



Frugalité en télédétection : évaluer la performance et la consommation énergétique en détection de changements d'images PolSAR

Matthieu VERLYNDE¹ Ammar MIAN¹ Yajing YAN¹

¹LISTIC, Université Savoie Mont Blanc, 5 chemin de Bellevue, 74940 Annecy-le-Vieux, France

Résumé – Dans les applications de télédétection, la demande d’algorithmes de traitements de données complexes augmente exponentiellement. En parallèle, évaluer la frugalité de telles méthodes est une priorité dans la communauté de l’apprentissage machine et de l’intelligence artificielle. Dans ce cadre, créer une mesure combinant performance et consommation énergétique est une tâche complexe. Cet article présente une méthodologie de collecte de données de consommation énergétique et explore trois méthodes de création de score de frugalité. À travers une étude de cas sur une tâche classique en télédétection, la détection de changements, nous montrons que ces trois scores capturent trois différents aspects de la frugalité. Nous recommandons alors une approche combinatoire de ces scores afin de réaliser une évaluation plus complète.

Abstract – In remote sensing applications, the demand for extensive data processing algorithms continues to grow exponentially. At the same time, assessing the frugality of data processing algorithms has become a priority in the machine learning and artificial intelligence community. However, defining a quantifiable measure that combines performance and energy consumption remains a complex challenge. This paper introduces a methodology for collecting energy consumption data and presents three different frugality scoring methods. Through a case study of a classical data processing task in remote sensing, change detection, we demonstrate that these three scores capture different aspects of frugality. We further suggest a combined approach for users to achieve a more comprehensive assessment.

1 Introduction

Alors que les outils d’apprentissage machine et d’intelligence artificielle se sont largement développés au sein des méthodes de traitement de données en télédétection [9], l’impact écologique de ces outils croît exponentiellement [5, 16]. C’est pourquoi il est essentiel d’approfondir la recherche sur l’évaluation et l’amélioration de leur frugalité [5] pour les acteurs de la télédétection. Dans cet article, nous qualifions de frugalité la recherche d’une faible consommation énergétique tout en permettant une performance satisfaisante de la méthode d’intérêt.

Mesurer la frugalité est une tâche complexe, et c’est pourquoi le coût écologique ou énergétique d’une méthode est souvent estimé à travers sa durée d’exécution [19, 8]. Cependant, la durée d’exécution ne couvre pas l’ensemble des informations de la consommation énergétique [2]. D’autres métriques, telles que la complexité algorithmique ou le nombre de lignes de code, peuvent également être utilisées [10, 14]. Néanmoins, elles n’apportent qu’une estimation de la durée d’exécution théorique, ce qui est difficilement convertible en empreinte carbone. Différents outils de mesure de consommation énergétique ont alors été publiés, tels que CarbonTracker [4] ou Code-Carbon¹. Enfin, une autre approche est la mesure empirique de la consommation énergétique du système via l’utilisation d’un capteur externe. Sous réserve que la fréquence d’acquisition des mesures soit suffisamment élevée, cette stratégie décrit plus efficacement la consommation énergétique des méthodes d’intérêt. Cette approche prend alors en compte l’ensemble des processus invisibles au sein du système dans la mesure de consommation énergétique.

Évaluer la frugalité implique la combinaison de la

consommation énergétique estimée de la méthode avec sa performance. Plusieurs études se concentrent sur la performance et la durée d’exécution [1, 8], mais la littérature fournit peu d’études proposant un score basé sur la consommation énergétique empirique. Pour cela, de nombreuses approches de création d’indice multi-critère sont possibles. Si l’utilisation d’une somme pondérée pour l’agrégation de métriques est critiquée pour son caractère linéaire et non généralisable, ainsi que pour la subjectivité possible des choix de ses poids [12], elle est une méthode classique d’établissement de score multi-critère [3, 17]. Des approches par logique floue sont également classiques pour des outils d’aide à la décision [7, 18]. Elles nécessitent, cependant, un savoir expert afin de définir les frontières de décision et sont peu adaptables à différentes méthodes de traitement de données.

Cet article explore, pour la première fois, la question de la frugalité dans le domaine spécifique de la télédétection. Nous proposons :

- une chaîne de collecte de mesures de consommation énergétique et de performance²,
- une stratégie de combinaison de métrique pour évaluer la frugalité à travers la création de trois scores,
- une étude de cas sur une tâche classique en télédétection : la détection de changements.

²<https://github.com/MattVerlynde/frugal-gretsi-2025.git>

¹<https://codecarbon.io/>

2 Consommation énergétique

Évaluer la frugalité nécessite l'accès à des mesures du hardware interne et externe et des software du système utilisé. Dans cet article, nous nous focalisons sur une mesure globale de la consommation énergétique d'une tâche réalisée sur une unique machine locale grâce à l'utilisation d'une prise intelligente connectée à une base de données InfluxDB sous un protocole Z-Wave. Les mesures sont réalisées selon trois périodes d'acquisition : une période de standardisation sans exécution de l'algorithme étudié, une période latente, puis une période d'exécution de l'algorithme. Avec $P(t)$ la puissance mesurée à un temps t , t_0 le début de la période de standardisation, t_1 le début de la période de latence et t_2 le début de la période d'exécution jusqu'à t_3 , la consommation énergétique mesurée E , utilisée pour évaluer la frugalité de la méthode, est calculée suivant eq. (1) :

$$E = \int_{t_2}^{t_3} P(t) dt - \frac{t_3 - t_2}{t_1 - t_0} \int_{t_0}^{t_1} P(t) dt. \quad (1)$$

La période de standardisation permet d'exclure les processus de fond du système de la mesure. La période de latence est une période avec une unique exécution de l'algorithme d'intérêt sans mesure, permettant au système de monter en température. Cette chaîne de traitement permet de collecter uniquement la mesure de la consommation énergétique de la méthode étudiée.

Une fois ces données collectées, la performance de la méthode est mesurée suivant la tâche effectuée. L'évaluation de la frugalité requiert alors l'agrégation des deux mesures, pour laquelle une stratégie est présentée à la section suivante.

3 Scores de frugalité

Créer un score de frugalité permet à l'utilisateur une lecture simple de cette dimension, et une comparaison directe entre différentes méthodes. Cependant, celle-ci nécessite la combinaison de la performance de la méthode α sur la tâche d'intérêt et de la consommation énergétique β durant la période d'exécution. Une combinaison classique de telles métriques est l'utilisation d'une somme pondérée. Cependant, celle-ci nécessite la normalisation des deux métriques d'intérêt, introduisant alors un biais au sein de la donnée. Notons $\epsilon \in [0, 1]$ le poids associé à la performance de la méthode comparée à la consommation énergétique, et $\alpha_n \in [0, 1]$ et $\beta_n \in [0, 1]$ les valeurs normalisées de α et β respectivement. Le score formé s_{WS} est alors présenté en eq. (2) :

$$s_{WS} = \epsilon \times \alpha_n + (1 - \epsilon) \times (1 - \beta_n), \quad (2)$$

$$\text{où } \alpha_n = \frac{\alpha - \min(\alpha)}{\max(\alpha) - \min(\alpha)}, \quad \beta_n = \frac{\beta - \min(\beta)}{\max(\beta) - \min(\beta)}.$$

Une autre méthode de combinaison de ces métriques est la moyenne harmonique, de manière similaire au calcul du F-score à partir des mesures de précision et de rappel pour les tâches de classification. Nous proposons cette méthode car celle-ci est particulièrement adaptée pour la combinaison de ratios [15] et est donc pertinente pour les métriques normalisées α_n et β_n . Notons $\kappa \in \mathbb{R}^+$ le poids associé à la consommation énergétique, et le score s_{HM} est alors calculé selon eq. (3) :

$$s_{HM} = (1 + \kappa^2) \frac{\alpha_n(1 - \beta_n)}{\kappa^2 \alpha_n + (1 - \beta_n)}. \quad (3)$$

Ces deux scores nécessitant la normalisation des mesures de performance et de consommation énergétique de la méthode d'intérêt, ils sont non généralisables à l'ajout de nouvelles mesures. Pour pallier cet aspect, nous proposons un score de frugalité s_F inspiré de [8] présenté en eq. (4) :

$$s_F = \alpha - \frac{w}{1 + \frac{1}{\beta}}. \quad (4)$$

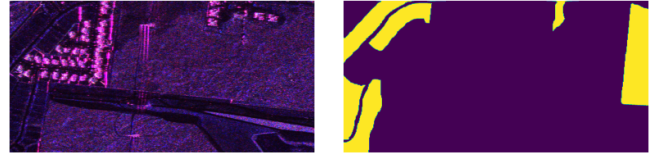
Ce dernier est formé à partir des deux métriques α et β , avec un poids $w \in [0, 1]$ associé à la ressource disponible, ici la consommation énergétique, en contraste avec la performance de la méthode. Il permet donc l'utilisation des valeurs brutes de ces mesures, et donc la généralisation des résultats. Des résultats de ce type, étudiés sur ces trois scores de frugalités, sont présentés dans la section suivante.

4 Résultats

Dans cette section, nous présenterons une étude de cas pour laquelle ces scores de frugalité seront utilisés, ainsi que les résultats obtenus.

4.1 Étude de cas

Pour évaluer l'efficacité de nos scores de frugalité pour des applications en télédétection, nous étudions une tâche de détection de changements, une tâche classique en traitement de données de télédétection.



(a) Composite RGB de l'image SAR à $t = 0$. (b) Vérité terrain de la détection de changements.

FIGURE 1 : Données pour la détection de changements issues de la base UAVSAR (Courtesy NASA/JPL-Caltech) [11, 13].

Pour évaluer les scores de frugalité sur la tâche de détection de changements, trois algorithmes sont appliqués sur des données de télédétection. Ces données sont formées d'une série temporelle de 4 images PolSAR à trois bandes issues de la base de données UAVSAR (Courtesy NASA/JPL-Caltech) [11] de taille 1000×500 pixels. Celles-ci comportent également une vérité terrain, présentée en figure 1, issue de [13]. Afin de prendre en compte l'aspect multivarié des données (les trois bandes de polarisation de chaque image) et le bruit de chatoiement (speckle noise), nous utilisons des méthodes basées sur les matrices de covariance entre chaque bande pour chaque pixel. Ainsi, avec p bandes par image, $T + 1$ images multi-bande, N pixels par bande, nous considérons $x_k^{(t)}$ avec $t \in \llbracket 0, T \rrbracket$ et $k \in \llbracket 1, N \rrbracket$ un échantillon de pixels sur l'ensemble des bandes au sein d'une fenêtre glissante sur l'image. Pour cette tâche, nous considérons alors $x_k^{(t)}$ comme la réalisation d'une variable aléatoire suivant un modèle probabiliste

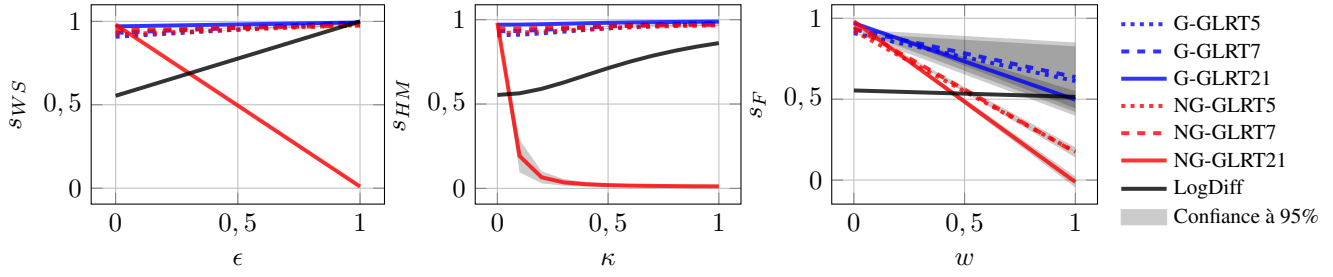


FIGURE 2 : Scores de frugalité moyens sur la tâche de détection de changements en fonction de l'importance donnée à l'économie d'énergie (gauche : faible importance, droite : forte importance).

$p_x(x_k, \theta_t)$ où θ_t est l'ensemble des paramètres pour l'image t . La détection de changements est donc définie comme celle de ces paramètres, donc la comparaison des deux hypothèses H_0 et H_1 suivant eq. (5) :

$$\begin{cases} H_0 : \theta_0 = \theta_1, \\ H_1 : \theta_0 \neq \theta_1. \end{cases} \quad (5)$$

La statistique de test utilisée est alors basée sur la matrice de covariance des bandes de polarisation sur un échantillon local, donc au sein de la fenêtre glissante. La première statistique de test est celle du test du Rapport de Vraisemblance Généralisé (GLRT). Nous considérons alors ce test sous l'hypothèse de distribution gaussienne des valeurs des pixels, donc avec $\mathbf{x}_k^{(t)}$ étant la réalisation d'une variable aléatoire $\mathbf{x}_k^{(t)} \sim \mathcal{CN}(0, \Sigma_t)$ avec $\Sigma_t = \theta_t$ une matrice de covariance. La méthode associée à cette statistique sera alors appelée **G-GLRT** dans cet article. L'implémentation choisie pour le calcul de ce test statistique est proposée par [6] avec une approche par paires des images successives au sein d'une série temporelle pour la détection de changements.

La deuxième méthode de détection de changements utilisée est basée sur la statistique de test GLRT étendue aux distributions non-gaussiennes sur des images PolSAR, proposée par [11], avec $\mathbf{x}_k^{(t)} \sim \mathbf{z}_k^{(t)} \sqrt{\tau_k^{(t)}}$ et $\mathbf{z}_k^{(t)} \sim \mathcal{CN}(0, \Sigma_t)$, utilisant une information de texture $\tau_k^{(t)} \in \mathbb{R}^+$ au sein de la donnée. La méthode associée à ce test est notée **NG-GLRT** dans cet article. Comparée à la méthode précédente, la statistique de test de celle-ci introduit la texture à travers un estimateur de point fixe (l'estimateur de Tyler), nécessitant un grand nombre de calculs. Les deux méthodes G-GLRT et NG-GLRT ont été appliquées à trois tailles de fenêtres glissantes (5, 7 et 21 pixels) pour calculer les matrices de covariance empiriques au sein des images. L'utilisation de ces valeurs permet alors d'identifier l'effet de ce paramètre clé sur la consommation énergétique de ces méthodes.

Enfin, une approche directe et extrêmement rapide d'estimation de ces changements est la somme des différences du logarithme des valeurs des moyennes inter-bande pour chaque pixel entre deux images successives de la série temporelle. La statistique associée $\hat{\Lambda}_{LD}$ est alors présentée en eq. (6). La méthode basée sur cette statistique est nommée **LogDiff** dans cet article.

$$\hat{\Lambda}_{LD} = \sum_{t=1}^T \frac{1}{p} \left(\sum_{\text{bandes}} \ln x^{(t)} - \sum_{\text{bandes}} \ln x^{(t-1)} \right). \quad (6)$$

La performance de ces trois méthodes de détection de changements est évaluée selon l'aire sous la courbe sensibilité/spécificité (AUC) basée sur les statistiques de test présentées précédemment. L'application de ces méthodes de détection de changements produit un ensemble de données de performances et de consommation énergétique, collectées suivant un plan d'expériences présenté dans la prochaine sous-section.

4.2 Plan d'expérience

Les expériences sont conduites suivant une implémentation basée uniquement sur CPU. Ainsi, celles-ci sont conduites sur un CPU Intel i5-12600 3.30GHz, une RAM de 2×32 Go, et une Smart Switch 7 Aeotec® avec un contrôleur USB Z-Stick 7 Aeotec®.

Afin de garantir une significativité statistique de la mesure de consommation énergétique, chaque exécution est répétée 30 fois. Pour chaque exécution, cette consommation énergétique ainsi que la performance de la méthode et ses paramètres sont collectés³. Les résultats obtenus sont alors analysés dans la section suivante.

4.3 Résultats obtenus

Pour chaque méthode, sa performance et sa consommation énergétique sont mesurées et intégrées dans le calcul de chaque score de frugalité associé.

TABLE 1 : Consommation énergétique des méthodes de détection de changements (moyenne et intervalle de confiance à 95%). Pour G-GLRT x et NG-GLRT x , x correspond à une taille de fenêtre de x pixels.

Méthode	Consommation (Wh)	AUC
G-GLRT5	0,892 ± 0,236	0,908
G-GLRT7	0,902 ± 0,273	0,932
G-GLRT21	0,924 ± 0,507	0,970
NG-GLRT5	3,02 ± 1,16	0,922
NG-GLRT7	3,55 ± 1,26	0,947
NG-GLRT21	137 ± 1,54	0,981
LogDiff	0,0000411 ± 0,00000509	0,554

Les trois scores de frugalité s'accordent sur le classement des méthodes appliquées, identifiant la méthode G-GLRT comme la plus frugale, tel que présenté en figure 2. La méthode NG-GLRT montre des performances proches de celles

³<https://github.com/MattVerlynde/frugal-gretsi-2025.git>

de G-GLRT et, en moyenne, la méthode LogDiff produit de moins bons résultats sur ces données, avec une AUC d'environ 0,6. Aussi, la taille de la fenêtre glissante utilisée pour G-GLRT et NG-GLRT affecte significativement la consommation énergétique. Une plus petite fenêtre induit, de plus, une meilleure performance sur la tâche pour G-GLRT, alors que l'effet inverse apparaît pour NG-GLRT. Plus l'énergie est prise en compte dans le calcul de s_{WS} et s_{HM} , plus la valeur de ces scores décroît uniquement pour NG-GLRT21. Cet effet est dû à la normalisation de α_n et β_n dans le calcul des deux scores de frugalité. Cependant, nous observons une différence dans leur évolution selon ϵ et κ . s_{WS} montre une évolution linéaire selon ϵ , alors que s_{HM} est une fonction rationnelle de κ . Ces variations montrent un impact plus complexe de la consommation énergétique sur le score. Par exemple, l'évolution de s_{WS} pour NG-GLRT21 montre que sa consommation énergétique est significativement supérieure à celle des autres méthodes. L'évolution de s_{HM} présente une forte pente sur $\kappa \in [0, 0.5]$ montrant une séparation plus visible entre les méthodes. Ainsi, pour une valeur de κ donnée, s_{HM} tend à discriminer les méthodes selon leur frugalité plus distinctement que s_{WS} . Cependant, ces deux scores nécessitent une normalisation des mesures, ce qui est dépendant des expériences menées. s_F permet une comparaison directe des méthodes, dépendant de la mesure de consommation énergétique brute. Comme présenté en figure 2, ce score sépare les méthodes G-GLRT, NG-GLRT et LogDiff plus distinctement que s_{WS} et s_{HM} , mais identifie moins distinctement l'influence des paramètres de chaque méthode.

5 Conclusions

Cette étude a montré la difficulté d'identifier l'aspect frugal d'une méthode de traitement de données en télédétection. Une méthodologie a été présentée pour collecter les données de consommation énergétique et les combiner avec les performances de méthodes de traitement de données via l'utilisation de trois scores de frugalité. Ces trois représentations ont alors fourni différentes informations dans l'étude de frugalité de méthodes dans le cadre d'une étude de cas en télédétection. s_{WS} est une approche simple et directe, mais s_{HM} discrimine plus efficacement les méthodes selon leur consommation énergétique, tout en garantissant qu'une méthode frugale ne présente pas la pire performance. s_F identifie les méthodes les plus frugales plus distinctement que les autres scores, même s'il est moins sensible à l'influence des paramètres de chaque méthode. Il peut aussi être étendu à de futures expériences sans prétraitement des mesures de consommation énergétique. Utiliser alors de multiples scores pour évaluer la frugalité apparaît donc pertinent afin de capturer ses différents aspects.

Références

- [1] Salisu ABDULRAHMAN et Pavel BRAZDIL : Measures for combining accuracy and time for meta-learning. *CEUR Workshop Proceedings*, 1201:49–50, 01 2014.
- [2] Sarah ABDULSALAM, Ziliang ZONG, Qijun GU et Meikang QIU : Using the greenup, powerup, and speedup metrics to evaluate software energy efficiency. In *2015 Sixth International Green and Sustainable Computing Conference (IGSC)*, pages 1–8. IEEE, 2015.
- [3] Naim H. AFGAN, Petros A. PILAVACHI et Maria G. CARVALHO : Multi-criteria evaluation of natural gas resources. *Energy Policy*, 35(1):704–713, 2007.
- [4] Lasse F. Wolff ANTHONY, Benjamin KANDING et Raghavendra SELVAN : Carbontracker : Tracking and predicting the carbon footprint of training deep learning models, 2020.
- [5] Verónica BOLÓN-CANEDO, Laura MORÁN-FERNÁNDEZ, Brais CANCELA et Amparo ALONSO-BETANZOS : A review of green artificial intelligence : Towards a more sustainable future. *Neurocomputing*, 599:128096, 2024.
- [6] Knut CONRADSEN, Allan Aasbjerg NIELSEN et Henning SKRIVER : Determining the points of change in time series of polarimetric sar data. *IEEE Transactions on Geoscience and Remote Sensing*, 54(5):3007–3024, 2016.
- [7] J Ronald EASTMAN : Multi-criteria evaluation and gis. *Geographical information systems*, 1(1):493–502, 1999.
- [8] Mikhail EVCHENKO, Joaquin VANSCHOREN, Holger H. HOOS, Marc SCHOENAUER et Michèle SEBAG : Frugal machine learning, 2021.
- [9] David J. LARY, Amir H. ALAVI, Amir H. GANDOMI et Annette L. WALKER : Machine learning in geosciences and remote sensing. *Geoscience Frontiers*, 7(1):3–10, 2016. Special Issue : Progress of Machine Learning in Geosciences.
- [10] Firdous Ahmad MALA et Rouf ALI : The big-o of mathematics and computer science. *Appl. Math. Comput.*, 6(1):1–3, 2022.
- [11] Ammar MIAN, Guillaume GINOLHAC, Jean-Philippe OVARLEZ et Abdourrahmane Mahamane ATTO : New Robust Statistics for Change Detection in Time Series of Multivariate SAR Images. *IEEE Transactions on Signal Processing*, 67(2):520–534, janvier 2019.
- [12] Sandro MORASCA : On the use of weighted sums in the definition of measures. In *Proceedings of the 2010 ICSE Workshop on Emerging Trends in Software Metrics, WETSoM '10*, pages 8–15, New York, NY, USA, mai 2010. Association for Computing Machinery.
- [13] Abraão D. C. NASCIMENTO, Alejandro C. FRERY et Renato J. CINTRA : Detecting changes in fully polarimetric sar imagery with statistical information theory. *IEEE Transactions on Geoscience and Remote Sensing*, 57(3):1380–1392, 2019.
- [14] Jukka K NURMINEN : Using software complexity measures to analyze algorithms—an experiment with the shortest-paths algorithms. *Computers & Operations Research*, 30(8):1121–1134, 2003.
- [15] Cornelis Joost VAN RIJSBERGEN : Foundation of evaluation. *Journal of documentation*, 30(4):365–373, 1974.
- [16] Wim VANDERBAUWHEDE : Frugal computing – on the need for low-carbon and sustainable computing and the path towards zero-carbon computing, 2023.
- [17] Rui WANG, Zhongbao ZHOU, Hisao ISHIBUCHI, Tianjun LIAO et Tao ZHANG : Localized weighted sum method for many-objective optimization. *IEEE Transactions on Evolutionary Computation*, 22(1):3–18, 2018.
- [18] B. WIRSAM, A. HAHN, E. O. UTHUS et C. LEITZMANN : Fuzzy sets and fuzzy decision making in nutrition. *European Journal of Clinical Nutrition*, 51(5):286–296, mai 1997. Publisher : Nature Publishing Group.
- [19] Tomofumi YUKI et Sanjay RAJOPADHYE : Folklore confirmed : Compiling for speed compiling for energy. In *International Workshop on Languages and Compilers for Parallel Computing*, pages 169–184. Springer, 2013.