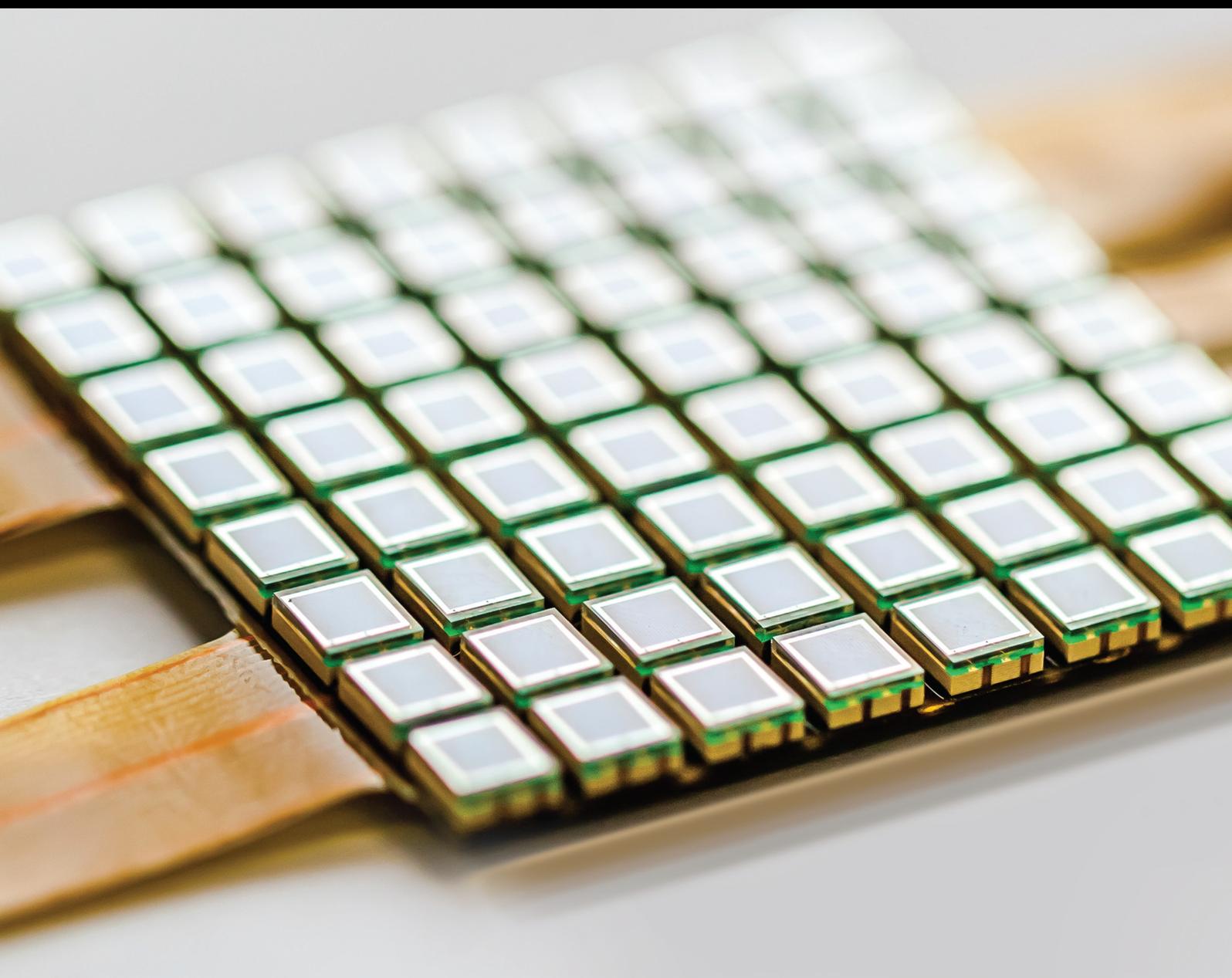


Deep Perception beyond the Visible Spectrum: Sensing, Algorithms, and Systems

Lead Guest Editor: Sidike Paheding

Guest Editors: Riad I. Hammoud and Vijayan K. Asari



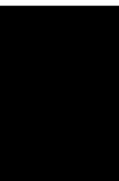


Deep Perception beyond the Visible Spectrum: Sensing, Algorithms, and Systems

**Deep Perception beyond the Visible
Spectrum: Sensing, Algorithms, and
Systems**

Lead Guest Editor: Sidike Paheding

Guest Editors: Riad I. Hammoud and Vijayan K.
Asari



Copyright © 2020 Hindawi Limited. All rights reserved.

This is a special issue published in "Journal of Sensors." All articles are open access articles distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Chief Editor

Harith Ahmad , Malaysia

Associate Editors

Duo Lin , China
Fanli Meng , China
Pietro Siciliano , Italy
Guiyun Tian, United Kingdom

Academic Editors

Ghufran Ahmed , Pakistan
Constantin Apetrei, Romania
Shonak Bansal , India
Fernando Benito-Lopez , Spain
Romeo Bernini , Italy
Shekhar Bhansali, USA
Matthew Brodie, Australia
Ravikumar CV, India
Belén Calvo, Spain
Stefania Campopiano , Italy
Binghua Cao , China
Domenico Caputo, Italy
Sara Casciati, Italy
Gabriele Cazzulani , Italy
Chi Chiu Chan, Singapore
Sushank Chaudhary , Thailand
Edmon Chehura , United Kingdom
Marvin H Cheng , USA
Lei Chu , USA
Mario Collotta , Italy
Marco Consales , Italy
Jesus Corres , Spain
Andrea Cusano, Italy
Egidio De Benedetto , Italy
Luca De Stefano , Italy
Manel Del Valle , Spain
Franz L. Dickert, Austria
Giovanni Diraco, Italy
Maria de Fátima Domingues , Portugal
Nicola Donato , Italy
Sheng Du , China
Amir Elzwawy, Egypt
Mauro Epifani , Italy
Congbin Fan , China
Lihang Feng, China
Vittorio Ferrari , Italy
Luca Francioso, Italy

Libo Gao , China
Carmine Granata , Italy
Pramod Kumar Gupta , USA
Mohammad Haider , USA
Agustin Herrera-May , Mexico
María del Carmen Horrillo, Spain
Evangelos Hristoforou , Greece
Grazia Iadarola , Italy
Syed K. Islam , USA
Stephen James , United Kingdom
Sana Ullah Jan, United Kingdom
Bruno C. Janegitz , Brazil
Hai-Feng Ji , USA
Shouyong Jiang, United Kingdom
Roshan Prakash Joseph, USA
Niravkumar Joshi, USA
Rajesh Kaluri , India
Sang Sub Kim , Republic of Korea
Dr. Rajkishor Kumar, India
Rahul Kumar , India
Nageswara Lalam , USA
Antonio Lazaro , Spain
Chengkuo Lee , Singapore
Chenzong Li , USA
Zhi Lian , Australia
Rosalba Liguori , Italy
Sangsoon Lim , Republic of Korea
Huan Liu , China
Jin Liu , China
Eduard Llobet , Spain
Jaime Lloret , Spain
Mohamed Louzazni, Morocco
Jesús Lozano , Spain
Oleg Lupan , Moldova
Leandro Maio , Italy
Pawel Malinowski , Poland
Carlos Marques , Portugal
Eugenio Martinelli , Italy
Antonio Martinez-Olmos , Spain
Giuseppe Maruccio , Italy
Yasuko Y. Maruo, Japan
Zahid Mehmood , Pakistan
Carlos Michel , Mexico
Stephen. J. Mihailov , Canada
Bikash Nakarmi, China

Ehsan Namaziandost , Iran
Heinz C. Neitzert , Italy
Sing Kiong Nguang , New Zealand
Calogero M. Oddo , Italy
Tinghui Ouyang, Japan
SANDEEP KUMAR PALANISWAMY ,
India
Alberto J. Palma , Spain
Davide Palumbo , Italy
Abinash Panda , India
Roberto Paolesse , Italy
Akhilesh Pathak , Thailand
Giovanni Pau , Italy
Giorgio Pennazza , Italy
Michele Penza , Italy
Sivakumar Poruran, India
Stelios Potirakis , Greece
Biswajeet Pradhan , Malaysia
Giuseppe Quero , Italy
Linesh Raja , India
Maheswar Rajagopal , India
Valerie Renaudin , France
Armando Ricciardi , Italy
Christos Riziotis , Greece
Ruthber Rodriguez Serrezuela , Colombia
Maria Luz Rodriguez-Mendez , Spain
Jerome Rossignol , France
Maheswaran S, India
Ylias Sabri , Australia
Sourabh Sahu , India
José P. Santos , Spain
Sina Sareh, United Kingdom
Isabel Sayago , Spain
Andreas Schütze , Germany
Praveen K. Sekhar , USA
Sandra Sendra, Spain
Sandeep Sharma, India
Sunil Kumar Singh Singh , India
Yadvendra Singh , USA
Afaque Manzoor Soomro , Pakistan
Vincenzo Spagnolo, Italy
Kathiravan Srinivasan , India
Sachin K. Srivastava , India
Stefano Stassi , Italy

Danfeng Sun, China
Ashok Sundramoorthy, India
Salvatore Surdo , Italy
Roshan Thotagamuge , Sri Lanka
Guiyun Tian , United Kingdom
Sri Ramulu Torati , USA
Abdellah Touhafi , Belgium
Hoang Vinh Tran , Vietnam
Aitor Urrutia , Spain
Hana Vaisocherova - Lislalova , Czech
Republic
Everardo Vargas-Rodriguez , Mexico
Xavier Vilanova , Spain
Stanislav Vitek , Czech Republic
Luca Vollero , Italy
Tomasz Wandowski , Poland
Bohui Wang, China
Qihao Weng, USA
Penghai Wu , China
Qiang Wu, United Kingdom
Yuedong Xie , China
Chen Yang , China
Jiachen Yang , China
Nitesh Yelve , India
Aijun Yin, China
Chouki Zerrouki , France

Contents

Using Artificial Intelligence Techniques to Improve the Prediction of Copper Recovery by Leaching

Victor Flores , Brian Keith, and Claudio Leiva 

Research Article (12 pages), Article ID 2454875, Volume 2020 (2020)

Location Fixing and Fingerprint Matching Fingerprint Map Construction for Indoor Localization

Marwan Alakhras , Mousa Hussein , and Mourad Oussalah 

Research Article (14 pages), Article ID 7801752, Volume 2020 (2020)

Improved Capacitive Proximity Detection for Conductive Objects through Target Profile Estimation

Fan Xia, Umme Zakia, Carlo Menon, and Behraad Bahreyni 

Research Article (11 pages), Article ID 3891350, Volume 2019 (2019)

Heterogeneous Gray-Temperature Fusion-Based Deep Learning Architecture for Far Infrared Small Target Detection

Junhwan Ryu  and Sungho Kim 

Research Article (15 pages), Article ID 4658068, Volume 2019 (2019)

Vision-Based Deep Q-Learning Network Models to Predict Particulate Matter Concentration Levels Using Temporal Digital Image Data

SungHwan Kim , SeHee Jung , SeongMin Yang , JiSeong Han, ByungYong Lee, JaeHwa Lee , and Sung Won Han 

Research Article (10 pages), Article ID 9673047, Volume 2019 (2019)

Research Article

Using Artificial Intelligence Techniques to Improve the Prediction of Copper Recovery by Leaching

Victor Flores ¹, Brian Keith,¹ and Claudio Leiva ²

¹Department of Computing & Systems Engineering, Universidad Católica del Norte, 1270709 Antofagasta, Chile

²Department of Chemical Engineering, Universidad Católica del Norte, 1270709 Antofagasta, Chile

Correspondence should be addressed to Victor Flores; vflores@ucn.cl

Received 21 June 2019; Revised 19 November 2019; Accepted 3 January 2020; Published 7 February 2020

Guest Editor: Sidike Paheding

Copyright © 2020 Victor Flores et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Copper mining activity is going through big changes due to increasing technological development in the area and the influence of industry 4.0. These changes, produced by technological context and more controls (e.g., environmental controls), are also becoming visible in Chilean mining. New regulations from the Chilean government and changes in the copper mining industry (such as a trend to underground mining) are fostering the search for better results in typical processes such as leaching. This paper describes an experience using artificial intelligence techniques, particularly random forest, to develop predictive models for copper recovery by leaching, using data from an enterprise present in northern Chile for more than 20 years. Two models, one of them with actual operational data and another one with data generated in a controlled environment (piling) are presented. Well-classified values of 98.90% for operational data and 98.72% for pile/piling data were obtained. The methodology devised for the study can be transferred to piling columns or piles with other characteristics, though the operation must focus on copper leaching. It can even be transferred to other leaching processes using another type of mineral, with proper adjustments.

1. Introduction

The Chilean mining industry, as in the whole world, is experimenting with big changes due to the rapid technological advance in the so-called industry 4.0 [1]. According to Pietrobelli et al. in [2], big mining companies typically tended to control their operations from remote centers located in multinational corporations, thus resulting in little local innovation and development. This way of operating helps the macroeconomy, but it makes difficult diversification, knowledge transfer, and regional innovation in the value chain [3]. Another factor producing changes in the above-mentioned trend is the significant fall of copper price since 2015, fostering both technological advances enabling companies to face production costs [4] and also greater regional innovation and development.

Chilean copper production represents 35% on a world basis [5, 6]. On a local basis, the copper production industry is the country's most profitable, providing almost 15% of Chilean GDP and representing 50% of exports [7, 8]. This Chilean predominant position in the copper industry is also

complemented with leadership in other mineral products, such as lithium. To keep this leadership in the world's mining activity, Chile must ensure mining profitability in the short term. A valid strategy for this may be investing in technology and innovation, together with mining industry diversification.

Recent papers [5–7] report a trend to technological diversification in the sector, even from mining suppliers. Furthermore, as stated in [6], a recent report from the Chilean government declares the objective of promoting the establishment of 250 local suppliers for the mining sector in 2035. This strategy is expected to create knowledge about business and technology appropriate for current challenges, both elements being directed to local mining development and exports as well. This would result in an income of about US\$10 thousand million.

For the aforementioned technological development and innovation, the Chilean mining industry is incorporating technology to develop intelligence system-type applications for supporting tasks such as copper recovery prediction. These systems are frequently based on artificial intelligence

computing models. Apart from representing a technological contribution, these models are becoming a great help for predicting or reducing production costs [9, 10], a very convenient fact for supporting modern technology characterized by a greater extraction complexity and increasing restrictions such as environmental ones [11].

In typical production processes such as leaching, predictive models have been satisfactorily used in the last decade to identify factors allowing production increase [9, 10]. There are several cases illustrating predictive model generation using artificial intelligence, specifically soft computing [12]. In particular, this paper fully describes the process for developing predictive models [13] to recover copper by leaching and the results were obtained at SCM Franke Company, from the KGHM International Group, present in Chilean mining exploitation since 2009.

Recently, research into the applicability of artificial intelligence techniques such as predictive model algorithms, for copper recovery prediction, has been conducted. In this context, comparative studies of which predictive model algorithms are the most appropriate according to the characteristics of the copper mining production process have been published. Thus, advantages of using support vector machine (SVM), random forest (RF), artificial neural networks (ANN), gradient boosted trees (GBT), or wavelet neural network (WNN) are frequently reported in the literature (such as [14]). For example, in [15], a predictive modeling using SVM for copper potential mapping in the Kerman copper bearing belt in the south of Iran is reported. In [16], a comparative analysis of ANN, WNN, and SVM models to mineral potential mapping for copper mineralization is presented. As a particular result of this work, the authors highlight that WNN exhibits excellent learning ability compared to the conventional ANN.

Also, in [17], SVM, ANN, and RF were used to conduct predictive modeling of mineral prospectively. For these algorithms, input data was obtained from GIS-based mineral prospectively mapping of the Tongling ore district (eastern China). As a conclusion from this work, authors highlight that the RF model outperformed the SVM and ANN models, giving a greater consistency and better predictive accuracy. Another example of comparative analysis of predictive models using GBT, ANNs, and RF is the work described in [16], where authors highlight that the RF models show the highest coefficients of determination (R^2) values and the lowest root-mean-square error (RMSE), and the highest residual prediction deviations (RPD) were obtained.

There are several papers that report that RF and GBDT perform the best (see Table 1 for a comparison among these methods); therefore, and based on the described information in the previous paragraphs, the use of RF can more appropriately lead to the achievement of the stated objective.

This paper describes the tasks done to generate predictive models for copper recovery in leaching piles with low-grade material, using data from actual pile operation and those produced in a controlled environment (pilot), using the same artificial intelligence technique (random forest technique) in both cases to develop predictive models.

TABLE 1: Accuracy ranking of different classification algorithms [32].

Algorithm	Accuracy ranking				
	Top 1	Top 2	Top 3	Top 4	Top 5
GBDT	42.25%	63.38%	67.61%	77.46%	83.10%
RF	21.13%	56.34%	77.46%	84.51%	92.96%
SVM	18.31%	30.99%	52.11%	64.79%	78.87%
ELM	15.49%	23.94%	30.99%	47.89%	56.34%
C4.5	12.68%	15.49%	28.17%	43.66%	56.34%
SRC	11.27%	22.54%	23.94%	33.80%	38.03%
LR	4.23%	11.27%	19.72%	26.76%	40.85%
AB	4.23%	7.04%	8.45%	19.72%	18.31%
KNN	2.82%	8.45%	12.68%	26.76%	36.62%
NB	2.82%	5.63%	7.04%	9.86%	19.72%
DL	1.41%	1.41%	2.82%	4.23%	4.23%

The remaining document is organized as follows: Section 2 describes the base concepts of the study and related work. Section 3 describes the experiment, the discretizing of the variables used in the model, data characteristics and how they were collected, work methodology, and the techniques used for analyzing results. Section 4 shows the results obtained for the two models, that is, operational data and piling data models. Section 5 deals with the discussion. Section 6 shows the conclusions of the paper. Finally, acknowledgments and bibliographical references are stated.

2. Concepts and Related Work

2.1. Leaching and Company Work. The copper leaching process involves tasks thoroughly identified by the industry, that is, irrigation beginning and maintenance, agglomerate condition evaluation, drainage distribution, pool solution inventory, PLS flow evaluation, and distribution and deposition of the material leached at the plant (harvest). These processes, due to the nature and variability of the input material, usually produce high levels of entropy and uncertainty (close to 20%) concerning copper recovery at the end of the harvest [9].

SCM Franke uses three industrial processes widely known in the industry of metallic copper production via hydrometallurgy. These processes are dynamic pile leaching, solvent extraction, and electrowinning [9]. The ultimate goal of these processes is to obtain the greatest copper production by saving resources and being the least possible aggressive to the environment (a kind of environmental trade-off). The leaching process has been shown to be one of the most convenient to achieve this environmental trade-off. The objective of this paper is predicting estimated copper recovery as accurately as possible at about 95% by dynamic pile leaching, using the least possible amount of leaching material and the best irrigation homogeneity.

2.2. Related Work and Predictive Models. The development of applications using predictive modeling to improve mineral

recovery estimation is prospectively becoming a central area of study in the mining industry [18–20].

Recent studies such as [3, 9, 10] reveal that one of the most critical tasks in prospective modeling is the selection of appropriate criteria and the application of sound innovating techniques to get the evidential characteristics of these criteria.

Traditionally, these criteria have been selected by different numerical methods, but in the last few decades, alternative techniques such as those from the artificial intelligence area have been applied for both criteria selection and the development of predictive models for mineral recovery [21]. In general, methods containing machine learning algorithms are being applied for building these predictive models.

In the literature, the methods referred to here have been grouped into two sets [21–23]: *knowledge-driven models* and *data-driven models*. Data-driven models are probabilistic models such as discriminant analysis or logistic regression [19, 24]. The algorithms of data-driven models, whose evidence of use is more often reported in the literature, are artificial neural networks (generally with backpropagation [25, 26]) (ANN) [19, 27, 28] and regression trees (RTs) [13, 24, 29] in sectors such as copper mining. Methods called support vector machines (SVM) and random forest (RF) [29] are sometimes used in this domain [13, 30, 31]. The common way of using the algorithms of the data-driven model group in concrete mining tasks such as studying copper recovery is using data themselves, while in knowledge-driven methods, an expert in mineral extraction via hydrometallurgy should be consulted for the job. As a whole, ANN, RT, or SVM models require enough amounts of records and parameters to achieve good quality in the models created as output.

The literature contains papers such as [32] that propose a comparison among the performance of predictive models. Table 1 shows that RF and GBDT perform the best, followed by SVM and ELM. Moreover, we observe that the interquartile ranges of RF, GBDT, and SVM are the smallest, showing that these three algorithms generally perform well, in terms of prediction accuracy, regardless of the datasets [32]. While ensemble and boosting methods have been reported to obtain good predictive performance in supervised learning, GBDT is generally less popular than RF. GBDT and RF show both best total average classification accuracy and best mean rank followed by SVM and ELM [32].

This study uses RF as a predictive model; it is a kind of predictive model based on decision trees. There are previous works as [33] that defined this kind of predictive models as “a type of predictive model that uses a decision tree to go from observations of an object (represented as the branches of a tree) to a certain conclusion about a target value of the object (represented by the tree leaves).”

Thus, the interest of using RF is twofold. First, data-driven model algorithms (like RF) are frequently used to predict values of the target variable influenced by other variables (predictor variables) in datasets [33, 34]. In this context, the RF model is adequate for generating a predictive model of the copper recovery by leaching (the target variable for this work), due to it providing a way to measure the influence

of each predictor variable on the target variable. And second, one of the main benefits of RF is that it can be used to determine the importance of variables in a regression or classification problem intuitively [35]. So, RF can be used to determine the importance of each predictive variable over the target variable.

Prediction is a highly interesting topic in machine learning, which is, in turn, one of the branches of artificial intelligence. As mentioned above, RF is based on decision trees (DT). DT have been widely used in areas such as medicine to yield a diagnosis since they are easy to interpret. Basically, DT is a hierarchical set of nodes (starting from a root node), where each node contains a decision based on the comparison of an attribute with a threshold value [36, 37]. DT-based learning goes from the observation of an object represented as branches of a tree to certain conclusions related to a target value of an object (represented by tree values) [36, 37].

Previous studies use artificial intelligence techniques for copper-related models. For example, in [8], a model based on fuzzy logic is reported to predict ground vibration and environmental impact due to blasts in the open-pit mine. For this model, the toolbox fuzzy logic of MATLAB was used. In [38], ANN was used to predict the copper ore flotation indices of separation efficiency within different operational conditions.

3. Materials and Methods

3.1. Experimental Description. Operational and piling data are available for attaining the objective set by SCM Franke company (environmental trade-off described in Section 2). The company keeps records of planning and copper recovery by heap leaching. These are called operational data (industrial operation). Work has also been done with data collected in a controlled environment. These data are known as piling data, which are the result of tests in leaching columns using strictly controlled measures on irrigation rates, acid concentration in irrigation solutions, and operational cycles.

For the specific case of this study, both operational and piling data were collected by two students in practice and Professor C. Leiva (students under the supervision of Professor C. Leiva, coauthor in this paper) all from the Chemical Department at the Universidad Católica del Norte, Chile. In a similar way to what worked in [9], the parameters of these data groups are fully described below:

- (i) Agglomerate is measured in mm, where 80% of the solids are below this value
- (ii) Irrigation rate (RL) ($L\text{ hr}/m^2$) is the surface flow of sulfuric acid in the pile
- (iii) H+fed (gpl) is the volumetric flow of ILS (intermediate liquid solution) recirculating in the pile
- (iv) The height of a pile is defined by the production goals expected to be accomplished; that is, the piled fine copper tonnage with which the production to be obtained will be determined

- (v) Total Cu grade (%) is the total copper percentage existing in the pile in the n day of operation
- (vi) CO_3 grade (%) is the carbonate percentage existing in the pile in the n day of operation
- (vii) Leaching ratio (m^3/TMS) is defined by the amount of sulfuric acid with respect to the total material to be leached
- (viii) Days of operation refer to the days elapsing from the pile starting up to the end of leaching
- (ix) Soluble Cu (%) is the percentage of copper soluble in sulfuric acid present in the pile in the n day of operation
- (x) Class R (%) is the percentage of leached copper in the n day with respect to the soluble copper present in the pile in day 1 of operation

3.2. Operational Data. Operational data were collected during time periods called leaching cycles emerging after soil piling and the beginning of the irrigation process since day 1 to the last day of harvest. The leaching cycles in the company are planned to last 65-70 days. Operational data were obtained with a frequency of 4 hours during one year. Due to the conditions of the process and operational decisions, the irrigation of some piles or modules in service was stopped, a fact that could render incongruent results when modeling the system. For this reason and with the purpose of avoiding unnecessary “noise” in the system, along with storing poor data for the statistic model, the records of the nonirrigation periods were deleted from the database.

3.3. Piling Data. Piling (or pilot plant) was conducted in two agglomerate tanks of the same dimensions with a material whose granulometry was less than 13 mm in diameter. The mineral was put in contact (irrigated) with a solution of sulfuric acid and water and refined to form lumps of fine material; this was made in order to give the mineral a proper uniform size for the leaching stage and also help copper sulfidation via contact with acid solutions. The aforementioned conditions vary according to leaching cycles to obtain piling scenarios as close as possible to actual pile mineral exploitation. Piling data were obtained in the same way as explained for operational data.

3.4. Random Forest. As previously mentioned, random forest (RF) is a predictive model based on decision tree (DT). The RF supervised learning algorithm is based on the machine learning theory which belongs to the *ensemble methods* family [34]. These methods use supervised learning methodology over a set of labelled data (training set) to make predictions and produce a model which can be later used to classify nonlabelled data [39]. It uses supervised learning methodology to collect data from parameter values and threshold values, working on a set of training data [40]. The method combines the idea of bagging with the random selection of characteristics, so as to build decision trees using controlled variance [37].

The RF model is successfully used in classification and regression tasks, operating via the construction of multiple decision trees during training, with the purpose of discovering patterns existing in data. The method generates several trees as subsets by combining several automatic learning algorithms appropriately selected [33]. This method is a general technique of random decision trees that combines the idea of bagging with a random selection of characteristics, with the intention of building decision trees with controlled variance [34, 35].

RF is an ensemble method for classification and regression tasks, which operates through the construction of multiple decision trees during training [34]. Additionally, RF is useful for calculating the influence of predictive variables on the target and also for calculating the importance of each of these influences over the target. The calculation of this importance is made with a metric calculated according to impurity decrease in each node used for partitioning data. In case of a classification, the class determined corresponds to the mode of the classes provided by each tree. In case of a regression, it corresponds to the average prediction of individual trees. Random decision trees correct the DT tendency to overadjust to their training set [41].

3.5. Case Study. Using operational and piling data, a case study was conducted with a database of about 30,000 records. For each parameter above, discrete values of low, normal, and high were devised according to threshold values previously defined by SCM Franke, which are commonly used in copper leaching. In particular, this discretization considered data standard deviation (σ) defining low (low value of the variable), corresponding to values lower than a $-\sigma$; normal (normal value of the variable), corresponding to values at the interval $[-\sigma, \sigma]$; and values considered high (high value of the variable), that is, those greater than $+\sigma$.

3.6. Methodology. The methodology consists of 4 steps. The initial step to collect data of both operation and piling are considered a stage previous to the methodology described below since these data (mainly operational data) were collected during several years of operation. Parameter values were grouped in periods including days of operation while class (recovery) is described for each day of operation per each period. Figure 1 shows examples of what was described above. Figure 1(a) shows daily recovery in two consecutive periods of operation, while Figure 1(b) shows daily recovery in two consecutive periods, but with pilot plant (piling) data. In detail, the steps of our methodology are as follows:

- (1) **Data Preparation.** This stage included filtration tasks and data selection per leaching cycles. Plant data were obtained with a frequency of four hours in one year. Due to process conditions and operational decisions, the irrigation of some piles or modules in service was stopped during some periods, a fact that could render incongruent results when modeling the system. To ensure operational data congruence, records corresponding to irrigation suppression periods were deleted from the database; these records were being

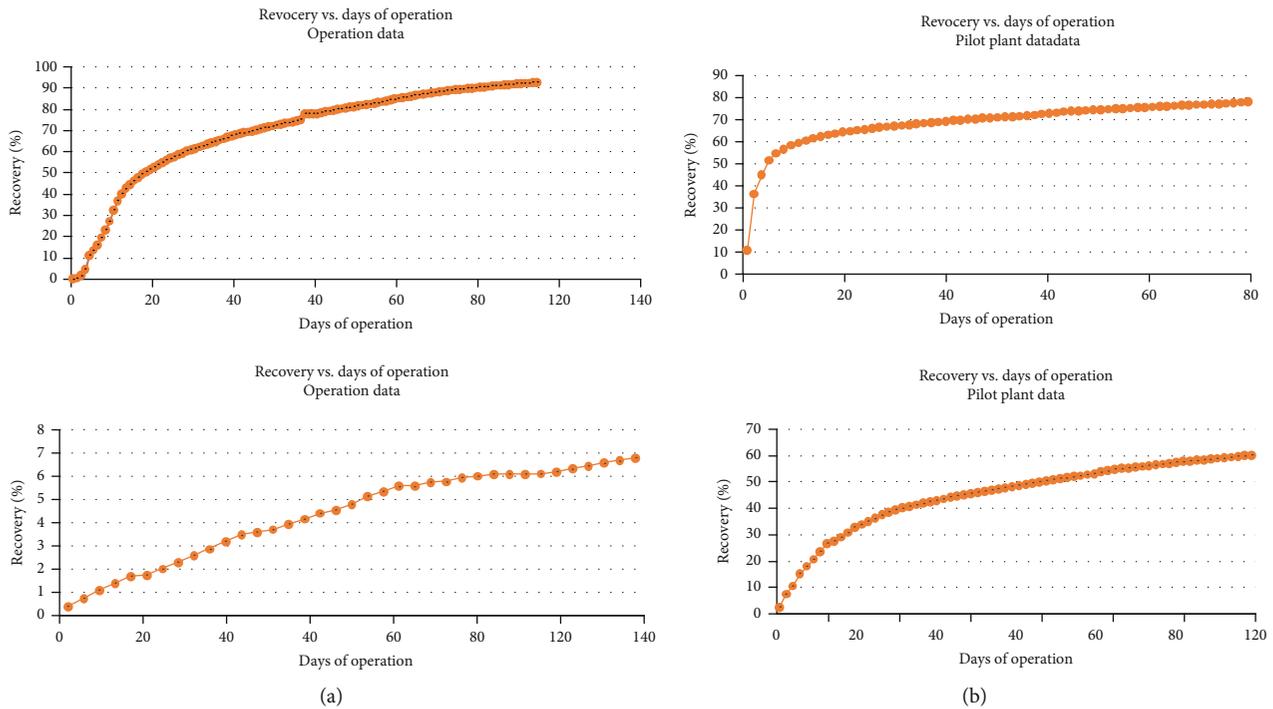


FIGURE 1: Examples of recovery periods. (a) shows daily recovery in two consecutive periods of operation, while (b) describes two consecutive periods with pilot plant (piling) data.

substituted by the leaching ratio. Leaching cycles with recovery values lower than 10% were also deleted after day 15 of the operation because this indicates an error in data acquisition

- (2) *Model Generation*. In this stage, data were collected and selected according to relevance in order to create a predictive copper recovery model on the conditions determined by the context of the study
- (3) *Model Visualization and Analysis*. In this stage, model results were visualized and analyzed to determine their validity. Evaluation consisted of checking the performance of the models obtained with RF for each dataset. To do this, values of certainty such as *accuracy*, *recall*, and *precision* were calculated and analyzed. The way these values of certainty were calculated and their importance for model quality are described below
- (4) *Result Analysis*. In this stage, the analysis is aimed at establishing if the results obtained are useful for the industry. This was done by analyzing aspects such as how optimal variable parametrization was or how well classified training set instances were (confusion matrix values)

To make the analysis in stage 3 above, a confusion matrix was considered. The confusion matrix facilitates the analysis necessary to determine an error in the classification, through a sample of error distribution in the different categories.

In this matrix, performance indicators [42] frequently used to evaluate classifier performance are described. They

are accuracy (*Acc*), recall (*r*), and precision (*p*). The way these indicators are calculated is described in Equations (1)–(3). The simplest indicator to evaluate a classifier performance is accuracy (*Acc*), corresponding to sample ratios correctly classified in the total number of examples of the dataset [33]. This indicator can be calculated on the basis of confusion matrix data according to Equation (1) (the dataset is supposed not to be empty). The other indicators, recall (*r*) and precision (*p*), are understood as relevance measures.

The *p* value is the ratio of true positives (*a*) among the elements predicted as positive (*a + b*). Conceptually, *p* value refers to the dispersion of the value set obtained from repeated measures of a quantity. Specifically, a high *p* value indicates low dispersion in measures. The *r* value is the ratio of true positives predicted among all the elements classified as negative.

$$\text{Acc} = \frac{a + b}{a + b + c + d}, \quad (1)$$

$$p = \frac{a}{a + b}, \quad (2)$$

$$r = \frac{a + d}{b + c}, \quad (3)$$

where *a* is the true positives, *b* is the false positives, *c* is the true negatives, and *d* is the false negatives.

4. Results

The problem described above was dealt with as a regression instance, looking for obtaining a copper recovery prediction numerically from data in each dataset (operational and piling). So, a model was obtained for both operational and piling data, the importance of associated variables being studied in both cases. To obtain the models, the free Rapid Miner Studio v 9.0 was used.

The strategy used in the model generation process was, first, preparing data according to task 1 of the methodology above. After the data preparation process (according to Section 3), a file with 1638 records for piling and another with 2001 records for operation were obtained (both files in CSV format). Previous studies such as [12, 34, 43] indicate that a minimum value of 1000 input cases for RF minimizes error in the classification and, at the same time, enables RF to make more stable predictions. So, both datasets are considered appropriate for generating the models.

In order to prepare the model evaluation and in a similar way to what is done in [34], a parameter tuning phase was performed. The models were evaluated using these parameters (40-fold crossvalidation 10 times) and averaging final results were taken. But the results of this validation were not good, for roundness. So, a method based on hold-out validation and similar to that performed in [34] was done as follows: for each dataset and using our defined optimal parameterization, one part of each dataset was taken to adjust the model and the rest of the sample for testing. In detail, to adjust the models, 70% of the total data in each dataset was used, leaving the remaining 30% for conducting the validation. The results and details of this are presented below.

4.1. Model Based on Random Forest Using Piling Data.

Table 2 summarizes the values obtained with RF in the parameter optimization process during training with the piling dataset. The parameters of interest for the optimal parametrization obtained in this model, that is, confidence (Con), number of trees (NoT), max depth (MDp), and accuracy (Acc), were used for interpreting results; these values are related to the confidence in a random tree model [43, 44].

Parameter Con is related to relative error, according to studies such as [1, 44]. Therefore, the values of $Con = \{25, 40, 55, 70, 85, 100\}$ were used for grouping the values of NoT, MDp, and Acc.

Figure 2 shows the values of Acc for each value of Con. Figure 2 also shows that all the graphs indicate a decreasing trend for parameter Acc, except for $Con = 40$. In this figure, the best mean value of Acc is for $Con = 100$, the following best values being for $Con = 40, 55, \text{ and } 85$. In all cases highlighted as the best, the average value of tree depth (MDp) is 8.5. This may be interpreted as follows: the best combination of parameters is given when the mean tree depth of 8.5 is achieved; that is, this value represents the optimal depth in this classification.

On the basis of the piling data, the confusion matrix of this model was also obtained. In this optimization, 80% data were used for crossvalidation and 20% for validation [40] (Table 3).

TABLE 2: Mean values of NoT, MDp, and Acc (as a percentage) for each value of Con; optimal parametrization for each operational set.

Con	NoT	MDp	Acc (%)
25	62.5	8.5	83.4
40	62.5	8.5	83.5
55	62.5	8.5	83.6
70	62.5	12.7	83.5
85	62.5	8.5	83.6
100	62.5	8.5	83.8

Table 4 shows the importance of variables for this model. The most important variable is “agglomerate H dose,” followed by variable “RL.” In contrast, the least important variable is “soluble Cu.” Variables “operation day,” “H fed,” and “CO₃ grade” are over 10% of the value of importance, a fact that may be interpreted as their having a good predictive capacity for this model. This is not so for variable “Soluble Cu,” which does not exceed the threshold value of 10%.

4.2. RF-Based Model Using Operational Data. This section describes the results obtained with the operational data. Table 5 summarizes the statistical values obtained with RF in the parameter optimization process during training with the operational dataset. Like the model using piling data, parameters NoT, MDp, and Acc of optimal parametrization were used for interpreting results, grouped according to parameter Con. Figure 3 shows that all the graphs indicate a decreasing trend for parameter Acc. Also, all the mean values of Acc are quite close to one another (Table 5).

As can be seen in Figure 3, the best is when $Con = 25$. Other important aspects are, on the one hand, that the mean depth of trees increased (mean value = 12.7) as compared with the previous model (mean depth = 9.2). This indicates that a greater number of depth cases per each tree were classified, which is good for the model. On the other hand, the number of trees decreased (mean value = 23.4) as compared with the number of trees of the piling data model (mean value of the number of trees = 62.5). This may indicate that, as a whole, data were easier to group for the model algorithm.

Thus, on the basis of the abovementioned data and as shown in Figure 3, it may be stated that optimal parametrization for the operational data model is better than its equivalent with piling data.

Similar to the previous piling model, the confusion matrix for this model was also obtained, optimization procedure being the same as the previous model. Table 6 shows that all the values of recall (r) exceed 93%, the lowest being for the label high, thus coinciding with the previous model. Given this coincidence, the conditions for classifying records in this label should be improved to make future classifications better. The performance of the model is reliable, given the value $p = 98.90\%$ and the value of accuracy.

Table 7 shows the importance of variables for this model. The two most important variables here are the same as those of the piling data model (agglomerate H dose = 22.76% relative importance and RL = 18.86% relative importance). As

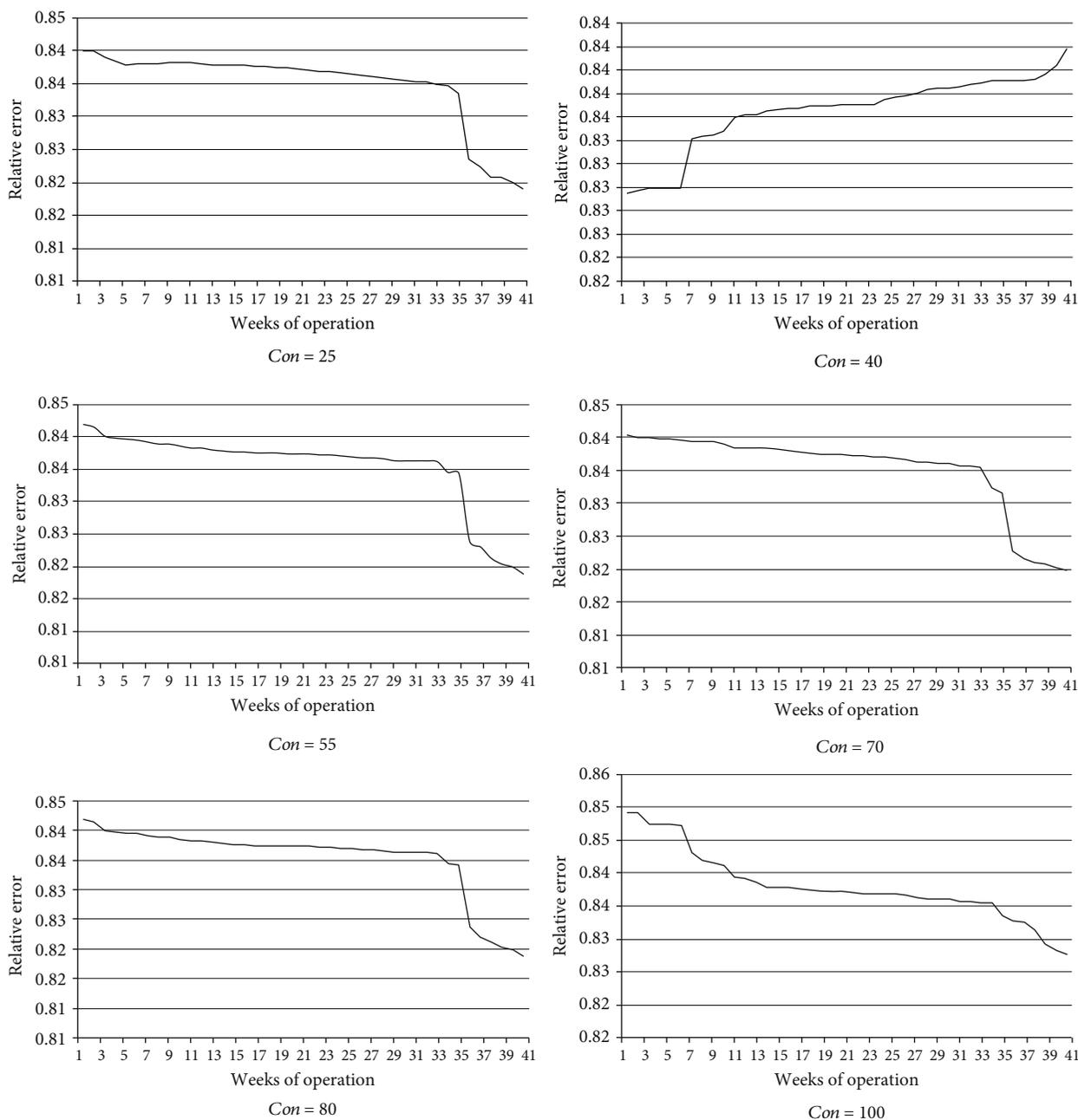


FIGURE 2: Evolution of parameter Acc for each of the values of parameter Con in optimization using piling data. The x values correspond to weeks of operations and y values correspond to accuracy values for each Con of relative error.

TABLE 3: Confusion matrix for operation with piling data. Value of accuracy = 98.72%.

	True low	True medium	True high	Class precision
Pred. low	1039	12	0	99.86%
Pred. medium	6	531	2	98.52%
Pred. high	0	1	47	97.92%
Class recall	99.43%	97.61%	95.92%	

can be seen in Table 7, the order or importance of variables is the same as shown in the previous model (Table 4), but the importance values are different. The least important variable in Table 7 is the same as in the previous model (Cu soluble). For this model, the percentage value of soluble Cu decreased in about 1%. This means that, although the order of importance of variables is maintained, the relative importance of the variables changes with respect to the previous model. Since this model was developed using operational data, it is prudent to consider that this order of importance is the most convenient. Figure 4 illustrates the contrast described above.

TABLE 4: Importance of variables for the piling data model.

Attribute	Importance	Relative importance (%)
Agglomerate H dose	13.74	19.31
RL	13.42	18.86
Total Cu grade	11.11	15.61
Day of operation	8.92	12.54
H fed	8.82	12.40
CO ₃ grade	8.39	11.79
Soluble Cu	6.75	9.49

TABLE 5: Mean values of NoT, MDp, and Acc (as a percentage) for each value of Con; optimal parametrization for each operational dataset.

Con	NoT	MDp	Acc (%)
25	23.0	13	91.9
40	23.6	12.7	91.6
55	23.6	12.7	91.6
70	23.6	12.7	91.6
85	22.9	12.7	91.6
100	24	12.7	91.6

Figure 5 summarizes the importance of variables according to RF models for each experiment. Particularly, the figure shows that variable H+fed (volumetric flow of ILS solution) is the most important, followed by variables RL, total Cu grade, and day of operation. The order of importance of the variables remains in both classifications; that is, reproductivity of the conditions of the leaching pile in a controlled environment (piling) is an accurate representation thoroughly describing the pile, and therefore, piling can be used to predict pile copper recovery, with a much lower cost and reliability in the predictive model resulting from piling.

5. Discussion

Artificial intelligence techniques, specifically soft computing, are being used in productive industry to generate predictive models that improve industrial activity [25]. Random forest (RF) was used in this study to predict copper recovery by leaching. Predictive models using RF have been recently published by the mining industry, showing good results such as those reported in [3, 12, 33], but these studies were directed to objectives different from copper recovery prediction.

In recent papers such as [9], artificial intelligence computing tools (particularly machine learning algorithms) have been reported, but no evidence of the use of RF has been found in the literature to predict copper recovery. However, these works have helped to identify and relate information that directly influences to improve the copper recovery process by leaching.

The study published in [3] highlights that machine learning algorithms, since they are artificial neural networks, regression trees, random forest, and support vector machines, make up powerful tools currently scarcely used

in the copper mining industry, though there should be a tendency to increasingly use these machine learning tools in the present mining industry.

In RF, each tree is developed on the basis of the bootstrap algorithm philosophy. This may mean that the classification obtained for each tree is precise, thus causing a positive impact on the models presented here. In addition, this philosophy of work has made it possible to use all datasets in the classification and generate the models. The model precision obtained in this study is similar in both cases. The model for both datasets shows that a wealth of information was used to interpret the influence of predictive variables on class. For example, the order of the variables of interest is similar in both models and the performance shown by variables Acc, p , and r enables concluding that both models have a good quality and could be used to predict copper recovery in new cases with a good reliability value.

The capacity to identify the importance of variables for the model using training data (piling) is similar to the one shown by the model using actual data (operation). This was an expected result since the leaching material was the same in both cases, but this result validates the applicability of the machine learning algorithm selected for generating the models.

On the basis of the above described information, the objective of environmental trade-off was accomplished because model performance is optimal, and in both cases, the greatest number of records was classified as normal, when the acid irrigation rate lies between 20 and 50 g/l (normal value).

6. Conclusions

Copper recovery prediction by hydrometallurgical methods and, particularly, leaching is usually made with the help of mathematical models, but soft computing techniques can help create complex computational models [45] that help in this prediction. Recently, an increase in using soft computing tools in the industry has been observed [9, 13, 39], but in this particular case, the literature does not contain many studies reporting the use of RF to generate a copper recovery prediction model.

This study resulted in the generation of two copper recovery prediction models using the leaching method. Actual data (operation) were used in one of the models, while the other model was generated with hive-simulated data which had the same characteristics as the material to be leached and the lixiviant. In both cases, the models achieved an excellent predictive quality, one of the cases reaching 100% prediction for the label high, the mean being higher than 95% precision. In this way, it excelled in what was posed in the objective of this study (described at the beginning of this document).

As recently published in [9], a comparison between a linear model and an artificial neural network (ANN) for predicting copper recovery is made. One of the conclusions of this study is that ANN exceeds the linear model in terms of precision, but as conclusion at the present work, the interpretation capacities of RF-generated models exceed those of

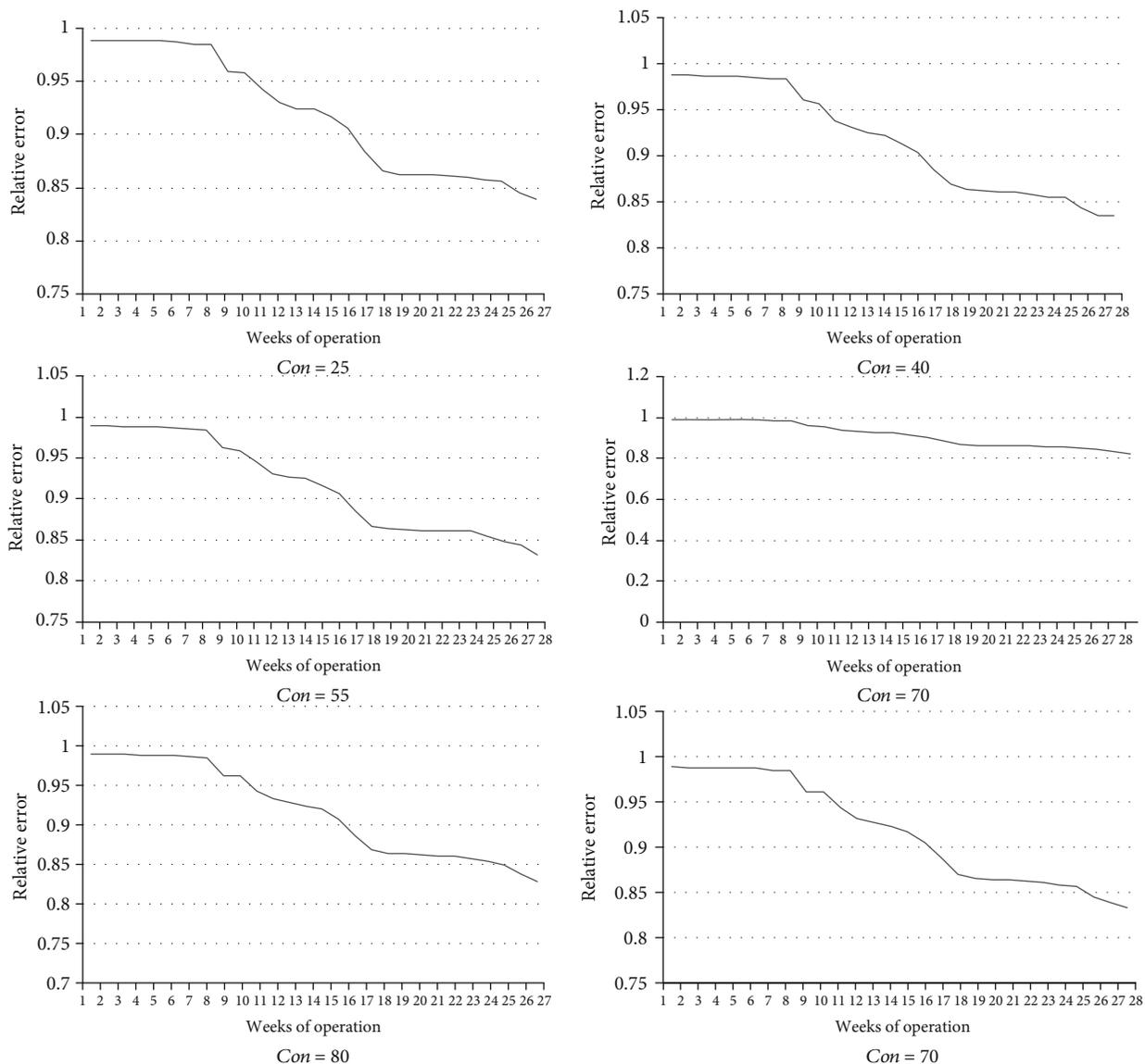


FIGURE 3: Evolution of parameter Acc for each value of parameter Con in optimization using operational data. The x values correspond to weeks of operations and y values correspond to accuracy values for each Con of relative error.

TABLE 6: Confusion matrix for the operation with operational data. Value of accuracy = 98.90%.

	True low	True medium	True high	Class precision
Pred. low	1487	7	0	99.53%
Pred. medium	10	434	5	96.58%
Pred. high	0	0	68	100.00%
Class recall	99.33%	98.38%	93.15%	

TABLE 7: Importance of variables for the operational data model.

Attribute	Importance	Relative importance (%)
Agglomerate H dose	23.02	22.76
RL	18.32	18.86
Total Cu grade	16.28	16.10
Day of operation	12.63	12.49
H fed	11.07	10.95
CO ₃ grade	11.01	10.89
Soluble Cu	8.79	8.69

ANN from the work previously mentioned, thus making it easier to arrive at conclusions.

This study helped make a comparison between two copper recovery prediction models in the same work context.

Adjustment precision measure indicates that the RF algorithm is highly useful for processes to predict future copper production.

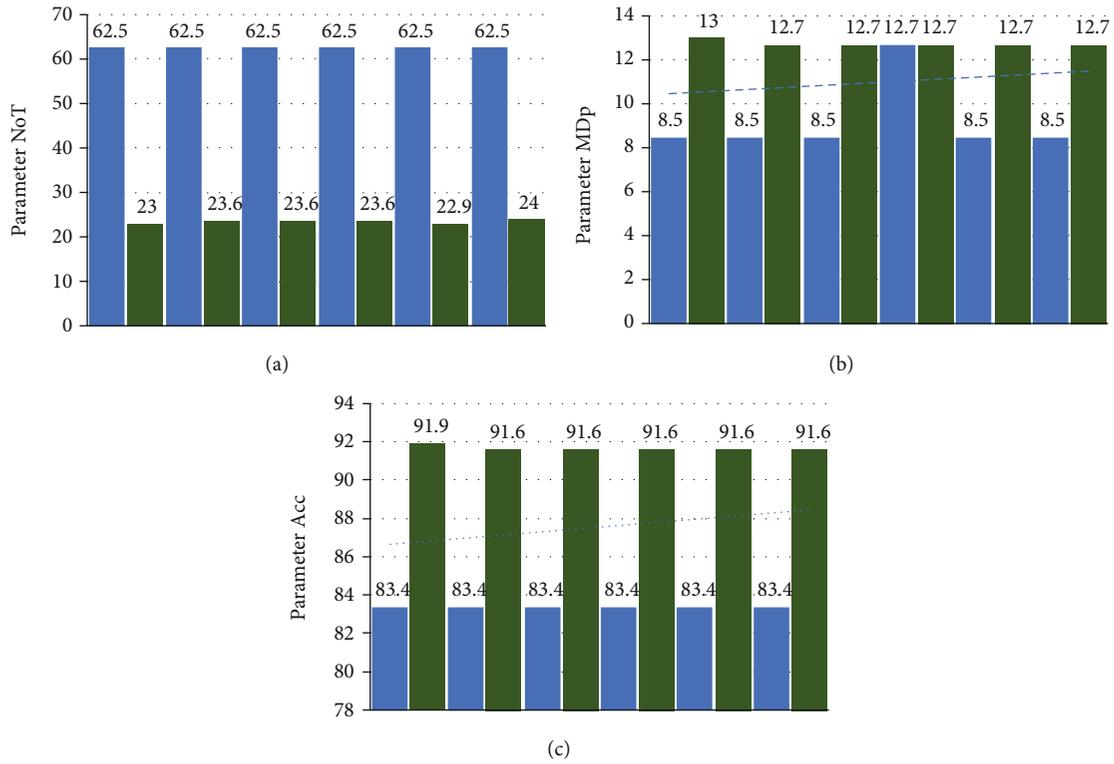


FIGURE 4: Comparison of parameter values for each value of parameter of interest for the optimal parametrization, obtained in each of the predictive models. (a) corresponds to parameter NoT, (b) to parameter MDP, and (c) to parameter Acc.

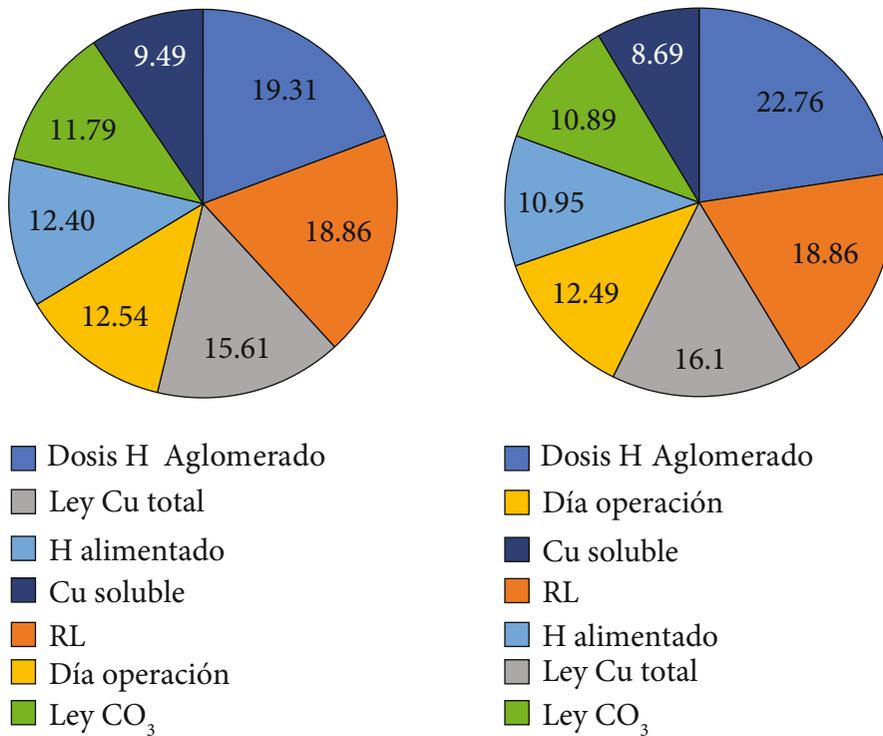


FIGURE 5: Importance of predictive variables for each RF-generated model.

In addition, experience was gained for defining and implementing the predictive model in the leaching domain on this specific work context. This experience may be used for other simulations of processes relative to the improvement of results to obtain copper at SCM Franke by means of soft computing techniques or other companies of the same industrial production sector.

What was said about model performance, the capacity to identify the influence of variables on class, and the capacity to interpret results, etc., is very important in the copper industry because it allows generating supporting tools for material exploitation planning, along with viewing, via indicators generated with this type of model, copper recovery results in the presence of a certain material. It also allows properly selecting both the most influential variables and the values of those variables to achieve the desired recovery. This may have a considerable impact on the intelligent exploitation of this mineral, considering the increasing demand and lack of this industrial activity.

To conduct this study, a methodology was proposed; results obtained by following the methodological steps devised show excellent quality and are replicable for other copper leaching piles to study the future performance of copper recovery using the prediction method. Also, this methodology can be transferred to other copper leaching processes, including the knowledge of this particular process to generate a predictive model. In this way, this study may indicate a future line of research.

Data Availability

The input data used to support the findings of this study could be available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflict of interest.

Acknowledgments

The authors thank the SCM Franke company for the material support provided and their collaboration to this study, particularly their support for obtaining and storing data from the copper production process using the leaching method, as well as data from the pilot plant (column simulation) and their experience to select the parameters and criteria used for generating predictive models, and are thankful for the financial support provided by Universidad Católica del Norte.

References

- [1] A. C. Pereira and F. Romero, "A review of the meanings and the implications of the industry 4.0 concept," *Procedia Manufacturing*, vol. 13, pp. 1206–1214, 2017.
- [2] C. Pietrobelli, A. Marin, and J. Olivari, "Innovation in mining value chains: new evidence from Latin America," *Resources Policy*, vol. 58, pp. 1–10, 2018.
- [3] J. Katz and C. Pietrobelli, "Natural resource based growth, global value chains and domestic capabilities in the mining industry," *Resources Policy*, vol. 58, pp. 11–20, 2018.
- [4] L. Stubrin, "Innovation, learning and competence building in the mining industry. The case of knowledge intensive mining suppliers (KIMS) in Chile," *Resources Policy*, vol. 54, pp. 167–175, 2017.
- [5] J. De Gregorio and F. Labbé, "Copper, the real exchange rate and macroeconomic fluctuations in Chile," *Beyond the Curse: Policies to Harness the Power of Natural Resources*, pp. 203–233, 2011.
- [6] Y. Ghorbani and S. H. Kuan, "A review of sustainable development in the Chilean mining sector: past, present and future," *International Journal of Mining, Reclamation and Environment*, vol. 31, no. 2, pp. 137–165, 2017.
- [7] E. L. Esquenazi, B. K. Norambuena, Í. M. Bacigalupo, and M. G. Estay, "Evaluation of soil intervention values in mine tailings in northern Chile," *PeerJ*, vol. 6, p. e5879, 2018.
- [8] E. J. Lam, B. F. Keith, Í. L. Montofré, and M. E. Gálvez, "Copper uptake by *Adesmia atacamensis* in a mine tailing in an arid environment," *Air, Soil and Water Research*, vol. 11, 2018.
- [9] C. Leiva, V. Flores, F. Salgado, D. Poblete, and C. Acuña, "Applying soft computing for copper recovery in leaching process," *Scientific Programming*, vol. 2017, 6 pages, 2017.
- [10] C. A. Leiva, K. V. Arcos, D. P. Poblete, E. A. Serey, C. M. Torres, and Y. Ghorbani, "Design and evaluation of an expert system in a crushing plant," *Minerals*, vol. 8, no. 10, p. 469, 2018.
- [11] P. Meller and A. M. Simpasa, *Role of Copper in the Chilean & Zambian Economies: Main Economic and Policy Issues*, Global Development Network, GDN Working Paper Series, 2011.
- [12] L. A. Zadeh, "Fuzzy logic, neural networks, and soft computing," *Communications of the ACM*, vol. 37, no. 3, pp. 77–84, 1994.
- [13] V. Rodriguez-Galiano, M. Sanchez-Castillo, M. Chica-Olmo, and M. Chica-Rivas, "Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines," *Ore Geology Reviews*, vol. 71, pp. 804–818, 2015.
- [14] S. Nawar and A. Mouazen, "Comparison between random forests, artificial neural networks and gradient boosted machines methods of on-line Vis-NIR spectroscopy measurements of soil total nitrogen and total carbon," *Sensors*, vol. 17, no. 10, p. 2428, 2017.
- [15] F. Zandiyyeh, M. R. Shayestefar, H. Ranjbar, and S. Saadat, "Prospectivity mapping of iron oxide-copper-gold (IOCG) deposits using support vector machine method in Feysaabad area (east of Iran)," *Journal of Himalayan Earth Sciences*, vol. 49, no. 2, pp. 50–62, 2016.
- [16] B. S. Saljoughi and A. Hezarkhani, "A comparative analysis of artificial neural network (ANN), wavelet neural network (WNN), and support vector machine (SVM) data-driven models to mineral potential mapping for copper mineralizations in the Shahr-e-Babak region, Kerman, Iran," *Applied Geomatics*, vol. 10, no. 3, pp. 229–256, 2018.
- [17] T. Sun, F. Chen, L. Zhong, W. Liu, and Y. Wang, "GIS-based mineral prospectivity mapping using machine learning methods: a case study from Tongling ore district, eastern China," *Ore Geology Reviews*, vol. 109, pp. 26–49, 2019.

- [18] V. Rodriguez-Galiano, B. Ghimire, M. Chica-Olmo, and J. Rigol-Sanchez, "An assessment of the effectiveness of a random forest classifier for land-cover classification," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 67, pp. 93–104, 2012.
- [19] A. Porwal, E. Carranza, and M. Hale, "Knowledge-driven and data-driven fuzzy models for predictive mineral potential mapping," *Natural Resources Research*, vol. 12, no. 1, pp. 1–25, 2003.
- [20] M. Abedi, S. A. Torabi, G. H. Norouzi, M. Hamzeh, and G. R. Elyasi, "PROMETHEE II: a knowledge-driven method for copper exploration," *Computers & Geosciences*, vol. 46, pp. 255–263, 2012.
- [21] H. K. Haghighi, M. Rafie, and D. Moradkhani, "Modeling on transition of heavy metals from Ni–Cd zinc plant residue using artificial neural network," *Transactions of the Indian Institute of Metals*, vol. 68, no. 5, pp. 741–756, 2015.
- [22] R. Xu, *Improvements to random forest methodology*, [Ph.D. thesis], Iowa State University, Ames, IA, USA, 2013.
- [23] T. Kohonen, "An introduction to neural computing," *Neural Networks*, vol. 1, no. 1, pp. 3–16, 1988.
- [24] S. Visweswaran and G. Cooper, "Learning instance-specific predictive models," *Journal of Machine Learning Research*, vol. 11, pp. 3333–3369, 2010.
- [25] V. Flores and M. Correa, "Performance of predicting surface quality model using softcomputing, a comparative study of results," in *International Work-Conference on the Interplay Between Natural and Artificial Computation*, pp. 233–242, Springer, 2017.
- [26] V. Skorpil and J. Stastny, *Neural networks and backpropagation algorithm*, Electronics, Bulgaria, Sozopol, 2006.
- [27] D. M. Skapura and J. A. Freeman, *Neural Networks, Algorithms, Applications and Programming Techniques*, Reading Addison Wesley, 1991.
- [28] M. Cilimkovic, *Neural networks and backpropagation algorithm*, Institute of Technology Blanchardstown, Blanchardstown Road North Dublin, 2015.
- [29] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [30] E. J. M. Carranza and A. G. Laborte, "Random forest predictive modeling of mineral prospectivity with small number of prospects and data with missing values in Abra (Philippines)," *Computers & Geosciences*, vol. 74, pp. 60–70, 2015.
- [31] T. K. Ho, "Random decision forests," in *Proceedings of 3rd International Conference on Document Analysis and Recognition*, pp. 278–282, Montreal, Quebec, Canada, Aug. 1995.
- [32] C. Zhang, C. Liu, X. Zhang, and G. Almpandis, "An up-to-date comparison of state-of-the-art classification algorithms," *Expert Systems with Applications*, vol. 82, pp. 128–150, 2017.
- [33] V. Flores and B. Keith, "Gradient boosted trees predictive models for surface roughness in high-speed milling in the steel and aluminum metalworking industry," *Complexity*, vol. 2019, Article ID 1536716, 15 pages, 2019.
- [34] L. Gutiérrez, V. Flores, B. Keith, and A. Quelopana, "Using the Belbin method and models for predicting the academic performance of engineering students," *Computer Applications in Engineering Education*, vol. 27, no. 2, pp. 500–509, 2019.
- [35] I. Barandiaran, "The random subspace method for constructing decision forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 832–844, 1998.
- [36] T. M. Oshiro, P. S. Perez, and J. A. Baranauskas, "How many trees in a random forest," in *International Workshop on Machine Learning and Data Mining in Pattern Recognition*, pp. 154–168, Springer, Berlin, Heidelberg, 2012.
- [37] R. Lior, "Data mining with decision trees: theory and applications," *World scientific*, vol. 81, pp. 11–50, 2014.
- [38] O. Salmani Nuri, E. Allahkarami, M. Irannajad, and A. Abdollahzadeh, "Estimation of selectivity index and separation efficiency of copper flotation process using ANN model," *Geosystem Engineering*, vol. 20, no. 1, pp. 41–50, 2017.
- [39] H. A. Nooruddin, F. Anifowose, and A. Abdulraheem, "Using soft computing techniques to predict corrected air permeability using Thomeer parameters, air porosity and grain density," *Computers & Geosciences*, vol. 64, pp. 72–80, 2014.
- [40] C. K. Chow and C. Liu, "Approximating discrete probability distributions with dependence trees," *IEEE Transactions on Information Theory*, vol. 14, no. 3, pp. 462–467, 1968.
- [41] P. M. Granitto, F. Gasperi, F. Biasioli, E. Trainotti, and C. Furlanello, "Modern data mining tools in descriptive sensory analysis: a case study with a random forest approach," *Food Quality and Preference*, vol. 18, no. 4, pp. 681–689, 2007.
- [42] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.
- [43] S. Bhattacharyya, "Confidence in predictions from random tree ensembles," *Knowledge and Information Systems*, vol. 35, no. 2, pp. 391–410, 2013.
- [44] S. Wager, T. Hastie, and B. Efron, "Confidence intervals for random forests: the jackknife and the infinitesimal jackknife," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1625–1651, 2014.
- [45] M. Milivojevic, S. Stopic, B. Friedrich, B. Stojanovic, and D. Drndarevic, "Computer modeling of high-pressure leaching process of nickel laterite by design of experiments and neural networks," *International Journal of Minerals, Metallurgy, and Materials*, vol. 19, no. 7, pp. 584–594, 2012.

Research Article

Location Fixing and Fingerprint Matching Fingerprint Map Construction for Indoor Localization

Marwan Alakhras ¹, Mousa Hussein ¹ and Mourad Oussalah ²

¹EE, United Arab Emirates University, UAE

²University of Oulu, Finland

Correspondence should be addressed to Marwan Alakhras; marwan2024@yahoo.com

Received 10 May 2019; Accepted 4 December 2019; Published 17 January 2020

Guest Editor: Sidike Paheding

Copyright © 2020 Marwan Alakhras et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Building the fingerprint map for indoor localization problems is a labour-intensive and time-consuming process. However, due to its direct influence on the location estimation accuracy, finding a proper mechanism to construct the fingerprint map is essential to enhance the position estimation accuracy. Therefore, in this work, we present a fingerprint map construction technique based on location fix determination and fingerprint matching motivated by the availability of advanced sensing capabilities in smartphones to reduce the time and labour cost required for the site survey. The proposed Location Fixing and Finger Matching (LFFM) method use a landmark graph-based localization approach to automatically estimate the location fixes for the Reference Points and matching the collected fingerprints, without requiring active user participation. Experimental results show that the proposed LFFM is faster than the manual fingerprint map construction method and remarkably improves the positioning accuracy.

1. Introduction

With the increase of ubiquitous deployment of Wi-Fi infrastructures both at indoor and at outdoor environments together with the exponential multiplication of mobile computing devices, intense attention has been paid to various approaches of Wi-Fi network aided location tracking [1]. Such positioning methods are extremely useful in applications such as visitor navigation in museums or large buildings where GPS signal is not available: elderly health care positioning systems, facility management, transportation, and emergency rescue [2, 3]. In this respect, fingerprinting techniques became particularly popular for indoor localization.

More formally, utilizing the existing Wi-Fi infrastructures [4], the fingerprinting approach involves two distinct phases: offline map construction and online location estimation. In the offline phase, a site survey is conducted to collect Received Signal Strength (RSS) at known locations called Reference Points (RPs), which usually contain the Media Access Control (MAC) address of visible Access Points (APs). The RPs' coordinates along with their corresponding

RSSs are then stored in a radio map database. The latter is used in the second phase, referred to testing or online phase, which, through some interpolation or machine learning-based technique, enables the system to infer the position of an unknown (mobile) target given its RSS value. Nevertheless, radio map construction often requires a large number of locations—labelled RSS fingerprints—gathered from either wireless APs or RPs, often evenly distributed within the area of interest, in order to ensure a good positioning accuracy during the testing phase. Besides, in order to account for the inherent deficit of wireless signal propagation, several samples, executed with special care, are needed at each measurement.

Therefore, and from our personal experiences as mentioned in [5], a major challenge in constructing efficient radio map is the expensive time requirement and labour-intensive site survey process, hindering a wide set of potential applications, which, in turn, promote RSS fingerprinting localization-based techniques. Besides, the complex and dynamic nature of the indoor wireless environment, which makes radio map maintenance difficult as the signal, is easily

influenced by the structures, layout, and pedestrians around the study site. In order to deal with the radio map construction challenge, many research works have been performed to expedite the site survey process while ensuring high-quality standards. For instance, the Simultaneous Localization and Mapping (SLAM) [6–8] can be used to construct the radio map with lower survey cost, but it may be unsuitable to work on the resource-limited handheld devices such as smartphones due to its heavy computation load [9]. Although many map management methods (e.g., Map Filtering [10] and pedestrian dead-reckoning [11]) can be employed to infer a user's initial location and improve SLAM performance accordingly, their associated high computational cost restricts their implementation in resource-limited ubiquitous devices. Alternatively, the crowdsourcing-based radio map construction has been proposed recently in both active and passive formats [10]. While the active crowdsourcing method requires active user participation, which reduces the need for professional surveyors, it may suffer from intentional frauds due to user participation. On the other hand, the passive crowdsourcing approach reduces user participation by utilizing information from smartphone inertial sensors in order to associate fingerprints to the corresponding RPs. Although the passive crowdsourcing is more practical, its main drawback lies in the low accuracy and the need for GPS readings, which, in turn, may constrain the application scope [12]. Therefore, using the high-end smart devices that are equipped with a variety of sensors (e.g., a barometer that detects floor change in the building) can be vital to improving user experience and crowdsourcing-based radio map construction, which, ultimately, enhances the indoor localization system and expands the applicability of the underlying localization services. However, the accumulated error of many inertial sensors grows over time for instance gyroscope; also, to the potential uncertainty pervading the locations of APs, it may force special care and a regular update that utilizes the environment knowledge.

In this respect, a new passive fingerprint crowdsourcing method has been proposed in this paper. More formally, this work proposes a landmark graph-based localization method for automatically estimating the location of RPs matched with the collected fingerprints in order to design a practical, fast, and reliable fingerprint data collection method using sensor-rich smart devices. Compared to other existing methods, our technique can significantly build a fast fingerprint map with a minimum user participation while ensuring high standards in terms of accuracy and reliability due to the incorporating of a *belief factor* that accounts for the RPs and assesses the quality of location fixes of RPs, so that only those location fixes that are associated with high quality are matched with the corresponding fingerprints. This achieves better location accuracy than Map Filtering and Pedestrian Dead Reckoning (PDR) [13]. Besides, the accuracy of the constructed radio map is compared to state-of-the-art approaches ZEE [10] and manual fingerprint map building techniques [14].

The rest of this paper is organized as follows: Section 2 describes the related work in the field. Section 3 details the

proposed system, while Section 4 describes the LFFM technique based on a landmark graph, Section 5 describes the fingerprint map construction, Section 6 explains the experimental findings, and Section 7 draws the conclusion.

2. Related Work

Indoor localization has been an active area of research for the past three decades, initially in the context of mobile robot navigation and more recently in the context of pervasive and mobile computing where the issue of environment mapping is crucial for a variety of applications requiring various levels of accuracy. In this course, one distinguishes approaches that make use of existing infrastructures (e.g., Wi-Fi APs as in RADAR [15]) and those that require special deployment infrastructure (e.g., infrared beacons as in Active Badge [16] and ultrasound devices as in Cricket [17]). In the area of mobile robotics, a significant step was achieved by the emergence of Simultaneous Localization and Mapping (SLAM) which allowed a robot to build a map of the indoor environment, usually, in terms of walls and other obstruction objects, while simultaneously determining its location with respect to the constructed map.

In the SLAM group, special attention has been drawn to the offline map construction techniques that often employ motion and system models, where several approaches have been proposed, typically, such methods differ according to the employed sensors, type of prior knowledge available, and the used mapping techniques [6–8]. For instance, one may distinguish the recently populated GraphSLAM [18], which transforms the SLAM posterior into a graphical network, where a greedy algorithm is usually used for data association. Traditionally, the positions of landmarks (objects) in the environment are estimated using a laser range finder like a sensor (either fixed or scanning through the environment) embedded in some mobile platform, which yields a relatively accurate estimate of the RPs with respect to the local frame. A critical assumption is that these landmarks do not change position during the journey. In the case of nonstatic of either the mobile platform or landmarks, due to the accumulation of the relative positioning error throughout the journey, a correction step is required, yielding an updated map of the environment. For example, the WiFi-SLAM system proposed in [8] uses a Gaussian Process Latent Variable Model to associate Wi-Fi fingerprints in combination with a motion dynamics model without requiring any location labels in the training data. Authors in [19] used GraphSLAM in order to improve the computational efficiency of the WiFi-SLAM system and relieve its dependency on the requirement of fingerprint uniqueness assumption. WiSLAM [7] uses a Bayesian framework to fuse WiFi RSS with data from a foot mounted inertial measurement units for localization and mapping. In [6], a Smart SLAM is proposed to construct a radio map through a fruitful combination of different fusion algorithms, namely, PDR algorithm, Fingerprint Extended Kalman Filter (FEKF), Fingerprint Extended Kalman Filter SLAM (FEKF-SLAM), and Distributed Particle SLAM (DPSLAM). However, the heavy computation load of these SLAM systems prevents them from being implemented on

resource-limited handheld devices such as smartphones. Moreover, another disadvantage of such filtering-like techniques is that the data, once processed, is often discarded. This makes it impossible to revisit all data at the time of a map building task [18].

In contrast to those above computationally expensive map building techniques and costly labour-based site survey methods, crowdsourcing has emerged as a natural participatory model that benefits from daily user activities and widespread of smartphone devices. In this respect, one distinguishes active crowdsourcing [20–22] and passive crowdsourcing [10, 23, 24]. The active crowdsourcing methods construct a radio map by utilizing the user feedback where the surveyors are volunteers who decided to share RSS signal at specific locations of the environment. Especially, authors in [9–11] highlighted several clear advantages of such techniques. First, the underlined radio map can provide robust and accurate fingerprint data even when it is built solely on short-duration RSS measurements. Second, there is no constraint on the type and the number of devices employed. As a result, the active crowdsourcing eliminates the need for costly professional surveyors. Nevertheless, this should not hide some implicit limitations as well. For instance, since users carry heterogeneous devices, this can result in a radio map built where RSS values originated from diverse devices with distinct chipsets and antenna designs, which affect the calibration or consistency of employed radio model, even when the devices were placed at exactly the same positions. It can also result in holes in terms of data sparsity with no designated fingerprint collection points. Indeed, since the radio map is updated by untrained voluntary users without centralized controls, different users can upload their fingerprint data that is collected at slightly different locations but with the same location label. Multiple fingerprint data indicating one particular location not only cause slow location estimation but also store space wastage in a radio map server. Finally, since it requires active user participation, this technique may suffer from intentional frauds [20]. A particular example of an active crowdsourcing system is the Organic Indoor Location (OIL) [20] that periodically asks the users to bind their measurements along with their locations on the floor plan, providing information about the nearby wireless resources, then the determined position is reflected into a global map. Similarly, FreeLoc [21] and Mobile Organic Localization Engine (Mol'e) [22] use semantic labels rather than exact floor maps to annotate fingerprints with locations such as rooms, hallways, and corridors.

On the other hand, passive crowdsourcing methods associate fingerprints with the corresponding RPs with the aid of smartphone inertial sensors. For example, EZ [23] uses occasional GPS fixes together with reported RSS measurements of specific APs arising from users' participation to build a radio map that does not require any predeployment effort. Nevertheless, the reliance on the method on the existence of occasional GPS fixes might be an issue in an indoor environment, which delays the map construction until the availability of GPS signal. Zee [10] utilizes smartphone inertial sensors to track the user while performing Wi-Fi scan simultaneously, which enabled the authors to construct a radio map in a non-

intrusive way. However, the approach uses magnetometers to calculate the direction, which is easily affected by building structure, service amenities, and furniture in indoor environments. Besides, its use of the particle filter to fuse inertial sensors with a floor plan is computationally expensive, which makes it unsuitable for running on resource-limited devices. LiFS [24] builds the radio map by exploiting the built-in sensors of the mobile device with the floor plan of the testing environment, which resulted in a relatively fast deployment process and a less labour effort. Especially, it was shown that LiFS works well in buildings exhibiting some spatial symmetry of the environment layout; e.g., office rooms are connected by a corridor and are equally distributed on both sides of the corridor, but may fail to work on other types of buildings. Although the passive crowdsourcing significantly reduces the labour cost of site surveying task and does not require active user participation, it has several problems too. This includes low accuracy, limited applicability to hand-device implementation constraints, and more importantly the requirement for GPS readings. Table 1 summarizes some of the most common fingerprint map construction techniques.

Our work falls in the class of methods that rely on the existing infrastructure, namely, Wi-Fi network, without recourse to any additional deployment or detailed floorplan and exact AP placements, which distinguishes it from works in [20, 22]. Similarly, our approach also belongs to the class of passive crowdsourcing methods in the sense of enabling random smartphone users to participate in the radio map construction task as in [21], but with the additional step of utilizing inertial device sensory information in order to estimate the RPs positions. In addition, our approach introduces a confidence factor (called belief factor) that accounts for the quality of location fixes of RPs, so that only those location fixes that are associated with high quality are matched with the corresponding fingerprints. Our approach also bears similarity with Gu et al.'s work [13] that makes use of mobile inertial sensors for position estimation but with the additional use of crowdsourcing technology and confidence factor analysis in the pattern-matching task. Likewise, our work bears similarity with EZ's approach [23] with the difference that our proposal does not require any GPS fix locations, making use of device inertial sensors. Besides, we also restrict to cases where some prior information about the environment through floor plan is provided, which excludes fully unknown environment scenarios as in some other studies.

3. System Overview

The outline of the proposed fingerprint collecting technique based on Location Fixing and Fingerprint Matching (LFFM) is shown in Figure 1. It is composed of two main modules: Location Fixing of RPs and Fingerprint Matching.

The Location Fixing module receives measurements from the built-in smartphone sensors and is fed with a landmark graph [13]. The sensor measurements—basically, *barometer*, *accelerometer*, *gyroscope*, and *magnetometer*—are used to determine the step size and directional heading of the

TABLE 1: Fingerprint map construction techniques.

Methods	System name	Algorithm and requirements	Accuracy of RM	Testbed area
Crowdsourcing	RedPIN [25]	Label position by user, indoor map	Room level (90%)	26 rooms
	Molé [22]	Kernel, accelerometer	Room level (91%)	3-floor building
	FreeLoc [21]	Relative RSS comparison	<3 m	A laboratory, a corridor
SLAM	WiFi-SLAM [8]	GP-LVM, initial ISO map model	3.97 m (ME)	250–500 m (traces)
	SignalSLAM [26]	Least square, PDR, GraphSLAM, landmarks, accelerometer, gyroscope, magnetometer	<16.5 m (MD)	200 m × 160 m
	Graph-SLAM [18, 19]	Sparse graph, constraint optimization, least square, linearization, approximation, EKF SLAM, accelerometer, GPS	<10 m	Urban area 600 m × 800 m
Inertial sensors	Zee [10]	DR, augmented particle filter, indoor map, accelerometer, gyroscope, magnetometer	1.2 m (50%), 1.8 m (80%)	65 m × 35 m
	LiFS [24]	DR, feature extraction, indoor map, accelerometer	5.88 m (ME)	70 m × 23 m
	WILL [27]	PDR, <i>K</i> -means, accelerometer	Room level (86%)	70 m × 23 m
Semisupervised learning	Manifold learning [28]	Manifold alignment, inherent spatial correlation of RSS, path loss model, partial RPs, APs' locations, indoor map	3.8 m (ME); 2.4 m (ME)	40 m × 30 m, 5 APs; 40 m × 20 m, 4 APs
	Coforest [29]	Implicit crowdsourcing sampling, random forest ensemble classifier, partial RPs, RSS	3.65 m (ME)	800 m ² , 30 APs
Unsupervised learning	WRM [30, 31]	HMM, EM, memetic algorithm, path loss model, indoor map, APs' locations	Around 3 m (ME)	80 m × 32 m, 30 APs
Path loss model	ARIADNE [32]	Ray tracing, path loss model, simulated annealing algorithm, APs' location, partial RPs, indoor map	3 m (ME), 2.5 m (STD)	65 m × 48 m, 5 APs
	Multiwall Path Loss Model (MWM) [33]	MWM, APs' location, indoor map, parameters setting for Gaussian distribution, Euclidean distance error, kNN	1.2 m (ME)	480 m ² , 3 APs
Interpolation	Inverse Distance Weighting (IDW) [34]	RSS, interpolation and extrapolation methods, estimation error statistics, uniform grid, IDW, probabilistic positioning	5~20 m (ME)	150 m × 60 m, 316 2.4 GHz APs, 106 5 GHz APs
	Kriging [35]	Kriging algorithm, spatial interpolation, semivariogram model fitting, unbiased estimation, RSS, <i>K</i> -weighted nearest neighbours, cubic spline, boundary condition	1.12 m (ME)	9.5 m × 2.5 m, 9 APs
	Forward Interpolation [36]	(fixed, zero-slope natural, nonnode), RMS, RSS	2.82 m (best)	5 rooms, 4 APs

particular user using the well-known Pedestrian Dead Reckoning (PDR) method described in [37]. The estimated step size and directional heading are then employed to infer the current user position in real time with respect to the landmarks according to the provided landmark graph. In this respect, a landmark processing task is necessary in order to reduce the error accumulation of the PDR, in addition to position calibration based on landmark detection.

The Fingerprint Matching module uses the fixed locations estimated by the Location Fixing module and the scanned Wi-Fi signal information (MAC address and corresponding RSS) to generate the fingerprint radio map. Location estimates together with the corresponding match fingerprint are then added into the radio map—only if they meet a certain quality requirement, which will be detailed

later on. This constraint is added to guarantee the accuracy and reliability of the generated fingerprint map. More details on the algorithm will be elaborated in the next section. However, initially, Table 2 will help the reader to navigate through the article by providing a list of abbreviations.

4. Location Fixing and Fingerprint Matching (LFFM) Based on Landmark Graph

4.1. Landmark Detection. Our approach utilizes the fact that naturally spread smartphones in indoor environments are rich with powerful sensors and sensing ability. In our work, a landmark will refer to spatial points where sensor measurements indicate an identifiable and distinguishable pattern or change in pattern. For example; corners and turns will force

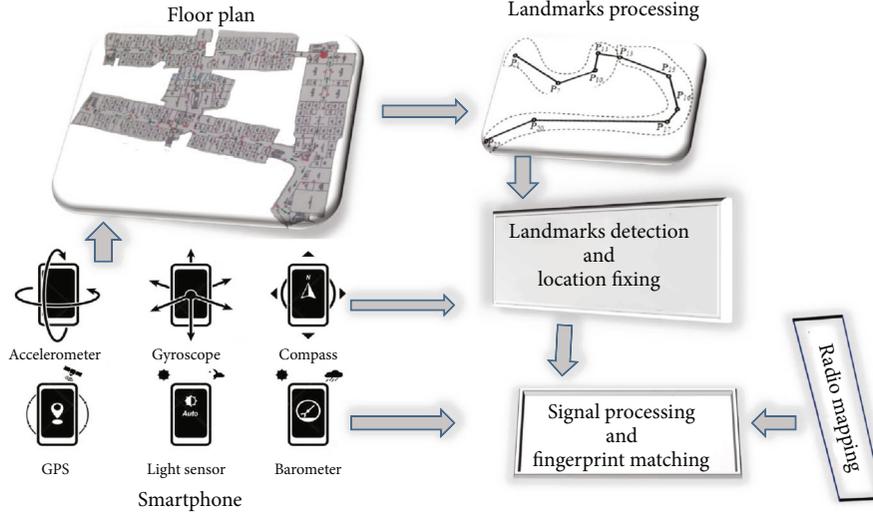


FIGURE 1: System architecture.

users to change walking direction, while, stairs and elevators are highlighted by elevation change. Similarly, doors are triggered whenever a change in the velocity is identified, and thus, a movement pattern can be predicted.

For this purpose, we utilize information issued from inertial sensors, mainly, barometer, gyroscope, and accelerometer for landmark detection. The detection of landmarks is based on the distinct change of pattern in one or more types of sensor readings. The locations of these landmarks correspond to the locations of elevators, stairs, corners, turns, and doors that can be simply obtained from floor plans, which is usually available for most indoor environments.

More formally, a landmark L can be defined as

$$L : \langle (x, y, z), (R_1, \dots, R_N) \rangle, \quad (1)$$

where (x, y, z) denotes the location of the landmark according to the underlined floor plan and (R_1, \dots, R_N) represents the set of detection rules, according to the different types of sensor measurements, satisfied by the corresponding landmark, where N corresponds to the number of rules triggered by this landmark. Typically, each device inertial sensor would trigger one or more rules associated to the constraints on the measurement values/patterns linked to each landmark type.

4.1.1. Barometer-Based Landmark. The barometer sensor measurements provide information about vertical movement detection, for example, a user going upstairs, going downstairs, or taking an elevator, since the barometric value changes with the altitude or height. Although the barometric pressure is influenced by many factors such as temperature and altitude, due to the relatively instantaneous measurement time, such factors are discarded in this work, and we only account for altitude information. The change in the barometer readings, when a user walks horizontally, upstairs/downstairs, or takes the elevator, can be shown in Figure 2.

Here, the entrance and exit of stairs and elevators are considered barometer landmarks due to their pattern repre-

sentation in barometer pressure measurements, since it is distinctive, identifiable, and stable. The entrance detection includes a “horizontal movement→vertical movement” pattern. Similarly, the exit detection includes a “vertical movement→horizontal movement” pattern.

More formally, having observed the change in barometer readings as shown in Figure 2, where it is almost a linear change when the user moves vertically, in this case, the linear model can be used to fit those readings; then, let p_i denote the pressure of the i^{th} window of air pressure readings at time t , and let β_{bar_1} and β_{bar_2} be the thresholds used to detect horizontal movement and vertical movement, respectively. Then R_{bar} rule to detect the entrance to a set of staircases or an elevator can be defined by the following rule:

$$R_{\text{bar}_1} : \left(\text{loc}_t \text{ when } (|p_i - p_{i-1}|) < \beta_{\text{bar}_1} \text{ AND } |S_1| = K_{p_1} \text{ AND } \left| p_i + K_{p_1} - p_i \right| > \beta_{\text{bar}_2} \right). \quad (2)$$

The first term is for detecting horizontal movement, and the latter two terms are for detecting vertical movement. The thresholds β_{bar_1} and β_{bar_2} on horizontal and vertical movements are set manually after an extensive testing phase. Although many algorithms can be used to determine the thresholds experimentally [38], for the time being, it is out of the scope of this work, thus all the thresholds are set manually based on the empirical test as shown in Table 2, for example, and as illustrated in Figure 2, β_{bar_1} can be calculated as $(1013.4 - 1013.2)/4 \text{ sec} = 0.05 \text{ hPa}$ for every second in case of going up or down on the stairs.

The function S_1 can be defined as

$$S_1 = \sum_{j=i+1}^{i+K_{p_1}} \left(\text{sgn} \left(p_j - p_{j-1} \right) \right), \quad (3)$$

TABLE 2: List of abbreviations.

R_{bar}	Barometer landmark rule
R_{acc}	Accelerometer landmark rule
R_{gyr}	Gyroscope landmark rule
β_{bar_1}	Barometer horizontal movement threshold
β_{bar_2}	Barometer vertical movement threshold
p_i	Pressure of the i^{th} window of air pressure readings at time t
m_t	Represents the motion state (e.g., walking and stationary) at time t
β_1	Accelerometer threshold for walking state
β_2	Accelerometer threshold for steady state
$\dot{\theta}_t$	Gyroscope measurement along the vertical direction
β_{gyr}	Gyroscope threshold
S_t	Step length
θ_t	Heading direction
P_t	Pressure values at time t
$P_{\text{diff}(a,b)}$	Pressure difference between floor a and floor b
$\text{con}(l_k)$	Confidence level of land mark
$\delta(R_k, R_t^*)$	Validity of rule detection
$r(\theta_k, \theta_t^*)$	Validity of path detection
$g(d_k, d_t^*)$	Ratio of moved distance
β_θ	Heading direction threshold
β_{con}	Confidence threshold
P_S	Path segment connecting two neighbour landmarks
$\text{bel}(P_S)$	Quality belief of path segment P_S
T	Step periodicity
$\beta_{\text{bel}(P_S)}$	Threshold of belief value $\text{bel}(P_S)$ of the path segment P_S
T_r	Trajectory

where sgn is the signum function, which can be described as

$$\text{sgn}(p_j - p_{j-1}) = \begin{cases} 1, & \text{if } p_j > p_{j-1}, \\ 0, & \text{if } p_j = p_{j-1}, \\ -1, & \text{if } p_j < p_{j-1}. \end{cases} \quad (4)$$

Similarly, the rule to detect the exit from a set of staircases or an elevator can be defined by the following rule:

$$R_{\text{bar}_2} : \left(\text{loc}_t \text{ when } (|p_i - p_{i+1}|) < \beta_{\text{bar}_1} \text{ AND } |S_2| = K_{p_2} \text{ AND } |p_i + K_{p_2} - p_i| > \beta_{\text{bar}_2} \right). \quad (5)$$

Then the function S_2 can be defined in the same spirit of (2):

$$S_2 = \sum_{j=i-K_{p_2}+1}^{i+K_{p_1}} \left(\text{sgn}(p_j - p_{j-1}) \right). \quad (6)$$

The two values K_{p_1} and K_{p_2} are not constant, but are determined dynamically. Their initial values can be set to 1 and gradually increase as long as the value of the signum function was kept unchanged.

4.1.2. Accelerometer-Based Landmark. The motion that presents a distinct change pattern can also be measured by the accelerometer. It can be considered, where a point that witnesses the changing pattern of “walking→steady→walking” (of course for an acceptable short period, i.e., one second or few seconds) can be regarded as a potential accelerometer landmark (the term *still* or *steady* refers to motionless action). This pattern may happen when passing doors as shown in Figure 3 or passing water fountain, which can be detected by comparing the magnitude of the accelerometer measurement against a predefined threshold (which is again set based on empirical results). A location point is regarded as an accelerometer landmark if the accelerometer readings present this changing pattern every time the user passes it. Formally, the rule R_{acc} of accelerometer landmarks is defined by the following rule:

$$R_{\text{acc}} : \left(\text{loc}_t \text{ when } m_{t-\beta_1, t} \text{ is walking AND } m_{(t+\beta_2)} \text{ is steady AND } m_{t+\beta_1} \text{ is walking} \right), \quad (7)$$

where m_t represents the motion state (e.g., walking, steady) at time t . β_1 and β_2 are two thresholds that are set empirically to determine the period of the corresponding motion state.

4.1.3. Gyroscope-Based Landmark. A location point where the gyroscope measurements present a distinct and stable pattern is considered a gyroscope landmark. The magnetometer can also be used to detect a change in direction, but its measurements tend to be affected by ferromagnetic materials. Therefore, gyroscope measurements seem to be more suitable for detecting the change in direction in our experiment. The target is to determine the right and left turn patterns as shown in Figure 4. This pattern can usually be witnessed at the location of a turn, corner, or door. The rule R_{gyr} to detect a gyroscope landmark can be set as

$$R_{\text{gyr}} : \left(\text{loc}_t \text{ when } |\dot{\theta}_t| > \beta_{\text{gyr}} \right), \quad (8)$$

where $\dot{\theta}_t$ is the gyroscope measurement along the vertical direction. When the absolute value of $\dot{\theta}_t$ is greater than a certain threshold β_{gyr} , this location point can be considered a potential gyroscope landmark. Trivially, the initial direction heading can be set to north or south depending on the underlying environment orientation.

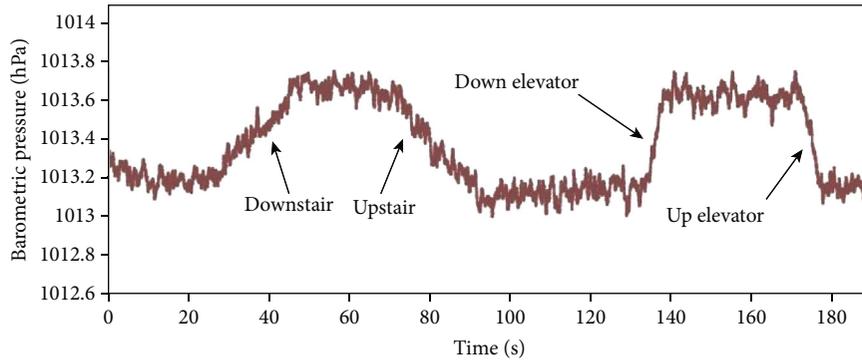


FIGURE 2: Barometer readings on changing elevations.

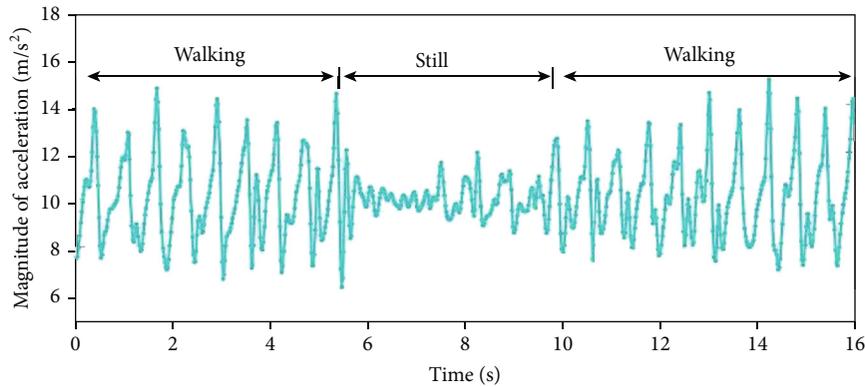


FIGURE 3: Accelerometer reading example when passing door landmark.

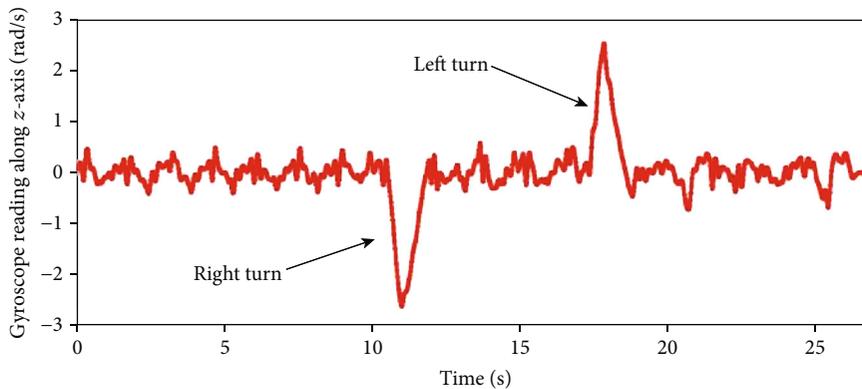


FIGURE 4: Gyroscope readings on right and left turns.

4.2. *Landmark Graph for Location Fixing of RPs.* A landmark graph can be defined as a directed graph where nodes are landmarks and edges are accessible paths with heading information.

Let $G = (L, E)$ denote a landmark graph where $L = \{l_1, l_2, \dots, l_N\}$ is a set of landmarks and $E = \{e_1, e_2, \dots, e_M\}$ is the set of edges in graph G . Each edge e_i consists of the two landmarks, direction from one landmark to another, and the corresponding distance between them, $e_i = \langle l_j, l_k, \theta_{jk}, d_i \rangle$. Note that the direction from landmark l_j to landmark l_k is different from that from l_k to landmark l_j . In other words, there are two edges between any two neighbour land-

marks. The general algorithm for Location Fixing and Fingerprint Matching (LFFM) using a landmark graph in indoor localization is shown in Figure 5.

The details of each task in Figure 5 is commented as follows:

- (1) The construction of the landmark graph requires location landmarks, which can be extracted from a floor plan. Most buildings are symmetric. For instance, floors of the same building have a similar or almost identical layout, so that by changing floor information on its landmark graph, we can easily obtain the landmark graph for another floor

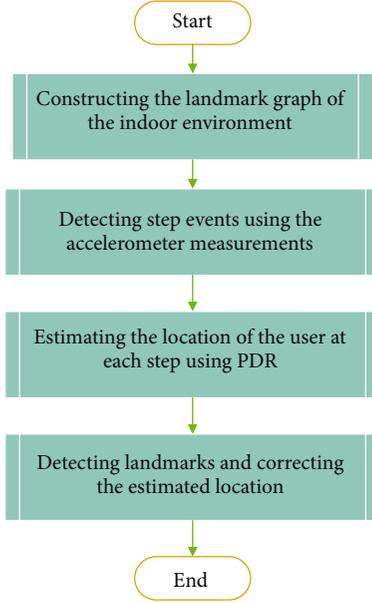


FIGURE 5: Landmark detection algorithm.

- (2) The step detection is done via detecting acceleration peaks, where each acceleration peak corresponds to one step. In order to distinguish the walking state from the steady state, we check if the change of the acceleration within the detected time window is greater than the set threshold. If a step is detected, then the corresponding heading direction and step size are computed
- (3) Using the inertial sensor measurements and the PDR technique, the user's position at each step can be calculated given his/her initial location as follows:

$$x_t = x_{t-1} + S_t \sin(\theta_t), \quad (9)$$

$$y_t = y_{t-1} + S_t \cos(\theta_t), \quad (10)$$

$$f_t = f_{t-1} - \frac{P_t - P_{t-1}}{P_{\text{diff}(a,b)}}, \quad (11)$$

where x_t and y_t are the user coordinates at floor f_t at time t and S_t is the corresponding step length and θ_t the heading direction. P_t and P_{t-1} are the air pressure values at time t and $t-1$, respectively. And $P_{\text{diff}(a,b)}$ is the air pressure difference between floor a and floor b . Adding the threshold values for both step detection and direction heading will ensure values and of course a robust position when these landmarks are added later to the localization map

- (4) While the PDR is conducted to estimate the location of the user, the measurements from the barometer, gyroscope, and accelerometer are simultaneously used to detect landmarks. Although both the gyroscope readings and compass readings (inferred from accelerometer readings and magnetometer readings)

can be used to estimate the heading, they cannot provide a robust heading estimation since the gyroscope has the drift problem and the compass is vulnerable to ferromagnetic materials [39]. Therefore, the landmark graph is used to assist with the heading direction estimation. If a user is detected to walk on the path connecting two landmarks in the landmark graph, the heading from this landmark graph will be used. Otherwise, the compass readings will be used as the heading. Also, the step size is updated when a user passes two neighbouring landmarks in the landmark graph. Let l_1 and l_2 indicate the two neighbour landmarks that a user passes subsequently. Given the (x, y) coordinates of each of these landmarks, and the number N_s of detected steps between them; then the step size S can be calculated as follows:

$$S = \frac{\sqrt{(X_{l_1} - X_{l_2})^2 + (Y_{l_1} - Y_{l_2})^2}}{N_s} \quad (12)$$

The step size estimation method does not require the user's stature information and can adapt to varying walking speeds, and it is updated as the user passes two neighbour landmarks on the landmark graph.

Using landmarks for assisting localization may result in a need to solve the data association issue [40], in other words, when there are multiple landmarks nearby, or when there is a case that one or more landmarks are missed from sensor data, or mainly it can be used to avoid false detections.

We need to have a measure of how much confidence a location points meeting the landmark detection rule. The confidence that a location point is matching should be constrained by the following facts; first, the detected landmark should be a valid detection for one of the landmark types; second, it has to fall within the same moving direction; in other words, there should be somehow a path between the estimated landmark at a specific time and the landmark at an earlier time; third, it has to have an acceptable distance from the previous detection. Formally, let l_k be a landmark then the confidence $\text{con}(l_k)$ in the landmark graph can be calculated as

$$\text{con}(l_k) = \delta(R_k, R_t^*) \cdot r(\theta_k, \theta_t^*) \cdot g(d_k, d_t^*), \quad (13)$$

where k is the index of a landmark in the landmark graph, R_k is the detection rule of the reference landmark l_k , and R_t^* is the type of the detected landmark at time t ; then $\delta(R_k, R_t^*)$ is the validity function of the detected landmark, with δ as the Dirac delta function that is denoted as

$$\delta(R_k, R_t^*) = \begin{cases} 1, & \text{if } R_k = R_t^*, \\ 0, & \text{otherwise.} \end{cases} \quad (14)$$

From (13), where θ_k and θ_t^* are the reference heading and the estimated heading from the time visiting the last

landmark to time t . Then, the path validity r is the rectangle function adopted from our earlier work [14] and can be described as

$$r(\theta_k, \theta_t^*) = \begin{cases} 1, & \text{if } |\theta_k - \theta_t^*| < \beta_\theta, \\ 0, & \text{otherwise,} \end{cases} \quad (15)$$

where β_θ is a heading direction threshold, when d_k and d_t^* are the reference distance and the traveled distance from the last landmark to the location point. Then the distance weight function g is defined as

$$g(d_k, d_t^*) = \frac{1}{|d_k - d_t^*|}. \quad (16)$$

Once every landmark is being associated with a confidence level, then, when multiple landmarks are nearby, the one with the highest confidence factor will be selected. Moreover, fake landmarks may occur due to user unpredictable actions, for instance making a turn in the middle of the corridor, which may be resulting in detection of a gyroscope landmark. To solve this issue, a confidence threshold β_{con} value may be used as such as

$$l_k : \text{con}(l_k) \geq \beta_{\text{con}}, \quad \forall l_k : (x, y)_t. \quad (17)$$

In some other cases, when landmarks are missed from sensor data, for instance, certain landmarks at the locations of doors can be missed if a door is left open since “walking→steady→walking” behaviour pattern will not be detected. In such a case, we have no option but to ignore the underlined landmark, which may result in increasing the accumulation error. But this error will be reduced back when the next landmark is detected since after the addition of every landmark the graph will be denser, having more nodes and, definitely, more connective paths, which will result in reducing the chance of error for the upcoming detections.

5. Construction of Fingerprint Map

5.1. Quality of Location Fixes of RPs. The construction of a fingerprint map is one essential step towards fine position estimation. Therefore, any location fix is subject to quality check before being matched to any fingerprint and added to the fingerprint map. As explained in the earlier section, the PDR method consists of two components: *step size estimation* and *heading direction estimation*. So, a robust algorithm will tend to bind the accumulative error of these two components. In [41], it has been proved that the step periodicity for the same motion state (e.g., walking and jogging) suffers from small variation when a user moves at a relatively constant speed. However, this will significantly vary when a user remains stationary (steady) while using the smartphone arbitrarily, for texting, playing phone games, etc. Therefore, limiting the step periodicity to a certain interval will reduce the

location estimation error. The step periodicity can be defined as the period of one step, which is equal to the time difference between two neighbour peaks of the accelerometer measurements as shown in Figure 3. Let P_S denote a path segment connecting two neighbour landmarks with N steps in between and $N + 1$ RPs.

$$P_S = \{(t_i, x_i, y_i, f_i), \quad \forall i = 1, 2, \dots, N + 1\}, \quad (18)$$

where x_i and y_i are the coordinates of RP_i at time t_i with corresponding floor information f_i . And let $T = \{T_1, \dots, T_N\}$ indicate the step periodicity set measured by the accelerometer for these N steps. Then the quality belief of path segment P_S going through those RP can be evaluated as

$$\text{bel}(P_S) = \frac{\sum_{i=1}^N \mathcal{X}(T_i) \cdot T_i}{\sum_{i=1}^N T_i} \cdot \frac{1}{\sigma_{T^*}}, \quad T^* \subseteq T, \quad \forall T_{\min} \leq T \leq T_{\max}, \quad (19)$$

where the first term indicates the ratio of valid steps with respect to the total number of steps after outliering false walks or frequent stops. The outliering step is carried out using the identifier function $\mathcal{X}(T_i)$, which is defined as

$$\mathcal{X}(T_i) = \begin{cases} 1, & \text{if } T_i \in [T_{\min}, T_{\max}], \\ 0, & \text{otherwise.} \end{cases} \quad (20)$$

The second term is reciprocal of the standard deviation of the valid time set T^* , which includes the valid walks only after outliering the fake walks. In this case, if the user walks with constant speed, then the deviation will be too small. Finally, the belief value $\text{bel}(P_S)$ of the path segment P_S is compared to a certain threshold $\beta_{\text{bel}(P_S)}$, and the location fix is considered accurate and reliable to be matched with and then added to the fingerprint map.

5.2. Fingerprint Matching and Construction of Fingerprint Map. It is necessary to evaluate the estimated location fix quality before using fingerprint matching. Because the accuracy of the RPs' location fix estimation has a direct impact on the accuracy of the construction of the fingerprint map.

Now, let T_r denote the trajectory that the user has traveled, including K path segments, which are divided by landmarks, namely, $T_r = \{T_1, \dots, T_K\}$. And the set of fingerprints collected along this trajectory $FP = \{fp_1, \dots, fp_N\}$, where N is the number of Wi-Fi scans, each fingerprint contains the RSS and the MAC address of its corresponding RP, since the time when a walk state event happens may be different from the time when the WiFi scan is conducted. Synchronizing the time to conduct the Wi-Fi scan with the time the walk event happens is necessary in order to match the fingerprint with the estimated location fixes. Suppose that the Wi-Fi scan at time t_j happens during the time period the user walks from the location (x_{k-1}, y_{k-1}) to (x_k, y_k) , namely, $t_{k-1} \leq t_j \leq t_k$; then

```

LFFM algorithm
Input:
  initila location  $(X_0, Y_0, f_0)$ 
  Trajectory  $T_r = \{T_1, \dots, T_k\}$ 
  vector of
   $FP = \{(RSS_1, MAC_{AP_1})_1, (RSS_2, MAC_{AP_2})_2, \dots, (RSS_p, MAC_{AP_p})_N\}$ 
Output:
  Fingerprint Map
for  $j = 1 : N$  do
  Compute the location  $(x_j, y_j, f_j)$  of the  $j$ -th potential Location fix;
  Search for the  $P_{S_i}$  that includes location  $(x_j, y_j, f_j)$ 
  Compute the belief  $bel(P_{S_i})$  of  $P_{S_i}$ 
  if  $bel(P_{S_i}) > \beta_{bel}$  then
    Match the fingerprint  $PF_j$  and the corresponding location  $(x_j, y_j, f_j)$ 
    Add the tuple  $(x_j, y_j, f_j, FP_j)$  to the fingerprint map;
  end if
end for

```

ALGORITHM 1: Location Fixing and Fingerprint Matching.

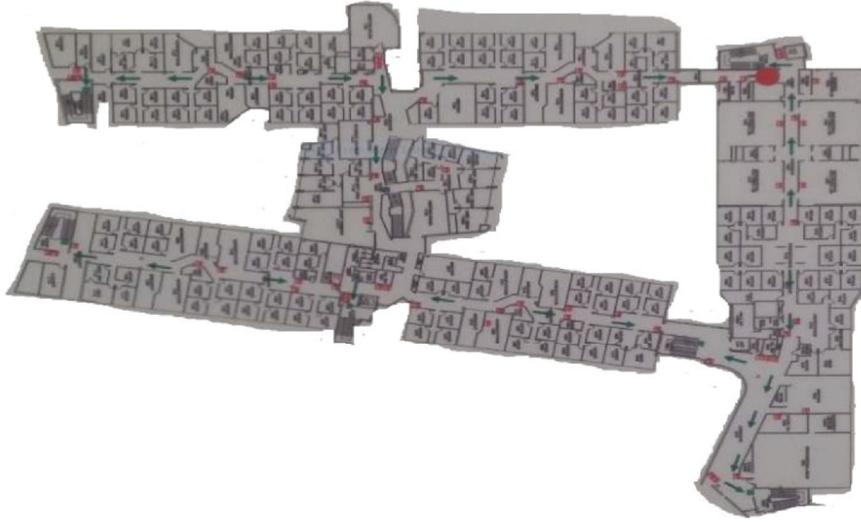


FIGURE 6: Experiment test bed layout.

we can estimate the location of the j^{th} potential location fix RPs by using a linear interpolation:

$$x_j = x_{(k-1)} + \frac{(x_k - x_{k-1}) \cdot (t_j - t_{k-1})}{t_k - t_{k-1}}, \quad (21)$$

$$y_j = y_{(k-1)} + \frac{(y_k - y_{k-1}) \cdot (t_j - t_{k-1})}{t_k - t_{k-1}}. \quad (22)$$

According to the calculated location (x_j, y_j) , then the path segment P_{S_i} can be found, such that it includes the location (x_j, y_j, f_j) . After this, we evaluate the quality of P_{S_i} , where only when the $bel(P_{S_i})$ meets the threshold requirement—it is essential to evaluate the quality since this will affect the map construction, and therefore, will affect the localization process—later on, the fingerprint is matched

with the corresponding location fix and then is added to the fingerprint map in term of RSS_j , MAC_{AP_j} , and its corresponding location (x_j, y_j, f_j) . This process is recursively done until all the elements in the FP set are used. A summary of the process is shown in Algorithm 1.

6. Experiments and Results

The proposed method was evaluated by experiments conducted in a three-story office building. The area of each floor is about 6,750 square meters. The testing path goes through two floors of this building, and its length is about 420 meters as shown in Figure 6. The building is made of precast concrete, blockwork, steel structure, aluminium, ceramic tiles, and zinc-coated materials and also offices and classroom furniture, lab equipment (electrical, electronic and mechanical), and it has many other home

TABLE 3: Parameters of experimental results.

Function	Parameter	Value
Barometer landmark	Pressure threshold β_{bar1}	0.05 hPa
	Pressure threshold β_{bar2}	0.3 hPa
Accelerometer landmark	Window size	50 samples
	Walking-state threshold	2 s
	Steady-state threshold	1-5 s
Gyroscope landmark	Window size	10 samples
	Gyro threshold β_{gyr}	1.1 rad/s
PDR	Pressure difference PR_{diff}	0.45 hPa
	Heading threshold β_{θ}	30°
	Confidence threshold β_{con}	0.25
	Initial step size	0.63 m
	Step periodicity threshold T_{max}	1 s
Quality evaluation	Step periodicity threshold T_{min}	0.4 s
	Belief threshold β_{s}	15

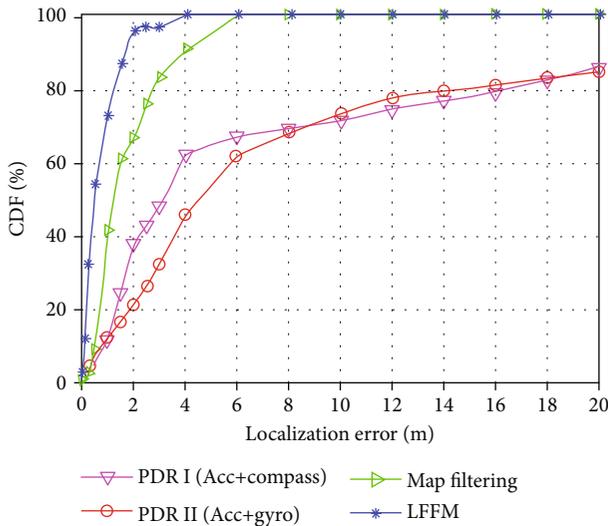


FIGURE 7: Localization accuracy comparison.

appliances. Using this environment and the test bed in Figure 6, a set of experiments were carried out to evaluate the proposed method.

The device used in the experiments is a Samsung Galaxy Note 3 smartphone equipped with Wi-Fi, accelerometer, magnetometer, gyroscope, and barometer. An android app was developed to collect the sensor data. A test user walked along the preset path with the phone in hand and clicked on the app to record the measurements of sensors and collect information about location fixes of RPs to evaluate the location accuracy. The data recorded include the MAC address of visible APs and corresponding RSS and readings from the accelerometer, gyroscope, and barometer. The values of parameters used in this work were empirically determined as shown in Table 3.

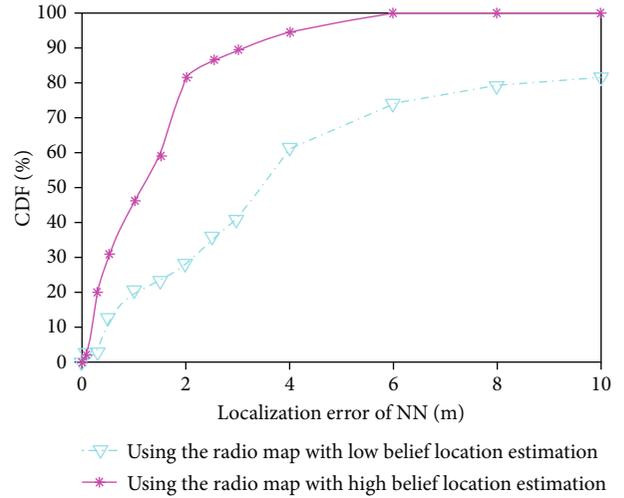


FIGURE 8: Localization error for kNN with different belief thresholds.

The results of LFFM based on the landmark graph for indoor localization method are compared with those of commonly used PDR-I and PDR-II methods in [13] and Map Filtering method in [42]. Both PDR methods used the step-counting approach to estimate the step size; however, to estimate the heading direction, PDR-I used the compass measurements, while PDR-II used the gyroscope measurements, which is almost the same as the sensors used in our work. The cumulative distribution of localization errors shown in Figure 7 shows that our method significantly outperforms the other methods, achieving a mean error of 0.71 meters.

Another experiment was carried out on a long straight path so that the heading direction could be ignored. This was necessary to evaluate the accuracy of location fixes, where in the first part the user walked along the straight distance with constant speed and in the second part the user walked the same distance with varying speeds and stopped at a few locations to imitate a fake walk status. The localization average error was less than 1 meter in the first part and about 5 meters in the second part.

Two different fingerprint maps were created with different belief thresholds introduced; later those maps were fused into the well-known algorithm k -nearest neighbour (kNN)—in the same spirit as in [5]—the localization error of kNN is shown in Figure 8. It can be seen that using the location estimation with high belief ($\text{bel}(P_s) > 18$) to match fingerprints for constructing a radio map achieves much higher accuracy than the one with low belief ($\text{bel}(P_s) < 10$). Therefore, it is necessary to control the quality of location estimation that is used to match fingerprints with location fixes, which has a direct effect on the accuracy of the fingerprinting localization method.

The best way to evaluate the constructed maps can be done by comparing the localization results using a map constructed using LFFM and another map. In this sense, another comparison was done for fingerprint map construction using the Zee system proposed in [10], and the manual fingerprint map construction technique, despite the huge time cost for

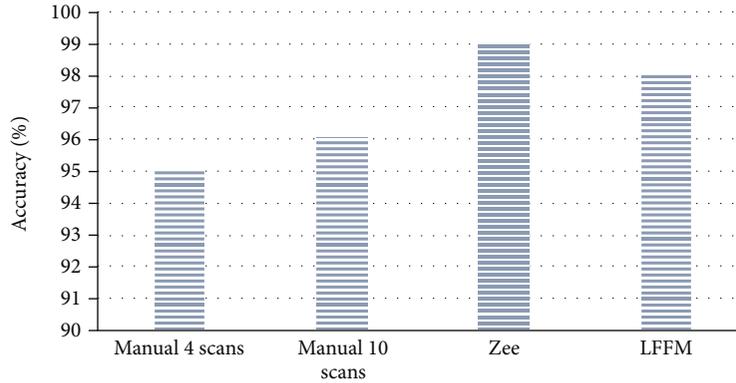


FIGURE 9: System recognition accuracy at -80 dBm.

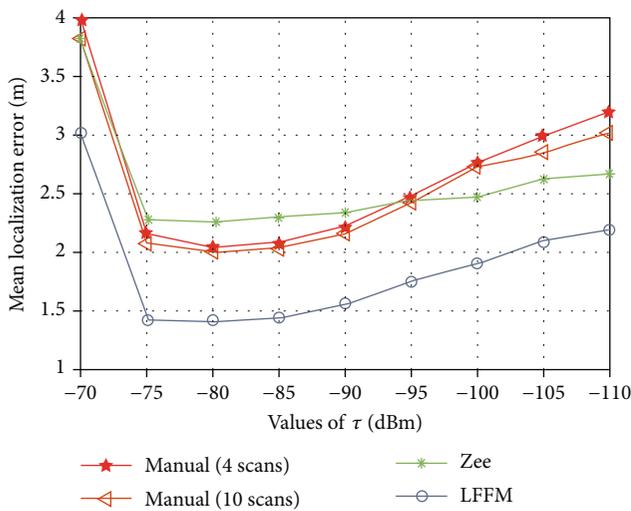


FIGURE 10: Effect of RSS threshold on the mean localization error.

constructing the fingerprint map manually, where RSS data has to be collected from a grid of points on the floor map. In this work two various Wi-Fi scans were made, the first with 4 Wi-Fi scans at each grid point, and the second with 10 Wi-Fi scans. These various scans seemed to be necessary to compensate for the orientation of the user, which is a very well-known problem in the fingerprint-based techniques. A total of 300 grid points were placed on the floor with a 2-meter distance interval. The localization errors were compared after fusing the resulting fingerprint maps into the kNN. The positive value fingerprint representation was used [43], namely, $positive_i(fp) = (RSS_i - \tau)$ if the i^{th} AP is present in the fingerprint fp and $RSS_i > \tau$, where RSS_i is the Received Signal Strength from the i^{th} AP and τ is a threshold value (APs whose RSS were lower than the threshold are considered not detected); otherwise, $positive_i(fp) = 0$. The accuracy of applied method is shown in Figure 9.

The effect of RSS threshold usually has a direct effect on the localization accuracy. Figure 10 demonstrates that the best performance of all the methods is achieved when the RSS threshold τ was set within $[-75, -85]$ dBm; in our experiments, we have set $\tau = -80$ dBm. A larger or smaller value of τ will lead to an increase in the mean localization error. This

is because increasing the RSS threshold may introduce Access Points with very weak signals, which are vulnerable to human movements, and decreasing the RSS threshold would exclude some useful Access Points that can help improve the localization accuracy.

7. Conclusion

This paper presented a fast and reliable Location Fixing and Fingerprint Matching to build a fingerprint map based on a landmark graph. Compared to the existing methods, namely ZEE and manual fingerprint map constructions performed in our earlier works, the new method outperforms the manual method and is almost equally as fast as the ZEE method. Moreover, it does not require active user participation; it takes less time and effort and can construct an accurate fingerprint map with the addition of the belief factor.

Data Availability

The data used to support the findings of this study are available from the corresponding authors upon request.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

References

- [1] Y. Gu, A. Lo, and I. Niemegeers, "A survey of indoor positioning systems for wireless personal networks," *IEEE Communication Surveys and Tutorials*, vol. 11, no. 1, pp. 13–32, 2009.
- [2] T. Cura, "A parallel local search approach to solving the uncapacitated warehouse location problem," *Computers & Industrial Engineering*, vol. 59, no. 4, pp. 1000–1009, 2010.
- [3] Y. S. Lee, J. W. Park, and L. Barolli, "A localization algorithm based on AOA for ad-hoc sensor networks," *Mobile Information Systems*, vol. 8, no. 1, 72 pages, 2012.
- [4] S. He and S.-H. G. Chan, "Wi-Fi fingerprint-based indoor positioning: recent advances and comparisons," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 466–490, 2016.

- [5] M. Alakhras, M. Hussein, and M. Oussalah, "Multivariable fuzzy inference with multi nearest neighbour for indoor WLAN localization based on RSS fingerprint," in *2013 UKSim 15th International Conference on Computer Modelling and Simulation*, pp. 656–662, Cambridge, UK, 2013.
- [6] R. M. Faragher and R. K. Harle, "SmartSLAM—an efficient smartphone indoor positioning system exploiting machine learning and opportunistic sensing," in *Proceedings of the 26th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2013)*, pp. 1006–1019, Nashville Convention Center, Nashville, Tennessee, September 2013.
- [7] L. Bruno and P. Robertson, "WiSLAM: improving FootSLAM with WiFi," in *2011 International Conference on Indoor Positioning and Indoor Navigation*, pp. 1–10, Guimaraes, Portugal, 2011.
- [8] B. Ferris, D. Fox, and N. Lawrence, "WiFi-SLAM using Gaussian process latent variable models," in *Proceedings of the 20th international joint conference on Artificial intelligence*, pp. 2480–2485, Hyderabad, India, January 2007.
- [9] D. Wagner and D. Schmalstieg, "History and future of tracking for mobile phone augmented reality," in *2009 International Symposium on Ubiquitous Virtual Reality*, pp. 7–10, Gwangju, South Korea, 2009.
- [10] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: zero-effort crowdsourcing for indoor localization Anshul," in *Proceedings of the 18th annual international conference on Mobile computing and networking - Mobicom '12*, pp. 293–304, Istanbul, Turkey, 2012.
- [11] O. Woodman and R. Harle, "Pedestrian localisation for indoor environments," in *Proceedings of the 10th international conference on Ubiquitous computing - UbiComp '08*, pp. 114–123, Seoul, Korea, 2008.
- [12] S. Jung, S. Lee, and D. Han, "A crowdsourcing-based global indoor positioning and navigation system," *Pervasive and Mobile Computing*, vol. 31, pp. 94–106, 2016.
- [13] F. Gu, A. Kealy, K. Khoshelham, and J. Shang, "Efficient and accurate indoor localization using landmark graphs," *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLI-B2, pp. 509–514, 2016.
- [14] M. Oussalah, M. Alakhras, and M. I. Hussein, "Multivariable fuzzy inference system for fingerprinting indoor localization," *Fuzzy Sets and Systems*, vol. 269, pp. 65–89, 2015.
- [15] P. Bahl and V. N. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," in *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No.00CH37064)*, pp. 775–784, Tel Aviv, Israel, 2000.
- [16] R. Want, A. Hopper, V. Falcão, and J. Gibbons, "The active badge location system," *ACM Transactions on Information Systems*, vol. 10, no. 1, pp. 91–102, 1992.
- [17] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The Cricket location-support system," in *Proceedings of the 6th annual international conference on Mobile computing and networking - MobiCom '00*, pp. 32–43, Boston, MA, USA, 2000.
- [18] S. Thrun and M. Montemerlo, "The graph SLAM algorithm with applications to large-scale mapping of urban structures," *The International Journal of Robotics Research*, vol. 25, no. 5–6, pp. 403–429, 2006.
- [19] J. Huang, D. Millman, M. Quigley, D. Stavens, S. Thrun, and A. Aggarwal, "Efficient, generalized indoor WiFi Graph-SLAM," in *2011 IEEE International Conference on Robotics and Automation*, pp. 1038–1043, Shanghai, China, 2011.
- [20] J.-g. Park, B. Charrow, D. Curtis et al., "Growing an organic indoor location system," in *Proceedings of the 8th international conference on Mobile systems, applications, and services - MobiSys '10*, pp. 271–284, San Francisco, CA, USA, 2010.
- [21] S. Yang, P. Dessai, M. Verma, and M. Gerla, "FreeLoc: calibration-free crowdsourced indoor localization," in *2013 Proceedings IEEE INFOCOM*, pp. 2481–2489, Turin, Italy, 2013.
- [22] J. Ledlie, J. Park, D. Curtis, A. Cavalcante, and L. Camara, "Mole: a scalable, user-generated WiFi positioning engine," in *2011 International Conference on Indoor Positioning and Indoor Navigation*, pp. 1–10, Guimaraes, Portugal, 2011.
- [23] K. Chintalapudi, A. Padmanabha Iyer, and V. N. Padmanabhan, "Indoor localization without the pain," in *Proceedings of the sixteenth annual international conference on Mobile computing and networking - MobiCom '10*, pp. 173–184, Chicago, IL, USA, 2010.
- [24] Z. Yang, C. Wu, and Y. Liu, "Locating in fingerprint space: wireless indoor localization with little human intervention," in *Proceedings of the 18th annual international conference on Mobile computing and networking - Mobicom '12*, pp. 269–280, ACM Press, Istanbul, Turkey, 2012.
- [25] P. Bolliger, "Redpin - adaptive, zero-configuration indoor localization through user collaboration," in *Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments - MELT '08*, pp. 55–60, San Francisco, CA, USA, 2008.
- [26] P. Mirowski, T. K. Ho, S. Yi, and M. MacDonald, "Signal-SLAM: simultaneous localization and mapping with mixed WiFi, Bluetooth, LTE and magnetic signals," in *International Conference on Indoor Positioning and Indoor Navigation*, pp. 1–10, Montbeliard-Belfort, France, 2013.
- [27] C. Wu, Z. Yang, Y. Liu, and W. Xi, "WILL: wireless indoor localization without site survey," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 4, pp. 839–848, 2013.
- [28] S. Sorour, Y. Lostanlen, S. Valaee, and K. Majeed, "Joint indoor localization and radio map construction with limited deployment load," *IEEE Transactions on Mobile Computing*, vol. 14, no. 5, pp. 1031–1043, 2015.
- [29] C. Song and J. Wang, "WLAN fingerprint indoor positioning strategy based on implicit crowdsourcing and semi-supervised learning," *ISPRS International Journal of Geo-Information*, vol. 6, no. 11, p. 356, 2017.
- [30] S. Jung, B. Moon, and D. Han, "Unsupervised learning for crowdsourced indoor localization in wireless networks," *IEEE Transactions on Mobile Computing*, vol. 15, no. 11, pp. 2892–2906, 2016.
- [31] S.-H. Jung and D. Han, "Automated construction and maintenance of Wi-Fi radio maps for crowdsourcing-based indoor positioning systems," *IEEE Access*, vol. 6, pp. 1764–1777, 2018.
- [32] Y. Ji, S. Biaz, S. Pandey, and P. Agrawal, "ARIADNE: a dynamic indoor signal map construction and localization system," in *Proceedings of the 4th international conference on Mobile systems, applications and services - MobiSys 2006*, pp. 151–164, Uppsala, Sweden, 2006.
- [33] I. H. Alshami, A. Ahmad, and S. Sahibuddin, "Automatic WLAN fingerprint radio map generation for accurate indoor positioning based on signal path loss model," *ARNP Journal of*

- Engineering and Applied Sciences*, vol. 10, no. 23, pp. 17930–17936, 2015.
- [34] J. Talvitie, M. Renfors, and E. S. Lohan, “Distance-based interpolation and extrapolation methods for RSS-based localization with indoor wireless signals,” *IEEE Transactions on Vehicular Technology*, vol. 64, no. 4, pp. 1340–1353, 2015.
- [35] S.-S. Jan, S.-J. Yeh, and Y.-W. Liu, “Received signal strength database interpolation by kriging for a Wi-Fi indoor positioning system,” *Sensors*, vol. 15, no. 9, pp. 21377–21393, 2015.
- [36] L. Peng, J. Han, W. Meng, and J. Liu, “Research on radio-map construction in indoor WLAN positioning system,” in *2010 First International Conference on Pervasive Computing, Signal Processing and Applications*, pp. 1073–1077, Harbin, China, 2010.
- [37] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury, “No need to war-drive: unsupervised indoor localization,” in *Proceedings of the 10th international conference on Mobile systems, applications, and services - MobiSys '12*, pp. 197–210, Low Wood Bay, Lake District, UK, 2012.
- [38] C. Dinh, D. Tantinger, and M. Struck, “Automatic emergency detection using commercial accelerometers and knowledge-based methods,” in *2009 36th Annual Computers in Cardiology Conference (CinC)*, pp. 485–488, Park City, UT, USA, 2009.
- [39] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. T. Campbell, “A survey of mobile phone sensing,” *IEEE Communications Magazine*, vol. 48, no. 9, pp. 140–150, 2010.
- [40] X. Qu, *Landmark Based Localization : Detection and Update of Landmarks with Uncertainty Analysis*, University Paris-Est, 2016.
- [41] F. Gu, K. Khoshelham, J. Shang, F. Yu, and Z. Wei, “Robust and accurate smartphone-based step counting for indoor localization,” *IEEE Sensors Journal*, vol. 17, no. 11, pp. 3453–3460, 2017.
- [42] F. Li, C. Zhao, G. Ding, J. Gong, C. Liu, and F. Zhao, “A reliable and accurate indoor localization method using phone inertial sensors,” in *Proceedings of the 2012 ACM Conference on Ubiquitous Computing - UbiComp '12*, pp. 421–430, Pittsburgh, Pennsylvania, 2012.
- [43] J. Torres-Sospedra, R. Montoliu, S. Trilles, Ó. Belmonte, and J. Huerta, “Comprehensive analysis of distance and similarity measures for Wi-Fi fingerprinting indoor positioning systems,” *Expert Systems with Applications*, vol. 42, no. 23, pp. 9263–9278, 2015.

Research Article

Improved Capacitive Proximity Detection for Conductive Objects through Target Profile Estimation

Fan Xia, Umme Zakia, Carlo Menon, and Behraad Bahreyni 

School of Mechatronic Systems Engineering, Simon Fraser University, Surrey, BC, Canada V3T 0A3

Correspondence should be addressed to Behraad Bahreyni; behraad@ieee.org

Received 16 April 2019; Revised 4 July 2019; Accepted 29 July 2019; Published 8 September 2019

Guest Editor: Sidike Paheding

Copyright © 2019 Fan Xia et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The accuracy of a capacitive proximity sensor is affected by various factors, including the geometry and composition of the nearby object. The quantitative regression models that are used to seek out the relationship between the measured capacitances and distances to objects are highly dependent on the geometrical properties of the objects. Consequently, the application of capacitive proximity sensors has been mainly limited to detection of objects rather than estimation of distances to them. This paper presents a capacitive proximity sensing system for the detection of metallic objects with improved accuracy based on target profile estimation. The presented approach alleviates large errors in distance estimation by implementing a classifier to recognize the surface profiles before using a suitable regression model to estimate the distance. The sensing system features an electrode matrix that is configured to sweep a series of inner-connection patterns and produce features for profile classification. The performance of the sensing modalities is experimentally assessed with an industrial robot. Two-term exponential regression models provide a high degree of fittings for an object whose shape is known. Recognizing the shape of the object improved the regression models and reduced the close-distance measurement error by a factor of five compared to methods that did not take the geometry into account. The breakthroughs made through this work will make capacitive sensing a viable low-cost alternative to existing technologies for proximity detection in robotics and other fields.

1. Introduction

The demand for industrial robots has accelerated considerably due to the ongoing trend toward automation and continued innovative technical improvements in industrial robots within the past decade. Despite significant advances in the field of automation, human intelligence is superior in terms of reasoning, comprehension, vision, and ingenuity. Robots and humans present complementary features for the development of manufacturing processes; therefore, the close cooperation of human and machine is highly demanded [1]. A great deal of attention has been paid to the human workers' safety as collisions between the worker and the robotic manipulators can be extremely dangerous. The data on industrial robot-related fatalities indicate that safety remains a major concern, especially because the human operators are by necessity physically close to mechanical arms or vehicles [2]. One highlighted approach that modifies the robot's

trajectory based on safety zones or separation distances has shown its superiority in practical applications. In this technique, nonintrusive sensors for distance measurement and localization are critical.

Among the different types of available proximity sensors, the capacitive sensors are appealing for industrial human-robot interaction applications due to their capability of detecting the presence of most obstacles with large coverage and accurately measuring small gaps with fast response [3]. Sensors used for safety applications should have a long detection range and high accuracy in order to secure proper response before colliding with nearby objects. A capacitive warning system developed in [4] can apply the brake and turn off the chainsaw to avoid the harmful effects at a distance up to 15 cm. The electrode field sensing method is also used in [5] to provide a sense of "pretouch" for a grasping system. In this work, multiple capacitive sensors were used so that the robotic hand could be guided from a distance up

to about 12 cm. However, these systems are not able to provide an accurate distance estimation; thus, a more intelligent control of the robot cannot be achieved.

Alternatively, some passive capacitive sensing systems were designed for estimating the human's location. Platypus, proposed in [6], was claimed as the first system to localize and identify people remotely and passively. By using a 6-sensor array covering an area of 2 m by 2.5 m, their system demonstrated a localization error of 16 cm and identification accuracy of 75% for 30 users. Another passive positioning system described in [7] was based on measuring the capacitance between multiple floor tiles and a receiving electrode. It can locate a standing person with 15 cm accuracy and track a walking human with 41 cm accuracy. Though with a large detection range and high accuracy, the large electrode areas and low speed limit the application of these systems in human-robot interaction fields.

A major challenge for capacitive sensors is the nonlinear response that makes the extraction of useful information from measurements a difficult computational problem. Moreover, the geometrical parameters of an approaching object and the environmental interferences will add additional unknown contributions to the capacitive responses.

Models that determine continuous properties are physically motivated. An approach for continuous 3D finger and hand localization problems with capacitive sensors was explained in [8]. Random decision forests were used in their system for regression. However, their device suffered from environmental interferences and limited detection range. In addition, the applications of proximity sensors always involve more than one detected state, so classifiers are frequently used. A Support Vector Machine (SVM) is one of the most frequently used methodologies for extracting information from sensing data as it can handle both linear tasks and more complex problems [9]. Laput et al. used SVM for touch recognition of uninstrumented, electrical, and electro-mechanical objects [10]. Across the 24 classes, SVM achieved an overall accuracy of 97.9% and 18 out of 24 tested objects reached the accuracy of 100%. SVM was also used in capacitive fingerprinting for user differentiation [11]. For single finger touches, the SVM yielded an all-pair average accuracy of 97.3%, while in distinguishing between users performing a variety of gestures, the achieved average accuracy was 97.8%.

Explorations on the design of a tri-mode capacitive proximity sensor have been demonstrated previously in [12]. This work is aimed at improving the accuracy of distance measurements by alleviating the issue of ambiguity due to an object profile. The focus of this work is on the detection of a nearby metallic object which is a common situation in various human-robot interaction domains. This is particularly important on factory floors where such sensors are needed to improve worker safety. The proposed capacitive proximity sensing system inherently involves inferring information from measured data, either continuously (i.e., for estimating the human's position) or discretely (i.e., for recognizing the shape of the object). We employ a classifier to identify the shape of an approaching object in conjunction with a library of regression models to improve the distance measurement accuracy.

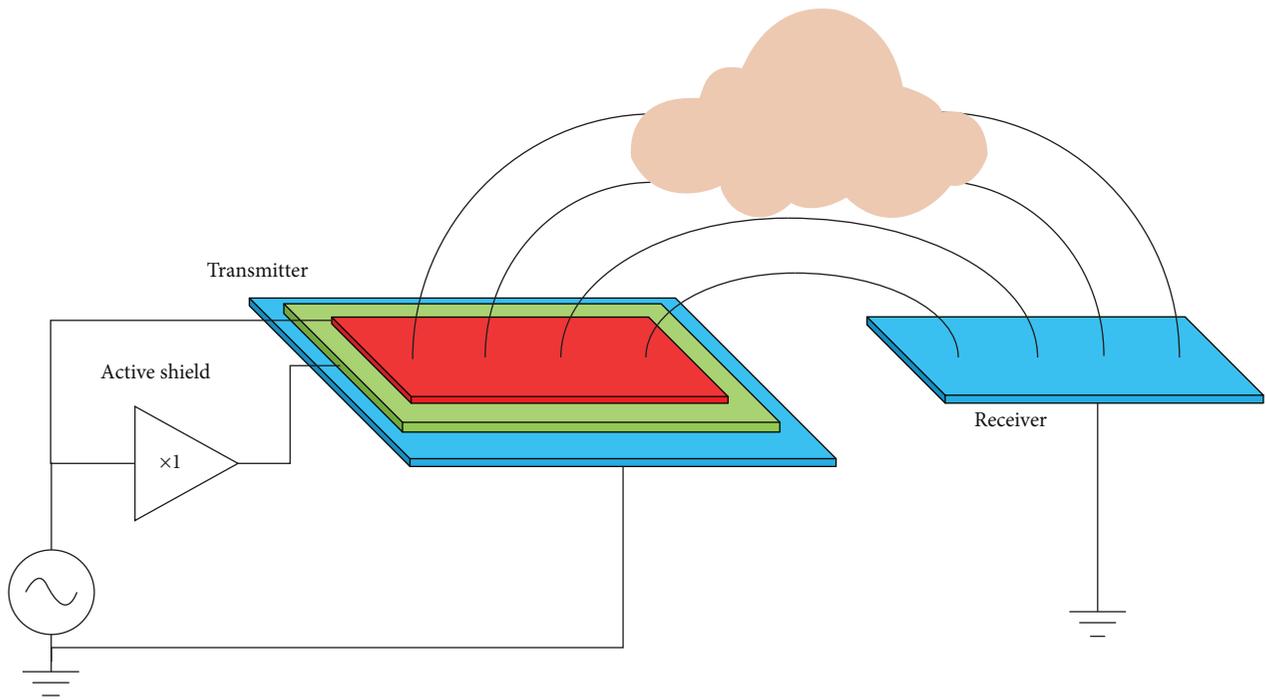
2. Materials and Methods

The general term electric field sensing is used to refer to a family of noncontact measurements that can be made with slowly varying electric fields [13]. Some of these measurements are lumped together under the rubric "capacitive sensing," in which a low-frequency voltage signal is applied to the transmit electrode. A displacement current flows from the transmitter to receivers through the mutual capacitors between them. Optimal hardware design and sensing configurations rely on a good understanding of fundamental principles and trade-offs.

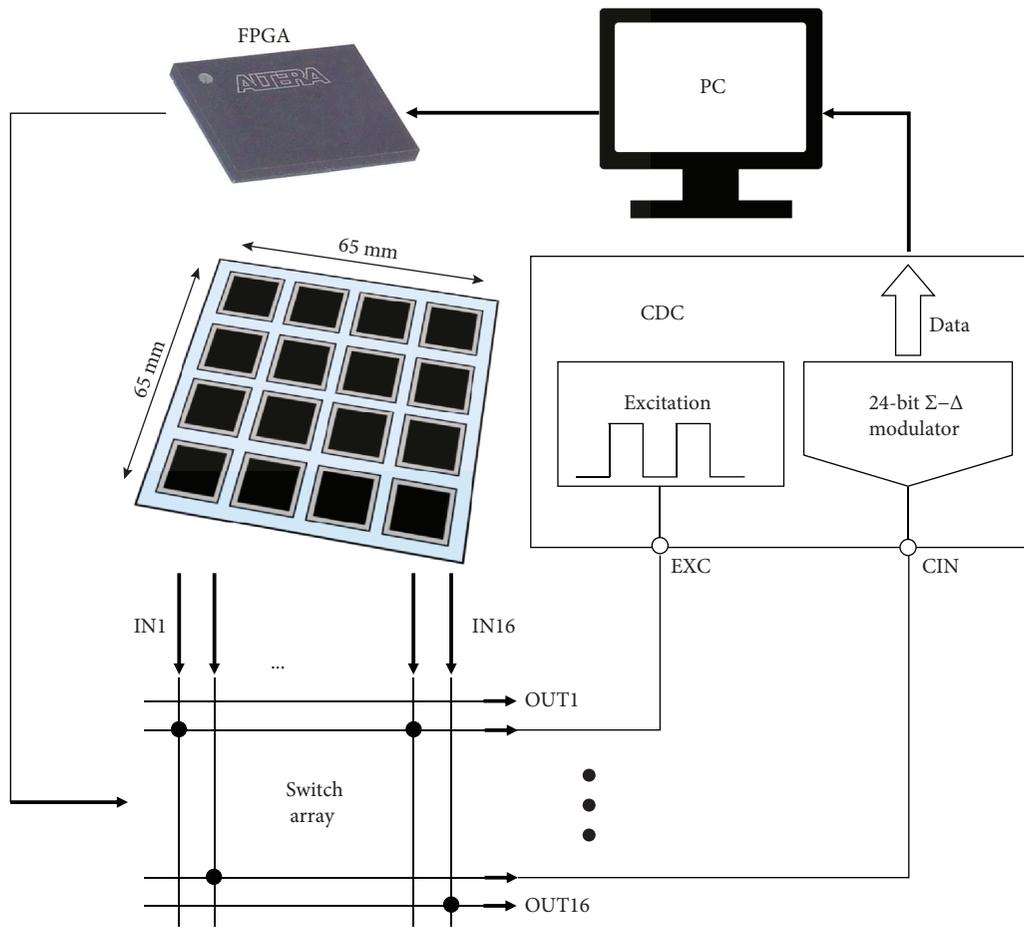
A two-electrode measurement setup, as depicted in Figure 1(a), is adopted as the basic architecture of this work. It is a three-terminal measurement where neither the transmitter nor the receiver is in contact with the object [14]. If the electric field distribution is mapped using N -independent electrodes, there will be $N(N-1)$ measurements resulting in $(1/2)N(N-1)$ -independent mutual capacitance values due to the symmetrical capacitance matrix. Moreover, single electrode-pair measurements can be combined with multiplexing methods allowing parallel access to multiple transmitters/receivers at the same time. In the context of the applications where the designed sensor should be fitted onto a piece of a working garment, the electrode matrix area is restricted to $6.5\text{ cm} \times 6.5\text{ cm}$. A 4×4 electrode matrix yielding 16 independent electrodes (see Figure 1(b)) is used as it meets the functional requirements with an acceptable complexity. In addition, a grounded backplane is placed underneath the sensor substrate to avoid the undesired detection from the backside. A 4×4 electrode array is added between the ground and the top electrodes. These electrodes, whose individual dimensions are slightly larger than those on the top surface and are centred below them, are driven with the same drive signal (after a buffer) that is applied to the sense electrodes. This active shield will push the electric field from the sense electrodes to the space above them, significantly improving the detection range of the sensor [12].

In order to implement different electrode configurations and extract the capacitive responses, the proximity sensing system composes five building blocks as depicted in Figure 1(b). The 4×4 electrode matrix together with its active shielding electrode array and the grounded backplane constitute the core sensing section. Two analog switch arrays are used for physical connections within sensing and active shielding matrices. A digital controlling module is mapped on a field programmable gate array (FPGA) to create the required signals to program the switch arrays. A capacitance-to-digital converter measures the capacitive responses and digitizes the data. Finally, a personal computer (PC) is used to collect and process the data.

The designed sensor has 16 independent electrodes, providing many possible configurations to generate the fringing electric fields so that multiple functionalities can be realized with the same sensing platform. The performance of the array-structured proximity sensor typically depends on the number and arrangement of electrodes that form the transmitter and receiver. In our previous work, three



(a)



(b)

FIGURE 1: (a) Detection of an object that interferes with electric field distribution between electrodes and (b) block diagram of the sensing system.

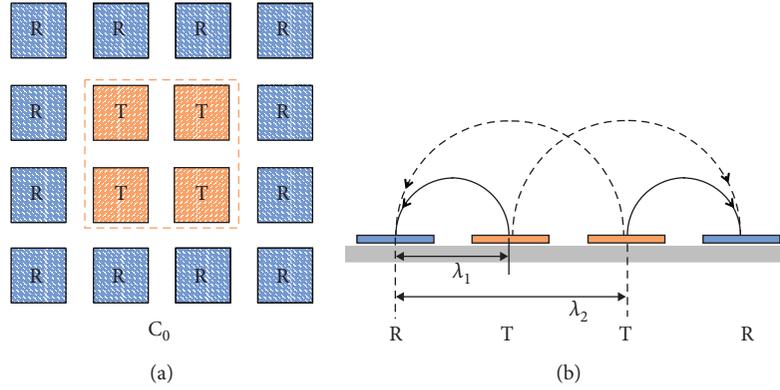


FIGURE 2: Electrode configuration for distance measurement. “T”: transmitter; “R”: receiver. (a) Four central electrodes are shorted to each other to form the transmitter, and all the rest electrodes are connected to work as the receiver. The generated capacitor is C_0 . (b) Cross-section view of the electric field formed by the electrode matrix. The larger the spatial wavelength (λ), the farther the electric field penetrates to the space and the longer the detection range can be achieved.

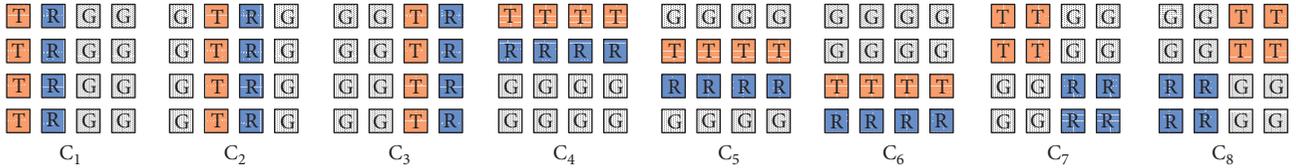


FIGURE 3: Sweeping pattern I that scans eight mutual capacitors C_1 to C_8 . Electrode fields are generated between two adjacent columns/rows (C_1 - C_6) as well as diagonal electrodes (C_7 and C_8). “G”: grounded connection.

electrode-connecting types that result in comb-structure, adjacent two-rectangular configuration, and two-edge patterns have been investigated [12]. According to Ye et al., electrodes with a spiral shape perform well in terms of dynamic range and sensitivity distribution homogeneity compared to those with the other shapes [15]. In this work, electrodes are connected as a symmetric structure, depicted in Figure 2(a), to mimic the spiral shape. Each letter in Figure 2(a) represents the electrical destination of the electrode, i.e., “T” for transmitter, “R” for receiver, and “G” for grounded connections. Electrodes labeled by the same letter are shorted to each other.

For distance measurement, the excitation signal is applied to the transmitters, i.e., the middle four electrodes, resulting in the mutual capacitance C_0 . The generated electric field is demonstrated in Figure 2(b). Penetration depth is a parameter that indicates how quickly the electrical field weakens with the distance [16]. A greater penetration depth will lead to a longer detection range. The depth that the electric field penetrates to the space is roughly proportional to the spatial wavelength λ , which is defined as the distance between two consecutive electrodes of the same polarity [17]. This electrode configuration provides two different spatial wavelengths, and the capacitive response is correlated with the distance.

One challenge of using capacitive sensors is that they are susceptible to shape and size of the object: a small object that is close to the sensor might result in the same response as a larger object at a further distance [18]. This confusion makes

it difficult to infer high-resolution information from the measured capacitances. Therefore, surface profile recognition is required to improve the accuracy in distance evaluation.

For the purpose of distinguishing different object shapes, more detailed information is required. The electrode matrix is programmed to form different distributed proximity sensors so that the nearby surroundings can be monitored by sweeping these electrode configurations.

The acquired capacitance values constitute the inputs of classification tools for profile recognition. Management of electrode sweeping modes plays an essential role in determining the shape information. The key point in reconstructing the desired obstacles is to distinguish the differences in terms of sensor responses among different circumstances. Two different sweeping approaches to mesh the close surroundings are proposed and examined in this work.

The first approach scans eight mutual capacitors C_1 to C_8 as described in Figure 3. The electric fields between adjacent columns and adjacent rows are generated by selecting one column/row of electrodes as the transmitter and a neighbouring column/row of electrodes as the receiver. That results in the first six matrices in Figure 3. Six individual capacitors are formed, and the nearby space can be well meshed along the X - Y plane. Moreover, to reduce the obscurity brought by objects with symmetrical appearance, two more capacitors between diagonal electrodes (i.e., C_7 and C_8 in Figure 3) are also measured. In this attempt, every single sweeping cycle of the sensor measures eight independent capacitors or features.

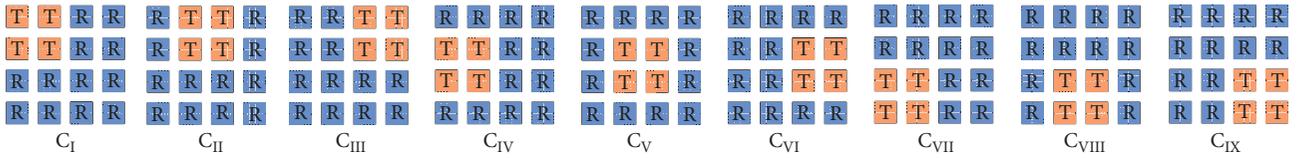


FIGURE 4: Sweeping pattern II that measures nine mutual capacitors C_I to C_{IX} . The generated electrode fields emit from the four neighbouring electrodes (connected as the transmitter) in every direction and fall to all the rest electrodes (collected as the receiver).

The inspiration for the second approach comes from the spiral-shaped connection structure: the electric field propagates from four neighbouring electrodes in every direction and terminates on the rest electrodes. The environment around the sensor can be meshed comprehensively when employing this mode to the whole sensing matrix as demonstrated in Figure 4. Each of the nine connection patterns generates multiple electric fields resulting in different spatial wavelengths and different penetration depths. By combining the capacitive responses (C_I to C_{IX}) from each sweeping cycle and using proper classifiers, the surface profile of the nearby object can be estimated.

2.1. Operation Flow. Distance estimation, the primary task for this work, is quantitative. The desired results take on numerical values making it a regression problem. The method is to build a regression model, which is a prediction equation that enables predicting response for given inputs with small errors [19]. A series of experiments are conducted in order to gather the required data to create regression models for objects with specific shapes. The actual shape of an object is approximated by a simple geometric shape that is easier to process. A plate, sphere, and cylinder are selected as representatives for the most frequently encountered shapes in an industrial working space. The experimental process is moving the selected object continuously from 1 cm to 20 cm away from the sensor and recording the values of the capacitor C_0 . The regression model is then built by using the measured capacitances as inputs and the corresponding distances as outputs. Consequently, a library that contains three prediction equations corresponding to the selected geometries is created.

Regression provides an effective way to deduce the distance information from measured capacitances. However, picking the proper regression model depends on the perception of the object's shape. The task of profile recognition is qualitative: the variables being predicted (i.e., different shapes) are discrete rather than continuous. Therefore, classification tools are used to take over this job. In order to provide enough information for shape recognition, the inputs of the classifiers are the combinations of the capacitances obtained from electrode configuration sweeping. These inputs are experimentally acquired by placing the objects at different locations within the detectable range and collecting capacitive responses generated by the sweeping patterns in Figures 3 and 4. The classifier output is a best estimate for the shape of the object (i.e., plate, sphere, or cylinder). The data is used for supervised training of the classifiers with the raw inputs and preknown targets.

The previous two steps, namely, building the regression model library and training the classifier, are completed offline before using the sensing system in real applications. In practice, when an object enters the detection range, the sensor is programmed to work in the scanning mode to determine the most probable shape information. Then, the sensor is switched to the spiral-shaped electrode connection to measure the capacitive responses caused by the approaching object. Lastly, the distance is calculated by selecting the proper regression model according to the object's profile. The operation flow of the presented sensing system is demonstrated in Figure 5.

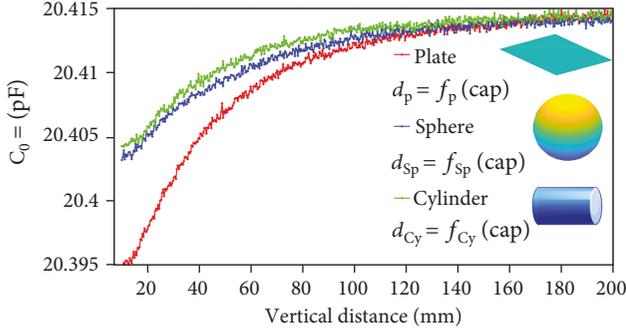
2.2. Data Processing. The goal of data processing is to obtain desired information from raw measurement results. Such a process is inherently interactive and iterative [20]. Preparing input data and selecting the most suitable classifiers are carried out with attention to the problem domain.

Data from capacitive proximity sensors is susceptible to internal and external parasitics, environmental interferences, noise, and potentially other error sources. Therefore, pre-processing of the data, including cleaning, normalization, and integration, is the foremost step before running the analysis [21].

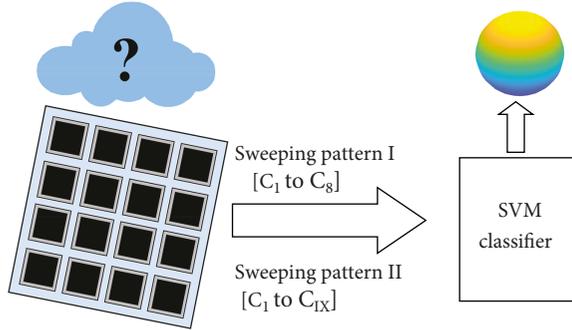
The environmental interferences add both high-frequency noise and low-frequency random walks to the capacitive response. The data cleaning process is to remove noise and correct for drift in the "dirty" raw dataset. More specifically, a DC notch filter (to eliminate slow environmental changes such as humidity and temperature variations) together with a low-pass averaging filter with the cut-off frequency of 20 Hz (to reduce high-frequency noise and interference) was used to compensate these non-ideal effects. Additional details and implementation can be found in [12].

One intermediate step between collecting raw measured capacitive values and applying the statistical learning tools is scaling the data. This step is important and almost required for most learning algorithms because scaling can avoid attributes in greater numeric ranges dominating those in smaller numeric ranges. In addition, it can also avoid numerical difficulties during the calculation. The applied method is linearly rescaling the range of all the input data to the range of $[-1, +1]$. Given the maximum capacitive value C_{\max} and the lower bound C_{\min} , a normalized value is calculated from $C_N = ((2 \times (C - C_{\min})) / (C_{\max} - C_{\min})) - 1$.

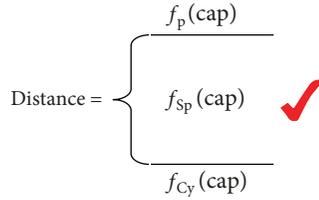
The final step for preparing the input datasets is data integration that merges data from multiple measurements. Every scanned capacitor has its contribution



Step1: create regression a model library for the three objects



Step2: recognize the shape of an approaching object



Step3: select the proper regression model to calculate the distance

FIGURE 5: Operation flow for improved distance measurement. First, create separate regression models for all the objects with experimental data (total capacitance is shown at the top, but several sweeping patterns are generated per measurement). When an object appears, use a classifier to recognize its surface profile based on one of the two sweeping patterns. Lastly, select the proper regression model depending on the classification result for advanced distance estimation.

to reconstructing the profile of a nearby object, so all the capacitances acquired by one complete sweeping cycle are combined together as one single input which can be referred to as a “feature vector.” For the first approach that measures eight mutual capacitors as depicted in Figure 3, the generated feature vectors are eight-dimensional (i.e., $\{C_1, C_2, \dots, C_8\}$), whereas the second scanning method, as explained in Figure 4, results in nine-dimensional feature vectors (i.e., $\{C_I, C_{II}, \dots, C_{IX}\}$).

Interpreting the experimental results requires the analysis of complex, multivariate, and multidimensional data. One approach that has grown popularity is the use of machine learning algorithms to train classifiers to decode behaviors and information of interest from the experimental data [22]. Thanks to the efficient pattern recognition performance for the nonlinear multiclass scenarios, Support Vector

Machines (SVM), a kernel-based learning method, are adopted in this work [23].

Originally developed for binary classification problems, SVM uses maximal margin hyperplanes to define decision boundaries separating data points of different classes. The equation of a linear decision surface is

$$w_0^T x + b_0 = 0, \quad (1)$$

where x is the input feature vector and w_0 and b_0 are the optimal weight vector and bias, respectively. The discriminant function is expressed as

$$g(x) = w_0^T x + b_0, \quad (2)$$

and provides an algebraic measure of the geometric distance from any x to the optimal hyperplane [24]. The input x can be described through

$$x = x_p + r \frac{w}{\|w_0\|}, \quad (3)$$

where x_p is the normal projection of x onto the optimal hyperplane and r is the algebraic distance such that r is positive when x_i belongs to the class of +1 and negative otherwise.

Given the dataset (x, l) where l is the target class, parameters (w_0, b_0) must satisfy the following constraints:

$$\begin{aligned} w_0^T x_1 + b_0 &\geq 0, & \text{for } l_i = +1, \\ w_0^T x_1 + b_0 &\leq 0, & \text{for } l_i = -1. \end{aligned} \quad (4)$$

The particular training data points for which one of the constraints is satisfied with the equality sign are the “support vectors.” Maximizing the margin of separation between classes is equivalent to minimizing the Euclidean norm of the weight vector w . This distinct property makes the SVM an effective tool in pattern recognition applications. The previous equations describe the foundation of SVM that classifies a binary problem which can be linearly separated. For more complex tasks, Gaussian kernel [25] is a reasonable first choice. This kernel function nonlinearly maps samples into a higher dimensional space allowing it to handle the cases where the relationship between class labels and inputs is nonlinear. The linear function is a special case of the Gaussian kernel.

The problem in this study requires the discrimination for more than two categories. So SVM is extended so that it can be suitable for more general cases where an arbitrary number of classes is important. Among all the proposals for modifying the SVM to the K -class case, the two most popular approaches are the “one versus one” (1V1) and “one versus rest” (1VR) [26]. The 1V1 approach is a pairwise decomposition. It evaluates all possible pairwise classes and therefore constructs $K(K-1)/2$ individual binary classifiers. Applying each classifier to a test sample will generate one vote to the winning class. The data point will be assigned to the class with the most votes. By contrast, the 1VR approach only constructs K separate binary classifiers for the K -class problem.

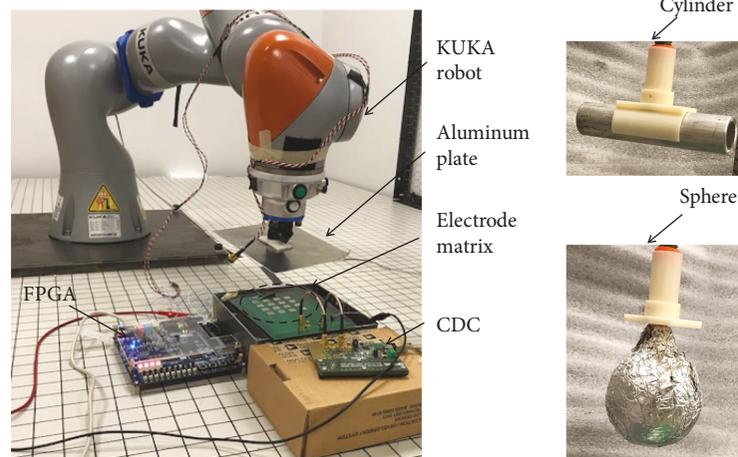


FIGURE 6: Experimental setup with an industrial robot. The electrode matrix is controlled by the FPGA to accomplish desired configurations, the robot is programmed to move the attached object based on predefined trajectories, and the capacitances are measured and quantified by the CDC. Three aluminum objects are a plate, a ball, and a cylinder.

Each time, one class is compared to all the remaining classes as a whole. The i^{th} classifier is trained using the data from the i^{th} class as positive examples and all the other data as negative samples. During testing, the class label is determined by the classifier that results in the maximum output value. It is noteworthy that there is no clear evidence that the 1V1 method can achieve higher accuracy compared to alternative multi-class SVM methods, but Hsu and Lin argue that 1V1 is more practical due to its faster training process [27].

The core structure of the presented sensing system together with the complementary electronics is fabricated on a four-layer printed circuit board (PCB). The top three layers of the PCB serve as carriers for the 4×4 electrode matrix, active shielding matrix, and backplane shielding. The analog switch array chips (AD75019 from Analog Devices) and required electronic components including bypass capacitors and resistors are fabricated on the bottom side. The FPGA board adopted in this work is the Terasic DE1 board featuring the ALTERA Cyclone II 2C20 FPGA device. In addition, an AD7746 CDC chipset is acquired to measure the capacitors, quantize the capacitances, and exhibit results in terms of both capacitive values and digital strings via a standard communication interface. The sampling rate it can achieve is 90.9 Hz.

The performance and behavior of the designed capacitive sensing system were investigated with an industrial robot KUKA LBR iiwa. The robot is a lightweight robot with a 7-axis jointed arm. All drive units and current-carrying cables are installed inside the robot so that it can autonomously move and transport objects. It has the capability of orienting itself independently in its surroundings and moving into positions for automation tasks with millimeter precision [28].

Ideally, the test objects would be electrically grounded during the system operation. However, to better resemble the actual working conditions during the experiments, the objects were isolated electrically from the robot or the circuitry (i.e., electrically floating).

3. Results and Discussion

To verify the distance measurement capability and the shape classification feature, a series of prototypical experiments are conducted. The established apparatus and the experimental setup are illustrated in Figure 6. The robot is programmed to perform predefined movements on the attached object, and the value d which represents the distance from the object to the sensor can be controlled with high accuracy. Three metallic objects, namely, a plate, a ball, and a cylinder, are used as actual instances of the three shapes of interest. The size of the plate is $16 \text{ cm} \times 13 \text{ cm}$, the diameter of the ball is 6.5 cm, and the radius and the length of the cylinder are 1.5 cm and 10 cm, respectively. During the experiments, the plate was brought towards the electrode at different inclination angles. The ball and cylinder were positioned at different lateral and vertical positions above the electrode array. The cylinder was brought in at different in-plane angles (parallel to the electrode array). With all these variations in positioning and relative angles, the objective was to improve the accuracy of distance estimation from the electrode array to the closest point on the object by taking into account the object geometry regardless of approaching angles or lateral position of the object relative to the electrode array.

3.1. Shape Recognition. The shape recognition is achieved by performing a series of dynamic experiments and using SVM as the classifier [29]. The classification accuracy is examined with respect to the electrode sweeping pattern.

The dynamic experiment is about automatically sweeping the inner electrode configurations while moving one of the objects to nine different locations. At each location, the object is held by the robot for a short period of time allowing 100 times of the complete sweeping cycle. For the first scan pattern, each sweeping cycle measures eight mutual capacitors as described in Figure 3, and the capacitive responses from one complete cycle can be fused to one eight-

TABLE 1: SVM testing accuracy with a grid search for cost parameter and gamma (%).

β	γ					
	0.5	1	2	3	4	5
Dataset I: inputs are 8-dimensional based on sweeping mode I (Figure 3)						
0.5	66.67	66.67	66.67	66.67	82.33	96.22
1	66.67	66.67	82.33	100	100	100
2	66.67	82.33	100	100	100	100
3	66.67	100	100	100	100	100
4	82.33	100	100	100	100	100
5	82.33	100	100	100	100	100
Dataset II: inputs are 9-dimensional based on sweeping mode II (Figure 4)						
0.5	66.67	66.67	66.67	66.67	66.67	69.67
1	66.67	66.67	66.67	75.56	82.67	82.67
2	66.67	66.67	82.67	82.67	82.67	82.67
3	66.67	75.56	82.67	82.67	82.67	82.89
4	66.67	82.67	82.67	82.67	82.67	82.67
5	69.67	82.67	82.67	82.89	82.67	82.67

dimensional feature vector $\{C_1, C_2, \dots, C_8\}$. Similarly, the second sweeping method results in nine-dimensional feature vectors $\{C_I, C_{II}, \dots, C_{IX}\}$ as shown in Figure 4. Consequently, two datasets, each containing 27×100 feature vectors, are acquired for classification. Dataset I, obtained with the first scan pattern, is composed of eight-dimensional feature vectors. Whereas the feature vectors in Dataset II are nine-dimensional. Each dataset is further split into two parts: one contains 18×100 feature vectors to train the SVM classifier and the rest 9×100 feature vectors are used for testing.

The test accuracy is determined by two kernel function parameters: cost parameter (β) and gamma (γ). It is not known beforehand which combination of the two parameters will lead to the best result for a specific problem. Consequently, a parameter search procedure is done separately on both datasets. Growing sequences of β and γ are tried for training the model, and the identified most efficient parameter pair is the one that leads to the highest testing accuracy. The involved values of β and γ for the grid search are the same: 0.5, 1, 2, 3, 4, and 5. The classification accuracies with respect to the cost parameter and gamma are summarized in Table 1. For both datasets, the testing accuracy starts from 66.67% when β and γ are both small. With the increase in the parameters, Dataset I shows its superiority: the classification accuracy reaches 100% when the value of either β or γ extends 2. That indicates for this specific task, and using Dataset I as the input for SVM is able to classify the capacitive responses to desired shape categories. After performing this classification step, the determined shape information is used for selecting the proper regression model.

3.2. Distance Estimation. The distance is measured with respect to the three objects so that the regression model library can be built. The distance range of interest in this work is 1 cm to 20 cm.

The implementation of the measurement can be divided into three steps: Configure the inner connection of the electrode matrix to be spiral-shaped by programming the FPGA board. Then, the robot moves one of the objects from 1 cm to 20 cm away from the surface of the sensor at a constant speed. The capacitive responses are recorded at the same time. Lastly, the acquired capacitances and their corresponding distance values are used to generate the desired regression model. This procedure is repeated for the other two objects so that three regression models are created.

The raw experimental capacitive samples for the three objects are plotted with black dots in Figures 7(a)–7(c). Note that the capacitance measured between the electrode groups decreases as the objects approached the electrode array. This is due to the fact that at closer distances some of the current from the driven electrodes will flow to the grounded metal object that was used. Fitted curves that stand for the mathematical formulas describing the relationship between a capacitance and its corresponding distance are denoted by the red lines. In this case, two-term exponential regression models are adopted to provide a high degree of fittings. “High degree” indicates the model has a small random error component and takes as much proportion of variance into account so that it is more useful for prediction. From the statistical point of view, the sum of squared errors of prediction (SSE) that measures the total deviation of the response values from fit values should be close to 0 and the coefficient of determination (R -squared or R^2) be close to 1 [30]. The regression model expressions together with corresponding coefficients and goodness indices (i.e., SSE and R^2) are also denoted in the figures.

Based on the analysis of the regression models, they can provide predictions on the distance with high confidence under the condition that the shape of the approaching object is preknown. However, when comparing the data acquired by different objects, it can be discovered that both the absolute capacitive values and the changing rates are different. To eliminate the error caused by drift, an offset cancelling procedure is performed: the plate’s asymptotic capacitance, measured when the plate is put infinitely far from the sensor (50 cm in this work as the capacitances stop changing), is taken as a reference. Then, shift the capacitance vs. distance curves of the other two objects to the same ending point by adding or subtracting a constant. The experimental capacitances after offset cancelling as functions of distances for the three objects are summarized in Figure 7(d). From which, it can be observed that the same capacitance value can result in very different distance values without knowledge of the object’s shape. This fact will lead to a decreased distance evaluation accuracy or even a wrong detection.

A more precise analysis of the distance errors $e(d)$ is demonstrated in Figure 7(d). Use the capacitive responses measured with the plate as references, and feed the capacitances to the three regression models to predict the distances $d'(P)$ for the plate, $d'(Sp)$ for the sphere, and $d'(Cy)$ for the cylinder. At a certain distance, the estimation of the distance error $e(d)$ can be calculated as the absolute difference between the predicted distance d' and the actual distance d . To get a clearer vision on how the distance error would affect

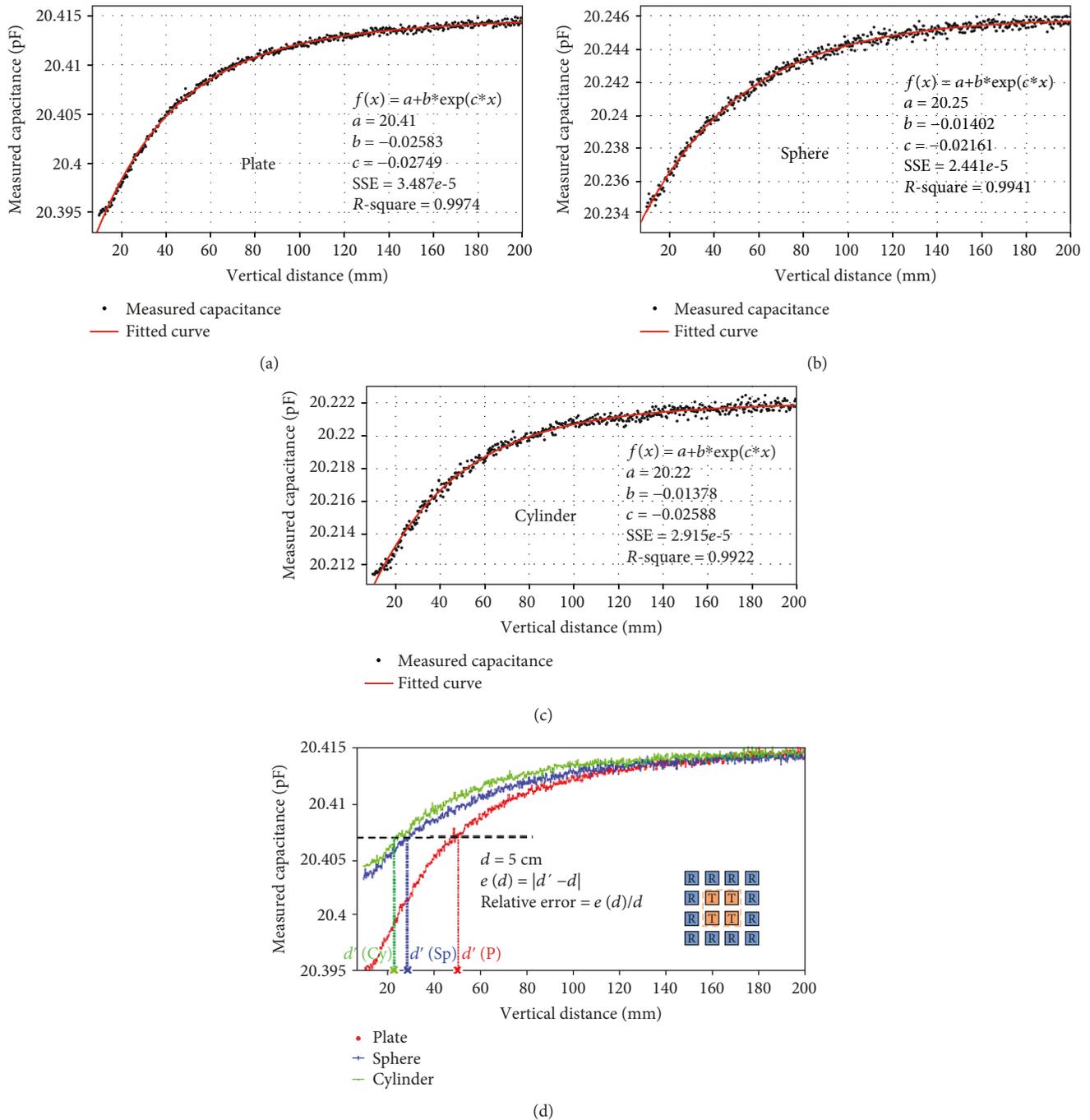


FIGURE 7: (a) Measured raw capacitive responses and the fitting curve for the plate. (b) Measured raw capacitive responses and the fitting curve for the sphere. (c) Measured raw capacitive responses and the fitting curve for the cylinder. (d) A comparison of capacitive responses for the three objects. An example of the calculation of the absolute and relative errors caused by using the wrong regression model is demonstrated.

the measurement accuracy, the relative error with respect to the distance $e(d)/d$ is also calculated. The relative errors at few different discrete distances are plotted with the bars in Figure 8, and the numerical absolute errors in millimeters are also provided. For the plate, the proper regression model has been selected so that the estimation errors are small: $e(d)$ is less than 1 cm when the distance is below 10 cm. The

increase in error with the distance is due to the worsening of the signal-to-noise ratio for the measurements and is limited to 25%. However, the same model would result in significantly larger errors when applied to the other objects, especially in short distances (i.e., less than 5 cm) such that the uncertainty in the measurements is comparable to the actual distances (i.e., the relative errors are around 100%).

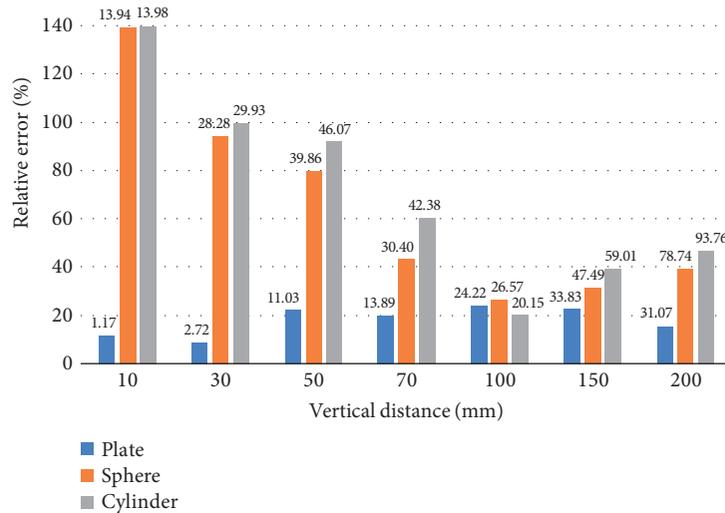


FIGURE 8: Relative errors for the three objects using the regression model for the plate as the reference. Bar labels are the numerical absolute error at that point.

However, at short ranges, a higher detection accuracy is in fact needed in most applications, hence necessitating profile recognition.

4. Conclusions

In this paper, a capacitive proximity sensing system with improved distance measurement accuracy is presented. Shunt detecting mode is used in combination with the 4×4 electrode matrix to provide more informative and flexible measurements. Statistical tools are employed for interpreting the experimental capacitive responses: quantitative regression models are built to seek out relationships between capacitances and distances while the SVM classifier is trained to recognize the surface profile of the approaching object. Different electrode sweeping patterns are implemented and compared in terms of classification accuracy.

The performance of the sensing modalities is experimentally assessed with an industrial robot and three objects with different shapes. The relative distance estimation error is limited to 25% under the condition that the proper regression model is selected. Otherwise, the errors can be competitive to the actual distance. The SVM classifier recognized the shape of an object with high accuracy. The classification result is used to choose the most suitable regression model with high confidence. The next step will be involving more objects to enrich the regression model library and collecting data with random locations to generalize the training dataset.

It is possible to expand the capabilities of the system so that one may obtain additional information regarding the nearby objects. For instance, if impedances over a range of frequencies are measured instead of capacitance values at a single frequency, it will be possible to deduce permittivity and conductance information for a nearby object. This additional information, however, would require new hardware with phase-sensitive measurements at different frequencies and collection of suitable training data from different objects and materials.

The sensor system operated robustly against slow environmental disturbances with tests producing reliable results over a span of several weeks after the collection of training data. This was in part done through removal of the contributions of slow changes by filtering out near DC signals. However, if the operating environment is significantly different from the test setup (e.g., all objects are grounded), the system can be retrained using data collected from the environment.

Data Availability

The collected training and evaluation data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

The authors also acknowledge the support from the Natural Sciences and Engineering Research Council of Canada, Canada Research Chairs program, and Canada Foundation for Innovation. This project was financially supported through the Automotive Partnership Canada grant #APCPJ 459264-13.

References

- [1] J. Krüger, T. K. Lien, and A. Verl, "Cooperation of human and machines in assembly lines," *CIRP Annals*, vol. 58, no. 2, pp. 628–646, 2009.
- [2] A. Bicchi, M. A. Peshkin, and J. E. Colgate, "Safety for physical human–robot interaction," in *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds., pp. 1335–1348, Springer, Berlin, Heidelberg, 2008.

- [3] H. T. Cheng, A. M. Chen, A. Razdan, and E. Buller, "Contactless gesture recognition system using proximity sensors," in *2011 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 149–150, Las Vegas, NV, USA, January 2011.
- [4] B. George, H. Zangl, and T. Bretterklieber, "A warning system for chainsaw personal safety based on capacitive sensing," in *2008 IEEE Sensors*, pp. 419–422, Lecce, Italy, October 2008.
- [5] B. Mayton, L. LeGrand, and J. R. Smith, "An electric field pre-touch system for grasping and co-manipulation," in *2010 IEEE International Conference on Robotics and Automation*, pp. 831–838, Anchorage, AK, USA, May 2010.
- [6] T. Grosse-Puppendahl, X. Dellangol, C. Hatzfeld et al., "Platypus: indoor localization and identification through sensing electric potential changes in human bodies," in *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services - MobiSys '16*, Singapore, June 2016.
- [7] M. Valtonen, J. Maentausta, and J. Vanhala, "TileTrack: capacitive human tracking using floor tiles," in *2009 IEEE International Conference on Pervasive Computing and Communications*, pp. 1–10, Galveston, TX, USA, March 2009.
- [8] M. Le Goc, S. Taylor, S. Izadi, and C. Keskin, "A low-cost transparent electric field sensor for 3d interaction on mobile devices," in *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14*, pp. 3167–3170, Toronto, ON, Canada, April-May 2014.
- [9] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the fifth annual workshop on Computational learning theory - COLT '92*, pp. 144–152, Pittsburgh, PA, USA, July 1992.
- [10] G. Laput, C. Yang, R. Xiao, A. Sample, and C. Harrison, "EM-Sense: touch recognition of uninstrumented, electrical and electromechanical objects," in *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology - UIST '15*, pp. 157–166, Charlotte, NC, USA, November 2015.
- [11] C. Harrison, M. Sato, and I. Poupyrev, "Capacitive fingerprinting: exploring user differentiation by sensing electrical properties of the human body," in *Proceedings of the 25th annual ACM symposium on User interface software and technology - UIST '12*, p. 537, Cambridge, MA, USA, October 2012.
- [12] F. Xia, F. Campi, and B. Bahreyni, "Tri-mode capacitive proximity detection towards improved safety in industrial robotics," *IEEE Sensors Journal*, vol. 18, no. 12, pp. 5058–5066, 2018.
- [13] J. R. Smith, "Field mice: extracting hand geometry from electric field measurements," *IBM Systems Journal*, vol. 35, no. 3.4, pp. 587–608, 1996.
- [14] T. G. Zimmerman, J. R. Smith, J. A. Paradiso, D. Allport, and N. Gershenfeld, "Applying electric field sensing to human-computer interfaces," in *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '95*, pp. 280–287, Denver, CO, USA, May 1995.
- [15] Y. Ye, J. Deng, S. Shen, Z. Hou, and Y. Liu, "A novel method for proximity detection of moving targets using a large-scale planar capacitive sensor system," *Sensors*, vol. 16, no. 5, p. 699, 2016.
- [16] X. B. Li, S. D. Larson, A. S. Zyuzin, and A. V. Mamishev, "Design principles for multichannel fringing electric field sensors," *IEEE Sensors Journal*, vol. 6, no. 2, pp. 434–440, 2006.
- [17] Y. Huang, Z. Zhan, and N. Bowler, "Optimization of the coplanar interdigital capacitive sensor," in *43rd Annual Review of Progress in Quantitative Nondestructive Evaluation*, vol. 36, p. 110017, Atlanta, GA, USA, 2017.
- [18] L. K. Baxter, *Capacitive Sensors: Design and Applications*, IEEE Press, New York, NY, USA, 1997.
- [19] W. Mendenhall and T. Sincich, *A Second Course in Statistics: Regression Analysis*, Prentice Hall, Boston, MA, USA, 7 edition, 2012.
- [20] H. Mannila, "Data mining: machine learning, statistics, and databases," in *Proceedings of 8th International Conference on Scientific and Statistical Data Base Management*, pp. 2–9, Stockholm, Sweden, June 1996.
- [21] "Data collection methods and data preprocessing techniques for healthcare data using data mining," *International Journal of Scientific & Engineering Research*, vol. 8, no. 6, pp. 1131–1136, 2017.
- [22] F. Pereira, T. Mitchell, and M. Botvinick, "Machine learning classifiers and fMRI: a tutorial overview," *NeuroImage*, vol. 45, no. 1, pp. S199–S209, 2009.
- [23] J. A. K. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Processing Letters*, vol. 9, no. 3, pp. 293–300, 1999.
- [24] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, Wiley-Interscience, New York, NY, USA, 2nd edition, 2012.
- [25] S. S. Keerthi and C.-J. Lin, "Asymptotic behaviors of support vector machines with Gaussian kernel," *Neural Computation*, vol. 15, no. 7, pp. 1667–1689, 2003.
- [26] Z. Wang and X. Xue, "Multi-class support vector machine," in *Support Vector Machines Applications*, p. 27.
- [27] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multi-class support vector machines," *IEEE Transactions on Neural Networks*, vol. 13, no. 2, pp. 415–425, 2002.
- [28] *LBR iiwa*, KUKA Roboter GmbH, Germany, 2016, Spez LBR iiwa V5 LBR.
- [29] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 1–27, 2011.
- [30] D. C. Montgomery, *Engineering Statistics*, John Wiley & Sons, 3rd edition, 2004.

Research Article

Heterogeneous Gray-Temperature Fusion-Based Deep Learning Architecture for Far Infrared Small Target Detection

Junhwan Ryu  and Sungho Kim 

Department of Electronic Engineering, Yeungnam University, Republic of Korea

Correspondence should be addressed to Sungho Kim; sunghokim@ynu.ac.kr

Received 1 March 2019; Revised 23 April 2019; Accepted 10 July 2019; Published 22 August 2019

Guest Editor: Sidike Paheding

Copyright © 2019 Junhwan Ryu and Sungho Kim. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes the end-to-end detection of a deep network for far infrared small target detection. The problem of detecting small targets has been a subject of research for decades and has been applied mainly in the field of surveillance. Traditional methods focus on filter design for each environment, and several steps are needed to obtain the final detection result. Most of them work well in a given environment but are vulnerable to severe clutter or environmental changes. This paper proposes a novel deep learning-based far infrared small target detection method and a heterogeneous data fusion method to solve the lack of semantic information due to the small target size. Heterogeneous data consists of radiometric temperature data (14-bit) and gray scale data (8-bit), which includes the physical meaning of the target, and compares the effects of the normalization method to fuse heterogeneous data. Experiments were conducted using an infrared small target dataset built directly on the cloud backgrounds. The experimental results showed that there is a significant difference in performance according to the various fusion methods and normalization methods, and the proposed detector showed approximately 20% improvement in average precision (AP) compared to the baseline constant false alarm rate (CFAR) detector.

1. Introduction

The problem of the robust detection of small targets is an important issue in surveillance applications, such as infrared search and track (IRST) and infrared (IR) remote sensing. Information about the objects that can be obtained from the image is extremely limited due to the small target size. In particular, targets located on a long distance have a low signal-to-clutter ratio (SCR) and eventually have an adverse effect on the detection performance. In addition, because of the small target size, it is relatively vulnerable to noise of the surrounding environment, such as sun glint, sensor noise, cloud, etc., making it difficult to detect accurately.

The problem of detecting small targets has been directed mainly at using the most suitable filter among the many filters available or to design a new filter. To solve the problem of a fixed filter, which does not reflect the size change according to the movement of the target, studies have been carried out to consider the scale. Moreover, studies have been conducted on using the classifier together with the conventional machine

learning based method. On the other hand, because of the characteristics of the hand-crafted, the small target is confined to a specific environment and severe noise prevents its detection.

This paper proposes a small target detection method based on deep learning capable of end-to-end training. The network structure and training strategy are inspired by the single shot multibox detector (SSD) [1], and the network structure is transformed to a single-scale because it deals only with small targets. The proposed network learned a small target dataset that was constructed directly with the various background clusters. By learning various backgrounds of the sky, this study solved the problem caused by the uncertain heterogeneous background, which was a problem in previous research. This study also compared the result of the fusion of radiometric temperature data by directly constructing raw infrared data as well as gray-scale data that is generally used as the input of a detector network. In addition, the performance was assessed and compared according to the normalization method in heterogeneous data fusion.

The contribution of this paper is summarized as follows.

- (i) A dataset targeting various backgrounds of the sky was constructed for the detection of far infrared small targets. Unlike other research areas, where open datasets exist, there is no dataset to detect and classify far infrared small targets.
- (ii) The dataset constructed in this paper includes infrared raw data. Unlike previous studies that used only intensity-based gray data (8 bits), raw data (14 bits) can be used together. Temperature information is available by applying a radiometric calibration to the raw data. The use of gray-scale and temperature data with physical meaning together as input to the network allows the use of more information and better detection results through fusion.
- (iii) A deep learning-based network for far infrared small target detection that can train and detect from end-to-end beyond conventional hand-crafted method is proposed. Using the proposed network, this study analyzed the effects of pixel-level fusion of gray-scale and radiometric temperature data and the effects of efficient normalization methods for data fusion.

The remainder of this paper is organized as follows. Section 2 briefly introduces previous studies related to the detection and recognition of small targets. Section 3 outlines the proposed method. Section 4 introduces the experimental results and datasets. Finally, Section 5 reports the conclusions.

2. Related Works

Object detection is an important research area of computer vision. Among them, the detection of small targets is a challenging problem because of the limited information. The research directions to solve this problem can be classified broadly into the traditional machine learning-based methods and deep learning-based methodologies, in which recent studies will be conducted.

One of the traditional methodologies is the filter-based method [2–9]. First, previous studies [2–5] examined the filter itself. For example, Barnett [3] evaluated a promising spatial filter for point target detection in infrared images and used a median subtraction filter. Schmidt [5] examined a modified matched filter (MMF) composed of a product of a nonlinear operator called an inverse Euclidean distance and a least-mean-square (LMS) filter to suppress cloud clutter. Studies on adaptively improved filters have been conducted [6–8]. Yang et al. [7] proposed a Butterworth high-pass filter (HPF) that can adaptively determine the cut-off frequency. Zhao et al. [9] proposed another method using a filter to fuse the results of several filters with different directions. Other methods [10–15] were based on the contrast mechanism of the human vision system (HVS). Qi et al. [10] were inspired by the attention mechanism to produce a color and direction-based Boolean map to fuse, and Chen et al. [11] proposed a method of obtaining a local contrast map using a new local contrast measure that measures the degree of difference between the current location and neighbors. After that, a target is detected

with an adaptive threshold inspired by the contrast. Han et al. [12] increased the detection rate through size-adaptation preprocessing and calculated the saliency map using the improved local contrast measure, unlike the conventional method using only the contrast. Deng et al. [13] improved the contrast mechanism by the weighted local difference measure, and a method that applies a classifier was proposed [14]. Han et al. [15] proposed a multiscale relative local contrast measure to remove the interference region at each pixel.

Another approach was to solve the size variation problem that occurs when the target moves [16–18]. For example, Kim et al. [17] proposed a Tune-Max of the SCR method to consider the problem of scale and clutter rejection inspired by the HVS. In the predetection step, target candidates maximizing Laplacian-scale space images are extracted and in the final-detection step. The scale parameters were adjusted to find target candidates with the largest SCR value. This method has shown good performance, but it consists of complicated steps.

The following methodologies [19–21] deal with methods for making the best use of features. Dash et al. [19] proposed a feature selection method that can use features efficiently in a classifier rather than directly relating to the problem of detecting a small target. Kim [20] analyzed various target features to determine which feature is useful for detecting small targets and proposed a machine learning-based target classification method. Bi et al. [21] used multiple novel features to solve the problem of many false alarms (FAs) that occur when existing methods consistently use single metrics for complex backgrounds. A total of seven features were used and a method to identify the final target through a classifier was proposed.

A range of machine learning-based methodologies can be used for small target detection [22–32]. Gu et al. [23] proposed a method to apply a constant false alarm rate (CFAR) detector to the target region after suppressing the clutter by predicting the background through a kernel-based nonparametric regression method. Qi et al. [29] proposed a directional saliency-based method based on observations that the background clutter has a local direction and treat it as a salient region-detection problem. The existing methods still raise the problem of not separating the background completely. Zhang et al. [30] used an optimization approach to separate the target from the background.

Over the last few decades, research has been conducted in various directions mentioned above and more studies are being conducted based on deep learning. Liu et al. [33] proposed that training a sample using a signal-to-noise ratio (SNR) with an appropriate constant value helps improve the performance over training with a randomly sampled SNR. The targets were generated and synthesized randomly and were not actual targets. Chen et al. [34] used a synthetic aperture radar (SAR) image and treated it as a convolutional neural network- (CNN-) based classification problem not a detector network. Because there is little data, it adopts a fully convolution structure except for a fully connected layer to prevent overfitting. Generative adversarial networks (GAN), which is not a general CNN-based structure, were proposed

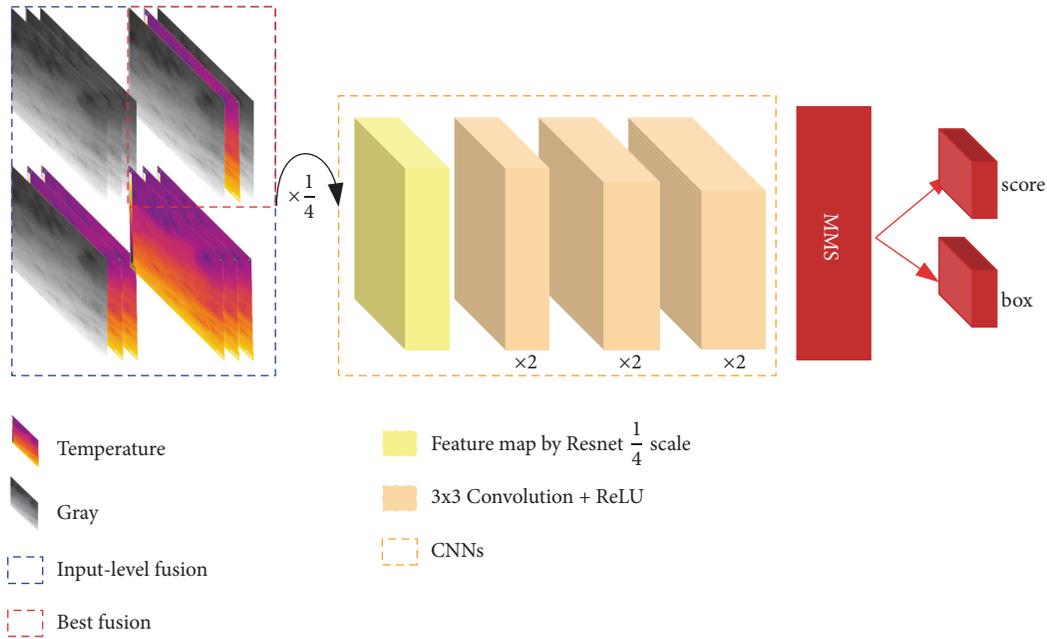


FIGURE 1: Proposed network structure and pixel-level fusion of temperature data.

[35]. The generator trains to transfer the representation of a small object similar to a large object. The discriminator, however, competes with the generator to identify the representation generated by the generator and allows the generator to have a representation that is useful for detection. Hu et al. [36] proposed a way to use the features extracted from other levels of features. Bosquet et al. [37] proposed the problem of a loss of target information as existing detector networks undergo downsampling. After several convolution layers, it assumes that the feature map has sufficient information to determine the area where the target exists and proposes a new concept called the region context network (RCN). In the feature map that passes the shallow convolution, the region with the highest possible likelihood of the target is extracted along with the context to perform a late convolution. The subsequent steps are similar to the general detector network.

Deep learning-based methodologies have been active in many areas in recent years. On the other hand, the problem of detecting small targets has not been actively researched because not only are there no publicly available datasets that can be verified, the information available from the image is limited and it is difficult to produce a situation where a dataset can be constructed.

3. Proposed Method

This section introduces the proposed network structure for the detection and fusion of small targets in the far-infrared region and compares the intensity-based gray-scale data with the radiometric temperature data obtained from the constructed data. This section also introduces the normalization method to fuse heterogeneous data.

Proposed Network Architecture. The proposed network was inspired by the SSD and uses a single-scale feature structure instead of a multiscale feature structure, which is an advantage of SSD because only small targets of up to 20 pixels are handled. The blue dashed line in Figure 1 represents input data and four cases where pixel-level fusion is possible. In addition, the first feature map is a feature map that passes through Resnet-34 [38], the base network. Subsequently, it goes through six convolution layers, and the detection result is obtained by removing redundant detection through the non-maximum suppression (NMS) in the last feature map. In Figure 1, x2 represents two convolutional blocks, so there are six convolutional layers in total. To minimize the loss of information, Resnet-34 was used up to a $1/4$ scale. Bounding box regression and score prediction for obtaining the final detection results have the same structure as the general object detection network but the NMS standard is somewhat relaxed because of the small target size. For training, the learning rate is set to 0.0001 and is a fully convolutional structure consisting only of a 3x3 convolution layer. The optimization method uses Adam optimizer [39], and He initialization [40] is used.

Comparison of Fusion Methods. The blue dotted box in Figure 1 shows the pixel-level fusion method for the fusion of an intensity-based gray-scale and radiometric temperature data. The gray-scale data has one channel and the radiometric temperature data is also made up of one channel, so the heterogeneous data can be concatenated in channel direction. Another common method is to try the feature-level fusion method. Hou et al. [41] used the late-fusion method and it is a method of concatenating feature maps using RGB and gray-scale data as inputs to different networks with the same structure. On the other hand, this paper used the

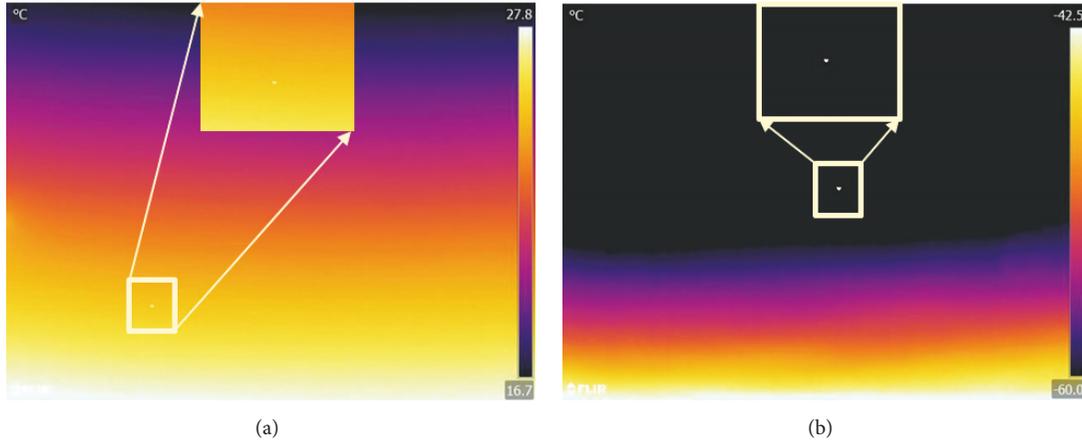


FIGURE 2: Comparison of the seasonally distorted target temperature data. (a) Midsummer in August, and (b) a midwinter in February.

pixel-level fusion method because the feature-level fusion method has not been detected properly. In addition to the pixel-level fusion method, which proposes a range of combinations based on three channels, there is also a method of accumulating three gray scale data, such as RGB and one radiometric temperature data, for a total of four channels. The pretrained deep network cannot be used when this fusion method is applied. Therefore, this paper compares several fusion methods that can fuse heterogeneous data with three channels. Proper normalization methods are required because gray scale data (8-bit) and radiometric temperature data (14-bit) with different ranges of values must be fused together at the pixel-level.

Thermal Normalization. Radiometric temperature data should be normalized. Kim. [42] dealt with temperature data for the problem of detecting pedestrians. At this time, a normalization method was used assuming a maximum temperature of 40°C due to human thermoregulation. On the other hand, the radiometric temperature data was distorted because the experimental environment of this paper dealt only with distant small targets. As a result, even in the same sky, as shown in Figure 2, there is a significant temperature deviation in the air according to the season. The temperature difference between the target and the surrounding air is not large at mid-summer (August, Figure 2(a)), whereas the difference is 20°C or more at mid-winter (February, Figure 2(b)).

At this time, both targets in Figures 2(a) and 2(b) were located in the same sky background and distance with different seasons. Owing to the distorted temperature data, the temperature of the target does not have a constant range. Therefore, the normalization method in the methods reported elsewhere cannot be used and normalized [42], as expressed in (1), to have a value of a specific range. The following were used to compare the results according to the various normalization methods: a normalization method with a specific range of -1 to 0 , -1 to 1 , and 0 to 1 ; a method of normalizing the mean and standard deviation to 0.5 ; and a precalculated mean and standard deviation of large

scale data. $I_n(x, y)$ is the input data and $\min(I)$, $\max(I)$ are the minimum and maximum values, respectively, for the entire input data, and subscripts x, y mean each pixel. The abbreviations, ub and lb , represent the upper and lower bounds of the normalization range, respectively. This makes lb when each pixel of the input data I is the min value, and ub when it is the max value, and the rest has a value between them. For example, if the input data should be normalized between -1 and 1 , set ub to 1 and lb to -1 .

$$I_n(x, y) = \frac{ub - lb}{\max(I(x, y)) - \min(I(x, y))} \times (I(x, y) - \min(I(x, y)) + lb) \quad (1)$$

4. Experimental Results

This section introduces the infrared small target dataset, augmentation method for training, comparison of the results with existing research directions, and various experiments.

4.1. Yeungnam University (YU) FIR Small Targets Dataset

Hardware Specifications. The FLIR T620 model in Figure 3(a) was the thermal imaging camera equipment used to build infrared small target data. FLIR T620 has a spatial resolution of 640×480 and a temperature range between -40°C and 650°C and stores data at 14 bits operating at 30 frames per second (FPS). Figure 3(b) presents small drones that serve as simulated targets and use the DJI's PHANTOM 4 PRO model. The model was 1.38kg, including the battery and propeller, and the size was not provided separately and was approximately $30 \times 30 \times 16$ (cm) when measured directly. The maximum flight time was approximately 30 minutes.

Experimental Environment and Data Acquisition. Experiments were conducted on a specific location, and Figure 3(c) shows the flight record by Google Earth®. The yellow line indicates the accumulated path that the actual target has flown. The target was experimented in various directions and elevation angles at specific locations. When constructing



FIGURE 3: Equipment used in the experiment and experimental environment. (a) is a FLIR T620 infrared camera, (b) is a PHANTOM 4 PRO drone model used as a target, and (c) is the flight trajectory of a target.

data under these circumstances, if all sequences are used, the similarity between the adjacent frames was so large that the frame was divided into 50m frames and frames up to 1km in length. Because the near target can be detected well by the conventional deep learning-based detector, the minimum distance of the target was set to 100m and the maximum distance was set to a maximum of 1km, corresponding to the dot target. The distances used in this paper were the actual distances between the infrared camera and the target. As shown in Figure 3(c), the maximum experiment distance was 1 km and most of the yellow lines (flight trajectory) were performed at distances of less than approximately 500 m. This is because seasons other than winter have smaller targets and less contrast with the surrounding backgrounds, making it impossible to collect data from images.

Dataset Construction. Small infrared target datasets were constructed around 1,000 images. Owing to the problems mentioned above, most of the dataset was composed of less than 500m, mainly from winter and summer. Figure 4 shows the distance of the dataset from 100m to 900m.

Augmentation Dataset. Because it takes considerable time and effort to construct the data, less data can be accumulated unconditionally. Therefore, a method for increasing the number of data is needed. Because the target is small, the methods of changing the image, such as random noise and blur, are difficult to use because the signal of the target is likely to be distorted. The augmentation method used in this paper is a commonly used technique, and random crop augmentation and flip augmentation were applied. An example shown in Figure 5 performed flip augmentation for the original image (a), as shown in (b). (c) and (d) are the results of random crop augmentation for (a) and (b), respectively. The two augmentations were applied together and approximately 7,000 data were used for training.

Label the Ground Truth. When data was extracted from the infrared sequence file from a minimum distance of 100m to a maximum distance of 1km in 50 m increments, the maximum target size corresponded to 20 square pixels, a 1 or 2 pixels minimum. The precise location information of the target must be extracted from the constructed data. Considerable effort is needed compared to the general object label for the following two reasons. First, it is difficult to judge whether there is a target, even if it is close (within 500 m) in the case of a low contrast season or weather due to background cluster, such as clouds. Second, if the target exceeds 500m, the size of the target corresponds to several pixels; hence, it is difficult to confirm the existence of the target. Therefore, sequence data, radiometric temperature data, and intensity-based gray-scale data should be considered together. First, ground truth data is generated based on gray-scale data. If the gray-scale data is difficult to identify with the naked eye, the approximate position of the target is obtained through the sequence, and the accurate position of the target is obtained from the radiometric temperature data.

4.2. Performance Evaluation of the Proposed Methods

Performance Comparison Pixel-Level Fusion and Normalization Method. Figure 6 shows the performance according to the normalization method and pixel-level fusion method. The gray-scale data and the radiometric temperature data showed inferior performance when they were normalized to different ranges. Therefore, radiometric temperature data and gray-scale data were fused at the pixel-level and the same normalization method was then used. As a result, it showed significant performance differences according to the normalization method. In particular, normalization with the mean and standard deviation calculated without normalizing to a specific range showed poor performance. Normalization to a specific range did not result in a significant difference

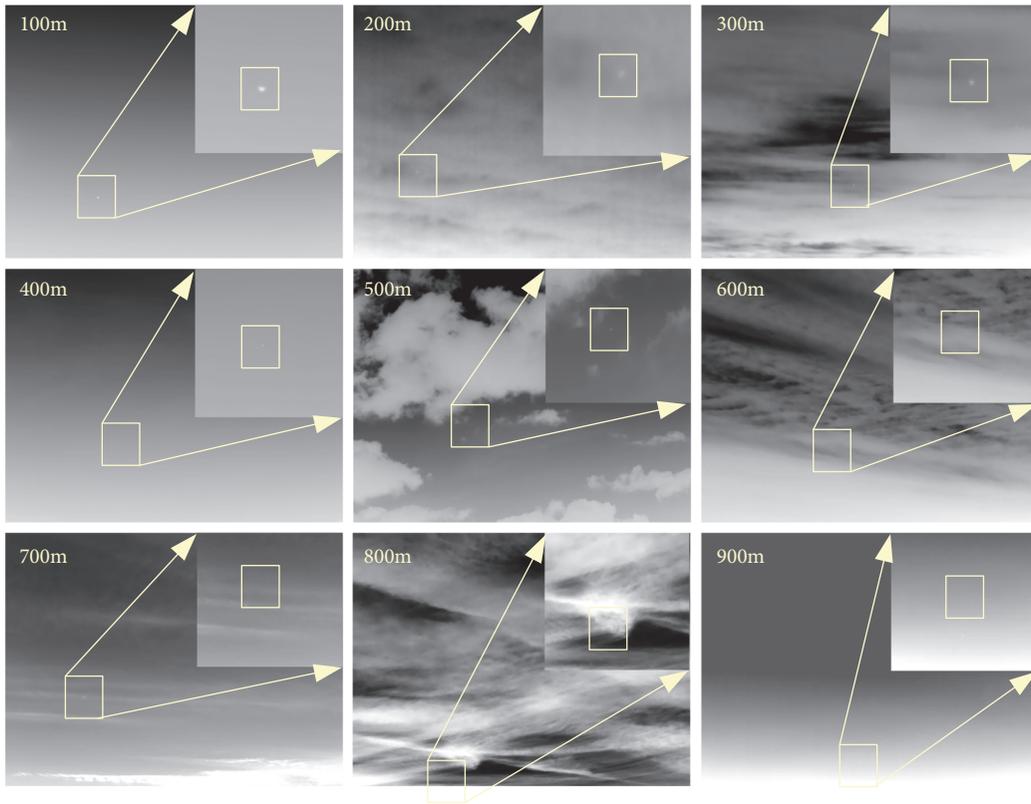


FIGURE 4: Example of infrared small target detection dataset distance from 100m to 900m.

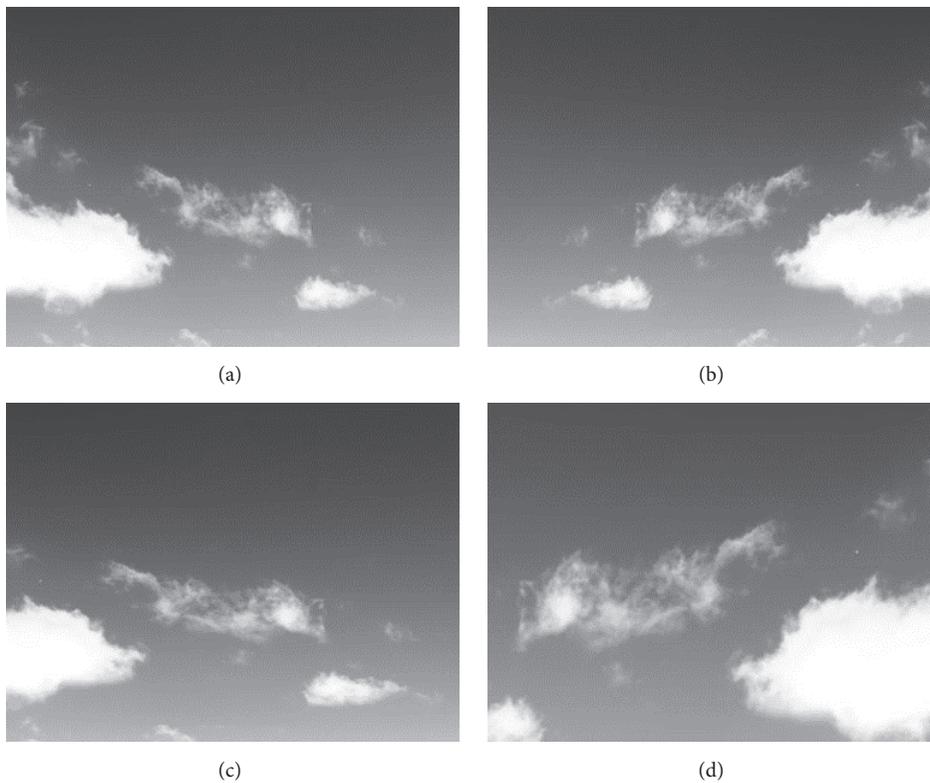


FIGURE 5: Examples of augmentation results. (a) Raw infrared, (b) flip augmented image, (c) image randomly cropped with respect to an original, and (d) random cropped image for a flip augmented image.

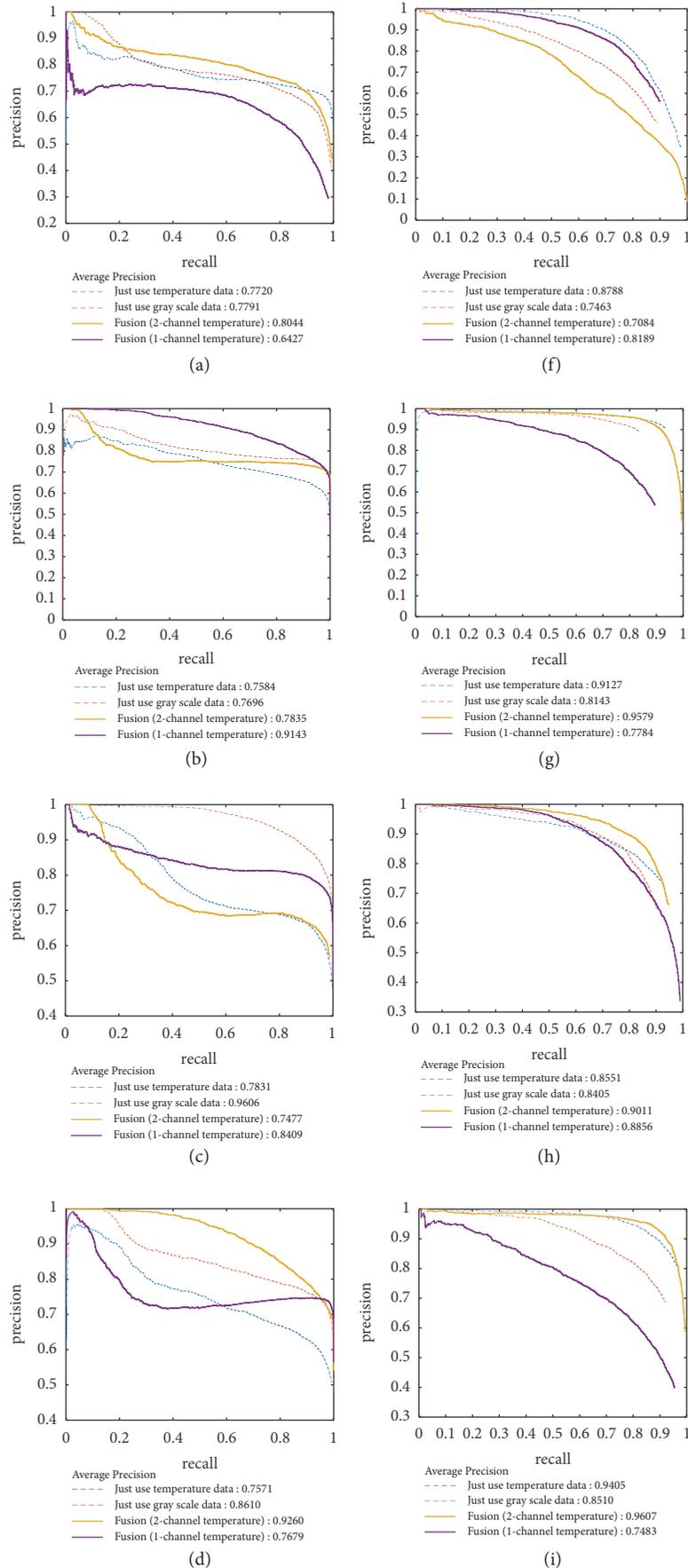


FIGURE 6: Continued.

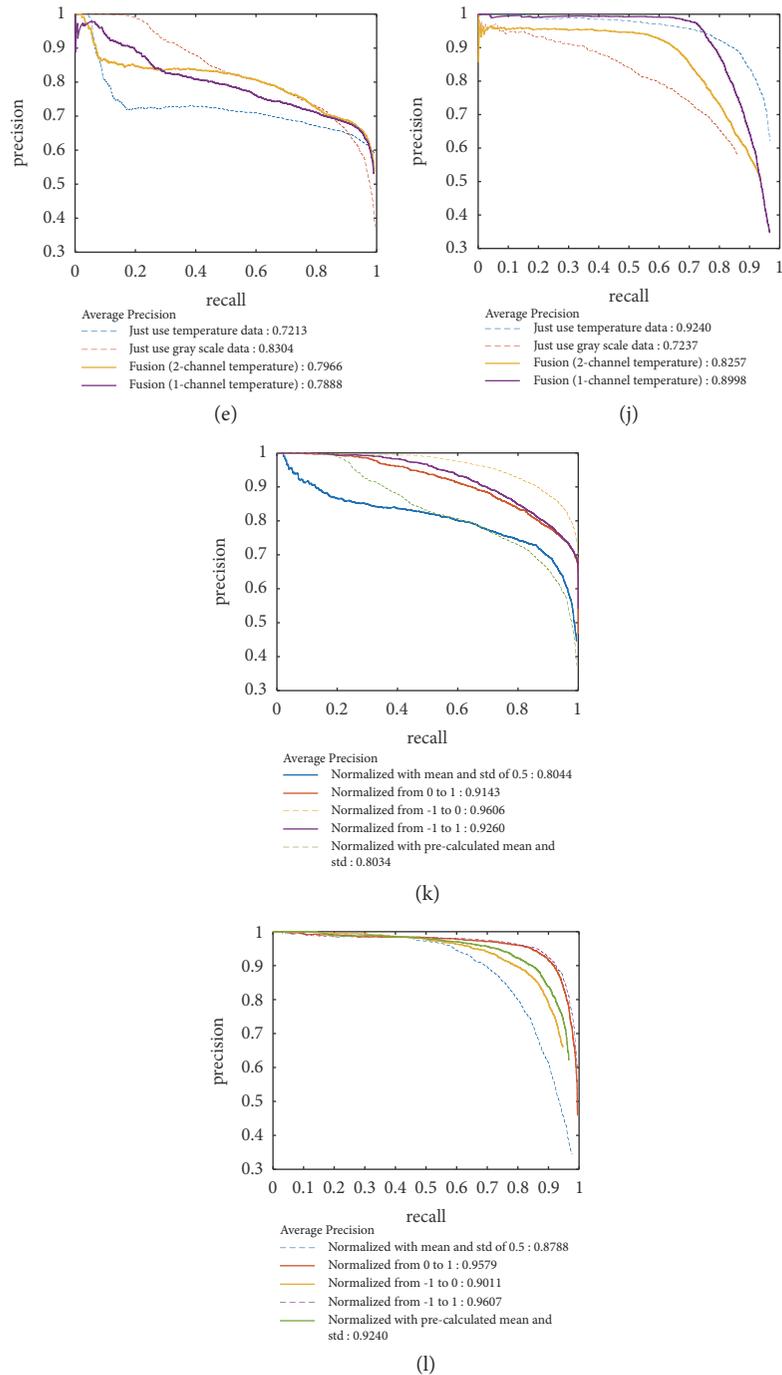


FIGURE 6: Performance comparison between the normalization and pixel-level fusion method. (a)-(e) are the results for the winter test set, and (f)-(j) are the results for the summer test set. The first row was normalized with the mean and standard deviation of 0.5; the second row is between 0 and 1; the third row is between -1 and 0; and the fourth row is between -1 and 1. The last row uses the precomputed mean and standard deviation for large scale dataset. (a)-(j) are the performance results according to the normalization method and fusion method, and (k) and (l) were obtained by collecting only the best performance of each normalization method for the summer and winter test set, respectively.

in performance between normalization methods, but overall, it was helpful to have the minimum of the normalization range to include -1. Figure 6 also shows that robust detection is possible without any significant effect on the seasonal variations.

Experiments in a Network Optimization Perspective. To obtain the optimized results, Table 1 compares the performance according to the network structure, batch normalization, and activation function. Because the ReLU [43] activation function does not use negative data, this study used the Leaky

TABLE 1: Performance comparison based on the network structure, batch normalization and Leaky ReLU.

# of Layers	Batch normalization	Leaky ReLU ($\alpha:0.01$)	Average precision
5	✓		0.7679
	✓	✓	0.7757
7	✓		0.8784
	✓	✓	0.7153
10	✓		0.7365
	✓	✓	0.7987
14	✓		0.6564
	✓	✓	0.6613
			0.6709
			0.6610
			0.6821
			0.6855

TABLE 2: Performance comparison based on the network structure, batch normalization and Leaky ReLU.

Normalization and fusion methods	Batch normalization	Leaky ReLU ($\alpha:0.01$)	Average precision
	✓	✓	+1% ~ +10%

ReLU [44] activation function with a slope factor α of 0.01 and applied batch normalization. In particular, approximately 10% of the Leaky ReLU activation function was improved compared to ReLU. The performance of the table is based on the normalization method with a value between -1 and 1, and the lowest performance fusion method was used to make a clear comparison. As listed in Table 2, the AP was improved by between 1% and 10% for the various normalization and fusion methods mentioned.

Experimental according to Fusion Method and Normalization Method. Figure 7 shows the detection results according to the data fusion method using the fixed normalization method and Figure 8 shows the detection results according to the normalization method using the fixed data fusion method. The fixed normalization method and data fusion method use the method that showed the best performance on average. At this time, the normalization method is a method of normalizing to a value between -1 and 1, and the data fusion method is a method using two sets of radiometric temperature data.

In Figure 7, (a) is the case when only radiometric temperature data was used; (b) is for gray-scale data only; (c) is for radiometric temperature data for one channel, and (d) is for radiometric temperature data for two channels. Based on the normalization method with a value between -1 and 1, a false alarm did not occur in (d) using two radiometric temperature data, which showed the best performance and in (c) based on temperature data fusion. A false alarm occurs in (a) and (b) because it uses only single data rather than fusion-based data. On the other hand, detection was performed correctly in all four cases.

In Figure 8, (a) shows the normalization method using the previously calculated mean and standard deviation for a large scale dataset; (b) normalizes the mean and standard deviation to 0.5; (c) is the normalized value between 0 and

1; (d) is the normalized value between -1 and 0, and (e) is the detection result according to the normalized value between -1 and 1. From the detection results of (a) and (b), which performed normalization based on a specific value, it can be confirmed that although the detection is correct, many false alarms are generated and the performance is poor.

Comparison with Existing Techniques. Figure 9 presents a test result image from a test dataset constructed on different days and was configured to include various background clusters. Figures 9(a), 9(b), and 9(c) show the result based on the CFAR detector, high-boost (HB) method [45], and the detection results of the proposed network using the best fusion method, respectively. The CFAR detector showed 0.7621 AP, which is similar to or less than that of the deep learning-based method. The HB method works well for locating small targets, but there is a problem that the threshold parameters must be changed according to the environment changes. This paper used test datasets that were built by distance, but the maximum distance of the test dataset was only 321m because the test was done only to that distance. Robust detection is possible using the proposed deep learning-based network, even in complex and various environments, where there is a strong clutter-like cloud. In addition, robust detection is possible without being affected by seasonal changes.

4.3. How Can the Radiometric Temperature Data Be Obtained?

The radiometric temperature data can be obtained using the procedure shown in Figure 10. Variable x is the raw input data and is a 14-bit digital count. The FLIR T620 infrared camera, which receives 14-bit digital count input, internally finds for a, b corresponding to the slope and intercept of the calibration curve. This process is called a radiometric calibration. The radiance y can be obtained using a and b of

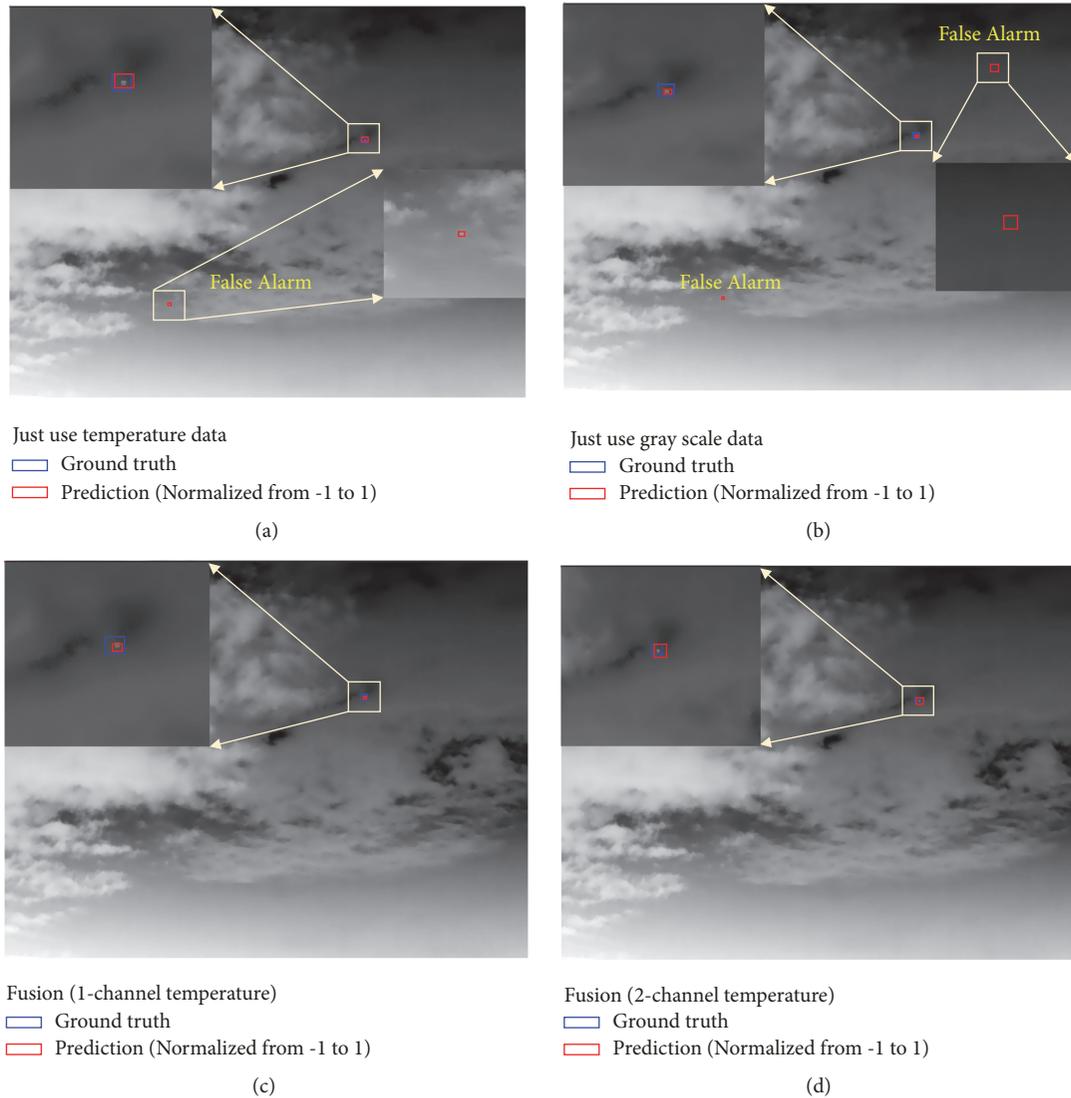


FIGURE 7: Comparison of detection results according to data fusion method. All the same normalization methods were used with values between -1 and 1. The image (a) shows the case where only the radiometric temperature data is used for three channels, (b) shows the case where only the gray scale data is used, (c) shows the case where the temperature data is fused using only one channel, and (d) is an example in which temperature data is fused to two channels.

the calibration curve and the 14-bit digital count input. The radiant energy emitted between T_1 and T_2 , the temperature range over which the FLIR T620 equipment operates, can be obtained by integrating the function and can be expressed in terms of $L(\lambda)$. This shows Planck's law as a function of the wavelength. When the radiance value corresponding to y is obtained through the calibration curve, $L(\lambda)$ can be solved using the equation for L_T to obtain the temperature data for the input data 14-bit digital count.

5. Conclusions

This paper proposed a deep learning-based method for the far-infrared detection of small targets. The proposed method directly constructs datasets containing raw IR data to include

a range of backgrounds. Therefore, this study could utilize radiometric temperature data as well as commonly used gray-scale data and attempted to use this temperature data to solve the problem of a lack of information due to the small target size. Various normalization and fusion methods were examined to efficiently combine gray-scale data with radiometric temperature data. In the case of normalization, the performance was better than that using a specific value or a precomputed value for a large scale dataset rather than using a specific range. The use of data fused at the pixel-level rather than using only single data resulted in better overall performance. The seasonal performance can be detected robustly by seasonal changes. The performance of the proposed detector is similar to or better than that of the conventional detector. A comparison of the detection

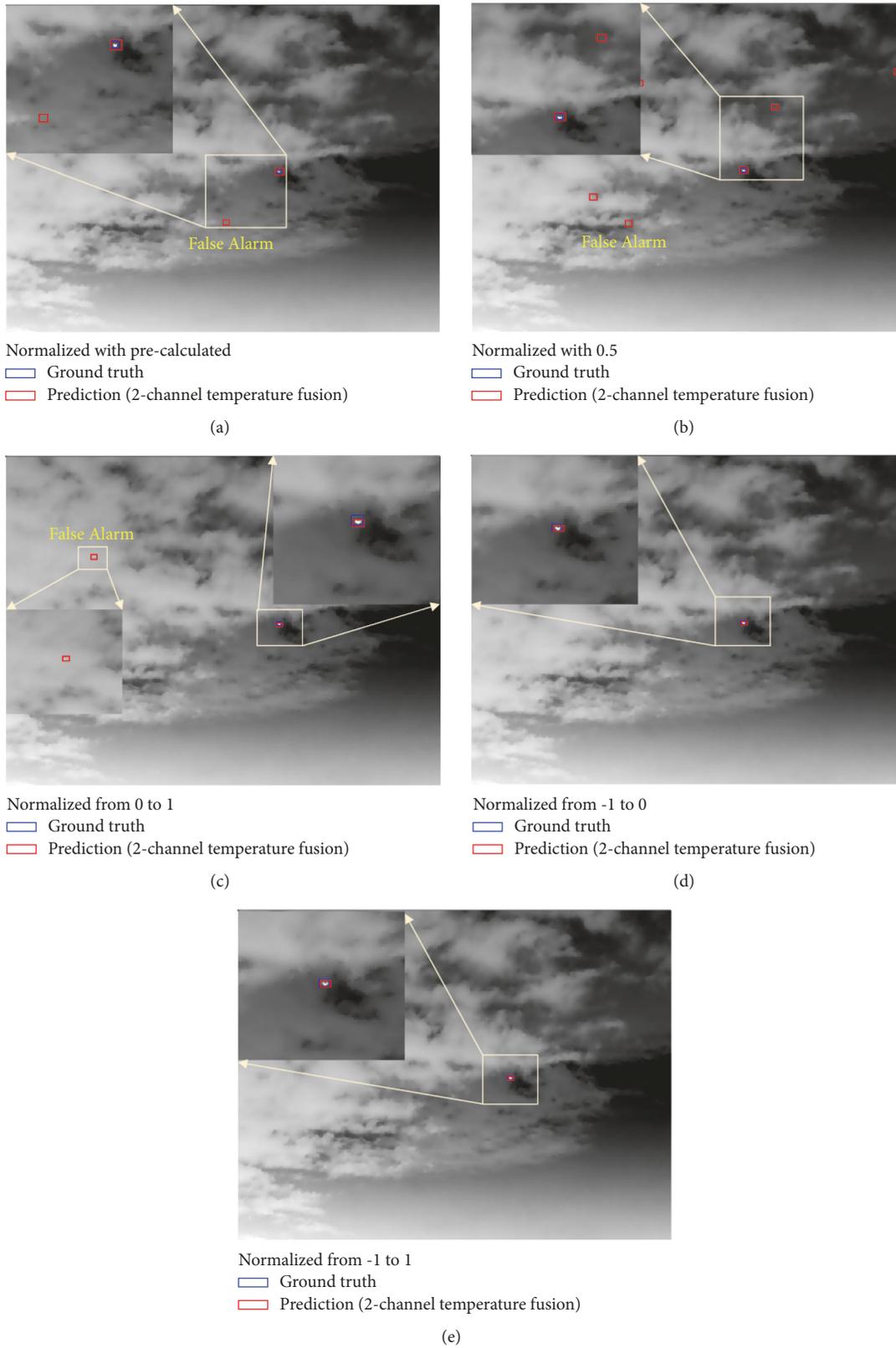


FIGURE 8: Comparison of the detection results according to the normalization method. The same fusion method was used for the two temperature channels from (a)-(e). The image (a) was normalized to a previously calculated mean and standard deviation for a large dataset; (b) was normalized to an arbitrary value of 0.5; and (c) is a value between 0 and 1. In normalization, (d) is normalized to a value between -1 and 0, and (e) is normalized to a value between -1 and 1.

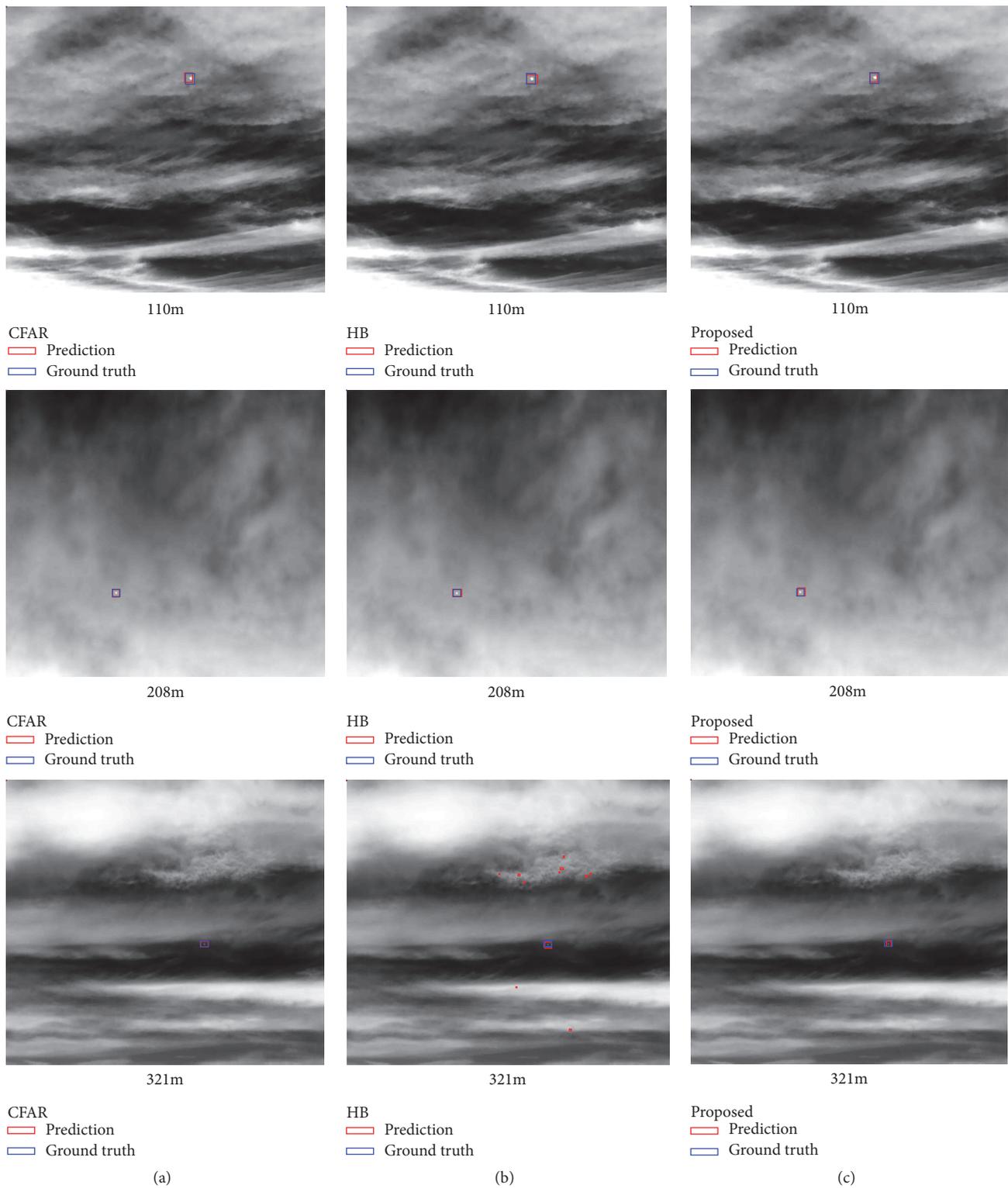


FIGURE 9: Comparison of the results of proposed deep learning based detector, conventional CFAR detector, and HB-based detector. The proposed detector is based on the fusion method using two sets of radiometric temperature data that showed the best performance and normalization method with a value between -1 and 1. In case of HB, the threshold parameter for detection at 208m was applied to 321m as it is.

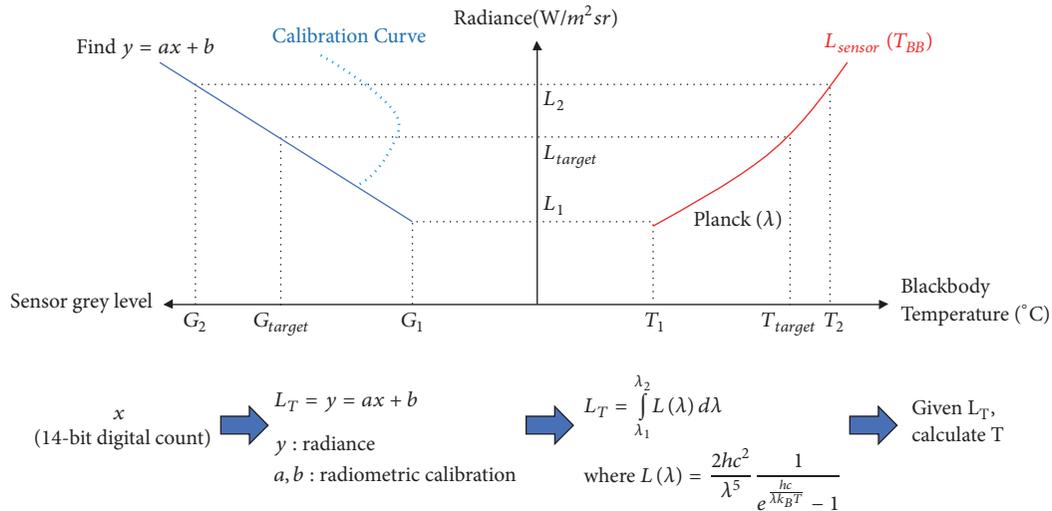


FIGURE 10: Process for obtaining radiometric temperature data. The radiance corresponding to the 14-bit digital count input data was calculated through the radiometric calibration process. Using the Planck equation to find the temperature corresponding to that radiance, the temperature is the target's radiometric temperature data.

results confirmed that the clutter can be detected robustly using the proposed deep learning-based method, even in very complicated and varying environments.

Data Availability

The infrared small target data used to support the findings of this study have not been made available because of security reasons.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgments

This work was supported by 2019 Yeungnam University Research Grants.

Supplementary Materials

The supplementary file compares the detection results of the proposed detector with a constant false alarm rate (CFAR) detector, which corresponds to the baseline method. The first page compares the detection results of the proposed detector with the CFAR detector for the winter season, and the upper left represents the flight record for constructing the test demo dataset. The yellow solid line is the flight record of the actual target. The second page compares the results of the CFAR detector with that of the proposed detector by comparing the detection results for summer. The third page is a total seasonal flight record for building a test demo dataset containing both seasons. (*Supplementary Materials*)

References

- [1] W. Liu, D. Anguelov, D. Erhan et al., "Ssd: Single shot multibox detector," in *Proceedings of the European Conference on Computer Vision*, pp. 21–37, Cham, Switzerland, 2016.
- [2] R. C. Warren, *Detection of Distant Airborne Targets in Cluttered Backgrounds in Infrared Image Sequences [Ph.D. thesis]*, University of South Australia, 2002.
- [3] J. Barnett, "Statistical analysis of median subtraction filtering with application to point target detection in infrared backgrounds," *Proceedings of SPIE - The International Society for Optical Engineering*, vol. 1050, pp. 10–18, 1989.
- [4] V. T. Tom, T. Peli, M. Leung, and J. E. Bondaryk, "Morphology-based algorithm for point target detection in infrared backgrounds," *Proceedings of SPIE*, vol. 1954, pp. 25–32, 1993.
- [5] W. Schmidt, "Modified matched filter for cloud clutter suppression," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 6, pp. 594–600, 1990.
- [6] T. Soni, J. R. Zeidler, and W. H. Ku, "Performance evaluation of 2-D adaptive prediction filters for detection of small objects in image data," *IEEE Transactions on Image Processing*, vol. 2, no. 3, pp. 327–340, 1993.
- [7] L. Yang, J. Yang, and K. Yang, "Adaptive detection for infrared small target under sea-sky complex background," *IEEE Electronics Letters*, vol. 40, no. 17, pp. 1083–1085, 2004.
- [8] B. Zhang, T. Zhang, Z. Cao, and K. Zhang, "Fast new small-target detection algorithm based on a modified partial differential equation in infrared clutter," *Optical Engineering*, vol. 46, no. 10, Article ID 106401, 2007.
- [9] B. Zhao, S. Xiao, H. Lu, and J. Liu, "Point target detection in space-based infrared imaging system based on multi-direction filtering fusion," *Progress in Electromagnetics Research*, vol. 56, pp. 145–156, 2017.
- [10] S. Qi, D. Ming, J. Ma, X. Sun, and J. Tian, "Robust method for infrared small-target detection based on Boolean map visual theory," *Applied Optics*, vol. 53, no. 18, pp. 3929–3940, 2014.

- [11] C. L. P. Chen, H. Li, Y. Wei, T. Xia, and Y. Y. Tang, "A local contrast method for small infrared target detection," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 1, pp. 574–581, 2014.
- [12] J. Han, Y. Ma, B. Zhou, F. Fan, K. Liang, and Y. Fang, "A robust infrared small target detection algorithm based on human visual system," *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 12, pp. 2168–2172, 2014.
- [13] H. Deng, X. Sun, M. Liu, C. Ye, and X. Zhou, "Small infrared target detection based on weighted local difference measure," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 7, pp. 4204–4214, 2016.
- [14] Z. Cui, J. Yang, S. Jiang, and J. Li, "An infrared small target detection algorithm based on high-speed local contrast method," *Infrared Physics & Technology*, vol. 76, pp. 474–481, 2016.
- [15] J. Han, K. Liang, B. Zhou, X. Zhu, J. Zhao, and L. Zhao, "Infrared small target detection utilizing the multiscale relative local contrast measure," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 4, pp. 612–616, 2018.
- [16] D. J. Gregoris, S. K. Yu, S. Tritchew, and L. Sevigny, "Detection of dim targets in FLIR imagery using multiscale transforms," *Proceedings of SPIE*, vol. 2269, pp. 62–72, 1994.
- [17] S. Kim and J. Lee, "Scale invariant small target detection by optimizing signal-to-clutter ratio in heterogeneous background for infrared search and track," *Pattern Recognition*, vol. 45, no. 1, pp. 393–406, 2012.
- [18] H. Fu, Y. Long, R. Zhu, W. An, H. Yu, and J. Dong, "Infrared small target detection based on multiscale center-surround contrast measure," in *Proceedings of the Ninth International Conference on Graphic and Image Processing*, vol. 10615, Qingdao, China, 2018.
- [19] M. Dash and H. Liu, "Feature selection for classification," *Intelligent Data Analysis*, vol. 1, no. 1–4, pp. 131–156, 1997.
- [20] S. Kim, "Analysis of small infrared target features and learning-based false detection removal for infrared search and track," *Pattern Analysis and Applications*, vol. 17, no. 4, pp. 883–900, 2014.
- [21] Y. Bi, X. Bai, T. Jin, and S. Guo, "Multiple feature analysis for infrared small target detection," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 8, pp. 1333–1337, 2017.
- [22] C. Gao, T. Zhang, and Q. Li, "Small infrared target detection using sparse ring representation," *IEEE Aerospace and Electronic Systems Magazine*, vol. 27, no. 3, pp. 21–30, 2012.
- [23] Y. Gu, C. Wang, B. Liu, and Y. Zhang, "A kernel-based nonparametric regression method for clutter removal in infrared small-target detection applications," *IEEE Geoscience and Remote Sensing Letters*, vol. 7, no. 3, pp. 469–473, 2010.
- [24] Y. Li and Y. Zhang, "Robust infrared small target detection using local steering kernel reconstruction," *Pattern Recognition*, vol. 77, pp. 113–125, 2018.
- [25] H. Deng, X. Sun, M. Liu, C. Ye, and X. Zhou, "Infrared small-target detection using multiscale gray difference weighted image entropy," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 1, pp. 60–72, 2016.
- [26] H. Deng, Y. Wei, and M. Tong, "Small target detection based on weighted self-information map," *Infrared Physics & Technology*, vol. 60, pp. 197–206, 2013.
- [27] Z. Wang, J. Tian, J. Liu, and S. Zheng, "Small infrared target fusion detection based on support vector machines in the wavelet domain," *Optical Engineering*, vol. 45, no. 7, Article ID 076401, 2006.
- [28] S. Kim, K.-J. Yoon, and I. S. Kweon, "Object recognition using a generalized robust invariant feature and Gestalt's law of proximity and similarity," *Pattern Recognition*, vol. 41, no. 2, pp. 726–741, 2008.
- [29] S. Qi, J. Ma, C. Tao, C. Yang, and J. Tian, "A robust directional saliency-based method for infrared small-target detection under various complex backgrounds," *IEEE Geoscience and Remote Sensing Letters*, vol. 10, no. 3, pp. 495–499, 2013.
- [30] L. Zhang, L. Peng, T. Zhang, S. Cao, and Z. Peng, "Infrared small target detection via non-convex rank approximation minimization joint l_2, l_1 norm," *Remote Sensing*, vol. 10, no. 11, 2018.
- [31] Z. Chen, G. Wang, J. Liu, and C. Liu, "Infrared small target detection algorithm based on self-adaptive background forecast," *International Journal of Infrared and Millimeter Waves*, vol. 27, no. 12, pp. 1619–1624, 2007.
- [32] X. Fan, Z. Xu, J. Zhang, Y. Huang, and Z. Peng, "Infrared dim and small targets detection method based on local energy center of sequential image," *Mathematical Problems in Engineering*, vol. 2017, Article ID 4572147, 16 pages, 2017.
- [33] M. Liu, H. Y. Du, Y. J. Zhao, L. Q. Dong, and M. Hui, "Image small target detection based on deep learning with snr controlled sample generation," in *Current Trends in Computer Science and Mechanical Automation*, vol. 1, pp. 211–220, Sciendo Migration, 2017.
- [34] S. Chen, H. Wang, F. Xu, and Y.-Q. Jin, "Target classification using the deep convolutional networks for SAR images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 8, pp. 4806–4817, 2016.
- [35] J. Li, X. Liang, Y. Wei, T. Xu, J. Feng, and S. Yan, "Perceptual generative adversarial networks for small object detection," in *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, pp. 1951–1959, USA, July 2017.
- [36] G. X. Hu, Z. Yang, L. Hu, L. Huang, and J. M. Han, "Small object detection with multiscale features," *International Journal of Digital Multimedia Broadcasting*, vol. 2018, 2018.
- [37] B. Bosquet, M. Mucientes, and V. M. Brea, "STDnet: A ConvNet for small target detection," in *Proceedings of the British Machine Vision Conference*, 2018.
- [38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 770–778, July 2016.
- [39] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proceedings of the International Conference on Learning Representations*, 2015.
- [40] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: surpassing human-level performance on imagenet classification," in *Proceedings of the 15th IEEE International Conference on Computer Vision (ICCV '15)*, pp. 1026–1034, IEEE, Santiago, Chile, December 2015.
- [41] Y. L. Hou, Y. Song, X. Hao, Y. Shen, and M. Qian, "Multispectral pedestrian detection based on deep convolutional neural networks," *Infrared Physics & Technology*, vol. 94, pp. 69–77, 2018.
- [42] T. Kim and S. Kim, "Pedestrian detection at night time in FIR domain: Comprehensive study about temperature and brightness and new benchmark," *Pattern Recognition*, vol. 79, pp. 44–54, 2018.
- [43] V. Nair and G. E. Hinton, "Rectified linear units improve Restricted Boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML '10)*, pp. 807–814, Haifa, Israel, June 2010.

- [44] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proceedings of the 30th International Conference on Machine Learning*, vol. 30, 2013.
- [45] Y. Shi, Y. Wei, H. Yao, D. Pan, and G. Xiao, "High-boost-based multiscale local contrast measure for infrared small target detection," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 1, pp. 33–37, 2018.

Research Article

Vision-Based Deep Q-Learning Network Models to Predict Particulate Matter Concentration Levels Using Temporal Digital Image Data

SungHwan Kim ¹, SeHee Jung ², SeongMin Yang ¹, JiSeong Han,¹
ByungYong Lee,³ JaeHwa Lee ^{4,5} and Sung Won Han ⁶

¹Department of Applied Statistics, Konkuk University, Seoul, Republic of Korea

²AI Analytics Team, Deep Visions, Seoul, Republic of Korea

³Marketing Team, INSUNGIT, Seoul, Republic of Korea

⁴Earth System Science Interdisciplinary Center, University of Maryland, College Park, MD, USA

⁵NASA Goddard Space Flight Center, Greenbelt, MD, USA

⁶School of Industrial Management Engineering, Korea University, Seoul, Republic of Korea

Correspondence should be addressed to Sung Won Han; swan@korea.ac.kr

Received 2 April 2019; Revised 21 May 2019; Accepted 27 May 2019; Published 26 June 2019

Guest Editor: Sidike Paheding

Copyright © 2019 SungHwan Kim et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Particulate matter (PM) has been revealed to have detrimental effects on public health, social economy, agriculture, and so forth. Thus, it became one of the major concerns in terms of a factor that can reduce “quality of life” over East Asia, where the concentration is significantly high. In this regard, it is imperative to develop affordable and efficient prediction models to monitor real-time changes in PM concentration levels using digital images, which are readily available for many individuals (e.g., via mobile phone). Previous studies (i.e., DeepHaze) were limited in scope to priorly collected data and thereby less practical in providing real-time information (i.e., undermined interprediction). This drawback led us to hardly capture drastic changes caused by weather or regions of interests. To address this challenge, we propose a new method called Deep Q-haze, whose inference scheme is built on an online learning-based method in collaboration with reinforcement learning and deep learning (i.e., Deep Q-learning), making it possible to improve testing accuracy and model flexibility in virtue of real-time basis inference. Taking into account various experiment scenarios, the proposed method learns a binary decision rule on the basis of video sequences to predict, in real time, whether the level of PM₁₀ (particles smaller than 10 in aerodynamic diameter) concentration is harmful ($>80\mu\text{g}/\text{m}^3$) or not. The proposed model shows superior accuracy compared to existing algorithms. Deep Q-haze effectively accounts for unexpected environmental changes in essence (e.g., weather) and facilitates monitoring of real-time PM₁₀ concentration levels, showing implications for better understanding of characteristics of airborne particles.

1. Introduction

Particulate is a minute particle that is in liquid or solid phase in the atmosphere and often refers to a particulate material having an aerodynamic diameter of $10\mu\text{g}/\text{m}^3$ or less (PM₁₀). This originates from anthropogenic sources, such as combustion of fossil fuels such as coal, oil, the exhaust gas of manufacturing factories, and automobile engines as well as natural sources, such as desert and ocean (mineral dust and sea salt). Particulates are also known to affect climate

and precipitation as well as human health [1, 2]. Moreover, confronting threats of PM to Asian countries becomes no longer negligible to the point that the media and research groups consistently reveal detrimental effects [3]. To our surprise, it is notable that the World Cancer Institute in October 2013 analyzed a large-scale cohort of 2,095 lung cancer patients out of 312,944 people in the nine European countries [4]. Evidences that PM was determined as primary carcinogens were due to the fact that the risk of lung cancer increased by 22% at an PM increment of $10\mu\text{g}/\text{m}^3$.

Of late days, air pollution remains intractable to be resolved in massively populated regions like Seoul in South Korea, where the presence of fine dust is easily detected in vision. It is reported that PM10 concentration in South Korea is measured twice as high as that of OECD countries on average [5]. This record is even higher than the major cities such as New York City and Paris. To circumvent air pollution, the government has put significant efforts for better forecasting and developing benchmarks. And yet, we still encounter many challenges, for instance, inaccurate reporting system particularly at a specific location because of the limited metering sites, costly gadgets, and so forth. No wonder the most complete way to resolve fine dust is to eliminate the sources. However, this strategy obviously takes demanding costs and time-consuming tasks. Under this circumstance, concerns to public health have increased at an unprecedented rate. Civilians believe that hourly reporting of PM levels might not be sufficient for real-time air quality [6]. Thus, it is necessary to suggest a method applicable to prompt measurements of PM concentrations without expensive devices and spacious place to install. This is the point where our research motivation comes in.

The predictive models of PM concentration are proposed in various ways. A majority of methods adopted an explorative way: (1) elementary statistic [7], (2) time-series visualization [8], (3) histogram on a yearly basis [9], and (4) image data [10]. Another choice is to use predictive models such as logistic regression, support vector machine (SVM), and deep neural network (DNN) [11]. To construct training data set, a majority of previous methods typically utilized regional, climatic, or daily publicly available weather data (e.g., humidity, insolation, etc.), whereas the image data-based method makes an exclusive use of RGB data (Red, Green, and Blue) calibrated on true PM levels.

To the best of our knowledge, the attention to artificial intelligence revives through the diverse fields due to the rethinking of reinforcement learning. AlphaGo broke down at the 9th stage against Lee Sedol. The level of artificial intelligence is much better than expected. AlphaGo is based on Google's deep Q algorithm [12]. It is an artificial intelligence algorithm system exploiting reinforcement learning. Originally reinforcement learning is inspired by behavioral psychology, in which an agent defined in an environment recognizes the current state and selects a behavior or sequence of actions that maximizes compensation among the selectable behaviors. These problems are so comprehensive that they are also studied in areas of game theory, control theory, operational science, information theory, simulation-based optimization, multiagent systems, flock intelligence, statistics, and genetic algorithms [13, 14].

The deep Q-network algorithm (a.k.a DQN) learns the optimal policy by learning the Q function predicting the expected value of the utility that would result from performing a given action in a given state. After learning the Q function, we can derive the optimal policy by performing the action that gives the highest Q in each state. The goal of the agent (decision maker) is to maximize the sum of the rewards. The choice is the action of getting the greatest reward in that state in the long run. The DQN predicts the Q-value

using the action-value function CNN (convolutional neural networks), one of the neural network-type decision rules. It is well known that the convolutional neural network (CNN) is an efficient image processing algorithm adapted for vision analysis and image recognition.

In this paper, we proposed a predictive model that builds on the deep Q-network algorithm in spirit of reinforcement learning in order to predict particulate levels. We call this algorithm Deep Q-haze. Inspired by conventional reinforcement learning, this predictive model assigns the state an image to evoke multiclass actions on the basis of the prespecified calibration of particulates (e.g., $80\mu\text{g}/\text{m}^3$ less or more). Subsequent to this, the reward and action to get the best reward are determined. Taken together, the proposed Deep Q-haze serves as an effective tool to predict particulate levels solely subject to image data. We hypothesize that superior predictive performance of Deep Q-haze leads to less chance of false detection compared to previous classification model (e.g., SVM, RF, and DeepHaze) and consequently improves practical utility.

2. Datasets and Related Work

2.1. Datasets. Below we describe particulate data that a predictive model learns on. For the most part, we collect the video sequence data in the major cities of South Korea (e.g., Seoul and Daegu), where the cities are featured with a large-scale industrial complex, automobiles, highly populated counties. In such mega cities, gas emission has been a years-long environmental challenge, and high-concentration dusts occupy the peninsula throughout the year. More importantly, it is asserted that air pollution is primarily attributed to contaminants of eastern and southern China [15] and thus this problem, at present, remains out of control. When it comes to data collection, we gauge particulate levels via a high-performance device (Aerosol Mass Monitor (AEROCET-831) manufactured by Met One Instruments; <http://metone.com/>), whose perceptible dust size ranges from PM2.5 to PM10. In this paper, we purposely focus on the level of PM10. Regarding nonfixed image sequences (i.e., manually taken via mobile phone), we retrieved image data in our recent research, where we take into account residential areas, a group of trees, and building complexes featured with only nonatmospheric information (i.e., absence of sky). The interested regions largely include diverse categories: (1) outdoor parking spots, (2) building complex on campus, (3) indoor office environment, (4) street regions by exhausts emission, (5) vicinity of construction sites, and (6) residential areas. On average, video sequences are recorded with 5~25 frames per second for a total of 268 sequences. To take a glance, thumbnails of each video sequence are presented in Table 3. The video sequences are taken by Samsung phone cameras (S7) and its built-in IP webcam. Data and programming codes are available online (<https://sites.google.com/site/sunghwanshome/>).

2.2. Deep Q-Network Algorithm. Briefly, the deep reinforcement learning (Deep RL) system combines reinforcement

TABLE 1: The proposed architecture of simplified CNN models.

type	patch size/stride	input size
conv	4×4/1	200×200×9
conv	2×2/1	200×200×10
flattening	-	1×1×400000
linear	-	1×1×100
softmax	classifier	1×1×2

learning and neural networks. As aforementioned, reinforcement learning relates to an area of machine learning, in which an agent defined in an environment recognizes the current state and selects a behavior or sequence of actions that maximizes the expectation of the sum of the rewards among the selectable behaviors as below:

$$R_t = r_t + \gamma (r_{t-1}) + \gamma^2 (r_{t-2}) + \dots + \gamma^T (r_{t-T}) \quad (1)$$

The objective of the agent is to find a strategy (a.k.a. policy) so as to maximize the expected sum of discounted rewards. In theory, the optimal policy is defined as the expectation of rewards that potentially earn in the future when continuing the actions along the policy π at the current state $s_t = s$.

$$\pi^*(s) = \arg \max_a E(R_t | s_t = s, a_t = a) \quad (2)$$

The action a (i.e., $\pi^*(s)$) is selected such that the expectation of the sum of the rewards is maximized. Instead of above, we learn $Q(s, a)$ and thereby find the optimal action $a_t (= a)$ in state $s_t (= s)$.

$$Q^{\text{new}}(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \left[r_{t+1} + \gamma \cdot \max_a Q(s_{t+1}, a) - Q(s_t, a_t) \right] \quad (3)$$

where $\gamma \in [0, 1)$ is the discount factor. This is the Q-learning method proposed by Watkins [15]. Stepping up beyond Q-learning, [12] proposed the Deep Q-network algorithm (a.k.a. DQN) that learns the optimal policy by the Q function on the basis of the deep convolution neural network (CNN) and approximates the action-state function. In this paper, we use the customized CNN (see Table 1) to detect the characteristics of the image and to determine the behavior of the agent.

$$\min_{\theta} \sum_{t=0}^T \left[\widehat{Q}(s_t, a_t | \theta) - \left(r_t + \gamma \cdot \max_a \widehat{Q}(s_{t+1}, a | \theta) \right) \right]^2, \quad (4)$$

where θ is the set of model parameters and \widehat{Q} is an estimated Q-function.

The learning process optimizes the cost function updating the weight to minimize the above equation. Importantly, two techniques designed to enhance predictive power get involved in the learning process. The first stage is called the capture and replay method. To put this plainly, this performs repetitive tasks between storing and taking data at random.

Due to the fact that sequential samples are likely to be strongly correlated, randomness of replay memory attenuates correlation and reduces the variance of updates. In the second stage, the networks learn on a target network and main network one after the other (i.e., constructing two networks). Meanwhile, the target network is fixed and only the main network is updated. The target network updates the values of the main network once every predetermined step. This trick tackles the problem of moving targets and continuously updates the Q-function to maximize the expectation of rewards in the future. All things taken together, the optimal behavior is determined by the updated main Q-function.

3. Methods

3.1. Augmented Temporal Image Features. In context of big data analytics, it is interesting to boost power of our predictive model. To this end, the proposed model combines multiple feature channels, each containing RGB, HSV, and its haze-related features (i.e., dark channel, color attenuation, and hue disparity; [16–18], (Fattal et al., 2008, and Koschmieder et al., 1925)), for a total of 9 channels. Needless to say, it is generally true that the larger data set we apply, the more potential signals the model may decipher. To take a glance, Figure 1 illustrates how we form augmented image data, which serve as a building block to measure the amount of dusts. Saturation index in HSV ranging from 0 to 255 represents the degree of saturation, which are closely linked to noises attributed to particulates. Combining all channels above, the state s_t in the Q-function at time t takes a $200 \times 200 \times 9$ multidimensional array. To account for particulate levels, we create difference values of two consecutive arrays followed by standardization and filtering outliers exceeding 90th quantile. These arrays of difference in image sequences play a role as building blocks of our predictive model (see Figure 2).

3.2. Resampling-Based Reinforcement Learning. Here, we propose the resampling-based reinforcement learning algorithm. Typically, environmental data are prone to being sequential, time-dependent, and seasonal. These characters naturally invite reinforcement learning-type models to come into play. In one sense, an atmospheric model is suited to reinforcement learning as consecutive variability relates to atmosphere. To the contrary, it is also found that reinforcement learning is hardly exploited to natural environment data, in the sense that repetitive tasks to mimic natural environment are challenging to be implemented, as compared to training for robot arms or video games to which reinforcement learning widely applies. However unlikely it may seem, we can create an artificial environment with regard to particulates such that we arbitrarily maneuver weather conditions in purpose (e.g., dust quantity) via bootstrap sampling. In doing so, we initially build an integrated data pool consisting of real image sequences in a proportion to balanced class labels (e.g., safe and harmful) to stably perform bootstrap sampling (e.g., with replacement). Importantly, such a sampling process allows consecutive learning tasks to construct a vast number of predictive models whose training

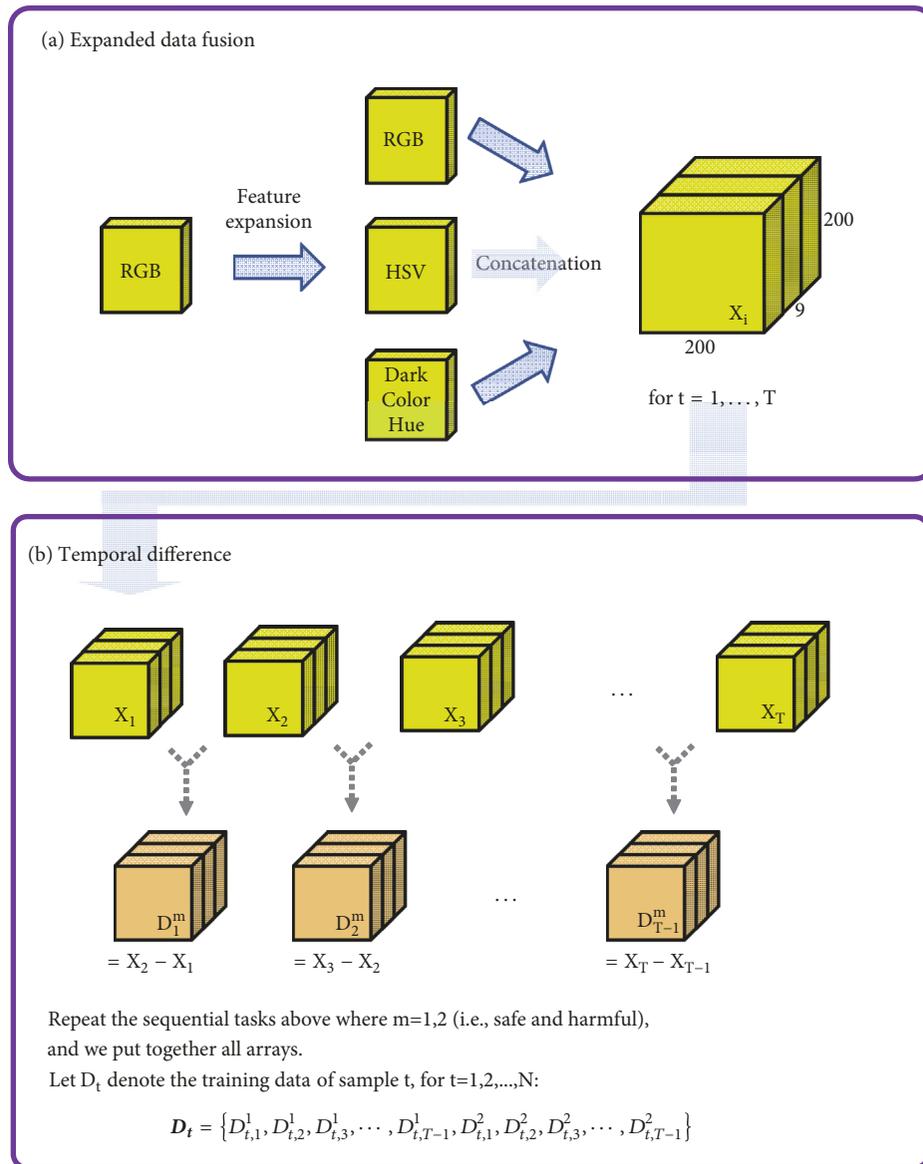


FIGURE 1: A workflow to implement Deep Q-haze and augmented image-related arrays constituting temporal differences between two consecutive arrays.

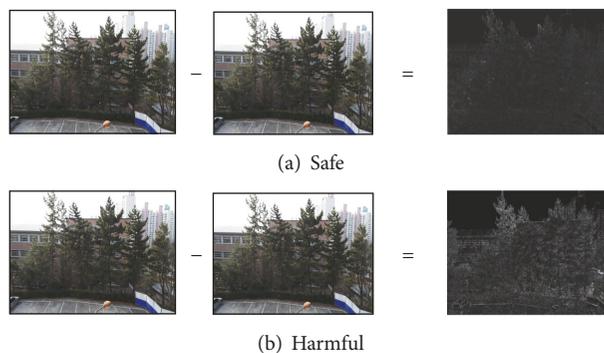


FIGURE 2: Shown are temporal differences between two consecutive arrays with regard to two scenarios (e.g., safe and harmful), respectively. The visual distinction of two difference arrays looks obvious in the sense that the harmful case generates explicit discordance as compared to the safe case.

TABLE 2: Deep Q-haze algorithm.

(1) Initialize model configuration

- (i) Initialize action-value function Q with random weights
- (ii) Construct N sequence arrays (i.e., s_t at time t) of nine channels and randomly sample bootstrap batch out of the integrated data pool (i.e., $N = 20$ as default)
- (iii) Initialize sequence s_t and preprocessed sequenced (i.e., standardization and filtering outliers exceeding 90th quantile) via ϕ , namely $\phi(s_t)$.

(2) Create difference values of two consecutive arrays

$$\phi_t = \phi(s_t) - \phi(s_{t+1})$$

(3) Repeat the following for $t = 1, \dots, T$:

- (i) To derive the optimized action, select a random action a_t , where $a_t \in \{0, 1\}$ (i.e., safe or harmful) with probability ϵ
- (ii) Otherwise select $a_t = \max_a Q(\phi_t)$
- (iii) Execute action a_t in the predictive rule and observe reward r_t and new incoming sequence s_{t+1}
- (iv) Set a_t, s_{t+1}, s_{t+2} and process $\phi_{t+1} = \phi(s_{t+1}) - \phi(s_{t+2})$ and calculate rewards determining actions and impose the weight according to testing outcome (i.e., true or false) and update Q every 10 times.
- (v) For $j = 1, \dots, N - 1$, set as follows

$$y_j = \begin{cases} r_j & \text{if terminal } \phi_{j+1} \\ r_j + \gamma \cdot \max_a Q(\phi_{j+1}, a; \theta) & \text{otherwise} \end{cases}$$

where

$$r_j = \begin{cases} v, & \text{if } z_j = \arg \max_{a_j} Q(\phi_j, a_j) \\ -v, & \text{otherwise} \end{cases}$$

and $z_j \in \{0, 1\}$, a true class label monitored via a device and set $v = 1$ in this paper.

- (vi) Perform a gradient descent step on $(y_j - Q(\phi_j, a_j; \theta))^2$.

data determine rewards, policy, and actions. Particulate levels and image data are monitored over the years, and models all in one, aiming at exclusion of possible seasonal and climate effects. In what follows, Table 2 encapsulates the major implementation schemes one at a step, in short, each including the kernels of deep Q-network [12] and vision-based DeepHaze [11] learning on differences vision of neighboring sequences. In our simulation, we, for simplicity, make in rewards to be small, equivalently adjusting future rewards to be quite negligible. With regard to the Q-function, we adopt the CNN architectures of the predictive model as presented in Table 1, and the CNN architectures are implemented by TensorFlow 1.10 in Python.

4. Results and Discussion

In this experiment, we evaluate the variants of the Deep Q-haze models learning on a range of frame numbers and compare them to other popularly used classifiers (e.g., DeepHaze, random forest, and SVM). With varying parameters, diverse experiment scenarios are considered to mimic real environments and to fortify universal applicability of the model. Tables 4 and 5 encapsulate the predictive performance of the Deep Q-haze and its competitor classifiers. It is evident to say that the proposed algorithm, when using all datasets,

outstandingly distinguishes a harmful atmospheric condition with high accuracy and low false detection (i.e., Youden index = sensitivity + specificity - 1; e.g., 0.9817 - 0.9894 for the indoor office).

4.1. Indoor Environment (an Office and an Experimental Chamber). It is certain that clean air quality in an indoor office is a critical part to maintain health. It is sensible, with that in mind, to purposely focus on image sequences in office at Konkuk University over the several months. We collect 2,200 video sequences (i.e., 1,100 clips of each class label), each containing at least 20 image frames per 1 min. Generally robustness of the algorithm is essential for practical utility. In what follows, we performed large-scale experiments under controlled conditions to verify if the Deep Q-haze is robust against various environmental factors. The experiments were carried out largely under four conditions: presence of windiness, high temperature, high humidity, and high light intensity. To this end, we construct the experiment chamber (i.e., large container) specially designed to create artificial circumstances (see Table 3 at the bottom). Beside factors of interest, other conditions remained the same at ordinary level. Table 6 shows that Deep Q-haze consistently maintains high predictive power regardless of environmental conditions (e.g., windiness, high temperature, etc.). It is found

TABLE 3: Thumbnails of target regions where particulate matters are analyzed to construct the predictive model.

Category	Thumbnails	# of video sequences
Indoor office		2,200
Outdoor parking lot of Konkuk Univ.		3,200
Outdoor parking lot of Keimyung Univ.		3,000
Mobile video clips		1,000
Experimental chamber		500

that Deep Q-haze is less likely to be deteriorated, even though varying environmental factors can promote the randomness of particulates. We collect 500 sequences of only harmful labels, consisting of at least 20 images per 1 min.

4.2. Outdoor Regions. Unsurprisingly, outdoor regions have a tendency to higher levels of particulate than indoor and, due to open space, facilitate visually gauging dust particles present in the air through a long distance. Considering that campus regions are filled up with automobiles, where population flows are relatively intense, we chose two regions: (1) a parking lot (Keimyung University) and (2) building complex (Konkuk University), where we install high-resolution cameras and dust measurement device (AEROCET-831). For several months (2017~2018), we monitored outdoor parking lots all day long and recoded image sequences. We collect 3,000 (an outdoor parking lot of Keimyung University) and 3,200 (an outdoor parking lot of Konkuk University) video

sequences of both safe and harmful labels, consisting of at least 20 image frames per 1 min. We focus on image captured from fixed camera and mobile phone camera in a different way due to perturbation that occurs when a mobile phone is manually controlled. To take a glimpse, refer to the thumbnail image in Table 3.

4.2.1. Image Sequences of Fixed Camera. Tables 4 and 5 show that Deep Q-haze outperforms DeepHaze, SVM, and RFs. Note that Deep Q-haze performs with high accuracy (0.9839 ~ 0.9914 of Keimyung Univ., 0.9040 ~ 0.9220 of Konkuk Univ.; hereafter this order is kept the same) as opposed to DeepHaze (0.6300 ~ 0.6336, 0.4560 ~ 0.4580), random forest (0.4100 ~ 0.4581, 0.4380 ~ 0.4690), and SVM (0.3800 ~ 0.4236, 0.6240 ~ 0.6560). It is interesting to see that predictive power tends to be increasing as the frames augmented from 5 to 20. Besides, Deep Q-haze suffers less from the false detection (i.e., high Youden index; Deep Q-haze: 0.9658

TABLE 5: Prediction accuracy of predictive models with an application to various video sequences.

# of frames	5 frames	10 frames	15 frames	20 frames
Konkuk Univ. Indoor				
Deep Q-Haze	0.9918	0.9945	0.9927	0.9817
Deep Haze	0.7112	0.7487	0.7450	0.7487
RF	0.6262	0.6300	0.5000	0.5000
SVM	0.5000	0.5000	0.5000	0.5000
Konkuk Univ. Outdoor				
Deep Q-Haze	0.9040	0.9100	0.9160	0.9220
Deep Haze	0.4560	0.4580	0.4520	0.4580
RF	0.4690	0.5060	0.4380	0.4610
SVM	0.6560	0.6270	0.6240	0.6440
Keimyung Univ.				
Deep Q-Haze	0.9839	0.9861	0.9877	0.9914
Deep Haze	0.6300	0.6336	0.6309	0.6336
RF	0.4100	0.4390	0.4418	0.4581
SVM	0.3800	0.4054	0.4318	0.4236
Mobile Phone				
Deep Q-Haze	0.8842	0.8796	0.8703	0.8657
Deep Haze	0.5046	0.5000	0.5023	0.5000
RF	0.4768	0.5046	0.4513	0.4791
SVM	0.3287	0.3217	0.3148	0.3148

TABLE 6: Simulation study: sensitivity (Sen), specificity (Spe), Youden index, and standard errors in parentheses with an application to various video sequences and environmental conditions. The simulation was repeated 50 times.

# of frames	5 frames	10 frames	15 frames	20 frames
Windiness (Use of fan)				
Sen	0.8411 (0.0408)	0.8495 (0.0452)	0.9152 (0.0335)	0.9116 (0.0304)
Spe	0.8796 (0.0299)	0.9141 (0.0266)	0.9411 (0.0221)	0.9419 (0.0200)
Youden	0.7207 (0.0377)	0.7636 (0.0526)	0.8563 (0.0738)	0.8535 (0.0496)
High Temperature (40°C)				
Sen	0.8664 (0.0291)	0.8866 (0.0315)	0.8844 (0.0357)	0.9090 (0.0291)
Spe	0.9160 (0.0236)	0.9524 (0.0201)	0.9601 (0.0157)	0.9587 (0.0170)
Youden	0.7824 (0.0336)	0.8390 (0.0398)	0.8445 (0.0427)	0.8677 (0.0371)
High Humidity (50%)				
Sen	0.8758 (0.0274)	0.9317 (0.0221)	0.9478 (0.0193)	0.9392 (0.0234)
Spe	0.9562 (0.0185)	0.9647 (0.0199)	0.9961 (0.0029)	0.9821 (0.0146)
Youden	0.8320 (0.0312)	0.8764 (0.0298)	0.9439 (0.0209)	0.9213 (0.0308)
High luminous Intensity (250lx)				
Sen	0.8990 (0.0276)	0.9290 (0.0247)	0.9259 (0.0251)	0.9211 (0.0278)
Spe	0.8669 (0.0308)	0.8947 (0.0307)	0.9071 (0.0293)	0.9265 (0.0278)
Youden	0.7659 (0.0342)	0.8237 (0.0396)	0.8330 (0.0404)	0.8476 (0.0430)

~ 0.9828, 0.7916 ~ 0.8233). Putting another way, the low Youden index values imply that random forest and SVM are not as efficient as Deep Q-haze with regard to image-based prediction.

4.2.2. Image Sequences of Mobile Phone Camera. We hypothesize whether our predictive model effectively applies to image sequences manually taken. Admittedly, chances are that our proposed method does not work due to unexpected

minute vibration; it is sensible to assess its performance in this scenario. Coherent to experiments above, Tables 4 and 5 show that the proposed models are superior in accuracy to DeepHaze, SVM, and RFs (i.e., Deep Q-haze: 0.8657 ~ 0.8842, random forest: 0.6500 ~ 0.6429, SVM: 0.3148 ~ 0.3287) and in low false detection (i.e., Deep Q-haze: 0.9733 ~ 0.9866, random forest: -0.0781 ~ 0.0303, and SVM: -0.3485 ~ -0.3187). Additionally, it is notable to see that indoor experiment designs generally show better results

compared to outdoor ones. This gap mainly results from the difference in experimental setups. Since extra variables (e.g., light and atmosphere) are adequately adjusted indoors, predictive power of indoor models tends to be superior to models of outdoor environments, where unexpected hardly controlled variables are present.

5. Conclusion

Recently, we dove into the season of burgeoning AI. Many are fascinated with its widespread applicability and practical benefits (e.g., self-driving car, robots, healthcare, etc.). Here we tried to take advantage of the flexible, highly efficient AI technique in air quality monitoring and bring spatial scale of the monitoring down to a “room scale”. Derivation of real-time PM concentrations (even in a semiquantitative way) at a “room scale” is essential, as it can provide information on quality of air that people actually inhale in their everyday life. There is no doubt that it would be even better if the task can be done relatively easily using data readily available to the public. We presented a novel deep learning approach to determine real-time PM10 level whether it is harmful or not from digital images acquired by nonindustrial level recording devices, including mobile phones. Our previous method (DeepHaze, Kim et al. [11]) triggered developing a vision-based predictive model and is found to be applicable in a range of experimental scenarios. Compared to the existing decision rule, Deep Q-haze in the model stretches to additional colorific features (e.g., RGB, HSV, and particulate related features), implicating that the predictive power noticeably improved due to the blessing of big data. Yet, there are urgent needs to synchronize pixels across image sequences (e.g., homogenous configuration), as taking images from flying drones or manual controls is possibly subject to external factors. This homogenous nature serves essential roles to make it to the exquisite differences between consecutive frames. Besides, it is recommended to ensure universal applicability regardless of the type of regions, weather, and the amount of light. Avoiding false detection is an intractable hurdle due to the fact that particulates in image are captured with weak signals for the most part. To enhance utility to the maximum extent by the public, Deep Q-haze is planned to be implemented in portable electronic gadgets in the form of mobile application software. The model needs to have advance extension; the model should be advanced toward multiclass prediction on the basis of moderate calibrations, together with aerosol-related features (e.g., image contrast or visibility [19–21]). To this end, another recurrent neural network-type architecture can potentially be a choice to improve accuracy. We leave these topics for future study.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request. Refer to author’s website (<https://sites.google.com/site/sunghwan-shome/>).

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this manuscript.

Acknowledgments

This paper was supported by Konkuk University in 2018.

References

- [1] S. G. Narasimhan and S. K. Nayar, “Vision and the atmosphere,” *International Journal of Computer Vision*, vol. 48, no. 3, pp. 233–254, 2002.
- [2] D. Muir and D. P. H. Laxen, “Black smoke as a surrogate for PM10 in health studies?” *Atmospheric Environment*, vol. 29, no. 8, pp. 959–962, 1995.
- [3] H. C. Kim, S. Kim, B. Kim et al., “Recent increase of surface particulate matter concentrations in the Seoul Metropolitan Area, Korea,” *Scientific Reports*, vol. 7, no. 1, p. 4710, 2017.
- [4] O. Raaschou-Nielsen, Z. J. Andersen, R. Beelen, E. Samoli, and M. Stafoggia, “Air pollution and lung cancer incidence in 17 European cohorts: prospective analyses from the European Study of Cohorts for Air Pollution Effects (ESCAPE),” *The Lancet Oncology*, vol. 14, no. 9, pp. 813–822, 2013.
- [5] *OECD Economic Surveys: KOREA*, OECD, 2018.
- [6] F. J. Kelly and J. C. Fussell, “Air pollution and public health: emerging hazards and improved understanding of risk,” *Environmental Geochemistry and Health*, vol. 37, no. 4, pp. 631–649, 2015.
- [7] W. Yuchi, E. Gombojav, B. Boldbaatar et al., “Evaluation of random forest regression and multiple linear regression for predicting indoor fine particulate matter concentrations in a highly polluted city,” *Environmental Pollution*, vol. 245, pp. 746–753, 2019.
- [8] M. L. Bell, J. M. Samet, and F. Dominici, “Time-series studies of particulate matter,” *Annual Review of Public Health*, vol. 2, no. 5, pp. 247–280, 2004.
- [9] C. Liu, F. Tsow, Y. Zou, N. Tao, and H. Liu, “Particle pollution estimation based on image analysis,” *PLoS ONE*, vol. 11, no. 2, p. e0145955, 2016.
- [10] Z. He, X. Ye, K. Gu, and J. Qiao, “Learn to predict PM2.5 concentration with image contrast-sensitive features,” in *Proceedings of the 2018 37th Chinese Control Conference (CCC)*, pp. 4103–4106, IEEE, 2018.
- [11] S. H. Kim and S. Kim, “Vision-based predictive model on particulates via deep learning,” *Journal of Electrical Engineering and Technology*, vol. 13, no. 5, pp. 2107–2115, 2018.
- [12] D. Silver, J. Schrittwieser, K. Simonyan et al., “Mastering the game of Go without human knowledge,” *Nature*, vol. 550, no. 7676, pp. 354–359, 2017.
- [13] L. Buşoniu, R. Babuška, and B. De Schutter, “Multi-agent Reinforcement learning: an overview,” in *Innovations in Multi-Agent Systems and Applications–I*, vol. 3, pp. 183–221, Springer, 2010.
- [14] L. Deng and D. Yu, “Deep learning: methods and applications,” *Foundations and Trends in Signal Processing*, vol. 7, no. 3–4, pp. 197–387, 2014.
- [15] R. A. Rohde and R. A. Muller, “Air pollution in china: mapping of concentrations and sources,” *PLoS ONE*, vol. 10, no. 8, p. e0135749, 2015.

- [16] B. Cai, X. Xu, K. Jia, C. Qing, and D. Tao, "DehazeNet: An end-to-end system for single image haze removal," *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5187–5198, 2016.
- [17] P. Carr and R. Hartley, "Improved single image dehazing using geometry," in *Proceedings of the Digital Image Computing: Techniques and Applications*, vol. 1, pp. 3–10, IEEE, 2009.
- [18] N. Jacobs, N. Roman, and R. Pless, "Consistent temporal variations in many outdoor scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–6, 2007.
- [19] W. C. Malm, K. K. Leiker, and J. V. Molenaar, "Human perception of visual air quality," *Journal of the Air Pollution Control Association*, vol. 30, no. 2, pp. 122–131, 1980.
- [20] W. Huang, J. Tan, H. Kan et al., "Visibility, air quality and daily mortality in Shanghai, China," *Science of the Total Environment*, vol. 407, no. 10, pp. 3295–3300, 2009.
- [21] C. A. Olman, K. Ugurbil, P. Schrater, and D. Kersten, "BOLD fMRI and psychophysical measurements of contrast response to broadband images," *Vision Research*, vol. 44, no. 7, pp. 669–683, 2004.