ToxicIA : Apprentissage Profond Appliqué à l'Analyse des Signaux Parasites Compromettants

Florian Lemarchand, Cyril Marlin, Florent Montreuil, Erwan Nogues, Maxime Pelcat

ToxicIA : Apprentissage Profond Appliqué à l'Analyse des Signaux Parasites Compromettants

Florian LEMARCHAND[1], Cyril MARLIN[2], Florent MONTREUIL[2], Erwan NOGUES[1,2], and Maxime PELCAT[3]

[1] Univ. Rennes, INSA Rennes, CNRS, IETR - UMR 6164
[2] DGA-MI
[3] Institut Pascal, Clermont-Ferrand, UMR CNRS 6602
Contact: florian.lemarchand@insa-rennes.fr

Abstract. Cet article propose une attaque par canal auxiliaire électromagnétique capable de reconstruire automatiquement un signal de type image. Le système proposé permet l'extraction de données compromettantes embarquées dans le signal intercepté. Basé sur l'apprentissage profond, le système est capable d'extraire sur le jeu de test proposé plus de 57% de l'information présente dans le signal intercepté, et ce pour différents type de signal vidéo qu'ils soient analogique ou numérique. Une extension du système est proposée ayant pour but l'audit d'un système d'information grâce à un mécanisme automatique d'alarme en cas de compromission du système d'information visé. Basée sur une architecture hétérogène radio-logicielle et processeur graphique, la solution est déployable facilement dans un système d'information existant manipulant de l'information devant rester secrète.

Keywords: Détection de SPC · Apprentissage · Débruitage

1 Introduction

Tout appareil électronique produit des émanations Électro-Magnétique (EM) qui non seulement interfèrent avec les appareils radio mais aussi compromettent les données traitées par l'appareil dans le cas d'un Système d'Information (SI). Une tierce partie peut alors effectuer une analyse des canaux auxilaires et extraire tout ou partie des informations d'origine, compromettant ainsi la confidentialité du SI. Alors que les travaux pionniers du domaine se concentraient sur les signaux analogiques [1], des études récentes étendent l'exploit d'écoute par canal auxilaire EM aux signaux numériques et circuits intégrés [2]. Le profil de l'attaquant prend également une nouvelle dimension avec les performances accrues des radio-logicielles (plus communément appelées Software-Defined Radio (SDR) en anglais). Avec les récentes avancées en matière d'équipement radio, un attaquant peut s'appuyer sur un traitement du signal avancé pour repousser encore plus loin les limites de l'attaque par canal auxiliaire utilisant des émanations EM [3].

Dans le même temps, les progrès de l'apprentissage machine ont ouvert la voie à l'interprétation automatisée des données interceptées. Utilisant de puissants processeurs graphiques (Graphics Processing Unit (GPU) en anglais) et des réseaux neuronaux profonds, un attaquant peut extraire des modèles ou même le contenu structuré complet des données interceptées et ce avec un haut degré de confiance et un temps d'exécution limité.

Dans cet article, nous proposons une méthode basée sur l'apprentissage profond pour débruiter et interpréter un signal intercepté. Nous démontrons un système de bout en bout allant de l'interception à l'interprétation, intégrant une SDR et un processus d'apprentissage profond qui respectivement détecte et extrait l'information fuitant à une distance de plus de 10 mètres. Plusieurs configurations de systèmes sont expérimentées et évaluées. À partir des résultats, nous proposons des solution de détection pour ce type d'attaque en intégrant une reconnaissance de caractère (Character Recognition (CR) en anglais) déclenchant automatiquement une alarme si la confidentialité du système est compromise.

L'article est organisé comme suit. La Section 2 présente les méthodes existantes pour extraire de l'information des émanations EM. La Section 3 décrit la méthode proposée pour l'extraction automatique de caractères. Les résultats expérimentaux et les performances détaillées sont exposés dans la Section 4. Une extension de la méthode à un système de protection de la confidentialité est proposée dans la Section 5. Enfin, la Section 6 conclut le document.

2 État de l'art

Van Eck *et al.* [1] ont publié les premiers rapports techniques révélant comment des émissions involontaires provenant de dispositifs électroniques peuvent être exploitées et compromettre les données. Alors que les premiers travaux du domaine ciblaient les écrans cathodiques et les signaux analogiques, Kuhn *et al.* [2] proposent d'utiliser des attaques par canal auxiliaire pour extraire des données confidentielles des écrans Liquid Crystal Display (LCD), visant ainsi des données numériques. Par la suite, d'autres types de systèmes ont été attaqués. Vuagnoux *et al.* [4] étendent le principe de l'attaque par canal auxiliaire EM pour capturer les données des claviers et, dans leurs travaux récents, Hayashi *et al.* présentent des méthodes d'interception basées SDR ciblant les ordinateurs portables, les tablettes [5] et les smartphones [6]. Ricordel *et al.* [7] exposent eux l'interception de signal vidéo émanant de câbles Digital Visual Interface (DVI) et High-Definition Multimedia Interface (HDMI). L'utilisation de SDR augmente la surface d'attaque des organisations étatiques aux pirates informatiques. Elle ouvre également de nouvelles possibilités de post-traitement qui améliorent les caractéristiques d'attaque. De Meulemeester *et al.* [8] s'appuient sur la SDR pour améliorer les performances de l'attaque et trouver automatiquement la structure des données acquises. En récupérant les paramètres de synchronisation du SI ciblé, le signal EM capturé peut être transformé d'un vecteur en une image matricielle, reconstruisant les informations visuelles sensibles en deux dimensions. Cette étape est appelée *rasterisation*.

Lors de l'extraction d'information visuelle à partir d'un signal EM (interception), une partie non négligeable de l'information originale est perdue. Cette perte entraîne une baisse du rapport signal sur bruit (Signal to Noise Ratio (SNR) en anglais) et une détérioration de la cohérence spatiale dans les échantillons reconstruits dans le cas de données image. Des méthodes de débruitage sont donc nécessaires. Le débruitage d'image par les techniques de traitement du signal a été largement étudié car c'est une étape indispensable dans de nombreuses applications de vision par ordinateur. Block-Matching 3D (BM3D) est une méthode bien connue d'élimination du bruit blanc Gaussien et a été proposée par Dabov *et al.* [9]. BM3D utilise le seuillage et le filtrage de Wiener dans le domaine de la transformée. Il sera utilisé comme référence dans les expérimentations de la Section 4.

Les algorithmes d'apprentissage profond se sont récemment démarqués en résolvant de nombreux problèmes de traitement du signal. Ces modèles entraînés ont une capacité extrême à s'adapter à des problèmes complexes. Les architectures récentes de GPU ont été optimisées pour prendre en charge les calculs complexe d'apprentissage profond et ont favorisé des réseaux toujours plus profonds, extrayant des informations structurées des données et fournissant ainsi des résultats là où les algorithmes classiques échouent. Denoising Convolutionnal Neural Network (DnCNN) [10] est un réseau neuronal convolutif (Convolutional Neural Network (CNN) en anglais) développé pour restorer la qualité d'image. DnCNN-B est implémenté pour éliminer le bruit Gaussien, sans connaissance préalable du niveau de bruit. D'autres techniques telles que les autoencodeurs de débruitage [11, 12] sont capables de débruiter des images sans restriction sur le type de bruit. Les autoencodeurs apprennent à associer leur entrée dans un espace latent (codage) et à projeter à l'inverse la représentation latente dans l'espace d'entrée (décodage). Les autoencodeurs apprennent un modèle de débruitage en minimisant une fonction de perte qui évalue la différence entre la sortie de l'autoencodeur et la référence. Des méthodes avancées, telles que Noise2Noise [13], déduisent des stratégies de débruitage sans aucune données de référence propre. L'algorithme Noise2Noise apprend une représentation du bruit en ne regardant que des échantillons bruités.

Dans le problème considéré, certaines composantes du bruit sont distribuées de façon non aléatoire et ont une cohérence spatiale. De plus, l'information est endommagée (partiellement perdue et répartie sur plusieurs pixels) par le processus d'interception/rasterisation. Aucune des méthodes précédemment exposées n'est adaptée à de telles natures de bruit et de distorsion des données interceptées, d'où le besoin d'un nouveau dispositif.

Il existe des approches conventionnelles pour protéger les dispositifs contre l'écoute clandestine. Ces approches apparaissent sous différents noms de code tels que TEMPEST [14] ou Emission Security (EMSEC). Elles consistent en des dispositifs de blindage [2] pour annuler les émanations, ou en l'utilisation de polices qui minimisent les émanations EM [15]. Cependant, ces approches sont soit des solutions coûteuses, soit techniquement difficiles à utiliser dans la

pratique, surtout lorsqu'il s'agit de garantir la confidentialité des données tout au long du cycle de vie d'un SI complexe.

3 Attaque par Canal Auxiliaire Proposée

3.1 Description du Système

La Figure 1 montre la solution bout en bout proposée. La méthode proposée reconstruit automatiquement les fuites d'informations visuelles dues à des émanations compromettantes. L'installation est composée de deux blocs principaux. Pour commencer, l'antenne et le traitement SDR capturent dans le domaine Radio Fréquence (RF) les informations affichées qui ont fuité. Ensuite, le signal démodulé est traité par l'ordinateur hôte, qui extrait une version bruitée de l'image originale [2]. D'autres techniques de traitement d'image, exclues du champ d'application du présent document, sont ajoutées pour améliorer le SNR et améliorer la qualité de l'information extraite. La méthode, basée sur l'apprentissage profond, comprend les premières étapes suivantes : débruitage, segmentation, détection/localisation des caractères et reconnaissance des caractères. Enfin, une transformée de Hough est appliquée pour la détection des lignes de texte et un algorithme Bitap [16] est appliqué pour approximer les informations de correspondance. L'installation complète détecte des émanations vidéo compromettantes (analogique ou numérique) et déclenche automatiquement une alarme si des informations critiques sont contenues dans ces émanations. Les sections suivantes décrivent en détails comment la méthode est entraînée et intégrée.

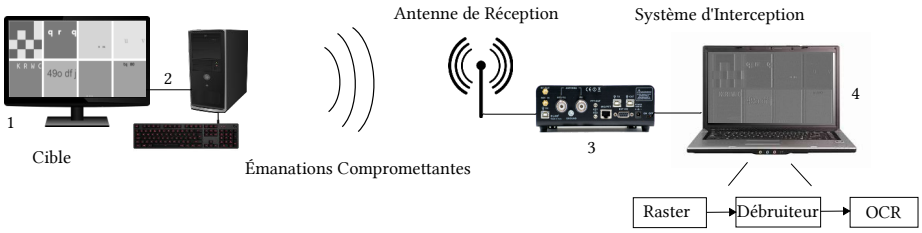


Fig. 1. La configuration expérimentale comprend : le système cible attaqué utilisant un écran (1), affichant des informations sensibles, connecté à un système d'information (2) ; la chaîne d'interception comprenant un récepteur SDR (3) envoyant des échantillons à un ordinateur hôte (4) qui met en œuvre un traitement de signal comprenant un débruiteur par apprentissage profond et une étape de CR.

3.2 Construction du Jeu de Données d'Entraînement

Un effort substantiel a été fourni pour construire un processus qui génère et labellise de façon semi-automatique les jeux de données pour l'apprentissage

supervisé. Chaque échantillon d'image est constitué d'un fond uniforme sur lequel sont imprimés des caractères divers. À l'aide de ce processus, un corpus de 123 610 images de taille 256×256 pixels, propres au problème en question, a été créé pour servir d'échantillons d'images étiquetés. Ils serviront de jeux de données d'apprentissage, de validation et de test.

La configuration proposée débruite les échantillons interceptés, puis en extrait le contenu, c'est-à-dire les caractères détectés et leurs positions. L'espace d'entrée à couvrir par l'ensemble de données d'apprentissage est très vaste et trois principaux types de variabilité d'interception peuvent être observés. Premièrement, l'interception induit une perte importante de l'information existant à l'origine dans les données interceptées. Cette perte elle-même provoque des incohérences dans la rasterisation des images et ces deux phénomènes combinés créent un bruit fort et difficilement prévisible. Deuxièmement, les émanations EM peuvent provenir de sources différentes, utilisant des technologies différentes, ce qui implique différents échantillons interceptés pour la même image de référence. Le jeu de données et l'attaque proposée couvrent les câbles et connecteurs Video Graphics Array (VGA), Display Port (DP) vers DVI et HDMI. En plus de cette variabilité inhérente à l'interception, un troisième type de variabilité synthétique est introduit pour résoudre de manière robuste le problème de l'extraction des caractères avec un modèle entraîné. Dans cet effort, de nombreux caractères différents ont été introduits dans le corpus, pour être affichés sur l'écran de la cible attaquée. De taille 11 à 70 points, les caractères sont à la fois des chiffres et des lettres, les lettres sont en majuscules et en minuscules. Des polices de caractère variées, des couleurs de caractère et des couleurs d'arrière-plan, ainsi que des positions de caractère variées dans l'échantillon ont été utilisées. Compte tenu de ces différentes sources de variabilité, l'ensemble de données a été construit en essayant d'obtenir une équi-représentation des différentes conditions d'interception.

Le choix a été fait d'afficher sur l'écran cible un échantillon contenant plusieurs patchs de 256×256 pixels (comme sur l'image en haut à gauche de la Figure 2). Pour la construction du corpus de données, le fait d'avoir plusieurs patchs permet d'introduire plus de variabilité dans le corpus à nombre d'interceptions équivalent. Le principal défi lors de la création de l'ensemble de données réside dans l'acquisition des échantillons eux-mêmes. En effet, une fois interceptés, les échantillons ne sont pas directement utilisables. Le processus d'interception produit des échantillons tels que celui de la Figure 2 (milieu-haut) où les caractères interceptés ne sont pas alignés (dans le temps et dans l'espace) avec les échantillons de référence respectifs. Nous introduisons une méthode automatisée qui utilise les marges, artificiellement colorées en rouge dans la Figure 2 (milieu-haut), pour aligner les échantillons dans l'espace. Les marges sont détectées à l'aide d'une recherche brute-force de grands gradients horizontaux et verticaux (pour trouver les marges verticales et horizontales, respectivement). Une étape de validation est ajoutée pour assurer l'alignement temporel, basée sur l'insertion d'un QRCode dans le patch en haut à gauche. Si le QRCode est similaire entre

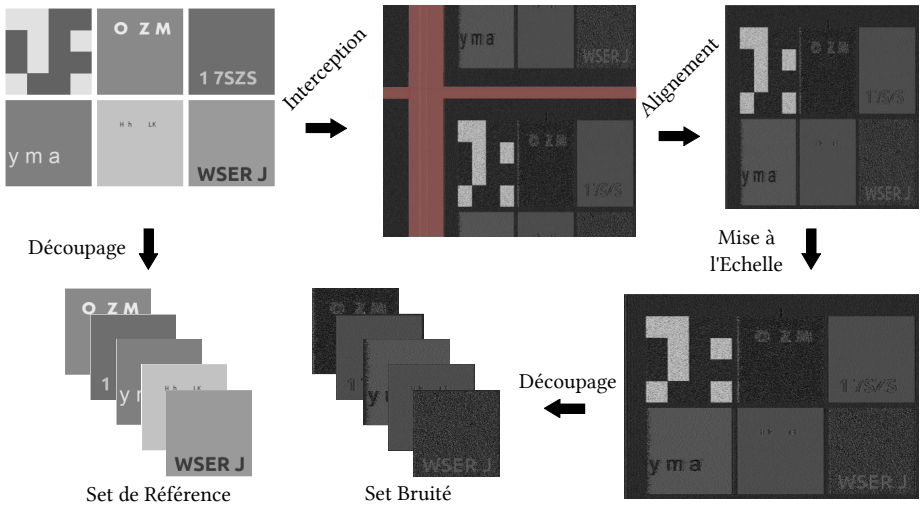


Fig. 2. Une image de référence est affichée sur l'écran ciblé (en haut à gauche). Le module d'interception fournit des échantillons non calibrés (en haut au milieu). Des marges verticales et horizontales (en rouge) permettent l'alignement et le retrait des marges (en haut à droite). Les échantillons sont ensuite mis à l'échelle afin d'avoir la même taille que la référence. Enfin, une division en patch est réalisée pour obtenir la même disposition que le set de référence.

l'image de référence et l'image interceptée, les patchs d'image sont introduits dans le jeu de données.

L'augmentation de données [17] est utilisée pour améliorer encore l'espace de variabilité couvert par le corpus. Elle se fait ici sur les patchs. Des méthodes conventionnelles sont appliquées aux échantillons bruts pour les transformer linéairement (flou Gaussien et médian, bruit sel et poivre, inversion de couleur et normalisation du contraste).

3.3 Solution Implémentée pour Intercepter les Données Compromettantes

Afin d'automatiser l'interception de données compromettantes, le réseau Mask R-CNN [18] a été transformé en débruiteur et classifieur. D'autres méthodes plus simples d'apprentissage profond et de traitement du signal, examinées dans la Section 4.2, sont également mises en œuvre pour évaluer la qualité de la proposition. Mask R-CNN est une solution adaptée de Faster R-CNN [19]. Le réseau se compose de trois étapes. La première étape est un réseau convolutif *ResNet101* [20] qui extrait des caractéristiques des échantillons d'entrée. La deuxième étape est un réseau de proposition de régions d'intérêt (Region Proposal Network (RPN) en anglais) qui propose des régions d'intérêt (Region of Interest (ROI) en anglais) en se basant sur les caractéristiques précédemment extraites. Les ROI sont des régions de l'image où l'information mérite une plus

grande attention. La troisième étape classifie le contenu de chaque ROI et renvoie les coordonnées des cadres de délimitation (bounding box en anglais) de chacune des ROI. La principale différence entre Faster R-CNN et Mask R-CNN réside dans une branche complètement convolutive (Fully Convolutionnal Network (FCN) en anglais) supplémentaire [21] fonctionnant en parallèle de la classification et extrayant un masque binaire pour chaque ROI, fournissant ainsi une localisation plus précise de l'objet d'intérêt.

Mask R-CNN n'a pas été conçu à l'origine pour être utilisé comme un débruiteur, mais plutôt pour réaliser de la segmentation d'instance. Cependant, il correspond bien au problème ciblé. En effet, le problème est similaire à une segmentation où le signal doit être séparé du bruit. Par conséquent, lorsqu'on alimente correctement un réseau Mask R-CNN entraîné avec des échantillons bruités contenant des caractères, on obtient des listes de labels (c'est-à-dire une reconnaissance de caractères), ainsi que leurs *bounding boxes* (localisation des caractères) et des masques binaires représentant le contenu de l'échantillon original *propre*.

Deux stratégies peuvent être utilisées pour l'application de Mask R-CNN au problème de ce document. La première idée est d'appliquer un moteur de reconnaissance optique de caractère (Optical Character Recognition (OCR) en anglais) à la sortie de la segmentation de Mask R-CNN (côté gauche de la Figure 3) afin d'extraire les caractères des masques. Une deuxième possibilité est d'utiliser la faculté de classification de Mask R-CNN (Figure 3 à droite) et ainsi obtenir une liste de labels sans utiliser de moteur OCR. Dans la pratique, la seconde méthode utilisant le classifieur de Mask R-CNN donne de meilleurs résultats, comme exposé dans la Section 4.2.

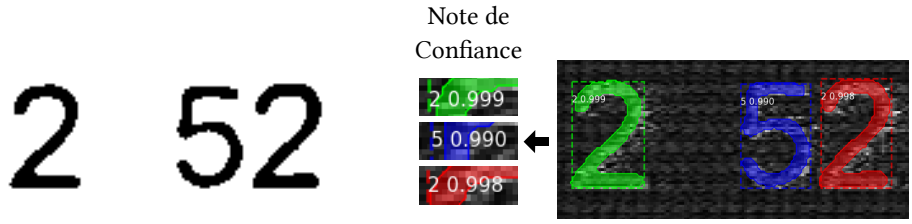


Fig. 3. La sortie de Mask R-CNN peut être utilisée de deux façons. Les masques de segmentation peuvent être imprimés (à gauche) et traités par un OCR, ou le classificateur de Mask R-CNN peut directement inférer le contenu de l'échantillon (à droite) et afficher des informations de confiance.

Outre le réseau ResNet101, Mask R-CNN est composé de plusieurs couches complètement connectées (Fully Connected (FC) en anglais) pour la classification. La stratégie d'entraînement consiste en une initialisation utilisant des poids pré-entraînés [22] pour le jeu de données MS COCO [23]. Le réseau est ensuite ré-entraîné pour correspondre parfaitement à l'application. D'abord les couches

FC sont ré-entrainées, puis le ResNet101, couche par couche, de la couche de sortie vers celle d'entrée. Ceci est réalisé pour assurer la convergence et accélérer l'apprentissage.

4 Résultats Expérimentaux

4.1 Configuration Expérimentale

Le protocole expérimental est le suivant. L'écran intercepté est à 10m de l'antenne d'interception. Les fuites sont émises soit par un câble VGA, un câble DP-vers-DVI ou un connecteur HDMI. Le système d'interception est illustré par la Figure 1. L'antenne est une antenne large bande bilog, la SDR retrouvant automatiquement les paramétres [8] est une Ettus X310 recevant avec 100 MHz de bande passante. L'ordinateur hôte qui effectue le post-calcul a un système d'exploitation Linux, une unité centrale de traitement (Central Processing Unit (CPU) en anglais) Intel Xeon W-2125, et un GPU Nvidia GTX 1080 Ti. L'ordinateur hôte rasterise les données compromettantes en utilisant principalement le CPU alors que l'OCR/débruiteur proposé s'exécute sur le CPU et le GPU.

4.2 Comparaison des Performances d'Interception

La méthode présentée a pour objectif d'analyser des émanations compromettantes. Une fois qu'un signal est détecté et rasterisé, les émanations interceptées doivent être classées comme compromettantes ou non. La Figure 4 illustre subjectivement la différence de performance des différents débruiteurs implémentés. Nous proposons d'évaluer la fuite de données en fonction de la capacité d'un modèle à extraire l'information originale. Nous utilisons comme mesure le rapport entre le nombre de caractères que notre méthode classifie correctement à partir d'un échantillon intercepté, et le nombre réel de caractères dans la référence *propre* correspondante.

Tout d'abord, un échantillon contenant un grand nombre de caractères est généré de façon pseudo-aléatoire (semblable à la construction d'un jeu de données). L'échantillon est alors affiché sur l'écran écouté et les émanations EM sont interceptées. La méthode de débruitage/extraction d'information proposée est appliquée et les résultats obtenus sont comparés à l'échantillon de référence. La méthode utilisant Mask R-CNN produit directement une liste des caractères extraits. D'autres méthodes, mises en œuvre pour évaluer l'efficacité de la méthode proposée, utilisent le débruitage en combinaison avec l' OCR Tesseract [24]. Tesseract est un moteur OCR performant, qui récupère des caractères dans des images. Il produit une liste de caractères extraits d'un échantillon débruité et, comme la sortie de Tesseract est du même type que celle du Mask R-CNN, des métriques peuvent être extraites pour les comparer de manière équitable. Le *F-score* (également connu sous le nom de F_1 *score* ou *F-measure*) sert à mesurer la précision d'un modèle. Le *F-score* est calculé à l'aide de l' Equation (1). Le *F-score* étant une mesure de classification binaire, un vrai positif est

défini ici comme étant la reconnaissance d'un caractère existant réellement dans l'échantillon de référence. De même, un vrai négatif correspond à un élément qui n'a pas été trouvé et n'existant pas dans la référence. Un faux positif est le fait de trouver un caractère non présent dans la référence et un faux négatif représente le fait de ne rien trouver là où un caractère existe.

$$F\text{-}score = 2 \times \frac{precision \times recall}{precision + recall} \tag{1}$$

Dans l'Équation (1), le *F-score* est calculé en utilisant *precision* et *recall*. *precision* est le nombre de vrais positifs divisé par le nombre de tous les positifs. *recall* est le nombre de vrais positifs divisé par le nombre d'échantillons pertinents, l'ensemble des échantillons pertinents étant l'union des vrais positifs et des faux négatifs.

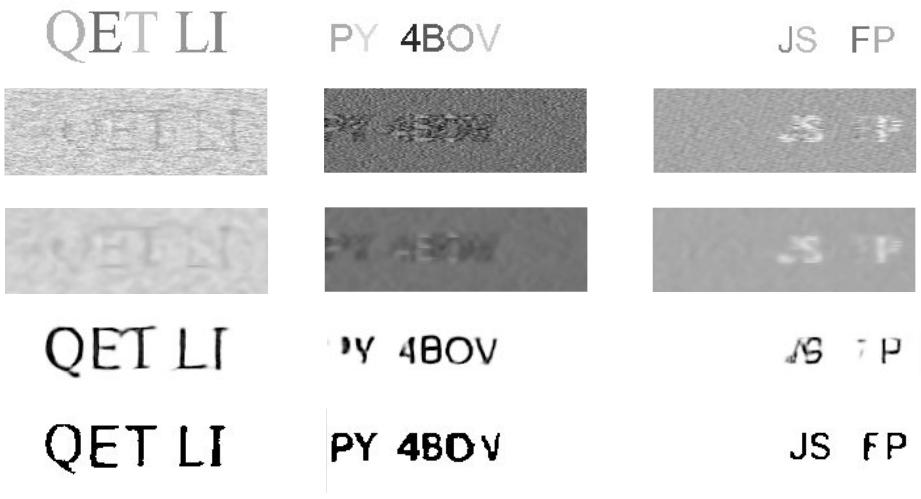


Fig. 4. Trois échantillons (gauche, milieu, droite) affichés à différentes étapes du flot d'interception et de débruitage. De haut en bas: le premier patch est l'échantillon affiché à l'écran (échantillon de référence) ; le second patch est l'échantillon après rasterisation (patch brut) ; les troisième, quatrième et cinquième patchs sont les patchs débruités via BM3D, autoencodeur et Mask R-CNN, respectivement.

Le Tableau 1 présente les résultats de différentes méthodes sur un ensemble de données de test de 12563 patchs. Toutes les méthodes sont testées en utilisant Tesseract, et comparées à la classification de Mask R-CNN utilisé comme OCR. Tesseract est d'abord appliqué sur des échantillons bruts (non débruités) comme point de référence. BM3D est la seule solution de débruitage classique testée. *Noise2Noise*, *AutoEncoder*, *RaGAN*, *UNet* sont différents réseaux d'apprentissage profond configurés comme débruiteurs. Comme le montre le Tableau 1, la

classification de Mask R-CNN surpasse nettement toutes les autres méthodes. La version de Mask R-CNN utilisant son propre classifieur est meilleure que le moteur OCR Tesseract appliqué à la sortie segmentée de Mask R-CNN.

La Figure 5 trace en deux dimensions les scores de *precision* et de *recall* des différentes configurations étudiées. Ces figures nous montrent que, comparé aux autres configurations, Mask R-CNN utilisant son propre classifieur ("MR-CNN Standalone" sur la Figure 5) prend plus de décision (à égalité avec "MR-CNN+Tesseract") et ces décisions sont meilleures.

BM3D a été utilisé ici comme point de référence pour des solutions basées sur l'apprentissage profond puisqu'il utilise des techniques classiques de traitement d'image. Les scores de BM3D montrent qu'il n'est pas efficace pour l'application considérée. Les mauvaises performances de BM3D sont dues à la nature complexe du bruit adressé, qui n'est pas seulement constitué d'un bruit blanc Gaussien.

Débruiteur	OCR	F-Score	precision	recall
Brut	Tesseract	0.04	0.20	0.02
BM3D		0.13	0.22	0.09
Noise2Noise		0.17	0.25	0.12
AutoEncoder		0.24	0.55	0.15
RaGAN		0.24	0.42	0.18
UNet		0.35	0.62	0.25
Mask R-CNN		0.55	**0.82**	0.42
Mask R-CNN	Mask R-CNN	**0.68**	**0.81**	**0.57**

Table 1. Performances de reconnaissance de caractère de plusieurs configurations utilisant le débruitage et Tesseract ou la classification Mask R-CNN. Le classifieur de Mask R-CNN surpasse les autres méthodes avec un *F-score* de 0.68 sur le jeu de test.

Un autre indicateur clé de performance des algorithmes d'apprentissage profond est le temps d'inférence (Tableau 2). L'implémentation proposée utilisant Mask R-CNN infère les résultats d'une entrée de 1200×1900 pixels en 4.04s en moyenne. Ce temps d'inférence, bien qu'inférieur à celui de BM3D, est supérieur à celui des autres réseaux utilisés. Néanmoins, le temps d'inférence de Mask R-CNN inclut tout le processus débruitage/OCR et fournit un score d'extraction largement meilleur. Dans le contexte d'une écoute continue des émanations EM, il fournit un compromis acceptable entre le temps de traitement et la performance de détection.

5 Discussion des Résultats

5.1 Impact de la Proposition sur la Menace

Les protocoles de cryptographie récents sont très difficiles à déchiffrer. Par conséquent, les attaques sont susceptibles de se déplacer là où les informations sont les plus vulnérables, c'est-à-dire avant le chiffrement. Les attaques

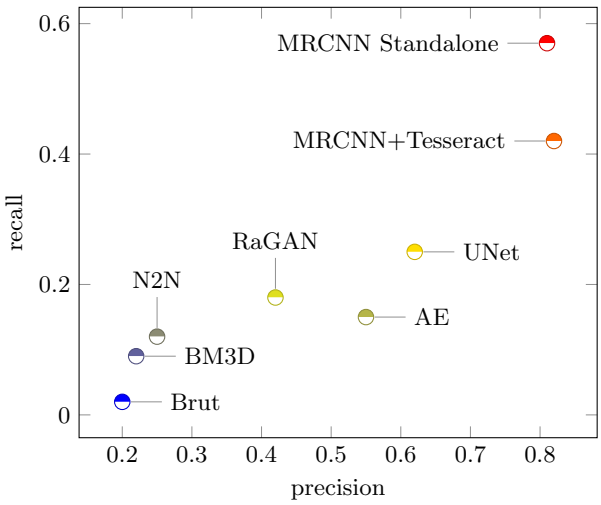


Fig. 5. Score *recall* et score *precision* pour les modèles BM3D, Noise2Noise (N2N), RaGAN, Autoencodeur (AE), UNet, Mask R-CNN (MRCNN) suivi par une reconnaissance Tesseract. Le point "Brut" correspond à l'application de Tesseract sans débruitage. Les scores pour Mask R-CNN utilisant son propre classifieur apparaissent également (MRCNN Standalone).

Denoiser	OCR	Inference Timing (s)
$\emptyset$	Tesseract	0.19
BM3D		21.8
Autoencoder		1.15
Mask R-CNN		4.22
Mask R-CNN	Mask R-CNN	4.04

Table 2. Temps d'inférence pour plusieurs configurations utilisant Tesseract ou le classifieur de Mask R-CNN comme OCR. La taille de l'échantillon d'entrée est de 1200×1900. La première ligne correspond au temps d'exécution de Tesseract. La version utilisant Mask R-CNN est plus lente que l'autoencodeur mais plus rapide que BM3D.

par canal auxilaire EM deviennent donc plus attrayantes pour des tiers voulant extraire des informations sensibles. Pour effectuer des attaques par canal auxilaire, les attaquants doivent être assez proches pour intercepter des émanations compromettantes. Comme l'illustre la Figure 4, la méthode proposée basée sur l'apprentissage profond augmente la capacité à extraire des informations du signal bruité. Par conséquent, elle repousse encore plus les limites de l'attaque.

La méthode proposée relève le défi de reconnaître efficacement les caractères dans des images bruitées, là où les approches traditionnelles échouent. Les fuites EM involontaires produisent continuellement des émanations compromettantes, ce qui conduit à une grande quantité de données à traiter lorsqu'on essaie de détecter de l'information. Raisonnablement, cette analyse ne peut pas être faite par un humain pour des raisons de temps et de coûts. La solution consiste alors à automatiser la détection des fuites de données à l'aide de la méthode proposée en essayant continuellement de reconstruire les données.

5.2 Application à la Protection de la Confidentialité

Compte tenu des améliorations précédemment présentées au sujet des attaques par canal auxilaire, il devient important pour les systèmes critiques de détecter et de prévenir la fuite d'information. La méthode que nous proposons peut être transformée en une solution à ce problème. Deux solutions principales peuvent être proposées. La première consiste à auditer un système pour savoir quelle taille de police est la limite sous laquelle il est théoriquement impossible d'extraire des données compromettantes. Une fois cette taille de caractère connue, il suffit de configurer le système d'information pour afficher des caractères plus petits que la taille critique. Une deuxième possibilité est d'afficher une bannière connue sur l'écran cible et d'attendre en permanence que cette bannière soit détectée. Si elle est détectée, il y a un risque de compromission des données traitées par le SI et un indicateur peut être levé pour être traité ultérieurement.

Dans la section 4.2, les résultats du réseau Mask R-CNN sont présentés lorsqu'il est appliqué à la récupération de caractères individuels et indépendants. Pour utiliser la méthode proposée à des fins de protection, il est intéressant de chercher des mots dans le signal qui fuite. Dans ce sens, un mécanisme simple a été mis en place. Tout d'abord, Mask-RCNN est appliqué pour intercepter les données et fournir une liste de caractères indépendants avec leurs *bounding boxes*. Cette liste de caractères est ensuite explorée afin de rechercher des mots d'alerte codés en dur. Comme la méthode ne récupère pas tous les caractères, un rapport arbitraire de caractères reconnus nécessaires pour considérer un mot comme détecté a été défini. Ce rapport est fixé à 80% et peut être changé facilement pour renforcer ou réduire le taux d'alerte. De plus, une vérification est faite pour s'assurer que tous les caractères d'un mot sont détectés sur la même ligne. En effet, les bannières de sécurité sont toujours imprimées sur une même ligne d'un document.

Comme Mask R-CNN récupère les caractères du jeu de données de test avec un score de 83% en utilisant une police supérieure à 34 (Figure 6), ce mécanisme d'extraction de mots peut assurer la levée d'une alerte de détection de mot dans

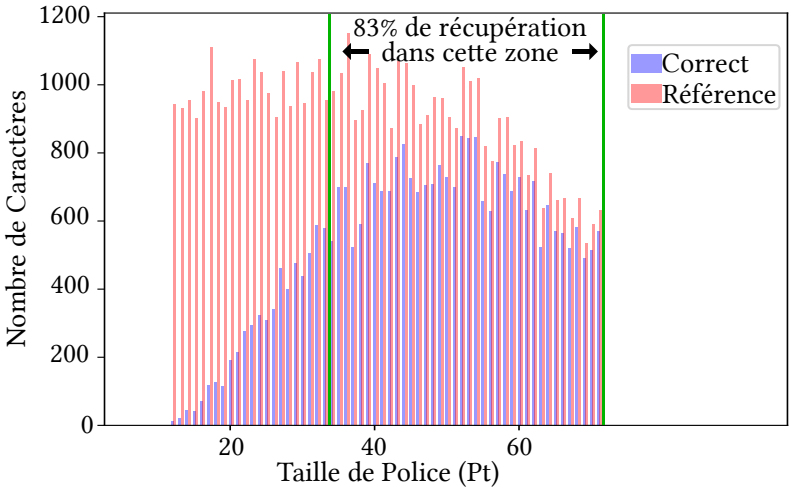


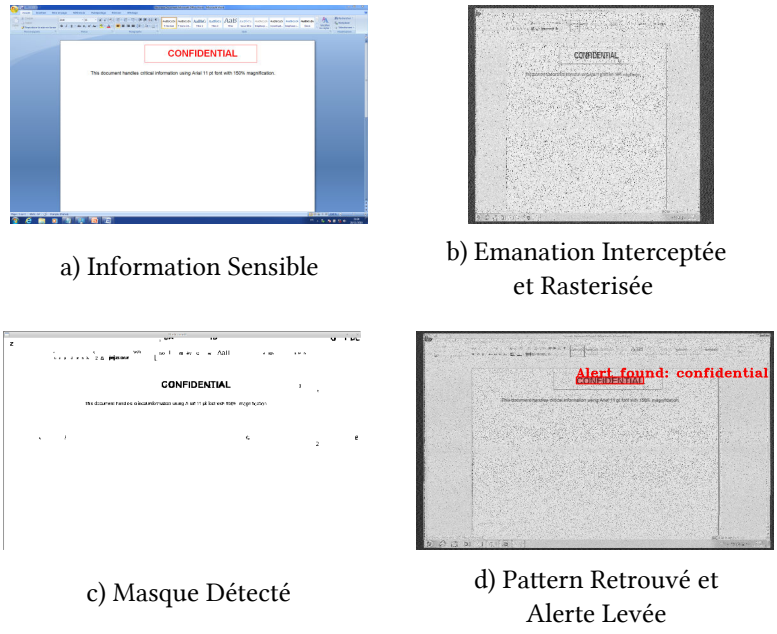
Fig. 6. Nombre de caractères extraits par Mask R-CNN en fonction de la taille des caractères. 83% des caractères de taille supérieure à 34 Pt sont correctement extraits (zone entre les deux lignes verticales vertes).

la plus part des cas. En effet, si 80% d'un mot est nécessaire pour le considérer comme détecté et que le Mask R-CNN extrait 83% des caractères, la détection est proche de certaine. Une fois le mot détecté, un message d'alerte peut être émis. Par exemple, dans la Figure 7, le mot "CONFIDENTIAL" est détecté et un message d'alerte est affiché à l'écran pour avertir l'utilisateur d'une éventuelle fuite de données.

5.3 Extensions Possibles

La méthode exposée présente un F-Score supérieur à 0,6 sur un jeu de test contenant 12563 patchs. Ce jeu de test présente plusieurs types de variabilité obtenus directement à partir de signaux réels interceptés. Le jeu d'entraînement contient lui 98722 patchs et est composé d'échantillons bruts et augmentés. En ce qui concerne le processus de génération des jeux de données, le nombre d'échantillons d'entraînement représente environ 5 000 interceptions complètes d'écran. Le bruit traité étant complexe, il est envisageable d'améliorer les résultats en élargissant le jeu de données d'entraînement et en introduisant plus de variabilité pour améliorer la couverture de l'espace d'entrée. Une autre perspective d'amélioration est l'intégration d'un réseau de proposition de régions voisines. Pour chaque ROI détectée, ce réseau proposerait deux voisins (gauche et droite) pour ajouter du contexte aux décisions et rendrait possible l'utilisation d'un modèle linguistique de type trigramme.

Le système proposé et son approche bout en bout permettant la détection d'émanations compromettantes sont ici basés sur la reconnaissance de caractères.



a) Information Sensible

b) Emanation Interceptée et Rasterisée

c) Masque Détecté

d) Pattern Retrouvé et Alerte Levée

Fig. 7. Système d'alerte proposé : a) le système à protéger traite les données sensibles et affiche une bannière dédiée indiquant "CONFIDENTIAL" ; la SDR et l'ordinateur hôte interceptent et rasterissent une émanation compromettante (b) ; Mask R-CNN débruite l'échantillon (c) et classifie les données, déclenchant une alerte car des données confidentielles sont détectées (d).

Cette partie reconnaissance de caractères est réalisée par le bloc Mask R-CNN. Mask R-CNN a été spécialisé ici pour reconnaître des caractères mais il serait possible d'étendre la méthode à d'autres motifs visuels en le ré-entraînant avec une autre base de données.

6 Conclusion

Gérer les données tout en garantissant la confiance et la confidentialité est un défi pour les concepteurs de systèmes d'information. Nous présentons dans cet article comment la surface d'attaque d'une attaque par canal auxiliaire EM peut être élargie par l'introduction de l'apprentissage profond. Notre méthode extrait automatiquement plus de 57% d'une fuite d'information à une distance de 10 mètres. Nous proposons donc d'utiliser cette méthode pour surveiller un système d'information déployé en temps réel et détecter tout défaut apparaissant sur le système. La proposition est implémentée en logiciel et fonctionne sur un ordinateur avec GPU et un système de SDR.

Remerciements

Ces travaux ont reçu le soutien du Pôle d'Excellence Cyber (PEC), initiative de la région Bretagne et du ministère des armées.

References

1. W. Van Eck. Electromagnetic radiation from video display units: An eavesdropping risk? *Computers & Security*, 4(4):269–286, 1985.
2. M. G. Kuhn. Compromising Emanations of LCD TV Sets. *IEEE Transactions on Electromagnetic Compatibility*, 55(3):564–570, 2013.
3. D. Genkin, M. Pattani, R. Schuster, and E. Tromer. Synesthesia: Detecting Screen Content via Remote Acoustic Side Channels. *arXiv:1809.02629*, 2018.
4. M. Vuagnoux and S. Pasini. Compromising Electromagnetic Emanations of Wired and Wireless Keyboards. *Proceedings of the 18th USENIX Security Symposium*, pages 1–16, 2009.
5. Y. Hayashi, N. Homma, M. Miura, T. Aoki, and H. Sone. A Threat for Tablet PCs in Public Space: Remote Visualization of Screen Images Using EM Emanation. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security - CCS '14*, pages 954–965, Scottsdale, Arizona, USA, 2014. ACM Press.
6. Y. Hayashi, N. Homma, Y. Toriumi, K. Takaya, and T. Aoki. Remote Visualization of Screen Images Using a Pseudo-Antenna That Blends Into the Mobile Environment. *IEEE Transactions on Electromagnetic Compatibility*, 59(1):24–33, 2017.
7. P.-M. Ricordel and E. Duponchelle. Risques associés aux signaux parasites compromettants : le cas des câbles DVI et HDMI. In *Symposium sur la Sécurité des Technologies de l'Information et des Communications (SSTIC)*, 2018.

8. P. De Meulemeester, L. Bontemps, B. Scheers, and G. A. E. Vandenbosch. Synchronization retrieval and image reconstruction of a video display unit exploiting its compromising emanations. In *2018 International Conference on Military Communications and Information Systems (ICMCIS)*, pages 1–7, Warsaw, 2018. IEEE.
9. K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering. *IEEE Transactions on Image Processing*, 16(8):2080–2095, 2007.
10. K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang. Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017.
11. P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.A. Manzagol. Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion. *Journal of Machine Learning Research*, 11:3371–3408, 2010.
12. O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, Lecture Notes in Computer Science, pages 234–241. Springer, 2015.
13. J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila. Noise2noise: Learning Image Restoration without Clean Data. *CoRR*, 2018.
14. National Security Agency. NACSIM 5000 TEMPEST FUNDAMENTALS, 1982.
15. M. G. Kuhn and R. J. Anderson. Soft Tempest: Hidden Data Transmission Using Electromagnetic Emanations. In D. Aucsmith, editor, *Information Hiding*, volume 1525, pages 124–142. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998.
16. G. Myers. A Fast Bit-vector Algorithm for Approximate String Matching Based on Dynamic Programming. *J. ACM*, 46(3):395–415, 1999.
17. A. Mikolajczyk and M. Grochowski. Data augmentation for improving deep learning in image classification problem. In *2018 International Interdisciplinary PhD Workshop (IIPhDW)*, pages 117–122, Swinoujście, 2018. IEEE.
18. K. He, G. Gkioxari, P. Dollar, and R. Girshick. Mask R-CNN. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, Venice, 2017. IEEE.
19. S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, 2017.
20. K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, Las Vegas, NV, USA, 2016. IEEE.
21. J. Long, E. Shelhamer, and T. Darrell. Fully Convolutional Networks for Semantic Segmentation. pages 3431–3440, 2015.
22. D. Mahajan, R. Girshick, V. Ramanathan, K. He, M. Paluri, Y. Li, and A. Bharambe. Exploring the Limits of Weakly Supervised Pretraining. page 23, 2018.
23. T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common Objects in Context. In *Computer Vision – ECCV 2014*, Lecture Notes in Computer Science, pages 740–755. Springer International Publishing, 2014.
24. R. Smith. An Overview of the Tesseract OCR Engine. In *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007) Vol 2*, pages 629–633, Curitiba, Parana, Brazil, September 2007. IEEE.