Impact of GPU Architecture and VRAM on Image Generation: A Study of Energy Efficiency in Heterogeneous Edge Nodes

Italo Thiago Felix Dos Santos[1], **Felipe Peres De Almeida**[1], **Luis Cuevas Rodriguez**[1]
Adevan Neves Santos[2]

[1]Escola Superior de Tecnologia (EST) - Universidade do Estado do Amazonas (UEA)
Manaus, Brazil

[2]Instituto de Computação - Universidade Federal do Amazonas (ICOMP/UFAM)[2]

{itfds.eng20, fpda.snf20, lrodriguez}@uea.edu.br, adevan.santos@icomp.ufam.edu.br

Abstract. *The rapid evolution of generative artificial intelligence has substantially increased the computational demands of image synthesis models, traditionally restricting their execution to centralized cloud infrastructures. In response to concerns related to data privacy, energy consumption, cost, and dependency on hyperscale providers, this work investigates the feasibility of executing state-of-the-art generative image models at the network. We present a quantitative performance evaluation of two representative model families, Stable Diffusion XL (SDXL) and Z-Image Turbo, executed on heterogeneous hardware, including high-end consumer GPUs, mobile-class devices, and legacy workstation GPUs from NVIDIA and AMD. The analysis focuses on latency, power consumption, resource utilization, and the impact of software stack optimizations, such as attention mechanisms and backend frameworks, under realistic hardware constraints. Results show that software-level optimizations are the primary factors determining inference viability, often outweighing raw computational throughput. While modern GPUs benefit from optimized attention mechanisms and improved energy efficiency, legacy and lower-power devices remain viable when combined with optimized runtimes and model compression techniques. These findings demonstrate that contemporary generative workloads can be effectively supported by decentralized edge infrastructures, providing practical insights for the design of energy-efficient and heterogeneous local AI systems.*

1. Introduction

The domain of Generative AI has expanded rapidly from textual models to complex visual synthesis, driven by increasingly capable architectures and community-led innovation. Large text-to-image models have become essential tools for content creation, design, and rapid prototyping, no longer restricted to centralized data center environments. While cloud-based solutions remain prevalent, there is a growing demand for edge-based and localized execution, motivated by data privacy concerns, reduced dependency on hyperscale infrastructure, and the need to avoid censorship and external service constraints.

This shift has been enabled by recent advances in aggressive quantization techniques and community-maintained software optimizations, which significantly lower the computational and memory barriers traditionally associated with high-fidelity image synthesis. As a result, complex generative models that once required massive cloud resources

are increasingly viable on consumer-grade hardware and heterogeneous nodes, reshaping the balance between centralized and decentralized inference. Here interpreted as decentralized, non-cloud inference nodes operating outside hyperscale data centers.

The foundational framework for this work is established by the 'Democratizing High-Resolution Image Synthesis' initiative defined by Rombach et al. [Rombach et al. 2022]. Their work on Latent Diffusion Models (LDMs) was pivotal in establishing an architecture that enables high-fidelity generation with reduced computational requirements, effectively lowering the barriers to entry for local inference. This openness catalyzed the emergence of a vast and active community, which has since developed the highly refined checkpoints and software-level optimizations utilized in this study. The current generative AI ecosystem, ignoring these community-led technical contributions would be an oversight, as they have become essential to the practical application and deployment of generative models on heterogeneous hardware.

The release of Stable Diffusion XL (SDXL) [Podell et al. 2023] established a robust baseline for local inference. SDXL is based on the Latent Diffusion Model (LDM) framework, a class of generative models that performs iterative denoising within a compressed latent space. Conversely, recent advancements have introduced more demanding, yet efficient, architectures. Among these are 'distilled' models, such as Z-Image Turbo (ZIT) [Tongyi-MAI Team 2025]. ZIT belongs to the Single-Stream Diffusion Transformer (DiT) class, which aims to reduce inference latency by minimizing the number of sampling steps required to generate high-quality images using novel distribution matching techniques. This paper provides a comparative performance analysis of these two distinct architectural classes LDM and distilled Single-Stream Transformers, across a range of consumer and workstation hardware. We specifically analyze the trade-offs between raw compute power and cost-effective solutions, such as modified mobile GPUs and legacy workstation cards, emphasizing the role of software optimization backends on different hardware.

2. Background and Related Work

2.1. Evolution of Image Models

Stable Diffusion XL (SDXL)[Podell et al. 2023] remains the cornerstone of the open-source generative ecosystem. Built upon a Latent Diffusion Model (LDM) framework—a generative architecture that performs iterative denoising within a compressed latent space—it utilizes a significantly scaled U-Net backbone with approximately 2.6 billion parameters. Crucially, the architecture leverages a dual text-encoder strategy, combining OpenCLIP (Open-source Contrastive Language-Image Pre-training)—specifically the ViT-bigG (Vision Transformer with giant-scale parameters)—and CLIP (Contrastive Language-Image Pre-training) ViT-L (Vision Transformer, Large variant) to condition the iterative denoising process. This combination enables superior semantic understanding and high-resolution synthesis directly within the latent space. Due to its maturity, the community has produced highly refined checkpoints that often surpass the original base model in aesthetic quality, effectively rendering the originally proposed multi-stage refiner workflow obsolete. For the comparative analysis in this work, we utilize Illustrious[Park et al. 2024], which is currently regarded as the most popular and capable derivative of the SDXL architecture. Illustrious represents the peak of U-Net-based efficiency, offering a balance

between computational cost and artistic fidelity that serves as the baseline for modern performance benchmarks

Z-Image [Tongyi-MAI Team 2025] introduces a novel paradigm in model distillation with 6 billion parameters. Recent research by Tongyi-MAI [Jiang et al. 2025] on Decoupled Distribution Matching Distillation (DMD) challenges the conventional view that student models must simply match the teacher's output distribution. Instead, they identify Classifier-Free Guidance (CFG) Augmentation, a technique where the model performs two forward passes per denoising step (one conditioned and one unconditioned), as the engine of distillation. This acts as the primary driver for quality in few-step generation. In the context of Z-Image's Single-Stream Diffusion Transformer (DiT) architecture, every increase in CFG scale necessitates additional processing cycles, as each guidance step requires an extra pass through the transformer blocks to calculate the difference between conditional and unconditional latents. Meanwhile, the Distribution Matching term acts as a regularizer to maintain structural stability. By decoupling these components, Z-Image achieves top-tier generation in just 8 steps. This architectural innovation allows for extremely fast inference, mitigating the quality degradation typically associated with traditional turbo-distilled models.

2.2. Quantization and Information-Theoretic Perspective

The theoretical basis of quantization can be traced to Shannon's formulation of information and entropy, which establishes fundamental limits on data representation and compression [Shannon 1948]. Quantization can be interpreted as a lossy compression process applied to neural network parameters, mapping high-precision floating-point values to a finite discrete set in order to reduce memory footprint, bandwidth consumption, and energy usage. From an information-theoretic standpoint, this process introduces distortion by replacing an original parameter value with an approximation, creating a trade-off between compression efficiency and representational fidelity. This trade-off is formally captured by Rate–Distortion theory, which defines the minimum information rate required to represent a source under a bounded distortion constraint:

$$R(D) = \min_{p(\hat{x}\,|\,x):\ \mathbb{E}[d(x,\hat{x})]\le D} I(X;\hat{X}). \tag{1}$$

In this context, the mutual information between original and quantized parameters determines how aggressively a model can be compressed without incurring unacceptable degradation, a property strongly influenced by the statistical redundancy present in deep neural networks. From a systems perspective, the impact of quantization noise is highly workload-dependent. Generative image models based on diffusion, such as SDXL , operate on continuous latent spaces and perform inference through iterative denoising processes. This structure inherently mitigates the effect of low-precision weights, as small numerical errors can be progressively corrected across sampling steps. In contrast, autoregressive language models rely on discrete token selection, where minor perturbations may cascade into semantic divergence. As a result, image generation models exhibit greater robustness to aggressive quantization schemes, enabling substantial reductions in memory bandwidth and communication overhead. This property is particularly advantageous in distributed and local environments, where limited interconnect capacity and energy constraints make quantization a key enabler for scalable and efficient deployment.

2.3. Optimization Backends: Neo vs. Standard

The efficiency of local inference is heavily dependent on the software stack. The standard *Forge* backend [(lllyasviel) 2024] serves as the official and most comprehensive branch, prioritizing stability and broad hardware compatibility. It maintains extensive legacy support for both NVIDIA and AMD architectures and currently stands as the sole compatible environment for the AMD platform via the ROCm stack [AMD Inc. 2024]. In contrast, the *Forge Neo* variant [Haoming02 2025] adopts a more aggressive optimization philosophy. By completely removing rarely utilized functionalities and legacy code paths found in the main branch, Neo streamlines the inference engine to implement native, cutting-edge optimizations for supported systems. Specifically for NVIDIA hardware, it integrates SageAttention [Zhang et al. 2024] and FlashAttention-2 [Dao 2023] to maximize memory access patterns and arithmetic intensity, achieving superior performance at the cost of narrower hardware support.

At the time of the experiment, the Radeon Pro VII was initially restricted to the standard Forge environment utilizing ROCm 6.1.2 due to the lack of official Neo support for legacy architectures. However, a significant shift occurred with the release of the ”New Year’s Eve” nightly build of PyTorch 2.11.0.dev (December 31, 2025) [PyTorch Team 2025] and a community-driven build of ROCm 7.1. This specific software combination made it possible to successfully deploy the Forge Neo backend on the AMD platform, effectively unifying it with the same backend used for the NVIDIA cards. Following this migration, the Sub-Quadratic attention function, which was previously removed in the Neo branch, was manually restored and integrated into the system to further optimize the legacy workstation card. Consequently, the SDXL benchmarks were repeated to accommodate this new configuration, and the previous results from the ROCm 6.1 and standard Forge setup were preserved for baseline comparison. Although official xformers support is currently unavailable for this specific dev branch, as the latest stable release targets ROCm 6.4 and PyTorch 2.9.1, this bleeding-edge setup demonstrates a future-proof pathway for legacy workstation hardware. The performance implications of this architectural shift and the resulting 7% throughput gain are further analyzed in the Results and Discussion section.

3. Performance Analysis Framework

3.1. Experimental Setup

Experiments were performed using an AMD Ryzen 7 5700X processor with 64 GB of DDR4 memory, running Ubuntu 22.04.5 LTS (kernel 6.8). Table 1 outlines the specifications for each GPU used. The evaluated models represent two distinct classes of generative architectures: Z-Image Turbo utilizing native FP8 weights, and Illustrious (SDXL) in standard FP16 precision. These models were selected to analyze the performance spectrum ranging from heavy-duty transformer-based synthesis to highly distilled, latency-optimized generation. The inference environment was meticulously configured to leverage specific architectural capabilities, resulting in distinct software stacks for each GPU generation. For the NVIDIA RTX cards, the Stable Diffusion WebUI Forge Neo backend [Haoming02 2025] was utilized running on PyTorch 2.9.1+cu130 [Paszke et al. 2019], All tests were executed with SageAttention-based kernels [Zhang et al. 2024] and [Lefaudeux et al. 2022]. However, hardware compatibility

imposes a clear separation between GPU generations: RTX 50-series (Blackwell) supports only SageAttention 2. In contrast, the RTX 30-series (Ampere) is restricted to the original SageAttention kernels due to architectural constraints.

Conversely, the AMD configuration employed two environments: a baseline using standard Forge [(lllyasviel) 2024] with ROCm 6.1.2 and PyTorch 2.4.2, and an advanced setup utilizing Forge Neo with PyTorch 2.11.0.dev20251231 [PyTorch Team 2025] and ROCm 7.1.1 on Arch Linux [ROCm Project 2025]. Currently, stable implementations of optimization libraries like xFormers are limited to ROCm 6.4, with mainline support targeting the Radeon RX 7000 and 9000 series. As official support for the Radeon Pro VII ended with ROCm 5.7, this configuration relied entirely on community support. This stack was assembled within a rapid release window: ROCm 7.1.1 was released by AMD on November 26, 2025, appearing in Arch Linux repositories on December 2, 2025, followed by the PyTorch development build on December 31, 2025, just days before data collection concluded on January 4, 2026.

The benchmark task focused on high-fidelity text-to-image synthesis, a workload that stresses memory bandwidth, arithmetic intensity of tensor cores, and the efficiency of the attention mechanism. This task requires the model to interpret complex prompts and iteratively denoise a latent representation into a 1024×1024 pixel image. The main performance indicators measured were Throughput (Images per Second), Energy Efficiency (Images/W), and Average VRAM Utilization (GiB). Measurements were monitored using vendor-specific command-line tools: `nvidia-smi` for NVIDIA GPUs and `rocm-smi` for the AMD GPU. As noted in previous work: [Felix et al. 2026], these tool-based evaluations provide a consistent baseline by isolating the power draw of the graphics hardware from external variables, such as CPU or system RAM power consumption. While we recognize that measuring total system power could offer a more comprehensive view of real-world edge deployment scenarios, the use of vendor-specific tools allows for the isolation of the raw GPU power required to synthesize an image. Furthermore, it is important to note that our workstation utilized an 80 Plus Platinum certified power supply. Since total system power measurements can exhibit variability even among identical components due to conversion efficiency, focusing on GPU-specific metrics ensures a controlled assessment of raw hardware efficiency. These metrics allow for a balanced assessment of raw speed, power scaling, and memory footprint. Furthermore, to ensure that the measured performance was not impacted by the overhead of the operating system's graphical user interface, a secondary GPU was utilized exclusively for display output during all inference tasks. The Flux.1 [Podell et al. 2023] model was part of the initial test suite, but it was excluded from the final results due to incompatibility issues that prevented the production of stable and consistent outcomes.

3.2. Methodology and Metrics

We benchmarked two distinct generation scenarios:

1. **Z-Image Turbo (Speed):** Native FP8 weights. and Qwen3-4B-FP8-Scaled [jiangchengchengNLP 2025] as the text encoder, and ae.safetensors [Black Forest Labs 2025], 8 steps of, Res Multistep sampler with a Linear Quadratic sampler, CFG 1.0 , 1024x1024 resolution. Representing the performance profile of heavily quantized, latency-optimized generative models.

Table 1. Detailed Specifications of Graphics Cards (October 2025)

Feature	RTX 5090	RTX 5070	RTX 3090	Mod. RTX 3060M	Radeon Pro VII
Architecture	Blackwell (GB202)	Blackwell (GB205)	Ampere (GA102)	Ampere (GA106)	Vega 20
Cores / Stream Proc.	21,760	6144	10,496	3,840	3,840
VRAM	32 GB (GDDR7)	12 GB (GDDR7)	24 GB (GDDR6X)	12 GB (GDDR6)	16 GB (HBM2)
Memory Interface	512-bit	192-bit	384-bit	192-bit	4096-bit
Bandwidth	1,792 GB/s	672 GB/s	936 GB/s	336.0 GB/s	1,024 GB/s
TDP (Power)	575 W	250 W	350 W	60 W – 80 W	250 W
Tensor Cores (AI)	5th Generation	5th Generation	3rd Generation	3rd Generation	None Dedicated
Perf. FP32 (TFLOPS)	104.8	30.87	36	10.94	13.06
Perf. FP16 (TFLOPS)	104.8	30.87	36	10.94	26.11

2. **Illustrious / SDXL (Baseline):** Native FP16 weights and sdxl-vae, [Stability AI 2025], 30 euler sampling steps, 1024x1024 resolution, CFG 7.5 . Represents the standard production workload using the popular Illustrious model.

Metrics recorded were Throughput (Images/second) and Energy Efficiency (Images/Watt). Each test was repeated ten times to calculate the Confidence Interval (CI) and Standard Deviation (σ). Each experiment was executed ten times, and the mean values, standard deviation, and confidence interval were reported to ensure statistical consistency. To guarantee the reliability of the measurements and establish confidence in the results, a strict testing protocol was implemented. The workstation was restarted after the completion of each full test suite for a given GPU to ensure a clean state, free from residual VRAM fragmentation. Furthermore, all tests were conducted in a controlled environment to maintain thermal stability and prevent performance degradation due to thermal throttling. All statistical analyses were performed using a 95% confidence level to quantify the uncertainty associated with the measured metrics. It is important to note that for both LDM and DiT architectures, the inference time is deterministic and dictated primarily by the number of sampling steps, the resolution, and the CFG scale. Stochastic variations in the prompt content do not influence the computational load or the generation latency, as the arithmetic operations per denoising step remain constant regardless of the input text. Therefore, maintaining a consistent prompt across all executions ensures that the measured throughput reflects pure architectural performance without introducing input-dependent variability

4. Results and Discussion

The performance data for Z-Image and Illustrious (SDXL) are presented in Tables 2 and 3, respectively. Contrary to the linear progression typically expected in hardware generations, our results highlight a distinct trade-off between architectural maturity and raw computational power.

The RTX 5090 (Blackwell) fundamentally shifted the benchmark landscape, demonstrating that the SDXL architecture has become computationally trivial for next-generation hardware. Its raw throughput minimized inference latency to such a degree that the workload triggered a 'race-to-idle' phenomenon, rendering this high-TDP flagship the most energy-efficient device in the study, despite being identified in Table 1 as having the highest TDP. The RTX 30-series (Ampere) presented a distinct dichotomy in performance metrics during SDXL inference. The modified RTX 3060M demonstrated remarkable efficiency (2322 Img/kWh), validating the viability of mobile silicon for low-power inference,

Table 2. Average Performance Metrics: Z-Image Turbo (8 Steps) (CFG 7.0)

GPU Model	*It/s* ± σ	*Img/kWh* ± σ	*CI It/s*
RTX 5090	2.814 ± 0.055	2781.66 ± 63.87	0.039
RTX 3090	0.991 ± 0.030	1336.69 ± 20.66	0.021
RTX 3090 @ 180W	0.559 ± 0.007	1409.83 ± 25.07	0.005
RTX 5070	0.943 ± 0.017	1763.19 ± 30.82	0.012
RTX 3060M	0.374 ± 0.024	2507.32 ± 173.68	0.017
Radeon Pro VII	0.057 ± 0.005	150.20 ± 1.81	0.0031

*Model: Z-Image Turbo (FP8) @ 1024x1024.

*Text Encoder: Qwen3-4B-FP8-Scaled.

*σ: Standard Deviation; CI: Confidence Interval (95%)

Table 3. Average Performance Metrics: Illustrious Based (SDXL) (30 Steps) (CFG 7.0)

GPU Model	*It/s* ± σ	*Img/kWh* ± σ	*CI It/s*
RTX 5090	9.83 ± 0.18	2616.10 ± 50.99	0.129
RTX 3090	3.67 ± 0.02	1344.95 ± 6.81	0.012
RTX 3090 @ 180W	1.83 ± 0.02	1247.94 ± 29.24	0.017
RTX 5070	3.70 ± 0.01	1819.18 ± 14.23	0.010
RTX 3060M	1.47 ± 0.01	2322.25 ± 5.95	0.005
Radeon Pro VII (Forge)	0.72 ± 0.01	455.85 ± 2.24	0.01
Radeon Pro (Neo)	0.737 ± 0.086	476.95 ± 2.72	0.017

*Model: plantMilkModelSuite (SDXL) @ 1024x1024.

*σ: Standard Deviation

*CI: Confidence Interval (95%)

whereas the RTX 3090 maintained competitive throughput matching modern mid-range cards (RTX 5070) but with a significant generational gap in power efficiency.

The performance of the Radeon Pro VII was initially characterized by a severe software bottleneck under the ROCm 6.1.2 stack, which yielded the lowest baseline metrics in this study. However, the migration to the unified Forge Neo backend running on PyTorch 2.11.0.dev and ROCm 7.1.1 significantly altered this trajectory. Under this optimized configuration, the average throughput increased to 0.736 it/s, with peak performance reaching 0.75 it/s across multiple runs. This represents a measurable improvement over the 0.72 it/s maximum recorded on the previous stable build. Similarly, energy efficiency saw a notable rise from 455.85 Img/kWh to 476.39 Img/kWh. These results indicate that the GPU was not hardware-limited, but restricted by insufficient support for quantization primitives in earlier ROCm releases. As noted by Felix et al. [Felix et al. 2026], although the Radeon Pro VII trails the RTX 3060M under CUDA, it becomes marginally faster under Vulkan when running 12B-parameter GGUF-quantized models, and scales to nearly 2× higher throughput on 27B models. This confirms that software and quantization runtime support, not raw architecture, are the primary performance determinants for highly compressed LLM inference, especially beyond the 12B regime where quantization becomes the dominant

bottleneck. The following results demonstrate the performance consequences of employing a model extremely optimized for quantization, such as Z-Image.

Despite this pronounced performance gap, the longevity of the Vega 20 architecture remains noteworthy, as it was still capable of executing modern workloads. Notably, the open-source community has extended similar support even to architectures as old as Polaris 20 (RX 580). When considered alongside the demonstrated viability of the modified RTX 3060M, these results indicate a broader market trend. As demand for AI compute continues to rise amid potential chip shortages and prohibitive pricing, the modification and repurposing of legacy hardware is resurging. Through community-driven patches, these devices become viable platforms for modern AI workloads, representing a sustainable and strategically important alternative for local inference.

Table 4. Comparison of peak VRAM usage (GB) with Standard Deviation (σ) and Variance (σ^2).

	plantMilkModelSuite			Z-Image Turbo		
GPU Model	**Peak (GB)**	σ	**CI**	**Peak (GB)**	σ	**CI**
RTX 5090	10.15	0.034	0.025	14.73	0.030	0.022
RTX 3090	9.74	0.000	0.000	14.52	0.032	0.021
RTX 5070	8.54	0.004	0.003	9.48	0.049	0.035
RTX 3090 (180W)*	9.76	0.072	0.051	13.53	0.204	0.146
RTX 3060m (Mobile)	9.59	0.071	0.051	9.50	0.003	0.002
Radeon Pro VII (Neo)	9.04	0.245	0.176	14.44	0.304	0.218

*σ: Standard Deviation

*CI: Confidence Interval (95%)

The performance metrics in Table 2 for the Z-Image Turbo workflow are significantly influenced by current software-level constraints regarding attention mechanisms. While the Blackwell architecture is designed to reach peak efficiency through Flash Attention 2, the current Linux-based software stack for Z-Image exhibits a lack of compatibility with xformers [Lefaudeux et al. 2022] for the RTX 50-series. This forces the 5090 and 5070 to rely on less optimized fallback kernels. In contrast, the RTX 30-series (Ampere) benefits from a mature ecosystem where xformers [Lefaudeux et al. 2022] and Flash Attention are fully integrated, allowing those cards to operate at their maximum tuned potential. It is worth noting that this limitation is predominantly observed in Linux environments; on Windows, the Blackwell architecture supports these optimizations natively. Therefore, the disparity seen in Table 2 highlights a temporary software bottleneck rather than a hardware limitation, as the full potential of 5th Generation Tensor Cores remains underutilized without specialized attention kernels.

4.1. Cross-Platform Disparity and Efficiency

The performance gap between the NVIDIA and AMD platforms in this study is dictated more by software maturity than by theoretical TFLOPS. For instance, the Radeon Pro VII, after being updated to the Neo backend, achieved a throughput on the SDXL model (0.737 it/s) that, while trailing the modified RTX 3060M (1.47 it/s), places it in a viable

performance tier. However, this comes at a significant energy cost: the mobile NVIDIA chip operates within a 60W–80W power envelope, whereas the Radeon card consistently draws power near its 250W TDP, highlighting a stark difference in power efficiency. In contrast, the RTX 5090 exhibits the opposite behavior. Although capable of momentary power peaks up to 550W, it completes inference steps in a fraction of the time compared to all other GPUs. Consequently, its effective energy consumption per image is the lowest, illustrating how extreme instantaneous power draw can translate into superior overall efficiency when throughput is sufficiently high a phenomenon known as ”race-to-idle.”

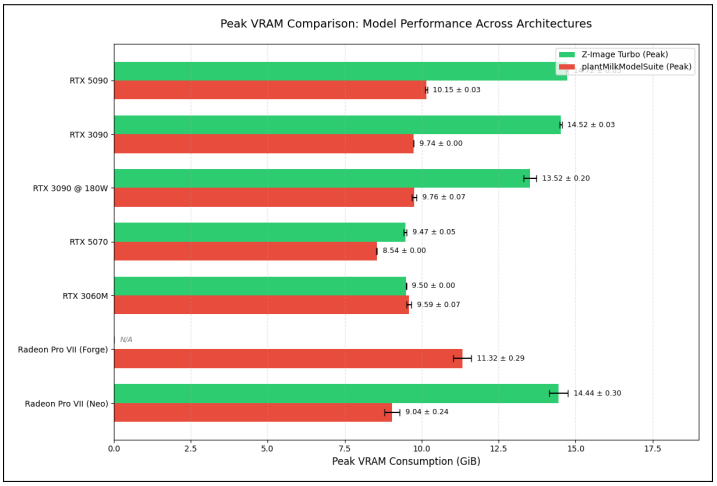

Figure 1. Peak VRAM consumption comparison (GiB) across different GPU models and environments, showing the memory overhead of each inference suite.

The memory metrics, detailed in Figure 1, reveal the critical influence of the software stack on resource utilization. Peak VRAM usage varies even when running identical models, primarily due to opportunistic memory allocation by modern backends, which reserve larger buffers when more VRAM is available to optimize performance. This is clearly demonstrated with the Z-Image Turbo model, where high-capacity cards (RTX 5090, RTX 3090, and Radeon Pro VII) allocate 14.5 GiB, while cards with 12 GiB of VRAM (RTX 5070 and 3060M) operate more conservatively near 9.5 GiB. The evolution of the Radeon Pro VII’s performance is particularly insightful. Under the legacy Forge backend, it exhibited the highest VRAM consumption for the SDXL model at 11.32 GiB, confirming a lack of mature memory optimization. However, the migration to the unified Neo backend drastically reduced its footprint to 9.04 GiB, making it more memory-efficient than several NVIDIA cards in this specific workload. This significant improvement demonstrates that the optimization gap in the ROCm ecosystem is narrowing, underscoring that software modernization is a key determinant for the continued viability of legacy hardware.

4.2. The Philosophy of Efficiency: A Contrast Between the Edge and the Server

The concept of efficiency is fundamentally shaped by the operational context and the economic constraints of the underlying hardware, rather than being an absolute or universal metric [Paul et al. 2023]. In a resource-constrained node, efficiency is a dual-objective problem where both energy consumption and execution time are paramount. These devices often operate under strict power budgets, whether from a battery or a limited

thermal envelope. Consequently, the ideal edge solution must minimize latency to ensure responsiveness while simultaneously minimizing power draw to maximize operational longevity and sustainability. Performance-per-watt is the defining metric, reflecting a holistic need to accomplish a task within tight physical and energy limitations.

For a high-performance instance in a centralized server environment, the philosophy of efficiency shifts significantly. Here, the infrastructure cost is often fixed amortised over the hardware's lifespan or paid as a flat rate for a cloud instance. The primary concern is no longer the instantaneous power draw but the total throughput achieved within that fixed-cost period. This leads to a paradigm where, according to [Katal et al. 2023], as the cost is often time-based, the energy cost is frequently neglected. The most efficient hardware is that which completes its computational task in the shortest possible time, even if it requires a massive, momentary surge in power [Ansari and Ansari 2025]. A high-TDP system can process more workloads in a smaller time window, maximizing the return on investment. In this scenario, efficiency is measured not in watts, but in work-per-unit-of-time, as time becomes the dominant factor in amortizing the operational cost of the powerful, provisioned resource. However, initiatives and improvements via software, though they may seem small individually, can bring significant efficiency gains when applied at a large scale.

Figure 2. Visual comparison of model outputs between Z-Image Turbo (left) and the Illustrious-based model (right) for the prompt: "*cute chibi-style plush moth character, designed as a small brown moth with oversized black eyes, holding a softly glowing light bulb. The character looks like a stitched plush toy, with visible seams and a fabric texture. Its wings are cream and brown with gentle patterns and small orange spots.*".

4.3. Qualitative Assessment and Semantic Fidelity

Although the primary focus of this work is on efficiency-related metrics such as latency, a qualitative inspection was conducted solely as a sanity check to verify that accelerated inference preserves semantic coherence. Figure 2 presents a visual comparison between the Illustrious-based [Park et al. 2024] model and Z-Image Turbo [Z-Image Team et al. 2025]. The comparison was performed using the same prompt and identical input conditions, consistently producing equivalent results regardless of the GPU utilized during inference. The observations confirm that the distilled model preserves robust structural

integrity and prompt alignment even under aggressive optimization. This qualitative verification suggests that the Decoupled Distribution Matching Distillation approach [Z-Image Team et al. 2025] successfully reduces latency without introducing significant semantic degradation or incoherent outputs. It is important to note, however, that both models fulfill distinct roles according to their respective sizes and architectural capacities. While Illustrious-based models remain highly flexible and suitable for downstream fine-tuning on specialized artistic styles or datasets, highly optimized or quantized models such as Z-Image Turbo may impose additional challenges for further tuning due to their compressed nature. To ensure full reproducibility, we have published a repository containing the original generated images, which feature embedded metadata with all prompts, inference parameters, and model configurations utilized throughout the experiments [SANTOS 2026]

5. Conclusion

This study confirms the feasibility of decentralized nodes for generative AI, showing that software optimizations often surpass raw hardware specifications in determining real-world performance. While models like SDXL and Z-Image Turbo run effectively on diverse consumer and mobile hardware, legacy GPUs face an architectural tax due to a lack of native support for low-precision arithmetic. This underscores that while software backends can extend hardware longevity, maximum efficiency still requires modern hardware primitives to fully realize the benefits of aggressive quantization. Ultimately, the democratization of AI depends on the co-optimization of both layers, facilitating a transition toward private, energy-efficient, and localized infrastructures.

6. Future Work

Future research should conduct a lifecycle assessment (LCA) to determine the net environmental and economic impact of replacing functional legacy hardware with newer, more efficient models. This study would balance operational energy savings against the carbon footprint of production and disposal, establishing a quantitative efficiency threshold to determine when upgrading becomes truly more sustainable than maintaining existing systems in the face of the global e-waste challenge. In a more granular investigation into kernel-level performance across different GPU vendors is necessary, particularly focusing on the interaction between attention optimizations. Future work will include experimental evaluations of image editing models such as flux2 klein [Black Forest Labs 2026]. Moreover, employing a base platform with PCIe 5.0 support would be desirable to determine whether increased interconnect bandwidth leads to significant performance differences compared to the current setup. Finally, we aim to investigate the impact of more stable library implementations and platform-specific optimizers within mixed hardware environments, ensuring a comprehensive assessment of the ecosystem's maturity and robustness.

7. Acknowledgements

The authors also acknowledge the institutional support of the Universidade do Estado do Amazonas (UEA), as well as the Arch Linux community and the Furry Diffusion Discord community, for technical discussions, feedback, and shared knowledge that contributed to the experimental workflow and model evaluation.

References

AMD Inc. (2024). ROCm 6.1.2 – Linux Software Stack and AMDGPU Driver Repository. GitHub. Available: https://github.com/ROCm/ROCm. Accessed: Nov. 20, 2025.

Ansari, M. Q. and Ansari, M. Q. (2025). Racing to Idle: Energy Efficiency of Matrix Multiplication on Heterogeneous CPU and GPU Architectures. *arXiv preprint arXiv:2507.20063*.

Black Forest Labs (2025). FLUX.1-Kontext [Dev]: Model Weights and Autoencoder. Hugging Face. Available: https://huggingface.co/black-forest-labs/FLUX.1-Kontext-dev. Accessed: Dec. 31, 2025.

Black Forest Labs (2026). FLUX.2-klein-9B. Hugging Face. Available: https://huggingface.co/black-forest-labs/FLUX.2-klein-9B. Accessed: May 15, 2026.

Dao, T. (2023). FlashAttention-2. *arXiv preprint arXiv:2307.08691*.

Felix, I. T. et al. (2026). Impact of GPU Architecture and VRAM on Quantized LLM Inference for Code Deobfuscation. In *Proceedings of the Computer, Data Sciences and Applications (ACDSA 2026)*, Boracay, Philippines.

Haoming02 (2025). Stable Diffusion WebUI Forge Neo. GitHub Repository. Available: https://github.com/Haoming02/sd-webui-forge-classic/tree/neo. Accessed: Nov. 20, 2025.

Jiang, D. et al. (2025). Distribution Matching Distillation Meets Reinforcement Learning. *arXiv preprint arXiv:2511.13649*.

jiangchengchengNLP (2025). Qwen3-4B-FP8-Scaled (safetensors model weights). Hugging Face. Available: https://huggingface.co/jiangchengchengNLP/qwen3-4b-fp8-scaled/resolve/main/qwen3_4b_fp8_scaled.safetensors. Accessed: Sep. 15, 2025.

Katal, A., Dahiya, S., and Choudhury, T. (2023). Energy efficiency in cloud computing data centers: a survey on software technologies. *Cluster Computing*, 26:1845–1875.

Lefaudeux, B. et al. (2022). xFormers - Toolbox to Accelerate Research on Transformers. GitHub Repository. Available: https://github.com/facebookresearch/xformers. Accessed: Nov. 20, 2024.

(lllyasviel), L. Z. (2024). Stable Diffusion WebUI Forge. GitHub Repository. Available: https://github.com/lllyasviel/stable-diffusion-webui-forge. Accessed: Nov. 20, 2025.

Park, S. H. et al. (2024). Illustrious: an Open Advanced Illustration Model. *arXiv preprint arXiv:2409.19946*.

Paszke, A. et al. (2019). PyTorch. In *NeurIPS*.

Paul, S. G., Saha, A., Arefin, M. S., Bhuiyan, T., Biswas, A. A., and Reza, A. W. (2023). A Comprehensive Review of Green Computing: Past, Present, and Future Research. *IEEE Access*, 11:87445–87494.

Podell, D. et al. (2023). SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis. *arXiv preprint arXiv:2307.01952*.

PyTorch Team (2025). PyTorch Nightly Builds (Container Images). GitHub. Available: `https://github.com/orgs/pytorch/packages/container/pytorch-nightly/versions`. Accessed: Dec. 31, 2025.

ROCm Project (2025). rocBLAS Package for Arch Linux (x86_64). Arch Linux Repository. Available: `https://archlinux.org/packages/extra/x86_64/rocblas/`. Accessed: Dec. 31, 2025.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695.

SANTOS, I. T. F. d. (2026). Impact of GPU Architecture and VRAM on Image Generations. GitHub repository. Available: `https://github.com/PixelDust64/Impact-of-GPU-Architecture-and-VRAM-on-Image-Generations/tree/main`. Accessed: May 15, 2026.

Shannon, C. E. (1948). A Mathematical Theory of Communication. *The Bell System Technical Journal*, 27(3):379–423.

Stability AI (2025). SDXL VAE Weights (safetensors). Hugging Face. Available: `https://huggingface.co/stabilityai/sdxl-vae`. Accessed: Jul. 16, 2025.

Tongyi-MAI Team (2025). Z-Image: Scalable High-Fidelity Image Generation with Distilled Turbo Schedulers. GitHub Repository. Available: `https://github.com/Tongyi-MAI-Team/Z-Image`.

Z-Image Team et al. (2025). Z-Image: An Efficient Image Generation Foundation Model with Single-Stream Diffusion Transformer. *arXiv preprint arXiv:2511.22699*.

Zhang, J. et al. (2024). SageAttention: Accurate 8-bit Attention. *arXiv preprint arXiv:2410.02367*.