Mathematical Modeling and Algorithm Application
ISSN: 3006-0842 | Vol. 8, No. 1, 2026

Research on an Early Detection Method for Residential Indoor Fires Based on the LSTM-FNN Algorithm

Hongyang Li*, Jun Li

Sichuan University of Science & Engineering, Yibin, China

* **Corresponding author**: Hongyang Li

Abstract: To address the issues of delayed response and high false alarm/miss rates in traditional residential indoor fire detection methods, an LSTM-FNN model integrating Long Short-Term Memory (LSTM) and Fuzzy Neural Network (FNN) was developed to enable early identification of smoldering and flaming fire stages. The model utilizes the LSTM module to extract temporal dynamic features of smoke, carbon monoxide concentration, and temperature. It combines the FNN module to handle feature uncertainty and incorporates an attention mechanism for feature selection. Standard fire scenarios were constructed using Pyrosim software to generate training data. A physical experimental platform was established to comprehensively validate the model's early warning capability and generalization performance. Results demonstrate that the LSTM-FNN model achieves remarkably low Mean Squared Error (MSE) values of 0.0009, 0.0002, and 0.0002 during no-fire, smoldering, and flaming phases, respectively. Its Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) metrics significantly outperform standalone models. Validation experiments on typical combustibles—beech wood, cotton rope, polyurethane, and n-heptane—confirm its effective capture of early-stage fire characteristics. This method offers superior early warning timeliness compared to traditional approaches, significantly reducing false alarm and missed detection rates while enhancing the timeliness and reliability of early fire detection.

Keywords: Indoor fire; LSTM-FNN; Early detection; Information fusion; Temporal features

1. Introduction

As urbanization continues to advance, while enhancing building functionality and living convenience, the increasing complexity of internal structures and environments has significantly amplified the safety risks associated with indoor fires. Once an indoor fire occurs, the high-temperature smoke it generates rapidly spreads throughout the space and is difficult to evacuate, posing a high risk of severe casualties and property damage [1.2]. Current fire detection methods primarily rely on fixed-threshold monitoring of single physical parameters such as temperature, smoke concentration, or specific gas levels [3]. These approaches frequently encounter issues like false alarms, missed detections, and delayed responses during practical application. With the rapid advancement of intelligent computing technologies, vision-based fire detection methods offer a novel technical pathway for fire early warning [4]. However, in indoor environments, this approach faces inherent limitations due to privacy protection requirements, complex background interference, and fire source obstruction. Particularly during the smoldering phase where visual features are less pronounced, it suffers from high computational complexity and insufficient warning reliability.

To overcome the aforementioned technical bottlenecks, multi-sensor data fusion methods have gradually emerged as a research hotspot. These include D-S evidence theory [5], deep learning [6], and machine learning classification algorithms [7]. Among these, LSTM, with its unique gating mechanism, can effectively capture long-term dependencies in time-series data, demonstrating distinct advantages in extracting temporal features of fire parameters. Liu et al. proposed an ELF-LSTM-based environmental fire detection method [8], while Xu, Zhaoyi et al. proposed integrating deep long short-term memory (LSTM) neural networks with variational autoencoders (VAE) for fire detection [9]. However, LSTM is inherently a deterministic model. While its strengths lie in precisely fitting temporal dynamics, it performs poorly when handling uncertainties and ambiguous information inherent in sensor data, such as noise fluctuations and environmental interference. This makes it prone to misjudgments during fire critical states.

Fuzzy neural networks provide an effective approach for processing uncertain information. Fuzzy logic reasoning converts variables that are difficult to quantify precisely into computable mathematical forms through membership functions, and performs inference using expert knowledge bases or data-driven rules. This effectively identifies and suppresses interference signals caused by daily activities. Park et al. proposed a fuzzy logic recognition method for fire signal patterns in an IoT-based fire detection system [10], while Rao et al. introduced a neural-fuzzy-based design approach for wildland fire detection systems [11]. However, traditional fuzzy logic reasoning exhibits significant limitations in handling complex temporal dynamic characteristics, making it challenging to independently achieve high-precision temporal modeling of the entire fire development process.

How to balance the extraction of subtle early-stage temporal features of fires with precise reasoning about uncertain information, thereby constructing an early-stage fire monitoring model suitable for indoor environments, remains a significant challenge in current research. To address these issues, this study leverages the complementary strengths of LSTM in temporal modeling and fuzzy logic reasoning in uncertainty inference. We propose an indoor fire early detection method based on an LSTM-FNN fusion algorithm. By constructing a deep LSTM-FNN fusion model, training it with multi-source fire data obtained from Pyrosim simulations, and validating it through an on-site experimental

platform, we achieve precise early detection of indoor fires.

2. Long Short-Term Memory Networks and Fuzzy Neural Networks

2.1. Long Short-Term Memory Network

Long Short-Term Memory (LSTM) networks represent a significant variant of recurrent neural networks (RNNs). By incorporating gating mechanisms, they effectively address the vanishing and exploding gradient issues that commonly arise in traditional RNNs during training on long sequences. The core of LSTM lies in regulating the flow of information within the cell state through the forget gate, input gate, and output gate, thereby enabling the processing of sequential information with long-term dependencies. The basic unit structure of an LSTM is illustrated in Figure 1.

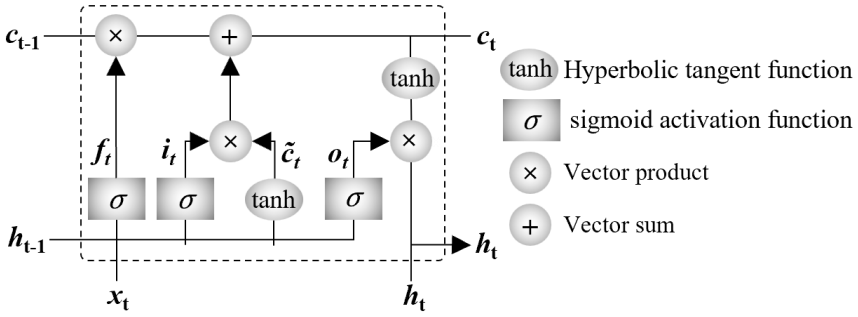


Figure 1. Basic structure of an LSTM network unit

At time step t, the computation process of the LSTM unit is as follows:

$$f_t = \sigma\left(w_f \cdot \left[h_{t-1}, x_t\right] + b_f\right) \quad (1)$$

$$i_t = \sigma\left(w_i \cdot \left[h_{t-1}, x_t\right] + b_i\right) \quad (2)$$

$$\tilde{c}_t = \tanh\left(w_c \cdot \left[h_{t-1}, x_t\right] + b_c\right) \quad (3)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \quad (4)$$

$$o_t = \sigma\left(w_o \cdot \left[h_{t-1}, x_t\right] + b_o\right) \quad (5)$$

$$h_t = o_t \odot \tanh\left(c_t\right) \quad (6)$$

In the equation, ft denotes the forget gate, determining the information discarded from the previous cell state ct-1. ht-1 represents the hidden state at the previous time step, xt denotes the input at the current time step, and [ht-1, xt] indicates vector concatenation. wf and bf are the forget gate weight matrix and bias term, respectively. it denotes the input gate, controlling the update magnitude of the current candidate value c̃t to the cell state. wi and bi represent the input gate weight matrix and bias term, respectively. wc and bc denote the weight matrix and bias term for the candidate cell state, respectively. ct indicates the cell update state. ⊙ denotes the Hadamard product. ot represents the output gate, with wo and bo being the output gate weight matrix and bias term, respectively. ht denotes the current hidden state.

2.2. Fuzzy Neural Network

Fuzzy neural networks integrate fuzzy logic systems with artificial neural networks. Through adaptive learning, they generate and optimize fuzzy rules and membership function parameters, significantly reducing reliance on prior knowledge. They excel at handling imprecise information and generalizing to unknown inputs. The structure of a fuzzy neural network is shown in Figure 2.

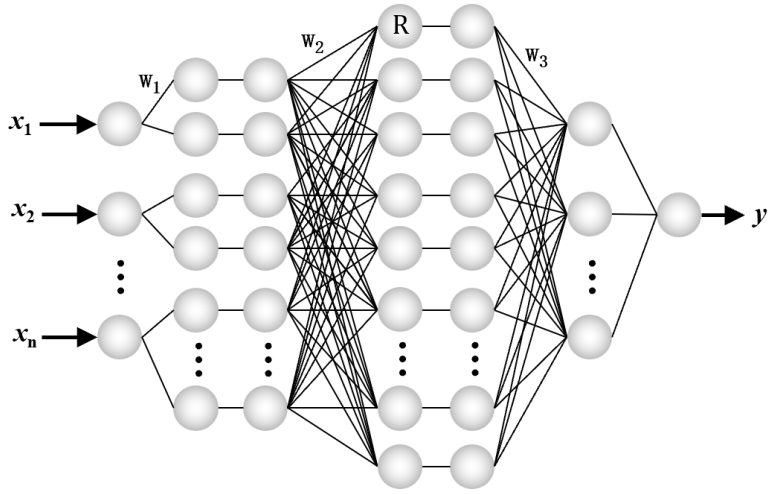


Figure 2. Fuzzy Neural Network Structure

Input Layer: Responsible for buffering input signals. Assuming n nodes, each node directly passes its input to the next layer. The output expression is:

$$o_i^{(1)} = x_i,\ i = 1, 2, ..., n \tag{7}$$

In the formula, xi represents the ith input variable, and n denotes the number of variable outputs.

Linear Connection Layer: Applies linear transformations to input signals through weight matrices, performing linear combinations to enhance the model's expressive capability. The output expression is:

$$o_i^{(2)} = x_i \cdot w_1 \tag{8}$$

In the equation, w_1 denotes the linear connection weight matrix for that layer.

Fuzzyfying Layer: Each input node corresponds to m membership functions, so this layer should have m*n nodes. It uses membership functions to fuzzyfy input signals. Common membership functions include Gaussian, triangular, trapezoidal, etc. Taking the Gaussian function as an example, the output expression for the jth fuzzy set of the ith input variable is:

$$o_i^{(3)} = u_i^j(x_i) = \exp\left(\frac{\left(x_i - c_{ij}\right)^2}{\sigma_{ij}^2}\right) \tag{9}$$

Among these, cij is the mean of the Gaussian function, and σij is the standard deviation.

Fuzzy Rule Layer: Responsible for implementing the antecedent matching and operations of fuzzy rules. Each node represents a fuzzy rule, and the activation strength of each rule is calculated by multiplying the membership degrees of its antecedents. The output expression is:

$$o_k^{(4)} = a_k = \prod u_i^j \cdot w_2, k = 1, 2, ..., R \tag{10}$$

In the formula, ak represents the activation intensity of the kth rule, w_2 denotes the connection weight of each rule, and R is the total number of rules.

Normalization Layer: Normalizes the activation intensities of all output rules to ensure their sum equals 1, preserving the relative importance among different rules. This provides standardized weights for subsequent fuzzy inference. The output expression is:

$$o_k^{(5)} = \beta_k = \frac{a_k}{\sum_{i=1}^{k} a_i} \tag{11}$$

De-blurring Layer: Responsible for weighting and combining normalized rule weights with the post-conditions of each rule to generate the network's final output. The output expression is:

$$o_y^{(6)} = \sum_{k=1}^{R} \beta_k \cdot w_3 \tag{12}$$

In the formula, w_3 denotes the normalized activation intensity matrix weight for each rule.

Output Layer: Responsible for integrating the state probability distributions generated by fuzzy inference into a single probability indicator, typically calculated using the area-of-heavyweight method. The output expression is:

$$y = o_y^{(7)} = \frac{\sum c_y \cdot o_y^{(6)}}{o_y^{(6)}} \tag{13}$$

In the equation, y represents the final quantified output probability, and cy denotes the value represented by each output state.

3. Establishment of a Fire Detection Method Based on LSTM-FNN

The development of a fire involves multiple chemical reaction products, primarily manifested in the early stages as smoke particulate matter, carbon monoxide generation, and temperature changes. To comprehensively characterize the dynamic properties of a fire, this study selected smoke concentration, carbon monoxide concentration, and temperature as the fundamental feature signals for fire identification. Leveraging the complementary advantages of long short-term memory networks and fuzzy neural networks, an LSTM-FNN hybrid model was constructed (Figure 3). Compared to single FNN detection methods, LSTM effectively captures the dynamic characteristics of multi-parameter temporal evolution. This enables more accurate representation of the continuous variation patterns during the early fire development process, rather than treating sensor data as isolated instantaneous observations.

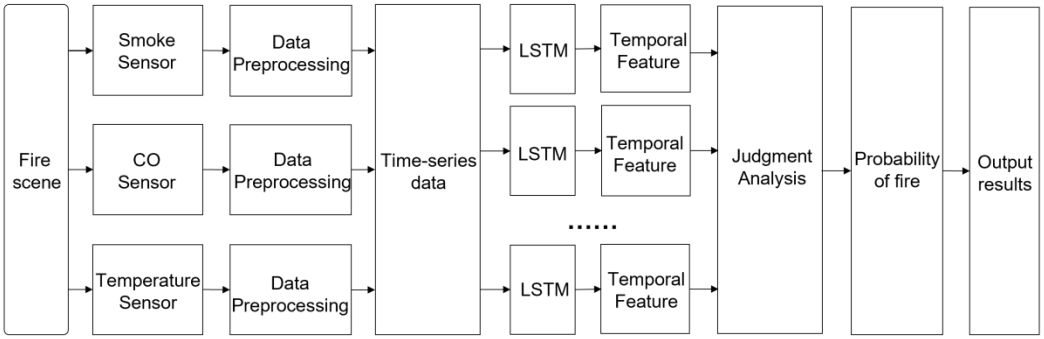


Figure 3. LSTM-FNN hybrid network structure

The specific implementation process is as follows:

Step 1: Data Preparation and Preprocessing: Collect carbon monoxide concentration, smoke concentration, and temperature data from the environmental monitoring system. Organize the data into time series and construct a sample set using a 10-second time window. Divide the complete dataset into training and testing sets in chronological order, ensuring temporal continuity. Input data undergoes normalization, mapping sensor readings to a unified numerical range to eliminate the impact of dimensional and magnitude

differences on model training.

Step 2: LSTM Feature Extraction Module Construction: Build an LSTM neural network to extract high-level feature representations from raw time-series data. Set the input dimension to 10 time steps and 3 sensor channels. Configure the LSTM layer with 64 hidden units to capture temporal dependencies. This is followed by a fully connected layer with 32 neurons, compressing the extracted high-level features into a 32-dimensional feature vector as input to the fuzzy neural network.

Step 3: Fuzzy Neural Network Construction: Design the fuzzy neural network architecture. To address potential redundancy in high-dimensional features, an attention mechanism is introduced for feature selection. Tailored to fire detection requirements, three Gaussian fuzzy sets are defined for each input feature, representing "no fire," "small fire," and "open flame" states. Each fuzzy rule's antecedent combines fuzzy sets from input features, while the consequent adopts a linear function of these features. After normalizing activated rules, the normalized rule weights are multiplied by the corresponding antecedent function outputs. The final fire risk score is obtained through weighted summation.

Step 4: Model Parameter Initialization: Initialize all trainable parameters in the hybrid model. The LSTM network's weight matrix is initialized using Glorot's uniform distribution, with biases set to zero. In the fuzzy neural network, the center parameters of Gaussian membership functions for the three fire states are set based on typical feature ranges for each state, while width parameters are determined empirically. Connection weights in the fuzzy rule layer are initialized using a random normal distribution, and post-term network coefficients are initialized using a Glorot uniform distribution.

Step 5: Training Process Configuration: Set model training hyperparameters with a batch size of 32 and an initial learning rate of 0.001. Given that the output represents fire risk scores, the mean squared error is selected as the loss function, and the Adam algorithm is used as the optimizer. The maximum training iterations are set to 100, with continuous monitoring of validation set performance during training to implement an early stopping strategy.

Step 6: Forward Propagation Calculation: In each training batch, the raw time-series data is input into the LSTM network. After time-series feature extraction, a 32-dimensional feature vector is output. This vector is fed into a fuzzy neural network to sequentially compute the membership degrees of each feature belonging to three fire states. The rule activation strength is obtained through multiplication. The activation strength undergoes normalization to derive rule weights, enabling calculation of the output values for each rule's post-condition function. Finally, weighted fusion yields the fire risk prediction result.

Step 7: Backpropagation and Parameter Update: Calculate the mean squared error loss based on the discrepancy between the model's predicted output and the actual fire risk labels. Employ the backpropagation algorithm to compute the gradient of the loss function across all levels of parameters. Utilize the chain rule to propagate the error signal backward layer by layer starting from the output layer, sequentially solving for the gradients of parameters within both the fuzzy neural network and the LSTM feature extraction module. The Adam optimizer updates all weight and bias parameters, including LSTM network parameters, membership function parameters, rule weights, and post-term coefficients.

Step 8: Model Validation and Early Stopping Mechanism: After each training iteration, evaluate model generalization performance using the validation set and compute its mean squared error loss. Continuously monitor the validation set performance trend. If the validation loss fails to decrease for 10 consecutive training epochs, trigger the early stopping mechanism to terminate training. Throughout the training process, retain the model parameters that achieved the best performance on the validation set to prevent overfitting.

Step 9: Model Testing and Performance Evaluation: Load the optimal model parameters from the validation set and conduct a comprehensive evaluation on an independent test set. Quantify the model's prediction performance using regression metrics such as Root Mean Square Error (RMSE) and Mean Absolute Error (MAE). Further analyze the model's discrimination capability across different fire state samples. Combine the confusion matrix to evaluate classification reliability among the three states—"No Fire," "Small Fire," and "Open Fire"—and calculate detection accuracy and false alarm rates for each category. The formulas for Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE) are as follows:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2 \tag{14}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2} \tag{15}$$

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{16}$$

In the equation, ŷi denotes the actual value, yi represents the predicted value, and n is the total number of samples.

Step 10: Fuzzy Rule Extraction and Analysis: Extract interpretable decision rules from the fully trained fuzzy neural network and analyze the parameter distribution characteristics of each membership function across the three fire states. Identify key rules significantly contributing to fire risk assessment. By analyzing rule weights and consequent function coefficients, reveal the influence mechanisms of different input features on state discrimination, thereby enhancing the interpretability and decision transparency of the hybrid model in fire detection tasks. The flowchart is shown in Figure 4.

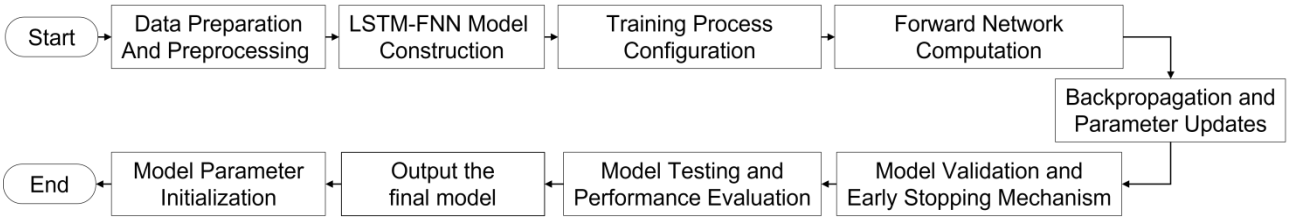


Figure 4. LSTM-FNN hybrid network structure

4. Experimental Results and Analysis

4.1. Data Acquisition

To achieve precise identification of early-stage indoor fire characteristics, model training requires extensive high-quality data covering the entire fire lifecycle and multiple physical fields. While large-scale physical fire experiments can yield authentic data, they inherently face limitations such as high costs, significant risks, and measurement blind spots. Therefore, this study employs Pyrosim fire dynamics simulation software to construct standardized fire scenarios, simulating critical processes including smoke flow, heat transfer, and chemical reactions during fires [12]. The generated simulation data provides a robust training foundation for the model.

The fire scene model was configured according to the combustion laboratory conditions specified in Appendix E of the Chinese National Standard "Point Smoke Fire Detectors" (GB 4715-2024). Its three-dimensional structure is shown in Figure 5. The room dimensions are 10m × 8m × 4m. The monitoring point is positioned at the center of the room ceiling. The fire source occupies a random ground location with an area of 0.25m². The relevant characteristics of the combustible material are detailed in Table 1. The grid size is determined using an empirical formula based on the characteristic flame diameter D*. The fire growth process is described using the t2 fire model [13].

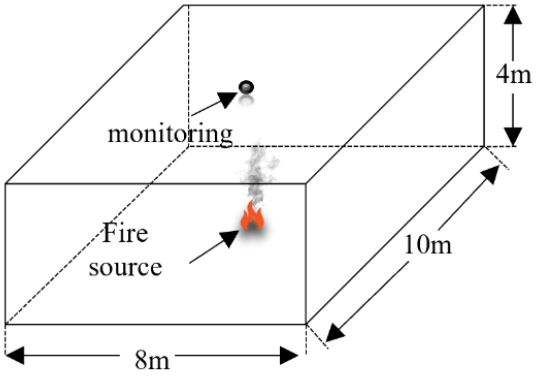


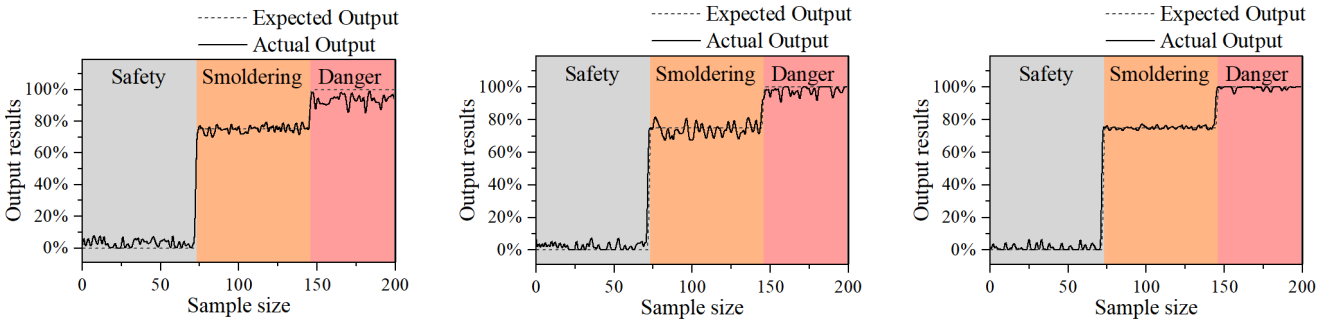
Figure 5. Fire Scene Model

Table 1. Material parameters

Material	Density (kg/m³)	Thermal Conductivity (W/m·K)	Specific Heat Capacity (kJ/kg·K)	Heat of Combustion (MJ/kg)
Beech wood	700	0.15	1.7	15
Cotton rope	400	0.05	1.3	17
Polyurethane	1145	0.04	1.8	23
n-Heptane	680	0.14	2.2	45

4.2. Analysis of Simulation Results

Monitoring points set in the Pyrosim simulation recorded temporal data on smoke, CO, and temperature during 300 seconds of combustion for four types of materials. For each material, 250 sets of temporal data with a step size of 10 were selected. Each set was categorized and labeled as no fire, smoldering, or flaming states. The 1000 data sets were divided into a training set and a test set in an 8:2 ratio. To more intuitively evaluate the model's performance in early fire detection and warning, this study compares the LSTM and FNN standard models with the LSTM-FNN model. The fire prediction results from the three models are shown in Figure 6. The image clearly demonstrates significant differences in recognition accuracy across different fire stages among the models. During the smoldering phase, the LSTM-based model demonstrated superior performance, particularly adept at capturing temporal dynamic features. Its gating mechanism is especially well-suited for identifying the slow-changing patterns characteristic of this stage. while the FNN-based model excels in detecting the flaming stage. Through deep perception and fuzzy reasoning, it effectively extracts key features while eliminating redundant interference. The LSTM-FNN model combines the strengths of both, enabling it to keenly capture early faint signals while accurately identifying the intense nonlinear changes of flaming fires. Consequently, it maintains outstanding performance across all fire stages.



(a) FNN (b) LSTM (c) LSTM-FNN

Figure 6. Fire Prediction Results

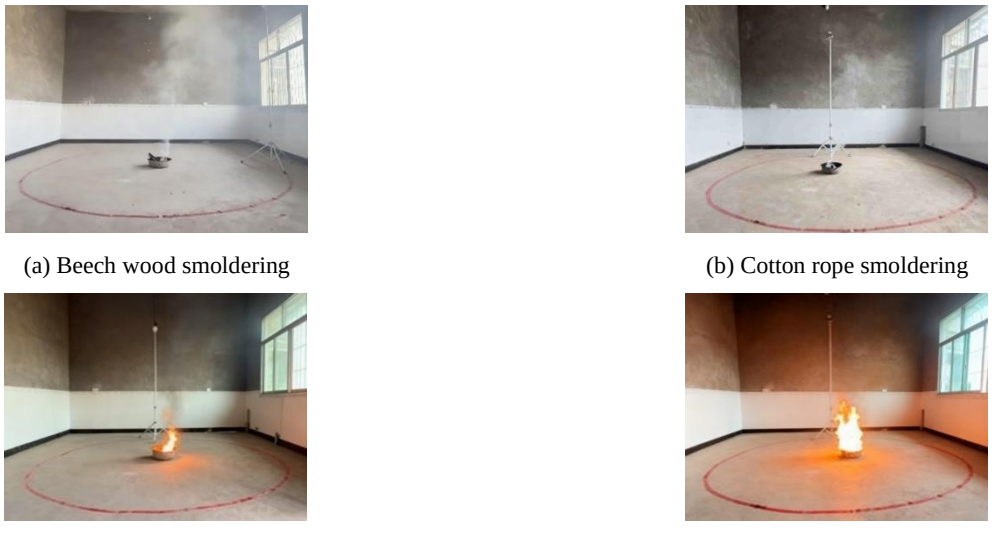
Table 2. Evaluation Metrics for Different Models

Status \ Model Algorithm		FNN	LSTM	LSTM-FNN
Safety	MSE	0.002	0.0014	0.0009
	RMSE	0.0447	0.0374	0.03
	MAE	0.0324	0.0237	0.0144
Smoldering	MSE	0.0011	0.0022	0.0002
	RMSE	0.0332	0.0469	0.0141
	MAE	0.0281	0.0358	0.011
Danger	MSE	0.0049	0.0018	0.0002
	RMSE	0.07	0.0424	0.0141
	MAE	0.0553	0.0258	0.007

Table 2 compares the specific values of each evaluation metric across the three models. Among them, MSE measures the expected value of squared prediction errors and is more sensitive to larger errors; RMSE shares the same units as the prediction target, facilitating intuitive interpretation; MAE represents the average of absolute errors, providing a robust reflection of the average prediction error level. Experimental results indicate that during the no-fire phase, the LSTM-FNN model achieved MSE, RMSE, and MAE values of 0.0009, 0.03, and 0.0144, respectively. During the smoldering phase, the corresponding metrics were 0.0002, 0.0141, and 0.011; during the flaming phase, they were 0.0002, 0.0141, and 0.007. All metrics reached optimal levels, indicating the highest consistency between the model's simulated outputs and expected outputs, demonstrating exceptional regression accuracy and recognition capability.

4.3. Experimental verification

To validate the method's effectiveness, an indoor fire simulation experiment was designed and constructed. Four representative combustible materials were selected to cover diverse fire scenarios: beech wood and cotton rope served as smoldering sources, representing the slow, incomplete combustion characteristics of solid materials; n-heptane and polyurethane acted as open-flame sources, respectively simulating the intense combustion processes of liquid fuels and rapidly pyrolyzing polymer materials. The system sensors included MQ-2 (smoke), MQ-7 (carbon monoxide), and DS18B20 (temperature). The microcontroller (MCU) collected and processed sensor signals, transmitting data via a Wi-Fi module to a PC. A trained LSTM-FNN model performed real-time fire state recognition. The experimental setup is shown in Figure 7.

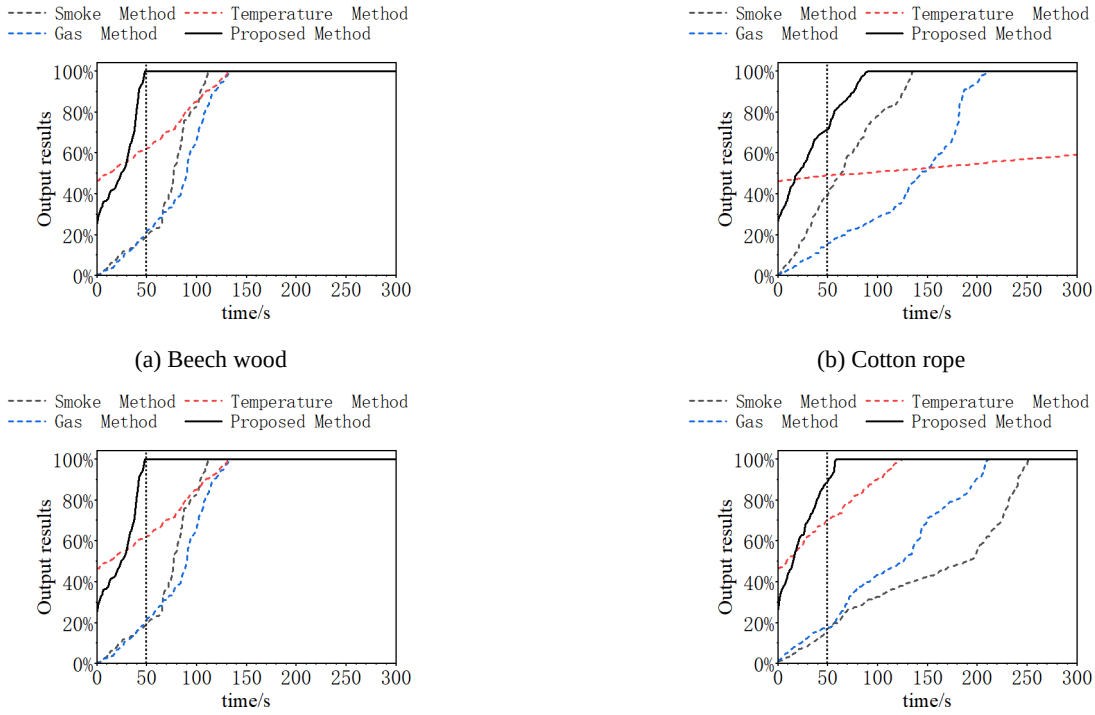


(a) Beech wood smoldering (b) Cotton rope smoldering

(c) Polyurethane open flame (d) n-Heptane open flame

Figure 7. Validation experiment site

Figure 8 displays the visualized results of the model outputs, encompassing both the dynamic trends of sensor characteristic values for different combustibles and the fire probability detected by the model. It is evident that different materials exhibit distinct combustion characteristics. During the smoldering phase, beech wood and cotton rope show minimal temperature changes, making early warning difficult to achieve solely through traditional temperature threshold methods. Polyurethane rapidly releases large amounts of CO and smoke particles during combustion, making it the earliest detectable material. n-Heptane, a simple hydrocarbon, primarily produces CO_2 and water during combustion with minimal CO and smoke generation. However, it exhibits the most pronounced temperature rise, providing the model with effective identification criteria. Experimental results demonstrate that the proposed LSTM-FNN model exhibits outstanding early recognition capabilities across diverse combustion scenarios. Compared to conventional detection methods, this approach significantly enhances the timeliness of early warnings while more accurately capturing dynamic feature changes during fire progression, showcasing strong potential for engineering applications.



(a) Beech wood

(b) Cotton rope

(c) Polyurethane

(d) n-Heptane

Figure 8. Model output results

5. Conclusion

To address the challenges of indoor fire detection, where smoldering fires in their early stages are highly concealed and traditional monitoring methods suffer from underreporting, false alarms, and delayed responses, this study proposes a multi-source data fire identification method based on an LSTM-FNN fusion algorithm. The conclusions are as follows:

(1) Simulation results indicate that during the no-fire phase, the mean squared error (MSE), root mean squared error (RMSE), and mean absolute error (MAE) are 0.0009, 0.03, and 0.0144, respectively; During the smoldering phase, the values were 0.0002, 0.0141, and 0.011, respectively; during the open-flame phase, the corresponding metrics were 0.0002, 0.0141, and 0.007. All metrics outperformed the comparison algorithms, indicating that this method maintains high accuracy in identifying early-stage fire conditions.

(2) Experimental validation confirms that this method identifies smoldering and flaming states earlier than traditional fire detection approaches. By leveraging LSTM's temporal feature capture capabilities and FNN's uncertainty processing advantages, the proposed method significantly enhances the sensitivity and timeliness of early fire warnings compared to single-model approaches, while effectively reducing false alarm and missed detection rates.

References

[1] Hikmat Y, Ahmad Z K, Min U F U, et al. A modified YOLOv5 architecture for efficient fire detection in smart cities[J].Expert Systems With Applications,2023,231DOI:10.1016/J.ESWA.2023.120465.\

[2] Pincott J, Tien P W, Wei S, et al. Indoor fire detection utilizing computer vision-based strategies[J]. Journal of Building Engineering, 2022, 61: 105154.

[3] Khan F, Xu Z, Sun J, et al. Recent advances in sensors for fire detection[J]. Sensors, 2022, 22(9): 3310.

[4] Jin C, Wang T, Alhusaini N, et al. Video fire detection methods based on deep learning: Datasets, methods, and future directions[J]. Fire, 2023, 6(8): 315.

[5] Wang H, Shi Y, Chen L, et al. A Tunnel Fire Detection Method Based on an Improved Dempster-Shafer Evidence Theory[J]. Sensors, 2024, 24(19): 6455.

[6] Vasconcelos R N, Franca Rocha W J S, Costa D P, et al. Fire detection with deep learning: A comprehensive review[J]. Land, 2024, 13(10): 1696.

[7] Diaconu B M. Recent advances and emerging directions in fire detection systems based on machine learning algorithms[J]. Fire, 2023, 6(11): 441.

[8] Liu P, Xiang P, Lu D. A new multi-sensor fire detection method based on LSTM networks with environmental information fusion[J]. Neural Computing and Applications, 2023, 35(36): 25275-25289.

[9] Xu Z, Guo Y, Saleh J H. Advances toward the next generation fire detection: Deep LSTM variational autoencoder for improved sensitivity and reliability[J]. Ieee Access, 2021, 9: 30636-30653.

[10] Park S H, Kim D H, Kim S C. Recognition of IoT-based fire-detection system fire-signal patterns applying fuzzy logic[J]. Heliyon, 2023, 9(2).

[11] Rao G N, Rao P J, Duvvuru R, et al. Neural fuzzy system design in forest fire detection[J]. Microsystem Technologies, 2024, 30(4): 455-467.

[12] Khandoker M A R, Mou R J, Muntaha M A, et al. Numerical simulation of fire in a multistoried ready-made garments factory using PyroSim[C]//AIP conference proceedings. AIP Publishing LLC, 2018, 1980(1): 050026.

[13] Xu M, Peng D. Pyrosim-based numerical simulation of fire safety and evacuation behaviour of college buildings[J]. International Journal of Safety and Security Engineering, 2020, 10(2): 293-299.