



# Master 1 IMAGINE

GASC Thibault

DIAB Ingo

## Compte Rendu 2 Projet : Détection de zones copiées-déplacées dans des images



Année Universitaire 2022-2023

## Table des matières

I	Enjeux et types de falsifications	2
II	Méthodes classique	2
III	Convolutional Neural Network	2
IV	Dataset	3

## I Enjeux et types de falsifications

Avec l’explosion d’internet et des réseaux sociaux, la transmission d’informations, que ce soit textes, images, vidéos, ... se fait de plus en plus rapidement et facilement. Mais cet avantage peut devenir un inconvénient lorsque ces informations sont volontairement ou involontairement faussées. Il est donc nécessaire de mesurer le plus précisément possible la fiabilité d’une information et, dans ce projet, la véracité d’une image.

Nous avons pu trouver 3 différents types de falsification :

- Copier-déplacer (Copy Move Forgery) qui consiste à copier une zone de l’image qui sera ensuite collé ailleurs dans la même image
- Copier-cr  er (Splicing Forgery) qui consiste à copier une zone d’une premi  re image qui sera ensuite coll   dans une seconde image
- Imitation (Imitation Forgery) qui consiste à modifier un texte en imitant la typographie du texte existant (surtout utilis   pour la falsification de documents num  riques)

## II M  thodes classique

Pour ce projet, nous nous int  ressons au type copier-d  placer. Nous avons trouver deux m  thodes pour d  tecter ce type de falsification :

- M  thode par points d’int  r  t : cette m  thode consiste, dans un premier temps, à extraire les points cl  s (points d’int  r  ts) de l’image en utilisant des m  thodes connues, comme l’algorithme SIFT (Scale Invariant Feature Transform) ou le d  tecteur de Harris. Chaque point d’int  r  t est ensuite associ   à un vecteur. Lorsque cela est fait, nous mettons en correspondance des points d’int  r  ts. Si nous trouvons un pattern similaire, alors l’image est falsifi  e.
- M  thode par la compression JPEG : lors de la compression JPEG, l’image subit une quantification. Cette   tape induit une grande perte d’information. Ces informations perdues lors de la quantification ne peuvent pas   tre reconstitu  es et cela am  ne à l’apparition de discontinuit  s au bord des blocs de l’image d  compress  e.

## III Convolutional Neural Network

En parall  le, nous pouvons d  velopper un mod  le de Deep Learning (nous voulons que ce soit le mod  le qui rep  re les caract  ristiques qui montre

qu'une image est falsifiée) capable de classer les images en deux groupes d'images, les images falsifiées et les images non falsifiées. Nous utilisons donc une classification binaire (falsifiée ou non, ce qui implique que l'output layer de notre réseau de neurone aura un seul neurone).

Nous aurons ensuite plusieurs hidden layers ainsi que plusieurs neurones par hidden layer (paramètres qui changeront durant le développement pour avoir un modèle plus fiable).

Avant de passer nos images à l'input layer, nous allons lui appliquer une suite de convolutions 2D et max/average pooling afin de réduire sa taille tout en gardant la spatialité de l'image. Une fois cette suite de convolution/pooling terminée, nous allons aplanir l'image et la donner à notre modèle.

Pour entraîner notre modèle, nous avons trouvé un dataset d'images de même tailles. Ce dataset contient 200 sets. Chaque set contient une image originale, une image falsifiée de type copié-déplacé ainsi que des variantes de ces deux images (image plus sombre, image plus clair, ...). Ces images seront labellisée (apprentissage supervisé). Nous allons séparer ces 200 sets en deux parties, un jeu d'apprentissage et un jeu de test (70-80% pour apprendre et 20-30% pour tester). Cette séparation sera fait plusieurs fois (K-Folds). A chaque fois, nous calculerons l'accuracy du modèle sur le jeu de test ainsi que la loss (écart entre le résultat fourni et le résultat attendu).

## IV Dataset

Le dataset étant de 3 Go (.rar), il n'a pas pu être push sur notre Github, à la place il a été mis sur Google Drive. Voici le lien : [https://drive.google.com/file/d/10FXiri0\\_XNcZBNdxLUTXqHam-ZMP5akt/view?usp=share\\_link](https://drive.google.com/file/d/10FXiri0_XNcZBNdxLUTXqHam-ZMP5akt/view?usp=share_link)

Ce dataset nous permettra d'entraîner/tester notre modèle et nous pourrions piocher dans ces images pour tester l'implémentation des méthodes classique. Ce dataset serait donc commun aux deux parties de ce projet. Pour comparer les performances entre méthodes classiques et deep learning, nous pourrions extraire du dataset certaines images. Ainsi ces images ne seraient pas utilisées pour l'apprentissage de notre modèle et la comparaison des résultats ne seraient donc pas biaisé (on nous pourrions simplement trouver un autre dataset ne contenant pas les mêmes images).