Adversarial Object-Aware Steering Attacks for Autonomous Driving Systems

Min Jeon, Hyun Jun Yook, Bo Woon Jeong, So Hyun Kang, and Youn Kyu Lee*
Department of Computer Science and Engineering
Chung-Ang University
Seoul, Republic of Korea
{mulsoap0504, hyunjun6, bowoon, thgus4734, younkyul}@cau.ac.kr

***Abstract*—Recent advances in deep learning have promoted autonomous driving, but adversarial attack methods on autonomous driving systems revealed safety vulnerabilities. However, existing studies have focused solely on manipulating vehicle control (e.g., steering) without explicit attack targets, limiting the effectiveness of attacks in real-world driving environments. To address this limitation, we propose a novel adversarial attack method against autonomous driving that can induce collisions with specific target objects. The proposed method generates adversarial perturbations that reflect the relative position of the target object during autonomous vehicle motion. The generated adversarial perturbations are injected into the autonomous vehicle on a frame-by-frame basis, enabling the vehicle to collide with the target object. Experimental results on the Udacity's self-driving car simulator show that the proposed method induces collisions with high success rates. Furthermore, we demonstrate that the proposed method can precisely guide an autonomous vehicle toward target objects to induce collisions.**

***Index Terms*—adversarial attack, autonomous driving, vehicle collision, object detection**

I. Introduction

An autonomous driving system enables a vehicle to perceive its environment and assess potential hazards using data collected from various sensors (e.g., camera images, distance-sensing data) without human intervention [1]. However, fatal accidents involving autonomous vehicles continue to occur. Recent studies have showed that safety vulnerabilities to adversarial attacks can be exploited to cause traffic sign misrecognition [2], [3].

To identify the adversarial vulnerabilities of autonomous driving systems, various attack methods have been studied. A representative method is the patch-based attack, which generates adversarial perturbations in the form of physical patches to manipulate vehicle steering of autonomous driving systems [4]. However, its effectiveness is limited due to the human-perceptible nature of the patches. To address this limitation, gradient-based attacks have been proposed, operating under scenarios where the attacker has access to the vehicle's sensor data [5]. These methods add imperceptible adversarial perturbations into the input images of the autonomous driving system. However, they are generally limited to relatively low-level manipulations, such as lane departure. Moreover, the lack of explicit attack objectives—such as inducing collisions with target objects (e.g., cars or pedestrians)—further limits their effectiveness. Attacking an autonomous vehicle with explicit attack objectives requires both identifying the target object and continuously tracking its position. However, as the vehicle moves, the relative position between the target object and the autonomous vehicle changes in real-time. Consequently, inducing a precise collision remains challenging, even when the target object is stationary.

Therefore, we propose a novel adversarial attack method against autonomous driving systems that can induce collisions with target objects. The proposed method precisely controls a victim vehicle with an autonomous driving system to maliciously collide with the target object. Specifically, the proposed method utilizes object detection to identify the position of the target object in real-time and generates perturbations that manipulate the victim vehicle's steering to direct it toward the target's center. These perturbations are added frame-by-frame into the victim vehicle's front camera images. This process continuously reflects the relative position of the target object during victim vehicle motion, thereby enabling precise collision induction.

We evaluated the effectiveness of the proposed method using Udacity's self-driving car simulator, which supports various driving scenarios. The results show that the proposed method successfully induced collisions with various target objects at high success rates. Furthermore, the quantitative analysis demonstrated that it can precisely guide an autonomous vehicle into target objects to induce collisions.

The main contributions of this paper are as follows:

1) Proposing a novel adversarial attack method for autonomous driving systems that can induce collisions with target objects at a high success rate.
2) Designing an adversarial perturbation generation mechanism that enables precise collisions by continuously reflecting the target object's position.
3) Validating the effectiveness of the proposed method through Udacity's self-driving car simulator under various driving scenarios.

This paper is organized as follows. Section II reviews related work. Section III explains the proposed method and Section IV presents the evaluation. Finally, Section V concludes the paper.

*Corresponding Author

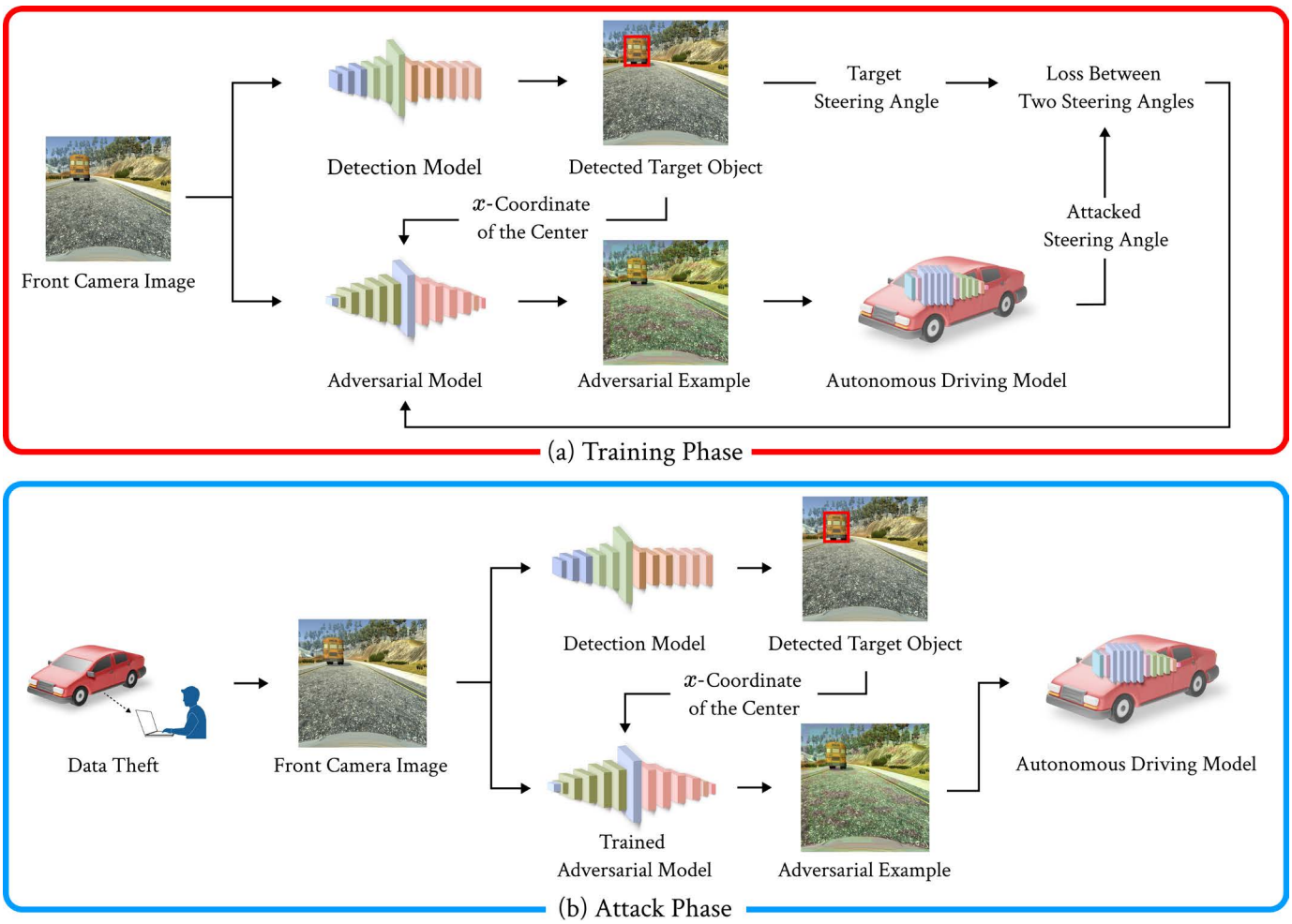

Fig. 1. Overview of the proposed method: (a) the training phase and (b) the attack phase.

II. Related Work

A. End-to-end Autonomous Driving Model

An autonomous driving system enables a vehicle to perceive its environment through various sensors and make driving decisions without human intervention [6]. While conventional approaches employ a hierarchical architecture composed of detailed modules for planning, control, and perception, recent end-to-end learning models (i.e., e2e models) integrate these modules into a single model. These models directly output control signals for planning autonomous driving from sensor data. This approach can reduce redundant computations and unnecessary information transfer between modules, thereby improving the efficiency and response speed of the system [6], [7]. However, since the e2e model integrates the input–output process into a single model, subtle perturbations applied to the input data can directly influence the vehicle control parameters—such as steering or throttle—without any separate filtering or verification process. Due to this structural characteristic, e2e models are considered inherently more vulnerable to adversarial attacks [8].

B. Adversarial Attacks on Autonomous Driving Model

An adversarial attack refers to a technique that induces a deep learning model to generate incorrect outputs by adding imperceptible perturbations to raw data [8]–[10]. Fast Gradient Sign Method (FGSM) generates adversarial perturbations by computing the sign of the loss function's gradient so that the loss increases [11]. Projected Gradient Descent (PGD) adds random perturbations to the input image and then generates adversarial perturbations through iterative updates conducted over multiple steps, each starting from different initial states [12]. Generative Adversarial Network-based attacks (GAN-based attacks) train a generator to produce realistic fake data that can fool the discriminator, thereby generating adversarial examples [13].

Recently, various methods that target autonomous driving models have been proposed. Wu et al. [5] proposed two types of adversarial attack methods (i.e., image-specific attack and image-agnostic attack) against a regression-based e2e model on the Udacity's self-driving car simulator. These methods are designed to guide the vehicle's steering angle into a specific direction. The image-specific attack generates adversarial per-

turbations individually calculated for each front-view image to manipulate the vehicle's steering. The image-agnostic attack uses a universal adversarial perturbation for all input images to manipulate the steering. Zhang et al. [14] proposed an image-agnostic white-box adversarial attack against an e2e model, employing a multi-objective optimization function and an adaptive weighting scheme to simultaneously manipulate the vehicle's steering angle and speed. Existing methods have mainly focused on manipulating vehicle control parameters, such as steering angle or throttle. However, methods targeting complex scenarios—such as dynamically adjusting the degree of steering manipulation in real-time or inducing collisions with specific targets—have been relatively limited. To build safer autonomous driving systems, it is required to develop more practical adversarial attack methods that can induce vehicle accidents in complex driving environments, such as by inducing collisions with target objects.

III. Proposed Method

In this paper, we propose a novel adversarial attack method that induces collisions with target objects by manipulating the autonomous driving model of a victim vehicle. As shown in Fig. 1, the proposed method consists of three components: a Detection Model, an Adversarial Model, and an Autonomous Driving Model. The Detection Model identifies the position of an attacker-defined target object, while the Adversarial Model manipulates the autonomous driving model by injecting perturbations. Specifically, the proposed method operates in two phases: training phase and attack phase. In the training phase, the Adversarial Model is trained to generate adversarial perturbations that induce collisions, using the detected target object's position as input. In the attack phase, front camera images from the autonomous vehicle are intercepted via a Man-in-the-Middle (MITM) attack and fed into the trained Adversarial Model, which generates adversarial examples by adding adversarial perturbations into the images. The generated adversarial examples can manipulate the autonomous driving model, enabling it to control the victim vehicle's steering toward the target object to induce a collision. The detailed process for each phase is described as follows.

A. Training Phase

In the training phase, the Adversarial Model is trained to generate adversarial perturbations based on the target object's position. As shown in Fig. 1(a), the Detection Model identifies the target object from the front camera image of the victim vehicle and extracts its center x-coordinate. Subsequently, the adversarial perturbation δ, generated by the Adversarial Model $G(\cdot)$ based on the extracted x-coordinate, is defined as follows,

$$\delta = G(t_x,\ i) \tag{1}$$

In Eq. 1, t_x denotes the central x-coordinate of the target object detected by the Detection Model, and i represents the front camera image. The perturbation δ is added to i at the pixel-level to generate the adversarial example i_{adv}. To ensure valid pixel intensities, values of i_{adv} are clipped to the RGB range [0, 255]. The attacked steering angle $\hat{y}$, predicted by the autonomous driving model $f(\cdot)$ based on the adversarial input i_{adv}, is defined as follows,

$$\hat{y} = f(i_{adv}) \tag{2}$$

To optimize $\hat{y}$ toward the target object position t_x, the target steering angle y is defined as follows,

$$y = \frac{t_x - 0.5W}{0.5W} \tag{3}$$

In Eq. 3, W denotes the width of i. The steering angle of f is normalized to the range [-1, 1], where the values of -1, 0, and 1 correspond to steering toward the left, center, and right of i, respectively. This normalization corresponds to the maximum steering range of the vehicle. To present the target object position t_x as the target steering angle, its relative position is first computed as the difference between t_x and the i's center x-coordinate $0.5W$. This difference is then divided by $0.5W$ to normalize it into the range [-1, 1], enabling the relative position of the target object to be consistently mapped to the steering angle range of f, regardless of W. The mean squared error loss $\mathcal{L}_{MSE}$ for training G is defined as follows,

$$\mathcal{L}_{MSE} = ||\hat{y} - y||_2^2 \tag{4}$$

Finally, by training G to minimize $\mathcal{L}_{MSE}$, the generator G is optimized to produce perturbation that manipulates the victim vehicle's steering toward the target object.

B. Attack Phase

In the Attack Phase, the camera images of the victim vehicle are intercepted, and the trained adversarial model G is employed to perform adversarial attacks on the autonomous driving system. First, the attacker defines the target object (i.e., human, car, bus, and traffic light) intended to induce a collision with the moving victim vehicle. As shown in Fig. 1(b), the Detection Model determines whether the defined target object appears in the intercepted image i, and computes its position to derive t_x. Then, G utilizes t_x and i as inputs and generates δ that manipulates the prediction of f toward y. The generated δ is added to i at the pixel level to produce the adversarial example i_{adv}. The adversarial example i_{adv} generated through this process is injected into the victim vehicle under the MITM attack scenario by replacing the original input i. This process is repeated frame-by-frame, causing f to predict $\hat{y}$ in real-time based on i_{adv}. Consequently, the victim vehicle is steered toward the target object according to $\hat{y}$, ultimately leading to a collision.

IV. Evaluation

We evaluate the effectiveness of our proposed method by addressing the following research questions:

- **RQ#1**: Does the proposed method effectively induce collisions with target objects?
- **RQ#2**: Does the proposed method precisely control vehicle steering toward the target object?

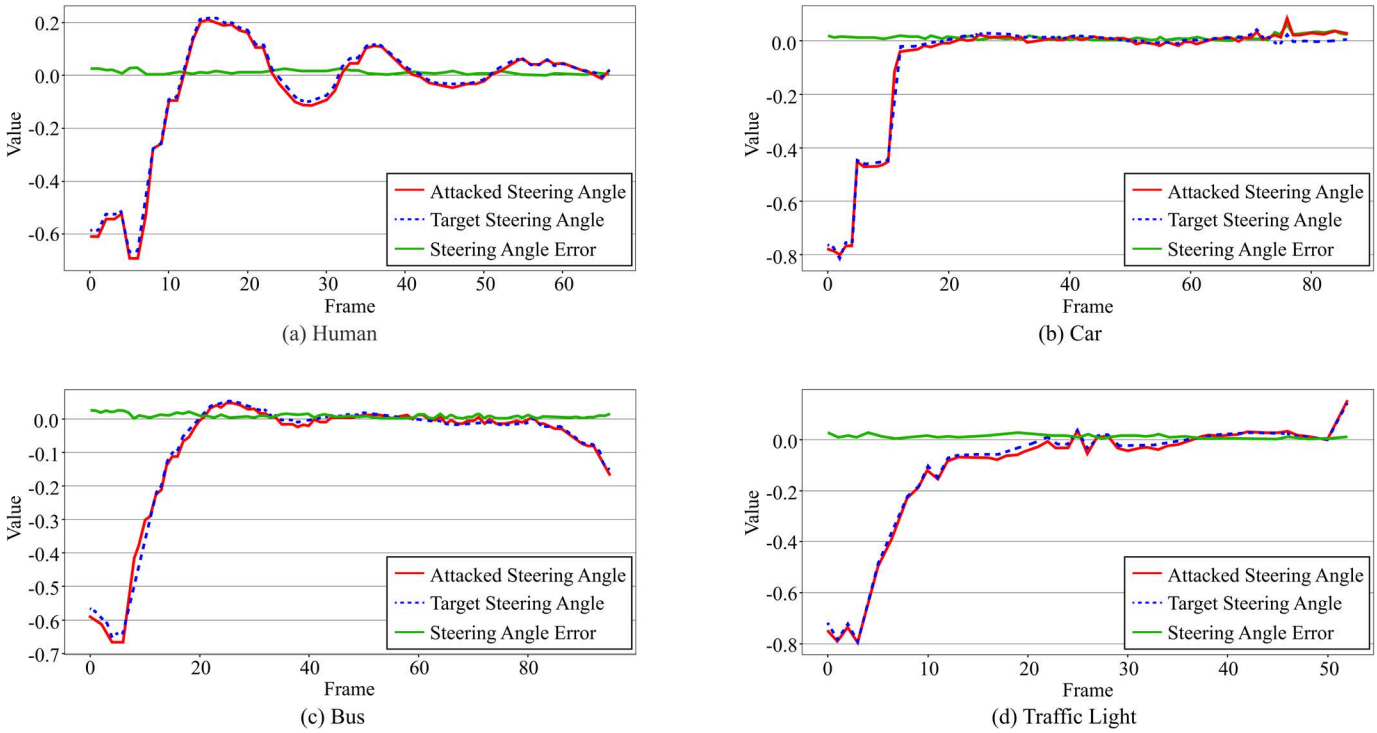

Fig. 2. Comparison of Attacked Steering Angle (red), Target Steering Angle (blue), and Steering Angle Error (green) for four different types of target objects.

A. Experimental Settings

To evaluate the attack performance of the proposed method, we constructed a driving environment using Udacity's self-driving car simulator. To ensure stable driving without deviation from the driving route, the vehicle's speed was limited to a maximum of 15 km/h. The target objects (i.e. human, car, bus, and traffic light), commonly encountered in real-world driving, were configured to remain stationary. The victim vehicle employed NVIDIA's DAVE2 [15], a representative autonomous driving model. For object-avoidance training, we built a dataset of 25,000 images collected from the simulator, consisting of 12,000 target-object images and 13,000 track images. The detection model adopted the YOLOv5 [16] architecture, pre-trained on the COCO dataset [17]. The adversarial model was implemented based on the U-Net [18] architecture and trained on images containing target objects from the constructed dataset.

The evaluation metrics are defined as follows:

- Attack Success Rate (ASR)

$$= \frac{\text{\# of Collisions}}{\text{\# of Encounters with Target}} \times 100$$

- Steering Angle Error (SAE)

$$= |\text{Target Steering Angle} - \text{Attacked Steering Angle}|$$

The environment and hyperparameters used in our evaluations are as follows: (1) Autonomous Driving Model: NVIDIA Geforce RTX 4060 GPU, Python 3.8.10, PyTorch 2.0.1+cu118, and 50 epochs; (2) Adversarial Model: NVIDIA Geforce RTX 3090 GPU, Python 3.8.10, PyTorch 2.0.1+cu118, and 10 epochs.

TABLE I
COMPARISON OF ATTACK SUCCESS RATE FOR THE PROPOSED METHOD, RANDOM PERTURBATIONS, AND WITHOUT PERTURBATIONS (W/O PERTURBATIONS).

Methods	Human	Car	Bus	Traffic-Light	Avg.
Proposed Method	**80.0%**	**100.0%**	**100.0%**	**80.0%**	**90.0%**
Random Perturbations	10.0%	20.0%	20.0%	20.0%	17.5%
W/O Perturbations	0.0%	10.0%	10.0%	20.0%	10.0%

B. Experimental Results

(RQ#1) Effectiveness of the adversarial perturbations in object collision: To evaluate RQ#1, we quantitatively compared the ASR of adversarial perturbations generated by the proposed method with that of random perturbations. As a baseline, we also present the ASR without perturbations. The comparison was conducted in a driving simulation involving four different types of target objects (human, car, bus, and traffic light). For each object, 10 attack trials were performed, resulting in a total of 40 attacks. Table I presents the ASRs of the proposed method, random perturbations, and W/O perturbations for each target object. The proposed method

achieved the highest average ASR of 90% across all target objects. Notably, while random perturbations achieved 0%p to 10%p higher ASR than W/O perturbations for each target object, the proposed method achieved an ASR that was at least 60%p and at most 90%p higher. These results demonstrate that the proposed method can generate adversarial perturbations that effectively manipulate the steering of the victim vehicle, inducing collisions with various types of target objects at a high success rate.

(RQ#2) Target steering angle tracking performance of the proposed method: To evaluate RQ#2, we assessed the performance of the proposed method in manipulating victim vehicle steering toward the center of the target object, measured by SAE. SAE is defined as the difference between the Target Steering Angle, required for collision with the target object, and the Attacked Steering Angle, manipulated by the proposed method. A smaller SAE value, closer to zero, indicates that the victim vehicle's steering is more accurately aligned with the target object. Fig. 2 illustrates the Target Steering Angle (blue), Attacked Steering Angle (red), and SAE (green) per frame for four types of target objects. The average SAE values were 0.0083 (human), 0.0105 (car), 0.0120 (bus), and 0.0101 (traffic light), while the maximum SAE values were 0.0258, 0.0275, 0.0606, and 0.0269 for the same objects. These represent an average SAE of approximately 1% and a maximum of 6% relative to the steering angle range (i.e. [-1, 1]), indicating that the proposed method can precisely control the victim vehicle while tracking the target object. These results demonstrate that the proposed method maintains accurate steering by reflecting real-time positional changes, thereby effectively guiding the victim vehicle to the target object.

V. Conclusion

In this paper, we proposed a novel adversarial attack method against autonomous driving systems that precisely controls the steering of a victim vehicle to induce collisions with target objects at high success rates. The proposed method generates optimized adversarial perturbations by reflecting the real-time relative position of the target object during victim vehicle motion, and maliciously manipulates the victim vehicle's steering to collide with the target object. The experimental results on the Udacity's self-driving car simulator, which supports various driving scenarios, show that the proposed method achieved an average ASR of 90% across all target objects and a low SAE close to zero. These results demonstrated that the proposed method successfully induced collisions by manipulating the victim vehicle's steering to direct it toward stationary target objects at high success rates.

For future work, we plan to extend adversarial attacks beyond steering control to other vehicle control parameters, such as throttle, thereby enabling collision with dynamic targets. In addition, we will investigate adversarial defense mechanisms to enhance the robustness of autonomous driving models.

Acknowledgment

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (RS-2025-16066331).

References

[1] L. J. Karam, J. Katupitiya, V. Milanes, I. Pitas, and J. Ye, "Autonomous Driving: Part 1-Sensing and Perception [From the Guest Editors]," *IEEE Signal Processing Magazine*, vol. 37, no. 4, pp. 11–13, 2020.

[2] E. Yurtsever, J. Lambert, A. Carballo, and K. Takeda, "A Survey of Autonomous Driving: Common Practices and Emerging Technologies," *IEEE Access*, vol. 8, pp. 58443–58469, 2020.

[3] W. Jia, Z. Lu, H. Zhang, Z. Liu, J. Wang, and G. Qu, "Fooling the eyes of autonomous vehicles: Robust physical adversarial examples against traffic sign recognition systems," *arXiv preprint arXiv:2201.06192*, 2022.

[4] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, F. Tramer, 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*, pp. 1625–1634, 2018.

[5] H. Wu, S. Yunas, S. Rowlands, W. Ruan, and J. Wahlström, "Adversarial Driving: Attacking End-to-End Autonomous Driving," in *IEEE Intelligent Vehicles Symposium*, pp. 1–7, 2023.

[6] L. Chen, P. Wu, K. Chitta, B. Jaeger, A. Geiger, and H. Li, "End-to-End Autonomous Driving: Challenges and Frontiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 12, pp. 10164–10183, 2024.

[7] P. Wu, L. Chen, K. Chitta, B. Jaeger, A. Geiger, and H. Li, "Trajectory-guided control prediction for end-to-end autonomous driving: A simple yet strong baseline," in *Advances in Neural Information Processing Systems*, vol. 35, pp. 6119–6132, 2022.

[8] N. Akhtar and A. Mian, "Threat of Adversarial Attacks on Deep Learning in Computer Vision: A Survey," *IEEE Access*, vol. 6, pp. 14410–14430, 2018.

[9] S. H. Park, S.-H. Lee, M. Y. Lim, P. M. Hong, and Y. K. Lee, "A comprehensive risk analysis method for adversarial attacks on biometric authentication systems," *IEEE Access*, 2024.

[10] P. M. Hong, S. H. Kang, J. Kim, J. H. Kim, and Y. K. Lee, "Adversarial2adversarial: Defending against adversarial fingerprint attacks without clean images," in *Proceedings of the International Conference on Information and Communication Technology Convergence*, pp. 1278–1282, 2023.

[11] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and Harnessing Adversarial Examples," *arXiv preprint arXiv:1412.6572*, 2014.

[12] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards Deep Learning Models Resistant to Adversarial Attacks," *arXiv preprint arXiv:1706.06083*, 2017.

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, p. 139–144, 2020.

[14] J. Zhang, J. W. Keung, Y. Xiao, Y. Liao, Y. Li, and X. Ma, "UniAda: Universal Adaptive Multiobjective Adversarial Attack for End-to-End Autonomous Driving Systems," *IEEE Transactions on Reliability*, vol. 73, no. 4, pp. 1892–1906, 2024.

[15] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, and K. Zieba, "End to End Learning for Self-Driving Cars," *arXiv preprint arXiv:1604.07316*, 2016.

[16] R. Khanam and M. Hussain, "What is YOLOv5: A deep look into the internal features of the popular object detector," *arXiv preprint arXiv:2407.20892*, 2024.

[17] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common Objects in Context," in *Proceedings of the European Conference on Computer Vision*, pp. 740–755, 2014.

[18] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proceedings of the Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, 2015.