

Optimizing internet of things based gas sensors: deep learning and performance optimization strategies

Mariam M. Abdellatif^{1,6}, Mehmet Akif Çifçi^{2,3}, Asmaa A. Ibrahim¹, Hany M. Harb⁴, Abeer S. Desuky^{1,5}

¹Faculty of Science, Al-Azhar University, Cairo, Egypt

²Engineering and Informatics Department, Klaipėdos Valstybinė Kolegija/Higher Education Institution, Klaipėda, Lithuania

³Department of Computer Engineering, Faculty of Engineering and Natural Sciences, Bandırma Onyedi Eylül University, Balıkesir, Türkiye

⁴Faculty of Engineering, Al-Azhar University (Boys), Cairo, Egypt

⁵Higher Institute of Computer Science and Information Systems, Cairo, Egypt

⁶ElSewedy University of Technology, Cairo, Egypt

Article Info

Article history:

Received Jul 10, 2024

Revised Jun 18, 2025

Accepted Jun 30, 2025

Keywords:

Artificial intelligence

Deep learning

Internet of things

Late fusion

Multimodal data fusion

Particle swarm optimization

Transfer learning

ABSTRACT

The rapid growth of industrialization and internet of things (IoT) driven advancements in Industry 5.0 necessitates efficient and user-friendly engineering solutions. Gas leakage incidents in coal mines, chemical enterprises, and households pose significant risks to ecosystems and human safety, emphasizing the need for automated and rapid gas-type detection. Traditional detection methods rely on single-source data and focus on isolated spatial or temporal features, limiting accuracy. This paper proposes a multimodal artificial intelligence (AI) fusion technique combining pre-trained convolutional neural networks (CNNs), such as VGG16, with a deep neural network (DNN) model. The particle swarm optimization (PSO) algorithm optimizes CNN hyperparameters, outperforming traditional trial-and-error methods. The system addresses challenges posed by gases being odorless, colorless, and tasteless, which limit conventional human detection methods. By leveraging sensor fusion, the late fusion technique integrates distinct network architectures for unified gas identification. Experimental results demonstrate 95% accuracy using DNN with gas sensor data, 96% with optimized VGG16 using thermal imaging, and 99.5% through multimodal late fusion. This IoT-enhanced solution outperforms single-sensor approaches, offering a robust and reliable gas leakage detection system suitable for industrial and smart city applications.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Mariam M. Abdellatif

Faculty of Science, Al-Azhar University

Cairo, Egypt

Email: mariemmohammed974@gmail.com

1. INTRODUCTION

In the rapidly advancing internet of things (IoT) era, gas sensors have become crucial in various settings, from homes to healthcare facilities [1], [2]. These sensors are vital for detecting harmful gases and ensuring air quality, but face challenges like cross-sensitivity and limited selectivity [3]. To overcome these, smart gas sensing integrates sensor arrays, signal processing, and machine learning, enhancing precision and sensitivity.

Despite technological advancements, gas leakage remains a significant risk, causing industrial disasters and environmental harm [4]. Efficient and automated gas leak detection systems are essential for safety. Recent innovations include metal oxide semiconductor (MOX) sensors in “electronic nose” (e-nose) setups, which use gas sensor arrays, signal processing modules, and pattern recognition systems [5]. Artificial

intelligence (AI) techniques have further improved gas detection, with applications in multiple industries [6]–[8]. AI enhances the e-nose's ability to accurately identify and classify gases, crucial for preventing accidents and health risks. Previous methods relied on single deep learning (DL) models, but combining multiple DL structures has improved detection accuracy [5]. Integrating spatial, spectral, and temporal data enhances performance, and reducing attribute sizes increases recognition accuracy [9]. An AI-driven approach using DL frameworks now integrates data from diverse sources, combining gas sensors and thermal imaging for effective gas detection. These advancements provide valuable insights into enhancing safety measures and developing robust gas-sensing solutions. In this approach, two independent models are first trained on their respective datasets. The gas sensors model processes the sequential data collected from various gas sensors over time, extracting valuable temporal patterns and trends related to gas concentrations. On the other hand, the thermal camera model analyzes the thermal images captured by the camera, detecting temperature variations that may indicate gas leaks or anomalies. After training, both models are used to make predictions on the same set of test samples or data instances. The gas sensors model provides predictions based on its sequence data analysis, while the thermal camera model generates predictions based on its interpretation of the thermal images. The predictions from both models are then combined or fused using a late fusion technique. Late fusion between gas sensors sequence data and thermal camera images data is particularly valuable in real-world scenarios where multiple sources of information are available, and the fusion of these diverse data sources can provide a more comprehensive and reliable gas leak detection solution. However, the effectiveness of the late fusion [10] approach relies on carefully designing the fusion strategy and ensuring that the models are well-trained on their respective datasets to capture relevant patterns and features specific to each modality. The significant contributions of this research are as follows:

- a. The particle swarm optimization (PSO) algorithm is utilized to fine-tune various hyperparameters associated with the VGG16 model.
- b. A pioneering AI-driven framework that integrates two distinct modalities is introduced to bolster the robustness and dependability of gas detection.
- c. The use of late fusion techniques for gas detection and the identification of leaked gases are demonstrated by leveraging the outputs of deep learning architectures VGG16 and deep neural network (DNN).
- d. More accurate and effective detection of gas is achieved based on the proposed approaches.

The structure of this paper is organized as follows: section 2 provides a review of related research. Section 3 covers the fundamentals and background of DNN, convolutional neural networks (CNNs), and PSO. Section 4 describes the dataset used in the experiments. Section 5 presents the proposed approach for diagnosing plant diseases. Section 6 focuses on the analysis and discussion of experimental results. Lastly, section 7 offers the concluding remarks.

2. RELATED WORKS

Numerous studies have investigated the integration of chemical and substance-detecting sensors into IoT platforms. For example, Venkatasubramanian *et al.* [11] enhances detection and management of breakdowns in industrial IoT (IIoT) devices using real-time sensor data for decision-making, addressing challenges like denoising and missing data with preprocessing and data fusion techniques. Models like PropensityNet, DNN, and CNN long short-term memory (CNN-LSTM) are used, proving effective with Case Western Reserve University (CWRU) data for fault detection. In [12], the focus is on hazardous gas leakage, emphasizing safety in chemical industries. The study proposes an IoT-based strategy using an Arduino UNO R3 and sensors to monitor gases like NO₂ and carbon monoxide. Machine learning enhances data accuracy, and hybrid hidden Markov and AI models improve error detection. Studies also combine gas sensors and thermal imaging for gas detection. For instance, Peng *et al.* [13] uses CNN to categorize gases using eight sensors, while [14] employs various DL models, including CNN, LSTM, and autoencoders for gas leak detection, achieving notable accuracy. Researches like in [15] and [16] utilize CNN and LSTM for extracting spatial-temporal information from sparse data. Researches like [17] and [18] explore machine learning techniques for analyzing infrared (IR) thermal images in gas detection. Bin *et al.* [19] introduces a tensor-based leakage detection method with ResNet50 for gas identification, and Liu *et al.* [20] uses an IR thermal imaging system with diverse machine learning algorithms for detecting flammable gas leaks. In [21], pattern recognition techniques like random forest, support vector machine (SVM), and shallow multi-layer perceptron algorithms identify gases at low concentrations using a SnO₂ gas sensor, with artificial neural networks achieving high effectiveness. Related studies [22] and [23] use pattern recognition methods for classifying gravitational search algorithm (GSA) data. Moreover, Nikolopoulos *et al.* [24] investigates thermal imaging for gas leakage detection, using VGG-16 and k-NN for classifying gases with high accuracy, depending on the specific application and trade-offs between computational cost and performance. These studies illustrate the synergy between advanced sensor technology and analytical techniques for enhancing industrial safety and gas detection capabilities.

3. BASICS AND BACKGROUND

3.1. Internet of things

The internet of things (IoT) connects devices and systems, enabling seamless communication, data exchange, and intelligent decision-making by linking physical objects with sensors and actuators to the internet. In terms of architecture, IoT consists of several interconnected components [25]. At the foundation are sensors and actuators, which capture data such as temperature and motion, and enable physical actions based on this data. These devices rely on connectivity mechanisms like Wi-Fi, Bluetooth, or cellular networks to communicate and transmit data efficiently. Once transmitted, cloud computing platforms store and process this data, providing the computational power necessary for real-time insights. Data analytics plays a crucial role at this stage, where machine learning techniques are employed to extract meaningful patterns and support predictive actions. The final layer involves user interfaces, such as applications and dashboards, which facilitate remote monitoring and management of IoT systems. IoT applications span a wide range of industries, including smart cities—where they support traffic management and energy optimization—and healthcare, enabling real-time health monitoring and improved patient outcomes. This work builds upon these capabilities by utilizing multimodal data for gas detection. Through the application of fusion techniques, discussed in the following section, the system aims to enhance detection accuracy and reliability in safety-critical environments.

3.2. Multimodal data fusion approaches

Fusion strategies in multimodal machine learning can be either model-based or model-agnostic. Model-agnostic fusion mixes modalities, such as thermal imaging and sensor data, employing early, late, or intermediate fusion techniques. Early fusion incorporates raw data during the initial processing stage, capturing interactions between modalities, whereas late fusion, or decision-level fusion, merges independent predictions using methods such as majority voting. Intermediate fusion mixes features from different modalities at higher abstraction levels, resulting in improved performance. Model-based fusion, also known as multitask fusion, trains models on many tasks at the same time while sharing representations across modalities to promote generalization. The next section discusses deep learning models for gas detection [26].

3.3. Deep learning models

Deep learning, a key subset of machine learning, excels in processing complex data. Two essential models are DNNs and CNNs, often enhanced by transfer learning techniques [27]. The following subsections provide a brief overview of DNNs, CNNs, and the role of transfer learning in enhancing their performance.

3.3.1. Deep neural networks

DNNs are advanced artificial neural networks with multiple hidden layers that automatically learn complex patterns from raw data. They excel in tasks such as image recognition, speech processing, and recommendation systems. Despite requiring significant computational resources, advancements in hardware and algorithms have enabled their widespread adoption [28].

3.3.2. Transfer learning with CNNs

CNNs consist of convolutional, pooling, and fully connected layers, each serving distinct roles in image processing. Transfer learning leverages pre-trained models like VGG16, known for its success in tasks such as ImageNet classification. By reusing VGG16's convolutional layers, transfer learning enables effective feature extraction or fine-tuning for new tasks with limited data. The next section addresses hyperparameter optimization for these models using particle swarm optimization [29].

3.4. Particle swarm optimization algorithm

PSO simulates the behavior of bird and fish groups, with particles distributed in a search space and evaluated based on an objective. Each particle adjusts its position using its current location, best-known position, and neighbors' positions. This iterative process continues until the desired outcome is achieved [30]. The velocity and position updates for each particle at iteration $t+1$ are given by:

$$v_{t+1} = \omega \times v_t + c_1 \times r_1 \times (p_{(best)t} - x_t) + c_2 \times r_2 \times (G_{(best)t} - x_t) \quad (1)$$

$$x_{t+1} = x_t + v_{t+1} \quad (2)$$

Where, t is the iteration number, ω (omega) is the weight, c_1 and c_2 are cognitive and social parameters, and r_1, r_2 are random numbers between 0 and 1. For this research, the cost function is the mean square error (MSE) of a neural network with one hidden layer and 10 neurons. The PSO algorithm systematically

minimizes this cost by refining inputs, starting with a random subset of features and gradually achieving optimal results, as shown in Figure 1.

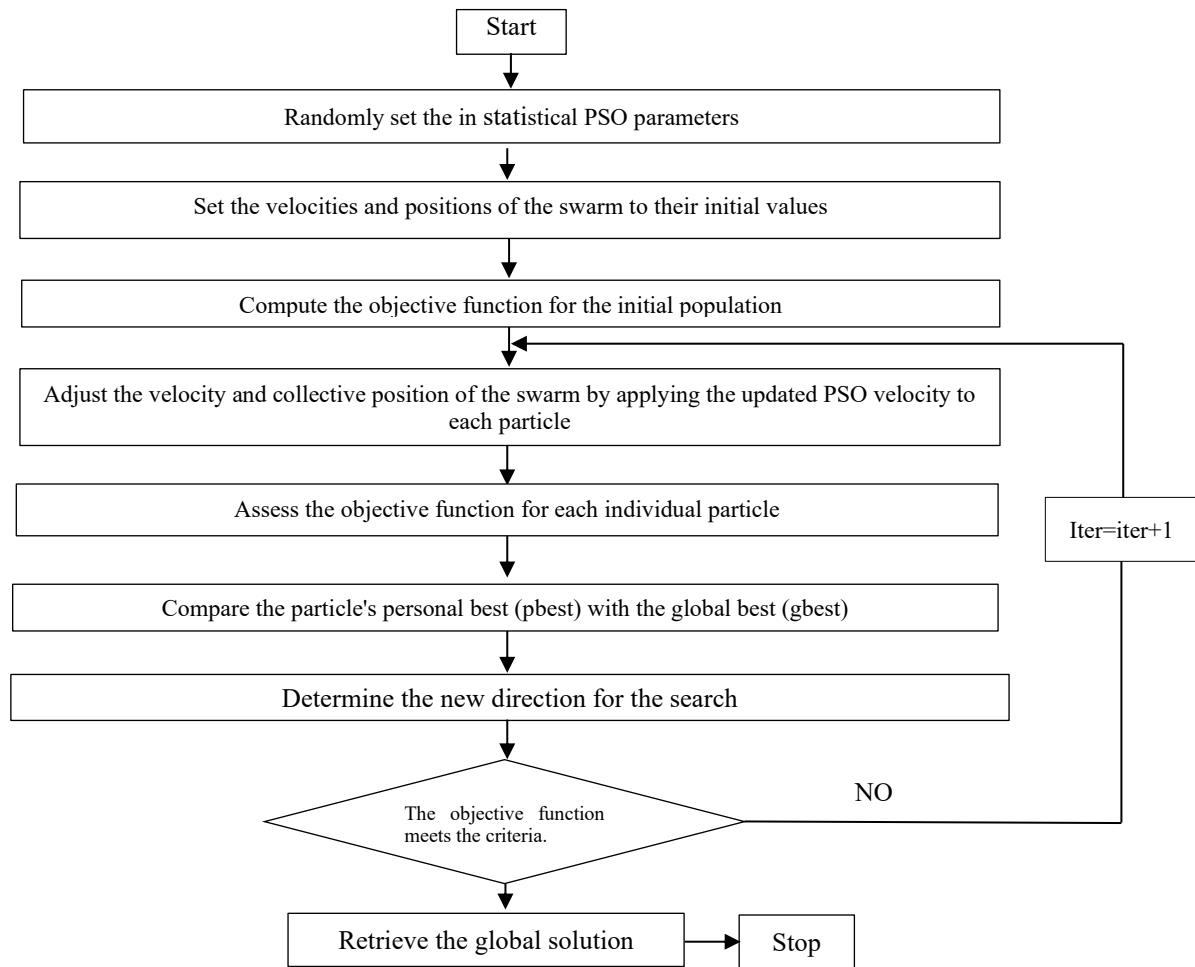


Figure 1. Flowchart of PSO algorithm

4. DATASET DESCRIPTION

The multimodal gas data is a comprehensive dataset that amalgamates vital information from seven distinct gas sensors and thermal images captured using a thermal camera. The dataset encompasses measurements of two different gases, creating four well-defined classes: Perfume, NoGas, Smoke, and Mixture of perfume and Smoke. The data collection process involved utilizing seven metal oxide gas sensors, MQ2, MQ5, MQ3, MQ8, MQ6, MQ7, and MQ135, along with a sophisticated thermal imaging camera. This multimodal approach enabled the simultaneous acquisition of numerical values from the gas sensors and thermal images, providing a diverse and informative dataset [31]. The comprehensive details of the dataset are elaborated in subsequent sections.

4.1. Gas sensors

Gas sensors detect gases by converting chemical data into electrical signals, with MOX sensors interfaced to a microcontroller for processing and data communication. Analog-to-digital converters (ADCs) transform analog outputs into digital data, while wired or wireless communication transmits the data for storage and analysis. The dataset includes seven MOX sensors (MQ2, MQ3, MQ5, MQ6, MQ7, MQ8, and MQ135) integrated into an IoT system for automated data collection, as shown in Figure 2. These sensors are compact, durable, and responsive to gases like CO, methane, LPG, and alcohol as shown in Table 1, with sensitivity, selectivity, and response time critical to their performance. During data collection, the sensors were placed 1 mm apart.

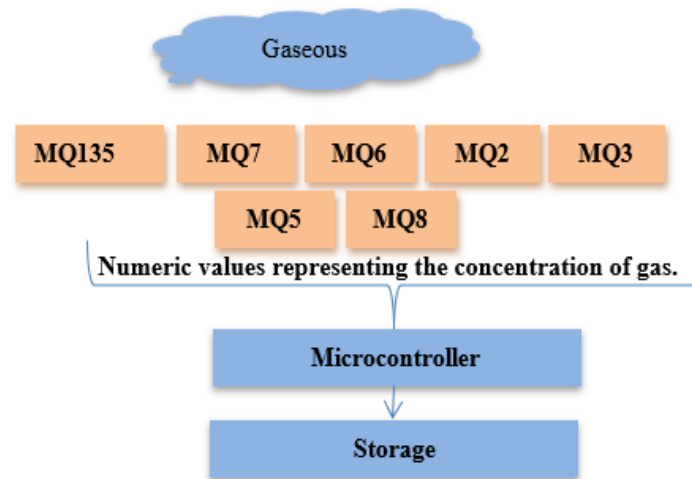


Figure 2. IoT-enabled the process of gathering data

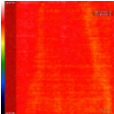
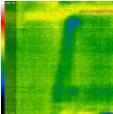
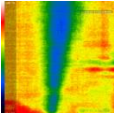
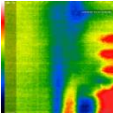
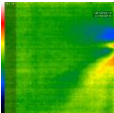
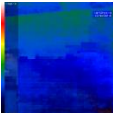
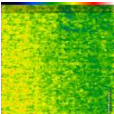
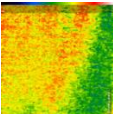
Table 1. Sensing gases and gas sensors

Sensor	Sensitivity gas
MQ2	Propane , Butane, Methane, LPG, Smoke
MQ8	
MQ3	Smoke, Ethanol, Alcohol
MQ135	Air Quality (Benzene, Smoke)
MQ6	Butane gas, LPG
MQ5	Natural gas, LPG
MQ7	Carbon Monoxide

4.2. Thermal camera images

The dataset uses a thermal camera that captures temperature fluctuations via infrared light, with each pixel acting as an infrared temperature sensor. Images are output in RGB format, enabling visualization irrespective of lighting conditions. The Seek thermal camera, with a 206×156 sensor, a -40 °C to 330 °C range, and 32,136 pixels, was used. Gas sensors and the thermal camera collected data simultaneously, as no public dataset combining thermal images and gas sensor data existed. Data was gathered by positioning seven gas sensors 1 mm apart, monitoring gases from perfumes and incense at intervals over 1.5 hours. Three classes—NoGas, Perfume, and Smoke—were sampled, with 6,400 total samples (1,600 per class). Sensor outputs were converted into 10-bit digital values for analysis, as shown in Table 2.

Table 2. Examples of the data for the thermal images and the related gas array readings

Class type	Gas	Thermal image	Gas	Thermal image
NoGas	[733,530,405,414,589,628,456]		[559,516,374,335,664,448,415]	
Perfume	[738,529,394,395,566,577,442]		[794,518,494,447,686,658,490]	
Smoke	[682,428,299,333,592,596,335]		[686,429,299,333,591,598,335]	
Mixture	[632,443,444,405,401,309,430]		[506,392,344,311,395,222,302]	

5. THE PROPOSED THERMAL-GAS FUSION DETECTION MODEL

Optimizing internet of things based gas sensors: deep learning and ... (Mariam M. Abdellatif)

The thermal-gas fusion detection model combines gas sensors and a thermal camera for accurate gas detection, with a block diagram as shown in Figure 3 and three scenarios detailing the data collection and training process. The following subsections outline the system architecture, data collection scenarios, and the training methodology used in the proposed model. This integrated approach is designed to improve detection accuracy and robustness across varying environmental conditions.

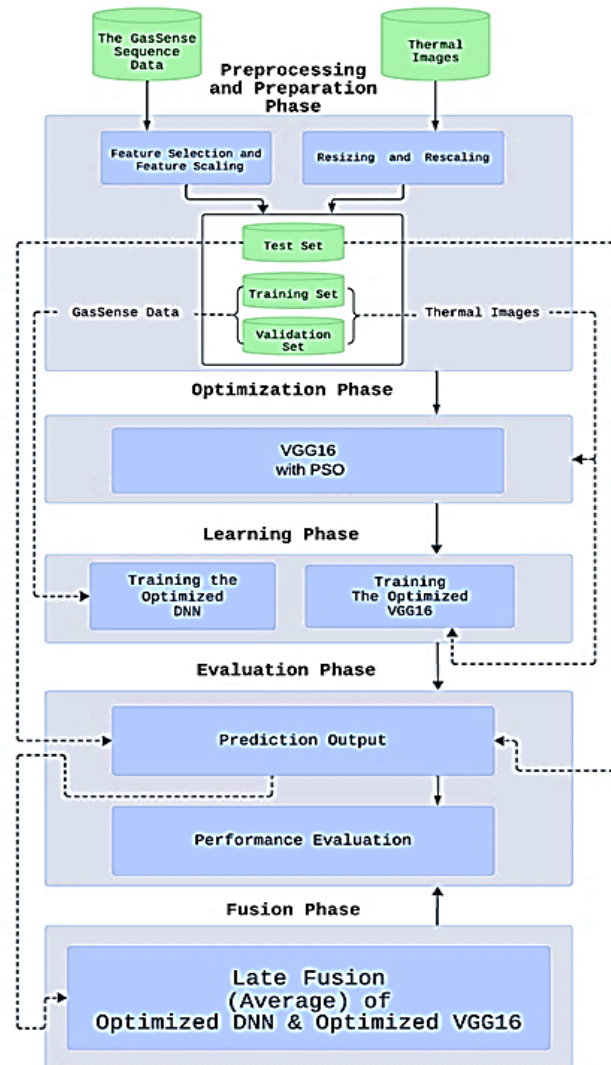


Figure 3. The schematic diagram represents the proposed approach

5.1. Gas sensors of scenario I

5.1.1. Preprocessing phase for gas sensors

In the preprocessing phase of the proposed approach, several crucial data preparation steps are performed to effectively prepare the dataset for training the machine learning model. Initially, the dataset is divided into feature vectors and their corresponding targets. Subsequently, feature selection is performed using the SelectKBest algorithm with the $f_classif$ scoring function, which selects the top 6 most significant features from the original feature set based on their relevance to the target variable. After that, label encoder is employed to convert the categorical labels of the target variable into numerical representations. Further enhancing the data, the encoded target variable is transformed into binary vectors using one-hot encoding, resulting in a binary matrix representing the classes. To ensure consistency and uniformity in feature scales, the numerical columns of the feature matrix are scaled using MinMaxScaler, rescaling their values between 0 and 1. Finally, the preprocessed data is split into training and testing sets, with 20% reserved for testing, while maintaining reproducibility by setting a random state of 0. These meticulous preprocessing steps

optimally equip the data for training and evaluating the deep learning model in the subsequent stages of the proposed approach.

5.1.2. Learning phase

In this phase, given the sequential nature of sensor measurements, a suitable choice for a sequence model is the deep neural network (DNN). In the learning phase of the proposed approach, the DNN model consists of multiple layers: input, hidden, and output. The input layer is defined using the 'Dense' function with 1,500 neurons and a ReLU activation function, ensuring non-linearity and effective feature representation. The input shape of (6) indicates that the input data has six features. Subsequently, two hidden layers are added with 1,500 and 1,000 neurons, respectively, employing ReLU activation functions to enhance feature extraction further and introduce non-linearity. Batch Normalization layers are integrated after each hidden layer to normalize the output, avoiding overfitting and improving convergence. Dropout layers are incorporated after the first two hidden layers to mitigate overfitting during training, with dropout rates of 0.7 and 0.1, respectively, randomly deactivating neurons during training. This regularization technique improves the generalization capabilities of the model and prevents it from becoming overly dependent on specific neurons. After the last hidden layer, the output layer is added with four neurons, representing the four classes, and utilizes the SoftMax activation function, enabling probabilistic predictions. The model is then compiled using the Adam optimizer with a learning rate 1e-4 and categorical cross-entropy as the loss function to optimize the model's performance.

5.2. Thermal images of scenario II

5.2.1. Data preparation phase for thermal image

The thermal camera is a non-invasive instrument for assessing temperature variations by detecting infrared light. Its image sensor's pixels serve as infrared temperature sensors, simultaneously recording temperatures in several places. Images representing the outcome are shown in RGB format, correlating with temperature data. The thermal camera has a substantial advantage over traditional cameras in that it can operate efficiently in various situations, regardless of their shape or roughness. The choice of the Seek thermal camera for this study was based on its compact size, a thermal sensor resolution of 206×156 pixels, a wide 36-degree field of view, temperature measurement capabilities ranging from -40 °C to 330 °C, a framerate of below 9 Hz, and a high thermal pixel count of 32,136, which allows for smooth visualization of thermal images. A validation set was created to ensure precise monitoring of the models' performance during the training phase by randomly assigning 20% of the samples from the original training set. The data set was split into three subsets: the training, validation, and test sets. Table 3 presents the number of samples in each category, along with the total count of samples in both the training and validation sets after the split. This division facilitates practical training and evaluation of the models, enabling them to learn from diverse data while allowing for unbiased assessment of their accuracy and generalization capabilities. Various data augmentation methods were applied to the training samples to prevent overfitting and improve generalization. These techniques include rotation, width adjustment, height adjustment, shear, zoom, and horizontal flipping. Subsequently, both the validation and test datasets were resized to a resolution of 256 x 256 and normalized by dividing the color value of each pixel by 255, ensuring that all pixel values fall within the range of [0, 1]. During the training phase, the augmentation settings involve rescaling the pixel values to the desired range of 0 to 1, rotating images by up to 30 degrees, shifting the height and width by 20%, applying shear transformations, horizontal flipping, and zooming the images by 30%. The '*fill_mode*' parameter is also set to 'nearest' to handle any potential missing pixels after the transformations. In contrast, the test data underwent rescaling from 0 to 1.

Table 3. The counts of samples for each category and the total combined samples in both the training and validation sets are provided

Crop category	Training set images	Validation set images
NoGas	1,280	160
Mixture	1,280	160
Perfume	1,280	160
Smoke	1,280	160
Total	5,120	640

5.2.2. Optimization phase for thermal image

The use of pre-trained CNN architectures places several restrictions on the optimization phase. The majority of these architectures' hyperparameters are fixed. Still, a few important ones need to be adjusted, including the *mini_batch* size, learning rate, quantity of units in dense layers, and dropout rate. To overcome

this, the study uses the optimized linear particle swarm optimization (OLPSO) algorithm to tune the hyperparameter. According to Table 4, the OLPSO algorithm uses two coordinates as particles to indicate the batch size and learning rate of the Adam optimizer. Random initialization creates the initial swarm particles. In particular, the initialization uses integer values from 25 to 128 and decimal values from 0.0 to 1.0. The OLPSO algorithm finds the best values for these hyperparameters through iterative optimization, ultimately improving the performance and accuracy of the proposed approach.

Table 4 details the use of the OLPSO algorithm for tuning hyperparameters in a thermal image analysis model using pre-trained CNN architectures. Specifically, it shows that OLPSO uses two coordinates as particles: one for batch size (P1) ranging from 25 to 128, and another for learning rate (P2) ranging from 0.0 to 1.0. Through iterative optimization, OLPSO identifies the best values for these hyperparameters, enhancing the performance and accuracy of the model.

Table 4. Particle specification for hyperparameter optimization in the proposed methodology

Particle (p)	Batch size	Learning rate
P ₁	25 to 128	Not Applicable
P ₂	Not Applicable	0.0 to 1.0

5.2.3. Learning phase

This phase proposed the classification of gas samples in this investigation using the optimized VGG16 model. A deep learning model is constructed to perform image classification using transfer learning. The model is based on the VGG16 architecture, a pre-trained CNN mentioned in section 3.3.2. The model architecture consists of a pre-trained VGG16 model as the base, followed by additional layers. The VGG16 base is frozen (non-trainable) to retain its pre-trained weights and features. A dense layer with 256 units and ReLU activation is added, followed by dropout and batch normalization layers to prevent overfitting. The output layer comprises four units with a SoftMax activation function for multiclass classification. The model is optimized using the Adam optimizer with a learning rate $2e-5$ and categorical cross-entropy as the loss function. The model checkpoint and early stopping callbacks are employed to improve model performance and prevent overfitting. During the training phase, the model was trained using the training set and validated using the validation set. Batches of augmented images were used for both training and validation. The model underwent training for 100 epochs, and the best model weights were saved for future use and evaluation after these pre-trained models had finished their training.

5.2.4. Evaluation phase for gas sensors and thermal images

During this phase, the predictive capabilities of the proposed approach were assessed using five commonly employed evaluation metrics in classification problems: the confusion matrix, accuracy, precision, F1-score, and recall. Accuracy quantifies the ratio of correct predictions to all predictions made and is typically expressed as a percentage, calculated using (3). Precision assesses the model's ability to predict values for specific categories accurately; its calculation is detailed in (4). Recall [31] gauges the proportion of correctly classified positive patterns and (5) outlines its derivation. The F1-score represents the weighted average of precision and recall, as computed in (6). Both macro and micro averages were employed to evaluate performance, excluding the confusion matrix comprehensively. The confusion matrix, a widely used tabular representation, illustrates the classification model's performance on the test set. It enables a comparison between predicted and actual outcomes, offering valuable insights into the model's accuracy in correctly identifying values. By aligning predicted and actual outcomes, the confusion matrix aids in assessing the overall performance of the model and identifying any misclassifications or errors that may have occurred during the classification process.

$$\text{Accuracy} = \frac{\text{correct prediction}}{\text{Total prediction}} \quad (3)$$

$$\text{Precision} = \frac{\text{Correct Prediction s for a Particular Category}}{\text{Total Prediction s for that Category}} \quad (4)$$

$$\text{Recall} = \frac{\text{Correctly Predicted Instances of a Category}}{\text{Total Instances of that Category}} \quad (5)$$

$$\text{F-Measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (6)$$

5.3. Fusion phase in scenario III

In this phase, the primary objective is to make precise decisions by amalgamating features derived from thermal images and gas sensor measurements. The designs of the models for fusing image and sequence data are depicted in Figure 1. The central aim is constructing a unified classifier that adeptly merges information from thermal images and the gas sensor sequence array. The outputs of the DNN and VGG16 models must be in the same feature space for the fusion to be effective. A Late Fusion model that uses decision-level fusion is also used to accomplish this. The individual predictions of the DNN and VGG16 models are obtained first. The final result of fusion, known as average fusion, is then considered the arithmetic average of each model prediction during the late fusion process. The available dataset is used to develop and validate the late fusion model. The performance of the fusion models is presented in the following section. The fusion process aims to make the most of the advantages of both thermal pictures and gas sensor data, enhancing classification precision for gas sample analysis.

6. RESULT AND DISCUSSION

This section presents and analyzes the outcomes of the proposed approach applied to three distinct gas detection scenarios. Leveraging the Keras library, a Python-based high-level API renowned for building and implementing deep learning architectures. The framework of the approach is seamlessly constructed and integrated with powerful numerical computational libraries like TensorFlow. A DNN is trained in scenario I using only the sequence gas sensor modality. In scenario II, VGG16 is utilized to process thermal images of gas sensors. Finally, late fusion is employed in scenario III, by combining features from the DNN and VGG16 models through decision-level fusion. In this approach, individual predictions are obtained from the DNN and VGG16 models before performing the late fusion to achieve gas detection objectives.

6.1. Gas sequences result of scenario I

The outcomes derived from this scenario are not just showcased, but meticulously examined through a comprehensive analysis conducted in three distinct and crucial phases. The subsequent subsections detail the three key phases of analysis: preprocessing, learning, and evaluation of the DNN model.

6.1.1. Preprocessing phase

In the paper's feature selection and scaling section, the authors utilized the scikit-learn library to perform these preprocessing steps on the dataset. First, the feature selection was applied using the SelectKBest method from scikit-learn's feature selection module. This method utilizes the ANOVA F-value ($f_classif$) to evaluate the importance of features and select the top ' k ' features. In this case, ' k ' was set to 6, indicating that the algorithm will select the six most relevant features for further analysis. The selected features were then obtained by calling `fit_transform()` on the data matrix ' X ' and the corresponding target ' y '. Next, to handle the target variable ' y ', the authors used Label Encoder from scikit-learn's preprocessing module to convert categorical class labels into numerical representations. This step is essential for certain machine learning algorithms that require numerical inputs. After converting class labels to numerical representations, the authors applied one-hot encoding. Moving on to feature scaling, the authors employed MinMaxScaler from scikit-learn's preprocessing module. Feature scaling is essential to ensure that all features are on the same scale, which helps prevent certain features from dominating the learning process during model training. In this code, the numerical features in ' X ' were scaled between 0 and 1 using Min-Max scaling, making the dataset suitable for various machine learning algorithms.

6.1.2. Learning phase

The model is designed as a Sequential stack of layers, featuring dense layers with ReLU activation for hidden layers and SoftMax for the output layer. Batch normalization layers are added for training stability, and dropout layers are utilized for regularization and overfitting prevention. The model is compiled with categorical cross-entropy loss and accuracy metrics for multi-class classification. During training, three callbacks are employed: model checkpoint saves the best model weights based on validation loss, early stopping halts training if validation loss stagnates, and reduce LR on Plateau adjusts the learning rate when validation loss plateaus. The model is trained for 300 epochs using the Adam optimizer with a learning rate of $1e-4$. After thorough testing, it was found that the models had reached their optimal validation results.

6.1.3. Evaluation of DNN model

To assess its effectiveness, the DNN model was evaluated on the test set using the five performance indicators. A detailed comparison of these results is provided in Table 5. As shown, the model utilizing the Adam optimizer achieved an impressive accuracy of 95%.

Figure 4 presents the provided confusion matrix as a 4x4 matrix that evaluates the performance of a classifier in a multi-class classification problem with four classes: 'NoGas,' 'Perfume,' 'Smoke,' and 'Mixture.'

The diagonal elements (1, 1), (2, 2), (3, 3), and (4, 4) represent the number of instances correctly classified for each respective class. Precisely, 185 instances of 'NoGas,' 133 instances of 'Perfume,' 145 instances of 'Smoke,' and 148 instances of 'Mixture' were correctly predicted.

The off-diagonal elements (2,3) and (3,2) represent misclassifications between the 'Perfume' and 'Smoke' classes, where 10 instances of 'Perfume' were incorrectly classified as 'Smoke,' and 19 instances of 'Smoke' were incorrectly classified as 'Perfume.' However, no misclassifications were observed between the other classes.

Table 5. The performance metrics (precision, recall, and F1-score) for the optimized DNN (Training Accuracy: 0.97, Testing Accuracy: 0.95) after Adam optimizer

Class	Precision	Recall	F-Score
NoGas	1.00	1.00	1.00
Perfume	0.88	0.93	0.90
Smoke	0.94	0.88	0.91
Mixture	1.00	1.00	1.00

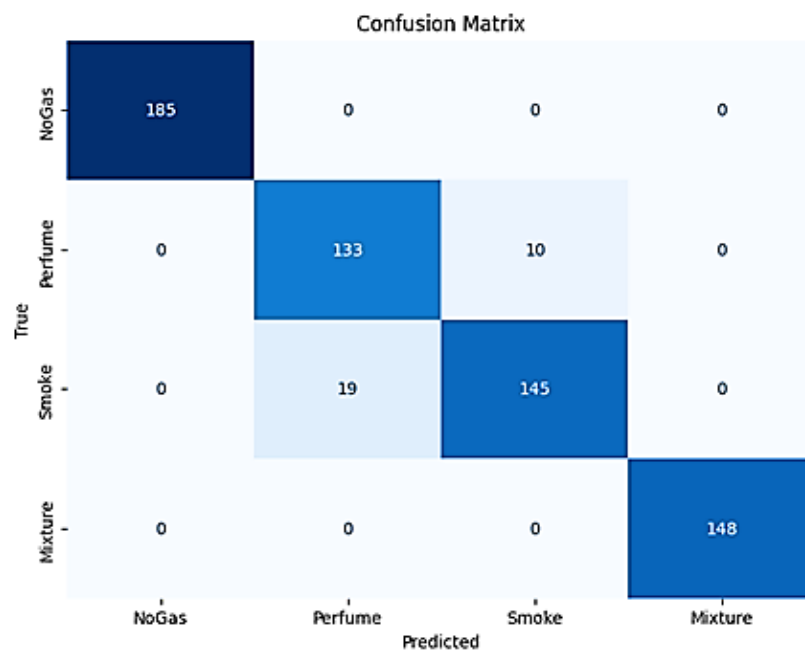


Figure 4. The confusion matrix of the DNN model

6.2. Thermal images results of scenario II

The results obtained from this scenario are presented and analyzed in three structured phases. These include the data augmentation process, the optimization of the model, and the evaluation of the optimized VGG16 model. Each phase plays a critical role in enhancing the model's performance and ensuring the reliability of the detection system.

6.2.1. Data augmentation

The image data generator function from the Keras library is utilized to implement data augmentation and to resize and rescale the samples. Table 6 provides a summary of the data augmentation techniques used, along with their corresponding values. Additionally, Figure 5 presents visual examples of the augmentation process. It includes two parts: Figure 5(a) in which the original image of a common rust-affected sample. Figure 5(b), augmented versions of the same image generated using the specified data augmentation techniques, demonstrating the diverse transformations applied. These data augmentation techniques are applied during the training process to enhance the training dataset's diversity and size, which helps improve the generalization capability of the deep learning model.

Table 6. Data augmentation methods and their associated parameters

Data augmentation method	Associated parameter
--------------------------	----------------------

Rescale	1./255
Rotation range	30 degrees
Height shift range	0.2
Width shift range	0.2
Shear range	0.1
Horizontal flip	True
Zoom range	0.3
Fill mode	'nearest'

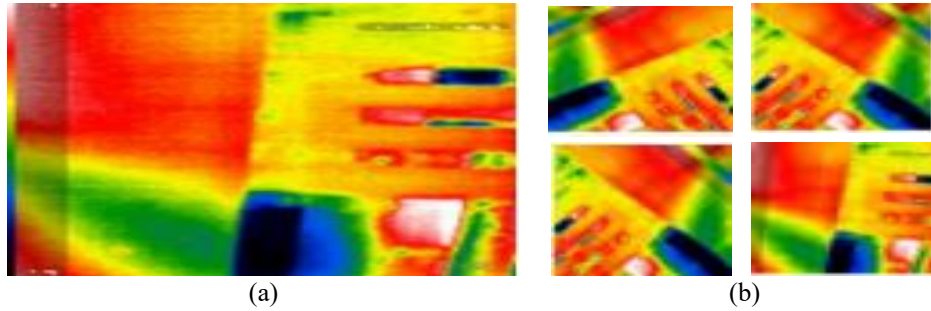


Figure 5. Examples of the augmentation process (a) original common rust image and (b) augmented images using data augmentation techniques

6.2.2. Optimization phase

In the optimization phase, hyperparameter tuning was performed to enhance the performance of the proposed model. Hyperparameter optimization is crucial in fine-tuning the model's architecture and parameters, enabling us to achieve better accuracy and generalization. The parameters of the PSO algorithm for each pre-trained model are configured based on Table 7.

Table 7. The parameter settings employed in the PSO algorithm

Parameter	Value
Swarm size	400
Iteration count	50

The PSO algorithm aims to find the best number of neurons in the hidden layer of the model that minimizes the validation loss. The PSO consists of a swarm of particles, each representing a candidate solution (a possible number of neurons). The PSO algorithm iterates over a fixed number of iterations. Each particle's velocity and position are updated in each iteration based on the cognitive and social components. For each PSO iteration, the model is trained using the number of neurons specified by the particle for a total of six epochs, and the validation loss is used to compute the particle's fitness. Once PSO terminates, the best hyperparameters, including optimal values for parameters like w , c_1 , c_2 , and the best number of neurons, are presented in Table 8, providing insights into the optimized VGG16 model achieved by PSO.

Table 8. Hyper-parameter values used in the PSO algorithm

Pre-trained model	Batch size	Learning rate
VGG16	32	0.00002 (2e-5)

6.2.3. The evaluation of the optimized VGG16 model

The pre-trained model was evaluated on the test set using the five indicators discussed in section 4.1.3. This evaluation aimed to assess the effectiveness of the optimization process. The results presented in Table 9 indicate that the optimized VGG16 model achieved an accuracy of 96%.

Figure 6 displays the confusion matrix obtained from the optimized VGG16 model. Class 0: There are 160 samples of NoGas, and all of them are correctly classified into class 0. There are no misclassifications into other classes; perfume: There are 146 samples of the Perfume class, and all of them are correctly classified into the Perfume class. However, there are 14 misclassifications in the Smoke class. Smoke class: There are 150 samples of the Smoke class, and all are correctly classified into the Smoke class.

There are no misclassifications into other classes. Mixture class: There are 158 samples of the Smoke class, and all are correctly classified into the Mixture class. There are no misclassifications into other classes.

Table 9. The performance metrics (precision, recall, and F1-score) for the fine-tuned VGG16 (training accuracy: 0.99, testing accuracy: 0.96) after PSO optimization

Class	Precision	Recall	F-Score
NoGas	1.00	1.00	1.00
Perfume	0.92	0.91	0.92
Smoke	0.91	0.94	0.93
Mixture	1.00	0.99	0.99

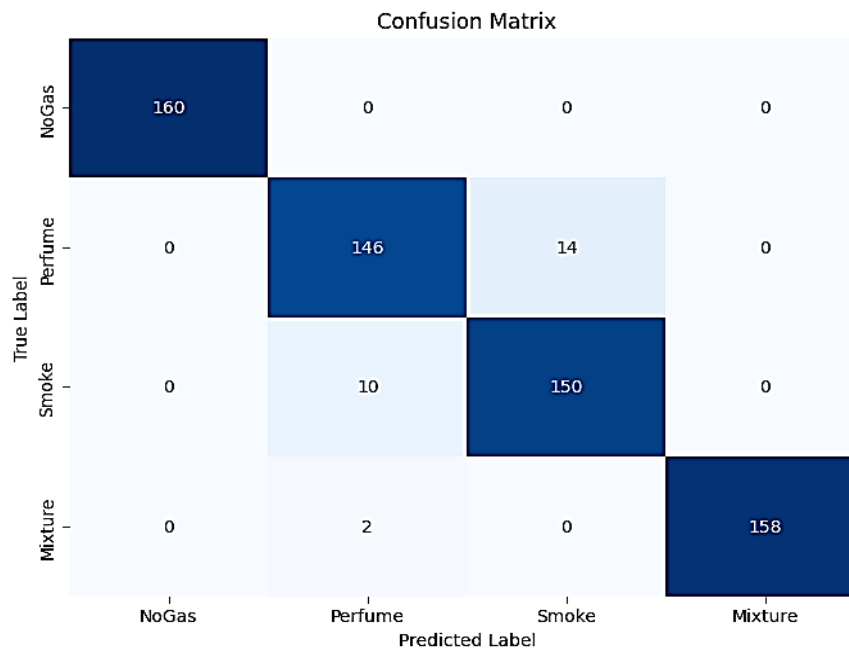


Figure 6. The confusion matrix of the optimized VGG16

6.3. Outcomes of late fusion in scenario III

This scenario shows the outcomes of the late fusion between the DNN model and the VGG16 model. Late fusion is a powerful technique used to combine the predictions of multiple models and harness their complementary strengths. In this study, the synergistic effect of combining the DNN model and the VGG16 model to enhance the overall classification performance is explored. The late fusion process involves aggregating the individual prediction probabilities from both models and creating an ensemble prediction. By leveraging the distinctive features learned by each model, the main aim is to achieve improved accuracy and robustness in our classification task. The experimental findings demonstrate that the late fusion approach significantly enhances the overall performance compared to using each model individually, as shown in Table 10. The fusion model's ability to capture diverse patterns and representations from different architectures contributes to better generalization and discriminative capabilities across the other classes in the problem domain accuracy obtained from the late fusion technique.

Table 10. The performance metrics (precision, recall, and F1-score) for the late fusion model (training accuracy: 0.997, testing accuracy: 0.99)

Class	Precision	Recall	F-Score
NoGas	1.00	1.00	1.00
Perfume	0.97	0.98	0.97
Smoke	0.98	0.98	0.98
Mixture	1.00	1.00	1.00

Figure 7 displays the confusion matrix and provides a detailed analysis of the model's performance for each class. In the dataset, the first row corresponds to instances belonging to the "NoGas" class, with 185

cases correctly classified as “NoGas” (true positives) and no instances wrongly predicted as other classes (false positives). The second row represents “Perfume” class instances, with 139 true positives and 4 false positives, misclassified as “Smoke.” Similarly, the third row corresponds to “Smoke” class instances, with 161 true positives and 3 false positives, misclassified as “Perfume.” Finally, the fourth row corresponds to “Mixture” class instances, with 148 true positives and no false positives, demonstrating accurate predictions.

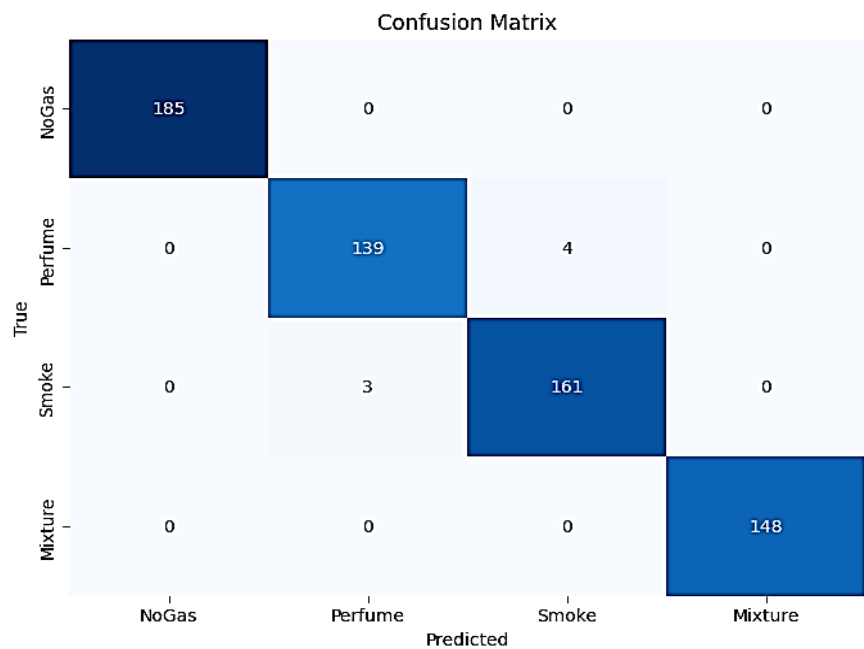


Figure 7. The confusion matrix of late fusion model

Figure 8 and Table 11 show that the DNN model achieved 94.9% accuracy, the optimized VGG16 model reached 95.9%, and the late fusion model outperformed both with 99.4% accuracy. These results highlight the superior performance of the late fusion model in gas leakage detection, demonstrating its robustness and efficiency for smart cities and industrial applications. The combination of IoT data and deep learning enhances detection accuracy, making the approach a valuable tool for safety and environmental protection in various scenarios.

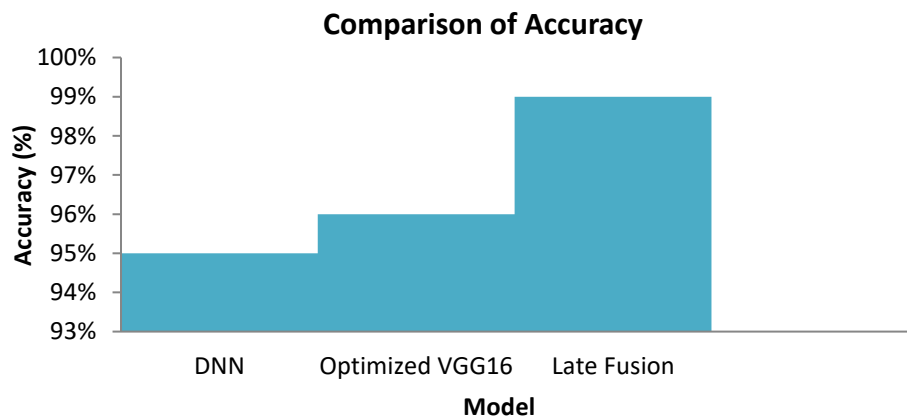


Figure 8. The comparison of accuracy among different models

Table 11. The overall accuracy of both the individual models and the fused model proposed in this study

Model	Accuracy (%)
-------	--------------

DNN	94.9
Optimized VGG16	95.9
Late fusion model	99.4

6.4. Comparison

To show the proposed approach's competitive capabilities. Table 12 displays the outcomes of this comparison. The table showcases the performance results for each fusion technique. The early fusion approach achieves an accuracy of 0.960. For the intermediate fusion and multitask fusion approaches, only accuracy is reported (0.945 and 0.969, respectively). Finally, the proposed DNN+ optimized VGG16 late fusion outperforms the other methods, achieving a high accuracy of 0.99. These results demonstrate the effectiveness of the proposed late fusion technique in significantly improving the overall performance of the gas leakage detection system.

Table 12. A comparative analysis of the proposed approach with previous relevant studies using the same multimodal dataset

Fusion approach	Accuracy	Sensitivity	Precision	F1-score
Early fusion [32]	0.960	0.963	0.963	0.963
Intermediate fusion [33]	0.945	-	-	-
Multitask fusion [33]	0.969	-	-	-
The proposed Thermal-Gas Fusion detection model	0.99	0.99	0.99	0.99

7. CONCLUSION AND FUTURE WORK

This paper presents an innovative approach to assess the reliability of intelligent multimodal data, including IoT-based data, for gas leakage detection. The late fusion method was evaluated against individual data modalities using pre-trained CNNs (VGG16) and a DNN model with gas sensor readings and thermal images. The DNN achieved 94.9% accuracy, while the optimized VGG16 reached 95.9%. However, the late fusion model excelled with 99.4% accuracy, highlighting the effectiveness of IoT-driven data fusion. To optimize CNN hyperparameters, the PSO algorithm was employed, outperforming traditional manual methods. The proposed multimodal IoT-based fusion method significantly enhanced gas detection accuracy, proving superior to single-sensor techniques and offering a reliable solution for industrial applications. However, the system requires substantial, diverse data samples during training to ensure network robustness. Furthermore, to achieve efficient and streamlined operations, a dedicated hardware processing module is essential. In the future, our emphasis will be on the accumulation of datasets containing a multitude of gases and their various combinations across a wide range of environmental conditions, harnessing the potential of data generated by the IoT. By doing so, we aim to enhance the system's performance further and broaden its applicability across a broader range of scenarios.




REFERENCES

- [1] P. Chen *et al.*, "Effectively detecting operational anomalies in large-scale IoT data infrastructures by using a GAN-based predictive model," *Computer Journal*, vol. 65, no. 11, pp. 2909–2925, Oct. 2022, doi: 10.1093/comjnl/bxao085.
- [2] A. S. P. Pamula, A. Ravilla, and S. V. H. Madiraju, "Applications of the internet of things (IoT) in real-time monitoring of contaminants in the air, water, and soil †," in *Engineering Proceedings*, Nov. 2022, vol. 27, no. 1, p. 26, doi: 10.3390/ecs-a9-13335.
- [3] J. Oh *et al.*, "Machine learning-assisted gas-specific fingerprint detection/classification strategy based on mutually interactive features of semiconductor gas sensor arrays," *Electronics (Switzerland)*, vol. 11, no. 23, p. 3884, Nov. 2022, doi: 10.3390/electronics11233884.
- [4] Y. Zhou, X. Zhao, J. Zhao, and D. Chen, "Research on fire and explosion accidents of oil depots," *Chemical Engineering Transactions*, vol. 51, pp. 163–168, 2016, doi: 10.3303/CET1651028.
- [5] O. Attallah, "Cercan-Net: Cervical cancer classification model via multi-layer feature ensembles of lightweight CNNs and transfer learning," *Expert Systems with Applications*, vol. 229, p. 120624, Nov. 2023, doi: 10.1016/j.eswa.2023.120624.
- [6] O. Attallah, "MB-AI-His: Histopathological diagnosis of pediatric medulloblastoma and its subtypes via AI," *Diagnostics*, vol. 11, no. 2, p. 359, Feb. 2021, doi: 10.3390/diagnostics11020359.
- [7] O. Attallah, "RADIC: A tool for diagnosing COVID-19 from chest CT and X-ray scans using deep learning and quad-radiomics," *Chemometrics and Intelligent Laboratory Systems*, vol. 233, p. 104750, Feb. 2023, doi: 10.1016/j.chemolab.2022.104750.
- [8] L. T. Liu, S. Wang, T. Britton, and R. Abebe, "Reimagining the machine learning life cycle to improve educational outcomes of students," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 120, no. 9, Feb. 2023, doi: 10.1073/pnas.2204781120.
- [9] O. Attallah and A. Samir, "A wavelet-based deep learning pipeline for efficient COVID-19 diagnosis via CT slices," *Applied Soft Computing*, vol. 128, p. 109401, Oct. 2022, doi: 10.1016/j.asoc.2022.109401.
- [10] S. A. Zamani and Y. Baleghi, "Early/late fusion structures with optimized feature selection for weed detection using visible and thermal images of paddy fields," *Precision Agriculture*, vol. 24, no. 2, pp. 482–510, Aug. 2023, doi: 10.1007/s11119-022-09954-8.
- [11] S. Venkatasubramanian *et al.*, "Fault diagnosis using data fusion with ensemble deep learning technique in IIoT," *Mathematical*




- Problems in Engineering*, vol. 2022, pp. 1–8, Jun. 2022, doi: 10.1155/2022/1682874.
- [12] J. Praveenchandar *et al.*, “IoT-based harmful toxic gases monitoring and fault detection on the sensor dataset using deep learning techniques,” *Scientific Programming*, vol. 2022, pp. 1–11, Aug. 2022, doi: 10.1155/2022/7516328.
 - [13] P. Peng, X. Zhao, X. Pan, and W. Ye, “Gas classification using deep convolutional neural networks,” *Sensors (Switzerland)*, vol. 18, no. 1, p. 157, Jan. 2018, doi: 10.3390/s18010157.
 - [14] C. Spandonidis, P. Theodoropoulos, F. Giannopoulos, N. Galiatsatos, and A. Petsa, “Evaluation of deep learning approaches for oil & gas pipeline leak detection using wireless sensor networks,” *Engineering Applications of Artificial Intelligence*, vol. 113, p. 104890, Aug. 2022, doi: 10.1016/j.engappai.2022.104890.
 - [15] X. Pan, H. Zhang, W. Ye, A. Bermak, and X. Zhao, “A fast and robust gas recognition algorithm based on hybrid convolutional and recurrent neural network,” *IEEE Access*, vol. 7, pp. 100954–100963, 2019, doi: 10.1109/ACCESS.2019.2930804.
 - [16] A. Kopbayev, F. Khan, M. Yang, and S. Z. Halim, “Gas leakage detection using spatial and temporal neural network model,” *Process Safety and Environmental Protection*, vol. 160, pp. 968–975, Apr. 2022, doi: 10.1016/j.psep.2022.03.002.
 - [17] S. Marathe, “Leveraging drone based imaging technology for pipeline and RoU monitoring survey,” in *Society of Petroleum Engineers - SPE Symposium: Asia Pacific Health, Safety, Security, Environment and Social Responsibility 2019*, Apr. 2019, doi: 10.2118/195427-ms.
 - [18] R. Zhou, H. Su, and Z. Wen, “Experimental study on leakage detection of grassed earth dam by passive infrared thermography,” *NDT and E International*, vol. 126, p. 102583, Mar. 2022, doi: 10.1016/j.ndteint.2021.102583.
 - [19] J. Bin, C. A. Rahman, S. Rogers, and Z. Liu, “Tensor-based approach for liquefied natural gas leakage detection from surveillance thermal cameras: A feasibility study in rural areas,” *IEEE Transactions on Industrial Informatics*, vol. 17, no. 12, pp. 8122–8130, Dec. 2021, doi: 10.1109/TII.2021.3064845.
 - [20] B. Liu, H. Ma, X. Zheng, L. Peng, and A. Xiao, “Monitoring and detection of combustible gas leakage by using infrared imaging,” in *IST 2018 - IEEE International Conference on Imaging Systems and Techniques, Proceedings*, Oct. 2018, pp. 1–6, doi: 10.1109/IST.2018.8577102.
 - [21] V. Krivetskiy *et al.*, “Selective detection of individual gases and CO/H₂ mixture at low concentrations in air by single semiconductor metal oxide sensors working in dynamic temperature mode,” *Sensors and Actuators, B: Chemical*, vol. 254, pp. 502–513, Jan. 2018, doi: 10.1016/j.snb.2017.07.100.
 - [22] N. Lagaros, D. Theodorou, and V. Riziotis, “Smart gas sensors deep learning for the identification and classification of various gaseous species by sensors,” National Technical University of Athens (NTUA), 2023, doi: 10.26240/heal.ntua.21511.
 - [23] A. Amkor and N. El Barbri, “Classification of potatoes according to their cultivated field by SVM and KNN approaches using an electronic nose,” *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 3, pp. 1471–1477, Jun. 2023, doi: 10.11591/eei.v12i3.5116.
 - [24] S. Nikolopoulos, I. Kalogeris, and V. Papadopoulos, “Non-intrusive surrogate modeling for parametrized time-dependent PDEs using convolutional autoencoders,” *arXiv:2101.05555*, Apr. 2021, doi: 10.48550/arXiv.2101.05555.
 - [25] F. Bhatti, M. A. Shah, C. Maple, and S. Ul Islam, “A novel internet of things-enabled accident detection and reporting system for smart city environments,” *Sensors (Switzerland)*, vol. 19, no. 9, p. 2071, May 2019, doi: 10.3390/s19092071.
 - [26] D. Lahat, T. Adali, and C. Jutten, “Multimodal data fusion: An overview of methods, challenges, and prospects,” *Proceedings of the IEEE*, vol. 103, no. 9, pp. 1449–1477, Sep. 2015, doi: 10.1109/JPROC.2015.2460697.
 - [27] P. Malhotra, S. Gupta, D. Koundal, A. Zaguia, and W. Enbeyle, “Deep neural networks for medical image segmentation,” *Journal of Healthcare Engineering*, vol. 2022, pp. 1–15, Mar. 2022, doi: 10.1155/2022/9580991.
 - [28] W. L. Hakim *et al.*, “Convolutional neural network (CNN) with metaheuristic optimization algorithms for landslide susceptibility mapping in Icheon, South Korea,” *Journal of Environmental Management*, vol. 305, p. 114367, Mar. 2022, doi: 10.1016/j.jenvman.2021.114367.
 - [29] S. Kumaresan, K. S. J. Aultrin, S. S. Kumar, and M. D. Anand, “Deep learning-based weld defect classification using VGG16 transfer learning adaptive fine-tuning,” *International Journal on Interactive Design and Manufacturing*, vol. 17, no. 6, pp. 2999–3010, May 2023, doi: 10.1007/s12008-023-01327-3.
 - [30] T. Y. Kim and S. B. Cho, “Particle swarm optimization-based CNN-LSTM networks for forecasting energy consumption,” in *2019 IEEE Congress on Evolutionary Computation*, 2019, pp. 1510–1516, doi: 10.1109/CEC.2019.8789968.
 - [31] S. Kim, E. Choi, Y. K. Jun, and S. Lee, “Student dropout prediction for university with high precision and recall,” *Applied Sciences (Switzerland)*, vol. 13, no. 10, p. 6275, May 2023, doi: 10.3390/app13106275.
 - [32] P. Narkhede, R. Walambe, P. Chandel, S. Mandaokar, and K. Kotecha, “MultimodalGasData: Multimodal dataset for gas detection and classification,” *Data*, vol. 7, no. 8, p. 112, Aug. 2022, doi: 10.3390/data7080112.
 - [33] P. Narkhede, R. Walambe, S. Mandaokar, P. Chandel, K. Kotecha, and G. Ghinea, “Gas detection and identification using multimodal artificial intelligence based sensor fusion,” *Applied System Innovation*, vol. 4, no. 1, pp. 1–14, Jan. 2021, doi: 10.3390/asi4010003.

BIOGRAPHIES OF AUTHORS






Mariam M. Abdellatif    received the B.Sc. degree in science from Al-Azhar University in 2020 and obtained her M.Sc. degree from Al-Azhar University in 2024. She is a lecturer assistant at ElSewedy University of Technology, Computer Engineering. She has published several research papers in the fields of computer science and artificial intelligence. She can be contacted at email: mariemmohammed974@gmail.com.






Mehmet Akif Çifci    is an associate professor currently affiliated with Klaipėda University. Dr. Cifci is a distinguished researcher specializing in artificial intelligence, machine learning, and natural language processing. With extensive academic experience, he has contributed to multiple international institutions, including universities in Lithuania, Turkey, and previously at TU Wien. His research focuses on cutting-edge AI technologies, with notable work in explainable AI, particularly in healthcare applications. Dr. Cifci holds dual bachelor's degrees in English language teaching and computer engineering, and advanced degrees in computer science from Istanbul Aydin University. A prolific researcher and academic, he has co-authored significant publications and actively contributes to the global data science community through platforms like Kaggle. He is currently involved in the EU Horizon 2020 FET-OPEN project “ECOLOPES – Ecological Building Envelopes” (Grant Agreement No. 964414), which integrates ecological principles into building design, and participates in the TÜBİTAK 2553 – Pakistan Science Foundation (PSF) Bilateral Cooperation Program. She can be contacted at email: akif.cifci@tuwien.ac.at






Asmaa A. Ibrahim    received the B.Sc. degree in science in 2009, and the M.Sc. and Ph.D. degrees in computer science in 2017 and 2021, respectively. She is currently a lecturer in computer science with the Mathematics Department, Faculty of Science, Al-Azhar University, Cairo, Egypt. She has published several research papers in the fields of AI, machine learning, software engineering, and internet of things. She can be contacted at email: asmaaabdemoniemibrahim1174.el@azhar.edu.eg



Hany M. Harb    received the B.Sc. in computers and control engineering from the Faculty of Engineering, Ain Shams University, Cairo, Egypt, in 1978; the M.Sc. degree in computers and systems engineering from the Faculty of Engineering, Al-Azhar University, Cairo, Egypt, in 1981; the M.Sc. degree in operations research (MSOR) from IIT in 1987; and the Ph.D. degree in computer science from Illinois Institute of Technology (IIT), Chicago, Illinois, USA, in 1986. He became a professor in 1997. He has published a number of research papers in the fields of artificial intelligence, information systems (business informatics), and information science. He supervises several master's and doctoral theses and is a reviewer for many local and international conferences and journals. She can be contacted at email: harbhany@yahoo.com.



Abeer S. Desuky    received the B.Sc. degree in science in 2003 and the M.Sc. and Ph.D. degrees in computer science in 2008 and 2012, respectively. She became an associate professor in 2017 and is currently a Professor of Computer Science and the head of the Mathematics Department, Faculty of Science, Al-Azhar University, Cairo, Egypt. She supervises several master's and Ph.D. theses. She has published numerous research articles in the fields of AI, machine learning, metaheuristic optimization, and data mining and analysis. She is a reviewer for many Scopus-indexed journals, such as IEEE Access, Egyptian Informatics Journal, and PeerJ. She can be contacted at email: abeerdesuky@azhar.edu.eg.