# Stéganographie robuste et sans erreur dans des images JPEG en utilisant les sorties des codeurs JPEG

J. Butora[1] P. Puteaux[1] P. Bas[1]

[1] Univ. Lille, CNRS, Centrale Lille, UMR 9189 CRIStAL, F-59000 Lille, France

## Résumé

*La stéganographie robuste est une technique qui consiste à dissimuler des messages secrets dans des images de manière à ce que le message puisse être récupéré après que cette image a été post-traitée. L'une des opérations de post-traitement les plus populaires est la recompression au format JPEG. La plupart des méthodes de stéganographie actuelles ne fournissent qu'une garantie probabiliste de récupération du secret et ne sont donc pas sans erreur. Cela pose un problème majeur lorsque le message secret a été chiffré avant son insertion : le changement d'un seul bit du message chiffré peut complètement empêcher sa reconstruction.*

*Dans nos travaux, nous proposons de créer un ensemble robuste de coefficients DCT en examinant leur comportement pendant la recompression, ce qui nécessite l'accès au codeur JPEG ciblé. Cette méthode est sans erreur par construction, ce qui signifie que le message inséré pourra toujours être reconstruit.*

## Mots clefs

Sécurité multimédia, Compression JPEG, Stéganographie.

# 1 Introduction

Avec l'utilisation massive des réseaux sociaux et des plateformes de partage, le schéma classiquement considéré en stéganographie, qui implique un canal sans perte entre Alice (la stéganographe qui insère une charge utile) et Bob (le stéganographe qui décode la charge utile), n'a plus de sens dans de nombreux scénarios pratiques. En effet, durant la transmission, un transcodage de l'image stégo (*i.e.* image qui contient un message), tel qu'une recompression JPEG par exemple, a lieu. Le canal de transmission entre Alice et Bob peut ainsi être considéré comme étant bruité. Dans ce cas, le décodage sans erreur de la charge utile n'est plus possible si Alice utilise un schéma d'insertion classique conçu pour une transmission sans perte, comme par exemple J-Uniward [1], UERD [2], ou encore J-Mipod [3] dans le domaine JPEG.

En raison du contexte pratique de la stéganographie, la robustesse du schéma doit être extrêmement élevée, car un seul bit erroné peut compromettre toute la transmission. Les schémas stéganographiques sans erreur sont donc recommandés.

Plusieurs travaux se sont intéressés à la problématique de robustesse au redimensionnement d'une image [4, 5]. En revanche, à notre connaissance, la stéganographie robuste à la recompression JPEG et sans erreur n'a pas encore été étudiée. En effet, même si les méthodes *Sign Steganography Revisited* (SSR) [6] et *MINImizing Channel Error Rate* (MINICER) [7] peuvent fournir de faibles taux d'erreur dans certains scénarios limités, elles ne sont pas robustes dans un contexte pratique.

Dans nos travaux, nous proposons une méthode de stéganographie sans erreur dans le domaine JPEG et robuste à la recompression JPEG. Les avantages de ce système sont les suivants : absence d'erreur (garantie que l'image stégo générée ne produira pas d'erreur pendant l'insertion), absence d'informations auxiliaires pour l'extraction de la charge utile et absence d'utilisation de codes correcteurs d'erreur. En outre, l'insertion permet de garantir la robustesse même lorsqu'une opération de filtrage est appliquée avant la recompression.

# 2 Schéma de sécurité considéré

Alice envoie une image stégo, notée $\mathbf{S}_1$, sur une plateforme qui compresse $\mathbf{S}_1$ en $\mathbf{S}_2$ à l'aide d'un codeur JPEG. Bob, le destinataire, télécharge l'image depuis la plateforme et tente de décoder la charge utile insérée. Comme nous supposons que l'image cover (*i.e.* image qui ne contient pas de message) d'Alice, notée $\mathbf{C}_1$, est également au format JPEG, l'image est par conséquent doublement compressée.

Nous supposons qu'Alice connaît à la fois le schéma et les paramètres de codage utilisés par la plateforme. En pratique, cela peut être fait en analysant les images téléchargées depuis la plateforme. La matrice de quantification JPEG est publique et le schéma de codage peut généralement être identifié en comparant l'image téléchargée depuis la plateforme avec les sorties de différents codeurs.

Notons que cela suit le principe de Kerckhoffs, selon lequel tout ce qui n'est pas lié au secret de l'application (ici, le fait qu'une charge utile est potentiellement transmise à Bob) doit être considéré comme public. Dans cette configuration, Eve, une attaquante qui cherche à détecter la présence du message secret inséré, a accès à la plateforme. Elle peut également télécharger des images depuis la plateforme pour tenter de différencier un contenu cover téléversé sur la plateforme ($\mathbf{C}_2$) d'un contenu stégo téléversé

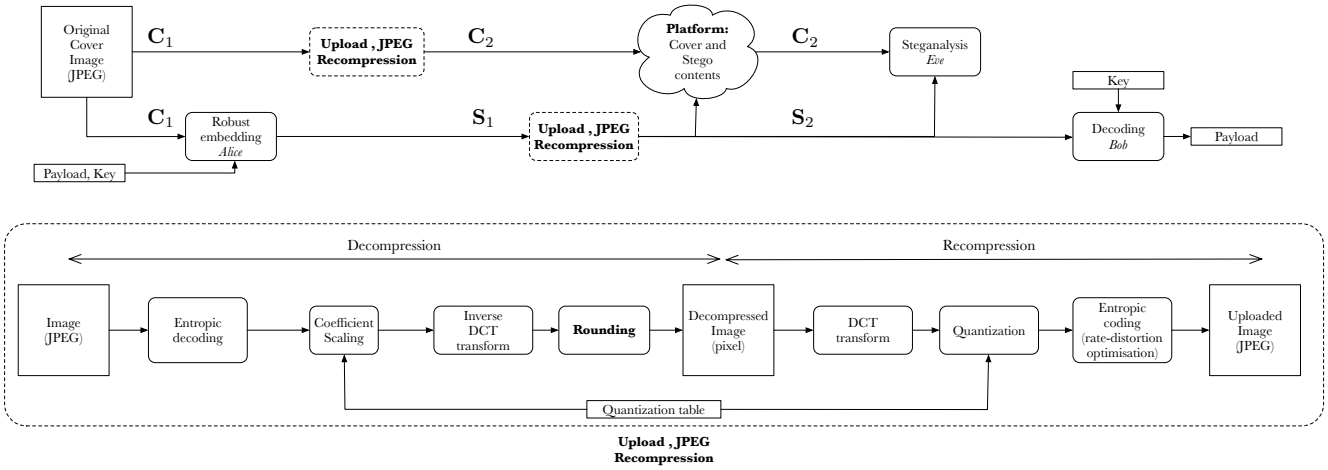


FIGURE 1 – *Schéma considéré pour une stéganographie robuste après compression JPEG : le diagramme du haut représente l'ensemble de la chaîne de traitement et les différents acteurs (Alice, Bob et Eve). Le diagramme du bas décrit les différentes opérations impliquées dans le processus de téléversement sur la plateforme.*

sur la plateforme ($\mathbf{S}_2$).
Afin de minimiser la différence entre l'image téléversée sur la plateforme (respectivement $\mathbf{C}_1$ ou $\mathbf{S}_1$) et l'image téléchargée depuis la plateforme (respectivement $\mathbf{C}_2$ ou $\mathbf{S}_2$), nous supposons que l'image téléversée sur la plateforme est codée avec les mêmes paramètres de codage que l'image téléchargée depuis la plateforme. Nous supposons que la compression JPEG est le seul processus effectué lors du téléversement sur la plateforme, ce qui signifie que l'image téléversée est déjà redimensionnée en vue d'éviter toute opération ultérieure de redimensionnement. Cette hypothèse est valable en pratique car cela est généralement le cas sur de nombreuses plateformes telles que Facebook, WhatsApp et FlickR.
Ces différents éléments sont repris dans le schéma général illustré dans la figure 1.

# 3 Robustesse d'un coefficient DCT

Pour faciliter la compréhension de la notion de robustesse d'un coefficient DCT, nous introduisons plusieurs définitions.

**Définition 3.1** (Modes traités)**.** Etant donné le $k$-ème mode DCT (avec un ordre pré-défini sur les modes), $k \in \{1, \ldots, 64\}$, $\mathcal{P}_k = \{1, \ldots, k-1\}$, $\mathcal{P}_1 = \emptyset$ désigne l'ensemble de tous les modes qui ont déjà été traités lors de l'exécution de l'algorithme.

**Définition 3.2** (Pseudo-stégo)**.** La $k$-ème pseudo-stégo est l'image cover dans laquelle l'insertion a déjà été réalisée dans les modes de $\mathcal{P}_k$.

Nous allons décrire le procédé pour un bloc de $8 \times 8$ coefficients DCT de la $k$-ème image pseudo-stégo $\mathbf{c} = \{c_n\}_{n=1}^{64} \in \mathbb{Z}^{64}$. Soit $i_n \in \mathbb{Z}^{64}$ un vecteur constitué de $i \in \mathbb{Z}$ à la $n$-ème coordonnée et de zéros partout ailleurs. Nous considérons $R(\mathbf{c}, i_n) \in \mathbb{Z}^{64}$ les coefficients recompressés de $(\mathbf{c} + i_n)$.

**Définition 3.3** (Coefficient robuste)**.** Un coefficient $c_k$ est dit robuste à un changement $i \in \{-1, 1\}$ lors de l'insertion, si, durant la recompression il :

(R1) : ne modifie pas les modes déjà parcourus :

$$\forall l \in \mathcal{P}_k : \; R(\mathbf{c}, i_k)_l = R(\mathbf{c}, \mathbf{0})_l.$$

(R2) : conserve un changement de $i$ :

$$R(\mathbf{c}, i_k)_k = c_k + i.$$

(R3) : reste inchangé si l'insertion n'implique pas de modification de sa valeur :

$$R(\mathbf{c}, \mathbf{0})_k = c_k.$$

Les ensembles de tous les coefficients robustes à une modification $+1$ et $-1$ sont nommés $\mathcal{R}_k^+$ et $\mathcal{R}_k^-$ respectivement. Si un coefficient n'appartient pas à un ensemble robuste, il est dit non robuste. L'ensemble des coefficients non robustes est nommé $\mathcal{R}_k^0$.

# 4 Synthèse des résultats obtenus

Nous avons évalué la sécurité (du point de vue de la détectabilité) de notre méthode à l'aide d'une stéganalyse par apprentissage automatique. Nous observons que le niveau de sécurité est affecté par tous les éléments du système : le facteur de qualité, le compresseur et l'ordre de parcours des modes DCT. A titre d'exemple, le tableau 1 présente le taux de détection d'erreur obtenu avec notre méthode en utilisant `convert` et un ordre de parcours des modes DCT aléatoire, en fonction de la charge utile considérée (exprimée en bits par coefficient AC non nul (bpnzAC)) et du facteur de qualité QF.

| QF | Charge utile (bpnzAC) | | | |
|---|---|---|---|---|
| | **0.1** | **0.2** | **0.3** | **0.4** |
| 75 | 0.5000 | 0.3999 | 0.2365 | 0.1732 |
| 95 | 0.4997 | 0.4583 | 0.2926 | 0.1386 |

TABLEAU 1 – *Taux de détection d'erreur obtenu en utilisant* `convert` *et un ordre de parcours aléatoire.*

Nous avons montré un lien entre la taille de l'ensemble robuste et le niveau de sécurité obtenu : plus la taille de l'ensemble robuste est petite, plus la détectabilité est importante. En outre, nous pouvons observer un niveau de sécurité inférieur par rapport à la version non robuste de l'algorithme de stéganographie. En effet, de nombreux coefficients dont les coûts d'insertion sont faibles ne sont pas exploitables pour une insertion robuste. Enfin, contrairement à tous les travaux précédents dans le domaine de la stéganographie robuste, notre méthode est sans erreur, ce qui nous permet de garantir une extraction correcte du message secret inséré. La seule exception à cette règle intervient lorsque nous utilisons le codeur Mozjpeg, qui permet d'optimiser le rapport débit/distorsion.

# 5 Conclusion

Dans nos travaux, nous avons introduit une méthodologie pour la stéganographie JPEG robuste à une recompression ultérieure. Bien que le compresseur JPEG doive être connu, cela ne présente aucun obstacle car, en pratique, il est généralement facile d'accéder au codeur. Nous avons tout d'abord introduit la notion de robustesse d'un coefficient DCT. Nous avons alors défini trois ordres de parcours pour les modes : basses vers hautes fréquences, hautes vers basses fréquences et ordre aléatoire. Nous avons montré que ces trois stratégies permettaient d'obtenir des ensembles robustes sensiblement différents. Nous avons ensuite combiné la robustesse des coefficients avec les coûts stéganographiques d'un algorithme de stéganographie non robuste de manière directe afin de rendre l'algorithme plus robuste. En outre, nous avons montré comment réaliser ces opérations dans un contexte pratique en utilisant les codes Syndrome-Trellis.

## Annexe

Cet article est le résumé d'un article soumis dans la revue internationale IEEE Transactions on Dependable and Secure Computing (TDSC), dont une version préliminaire est disponible sur arXiv [8].

# Références


[1] Vojtěch Holub, Jessica Fridrich, et Tomáš Denemark. Universal distortion function for steganography in an arbitrary domain. *EURASIP Journal on Information Security*, 2014(1) :1–13, 2014.

[2] Linjie Guo, Jiangqun Ni, Wenkang Su, Chengpei Tang, et Yun-Qing Shi. Using statistical image model for JPEG steganography : Uniform embedding revisited. *IEEE Transactions on Information Forensics and Security*, 10(12) :2669–2680, 2015.

[3] Quentin Giboulot, Rémi Cogranne, et Patrick Bas. Detectability-based JPEG steganography modeling the processing pipeline : the noise-content trade-off. *IEEE Transactions on Information Forensics and Security*, 16 :2202–2217, Janvier 2021.

[4] Yue Zhang, Xiangyang Luo, Jinwei Wang, Chunfang Yang, et Fenlin Liu. A robust image steganography method resistant to scaling and detection. *Journal of Internet Technology*, 19(2) :607–618, 2018.

[5] Liyan Zhu, Xiangyang Luo, Yi Zhang, Chunfang Yang, et Fenlin Liu. Inverse interpolation and its application in robust image steganography. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(6) :4052–4064, 2021.

[6] Xiaoshuai Wu, Tong Qiao, Yanli Chen, Ming Xu, Ning Zheng, et Xiangyang Luo. Sign steganography revisited with robust domain selection. *Signal Processing*, 196 :108522, 2022.

[7] Kai Zeng, Kejiang Chen, Weiming Zhang, Yaofei Wang, et Nenghai Yu. Improving robust adaptive steganography via minimizing channel errors. *Signal Processing*, 195 :108498, 2022.

[8] Jan Butora, Pauline Puteaux, et Patrick Bas. Errorless robust JPEG steganography using outputs of JPEG coders. *arXiv preprint arXiv :2211.04750*, 2022.