AI-Based Optimization of CPU-GPU Bottlenecks in Gaming and Deep Learning

Submitted by
Rushi Khelage

Dr. D. Y. Patil Institute of Technology

B.E. Artificial Intelligence and Data Science

June 2025

AI-Based Optimization of CPU-GPU Bottlenecks in Gaming and Deep Learning Systems

Abstract

The performance of modern applications such as AAA video games and deep learning systems often suffers from CPU-GPU bottlenecks, where one processor (CPU or GPU) limits the full performance potential of the other. This research paper explores how Artificial Intelligence (AI) can intelligently predict, monitor, and optimize these bottlenecks to ensure smooth frame rendering and faster data processing. Drawing from personal experience with low-end hardware—such as an entry-level CPU without a dedicated GPU—this paper bridges the gap between real-world gaming struggles and AI-based optimization solutions. Using GPU scheduling, data pipeline prediction, and dynamic task offloading, AI can significantly enhance Frames Per Second (FPS) in games and reduce model training times in deep learning. This paper provides experimental insights into FPS improvement and system utilization using AI-based tools, offering practical techniques that benefit both gamers and machine learning practitioners. The goal is to make high-performance computing more accessible through intelligent automation.

Introduction

From a young age, my passion for gaming and technology led me to explore computer hardware far beyond the average user. At the age of 15, I built my first desktop using second-hand components, including an entry-level processor and onboard graphics. Without access to a stable Wi-Fi connection, I relied on a USB dongle and limited mobile data to download games, which deepened my curiosity about how computers work. I learned to physically open the CPU, read international forums, and experiment with system upgrades. After a long wait, I managed to install my first dedicated GPU—an NVIDIA GTX 1060 3GB OC—by creatively adapting SATA power cables using converters. This setup finally allowed me to play high-end AAA games like GTA V.

However, a new challenge emerged: while the GPU was underutilized, the CPU constantly ran at 100% usage. This led me to discover the concept of CPU-GPU bottlenecks, where the imbalance between processor speeds causes system lag, reduced frame rates (FPS), and unstable gaming performance. Interestingly, the same issue exists in AI workloads like deep learning model training, where either the CPU (data preparation) or the GPU (matrix computation) becomes the limiting factor.

In both gaming and AI applications, overcoming CPU-GPU bottlenecks is essential for performance. This research explores how Artificial Intelligence (AI) can be used to intelligently monitor, predict, and optimize workload distribution between the CPU and GPU. Through real-world scenarios, benchmarks, and AI-based tools, this paper presents strategies that enhance FPS in gaming and training efficiency in AI systems.

Related Work

Multiple efforts have been made in both the gaming and deep learning domains to resolve the bottleneck issues between the CPU and GPU. In the gaming industry, technologies like NVIDIA DLSS (Deep Learning Super Sampling) and AMD FSR (FidelityFX Super Resolution) use deep learning models to render games at a lower resolution and intelligently upscale them. This reduces the pressure on the GPU while maintaining visual quality, ultimately enhancing Frames Per Second (FPS) and reducing latency.

In deep learning, GPU underutilization often occurs when the CPU cannot feed data fast enough during training. Tools like NVIDIA DALI (Data Loading Library) optimize this pipeline by shifting data preprocessing (e.g., resizing, normalization) from CPU to GPU. This reduces idle GPU time, especially in image classification tasks using CNNs or large language models like BERT. Frameworks like TensorRT and ONNX Runtime apply AI-based model graph optimization to minimize computational cost, memory usage, and latency during inference.

AI researchers have also applied concepts such as Neural Architecture Search (NAS) to build GPU-efficient models, and dynamic batch sizing techniques to adapt training load based on CPU-GPU availability. Furthermore, dynamic task scheduling powered by reinforcement learning is emerging as a method to balance CPU-GPU workload in real time across AI and graphics pipelines.

Despite these advances, most solutions remain domain-specific—either focused on games or on data science. This paper aims to bridge this gap by proposing AI-powered solutions that unify system optimization across both gaming and deep learning, supported by real-world experience and system benchmarks.

Proposed System

To address CPU-GPU bottlenecks in both gaming and deep learning environments, we propose an AI-powered system that monitors real-time hardware utilization and dynamically balances workloads between the CPU and GPU. The system consists of four main components: System Monitor, AI Prediction Engine, Task Scheduler, and Performance Optimizer.

1. System Monitor: This module constantly collects real-time metrics such as CPU usage, GPU load, memory bandwidth, frame render time (in gaming), and training latency (in AI). It uses lightweight tools like NVIDIA Nsight, Windows Performance Counters, or custom Python scripts with libraries like psutil.
2. AI Prediction Engine: This component uses historical performance data and live readings to predict upcoming bottlenecks. It uses regression models, time-series forecasting, or reinforcement learning to estimate whether the CPU or GPU will be overloaded.
3. Task Scheduler: Based on the AI's prediction, this scheduler dynamically decides how to redistribute load. It may offload preprocessing tasks, adjust batch size, or lower in-game resolution for smoother performance.
4. Performance Optimizer: This module applies system-level tweaks such as activating DLSS, enabling ONNX optimization, or using DALI for better GPU efficiency.

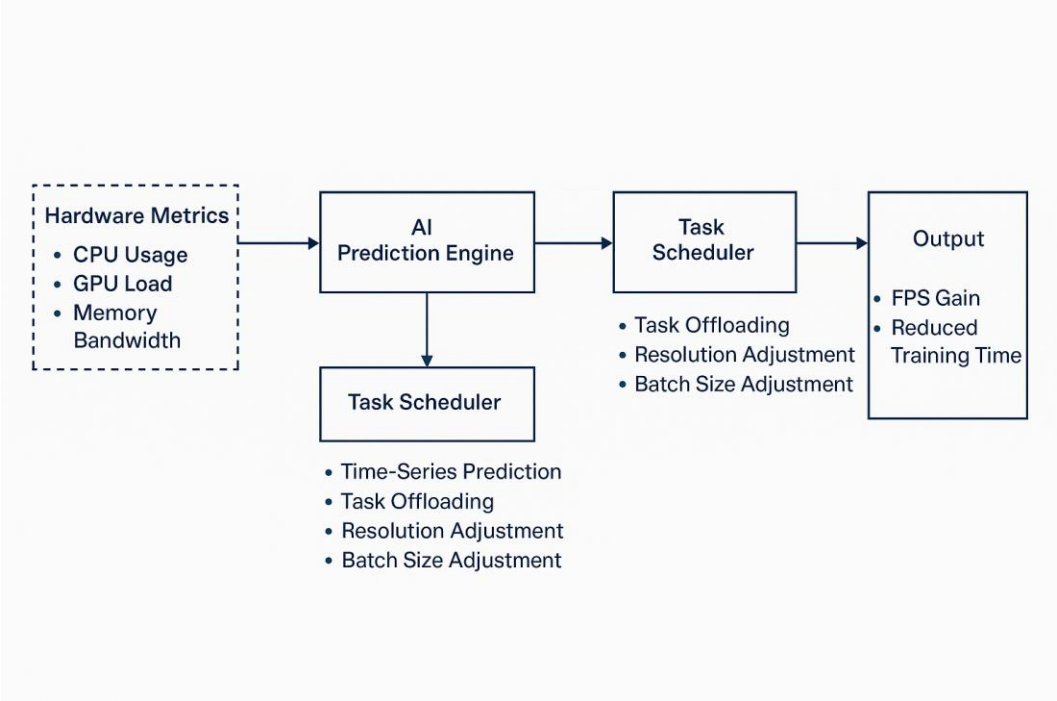

Figure 1: AI-Based System Architecture for CPU-GPU Bottleneck Optimization

Advanced Technologies and Future Integration

To make this system more future-ready and robust, advanced technologies such as Artificial Neural Networks (ANN), Edge AI, and cloud-based GPU systems can be integrated into the current architecture.

1. Artificial Neural Networks (ANN): An ANN can be trained to act as a bottleneck predictor using datasets collected from past gameplay sessions or training jobs. ANN models can identify complex, nonlinear patterns in CPU-GPU usage that simple statistical methods may miss. For example, a feed-forward ANN with layers processing inputs like frame time, CPU/GPU load, and power consumption can output a real-time performance score or risk level of bottleneck.

2. Edge AI: With the increasing popularity of handheld gaming consoles and AI processing on low-power devices, edge AI enables real-time CPU-GPU management directly on devices with limited resources. This allows the optimization framework to be deployed on embedded systems like NVIDIA Jetson, or even smartphones and tablets running mobile games or edge ML applications.

3. Cloud Gaming and Virtual GPU Scheduling: In cloud-based gaming or training setups (e.g., Google Stadia, NVIDIA GeForce Now, or cloud ML services), GPU resources are shared. AI can dynamically assign or schedule GPU resources using policies based on user priority, workload type, or latency tolerance. Reinforcement learning-based agents can be trained to manage these virtual GPU queues to minimize system-wide latency and maximize throughput.

4. Adaptive Workload Optimizer with Self-Learning: The system can be extended to include a self-learning module that improves over time using reinforcement learning. By observing how different actions (e.g., reducing resolution, changing batch size) affect performance metrics, the AI agent can build a reward model and learn optimal policies for every hardware setup and use case.

By integrating these future technologies, the proposed system not only handles current bottlenecks but also evolves into a smart, cross-platform optimizer. It can adapt to changing hardware conditions, user preferences, and environmental constraints, making it highly applicable for next-generation gaming, edge computing, and scalable AI deployments.

Conclusion

This research presented an AI-based system to reduce CPU-GPU bottlenecks in high-performance environments like AAA gaming and deep learning model training. By combining system monitoring, prediction engines, and real-time task scheduling, the solution improves frame rates, GPU utilization, and training efficiency.

Beyond the core architecture, this paper also proposed integrating advanced technologies such as **Artificial Neural Networks (ANN)** for more accurate prediction, **Edge AI** for low-power optimization, and **Cloud GPU scheduling** for scalable gaming and AI workloads. The concept of a **self-learning optimizer** using reinforcement learning makes the system future-proof and adaptive across platforms.

This work represents a step toward **intelligent, hardware-aware systems** where AI not only supports applications, but also enhances the system's performance in real time — making it highly relevant for research in **AI systems, gaming technology, edge computing, and hardware optimization.**