Adversarial Attack Bypass by Stochastic Computing

Faeze S. Banitaba, *Graduate Student Member, IEEE*, Sercan Aygun, *Senior Member, IEEE*, Mehran Shoushtari Moghadam, *Graduate Student Member, IEEE*, Amirhossein Jalilvand, Bingzhe Li, *Member, IEEE*, and M. Hassan Najafi, *Senior Member, IEEE*

***Abstract*—Deep learning excels by utilizing vast datasets and sophisticated training algorithms. It achieves superior performance across many machine learning challenges compared to traditional methods. However, deep neural networks (DNNs) are not flawless; they are particularly susceptible to adversarial samples during the inference phase. These inputs area deliberately designed by attackers to cause DNNs to make incorrect classifications, exploiting the networks' vulnerabilities. This letter proposes a novel perspective to fortify the neural network (NN) defense against adversarial attacks. We enhance the NN security by employing an emerging model of computation, namely, stochastic computing (SC). We show that strengthening NN with SC counteracts the adverse effects of these attacks on an NN output and adds a vital defense layer. Our evaluation results reveal that SC notably increases NN robustness and decreases susceptibility to interference, creating secure, reliable NN systems. The proposed method improves accuracy and reduces hardware footprint and energy consumption by up to 85%, 88%, and 95%, respectively.**

***Index Terms*—Adversarial attack, low-cost hardware design, secure neural networks (NNs), stochastic computing (SC).**

I. Introduction

RAPID deployment of neural networks (NNs) in various industries has led to significant advancements in data analysis, pattern identification, and strategic decision-making [1]. However, integrating these technologies into critical sectors presents a significant challenge: protecting these systems from sophisticated adversarial attacks designed to compromise NN output accuracy and reliability. Depending on the attack features, the attacker's level of access to network specifications varies. With this knowledge, the attacker manipulates the data before it enters the network for classification. The goal is to make minimal changes to the input data, which would result in misclassification by the network. While some strategies aim to neutralize attacks by understanding and reversing their effects or by blocking the attacker's influence [2], the proposed solution of this letter takes a different approach.

We demonstrate that, with modifications to the computational units, the network can effectively operate in the presence of adversarial attacks, where an attacker manipulates input data prior to classification, and achieves reliable performance both under attack and in normal conditions.

This innovative idea proposes a new approach, designing networks to maintain performance against adversarial actions by emphasizing resilience and adaptability instead of confrontation or avoidance.

Stochastic computing (SC) is an emerging computational model with high error tolerance in computations and data representation [3]. This letter examines the effectiveness of SC as a protective strategy against adversarial threats, which is particularly important for industries where data accuracy and integrity are crucial. In essence, this letter provides insights into how SC can improve the security of NN, representing a significant advancement in the ongoing effort to create resilient artificial intelligence (AI) systems. The main contributions of this letter are as follows.

1) We adopt quasi-random SC with high accuracy in a single iteration in contrast to the prior SC solutions with pseudo-random sequences and multi-iterations for optimum accuracy.
2) We show that using SC for only the initial layers significantly improves the network's robustness.
3) Our approach ensures accurate outcomes even if successful attacks occur, bypassing the need for the system to detect or block attackers' data manipulations.
4) Our method is adaptable to both simple and complex networks, making it a versatile defense strategy.
5) Our method shows energy efficiency, reduced area footprint, and optimized system performance, useful for applications with limited resources.
6) Our approach improves NN accuracy against adversarial attacks without requiring the detection of them, performing effectively in various scenarios for reliable performance.

II. Adversarial Attacks

NNs have received significant attention in recent years and, as a result, have increasingly become targets for adversarial attacks. Classification NNs, a versatile subset of machine learning models, are designed to classify input data into some predefined classes. These networks, built upon layers

Received 4 December 2024; revised 9 January 2025; accepted 22 January 2025. Date of publication 4 February 2025; date of current version 15 August 2025. This work was supported in part by the National Science Foundation (NSF) under Grant 2019511 and Grant 2339701, and in part by Nvidia. This manuscript was recommended for publication by F. Merchant. *(Corresponding author: Faeze S. Banitaba.)*

Faeze S. Banitaba, Sercan Aygun, and Amirhossein Jalilvand are with the School of Computing and Informatics, University of Louisiana at Lafayette, Lafayette, LA 70504 USA (e-mail: faeze.banitaba@louisiana.edu).

Mehran Shoushtari Moghadam and M. Hassan Najafi are with the Electrical, Computer, and Systems Engineering Department, Case Western Reserve University, Cleveland, OH 44106 USA.

Bingzhe Li is with the Computer Science Department, University of Texas at Dallas, Richardson, TX 75080 USA.

Digital Object Identifier 10.1109/LES.2025.3538552

of interconnected nodes, learn to classify from training data. Their versatility, demonstrated in various applications, from alternative energy resources [4] to communications [5] makes them popular. While these NNs are powerful for data analysis and interpretation, they are vulnerable to attacks [6]. The attacks can subtly manipulate input data, causing the network to misclassify. For example, an image slightly altered at the pixel level could be wrongly identified, or a text with minor alterations might be misinterpreted.

The evolution of *adversarial attacks* is driven by a constant interplay between the development of sophisticated attack techniques and the establishment of robust defenses. This dynamic has led to the development of a wide range of attack strategies, each designed to test NNs in different situations [7], [8]. Adversarial attacks on deep learning systems are typically analyzed by focusing on specific key characteristics. In what follows, we discuss two critical attributes that define and differentiate these attacks.

A. Glass-Box Versus Closed-Box

The nature of the adversarial attack changes depending on how much access the attacker has to the target model. In Glass-Box attacks, adversaries possess full knowledge about the network, including its structure, parameters, weights, and how it processes information. This knowledge allows them to create precisely engineered inputs that can mislead the network, exploiting their deep understanding to conduct highly effective attacks. Closed-Box attacks, in contrast, occur when adversaries have no insight into the network's internal workings, which means attackers lack access to internal information about the prey model, including gradient information. They can only observe NN inputs and outputs, using assumption and iterative techniques to craft deceptive inputs. Existing closed-box adversarial attacks primarily rely on query-based, transfer-based, and meta-learning-based strategies to obtain output results from the prey model. Consequently, generating effective attacks becomes more challenging. Although Closed-Box attacks may seem less threatening due to the limited information available to the attacker, they are more common and realistic in everyday scenarios. Adversaries typically operate with constraints, making Closed-Box attacks a prevalent challenge for real-world NN applications [9].

B. Targeted Versus Untargeted

In a targeted attack, the adversary meticulously crafts disturbances to mislead the model into categorizing an input with a specific, erroneous label. This attack is strategic, with the attacker's goal to manipulate the model's output to match a predetermined incorrect category. Conversely, untargeted attacks are not about achieving a specific misclassification; instead, the attacker's sole objective is to ensure the model fails to assign the correct label. The exact nature of mislabeling is irrelevant in untargeted attacks as long as the outcome deviates from the truth. This dichotomy highlights the varied strategies attackers employ to compromise model integrity, ranging from focused precision to broad disruption [6].

III. Adversarial Defenses

This section briefly explains state-of-the-art (SOTA) research in defending against attacks and discusses their drawbacks. *Adversarial training* is a defense method that enhances an NN's robustness by exposing it to adversarial samples during training. Many recent studies indicate this method as one of the most effective defenses. It could improve a model's resistance to specific types of adversarial attacks. However, it has some notable drawbacks, such as potential tradeoffs in model performance and challenges in generalization and scalability. Most importantly, it incurs substantial computational costs, making adversarial training impractically expensive for deployment [10]. Some other recent defense mechanisms use *randomization* schemes to counteract the effects of adversarial perturbations in the input or feature domain. Defenses based on randomization have shown similar effectiveness in Closed-Box scenarios, providing reassurance about their applicability in such scenarios. This potential effectiveness in Closed-Box scenarios is a reason for optimism, but not in Glass-Box scenarios [11].

Denoising-based is another defense method group that removes added noise from input data. It is a straightforward approach for mitigating adversarial perturbations. Methods in this group include but are not limited to, feature squeezing adversarial detection and gradient masking/obfuscation. Nevertheless, as demonstrated in [2], it is susceptible to adaptive and knowledgeable adversaries and may not offer a strong defense against certain attacks [10], [12]. Another group of defenses theoretically guarantees the error rate, assuming heuristic methods might be broken by a new attack in the future because their effectiveness is only experimentally validated. These methods can maintain a certain accuracy under a well-defined class of attacks. However, they fail to provide the same high level of accuracy as heuristic methods [13]. To wrap it all up, regarding effectiveness, adversarial training demonstrates the best performance but at a significant computation cost. On the other hand, regarding efficiency, recent works show that randomization and denoising-based mechanisms are not as effective as they claim to be [12]. At the same time, theoretically proving methods are far from meeting the practical requirements (in both their accuracy and their efficiency). Therefore, we still need a defense strategy that could balance *effectiveness* and *efficiency*.

In what follows, we propose a hardware-based method that achieves high accuracy against Glass-Box attacks with a fixed computation cost for all networks.

IV. SC Against Adversarial Attacks

The exploration of defense mechanisms extends beyond the adversarial spectrum, encompassing the compromise and exploitation of complex machine-learning models. SC emerges as a potential safeguard in this intricate space, offering high resilience. SC is a computational model representing and processing data in uniform random bit-streams (rather than traditional binary formats). This unconventional computing model encodes values as the probability of observing a "1" in a random sequence of "1"s and "0"s. SC is known for its

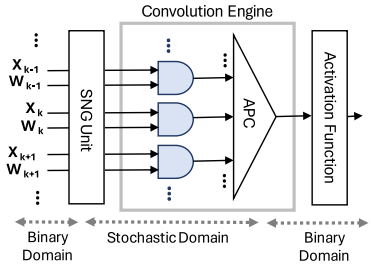

Fig. 1. Enhancing convolution engine with SC (SNG: Stochastic number generator and APC: Approximate parallel counter).

TABLE I
CLASSIFICATION PERFORMANCE (%) AFTER ATTACK: ✢ → LENET-5, FASHION-MNIST, FGSM / ✪ → LENET-5, MNIST-C, FGSM / ✧ → LENET-5, MNIST, FGSM / ★ → LENET-5, MNIST, CW / ☆ → RESNET-20, CIFAR-10, CW

Design Approach		Correct Prediction	Attacker's Success	Any Other Predictions
✢	Binary (8 bit)	8.6	91.4	-
	SC (N=128)	**41.4**	58.6	-
✪	Binary (8 bit)	20.8	79.2	-
	SC (N=16)	**51.2**	48.8	-
✧	Binary (8 bit)	42.8	57.2	-
	SC (N=32)	**93.0**	7.0	-
★	Binary (8 bit)	5.8	10.3	83.9
	SC (N=16)	**79.0**	8.8	10.9
☆	Binary (8 bit)	0.1	99.3	0.6
	SC (N=256)	**84.7**	2.9	12.4

$\epsilon = 0.3$ for FGSM || train/test/validation splits: (55,000/5,000/5,000) → MNIST, Fashion-MNIST, MNIST-C (45,000/4,000/4,000) → CIFAR-10

high fault tolerance and cost efficiency [3] offering a unique advantage in dealing with noisy or incomplete data and in scenarios where precision can be traded for lower power consumption and hardware simplicity.

This letter focuses on two sophisticated attack strategies. The first one is a *Glass-Box untargeted* attack. The attacker utilizes the gradient of the loss function to alter the data, resulting in misclassification [14]. The second one combines the *targeted* approach with *Glass-Box* knowledge. This combination represents one of the most formidable challenges in maintaining network robustness. In these scenarios, the attacker aims to manipulate the network's classification output and possesses complete knowledge of the network's architecture, parameters, and gradients (Glass-Box). This level of understanding allows the attacker to craft highly effective adversarial examples, making it crucial for defense mechanisms to be remarkably resilient. Proving the system's robustness against the second attack also implies its reliability against other types of attacks [15].

A. Proposed Architecture

In NNs, the first convolutional layers contribute significantly to the total area and power costs due to the *row-column* nature of the process and the input data. The fundamental operation within these layers is dot-product, which multiplies input data by the weights and then accumulates the results. An activation function typically follows this to introduce nonlinearity and help the network learn complex patterns. This letter explores the idea that by integrating SC in inference, we can significantly enhance the network's resilience to adversarial attacks. As illustrated in Fig. 1, we propose to perform multiplication operations, particularly in the first convolutional layers, in the SC domain. This can be achieved by simple bit-wise AND operations, avoiding the complexity and cost associated with traditional binary multipliers [16]. We exploit SOTA *quasi-random* bit-streams for accurate multiplication with short bit-streams [17]. For the first time, this letter exploits deterministic quasi-random sequences to enhance the robustness of NNs against adversarial attacks with SC. Our approach simplifies the computation process, enhancing efficiency and saving computation resources. Quasi-random sequences (e.g., Sobol sequences) offer shorter processing time, lower energy consumption, and higher accuracy compared to traditional linear feedback shift register (LFSR)-based sequences due to their fast converging and low-discrepancy nature [17], [18]. Unlike pseudo-random sequences that need multiple runs for maximum possible accuracy, quasi-random sequences obtain it with a single run. Our solution maintains the accuracy of NNs even when they are under attack, such as from adversarial threats, thereby ensuring the system's security.

Prior works showed that the savings from the simple computation logic (e.g., AND gate for multiplication) could well compensate for the overhead cost of converting data to bit-stream format [19]. Our approach leaves the training phase of the network untouched and only replaces the multiplication operations in the inference phase. By simply modifying the convolution unit in the chosen layers, we enhance the network's robustness during the testing stage without revisiting the initial training operations. This strategy ensures that existing NNs can be easily adapted to be more resilient against attacks without requiring comprehensive retraining or significant architectural alterations.

B. Implementation

We explore how SC can augment the network's ability to tolerate adversarial manipulations. First, we observe the attacker's performance on unenhanced NN to assess its impact and severity. Then, we apply the proposed defense mechanism to the LeNet-5 as a simpler and to the ResNet-20 as a more complex NN architecture. To evaluate the robustness against attacks, we simulate two attacks.

1) Fast gradient sign method (FGSM), which slightly alters the input by introducing a small, intentional noise (epsilon, ϵ) that follows the path of the model's loss gradient. The goal is to trick the model into making a wrong prediction [14]. In our implementations, we used $\epsilon = 0.3$.
2) The Carlini and Wagner (CW) attack [15] as a more complex targeted L2 attack. This attack is grounded in one of the most common distance metrics for crafting adversarial examples. The L2 distance metric quantifies the Euclidean distance between the original input (x) and its perturbed version (x') that is close in terms of input space but misclassified in a different class. The L2-norm attack exploits this metric to identify and generate x'. The CW attacks achieve 95%–100% attack success rate on naturally trained deep NNs (DNNs) for MNIST and CIFAR-10 [15].

In our experiments, we assume the adversary has full knowledge of the NN, including its architecture and parameters, emulating a Glass-Box attack. The expectation is that a defense

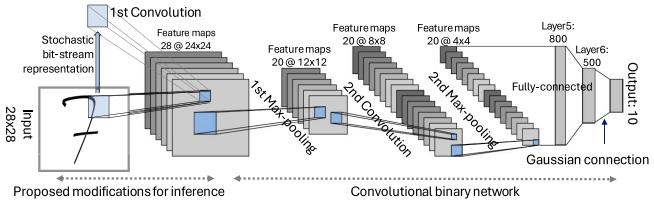

Fig. 2. Proposed robust NN setup for LeNet-5 model; the first convolution layer is equipped with our model, transitioning data to the stochastic domain, followed by SC multiplications.

mechanism effective against the L2 attack will likely be capable of counteracting other adversarial strategies.

Table I presents performance measurements where correct predictions denote the samples successfully classified by the NN after the attacker manipulates the data. The attacker's success rate indicates the degree to which the attacker achieved its goal (its target label for targeted attack or any misclassifications for untargeted one). Lastly, the last column displays the number of misclassifications, though these instances did not align with the adversary's intended target label. Hence, for FGSM attack, as an untargetted attack, this column is not applicable. Fig. 2 shows the NN setup for the LeNet-5 model, which we used with the MNIST, fashion-MNIST, and MNIST-C [18] datasets. This model comprises five layers, including two convolution and three dense layers. After each layer, an activation function is applied. The first two utilize the rectified linear unit (ReLU) activation, while the final layer employs the "Softmax" function for output normalization. For bit-stream generation, Sobol sequences are employed. We convert data from binary to quasi-random bit-streams using a stochastic number generator (SNG) unit in the targeted convolution layer. We vary the bit-stream lengths as needed. The multiplication operations are performed in the target layer by bit-wise AND operation between the bit-streams. All other operations within the network are conducted as per standard procedures. We further extend our implementation to a more complex model, ResNet-20, classifying the CIFAR-10 dataset. Our proposed architecture demonstrates the capability to *thwart attacks* across all tested network configurations. Our evaluation results demonstrate our architectural modifications significantly enhance the network's resilience against both attacks for all four datasets.

C. Results

Random number generation and the quality of random bit-streams are fundamental to the performance of SC systems. We employ quasi-random bit-streams for optimum energy efficiency, accuracy, and robustness, distinguishing our approach from prior pseudo-random SC techniques. We conduct a series of experiments to gauge the impact of employing quasi-random SC under attack stress for two NN models. Both networks were deliberately compromised throughout the tests to achieve almost zero accuracy while simulating a successful CW attack and degrade to less than half accuracy while simulating an FGSM attack. Any deviation from this state, leading to a restoration of accuracy to ideal values, shows the success of the defense mechanism in shielding against adversarial attacks. Table II presents the classification performance of LeNet-5 on the MNIST dataset before and after two attacks. Prior to the attack, the presence of SC did not affect the results significantly, achieving comparable accuracy to that of the conventional non-SC implementation (99%). We assessed the involvement of SC in the first and second layers independently. We observed that employing SC in the second layer does not yield advantageous outcomes, particularly for smaller bit-stream sizes. After the attack, substantial improvement is observed in the reported classification numbers, with the accuracy reaching up to 79% and 93%, compensating for the initial 0.1% and 42.8% accuracy resulting from the CW and FGSM attacker's success across the entire test set, respectively. Following our evaluation of LeNet-5, we explore ResNet-20 model, a notably more intricate architecture. Table III reports the SC's impact on this model. We note that we need longer bit-streams (e.g., $N = 256$) to achieve conventional network (non-SC) accuracy, which is 92%, with this model. As can be seen, SC proves to be effective to defend the attack across all bit-stream sizes, efficiently mitigating the effects of malicious alterations on the network. We observe restored accuracy levels up to 85%, healing from the worst-case scenario of 0% accuracy resulting from the attacker's full success on the non-SC model. Compared to SOTA, our solution demonstrates a misclassification rate of 15% in a safeguarded version using SC in the presence of an attack. In contrast, the SOTA approach [19] exhibits a 79.1% misclassification rate for the same CIFAR-10 dataset. Our proposed method further provides adaptability to any NN and scalability to any dataset size due to its minimum network modification.

As it can be seen in Table II, when the bit-stream length gets larger, and SC is applied to the first convolutional layer, the accuracy results of the network before the attack become increasingly similar to those of the baseline binary network. This trend arises from the inherent bit-stream-based nature of the process: as the length of the bit stream increases, the data becomes more representative of real data, thereby reducing errors and approaching the accuracy levels of the binary network. The same trend persists when SC is applied to the second layer of the network. In fact, the difference in accuracy numbers becomes even more pronounced in this

TABLE II
LENET-5 CLASSIFICATION ACCURACY (%) FOR MNIST DATASET, WHEN SC APPLIED TO THE 1ST OR 2ND CONVOLUTION LAYERS (N: BIT-STREAM LENGTH)

N	Accuracy (No Attack)		Accuracy (CW Attack)		Accuracy (FGSM Attack)	
	1st Conv. in SC	2nd Conv. in SC	1st Conv. in SC	2nd Conv. in SC	1st Conv. in SC	2nd Conv. in SC
8	92.8	56.0	69.0	28.0	84.4	21.0
16	96.4	79.0	**79.0**	36.0	92.2	45.8
32	98.0	93.0	61.0	41.0	**93.0**	52.2
64	98.1	97.0	57.0	39.0	91.4	86.6
128	98.6	97.9	55.0	33.1	91.4	88.0
256	98.9	98.2	52.0	32.9	90.0	87.4
512	99.0	98.4	49.1	33.1	91.0	86.2

TABLE III
ACCURACY (%) OF RESNET-20 CLASSIFICATION FOR CIFAR-10 (1ST CONV. IN SC DOMAIN)

Accuracy	N=8	16	32	64	128	256	512
No Attack	31	53	72	88	87	90	90
CW Attack	32	65	72	83	83	**85**	84

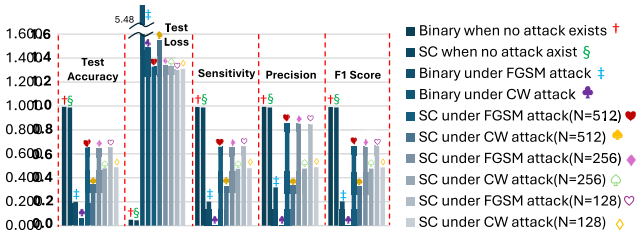

Fig. 3. Comparison of classification metrics: traditional binary versus SC-based network under two adversarial attacks: FGSM attack and CW attack—using LeNet-5 and MNIST dataset (N: bit-stream length).

TABLE IV
SYNTHESIS RESULTS OF CONVOLUTION LAYER (24 × 24 CONVOLUTION ENGINES + BIT-STREAM GENERATORS)

Bit-width (M)	Design Approach	Area (μm^2)	CPL✦ (ns)	Power✱ (W)	Energy/ cycle (nJ)
5	Binary	5,736,960	1.51	7.14	10.79
	SC	1,455,553	1.04	0.37	0.39
6	Binary	9,158,400	1.55	8.52	13.21
	SC	1,521,019	1.05	0.49	0.51
7	Binary	13,570,560	1.88	12.44	23.39
	SC	1,589,313	1.06	0.54	0.58
8	Binary	14,768,640	1.97	13.59	26.78
	SC	1,675,835	1.07	0.74	0.79

✦: Critical Path Latency ‖ ✱: Power at maximum frequency.

case. We observed that applying SC to the first layer has the most significant impact compared to applying the same modification to other convolutional layers. We demonstrate this by presenting the results from the initial layers of the LeNet-5. Extending evaluations to a more intricate network like ResNet-20 also validated our observation that employing SC in the first layer yields the most favorable outcome.

As we can see in Table II, when the attacker manipulates the input data while we utilize our modified network with stochastic convolution, a decrease in accuracy is observed as the bit-stream length approaches 512 bits. This decline can be attributed to the same principle discussed earlier: by increasing the length, the data representation becomes more akin to that of a binary presentation, resulting in accuracy levels converging toward those of the binary network. Conversely, reducing the bit-stream length and getting closer to 8 bits leads to a loss in data precision, consequently causing a drop in accuracy once more. Ultimately, there exists an optimal point for the bit-stream length where the most favorable outcome is achieved. The same trends can also be seen in the numbers reported in Table III. Table IV illustrates the synthesis results of the initial convolution layer. In this layer, every pixel of an input image undergoes parallel processing with a single filter. It concurrently implements 24×24 convolution units, using both traditional weighted binary methods (multiplication and addition in binary) and SC-based approach, as illustrated in Fig. 3. We utilized Synopsys Design Compiler with a 45nm gate library [20] to synthesize the designs. Additionally, we employed the Sobol sequences [17] to generate high-quality bit-streams in SC-based design. As observed, due to the simple operations in the SC domain, the implementation cost is greatly reduced. For instance, when implementing the first convolution layer with a 7-bit fixed-point precision ($M = 7$), the SC-based design notably decreases the hardware area footprint by 88%. Regarding power consumption, the SC design achieves a 95% reduction compared to the conventional weighted binary design. The critical path latency (CPL) of the SC-based design is 1.06 ns, compared to 1.88 ns for the weighted binary approach. The table also presents the energy per cycle of the proposed design, which is the product of power consumption by the CPL. For $M = 7$, the SC design saves energy per cycle by 97%. It is worth noting that the energy consumption value needs to be multiplied by the length of the bit-streams in the SC design. For example, with $M = 7$, $0.58 \times 128 = 73.76$. However, high accuracy can mostly be achieved by short bit-streams that minimize the energy.

V. CONCLUSION, LIMITATIONS, AND FUTURE WORK

This letter proposes an SC layer as an effective defense mechanism to strengthen NNs against adversarial attacks. The enhanced network is accurate and reliable whether adversarial challenges exist or not. Even if attackers have already penetrated the network, it is not too late to implement a lightweight SC layer to bypass malicious activities by intruders. Our evaluation results reveal that SC notably increases NN robustness and decreases susceptibility to interference, creating secure, reliable NN systems. This letter focuses on developing a low-power, hardware-efficient NN architecture while enhancing system security against adversarial attacks. For high accuracy and energy efficiency, our solution is applied to the early layers of the network. For other layers, longer bit-streams are needed to maintain the accuracy, which as a result, increases the latency and energy consumption. The experiments conducted primarily target classification-based tasks using standard datasets. However, these datasets may not fully represent the complexity of applications in critical sectors, especially those with specialized architectures and domain-specific considerations. A broader exploration of these safety-critical applications is necessary to evaluate the performance and robustness of the proposed approach under more diverse and demanding scenarios. Future work will focus on extending the framework to include more representative of such domains, alongside further analysis of the system's limitations and potential improvements.

REFERENCES

[1] M. S. Vahdatpour and Y. Zhang, "Latency-based motion detection in spiking neural networks," *Int. J. Cogn. Lang. Sci.*, vol. 18, no. 3, pp. 150–155, 2024.
[2] W. He, J. Wei, X. Chen, N. Carlini, and D. Song, "Adversarial example defense: Ensembles of weak defenses are not strong," in *Proc. 11th USENIX Workshop Offensive Technol.*, 2017, pp. 1–11.
[3] A. Alaghi and J. P. Hayes, "Survey of stochastic computing," *ACM Trans. Embed. Comput. Syst.*, vol. 12, no. 2s, pp. 1–19, 2013.
[4] K. Sheida, M. Seyedi, and F. Ferdowsi, "Adaptive voltage and frequency regulation for secondary control via reinforcement learning for islanded microgrids," in *Proc. IEEE TPEC*, 2024, pp. 1–6.
[5] L. Kouhalvandi, O. Ceylan, and S. Ozoguz, "Automated deep neural learning-based optimization for high performance high power amplifier designs," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 67, no. 12, pp. 4420–4433, Dec. 2020.
[6] K. Ren, T. Zheng, Z. Qin, and X. Liu, "Adversarial attacks and defenses in deep learning," *Engineering*, vol. 6, no. 3, pp. 346–360, 2020.
[7] N. Nazari et al., "Adversarial attacks against machine learning-based resource provisioning systems," *IEEE Micro*, vol. 43, no. 5, pp. 35–44, Sep./Oct. 2023.

[8] A. Roohi and S. Angizi, "Efficient targeted bit-flip attack against the local binary pattern network," in *Proc. IEEE Int. Symp. Hardw. Oriented Security Trust (HOST)*, 2022, pp. 89–92.
[9] A. Thangaraju and C. Merkel, "Exploring adversarial attacks and defenses in deep learning," in *Proc. IEEE CONECCT*, 2022, pp. 1–6.
[10] C. Xie, Y. Wu, L. Maaten, A. L. Yuille, and K. He, "Feature denoising for improving adversarial robustness," in *Proc. IEEE/CVF CVPR*, 2019, pp. 501–509.
[11] A. Athalye, L. Engstrom, A. Ilyas, and K. Kwok, "Synthesizing robust adversarial examples," in *Proc. 35th Int. Conf. Mach. Learn.*, 2018, pp. 284–293.
[12] A. Athalye, N. Carlini, and D. Wagner, "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples," in *Proc. 35th ICML*, 2018, pp. 274–283.
[13] A. Raghunathan, J. Steinhardt, and P. Liang, "Semidefinite relaxations for certifying robustness to adversarial examples," in *Proc. 32nd Adv. Neural Inf. Syst.*, 2018, pp. 1–11.
[14] 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.
[15] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *Proc. IEEE SP*, 2017, pp. 39–57.
[16] S. R. Faraji, M. H. Najafi, B. Li, D. J. Lilja, and K. Bazargan, "Energy-efficient convolutional neural networks with deterministic bit-stream processing," in *Proc. DATE*, 2019, pp. 1757–1762.
[17] M. H. Najafi, D. Jenson, D. J. Lilja, and M. D. Riedel, "Performing stochastic computation deterministically," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 27, no. 12, pp. 2925–2938, Dec. 2019.
[18] N. Mu and J. Gilmer, "MNIST-C: A robustness benchmark for computer vision," 2019, *arXiv:1906.02337*.
[19] F. Neugebauer, V. Vekariya, I. Polian, and J. P. Hayes, "Stochastic computing as a defence against adversarial attacks," in *Proc. 53rd IFIP DSN-W*, 2023, pp. 191–194.
[20] "NCSU FreePDK 45nm library." Accessed: Dec. 29, 2022. [Online]. Available: https://eda.ncsu.edu/freepdk/freepdk45/