Delft University of Technology

Gas leakage detection using spatial and temporal neural network model

Kopbayev, Alibek; Khan, Faisal; Yang, Ming; Halim, Syeda Zohra

DOI
10.1016/j.psep.2022.03.002

Publication date
2022

Document Version
Final published version

Published in
Process Safety and Environmental Protection

Citation (APA)
Kopbayev, A., Khan, F., Yang, M., & Halim, S. Z. (2022). Gas leakage detection using spatial and temporal neural network model. *Process Safety and Environmental Protection*, *160*, 968-975. https://doi.org/10.1016/j.psep.2022.03.002

Important note
To cite this publication, please use the final published version (if applicable). Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

Green Open Access added to TU Delft Institutional Repository

'You share, we take care!' - Taverne project

https://www.openaccess.nl/en/you-share-we-take-care

Otherwise as indicated in the copyright section: the publisher is the copyright holder of this work and the author uses the Dutch legislation to make this work public.

Process Safety and Environmental Protection 160 (2022) 968–975

Contents lists available at ScienceDirect

Process Safety and Environmental Protection

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

Gas leakage detection using spatial and temporal neural network model

Alibek Kopbayev[a], Faisal Khan[a,b,*], Ming Yang[c], Syeda Zohra Halim[b]

[a] Centre for Risk, Integrity and Safety Engineering (C-RISE),Faculty of Engineering and Applied Science, Memorial University of Newfoundland, St. John's A1B 3×5, NL, Canada
[b] Mary Kay O'Connor Process Safety Center, Artie McFerrin Department of Chemical Engineering Texas A&M University, College Station, TX, USA
[c] Safety and Security Sciences Section, Faculty of Technology, Policy and Management, TU Delft, the Netherlands

ARTICLE INFO

Article history:
Received 9 December 2021
Received in revised form 1 March 2022
Accepted 1 March 2022
Available online xxxx

Keywords:
Gas leak detection
Convolutional neural network
Gas leak
Safety design

ABSTRACT

Natural gas leakage can impose significant danger on a facility and its surrounding communities. Methods for early detection and diagnosis of such leakages have been developed and widely used for gas pipelines and storage tanks. Most techniques include inspection of sensor-aided mathematical models. Application of machine learning techniques to gas leakage detection has been rarely explored. In the present work, convolutional network (to model spatial likelihood of leak) is combined with bi-directional long short-term memory layer network, or BiLSTM (to model temporal dependence of leak likelihood) to perform leak detection and diagnosis. The developed model was trained and tested using sequence of concentration profiles generated using open-source simulated data. The model learned successfully to predict gas leakage and classify its size. The study also explores the flexibility of this network to perform quick detection and diagnose with the limited data. While the networks did not require parameter adjustments to achieve high prediction accuracy, further optimization is possible through data selection and pre-processing. The model needs to be further tested for wide range of leak scenarios. At its present condition, the combined application of convolutional network and BiLSTM shows promising results for early and accurate leak detection in natural gas facilities. Experimental results are needed to confirm the effectiveness of the model and data uncertainty.

© 2022 Institution of Chemical Engineers. Published by Elsevier Ltd. All rights reserved.

1. Introduction

Natural gas has become an integral part of society. It is widely used for residential and industrial purposes. Many systems were built to sustain the demand, such as pipeline networks, loading and receiving terminals, storage vessels. All these systems are prone to deterioration due to corrosion or aging, and this will become more prevalent as time goes on. This is associated with one of the primary issues in the chemical industry – gas leakage (Eckerman, 2005). The leakage of natural gas can result in severe environmental impact as it is a significant greenhouse gas. In addition, it can result in intoxication, suffocation, or explosion, which results in damage to human health, property, reputation, and finances (Bonvicini et al., 2015). To avoid these consequences, the safety of these systems must be developed. While inspection and maintenance can improve the reliability of pipelines and storage vessels, it is practically impossible to avoid gas leaks. Therefore, an early response plan needs to be put in place to reveal the leak and prevent escalation (Datta and Sarkar, 2016). The first and crucial step is the detection of gas leakage, which has attracted a lot of attention from industry and research lately.

Commonly used gas leakage detection methods rely on the manual inspection of pipelines and vessels. These methods require significant investments, time and labor but are not efficient in nature. With increasing pipeline distance and plant structure complexity, the effectiveness of these manual techniques is further reduced. Other methods include monitoring process parameters, such as pressure, temperature, and flowrate (Xiao et al., 2018). These methods heavily rely on proper data acquisition and the accuracy of the mathematical models used (Doshmanziari et al., 2020). Specialized sensors are used for this purpose, such as acoustic, optical, electrochemical, and so on (Meribout et al., 2020). Use of image processing methods for gas leakage monitoring via infrared cameras have been suggested. This can detect methane molecules on the infrared spectrum (Fahimipirehgalin et al., 2021; Vollmer and Möllmann, 2017).

In recent years, with the rapid development of machine learning techniques, neural networks have become popular in outlier detection. Much work has been published in regards to the application of

* Corresponding author.
E-mail address: fikhan@tamu.edu (F. Khan).

https://doi.org/10.1016/j.psep.2022.03.002
0957-5820/© 2022 Institution of Chemical Engineers. Published by Elsevier Ltd. All rights reserved.

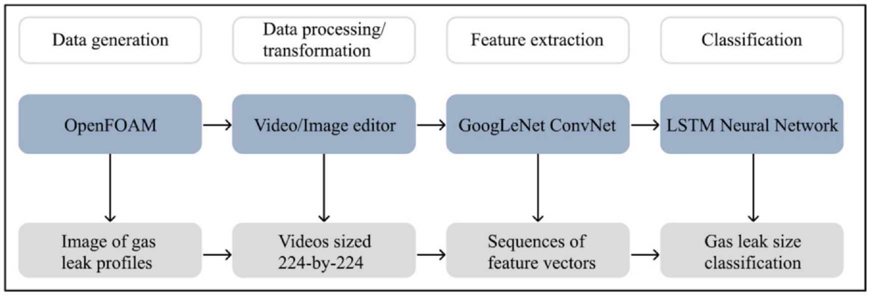

Fig. 1. Methodology for gas leak detection using a combination of ConvNet and LSTM.

neural networks for the detection of gas leaks (Ning et al., 2021; Pérez-Pérez et al., 2021; S. J. Song and Jang, 2018; Y. Song and Li, 2021; Travis et al., 2020; Wang et al., 2021). These focus mainly on pipeline leak detection and involve various types of neural networks, such as convolutional neural networks, recurrent neural networks, hybrid networks, etc. For example, Wang et al. used artificial neural network to enhance the precision of frequency analysis in pipeline leak detection (Wang et al., 2021). The trained model achieved high accuracy but required a series of filters and signal transformation techniques. In another work, artificial neural network was used to predict near real-time gas leakage at a testing site using both simulated and field data (Travis et al., 2020). However, as in many cases with machine learning, this model was highly dependent on sensor data, and interference of unexpected winds caused the model to overestimate the leak rates by a significant factor. A separate work that investigated the application of artificial neural network in pipe leak detection achieved high accuracy in the detection of leakages and their locations (Pérez-Pérez et al., 2021). However, such accuracy was achieved only under pressurized flow and was highly dependent on network configuration.

A combined network of convolutional layer and convolutional long short-term memory (LSTM) layer was employed to predict leakage locations in an enclosed space (D. Song et al., 2021). In this work, the convolutional layer extracted spatial representations, while the LSTM layer extracted temporal representations. The model successfully adapted to the building layout but did not account for the leakage size and used input data not verified by an experiment. The authors did not utilize a "healthy" state where no leakage occurred, and misclassifications were not explored. In another work, convolutional network was used as a primary detection mechanism for gas leakages in galvanized steel pipes (Song and Li, 2021). The authors investigated many network architectures to see the effectiveness in separating leak detection from internal flow noise in the frequency domain. The difficulty of this model lied in the necessity of substantial data denoising and preprocessing, and it required input produced by a specific sensor. Another work used convolutional network as both feature extractor and classifier in a gas pipeline setting (Ning et al., 2021). The network was aided by spectrum enhancement, which improved prediction accuracy and reduced training times. It also uses sound signals which may not be available for every pipe/vessel in an industrial setting and covers a limited number of leakage types.

Given that leak of natural gas, as in an LNG terminal, can be a time-critical issue, its early detection and classification of the release size becomes very important from an accident escalation and mitigation perspective. While detection of the gas leak is a primary objective in risk mitigation, classifying the leak can provide insight into the severity of the accident. Hence, if the leak cannot be brought to a stop in a short time, knowing the size of the leak may help in developing further measures, such as emergency plant shutdown or evacuation of nearby cities. In some scenarios, the classification of a leak may indicate whether the leak can be managed quickly or not. Leak classification can also be used in models that estimate risk. Since risk consists of accident probability and severity, leak classification can be used for severity assessment. Visual data, for example, from infrared camera, can be utilized through machine learning to serve this purpose. However, only limited work has been done to analyze visual data with neural networks and its application to a leak scenario in a plant setting. While studies by Song et al. and Ning et al. achieved significant leak detection accuracy, the visual input data was not used in any of their works.

This work proposes a combined model that primarily focuses on leakage detection and classification in case of a receiving terminal. It uses a pretrained GoogLeNet convolutional network as a feature extraction tool and LSTM as a classification tool. Training and testing data were generated using a CFD software that was previously validated by field tests. This model employs visual input data and tries to optimize the speed of leak detection to reduce escalation and damage. The next part of this paper will explain the data generation procedure, the role of neural networks. Classification results will be presented in the following section, and different setups will be evaluated to optimize leak detection speed. Limitations of the proposed model will be discussed, and suggestions for future work will be given.

2. Methodology

2.1. Process overview

Fig. 1 shows a four-step process for leak detection and classification: data generation using OpenFOAM, data processing and transformation, feature extraction using GoogLeNet, and gas leakage classification using the LSTM neural network. OpenFOAM software was used to generate a gas leak scenario in a receiving terminal and allowed for tracking the changes that happen within the first 50 s of the accident. It generated a series of images that were then transformed into videoclips via MATLAB functions. The major limitation of this study is that it is based on the generated data, which is difficult to obtain in a live receiving terminal. However, it is possible to link real plant data (Wu et al., 2021) to the generated profiles for further analysis. Another area of application may lie in digital twin technology, where digital twins can generate such profiles, and the hybrid model can be used for leak detection and classification.

Video footage is then fed into GoogLeNet convolutional network to extract sequences of feature vectors. These sequences of features are used in bidirectional long short-term memory layer neural network (BiLSTM) training to learn the classification of those sequences into gas leak categories. This model utilized the suitability of convolutional networks for feature extraction from visual data, making the network training to produce more accurate results. Since data generated from OpenFOAM was proven reliable based on previous studies (Fiates and Vianna, 2016; Wu et al., 2021), the combined model showed promise for producing accurate results.

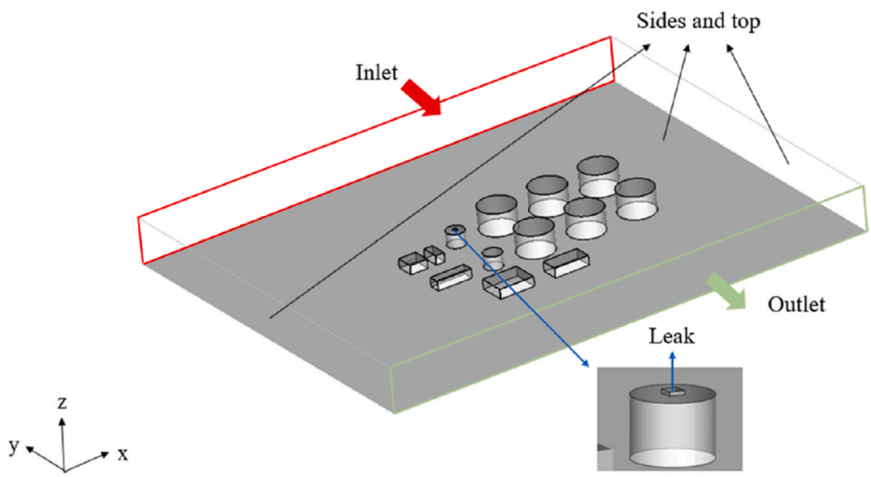

Fig. 2. The layout of the receiving terminal used for gas leak detection tests (with permission from Wu et al., 2021).

2.2. Generation of training data

Prior to building the model, available data on gas leaks were sought for use in network training and testing. It appeared there were not enough resources freely available online that could satisfy the neural network training requirements. Although Convolutional Network can generate additional data for proper training (Jain, 2017), a bare minimum could not be met. Therefore, a validated mathematical simulation of a gas leak using OpenFOAM that had been published recently by (Wu et al., 2021) was adopted for the current purpose This generated sufficient data for network training. The upside of using this established model was that it was ready for immediate use following installation, and some key parameters were easily configured. The downside was that the result generation speed was limited by the virtual machine characteristics and could not be accelerated using a more computationally powerful engine. Another downside was the inability to change the geometry of the simulation since it was built and validated for that specific geometrical setup. Changing the setup could potentially lead to inaccurate results.

The work published by Wu et al. extensively describes the model developed. In short, the three-dimensional model combines computational fluid dynamics (CFD) with the ensemble Kalman filter (EnKF) to simulate leaked LNG vapor propagation in a typical receiving terminal setting. The CFD module simulates the dispersion process through calculation of governing equations and turbulence models while the EnKF module allows assimilation of the data. The results of the model have been compared to the field test (the Burro 8 spill test) conducted by the Lawrence Livermore National Laboratory (LLNL) at the Naval Weapons Center (Koopman et al., 1981). In addition, CFD simulations were compared to an ANSYS FLUENT simulation. These tests have verified that CFD and EnKF coupling reproduced accurate results, and *rhoReactingBuoyantFoam* solver can be used as an alternative to simulate LNG vapor dispersion.

Data generation was performed using a receiving terminal setup with a complex layout, obstacles, buoyancy forces, etc., taken from Wu et al., Fig. 2 shows the layout of the receiving terminal and the location of the leakage (Wu et al., 2021). More details regarding geometrical setup, input parameters, and boundary conditions can be found in their work. The output of this simulation is the concentration profiles of the plant, which is difficult to attain in real life. However, if some leakage data can be extracted, for example, in a controlled experiment, then it would be possible to generate a sufficient dataset for network training. Since leakage and non-leakage have substantial differences, the network should be tuned with little data. In this case, background noise should be taken into account depending on the specifics of the process. If the data is not sufficient, the available data can be used to tune the CFD model, which can then generate leakage profiles for network training and testing.

To train a neural network, it is required to provide data of different scenarios, such as different leakage locations or sizes. Since there was no provision to change the geometry of the simulation, the leakage velocity was set as the variable parameter. For our simulations, five scenarios were chosen: no leak; 10 m/s, 20 m/s, 30 m/s and 50 m/s leakage velocity. Because the leak size can be directly tied to the severity level, these scenarios reflect different states of the terminal: safe state, unsafe state with variable danger level. All scenarios were simulated within the first 50 s of the leak, which would allow for a quick response to the accident.

The results of each simulation contained 51 PNG files where the 1st PNG was the state of the system at $t = 0$, where the leak starts, and the following 50 files show the concentration profile at every second following the 1st leak at an elevation of 20 m from ground level. Fig. 3 shows an example of the concentration profile for 30 m/s scenario at $t = 35$ s; the axes and color description were removed for clarity. For each leakage velocity, the simulations were repeated 25 times with the leakage velocity being altered slightly at each run (for example, simulations for 30 m/s contained simulations from 29 m/s

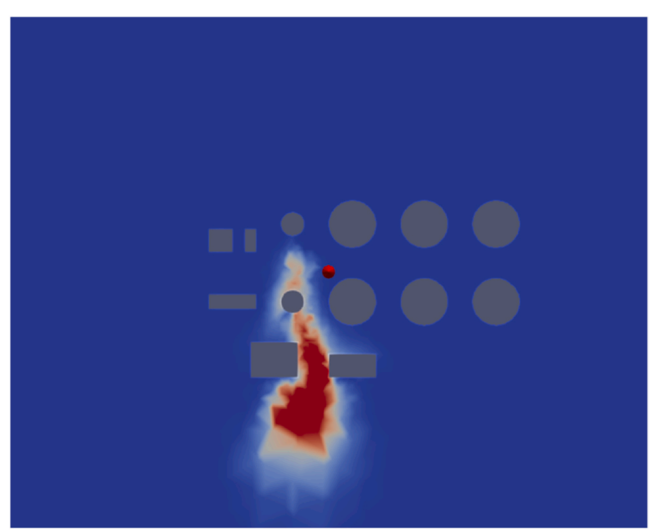

Fig. 3. LNG vapor dispersion at $t = 35$ s with leak velocity of $v = 30$ m/s.

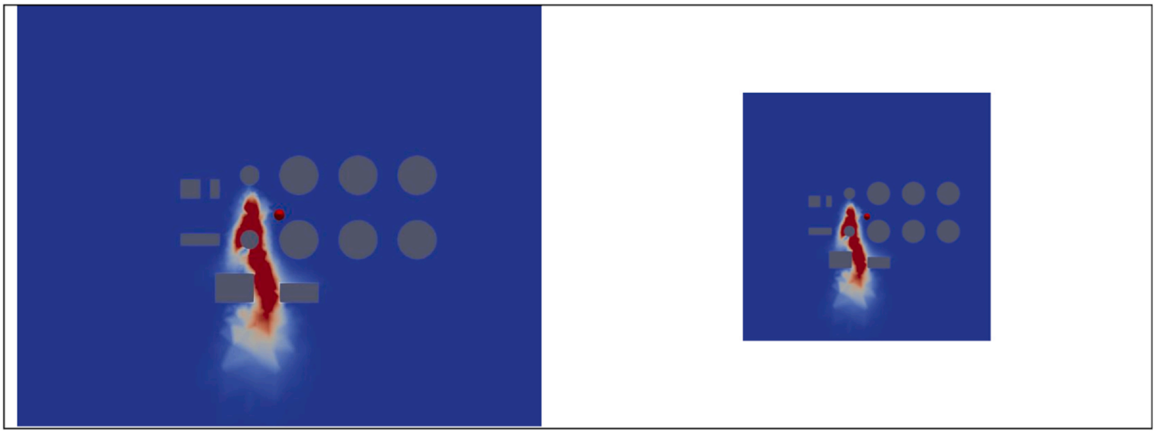

Fig. 4. Initial and resized versions of the frame, $v = 20$ m/s and $t = 35$ s.

to 31 m/s). Of these 25 simulations, 20 were for network training and 5 for network testing. This allowed generalization of the dataset to ensure proper learning at each epoch. In addition, turbulence-related coefficients were changed accordingly for each velocity to imitate the LNG dispersion closely.

2.3. Data Transformation

The generated data consisted of several thousands of images that were 759 by 610 pixels (the default setting for the simulation in OpenFOAM software). This format and amount of data is not convenient to use for network training, as the images are not independent and are sequences of different case scenarios. The data was first transformed into video format to simplify the input process, generating only 100 files for training and 25 for testing.

In order to ensure homogeneity and integrity of the data, quality check was performed following each simulation run to ensure all files were of the same format and dimensions and were not corrupted in the process of transfer from the virtual machine to a local disk. For files that were altered or damaged, the simulation was repeated, and the new results replaced the corrupted ones. Finally, the sets of images were transformed into videos using MATLAB *writeVideo* function at 17 frames per second to produce 3-second clips for each simulation.

Finally, the videoclips were resized to fit the input resolution of the GoogLeNet network using additional function that would transform 759 by 610-pixel frames into 224 by 224-pixel frames. The void space of each image was cropped from each frame to concentrate attention to the important information. An example of the initial and resized frame is shown in Fig. 4. This process was conducted twice: 100 cases for network training and 25 cases for network testing.

2.4. GoogLeNet Feature Extraction

Gas leak detection is an integral part of accident mitigation and escalation prevention. In this paper, a Convolutional Neural Network (ConvNet) was trained to detect gas leaks and leak size. Labeled data was gathered to assist the training process as an example of supervised learning. Gas detection has been treated similar to a classification problem that artificial neural networks have been widely used for. Since our input data consisted of visual content, ConvNet was chosen accordingly, as it is well suited for such input type.

A typical neural network consists of an input layer, output layer, and a hidden layer (one or more). Input data entering the input layer, goes through hidden layer(s), and is then passed on to the output layer. Hidden layers transform data into sets of higher feature vectors through nonlinear operations; in the case of ConvNets, hidden layers perform convolution operations that generate feature maps that become the input of the next layer. In some cases, hidden layers may contain pooling layers, fully connected layers, and normalization layers.

Convolutional networks can be used for classification tasks or just feature extraction; and they can be built from scratch or adapted from existing networks. As such, this work took advantage of the existing GoogLeNet convolutional neural network that is primarily used for image classification developed by researchers at Google (Szegedy et al., 2015). It is a complex network that consists of 27 layers and includes convolution layers, pooling layers, inception modules, dropout layers, output layer and a SoftMax layer. The input layer accepts images of 224 by 224 pixels, while the output layer was initially designed to classify 1000 different images, and hence has an output layer of 1000 cells. In the proposed model, GoogLeNet was used as a feature extraction tool; extracted features are then sent to long short-term memory (LSTM) neural network for classification.

To extract features out of frames in each video, GoogLeNet's last pooling layer "pool5–7 × 7_s1" and *activations* function was used to generate sequences of feature vectors. Each sequence is 51 cells long, corresponding to 51 frames of each video. Each sequence has 1024 features that are used as input to the LSTM network.

2.5. BiLSTM Network Training

The next step is to build an LSTM network to classify gas leakages. The sequences of feature vectors are passed to the Sequence Input Layer with 1024 nodes. The input layer is connected to a bi-directional long short-term memory layer (BiLSTM) with 2000 units and a 50% dropout layer. BiLSTM is the learning mechanism for classifying sequential data that can account for the complete time series at each time step. It is achieved by passing the input data in both forward and backward direction to allow for more information to be readily available for the neural network, which preserves the context of input data. A dropout layer removes a portion of hidden layer units after each repetition during training and prevents the overfitting of the model (Srivastava et al., 2014). This allows generalization of the model and prevents excess reliance on only a few of its input.

A fully connected layer with an output size of 5 cells corresponding to gas leak size is used as an output layer. SoftMax layer and classification layer are used to assign probabilities of each class and subsequently make a decision (Goodfellow et al., 2016). In this work, we classify video footage into 5 categories:

- No leak, 0 m/s

Table 1
Turbulence parameters for data generation for different velocities.

Velocity v, m/s	Turbulence kinetic energy k, J/kg	Specific dissipation rate ω, 1/s
10 m/s	0.082	0.936
20 m/s	0.277	1.717
30 m/s	0.563	2.448
50 m/s	1.378	3.828

- Leak, 10 m/s
- Leak, 20 m/s
- Leak, 30 m/s
- Leak, 50 m/s

Although Wu et al. used leakage velocity of 15 m/s, for this work the leakage velocity was varied to showcase different accident scenarios.

Additional networks settings included: minibatch size of 16; adaptive moment estimation (Adam) solver; learning rate increasing from 0.0001 to 2.0; validation as well as data shuffling after each epoch. Minibatch size affects the number of times the network is updated in an epoch. Minibatch size and learning rate can be adjusted if needed for convergence purposes. Minibatch size also reduces memory requirements since the algorithm must perform calculations only over a fraction of the dataset at a time.

The accuracy of classification is then measured by counting the number of correct predictions and dividing by the total number of cases:

$$Accuracy = \frac{\#(pred == \ true)}{\#true} X \quad 100\%$$

A confusion matrix is then constructed to provide a visual representation of the results. It compares the number of predicted and true samples of all categories in a single figure (Ting, 2017). The network was then trained and tested for different combinations of time durations to see the flexibility of the model in a context of limited data.

3. Results

3.1. Data generation and preparation

As mentioned before, concentration profiles of LNG were generated over 50 s of a gas leak scenario in a receiving terminal. The parameter settings for the simulation provided by Wu et al. were kept mostly unchanged. Thus, ambient temperature, wind velocity, direction, and vessel pressure were not altered. The only variable part was leakage velocity, and turbulence parameters, accordingly. Another choice was regarding elevation selection. Wu et al. documented concentration profiles at three heights: 20 m, 32 m, and 40 m. All elevations were shown to produce results in accordance with test data. Therefore, any of the three could be chosen in this work. The decision to choose 20 m was made to allow inclusion of as many obstacles as possible, as going higher would leave only tall vessels in the dispersion pathway. Other elevations can be considered in future for comparing the results of gas leak detection and possibly combining the data from different heights for further analysis for a relatively complete picture.

Simulations were run 25 times for each leakage velocity, generating a total of 125 runs. The computational capacity of OpenFOAM was limited to the computational capacity of the Virtual Machine used. A standard laptop with Ubuntu 18.04 operating system has been used in this case. With this setting, a single simulation of a gas leak had a runtime of approximately 25–40 min. An additional 3–5 min were required to format and save the results using a separate software (ParaView), according to a template that mitigates noise during training. Running the simulations was the lengthiest part of the project.

As mentioned, each of the five different leak velocities were slightly altered to avoid the same results being produced multiple times. For example, in the case of 30 m/s leak velocity, random velocities in a range from 29.0 to 31.0 were chosen for both training and testing purposes. Only the 0 m/s velocity was not changed (no leak case). The values of turbulence parameters were chosen according to an online calculator. (IChrome, 2016). These values are shown in Table 1. The k-Omega model was utilized instead of other options since it was verified by Wu et al., the reference length was chosen as 8 m, and kinematic viscosity v of the natural gas taken as 1.70e-5 m^2/s.

The data was checked for integrity and saved in folders as sets of images (.png) with 51 images for each simulation. Using MATLAB *writeVideo* function, the dataset of images was transformed into a dataset of videos, producing one video per set (folder). Since there were 51 frames per simulation, a frame rate of 17 was chosen to make 3 second clips. The videos were in the same format as the images (759 by 610 pixels) and were resized to match the GoogLeNet neural network input specifications of 224 by 224 pixels. Alternatively, one could resize the images first and then compile them into videos.

It should also be noted that all videos were 3 ss long, which simplified the data preparation process significantly. Using videos with large variations in time sequence may reduce the classification accuracy significantly. In such cases, an additional step may be necessary to either remove or trim down the longer sequences.

3.2. Extraction of feature vectors and network training

Once the videos were prepared, they were used to extract sequences of feature vectors. The *activations* function of the GoogLeNet returns feature vectors of each frame of the video, which were saved separately. The output of the *activations* produced 1024 by 51 cell arrays for each video, where 51 was the number of frames and 1024 was the number of features.

The sequences of feature vectors were then used as a sequence input layer to train the bidirectional long short-term memory (BiLSTM) network to classify the sequences into gas leak categories. 20% of the training dataset was used for validation. The training procedure achieved 100% validation accuracy by the 6th epoch. The trained network is then used to classify testing samples, and the results are shown in the form of a confusion matrix in Fig. 5.

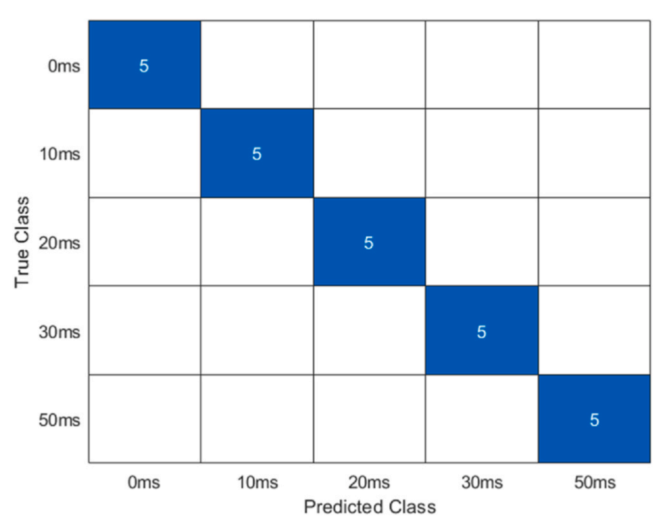

Fig. 5. Confusion matrix for gas leak size classification at t = 50 s.

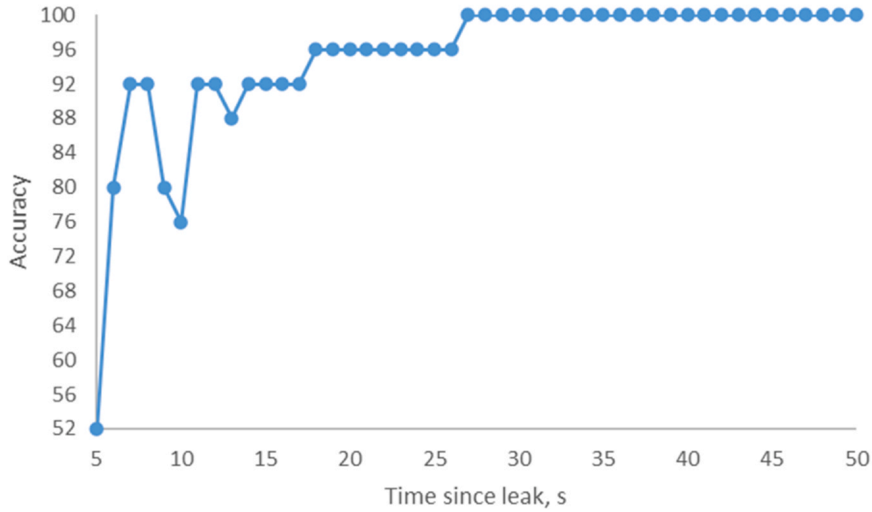

Fig. 6. Accuracy of gas leak classification using the same duration of training and testing data.

The trained network correctly classified all 25 videos of gas leaks into correct categories.

Next, the number of frames presented to the network for training were limited to see how reduced information affected classification accuracy. This was done to measure the prediction accuracy for different times since leak to see how the model could correctly identify leaks. A minimum time of 5 s was set and train-tested the model repeatedly until $t = 50$ s. Classification accuracy over time is shown in Fig. 6. At $t = 27$ s, the accuracy became 100%, and a steady reduction in epochs needed for network training was noted, as expected. However, in this case, the network was trained and tested using the same time, which means that in this situation, it would be enough to gather data of the first 27 s only.

Further investigation was conducted to check whether a model trained on a complete 50-second data could be applied to a limited time testing data. In other words, how well would a complete model be able to predict incomplete data. Fig. 7 shows the accuracies from 10 s to 50 s. The data suggest that applying the incomplete data up till the first 26 s produces low accuracy results; the accuracy only starts to increase at $t = 27$ s. However, the results do not get significantly better until t = 39 s, where classification accuracy reaches 84%, and for t > 39, it finally reaches 100%. Therefore, applying a complete model to an incomplete dataset can be allowed only for situations where at least 40 s have elapsed after the leakage. For small t (less than 30 s), the model struggled to identify leaks at all. With more time available for analysis, the model would identify leakage, but had issues differentiating between 10 and 20 m/s, and 30–50 m/s.

Analyzing confusion plots for the incomplete data scenarios, it was found that for lower times since leak, the model would predict a "no leak" result. The model would always underpredict rather than overpredict, meaning that there were no false positives generated by the network. As time increased, the model started to detect leaks but would often misclassify their magnitudes.

4. Discussion

The results of the initial test run are shown in Fig. 5. The model successfully classified leakage sizes based on provided sequences of concentration profiles over a period of 50 s since leak initiation. This accuracy was achieved partly due to the homogenous nature of the data: it was generated via CFD simulation, which did not account for numerous sources of noise. If real data were used, it is possible the prediction accuracy would decline. However, since the model was proven to be in accordance with testing on a real receiving terminal site, it can be concluded that the model's performance is relevant. It

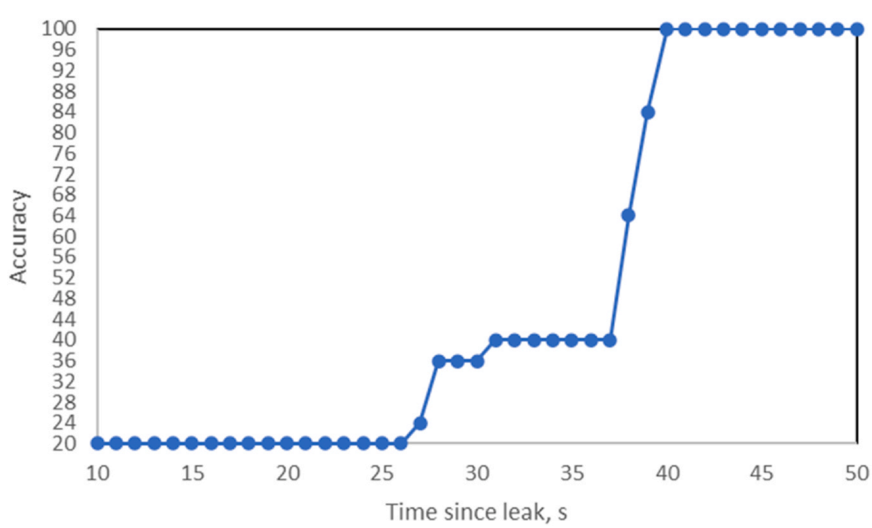

Fig. 7. Accuracy of gas leak classification using a 50 second-trained model on limited duration testing data.

is also important to note that only 100 sets of data were used to train the model, which only accounted for one factor of the gas leak phenomenon – gas leak velocity. It is essential to test the model's flexibility in other factors, such as leak location, vessel pressure, wind velocity and direction, ambient conditions, etc. The model was able to isolate one parameter and make accurate predictions successfully. The leak velocity is proportional to the size of the formed gas cloud, which plays an important role in the extent of potential damage that can arise from dispersion. If the model can identify the severity of the accident, it can result in a more educated response to avoid further escalation and loss. In addition, this model can be incorporated into a risk assessment framework, which would use the leak size to determine the severity of an accident.

Another matter to consider is the type of data used for prediction. We used concentration profiles which are generated via CFD software. While it is a great leak indicator, this type of data may not be readily available at any chemical plant. The model needs to be tested with other types of input, such as infrared thermal imaging, gas sensor data from various locations, and a combination of data types. Since ConvNet is used for feature extraction, it is advised to utilize as much visual data as possible.

Although the model performed with absolute accuracy on a 50-second simulated experiment, it was further tested to determine how quickly it could detect and classify the gas leaks. For this, the amount of data fed to the model was reduced by reducing the time from leak to detection. The data was reduced by one frame at a time and the model was re-trained after each reduction to observe the change in performance. As Fig. 6 shows, the model retained its performance to a point where only 26 s were used. At times less than 26 s, the model was still able to identify leaks but struggled to classify its size correctly. As the available data kept decreasing, the prediction accuracy also decreased, and below 10 s, it produced undesirable results. This indicated that it is possible to accurately detect and classify leaks within a short time interval, allowing for quick response.

Next, the model was analyzed to see if it could correctly predict the leak classification based on a limited amount of data. Fig. 7 shows the change in performance when a model was trained on a full 50 s of data and was forced to make a prediction early. The results suggest that this model does not perform well unless fed at least 40 s of data. In comparison to the previous analysis, it would be more proficient at setting up a gas leakage detection model that is trained only on 30 s available data and let it decide based on full 30 s, rather than setting up a 50-seconds trained model which is only able to predict after 40 s. This suggests that there is a certain amount of optimization that can be done to improve the model.

Aside from model optimization, the model needs to be validated by data acquired from a test site, in addition to the fact that the CFD model used to generate data was proven legitimate. A different geometrical setup should also be tried, as well as different leak locations need to be tested. The ability of the model to predict leak location and its size needs to be examined. Finally, the effect of other parameters such as wind velocity and direction, atmospheric stability etc. should be investigated on the model's performance.

5. Conclusions

This paper presents a neural network model to detect and classify gas leakage based on a series of concentration profiles from temporal visual data generated due to a leak scenario in an LNG terminal. The model consists of feature extraction and classification elements. GoogLeNet pre-trained convolutional network was used to extract features from visual data to convert input video files into sequences of feature vectors; classification was performed using a bidirectional long short-term memory layer neural network. The data is analyzed using a hybrid model that incorporated CFD and the ensemble Kalman filter and was shown to produce results comparable to field tests. The significant advantage of the proposed method is that it does not require fine-tuning of layer configuration and other training parameters to produce satisfying results.

The model had 100% accuracy when trained and tested on a 50-second dataset. The generated input data lacked the presence of noise, and it only examined one parameter of leakage – its velocity. While the leak velocity can indicate the risk level, other factors can also be considered, such as leak location, variation of ambient parameters, gas temperature, and pressure.

The classification accuracy was analyzed further based on a shortened dataset. For the receiving terminal used, as few as 27 s of footage were enough for the model to accurately classify faults. The model's ability to make predictions based on limited information was also analyzed. As such, the model was trained on 50 ss of data testing samples that contained shortened datasets. The results show that the ability to classify leakage was reduced significantly and the model produced acceptable results only when it was fed almost the entire dataset (40 + seconds). Under these circumstances, the model did not produce false negatives and mostly suffered in its ability to classify detected leaks. These findings suggest that optimizations can be done in training the neural network. The model can be tuned to detect and classify gas leaks in a short period with high accuracy. Taking a timely response would stop the escalation and reduce damage.

The model needs further testing through the incorporation of actual data. In addition to testing more gas leakage factors, other types of visual data need to be considered, such as thermal imaging. The incorporation of sensor data and its effect on prediction accuracy should be investigated to improve the model efficiency. Finally, GoogLeNet needs to be compared to other pre-trained convolutional networks to determine the most fitting one for this type of input data.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors thankfully acknowledge the financial support provided by the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Canada Research Chair (Tier I) Program in Offshore Safety and Risk Engineering.

References

Bonvicini, S., Antonioni, G., Morra, P., Cozzani, V., 2015. Quantitative assessment of environmental risk due to accidental spills from onshore pipelines. Process Saf. Environ. Prot. 93, 31–49. https://doi.org/10.1016/j.psep.2014.04.007
Datta, S., Sarkar, S., 2016. A review on different pipeline fault detection methods. J. Loss Prev. Process Ind. 41, 97–106. https://doi.org/10.1016/j.jlp.2016.03.010
Doshmanziari, R., Khaloozadeh, H., Nikoofard, A., 2020. Gas pipeline leakage detection based on sensor fusion under model-based fault detection framework. J. Pet. Sci. Eng. 184, 106581. https://doi.org/10.1016/j.petrol.2019.106581
Eckerman, I., 2005. The Bhopal Saga - Causes and Consequences of the World's Largest Industrial Disaster. Universities Press,, India.
Fahimipirehgalin, M., Trunzer, E., Odenweller, M., Vogel-Heuser, B., 2021. Automatic Visual Leakage Detection and Localization from Pipelines in Chemical Process Plants Using Machine Vision Techniques. Engineering 7 (6), 758–776. https://doi.org/10.1016/j.eng.2020.08.026
Fiates, J., Vianna, S.S.V., 2016. Numerical modelling of gas dispersion using OpenFOAM. Process Saf. Environ. Prot. 104, 277–293. https://doi.org/10.1016/j.psep.2016.09.011
Goodfellow, I., Bengio, Y., Courville, A., 2016. Softmax Units for Multinoulli Output Distributions. In *Deep Learning*. MIT Press, pp. 180–184.http://www.deeplearningbook.org.
IChrome. (2016). *Turbulence Calculator*. ⟨http://ichrome.com/blogs/archives/342⟩.

Jain, S. (2017). Data Augmentation | How to use Deep Learning when you have Limited Data. ⟨https://medium.com/nanonets/nanonets-how-to-use-deep-learning-when-you-have-limited-data-f68c0b512cab⟩.
Koopman, R.P., Kamppinen, L.M., Hogan, W.J., & Lind, C. (1981). Burro series data report: LLNL/NWC 1980 LNG spill tests.
Meribout, M., Khezzar, L., Azzi, A., Ghendour, N., 2020. Leak detection systems in oil and gas fields: Present trends and future prospects. Flow. Meas. Instrum. 75, 101772. https://doi.org/10.1016/j.flowmeasinst.2020.101772
Ning, F., Cheng, Z., Meng, D., Duan, S., Wei, J., 2021. Enhanced spectrum convolutional neural architecture: An intelligent leak detection method for gas pipeline. Process Saf. Environ. Prot. 146, 726–735. https://doi.org/10.1016/j.psep.2020.12.011
Pérez-Pérez, E.J., López-Estrada, F.R., Valencia-Palomo, G., Torres, L., Puig, V., Mina-Antonio, J.D., 2021. Leak diagnosis in pipelines using a combined artificial neural network approach. Control Eng. Pract. 107, 104677. https://doi.org/10.1016/j.conengprac.2020.104677
Song, D., Lee, K., Phark, C., Jung, S., 2021. Spatiotemporal and layout-adaptive prediction of leak gas dispersion by encoding-prediction neural network. Process Saf. Environ. Prot. 151, 365–372. https://doi.org/10.1016/j.psep.2021.05.021
Song, S.J., & Jang, Y.G. (2018, September 21). Construction of digital twin geotechnical resistance model for liquefaction risk evaluation. ACM International Conference Proceeding Series. ⟨https://doi.org/10.1145/3284557.3284739⟩.
Song, Y., Li, S., 2021. Gas leak detection in galvanised steel pipe with internal flow noise using convolutional neural network. Process Saf. Environ. Prot. 146, 736–744. https://doi.org/10.1016/j.psep.2020.11.053
Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15 (1), 1929–1958.
Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Venhoucke, V., & Rabinovich, A. (2015). Going Deeper with Convolutions. CVPT2015. ⟨https://static.googleusercontent.com/media/research.google.com/ru//pubs/archive/43022.pdf⟩.
Ting, K.M., 2017. Confusion Matrix. In: Sammut, C., Webb, G.I. (Eds.), Encyclopedia of Machine Learning and Data Mining. Springer, US, pp. 260. https://doi.org/10.1007/978-1-4899-7687-1_50
Travis, B., Dubey, M., Sauer, J., 2020. Neural networks to locate and quantify fugitive natural gas leaks for a MIR detection system. Atmos. Environ.: X 8, 100092. https://doi.org/10.1016/j.aeaoa.2020.100092
Vollmer, M., Möllmann, K.P., 2017. Infrared thermal imaging: fundamentals, research and applications. John Wiley & Sons,.
Wang, W., Mao, X., Liang, H., Yang, D., Zhang, J., Liu, S., 2021. Experimental research on in-pipe leaks detection of acoustic signature in gas pipelines based on the artificial neural network. Measurement 183, 109875. https://doi.org/10.1016/j.measurement.2021.109875
Wu, J., Cai, J., Yuan, S., Zhang, X., Reniers, G., 2021. CFD and EnKF coupling estimation of LNG leakage and dispersion. Saf. Sci. 139 (June 2020), 105263. https://doi.org/10.1016/j.ssci.2021.105263
Xiao, Q., Li, J., Sun, J., Feng, H., Jin, S., 2018. Natural-gas pipeline leak location using variational mode decomposition analysis and cross-time–frequency spectrum. Measurement 124, 163–172. https://doi.org/10.1016/j.measurement.2018.04.030