Étude des réseaux de neurones sur la stéganalyse

Marc CHAUMONT[1,2,3], Dino IENCO[1,4], Jérôme PASQUET[1,3], Lionel PIBRE[1,3]

[1]Université de Montpellier

[2]Université de Nîmes

[3]CNRS

[4]IRSTEA

19 Mai 2016

CORESA 2016 - Nancy

Introduction

Un réseau de neurones ? Pourquoi faire ?

Des résultats impressionnants !

	RM+EC	CNN	FNN
Max	24.93%	7.94%	8.92%
Min	24.21%	7.01%	8.44%
Variance	0.14	0.12	0.16
Average	24.67%	7.4%	8.66%

TABLE: Résultats de stéganalyse (P_E) avec S-UNIWARD, 0.4 bpp, scénario clairvoyant pour RM+EC, CNN et FNN.

L'insertion a été faite dans un scénario artificiel !

Sommaire

Un exemple de réseau

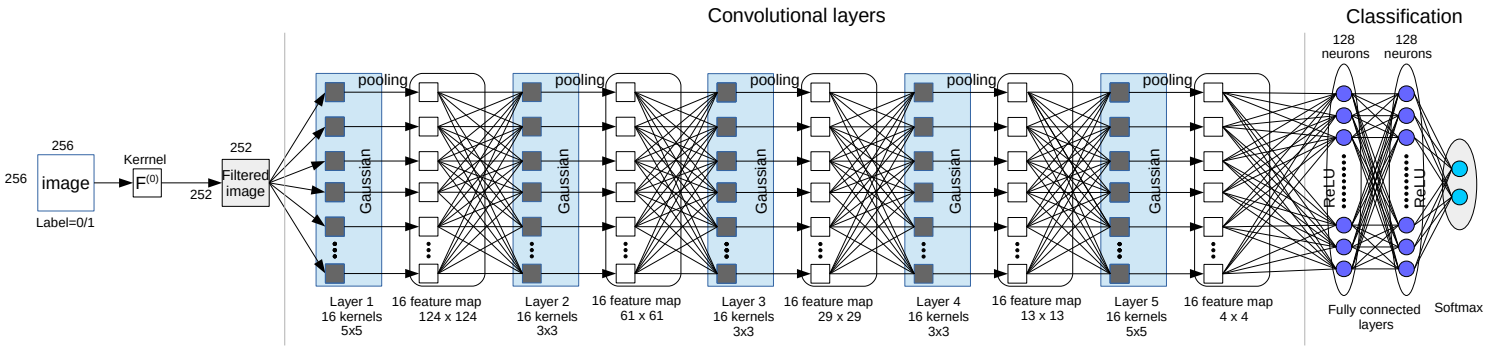


FIGURE: **Schéma du réseau de Qian et al.**

- Inspiré du réseau de Krizhevsky *et al.* 2012
- Moins bon de 3% à 4% par rapport à RM+EC

A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems 25, NIPS*, 2012.

Y. Qian, J. Dong, W. Wang, and T. Tan, "Deep Learning for Steganalysis via Convolutional Neural Networks," in *Proceedings of SPIE Media Watermarking, Security, and Forensics*, 2015.

Le pré-traitement spécifique à la stéganalyse

$$F^{(0)} = \frac{1}{12}\begin{pmatrix} -1 & 2 & -2 & 2 & -1 \\ 2 & -6 & 8 & -6 & 2 \\ -2 & 8 & -12 & 8 & -2 \\ 2 & -6 & 8 & -6 & 2 \\ -1 & 2 & -2 & 2 & -1 \end{pmatrix}$$

Appliquer ce noyau sur les images d'entrée permet au réseau de converger plus rapidement.

Les réseaux convolutionnels

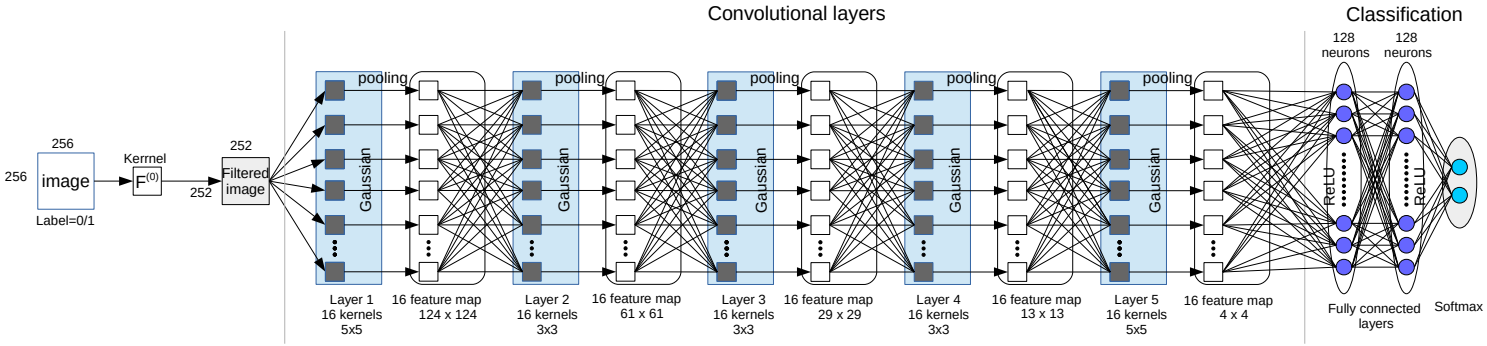


FIGURE: **Schéma du réseau de Qian et al.**

Une couche est composée de plusieurs éléments :

- La convolution,
- L'application d'une fonction d'activation,
- Le *pooling* (sous-échantillonnage),
- La normalisation.

La convolution

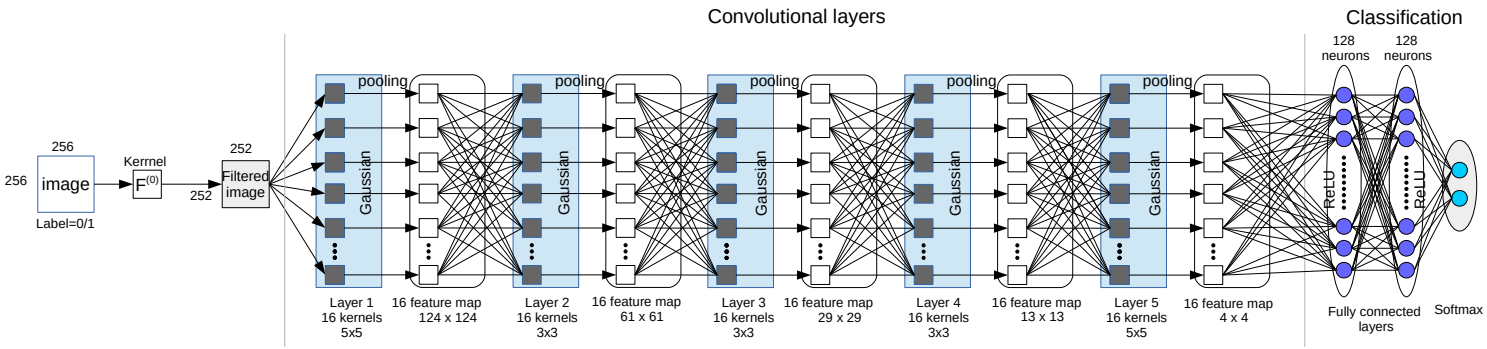


FIGURE: **Schéma du réseau de Qian et al.**

La première couche :

$$\tilde{I}_k^{(1)} = I^{(0)} \star F_k^{(1)} \tag{1}$$

Les autres couches :

$$\tilde{I}_k^{(l)} = \sum_{i=1}^{i=K^{(l-1)}} I_i^{(l-1)} \star F_{k,i}^{(l)} \tag{2}$$

Les fonctions d'activation

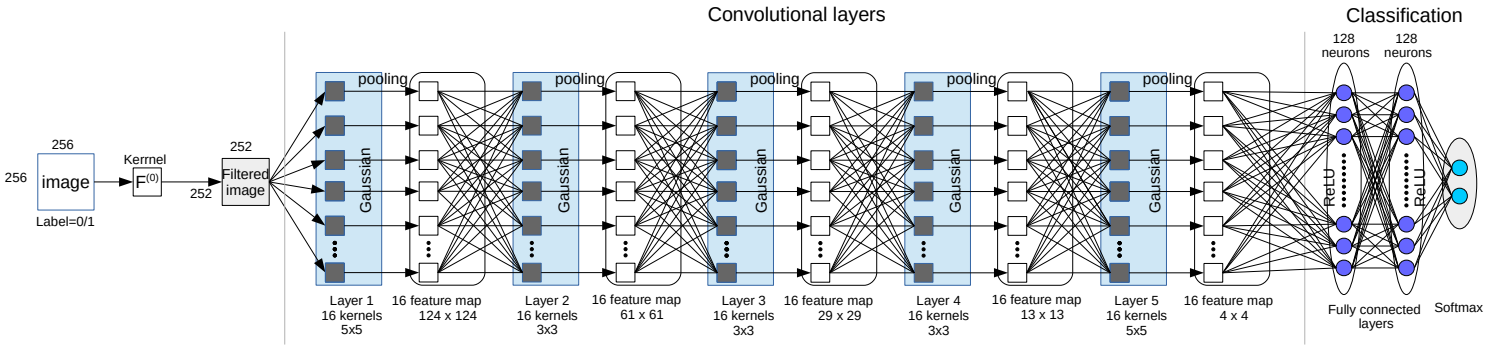


FIGURE: **Schéma du réseau de Qian et al.**

Il existe de nombreuses fonctions d'activation :

- Valeur absolue : $f(x) = |x|$,
- Tangente hyperbolique : $f(x) = tanh(x)$,
- ReLU (*Rectified Linear Unit*) : $f(x) = max(0, x)$,
- Gaussienne : $f(x) = \frac{e^{x^2}}{\sigma^2}$.

Le *pooling*

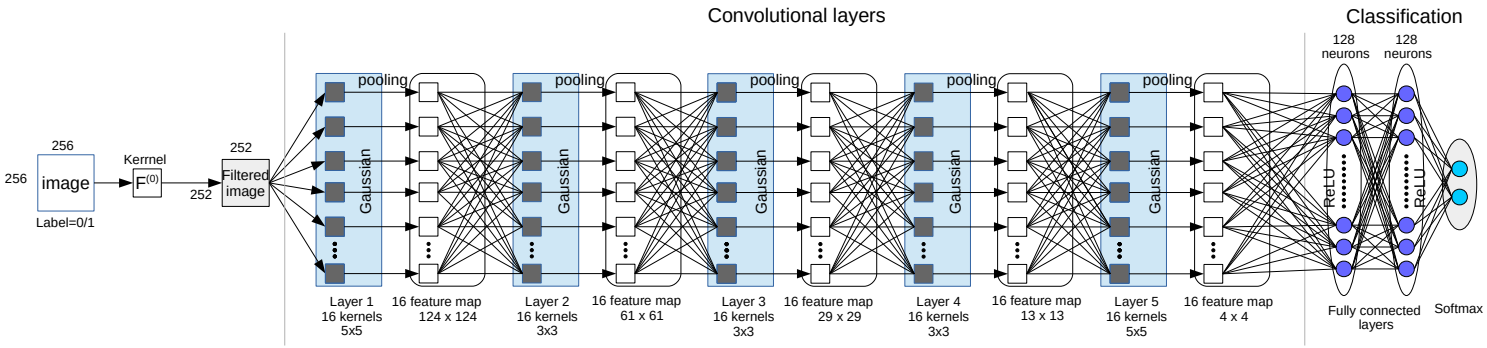


FIGURE: **Schéma du réseau de Qian et al.**

Opération locale calculée sur un voisinage :

- Moyenne locale (conserve le signal),
- Maximum local (invariance à la translation),
- Sous-échantillonnage.

La normalisation

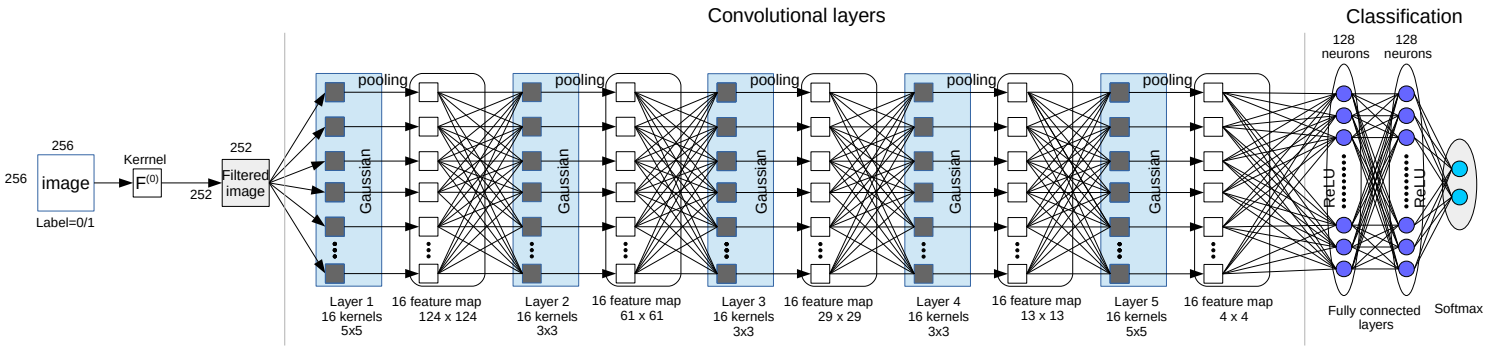


FIGURE: **Schéma du réseau de Qian et al.**

Lorsque l'on normalise entre les *feature maps* :

$$norm(u_g^{x,y}) = \frac{u^{x,y}}{\left(1 + \frac{\alpha}{N} \sum_{g'=max(0,g-\lfloor N/2 \rfloor)}^{min(F,g-\lfloor N/2 \rfloor+N)} (u_{g'}^{x,y})^2\right)^{\beta}} \quad (3)$$

Les couches *fully connected*

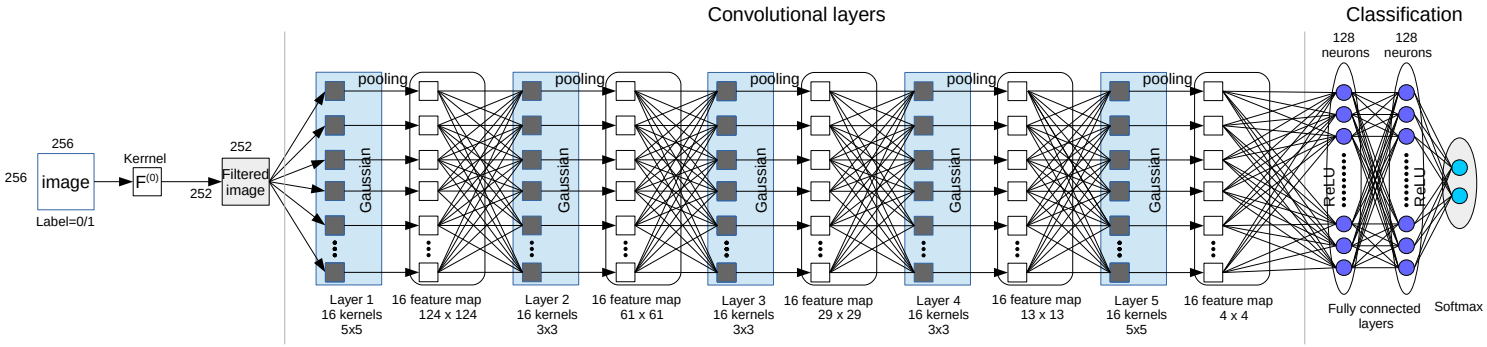


FIGURE: **Schéma du réseau de Qian et al.**

La partie *fully connected* est composée de plusieurs éléments :

- Deux couches entièrement connectées,
- Une fonction appelée *Softmax* qui normalise les valeurs entre [0,1],
- Le réseau donne une valeur *cover* (resp. *stégo*).

Notre réseau

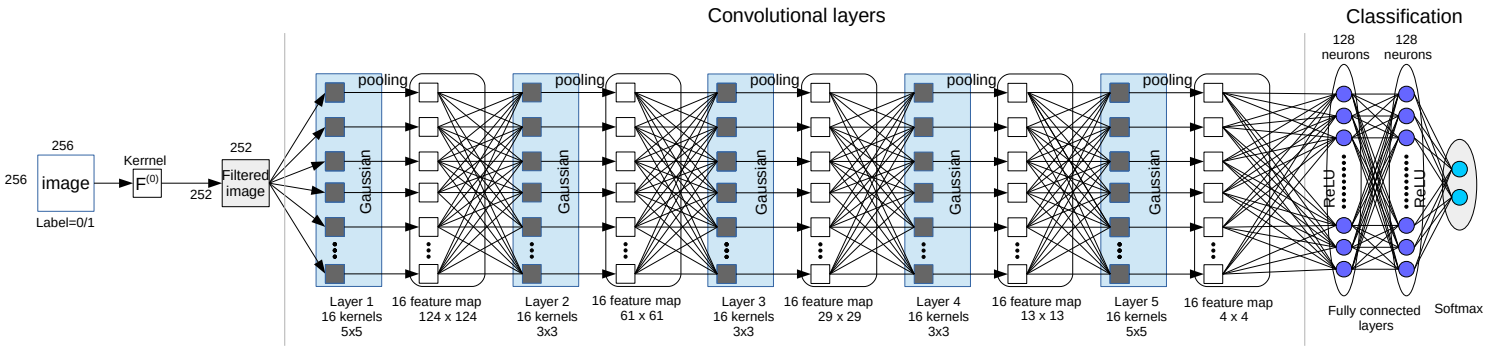


FIGURE: **Schéma du réseau de Qian et al.**

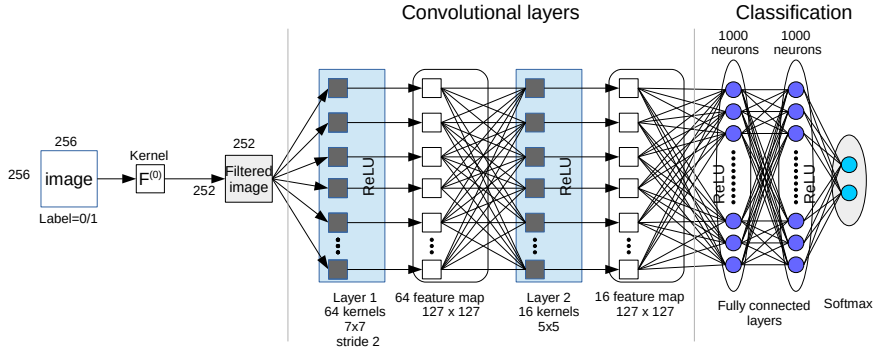
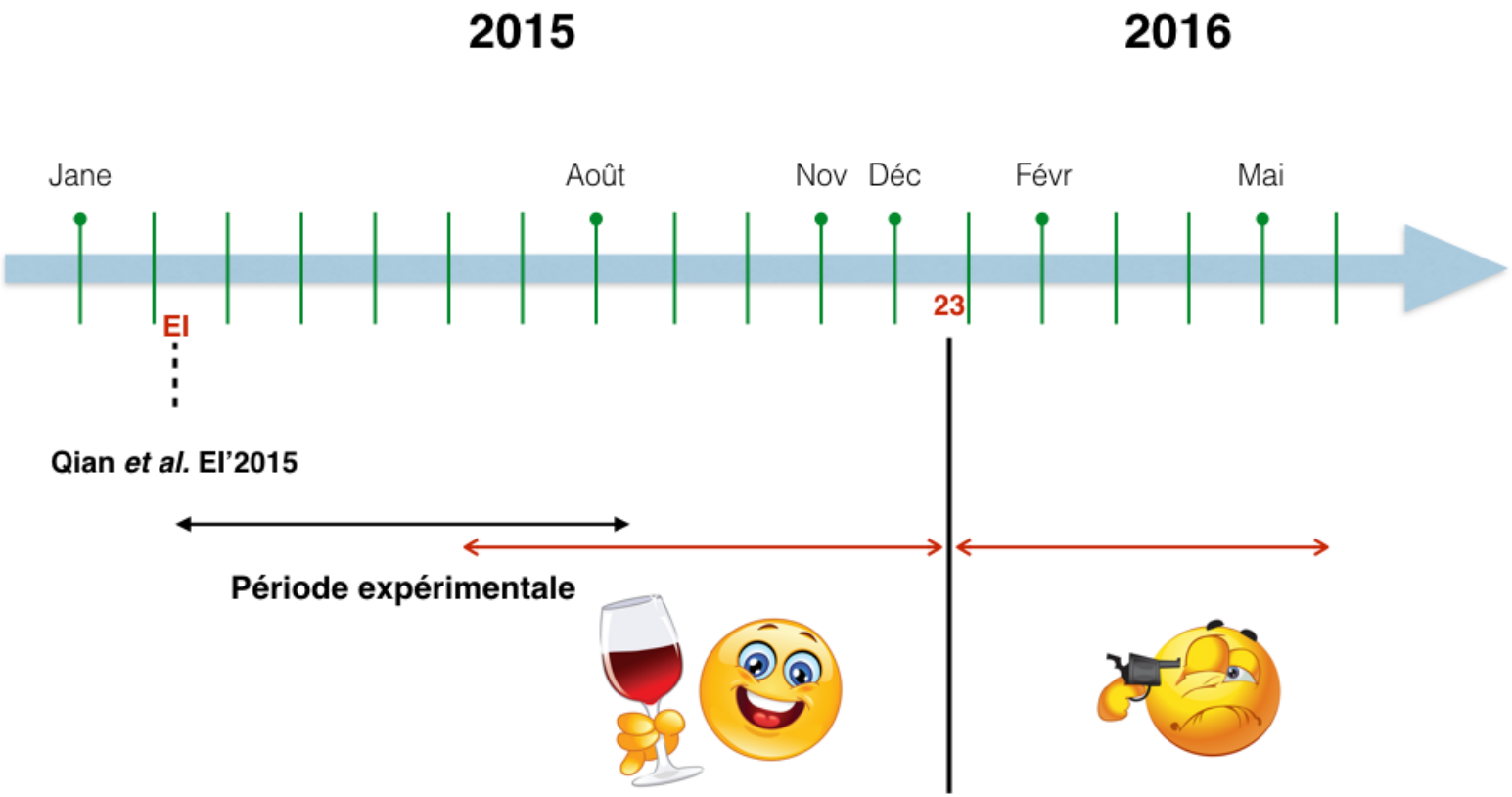


FIGURE: **Schéma de notre réseau.**

Sommaire

Toute une histoire



Sommaire

Expérimentations

- 40 000 images *cover* de taille 256×256 provenant de BossBase,
- S-UNIWARD avec 0.4 bits par pixel,
- Utilisation du simulateur avec la même clé d'insertion,
- Apprentissage sur 60 000 images.

Pourquoi utiliser la même clé ?

Ce n'était pas volontaire...

- Il y a une erreur de documentation dans la version C++ de S-UNIWARD,
- Qian *et al.* ont fait la même erreur.

Utilisation d'une seule clé d'insertion

Lorsque l'on utilise qu'une seule clé d'insertion...

- La clé d'insertion initialise le générateur de la séquence de nombres pseudo-aléatoires
  $\Rightarrow$ la séquence de nombres pseudo-aléatoires $\in [0,1]^n$ **est toujours la même !**
- L'insertion va se faire toujours au même endroit et **avec la même polarité (-1 ou +1)**.

Impact sur l'insertion

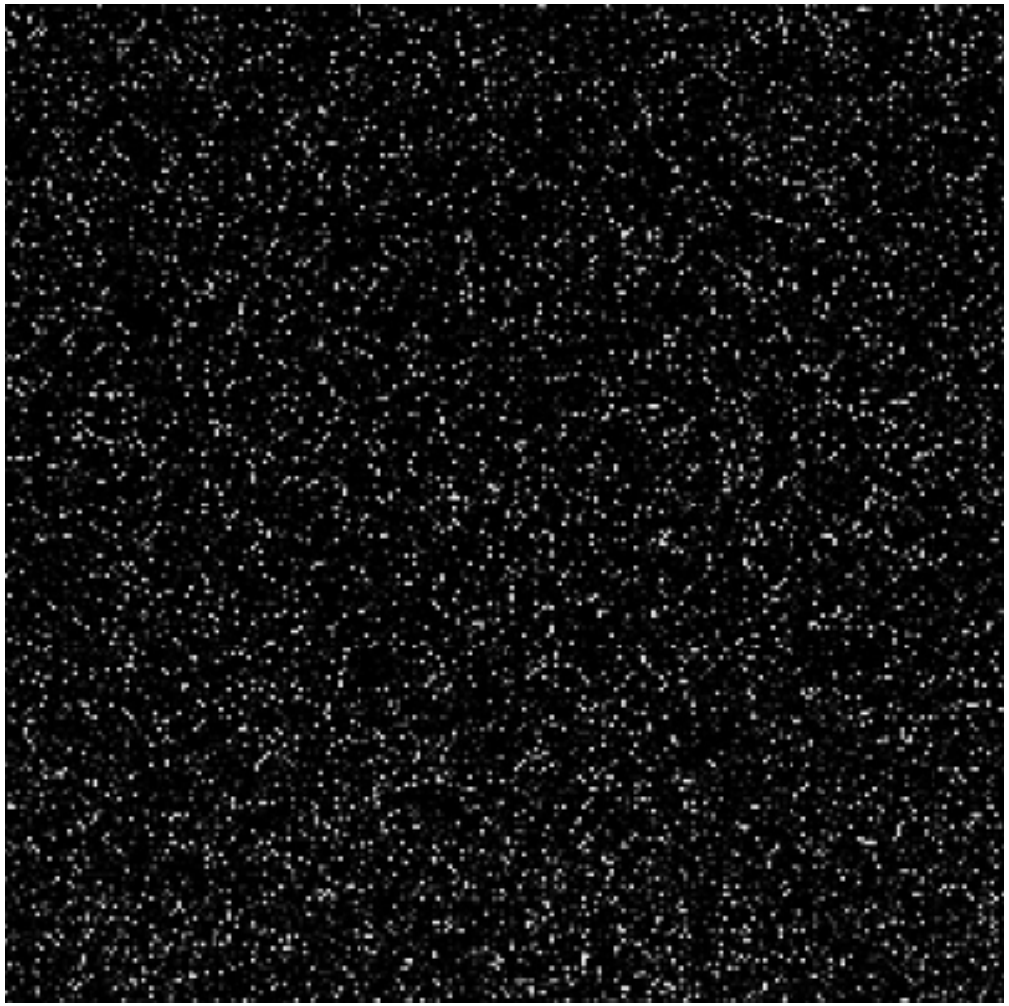


FIGURE: Probabilité de modification. En blanc les zones avec les plus fortes probabilités et en noir celles avec les probablités les plus faibles.

Résultats

- 40 000 images *cover* de taille 256×256 provenant de BossBase,
- S-UNIWARD avec 0.4 bpp,
- Simulateur avec la même clé d'insertion,
- Apprentissage sur 60 000 images.

	RM+EC	CNN	FNN
Max	24.93%	7.94%	8.92%
Min	24.21%	7.01%	8.44%
Variance	0.14	0.12	0.16
Average	24.67%	7.4%	8.66%

TABLE: Résultats de stéganalyse (P_E) avec S-UNIWARD, 0.4 bpp, scénario clairvoyant pour RM+EC, CNN et FNN.

En utilisant des clés d'insertion différentes, le même réseau avec plus de neurones obtient 38.1% de probabilité d'erreur !

Sommaire

Conclusion de cette histoire

- Faire attention à l'implémentation et à la documentation des logiciels (peu importe leur provenance),
- Toujours utiliser différentes clés d'insertion !
- Le Rich Models n'est pas efficace pour détecter un phénomène spatial.

Les CNNs ne sont pas morts... il y a toujours de l'espoir !

A. Xu, H. Z. Wu, and Y. Q. Shi, "Structural Design of Convolutional Neural Networks for Steganalysis," in *IEEE Signal Processing Letters*, vol. 23, no. 5, pp. 708-712, May 2016.

Fin

Merci de votre attention !

Cover-Source Mismatch

Pourquoi avoir fait des tests dans le scénario avec *Cover-Source Mismatch* ?

Nous n'avions pas encore connaissance de cette erreur lors des tests !