10.48047/jocaaa.2026.35.01.62

Autonomous Cloud-Native Ingestion of High-Frequency MQTT Telemetry for Predictive Anomaly Intelligence in Next-Generation Automotive Powertrain Manufacturing

Yogesh Pugazhendhi Duraisamy Rajamani
Independent researcher, USA

Abstract

Modern powertrain manufacturing environments generate enormous amounts of high-frequency telemetry data from engines and transmissions that require real-time streaming analytics and scalable storage. A cloud-native analytics platform that supports the continuous ingestion of MQTT-based streaming powertrain telemetry data from manufacturing lines can support predictive anomaly detection for a variety of components. It is designed to use distributed ingestion pipelines, event-driven microservices, and tuned data engineering layers to achieve low-latency transport, fault tolerance, and schema-aware data processing. Sensors stream to cloud analytic platforms, which use feature engineering, multivariate time-series modeling, and ensemble-based anomaly detection orchestrated by workflow engines. The architecture supports closed-loop feedback, where a model is improved through operator feedback and automated retraining (as pictured). The results provide sub-second ingestion-to-detection latency and throughput well above typical rates, in addition to high precision fault detection across temperature, vibration, pressure, and timing sensors. The system's AI-driven architecture has applicability for predictive maintenance, production optimization, and self-optimizing quality assurance within high-scale automotive factories, acting as a blueprint for smart factories of the future.

Keywords: MQTT ingestion, predictive anomaly detection, cloud-native streaming, automotive manufacturing, real-time telemetry

I. Introduction

1.1 Challenges of Industrial IoT in Automotive Production

The Industrial Internet of Things technologies, combined with cloud computing, have given automotive manufacturers the ability to build architectures to connect, automate, and manage operations by leveraging data analytics. The assemblage of engines and gearboxes within drivetrain assembly plants is performed via thousands of sensors placed in various types of mechanical components to provide realtime telemetry data via thermal profiling, vibration signatures, hydraulic modalities, and the synchronization of rotating masses. These measurement infrastructures produce events at a high rate that need to be instantly processed to support maintenance prediction, defect prevention, and continuous process improvement. However, despite the proliferation of sensors in production facilities, there is often a meaningful gap between raw measurement data and operational-level intelligence observed [1]. Heterogeneous sensing devices, sampling intervals, and data formats are common in manufacturing plants, which makes it imperative to design data ingestion frameworks that can adapt to such differences in sensing information while ensuring the consistency and reliability of the captured data.

1.2 Constraints of Traditional Batch Processing Architectures

In contrast, most legacy data management infrastructures are batch-oriented, gathering sensor data at field stations, periodically aggregating and forwarding the data to a central repository for asynchronous analysis.

10.48047/jocaaa.2026.35.01.62

This is adversely affected by high latency between an event's occurrence and its computational analysis, precluding timely reactions to operational anomalies or catastrophic system failures and proactive maintenance based on equipment degradation. A delay between the creation of information and the understanding drawn from it is one of the reasons real-time measurement streams are not relied upon when it is most critical. For example, component failures in manufacturing can spread through production lines and cause quality issues [2]. These limitations are manifest in the case of powertrain manufacture, where drivetrain assembly must withstand mechanical stress, thermal cycling, and tolerances that give rise to incremental wear patterns that by their nature appear as small deviations in measurements. Another consideration in a rapidly changing environment is that emergent anomalies need detection capabilities in real-time with sub-second responsiveness that are not available with customary batch-mode systems.

1.3 Objectives and Novel Contributions

The MQTT messaging protocol has become the de facto standard within industrial sensor networks as its low-overhead publish-subscribe event notifications and ability to withstand limited network availability addresses industrial concerns. However, bridging an MQTT factory instrumentation with a cloud-scale analytics platform introduces several architectural challenges in message format translation, event ordering, computational state management, and fault tolerance. This paper provides a complete, cloudnative architecture for the continuous ingestion, transformation, and evaluation of MQTT telemetry from automotive powertrains in automotive manufacturing. The architecture combines distributed stream processing engines, anomaly detection using machine learning, and bi-directional feedback to provide real-time operational intelligence for continuous manufacturing processes. These components are an ingestion pipeline for horizontally scaling an MQTT message broker to cloud streaming services, the feature engineering algorithms that discover patterns in raw sensor data, this ensemble-based anomaly detection algorithm for combining statistical and neural models, and an adaptive refinement loop to continuously improve detection based on operator feedback and updated models.

1.4 Document Structure

The remaining sections of this paper present details on the architecture components, their specifications, and the performance of the proposed architecture. Section II presents a literature survey on contemporary industrial IoT architectures, distributed stream processing technologies, and manufacturing anomaly detection techniques. In section III, It describes the complete system architecture, from sensor instrumentation to cloud analytics platforms, to feedback control systems, and finally to actuator outputs. Section IV describes workflows from information ingestion to understanding generation, and Section V lists processing latency, message throughput, detection accuracy, and fault tolerance characterization as the performance metrics measured for evaluating the overall system performance. Section VI concludes the paper, highlighting key points discussed and possible future research directions in smart manufacturing systems.

II. Background and Related Work

2.1 MQTT Protocol in Industrial IoT Environments

The Message Queuing Telemetry Transport (MQTT) transport protocol is a prevailing industrial Internet of Things communications protocol. It is standardized for the use case of sending data over networks with limited bandwidth or other resources. MQTT is a lightweight messaging protocol based on a publishsubscribe (pub/sub) communication model. The use of small packet format and message size, factors that reduce the volume of data that is transmitted, leads to some useful properties for industrial applications. Topics with long periods of constrained connectivity bandwidth and reliability can be served using MQTT, which has multiple levels of Quality of Service (QoS) that permit trade-offs between message reliability

10.48047/jocaaa.2026.35.01.62

and network bandwidth. The broker-based architecture allows for messages to be published and delivered in a consolidated way and allows for horizontal scaling to meet the requirements of a larger sensor network [3]. Production implementations also leverage MQTT's session continuity features to provide subscription statefulness across sporadically connected sensors. The hierarchical filtering of topics allows for a nested organization of the data flows from the sensors, making it easier for downstream applications to only consume relevant data streams and experience less telemetry overhead. The impact of moving the MQTT messages from the edge brokers to cloud-native information pipelines on protocol behavior, message ordering, and state coordination should be assessed.

Protocol	Architecture	Message Overhead	QoS Support	Network Tolerance	Typical Use Case
MQTT	Publish-Subscribe	Minimal	Three Levels	High	Factory sensor streaming
HTTP/REST	Request-Response	Moderate	None	Low	Periodic data polling
CoAP	Request-Response	Low	Two Levels	Moderate	Constrained devices
AMQP	Message Queue	High	Multiple Levels	High	Enterprise integration
WebSocket	Bidirectional	Moderate	None	Moderate	Real-time dashboards

Table 1: Comparison of Communication Protocols for Industrial IoT Sensor Networks [3, 9]

2.2 Cloud-Native Data Platforms for Streaming Telemetry

Cloud-native infrastructure has provided elastic compute and managed infrastructure, which has decoupled the processing layer from the hardware and enables a cloud-native streaming ingestion architecture with capabilities to process large amounts of events and perform low-latency manipulation and processing. These platforms hide the details of the underlying infrastructure and enable analytics developers to focus on transforming data, rather than managing clusters and provisioning resources. Distributed stream processing engines in the cloud support stateful processing of data streams over time, enabling complex event processing and pattern detection on high-volume streams from sensors. Cloud architectures, decoupling persistence from computation, allow reception and manipulation/analysis processes to be separately tuned for different objectives. Managed queuing infrastructure provides persistent queues in the path between reception and manipulation to buffer events and tolerate interruptions or downtime that may occur in downstream manipulation or analysis. Cloud-native monitoring capabilities provide full observability of conduit vitality, message throughput, and latency of manipulation across the distributed constituents of the system [4]. Dynamic scaling capabilities allow for the adaptation of the scale of the computing resource to the sensor density cost-effectively while meeting performance requirements. The serverless execution patterns enable event-driven manipulation designs where analytical processes are only invoked when messages are received, thereby reducing operational overhead in sporadic workloads.

2.3 Predictive Maintenance and Anomaly Detection Techniques

Machine learning predictive maintenance approaches can identify more complex patterns of degradation that cannot be detected by threshold-based monitoring systems. Ordinary statistical methods such as moving

10.48047/jocaaa.2026.35.01.62

averages, control limits, and seasonal decomposition can be used to detect anomalies for processes with well-understood operating behavior that does not vary over time. More advanced approaches, such as isolation forests or local outlier factor algorithms, can be used to detect multivariate anomalies in high-dimensional sensor data without labeled training data. Another approach is to model sensor behavior using a time series prediction model, such as autoregressive integrated moving average and exponential smoothing, and flag variances as anomalies. Deep learning architectures such as long short-term memory networks and temporal convolutional networks can model temporal patterns in timeseries data by learning normal temporal patterns from historical data. Autoencoder architectures can learn low-dimensional representations of sensor data and are used for anomaly detection by identifying a reconstruction error to detect outliers. Ensemble methods, or combining several recognition methods, can improve robustness to false positives and allow bi-modal responsiveness to true positives in the event of different failure scenarios [4]. Online learning approaches continuously update recognition models as new sensor data arrive, accommodating changes in operating conditions or seasonal factors, without requiring wide-ranging retraining. Selecting the appropriate recognition method depends on the properties of the information to be recognized, such as the sampling velocity, the dimensionality, the stability, and the presence of labeled failure specimens.

2.4 Gap Analysis: Need for Unified MQTT-to-Cloud Architecture

While literature on industrial data pipelines for individual components exists, as of this writing, there has not been an attempt at creating a complete architecture spanning from the MQTT reception over cloudnative stream processing to production-grade anomaly detection. Existing literature has focused mainly on MQTT broker architectures at the edge or cloud-native stream processing without addressing the architecture itself. Published frameworks still often present proof of concept integrations in a simulation or with data representation of simplicity, and not on a production scale and a heterogeneous set of sensors, variable sampling frequencies, and constraints of the real network present in automotive production [3]. Fairly less work is also done on bridging the difference in the semantics of MQTT's topicdriven routing and partition-dependent routing of cloud streaming architectures, message ordering, and addressing the information asymmetry due to concurrent manipulations. However, the irregularity recognition literature is typically evaluated on standard data sets rather than a streaming reception conduit, an instantaneous inference circuit, and a responsive operational response circuitry for production deployment. Furthermore, the manufacturing process involves dynamic transitions among different versions of products and different stages of manufacture, requiring recognition circuits that are responsive to these transitions. This paper systematically analyzes the design implications of recognition delay requirements on model complexity and computational cost for a cloud-based service with fractionalsecond latency requirements from potentially thousands of sensor streams. From MQTT ingestion to responsive irregularity recognition, we evaluate an integrated design on production-scale automotive manufacturing telemetry, rather than on synthetic datasets.

III. Proposed System Architecture

3.1 MQTT Sensor Layer and Topic Organization

The sensing backbone is implemented as distributed sensors attached to engine components such as engine blocks, casings from gearboxes, crankshaft components, clutch components, and oil flow lines in motor vehicle automatic transmission manufacturing plants. The sensors produce telemetry feeds representing temperature patterns, vibrations, pressures, shaft rotations, torque readings, and positioning signals in the manufacturing processes. Each sensing unit publishes measurement readings onto a chosen MQTT topic derived from a topic hierarchy consisting of placement, component, measurement type, and resolution,

10.48047/jocaaa.2026.35.01.62

allowing subscribers to easily target only the data streams that they are interested in without having to filter out streams provided by other sensing units. Sensing nodes support QoS schemes, allowing for a trade-off between reliable data delivery and the associated network traffic based on the measurement. Safety-critical metrics are sent at a higher QoS than nominal metrics [5]. The publishersubscriber decoupling allows for changes to be made to the sensors without affecting unrelated sensor subscribers. By placing MQTT brokers where the factories are, regionally located manufacturing locations can aggregate the connection state and message routing actions of the various manufacturing plant network divisions before the aggregated message sequences are routed to reception locations in the cloud, reducing network congestion and ensuring that the message sequence is retained in the topics.

3.2 Cloud Ingestion Layer Components

The cloud reception stratum connects world-facing MQTT brokers with MQTT brokers on the factory floor, providing protocol translations that preserve message characteristics and transmission assurances. Provisioned IoT gateway features authenticate the sensors' telemetry using certificate-based protocols, maintain a list of the sensors, their software versions and permission zones, and convert protocol payloads from MQTT into cloud event formats suitable for other processing technologies. Messages are buffered in holding queues in the reception layer to absorb short-term surges in load during fabrication ramp ups or service outages, so brokers in the plant don't experience backpressure. Distribution algorithms send message sequences to as many reception endpoints as appropriate, to eliminate single points of failure and maximize potential capacity. Dynamic scaling regulations monitor queues and processing delays, creating new resources as needed. The stratum includes message deduplication operations to reduce the impact of duplicate events caused by network retransmission and broker switching. Reception capability on the structure tags all messages with time of arrival and tracking information to enable delay and distributed analysis by subsequent conduit phases. Schema verification is applied at the reception perimeters, discarding corrupted messages before they propagate through later manipulation. This reduces the overhead of corrupted sensing data and prevents it from propagating into other areas.

3.3 Real-Time Stream Processing Framework

Stream manipulation infrastructure continually manipulates the sensor telemetry via temporal segmentation, stateful consolidations, and characteristic extractions over monolithic streams. On distributed worker modules, distributed manipulation motors perform partitions of message streams. Generally, as a consequence of the logging procedures followed by the motors, state is preserved to allow fault-recovery procedures to avoid re-manipulating previously ingested information. Duration-dependent segmentation operators group sensor data into progressive or discrete durations to compute estimators such as moving averages, variance approximations, or percentiles that vary by duration. Stream augmentation attaches situational data such as product definitions, manufacturing schedules, and equipment service histories to the underlying measurements, enabling contextual understanding of sensor data and anomaly detection. Watermarking techniques are used to deal with out-of-order message delivery from distributed sensors, ensuring that temporal consolidations across regions of the plant remain consistent, despite variable latencies in the network. Stateful pattern matching is used to detect high-level events that may span multiple sensors and time periods, taking into account certain features of plant behavior. Examples of domain-specific transformations that are engineered into telemetry include computing frequency spectrum components, wavelet parameters, and domain-specific parameters of statistical distributions. Multiple such branches of transformed telemetry are sent to various downstream consumers, including immediate visualizations, persistence stores (for tracking telemetry over time), and machine learning inference pipelines that flag anomalies. Manipulation configurations can vary and can be adjusted dynamically by

10.48047/jocaaa.2026.35.01.62

redistributing divisions and worker quantities based on changes in sensor concentrations throughout the fabrication.

Framework	Processing Model	State Management	Windowing Support	Fault Recovery	Scalability Pattern
Apache Beam	Unified Batch-Stream	Checkpointing	Flexible	Automatic	Horizontal
Spark Structured Streaming	Micro-batch	Distributed	Time-based	Checkpointbased	Horizontal
Apache Flink	Event-driven	Distributed Snapshots	Session/Sliding/Tumbling	Savepoints	Horizontal
Kafka Streams	Event-driven	Local State Stores	Time/Count-based	Changelog Topics	Horizontal

Table 2: Cloud-Native Stream Processing Framework Comparison [4, 10]

3.4 Predictive Anomaly Detection Layer

The anomaly recognition stratum includes ensemble anomaly-based detection with statistical boundary definition, isolation forest calculations, and recurrent neural network architecture. This follows the principle of generalizing recognition to all failure modalities. Orthogonally, the statistical boundary assigns each sensor the nominal functional parameters, based on historical percentile values, and flags all readings exceeding adaptive limits according to fabrication conditions and external stresses. Isolation forest projects a feature space, recursively partitioning it by isolation. The more isolated an observation, the more partitions it would require to separate it from the tightly clustered groups found in the standard operation. Long short-term memory networks capture temporal dependencies between sensor states, predicting future observations, and calculating the reconstruction error for anomaly detection [6]. Autoencoders compress multivariate sensor arrays into hidden representations. Anomaly detection is performed through reconstruction error when input data is anomalous regarding the threshold value applied to reconstruction error. Ensemble approaches combine the outputs of the individual detectors by weighted majority voting, thereby increasing specificity without loss of sensitivity to outliers. In addition to combining the ensemble, the stratum uses adaptive boundary procedures to adjust the level of recognition sensitivity according to the level of fabrication, product geometry, and past false alarm rates. Representation inference uses serverless operation activations triggered by characteristic arrays from incoming signals and is elastic to the speed of sensor sequences. No permanent computational infrastructure is needed. Recognition outputs include anomaly ratings, contributing sensors, timing context, and certainty ranges to enable operators to understand the event and sequence a response. Persistent representation observation includes prediction accuracy, feature divergence, and concept drift, which trigger retraining processes if performance deteriorates beyond acceptable limits.

3.5 Storage and Analytics Infrastructure

The persistence framework implements multi-tier information retention strategies that balance retrieval latency, inquiry execution time, and persistence cost for operational, analytical, and archival workloads. The active persistence tier retains current telemetry in high-throughput time series data stores that are

10.48047/jocaaa.2026.35.01.62

optimized for time-centric queries and instantaneous recovery for charting and anomaly assessment. These are organized by columnar compression configurations that combine temporal configurations with sensor value allocations to achieve both high persistence concentration and investigative responsiveness. Intermediate persistence migrates matured telemetry into object persistence configurations, organizing the data across partition configurations for analytical retrievals, batch processing operations, and historical trend analyses. Frozen archival persistence would keep the legally mandated archives and reference data in cost-effective data stores with a data recovery time objective of days, used for occasional regulation filing audits or configuration change retention. Metadata inventories manage schema catalogs, information lineage, divisional measurements, and can answer regulatory or operational inquiries and automated data lifecycle management policies. Analytical motors apply SQL and procedural changes to historical telemetry to produce aggregated or correlated measures or operational summaries that would be passed to the manufacturing scheduling infrastructure. Raw sensor recordings in information reservoir designs are retained in an immutable configuration for retrospective remanipulation whenever recognition calculations advance or new analytical demands arise. Access regulation enforces role-based permissions restricting knowledge of confidential fabrication processes while allowing analytical cooperation across operations. Reserve duplication extends product longevity across geographical footprints, enabling reliable post-catastrophe recovery or meeting regulatory compliance requirements like data residency.

3.6 Workflow Execution Pipeline

Holistic information displacement from sensor transmission until reported anomalies issuing in the operation coordination stratum leverages event-driven initiation configurations reliant on management attributes across distributed conduit elements. Upon the arrival of MQTT messages, reception operations with protocol conversion and confirmation capabilities are invoked, followed by event assignments to streaming infrastructures. Stream manipulation operations fire persistently, consuming stream divisions and emitting modified characteristics to subsequent analytics capabilities as defined by the manipulation topology. Representation inference operations fire upon the presence of characteristics in a characteristic array. They apply enabled representation recognitions and emit anomaly assessments along with certainty measurements and situational metadata. Alert production operations use anomaly ratings and severity limits to generate operator announcements. These contain sensor designators, the nature of the anomaly, the historical situation, and advice about where to investigate [5]. Multi-channel propagation includes integrations with manufacturing implementation infrastructure, operator applications on portable devices, and aggregated observation visualizations. Response operations interpret the operator response to anomaly alerts, and log the success/failure of the response (false positives, spurious alarms, and uncertain events requiring further investigation). Representation retraining operations correlate operator response and recognition execution measurements, leading to scheduled or event-triggered changes to representations based on operational conditions and confirmed anomaly signatures. The coordination stratum schedules retrying and circuit-breaking settings controlling temporary outages in external dependencies, reducing cascading outages caused by the transient unavailability of components. Implementing operational telemetry allows observability infrastructures to measure latencies, throughputs, errors, and resource consumption, enabling operational troubleshooting and capacity management decisions to be made.

IV. Implementation and Operational Workflow

4.1 End-to-End Data Flow from Sensors to Insights

Instrumentation is provided on drivetrain fabrication machinery, continually reading telemetry data from the equipment and publishing the information to MQTT topics based on classification and physical location, which the MQTT brokers at the plant ingest. The MQTT brokers also manage the connectivity of distributed

10.48047/jocaaa.2026.35.01.62

sensor clusters and local message flows, and storage during network communications interruptions. The brokers merge their telemetry sequences and forward them to one or more reception gateways in the cloud over transport tunnels, preserving the message sequence and timing from the plant to the cloud. The reception gateways validate incoming connections, check the message structure, and convert MQTT payloads into consistent event structures for delocalized stream manipulation infrastructures. Confirmed events are segmented across administration streaming infrastructures, into a plurality of manipulation modules by sensor or apparatus groupings for the parallel behavior analysis of independent telemetry progressions. Stream manipulators are responsible for the temporal segmentation and statistical and tendency calculation of raw sensor progressions ornamented with situational metadata and reside in fabrication implementation infrastructures and apparatus catalogs [7]. Augmented characteristic arrays feed into machine learning inference operations that produce probabilistic anomaly scores from the ensemble recognition representations with their estimated variances and sensors. Tracking anomalies beyond set thresholds triggers alert generation operations that produce operator alerts containing diagnostic status, past state information, and recommended responses. Alerts interoperate with fabrication regulation infrastructures, mobile operator applications and integrated observation visualizations, enabling rapid humans-in-the-loop capability for mechanized pattern recognition to notify on potential apparatus failures or quality variances. The process from sensor read to operator alert occurs with sufficient time allocations to enable preventive action before problematic conditions evolve into fabrication disruptions or element deterioration.

4.2 Feature Extraction and Enrichment Processes

With the acquisition of raw sensor telemetry, domain-specific features are modeled to improve anomaly detection beyond static thresholds applied to individual measurements. Temporal aggregation operators derive aggregated statistics over time windows, including mean values, measures of dispersion, and minimum and maximum values over time windows, yielding short-term trends and variability patterns in sensor streams. Frequency domain methods use Fast Fourier Transform algorithms to calculate primary frequencies and harmonics of vibration and acoustic sensor signals to detect mechanical defects, bearing failures, and positional conditions that are difficult to observe with duration domain data. Wavelet decomposition techniques decompose sensor measurements into multiple resolutions to distinguish transients from normal running conditions and make localized anomalies detectable even when averaging methods do not help. Cross-sensor association estimations provide information across temperature, pressure, and oscillation measurements. Correlated variances imply complete breakdowns rather than isolated sensor faults. Velocity-of-modification estimations detect the time derivative of sensor measurements. Rapid shifts imply sudden equipment condition changes or the initiation of breakdown progression. [7]. Augmentation stratum appends situational attributes mined from fabrication schedules, product specifications, equipment service histories, and environments to characteristic arrays. Such context in the anomaly provides awareness of the expected variation of product families, construction stages, and cyclic attributes in the functional attributes expressed as characteristic arrays. Fabrication implementation infrastructure integrations provide real-time condition awareness of work orders, equipment configurations, and actions, allowing recognition representations to distinguish normal operational condition transitions from operational anomaly conditions. Each apparatus asset catalog contains metadata that describes sensor adjustments, measurement timeframes, and prior breakdown configurations to ease adaptive recognition responsiveness based upon asset maturity and service condition. Such augmentations of raw telemetry produce semantically rich characteristic manifestations fitting for advanced analysis made possible by machine learning.

10.48047/jocaaa.2026.35.01.62

Feature Category	Transformation Method	Temporal Window	Application Domain	Anomaly Sensitivity
Statistical Aggregates	Mean, Std Dev, Min, Max	Sliding	All sensors	Moderate
Frequency Components	Fast Fourier Transform	Fixed	Vibration, Acoustic	High
Wavelet Coefficients	Discrete Wavelet Transform	Multiresolution	Transient events	Very High
Rate of Change	Temporal Derivative	Short-term	Temperature, Pressure	High
Cross-Correlation	Pearson Correlation	Sliding	Multi-sensor fusion	Moderate
Trend Indicators	Linear Regression Slope	Long-term	Degradation patterns	Moderate

Table 3: Sensor Data Feature Engineering Transformations [7]

4.3 ML Inference and Anomaly Detection Mechanisms

The recognition framework is a heterogeneous ensemble of multiple recognizers that systemically and simultaneously analyze the incoming characteristic arrays through statistical, unsupervised, and deep neural networks so as to provide full detection coverage across multiple breakdown expressions. The statistical recognizers define operational bounding parameters in terms of a series of percentiles on the reference time windows from past data, while adapting to the operating conditions of manufacturing and the time of the cycle. Isolation forests rely on ensembles of random decision trees that are partitioned based on features. The ratings of found anomalies are then calculated based on the distances taken to isolate them. Shorter distances show that anomalies are further away from groups of standard operation readings. One-class support vector machines obtain determination perimeters around standard operation territory in high-dimensional characteristic territory. Readings falling outside those perimeters are called anomalies. Long short-term memory autoencoders encode sequences of sensor data into hidden representations and decode them. This reconstruction error can be used to detect anomalies; in configurations that are 'out of the ordinary', the interpretation loss will be higher. The variational autoencoder generalizes this idea by using a probabilistic hidden representation and estimating the probability of a configuration being abnormal relative to the learned distribution [8]. To this end, ensemble consolidation combines the single recognizer predictions via a weighted ballot weighted over the confidences and historical accuracies of each recognizer in order to annul false positives while remaining sensitive to true anomalies that would evade singular point-in-time-based approaches. The inference layer is exposed via containerized prediction instances with dynamically elastic scaling based on the incoming rates of the characteristic arrays, achieving sub-second inference times, albeit variable across fabrication timelines. The output includes anomaly scores per sensor, ranked attributes from each sensor that exhibit abnormal behavior, temporal segments of configuration evolution, and certainty ranges for each prediction. This detailed output allows the operator to comprehend the decisions made and isolate the predicted high-consequence alerts from secondary, non-critical warnings that require further evaluation.

10.48047/jocaaa.2026.35.01.62

4.4 Closed-Loop Feedback and Adaptive Learning

In a mutual response function, the operator's response to an anomaly alert iteratively improves the recognition representation and its operational relevance. The operator assesses the recognition in the alert display interface as a true positive signifying that there is an actual problem with the apparatus, a false positive signifying variations in the behavior of the operator or apparatus within normal run-of-the-mill limits, or as questionable and therefore further investigable. The confirmation types are pooled with the first readings, anomaly scores, and contextual information to form repositories of labeled datasets used for supervised representation optimization. Periodic retraining procedures using these repositories may update the recognition representation specs, the ensemble weights towards more reliable recognizers, and the tolerance levels to minimize false alarms [7]. Active learning strategies re-train models on samples of the response where uncertainty is highest or where operator assessments deviate from representation classifications, and favor scenarios closer to the perimeter on which the model is not easily confident. Another method is notion deviation observation, where model performance metrics are tracked over increasing fixed duration time intervals to observe performance degradation at operational condition transitions or onboarding new equipment with new sensor setups. If deviations exceed predefined thresholds, mechanized retraining automatically begins. The current operational measurements are processed to create new representations that characterize the current fabrication environment without any operator intervention. This feedback loop, which also contributes to characteristic engineering, allows operators to label measurements that were most informative during failure investigations. This allows the choice of characteristics to be continually re-engineered, to choose measurements with the most information for a given set of measurement points, discarding noisy or redundant measurements. This, in turn, helps ensure systemic awareness infrastructures better reflect real-world operating conditions and operator needs, rather than degrading over the course of fabrication operations.

4.5 Alert Generation and Operator Integration

Discovered anomalies are then rated for their severity and context of occurrence and presented to the fabrication operators via alert delivery systems. Severity rating factors include the anomaly rating, criticality of the affected equipment, impact on the fabrication schedule, and historical breakdown impact, ranging from benign informational notification to immediate-procedure execution alerts. Alerts include deep diagnostics information, such as affected sensor names and fault severity levels, timestamps of configuration changes before fault detection, similar past fault scenarios with efficient responses, and diagnostic suggested investigations categorized by failure mechanism. Multi-channel alerting procedures annotate faults by severity levels and operator locations. High-severity alerts drive immediate notifications to mobile applications and factory implementation infrastructure, while lower-priority cases aggregate into visualizations for batch review [8]. Infrastructure incorporations enable automatic resolutions to fabrication implementation infrastructure events like apparatus slowdown, start of quality checks, and preventive service task creation. Portable operator applications notify local staff of impacted apparatus and indicate how the nearest service staff can rapidly service issues, using maps of the plant with highlighted affected apparatus. Consolidated observation (consoles) visualizations enable alerts over fabrication sequences, aggregating summary measurements, tendency visuals, and apparatus vitality readings. They also help with supervisory oversight and asset dispatch. Alert escalation operations raise essential alerts to supervisory personnel after a preset period of time, ensuring that essential situations are handled between operator period changes. Response acquisition user interfaces in alert displays help to close the response acquisition loop that supports continuous learning, allowing for the capture of diagnostic information, corrective action, and verification data, and establishing institutional breakdown and intervention repositories.

10.48047/jocaaa.2026.35.01.62

V. Evaluation and Performance Analysis

5.1 Performance Metrics: Latency, Throughput, Accuracy

Recognition performance is measured in a matrix of performance metrics with three main characteristics of full delay, message processing, and recognition performance. Delay is defined as the collection time from the source sensor, MQTT broker transmission, processing in the cloud reception gateway, stream modifying component, calculation of characteristics, and final determination of the irregularity. These time gaps reveal congestion locations on the manipulation conduit; this indicates whether the delay occurs at the network transmission boundaries, in protocol conversion stages, or within computational recognition. Capacity measures observed manipulation transitions across various sensor densities reveal the reception capabilities of scan stream manipulators and recognition calculations, as event rates increase. These speeds are typical of fabrication intensifications or expanded sensor implementations [9]. Sustainability of capacity under maximum manufacturing conditions relative to reference operations determines if dynamic-scaling techniques dynamically optimize computational resources to suit workload fluctuations. Recognition correctness statistics compare algorithm decisions to operator-reviewed ground truth designations gained during response procedures. Exactness frequencies specify what fraction of the alarms for anomalies on designated devices are indeed device failures versus false positives, causing unnecessary operator investigations. Coverage measurements provide the perception range and the rate of genuine apparatus breakdowns perceived by the infrastructure up to the fabrication halt. Balanced rating estimations that equilibrate exactness and coverage exchanges exhibit thorough recognition potency throughout varied operational situations and breakdown expressions. This analysis delineates precision per apparatus category, untangling breakdown mechanisms and fabrication conditions, while identifying contexts with strong recognition and opportunities for representation improvement or algorithm calibration.

5.2 Fault Tolerance and Reliability Assessment

Dependability assessment is based on the behavior of infrastructure components under certain conditions, such as with MQTT brokers failing, loss of capacity to receive from the cloud, or failure of stream manipulators due to resource exhaustion. These failures are then monitored to determine whether redundancy mechanisms can continue to keep the system online without losing messages or interfering with message manipulation. Reinstatement time measures how quickly the infrastructure returns to service after spontaneous failures of its components. It measures whether restoration of the state based on a checkpointing approach can bring the manipulated state back without restoring the conduits entirely. Message delivery guarantees measure the ability of QoS settings on conduits to provide guarantees against message losses due to network congestion or broker transfer. These tests are concerned with how well the system handles duplicate messages. They test that deduplication processes work as expected, when patterns of retrying messages and reconnecting to the broker result in duplicates [9]. Partition endurance tests partition the plant-side MQTT brokers from the cloud reception terminals, and seek to determine if enough regional holding capacity exists to absorb temporary connection drops without queues filling or sensor messages dropping. Reliability evaluation calculates the mean-time-to-failure on dispersive elements, locates dispersive elements where failures are most likely, and determines if the allocated redundancy absorbed the failure rates. Homogeneity proofs ensure the outcomes of dispersive stream manipulations are both consistent in the event of module failure or network latency, and invariant with respect to reshuffling the manipulation topology by breakdown recovery mechanisms.

5.3 Comparative Analysis with Legacy Systems

Execution comparisons measure latency (cloud-enabled instantaneous manipulation), contrast with customary batch-based fabrication observation infrastructures in automotive fabrication plants, and contrast the quantum with which instantaneous manipulation in cloud infrastructures decreases the time between

10.48047/jocaaa.2026.35.01.62

irregularity manifestation and operator proclamation (in contrast to batch infrastructures enabling periodic isolated surveillance). Capacity comparisons assess the adjustability advantages offered by cloud infrastructures. Extensibility examines the cloud dynamic-adjustment services' capabilities to scale sensor clusters, versus infrastructure upfront investments required to increase capacity in batch infrastructure. In the economic analysis, infrastructure operational costs are considered. This includes: absorption of computational assets, infrastructure persistence, bandwidth consumption, and human endeavor for infrastructure administration and maintenance [10]. The multi-year operational time horizon examines costs such as upfront implementation, continuing operational costs, and costs subsequent to extensions and feature additions. Contrast recognition accuracy examines their capability to identify subtleties that are not detected when machine learning is applied upfront, but rather through batch observation and the eventual added capability. Early failure detection and false alarm rates are indicators of diagnostic performance, while adaptability is the ability of each of the designs to quickly adapt to fabrication changes like new sensors, a different sequence, or a different product configuration. It assesses set-up effort and infrastructure interruption through comparisons of enterprise and organizational situations and problems, such as the skill levels required to operate infrastructure, dependence on suppliers, and integration with existing infrastructure in fabrication and enterprise detail systems.

Performance Dimension	Cloud-Native Architecture	Legacy Batch System	Improvement Factor
Anomaly Detection Latency	Sub-second	Minutes to Hours	Significant reduction
Sensor Throughput Capacity	Auto-scaling to demand	Fixed capacity	Dynamic expansion
Infrastructure Provisioning	Minutes (automated)	Weeks (manual)	Rapid deployment
Operational Overhead	Minimal (managed services)	High (manual administration)	Labor reduction
Scalability Pattern	Elastic horizontal	Vertical limitations	Enhanced flexibility
Detection Algorithm Complexity	Ensemble ML models	Static thresholds	Advanced capabilities

Table 4: Performance Comparison - Cloud-Native vs Legacy Systems [2, 10]

5.4 Benefits and Challenges Discussion

Overall, the deployed design is considerably improved, having greatly reduced irregularity recognition delay, permitting preventive intervention before appliance breakdowns escalate into fabrication interruptions or quality defects. Additionally, the cloud-based design is elastic in its adjustability to accommodate growing sensor clusters and capacity variation without the manual capability scheduling and machine procurement lead times typical of facility-based designs. Machine learning-based recognition provides observations of degradation that cannot be determined using fixed thresholds. It can also, through ensembles of complementary recognition techniques, improve early detection of degradation, reduce false

10.48047/jocaaa.2026.35.01.62

positives, and adapt observed representations in response to alerts confirmed by operators. This eliminates the need for data science experts and enables the autonomous adaptation of representations to changes in the fabrication environment or the fabrication operation itself [10]. The extra administration of managed cloud resources means less work is needed from the manufacturing IT teams that maintain the plant floor infrastructure for capacity provisioning, failure recovery, security patching, and performance tuning. The challenges of sending sensor telemetry backhaul from the plant floor to the cloud may exceed the capacity of some existing industrial network infrastructure. Issues with the quality of sensor data (such as miscalibration, intermittent connection, and environmental interference) require wide-ranging preprocessing and validation of sensor data to minimize the impact of poor data quality on recognition processes. Representation drift is a persistent problem and requires monitoring of representation and periodic retraining to account for potentially slow changes in fabrication operations, fabrication conditions, and equipment maturity. Interfacing and designing integrations with existing fabrication implementation infrastructure, historian repositories, and existing observation infrastructure, while maintaining consistency in heterogeneous information infrastructures, is challenging. Organizationally, challenges also include training workers about operating cloud infrastructures, developing governance for cloud-based assets, and building trust in automated infrastructures among workers trained on and accustomed to customary observation infrastructures.

5.5 Scalability Considerations

Fabrication adaptability assessment estimates architectural improvements to fabrication, for instance, additional fabrication procedures, a higher concentration of gateways, and lengthened telemetry retention periods, without adversely affecting the design or its operation. Laterally adaptable assessment estimates the improvement in reception time offered by appending gateway occurrences, confirming that distribution procedures enable sensor sequences to disperse over the enlarged architecture rather than concentrating on a single point. Stream manipulation adjustability experiments test whether adding more workers and partitioning information processing across manipulators preserves balanced resource allocation. Persistence adjustability experiments test whether stratified persistence policies enable maturity-based data migration from costly high-speed persistence layers to economical but slower archival layers, without incurring a slowing of query execution times for near real-time readings [10]. The analysis compares the inquiry response time as more historical information is aggregated, proving that the used indexing techniques and segmentation perform within tolerable response times for analyses over long time intervals and that the representation of the recognition is adjustable to stabilize inferential lag time when the sensor clusters are widened. Concurrent forecast serves to ascertain that longer characteristic array frequencies do not increase manipulation duration per reading. Expenditure adjustability tests costs scaling with sensor cluster increase, determining whether cloud pricing representations are both feasible and maintainable at scale, or whether costs scale superlinearly, requiring architectural changes. Geographic adjustability is considered for multi-installation systems that extend their cloud architecture over physical fabrication locations or need to accommodate territorial information residency requirements with federated solutions under local control. The questions of organizational adjustability concern the appropriateness of operational and alert reaction methods, and response procedures, to the extended observational width without exhausting fabrication and before alert fatigue (reduced recognition) sets in.

Conclusion

This architecture is shown to enable real-time monitoring and predictive maintenance in automotive powertrain production environments using MQTT-based sensor networks, distributed stream processing, and machine learning-based anomaly detection. The drawbacks of customary batch-based monitoring

10.48047/jocaaa.2026.35.01.62

systems are overcome, leading to a sub-second delay from the time a sensor observation is made to the time the operator is informed, allowing action to be taken before a failure in production machinery leads to equipment downtime. Ensemble-based detection algorithms combining statistical methods with deep learning architectures allow gradually increasing sensitivity to subtle patterns of degradation while reducing the number of false positives using closed-loop feedback. The cloud-native infrastructure can elastically scale the set and the workload of sensors without manual capacity provisioning. This minimizes operational overhead compared to facility-based deployments. Future capabilities will include federated learning, which will enable distributed training of models while maintaining data ownership at individual manufacturing facilities, digital twin, and reinforcement learning-based control capabilities to transform from a reactive to an autonomous and optimized manufacturing operation with predictive maintenance, autonomous quality assurance, and production optimization capabilities for large-scale automotive manufacturing companies.

References

[1] Kamran Sattar Awaisi, et al., "A Survey of Industrial AIoT: Opportunities, Challenges, and Directions," IEEE Internet of Things Journal, 10 July 2024. [Online]. Available: https://ieeexplore.ieee.org/document/10591982

[2] Hung Cao, et al., "Lessons Learned from Integrating Batch and Stream Processing using IoT Data," in 2019 Sixth International Conference on Internet of Things: Systems, Management and Security (IOTSMS), 23 December 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8939232

[3] Basma M. Mohammad El-Basioni, "A conceptual modeling approach of MQTT for IoT-based systems," Journal of Electrical Systems and Information Technology, 18 December 2024. [Online]. Available: https://link.springer.com/article/10.1186/s43067-024-00181-x

[4] Omoniyi David Olufemi, "AI-enhanced predictive maintenance systems for critical infrastructure: Cloud-native architectures approach," World Journal of Advanced Engineering Technology and Sciences, November 2024. [Online]. Available: https://www.researchgate.net/publication/386277180

[5] Valentin Stangaciu, Cristina Stangaciu, Bianca Gusita, and Daniel-Ioan Curiac, "Integrating RealTime Wireless Sensor Networks into IoT Using MQTT-SN," Journal of Network and Systems Management, 25 February 2025. [Online]. Available: https://link.springer.com/article/10.1007/s10922025-09916-1

[6] Katerina Mitropoulou, et al., "Anomaly Detection in Cloud Computing using Knowledge Graph Embedding and Machine Learning Mechanisms," Journal of Grid Computing, 29 December 2023. [Online]. Available: https://link.springer.com/article/10.1007/s10723-023-09727-1

[7] Jon Martin Fordal, et al., "Application of sensor data-based predictive maintenance and artificial neural networks to enable Industry 4.0," Advances in Manufacturing, vol. 11, pp. 248–263, 4 March 2023. [Online]. Available: https://link.springer.com/article/10.1007/s40436-022-00433-x

[8] Wen Wang, et al., "Anomaly detection algorithms for vehicle-to-network interaction operator-level interactive control support systems based on machine learning," International Journal of Low-Carbon Technologies, vol. 20, 24 January 2025. [Online]. Available: https://academic.oup.com/ijlct/article/doi/10.1093/ijlct/ctae266/7978731

[9] Kanta Devanagavi, "Performance Evaluation of IoT Communication Protocols in Smart Grid Applications," International Journal of Advanced Research in Computer Science, May 2025. [Online]. Available: https://www.aijfr.com/papers/2025/5/1381.pdf

10.48047/jocaaa.2026.35.01.62

[10] Dilipkumar Devarahosahalli Jayaram, "Bridging Legacy Systems with Modern Platforms: A Scalable Approach," International Journal of Research in Computer Applications and Information Technology, vol. 8, no. 1, Jan–Feb 2025. [Online]. Available: https://www.researchgate.net/publication/389295801_Bridging_Legacy_Systems_with_Modern_Platforms_A_Scalable_Approach