This is the Pre-Published Version.
X. Wu, Y. Park, A. Li, X. Huang, F. Xiao, A. Usmani (2020) *Smart Detection of Fire Source in Tunnel Based on the Numerical Database and Artificial Intelligence*, **Fire Technology**, https://doi.org/10.1007/s10694-020-00985-z
This is a post-peer-review, pre-copyedit version of an article published in Fire Technology. The final authenticated version is available online at: https://doi.org/10.1007/s10694-020-00985-z.

Smart Detection of Fire Source in Tunnel Based on the Numerical Database and Artificial Intelligence

Xiqiang Wu[1], Younggi Park[1], Ao Li[1], Xinyan Huang[1,2,*] , Fu Xiao[1], Asif Usmani[1],

[1]*Department of Building Services Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong*

[2]*Research Institute for Sustainable Urban Development, The Hong Kong Polytechnic University, Kowloon, Hong Kong*

ABSTRACT

The fire event in a tunnel creates a rapid spread of heat and smoke flows in a long and confined space, which not only endangers human life but also challenges the fire-evacuation and firefighting strategies. A quick and accurate identification for the location and size of the original fire source is of great scientific and practical value in guiding fire rescue and fighting the tunnel fire. Nevertheless, it is a big challenge to acquire fire-source information in an actual tunnel fire event. In this study, the framework of artificial intelligence (AI) and big data is applied to predict the fire source in a numerical model of the tunnel. A big tunnel fire database of numerical simulations, with varying fire locations, fire sizes, and ventilation conditions, is constructed. Temporally varied temperatures measured by multiple sensor devices are used to train a long-short term memory (LSTM) recurrent neural network (RNN). Results demonstrate that the location and size of the tunnel fire and the ventilation wind speed can be predicted by the trained model with an accuracy of 90%. Sensitivity analysis is also carried out to optimize the database configuration and spatial-temporal arrangement of sensors in order to achieve a fast and reliable fire prediction. This work addresses the possibility of AI-based detection and prediction of fire source and hazard, thus, providing scientifically based guidance for smart-firefighting technologies and paving the way for future emergency-response tactics in a smart city.

KEYWORDS

Tunnel fires; Smart firefighting; CFD; deep learning; recurrent neural networks; LSTM.

Abbreviations

AI	artificial intelligence	HRR	heat release rate (MW)
ANN	artificial neural networks	LSTM	long short-term memory
CFD	computational fluid dynamics	MSE	mean squared error
CNN	convolutional neural network	RNN	recurrent neural network
CVV	critical ventilation velocity	SVM	support vector machine
FDS	fire dynamics simulator		

1. Introduction

Road tunnels are popular in modern cities and intercity transportation due to their efficiency in transport networks and recent advances in construction technologies. Although statistics show that the frequency of serious fire catastrophe in a specific tunnel is vanishingly small [1], the growing number of underground tunnels inevitably increases the risk of tunnel fire incidents, and any fire incident in such a confined and critical infrastructure has the potential to be catastrophic [2]. For instance, the Mont Blanc Tunnel fire in 1999 caused 39 fatalities, and the arson fire in a metro station in Daegu, South Korea, in 2003 caused 198 deaths and 146 injuries [3]. Tunnel fires may also cause severe damage to infrastructure and require enormous sums of money for repair and reconstruction including disruption of the local transportation network [4, 5]. Rapid emission of hot smoke and toxic gases during a tunnel fire makes timely and accurate identification of the fire characteristics of critical importance in guiding firefighting and rescue operations and mitigating infrastructure damage and other losses [5, 6].

Various types of fire-detection techniques have been adopted in practice for tunnel fire protection systems. For example, smoke particles produced during fires can be detected by light extinction sensors, although their reliability is questionable under poor air quality and visibility in tunnels [7]. Video-image-processing systems have witnessed a boom in recent years and have shown the outstanding capability for early fire detection of flame and smoke in tunnels [8–10]. However, low illumination caused by vehicle emissions limits the efficacy of such systems in tunnel fire detection. Furthermore, sophisticated algorithms need to be developed to exclude the interference of sunlight and vehicle lights on the video-image-processing systems to avoid false alarms [11]. Another widely used system is the line-type heat detector [12], which is capable of identifying the exact location of fires but has a high installation and maintenance cost [13]. Although fire locations in a tunnel could also be identified using a massive array of sensors, full-scale tunnel-fire experiments have shown a great deal of reduction in detection accuracy under longitudinal airflow [14]. Moreover, the identification of heat release rates (HRRs) is of great importance since the burning of different vehicles has different HRRs. Ventilation, as the most common tunnel fire safety operation, also has a major effect on the development of fire and the airflow of smoke and toxic gases. In summary, current fire detection systems are not smart enough to determine the fire characteristics [15], such as location, size, and ventilation conditions, which limit response options during a tunnel fire emergency [16].

To achieve real-time prediction of the development of compartment fires, the FireGrid framework [17] was proposed previously to collect real-time sensor data and predict the evolution of compartment fires based on a zone model. Benefitting from the evolutions of artificial intelligence (AI) and big data, data-driven methods especially artificial neural networks (ANNs), have been increasingly applied in fire safety engineering. To detect general fire incidents, Choi and Choi [18] proposed a system including a flame detector with a visual light camera. Fire detection was handled by solving a nonlinear classification problem with a support vector machine (SVM) classifier, but it has a high false-alarm rate. To solve this problem, deep convolutional neural networks (CNNs) have been developed [19, 20]. Without detailed engineering feature extraction, CNNs were claimed to automatically increase the fire-detection accuracy via deep learning. Deep CNNs also showed its capability in detecting large-scale wildland fire with geoinformatics data [21]. However, these methods are mainly based on camera images, and their feasibility under poor visibility needs further evaluation.

Alternatively, sensors that can avoid the visibility problem could enable fast processing and real-time forecasting [17]. For example, data acquired from multiple types of fire sensors, such as smog

sensor, CO sensor, and heat sensor, were used to train a 3-layer neural network for fire detection [22–24]. Temperature, humidity, and gases collected from sensors were also used to train a simple 2-layer feed-forward fully connected neural network [25]. As another application of ANN, studies on gas source identification can provide valuable references for fire safety engineering. Li *et al.* [26] proposed a 3-layer feed-forward neural network to determine the source location of contaminants. Kim *et al.* [27] used gas sensor data obtained from Computational Fluid Dynamics (CFD) models to train a recurrent neural network (RNN) long short-term memory (LSTM) model to predict the location of chemical leakage source. Qian *et al.* [28] predicted the diffusion rule of toxic gases using an LSTM model. The excellent performance of the LSTM model compared with the physical-based Gaussian diffusion model and the SVM model was also demonstrated. It could be possible to apply similar LSTM models on tunnel fire safety; however, none of the existing ANN models have been applied to tunnel fires.

This paper presents the identification of fire characteristics, including fire location, fire size, and ventilation conditions, in a tunnel infrastructure using an LSTM model. Sensor temperature information obtained from CFD models is used for the training of this model. Fire location, heat release rate of the fire and ventilation conditions in the tunnel is then identified.

2. Method framework

2.1. Background of deep-learning algorithm

RNN is specially designed for the prediction or classification of temporal data. Figure 1(a) shows the typical structure of RNN models. The loop structure can be unfolded into a chain of cells. It should be noted that the RNN cells are the same cell at various states, rather than different cells. The order-dependent data are put into the structure and reserved, and then the information hidden inside the sequence of data can be interpreted as the output of the structure. Though has been used in many areas such as speech recognition [29], the major limitation of RNN is the difficulty in treating with long pieces of sequence [30].

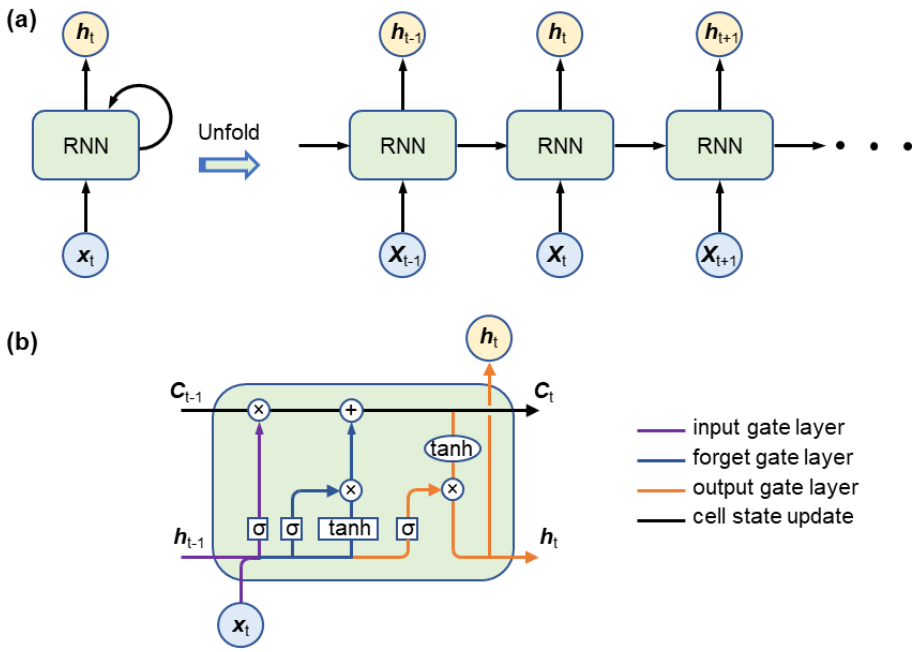


Fig. 1. Data processing in (a) typical RNN; and (b) a typical LSTM cell.

To handle this problem, Hochreiter [31] proposed a new type of RNN, i.e., LSTM, by introducing a specific internal structure. Figure 1(b) shows the analogy diagram of an LSTM cell comprising of three gates. When data are imported into the cell, the forget gate, input gate, and output gate together decide whether to block or pass it on, how to treat it, and how to output it, respectively [32]. The information inside the cell will be updated by forgetting and inputting gates. Therefore, after importing a piece of data sequentially, useful information can be remembered and translated for designed output. The data sequence is processed the same way as RNN shown in Fig. 1(a) by replacing the RNN cell with the LSTM cell. LSTM network has been demonstrated as a powerful network to effectively solve complex prediction problems [33]. In this paper, the LSTM network is adopted for tunnel fire identification. The detailed structure of the network is illustrated in Section 3.2. Figure 2 shows an overview of the proposed method framework to identify the fire source information in the tunnel. First, a large number of tunnel fire scenarios were simulated. Then the temperature histories measured by sensors were extracted from the simulation results to form a big database. Finally, the database was used to train an LSTM model to identify the fire source in tunnels.

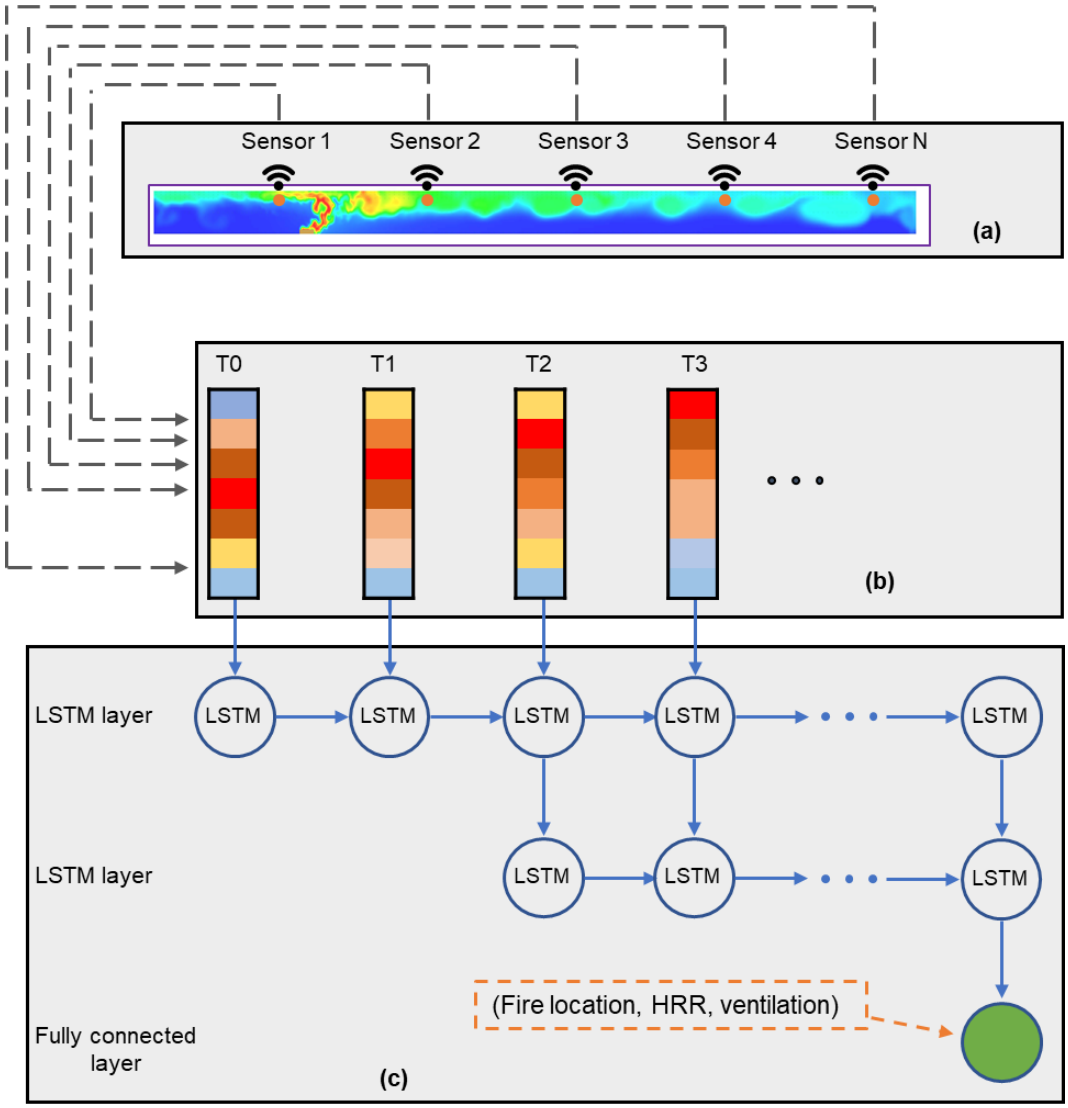


Fig. 2. Overview of the method framework: (a) CFD simulation; (b) Database generation; (c) LSTM model.

2.2. CFD modeling of tunnel fire

2.2.1. Fire scenarios

It would be ideal if adequate sensor data can be obtained from full-scale fire experiments or real tunnel fire incidents. However, the existing tunnel-fire data from the literature is not large enough or sufficiently well organized to support deep learning. Alternatively, a database formed by massive numerical simulation could enable AI-based fire forecasting. In this work, CFD-based numerical simulations and a numerical tunnel-fire database are adopted to demonstrate a tunnel fire, as shown in Fig. 3.

The numerical model is established in Fire Dynamics Simulator (FDS) version 6.7 [34], which is developed by the National Institute of Standards and Technology. FDS is an open-source CFD tool using the large eddy simulation turbulence model. This CFD code has gained extensive validation and verification and has been widely used to simulate tunnel fires for research and design [35, 36]. Three parameters, i.e., the fire location, HRR, and outlet ventilation velocity, were considered. As a first-step demonstration, only 4 to 5 varieties were considered for each parameter to form an event library of 100 cases. This event library can be enlarged for better performance of identification by incorporating more parameters, varieties, and tunnel configurations.

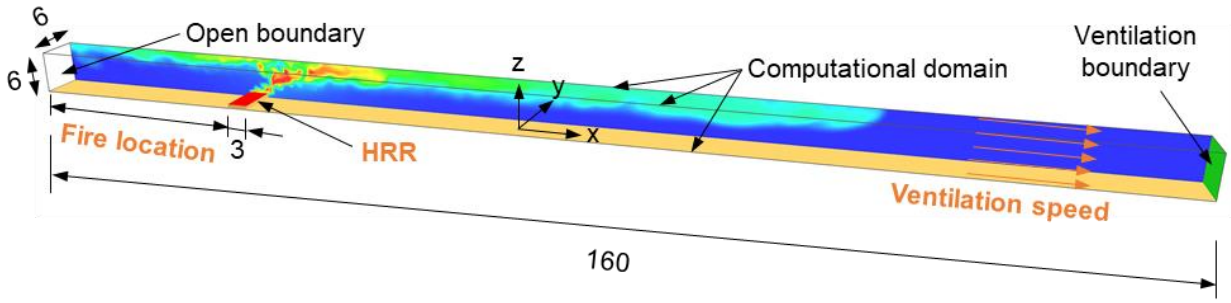


Fig. 3. Illustration of the numerical model of tunnel fire and key parameters (unit: m).

The location of the fire source plays a critical role in emergency response and firefighting activities. From the viewpoint of tunnel fire dynamics, the fire location controls the gas temperature distribution and smoke movement in tunnels [37]. To identify the fire hazard and facilitate prompt operations of fire suppression and rescue, it is essential to determine fire fuel location. In practice, five fire locations relative to the total length of the tunnel are randomly set 0.1, 0.2, 0.4, 0.6 and 0.8, that is, 16 m, 32 m, 64 m, 96 m, and 128 m, respectively, from the inlet on the left-hand side.

The HRR is the most important parameter to describe the severity of a tunnel fire, while the magnitude of HRR varies in a large range, depending much on vehicle types. Specifically, the peak HRRs are 1.7-4.6 MW for passenger cars, 29-35 MW for buses, and 60-200 MW for tanker trucks, respectively [2, 3, 38, 39]. Since generally fire first undergoes a developing period before reaching the peak HRR, the early-stage vehicle fire should have a relatively lower HRR. In this paper, five HRR values of 5 MW, 10MW, 20 MW, 50 MW and 80 MW are assumed for different tunnel fire scenarios. HRR is assumed constant from the ignition of fire.

Smoke, toxic gases and hot airflow produced from the fire threaten people trapped in the tunnel. To mitigate the fire hazard, the ventilation system has been regarded as the most important facility in tunnel fire safety design. For the current study, a simple and uniform longitudinal ventilation was applied to the tunnel exit on the right-hand side. The critical ventilation velocity (CVV) is the minimum velocity to prevent the back-layering of smoke in tunnels, and its value increases with HRR

[3]. For this tunnel, CVV can be calculated as 2.82 m/s, when HRR is 80 MW, by using the empirical correlation proposed in [40]. The ventilation speed is thus supposed as 0 m/s, 1 m/s, 2 m/s and 4 m/s to consider the variety of real ventilation conditions. Overall, 5 fire locations, 5 HRRs and 4 ventilation conditions constitute 100 (= 5 × 5 × 4) fire scenarios.

2.2.2. Model setup

Prohibitive magnitudes of computational resources would be required for conventional 3-D modeling of full-scale tunnels. Thus, as a first approximation and demonstration, 2-D modeling was considered to be a better choice for its simplicity, efficiency and cost-effectiveness. The accuracy and efficiency of 2D modeling tunnel fires using FDS have been demonstrated in the literature [35, 41, 42], and Fig. A1 demonstrates the similarity in the modelled temperature evolution between 2-D and 3-D model. Considering a large number of cases that must be simulated, all the fire scenarios were computed with 2-D models ignoring the variation in the transverse (y) horizontal direction.

Figure 3 shows the tunnel fire model built up using FDS. The tunnel is 160 m in length (x), 6 m in width (y), and 6 m in height (z), which mimics the full-scale road testing tunnel at Sichuan Fire Research Institute, China. The thermal boundaries of tunnel surfaces were set as default inert wall with the temperature fixed at ambient temperature of 20 ℃. The left exit of the tunnel was set to open. The right exit was also set as open boundary if ventilation equals zero; otherwise, it was set as a constant value of the ventilation speed in the direction of positive x-axis to exhaust the smoke and gases produced. The inner surfaces of the tunnel were assumed as perfect smooth with roughness set as 0. The fuel representing a burning vehicle was modeled with a rectangular burner of 3 m along the x-axis and 6 m along the y-axis. The burner was located 0.5 m above the ground to mimic a vehicle fire. A constant value of HRR from the ignition of the fire was specified for the burner.

The grid size is a key parameter for the accuracy and computational speed of modeling using FDS. The non-dimensional expression $D^*/\delta x$ is a criterion to evaluate the quality of grid resolution, where D^* is the characteristic fire diameter, and δx is the nominal length of cell edge [43]. Studies showed that accurate predictions of airflow velocity and temperature could be achieved when $D^*/\delta x$ was no smaller than 10 in fire regions [44]. Then, the calculated maximum cell size changed from 0.2 m to 0.54 m with the increase of HRR from 5 MW to 80 MW. The same cell size was deployed in both x- and z-axis.

The same mesh method is adopted for all the fire models. In each fire model, the whole computational domain was divided into three regions. The region within 20 meters to the center of the burner in the x-axis was meshed with smaller cells of 0.05 m, while the remaining two regions had a larger cell of 0.2 m, which conformed to the calculated mesh size. The y-axis was meshed with a single cell throughout the tunnel for 2-D modeling using FDS. Thus, the total number of cells was 69,000 for all cases. Each tunnel fire scenario was calculated for 300 s, which was long enough for all fire scenarios to reach the steady stage, where temperatures at different locations of the tunnel evolves periodically (see more details in Fig. A2). Temperature sensors, which measure the gas temperature distribution, were evenly distributed 0.1 m lower from the tunnel ceiling with an interval of 1 m. The temperature was output at every one second.

2.2.3. Database generation

For each of the 100 CFD simulations, temperatures measured by 160 sensors during a period of 300 s from the beginning of the simulation with an interval of 1 s can be recorded and stored in a data sheet. Figure 4 shows the flow chart to generate the database for the training of the LSTM network. Then, each data sheet could be cut into 241 data samples through sweeping, each of which has a

temporal length of 60 s. The 241 data samples multiplying 100 scenarios form a database having 24,100 samples.

With supervised learning, each sample should be labeled with an expected output of the LSTM model, and this was done accompanied by the data segmentation. Here in this paper, the output is the fire location, HRR and ventilation. Then, all the labeled samples were randomly split into a training dataset, a validation dataset and a test dataset with a scale of 60%, 20%, and 20%, respectively. The training dataset is used for training the model, the validation dataset is used to evaluate the fitted model while training, and the test dataset is used to estimate the quality of the fitted model after training.

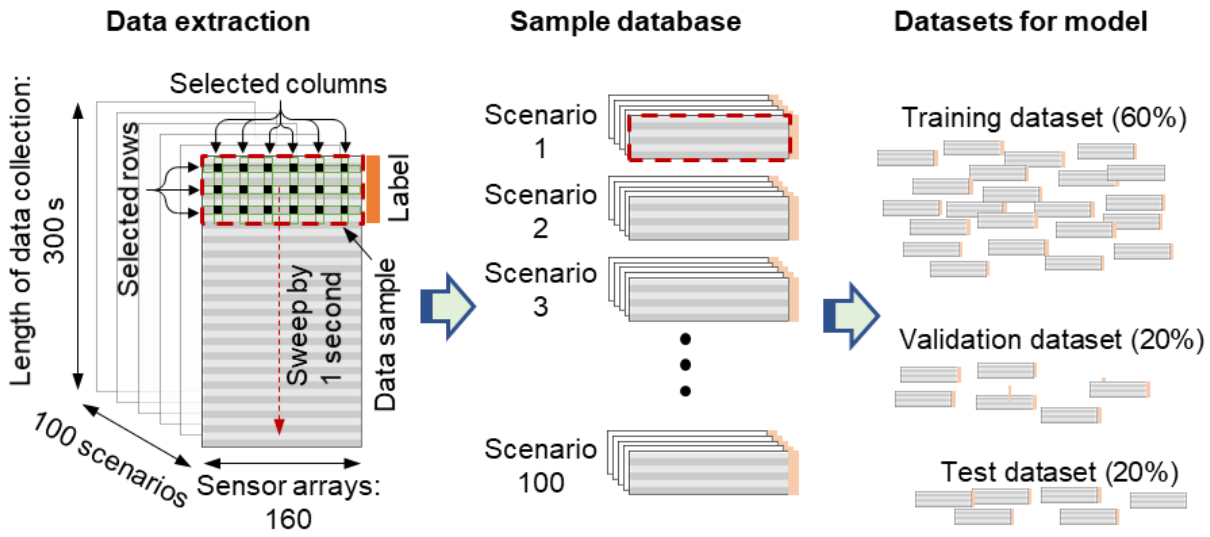


Fig. 4. Process of datasets generation.

2.2.4. Optimization of sensor allocation

In practice, it is very costly to install and maintain many temperature sensors, such as thermocouples, especially for a long tunnel. Moreover, for data collection, too many redundant data will be produced if data collection is too frequent, such as every second, which may slow down the data processing. While valuable information would be missed if the insufficient data is collected, which may delay the detection and subsequent firefighting operation. For real-time fire prediction, a balance between the size of data and the speed of processing is required based on a rational arrangement of sensors.

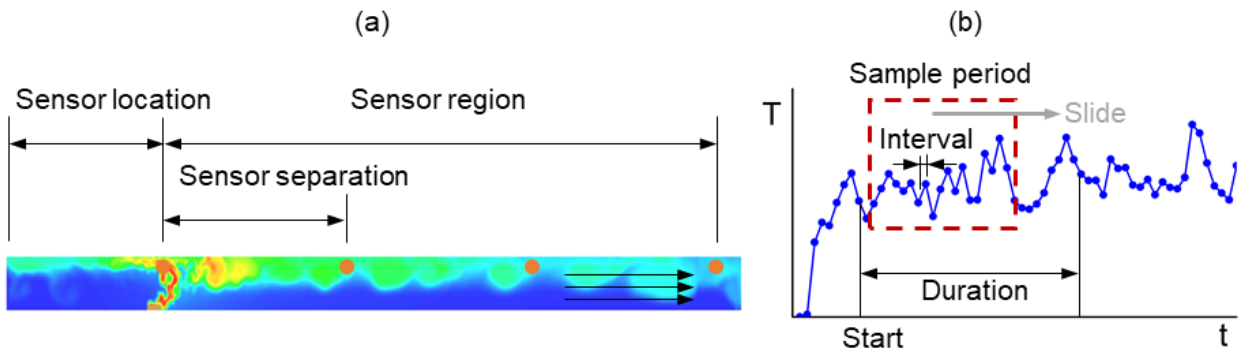


Fig. 5. Optimization for sensor allocation: (a) spatial factors; (b) temporal factors.

Figure 5 shows seven factors of sensor data that are evaluated for the accurate identification of tunnel fires. That is, three spatial factors in Fig. 5a, (1) sensor location, which is the distance from the left tunnel to the left sensor, (2) sensor region, in which area the sensor is located, and (3) sensor separation, which is the separation between adjacent sensors; and another four temporal factors, (4) sample period, during which the data will be trained in a sample, (5) start of data collection, which is the start time to collect data, (6) collection duration, during which data are collected, and (7) collection interval, which is the interval of data collection, in Fig. 5b, are considered. The number of temporal points in a sample equals the sample period divided by the collection interval. Overall, 7 groups containing 72 cases are studied. More detailed information on all the cases is shown in Table A1 of the Appendix. Besides, a study case using all these simulation results is marked as the benchmark case. The default values for the 7 factors are 0 m, 160 m, 20 m, 10 s, 0 s, 30 s, 1 s, respectively. For clear showing, default values are not displayed unless otherwise specified. The model of the benchmark case was trained first, and the models for each case were trained individually. The influences of these factors on the accuracy of predicting tunnel fire (i.e., location, HRR, and ventilation velocity) are evaluated group by group. To quantify the factor's sensitivity on prediction in each group, only the value of the corresponding factor changes while keeping the values of the remaining factors constant, i.e., the method of controlling variable. However, factor values in a current group are determined based on study on previous groups; thus, they may vary among groups.

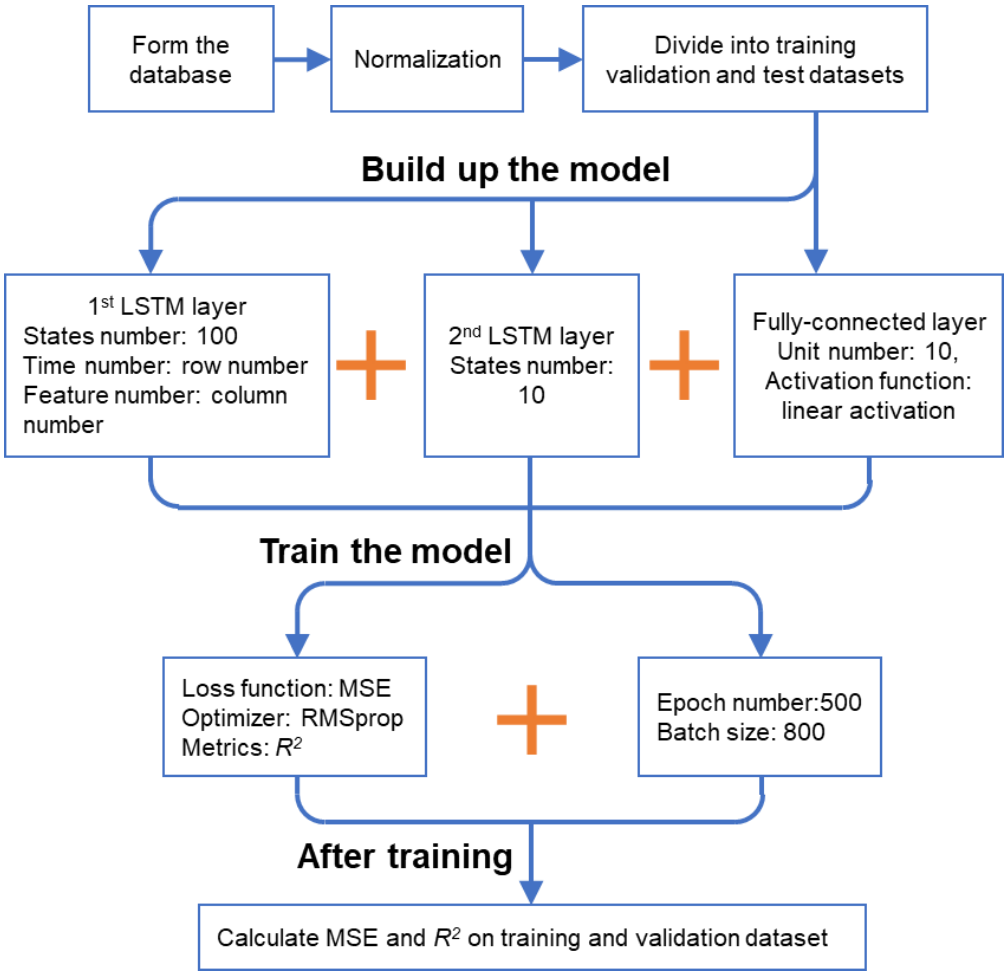


Fig. 6. Model development and validation process: normalization and divide the database; build up the model; train the model; calculate the metrics.

It should be noted that the training of all the cases uses the 100 CFD simulations, no further numerical simulations are needed. For data extraction shown in the first step in Fig. 4, only the columns corresponding to the selected sensor locations calculated from the spatial factors, and the rows corresponding to the collection time points calculated from the temporal factors are extracted. The following sample database and datasets for the model are still applied.

2.3. LSTM model

2.3.1. Model architecture

In this section, a regression LSTM network model is built up for identifying fire scenarios in tunnels. Deep LSTM networks having more than three layers are difficult to train due to the potentially large number of parameters to be fitted [45]. This can be more challenging if the database is small. The proposed network (Fig. 2) has two LSTM layers, followed by a fully connected layer as the output layer. The input layer has 100 hidden states, among which the outputs of the last 10 states are transferred into the second layer. The output of the last state of the second layer is exported to the output layer. Overfitting problem could be caused if detailed characteristics or noise of the training data is also learned by the model. The performance of the model would then be too perfect on trained data meanwhile unsatisfactory on untrained data [46]. To decrease the possibility of overfitting, a dropout ratio can be set after each layer. In the current model, an initial dropout ratio of 0.1 [47] was set after each LSTM layer.

The samples generated in the training dataset are then used to train the established model. For each sample, the LSTM unit in the input layer receives a vector of temperatures measured by sensors sequentially. The received data is then treated and passed on to the next state. A three-element vector (fire location, HRR, and ventilation speed) can be predicted by the unit in the output layer. The database used in this study is available on Github: https://github.com/wu-xiqiang/Prediction-of-Fire-Source-in-Tunnel. The programme is written using Python 3. The deep learning library Keras [54] and related third-party libraries are included in the modules.

2.3.2. Training settings

It is critical to ensure that the training dataset, validation dataset and test dataset are representative of the overall distribution of the samples. Thus, all labeled samples were randomly divided into these datasets. As illustrated in Fig. 3 and Section 2.2.1, the labels, i.e., fire location, HRR and ventilation, have evidently distinct range. It is also required to avoid the situation that the feature having a larger range would dominate the computation of similarity [48, 49]. Thus, the labels are individually normalized to make them have the same range between 0 and 1 with the MinMaxScaler function [50].

The mean squared error (MSE), root mean squared error (RMSE) and mean absolute error (MAE) are the most common optimized metrics for regression models. MSE calculate the squared error between target and predicted values, and it has been widely adopted due to its simplicity. RMSE is the root of MSE. MSE and RMSE are much similar except that they have different gradient. MAE measures the absolute differences between actual values and predictions. Compared with the other two metrics, MAE is insensitive to large deviations in the predictions. In the current study, MSE is used as the loss function to characterize how well the model behaves. The training of the model is conducted by minimizing the value of loss function (MSE in this case). The above metrics can also be used as performance metrics to evaluate the models. Though they can give the exactly values of error in predictions, they cannot tell whether a model needs to be further improved by just checking these values, e.g. we don't know whether a model gaining an MSE of 0.02 is good enough or not. As another metrics, the coefficient of determination R^2 is scale-free, meaning that it is independent of the

exact differences of the preditions, and performance of models can be directly evaluated compared with this value. Thus, R^2 has been widely adopted for the regression model [51, 52]. In this method, MSE and R^2 are two criteria that can be used for evaluating the fire prediction. It should be noted that R^2 is just calculated during the training but is not used for training the model. For the optimizer method, RMSprop is selected as the optimizer method for its outstanding performance in [28].

Figure 6 summarizes the detailed process of training the proposed model. Linear activation is adopted for the last full connected layer [53]. Apart from the dropout technique, the choice of training epochs can also tackle the overfitting problem. The loss of the model prediction showed a decreasing but then an increasing trend with the increase of training epochs [27]. For the current LSTM model, preliminary study showed that 500 epochs would be enough to obtain a minimum loss while not causing overfitting. The batch size determining how many samples would be trained in an iteration should better be large while this is limited by the physical memory of the computer or server on which an ANN is trained.

3. Results and discussions

3.1. Benchmark case

Figure 7 shows the calculated performance evaluations, including R^2 and loss on training dataset and validation dataset for benchmark case evolving with training epochs. The performance evaluations on three fire parameters of fire location, HRR, and ventilation speed are also calculated individually, as shown in Fig. 7a-c. Their overall loss, which equals to the average of three individual losses, is the loss function minimized in the training process (Fig. 7d). Similarly, the overall R^2 uses the average for three parameters (Fig. 7e).

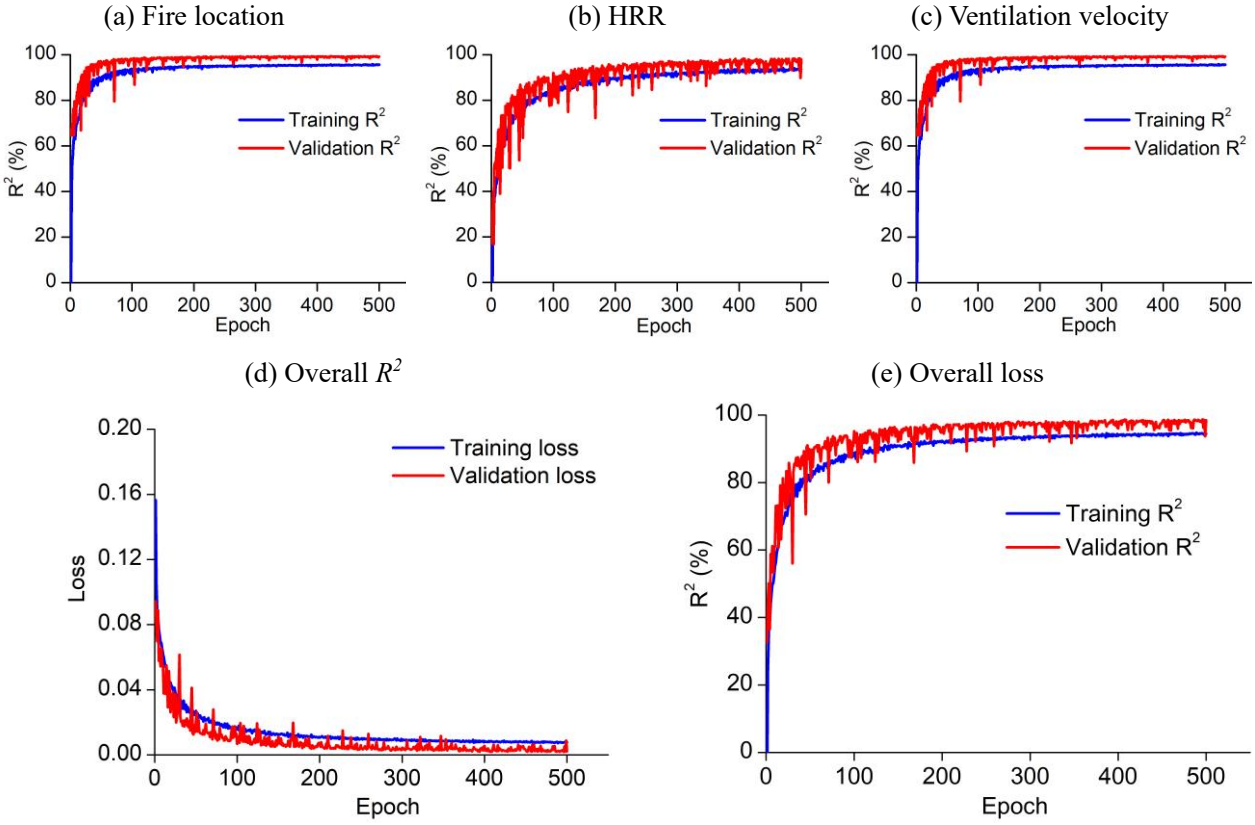


Fig. 7. The variable R^2 with (a) fire location, (b) HRR, (c) ventilation speed, (d) average of three tunnel fire parameters, and (d) the overall loss during the training process.

Compared to HRR and ventilation velocity, R^2 of fire location increases fastest with the training epochs, indicating that it is easiest to train and find the fire location inside the tunnel. From the viewpoint of tunnel fire dynamics, this is due to the more evident distinctions of temperature distribution in the tunnel caused by scenarios with fire located at different positions. Despite that, the identification of all three fire information parameters can be trained more effectively, and finally converges to a high value of R^2 approaching 100%.

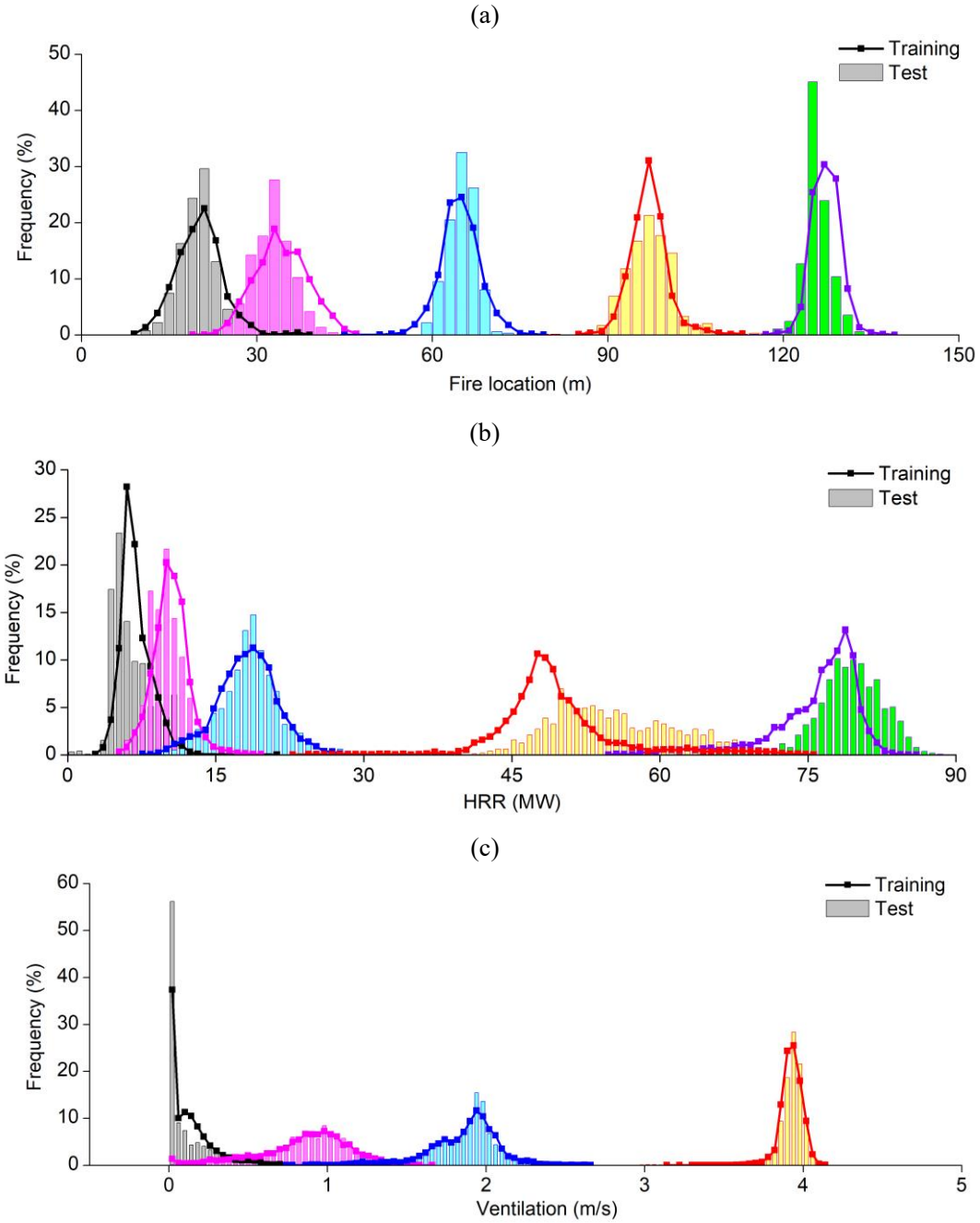


Fig. 8. Distributions of the prediction on: (a) fire location; (b) HRR; and (c) ventilation.

Figure 7e also shows that the overall loss decreases with the number of training epochs on the training dataset, as expected. The training efficiency is high at the starting of training, while no apparent minimization is observed after epochs of 200 epochs, which demonstrates that the model has converged, and the pre-set training epoch number of 500 is sufficiently large. The loss on the

validation dataset is comparable to that on the training dataset, meaning that the overfitting problem was not caused. The validation loss is even smaller than training loss due to the setting of dropout after each LSTM layer. The loss fluctuates slightly on the training dataset, while more drastic fluctuations were found on the validation dataset, which is a common consequence of machine learning [55, 56].

To further check the detailed performance of the model on all the samples, the predictions of fire information corresponding to respective factor values are counted, and the frequency distribution is illustrated in Fig. 8. The predicted values have shown a certain distribution around, rather than equal to the actual values. The varieties of the distribution are larger for the prediction of HRR and ventilation. As revealed in Fig. 7, the converging speeds of HRR and ventilation are slower than the fire location, larger variations could be attributed to the similar nature of temperature distribution near the ceiling of the tunnel. It should be noted that despite the uncertainty in prediction, the prediction attaining the highest frequency is very close to the respective actual value, and the distribution is only confined to a small range around it.

3.2. Sensor optimization

The overall R^2 of the cases in all groups are calculated, and the optimization of sensor allocation is done by comparing their R^2 values. Specifically, more attention should be paid on the factors sensitive to changes, and the increasing trend of R^2 points out the direction for optimization.

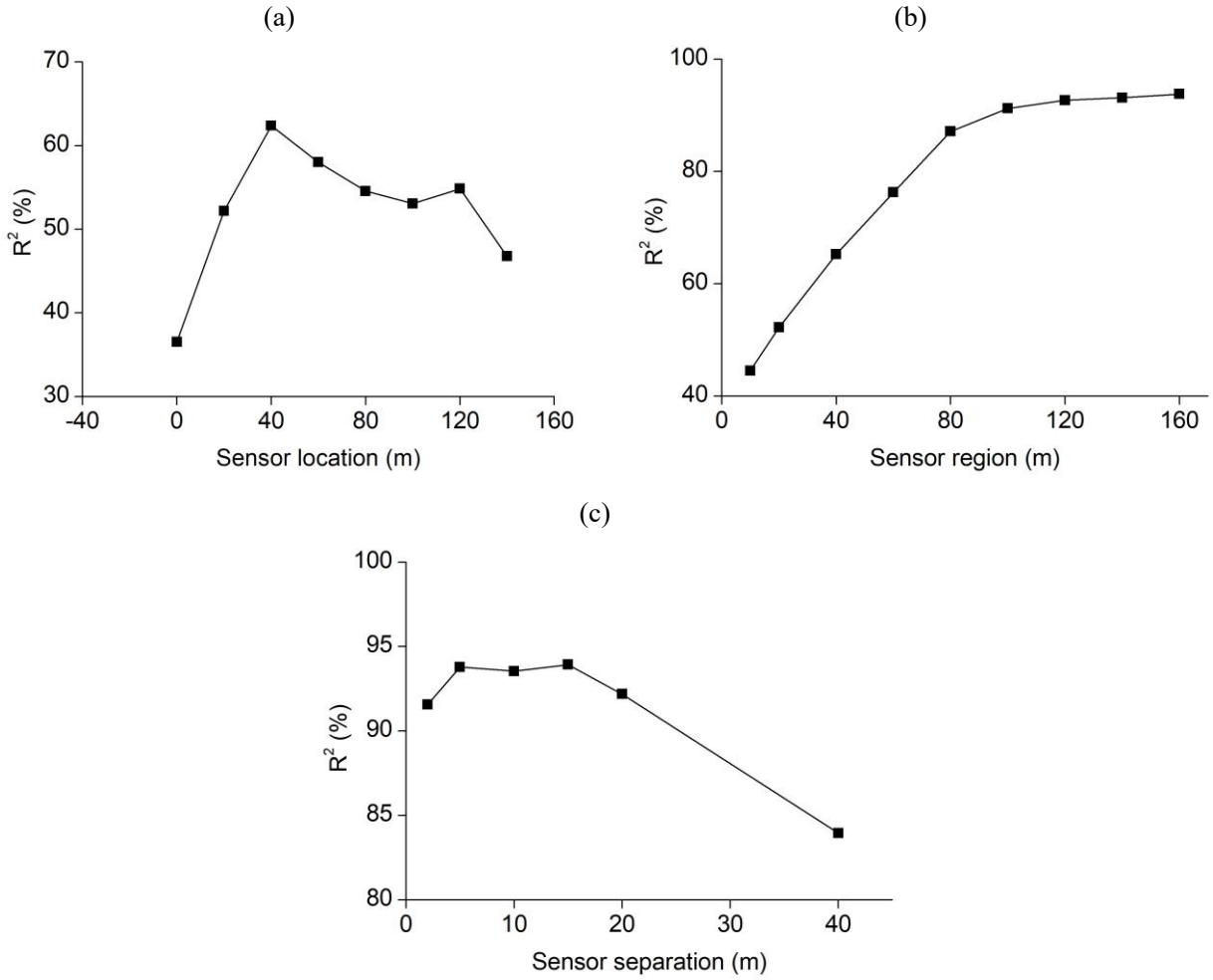


Fig. 9. R^2 of three spatial factors for optimization: (a) sensor location, (b) sensor region, and (c) sensor separation.

Figure 9 compares the value of R^2 on the training dataset for spatial arrangement factors of temperature sensors. In Group 1, the sensor location varies from extreme left to the extreme right of the tunnel. As shown in Fig. 9a, the prediction is more accurate by placing these sensors near the middle area of the tunnel, which may be for the reason that these sensors can acquire plenty of information when the fire is located in the middle of the tunnel. Meanwhile, they are also capable of detecting the fire when the fire is placed on the side of the tunnel. Based on this finding, the sensors in Group 2 are positioned centering in the middle of the tunnel. The sensor region expands from 10 in the center to the whole tunnel.

As illustrated in Fig. 9b, the sensor region has a significant influence on predicting accuracy. The value of R^2 increases almost linearly until a range of 80 m in Case 13 but shows only a slight increase after then. The sensor region is then set as the whole tunnel, and the sensor separation changes between 2 m and 40 m in Group 3. As shown in Fig. 9c, the prediction will not be influenced much if the separation is no larger than 20 m. The sensor separation of 20 m is then adopted in the following study on the factors influencing the temporal collection of data.

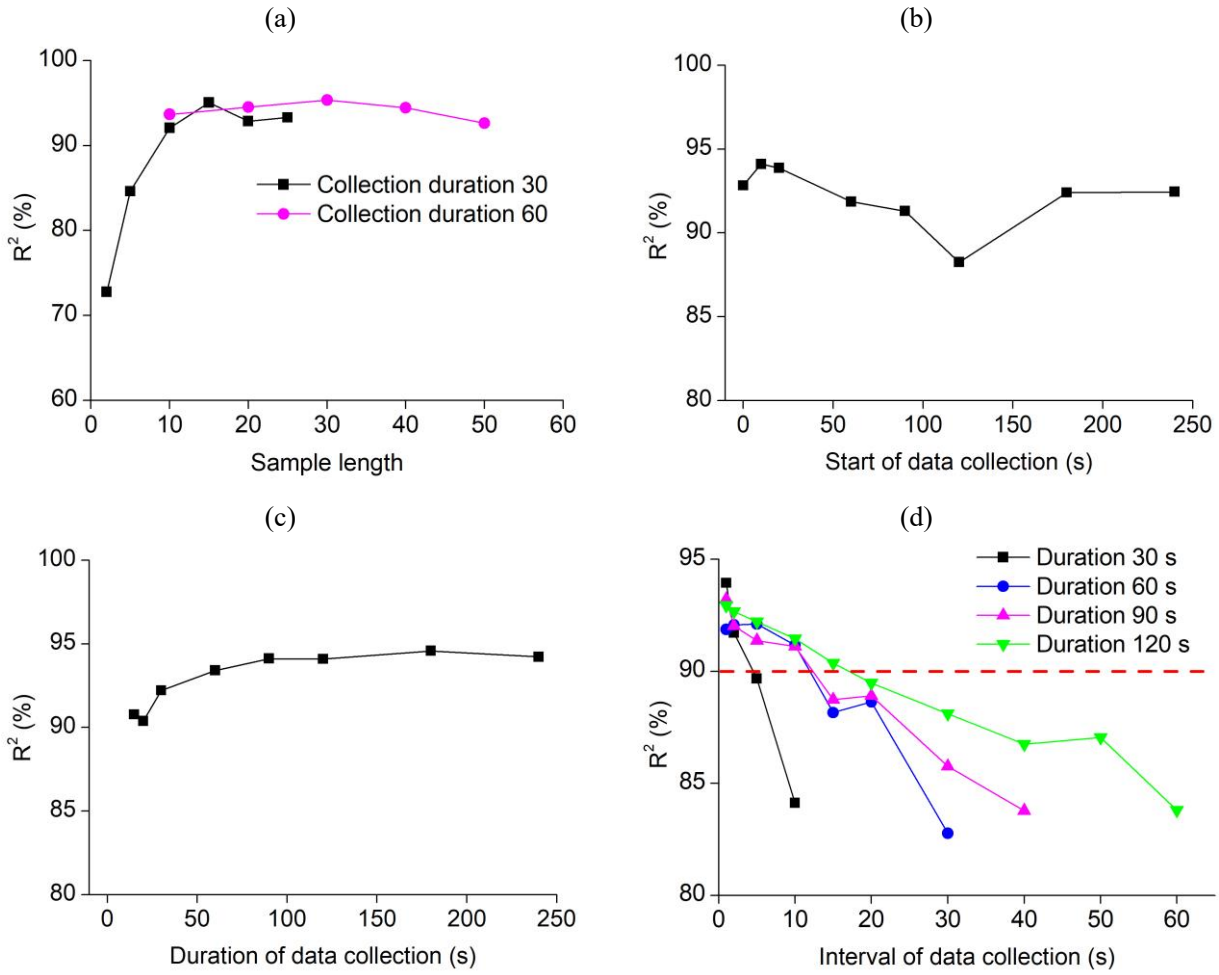


Fig. 10. R^2 of four temporal factors for optimization: (a) sampling period, (b) start of data collection, (c) duration of data collection, and (d) interval of data collection.

The influence of the sampling period is estimated in Group 4 (Fig. 10a). Two collection duration, i.e., 30 s and 60 s, are evaluated. A longer duration can be cut into samples with a larger period. Generally, a larger sampling period will result in a better prediction. It is because a larger sampling

period carries more information about a specific fire scenario, which specializes in the scenario for identification. The predictions will be accurate if the sample period is no smaller than 10 s.

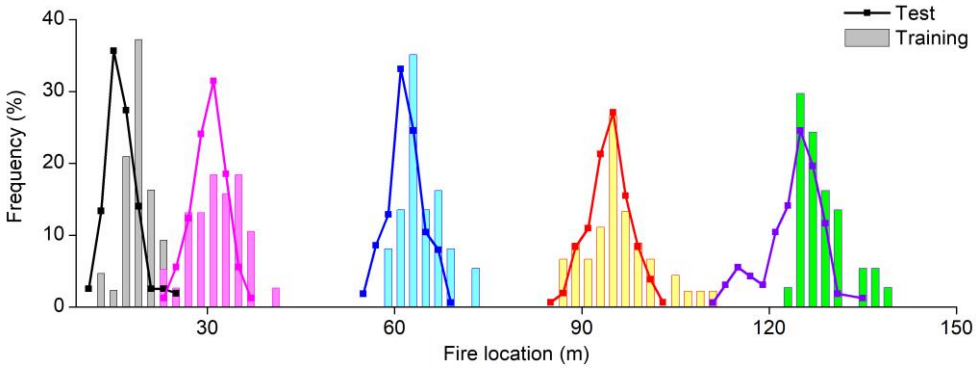
In Group 5, the start of the data collection changes from ignition to 240 s after ignition. As illustrated in Fig. 10b, the predicting accuracy is not obviously influenced by the delay of the start of data collection. In Group 6, the collection duration changes from 15 s to 240 s (Fig. 10c), overall, the prediction approximately becomes more accurate for a longer duration of data collection. The predicting accuracy will not increase obviously if the duration is longer than 90 s.

In Group 7 (Fig. 10d), the data interval for collection durations varying from 30 s to 120 s is evaluated. A smaller interval produces a more accurate prediction. However, Case 46, with a sampling period of 20 s and an interval of 2 s, attains a comparable accuracy as Case 41, with a sampling period of 10 s and an interval of 1 s. It should be noted that the number of temporal points for Case 46 and Case 41 is the same at 10. Thus, the predicting accuracy could be attributed to the number of data collected, rather than the interval. Nevertheless, for a limited data collection period, the influence of the number of data collected can be interpreted as the interval of data collection.

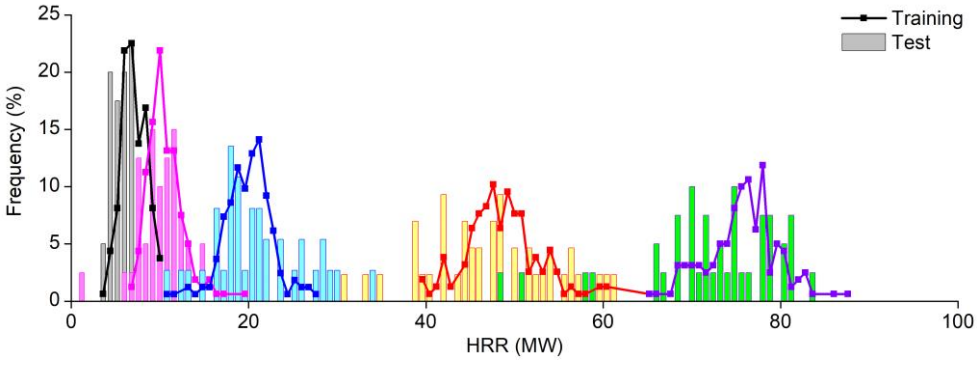
3.3. Suggestions for tunnel fire identification

Based on the evaluation of the factors considered in the optimization of sensor allocation, suggestions can be given on the sensor allocation for identifying key fire scenarios in tunnels. Sensors should be arranged throughout the tunnel with a separation of 20 m. The temporal collection of data depends on practical requirements. This evolves criteria of acceptable accuracy, the delay for identification, and the interval of data collection. The suggestions are mainly made based on Figs. 9 and 10. For example, the criterion of acceptable accuracy is 90%, and as early identification, the delay for identification of the tunnel fire scenario after ignition should be no longer than 30 s, then the interval of data provided by the sensor should be smaller than 5 s. In this case, we can identify the tunnel fire scenario with the parameters in Case 46 attaining accuracy of 92%.

(a)



(b)



(c)

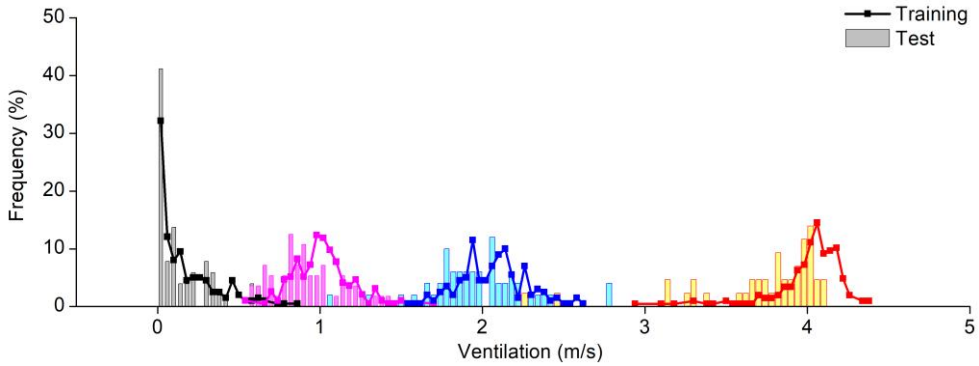


Fig. 11. Distributions of the prediction for: (a) fire location; (b) HRR; and (c) ventilation when acceptable accuracy is 90%.

Figure 11 shows the distribution of the frequencies predicted for individual factors on the test dataset. Compared with those of the benchmark case shown in Fig. 8, distributions predicted by Case 46 is more dispersive, which could be attributed to the lower accuracy of Case 46. However, it should be noted that despite its vast variety in distribution, the distributions are still confined to respective actual values, and the predicted value achieving the highest frequency is also very close to the actual values.

The closeness of the predicted fire source to the actual fire scenario indicates that the proposed LSTM network is a promising method for identifying the fire sources in tunnels. However, better performance of the model could be achieved by tuning the numbers of LSTM layers, fully connected layers and units in each layer. The analysis assumes that the CFD simulations are well representative of the evolution of real tunnel fire scenarios. Further training of the proposed model can be done using data obtained from the real fire or large-scale fire tests. Regarding more rational suggestions for practical engineering, further optimization can be made by analyzing more cases. Also, the feasibility of these suggestions for a tunnel with different shapes or dimensions needs further evaluation.

4. Conclusions

This study demonstrates the feasibility of AI and big data on identification fire source in tunnels. An LSTM model with two LSTM layers and a fully connected layer is proposed, and a big numerical database is established by the sensor temperature measurements of 100 tunnel-fire scenarios. The database was then used to train the proposed LSTM model. Results show that the proposed LSTM model is capable of identifying the fire sources in tunnels. Though the predicted values showed a certain frequency distribution around the respective actual value, the distributions were very condensed and the predicted value having the highest frequency was quite close to the actual value. Overall, the fire information including fire location, HRR and ventilation can be well predicted. A predicting accuracy of 94% can be achieved when the sensors are deployed over the whole tunnel with a separation of 1 m.

The optimization showed that the sensor region has an evident influence on predicting accuracy, while the influence of sensor separation and the sensor location are relatively insignificant. Therefore, it is suggested that the sensors are evenly distributed throughout the tunnel with a separation of 20 m. The amount of data collected or the interval of data collection has a major effect on the performance of the model provided with a limited collection duration, while the influence of the start of collection and the duration are negligible. Generally, a prediction with an accuracy of greater than 90% can be made if provided with no less than 10 sequentially collected data.

The practical design of fire source identification in tunnels can be conducted with the findings on the factors considered in the optimization. Once the requirements of a satisfactory predicting accuracy and an acceptable delay of prediction are proposed, the largest interval of data collection by the sensors can be determined. To provide more rational suggestions, a further detailed evaluation of the sensor allocation is necessary for a tunnel with different shapes or dimensions. This work addresses the possibility of AI-based detection and prediction of fire source and hazard, potentially providing scientifically-based guidance for smart-firefighting technologies and paving the way for future emergency-response tactics in a smart city.

Acknowledgments

This work is funded by the Hong Kong Research Grants Council Theme-based Research Scheme (T22-505/19-N) and the PolyU Emerging Frontier Area (EFA) Scheme of RISUD (P0013879).

References

1. Beard A, Carvel R, Carvel R, Marlair G (2005) 1. A history of fire incidents in tunnels. In: The handbook of tunnel fire safety. pp 1–41
2. Li YZ, Ingason H (2018) Overview of research on fire safety in underground road and railway tunnels. Tunnelling and Underground Space Technology 81:568–589 . doi: 10.1016/j.tust.2018.08.013
3. Ingason H, Li YZ, Lönnermark A (2014) Tunnel fire dynamics. Springer, London
4. Zhong W, Lv J, Li Z, Liang T (2013) A study of bifurcation flow of fire smoke in tunnel with longitudinal ventilation. International Journal of Heat and Mass Transfer 67:829–835 . doi: 10.1016/j.ijheatmasstransfer.2013.08.084
5. Han D, Lee B (2009) Flame and smoke detection method for early real-time detection of a tunnel fire. Fire Safety Journal 44:951–961 . doi: 10.1016/j.firesaf.2009.05.007
6. Li J, Liu J (2020) Science Mapping of Tunnel Fires: A Scientometric Analysis-Based Study. Fire Technology. doi: 10.1007/s10694-020-00969-z
7. Aralt TT, Nilsen AR (2009) Automatic fire detection in road traffic tunnels. Tunnelling and Underground Space Technology 24:75–83 . doi: 10.1016/j.tust.2008.04.001
8. Diehl DA (1976) Fire Detection Systems. In: Plumbing Eng. pp 23–26
9. Noda S, Ueda K (2002) Fire detection in tunnels using an image processing method. pp 57–62
10. Cho BH, Bae JW, Jung SH (2008) Image processing-based fire detection system using statistic color model. Proceedings - ALPIT 2008, 7th International Conference on Advanced Language Processing and Web Information Technology 245–250 . doi: 10.1109/ALPIT.2008.49
11. Çetin AE, Dimitropoulos K, Gouverneur B, et al (2013) Video fire detection - Review. Digital Signal Processing: A Review Journal 23:1827–1843 . doi: 10.1016/j.dsp.2013.07.003
12. Liu Z, Kim AK (2003) Review of recent developments in fire detection technologies. Journal of Fire Protection Engineering 13:129–151 . doi: 10.1177/1042391503013002003
13. Jevtić RB, Blagojević MDJ (2014) On a linear fire detection using coaxial cables. Thermal Science 18:603–614 . doi: 10.2298/TSCI130211102J
14. Liu ZG, Kashef AH, Lougheed GD, Crampton GP (2011) Investigation on the Performance of Fire Detection Systems for Tunnel Applications--Part 2: Full-Scale Experiments Under Longitudinal Airflow Conditions. Fire Technology 47:191–220 . doi: 10.1007/s10694-010-0143-3

15. Smith RL (1987) ASKBUDJr: A precursor of an expert system for the evaluation of fire hazard. Fire Technology 23:5–18 . doi: 10.1007/BF01038362
16. Anon (1992) Fire safety and the law. In: Fire Prevention, 2nd ed. pp 39–43
17. Han L, Potter S, Beckett G, et al (2010) FireGrid: An e-infrastructure for next-generation emergency response support. Journal of Parallel and Distributed Computing 70:1128–1141 . doi: 10.1016/j.jpdc.2010.06.005
18. Choi J, Choi JY (2016) An integrated framework for 24-hours fire detection. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). pp 463–479
19. Muhammad K, Ahmad J, Mehmood I, et al (2018) Convolutional Neural Networks Based Fire Detection in Surveillance Videos. IEEE Access 6:18174–18183 . doi: 10.1109/ACCESS.2018.2812835
20. Kim NK, Jeon KM, Kim HK (2019) Convolutional recurrent neural network-based event detection in tunnels using multiple microphones. Sensors (Switzerland) 19: . doi: 10.3390/s19122695
21. Cui F (2020) Deployment and integration of smart sensors with IoT devices detecting fire disasters in huge forest environment. Computer Communications 150:818–827 . doi: 10.1016/j.comcom.2019.11.051
22. Pei Y, Gan F (2009) Research on data fusion system of fire detection based on neural-network. Proceedings of the 2009 Pacific-Asia Conference on Circuits, Communications and System, PACCS 2009 665–668 . doi: 10.1109/PACCS.2009.134
23. Yao Y, Yang J, Huang C, Zhu W (2010) Fire monitoring system based on multi-sensor information fusion. 2010 2nd International Symposium on Information Engineering and Electronic Commerce, IEEC 2010 448–450 . doi: 10.1109/IEEC.2010.5533209
24. Xue CJ (2010) The road tunnel fire detection of multi-parameters based on BP neural network. CAR 2010 - 2010 2nd International Asia Conference on Informatics in Control, Automation and Robotics 3:246–249 . doi: 10.1109/CAR.2010.5456677
25. Dubey V, Kumar P, Chauhan N (2019) Forest Fire Detection System Using IoT and Artificial Neural Network. Springer Singapore
26. Li Z, Rizzo D, Hayden N (2006) Utilizing Artificial Neural Networks to backtrack source location. Proceedings of the iEMSs 3rd Biennial Meeting," Summit on Environmental Modelling and Software"
27. Kim H, Park M, Kim CW, Shin D (2019) Source localization for hazardous material release in an outdoor chemical plant via a combination of LSTM-RNN and CFD simulation. Computers and Chemical Engineering 125:476–489 . doi: 10.1016/j.compchemeng.2019.03.012
28. Qian F, Chen L, Li J, et al (2019) Direct prediction of the toxic gas diffusion rule in a real environment based on LSTM. International Journal of Environmental Research and Public Health 16: . doi: 10.3390/ijerph16122133
29. Lee D, Lim M, Park H, et al (2017) Long short-term memory recurrent neural network-based acoustic model using connectionist temporal classification on a large-scale training corpus. China Communications 14:23–31 . doi: 10.1109/CC.2017.8068761
30. Bengio Y, Simard P, Frasconi P (1994) Learning Long-Term Dependencies with Gradient Descent

is Difficult. IEEE Transactions on Neural Networks 5:157–166 . doi: 10.1109/72.279181

31. Hochreiter S (1997) Long Short-Term Memory. 1780:1735–1780
32. Bermúdez JD, Achanccaray P, Sanches ID, et al (2017) EVALUATION OF RECURRENT NEURAL NETWORKS FOR CROP RECOGNITION FROM MULTITEMPORAL REMOTE SENSING IMAGES Rio de Janeiro State University , Brazil. 800–804
33. Greff K, Srivastava RK, Koutnik J, et al (2017) LSTM: A Search Space Odyssey. IEEE Transactions on Neural Networks and Learning Systems 28:2222–2232 . doi: 10.1109/TNNLS.2016.2582924
34. McGrattan K, Hostikka S, McDermott R, et al (2017) FDS technical reference guide volume 1 : Mathematical Model
35. Kashef, A., Bénichou, N. & Lougheed G (2003) Numerical Modelling of Movement and Behaviour of Smoke Produced from Fires in the Ville-Marie and L . -H . – La Fontaine Tunnels : Literature Review Kashef , A .; Bénichou , N .; Lougheed , G . IRC-RR-141 September 2003. National Research Council of Canadá
36. Vermesi I, Rein G, Colella F, et al (2017) Reducing the computational requirements for simulating tunnel fires by combining multiscale modelling and multiple processor calculation. Tunnelling and Underground Space Technology 64:146–153 . doi: 10.1016/j.tust.2016.12.016
37. Yao Y, Cheng X, Zhang S, et al (2017) Maximum smoke temperature beneath the ceiling in an enclosed channel with different fire locations. Applied Thermal Engineering 111:30–38 . doi: 10.1016/j.applthermaleng.2016.08.161
38. Ingason H, Lönnermark A (2012) Heat release rates in tunnel fires: a summary. Handbook of Tunnel Fire Safety 309–328 . doi: 10.1680/htfs.41530.309
39. Hurley MJ, Gottuk D, Hall JR, et al (2016) SFPE handbook of fire protection engineering, fifth edition. SFPE Handbook of Fire Protection Engineering, Fifth Edition 1–3493 . doi: 10.1007/978-1-4939-2565-0
40. Danziger NH, Kennedy WD (1982) Longitudinal ventilation analysis for the Glenwood Canyon tunnels. In: Proc. of the 4th Int. Symp. Aerodynamics & Ventilation of Vehicle Tunnels
41. Zhong HY, Jing Y, Liu Y, et al (2019) CFD simulation of “pumping” flow mechanism of an urban building affected by an upstream building in high Reynolds flows. Energy and Buildings 202:109330 . doi: 10.1016/j.enbuild.2019.07.047
42. Mei SJ, Luo Z, Zhao FY, Wang HQ (2019) Street canyon ventilation and airborne pollutant dispersion: 2-D versus 3-D CFD simulations. Sustainable Cities and Society 50:101700 . doi: 10.1016/j.scs.2019.101700
43. Mcgrattan K, Mcdermott R (2015) Fire Dynamics Simulator User ’ s Guide (FDS Version 6.3.0)
44. Baum H, Mccaffrey B (1989) Fire Induced Flow Field - Theory And Experiment. Fire Safety Science 2:129–148 . doi: 10.3801/iafss.fss.2-129
45. Rawal A, Miikkulainen R (2016) Evolving deep LSTM-based memory networks using an information maximization objective. GECCO 2016 - Proceedings of the 2016 Genetic and Evolutionary Computation Conference 501–508 . doi: 10.1145/2908812.2908941
46. Tetko I V., Livingstone DJ, Luik AI (1995) Neural Network Studies. 1. Comparison of Overfitting and Overtraining. Journal of Chemical Information and Computer Sciences 35:826–833 . doi: 10.1021/ci00027a006

47. Park S, Kwak N (2017) Analysis on the dropout effect in convolutional neural networks. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 10112 LNCS:189–204 . doi: 10.1007/978-3-319-54184-6_12
48. Aksoy S, Haralick RM (2001) Feature normalization and likelihood-based similarity measures for image retrieval. Pattern Recognition Letters 22:563–582 . doi: 10.1016/S0167-8655(00)00112-4
49. Kumar J, Goomer R, Singh AK (2018) Long Short Term Memory Recurrent Neural Network (LSTM-RNN) Based Workload Forecasting Model for Cloud Datacenters. Procedia Computer Science 125:676–682 . doi: 10.1016/j.procs.2017.12.087
50. Komer B, Bergstra J, Eliasmith C (2014) Hyperopt-Sklearn: Automatic Hyperparameter Configuration for Scikit-Learn. Proceedings of the 13th Python in Science Conference 32–37 . doi: 10.25080/majora-14bd3278-006
51. Koekkoek EJW, Booltink H (1999) Neural network models to predict soil water retention. European Journal of Soil Science 50:489–495 . doi: 10.1046/j.1365-2389.1999.00247.x
52. Adamowski J, Karapataki C (2010) Comparison of multivariate regression and artificial neural networks for peak urban water-demand forecasting: Evaluation of different ANN learning algorithms. Journal of Hydrologic Engineering 15:729–743 . doi: 10.1061/(ASCE)HE.1943-5584.0000245
53. Yao K, Peng B, Zhang Y, et al (2014) Spoken language understanding using long short-term memory neural networks. 2014 IEEE Workshop on Spoken Language Technology, SLT 2014 - Proceedings 189–194 . doi: 10.1109/SLT.2014.7078572
54. Chollet F, others (2018) Keras: The python deep learning library. Astrophysics Source Code Library
55. Kavzoglu T, Mather PM (2003) The use of backpropagating artificial neural networks in land cover classification. International Journal of Remote Sensing 24:4907–4938 . doi: 10.1080/0143116031000114851
56. Alzubaidi L, Al-Shamma O, Fadhel MA, et al (2020) Classification of red blood cells in sickle cell anemia using deep convolutional neural network. Advances in Intelligent Systems and Computing 940:550–559 . doi: 10.1007/978-3-030-16657-1_51

Appendix

Figure A1 compares the temperature measured at the middle of the tunnel in 2 and 3D modellings when the fire location, HRR and wind equal to 64 m, 50 MW and 1 m/s, respectively. Roughly they coincide with each other though the temperature curve of 2D modelling shows a slightly larger fluctuation. Considering that the main goal of this paper is to demonstrate the use of AI method in the prediction of fire source in tunnel rather than precisely modelling the fire behavior, it is rational to utilize 2-D modelling in this study.

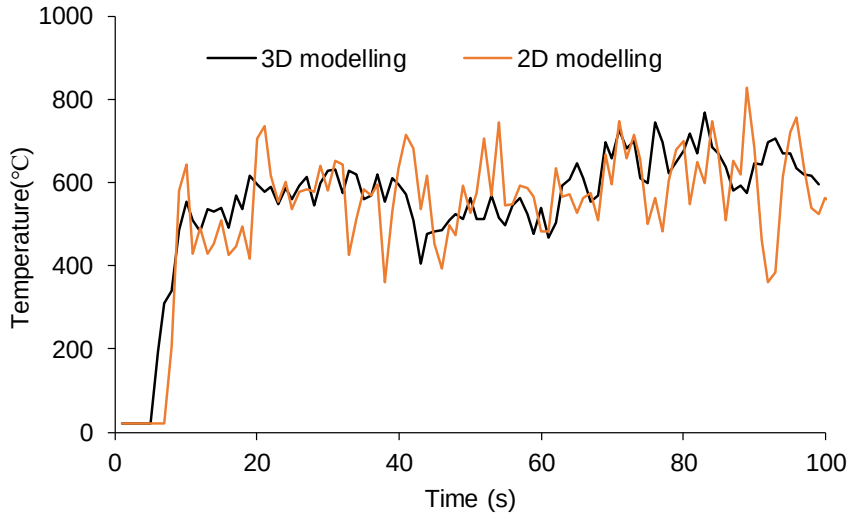


Fig. A1. Comparison of temperatures measured at the middle of the tunnel between 2 and 3D modeling.

Figure A2 shows that the temperature measured at left exit, middle and right exit of the tunnel, indicating that temperature varies periodically after around 30 s. The steady stage is assumed to be reached when the temperature measured at various locations of the tunnel varies periodically.

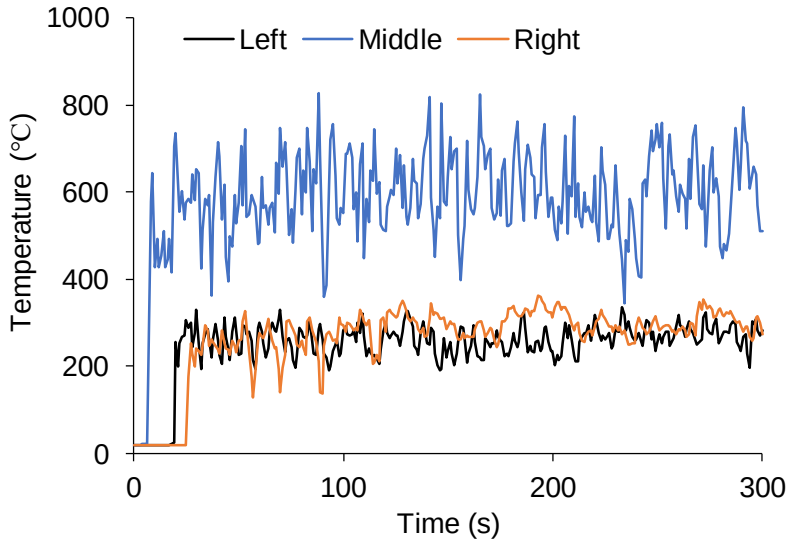


Fig. A2. Temperature measured at various locations when the fire location, HRR and wind equal to 64 m, 50 MW and 1 m/s, respectively.

Table 1. Modeled tunnel fire cases for sensor allocation optimization

Group	Case	Sensor location (m)	Sensor region (m)	Sensor separation (m)	Sample period (s)	Start of data collection (s)	Collection duration (s)	Interval of data collection (s)
	Benchmark	0	160	1	60	0	300	1
1 Sensor location	1	0	20	5				
	2	20	20	5				
	3	40	20	5				
	4	60	20	5				
	5	80	20	5				
	6	100	20	5				
	7	120	20	5				
	8	140	20	5				
2 Sensor region	9	75	10	5				
	10	70	20	5				
	11	60	40	5				
	12	50	60	5				
	13	40	80	5				
	14	30	100	5				
	15	20	120	5				
	16	10	140	5				
	17			5				
3 Sensor separation	18			2				
	19			10				
	20			15				
	21							
	22			40				
4 Sample length	23				2			
	24				5			
	25				15			
	26				20			
	27				25			
	28						60	
	29				20		60	
	30				30		60	
	31				40		60	
	32				50		60	
5 Start of collection	33					10	40	
	34					20	50	
	35					60	90	
	36					90	120	
	37					120	150	
	38					180	210	
	39					240	270	
6 Collection duration	40						15	
	41						20	
	42						90	
	43						120	
	44						180	

	45		240	
	46	20		2
	47	20		5
	48	20		10
	49	50	60	2
	50	50	60	5
	51	50	60	10
	52	50	60	15
	53	50	60	20
	54	50	60	30
	55	80	90	
	56	80	90	2
	57	80	90	5
7	58	80	90	10
Time	59	80	90	15
interval	60	80	90	20
	61	80	90	30
	62	80	90	40
	63	110	120	
	64	110	120	2
	65	110	120	5
	66	110	120	10
	67	110	120	15
	68	110	120	20
	69	110	120	30
	70	110	120	40
	71	110	120	50
	72	110	120	60