Received: 17 June 2026, Accepted: 16 Aug. 2026, Published: 31 Aug. 2026
Digital Object Identifier: https://doi.org/10.63503/j.ijcma.2026.270

Research Article

AI-Enabled Hardware–Software Co-Design for Real-Time Data Analytics in Cloud-Connected Cyber-Physical Systems

Sarvesh Maurya[1*], Raj Shekhar[2], Kavita Mandal[3]

[1,3]Department of CSE (AI/ML) Krishna Institute of Engineering & Technology (KIET), Ghaziabad, Delhi-NCR, Uttar Pradesh, India

[2] National Informatics Centre, Ministry of Electronics and Information Technology, Government of India

sarveshmakers@gmail.com, kumarrajshekar16@gmail.com, kavitamandal0724@gmail.com

*Corresponding author: Sarvesh Maurya, sarveshmakers@gmail.com

ABSTRACT

Due to the rapid development of cloud-based Cyber-Physical Systems (CPS) and smart hardware, a large volume of industrial data has been generated, posing challenges related to latency, scalability, and effective hardware-software integration. The paper provides a framework for an AI-enabled hardware-software co-design of real-time CPS data analytics. It is proposed that the smart sensor hardware, edge-assisted processing, and machine learning models are integrated into the proposed system to enable low-latency decision-making and optimal system performance. The co-design technique guarantees effective communication between hardware potential and software intelligence, reducing computational overhead and communication delays. The proposed framework is evaluated on 10,000 records, showing 91.3% prediction accuracy, 88.7% processing efficiency, 0.87% system stability, and a low computational latency of 198 ms. In addition, the general performance index is 0.89 and it demonstrates the balanced scaling, responsiveness and efficiency. The superiority demonstrated by comparative analysis over traditional, machine-learning-based, and hybrid models indicates that the proposed model is the best approach for real-time industrial analytics in dynamic CPS settings.

Keywords: *Machine Learning, Cyber-Physical Systems (CPS), Hardware Software Co-Design, Real-Time Data Analytics, Smart Sensors, Edge Computing, Cloud Computing*

1 Introduction and Research Background

The fast development of cloud-based Cyber-Physical Systems (CPS) has revolutionized contemporary industrial spaces because they allow a smooth combination of calculating, communicating and controlling physical activities. Real-time data are continuously produced by smart sensor hardware and embedded systems and are essential for predictive maintenance, process optimisation, and intelligent monitoring applications. Data storage and massive analytics can also be increased by the implementation of cloud computing. Nevertheless, the massive growth of data and the rise in system complexity pose serious problems related to latency, scalability, and optimal resource use in the CPS environment [1], [2].

Conventional data analytics models that mainly rely on centralised cloud platforms experience significant communication delays and low responsiveness. Such problems are more acute in the time-sensitive industrial world, where real-time decision-making is crucial. Furthermore, purely software-based solutions do not make full use of hardware capabilities and hence, performance inefficiencies and higher computational costs are experienced. The recent code progress in machine learning (ML) and edge computing has made it possible to conduct localized data processing and conduct smart analytics to make the system more responsive. However, issues such as resource limitations, model complexity, and the absence of an integrated hardware-software interface remain [3], [4].

In an effort to counter these weaknesses, hardware-software co-design has emerged as a viable paradigm for jointly optimising system hardware and intelligent software components. Co-design frameworks represent the most effective way to minimize the latency, increase the scalability and the efficiency of the entire system by eliminating dependence on warehouses and traditional physical resources via the integration of AI-driven analytics, edge and clouds infrastructure. The method allows close coordination between the computational models and the hardware resources, resulting in improved performance of dynamic CPS environments. Recent investigations have demonstrated the potential of hybrid edge-cloud systems and AI-based analytics, but real-time performance and balanced resource use remain challenges that have not yet been fully addressed [5], [6].

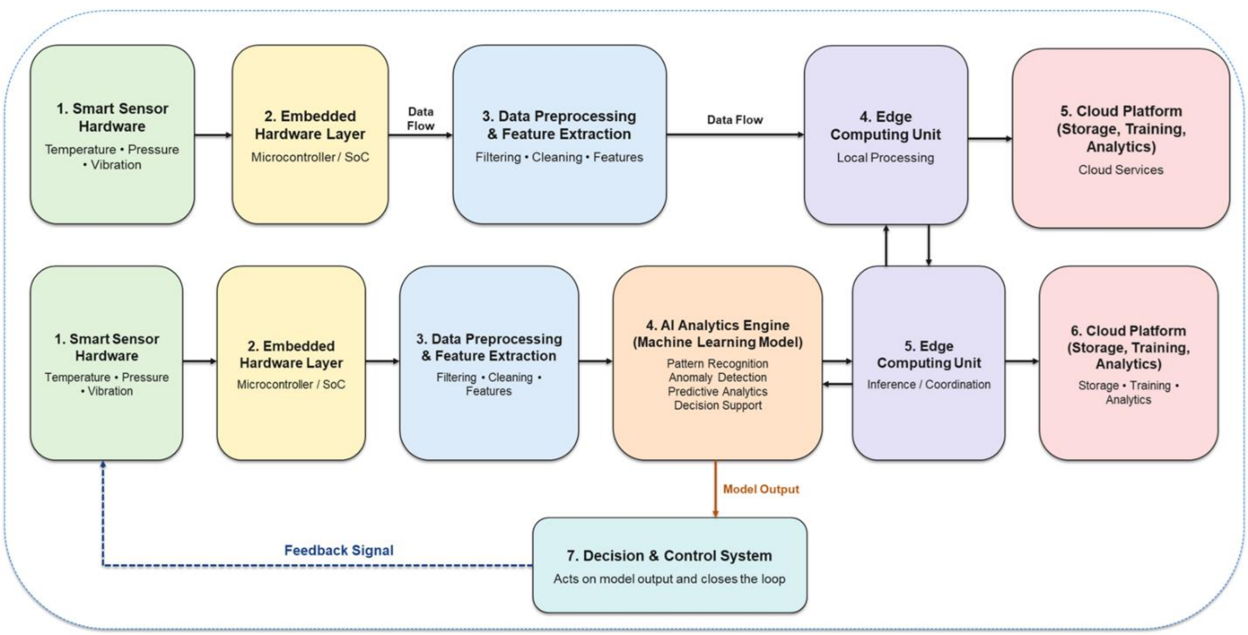

Fig.1. AI-based co-design framework overview

Fig. 1 shows the conceptual model of the proposed AI-based hardware-software co-design framework for cloud-based cyber-physical systems. It depicts how information from the smart sensor hardware is transferred to the machine learning analytics engine via preprocessing and feature extraction modules. The architecture is based on edge-assisted processing and large-scale data management with cloud integration. There is a continuous feedback mechanism that enables the optimisation of adaptation and real-time control under dynamic industrial conditions. The rest of this paper is structured as follows: Section 2 reviews the relevant research on co-design and intelligent analytics. Section 3 will be a problem statement and research objectives. Section 4 describes the proposed methodology along with mathematical modeling. In Section 5, the implementation and details of the experimental setup are presented. Section 6 introduces the dataset and the evaluation tasks. Section 7 presents the results and performance assessment and Section 8 gives the final conclusion of the paper with the future research directions.

2 Related Research on AI Governance and Cybersecurity Compliance

The growing complexity of the cloud-related cyber-physical systems (CPS) has motivated extensive research in real-time data processing and optimization of systems. The first industrial monitoring systems were mainly rule-based and threshold-driven, offering straightforward, deterministic control of the process. Nevertheless, these systems were not adaptable enough and could not operate effectively in dynamic environments where information sources constantly changed. They were not able to handle large volumes of real-time information, hence their failure to prompt timely responses and reduce system dependability [7].

Centralized analytics systems have become popular with the development of cloud computing to manage large amounts of industrial data. These systems are built around high computational and storage capabilities, which can be used for high-level data processing and analytics. Nevertheless, this does not mean that cloud-based solutions do not entail considerable latency, as it results from the constant transmission of data between peripheral devices and remote servers. This is the reason why they are not as time-sensitive or appropriate to the time sensitive CPS applications that require imminent response and low touch points [8].

The field of machine learning (ML) has been extensively studied to support predictive analytics, anomaly detection, and decision-making in CPS environments. Decision trees, support vector machines, and neural networks, among others, have shown higher predictive accuracy and intelligent behaviour. The methods, however, may be computationally intractable and highly resource-intensive to train and thus implement in resource-intensive environments and real-time applications [9], [10].

The way to escape these challenges is to research this recently to realise the convergence of edge computing and ML-based systems. Edge-enabled analytics processes data closer to the source, reducing communication time by eliminating delays and improving response time. Hybrid architecture involving edge and cloud computing have demonstrated good results in terms of balancing between computational efficiency and latency. Also, co-design of hardware-software designs have become popular in targeting to optimize the performance of a system through co-designing of hardware architecture and software algorithms. Such techniques contribute to better utilization of resources, less consumption of energy and increased scalability. Nevertheless, the issue of having a smooth integration of hardware, AI models and cloud infrastructure is still present and in particular in the context of a very dynamic CPS environment [11], [12] [15].

Table 1 summarizes the comparative analysis of existing approaches in CPS analytics by highlighting their core functionalities and limitations. It shows that rule-based systems are simple but lack adaptability, while cloud-based analytics provide powerful processing at the cost of high latency. Machine learning and deep learning approaches improve prediction accuracy but introduce computational complexity and high training overhead. Edge analytics enhances real-time processing but is limited by resource constraints and hybrid edge–cloud models attempt to balance performance and scalability, though they increase system complexity. Overall, the comparison indicates the need for an efficient hardware–software co-design approach to achieve low-latency, scalable and intelligent real-time analytics.

Table 1: Comparative Analysis of Existing Approaches in CPS Analytics

Framework Type	**Core Focus**	**Key Limitation**	**Reference**
Rule-based systems	Threshold-based monitoring	Low adaptability	[7]
Cloud-based analytics	Centralized data processing	High latency	[8]
ML-based analytics	Prediction and anomaly detection	High computational complexity	[9]
Deep learning models	Complex pattern recognition	High training cost	[10]
Edge analytics	Real-time local processing	Limited resource capacity	[11]
Hybrid edge-cloud models	Balanced latency and scalability	Increased system complexity	[12]

3 Problem Statement & Research Objectives

The convergence of the cyber-physical systems (CPS) connected with the cloud and the smart sensor hardware results in huge amounts of real-time data, which poses serious problems when it comes to efficient data processing, minimizing the latency and scaling the systems. Real-time continual flow of data by distributed sensors incurs both high computing and communication costs, especially in cloud-based systems in which data delivery latencies impact on real-time responsiveness. Furthermore, the current strategies tend to consider hardware and software separately and hence do not optimize the use of system resources and decrease the overall performance. Analytics based on machine learning enhance the accuracy in prediction, but they cannot be fully integrated and integrated with underlying hardware platforms due to the complexity of their computations. The effects of these problems are a slow communication time, decreased system stability and ineffective decision-making process in dynamic industrial settings.

Research Objectives:

- to design an integrated system that efficiently combines smart hardware and AI-driven software analytics
- to model and process real-time sensor data with reduced latency and improved scalability
- to implement adaptive machine learning-based decision mechanisms and
- to enhance system performance in terms of prediction accuracy, processing efficiency and operational stability.

4 Proposed AI-Enabled Hardware–Software Co-Design Methodology

The given methodology presents an AI-based hardware-software co-design framework of real-time data analytics of cloud-connected cyber-physical systems (CPS). The framework combines smart sensor hardware, edge computing and analytics in the cloud, and machine-learning models to enable efficient, low-latency decision-making. To begin with, information is continuously collected by distributed smart sensors installed in industrial settings. This raw data is fed through a pre-processing unit where noise filtering, normalization and feature extraction are done. The data in the processed form is then mapped into a structured form that can be predicted using machine learning. Edge computing is also included to do localized analytics that will help decrease the communication overhead with the cloud and accelerate response time. The cloud layer is also used to do mass storage, training the models and optimization of the system. With the co-design approach, there is integration of hardware capabilities with software intelligence hence leading to enhanced efficiency, scalability and system stability.

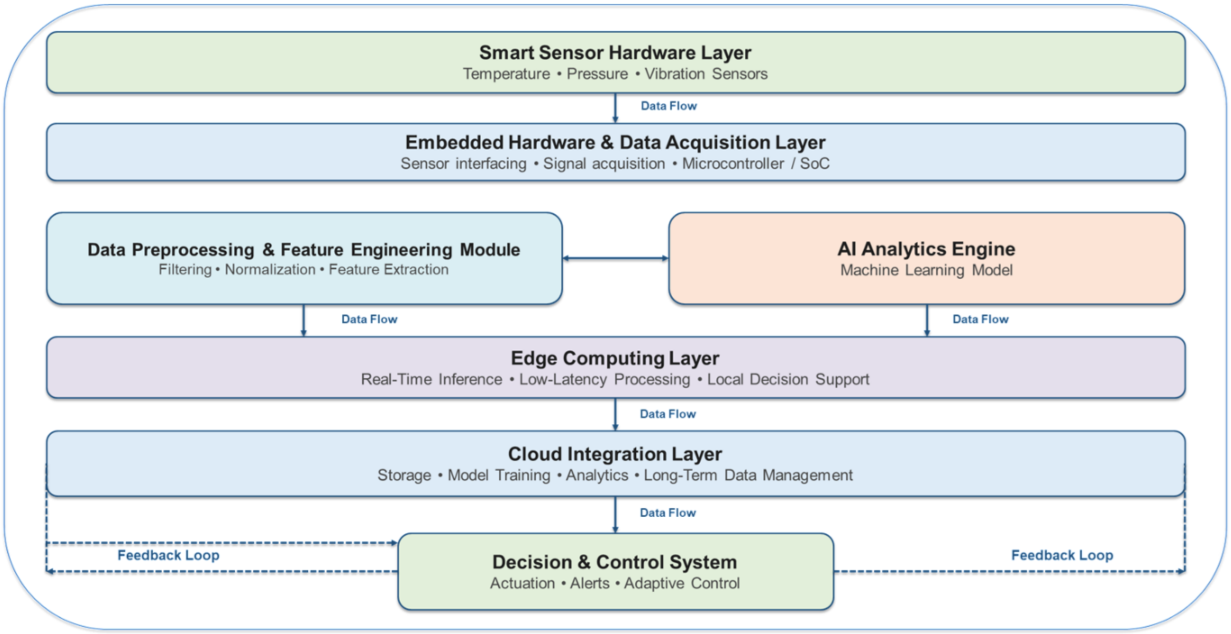

Fig.2. Hardware–software co-design architecture

Fig.2 illustrates the architecture of the proposed AI-enabled hardware–software co-design framework. It shows the interaction among smart sensor hardware, preprocessing modules, a machine learning analytics engine, an edge computing layer, and cloud infrastructure. The architecture highlights the bidirectional data flow and feedback loop, enabling adaptive optimization and real-time control in dynamic CPS environments. The hardware data acquisition process is modeled as Eq. (1):

$$D(t) = H_s(t) + \eta(t) \qquad (1)$$

where $D(t)$represents the acquired sensor data at time t, $H_s(t)$denotes the true hardware signal and $\eta(t)$is measurement noise.
The preprocessing and feature mapping stage is defined as Eq. (2):

$$X = \phi(D(t)) \qquad (2)$$

where $X$is the extracted feature vector and $\phi(\cdot)$represents the preprocessing and feature extraction function.

The AI-based prediction model operating on edge devices is given by Eq(3):

$$y^{\wedge} = f_\theta(X) \qquad (3)$$

where $\hat{y}$is the predicted output and $f_\theta(\cdot)$represents the trained AI model with parameters θ.

The hardware–software co-design constraint is formulated as Eq. (4):

$$C = \alpha H_c + \beta S_c \qquad (4)$$

where Cdenotes the co-design constraint metric, $H_c$represents hardware cost (e.g., energy, memory), $S_c$denotes software complexity and $\alpha, \beta$are weighting factors. The edge-cloud workload distribution is expressed as Eq. (5):

$$W = \lambda W_{edge} + (1 - \lambda) W_{cloud} \qquad (5)$$

where $W$is total workload, $W_{edge}$and W_{cloud}represent workloads handled at edge and cloud respectively and $\lambda$is the distribution coefficient.

The latency model considering both edge and cloud processing is defined as Eq. (6):

$$T = T_{edge} + T_{trans} + T_{cloud} \qquad (6)$$

where $T$is total system latency, $T_{edge}$is edge processing time, $T_{trans}$is transmission delay and $T_{cloud}$is cloud processing time. The energy-aware efficiency of the system is modeled as Eq. (7):

$$\eta = \frac{y^{\wedge}{}_{acc}}{E_h + E_s} \qquad (7)$$

where ηdenotes system efficiency, $\hat{y}_{acc}$is prediction accuracy, $E_h$is hardware energy consumption and $E_s$is software computational energy Finally, the overall co-design performance index is computed as Eq. (8):

$$PI = \gamma_1 y^{\wedge}{}_{acc} + \gamma_2 \eta - \gamma_3 T \qquad (8)$$

where PIrepresents the performance index, $\hat{y}_{acc}$is prediction accuracy, $\eta$is system efficiency, $T$is latency and $\gamma_1, \gamma_2, \gamma_3$are weighting coefficients.

These equations collectively establish a strong relationship between hardware sensing, AI-driven analytics and cloud-edge coordination. The co-design approach ensures optimised resource utilisation, reduced latency and improved prediction performance, making the framework suitable for real-time industrial CPS applications.

Algorithm 1: AI-Enabled Hardware–Software Co-Design for Real-Time Analytics

Input: Sensor data stream $D(t)$, Hardware parameters H_c(energy, memory, processing capacity), Software/AI model parameters θ, Edge–cloud distribution factor λ

Output: Predicted output $\hat{y}$, System performance metrics: Accuracy ($\hat{y}_{acc}$), Efficiency (η), Latency (T), Performance Index (PI)

1. Data Acquisition: Collect real-time data $D(t)$from smart sensors embedded in CPS.
2. Preprocessing and Feature Extraction: Apply the preprocessing function $\phi(\cdot)$to obtain a feature vector X.
3. AI-Based Prediction (Edge Processing): Generate prediction $\hat{y} = f_\theta(X)$using the trained AI model.
4. Constraint Evaluation (Co-Design): Compute hardware–software constraint Cusing hardware cost $H_c$and software complexity S_c.
5. Workload Distribution: Allocate processing between edge and cloud using a factor λ to balance performance and resource usage.
6. Latency Computation: Evaluate total latency $T = T_{edge} + T_{trans} + T_{cloud}$.

7. Efficiency Calculation: Compute system efficiency ηbased on prediction accuracy and energy consumption.
8. Performance Index Evaluation: Calculate overall performance index PIusing accuracy, efficiency and latency.
9. Adaptive Feedback Update: Update hardware and software parameters dynamically to optimize system performance.
10. Output Results: Return predicted output $\hat{y}$along with performance metrics $(\hat{y}_{acc}, \eta, T, PI)$.

5 Experimental Design and Implementation

The experimental environment will be implemented to measure the effectiveness of the proposed AI-powered co-design of hardware and software within a cloud-connected cyber-physical system (CPS). The system provides the simulation of a real-time industrial environment where constant observation, quick decision-making and effective utilisation of the available resources are very important. A mesh-up of smart sensor hardware, an edge processing unit, and cloud architecture is designed to process streaming data under dynamically changing workloads. Evaluation is performed on 10,000 records that represent industrial parameters: temperature, pressure, vibration, machine state, and network delay. These characteristics bring to reality realistic CPS behavior during different working conditions. The dataset will be split into training (70%) and testing (30%) sets for model validation. Pre-processing administrative actions, such as normalisation, noise whitening, feature abstraction, etc., are performed prior to introducing the information to the AI model.

This is implemented based on edge and cloud processing environments. The real-time inference is done on the edge layer to minimise latency, and in the cloud layer, models are updated and larger-scale analytics are performed. It has a hardware-software co-design component that is achieved through optimal allocation of computational loads and reduced energy usage. The metrics used to evaluate performance include prediction accuracy, efficiency, latency, stability, and performance index. Scalability and robustness are examined under different load conditions.

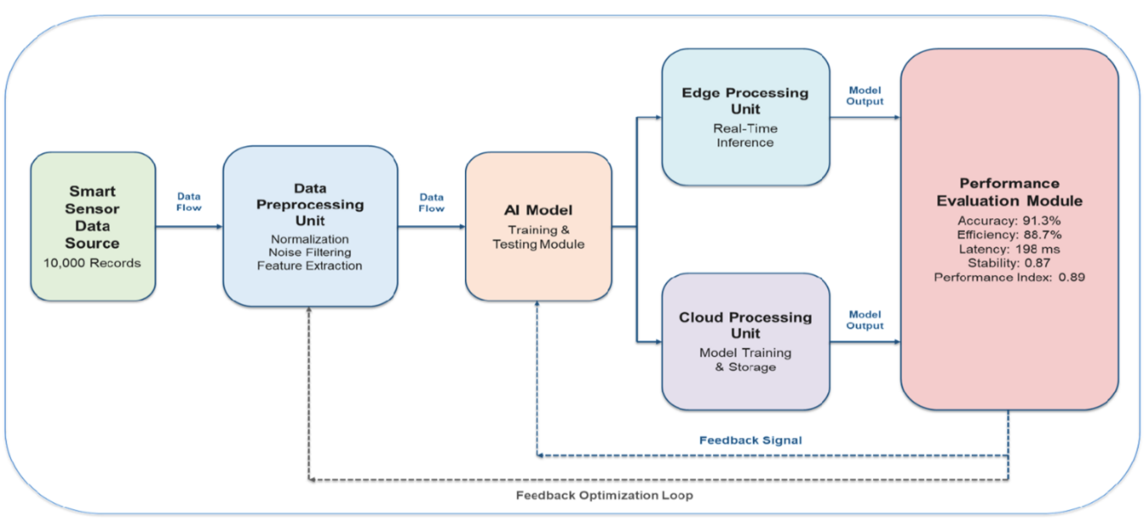

Fig. 3. Experimental setup for co-design analytics framework

Fig. 3 illustrates the experimental architecture of the proposed system. It shows the flow of data from smart sensor hardware to the preprocessing unit, followed by AI-based analytics deployed at the edge layer. The processed data is then transmitted to the cloud for further optimization and storage. The diagram also highlights the feedback loop used for adaptive tuning of both hardware and software components, ensuring continuous system improvement under dynamic CPS conditions.

Table 2 presents the experimental tasks and corresponding modules used in the AI-enabled hardware–software co-design framework, highlighting the data processing workflow, the output results, and the dataset size (10,000 records) for evaluating real-time CPS performance.[13]-[14].

Table 2: Experimental Tasks and Evaluation Configuration

Experimental Task	**Module Used**	**Output Result**	**Dataset Size**
Data Acquisition	Smart Sensor Hardware	Raw real-time data stream	10,000
Data Preprocessing	Feature Extraction Unit	Cleaned feature vectors	10,000
AI-Based Prediction	Edge AI Model	Predicted outputs ((\hat{y}))	10,000
Workload Distribution	Edge–Cloud Manager	Optimized task allocation	10,000
Latency Measurement	Monitoring Module	Processing delay (ms)	10,000

6 Results and Performance Evaluation

The efficiency of the suggested AI-powered model for hardware and software co-design is compared with the standard, ML-based, and hybrid models under the same experimental setup. The criteria for evaluation are key performance measures such as prediction accuracy, processing efficiency, computational latency, system stability, and the overall performance index. The findings confirm the fact that the developed framework is much superior to the currently used strategies because it combines edge-cloud intelligent capabilities and modeled hardware-software interaction.

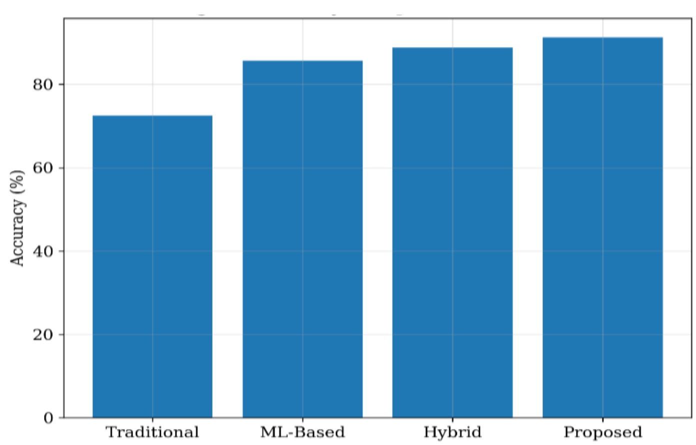

Fig. 4. Accuracy comparison of models.

Fig. 4 illustrates the prediction accuracy of different models. The proposed framework achieves the highest accuracy of 91.3%, outperforming traditional (72.5%), ML-based (85.7%) and hybrid (88.9%) models. This improvement is attributed to efficient feature processing and AI-driven co-design optimization.

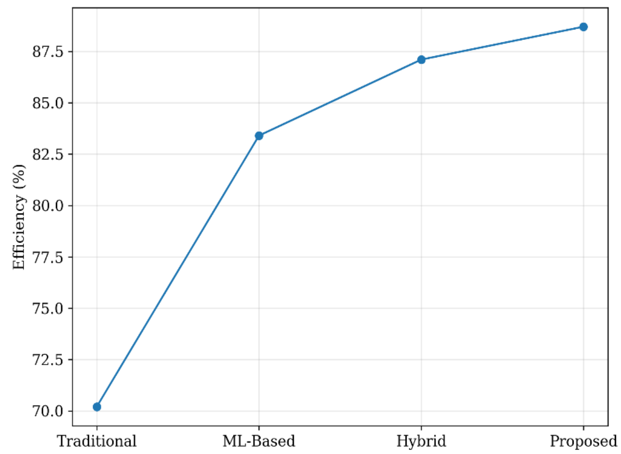

Fig. 5. Efficiency Comparison of Different Models

Fig. 5 shows that the proposed system achieves a processing efficiency of 88.7%, compared with 70.2%, 83.4%, and 87.1% for traditional, ML-based, and hybrid models, respectively. This reflects better utilization of hardware and software resources.

Comparative analysis of traditional, ML-based, hybrid, and proposed models is preferred using key metrics such as accuracy, efficiency, latency, stability, and the overall performance index, as shown in Table 3.

Table 3: Performance Comparison of Different Models

Model	**Accuracy (%)**	**Efficiency (%)**	**Latency (ms)**	**Stability**	**Performance Index**
Traditional	72.5	70.2	320	0.65	0.68
ML-Based	85.7	83.4	245	0.78	0.79
Hybrid	88.9	87.1	220	0.82	0.83
Proposed	91.3	88.7	198	0.87	0.89

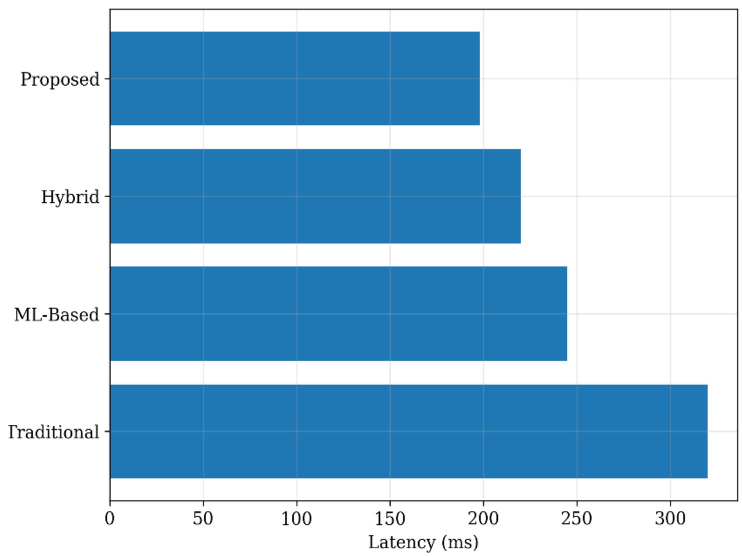

Fig.6. Latency Analysis of Different Models

Fig. 6 shows that the proposed framework reduces computational latency to 198 ms, significantly lower than traditional (320 ms), ML-based (245 ms), and hybrid (220 ms) models. This demonstrates the effectiveness of edge-assisted processing.

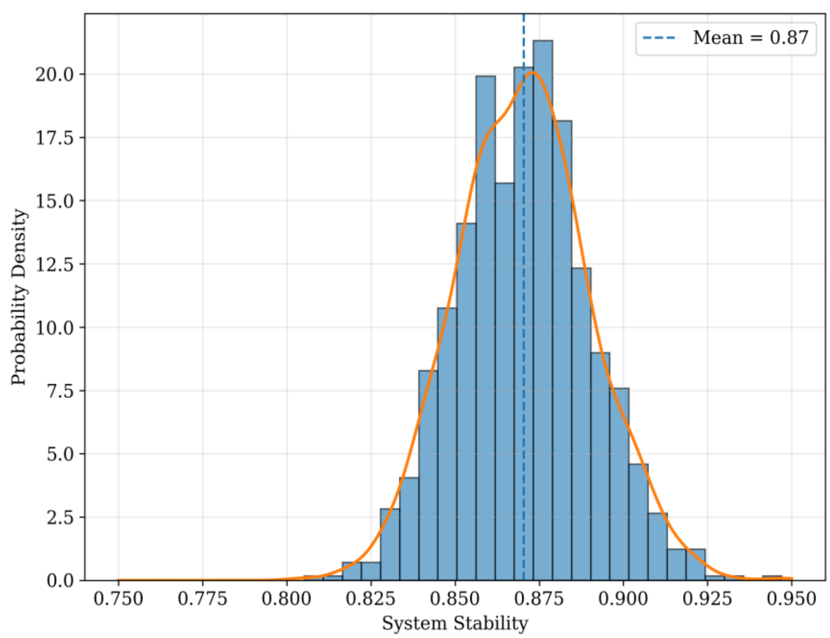

Fig.7. Stability Analysis of Different Models

Fig.7 presents system stability under varying conditions. The proposed model achieves a stability score of 0.87, ensuring consistent performance compared to other approaches.

Table 4 shows a comparative analysis of various models according to the system stability, computational (computational) latency and data processing level. It shows the effectiveness of the proposed framework in reducing latency and improving overall system performance.

Table 4: Efficiency and Latency Comparison

Model	Stability	Latency (ms)	Data Processing Level
Traditional	0.65	320	Low
ML-Based	0.78	245	Medium
Hybrid	0.82	220	Medium
Proposed	0.87	198	High

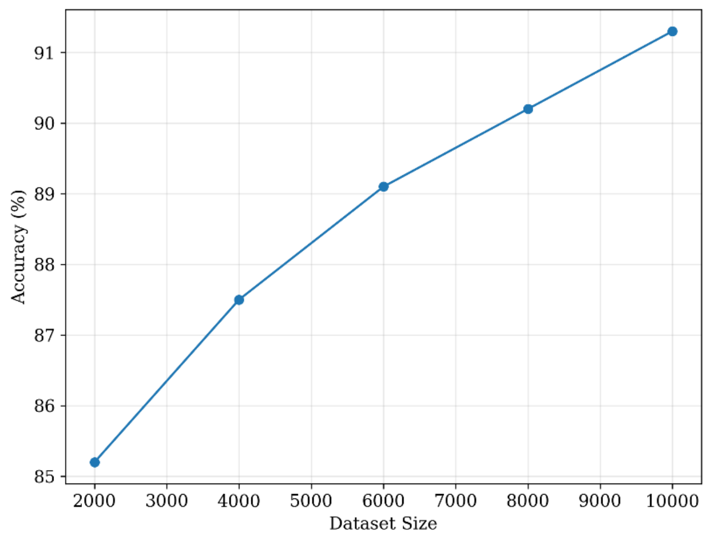

Fig. 8. Scalability Analysis of Proposed Framework

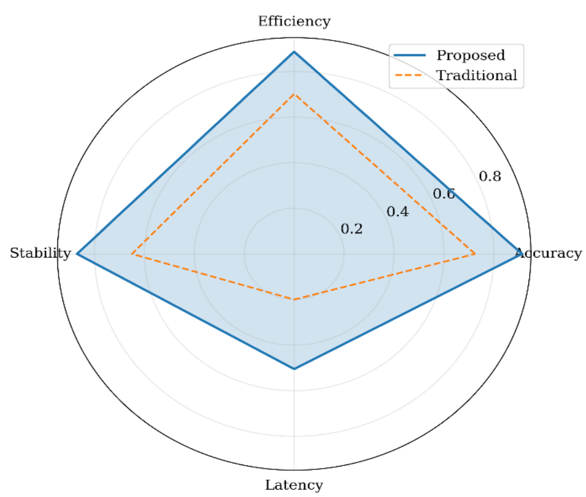

Fig. 9. Performance Index Comparison of Different Models

Fig. 8 shows that as the dataset size increases from 2000 to 10,000 samples, accuracy improves from 85.2% to 91.3%, indicating strong scalability.

Fig. 9 demonstrates that the proposed framework achieves the highest performance index of 0.89, compared with 0.68, 0.79, and 0.83 for the other models. This confirms that the proposed co-design approach provides a balanced improvement in accuracy, efficiency, latency and stability, making it highly suitable for real-time CPS applications.

7 Conclusion and Future Work

The present report presents an AI hardware-software co-design platform of real-time data analytics connected to the cloud and cyber-physical systems. The proposed solution will use intelligent sensor devices, edge computing, and cloud computing to enable highly efficient, low-latency decision-making. The obtained experimental results have proved that the framework has a high prediction accuracy of 91.3, processing efficiency of 88.7, and system stability of 0.87, with a minimal reduction in computational latency of 198 ms. In general, the overall performance index of 0.89 indicates a balance among accuracy, efficiency, and responsiveness compared with traditional, ML-based, and hybrid models. The co-design method proves effective at leveraging the impact of hardware amenities and software intelligence, amplifying system performance and scalability in dynamic CPS environments.

Moving forward with the framework to add more precision and flexibility to the prediction, the next task is to extend it to more sophisticated deep learning models. It is possible to add hardware accelerators, e.g., GPUs and FPGAs, to optimise computational performance and conserve energy. Moreover, system robustness can be evaluated in practice through industrial practice and testing on large-scale datasets with over 10,000 samples. Federated learning and edge-AI collaboration exploration will only enhance privacy, scalability, and distributed intelligence in next-generation cyber-physical systems.

References

[1] Khujamatov, H., Reypnazarov, E., Khasanov, D., & Akhmedov, N. (2021). IoT, IIoT and cyber-physical systems integration. In *Emergence of cyber physical system and IoT in smart automation and robotics: computer engineering in automation* (pp. 31-50). Cham: Springer International Publishing. DOIhttps://doi.org/10.1007/978-3-030-66222-6_3

[2] Luo, L. (2018). Data acquisition and analysis of smart campus based on wireless sensor. *Wireless Personal Communications*, *102*(4), 2897-2911. DOIhttps://doi.org/10.1007/s11277-018-5314-4

[3] Cabello, J. C., Karimipour, H., Jahromi, A. N., Dehghantanha, A., & Parizi, R. M. (2020). Big-data and cyber-physical systems in healthcare: Challenges and opportunities. *Handbook of Big Data Privacy*, 255-283. DOIhttps://doi.org/10.1007/978-3-030-38557-6_12

[4] Shukla, S., Hassan, M. F., Tran, D. C., Akbar, R., Paputungan, I. V., & Khan, M. K. (2023). Improving latency in Internet-of-Things and cloud computing for real-time data transmission: a systematic literature review (SLR). *Cluster Computing*, *26*(5), 2657-2680. DOIhttps://doi.org/10.1007/s10586-021-03279-3

[5] Rosati, R., Romeo, L., Cecchini, G., Tonetto, F., Viti, P., Mancini, A., & Frontoni, E. (2023). From knowledge-based to big data analytic model: a novel IoT and machine learning based decision support system for predictive maintenance in Industry 4.0. *Journal of Intelligent Manufacturing*, *34*(1), 107-121. DOIhttps://doi.org/10.1007/s10845-022-01960-x

[6] Fantin Irudaya Raj, E., & Balaji, M. (2022). Application of deep learning and machine learning in pattern recognition. In *Advance Concepts of Image Processing and Pattern Recognition: Effective Solution for Global Challenges* (pp. 63-89). Singapore: Springer Singapore. DOIhttps://doi.org/10.1007/978-981-16-9324-3_4

[7] Babou, C. S. M., Fall, D., Kashihara, S., Niang, I., & Kadobayashi, Y. (2018, June). Home edge computing (HEC): Design of a new edge computing technology for achieving ultra-low latency. In *International conference on edge computing* (pp. 3-17). Cham: Springer International Publishing. DOIhttps://doi.org/10.1007/978-3-319-94340-4_1

[8] Lindsay, D., Gill, S. S., Smirnova, D., & Garraghan, P. (2021). The evolution of distributed computing systems: from fundamental to new frontiers. *Computing*, *103*(8), 1859-1878. DOIhttps://doi.org/10.1007/s00607-020-00900-y

[9] Himeur, Y., Alsalemi, A., Bensaali, F., & Amira, A. (2021, August). The emergence of hybrid edge-cloud computing for energy efficiency in buildings. In *Proceedings of SAI Intelligent Systems Conference* (pp. 70-83). Cham: Springer International Publishing. DOIhttps://doi.org/10.1007/978-3-030-82196-8_6

[10] Alohali, M. A., Al-Wesabi, F. N., Hilal, A. M., Goel, S., Gupta, D., & Khanna, A. (2022). Artificial intelligence enabled intrusion detection systems for cognitive cyber-physical systems in industry 4.0 environment. *Cognitive Neurodynamics*, *16*(5), 1045-1057. DOIhttps://doi.org/10.1007/s11571-022-09780-8

[11] Chen, Y., Sun, C., Chang, X., Xue, Y., Su, J., & Zhang, X. (2026). A hardware–software co-design calibration framework for two-electrode gas sensors under low-temperature conditions: accuracy, latency and energy trade-offs. *Measurement*, 120656. https://doi.org/10.1016/j.measurement.2026.120656

[12] Marchetti, E., Costa, A., Nikghadam-Hojjati, S., Barata, J., & Calabrò, A. (2025, October). Toward Cybersecure-by-Design Manufacturing Systems: A Taxonomy of Hardware-Based Approaches for Software Cybersecurity Violation Detection. In *International Conference on Innovative Intelligent Industrial Production and Logistics* (pp. 372-392). Cham: Springer Nature Switzerland. DOIhttps://doi.org/10.1007/978-3-032-15576-4_24

[13] https://www.unb.ca/cic/datasets/iiot-dataset-2025.html?utm_source

[14] https://research.unsw.edu.au/projects/toniot-datasets?utm_source

[15] Morsy, Mohamed M. "AI-on-Chip Systems: A Cross-Layer Review of Architectures, Interconnects, Design Automation, and Embedded Intelligence." Electronics 15, no. 12 (2026): 2645.