A Ubiquitous Data Logger
that Selects an Efficient Data Compression Method

Kenji Ono
Graduate School of Engineering
Kobe University
Kobe, Japan
Email: k-ono@stu.kobe-u.ac.jp

Tsutomu Terada
Graduate School of Engineering
Kobe University
Kobe, Japan
Email:tsutomu@eedept.kobe-u.ac.jp

Masahiko Tsukamoto
Graduate School of Engineering
Kobe University
Kobe, Japan
Email:tuka@eedept.kobe-u.ac.jp

***Abstract*—Recently, ubiquitous computing has been getting a great deal of attention. In ubiquitous computing environments, sensor nodes connected to a network are used for collecting sensor data. However, considering the cost to set up the network, it may be more convenient to use data logger that directly sends data to the user compared with the use of a sensor network. In this paper, we propose a ubiquitous data logger that has a function of data management considering sensor characteristics. The proposed ubiquitous data logger reduces the volume of data by selecting an appropriate compression algorithm considering sensor characteristics because the sensor characteristics vary depending on the sensor data.**

***Keywords*-Ubiquitous Data logger; Compression algorithm; Sensor characteristic;**

I. Introduction

Recently, ubiquitous computing has been getting a great deal of attention because of the miniaturization and the technical advancement of computers. Ubiquitous computing is the concept that Mark Weiser proposed[1]. In the ubiquitous computing environment, we can use many small computers that are scattered in environments such as office and home without any conscious of the computers. For example, an agricultural supporting system by temperature, humidity, and illumination sensors can support users by acquisition of action history and its situations. In these systems, we generally use data loggers that are connected to the network when we need sensor data. However, considering the cost to set up the network, it is more convenient for us to use a data logger communicating by ad hoc without any base station. For example, when we conduct climate investigation on farmland, we disseminate data loggers over observed place and make them keep on sensing, and we receive sensor data directly from these data loggers by walking through the communication area of the data loggers. In this case, these data loggers have to keep a large amount of data and it takes long time to transmit the sensor data to the user. The transmission of a large amount of data not only consumes the battery life of the device but also increases the possibility of transmission failure because the user may pass the communication area when it takes too long time for data transmission. Therefore the data logger should manage the sensor data efficiently, for example by compressing the sensor data.

In this paper, we propose a ubiquitous data logger that has a function of data management considering sensor characteristics. It applies an efficient compression algorithm to sensor data considering the sensor characteristics such as the data changing gradually like humidity sensor, the data expressed by ON/OFF like human detection, and the data changing intensely like accelerometer.

We propose the effective compression algorithm especially in sensor data which changes gradually and intensely. We apply the compression algorithm which transposes sensor data to an approximation function to sensor data which changes gradually. To sensor data which changes intensely, we apply the compression algorithm which transposes sensor data to the power spectrum of the Fourier transform.

The remainder of this paper is organized as follows. Section II explains related work. Section III describes the environmental assumptions and the systems of proposal data logger. We then present the results from an evaluation of proposal compression method. Finally, we present our conclusions and work we plan to do in the future.

II. Related Work

Several researches are done on compression of sensor data[2][3][4]. Especially, in Nakajima's research[5], a method of managing sensor data is proposed to reduce battery consumption for sensing. It reduces the sampling frequency when high accuracy is not required for a sensor, which is judged by a feature value of obtained sensor data. In addition, the authors compared several methods for completing the removed data; the method of completing same values as preceding sensor data, the method of linear completing to complete sensor data in the line which was calculated from the two points just before eliminating, window copy method to complete copies of preceding shapes of wave in the case of predicting to repeat same shapes and so on. In Masuda's research, the authors made transfer through the network faster by dealing with parameter such as points of deciding curved aspects as regular line of value and changing them into frequency area in order to increase efficiency when we needed to send and receive massive data, for example design for three-dimensional products. Generally floating point such as control points' coordinates often lose regulations when floating point become lower figure, and he achieved high compression rate by using the compressional method which is similar to JPEG compression. However these methods are not appropriate, for they remove the data which is far from the data flow which look like a noise apparently, while they may have important meanings. Performance of sensor node may also be limited because we use small sensor nodes in the environment of ubiquitous computing. Thus, the existing methods are not practical due to a large amount of calculation.

III. Design of Ubiquitous Data Logger

In our work, we assume the situation that mobile computers and wearable devices use sensor data stored by a data logger scattered over the environment. For example, when people make use of sensor data stored in a portable game machine, the machine can obtain environmental information such as temperature and humidity from data loggers. A game program can change monster appearing in the game according to the environmental information. Although we can realize this game by scattering a lot of loggers over the environment, it is not easy to construct it broadly from the view point of cost. Therefore, we can say that ad-hoc or one-shot direct communication is reasonable for this purpose rather than standard network communication. Based on this consideration, we aim at designing ubiquitous data logger especially focusing on smooth exchange of data.

A. Requirements to System

1) Data Compression Method: Considering capacity and battery consumption of a ubiquitous data logger, it is required to compress data when it obtains sensor data. In some situations, communication time may be unable to be secured enough. For example, when we conduct climate investigation on farmland, we collect sensor data directly from each data loggers, which is scattered over the preserved place and has kept on sensing for long time, by approaching them. In this case, the algorithm which makes sensor data as small as possible is required because data logger have to save data for several weeks or years and complete data transmission while a user passes through the range where communication is made.

2) Data Compression algorithm which consider sensor characteristics: In ubiquitous computing environment, applications are assumed to use various kind of sensors which have variety of characteristics. Therefore, data logger is required to select data compression algorithm according to using sensor characteristics in order to make the compression efficient. Similar to the example mentioned above, in the

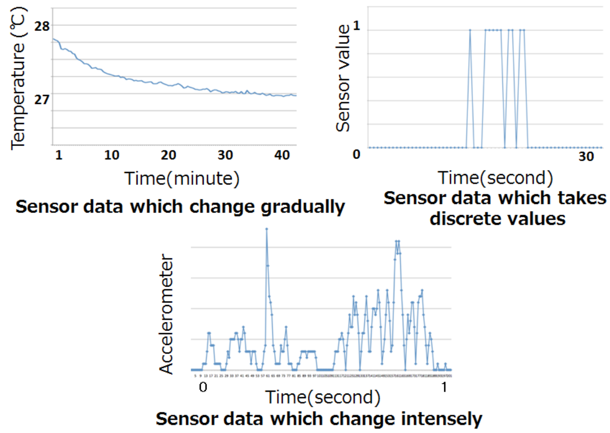

Figure 1. Classification of sensor data

situation where many and unspecified people use sensor data, accuracy and data volume of sensor data are different by applications. For example, people may permit a large error, and others may want to receive sensor data with small volume as possible as the system can. Therefore, data logger is required to change data before transmission so that user's demand can be filled when there is a demand of data conversion, while stored data is usually transmitted as it is. In addition, different from the compression of picture data, problems may be occur in application if noise of sensor data is removed. This is because sensor data has an important meaning in sudden abnormal value in many cases. Therefore, a new algorithm is required to compress sensor data where no every data exceeds the permissible error.

B. Classification of sensor data

The typical sensor data used in an environment of ubiquitous computing is classified as follows:

1) Gradual changing data: As shown in the upper left of Fig.1, sensor data do not change so much within a short time, which is expressed by a loose curve in its graph. This data was obtained when indoor humidity was measured for 30 minutes for every second using moisture sensor $CHS-UGR$ made by TDK Corp. A temperature sensor, an illumination sensor, a smell sensor, etc. are classified in this class.

2) Intensive changing data: As shown in the row of Fig.1, in some sensors, data changes a lot within a short time , that is, it changes intensely. This data was obtained when the corrugated paper to which 3-axis accelerometer $MNA7260Q$ was attached was struck by hand, and its sampling rate is 100 Hz. Since it intensely changes within a short time, in order to measure it correctly, it is necessary to set up a sampling frequency greatly. In addition to this, an oscillating sensor, a sound sensor are classified in this class.

3) Binary data: As shown in the upper right of Fig.1, there are sensors whose data is binary. This data was

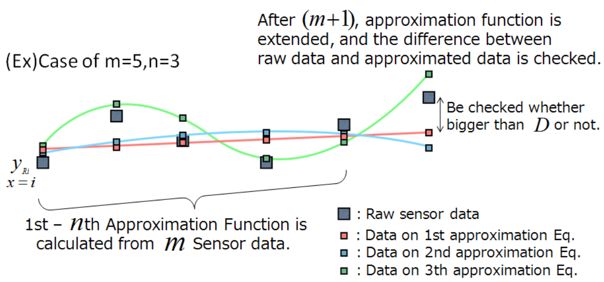

Figure 2. Diagrammatic illustration of proposal method

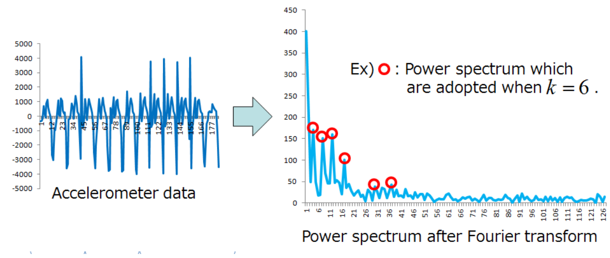

Figure 4. Diagrammatic illustration of proposal method

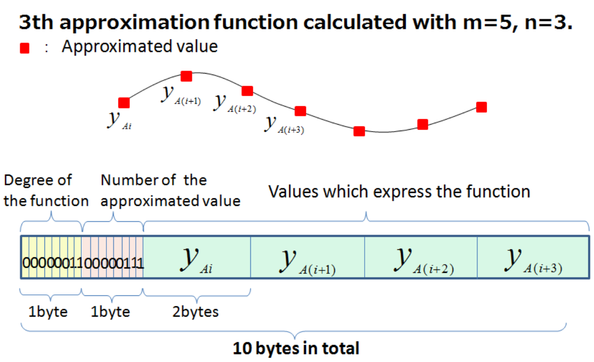

Figure 3. Record format when the third function is adopted

obtained when we installed an infrared distance sensor $GP2D12$ made by $Sharp$, which measures distance from people by using infrared reflection, in front of the door and changed its sensor data into boolean every second by taking a threshold with a comparator. As the characteristic of the sensor classified here, it can be mentioned that measured value is restricted to two values of 0 and 1 and that the sensor keeps on taking the value same for a long time. In addition to this, a motion sensor using an ultrasonic wave, an electrostatic sensor which detects people are classified in this class.

C. Management technique of data

1) Data compression method for gradual changing data: The ubiquitous data logger reduces data volume by transposing sensor data to an approximation function within the permissible error. The ubiquitous data logger compresses the data simultaneously with sensing, and it records only the compression result in it. The schematic view of proposal compression algorithm is shown in Fig.2. In this figure, the parameter n shows the maximum degree of the approximate function to be used, and the parameter m shows the maximum number of the sensor data which can be used for calculating an approximation function. A user may set the parameter to any value. As shown in Fig.2, an approximation function is calculated by using m sensor data. After the $(m+1)$th sensor data, the approximation function is used continuously if the difference between sensor data and the value acquired by extending the approximation function is smaller than the permissible error. If the difference is larger than the permissible error, the parameter of the present approximation function is saved, and a new approximation function is calculated from the sensor data which was bigger than permissible error. We showed the example in which the parameters are set as $n = 3$, $m = 5$.

A format for saving data consists of a degree, a number, and data for expressing the approximation function. An example of saved data when the 3rd approximation function is adopted is shown in Fig.3. In order to express the degree of the adopted approximation function and the number of the approximated values on an approximation function, 1 byte is prepared, respectively. As the data which is needed for restoring an approximation function, the first $n + 1$ sensor data on the adopted approximation function is saved. This is because if n approximate values on the approximation function are known, a $(n + 1)$th approximation function can be expressed correctly. 2 bytes are prepared for each approximated values.

2) Data compression method for intensive changing data: The ubiquitous data logger disassembles sensor data into the power spectrum in a frequency domain by the Fast Fourier Transform, and reduces data volume by deleting a weak power spectrum which has little influence to restoration of sensor data while it saves strong power spectrum. In addition, because accelerometer sensor data are used for an action cognitive algorithm in many cases, after being changed into variance as an amount of the features, the error of variance after compression must not exceed a permissible error. The schematic view of proposal compression algorithm is shown in Fig.4. In this figure, the parameter k shows the number of the power spectrum saved at compression. When 256 sensor data are acquired, proposal data logger changes the sensor data into frequency. At this time, k power spectra counted from the larger one are adopted. Next, the sensor data after compression is restored and a variance is calculated from them. That variance is compared with the variance which is calculated from the raw sensor data. If the difference between that two variance is smaller than the permissible error, the k spectra are saved and compression of the following 256 sensor data is started. If the difference is larger than the permissible error, the number of spectrum adoption is increased to $k = k*2$, and sensor data is restored and the variance is calculated again, and it is compared with the variance of raw data. The process in which the number of spectrum adoption is increased is performed as long as $k <= 128$, and if $k > 128$, compression is not performed but raw data is saved. The Fourier transform function is shown in the formula(1), and the clause which is needed for restoration of sensor data is shown in the formula(2), formula(3), and formula(4). 2 bytes are prepared for the first term and every 4 bytes are prepared for every power spectrum.

$$f(x) = a_0 + \sum_{n=0}^{\infty}(a_n \cos\frac{n\pi}{L}x + b_n \sin\frac{n\pi}{L}x) \quad (1)$$

$$a_0 = \frac{1}{2L}\int_{-L}^{L} f(x)dx \quad (2)$$

$$a_n = \frac{1}{L}\int_{-L}^{L} f(x)\cos\frac{n\pi x}{L}dx \quad (3)$$

$$b_n = \frac{1}{L}\int_{-L}^{L} f(x)\sin\frac{n\pi x}{L}dx \quad (4)$$

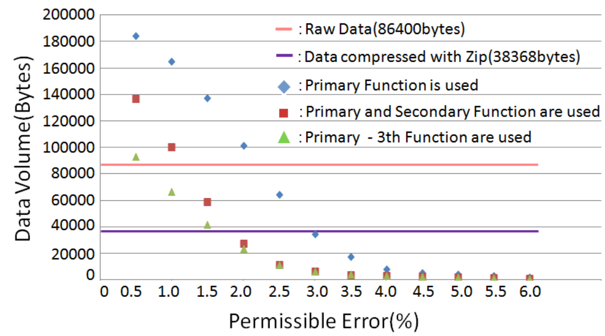

Figure 5. Relation between permissible error and compressibility

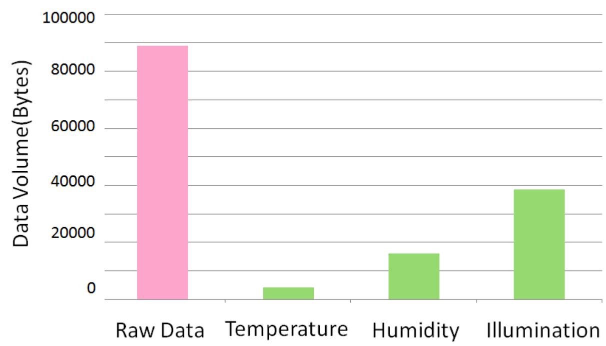

Figure 6. Compression result of a variety of sensor data

IV. Evaluation Experiments

In order to investigate the usefulness of the proposal technique, we conducted three kinds of evaluation experiments. The evaluation experiment was done on simulation with PC by using the sensor data which is actually measured.

A. Evaluation of data compression method for gradual changing data

First, in the compression method for the sensor data which changes gradually, we evaluated from the viewpoint of degree of the approximation function and of a compression ratio. We used the sensor data which is measured for every second during a day by humidity sensor $CHS - UGR$ made by TDK Corp. The evaluation result is shown in the Fig.5. We showed the result of having used only the primary approximation function, the result of having used a primary approximation function and the secondary one, and the result of having used from a primary approximation function to the 3rd one, respectively. We showed 86,400 bytes for raw data of a hole day. And we also showed the data volume obtained when zip compression was applied to raw data in order to compare with the proposal method.

When the permissible error rate is set to 0.5 percent, the compressed data volume was larger than raw data. It may be because of the reason that the data area prepared for saving the parameters of an approximation function became redundant because only a little sensor data is approximated with one approximation function when the permissible error is set small. The higher the degree of approximation function which can be used became, the better the compression ratio became because the compression algorithm can respond to the change of sensor data flexibly. We can reduce the data volume when 1 percent or more error was permitted for error when from a primary approximation function to the 3rd one are used. By the way, the error on the specification of the sensor used for evaluation is 3 percent. So data volume can certainly be reduced by using the proposal method in a practical range.

Next, we applied the proposal method to the sensor data which has been recorded for a long time. We used temperature sensor data, humidity sensor data, and illuminometer data, which are measured for every 5 minutes during a whole year. We permitted 1 degree Celsius, 1 percent, and 10 Lux as error for each sensor. The evaluation result is shown in the Fig.6.

In the Fig.6, the raw data volume was 89,026 bytes. We can compress effectively not only moisture sensor data but other sensor data. The reason why data volume became large in order of temperature sensor, moisture sensor, and

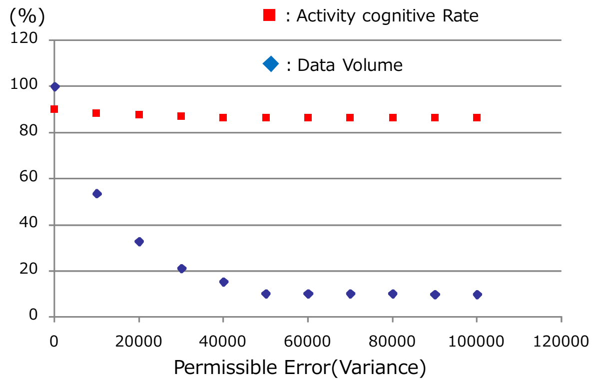

Figure 7. Relation between permissible error and compressibility, and activity recognition rate

illumination sensor is that compression ratio depends on the amount of sensor data which changes in a day. We can prove that the proposal method is useful also for the data which has been recorded for a long period.

B. Evaluation of data compression method for intensive changing data

Next, we evaluated our compression method for sensor data which intensely change from the viewpoint of compression ratio and accuracy of action recognition. We used the $K-NN$ technique widely used for action cognition. We used accelerometer data which is acquired when six actions such as walking, running, rinding on a bicycle, climbing stairs, standing, and sitting were done for ten seconds per action by each subject who attached 3 axis accelerometer whose sampling frequency 50 Hz to his waist. We used accelerometer data for 180 seconds in which the above-mentioned actions come out at random every 20 seconds. The evaluation result is shown in the Fig.7. Here, we show the data volume when the permissible error is changed from 0 to 100,000. We also showed the cognitive rate when the data which were compressed with each permissible error were applied to action cognition.

As a result, we could reduce data volume greatly according to the each permissible error. We could not reduce the data volume after the permissible error became bigger than 50,000. It is considered to be the cause that the number of spectrum adoption keeps the minimum k after the permissible error became 50,000. In addition, we could keep the action cognitive rate high even after the permissible error became large. The reason for this is that we could set the minimum k appropriately.

V. Conclusion and Future Work

In this paper, we explained the usefulness of a data logger not belonging to a network in the environment of ubiquitous computing. Further, we proposed compression algorithm considering sensor characteristics because data compression is required in such an environment.

As a future work, we must improve the compress algorithm so that sensor data can be compressed more efficiently while an error is kept small according to the situations in which sensors are used. More detailed research is especially required for the compression algorithm for accelerometer because the choice of a power spectrum greatly affects recognition accuracy. Further, we must design an algorithm which compresses data again when a user request. We also consider that it is necessary to evaluate the proposal method in real situations not by simulation.

Acknowledgment

This research was supported in part by a Grant in aid for PRESTO from Japan Science and Technology Agency and Scientific Research(A)(23240010)(20240009) from Japanese Ministry of Education, Culture, Sports, Science and Technology.

References

[1] Mark. Weiser: *The Computer for the Twenty-first Century*, Scientific American, Vol. 265, No. 3, pp. 94–104, 1991.

[2] Mo. Chen, Mark L. Fowler: *Data compression trade-offs in sensor networks*, SPIE, 5561, 96, 2004.

[3] Ranganathan. Vidhyapriya, Ponnusamy. Vanathi: *Energy Efficient Data Compression in Wireless Sensor Networks*, The International Arab Journal of Information Technology, Vol. 6, No. 3, 2009.

[4] Sorabh. Gandhi, Suman. Nath, Subhash. Suri, Jie. Liu: *Compressing Multi Sensor Data by Grouping and Amplitude Scaling*, ACM, SIGMOD, 2009.

[5] Yuki Nakajima, Kazuya Murao, Tsutomu Terada, Masahiko Tsukamoto: *A Method for Energy Saving on Context-aware System by Sampling Control and Data Complement*, ISWC 2010, pp. 129-132, 2010. 2009, pp.1693-1700, 2009.