2026 Design, Automation & Test in Europe Conference (DATE 2026) – Late Breaking Results

Late Breaking Results: Uncovering the Limits of ECCs in Vision Transformers and a Zero-Cost Reliability Enhancement

Mohammad Hasan Ahmadilivani[1], Marten Roots[1], Marco Restifo[2],
Sven-Markus Loorits[1], Luca Di Mauro[2], and Jaan Raik[1]
[1]Tallinn University of Technology, Tallinn, Estonia
[2]ARM Ltd., Cambridge, United Kingdom
[1]{mohammad.ahmadilivani, jaan.raik}@taltech.ee

***Abstract*—Modern Artificial Intelligence (AI) workloads have increasingly penetrated safety-critical domains (e.g., automotive systems) and reliability-sensitive infrastructures, where transient hardware faults pose significant risks to system dependability. In this work, we conduct the first study of the impact of Error Correction Codes (ECCs) on the reliability of Vision Transformers (ViTs). Building on these insights, we introduce MSET, a lightweight, application-aware protection scheme that selectively hardens the most vulnerable bits in ViT parameters without incurring any memory overhead. Our experiments demonstrate that MSET significantly improves the reliability of ViT inference, surpassing conventional SECDED ECCs. Moreover, the method is fully compatible with existing ECC mechanisms and can be integrated alongside them to achieve even higher resilience. Overall, this study underscores that fine-grained, model-level protection strategies offer substantial reliability benefits for memory-intensive ViTs, enabling reliable deployment in highly error-prone environments.**

I. Introduction

With the superior performance of Vision Transformers (ViTs) across a wide range of computer vision tasks, their adoption in safety-critical domains, such as autonomous driving, healthcare, and space, has accelerated [1]. However, reliably deploying these models in environments where hardware faults may occur remains a significant challenge. Modern ViTs exhibit substantial computational and memory demands, stressing heterogeneous computing platforms and increasing their susceptibility to hardware faults. In safety-critical systems, even rare computational errors can propagate through the inference pipeline and induce erroneous decisions [2].

The massive number of parameters of ViTs further amplifies the reliability challenge. State-of-the-art models contain hundreds of millions or even billions of parameters [3], each of which is stored in off-chip memory that is highly error-prone. In particular, with transistor scaling, parameters become increasingly vulnerable to soft errors [4]. Such low-level faults can silently corrupt attention weights or intermediate activations, thereby compromising the integrity of the entire inference process [5]. Multiple research works have shown that the accuracy of ViTs can be significantly influenced by faults in memory, using radiation [6]–[8] or simulation [5], [9], [10].

Many commercial computing platforms, including NVIDIA GPUs, integrate hardware-level Error Correction Codes (ECCs) based on Single-Error Correction and Double-Error Detection (SECDED) schemes to mitigate soft errors in memory subsystems [11]. While SECDED protects a subset of memory faults by correcting a single erroneous bit per memory line, its fault coverage remains inherently limited. Moreover, these mechanisms introduce non-negligible performance and energy overheads at runtime [12], [13]. Recent experimental studies conducted under neutron irradiation [8] demonstrate that enabling ECC on GPUs does not guarantee full protection against critical Silent Data Corruptions (SDCs). Specifically, the authors observe that (1) ECC reduces, but does not eliminate, SDCs and may even increase Detected Unrecoverable Error (DUE) rates, and (2) single-event upsets occurring in shared memory or the warp scheduler can propagate large numerical faults (e.g., inf, NaN, or high-magnitude values), ultimately leading to critical SDCs in Vision Transformer (ViT) workloads.

To mitigate vulnerability at the model level, [9] proposes a bit-pattern–based protection scheme for parameters with float-32 datatype in ViTs. Their method enforces the pattern 011 on bits 30–28, duplicates bits 27–25, and replaces mismatched duplicated bits with zero during correction. Although effective in certain configurations, this approach may unintentionally introduce additional perturbations in the exponent bits (potentially amplifying numerical errors), and its applicability across different ViT architectures and numerical formats remains limited.

To the best of our knowledge, no prior work systematically investigates the interaction between SECDED ECC mechanisms and the reliability of ViT parameters. In this study, we conduct extensive fault injection campaigns targeting ViT parameters and rigorously evaluate the impact of SECDED protection on model functionality. Our analysis reveals fundamental limitations of ECC when applied to ViTs, highlighting scenarios where faults can bypass correction or produce critical SDCs. Furthermore, we introduce the first zero-overhead protection strategy (MSET) for both float-32 and float-16 parameters, demonstrating superior detection and correction capabilities compared to conventional ECC approaches, without imposing hardware or runtime costs.

II. Methodology: ECCs on ViTs and Bit-Specific Protection

A. Memory Model

In this work, we focus on the main memory subsystem of an AI accelerator responsible for storing the parameters of a ViT. To capture a broad range of hardware architectures, we consider two representative memory interface widths: 64 bits and 128 bits, which correspond to common configurations in edge-AI accelerators. We evaluate two widely adopted numerical formats for ViT parameters: float-32 and float-16, and analyze how their accommodation within memory lines interacts with error-correction mechanisms.

Each memory line is assumed to be protected using a standard SECDED ECC scheme, with the corresponding check bits stored in dedicated ECC memory. For elaboration, under a 128-bit memory line storing float-16 parameters, each line contains eight scalar parameters, meaning that SECDED can correct at most one faulty parameter per eight parameters. Using this memory model, we extract all parameters across a ViT architecture, including attention blocks, normalization and FC layers, and apply our proposed protection techniques directly to these parameter sets.

B. Fault model and Fault simulation

For the memory fault model, we simulate soft errors by injecting random bit flips uniformly across both the parameter bits and ECC check bits. To capture the cumulative impact of memory faults, we conduct fault Injection (FI) experiments across a range of Bit Error Rates (BERs). For each BER, we corrupt the model by introducing random bit flips at randomly selected bit positions throughout the entire parameter space prior to an inference. This procedure is repeated across multiple iterations to obtain statistically meaningful results. Iterations continue until the variation in the mean accuracy converges

978-3-9826741-1-7/DATE26/© 2026 EDAA

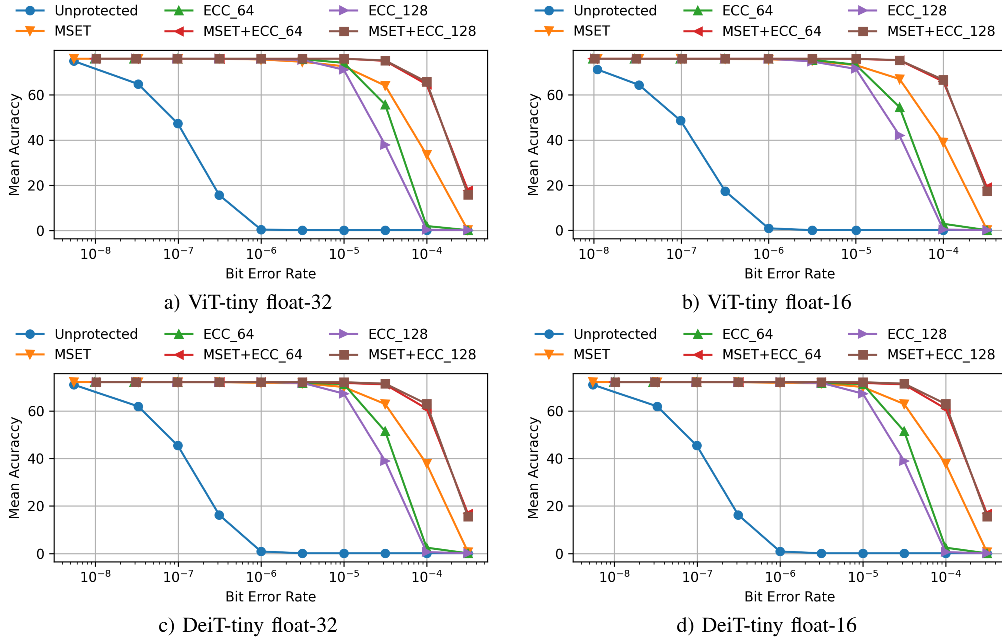

Fig. 1: Average accuracy results for different ViT models with float-32 and float-16 datatypes, using ECC, MSET, and MSET+ECC, under various BERs.

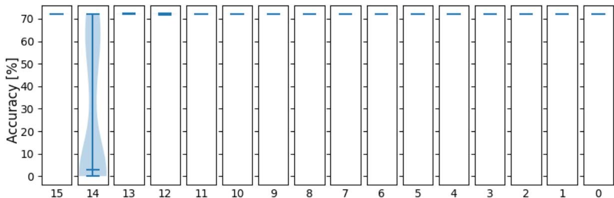

Fig. 2: The impact of individual bitwise bitflips on the DeiT-tiny accuracy with float-16 parameters. It is observed that bit 14 (exponent MSB) is the most critical one, while other bits show high resilience.

to within 1%, to ensure stability and confidence in the results. The reliability metric is considered the average inference accuracy on the validation set over all iterations for each BER.

C. MSET: Zero-Cost Parameters Protection

Based on a bit-level resilience analysis for ViTs (as indicated in Fig. 2), we observe that the Most Significant Bit (MSB) of exponents in both float-32 and float-16 formats exhibits the highest vulnerability, dramatically influencing the ViT accuracy when corrupted. Motivated by this finding, we introduce MSET, which stands for Most Significant Exponent Triplication, a bit-specific protection scheme tailored to ViT parameters. The method leverages the observation that the two least significant mantissa bits contribute negligibly to the numerical value and have no measurable effect on ViT accuracy. We repurpose these two bits by overwriting them with two redundant copies of the exponent MSB.

Before an inference, fault detection and correction are performed by comparing the stored redundant bits with the exponent MSB. A majority-voting mechanism is applied across the three bits, enabling recovery from single-bit errors in the exponent MSB without requiring dedicated ECC capabilities. This approach yields a lightweight, zero-cost protection strategy that enhances the reliability of ViT parameters while preserving the model accuracy.

III. Experimental Results

We present the results of FI experiments for two ViT models (ViT-tiny [14] and DeiT-tiny [15], both with a patch size of 16, loaded from Hugging Face) on the ImageNet dataset. The FI experiments, encoding-decoding for ECC and MSET, are implemented in Rust and integrated into a unified open-source framework[1] with PyTorch to load the ViT models. The experiments are performed on an NVIDIA A100 GPU with an AMD EPYC 7742 CPU. The FI experiments are repeated between 400 and $1,000$, depending on the models, BER, and protection scheme, until the average accuracy is stabilized within 1%.

As observed in Fig. 1, in all configurations, the accuracy of the unprotected models drops drastically even at relatively low BERs (e.g., 10^{-8}, i.e., fewer than 6 faults), highlighting the inherent vulnerability of ViTs parameters to memory-induced soft errors. Noteworthy that float-16 shows a slightly higher resilience than the float-32 datatype.

Incorporating SECDED ECC substantially improves resilience, maintaining high accuracy up to BER=10^{-5}. Furthermore, the results indicate that smaller memory bit-widths provide higher effective protection under ECC. This improvement arises because fewer parameters are accommodated into each memory line; consequently, each SECDED encodes a smaller group of parameters, yielding higher correction coverage and improved overall reliability.

However, this improved protection comes at the cost of increased memory overhead. The additional ECC checkbits overhead is 12.5% and 7% for 64 and 128 memory lines, respectively. For both ViT models, the number of bits in float-32 representation is $181,740,800$, meaning that applying SECDED ECC with a 64-bit memory line requires an additional $22,717,600$ checkbits; which is non-trivial.

The results demonstrate that the proposed MSET technique provides substantially stronger protection for ViTs than conventional ECC. Throughout the results, the average accuracy of ViTs is consistently higher than that of ECCs, particularly at higher BERs, enabling ViTs to remain functional even at BER=3×10^{-5}. These observations indicate that selectively protecting only the exponent MSB is sufficient to eliminate the need for ECC and its associated memory overhead in many deployment configurations.

The MSET encoding–decoding mechanism can be seamlessly integrated into the ViTs software implementation with small runtime cost and zero memory overhead. Moreover, MSET is fully compatible with hardware ECC for enhanced reliability in highly fault-prone environments. As shown in our experiments, the joint exploitation of MSET and ECC enables ViTs to sustain reliable operation at BERs as high as 10^{-4} (i.e., up to $20,000$ faults), highlighting the effectiveness of fine-grained, model-aware protection strategies.

[1] https://github.com/rezzubs/faultforge

ACKNOWLEDGED

This paper is supported in part by EU Grant Project 101160182 “TAICHIP”, and the EU Grant 101194287 “NexTArc”.

REFERENCES

[1] S. Liu, Z. Wang, Z. Gao, P. Reviriego, F. Niknia, X. Tang, J. Zhou, and F. Lombardi, “Are emerging machine learning models dependable at the nanoscales?” *IEEE Nanotechnology Magazine*, 2024.

[2] M. H. Ahmadilivani, M. Taheri, J. Raik, M. Daneshtalab, and M. Jenihhin, “A systematic literature review on hardware reliability assessment methods for deep neural networks,” *ACM Computing Surveys*, vol. 56, no. 6, pp. 1–39, 2024.

[3] X. Zhai, A. Kolesnikov, N. Houlsby, and L. Beyer, “Scaling vision transformers,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 12 104–12 113.

[4] E. Ibe, H. Taniguchi, Y. Yahagi, K.-i. Shimbo, and T. Toba, “Impact of scaling on neutron-induced soft error in srams from a 250 nm to a 22 nm design rule,” *IEEE Transactions on Electron Devices*, vol. 57, no. 7, pp. 1527–1538, 2010.

[5] X. Xue, C. Liu, Y. Wang, B. Yang, T. Luo, L. Zhang, H. Li, and X. Li, “Soft error reliability analysis of vision transformers,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 31, no. 12, pp. 2126–2136, 2023.

[6] J. M. Badia, I. Martin-Salinas, G. Leon, A. Amor-Martin, L. Frias-Dominguez, J. A. Belloch, M. Garcia-Valderas, A. Lindoso, C. Cazzaniga, and L. Entrena, “Reliability of vision transformers and cnns on edge ai systems under neutron radiation,” *IEEE Transactions on Nuclear Science*, 2025.

[7] P. R. Bodmann, P. Rech, and M. Saveriano, “Evaluating the reliability of vision transformers for space robotics applications,” in *2024 International Conference on Space Robotics (iSpaRo)*. IEEE, 2024, pp. 278–283.

[8] L. Roquet, F. Fernandes dos Santos, P. Rech, M. Traiola, O. Sentieys, and A. Kritikakou, “Cross-layer reliability evaluation and efficient hardening of large vision transformers models,” in *Proceedings of the 61st ACM/IEEE Design Automation Conference*, 2024, pp. 1–6.

[9] S. Ahsaei and M. Raji, “Lost-vit: A low overhead soft error tolerance framework for vision transformers via model compression and selective bit-level redundancy,” *Journal of Systems Architecture*, p. 103623, 2025.

[10] G. Gavarini, A. Ruospo, and E. Sanchez, “Evaluation and mitigation of faults affecting swin transformers,” in *2023 IEEE 29th International Symposium on On-Line Testing and Robust System Design (IOLTS)*. IEEE, 2023, pp. 1–7.

[11] M. B. Sullivan, N. Saxena, M. O’Connor, D. Lee, P. Racunas, S. Hukerikar, T. Tsai, S. K. S. Hari, and S. W. Keckler, “Characterizing and mitigating soft errors in gpu dram,” in *MICRO-54: 54th Annual IEEE/ACM International Symposium on Microarchitecture*, 2021, pp. 641–653.

[12] H.-M. Chen, S.-Y. Lee, T. Mudge, C.-J. Wu, and C. Chakrabarti, “Configurable-ecc: Architecting a flexible ecc scheme to support different sized accesses in high bandwidth memory systems,” *IEEE Transactions on Computers*, vol. 68, no. 5, pp. 646–659, 2018.

[13] S. T. Ahmed, S. Hemaram, and M. B. Tahoori, “Nn-ecc: Embedding error correction codes in neural network weight memories using multi-task learning,” in *2024 IEEE 42nd VLSI Test Symposium (VTS)*. IEEE, 2024, pp. 1–7.

[14] A. Dosovitskiy, “An image is worth 16x16 words: Transformers for image recognition at scale,” *arXiv preprint arXiv:2010.11929*, 2020.

[15] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, “Training data-efficient image transformers & distillation through attention,” in *International conference on machine learning*. PMLR, 2021, pp. 10 347–10 357.