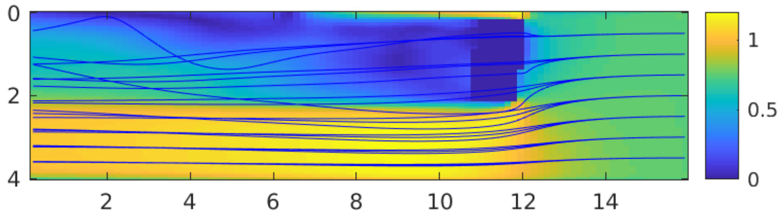
SU 2022-2023, SMT, DISAL-SU37

Start: 10.07.2023
Finish: 01.09.2023

Gas Source Location Classification in Built Environment with a Sensor Network

Mahdi Atallah



Professor: Alcherio Martinoli
Assistant: Wanting Jin

This page intentionally left blank.

Contents

Chapter 1 Introduction

Gas Source Localization (GSL) [1] refers to the process of determining the origin of a gas release within a given environment. It is a concept that plays a crucial role in ensuring environmental safety, industrial security, and effective emergency response strategies. As industries continue to expand, the potential for gas releases, whether accidental or intentional, demands accurate methods for swiftly identifying and locating their sources.

However, sometimes the hazardous nature of certain gas leaks poses significant risks to human safety, making it impractical for individuals to directly engage in locating these leaks. In such scenarios, the application of autonomous techniques, such as localization through stationary sensor networks or mobile robots, offers the advantage of exploring hazardous environments without endangering human lives.

An effective strategy for autonomous GSL involves leveraging a static sensor network to provide an initial, approximate source location. Subsequently, this approximate estimate is further refined by a mobile robot specifically tasked with exploring the identified area. This approach optimizes the efficiency and precision of GSL by striking a balance between the inherent limitations of the sensor network – constrained by the number and placement of sensors – and the gradual nature of GSL undertaken by mobile robots.

We will focus on the former part, utilizing a sensor array to approximate the source's position. In this context, the significance of utilizing time-series gas measurements cannot be overstated. Time-series data, as exemplified in the study by [2], assumes a pivotal role in augmenting the precision and dependability of gas source localization techniques. Furthermore, recent research endeavors have explored the intriguing potential of leveraging time-series measurements not only for gas concentration but also for wind patterns. For instance, [3] and [4] harnessed Long Short-Term Memory (LSTM) machine learning models, capitalizing on time series data for GSL. In [3], the focus lay in discerning the gas source from a list of potential leak locations within an outdoor constructed environment. Conversely, in [4], researchers concentrated on partitioning the space of prospective source locations within an obstacle-free outdoor setting into distinct regions, thereby identifying the precise region harboring the gas source. Our approach mirrors that of [3] and [4], treating this challenge as a classification problem, resulting in the subdivision of the space containing possible sensor locations into multiple distinct regions. The goal is to pinpoint the specific region containing the gas source. In contrast to the specific focus of [3], who concentrated on a single outdoor built environment with a limited number of sensor locations, and to the work by [4], which examined an outdoor obstacle-free environment where the gas source consistently occupied one fixed location within each region, our project takes a different approach. In our study, the gas source can be positioned anywhere on the map, resulting in multiple viable source locations within each region. Additionally, we aim to develop a model capable of predicting the gas source region for any given environment, without being constrained to a specific built setting.

In the upcoming chapters, we will explore our data generation process, detail the methodologies used for source localization, present our findings, and discuss our results. Our work specifically entailed implementing two hybrid architectures, combining CNN and LSTM, with the aim of capturing the spatial and temporal characteristics of the problem. We subsequently evaluated these architectures in comparison to a standalone LSTM network, all conducted within a virtual wind tunnel environment. Our experiments have generated promising results. However, it's important to acknowledge that further research and development are essential to fully establish the viability of this approach.

Chapter 2 Data generation

In this chapter, we delve into the process of data generation, where we elaborate on the various aspects related to the creation and preparation of the dataset used for our study.

2.1 Problem Description

The challenge revolves around the utilization of measurements obtained from an array of sensors as well as information from the environment to accurately classify the source position into specific regions. When dealing with many regions, we also take into account the top-k accuracy, where k is chosen appropriately. Variables such as the quantity and distribution of employed sensors across the space, the number of regions, the method of spatial division for these regions, and the chosen value of k are pivotal factors that can substantially influence the complexity of the problem. Consequently, these variables will be discussed in different sections.

2.2 Experimental Environment

The experimental setup revolves around a virtual environment designed to emulate the wind tunnel facility used in [5]. Through this simulation, we generate maps and execute OpenFOAM simulations at defined wind velocities to capture airflow patterns. Subsequently, leveraging Webots, we simulate a controlled gas emission with a predetermined release rate and sensor placement within the environment. By integrating airflow data into Webots, the software models dispersion phenomena, providing us with sensor readings as the outcome. Detailed insights into this process are discussed further in the next section.

2.3 Data Collection Pipeline

The data generation pipeline we employ builds upon the framework introduced in [5], which is elaborated further in the STE with a Data-Driven Plume Model user manual.

In adhering to the original pipeline, we produced 200 training maps, the maximum yield of the pipeline, each accompanied by its corresponding airflow characteristics. Additionally, we acquired airflow data for the 8 test maps stipulated by the pipeline. Wind speed of 1.0 m/s and a release rate of 53 units were maintained. The sensor and source ranges remained consistent, with the x-coordinate spanning 2.75m to 12.7418m and the y-coordinate from 0.01m to 3.9916m within the 20m by 4m wind tunnel. For map-specific data, we continued to use the existing pipeline to leverage flow definitions and signed distance functions.

Our modifications to the pipeline aimed to tailor it to our project's requirements:

- **Obstacle-Free Map Generation:** We initiated by generating an obstacle-free map. This allowed us to assess our model's performance in a simplified setting.

- **Time-Series Data Enhancement:** Recognizing the significance of the temporal dynamics in gas dispersion, we extended the pipeline to yield time-series data. Unlike the previous approach, which recorded only the mean and standard deviation of 150 measurements per feature, we retained the entirety of these 150 measurements. This enhancement proved vital in capturing the intricate temporal characteristics of the gas dispersion phenomenon.

- **Diverse Sensor Configurations:** We extended our data collection efforts to encompass information from 216 sensors arranged in an 18 by 12 configuration (refer to Figure **??**). While not all sensor data is utilized for our primary task, this configuration offers the flexibility to experiment with various sensor arrangements without necessitating a complete dataset regeneration.

  The sensor layout's adaptability is enabled through adjustments to the `static_sensor_net_grid_controller.cpp` source code. We've slightly modified the code, making configuration changes as simple as adjusting the `XNUM` and `YNUM` constants at the file's outset. These constants respectively dictate the number of sensors along the x and y axes in the grid. Additionally, this file controls the sensor data collected, offering customization options.

- **Source Position Flexibility:** Our `C++` code was adapted to confer flexibility to the source's position on the map. This departure from the exclusive upwind positioning allows our model to predict source locations regardless of wind direction. Future analyses can easily focus on solely upwind or downwind source positions without requiring dataset regeneration.

- **Expanded Data Generation:** In contrast to the approach in [5], we generated a more extensive dataset. For each of the 200 training maps, we generated data for 200 distinct source positions, offering richer training samples for our proposed models.

  This expansion was driven by the complexity of our envisioned models. Additionally, the larger dataset enabled the exploration of model behavior on specific maps, a strategy that can expose potential weaknesses in certain conditions. However, it's important to note that while we conceived this strategy, we didn't explore it in this project.

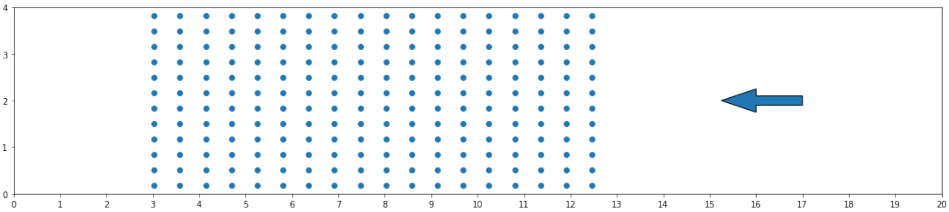


Figure 2.1: Placement of 216 sensors on an unoccupied map.

2.4 Data Analysis and Transformation

With the dataset in hand, we conducted a thorough analysis to discover potential insights and subsequently applied transformations to enhance its utility.

2.4.1 Dataset Balance Assessment

A pivotal consideration is determining whether our dataset maintains balance. As elucidated in the preceding section, the source's random positioning adheres to an area where the x-coordinate ranges from 2.75m to 12.75m, and the y-coordinate spans from 0.01m to 3.9916m. Consequently, when employing regions of equal area, the resultant labels exhibit a uniform distribution.

In this project, we partition the space into a grid of cells, each of which possesses equal area. The configuration file of the project governs the number of cells along the x and y axes. Notably, each region is assigned a unique numeric identifier, with 0 designated for the bottom-left cell, followed by sequential numbering extending to the right and upwards. Given our deliberate adherence to equal area regions, no additional intervention is required in this regard. It is worth noting, however, that introducing regions with disparate areas would necessitate the application of techniques such as resampling methods or cost-sensitive learning to mitigate class imbalance.

2.4.2 Sensor Data Analysis and Preprocessing

Our sensor data encompasses gas concentration, wind speed, and wind angle measurements. We employed min-max normalization to ensure uniform scaling across all data points, transforming the values to the interval [0, 1].

Furthermore, we implemented a filtering mechanism to select specific sensors based on our desired configuration. This selection process is guided by the configuration settings we aim to achieve.

2.4.3 Map Data

For each map, we obtained the flow definition and signed distance function, as described in [5]. The flow definition entailed extracting obstacle information, represented as a binary value denoting whether a specific pixel resided within an obstacle. To maintain consistency with the normalization applied to the sensor data, we subjected the map data to min-max normalization. This normalization aligns the map data's scale with that of the sensor data, ensuring uniformity across the dataset.

To provide visual insights into our dataset, representations of the maps used in the training set and the testing set are displayed in the appendix (see Figures 5.1 and 5.2).

Chapter 3 Methods

In this chapter, we present the methodologies employed in our study. Specifically, we delve into the utilization of LSTM networks and combined architectures that integrate LSTM and CNN components to enhance our gas source localization approach.

3.1 LSTM Network

In this section, we provide an in-depth exploration of the LSTM network methodology incorporated in our project. The LSTM network is designed to exploit the temporal characteristics of the data, enabling accurate predictions. Detailed insights into the LSTM architecture can be found in [4] and [3].

The input to the LSTM network consists of sensor data, encompassing essential measurements such as gas concentration, wind speed, and wind angle. It's noteworthy that map data, due to its lack of temporal dynamics, is intentionally excluded from the input to the LSTM network.

We chose to further preprocess the data. By aggregating 15 consecutive measurements, we construct a single time frame. The mean value of each group of measurements is then computed, yielding 10 time frames per sample.

The architecture of our LSTM network is visually represented in Figure 3.1:

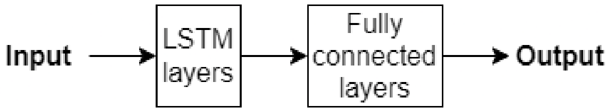


Figure 3.1: LSTM Network Architecture

For training our LSTM network, we employ the Adam optimizer and the cross-entropy loss function. These choices provide a suitable framework for optimizing the network's performance and guiding its learning process.

3.2 Hybrid Network

In this section, we extend our methodology to encompass hybrid network configurations, aiming to exploit the advantages of LSTM and CNN components. Our exploration involves two distinct strategies for effectively combining these inputs.

3.2.1 First Hybrid Network

Parallel to the LSTM network, we undertake similar prepossessing steps for sensor data. We execute this stage prior to combining it with the map data.

In this configuration, the input is represented as a 64×64 image with 3 channels:

1. The first channel represents obstacle information in binary form.
2. The second channel corresponds to the signed distance function.

Both of these channels undergo processing through CNN layers to extract spatial features. Subsequently, the last channel contains sensor data, as exemplified in Figure 3.3.

Figure 3.2 provides a visual representation of our network architecture.

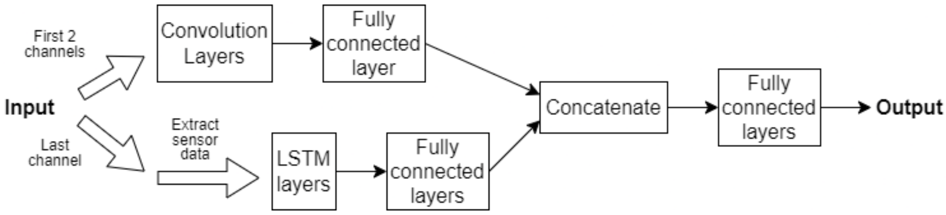


Figure 3.2: First Hybrid Network Architecture

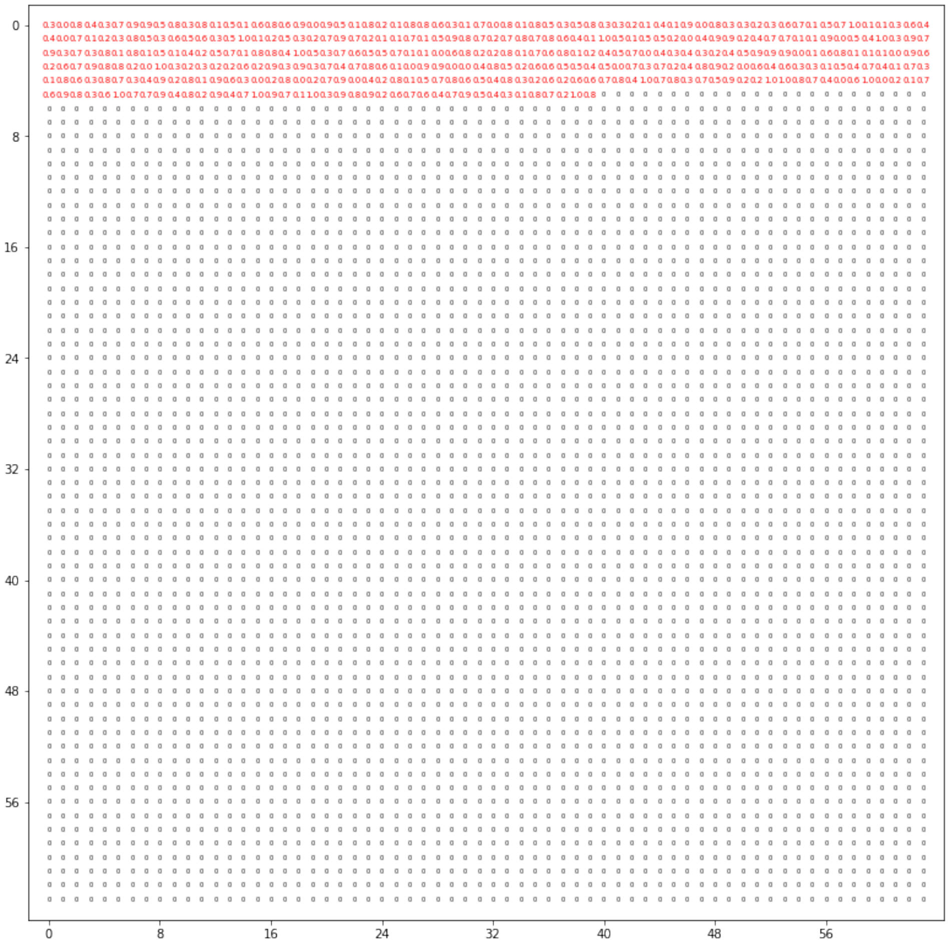


Figure 3.3: Example of the last channel in the input

Similar to the LSTM network, we employ the Adam optimizer and the cross-entropy loss function for training purposes. Our model undergoes training for 100 epochs with a learning rate set at 0.001.

3.2.2 Second Hybrid Network

In keeping with the preprocessing methodology employed in previously explored network configurations, we apply the same preprocessing steps to our data in the second hybrid network. As in the first hybrid network, this preprocessing occurs before the fusion of the two inputs.

We introduce an alternative approach for incorporating both sensor data and map data into our network architecture. The integration process is outlined as follows:

1. The first channel contains binary-encoded obstacle information.

2. The second channel represents the signed distance function.
3. Each subsequent channel corresponds to sensor array measurements for a specific feature at a particular time frame. The image mirrors the map, with pixels aligning with sensor locations reflecting the corresponding sensor measurement for the given feature and time frame. In cases of missing sensor data, a pixel value of 0 is assigned. With three features (gas concentration, wind speed, and wind angle) and ten time frames, we introduce an additional 30 channels to the representation. The first three channels correspond to the features of the initial time step, followed by the next three channels for the subsequent time step, and so on.

All these channels pass through convolution layers, leading to an image with multiple channels that is later flattened into time-series data. Notably, in our implementation, the input is structured as a 32-channel image, with the first two channels containing map information, and the remaining 30 channels representing data for ten time frames and three features. The output also assumes the form of a 30-channel image, which we interpret as data for ten time frames and three features.

In Figure 3.5, we illustrate an example channel of our input data used for sensor data alongside the corresponding sensor placements shown in Figure 3.4. This presentation aims to provide insight into the nature of the input data.

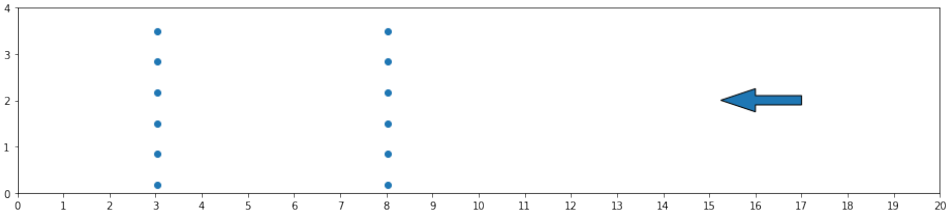


Figure 3.4: Visualization of Input Data and Sensor Placement

Furthermore, we present a visual depiction of the architecture employed in figure 3.6:

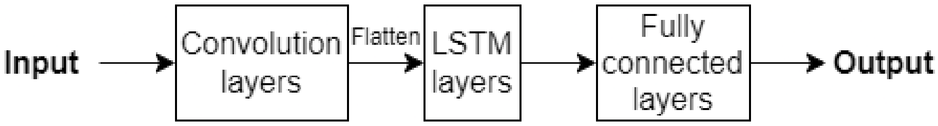


Figure 3.6: Second Hybrid Network Architecture

Consistent with the overarching approach of our network configurations,

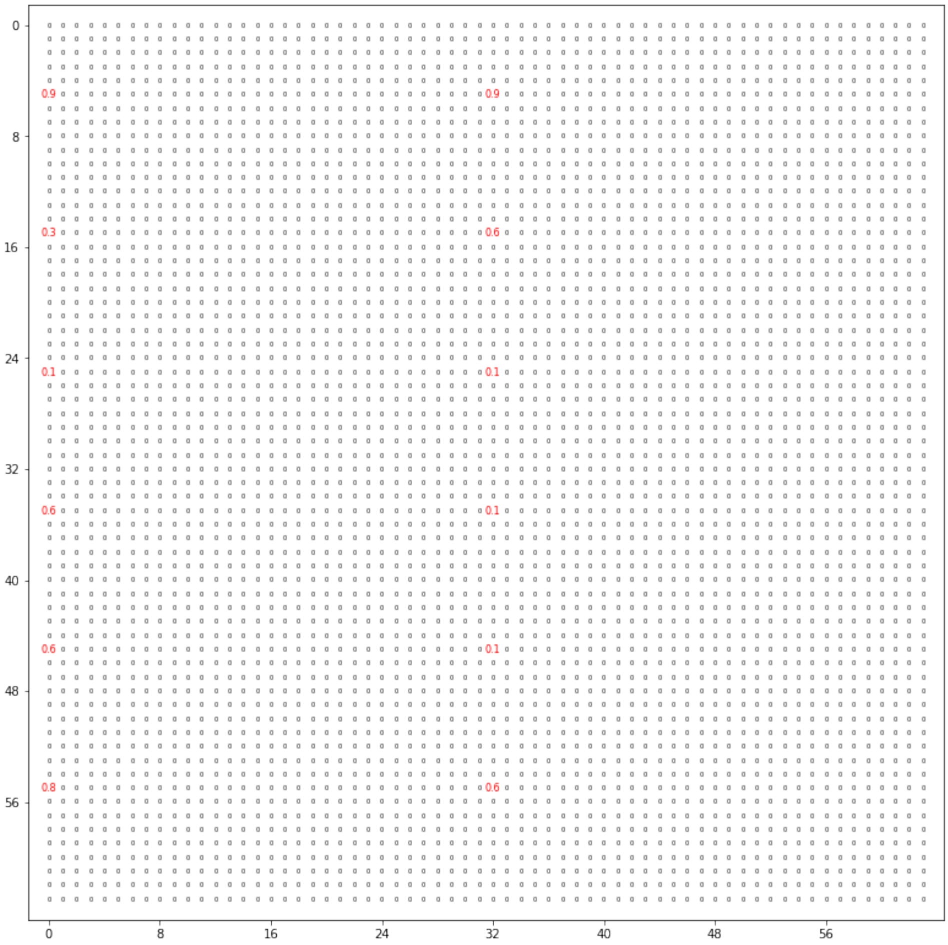


Figure 3.5: Example of one of the last 30 channels in the input

we continue to employ the Adam optimizer and the cross-entropy loss function for the training of the this hybrid network.

Chapter 4 Results and Discussion

Since, as discussed in Section 2.4.1, sensor placement and the region partition have a significant impact on the difficulty of the problem, we will compare the results of the methods we used for different configurations. We will use accuracy, precision, recall, as well as top-2 accuracy to evaluate our models' performance. Additionally, we will provide the confusion matrices in the appendix.

We chose to examine two configurations: a 2 × 6 sensor placement as well as a 2 × 3 sensor placement with a region partition of 2 × 6 and 2 × 3 respectively (see Figure 4.1).

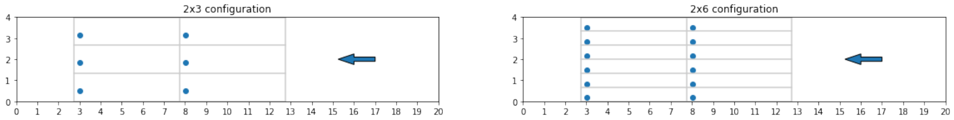


Figure 4.1: Sensor Configurations

4.1 Obstacle-free Environment

Tables 4.1 and 4.2 provide a comprehensive analysis of model performance within an obstacle-free environment.

We observe that both the 2 x 3 and 2 x 6 configurations exhibit similar performance characteristics for both the LSTM network and the first hybrid network. This similarity arises from the fact that in an environment devoid of obstacles, these networks essentially receive equivalent information. Notably, the second hybrid network demonstrates slightly superior performance, possibly attributed to its ability to leverage sensor position information within the map.

It is important to highlight that in the 2x6 configuration, accuracy is notably higher. This observation implies that despite doubling the number of regions, the doubling of sensor density along the y-axis significantly contributes to the model's enhanced predictive capabilities.

One noteworthy insight obtained from our experimentation in an obstacle-free setting pertains to sensor positioning. We discovered that achieving adequate sensor density along the y-axis is paramount for optimal performance. Additionally, positioning sensors at the extremities of a region correlates with improved predictive accuracy. This observation aligns logically with the idea that if the initial row of sensors fails to detect any signals, it strongly suggests that the source is not located upwind of that row.

		Accuracy	Precision	Recall	Top-2 Accuracy
LSTM	Train set	83.94%	86.77%	83.81%	91.81%
	Test set	84.30%	87.03%	84.14%	91.92%
First hybrid network	Train set	83.38%	87.14%	83.25%	89.66%
	Test set	82.95%	86.80%	82.82%	89.55%
Second hybrid network	Train set	90.00%	90.65%	89.93%	97.97%
	Test set	90.05%	90.58%	90.06%	97.88%

Table 4.1: Performance Comparison for the 2 × 3 Sensor Configuration in an Obstacle-Free Environment

		Accuracy	Precision	Recall	Top-2 Accuracy
LSTM	Train set	93.38%	93.73%	93.30%	99.97%
	Test set	93.45%	93.81%	93.40%	99.98%
First hybrid network	Train set	93.28%	93.60%	93.14%	99.97%
	Test set	93.17%	93.48%	93.05%	99.95%
Second hybrid network	Train set	96.59%	96.68%	96.59%	99.97%
	Test set	96.50%	96.60%	96.49%	99.98%

Table 4.2: Performance Comparison for the 2 × 6 Sensor Configuration in an Obstacle-Free Environment

4.2 Built Environment

Tables 4.3 and 4.4 showcase the performance of various models in a built environment.

The comparable performance levels observed in both the LSTM network and the first hybrid network suggest that the strategy of processing map data and sensor data separately, followed by their integration for further processing, may not be an optimal approach. Notably, the first hybrid network, in contrast to the second hybrid network, does not leverage sensor placement information within the map, potentially contributing to this performance similarity.

Intriguingly, the second hybrid network in the 2x3 configuration outperforms other models in both the training and test datasets, possibly attributable to its unique architectural design. However, the second hybrid network in the 2x6 configuration exhibits significantly improved performance in the training dataset but struggles to replicate this advantage in the test dataset. This discrepancy highlights potential limitations in the generalizability of the specific architecture employed to previously unseen maps.

Moreover, a significant drawback of the second hybrid network lies in its inefficient utilization of space, employing a 64x64 image while occupying only a small portion of the available pixels. This inefficiency results in high RAM usage, forcing the model to be trained on a CPU, thereby slowing down the training process. Additionally, it complicates distinguishing between pixels representing sensor measurements of 0 and pixels corresponding to the absence of sensors, further challenging the model's understanding.

Across all methods explored, a common limitation stems from the fixed positions of sensors for each obstacle map. This limitation can lead to instances where sensors are positioned within obstacles, resulting in the loss of valuable information that could have been captured with different sensor placements. In a broader context, relying on a uniform sensor placement strategy for various environments may result in suboptimal sensor arrangements, contrary to the approach presented in [3], where optimal sensor placements were specifically tailored to each given map.

		Accuracy	Precision	Recall	Top-2 Accuracy
LSTM	Train set	75.45%	77.08%	75.43%	89.91%
	Test set	51.56%	52.80%	51.70%	70.62%
First hybrid network	Train set	74.09%	75.48%	73.99%	90.71%
	Test set	52.19%	55.76%	52.65%	72.81%
Second hybrid network	Train set	84.82%	84.89%	84.79%	95.80%
	Test set	64.38%	64.24%	64.53%	85.62%

Table 4.3: Performance Comparison for the 2×3 Sensor Configuration in a Built Environment

		Accuracy	Precision	Recall	Top-2 Accuracy
LSTM	Train set	69.67%	70.25%	75.43%	87.67%
	Test set	53.44%	55.05%	53.76%	74.69%
First hybrid network	Train set	72.01%	72.59%	71.79%	89.74%
	Test set	54.06%	55.77%	54.67%	75.00%
Second hybrid network	Train set	91.07%	91.16%	91.02%	97.25%
	Test set	55.00%	55.57%	55.04%	72.50%

Table 4.4: Performance Comparison for the 2×6 Sensor Configuration in a Built Environment

Chapter 5 Conclusion

In our study, we concentrated on Gas Source Localization (GSL) using static sensors, where a sensor array approximates the source's position. We acknowledged the critical role of time-series gas measurements, as seen in the utilization of Long Short-Term Memory (LSTM) machine learning models for GSL, which enhances precision and dependability.

Our approach, akin to strategies found in [3] and [4], treated the problem as a classification challenge, effectively dividing the space into multiple regions to identify the specific one containing the gas source. However, our work introduced significant distinctions. We permitted gas source positioning anywhere on the map, resulting in multiple viable source locations within each region. Moreover, our model aimed for versatility, capable of predicting gas source regions across various environments, unrestricted by specific built settings.

Our experimental journey involved the implementation of two hybrid architectures, combining Convolutional Neural Networks (CNN) and LSTM, evaluated against a standalone LSTM network. These experiments, carried out within a virtual wind tunnel environment, delivered encouraging results.

While our outcomes may not have been groundbreaking, they establish a valuable foundation. We've developed infrastructure and software that can serve as a robust starting point for future experiments and enhancements. The intricacies of GSL should inspire further exploration and technique refinement as we strive for more effective and reliable solutions in the quest for safer and more secure environments.

Bibliography

[1] A. Francis, S. Li, C. Griffiths, and J. Sienz, "Gas source localization and mapping with mobile robots: A review," *Journal of Field Robotics*, 2022.

[2] M. Schmuker, V. Bahr, and R. Huerta, "Exploiting plume structure to decode gas source distance using metal-oxide gas sensors," *Sensors and Actuators B: Chemical*, 2016.

[3] H. Kim, M. Park, C. W. Kim, and D. Shin, "Source localization for hazardous material release in an outdoor chemical plant via a combination of lstm-rnn and cfd simulation," *Computers Chemical Engineering*, 2019.

[4] C. Bilgera, A. Yamamoto, M. Sawano, H. Matsukura, and H. Ishida, "Application of convolutional long short-term memory neural networks to signals collected from a sensor network for autonomous gas source localization in outdoor environments," 2018.

[5] W. Jin, F. Rahbar, C. Ercolani, and A. Martinoli, "Towards efficient gas leak detection in built environments: Data-driven plume modeling for gas sensing robots," 2023.

Appendix

A Maps

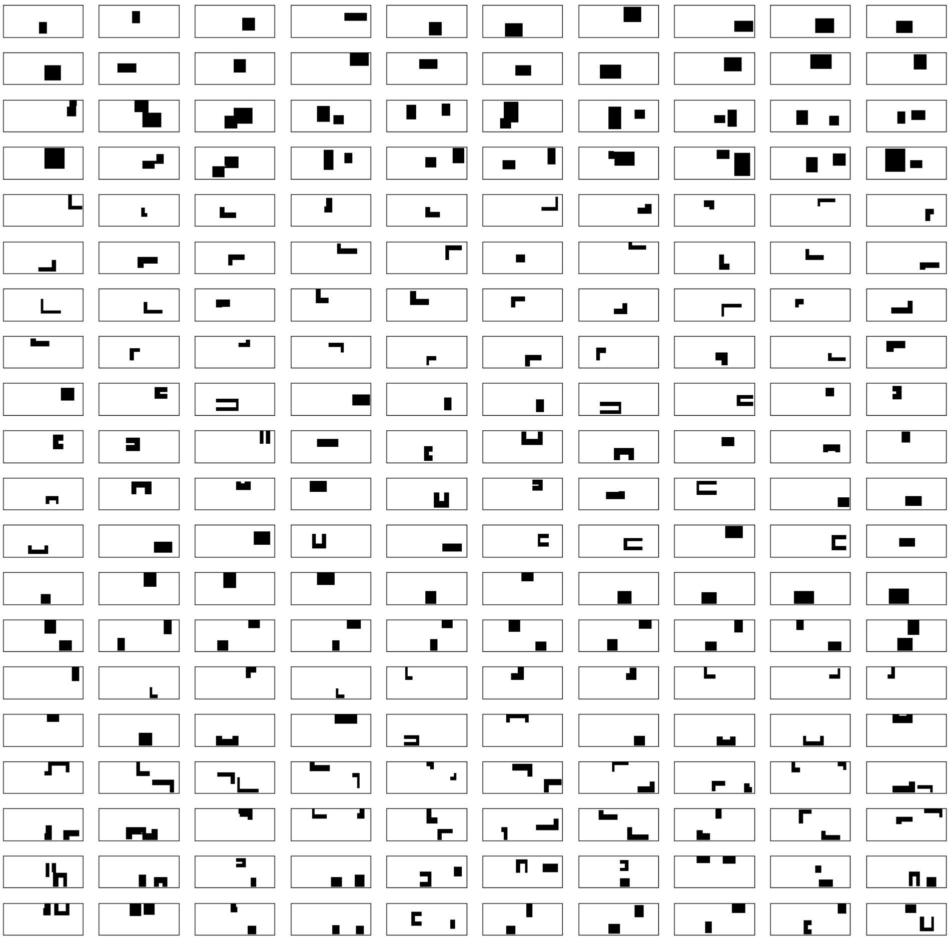


Figure 5.1: Maps in the training set.

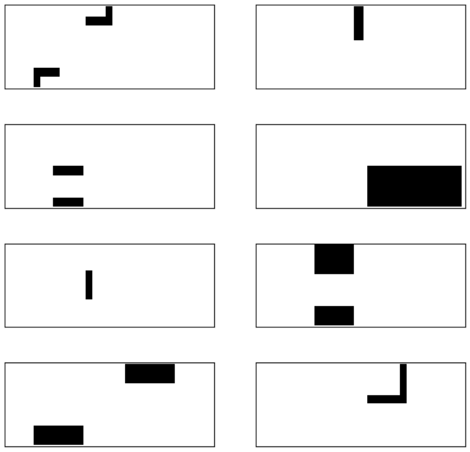


Figure 5.2: Maps in the testing set.

B Confusion matrices

B.1 Obstacle-free environment

LSTM

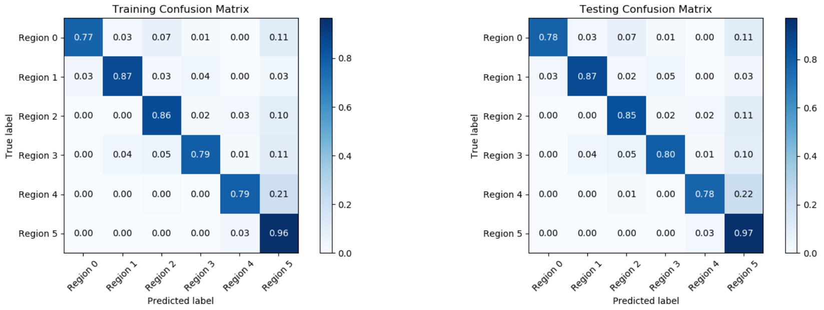


Figure 5.3: Confusion matrix of the LSTM network in a 2 × 3 configuration in an obstacle-free environment.

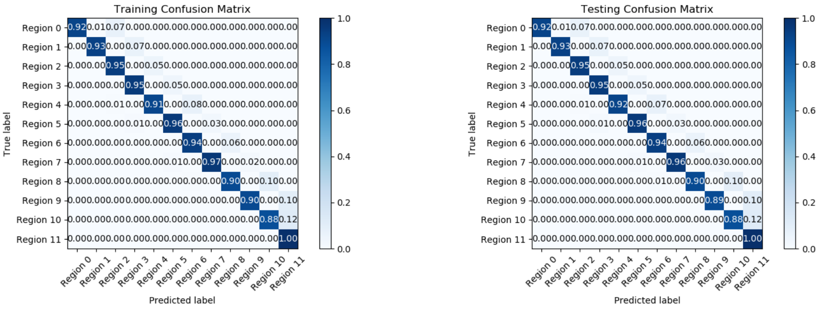


Figure 5.4: Confusion matrix of the LSTM network for the 2 × 6 configuration in an obstacle-free environment.

First hybrid network

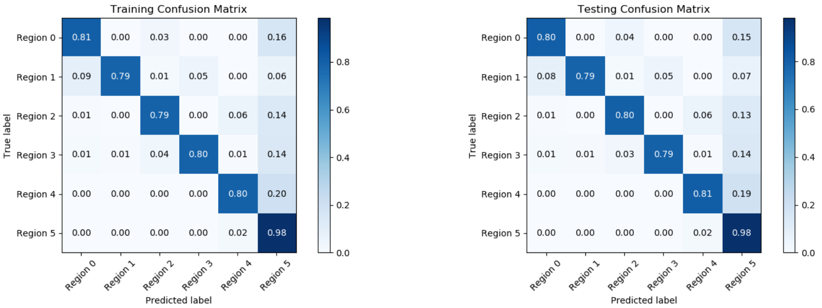


Figure 5.5: Confusion matrix of the first hybrid network in a 2 × 3 configuration in an obstacle-free environment.

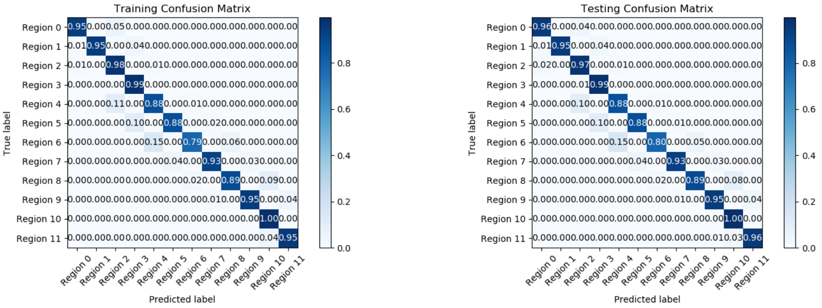


Figure 5.6: Confusion matrix of the first hybrid network for the 2 × 6 configuration in an obstacle-free environment.

Second hybrid network

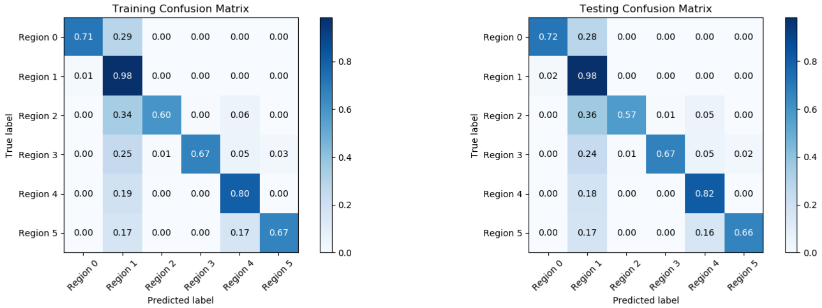


Figure 5.7: Confusion matrix of the second hybrid network in a 2 × 3 configuration in an obstacle-free environment.

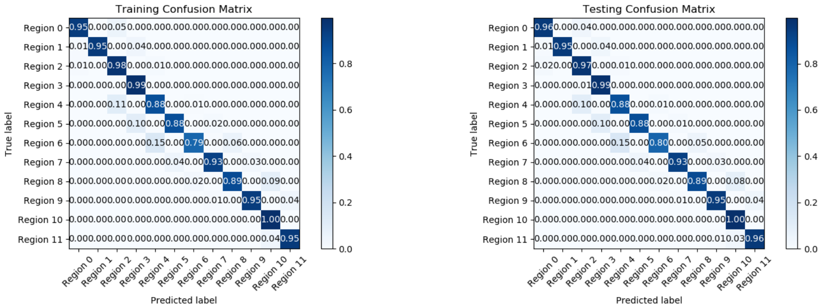


Figure 5.8: Confusion matrix of the second hybrid network for the 2 × 6 configuration in an obstacle-free environment.

B.2 Built environment

LSTM

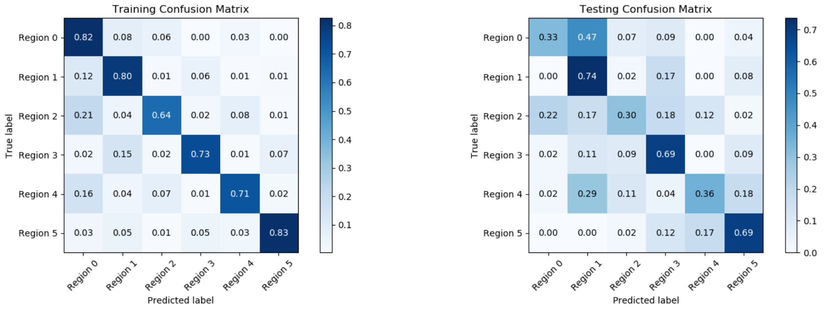
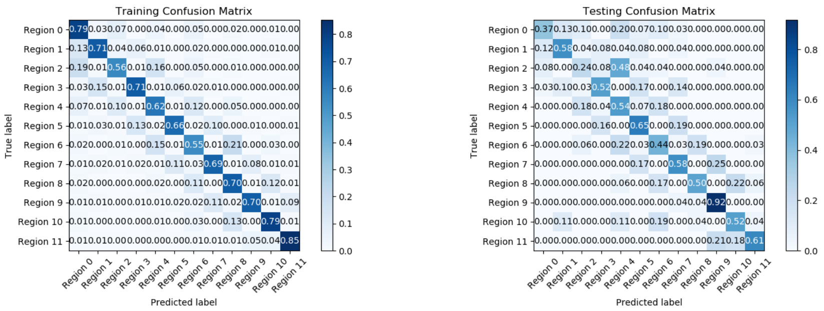


Figure 5.9: Confusion matrix of the LSTM network in a 2 × 3 configuration in a built environment.



Training Confusion Matrix

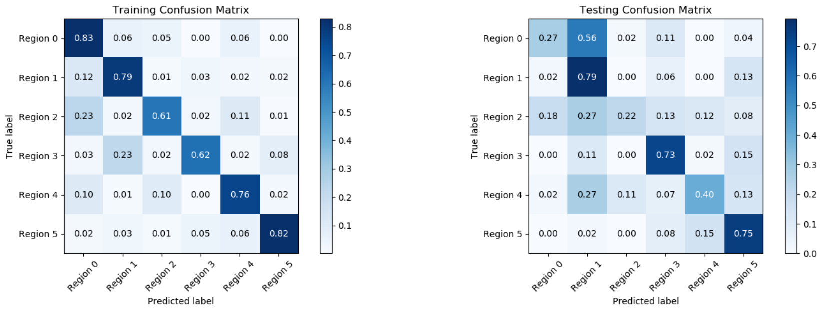
	Region 0	Region 1	Region 2	Region 3	Region 4	Region 5	Region 6	Region 7	Region 8	Region 9	Region 10	Region 11
Region 0	0.79	0.03	0.07	0.00	0.04	0.00	0.05	0.00	0.02	0.00	0.01	0.00
Region 1	0.13	0.71	0.04	0.06	0.01	0.00	0.02	0.00	0.00	0.00	0.01	0.00
Region 2	0.19	0.01	0.56	0.01	0.16	0.00	0.05	0.00	0.01	0.00	0.00	0.00
Region 3	0.03	0.15	0.01	0.71	0.01	0.06	0.02	0.01	0.00	0.00	0.00	0.00
Region 4	0.07	0.01	0.10	0.01	0.62	0.01	0.12	0.00	0.05	0.00	0.00	0.00
Region 5	0.01	0.03	0.01	0.13	0.02	0.66	0.02	0.10	0.00	0.01	0.00	0.01
Region 6	0.02	0.00	0.01	0.00	0.15	0.01	0.55	0.01	0.21	0.00	0.03	0.00
Region 7	0.01	0.02	0.01	0.02	0.01	0.11	0.03	0.69	0.01	0.08	0.01	0.01
Region 8	0.02	0.00	0.00	0.00	0.02	0.00	0.11	0.00	0.70	0.01	0.12	0.01
Region 9	0.01	0.01	0.00	0.01	0.01	0.02	0.02	0.11	0.02	0.70	0.01	0.09
Region 10	0.01	0.00	0.00	0.00	0.01	0.00	0.03	0.00	0.13	0.00	0.79	0.01
Region 11	0.01	0.01	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.05	0.04	0.85

Testing Confusion Matrix

	Region 0	Region 1	Region 2	Region 3	Region 4	Region 5	Region 6	Region 7	Region 8	Region 9	Region 10	Region 11
Region 0	0.37	0.13	0.10	0.00	0.20	0.07	0.10	0.03	0.00	0.00	0.00	0.00
Region 1	0.12	0.58	0.04	0.08	0.04	0.08	0.00	0.04	0.00	0.00	0.00	0.00
Region 2	0.08	0.00	0.24	0.08	0.48	0.04	0.04	0.00	0.00	0.04	0.00	0.00
Region 3	0.03	0.10	0.03	0.52	0.00	0.17	0.00	0.14	0.00	0.00	0.00	0.00
Region 4	0.00	0.00	0.18	0.04	0.54	0.07	0.18	0.00	0.00	0.00	0.00	0.00
Region 5	0.00	0.00	0.00	0.16	0.00	0.65	0.00	0.19	0.00	0.00	0.00	0.00
Region 6	0.00	0.00	0.06	0.00	0.22	0.03	0.44	0.03	0.19	0.00	0.00	0.03
Region 7	0.00	0.00	0.00	0.00	0.00	0.17	0.00	0.58	0.00	0.25	0.00	0.00
Region 8	0.00	0.00	0.00	0.00	0.06	0.00	0.17	0.00	0.50	0.00	0.22	0.06
Region 9	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.04	0.92	0.00	0.00
Region 10	0.00	0.11	0.00	0.00	0.11	0.00	0.19	0.00	0.04	0.00	0.52	0.04
Region 11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.21	0.18	0.61

Figure 5.10: Confusion matrix of the LSTM network for the 2 × 6 configuration in built environment.

First hybrid network



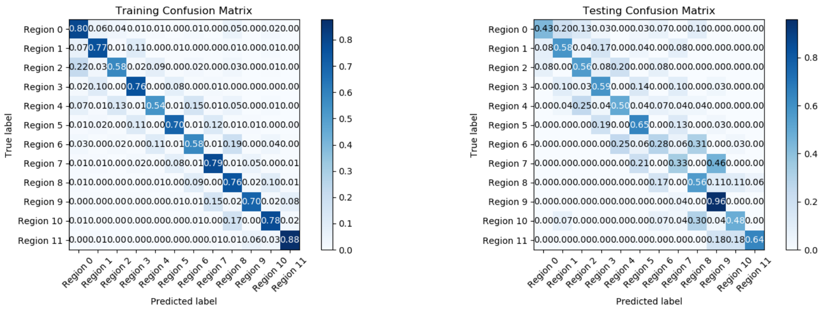
Training Confusion Matrix

	Region 0	Region 1	Region 2	Region 3	Region 4	Region 5
Region 0	0.83	0.06	0.05	0.00	0.06	0.00
Region 1	0.12	0.79	0.01	0.03	0.02	0.02
Region 2	0.23	0.02	0.61	0.02	0.11	0.01
Region 3	0.03	0.23	0.02	0.62	0.02	0.08
Region 4	0.10	0.01	0.10	0.00	0.76	0.02
Region 5	0.02	0.03	0.01	0.05	0.06	0.82

Testing Confusion Matrix

	Region 0	Region 1	Region 2	Region 3	Region 4	Region 5
Region 0	0.27	0.56	0.02	0.11	0.00	0.04
Region 1	0.02	0.79	0.00	0.06	0.00	0.13
Region 2	0.18	0.27	0.22	0.13	0.12	0.08
Region 3	0.00	0.11	0.00	0.73	0.02	0.15
Region 4	0.02	0.27	0.11	0.07	0.40	0.13
Region 5	0.00	0.02	0.00	0.08	0.15	0.75

Figure 5.11: Confusion matrix of the first hybrid network in a 2 × 3 configuration in a built environment.



Training Confusion Matrix

	Region 0	Region 1	Region 2	Region 3	Region 4	Region 5	Region 6	Region 7	Region 8	Region 9	Region 10	Region 11
Region 0	0.80	0.06	0.04	0.01	0.01	0.00	0.01	0.00	0.05	0.00	0.02	0.00
Region 1	0.07	0.77	0.01	0.11	0.00	0.01	0.00	0.00	0.01	0.00	0.01	0.00
Region 2	0.22	0.03	0.58	0.02	0.09	0.00	0.02	0.00	0.03	0.00	0.01	0.00
Region 3	0.02	0.10	0.00	0.76	0.00	0.08	0.00	0.01	0.00	0.00	0.00	0.00
Region 4	0.07	0.01	0.13	0.01	0.54	0.01	0.15	0.01	0.05	0.00	0.01	0.00
Region 5	0.01	0.02	0.00	0.11	0.00	0.70	0.01	0.12	0.01	0.01	0.00	0.00
Region 6	0.03	0.00	0.02	0.00	0.11	0.01	0.58	0.01	0.19	0.00	0.04	0.00
Region 7	0.01	0.01	0.00	0.02	0.00	0.08	0.01	0.79	0.01	0.05	0.00	0.01
Region 8	0.01	0.00	0.00	0.00	0.01	0.00	0.09	0.00	0.76	0.02	0.10	0.01
Region 9	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.15	0.02	0.70	0.02	0.08
Region 10	0.01	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.17	0.00	0.78	0.02
Region 11	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.06	0.03	0.88

Testing Confusion Matrix

	Region 0	Region 1	Region 2	Region 3	Region 4	Region 5	Region 6	Region 7	Region 8	Region 9	Region 10	Region 11
Region 0	0.43	0.20	0.13	0.03	0.00	0.03	0.07	0.00	0.10	0.00	0.00	0.00
Region 1	0.08	0.58	0.04	0.17	0.00	0.04	0.00	0.08	0.00	0.00	0.00	0.00
Region 2	0.08	0.00	0.56	0.08	0.20	0.00	0.08	0.00	0.00	0.00	0.00	0.00
Region 3	0.00	0.10	0.03	0.59	0.00	0.14	0.00	0.10	0.00	0.03	0.00	0.00
Region 4	0.00	0.04	0.25	0.04	0.50	0.04	0.07	0.04	0.04	0.00	0.00	0.00
Region 5	0.00	0.00	0.00	0.19	0.00	0.65	0.00	0.13	0.00	0.03	0.00	0.00
Region 6	0.00	0.00	0.00	0.00	0.25	0.06	0.28	0.06	0.31	0.00	0.03	0.00
Region 7	0.00	0.00	0.00	0.00	0.00	0.21	0.00	0.33	0.00	0.46	0.00	0.00
Region 8	0.00	0.00	0.00	0.00	0.00	0.00	0.17	0.00	0.56	0.11	0.11	0.06
Region 9	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.96	0.00	0.00
Region 10	0.00	0.07	0.00	0.00	0.00	0.00	0.07	0.04	0.30	0.04	0.48	0.00
Region 11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.18	0.18	0.64

Figure 5.12: Confusion matrix of the first hybrid network for the 2 × 6 configuration in a built environment.

Second hybrid network

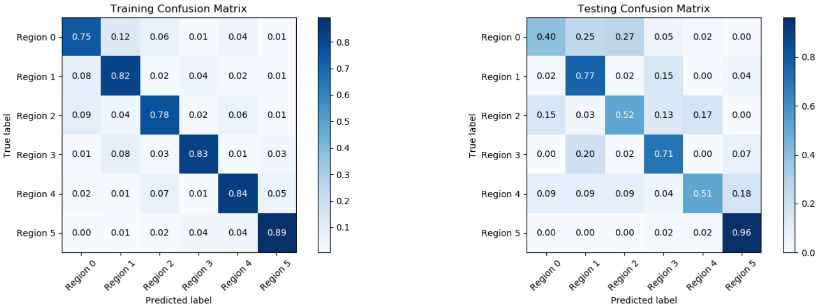


Figure 5.13: Confusion matrix of the second hybrid network in a 2 × 3 configuration in a built environment.

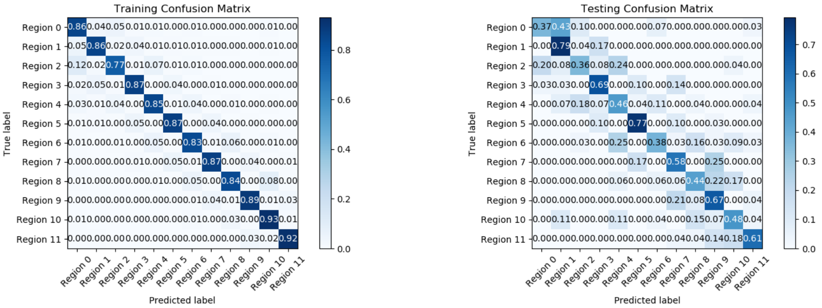


Figure 5.14: Confusion matrix of the second hybrid network for the 2 × 6 configuration in a built environment.