

Estimation de performances d'un imageur hyperspectral compressif sur système embarqué pour des applications de télédétection

Olivier LIM^{1,2} Stéphane MANCINI¹ Mauro DALLA MURA^{2,3}

¹Univ. Grenoble Alpes, CNRS, Grenoble INP, TIMA, 46 avenue Félix Viallet 38031 Grenoble Cedex, France

²Univ. Grenoble Alpes, CNRS, Grenoble INP, GIPSA-lab, 11 rue des Mathématiques 38402 Saint Martin d'Hères, France

³Institut Universitaire de France (IUF), 1 rue Descartes 75231 Paris Cedex 05, France

Résumé – Ce travail est une étude prospective sur les performances pour la reconstruction de données issues d'un imageur à base d'acquisition comprimée, le Double-Disperser Coded Aperture Snapshot Spectral Imager (DD CASSI), sur un système embarqué tel qu'une carte graphique (GPU) ou un circuit logique programmable (FPGA) dans le but de réaliser des captures hyperspectrales. Cette étude cible le cadre de la télédétection et les contraintes associées. C'est novateur vis-à-vis de la littérature puisque la plupart des travaux sur l'acquisition comprimée porte sur la qualité de reconstruction et ignore les exigences pour des applications en temps-réel en ce qui concerne les coûts algorithmiques ou la bande passante. De plus, les travaux utilisant des systèmes embarqués sont très rares. Des moyens pour améliorer les performances sont présentés et l'amélioration apportée est évaluée.

Abstract – This work is a prospective study and proposes a method to evaluate the performances we can expect when reconstructing data from a compressive sensing imager, the Double-Disperser Coded Aperture Snapshot Spectral Imager (DD CASSI) [8] in an embedded system, i.e. on either a Graphics Processing Unit (GPU) or an Field-Programmable Gate Arrays (FPGA). This study focuses on a remote sensing case of use and drawbacks associated to it. This is original in the literature since most Compressive sensing (CS) works focus on reconstruction quality and overlook the requirements for real-time, namely computational cost and data bandwidth. Moreover, works that use an embedded system are even more scarce. Ways to improve these requirements are presented and the improvement they brought is evaluated.

1 Introduction

L'acquisition comprimée [5] est une technique de traitement du signal qui permet de capturer les informations d'une scène avec moins d'acquisitions que les techniques d'imagerie classique, voire même en dessous de la quantité requise par le théorème de Nyquist. Grâce à l'acquisition comprimée, il est possible de réaliser des captures hyperspectrales (en « 3D ») à partir d'une seule acquisition d'une matrice de capteurs (en « 2D »), permettant ainsi une réduction de la mémoire nécessaire, de la bande passante et un imageur plus compact. De nombreux dispositifs exploitant cette technique ont été proposés, par exemple le *Coded Aperture Snapshot Spectral Imager* (CASSI) [20], ainsi que de nombreuses variantes, le *Double-Disperser Coded Aperture Snapshot Spectral Imager* (DD CASSI) [8], le *3D-CASSI*[6] ou le *Dual-Camera Compressive Hyperspectral Imager* (DCCHI) [21].

En revanche, une phase de reconstruction de la scène à partir des données brutes acquises par le capteur est nécessaire. Il faut résoudre un problème d'optimisation ℓ_1 inverse mal posé. Il existe de nombreux algorithmes pour cela, voir [7], et également en Deep Learning, voir [22]. Ces méthodes ont généralement un coût algorithmique élevé et imposent donc de longs temps d'exécution ce qui est un facteur limitant pour des applications temps-réel. Il est possible de réduire ces temps de reconstruction en combinant accélérations algorithmique et matérielle, par exemple en implémentant les algorithmes sur des cartes graphiques (*Graphics Processing Units* – GPU) ou des circuits logiques programmables (*Field-Programmable Gate Arrays* – FPGA).

Dans la littérature, il existe très peu de travaux portant sur la reconstruction de données hyperspectrales à base d'acquisition comprimée et sur système embarqué. Parmi ceux-là, nous pouvons citer [13] qui propose une approche basée sur le CASSI et le Deep Learning, en couplant U-net [17] et *Generative Adversarial Network* [23] pour la reconstruction. Alors que [24] utilise le DCCHI et le *Two-Step Iterative Shrinkage/Thresholding* [3] en plus de l'*Alternating Direction Method of Multipliers* [4] pour la reconstruction. Ces deux travaux utilisent des GPU pour la reconstruction et nous ne connaissons pas de travaux avec une implémentation sur FPGA.

Ce travail propose une étude sur la faisabilité et les performances (en résolution et qualité de reconstruction) d'un imageur à acquisition comprimée pour la télédétection et l'imagerie hyperspectrale en considérant une implémentation sur système embarqué, soit GPU ou FPGA. Cette étude repose sur un travail antérieur [12] en prenant en compte des contraintes de la télédétection. C'est-à-dire que la phase de reconstruction est limitée à une certaine durée. Cette durée est définie par le temps d'exposition de la scène survolée par les plateformes d'acquisition (soit satellite ou avion). Ensuite, les simulations portent sur des scènes de télédétection afin de mesurer la qualité obtenue pour ce cadre d'application.

2 Méthode

Description du système Pour le reste de ce travail, nous considérerons un système d'imagerie constitué du DD CASSI et du *Conjugate Gradient for Normal Equation* (CGNE) [9] comme algorithme de reconstruction. Nous avons fait ce choix

car ils sont tous deux bien connus et seront considérés comme représentatifs respectivement des systèmes de type CASSI et des algorithmes d'optimisation convexe [16].

Dans cette optique, nous nous intéressons à une approche de reconstruction proposée par Ardi et al. [1] et basée sur le DD CASSI, CGNE et la régularisation de Tikhonov. En revanche, nous considérons une reconstruction ligne par ligne, comme un imageur *pushbroom*, car plus appropriée à la télé-détection, ensuite car cela réduit la taille des variables et, ainsi, leur empreinte mémoire pour permettre de reconstruire de plus grandes scènes avant que le *memory swapping*¹ ne soit nécessaire.

Modélisation du système Le problème d'optimisation à résoudre est le suivant :

$$\hat{o} = \arg \min_{\mathbf{o}} \{ \|\mathbf{d} - \mathbf{H}\mathbf{o}\|^2 + \Omega(\mathbf{o}) \} \quad (1)$$

avec \mathbf{d} les données acquises par le DD CASSI, \mathbf{H} la matrice représentant la fonction de transfert du DD CASSI, \hat{o} l'estimation de la scène observée \mathbf{o} , $\Omega(\mathbf{o}) = \mu_x \|\mathbf{D}_x \mathbf{o}\|^2 + \mu_\lambda \|\mathbf{D}_\lambda \mathbf{o}\|^2$ la fonction de régularisation où \mathbf{D}_x , et \mathbf{D}_λ sont les matrices de différences finies selon la dimension spatiale x et la dimension spectrale λ , et μ_x , μ_λ les coefficients de régularisation associés. Ardi et al. montrent que cela revient à résoudre un système $\mathbf{A}\mathbf{x} = \mathbf{b}$, où $\mathbf{A} = \mathbf{M}^\top \mathbf{M}$ et $\mathbf{b} = \mathbf{H}^\top \mathbf{d}$ avec $\mathbf{M} = [\mathbf{H}, \sqrt{\mu_x} \mathbf{D}_x, \sqrt{\mu_\lambda} \mathbf{D}_\lambda]$.

Étude de complexité Pour une ligne de taille $C \times B$, soit C colonnes et B bandes spectrales, le CGNE a un coût algorithmique de $2(CB)^2 + i(2(CB)^2 + 5CB)$ opérations *multiply-accumulate* (MAC)², avec i le nombre d'itérations du CGNE. Quant à l'empreinte mémoire, il faut stocker $(CB)^2 + 4CB + 3$ valeurs. En considérant $i = \frac{1}{2}CB$, une ligne du jeu de données du satellite Landsat est de taille $C \times B = 6000 \times 10$ et une ligne d'Aviris [14] est de taille 677×224 . La reconstruction d'une de ces lignes nécessite respectivement environ $2,2 \times 10^{14}$ et $3,5 \times 10^{15}$ MAC pour Landsat et Aviris. Or, les GPU et FPGA modernes, tels que la NVIDIA RTX 3090 Ti ou le Xilinx Ultrascale+ VU13P, ont une puissance de calcul de l'ordre de 10^{12} MAC par seconde. Enfin, avec un temps d'exposition par ligne d'environ 4,4 ms pour Landsat et 100 ms pour Aviris, on voit qu'il n'est pas possible de réaliser les reconstructions en temps-réel.

Proposition Pour améliorer les performances, nous avons étudié deux améliorations. En premier lieu, l'utilisation de formats de matrice creuse. Lorsqu'elles sont appliquées à des matrices creuses, elles permettent de réduire à la fois l'empreinte mémoire et aussi le nombre d'opérations pour les produits matrice-vecteurs. En observant de plus près la matrice \mathbf{A} , on remarque qu'elle est composée de diagonales et on peut déterminer le nombre maximal d'éléments $E = (B + 2)CB - 2$. Dans le cas de Landsat, \mathbf{A} présente une densité de 0.02%. Il existe de nombreux formats de matrice creuse [18]. Parmi

¹Le *memory swapping* intervient quand la mémoire de « calcul » est pleine, il est nécessaire de recourir à une mémoire « externe ». Cela impose des transferts de données supplémentaires.

²Une opération MAC est le produit de deux nombres agrégé dans un accumulateur : $a \leftarrow a + (b \times c)$. Les GPU et FPGA sont conçus pour effectuer efficacement ces opérations.

ceux-là, du fait du modèle optique qui impose le schéma selon lequel les éléments de \mathbf{A} sont disposés, les formats Compressed Sparse Row (CSR) et Diagonal (DIA) seraient les plus pertinents. Par ailleurs, selon les expériences réalisées dans la Section 3, le format CSR présente de meilleurs gains pour le temps de reconstruction. Avec ce format, \mathbf{A} a une empreinte mémoire de $E \times (N_b + \lceil \log_2(CB) \rceil) + (CB + 1) \times \lceil \log_2(E) \rceil$ bits au lieu de $N_b \times (CB)^2$ bits, avec N_b le nombre de bits pour représenter un élément de \mathbf{A} et le coût algorithmique du CGNE est de $2(CB^2 + 2CB - 2) + i \times [2CB^2 + 9CB - 4]$ par ligne.

La seconde amélioration porte sur le format de représentation des données. En l'adaptant, on peut réduire l'empreinte mémoire des valeurs stockées et, ainsi, réduire la bande passante utilisée tout en ajustant le compromis avec la qualité de reconstruction. Par ailleurs, au niveau matériel, les FPGA peuvent tirer avantage de la représentation en virgule fixe. Dans cette représentation, une valeur dispose d'un nombre fixe de bits pour représenter la partie entière et la partie décimale. Contrairement à la représentation en virgule flottante, plus courante, celle à virgule fixe n'a pas de d'exposant et donc l'étendue et la précision des valeurs représentables sont bien plus restreintes. Malgré tout, en plus de proposer plus de liberté dans le choix du nombre de bits, elle permet des gains en consommation énergétique et en nombre de portes logiques, voir [2]. Toutefois, lorsqu'on modifie la représentation des données, il faut prendre en compte le bruit de calcul, c'est-à-dire l'accumulation d'imprécision qui peut mener à des résultats faussés ou dégradés. Les résultats expérimentaux montrent que les représentations en virgule flottante sur 64 bits (FP64), sur 32 bits (FP32) et en virgule fixe avec $v = \{32, 24, 16\}$ bits de précision pour la partie décimale ont une qualité identique. La qualité se dégrade significativement pour $v = \{12, 8\}$.

3 Simulations

Des simulations sont réalisées sur des scènes obtenues lors des missions d'observation Landsat [15], AVIRIS [14] ou encore avec l'imageur ROSIS [10]. Ces scènes sont présentées en Figure 1. Les simulations consistent à simuler une acquisition de DD CASSI pour les différentes scènes et ensuite à appliquer le CGNE sur ces données afin de reconstruire la scène. Nous avons notamment pris en compte la vitesse de déplacement des plateformes pour définir le temps dont nous disposons pour la reconstruction et, donc, le nombre d'itérations pour le CGNE. Les micro-miroirs du DMD en transmission sont choisis de manière aléatoire selon une loi de Bernoulli de paramètre p . Pour chaque scène, des simulations préliminaires sont effectuées sur une portion de la scène en utilisant des valeurs aléatoires pour les paramètres p , μ_x et μ_λ . Ensuite, des simulations sont réalisées sur l'ensemble de la scène en utilisant les meilleures valeurs de paramètres, c'est-à-dire celles offrant la meilleure qualité et un faible nombre d'itérations. Les résultats des scènes de Landsat, sont présentés conjointement en faisant la moyenne de leurs valeurs, car leurs dimensions sont identiques et leurs résultats sont proches.

Temps de reconstruction L'amélioration apportée par les formats de matrice creuse est évaluée en comparant leur temps de reconstruction avec celui obtenu en utilisant le format par défaut. Ces temps de reconstruction sont mesurés avec des

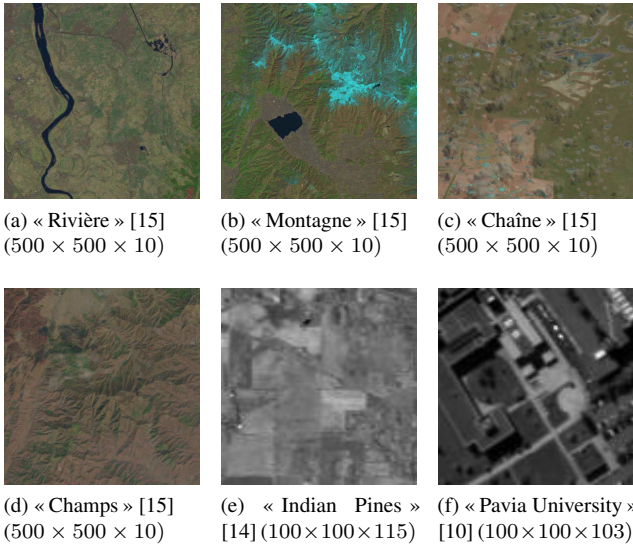


FIGURE 1 : Scènes utilisées lors des simulations avec leurs dimensions et l’imageur utilisé

simulations sur processeur (AMD Ryzen 5800X). En plus des formats CSR et DIA, cette comparaison est également faite avec les formats Block Sparse Row (BSR) et COOrdinate (COO), décrits dans [18]. La comparaison des temps de reconstruction est rapportée dans la Table 1. Le format CSR offre le meilleur gain avec un facteur d’amélioration de 2 152,5, 34,9 et 34,3 pour Landsat, Indian Pines et Pavia University, respectivement. Cette différence s’explique par des dimensions et, notamment, des densités de A inégales, de l’ordre de 0,1% pour Landsat et 1% pour les autres scènes.

	Landsat	Indian Pines	Pavia University
Défaut	111,8	1 220,5	635,1
BSR	1,19 (93, 6)	210,5 (5, 8)	99,0 (6, 4)
COO	0,09 (1 230, 8)	103,0 (11, 8)	54,1 (11, 8)
CSR	0,05 (2 152,5)	35,0 (34,9)	18,5 (34,3)
DIA	0,19 (592, 6)	263,1 (4, 6)	132,9 (4, 8)

TABLE 1 : Temps de reconstruction moyen sur processeur, AMD Ryzen 5800X, par ligne (en seconde) avec le facteur d’amélioration entre parenthèses.

Bruit de calcul Pour l’évaluation du bruit de calcul, les reconstructions sont effectuées avec différents formats de représentation des valeurs. En virgule flottante, on utilise FP64, qui sert également de référence puisqu’elle est la plus précise, et FP32, puisque ce format est celui utilisé par la majorité des cœurs de calcul sur GPU. Pour la représentation en virgule fixe, les reconstructions sont faites avec différentes valeurs de nombre v de bits pour la partie décimale.

Afin de distinguer le bruit de calcul du bruit de reconstruction induit par le CGNE, les simulation de reconstruction sont faites avec un nombre N croissant d’acquisitions jusqu’à atteindre B , le nombre de bande spectrales de la scène. Et, bien que le DD CASSI soit un imageur à capture instantanée (dit *snapshot*), augmenter N permet de savoir s’il est nécessaire d’effectuer plusieurs acquisitions pour obtenir une qualité satisfaisante. La qualité de reconstruction est mesurée par le *Peak Signal-to-Noise Ratio* (PSNR) et *Structural Similarity*

Index (SSIM), tous deux décrits dans [19], et est présentée dans la Figure 2.

FP64, FP32 et la représentation en virgule fixe avec $v = \{32, 24, 16\}$ offrent la même qualité de reconstruction, puisque leurs courbes de PSNR et SSIM respectives se superposent sur la Figure 2, quelque soit la scène étudiée. Ensuite, les cas $v = \{12, 8\}$ peuvent être abandonnés car la qualité est fortement dégradée. Les scènes Landsat ont un PSNR situé entre 23 et

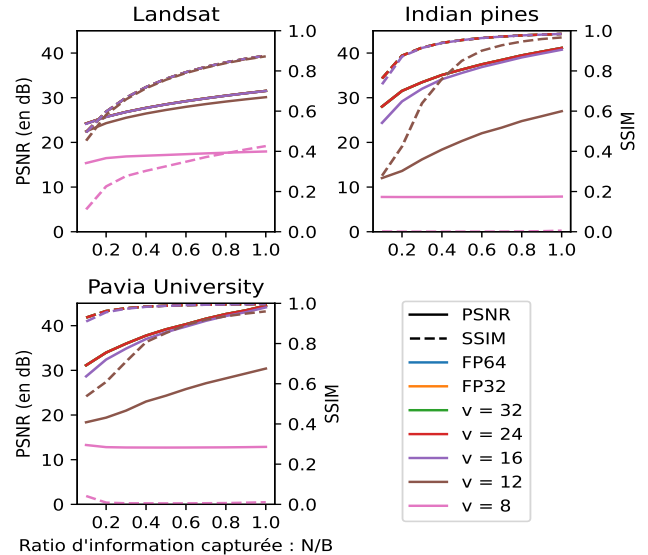


FIGURE 2 : Qualité de reconstruction selon la représentation des valeurs. v désigne le nombre de bits pour la partie décimale de la représentation à virgule fixe.

34 dB, entre 28 et 41 dB pour Indian Pines et entre 31 et 44 dB pour Pavia University. Pour atteindre un seuil de 30 dB, il faut 8 acquisitions en moyenne pour Landsat, aux alentours de 20 pour Indian Pines et environ 10 pour Pavia University. Par ailleurs, le PSNR inférieur pour Landsat se justifie par la non continuité des bandes spectrales du jeu de données, alors que notre système favorise la continuité des bandes dans la reconstruction, mais également avec la nature dispersive du DD CASSI. Néanmoins, pour les scènes de Landsat, en faisant une étude visuelle, 6 acquisitions suffisent pour avoir une image très proche de la scène d’origine.

4 Conclusion

L’utilisation d’un imageur à acquisition comprimée, le DD CASSI, sur système embarqué pour la télédétection est possible. Toutefois, bien que le DD CASSI soit un imageur à capture instantanée, il est nécessaire de réaliser plusieurs acquisitions de la scène afin d’obtenir une qualité satisfaisante. Cela est envisageable en utilisant, par exemple, des photodétecteurs à report et intégration [11]. En estimant les transferts de données du CGNE, la bande passante limite à 37 bandes spectrales une implémentation de notre système sur Landsat, à 115 bandes spectrales pour AVIRIS et 170 pour ROSIS, c’est-à-dire en considérant la résolution spatiale de ces imageurs et l’empreinte mémoire déterminé dans ce travail. Cela permettrait d’augmenter la résolution spectrale pour Landsat et ROSIS, mais la réduirait pour AVIRIS. Néanmoins, quelque soit l’imageur, grâce au multiplexage des acquisitions, il est possible de réduire la taille des capteurs. Puisqu’au lieu de

répéter des lignes de capteurs selon le nombre de bandes spectrales, il suffit de les répéter selon le nombre d'acquisitions nécessaires. On diviserait donc par environ 5 ou 10 la taille des capteurs.

Quant à la comparaison GPU et FPGA, les reconstructions étant limitées par la bande passante, au premier abord on pourrait dire que les FPGA ont l'avantage avec une mémoire de calcul plus élevée réduisant ainsi les problèmes de *memory swapping*. Néanmoins, pour en être certain, il faudrait étudier les modalités d'accès à la mémoire. D'autant plus que sur FPGA, il est possible d'adapter les formats de matrice pour exploiter au mieux l'architecture matérielle et les accès mémoire.

Parmi les perspectives de ce travail, des simulations matérielles de FPGA seraient intéressantes notamment pour mesurer les améliorations matérielles apportées par la virgule fixe et également avoir une idée plus précise sur les temps de reconstruction. On peut également se pencher sur l'utilisation d'arithmétique à précision variable puisqu'on effectue une descente de gradient.

Références

- [1] I. ARDI, H. CARFANTAN, S. LACROIX et A. MONMAYRANT : Fast hyperspectral cube reconstruction for a double disperser imager. *In 26th European Signal Processing Conference (EUSIPCO 2018)*, Roma, Italy, Sep 2018.
- [2] S. AYAN : Introduction to fixed point signal processing. <https://shafq.at/introduction-to-fixed-point-signal-processing.html>, 2020.
- [3] J. M. BIOUSCAS-DIAS et M. A. T. FIGUEIREDO : A new twist : Two-step iterative shrinkage/thresholding algorithms for image restoration. *IEEE Transactions on Image Processing*, 16(12):2992–3004, 2007.
- [4] S. BOYD, N. PARIKH, E. CHU, B. PELEATO et J. ECKSTEIN : *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*. Now Foundations and Trends, 2011.
- [5] E. CANDÈS et J. ROMBERG : Sparsity and incoherence in compressive sampling. *Inverse Problems*, 23(3):969–985, Apr 2007.
- [6] C. V. CORREA, H. ARGUELLO et G. R. ARCE : Snapshot colored compressive spectral imager. *J. Opt. Soc. Am. A*, 32(10):1754–1763, Oct 2015.
- [7] I. GARCÍA-SÁNCHEZ, O. FRESNEDO, J. P. GONZÁLEZ-COMA et L. CASTEDO : Coded aperture hyperspectral image reconstruction. *Sensors*, 21(19), 2021.
- [8] M. E. GEHM, R. JOHN, D. J. BRADY, R. M. WILLETT et T. J. SCHULZ : Single-shot compressive spectral imaging with a dual-disperser architecture. *Opt. Express*, 15(21):14013–14027, Oct 2007.
- [9] M. HANKE : *Conjugate Gradient Type Methods for Ill-Posed Problems*. CRC Press, 2017.
- [10] B. KUNKEL, F. BLECHINGER, R. LUTZ, R. DOERFFER, H. van der PIEPEN et M. SCHRODER : ROSIS (Reflective Optics System Imaging Spectrometer) - A Candidate Instrument For Polar Platform Missions. *In C. Stuart BOWYER et John S. SEELEY, éditeurs : Optoelectronic Technologies for Remote Sensing from Space*, volume 0868, pages 134 – 141. International Society for Optics and Photonics, SPIE, 1988.
- [11] G. LEPAGE, J. BOGAERTS et G. MEYNANTS : Time-delay-integration architectures in cmos image sensors. *IEEE Transactions on Electron Devices*, 56(11):2524–2533, 2009.
- [12] O. LIM, S. MANCINI et M. Dalla MURA : Feasibility of a real-time embedded hyperspectral compressive sensing imaging system. *Sensors*, 22(24), 2022.
- [13] X. MIAO, X. YUAN, Y. PU et V. ATHITSOS : lambda-net : Reconstruct hyperspectral images from a snapshot measurement. *In 2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4058–4068, 2019.
- [14] NASA : Aviris - airborne visible / infrared imaging spectrometer. <https://aviris.jpl.nasa.gov/index.html>.
- [15] NASA : Landsat science. <https://landsat.gsfc.nasa.gov/>.
- [16] B. POLYAK : *Introduction to Optimization*. 07 2020.
- [17] O. RONNEBERGER, P. FISCHER et T. BROX : U-net : Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015.
- [18] Y. SAAD : *Sparskit : a basic tool kit for sparse matrix computations*, 1994.
- [19] K.-H. THUNG et P. RAVEENDRAN : A survey of image quality measures. *In 2009 International Conference for Technical Postgraduates (TECHPOS)*, pages 1–4, 2009.
- [20] A. WAGADARIKAR, R. JOHN, R. WILLETT et D. BRADY : Single disperser design for coded aperture snapshot spectral imaging. *Appl. Opt.*, 47(10):B44–B51, Apr 2008.
- [21] L. WANG, Z. XIONG, D. GAO, G. SHI et F. WU : Dual-camera design for coded aperture snapshot spectral imaging. *Appl. Opt.*, 54(4):848–858, Feb 2015.
- [22] X. YUAN, D. J. BRADY et A. K. KATSAGGELOS : Snapshot compressive imaging : Theory, algorithms, and applications. *IEEE Signal Processing Magazine*, 38(2):65–88, 2021.
- [23] H. ZHANG, I. GOODFELLOW, D. METAXAS et A. ODENA : Self-attention generative adversarial networks, 2019.
- [24] S. ZHANG, H. HUANG et Y. FU : Fast parallel implementation of dual-camera compressive hyperspectral imaging system. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(11):3404–3414, 2019.