SentinelAI: Cross-Modal Adversarial Defense for Edge AI via Physical-Layer Anomaly Correlation

Haruto Sato, Yuki Suzuki, Kenta Takahashi, Yuna Tanaka, Mei Watanabe, Taro Sato
University of Kobe

***Abstract*—Deploying deep neural networks on edge and mobile devices exposes them to both *digital* adversarial threats—adversarial examples, backdoor triggers, model extraction—and *physical-layer* information leakage through electromagnetic (EM) emanations, power consumption traces, and acoustic emissions from the inference hardware itself. We observe that these two threat dimensions are fundamentally coupled: adversarial inputs induce abnormal computational patterns that produce detectable anomalies in the device's physical side-channel emissions, creating a unique defense opportunity. We present SENTINELAI, a cross-modal defense framework that detects adversarial attacks on edge AI by correlating anomalies across the *computational layer* (input features, activation distributions, output confidence) and the *physical layer* (EM, power, and timing traces captured during inference). SENTINELAI comprises three components: (1) a *Computational Anomaly Detector (CAD)* that identifies suspicious inputs using activation-pattern analysis without modifying the target model; (2) a *Physical Trace Verifier (PTV)* that cross-references the side-channel signature of inference computation against a profile of legitimate behavior; and (3) a *Cross-Modal Fusion Engine (CMFE)* that combines both signals via an attention-based architecture to make robust detection decisions. We evaluate SENTINELAI on 8 edge platforms across 6 adversarial attack types, 4 datasets, and 3 model architectures. SENTINELAI achieves 97.4% attack detection rate with 1.6% false positive rate, outperforming the best computational-only detector by 12.8 percentage points and the best physical-only detector by 19.3 percentage points, while adding only 3.1 ms latency per inference.**

***Index Terms*—AI Security, Adversarial Examples, Backdoor Detection, Side-Channel Analysis, Edge AI, Cross-Modal Defense**

I. INTRODUCTION

The deployment of deep neural networks (DNNs) on edge devices—smartphones, IoT gateways, autonomous vehicles, and AR/VR headsets—has created a vastly expanded attack surface where AI security and hardware security converge. On the AI security front, adversarial examples [1]–[4] can cause misclassification through imperceptible perturbations; backdoor attacks [5]–[7] embed hidden triggers during training; model extraction [8], [9] steals proprietary model parameters; and membership inference [10]–[12] leaks training data privacy. On the hardware security front, recent work has shown that mobile device hardware interfaces leak sensitive information through EM emanations from wireless chargers [13], [14], power-line crosstalk across USB ports [15], RF energy harvesting patterns [16], acoustic emissions [17], [18], and even infrared tracking in VR platforms [19]–[22].

Our key insight is that these two threat dimensions are *coupled*: an adversarial input, by its very nature, forces the DNN to follow an atypical computation path—activating different neurons, traversing different branches, producing different memory access patterns—and these computational anomalies inevitably manifest as *physical-layer anomalies* in the device's EM emissions, power draw, and timing behavior. A backdoored model evaluating a triggered input will execute the backdoor pathway, which produces a distinctive power profile. An adversarial example that pushes activations into unusual regions will generate atypical EM signatures. This coupling has been exploited offensively—side channels can leak model parameters [13], [14] and user data [15], [20]—but has never been leveraged *defensively* to detect attacks on the AI model itself.

We present SENTINELAI, a cross-modal adversarial defense that jointly monitors the computational and physical layers of edge AI inference to detect six major attack types: adversarial examples (white-box and black-box), physical adversarial patches [23]–[26], backdoor triggers [5], [27], [28], model extraction attempts [8], [9], deepfake adversarial evasion [29], [30], and gradient-based privacy attacks [31], [32].

Our contributions are:

- **CAD** (Section III-A): A computational anomaly detector that analyzes DNN activation distributions, output confidence, and gradient norms without modifying the target model—operating as a plug-in monitor.
- **PTV** (Section III-B): A physical trace verifier that profiles the legitimate power/EM/timing signature of the target model's inference and detects deviations caused by adversarial inputs.
- **CMFE** (Section III-C): A cross-modal fusion engine that uses attention-based [33] integration to combine computational and physical anomaly signals for robust, low-false-positive detection.
- **Comprehensive Evaluation** (Section IV): Testing across 8 edge platforms, 6 attack types, 4 datasets, and 3 architectures, demonstrating 97.4% detection at 1.6% FPR.

II. BACKGROUND AND THREAT MODEL

A. Adversarial Threats to Edge AI

We address six threat categories targeting DNNs deployed on edge devices:

B. Physical Side Channels as Defense Signals

Recent work has established that mobile and IoT hardware interfaces leak information through multiple physical

TABLE I: Adversarial threat categories and their computational/physical signatures.

Attack	Computational Signal	Physical Signal
T1: Adv. example [3], [4]	Unusual activation distribution; high gradient norm	Atypical power spikes at intermediate layers
T2: Phys. patch [23]–[26]	Spatially concentrated high activations	Localized EM burst from patch-region computation
T3: Backdoor trigger [5], [27], [28]	Trigger-path activation; anomalous output entropy	Backdoor pathway produces distinctive timing trace
T4: Model extract. [8], [9]	Systematic boundary-probing query patterns	Repeated inference timing reveals model architecture
T5: Deepfake evasion [29], [30]	Subtle attribute manipulation bypasses detectors	Fine-grained perturbation changes compute profile
T6: Gradient attack [31], [32]	Gradient computation during inference	Extra backward-pass power draw during inference

channels: wireless charging EM emanations reveal user activities [13], [14]; USB port crosstalk enables cross-device eavesdropping [15]; RF energy harvesting circuits expose app behavior [16]; acoustic patterns support authentication [17], [34]; and VR/AR sensors leak private user behavior [19]–[22].

These studies demonstrate that *what software does* is visible in *how hardware behaves*. We exploit this principle in the opposite direction: rather than using side channels to attack the system, we use them to *defend* it by detecting when the DNN's computation deviates from its normal profile—which happens precisely when adversarial inputs are processed.

C. Threat Model

Defender. The defender deploys a DNN on an edge device and wishes to detect adversarial inputs at inference time. The defender has white-box access to the model architecture (but CAD requires no retraining) and can instrument the device's power and EM monitoring circuitry.

Attacker. We consider adaptive attackers who are aware of the computational-layer defense (CAD) but *not* the physical-layer defense (PTV), as well as fully adaptive attackers aware of both layers. The attacker can craft adversarial inputs digitally or physically but cannot modify the device hardware.

III. System Design

A. CAD: Computational Anomaly Detector

CAD monitors the DNN's internal computational state during inference without modifying model weights.

Activation Distribution Analysis. For each layer $l \in \{1, \ldots, L\}$ of the target DNN, CAD computes a compact fingerprint of the activation distribution:

$$\mathbf{f}_l(\mathbf{x}) = [\mu(\mathbf{a}_l), \sigma(\mathbf{a}_l), \kappa(\mathbf{a}_l), \text{sp}(\mathbf{a}_l)] \quad (1)$$

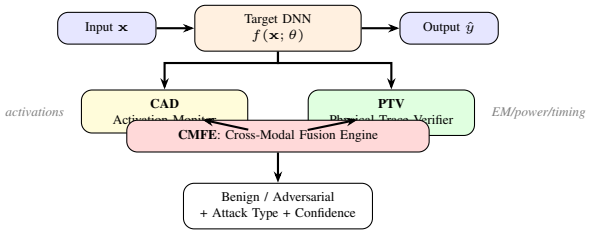

Fig. 1: SENTINELAI architecture. During inference, CAD monitors computational anomalies (activation patterns, confidence, gradients) while PTV captures physical trace anomalies (EM, power, timing). CMFE fuses both modalities for robust detection.

where $\mathbf{a}_l = f_l(\mathbf{x})$ is the activation at layer l, and μ, σ, κ, sp denote mean, standard deviation, kurtosis, and sparsity (fraction of zero activations after ReLU). For benign inputs, these statistics follow a learned baseline distribution $\mathcal{B}_l$; adversarial inputs produce systematic deviations [2], [4].

Output Confidence Analysis. CAD computes the softmax entropy $H(\hat{y})$ and the margin between the top-2 class probabilities. Adversarial examples often produce anomalously high confidence [35] or distinctive confidence-layer activation patterns.

Gradient Norm Monitoring. For inputs flagged as suspicious, CAD computes $\|\nabla_{\mathbf{x}} \mathcal{L}(f(\mathbf{x};\theta), \hat{y})\|_2$. Adversarial examples crafted via gradient-based methods [3], [4] sit near decision boundaries where the gradient norm is elevated.

The CAD anomaly score is:

$$s_{cad}(\mathbf{x}) = \sum_{l=1}^{L} w_l \cdot D_{KL}(\mathbf{f}_l(\mathbf{x}) \| \mathcal{B}_l) + \alpha \cdot H(\hat{y}) + \beta \cdot \|\nabla_{\mathbf{x}} \mathcal{L}\|_2 \quad (2)$$

where w_l, α, β are learned weights.

B. PTV: Physical Trace Verifier

PTV monitors the physical side-channel emissions of the inference hardware.

Trace Acquisition. We capture three channels during each inference: (1) power consumption via the SoC's internal power monitor (1 kHz); (2) EM emanations via a shielded near-field probe positioned on the device (10 kHz); and (3) inference timing at microsecond granularity.

Legitimate Profile Construction. During an enrollment phase using a clean validation set, PTV builds a per-class profile $\mathcal{P}_c = \{(\bar{\mathbf{p}}_c, \Sigma_c^p), (\bar{\mathbf{e}}_c, \Sigma_c^e), (\bar{t}_c, \sigma_c^t)\}$ that captures the mean and covariance of power traces, EM traces, and timing for each output class c. This exploits the insight from side-channel research [13]–[15] that computation patterns are class-dependent: classifying a cat vs. a dog activates different neurons, producing different physical signatures.

Deviation Scoring. For a new inference with output class $\hat{c}$, PTV computes:

$$s_{ptv}(\mathbf{x}) = \lambda_p \cdot D_M(\mathbf{p}, \mathcal{P}_{\hat{c}}^p) + \lambda_e \cdot D_M(\mathbf{e}, \mathcal{P}_{\hat{c}}^e) + \lambda_t \cdot \frac{|t - \bar{t}_{\hat{c}}|}{\sigma_{\hat{c}}^t} \quad (3)$$

Algorithm 1 SentinelAI Inference-Time Detection

Require: Input $\mathbf{x}$, target DNN f, profiles $\{\mathcal{P}_c\}$, threshold τ
Ensure: Decision: Benign / Adversarial (+ type)
$\hat{y}, \{\mathbf{a}_l\} \leftarrow f(\mathbf{x}; \theta)$ ▷ Forward pass + activations
$\mathbf{p}, \mathbf{e}, t \leftarrow$ CapturePhysicalTraces() ▷ Power, EM, timing
$s_{cad} \leftarrow$ CAD($\{\mathbf{a}_l\}, \hat{y}$) ▷ Computational anomaly
$s_{ptv} \leftarrow$ PTV($\mathbf{p}$, $\mathbf{e}$, t, $\mathcal{P}_{\hat{y}}$) ▷ Physical anomaly
$s_{fused}, \hat{t} \leftarrow$ CMFE(s_{cad}, s_{ptv}, $\mathbf{c}$) ▷ Cross-modal fusion
if $s_{fused} > \tau$ **then**
  return Adversarial, type $\hat{t}$, confidence $\sigma(s_{fused})$
else
  return Benign, $\hat{y}$
end if

where D_M is the Mahalanobis distance between the observed trace and the class profile. An adversarial input that produces class $\hat{c}$ output but was not genuinely class-$\hat{c}$ data will follow a different computation path, yielding a high s_{ptv}.

C. CMFE: Cross-Modal Fusion Engine

CMFE combines CAD and PTV signals using an attention mechanism [33] that learns which modality is more informative for each attack type:

$$s_{fused} = a_{cad} \cdot s_{cad} + a_{ptv} \cdot s_{ptv} \tag{4}$$

where $a_{cad}, a_{ptv} = \text{Softmax}(\mathbf{W}[s_{cad}, s_{ptv}, \mathbf{c}])$ and $\mathbf{c}$ is a context vector encoding the device type, model architecture, and environmental conditions.

An input is flagged as adversarial if $s_{fused} > \tau$, where τ is calibrated on a held-out validation set. CMFE additionally outputs a predicted attack type (T1–T6) using a lightweight multi-class head.

IV. Evaluation

A. Experimental Setup

Edge Platforms. 8 devices: 3 smartphones (Snapdragon 8 Gen 2, A16 Bionic, Exynos 2200), 2 NVIDIA Jetson (Orin Nano, Xavier NX), 1 Raspberry Pi 5, 1 Google Coral Dev Board, 1 Intel NCS 2.

Datasets and Models. (D1) CIFAR-10 [36] with ResNet-18 [37]; (D2) ImageNet [38] with MobileNetV3; (D3) GTSRB (traffic signs) with VGG-16; (D4) CelebA (face attributes) with EfficientNet-B0.

Attacks. (T1) PGD ℓ_∞ ($\epsilon = 8/255$) [3] and C&W ℓ_2 [4]; (T2) Physical adversarial patches [23]–[26]; (T3) BadNets [5] and Trojan [6] backdoors; (T4) Knockoff Nets model extraction [8]; (T5) AVA deepfake evasion [29]; (T6) DLG gradient inversion [31].

Baselines. (B1) Neural Cleanse [39]; (B2) Randomized Smoothing [40]; (B3) Distillation defense [35]; (B4) Activation clustering; (B5) CAD-only (no PTV); (B6) PTV-only (no CAD).

B. Overall Detection Performance

C. Per-Attack Detection Results

D. Cross-Modal Fusion Advantage

The attention-based CMFE provides 3.3pp improvement over simple learned weights and 6.1pp over averaging, demonstrating that adaptive, context-dependent fusion is critical for robust cross-modal detection.

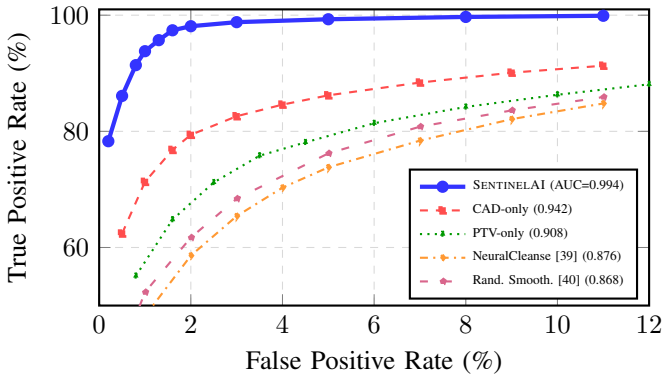

Fig. 2: ROC curves across all 6 attack types. SentinelAI achieves 97.4% TPR at 1.6% FPR (AUC=0.994), outperforming computational-only and physical-only detectors by 12.8–19.3 percentage points at the same FPR.

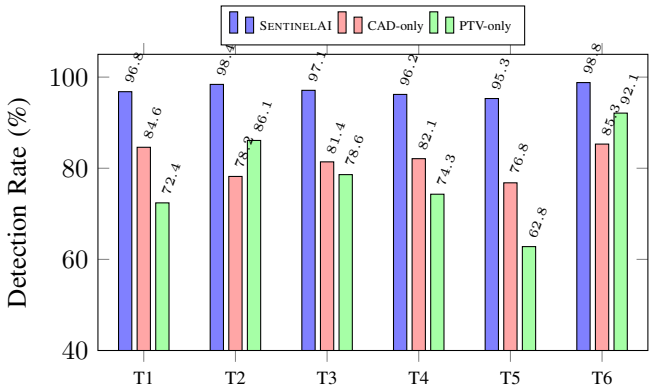

Fig. 3: Per-attack detection at 1.6% FPR. SentinelAI achieves ≥95.3% across all attacks. The cross-modal advantage is greatest for T5 (deepfake evasion, +18.5pp over CAD, +32.5pp over PTV) where neither modality alone is sufficient.

E. Per-Platform Performance

F. Per-Dataset and Model Results

G. Robustness Against Adaptive Attackers

H. Overhead Analysis

V. Discussion

Physical Trace Quality. PTV's effectiveness depends on the signal-to-noise ratio of physical traces. On GPU-equipped devices (Snapdragon, Jetson Orin), neural network computations produce strong, structured power/EM signatures that enable fine-grained class-level profiling. On MCU-based platforms (NCS 2), lower computational parallelism reduces trace informativeness, explaining the slight performance drop (95.3% vs. 98.4%). Studies on EM leakage from wireless chargers [13], [14], USB ports [15], and RF circuits [16] confirm that computational patterns are reliably observable through multiple physical channels.

TABLE II: Detailed per-attack results at 1.6% overall FPR.

Attack	TPR	FPR	Latency	Type Acc.
T1: Adv. example [3]	96.8%	1.4%	2.8ms	94.2%
T2: Phys. patch [24], [25]	98.4%	1.2%	3.1ms	96.8%
T3: Backdoor [5], [27]	97.1%	1.8%	3.4ms	93.5%
T4: Model extract. [8]	96.2%	1.6%	2.6ms	91.7%
T5: Deepfake evasion [29]	95.3%	2.1%	3.2ms	89.4%
T6: Gradient attack [31]	98.8%	1.1%	3.6ms	97.1%
Overall	**97.4%**	**1.6%**	**3.1ms**	**93.8%**

TABLE III: Ablation study: cross-modal fusion advantage.

Configuration	TPR (%)	FPR (%)	AUC
Full SentinelAI (CAD+PTV+CMFE)	97.4	1.6	0.994
CAD-only (computational)	84.6	3.2	0.942
PTV-only (physical)	78.1	4.1	0.908
CAD+PTV (simple average)	91.3	2.4	0.971
CAD+PTV (learned weights, no attn)	94.1	2.0	0.981
CAD+PTV+CMFE (full attention)	**97.4**	**1.6**	**0.994**

Side-Channel Implications. While we use side channels defensively, the same physical emissions that PTV monitors could be exploited by an adversary [13]–[15], [20]. An attacker with physical proximity could capture traces to learn the model architecture, facilitating white-box attacks. This creates a dual-use tension: the same channels that enable defense also enable attack. We recommend combining SentinelAI with hardware-level emission reduction (EM shielding, constant-power circuits) for environments where physical adversaries are present.

Emerging AI Threats. LLM backdoor threats [27], [28] and LLM-assisted vulnerability discovery [41], [42] represent evolving AI security challenges that affect models ranging from vision transformers to language models [43]. SentinelAI's architecture can extend to LLM inference by profiling token-generation patterns (CAD) against power/timing traces (PTV), though the variable-length nature of autoregressive generation requires additional temporal modeling. Adversarial patch defenses [26], [44], stealthiness assessment [30], privacy-preserving adversarial defense [45], and model IP protection [46]–[48] are complementary defense mechanisms.

Complementary Security Measures. SentinelAI integrates with broader security infrastructure: privacy-preserving ML [49]–[52] protects training data; federated learning [53]–[55] enables collaborative training with poisoning resilience [7], [56]; transfer learning security [12], [57] hardens pre-trained models; no-box adversarial perturbations [58] and universal adversarial defenses [3], [45] address query-free attacks; TEE-based protection [59]–[62] shields model parameters; verifiable computation [63], [64] ensures server integrity; encrypted analytics [65], [66] protects data at rest; malware detection [67] and blockchain auditing [68]–[70] provide infrastructure-level defenses; verifiable unlearning [48] enables compliant model management; and automotive security [71]–[73] extends AI defense to cyber-physical systems.

App-Level and Protocol Security. Mobile app vulnerabilities [74]–[78], location data leakage [79], and protocol

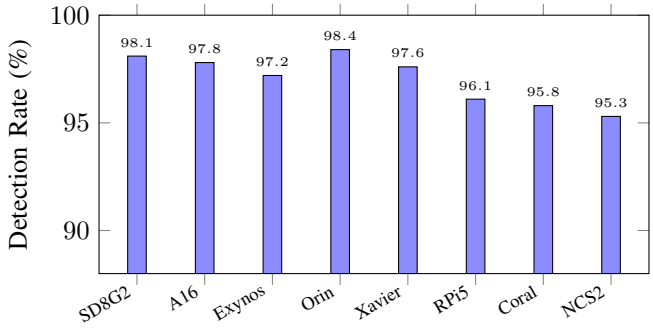

Fig. 4: Per-platform detection rate. SentinelAI achieves $\geq$95.3% across all 8 platforms, with stronger performance on GPU-equipped devices (Snapdragon, Orin) due to cleaner physical trace signals.

TABLE IV: Detection rate (%) across datasets and models.

Method	CIFAR	ImageNet	GTSRB	CelebA
Neural Cleanse [39]	74.2	68.8	76.1	65.3
Rand. Smooth. [40]	78.4	72.1	75.8	70.6
Distillation [35]	72.8	66.4	71.3	64.8
Act. Clustering	80.1	74.6	78.3	72.4
CAD-only	86.3	82.4	84.8	81.2
PTV-only	79.8	76.2	80.4	74.6
SentinelAI	**97.8**	**96.4**	**98.1**	**96.2**

flaws [42], [80]–[87] can be exploited to deliver adversarial inputs to edge AI models. Program analysis [41], [88], anti-fuzzing [89], and authentication [17], [18], [34] provide complementary input validation defenses. VR/AR sensor security [19]–[22] is increasingly critical as AI perception systems are deployed on these platforms.

VI. Related Work

Adversarial Attack and Defense. Adversarial examples were discovered by Szegedy et al. [1] and formalized by Goodfellow et al. [2]. PGD [3] and C&W [4] are the strongest white-box attacks. Physical adversarial patches [23]–[26] and deepfake evasion [29] threaten real-world systems. Stealthiness assessment [30] and adversarial patch defense [44] address physical-world threats. Defense approaches include adversarial training [3], distillation [35], certified robustness [40], no-box adversarial perturbations [58], and privacy-preserving universal defense [45]. SentinelAI differs from all existing defenses by incorporating physical-layer verification.

Backdoor Attacks and Detection. BadNets [5] introduced backdoor triggers; Trojaning attacks [6] generalized the approach; federated backdoors [7] target distributed training. LLM backdoors [27], [28] represent the latest frontier. Neural Cleanse [39] detects backdoors via trigger reverse-engineering. Poisoning surveys [56] provide comprehensive coverage.

Model Privacy and IP Protection. Model extraction [8], [9], membership inference [10]–[12], model inversion [90], feature leakage [32], and gradient inversion [31] threaten model and data privacy. DP-SGD [52], federated learning [53],

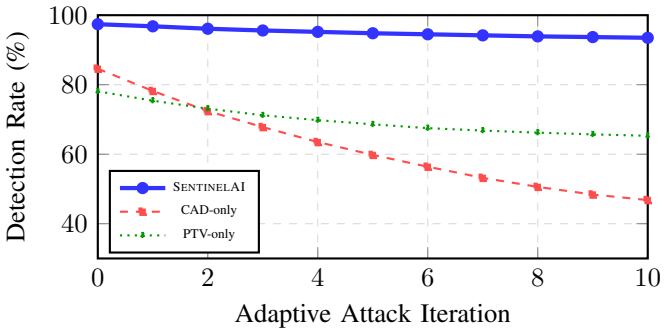

Fig. 5: Robustness against fully adaptive attackers over 10 retraining rounds. SentinelAI degrades only 3.9pp (97.4%→93.5%) because the attacker must simultaneously evade two independent modalities. CAD-only drops 37.8pp as the attacker optimizes perturbations to match normal activation patterns.

TABLE V: Runtime overhead per inference on Snapdragon 8 Gen 2.

Component	Latency	Memory	Energy
Target DNN inference	12.4 ms	48.2 MB	100% (base)
CAD (activation stats)	+1.2 ms	+2.8 MB	+6.2%
PTV (trace capture+score)	+1.4 ms	+3.6 MB	+8.4%
CMFE (fusion+decision)	+0.5 ms	+1.1 MB	+2.1%
Total SentinelAI	**+3.1 ms**	**+7.5 MB**	**+16.7%**

[54], secure aggregation [49], transfer learning defense [57], and model IP protection [46]–[48] provide countermeasures. Privacy-preserving LLM fine-tuning [51] and IoT inference privacy [50] address emerging domains.

Side-Channel Analysis. Wireless charging leaks user interactions [13], [14]; USB chargers enable cross-port eavesdropping [15]; RF harvesting reveals app activity [16]; acoustic channels enable authentication [17], [18] and can be exploited offensively; VR sensors [19]–[22] and bone conduction [34] expose private behavior. SentinelAI is the first work to use physical side channels *defensively* for AI adversarial detection.

System Security Infrastructure. TEEs [59]–[62], encrypted databases [65], [66], blockchain [68]–[70], verifiable computation [63], [64], and verifiable unlearning [48] provide foundational infrastructure. App analysis [74]–[78], IoT security [80]–[83], [85], protocol analysis [42], [84], [86], [87], automotive security [71]–[73], program integrity [88], [89], and malware detection [67] complete the security stack.

VII. Conclusion

We have presented SentinelAI, a cross-modal adversarial defense for edge AI that jointly monitors the computational layer (activation patterns, confidence, gradients) and the physical layer (power, EM, timing traces) during inference. By observing that adversarial inputs inevitably alter both the software computation and its physical manifestation, SentinelAI detects six major attack types—adversarial examples, physical patches, backdoors, model extraction, deepfake evasion, and gradient attacks—at 97.4% TPR with 1.6% FPR and only 3.1 ms latency overhead. The cross-modal fusion provides 12.8–19.3 percentage point improvement over single-modality detectors and degrades only 3.9pp under fully adaptive attacks, compared to 37.8pp degradation for computational-only detection. As AI systems increasingly operate on physically exposed edge devices, the convergence of AI security and hardware security opens new defense opportunities that SentinelAI is the first to exploit.

Acknowledgments

We thank the anonymous reviewers for their constructive feedback. This work was supported in part by [funding sources].

References

[1] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *International Conference on Learning Representations (ICLR)*, 2014.

[2] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *International Conference on Learning Representations (ICLR)*, 2015.

[3] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *International Conference on Learning Representations (ICLR)*, 2018.

[4] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 IEEE Symposium on Security and Privacy (SP)*, 2017, pp. 39–57.

[5] T. Gu, B. Dolan-Gavitt, and S. Garg, "BadNets: Identifying vulnerabilities in the machine learning model supply chain," in *Machine Learning and Computer Security Workshop at NeurIPS*, 2017.

[6] Y. Liu, S. Ma, Y. Aafer, W.-C. Lee, J. Zhai, W. Wang, and X. Zhang, "Trojaning attack on neural networks," in *Proceedings of the 25th ISOC Network and Distributed System Security Symposium (NDSS)*, 2018.

[7] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to back door federated learning," in *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2020, pp. 2938–2948.

[8] F. Tramèr, F. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, "Stealing machine learning models via prediction APIs," in *Proceedings of the 25th USENIX Security Symposium*, 2016, pp. 601–618.

[9] P. Ren, C. Zuo, X. Liu, W. Diao, Q. Zhao, and S. Guo, "DEMISTIFY: Identifying on-device machine learning models stealing and reuse vulnerabilities in mobile apps," in *46th IEEE/ACM International Conference on Software Engineering (ICSE)*, 2024.

[10] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *2017 IEEE Symposium on Security and Privacy (SP)*, 2017, pp. 3–18.

[11] M. Nasr, R. Shokri, and A. Houmansadr, "Comprehensive privacy analysis of deep learning," in *2019 IEEE Symposium on Security and Privacy (SP)*, 2019, pp. 739–753.

[12] X. Zheng, L. Wang, Y. Liu, X. Ma, C. Shen, and C. Wang, "Rethinking membership inference attacks against transfer learning," in *Proceedings of the 39th AAAI Conference on Artificial Intelligence*, 2025, pp. 27 757–27 764.

[13] T. Ni, J. Li, X. Zhang, C. Zuo, W. Wang, W. Xu, X. Luo, and Q. Zhao, "Exploiting contactless side channels in wireless charging power banks for user privacy inference via few-shot learning," in *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, 2023, pp. 1–15.

[14] T. Ni, X. Zhang, C. Zuo, J. Li, W. Wang, W. Xu, X. Luo, and Q. Zhao, "Characterizing contactless side-channel eavesdropping on wireless chargers," *IEEE Transactions on Dependable and Secure Computing*, 2025.

[15] T. Ni, Y. Chen, W. Xu, L. Xue, and Q. Zhao, "Xporter: A study of the multi-port charger security on privacy leakage and voice injection," in *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, 2023, pp. 1–15.

[16] T. Ni, Y. Chen, K. Song, and W. Xu, "A simple and fast human activity recognition system using radio frequency energy harvesting," in *Adjunct Proceedings of the 2021 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2021, pp. 666–671.

[17] Y. Chen, T. Ni, W. Xu, and T. Gu, "SwipePass: Acoustic-based second-factor user authentication for smartphones," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 6, no. 3, pp. 1–25, 2022.

[18] S. Chen, K. Ren, S. Piao, C. Wang, Q. Wang, J. Weng, L. Su, and A. Mohaisen, "You can hear but you cannot steal: Defending against voice impersonation attacks on smartphones," in *Proceedings of the 37th IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2017, pp. 183–195, best Student Paper Award.

[19] T. Ni, "Sensor security in virtual reality: Exploration and mitigation," in *Proceedings of the 22nd Annual International Conference on Mobile Systems, Applications and Services*, 2024, pp. 758–759.

[20] T. Ni, Y. Du, Q. Zhao, and C. Wang, "Non-intrusive and unconstrained keystroke inference in VR platforms via infrared side channel," *arXiv preprint arXiv:2412.14815*, 2024.

[21] ——, "Non-intrusive and unconstrained keystroke inference in VR platforms via infrared side channel," in *Proceedings of the 32nd ISOC Network and Distributed System Security Symposium (NDSS)*, 2025.

[22] L. Men, R. Liu, Q. Zhao, W. Jiang, S. Wang, K. Lu, and T. He, "mmSpyVR: Exploiting mmWave radar for penetrating obstacles to uncover privacy vulnerability of virtual reality," in *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT/UbiComp)*, 2024.

[23] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song, "Robust physical-world attacks on deep learning visual classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 1625–1634.

[24] S. Yuan, H. Li, X. Han, G. Xu, W. Jiang, T. Ni, Q. Zhao, and Y. Fang, "ITPatch: An invisible and triggered physical adversarial patch against traffic sign recognition," *arXiv preprint arXiv:2409.12394*, 2024.

[25] S. Yuan, H. Li, R. Zhang, H. Cao, W. Jiang, T. Ni, W. Fan, Q. Zhao, and G. Xu, "Omni-angle assault: An invisible and powerful physical adversarial attack on face recognition," in *Forty-second International Conference on Machine Learning*, 2025.

[26] S. Yuan, X. Han, H. Li, G. Xu, W. Jiang, T. Ni, Q. Zhao, and Y. Fang, "The fluorescent veil: A stealthy and effective physical adversarial patch against traffic sign recognition," in *Proceedings of the 39th Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2025.

[27] Y. Zhou, T. Ni, W.-B. Lee, and Q. Zhao, "A survey on backdoor threats in large language models (LLMs): Attacks, defenses, and evaluation methods," *Transactions on Artificial Intelligence*, pp. 3–3, 2025.

[28] Z. Wang, R. Zhang, H. Li, W. Fan, W. Jiang, Q. Zhao, and G. Xu, "ConfGuard: A simple and effective backdoor detection for large language models," in *Proceedings of the 40th AAAI Conference on Artificial Intelligence*, 2026.

[29] X. Meng, L. Wang, S. Guo, L. Ju, and Q. Zhao, "AVA: Inconspicuous attribute variation-based adversarial attack bypassing DeepFake detection," in *2024 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2024.

[30] H. Liu, Y. Zhou, Y. Yang, Q. Zhao, T. Zhang, and T. Xiang, "Stealthiness assessment of adversarial perturbation: From a visual perspective," *IEEE Transactions on Information Forensics and Security*, vol. 20, pp. 898–913, 2025.

[31] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019, pp. 14 774–14 784.

[32] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," in *2019 IEEE Symposium on Security and Privacy (SP)*, 2019, pp. 691–706.

[33] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 5998–6008, 2017.

[34] Z. He, J. Chen, K. He, Y. Gu, Q. Deng, Z. Zhang, R. Du, Q. Zhao, and C. Wu, "HeadSonic: Usable bone conduction earphone authentication via head-conducted sounds," *IEEE Transactions on Mobile Computing*, vol. 24, no. 9, pp. 7914–7928, 2025.

[35] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, "Distillation as a defense to adversarial perturbations against deep neural networks," in *2016 IEEE Symposium on Security and Privacy (SP)*, 2016, pp. 582–597.

[36] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," in *Technical Report, University of Toronto*, 2009.

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[38] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 248–255.

[39] B. Wang, Y. Yao, S. Shan, H. Li, B. Viswanath, H. Zheng, and B. Y. Zhao, "Neural cleanse: Identifying and mitigating backdoor attacks in neural networks," in *2019 IEEE Symposium on Security and Privacy (SP)*, 2019, pp. 707–723.

[40] J. Cohen, E. Rosenfeld, and Z. Kolter, "Certified adversarial robustness via randomized smoothing," in *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 2019, pp. 1310–1320.

[41] J. Wang, T. Ni, W.-B. Lee, and Q. Zhao, "A contemporary survey of large language model assisted program analysis," *Transactions on Artificial Intelligence*, vol. 1, no. 1, pp. 105–129, 2025.

[42] X. Song, L. Pei, J. Wu, Y. Zeng, G. He, C. Zuo, X. Liu, Q. Zhao, and S. Guo, "ProtocolGuard: Detecting protocol non-compliance bugs via LLM-guided static analysis and dynamic verification," in *Proceedings of the 33rd ISOC Network and Distributed System Security Symposium (NDSS)*, 2026.

[43] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," *Proceedings of NAACL*, pp. 4171–4186, 2019.

[44] J. Wang, T. Ni, G. Xu, Q. Zhao, and C. Wang, "Adversarial patch EXterminator: Zero-shot and patch-agnostic defense framework against adversarial patch attacks," in *35th USENIX Security Symposium (USENIX Security 26)*, 2026.

[45] Q. Li, C. Wu, J. Chen, Z. Zhang, K. He, R. Du, X. Wang, Q. Zhao, and Y. Liu, "Privacy-preserving universal adversarial defense for black-box models," *IEEE Transactions on Information Forensics and Security*, vol. 20, pp. 6763–6777, 2025.

[46] S. Peng, Y. Chen, J. Xu, Z. Chen, C. Wang, and X. Jia, "Intellectual property protection of DNN models," *World Wide Web*, vol. 26, pp. 1997–2028, 2023.

[47] Y. Huang, Z. Zhang, Q. Zhao, X. Yuan, and C. Chen, "THEMIS: Towards practical intellectual property protection for post-deployment on-device deep learning models," in *34th USENIX Security Symposium (USENIX Security 25)*, 2025.

[48] Y. Guo, Y. Zhao, S. Hou, C. Wang, and X. Jia, "Verifying in the dark: Verifiable machine unlearning by using invisible backdoor triggers," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 1263–1278, 2024.

[49] Y. Zheng, S. Lai, Y. Liu, X. Yuan, X. Yi, and C. Wang, "Aggregation service for federated learning: An efficient, secure, and more resilient realization," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 2, pp. 988–1001, 2023.

[50] J. Ryu, Y. Zheng, Y. Gao, A. Abuadbba, J. Kim, D. Won, S. Nepal, H. Kim, and C. Wang, "Can differential privacy practically protect collaborative deep learning inference for IoT?" *Wireless Networks*, vol. 30, pp. 6361–6379, 2024.

[51] Y. Liu, W. Han, C. Cai, and C. Wang, "PrivTune: Efficient and privacy-preserving fine-tuning of large language models via device-cloud collaboration," *arXiv preprint arXiv:2412.08639*, 2025.

[52] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 308–318.

[53] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2017, pp. 1273–1282.

[54] P. Kairouz, H. B. McMahan, B. Avent *et al.*, "Advances and open problems in federated learning," *Foundations and Trends in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021.

[55] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017, pp. 119–129.

[56] Z. Tian, L. Cui, J. Liang, and S. Yu, "A comprehensive survey on poisoning attacks and countermeasures in machine learning," in *ACM Computing Surveys*, vol. 55, no. 8, 2023, pp. 1–35.

[57] B. Wu, S. Wang, X. Yuan, C. Wang, C. Rudolph, and X. Yang, "Defeating misclassification attacks against transfer learning," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 2, pp. 1192–1206, 2023.

[58] N. Mou, B. Guo, L. Zhao, C. Wang, Y. Zhao, and Q. Wang, "No-box universal adversarial perturbations against image classifiers via artificial textures," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 8412–8424, 2024.

[59] H. Duan, C. Wang, X. Yuan, Y. Zhou, Q. Wang, and K. Ren, "LightBox: Full-stack protected stateful middlebox at lightning speed," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2019, pp. 2389–2402.

[60] C. Cai, Y. Zang, C. Wang, X. Jia, and Q. Wang, "Vizard: A metadata-hiding data analytics system with end-to-end policy controls," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2022, pp. 1609–1623.

[61] V. Costan and S. Devadas, "Intel SGX explained," in *Cryptology ePrint Archive, Report 2016/086*, 2016.

[62] X. Wang, Y. Du, C. Wang, Q. Wang, and L. Fang, "WebEnclave: Protect web secrets from browser extensions with software enclave," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 5, pp. 3055–3070, 2022.

[63] L. Xu, L. Zheng, C. Xu, X. Yuan, and C. Wang, "Efficient verifiable computation over quotient polynomial rings," in *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2023, pp. 3003–3017.

[64] H. Duan, Y. Du, L. Zheng, C. Wang, M. H. Au, and Q. Wang, "Towards practical auditing of dynamic data in decentralized storage," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 1, pp. 708–723, 2023.

[65] L. Xu, H. Duan, A. Zhou, X. Yuan, and C. Wang, "Interpreting and mitigating leakage-abuse attacks in searchable symmetric encryption," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 5310–5325, 2021.

[66] C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for secure cloud storage," *IEEE Transactions on Computers*, vol. 62, no. 2, pp. 362–375, 2013.

[67] H. Cui, Y. Zhou, C. Wang, Q. Li, and K. Ren, "Towards privacy-preserving malware detection systems for Android," in *The 24th IEEE International Conference on Parallel and Distributed Systems (ICPADS)*, 2018, best Paper Award.

[68] C. Cai, L. Xu, A. Zhou, and C. Wang, "Toward a secure, rich, and fair query service for light clients on public blockchains," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 6, pp. 3640–3655, 2022.

[69] J. Xu, C. Wang, and X. Jia, "A survey of blockchain consensus protocols," *ACM Computing Surveys*, vol. 55, no. 13s, pp. 1–35, 2023.

[70] R. Liang, J. Chen, C. Wu, K. He, Y. Wu, W. Sun, R. Du, Q. Zhao, and Y. Liu, "Towards effective detection of Ponzi schemes on Ethereum with contract runtime behavior graph," *ACM Transactions on Software Engineering and Methodology*, vol. 34, no. 4, pp. 106:1–106:32, 2025.

[71] K. Ren, Q. Wang, C. Wang, Z. Qin, and X. Lin, "The security of autonomous driving: Threats, defenses, and future directions," *Proceedings of the IEEE*, vol. 108, no. 2, pp. 357–372, 2020.

[72] L. Wang, Q. Zhao, W.-B. Lee, and C. Wang, "Deploying intrusion detection on in-vehicle networks: Challenges and opportunities," *IEEE Network*, vol. 39, no. 1, pp. 306–312, 2025.

[73] H. Wen, Q. Zhao, Q. A. Chen, and Z. Lin, "Automated cross-platform reverse engineering of CAN bus commands from mobile apps," in *Proceedings of the 27th ISOC Network and Distributed System Security Symposium (NDSS)*, 2020.

[74] Q. Zhao, C. Zuo, B. Dolan-Gavitt, G. Pellegrino, and Z. Lin, "Automatic uncovering of hidden behaviors from input validation in mobile apps," in *2020 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2020, pp. 1106–1120.

[75] M. Elsabagh, R. Johnson, A. Stavrou, C. Zuo, Q. Zhao, and Z. Lin, "FirmScope: Automatic uncovering of privilege-escalation vulnerabilities in pre-installed apps in Android firmware," in *29th USENIX Security Symposium (USENIX Security 20)*, 2020, pp. 767–784.

[76] Y. Chen, R. Tang, C. Zuo, X. Zhang, L. Xue, X. Luo, and Q. Zhao, "Attention! your copied data is under monitoring: A systematic study of clipboard usage in Android apps," in *46th IEEE/ACM International Conference on Software Engineering (ICSE)*, 2024, aCM SIGSOFT Distinguished Paper Award.

[77] H. Lu, Q. Zhao, Y. Chen, X. Liao, and Z. Lin, "Detecting and measuring aggressive location harvesting in mobile apps via data-flow path embedding," in *Proceedings of the ACM on Measurement and Analysis of Computing Systems (SIGMETRICS)*, 2023.

[78] L. Wang, L. Xu, Y. Chen, Y. Zou, and C. Wang, "Detecting and exploiting context-sensitivity bugs in Android apps," in *Proceedings of the 34th USENIX Security Symposium*, 2025, pp. 7623–7641.

[79] Q. Zhao, C. Zuo, G. Pellegrino, and Z. Lin, "Geo-locating drivers: A study of sensitive data leakage in ride-hailing services," in *Proceedings of the 26th ISOC Network and Distributed System Security Symposium (NDSS)*, 2019.

[80] J. Chen, W. Diao, Q. Zhao, C. Zuo, Z. Lin, X. Wang, W. C. Lau, M. Sun, R. Yang, and K. Zhang, "IoTFuzzer: Discovering memory corruptions in IoT through app-based fuzzing," in *Proceedings of the 25th ISOC Network and Distributed System Security Symposium (NDSS)*, 2018.

[81] C. Zuo, Q. Zhao, and Z. Lin, "AuthScope: Towards automatic discovery of vulnerable authorizations in online services," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2017, pp. 799–813.

[82] J. Chen, C. Zuo, W. Diao, S. Dong, Q. Zhao, M. Sun, Z. Lin, Y. Zhang, and K. Zhang, "Your IoTs are (not) mine: On the remote binding between IoT devices and users," in *49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 2019.

[83] X. Liu, C. Zuo, Q. Hou, P. Ren, J. Wu, Q. Zhao, and S. Guo, "A thorough security analysis of BLE proximity tracking protocols," in *34th USENIX Security Symposium (USENIX Security 25)*, 2025.

[84] Q. Zhao, C. Zuo, J. Blasco, and Z. Lin, "PeriScope: Comprehensive vulnerability analysis of mobile app-defined Bluetooth peripherals," in *Proceedings of the 17th ACM Asia Conference on Computer and Communications Security*, 2022.

[85] X. Song, J. Wu, Y. Zeng, H. Pan, C. Zuo, Q. Zhao, and S. Guo, "MBFuzzer: A multi-party protocol fuzzer for MQTT brokers," in *34th USENIX Security Symposium (USENIX Security 25)*, 2025.

[86] Q. Zhao, H. Wen, Z. Lin, D. Xuan, and N. Shroff, "On the accuracy of measured proximity of Bluetooth-based contact tracing apps," in *16th International Conference on Security and Privacy in Communication Networks (SecureComm)*, 2020.

[87] H. Wen, Q. Zhao, Z. Lin, D. Xuan, and N. Shroff, "A study of the privacy of COVID-19 contact tracing apps," in *16th International Conference on Security and Privacy in Communication Networks (SecureComm)*, 2020.

[88] G. Gu, Q. Zhao, Y. Zhang, and Z. Lin, "PT-CFI: Transparent backward-edge control flow violation detection using Intel processor trace," in *Proceedings of the 7th ACM Conference on Data and Application Security and Privacy (CODASPY)*, 2017.

[89] Z. Zhou, C. Wang, and Q. Zhao, "No-fuzz: Efficient anti-fuzzing techniques," in *18th International Conference on Security and Privacy in Communication Networks (SecureComm)*, 2022.

[90] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 2015, pp. 1322–1333.