A Cross-Verification Approach with Publicly Available Map for Detecting Off-Road Attacks against Lane Detection Systems

Takami Sato†, Ningfei Wang†, Yueqiang Cheng‡, Qi Alfred Chen†
†University of California, Irvine; ‡NIO Security Research

***Abstract*—Automated Lane Centering (ALC) is one of the most popular autonomous driving (AD) technologies available in many commodity vehicles. ALC can reduce the human driver's efforts by taking over their steering work. However, recent research alerts that ALC can be vulnerable to off-road attacks that lead victim vehicles out of their driving lane. To be secure against off-road attacks, this paper explores the potential defense capability of low-quality localization and publicly available maps against off-road attacks against autonomous driving. We design the first map-fusion-based off-road attack detection approach, LaneGuard, LaneGuard detects off-road attacks based on the difference between the observed road shape and the driver-predefined route shape. We evaluate LaneGuar on large-scale real-world driving traces consisting of 80 attack scenarios and 11,558 benign scenarios. We find that LaneGuard can achieve an attack detection rate of 89% with a 12% false positive rate. In real-world highway driving experiments, LaneGuard exhibits no false positives while maintaining a near-zero false negative rate against simulated attacks.**

I. Introduction

Automated Lane Centering (ALC) [1], classified as Level-2 driving automation, is one of the most popular autonomous driving (AD) technologies widely available in many commercial vehicles such as Tesla, GM Cadillac, Honda Accord, Toyota RAV4, and Volvo XC90. ALC can automatically steer a vehicle and maintain it within its current driving lane, which is detected by windshield cameras. Despite its convenience for drivers, the ALC system carries significant security and safety implications. Despite its convenience for human drivers, ALC systems carry significant security and safety implications. DRP attack [2] demonstrated that the commercial ALC system can be vulnerable to maliciously created dirty road patches. RAP-ALC attack [3] shows that a malicious patch on the back of a leading vehicle can mislead the victim vehicle controlled by an ALC to out of the driving lane. Due to the nature of ALC, which primarily controls steering, the attack effect on ALC is realized by lateral movement that causes the victim vehicle to deviate from its driving lane. We call this attack *off-road* attack by following the prior work [4].

To fundamentally defend against off-road attacks, two major approaches are actively researched: sensor fusion and map fusion. Sensor fusion utilizes sensing information from multiple sensors other than the windshield cameras to cross-check the detected lanes. However, sensor fusion has critical limitations to fully defend against off-road attacks: (1) The lane marking is not easy to detect other than windshield cameras. While prior research [5] demonstrates lane detection with LiDAR, it is fundamentally hard to sense the pattern on the road for other sensors than cameras; (2) The use of other sensors may open another attack vectors as there is no guarantee that the newly added sensors are more secure than the sensing with the windshield cameras.

Map fusion utilizes off-line map information to cross-check the detected lanes or to merely use the lane information in the map data. This approach is commonly used in Level-4 AD vehicles [1]. For example, Baidu Apollo [6] and Autoware [7] merely use the lane line information in map data without detecting lane lines on the fly. Since map data is typically managed by companies privately, it is not easy to compromise the data by the attacker. However, there are two major challenges in the current map fusion used in the Level-4 AD: (1) the Level-4-AD grade map, so-called High Definition (HD) map [8], needs tremendous cost to create and maintain and thus it is hard to scale to cover major roads. For example, Waymo [9] can only operate within the geo-fences such as the central area of San Francisco or Phoenix; (2) the map fusion generally requires very high-accurate vehicle localization, which is typically enabled by LiDAR localizer in the Level-4 AD. However, AD-grade LiDARs are still too expensive to install into commodity vehicles. Due to the two challenges, map fusion is not adopted in the current popular ALC systems. The map data is only referenced for navigation as in Tesla [10] and OpenPilot [11].

However, map fusion may still have meaningful information to defend against off-road attacks even with low-quality localization (e.g., GNSS) and publicly available maps (e.g., Google Map [12] and OpenStreetMap [13]). This question motivates us to design a cross-verification approach, LaneGuard, to detect off-road attacks by cross-checking the lane line detection with the lane information in publicly available maps. In §III, we explain the detailed design of our cross-verification approach, LaneGuard. In §III-D, we evaluate the attack detection capability of LaneGuard and perform false positive analysis in benign scenarios. We also conduct a feasibility study during online real-world driving. Finally, we discuss the insights and limitations of our study in §IV.

In summary, our study makes the following contributions:

Symposium on Vehicles Security and Privacy (VehicleSec) 2024
26 February 2024, San Diego, CA, USA
ISBN 979-8-9894372-7-6
https://dx.doi.org/10.14722/vehiclesec.2024.23032
www.ndss-symposium.org

- We are the first to design a map-fusion-based off-road attack detection method under low-quality localization and publicly available maps.
- We conduct a large-scale evaluation with multiple publicly available maps and real-world driving traces consisting of 80 attack scenarios and 11,558 benign scenarios. We find that LaneGuard can detect 89% of off-road attacks with a 12% false positive rate.
- We perform a feasibility study of LaneGuard on a real-world highway. LaneGuard shows 0% false positive rate with almost 0% false negative rate against simulated attack traces.

II. Background

A. Off-Road Attacks against ALC Systems

Recent studies demonstrate that ALC systems are vulnerable to off-road attacks enabled by adversarial attacks [2], [3]. These attacks compromise the DNN-based lane detection in the ALC systems to lead the victim out of their driving lane. DRP attack [2] demonstrates that it can lead an ALC-controlled vehicle out of the driving lane by fooling the DNN-based lane detection with a maliciously designed road patch pretending a benign but dirty road patch. Another study [14] shows that they can mislead Tesla Model S to the adjacent lane by putting several small stickers on the road without the original lane line. Phantom attack [15] also demonstrates that they can mislead Tesla Model S by projecting fake lane lines from a drone in the nighttime. The motivation of this study is to design an effective attack-defection methodology to defend against off-road attacks including not only these existing attacks but also any attacks compromising lane detection.

B. Prior Countermeasures against Off-Road Attacks

To defend against off-road attacks, there are 3 potential approaches: sensor-fusion, map-fusion, and software-based defenses. As discussed in §I, sensor-fusion-based defense requires more cost for additional sensors (e.g., LiDAR [5]) and these sensors could be new attack channels. For map-fusion-based defense, no prior work evaluates the capability in lower-level AD setups than Level-4 AD. This is one of the motivations for our research. For software-based defense, Sato et al. [2] evaluate possible defenses but none of them can be effective without harming the performance in benign cases. So far, none of the defenses against adversarial attacks are successful and the newly-proposed defenses are constantly being defeated over time [16], [17]. This also motivates us to seek a map-fusion-based defense.

C. Publicly Available Maps

Many recent web platformers host their routing and navigation services on their map data as a part of their services, such as Google Maps [12] and Bing Maps [18]. These map services allow users to download the navigation routes upon their query via apps or APIs. OpenStreetMap [13] (OSM) is a free and open geographic database, which is maintained by volunteers around the world. While OSM itself is just indexed map data, many routing and navigation services are developed on the OSM such as OSRM [19] and OpenRouteService [20]. We can also import the OSM data into PostGIS [21] and query a route with pgRouting [22].

III. Methodology

In this section, We illustrate the design details of LaneGuard, which is a cross-verification approach to detect off-road attacks by comparing the lane line detection with the lane information in publicly available maps.

A. Threat Model

We assume that the attacker can launch off-road attacks by compromising the detected lane line information in an ALC system installed in the victim vehicle by some attack vectors, e.g., physical-world adversarial attacks discussed in §II-A or malware. The victim's ALC can access publicly available map data and its routing services like those discussed in §II-C. With GPS and the IMU data, we assume that the victim's ALC has meter-level vehicle localization information that can know which road the victim is driving and its rough location, but the victim vehicle's accurate postural is not available.

B. Design Overview of LaneGuard

Fig. 1 illustrates the detection procedure of our LaneGuard. As described, (1) we first map the current driving route and the vehicle positions at the current and previous frames. The route is represented by a parameterized curve, e.g., spline and polynomial curves. We calculate the heading angle based on the position difference between the current and previous frames since we cannot directly obtain the vehicle heading due to its ill-quality localizations discussed in §III-A. (2) We project the route trajectory and the vehicle positions into the vehicle's local coordinate system where the vehicle heading is along with the x-axis. In this coordinate system, we can also plot the detected lane center obtained from the ALC system. The lane center is also represented by a parameterized curve. (3) We translate the route trajectory along with the y-axis to go through the origin where is the current vehicle position. Major routing services generally do not provide lane-level information, but road-level information, i.e., the route trajectory is drawn to pass through the center of the road. This is our approach to handling the coarse-grained route information from the map. We assume that the victim is not currently under attack effects and successfully driving the road. (4) Finally, we calculate the area between the shifted route trajectory and the detected lane center and use this area as a detection metric δ to judge if the vehicle is under off-road attack, i.e., LaneGuard considers that an off-road attack is ongoing if δ is larger than a predefined threshold. For the calculation of δ, we set a distance threshold D and only consider the area until the distance because the lane line detection at far points is generally inaccurate. In summary, the attack detection metric δ can be calculated as follows:

$$\delta := \int_0^D |\mathrm{ShiftedRoute}(x) - \mathrm{LaneCenter(x)}|dx. \quad (1)$$

C. Route Path Smoothing

As described in §III-B, LaneGuard requires the driving route trajectory represented by a parameterized curve. However, the major routing services discussed in §II-C return the route information represented by a sequence of points in a geodetic coordinate system. The simplest method is to connect them with a wire, but we find that this significantly degrades

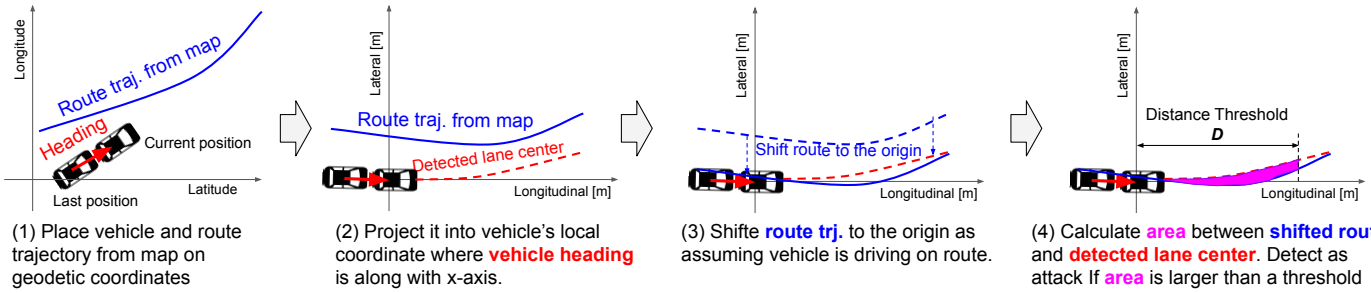
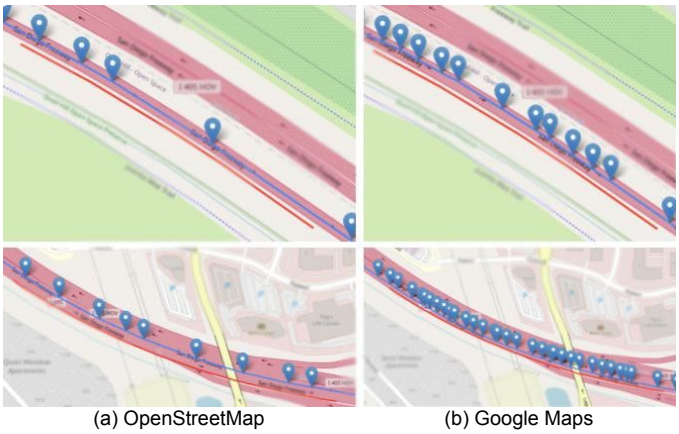


Fig. 1: Overview of our LaneGuard. Its effective but simple idea is to detect off-road attacks by cross-checking the difference between the route trajectory shape and the detected road center line. To handle the low map and localization qualities, we assume that the vehicle heading matches the road direction as it is driving the road if off-road attacks have not been effective yet.



(a) OpenStreetMap (b) Google Maps

Fig. 2: Comparison of OpenStreetMap and Google Maps on a highway. The markers represent waypoints of the driver-defined route. The blue and red lines are the driver-defined route and the GNSS trajectory of our driving, respectively.

the detection accuracy of LaneGuard. The frequency of points in a route varies widely between routing services. Particularly, Google Maps [12] generates a route with very high frequency but perturbed points, which should be automatically generated by Google users' prob data. If we directly connect them with a line or interpolate them with a curve (e.g., spline), the calculated route trajectory will be highly zig-zagged. Fig. 2 shows To handle this issue, we first apply a Gaussian filtering-based path smoothing for the route points, and then we apply the cubic spline interpolation for the smoothed points to get the parameterized curve.

D. Evaluation

We evaluate the performance of the LaneGuard with respect to its detection capability against off-road attacks and its sensitivity against benign scenarios.

1) Evaluation on Attack Detection Capability:

Experimental Setup. We simulate the attack effect of the off-road attacks by using the attack traces used in the DRP attack [2], which is one of the most effective off-road attacks so far and the only prior work that shared their evaluation code and scenarios with real-world driving traces. The scenarios are extracted from the comma2k19 dataset [23] and cover 40 eligible 10-second driving clips. Half of them are on highways, the other half are on local roads. For the vehicle location, we use the GNSS positions in the comma2k19 dataset [23] corrected with a smartphone-grade UBlox's GNSS receiver. For each clip, we generate attacks to lead the victim to the left and right, respectively. In total, there are 80 different attack scenarios. To simulate the attack-influenced driving, we use the input transformation with the perspective transformation as used in [2]. We use the lane detection model used in the OpenPilot v0.7.0 [11]. With these scenarios, we generate the DRP attacks and evaluate the attack detection rates of the LaneGuard. We place the DRP attack patch 7 m away from an attack generation point.

We start the driving simulation 1 second before the attack generation point, i.e., the distance from the simulation start point to the patch varies in each scenario based on the vehicle speed. From the simulation start point, we simulate the driving for 2 seconds considering the average attack success time of the DRP attack is around 1 second. For publicly available maps, we evaluate 2 different maps: OpenStreetMap [13] and Google Maps [12]. To evaluate the quality of the publicly available maps, we also evaluate 2 other route trajectories that can be seen as similar to ground truth. One is the human driving trajectory of the comma2k19 dataset. Another one is a simulated driving trajectory with benign scenarios. The simulated driving trajectory is generated by calculating the vehicle position by using the bicycle vehicle motion model [24] based on the lane detection results.

Results. Table I lists the detection rates, the best thresholds δ to achieve them, and the average δ in attacked and benign scenarios. As shown, the publicly available maps, OpenStreetMap and Google Maps, have the highest detection rate as 89% of the off-road attacks are correctly detected. Meanwhile, the detection rates with the human driving and simulated route trajectory are lower than these publicly available maps. We consider that these approaches suffer from inaccuracies in their GNSS and/or bicycle models, and this observation highlights the necessity of utilizing offline map information for defense since online sensing always contains a certain level of error.

For the results with OpenStreetMap and Google Maps, the average benign δ is around 30 m^2. This means that the route trajectory and the detected lane center have around 1.5 m deviation average since we use $D = 20$ m (§III-B). As typical

TABLE I: Attack detection rates of LaneGuard on different publicly available maps and baselines.

Map/Baseline	Detection Rate	Threshold θ [m^2]	Avg. Attacked δ [m^2]	Avg. Benign δ [m^2]
OpenStreetMap	89%	89.8	141.1	28.6
Google Maps	89%	92.3	143.1	33.1
Human Driving	85%	74.2	141.0	28.8
Simulated Route	76%	125.7	142.9	43.3

smartphone-grade GNSS is accurate to within a 4.9 m [25], the 1.5 m deviation in benign scenarios is reasonable. Considering the typical lane width is 2-3 m, this deviation is not so accurate, but it still has the potential to detect off-road attacks that try to largely deviate the victim vehicles out of the lane. On the other hand, The average attacked δ is around 140 m^2, i.e., 7 m average deviation between the detected lane center and the route trajectory. The best threshold θ for the attack detection thus can be around 90 m^2 (4.5 m average deviation). Since these deviations are close to the accuracy of the smartphone-grade GNSS, it may be challenging for this naive single-frame δ-based approach to detect all attacks as 11% of attacks are not correctly detected even with OpenStreetMap and Google Maps. To further explore this, we will evaluate more advanced detection designs leveraging the knowledge in multi frames.

2) False Positive Analysis: For a defense mechanism to be effective, it must not do much harm to the usability of the application it defends. As the LaneGuard is designed to detect off-road attacks, we evaluate the sensitivity of LaneGuard detection in large-scale benign driving traces.

Experimental Setup. To evaluate the false positive rate on benign driving, we extracted further scenarios from the comma2k19 dataset [23]. We split all driving traces in the comma2k19 dataset into every 5 seconds and obtain 30,565 5-second driving traces. Among the driving traces, we select the traces for the evaluation with the following criteria: (1) its minimum speed is larger than 45 km/h, which is a typical operational speed of OpenPilot [11]; (2) the lateral deviation between the human driving in the trace and simulated trajectory based on the lane detection result = is less than 1 m because such scenarios should not be in the operational domain of ALC (e.g., lane changing and turning at an intersection). After the filtering, we eventually find 11,558 valid driving traces and calculate the detection metric δ for the first 20 frames (1 second) of the traces, i.e., we evaluate the false positive with 231,160 frames. Note that each frame has a 0.05-second duration. For the map, we use OpenStreetMap and the route trajectory is calculated by pgRouting [22].

Results. Fig. 3 shows the histogram of the detection metric δ in the 57,790 frames. The red vertical line represents the detection threshold θ = 89.8 m^2 which is the best threshold for the attack detection as discussed in §III-D1. With this threshold θ, the false positive rate is 12%. As shown, the LaneGuard does not cause false positives for the majority of the benign frames. There could be multiple potential causes such as inaccurate matching with map and location, inaccurate vehicle heading calculation, and inaccuracies in the GNSS localization. To diagnose 12% of long-tail false positive cases, we will perform further real-world online detection evaluation on a highway, which is the main operational domain of ALC.

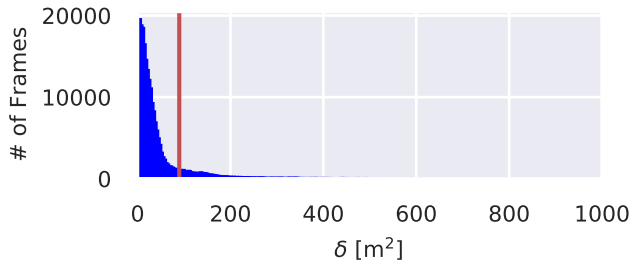


Fig. 3: Histogram of the detection metric δ in the 57,790 frames. The red vertical line is the detection threshold θ = 89.8 m^2 with a 12% false positive rate.

(a) Failure on map matching (b) Mismatch on heading and road curvature

Fig. 4: Case studies on false positives: (a) The map is associated with the wrong point as the road curve should start at a further point; (b) the vehicle heading does not match with the road curvature while the detected lane center looks accurate.

3) Failure Case Analysis: To diagnose the root causes of the false positives, we diagnose them and find 2 common causes for false positives. Fig. 4 shows the 2 representative scenarios. In (a), the driver-defined route (dotted yellow line) starts curving to the left even though the road is still straight and the detected lane center is also detected like so. The road soon starts curving after around 100 m away. This false positive is caused by associating the GNSS position with the wrong point of the road. In (b), the false positive is caused by breaking the assumption that the vehicle heading and road curvature are the same. As shown, the vehicle is slightly more heading to the left. Furthermore, we think that the detected lane center represents the actual road more accurately than the map route. While the LaneGuard can handle the majority of benign scenarios, the false positives derived from the ill-quality of localization and maps are inevitable.

4) Attack Detection with Multi-Frame Metrics: As a simple extension, LaneGuard can leverage the information in the past multiple frames instead of the single-frame δ. LaneGuard depends on low-quality GNSS localization and publicly available maps, multi-frame aggregation of the detection metric δ (e.g., averaging) may improve the detection accuracy and reduce false positive rates.

Experimental Setup. We use the same setups used in §III-D1 (for attack detection rate analysis) and §III-D2 (for false positive analysis). We denote the multi-frame attack detection metrics as δ^{f}_{w}, where f means the aggregation method and w means the number of frames to aggregate δ from the current frame to the past with f. In this work, we explore four aggregation functions: mean, median, min, and max.

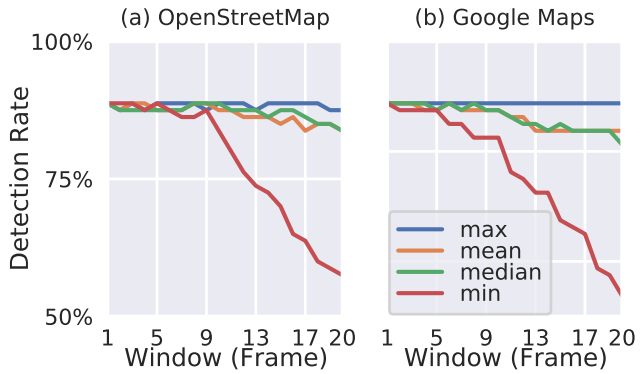


Fig. 5: Attack detection rates on different multi-frame windows and their aggregation methods.

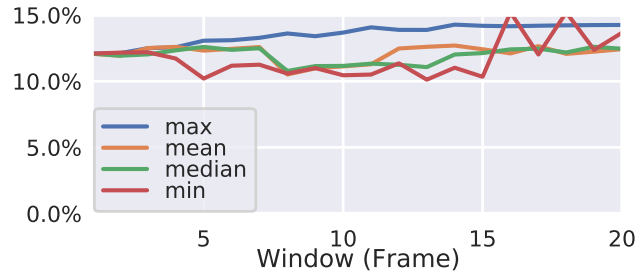


Fig. 6: False positive rates on different multi-frame windows and their aggregation methods.

The attack detection threshold θ is decided to maximize the detection rate and used for the false positive analysis for each aggregation method and window size.

Results. Fig. 5 and 6 show the attack detection rates and false positive rates on different multi-frame windows and their aggregation methods. As shown, the multi-frame detection metrics do not show any improvements in both detection and false positive rates. While longer aggregation frames with minimum aggregation slightly reduce the false positive rates, minimum aggregation largely harms the attack detection rates. We consider that these counterintuitive results should be due to the side-effect of the LaneGuard design, which is designed to be robust against low-quality localization and publicly available maps. As described in §III-B, LaneGuard assumes that the current vehicle heading is the same as the curvature of the current driving road; path smoothing is also applied for the road shape. These operations may dismiss the difference between δ of the close frames, and produce similar δ values. The aggregations thus do not make meaningful differences from the single-frameδ.

5) Ablation Study on Path Smoothing: As discussed in §III-C, several map services such as Google Maps generate routes with highly frequent points that should be automatically generated by their users' prob data. Table II lists the detection rates, the best thresholds δ to achieve them, and the average δ in attacked and benign scenarios, when we disable the path smoothing in the driver-defied route trajectory. As expected, the detection rates are significantly dropped from

TABLE II: Attack detection rates of LaneGuard *without the path smoothing* on different publicly available maps.

Map	Detection Rate	Threshold θ [m^2]	Avg. Attacked δ [m^2]	Avg. Benign δ [m^2]
OpenStreetMap	91%	106.8	161.6	30.6
Google Maps	65%	109.4	198.0	66.2

89% to 65% on Google Maps compared to the case when the path smoothing is enabled as in Table I. Meanwhile, the detection rate of Open Steet Map is slightly improved even without the smoothing. These results indicate that the benefit of path smoothing highly depends on the map source. For Google Maps, path smoothing is quite effective since the path waypoints are frequent but noisy. For OpenStreetMap, the waypoints are sparse and could be already smoothed. Smoothing can reduce the noise effects, but it also causes information losses. For LaneGuard implementation, we need a pre-assessment of the publicly available maps we plan to use, particularly about the characteristics of the route points.

6) Feasibility Study on Real-World Highway: We evaluated the detection capability of the LaneGurad in §III-D1 and its false positive analysis in §III-D2 with the driving traces in the comma2k19 dataset [23].

Experimental Setup. As shown in Fig. 7, we installed an EON Devkit, the official dashcam device of OpenPilot [11], onto the windshield of a sedan vehicle. We drove the 2 km highway route without changing lanes to be consistent with the driving controlled by ALC. The vehicle speed was maintained around 120 km/h. We used a laptop to connect the dashcam and obtained real-time driving logs via SSH. LaneGuard operated with the logs and showed online detection results on the laptop. To simulate the attack scenario, we added the offset of the average DRP attack trace generated in §III-D1 into the online detected lane center. For the map data, we use OpenStreetMap [13] and pgRouting [22].

Results. For the attack detection accuracy, we did not observe any false positives and only observed four times false negatives for the simulated attack. All false negatives only lasted 1 frame (0.05 sec) and thus the driver can get an attack detection alert at least within 0.1 sec, which is an ignorable delay considering human reaction time. For the latency of LaneGuard, we did not see particular delays. We find that LaneGuard is efficient enough to handle real-time driving logs sent from OpenPilot at every 0.05 sec (20 Hz).

IV. Discussions and Limitations

Localization Quality. In this work, we only use GNSS localization with a smartphone-grade receiver. While other expensive localizers such as LiDAR localizers are not available for typical ALC systems, we may still use other information sources to improve the localization quality. For example, IMU data is widely used for improving localization quality [6], [7]. In our setup, we may also use the driver-define route under some assumptions, e.g., the vehicle should be on the route.

Further Improvement on Detection Algorithm. This study is motivated by our question about the potential of low-quality localization and publicly available maps to defend against

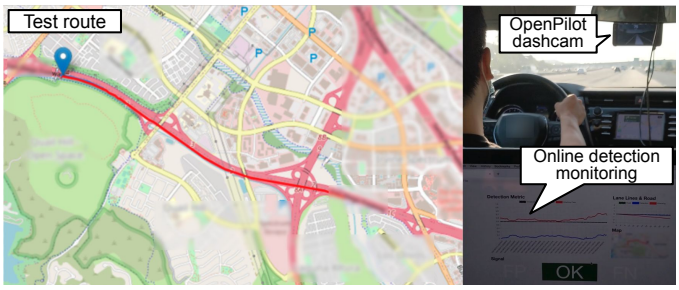


Fig. 7: Overview of our feasibility study on a real-world highway: The highway route for the real-world evaluation (left) and the setup for the online LaneGuard detection on a real vehicle (right). The LaneGuard is operated on the laptop connecting to the OpenPilot dashcam via SSH.

off-road attacks. To simply answer this question, we focus more on exploring the usefulness of information sources (low-quality localization and publicly available maps) rather than designing complex but more effective detection algorithms as we are the first to design map-fusion-based defense against off-road attacks. As a straightforward way to improve detection accuracy, we can adopt a machine learning (ML)-based detection approach. In addition to our detection metric δ, we can integrate any information as a feature such as vehicle speed, steering angle, previous locations, and route information (e.g., highway and local road). However, for safety-critical applications, accountability is as important as detection accuracy. ML-based approaches typically need large theoretical or empirical efforts to ensure their accountability. We thus leave the ML-based detection for future work.

Latency and Energy Consumption of LaneGuard. In §III-D6, we find that LaneGuard can handle online driving logs sent from OpenPilot at every 0.05 sec. However, we run the LaneGuard on our laptop, which has much higher computational power than the smartphone-like OpenPilot's EON DevKit. To install the LaneGuard along with ALC systems, more detailed and quantitative evaluation of its latency and energy consumption analysis may help ALC developers estimate the cost of integrating the LaneGuard.

Detection against Adaptive Attacks. As LaneGuard detects off-road attacks based on the difference between road shape and the detected lane center, the attacker may design more stealthy attacks that gradually lead victims out of the lane while always keeping the difference below the threshold. However, this type of attack should require more time to lead the victim out of the lane as the attacker cannot largely compromise the detected lane center. For the attack against ALC, the attack duration needs to be below the driver's reaction time otherwise the driver can take over the driving and apply countersteering. We thus leave this for future study as this needs more advanced attack design and careful evaluation design for user study.

V. Conclusion

In this work, we explore the potential defense capability of low-quality localization and publicly available maps against off-road attacks. We design the first map-fusion-based off-road attack detection, named LaneGuard. To handle the ill-quality localization and map data, we introduce a key assumption that vehicle heading and the curvature of driver-defined route trajectory should match if off-road attacks have not been effective yet. To evaluate the performance of the LaneGuard, we perform the attack detection capability analysis on 80 attack scenarios and evaluate the false positive analysis on 11,558 benign scenarios. Through the evaluation, we find that LaneGuard can detect 89% of off-road attacks with 12% false positive rates. To further evaluate the usability of LaneGuard, we conduct real-world driving experiments on the highway, which is the main operational domain of ALC. In the experiments, we do not observe any false positives, i.e., 0% false positive rate while keeping almost 0% false negative rate against simulated attacks. Recently, map information has become publicly available on the internet, but its majority application is navigation. While publicly available maps are not as high quality as merely supporting AD, we hope that our study facilitates further utilization of map information to secure AD vehicles.

Acknowledgements

This research was supported in part by the NSF CNS-2145493, CNS-1929771, CNS-1932464, and USDOT UTC Grant 69A3552348327. We would like to sincerely thank the people at NIO Inc. for providing us with the internship opportunity to complete this work.

References

[1] "Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles," *SAE International*, 2016.

[2] T. Sato, J. Shen, N. Wang, Y. Jia, X. Lin, and Q. A. Chen, "Dirty Road Can Attack: Security of Deep Learning based Automated Lane Centering under Physical-World Attack," *USENIX Security*, 2021.

[3] Y. Cao, Y. Guo, T. Sato, Q. A. Chen, Z. M. Mao, and Y. Cheng, "Demo: Remote Adversarial Attack on Automated Lane Centering," in *AutoSec*, 2022.

[4] J. Shen, J. Y. Won, Z. Chen, and Q. A. Chen, "Drift with Devil: Security of Multi-Sensor Fusion based Localization in High-Level Autonomous Driving under GPS Spoofing," in *USENIX Security*, 2020.

[5] A. Hata and D. Wolf, "Road Marking Detection using LIDAR Reflective Intensity Data and Its Application to Vehicle Localization," in *ITSC*. IEEE, 2014.

[6] Baidu Inc., "Apollo," https://github.com/ApolloAuto/apollo, 2023.

[7] S. Kato, S. Tokunaga, Y. Maruyama, S. Maeda, M. Hirabayashi, Y. Kitsukawa, A. Monrroy, T. Ando, Y. Fujii, and T. Azumi, "Autoware On Board: Enabling Autonomous Vehicles with Embedded Systems," in *ICCPS'18*. IEEE Press, 2018, pp. 287–296.

[8] "HD Maps: New Age Maps Powering Autonomous Vehicles," https://www.geospatialworld.net/article/hd-maps-autonomous-vehicles/, 2017.

[9] "Waymo Has Launched its Commercial Self-Driving Service in Phoenix and it's Called Waymo One," https://www.businessinsider.com/waymo-one-driverless-car-service-launches-in-phoenix-arizona-2018-12, 2018.

[10] Tesla, Inc., "Tesla Model 3 Owner's Manual," https://www.tesla.com/sites/default/files/model_3_owners_manual_north_america_en.pdf, 2020.

[11] comma.ai, "OpenPilot: Open Source Driving Agent," https://github.com/commaai/openpilot, 2023.

[12] "Google Maps," https://www.google.com/maps/, accessed: 2023-12.

[13] "OpenStreetMap," https://www.openstreetmap.org, accessed: 2023-12.

[14] P. Jing, Q. Tang, Y. Du, L. Xue, X. Luo, T. Wang, S. Nie, and S. Wu, "Too Good to Be Safe: Tricking Lane Detection in Autonomous Driving with Crafted Perturbations," in *USENIX Security*, 2021.

[15] B. Nassi, Y. Mirsky, D. Nassi, R. Ben-Netanel, O. Drokin, and Y. Elovici, "Phantom of the ADAS: Securing Advanced Driver-Assistance Systems from Split-Second Phantom Attacks," in *ACM CCS*, 2020.

[16] A. Athalye, N. Carlini, and D. Wagner, "Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples," in *ICML*, 2018.

[17] F. Tramer, N. Carlini, W. Brendel, and A. Madry, "On Adaptive Attacks to Adversarial Example Defenses," *arXiv preprint arXiv:2002.08347*, 2020.

[18] "Bing Maps," https://www.bing.com/maps, accessed: 2023-12.

[19] O. contributors, "Open Source Routing Machine," https://github.com/Project-OSRM/osrm-backend, accessed: 2023-12.

[20] "OpenRouteService," https://openrouteservice.org/, accessed: 2023-12.

[21] "PostGIS, Spatial and Geographic Objects for PostgreSQL," https://postgis.net, 2018.

[22] "pgRouting: A routing library for PostgreSQL," https://pgrouting.org/, accessed: 2023-12.

[23] H. Schafer, E. Santana, A. Haden, and R. Biasini, "A Commute in Data: The comma2k19 Dataset," *arXiv preprint arXiv:1812.05752*, 2018.

[24] R. Rajamani, *Vehicle Dynamics and Control*. Springer Science & Business Media, 2011.

[25] F. Van Diggelen and P. Enge, "The World's First GPS MOOC and Worldwide Laboratory using Smartphones," in *ION GNSS+*, 2015.