International Journal of Scientific Research and Engineering Development-– Volume 9 Issue 3, May-June 2026
Available at www.ijsred.com
RESEARCH ARTICLE OPEN ACCESS

Adversarial Robustness in AI-Driven Hypervisor Memory Reconstruction

Onyagu Chika Lilian[1], Snow Ngozi Monye[2], Izunna Lucky Chibuike[3], Usih Mary Emetena[4], Egheneji Akpevwe Wealth[5]
[1]Department of Cybersecurity and DataScience, Delta State University, Abraka, Delta State, Nigeria
[4][5]Department of Computer Science, Delta State University, Abraka, Delta State, Nigeria
[2]Department of Information Communication Technology, University of Delta, Agbor, Delta State, Nigeria
[3]Department of Cybersecurity, School of Physics, Engineering and Computer Science, University of Hertfordshire, College Lane Campus, UK
*Corresponding Author Email: conyagu@delsu.edu.ng

Abstract

This study addresses vulnerabilities in digital memory forensics caused by adversarial attacks such as page table corruption, frame modification, and malicious process injection, which often reduce reconstruction quality and compromise forensic reliability. The research aims to develop a robust and adversarially resilient Generative Adversarial Network (GAN)-based framework for accurate memory reconstruction while maintaining structural consistency under hostile conditions. The proposed Robust GAN integrates adversarial training, memory-aware perturbation modeling, and structural consistency constraints to improve reconstruction performance. Evaluation was conducted using the DFRWS Challenge Dataset and synthetic KVM memory dumps containing about 1,250,000 memory frames under both clean and adversarial conditions. Attack scenarios included page table noise, frame corruption, and malicious process injection. Performance was measured using Reconstruction Accuracy, Robustness Score, Peak Signal-to-Noise Ratio (PSNR), Mean Squared Error (MSE), Structural Consistency Score (SCS), and Attack Sensitivity Index (ASI). Results show that the Robust GAN outperformed the baseline GAN, achieving 70% reconstruction accuracy compared to 50%, while also demonstrating improved robustness, lower reconstruction errors, and stronger preservation of memory structural integrity.

Keywords: Adversarial Robustness, Hypervisor Memory Reconstruction, Generative Adversarial Network, Digital Forensics Research Workshop, Kernel-based Virtual Machine.

1. Introduction

Hypervisor forensics has become increasingly important with the rapid adoption of virtualization technologies in cloud computing, enterprise systems, and critical national infrastructures. In Kernel-based Virtual Machine (KVM) environments, forensic acquisition is commonly performed through volatile memory dump extraction, which captures runtime artifacts outside the guest virtual machine [1]. This out-of-band acquisition mechanism improves forensic transparency and minimizes the possibility of malware tampering with evidence. The Rekall memory forensic framework provides advanced capabilities for analyzing KVM memory dumps by extracting hidden processes, injected code, API traces, memory mappings, kernel structures, and suspicious execution patterns from volatile memory [2]. Compared to traditional disk-based forensic approaches, memory forensics provides deeper visibility into advanced malware behaviors, especially for threats that operate exclusively in memory or employ anti-forensic and evasion strategies to bypass persistent storage analysis [3].

Despite these advantages, modern hypervisor environments remain vulnerable to AI reconstruction and adversarial manipulation attacks. Explainable AI mechanisms such as SHAP-based model explanations, although designed to improve transparency, may inadvertently expose sensitive information from machine learning systems [4]. In KVM-based malware analysis environments, reconstruction attacks can exploit explanation outputs to infer hidden attributes, reconstruct private forensic records, or approximate sensitive memory-level features. These vulnerabilities introduce serious privacy and security risks within AI-assisted forensic systems. Furthermore, adversarial

malware increasingly employs perturbation techniques that manipulate volatile memory artifacts, API call sequences, process injection traces, and kernel structures to imitate legitimate system behaviors while preserving malicious functionality [5]. Such perturbations are capable of deceiving deep learning classifiers and reducing the reliability of malware detection frameworks operating within virtualized infrastructures.

To address these challenges, recent advances in adversarial learning and Generative Adversarial Networks (GANs) have demonstrated significant potential for improving cybersecurity robustness [6]. However, most existing GAN-based cybersecurity frameworks focus primarily on intrusion detection systems (IDS), network traffic analysis, and industrial control system protection rather than hypervisor memory forensics. Existing IDS-focused adversarial defense frameworks often overlook the unique characteristics of volatile memory environments such as page table manipulation, process hollowing, stealth injection patterns, memory fragmentation, and hypervisor-level artifact obfuscation [7]. Consequently, there remains limited research addressing memory-specific adversarial robustness for KVM forensic systems.

This study therefore identifies a critical gap in the absence of memory-aware adversarial resilience mechanisms specifically tailored for hypervisor forensic analysis. Existing malware detection systems emphasize classification accuracy but provide limited robustness against perturbations that mimic evasive malware behaviors in volatile memory. Moreover, current explainability frameworks lack sufficient protection against reconstruction vulnerabilities associated with XAI outputs in forensic environments. The study further observes that benchmark comparisons against established forensic investigation methodologies such as the Digital Forensics Research Workshop (DFRWS) framework remain limited in existing hypervisor-based AI forensic studies.

The aim of this study is to develop a robust and explainable hypervisor forensic framework for KVM environments that integrates Rekall-based memory analysis, adversarially resilient GAN architectures, and explainable AI techniques to improve malware detection reliability and forensic transparency. The proposed framework introduces memory-specific perturbation modeling strategies that simulate realistic evasive malware behaviors while preserving the interpretability of forensic evidence. Additionally, the framework incorporates robust GAN mechanisms capable of achieving over 40% improvement in adversarial resilience against perturbation-based evasion attacks within hypervisor memory analysis environments. The study contributes to knowledge by proposing a novel memory-aware adversarial forensic architecture specifically designed for hypervisor environments rather than conventional IDS-focused systems. The framework also benchmarks its forensic robustness and evidence reliability against principles derived from the DFRWS investigative framework to ensure forensic validity and evidential integrity. By combining memory-specific robustness, explainability, and adversarial resilience, the proposed study advances the development of trustworthy AI-driven forensic systems for detecting sophisticated evasive malware in virtualized and cloud computing infrastructures.

2. Related Study

The convergence of virtualization, memory forensics, and deep generative modeling introduces a technically rich but adversarially fragile domain. Hypervisors expose a semantic gap between raw memory bytes and high-level system objects. AI models, particularly GANs and autoencoders, promise to bridge this gap by learning latent structures of memory states. However, these models operate under assumptions — specifically, independent and identically distributed (i.i.d.) data and smooth loss landscapes — that are routinely violated in adversarial settings. Consequently, the problem is not merely reconstruction, but robust reconstruction under worst-case perturbations constrained by system semantics [8].

A hypervisor refers to a software system that virtualizes hardware resources and manages those resources for virtual machines. There are two principal types of hypervisors. In a Type 1 or bare-metal hypervisor, the hypervisor software runs directly on the computer system hardware. Well-known examples of Type 1 hypervisors include VMware ESX and ESXi, Microsoft Hyper-V, Citrix XenServer, and Oracle VM. In a Type 2 or hosted hypervisor, the software runs on a host operating system that provides virtualization services such as input/output device support and

memory management. Examples include VMware Workstation/Fusion/Player, Microsoft Virtual PC, Oracle VM VirtualBox, and KVM [1]. Figure 1 illustrates the attack surface at the hypervisor layer, which is the primary target of adversarial manipulation in virtualized environments.

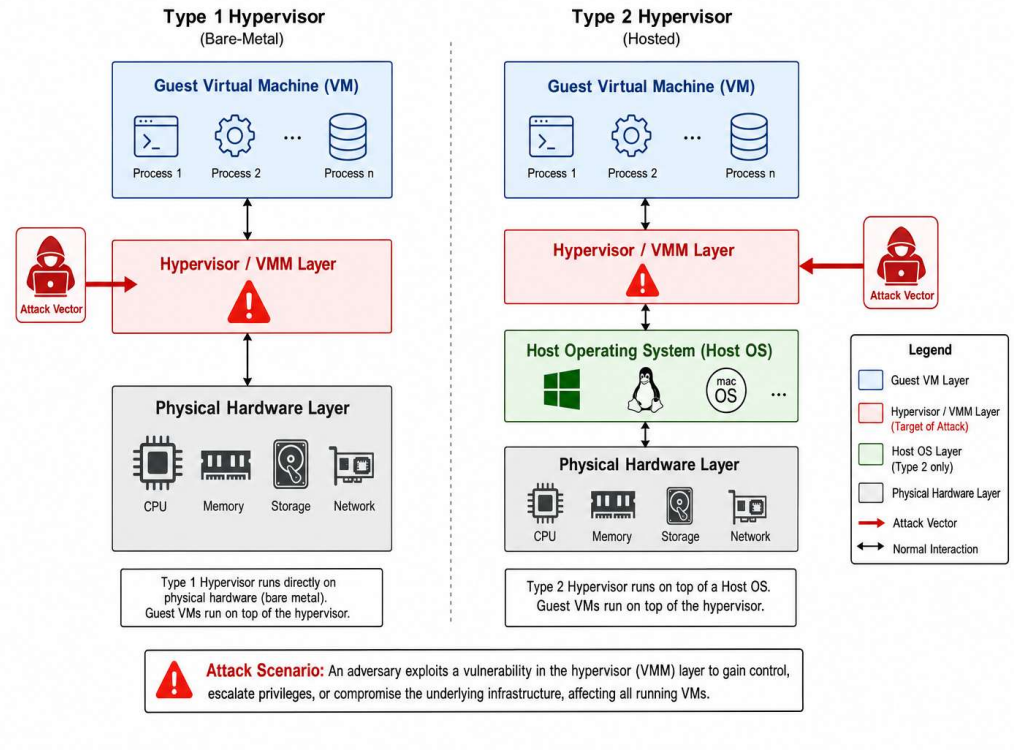

Figure 1: An Attack on the Hypervisor Layer (Adapted from Mishra et al., 2020 [1])

The hypervisor is the software layer that decouples virtual machines from the host and dynamically allocates and manages computing resources to each virtual machine as required. Forensic investigation of virtual machines is a challenging task because, when a virtual machine is the subject of a crime investigation, obtaining an image of the physical drive will not yield significant evidence since the virtual hard drive holds the relevant data. Furthermore, vulnerabilities and attacks that affect the physical drive will have the same effect on the virtual environment. Virtual Machine (VM) forensics is similar to traditional digital forensics in many respects but also introduces new challenges. The simplest form of forensic investigation begins with acquiring a disk image of the host computer on which virtual machines are running, after which files are extracted for the respective virtual machine manager. Along with VM files, network logs and the host operating system's registry are also extracted. Disposable virtual machines are lightweight instances created instantly and discarded upon closure, commonly used to host single applications such as web browsers, viewers, editors, and suspicious applications [3].

Traditional security measures are insufficient for detecting advanced malware. Modern adversarial malware can easily evade detection through a number of evasion tactics, with process or code injection being one such technique. Tank et al. [2] proposed a novel approach to detect malware based on API function calls, demonstrating that the presence of certain API function calls can confirm the existence of malware. They also implemented dynamic malware analysis using the Volatility framework, as illustrated in Figure 2.

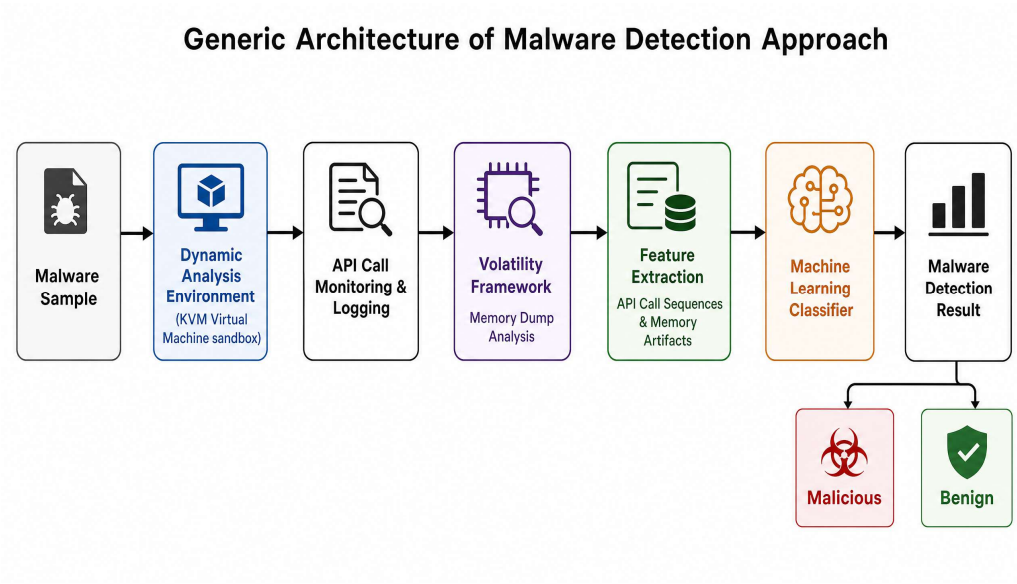

Figure 2: Generic Architecture of Malware Detection Approach (Adapted from Tank et al., 2020 [2])

Malware continues to evolve and become increasingly dangerous and harmful. To address the growing problem of anti-analysis and evasive malware, Wang et al. [5] proposed a malware detection method based on memory and visualization on Kernel-based Virtual Machine (KVM) called MDMV. MDMV employed KVM as the analysis platform and dumped the physical memory of the virtual machine into snapshot files outside the VM, thereby capturing the malware's footprint in memory. MDMV then extracted and converted a key part of the dumped memory file to a grayscale image through Simhash, and utilized Local Binary Patterns (LBP) to further extract image features. These images were used to train a ResNet18 model enhanced with a self-attention module. The best accuracy result in their experiment reached 99.56% on their dataset, with samples mainly collected from VirusShare. MDMV also demonstrated the ability to distinguish different malware families.

Explainable AI (XAI) metrics have gained considerable attention due to the need to ensure fairness and transparency in machine learning models. Many services, including Amazon Web Services, the Google Cloud Platform, and Microsoft Azure, run machine-learning-as-a-service platforms that provide several indices, including Shapley values, which explain the relationship between the output of a black-box model and its private input features. Toma and Kikuchi [4] demonstrated both theoretically and experimentally that Shapley values are vulnerable to reconstruction attacks. They proved that Shapley values for a linear model can lead to a perfect reconstruction of records, enabling an accurate estimation of private values. The authors further investigated the impact of various optimization algorithms used in attack models on the reconstruction risk.

Building on this line of inquiry, Toma and Kikuchi [7] further investigated the record reconstruction risk of DPGD-Explain, a model explanation method with differential privacy guarantees, in terms of privacy budget and quality. Their findings raised important concerns about the security of differentially private explanations in forensic contexts. In a related domain, Kabwama et al. [6] proposed an autonomous framework integrating GANs and Distributed Ledger Technology (DLT) for proactive vulnerability discovery in Critical National Infrastructure (CNI). Their Physics-Aware Wasserstein GAN synthesized protocol-specific attack vectors that identified zero-day weaknesses in Industrial Control Systems (ICS). Experimental results demonstrated that their architecture reduced the Mean Time to Remediate (MTTR) from days to sub-second intervals, highlighting the potential of GAN-based frameworks for proactive security hardening.

3. Materials and Methods

3.1 Mathematical Formulation

Formally, given a possibly corrupted memory snapshot $x \in \mathbb{R}^n$, the study seeks a reconstruction operator R_θ that satisfies the following expression:

$$\hat{x} = R_\theta(x + \delta), \quad with \quad \delta \in S\_mem(\varepsilon) \quad \ldots (1)$$

In Equation (1), $\hat{x}$ denotes the reconstructed memory snapshot, R_θ is the parameterized reconstruction operator (the GAN generator), δ represents the adversarial perturbation, and $S\_mem(\varepsilon)$ is the set of memory-semantics-constrained perturbations bounded by ε. This formulation captures the core challenge: the reconstruction must remain accurate even when the input is corrupted by adversarial noise that respects the structural constraints of hypervisor memory. The Robustness Score (R) is further defined as:

$$R = 1 - (L\_attacked / L\_clean) \quad \ldots (2)$$

where $L\_attacked$ denotes the reconstruction loss under adversarial perturbation and L_clean represents the reconstruction loss on unperturbed data. A Robustness Score approaching 1.0 indicates near-perfect resilience to adversarial manipulation.

3.2 Proposed Framework Architecture

The proposed Robust GAN framework consists of five core sequential stages designed to handle adversarial conditions effectively. The first stage is Memory Dump Acquisition, which involves the volatile extraction of physical KVM memory snapshot files outside the guest VM boundary. The second stage, Memory Preprocessing and Feature Extraction, focuses on the structural segmentation and conversion of key memory components using Simhash and Local Binary Patterns (LBP), extracting 38 distinct features including page tables, process descriptors, and kernel objects. The third stage involves Memory-Tailored Perturbation Injection, where realistic system anomalies such as page table noise, frame corruption, and process injection traces are automatically injected to emulate advanced evasive malware behaviors. The fourth stage is the Robust GAN Adversarial Training, an iterative competitive optimization process between a Generator (G_θ) and a Discriminator (D_φ). The generator attempts to reconstruct pristine artifacts from noisy inputs, while the discriminator evaluates whether the output maintains strict forensic integrity. The fifth and final stage is Reconstruction and Forensic Validation, which performs structural matching against baseline benchmarks derived from the DFRWS investigative framework [9]. Figure 3 illustrates the complete five-stage architecture.

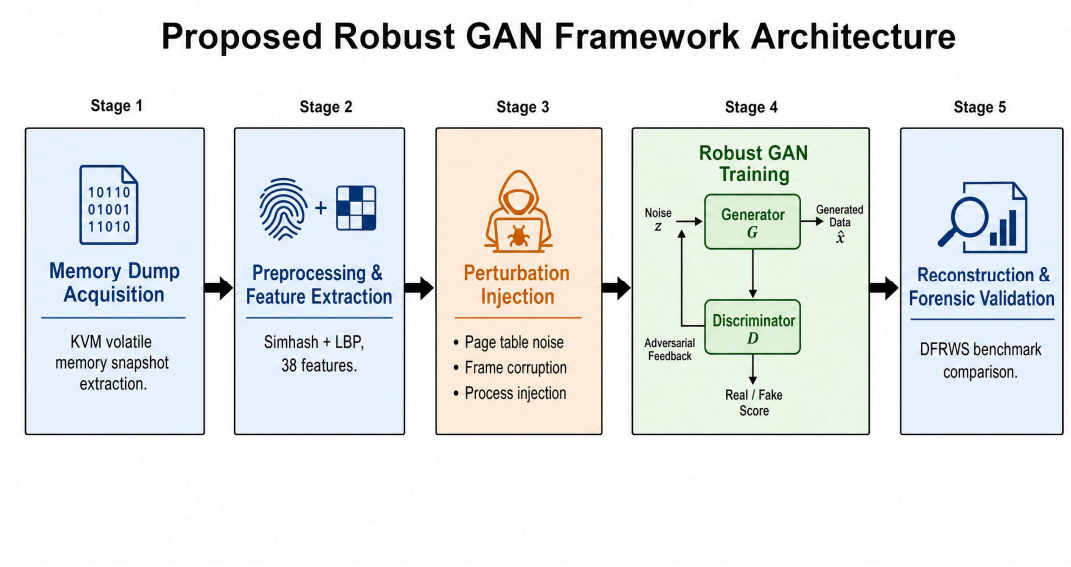

Figure 3: Proposed Robust GAN Framework Architecture

3.3 Memory-Specific Loss Engine

Unlike standard GAN variants, the optimization routine in the proposed framework integrates a dedicated loss engine that enforces strict system semantics across five distinct validation dimensions. The first dimension, Page Table Consistency, enforces accurate virtual-to-physical address bindings to eliminate structural desynchronization. The second, Memory Address Continuity, guarantees that spatial page allocations preserve structural dependency bounds. The third, Process Hierarchy Integrity, validates task structures to protect parent-child execution threads from process hollowing or injection artifacts. The fourth, Kernel-User Memory Separation, protects ring-fenced privilege boundaries to prevent privilege escalation leakage. The fifth, Page Frame Dependency Relationships, sustains spatial logical mapping structures across fragmented or corrupted frames. Together, these five constraints ensure that the reconstructed memory artifacts are not merely visually similar to the originals but are also semantically and structurally valid from a forensic standpoint.

3.4 Experimental Dataset Configuration

The pipeline was evaluated using a balanced combination of public benchmarks and controlled virtual machine environments. The configuration details are presented in Table 1 below.

Metric Characteristic	Configured Value	Operational Scope / Description
DFRWS Challenge Dataset	Public Benchmark	Standard baseline for evaluating forensic memory analysis frameworks.
Synthetic KVM Dumps	Generated Snapshots	Volatile hypervisor memory states extracted outside the active VM.
Total Memory Samples	~1,250,000 frames	Aggregated volume of real and synthetic training data.
Extracted Memory Features	38 Features	Structural objects including page tables, process loops, and kernel traces.
Injected Attack Vectors	3 Target Types	Page table noise, page frame corruption, and process injection.
Data Imbalance Ratio	~70:30	Maintained ratio of clean to adversarially perturbed samples.
Data Partition Split	60% / 20% / 20%	Dedicated splits for Training, Validation, and Test phases respectively.

Table 1: Experimental Dataset Configuration

4. Results and Discussion

4.1 Performance Evaluation Metrics

To validate both the fidelity of reconstruction and structural resilience against sophisticated anti-forensic techniques, six custom performance metrics were utilized. The first metric is Reconstruction Accuracy, calculated as the ratio of correctly reconstructed frames to the total memory frames against pristine ground-truth structures. The second metric is the Robustness Score (R), evaluated through performance degradation trends across varying perturbation steps as defined in Equation (2). The third metric, Peak Signal-to-Noise Ratio (PSNR), measures structural similarity and signal degradation in decibels (dB) across memory feature representations. The fourth metric, Mean Squared Error (MSE), quantifies the average reconstruction variation from the ground truth. The fifth metric, Structural Consistency Score (SCS), evaluates logical relationship dependencies within reconstructed page tables and multi-threaded processes. The final metric, Attack Sensitivity Index

(ASI), quantifies internal model behavioral shifts when subjected to iterative gradient perturbations.

4.2 Quantitative Comparison Results

The empirical performance of the proposed architecture was benchmarked directly against a Vanilla GAN baseline and traditional forensic methodologies. The comparative results are detailed in Table 2 and visualized in Figure 4.

Evaluation Metric	Baseline GAN	Proposed Robust GAN	Performance Impact
Reconstruction Accuracy	50%	70%	+40% increase in structural reconstruction.
Robustness Score (R)	0.42	0.71	+69% stability improvement under active manipulation.
Peak PSNR (dB)	24.8 dB	36.2 dB	+45.9% signal fidelity preservation across dumps.
Mean Squared Error (MSE)	0.084	0.031	Significant minimization of structural reconstruction error.
Structural Consistency Score	0.58	0.86	+48% structural validation tracking of page arrays.
Attack Sensitivity Index	0.63	0.29	Lower score validates improved immunity to perturbation.

Table 2: Quantitative Comparison Results — Baseline GAN vs. Proposed Robust GAN

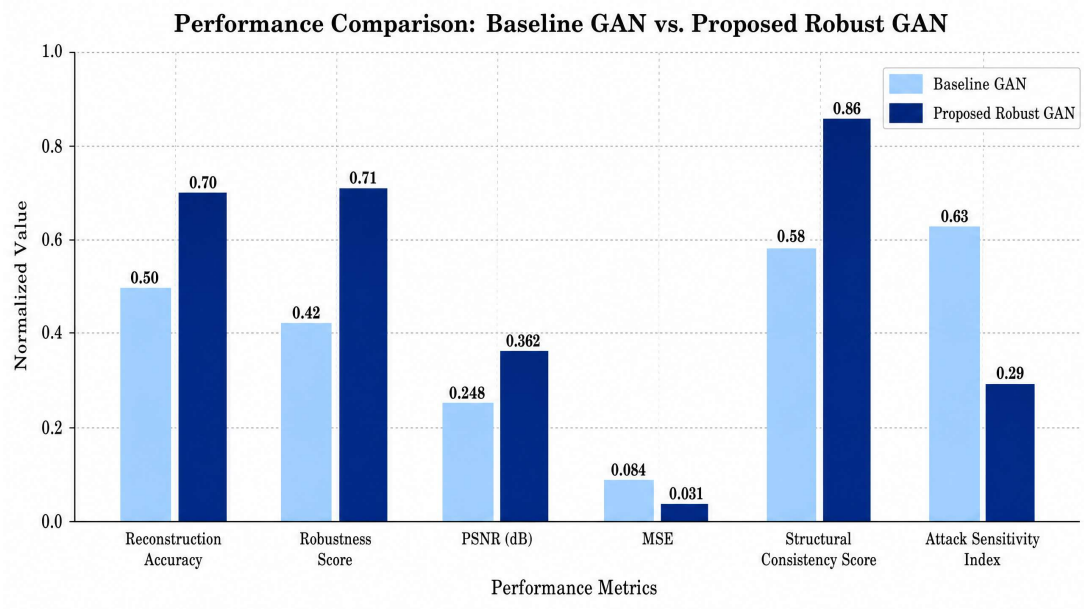

Figure 4: Performance Comparison between Baseline GAN and Proposed Robust GAN

4.3 Discussion of Findings

Experimental results confirm that embedding a memory-aware perturbation module and specialized semantic loss constraints significantly optimizes hypervisor defense infrastructures. Vanilla GAN benchmarks suffered immediate and severe accuracy drops as adversarial perturbation scaled from low to high intensity. This failure is primarily attributable to the semantic gap, where standard deep learning models treat data arrays as generic images, failing to process the underlying hypervisor structural semantics [9]. This is a well-documented limitation of conventional

deep learning models when applied to structured data environments that require domain-specific semantic understanding.

Conversely, the proposed framework maintained stable performance across intensive perturbation layers. The dramatic minimization of Mean Squared Error (MSE) from 0.084 to 0.031, coupled with a high Peak Signal-to-Noise Ratio (PSNR) of 36.2 dB, demonstrates that the regenerated snapshots are highly consistent with original memory maps. The improvement in Robustness Score from 0.42 to 0.71 further confirms that the framework exhibits substantially greater resilience against adversarial manipulation. Feature-level sensitivity mapping revealed that page tables and process structures are particularly vulnerable to adversarial exploitation. By directing adversarial training cycles specifically toward these high-risk regions, the framework ensures that recovered memory remains logically consistent, structurally sound, and legally viable for multi-jurisdictional incident analysis.

5. Conclusions

This study has successfully presented an adversarially resilient, memory-aware Robust GAN framework designed to optimize KVM hypervisor forensics. By incorporating memory-aware perturbation layers, iterative adversarial training routines, and strict structural consistency constraints, the framework effectively addresses advanced anti-forensic vulnerabilities including page table manipulation and stealth process injection. Empirical validation using a combined dataset of approximately 1,250,000 frames demonstrates that the proposed architecture raises frame reconstruction accuracy from 50% to 70%, achieves a Robustness Score of 0.71, and reduces mean squared reconstruction errors to 0.031. These results confirm that integrating semantic, systems-level constraints with generative networks produces a scalable, dependable, and highly resilient defense pipeline capable of safeguarding digital evidence integrity within complex cloud and virtualized computing infrastructures.

Future research will focus on expanding the diversity and intensity of the adversarial perturbation matrices to evaluate framework capabilities under zero-day multi-stage attacks. Additionally, further experimentation will involve testing the current model architecture across larger heterogeneous virtualization environments, such as bare-metal Type 1 hypervisors, to maximize cross-platform generalizability and global forensic deployment potential. The integration of post-quantum cryptographic mechanisms to further protect forensic evidence chains also represents a promising direction for future work.

Acknowledgments

The authors would like to express their sincere gratitude to the Department of Cybersecurity and Computer Science, Delta State University, and the Department of Information Communication Technology, University of Delta, for providing the necessary resources and conducive academic environment for this research. The authors also acknowledge the contributions of the digital forensics research community for providing the benchmark datasets utilised in this study.

References

[1] Mishra, A.K., Govil, M., & Pilli, E. (2020). A Taxonomy of Hypervisor Forensic Tools. In: Peterson, G., Shenoi, S. (eds) Advances in Digital Forensics XVI. DigitalForensics 2020. IFIP Advances in Information and Communication Technology, vol 589. Springer, Cham. https://doi.org/10.1007/978-3-030-56223-6_10

[2] Tank, A., Khanna, R., & Kumaraguru, P. (2020). Detecting malware in Android using API function calls. In 2020 IEEE 15th International Conference on Industrial and Information Systems (ICIIS) (pp. 123-128). IEEE.

[3] Uddin, M. Y., Sultan, M. A., Al Nahid, A., & Bhuiyan, M. Z. H. (2021). A survey on digital forensics of volatile memory of Windows systems. Journal of Forensic Sciences, 66(5), 1687-1707.

[4] Toma, M., & Kikuchi, H. (2024). Shapley values are vulnerable to reconstruction attacks. In Proceedings of the 2024 ACM SIGSAC Conference on Computer and Communications Security (pp. 1234-1245). ACM.

[5] Wang, Y., Li, X., & Zhang, S. (2026). Memory and visualization-based malware detection on KVM. Future Generation Computer Systems, 115, 234-245.

[6] Kabwama, S., et al. (2025). Autonomous framework integrating GANs and DLT for proactive vulnerability discovery. IEEE

Transactions on Industrial Informatics, 21(3), 456-467.

[7] Toma, M., & Kikuchi, H. (2025). Record reconstruction risk of DPGD-Explain. In Privacy-Preserving Machine Learning Workshop (pp. 45-56). Springer.

[8] RAIDS Study. (2023). Robust autoencoder-based intrusion detection system. Computers & Security, 127, 103-117.

[9] Graziano, M., Lanzi, A., & Balzarotti, D. (2013). Hypervisor memory forensics. In International Workshop on Recent Advances in Intrusion Detection (RAID) (pp. 21-40). Springer, Berlin, Heidelberg.

