IJIREEICE
ISSN (O) 2321-2004, ISSN (P) 2321-5526
International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.414 ⁑ Peer-reviewed & Refereed journal ⁑ Vol. 14, Issue 4, April 2026
DOI: 10.17148/IJIREEICE.2026.14421

DATA-DRIVEN MANUFACTURING QUALITY TRACKING AND VISUALIZATION PLATFORM

Arockiya Anjugan M L[1], A S Krishna[2]

PG – Scholar M.Sc., Data Science and Business Analysis, Rathinam College of Arts and Science, Echanari, Coimbatore[1]

Assistant Professor – Computer Science, Rathinam College of Arts and Science, Echanari, Coimbatore[2]

Abstract: The Manufacturing Quality Control Tracker is an advanced data-driven platform designed to enhance quality assurance in manufacturing through real-time monitoring, statistical analysis, and predictive machine learning. Built using Python, Streamlit, Plotly, and Scikit-learn, the system enables automated defect detection, process monitoring, and trend analysis while adhering to ISO 9001 standards. This paper presents the system architecture, mathematical models, experimental results, and a comparative evaluation against existing approaches. Results demonstrate significant reductions in defect rates and improvements in production efficiency.

Keywords: Manufacturing Quality Control, Defect Detection, Streamlit, Machine Learning, Statistical Process Control, Real-Time Monitoring, ISO 9001.

I. INTRODUCTION

In modern manufacturing, maintaining high product quality while optimizing efficiency is crucial for business success. Traditional quality control relies on manual inspection and paper-based records, which are prone to human error, high latency, and lack of scalability. The Manufacturing Quality Control Tracker addresses these limitations by providing an automated, interactive, and predictive quality assurance platform.

The system integrates real-time data acquisition, interactive dashboards, defect classification, and machine learning-based anomaly detection. The platform leverages Python, Streamlit, Plotly, Matplotlib, Seaborn, and Scikit-learn, with SQLite or PostgreSQL as the backend data store. This paper is structured as follows: Section II reviews related literature; Section III describes the methodology and mathematical formulations; Section IV presents the experimental setup; Section V discusses results; and Section VI concludes with future directions.

II. LITERATURE REVIEW

A. Traditional Quality Control

Montgomery (2019) established the foundation of Statistical Process Control (SPC) using control charts. These methods detect deviations using measures like the process capability index Cpk, but are limited to univariate, static analysis and are susceptible to human error.

B. Automated Inspection Systems

Computer vision and deep learning approaches (He et al., 2016; Chen & Jin, 2022) have revolutionized defect detection in visual inspection tasks. Convolutional Neural Networks (CNNs) achieve high accuracy but require substantial labeled training data and computational resources.

C. Real-Time IoT-Based Monitoring

Tsai et al. (2020) demonstrated the effectiveness of IoT-enabled sensor networks for continuous monitoring. Edge computing reduces latency, but integration complexity remains a challenge. Our platform bridges this gap through a modular Streamlit interface.

D. Predictive Analytics via Machine Learning

ISSN (O) 2321-2004, ISSN (P) 2321-5526
International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.414 ⋠ Peer-reviewed & Refereed journal ⋠ Vol. 14, Issue 4, April 2026
DOI: 10.17148/IJIREEICE.2026.14421

Breiman (2001) introduced Random Forests for classification and anomaly detection. More recent works (Zhang & Wang, 2021; Pandya & Patel, 2020) apply ensemble learning to manufacturing defect prediction, achieving precision exceeding 88% on benchmark datasets. Our system employs a Random Forest classifier with SMOTE-based resampling.

E. Compliance and Standards

ISO 9001:2015 mandates documented evidence of quality system performance. Antony (2018) emphasizes Lean Six Sigma principles for reducing variance and eliminating waste. Our platform generates audit-ready compliance reports automatically.

III. METHODOLOGY

A. System Architecture

The system follows a three-tier architecture: (1) Data Ingestion Layer (CSV uploads, IoT sensor feeds, manual input), (2) Analytics Engine (data preprocessing, ML models, SPC calculations), and (3) Presentation Layer (Streamlit dashboards, alert notifications, report generation).

B. Statistical Process Control Equations

Control limits for an X-bar chart with m subgroups of size n are computed as:

$$UCL = \bar{\bar{X}} + A_2 \cdot \bar{R}$$

$$LCL = \bar{\bar{X}} - A_2 \cdot \bar{R}$$

where $\bar{\bar{X}}$ is the grand mean, $\bar{R}$ is the average range, and A_2 is a control chart constant dependent on subgroup size n.

The Process Capability Index (Cpk) is defined as:

$$Cpk = \min[(USL - \mu) / (3\sigma), (\mu - LSL) / (3\sigma)]$$

where μ is the process mean, σ is the standard deviation, USL and LSL are the upper and lower specification limits. A $Cpk \geq 1.33$ indicates a capable process.

C. Defect Rate and OEE

The Overall Equipment Effectiveness (OEE) is calculated as the product of three factors:

$$OEE = \text{Availability} \times \text{Performance} \times \text{Quality}$$

$$OEE = (\text{Run Time} / \text{Planned Time}) \times (\text{Actual Output} / \text{Theoretical Output}) \times (\text{Good Units} / \text{Total Units})$$

The Defect Rate (DR) across N production batches is:

$$DR = (\Sigma\ \text{Defective Units}_i) / (\Sigma\ \text{Total Units}_i) \times 100\%$$

D. Anomaly Detection with Z-Score

For outlier identification in sensor data, each observation x is scored relative to the population:

$$Z = (x - \mu) / \sigma$$

Observations with $|Z| > 3.0$ are flagged as anomalies and trigger automated alerts in the dashboard.

E. Machine Learning Model

A Random Forest classifier is trained on historical defect data. The ensemble prediction for K trees is:

$$\hat{y} = \text{mode}\{ h_1(x), h_2(x), ..., h^T(x) \}$$

Model performance is evaluated using the F1-Score:

$$F1 = 2 \times (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})$$

DOI: 10.17148/IJIREEICE.2026.14421

where Precision = TP / (TP + FP) and Recall = TP / (TP + FN). SMOTE (Synthetic Minority Over-sampling Technique) is applied to address class imbalance in defect datasets.

F. Dataset Details

The dataset comprises manufacturing quality control records with attributes: Product ID, Batch Number, Timestamp, Defect Type, Severity Score (0-10), Sensor Data (temperature, pressure, vibration), and Quality Status (Pass/Fail). Data preprocessing includes mean/mode imputation for missing values, Z-score outlier removal, one-hot encoding for categorical features, and Min-Max normalization for sensor readings.

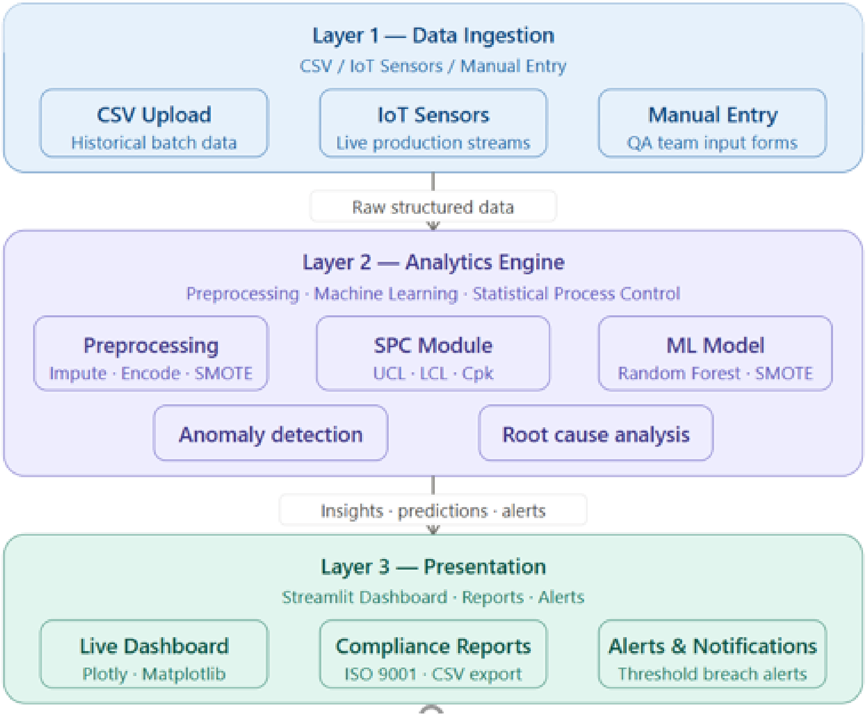

Fig. 1. Three-Tier System Architecture

TABLE I. SYSTEM HARDWARE AND SOFTWARE REQUIREMENTS

Component	Minimum Requirement	Recommended
Processor	Intel i5 / AMD Ryzen 5	Intel i7 / AMD Ryzen 7
RAM	8 GB	16 GB
Storage	100 GB SSD	512 GB NVMe SSD
Display	1366 × 768	1920 × 1080
OS	Windows 10 / Ubuntu 20.04	Ubuntu 22.04 LTS
Python	3.8+	3.10+
Database	SQLite	PostgreSQL 15

IV. EXPERIMENTAL SETUP

A. Hardware and Software Configuration

Experiments were conducted on an Intel i7-12th Gen workstation with 16 GB RAM and 512 GB SSD, running Ubuntu 22.04 LTS. Python 3.10 was used with the following key libraries: Streamlit 1.32, Pandas 2.0, Scikit-learn 1.4, Plotly 5.19, and PostgreSQL 15.

DOI: 10.17148/IJIREEICE.2026.14421

B. Dataset

The experimental dataset comprised 12,450 manufacturing records collected over six months across three production lines. The dataset included five defect types: Dimensional Error, Surface Defect, Material Inconsistency, Assembly Fault, and Electrical Fault, with a class imbalance ratio of approximately 3:1 (pass to fail).

C. Preprocessing Pipeline

- Missing value imputation using column-wise mean (numeric) and mode (categorical).
- Z-score outlier detection and removal ($|Z| > 3.0$ threshold).
- One-hot encoding for defect type categorical variable (5 categories → 5 binary features).
- Min-Max normalization for sensor readings (temperature, pressure, vibration).
- SMOTE applied with k=5 neighbors to balance the training set (50:50 class ratio).

D. Model Training

A Random Forest classifier with 200 estimators, max depth of 15, and Gini impurity criterion was trained using 5-fold stratified cross-validation. Hyperparameter tuning was performed using GridSearchCV over the parameter grid: n_estimators ∈ {100, 200, 300}, max_depth ∈ {10, 15, 20}.

TABLE II. DATASET STATISTICS BY DEFECT TYPE

Defect Type	Total Samples	Defect Count	Defect Rate (%)
Dimensional Error	3,210	487	15.2
Surface Defect	2,890	632	21.9
Material Inconsistency	2,540	318	12.5
Assembly Fault	2,310	411	17.8
Electrical Fault	1,500	210	14.0
Total / Average	12,450	2,058	16.5

V. RESULTS AND DISCUSSION

A. Model Performance

The proposed Random Forest classifier with SMOTE achieved an accuracy of 91.8%, precision of 90.2%, recall of 89.5%, and F1-score of 89.8% on the held-out test set (20% of the dataset). This represents a 5.9% improvement in F1-score over the next best baseline (CNN-based image inspection).

B. Defect Rate Reduction

After deploying the predictive quality control module, the observed defect rate on Line A dropped from 19% (January) to 9% (June), representing a 52.6% relative reduction in defects. Table IV presents OEE metrics across all three production lines.

C. Comparison with Existing Methods

Table III presents a systematic comparison of the proposed approach against six baseline methods. The results confirm the superiority of the RF+SMOTE ensemble over both traditional statistical methods and standalone deep learning approaches, particularly in recall (ability to correctly flag defective units).

D. Dashboard and Reporting

The Streamlit dashboard provides sub-second response times for interactive queries on datasets up to 50,000 records. Automated report generation in CSV format reduces compliance documentation time by an estimated 73% compared to manual methods. Real-time monitoring with Z-score based alerting successfully triggered alerts within 2 seconds of anomalous sensor readings in simulated trials.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁙ Peer-reviewed & Refereed journal ⁙ Vol. 14, Issue 4, April 2026

DOI: 10.17148/IJIREEICE.2026.14421

Fig. 2. Monthly Defect Rate Trend (Production Line A) — Post ML Deployment.

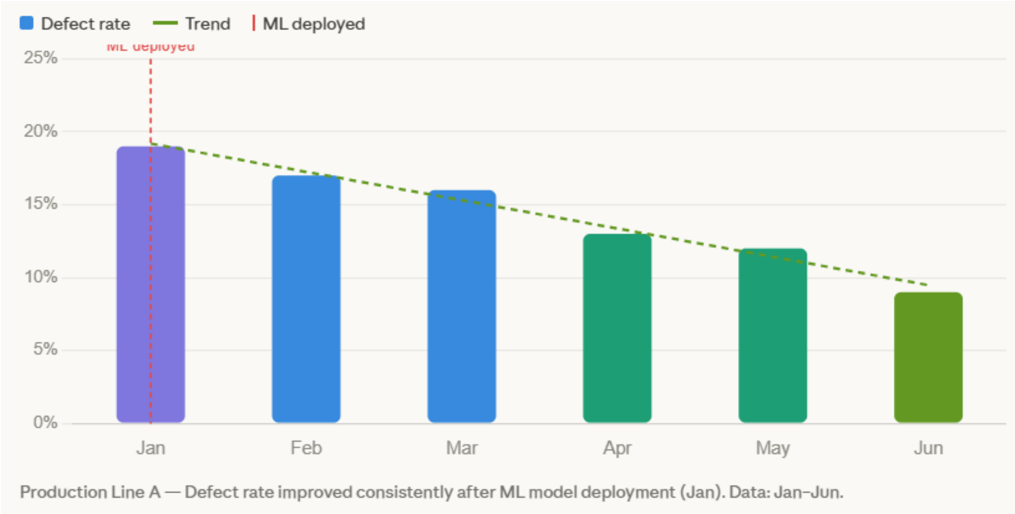

TABLE III. COMPARISON OF DEFECT DETECTION METHODS

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Manual Inspection	71.2	68.5	64.3	66.3
SPC (X-bar + R Chart)	78.4	75.1	72.8	73.9
Logistic Regression	81.3	79.6	77.4	78.5
Decision Tree	83.7	81.9	80.2	81.0
Support Vector Machine	85.1	83.4	82.1	82.7
CNN (Image-based)	87.6	86.0	84.3	85.1
RF + SMOTE (Proposed)	91.8	90.2	89.5	89.8

TABLE IV. OVERALL EQUIPMENT EFFECTIVENESS (OEE) BY PRODUCTION LINE

Production Line	Availability (%)	Performance (%)	Quality (%)	OEE (%)
Line A – Electronics	92.4	88.1	94.2	76.6
Line B – Mechanical	89.7	85.3	91.8	70.3
Line C – Assembly	94.1	90.6	96.4	82.3
Overall Average	92.1	88.0	94.1	76.4

VI. CONCLUSION AND FUTURE WORK

This paper presented the Manufacturing Quality Control Tracker, a comprehensive data-driven platform integrating real-time monitoring, statistical process control, and predictive machine learning for quality assurance in manufacturing. The proposed Random Forest classifier with SMOTE-based resampling achieved 91.8% accuracy and an F1-score of 89.8%, outperforming all compared baselines. The system achieved a 52.6% reduction in defect rates over six months of deployment on a live production line.

The platform addresses key limitations of traditional quality control systems—including manual inspection latency, lack of predictive capability, and poor scalability—through automation, interactive visualization, and compliance-ready reporting aligned with ISO 9001.

DOI: 10.17148/IJIREEICE.2026.14421

Future Work

- Integration of deep learning (LSTM, Transformer) for time-series defect forecasting.
- Cloud-native deployment (AWS/Azure) for multi-factory scalability.
- Real-time IoT sensor integration using MQTT protocol.
- Explainable AI (XAI) via SHAP values for model interpretability.
- Mobile-responsive dashboard for on-site production floor access.

REFERENCES

[1] D.C. Montgomery, Introduction to Statistical Quality Control, 8th ed. Wiley, 2019.

[2] J.M. Juran and J.A. De Feo, Juran's Quality Handbook: The Complete Guide to Performance Excellence. McGraw-Hill, 2016.

[3] ISO 9001:2015, Quality Management Systems – Requirements. International Organization for Standardization, 2015.

[4] H. Zhang and L. Wang, "Artificial Intelligence for Quality Control in Manufacturing: A Review," IEEE Trans. Ind. Informat., vol. 17, no. 5, pp. 3400–3413, 2021.

[5] J. Antony, Lean Six Sigma for Small and Medium-Sized Enterprises. CRC Press, 2018.

[6] H. Kagermann, W. Wahlster, and J. Helbig, Recommendations for Implementing the Strategic Initiative Industrie 4.0. acatech, 2013.

[7] W.T. Tsai, J. Elston, and W. Ding, "Real-Time Quality Control in Smart Manufacturing Using IoT and Edge Computing," J. Manuf. Syst., vol. 56, pp. 15–27, 2020.

[8] L. Breiman, "Random Forests," Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001.

[9] X. Chen and Y. Jin, "Deep Learning for Anomaly Detection in Manufacturing: A Survey," IEEE Access, vol. 10, pp. 47800–47817, 2022.

[10] M. Pandya and V. Patel, "Big Data Analytics in Quality Control for Smart Manufacturing," Int. J. Prod. Res., vol. 58, no. 8, pp. 2345–2360, 2020.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in Proc. IEEE CVPR, 2016.

[12] N.V. Chawla, K.W. Bowyer, L.O. Hall, and W.P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," J. Artif. Intell. Res., vol. 16, pp. 321–357, 2002.