Adversarial Patch Detection: Leveraging Depth Contrast for Enhanced Threat Visibility

1st Niklas Bunzel
Fraunhofer SIT / ATHENE
Darmstadt, Germany
bunzel@sit.fraunhofer.de 0000-0002-8921-1562

2nd Jannis Hamborg
Hochschule Darmstadt / ATHENE
Darmstadt, Germany
jannis.hamborg@h-da.de 0000-0002-8521-8230

***Abstract*—Neural networks have proven to be extremely effective at tasks such as image classification and object detection. However, their security and robustness are controversial. Even state-of-the-art object detectors can be fooled by localized patch attacks, which might lead to safety-critical incidents. In these attacks, adversaries place a subtle adversarial patch in an image, causing detectors to either miss real objects or detect phantom objects. These adversarial patches often force state-of-the-art detectors to make highly confident but incorrect predictions. The practical implications of these attacks in real-world settings further increase the concern. This paper presents a unique method for detecting real-world adversarial patches using entropy-sensitive depth estimation. Therefore, we take advantage of the fact that adversarial patches typically introduce high local entropy and are located in front of an object. We have fine-tuned a monocular depth estimation neural network to exploit these two features to extract adversarial patches from an image. Using this approach, we are able to achieve a true positive detection rate of 77.5% on the APRICOT test set.**

***Index Terms*—Adversarial Patches, Detector, Depth Estimation, Adversarial Machine Learning**

I. Introduction

Deep neural networks have led to significant advances in image recognition [1], [2], [3]. However, their use in sensitive areas such as facial recognition and autonomous driving raises security concerns due to their vulnerability to adversarial threats [4], [5], [6], [7], [8], [9]. Many of these threats focus on subtle changes, often imperceptible to humans, in the form of perturbations to the network's input to alter its predictions. Such perturbations can be implemented as small spots in the image, known as patches, or as noise patterns that overlay the entire image. The concept of adversarial patches, which alter pixels within limited regions to fool object detectors [10], [11], [2], [1] or misclassify images by classifiers [12], has emerged as a notable problem [9], [13], [14], [15]. Although numerous countermeasures have been presented [16], [17], [18], [19], [20], [21], challenges remain in effectively detecting or neutralizing these adversarial patch attacks. In this paper, we address the identification and mitigation of adversarial patches by combining enhanced monocular depth estimation with image inpainting, specifically targeting high entropic adversarial patches typically placed in the foreground of objects. Our approach begins with a monocular depth estimation network, which is fine-tuned to better discern high-entropy regions, thus isolating potential adversarial patches. Once identified, these patches are mitigated through inpainting methods, improving the robustness of the object detector against such attacks.

II. Related Work

a) Object Detection: Object detection is used to locate and classify objects in an image. It has practical importance in areas such as face detection [3] and autonomous driving, including the detection of traffic signs or vehicles [22], [23]. Multi stage object detectors operate through a sequence of distinct phases to identify and classify objects within images. Initially, they generate a set of region proposals where objects are likely to be found, typically through a separate network or algorithm designed to highlight areas of interest. These proposals are then passed through a second stage, where a more detailed analysis is conducted to refine the bounding boxes and classify the contents of each proposed region. The R-CNN [10] family is an example of multi stage object detectors. They work by extracting about 2000 region proposals from an input image using selective search algorithms. These proposals are then processed by a large convolutional neural network (CNN) and classified using pre-trained SVMs. Fast R-CNN [11], as an advancement, provides a faster and more accurate approach. It integrates a unique layer known as ROI pooling to infer features, distributing the computation across all proposals. For real-time processing capabilities, Faster R-CNN [2] uses a region proposal network (RPN) that works with the convolutional layers of Fast R-CNN.
Single-stage object detectors streamline the detection process by eliminating the need for separate region proposal generation, directly predicting object classes and bounding box coordinates in one go from the full image. Examples are models like YOLO (You Only Look Once) [1], [24], SSD (Single Shot Multibox Detector) [25] or DETR (Detection Transformer) [26].
b) Depth Estimation: Estimating depth from a single image is an inherently difficult problem. The same input image can be projected to multiple plausible depths. Depth estimation, has seen significant advancements due to deep learning and neural networks. The state of the art in this domain reflects these technological strides, particularly in the way depth information is extracted from visual data. Key contributions include [27],

2325-6664/24/$31.00 ©2024 IEEE
DOI 10.1109/DSN-W60302.2024.00020

which introduced a novel ordinal regression framework to the field, significantly improving the accuracy of depth predictions from single images. Another notable work is [28], which pioneered a self-supervised learning approach using stereo pairs or monocular sequences, thereby circumventing the need for depth annotations. Additionally [29] leveraged transfer learning from pre-trained models to enhance the depth estimation quality further. More recently, [30] introduced an approach focusing on multi-scale local planar guidance, offering a significant boost in handling the scale ambiguity inherent in monocular depth estimation. More recent approaches, such as DPT [31] and DPT-DinoV2 [32] utilizing vision transformers.

c) Adversarial Patch Attacks: An adversarial patch attack introduces a perturbation to a specific region of an image with the intention of misleading image classifiers [12] or manipulating the results of object detectors [1], [10], [11], [2]. These patches can be universal, robust, and designed with specific targets in mind [9], making them viable for real-world scenarios. Image classification often depends on the most dominant element within the frame. By creating a patch that is more salient than any real-world object, adversarial patches can fool this classification process [9]. While these patches are typically visible to the human eye, they can easily fool machine learning models. The method proposed by [9] focuses on fooling image classifiers, but falls short against object detectors. Modern object detectors, such as Faster R-CNN [2], can identify multiple objects in a single image and mark them with bounding boxes. Liu et al. introduced an attack called DPatch [13] that simultaneously interferes with bounding box regression and object classification. Due to its compact nature, DPatch is highly effective for both targeted and untargeted real-world attacks.

d) Adversarial Patch Detection: Several defense strategies have been developed to identify adversarial patches [33], [17], [16]. Liu et al. introduced the "Segment and Complete" (SAC) defense [34], a holistic framework that uses a patch segmentation model to detect adversarial patches at the pixel level. Subsequently, a robust shape completion algorithm removes the adversarial patch from the image. Chou et al. propose a framework called SentiNet [35], similar to the signature-based detection described in [16]. Here, the regions that most influence the prediction are suspected to host the adversarial patch. Within SentiNet, this influence is measured using the results of a Grad-CAM analysis [36]. In addition, Chen et al. [18] presented an alternative defense against universal adversarial patches. Using tools such as [37], [38], [36], they identified critical image features that affect the final judgment and integrated them into a neutral image. A shift in the prediction of this benign image typically indicates the presence of an adversarial patch. In [20] they exploit the fact that adversarial patches are highly entropic regions and detect them as an anomalous clustering of edges. [39] improves this approach by using ELA and GLCM to detect patch candidates.

e) Error Level Analysis: Error Level Analysis (ELA) can detect areas in a JPEG image that may have been altered by various forms of image manipulation, such as splicing attacks. This leads to discrepancies in the compression history between genuine and altered areas. This is done by recompressing the original image with various quality factors and then determining the difference between these recompressed images and the original image. This process reveals any anomalies in the compression history once the correct quality factor for compressing either the authentic or the tampered region is identified.

III. Approach

A. The Detection and Mitigation Pipeline

To identify adversarial patch candidates and mitigate them, we implement a pipeline based on monocular depth estimation and image inpainting. Adversarial patches are usually placed in the foreground of the object. Traditional depth estimation techniques are often unable to detect these manipulated areas accurately due to the minimal physical separation between the patch and the background. Recognizing this challenge, we approach this as follows:

1) *Enhanced Depth Estimation:* The first step in our pipeline is to utilize a monocular depth estimation network, which, while not sufficiently accurate on its own, serves as a foundation for further refinement. In this paper, we use the MiDaS network [31], [40] as a proof of concept. The core of our methodology is the use of a fine-tuned depth estimation network. This network is fine-tuned with a focus on mixing depth data with high entropy regions.
2) *Depth Difference Analysis:* We compute the difference between the initial depth estimation and the result provided by the fine-tuned network to increase the contrast between benign and malicious segments of the image. This difference in depth estimation highlights the discrepancies attributed to adversarial patches. By binarizing the resulting depth difference image, we generate candidates for adversarial patches, effectively isolating these regions for further action.
3) *Patch Mitigation and Detection:* Once potentially adversarial patches are identified, we proceed with mitigation through inpainting techniques, either diffusion-based [41] or signal-based [42]. For this proof of concept, we used the latter, with an implementation from OpenCV as in [20], [39]. An example can be seen in Figure 1. Inpainting allows us to reconstruct the affected areas with high fidelity, erasing the traces of the adversarial patch. For the detection phase, we adopt a comparative analysis strategy inspired by [20], where both the original and the inpainted images are evaluated by the object detection network. A discrepancy in the network's predictions signals the presence of an adversarial patch, confirming its prior existence and the effectiveness of the mitigation process. Contrary to the

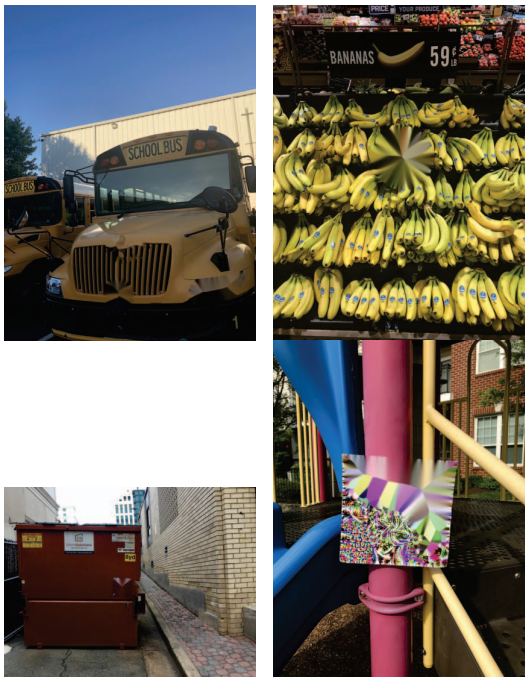


Fig. 1: Examples of adversarial patch mitigations using our approach.

approach described by [20], our method produces a single candidate mask for mitigation, significantly simplifying the mitigation process by reducing the number of required actions from N to 1. This also lowers the computational effort needed for subsequent comparisons in the detection phase.

B. Network Fine-tuning

For fine-tuning the depth estimation model, a subset of the APRICOT [43] development set was utilized. We used an Error Level Analysis as proposed in [39]. We implemented it to generate a grayscale image that approximates the entropy levels, as can be seen in Figure 2. We use a pre-trained MiDaS large DPT [31] to generate the initial depth estimates. We then weight the depth estimates by the ELA images.
This weighted depth image, representing our expected output, was then used to fine-tune the MiDaS depth estimation network. We implemented different weighting functions - namely, addition, maximum, multiplication. The output of the MiDaS network fine-tuned with the different weighting methods can be seen in Figure 3.
Our chosen loss function was the Mean Squared Error (MSE), and we used the Stochastic Gradient Descent (SGD) optimizer to train the network. The learning rate was set to 0.01, along with a momentum of 0.5. We fine-tuned the model for 100 epochs, where each epoch processed a batch of 64 images.

(a) Original (b) ELA

Fig. 2: Adversarial patch and ELA variant.

IV. Evaluation

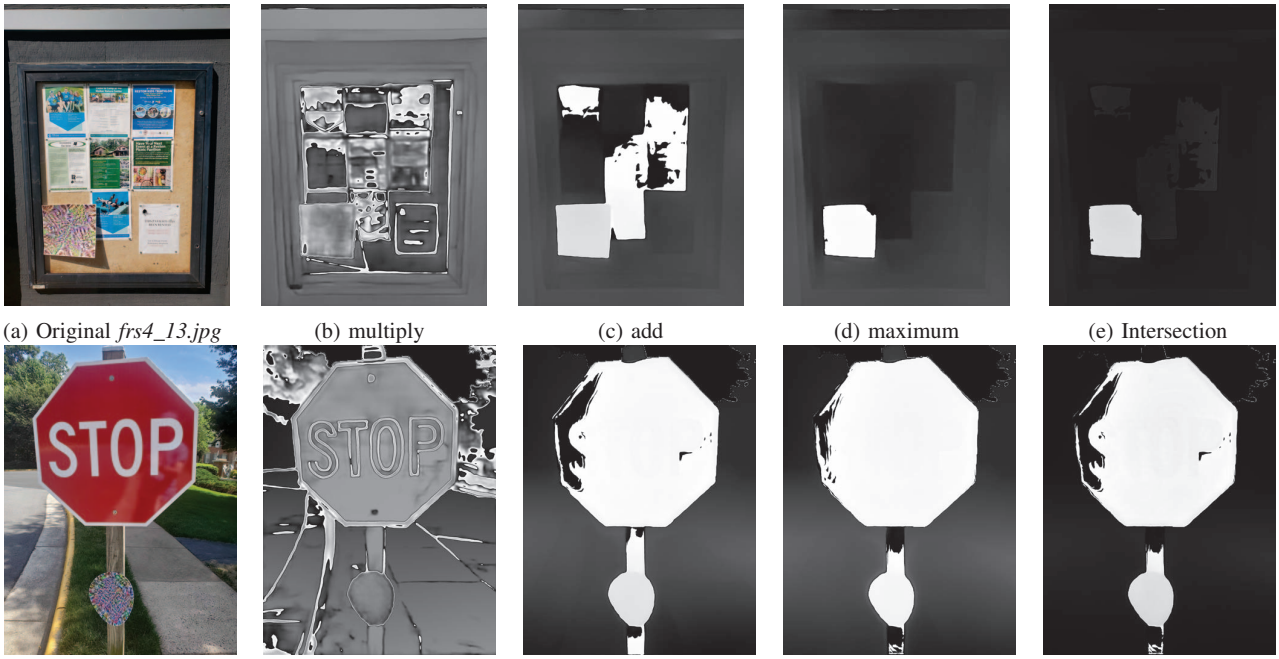
In the following, we first evaluate the effects of the different weighting mechanisms on patch detection performance. Then we evaluate the pixelwise patch candidate identification, for the pixelwise evaluation we use the mask annotations from [34]. To conclude we evaluate the whole mitigation and detection pipeline and compare it with state of the art methods. For evaluation on real-world adversarial patch images we use the APRICOT [43] dataset and for benign images we use the COCO [44] dataset.

A. Weighting Functions

In Figure 3 it can be seen that in the difference image in add- and max-weighting mode, the patches are displayed very contrastly. This means that the respective neural network produces a large difference from the original depth image. Patches clearly stand out from the surroundings. The patch detection is thus successful. The add- and max-weighting mode produce different amounts of artifacts. However, this is independent of the entropy of the respective image. In both modes, the patches are clearly highlighted. To improve the detection performance, the intersection of the difference images of the add- and max-weighting mode can be formed. However, this strongly increases the required computational power, since all depth estimates have to be computed twice. The multiply-weighting mode results in lower contrast difference images; this means that there is a smaller difference to the original depth image. Patches blur with the surroundings. Patch detection is therefore not successful.

B. Pixelwise Patch Identification

To identify the masks of the patch candidates, we need to binarize the depth difference image, as described in step 2 of our approach. In the depth difference images of the APRICOT [43] development set, the patches typically appear in shades ranging from light gray to nearly white, prompting our evaluation of specific threshold values. For our evaluation,



(a) Original *frs4_13.jpg* (b) multiply (c) add (d) maximum (e) Intersection

(f) Original *rrc3_2.jpg* (g) multiply (h) add (i) maximum (j) Intersection

Fig. 3: Qualitative comparison of weighting modes (*add, multiply, maximum*) and the intersection of *add* and *multiply*.

we set the pixel value of 205 as the lower bound and 240 as the upper bound because preliminary observations on the development set indicated that these values were promising; however, it should be noted that we did not further optimize these thresholds for the results dependent on the different weighting functions. We investigated the use of the lower bound alone, which results in an increase in the detection of false positive pixels, and the combination of both lower and upper bounds, which results in an increase in the detection of false negative pixels.

When applying a lower bound only, the maximum weighting algorithm emerged as the most effective, yielding a high true positive (TP) rate of 84.69% and a true negative (TN) rate of 75.41%, as depicted in Figure 4. The addition weighting also showed a relatively strong performance with a TP of 75.44% and a TN of 69.86%, but at the expense of increased FP and FN rates.

Utilizing both lower and upper bounds, the addition weighting algorithm outperformed the others, achieving a TP rate of 61.99% and a TN rate of 93.83%. Notably, the maximum approach, while lowering the TP rate to 52.81%, significantly excelled in TN accuracy, reaching 96.12% as can be seen in Figure 5.

The multiplication and intersection algorithms, regardless of the thresholding strategy, exhibited challenges, especially in maintaining high TP rates.

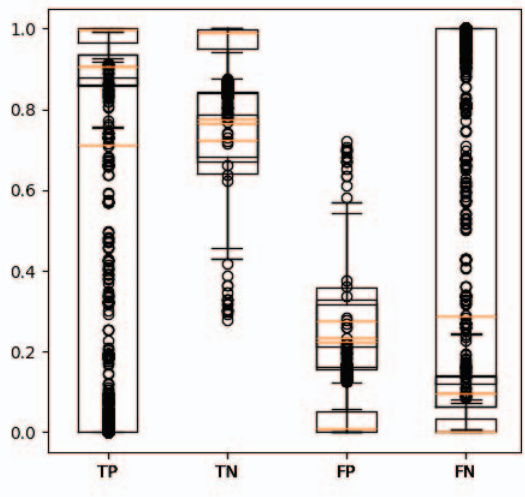


Fig. 4: Pixelwise metrics, with maximum weighting mode, lower bound thresholding binarization.

C. Patch Mitigation & Detection

The 3rd and last phase of our patch identification pipeline focuses on the mitigation and detection of adversarial patches. We examined the efficacy of our approach in this regard, particularly evaluating how the different weighting strategies

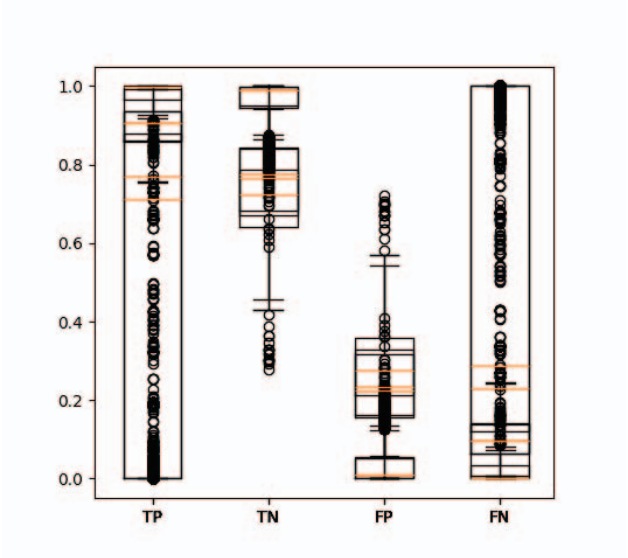


Fig. 5: Pixelwise metrics, with maximum weighting mode, lower bound and upper bound thresholding binarization.

influence the final results in conjunction with the lower and upper bound binarization. We therefore used the Faster R-CNN [2] object detector in conjunction with the subset of the APRICOT test set that successfully fools this detector.

The maximum weighting approach demonstrated a true positive rate of 67.57% and a true negative rate of 42.64%. This suggests a reasonable ability to identify true patches, though it shows issues in confirming the absence of a patch.

In the case of the addition weighting there was a notable increase in TPR to 77.48%, indicating a stronger proficiency in detecting true patches. However, this came at the expense of TNR, which dropped to 30.2%, highlighting a higher rate of false alarm where non-adversarial regions may be mistaken for patches.

The intersection approach, which combines maximum and addition weightings achieved a TPR of 72.07%, the TNR was the lowest among our methods, at 20.38%. Indicating a significant tendency to misidentify non-patched areas as adversarial.

The challenge with benign samples arises from the presence of regions in the foreground that exhibit high entropy. These areas are frequently flagged by our detection method. Consequently, when inpainting is applied to these regions, it can alter the predictions made by the object detection algorithm. Compared to other state-of-the-art methods, our approach is more effective in detecting adversarial patches as can be seen in Table I. The Edge Detector method [20] presented a higher TNR, surpassing our maximum weighting approach. The JPEG Detector showed a much lower performance than the rates achieved by our methods.

Method	Classification	Adversarial	Benign
JPEG Detector	Adversarial	35%	65%
	Benign	67%	33%
Edge Detector	Adversarial	72%	28%
	Benign	44%	56%
Depth Detector$_{\text{Add}}$ (ours)	Adversarial	**77.48%**	22.52%
	Benign	69.8%%	30.2%

TABLE I: Detection performance of our Depth-based detector in comparison with an edge detection-based detector and a JPEG Detector against adversarial patch images from APRICOT and benign images from COCO.

V. Conclusion & Future Work

In this paper, we present a novel approach to adversarial patch detection that exploits the capabilities of a fine-tuned depth estimation model. By merging signal based adversarial patch detection techniques, namely ELA, with depth estimation. The results underscore the potential of depth-aware models to improve the robustness and resilience of systems against adversarial attacks. Our approach reaches a true positive detection of 77.48%. A remaining issue are benign foreground regions of high entropy that are identified as potential adversarial patches and, through mitigation artifacts, change the label and produce a false positive detection.

In the future we want to examine gray level co-occurrence matrix (GLCM) as it provides insights into the textural patterns and variations within an image by revealing the spatial relationships between pixels. For real-world patches this should perform better than ELA in identifying areas with potential adversarial patches, allowing for a more effective weighting and fine-tuning of the depth estimation network. We plan to optimize the threshold settings depending on the used weighting function to achieve a more refined balance, enhancing the true positive and true negative rates. By fine-tuning these thresholds, we aim to improve the model's precision in distinguishing between adversarial and legitimate content, thereby boosting overall detection performance. The issue of false positve detections may be reduced by performing the inpainting with state of the art diffusion based methods.

Acknowledgment

This research work has been partly funded by the German Federal Ministry of Education and Research and the Hessian Ministry of Higher Education, Research, Science and the Arts within their joint support of the National Research Center for Applied Cybersecurity ATHENE and the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - project number 503329135.

References

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," 2015. [Online]. Available: https://arxiv.org/abs/1506.02640

[2] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," 2015. [Online]. Available: https://arxiv.org/abs/1506.01497

[3] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," 2018. [Online]. Available: https://arxiv.org/abs/1801.07698

[4] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," 2014. [Online]. Available: https://arxiv.org/abs/1412.6572

[5] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," 2017. [Online]. Available: https://arxiv.org/abs/1706.06083

[6] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," 2016. [Online]. Available: https://arxiv.org/abs/1608.04644

[7] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: a simple and accurate method to fool deep neural networks," 2015. [Online]. Available: https://arxiv.org/abs/1511.04599

[8] N. Bunzel and L. Graner, "A concise analysis of pasting attacks and their impact on image classification," in *53rd Annual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2023 - Workshops, Porto, Portugal, June 27-30, 2023*. IEEE, 2023, pp. 136–140.

[9] T. B. Brown, D. Mané, A. Roy, M. Abadi, and J. Gilmer, "Adversarial patch," *CoRR*, vol. abs/1712.09665, 2017. [Online]. Available: http://arxiv.org/abs/1712.09665

[10] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," 2013. [Online]. Available: https://arxiv.org/abs/1311.2524

[11] R. Girshick, "Fast r-cnn," 2015. [Online]. Available: https://arxiv.org/abs/1504.08083

[12] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014. [Online]. Available: https://arxiv.org/abs/1409.1556

[13] X. Liu, H. Yang, Z. Liu, L. Song, H. Li, and Y. Chen, "Dpatch: An adversarial patch attack on object detectors," 2018. [Online]. Available: https://arxiv.org/abs/1806.02299

[14] S. Thys, W. Van Ranst, and T. Goedemé, "Fooling automated surveillance cameras: adversarial patches to attack person detection," 2019. [Online]. Available: https://arxiv.org/abs/1904.08653

[15] S. Komkov and A. Petiushko, "AdvHat: Real-world adversarial attack on ArcFace face ID system," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, jan 2021. [Online]. Available: https://doi.org/10.1109%2Ficpr48806.2021.9412236

[16] B. Liang, J. Li, and J. Huang, "We can always catch you: Detecting adversarial patched objects with or without signature," 2021. [Online]. Available: https://arxiv.org/abs/2106.05261

[17] N. Ji, Y. Feng, H. Xie, X. Xiang, and N. Liu, "Adversarial yolo: Defense human detection patch attacks via detecting adversarial patches," 2021. [Online]. Available: https://arxiv.org/abs/2103.08860

[18] Z. Chen, P. Dash, and K. Pattabiraman, "Turning your strength against you: Detecting and mitigating robust and universal adversarial patch attacks," 2021.

[19] O. Knagg. Know your enemy. [Online]. Available: https://towardsdatascience.com/know-your-enemy-7f7c5038bdf3

[20] N. Bunzel, A. Siwakoti, and G. Klause, "Adversarial patch detection and mitigation by detecting high entropy regions," in *53rd Annual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2023 - Workshops, Porto, Portugal, June 27-30, 2023*. IEEE, 2023, pp. 124–128.

[21] C. Xiang and P. Mittal, "Patchguard++: Efficient provable attack detection against adversarial patches," 2021. [Online]. Available: https://arxiv.org/abs/2104.12609

[22] P. S. Zaki, M. M. William, B. K. Soliman, K. G. Alexsan, K. Khalil, and M. El-Moursy, "Traffic signs detection and recognition system using deep learning," 2020. [Online]. Available: https://arxiv.org/abs/2003.03256

[23] R. Chandrika, N. Ganesh, A. Mummoorthy, and K. Raghunath, "Vehicle detection and classification using image processing," in *2019 International Conference on Emerging Trends in Science and Engineering (ICESE)*, vol. 1, 2019, pp. 1–6.

[24] Ultralytics, "Yolov8," 2023, [Online; accessed 05-1-2024]. [Online]. Available: https://github.com/ultralytics/ultralytics

[25] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C. Fu, and A. C. Berg, "SSD: single shot multibox detector," in *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part I*, ser. Lecture Notes in Computer Science, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., vol. 9905. Springer, 2016, pp. 21–37. [Online]. Available: https://doi.org/10.1007/978-3-319-46448-0_2

[26] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part I*, ser. Lecture Notes in Computer Science, A. Vedaldi, H. Bischof, T. Brox, and J. Frahm, Eds., vol. 12346. Springer, 2020, pp. 213–229. [Online]. Available: https://doi.org/10.1007/978-3-030-58452-8_13

[27] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, "Deep ordinal regression network for monocular depth estimation," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 2002–2011.

[28] C. Godard, O. M. Aodha, M. Firman, and G. J. Brostow, "Digging into self-supervised monocular depth estimation," in *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE, 2019, pp. 3827–3837.

[29] I. Alhashim and P. Wonka, "High quality monocular depth estimation via transfer learning," 2019.

[30] J. H. Lee, M.-K. Han, D. W. Ko, and I. H. Suh, "From big to small: Multi-scale local planar guidance for monocular depth estimation," 2021.

[31] R. Ranftl, A. Bochkovskiy, and V. Koltun, "Vision transformers for dense prediction," in *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*. IEEE, 2021, pp. 12 159–12 168.

[32] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski, "Dinov2: Learning robust visual features without supervision," 2023.

[33] C. Xiang and P. Mittal, "Detectorguard: Provably securing object detectors against localized patch hiding attacks," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 3177–3196. [Online]. Available: https://doi.org/10.1145/3460120.3484757

[34] J. Liu, A. Levine, C. P. Lau, R. Chellappa, and S. Feizi, "Segment and complete: Defending object detectors against adversarial patch attacks with robust patch detection," *CoRR*, vol. abs/2112.04532, 2021. [Online]. Available: https://arxiv.org/abs/2112.04532

[35] E. Chou, F. Tramèr, and G. Pellegrino, "Sentinet: Detecting localized universal attacks against deep learning systems," in *2020 IEEE Security and Privacy Workshops (SPW)*, 2020, pp. 48–54.

[36] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," *International Journal of Computer Vision*, vol. 128, no. 2, pp. 336–359, oct 2019. [Online]. Available: https://doi.org/10.1007%2Fs11263-019-01228-7

[37] D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg, "Smoothgrad: removing noise by adding noise," 2017.

[38] M. Ancona, E. Ceolini, C. Öztireli, and M. Gross, "Towards better understanding of gradient-based attribution methods for deep neural networks," 2018.

[39] N. Bunzel, R. A. Frick, G. Klause, A. Schwarte, and J. Honermann, "Signals are all you need: Detecting and mitigating digital and real-world adversarial patches using signal-based features," in *Proceedings of the 2024 Secure and Trustworthy Deep Learning Systems Workshop*, ser. SecTL '24. New York, NY, USA: Association for Computing Machinery, 2024.

[40] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun, "Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 3, pp. 1623–1637, 2022. [Online]. Available: https://doi.org/10.1109/TPAMI.2020.3019967

[41] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," 2022.

[42] A. Telea, "An image inpainting technique based on the fast marching method," *Journal of graphics tools*, vol. 9, no. 1, pp. 23–34, 2004.

[43] A. Braunegg, A. Chakraborty, M. Krumdick, N. Lape, S. Leary, K. Manville, E. Merkhofer, L. Strickhart, and M. Walmer, "Apricot: A dataset of physical adversarial attacks on object detection," 2019. [Online]. Available: https://arxiv.org/abs/1912.08166

[44] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, "Microsoft coco: Common objects in context," 2014, cite arxiv:1405.0312Comment: 1) updated annotation pipeline description and figures; 2) added new section describing datasets splits; 3) updated author list. [Online]. Available: http://arxiv.org/abs/1405.0312