Adaptation de domaine pour l'analyse forensique d'images

Rony ABECIDAN[1], Vincent ITIER[2,3], Jérémie BOULANGER[1], Patrick BAS[1]

[1]Univ. Lille, CNRS, Centrale Lille,UMR 9189 CRIStAL, F-59000 Lille,France

[2]IMT Nord Europe, Institut Mines-Télécom, Centre for Digital Systems, F-59000 Lille,France

[3]Univ. Lille, CNRS, Centrale Lille, Institut Mines-Télécom, UMR 9189 CRIStAL, F-59000 Lille, France

rony.abecidan@univ-lille.fr, vincent.itier@imt-nord-europe.fr,
jeremie.boulanger@univ-lille.fr, patrick.bas@cnrs.fr

Résumé – Ce papier étudie une stratégie d'adaptation de domaine non-supervisée appliquée à l'analyse forensique d'images. Pour réaliser l'analyse forensique, l'opérateur dispose d'une base source de données étiquetées permettant l'entraînement d'un modèle. Dans un contexte opérationnel, il est commun de devoir travailler sur une base cible de données non-étiquetées, provenant d'une distribution différente de celle de la source, ce qui dégrade les performances du modèle. Nous souhaitons donc généraliser l'apprentissage d'un modèle forensique réalisé sur une base source à une base cible. Pour atteindre cet objectif, il est courant de rechercher un espace où ces deux distributions sont similaires. En se basant sur cette nouvelle représentation des données, on peut alors initier un transfert d'apprentissage depuis la source vers la cible. Cette recherche d'espace invariant peut se faire via une rétro-propagation du gradient sur un coût d'adaptation bien choisie, représentant une distance entre les deux distributions projetées. Attirés par la simplicité de cette stratégie, nous avons décidé de la tester sur un détecteur de manipulations visuelles entraîné pour une situation réaliste.

Abstract – This article studies a domain adaptation strategy applied to digital forensics on images. In order to perform a forensic analysis, the operator is using a labelled source base enabling to train a model. Within an operational framework, we usually work on an unlabeled target base coming from a different distribution compared to the source, hence damaging the model performances. In that situation we want to be able to generalize the learning of a forensics model we can get from a source, to a target. To reach that goal, it is common to search for a space where these two distributions look similar. By taking advantage of this new data representation, we can start a knowledge transfer from the source to the target. For instance, this search of an invariant feature space can be done using a backpropagation mechanism on an well-chosen adaptation loss, representing a distance between the two projected distributions. Attracted by the simplicity of this strategy, we decided to test it on a forensics detector searching for visual manipulations in images trained for a realistic scenario.

1 Introduction

Il est aujourd'hui facile de falsifier des images afin de tromper les observateurs. La manipulation "copier-coller" est par exemple très utilisée dans les réseaux sociaux. Cette manipulation malicieuse, réalisée avec des outils de photomontage de plus en plus efficaces, est pourtant souvent difficile à détecter à l'oeil-nu en analysant sa sémantique. En effet, la zone falsifiée de l'image ne se distingue plus que par ses traces de bruit résiduel liées à l'acquisition, la compression, et aux différents traitements de l'image. Pour détecter ce type de falsification, l'analyste forensique cherche alors à construire un modèle permettant de détecter automatiquement la présence de bruits possédant une signature différente au sein d'une même image.

L'état de l'art dans ce domaine s'appuie sur des modèles d'apprentissage profond [2] qui nécessitent un très grand nombre de données annotées pour leurs entraînements. En forensique, créer une grande base de données étiquetées est un travail fastidieux qui demande beaucoup de ressources, mais il s'agit d'une étape obligatoire pour garantir un apprentissage pertinent. Néanmoins, même s'il existe des détecteurs efficaces pour la détection d'images falsifiées, ils sont souvent perturbés par des images provenant d'une distribution relativement différente de celle de l'entraînement.

Plus précisément, si la base d'images utilisée pour l'entraînement (a.k.a. la source) n'est pas exactement issue de la même chaîne de développement que celle de la base d'images à analyser en pratique (a.k.a. la cible), la performance du détecteur peut être impactée. Cela est d'autant plus embarrassant qu'il est généralement difficile de connaître la chaîne de développement des images que l'on veut analyser en pratique. Nous avions déjà souligné ce fait dans [1] en se concentrant sur l'impact d'une forte compression jpeg pour la détection de falsifications. Nous proposons désormais d'aborder cette problématique avec des chaînes de développement plus complètes et réalistes simulées à l'aide de **RawTherapee**. Il existe différents scénarios pour y répondre. Nous les décrivons dans ce qui suit en utilisant la

terminologie proposée dans [4].

- Si la base d'images cible peut être associée à une autre base d'images provenant de la même distribution, le problème devient alors supervisé. Ce scénario n'est pas vraiment réaliste en forensique puisqu'il est coûteux de construire des images falsifiées. Dans ce contexte que l'on nomme **TgtOnly**, il est alors possible de re-éntrainer le détecteur en utilisant des images provenant de la distribution cible plutôt que de la distribution source (autre cas que nous nommons **SrcOnly**).

- Il est en fait rare de pouvoir identifier la distribution cible en situation réelle et le problème est alors non-supervisé. C'est dans ce cadre là que l'adaptation de domaine non-supervisée est envisageable. Nous avons le sentiment que cet ensemble de méthodes est particulièrement adapté pour l'analyste forensique puisqu'il doit généralement analyser un petit ensemble d'images provenant d'un même domaine cible inconnu. Dans ce cas, il est nécessaire de mettre à jour le détecteur en l'entraînant de nouveau pour considérer l'existence du domaine cible. Nous nommons ce cas **Update**.

Dans ce papier, nous évaluons l'efficacité de l'adaptation de domaine non-supervisée pour répondre au problème d'hétérogénéité des bases que rencontre l'analyse forensique. Nous nous concentrons sur les stratégies consistant à mettre à jour les poids des réseaux de neurones en minimisant conjointement un coût d'entropie croisée calculé grâce aux étiquettes de la source, et, un coût d'adaptation pouvant être assimilée à une distance entre les projections des distributions sources et cibles au niveau des couches denses les plus profondes. Ce genre d'approche est particulièrement intéressante puisqu'elle est simple à mettre en œuvre et n'allonge pas de beaucoup le temps d'entraînement.

2 Adaptation de domaine non-supervisée

On considère une base source $\mathcal{D}_s = \{(\mathrm{x}_i^s, y_i^s)\}_{i=1}^{n_s}$ constituée de n_s observations étiquetées et, une base cible $\mathcal{D}_t = \left\{\mathbf{x}_j^t\right\}_{j=1}^{n_t}$ constituée de n_t observations non-étiquetées. La distribution qui a générée les observations de la source est nommé p_s et la distribution qui a générée les observations de la cible p_t. On suppose que $p_s \neq p_t$ tout en considérant que ces deux distributions sont raisonnablement proches. Un objectif commun en adaptation de domaine consiste à concevoir un classifieur f :

- capable de plonger les deux domaines dans un espace $\mathcal{F}$ au sein duquel on ne peut considérer qu'ils proviennent de deux distributions distinctes. Dans ce cas, on dit que les caractéristiques ainsi obtenues sont invariantes par rapport au domaine.

- qui minimise le risque d'erreur sur la cible :

$$\begin{aligned}\mathcal{R}_t(f) &= \mathbb{E}_{(\mathbf{x},y)\sim p_t}\left(\mathbb{1}_{f(\mathbf{x})\neq y}\right)\\ &= \mathbb{P}_{(\mathbf{x},y)\sim p_t}[f(\mathbf{x})\neq y]\end{aligned}$$

Soit $\Phi_{\mathcal{F}}$ un plongement qui nous permet d'obtenir des caractéristiques pertinentes des domaines cibles et sources. Trouver ce plongement $\Phi_{\mathcal{F}}$ n'est pas une tâche facile mais des récentes approches montrent qu'il est possible de l'apprendre en utilisant le mécanisme de rétro-propagation du gradient. Dans ce contexte, $\Phi_{\mathcal{F}}$ est appris de telle façon à ce que l'adaptation de domaine soit la plus efficace possible. Concrètement, cette stratégie consiste à ajouter au coût de classification classique (entropie croisée binaire) un coût d'adaptation modélisant une distance entre $\Phi^*_{\mathcal{F}}(p_s)$ et $\Phi^*_{\mathcal{F}}(p_t)$:

$$\mathcal{L} = \mathcal{L}_{classification} + \lambda \underbrace{\mathcal{L}_{adaptation}}_{d(\Phi^*_{\mathcal{F}}(p_s),\Phi^*_{\mathcal{F}}(p_t))}$$

où λ contrôle le compromis entre une bonne performance et une bonne adaptation. Dans le contexte forensique, le coût de classification permet d'apprendre au réseau à reconnaître une falsification en utilisant exclusivement la base source et, le coût d'adaptation l'encourage à plonger nos deux distributions source et cible dans un espace où elles semblent similaires juste avant d'effectuer cette détection. Ce coût peut être vu comme une forme de régularisation. La recherche du plongement pertinent devient alors une recherche de distances entre distributions pertinentes. Idéalement, cette distance doit satisfaire plusieurs conditions :

- Étant donné qu'on souhaite utiliser une rétro-propagation du gradient, elle se doit d'être différentiable.

- Nous n'avons pas une connaissance exacte des distributions source et cible, il faut alors être capable d'estimer raisonnablement cette distance à partir d'échantillons provenant de ces distributions.

- Enfin, cette distance ne doit pas être trop gourmande en termes de complexité et d'espace mémoire.

Dans la littérature, il existe de nombreuses distances qui répondent à ces conditions. Actuellement, il y a une tendance à l'utilisation de distances basées sur des noyaux reproduisants (Maximum Mean Discrepancy) [6] ou bien dérivées de la théorie du transport optimal (Wasserstein) [3]. Plus particulièrement, nous nous intéressons aux distances suivantes :

- Les MMDs :

$$\begin{aligned}\mathrm{MMD}^2(\Phi^*_{\mathcal{F}}(p_s),\Phi^*_{\mathcal{F}}(p_t)) = &\ \mathbb{E}_{X,X'\sim\Phi^*_{\mathcal{F}}(p_s)}k\left(X,X'\right)\\ &+ \mathbb{E}_{Y,Y'\sim\Phi^*_{\mathcal{F}}(p_t)}k\left(Y,Y'\right)\\ &- 2\mathbb{E}_{\substack{X\sim\Phi^*_{\mathcal{F}}(p_s)\\ Y\sim\Phi^*_{\mathcal{F}}(p_t)}}k(X,Y)\end{aligned}$$

issues des kernels :

— $k(x,y) = -\|x-y\|$ (Distance énergie)

— $k(x,y) = \exp(-\|x-y\|/\sigma)$ (MMD-Laplacien)

— $k(x,y) = \exp\left(-\|x-y\|^2/2\sigma^2\right)$ (MMD-Gaussien)

- Les distances de Wasserstein d'ordre 1 et 2 (W_1 & W_2) :

$$W_p(\Phi^*_{\mathcal{F}}(p_s),\Phi^*_{\mathcal{F}}(p_t)) = \inf_{(X,Y)\sim\Phi^*_{\mathcal{F}}(p_s),\Phi^*_{\mathcal{F}}(p_t)} \mathbb{E}\left[d(X,Y)^p\right]^{1/p}$$

Toutes ces distances peuvent être estimées correctement à l'aide de la librairie *geomloss* compatible avec *pytorch* [5].

Nous avons donc testé en pratique l'effet d'une adaptation de domaine utilisant ces distances dans un contexte d'analyse forensique.

3 Expériences

Pour nos expériences, nous modifions le fameux classifieur d'images EfficientNet-B0 (E-B0) [8] pour construire notre détecteur de falsifications, en imposant la contrainte de Bayar [2] dans la première couche convolutionnelle :

$$\begin{cases} w_k^{(1)}(0,0) = -1 \\ \sum_{m,n\neq(0,0)} w_k^{(1)}(m,n) = 1. \end{cases}$$

En faisant converger les premières convolutions vers des filtres passe-haut, elle permet de contraindre le réseau à se concentrer dès le début sur le bruit de l'image plutôt que sur le contenu sémantique. La couche la plus profonde de E-B0 est une couche dense. Il s'agit naturellement de la couche qui est la plus spécifique à la distribution d'entraînement. Nous décidons donc d'encourager une adaptation de domaine au niveau de la sortie de cette couche. Concernant le coût d'adaptation, nous n'avons pas explicitement choisi un paramètre λ pour contrôler son impact mais nous l'avons plutôt normalisé en s'appuyant sur la toute première valeur prise par ce dernier.

3.1 Construction des domaines

Nous travaillons avec toutes les images de la catégorie "Splicing" de la base publique DEFACTO [7] dédiée à l'analyse forensique. Nous nous concentrons sur les falsifications de type "copier-coller" car ce sont les plus complexes à identifier en pratique. Le format TIF des images de cette catégorie nous permet d'autre-part de contrôler plus facilement leurs post-traitements.

Pour construire nos domaines, nous découpons notre base d'images en deux ensembles d'images indépendants de taille égales. Puisque chaque image a une dimension spécifique, nous coupons nos images en blocs de taille 128×128. Pour chaque bloc provenant de la source ou de la cible, la classe "falsifiée" est construite à partir des blocs qui ont une surface falsifiée qui occupe entre 5% et 10% de leur surface totale. Nous avons également remarqué que dans le cas où la contrainte de Bayar n'est pas imposée sur la première couche convolutionnelle du détecteur, ce dernier a tendance à s'attacher à la sémantique des images de DEFACTO pour prédire si le bloc est falsifié ou non. En effet, les falsifications concernant peu de types d'objets le détecteur à plus rapidement fait d'apprendre à détecter ces objets. Dans ce contexte, les blocs sélectionnés ne doivent pas contenir une partie trop importante de la falsification.

En se basant sur le nombre de blocs falsifiés, les blocs authentiques sont ensuite sélectionnés de façon aléatoire en quantité égale pour constituer des bases équilibrées. Dans le cas où une chaîne de développement est utilisée pour une base, cette dernière est appliquée avant de découper les images en blocs afin d'éviter l'apparition d'artefacts.

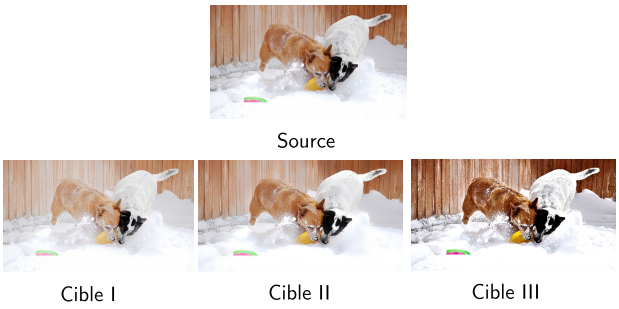

FIGURE 1 – Illustration représentant une image développée selon les différentes chaînes de développements étudiées.

Dans le but de créer des domaines légèrement différents en termes de distributions, nous avons cherché des chaînes de développements réalistes avec le logiciel opensource **RawTherapee**. Trois ont été retenues après avoir observé leurs impacts négatifs sur les performances d'un détecteur entraîné sur les images de la source, compressées avec un facteur de qualité JPEG de 100. La Figure 1 montre visuellement l'effet des chaînes de développements proposées. Comme on peut le voir, ces chaînes de développement sont parfaitement plausibles dans une situation réelle.

3.2 Protocole Expérimentale

Afin d'obtenir des résultats fiables concernant la capacité de notre détecteur à généraliser sur différentes bases, nous réalisons une validation croisée avec 3 découpages, sur la source et la cible en même temps, et ce, pour chaque expérience. À chaque fois environ 20.000 blocs sont utilisés pour l'entraînement et 10.000 pour l'évaluation. Les choix suivants sont faits pour les hyperparametres :

- Le nombre maximal d'epochs est fixé à 20, un nombre suffisant pour observer une convergence en pratique
- La méthode d'optimisation choisie est Adam connue dans la communauté pour son efficacité générale.
- La taille des batchs est fixé à 128, une taille raisonnable pour effectuer des calculs sur des GPUs classiques tout en garantissant un bon comportement pour notre entraînement.
- Le taux d'apprentissage est fixé à 10^{-4}, le meilleur choix trouvé en pratique en se basant sur la source uniquement
- L'initialisation des poids est celle de pytorch par défaut. Nous initialisons à chaque fois notre détecteur avec une seed commune égale à 2022 afin de rendre nos résultats reproductibles tout en assurant une comparaison fiable de nos résultats.

Via geomloss [5], nous avons accès à de nombreuses estimations de distances entre distributions. Nous en avons testés plusieurs afin de pouvoir en tirer des comparaisons. Les paramètres relatifs aux distances s'appuyant sur la théorie du transport optimal sont automatiquement optimisés par la librairie geomloss. Les autres paramètres sont fixés expérimentalement.

Cible	SrcOnly	Update(Distance énergie)	Update(MMD Laplacien)	Update(MMD Gaussien)	TgtOnly
rt-I	**71.9% +/- 1%**	77.6% +/- <1 %	75.6% +/- 1%	76.8 +/- <1 %	**78.5% +/- <1%**
rt-II	**70.3% +/- 1%**	76.8% +/- <1%	75.4% +/- 1%	75.4% +/- <1%	**76.5% +/- 1%**
rt-III	**68.7% +/- <1%**	75.1% +/- <1%	74.3% +/- 1%	74.4% +/- 1%	**77.4 +/- 1%**
qf(100)	**81.9% +/- 1%**	inutile	inutile	inutile	inutile

Cible	SrcOnly	Update(W_1)	Update(W_2)	Update(W_2 non-équilibrée)	TgtOnly
rt-I	**71.9% +/- 1%**	77.6% +/- 1%	77.0% +/- 1%	77.4 +/- 1%	**78.5% +/- <1%**
rt-II	**70.3% +/- 1%**	76.7% +/- <1%	76.7% +/- 1%	76.6% +/- 1%	**76.5% +/- 1%**
rt-III	**68.7% +/- <1%**	74.6% +/- 1%	74.2% +/- <1%	74.4% +/- 1%	**77.4 +/- 1%**
qf(100)	**81.9% +/- 1%**	inutile	inutile	inutile	inutile

TABLE 1 – Performances du detecteur dans différents contextes (en terme de précision).

4 Résultats et interprétations

Nous présentons dans Tab. 1 les résultats de nos expériences. Dans un premier temps, nous voyons qu'un simple changement de chaîne de développement suffit pour perdre plus de 10% d'efficacité sur un ensemble d'images de sémantique identique. Dans un second temps, toutes les distances que nous avons utilisées nous ont permis d'améliorer nettement les performances sur les cibles. En regardant de plus près, la distance énergie est celle qui a permis la meilleure adaptation pour les 3 cibles. Ce résultat est d'autant plus surprenant qu'il s'agit de la seule distance ne nécessitant aucun paramètre à optimiser. On remarque que la performance obtenue en utilisant 20.000 blocs non étiquetés avec cette distance pour l'adaptation $qf(100) \rightarrow rt - II$, est similaire à celle obtenue en étiquetant ces 20.000 morceaux, ce qui remet en question la nécessité d'étiqueter des milliers d'images en pratique. Face à ce constat, nous avons décidé de regarder également l'effet d'une adaptation avec cette même distance, lorsque l'on dispose que d'un petit nombre de blocs étiquetés et non pas 20.000, ce qui est en fait le cas de l'analyse forensique. Nous avons observé qu'il était autant efficace d'utiliser une stratégie d'adaptation de domaine avec seulement 10 blocs non-étiquetés que de prendre le temps d'en étiqueter 5000. En augmentant l'ordre de grandeur du nombre de blocs non-etiquetés en entraînement, nous avons aussi constaté un accroissement de performances cohérent, ainsi que, l'existence claire d'une limite d'amélioration. En l'occurrence, nous n'avons gagné que 0.6% de performance en test en passant de 1000 à 20.000 blocs non-étiquetés.

5 Conclusion & Perspectives

Dans ce papier, nous utilisons une stratégie d'adaptation de domaine afin d'atténuer le problème de décalage de sources en analyse forensique dans une situation aveugle où il n'est pas possible d'accéder aux étiquettes de la base cible. Nous montrons que l'adaptation pour différentes chaînes de développements est possible et efficace en utilisant des distances entre distributions issues de noyaux reproduisants ou du transport optimal. La méthode de mise à jour du réseau exploitée ici ne nécessite pas d'étiquetage de la cible ni d'un grand nombre d'images pour fonctionner. Cette dernière se doit alors d'être étudier plus largement avec d'autres ensembles de données mais également dans des cas où la cible est déséquilibrée.

6 Remerciements

Nos expériences ont été réalisées grâce aux moyens de calcul de l'IDRIS au travers de l'allocation de ressources 2021-AD011013285 attribuée par GENCI. Les travaux présentés dans ce papier ont également reçu un financement de l'Agence Innovation Défense et du programme H2020 de l'Union Européenne, accord de financement No 101021687, projet "UNCOVER".

Références

[1] R. Abecidan, V. Itier, J. Boulanger, and P. Bas. Unsupervised jpeg domain adaptation for practical digital image forensics. In *International Workshop on Information Forensics and Security*, pages 1–6. IEEE, 2021.

[2] B. Bayar and M. C Stamm. A deep learning approach to universal image manipulation detection using a new convolutional layer. In *Workshop on information hiding and multimedia security*, pages 5–10. ACM, 2016.

[3] N. Courty, R. Flamary, A. Habrard, and A. Rakotomamonjy. Joint distribution optimal transportation for domain adaptation. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[4] H. Daumé III. Frustratingly easy domain adaptation. In *Annual Meeting of the Association of Computational Linguistics*, pages 256–263. Association for Computational Linguistics, 2007.

[5] J. Feydy, T. Séjourné, F.-X. Vialard, S.-I. Amari, A. Trouve, and G. Peyré. Interpolating between optimal transport and mmd using sinkhorn divergences. In *International Conference on Artificial Intelligence and Statistics*, pages 2681–2690. PMLR, 2019.

[6] M. Long, Y. Cao, J. Wang, and M. Jordan. Learning transferable features with deep adaptation networks. In *International Conference on Machine Learning*, pages 97–105. PMLR, 2015.

[7] g Mahfoudi, B. Tajini, F. Retraint, F. Morain-Nicolier, J.-L. Dugelay, and M. Pic. DEFACTO : image and face manipulation dataset. In *European Signal Processing Conference*, pages 1–5. IEEE, 2019.

[8] M. Tan and Q. Le. Efficientnet : Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.