Engineering Applications of Artificial Intelligence 128 (2024) 107591

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

GAMMA: A universal model for calibrating sensory data of multiple low-cost air monitoring devices

Anh Duy Nguyen [a,1], Thu Hang Phung [a,1], Thuy Dung Nguyen [a], Huy Hieu Pham [b], Kien Nguyen [c,d], Phi Le Nguyen [a,*]

[a] School of Information and Communication Technology, Hanoi University of Science and Technology, Hanoi, Vietnam
[b] College of Engineering & Computer Science and VinUni-Illinois Smart Health Center, VinUniversity, Hanoi, Vietnam
[c] Institute for Advanced Academic Research, Chiba University, Chiba, Japan
[d] Graduate School of Engineering, Chiba University, Chiba, Japan

ARTICLE INFO

Keywords:
Smart cities
Air quality monitoring
Low-cost sensors
Calibration
AI
Multi-task learning

ABSTRACT

Due to global air pollution, there is a growing demand for accurate and large-scale air quality monitoring systems. Consequently, low-cost air monitoring devices have emerged as a potential alternative to expensive conventional ones. However, the low-cost devices' major drawback is their insufficient level of accuracy. This work investigates the problem of calibrating the sensory data, especially PM2.5 concentration, collected by low-cost sensor-based air quality monitoring devices. Recently, deep learning has emerged as a potential solution for data calibration instead of using traditional methods, whose accuracy is relatively low. Nevertheless, it generally incurs significant costs. Moreover, it is necessary to employ a dedicated calibration model for each device to increase precision, resulting in additional expenditures. To address the issue, this study provides a novel approach named GAMMA, which entails the development of a deep learning-based model capable of accurately calibrating data for multiple devices simultaneously. The proposed method leverages the multitask learning paradigm to solve the challenge of concurrently processing several devices' data. This involves capturing common features across all devices' data while also distinguishing the device-specific characteristics. Furthermore, GAMMA also employs the adversarial training approach to augment the accuracy of predictions. This method has been implemented and integrated into an air quality monitoring system in Hanoi, Vietnam. Comprehensive experiments are conducted on real-world data to demonstrate the superiority of GAMMA against the comparison benchmarks in terms of various metrics. Notably, GAMMA reduces MAE from 60.19% to 74.09% compared to the best comparison baseline. The source code is available at https://github.com/anhduy0911/FimiCalibIdea/tree/multi_attention.

1. Introduction

Rapid urbanization and industrialization, especially in developing countries, have indisputably contributed to global air pollution, affecting an estimated 96% of the world's population living in regions where air quality surpasses safe thresholds. According to the World Health Organization (WHO), in 2016, air pollution was associated with 4.2 million annual mortality (11.6 percent of all deaths) (WHO, 2016). In addition, it is claimed that air pollution is linked to acute and chronic diseases, including cardiovascular diseases (Andersen et al., 2012b; Goldberg, 2008), lung diseases (Andersen et al., 2012a; Gehring et al., 2010), and numerous types of cancer (Sakhvidi et al., 2020; Bai et al., 2020). In such a situation, air quality monitoring becomes essential. Normally, air quality can be evaluated using a variety of common air quality measured quantities, which are particulate matter (PM), Carbon monoxide (CO), Ozone (O_3), Nitrogen dioxide (NO_2), Sulfur dioxide (SO_2) (WHO, 2022). Among these, PM is a prevalent proxy pollutant, with two frequent types $PM_{2.5}$ and PM_{10} usually seen in many air quality sensors. Traditionally, air quality monitoring is conducted by a network of highly accurate monitoring stations following stringent criteria (Kulmala, 2018; Lagerspetz et al., 2019). These monitoring stations offer the remarkable advantage of exceptional precision. However, their installation and maintenance are extremely expensive, with each station costing hundreds of thousands or even millions of dollars. The high costs have impeded the widespread deployment of monitoring

[*] Corresponding author.
E-mail addresses: duy.na184249@sis.hust.edu.vn (A.D. Nguyen), hang.pt194758@sis.hust.edu.vn (T.H. Phung), dung.nt184244@sis.hust.edu.vn (T.D. Nguyen), hieu.ph@vinuni.edu.vn (H.H. Pham), nguyen@chiba-u.jp (K. Nguyen), lenp@soict.hust.edu.vn (P.L. Nguyen).
[1] Both authors contributed equally.

https://doi.org/10.1016/j.engappai.2023.107591
Received 24 May 2023; Received in revised form 14 November 2023; Accepted 19 November 2023
Available online 9 December 2023

stations, leading to limited coverage in air quality monitoring. Consequently, new technologies are necessary to provide spatial resolution in air quality monitoring without sacrificing precision.

Recently, low-cost sensors have emerged as a potentially cost-effective alternative, with sensors typically costing only a few thousands to several tens of thousands of dollars. Numerous low-cost sensor-based monitoring systems for air quality have been deployed (Liu et al., 2020; deSouza et al., 2020; Motlagh et al., 2020; Idrees and Zheng, 2020; Christakis et al., 2020; Migos et al., 2019; Moumtzidou et al., 2016). Christakis et al. in Christakis et al. (2022) introduced an open-source framework facilitating the monitoring of air quality, which comprises low-cost air quality stations and an application server for visualizing data. However, the major weakness of low-cost air quality sensors lies in their low accuracy, making the data provided by low-cost air quality sensors less reliable (Yadav et al., 2022; Dubey et al., 2022). Patton et al. (2022) assert that data obtained from low-cost sensors often indicates non-linear biases related to environmental conditions that cannot be captured by linear models. Besides, low-cost sensors provide less precise measurements compared to expensive reference equipment, which is mainly due to their sensitivity to other measured factors and their tendency to drift over time (Hofman et al., 2022). Furthermore, it is well established that low-cost sensors tend to overestimate $PM_{2.5}$ and PM_{10}. Specifically, as demonstrated in the evaluation conducted by Sayahi et al. (2019), two types of PM sensors (PM 1003s and PM 5003) overestimated $PM_{2.5}$ by factors of 1.89 and 1.47, respectively. Similarly, this overestimation exceeds a factor of 1.5 for PM_{10} (Carratù et al., 2020). Therefore, sensory data calibration is indispensable to enhancing the reliability of low-cost sensor-based air quality monitoring systems. In the literature, numerous calibration methods have been proposed, which can be classified into two main categories: hardware- and software-based approaches (Maag et al., 2018; Chattopadhyay et al., 2022; De Vito et al., 2020). This study focuses on the software-based calibration approach. The software-based methodology involves the utilization of sensory data that requires adjustment. This data is subsequently subjected to a calibration algorithm, which generates the corrected data. To accomplish this task, the calibration algorithm often utilizes data acquired from a reference device and strives to manipulate the sensory data to align more closely with the data generated by the reference device (Aula et al., 2022). Calibration algorithms can be classified into two main categories: traditional methods that utilize probabilistic (Saukh et al., 2015; Liu et al., 2021b) or machine learning techniques (Cordero et al., 2018; Hu et al., 2017; Liu et al., 2021a; Zimmerman et al., 2018; Johnson et al., 2018; Kumar and Sahu, 2021), and deep learning-based methods that leverage deep neural networks (Yu et al., 2020a,b; Wang et al., 2017; Bhatnagar et al., 2022; Loy-Benitez et al., 2020).

The traditional calibration approach usually leverages well-known machine learning techniques such as Autoregressive Moving Average (ARMA), k-nearest Neighbour (kNN), Random Forest (RF), and Gradient Boosting (GB). The authors in Tancev and Toro (2022) exploited variational Bayesian linear regression and variational Bayesian neural networks to correct the sensory data obtained from low-cost gas sensors. The authors utilized a nonlinear multivariate field calibration model in their study (Hofman et al., 2022) to tackle the issue of distant calibration. Instead of employing a reference device placed precisely at the same location as the target sensor, this methodology involves gathering historical data from several regulatory monitoring stations and employing it for calibration. Patton et al. (2022) employed probabilistic Gradient Boosted Decision Tree (GBDT) to construct direct field-calibration models, thereby obviating the necessity for labor-intensive laboratory calibration. Although basic machine learning methods are widely used because of their simplicity, they suffer from several weaknesses, including the inability to model complex relationships between features (for instance, linear regression fails to capture non-linear correlation); and the sensitivity to model parameters (e.g., Support Vector Machine (SVR) is very kernel-sensitive). In recent years, deep learning models have attracted attention and recognition for their ability to effectively capture intricate and non-linear relationships. Hence, the utilization of deep learning-based calibration presents a promising prospect for attaining accurate calibration of sensory data. Despite its potential, the utilization of deep learning in the calibration of sensory data is still at an early stage, with limited research available in the academic literature. Yousuf et al. in Hashmy et al. (2023) utilized Deep Neural Networks (DNNs) to calibrate several air quality indicators, including CO, SO2, NO2, O3, PM1.0, PM2.5, and PM10. Additionally, a comparative analysis was performed to evaluate the effectiveness of Deep Neural Networks (DNN) in comparison to various machine learning techniques, including Support Vector Regression (SVR), Random Forest (RF), K-Nearest Neighbors (KNN), and Linear Regression. In addition, the k-fold cross-validation method was employed to determine the optimal number of hidden layers and the corresponding number of perceptrons within each layer. In Liu et al. (2022), the authors leveraged meta-learning and few-shot learning techniques to tackle the data scarcity and multi-condition challenges in the sensory data calibration problem. Specifically, the historical data is employed to formulate several tasks corresponding to various conditions. Then, meta-learning is utilized to train a base model capable of quickly adapting to unknown conditions. The authors in Li et al. (2023) proposed a Variational Bayesian Blind Calibration algorithm to address the shortage of real data.

Despite attracting significant attention, sensory data calibration still remains an open issue. One of the most difficult challenges pertains to the fact that existing calibration methods are device-specific, i.e., each calibration algorithm is tailored to optimize for a particular device's data. The reason for this approach is that the data on each device is highly device-dependent; thus, a model trained for one device cannot calibrate data from other devices, as demonstrated in Fig. A.8 and Table A.10. This certainly cannot guarantee the scalability of the system when hundreds of devices are deployed on a broad scale. Then, ensuring both scalability and precision in calibrating sensory data is a critical challenge.

In this work, we aim to propose a universal calibration model that can effectively calibrate sensory data from various sensors. To accomplish this goal, it is crucial to incorporate two essential elements: (1) the identification of shared characteristics presenting in the sensory data from all sensors, which will be utilized in the calibration procedure for all sensors' data, and (2) the determination of unique features in the sensory data from each device, which will be employed to calibrate each device separately. This research proposes a novel deep learning-based approach named GAMMA (stands for **G**enerative **A**dversarial **M**ultitask Learning-based **M**ulti-C**A**libration), which fulfills the two criteria mentioned above. Fig. 1 illustrates the overview of GAMMA, which leverages the advantages of multitask learning and generative adversarial network (GAN) techniques. The multitask learning technique enables the performance of many tasks simultaneously, with each task responsible for the calibration of one device. The proposed multitask learning framework is built upon the following three fundamental principles. The first one is the extraction of common properties across all devices using a shared layer. Simultaneously, the attention mechanism is leveraged to identify inter-device correlations, thereby emphasizing the most critical relationships. Finally, separate prediction heads are assigned to each device to ensure accurate prediction results. In addition to multitask learning, the GAN technique is employed for deeper data calibration. The GAN architecture comprises a generator and a discriminator. The generator receives sensory data as input and attempts to produce calibrated signals that are as similar to the reference data as possible. Whereas the discriminator is trained to distinguish between the calibrated data generated by the generator and the ground truth. By utilizing such an adversarial training strategy, the generator will gradually generate calibrated data that resembles the reference data.

The main contributions of the research are as follows.

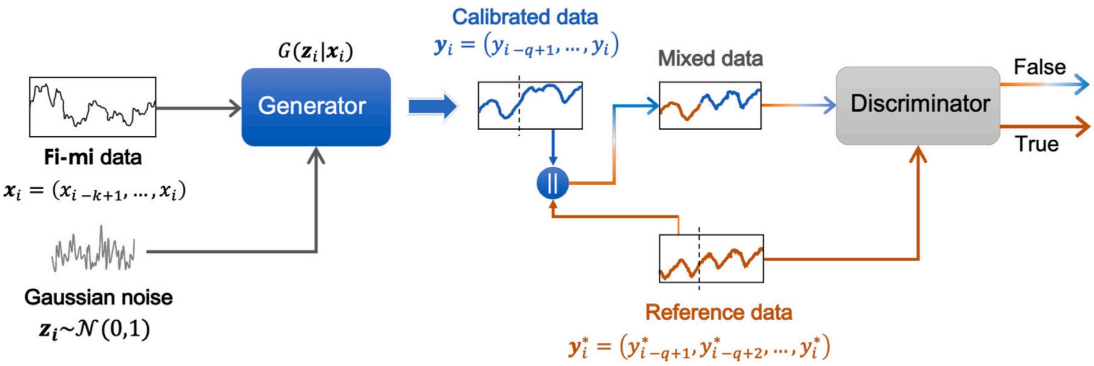

$G(\boldsymbol{z}_i|\boldsymbol{x}_i)$
Generator
Calibrated data
$\boldsymbol{y}_i = (y_{i-q+1}, \dots, y_i)$
Mixed data
Discriminator
False
True
Fi-mi data
$\boldsymbol{x}_i = (x_{i-k+1}, \dots, x_i)$
Gaussian noise
$\boldsymbol{z}_i \sim \mathcal{N}(0,1)$
Reference data
$\boldsymbol{y}_i^* = (y_{i-q+1}^*, y_{i-q+2}^*, \dots, y_i^*)$

Fig. 1. Simplified view of the GAN-based GAMMA framework.
The framework comprises two major blocks: the Generator and the Discriminator. The former is responsible for generating calibration data, while the discriminator is in charge of distinguishing between the ground truth and the calibrated time series.

- In this work, for the first time, a simultaneous multi-station air-quality data calibration for various measured quantities is introduced.
- GAMMA, a novel deep learning-based approach, is proposed to simultaneously calibrate multiple air quality quantities acquired by multiple low-cost sensors. This model utilizes the benefits of multitask learning and GAN approaches, where multitask learning is responsible for handling multiple tasks, while GAN contributes to improving calibration precision.
- A large number of actual sensors are implemented, and comprehensive experiments are conducted in real-world settings in order to evaluate the performance of GAMMA and compare it to the existing literature. The experimental results demonstrate that the proposed method surpassed state-of-the-art approaches. In particular, GAMMA reduces the calibration error by 60.19% to 74.09% in terms of the mean absolute error (MAE) metric when compared to the best competitor approach.

It is worth noting that this research does not aim to propose a solution for substituting sparse ground observations. It instead focuses on calibrating inaccurate sensory data. The proposed method can be applied to any sensor-based air quality monitoring device. As previously stated, the calibration of sensory data necessitates the utilization of precise data acquired from reference equipment. Therefore, before delving into the details of the proposed approach, we will describe a low-cost sensor-based air quality monitoring system as well as the reference device from which we obtained the data. The remainder of the paper is organized as follows. Section 2 presents the infrastructure of the low-cost sensor-based air quality monitoring system. The specifics of our proposed calibration approach are explained in Section 3. Section 4 depicts the experimental results. Finally, the conclusion and potential future directions are discussed in Section 8.

2. A low-cost sensor-based air quality monitoring system

In this study, we obtain sensory data from devices from a so-called Fi-Mi project. Therefore, in the following, we will give a brief introduction to Fi-Mi and the details of sensor-based air monitoring devices used in Fi-Mi in Sections 2.1 and 2.2, respectively. After that, we present the details of the reference instrument in Section 2.3. Finally, we describe our process to collect data and a statistical analysis concerning the collected data in Section 2.4.

2.1. System overview

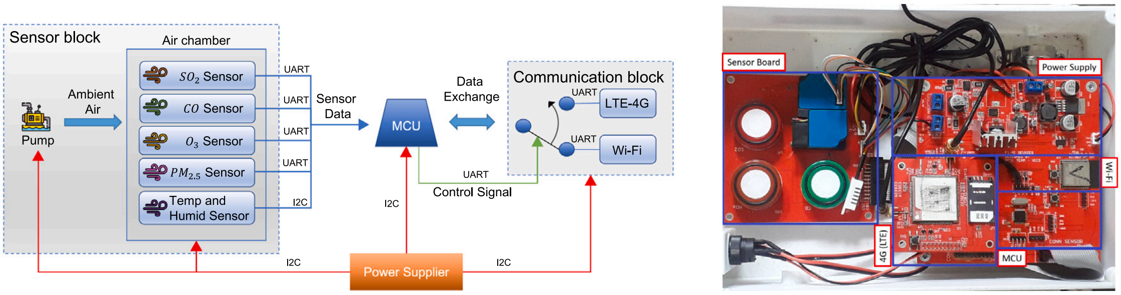
Fi-Mi is a low-cost sensor-based air quality monitoring system implemented in Hanoi, Vietnam. In Fi-Mi, the devices are positioned on mobile buses and periodically transmit the collected data to the back-end service for the purpose of data processing and filtering. The collected data is subsequently calibrated in order to minimize discrepancies with the use of reference devices and is then stored for future predictive purposes. Fi-Mi system consists of three layers as follows.

- **Sensing layer:** This is the lowest layer, which comprises low-cost air quality monitoring devices; each is called a Fi-Mi device and is equipped with several sensors capable of gathering air quality and meteorological data. Each Fi-Mi device consists of four main components, namely sensors, Micro Control Unit (MCU), Global Positioning System (GPS) Unit, and communication modules. The system utilizes sensors of Wisen and Alphasense, MCU of ST Microelectronics, GPS unit of SimCom, and communication modules of Espressif Systems.
- **Data acquisition layer:** This layer is responsible for transferring data from the monitoring devices to the servers, as well as preprocessing and calibrating that data. Once collected from the Fi-Mi devices, the data undergoes preprocessing, which involves tasks such as noise filtering and format adjustments. In essence, this preprocessing phase refines the raw data obtained from low-cost devices, which is initially minimally processed, including noise filtering and outlier removal. After that, the preprocessed data needs to be passed through a calibration model with the objective of aligning the calibrated data as closely as possible with measurements obtained from the reference instrument at the same location. In particular, deep learning is employed to calibrate the data before storing it for further prediction.
- **Data visualization layer:** This layer is a web application (http://demo.fi-mi.vn/app) that provides real-time air quality measured quantities to the end-users (as illustrated in Fig. A.9).

Prior to implementing low-cost devices into the bus system, it is critical to establish an effective methodology for accurately calibrating the data collected by those devices. From that motivation, this research was conducted in a laboratory to provide reliable multi-device sensory data calibration using only one model. The calibration process carried out involves the participation of five Fi-Mi devices and a reference instrument acting as the ground-truth for the model. These devices are placed in the same monitoring location in Hanoi, Vietnam, which will be described in detail in Section 2.3. The following section provides a comprehensive description of the design of each Fi-Mi device, followed by the introduction of the reference device intended for the calibration of Fi-Mi devices. Subsequently, the analysis of the data collected from all low-cost devices and the reference instrument is presented.

2.2. Air quality monitoring devices

Figs. 2(a) and 2(b) depict the design and real implementation of a Fi-Mi device, respectively. The Fi-Mi device consists of four components: a sensor block, a power supply, a microcontroller unit (MCU),



(a) Block diagram (b) Real implementation

Fig. 2. Block diagram (a) and capture of a Fi-Mi device's real implementation (b).
A Fi-Mi device comprises four main components: a sensor block, a power supply, a microcontroller unit (MCU), and a communication block. The air-quality information collected by sensors is exchanged through an MCU (Micro Control Unit). The communication is performed over LTE-4G or Wi-Fi. The power supplier provides all other components with their energy demands.

and a communication block. The device's components originate from numerous manufacturers. The MCU is supplied by ST Microelectronics, while the power supply is provided by Analog Devices. The sensors were developed by Wisen and Alphasense. The Wi-Fi module is produced by Espressif Systems. Lastly, the LTE and GPS components are supplied by SimCom. Among these elements, the sensor block consists of two main elements: the air chamber and the pump. This pump is attached to the door of the air room and circulates the atmosphere within the room, while the air chamber receives the ambient air from the pump and processes the data using various small instruments. There are five sensors that measure the density of Sulfur dioxide (SO_2), Carbon monoxide (CO), Ozone (O_3), PMx (i.e., PM_1, $PM_{2.5}$, and PM_{10}), temperature, and humidity. The data measured by each sensor is stored separately, allowing for the direct extraction of PM2.5-related information from the data provided by the PM2.5 sensor. Those sensors are set up according to the WHO Global Air Quality Guidelines on thresholds for key air pollutants, which are applied worldwide for outdoor environments (WHO, 2022). The measurement ranges of each air quality index and meteorological quantity are demonstrated in Table 1.

The microcontroller unit in the Fi-Mi device is the STM32F103C6T8 MCU. There are several exchange protocols utilized by this communication, which include Universal Asynchronous Receiver/Transmitter (UART), I2C, and Serial Peripheral Interface (SPI). The periphery list and the associated protocols are summarized in Table 2. Among these protocols, SPI is not demonstrated in Fig. 2(a) since it represents the communication standard between the MCU and the external memory.

The Fi-Mi communication block is developed to accommodate two prevalent wireless technologies (i.e., Wi-Fi and 4G Long Term Evolution (LTE)), and the Global Positioning System (GPS). For the Wi-Fi connection, the equipment employs the microcontroller ESP32 with embedded Wi-Fi. Besides, the Fi-Mi gadget utilizes the multifunctional module for 4G (LTE) and GPS. Due to the low-cost advantage, Wi-Fi is prioritized over LTE communication. Therefore, when the device needs to transmit data, it first verifies if the Wi-Fi connection is accessible or not. If yes, the data will be transferred over Wi-Fi; otherwise, the 4G (LTE) connection will be utilized.

The device's power management unit is the IC LTC4162, which operates in two modes: normal mode and sleep mode. When Fi-Mi's power falls below the predefined threshold, it enters sleep mode so as to conserve energy. There are three types of direct current (DC) supply voltages: 5 V for the sensor block, 4.2 V for the 4G (LTE) communication module, and the same potential of 3.3 V for both the MCU module and Wi-Fi module. To provide those components with their power demand, the Fi-Mi device might derive energy from three possible resources: a rechargeable 3.7 V 4000 mAh battery, a solar panel, and the associated vehicle's accumulator (see Fig. 3).

Table 1
The measurement ranges of the sensors.

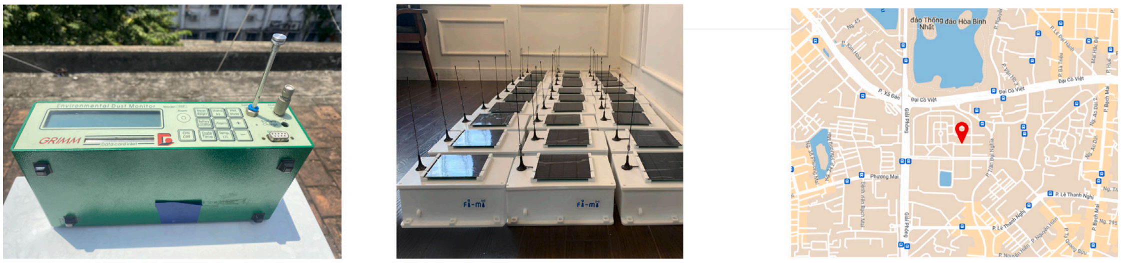
	Sensor	Range	Unit
PMx.x	ZH-03B	0 to 1000	$\mu g/m^3$
SO_2	ZE-12	0 to 2	ppm
CO	ZE-15	0 to 500	ppm
O_3	ZE-25	0 to 10	ppm
Temperature	SHT21	−40 to 125	°C
Humidity	SHT21	0 to 100	$\%RH$

Table 2
The communication interfaces of the MCU and Peripheral devices.

Peripheral	Function	Communication
SIM7600CE	4G (LTE)	UART
ESP32	Wi-Fi	UART
ZE-12	SO_2 sensor	UART
ZE-25	O_3 sensor	UART
ZH-03B	Particulate matter sensor	UART
SD card	External memory	SPI
SHT21	Temperature and humidity sensor	I2C
LTC4162	Power management	I2C

2.3. Reference instrument

This work utilizes a highly accurate air quality monitoring device, GRIMM 107 (GRIMMTechnologies Inc., 2005), located in Hanoi, Vietnam, as the reference monitoring station. GRIMM 107 is a monitoring instrument that measures fine-grained air pollution indices (i.e., PM_1, $PM_{2.5}$, and PM_{10}). GRIMM 107 employs a flow-controlled pump to get a continuous air sample. The measurement of particles is conducted using the physical principle of orthogonal light scattering. It provides online readings (set at 1-min intervals in this study, which is equivalent to the sampling rate of 1 : 60 Hz.) and is hence useful to compare with low-cost sensor data of the same temporal resolution. For this reason, GRIMMs are usually used in similar calibration exercises (Wang et al., 2020). The reference device is located at the Center for Environmental Monitoring, which belongs to Hanoi University of Science and Technology, on the rooftop of a three-storey building named C5. The monitoring station's location is illustrated in Fig. 3(c). This station is 10 meters above the ground and located adjacent to Giai Phong Street, one of Hanoi's arterial roads. The street is lined with numerous apartment buildings, hospitals, and universities. This street is traversed daily by various vehicles, including cars, buses, and trucks, resulting in frequent air pollution. This study focuses on calibrating $PM_{2.5}$ and PM_{10}, two of the most critical pollution measured quantities affecting humans significantly (Zhang et al., 2022; Baron, 2022). Accordingly, $PM_{2.5}$ and PM_{10} indices monitored by GRIMM 107 are extracted and employed as the ground truth of the calibration model.



(a) GRIMM 107

(b) Fi-Mi devices

(c) Monitoring location. The red symbol indicates the accurate location where all the low-cost devices as well as the reference instrument are placed.

Fig. 3. The reference instrument (GRIMM 107), Fi-Mi devices, and their locations. The reference instrument and Fi-Mi devices are located at the same location.

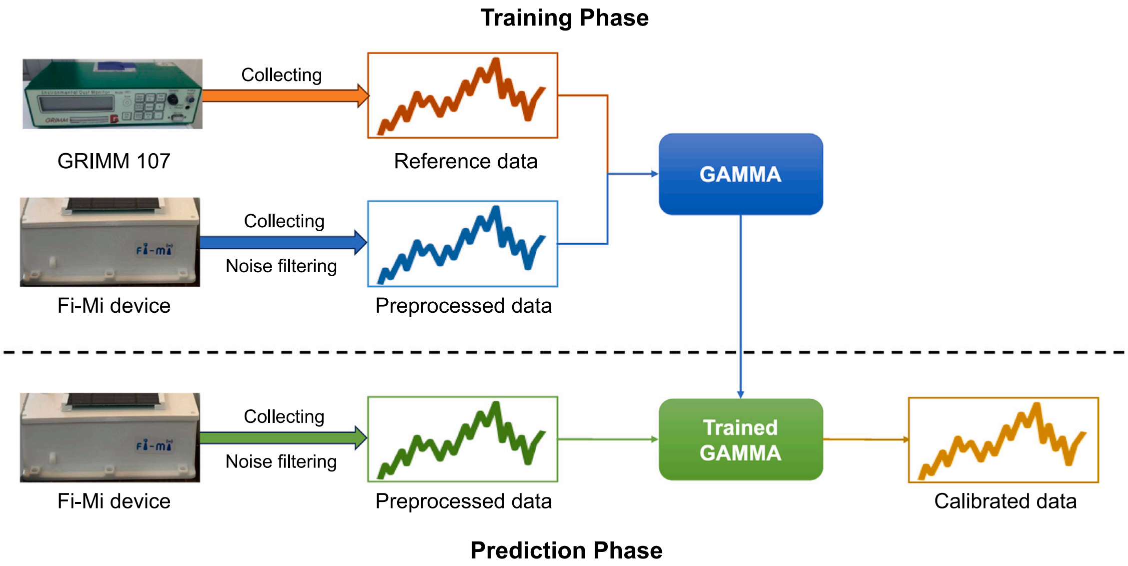


Fig. 4. Flowchart of the proposed model.
Data collected from Fi-Mi devices is filtered to remove noisy instances and then passed through the GAMMA model to get calibrated sensory data.

Given the reference instrument GRIMM 107 and the low-cost Fi-Mi devices (as presented in Fig. 3(a) and Fig. 3(b)), Fig. 4 illustrates the flowchart of GAMMA regarding the training phase and prediction phase, outlining the procedure for transforming raw sensory input into calibrated data.

2.4. Data collection and analysis

The data was gathered using five Fi-Mi devices and the reference instrument, which are located at the same monitoring location as mentioned before, from February 11, 2022, to February 22, 2022. Each Fi-Mi device has a sampling rate of 1 : 10 Hz, whereas the sampling rate for GRIMM 107 is 1 : 60 Hz. Due to the varying time resolutions between the measurements of Fi-Mi devices and the reference equipment, the data is integrated and averaged every minute in order to construct a synchronized database. Specifically, the final value for a minute is calculated by averaging the sensory data values from low-cost devices over the same minute of time. This synchronized database can be accessed through (Anon, 2023). Hence, it can be inferred that within a time interval of one minute, a data point is obtained from each device, which is relevant to a sampling rate of 1 : 60 Hz.

There were several minor data gaps in the measurements of Fi-Mi devices and the reference instrument. To detect the anomaly in the dataset, exponentially weighted moving average (EWMA) (Hunter, 1986) is employed as the anomaly detection module. Due to technical issues with data acquisition, sensor measurements, and communication interruptions, the missing data could not be recovered. From the total data points, the longest consecutive missing data is 7.24%, therefore, this anomaly is deleted from the dataset. For example, if data from time step t is removed, then data from time step $t-1$ and time step $t+1$ are concatenated. Moreover, the same time step t in the reference data is also removed to ensure that the input has the same dimension as the ground-truth. As a result, a dataset of 11030 preprocessed samples is acquired, each sample encompasses various air quality and meteorological features, such as $PM_{2.5}$, PM_{10}, humidity, and temperature. In comparison to previous studies conducted on the same subject, the quantity of data available can be considered sufficiently reliable. For instance, Loy-Benitez et al. (2020) uses a dataset containing only 716 samples collected over a month at an hourly resolution. Additionally, two other datasets consisting of 7674 and 7753 instances, respectively, with each instance representing a one-hour time period, are chosen in Yu et al. (2020a).

Table 3
The performance of Fi-Mi devices against the reference instrument before calibration.

Feature	Metrics	Fi-Mi #1	Fi-Mi #2	Fi-Mi #3	Fi-Mi #4	Fi-Mi #5
$PM_{2.5}$	MAE	21.47	17.94	21.76	22.50	18.68
	RMSE	27.23	23.78	28.41	29.73	23.69
	R2	−0.46	−0.11	−0.59	−0.74	−0.10
PM_{10}	MAE	25.37	23.37	26.00	27.37	22.89
	RMSE	32.47	30.63	33.97	35.92	29.17
	R2	−0.48	−0.32	−0.62	−0.81	−0.20

The statistical analysis is performed concerning the two measured quantities, namely $PM_{2.5}$ and PM_{10}. An adequate calibration model can be created using the data preprocessed by the procedure described in the previous section and the insights obtained from the analysis. In the following, an exploratory data analysis was conducted using sensory data from Fi-Mi devices and a reference instrument. The analysis involved several metric computations and the generation of histograms and correlation graphs (see Appendix A for graphic information).

There are three analysis metrics considered in this section: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R2 Score (R2). MAE and RMSE are utilized to determine the precise difference, while R2 is used to compare the trends of various devices. Performance of Fi-Mi devices before calibration is illustrated in Table 3. These statistical results validate the correlation between the reference device and Fi-Mi devices. Specifically, Fi-Mi #2 and Fi-Mi #5 are the most similar devices to GRIMM 107 with RMSE values of 23.78 and 23.69 concerning $PM_{2.5}$, respectively. Regarding the MAE metric, Fi-Mi #2 has the most similar data to the reference data, which possesses the MAE of 17.94 concerning $PM_{2.5}$ and 23.4 for PM_{10}. This observation also holds when considering the RMSE error. Notably, R2 scores are negative in all cases. This phenomenon indicates that the preprocessed data from Fi-Mi devices does not follow the trend of the reference one.

3. GAMMA: Generative adversarial multitask learning-based multiple calibration

In this section, first, an explicit mathematical formulation of the problem is provided in Section 3.1. Next, the overview of the proposed sensory data calibration method, named GAMA, is introduced in Section 3.2. As GAMMA is based on the GAN structure, it consists of three major components: the Generator, the Discriminator, and the Loss function. These components are described in Sections Section 3.3, 3.4, and 3.5, respectively.

3.1. Problem formulation

Let $S_1, \ldots, S_n$ be the n low-cost air quality monitoring devices. For each device S_i ($i = 1, \ldots, n$), the vector $x_t^{(i)}$ represents sensory air quality and meteorology measured quantities collected by S_i at time step t. Given the sensory data collected by n devices in m time steps (denoted as $\mathbf{x}^{(i)}$ collected by S_i), along with the data gathered by the reference instrument at the same time and location (denoted as $\mathbf{y}$), the objective is to adjust the sensory data to maximize its similarity to the reference data. Below is the mathematical formulation.

Input:

$$\mathbf{x}^{(i)} = \{x_j^{(i)}, x_{j+1}^{(i)}, \ldots, x_{j+m-1}^{(i)}\}, (i = 1, \ldots, n; j \geq 0)$$

$$\mathbf{y} = \{y_j, y_{j+1}, \ldots, y_{j+m-1}\},$$

Output:

$$\begin{aligned}\tilde{\mathbf{y}}^{(i)} &= f(\mathbf{x}^{(i)})\\ &= \{\tilde{y}_{j+m-q}^{(i)}, \tilde{y}_{j+m-q+1}^{(i)}, \ldots, \tilde{y}_{j+m-1}^{(i)}\},\end{aligned}$$

Objective function:

$$\mathcal{L} = \sum_{i=1}^{n} l(\mathbf{y}, \tilde{\mathbf{y}}^{(i)}),$$

where q is the length of the calibrated output series, and l Note that the length of the output could be any positive integer as long as it does not exceed the length of the input sequence, m.

3.2. Overview of GAMMA

This section will begin with an introduction of the motivation and intuition behind the framework, followed by the details of the proposed training procedure and loss function.

Motivation. Data calibration can be considered a conventional regression task with sensory data as input and calibrated data as output. For such a task, a common approach is to construct a model that provides the exact calibrated data, which is as close to the ground truth as possible. However, it is observable that the measurement errors induced by low-cost sensors exhibit significant fluctuations. These constraints make the calibration task more challenging, leading to substantial variations in calibration accuracy. To this end, instead of following the traditional approach, which offers the calibration results directly, this work instead employs a novel approach that aims at predicting the distribution of the calibration results. Formally, the objective now shifts towards modeling $P(\tilde{\mathbf{y}}|\mathbf{x})$, the probability distribution for $\tilde{\mathbf{y}}$, given the raw sensory data $\mathbf{x} = \{x_i, x_{i+1}, \ldots, x_{i+m-1}\}$ Given the projected distribution, calibrated data can be generated using the mean, and the model's confidence can be approximated through the distribution's variance. This probabilistic prediction technique aligns well with the constructed real-time air quality monitoring system, allowing for the assessment of system reliability and the subsequent design of a suitable continual learning strategy.

Model architecture. Inspired by Conditional GAN (CGAN) (Mirza and Osindero, 2014), the novel calibration framework GAMMA is introduced that (1) leverages the idea of probabilistic distribution prediction to learn the underlying probability distribution of the input data (i.e., the sensory data) and generate samples following the desired distribution (i.e., the calibrated time series); and (2) employs the adversarial training paradigm to train the calibrator. GAMMA consists of two major components: the Generator and the Discriminator. The Generator functions intuitively as the calibrator, receiving sensory data and random noise to produce a calibrated time series. In this work, Gaussian noise following the distribution of $\mathcal{N}(0, 1)$ is employed. Specifically, instead of producing a single calibrated time series for each sensory data input, a multitude of random noises are generated and applied to produce diverse calibrated time series. This approach facilitates the capture of the underlying distribution of the output. The mean of this distribution is then utilized as the final output for calibration. The Discriminator is employed to distinguish between counterfeit (i.e., calibrated data) and legitimate (i.e., the ground truth provided by the reference instrument) data. While the Generator is trained to produce calibrated time series that resemble the ground truth as closely as possible, the Discriminator considers the ground truth to be authentic samples and attempts to separate them from the others. By training the Discriminator and Generator in such an adversarial manner, the Generator eventually acquires the ability to deceive the Discriminator. This is achieved when the calibrated time series generated by the Generator exhibits a close resemblance to that of the reference instrument. Fig. A.10 depicts the detailed overall architecture of GAMMA.

Training strategy. Because of the MinMax optimization problem, GAN-based frameworks are known for their unstable training procedure (Goodfellow et al., 2016; Metz et al., 2016; Gong et al., 2019; Chu et al., 2020). Given the opposing optimization goals of the Generator and Discriminator, the prevention of model collapse during the training phase necessitates the implementation of a robust training approach. In light of this, a two-stage training approach is proposed as follows. The training phase of the framework commences without introducing noise into the generator. In this manner, the training process becomes more stable and straightforward. After the training, only the layers that are directly related to the noise input in the second stage are replaced. This soft-mapping procedure can be regarded as a warm-up step whose trained weights are subsequently utilized in the second stage to initialize the model. During the second phase, transfer learning is applied to the complete GAMMA framework, involving the utilization of random noise in the generator. Furthermore, acknowledging the discrepancy in convergence rates between the training of the Generator and Discriminator, the Discriminator is trained k times more frequently than the Generator within each epoch. This k is a hyperparameter that should be tuned. Finally, instead of presenting the Discriminator with the calibrated data generated by the Generator, a mixed time series is constructed. This mixed series is created by concatenating segments of the calibrated time series with other segments of the ground truth data. These interleaved time series are regarded as counterfeits and must be identified by the Discriminator. The training approach is specified by the Algorithm 1.

3.3. *Multitask learning-based generator*

In the literature, several multitask learning approaches have been introduced. However, because these methods are designed for general purposes, they are not optimal for the problem under consideration. Therefore, this study leverages the main idea of the multitask paradigm and designs a multitask learning-based generator tailored to the problem. In this work, for the first time, a simultaneous multi-station air-quality data calibration for various measured quantities is introduced. Moreover, the proposed multitask learning model is optimized for capturing the sensory data's characteristics. Due to the fact that the data for the tasks is measured by Fi-Mi devices of the same design, they possess both shared and device-specific characteristics. Consequently, the Generator component consists of a shared feature extractor and a device-specific feature extractor. The former helps extract generic information from the data of all devices, while the latter is responsible for capturing device-specific data. In addition, the proposed method also utilizes the attention mechanism to improve the model's precision.

This section begins with an overview of the Generator, followed by an explanation of each module's details.

3.3.1. *Overview*

The Generator comprises two modules, namely the *Shared feature extractor* and *Device-specific feature extractor* as depicted in Fig. 5. The former, which consists of several LSTM layers, is responsible for identifying shared characteristics among all devices. In the latter, a Device Identification module is employed to determine each device's identity. This block's output is then combined with the features generated by the Shared feature extractor and passed into the Inter-device Attention block to generate context vectors corresponding to the devices. Additionally, an Intra-device Attention mechanism is developed to capture the inherent correlations within the time series of each device. This mechanism is accountable for highlighting the most important features.

3.3.2. *Shared feature extractor*

The Shared Feature Extractor is responsible for extracting all devices' shared characteristics. To do this, a single LSTM layer is deployed to receive and process input data from all devices. The key idea behind GAMMA is that, instead of independently training n LSTM layers for all devices under investigation, the gate outputs of the single LSTM layer are merged and applied to the input of every device during each time step. This combination generates unified gates that incorporate information from all devices. With this idea, a substantial reduction in the number of parameters to be learned by GAMMA is achieved. Moreover, a novel module capable of extracting the most representative, device-unspecific features from the time series is also being developed. Let $f_k^{(t)}, i_k^{(t)}$, and $o_k^{(t)}$ be the Forget, Input, and Output gates of the kth LSTM layer at time step t ($t = 1, \dots, m$), then the $f_k^{(t)}, i_k^{(t)}$ and $o_k^{(t)}$ ($k = 1, \dots, n$) is integrated by a fully connected (FC) layer to produce new gates $f_k^{(t)'}, i_k^{(t)'}$ and $o_k^{(t)'}$, which are then put back into the corresponding LSTM layer. This mechanism allows the information extracted from the time series of a device to spread over the entire shared LSTM layer and affect other devices' output presentations, thereby enriching the information used in calibrating each device. Moreover, as the combined gates are constructed by fusing the gate outputs of the shared LSTM layer, these gate outputs are trained using data from all devices, allowing the extraction of general features shared by all devices. The mathematical formulation for each time step t in the Shared Feature Extractor can be expressed as follows.

$$\begin{aligned} f_k^{(t)'} &= \sigma_f\left(f_1^{(t)}, f_2^{(t)}, \dots, f_n^{(t)}\right), \\ i_k^{(t)'} &= \sigma_i\left(i_1^{(t)}, i_2^{(t)}, \dots, i_n^{(t)}\right), \\ o_k^{(t)'} &= \sigma_o\left(o_1^{(t)}, o_2^{(t)}, \dots, o_n^{(t)}\right). \end{aligned} \tag{1}$$

In Eq. (1), σ denotes a fully connected layer, which takes all devices' gates as its input and produces all devices' modified gates. The adjusted gates are then fed back into the original LSTM cell to calculate the subsequent timesteps. Fig. 5 depicts the *roll-out view* of this module, allowing for a better observation of its inner workings.

3.3.3. *Device-specific feature extractor*

This module is in charge of extracting specific features regarding each device. It includes the Device Identification Module, Intra-device Attention, and Inter-device Attention blocks. The Device identification module emphasizes the most identifying characteristics that assist in distinguishing each device from the others. In the meantime, the two attention blocks (i.e., intra-device and inter-device attentions) model correlation within the time series of each device and across devices, highlighting the essential components and boosting the calibration accuracy.

Device Identification. One common architectural design in multi-task learning is to use a multi-stream approach (Duong et al., 2015)(Yang and Hospedales, 2016), in which multiple independent blocks are introduced, each extracting individual information concerning a particular task. However, this strategy will result in a linear expansion of model parameter size as the number of tasks increases. In this paper, an alternative approach involves the utilization of a device-specific feature extractor block to extract information specific to every device. To this end, the device Identification module acts as an identifier, assisting in generating a unique vector representing a device. Specifically, the identifiers are generated by passing the device IDs through two fully connected layers σ_{i1}, σ_{i2}, combined with the activation function ϕ to obtain n identification vectors $\mathbf{i}_1, \dots, \mathbf{i}_n$ corresponding to n devices. Below is the formula for the Device Identification block

$$\mathbf{i}_j = \sigma_{i2}(\phi(\sigma_{i1}(ID_j))), \ \forall j = 1, \dots, n,$$

where ID_j is the one-hot vector representing the ID of device S_j.

Inter-device Attention. Given the identification vectors from the Device Identification block alongside the features extracted by the Shared

Algorithm 1 Mini-batch stochastic gradient descent training of generative adversarial networks.

```
Input:
▷ $\mathcal{M}(\theta_G, \theta_D)$ - GAMMA model
▷ $\mathcal{G}(\hat{\theta}_G)$ - GAMMA Generator model
▷ $p_{\text{data}}$ - Data distribution
▷ $I$ - First stage's iteration limit
▷ $T$ - Second stage's iteration limit
▷ $K$ - Discriminator iteration limit
▷ $\eta$ - Global learning rate
▷ $\mathcal{L}_G, \mathcal{L}_D$ - Chosen loss functions for Generator and Discriminator
Output:
▷ $\mathcal{M}(\tilde{\theta}_G, \tilde{\theta}_D)$ - GAMMA model with trained weight

Initialize $\hat{\theta}_G^0 \leftarrow \hat{\theta}_G^0$; $\theta_G^0 \leftarrow \theta_G^0$; $\theta_D^0 \leftarrow \theta_D^0$
for $i = 1, \ldots, I$ do                                                          ▷ First stage training loop
    $b_x \leftarrow \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$; $\forall i : \boldsymbol{x}^{(i)} \sim p_{\text{data}}$          ▷ Sample real minibatch
    $b_y \leftarrow \{\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(m)}\}$;                                  ▷ Corresponding groundtruth minibatch
    $b_z \leftarrow \{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$; $\forall i : \boldsymbol{z}^{(i)} \leftarrow \mathcal{G}(\boldsymbol{x}^{(i)})$     ▷ Corresponding synthesis minibatch
    $\hat{\theta}_G^i \leftarrow \hat{\theta}_G^{i-1} - \eta \nabla_{\theta_{\mathcal{G}}} MSE(b_z)$                          ▷ Update the Generator
end for
Softmapping $\theta_G \leftarrow \hat{\theta}_G^I$
for $t = 1, \ldots, T$ do                                                          ▷ Second stage training loop
    for $k = 1, \ldots, K$ do
        $b_x \leftarrow \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$; $\forall i : \boldsymbol{x}^{(i)} \sim p_{\text{data}}$      ▷ Sample real minibatch
        $b_n \leftarrow \{\boldsymbol{n}^{(1)}, \ldots, \boldsymbol{n}^{(m)}\}$; $\forall i : \boldsymbol{n}^{(i)} \sim p_{\text{noise}}$     ▷ Sample noise minibatch
        $b_y \leftarrow \{\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(m)}\}$;                              ▷ Corresponding groundtruth minibatch
        $b_z \leftarrow \{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$; $\forall i : \boldsymbol{z}^{(i)} \leftarrow G(\boldsymbol{x}^{(i)}, \boldsymbol{n}^{(i)})$   ▷ Corresponding synthesis minibatch
        $\theta_D^k \leftarrow \theta_D^{k-1} - \eta \nabla_{\theta_D} \mathcal{L}_D(b_y, b_z)$                     ▷ Update the Discriminator
    end for
    $b_x \leftarrow \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$; $\forall i : \boldsymbol{x}^{(i)} \sim p_{\text{data}}$          ▷ Sample real minibatch
    $b_n \leftarrow \{\boldsymbol{n}^{(1)}, \ldots, \boldsymbol{n}^{(m)}\}$; $\forall i : \boldsymbol{n}^{(i)} \sim p_{\text{noise}}$         ▷ Sample noise minibatch
    $b_z \leftarrow \{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$; $\forall i : \boldsymbol{z}^{(i)} \leftarrow G(\boldsymbol{x}^{(i)}, \boldsymbol{n}^{(i)})$       ▷ Corresponding synthesis minibatch
    $\theta_G^t \leftarrow \theta_G^{t-1} - \eta \nabla_{\theta_G} \mathcal{L}_G(b_z)$                               ▷ Update the Generator
end for
Return $G(\tilde{\theta_G})$
```

Feature Extractor, the Inter-device Attention module will be discussed in detail. This module's objective is to capture the correlation between the devices. Denote by n feature vectors $\mathbf{h}_1, \ldots, \mathbf{h}_n$ the output of the Shared Feature Extractor, each of which conveys information about the last time step of each device's time series. Clearly, the correlations between the devices are different, i.e., for a considered device, some devices may have substantial connections with it while others may not exhibit any relevant properties. Consequently, the attention block is applied to determine the relationship between the devices and to emphasize the most significant matters. Specifically, for each device S_j, its identification vector $\mathbf{i}_j$ is used as the query, and $\mathbf{h}_1, \ldots, \mathbf{h}_n$ are leveraged as the keys. In the meantime, the latent series $\mathbf{f}_1, \ldots, \mathbf{f}_n$ produced by the LSTM layer of the Shared Feature Extractor are directly used as the value of the Inter-device Attention. Finally, the context vectors are obtained by performing a weighted sum on the values using the attention weights.

Given the identification vector $\mathbf{i}_j$ generated by the Device Identification block and the latent features extracted by the Shared Feature Extractor, the working of the Inter-device Attention can be mathematically formulated as follows.

$$s_{ij} = \sigma(\mathbf{h}_i || \mathbf{h}_j), \;\; w_{ij} = \frac{\exp(s_{ij})}{\sum_{\forall k} \exp(s_{ik})},$$
$$\mathbf{c}_i = \sum_{\forall k} w_{ik} \times \mathbf{f}_k.$$

Intra-device Attention. After passing through the Device Identification module and Inter-device Attention block, for each device S_j, the identification i_j, and context vector c_j are obtained. In the final block of the Generator, an additional attention mechanism named Intra-device attention is applied. This mechanism aims to capture the internal correlation within each device's data, thereby enhancing the quality of the final calibration. To elaborate, a self-attention mechanism is initially employed within the time series c_j to model temporal correlations across all time steps. This approach serves to emphasize time steps that correlate most with each other. Moreover, a global attention is harnessed using the identification vector i_j as the query and the context vector c_j as the key/value. It is worth noting that i_j serves as the identifier of the jth device and conveys the most distinguishing information about this device. Therefore, by employing i_j as the query to perform attention on c_j, the device-specific semantic information can be learned, showing which parts of c_j are the most important to the time series of the jth device. The formulas concerning Intra-device Attention are identical to those of Inter-device attention mentioned previously.

3.4. Discriminator

While the primary aim is the establishment of a robust Generator architecture, the Discriminator plays a crucial role in achieving this goal. This module's architecture must be capable of distinguishing between the reference series and the calibrated series while simultaneously being simple and shallow enough for improved convergence. Here, the Discriminator consists of a bidirectional LSTM bi-LSTM block followed by a fully connected layer. First, the input time series is passed through the bi-LSTM block to extract temporal features. Through this block, the critic is empowered to analyze the series from two perspectives and extract the most important features to differentiate them. The output of the bi-LSTM layer is then sent to the FC layer, which acts as a classifier, generating the final prediction result indicating whether the inference equipment generated the input time series.

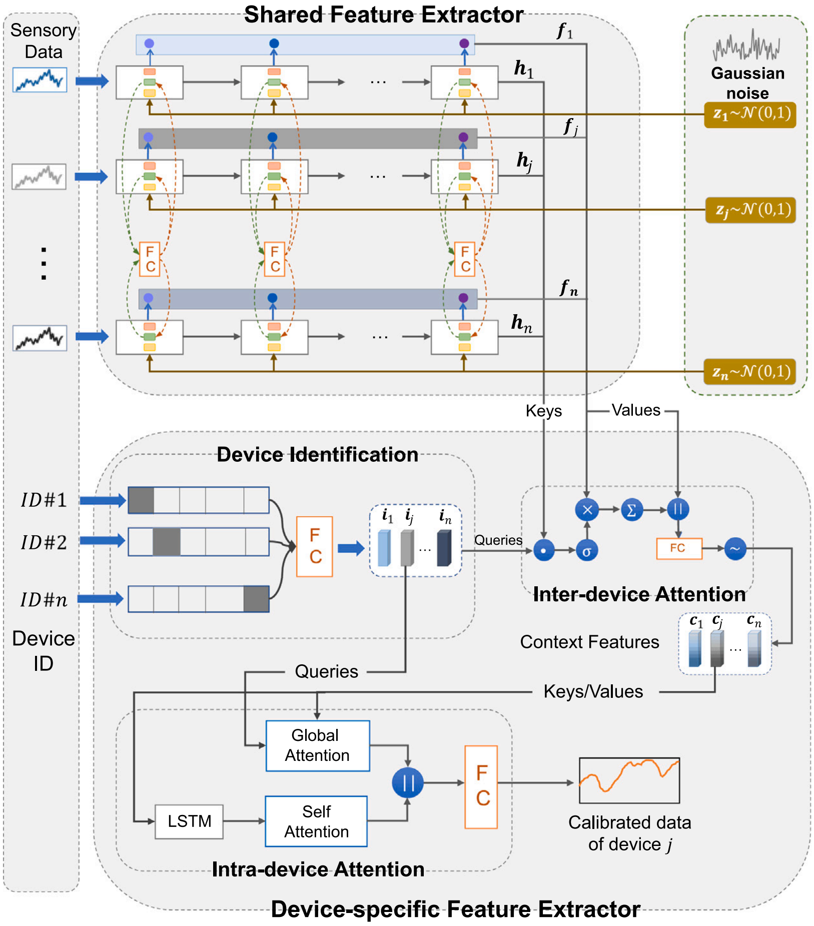


Fig. 5. Detailed architecture of GAMMA's Generator module.
The Generator comprises two primary components: the Shared Feature Extractor and the Device-specific Feature Extractor. The former consists of a shared LSTM layer, which is responsible for extracting representative characteristics shared by all devices. In the meantime, the latter includes the Device Identification module and two attention mechanisms. The Identification module identifies the devices, whereas the Attention modules simulate inter- and intra-device correlation to obtain context vectors utilized to produce the final calibration result.

3.5. Loss function

This section begins with a discussion of the primary loss employed in adversarial training of both the Generator and Discriminator. Following this, a novel contrastive loss is introduced to enhance device-specific representation.

GAN-based losses. The original work (Goodfellow et al., 2014) combines the Sigmoid function and the cross entropy loss to indicate the probabilistic result. However, this function may get saturated very quickly. For example, when the input value x is significantly significant, the difference between the Sigmoid function's outputs becomes trivial, resulting in a gradient that approaches 0 and so triggering the vanishing gradient problem. Moreover, as pointed out in Mao et al. (2017), the log loss employed disregards the distance between the output value and the decision boundary, hence failing to accurately represent the optimization target. To this end, a novel loss is proposed, combining the Least Squares Loss and the gradient penalty introduced in Gulrajani et al. (2017), serving as the principal GAN training objective. The primary loss function can be expressed as follows.

$$\begin{aligned}\mathcal{L}_D =& \frac{1}{2}\mathbb{E}_{\tilde{\mathbf{y}}\sim p_{\text{data}}(\tilde{\mathbf{y}})}\left[(D(\tilde{\mathbf{y}}|\mathbf{x})-1)^2\right] + \\ & \frac{1}{2}\mathbb{E}_{\boldsymbol{n}\sim p_{\text{noise}}(\boldsymbol{n})}\left[(D(G(\boldsymbol{n}|\mathbf{x})))^2\right], \\ \mathcal{L}_G =& \frac{1}{2}\mathbb{E}_{\boldsymbol{n}\sim p_{\text{noise}}(\boldsymbol{n})}\left[(D(G(\boldsymbol{n}|\mathbf{x}))-1)^2\right] + \\ & \lambda\mathbb{E}_{\boldsymbol{n}_D\sim p_{\text{noise}_D}(\boldsymbol{n}_D)}[(||\nabla_{\boldsymbol{n}_D} D(\boldsymbol{n}_D|\mathbf{x})||_2-1)^2].\end{aligned} \tag{2}$$

Specifically, in Formula (2), $p_{\text{noise}_\text{D}}$ is defined to be the distribution of points lying in straight lines between pairs of points sampled from the data distribution p_{data} and the generator distribution p_z. The gradient penalty is introduced as the regularization term in $\mathcal{L}_G$. Here, it constrains the gradient of the Discriminator to have a unit norm, which, in turn, ensures this module satisfies the condition of a 1-Lipschitz function.

Contrastive-based auxiliary loss for device-specific presentation learning. As described in the preceding section, the Generator consists of two blocks with distinct functions: a Shared Feature Extractor and a Device-specific Feature Extractor. While the first block attempts to learn the most representative properties of all the measured data, the distinctive context vectors produced in the second block are the most crucial elements that directly affect the model output.

To facilitate GAMMA in generating the intended context features, a novel contrastive-based loss is introduced. This loss aims to strengthen the connection between the context vectors produced for the same device while also differentiating them from the ones produced for different devices. The formula for this auxiliary loss is as follows.

$$\mathcal{L}_A = -\log \frac{\sum_{u,v\in B}\exp\left(\frac{\mathbf{c}_{i_u}\cdot\mathbf{c}_{i_v}}{\tau}\right)}{\sum_{u,v\in B}\exp\left(\frac{\mathbf{c}_{i_u}\cdot\mathbf{c}_{i_v}}{\tau}\right)+\sum_{u,l\in B;k\neq i}\exp\left(\frac{\mathbf{c}_{i_u}\cdot\mathbf{c}_{k_l}}{\tau}\right)}, \tag{3}$$

where $\mathbf{c}_{i_u}$ and $\mathbf{c}_{i_v}$ are context vectors of the m- and nth time series recorded by device S_i respectively, and $\mathbf{c}_{k_l}$ is the lth time series collected by S_k in a mini-batch B; τ is the temperature hyperparameter.

The final loss function is formed by combining the GAN-based losses $\mathcal{L}_G, \mathcal{L}_D$ and Contrastive-based auxiliary loss $\mathcal{L}_A$ as follows.

$$\mathcal{L} = \mathcal{L}_G + \mathcal{L}_D + \mathcal{L}_A.$$

4. Accuracy evaluation

In this section, experiments are performed to determine how closely the calibrated data relates to the actual data (i.e., data from the reference device). Specifically, this work investigates the following research questions:

RQ1 Does the proposed model outperform the baseline approaches? To answer this question, this study compares the gaps of the data calibrated by the models and the groundtruth obtained by the reference instrument.

RQ2 How efficient is the proposed framework in terms of inference time and memory resources? Concerning this question, this research measures the inference time and memory resources required by each model.

RQ3 To what extent do the hyperparameters affect the proposed model? For this one, experiments are carried out with various values of hyperparameters which facilitates the analysis of the performance change of the proposed data calibration method.

RQ4 How much does each design decision impact the model? To do this, different components of the proposed model are sequentially removed to study the performance change of the method.

In the following, we first describe our experimental methodology in Section 4.1, we then present the settings and evaluation benchmarks in Sections 4.2, and 4.3. The detailed results are described in Sections 4.4, 4.5, 4.6, and 4.7.

4.1. Experimental methodology

We perform the following procedure to evaluate the performance of the proposed method:

- We gather sensory data from our devices and a reference instrument placed at the same location. Information regarding the devices and the reference instrument has been provided in Sections 2.2 and 2.3.
- We then divide the data into three distinct subsets: the training, validation, and testing datasets. Section 4.2 presents comprehensive information regarding the datasets, encompassing details such as the number of data points, the ratio employed for data splitting, and the sampling frequency.
- The training dataset is utilized to train both the proposed model and the comparison benchmarks. While the validation dataset is used to select the best parameters for the models. The details for the comparison benchmarks will be provided in Section 4.3.
- Finally, we compare our proposed model and comparison benchmarks on the testing dataset. The findings are explained in Sections 4.4, 4.5, and 4.6. Furthermore, ablation studies are conducted to assess the individual effects of each module within our proposed technique, as discussed in Section 4.7.

4.2. Experimental settings

Dataset. The dataset comprises sensory air quality and meteorological data collected by one reference instrument and five Fi-Mi devices located in Hanoi, Vietnam (as described before in Section 2.3), from 11 February 2022 to 22 February 2022. During this time interval, a data point is obtained every minute from each device, which is relevant to a sampling rate of 1 : 60 Hz. All of the Fi-Mi devices and the reference device (GRIMM 107) are placed at the same location in a metropolitan region with an industrial zone and dense traffic networks; hence, elevated air pollution indices are regularly recorded. This dataset contains a total of 11030 records, each of which holds several air quality attributes, including $PM_{2.5}$, PM_{10}, humidity, and temperature. The two most crucial air quality measured quantities, $PM_{2.5}$ and PM_{10}, are the calibration objectives for the experiments. Data from the reference instrument GRIMM 107 is considered the standard, while the low-cost sensor data serves as the calibration model's input. The output of this calibration model is expected to resemble the standard as much as possible. To this end, data from each device is divided into a training set, a validation set, and a test set, following the splitting ratio of 0.5 : 0.1 : 0.4, as shown in Table 4. The proposed model, GAMMA, will be verified with the data from five Fi-Mi devices and GRIMM 107 at the same location as in the training phase. This testing data covers the period of February 18, 2022, to February 22, 2022, and contains 4412 data points.

Table 4
Hyper-parameters setting for GAMMA model.

Hyper-parameter	Value
Train Dataset Ratio	0.5
Validate Dataset Ratio	0.1
Test Dataset Ratio	0.4
Hidden Dimension	64
Noise Dimension	32
τ	0.1
First stage epoch - I	100
Second stage epoch - T	500
Discriminator epoch - K	5
Batch size - m	128

Hyper-parameters settings. Details of the hyper-parameter setting are presented in Table 4. Notably, when implementing GAMMA, the dimensions of latent presentations are set as 64, while the noise dimension in latent space is 64. Experiments show that this value set for two hyper-parameters produces the best balance point between model complexity and its accuracy. The optimizer used is AdamW (Loshchilov and Hutter, 2017), and the initial learning rate is 0.001. All the implementation is performed with the help of *Pytorch* framework; the experiments are conducted on an Intel Xeon Silver 4210 2.20 GHz system with 128 GB of main memory and a GeForce RTX 3090 graphic card. To minimize the impact of randomness, a uniform seed is initially established for all experiments, and the results are averaged across three runs for each air quality dataset. All algorithms are evaluated on the same standard (implemented in the same Python environment and tested in the same configuration) to guarantee fairness.

Evaluation metrics. This work leverages the five statistical metrics, namely mean absolute error (MAE), root mean square error (RMSE), mean absolute percentage error (MAPE), median absolute percentage error (MdAPE), and R2 score (R2), to evaluate the performance of the proposed method, i.e., GAMMA, and compare it to existing approaches. The first four metrics (i.e., MAE, RMSE, MAPE, and MdAPE) assess the difference between the calibrated sensory data and the ground-truth data acquired from the reference device. Whereas the last metric (i.e., R2) reflects the degree of correlation between the calibrated sensory data and the ground-truth data. These metrics are often used in evaluating the performance of data calibration methods (Giordano et al., 2021).

4.3. Comparison benchmarks

To evaluate the performance of the proposed model GAMMA, a comparison is drawn against five existing state-of-the-art approaches for time series forecasting, namely Convolutional Recurrent Neural Network (CRNN) (Cirstea et al., 2018), Auto-Encoder Convolutional Recurrent Neural Network (AECRNN) (Cirstea et al., 2018), multitask learning Gated Recurrent Units (MTLGRU) (Ma and Tan, 2020), Multi-series Jointly Forecasting (MSJF) (Zhang et al., 2020), and Shared-private Attention (SPA) (Zhang et al., 2020). The rationale behind employing these forecasting methods lies in their potential to function as calibration models that exhibit superior performance in comparison with existing sensory calibration methods. The utilization of these strategies presents a more challenging comparison to the proposed model, GAMMA.

CRNN (Cirstea et al., 2018) Cirstea et al. proposed a model named CRNN that combines Convolutional Neural Networks (CNNs) with Recurrent Neural Networks (RNNs) to provide p-step ahead forecasting of correlated time series. This approach employs CNNs to extract distinct characteristics from each of the connected multiple time series and then applies RNNs to capture the sequential dependencies of the combined output of CNNs.

AECRNN (Cirstea et al., 2018) In AECRNN, auto-encoders are integrated into CNNs to identify robust features while disregarding outlier characteristics. These auto-encoders also provide additional regularization, aiding in the extraction of representative features from the input time series without overfitting to unique attributes in the training data used for forecasting the target time series.

MTLGRU (Ma and Tan, 2020) Ma et al. introduced a GRU-based multi-task solution. Although GRU was inspired by the LSTM unit, it is considered simpler to calculate and implement. It maintains the LSTM's resilience to the problem of vanishing gradients.

MSJF (Zhang et al., 2020) In this work, Zhang et al. use a private encoding layer for each device and a shared encoding layer for several devices. The private encoding and shared encoding features are concatenated and finally fed into the forecasting module.

SPA (Zhang et al., 2020) Similar to the MSJF, the SPA architecture employs both a shared and a private encoding layer for various devices. The difference between SPA and MSJF is that the feature combination of shared and private encoding is done using an attention module between different devices. The output of this process is subsequently transmitted to the forecasting module.

4.4. Evaluation of calibration accuracy

This section is dedicated to addressing research question **RQ1**. Specifically, it presents an assessment of the similarity between the proposed calibrated data and those obtained from the reference device. Moreover, this work also evaluates the performance of the proposed method and that of the comparison benchmarks listed in Section 4.3.

Deterministic results drawn from GAMMA. Since most state-of-the-art follow the deterministic approach, GAMMA results are also adapted into a single-value set for better comparison. All numbers and statistics reported in the following sections are based on the median of the results generated by GAMMA. Specifically, prediction for each input time series is performed 200 times, each using a random noise drawn from the Gaussian distribution $\mathcal{N}(0, 1)$.

Table 5 demonstrates the detailed performances of proposed approaches and baseline models in terms of calibration accuracy. It can be seen from Table 5 that GAMMA significantly outperforms other baselines in all the evaluation metrics. Specifically, the proposed method improves from 60.19% to 74.09% in terms of the MAE metric compared to the others. For the remaining measurements, the experimental results of GAMMA remain superior to those of contemporary methods. For example, this approach enhances the performance from 62.30% to 79.16% concerning RMSE, whereas this number spans from 60.89% to 64.56% for MdAPE. Concerning the R2 metric, GAMMA improves from 1.15 times to 9.67 times compared with the others. Overall, the proposed model possesses the best metrics' figures, with MAE and MAPE being 2.5 and 13.86 when considering reference data as the groundtruth.

Fig. 6 visualizes the calibration results generated by various methods and the reference series. Intuitively, the more the scatter is aligned to the main diagonal, the more accurate the calibration result is. GAMMA consistently beats all other algorithms on all devices, as demonstrated. GAMMA can nonetheless yield calibration results that are nearly identical to the ground truth, and mostly lie inside the 20% error band, even for devices with the most divergent patterns relative to the reference, such as Fi-Mi #3 and #4 (refer to Fig. A.12 for more details). In contrast, some models, notably CRNN and AECRNN, produce biased data concentrated along the x-axis when calibrating devices Fi-Mi #3 as well as #4. This is the situation in which these models produce predicted values that are smaller than the ground truth. The SPA model's results display the opposite phenomenon for most devices. Furthermore, for each model other than GAMMA, a notable quantity of predictions are observed to fall beyond the 20% error margin, indicating a quantitatively unreliable performance.

The experiments and analyses presented provide significant evidence for the superior efficacy of GAMMA in the calibration of sensory data obtained from diverse sensors.

Probabilistic view produced by GAMMA. Through the training process, GAMMA can learn the reference series' underlying distribution. Fig. 7 demonstrates the reference data regarding the model's prediction results with the confidence bands of 80% and 95%. As shown, the ground truth is almost always contained within the bands. It suggests the robustness of the proposed method towards generalizing the accurate interval of the reference device's measured data.

4.5. Efficiency comparison

In addressing question **RQ2**, experiments are conducted to demonstrate GAMA's ability in terms of resource efficiency. Specifically, a comparison is made between the number of parameters and the inference time of GAMMA and those of the other baseline methods. The results of these experiments are described in Table 6. The inference time in this table is obtained by averaging the entire inference time (s) over all samples of the test set. According to

Table 6, GAMMA reduces more than half the number of parameters compared to the AECRNN model. Besides, the proposed method also operates relatively faster than CRNN and AECRNN, with approximately 0.0018 s per sample versus 0.0032 and 0.0070 s per sample, respectively. Compared to the current lightest model among the baselines, MSJF, GAMMA has the same number of parameters (1.3M), but boosts the performance by up to 113% regarding R2 metrics (see Table 5 for the details). When put together with SPA, the fastest model among all baselines, the proposed method possesses an increase in inference time of 50% whilst witnessing a calibration error reduction of 74.09% regarding the MAE metric. From this experiment, GAMMA has demonstrated its capacity to strike a balance between complexity and precision.

4.6. Hyperparameters sensitivity analysis

Within this section, the sensitivity analysis of hyperparameters regarding the proposed model's performance is described in response to question **RQ3**. According to prior research (Metz et al., 2016; Gong et al., 2019; Chu et al., 2020), training generative adversarial networks is relatively unstable; therefore, it is essential to examine the selection of parameters. Several experiments are conducted to evaluate the sensitivity analysis of different training factors for generative adversarial networks. The investigation encompasses the dimensions of LSTM cells and noise vectors. The subsequent section presents the experimental results concerning these dimensions.

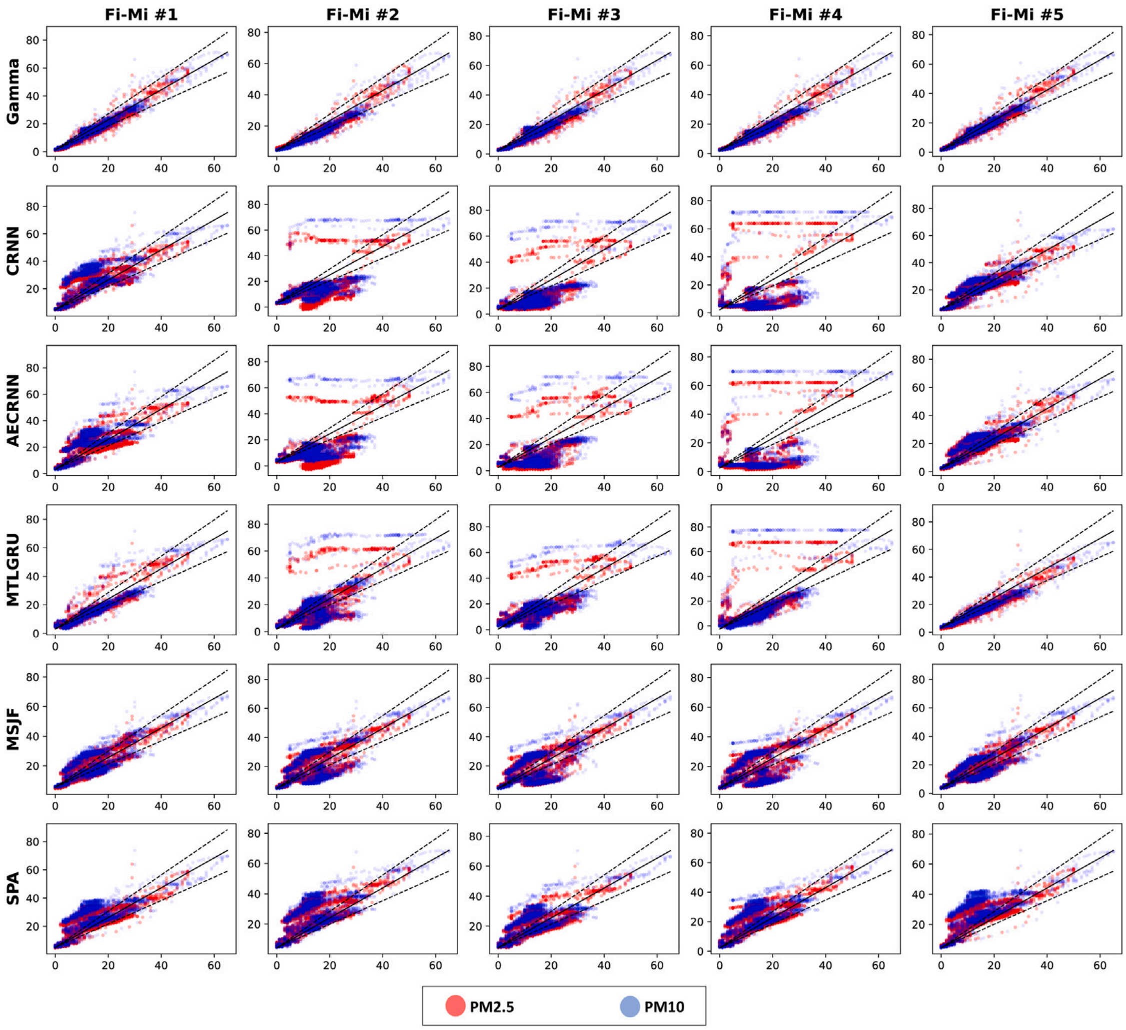


Fig. 6. The scatter plot shows the correlations of GAMMA and different frameworks' calibration results with the reference series.
For all sub-figures, the *x*-axis indicates the ground-truth value; here is the reference series, while the *y*-axis represents the models' calibrated values. GAMMA results produce the scatters which is closest to the perfect calibration line (the diagonal line) in almost all devices, mostly lying in the 20% error band (the two dashed lines).

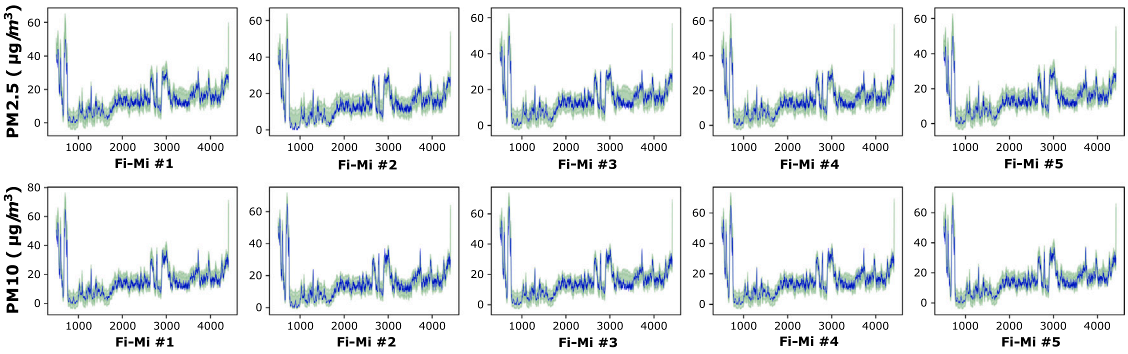


Fig. 7. Probabilistic results produced by GAMMA. For all sub-figures, the *x*-axis indicates the calibration timestep. The blue line represents the groundtruth data measured by the referenced device; the light green bands represent the 80% and 95% confidence bands of GAMMA's predictions.

Dimension of the LSTM cells. The number of LSTM layer's dimensions is altered from 64 to 256 to examine the effects of this number on GAMMA's performance. These variations are observed in three calibration errors (i.e., MAE, RMSE, and MAPE errors). The findings of the experiment are presented in Table 7. As anticipated, all three metrics decline when the dimension of LSTM layers is increased (i.e, from 64 to 128). This phenomenon can be explained by the learning capacity of LSTM cells as their dimensions increase. However, the performance degrades as the LSTM cell size is larger. When the dimension of the LSTM cell is excessively large, it is more likely to face the problem of overfitting. In addition, it is obvious that increasing the LSTM size will increase processing time and necessitate additional memory for storing

Table 5
Average calibration errors achieved by all the methods. In each column, the best results are marked in **bold**.

Method	MAE	RMSE	MAPE	MDAPE	R2
CRNN	9.57	12.36	52.98	49.83	0.087
AECRNN	9.33	11.99	51.70	48.57	0.127
MTLGRU	6.28	9.31	34.71	30.84	0.382
MSJF	8.44	9.81	46.70	44.03	0.432
SPA	9.65	11.84	53.30	57.86	0.169
GAMMA	**2.50**	**3.51**	**13.86**	**12.06**	**0.928**

Table 6
Comparison of the number of parameters and inference time between GAMMA and five baseline models. The best results are marked in **bold**.

Method	Parameters (M)	Inference time (s)
CRNN	2.1M	0.0032
AECRNN	3.9M	0.0070
MTLGRU	3.3M	0.0019
MSJF	**1.3M**	0.0016
SPA	1.5M	**0.0012**
GAMMA (Our)	**1.3M**	0.0018

Table 7
Impact of the dimension of LSTM cells in generative adversarial training.

Dimension of LSTM cells	MAE	RMSE	MAPE
64	3.5	4.59	10.70
128	**2.50**	**3.51**	**13.86**
256	3.45	4.49	9.83

Table 8
Influence of noise vector's dimension in generative adversarial training.

Dimension of noise vector	MAE	RMSE	MAPE
32	3.52	4.58	11.50
64	**2.50**	**3.51**	**13.86**
128	3.56	4.50	10.55

Table 9
Effects of different components on the model's performance. The best results are marked with **bold**.

Model	MAE	RMSE	MAPE	MDAPE	R2
GAMMA-1	6.64	7.36	36.83	46.36	0.555
GAMMA-2	5.57	7.33	30.77	28.93	0.680
GAMMA-3	4.75	5.45	26.32	32.61	0.780
GAMMA-4	4.76	5.70	26.32	27.56	0.811
GAMMA	**2.50**	**3.51**	**13.86**	**12.06**	**0.928**

the model. As a result, a dimension value of 128 is selected for the LSTM layer that strikes a balance between accuracy and efficiency.

Dimension of the noise vectors. Random noise is fed into the GAMMA generator to improve the diversity of the training data, hence boosting the model's ability to learn the data distribution. Intuitively, the dimension of the noise vector determines the degree of distortion imposed on the original data. In this section, the effects of the injected noise are investigated by altering its dimension from 32 to 128 and examining how the calibration performance varies. The results are summarized in Table 8. In general, the calibration errors tend to drop as the dimension of the noise vector grows from 32 to 64, but increase beyond that. This phenomenon can be explained as follows. When the dimension of the noise vector is too modest (e.g., 32), the fraction of noise injected is insignificant; hence, it cannot boost the diversity of the data significantly. When the dimension of the noise vector is adequately extended, the diversity of the data is also enhanced, consequently improving the calibration capability. Nonetheless, if the dimension of the noise vector is too large, the injected noise may dominate the original data and skew its distribution. As a result, the precision of the calibration will deteriorate. According to the experiment results, 64 is a moderate value of the noise vector's dimension.

4.7. *Efficacy of GAMMA's modules*

The proposed model comprises four main components: Shared Feature Extractor, Inter-device Attention, Intra-device Attention, and loss function. Section 3.2 explains the reasoning behind the design of these components. In this section, experiments are conducted to empirically support the hypothesis. In particular, the effectiveness of four experiments involving the progressive simplification of the four modules is evaluated. The details of these experiments are as follows.

- GAMMA-1: The inter-device connection in the Shared LSTM Layer, which was responsible for learning shared characteristics among all devices, is eliminated. Specifically, the Fully Connected Layers that play the role of exchanging information across devices are removed, and only the LSTM layer remains.
- GAMMA-2: To assess the influence of the Inter-device Attention block, experiments are conducted on GAMMA without the incorporation of this module.The context features are now the output of the Shared LSTM Layer. Besides, the identity presentation generated by the Device Identification module is now put directly through the Intra-device Attention Layer.
- GAMMA-3: In this variation, a segment of the Intra-device Attention is excluded to evaluate its influence while preserving the model's overall structure. Precisely, the global attention component is removed, retaining only the self-attention block. This adjustment allows for the assessment of whether the direct integration of device identities contributes to the model's performance enhancement.
- GAMMA-4: To assess the impact of the contrastive-based auxiliary loss on device-specific representation learning, the loss $\mathcal{L}_A$ is excluded while retaining only the GAN-based losses $\mathcal{L}_G$ and $\mathcal{L}_D$.

Table 9 gives information about the results of the aforementioned ablation models utilizing five distinct evaluation metrics. Apparently, the entire proposed framework outperforms the other variants, indicating the positive influence of each architectural decisions. In particular, the full version of GAMMA surpasses GAMMA-1 by 62.36% for the MAE metric. In addition, the MAE value of GAMMA-1 is the highest among all versions, highlighting the advantages of utilizing the shared Feature Extractor. A similar drop in the average MAE at 55.12% and 47.39% can be observed for GAMMA-2 and GAMMA-3, respectively. These reductions highlight the significance of the remaining two components in the proposed method. However, as the statistics provide, it is noticeable that Inter-device Attention has a more significant influence than Intra-device Attention since removing this component results in higher loss values for all metrics. The contrastive-based loss contribution to the complete framework is also significant, as eliminating this loss leads to an increase in MAE error of 90.4%. Overall, all architectural selections and the additional contrastive loss have irreplaceable impacts on the entire GAMMA model.

8. Conclusion

This work focused on calibrating the sensory data, especially PM 2.5 concentration, of low-cost sensor-based air quality monitoring devices. A novel deep learning approach named GAMMA was introduced, which is capable of simultaneous calibration of multiple devices through a single model. The method was implemented and tested on an operational air quality monitoring system situated in Hanoi, Vietnam. Extensive experiments were conducted using real datasets to assess the model's performance and compare it against alternative techniques. The experimental findings revealed that GAMMA strongly surpassed the others in terms of calibrating precision and attained a competitive level of

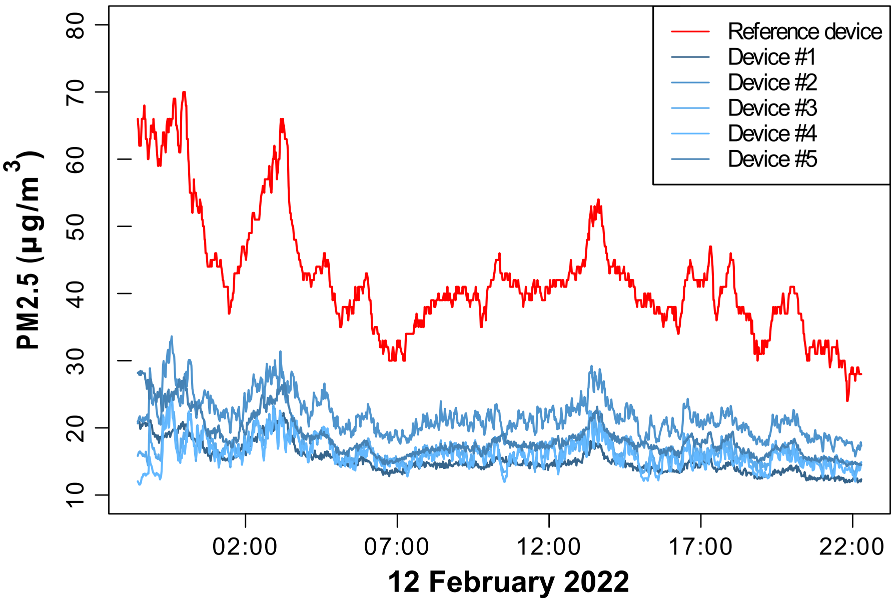


Fig. A.8. Visualization of preprocessed $PM_{2.5}$ gathered by the devices.
The red line reflects the data gathered by the reference device, whereas the blue lines show the data collected by low-cost devices. There are significant discrepancies between data from all devices. Moreover, even inside Fi-Mi devices, the data collected exhibits distinct distributions.

time and resource efficiency. In particular, these findings demonstrate that the use of GAMMA leads to a significant decrease in the calibration error, reducing it from 60.19% to 74.09% as measured by the MAE metric, when compared to the most effective alternative strategy. Moreover, this approach enhances the performance from 62.30% to 79.16% concerning RMSE, whereas this number spans from 60.89% to 64.56% for MdAPE. Concerning the R2 score, GAMMA exhibits a significant improvement ranging from 1.15 times to 9.67 times when compared to the other methods. Among GAMMA's modules, the inter-device connection in the Shared Feature Extractor exhibits the greatest significance, as it possesses the capability to learn shared characteristics across multiple devices.

In the future, it is essential to validate the proposed model within a system incorporating a larger quantity of low-cost sensors, each placed at a distinct location. Furthermore, there is a potential direction to investigate the issue of utilizing point-measurement data obtained from sensors to interpolate and generate area measurements.

CRediT authorship contribution statement

Anh Duy Nguyen: Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft. **Thu Hang Phung:** Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft and Revision, Visualization. **Thuy Dung Nguyen:** Software, Validation, Visualization. **Huy Hieu Pham:** Supervision, Writing – review & editing, Funding acquisition. **Kien Nguyen:** Conceptualization, Supervision, Writing – review & editing, Funding acquisition. **Phi Le Nguyen:** Conceptualization, Methodology, Formal analysis, Writing – review & editing, Supervision, Project administration.

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.

Data availability

Data will be made available on request.

Acknowledgments

This work was funded by Vingroup Joint Stock Company (Vingroup JSC), Vingroup and supported by Vingroup Innovation Foundation (VINIF) under project code VINIF.2020.DA09. This research is funded by Hanoi University of Science and Technology (HUST) under grant number T2023-PC-028.

Appendix A. Appendix

Fig. A.11 shows the histograms of the preprocessed $PM_{2.5}$ and PM_{10} collected by Fi-Mi devices and the reference instrument. In general, the data collected by Fi-Mi devices tends to focus on a limited spectrum of values, while the data obtained by GRIMM 107 devices exhibits a broader range of values. Fi-Mi devices witness a smaller range of values, which is approximately 10 to 40 $\mu g/m^3$ compared to that of GRIMM 107, which ranges up to 105 $\mu g/m^3$ for $PM_{2.5}$ and 115 $\mu g/m^3$ for PM_{10}. Furthermore, the frequencies of data gathered by Fi-Mi devices have a significantly higher value than those collected by GRIMM. Specifically, the most notable frequency of Fi-Mi's data is approximately five times that of GRIMM 107.

A global trend reveals that the overlapping values between the Fi-Mi device and the reference instrument constitute only a minor fraction, often around one-fourth of the total data range provided by the reference instrument. Furthermore, the fact that the majority of data supplied by Fi-Mi devices is restricted to a short range prevents the generation of high-range or even extremely large air quality quantity values. These constraints make the problem of calibrating Fi-Mi devices more challenging. Fig. A.12 shows the Pearson correlation coefficient (PCC) of the data gathered by Fi-Mi devices and those of the reference instrument, which can then be used to investigate the correlation between them. The color close to red signifies average correlations, while the blue-like color indicates significant correlations. As can be seen from Fig. A.12, correlation levels between the reference instrument and the first or fifth Fi-Mi device are more significant compared to the remaining devices for both air quality indices. The Fi-Mi #4 shows a weaker relationship to the GRIMM 107 with a correlation score of approximately 0.6. Additionally, this device has a relatively poor relationship with other Fi-Mi devices. This phenomenon imposes difficulty in developing a multi-task calibrating model for all devices (see Fig. A.9).

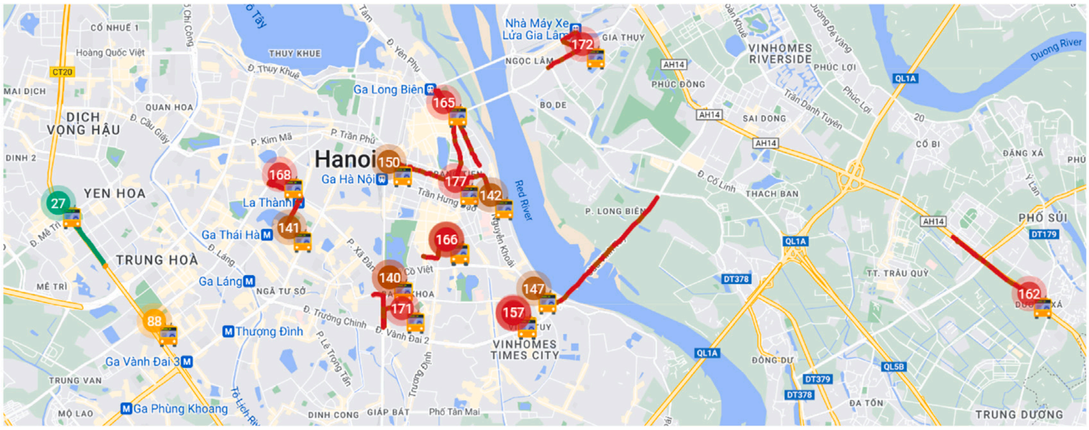


Fig. A.9. Air quality map when implementing Fi-Mi devices on the Hanoi bus system.
The map shows the current location of sensors, along with circles with AQI data at that point. When a bus equipped with a moving sensor moves, the website interface (i.e., http://demo.fi-mi.vn/app) will leave a trace of the route the bus has traveled with colors corresponding to the AQI status at the measurement point.

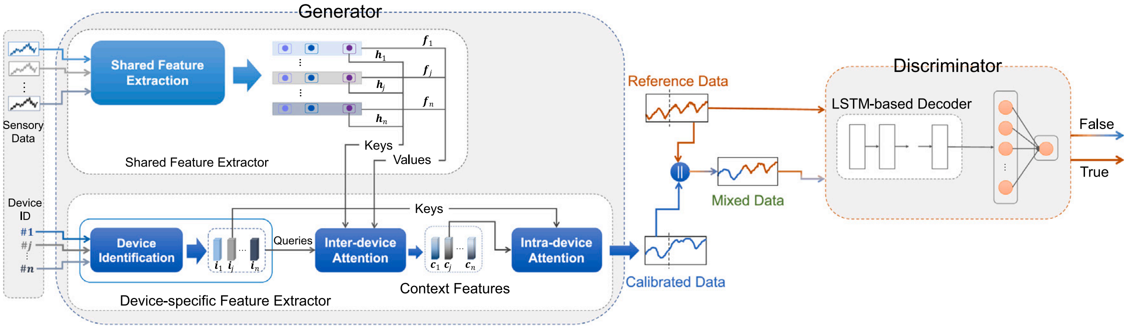


Fig. A.10. Overview of GAMMA.
GAMMA calibrates multiple sensor-based air quality monitoring devices concurrently using a single model. It employs the multitask learning paradigm to handle numerous calibration tasks simultaneously and the GAN approach to achieve precise calibration. The Generator comprises two primary modules: a *Shared Feature Extractor* for extracting representative characteristics shared by all sensors and a *Device-specific Feature Extractor* responsible for capturing the unique properties of each device.

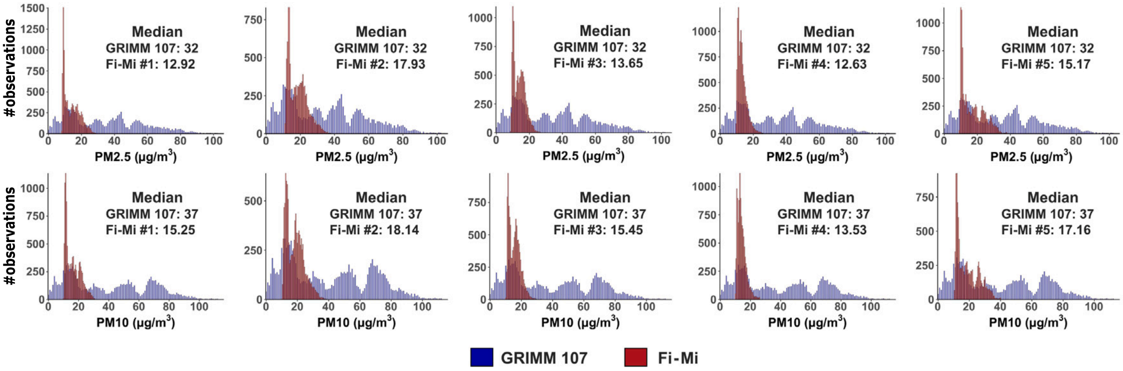


Fig. A.11. Histograms of $PM_{2.5}$ and PM_{10} for each pair of a Fi-Mi device and GRIMM 107.
There is a considerable difference between the number of observations (#observations) of the preprocessed sensory data collected by Fi-Mi devices (red bars) and the reference data. In contrast to the tall and narrow shape of the Fi-Mi data, the reference data (blue bars) are short and wide.

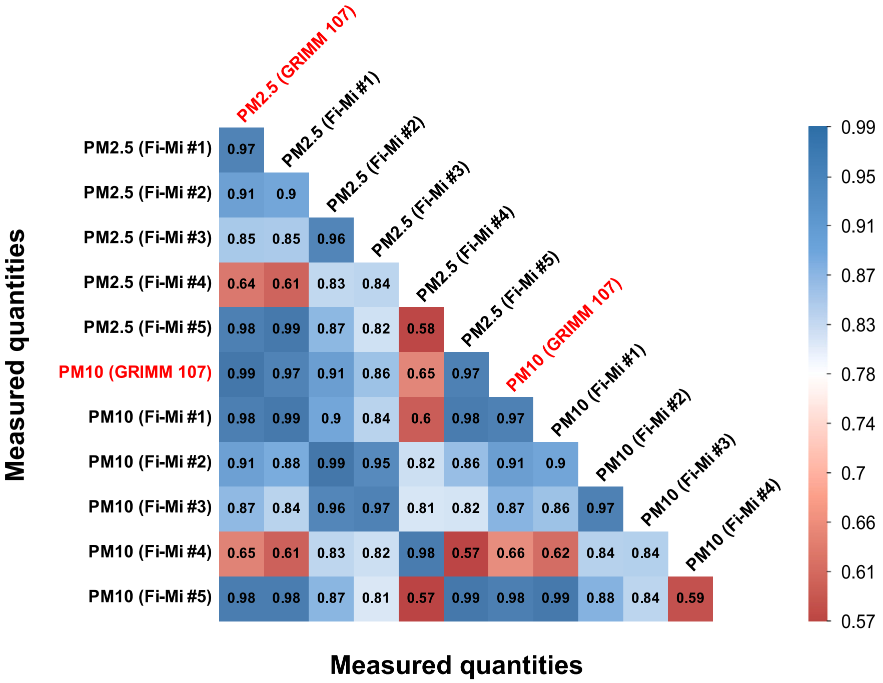


Fig. A.12. Pearson correlation coefficient of $PM_{2.5}$ and PM_{10} gathered by Fi-Mi devices and GRIMM 107.
The data from Fi-Mi devices correlates inconsistently with the reference data. Some devices, such as Fi-Mi #2, have a strong correlation with the reference, while others, such as Fi-Mi #4, have a very weak association.

Table A.10
Error percentage difference when evaluating baseline models with testing data from two distinct devices, whereas training with one device only.
Experiments were conducted using Fi-Mi #5 for training and assessment, along with data from its most different device, Fi-Mi #4. This table illustrates the difference in percentage of test metrics between these two devices' data. Hence, the utilization of a model trained with data from a specific device is not transferable to data obtained from different devices.

Method	MAE	RMSE	MAPE
CRNN	13.98	6.48	13.98
AECRNN	8.68	5.35	8.68
MTLGRU	31.30	21.23	31.30
MSJF	35.66	23.92	35.66
SPA	19.98	15.92	19.98
Average	21.92	14.58	21.92

References

Andersen, Z.J., Bønnelykke, K., Hvidberg, M., Jensen, S.S., Ketzel, M., Loft, S., Sørensen, M., Tjønneland, A., Overvad, K., Raaschou-Nielsen, O., 2012a. Long-term exposure to air pollution and asthma hospitalisations in older adults: a cohort study. Thorax 67 (1), 6–11.

Andersen, Z.J., Kristiansen, L.C., Andersen, K.K., Olsen, T.S., Hvidberg, M., Jensen, S.S., Ketzel, M., Loft, S., Sørensen, M., Tjønneland, A., et al., 2012b. Stroke and long-term exposure to outdoor air pollution from nitrogen dioxide: a cohort study. Stroke 43 (2), 320–325.

Anon, 2023. Dataset. github.com/anhduy0911/FimiCalibIdea/blob/multi_attention/Data/fimi_resample/envitus_fimi_overlapped.csv.

Aula, K., Lagerspetz, E., Nurmi, P., Tarkoma, S., 2022. Evaluation of low-cost air quality sensor calibration models. ACM Trans. Sensor Netw. 18 (4), 1–32.

Bai, L., Shin, S., Burnett, R.T., Kwong, J.C., Hystad, P., van Donkelaar, A., Goldberg, M.S., Lavigne, E., Weichenthal, S., Martin, R.V., et al., 2020. Exposure to ambient air pollution and the incidence of lung cancer and breast cancer in the ontario population health and environment cohort. Int. J. Cancer 146 (9), 2450–2459.

Baron, Y.M., 2022. Are there medium to short-term multifaceted effects of the airborne pollutant PM2.5 determining the emergence of SARS-CoV-2 variants? Med. Hypotheses 158, 110718. http://dx.doi.org/10.1016/j.mehy.2021.110718.

Bhatnagar, S., Chang, W., Kim, S., Wang, J., 2022. Computer model calibration with time series data using deep learning and quantile regression. SIAM/ASA J. Uncertain. Quantif. 10, 1–26.

Carratù, M., Ferro, M., Paciello, V., Sommella, P., Lundgren, J., O'Nils, M., 2020. Wireless sensor network calibration for PM10 measurement. In: 2020 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications. CIVEMSA, IEEE, pp. 1–6.

Chattopadhyay, A., Huertas, A., Rebeiro-Hargrave, A., Fung, P.L., Varjonen, S., Hieta, T., Tarkoma, S., Petäjä, T., 2022. Low-cost formaldehyde sensor evaluation and calibration in a controlled environment. IEEE Sens. J. 22 (12), 11791–11802. http://dx.doi.org/10.1109/JSEN.2022.3172864.

Christakis, I., Hloupis, G., Stavrakas, I., Tsakiridis, O., 2020. Low cost sensor implementation and evaluation for measuring NO2 and O3 pollutants. In: 2020 9th International Conference on Modern Circuits and Systems Technologies. MOCAST, IEEE, pp. 1–4.

Christakis, I., Hloupis, G., Tsakiridis, O., Stavrakas, I., 2022. Integrated open source air quality monitoring platform. In: 2022 11th International Conference on Modern Circuits and Systems Technologies. MOCAST, IEEE, pp. 1–4.

Chu, C., Minami, K., Fukumizu, K., 2020. Smoothness and stability in gans. http://dx.doi.org/10.48550/arXiv.2002.04185.

Cirstea, R.G., Micu, D.V., Muresan, G.M., Guo, C., Yang, B., 2018. Correlated time series forecasting using multi-task deep neural networks. In: Proceedings of the 27th Acm International Conference on Information and Knowledge Management. pp. 1527–1530.

Cordero, J.M., Borge, R., Narros, A., 2018. Using statistical methods to carry out in field calibrations of low cost air quality sensors. Sensors Actuators B 267, 245–254. http://dx.doi.org/10.1016/j.snb.2018.04.021.

De Vito, S., Esposito, E., Castell, N., Schneider, P., Bartonova, A., 2020. On the robustness of field calibration for smart air quality monitors. Sensors Actuators B 310, 127869. http://dx.doi.org/10.1016/j.snb.2020.127869.

deSouza, P., Anjomshoaa, A., Duarte, F., Kahn, R., Kumar, P., Ratti, C., 2020. Air quality monitoring using mobile low-cost sensors mounted on trash-trucks: Methods development and lessons learned. Sustainable Cities Soc. 60, 102239. http://dx.doi.org/10.1016/j.scs.2020.102239.

Dubey, R., Patra, A.K., Joshi, J., Blankenberg, D., Kolluru, S.S.R., Madhu, B., Raval, S., 2022. Evaluation of low-cost particulate matter sensors OPC N2 and PM nova for aerosol monitoring. Atmospheric Pollut. Res. 13 (3), 101335.

Duong, L., Cohn, T., Bird, S., Cook, P., 2015. Low resource dependency parsing: Cross-lingual parameter sharing in a neural network parser. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers). pp. 845–850.

Gehring, U., Wijga, A.H., Brauer, M., Fischer, P., de Jongste, J.C., Kerkhof, M., Oldenwening, M., Smit, H.A., Brunekreef, B., 2010. Traffic-related air pollution and the development of asthma and allergies during the first 8 years of life. Am. J. Respir. Crit. Care Med. 181 (6), 596–603.

Giordano, M.R., Malings, C., Pandis, S.N., Presto, A.A., McNeill, V., Westervelt, D.M., Beekmann, M., Subramanian, R., 2021. From low-cost sensors to high-quality data: A summary of challenges and best practices for effectively calibrating low-cost particulate matter mass sensors. J. Aerosol Sci. 158, 105833.

Goldberg, M., 2008. A systematic review of the relation between long-term exposure to ambient air pollution and chronic diseases. Rev. Environ. Health 23 (4), 243–298.

Gong, X., Chang, S., Jiang, Y., Wang, Z., 2019. Autogan: Neural architecture search for generative adversarial networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3224–3234.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press, [Online]. URL: http://www.deeplearningbook.org. (Accessed 15 March 2022).

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial nets. Adv. Neural Inf. Process. Syst. 27.

GRIMMTechnologies Inc., 2005. GRIMM EDM 107. [Online]. URL: https://studylib.net/doc/7683874/operating-manual-107---grimm-technologies--inc. (Accessed 25 October 2022).

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A., 2017. Improved training of wasserstein GANs. http://dx.doi.org/10.48550/arXiv.1704.00028.

Hashmy, Y., Khan, Z.U., Ilyas, F., Hafiz, R., Younis, U., Tauqeer, T., 2023. Modular air quality calibration and forecasting method for low-cost sensor nodes. IEEE Sens. J. 23 (4), 4193–4203. http://dx.doi.org/10.1109/JSEN.2023.3233982.

Hofman, J., Nikolaou, M., Shantharam, S.P., Stroobants, C., Weijs, S., La Manna, V.P., 2022. Distant calibration of low-cost PM and NO2 sensors; evidence from multiple sensor testbeds. Atmospheric Pollut. Res. 13 (1), 101246. http://dx.doi.org/10.1016/j.apr.2021.101246.

Hu, K., Rahman, A., Bhrugubanda, H., Sivaraman, V., 2017. HazeEst: Machine learning based metropolitan air pollution estimation from fixed and mobile sensors. IEEE Sens. J. 17 (11), 3517–3525. http://dx.doi.org/10.1109/JSEN.2017.2690975.

Hunter, J.S., 1986. The exponentially weighted moving average. J. Qual. Technol. 18 (4), 203–210.

Idrees, Z., Zheng, L., 2020. Low cost air pollution monitoring systems: A review of protocols and enabling technologies. J. Ind. Inf. Integr. 17, 100123. http://dx.doi.org/10.1016/j.jii.2019.100123.

Johnson, N.E., Bonczak, B., Kontokosta, C.E., 2018. Using a gradient boosting model to improve the performance of low-cost aerosol monitors in a dense, heterogeneous urban environment. Atmospheric Environ. 184, 9–16.

Kulmala, M., 2018. Build a global earth observatory. Nature 553 (7686), 21–23.

Kumar, V., Sahu, M., 2021. Evaluation of nine machine learning regression algorithms for calibration of low-cost PM2.5 sensor. J. Aerosol Sci. 157, 105809. http://dx.doi.org/10.1016/j.jaerosci.2021.105809.

Lagerspetz, E., Motlagh, N.H., Arbayani Zaidan, M., Fung, P.L., Mineraud, J., Varjonen, S., Siekkinen, M., Nurmi, P., Matsumi, Y., Tarkoma, S., Hussein, T., 2019. MegaSense: Feasibility of low-cost sensors for pollution hot-spot detection. In: 2019 IEEE 17th International Conference on Industrial Informatics, Vol. 1. INDIN, pp. 1083–1090. http://dx.doi.org/10.1109/INDIN41052.2019.8971963.

Li, G., Wu, Z., Liu, N., Liu, X., Wang, Y., Zhang, L., 2023. A variational Bayesian blind calibration approach for air quality sensor deployments. IEEE Sens. J. 23 (7), 7129–7141. http://dx.doi.org/10.1109/JSEN.2022.3212009.

Liu, X., Jayaratne, R., Thai, P., Kuhn, T., Zing, I., Christensen, B., Lamont, R., Dunbabin, M., Zhu, S., Gao, J., Wainwright, D., Neale, D., Kan, R., Kirkwood, J., Morawska, L., 2020. Low-cost sensors as an alternative for long-term air quality monitoring. Environ. Res. 185, 109438. http://dx.doi.org/10.1016/j.envres.2020.109438.

Liu, B., Jin, Y., Li, C., 2021a. Analysis and prediction of air quality in nanjing from autumn 2018 to summer 2019 using PCR–SVR–ARMA combined model. Sci. Rep. 11 (1), 1–14.

Liu, B., Jin, Y., Xu, D., Wang, Y., Li, C., 2021b. A data calibration method for micro air quality detectors based on a LASSO regression and NARX neural network combined model. Sci. Rep. 11 (1), 1–12.

Liu, N., Wu, Z., Li, G., Liu, X., Wang, Y., Zhang, L., 2022. MAIC: Metalearning-based adaptive in-field calibration for IoT air quality monitoring system. IEEE Internet Things J. 9 (17), 15928–15941.

Loshchilov, I., Hutter, F., 2017. Decoupled weight decay regularization. http://dx.doi.org/10.48550/arXiv.1711.05101.

Loy-Benitez, J., Heo, S., Yoo, C., 2020. Imputing missing indoor air quality data via variational convolutional autoencoders: Implications for ventilation management of subway metro systems. Build. Environ. 182, 107135.

Ma, T., Tan, Y., 2020. Multiple stock time series jointly forecasting with multi-task learning. In: 2020 International Joint Conference on Neural Networks. IJCNN, IEEE, pp. 1–8.

Maag, B., Zhou, Z., Thiele, L., 2018. A survey on sensor calibration in air pollution monitoring deployments. IEEE Internet Things J. 5 (6), 4857–4870. http://dx.doi.org/10.1109/JIOT.2018.2853660.

Mao, X., Li, Q., Xie, H., Lau, R.K., Wang, Z., Smolley, S., 2017. Least squares generative adversarial networks. In: 2017 IEEE International Conference on Computer Vision. ICCV, IEEE Computer Society, Los Alamitos, CA, USA, pp. 2813–2821, URL: https://doi.ieeecomputersociety.org/10.1109/ICCV.2017.304.

Metz, L., Poole, B., Pfau, D., Sohl-Dickstein, J., 2016. Unrolled generative adversarial networks. http://dx.doi.org/10.48550/arXiv.1611.02163.

Migos, T., Christakis, I., Moutzouris, K., Stavrakas, I., 2019. On the evaluation of low-cost PM sensors for air quality estimation. In: 2019 8th International Conference on Modern Circuits and Systems Technologies. MOCAST, IEEE, pp. 1–4.

Mirza, M., Osindero, S., 2014. Conditional generative adversarial nets. http://dx.doi.org/10.48550/arXiv.1411.1784.

Motlagh, N.H., Lagerspetz, E., Nurmi, P., Li, X., Varjonen, S., Mineraud, J., Siekkinen, M., Rebeiro-Hargrave, A., Hussein, T., Petaja, T., Kulmala, M., Tarkoma, S., 2020. Toward massive scale air quality monitoring. IEEE Commun. Mag. 58 (2), 54–59. http://dx.doi.org/10.1109/MCOM.001.1900515.

Moumtzidou, A., Papadopoulos, S., Vrochidis, S., Kompatsiaris, I., Kourtidis, K., Hloupis, G., Stavrakas, I., Papachristopoulou, K., Keratidis, C., 2016. Towards air quality estimation using collected multimodal environmental data. In: Collective Online Platforms for Financial and Environmental Awareness: First International Workshop on the Internet for Financial Collective Awareness and Intelligence, IFIN 2016 and First International Workshop on Internet and Social Media for Environmental Monitoring, ISEM 2016. Florence, Italy, September 12, 2016, Revised Selected Papers 1, Springer, pp. 147–156.

Patton, A., Datta, A., Zamora, M.L., Buehler, C., Xiong, F., Gentner, D.R., Koehler, K., 2022. Non-linear probabilistic calibration of low-cost environmental air pollution sensor networks for neighborhood level spatiotemporal exposure assessment. J. Expo. Sci. Environ. Epidemiol. 32 (6), 908–916.

Sakhvidi, M.J.Z., Lequy, E., Goldberg, M., Jacquemin, B., 2020. Air pollution exposure and bladder, kidney and urinary tract cancer risk: A systematic review. Environ. Pollut. 267, 115328.

Saukh, O., Hasenfratz, D., Thiele, L., 2015. Reducing multi-hop calibration errors in large-scale mobile sensor networks. In: IPSN '15. Association for Computing Machinery, New York, NY, USA, pp. 274–285. http://dx.doi.org/10.1145/2737095.2737113.

Sayahi, T., Butterfield, A., Kelly, K., 2019. Long-term field evaluation of the plantower PMS low-cost particulate matter sensors. Environ. Pollut. 245, 932–940.

Tancev, G., Toro, F.G., 2022. Variational Bayesian calibration of low-cost gas sensor systems in air quality monitoring. Measurement: Sensors 19, 100365. http://dx.doi.org/10.1016/j.measen.2021.100365.

Wang, W.C.V., Lung, S.C.C., Liu, C.H., 2020. Application of machine learning for the in-field correction of a PM2. 5 low-cost sensor network. Sensors 20 (17), 5002.

Wang, Y., Yang, A., Chen, X., Wang, P., Wang, Y., Yang, H., 2017. A deep learning approach for blind drift calibration of sensor networks. IEEE Sens. J. 17, 4158–4171.

WHO, 2016. Ambient Air Pollution: A Global Assessment of Exposure and Burden of Disease. World Health Organization, URL: https://apps.who.int/iris/handle/10665/250141.

WHO, 2022. Ambient (outdoor) air pollution. [Online]. URL: https://www.who.int/news-room/fact-sheets/detail/ambient-(outdoor)-air-quality-and-health. (Accessed 18 December 2022).

Yadav, K., Arora, V., Kumar, M., Tripathi, S.N., Motghare, V.M., Rajput, K.A., 2022. Few-shot calibration of low-cost air pollution (PM _{2.5}) sensors using meta learning. IEEE Sensors Lett. 6 (5), 1–4.

Yang, Y., Hospedales, T.M., 2016. Trace norm regularised deep multi-task learning. http://dx.doi.org/10.48550/arXiv.1606.04038.

Yu, H., Li, Q., Geng, Y.a., Zhang, Y., Wei, Z., 2020a. Airnet: A calibration model for low-cost air monitoring sensors using dual sequence encoder networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, No. 01. pp. 1129–1136.

Yu, H., Li, Q., Wang, R., Chen, Z., Zhang, Y., Geng, Y.a., Zhang, L., Cui, H., Zhang, K., 2020b. A deep calibration method for low-cost air monitoring sensors with multilevel sequence modeling. IEEE Trans. Instrum. Meas. 69 (9), 7167–7179.

Zhang, K., Wu, L., Zhu, Z., Deng, J., 2020. A multitask learning model for traffic flow and speed forecasting. IEEE Access 8, 80707–80715.

Zhang, P., Yang, L., Ma, W., Wang, N., Wen, F., Liu, Q., 2022. Spatiotemporal estimation of the PM2.5 concentration and human health risks combining the three-dimensional landscape pattern index and machine learning methods to optimize land use regression modeling in Shaanxi, China. Environ. Res. 208, 112759. http://dx.doi.org/10.1016/j.envres.2022.112759.

Zimmerman, N., Presto, A.A., Kumar, S.P., Gu, J., Hauryliuk, A., Robinson, E.S., Robinson, A.L., Subramanian, R., 2018. A machine learning calibration model using random forests to improve sensor performance for lower-cost air quality monitoring. Atmos. Meas. Tech. 11 (1), 291–313.