ECAI 2023
K. Gal et al. (Eds.)

doi:10.3233/FAIA230369

Exploring the Feasibility of Physical Adversarial Attacks: A Cybersecurity Study

Ronan Hamon[a;*] **and Henrik Junklewitz**[a]

[a]Joint Research Centre, European Commission, Ispra, Italy
ORCiD ID: Ronan Hamon https://orcid.org/0000-0003-1987-5707,
Henrik Junklewitz https://orcid.org/0000-0002-0452-6865

Abstract.

Adversarial machine learning (AML), by designing attacks that intentionally break or misuse state-of-the-art machine learning models, has become the most prominent scientific field to explore the security aspects of Artificial Intelligence. A whole range of vulnerabilities, previously irrelevant in traditional ICT, have effectively emerged in these studies. In the light of upcoming legislations mandating security requirements for AI products and services, there is a need to understand how AML techniques connect with the broader field of cybersecurity, and how to articulate more tightly threat models with realistic cybersecurity procedures.

This article aims to contribute to closing the gap between AML and cybersecurity by proposing an approach to study the feasibility of an attack in a cybersecurity risk assessment framework, illustrated with a specific use case of an evasion attack designed to fool traffic sign recognition systems in the physical world. The importance of considering the feasibility of carrying out such attacks under real conditions is emphasized through the analysis of two factors: the reproducibility of the attack according to a published description or existing code, and the applicability of the attack by a malicious actor operating in a real-world environment.

1 Introduction

Artificial intelligence (AI) has become one of the focal points of the ongoing digital transformation. As with any technology, proper regulation and standardisation are eventually needed to ensure that its use will stay safe, secure and respectful of fundamental rights, societal values and law. This idea, often summarised under the term "Trustworthy AI" [21], applies transversely in a number of sectors with potential harmful impacts on the people's life, such as health, banking, or transport. To this end, regulatory bodies and governments all over the world are already advancing respective digital policy agendas [30], with the proposal for a regulation of AI published in 2021 by the European Commission [12] (the so-called "AI Act") being one of the flagship policy developments introducing a risk-based approach and a set of requirements to achieve trustworthy AI.

A central pillar to achieve trustworthiness in high-risk AI applications is cybersecurity, which plays a major role in addressing concerns related to the robustness and reliability of AI systems when operating in adverse conditions. From a cybersecurity perspective, the most relevant field of AI research is currently Adversarial Machine Learning (AML), focusing on the design and evaluation of attacks that intentionally break or misuse features of state-of-the-art machine learning (ML) models. Since its beginnings in the early 2000s [5] the field has highlighted intrinsic flaws of ML models and a whole range of vulnerabilities specific to ML models previously irrelevant in the cybersecurity of traditional ICT systems.

To leverage this accumulated academic knowledge for daily cybersecurity practice, it is necessary to connect AML tools and threat models more tightly to realistic cybersecurity procedures, a need well understood in scientific communities [17, 32]. In cybersecurity, concrete threats for deployed software systems are usually analysed following an applied and system-specific risk analysis framework [38], which is much broader in scope than in typical AML studies. To this date, studying more applied approaches of modelling threats in AML remains an underrepresented field of study [8, 17], especially for complex deep models and in cyber-physical contexts. As a matter of fact, many technical challenges considered as core components to any cybersecurity conformity testing with regulatory requirements remain open questions, such as the feasibility of measuring robustness against cyberattacks on ML models [6], cybersecurity testing of AI software [51], or properly assessing the strength of model defences [46]. With more AI regulation such as the AI Act becoming enacted in the coming years, these are especially crucial questions for international AI standardisation efforts [39]. In cybersecurity, questions of AI security testing and threat modelling will have to be handled eventually in the context of AI risk management [43] and information security standards such as the ISO 27000 series [22].

This work aims to contribute to closing the gaps between AML and cybersecurity and thereby add to a formalisation of AI cybersecurity, and to shed light on research beyond the focus on performance and accuracy of AI models as the dominating research objectives. We consider this shift towards security aspects of ML as crucial to achieving trustworthiness of safety-critical AI systems. In more detail, potential threats from physical adversarial attacks are addressed from an applied cybersecurity perspective, and the inclusion of AML with cybersecurity is formalized in a conceptual model for AI cybersecurity risk assessment, where the feasibility of carrying out an adversarial attack influences the modelling of threats. The feasibility is studied focusing on two broad aspects: the reproducibility of the attack from published description or code, and the applicability of conducting the attack under real conditions. This conceptual model is illustrated through a use case of an evasion attack from the recently published GRAPHITE framework [16], designed to fool traffic sign

* Corresponding Author. Email: ronan.hamon@ec.europa.eu

recognition systems by applying adversarial physical patches. The aim is not to carry out an exhaustive quantitative analysis, but rather to provide elements to discuss the benefits and shortcomings of current practices in AML with regard to the evaluation of cybersecurity in real-world systems.

The paper is structured as follows: Sec. 2 provides a background on AML and physical evasion attacks, giving context for the presentation of the conceptual model in Sec. 3 and the description of the experiments in Sec. 4. The use case is presented throughout the paper to illustrate specific points. Finally, Sec. 5 provides a set of concluding remarks and lessons learnt from the use case.

2 Background

2.1 Adversarial Machine Learning

The field of AML started from early work which focused on applying cybersecurity principles to ML problems for security controls such as network intrusion detection systems or spam filtering [4,5,9]. Modern AML emerged with the introduction of evasion attacks on deep neural networks [42] and expanded to other types of ML systems (e.g., reinforcement learning [13], language models [24], etc.), and to new types of attacks (backdoors [20], membership inference [37], etc.). In general, today, studies in AML focus on research into intentionally attacking, breaking or misusing features of general machine learning models and how to increase and measure robustness against such approaches. In particular, current research looks into deep neural networks and their vulnerabilities. These topics are increasingly relevant in the task of practically securing AI systems, and it stands to reason that described attacks and approaches may lead to the exploitation of a new class of vulnerabilities by threat actors to attack real-world AI systems [26,32].

However, the AML literature is often actually concerned with fundamental questions of generalizability and theoretical robustness of models [18,42], and not necessarily always with applied problems of practical cybersecurity [5,6,17]. For instance, many works on adversarial examples rely on restricted mathematical threat models based on constrained optimisation (e.g., L_p-norm based adversarial attacks aiming at enforcing low-intensity perturbations). This provides valuable insights about the functioning, accuracy and reliability of models, but it has been argued that these specific types of L_p-norm based threat models can be of limited use when connected to real-world cybersecurity problems [6,17].

Therefore, connecting currently available AML tools and mathematical threat models more tightly to realistic cybersecurity procedures will be needed in order to leverage the accumulated academic knowledge for daily cybersecurity practice [32]. As it stands, many technical challenges considered as core components to any cybersecurity testing of ML models remain open questions, such as measuring robustness against cyberattacks on ML models [6,51], properly assessing the strength of defences and model resilience [46] or understanding the feasibility of proposed attacks and threat models under real-world conditions [17].

2.2 Evasion attacks on image classifiers

Among the new types of vulnerabilities discussed in AML, the creation of adversarial examples in so-called evasion attacks have been most prominently studied in the research community. They are identified as a major potential threat for computer vision systems, in particular when employed to sense and understand the environment in an automated way. Attacks on classification models typically aim to alter the class returned by the system, either in a non-targeted fashion (the attacker only wants the system to return an incorrect class) or in a targeted setting (the attacker selects beforehand the desired class they want the system to output). Since the introduction of evasion attacks in early works [5,42], a wide number of studies have focused on their design and on improving the optimisation algorithms to solve them, suggesting new ways to increase the success rate of attacks and their versatility in constrained settings [7,19,27,41,42].

2.2.1 Physical adversarial evasion attacks

We consider an RGB image $\boldsymbol{x} \in \mathbb{R}^{W\times H\times C}$, with W and H respectively the width and height of the image and $C = 3$ the number of channels. The image represents an instance of a given category c belonging to a larger set of categories C. The task of image classification consists in training a classifier f_θ with θ the list of trainable parameters, such as the output $f(\boldsymbol{x}) = y$ equals the category of the instance present on the image $\boldsymbol{x}$.

An adversarial attack consists in finding and applying a perturbation $\boldsymbol{\delta} \in \mathbb{R}^{W\times H\times C}$ on an image such that it leads to an incorrect classification by a classifier f_θ, i.e.,

$$f_\theta(\boldsymbol{x} + \boldsymbol{\delta}) = f_\theta(\tilde{\boldsymbol{x}}) = \tilde{y} \neq c \tag{1}$$

where the $+$ sign indicates element-wise addition and $\tilde{\boldsymbol{x}}$ is called the adversarial image. Most adversarial attacks formulate the problem as an optimisation problem, where an objective function describing the attack is minimized over the perturbation by gradient descent techniques $\boldsymbol{\delta}^* = \arg\min_{\boldsymbol{\delta}} H(\boldsymbol{x}, y_t, \boldsymbol{\delta}; \lambda, \theta)$. The various types of attack found in the literature are characterized by the design of specific objective functions and the use of efficient optimisation schemes.

The classical framework of adversarial attacks (referred to as digital attacks) considers that the perturbation $\boldsymbol{\delta}$ is applied on the digital representation of the image, i.e., affecting the numerical value of pixels before inputting the classification model. In physical adversarial attacks, the perturbation is meant to be applied on physical objects, before the digital acquisition of the image. In this setting, the adversary has only control over a portion of the image, characterized as a mask $\boldsymbol{M} \in \{0,1\}^{W\times H}$ indicating whether the pixel can be perturbed or not. The perturbation is then defined as an element-wise product $\boldsymbol{M} \cdot \boldsymbol{\delta}$. The selection of the mask is either done by the adversary, or driven by the attack. In the following, we refer to the perturbation as the adversarial patch.

Several works have proposed to generate physical adversarial examples in real-world scenarios. The work by Eykholt et al. [15] serves as a major reference for physical attacks on traffic signs. The objective is to examine the possibility to craft physical perturbations for real-world objects that are able to change the prediction of a classifier, while remaining effective under various physical conditions. The resulting perturbations do not aim to be invisible to human but rather inconspicuous, for example looking like graffiti, leading at least casual observers to ignore the perturbations. More recently, Feng et al. [16] introduced the GRAPHITE framework for optimising the generation of physical adversarial examples, in white-box and black-box settings. As the most advanced, sound framework to generate physical adversarial patches, which is accompanied by sufficient documentation and code, the GRAPHITE attack, and more specifically the attack conducted in white-box settings, will be used as illustration in Sec. 4.

3 AI Cybersecurity: conceptual model and use case illustration

Applied cybersecurity is driven by an analysis of the security risks associated with a concrete software system under a specific set of conditions, paired with the development of proof-of-concept exploits designed to demonstrate security weaknesses in the system. The underlying concept is to conduct a cybersecurity risk assessment, a crucial aspect of which is to estimate the specific expected threats to systems. Often these are best estimated by concretely demonstrating the feasibility of attacks to exploit vulnerabilities.

The inclusion of ML in digital systems introduces a number of novel cybersecurity risks and specific issues that, while not changing the basic premises of the approach, require deeper consideration to develop a model for AI cybersecurity risk. To some degree it is clear that ML programs are basically another form of software for which many processes and practices from classical cybersecurity should apply. In particular, any practically relevant ML software will be part of a larger system, including both AI and classical software components. Thus, in principle, basic and well-tested approaches in cybersecurity risk assessment and organisational aspects of information security do apply as much to ML as to other digital systems [32,44]. Beyond that, the aim should be to address AI-specific issues by bringing together perspectives of AML, complexities of deep neural networks, and cybersecurity principles [31,32]. It should be noted that this system-based view of cybersecurity likely permits addressing AI-specific vulnerabilities with non-AI based mitigation measures and controls, albeit at additional costs, either economical or by limiting the usability of AI components in the larger system, e.g. through only controlled access. This study deliberately will not focus on the system perspective to cybersecurity risk and instead focuses on the potential and challenges of AI-model specific cybersecurity. In any case, even for non-AI based mitigation strategies, it will be necessary to properly address the threats from AI-based attack vectors, such as evasion attacks of individual AI components.

By now, a range of works has been published on the adaptation of applied cybersecurity terminology and concepts to machine learning systems, some of which with quite a practical security perspective [44, 45]. However, neither the maturity of available technical tools nor practical experience with modern ML systems in widespread deployment are already sufficient to observe established practices. The crucial task of threat modelling for AI systems is thus one of the most underdeveloped steps in practice, and demonstrating the feasibility of conducting AI-specific attacks relevant for cybersecurity remains an active field of research and development. As an illustration, we provide a discussion of a use case of traffic sign recognition in autonomous vehicles (AVs), inspired by the more in-depth cybersecurity analysis found in [10].

3.1 AI Cybersecurity Risk

Risk can formally be understood as the product of the likelihood of an event to occur with its estimated impact. In practice, for the cybersecurity risk assessment of a specific software — including ML — this approach essentially aims at estimating the likelihood of an attack from an analysis of threats and vulnerabilities combined with an understanding of the potential impacts of an attack on the system to be protected [33,38]. Often a quantitative analysis of cybersecurity risk is prohibited by the enormous hurdles to achieve meaningful measures, metrics and enough test data, a problem which for AI cybersecurity is exacerbated by the complexities introduced by ML models. It should be noted, that eventually, the mathematically grounded nature of ML algorithms may actually allow a more direct integration into more rigorous risk frameworks such as using the theory of PAC learning with noisy labels [32], or methods of formal verifications. Currently, most approaches will rather focus on empirically exploring the factors describing the risk:

Vulnerabilities and impacts (system-internal) Identifying vulnerabilities in a system and thereby systematically probing its attack surface and understanding possible impacts when the system would be compromised are internal factors of the risk that depends on the system to be protected. This includes understanding and potentially estimating the robustness of the ML system against cyberattacks, and analysing the possible space of attack types that could be employed against a particular ML systems;

Threats (system-external) : Analysing the threats is focusing on the external factors of the risk, and includes the estimation of an adversary's goals and capabilities. This includes the extent of available knowledge and controllable context factors, choice of goals, attack selection and the feasibility to conduct them.

The focus is placed in this paper on the external threats and their modelling for a concrete machine learning based application, since the factors are the least understood and developed elements in AML [17] from the perspective of applied cybersecurity. Usually, concrete threats for deployed software systems are analysed following a very applied and system-specific threat analysis framework [38], which is much broader in scope than for typical discussions in the AML literature [8,17], especially for complex deep models and in cyber-physical contexts. In addition, threat modelling has received much theoretical attention in previous works discussing the cybersecurity of ML [17,32].

While we consequently not address in detail the ongoing discussions about how to measure the robustness of ML models [8] and how to address the arms cycle of new attacks and defences in AML [6,46], we clarify that these dimensions are of course of equal importance to estimate overall cybersecurity risk. We will address aspects of them throughout the paper, but without any claim to be exhaustive.

3.2 Feasibility of attacks in AI threat modelling

As described in [32], besides the goals and motivations of an attacker, technical factors are equally entering threat modelling. For ML systems these largely depend on the analysis of potential capabilities of adversaries, which directly connects to an analysis of the feasibility to conduct attacks in the chosen adversarial context. The feasibility may be affected by factors specific to the chosen attack, such as their implementation or ease of use and by contextual factors, which are out of the direct control of an attacker, but derived from the environment in which an attack is taking place. These could be due to details of the ICT infrastructure in case of a purely digital attack, but also due to real-world influences in case of a physical attack on a cyber-physical systems, such as an AV. An additional element entering a feasibility analysis of an attack is stemming from the fact that, while attacks may be published in the AML literature, it cannot be assumed that they can be reproduced as is. This depends largely on publicly available research code, which often is not yet mature enough, or does not follow enough established coding practices [35, 49], to be readily used in an applied cybersecurity setting.

The exploration in this paper is thus based on dividing the practical analysis of technical aspects of AI threat modelling into two parts: reproducibility and applicability.

3.2.1 Reproducibility

Reproducibility looks at the implementation of the attack itself, and in particular how it would be possible, for a cybersecurity analyst, to include the impact of the attack in a risk assessment. In the applied cybersecurity context of this paper, this mostly amounts to a discussion around available implementations of published research in AML and their readiness for direct application. However, this aspect touches as well upon a general discussion of reproducibility of results in machine learning, usually in a context of establishing more rigorous scientific practice. Reproducibility of scientific results is a well-identified, yet often overlooked challenge in the machine learning research community [34]. In general, being able to conduct experiments and obtain similar outcomes is not only crucial to ensure trust in scientific findings inside the scientific community, but also to allow for a faster integration with other communities.

For implementations in ML, reproducibility issues can stem from two different sources. Firstly, inaccuracies may be inherent to the used algorithms and approaches because of stochastic elements, for example when a random seed is used for initialisation, or when the exact testing conditions cannot be reproduced because the exact same model weights are not available. Secondly, reproducibility may be limited because of issues with publicly available implementations, which may differ from the one used to publish results. This problem may be due to small factors, e.g. missing information on specific hyperparameter settings, limited documentation or heterogeneity if coding practices. Or, it may stem from altogether missing features in a published code or from an insufficient description of the exact algorithmic procedures in a paper, so that exact re-implementation simply becomes impossible. In general, for better or worse, it cannot be assumed that research-made code is always following established typical software development practices [35,49].

In our context of applied cybersecurity and threat modelling, the reproducibility of an attack mainly also becomes relevant for the capacity of a cybersecurity analyst to read and understand the attack, to use and extend it in different contexts (e.g., with other models, data, computing environment, objectives), and if there are fundamental limits to the reproducibility of published results (e.g. due to stochastic elements in the code or different available models for testing the attack), to understand and quantify these limits. Understanding the reproducibility of published results on attacks is a first element to judge the feasibility of using the attack in a real-world setting, and uncovering limiting factors in terms of their use by attackers, notably in terms of resources.

3.2.2 Applicability

Applicability describes how feasible the attack is in adversarial and uncontrollable environments, and is transferable to other types of adversarial conditions. For evasion attacks, this means looking into factors that influence the robustness of the attack in a real setting, from simulating various effects in the digital context, testing transferability to similar models in a grey box setting, and analysing the feasibility and complexity of conducting physical attacks. From a cybersecurity perspective, this goes into analysing the robustness of the attack, and also connects to discussions about an adversary's resources and strategies. For instance, strictly speaking, if the result of an attack can be reproduced with spending less resources by using a different attack vector, it should not be considered as very likely to occur [17].

Compared to digital attacks, the application of the perturbation happens in the physical world before the acquisition of the image, leading to a number of variability factors that strongly alter the way the adversarial patch is perceived. This affects the shape of an adversarial patch (distortions, occlusions, resizing, etc.) and the colours (reduced contrast, colour blending, fading, etc.). For the use case of physical adversarial attacks on image classifiers, we can thus further distinguish applicability in sources stemming from the execution of the attack and from the image acquisition.

At the time of execution of the attack, the main factors of variability include the generation of the patch from a source image using a dedicated attack framework; the printing of the patch; and the placing of the patch on the target object. At the time of the image acquisition, the main factors of variability are either environmental or due to the acquisition material, and includes the lighting, with an effect on the brightness of the object that may vary because of weather or uncontrolled sources of lights; possible occlusion of the object and of the patch due to the presence of debris between the object and the camera or due to environmental effects such as weather or dirt; the distance and angle to which the camera sees the object; the camera sensitivity and noise.

3.3 Example: Evasion attack on traffic sign detection in autonomous vehicles

The following scenario is based on the discussions on the cybersecurity of ML presented in [10], with the uptake of computer vision techniques in autonomous vehicles (AVs).

Threat scenario: Adversaries may introduce physical perturbations on traffic signs to deceive the AI-based perception module of AVs into perceiving wrong information about the environment. This includes alterations or the placement of stickers on road signs. Alternatively, attackers may gain remote access to AVs or even the firmware servers and could potentially conduct similar attacks digitally and at scale across a fleet of AVs. Both variants of the use case involve the assumption of technically adept attackers investing a considerable amount of resources, with knowledge in AML and in attacking classical software systems.

Vulnerabilities: The possibility to conduct evasion attacks against computer vision models is well understood and proven. All known ML algorithms including deep neural network architectures are known to be susceptible to adversarial attacks. Furthermore, it is likely that adversarial patches can be transferred between different models. Mitigation is only partially possible, and effectively only against known attacks at training time.

Impacts: Carefully crafted adversarial patterns may lead to a misclassification of objects or symbols on traffic signs, and subsequently to potential impacts on road and passenger safety, including death. A successful attack at scale against a fleet of AVs would increase the potential impact by orders of magnitude. Affected stakeholders or targets may also include the AV manufacturers.

Threat modelling: The likelihood of an occurrence of this threat scenario is hard to estimate. Real threat models on attacker's motivations and cost-benefits of evasion attacks against traffic signs are not empirically backed-up and even debated in the literature. It is for instance argued that causing a misclassification of a traffic sign can be achieved without conducting an elaborate adversarial attack [17]. This analysis would change substantially if this attack would occur at scale and be conducted against a fleet of AVs.

A non-exhaustive study using this illustrative use case is presented in Sec. 4. It should be noted that even if this illustration is targeted to computer vision problems, the principles of the study follow typical work in AML and can basically be applied to any type of evasion

attacks, models, and data.

For the study, we leave out details of modelling potential motivations of attackers to conduct operations against AVs, which do not depend significantly on the AI technologies involved, to focus on the technical feasibility of conducting such an evasion attack under real conditions. These aspects are still not well understood, nor easily verified empirically for lack of available data on traffic sign recognition model performance in real applications.

4 Experimental evaluation of the feasibility of a physical adversarial attack

In this section, we propose an analysis of the feasibility of a physical adversarial attack in order to illustrate the conceptual model and its challenges in the context of the AV use case, as described in Sec. 3.

The use of computer vision systems in AVs has become a field of relevance in cybersecurity for its potential impact on the safety of citizen. Almost all perception tasks (e.g., sign and object recognition, localisation or object tracking) performed in AVs are currently based on deep neural network architectures [25,28,29,36,48,52]. All these systems are susceptible to evasion attacks and other vulnerabilities specific to such systems [2,14,47,50].

For our analysis, we focus on the recently introduced attack framework GRAPHITE [16], and more specifically the attack under white-box settings. This attack is analysed with regard to the needs of a cybersecurity analyst following the conceptual cybersecurity model introduced in Sec. 3, in particular with regard to the concepts of reproducibility and applicability. For the purposes of our experiments, we are using the same data set and model that was used in [16]. The data for training the model and testing the evasion attacks is from the German Traffic Sign Recognition Benchmark (GTSRB) [40]. The traffic sign recognition model GTSRBNet is trained on GTSRB augmented with random translation, rotation and shear, and is based on a basic convolutional neural network architecture.

4.1 The GRAPHITE white-box attack

The GRAPHITE attack allows generating physical patches to fool image classifier, and presents the following features: handling of fixed mask shape, robustness to environmental conditions (lighting, viewpoint, etc.), and working in white-box and black-box settings.

The attack is a two-stage process: for a given initialisation mask (typically covering all the image), an adversarial patch is obtained by solving an optimisation problem similar to the C &W ℓ_0 attack [7]. This includes the optimisation over a set of n_t transformations of the reference image with a maximum number of iterations n_r, in order to take into account variability due to environmental conditions. The EoT robustness is computed as the success rate of the attack over the set of transformations. Once a minimal value of robustness τ_r is achieved, the mask is cut out by removing regions with low overall gradients. A new patch is then generated, and this process iterates until the attack goes unsuccessful, the final patch being the result of the last successful attack.

We implemented our own version of the GRAPHITE white-box attack based on the information in the publication and the publicly available code[1]. As in [16], we use the following values for the parameters: $n_t = 100$, $n_r = 200$, $\tau_r = 0.8$, and run the experiments over ground-truth/target pairs of a ten-class subset of the GTSRB dataset, containing the labels: `Stop (14)`, `Speed Limit 30 (1)`, `Speed Limit 80 (5)`, `Pedestrians (27)`, `Turn Left (34)`, `Yield (13)`, `Caution (18)`, `Roundabout (40)`, `End of Overtaking (41)`, `Do Not Enter (17)`.

The implementation was written using the `python` language, the `pytorch` framework for the ML components, and the `kornia` library for image transformation. All experiments were run on two NVIDIA A6000. The code to reproduce the experiments is available on the platform https://code.europa.eu.

4.2 Reproducibility

Two aspects of reproducibility are studied in the context of the analysis of the GRAPHITE white-box attack:

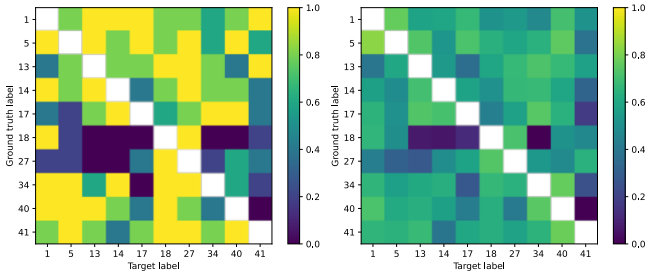
1. How easy it is to reproduce the attack itself, including re-implementing the details of the attack based on the details in [16] and compare to the original implementation;
2. How easy it is to reproduce the results presented in the publication.

Experiment 1: faithfulness of outputs. We test how well the outputs of the published code can be reproduced by our own implementation of the GRAPHITE white-box attack following the descriptions in [16]. One of the difficulties encountered in doing this alone was the absence of a precise description of the different steps followed to implement the white-box attack, especially with regard to hyper-parameter settings. There are several settings in the attack for which default parameters are only documented in the original implementation of the code, for instance the number of iterations and step size used, or some of the parameter settings for the l_0-optimised pruning of the adversarial patch. In addition, we had to fix a few technical issues in the published code. After overcoming these challenges, we can reproduce outputs within a certain degree. We see that outputs for the adversarial image are within an average difference of $\bar{\delta}_p = 0.011 \pm 0.038$, where the average is over a subset of the tested image-target pairs. Exploring the range of parameters gives a variety of results, for example reducing the number of transforms in the expectation-over-transforms optimisation from 100 to 10 increases the average difference to $\bar{\delta}_p^{n=10} = 0.024 \pm 0.081$ and there is considerable fluctuation depending on the image-target pair, with some being quite accurate and others worse than the average results. For a 24-bit RGB image, the reported results correspond to differences of 2 to 5 in terms of pixel intensity.

Experiment 2: faithfulness of results In Tab. 1, we show the results of reproducing the published white-box attack results in Tab. 2 of [16]. We compare the published results with the results from our own implementation, having shown that we were able to reproduce outputs reasonably well in the previous experiment.

The results in Tab. 1 show firstly that by using the default set of transforms for the optimisation of the adversarial patch as used in the published results, we can reproduce the published results reasonably well. Interestingly, applying no transformations during the optimisation for the patch (i.e., on the original image with obtained patch, without other modifications), the robustness performance of the GRAPHITE white-box drops notably. Even more significantly, exchanging the set of transforms during the attack decreases the performance quite significantly, from a reported 77.53% to an average of 54.97% over several runs and transform types. Both observations indicate that the attack may somewhat overfit to the specific set of transforms used in the default settings. In Fig. 1, we showcase how

[1] See https://github.com/ryan-feng/GRAPHITE



(a) No transformation (b) Different transformations

Figure 1: EoT robustness for multiple pairs (y_{gt}, y_{tg}) calculated over different sets of transformation and averaged over multiple repetitions. **(a)** No transformations applied. **(b)** A different set of $n = 100$ transformations than the default settings for GRAPHITE.

these results further depend on the specific image-target pair (y_{gt}, y_{tg}), which is the main source of variance in the experiments.

Table 1: Reproducibility of the white-box GRAPHITE attack using the publicly available implementation and our implementation. (OI: Original Implementation; Own Implementation: NoT: No transformation; AT: Attack transformations; DT: Different set of transformations).

Source	OI	NoT	AT	DT
Avg. EoT Robustness	77.53%	68.44%	75.37%	54.97%

4.3 *Applicability*

The main objective behind the testing of applicability of the GRAPHITE white-box attack is to further assess the overall feasibility in the context of threat modelling, and to give some estimate of the actual likelihood of the physical GRAPHITE attack occurring in a real threat scenario. The analysis is not intended to be exhaustive, but aims to provide elements about what would kind of analysis would be needed to assess the threat more thoroughly.

Applicability is largely governed by the robustness of physical adversarial attacks, which, as discussed in Sec. 3.2.2, depends on external factors beyond code reproducibility, mostly the attack execution, such as errors in patch fabrication or the accuracy of applying a physical patch, and the image acquisition, which may introduce variability due to environmental factors and camera sensor capabilities.

Applicability can be tested to some degree in purely digital environments by simulating these effects. To model both, execution and acquisition, transformations may be applied to the digital patch and image. Typically, a set of such image transformations is used to increase the robustness of physical attacks in applying the expectation-over-transformations formalism to optimise the creation of the adversarial example over a range of transformations [3, 15]. Image transformations usually are either taken to be from a set of typical image-wise digital manipulations such as geometric or intensity variations as in for GRAPHITE [16], or use a combination of digital transformations and real-world variability of recorded effects to account for complexities of the physical world [15]. In addition, we argue that camera and sensor effects might need to be taken into consideration and, crucially, that patch-wise transformation should be added to account for variability in physically applying the patch on a real sign.

4.3.1 *Digital experiments*

Experiment 3: Applicability We consider two levels of transformations: 1) patch-wise level for execution errors and 2) image-wise level for real-world variability. All transformations are further differentiated between geometric and intensity transformations, which include camera effects such as blurring and filtering, largely in line with [16]. A patch P is obtained using a physical adversarial attack with target label y_{t_x} on a reference image x with ground truth label y_{g_x}. In our experiments, we consider the set of transformations introduced in Fig. 2. When the number of transformation increases, the EoT robustness is lower, showing the impact on performances of transformations that may arise in real-world conditions.

Table 2: Applicability of the white-box GRAPHITE attack. The EoT Robustness is averaged over all image-target pairs from the subset of GTSRB. Patch-wise transformations: rotation of 5°(R), change of hue of 0.5 (H). Image-wise transformations: Gaussian blurring ($\sigma = 0.5$) and Gaussian noise ($\sigma = 0.1$) + negative or positive change of brightness of 0.5 (B- and B+), and of saturation of 0.5(S- and S+), change of perspective of 0.5 (P).

R	**H**	**B**	**S**	**P**	**EoT Robustness**
					0.70 ($\sigma = 0.34$)
	x				0.65 ($\sigma = 0.35$)
		-			0.54 ($\sigma = 0.37$)
		+			0.35 ($\sigma = 0.35$)
			-		0.54 ($\sigma = 0.36$)
			+		0.65 ($\sigma = 0.35$)
				x	0.23 ($\sigma = 0.29$)
x	x	-	-	x	0.18 ($\sigma = 0.25$)
x	x	-	+	x	0.14 ($\sigma = 0.21$)
x	x	+	-	x	0.13 ($\sigma = 0.24$)
x	x	+	+	x	0.07 ($\sigma = 0.17$)

4.3.2 *Physical experiments*

In this experiment, the objective is not to conduct an extensive testing nor to emulate a real situation on the roads, but rather to qualitatively explore the full challenge when assessing the thread of a physical adversarial attack in the real-world. One difficulty in physical testing is how to control the full range of physical effects and environmental conditions. We have conducted the tests in the somewhat controlled conditions of a lab with a real European STOP sign. This allows us to have a certain degree of control of the distance, angle, artificial lighting and occlusion effects of the STOP sign. The adversarial patch is applied as accurately as possible at the exact position as determined by GRAPHITE. The camera is of a type Canon EOS 300, the adversarial patches are printed on a standard A4 80g white sheet of paper using a printer of model Canon ImageRunner Advance C5550i.

Fig. 3 shows some results of the experiment. As indicated by the previous experiment, qualitative tests shows that the attack is very susceptible to small transformations and noise introduced in the process. The difference between Fig. 3b and Fig. 3c is a small adjustment in camera angle and direction, changing the attack success. While the attack was relatively robust to changes in lighting, changes in angle or distance had a strong effect on robustness.

5 Conclusion

In this article, we presented a first analysis of AI cybersecurity experiments designed to explore the threat models behind physical evasion

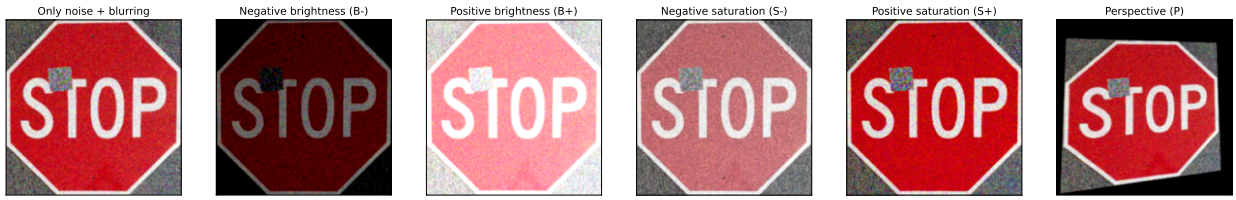


Figure 2: Image-wise transformations used in our experiment to simulate environmental conditions. In addition to these transformations, two patch-wise transformations are used: Rotation (R) and positive hue change (H).

(a) Stop sign (b) Stop sign (c) Stop sign

(d) Stop sign (e) Stop sign (f) Stop sign

Figure 3: Examples of outputs obtained with the GRAPHITE attack.

attacks against traffic sign recognition systems. The aim of this work was to understand how to practically assess the associated cybersecurity risks of physical adversarial attacks through the testing of a deep learning-based perception system, potentially in use in driving assistance functions. Another objective was to establish a firmer understanding of what is needed to produce robust physical adversarial patches even just under standard current laboratory conditions.

Among the many lessons learnt during the process, we note that, first, it was not possible on a narrow case — with limited time resources but expertise in the fields of AI and cybersecurity — to successfully reproduce current research results and establish a stable pipeline for producing robust physical adversarial patches on traffic signs case, even under our strict laboratory conditions. We conclude that it is not easily possible to conduct these type of attacks under real road conditions without a significant investment of resources and knowledge. This connects to previously discussed issues in reproducibility in general in the field of ML.

Second, the feasibility of reproducing the GRAPHITE white-box attack was severely hampered by the limited availability of technical details in published paper and code. A significant portion of our work was invested into writing our own code and solutions and robustly reproduce published results. Thus, while we succeeded in it, this came at a considerable investment of programming resources and time investigating the codes. Currently, this would realistically lower any threat assessment for the discussed use case, especially when considering that similar effects might be achievable by simply randomly modifying a sign [17].

Third, if we go beyond the default settings of the attack, we see indications that the GRAPHITE attack may be less robust to general transformations than one could expect, and that results may somewhat more significantly depend on the choice of transforms and parameters. This may put into question how much published AML results depend on parameter settings and test data and can be used to extrapolate threats due to theoretically studied attacks. In addition, the publicly available implementation had some technical issues that call into question the reliability of some results reported in the publication. The availability of standardised libraries of attacks such as ART [1], alongside benchmark platforms such as RobustBench [8], is a significant step to harmonise practices and broaden access to attacks to non-ML specialists.

Fourth, the feasibility of conducting physical attacks seems severely limited by the complexity involved in actually transferring digital attacks into physical adversarial patches, in particular with respect to the printing and placing of the patches on targets due to concrete issues such as imperfections and noise introduced by printers cameras, and manual handling of the patches. These aspects tend to be overlooked in the description of experimental protocol in scientific publications and are therefore not well-documented nor standardised, limiting reproducibility.

A central conclusion of our work is that at least when considering the GRAPHITE white-box attack, there is currently no standard software solution for creating physical adversarial patches for practical use. As a follow-up of this work, a deeper analysis of GRAPHITE attacks and other frameworks for creating physical adversarial examples would be needed. In particular, more analysis could be placed on the technical aspects that are not carefully addressed in this study, such as the details of implementation and a standardised evaluation process for physical adversarial examples.

These results provide additional insights on the risks that physical adversarial machine learning presents to computer vision models and better anticipate potential cybersecurity vulnerabilities in AI systems. At the current stage, it remains hard to assess these risks, considering the difficulty to measure the robustness and the feasibility of attacks, as discussed in this work. Furthermore, efforts on standardising processes and methods in AI cybersecurity eventually will be faced with defining tested and useful tools and processes for AI threat modelling and security testing. For instance, standardisation bodies such as ISO/IEC or ETSI are already developing standards with regard to AI security controls [11,23]. Currently, analysing, understanding and countering the threat of physical adversarial examples for individual AI models seems to be outside the technological availability of developed tools.

References

[1] Adversarial Robustness Toolbox (ART) v1.14. Trusted-AI, 2023.

[2] S. Agarwal, J. O. D. Terrail, and F. Jurie. Recent Advances in Object Detection in the Age of Deep Convolutional Neural Networks. Preprint arxiv: 1809.03193, 2019.

[3] A. Athalye, L. Engstrom, A. Ilyas, and K. Kwok. Synthesizing Robust Adversarial Examples. In *Proceedings of the 35th International Conference on Machine Learning*, pages 284–293. PMLR, July 2018.

[4] M. Barreno, B. Nelson, R. Sears, A. D. Joseph, and J. D. Tygar. Can machine learning be secure? In *Proceedings of the 2006 ACM Symposium on Information, Computer and Communications Security*, ASIACCS '06, pages 16–25. ACM, 2006.

[5] B. Biggio and F. Roli. Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition*, 84:317–331, 2018.

[6] N. Carlini, A. Athalye, N. Papernot, W. Brendel, J. Rauber, D. Tsipras, I. Goodfellow, A. Madry, and A. Kurakin. On Evaluating Adversarial Robustness, 2019.

[7] N. Carlini and D. Wagner. Towards evaluating the robustness of neural networks. In *Proceedings of the IEEE Symposium on Security and Privacy*, 2017.

[8] F. Croce, M. Andriushchenko, V. Sehwag, E. Debenedetti, N. Flammarion, M. Chiang, P. Mittal, and M. Hein. RobustBench: A standardized adversarial robustness benchmark. Preprint arXiv:2010.09670, 2021.

[9] N. Dalvi, P. Domingos, Mausam, S. Sanghai, and D. Verma. Adversarial classification. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 99–108, New York, NY, USA, August 2004. ACM.

[10] G. Dede, R. Hamon, H. Junklewitz, R. Naydenov, A. Malatras, and I. Sanchez. Cybersecurity challenges in the uptake of artificial intelligence in autonomous driving. Technical Report JRC122440, Publications Office of the European Union, 2021.

[11] ETSI. ETSI GR SAI 004 - Securing AI problem statement, 2020.

[12] European Commission. Proposal for a Regulation laying down harmonised rules on Artificial Intelligence, 2021.

[13] T. Everitt, V. Krakovna, L. Orseau, and S. Legg. Reinforcement learning with a corrupted reward channel. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, 2017.

[14] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, F. Tramer, A. Prakash, T. Kohno, and D. Song. Physical Adversarial Examples for Object Detectors. 2018.

[15] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song. Robust Physical-World Attacks on Deep Learning Visual Classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1625–1634, 2018.

[16] R. Feng, N. Mangaokar, J. Chen, E. Fernandes, S. Jha, and A. Prakash. GRAPHITE: Generating Automatic Physical Examples for Machine-Learning Attacks on Computer Vision Systems. In *2022 IEEE 7th European Symposium on Security and Privacy (EuroS&P)*, pages 664–683, June 2022.

[17] J. Gilmer, R. P. Adams, I. Goodfellow, D. Andersen, and G. E. Dahl. Motivating the Rules of the Game for Adversarial Example Research. Preprint arXiv: 1807.06732, 2018.

[18] J. Gilmer, N. Ford, N. Carlini, and E. Cubuk. Adversarial Examples Are a Natural Consequence of Test Error in Noise. In *Proceedings of the International Conference on Machine Learning*, pages 2280–2289. PMLR, 2019.

[19] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. Preprint arXiv:1412.6572v3, 2015.

[20] T. Gu, K. Liu, B. Dolan-Gavitt, and S. Garg. BadNets: Evaluating Backdooring Attacks on Deep Neural Networks. *IEEE Access*, 7:47230–47244, 2019.

[21] High Level Expert Group on Artificial Intelligence. Ethics Guidelines for Trustworthy AI, 2019.

[22] ISO/IEC JTC 1/SC 27. ISO/IEC 27001:2022 - Information security, cybersecurity and privacy protection — Information security management systems — Requirements, 2022.

[23] ISO/IEC JTC 1/SC 7. ISO/IEC AWI 27090 - Cybersecurity — Artificial Intelligence — Guidance for addressing security threats and failures in artificial intelligence systems, 2023.

[24] R. Jia and P. Liang. Adversarial Examples for Evaluating Reading Comprehension Systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017.

[25] K. Lim, Y. Hong, Y. Choi, and H. Byun. Real-time traffic sign recognition based on a general purpose GPU and deep-learning. *PLOS ONE*, 12(3):e0173317, 2017.

[26] A. Malatras, I. Agrafiotis, and M. Adamczyk. Securing machine learning algorithms. Technical report, ENISA, 2021.

[27] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard. DeepFool: A simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2574–2582, 2016.

[28] G. Mu, Z. Xinyu, L. Deyi, Z. Tianlei, and A. Lifeng. Traffic light detection and recognition for autonomous vehicles. *The Journal of China Universities of Posts and Telecommunications*, 22(1):50–56, 2015.

[29] D. Nassi, R. Ben-Netanel, Y. Elovici, and B. Nassi. MobilBye: Attacking ADAS with Camera Spoofing. Preprint arxiv: 1906.09765, 2019.

[30] OECD. National AI polices and strategies dashboard. https://oecd.ai/en/dashboards/overview, 2022.

[31] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami. The limitations of deep learning in adversarial settings. In *Proceedings of IEEE European Symposium on Security and Privacy*, pages 372–387. IEEE, 2016.

[32] N. Papernot, P. McDaniel, A. Sinha, and M. P. Wellman. SoK: Security and privacy in machine learning. In *Proceedings of the IEEE European Symposium on Security and Privacy*, pages 399–414, 2018.

[33] C. P. Pfleeger and S. L. Pfleeger. *Analyzing Computer Security: A Threat/Vulnerability/Countermeasure Approach*. Prentice Hall Professional, 2012.

[34] J. Pineau, P. Vincent-Lamarre, K. Sinha, V. Larivière, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and H. Larochelle. Improving reproducibility in machine learning research (a report from the NeurIPS 2019 reproducibility program). *The Journal of Machine Learning Research*, 22(1):164:7459–164:7478, 2022.

[35] R. Pruim, M.-C. Gîrjău, and N. J. Horton. The importance of good coding practices for data scientists, 2022.

[36] Y. Saadna and A. Behloul. An overview of traffic sign detection and classification methods. *International Journal of Multimedia Information Retrieval*, 6(3):193–210, 2017.

[37] R. Shokri, M. Stronati, C. Song, and V. Shmatikov. Membership Inference Attacks Against Machine Learning Models. In *Proceedings of the 2017 IEEE Symposium on Security and Privacy*, pages 3–18, 2017.

[38] A. Shostack. *Threat Modeling: Designing for Security*. Wiley Publishing, 1st edition, 2014.

[39] J. Soler Garrido, S. Tolan, I. H. Torres, D. Llorca, Fernandez, V. Charisi, E. G. Gutierrez, H. Junklewitz, R. Hamon, D. F. Yela, and C. Panigutti. AI watch: Artificial intelligence standardisation landscape update. Technical Report JRC131155, European Commission, 2023.

[40] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural Networks*, 32:323–332, 2012.

[41] J. Su, D. V. Vargas, and K. Sakurai. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 2019.

[42] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. In *Proceedings of the International Conference on Learning Representations*, 2014.

[43] E. Tabassi. Artificial Intelligence Risk Management Framework (AI RMF 1.0). Technical report, 2023.

[44] E. Tabassi, K. J. Burns, M. Hadjimichael, A. D. Molina-Markham, and J. T. Sexton. A taxonomy and terminology of adversarial machine learning. Draft NISTIR 8269, 2019.

[45] The MITRE Corporation. MITRE ATLAS. https://atlas.mitre.org/, 2022.

[46] F. Tramer, N. Carlini, W. Brendel, and A. Madry. On Adaptive Attacks to Adversarial Example Defenses. In *Advances in Neural Information Processing Systems*, volume 33, pages 1633–1645. Curran Associates, Inc., 2020.

[47] D. Wang, C. Li, S. Wen, Q.-L. Han, S. Nepal, X. Zhang, and Y. Xiang. Daedalus: Breaking Non-Maximum Suppression in Object Detection via Adversarial Examples. 2020.

[48] Y. Wang, A. Fathi, A. Kundu, D. Ross, C. Pantofaru, T. Funkhouser, and J. Solomon. Pillar-based Object Detection for Autonomous Driving. *arXiv:2007.10323 [cs]*, 2020.

[49] G. Wilson, D. A. Aruliah, C. T. Brown, N. P. C. Hong, M. Davis, R. T. Guy, S. H. D. Haddock, K. D. Huff, I. M. Mitchell, M. D. Plumbley, B. Waugh, E. P. White, and P. Wilson. Best Practices for Scientific Computing. *PLOS Biology*, 12(1):e1001745, 2014.

[50] C. Xie, J. Wang, Z. Zhang, Y. Zhou, L. Xie, and A. Yuille. Adversarial examples for semantic segmentation and object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1369–1378, 2017.

[51] J. M. Zhang, M. Harman, L. Ma, and Y. Liu. Machine Learning Testing: Survey, Landscapes and Horizons. *IEEE Transactions on Software Engineering*, pages 1–1, 2020.

[52] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu. Traffic-Sign Detection and Classification in the Wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2110–2118, 2016.