X. Wu, X. Zhang, X. Huang, F. Xiao, A. Usmani (2021) *A Real-time Forecast of Tunnel Fire Based on Numerical Database and Artificial Intelligence*, **Building Simulation**, doi: 10.1007/s12273-021-0775-x

# A Real-time Forecast of Tunnel Fire Based on Numerical Database and Artificial Intelligence

**Xiqiang Wu[1,2], Xiaoning Zhang[1], Xinyan Huang[1,*], Fu Xiao[1], Asif Usmani[1,2]**

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

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

*Corresponding author: xy.huang@polyu.edu.hk

## ABSTRACT

The extreme temperature induced by fire and hot smoke in tunnels threaten the trapped personnel and firefighters. To alleviate the potential casualties, fast while reasonable decisions should be made for rescuing, based on the timely prediction of fire development in tunnels. This paper targets to achieve a real-time prediction (within 1 s) of the spatial-temporal temperature distribution inside the numerical tunnel model by using artificial intelligence (AI) methods. A CFD database of 100 simulated tunnel fire scenarios under various fire location, fire size, and ventilation condition is established. The proposed AI model combines a Long Short-term Memory (LSTM) model and a Transpose Convolution Neural Network (TCNN). The real-time ceiling temperature profile and thousands of temperature-field images are used as the training input and output. Results show that the predicted temperature field 60 s in advance achieves a high accuracy of around 97%. Also, the AI model can quickly identify the critical temperature field for safe evacuation (i.e., a critical event) and guide emergency responses and firefighting activities. This study demonstrates the promising prospects of AI-based fire forecasts and smart firefighting in tunnel spaces.

**Keywords:** Tunnel fires; smart firefighting; critical event; CFD; deep learning; LSTM/TCNN

## Abbreviations

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

## 1. Introduction

Over the last few decades, the construction of underground space and tunnels has been a favorable option to meet the increasing demand for more efficient urban transportation system for megacities of a high population density (Dindarloo and Siami-Irdemoosa 2015). Although the probability of fire incidents happened in tunnels is smaller than those in open spaces (Ingason *et al.* 2015; Li and Ingason 2018; Carvel 2019), the absolute annual number of tunnel fire accidents is still very high, considering a large number of tunnels around the world and their high traffic densities. Statistics show that China alone had been 161 medium and large tunnel fire accidents from 2000 to 2016 (Ren *et al.* 2019). Moreover, once a tunnel fire occurs, it could be fatal and cause a catastrophic economic loss (Beard 2009; Beard and Carvel 2012; Casey 2020; Li and Liu 2020). In 1987, a catastrophic tunnel fire incident in Azerbaijan was induced by an electrical fault and finally caused 289 deaths (Haack 2002). In 2020, a fire accident occurred in the Samae 2 Tunnel, Korea, after the collision of dozens of tanks and trucks, which killed four people and injuring more than 40 others. The severe consequences of tunnel fire can attribute to challenging evacuation from poorly ventilated spaces with high-temperature and high-density of smokes and toxic gases. Furthermore, the rapid and complex development of fire inside the tunnel makes it difficult to predict and guide the rescuing and firefighting actions. Therefore, an accurate and timely prediction of tunnel fire is in urgent need (Beard 2009; Wu *et al.* 2020; Zhang *et al.* 2021).

Driven by the recent booming of artificial intelligence (AI) and high-performance computing, numerous studies have proposed AI-based fire detection and fire risk assessment, particularly in wildland fires (Cohen *et al.* 1989; Akhloufi *et al.* 2018; Sayad *et al.* 2019). Dubey *et al.* (2019) proposed a simple 2-layer feed-forward fully connected neural network (NN) for the detection of wildland fires, based on the input of wireless sensor networks. Massive images of smoke and flame have also trained via deep learning to improve the accuracy of fire detection and reduced the rate of false alarm (Cao *et al.* 2019; Govil *et al.* 2020). In particular, the convolutional neural networks (CNNs) has also been used in fast pinpointing the wildfire location (Ghoreishi 2019). Based on massive geographic information data and academic codes, the data-driven forecast has also been demonstrated and applied in wildland fire by applying the algorithm of logistic regression (LR) (De Vasconcelos *et al.* 2001), CNN (Hodges and Lattimer 2019), multilayer artificial neural network (ANN) (Zhai *et al.* 2020), and support vector machine (SVM) (Ngoc Thach *et al.* 2018).

As to structural fire, sensor data from monitoring smoke, CO, and heat are often utilized for the detection of fire using multiple types of NN (Pei and Gan 2009; Xue 2010; Yao *et al.* 2010), because of the low cost and maturity of these sensor technologies. Camera images are more difficult for fast processing, and not available in many existing infrastructures, and more importantly, but their visibility quickly decays inside the structure in case of fire (Wu *et al.* 2020). Xue (2010) proposed a classic three-layer ANN for detecting the occurrence of fire in tunnels based on sensor data of temperature and densities of smoke and CO. High accuracy and generalization ability were achieved with this method, although this NN was trained using numerical simulations due to the lack of sufficient on-site data. Hodges *et al.* (2018, 2019) adopted a transpose convolutional neural network (TCNN) to predict the steady-state temperature distribution inside a compartment, based on training a numerical database. While in this study, it was assumed that the information of room geometry, ventilation, fire size, and fire location was available, which is quite unrealistic.

So far, very few studies have realized the fire forecast in the urban fire environment. Previously, the

FireGrid framework was proposed to collect real-time sensor data and predict the evolution of compartment fires based on a zone model (Han *et al.* 2010). Carvel *et al.* (2005) employed Bayesian methods to forecast the probability of downstream flame extension length. Ji *et al.* (2018) proposed inverse modeling with an ensemble Kalman filter to predict the temperature distribution and smoke layer thickness in tunnels. Dexters *et al.* (2020) proposed an LR model to predict the occurrence of flashover in a compartment utilizing the information of fire intensity, fire duration, and the thickness of combustible materials from past experiments. In our previous work, we proposed a long short-term memory (LSTM) model to successfully identify the size and location of fire inside the tunnel, based on the training of 100 tunnel-fire numerical simulations (Wu *et al.* 2020); and predict the required critical ventilation velocity by training an organized fire-test database (Zhang *et al.* 2021).

In this work, we aim to predict the temperature distribution inside tunnels in advance based on limited temperature sensor data. First, a numerical database of 100 tunnel fire scenarios under various fire locations, fire sizes, and ventilation conditions was established. The temperature at multiple locations near the ceiling of the tunnel was recorded during the fire development to mimic the heat sensor in the real tunnel. Then, the sensor data and thousands of temperature-field images were post-processed to form a sizeable numerical database for the training of LSTM and TCNN layers. Finally, the AI model can achieve a fast prediction (within 1 s) of the tunnel temperature distribution 60 s afterward under various fire scenarios.

## 2. Methodology

### *2.1. Background of AI algorithms*

The proposed model consists of two main types of algorithms, i.e., LSTM and TCNN. In this section, the fundamental concepts and architectures of these two algorithms are briefly introduced.

**LSTM model**. The recurrent neural network (RNN) is designed to extract the feature hidden in sequential data (Chehreh Chelgani *et al.* 2018). The data chain is imported into the RNN model sequentially, and certain data can then be reserved inside the model. This type of model has been widely adopted in speech recognition (Lee *et al.* 2017). However, RNN cannot accurately figure out the information of long sequences (Bengio *et al.* 1994). Then, LSTM, as a special type of RNN introduced by Hochreiter (Hochreiter 1997), effectively tackles long-term dependency problems. After importing a piece of data, useful information can be remembered and translated for designed output. A more detailed structure of the network can refer to our previous work (Wu *et al.* 2020).

**CNN**. The CNN algorithms have been widely adopted in many fields related to computer vision, such as image recognition, analysis, classification, and recommendation (Hayou *et al.* 2019). Compared with convolutional ANN, CNN is more efficient due to its distinguishing feature of shared weight. Technically, the input images pass through a series of convolutional, pooling, and fully-connected layers defined in a CNN model (He *et al.* 2016). Fig. 1(a) shows a typical process of extracting features between convolutional layers. Often the volume of the input (Layer 1) is larger than that of the output (Layer 2). The value of each neuron on Layer 2 is computed by performing a dot product between the weights (filter layer) and the corresponding local spatial region on Layer 1. The values on filter layers will be updated during the training process.

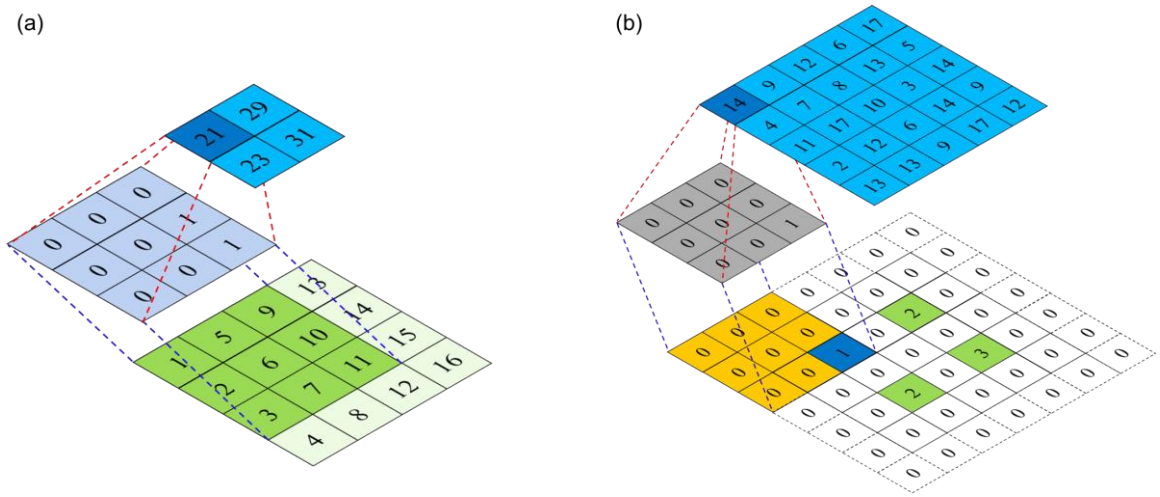


**Fig. 1.** Data transfer between (a) convolutional layer of CNN, and (b) transposed convolutional layers of TCNN.

**TCNN.** As the decoder of CNN, the TCNN behaves in the opposite direction to transform simple input data into higher-dimensional space (Dumoulin and Visin 2016). Fig. 1(b) depicts the data transformation between transposed convolutional layers. As shown, the calculation procedure is similar to that of CNN layers, while after padding with zero values, a larger size of output (Layer 2) is generated compare with that of the input (shaded values in Layer 1). In the current study, the TCNN architecture is adopted to generate images showing the temperature distribution in tunnels.

### *2.2. CFD modeling of tunnel fire*

Plenty of real-scale and reduced-scale tunnel fire tests have been conducted to quantify the key parameters related to the fire safety of the tunnel, including maximum temperature, smoke layer, and critical ventilation velocity (CVV) (Beard 2009; Li and Ingason 2018; Zhang *et al.* 2021). Most of these studies installed many temperature sensors (e.g., thermocouple trees) and laser scanners for visualizing the smoke layer, while it is practically difficult to quantify the full temperature fields of a tunnel fire. Moreover, the thick dark smoke inside an under-ventilated tunnel (Hu *et al.* 2010) quickly lowers the visibility. Thus, even if visual and infrared cameras are pre-installed, it is difficult to monitor the tunnel fire continuously (Gong *et al.* 2016). Recently, Computational Fluid Dynamics (CFD) has been widely applied to simulate tunnel fires. Particularly, the open-source program of Fire Dynamics Simulator (FDS) developed by NIST (Mcgrattan and Mcdermott 2015) has been extensively verified and validated and used as a standard tool for the performance-based fire safety design of the tunnel. Moreover, the CFD modeling results can provide much more detailed temperature information than any experiment, which is ideal for the verification of the proposed AI method for the fire forecast.

Fig. 2 shows a representative real-scale tunnel having a length of 160 m, a width of 6 m, and a height of 6 m, which mimics the full-scale road tunnel at Sichuan Fire Research Institute (SCFRI). The 3-D numerical tunnel model is constructed in FDS 6.7 (Mcgrattan and Mcdermott 2015), which was upgraded from the previous 2-D model (Wu *et al.* 2020). The inner surfaces of the tunnel were assumed as perfectly smooth. Thermally, the default inert wall and an ambient temperature of 20 °C were assumed. The left entrance of the tunnel was set as an open boundary, and the right exit was also set to open unless there was ventilation. The modeled fire with a fixed HRR was a rectangular burner, which was 3 m along the *x*-axis, 6 m along the *y*-axis, and 0.5 m above the ground to represent a vehicle fire.

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 $\delta x$ is the nominal length of the cell edge (Mcgrattan and Mcdermott 2015). Studies showed that accurate predictions of airflow velocity and temperature could be achieved if $D^*/\delta x$ was in a range of 4-16 in regions near the fire source(Mcgrattan and Mcdermott 2015). The calculated maximum cell size increased from 0.2 m to 0.54 m with the increase of HRR from 5 MW to 80 MW. For all the models, the mesh size of 0.2 m is adopted along *x*-, y-, and *z*-axises, which also coincide with the settings by other researchers (Ji *et al.* 2015).

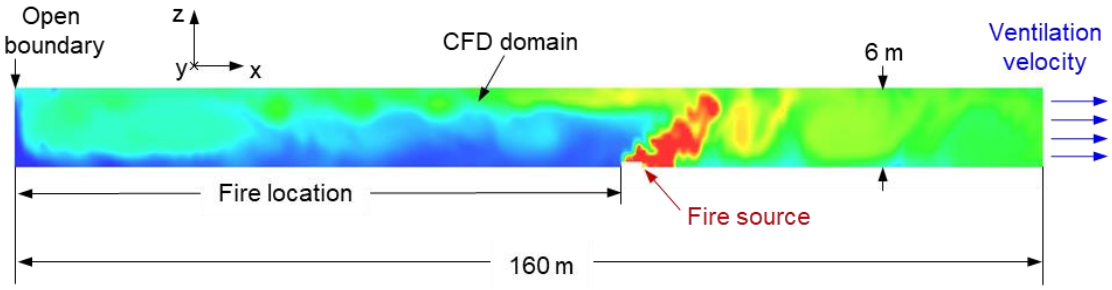


**Fig. 2**. Key parameters for the computational domain of the tunnel fire model (not to scale).

Three key parameters, namely, heat release rate (HRR), fire location, and ventilation velocity, were varied in the model to simulate different fire scenarios.

(1) **HRR** is one of the most significant fire parameters (Babrauskas and Peacock 1992), and for tunnel fire, the primary fuel is the vehicle. The HRR of the vehicle is affected by the vehicle dimension, type, separation, as well as ventilation conditions (Ingason and Lönnermark 2012; Hurley *et al.* 2016; Li and Ingason 2018; Sun *et al.* 2020). Ingason and Lönnermark (Ingason and Lönnermark 2012) summarized that the peak HRRs are 1.7-4.6 MW for a passenger car, 29-35 MW for a bus, and 60-200 MW for a heavy-duty vehicle or a tanker truck, respectively. Therefore, five constant HRR values, i.e., 5 MW, 10MW, 20 MW, 50 MW, and 80 MW, are set for the numerical models.

(2) **The location of fire** plays a key role in the temperature distribution inside the tunnel, but it is difficult to locate in cases where large quantities of smoke are produced. A survey (Nævestad and Meyer 2014) showed that most of the fire incidents occurred near the middle of the tunnel, while only small amounts of them happened near the exit or entrance zones. To cover different possible tunnel fire scenarios, different fire locations should be considered. Specifically, five fire locations are defined relative to the tunnel length of 0.1, 0.2, 0.4, 0.6, and 0.8, that is, 16 m, 32 m, 64 m, 96 m, and 128 m from the left entrance (Fig. 2).

(3) **The smoke ventilation capability** remains the central issue for tunnel fire safety. CVV is defined as the minimum longitudinal ventilation velocity that prevents the upstream movement of the combustion product from fire and the back-layering (Oka and Atkinson 1995). In this study, a simple but classical longitudinal ventilation is assumed to be mounted at the right-hand exit to extract smoke from the tunnel. This is realized by defining a constant velocity across the whole tunnel section. Given that a CVV of 2.8 m/s is required to prevent smoke back-layering from an 80 MW fire (Danziger and Kennedy 1982; Ingason *et al.* 2015; Zhang *et al.* 2021), four different ventilation speeds of 0 m/s, 1 m/s, 2 m/s, and 4 m/s, were modeled. Overall, 5 fire locations, 5 HRRs and 4 ventilation conditions constitute 100 (= 5 × 5 × 4) fire scenarios.

For all the cases, 32 sensor devices 0.5 m below the ceiling were evenly distributed along the tunnel

to mimic thermocouples (i.e., one sensor every 5 m) and record the gas temperature every 1 s, which is sufficient to describe the temperature evolution inside the tunnel. A temperature-contour slice was defined at the mid-width location $y = 3$ m to monitor the spatial variation of temperature. This definition enables the visualization of the temperature contour animation using the visualizing tool SmokeView (Forney 2010). Each tunnel fire scenario was calculated for 10 min (600 s), which was long enough for all fire scenarios to reach the steady stage in the previous work (Wu *et al.* 2020).

### *2.3. Dataset generation*

We constructed a large database containing the temperature measurements of 32 thermocouples every 5 m (sensor data) and the temperature field of the central plane (slice images) every 1 s. Fig. 3 presents the post-processing process of numerical results to form the database. The temperature sensor data were recorded into files in the format of Comma Separated Values (CSV) at an interval of 1 s continuously from the ignition of fire ($t = 0$). In other words, each temperature datasheet is a matrix of 32 (No. of the sensor) × 600 (s). Then, the first 9 min of datasheets were chosen to form 505 data groups with a temporal length of 30 s as the input, i.e., 0-30 s, 1-31 s, 2-32 s, …, 504-534 s. These 505 data samples multiplying 100 scenarios form a database of 50,500 samples.

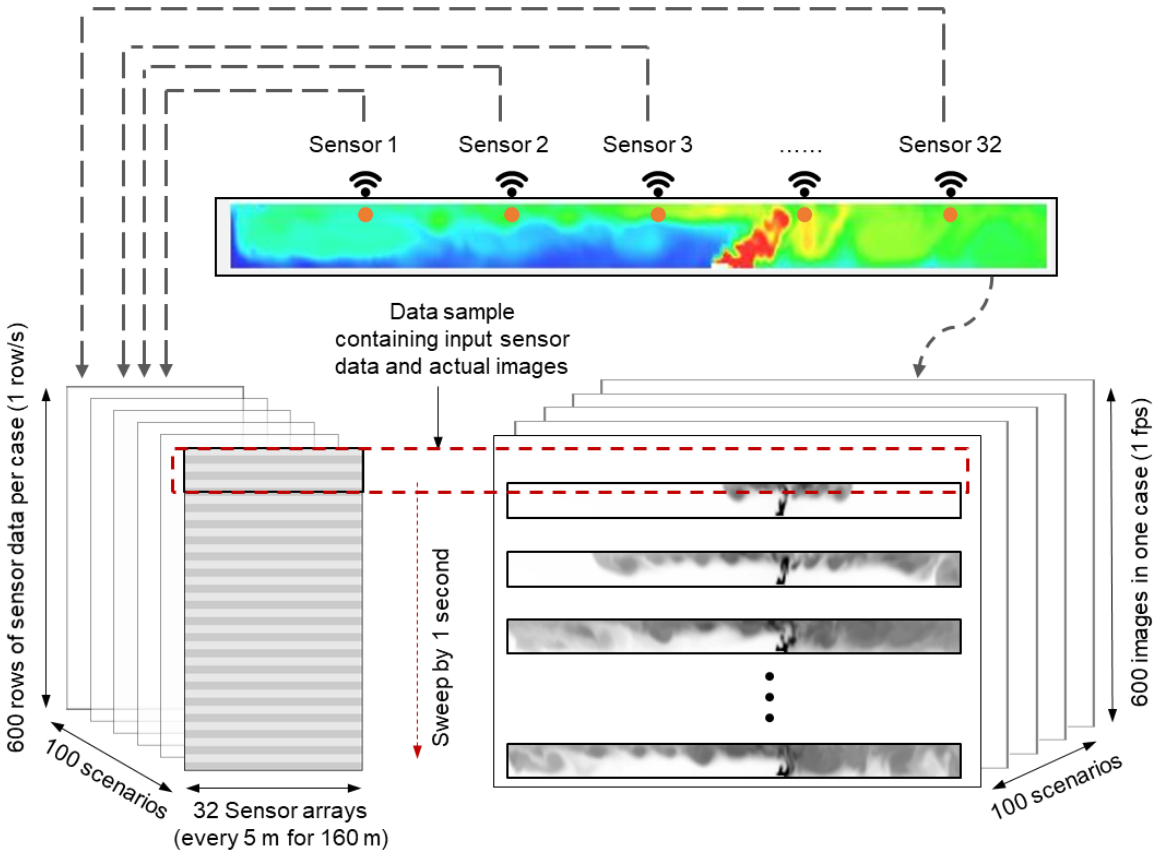


**Fig. 3.** Generation of the training database

The temperature field can be output as the grey or color image through the default software Smokeview or an in-house code. For the greyscale image, essentially, the value of temperature in each pixel is linearly normalized from 0 (white for the ambient temperature 20 ℃) to 255 (black for the highest temperature, depending on the HRR). If the upper limit of temperature is set higher, such as

1,000 ℃ for flame, the greyscale contrast in the low-temperature regions far away from the fire source becomes too low to recognize accurately. On the other hand, converting the temperature field into a color image enhances the image contrast, but it triples the database and requires a much longer training time. Three imaging mapping methods are tested in the preliminary training.

(i) The greyscale images of the temperature field in the central plane were exported at a constant rate of 1 fps with a resolution of 600 × 30 pixels for the first 9 min, i.e., 505 original greyscale images.

(ii) The exported greyscale images were averaged by their respective antecedent and subsequent 5 images, that is, temporally smoothed in a period of 10 s. The averaged image hides the unnecessary details of the temperature distribution, such as the turbulent vortex structure.

(iii) Color images were exported at the same rate of 1fps and temporally smoothed in a period of 10 s.

After training, the forecast quality of mapping Method (i) is found to be quite low, probably because the training is influenced by the periodical vortex motion. The forecast quality of methods (ii) and (iii) are much better, but the training time is much longer for color images due to the larger number of coefficients correlated to 3 channels of RGB images. Thus, method (ii) is chosen. The averaged images with a resolution of 600 × 30 pixels are then linearly scaled into 432 × 30 pixels, which matches the size of the output layer of the proposed AI model. Fig. A1 of the Appendix shows an example of images before and after scaling. The good similarity indicates the feasibility of scaling.

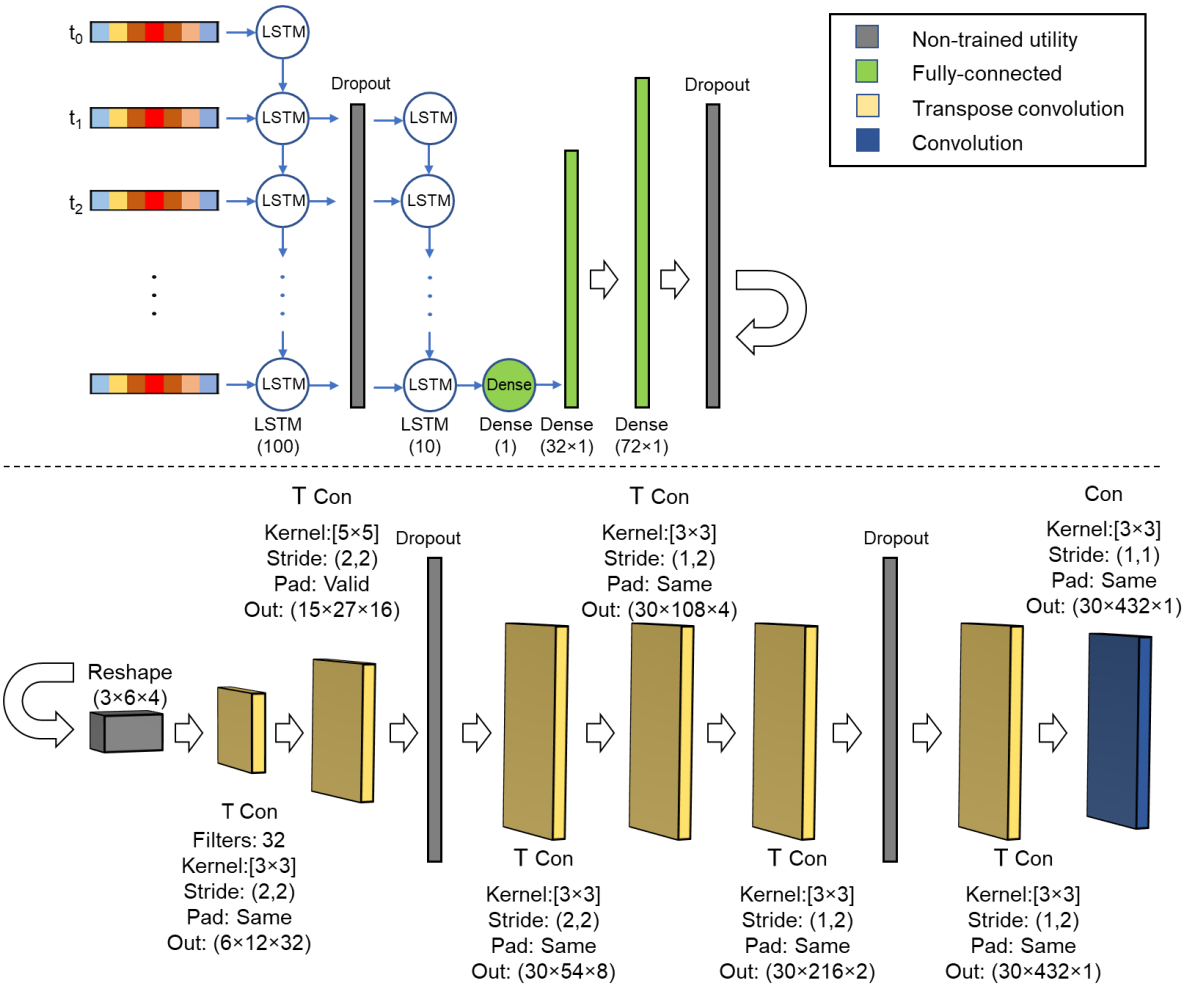


**Fig. 4.** Architecture of the proposed AI model.

For supervised learning algorithms, true values of the output have been already known. To evaluate the forecast quality, the predicted temperature field 60 s after the collection of the sensor data was compared with the original CFD modeling results. Then, the samples in the database were randomly disordered using the function of "random.shuffle" available in numpy library, and then they were split into 60% (the training dataset), 20% (the validation dataset), and 20% (the test dataset), 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 quantify the forecast quality.

### *2.4. Proposed framework*

Fig. 4 describes the details of the proposed framework composed of LSTM, dense, and TCNN layers. The input layer of the model sequentially receives a vector of sensor temperature data of each training sample. The features hidden in the input data are extracted by 2 LSTM and 1 dense layer, and then they are enriched by another 2 dense layers, and finally, high-dimension output, i.e., images showing the spatially varied temperature distribution 60 s in advance, is exported after passing through 6 TCNN layers and 1 CNN layer.

The architecture of the LSTM layers is much similar to that of our previous model (Wu *et al.* 2020). The input layer read a data of 32 (No. of the sensor) × 30 (s) (length of a data sample). The following 2 fully connected dense layers having more neurons are used to enrich the information extracted by the LSTM layers. The reshape layer packs up the flow of data into a box having a size of (3, 6, 4), which represents the height and width of a matrix and the number of stacked matrixes, respectively. The same size format applies to the following TCNN and CNN layers. After up-sampling through for 6 TCNN layers 1 CNN layer with various padding and stride parameters, a greyscale image with dimensions of (30, 432) illustrating its height and width can be generated. The generated image can then be compared with the actual one, and their difference will be minimized during the training process.

It should be noted that severe overfitting problems could be induced if too detailed characteristics or noises from the input data are learned during the fitting process. The performance of the model would then be too perfect on trained data, meanwhile unsatisfactory on untrained data (Tetko *et al.* 1995). To lower the occurrence of overfitting, 4 dropout layers with a rate of 0.1 were allocated to the model.

### *2.5. Implementation details*

Deep learning algorithms have been reported to achieve satisfactory results in many areas. However, unlike some machine learning methods such as LR and decision tree whose mechanisms can be easily interpreted (Chen 2011), the models adopted deep learning algorithms are regarded as "black box" (Wu *et al.* 2013), and their parameters need to be verified to gain a good generality.

Generally, better performance of deep learning can be achieved with more coefficients, while larger databases and training resources are unavoidable. Thus, the amount of these coefficients should be determined by balancing the training efficiency and prediction accuracy. The volume of the weights largely depends on the number of layers, hidden states, neurons, and filters in the proposed model, LSTM layers, dense layers, and TCNN layers, respectively. Though the coefficients attached to each layer can be automatically updated through training, attention still has to be paid to the settings of the model for achieving high performance.

First, various standard methods (Glorot and Bengio 2010), i.e., "he_uniform," "he_normal," and "glorot_uniform," can be adopted as an initializer to assign the initial values to the coefficient. Then,

the weighted layer output can be linearly or nonlinearly transformed with activation functions like "tanh," "relu," and "sigmoid" (Arora *et al.* 2018). For the output of the end layer, suitable loss functions, options including "mean squared error," "mean absolute error," and "binary cross-entropy" (Sun *et al.* 2017), should be defined to be compared with the actual value. The error between the predicted and true values is then minimized by optimizer function, e.g., "adam" and "RMSprop" (Morales *et al.* 2018). Here, "he_uniform," "tanh," "binary cross-entropy," and "adam" are adopted for the AI model after a verification study on these set parameters. More details about the verification are given in Section 3.1.

Fig. 5 summarizes the detailed training process of the proposed AI model. It is crucial to ensure that the samples in the training dataset, validation dataset, and test dataset are representative of the overall database. In this study, all samples were randomly divided into these three datasets. It is reported that the feature having a broad range would be difficult to be trained (Aksoy and Haralick 2001). Thus, the pixel values in images are normalized to the range between 0 and 1 with the "MinMaxScaler" function (Komer *et al.* 2014). The normalized samples are then used to train the proposed AI model for a large number of epochs, which is the number of times all of the training samples are used once. A preliminary study showed that 1,000 epochs would be enough to obtain a minimum loss while not causing overfitting. Generally, 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 AI model is trained. In this study, 30,300 (i.e., 60% of 50,500) samples were trained in a batch on a server having 32 cores and 124 GB physical memories, which lasted for about 9 h.

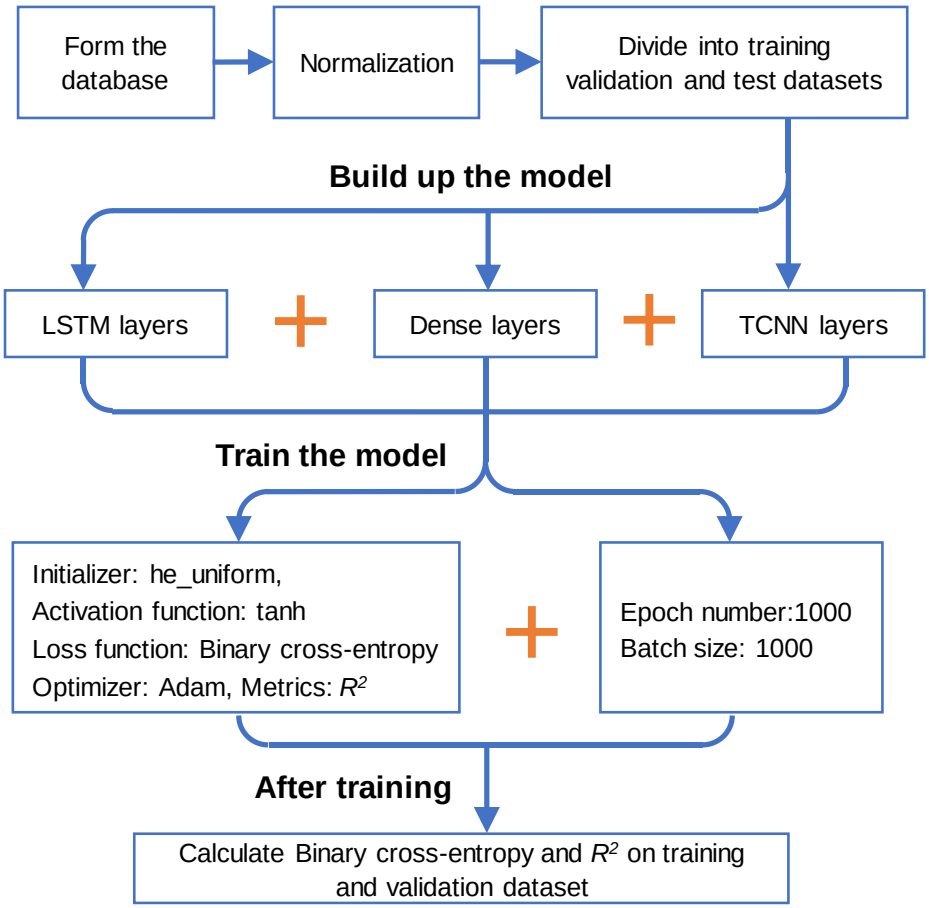


**Fig. 5.** Flow chart for model training: normalization and divide the database; build up the model; train the model; and calculate the metrics.

## 3. Results and discussions

### *3.1. Model Verification*

The impacts of these properties on the prediction accuracy of the model were verified and shown in Fig. 6. To save computing time, these preliminary cases were trained for 100 epochs. The performance of the trained model can be evaluated with the defined loss functions or a judging metric such as the coefficient of determination $R^2$. Compared with loss functions, one distinct characteristic of $R^2$ is its scale independence (Koekkoek and Booltink 1999; Adamowski and Karapataki 2010), so it is adopted in this study to evaluate the model performance.

Fig. 6 shows that both the initializer and loss functions have an insignificant influence on model performance. For the loss function (Fig. 6c), after a stage where almost the same accuracies attained for models adopting different loss functions, the "binary cross entropy" gained a slightly higher score than the loss functions of "mean squared error" and "mean absolute error."

The activation function affects both the training efficiency and final accuracy. The model adopted the function of "tanh" attained a high accuracy of prediction immediately after training, indicating that "tanh" is more suitable than "relu" (Fig. 6b). Regarding the optimizer, "adam" achieved a higher accuracy after an initial lower performance (Fig. 6d). The difference in model performance adopting these various properties revealed the importance of the model verification.

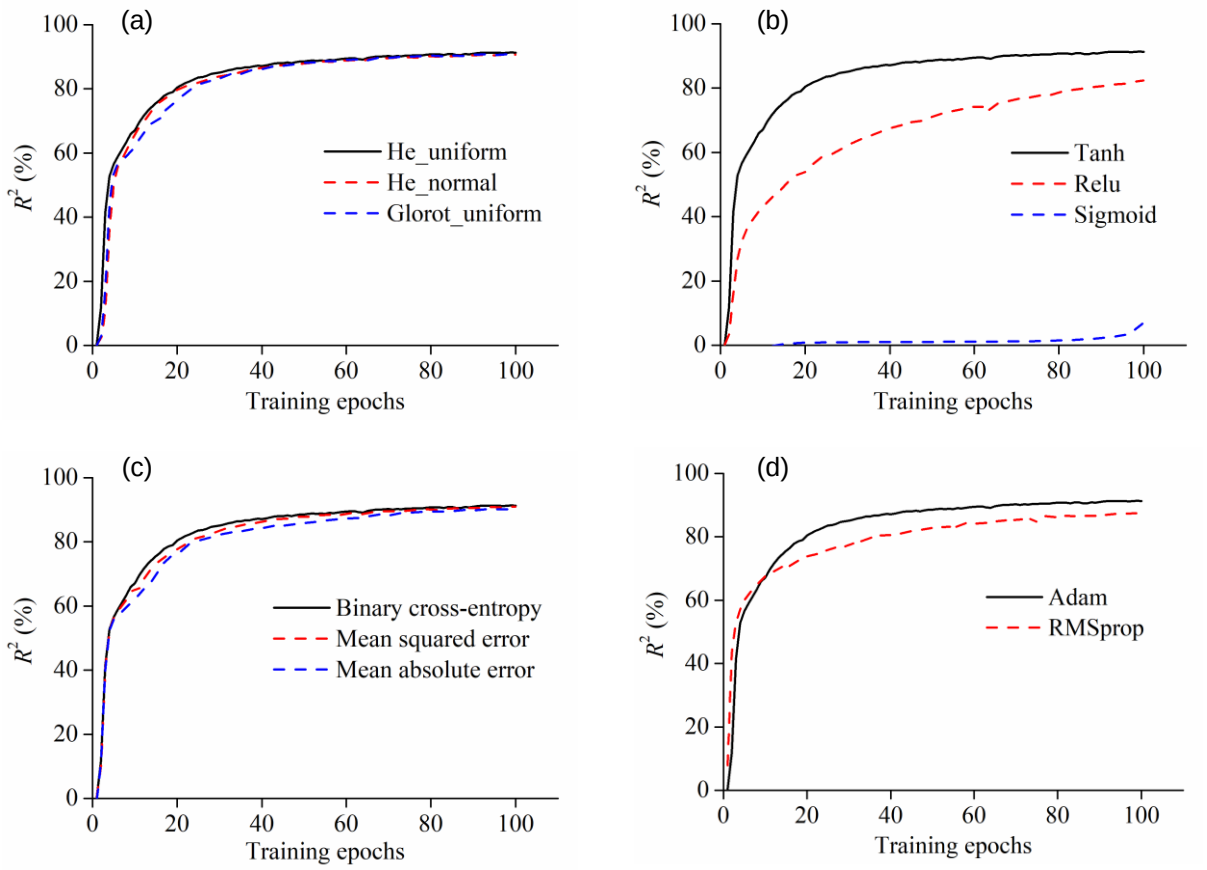


Fig. 6. Impacts of (a) initializer, (b) activation function, (c) loss function and (d) optimizer on the prediction accuracy.

### *3.2. Forecast quality*

Fig. 7 shows (a) the loss and (b) $R^2$ on the training dataset and validation dataset evolving with training epochs. The loss decreases with the number of training epochs on the training dataset, as expected. The training accuracy is extremely high at the starting of training, while no apparent minimization is observed after 50 epochs, indicating that the model has almost converged, and the pre-set training epoch number of 1,000 is sufficiently large.

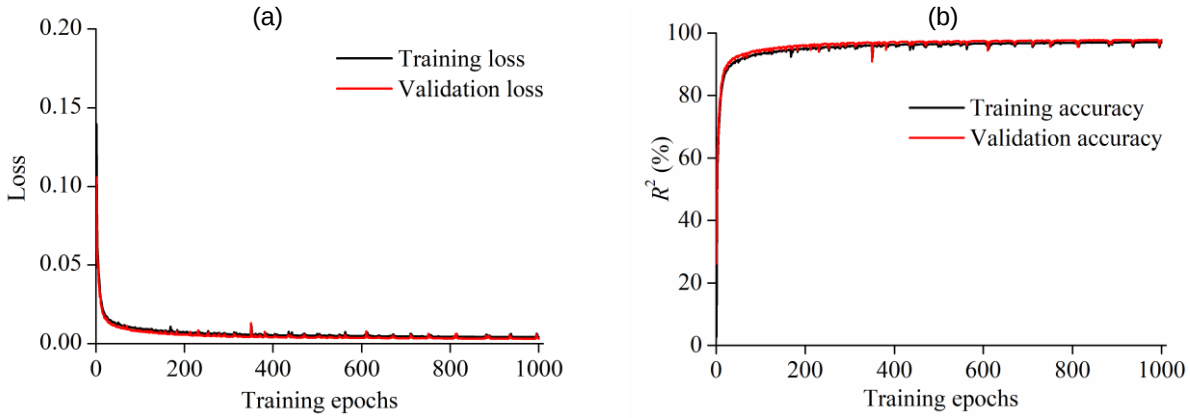


**Fig. 7.** The evolvements of (a) loss and (b) $R^2$ during the training process

The coefficient of determination $R^2$ on the training dataset increases quite fast at the beginning. Finally, $R^2$ on both training validation dataset converges to 97%. The high score approaching 100% indicates the high performance of the trained model on the prediction of temperature distribution 60 s after the collection of sensor data. It is worth noting that the loss and accuracy on the validation dataset are comparable to those on the training dataset, meaning that the overfitting problem was not caused. The setting of dropout layers causes a smaller loss and higher accuracy in validating the dataset. The slight fluctuation observed on loss accuracy is a common consequence of machine learning (Kavzoglu and Mather 2003; Alzubaidi *et al.* 2020).

The forecasting accuracy during a fire scenario, where fire location, fire size, and ventilation are 32 m, 20 MW, and 1 m/s, is presented in Fig. A2 of the Appendix and Video S1. High accuracy is achieved from the start of the fire, which could be due to the setting of a constant fire source without considering the growth or spreading of the fire. Note that the current database (including only constant HRRs), so that the prediction of a transient fire HRR by AI will have lower accuracy and a time delay, as illustrated by Fig. A3 in Appendix. To achieve an accurate real-time prediction for a more realistic fire scenario in the future, a larger database, including more fire cases with transient HRRs, is required.

### *3.3. Spatial varied temperature distribution*

This section quantifies the quality of the forecast and the difference between the forecasted and true temperature fields. Figs. 8-10 show the current temperature field (at 30 s, temperature profile as input) and the forecasted temperature field at 90s, for cases of various HRR, fire location, and ventilation velocity, respectively. Note that once inputting the data, it took less than 1 s for the AI model to output the forecasted temperature field, so it is a dynamic process and almost real-time. To avoid misunderstanding on the images of temperature distributions with those of smoke, the original grey images were converted to colorful images for visualization. Specifically, white (RGB value 255, 255,2

55) representing a temperature of 20 ℃ and black (RGB value 0, 0, 0) representing a temperature of 250 ℃ are respectively mapped to blue (RGB value 0, 0, 255) and red (RGB value 255, 0, 0). Fig. A1 in the Appendix shows a comparison of the images before and after color mapping.

Fig. 8(a-e) compare the true and forecasted temperature fields at 90 s, where the data of 32 temperature sensor at the first 30 s are used as the input for the AI model. For all cases, the fire is located 64 m from the left boundary, the ventilation velocity is 2 m/s, and the fire HRR varies from 5 MW to 80 MW. As shown, the AI model is able to predict the hotter regions near fire sources and tunnel ceiling, as well as the cooler regions far away from the fire. Video S2 compares the AI prediction with simulated temperature distribution evolving with time.

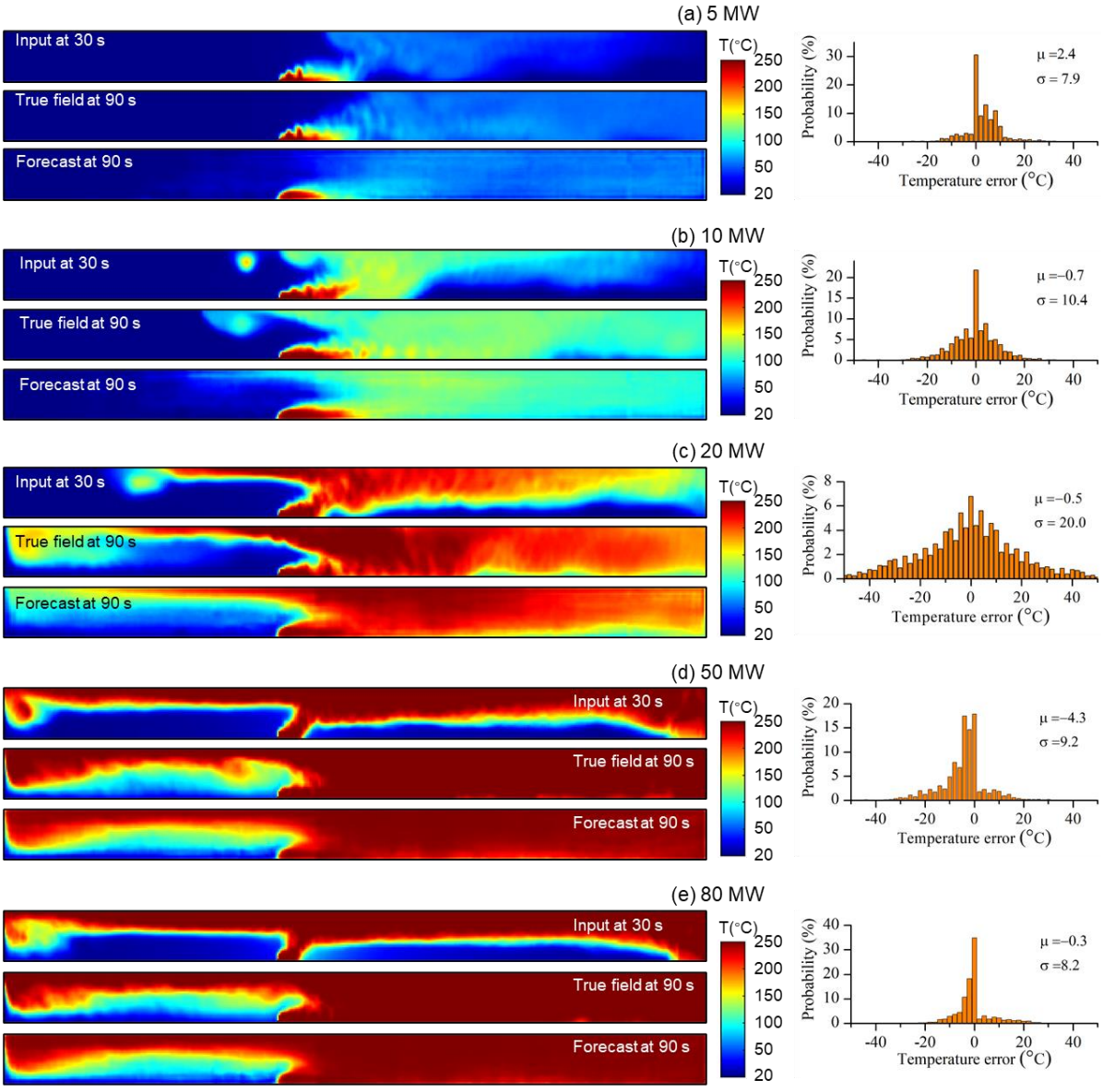


**Fig. 8.** Comparison between the true and forecasted temperature fields at 90 s, where inputs are 32 temperature sensor data of the first 30 s (left); and the deviation of forecasted temperature (right), where the fire location is 64 m from the left boundary, the ventilation velocity is 2 m/s, and the fire HRR is (a) 5 MW, (b) 10 MW, (c) 20 MW and see Video S2, (d) 50 MW and (e) 80 MW.

To quantify the quality of the forecast, the right column of Fig. 8 shows the distribution of the difference between the simulated and forecasted temperature field. The mean value of temperature difference ($\mu$) and standard deviation ($\sigma$) of the temperature difference are roughly within ±5 ℃ and 20 ℃, respectively. $\mu > 0$ (or $\mu < 0$) means the predicted temperature field is higher (or lower) than the true field. The larger $\sigma$ of 20 ℃ for the 20 MW case in Fig. 8b could be attributed to the inaccurate prediction of downstream temperature near the fire source. This indicates that the extreme temperature near the fire source is harder to be learned by the AI model due to the lack of expert knowledge. This prediction error is acceptable, considering the large variety of fire-induced environment temperature inside the tunnel in practice. The current temperature distribution and the sensor data at 30 s are also shown for comparison. The difference in temperature distribution between current and 60 s afterward indicates the importance of forecasting.

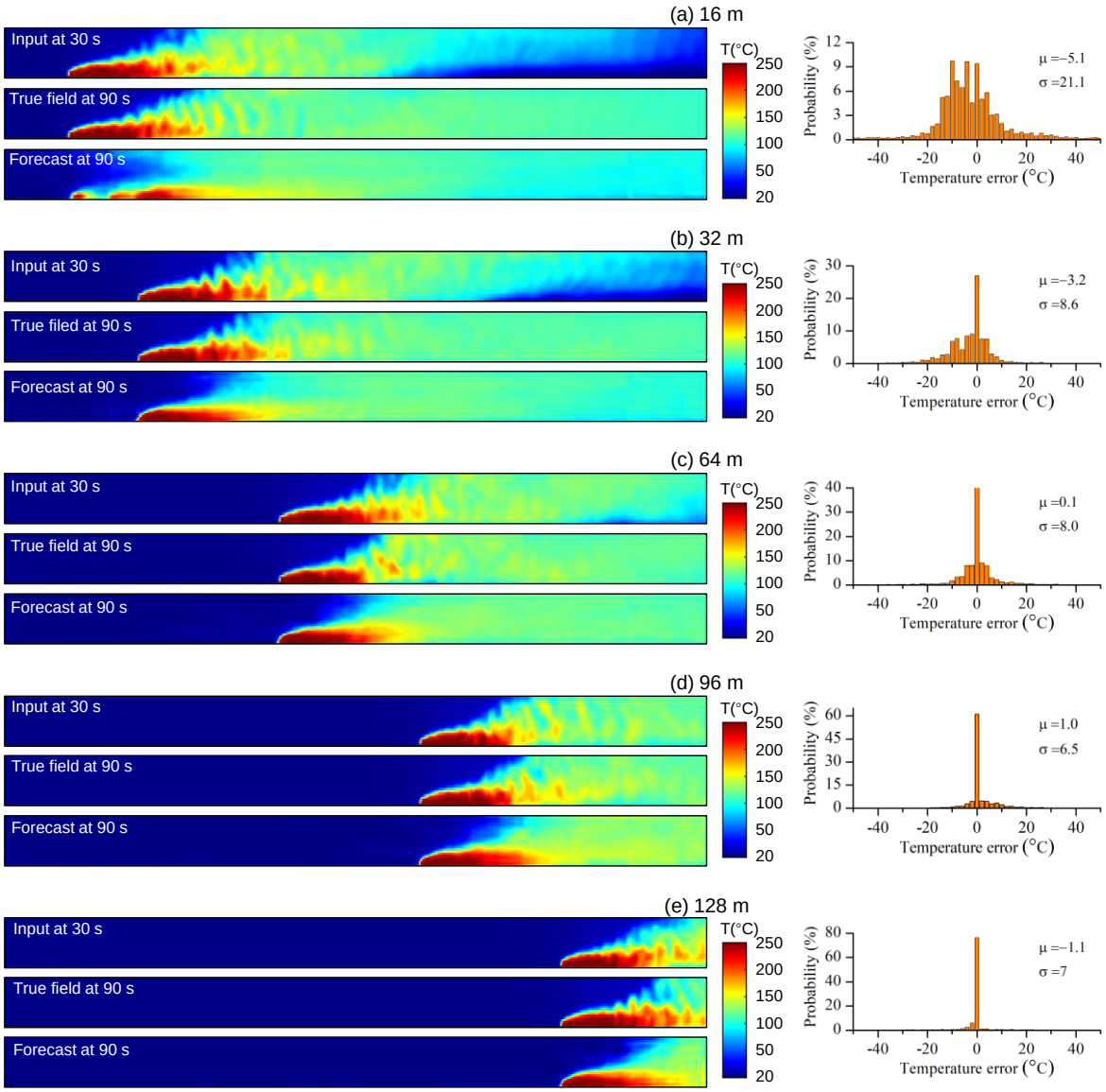


**Fig. 9.** Comparison between the true and forecasted temperature fields at 90 s, where inputs are 32 temperature sensor data of the first 30 s (left); and the deviation of forecasted temperature (right), where the fire HRR is 50 MW, the ventilation velocity is 2 m/s, and the fire location is (a) 16 m, (b) 32 m, (c) 64 m, (d) 96 m and (e) 128 m from the left boundary.

A similar phenomenon that higher temperature occurred near the fire location can be well predicted for cases with various fire locations (Fig. 9) and various ventilation conditions (Fig. 10). The mean values of the temperature difference ($\mu$) also range within ±5 ℃, while the deviation ($\sigma$) generally decreases as the distance of fire source moves closer to the ventilation side (Fig. 9) and as the ventilation speed increases (Fig. 10). This could be attributed to that it is easier for the temperature field to reach a steady stage under a better ventilation condition.

A fire scenario with a larger HRR of 130 MW, which out of the scope the AI model has learned, is modeled to further check the forecasting capability of the proposed model. Fig. A4 in Appendix gives the comparison between the predicted and forecasted temperature distribution in this case. The mean value and the standard deviation of the temperature difference are around -9.7 ℃ and 21.8 ℃, respectively. High forecasting accuracy of 97% is achieved throughout the fire. These good performances demonstrate the capacity of the AI model even for unseen fire scenarios. It is worth noting that modeling a case of fire scenario in the current study for 600 s using FDS on a single core requires around 24 hours. In contrast, it needs only 3.4 ms for the trained model to give a prediction on the temperature distribution. The negligible computing time demonstrates the feasibility of adopting a well-tuned AI model in guiding emergency actions during a fire incident.

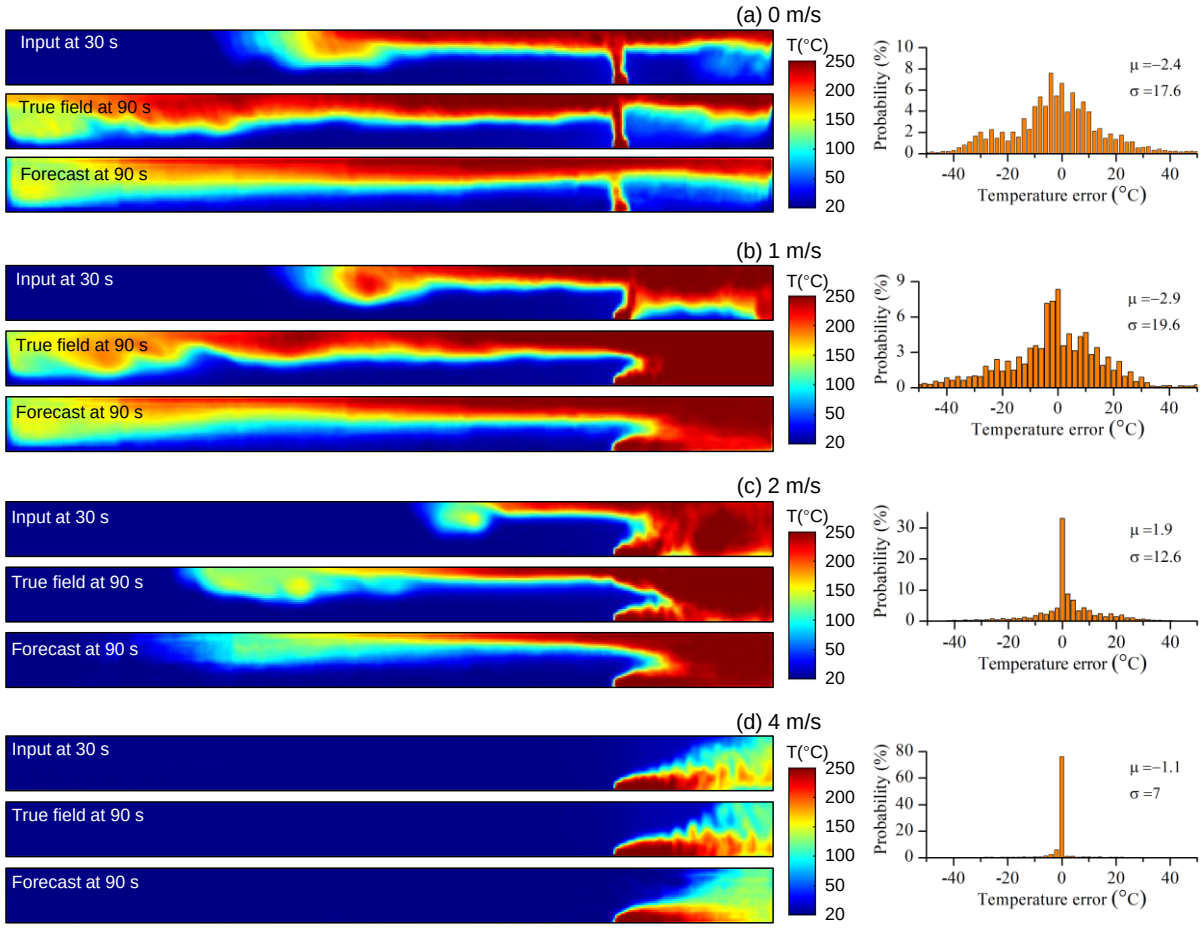


**Fig. 10.** Comparison between the true and forecasted temperature fields at 90 s, where inputs are 32 temperature sensor data of the first 30 s (left); and the deviation of forecasted temperature (right), where the fire is located 128 m from the left boundary, the fire HRR is 20 MW, and the ventilation velocity is (a) 0 m/s, (b) 1 m/s, (c) 2 m/s, and (d) 4 m/s.

### *3.4. Identification of critical gas temperature*

In case of a fire, it is crucial to predict the occurrence of critical events, such as flashover and backdraft for compartment fires, and smoke back layering, and the lack of available egress time for tunnel fires. Successful identification of these critical events can support the decision making of the fire commander and reduce the casualties of trapped personnel and firefighters. On top of the AI prediction, expert knowledge can be further applied to judge the severity of the fire incident as well as the egress situation. Then, the AI model could send an early alert to firefighters before the fire develops into uncontrollable situations. Typical critical conditions of tunnel fire include the gas temperature above 60 ℃ at the height of 2 m, a CO concentration above 1500 ppm, a radiation intensity above 6.3 kW/m$^2$, and the visibility below 10 m (Chen *et al.* 2009). As the proposed AI model has demonstrated the capability of forecasting the temperature field, the forecast of critical gas temperature is performed.

As shown in Figs. 8-10, the hot smoke inclines toward downstream and mixes with cool gases under strong ventilation conditions. By excluding the high-temperature fatal zone near the fire source, the critical condition is reached if the average temperature of the downstream region below 2 m exceeds the maximum tolerant body temperature of 60 ℃ (Chen *et al.* 2009). Two approaches can be used to identify this critical event, (I) directly calculating the average temperature with the predicted temperature field, and (II) defines the classification model. Approach I may be easily misled by the deviation between the true and predicted temperature field images. The performance of classification models is generally evaluated by a confusion matrix (Salhi *et al.* 2019; Sayad *et al.* 2019). As shown in Fig. 11, the four cells of

- True positive (TP), cases critical are correctly identified as critical;
- False positive (FP), cases not critical are wrongly identified as critical;
- False negative (FN), cases critical are wrongly identified as not critical; and
- True negative (TN), cases not critical are correctly identified as not critical.

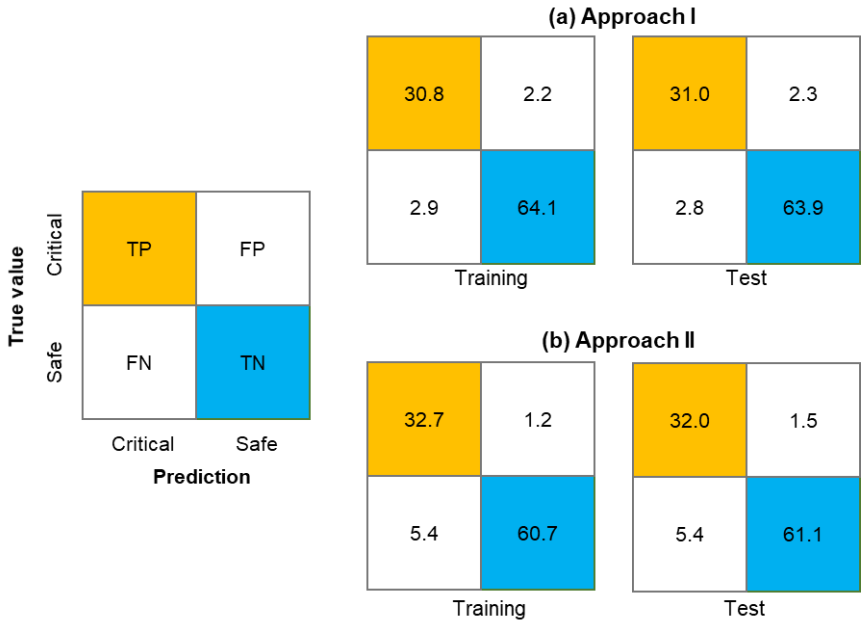


**Fig. 11.** Confusion matrix for training and testing, (a) Approach I, and (b) Approach II.

Ideally, all the data should be along with the diagonal cells, indicating that the predicted and true values match perfectly. As shown, most of the cases in training, validation, and test datasets can be accurately classified for both methods. Further attention could be paid to the FN cases since they are critical while they are not figured out, which may lead to difficult situations due to the absence of necessary actions. The method leveraging the classification method performs better than the method using predicted images directly. Overall, both methods behave quite well, considering the limited amount of error cases.

## 4. Conclusions

In this work, we achieve a real-time prediction (within 1 s) of the spatial-temporal temperature distribution inside the numerical tunnel model by using artificial intelligence (AI) methods. A numerical database of 100 tunnel fire scenarios under various fire location, fire size, and ventilation condition is established. The proposed AI model combines a Long Short-term Memory (LSTM) model and a Transpose Convolution Neural Network (TCNN). The real-time ceiling temperature profile by 32 sensors with an interval of 5 m and thousands of temperature-field greyscale images are used as the training input and output.

Results show that the predicted temperature field 60 s in advance achieves a high accuracy of around 97%. The phenomenon that higher temperature occurred near the fire location and tunnel ceiling can be well predicted. While it should be noted that the extreme temperature near the fire source is harder to be learned by the AI model, which could be attributed to that expert knowledge is not input into the model. Quantitative checking on the prediction error shows that the average of the difference between predicted and simulated temperature distribution is roughly within 10 ℃. This prediction error is acceptable, considering the large variety of fire-induced environment temperature inside the tunnel in practice. Moreover, the proposed fire forecast is able to quickly identify the critical temperature field (i.e., a critical event) in the order of ms, showing its capacity for guiding emergency responses and firefighting activities in real-time.

This study demonstrates the promising prospects of AI-based real-time fire forecast and set an example for applying smart firefighting technologies in tunnel spaces. However, it should be noted that in practice, there are many uncertainties related to fire development or fire spreading, which has not been considered in the CFD models. A larger database containing the simulation results of more realistic fire scenarios could be constructed in the future to train the proposed AI model before being applied in practical firefighting.

## 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

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.

Akhloufi MA, Booto Tokime R, Elassady H (2018) Wildland fires detection and segmentation using deep learning. In 'Proceedings Volume 10649, Pattern Recognition and Tracking XXIX; 106490B', 11

Aksoy S, Haralick RM (2001) Feature normalization and likelihood-based similarity measures for image retrieval.

*Pattern Recognition Letters* **22**, 563–582.

Alzubaidi L, Al-Shamma O, Fadhel MA, Farhan L, Zhang J (2020) Classification of red blood cells in sickle cell anemia using deep convolutional neural network. *Advances in Intelligent Systems and Computing* **940**, 550–559.

Arora R, Basu A, Mianjy P, Mukherjee A (2018) Understanding deep neural networks with rectified linear units. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings* 1–17.

Babrauskas V, Peacock DR (1992) Heat release rate: the single most important parameter in fire hazard. *Fire Safety Journal* **18**, 255–272.

Beard AN (2009) Fire safety in tunnels. *Fire Safety Journal* **44**, 276–278.

Beard A, Carvel R (2012)' Handbook of tunnel fire safety.' (ICE Publishing)

Bengio Y, Simard P, Frasconi P (1994) Learning Long-Term Dependencies with Gradient Descent is Difficult. *IEEE Transactions on Neural Networks* **5**, 157–166.

Cao Y, Yang F, Tang Q, Lu X (2019) An attention enhanced bidirectional LSTM for early forest fire smoke recognition. *IEEE Access* **7**, 154732–154742.

Carvel R (2019) A review of tunnel fire research from Edinburgh. *Fire Safety Journal* **105**, 300–306.

Carvel RO, Beard AN, Jowitt PW (2005) Fire spread between vehicles in tunnels: Effects of tunnel size, longitudinal ventilation and vehicle spacing. *Fire Technology* **41**, 271–304.

Casey N (2020) Fire incident data for Australian road tunnels. *Fire Safety Journal* **111**, 102909.

Chehreh Chelgani S, Shahbazi B, Hadavandi E (2018) Support vector regression modeling of coal flotation based on variable importance measurements by mutual information method. *Measurement: Journal of the International Measurement Confederation* **114**, 102–108.

Chen MY (2011) Predicting corporate financial distress based on integration of decision tree classification and logistic regression. *Expert Systems with Applications* **38**, 11261–11272.

Chen S, Weng Y, Ke M, Chen S (2009) Safety Reliability Method Applied to Evaluate Smoke Control System of Fire Accident in the Tunnel of Mass Rapid Transit System. *Lecture Notes in Engineering and Computer Science* **2177**, 1426–1430.

Cohen P, Greenberg M, Hart D, Howe A (1989) AI Magazine Volume 10 Number 3 (1989) (© AAAI). *AI magazine* **10**, 32–48.

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',

Dexters A, Leisted RR, Van Coile R, Welch S, Jomaas G (2020) Testing for knowledge: Application of machine learning techniques for prediction of flashover in a 1/5 scale ISO 13784-1 enclosure. *Fire and Materials* 1–12.

Dindarloo SR, Siami-Irdemoosa E (2015) Maximum surface settlement based classification of shallow tunnels in soft ground. *Tunnelling and Underground Space Technology* **49**, 320–327.

Dubey V, Kumar P, Chauhan N (2019) 'Forest Fire Detection System Using IoT and Artificial Neural Network.' (Springer Singapore)

Dumoulin V, Visin F (2016) A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv:160307285*.

Forney GP (2010) Smokeview (Version 5) A Tool for Visualizing Fire Dynamics Simulation Data Volume I : User's Guide.

Ghoreishi (2019) Review of the Punching Shear Behavior of Concrete Flat Slabs in Ambient and Elevated Temperature Mehrafarid. 8301–8305.

Glorot X, Bengio Y (2010) Understanding the difficulty of training deep feedforward neural networks. *Journal of Machine Learning Research* **9**, 249–256.

Gong L, Jiang L, Li S, Shen N, Zhang Y, Sun J (2016) International Journal of Thermal Sciences Theoretical and experimental study on longitudinal smoke temperature distribution in tunnel fires. *International Journal of Thermal Sciences* **102**, 319–328.

Govil K, Welch ML, Ball JT, Pennypacker CR (2020) Preliminary results from a wildfire detection system using deep learning on remote camera images. *Remote Sensing* **12**,.

Haack A (2002) Current safety issues in traffic tunnels. *Tunnelling and Underground Space Technology* **17**, 117–127.

Han L, Potter S, Beckett G, Pringle G, Welch S, Koo SH, Wickler G, Usmani A, Torero JL, Tate A (2010) FireGrid: An e-infrastructure for next-generation emergency response support. *Journal of Parallel and Distributed Computing* **70**, 1128–1141.

Hayou S, Doucet A, Rousseau J (2019) On the impact of the activation function on deep neural networks training. *36th International Conference on Machine Learning, ICML 2019* **2019-June**, 4746–4754.

He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In 'Proceedings of the IEEE conference on computer vision and pattern recognition', 770–778

Hochreiter S (1997) Long Short-Term Memory. **1780**, 1735–1780.

Hodges JL (2018) Predicting Large Domain Multi-Physics Fire Behavior Using Artificial Neural Networks. Virginia Polytechnic Institute and State University.

Hodges JL, Lattimer BY (2019) Wildland Fire Spread Modeling Using Convolutional Neural Networks. *Fire Technology* **55**, 2115–2142.

Hodges JL, Lattimer BY, Luxbacher KD (2019) Compartment fire predictions using transpose convolutional neural networks. *Fire Safety Journal* **108**, 102854.

Hu LH, Tang F, Yang D, Liu S, Huo R (2010) Longitudinal distributions of CO concentration and difference with temperature field in a tunnel fire smoke flow. *International Journal of Heat and Mass Transfer* **53**, 2844–2855.

Hurley MJ, Gottuk D, Hall JR, Harada K, Kuligowski E, Puchovsky M, Torero J, Watts JjM, Wieczorek C (2016) SFPE handbook of fire protection engineering, fifth edition. *SFPE Handbook of Fire Protection Engineering, Fifth Edition* 1–3493.

Ingason H, Li YZ, Lonnermakr A (2015) 'Tunnel Fire Dynamics.' (Springer: London)

Ingason H, Lönnermark A (2012) Heat release rates in tunnel fires: a summary. *Handbook of Tunnel Fire Safety* 309–328.

Ji J, Tong Q, (Leon) Wang L, Lin CC, Zhang C, Gao Z, Fang J (2018) Application of the EnKF method for real-time forecasting of smoke movement during tunnel fires. *Advances in Engineering Software* **115**, 398–412.

Ji J, Wan H, Li K, Han J, Sun J (2015) A numerical study on upstream maximum temperature in inclined urban road tunnel fires. *International Journal of Heat and Mass Transfer* **88**, 516–526.

Kavzoglu T, Mather PM (2003) The use of backpropagating artificial neural networks in land cover classification. *International Journal of Remote Sensing* **24**, 4907–4938.

Koekkoek EJW, Booltink H (1999) Neural network models to predict soil water retention. *European Journal of Soil Science* **50**, 489–495.

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.

Lee D, Lim M, Park H, Kang Y, Park JS, Jang GJ, Kim JH (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.

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.

Li J, Liu J (2020) Science Mapping of Tunnel Fires: A Scientometric Analysis-Based Study. *Fire Technology* **56**, 2111–2135.

Mcgrattan K, Mcdermott R (2015) Fire Dynamics Simulator User' s Guide (FDS Version 6.3.0).

Morales G, Huam SG, Telles J (2018) 'Cloud Detection in High-Resolution Multispectral Satellite Imagery Using Deep Learning : 27th International Conference on Artificial Neural Networks , Cloud Detection in High-Resolution Multispectral Satellite Imagery Using Deep Learning.' (Springer International Publishing)

Nævestad TO, Meyer S (2014) A survey of vehicle fires in Norwegian road tunnels 2008-2011. *Tunnelling and Underground Space Technology* **41**, 104–112.

Ngoc Thach N, Bao-Toan Ngo D, Xuan-Canh P, Hong-Thi N, Hang Thi B, Nhat-Duc H, Dieu TB (2018) Spatial pattern assessment of tropical forest fire danger at Thuan Chau area (Vietnam) using GIS-based advanced machine learning algorithms: A comparative study. *Ecological Informatics* **46**, 74–85.

Oka Y, Atkinson GT (1995) Control of smoke flow in tunnel fires. *Fire Safety Journal*.

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.

Ren R, Zhou H, Hu Z, He S, Wang X (2019) Statistical analysis of fire accidents in Chinese highway tunnels 2000–2016. *Tunnelling and Underground Space Technology* **83**, 452–460.

Salhi L, Silverston T, Yamazaki T, Miyoshi T (2019) Early Detection System for Gas Leakage and Fire in Smart Home Using Machine Learning. *2019 IEEE International Conference on Consumer Electronics, ICCE 2019* 1–6.

Sayad YO, Mousannif H, Al Moatassime H (2019) Predictive modeling of wildfires: A new dataset and machine learning approach. *Fire Safety Journal* **104**, 130–146.

Sun P, Bisschop R, Niu H, Huang X (2020) A Review of Battery Fires in Electric Vehicles. *Fire Technology* 10694.

Sun M, Raju A, Tucker G, Panchapagesan S, Fu G, Mandal A, Matsoukas S, Strom N, Vitaladevuni S (2017) Max-

pooling loss training of long short-term memory networks for small-footprint keyword spotting. *2016 IEEE Workshop on Spoken Language Technology, SLT 2016 - Proceedings* 474–480.

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.

De Vasconcelos MJP, Silva S, Tomé M, Alvim M, Pereira JMC (2001) Spatial prediction of fire ignition probabilities: Comparing logistic regression and neural networks. *Photogrammetric Engineering and Remote Sensing* **67**, 73–81.

Wu Y, Liu Y, Li J, Liu H, Hu X (2013) Traffic sign detection based on convolutional neural networks. *Proceedings of the International Joint Conference on Neural Networks*.

Wu X, Park Y, Li A, Huang X, Xiao F, Usmani A (2020) Smart Detection of Fire Source in Tunnel Based on the Numerical Database and Artificial Intelligence. *Fire Technology*.

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.

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.

Zhai C, Zhang S, Cao Z, Wang X (2020) Learning-based prediction of wildfire spread with real-time rate of spread measurement. *Combustion and Flame* **215**, 333–341.

Zhang X, Wu X, Park Y, Zhang T, Huang X, Xiao F, Usmani A (2021) Perspectives of big experimental database and artificial intelligence in tunnel fire research. *Tunnelling and Underground Space Technology* **108**, 103691.

## Appendix

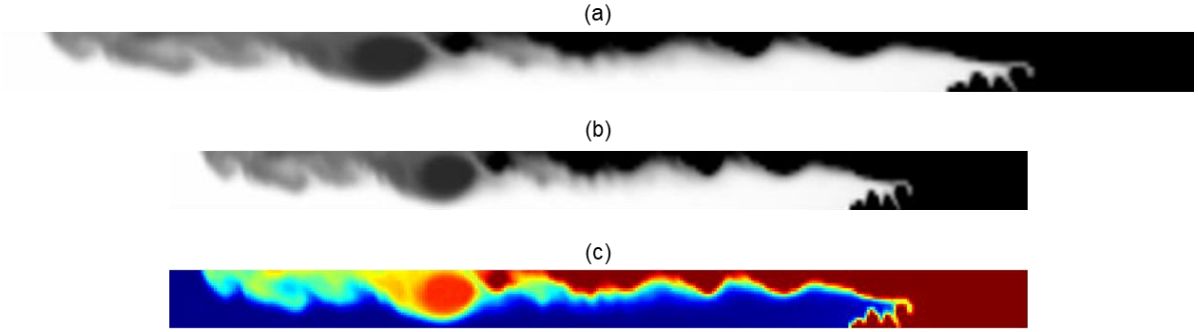


**Fig. A1**. Images showing the temperature field: (a) before scaling; (b) after scaling; (c) after colour mapping.

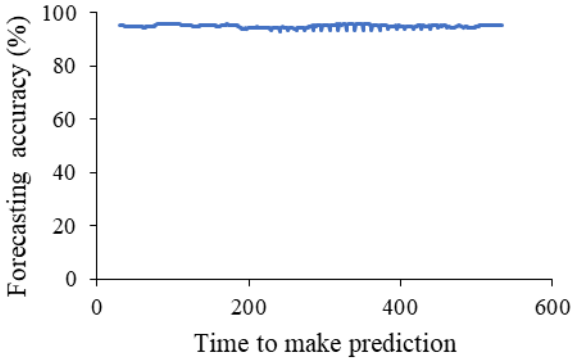


**Fig. A2**. The evolvement of forecasting accuracy with time.

To further check the trained AI model's capacity in forecasting the fire scenarios with transient HRR. Typical $t^2$-growth HRRs (HRR = $\alpha t^2$) with a subsequent constant maximum period are modeled. The moderate value of α is suggested as 0.1 kW/s$^2$ for a fire burning from a bus (Ingason and Lönnermark

2012), and another fast ($\alpha$ = 0.333 kW/s$^2$) and slow ($\alpha$ = 0.02 kW/s$^2$) fires are also modeled (Fig. A3a). Fig. A3b shows the forecasting accuracy ($R^2$) evolving with time calculated by comparing the simulation and predicted results 60 s afterward.

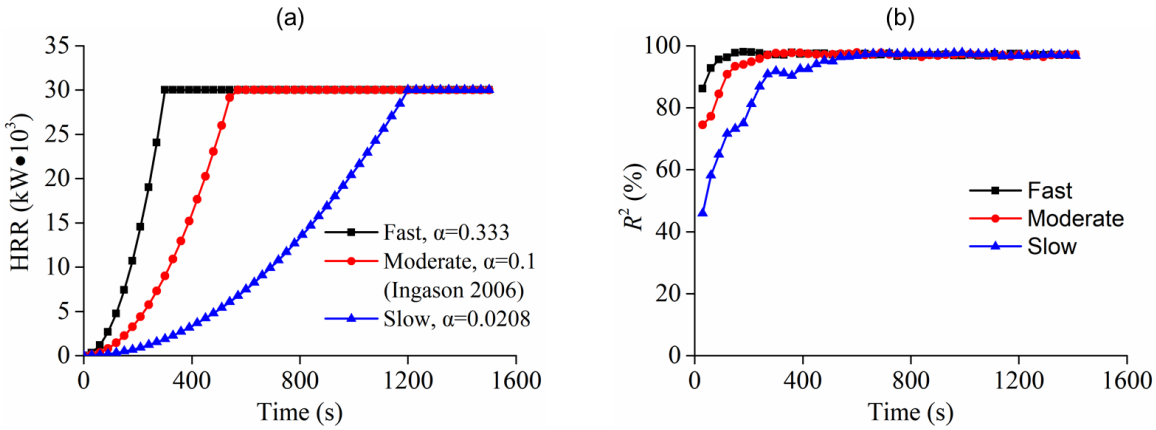


**Fig. A3.** (a) HRR growth curves and (b) AI-prediction accuracy for fast, moderate, and slow fire scenarios.

As expected, the accuracy is lower at the initial growth stage. One reason is that the HRR at the initial stage is lower than the minimum value of 5 MW the AI model has learned. Counterintuitively, higher prediction accuracy is gained for the faster-growth HRR. It is because the AI model is trained with constant HRR cases (i.e., the infinite-fast growth). The accuracy increases rapidly to a level higher than 90% after 60 s, 120 s, and 270 s for the fast, moderate, and slow fire-growth scenarios, respectively. After 600 s while the HRR is still increasing for the slow fire scenario, the predicting accuracy keeps almost constant at 97% for all the scenarios. This demonstrates that the constant HRR could be a reasonable simplification of the complicated fire scenarios when constructing the training database. However, to achieve higher accuracy at the initial stage, the AI model needs to be trained with fire scenarios with smaller and more transient HRRs.

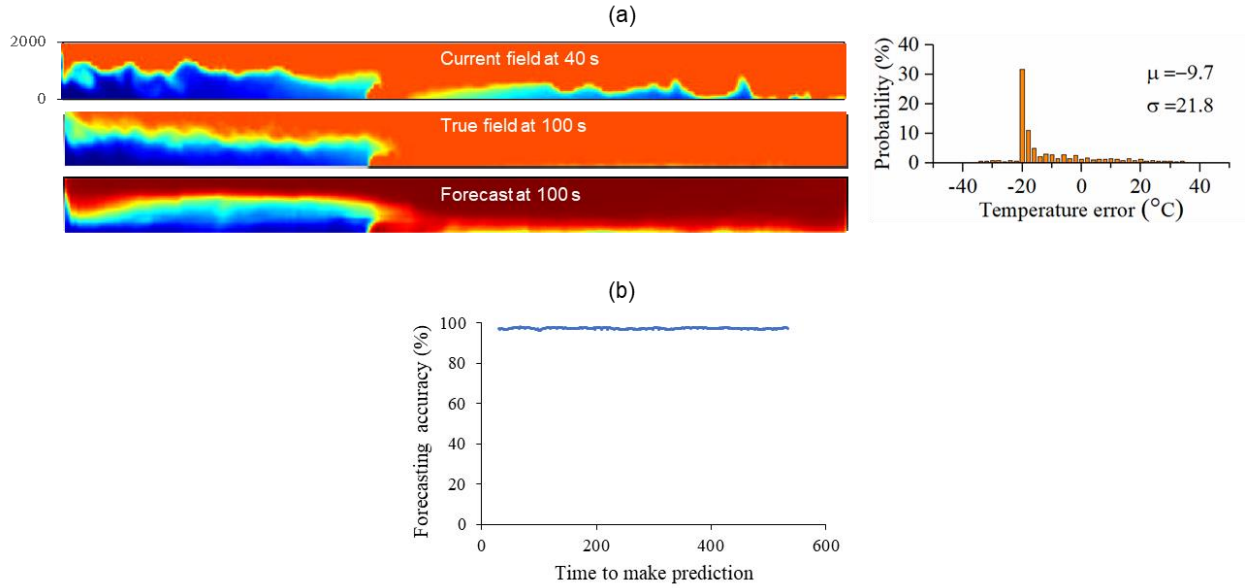


**Fig. A4**. Fire scenario where the fire is located at 64 m from the left boundary, the fire HRR is 130 MW and the ventilation speed is 2 m/s, (a) comparison between the true and forecasted temperature fields at 40 s; (b) evolvement of forecasting accuracy with time.