CR_4

Avancée

Une petite base de données de 7 photos de tableaux a été constituée afin de pouvoir tester notre algorithme.

Codage du chiffrement des images :

- Décomposition RGB de l'image
- Pour chaque composante : passage de l'image une image pour laquelle un pixel est la valeur moyenne d'un bloc de 16*16 → réduction de la taille de l'image
- Calcul de la suite de Chebyshev à partir de x_0 = 0.3 et k = 4.0 (la clé choisie 0.3 et 4.0)
- Tri des valeurs de cette suite
- Pour chaque composante, permutation des pixels selon le tri de la suite
- Regroupement des composantes
- Remise à l'échelle

Codage du déchiffrement des images (même principe que pour le chiffrement, mais en inversant les permutations)

→ ERREUR : MAUVAIS RÉSULTAT APRÈS DÉCHIFFREMENT (donc problème lors du chiffrement ou du déchiffrement)

On a inversé les étapes de recomposition à partir des 3 composantes et de remise à l'échelle, ce qui nous à permis d'obtenir un résultat.
On a donc effectué notre test sur l'image suivante :

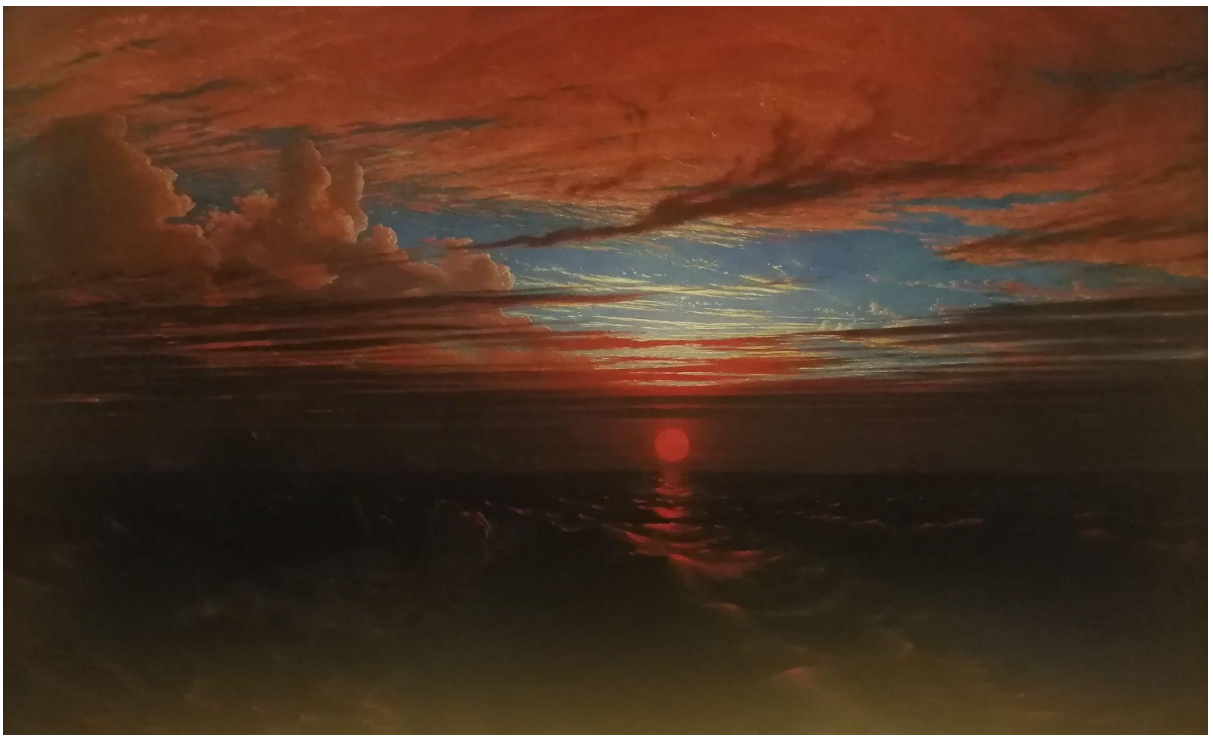

Image de base

Les résultats ont été :

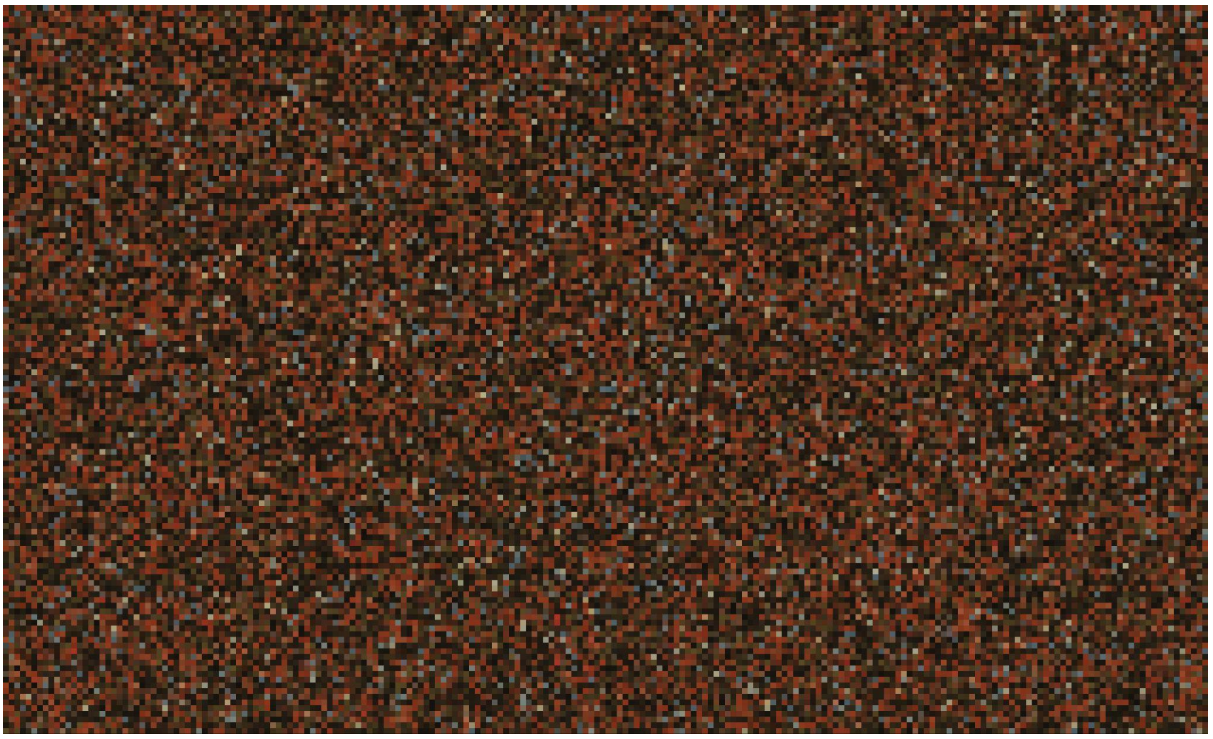

Image chiffrée

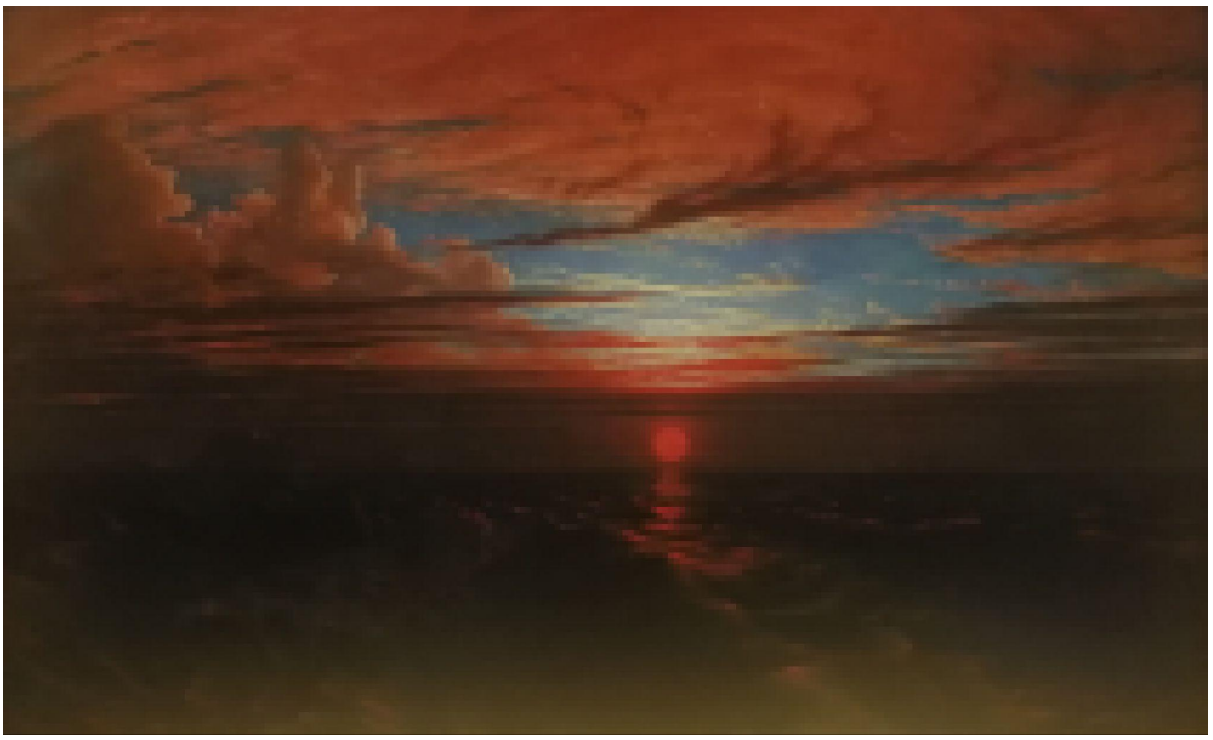

Image déchiffrée

Avec un PSNR entre l'image de base et l'image déchiffrée égale à 38.87dB, montrant que le déchiffrage est très correct.

Comme on peut le voir, le fait d'utiliser des blocs pour chiffrer l'image, amène une pixellisation de cette image, toutefois, nous craignons que si les blocs sont plus petits, il soit difficile de les retrouver à partir d'une photo.

Pour le recalage, on commence par appliquer un recalage sur une image sans rotation (créée par ordinateur à partir de notre image chiffrée).

On identifie sur cette image le cadre de notre image chiffrée.
On identifie ensuite l'intérieur du cadre pour se débarasser du cadre.

Reste alors la partie un peu plus compliquée.
On applique une détection de contour à notre image pour identifier la taille des blocs. Pour chaque bloc, on récupère le pixel central comme valeur du bloc (on peut aussi faire une moyenne pondérée par le centre), et on stocke ces pixels dans une nouvelle image qui nous suffit de redimensionner en bloc de 16*16 pixels pour ensuite pouvoir appliquer notre chiffrement.