CityLightSense: A Participatory Sensing-based System for Monitoring and Mapping of Illumination levels

ASIF IQBAL MIDDYA, SARBANI ROY, and DEBJANI CHATTOPADHYAY, Jadavpur University, India

Adequate nighttime lighting of city streets is necessary for safe vehicle and pedestrian movement, deterrent of crime, improvement of the citizens' perceptions of safety, and so on. However, monitoring and mapping of illumination levels in city streets during the nighttime is a tedious activity that is usually based on manual inspection reports. The advancement in smartphone technology comes up with a better way to monitor city illumination using a rich set of smartphone-equipped inexpensive but powerful sensors (e.g., light sensor, GPS, etc). In this context, the main objective of this work is to use the power of smartphone sensors and IoT-cloud-based framework to collect, store, and analyze nighttime illumination data from citizens to generate high granular city illumination map. The development of high granular illumination map is an effective way of visualizing and assessing the illumination of city streets during nighttime. In this article, an illumination mapping algorithm called Street Illumination Mapping is proposed that works on participatory sensing-based illumination data collected using smartphones as IoT devices to generate city illumination map. The proposed method is evaluated on a real-world illumination dataset collected by participants in two different urban areas of city Kolkata. The results are also compared with the baseline mapping techniques, namely, Spatial k-Nearest Neighbors, Inverse Distance Weighting, Random Forest Regressor, Support Vector Regressor, and Artificial Neural Network.

CCS Concepts: • **Information systems** → **Spatial-temporal systems**; • **Human-centered computing** → **Ubiquitous and mobile computing systems and tools**; **Smartphones;**

Additional Key Words and Phrases: Participatory sensing, illumination mapping, smartphone sensors

ACM Reference format:
Asif Iqbal Middya, Sarbani Roy, and Debjani Chattopadhyay. 2021. CityLightSense: A Participatory Sensing-based System for Monitoring and Mapping of Illumination levels. *ACM Trans. Spatial Algorithms Syst.* 8, 1, Article 5 (October 2021), 22 pages.
https://doi.org/10.1145/3487364

5

The research work of Asif Iqbal Middya is supported by UGC-NET Junior Research Fellowship (UGC-Ref. No. 3684/(NET-JULY 2018)) provided by the University Grants Commission, Government of India. This research work is also supported by the "Participatory and Realtime Pollution Monitoring System For Smart City" project, funded by the Department of Science & Technology, Government of West Bengal, India.
Authors' addresses: A. I. Middya, S. Roy, and D. Chattopadhyay, Dept. of Computer Science and Engineering, Jadavpur University, Kolkata, India; emails: asifim.rs@jadavpuruniversity.in, sarbani.roy@jadavpuruniversity.in, chattopadhyay.debjani01@gmail.com.

2374-0353/2021/10-ART5 $15.00
https://doi.org/10.1145/3487364

1 INTRODUCTION

Smartphones are rapidly becoming an integral part of citizens' everyday lives. Importantly, the smartphones of today are equipped with a rich set of low-cost but powerful built-in sensors (e.g., GPS, light sensor, accelerometer, proximity sensor, etc.). Such smartphone embedded sensors can be utilized to monitor the illumination dynamics of city streets during the nighttime. Regular monitoring of illumination levels of city streets during the nighttime is necessary for ensuring safe vehicle and pedestrian movement, deterrent of crime, improvement of the citizens' perceptions of safety, and so on [4, 13]. The illumination levels at different road segments in a city during the nighttime depend on various factors that make it difficult for measuring and mapping of lux levels across the streets of a city in real-time. For example, illumination levels (i.e., lux values) at a road segment is usually influenced by street lighting design scheme (i.e., distribution of street lamps, types of lamps, etc.), the presence of lighting system in adjacent Point of Interests (e.g., roadside shops, restaurants, shopping malls, gas stations, etc.), current weather conditions (e.g., foggy nights), and so on. The widespread adoption of smartphones across all age groups makes it feasible to solve the problem of monitoring illumination levels during the night times.

Now, there are different traditional ways of monitoring street lighting across the city. The most common approach is to perform a manual inspection [4] of the street lighting by the city officials. Here, the officials carry lux meter to measure illumination levels. This approach of manual monitoring suffers from a lack of scalability. Also, this approach may require that the road segment be blocked during such inspection, and may eventually cause serious inconvenience to citizens. Another approach is to deploy static light sensor modules on streetlight poles for monitoring the illumination levels and control them remotely [14, 15]. But, such an approach also suffers from a number of limitations like expensive equipment, periodic maintenance, human expertise, and so on. This approach may not be effective in the cases where it is difficult to upgrade the existing lighting system. Besides, the precise calculation of street-level illumination is not achievable in this method, as the static sensor modules are mounted on the frames or poles of the lamps and can thus measure the lighting levels near the lamps. Finally, there is an approach that involves the deployment of vehicle-mounted sensor system [18] for illumination monitoring and mapping. However, mounting many costly sensors on a car-top to periodically collect data, is not a cost-effective option for developing countries. Hence, there is a need for a low cost and scalable way to measure, monitor, and map the illumination levels in the city streets.

To overcome the above-mentioned limitations, a participatory sensing-based (similar to crowdsensing) approach [7, 16, 23–25, 27] can be used for monitoring illumination levels of city streets. It utilizes the power of citizens' smartphone-equipped sensors and **Internet-of-Things (IoT)**-cloud-based framework to collect, store, and analyze nighttime illumination levels. In other words, the concept of participatory sensing enables people and communities to contribute sensory information to explore illumination dynamics of city streets during nighttime. Citizens use **Global Positioning System (GPS)**-enabled smartphones as IoT devices to collect and share geotagged illumination levels. Eventually, it will provide a scalable, high granular, and cost-effective solution for illumination monitoring.

In this article, an illumination mapping and monitoring system called *CityLightSense* is proposed. An algorithm called **Street Illumination Mapping (SIM)** is presented that can be used to monitor and map illumination levels in the city streets during nighttime. SIM works on participatory sensing-based illumination data collected using smartphone embedded ambient light sensor of the citizens. Note that the raw data from smartphone sensors may suffer from several inaccuracies. For example, there may have errors with the raw sampling locations (i.e., the smartphone provided GPS locations of the samples). Also, citizens may collect data from the

inappropriate sensing contexts (e.g., the data that are collected while the phone is in the pocket/bag are not useful for illumination monitoring). Moreover, low-cost smartphone sensors do not provide high-quality illumination data. To address these issues, several pre-processing techniques (namely, map matching, context-aware data cleaning, and calibration) are built to pre-process the raw sensor data. SIM makes use of **Density-based spatial clustering of applications with noise (DBSCAN)** clustering [10] and **inverse distance squared (IDS)**-based [3] interpolation to perform the spatial mapping of illumination in city streets. Several techniques like **Spatial k-Nearest Neighbors (Spatial kNN)** [33], **Inverse Distance Weighting (IDW)** [20], **Random Forest Regressor (RFR)** [5], **Support Vector Regressor (SVR)** [11, 22], **Artificial Neural Network (ANN)** [21], among others, are considered as baselines to compare and analyze the effectiveness of SIM. The major contributions are listed as follows:

- A participatory sensing-based IoT-cloud framework (CityLightSense) is proposed for measuring, monitoring and mapping of illumination levels at city streets during the nighttime.
- A novel illumination mapping algorithm called SIM is proposed that works on the illumination data collected using smartphones to generate the illumination maps of city streets.
- Several pre-processing techniques are developed to ensure the data quality of the smartphones.
- The proposed approach SIM is evaluated on real-world participatory sensing-based illumination data. To evaluate the efficacy of SIM, it is also compared with the baseline techniques.

The rest of this article is organized as follows: Section 2 presents the existing works on street illumination monitoring. Section 3 provides the problem statement and system overview. In Section 4, sensing and data collection are discussed. Section 5 presents different pre-processing tasks (namely, data cleaning, map matching, and calibration). Section 6 provides the details of the model building. In Section 7, evaluation and results are discussed. Section 8 concludes this work.

2 RELATED WORK

In this section, literature reviews on the existing light monitoring systems for city streets are provided.

There are a few significant works available in this area of research. One such important work is performed by Kumar et al. [18] that presents a framework to collect, log, and evaluate data on street lighting infrastructure during the night-time. The authors utilized a car-mounted sensor system that consists of lux meters for illumination level mapping, cameras to detect street lamps, GPS, and OBD-II for location estimation, and so on. However, mounting several expensive sensors on car-top to collect data regularly is not a cost-effective solution for the developing countries. In Reference [19], Lee et al. present a non-invasive system to detect defective street lights by analyzing street illumination levels. They built an equipment called Hitchhiker, which is mounted on vehicles to measure street lighting levels. However, they did not address the algorithmic challenges of spatially continuous street illumination mapping using real-world noisy data. In Reference [35], Zatari et al. propose a system to measure glare, luminance, and illuminance of urban street lighting. Here, the authors utilized the vehicle-mounted CCD digital video cameras with storage capabilities to measure the above-mentioned lighting parameters. But, this system incurs high cost due to expensive equipment, off-line analysis, and human expertise.

Also, there are some existing works that use static light sensor modules for street light monitoring. For example, in Reference [15], Jing et al. investigated a system to remotely track street lights by installing dedicated light sensors on streetlight poles. In this work, the system fuses the sensed data with the help of wireless sensor network technology that mainly consists of a **remote**

terminal unit (RTU), a **general packet radio service (GPRS)** module, and a control center. However, this type of system is less scalable, needs periodic maintenance and not cost-effective.

The recent advancements in smartphone technologies enable their use in monitoring city dynamics [6, 9, 36] by utilizing inbuilt sensors. Reference [1] uses smartphone-based participatory sensing (also called crowdsensing) method for monitoring urban street lighting. In Reference [1], Alam et al. develop a crowdsensing-based system to monitor the illumination of streets during night-time. In this work, the smartphone-equipped light sensor and GPS sensor are mainly utilized, where the smartphones are mounted on vehicles. Although the raw sensor data from low-cost smartphone sensors suffer from uncertainties with the measurements, the authors directly use the raw data without performing any pre-processing task. A detailed list of existing works is summarized in Table 1.

In this work, our proposed approach differs from the existing literature in the following aspects: (i) Citizens' smartphone-equipped sensor data (e.g., GPS, light sensor, proximity, accelerometer, etc.) are utilized for illumination mapping. There is no context-aware data-cleaning technique for smartphone-based illumination level sensing that we are aware of. As a result, in smartphone-based ambient illumination sensing, we developed a predictive model to identify appropriate and inappropriate sensing contexts. Specifically, after feature extraction and selection, a random forest-based classification model is established using the real-world data of smartphone-equipped sensors. The model is then used to discover the sensing context and remove data that are coming from the inappropriate sensing contexts. (ii) There is currently no mapping technique that employs DBSCAN (density-based spatial clustering of applications with noise) clustering on crowdsensing data to eliminate outliers before spatial interpolation. However, in our proposed work, by incorporating DBSCAN before spatial mapping of street illumination level has proved to be an effective approach that has improved the model accuracy (see Section 7.4 for more details). (iii) An IDS-based approach is designed and implemented for illumination level prediction. The findings highlight that it is simple yet a very powerful technique for the prediction of illumination level, which has improved the prediction accuracy to a large extent. To the best of our knowledge, there are no existing works that identify the potential of an IDS-based method for crowdsensing-based city illumination mapping. (iv) Instead of using any existing calibration model, in this study, a smartphone model-specific calibration technique for illumination level sensing is also proposed and built. (v) Our overall approach and analysis are significantly different in terms of purpose (i.e., city illumination mapping), outputs, insights and reporting from the existing crowdsensing-based works.

3 PROBLEM STATEMENT AND SYSTEM OVERVIEW

Let $\mathcal{U} = \{u_1, u_2, u_3, \ldots, u_l\}$ is a set of smartphone users who are using *CityLightSense* application for city illumination level monitoring and $\mathcal{R}_\mathcal{N}$ is the road network (Definition 2) of the city. Also, let us consider $\mathcal{D} = \{s_1, s_2, s_3, \ldots, s_z\}$ is a set of illumination data samples (Definition 1) collected by the participants. Now, let us consider $\mathcal{G} = \{g_1, g_2, g_3, \ldots, g_h\}$ is a set of square grids on the geographic space. Now, suppose $\mathcal{G}' = \{g_i | g_i \in \mathcal{G}, g_i \; overlaps \; with \; \mathcal{R}_\mathcal{N}\}$. Our objective is to come up with an efficient technique that can address several issues (e.g., errors with sampling locations, inappropriate sensing context, inaccurate measurements by smartphone sensors, etc.) and find the illumination levels at the grids that overlap with the road network $\mathcal{R}_\mathcal{N}$. In other words, we need to find a set of predicted grids' illumination levels $\mathcal{I}_{\mathcal{G}'}{}^{(pred)} = \{\mathcal{I}_{g_1}^{(pred)}, \mathcal{I}_{g_2}^{(pred)}, \ldots, \mathcal{I}_{g_{|\mathcal{G}'|}}^{(pred)}\}$, where, $\mathcal{I}_{g_i}^{(pred)}$, $1 \leq i \leq |\mathcal{G}'|$ is the predicted grid illumination level of grid $g_i \in \mathcal{G}'$. Moreover, our aim is to produce an illumination map using those estimated grid illumination levels for the visual assessment of city streets' illumination.

Table 2 summarizes some key notations that appear throughout the paper.

Table 1. A Summary of Existing Works

Approach	Reference & Year	Sensors/devices
Smartphone-based crowdsensing	Alam et al. [1], 2018	Ambient light and GPS sensors
Deployment of vehicle mounted sensor system	Kumar et al. [18], 2016	Light sensor, Camera, GPS, IMU, OBD
	Zatari et al. [35], 2005	CCD camera, GPS, 3D orientation sensor
	Lee et al. [19], 2014	Light sensor, GPS, transceiver.
	Vaaja et al. [31], 2018	GNSS, IMU, camera, laser scanner
Deployment of static sensor modules on streetlight poles	Jing et al. [15], 2007	Light sensor, RTU, GPRS
	Huang et al. [14], 2004	MOTES sensor nodes
	Sunehra et al. [30], 2017	PIR and LDR sensor, transceiver, GSM
Manual inspection	Boyce et al. [4], 2000	Light meter

OBD: On-board diagnostics; IMU: Inertial Measurement Unit; GPS: Global Positioning System; CCD: Charge-coupled device; GNSS: global navigation satellite system; RTU: remote terminal unit; GPRS: general packet radio service; PIR: passive infrared sensor; LDR: light-dependent resistor.

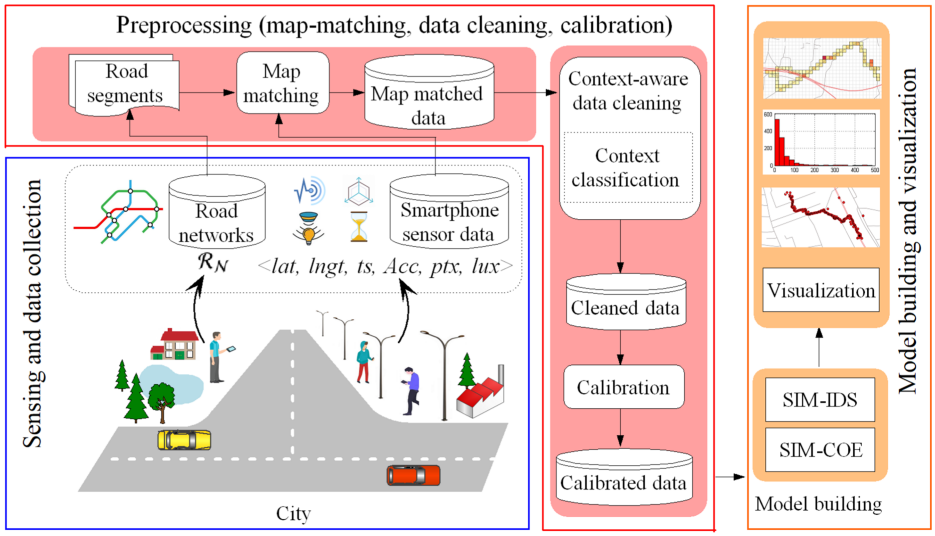
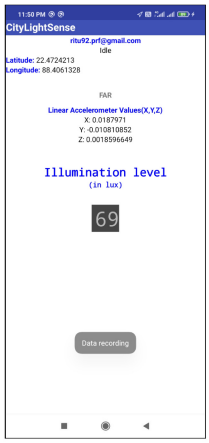


Fig. 1. A participatory sensing-based framework for city streets' illumination monitoring.

Definition 1 (Illumination Sample). A participatory sensing-based illumination level sample s_i is denoted using following format $s_i = \langle l_i, ts_i, Lux_i, Acc_i^x, Acc_i^y, Acc_i^z, Ptx_i \rangle$, where $l_i = \{lat_i, lngt_i\}$ is the sampling location in the form of latitude and longitude, respectively, ts_i is the timestamp when the sample is collected, Lux_i is the recorded illumination level, Acc_i^x, Acc_i^y and Acc_i^z are the three-axis accelerometer values, and Ptx_i is the proximity value.

Definition 2 (Road Network). A road network $\mathcal{R}_N$ is viewed as an undirected graph $\mathcal{R}_N = (V, R)$, where $\mathcal{R}^{(s)} = \{r_1, r_2, \ldots, r_m\}$ is a set of road segments and V is a set of terminal points of the road segments.

Figure 1 provides an overview of the participatory sensing-based monitoring and mapping of illumination levels. The framework mainly consists of three major parts-sensing and data collection, pre-processing, model building and visualization. Based on pre-processed data, the proposed



(a)

	A	B	C	D	E	F	G	H
1	YYYY-MO-DD HH-MI-SS_SSS	Latitude	Longitude	Illumination (lux)	Acc_X	Acc_Y	Acc_Z	Proximity
2	2020-06-09 18:57:28:951	22.36132	88.43182	36	-0.201487	-0.09754	-0.413079	FAR
3	2020-06-09 18:57:30:951	22.361343	88.43181	45	-0.884255	0.091125	-0.387211	FAR
4	2020-06-09 18:57:32:951	22.361355	88.4318	41	0.117478	-0.191819	0.190938	FAR
5	2020-06-09 18:57:34:951	22.361385	88.43178	38	0.373873	-0.826819	-0.175091	FAR
6	2020-06-09 18:57:36:951	22.361406	88.43178	40	-0.085436	0.134063	0.263522	FAR
7	2020-06-09 18:57:38:951	22.361423	88.43177	39	0.126345	-0.21738	0.162717	FAR
8	2020-06-09 18:57:40:951	22.361444	88.43175	25	-3.772894	-0.29881	3.153038	FAR
9	2020-06-09 18:57:42:951	22.361467	88.43173	15	-0.497799	0.588269	1.338777	FAR
10	2020-06-09 18:57:44:951	22.361483	88.43172	21	-0.608602	0.152836	0.479507	FAR
11	2020-06-09 18:57:46:951	22.3615	88.4317	17	2.842704	-3.122463	5.840481	FAR
12	2020-06-09 18:57:48:951	22.361519	88.43168	33	-0.517144	0.450414	-0.168716	FAR
13	2020-06-09 18:57:50:951	22.361536	88.43166	34	-1.876894	1.191573	0.245272	FAR
14	2020-06-09 18:57:52:951	22.361557	88.43165	77	-0.192577	-0.763679	-0.115142	FAR
15	2020-06-09 18:57:54:951	22.361572	88.43163	70	0.220532	-0.009765	0.328124	FAR
16	2020-06-09 18:57:56:951	22.361588	88.431625	73	-0.640192	-0.477233	-0.080422	FAR
17	2020-06-09 18:57:58:951	22.361609	88.43162	63	0.760359	-0.53776	-0.803421	FAR
18	2020-06-09 18:58:00:951	22.361628	88.43162	69	0.741318	-0.777581	3.605336	FAR
19	2020-06-09 18:58:02:951	22.361649	88.4316	71	0.381826	0.921332	-1.634994	NEAR
20	2020-06-09 18:58:04:951	22.361673	88.431595	0	0.57065	-1.09935	1.552518	NEAR
21	2020-06-09 18:58:06:951	22.361692	88.431595	1	-0.790835	0.065525	-0.244766	NEAR
22	2020-06-09 18:58:08:951	22.361702	88.43157	1	0.434006	-1.641697	-0.083726	NEAR
23	2020-06-09 18:58:10:951	22.361713	88.431564	1	0.853156	0.498547	0.006607	NEAR
24	2020-06-09 18:58:12:951	22.361723	88.43157	0	-0.688509	-1.057586	0.328407	NEAR
25	2020-06-09 18:58:14:951	22.361738	88.43157	0	-0.481825	0.113321	-1.37567	NEAR

(b)

Fig. 2. (a) User interface of data logging activity in the CityLightSense. (b) Sample dataset of various smartphone sensor data (e.g., illumination level, accelerometer, etc.) used in the illumination monitoring and mapping.

illumination level mapping technique SIM is used to monitor and map city illumination levels. This technique makes use of inverse distance squared-based non-geostatistical interpolation along with DBSCAN for illumination level mapping. Finally, state-of-the-art visualization techniques are employed to visualize the results.

4 SENSING AND DATA COLLECTION

Participatory sensing enables citizens to participate in sensing city illumination levels with the help of their smartphones. An Android application called *CityLightSense* is developed to collect spatio-temporal illumination levels (i.e., illumination levels with GPS location and timestamp). The measured illumination levels are further tagged with some other smartphone-equipped sensor values like three-axis accelerometer, proximity sensor value, and so on. These additional sensors' information are necessary for context-aware data cleaning (more details in Section 5.2). The data samples are logged using *CityLightSense* at every 2 s interval in the format: $\langle l_i, ts_i, Lux_i, Acc_i^x, Acc_i^y, Acc_i^z, Ptx_i \rangle$, where $l_i = \{lat_i, lngt_i\}$, ts_i, Lux_i, Acc_i^x, Acc_i^y, Acc_i^z, and Ptx_i are location (in the form of latitude and longitude), timestamp, illumination level, x-component of accelerometer, y-component of accelerometer, z-component of accelerometer, and proximity sensor value, respectively. The user interface of data logging activity of the Android application (i.e., CityLightSense) is provided in Figure 2(a). However, a sample dataset of the above-mentioned sensors' readings is presented in Figure 2(b).

The participatory sensing-based illumination level data from the users are aggregated and stored in the cloud for city illumination level monitoring and mapping. As shown in Figure 1, a series of pre-processing tasks (namely, map matching, data cleaning, and calibration) are performed on raw sensor data. Road network data $\mathcal{R}_N$ and map-matching technique [34] are utilized to enhance estimates of sampling location. In context-aware data cleaning, the smartphones' three-axis accelerometer values and proximity sensor values are used to detect the sensing context (i.e., whether the smartphone was in hand or bag/pocket during sensing). Context detection is necessary to perform context-aware data cleaning to remove those data samples that are collected while the phone was

Table 2. Notation

Symbols	Description
$\mathcal{D}$	Dataset containing illumination data samples
s_j	j^{th} data sample
$s_j.t_s$	Timestamp when data sample s_j is recorded.
$s_j.lux$	Lux value of sample s_j.
$s_j.loc$	Location of sample s_j.
t_{pq}	Time interval of prediction
$\mathcal{G}$	A set of square grids
n	Dimension of square grids
n_s	Total number of illumination samples
g_i	i^{th} grid
m_i	Grid midpoint of $g_i \in \mathcal{G}$
$\mathcal{R}_\mathcal{N}$	Road network
C	Set of density clusters
C_i	i^{th} density cluster
$I_{g_i}^{(pred)}$	Estimated illumination level at grid $g_i \in \mathcal{G}'$
ϵ	Radius parameter of DBSCAN
$MinPts$	Neighborhood density threshold of DBSCAN
O	Outlier data samples
$\mathcal{L}$	A sequence of sampling locations
$\mathcal{R}^{(s)}$	A set of road segments
r_i	i^{th} road segment
$\mathcal{D}_{pq}$	Dataset containing samples within interval t_{pq}
d_{ij}	Distance between m_i and $s_j.loc$
$\mathcal{N}_{C_j}^{(g)}$	The number of grids containing samples of cluster C_j
Acc_x	x-axis data of accelerometer reading
Acc_y	y-axis data of accelerometer reading
Acc_z	z-axis data of accelerometer reading
η	Distance between closest pair of samples.
Ptx	Proximity sensor reading
K	Maximum number of samples used for prediction at each grid.

in bag/pocket. Calibration is also necessary as a pre-processing task to estimate actual measurements from smartphone sensor responses.

5 PRE-PROCESSING

In this section, three pre-processing techniques, namely, map matching, context-aware data cleaning, and calibration, are discussed that are utilized to ensure data quality.

5.1 Map matching

The GPS locations of the recorded data samples may suffer from positioning errors and eventually differs from the actual sampling locations. To address this problem, map-matching [34] technique is utilized. Map matching matches the raw GPS sampling locations to the locations on the road network. In this work, an existing interactive-voting-based method [34] is employed for map matching. Let us consider $\mathcal{L} = \langle s_1.l_1, s_2.l_2, s_3.l_3, \ldots, s_w.l_w \rangle$ is a sequence of sampling locations of

consecutive samples $s_1, s_2, s_3, \ldots, s_w$ collected by an arbitrary user. Let, $\mathcal{R}^{(s)} = \{r_1, r_2, r_3, \ldots, r_m\}$ is a set of road segments. Now, map matching is performed by the following steps:

- *Candidate determination:* For each $l_i \in \mathcal{L}$, a set of candidate locations $CL_i^{(d)} = \{C_1^i, C_2^i, C_3^i, \ldots, C_t^i\}$, where $C_j^i \in CL_i^{(d)}$ is a projection of l_i on a road segment in $\mathcal{R}$ such that $dist(l_i, C_j^i) \leq d$. Here, $dist(l_i, C_j^i)$ denotes the distance between l_i and C_j^i. Next, the observation probability of $C_j^i \in CL_i^{(d)}$ with respect to $l_i \in \mathcal{L}$ is given as [32]

$$Z(C_j^i) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(dist(l_i, C_j^i)-\mu)^2}{2\sigma^2}}. \tag{1}$$

- *Score matrix generation:* Next, for $l_i \in \mathcal{L}$, a score matrix is generated [34]. Each element of this matrix indicates the probability of the point C_j^i to be a correct projection. Now, for each $C_j^i \in CL_i^{(d)}$, a sequence of candidate points $Seq_j^i = \langle C_s^i, \ldots, C_j^i, \ldots, C_q^w \rangle$ is created that has maximum score summation.
- *Interactive voting:* Next, each C_j^i votes for the candidates in Seq_j^i [34]. Let $\mathcal{V}_i = \{V_{i1}, V_{i2}, V_{i3}, \ldots, V_{it}\}$ represent the votes obtained by the candidate points for $l_i \in \mathcal{L}$, where V_{ij} denotes the number of votes acquired by $C_j^i \in CL_i^{(d)}$. Next, the candidate point $C_p^i \in CL_i^{(d)}$ is found that has highest votes for $l_i \in \mathcal{L}$, i.e., $V_{ip} = max(\mathcal{V}_i)$. This candidate point is called winner candidate point and denoted by $C_i^{(win)}$.
- *Map-matched location sequence:* The winning candidate points are combined to find map-matched sequence of sampling locations. hence, map-matched sequence of sampling locations for $\mathcal{L} = \langle s_1.l_1, s_2.l_2, s_3.l_3, \ldots, s_w.l_w \rangle$ will be $\mathcal{L}^{map-matched} = \langle C_1^{(win)}, C_2^{(win)}, C_3^{(win)}, \ldots, C_w^{(win)} \rangle$.

5.2 Context-aware Data Cleaning

In the smartphone-based light-sensing application, the usefulness of the illumination data collected from the participants depends on the phone sensing context (i.e., where the smartphone is kept). A previous study [8] shows that the common location of keeping smartphones are hand, bag, and pocket. Note that keeping smartphones in pocket/bag during the ambient illumination data collection will provide inaccurate measurements of the ambient illumination levels. Hence, it is required to remove the samples that are measured from those inappropriate sensing contexts (e.g., pocket/bag). In other words, for context-aware data cleaning, we need to develop predictive model for identifying the appropriate (i.e., "hand") and inappropriate (i.e., "pocket/bag") sensing contexts. As far as we know, there is no context-aware data-cleaning solution available for smartphone-based illumination level sensing. As a result, in smartphone-based illumination sensing, a binary classifier is necessary to build for identifying whether the sensing context is hand or pocket/bag. In this work, a random forest-based classifier is developed for context classification based on the real-world data of smartphone-equipped sensors. The context classification model yields an accuracy of 94%. Note that the smartphone data of three-axis accelerometer (Acc_x, Acc_y, and Acc_z), three-axis gyroscope (G_x, G_y, and G_z) and the proximity sensor (Ptx) values are utilized to build the context classification model. Some volunteers are appointed to acquire these sensors' readings from different contexts (i.e., when the phone is in hand and pocket/bag). Each of the volunteers is asked to record a total of 20 min of data for each sensing context. After examining the sensor data, it is decided to exclude the three-axis gyroscope data (G_x, G_y, and G_z) values as they are not significantly different in two sensing contexts. Since the accelerometer sensor is direction sensitive, the magnitude $M_g = \sqrt{Acc_x^2 + Acc_y^2 + Acc_z^2}$ is computed and used along with

Table 3. Extracted Features

Features	Formula
Max amplitude	$S_{max} = max\{S[v]\}$
Min amplitude	$S_{min} = min\{S[v]\}$
Mean	$\mu = \frac{1}{T}\sum_{v=1}^{T} S[v]$
Variance	$\sigma^2 = \frac{1}{T}\sum_{v=1}^{T}(S[v] - \mu)^2$
Kurtosis	$Ku = m_4/m_2^2$
Skewness	$Sk = (m_3) \times (m_3^{3/2})$

$S[]$: Sensor data segment; $S[v]$: vth sample of the segment; T: Total number of sensor data samples within the segment; m_2: 2nd moment about the mean; m_3: 3rd moment about the mean; m_4: 4th moment about the mean.

the three-axis accelerometer sensor data. Next, Acc_x, Acc_y, Acc_z, M_g, and Ptx data are split into fixed-sized small segments of 5 s with 50% overlap. Now, features like mean, variance, kurtosis, and skewness are extracted for each segment of the sensor data. The extracted features along with their statistical formulas are listed in Table 3. In the case of accelerometer data, the mean and variance of both Acc_z and M_g have high feature importance score [2]. However, for proximity sensor data, only the feature mean has a high feature importance score.

5.3 Calibration

One of the problems with the smartphone-equipped light sensor is the uncertainties in the measurements of ambient light [12]. Here, by the word "uncertainty," we mean that there could be a variation in measured values between different smartphones for the same illumination level. The crowd/participatory sensing-based approach utilizes the smartphone-equipped low-cost light sensor to generate high granular illumination data of city streets. However, these low-cost sensors may not provide measurements as accurate as the highly expensive professional devices designed specifically for ambient light measurement. It is observed that the sensor readings may vary in different smartphone models (probably because of the quality of sensors used). To address this problem, calibration [12] of smartphone light sensors is important to estimate actual measurements from the smartphone sensor responses. The readings from smartphone light sensor and professional light meter are measured and compared, which reveals that there exists a linear relationship between smartphone light sensor responses and professional light meter measurements. More specifically, a linear model (Equation (2)) is proposed to determine the relationship between smartphone light sensor responses (R) and professional light meter measurements (R'):

$$R' = \beta_0 + \beta_1 R + \xi, \tag{2}$$

where ξ denotes the error term and β_0 and β_1 represent the coefficients. The least-square estimation is employed to find the value of β_0 and β_1. If $(R'_i, R_i), i = 1, 2, \ldots, u$ denotes a set of samples of smartphone responses and actual measurements, then the value of β_0 and β_1 need to be computed for which the **residual sum of squares (RSS)** is minimized. Hence the objective function could be written as follows (Equation (3)). Note that, in Equation (3), $\widehat{\beta_0}$ and $\widehat{\beta_1}$ are estimates of β_0 and

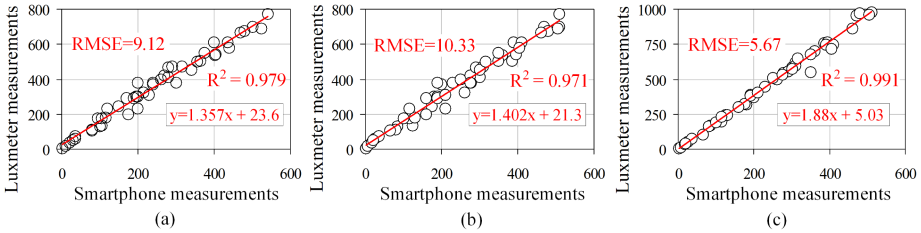


Fig. 3. Scatter plots of smartphone measurements and corresponding lux meter readings (in lux) for three different smartphone models: (a) Moto Z2 Play, (b) Redmi Y2, and (c) Moto e7 power. The calibration lines (in red) represent a linear relationship between smartphone measurements and ground-truth lux values. The root mean square error (RMSE) and R^2 (i.e., coefficient of determination) values represent the quality of model fitting.

β_1, respectively:

$$\min \sum_{i=1}^{u} \left(R'_i - \widehat{\beta}_0 - \widehat{\beta}_1 R_i\right)^2. \tag{3}$$

To find β_0, RSS is first differentiated with respect to β_0, which yield Equation (4) after simplification:

$$\widehat{\beta}_0 = \frac{\sum_{i=1}^{u}\left(R'_i - \widehat{\beta}_1 R_i\right)}{u} = \sum_{i=1}^{u}\frac{R'_i}{u} - \widehat{\beta}_1 \sum_{i=1}^{u}\frac{R_i}{u} = \bar{R}' - \bar{R}\widehat{\beta}_1, \tag{4}$$

where u is the total number of data points, and $\bar{R}$ and $\bar{R}'$ denote the mean of R and R', respectively. Now, differentiating the RSS with respect to β_1 will yield Equation (5) after some simplification. Note that, β_0 and β_1 from Equations (4) and (5) together define the least-square coefficient estimates:

$$\widehat{\beta}_1 = \frac{u\sum_{i=1}^{u} R_i R'_i - \sum_{i=1}^{u} R_i \sum_{i=1}^{u} R'_i}{u\sum_{i=1}^{u} R_i^2 - \left(\sum_{i=1}^{u} R_i\right)^2} = \frac{\sum_{i=1}^{u} R_i R'_i - u\bar{R}\bar{R}'}{\sum_{i=1}^{u} R_i^2 - u\bar{R}^2}, \tag{5}$$

where $\bar{R}$ and $\bar{R}'$ denote the mean of R and R', respectively. Given smart phone response (R'_i) and the actual measurement (R_i) pairs, i.e., $\{(R'_1, R_1), (R'_2, R_2), \ldots, (R'_u, R_u)\}$, the coefficients can be determined using Equations (4) and (5). Now, different smartphone models may produce different light sensor responses for the same illuminance [17]. Eventually, smartphone model specific calibration is required. In this context, it should be noted that the data of a participant can be immediately used in the illumination monitoring and mapping if a pre-build calibration model corresponding to the smartphone model used by the participant already exists in the system. If a pre-build calibration model is not available in the system for a particular smartphone model, then we have to create the calibration model before using the data. For three smartphone models (namely, Moto Z2 Play, Redmi Y2, and Moto e7 power), the scatter plots of smartphone measurements and corresponding lux meter readings along with the linear models are presented in Figure 3. The linear models are used to map smartphone specific light sensor responses to professional light meter readings. The **root-mean-square error (RMSE)** value of the linear fitting is provided in each calibration model in Figure 3 as a measure of uncertainties. Moreover, the R^2 (i.e., coefficient of determination) values are also provided. The RMSE and R^2 together can also represent the quality of model fitting.

6 MODEL BUILDING

The SIM algorithm is presented in Algorithm 1. SIM consists of two major parts (Figure 4): *clustering and outlier elimination* (SIM-COE) and *inverse distance squared interpolation* (SIM-IDS). Given a

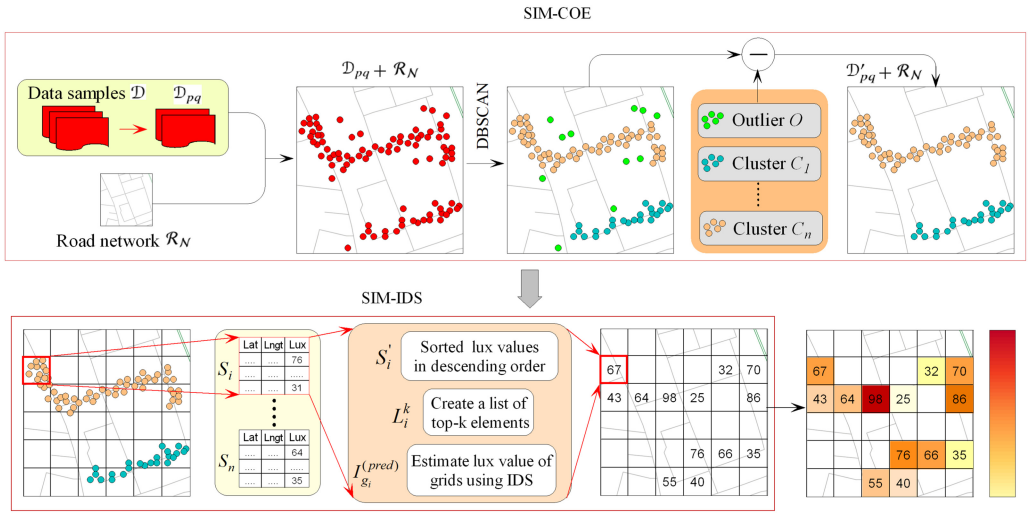


Fig. 4. Two major stages of Street Illumination Mapping (SIM) algorithm: SIM-COE (Algorithm 2) and SIM-IDS (Algorithm 3).

dataset $\mathcal{D}$ of geotagged illumination samples, SIM-COE creates a new dataset $\mathcal{D}_{pq}$ of samples that belong to a specified time interval (for example, data samples belonging to the last one-month interval may be used to explore street illumination mapping). Next, SIM-COE finds the density-based clusters $C_1, C_2, \ldots, C_m$ and outliers O of the sampling locations using DBSCAN. It eliminates the set of outliers O (i.e., those sample points that are not part of any clusters) from the data. Density-based clustering is required before illumination prediction and mapping at any area of interest. Note that the proposed approach of SIM uses crowd/participatory sensing data where the mobility pattern of the participants affects illumination data collection. In this context, there could be a lack of data samples on some streets of the city that are not covered by many participants. Here, the intuition behind density-based clustering of data samples is that the prediction accuracy depends on the density of the illumination samples in an area. The existing literature [20] also highlights that the performance of an interpolation-based mapping method is influenced by the density of the samples in the geographic space. Hence, the DBSCAN model is built to eliminate the set of those sample points that are not part of any density clusters. There is currently no mapping technique that uses DBSCAN clustering on crowdsensing data to eliminate outliers before spatial interpolation. Note that introducing DBSCAN before spatial mapping of street illumination levels has proven to be an effective approach in this work, which has greatly improved predictive performance (see Section 7.4 for more details). Next, SIM-IDS divides the geospace in the square area of grids. After that, it predicts illumination levels at those grids that overlap with the road network using the IDS-based interpolation method. Here, an IDS-based approach is provided for illumination level prediction, mainly due to the fact that the intensity of light diminishes in the inverse square of the distance. It is a simple but effective method for predicting illumination levels, which has improved prediction accuracy (see Section 7 for more details).

Initially, SIM calls SIM-COE($\mathcal{D}$,ϵ, $MinPts$, t_{pq}) function (Algorithm 2). SIM-COE() takes sample dataset $\mathcal{D}$, a radius parameter ϵ, time interval of prediction t_{pq}, and a neighborhood density threshold $MinPts$. Suppose, $t_{pq} = [t_p, t_q]$ is the given prediction interval or the illumination monitoring interval of the city. Now, in Algorithm 2, first a new dataset $\mathcal{D}_{pq} \subseteq \mathcal{D}$ is created that contains data samples s_j having timestamp $s_j.t_s \in [t_p, t_q]$. Next, an object of DBSCAN model is created

ALGORITHM 1: SIM($\mathcal{D}, \epsilon, MinPts, t_{pq}, \mathcal{R}_\mathcal{N}, K, n$)

Input: Sample data $\mathcal{D}$ Radius parameter ϵ, Neighborhood density threshold $MinPts$, Time interval of prediction t_{pq}, Threshold parameter K, Grid dimension n.

Output: Set of grids $\mathcal{G}'$ that overlaps with $\mathcal{R}_\mathcal{N}$, Predicted illumination levels $I_{\mathcal{G}'}^{(pred)}$ for the grids in $\mathcal{G}'$

1: $(\mathcal{D}'_{pq}, C) \leftarrow$ SIM-COE($\mathcal{D}, \epsilon, MinPts, t_{pq}$)
2: $(\mathcal{G}', I_{\mathcal{G}'}^{(pred)}) \leftarrow$ SIM-IDS($\mathcal{D}'_{pq}, C, n, \mathcal{R}_\mathcal{N}, K$)
3: **return** $(\mathcal{G}', I_{\mathcal{G}'}^{(pred)})$

ALGORITHM 2: SIM-COE($\mathcal{D}, \epsilon, MinPts, t_{pq}$)

Input: Sample data $\mathcal{D}$ Radius parameter ϵ, Neighborhood density threshold $MinPts$, Time interval of prediction t_{pq}

Output: Refined data $\mathcal{D}'_{pq}$, Density clusters C.

1: $\mathcal{D}_{pq} = \{s_j | s_j \in \mathcal{D}, s_j.t_s \in [p, q]\}$;
2: $Model = DBSCAN(\epsilon, MinPts)$ //create an object of DBSCAN model
3: $Model.fit(\mathcal{D}_{pq})$ //fit the data samples
4: $C_1, C_2, \ldots, C_m = Model.cluster()$ // the density clusters produced by DBSCAN
5: $O = Model.outlier()$ // set of outliers produced by DBSCAN
6: $C = \{C_1, C_2, \ldots, C_m\}$
7: $\mathcal{D}'_{pq} = \mathcal{D}_{pq} - O$
8: **return** $(\mathcal{D}'_{pq}, C)$

using the radius parameter (ϵ) [10] and the neighborhood density threshold ($MinPts$) [10]. After that, the DBSCAN clustering is used to perform the density-based clustering. It will return a set of clusters C and a set of samples O that is not part of any cluster. More specifically, outlier samples are $O = \{s_y | s_y \notin C_i\}$, where $C_i \in C$. Next, the samples $s_y \in O$ are removed from the dataset $\mathcal{D}_{pq}$ to create a new dataset $\mathcal{D}'_{pq}$ i.e., $\mathcal{D}'_{pq} = \mathcal{D}_{pq} - O$. After that, SIM calls SIM-IDS() function (Algorithm 3) for illumination level mapping on the road network. SIM-IDS() takes the data samples $\mathcal{D}'_{pq}$, set of density clusters C, grid dimension n, road network $\mathcal{R}_\mathcal{N}$, and threshold parameter K as input. Note that K is an input threshold that indicates the maximum number of samples considered during the prediction of illumination level at each grid midpoint. The threshold is required to filter out the samples with very low (or relatively low) illumination levels. This filtering is necessary, because if we allow all the samples, then there could be many samples for which the phone is improperly aligned with the light source [17]. First, SIM-IDS() divides the geospace into a set $\mathcal{G}$ of square grids g_i. Now, in line 3, a subset $\mathcal{G}' \subseteq \mathcal{G}$ is created. $\mathcal{G}'$ contains only those grids that overlap with the road segments of $\mathcal{R}_\mathcal{N}$. Next, let $S_i \subseteq \mathcal{D}'_{pq}$ is the set of samples that are within grid $g_i \in \mathcal{G}'$. The samples $s_j \in S_i$ are sorted in non-increasing order of their illumination levels at grid $g_i \in \mathcal{G}'$ to produce a list S'_i. Let, L_i^k denote first k elements of S'_i, where $k = Min(K, t)$. Note that, except some worst case scenario (as discussed in the Conclusion section) for which the dataset is sparse, participatory sensing usually allow us to collect high granular and dense data that leads to having many samples in each grid and hence the value of K (in this work K = 10) is expected to be much less compared to t. Here, t is the number of samples present in a grid, i.e., number of samples in the ordered list S'_i. Next, for each grid $g_i \in \mathcal{G}'$, we calculate the distances d_{ij} of the sampling location $s_i^j.loc$ of sample $s_i^j \in L_i^k$, $i \neq j$ from the midpoint m_i of grid g_i. Let, for each grid $g_i \in \mathcal{G}'$,

D_i^k denotes all such distances. Next, from lines 20 to 22, we estimate the grid illumination level of grid $g_i \in \mathcal{G}'$. IDS-based interpolation is used to estimate the illumination level $I_{g_i}^{(pred)}$ as follows:

$$I_{g_i}^{(pred)} = \frac{\sum_{j=1}^{|D_i^k|} \frac{1}{(d_{ij})^2} s_i^j.lux}{\sum_{j=1}^{|D_i^k|} \frac{1}{(d_{ij})^2}}. \tag{6}$$

Here, $\frac{1}{(d_{ij})^2}$ is the weight associated with $s_i^j.lux$, where $s_i^j.lux$ denotes the illumination level of the sample s_i^j. Note that, as the distance d_{ij} between m_i and $s_i^j.loc$ decreases, the associated weight $\frac{1}{(d_{ij})^2}$ increases. In other words, IDS sets greater weights to the illumination samples that are spatially closer to the prediction location (i.e., grid midpoint g_i) than those samples that are far away. Here, the IDS method is a special case of **inverse distance weighting (IDW)** [3] interpolation when the power parameter of IDW is 2. Each prediction $I_{g_i}^{(pred)}$ comes with a *support of prediction*. Suppose, the samples within a grid g_i are coming from cluster C_j, then the support of prediction (Definition 3) is the cluster density (Definition 3) of samples in cluster C_j. If cluster density ($D_j^{(c)}$) is high, then it is expected to produce a better prediction for the grids that are within C_j. It is because a high cluster density indicates that a large number of samples are participating in the prediction process of grids' illumination levels that lead to better prediction accuracy.

Definition 3 (Support of Prediction). Suppose, $I_{g_i}^{(pred)}$ is the predicted illumination level at grid $g_i \in \mathcal{G}$. Also, let us consider, $|S_i \cap C_j| \geq |S_i \cap C_q|, \forall C_q \in C$ and $j \neq q$. Now the support of the predicted illumination level $I_i^{(pred)}$ is the cluster density $D_j^{(c)}$ of samples in cluster C_j. Here, the cluster density $D_j^{(c)}$ is defined as the average number of samples per grid within that cluster. Mathematically,

$$\mathcal{S}_i^{(pred)} = D_j^{(c)} = \frac{|C_j|}{\mathcal{N}_{C_j}^{(g)}}, \tag{7}$$

where $\mathcal{N}_{C_j}^{(g)}$ is the number of grids containing samples of cluster C_j.

In clustering and outlier elimination (SIM-COE) stage (Algorithm 2), DBSCAN takes $O(n_s^2)$ time in the worst case [29], where n_s is the total number of sample points. Now, in the SIM-IDS stage (Algorithm 3), the for loop in line 4 iterates over a maximum of $O(|\mathcal{G}'|)$ times. The sorting in line 6 takes $O(t \log t)$ times in the worst case. The for loop in line 11 iterates over a maximum of $O(k)$ times. Now, prediction in line 21 takes a maximum of $O(k)$ times. Eventually, in worst case Algorithm 3 takes $O(|\mathcal{G}'|(t \log t + k + k)) \approx O(|\mathcal{G}'|(t \log t)) \approx O(|\mathcal{G}'|(n_s \log n_s))$, since $t \geq k$ and in worst case $t = n_s$. Hence, overall, time complexity of SIM is $O(n_s^2 + |\mathcal{G}'|(n_s \log n_s))$.

In the SIM-COE stage (Algorithm 2), DBSCAN takes $O(n_s)$ storage space [29]. In the SIM-IDS stage (Algorithm 3), to maintain all the grids, $O(|\mathcal{G}'|)$ space is required. The sorting operation takes $O(t)$ space. To maintain the distances between the midpoint and the samples $O(k)$ space is required. To store prediction results in $I_{\mathcal{G}'}^{(pred)}$ the space required is $O(|\mathcal{G}'|)$. The amount of storage space required by Algorithm 3 is $O(|\mathcal{G}'| + t + k + |\mathcal{G}'|)$. Since in worst case $t = n_s \geq k$, the space required by Algorithm 3 can be written as $O(|\mathcal{G}'| + t + k + |\mathcal{G}'|) \approx O(|\mathcal{G}'| + n_s)$. Hence, overall space complexity of SIM is $O(n_s) + O(|\mathcal{G}'| + n_s) \approx O(|\mathcal{G}'| + n_s)$.

Note that some aspects related to the illumination mapping such as the nature of participatory sensing data, gridding, and model parameters may influence the grid-based illumination level prediction in SIM. In this context, Theorems 1, 2, and 3 attempt to highlight how the above-mentioned

ALGORITHM 3: SIM-IDS($\mathcal{D}'_{pq}, C, n, \mathcal{R}_N, K$)

Input: Data samples $\mathcal{D}'_{pq}$, Density clusters C, Grid dimension n, Road network $\mathcal{R}_N$, Threshold parameter K

Output: Set of grids $\mathcal{G}'$ that overlaps with $\mathcal{R}_N$, Predicted illumination levels $I_{\mathcal{G}'}^{(pred)}$ for the grids in $\mathcal{G}'$

$I_{\mathcal{G}'}^{(pred)} \leftarrow \emptyset$
Create a set of grids $\mathcal{G} = \{g_1, g_2, \ldots, g_h\}$ with dimension $n \times n$;
$\mathcal{G}' = \{g_i | g_i \in \mathcal{G}, g_i$ overlapps wirh $\mathcal{R}_N\}$
for each $g_i \in \mathcal{G}'$ **do**
  $S_i = \{s_j | s_j \in \mathcal{D}'_{pq}$ and $s_j.loc$ resides in $g_i \in \mathcal{G}'\}$
  Sort the samples in S_i to create a ordered list $S'_i = \langle s_i^1, s_i^2, \ldots, s_i^t \rangle$ such that $s_i^1.lux \geq s_i^2.lux \geq \cdots \geq s_i^t.lux$
  Create a set $L_i^k = \{s_i^1, s_i^2, \ldots, s_i^k\}$ with first k elements of S'_i, where $k = Min(K, t)$
  $m_i \leftarrow$ grid midpoint of g_i
  $D_i^k \leftarrow \emptyset$
  $I_{g_i}^{(pred)} \leftarrow NULL$
  for each sample $s_i^j \in L_i^k$ **do**
    $d_{ij} \leftarrow$ distance between m_i and $s_i^j.loc$.
    if $d_{ij} == 0$ **then**
      $I_{g_i}^{(pred)} = s_i^j.lux$
      $I_{\mathcal{G}'}^{(pred)} = I_{\mathcal{G}'}^{(pred)} \cup I_{g_i}^{(pred)}$
      break
    end if
    $D_i^k \leftarrow D_i^k \cup \{d_{ij}\}$
  end for
  if $I_{g_i}^{(pred)} == NULL$ and $|D_i^k| \geq 1$ **then**
    $$I_{g_i}^{(pred)} = \frac{\sum_{j=1}^{|D_i^k|} \frac{1}{(d_{ij})^2} s_i^j.lux}{\sum_{j=1}^{|D_i^k|} \frac{1}{(d_{ij})^2}}$$
    $I_{\mathcal{G}'}^{(pred)} = I_{\mathcal{G}'}^{(pred)} \cup I_{g_i}^{(pred)}$
  end if
end for
return $(\mathcal{G}', I_{\mathcal{G}'}^{(pred)})$

aspects affect the mapping of illumination levels. First, participatory sensing may sometimes produce sparse datasets as the data collection in participatory sensing depends on the mobility pattern of the participants. In this context, we may have a scenario when there exist non-empty grids with a single data sample. Theorem 1 shows that, in this case, the proposed approach SIM is reduced to a simple mapping where the predicted value of illumination level will be the illumination level of the single sample contained in that grid. Second, the illumination mapping using SIM can be indirectly affected by the grid dimension. Theorem 2 shows that if grid size $n < \eta/\sqrt{2}$, then each grid contains at most one sample, where η is the distance between the closest pair of samples. This theorem indirectly highlights that the grid dimension should be sufficiently large; otherwise, many grids may remain unpredicted by SIM because of the absence of the samples within the grids.

Third, the grid illumination mapping in SIM may be highly affected by the value of threshold K. For instance, as discussed in Theorem 3, if K is too small (say K = 1), then the approach SIM will be reduced to a mapping where the predicted value of illumination level of a grid will be the highest illumination level among all the samples that reside in the grid.

Theorem 1. *If a grid $g_i \in \mathcal{G}$ contains only one sample s_j, then the predicted grid illumination level $I_{g_i}^{(pred)} = s_j.lux$, where $s_j.lux$ is the lux value of sample s_j.*

Proof. Since $g_i \in \mathcal{G}$ contains only one sample s_j, we can write $S_i = L_i^k = \{s_j\}$. Let, the distance between grid midpoint m_i and the sampling location of s_j is $d_{ij} = d$. Now, two cases may occur. *Case-1:* If $d = 0$, then according to SIM-IDS, $I_{g_i}^{(pred)} = s_j.lux$. *Case-2:* If $d > 0$, then the set $D_i^k = \{d\}$. Eventually, $I_{g_i}^{(pred)} = (\frac{1}{d^2} s_j.lux)/(\frac{1}{d^2}) = s_j.lux$. Hence, for both *Case-1* and *Case-2*, $I_{g_i}^{(pred)} = s_j.lux$. □

Theorem 2. *Suppose, η is the distance between the closest pair of samples. Now, if grid dimension $n < \eta/\sqrt{2}$, then each grid $g_i \in \mathcal{G}$ contains at most one sample.*

Proof. Given, η is the distance between the closest pair of samples (say s_j and s_k). Suppose n' is the dimension of smallest possible grid (i.e., grid of size $n' \times n'$) that can contain s_j and s_k. In this case, the sampling locations $s_j.loc$ and $s_k.loc$ of the samples s_j and s_k must be at two endpoints of any diagonal of the $n' \times n'$ grid. Hence, $\eta = \sqrt{2} \times n'$, i.e., $n' = \eta/\sqrt{2}$. Now, any grid of dimension n smaller than n' will contain at most one sample. In other words, if $n < n' = \eta/\sqrt{2}$, then each grid can at most one sample. □

Theorem 3. *If the parameter K = 1, then the predicted grid illumination level of g_i will be the highest lux value among all the samples that reside within g_i.*

Proof. According to SIM-IDS (Algorithm 3), $S_i' = \langle s_i^1, s_i^2, s_i^3, \ldots, s_i^t \rangle$ is the sorted list of samples that resides in $g_i \in \mathcal{G}'$ such that $s_i^1.lux \geq s_i^2.lux \geq \cdots \geq s_i^t.lux$. Now, if K = 1, then $L_i^k = \{s_i^1\}$. Note that $s_i^1.lux$ is the maximum among all the samples, i.e., $s_i^1.lux = Max(s_i^1.lux, s_i^2.lux, \ldots, s_i^t.lux)$. Now two cases may occur. *Case-1:* If $d_{i1} = 0$, then predicted grid illumination level $I_{g_i}^{(pred)} = s_i^1.lux$. *Case-2:* If $d_{i1} > 0$, then $D_i^k = \{d_{i1}\}$. Eventually, $I_{g_i}^{(pred)} = (\frac{1}{d_{i1}^2} s_i^1.lux)/(\frac{1}{d_{i1}^2}) = s_i^1.lux$. Hence, for both *Case-1* and *Case-2*, $I_{g_i}^{(pred)} = s_i^1.lux$, which is the highest lux value among all the samples that resides in g_i. □

7 EVALUATION AND RESULTS

In this section, the experiments are conducted to evaluate the performance of proposed illumination monitoring and mapping algorithm SIM. SIM is implemented in Python 3.6 (https://www.python.org). All the experiments are performed on a 64-bit Windows PC with 1.8 GHz Intel Core i7 CPU and 16 GB of RAM.

7.1 Study Area and Dataset

The model SIM is evaluated on two real-world participatory sensing-based illumination level datasets. The first dataset is created by the participants for the Jadavpur University area of city Kolkata. Here, data are collected along the following route: *"Prayukti Bhavan → Aurobindo Bhavan → JU Central Library → JU Gate No 3 → 8B Bus Stand → JU Gate No 1 → Jadavpur Rail Station."* This dataset contains 3,128 illumination level data samples. The other dataset is created by the participants for the Baruipur area of city Kolkata. Here, data are collected along the following route: *"Baruipur Rail Station → Baruipur Police Station → Baruipur Rail Gate → Baruipur*

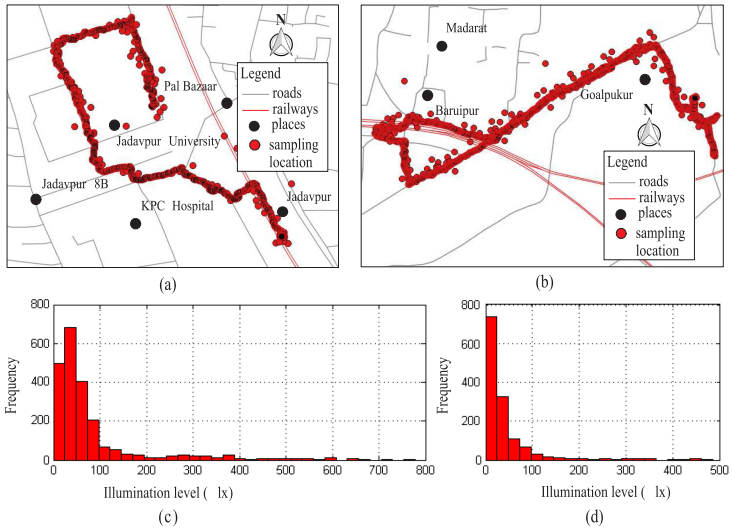


Fig. 5. Sampling locations in (a) Jadavpur area and (b) Baruipur area; distribution of Illumination levels in (a) Jadavpur area and (b) Baruipur area.

Hospital → Goalpukur." This second dataset contains a total of 2,232 data samples. Note that, for ground-truth illumination data collection, we have used MT-912 Light Meter.[1] The road network data are collected from Open Street Map [26]. Figures 5(a) and 5(b) show the sampling locations of two study areas Jadavpur University area and Baruipur area, respectively.

7.2 Baselines

To the best of our knowledge, there is no existing work that uses predictive models for crowd/participatory sensing-based city illumination mapping. Hence, we have employed some of the models as baselines that are used in the existing literature for mapping tasks in other city dynamics. More specifically, to evaluate the usefulness of the proposed method SIM, it is compared with six baselines as given below.

- *Spatial kNN:* The method Spatial kNN [33] predicts the illumination level at a grid midpoint by averaging lux values of some spatially nearest data samples. It can be defined as follows:

$$I_{g_i}^{(pred)} = \frac{1}{Q} \sum_{i=1}^{Q} s_i.lux, \tag{8}$$

  where Q is the number of spatially nearest data samples, $I_{g_i}^{(pred)}$ is the predicted lux value at the midpoint of g_i, and $s_i.lux$ denotes the spatially ith closest sample where $i \leq Q$.
- *Inverse Distance Weighting (IDW):* IDW [20] predicts the illumination level at a grid midpoint using a weighted average of the lux values from the observed data samples. Here, the weights are inversely proportional to the spatial distances. The estimation in IDW is expressed as follows:

$$I_{g_i}^{(pred)} = \frac{\sum_{i=1}^{N} W_i \times S_i.lux}{\sum_{i=1}^{N} W_i}, \tag{9}$$

[1] https://www.urceri.com/mt-912-light-meter.html.

where, $W_i = 1/(d_i)^p$ is the weighting function, d_i is the distance between the grid midpoint m_i and the sampling location, and $p \in \mathbb{R}_+$ is the power parameter.

- *Linear Regression (LR):* Linear regression [28] using the least-squares method is utilized as a baseline. It is the most basic and widely used form of regression.
- *Random Forest Regressor (RFR):* RFR [5] is chosen as another baseline. The RFR method simply uses all the geo-tagged illumination sample data from all the participants for training to build a model.
- *Support Vector Regressor (SVR):* The regression version of **Support Vector Machine (SVM)** [11, 22] is utilized to estimate lux value at grid midpoints. Similar to RFR, SVR uses the geo-tagged illumination sample data from all the participants to build a model and then predict illumination levels at unknown grid midpoints.
- *Artificial Neural Network (ANN):* ANN [21] with back-propagation is also considered as a baseline method. The ANN model, which is built for comparison, has a single hidden layer.

7.3 Evaluation Metrics and Parameter Setting

The proposed method is evaluated using two evaluation metrics, namely, **mean absolute error (MAE)** and RMSE. MAE and RMSE are defined as follows:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|E(x_i) - \widehat{E}(x_i)|, \tag{10}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\{E(x_i) - \widehat{E}(x_i)\}^2}, \tag{11}$$

where N is the number of data samples in test set, $\widehat{E}(x_i)$ and $E(x_i)$ denote the estimated and the ground-truth value of illumination level, respectively. To find the best setting of the model parameters K and n, the performance of SIM is evaluated with different values of K and n. The best performance is achieved for the following setting: $K = 10$ and $n = 20$.

7.4 Results and Discussion

To assess the effectiveness of the proposed model SIM, it is evaluated on participatory sensing-based real-world illumination level datasets (see Section 7.1). SIM is also compared with six aforementioned baselines, namely, Spatial kNN [33], IDW [20], LR [28], RFR [5], SVR [11, 22], and ANN [21]. A summary of the models' performance in terms of two performance metrics RMSE and MAE is provided in Table 4. A 10-fold **cross validation (CV)** technique is used for evaluating the performance of the models. In each iteration of CV, a different set of grids (that overlap with the road network) is considered as an unobserved set of grids and is used as a validation set. Since we are following the CV approach, each of the grids gets a chance to be in the unobserved set of grids (i.e., in the validation set). The results indicate that SIM dominates over the baselines in terms of mean predictive accuracy (RMSE = 14.65, MAE = 6.51). It is found that if DBSCAN is not used in SIM, then the error in prediction (RMSE = 17.33, MAE = 8.42) increases. In addition, Table 4 shows that in the case of SIM, the uncertainties in the accuracy (expressed in terms of standard deviation) is relatively less. It's worth noting that the accuracy of a model can vary in different runs depending on the distribution of samples in 10 different folds of CV. Hence, accuracy is computed here in multiple runs (in our case 100 runs) to assess the consistency in prediction performance. For instance, if $\{a_1, a_2, a_3, \ldots, a_{100}\}$ denotes the set of prediction accuracies in 100 different runs, then we present the overall accuracy of the model in a form $(x \pm y)$, where x and y denote the mean and **standard deviation (SD)** of the set of values. Here, SD, i.e., y helps us to measure the amount

Table 4. Performance Comparison with Baselines

Model	Performance of Models in percent (mean ± sd)	
	RMSE	MAE
Spatial kNN [33]	19.24 ± 1.40	09.94 ± 0.58
IDW [20]	61.10 ± 9.03	33.44 ± 3.88
LR [28]	29.67 ± 1.86	20.03 ± 0.85
RFR [5]	24.36 ± 2.08	15.12 ± 1.42
SVR [11, 22]	26.38 ± 2.14	12.54 ± 0.99
ANN [21]	27.77 ± 2.52	17.38 ± 1.38
SIM	**14.65 ± 1.35**	**06.51 ± 1.19**

kNN: k Nearest Neighbor; IDW: Inverse Distance Weighting; LR: Linear Regression; RFR: Random Forest Regression; SVR: Support Vector Regressor; ANN: Artificial Neural Network.

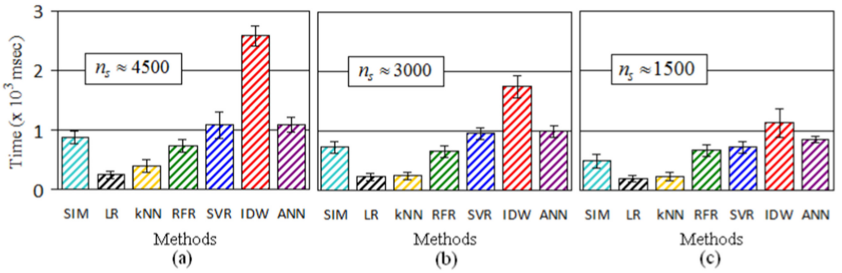


Fig. 6. Execution time of the methods by varying total number of data samples (a) $n_s \approx 4{,}500$, (b) $n_s \approx 3{,}000$, and (c) $n_s \approx 1{,}500$.

of variation in the prediction accuracies from the mean accuracy x in 100 runs. Note that, in this work, this variation in prediction accuracy is considered as the uncertainly in the in the prediction performance. High uncertainty in accuracy denotes that the prediction accuracy can vary largely from the mean, whereas low uncertainty denotes that the variation in accuracy across runs is less. Figures 6(a)–6(c) compare the execution time of SIM with various baselines by varying the number of illumination samples ($n_s \approx 4{,}500, n_s \approx 3{,}000, n_s \approx 1{,}500$). Here, the execution time of IDW is found to be maximum compared to the execution time of other approaches.

Figures 7(a) and 7(b) show the grid-based illumination mapping along two different paths in Jadavpur (an urban area) and Baruipur (semi-urban area), respectively. The illumination mapping in Figure 7(a) indicates that the illumination levels are not uniform along the path. However, in Figure 7(b) illumination levels are much more uniform along the path. Also, it is clearly observed that the illumination levels along the path in Jadavpur area (Figure 7(a)) are relatively high compared to the illumination levels along the path in Baruipur area (Figure 7(b)). It is because the Jadavpur area has better lighting infrastructure compared to the Baruipur area.

Figure 8 shows how the error of prediction ($|O_i - I_{g_i}^{(pred)}|$) depends on the support of prediction $S_i^{(pred)}$ for some randomly selected grid $g_i \in \mathcal{G}'$, where O_i and $I_{g_i}^{(pred)}$ are the observed and the predicted illumination level at $g_i \in \mathcal{G}'$, respectively. It is observed that the error in prediction for the grids decreases with the increase of the support of the prediction. It is because a high support value indicates a high density of samples that leads to a better estimation of the grid illumination level.

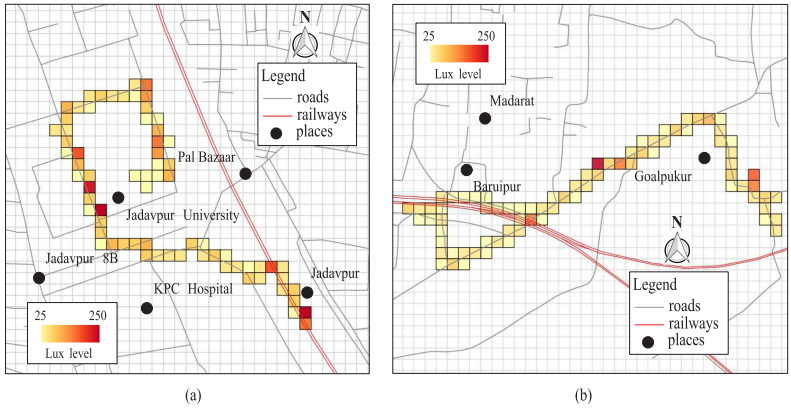


Fig. 7. Lux map for two paths (a) Jadavpur area: *"Prayukti Bhavan → Aurobindo Bhavan → JU Central Library → JU Gate No 3 → 8B Bus Stand → JU Gate No 1 → Jadavpur Rail Station."* (b) Baruipur area: *"Baruipur Rail Station → Baruipur Police Station → Baruipur Rail Gate → Baruipur Hospital → Goalpukur."*

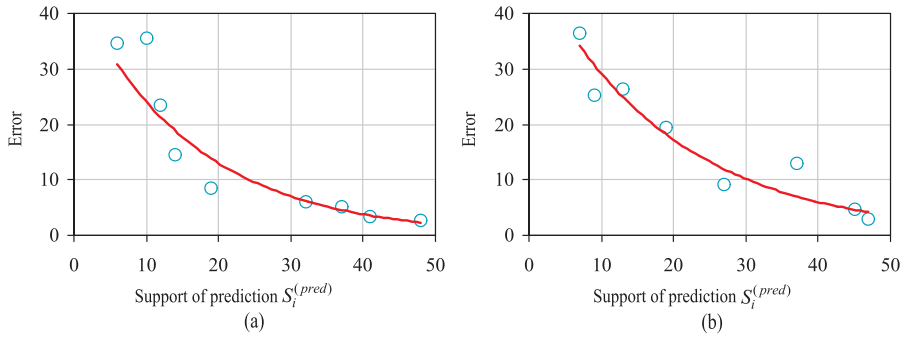


Fig. 8. Support of prediction ($S_i^{(pred)}$) versus error of prediction (i.e., $|O_i - I_{g_i}^{(pred)}|$) plots for some randomly selected grids $g_i \in \mathcal{G}'$ in (a) Jadavpur area and (b) Baruipur area.

7.5 Impact of k and n

In the SIM-IDS phase (Algorithm 3), for each grid $g_i \in \mathcal{G}'$ of dimension $n \times n$, the first k elements are selected from the set L_i^k. The performance of SIM depends on the values of k and n. The box plots in Figures 9(a) and 9(b) show the RMSEs of SIM for different settings. The best performance is achieved for the setting $n = 20, k = 10$. In Figure 9(a), grid dimension n is fixed to 20 and k varies from $5 \leq k \leq 25$. However, in Figure 9(b), k is fixed to 10 and n varies from $10 \leq k \leq 50$. If k or n is very small (Theorems 2 and 3), then the accuracy becomes very less (high RMSE) due to an insufficient number of samples within the grids. However, if k or n is large, then the prediction accuracy is also less. It is because when n is large, there can be a lot of samples that are far away from the grid midpoints, eventually they do not represent the illumination levels of grid midpoints. Now, a large value of k may allow many samples for which the phone is improperly aligned with the light source [17]. A proper alignment indicates that the smartphone is perpendicular to the light rays of the light source. A proper alignment of smartphone with the light source collects most lights and provides the highest lux reading. However, an improper alignment will decrease the lux readings. Eventually, it reduces the prediction accuracy.

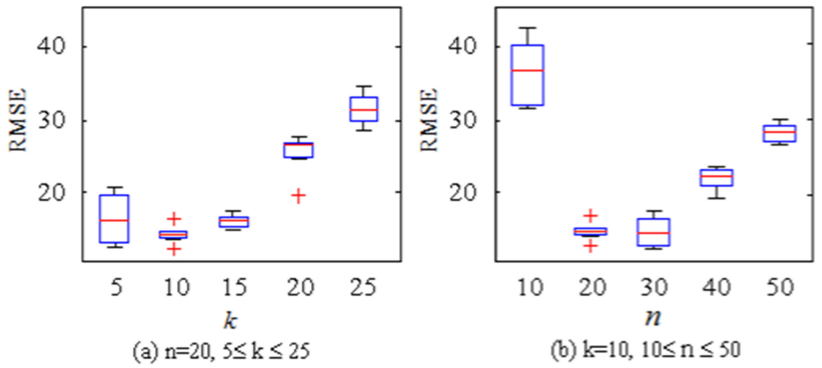


Fig. 9. Impact of two parameters on the performance (in terms of RMSE) of SIM (a) k and (b) grid dimension n.

8 CONCLUSION

In this article, we have proposed a participatory sensing-based illumination level monitoring and mapping system, called *CityLightSense*. A Street Illumination Mapping algorithm (SIM) is proposed for illumination mapping of city streets. The performance of the proposed algorithm SIM is evaluated on real-world participatory sensing-based illumination levels dataset collected by participants over a period of 1 month in two different urban areas (Jadavpur area and Baruipur area) of the city Kolkata.

The proposed model achieves better accuracy (RMSE = 14.65 ± 1.35, MAE = 6.51 ± 1.19) compared to the baseline methods (namely, spatial kNN, IDW, LR, RFR, SVR, and ANN). The evaluation results also validated that the proposed model is computationally efficient. In this article, the illumination mapping with participatory sensing data collected using CityLightSense application demonstrates its applicability in a real-world setting. The illumination maps have the potential to provide the citizens with a fine-grained spatial visualization of illumination mapping along the city streets.

It is important to note that the proposed system may have some limitations as follows. First, the article uses a participatory sensing approach for illumination level sensing that can not ensure the spatial uniformity of data samples. The performance of the illumination mapping algorithm may be negatively affected by non-uniform data samples. Second, the mobility pattern of the participants affects participatory sensing-based data collection. In this context, there can be a lack of data samples on some streets of the city that are not covered by many participants. Third, there may be some effect on the illumination levels of city streets due to the presence of buildings and tree covers. However, this article does not consider these factors during illumination mapping. Fourth, in crowd/participatory sensing based illumination monitoring, the participants may suffer from several potential issues such as costs due to sensing and uploading data and privacy issues due to revealing sensitive information (e.g., GPS location of the participants). These issues may demotivate the users to actively participate in the sensing tasks. Fifth, designing a good incentive mechanism for crowd/participatory sensing-based illumination data collection may also be considered as an important issue.

The above-mentioned limitations can be addressed in the future to improve the system further. Moreover, as our future work, we will try to come up with a dynamic city illumination map generation technique with high spatio-temporal granularity.

ACKNOWLEDGMENTS

We thank all the volunteers who have participated in the data collection process.

REFERENCES

[1] Munshi Yusuf Alam, Shahrukh Imam, Harshit Anurag, Sujoy Saha, Subrata Nandi, and Mousumi Saha. 2018. LiSense: Monitoring city street lighting during night using smartphone sensors. In *Proceedings of the IEEE International Conference on Data Mining (ICDMW'18)*. IEEE, 596–603.

[2] Chris Albon. 2020. Feature Selection Using Random Forest. Retrieved from https://chrisalbon.com/machine_learning/trees_and_forests/feature_selection_using_random_forest/.

[3] Arcgis. 2020. Inverse distance squared weighted interpolation. Retrieved from https://pro.arcgis.com/en/pro-app/help/analysis/geostatistical-analyst/how-inverse-distance-weighted-interpolation-works.htm.

[4] Peter R. Boyce, Neil H. Eklund, Barbara J. Hamilton, and Lisa D. Bruno. 2000. Perceptions of safety at night in different lighting conditions. *Int. J. Light. Res. Technol.* 32, 2 (2000), 79–91.

[5] Rodrigo Bustamante, Cathy Dichmont, Nick Ellis, Wayne Rochester, Shane Griffiths, Peter Rothlisberg, Michele Burford, Quinton Dell, Mark Tonks, Hector Lozano-Montes, et al. 2011. Effects of trawling on the benthos and biodiversity: Development and delivery of a spatially-explicit management framework for the northern prawn fishery. CSIRO Marine and Atmospheric Research. https://publications.csiro.au/publications/publication/PIcsiro:EP112693.

[6] Andrea Capponi, Claudio Fiandrino, Burak Kantarci, Luca Foschini, Dzmitry Kliazovich, and Pascal Bouvry. 2019. A survey on mobile crowdsensing systems: Challenges, solutions, and opportunities. *IEEE Commun. Surveys Tutor.* 21, 3 (2019), 2419–2465.

[7] Baidik Chandra, Asif Iqbal Middya, and Sarbani Roy. 2021. Spatio-temporal prediction of noise pollution using participatory sensing. In *Emerging Technologies in Data Mining and Information Security*. Springer, 597–607.

[8] Yanqing Cui, Jan Chipchase, and Fumiko Ichikawa. 2007. A cross culture study on phone carrying and physical personalization. In *Proceedings of the International Conference on Usability and Internationalization*. Springer, 483–492.

[9] Joy Dutta, Chandreyee Chowdhury, Sarbani Roy, Asif Iqbal Middya, and Firoj Gazi. 2017. Towards smart city: Sensing air quality in city based on opportunistic crowd-sensing. In *Proceedings of the 18th International Conference on Distributed Computing and Networking*. 1–6.

[10] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD'96)*. 226–231.

[11] Nicolas Gilardi. 2002. *Machine Learning for Spatial Data Analysis*. Ph.D. Dissertation. Verlag nicht ermittelbar.

[12] Jose-Maria Gutierrez-Martinez, Ana Castillo-Martinez, Jose-Amelio Medina-Merodio, Juan Aguado-Delgado, and Jose-Javier Martinez-Herraiz. 2017. Smartphones as a light measurement tool: Case of study. *Appl. Sci.* 7, 6 (2017), 616.

[13] Antal Haans and Yvonne A. W. De Kort. 2012. Light distribution in dynamic street lighting: Two experimental studies on its effects on perceived safety, prospect, concealment, and escape. *J. Environ. Psychol.* 32, 4 (2012), 342–352.

[14] Xin-Ming Huang, Jing Ma, and Lawrence E. Leblanc. 2004. Wireless sensor network for streetlight monitoring and control. In *Digital Wireless Communications VI*, Vol. 5440. International Society for Optics and Photonics, 313–321.

[15] Chunguo Jing, Dongmei Shu, and Deying Gu. 2007. Design of streetlight monitoring and control system based on wireless sensor networks. In *Proceedings of the 2nd IEEE Conference on Industrial Electronics and Applications*. IEEE, 57–62.

[16] Devroop Kar, Asif Iqbal Middya, and Sarbani Roy. 2019. An approach to detect travel patterns using smartphone sensing. In *Proceedings of the IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS'19)*. IEEE, 1–6.

[17] Krukarius. 2020. Light around us and how to measure it. Retrieved from http://www.mkrgeo-blog.com/light-around-us-and-how-to-measure-it/.

[18] Sumeet Kumar, Ajay Deshpande, Stephen S. Ho, Jason S. Ku, and Sanjay E. Sarma. 2016. Urban street lighting infrastructure monitoring using a mobile sensor platform. *IEEE Sensors J.* 16, 12 (2016), 4981–4994.

[19] Huang-Chen Lee and Huang-Bin Huang. 2014. A low-cost and noninvasive system for the measurement and detection of faulty streetlights. *IEEE Trans. Instrument. Measure.* 64, 4 (2014), 1019–1031.

[20] Jin Li and Andrew D. Heap. 2011. A review of comparative studies of spatial interpolation methods in environmental sciences: Performance and impact factors. *Ecol. Inform.* 6, 3–4 (2011), 228–241.

[21] Jin Li and Andrew D. Heap. 2014. Spatial interpolation methods applied in the environmental sciences: A review. *Environ. Model. Softw.* 53 (2014), 173–189.

[22] Jin Li, Andrew D. Heap, Anna Potter, Zhi Huang, and James J. Daniell. 2011. Can we improve the spatial predictions of seabed sediments? A case study of spatial interpolation of mud content across the southwest Australian margin. *Continent. Shelf Res.* 31, 13 (2011), 1365–1376.

[23] Asif Iqbal Middya and Sarbani Roy. 2021. Spatial interpolation techniques on participatory sensing data. *ACM Trans. Spatial Algor. Syst.* 7, 3 (2021), 1–32.

[24] Asif Iqbal Middya, Sarbani Roy, Joy Dutta, and Rituparna Das. 2020. JUSense: A unified framework for participatory-based urban sensing system. *Mobile Netw. Appl.* 25, 4 (May 2020), 1249–1274. https://doi.org/10.1007/s11036-020-01539-x

[25] Asif Iqbal Middya, Sarbani Roy, Saptarshi Mandal, and Rahul Talukdar. 2021. Privacy protected user identification using deep learning for smartphone-based participatory sensing applications. *Neural Comput. Appl.* (July 2021). https://doi.org/10.1007/s00521-021-06319-6

[26] OSM. 2020. Open Street Map. Retrieved from https://bit.ly/3zmjCTF.

[27] Susmita Patra, Asif Iqbal Middya, and Sarbani Roy. 2021. PotSpot: Participatory sensing based monitoring system for pothole detection using deep learning. *Multimedia Tools Appl.* 80, 16 (Apr. 2021), 25171–25195. https://doi.org/10.1007/s11042-021-10874-4

[28] Scikit-learn. 2020. Linear Regression. Retrieved from https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html.

[29] P. Singh and P. A. Meshram. 2017. Survey of density based clustering algorithms and its variants. In *Proceedings of the International Conference on Inventive Computing and Informatics (ICICI'17)*. 920–926.

[30] Dhiraj Sunehra and Sangem Rajasri. 2017. Automatic street light control system using wireless sensor networks. In *Proceedings of the IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI'17)*. IEEE, 2915–2919.

[31] Matti T. Vaaja, Matti Kurkela, Mikko Maksimainen, Juho-Pekka Virtanen, Antero Kukko, Ville V. Lehtola, Juha Hyyppä, Hannu Hyyppä, et al. 2018. Mobile mapping of night-time road environment lighting conditions. *Photogram. J. Finland* 26 (2018), 1–17.

[32] F. Van Diggen. 1999. Gps accuracy: Lies, damn lies and statistics. *GPS World* 1 (1999), 41.

[33] Jay M. Ver Hoef and Hailemariam Temesgen. 2013. A comparison of the spatial linear model to nearest neighbor (k-NN) methods for forestry applications. *PloS One* 8, 3 (2013), e59129.

[34] Jing Yuan, Yu Zheng, Chengyang Zhang, Xing Xie, and Guang-Zhong Sun. 2010. An interactive-voting based map matching algorithm. In *Proceedings of the 11th International Conference on Mobile Data Management*. IEEE, 43–52.

[35] Ashraf Zatari, Gordon Dodds, Karen McMenemy, and Richard Robinson. 2005. Glare, luminance, and illuminance measurements of road lighting using vehicle mounted CCD cameras. *Leukos* 1, 2 (2005), 85–106.

[36] Yu Zheng, Licia Capra, Ouri Wolfson, and Hai Yang. 2014. Urban computing: Concepts, methodologies, and applications. *ACM Trans. Intell. Syst. Technol.* 5, 3 (2014), 1–55.

Received January 2021; revised July 2021; accepted September 2021