Soc-IoT: A Proof-of-Concept for Citizen-Centric Environmental Monitoring

Sachit Mahajan

***Abstract*— Cities around the world are struggling with environmental pollution. The conventional monitoring approaches are not effective for undertaking large-scale environmental monitoring due to logistical and cost-related issues. The availability of low-cost and low-power Internet of Things (IoT) devices has proved to be an effective alternative to monitor the ambient environment. Such systems have opened up environment monitoring opportunities to researchers and citizens while simultaneously confronting them with challenges like sensor accuracy, accumulation of large data sets, and data analysis, which itself is a formidable task that requires extensive computational resources and technical expertise. To address this challenge, a social, open-source, and citizen-centric IoT (Soc-IoT) framework is proposed that combines tools for real-time environmental sensing with an intuitive data analysis and visualization application. Soc-IoT has two main components: (1) CoSense Unit – a resource-efficient, portable and modular environment monitoring device intended for citizen sensing and complementing official environment monitoring infrastructure, and (2) *exploreR* – an intuitive cross-platform data analysis and visualization application that offers a comprehensive set of tools for systematic analysis of sensor data without any coding requirement. Developed as a proof-of-concept framework to monitor the environment at scale, Soc-IoT aims to promote environmental resilience and open innovation by reducing technological barriers.**

***Index Terms*— Smart City, Air Quality Monitoring, PM2.5, Data Analysis**

I. Introduction and Background

Over the past years, the world has seen massive growth in urbanization at regional and national levels. The blind pursuit of urban and economic growth has also intensified environmental degradation [1]. Activities like excessive use of fossil fuels for energy production and deforestation to create more urban spaces are already contributing to the degradation of air quality. The effect is not only limited to developing or under-developed countries, but even high-income countries are getting adversely affected by it [2]. According to a report by World Health Organization [3], indoor and outdoor air pollution exposure is strongly linked to heart and cardiovascular diseases. Among different pollutants, particulate matter (PM) is known to be more dangerous for human health as compared to gaseous components [4]. While there have been numerous efforts by governments and environmental protection agencies to combat the threat of air pollution, there has been limited success in a reduction in the levels of pollutants like PM. This has been mainly due to the limited availability of accurate and fine-grained air quality data to create effective policies. The official monitoring networks used in most of the countries around the world comprise a limited number of fixed monitoring stations. They are accurate but only covered a limited geographical area [5]. Due to the expensive and bulky nature of such stations, it is not logistically possible to do a mass deployment of such stations.

With the emergence of smart cities and the idea of environment monitoring using Wireless Sensor Networks (WSN), it has become easier than ever to perform large-scale environmental monitoring [6], [7]. The availability and use of low-cost Internet of Things (IoT) based sensor systems have already changed the technological paradigm by introducing new types of intelligence that connect people and the environment and promote interaction between them. Such technologies are transforming smart cities into sustainable smart cities by allowing citizens to engage with smart city ecosystems using digital means [8]. The IoT systems enable air quality monitoring at a finer spatio-temporal scale by using a network of monitoring devices around the city. These devices provide real-time air quality data that can be useful for understanding the ambient environment and assisting decision-makers in making better policies for pollution control. There have been several examples of how low-cost environmental monitoring solutions have been implemented around the world to raise air pollution awareness [9]–[11], create air pollution data sets [12]–[14], promote citizen participation in air quality monitoring [15]–[18], and create applications for data-informed decision making [19]–[22]. The IoT technology has been used to work towards creating more inclusive and resilient cities that can advance the knowledge and information-sharing capabilities of citizens with crowdsourcing and crowd-sharing platforms. The valuable data that is crowdsourced using IoT directly impacts the location-based services that are offered to the citizens. The data is instrumental in creating advanced air quality data analysis frameworks [23], [24], $PM_{2.5}$ forecasting systems [25], [26], and ecosystems for smart environment

"The author acknowledges support through the project "CoCi: Co-Evolving City Life", which has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme under grant agreement No. 833168." *(Corresponding author: S. Mahajan).* Sachit Mahajan is with the Computational Social Science, ETH Zurich, Zurich 8092, Switzerland (e-mail: sachit.mahajan@gess.ethz.ch).

governance [27].

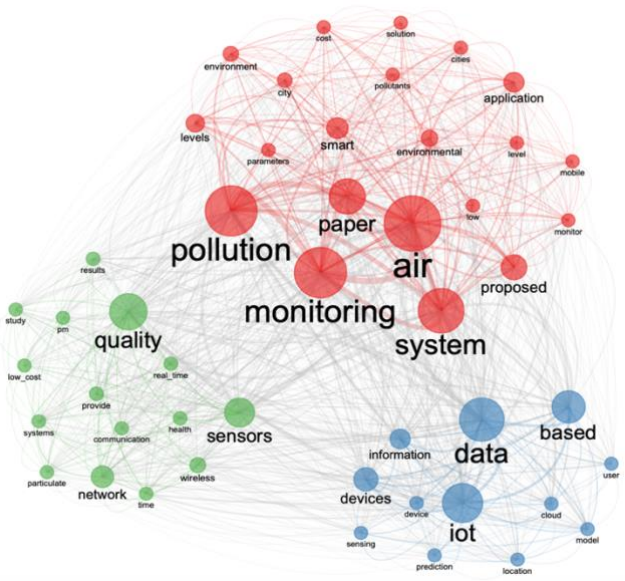

Fig.1 Network visualization of frequently occurring terms within the existing literature related to keywords "Internet of Things" and "Air pollution monitoring".

Bibliometric Analysis: Bibliometric analysis is an efficient method to understand research trends and scholarly networks in different disciplines [28]. To understand how the keywords like "Internet of Things" and "Air pollution monitoring" have been used within the existing literature and in what context, quantitative bibliometric analysis and knowledge mapping approaches were used. The term co-occurrence method was used to find the keywords that are discussed more frequently together. To perform the analysis, first, a search query was created that searched all the papers indexed in the Web of Science[1] database containing the topics "Internet of Things" AND "Air pollution monitoring". The search query resulted in 65 papers. The data from those 65 papers were used to create the keyword co-occurrence network graph, shown in Fig. 1. *bibliometrix* package of R was used to perform the network analysis [29]. For creating the network, 50 highly occurring terms were chosen as the nodes. Each node had at least two edges. To detect the communities in the network, the Louvain method of community detection was implemented [30]. It is a clustering algorithm that is based on the greedy approach to modularity optimization. In the beginning, every node is assigned to a unique cluster. This is followed by placing each node is into another cluster to make the network more modular. The process is repeated several times until there is no further scope for improving the network modularity. It can be observed in Fig. 1 that there are three key research clusters. The largest cluster is mainly focused on air pollution monitoring systems, the environment, and smart cities. Between the other two clusters, one focuses on the IoT devices, data, and information while the other is more centered around PM, networks, and sensors. Despite a strong focus of existing research on IoT systems, environment, data, and cities, surprisingly there was no mention of keywords like 'citizens', 'community', 'open-source', or 'sustainability'. There is a clear gap when it comes to bridging the IoT, air quality monitoring, citizen participation, and open-source solutions. This reinforces the relevance of this study that aims at creating a proof-of-concept framework for environmental monitoring citizen-centric, open-source, and sustainable.

Motivation: While the use of low-cost sensors has improved the air quality data availability and access, several challenges still need to be addressed. Data quality and accuracy of low-cost sensors remain one of the key challenges [31]–[33]. It is been widely discussed how an IoT application could be considered useless due to poor sensor data quality [34]. This not only restricts the potential use of IoT data for various applications but also creates an environment where the acceptability of citizen-generated data reduces to lack of accuracy. This makes it imperative that the hardware and software components of the IoT framework can successfully handle the sensor data with minimum errors and missing data. It has also been observed that sometimes IoT systems are designed in less human-centric ways. This can be related to highly automated sensors, black-box algorithms, data accessibility, and complex data analysis tools. The lack of value-sensitive design often results in user disempowerment followed by disengagement [8], [35]. This is a critical concern as the majority of citizen science air quality monitoring projects depend on volunteers who are investing their time and resources. For example, in many citizen science air quality monitoring projects, the citizens rely on experts to do the data analysis and interpretation. Though scientific expertise is needed to analyze data but creating opportunities for citizens to do data analysis and interpretation allows bridging the gap between experts and non-experts. It also fosters a sense of collaboration and trust that are important for successfully doing citizen science. Another pressing issue is the consideration of sustainability factors for the design, development, and implementation of low-cost sensor systems. Based on a study [36], it was found that there is limited literature when it comes to understanding the long-term sustainability of low-cost sensor solutions for environmental monitoring. The predominant focus of most of the studies has been on data collection and analysis. This could be partly because most of these sensor studies are conducted in regions that have significant resources and infrastructure [36]. This paper addresses these challenges by outlining the design, implementation, and potential impact of a social, open-source, and citizen-centric IoT (Soc-IoT, pronounced as '*Society'*) framework. Fig. 2 shows the overview of the proposed framework. Soc-IoT is proposed as an environment monitoring framework that is a combination of two components. *CoSense Unit*, a modular and open-source environment sensing device that can provide consistent and accurate air quality data. It has been extensively tested in a real-world environment as well as evaluated by co-locating it with the official environment

[1] https://www.webofscience.com/wos/woscc/basic-search

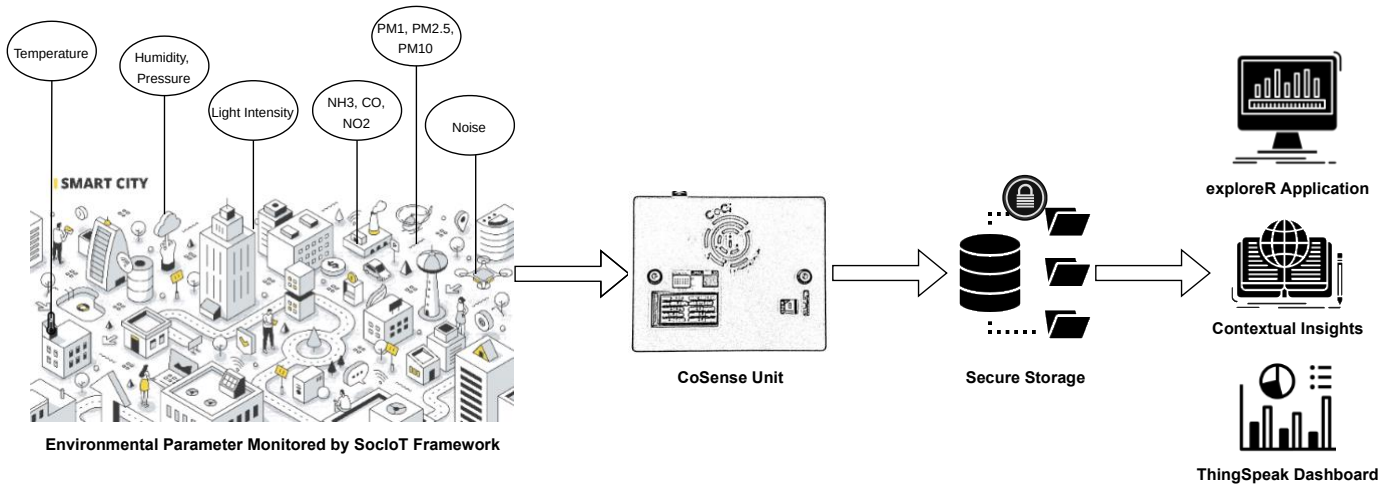

Fig. 2 Overview of the proposed framework.

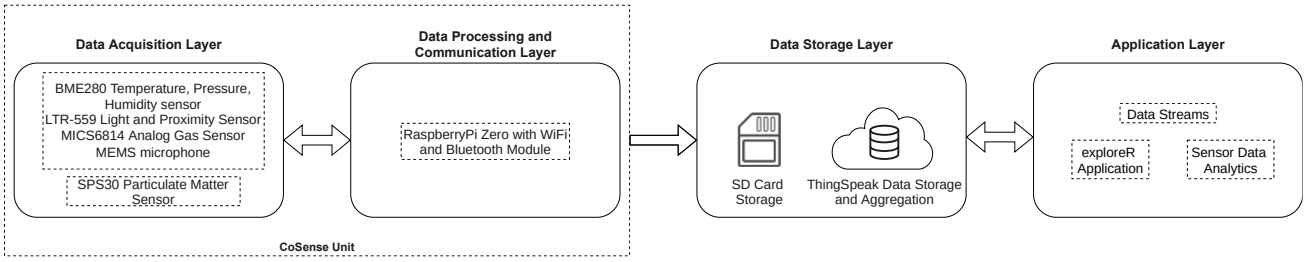

Fig. 3 Soc-IoT system architecture.

the monitoring station in Switzerland. The sustainability aspect of the *CoSense Unit* is also investigated by examining the carbon footprint and energy consumption of these low-cost devices. The second core component is the framework is the *exploreR* that is an open-source RShiny[2] based data analysis and visualization application. The application is designed to reduce the technological barriers especially related to programming and allow citizens as well as experts to analyze and interpret sensor data in a meaningful way. The complete framework is designed to create an innovative ecosystem that enables collaboration, sustainable practices, and inclusion to address the pressing issue of collaborative environmental sensing.

The paper is organized as follows: Section 2 discusses the methodology behind the design and implementation of the Soc-IoT framework, describing the hardware and software components. Section 3 discusses the sensor validation and evaluation. The conclusions are reported in Section 4.

II. Methods

This section describes the methodology behind the design of the proposed Soc-IoT framework. The following paragraphs provide a detailed overview of the system architecture, sensor prototype, and data analysis application.

A. System Architecture

The Soc-IoT framework[3] is based on the principle of open-source hardware and software. Fig. 3 shows the system architecture of the proposed framework. It comprises four major components:

- Data Acquisition Layer: This layer consists of the sensors that are responsible for sensing the environmental variables monitored by the CoSense Unit. The current version of the CoSense Unit consists of a Sensirion SPS 30 PM sensor that can sense PM_1, $PM_{2.5}$, and PM10. The Enviro+ board for Raspberry Pi is used to monitor temperature, pressure, humidity, light intensity, noise, and gas concentration (NO_2, NH_3, and CO). As the codes for these sensors are open-source, the users can easily reprogram the sensors based on their requirements as well as examine and verify the sensors without any complications. More details about the hardware components are available in the next section.
- Data Processing and Communication Layer: This layer is responsible for processing and integrating data from different sensors and communicating it to the data storage layer. A Raspberry Pi Zero handles all the functions related to data processing and communication. The Wi-Fi module of Raspberry Pi Zero is used to create an access point that allows a continuous flow of data from the Raspberry Pi Zero to the data storage layer. Different data transmission protocols were considered for data transmission. The current version of the CoSense Unit uses the Hyper Text Transfer Protocol (HTTP) due to its high

[2] https://shiny.rstudio.com/

[3] https://github.com/sachit27/Soc-IoT

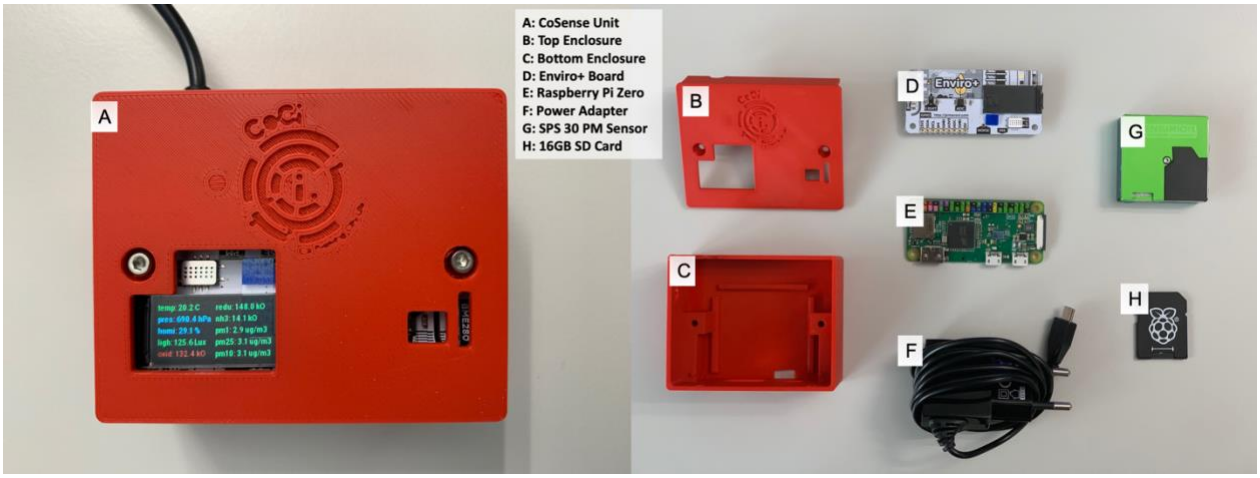

Fig. 4 A complete and exploded view of the CoSense Unit with annotations.

transmission reliability and infrastructure [33], [37].

- Data Storage Layer: This layer is responsible for securely storing the data. The current version of the framework allows two storage options. Either the data can be directly transmitted to the ThingSpeak database or the user can save the data locally in the SD card that comes with the Raspberry Pi. This is beneficial in case of unavailability of the internet to send the data to ThingSpeak cloud. The users can simply upload the data from the SD card to their data stream at a later stage. This also provides more control to the users over their data. If the users prefer not to share their data, they can opt-out of making their data stream public and use the data from the stream and the SD card for their information.
- Application Layer: The data from the storage layer is used to create applications that are used to make sense of the raw data. This includes data streams, visualizations, and data analysis applications. The Soc-IoT framework includes two core applications: (1) ThingSpeak dashboard that allows a user to create data streams, visualize data, and use Matlab functions to perform data analysis. (2) An R-based application that allows a user to do data processing, analysis, visualization, and performs Machine Learning (ML) on the data. Section 3 includes more details about the applications.

B. Hardware Implementation

The CoSense Unit is the hardware component that is responsible for environmental monitoring. It has been designed using state-of-the-art sensors and a single board computer. The current version of the CoSense Unit measures: (1) PM concentration in the air; (2) temperature, pressure, humidity; (3) gas (NO_2, NH_3, CO) concentration; (3) light intensity; and (4) noise. The modular nature of the device allows users to easily remove and add more sensors based on their requirements. The CoSense Unit is easy to assemble and can be used for indoor and outdoor environment sensing. For building a participatory sensing unit, it is important to select the most suitable sensors. While there are a lot of low-cost sensors circulating in the market, not all of them are accurate and efficient when it comes to long-term environmental monitoring. For PM monitoring, the CoSense Unit uses a Sensirion SPS30 PM sensor. The sensor was selected because of its high precision, accuracy, and low bias as compared to other available PM sensors like Plantower PMS5003, SM-UART-04L PM sensor [38], [39]. The SPS30 is capable of monitoring PM_1, $PM_{2.5}$, PM_4, and PM_{10} using the light-based scattering principle. The current version of the CoSense Unit is programmed to monitor PM_1, $PM_{2.5}$, and PM_{10}. In addition to the SPS30 sensor, a sensor array called Enviro Plus[4] that has sensors like BME280 (temperature, humidity, pressure), MICS6814 analog gas sensor (NO_2, NH_3, and CO), LTR-559 light and proximity sensor, and a MEMS microphone (noise) is also added to the CoSense Unit. It also includes the ADS1015 analog to digital converter for converting data from the analog gas sensor and a color LCD. The data produced by the analog gas sensor is in kOhms, which is not the standard unit for gas concentration monitoring. The sensor program converts it into parts per million (ppm) to get an indicative value. Due to a lot of conversion processes, it is difficult to precisely validate it with a regulatory or industry-grade monitor. Nevertheless, the values from the gas sensor can be used as indicative values for understanding how the concentration is changing in a given environment, as highlighted by many studies [40], [41].

Enviro Plus is particularly efficient due to its small size, seamless sensor integration, and compatibility with single board computers like Raspberry Pi. The CoSense Unit uses a Raspberry Pi Zero to communicate with the sensors using the General-Purpose Input Output (GPIO) ports. As Raspberry Pi has multiple GPIO ports, it allows flexibility to add more sensors based on the requirement of a user. Fig. 4 shows the detailed view of the CoSense Unit with components and annotations. All the components are housed within a 3D-printed enclosure. The CoSense Unit is powered using a USB cable to

[4] https://shop.pimoroni.com/products/enviro?variant=31155658457171

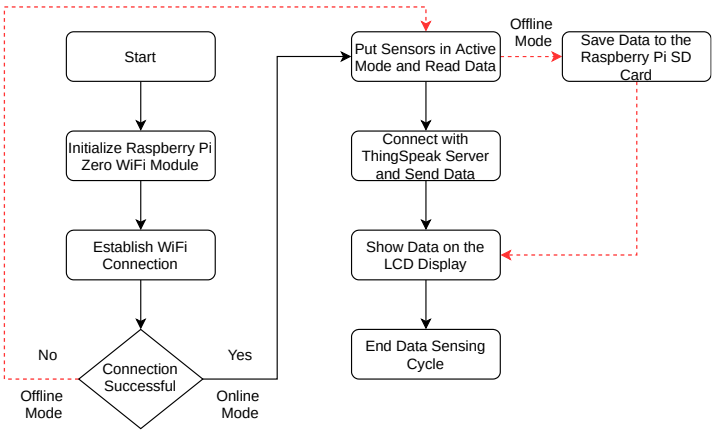

Fig. 5 Flowchart of CoSense Unit software.

provide a 5V supply. The users have a choice to use an adapter or a power bank for powering the Raspberry Pi. This allows the device to be used flexibly for mobile or stationary environmental monitoring.

C. Software Implementation

The CoSense Unit uses a Raspberry Pi Zero to communicate with sensors and handles tasks related to network creation, data transmission, and storage in an SD card. Fig. 5 shows the flowchart of the CoSense Unit source code.

The CoSense Unit source code is written in Python programming language [42] and uses standard sensor libraries to communicate with the sensors. As shown in Fig. 5, once the Raspberry Pi is powered on, it goes in the set-up mode. The Wi-Fi module of the Raspberry Pi goes into the Access Point (AP) mode and allows the user to connect to the device's Wi-Fi network. Once this connection is successful, the users are redirected to a web interface that allows them to connect to a secure Wi-Fi network. The device automatically saves the Wi-Fi credentials that allow the device to connect to the saved Wi-Fi network in case of a reboot. In case no Wi-Fi network is available, the device goes into offline mode. In either case, the sensors are put in the active mode following the connection test. The sensors stay awake for 30 seconds and do the measurement. The measured data is stored in the Raspberry Pi's SD card in CSV format. When the device is in an online mode, an HTTP connection is created and the measured data is sent to the ThingSpeak server using the GET request. Once the acknowledgment is received from the server, the connection is closed. To secure the data transmission, private keys are generated by ThingSpeak before a data stream can be created. The LCD screen shows the data values from the sensors. The availability of online and offline modes allows continuous sensing of data. It is also useful in case environmental monitoring needs to be done in a remote location without internet connectivity. The current version of the prototype measures data every 5 minutes and goes to sleep mode after the measurement. The users can change the sampling frequency based on their needs.

III. Results And Discussion

This section describes the criterion that was used to validate and evaluate the performance of the CoSense Unit, specifically focusing on $PM_{2.5}$ concentration. The results are followed by a discussion to understand how the prototype works in real-life conditions. This section also looks into the design and development of the data analysis and visualization application and how the proposed setup compares with existing environmental monitoring infrastructures.

A. Sensor Validation

Sensor validation is a key step in the development of environmental monitoring infrastructure. There are different ways to perform quality assurance and control of a sensing unit. This study followed a standard approach for validating the sensor by looking at the inter-sensor variability and comparing the sensor output with the official air quality monitoring station [43], [44].

Field Co-location: During the summer of 2021, two CoSense Units were tested in the field in Zurich, Switzerland. To analyze the accuracy of the sensors and evaluate the performance, two units were collocated at one of the sites of the National Air Pollution Monitoring Network (NABEL). NABEL monitors air quality at 16 sites in Switzerland. For this study, the sensor units were collocated at the NABEL station in Dubendorf[5]. Fig. 6 (a) shows the location of the test site. Fig. 6 (b) shows the actual setup of CoSense Units for colocation at the NABEL reference monitoring station. The station is located at a suburban location. The area is densely populated with a network of heavily used roads and railway lines. The field test was conducted between 4 June 2021 and 8 June 2021. The $PM_{2.5}$ was sampled every five minutes and it was averaged to one hour to maintain consistency with the $PM_{2.5}$ data obtained from the reference monitor. Overall, the data was compared for 100 hours.

Fig. 7 presents a line plot that compares the data obtained from two CoSense Units (denoted by Sensor 1 and Sensor 2) and the reference monitor. It can be observed that the CoSense Units can match the variations recorded by the reference monitor. This highlights that the CoSense Unit can successfully capture sudden variations in $PM_{2.5}$ concentration in a real-world environment. The average error between the $PM_{2.5}$ recorded by the reference monitor and Sensor 1 was 1 $\mu g/m^3$. In the case of Sensor 2, it was 1.2 $\mu g/m^3$.

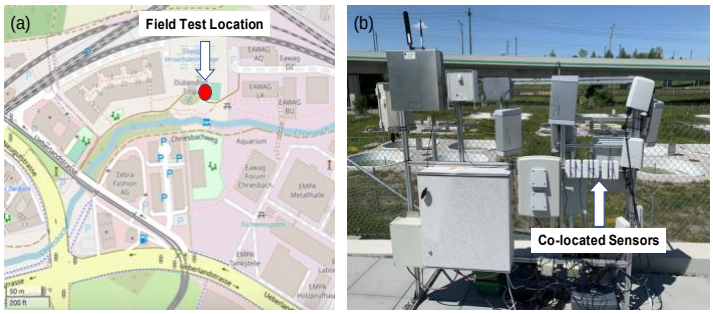

Fig. 6 (a) Red dot on the map shows the field test location. (b) Co-location setup at NABEL monitoring station.

[5] https://www.empa.ch/web/s604/nabel-station-2020

Fig. 7 Line plot of PM2.5 data obtained from two CoSense Units located with the reference monitor at NABEL station.

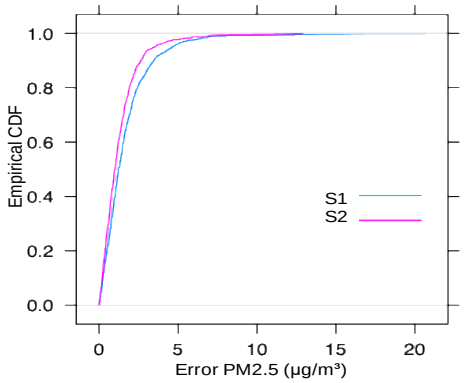

Fig. 8 CDF of the difference between the $PM_{2.5}$ values recorded by the reference monitor and two sensors (S1 and S2).

The error value is very low and shows high accuracy and reliability of the data sensed by the CoSense Units. Fig. 8 shows the empirical cumulative distribution function (CDF) to understand the PM2.5 measurement offset between the reference monitor and the two sensors. It can be observed that more than 85% of the observations have an offset below 5 µg/m^3. A statistical summary of the co-located data is presented in Table I. The statistical parameters show strong similarity between the data obtained from the reference monitor and two CoSense Units.

TABLE I
Summary statistics of PM2.5 (µg/m3) values recorded by the reference station and two CoSense Units

	Reference $PM_{2.5}$	Sensor1 $PM_{2.5}$	Sensor2 $PM_{2.5}$
Mean	5.09	5.04	5.43
Standard Error	0.31	0.27	0.31
Median	3.95	3.99	4.42
Standard Deviation	3.12	2.75	3.10
Sample Variance	9.75	7.55	9.57
Minimum	1.20	2.43	2.54
Maximum	14.1	15.5	16.32

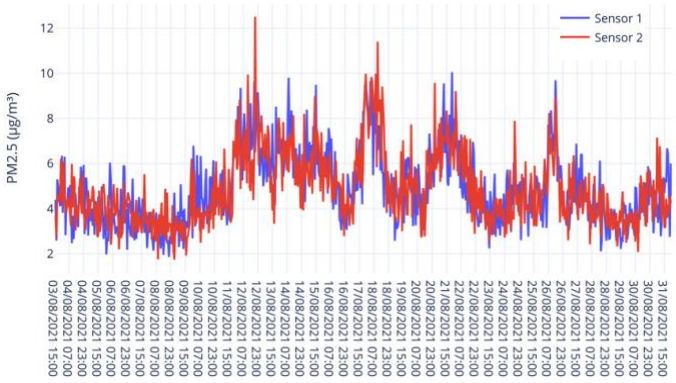

Fig. 9 Line plot of PM2.5 data obtained from two collocated CoSense Units.

TABLE II
Summary statistics of PM2.5 (µg/m3) values recorded by two CoSense Units

	Sensor1 PM2.5	Sensor2 PM2.5
Mean	4.86	4.86
Standard Error	0.06	0.06
Median	4.67	4.56
Standard Deviation	1.60	1.62
Sample Variance	2.55	2.62
Minimum	1.86	1.74
Maximum	10.05	12.51

Inter-Unit Variability: Inter-unit variability is an important method to measure the similarity of data produced by the same sensor units. It is a useful metric that has been widely used to measure the data reproducibility of sensor units [33], [43]. For this study, two CoSense Units were collocated and the $PM_{2.5}$ data were analyzed to understand the similarity in data reported by two units. The study was conducted between 3 August 2021 and 31 August 2021. Fig. 9 shows the line plot based on the data obtained from two units. The data from both the units show a similar trend, except for some outliers. The data was sampled every 5 minutes. For analysis, the data were aggregated to hourly data. Two units were compared for a total of 681 hours. As observed from Table II, the data from the two units showed high similarity. The comparison showed similarities in the observed mean and standard error. Strong linearity was observed over the entire range of hourly averaged $PM_{2.5}$ data.

Sensor Sustainability Analysis: As discussed earlier in the Introduction, the sustainability of IoT devices is also a critical component when discussing resource efficiency. Most of the sensors-related studies usually look into the power consumed by the sensors to address the sustainability of low-cost sensor technology. This work looks at sustainability through a different lens. The approach is not just limited to examining the energy consumption of the IoT device but also to understanding the carbon footprint of the sensor code. To the best of our knowledge, this is no work in air quality monitoring literature that look into this aspect of sensors. This can potentially help in promoting sensor code optimization as well as resource-aware

IoT deployment. For this study, the focus was on two parameters: Emissions (Emissions as CO_2-equivalents, kg of CO_2 emitted per kilowatt-hour of electricity) and Energy Consumed (power consumed in kilowatt-hours). A CoSense Unit with a sampling frequency of one hour would emit approximately 0.029 kg of CO_2 for a month of regular sampling. Similarly, the energy consumption for one month's use of the CoSense Unit would be approximately 0.072 kilowatt-hours. To put these values in context, watching Netflix[6] for half an hour produces 0.4 kg of CO_2, and running an air purifier[7] for twelve hours would use 0.60 kilowatt-hours. These values can give us an idea about how properly designed and optimized sensors can potentially be used in a sustainable way for monitoring the environment in the long run.

B. Data Analysis and Visualization

A key part of any IoT infrastructure is an intuitive and efficient data analysis and visualization platform. IoT devices produce a massive amount of data and to make sense of such that it is important to have user-friendly platforms that can be easily used by experts as well as non-experts. Soc-IoT framework provides two options to visualize and analyze sensor data. The first option uses the in-built data analysis and visualization feature of the ThingSpeak platform[8]. It allows the users to visualize data in real-time, create interactive graphs, set alerts, and statistically analyze the data using MATLAB functions. In addition to this, another non-sensor-specific sensor data analysis and visualization application called *exploreR* is proposed.

exploreR[9] is an open-source online application that has been developed using the Shiny[10] package in the R programming language. RShiny package has been widely used in recent years to create interactive applications for data analysis and visualizations [45]–[47]. Such applications have been used as a motivation to create *exploreR* that is designed to reduce the technical barriers especially related to coding when it comes to analyzing and visualizing citizen-generated data. The next few paragraphs explain the design and architecture of the *exploreR* application.

Design and Architecture: *exploreR* is designed as an intuitive and easy-to-use sensor data analysis and visualization. The application Graphical User Interface (GUI) is designed in a way that guides the user during the analysis process. Fig. 10 shows a snapshot of the GUI of the *exploreR* application. The left column of Fig. 10 (a) holds the main functions that expand once the user decides to use them for data analysis. Fig. 10 (b)-(d) shows the examples of the application of different functions supported by the *exploreR* application. The application framework is designed in a way that follows a series of steps that cover the complete cycle of data input, pre-processing, visualization, and analysis. Fig. 11 shows the schematic representation of the *exploreR* pipeline.

While designing *exploreR*, one of the objectives was to create an application that would facilitate usability for people from diverse backgrounds. Different integrated workflows within the application allow the user to meaningfully interpret the data

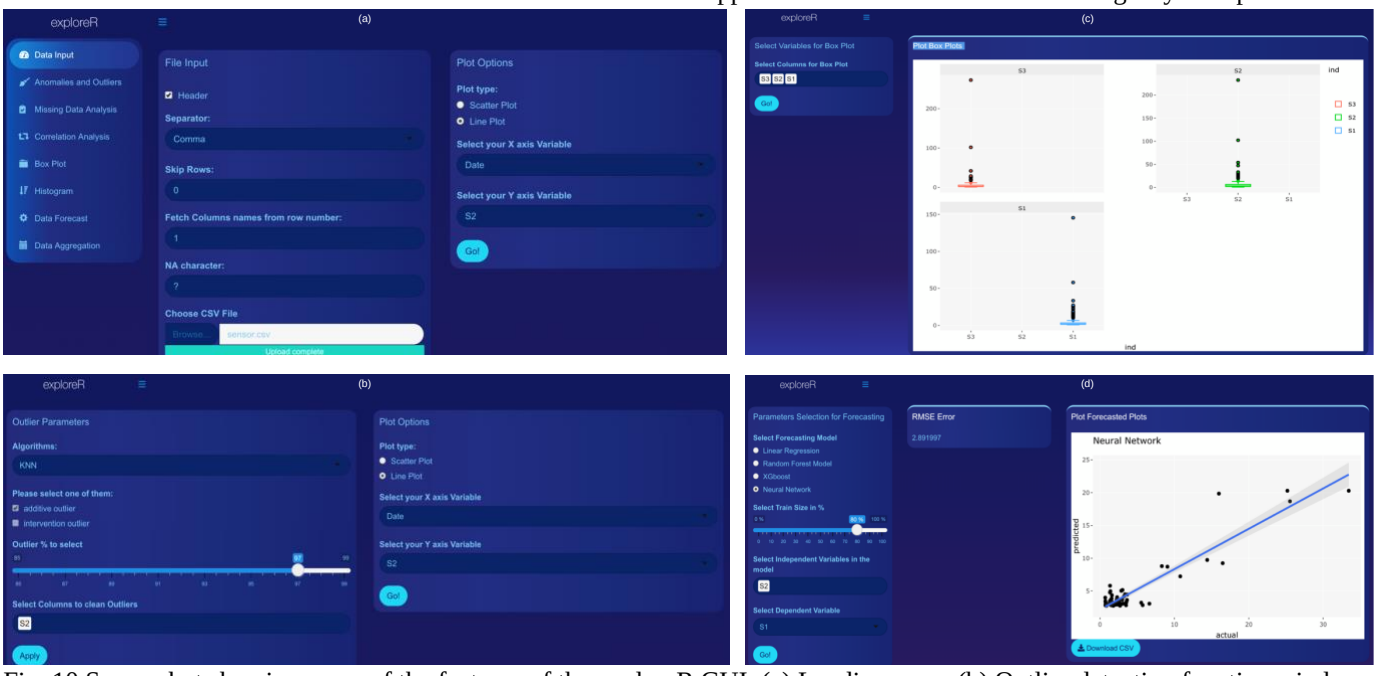

Fig. 10 Screenshot showing some of the features of the exploreR GUI: (a) Landing page, (b) Outlier detection function window, (c) Box plot function window and (d) Data forecast function window.

[6] https://www.iea.org/commentaries/the-carbon-footprint-of-streaming-video-fact-checking-the-headlines

[7] https://reviewsofairpurifiers.com/air-purifier-electricity-consumption-calculator/

[8] https://uk.mathworks.com/products/thingspeak.html

[9] https://sachitmahajan.shinyapps.io/exploreR/

[10] https://shiny.rstudio.com/

without any need for coding. Here is a summary of functions supported by the current version of the application:

- *Data Processing*: The application accepts the data in CSV format and allows the users to filter rows/columns as well as view data summary and plot the raw data. The plots are generated using Plotly[11] which is an interactive graphing library. The generated plots can easily be analyzed using the inbuilt functions like zoom-in/zoom-out, rescaling, among others. The users can save the generated plots in PNG format.
- *Outlier Detection*: The users can use sophisticated statistical and machine learning methods like k-Nearest Neighbour, ARIMA, and Artificial Neural Networks (ANN) to perform anomaly and outlier detection. Data reliability is an important topic that is widely discussed in low-cost sensor literature [44], [48], [49]. The outlier detection function allows the user to look for anomalies, plot them and later clean them using state-of-the-art methods.
- *Gap Filling*: This function allows the users to fill gaps due to missing data or gaps that are generated after removing the outliers in the previous stage. The current version of the applications supports two methods: linear interpolation and Kalman filter. These methods have been used due to their widespread use in sensor literature as well as overall accuracy [50], [51].
- *Exploratory Data Analysis*: This feature allows the users to implement different functions on the dataset to understand the data in more detail as well observe the strengths of the relationship between different variables within the data set. The users can use the Correlation Matrix function to calculate Pearson correlation. Such information can be valuable while creating sensor calibration models [52]. The users can also box plots and histograms to perform a visual analysis of data. The plots can be downloaded as files in PNG format.
- *Data Forecasting*: *exploreR* also has features that can be used for more advanced analysis and understanding of the air quality data. The application allows users to use advanced machine learning algorithms to perform data forecasts. $PM_{2.5}$ forecast is a major challenge as has been widely studied by researchers in atmospheric science, environment monitoring, and computer science domains. The data forecast functions allow the users to use methods ranging from simple to more complex to analyze which method performs well. The current version supports methods like Linear Regression (LR), Random Forest (RF) Model, XGBoost, and ANN. The reason for selecting these models is their widespread use in time-series forecasting research [53], [54]. Having multiple models allow users to compare model performance and potentially use those findings for creating real-time forecasting applications. The forecasting results can be viewed in the application as well as downloaded in CSV format.
- *Data Aggregation*: Different air quality sensors are programmed to record data at a different frequency. Sometimes the data may be too granular or not granular at all. This can lead to an imbalanced time series and adversely impact the overall analysis. To address this challenge, *exploreR* allows the users to downsample the data to daily, weekly, monthly and yearly data. The user can either use the sum or mean to aggregate the data. The aggregated data can be downloaded in CSV format.

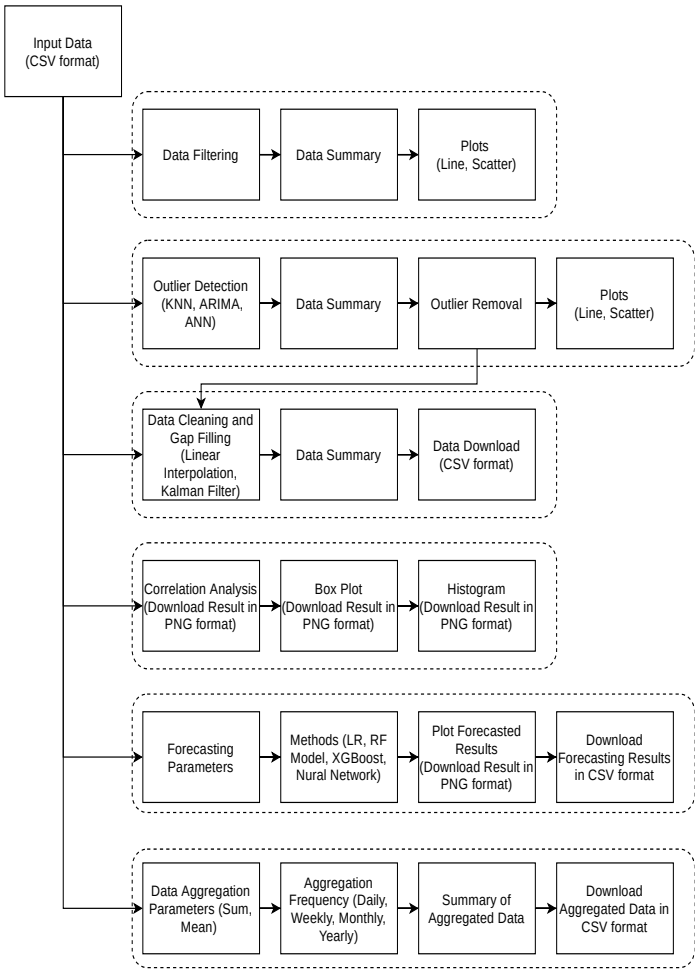

Fig. 11 Schematic of the *exploreR* pipeline.

exploreR is a major component of the Soc-IoT framework and is aimed at the easy analysis of sensor data as well as assisting citizen scientists, policymakers, researchers from non-programming backgrounds to perform data analysis. Such an open-source tool can potentially bridge the gap between experts and non-experts as well as allow citizen scientists to add context while analyzing their data, which is usually missing in case the data is analyzed by a third party.

[11] https://plotly.com/r/

Table III

Comparison of *exploreR* with other air quality data analysis tools and softwares.

Name	Open Source	GUI	Sensor Specific	Programming Requirement	Data Analysis	Data Visualization	Data Forecast
OpenAir [55]	Yes	-	No	Yes	Yes	Yes	Yes
AirSensor [56]	Yes	-	Yes	Yes	Yes	Yes	No
Vayu [57]	Yes	Desktop-based	No	No	Yes	Yes	No
Data Viewer [56]	Yes	Web-based	Yes	No	Yes	Yes	No
Sense Your Data [47]	Yes	Web-based	Yes	No	Yes	Yes	No
PWFSLSmoke [58]	Yes	-	No	Yes	Yes	Yes	Yes
exploreR	Yes	Web-based	No	No	Yes	Yes	Yes

Furthermore, *exploreR* facilitates the easy export of figures and files that can be used for reporting data, publications, and data dissemination.

Comparison with Existing Applications: To understand how this application contributes to the field of open-source sensor data analysis, *exploreR* is compared with similar air quality sensor data analysis applications and softwares. Different applications and softwares have been proposed over the years, with each of them having some strengths and weaknesses. Most of the applications are usually designed for the data from a specific sensor. It works well for data from particular sensors, but with data from different IoT devices, it might not work well. This is mainly due to different data formats as well as the organization of the data. Similarly, with programming-intensive tools, the users who are technically experienced can easily analyze the data but it becomes difficult in case the user has no background in programming languages. Keeping these points in mind, *exploreR* is designed as a non-sensor-specific application that doesn't require any prior knowledge of programming. This allows the users to analyze data from different sensors with ease and without worrying about technical complexities. At the same time, the open-source nature of the application allows the users with training in programming to improve the existing framework by using their skills to add more functions to the application.

Table III compares *exploreR* with other existing open-source tools and softwares that have been widely used for analyzing air quality data obtained using low-cost sensors. Most of the existing solutions are designed keeping in mind specific sensors and user groups. The comparison highlights that exploreR successfully combines features that allow the analysis of data from different sensors without any need for programming.

IV. Conclusion

Leveraging the growth in the Internet of Things (IoT) and its interplay with sustainable practices and open-source principles, this paper proposes Soc-IoT, a proof-of-concept framework for citizen-centric environmental monitoring. The framework promotes accurate and efficient environmental monitoring by integrating open-source hardware and software. The core part of the framework focuses on enhancing embedded spatial intelligence where citizen empowerment meets smart environments and sustainable design. The proposed open-source framework has the potential to promote and encourage collaboration among a wide range of stakeholders ranging from scientists to policymakers and citizen scientists to Maker groups. It can also be used for educational as well as awareness purposes. The high reliability and accuracy of data sensed by the CoSense Unit allows it to be potentially used for complementing official monitoring networks. The simple and extensible nature of the proposed framework would hopefully encourage others to use it as a development platform rather than reinventing everything from scratch.

Future work will include more investigation into dynamic calibration and edge analytics. Additional enhancements to the data analysis tool will include improvement in the user interface and the addition of more functionalities.

Acknowledgment

The author would like to thank Mr. Beat Schwarzenbach and Dr. Christoph Hüglin who supported in testing the CoSense Units at the NABEL facility at Empa, Dubendorf, and Mr. Manuel Knott for designing the 3D model for the CoSense Unit enclosure. The author also wishes to thank Christoph Laib, Thomas Maillart, Stefan Klauser, and Octanis Instruments for their early work related to air quality sensors during the Climate City Cup initiative, and Sensirion for donating the SPS30

modules. Special thanks are due to the CoCi project team for their contribution during the development of the CoSense Unit.

REFERENCES

[1] W. Liang and M. Yang, "Urbanization, economic growth and environmental pollution: Evidence from China," *Sustain. Comput. Informatics Syst.*, vol. 21, Mar. 2019.

[2] F. Perera, "Pollution from Fossil-Fuel Combustion is the Leading Environmental Threat to Global Pediatric Health and Equity: Solutions Exist," *Int. J. Environ. Res. Public Health*, vol. 15, no. 1, Dec. 2017.

[3] WHO, "Ambient (outdoor) air quality and health," 2014.

[4] R. B. Hamanaka and G. M. Mutlu, "Particulate Matter Air Pollution: Effects on the Cardiovascular System," *Front. Endocrinol. (Lausanne).*, vol. 9, Nov. 2018.

[5] H. Riojas-Rodriguez, A. S. da Silva, J. L. Texcalac-Sangrador, and G. L. Moreno-Banda, "Air pollution management and control in Latin America and the Caribbean: implications for climate change," *Rev. Panam. Salud Pública*, vol. 40, pp. 150–159, 2016.

[6] J. Shah and B. Mishra, "IoT enabled environmental monitoring system for smart cities," in *2016 International Conference on Internet of Things and Applications (IOTA)*, 2016.

[7] S. Dhingra, R. B. Madda, A. H. Gandomi, R. Patan, and M. Daneshmand, "Internet of Things Mobile–Air Pollution Monitoring System (IoT-Mobair)," *IEEE Internet Things J.*, vol. 6, no. 3, Jun. 2019.

[8] D. Helbing *et al.*, "Ethics of Smart Cities: Towards Value-Sensitive Design and Co-Evolving City Life," *Sustainability*, vol. 13, no. 20, Oct. 2021.

[9] N. Castell *et al.*, "Can commercial low-cost sensor platforms contribute to air quality monitoring and exposure estimates?," *Environ. Int.*, vol. 99, pp. 293–302, Feb. 2017.

[10] L.-J. Chen *et al.*, "An Open Framework for Participatory PM2.5 Monitoring in Smart Cities," *IEEE Access*, vol. 5, pp. 14441–14454, 2017.

[11] S. Mahajan *et al.*, "A citizen science approach for enhancing public understanding of air pollution," *Sustain. Cities Soc.*, 2020.

[12] Luftdaten, "Measuring air data with citizen science.," 2021. [Online]. Available: https://luftdaten.info/. [Accessed: 02-Apr-2021].

[13] F. Today, "Curious noses measure air quality across Flanders." 2018.

[14] OpenAQ, "Fighting air inequality through open data and community," 2021. [Online]. Available: https://openaq.org/#/locations?page=1. [Accessed: 10-Jun-2021].

[15] S. Mahajan, C.-H. Luo, D.-Y. Wu, and L.-J. Chen, "From Do-It-Yourself (DIY) to Do-It-Together (DIT): Reflections on designing a citizen-driven air quality monitoring framework in Taiwan," *Sustain. Cities Soc.*, vol. 66, p. 102628, 2021.

[16] H. Pritchard and J. Gabrys, "From Citizen Sensing to Collective Monitoring: Working through the Perceptive and Affective Problematics of Environmental Pollution," *GeoHumanities*, vol. 2, no. 2, pp. 354–371, 2016.

[17] A. Commodore, S. Wilson, O. Muhammad, E. Svendsen, and J. Pearce, "Community-based participatory research for the study of air pollution: a review of motivations, approaches, and outcomes," *Environ. Monit. Assess.*, vol. 189, no. 8, p. 378, Aug. 2017.

[18] Sensors.Africa, "Citizen science initiative that uses sensors to monitor air, water and sound pollution," 2021. [Online]. Available: https://sensors.africa/. [Accessed: 09-Jun-2021].

[19] S. Mahajan, T.-C. Tsai, W.-L. Wu, and L.-J. Chen, "Design and implementation of IoT-enabled personal air quality assistant on instant messenger," in *MEDES 2018 - 10th International Conference on Management of Digital EcoSystems*, 2018.

[20] S. Mahajan, Y.-S. Tang, D.-Y. Wu, T.-C. Tsai, and L.-J. Chen, "CAR: The Clean Air Routing Algorithm for Path Navigation With Minimal PM2.5 Exposure on the Move," *IEEE Access*, vol. 7, pp. 147373–147382, 2019.

[21] HabitatMap, "AirCasting Map." [Online]. Available: https://www.habitatmap.org/airbeam. [Accessed: 03-Apr-2021].

[22] Toma, Alexandru, Popa, and Zamfiroiu, "IoT Solution for Smart Cities' Pollution Monitoring and the Security Challenges," *Sensors*, vol. 19, no. 15, Aug. 2019.

[23] L.-J. Chen, Y.-H. Ho, H.-H. Hsieh, S.-T. Huang, H.-C. Lee, and S. Mahajan, "ADF: An Anomaly Detection Framework for Large-Scale PM2.5 Sensing Systems," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 559–570, Apr. 2018.

[24] G. Camprodon *et al.*, "Smart Citizen Kit and Station: An open environmental monitoring system for citizen participation and scientific experimentation," *HardwareX*, 2019.

[25] J. Ma, Y. Ding, J. C. P. Cheng, F. Jiang, V. J. L. Gan, and Z. Xu, "A Lag-FLSTM deep learning network based on Bayesian Optimization for multi-sequential-variant PM2.5 prediction," *Sustain. Cities Soc.*, vol. 60, Sep. 2020.

[26] C.-H. Luo, H. Yang, L.-P. Huang, S. Mahajan, and L.-J. Chen, "A Fast PM2.5 Forecast Approach Based on Time-Series Data Analysis, Regression and Regularization," in *2018 Conference on Technologies and Applications of Artificial Intelligence (TAAI)*, 2018, pp. 78–81.

[27] M. Van Oudheusden and Y. Abe, "Beyond the Grassroots: Two Trajectories of 'Citizen Sciencization' in Environmental Governance," 2021.

[28] X. Zhang, R. C. Estoque, H. Xie, Y. Murayama, and M. Ranagalage, "Bibliometric analysis of highly cited articles on ecosystem services," *PLoS One*, vol. 14, no. 2, 2019.

[29] M. Aria and C. Cuccurullo, "bibliometrix: An R-tool for comprehensive science mapping analysis," *J. Informetr.*, vol. 11, no. 4, 2017.

[30] H. Lu, M. Halappanavar, and A. Kalyanaraman, "Parallel heuristics for scalable community detection," *Parallel Comput.*, vol. 47, 2015.

[31] L. Spinelle, M. Gerboles, M. G. Villani, M. Aleixandre, and F. Bonavitacola, "Calibration of a cluster of low-cost sensors for the measurement of air pollution in ambient air," in *IEEE SENSORS 2014 Proceedings*, 2014, pp. 21–24.

[32] M. Balestrini, A. Kotsev, M. Ponti, and S. Schade, "Collaboration matters: capacity building, up-scaling, spreading, and sustainability in citizen-generated data projects," *Humanit. Soc. Sci. Commun.*, vol. 8, no. 1, p. 169, 2021.

[33] S. Mahajan, J. Gabrys, and J. Armitage, “AirKit: A Citizen-Sensing Toolkit for Monitoring Air Quality,” *Sensors*, vol. 21, no. 12, p. 4044, Jun. 2021.

[34] H. Y. Teh, A. W. Kempa-Liehr, and K. I.-K. Wang, “Sensor data quality: a systematic review,” *J. Big Data*, vol. 7, no. 1, Dec. 2020.

[35] E. Fiore, “Ethics of technology and design ethics in socio-technical systems,” *FormAkademisk - forskningstidsskrift Des. og Des.*, vol. 13, no. 1, Jan. 2020.

[36] F. Mao, K. Khamis, S. Krause, J. Clark, and D. M. Hannah, “Low-Cost Environmental Sensor Networks: Recent Advances and Future Directions,” *Front. Earth Sci.*, vol. 7, Sep. 2019.

[37] G. Mois, S. Folea, and T. Sanislav, “Analysis of Three IoT-Based Wireless Sensors for Environmental Monitoring,” *IEEE Trans. Instrum. Meas.*, vol. 66, no. 8, pp. 2056–2064, Aug. 2017.

[38] S. Sousan, S. Regmi, and Y. M. Park, “Laboratory Evaluation of Low-Cost Optical Particle Counters for Environmental and Occupational Exposures,” *Sensors*, vol. 21, no. 12, p. 4146, Jun. 2021.

[39] J. Tryner, J. Mehaffy, D. Miller-Lionberg, and J. Volckens, “Effects of aerosol type and simulated aging on performance of low-cost PM sensors,” *J. Aerosol Sci.*, vol. 150, p. 105654, Dec. 2020.

[40] G. Marques and R. Pitarma, “A cost-effective air quality supervision solution for enhanced living environments through the internet of things,” *Electron.*, vol. 8, no. 2, 2019.

[41] C. T. Dang, A. Seiderer, and E. André, “Theodor: A step towards smart home applications with electronic noses,” in *ACM International Conference Proceeding Series*, 2018.

[42] M. F. Sanner, “Python: A programming language for software integration and development,” *Journal of Molecular Graphics and Modelling*, vol. 17, no. 1. 1999.

[43] M. Tagle *et al.*, “Field performance of a low-cost sensor in the monitoring of particulate matter in Santiago, Chile,” *Environ. Monit. Assess.*, vol. 192, no. 3, pp. 1–18, 2020.

[44] B. Fishbain *et al.*, “An evaluation tool kit of air quality micro-sensing units,” *Sci. Total Environ.*, vol. 575, 2017.

[45] Y. Yu, Y. Ouyang, and W. Yao, “ShinyCircos: An R/Shiny application for interactive creation of Circos plot,” *Bioinformatics*, vol. 34, no. 7, 2018.

[46] K. K. Nisa, H. A. Andrianto, and R. Mardhiyyah, “Hotspot clustering using DBSCAN algorithm and shiny web framework,” in *Proceedings - ICACSIS 2014: 2014 International Conference on Advanced Computer Science and Information Systems*, 2014.

[47] S. Mahajan and P. Kumar, “Sense Your Data: Sensor Toolbox Manual, Version 1.0.,” 2019.

[48] B. Maag, Z. Zhou, and L. Thiele, “A Survey on Sensor Calibration in Air Pollution Monitoring Deployments,” *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4857–4870, Dec. 2018.

[49] E. S. Cross *et al.*, “Use of electrochemical sensors for measurement of air pollution: Correcting interference response and validating measurements,” *Atmos. Meas. Tech.*, 2017.

[50] C. D. Dorich *et al.*, “Global Research Alliance N2O chamber methodology guidelines: Guidelines for gap-filling missing measurements,” *J. Environ. Qual.*, vol. 49, no. 5, 2020.

[51] N. Alavi, J. S. Warland, and A. A. Berg, “Filling gaps in evapotranspiration measurements for water budget studies: Evaluation of a Kalman filtering approach,” *Agric. For. Meteorol.*, vol. 141, no. 1, 2006.

[52] S. Mahajan and P. Kumar, “Evaluation of low-cost sensors for quantitative personal exposure monitoring,” *Sustain. Cities Soc.*, vol. 57, 2020.

[53] B. Pan, “Application of XGBoost algorithm in hourly PM2.5 concentration prediction,” in *IOP Conference Series: Earth and Environmental Science*, 2018, vol. 113, no. 1.

[54] S. Mahajan and P. Kumar, “Evaluation of low-cost sensors for quantitative personal exposure monitoring,” *Sustain. Cities Soc.*, vol. 57, p. 102076, Jun. 2020.

[55] D. C. Carslaw and K. Ropkins, “Openair—an R package for air quality data analysis,” *Environ. Model. \& Softw.*, vol. 27, pp. 52–61, 2012.

[56] B. Feenstra, A. Collier-Oxandale, V. Papapostolou, D. Cocker, and A. Polidori, “The AirSensor open-source R-package and DataViewer web application for interpreting community data collected by low-cost sensor networks,” *Environ. Model. \& Softw.*, vol. 134, p. 104832, 2020.

[57] S. Mahajan, “Vayu: An Open-Source Toolbox for Visualization and Analysis of Crowd-Sourced Sensor Data,” *Sensors*, vol. 21, no. 22, 2021.

[58] J. Callahan *et al.*, “PWFSLSmoke: Utilities for Working with Air Quality Monitoring Data,” *R Packag. Version*, vol. 1, p. 111, 2019.