UNIVERSITY OF
FORWARD
THINKING
WESTMINSTER

WestminsterResearch
http://www.westminster.ac.uk/westminsterresearch

Detection of Physical Adversarial Attacks on Traffic Signs for Autonomous Vehicles

Villarini, B., Radoglou-Grammatikis, P., Lagkas, T., Sarigiannidis, P. and Argyriou, V.

This is a copy of the author's accepted version of a paper subsequently published in the proceedings of the 2023 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT), Bali, Indonesia 13 - 15 May 2023.

The final published version will be available online at:

UNKNOWN DOI

© 2023 IEEE . Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

The WestminsterResearch online digital archive at the University of Westminster aims to make the research output of the University available to a wider audience. Copyright and Moral Rights remain with the authors and/or copyright owners.

Detection of Physical Adversarial Attacks on Traffic Signs for Autonomous Vehicles

Barbara Villarini
School of Computer Science and Engineering
University of Westminster
London, United Kingdom
b.villarini@westiminster.ac.uk

Panagiotis Radoglou-Grammatikis
K3Y Ltd
Sofia 1612
Bulgaria
pradoglou@k3y.bg

Thomas Lagkas
Department of Computer Science
International Hellenic University
Kavala, Greece
tlagkas@cs.ihu.gr

Panagiotis Sarigiannidis
Department of Electrical and Computer Engineering
University of Western Macedonia
Kozani, Greece
psarigiannidis@uowm.gr

Vasileios Argyriou
Department of Networks and Digital Media
Kingston University
London, United Kingdom
vasileios.argyriou@kingston.ac.uk

***Abstract*—Current vision-based detection models within Autonomous Vehicles, can be susceptible to changes within the physical environment, which cause unexpected issues. Physical attacks on traffic signs could be malicious or naturally occurring, causing incorrect identification of the traffic sign which can drastically alter the behaviour of the autonomous vehicle. We propose two novel deep learning architectures which can be used as detection and mitigation strategy for environmental attacks. The first is an autoencoder which detects anomalies within a given traffic sign, and the second is a reconstruction model which generates a clean traffic sign without any anomalies. As the anomaly detection model has been trained on normal images, any abnormalities will provide a high reconstruction error value, indicating an abnormal traffic sign. The reconstruction model is a Generative Adversarial Network (GAN) and consists of two networks; a generator and a discriminator. These map the input traffic sign image into a meta representation as the output. By using anomaly detection and reconstruction models as mitigation strategies, we show that the performance of the other models in pipelines such as traffic sign recognition models can be significantly improved. In order to evaluate our models, several types of attack circumstances were designed and on average, the anomaly detection model achieved 0.84 accuracy with a 0.82 F1-score in real datasets whereas the reconstruction model improved performance of traffic sign recognition model from average F1-score 0.41 to 0.641.**

***Index Terms*—scene analysis, anomaly detection, Generative Adversarial Network, attack restoration, autonomous vehicles**

I. Introduction

Modern intelligent and automated cars are vulnerable to environmental attacks, for example actions against traffic signs [1] would impact the functionality of autonomous vehicles [2]. Academic and industrial research groups have performed advanced vision-based scene analysis for detecting anomalies and enhance cybersecurity. These improve robustness and car safety, particularly with respect to privacy, authenticity, and integrity, in order to address existing security issues and vulnerabilities of autonomous vehicles. Research works have demonstrated the application of adversarial Machine Learning (ML) methods to scene structural entities like pedestrians [3], [4], traffic signs, Advanced Driver-Assistance System notifications and alerts, which can be compromised. Adversarial attacks cause minor input changes, which despite the fact that they cannot be noticed by humans, they can produce significant deviations in the detection models' estimates. This type of attacks only concern Deep Learning models, with latest studies in [5], [6], revealing that printed adversarial attacks applied to networks in different illumination conditions, could also work. 3D printed samples are also erroneously classified by networks at various orientations and scales, as described in [7].

Processes like sensor fusion [8], perception [9], scene analysis [10], and path planning [11] in smart and autonomous vehicles can be executed using Machine Learning approaches. ML methods and especially Deep Learning techniques, are susceptible to specific visual-based attacks [1], [5], [6], [12], [13] than can induce unexpected or even dangerous behaviours in autonomous vehicles. Shortcomings in the existing autonomous and connected vehicles make them susceptible to attacks on physical objects like traffic signs [1]. Hence, the formulation of a system design to address physical attacks is necessary.

This work concerns traffic signs issues, which are considered crucial for the following reasons:

- Cases such as fake traffic signs tricking Tesla cars [14], have shown that addressing such issues is needed for the development of the automotive industry.
- Traffic signs are found in noisy uncontrolled setups, with highly dynamic physical conditions such as weather, lighting, viewing angle, and distance.
- Traffic signs can be simply accessed in public, insecure spaces, hence, they can be easily targeted by physical adversarial attacks on the behaviour of autonomous vehicles [15], without requiring special knowledge or tools.
- Traffic signs are inherently crucial for the safety of transportation.

The main contributions of our work are:

- A novel anomaly detection and mitigation strategies for the environmental attack specially focusing on the traffic signs.
- Two novel Deep Learning architectures, first to detect anomalies and second to generate clean traffic signs without any anomalies, in order to have a robust traffic sign architecture.
- Considering the lack of standardised approach to evaluate the physical attacks, we propose two different groups of attacks to study the effectiveness on vastly different methods.
- Finally, we evaluate our attacks on the real-world GTSRB dataset [16] and demonstrate that a general purpose Deep Learning architecture can be used to detect such completely different types of anomalies.

II. Attack overview

This section provides an overview of the proposed adversarial attacks. In the threat modelling section, we establish our assumption and the requirements for an attack, followed by the proposed attack methods.

A. Threat Modelling

In this experiment, we have made a few key assumptions regarding the attacker, their approach and the outcome of the attack. We based our threat modelling on [2] and they are a) knowledge threshold, equipment awareness, attacker position, limitations and attack outcome.

- *Knowledge threshold* - We assumed that the attacker does not have any prior information about any sensors used in targeting the autonomous vehicle.
- *Equipment awareness* - Again, we assumed that the attacker does not have any access to equipment used in the autonomous vehicle. However, the attacker might have general information about autonomous vehicle architecture (i.e., we assumed that the attacker is aware about the use of cameras for navigation purposes).
- *Attacker position* - Since our experiment deals with physical adversarial attack, we assume that the attacker is positioned outside the autonomous vehicle and has no direct access to the vehicle.
- *Attack outcome* - The attack is an black-box attack and the goal of such attack would be to degrade the performance of the decision engine in the autonomous vehicle and cause anomalous behaviour.

B. Attack Models

In the following section, we present the brief overview of several types of attack models that have been considered for physical adversarial attack on the traffic signs. These attacks are an untargeted attack and our experiments only considered black-box where the attacker has no access to decision engines.

- *Gaussian Noise Attack* - Gaussian Noise is a class of statistical noise with a probability density function following the normal distribution, also known as Gaussian Distribution. The Gaussian or Normal noise is the main cause behind the grey value distribution in digital images [17].
- *Poisson Noise Attack* - Poisson noise is a basic form of uncertainty related with light measurements, which is intrinsic to light's quantized nature and the independence of photon detection. The expected magnitude depends on the signal properties and is the main source of image noise, except in conditions of limited light [18].
- *Speckle Noise Attack* - Speckle noise is a granular pattern, multiplicative in nature, usually present in synthetic-aperture radar or satellite images.
- *Pattern Attack* - Pattern attacks constitute a type of physical attacks on a specific entity. They are also known as graffiti attacks. For instance, if a tape, sticker, or any particular pattern is applied to an object, it could lead to misclassification of the object or make it completely undetectable.
- *Mask Attack* - In this case, a mask is formed to delimit the surface area of the target object. This defines the physical region, or spatial locality, which will be attacked [1].

III. Methodology

In the following, we describe our approach on the dataset selection, generation as well as our methods used for traffic sign attacks.

A. Dataset

The deep learning model needs vast amounts of data for training [19]. In the following, we describe the datasets that were used for training and testing purposes for our models.

1) German Traffic Sign Recognition Benchmark (GTSRB) dataset: GTSRB is a traffic sign recognition dataset [16] and is widely used for the traffic sign recognition benchmarking purpose. The dataset consists of multi-class traffic signs captured in various locations in Germany. It has 43 different labeled traffic signs with varying image resolution. It contains a total of $50,000$ images. Likewise, the dataset only consists of no duplicate instance of any given traffic signs.The traffic sign are captured in day time only.

2) German Traffic Sign Recognition Meta (GTSRM) Dataset: In order to train our proposed mitigation model (GAN model), a twin dataset of GTSRB was generated. The dataset has an exact number of traffic signs as well as classes. The only difference is the GTSRM dataset does not consist of any background information. In addition, only 43 different images of each traffic class are generated and these are duplicated to mirror the classes and the samples of GTSRB (i.e., the dataset consists of same instance of traffic signs for each class).

B. Model Architecture

In the following, we describe two proposed Deep learning architectures for traffic sign anomaly detection and for reconstruction models.

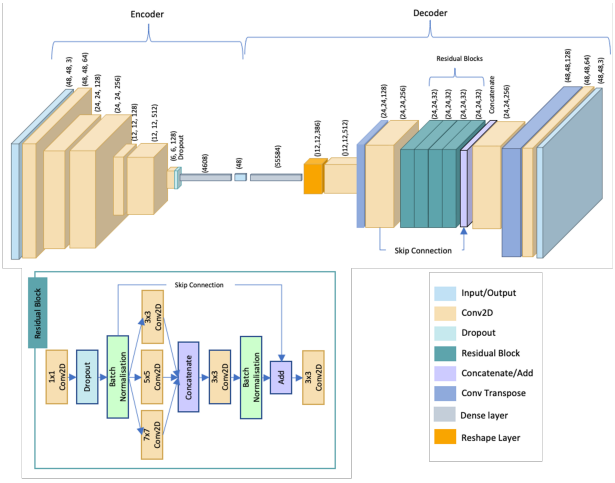

Fig. 1. The proposed architecture of anomaly detection used for real traffic signs.

Fig. 2. Preview of input and output of the proposed anomaly detection model described at section III-B1.

1) Anomaly Detection Model: Our proposed anomaly detection model is based on Autoencoder (AE) architecture. The model is trained in an unsupervised manner and does not require any anomaly samples for training [20]. The primary goal of the architecture is to learn the underlying general attributes of the normal data instance in a lower dimension and be able to reconstruct the normal instance from it [21]. The assumptions of such an architecture and training approach is that normal instances of data can be better reconstructed from the lower dimension than anomalies. Thus, this characteristic of the architecture can be also utilized for abnormal data detection considering them as failed reconstructions.

Our proposed anomaly detection is visualised in Fig. 1 and Table II shows the breakdown of the layer information.

The encoder is designed to capture the minute details of the input data instance and preserve the most essential aspect in the lower dimension. The encoder consists of several convolutional layers followed by fully connected layers (i.e., dense layers). The LeakyRELU function [22] is used as an activation function. The dropout layer is used before the fully connected layers. The dropout layers are used as a regularisation function and the primary objective is to reduce the overfitting as well as improve the overall robustness of the model. Furthermore, instead of using the Maxpool layers which are commonly employed to reduce the dimensions, we used the stride option in the convolutional layer to reduce the feature dimension based on the observation in [23]. This helped to improve the simplicity in the encoder architecture design. The encoder model receives an input image of dimension $48 \times 48 \times 3$. Total of 7 convolutional layers are used in sequential order with decreasing height and width ($H \times W$) resolution with various features sizes. In general, feature sizes are usually increased when the $H \times W$ are decreased. However, in our case, the filter size of 128 is used multiple times corresponding with the resolution downsampling. Our method forces the model to learn not only the downsampling of the height and width but also on the feature size. The aim is to capture the most important features from all dimensions. Furthermore, half of all the features are dropped using a dropout layer. The features are then flatten and fully connected layers are used to reduce features dimension to 48. These features points of the latent space are fed to the decoder. The decoder is designed to perform the heavy lifting since it needs to learn to represent the latent space data into input data instances. Hence, additional techniques such as convolutional transpose, residual block and skip connection have been used. The decoder consists of 10 layers including input and output layers (excluding the residual block). We used Convolutional transpose to upscale the features instead of simply using the Upsampling layer. Since Upsampling layer does not perform any features extraction, we choose to use convolution transpose instead of Upsampling layer. Although this has increased the complexity in the architecture, we also gain more features by using our method.

In the decoder, the residual block has been utilized to extract features useful in varying distances. The scale variety kernel utilised to handle the traffic with different sizes (i.e., even though the image size is cropped to 48×48 ($H \times W$), the traffic within the image can be small or big). Hence, the residual block is designed to be robust to such scale variants by capturing features in different scales. The residual block is repeated multi times (i.e., 4 times) and these features are passed to the following convolutional layers. The features are upsampled using a convolutional transpose layer and compressed to $48 \times 48 \times 3$ ($H \times W \times C$). Table I shows all the parameters of the encoder as well as the decoder.

- *Anomaly detection method* - When the attack detection model is fed with normal or abnormal traffic signs, the model tries to reconstruct the input images. Fig. 2 shows the reconstruction of normal images by anomaly detection model. However, since the anomaly detection models were only trained with normal data, we assumed that the model performance would degrade with abnormal data. This knowledge can be used for anomaly detection. In order to detect anomalies, we need to first establish a threshold value. This threshold value will be used to identify if the traffic sign has an anomaly or not. Hence, depending on the split ratio of normal/abnormal data in testset, one can estimate threshold by calculating error

TABLE I
OVERVIEW OF THE ANOMALY DETECTION MODEL USED FOR REAL TRAFFIC SIGNS.

Layer Type	Output Shape	Parameters
Input Layer	48 x 48 x 3	0
Encoder Layer	48	2,067,376
Decoder Layer	48 x 48 x3	6,643,383
Total Parameters		8,710,759

TABLE II
THE PROPOSED ENCODER ARCHITECTURE OF ANOMALY DETECTION USED FOR REAL TRAFFIC SIGNS.

Type	Output	Strides	Activation	Filter
Input	48x48	1	Leaky ReLU	3
Conv2D- 1	48x48	2	Leaky ReLU	64
Conv2D- 2	24x24	1	Leaky ReLU	128
Conv2D- 3	24x24	2	Leaky ReLU	256
Conv2D- 4	12x12	1	Leaky ReLU	128
Conv2D- 6	12x12	2	Leaky ReLU	512
Conv2D- 7	6x6	1	Leaky ReLU	128
Dropout(0.5)	-	-	-	-
Flatten	4608	-	-	-
Dense - 1	48	-	ReLU	-
Dense - 2	55584	-	ReLU	-
Reshape	12x12	-	ReLU	386
Conv2D-8	12x12	1	ReLU	512
Conv2D Transpose-1	12x12	2	ReLU	128
Conv2D-9	24x24	1	ReLU	256
Residual Block	24x24	1	ReLU	1 - 32
Concatenation	24x24	-	-	320
Conv2D-10	24x24	2	ReLU	256
Conv2D Transpose-2	48x48	1	ReLU	128
Conv2D-11	48x48	1	ReLU	64
Dropout(0.5)	-	-	-	-
Conv2D-12	48x48	1	Tanh	3

TABLE III
OVERVIEW OF THE RECONSTRUCTION MODEL.

Type	Output Shape	Parameters
Input Layer	$48 \times 48 \times 3$	0
Generator (Model)	$48 \times 48 \times 3$	$10,512,279$
Discriminator (Model)	1	$702,849$
Total Parameters		$11,215,128$

(e.g., mean square error) and N quantile error value which splits the data into normal and abnormal is a threshold.

2) Reconstruction Model: The reconstruction model is a GAN model which consists of generator and discriminator as shown in Fig. 3. Our proposed reconstruction model is inspired by image in-painting GAN architectures [30] where the generator is trained to generate meta-traffic signs for both normal and abnormal data instances, whereas the discriminator is trained to identify if the meta-traffic is the correct or incorrect signs. The core aim of the generator is to learn the mapping

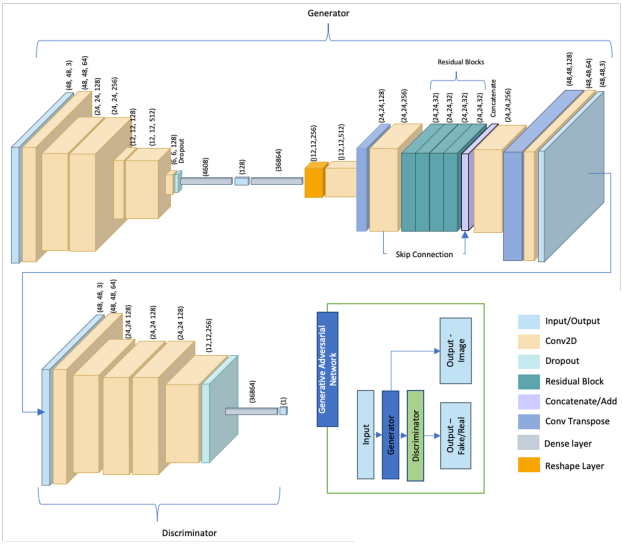

Fig. 3. The proposed architecture of reconstruction model used for real traffic signs.

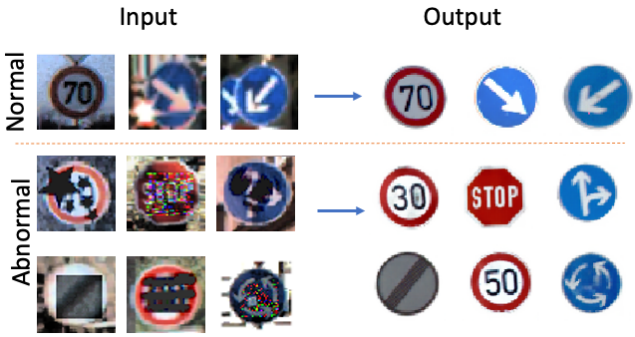

Fig. 4. The figure shows the input and out of our proposed reconstruction model. The first row shows normal traffic signs with their reconstructed meta-traffic signs. The others images are attack images and the reconstructed images.

between the real traffic signs (normal or abnormal) to the meta-traffic signs. The discriminator is trained to classify correct as well as good quality meta-traffic signs. The generator model within our proposed reconstruction model follows the similar architecture used in the anomaly detection model and only differs in the decoder section. The model not only needs to map the normal traffic signs to meta-traffic signs but also needs to understand the general features of traffic signs as well as map the abnormal traffics to the clean meta-traffic signs. Hence, additional latent space as well as filter have been applied in the generator model (see Fig. 3, and Tables III, IV).

- *Reconstruction Method* - Although the reconstruction model consists of generator and discriminator, only generator will be used for traffic sign anomaly reconstruction. The discriminator is only used during the training process. Once the anomaly detection model detects an anomaly, the traffic signs are input into the reconstruction

TABLE IV
THE PROPOSED ARCHITECTURE OF RECONSTRUCT MODEL USED FOR REAL TRAFFIC SIGNS.

Type	Output	Strides	Activation	Filter
Input	48x48	1	Leaky ReLU	3
Conv2D- 1	48x48	2	Leaky ReLU	64
Conv2D- 2	24x24	1	Leaky ReLU	128
Conv2D- 3	24x24	1	Leaky ReLU	128
Conv2D- 4	24x24	2	Leaky ReLU	256
Conv2D- 5	12x12	1	Leaky ReLU	128
Conv2D- 6	12x12	2	Leaky ReLU	512
Conv2D- 7	6x6	1	Leaky ReLU	128
Dropout(0.5)	-	-	-	-
Flatten	4608	-	-	-
Dense - 1	128	-	ReLU	-
Dense - 2	36864	-	ReLU	-
Reshape	12x12	-	ReLU	256
Conv2D-8	12x12	1	ReLU	512
Conv2D Transpose-1	12x12	2	ReLU	128
Conv2D-9	24x24	1	ReLU	256
Residual Block	24x24	1	ReLU	1 - 32
Concatenation	24x24	-	-	320
Conv2D-10	24x24	2	ReLU	256
Conv2D Transpose-2	48x48	1	ReLU	128
Conv2D-11	48x48	1	ReLU	64
Dropout(0.5)	-	-	-	-
Conv2D-12	48x48	1	Tanh	3

model (generator) to generate clean meta-traffic signs.

3) Traffic sign recognition model: Traffic sign recognition model is a widely researched field hence, we proposed a simple traffic sign recognition model in order to enhance our evaluation process for anomaly and reconstruction methods. The traffic recognition model takes in $48 \times 48 \times 3$ traffic signs and outputs the class of the traffic signs. The model is able to recognise 43 different traffic signs classes. Since the architecture design of the traffic sign recognition model is out of the scope of the paper, we only present the performance of the traffic sign recognition model.

4) Training: The datasets described in section III-A, were used for training and testing of anomaly detection, reconstruction and recognition models. The GTSRB dataset [16] was utilized to train and test the anomaly detection and recognition models. Whereas our generated dataset GTSRM was used to train to reconstruct the anomaly-free traffic signs (i.e., clean meta-traffic signs). Likewise, all the images in the dataset were pre-processed in order to fit the requirement of the proposed models. Images were resized to $48 \times 48 \times 3$ resolution, histogram equalisation (image colour balance) as well as central cropping (removes unnecessary space around traffic signs borders) and image normalisation were applied to all the traffic signs as a pre-processing task.

- *Anomaly Detection Model*: An NVIDIA GTX 2080 Ti GPU was used to train and evaluate the model, adopting the Keras-Tensorflow framework. As a gradient optimiser, we employed the Adam optimiser. We applied a variety of data augmentation schemes to the data, during the training phase. The augmentation task is performed to enhance the robustness of the attack detection and reconstruction models, via generating new varieties on the input data.
- *Reconstruction Model*: The generator and the discriminator models are jointly trained. Nevertheless, only the generator is utilized for the testing process. In the training phase, the discriminator is trained to distinguish between the generated image and the input image. The input image for the generator is the normal/abnormal image and the target is a clean meta traffic image. Again, for the training process, an NVIDIA GTX 2080 Ti was utilized along with the Keras-Tensorflow framework.
- *Recognition Model*: The traffic sign recognition model was also trained in a similar fashion to above methods. The traffic sign recognition model was trained to classify 43 different traffic sign using GTRSB dataset [16].

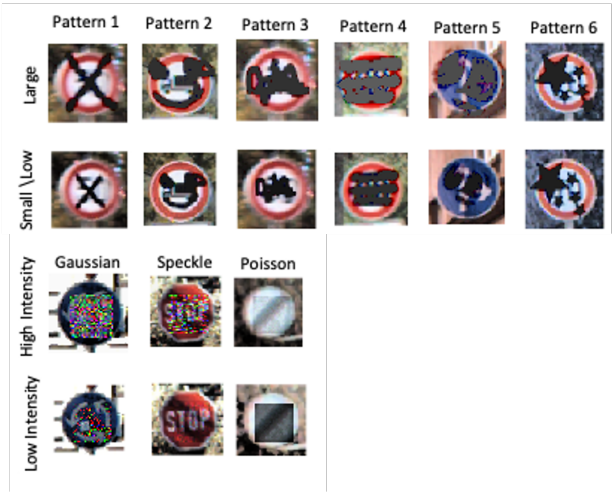

Fig. 5. Example images of different attack types.

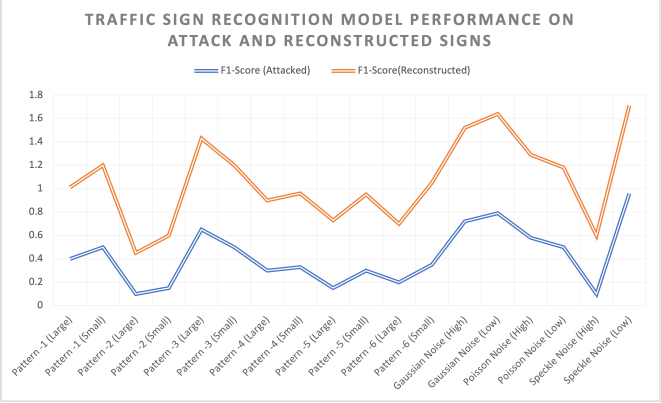

Fig. 6. Model performance with attack and with reconstruction traffic signs.

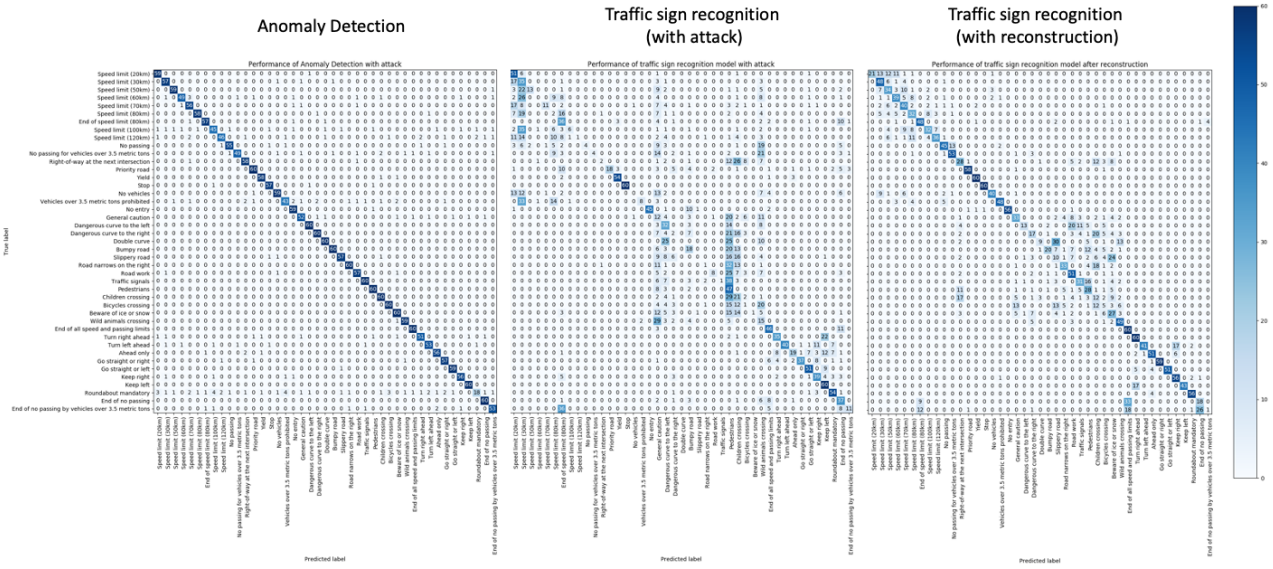

Fig. 7. Sample image of confusion matrix for anomaly detection and traffic sign recognition model. The confusion matrix is for attack types pattern - 1 (Large). The left is for anomaly detection, middle and right is for traffic sign recognition (with attack and with generated images by reconstruction model).

TABLE V
THE PERFORMANCE OF THE ANOMALY DETECTION MODEL ON VARIOUS TYPES OF ATTACKS.

Type	Precision	Recall	F1-Score	Accuracy
Patter 1 - Larger	0.8833	0.9264	0.9043	0.8955
Pattern 1 - Small	0.8790	0.8950	0.8869	0.8798
Pattern 2 - Larger	0.8540	0.7368	0.7911	0.8008
Pattern 2 - Small	0.8624	0.7581	0.8069	0.8114
Pattern 3 - Larger	0.8845	0.9128	0.8984	0.8888
Pattern 3 - Small	0.8830	0.9074	0.8950	0.8860
Pattern 4 - Larger	0.8816	0.9109	0.8960	0.8878
Pattern 4 - Small	0.8842	0.9205	0.9020	0.8926
Pattern 5 - Larger	0.8824	0.9124	0.8972	0.8886
Pattern 5 - Small	0.8799	0.9000	0.8898	0.8824
Pattern 6 - Larger	0.8763	0.8721	0.8742	0.8684
Pattern 6 - Small	0.8744	0.8674	0.8709	0.8661
Gaussian Noise - Low	0.8881	0.9233	0.9054	0.8940
Gaussian Noise - High	0.8929	0.9900	0.9389	0.9324
Poisson - Low	0.6874	0.2395	0.3553	0.5521
Poisson Noise - High	0.5106	0.1566	0.2397	0.5107
Speckle Noise - Low	0.8918	0.9891	0.9379	0.9269
Speckle Noise - High	0.8929	0.9900	0.9389	0.9324
Average Model Performance			0.824	0.844

IV. EVALUATION

Extensive experiments were performed to evaluate anomaly detection and reconstruction models. For the evaluation pur-

TABLE VI
THE PERFORMANCE OF THE TRAFFIC SIGN RECOGNITION MODEL ON NORMAL TRAFFIC SIGNS.

Type	Precision	Recall	F1-Score	Accuracy
Normal	0.978	0.99	0.983	0.99

TABLE VII
THE PERFORMANCE OF THE TRAFFIC SIGN RECOGNITION MODEL ON NORMAL TRAFFIC SIGNS.

Attack Types	F1-Score (Attacked)	F1-Score (Reconstructed)
Pattern - 1 (Large)	0.4	0.61
Pattern - 1 (Small)	0.5	0.7
Pattern - 2 (Large)	0.1	0.35
Pattern - 2 (Small)	0.15	0.45
Pattern - 3 (Large)	0.65	0.78
Pattern - 3 (Small)	0.5	0.7
Pattern - 4 (Large)	0.3	0.6
Pattern - 4 (Small)	0.33	0.63
Pattern - 5 (Large)	0.15	0.58
Pattern - 5 (Small)	0.3	0.65
Pattern - 6 (Large)	0.2	0.5
Pattern - 6 (Small)	0.35	0.7
Gaussian Noise (High)	0.72	0.8
Gaussian Noise (Low)	0.79	0.85
Poisson Noise (High)	0.58	0.71
Poisson Noise (Low)	0.5	0.68
Speckle Noise (High)	0.1	0.5
Speckle Noise (Low)	0.96	0.75
Average	**0.421**	**0.641**

pose, we generated a total of 9 different types of attacks. The first 6 were pattern attacks whereas the rest were noise based attacks. In addition, the attacks were divided into groups. For the pattern attack, we have large and small patterns whereas for the noise attacks we have high and low intensity noise attacks. Likewise, in order to perform evaluation of the attacks on traffic signs, we also trained a traffic sign recognition model.

For anomaly detection and reconstruction model performance evaluation, we prepared test sets as following: A total of 2,580 images were used from GTSRB with each class containing 60 images (i.e. 60 image image instances per 43 classes). These images are then used to create abnormal test datasets where $2,580$ images $\times 9$ types of attacks (i.e. total of 23220). Fig. 5 shows all the attack types. Based on our test setup, we calculated precision, recall, F1-score and model accuracy for the anomaly detection. The results of our experiments are shown in Table V. On average, the anomaly detection model achieved 0.824 for F1-score and 0.844 for model accuracy. Likewise, in order to evaluate the reconstruction model, we used traffic sign recognition model as our evaluation metrics. We compared the traffic sign recognition performance with the attack and with the reconstructed traffic signs. Fig. 7 shows a confusion matrix of traffic sign recognition model performance. The traffic sign recognition model performs significantly better on normal signs with F1-Score 0.983 and model accuracy 0.99 (see Table VI). As expected, performance of the model degraded significantly on anomaly traffic signs and achieved only F1-score 0.421 on average. However, with the reconstructed traffic sign, the model achieved 1.5 times better with F1-score as 0.641. Overall, all the metrics scores were improved with reconstructed meta-traffic signs as shown in Table VII and Fig. 6.

V. Conclusion

In this work, we devised and developed Vision-based Anomaly Detection and Reconstruction models to address and mitigate attacks on physical adversarial environments, specifically on traffic signs which are subject to unsecure and public locations. Anomaly detection techniques and mitigation architecture were deployed and tested against a variety of attacks, including noise and pattern attacks on traffic signs as well as adversarial attacks on Deep Learning models. For training and testing purposes, publicly available datasets were used. Furthermore, additional datasets were generated for mitigation purposes. The details on the performance of the models were presented, and the implementation of the prototype algorithms was discussed. As shown in the evaluation, the anomaly detection model achieved 0.844 accuracy with a 0.824 F1-score on the real traffic sign dataset.

In regards to future works, additional research can be performed in order to design and develop a light version of the model to be used in IoT devices. At the moment, the models have a large number of parameters, and performing well in IoT devices would be probably challenging. Additional studies can also be carried out on the effects of the residual blocks and their contribution to overall model performance.

Acknowledgment

This project has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No 101070450. Disclaimer: Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Commission. Neither the European Union nor the European Commission can be held responsible for them.

References

[1] 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 CVPR*, pp. 1625–1634, 2018.

[2] C. Yan, W. Xu, and J. Liu, "Can you trust autonomous vehicles: Contactless attacks against sensors of self-driving vehicle," p. 13, 2016.

[3] N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: A survey," vol. 6, pp. 14410–14430, 2018. Conference Name: IEEE Access.

[4] J. Lu, H. Sibai, E. Fabry, and D. Forsyth, "Standard detectors aren't (currently) fooled by physical adversarial stop signs," *arXiv preprint arXiv:1710.03337*, 2017.

[5] A. Ranjan, J. Janai, A. Geiger, and M. J. Black, "Attacking optical flow," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2404–2413, 2019.

[6] W. Liu, M. Salzmann, and P. Fua, "Using depth for pixel-wise detection of adversarial attacks in crowd counting," *arXiv preprint arXiv:1911.11484*, 2019.

[7] A. Athalye, N. Carlini, and D. Wagner, "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples," in *ICML*, pp. 274–283, PMLR, 2018.

[8] J. Fayyad, M. A. Jaradat, D. Gruyer, and H. Najjaran, "Deep learning sensor fusion for autonomous vehicle perception and localization: A review," *Sensors*, vol. 20, no. 15, p. 4220, 2020.

[9] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, "Deepdriving: Learning affordance for direct perception in autonomous driving," in *Proceedings of the IEEE ICCV*, pp. 2722–2730, 2015.

[10] L. Heng, B. Choi, Z. Cui, M. Geppert, S. Hu, B. Kuan, P. Liu, R. Nguyen, Y. C. Yeo, A. Geiger, G. H. Lee, M. Pollefeys, and T. Sattler, "Project AutoVision: Localization and 3d scene perception for an autonomous vehicle with a multi-camera system," in *2019 ICRA*, pp. 4695–4702, 2019. ISSN: 2577-087X.

[11] V. Mazzia, F. Salvetti, D. Aghi, and M. Chiaberge, "Deepway: a deep learning estimator for unmanned ground vehicle global path planning," *arXiv e-prints*, pp. arXiv–2010, 2020.

[12] F. Lambert, "Understanding the fatal tesla accident on autopilot and the nhtsa probe. 2016," 2019.

[13] N. Morgulis, A. Kreines, S. Mendelowitz, and Y. Weisglass, "Fooling a real car with adversarial traffic signs," 2019.

[14] T. K. S. Lab, "Experimental security research of tesla autopilot," 2019.

[15] S. Loveday, "Tesla model 3 traffic sign recognition tricked with fake signs in UK," 2020.

[16] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "The german traffic sign recognition benchmark: a multi-class classification competition," in *2011 international conference on NN*, pp. 1453–1460, IEEE, 2011.

[17] A. K. Boyat and B. K. Joshi, "A review paper: noise models in digital image processing," 2015.

[18] S. W. Hasinoff, "Photon, poisson noise," in *Computer Vision* (K. Ikeuchi, ed.), pp. 608–610, Springer US, 2014.

[19] P. Dar, "Datasets for deep learning open datasets," 2018.

[20] C. Zhou and R. C. Paffenroth, "Anomaly detection with robust deep autoencoders," in *23rd ACM SIGKDD*, pp. 665–674, 2017.

[21] Y. Yu, J. Long, and Z. Cai, "Network intrusion detection through stacking dilated convolutional autoencoders," vol. 2017, 2017. Hindawi.

[22] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. icml*, vol. 30, p. 3, Citeseer. Issue: 1.

[23] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," *arXiv1412.6806*, 2014.