



Compte Rendu N°2

FONTAINE Emmanuel
HARCHA Ibrahim

Master IMAGINE
Faculté des Sciences
Université de Montpellier

03/03/2024 - 10/03/2024

Table des matières

1	Introduction	3
2	Extraction du Bruit (Programme : extract_noise.py)	3
3	Error Level Analysis (ELA) (Programme : ela_detection.py)	3
4	Double Quantification (Programme : double_quantificazion_detect.py)	4
5	Filtre DCT (Programme : dct_filter.py)	5
6	Conclusion	6

1 Introduction

Dans cette première semaine du projet, nous nous sommes attardés sur la détection par compression, ainsi, nous avons pu développer différents programmes. L'objectif de ces programmes est de détecter la falsification d'images en utilisant différentes techniques d'analyse. Chaque programme a une approche distincte pour repérer des altérations potentielles, telles que la compression multiple, l'ajout de bruit, les anomalies fréquentielles, et l'analyse d'erreurs de compression.

2 Extraction du Bruit (Programme : extract_noise.py)

Dans un premier temps, nous avons créé un extracteur de bruit. L'objectif de ce programme est d'extraire le bruit présent dans une image en utilisant un filtre de médiane. Cela peut révéler des altérations subtiles ou des manipulations de l'image.

Techniques utilisées :

- Application d'un filtre de médiane pour extraire le bruit.
- Soustraction de l'image filtrée de l'image originale pour obtenir le bruit.

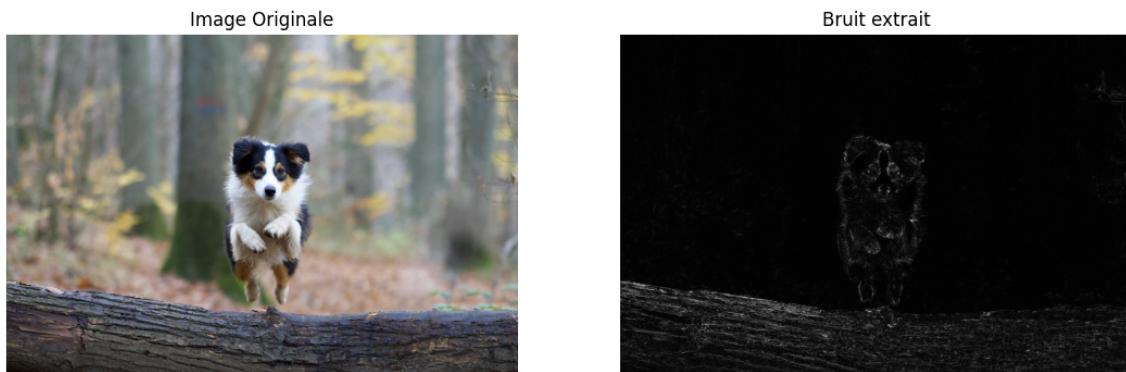


FIGURE 1 – Extraction du bruit

3 Error Level Analysis (ELA) (Programme : ela_detection.py)

Dans un second temps, nous nous sommes intéressés, sur l'analyse, d'erreur de niveau d'une image. Le programme `ela_detection` utilise l'analyse d'erreur de niveau pour mettre en évidence des zones où des altérations ou manipulations peuvent avoir eu lieu. Il compare l'image originale avec une version réenregistrée après compression JPEG.

Techniques utilisée :

- Compression JPEG de l'image originale avec deux niveaux de qualité différents.
- Calcul de l'image ELA (Error Level Analysis) en comparant les deux versions.



FIGURE 2 – ELA

4 Double Quantification (Programme : double_quanti-ficazion_detect.py)

Ici, ce programme vise à identifier les incohérences résultant d'une double quantification lorsqu'un objet étranger est inséré dans une image JPEG. La région altérée apparaîtra plus claire que le reste de l'image.

Techniques utilisée :

- Conversion de l'image en espace de couleur YCrCb.
- Compression de l'image avec deux niveaux de qualité différents.
- Comparaison des deux images compressées pour détecter des différences.
- Création d'une carte des incohérences mettant en évidence les zones altérées.



FIGURE 3 – Double Quantification

5 Filtre DCT (Programme : `dct_filter.py`)

Enfin, nous avons utilisé le filtre DCT qui a pour but de détecter des incohérences dans les fréquences d'une image en appliquant la transformée en cosinus discrète (DCT). Son objectif est de mettre en évidence des zones potentiellement altérées par des variations significatives dans les hautes fréquences.

Techniques utilisée :

- Conversion de l'image en niveaux de gris ou en espace de couleur YCrCb.
- Division de l'image en blocs 8x8 pixels.
- Application de la transformée DCT à chaque bloc.
- Mise en évidence des hautes fréquences (filtre fréquentiel).
- Analyse des anomalies fréquentielles.

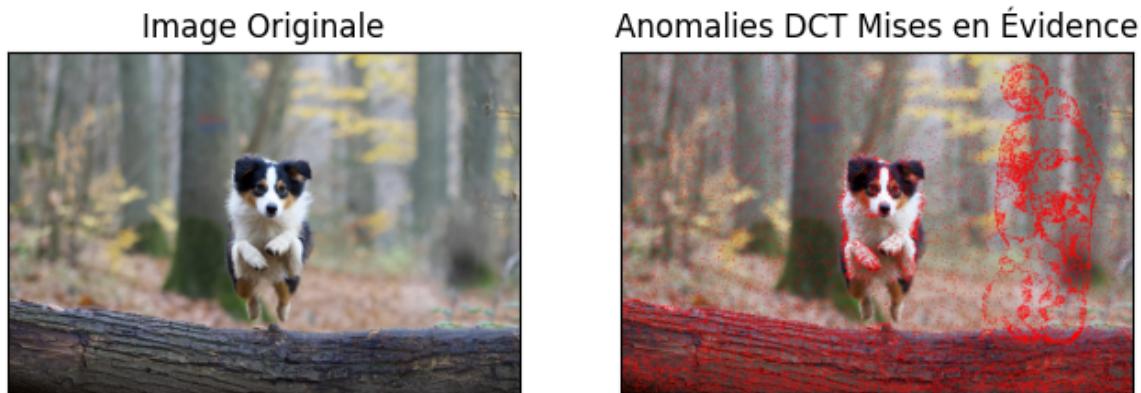


FIGURE 4 – Filtre DCT

Note : On remarque que ici le filtre DCT a bien repéré une anomalie dans l'image. En effet, ceci est une image retouchée auquel on a enlevé un élément. La méthode DCT a permis de mettre en évidence des variations significatives dans les hautes fréquences de l'image, ce qui indique une possible altération. Ces résultats démontrent l'efficacité de l'approche fréquentielle du filtre DCT pour détecter des modifications subtiles ou des suppressions dans une image.



(a) Image Original

(b) Image Falsifiée

FIGURE 5 – Image I0 et I1

6 Conclusion

En conclusion, au cours de cette première semaine, nous avons initié le développement de premières approches visant à déterminer l'authenticité d'une image. Nous avons débuté par la détection par compression, offrant ainsi un point de départ pour explorer de nombreuses autres possibilités de détection, notamment par l'utilisation de divers filtres. Chaque méthode présentée possède ses avantages spécifiques, pouvant être ajustée en fonction des caractéristiques propres à une analyse d'image. De plus, la combinaison de ces techniques peut renforcer la robustesse de la détection face à diverses manipulations.