UAV-Enabled Spatial Data Sampling in Large-Scale IoT Systems Using Denoising Autoencoder Neural Network

Tianqi Yu, *Student Member, IEEE*, Xianbin Wang, *Fellow, IEEE*, and Abdallah Shami, *Senior Member, IEEE*

***Abstract*—Internet of Things (IoT) technology has been pervasively applied to environmental monitoring, due to the advantages of low cost and flexible deployment of IoT enabled systems. In many large-scale IoT systems, accurate and efficient data sampling and reconstruction is among the most critical requirements, since this can relieve the data rate of trunk link for data uploading while ensure data accuracy. To address the related challenges, we have proposed an unmanned aerial vehicle (UAV) enabled spatial data sampling scheme in this paper using denoising autoencoder (DAE) neural network. More specifically, a UAV-enabled edge-cloud collaborative IoT system architecture is first developed for data processing in large-scale IoT monitoring systems, where UAV is utilized as mobile edge computing device. Based on this system architecture, the UAV-enabled spatial data sampling scheme is further proposed, where the wireless sensor nodes of large-scale IoT systems are clustered by a newly developed bounded-size *K*-means clustering algorithm. A neural network model, i.e., DAE, is applied to each cluster for data sampling and reconstruction, by exploitation of both linear and nonlinear spatial correlation among data samples. Simulations have been conducted and the results indicate that the proposed scheme has improved data reconstruction accuracy under the sampling ratio without introducing extra complexity, as compared to the compressive sensing-based method.**

***Index Terms*—Data sampling, denoising autoencoder (DAE), large-scale Internet of Things (IoT) system, neural network, spatial correlation, unmanned aerial vehicle (UAV).**

I. Introduction

With the advantages of low cost and flexible deployment, large-scale Internet of Things (IoT) systems have been widely applied to environmental monitoring, including oceanic meteorology monitoring and forest fire surveillance [1]. A general architecture of such system consists of a large number of connected wireless sensor nodes and a cloud platform, where the sensor nodes as data collection layer are pervasively deployed in the target areas for environmental sensing and sampling, while the cloud platform is utilized as the remote data center for data processing and analysis [2].

However, considering the harsh environment of operation fields, wireless communications between sensor nodes are vulnerable to different kinds of obstacles and interference. Additionally, with the enlarging scale of IoT system, tremendous amount of data uploading imposes a heavy burden on the bandwidth requirement of trunk link. Thus, accurate and efficient data sampling and reconstruction is among the most critical technical demands for the design and operation in the cloud-enabled IoT systems. In order to overcome this challenge, unmanned aerial vehicle (UAV) has been introduced into the large-scale IoT system as mobile edge computing device [3]. Here, the UAV-enabled edge device serves as the intermediate layer of IoT system [4]. Given the special location of intermediate layer, the UAV can support real-time responses for the sensor nodes and offload tasks from the cloud by preliminary data processing and analysis. Through the deployment of UAV, an edge-cloud collaborative IoT system architecture has been developed for data processing in large-scale IoT monitoring systems.

Based on this system architecture, a novel spatial data sampling scheme has been further proposed, which can reduce the amount of data sampled at sensor nodes and relieve the bandwidth requirement of the link between UAV and cloud. The principle behind the proposed scheme is the spatial and temporal correlation between sensor data. In a complex environment, the correlation between different types of physical sensor data is not simple as linearity [5]. Therefore, a neural network model, i.e., denoising autoencoder (DAE) [6], is utilized in this paper, which has the capability of compressing both linearly and nonlinearly correlated data.

The proposed sampling scheme consists of three phases, namely, system initialization, model training, and data sampling. During the first stage, a UAV hovers above the target area served by the large-scale IoT system and the cloud. All sensor nodes keep active and upload data to the cloud through UAV. Based on the collected data, sensor nodes are clustered by the newly developed bounded-size *K*-means clustering algorithm. In the second phase, certain sensor nodes within each cluster are selected as data sampling representatives. DAE models for the clusters are trained in the cloud. Parameters of encoders in DAE models are sent to the UAV, while parameters of decoders are kept in the cloud. In the phase of data

Manuscript received April 29, 2018; revised September 17, 2018; accepted October 9, 2018. Date of publication October 18, 2018; date of current version May 8, 2019. This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) under Discovery Grant RGPIN-2018-06254. *(Corresponding author: Xianbin Wang.)*

The authors are with the Department of Electrical and Computer Engineering, Western University, London, ON N6A 5B9, Canada (e-mail: tyu69@uwo.ca; xianbin.wang@uwo.ca; ashami2@uwo.ca).

Digital Object Identifier 10.1109/JIOT.2018.2876695

sampling, data are sampled from selected representatives and then encoded by the UAV before being forwarded to the cloud. The full dataset is finally decoded and reconstructed in the cloud. With the support of cluster formation and UAV, the efficiency of data sampling can be improved. Performance evaluation is conducted, where compressive sensing (CS) as a conventional data sampling method in IoT systems is utilized as the benchmark method. According to the numerical results, the proposed scheme has dramatically improved the data reconstruction accuracy under the same sampling ratio without introducing additional computational complexity.

The contributions of this paper are summarized as follows.

1) A UAV-enabled edge-cloud collaborative IoT system architecture is developed for data processing in large-scale IoT systems, which overcomes the critical challenges of cloud-enabled IoT systems, including high latency, bandwidth overload, and unstable connection to the cloud.
2) A novel spatial data sampling scheme has been proposed for efficient data sampling and reconstruction in the large-scale IoT monitoring systems. In order to fully exploit the spatial data correlation, DAE neural network has been selected as the fundamental data sampling and reconstruction model. With DAE, the sampled data can be precisely reconstructed in the cloud. In the meantime, by locating the encoder in DAE at the UAV, the amount of data uploaded to the cloud is dramatically reduced and thus the burden on the trunk link is relieved.
3) A novel bounded-size K-means clustering algorithm has been developed specifically for cluster formation and the cluster-based spatial data sampling in the proposed scheme. In the novel clustering algorithm, the lower and upper bounds of cluster size are predetermined, which considers the effect of cluster size on the intracluster communications and data sampling.

The remaining of this paper is organized as follows. Section II summarizes the related works on spatial data sampling in IoT systems. In Section III, DAE neural network model is explained in details. The architecture of UAV-enabled edge-cloud collaborative IoT system is developed in Section IV, and the novel spatial data sampling scheme is then proposed in Section V. Performance evaluation is conducted in Section VI. Finally, this paper is concluded in Section VII.

II. Related Work

Spatial correlation-based data sampling in remote sensing field has been well studied in recent years. According to the different fundamental models used, related works are classified into the following categories.

CS is a data compression technique that can map high-dimensional data into sparse domain by utilizing random sensing matrix. In CS-based methods, the sensing field is considered as sparse domain, where data are sparsely sampled from the field and fully recovered at the receiver. Compressive data gathering was the first CS-based method proposed for large-scale wireless sensor networks (WSNs) [7]. WSNs were deployed as the data collection layer of IoT systems. Data were converted to the sparse domain by discrete cosine transform and compressed along multihop routing path. Quer *et al.* [8] proposed a well-developed CS-based framework for data sensing, sampling, and recovery, where principal component analysis (PCA) was used to generate the sparse domain. A cluster-based random sampling algorithm was proposed in [9]. The sparse matrix was generated at the sink by random sampling at both intracluster and intercluster levels.

As stated, several research efforts have been spared on CS-based data sampling in IoT systems, while the weaknesses of these methods are mostly due to the intrinsic constraints of CS technique. Application of CS is limited by the restricted isometry property. However, sparse domain sometimes may not exist for data sampled from complex circumstances. Additionally, although mapping data into special sparse domain can further compress data, the complexity of data recovery algorithm will be dramatically increased as a result.

PCA is a linear correlation-based feature extraction model. Therefore, PCA and variations of PCA-based spatial data aggregation has been widely used in WSNs and IoT systems. In [10], distributed compressive-project PCA was proposed in cooperation with second-order data-coupled clustering algorithm for efficient data collection in large-scale WSNs. Similarly, Yu *et al.* [11] proposed a cluster-based framework as well, aiming at outlier-free data aggregation in IoT systems. The difference was that recursive PCA was used in [11] for adaptively updating PCA models.

Autoencoder (AE) is a neural network model for feature extraction, which can be considered as nonlinear PCA. Given the outstanding performance on data modeling and processing, neural network models have attracted attentions from both industrial and academic institutions. In terms of spatial data sampling in large-scale IoT systems, AE has been used in replace of PCA given the nonlinear processing capability. Alsheikh *et al.* [12] proposed a data compression algorithm with error bound guarantee, where data were spatially compressed by AE-based nonlinear feature extraction.

However, both PCA and AE-based methods sample full dataset from the sensing field, and then spatially compress data at a cluster head or fusion center. By contrast, CS-based methods have the capability of sparsely sampling from the sensing field directly, so that both sampling and communication related processing and cost can be further saved. By exploitation of DAE, our proposed scheme can sample subset of data directly from the sensing field as well. As compared to CS, the data reconstruction accuracy has been improved under the same sampling ratio.

III. DAE Neural Network

The fundamental mathematical model behind our proposed scheme is DAE, which is a neural network model that can be used to reconstruct full dataset from sampled subset [6]. In this section, DAE is explained based on the introduction to basic AE.

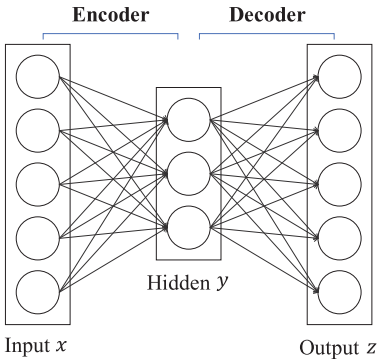

Fig. 1. AE neural network with single hidden layer.

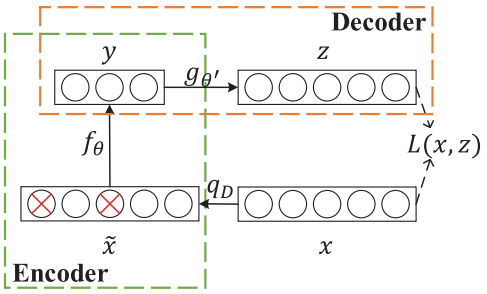

Fig. 2. Structure of DAE.

A. Basic AE

AE is a neural network model for feature extraction. The difference to PCA model is that AE has the capability of dealing with nonlinear data. As a neural network model, AE is also consisted of input, hidden and output layers, while the special case is that the target output of AE is its input. A general structure of AE with single hidden layer is shown in Fig. 1, where the projection from input layer to hidden layer is termed as encoder and hidden to output is decoder.

The mapping function of encoder is expressed as

$$\mathbf{y} = f_\theta(\mathbf{x}) = f\left(\mathbf{W} \cdot \mathbf{x} + \mathbf{b}_f\right) \tag{1}$$

where $\mathbf{x}$ is the input vector in n dimensions, while $\mathbf{y}$ is the hidden layer readout with k units. $f(\cdot)$ is a nonlinear activation function, and sigmoid function is generally adopted. $\mathbf{W}_{[k\times n]}$ is the input weight matrix, and $\mathbf{b}_f$ is the input bias vector.

Correspondingly, the decoder is given by

$$\mathbf{z} = g_{\theta'}(\mathbf{y}) = g\left(\mathbf{V} \cdot \mathbf{y} + \mathbf{b}_g\right) \tag{2}$$

where $\mathbf{z}$ is the output vector with the same dimension as input $\mathbf{x}$. $g(\cdot)$ is the activation function of the decoder. Both identity and sigmoid function are frequently used. $\mathbf{V}_{[n\times k]}$ is the output weight matrix, and $\mathbf{b}_g$ is the output bias vector.

To find out the optimal parameter sets $\theta = \{\mathbf{W}, \mathbf{b}_f\}$ and $\theta' = \{\mathbf{V}, \mathbf{b}_g\}$, the cost function of basic AE is given by

$$\mathcal{J}_{\theta,\theta'} = \frac{1}{m}\sum_{i=1}^{m} \|\mathbf{z}^{(i)} - \mathbf{x}^{(i)}\|_2^2 \tag{3}$$

which penalizes the squared error between input $\mathbf{x}$ and output $\mathbf{z}$. m is the size of training dataset.

B. DAE

Based on the basic AE, DAE is further proposed by Vincent *et al.* [6] to extract features and reconstruct original data from corrupted data as shown in Fig. 2.

Original data $\mathbf{x}$ is corrupted to $\tilde{\mathbf{x}}$ by

$$\tilde{\mathbf{x}} = q_D(\mathbf{x}) \tag{4}$$

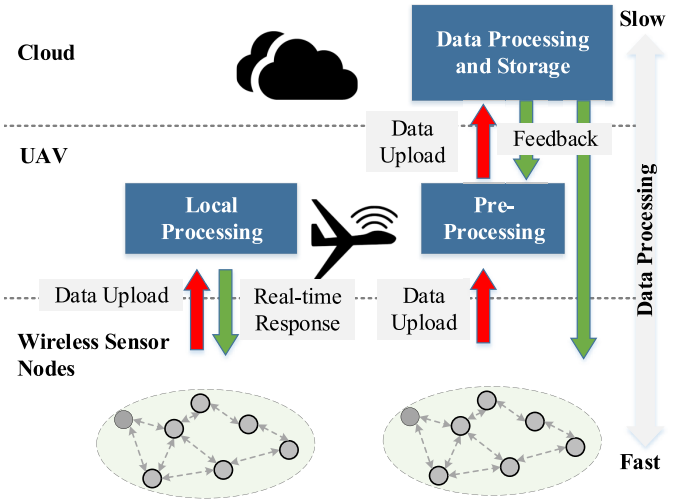

Fig. 3. UAV-enabled edge-cloud collaborative architecture for data processing in large-scale IoT monitoring systems.

where q_D is corruption function. In our data sampling scheme, q_D is defined as a mask function that makes $\tilde{\mathbf{x}}$ a subset of $\mathbf{x}$.

As shown in Fig. 2, the corrupted data vector $\tilde{\mathbf{x}}$ is encoded to $\mathbf{y}$ and then decoded to $\mathbf{z}$ by

$$\mathbf{y} = f_\theta\left(\tilde{\mathbf{x}}\right), \quad \mathbf{z} = g_{\theta'}(\mathbf{y}). \tag{5}$$

Since the objective of DAE is to recover the original data $\mathbf{x}$ from the corrupted data $\tilde{\mathbf{x}}$, the cost function is defined as the squared error between original $\mathbf{x}$ and reconstructed $\mathbf{z}$ as

$$\mathcal{J}_{\theta,\theta'} = \frac{1}{m}\sum_{i=1}^{m}\left\|\mathbf{z}^{(i)} - \mathbf{x}^{(i)}\right\|_2^2 = \frac{1}{m}\sum_{i=1}^{m}\left\|g_{\theta'}\left(f_\theta\left(\tilde{\mathbf{x}}^{(i)}\right)\right) - \mathbf{x}^{(i)}\right\|_2^2. \tag{6}$$

Mini-batch-based gradient descent (GD) algorithm [13] is used to solve the problem and learn the parameters. Though the training procedure occupies certain computational load and memory, it is executed in the cloud platform and does not impose additional burden on the sensor nodes nor the UAVs.

IV. UAV-Enabled Edge-Cloud Collaborative IoT System Architecture

A UAV-enabled edge-cloud collaborative IoT system architecture for data processing in large-scale IoT monitoring systems is developed as shown in Fig. 3, which consists of three major components, namely, wireless sensor nodes as end

devices, UAVs as mobile edge devices and IoT cloud platform. Details of each component are given below.

1) *IoT cloud platform* is the remote data and control center for the IoT system, leveraging cloud computing to achieve complex data processing and analysis, cluster formation for wireless sensor nodes, as well as coordination of UAV flight paths. Particularly, since the training process of the DAE models is too complex to be loaded on either sensor nodes or UAVs, the parameter sets are learned through the training in the cloud. The parameters of encoders in DAE models are then sent to UAV for data encoding. The parameters of decoders are kept in the cloud for data reconstruction.
2) *UAVs* are utilized as mobile edge computing devices, which can support both local processing for the local events with critical real-time requirements and preliminary processing to offload the computational tasks from the cloud and relieve the bandwidth requirements of the underlying trunk link. In terms of wireless communications, UAVs are able to carry different RF modules and support different protocols. For instance, UAVs have the capability of communicating with sensor nodes in a self-organized way through ZigBee modules, and possibly serve as relays to forward the information to the cloud. Therefore, in the proposed scheme, UAV is utilized to collect and encode the sampled data before uploading them to the cloud. Depending on the service area of the large-scale IoT system, one or multiple UAVs could be used. Multiple UAVs can improve the efficiency of data sampling and encoding. However, exploitation of multiple UAVs adds more cost on device management and also introduces the issue of multiple-UAV cooperation into the system in the meantime.
3) *Wireless sensor nodes* are the fundamental components in IoT systems, which are normally deployed in the target areas in random or predetermined way to sense and sample environmental information. For instance, in forest fire surveillance system, temperature, smoke, and humidity sensors are utilized for fire detection. These nodes are able to be self-organized into WSNs. Furthermore, a WSN is modeled as an undirected graph $G = (V, E)$ here. Sensor nodes are modeled as vertices V, and wireless communication links between nodes are modeled as edges E. Degree of a vertex is modeled by the number of valid neighbors of a sensor node. Only the nodes with valid wireless communication capability are defined as valid neighbors.

V. UAV-Enabled Spatial Data Sampling Scheme Using DAE Neural Network

A UAV-enabled spatial data sampling scheme for large-scale IoT monitoring systems is proposed in this section. As stated in Algorithm 1, the scheme consists of three phases, namely, system initialization, model training, and data sampling. The dataflow in three phases is shown in Fig. 4. More details are given in the following paragraphs.

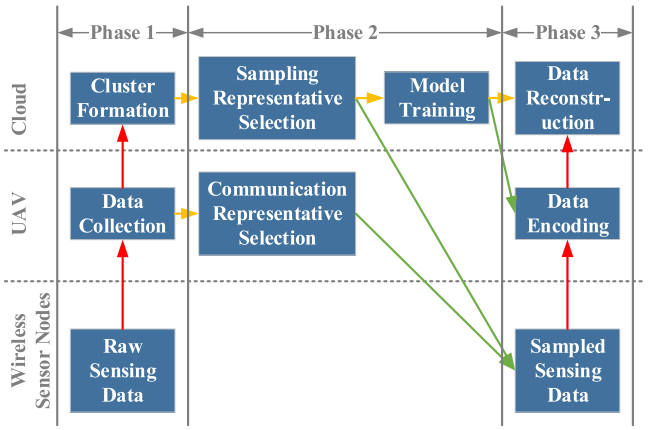

Fig. 4. Dataflow in UAV-enabled spatial data sampling scheme.

Algorithm 1 UAV-Enabled Spatial Data Sampling Using DAE

System Initialization:
set up UAV-IoT communication system
construct the physical topology of WSN in the cloud
UAV hovers above the target area as mobile relay and forwards raw data samples from sensor nodes to the cloud
cluster sensor nodes by Algorithm 2
Model Training:
rank the link quality based on RSSI and LQI at UAV
select communication and data sampling representatives
send *dissociation_notification* to the remaining ones
train $\{\theta, \theta'\}$ with random masks q_D in the cloud
send $\theta = \{\mathbf{W}, b_f\}$ to UAV for data encoding
Data Sampling:
collect data $\tilde{\mathbf{x}}$ from representatives to UAV
if RSSI or LQI is below threshold **then**
  trigger model training procedure
else
  encode data by $\mathbf{y} = f_\theta(\tilde{\mathbf{x}})$, and forward $\mathbf{y}$ to the cloud
end if
the original data is reconstructed by $g_{\theta'}(\mathbf{y})$ in the cloud

A. System Initialization

Wireless communications between the components in IoT system are set up first. More specifically, wireless sensor nodes embedded with ZigBee RF modules are randomly deployed in the target area and self-organized into WSNs. UAV hovers above the target area, and wirelessly communicates with the nodes and the cloud through ZigBee and Wi-Fi, respectively.

1) Physical Topology Construction: Considering the randomness and self-organization features, physical topology of the WSN cannot be known in advance, which needs to be constructed in the cloud by exploitation of the physical topology discovery scheme proposed in our previous work [14]. Physical topology provides the physical locations of sensor nodes and logical topology of the WSN.

2) Raw Data Collection: UAV keeps hovering above the target area and broadcasting beacon signal. According to

Algorithm 2 Bounded-Size K-Means Clustering Algorithm

1: **Input:** node set S, lower bound MIN_CZ, upper bound MAX_CZ, initial value and offset of ε (ε_{INI}, ε_{OFFSET})
2: **initialize** $K = 1$, $\varepsilon = \varepsilon_{INI}$, S_1 as centroid of cluster 1
3: **while** minimal cluster size < MIN_CZ **do**
4: **for** each node S_i in S **do**
5: **for** cluster $j = 1 : K$ **do**
6: **if** Eq.(7) satisfied **and** size of j <MAX_CZ **then**
7: **assign** S_i to cluster j, **update** centroids of j
8: **break**
9: **end if**
10: **end for**
11: **if** S_i is not assigned to existing clusters **then**
12: $K = K + 1$, and S_i as centroid of cluster K
13: **end if**
14: **end for**
15: $\varepsilon = \varepsilon + \varepsilon_{OFFSET}$
16: **end while**
17: **Output:** K and generated clusters

IEEE802.15.4, sensor nodes would passively scan the channel, and send *association_request* to the UAV once the beacon signal is detected [15]. After the association is set up, raw data packets are transmitted from the sensor node to the UAV. UAV measures and records the received signal strength indicator (RSSI) and link quality indicator (LQI) of the received data packet, and then forwards the packet to the cloud.

3) Clustering: Based on the physical locations and raw data obtained in the first two steps, sensor nodes are clustered by the newly proposed bounded-size K-means clustering algorithm in the cloud. Pseudocode is listed in Algorithm 2. In the proposed clustering algorithm, sizes of generated clusters are bounded in range [MIN_CZ, MAX_CZ], which are predetermined lower and upper bounds, respectively.

Physical distance between locations and Euclidean distance between data are jointly utilized as the clustering criterion

$$\left\|\mathbf{l}_i - \mathbf{l}_{C_j}\right\|_2 + \beta \left\|\mathbf{d}_i - \mathbf{d}_{C_j}\right\|_2 \leq \varepsilon \tag{7}$$

where $\mathbf{l}_i$ and $\mathbf{d}_i$ are location and data of sensor node i, while $\mathbf{l}_{C_j}$ and $\mathbf{d}_{C_j}$ indicate average location and data centroids of cluster j. β is the weight to balance these two metrics and ε is the threshold. All collected data are normalized first to remove the impact of different scales.

The procedure of cluster formation using Algorithm 2 is further explained as follows.

1) The first cluster is formed up by regarding the location and data of the first sensor node as cluster centroids.
2) For the remaining nodes in the network, if a node satisfies the clustering criterion of a cluster and the cluster size is not beyond the upper bound MAX_CZ (line 6 in Algorithm 2), the node is assigned to such cluster and the cluster centroids are updated with the new average values of location and data. If a node cannot be assigned to any existing clusters, a new cluster is formed with the location and data of such node as cluster centroids.
3) The proposed bounded-size K-means clustering algorithm is an iterative algorithm, the stopping condition is that the minimal cluster size of the generated clusters is larger than or equal to the lower bound MIN_CZ.

For the generated clusters, the dataset of cluster j at time t, $\mathbf{x}_j^{(t)}$, is the concatenation of data from member sensor nodes, which is considered as the original data vector in DAE.

TABLE I
RECORD OF LINK QUALITY

Device ID	MAC Address	RSSI	LQI	Cluster ID

B. Model Training

Within each cluster, two types of representatives are selected for communication with the UAV and data sampling, respectively. Communication representatives are chosen by the UAV according to link quality, while data sampling representatives are determined by the cloud. Based on the selections, corresponding DAE models are trained for the clusters.

1) Communication Representative Selection: During the stage of system initialization, RSSI and LQI are measured and recorded at UAV as shown in Table I.

RSSI and LQI are jointly used to evaluate the link quality, which is calculated as

$$\text{quality} = \frac{\text{RSSI}}{\text{RSSI\_MAX}} + \frac{\text{LQI}}{\text{LQI\_MAX}} \tag{8}$$

where RSSI and LQI indicate the power strength of received signal and the success of received packet demodulation, respectively. In communication protocols, such as IEEE802.11 and IEEE802.15.4, RSSI and LQI are both defined in range 0x00∼0xFF, namely, RSSI_MAX = 0xFF, LQI_MAX = 0xFF, where higher value indicates better quality. In practical applications, chipset manufacturers can self-define the value of RSSI_MAX and LQI_MAX. However, by scaling RSSI and LQI, the *quality* defined in (8) always ranges from 0 to 2 and 2 indicates the best link quality.

Based on the ranking of *quality*, the sensor node with the best link quality in a cluster is selected as the communication representative. The working mode of the selected node is converted to coordinator. The remaining sensor nodes within the same cluster upload data through the coordinator instead of communicating with UAV directly. By this way, the time duration of UAV-enabled data sampling can be reduced.

2) Data Sampling Representative Selection: Given a cluster j ($j = 1, 2, \ldots, K$), NC_j sensor nodes are contained. NR_j out of NC_j sensor nodes are selected as representatives for data sampling. Based on the knowledge of logical topology, degree of each sensor node can be calculated. Within each cluster, order the member sensor nodes according to node degree. Node with the highest degree and the lowest $(\text{NR}_j - 1)$ ones are selected as data sampling representatives.

The communication representative only communicates with the selected sampling representatives for data uploading, and sends *disassociation_notification* to the remaining ones.

3) Model Training: Random masks are generated to project original data vector $\mathbf{x}_j^{(t)}$ to subset $\tilde{\mathbf{x}}_j^{(t)}$, as shown in Fig. 5. In terms of the masks, a fraction of original $\mathbf{x}_j^{(t)}$ would be

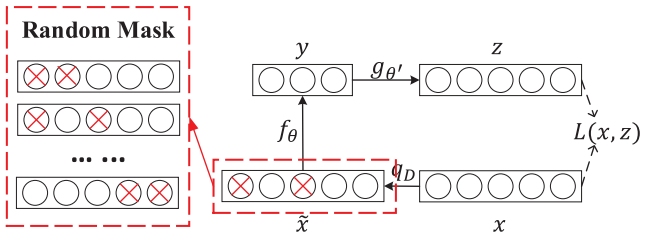

Fig. 5. Masks are randomly generated.

dropped off, namely, $(\mathrm{NC}_j - \mathrm{NR}_j)$ out of NC_j in $\mathbf{x}_j^{(t)}$ would be replaced by **nan** (not a number). Taking Fig. 5 as an example, the original data vector is

$$\begin{aligned}\mathbf{x}_j^{(t)} &= \left[\mathbf{d}_1^{(t)}; \mathbf{d}_2^{(t)}; \ldots; \mathbf{d}_5^{(t)}\right] \\ &= \left[d_{1,1}^{(t)}, d_{1,2}^{(t)}, \ldots, d_{1,p_1}^{(t)}, d_{2,1}^{(t)}, d_{2,2}^{(t)}, \ldots, d_{2,p_2}^{(t)}, \ldots\ldots, d_{5,1}^{(t)}\right. \\ &\qquad \left. d_{5,2}^{(t)}, \ldots, d_{5,p_5}^{(t)}\right] \end{aligned} \tag{9}$$

and the sampling subset is

$$\begin{aligned}\tilde{\mathbf{x}}_j^{(t)} &= \left[\mathbf{nan}; \mathbf{d}_2^{(t)}; \mathbf{nan}; \mathbf{d}_4^{(t)}; \mathbf{d}_5^{(t)}\right] \\ &= \left[\mathrm{nan}, \ldots, \mathrm{nan}, d_{2,1}^{(t)}, d_{2,2}^{(t)}, \ldots, d_{2,p_2}^{(t)}, \mathrm{nan}, \ldots\right. \\ &\qquad \left. \mathrm{nan}, d_{4,1}^{(t)}, d_{4,2}^{(t)}, \ldots, d_{4,p_4}^{(t)}, d_{5,1}^{(t)}, d_{5,2}^{(t)}, \ldots, d_{5,p_5}^{(t)}\right] \end{aligned} \tag{10}$$

where $\mathbf{d}_i^{(t)}$ is the data vector generated by sensor node i in cluster j at time t, and p_i is the number of measured physical variables. Namely, the dimension of $\mathbf{d}_i^{(t)}$ is p_i.

DAE model parameter sets $\{\theta_j, \theta_j'\}$ of cluster j are learned by minimizing the cost function

$$\mathcal{J}_{\theta_j,\theta'_j} = \frac{1}{m}\sum_{t=1}^{m}\left\| g_{\theta'_j}\left(f_{\theta_j}\left(q_{Dt}\left(\mathbf{x}_j^{(t)}\right)\right)\right) - \mathbf{x}_j^{(t)}\right\|_2^2 \tag{11}$$

where q_{Dt} is the mask randomly generated at time t. $f(\cdot)$ is sigmoid function, and $g(\cdot)$ is linear function. m is the amount of historical data samples memorized in the cloud for training. Mini-batch GD algorithm is applied to solve (11). $\theta = \{\mathbf{W}, \mathbf{b}_f\}$ is sent to the UAV for data encoding. $\theta' = \{\mathbf{V}, \mathbf{b}_g\}$ is maintained in the cloud for data reconstruction.

C. Data Sampling

Dataflow of spatial data sampling and reconstruction has been shown in Fig. 4. Data processing at each component is specifically provided as follows.

1) Data Sampling: Based on the clusters setup and representatives selected in Sections V-A and V-B, data are collected from the data sampling representatives to the communication representative and then forwarded to the UAV.

2) Data Encoding: The collected data samples are encoded at the UAV by

$$\mathbf{y}^{(t)} = \frac{1}{1 + e^{-(\mathbf{W}\tilde{\mathbf{x}}^{(t)} + \mathbf{b}_f)}} \tag{12}$$

where $\mathbf{W}$ and $\mathbf{b}_f$ are the parameters obtained from the training in Section V-B3. $\mathbf{y}^{(t)}$ is forwarded to the cloud.

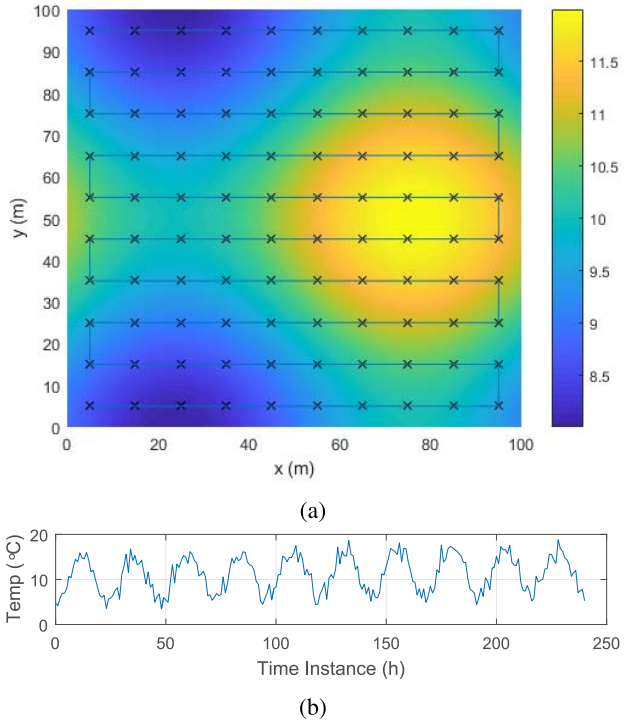

Fig. 6. (a) Geographical distribution and (b) temporal variance of temperature field.

Simultaneously, RSSI and LQI of received data packet are evaluated as well. If either RSSI or LQI is below a predefined threshold, a warning is sent to the cloud. The model training procedure is retriggered cooperatively by UAV and cloud.

3) Data Reconstruction: In the cloud platform, data from each cluster is reconstructed by

$$\mathbf{z}^{(t)} = \mathbf{V}\mathbf{y}^{(t)} + \mathbf{b}_g \tag{13}$$

where $\mathbf{V}$ and $\mathbf{b}_g$ are the parameters learned and maintained from the training in Section V-B3.

VI. Performance Evaluation

Simulations are conducted in this section to analyze the clustering result and accuracy of final data reconstruction, based on the simulation settings given in Section VI-A.

A. Simulation Settings

1) Fundamental Settings: Fig. 6 shows both the geographical distribution and temporal variance of temperature field. Fig. 6(a) is a 100 m × 100 m field, where temperature varies continuously. The temporal trend in Fig. 6(b) indicates the variance of mean value of the geographical temperature field within ten days. The unit of horizontal axis in Fig. 6(b) is hour. One hundred sensor nodes are randomly deployed in the area (not shown). The altitude coordinate of a sensor node is the height of the deployed location. The transmitting power of sensor nodes is homogeneously set to −10 dBm and the receiver sensitivity is −90 dBm.

UAV flight path is also demonstrated in Fig. 6(a). UAV hovers above the target area with an even interval. Hovering

interval has direct influence on the localization accuracy [14], but does not have much effect on the following investigations. Hence, the interval is set to 10 m without losing generality. The hovering height is 20 m above the field. The hovering bias is ±1.5 m in latitude and longitude and ±0.5 m in altitude.

2) Wireless Communication Channel Models: For the signal propagation from UAV to sensor nodes, and peer-to-peer channels among sensor nodes, two-ray ground and free-space outdoor models are, respectively used, considering the different signal propagation environments.

For the air-to-ground signal propagation from UAV to sensor node, two-ray ground model is commonly used, which considers both the line-of-sight (LOS) and ground-reflected rays. For wireless communications among sensor nodes, the signal propagation channel quality is worse, given the potential near-ground scatters. Instead of two-ray ground model, free-space outdoor model is thus adopted. This is a channel model designed specifically for WSNs in the outdoor open areas, which jointly considers the effect of the free-space propagation, ground reflection, RSS uncertainty and antenna radiation impact.

a) Two-ray ground model: For large distance d, the received power P_r (in dBm) can be derived by the two-ray ground model as [16]

$$P_r(dBm) = P_t + 10\log(G_t G_r) + 20\log(H_t H_r) - 40\log(d) \tag{14}$$

where P_t is the transmitting power. d is the horizontal distance between transmitter and receiver. G_t and G_r are the antenna gains of transmitter and receiver, $G_t = G_r = 1$. H_t and H_r are the antenna heights of transmitter and receiver.

b) Free-space outdoor model: The received power is modeled as [17]

$$P_r(dBm) = P_t + 20\log\left(\frac{\lambda}{4\pi d}\right) + 10\log\left(K_1^2 + K_2^2\Gamma^2 + 2K_2\Gamma\cos\left(\frac{2\pi}{\lambda}\Delta L\right)\right) + X_\sigma \tag{15}$$

where λ is the propagation wavelength, and K_1 and K_2 are coefficients irregularity in antenna radiation pattern. ΔL is the path difference between LOS and ground-reflected rays. X_σ is the RSS uncertainty that follows Gaussian distribution. Γ is the ground reflection coefficient

$$\Gamma = \frac{\sin\theta - \sqrt{(\varepsilon - jx_\Gamma) - \cos^2\theta}}{\sin\theta + \sqrt{(\varepsilon + jx_\Gamma) - \cos^2\theta}} \tag{16}$$

where parameters of average ground are used without losing generality, $\varepsilon = 15, x_\Gamma = 3.75 \times 10^{-2}$. θ is the reflection angle.

B. Clustering Analysis

The proposed bounded-size K-means clustering algorithm is analyzed in this section. Since the proposed algorithm is threshold-based, the traditional threshold-based clustering algorithm [18] is selected as benchmark. The improvement of the proposed clustering algorithm as compared to the benchmark method is first provided. Influence of the parameters including lower bound, upper bound, ε_{INI} and $\varepsilon_{\text{OFFSET}}$ on the clustering results is further investigated.

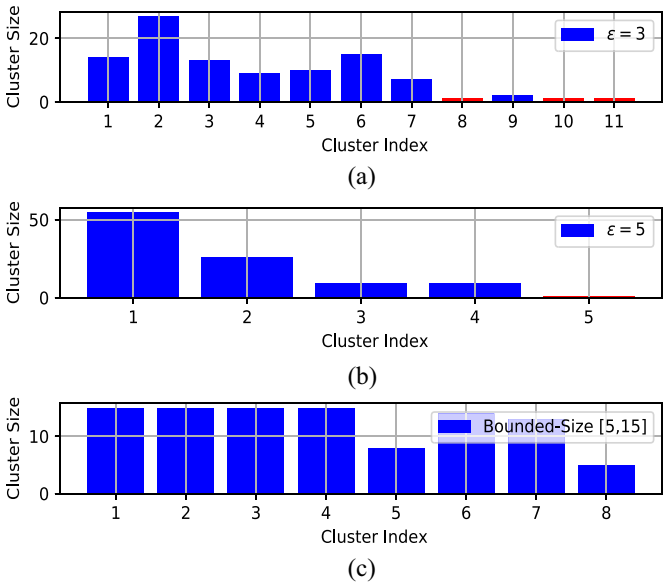

Fig. 7. Comparison between traditional threshold-based clustering algorithm (a) $\varepsilon = 3$ (b) $\varepsilon = 5$ and the proposed bounded-size K-means clustering algorithm (c) [5, 15] and $\varepsilon_{\text{INI}} = 3$.

With the traditional clustering algorithm, when the threshold ε is set to 3, the number of sensor nodes in each of the generated clusters is shown in Fig. 7(a), which illustrates that eleven clusters are generated and three of them contain only a single node as highlighted in red. When $\varepsilon = 5$, five clusters are generated and there is one cluster containing a single node as shown in Fig. 7(b). The results in Fig. 7(a) and (b) indicate that with the increment in threshold ε, the number of clusters with single node decreases indeed. However, it may result in some "huge" clusters in the meantime. The huge cluster refers to the cluster with extremely large amount of sensor nodes, for example, in Fig. 7(b), cluster 1 containing 55 sensor nodes.

In our proposed data sampling scheme, the communication representative in each cluster is functioned as a coordinator and directly communicates with the UAV, while the other cluster members communicate with the coordinator locally. Therefore, in the huge clusters, the intracluster communications would be overload with multiple hops and also vulnerable to the environmental interference of the wild fields. In addition to the huge clusters, in the cluster with single node, the single node has to be regarded as both data sampling representative and communication representative in the meantime, which can result in the early death of such node. Hence, we have added new attributes in the proposed clustering algorithm, namely, the upper and lower bounds of cluster size [MIN_CZ, MAX_CZ]. As shown in Fig. 7(c), when the bounds are set to [5, 15], $\varepsilon_{\text{INI}} = 3$, and $\varepsilon_{\text{OFFSET}} = 0.01$, sizes of the generated clusters are more balanced.

Influence of the parameters on clustering results is shown in Fig. 8, where Fig. 8(a) shows the effect of the lower and upper bounds [MIN_CZ, MAX_CZ] with $\varepsilon_{\text{INI}} = 3$ and $\varepsilon_{\text{OFFSET}} = 0.01$, while Fig. 8(b) shows the influence of ε_{INI} and $\varepsilon_{\text{OFFSET}}$ with MIN_CZ = 2 and MAX_CZ = 15. From Fig. 8(a) it can be seen that with the increment in MIN_CZ, the number

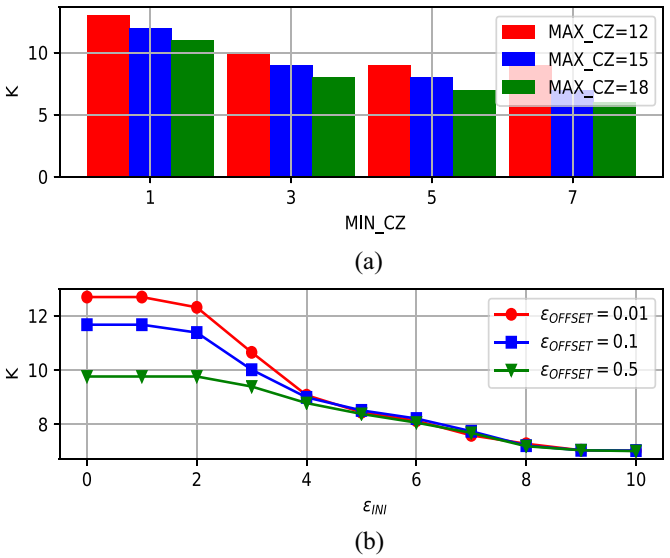

Fig. 8. Influence of parameters on clustering results. (a) MIN_CZ. (b) ε_{INI}.

of clusters generated (namely, K in Fig. 8) decreases, which is due to the iteratively increased threshold ε. In addition, given the fixed MIN_CZ, with the increment in MAX_CZ, the number of clusters generated reduces, which is because the clustering result is mainly affected by the setting of MAX_CZ in such condition. From Fig. 8(b) we can notice that with the increment in ε_{INI}, the number of clusters generated decreases. In the meantime, with the increasing $\varepsilon_{\text{OFFSET}}$, the value of K converges faster. The reason is that with higher threshold ε, more sensor nodes would satisfy the threshold and be gathered into the same cluster and the huge clusters are then bounded by MAX_CZ. Overall, the clustering result is jointly affected by these parameters, which need to be seriously predetermined by the requirements of specific applications.

C. Data Reconstruction Analysis

Data reconstruction accuracy is investigated in this section. Bounds in the clustering algorithm are set to [2, 15], $\varepsilon_{\text{INI}} = 3$, $\varepsilon_{\text{OFFSET}} = 0.01$, and $\beta = 0.1$, and ten clusters are generated. DAE model of each cluster is trained by mini-batch GD algorithm, where the batch size is set to 48. The length of training dataset is 480 (about 20 days), while the length of testing dataset is 120 (5 days).

Fig. 9 is demonstrated as an example, which shows the original temperature readings from 15 sensor nodes within a cluster (labeled 1∼15), and also the sampled and reconstructed values. It indicates that with 12 sensor nodes selected as data sampling representatives, the reconstructed data can have an accurate approximation of the original data.

In order to quantitatively evaluate the reconstruction accuracy, *data reconstruction error* is defined by the average squared $l2$-norm of difference between reconstructed and original data

$$\text{error} = \frac{1}{T}\sum_{t=1}^{T}\left\|\mathbf{z}^{(t)} - \mathbf{x}^{(t)}\right\|_2^2 \tag{17}$$

where $\mathbf{z}$ and $\mathbf{x}$ are reconstructed and original data vectors, respectively. T is the length of testing dataset.

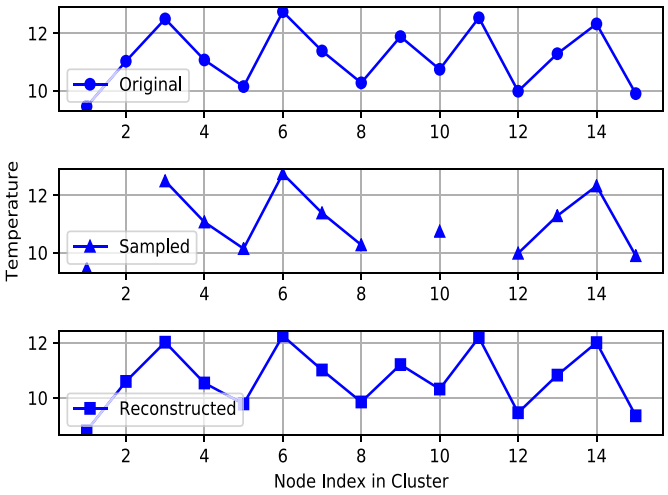

Fig. 9. Original, sampled, and reconstructed temperature values (°C) from 15 sensor nodes within a cluster.

TABLE II
COMPARISON ON DATA SAMPLING REPRESENTATIVE SELECTION CRITERIA

Method	Proposed	Highest	Lowest	Random
Sampling Ratio = 0.6	0.0943	0.1056	0.1003	0.1206
Sampling Ratio = 0.7	0.0217	0.0277	0.0241	0.0249
Sampling Ratio = 0.8	0.0137	0.0140	0.0146	0.0165

1) Data Sampling Representative Selection Analysis: As proposed in Section V-B2, the node with the highest degree and the nodes with lowest degrees in each cluster are selected as the data sampling representatives. Data reconstruction error generated by using the proposed selection criterion is evaluated here, as compared to other selection criteria, including the selection of nodes with highest degrees, selection of nodes with lowest degrees and random selection. Comparison under different sampling ratios is listed in Table II, where the *sampling ratio* refers to the ratio of the number of representatives over the total number of sensor nodes in the cluster.

It can be seen that the data reconstruction error dramatically decreases with the increment in the sampling ratio, while for different selection criteria the difference in error is trivial. The reason is that during the training procedure of DAE model, random masks are used. Therefore, from the perspective of data reconstruction, there is minor difference between these selection criteria. The proposed scheme mainly concerns the physical meanings of the data samples in the actual applications. In the clusters of sensor nodes, the node with the highest degree is located at the hot spot of the cluster and can represent its densely distributed neighbor nodes, while the nodes with lowest degrees are possibly located at the edge of the cluster or the area with sparse node distribution which can hardly be represented by others. That is the reason why the data samples measured by these nodes are collected.

2) Comparison With Compressive Sensing: Comparison on the error generated by two methods under different sampling ratios is shown in Fig. 10, where DAE represents our proposed scheme and CS refers to the CS-based benchmark method. It can be seen that with the increment in the sampling ratio, the data reconstruction error decreases. The reason

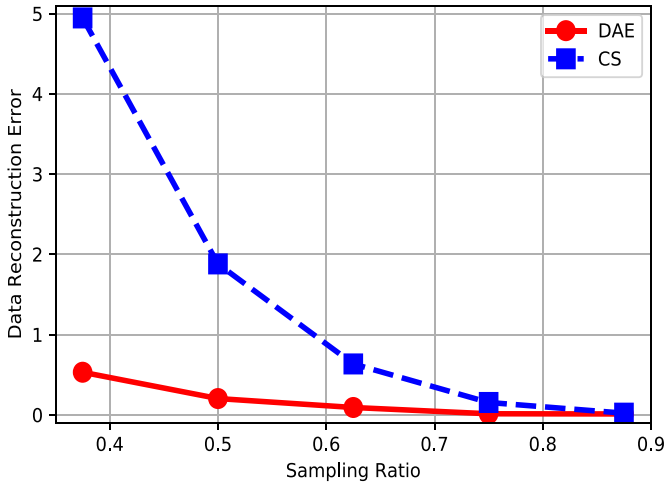

Fig. 10. Comparison on data reconstruction error between DAE and CS under different sampling ratios.

is that with higher sampling ratio, the uncertain proportion of collected data is less, which further improves the reconstruction accuracy. Additionally, the error curves of "DAE" and "CS" indicate that the proposed scheme outperforms CS-based method, especially when the sampling ratio is low.

In terms of the complexity analysis, the computational complexity of CS-based method is dominated by the recovery algorithm. Therefore, the overall complexity is determined by the selection of recovery algorithm. In our simulation, iterative reweighted least squares (IRLS) algorithm is exploited for data recovery [19]. While for DAE-based method, the computational complexity of the proposed method is dominated by the model training procedure, where mini-batch GD algorithm is used to learn the parameters. IRLS and mini-batch GD are both iterative algorithms. IRLS algorithm needs fewer iterations to converge, while the cost of IRLS at each iteration is higher [20]. Therefore, the comparison on the computational complexity between IRLS and mini-batch GD is determined by the features of data.

VII. Conclusion

In order to address the challenge of accurate and efficient data sampling and reconstruction in large-scale IoT systems, we have proposed a cluster-based spatial data sampling scheme using DAE neural network, by exploitation of the spatial data correlation. UAV was utilized as the mobile edge device and an edge-cloud collaborative data processing architecture was then developed, where wireless sensor nodes and cloud platform were involved for environmental sensing and complex data analysis, respectively. In achieving the cluster formation for the proposed data sampling scheme, a novel bounded-size K-means clustering algorithm was proposed. A neural network model, DAE, was exploited to fully extract the spatial data correlation and perform data sampling and reconstruction for each cluster. Specifically, the encoders in DAE models were deployed at the UAV for encoding data collected from sampling representatives, while the decoders were located in the cloud for data reconstruction. Simulations were conducted and numerical results indicated that our scheme improved the data reconstruction accuracy under the same sampling ratio, as compared to the CS-based method.

References

[1] G. Xu, E. C.-H. Ngai, and J. Liu, "Ubiquitous transmission of multimedia sensor data in Internet of Things," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 403–414, Feb. 2018.

[2] H. Cai, B. Xu, L. Jiang, and A. V. Vasilakos, "IoT-based big data storage systems in cloud computing: Perspectives and challenges," *IEEE Internet Things J.*, vol. 4, no. 1, pp. 75–87, Feb. 2017.

[3] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Mobile unmanned aerial vehicles (UAVs) for energy-efficient Internet of Things communications," *IEEE Trans. Wireless Commun.*, vol. 16, no. 11, pp. 7574–7589, Nov. 2017.

[4] M. Chiang and T. Zhang, "Fog and IoT: An overview of research opportunities," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 854–864, Dec. 2016.

[5] F. Alam, R. Mehmood, I. Katib, N. N. Albogami, and A. Albeshri, "Data fusion and IoT for smart ubiquitous environments: A survey," *IEEE Access*, vol. 5, pp. 9533–9554, 2017.

[6] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. ACM 25th Int. Conf. Mach. Learn.*, Helsinki, Finland, 2008, pp. 1096–1103.

[7] C. Luo, F. Wu, J. Sun, and C. W. Chen, "Compressive data gathering for large-scale wireless sensor networks," in *Proc. 15th Annu. Int. Conf. Mobile Comput. Netw.*, 2009, pp. 145–156.

[8] G. Quer, R. Masiero, G. Pillonetto, M. Rossi, and M. Zorzi, "Sensing, compression, and recovery for WSNs: Sparse signal modeling and monitoring framework," *IEEE Trans. Wireless Commun.*, vol. 11, no. 10, pp. 3447–3461, Oct. 2012.

[9] X. Li, X. Tao, and G. Mao, "Unbalanced expander based compressive data gathering in clustered wireless sensor networks," *IEEE Access*, vol. 5, pp. 7553–7566, 2017.

[10] M. I. Chidean, E. Morgado, E. del Arco, J. Ramiro-Bargueño, and A. J. Caamaño, "Scalable data-coupled clustering for large scale WSN," *IEEE Trans. Wireless Commun.*, vol. 14, no. 9, pp. 4681–4694, Sep. 2015.

[11] T. Yu, X. Wang, and A. Shami, "Recursive principal component analysis-based data outlier detection and sensor data aggregation in IoT systems," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 2207–2216, Dec. 2017.

[12] M. A. Alsheikh, S. Lin, D. Niyato, and H.-P. Tan, "Rate-distortion balanced data compression for wireless sensor networks," *IEEE Sensors J.*, vol. 16, no. 12, pp. 5072–5083, Jun. 2016.

[13] G. Hinton, N. Srivastava, and K. Swersky. (2012). *Neural Networks for Machine Learning, Lecture 6A, Overview of Mini-Batch Gradient Descent*. [Online]. Available: https://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf

[14] T. Yu, X. Wang, J. Jin, and K. McIsaac, "Cloud-orchestrated physical topology discovery of large-scale IoT systems using UAVs," *IEEE Trans. Ind. Informat.*, vol. 14, no. 5, pp. 2261–2270, May 2018.

[15] LAN/MAN Standards Committee IEEE Computer Society, *IEEE Standard for Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks Specific Requirements Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)*, IEEE Standard 802.15.4-2003, 2003. [Online]. Available: http://profsite.um.ac.ir/~hyaghmae/ACN/WSNMAC1.pdf AQ2

[16] S. Kurt and B. Tavli, "Path-loss modeling for wireless sensor networks: A review of models and comparative evaluations," *IEEE Antennas Propag. Mag.*, vol. 59, no. 1, pp. 18–37, Feb. 2017.

[17] T. Stoyanova, F. Kerasiotis, A. Prayati, and G. Papadopoulos, "A practical RF propagation model for wireless network sensors," in *Proc. IEEE SENSORCOMM*, Glyfada, Greece, 2009, pp. 194–199.

[18] S. K. Bhatia, "Adaptive K-means clustering," in *Proc. AAAI 7th Int. Florida Artif. Intell. Res. Soc. Conf.*, Menlo Park, CA, USA, 2004, pp. 695–699.

[19] R. Chartrand and W. Yin, "Iteratively reweighted algorithms for compressive sensing," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, Las Vegas, NV, USA, 2008, pp. 3869–3872.

[20] T. P. Minka. (2007). *A Comparison of Numerical Optimizers for Logistic Regression*. [Online]. Available: https://tminka.github.io/papers/logreg/minka-logreg.pdf

Tianqi Yu (S'16) received the B.E. degree in communication engineering from Wuhan University, Wuhan, China, in 2013, and the M.E.Sc. degree in electrical and computer engineering from Western University, London, ON, Canada, in 2015, where she is currently pursuing the Ph.D. degree in electrical and computer engineering.

Her current research interests include wireless sensor networks, edge computing, and data analytics in Internet of Things systems.

Ms. Yu was a recipient of the Best Student Paper Award of IEEE VTC 2015 Spring.

Xianbin Wang (S'98–M'99–SM'06–F'17) received the Ph.D. degree in electrical and computer engineering from the National University of Singapore, Singapore, in 2001.

He was with Communications Research Centre Canada as a Research Scientist/Senior Research Scientist from 2002 to 2007. He is a Professor and Tier-I Canada Research Chair with Western University, London, ON, Canada. From 2001 to 2002, he was a System Designer with STMicroelectronics, where he was responsible for the system design of DSL and Gigabit Ethernet chipsets. His current research interests include 5G technologies, Internet-of-Things, communications security, machine learning and locationing technologies. He has over 300 peer-reviewed journal and conference papers, in addition to 26 granted and pending patents and several standard contributions.

Dr. Wang was a recipient of many awards and recognitions, including the Canada Research Chair, CRC President Excellence Award, the Canadian Federal Government Public Service Award, the Ontario Early Researcher Award, and five IEEE Best Paper Awards. He currently serves as an Editor/Associate Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS, the IEEE TRANSACTIONS ON BROADCASTING, and the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY. He was also an Associate Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS from 2007 to 2011 and IEEE WIRELESS COMMUNICATIONS LETTERS from 2011 to 2016. He was involved in many IEEE conferences including GLOBECOM, ICC, VTC, PIMRC, WCNC, and CWIT, in different roles such as the Symposium Chair, the Tutorial Instructor, the Track Chair, the Session Chair, and the TPC Co-Chair. He is a Fellow of the Canadian Academy of Engineering and an IEEE Distinguished Lecturer.

Abdallah Shami (M'03–SM'09) received the B.E. degree in electrical and computer engineering from Lebanese University, Beirut, Lebanon, in 1997, and the Ph.D. degree in electrical engineering from the Graduate School and University Center, City University of New York, New York, NY, USA, in 2002.

In 2002, he joined the Department of Electrical Engineering, Lakehead University, Thunder Bay, ON, Canada, as an Assistant Professor. Since 2004, he has been with Western University, London, ON, Canada, where he is currently a Professor with the Department of Electrical and Computer Engineering. His current research interests include network optimization, cloud computing, and wireless networks.