Privacy-by-Sensing with Time-domain Differentially-Private Compressed Sensing

Jianbo Liu, Boyang Cheng, Pengyu Zeng, Steven Davis, Muya Chang, Ningyuan Cao
Department of Electrical Engineering, University of Notre Dame, Notre Dame, South Bend, 46556

***Abstract*—With the ubiquitous IoT sensors and enormous real-time data generation, data privacy is becoming a critical societal concern. State-of-the-art privacy protection methods all demand significant hardware overhead due to computation-insensitive algorithms and divided sensor/security architecture. In this paper, we propose a generic time-domain circuit architecture that protects raw data by enabling a differentially-private compressed sensing (DP-CS) algorithm secured by physical unclonable functions (PUF). To address privacy concerns and hardware overhead at the same time, a robust unified PUF and time-domain mixed-signal (TD-MS) module are designed, where PUF enables private and secure entropy generation. To evaluate the proposed design against a digital baseline, we performed experiments based on synthesized circuits and SPICE simulation and measured a $2.9\times$ area reduction and $3.2\times$ energy gains. We also measured high-quality PUF generation with TD-MS circuit with a inter-die Hamming distance of 52% and a low intra-die Hamming distance of 2.8%. Furthermore, we performed attack and algorithm performance measurements demonstrating the proposed design preserves data privacy even under attack, and the machine learning performance has minimal degradation (within 2%) compared to the digital baseline.**

Keywords: privacy-preserving computation, compressed sensing, mixed-signal computation, internet-of-things

I. Introduction

With the emergence of artificial intelligence and internet of things (IoT), there is enormous raw data generation for real-time applications, such as video streaming, VR/AR, surveillance, and so on. However, the extensive data exchange also introduces severe concerns for individual and public data privacy. To simultaneously preserve privacy and enable computation, great efforts have been made to develop diverse privacy-preserving algorithms and systems, such as homomorphic encryption [1], [2], data perturbation [3], [4], and partitioned deep-learning [5], [6].

However, incorporating privacy protection into sensors is challenging due to constrained in-sensor resources (e.g., power, area) and stringent performance requirements (e.g., latency, throughput). Firstly, most privacy-preserving algorithms are costly: they either introduce tremendous computation/communication overhead (e.g., homomorphic encryption [7]) and/or require significant memory/processing resources (e.g., federated learning [8]). Secondly, light-weight algorithms, such as differentially-private compressed sensing (DP-CS) [3], [4], are prone to various attacks, such as the maximum a posteriori probability (MAP) estimate and/or least mean square matrix inversion (LMI) [9], [10]. This additional threat requires many security resources, such as a projection matrix repository. Most importantly, state-of-the-art digital hardware architecture (shown in Fig. 1) for privacy protection has separate sensing, data conversion, and data security (e.g., entropy generation) modules which is highly inefficient for in-sensor implementation. The overhead is even worse when the security specification, such as the number of challenge-response-pairs (CRP) for the physically-unclonable-function (PUF), is high.

To address the challenges and enable "privacy-by-sensing", this paper proposes a generic time-domain mixed-signal data perturbation hardware-software co-design framework. We summarize the major contributions of the paper as the followings:

1) A light-weight PUF-secured differentially private compressed sensing (DP-CS) framework for privacy preserving edge-cloud collaborative machine learning.
2) A mixed-signal circuit featuring a unified PUF and time-domain multiplication-and-accumulation (MAC) module for ultra-low-power, ultra-small-footprint private-by-sensing implementation (as shown in Fig. 1).
3) Validated efficiency gain, robustness of PUF, attack resilience, and algorithmic performance via TSMC 65nm pre-silicon implementation and SPICE simulation.

The remainder of the paper is organized as the following. In Section II, we will briefly discuss state-of-the-art privacy-preserving computation algorithms, especially the algorithm fundamentals of DP-CS. In Section III, we will discuss the proposed PUF-secured DP-CS framework featuring a secure random projection matrix update. Further, in Section IV, the proposed MS DP-CS architecture will be illustrated for the proposed framework, and a detailed TD-MS circuit implementation will be discussed. Finally, we will compare hardware effectiveness (e.g., area, energy) and accelerator algorithmic performance (e.g., accuracy) with the digital baseline, discuss the performance of our PUF design, carry out an attack analysis and present the algorithmic performance in Section V.

II. Background and Related Works

In this section, we discuss the background and related works for in-sensor privacy protection, as well as algorithm fundamentals for lightweight data perturbation methods, such as differential privacy (DP) and compressed sensing (CS).

As mentioned in Section I, state-of-the-art privacy-preserving methods, including homomorphic encryption and local feature extraction, suffer from large computation and communication overhead, making it impractical for IoT devices.

Meanwhile, data perturbation methods feature value alternation that prevents input recovery and maintains computational

978-3-9819263-7-8/DATE23/© 2023 EDAA

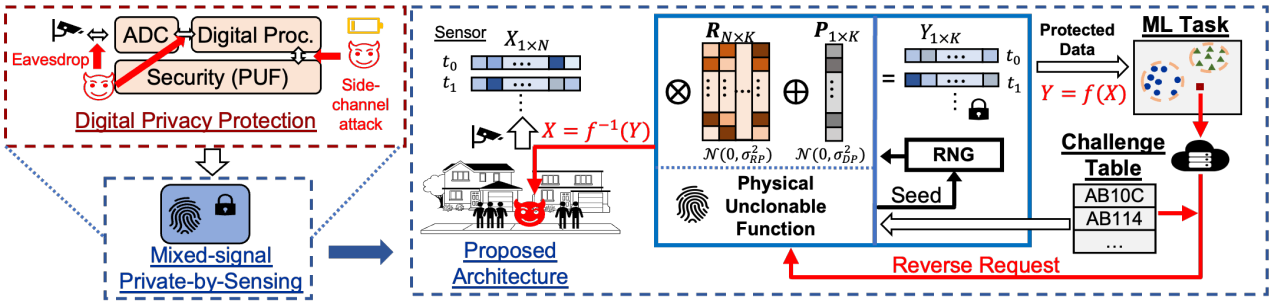


Fig. 1: Private-by-sensing is to embed privacy-compliance into resource-constrained sensors. The mixed-signal technique is a promising method for integrated computation and embedded stochasticity. On the right hand side, a sensor is sensing constantly at time $t_0, t_1, \cdots$. For each instance, the original data is projected and protected.

capability of the original data. Differential Privacy (DP) [11] is is a common data perturbation algorithm that guarantees that with any input data sets, X_1 and X_2 differing in at most one attribute, that for all possible outputs $O \subseteq Range(A)$ of any algorithm A, $P[A(X_1) \in O] \leq e^{\epsilon} \cdot P[A(X_2) \in O]$ holds, where $P[\cdot]$ is the probability function, ϵ is the privacy budget of the DP algorithm. With a user defined ε, a calibrated noise vector $P_{1\times K}$ then can be added to the original data, as shown in [12], formulated by $Y_{1\times K} = X_{1\times K} + P_{1\times K}$, in which each element of $P_{1\times K} \sim \mathcal{N}(0, \sigma_{DP}^2)$ as shown in Fig. 1.

To reduce the noise at the required privacy budget, compressed sensing is usually applied before noise injection. Random projection (RP) is one of the compress algorithms that projects high dimensional data into low dimension space using a randomly generated matrix. The Johnson-Lindenstrauss lemma [13] ensures that the projection preserves relative distance information between data points. The elements of the random projection matrix ($R_{N\times K}$) are drawn from an i.i.d. Gaussian distribution $\mathcal{N}(0, \sigma_{RP}^2)$ with $\sigma_{RP}^2 = 1/K$. Given $0 < \lambda < 1$, a set X of m points in $\mathbb{R}^N$, and a $K > 8\ln(m)/\lambda^2$, there is a linear map $f : \mathbb{R}^N \to \mathbb{R}^K$ s.t. $(1-\lambda)\|u-v\|^2 \leq \|f(u)-f(v)\|^2 \leq (1+\lambda)\|u-v\|^2$ for all $u, v \in X$. As shown in Fig. 1 given an N-dimensional data $X_{1\times N}$, the projection $X_{1\times K} = X_{1\times N}R_{N\times K}$ is directly matrix multiplication. The simplicity of MAC operation makes RP a hardware-friendly algorithm for private-by-sensing.

However several attacks have been proposed against random projection such as MAP, LMI, and linear regression and compressive sensing (LRCS) [9], [10], [14]. These known input-output attacks only require the attacker to acquire a few records to breach secured data, making it the most plausible way of attack. This risk requires extra security measurements, for example, periodic projection matrix updates. We will present further experimental results in Section V.

III. PUF-secured DP-CS framework

As discussed in Section II, DP-CS is prone to known input/output attacks suck as MAP, LMI, and LRCS attacks when the projection matrix is compromised with a high volume of input-output pairs. To mitigate this concern, we propose a PUF-secured projection matrix generation/storage framework for DP-CS algorithms, shown in Fig. 1. Beyond the conventional DP-CS, the proposed algorithm features edge-cloud collaborative challenge-response-based random matrix generation. On the cloud side, a table with all feasible challenges for an edge device will be stored. Before each task, the cloud will issue a series of challenges to the sensor, and the sensor will generate a unique projection matrix with a seed produced by the embedded PUF. Since the PUF is securely stored by the hardware fabrication variations, the projection update process is secure against external privacy attackers and robust to intermittent power supply for energy-constrained sensors. The assumption in place is that only the cloud knows the challenge and the cloud is a trusted third party and thus cannot be the attacker.

Since PUFs encode entropy intrinsically, our system is also reversible. In case of the system needing the original data, for example a criminal is identified and the raw data is required, as depicted in Fig. 1, the cloud can resent the challenges and the exact same $R_{N\times K}$ and $P_{1\times K}$ are regenerated. With the protected data $Y_{1\times K}$, the original data $X_{1\times N}$ can be reverted with minimum error. Since the PUFs uniquely encode the static entropy, no one else other than the device who generated the data can recover it.

Despite the security and reversibility features, the state-of-the-art PUFs, such as SRAM PUFs, introduce significant area and power overhead. To address this issue, we propose a mixed-signal circuit featuring a unified PUF and time-domain MAC module.

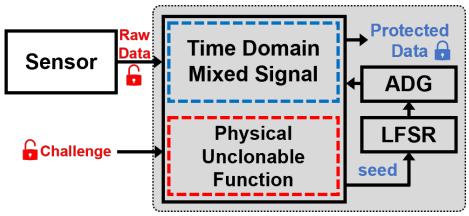


Fig. 2: Proposed MS PUF-secured DP-CS architecture.

IV. MS DP-CS Architecture

This section introduces a time-domain mixed-signal (TD-MS) circuit technique for the proposed PUF-secured DP-CS algorithm with low-power and small-area implementation.

A. MS DP-CS Architecture

The MS architecture for proposed DP-CS is shown in Fig. 2. There are two major functionalities: TD-MS parallel computation and PUF-enabled arbitrary-distribution-generation (ADG). The TD-MS computation module directly interfaces with sensors and projects original data with TD-MS multiplication-and-accumulation (MAC) arrays. The projection matrix and random vector are generated with ADG in run-time whose distributions (e.g., standard deviation) are programmable.

At the time of the projection matrix update and cloud challenge, a hardware-fingerprint sequence is generated by the PUF (in the red dash box in Fig. 2) embedded in the TD-MS MAC array (e.g., ring oscillator frequency variations). The generated random number will act as the seed to enable a linear-feedback shift register (LFSR)-based ADG whose distribution is determined by the probability segmentation sets for the uniformly-distributed LFSR output.

B. TD-MS Circuit Implementation

TD-MS computation has demonstrated its advantages of superior computation efficiency, reduced interconnection, voltage scalability and compatibility to pulse-modulated sensors (e.g., radar [15]) in various low-power applications (e.g., swarm robotics [16], acoustic signal processing [17]).

1) TD-MS MAC array: A typical time-domain MAC array encodes one operand of multiplication as the time pulse and the other as the clock frequency, both proportional to the operands' magnitude. To give it the ability to handle signed numbers, we use a sequential up-down counter triggered by the pulse-enabled clock, and a result proportional to the vector dot product is produced at the counter output. Specifically, the circuit we implemented consists of a B_{DCO}-bit digitally-controlled oscillator (DCO) and an asynchronous $2B_{DCO}$-bit up-down counter. The DCO is used to generate a clock whose frequency is proportional to the magnitude of random number r (represented by a thermometer code) from ADG in the projection matrix. The B_{DCO}-bit DCO is built with a 3-stage ring oscillator and an AND gate for input pulse enable. In each stage, there are 3 inverters in series. Its frequency is governed by a controllable current mirror which consists of $M = 2^{B_{DCO}} - 1$ foot transistors, as shown in Fig. 3 (B). The thermometer code controls the foot transistors to regulate frequency: oscillator frequency increases linearly with the number of turned on foot transistors. There are also biasing PMOSs whose gate voltage act as a biasing voltage to adjust the frequency range. When a pulse sensor input enables the DCO, the generated clock is going to trigger the up-down counter. The counter is a typical asynchronous $2B_{DCO}$-bit counter with an up-down counting selection. For K fully-parallel vector multiplication threads, a TD-MS MAC array is formed with shared input data. Because data is carried on a single wire, the interconnection area is minimal. Further, at the output of each thread, the data is already in digital representation, which features seamless post-digital processing.

2) PUF: In the entropy generation phase, the proposed TD-MS MAC array has DCOs which can be effectively reused for the PUF with large challenge-response-pairs (CRP), as shown in Fig. 3 (C). In the proposed TD-MS PUF challenge, C is divided into C_{MSB} for DCO selection and C_{LSB} for foot transistor selection. C_{MSB} will choose two DCOs for count racing, and C_{LSB} will choose one specific foot transistor in these DCOs, and the process variation between them will cause a unique discharging current variation of ring oscillators and further influence its frequency. The output of DCOs will trigger a counter, and the one with a higher intrinsic frequency will overflow in a shorter time, thus providing a binary output at the output of a digital arbiter. With an increasing DCO MAC array dimension, the number of CRPs increases quadratically. In a K-parallel TD-MS MAC array with bit precision of B_{DCO}, there are K DCOs and $\mathcal{O} \sim (K^2)$ DCO pairs. Ideally, there are around $\mathcal{O} \sim (2^{B_{DCO}} K^2)$ CRPs. Since the PUF mapping is secure, the available challenge can be stored externally.

C. Entropy Generation

The distributions of projection matrix and random noise are both strong functions of data sensitivity and privacy budget. As such, an ultra-low-power run-time reconfigurable LFSR-based ADG is proposed for versatile privacy protection. After a response is generated by the embedded PUF, it will serve as the seed for the LFSR whose output is a uniformly distributed pseudo-random number U. Meanwhile, the ADG is comprised of an array of comparators, each assigned with a predetermined segmentation seg_i. It outputs a thermometer code for desired distribution by comparing U and seg_i. The adjacent segmentation can be determined by computing the integral of a specific distribution function, and the distance between them is proportional to the probability, with a Gaussian distribution example shown in Fig. 3 (D). The number of comparators scales with a granularity of stochastic sampling. Although we only use Gaussian distributed numbers in this paper, the proposed LFSR-based ADG is general to provide an arbitrary distribution.

V. Hardware/Algorithm Evaluation

In this section, we will discuss the hardware and algorithm experimental results of the proposed TD-MS DP-CS accelerator and digital baseline. All the circuits are implemented and simulated with TSMC 65nm low power (LP) technique across a wide power supply dynamic range at room temperature. The digital components are synthesized by Synopsys Design Compiler for the power and area, and the analog components are simulated with Synopsys HSpice for the power and Cadence Virtuoso for the area.

A. Hardware overhead

To evaluate the performance of the proposed TD-MS accelerator, we implement a digital baseline. Given the limitations of IoT devices, the following baseline was chosen to simplify the baseline circuit, thus making a fairer comparison. All the circuit implementations share the same technology node, system clock frequency (40MHz), bit-precision (4-bit), power supply range

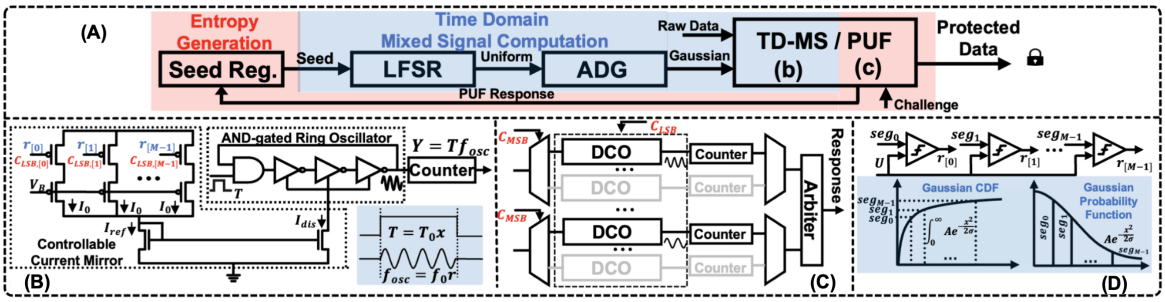


Fig. 3: Proposed TD-MS/PUF unit. (A) Entropy generation phase and time domain mixed-signal computation phase of the proposed MS DP-CS flow. (B) Proposed digitally-controlled oscillator (DCO) based time-domain mixed-signal (TD-MS) MAC. (C) Architecture of proposed DCO-based physical unclonable function (PUF). (D) Proposed LFSR-based ADG circuit.

(0.6V-1.2V), computation/stochastic data flow, and functionality. In the digital version, a 4-bit switched capacitor ADC is included to interface with the sensor, and the acquired data is shared across K parallel digital stochastic DP-CS threads. Within each DP-CS thread, both designs consist of a 4-bit MAC unit, and a 12-bit LFSR. We use a conventional 0.5kB single-port SRAM to map and store uniform random numbers to programmed distributed numbers. The LFSR output will act as the address to access pre-stored values, and, as such, it can emulate arbitrary stochastic distribution by adjusting the stored values. Meanwhile, we use a similar RO-based PUF, which can generate the same number of CRPs in the baseline for a fair comparison. From Fig. 4 (A), we observe that the proposed TD-MS design consumes significantly less (3.2×) energy than digital, and the major cause of significant energy expenditure of baseline is the SRAM (46%). From Fig. 4 (B), we can find that because of the reusing of DCO for both computation and PUF, it achieves a significant area saving (2.9×). And the overall power-performance-area (PPA) gain is 9.4×.

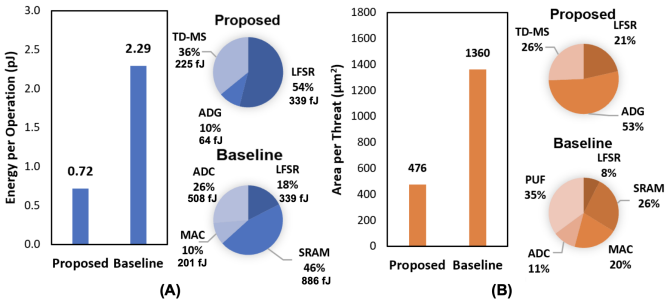
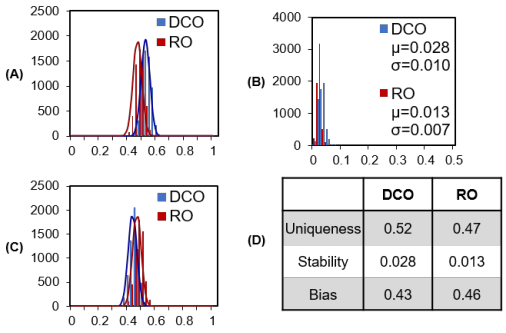


Fig. 4: Measured energy consumption/operation and area for single thread MAC operations ($V_{DD} = 1V$). (A) Measured energy consumption per operation and breakdown. (B) Measured area and breakdown.

B. Performance of PUF

The source of PUF is the fabrication imperfection during the chip manufacturing process, such as critical dimension variations. There are two major factors: first is intrinsic randomness, which gives rise to the independent random distribution of the critical dimension. The other is introduced by the specific manufacturing procedure, which usually imposes a spatial correlation to the random distribution and is called systematic variation. It can be shown that the independent random variation follows a normal distribution, and we denote its standard deviation σ_{rnd}. The systematic variation follows a multivariate normal distribution, which the elements of the covariance matrix has analytic form for location index i and j, $\Sigma_{i,j} = \sigma^2_{sys} \cdot \rho_{i,j}$ where the $\rho_{i,j} = 1 - 1.5 * dist/th + 0.5 * dist^3/th^3$ is a spatial-correlation function of the distance $dist$ between two locations i and j, with a correlation threshold $th = 0.1$ as discussed in [18] and [19]. The overall variation for a specific MOSFET can be expressed by $\sigma^2_{total} = \sigma^2_{sys} + \sigma^2_{rnd}$. This difference in critical dimension ultimately gives different threshold voltages to all the MOSFETs in one die.

To evaluate the performance, we use HSpice to simulate our circuit. First we generate the threshold voltage variation according to the aforementioned distribution, and for each die we set $\sigma_{rnd} = 0.05V_{th}$ and the ratio between systematic variation and random variation be $\sigma_{sys}/\sigma_{rnd} = 0.6$ as discussed in [20]. A response is recorded by giving the circuit a predefined challenge. Furthermore, we simulate the response from a 15-stage ring oscillator PUF. The uniqueness and the stability of the PUF are measured by fractional Hamming distance (fHD) and the 0-1 bias by fractional Hamming weight (fHW), which are the corresponding Hamming distance/weight normalized by the length of the response. The results are shown in Fig. 5.



	DCO	RO
Uniqueness	0.52	0.47
Stability	0.028	0.013
Bias	0.43	0.46

Fig. 5: Y axis is the number of occurrence, DCO is our design and RO is the baseline, (A) Uniqueness. (B) Stability when VDD changes 10%. (C) 0-1 Bias. (D) Summary of the average performance.

From Fig. 5 (A), we can see our design is at the same level in terms of uniqueness as the baseline. Note that all the response results we report here are raw outputs from PUF with

no post-processing. This leads to instability when the supply voltage changes, as depicted in Fig. 5 (B), where there are always some bits that flip from 0 to 1 and vice versa. However, we can mitigate the problem by adding error correction or a majority voting mechanism. Due to the spatial correlation of the variation distribution, we notice the bias of our design is further off center, i.e., 0.5, in Fig. 5 (C). This indicates that the systematic variation will inevitably influence some dies, making those dies unsuitable for use as a PUF.

C. Attack analysis on DP-CS

The following experiments are based on the MNIST and Fashion-MNIST dataset. MNIST and Fashion-MNIST is a handwritten number database and a fashion image database respectively, each containing 70000 images. Each image has 28×28 pixels of 4-bit grayscale information.

Firstly, we examine the attacks on random projection. The first attack method utilized was the aforementioned maximum a posteriori probability (MAP) attack, however, given the $N = 28 \times 28$ dimension of the input images, this method did not converge to a result. The next algorithm for attacking the data uses the method of least mean square matrix inversion (LMI) to linearly regress input/output pairs that the attacker has access to [10]. The assumptions include that the attacker has the protected data vector, $y_{RP} \in \mathbb{R}^K$, following the equation $y_{RP} = xR$, with $R \in \mathbb{R}^{N \times K}$ as the random projection matrix and $x \in \mathbb{R}^N$ as the input vector. The attacker also has p known input/output pairs that use the same projection matrix R. These pairs are corresponding rows of the known input matrix, $X_p \in \mathbb{R}^{p \times N}$, and known output matrix, $Y_p \in \mathbb{R}^{p \times K}$. The LMI attack was capable of producing images containing some protected data with as little as 5 known input/output pairs, however, these images were not a clear reconstruction, and only contained small pieces of the original image. The attack also performed with inconsistent behavior and results that did not improve as the number of pairs in the training data increased, thus making it less desirable for further testing.

Therefore, the third attack method shown below utilizes the same assumptions as before, and linearly regresses the known input/output pairs to recreate the random projection matrix R. From here, the algorithm assumes the recreated matrix of R to be the true matrix and uses single-pixel imaging (SPI) to output the original image from a protected data point that is not within the known input/output training set. This attack has been referred to as linear regression and compressive sensing (LRCS) [14]. In particular, SPI via compressive sampling is required for the CS encryption. SPI is capable of recreating high-quality images from smaller measurements using the inner products between the protected data and the random projection matrix that originally protected the data [21].

The algorithm recreates R using X_p and Y_p through linear regression, and the output is $\hat{R}_p \in \mathbb{R}^{N \times K}$. In particular, the columns of $\hat{R}_p$, $\hat{R}_p(:, i)$ as the i^{th} column, are individually computed with the equation $X_p \hat{R}_p(:, i) = Y_p(:, i)$. By calling the least squares regression function on X_p and $Y_p(:, i)$, the best fit for $\hat{R}_p(:, i)$ is returned. This algorithm is repeated for all K columns to form $\hat{R}_p$. $\hat{R}_p$ is then used along with the protected data being attacked, y_{RP}, where $y_{RP} \notin Y_p$, by the SPI function to return the best recovery of x, $\hat{x} \in \mathbb{R}^N$.

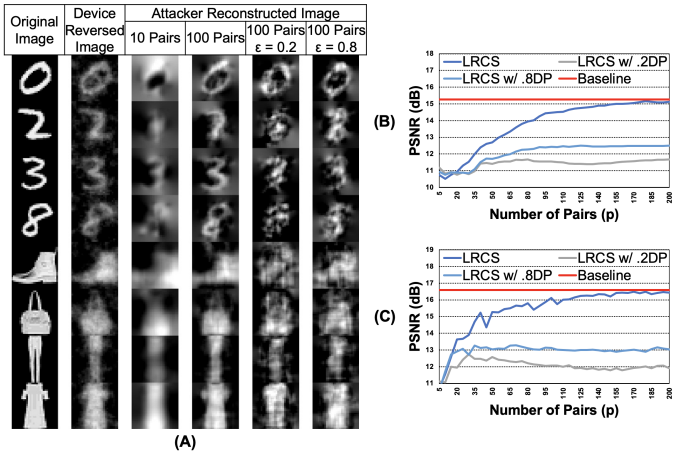


Fig. 6: (A) Recovered images from the LRCS attack for MNIST (top 4 rows) and Fashion-MNIST (bottom 4 rows) datasets. (B) The average PSNR of 200 recovered images across p (MNIST). (C) The average PSNR of 200 recovered images across p (Fashion-MNIST).

This LRCS algorithm consistently performed better as the number of input/output pairs in the training data increased. Notice in Fig. 6 (A) that before DP there is a significant amount of information from the attack with only 10 known pairs (column 3) when compared to the original data (column 1), and the image quality improves moving to 100 known pairs (column 4). It requires more than this significant amount of training data (100+ pairs), however, for the attacked image clarity to reach the quality of the device reconstructed image, where the true RP matrix (R) is known by the attacker (column 2). After DP-CS, with 0.2 epsilon privacy budget (column 5) the image in most cases is unrecognizable, and with a 0.8 epsilon privacy budget (column 6), there is slightly more information presented in the recreated image, but overall has a significantly reduced image quality, especially in the more detailed Fashion-MNIST data. This result exhibits the benefit of DP and how the addition of noise lowers the effectiveness of the LRCS attack. Furthermore, the effectiveness of the LRCS attack for both the MNIST and Fashion-MNIST dataset was quantified in Fig. 6 (B) and Fig. 6 (C) respectively by using the peak signal-to-noise ratio (psnr) between the attacker reconstructed and original image. The red line of both plots represent the average device reconstructed image psnr if the the true RP matrix and noise added is known and the SPI algorithm is used from there. While the value from the MNIST dataset can usually be recognized in many attack recreations given enough training data, the Fashion-MNIST images show that the proposed DP-CS algorithm can protect details of more complex data from an LRCS attack with any DP privacy budget, even after a significant amount of input/output pairs are compromised, as viewed in Fig. 6 (A).

D. Algorithm evaluation

To test the impact of the parameters, we quantize each pixel into a 4-bit representation and compare the results across different privacy budgets and protected data dimensions. To evaluate the performance of the DP-CS, we use k-means clustering to test the effectiveness of the algorithm. For k-means clustering, we do 0-1 clustering. All the images with labels 0 and 1 are used as input data. All experimental results are the average of 50 times.

Minimal performance degradation, within 2%, is observed despite the hardware non-linearity, as presented in Fig. 7. When changing the ϵ from 0.1 to 1.0 and K from 128 to 768, the accuracy increases as the dimension gets larger, and the accuracy decreases as ϵ gets smaller when we are adding more noise to the data. However, a bigger K means more data needs to be transmitted, and by decreasing K the data size proportionally decreases with K over the original input vector length. A larger ϵ also gives less protection over data privacy, and the increased amount of data retrieved after an attack seen between Fig. 6 (A) columns 5 and 6 further shows this decrease in protection. A trade-off can be achieved by choosing a combination of K and ϵ depending on the scenario and the need for edge-cloud collaboration.

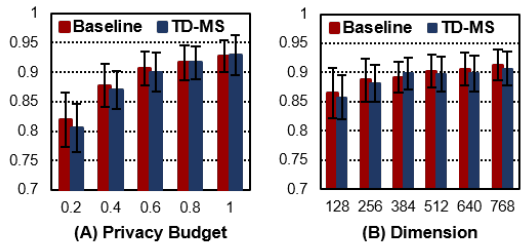


Fig. 7: Measured (A) MNIST 0-1 digit classification accuracy across privacy budget ϵ. (B) Accuracy across protected data dimension K.

VI. Conclusion

This paper presents a hardware/algorithm co-design for "private-by-sensing". On one hand, the PUF-secured differentially-private compressed sensing greatly enhanced the projection matrix update procedure. On the other hand, the proposed in-sensor mixed-signal DP-CS architecture and time-domain circuit implementation demonstrate advanced power, energy, and area efficiency. Compared with the digital baseline (digital-AGD), the proposed TD-MS DP-CS accelerator measures $3.2\times$ energy reduction, $2.9\times$ area savings with $V_{DD} = 1V$, and the number of threads is 128. An attack analysis further shows the data privacy is well preserved. Algorithm accuracy on clustering demonstrates the effectiveness, despite moderate hardware non-linearity, of proposed privacy preserving edge-cloud collaborative machine learning.

References

[1] A. Al Badawi, Y. Polyakov, K. M. M. Aung, B. Veeravalli, and K. Rohloff, "Implementation and Performance Evaluation of RNS Variants of the BFV Homomorphic Encryption Scheme," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 2, pp. 941–956, Apr. 2021.

[2] S. Gupta, R. Cammarota, and T. v. Rosing, "Memfhe: End-to-end computing with fully homomorphic encryption in memory," *ACM Trans. Embed. Comput. Syst.*, nov 2022, just Accepted.

[3] C. Xu, J. Ren, Y. Zhang, Z. Qin, and K. Ren, "Dppro: Differentially private high-dimensional data release via random projection," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 12, pp. 3081–3093, 2017.

[4] B. Khaleghi, M. Imani, and T. Rosing, "Prive-hd: Privacy-preserved hyperdimensional computing," in *2020 57th ACM/IEEE Design Automation Conference (DAC)*, 2020, pp. 1–6.

[5] Z. He, T. Zhang, and R. B. Lee, "Attacking and Protecting Data Privacy in Edge–Cloud Collaborative Inference Systems," *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9706–9716, Jun. 2021.

[6] W. Miao, Z. Zeng, L. Wei, S. Li, C. Jiang, and Z. Zhang, "Adaptive DNN Partition in Edge Computing Environments," in *2020 IEEE 26th International Conference on Parallel and Distributed Systems (ICPADS)*, Dec. 2020, pp. 685–690.

[7] J. Zouari, M. Hamdi, and T.-H. Kim, "A privacy-preserving homomorphic encryption scheme for the Internet of Things," in *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, Jun. 2017, pp. 1939–1944.

[8] S. Truex, N. Baracaldo, A. Anwar, T. Steinke, H. Ludwig, R. Zhang, and Y. Zhou, "A hybrid approach to privacy-preserving federated learning," in *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security*, ser. AISec'19. New York, NY, USA: Association for Computing Machinery, 2019, p. 1–11.

[9] K. Liu, C. Giannella, and H. Kargupta, "A survey of attack techniques on privacy-preserving data perturbation methods," in *Privacy-Preserving Data Mining*. Springer, 2008, pp. 359–381.

[10] W. A. E. Ali, D. A. E. Mohamed, and A. H. G. Hassan, "Performance analysis of least mean square sample matrix inversion algorithm for smart antenna system," in *2013 Loughborough Antennas & Propagation Conference (LAPC)*, 2013, pp. 624–629.

[11] C. Dwork and A. Roth, *The Algorithmic Foundations of Differential Privacy*. Hanover, MA, USA: Now Publishers Inc., aug 2014, vol. 9, no. 3–4.

[12] B. Balle and Y.-X. Wang, "Improving the gaussian mechanism for differential privacy: Analytical calibration and optimal denoising," in *International Conference on Machine Learning*. PMLR, 2018, pp. 394–403.

[13] J. Matoušek, "On variants of the johnson–lindenstrauss lemma," *Random Structures & Algorithms*, vol. 33, no. 2, pp. 142–156, 2008.

[14] S. Jiao, Y. Gao, J. Feng, T. Lei, and X. Yuan, "Does deep learning always outperform simple linear regression in optical imaging?" *Opt. Express*, vol. 28, no. 3, pp. 3717–3731, Feb 2020. [Online]. Available: https://opg.optica.org/oe/abstract.cfm?URI=oe-28-3-3717

[15] A. Amravati, S. B. Nasir, S. Thangadurai, I. Yoon, and A. Raychowdhury, "A 55nm time-domain mixed-signal neuromorphic accelerator with stochastic synapses and embedded reinforcement learning for autonomous micro-robots," in *2018 IEEE International Solid - State Circuits Conference - (ISSCC)*, Feb. 2018, pp. 124–126.

[16] N. Cao, M. Chang, and A. Raychowdhury, "A 65-nm 8-to-3-b 1.0-0.36-V 9.1-1.1-TOPS/W Hybrid-Digital-Mixed-Signal Computing Platform for Accelerating Swarm Robotics," *IEEE Journal of Solid-State Circuits*, pp. 1–11, 2019.

[17] M. Yang, C.-H. Yeh, Y. Zhou, J. P. Cerqueira, A. A. Lazar, and M. Seok, "A 1μW voice activity detector using analog feature extraction and digital deep neural network," in *2018 IEEE International Solid - State Circuits Conference - (ISSCC)*, Feb. 2018, pp. 346–348.

[18] J. Zhang and S. Gupta, "SRAM array yield estimation under spatially-correlated process variation," in *2014 IEEE 23rd Asian Test Symposium*, 2014, pp. 149–155, ISSN: 2377-5386.

[19] J. Xiong, V. Zolotov, and L. He, "Robust extraction of spatial correlation," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 26, no. 4, pp. 619–631, 2007.

[20] W. Zhao, F. Liu, K. Agarwal, D. Acharyya, S. R. Nassif, K. J. Nowka, and Y. Cao, "Rigorous extraction of process variations for 65-nm cmos design," *IEEE Transactions on Semiconductor Manufacturing*, vol. 22, no. 1, pp. 196–203, 2009.

[21] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 83–91, 2008.