



UNIVERSITÉ DE MONTPELLIER

MASTER 1 IMAGINE

---

# Projet Image et Compression

---

Compte Rendu N°7

*FONTAINE Emmanuel*  
*HARCHA Ibrahim*

*14 avril 2024*

# Introduction

Pour cette septième semaine de projet, comme mentionné précédemment, nous nous sommes intéressés aux méthodes d'apprentissage automatique, en particulier à l'utilisation de CNN (Convolutional Neural Network). Nous avons donc exploré cette approche et décidé d'implémenter un modèle CNN. À la lumière de nos recherches, nous avons constaté que cette méthode est souvent utilisée avec ELA (Error Level Analysis) et peut produire de bons résultats pour évaluer la véracité d'une image.

## 1. Base de données (Dataset)

Pour entraîner notre modèle, nous avons besoin d'une base de données conséquente. Pour cela, nous avons opté pour l'utilisation du jeu de données CASIA2, disponible à ce [lien](#). Cette base de données présente une diversité d'images authentiques et falsifiées, ce qui nous permettra d'entraîner un modèle robuste capable de détecter les altérations d'image.

La dataset CASIA2 comprend **7491 images authentiques** représentant des captures non altérées et **5123 images falsifiées** contenant des altérations telles que des superpositions, des modifications, ou des retouches numériques. Cette variété d'images nous permettra de couvrir un large éventail de scénarios possibles d'altération d'image, améliorant ainsi la capacité de généralisation de notre modèle.

## 2. Création du Modèle et Entraînement

La création du modèle et son entraînement constituent une étape cruciale de notre projet de détection de falsification d'images. Nous avons conçu un réseau de neurones convolutionnels (CNN) adapté à cette tâche spécifique, en utilisant une architecture bien étudiée pour extraire efficacement les caractéristiques des images.

Après avoir pré-traité nos données et configuré les paramètres d'entraînement, nous avons procédé à l'optimisation du modèle à l'aide d'une fonction de perte appropriée et d'une métrique d'évaluation pertinente. Enfin, nous avons entraîné le modèle sur une grande base de données, en ajustant les hyperparamètres et en utilisant des techniques telles que l'arrêt anticipé pour garantir des performances optimales et éviter le sur-ajustement.

Dans la phase de conception du modèle et pendant l'entraînement, nous avons ajusté plusieurs paramètres, notamment le nombre de couches de convolution, pour améliorer les performances du modèle. Cependant, malgré nos efforts, nous n'avons pas obtenu des résultats conclusifs. Il est probable que le choix du nombre de couches de convolution n'était pas optimal, ce qui souligne la nécessité de poursuivre l'exploration et l'optimisation des architectures de réseaux neuronaux pour la détection de la falsification d'images.

```

Epoch 1/30
214/214 — 100s 457ms/step - accuracy: 0.7844 - loss: 0.4128 - val_accuracy: 0.8566 - val_loss: 0.2732
Epoch 2/30
214/214 — 95s 442ms/step - accuracy: 0.8581 - loss: 0.2807 - val_accuracy: 0.8618 - val_loss: 0.2553
Epoch 3/30
214/214 — 96s 448ms/step - accuracy: 0.8713 - loss: 0.2647 - val_accuracy: 0.8816 - val_loss: 0.2309
Epoch 4/30
214/214 — 95s 442ms/step - accuracy: 0.8881 - loss: 0.2410 - val_accuracy: 0.8908 - val_loss: 0.2152
Epoch 5/30
214/214 — 95s 442ms/step - accuracy: 0.8996 - loss: 0.2167 - val_accuracy: 0.8934 - val_loss: 0.2140
Epoch 6/30
214/214 — 95s 443ms/step - accuracy: 0.9158 - loss: 0.1995 - val_accuracy: 0.9066 - val_loss: 0.2102
Epoch 7/30
214/214 — 95s 441ms/step - accuracy: 0.9194 - loss: 0.1786 - val_accuracy: 0.9263 - val_loss: 0.1926
Epoch 8/30
214/214 — 95s 444ms/step - accuracy: 0.9321 - loss: 0.1694 - val_accuracy: 0.9211 - val_loss: 0.1873
Epoch 9/30
214/214 — 94s 441ms/step - accuracy: 0.9357 - loss: 0.1605 - val_accuracy: 0.9329 - val_loss: 0.1854
Epoch 10/30
214/214 — 95s 442ms/step - accuracy: 0.9398 - loss: 0.1505 - val_accuracy: 0.9211 - val_loss: 0.1926
Epoch 11/30
214/214 — 98s 457ms/step - accuracy: 0.9421 - loss: 0.1472 - val_accuracy: 0.9395 - val_loss: 0.1691
Epoch 12/30
214/214 — 97s 455ms/step - accuracy: 0.9509 - loss: 0.1348 - val_accuracy: 0.9263 - val_loss: 0.1706
Epoch 13/30
...
214/214 — 95s 445ms/step - accuracy: 0.9718 - loss: 0.0886 - val_accuracy: 0.9368 - val_loss: 0.1737
Epoch 23/30
214/214 — 95s 443ms/step - accuracy: 0.9761 - loss: 0.0856 - val_accuracy: 0.9395 - val_loss: 0.1910
Epoch 23: early stopping

```

Figure 1 – Phase d'entraînement

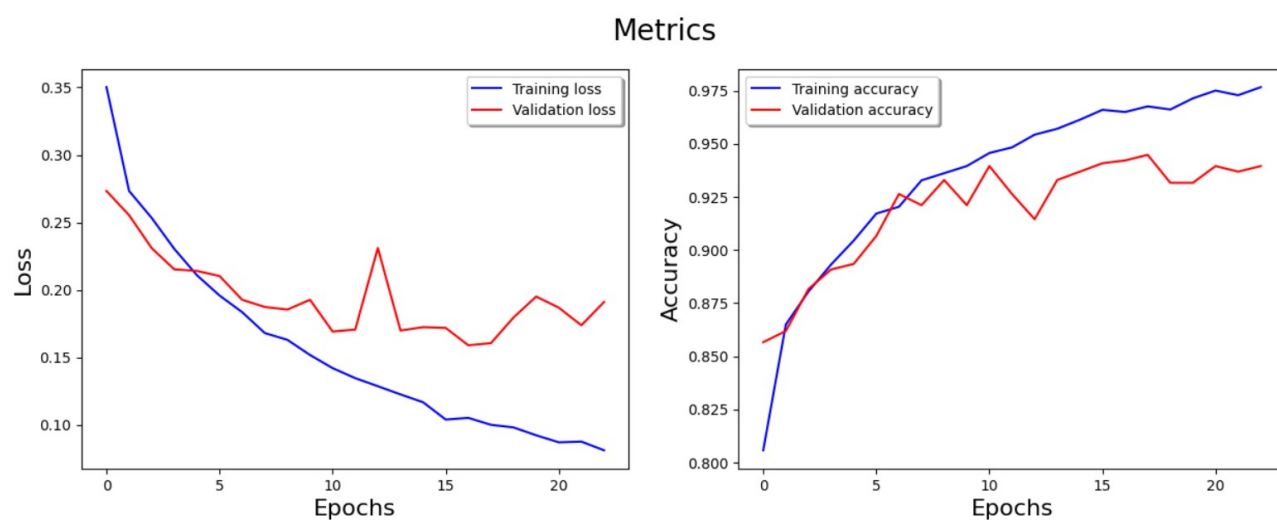


Figure 2 – Graphe des Métriques

### 3. Conclusion

En résumé, bien que notre modèle de CNN ait atteint une précision de 94 % sur la base de données utilisée, les résultats ne sont pas encore concluants. Malgré nos efforts et plusieurs essais, la détection de la falsification d'images reste un défi complexe.