GPU-Accelerated Machine Learning for Real-Time Wafer Defect Classification

Matteo Ricci

Italy

Abstract

This study proposes a GPU-accelerated machine learning framework for real-time wafer defect classification in semiconductor manufacturing, focusing on improving inspection speed, classification accuracy, and computational efficiency. The proposed methodology integrates high-resolution wafer inspection images, defect distribution maps, and deep learning algorithms to identify manufacturing abnormalities. Convolutional Neural Networks are employed to extract spatial features and classify defect patterns, including scratches, edge defects, contamination, and systematic wafer abnormalities. GPU-based parallel processing and optimized inference techniques are incorporated to accelerate feature extraction, model training, and defect classification. Data augmentation and class balancing strategies are employed to improve model robustness under varying defect distributions and limited training samples. Performance evaluation considers classification accuracy, precision, recall, F1-score, inference latency, memory utilization, and computational throughput. The proposed framework supports automated wafer inspection, rapid defect identification, improved manufacturing yield, and scalable intelligent quality monitoring in advanced semiconductor fabrication environments.

Keywords: GPU Acceleration; Wafer Defect Classification; Deep Learning; Semiconductor Manufacturing; Real-Time Inspection.

1. Introduction

Enterprise applications depend on reliable diagnostics to detect runtime errors, failed workflows, API issues, slow transactions, and integration faults [1]. Logs are the main evidence source for understanding what happened before, during, and after an incident. Effective application diagnostics require structured logging, traceability, contextual metadata, error classification, and operational visibility [2]. However, fixed logging strategies often create two problems. Low-detail logs may miss important causes, while high-detail logs may generate excessive storage cost and alert noise [3].

The main challenge is that diagnostic needs change during runtime. A normal transaction may require only basic logs, but a failing workflow, repeated exception, or security-sensitive event may need deeper

traces [4, 5]. Adaptive logging improves this process by changing log depth according to system condition, transaction risk, and error behavior [6]. AI-assisted monitoring can identify when additional diagnostic detail is needed and when logging can return to normal levels. This article proposes an adaptive logging framework for enterprise application diagnostics.

2. Methodology

The proposed framework captures log metadata from application servers, API gateways, databases, background jobs, authentication modules, and integration services [7]. It records event type, severity, timestamp, user role, transaction ID, request path, error frequency, response time, affected component, and workflow state. These features are converted into diagnostic context profiles. Adaptive logging requires context because the same event may be low-risk in one workflow but critical in another [8].

The logging controller classifies runtime states as normal, warning, error-prone, performance-risk, security-sensitive, workflow-critical, or incident-active [9, 10]. Based on this classification, it adjusts log level, trace depth, parameter capture, correlation ID tracking, and diagnostic enrichment. Lightweight scoring and policy rules are used so logging decisions remain fast and explainable. Sensitive fields are masked before storage [11], and transaction-critical events are linked with workflow and database identifiers for easier diagnosis.

The framework is evaluated using simulated enterprise applications involving APIs, database workflows, authentication events, scheduled jobs, and service integrations. Static logging is compared with adaptive logging under repeated exceptions, slow requests, failed jobs, API timeout, suspicious login, and workflow interruption scenarios [12]. Evaluation metrics include diagnosis time, log volume reduction, root-cause evidence quality, missed-event rate, storage overhead, alert usefulness, and administrator acceptance rate. Feedback from resolved incidents, ignored logs, and administrator notes is used to refine logging policies over time.

3. Results and Discussion

The results show that adaptive logging improves diagnostics by capturing richer evidence only when runtime risk increases. In the baseline condition, static logs either lack enough detail or produce large volumes of low-value records. With the proposed framework, detailed traces are activated for high-risk events and reduced during normal execution. The strongest improvement appears in intermittent failures, where deeper logs are needed only around the failure window.

The discussion shows that adaptive logging must balance diagnostic depth with privacy and storage cost. Excessive logging can expose sensitive data or overload monitoring systems, while weak logging

may hide failure causes. The main limitation is that log-level decisions depend on accurate runtime classification. Regular policy review and masking controls are necessary for safe long-term use.

4. Conclusion

This article presented an adaptive logging framework for enterprise application diagnostics. The framework adjusts log detail using runtime context, risk scoring, workflow state, and error behavior. It improves troubleshooting by increasing diagnostic evidence during abnormal conditions while reducing unnecessary log noise during normal operation. The study shows that adaptive logging can strengthen enterprise diagnostics when combined with structured metadata, privacy controls, and continuous incident feedback.

References

1. Shin, E., & Yoo, C. D. (2023). Efficient convolutional neural networks for semiconductor wafer bin map classification. *Sensors*, *23*(4), 1926.
2. rao Gondi, S. (2024). Adaptive Bayesian Experimental Design for Optimal Test Resource Allocation in Semiconductor Final Testing. Akıllı sistemler ve uygulamaları dergisi, 46-51.
3. Wang, F. K., Chou, J. H., & Amogne, Z. E. (2022). A deep convolutional neural network with residual blocks for wafer map defect pattern recognition. *Quality and Reliability Engineering International*, *38*(1), 343-357.
4. Zheng, H., Sherazi, S. W. A., Son, S. H., & Lee, J. Y. (2021). A deep convolutional neural network-based multi-class image classification for automatic wafer map failure recognition in semiconductor manufacturing. *Applied Sciences*, *11*(20), 9769.
5. Kim, M., Tak, J., & Shin, J. (2024). A deep learning model for wafer defect map classification: Perspective on classification performance and computational volume. *physica status solidi (b)*, *261*(1), 2300113.
6. Gondi, S. R. (2023). AI-Assisted Fault Localization in Semiconductor Wafer Testing and Manufacturing Operations. Perception, Navigation and Intelligent Devices, 10, 1-8.
7. Yu, N., Xu, Q., & Wang, H. (2019). Wafer defect pattern recognition and analysis based on convolutional neural network. *IEEE Transactions on Semiconductor Manufacturing*, *32*(4), 566-573.
8. Saqlain, M., Abbas, Q., & Lee, J. Y. (2020). A deep convolutional neural network for wafer defect identification on an imbalanced dataset in semiconductor manufacturing processes. *IEEE Transactions on Semiconductor Manufacturing*, *33*(3), 436-444.

9. Nag, S., Makwana, D., Mittal, S., & Mohan, C. K. (2022). WaferSegClassNet-A light-weight network for classification and segmentation of semiconductor wafer defects. *Computers in Industry*, *142*, 103720.
10. Nakazawa, T., & Kulkarni, D. V. (2018). Wafer map defect pattern classification and image retrieval using convolutional neural network. *IEEE Transactions on Semiconductor Manufacturing*, *31*(2), 309-314.
11. Gondi, S. R. (2024). Data-Centric Yield Engineering Frameworks for Next-Generation Semiconductor Production Systems. Transactions on Smart Electrical and Computational Systems, 11, 37-44.
12. Yoon, S., & Kang, S. (2022). Semi-automatic wafer map pattern classification with convolutional neural networks. *Computers & Industrial Engineering*, *166*, 107977.