

RAPPORT DE STAGE

MÉTHODES DE COMPRESSION DE DONNÉES

RÉALISÉ PAR INA ELENA CAMPAN

*Étudiante en 2ème année de la double licence
Mathématiques-Informatique (Parcours PIMA) de
Sorbonne Université*

ENCADRANT : STÉPHANE LABBÉ

*Professeur des Universités au laboratoire
de mathématiques appliquées Jacques-Louis Lions*



Juillet - Août 2023

Table de matières

1	Motivation	2
2	Introduction aux concepts mathématiques	3
2.1	Généralités mathématiques	3
2.2	Fonctions et la base de Haar	3
2.3	Transformée en ondelettes	5
2.4	Applications	6
3	Algorithmes & implémentation en Python	7
3.1	Organisation des fichiers et structures utilisées	7
3.2	Répertoire GitHub	7
4	Résultats	8
4.1	Tests et observations	8
4.2	Améliorations possibles	9
5	Poursuite des recherches et conclusion	10
5.1	Nouvelle application	10
5.2	Conclusion	12

1 Motivation

Sous la direction de M. Labbé, le stage a porté sur l'étude de diverses méthodes de compression de données. D'un point de vue mathématique, les notions acquises concernent la compréhension des bases de Haar et les transformations en ondelettes. Cette étude a été appliquée à des tests de compression d'image en noir et blanc, qui ont relevé une très bonne efficacité : une compression importante sans perte de détails. Le code a été implémenté en Python, le but étant de réécrire les algorithmes qui se retrouvent dans des bibliothèques prédéfinies du langage.

Pour la deuxième partie du stage, nous avons débuté une application pour l'accélération des calculs en électro-magnétisme, en particulier pour des réseaux d'antennes.

Concernant la poursuite de mes études, ce stage a été un outil pour découvrir le milieu de la recherche et continuer vers un master orienté vers ce cadre. Le Laboratoire Jacques-Louis Lions, affilié à Sorbonne Université, constitue un des plus grands laboratoires de France pour la formation et la recherche en mathématiques appliquées, ce qui m'a donné l'occasion de travailler et de rencontrer des experts dans la recherche scientifique.

Le stage a constitué une première expérience professionnelle qui m'a permis d'appliquer mes connaissances en mathématiques et, en même temps, en informatique, dans un contexte très concret et pratique.

2 Introduction aux concepts mathématiques

2.1 Généralités mathématiques

On appelle **espace vectoriel sur \mathbb{K}** (ou \mathbb{K} -espace vectoriel) un ensemble E muni de deux lois : [1]

1. une loi interne, notée $+$, telle que $(E, +)$ soit un groupe commutatif
2. une loi externe, notée $*$, qui est une application de $K \times E$ dans E vérifiant :
 - $\forall (\alpha, \beta) \in \mathbb{K}^2, \forall x \in E : (\alpha + \beta) * x = \alpha * x + \beta * x$
 - $\forall \alpha \in \mathbb{K}, \forall (x, y) \in E^2 : \alpha * (x + y) = \alpha * x + \alpha * y$
 - $\forall (\alpha, \beta) \in \mathbb{K}^2, \forall x \in E : \alpha * (\beta x) = (\alpha\beta) * x$
 - $\forall x \in E : 1 * x = x$

Un **produit scalaire** est une forme bilinéaire symétrique définie positive sur un espace vectoriel sur les nombres réels.

On introduit aussi la notion de **base hilbertienne** : Soit H un espace préhilbertien et (e_n) une suite d'éléments de H . On dit que (e_n) est une base hilbertienne de H si :

1. la suite est *orthonormale* : pour $\forall n \neq m$ on a $\langle e_n, e_m \rangle = 0$ et $\|e_n\| = 1$
2. la suite est *totale* : l'espace vectoriel engendré par les e_n est dense dans H .

Remarque :

Un intérêt particulier a été porté sur l'espace $L^2([0, 1]) := L^2([0, 1], L(\mathbb{R}), \lambda)$, qui constitue un *espace Euclidien* :

1. pour le produit scalaire $(f, g) \rightarrow \langle f, g \rangle = \int_0^1 fgd\lambda$, avec f, g deux fonction de l'espace $L^2([0, 1])$
2. $L(\mathbb{R}) \rightarrow$ la tribu de Lebesgue
3. $\lambda \rightarrow$ la mesure de Lebesgue.

2.2 Fonctions et la base de Haar

Pour le traitement des images, la transformée de Fourier (notion vue en cours) a ses limitations : [2]

- elle ne permet pas de délimiter les zones riches en informations des zones pauvres ;

- la modification d'un seul coefficient de Fourier va impacter la totalité du signal ou de l'image en cause.

Ces inconvénients entraînent la préférence pour les ondelettes, en particulier, celles de Haar.

Définition :

On définit une *suite de fonctions par niveaux* :

1. Posons les intervalles $I_{n,k} = [\frac{k}{2^n}, \frac{k+1}{2^n}[$, avec $n \in \mathbb{N}$ et $k \in \{0, 1, \dots, 2^n - 1\}$.
2. Soit $\varphi_{n,k} : I_{n,k} \rightarrow \{0, 2^{n/2}\}$; $\varphi_{n,k} = 2^{n/2} \chi_{I_{n,k}}$
3. **Remarque** : χ est la fonction indicatrice
4. Soit $V_n := \{\varphi_{n,k} : k = 0, 1, \dots, 2^n - 1\}$ (base orthonormée)
5. Ainsi, nous construisons de manière récursive les fonctions :

$$\varphi_{n,k} = \frac{1}{\sqrt{2}}(\varphi_{n+1,2k} + \varphi_{n+1,2k+1})$$

et

$$\psi_{n,k} : I_{n,k} \rightarrow]-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}[$$

$$\psi_{n,k} = \frac{1}{\sqrt{2}}(\varphi_{n+1,2k} - \varphi_{n+1,2k+1})$$

Remarque :

La base de Haar va être définie de la manière suivante :

$$\mathcal{H}_n = \{\varphi_{0,0}\} \bigcup_{j=0}^{n-1} \{\psi_{n,k}; k = 0, 1, \dots, 2^j - 1\}$$

2.3 Transformée en ondelettes

La transformée en ondelettes offre la possibilité d'analyser un signal simultanément dans le domaine du temps et celui des fréquences. **La transformée en ondelettes** est un *opérateur linéaire, invariant par translation* («glissement d'un objet», sans rotation) et *par dilatation*. Quelle que soit l'échelle et quel que soit l'endroit, l'analyse du signal se fait avec la même fonction, grâce à l'ondelette mère.

On définit les coefficients d'ondelettes d'un signal à partir d'une série de lissages, à des résolutions de plus en plus grossières. Le lissage constitue une technique qui consiste à réduire les irrégularités et singularités d'une courbe en mathématiques, pour atténuer ce qui peut être considéré comme une perturbation ou un bruit de mesure.

Dans la partie qui suit, nous allons regarder l'algorithme qui permet de décomposer un signal en une seule dimension. Dans le cas d'une image, qui représente un signal en deux dimensions, au lieu d'un tableau de coefficients on va plutôt construire des matrices de coefficients d'ondelettes.

Définition :

Soit f une fonction continue en fonction d'un temps t (signal en une dimension), qu'on va étudier sur l'intervalle $[0, 1]$. La modélisation du signal va être réalisée sur des petits intervalles, suivant les formules, en reprenant les notions définies auparavant :

$$c_{n,k} = \langle f, \varphi_{n,k} \rangle = \frac{1}{\sqrt{2}}(c_{n+1,2k} + c_{n+1,2k+1})$$

$$f \in V_n : \quad f = c_{0,0}\varphi_{0,0} + \sum_{j=1}^{n-1} \sum_{k=0}^{2^j-1} d_{j,k}\psi_{j,k}$$

$$d_{j,k} = \langle f, \psi_{j,k} \rangle = \frac{1}{\sqrt{2}}(c_{j+1,2k} - c_{j+1,2k+1}) \quad (1)$$

Les calculs mènent à la décomposition :

$$f = \sum_{k=0}^{2^{n-1}-1} c_{n-1,k}\varphi_{n-1,k} + \sum_{k=0}^{2^{n-1}-1} d_{n-1,k}\psi_{n-1,k}$$

La compression consiste à mettre à 0 tous les coefficients $d_{n,k}$ (dans (1)) qui seront considérés comme négligeables. **Nous effectuons cette compression avant la reconstruction** (réalisée aussi avec un algorithme récursif).

2.4 Applications

En ce qui consiste les applications, la **compression** peut être réalisé dans les deux dimensions : *le signal 1D* correspond au format MP3 et *le signal 2D* au format JPEG ou PNG. De plus, le **débruitage** d'un signal en 1D consiste à enlever les parasites d'une conversation brouillée, rallonger ou réduire le tempo d'un signal audio, ou bien pour certaines applications pour le radar, les satellites, le scanner ou réalisation des échographie. En ce qui concerne le débruitage d'un signal en 2D, cela signifie de lisser une image.

La compression des images a fait l'objet principale d'étude du stage.

3 Algorithmes & implémentation en Python

3.1 Organisation des fichiers et structures utilisées

Le langage choisit a été Python, pour une implémentations facile de la lecture et de la reconstruction des images. Ainsi, le fichier *main.py* regroupe des tests effectués sur des images de taille $2^n \times 2^n$ pixels. Les fonctions se retrouvent dans les fichiers *images.py* et *haar.py*.

Pour la compression, pour une image quelconque, **l’algorithme suit les étapes suivantes** [3] :

1. **lecture de l’image** (*read_image* : retourne la matrice de l’image en noir et blanc et sa taille)
2. l’appel à la fonction *HaarImageCompression* qui prends en argument la matrice, sa taille (n) et un ε qui indique le seuil des coefficients négligeables :
 - *create_matrix* : création de la matrice de la base de Haar (A), pour la bonne taille n
 - *multiply_matrix* : la multiplication $N = A^t \times M \times A$
 - *remove_small_values* : filtrage des valeurs inférieurs à ε de la matrice N
 - *new_matrix* : la matrice de l’image compressée obtenue avec la formule $M' = (A^{-1})^t \times N \times A^{-1}$
3. **affichage de l’image compressée** en noir et blanc (*generate_image*)

3.2 Répertoire GitHub

Le code, accompagné d’une documentation et du présent rapport, se retrouve au lien suivant : <https://github.com/inaelenacampan/StageCompressionDeDonnees>.

4 Résultats

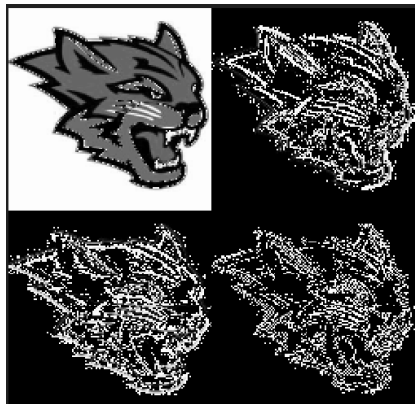
4.1 Tests et observations



((a)) Image originale (256 x 256, 11.4 Ko, 24 bit)



((b)) Image en noir et blanc



((c)) Décomposition niveau 1, $\varepsilon = 200$



((d)) Image synthétisée (256 x 256, 5.5 Ko, 8 bit)

FIGURE 1 – Étapes intermédiaires de la compression de l'image cat.jpg

On remarque dans cet exemple une compression d'environ 50% de la taille originale de l'image, sans perdre les détails essentiels de la photo. Dans la sous-

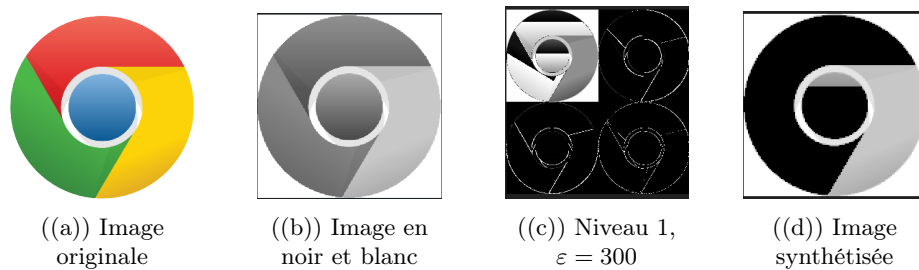


FIGURE 2 – Un deuxième exemple avec l'image Google.png

figure *c*, on observe une décomposition de niveau 1 qui met en évidence des informations sur notre image dans chaque petit bloc (haut-gauche pour les coefficients approchés et les autres pour les coefficients détaillés), qui regroupés, après le filtrage, vont donner l'image synthétisée (sous-figure *d*). [4]

4.2 Améliorations possibles

Il est vrai que le code réalisé connaît des limitations sur le format des images, mais aussi sur les résultats fournis. Ce projet peut être amélioré : en effet, la fonctionnalité de compression des images en couleur directement pourrait être implémentée, en décomposant l'image initiale en trois matrices pour les couleurs rouge, bleu et vert.

De plus, en utilisant les notions mathématiques présentées dans la deuxième section, on pourrait envisager plusieurs niveaux de compression et comparer les résultats obtenus.

5 Poursuite des recherches et conclusion

5.1 Nouvelle application

Dans la deuxième partie du stage, l'objectif a été de voir une nouvelle application concernant les calculs en électromagnétisme [5], qui utiliserait la base de Haar. On définit un **dipôle électrique** comme l'ensemble de deux charges électriques ponctuelles très proches, égales et de signes contraires. L'objectif de cette partie de stage était de pouvoir, à l'aide de la base de Haar, délimiter un champ en plusieurs zones, ou l'application m (définie dans le paragraphe qui suit) soit approximativement constante.

Ainsi, on pose Ω un ouvert borné de \mathbb{R}^3 et l'application $m \in F(\Omega, \mathbb{R}^3)$ (champs de vecteurs), avec le but de construire une nouvelle application $C \in L(F(\Omega, \mathbb{R}^3), V_N)$, où $V_N = \text{Vect}(v_n)_{n \in \{1, 2, \dots, N\}}$ une famille libre.

$$C(m) = \sum_{i=1}^N m_n V_n (\text{avec } m \text{ presque constant})$$

Ensuite, pour simplifier le problème, nous redéfinissons $m \in L^2(\mathbb{R}^3, \mathbb{R}^3)$ avec $\text{supp}(m) \subset \Omega$ (Ω ouvert borné de \mathbb{R}^3).

Définition :

On pose :

$$D'(\mathbb{R}^3, \mathbb{R}^3) = \begin{cases} \text{rot} & h_d(m) = 0 \\ \text{div} & h_d(m) = -\text{div } m \end{cases}$$

De plus,

$$\forall x \in \mathbb{R}^3, \quad h_d(m)(x) = -\frac{1}{4\pi} \int_{\Omega} \nabla_y \text{div}_y \left(\frac{m(y)}{|x-y|} \right) dy \quad (2)$$

qui est équivalente à la formule :

$$\forall x \in \mathbb{R}^3, \quad h_d(m)(x) = -\frac{1}{4\pi} \int_{\Omega} \left(\frac{\text{Id}}{|x-y|^3} - \frac{3|x-y|^t|x-y|}{|x-x_i|^5} \right) m dy \quad (3)$$

Observation : $h_d(m)$ est une application linéaire.

Pour montrer l'équivalence, nous avons utilisé plusieurs propriétés du gradient et nous avons aussi regarder l'équation de Laplace :

$$\begin{aligned}
h_d(m)(x) &= \nabla \Delta^{-1} \operatorname{div} m = \nabla \left(\left(\frac{1}{4\pi r} \right) * \operatorname{div} m \right) = \\
&= \left(\frac{1}{4\pi r} \right) * (\nabla \operatorname{div} m) = \nabla \operatorname{div} \left(\frac{1}{4\pi r} \right) * m
\end{aligned}$$

Ensuite, nous posons $r = \sqrt{x^2 + y^2 + z^2}$, d'où l'équivalence :

$$\nabla \operatorname{div} \left(\frac{1}{4\pi \sqrt{x^2 + y^2 + z^2}} \right) = \nabla \left(-\frac{1}{8\pi} (2x + 2y + 2z)(x^2 + y^2 + z^2)^{-\frac{3}{2}} \right)$$

Cette observation nous permet de conclure.

Ensuite, le but est de calculer $h(m)$ en dehors de Ω . Pour cela, on doit trouver en premier une formule approchée moins coûteuse.

Remarque :

On peut approcher avec :

$$\forall x \in \mathbb{R}^3 \setminus \Omega : h_{\text{dip}}(m)(x) = \sum_{i=1}^n -\frac{1}{4\pi} \left(\frac{\operatorname{Id}}{|x - x_i|^3} - \frac{3|x - x_i|^t |x - x_i|}{|x - x_i|^5} \right) \mu_i \quad (4)$$

I_d représente la matrice identité en dimension 3 et μ_i caractérise le dipôle

Il est donc question de comparer de manière rigoureuse, en dehors de Ω , les équations (3) et (4), en remplaçant dans la quatrième x_i par le centre de gravité de Ω et μ_i par m .

Cette deuxième partie du stage a été la plus difficile et laborieuse, car avant de pouvoir rédiger des preuves, j'ai dû me renseigner sur des notions pas encore traitées dans mes cours, comme par exemple les distributions, les équations aux dérivées partielles (en particulier l'équation de Laplace [6]) ou encore des notions d'analyse fonctionnelle [7].

Pour effectuer la comparaison, nous notons :

$$g(x, y) := \frac{\operatorname{Id}}{|x - y|^3} - \frac{3|x - y|^t |x - y|}{|x - x_i|^5}$$

$$h_d(m)(x) - h_{\text{dip}}(m)(x) = \left(\int_{\Omega} g(x, y) m dy \right) - g(x, G(\Omega)) |\Omega| m \quad (5)$$

$$h_d(m)(x) - h_{\text{dip}}(m)(x) = \int_{\Omega} (g(x, y) - g(x, G(\Omega))) m dy \quad (6)$$

Les calculs n'ont pas été menés jusqu'au but, par manque de temps, mais l'astuce pour résoudre (6) serait d'utiliser un développement limité de Taylor en dimension 3, ce qui permettrait une majoration par $k^{-\alpha}$, qui va se rapprocher de 0 en limite.

5.2 Conclusion

Mon expérience de stage a été très enrichissante. D'un point de vue scientifique, ça m'a poussé à chercher beaucoup plus loin sur des notions que j'avais déjà rencontré dans mes cours, mais aussi d'apprendre par moi même des concepts qui manquaient afin d'avancer sur les tâches qui m'ont été confiées par mon encadrant. J'estime que cette expérience va être utile pour ma troisième année de licence, mais pas seulement. Ce stage m'a aussi donné un aperçu de la vie d'un chercheur, carrière que j'aimerais moi-même poursuivre plus tard.

Références

- [1] <https://www.bibmath.net/>.
- [2] Gaëtan Delers, Vincent Nolot (mai 2008) *Projet de mathématiques, Licence Semestre 6 - Compression d'un signal par les ondelettes de Haar* : <http://nolot.perso.math.cnrs.fr/ProjetL3.pdf>.
- [3] Rémi Pauchet, Damien Velleman *Compression d'images par ondelettes* - en ligne, à l'adresse web suivante : <http://rpauchet.chez.com/lapageTIPE/TIPEixl/TIPEondelettes/index.html>.
- [4] Rasolomanana J.F., Randriamitantsoa P.A., Rakotomalala M.A. (2016) *Compression d'images par ondelettes de Haar*.
- [5] Gérard Smadja, Yannick Copin (2021) *Le champs électromagnétique - Des notions fondamentales aux méthodes avancées*, Ellipses, Paris.
- [6] Raphaël Danchin (année 2020 - 2021) *Cours d'équations aux dérivées partielles - M1 de Mathématiques*.
- [7] Haïm Brezis (1983) *Analyse fonctionnelle - Théorie et applications*, Masson, Paris.