International Journal for Modern Trends in Science and Technology
Volume: 03, Special Issue No: 02, March 2017
ISSN: 2455-3778
http://www.ijmtst.com

# Using Kernel Dataset from Sensory Data in Wireless Sensor Networks

V.K.Yamini[1] | A.Sri Lakshmi[2] | Md.Rubeena[3] | N.Mrinalini[4]

[1,2,3,4]Department of IT, Andhra Loyola Institute of Engineering & Technology, Vijayawada, Andhra Pradesh, India.

**To Cite this Article**
V.K.Yamini, A.Sri Lakshmi, Md.Rubeena and N.Mrinalini, "Using Kernel Dataset from Sensory Data in Wireless Sensor Networks", *International Journal for Modern Trends in Science and Technology*, Vol. 03, Special Issue 02, 2017, pp. 52-56.

## ABSTRACT

*The amount of sensory data usage faces an huge growth due to the increasing usage and popularity of Wireless Sensor Networks(WSNs). The scale of sensory data in many applications has already exceeded several petabytes annually, which is beyond the computation and transmission capabilities of conventional WSNs. On the other hand, the information carried by big sensory data has high redundancy because of strong correlation among sensory data. In this paper, we introduce the novel concept of $\epsilon$-Kernel Dataset, which is only a small data subset and can represent the vast information carried by big sensory data with the information loss rate being less than $\epsilon$, where $\epsilon$ can be arbitrarily small. We prove that drawing the minimum $\epsilon$-Kernel Dataset is polynomial time solvable and provide a centralized algorithm with time complexity. Furthermore, a distributed algorithm with constant complexity O(1) is designed. It is shown that the result returned by the distributed algorithm can satisfy the $\epsilon$ requirement with a near optimal size. Furthermore, two distributed algorithms of maintaining the correlation coefficients among sensor nodes are developed. Finally, the extensive real experiment results and simulation results are presented. The results indicate that all the proposed algorithms have high performance in terms of accuracy and energy efficiency.*

***KEYWORDS:*** *Big Sensory Data, Kernel Dataset, Wireless Sensor Networks*

## I. Introduction

With the increasing popularity of Wireless Sensor Networks (WSNs), the amount of sensory data usage has an explosive growth. In many applications, the amount of the sensory data has already exceeded several petabytes (PB) annually. At present, the Large Hadron Collider experiment occurred in Europe use's 150 million sensors that deliver data over 40 million times per second. InBeijing There are about 67,000 taxis and 400 thousand electronic eyes in resulting more than 48PB GPS data and 1440PB other monitoring data each year. The data storage Capability whose growth rate is only 40% per year but, the growth rate of sensory data is more than 58%. The world already produced over twice much data as that is stored in 2011 [2].All the information and increasing usage of WSN leads to the era of Big Sensory Data (BSD), which brings us many new challenges as well as opportunities. Correlation. Therefore, a group of new data collection and computation algorithms are expected for BSD management.Since the volume of BSD is beyond the computation and transmission capabilities of conventional WSNs, one feasible solution is to is to dramatically reduce the amount of sensory data involved in computation, which is known as"From big to Small". Based on such a motivation, several algorithms were proposed. These algorithms sample a small portion of sensory data to answer queriesbased on the user-specified precision requirements. Although the sampling based algorithms are efficient and effective for processing queries in WSNs for BSD, the characteristics and correlations of sensory data are overlooked during sampling and collection periods. Thus, they are only suitable for some simple queries such as aggregation, and unable to restore the original information with high precision. Another option to reduce the amount of BSD is compression. Many sensory data compression techniques have been proposed, including the sketch based compression linear regression based compression, source coding based compression,information entropy based compression et. However, for most existing compression methods, the computation task on the compressed data cannot be carried out without a decompression process, which results in additional energy and time consumptions. Other compression techniques, such as the Sketch based algorithm in only deals with aggregation queries. The above facts motivate us to investigate a new data reduction algorithm which can support both recovery and efficient computation of BSD. Recently, many research works reveal that sensory data are strong correlated in both temporal and spatial spaces since the monitored physical world always varies continuously in space and time. Such strong correlations incur high redundancy in BSD. More specifically, a BSD set in a time window can be regarded as a data matrix with size m × n, where m is the sampling times in the given time window and n is the number of sensors. Such a data matrix has high redundancy due to strong correlations among sensory data. Thus, another data matrix with much smaller dimension (size) is expected to represent the original one. Such a smaller data matrix is referred as a Kernel Dataset of BSD. For any given threshold ϵ, an ϵ-Kernel Dataset is denoted by a data matrix with theinformation loss rate being smaller than ϵ compared with the original BSD. In order to reduce the costs of storage, transmission and processing as much as possible, the optimal goal of seeking an ϵ-Kernel Dataset is to minimize its size. Based on our experimental results,an ϵ-Kernel Dataset saves more than 90% storage resource on condition that 95% information of the original BSD is captured. Similarly, the costs of transmission and processing can be significantly reduced by managing an ϵ-Kernel Dataset instead of the original one. Meanwhile, it can effectively support data recovery and computation without decompression. Due to these reasons, we investigate how to extract an ϵ-Kernel Dataset for BSD. The main contributions in these paper are as follows.The definitions of information loss rate, ϵ-Kernel Dataset and minimum ϵ-Kernel Dataset are introduced.We prove that extracting a minimum ϵ-Kernel Dataset can be solved in polynomial time, and propose a centralized algorithm to solve it in O(n3) time. The computation and communication complexities of the algorithm are analysed. (3). To reduce the cost of the centralized algorithm, a distributed algorithm with constant complexity O(1) is designed. We prove that the result returned by the distributed algorithm can satisfy the ϵ requirement with a near optimal size. The proposed algorithms have high performance in terms of accuracy and energy efficiency. This is large enough so that there exist at least two distinct values in S(w) i,1 ≤ i ≤ n.All the sensed values collected during [T(s) w ,T(f) w ] can be denoted by S(w) = [S(w) 1 ,S(w) 2 ,...,S(w) n ], where S(w) is an m×n matrix. Since the size of a WSN (n) is usually quite large, the transmission and storage of S(w) cost a huge amount of energy and may be impossible in some cases. Therefore, we expect to find a subspace, denoted by [U1, U2,Up] so that S(w) can project in this subspace and get a smaller data matrix, where p ≪ n. Meanwhile, it is expected that the information lost by such a projection is minimized. In this paper, the information of loss rate of a projection is measuredbytheproportionofthedatacharacteristics itdrops. Considering that the metrics of different types of sensory data are not the same, e.g. the metric of light data is different from that of temperature data. It is difficult to evaluate the data characteristics of sensory data from different nodes. Thus, the normalization of sensory data is required. Based on the normalization method [21], the definitions of data characteristic vector and data characteristic matrix are given as follows.

## II. KERNEL DATASET DRAWING ALGORITHMS

### *2.1 Centralized Algorithm*

The centralized algorithm to draw the ϵ-Kernel Dataset in a given time window [T(w) s ,T(w) f ] contains five steps.

**First** the sensors in a network are organized as a spanning tree rooted at the sink. The spanning tree construction and maintenance algorithms are in. The sink broadcasts a command along the spanning tree when it expects the ϵ-Kernel Dataset.

**Second,** let $T(f)\ w = tc$ and $T(s)\ w = T(f)\ w - m/f$ be the end time and start time of the current time window. Each sensori ($1 \leq i \leq n$) transmits $\{r(w)\ ij \mid j \in Ni\}$ to the sink along the spanning tree.**(**Node j is called a neighbor of node i if $dis(i,j) \leq d$. $Ni = \{j \mid dis(i,j) \leq d\}$ denotes the neighbor set of i, where dis(i,j) is the distance between i and j**)**

**Third**, the sink computes the eigenvalues and eigenvectors of C(w). Let $\lambda 1,\lambda 2,\ldots,\lambda n$ be the eigenvalues of C(w), and Ii be the associate eigenvector of λi for any $1 \leq i \leq n$, where $\lambda 1 \geq \lambda 2 \geq \ldots \geq \lambda n$, and $I1,I2,\ldots,In$ are standard orthogonal bases.

**Fourth** let Iij denote the i-th element of vector Ij, where $1 \leq i \leq n$ and $1 \leq j \leq q$. For any sensor i ($1 \leq i \leq n$), it computes a new data matrix Di by $Di = [Ii1S(w)\ i\ ,Ii2S(w)\ i\ ,\ldots,IiqS(w)\ i\ ]m\times p$, where $S(w)\ i = [sitw1,\ldots,sitwm]T$ denotes the sensory values of i in $[T(s)\ w\ ,T(f)\ w\ ]$.

**Fifth** each node i transmits data matrix Di along the spanning tree towards the sink. The data matrices from different nodes are added together during the transmission. Finally, the sink obtains ϵ-Kernel Dataset D(w) q in the current time window, i.e., $D(w)\ q = \sum n\ i=1\ Di$.

From collorythe size of D(w) q returned by the centralized algorithm is also minimum.

Next, we analyse the computation and communication complexities of the above centralized algorithm, where e1 and e2 are the energy costs of a sensor node for sending and receiving one byte, and $dmax = max1 \leq i \leq n|Ni|$.

## III. Performance Evaluation

To evaluate the performance of the Correlation Coefficient Matrix Maintenance algorithm, we firstly use TelosB motes to continuously sample indoor temperature, humidity and light intensity. The transmission radius of each sensor is 20m. The data sampling system is built on TinyOS 2.1.0., and the light intensity datasets are used for evaluation. Secondly, we use Tossim to simulate a network with 1024 sensor nodes. The network is deployed in a 160m × 160m rectangular region,and the transmission radius of each sensor is also 20m. The sensorydatais based on the real dataset from the Intel Berkeley Research Lab. To evaluate the performance of the centralized and distributed Kernel Dataset drawing algorithms in large scale networks, we use two simulators, Tossim and NS2, construct simulated networks with different sizes. For a network whose size is smaller than 1024, Tossim is used. Otherwise, NS2 is adopted. The transmission radius of each sensor node in all the simulated networks is 20m, and the sensory data is also generated.The energy cost of a sensor to send and receive 1 byte message is 0.0144mJ and 0.0057mJ. For convenience, we use CCM and S-CCM to denote the accurate Correlation Coefficient matrix Maintenance algorithm and the Sampling based Correlation Coefficient matrix Maintenance algorithm respectively, and use Centralized-KDD and Distributed-KDD to represent the centralized and distributed Kernel Dataset Drawing algorithms.

### 3.1 Performance of CCM

The correlation coefficient between two different sensors will decline sharply with the growth of their distance due to the spatial correlation. To verify such a fact, the following experiments are carried out. The first group of experiments is based on the real sensor networks. In the experiments, the absolute value of the correlation coefficient of two sensors is calculated while their distance d increases from 0m to 8m, and the time window size m is 30, 50 and 100 respectively. The results indicate two facts. First, the absolute value of the correlation coefficient of two sensors declines sharply with the increment of d. Considering that the correlation coefficient of two sensors is almost 0 when d is increased to the transmission radius, the sensors correlated with sensor i ($1 \leq i \leq n$) can be reached by one-hop transmission in most cases. Second, the correlation coefficient calculated when m = 50 is almost the same as that in the case when m = 100, therefore, it becomes stable when the time window size reaches a certain value. Thesecond groupof experimentsis to investigatetheimpact of d on the correlation of two sensors in the simulated networks. In the experiments, the absolute value of the correlation of two sensors is calculated while their distance d increases from 0m to 14m, and the time window size m is 10, 20, and 50 respectively.The third group of experiments is to investigate the relationship between the energy cost of CCM and d. Since CCM is divided into two phases, the energy costs of these two phases are investigated separately. In the experiments, the energy cost by the two phases are calculated while d is increased from 0m to 20m, and m is 20 and 50. The energy consumed by the two phases increases with the growth of d since the sensory data of each sensor need to be broadcasted in range of d. It also shows that the energy costby the maintenance phase is extremely small compared with the initial phase. Since the initial phase only happens once, the total energy cost of CCM is very small.

The fourth group of experiments is to investigate the impact of m on the energy consumed by CCM. In the experiments, the energy costs of two phases are calculated while m increases from 10 to 50, and d is set to be 10m and 20m.

However, the energy cost of the maintenance phase is stable even when m becomes large. Because the history information is fully used while the time window is sliding so that the energy cost is independent with m. Moreover, the total energy cost of CCM is very small since the maintenance phase happens many times and the energy cost of CCM mainly depends on it.

### *3.2Performanace of S-CC,Centralised and Distribted KDD*

the S-CCM algorithm is designed for a quite large time window since the correlation coefficient between two sensors becomes stable when the size of time window, m, is large. Moreover, it is also not necessary to use the sampling based algorithm as the accurate algorithm consumes quite little energy when m issmall. Due to these two reasons, we only evaluate the performance of S-CCM when m is large.

The first group of the experiments is to investigate the variance of correlation coefficient with growth of m in the real and simulated sensor networks, where the length of a period equals the number of snapshots it contains. In the real sensor network, two pairs of TelosB motes are randomly selected to sense light intensity, and their correlation coefficient is calculated while m increases from 40 to 180.

The second group of the experiments is to investigate the sampling ratio of S-CCM. The sampling ratio is equal to the percentage of sensory data being sampled for calculation, and it is important to evaluate the performance of a sampling-based algorithm. Firstly, the sampling ratio are calculated while θ is increased from 0.03 to 0.3, δ is in {0.01,0.2}, and m is in {10000,5000}. For example, the sampling ratio is less than 3.5% when θ = 0.03, δ = 0.01 and m = 10000. Therefore, our S-CCM algorithm saves lots of energy for computing the correlation coefficient of sensor nodes since the required sampling ratio is quite small. Meanwhile, it also shows that our S-CMM algorithm is more efficient to deal with the situation that the length of the given period is quite large since the sampling ratio becomes smaller for a long period. The last group of the experiments is to investigate the accuracy of S-CMM. In the experiments, the error between the accurate correlation coefficient and the approximate one returned by S-CMM is calculated while the sampling ratio increases from 0.01 to 0.3, and m is in {10000,5000}. The results are shown in Fig.6. It shows that the error generatedby our S-CMM algorithm is extremely small even when few sensory values are sampled. For example, the error of SCMM is less than 0.03 if the sample ratio is larger than 0.05. Furthermore, it verifies that S-CMM is more suitable for calculating correlation coefficient in a longer period since the error generated by it when m = 10000 is smaller than that when m = 5000 in most cases according to Fig.6. Finally, since S-CMM only needs to run once during a long period, much energy is saved for determining the correlation coefficient between two sensor nodes.

### 3.3Performance of Centralized and Distributed KDD

The ratio of a Kernel Dataset equals to its size divided by the size of the whole data generated by a WSN in a time window. It is an important parameter. To evaluate the compression ability of a Kernel Dataset drawing algorithm.The first group of experiments is to investigate the relationship between ϵ, n and the ratios of ϵ-Kernel Datasets returned by Centralized-KDD and Distributed-KDD respectively, where n denotes the size of the network. Firstly, the ratios of ϵ-Kernel Datasets returned by the two algorithms are calculated while ϵ is increased from 0.05 to 0.5, and n ∈ {400,1024,2025}. Secondly, such ratios are calculated while n is increased from 441 to 2025, and ϵ is 0.05, 0.25 and 0.4. the ratios of ϵ-Kernel Datasets are quite small even when ϵ is small for both Centralized-KDD and Distributed-KDD. For example, the ratios of the Kernel Datasets returned by both of the two algorithms are less than 6.55% when ϵ = 0.05 and n = 1024, i.e., the Kernel Dataset only uses 6.55% data values to guarantee that 95% information from the raw sensory data is preserved. Since the density of the networks becomes larger and more redundant data are generated with the growth of n. Thus, more useless data are filtered by the two algorithms.

The second group of the experiments is to compare the ratios of the ϵ-Kernel Datasets returned by Centralized-KDD and Distributed-KDD. In the experiments, we firstly calculate the ratios of the ϵ-Kernel Datasets returned by CentralizedKDD and Distributed-KDD while ϵ is increased from 0.1 to 0.4 and n = 1024. Then, these ratios are computed while n is increased from 400 to 2025 and ϵ = 0.2. that the ratios of the Kernel Dataset drawn by the two algorithms are almost the same, which verifies the near optimal property of Distributed-KDD since the size of dominant data drawn by Centralized-KDD is minimized.

## IV.CONCLUSION

This paper studies how to draw the minimum ϵ-Kernel Dataset from a WSN. We prove that it is a problem and provide an accurate centralized algorithm with O(n3) complexity. A distributed algorithm with constant complexity is also proposed to draw the ϵ-Kernel Dataset in order to save energy and computation resources. We prove that the result returned by the distributed algorithm has a near optimal size. Furthermore, two in-network correlation coefficient matrix maintenance algorithms are designed. The extensive experimental results verify that the proposed algorithms have high performance in terms of accuracy and energy efficiency

***Future Enhancement***

Furthermore, two in-network correlation coefficient matrix maintenance algorithms are designed. The extensive experimental results verify that the proposed algorithms have high performance in terms of accuracy and energy efficiency. In the future we may have in-network correlation coefficient matrix maintenance algorithms.

## References

[1] G. Brumfiel, “Down the petabyte highway,” Nature, vol. 469, no. 20, pp. 282–283, 2011.

[2] R. G. Baraniuk, “More is less: signal processing and the data deluge,” Science(Washington), vol. 331, no. 6018, pp. 717–719, 2011.

[3] M. Gupta, L. V. Shum, E. L. Bodanese, and S. Hailes, “Design and evaluation of an adaptive sampling strategy for a wireless air pollution sensor network,” in LCN, 2011, pp. 1003–1010.

[4] D. J. Abadi, S. Madden, and W. Lindner, “Reed: Robust, efficient filtering and event detection in sensor networks,” in VLDB, 2005, pp. 769–780.

[5] Z. Cai, R. Goebel, and G. Lin, “Size-constrained tree partitioning: approximatingthemulticastk-treeroutingproblem,”Theoretical Computer Science, vol. 412, no. 3, pp. 240–245, 2011.

[6] J. Considine, F. Li, G. Kollios, and J. Byers, “Approximate aggregation techniques for sensor databases,” in ICDE, 2004, pp. 449–460.

[7] Z. Cai, G. Lin, and G. Xue, “Improved approximation algorithms for the capacitated multicast routing problem,” in International.

[8] S. Cheng and J. Li, “Sampling based (ϵ, δ)-approximate aggregation algorithm in sensor networks,” in ICDCS, 2009, pp. 273–280.

[9] S. Cheng, J. Li, Q. Ren, and L. Yu, “Bernoulli sampling based (ϵ, δ)approximate aggregation in large-scale sensor networks,” in INFOCOM, 2010, pp. 1181–1189.

[10] J. Li and S. Cheng, “(ϵ, δ)-approximate aggregation algorithms in dynamic sensor networks,” TPDS, vol. 23, no. 3, pp. 385–396, 2012.

[11] Z. He, Z. Cai, S. Cheng, and X. Wang, “Approximate aggregation for tracking quantiles and range countings in wireless sensor networks,” Theor. Comput. Sci., vol. 607, pp. 381–390, 2015.

[12] J. Considine, F. Li, G. Kollios, and J. Byers, “Approximate aggregation techniques for sensor databases,” in ICDE, 2004, pp. 449–460.

[13] C.-H. Wu and Y.-C. Tseng, “Data compression by temporal and spatial correlations in a body-area sensor network: A case study in pilates motion recognition,” IEEE TMC, vol. 10, no. 10, pp. 1459–1472, 2011.

[14] A. Deligiannakis and Y. Kotidis, “Data reduction techniques in sensor networks,” IEEE Data Eng. Bull., vol. 28, no. 1, pp. 19–25, 2005.

[15] A. Deligiannakis, Y. Kotidis, and N. Roussopoulos, “Dissemination of compressed historical information in sensor networks,” VLDB J., vol. 16, no. 4, pp. 439–461, 2007.

[16] Sensory data in wireless sensor networks,” in INFOCOM, 2015, pp. 531–539.