Fire Safety Journal 108 (2019) 102854

Contents lists available at ScienceDirect

Fire Safety Journal

journal homepage: www.elsevier.com/locate/firesaf

Compartment fire predictions using transpose convolutional neural networks

Jonathan L. Hodges*, Brian Y. Lattimer, Kray D. Luxbacher

ARTICLE INFO

Keywords:
Compartment fire
Machine learning
ANN neural network
Computational fluid dynamics
CFD
Convolutional
CNN

ABSTRACT

This paper presents a data-driven approach to predict spatially resolved temperatures and velocities within a compartment based on zero-dimensional zone fire modeling using a transpose convolutional neural network (TCNN). A total of 1333 Fire Dynamics Simulator (FDS) simulations of simple two-compartment configurations with different fire locations, fire sizes, ventilation configurations, and compartment geometries were used in training and testing the model. In the two-compartment test cases 95% of TCNN predicted temperatures and velocities were within ± 17.2% and ± 0.30 m/s of FDS predictions. Although the model was trained and tested using a simple two-compartment configuration, the TCNN approach was validated with two more complex multi-compartment FDS simulations by processing each compartment individually. Overall, the flow fields in the multi-compartment validation tests agreed well with FDS predictions with 95% of TCNN predicted temperatures and velocities within ± 11% and ± 0.25 m/s of FDS predictions. Coupling a zone fire model with the TCNN approach presented in this work can provide spatially resolved temperature and velocity predictions without significantly increasing the computational requirements. Since the approach is based on a zone fire model, the TCNN approach presented in this work is limited to simplified geometries which can be sufficiently modeled using a zone fire model.

1. Introduction

Understanding the transport of energy and combustion products in a compartment fire is vital in fire hazard analysis. Although advancements in computing technology have made computational fluid dynamics (CFD) of compartment fires possible, the computational cost can be prohibitive. Parametric studies and predictions of large structures often rely on more coarse predictions such as those from zero-dimensional zone fire models. Researchers have presented multi-scale modeling approaches to fuse three-dimensional CFD with zero-dimensional zone models; however, each approach inherits the computational cost of CFD. Recent advancements in the field of generative modeling using machine learning have made it possible to develop rapid data-driven predictions from a small vector of high level descriptors. The applicability of generative modeling to multiple room fire simulations is particularly promising for eventual use in fire safety design and risk analysis in large built structures which rely on repetitive geometry in terms of topology (e.g., tunnel systems, underground storage facilities, mines, etc.). A new compartment fire model which uses the predictions from a zone fire model as inputs to a generative machine learning model may be able to provide high resolution predictions of intensive properties at low computational costs for these structures.

CFD fire models solve conservation equations of mass, momentum, and energy for each grid cell, often numbering $10^5 - 10^7$ [1]. These models are often used in fire protection engineering to predict complex flow fields for smoke control, to estimate smoke detector and sprinkler activation times, and to estimate ignition times of different surfaces. However, the high spatial-temporal resolution comes at a computational cost, where simulation times are often measured in hours, days, or even weeks depending on the simulation. In particular, the cost associated with CFD predictions of large domains or parametric studies can be prohibitive.

More rapid predictions over large domains are possible with zone fire models. Zone based compartment fire models take advantage of the thermal stratification typical in enclosure fires to represent each room as either one or two well-mixed gas regions. Since the number of grid points using a zero-dimensional zone model is significantly less, rapid predictions of the mean transport of energy and species through compartments can be calculated. Predictions of mean properties have been shown to correlate well with experimental measurements [2]. It is not possible to determine local variations in temperature and flow velocity using a zone fire model since the gas regions are assumed to be well-mixed. Although zone fire models provide valuable insight, they are not capable of making predictions for complex configurations and lack the

* Corresponding author. Jensen Hughes, 2020 Kraft Drive, Suite 3020, lacksburg, VA, 24060, USA.
E-mail address: jhodges@jensenhughes.com (J.L. Hodges).

https://doi.org/10.1016/j.firesaf.2019.102854
Received 12 November 2018; Received in revised form 31 May 2019; Accepted 22 July 2019
Available online 25 July 2019
0379-7112/

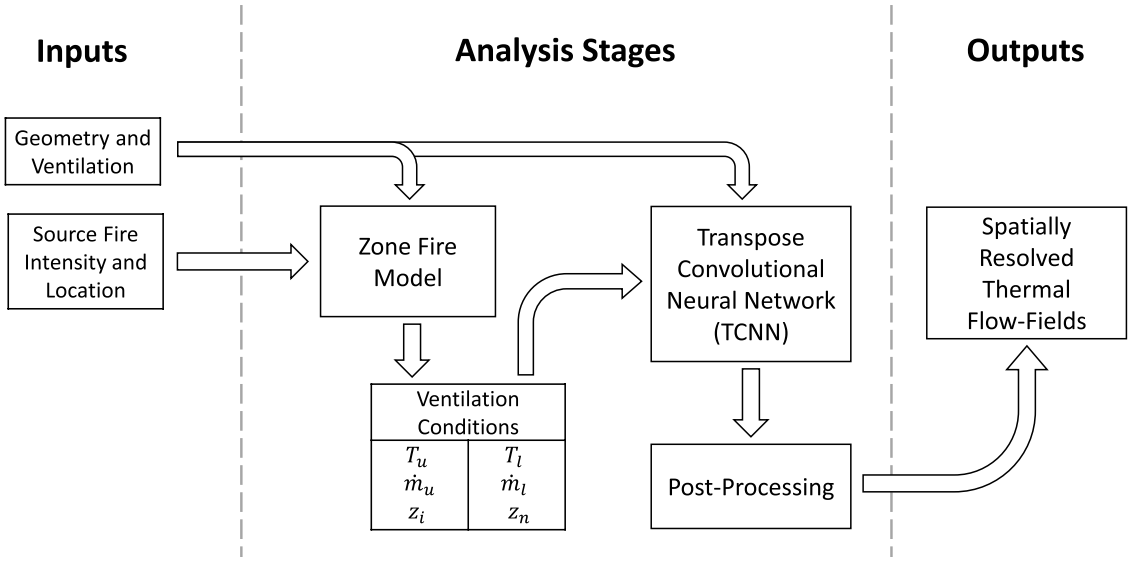


Fig. 1. Schematic of solution algorithm.

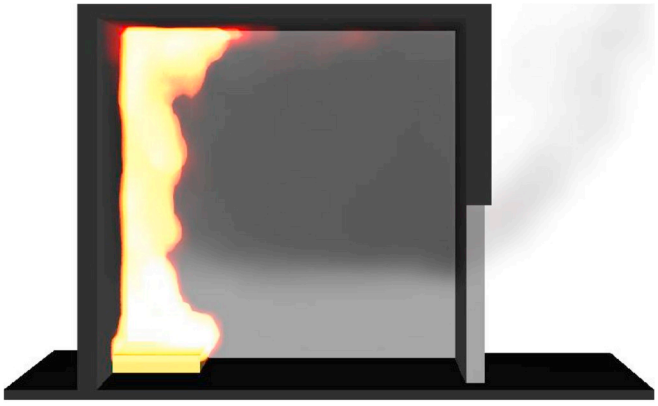
resolution that is sometimes needed in hazard analysis.

Multi-scale modeling approaches couple a CFD model in regions with high spatial variation (near to fire) with a more coarse model further away. This type of approach has been used in many structural applications including contaminant dispersion [3,4], energy efficiency [5–10], and fire response [11–18]. Researchers in the area of tunnel fire modeling have shown this type of approach can reduce computational times with relatively low loss in accuracy [17,18]. Although the computational time reduction can be significant, simulations of large domains are on the order of hours.

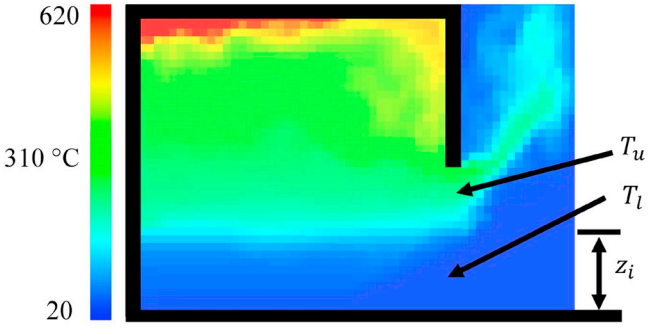
Generative modeling is a new field of research in the machine learning community which has focused primarily on developing techniques to generate RGB images. Although generating data, designing, and training the model can take a significant amount of time, a generative model is able to make rapid predictions of new scenarios after training. In addition, if the model is trained on a generalized data set, the model does not need to be re-trained to make rapid predictions of new scenarios. Researchers have started to apply generative modeling techniques to CFD applications such as studying wake flow dynamics [19–22], modeling turbulence [23,24], and accelerating the computations of time-dependent ODEs and PDEs [25–27]. Chang presented an overview of current neural network based CFD modeling techniques in Ref. [28]. Each of these approaches have focused on using generative modeling to fill gaps in data within the flow field used to train the model. While this approach can provide significant insight into the physics of the system, it requires that the model be re-trained for each new scenario. Few researchers have attempted to develop a generative model which can be used to predict flow fields outside the training dataset. The most similar of these works to the present study was presented by Lee et al. [21]. The authors used a deep recurrent attentive writer (DRAW) generative network similar to Ref. [29] to predict unsteady flow fields over cylinders at different Reynolds numbers. The generative network used four sequential flow field predictions to predict a fifth flow field to good effect. However, the predictions are based on CFD calculations at a similar resolution to the network predictions, and the separate flow fields used to train and evaluate the network were similar. Researchers have not applied generative modeling techniques to model reacting flow fields or predict high resolution thermal flow fields from zero-dimensional zone models.

There are two learning tasks associated with generative modeling: the data model (optimal vector to describe the data), and the generator (architecture to produce a detailed description from the input vector) [30]. Much of the work in the literature focuses on unsupervised learning of the data model [31,32], or coupling these two tasks [33–35]. These type of approaches excel at developing classification schemes through unsupervised learning; however, in physics, modeling regression problems are typically of interest which are not trivial to approach in unsupervised learning. This work was inspired by the approach presented by Dosovitskiy et al. [30,36] who focused on developing an up-convolutional (or transpose convolutional) neural network approach to the generator task to create two-dimensional rendered images of three-dimensional objects using an assumed data model and supervised learning. The data model consisted of a vector describing the class of the object, perspective information, and transformation information.

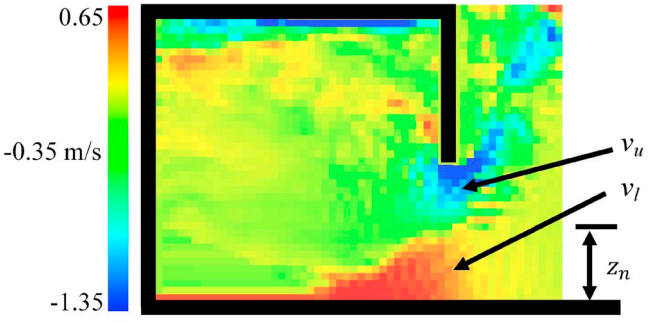
The objective of this study was to develop a data-driven generative model to provide spatially resolved predictions of temperature and velocity based on coarse predictions from a zero-dimensional zone fire model. The vector of inputs to the generative model included geometric parameters and predictions of physical parameters from the zone fire model (such as mass flow rates and gas temperatures at doors). A total of 12 output channels were predicted which consisted of two-dimensional slices of temperature and three-component velocity through two vertical center-lines of the compartment, and one horizontal plane 0.1 m below the ceiling. The network architecture used in this work was a transpose convolutional neural network (TCNN) similar to that of Dosovitskiy [30]. Data for use in training and testing the model was generated using Fire Dynamics Simulator (FDS) version 6.2 developed by NIST [37]. Since the model was trained using a wide range of room geometries and thermal exposures, the model can provide rapid estimates of spatially resolved



(a) Visualization of smoke layer.



(b) Visualization of upper and lower layer temperature and interface height.



(c) Visualization of neutral plane and velocity of gas into and out of ventilation.

Fig. 2. Example compartment fire simulation (side view).

temperatures and velocities for new scenarios without retraining the model. The following sections describe the network architecture, data generation, and results. This work demonstrates a methodology to predict spatially resolved flow fields in simplified compartment fires by coupling a zero-dimensional zone fire model with a TCNN.

2. Methods

A schematic showing a high level view of the solution algorithm is shown in Fig. 1. Initial predictions of a fire scenario are made using a zone fire model. These predictions and a description of the geometry are input to the TCNN. The TCNN then uses its prior training to predict the thermal flow-field within each compartment individually. These predictions are then post-processed to output final predictions. The following subsections describe the network architecture, data generation, and post-processing used in this work.

2.1. Network architecture

The artificial neural network approach presented in this work falls in the class of generative modeling in the machine learning community. The focus of generative modeling is to use a small number of input parameters to predict a significantly more detailed output. The specific inspiration for this work comes from the work of Dosovitskiy et al. [30]. The authors created a transpose convolutional neural network which would render a two-dimensional image of different three-dimensional objects viewed from different angles and lighting from high level descriptors of the object. The authors observed after training that the TCNN was able to accurately interpolate new views of the objects as well as extrapolate beyond the views in the training set. For the TCNN architecture presented herein, the high level descriptors are replaced with zero-dimensional zone fire model predictions of compartment conditions, and the rendered images are replaced with spatially resolved slices of gas temperature and velocity.

An example CFD simulation of a compartment fire is shown Fig. 2a. The CFD predictions of temperature and velocity through the center of the compartment are shown in Fig. 2b and c, respectively. Fig. 2b shows the air in the compartment is generally split into two zones, the hot gas layer and the lower layer. The transition height between these layers is called the smoke layer height/interface height. Fig. 2c shows the air flow through the door is generally split into two zones where the speed and direction of the air varies with height. The transition height where the direction of airflow switches is called the neutral plane. Zone fire models assume each layer is well mixed with a single mean value for each intensive property (e.g. temperature, concentration, pressure) representative of the space. Conservation laws are solved for each zone to predict the change in mean value over time. Typical predictions from a zone fire model include predictions of temperature in the upper and lower layer and the smoke layer height/interface height, in addition to mass flow rates and neutral plane predictions for each connection to other rooms.

The TCNN architecture used in this work is shown in Fig. 3. The architecture consists of a total of eleven hidden layers, of which nine are trained processing layers. The input layer consists of a vector of 35 points. Three numbers correspond to the basic compartment geometry (L, W, H). The remaining 32 points are split evenly between each wall surface (North, South, East, and West) for a total of eight numbers describing the junction to the neighboring space for each wall. These eight points describe the geometry and flow conditions of the ventilation. The points contained in the input vector are summarized in Table 1. The three dense (fully-connected) layers are used to develop a high dimensional representation of the input parameters prior to processing through the TCNN and CNN layers. A single dropout layer after this first fully-connected layer was used to prevent over-fitting during training [38]. Similar to Dosovitskiy et al., the x-y resolution of the data was increased through 3 cycles of back to back TCNN and CNN layers. The output images were 50×50 pixels with twelve image channels corresponding to spatially resolved temperature, x-axis velocity (U-velocity), y-axis velocity (V-velocity), and z-axis velocity (W-velocity)

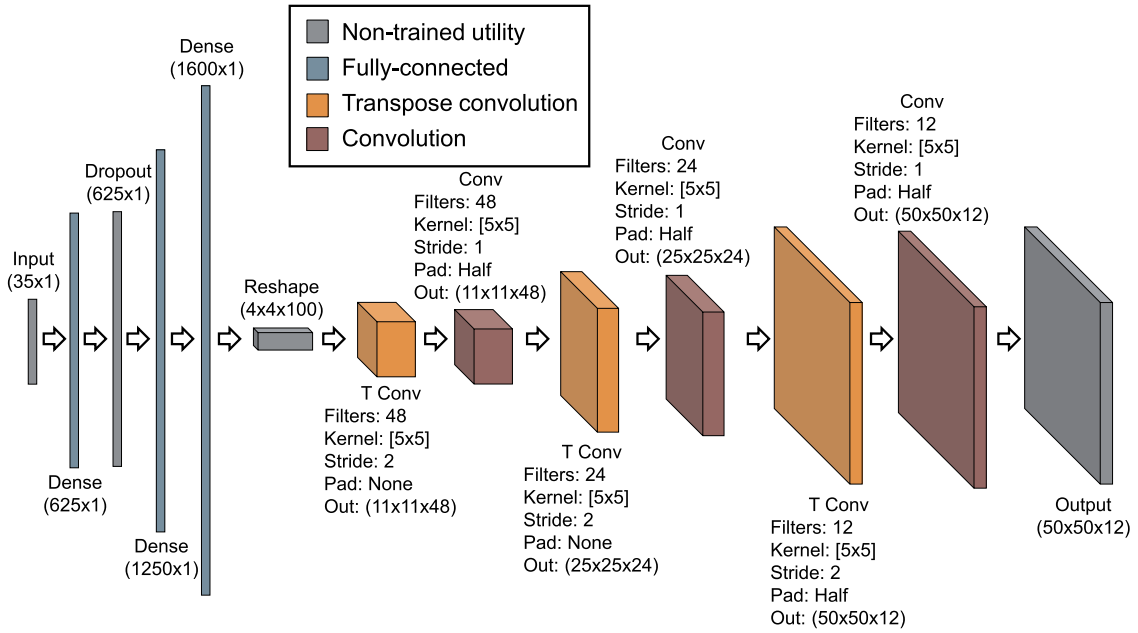


Fig. 3. Schematic of the TCNN architecture used in this work.

Table 1
Description of points in input vector.

Variable	Name	Description
L	Length	Total extent of room along the x-axis
W	Width	Total extent of room along the y-axis
H	Height	Total extent of room along the z-axis
S	State	Whether ventilation is open or closed
D_S	Size	Class of ventilation size. 0: 0.625 m × 1.9 m 1: 1.250 m × 1.9 m 2: Full wall
T_u	Upper layer temperature	Temperature above the interface height
T_l	Lower layer temperature	Temperature below the interface height
z_i	Interface height	Height of the hot gas layer
$\dot{m}_u$	Upper layer mass flow rate	Flow rate above the neutral plane
$\dot{m}_l$	Lower layer mass flow rate	Flow rate below the neutral plane
z_n	Neutral plane height	Height of inflection point in gas velocity

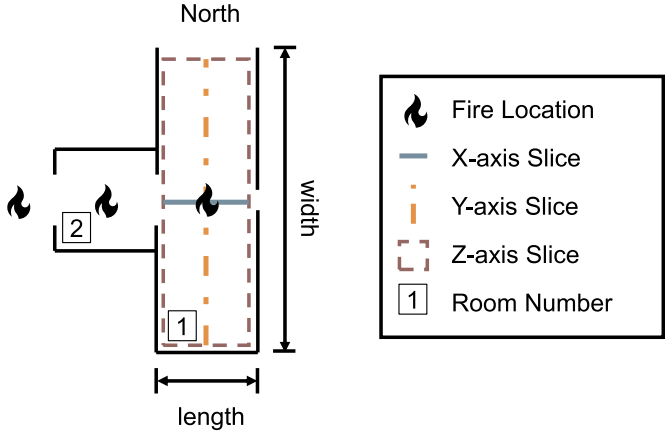


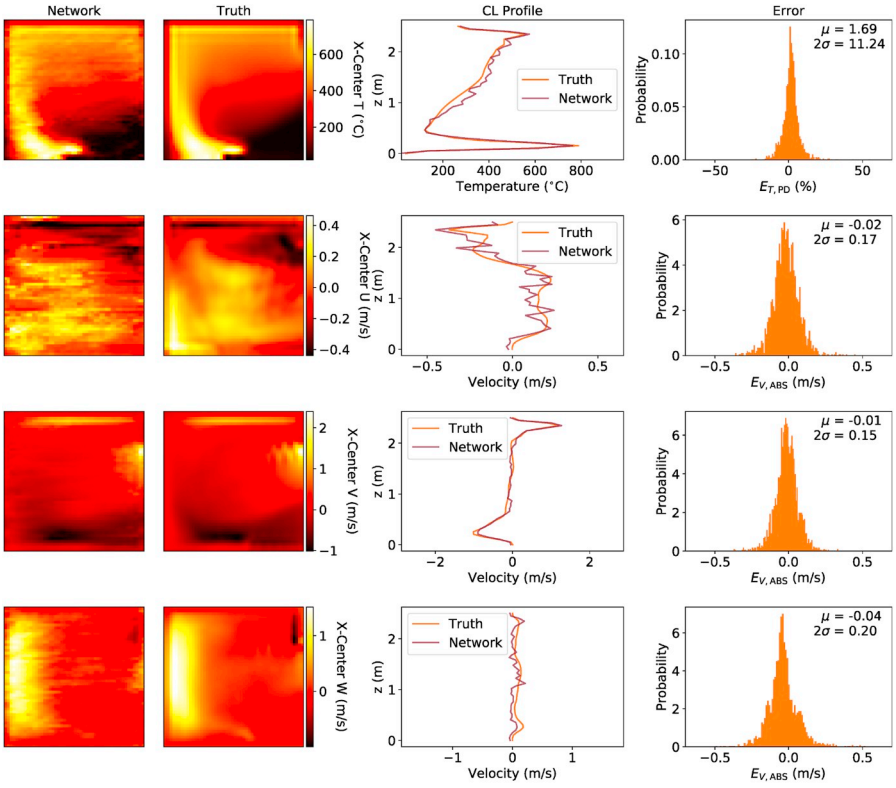
Fig. 4. Schematic of two-room configuration (plan view). Predictions of temperature and velocity in Room 1 were used in training the model.

Table 2
Limits of each parameter in study.

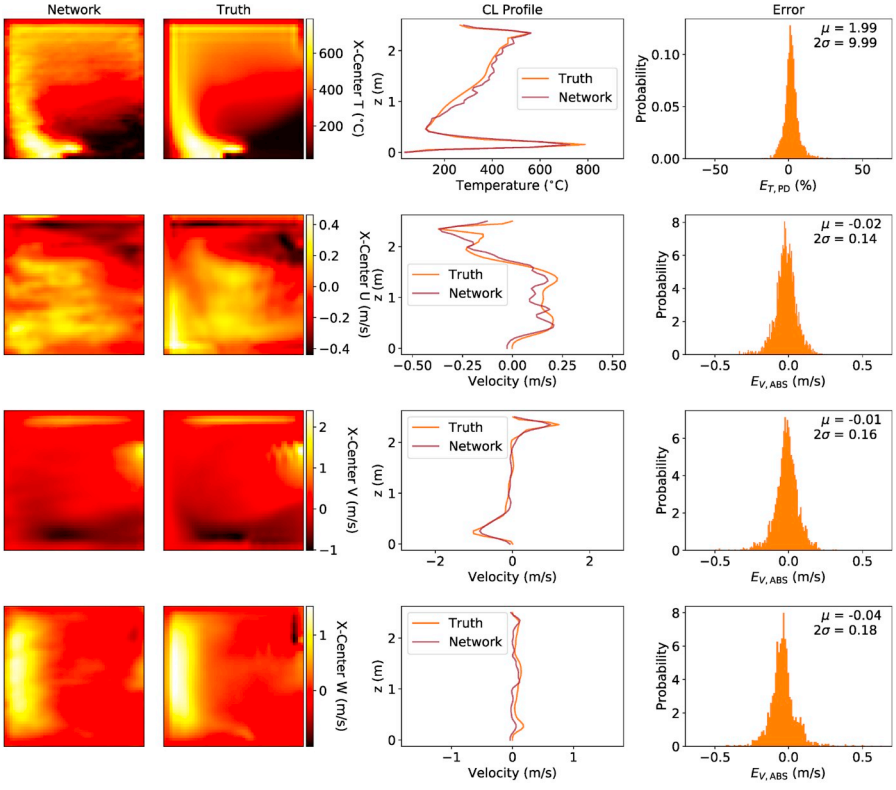
Room Geometry	
Width	2.5–10.0 m
Length	2.5–10.0 m
Height	2.5 m
Wall Ventilation	North, East, South, West, West-West
Existence	True/False
Ventilation Size	0. 0.625 m × 1.9 m door 1. 1.250 m × 1.9 m door 2. Full wall
Source Configuration	
Location	0. Fire in Room 1, 1. Fire in Room 2, 2. Forced ventilation west of Room 2
Intensity	
If Fire	
Heat Release Rate	50-1000 kW
If Two-Layer Forced Ventilation	
Temperature	500-1000 °C
Velocity	1–10 m/s
Interface Height	0–1.9 m
Neutral Plane	0–1.9 m

through the room. To demonstrate the approach, two vertical slices were located along the center-line of the x-axis and y-axis as well as one horizontal slice was located 0.1 m below the ceiling. The number of filters and step size in each CNN and TCNN layer and the number of neurons in the fully connected layers were specified to steadily increase the degrees of freedom of the system from 35 in the input layer to 30,000 in the output layer. All layers in the TCNN network used leaky rectified linear unit (ReLU) activation functions, as recommended by other researchers for regression problems [39,40].

The network architecture was built using the Python 3 bindings for TensorFlow [41]. The model was trained using stochastic gradient descent with a batch size of 100 samples. All weights and biases were initialized from a uniform distribution between −1 and 1. The learning rate was fixed for all layers throughout training at 10^{-4}. The cost function used in training was based on square error, as recommended



(a) Raw predictions.



(b) Post-processed predictions.

Fig. 5. Impact of post-processing on the noise in raw predictions.

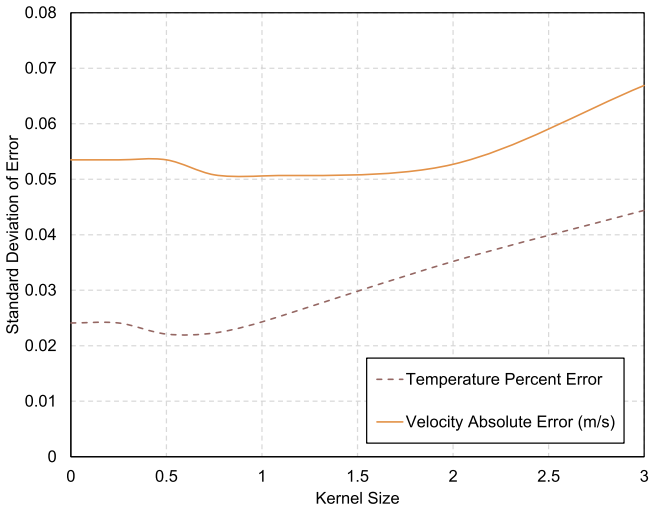


Fig. 6. Standard deviation of error from training data with different standard deviation kernels for Gaussian filter.

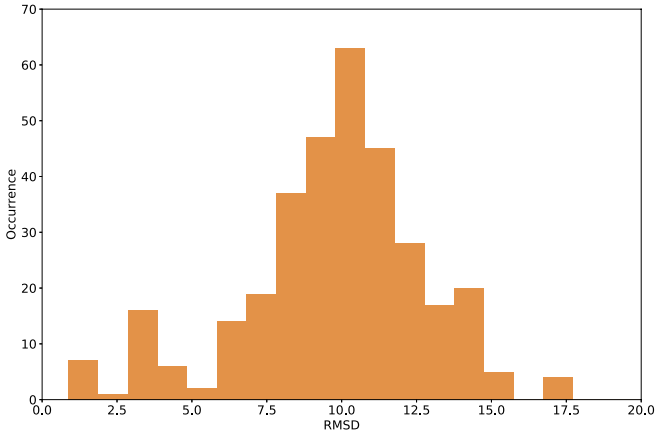


Fig. 7. Histogram of RMSD between each testing case and the nearest case in the training set.

by Dosovitskiy. Over-fitting was reduced by using 75% dropout on the dropout layer and shuffling the order of the samples during training. The network was trained using 1000 simulations (4990 unique scenarios) for 200,000 cycles using a single NVIDIA Quadro K620 graphics card. The total time to train the network was 420 h. The total time for the TCNN to predict the spatially resolved temperature and velocities for 5000 scenarios was 8 s.

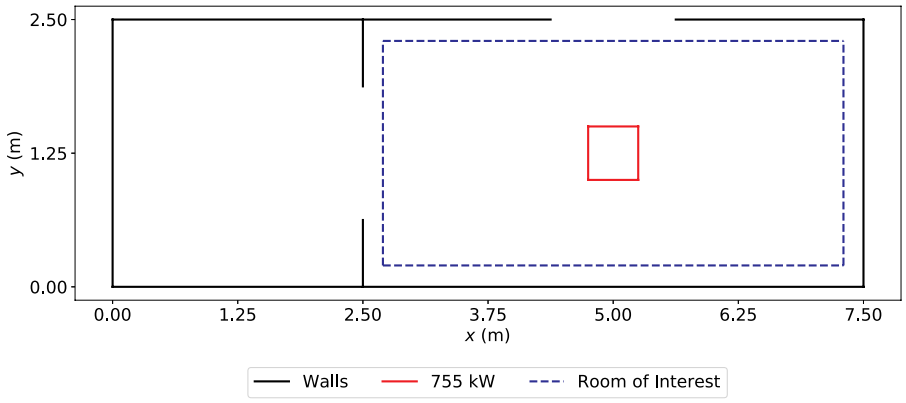
2.2. CFD data generation

It would be ideal to use high resolution fire simulations of entire structures when training the neural network to ensure the necessary physics is represented. However, since numerous simulations are required (on the order of 1000–10,000) to train the model, the computational cost of generating a data set using full structures was prohibitive. Instead a two-room configuration was used to generate data for use in training the model. An example schematic of a two-room configuration is shown in Fig. 4. This simplification is reasonable as long as the range of initial conditions, opening flow conditions, and boundary conditions of the examined room in the two-room configuration capture the range of realistic conditions for the rooms in a full structure. This variation was modeled by varying the geometry of Room 1, the position of the source fire within the two-room configuration, and the intensity of the source fire. For this study the impact of thermal losses to the boundary were removed by considering all boundary surfaces as adiabatic.

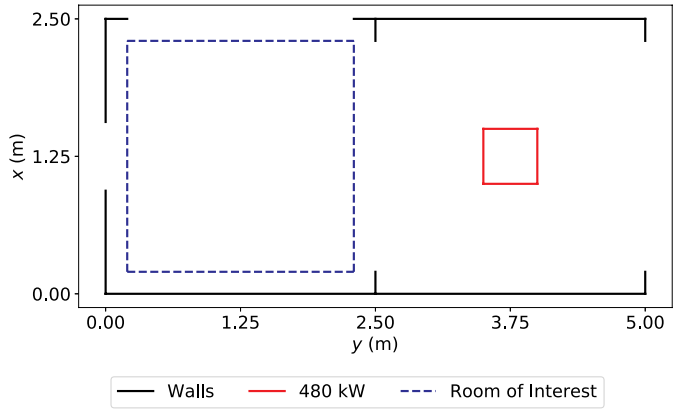
The parameters varied in this study are shown in Table 2. Monte-Carlo sampling with uniform distributions was used to select unique scenarios from the parameters in Table 2. Either the length or width of Room 1 was fixed at 2.5 m and the other selected from the range of 2.5–10.0 m. The existence of a door on each wall was selected independently; however, at least one door was required to allow air flow. If a door was present, it was selected to be one of three possible sizes: 0.625 m × 1.9 m, 1.250 m × 1.9 m, or the full extent of the wall (representing a hallway connection). The heat source was randomly selected to be either a fire in the center of Room 1, a fire in the center of Room 2, or a two-layer forced ventilation outside Room 2 (representing a distant fire). If the heat source was not in Room 1, a door was always open between Room 1 and Room 2. If the source was a fire, the heat release rate was selected from 50–1000 kW. If the source was forced ventilation, the upper layer temperature was selected from 500–1000 °C, the velocity from 1 to 10 m/s, the interface height from 0 to 1.9 m, and the neutral plane from 0 to 1.9 m.

A total of 30 s of exposure was simulated for 1000 unique configurations using FDS. Although 30 s of exposure is not sufficient for these configurations to reach steady state, this time interval was sufficient to relate the door flows to spatially resolved intensive properties in the room. For each simulation, the temperature, U-velocity, V-velocity, and W-velocity along the x-axis center-line, y-axis center-line, and z-axis just under the ceiling were averaged from 10 to 30 s of exposure and output as two-dimensional data files. Longer simulation times could have been used to extract multiple time-averaged samples from each simulation; however, this study used a single sample per simulation to limit the computational cost per scenario so more geometric and heat source variation could be considered. The grid resolution was fixed at 0.1 m along each axis for each simulation to limit the computational cost per simulation. Although this grid resolution does not fully resolve the source fire, the predictions were adequate for preliminary analysis, where the focus was on reproducing CFD simulations from zone simulations rather than comparing to experimental results.

The size of the data-set was artificially increased through rotation up-sampling and ambient re-sampling. Rotation up-sampling was used to mitigate the impact of Room 2 being in a fixed position relative to Room 1 on the TCNN predictions. This process consisted of rotating the coordinate axes of the simulation without adjusting the numeric values. Since all FDS simulations included in the data-set originally included a heat source near the room of interest, preliminary predictions from the TCNN approach failed for rooms far from the initiating fire. The impact of this bias was mitigated through ambient re-sampling which physically corresponds to adding a non-flaming case for each geometric configuration. This process consisted of fixing all temperatures in input and output data to an ambient temperature of 20 °C and fixing the mass



(a) Test Case 1.



(b) Test Case 2.

Fig. 8. Geometry from multi-room configuration validation studies. The red square is the source fire position. The blue box is the room of interest. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

flow rates in the upper and lower layer to zero. A total of 5000 unique scenarios were in the database after rotation up-sampling and ambient re-sampling.

2.3. Zone fire model data generation

Estimates of zone model predictions for training and testing the TCNN were developed from the CFD predictions rather than using a zone model to predict the scenarios. This decision was made to ensure the errors in predictions were from the TCNN approach and not from differences in the CFD and zone modeling schemes. The differences resulting from changing the inputs to the TCNN from spatially averaged CFD predictions to zone modeling predictions is discussed in Section 4.

The reduction to a two layer estimate was calculated using the method presented by Janssens as recommended in the FDS validation guide [37,42]. The interface height was calculated using the equations

$$I_1 = (H - z_{int})T_u + z_{int}T_l = \int_0^H T(z)dz \tag{1}$$

$$I_2 = (H - z_{int})\frac{1}{T_u} + z_{int}\frac{1}{T_l} = \int_0^H \frac{1}{T(z)}dz \tag{2}$$

$$z_{int} = \frac{T_l(I_1 I_2 - H^2)}{I_1 + I_2 T_l^2 - 2T_l H} \tag{3}$$

where H was the total height in m, z_{int} was the interface height in m, T_u was the upper layer gas temperature in K, T_l was the temperature of the lowest grid cell in K, and $T(z)$ was the temperature in K through the center-line temperature of the ventilation. The upper layer temperature was calculated using the equation

$$T_u = \frac{1}{(H - z_{int})}\int_{z_{int}}^H T(z)dz. \tag{4}$$

The neutral plane, z_n is defined as the height in m at which the direction of the air flow switches from an inflow to an outflow and was determined directly from the CFD predicted velocity profile. The mass flow rate of the upper and lower layer, $\dot{m}_u$ and $\dot{m}_l$ respectively in kg/s, were calculated using the equations

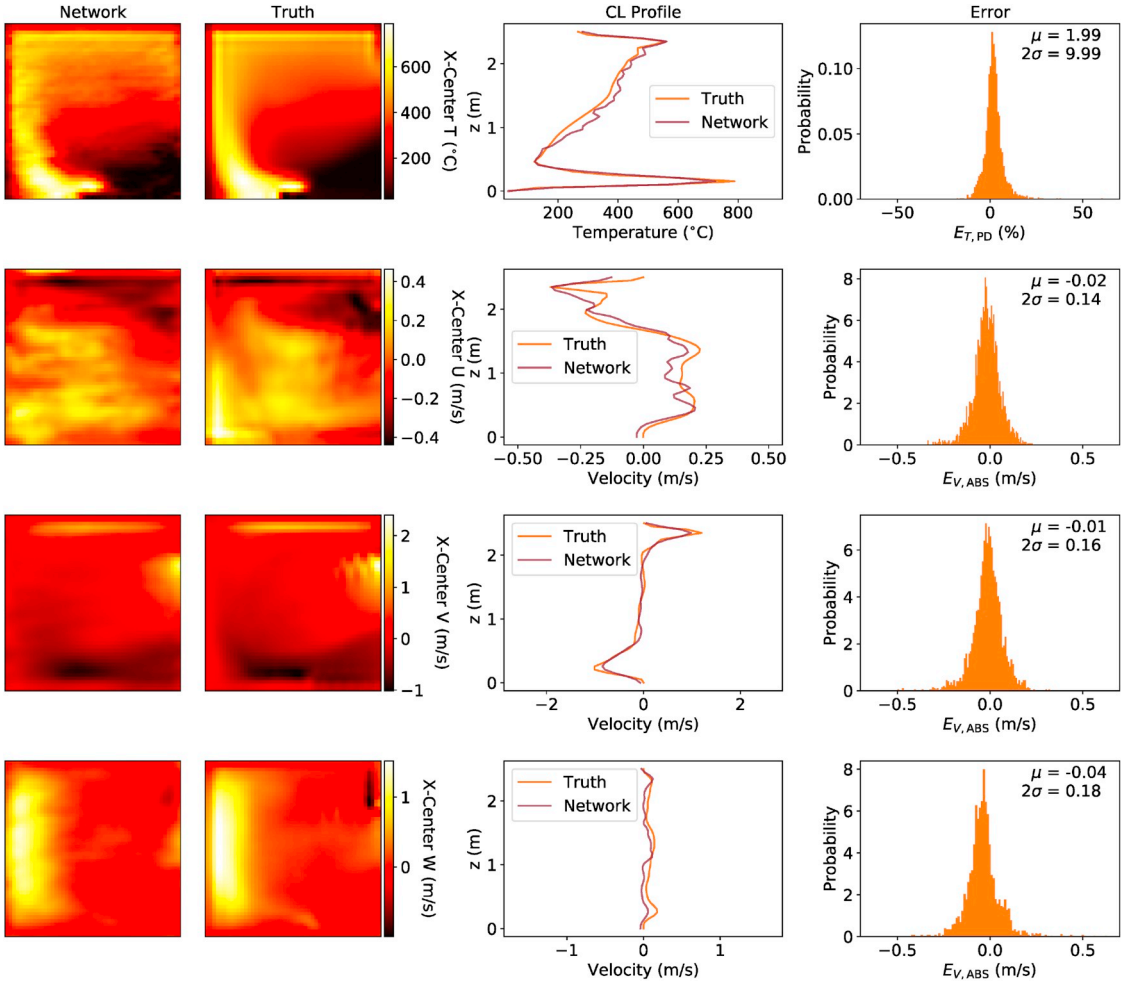


Fig. 9. TCNN predictions of x-axis center-line temperature and velocity compared to CFD simulations for Testing Case 1.

$$\dot{m}_u = \int_{z_n}^{H} \int_0^x v(x', z)\rho(x', y)dx'dz \tag{5}$$

$$\dot{m}_l = \int_0^{z_n} \int_0^x v(x', z)\rho(x', y)dx'dz \tag{6}$$

where x is the width of the ventilation in m, and ρ is the density of the air in kg/m^3. The density of the air was calculated using the equation

$$\rho = \frac{353.4}{T}. \tag{7}$$

2.4. Post processing

The approach was found to under-predict the temperature when the temperature was low. This resulted in a mean under-prediction percent error in temperature of −0.4% in the testing data. Since the ReLU activation function has an unbounded magnitude, using this function in the output layer of the neural network can lead to extrapolation errors. This error could be reduced by including ambient conditions as an input to the neural network and varying ambient conditions in the data generation. However, this would increase the number of parameters needed for the neural network to learn, which would increase the number of simulations needed to train the network. Since this study was not interested in capturing the impact of ambient conditions, any temperatures less than ambient were fixed to ambient in post-processing. This post-processing step resulted in a mean under-prediction percent error in temperature of −0.1% in the testing data.

The raw temperature and velocity predictions from the TCNN were found to have some noise which was not present in the CFD predictions, as shown in Fig. 5a. This noise can be reduced through additional training, additional convolution layers in the network, or through post-processing. In this study it was decided to use post-processing to reduce the noise so that network would not need to be re-trained. The noise was minimized by using a Gaussian filter to smooth the predictions. The standard deviation in the temperature and velocity error from the training data was used to determine the optimal standard deviation for the filter, as shown in Fig. 6. The temperature and velocity errors were found to be a minimum with a kernel size of 0.50 and 1.00 for the Gaussian kernel, respectively. While Fig. 5b shows the post-processing qualitatively improves the agreement; Fig. 6 shows the statistical

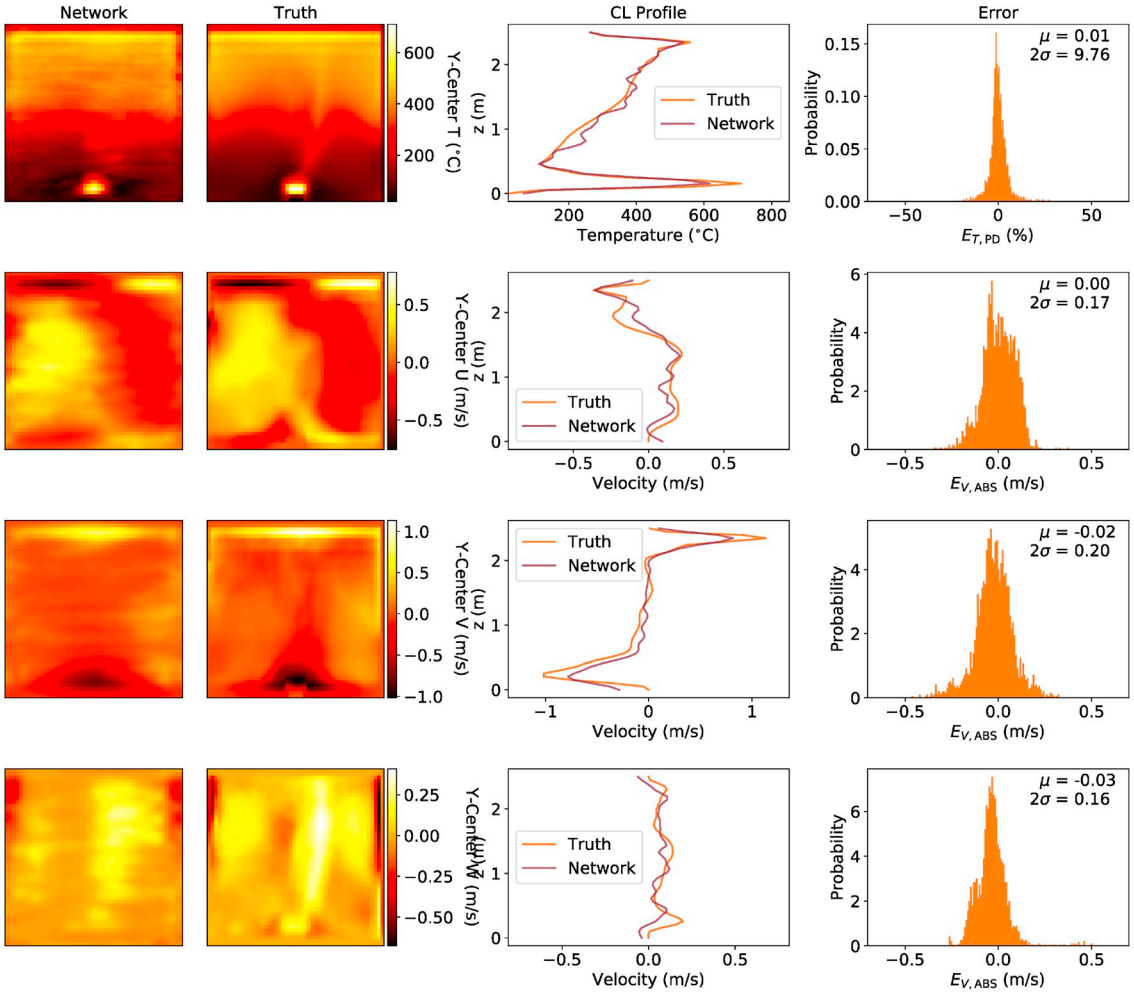


Fig. 10. TCNN predictions of y-axis center-line temperature and velocity compared to CFD simulations for Testing Case 1.

impact of the post-processing on the error is not significant.

3. Results

A total of 333 scenarios not included when training the neural network were used to test the performance on new scenarios. Each of these scenarios were randomly sampled from the same distributions as the training data, shown in Table 2; however, none of the test scenarios are the exact same as the input scenarios. The root mean square difference (RMSD) of each testing scenario from its nearest neighbor in the training set was calculated. Prior to calculating the RMSD, the inputs to the neural network for each training and testing scenario were normalized by the max and min for each field in the training set. Thus, RMSD ranges from a perfect match at 0.0, to an exact opposite match at 100.0. Realistically, an upper limit on RMSD is closer to 75 due to the limited permutations in ventilation sizes. Fig. 7 shows a histogram of the RMSD for all the testing scenarios. The mean and standard deviation of RMSD were 9.8 and 3.0, respectively.

Schematics of two scenarios are included in Fig. 8. In Test Scenario 1 the room of interest was 2.5 m × 4.9 m x 2.5 m. The room of interest was connected to a separate room on the west side which was 2.5 m × 2.5 m x 2.5 m through a 1.25 m × 1.9 m door, as well as open to ambient through a 0.625 m × 1.9 m door on the north side. The adjacent room had no additional ventilation paths. A 755 kW fire was located in the center of the room of interest. The RMSD for Test Scenario 1 was 9.5. In Test Scenario 2 the room of interest was 2.5 m × 2.5 m x 2.5 m. The room of interest was connected to a separate room on the north side which was also 2.5 m × 2.5 m x 2.5 m through a full-length hallway style opening, as well as a hallway style opening to ambient on the east side, and a 0.625 m × 1.9 m door to ambient on the south side. The adjacent room had an additional ventilation path to ambient on the north side through a hallway style opening. A 480 kW fire was located in the center of the adjacent room. The RMSD for Test Scenario 2 was 9.5.

Detailed neural network predictions of temperature and velocity are compared with CFD simulations for Test Scenario 1 and Test Scenario 2 in Figs. 9–14. In Figs. 9–14 the rows correspond to the different two-dimensional slices predicted by the network. The left column of each

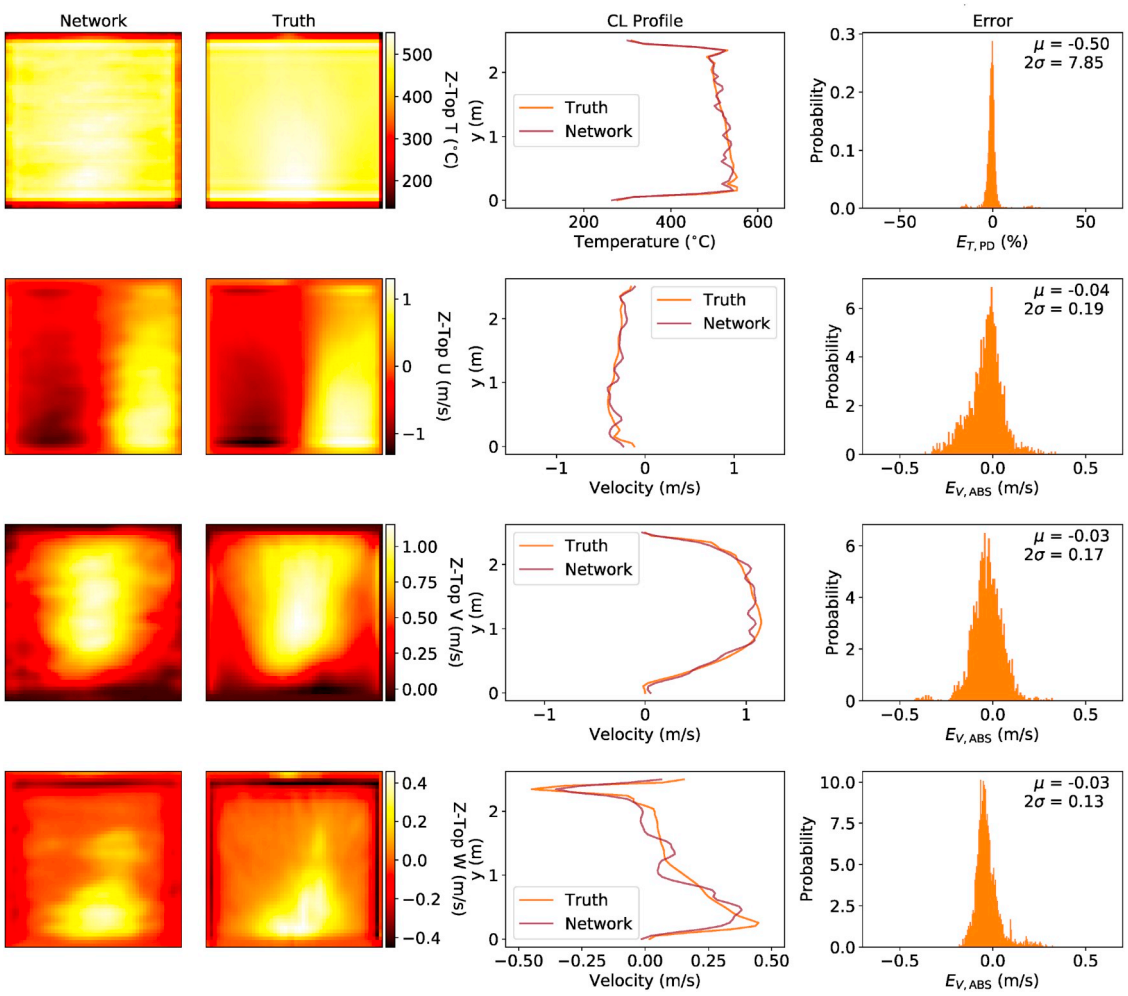


Fig. 11. TCNN predictions of z-axis ceiling temperature and velocity 0.1 m below the ceiling compared to CFD simulations for Testing Case 1.

image pair is the neural network prediction. The left-middle column of each image pair is the CFD prediction. The contour axis is scaled the same for each image pair. The right-middle column compares the centerline profile of the image pair. The histogram on the right shows the error in the neural network prediction compared to the CFD prediction. The titles correspond to the slice location and intensive property shown. Fig. 9 shows the x-axis center-line temperature and three-component velocity comparison for Testing Case 1. Fig. 10 shows the y-axis center-line temperature and three-component velocity comparison for Testing Case 1. Fig. 11 shows the z-axis temperature and three-component velocity 0.1 m below the ceiling comparison for Testing Case 1. Fig. 12 shows the x-axis center-line temperature and three-component velocity comparison for Testing Case 2. Fig. 13 shows the y-axis center-line temperature and three-component velocity comparison for Testing Case 2. Fig. 14 shows the z-axis temperature and three-component velocity 0.1 m below the ceiling comparison for Testing Case 2.

The percent error is shown for temperature predictions, calculated as

$$E_{T,\mathrm{PD}} = \frac{T_{\mathrm{cfd}} - T_{\mathrm{TCNN}}}{T_{\mathrm{cfd}}} \times 100 \tag{8}$$

where $E_{T,\mathrm{PD}}$ is the percent error, T_{TCNN} is the temperature predicted by the neural network, and T_{cfd} is the temperature predicted by CFD. Since many of the velocities are close to zero, the percent error was not a good metric to examine the robustness of the predictions. Instead the absolute error is shown for the velocity predictions, calculated as

$$E_{V,\mathrm{ABS}} = V_{\mathrm{cfd}} - V_{\mathrm{TCNN}} \tag{9}$$

where $E_{V,\mathrm{ABS}}$ is the absolute error, V_{TCNN} is the velocity predicted by the neural network, and V_{cfd} is the velocity predicted by CFD. For the examples shown in Figs. 9–14 temperatures are typically within ± 10.0% and velocities within ± 0.22 m/s.

The discrete probability density function of temperature percent error and velocity absolute error for all training and test data is shown Fig. 15 and Fig. 16, respectively. The mean and standard deviation of error from the training and test data sets is summarized in Table 3. An error of zero corresponds to perfect agreement between the ANN and

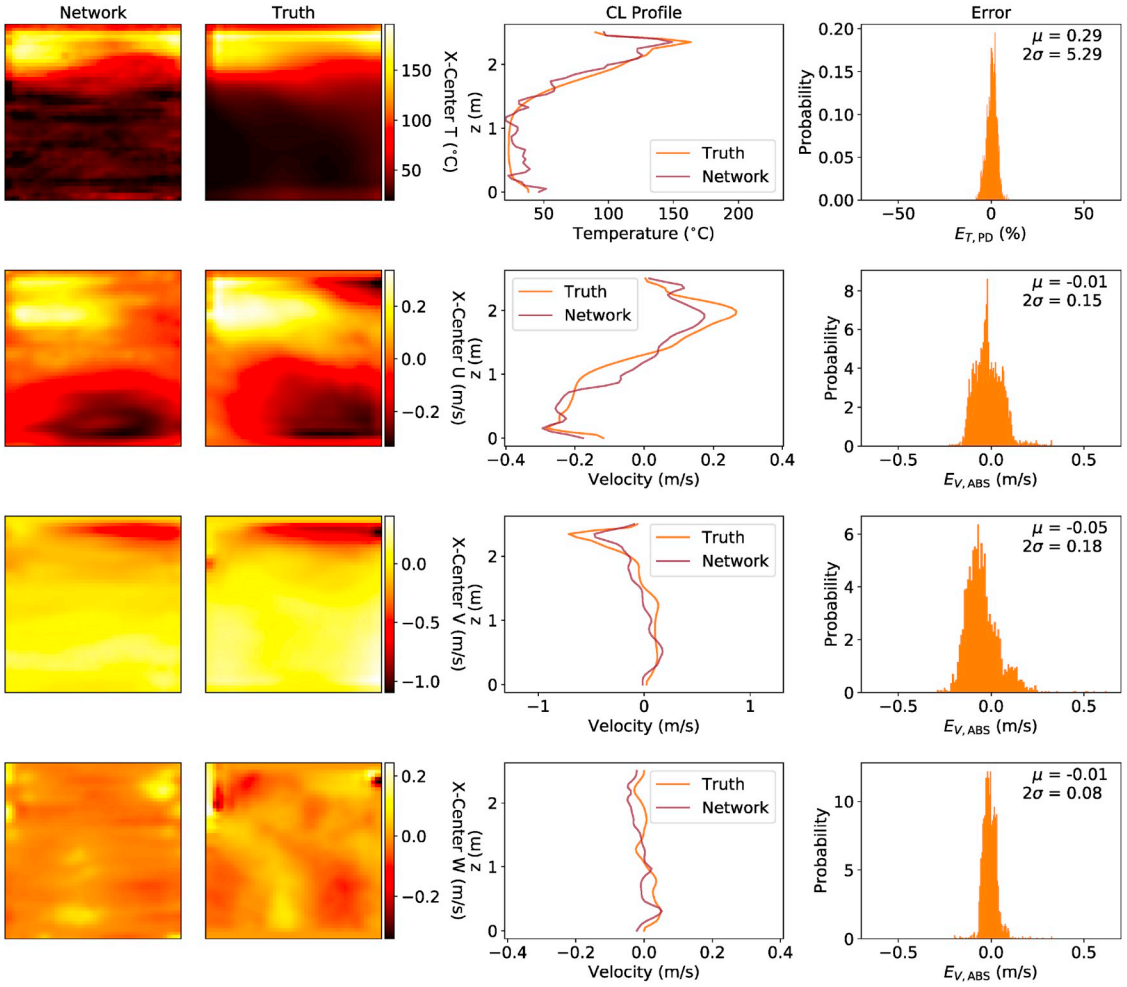


Fig. 12. TCNN predictions of x-axis center-line temperature and velocity compared to CFD simulations for Testing Case 2.

simulation predictions. Overall, the error was low for both temperature percent error and velocity absolute error.

4. Discussion

The example scenarios shown in Figs. 9–14 show the overall temperature profiles predicted using the TCNN agree with the CFD predictions. For example, the temperature in Fig. 9 shows the fire plume in the center of the room was tilted towards the left-hand side of the compartment due to the ventilation configuration. The TCNN was able to correctly predict this behavior based on the ventilation conditions at each door even though this information was not directly input to the neural network. Similarly, the neural network was able to predict the spatially resolved change in interface height shown in the x-center and y-center temperature profiles in Figs. 12 and 13. Additionally the TCNN was able to predict the presence of a cooler region in front of the door on the right hand side of the z-top temperature in Fig. 14. Table 3 shows the mean temperature percent error is less than 1.0% for each spatial slice, and the standard deviation is 2.2% for the training set and 8.0% for the test set. Based on these results, 95% of TCNN temperature predictions should be within ± 16.0% of the CFD predictions. Much of the temperature error was identified to come from two sources. One source of observed error was an under-prediction in the peak temperature when the flame directly intersects the temperature field. Since the TCNN has no a priori knowledge of the presence of the flame, the peak temperature must be inferred directly from ventilation flows. The second source of observed error was non-physical noise in the lower layer when the interface height was high, as shown in Figs. 12 and 13. No significant difference in error between the x-center, y-center, and z-top temperatures was observed.

The example scenarios shown in Figs. 9–14 highlight the capability of the method to predict complex flow fields even though the inputs to the TCNN are coarse. For example, the y-center U-velocity in Fig. 10 shows the TCNN was able to predict the multiple points of inflection in the velocity. Table 3 shows the mean velocity error is 0.02 m/s and a standard deviation of 0.05 m/s for the training set and 0.15 m/s for the test set. Based on these results, 95% of TCNN velocity predictions should be within ± 0.30 m/s of the CFD predictions. Much of the

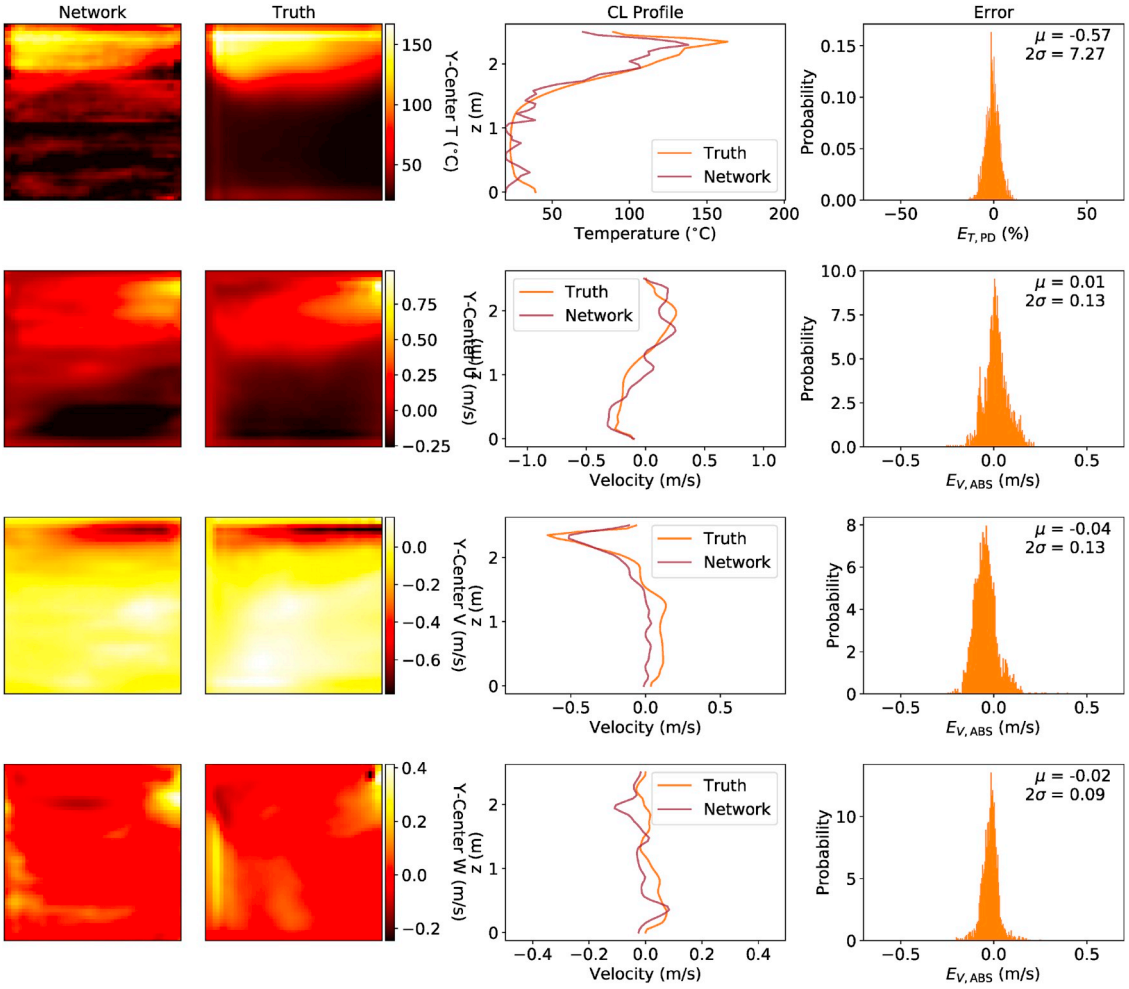


Fig. 13. TCNN predictions of y-axis center-line temperature and velocity compared to CFD simulations for Testing Case 2.

velocity error was identified to come from two primary sources. One source of observed error was a difficulty to predict recirculation regions in the ceiling above ventilation, such as the recirculation region near the ceiling on the right hand side of the x-center U-velocity in Fig. 12. The second source of observed error was a tendency to under-predict the peak velocity near boundaries as can be seen in the x-center W-velocity contour in Fig. 9. No significant difference in error between the x-center, y-center, and z-top velocities was observed.

The computational time required for each FDS simulation ranged from 1200–3600 s dependent on the fire size and size of the computational domain. The computational time required for a similar geometry in a zone fire model such as CFAST is on the order of 1–30 s [43]. The total time to predict spatially resolved temperatures and velocities with the TCNN with all 5000 scenarios was 8 s, or approximately 0.0016 s per evaluation. For a multi-room configuration, each room is processed through the TCNN independently, and the resulting predictions are stitched together. Since the predictions made with this approach are fully independent, there is no limitation on the number of compartments which can be evaluated. Since a batch size of 100 was used in this work, the computational cost to run a 1–100 room compartment configuration is similar. The computational cost will linearly increase with number of rooms for each 100 rooms to be modeled. These results show there is a negligible impact on computational cost when coupling a zone fire model with the TCNN framework.

The capability of the TCNN to predict fire scenarios beyond the two-room configuration was examined through the two multi-room validation case studies shown in Fig. 17. The validation case studies were run for a longer duration so that the impact of simulation time on the predictions could be examined. The detailed comparisons of TCNN predictions in Fig. 18 and Figs. 20–23 were based on time-averaged quantities from 10 to 120 s of fire exposure. The impact of simulation duration on the error is presented later in this section for Validation Case 2.

Validation Case 1 shown in Fig. 17a consists of four rooms. The two-left rooms are 2.5 m × 2.5 m x 2.5 m and the two-right rooms are 5.0 m × 2.5 m x 2.5 m. All doors shown in Fig. 17a are 0.625 m × 1.9 m in size. A 0.25 m × 0.25 m burner with a heat release rate of 250 kW was located in the center of the south-west room. The temperature and velocity predictions in Validation Case 1 from the TCNN are compared with CFD predictions in Fig. 18. The RMSD for each room in Validation

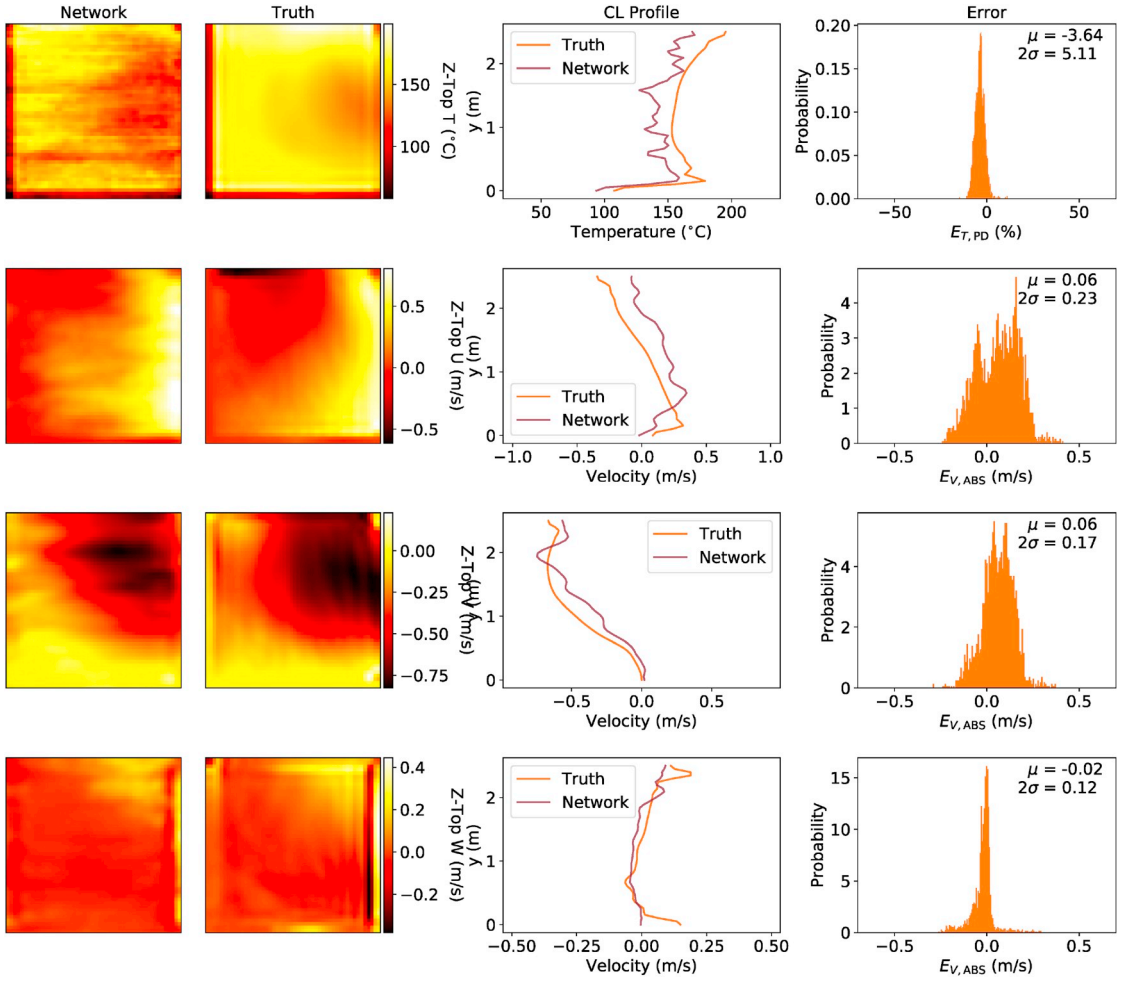
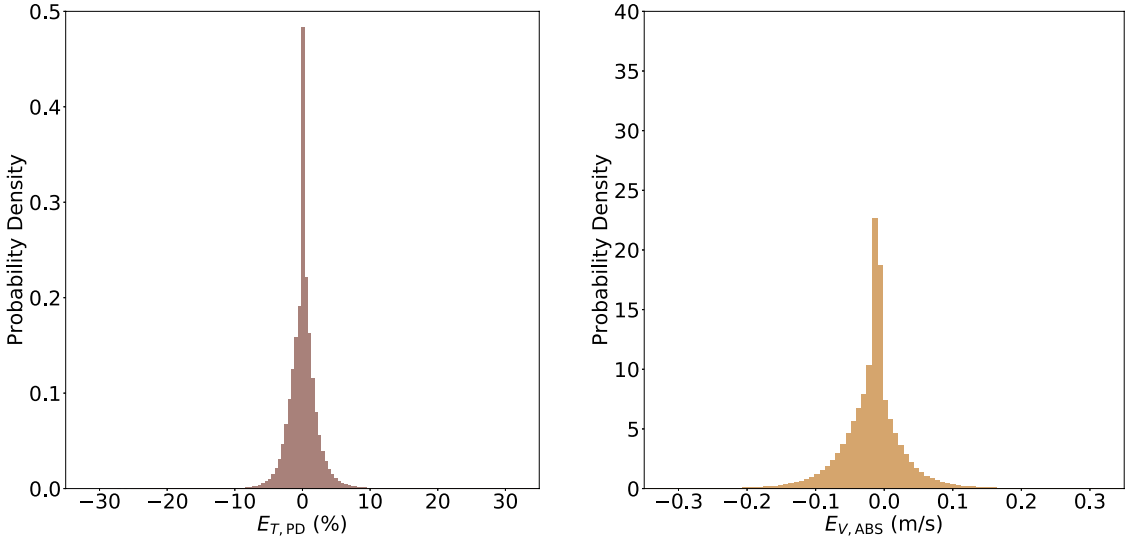
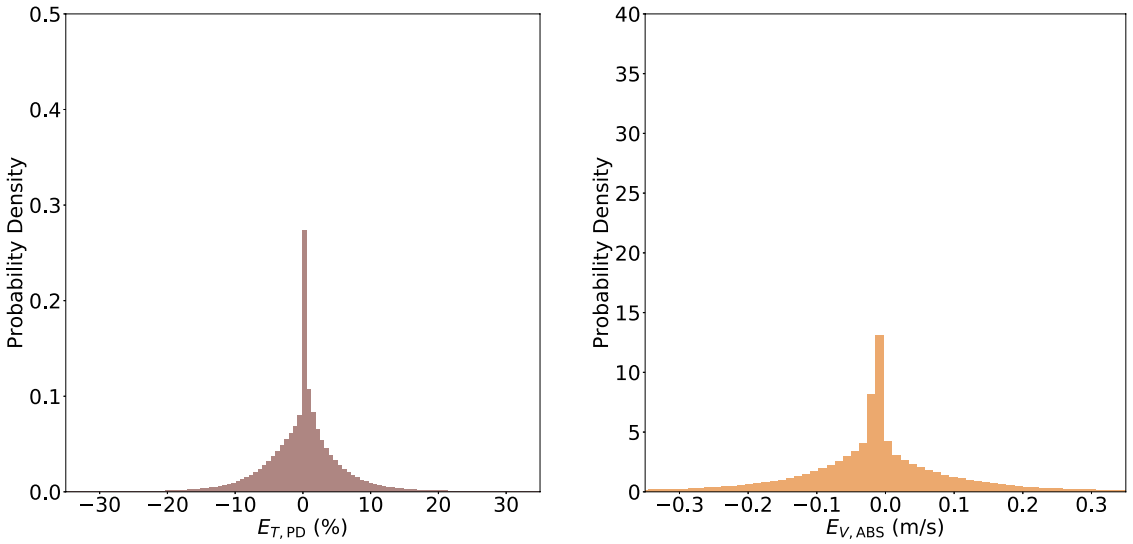


Fig. 14. TCNN predictions of z-axis ceiling temperature and velocity 0.1 m below the ceiling compared to CFD simulations for Testing Case 2.



(a) Temperature percent error. (b) Velocity absolute error.

Fig. 15. Discrete probability density functions of TCNN error from training data set.

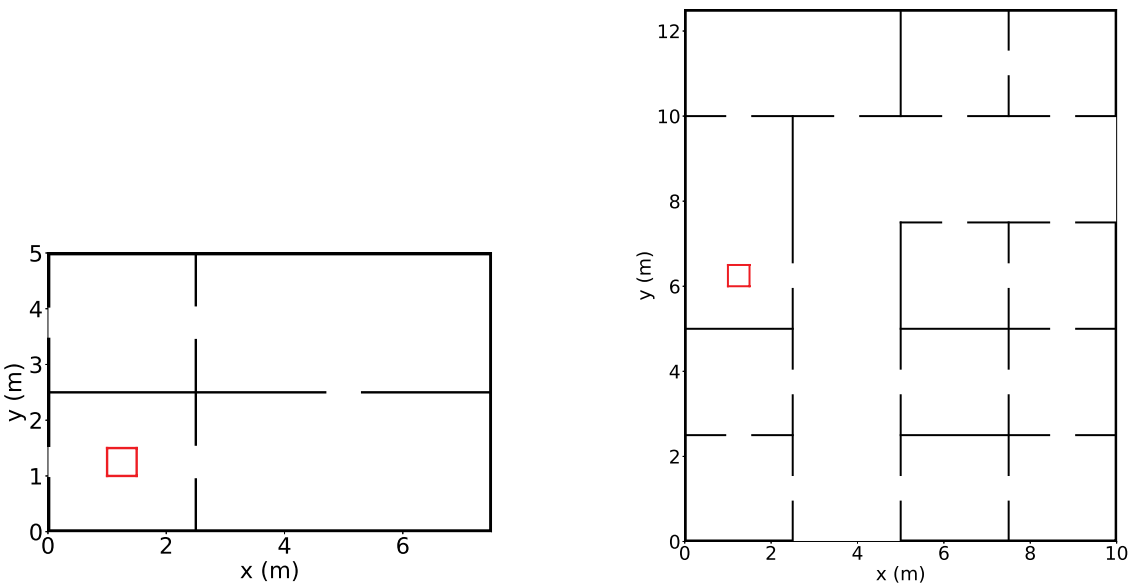


(a) Temperature percent error. (b) Velocity absolute error.

Fig. 16. Discrete probability density functions of TCNN error from testing data set.

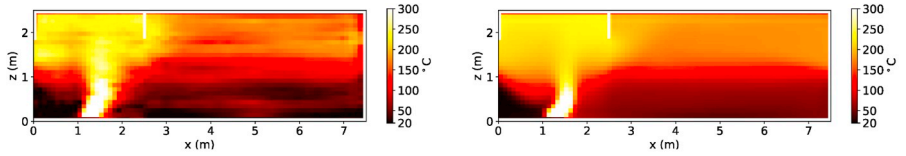
Table 3
Summary of performance of TCNN predictions on training and test data sets, values correspond to $\mu \pm 2\sigma$.

Intensive Property	Location	Training Set Error	Testing Set Error	Units
Temperature	x-center	− 0.8 ± 5.0	0.9 ± 15.6	%
Temperature	y-center	0.3 ± 4.4	− 0.3 ± 16.0	%
Temperature	z-top	0.3 ± 3.4	− 0.8 ± 8.8	%
U-velocity	x-center	0.01 ± 0.10	− 0.01 ± 0.24	m/s
U-velocity	y-center	0.01 ± 0.12	− 0.01 ± 0.26	m/s
U-velocity	z-top	0.02 ± 0.10	− 0.03 ± 0.38	m/s
V-velocity	x-center	0.02 ± 0.12	− 0.01 ± 0.24	m/s
V-velocity	y-center	0.02 ± 0.10	− 0.01 ± 0.24	m/s
V-velocity	z-top	0.02 ± 0.10	− 0.01 ± 0.38	m/s
W-velocity	x-center	0.02 ± 0.10	− 0.03 ± 0.34	m/s
W-velocity	y-center	0.02 ± 0.10	− 0.03 ± 0.32	m/s
W-velocity	z-top	0.02 ± 0.10	− 0.03 ± 0.18	m/s
Temperature	all	0.00 ± 4.4	− 0.1 ± 14.0	%
Velocity	all	0.02 ± 0.10	− 0.02 ± 0.30	m/s

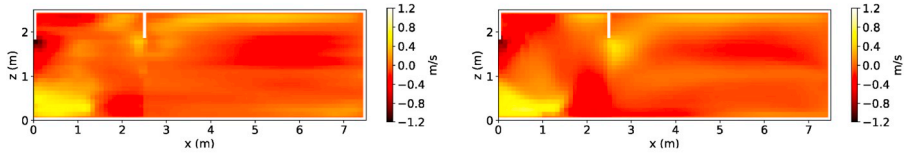


(a) Validation Case 1. (b) Validation Case 2.

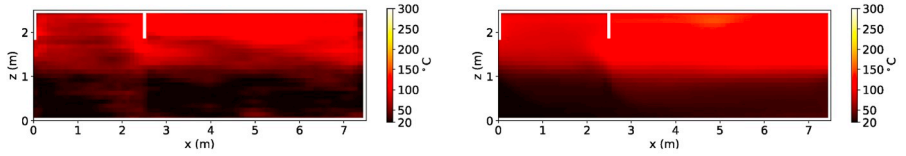
Fig. 17. Geometry from multi-room configuration validation studies. The red square is the source fire position. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)



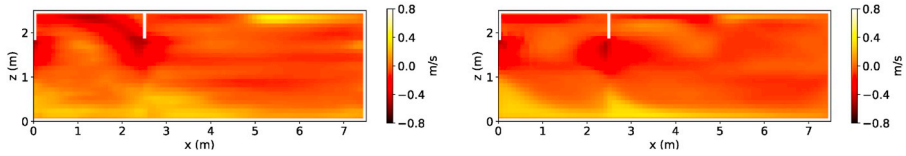
(a) Temperature at $y = 1.25$ m.



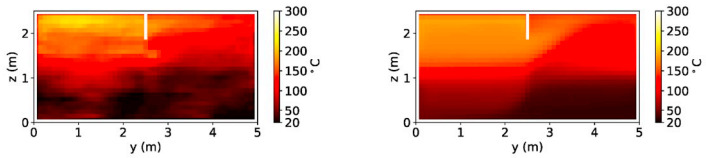
(b) U-velocity at $y = 1.25$ m.



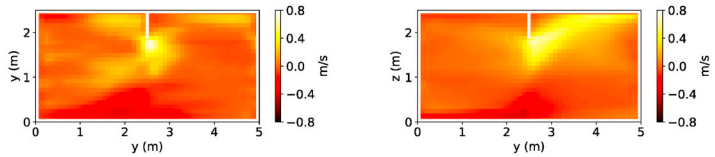
(c) Temperature at $y = 3.75$ m.



(d) U-velocity at $y = 3.75$ m.



(e) Temperature at $x = 5.00$ m.



(f) V-velocity at $x = 5.00$ m.

Fig. 18. Comparison of TCNN (left) and CFD (right) predictions in Validation Case 1. The white borders correspond to obstructions to the flow.

Case 1 to its nearest neighbor in the training set ranged from 3.6 to 7.2. Overall temperature and flow profiles agree well with the CFD predictions. Adjacent predictions from neighboring rooms are fairly continuous, although a small discontinuity can be seen in the region around $x = 2.5$ in Figs. 18a–d and $y = 2.5$ in Fig. 18e–f. The overall error in temperature and velocity is summarized for each room in Validation Case 1 in Fig. 19, where the rooms are sorted by distance to the initiating fire (0 is the south west room, 1 is the south east room, etc). The error bars in Fig. 19 correspond to 2σ. Fig. 19 shows the error in temperature is greater in the rooms without the initiating fire, but is not increasing with distance. Additionally, the error in velocity is not impacted by distance from the initiating fire.

Validation Case 2 shown in Fig. 17b represents a more complex geometric configuration than the others presented in this work. There are a total of ten 2.5 m × 2.5 m x 2.5 m rooms, two 5.0 m × 2.5 m x 2.5 m rooms, and a hallway which wraps around a corner. All doors

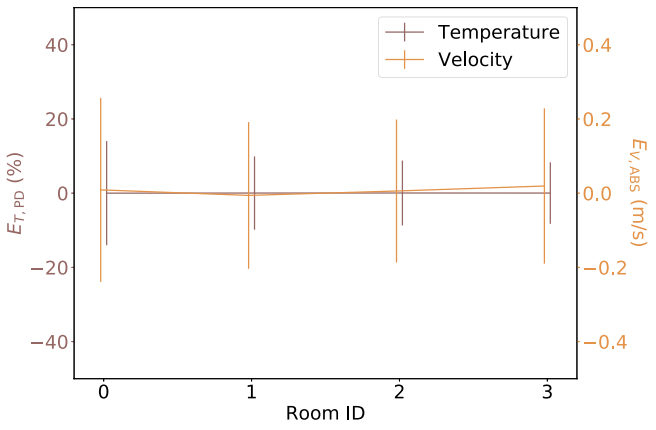
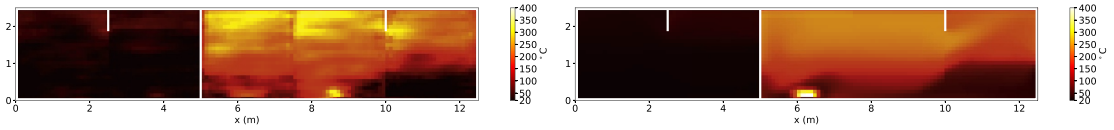
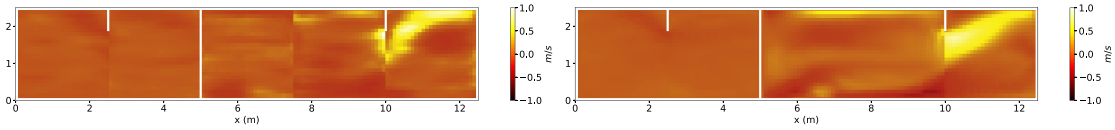


Fig. 19. Validation Case 1 comparison of change in error with distance from the fire.



(a) Temperature at $x = 1.25$ m.



(b) V-velocity at $x = 1.25$ m.

Fig. 20. Comparison of vertical slices TCNN (left) and CFD (right) predictions in Validation Case 2. The white borders correspond to obstructions to the flow.

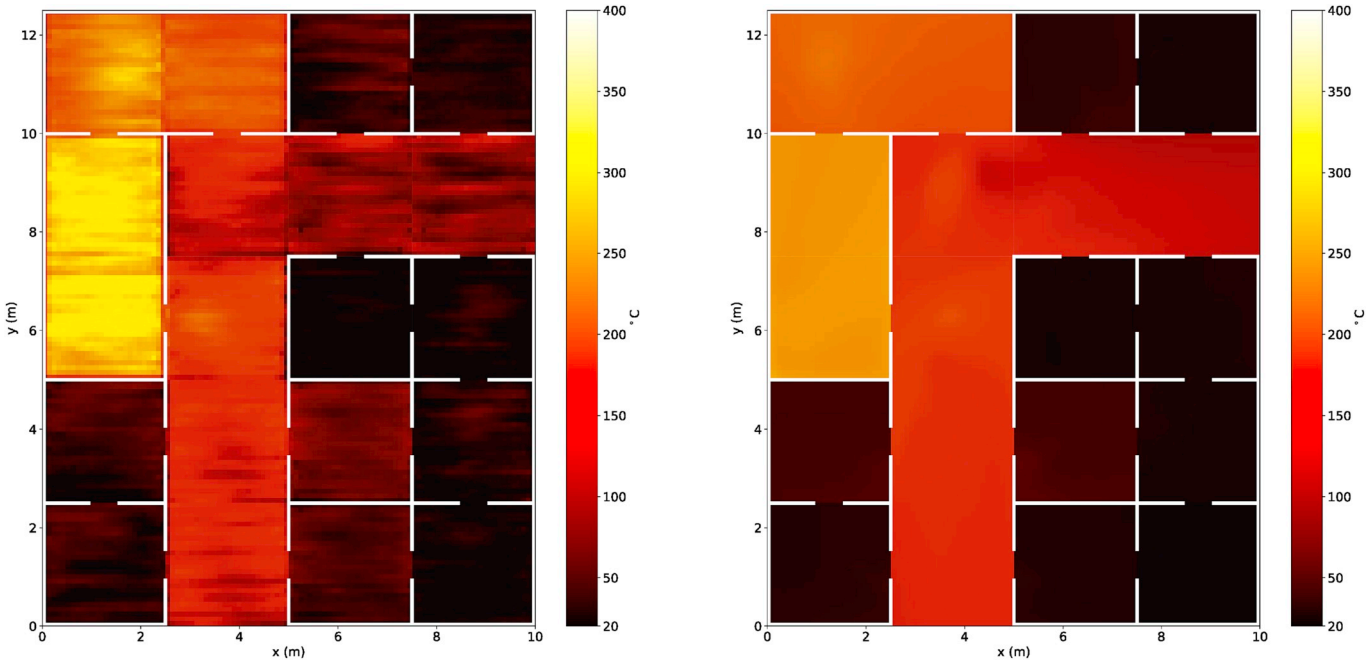


Fig. 21. Comparison TCNN (left) and CFD (right) predictions of temperature 0.1 m below the ceiling in Validation Case 2. The white borders correspond to obstructions to the flow.

shown in Fig. 17b are 0.625 m × 1.9 m in size. A 0.25 m × 0.25 m burner with a heat release rate of 250 kW was located in the position shown in Fig. 17b. Since the doors for the larger rooms and hallways are not always in the center of the room, these rooms were split into smaller zones before processing with the TCNN. Effectively, the two larger rooms were split into two adjacent smaller rooms with a full-wall ventilation path connecting them. Similarly the hallway was split into six segments. The RMSD for each room in Validation Case 2 to its nearest neighbor in the training set ranged from 8.6 to 18.3.

The temperature and velocity predictions in Validation Case 2 from the TCNN are compared with CFD predictions in Figs. 20–23. Overall the TCNN and CFD predictions of temperature are comparable to those in development and in Validation Case 1. Fig. 20a–b shows the TCNN predicted a hot spot near the floor in both zones which made up the source fire room. Otherwise, the temperature and velocity fields were well described. This shows the TCNN was able to make good predictions of the flow field even though the TCNN had not been trained with a fire source in the large split room configuration. The TCNN predicted temperatures 0.1 m below the ceiling shown in Fig. 21 were able to capture the higher temperature regions where smoke was entering the hallway. Figs. 22 and 23 show the TCNN had difficulty predicting the velocity field below the ceiling in the source fire room and large room

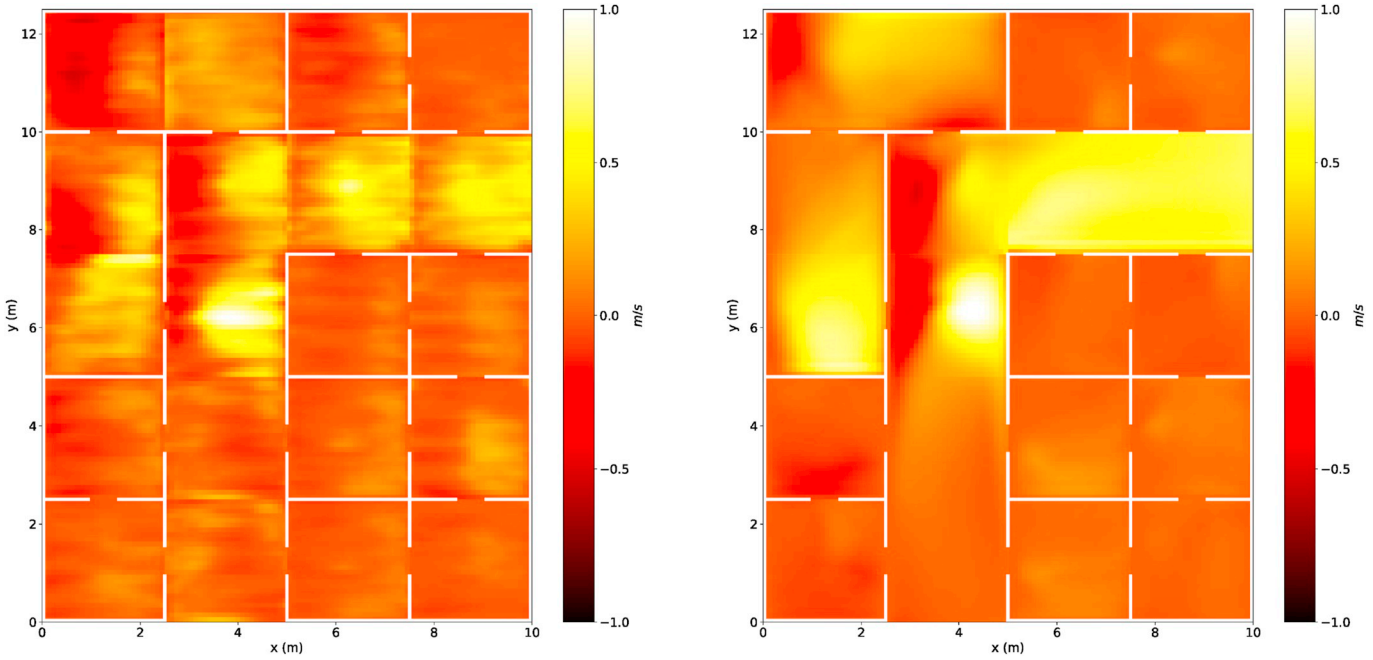


Fig. 22. Comparison TCNN (left) and CFD (right) predictions of U-velocity 0.1 m below the ceiling in Validation Case 2. The white borders correspond to obstructions to the flow.

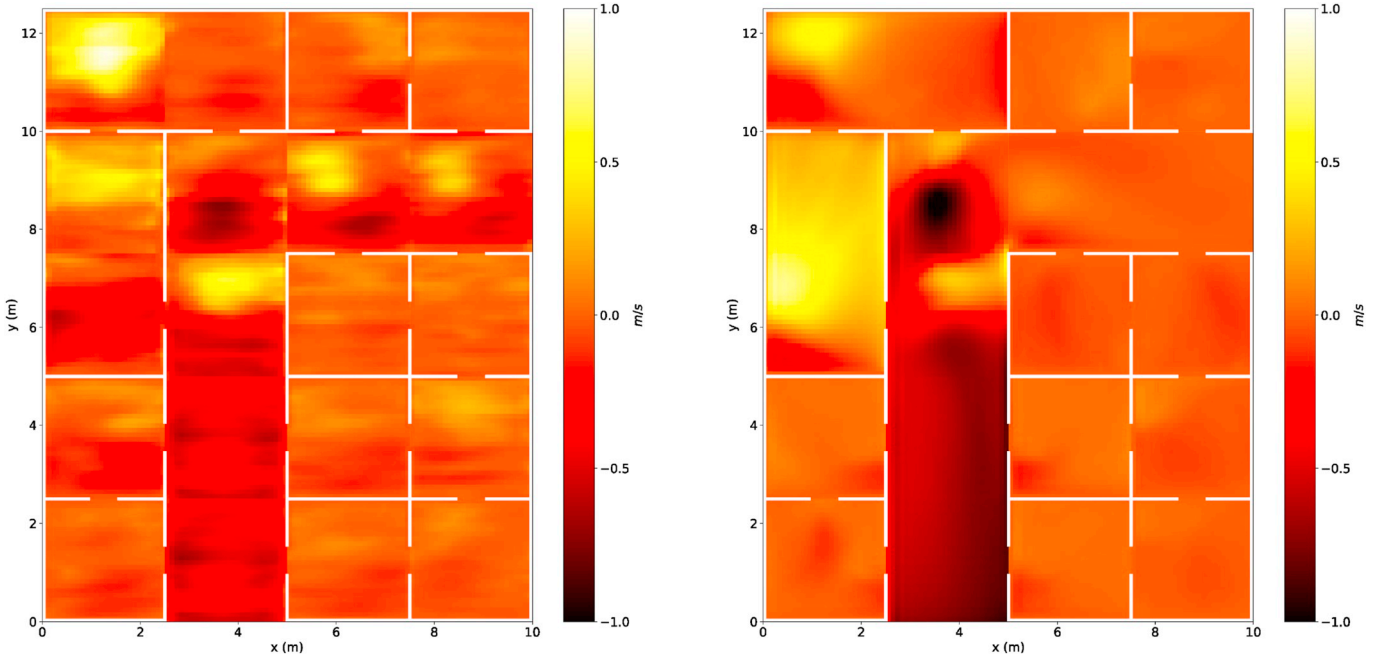


Fig. 23. Comparison TCNN (left) and CFD (right) predictions of V-velocity 0.1 m below the ceiling in Validation Case 2. The white borders correspond to obstructions to the flow.

Table 4
Summary of performance of TCNN predictions on validation cases, values correspond to $\mu \pm 2\sigma$.

Intensive Property	Location	Validation Case 1	Validation Case 2	Units
Temperature	x-center	3.0 ± 11.2	− 1.2 ± 9.8	%
Temperature	y-center	2.9 ± 10.6	0.0 ± 10.8	%
Temperature	z-top	1.9 ± 5.4	− 0.3 ± 12.2	%
U-velocity	x-center	0.01 ± 0.22	0.01 ± 0.20	m/s
U-velocity	y-center	0.00 ± 0.20	0.00 ± 0.10	m/s
U-velocity	z-top	0.04 ± 0.30	0.07 ± 0.34	m/s
V-velocity	x-center	0.01 ± 0.20	0.00 ± 0.22	m/s
V-velocity	y-center	0.00 ± 0.20	0.00 ± 0.20	m/s
V-velocity	z-top	0.00 ± 0.30	− 0.01 ± 0.44	m/s
W-velocity	x-center	0.01 ± 0.14	0.00 ± 0.16	m/s
W-velocity	y-center	− 0.02 ± 0.18	0.00 ± 0.22	m/s
W-velocity	z-top	0.02 ± 0.12	0.02 ± 0.12	m/s
Temperature	all	2.6 ± 11.0	− 0.5 ± 11.0	%
Velocity	all	0.01 ± 0.22	0.01 ± 0.24	m/s

Table 5
Comparison of computational time for CFD and TCNN predictions of validation cases.

	Computational time (seconds)	
Case	CFD Model	TCNN Model
1	9310	0.76
2	22841	0.90

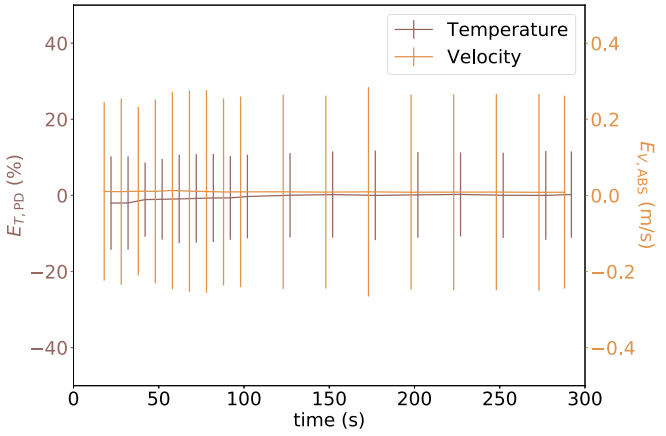


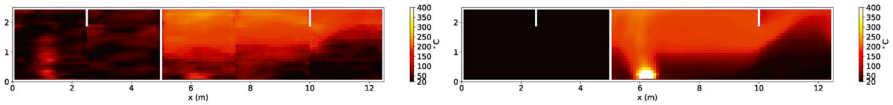
Fig. 24. Validation Case 2 comparison of change in error versus simulation time.

north of the fire source as well as in the corner zone of the hallway. Recall in the two-room configuration, the smoke flow could not alter direction more than 90°; however, the smoke flow in the large room north of the fire source enters and exits the space at a full 180° difference. This constitutes significant vorticity in the flow which was not captured in the database from the two-room configuration. To minimize these effects, the data-set could be augmented with scenarios with hot gases entering the room from multiple sources such as occurs in the hallway corner zone, or simulations with a higher number of rooms. The discontinuities in the velocity predictions between neighboring rooms are more significant in Validation Case 2 than in Validation Case 1, as shown in Figs. 22 and 23. These discontinuities are an artifact of the sampling used in generating the two-room configurations (there were few simulations used in training where the forced ventilation configuration was selected and both the west and east wall of Room 2 was a hallway-type connection). These discontinuities could be improved by augmenting the data set with scenarios with hallway type connections, or altering the approach to directly couple the predictions in adjacent rooms (either by adjusting the TCNN approach or additional post-processing).

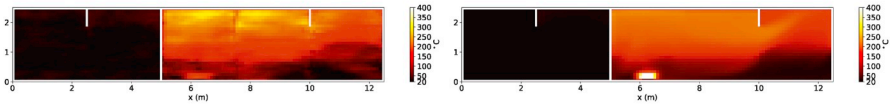
The error for each temperature and velocity field for Validation Case 1 and 2 is shown in Table 4. The mean temperature percent error was higher in Validation Case 1 at 2.6% than in Validation Case 2 at −0.5%. The standard deviation of the temperature error was the same in both Validation Case 1 and Validation Case 2 at 5.5%. The velocity error was not significantly different between the two cases. The mean absolute velocity error was the same between the two cases at 0.01 m/s. The standard deviation of the absolute velocity error was 0.11 m/s in Validation Case 1 and 0.12 m/s in Validation Case 2. The standard deviation of U-velocity and V-velocity error in the z-top slice was higher than the two vertical slices. The total computational time required to evaluate the validation cases using the CFD model and the TCNN model are shown in Table 5. Overall the TCNN is significantly faster, by a factor of 10^4.

Although the TCNN was trained on CFD predictions averaged from 10 to 30 s of exposure, the impact of time since exposure on the predictions was examined. A moving 20 s average of temperature and velocity was calculated for Validation Scenario 2 from 10 to 30 s of exposure up to 280–300 s of exposure. The progression in spatially resolved temperature predictions over time are compared with the CFD predictions in Fig. 25. The error versus time is summarized in Fig. 24. The time in Fig. 24 corresponds to the center of the moving average window. The horizontal lines correspond to the mean error, and the vertical error bars correspond to two standard deviations. Even though the TCNN was trained on exposures early in fire growth, the error of 95% of predictions of temperature and velocity are within the margins seen during testing at ± 16% for temperature and ± 0.30m/s for velocity. This shows the relationships between door flows and spatially resolved thermal flow fields the neural network learned during training are robust.

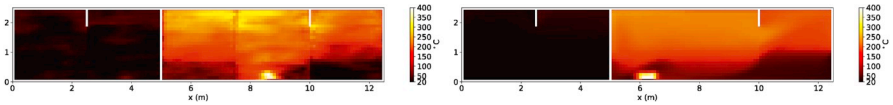
The impact on the predictions due to error in zone fire modeling was examined. The zone fire model CFAST was used to simulate each scenario in the training and testing data set which consisted of a source fire in either room 1 or room 2. The same rotation up-sampling and ambient re-sampling was applied to the simulations, resulting in 4000 training simulations and 333 testing simulations. The TCNN was initialized using the TCNN trained with spatially averaged CFD predictions rather than randomly initialized. This approach allows the new TCNN to take advantage of previously learned spatial relationships and focus on re-mapping the new set of inputs which in turn reduces the time requirements to train the network. The TCNN was trained for a total of



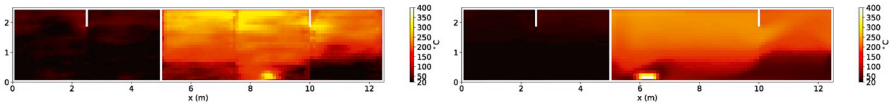
(a) Temperature at $x = 1.25$ m at $t = 20$ s.



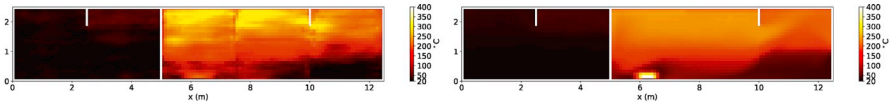
(b) Temperature at $x = 1.25$ m at $t = 40$ s.



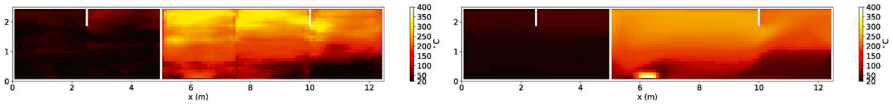
(c) Temperature at $x = 1.25$ m at $t = 60$ s.



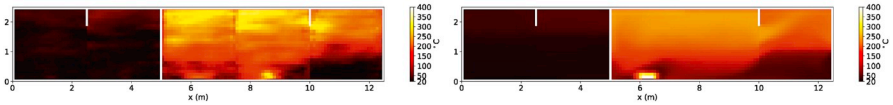
(d) Temperature at $x = 1.25$ m at $t = 80$ s.



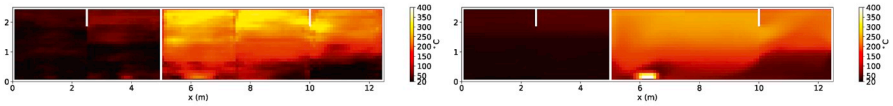
(e) Temperature at $x = 1.25$ m at $t = 100$ s.



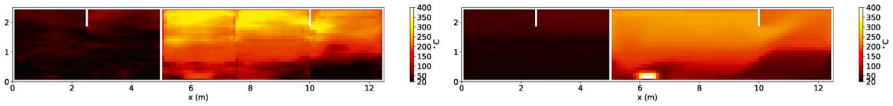
(f) Temperature at $x = 1.25$ m at $t = 150$ s.



(g) Temperature at $x = 1.25$ m at $t = 200$ s.



(h) Temperature at $x = 1.25$ m at $t = 250$ s.



(i) Temperature at $x = 1.25$ m at $t = 290$ s.

Fig. 25. Comparison of vertical slices TCNN (left) and CFD (right) predictions in Validation Case 2 at different simulation times.

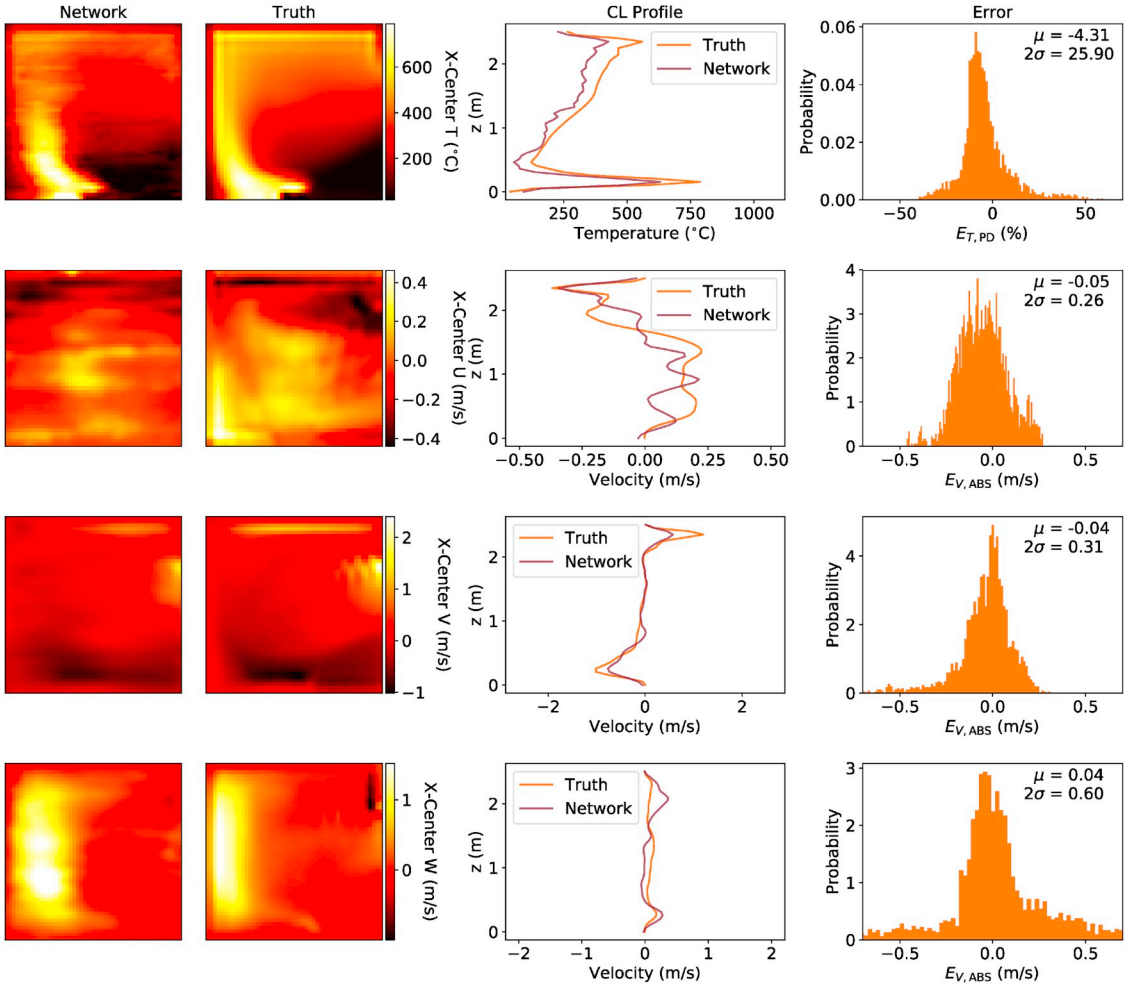


Fig. 26. Coupled CFAST-TCNN predictions of x-axis center-line temperature and velocity compared to CFD simulations for Testing Case 1.

15,000 cycles which required a total of 30 h. Predictions of Testing Cases 1 and 2 using the CFAST trained network with inputs from CFAST are shown in Figs. 26 and 27, respectively. The mean error in the testing temperature predictions was −0.6%, and 95% of temperature predictions were within ± 17.2%. The mean error in the testing velocity predictions was 0.0 m/s, and 95% of velocity predictions were within ± 0.30 m/s. The error observed using CFAST predictions as input to the TCNN was similar to the error using zone conditions produced using FDS results.

5. Conclusion

A data-driven approach was presented to predict full-field temperature and velocity within a compartment based on zero-dimensional zone fire modeling using a transpose convolutional neural network (TCNN). The network was trained using 1000 CFD simulations of a simple two-compartment configuration with different fire locations, fire sizes, ventilation configurations, and compartment geometries. The robustness of the approach was tested using 333 two-compartment CFD simulations not included when training the network. In the two-compartment test cases 95% of TCNN predicted temperatures were within ± 16.0% of CFD predictions, and 95% of TCNN predicted velocities were within ± 0.30 m/s of CFD predictions. The largest temperature error was due to under-prediction in peak temperatures and noise in the prediction of the lower layer temperatures. The largest velocity error was due to under-predicted peak velocities near boundaries and difficulty to predict some recirculation regions near ventilation. The computational time required for each CFD simulation in the database ranged from 1200–3600 s dependent on the size of the compartment. The computational time to evaluate 5000 scenarios with the TCNN framework was 8 s, which is a negligible increase in computational time when coupled with the zone fire model.

Although the model was trained and tested using a simple two-compartment configuration, the TCNN approach was validated with two more complex multi-compartment CFD simulations by processing each room individually. Overall the flow fields in the multi-compartment tests agreed well with CFD predictions with 95% of TCNN predicted temperatures within ± 11% of CFD predictions, and 95% of TCNN predicted velocities within ± 0.24 m/s of CFD predictions. The increase in velocity error observed in the multi-compartment tests is attributed to the simple two-room configuration not capturing the impact of high vorticity due to the mixing of multiple hot gas flows, and a

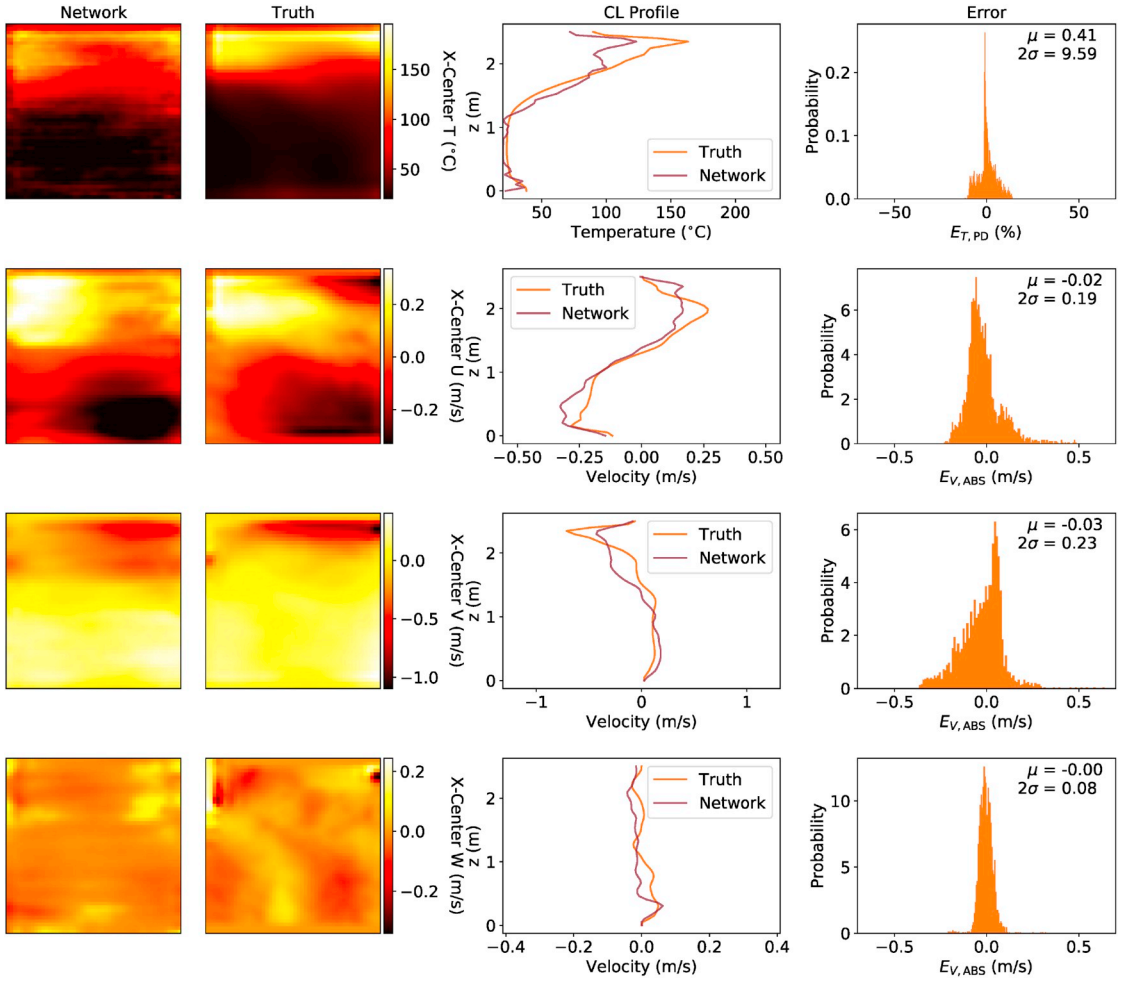


Fig. 27. Coupled CFAST-TCNN predictions of x-axis center-line temperature and velocity compared to CFD simulations for Testing Case 2.

small number of hallway-style scenarios in the database.

Although the TCNN presented in this work was not designed to capture transient behavior, the 20 s averaging window was found to be sufficient to correlate door flow to spatially resolved temperature and velocity during throughout the development of the fire. The error in Validation Case 2 was found to not vary significantly with simulation time, with the statistical limits within the margins identified during the two-compartment test cases. This shows the relationships between door flows and spatially resolved thermal flow fields the neural network learned during training are robust.

The network was originally trained and tested using spatially averaged parameters rather than zone model predictions to understand the potential of the methodology. After re-training and testing the network using zone model predictions as inputs, the overall error in spatially resolved temperature and velocity predictions did not significantly increase. This work demonstrates coupling a zone fire model with the TCNN approach provides spatially resolved temperature and velocity predictions without significantly increasing the computational requirements. Since the approach is based on a zone fire model, the TCNN approach presented in this work is limited to simplified geometries which can be sufficiently modeled using a zone fire model. In addition, future work is needed to incorporate other significant factors (such as fuel and reaction parameters, losses to boundaries, fire spread, and CFD settings) before it will be ready for use in the design phase of fire safety engineering. However, this work has shown for appropriate scenarios, the coupled zone model-TCNN approach is able to provide detailed flow fields 1000–10,000x faster than a full CFD model.

The applicability of this technique to the multiple room case is particularly promising for eventual use in fire safety design and risk analysis in large built structures which rely on repetitive geometry in terms of topology (e.g., tunnel systems, underground storage facilities, mines, etc.). However, additional development work is needed to incorporate additional physics (such as fuel and reaction parameters, boundary conditions, etc) into the model before it will be ready for use in design. While the technique presented here will still require considerable computational time up front if the conditions to be modeled lie outside the parametric study used to generate the training data, the need to generate new data and re-train will decrease over time as more data is incorporated into the model. While there is no replacement for experimental measurements and high resolution CFD in understanding complex physics and behavior of fire in complex geometries, this methodology can provide additional details from a zone fire model

without increasing computational overhead. This additional knowledge can be used to better screen out scenarios in parametric studies, highlight complex flow fields which are not well represented by a zone model, and improve the risk assessment of large structures.

Acknowledgments

This research was supported by Contract No. 200-2014-59669, awarded by the National Institute for Occupational Safety and Health (NIOSH). The findings and conclusions in this report are those of the authors and do not reflect the official policies of the Department of Health and Human Services; nor does mention of trade names, commercial practices, or organizations imply endorsement by the U.S. Government.

References

[1] K. McGrattan, S. Miles, Modeling Fires Using Computational Fluid Dynamics (CFD), SFPE Handbook of Fire Protection Engineering, (2016), pp. 1034–1065.
[2] R.D. Peacock, K. McGrattan, G.P. Forney, P.A. Reneke, CFAST-consolidated Fire and Smoke Transport (Version 7)-volume 3: Verification and Validation Guide, NIST Technical Note 1889v3, National Institute of Standards and Technology, Gaithersburg, MD, 2015 2015.
[3] L. Wang, Q. Chen, Theoretical and numerical studies of coupling multizone and CFD models for building air distribution simulations, Indoor Air 17 (5) (2007) 348.
[4] Y. Ishida, S. Kato, Method for coupling three-dimensional transient pollutant transport into one-dimensional transport simulation based on concentration response factor, ASHRAE Transact. 114 (2008) 259.
[5] J. Clarke, J. Hensen, C. Negrao, Predicting indoor airflow by combining network, cfd, and thermal simulation, 16th AIVC Conf. "Implementing the Results of Ventilation Research", 1995, pp. 145–154.
[6] C.O. Negrão, Integration of computational fluid dynamics with building thermal and mass flow simulation, Energy Build. 27 (2) (1998) 155.
[7] M. Bartak, I. Beausoleil-Morrison, J. Clarke, J. Denev, F. Drkal, M. Lain, I. Macdonald, A. Melikov, Z. Popiolek, P. Stankov, Integrating CFD and building simulation, Build. Environ. 37 (8–9) (2002) 865.
[8] E. Djunaedy, J. Hensen, M. Loomans, External coupling between CFD and energy simulation: implementation and validation, ASHRAE Transact. 111 (1) (2005) 612.
[9] K. Hiyama, S. Kato, Integration of three-dimensional CFD results into energy simulations utilizing an advection–diffusion response factor, Energy Build. 43 (10) (2011) 2752.
[10] W. Zhang, K. Hiyama, S. Kato, Y. Ishida, Building energy simulation considering spatial temperature distribution for nonuniform indoor environment, Build. Environ. 63 (2013) 89.
[11] L. Wang, Q. Chen, Applications of a coupled multizone-CFD model to calculate airflow and contaminant dispersion in built environments for emergency management, HVAC R Res. 14 (6) (2008) 925.
[12] F. Colella, G. Rein, R. Borchiellini, R. Carvel, J.L. Torero, V. Verda, Calculation and design of tunnel ventilation systems using a two-scale modelling approach, Build. Environ. 44 (12) (2009) 2357.
[13] J. Floyd, Coupling a network hvac model to a computational fluid dynamics model using large eddy simulation, Fire Saf. Sci. 10 (2011) 459.
[14] F. Colella, G. Rein, V. Verda, R. Borchiellini, Multiscale modeling of transient flows from fire and ventilation in long tunnels, Comput. Fluids 51 (1) (2011) 16.
[15] F. Colella, G. Rein, R. Borchiellini, J.L. Torero, A novel multiscale methodology for simulating tunnel ventilation flows during fires, Fire Technol. 47 (1) (2011) 221.
[16] M. Cha, S. Han, J. Lee, B. Choi, A virtual reality based fire training simulator integrated with fire dynamics data, Fire Saf. J. 50 (2012) 12.
[17] I. Vermesi, G. Rein, F. Colella, M. Valkvist, G. Jomaas, Reducing the computational requirements for simulating tunnel fires by combining multiscale modelling and multiple processor calculation, Tunn. Undergr. Space Technol. 64 (2017) 146.
[18] A. Haghighat, K. Luxbacher, B.Y. Lattimer, Development of a Methodology for Interface Boundary Selection in the Multiscale Road Tunnel Fire Simulations, Fire Technology, 2018, pp. 1–38.
[19] T.P. Miyanawala, R.K. Jaiman, An Efficient Deep Learning Technique for the Navier-Stokes Equations: Application to Unsteady Wake Flow Dynamics, (2017) arXiv preprint arXiv:1710.09099.
[20] S. Lee, D. You, Prediction of Laminar Vortex Shedding over a Cylinder Using Deep Learning, (2017) arXiv preprint arXiv:1712.07854.
[21] S. Lee, D. You, Data-driven Prediction of Unsteady Flow Fields over a Circular Cylinder Using Deep Learning, (2018) arXiv preprint arXiv:1804.06076.
[22] M. Raissi, A. Yazdani, G.E. Karniadakis, Hidden Fluid Mechanics: A Navier-Stokes Informed Deep Learning Framework for Assimilating Flow Visualization Data, (2018), p. 04327 arXiv preprint arXiv:1808.
[23] R. Maulik, O. San, A neural network approach for the blind deconvolution of turbulent flows, J. Fluid Mech. 831 (2017) 151.
[24] J. Lee, Deep Learning Approach in Multi-Scale Prediction of Turbulent Mixing-Layer, (2018), p. 07021 arXiv preprint arXiv:1809.
[25] M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics Informed Deep Learning (Part I): Data-Driven Solutions of Nonlinear Partial Differential Equations, (2017) arXiv preprint arXiv:1711.10561.
[26] M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics Informed Deep Learning (Part Ii): Data-Driven Discovery of Nonlinear Partial Differential Equations, (2017) arXiv preprint arXiv:1711.10566.
[27] S. Mishra, A Machine Learning Framework for Data Driven Acceleration of Computations of Differential Equations, (2018), p. 09519 arXiv preprint arXiv:1807.
[28] C.W. Chang, N.T. Dinh, Classification of machine learning frameworks for data-driven thermal fluid models, Int. J. Therm. Sci. 135 (2019) 559.
[29] K. Gregor, I. Danihelka, A. Graves, D.J. Rezende, D. Wierstra, Draw: A Recurrent Neural Network for Image Generation, (2015), p. 04623 arXiv preprint arXiv:1502.
[30] A. Dosovitskiy, J.T. Springenberg, M. Tatarchenko, T. Brox, Learning to generate chairs, tables and cars with convolutional networks, IEEE Trans. Pattern Anal. Mach. Intell. 39 (4) (2017) 692.
[31] G.E. Hinton, R.R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science 313 (5786) (2006) 504.
[32] R. Memisevic, G. Hinton, Unsupervised learning of image transformations, IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8.
[33] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, Adv. Neural Inf. Process. Syst. (2014) 2672–2680.
[34] H. Lee, R. Grosse, R. Ranganath, A.Y. Ng, Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations, Proceedings of the 26th Annual International Conference on Machine Learning, 2009, pp. 609–616.
[35] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, J. Xiao, 3D ShapeNets: a deep representation for volumetric shapes, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1912–1920.
[36] A. Dosovitskiy, J. Tobias Springenberg, T. Brox, Learning to generate chairs with convolutional neural networks, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1538–1546.
[37] K. McGrattan, S. Hostikka, R. McDermott, J. Floyd, M. Vanella, Fire dynamics simulator (version 6), technical reference guide, NIST - Spec. Publ. 1018 (6) (2018).
[38] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (1) (2014) 1929.
[39] A.L. Maas, A.Y. Hannun, A.Y. Ng, Rectifier nonlinearities improve neural network acoustic models, Proceedings of the 30th International Conference on Machine Learning, 30 2013, p. 3.
[40] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: surpassing human-level performance on imagenet classification, Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1026–1034.
[41] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G.S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng TensorFlow, Large-scale machine learning on heterogeneous systems, https://www.tensorflow.org/. Software available from tensorflow.org, (2015).
[42] M. Janssens, H.C. Tran, Data reduction of room tests for zone model validation, J. Fire Sci. 10 (6) (1992) 528.
[43] R.D. Peacock, W. Jones, P. Reneke, G. Forney, CFAST-consolidated Model of Fire Growth and Smoke Transport (Version 6) User's Guide, NIST Special Publication, 2005, p. 1041.