0.07

TABLEAU 1 – *Gain du MT-CNN-LS par rapport à FCNN-LS [14] en terme de métrique de Bjøntegaard.*

	gain de débit (in %)		**gain de PieAPP**	
Bases	**bas**	**moyens**	**bas**	**moyens**
Kodak	-57.01	-26.57	-0.55	-0.23
Tecnick	-51.82	-19.40	-0.44	-0.12

TABLEAU 2 – *Gain du MT-CNN-LS par rapport à CNN-LS [12] en terme de métrique de Bjøntegaard.*

5 Conclusion

Dans cet article, une nouvelle architecture de schéma de lifting basée sur un réseau CNN multi-tâches a été proposée. Le potentiel de cette architecture a été démontré dans le contexte de la compression d'images, et mériterait d'être exploré pour d'autres tâches d'analyse d'images.

1. https ://www.kaggle.com/datasets/adityajn105/flickr8k
2. https ://www.r0k.us/graphics/kodak/
3. https ://testimages.org/

Références

[1] J.-H. Jacobsen, A. W. M. Smeulders, et E. Oyallon. i-RevNet : Deep invertible networks. Dans *International Conference on Learning Representations*, pages 1–11, Vancouver, Canada, May 2018.

[2] J. J. Huang et P. L. Dragotti. LINN : Lifting inspired invertible neural network for image denoising. Dans *European Signal and Image Processing Conference*, pages 1–5, Dublin, Ireland, September 2021.

[3] T.-S. Nguyen, M. Luong, M. Kaaniche, L. H. Ngo, et A. Beghdadi. A novel multi-branch wavelet neural network for sparse representation based object classification. *Pattern Recognition*, 135 :109155, 2023.

[4] Y. Xing, M. Kaaniche, B. Pesquet-Popescu, et F. Dufaux. Adaptive non separable vector lifting scheme for digital holographic data compression. *Applied Optics*, 54(1) :A98–A109, January 2015.

[5] E. Martinez-Enriquez, J. Cid-Sueiro, F. Diaz de Maria, et A. Ortega. Directional transforms for video coding based on lifting on graphs. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(4) :933–946, November 2016.

[6] A. Gouze, M. Antonini, M. Barlaud, et B. Macq. Design of signal-adapted multidimensional lifting schemes for lossy coding. *IEEE Transactions on Image Processing*, 13(12) :1589–1603, December 2004.

[7] M. Kaaniche, B. Pesquet-Popescu, A. Benazza-Benyahia, et J.-C. Pesquet. Adaptive lifting scheme with sparse criteria for image coding. *EURASIP Journal on Advances in Signal Processing : Special Issue on New Image and Video Representations Based on Sparsity*, 2012(1) :1–22, January 2012.

[8] J. Solé et P. Salembier. Generalized lifting prediction optimization applied to lossless image compression. *IEEE Signal Processing Letters*, 14(10) :695–698, October 2007.

[9] A. Benazza-Benyahia, J.-C. Pesquet, J. Hattay, et H. Masmoudi. Block-based adaptive vector lifting schemes for multichannel image coding. *EURASIP International Journal of Image and Video Processing*, 2007(1) :10 pages, January 2007.

[10] B. Pesquet-Popescu. *Two-stage adaptive filter bank*. First filling date 1999/07/27, official filling number 99401919.8, European patent number EP1119911, 1999.

[11] M. Kaaniche, A. Benazza-Benyahia, B. Pesquet-Popescu, et J.-C. Pesquet. Non separable lifting scheme with adaptive update step for still and stereo image coding. *Elsevier Signal Processing : Special issue on Advances in Multirate Filter Bank Structures and Multiscale Representations*, 91(12) :2767–2782, January 2011.

[12] H. Ma, D. Liu, R. Xiong, et F. Wu. iWave : CNN-based wavelet-like transform for image compression. *IEEE Transactions on Multimedia*, 22(7) :1667–1697, July 2020.

[13] H. Ma, D. Liu, N. Yan, H. Li, et F. Wu. End-to-end optimized versatile image compression with wavelet-like transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, September 2020.

[14] T. Dardouri, M. Kaaniche, A. Benazza-Benyahia, et J.-C. Pesquet. Dynamic neural network for lossy-to-lossless image coding. *IEEE Transactions on Image Processing*, 31 :569–584, December 2021.

[15] T. Dardouri, M. Kaaniche, A. Benazza-Benyahia, G. Dauphin, et J.-C. Pesquet. Joint learning of fully connected network models in lifting based image coders. *IEEE Transactions on Image Processing*, 33 :134–148, March 2023.

[16] J. Ballé, V. Laparra, et E. P. Simoncelli. End-to-end optimized image compression. Dans *International Conference on Learning Representations*, pages 1–27, Toulon, France, April 2017.

[17] E. Agustsson, M. Tschannen, F. Mentzer, R. Timofte, et V. G. Luc. Generative adversarial networks for extreme learned image compression. Dans *International Conference on Learning Representations*, pages 1–31, New Orleans, LA, USA, May 2019.

[18] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, et N. Johnston. Variational image compression with a scale hyperprior. Dans *International Conference on Learning Representations*, pages 1–47, Vancouver, Canada, May 2018.

[19] J. Li, B. Li, J. Xu, R. Xiong, et W. Gao. Fully connected network-based intra prediction for image coding. *IEEE Transactions on Image Processing*, 27(7) :3236–3247, July 2018.

[20] T. Dumas, A. Roumy, et C. Guillemot. Context-adaptive neural network-based prediction for image compression. *IEEE Transactions on Image Processing*, 29(1) :679–693, August 2019.

[21] S. Vandenhende, S. Georgoulis, W. V. Gansbeke, M. Proesmans, D. Dai, et L. V. Gool. Multi-task learning for dense prediction tasks : A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7) :3614–3633, 2022.

[22] N. Asuni et A. Giachetti. Test images : A large data archive for display and algorithm testing. *Journal of Graphics Tools*, 17(4) :113–125, February 2015.

[23] E. Prashnani, H. Cai, Y. Mostofi, et P. Sen. PieAPP : Perceptual image-error assessment through pairwise preference. Dans *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–10, Salt Lake City, UT, USA, June 2018.