

UNIVERSITÉ DE MONTPELLIER
M1 IMAGINE



Compte Rendu n°2

HAI804I & HAI809I : TRAITEMENT D'IMAGES & CODAGE ET COMPRESSION
MULTIMÉDIA

Étudiants :

Arthur VILLARROYA-PALAU
Daniel BLANCHARD
Marie BOCQUELET

Année : 2022 – 2023

Encadrant :

William PUECH

État de l'art de la détection falsifications dans les images

Enjeux

La détection de la falsification d'une image numérique est devenue un enjeu important dans de nombreux domaines, notamment dans le domaine de la sécurité (images de propagandes), de la justice (falsification de preuves), de la presse (fausses informations) et de l'art (fraude artistique). Nous pouvons notamment noter l'exemple des missiles Iraniens dont les autorités avaient retouchés l'image du pas de tir des missiles afin de cacher le dysfonctionnement de l'un d'entre eux.



Figure 1: Falsification de l'image des missiles

types de falsifications

Les images numériques peuvent être falsifiées de différentes manières, ce qui rend leur détection difficile. Les principaux types de falsifications sont les suivants :

- La modification de la couleur et de la luminosité de l'image pour masquer des informations.
- La suppression ou l'ajout de certains éléments de l'image.
- La superposition de plusieurs images pour créer une nouvelle image.
- Le copié-collé d'élément.

De notre côté lors de ce projet nous nous contenterons de travailler sur la détection de falsification par copié-collé comme pour l'exemple des missiles.

méthodes traditionnelles

Les techniques traditionnelles de détection de la falsification d'une image numérique comprennent :

- L'analyse de la cohérence des couleurs et des textures de l'image pour détecter les zones falsifiées.
- L'analyse de la structure de l'image pour détecter les zones où des éléments ont été ajoutés ou supprimés.
- L'analyse de la compression de l'image pour détecter les zones où l'image a été modifiée.
- L'utilisation de l'intelligence artificielle.

Travail réalisé pendant la semaine

Organisation

- Un dossier par sources
- Img pour les images (Penser à split pour les références et les falsifications)
- SRC les ressources externes
- CR pour les .pdf

Utilities.py

- Juste un fichier pour rassembler les ressources
- Rajouter des opérations en plus, linkage,... pour l'instant teste les fonctions des entêtes

Analyse.py

- Classe timer pour analyse assez générale de (complexité, performances, ...)
- Fonctions classiques
- exploration d'arbres (profondeur, largeur, ...)
- Fonctions complexité générales (temps, poids, complexité, ...)
- Fonctions complexité 2 (mémoire réellement utilisée, estimation différence entre mémoire utilisée et mémoire maximale disponible, simule l'accès à strvar par une opération lambda strval)

Image.py

- Classe pour opérations sur les images qui utilise sklearn+numpy
- Liste format valides couleurs + grey
- Création et stockage de l'image dans un tableau en une dimension avec toutes les informations nécessaire (format, path, taille, ...)
- Affichage de l'image
- Traitement de l'image (kmean, comparaison, segmentation)

Git :

<https://github.com/Cubevayster/Projet-Image>