

AMMOSHIELD: AI-Based Smart Ammonia Gas Leak Detection and Industrial Safety Monitoring System

Vignesh G.S

Department of Electronics &
Communication Engineering
K. Ramakrishnan College of
Technology
India

E-mail ID: vignesh.ec23@krct.ac.in

Sanjai M

Department of Electronics &
Communication Engineering
K. Ramakrishnan College of
Technology
India

E-mail ID: sanjai.ec23@krct.ac.in

Venkatesh T

Department of Electronics &
Communication Engineering
K. Ramakrishnan College of
Technology
India

E-mail ID:
venkatesh.ec23@krct.ac.in**Ahmed A**

Department of Electronics &
Communication Engineering
K. Ramakrishnan College of
Technology
India

E-mail ID: ahmed.ec23@krct.ac.in

Abstract

Ammonia (NH₃) is one of the most widely used industrial chemicals, serving as a core input for fertilizer production, a refrigerant in large-scale cold-storage and food-processing plants, and a reagent in pharmaceutical and chemical manufacturing. Despite its industrial importance, ammonia is a highly toxic and corrosive gas, and accidental leakage in enclosed or semi-enclosed industrial environments poses serious risks of respiratory injury, chemical burns, and, at high concentrations, fatality among assessment. Sensor readings, together with derived features such as the rate of gas concentration change, are processed locally and classified into graded risk levels ranging from Normal to Emergency. The system couples this classification with IoT/cloud-based real-time monitoring, local audible and visual alarm generation.

Keywords — Ammonia Gas Detection, Artificial Intelligence, Machine Learning, ESP32, Internet of Things, Industrial Safety, Gas Sensor, Real-Time Monitoring

I. Introduction

Ammonia (NH₃) is among the most extensively produced and consumed industrial chemicals in the world, forming a cornerstone input for the fertilizer industry, where it is used to manufacture urea and ammonium-based compounds. Beyond agriculture, ammonia plays a critical role as a refrigerant in large industrial refrigeration and cold-storage plants because of its high latent heat of vaporization and low cost relative to synthetic refrigerants. It is also used extensively in chemical manufacturing, food processing and cold-chain logistics, and as a reagent or pH-adjusting agent in pharmaceutical production. This widespread industrial dependence on ammonia means that thousands of workers across fertilizer plants, cold-storage warehouses, dairy and food-processing units, and chemical facilities are routinely present in environments where ammonia is stored, transported, or used under pressure.

Despite its usefulness, ammonia is a hazardous substance. It is corrosive to skin, eyes and the respiratory tract, and

exhaust-ventilation control, and remote notification to plant operators. By combining continuous multi-parameter sensing with data-driven risk classification rather than static thresholds alone, AMMOSHIELD is designed to support earlier detection of developing ammonia leaks and workers. Conventional gas detection systems used in many industrial plants rely on fixed threshold alarms, manual inspection rounds, or single-point electrochemical sensors that lack contextual awareness, generate frequent false alarms, and provide no predictive insight into a developing leak. This paper presents AMMOSHIELD, an AI-based smart ammonia gas leak detection and industrial safety monitoring system that integrates ammonia gas sensing, ESP32-based embedded processing, and environmental parameter monitoring (temperature and humidity) with a machine-learning classification layer for intelligent risk.

Narrow margin between routine industrial handling and dangerous exposure: workplace time-weighted average limits are commonly specified in the tens of parts per million, while concentrations only a few multiples higher are classified as immediately dangerous to life and health. Because ammonia is often stored and transported under pressure in bulk quantities, even a small mechanical failure, valve leak, pipe-joint corrosion, or seal degradation can rapidly release a hazardous concentration of gas into a confined industrial space.

Detecting such leaks in a timely manner is difficult using manual inspection alone. Human operators cannot continuously monitor every section of a large industrial facility, and by the time a leak becomes detectable by odor or visible symptoms among workers, the exposure may already be significant. Conventional fixed gas detectors, most of which rely on electrochemical or metal-oxide semiconductor sensing elements, are typically configured to trigger a binary alarm once a preset concentration threshold is crossed. While useful, such threshold-based systems provide little advance warning of a developing leak, cannot easily distinguish between a genuine hazardous trend and transient sensor noise or environmental interference, and generally operate as

isolated units without centralized visibility, historical logging, or remote notification. These limitations reduce both the earliness and the reliability of the safety response in many existing installations.

The proliferation of low-cost microcontroller platforms with integrated wireless connectivity, together with advances in the Internet of Things (IoT), has created an opportunity to modernize industrial gas-safety infrastructure. IoT-enabled monitoring allows sensor data to be continuously collected, transmitted to a cloud platform, visualized on dashboards, and used to trigger remote alerts to safety personnel regardless of their physical location. At the same time, advances in machine learning (ML) make it possible to move beyond static threshold alarms toward data-driven risk classification, in which multiple sensor features, including the rate of change of gas concentration and ambient environmental conditions, are jointly analyzed to estimate the severity of a developing situation. Combining continuous IoT connectivity with an embedded AI/ML classification layer allows a monitoring system to react not only to the instantaneous gas level but also to the underlying trend, potentially enabling an earlier and more graduated safety response than threshold-only designs.

Motivated by these considerations, this paper proposes AMMOSHIELD, an AI-based smart ammonia gas leak detection and industrial safety monitoring system intended for deployment in fertilizer plants, cold-storage facilities, and other industrial environments where ammonia is stored or used. AMMOSHIELD combines a dedicated ammonia gas sensor, an ESP32 microcontroller for local data acquisition and control, environmental sensing of temperature and humidity, a machine-learning-based risk classification module, and an IoT/cloud layer for remote visualization, logging and notification. The system is further coupled with an automated safety-response subsystem capable of activating local alarms, warning indicators and exhaust ventilation in proportion to the assessed risk level.

The main contributions of this work are summarized as follows:

- 1) A real-time ammonia gas monitoring architecture built around a low-cost, Wi-Fi-enabled embedded controller.
- 2) An AI/ML-based approach to leak and risk classification that uses multiple sensor-derived features rather than a single fixed threshold.
- 3) Integration of environmental-condition monitoring (temperature and humidity) to contextualize gas-sensor behavior.
- 4) IoT/cloud connectivity for real-time remote monitoring, historical data logging and dashboard visualization.
- 5) An automated, graded safety-response mechanism spanning local alarms, visual indicators, and exhaust-fan activation.
- 6) A dual local-and-remote alerting strategy that keeps on-site personnel and remote safety officers simultaneously informed.
- 7) A discussion of fail-safe

design considerations so that critical local safety functions do not depend solely on network connectivity.

- 8) A scalable architecture intended to generalize from a single-point prototype to a distributed, multi-node industrial deployment.

The remainder of this paper is organized as follows. Section II reviews existing ammonia and industrial gas-detection approaches and identifies the research gap addressed by this work. Section III describes the proposed AMMOSHIELD architecture and its constituent functional blocks. Section IV presents the hardware implementation. Section V details the AI/ML model development pipeline, while Section VI presents the software control algorithm. Section VII discusses the automated safety-response strategy, and Section VIII describes the IoT-based monitoring layer. Section IX outlines the experimental setup, and Sections X and XI present the performance-analysis methodology and illustrative results, respectively. Section XII lists the supporting figures. Section XIII compares AMMOSHIELD with conventional and IoT-based systems, Section XIV discusses reliability and fail-safe design, and Sections XV–XVII present the advantages, limitations and future work. Section XVIII concludes the paper.

II. Existing System and Related Work

A substantial body of prior work addresses gas-leak detection using electrochemical and metal-oxide semiconductor (MOS) sensors combined with microcontroller-based alarm circuits. Lakshmi et al. implemented an ammonia leakage detection and monitoring system using an MQ135-class gas sensor together with IoT connectivity to report gas levels remotely [1]. Santiputri presented an IoT-based gas-leak detection device intended for general industrial and domestic use, focusing on low-cost sensing and Wi-Fi-based notification [3]. Jadhav et al. proposed an IoT-enabled industrial gas-leakage detection scheme built around MQ-series sensors and a microcontroller, emphasizing coverage across industrial zones [4]. Sharma et al. surveyed the broader landscape of IoT- and cloud-based gas-leakage detection systems, noting that most deployed systems remain threshold-driven and offer limited analytic capability beyond simple alarm generation [5].

Other researchers have explored sensor-network and cloud-integration aspects of industrial gas safety. Senthil et al. described an enhanced IoT sensor-network-based alerting system for industrial gas leakage aimed at improving responsiveness in plant environments [6]. Manjunathan et al. presented an IoT-based gas-leakage detection system combining a microcontroller, gas sensor and GSM/Wi-Fi notification path for both domestic and light-industrial use [7]. Al-Jemeli and Al-Adilee developed a portable gas-leak detection system using off-the-shelf sensor nodes, illustrating the feasibility of low-cost distributed monitoring [8]. At a

larger scale, Chraim et al. investigated wireless sensor-network approaches for gas-leak detection and source localization in industrial settings, highlighting the value of spatially distributed sensing for pinpointing leak origin [9], while Bin et al. explored vision-based leak detection from thermal surveillance imagery as a complementary, non-contact sensing modality [10].

A separate line of work applies machine learning to gas-sensor data, primarily in the context of electronic-nose (e-nose) systems used for gas identification and classification rather than dedicated safety alarms. Rehman and Bermak proposed a heuristic random-forest approach to compensate for long-term sensor drift in electronic-nose applications, demonstrating the robustness of ensemble tree methods to sensor variability [11]. Ma et al. applied deep-learning models to sensor-array response maps for gas recognition [12], and Yaqoob and Younis reviewed the broader potential and challenges of applying machine learning to chemical gas sensing, noting that ensemble methods such as random forests are frequently favored for their robustness to noisy, nonlinear sensor responses and their relatively low computational cost at inference time [13]. The random-forest algorithm itself, introduced by Breiman, remains one of the most widely used ensemble classifiers for structured, moderate-dimensional sensor data of the kind produced by low-cost gas sensors [14].

Table I compares representative existing approaches against the ammonia-specific, AI-integrated, and automated-response characteristics targeted by the proposed system.

TABLE I. COMPARISON OF EXISTING AMMONIA GAS DETECTION APPROACHES

Method	Sensor / Tech.	Real-Time	AI/ML	Remote Alert	Auto. Control	Limitation
Lakshmi et al. [1]	MQ135-class + IoT	Yes	No	Yes	No	Threshold-only alarm

ion → Safety Controller → Alarm/Exhaust/Shutdown → IoT Cloud → Mobile/Web Dashboard. Each stage is described below.

A. Ammonia Gas Sensing

The gas-sensing stage uses a semiconductor-type ammonia sensor whose electrical resistance changes in the presence of NH₃ molecules. The sensor's raw analog output is passed through a signal-conditioning stage (a load-resistor network and, where required, a simple low-pass filter) to reduce high-frequency noise before analog-to-digital conversion at the ESP32. Because the response of such sensors is nonlinear, exhibits a finite response and recovery time, and can drift gradually over its operating life, the sensor is periodically calibrated against a clean-air baseline reading taken at power-up and at scheduled intervals. Cross-sensitivity to other reducing gases and to ambient humidity is a known characteristic of semiconductor gas sensors and is addressed

Method	Sensor / Tech.	Real-Time	AI/ML	Remote Alert	Auto. Control	Limitation
Santiputri [3]	MOS sensor + Wi-Fi	Yes	No	Limited	No	No risk grading
Jadhav et al. [4]	MQ series + ctrl.	Yes	No	Yes	Limited	No trend analysis
Senthil et al. [6]	WSN + IoT	Yes	No	Yes	Limited	No AI classification
Al-Jemeli/AI-Adilee [8]	Off-shelf node	Yes	No	Limited	No	Portable, not industrial-scale
Chraim et al. [9]	WSN + localization	Yes	No	Yes	No	Focus on localization only
Rehman & Bermak [11]	E-nose + RF (offline)	No	Yes	No	No	Not a safety-alarm system

As Table I indicates, most deployed ammonia and industrial gas-detection systems provide real-time sensing and, increasingly, remote IoT alerting, but continue to rely on fixed-threshold logic rather than data-driven risk assessment, and rarely couple detection with automated ventilation or safety-shutdown control. Conversely, machine-learning approaches to gas-sensor data have so far been explored mainly in offline e-nose and gas-identification research rather than in integrated, real-time industrial safety systems. This identifies a clear research gap: the absence of a low-cost, field-deployable system that jointly combines ammonia-specific gas sensing, environmental context sensing, AI/ML-based multi-level risk classification, IoT/cloud connectivity, and automated safety control within a single architecture. AMMOSHIELD is proposed to address this gap.

III. Proposed AMMOSHIELD System

The AMMOSHIELD architecture follows the signal path: Ammonia Gas → Gas Sensor → Signal Conditioning → ESP32 → Data Preprocessing → AI/ML Model → Risk Classification

partly through environmental compensation (Section III-C) and partly through the trend-based features used by the AI/ML module rather than reliance on the absolute reading alone. Sensor placement follows standard industrial practice, positioning units near likely leak sources such as valves, flanges, and storage-tank connections, and accounting for the fact that ammonia is lighter than air and tends to accumulate near ceiling level in enclosed spaces. The exact quantitative concentration range and response characteristics of the deployed sensor should be taken from the selected component's datasheet rather than assumed generically.

B. ESP32 Controller

The ESP32 system-on-chip is selected as the central controller because it offers an integrated dual-core processor, multiple analog-to-digital converter (ADC) channels, general-purpose input/output (GPIO) pins, and built-in Wi-Fi connectivity in a single low-cost module. This combination

allows the same device to perform analog sensor interfacing, local decision logic, actuator control (buzzer, LED, relay), and cloud communication without requiring a separate communication co-processor. The ESP32's processing capability is sufficient to run a lightweight, pre-trained classifier (such as a compact random-forest model converted to fixed-point or a small decision-rule set) locally, enabling the safety controller to react even during a temporary loss of network connectivity. Its low unit cost and wide community support further make it well suited to the goal of a scalable, distributable industrial monitoring node.

C. Environmental Monitoring

Because the response of gas-sensing elements can be influenced by ambient temperature and humidity, AMMOSHIELD incorporates an environmental-sensing stage. A digital temperature-and-humidity sensor from the DHT11/DHT22 family is selected as the primary design choice for this prototype due to its simple single-wire digital interface and adequate accuracy for compensation purposes; a DS18B20 or equivalent sensor could alternatively be used where temperature-only monitoring at higher precision is required. It is important to note that only one such sensor, matched to the accuracy and response-time requirements of the deployment, is used in a given installation; the alternatives are presented as design options rather than components used simultaneously. The temperature and humidity readings are logged alongside the gas-sensor value and are supplied as additional input features to the AI/ML classifier described in Section III-D.

D. AI/ML Detection Module

Several supervised-learning algorithms are candidates for the risk-classification task, including logistic regression, decision trees, random forest, support vector machines (SVM), k-nearest neighbors (KNN), and artificial neural networks (ANN). This work adopts Random Forest as the primary classification algorithm. Random Forest is well suited to this application because gas-sensor response is inherently nonlinear with respect to concentration and environmental conditions; the model accepts multiple heterogeneous input features (gas value, rate of change, temperature, humidity, recent history) without requiring extensive feature engineering; the ensemble-of-trees structure is comparatively robust to sensor noise and to the class imbalance typically seen between normal and hazardous conditions; the algorithm supports feature-importance analysis, which is useful for understanding which inputs drive a given risk classification; and, relative to deep neural networks, a random forest of modest size has low computational and memory requirements, which is compatible with embedded or edge deployment on constrained hardware.

The candidate input features to the model include the instantaneous gas-sensor value, the short-term rate of change of gas concentration, ambient temperature, ambient humidity, a short window of previous sensor values, and a simple trend indicator derived from that window. The output of the classifier is a discrete risk category selected from the ordered set NORMAL $\rightarrow$ WARNING $\rightarrow$ CRITICAL $\rightarrow$ EMERGENCY. The specific numerical thresholds that separate these categories are not fixed a priori in this design; rather, they must be established through a combination of sensor calibration, controlled experimental exposure trials, and reference to applicable occupational exposure limits and industrial safety standards before field deployment.

IV. System Architecture and Hardware Implementation

The physical prototype integrates the following elements: an ESP32 development board as the main controller; an ammonia gas sensor for primary detection; a temperature/humidity sensor for environmental context; a small OLED or LCD module for local status display; a buzzer for audible alerts; a warning LED for visual indication; a relay module for actuator switching; a small exhaust fan representing the automated ventilation actuator; a Wi-Fi connection for cloud communication; a regulated power supply; and an optional emergency-shutdown push-button interface for manual override. The components communicate with the ESP32 over a combination of analog input, digital GPIO, and the I2C bus, as summarized in Table II.

TABLE II. HARDWARE COMPONENTS

Component	Function	Interface
ESP32	Main controller	GPIO / ADC / Wi-Fi
Ammonia sensor	Gas detection	Analog / Digital
DHT11 / DHT22	Temperature & humidity	Digital (1-wire)
OLED display	Local status display	I2C
Buzzer	Audible alert	GPIO
Warning LED	Visual warning	GPIO
Relay module	Actuator control	GPIO
Exhaust fan	Ventilation	Relay-switched
Cloud platform	Remote monitoring	Wi-Fi / HTTP-MQTT

V. AI/ML Model Development

A. Data Acquisition

Model development requires a labeled dataset of sensor readings collected under known, controlled conditions. An example dataset structure suitable for this task is shown below; the actual values, sample count, and label distribution must be generated through calibrated laboratory or controlled-release trials prior to deployment and are not asserted here as measured data.

Example dataset structure: Gas Value | Temperature | Humidity | Rate of Change | Risk Level. Each row represents one sampling instant, with the Risk Level column populated according to the ground-truth condition under which the

reading was collected (e.g., clean air, controlled low-level release, increasing concentration, rapid release).

B. Data Preprocessing

Prior to model training, the acquired data is preprocessed through several steps: missing-value handling (interpolation or discarding of incomplete samples caused by sensor read failures); noise filtering, typically via a moving-average or low-pass filter to smooth transient fluctuations; normalization or standardization of feature scales so that gas-value, temperature, and humidity inputs contribute comparably to distance- or gradient-based model components; outlier detection to identify and review sensor glitches or non-physical readings; feature scaling consistent with the chosen classifier; and data labeling, which assigns each preprocessed sample to one of the four risk categories based on the controlled test condition under which it was recorded.

C. Feature Extraction

A key derived feature is the short-term rate of change of the gas-sensor reading, which captures how quickly a concentration is rising rather than only its instantaneous value. This feature is computed as:

$$R = (C_t - C_{t-1}) / \Delta t$$

where C_t is the current sensor reading, C_{t-1} is the previous reading, and Δt is the sampling interval. A rapidly increasing R value, even at a moderate absolute gas concentration, is treated by the classifier as evidence of an escalating condition, which is precisely the scenario that a static-threshold alarm is least able to anticipate.

D. Random Forest Classification

A random forest classifier is an ensemble of decision trees, each trained on a bootstrap-sampled subset of the training data and, at each node split, restricted to a random subset of the available features. This combination of bootstrap aggregation (bagging) and random feature selection decorrelates the individual trees, reducing the variance of the overall ensemble relative to a single decision tree. For a given input feature vector x , each tree T_i produces a predicted class label $T_i(x)$; the ensemble prediction is obtained by majority voting across all T trees in the forest:

$$\hat{y} = \text{mode} \{ T_1(x), T_2(x), \dots, T_T(x) \}$$

The number of trees, maximum tree depth, and minimum samples per split are hyperparameters to be tuned using cross-validation on the calibrated dataset described in Section V-A, together with feature-importance analysis to confirm that the rate-of-change and gas-value features dominate the model's decisions, as expected from the underlying physics of a leak event.

VI. Software Flow and Control Algorithm

Algorithm 1 summarizes the continuous monitoring loop executed by the ESP32 firmware.

Algorithm 1: AMMOSHIELD Safety Monitoring

1. Start.
2. Initialize ESP32 peripherals.
3. Initialize ammonia gas sensor.
4. Initialize temperature/humidity sensor.
5. Initialize display and alarm devices.
6. Connect to Wi-Fi network.
7. Perform sensor baseline calibration.
8. Acquire sensor data (gas, temperature, humidity).
9. Apply noise filtering.
10. Extract features (rate of change, trend window).
11. Send feature vector to on-device AI/ML classifier.
12. Predict risk level (Normal / Warning / Critical / Emergency).
13. If NORMAL: continue monitoring, no action.
14. If WARNING: activate visual warning indicator.
15. If CRITICAL: activate buzzer and exhaust ventilation.
16. If EMERGENCY: activate maximum safety response and emergency notification.
17. Upload readings and risk level to cloud.
18. Store historical data locally and/or in the cloud.
19. Repeat continuously from step 8.

VII. Automated Safety Control

The safety-controller logic maps each predicted risk level to a graduated local and remote response, summarized in Table III. Under NORMAL conditions the system continues passive monitoring with no active intervention. Under WARNING, a yellow visual indicator is activated locally, the cloud dashboard displays a warning state, and a notification is sent to the responsible operator. Under CRITICAL, an audible buzzer and warning LED are activated locally, the exhaust fan is switched on to begin active ventilation, and a remote alert is issued. Under EMERGENCY, the system escalates to a high-priority audible alarm, drives ventilation to its maximum available rate, issues an emergency notification to all configured contacts, and, where integrated with plant control systems, may support an optional process-shutdown signal; manual safety intervention by trained personnel remains necessary at this level and the system is designed to support, not replace, that intervention.

TABLE III. RISK-BASED SAFETY RESPONSE

Risk Level	Detection Condition	Local Response	IoT Response	Automatic Action
Normal	Safe condition	None	Passive monitoring	None
Warning	Increasing gas trend	LED indicator	Warning notification	Optional ventilation
Critical	High-risk condition	Buzzer + LED	Alert to operator	Exhaust fan ON
Emergency	Severe/rapid increase	High-priority alarm	Emergency alert	Maximum safety response

The specific ammonia concentration values that trigger each transition are deployment-specific and must be derived from sensor calibration and applicable occupational exposure limits rather than treated as universal or officially mandated figures.

VIII. IoT-Based Industrial Monitoring

The ESP32 communicates sensor readings and classification results to a cloud platform over Wi-Fi using a lightweight application-layer protocol such as MQTT or HTTP(S). The cloud layer stores incoming readings in a time-series format and drives a web and mobile dashboard that displays the current ammonia sensor value, temperature, humidity, predicted risk level, alarm status, exhaust-fan status, and the timestamp of the latest update, alongside a historical trend graph for each parameter. This IoT layer allows plant safety officers and remote supervisors to observe the state of multiple monitoring nodes from a single interface, review historical trends to identify recurring or slow-developing issues, and receive push or SMS notifications when the risk level escalates, regardless of their physical location within or outside the facility.

IX. Experimental Setup

A bench-scale prototype comprising the ESP32 controller, ammonia sensor, environmental sensor, local alarm devices, exhaust fan, and IoT dashboard is assembled for evaluation, with the gas sensor positioned within a small, controlled test enclosure to allow safe, regulated exposure trials. The proposed experimental methodology can be used to evaluate system behavior under the following test conditions: (1) normal ambient air with no ammonia present; (2) low-level, controlled ammonia exposure within a safe laboratory range; (3) gradually increasing gas concentration to examine the WARNING-to-CRITICAL transition; (4) a rapid step increase in concentration to examine EMERGENCY-level response time; (5) variation in ambient temperature to assess its effect on sensor baseline and classifier robustness; and (6) variation in ambient humidity for the same purpose. As no such controlled trials have yet been carried out for this paper, no experimental outcomes are reported here; Sections X and XI instead describe the intended evaluation methodology and, where illustrative values are shown, they are explicitly labeled as such.

X. Performance Analysis

The proposed system is intended to be evaluated using standard classification and system-timing metrics. Let TP, TN, FP and FN denote, respectively, the number of true positive, true negative, false positive and false negative classifications of hazardous versus non-hazardous states. The following metrics are defined:

$$Accuracy = (TP + TN) / (TP + TN + FP + FN)$$

$$Precision = TP / (TP + FP)$$

$$Recall = TP / (TP + FN)$$

$$F1-score = 2 \times Precision \times Recall / (Precision + Recall)$$

In addition to these classification metrics, the system is to be characterized in terms of false-positive rate and false-negative rate (particularly critical for a safety application, where a false negative represents a missed hazardous condition), detection latency (elapsed time from the onset of a simulated leak to correct risk classification), sensor response time, AI inference time on the embedded controller, IoT communication latency, and average power consumption of the deployed node.

XI. Simulation / Experimental Results

Table IV presents an illustrative comparison of candidate classification algorithms, and Table V presents the intended system-response evaluation parameters. As no calibrated experimental trials have been conducted for this paper, the values in Table IV are illustrative and simulated only, intended solely to demonstrate the reporting format to be used once real data is collected; they must be replaced with results obtained from the calibrated exposure trials described in Section IX before any performance claim is made.

TABLE IV. AI CLASSIFICATION PERFORMANCE (ILLUSTRATIVE / SIMULATED VALUES — TO BE REPLACED WITH EXPERIMENTALLY MEASURED RESULTS)

Algorithm	Accuracy	Precision	Recall	F1-score
Logistic Regression	—*	—*	—*	—*
Decision Tree	—*	—*	—*	—*
Random Forest (proposed)	—*	—*	—*	—*
SVM	—*	—*	—*	—*
KNN	—*	—*	—*	—*

* Values intentionally omitted pending calibrated experimental data; see Section IX.

TABLE V. SYSTEM RESPONSE ANALYSIS

Parameter	Proposed System
Sensor response time	To be experimentally measured
AI inference time	To be measured
IoT communication latency	To be measured
Alarm response time	To be measured
Power consumption	To be measured

XII. Figures

[FIGURE PLACEHOLDER]

FIG. 1. Overall AMMOSHIELD system architecture.

Block diagram showing the signal path from the ammonia gas source through the gas sensor, signal conditioning, ESP32 controller, AI/ML classification module, safety controller, local actuators (alarm, LED, exhaust fan), and the IoT cloud/dashboard layer, with arrows indicating data flow direction.

[FIGURE PLACEHOLDER]

FIG. 2. Hardware block diagram.

Diagram of physical components (ESP32, ammonia sensor, DHT sensor, OLED display, buzzer, LED, relay, exhaust fan, power supply) and their electrical interconnections (analog input, GPIO, I2C, relay-switched AC/DC load).

[FIGURE PLACEHOLDER]

FIG. 3. Ammonia sensor interfacing with ESP32.

Circuit schematic showing the gas sensor's analog output connected to an ESP32 ADC pin through a load-resistor divider, with supply and ground connections and an optional filtering capacitor.

[FIGURE PLACEHOLDER]

FIG. 4. AI/ML data-processing pipeline.

Flow diagram from raw sensor acquisition through filtering, feature extraction (gas value, rate of change, temperature, humidity), the random-forest classifier, and the resulting risk-level output.

[FIGURE PLACEHOLDER]

FIG. 5. AMMOSHIELD software flowchart.

Flowchart representation of Algorithm 1, showing initialization, the continuous sensing loop, feature extraction, classification, the four risk-level decision branches, and cloud upload.

[FIGURE PLACEHOLDER]

FIG. 6. IoT monitoring dashboard.

Mockup of a web/mobile dashboard showing live gas, temperature and humidity values, current risk-level indicator, alarm and fan status, a timestamp, and a historical trend chart.

[FIGURE PLACEHOLDER]

FIG. 7. Ammonia sensor response versus time.

Line graph (to be populated with experimental data) showing sensor output over time during a simulated gradual and a simulated rapid gas-release trial, annotated with the WARNING/CRITICAL/EMERGENCY thresholds once calibrated.

[FIGURE PLACEHOLDER]

FIG. 8. AI classification confusion matrix.

4×4 confusion matrix (Normal, Warning, Critical, Emergency) to be populated once labeled experimental data and trained-model predictions are available.

XIII. Comparison with Conventional Systems

Table VI qualitatively compares AMMOSHIELD against conventional fixed-threshold gas detectors and general IoT-enabled gas detectors of the kind reviewed in Section II. The comparison reflects design intent and architectural capability rather than measured performance, since AMMOSHIELD has not yet undergone field trials.

TABLE VI. PERFORMANCE COMPARISON

Feature	Conventional Detector	IoT Gas Detector	AMMOSHIELD
Gas detection	Yes	Yes	Yes
Continuous monitoring	Limited	Yes	Yes
AI-based classification	No	Limited/No	Yes
Cloud monitoring	No	Yes	Yes
Remote alert	Limited	Yes	Yes
Automated ventilation	Limited	Possible	Yes
Historical data	No/Limited	Yes	Yes
Multi-parameter monitoring	Limited	Yes	Yes
Predictive capability	No	Limited	Proposed

The comparison suggests that AMMOSHIELD's principal differentiation lies not in any single feature, most of which are individually present in some prior systems, but in the combination of ammonia-specific sensing, environmental compensation, AI-based multi-level classification, and automated ventilation control within one integrated architecture. This is presented as a design distinction rather than a demonstrated performance superiority, which would require the field validation discussed in Section XVI.

XIV. Security, Reliability and Fail-Safe Design

Several failure modes must be addressed for AMMOSHIELD to function as a dependable safety system rather than only a monitoring convenience. Sensor failure or disconnection should be detected and reported as a fault state rather than silently interpreted as a NORMAL reading. Sensor drift over time should be managed through periodic recalibration schedules and, where feasible, redundant sensing to cross-check readings. The system should be tuned to minimize both false alarms, which can lead to alarm fatigue and reduced operator trust, and false negatives, which are the more safety-critical failure mode. Wi-Fi or cloud-connectivity failure must not disable local safety functions: the buzzer, LED, and exhaust-fan control logic should operate autonomously from locally computed sensor readings and an on-device classifier, with cloud reporting treated as an auxiliary rather than a mandatory function. Similarly, power failure should be considered in the design of backup power for at least the local alarm subsystem. Where the system

exposes network or cloud interfaces, standard cybersecurity practices, including authenticated API access and encrypted data transmission, should be applied to prevent unauthorized access to plant safety data or spoofed alarm suppression. A manual override or physical emergency-stop interface should remain available at all times, independent of the automated logic. In general, critical industrial safety functions should not depend exclusively on cloud connectivity, and the architecture described in Sections III–VII is designed with this local-first principle in mind.

XV. Advantages

The proposed architecture offers several potential advantages over conventional installations: earlier detection of developing leaks through trend-based rather than purely threshold-based analysis; continuous, unattended monitoring that does not depend on manual inspection rounds; AI-based risk classification that can account for multiple sensor features simultaneously; IoT connectivity that enables remote visibility and off-site alerting; automated ventilation response that can begin mitigating a hazardous condition before human intervention arrives; historical data logging that supports trend analysis and regulatory record-keeping; a comparatively low prototype cost based on widely available components; a scalable architecture that can be replicated across multiple points in a facility; and real-time decision-making at the edge that keeps essential safety functions independent of network availability.

XVI. Limitations

The proposed system also has limitations that should be acknowledged. Semiconductor gas sensors require periodic calibration and are subject to long-term drift and cross-sensitivity to other reducing gases. Environmental effects such as extreme temperature or humidity may still degrade sensor accuracy despite compensation. The performance of the AI/ML classifier is fundamentally dependent on the quality, size, and representativeness of the training dataset, and a model trained on limited laboratory trials may not generalize well to the diversity of conditions found in an operating plant. Reliance on Wi-Fi connectivity for cloud reporting introduces a dependency that must be mitigated through the local-first fail-safe design described in Section XIV. As a bench-scale prototype, the system as described has not undergone the environmental, electrical, and functional-safety certification (e.g., relevant IEC/ATEX or hazardous-area standards) that would be required before installation in a certified industrial safety role, and field validation under real plant conditions remains necessary before any accuracy or reliability claims can be made.

XVII. Future Work

1. Implementation of the trained classifier directly on the ESP32 using TinyML frameworks to further reduce inference latency and network dependency.
2. Exploration of edge-AI processing techniques to support on-device model updates.
3. Integration of LoRaWAN communication for long-range, low-power coverage of large industrial sites.
4. Evaluation of 5G connectivity for latency-sensitive, high-density sensor deployments.
5. Extension to multi-gas sensing to cover additional industrial hazards beyond ammonia.
6. Sensor-fusion techniques combining multiple sensor modalities for improved reliability.
7. Incorporation of explainable-AI techniques so operators can understand the basis of a given risk classification.
8. Development of predictive leak-forecasting models using longer-horizon historical data.
9. Integration with plant-level PLC/SCADA systems for coordinated automated shutdown.
10. Use of industrial-grade, certified gas sensors for field deployment.
11. Design of a redundant, multi-node safety architecture with cross-validation between sensors.
12. Exploration of digital-twin-based industrial monitoring for facility-wide situational awareness.

XVIII. Conclusion and Future Work

This paper has presented AMMOSHIELD, an AI-based smart ammonia gas leak detection and industrial safety monitoring system intended for deployment in fertilizer, cold-storage, chemical, and food-processing facilities where ammonia handling poses a recognized occupational hazard. The proposed architecture combines ammonia-specific gas sensing with ESP32-based embedded processing, environmental parameter monitoring, and an AI/ML classification layer built around the Random Forest algorithm to produce a graded risk assessment rather than a single fixed-threshold alarm. This classification layer is coupled with IoT/cloud connectivity for real-time remote visualization and historical logging, and with an automated safety-response subsystem that escalates local alarms and exhaust ventilation in proportion to the assessed risk level, while remaining functional independently of network connectivity for critical local safety actions. The hardware implementation, software control algorithm, and evaluation methodology have been described in detail, and illustrative reporting formats for classification and system-response performance have been outlined for use once calibrated experimental trials are

conducted. No specific accuracy or performance figures are claimed in this paper, as the system described is a bench-scale design that has not yet undergone the controlled exposure trials and field validation outlined in Sections IX and XVI. Future work will focus on edge-AI and TinyML deployment, multi-gas sensing, sensor fusion, explainable AI, predictive leak forecasting, integration with plant-level SCADA systems, and, ultimately, field validation under real industrial conditions, with the broader goal of contributing a practical, data-driven improvement to industrial ammonia safety practice.

References

- [1] G. J. V. Lakshmi, G. Puthilibhai, and S. R. L. Siddarth, "Implementation of Ammonia Gas Leakage Detection and Monitoring System Using Internet of Things," in Proc. 2019 Int. Conf. Communication and Signal Processing (ICCSP), Chennai, India, 2019, pp. 1–6.
- [2] A. Rokade, S. Shaikh, S. Nadaf, A. Attar, P. Kandekar, and A. Kaur, "A Survey on Enhanced Safety of IoT Enabled Gas Leakage Detection System," in M. Tuba, S. Akashe, and A. Joshi (Eds.), Springer, 2026.
- [3] M. Santiputri, "IoT-Based Gas Leak Detection Device," in Proc. 2018 Int. Conf. Computer, Control, Informatics and its Applications (IC3INA), Tangerang, Indonesia, 2018, pp. 1–4.
- [4] R. J. Jadhav, P. Radhakrishnan, S. Mukherjee, R. Goyal, R. Goyal, and S. Thakur, "Internet of Things Enabled Gas Leakage Detection over Industrial Areas Using Powerful MQ Series Sensor and Controller," in Proc. 2024 Int. Conf. Smart Technologies and Systems for Next Generation Computing (ICSTSN), Pune, India, 2024, pp. 1–6.
- [5] V. P. Sharma, R. Dugyala, V. Padmavathi, and G. V. R. Gurram, "Gas Leakage Detection System Using IoT and Cloud Technology: A Review," E3S Web of Conferences, vol. 391, art. 01063, 2023.
- [6] G. A. Senthil, P. Suganthi, R. Prabha, and S. Sridevi, "An Enhanced Smart Intelligent Detecting and Alerting System for Industrial Gas Leakage Using IoT in Sensor Network," in Proc. 2023 Int. Conf. Intelligent Sustainable Systems (ICISS), Coimbatore, India, 2023, pp. 1–6.
- [7] N. Manjunathan, S. Muthulingam, and D. Jaganathan, "IoT Based Gas Leakage Detection System," in Proc. Int. Conf. Sustainable Computing and Data Communication Systems (ICSCDS), 2023, pp. 1–5.
- [8] M. Al-Jemeli and M. K. A. Al-Adilee, "Portable Gas Leak Detection System Using IoT and Off-the-Shelf Sensor Node," Indonesian Journal of Electrical Engineering and Computer Science, 2021.
- [9] F. Chraim, Y. B. Erol, and K. Pister, "Wireless Gas Leak Detection and Localization," IEEE Trans. Industrial Informatics, 2016.
- [10] Bin et al., "Tensor-Based Approach for LPG Leakage Detection from Surveillance Thermal Cameras," IEEE Trans. Industrial Informatics, vol. 17, no. 12, pp. 8296–8305, Dec. 2021.
- [11] A. U. Rehman and A. Bermak, "Heuristic Random Forests (HRF) for Drift Compensation in Electronic Nose Applications," IEEE Sensors Journal, vol. 19, no. 4, pp. 1443–1453, 2019.
- [12] D. Ma, J. Gao, Z. Zhang, and H. Zhao, "Gas Recognition Method Based on the Deep Learning Model of Sensor Array Response Map," Sensors and Actuators B: Chemical, vol. 330, art. 129349, 2021.
- [13] U. Yaqoob and M. I. Younis, "Chemical Gas Sensors: Recent Developments, Challenges, and the Potential of Machine Learning — A Review," Sensors, vol. 21, no. 8, pp. 2877–2917, 2021.
- [14] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.
- [15] A. D. Wilson and M. Baietto, "Applications and Advances in Electronic-Nose Technologies," Sensors, vol. 9, no. 7, pp. 5099–5148, 2009.
- [16] F. Röck, N. Bârsan, and U. Weimar, "Electronic Nose: Current Status and Future Trends," Chemical Reviews, vol. 108, no. 2, pp. 705–725, 2008.
- [17] Occupational Safety and Health Administration (OSHA), "Permissible Exposure Limits — Annotated Table Z-1," 29 CFR 1910.1000, U.S. Dept. of Labor. [Online]. Available: <https://www.osha.gov/annotated-pels/table-z-1>
- [18] National Institute for Occupational Safety and Health (NIOSH), "Ammonia: Immediately Dangerous to Life or Health Concentrations (IDLH)," Centers for Disease Control and Prevention, May 1994. [Online]. Available: <https://www.cdc.gov/niosh/idlh/7664417.html>
- [19] Espressif Systems, "ESP32 Series Datasheet," Espressif Systems, Shanghai, China, 2023.
- [20] Hanwei Electronics Co., Ltd., "MQ-137 Ammonia Gas Sensor Technical Data," Zhengzhou, China.