Et si les images adverses étaient des images ?

Teddy Furon, Benoit Bonnet, Patrick Bas

Et si les images adverses étaient des images ?*

Benoit Bonnet[1], Teddy Furon[1], and Patrick Bas[2]

[1] Univ. Rennes, Inria, CNRS, IRISA
[2] Univ. Lille, CNRS, Centrale Lille, CRIStAL
teddy.furon@inria.fr

Abstract. Nous partons d'un constat : les images adverses dans la littérature ne sont souvent pas des images au sens où la valeur des pixels n'est pas quantifiée. Même le règlement de la compétition internationale NeurIPS autorise cette absurdité. Cet article propose un post-traitement rapide pour quantifier intelligemment ces images adverses. Il vise à faire un compromis entre l'adversité de l'image et la distortion / détectabilité de la perturbation. Ce papier résume les publications [3, 2].

Keywords: Réseaux de neurones · Apprentissage · Sécurité

1 Introduction

Les données adverses (en anglais *adversarial samples*) sont des petites perturbations appliquées à des données en entrée d'un algorithme IA pour en modifier la sortie de façon arbitraire. La littérature considère en général des données de type images pour facilement illustrer le phénomène mais tout autre type de données est possible : vidéos [10, 22], audio [17, 6], textes [1], séries temporelles [7], *malwares* [16]. De même, en général l'algorithme attaqué est un classifieur, mais d'autres fonctionalités (régression, segmentation, détection, suivi d'objet) sont aussi vulnérables.

Ces perturbations ne sont pas aléatoires mais créées par un adversaire. Dans un scénario en boite blanche, leur création est facilitée car l'attaquant connait les entrailles de l'algorithme ciblé. De nombreuses attaques utilisent le gradient de l'algorithme de manière itérative, calculé par le mécanisme de propagation arrière (*backpropagation*) pour 'inverser' localement un réseau de neurones. Les attaques connues pour leurrer la classification d'images (de la plus simple à la plus évoluée) : FGSM [8], PGD [15], DDN [18], BP [24] et CW [5].

Le constat est le suivant : les perturbations sont de faible amplitude et ainsi à peine visibles à l'oeil nu. Cette extrême sensibilité des réseaux est bien sûr une vulnérabilité quand ceux-ci sont utilisés à des fins de sécurité. Plus largement encore, les données adverses remettent en cause le terme d' Intelligence Artificielle. La communauté vision par ordinateur a réussi à créer des algorithmes neuronaux s'acquittant de la tache difficile de classification mieux que l'humain (plus rapide et avec moins d'erreurs sur le benchmark ImageNet ILSVRC). Ces algorithmes

* Thèse financée par DGA / Inria.

sont donc a priori dignes du label 'Intelligence Artificielle'. Or, les images adverses sont des exemples où ces algorithmes se trompent quasi systématiquement alors qu'aucun humain n'aurait commis d'erreur.

Les données adverses sont un phénomène à la mode en recherche. La littérature foisonne de travaux proposant des attaques, des contre-attaques ou des explications théoriques de la vulnérabilité des réseaux. La communauté s'est aussi organisée en proposant la compétition [11] liée à la conférence annuelle NeurIPS. Les attaques y sont comparées en terme de distortion, probabilité de succès et temps de calcul.

L'idée de ce papier est simple : Dans la grande majorité de ces travaux, et y compris dans cette compétition NeurIPS, **les images adverses ne sont en fait pas des images**. La section 2 introduit quelques notations et défend le constat ci-dessus. Elle montre les impacts de cette brèche. Pallier ce problème n'est pas si simple, la section 3 propose un post-traitement à appliquer après une attaque pour s'assurer que le résultat est bien avant tout une image, qui soit adverse et dont la perturbation est invisible. Ces mécanismes sont inspirées de la dissimulation de l'information. Ils aident à rendre la perturbation invisible que ce soit à l'oeil nu (comme en tatouage numérique) ou statistiquement (comme en stéganographie). Cette publication est une synthèse des papiers [3, 2].

2 Le constat

2.1 Notations

Soit I une image composée de 3 canaux couleurs, de L lignes et C colonnes. Les valeurs des $3LC$ pixels sont codées par des entiers entre 0 et 255. Ainsi, l'image I est un objet discret qui vit dans l'ensemble $\{0, \dots, 255\}^{3LC}$. On considère un classifieur d'image : à une image I, il associe une classe $c(I)$ parmi C classes apprises lors de son entrainement. Ainsi, $c(I) \in \{1, 2, \dots, C\}$.

En général, ce classifieur est composé de trois briques. Un premier traitement normalise l'image : elle calcule une représentation $x = T(I)$. Souvent $x \in [0, 1]^{3LC}$, mais cela peut aussi être $[-1, 1]^{3LC}$. Parfois, $T(\cdot)$ est une simple division par 255 de la valeur des pixels, mais cela peut être $x_i = \alpha(I_i - \beta)$ avec (α, β) dépendant du canal couleur. Ce qui est sûr, c'est que ce pré-traitement est ad-hoc et qu'il est fixe. Il ne fait pas partie de l'apprentissage. Ce traitement d'image est déconsidéré par la communauté *machine learning* car il y a rien à apprendre.

La deuxième brique est le réseau de neurones qui prend en entrée x et donne en sortie les logits $y \in \mathbb{R}^C$. Plus y_i est grand plus l'image est probablement de classe i. La dernière brique est l'opérateur softmax qui normalise y en un vecteur de probabilités $p \in [0, 1]^C$ avec $\sum_{i=1}^{C} p_i = 1$. Ce dernier étage donne aussi la classe prédite comme étant celle qui a la plus grande probabilité associée : $c(I) = \arg\max_i p_i$.

2.2 La brèche

Partant d'une image originale I_o de la classe c_o que l'on suppose bien classée ($c(I_o) = c_o$), l'attaquant souhaite trouver une image adverse I_a proche de I_o mais mal classée : $c(I_a) \neq c_o$ (attaque non ciblée). La majorité des papiers définissent l'image adverse optimale par l'équation:

$$x_a^\star = \arg \min_{x \in [0,1]^{3LC},\, c(x) \neq c_o} \|x - x_o\|. \tag{1}$$

La vraie définition devrait être :

$$I_a^\star = \arg \min_{I \in \{0,1,\ldots,255\}^{3LC},\, c(I) \neq c_o} \|I - I_o\|. \tag{2}$$

Sur les 25 papiers traitant d'images adverses aux conférences CVPR 2019 et ECCV 2019, 88% utilisent (1) au lieu de (2). Pour eux, une donnée adverse est un tenseur x_a (une matrice 3D) contenant des réels codés sur 4 octets en virgule flottante, et non des entiers entre 0 et 255. A notre connaissance, un seul papier propose une attaque (DDN [18]) produisant directement des images quantifiées. Le reste des papiers ne commet pas cette erreur car ils étudient les images adverses dans le monde physique : celles-ci sont imprimées et donc quantifiées.

De la même manière, la règle [11] de la compétition internationale du challenge NeurIPS est sidérante : "*The adversary has direct access to the actual data fed into the model* [c'est-à-dire x]. *In other words, the adversary can choose specific float32 values as input for the model*". Il y a clairement une mauvaise analyse des menaces. Le scénario 'boite blanche' signifie que l'attaquant cible un classifieur (disponible sur un site web, dans un produit fermé *etc*) dont il possède une copie de l'algorithme qu'il est libre d'analyser dans son garage. Cependant, à l'extérieur du garage, c'est une image I_a qu'il doit fournir en entrée à ce classifieur cible ; il ne peut pas modifier ses variables internes. Or, le pré-traitement $T(\cdot)$ fait partie intégrante du classifieur et l'attaquant ne peut pas directement imposer un x_a.

Pour justifier ce choix étonnant, NeurIPS écrit "*In a real world, this might occur when an attacker uploads a PNG file to a web service, and intentionally designs the file to be read incorrectly.*" On imagine que les auteurs pensent à des attaques par dépassement de pile. Ce sont des attaques informatiques qui menacent le décodage du fichier (et non pas l'algorithme de classification) et on connait depuis longtemps des contre-mesures.

2.3 L'impact

L'impact de cette mauvaise définition est multiple.

Tout d'abord, il n'est pas trivial de créer de images adverses quantifiées. On pourrait croire qu'il suffit de trouver x_a, de le faire passer dans la fonction réciproque $T^{-1}(\cdot)$ (linéaire, souvent multiplication par 255), et enfin d'arrondir chaque pixel à l'entier le plus proche entre 0 et 255. Ce procédé est hasardeux. La

caractéristique des données adverses est leur faible distortion. Autrement dit, la perturbation $T^{-1}(x_a - x_o) = T^{-1}(x_a) - I_o$ est de très faible amplitude et l'arrondi à l'entier le plus proche la détruit. La perturbation est quantifiée à 0 sur de nombreux pixels et après quantification, l'image n'est plus adverse. L'article [3] donne une justification théorique. Les papiers de la littérature sont généralement illustrés par des images attaquées. Elles ont forcément été quantifiées, donc ces images présentées comme adverses ne le sont peut-être pas !

La quantification est une contrainte supplémentaire forte. Il est clair qu'en augmentant l'amplitude de la perturbation pour qu'elle résiste à la quantification, on augmente les chances d'obtenir une image quantifiée et toujours adverse. Mais est-ce le meilleur procédé ? La perturbation ne devient-elle pas visible ?

Cette brèche empêche de comparer les classifieurs. Comme dit auparavant, tous n'ont pas le même pré-traitement $T(\cdot)$. Ainsi, mesurer la vulnérabilité d'un classifieur par la distortion moyenne $\|x_a - x_o\|$ ne veut rien dire car ce n'est pas une mesure invariante. Par exemple, il est facile d'augmenter ou diminuer cette mesure arbitrairement: en substituant à $T(\cdot)$ le pré-traitement $T'(\cdot) = \alpha T(\cdot)$ et en multipliant tous les poids de la première couche du réseau par $1/\alpha$, alors on obtient un nouveau classifieur qui fait exactement les mêmes prédictions mais dont la 'vulnérabilité' est multipliée par α.

Cette littérature contient autant de papiers proposant des attaques que des contre-attaques. Ces auteurs montrent que i) leur défense ne dégrade pas les performances du classifieur sur des images originales (donc quantifiées), ii) qu'elle est efficace en soumettant des images attaquées (donc, non quantifiées). Il est amusant de voir que simplement détecter si les données d'entrée sont quantifiées bloquerait la plupart des attaques.

3 La quantification comme un post-traitement

Notre idée n'est pas de construire une nouvelle attaque mais un post-traitement qui quantifie intelligemment une image attaquée. Le schéma est le suivant : partant d'une image I_o, le pré-traitement calcule $x_o = T(I_o)$, une attaque de la littérature donne x_a, et notre post-traitement calcule $\tilde{I}_a = T^{-1}(x_a) \in \mathbb{R}^{3LC}$, puis quantifie intelligemment en $I_a = Q(\tilde{I}_a) \in \{0, 1, \ldots, 255\}^{3LC}$.

3.1 Compromis distortion - adversité

Supposons que l'attaque de la littérature ait réussi, $c(\tilde{I}_a) = c_a \neq c_o$, alors notre post-traitement doit trouver I_a quantifiée proche de I_o tout en restant adverse, c'est à dire de classe c_a. Idéalement, on veut résoudre le problème :

$$I_a = \arg \min_{I \in \{0,\ldots,255\}^{3LC},\, c(I)=c_a} \|I - I_o\|^2. \tag{3}$$

Une première étape propose une formulation Lagrangienne:

$$I_a^{(\lambda)} = \arg \min_{I \in \{0,\ldots,255\}^{3LC}} \|I - I_o\|^2 + \lambda L(I) \tag{4}$$

avec $L(I) = p_{c_0}(I) - p_{c_a}(I)$. En clair, $L(I)$ est la différence entre la probabilité prédite pour la classe originale c_o et celle de la classe c_a. L'image I est adverse si $L(I) < 0$ car alors $c(I) \neq c_o$. Plus $L(I)$ est négatif, plus le classifieur est confiant dans son erreur.

Pour $\lambda = 0$, seule la distortion compte dans le problème (4) et la solution évidente $I_a^{(0)} = I_o$ n'est pas adverse. Pour λ très grand, seule l'adversité compte, et $I_a^{(\infty)}$ est une image (trop ?) adverse mais très éloignée de I_o. Il faut faire un compromis par une recherche dichotomique sur λ. Pour un λ donné, calculer $I_a^{(\lambda)}$ et voir si cette image est adverse. Si oui, on peut baisser λ et voir si on obtient une nouvelle image adverse et plus proche de I_o, sinon on augmente λ.

3.2 Linéarisation

Supposons que l'on se donne q degrés de liberté par pixel (q entier pair). Le pixel $\tilde{I}_{a,i}$ n'est a priori pas un entier et on va le quantifier sur un des q entiers les plus proches : $\lfloor \tilde{I}_{a,i} \rfloor + \{-(q/2-1), \ldots, -1, 0, 1, ..., q/2\}$ (sauf si $\tilde{I}_{a,i}$ est trop proche de 0 ou 255). Pour λ donné, résoudre (4) demande de passer en revue les q^{3LC} combinaisons possibles, soit une complexité exponentielle avec le nombre de pixels.

La linéarisation $L(I) \approx L(\tilde{I}_a) + (I - \tilde{I}_a)^\top \nabla_I L(\tilde{I}_a)$ simplifie le problème en

$$\arg\min \sum_{i=1}^{3LC} (I_i - I_{o,i})^2 + \lambda (I_i - \tilde{I}_{a,i}) g_i + cte \tag{5}$$

où g_i est la i-ème composante du gradient $\nabla_I L(\tilde{I}_a)$. Ce gradient est facilement calculé par la propagation arrière. Cette approximation a cassé un problème NP en une suite de $3LC$ problèmes très simples puisqu'on peut minimiser chaque terme de la somme indépendamment.

3.3 Généralisation

La distortion de la perturbation est mesurée jusqu'à présent par la norme Euclidienne au carré, $\|I - I_o\|^2$. Pour de faible amplitude, cette mesure de la visibilité n'est pas si mal. On peut la remplacer par n'importe quelle autre distance du moment qu'elle reste séparable de la forme $d(I, I_o) = \sum_i w_i(I_i, I_{o,i})$. Nous pensons notamment à des coûts utilisés en stéganographie comme HILL [12], MiPod [19], ou GINA [13, 21]. Ils modélisent non pas la distortion visible mais la détectabilité statistique de la perturbation $I - I_o$.

L'algorithme est alors simple : i) calculer la fonctionnelle $w_i(I_i, I_{o,i}) + \lambda(I_i - \tilde{I}_{a,i})g_i$ pour les q valeurs possibles de I_i, ii) trouver pour quelle valeur la fonctionnelle est à son minimum, iii) itérer sur tous les pixels. D'où une complexité linéraire en nombre de pixels : $O(3LCq \log q)$. Quelques astuces sont possibles, notamment si $w_i(I_i, I_{o,i})$ a une forme quadratique comme dans (5), alors le minimum recherché a une expression simple [3, 2], ce qui évite un tri rapide en $O(q \log q)$ à chaque pixel.

4 Investigation expérimentale

4.1 Protocole

Nos expériences utilisent les images de la compétition NeurIPS [11]. C'est en fait un sous-ensemble d'ImageNet. Les images ont la taille 224×224. Nous testons différents réseaux: ResNet-18, ResNet-50 [9], ResNet-50R qui est une version robustifiée par entrainement adverse [15], mais aussi les tout nouveaux EfficientNet-b0 [20] et sa version robustifiée [23].

Nous mesurons la distortion par la norme Euclidienne normalisée au nombre de pixels $\overline{d} = \|I_a - I_o\|/\sqrt{3LC}$. Les attaques étant des processus à plusieurs paramètres, pour chaque image nous essayons plusieurs jeux de paramètres et retenons celui qui offre une image adverse avec la plus petite distortion. Nous introduisons le concept de caractéristique $\overline{d} \to P_{suc}(\overline{d})$, probabilité que l'attaque réussisse avec une distortion inférieure à $\overline{d}$.

4.2 Arrondir à l'entier le plus proche ne fonctionne pas

La première expérience compare les attaques classiques de la littérature avec et sans quantification. La figure 1 montre clairement que la quantification naïve par arrondi à l'entier le plus proche est une catastrophe : plus aucune image n'est adverse sauf si la distortion est supérieure à 1. Notre quantification est bien plus performante.

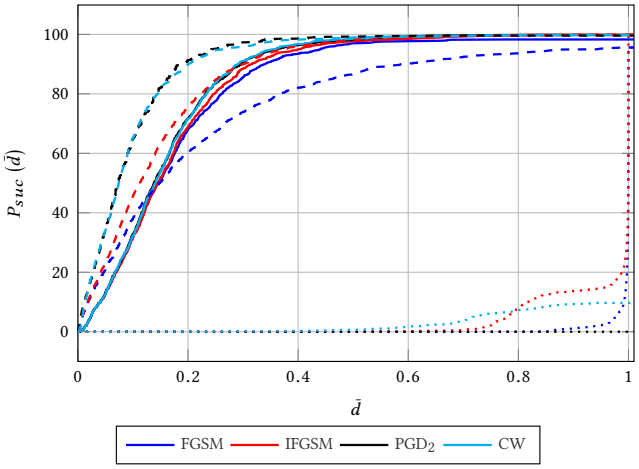

Fig. 1. Caractéristique des attaques FGSM, IFGSM, PGD et CW contre ResNet-18 : sans quantification (tiret), avec quantification par arrondi (pointillé), ou avec la quantification proposée (plein).

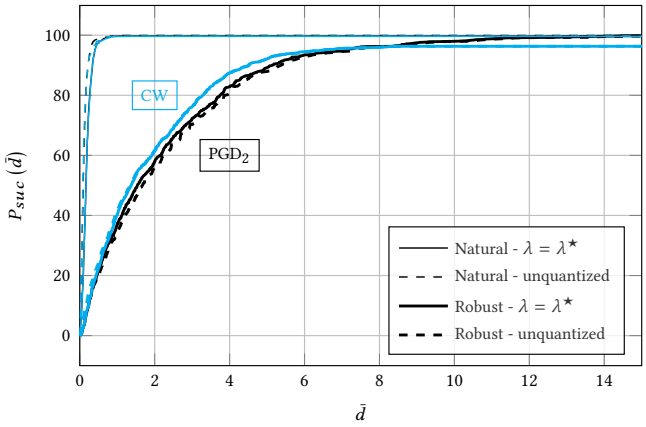

Fig. 2. Caractéristiques des attaques PGD et CW contre ResNet-50 version naturelle et version robustifiée, avec et sans notre quantification.

4.3 La quantification n'est pas une contrainte forte

La deuxième expérience montre que la quantification n'est pas une contrainte forte au sens où les caractéristiques sans ou avec *notre* quantification sont très proches. La figure 2 montre aussi que la version robuste ResNet-50R est effectivement moins vulnérable : l'attaquant doit consommer 10 fois plus de distortion pour leurrer le classifieur.

4.4 Des images adverses non détectables

Cette expérience utilise les nouveaux réseaux EfficientNet (en version naturelle et robuste) et l'attaque PGD$_2$ qui offre des performances similaires à CW tout en étant plus rapide. Le post-traitement quantifie les pixels pour faire un compromis entre adversité et distortion Euclidienne (1ère ligne du tableau) ou détectabilité d'un modèle stéganographique (HILL, MiPod, ou Gina). Pour voir si cette recherche d'indétectabilité est concluante, le défenseur utilise des détecteurs d'images adverses : SCRMQ1 [14] et SRNet [4]. Ces détecteurs sont réglés pour commettre des faux positifs avec une probabilité de 5%.

Le tableau 1 montre que la version robuste du réseau est effective : les attaques sont moins puissantes (probabilité de succès plus petite) tout en consommant plus de distortion. La quantification qui minimise la distortion est celle que l'on a vue jusqu'à présent (1ère ligne du tableau). Cependant, elle est très détectable (à $\approx 90\%$ avec SRNET). Le détecteur SCRMQ1 est moins puissant que SRNet. Les quantifications qui utilisent des couts stéganographiques consomment plus de distortion mais sont moins détectables (avec GINA, 4 images sur 5 ne sont pas détectées). Ceci s'explique par des coûts non stationnaires. L'image originale est analysée et les coûts pour les pixels dans des régions texturées de grande dynamique sont inférieurs à ceux des régions uniformes. Ainsi, la perturbation se concentre dans les régions texturées, qui cachent / masquent ce signal faible de manière perceptuelle et statistique.

Table 1. Détectabilité de l'attaque PGD_2 contre EfficientNet-b0 naturel ou robuste, en fonction de la quantification en post-traitement avec q degrés de liberté par pixel.

	q	P_{suc} (%)		$\overline{d}$		SCRMQ1(%)		SRNet(%)	
		Nat	Rob	Nat	Rob	Nat	Rob	Nat	Rob
$\overline{d}$	2	88.0	71.8	**0.22**	**0.29**	81.2	76.4	93.3	87.5
HILL	2	88.0	71.8	0.24	0.30	74.8	66.3	86.1	77.6
HILL	4	**88.8**	**72.6**	0.27	0.33	72.4	72.4	85.5	72.3
MiPod	2	87.9	71.8	0.26	0.32	74.9	64.3	84.0	76.1
MiPod	4	88.2	72.2	0.29	0.35	72	57.0	82.6	67.5
GINA	2	88.0	71.8	0.43	0.47	5.4	**3.0**	44.2	33.5
GINA	4	88.2	71.9	0.60	0.63	**3.8**	3.1	**20.7**	**14.2**

5 Conclusion

Ce papier a exploré le jeu entre l'attaquant et le défenseur lorsque ces acteurs utilisent les armes les plus récentes : classifieur EfficientNet, attaques CW, détecteur SRNET, et cout stéganographie GINA. La conclusion est sans appel : l'attaquant gagne le jeu. Il a 70% de chances de trouver une image qui leurre à la fois le classifieur et le détecteur en un temps raisonnable. Mais ce jeu du gendarme et du voleur n'est pas fini. Nos résultats sont donnés pour des détecteurs qui n'ont jamais vu d'images adverses a la 'GINA'. La prochaine étape est de les nourrir de ces images à l'apprentissage.

References

1. Behjati, M., Moosavi-Dezfooli, S., Baghshah, M.S., Frossard, P.: Universal adversarial attacks on text classifiers. In: ICASSP. pp. 7345–7349. IEEE (2019)
2. Bonnet, B., Furon, T., Bas, P.: Adversarial images through stega glasses. In: submitted to IEEE WIFS'20 (2020)
3. Bonnet, B., Furon, T., Bas, P.: What if adversarial samples were digital images? In: Proc. of ACM IH&MMSec '20. pp. 55–66 (2020). https://doi.org/10.1145/3369412.3395062
4. Boroumand, M., Chen, M., Fridrich, J.: Deep residual network for steganalysis of digital images. IEEE Transactions on Information Forensics and Security **14**(5), 1181–1193 (2018)
5. Carlini, N., Wagner, D.: Towards evaluating the robustness of neural networks. In: IEEE Symp. on Security and Privacy (2017)
6. Carlini, N., Wagner, D.A.: Audio adversarial examples: Targeted attacks on speech-to-text. In: IEEE Symposium on Security and Privacy Workshops. pp. 1–7. IEEE Computer Society (2018)
7. Fawaz, H.I., Forestier, G., Weber, J., Idoumghar, L., Muller, P.: Adversarial attacks on deep neural networks for time series classification. In: IJCNN. pp. 1–8. IEEE (2019)
8. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. In: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015), http://arxiv.org/abs/1412.6572

9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 770–778 (June 2016). https://doi.org/10.1109/CVPR.2016.90
10. Jiang, L., Ma, X., Chen, S., Bailey, J., Jiang, Y.: Black-box adversarial attacks on video recognition models. In: ACM Multimedia. pp. 864–872. ACM (2019)
11. Kurakin, A., Goodfellow, I., Bengio, S., Dong, Y., Liao, F., Liang, M., Pang, T., Zhu, J., Hu, X., Xie, C., Wang, J., Zhang, Z., Ren, Z., Yuille, A., Huang, S., Zhao, Y., Zhao, Y., Han, Z., Long, J., Berdibekov, Y., Akiba, T., Tokui, S., Abe, M.: Adversarial attacks and defences competition (2018)
12. Li, B., Wang, M., Huang, J., Li, X.: A new cost function for spatial image steganography. In: Image Processing (ICIP), 2014 IEEE International Conference on. pp. 4206–4210. IEEE (2014)
13. Li, B., Wang, M., Li, X., Tan, S., Huang, J.: A strategy of clustering modification directions in spatial image steganography. Information Forensics and Security, IEEE Trans. on **10**(9) (2015)
14. Liu, Y., Moosavi-Dezfooli, S.M., Frossard, P.: A geometry-inspired decision-based attack. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019)
15. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. In: 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings (2018), https://openreview.net/forum?id=rJzIBfZAb
16. Martins, N., Cruz, J.M., Cruz, T., Abreu, P.H.: Adversarial machine learning applied to intrusion and malware scenarios: A systematic review. IEEE Access **8**, 35403–35419 (2020)
17. Qin, Y., Carlini, N., Cottrell, G.W., Goodfellow, I.J., Raffel, C.: Imperceptible, robust, and targeted adversarial examples for automatic speech recognition. In: ICML. Proceedings of Machine Learning Research, vol. 97, pp. 5231–5240. PMLR (2019)
18. Rony, J., Hafemann, L.G., Oliveira, L.S., Ayed, I.B., Sabourin, R., Granger, E.: Decoupling direction and norm for efficient gradient-based l2 adversarial attacks and defenses. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
19. Sedighi, V., Cogranne, R., Fridrich, J.: Content-adaptive steganography by minimizing statistical detectability. Information Forensics and Security, IEEE Transactions on **11**(2), 221–234 (2016)
20. Tan, M., Le, Q.V.: Efficientnet: Rethinking model scaling for convolutional neural networks. arXiv (2019)
21. Wang, Y., Zhang, W., Li, W., Yu, X., Yu, N.: Non-additive cost functions for color image steganography based on inter-channel correlations and differences. IEEE Trans. on Information Forensics and Security (2019)
22. Wei, X., Liang, S., Chen, N., Cao, X.: Transferable adversarial attacks for image and video object detection. In: IJCAI. pp. 954–960. ijcai.org (2019)
23. Xie, C., Zhang, Z., Zhou, Y., Bai, S., Wang, J., Ren, Z., Yuille, A.L.: Improving transferability of adversarial examples with input diversity. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
24. Zhang, H., Avrithis, Y., Furon, T., Amsaleg, L.: Walking on the edge: Fast, low-distortion adversarial examples. IEEE Trans. on Information Forensics and Security (2020)