Regular Paper

Automatic Measurement of Lead Time of Repetitive Assembly Work in a Factory Using a Wearable Sensor

Yasuo Namioka[1,a)] Daisuke Nakai[2,b)] Kazuya Ohara[2,c)] Takuya Maekawa[2,d)]

Received: December 23, 2016, Accepted: July 4, 2017

Abstract: In this paper, we attempt to estimate lead time (duration) of each period of an operation process by a factory worker using a wrist-worn accelerometer. In a factory line production system, a worker repetitively performs predefined operation processes, and the lead time greatly affects productivity of the line production system. Our proposed method automatically finds a frequent sensor data segment as a "motif" that occurs once in each operation period using only prior knowledge about predefined standard lead time of the operation process, and uses the occurrence intervals of the motif to estimate the lead time.

Keywords: factory worker, activity recognition, unsupervised method, wearable sensors

1. Introduction

1.1 Background

Recognition of daily activity using sensor data obtained from body-worn smart devices is currently one of the most active topics in the ubiquitous and wearable computing research communities [1], [2], [3], [4], [5]. Activity recognition techniques are expected to be applied to industrial applications such as work analysis of factory workers [6], [7] as well as daily applications such as healthcare, elderly care, and lifelogging [8], [9], [10]. This paper also focuses on factory worker assembly tasks and attempts to analyze the work by using a wrist-worn acceleration sensor.

Many factories have applied a line production system where each product passes through the same sequence of operation processes. Assembly work by factory workers still constitutes the core of the production system and improvement of assembly work is one of the most important tasks for increasing productivity [7], [11]. In the line production system, a worker repetitively performs predefined operation processes, and each operation process consists of a sequence of operations such as setting a board on a work bench and screwing parts onto the board. The duration of one period of a worker's operation process is referred to as "lead time," and the lead time directly affects productivity of the line production system. Therefore, management of the periods of operation processes is necessary for improving productivity of the line production system.

The first important task in managing operation process periods is to measure the lead time of each period. With the measured durations, a line manager can easily know which worker is the bottleneck of the line. Also, from the transition in worker lead times, the line manager can estimate the degree of worker fatigue and the extent to which the worker is habituated to the work, and thus can estimate the potential bottleneck of the line. When a line manager finds that the lead time of an operation process is much longer or shorter than usual lead times, the line manager should determine the reasons, e.g., lack of a certain operation or the extension or shortening of the duration of a certain operation. Therefore, the second important task is the analysis of the problematic operation process. This paper mainly focuses on the first task.

Although it is possible to manage and maintain the entire assembly line by having a line manager manually measure the lead time of each worker's operation process, this imposes heavy burdens on the manager because the line consists of many workers. Although working devices with timing measurement functions do exist, such devices are not applied to all workers and all processes. Analyzing a problematic operation process by using the line manager is far more difficult.

1.2 Research Goal

Therefore, an easy and unobtrusive way to automatically measure the lead time of each operation process and analyze the operation process is required. One possible way of measuring the lead time is to use a wearable sensor and machine learning techniques. By detecting a sensor data segment that appears in each operation period, we can measure the lead time based on the frequency (occurrence interval) of the segment. For example, when a screwing action occurs at the start of each operation period, we can measure the lead time of each period by learning and detecting sensor data segments of the action. (When the action occurs several times in each operation period, this method does not work well.) Supervised learning approaches can also be applied to the operation process analysis. By learning a labeled sequence of operations in

[1] Corporate Manufacturing Engineering Center, Toshiba Corporation, Yokohama, Kanagawa 235–0017, Japan
[2] Graduate School of Information Science and Technology, Osaka University, Suita, Osaka 565–0871, Japan
[a)] yasuo.namioka@toshiba.co.jp
[b)] daisuke.nakai@ist.osaka-u.ac.jp
[c)] ohara.kazuya@ist.osaka-u.ac.jp
[d)] maekawa@ist.osaka-u.ac.jp

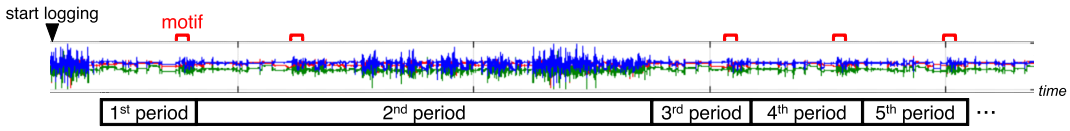


Fig. 1 Example of three-axis acceleration data obtained from a factory worker in an assembly line who wore a smart watch on her right wrist. Red, green, and blue lines show x-, y-, and z-axes acceleration data, respectively. The sensor sampling rate is about 60 Hz.

advance, we can find a missing operation or an operation whose duration is longer or shorter than usual. However, these supervised approaches requiring training data collection have the following problems. (1) Since an operation process depends on each worker, collecting training data from each worker in advance imposes substantial costs. (2) Operation processes can frequently change (e.g., weekly or monthly) due to frequent revisions on the production system.

In this study, we attempt to investigate the feasibility of unsupervised understanding of operation processes in line production systems. Specifically, this study focuses on measuring lead times, and we propose an unsupervised measurement method for estimating the lead time of each operation period using a wearable sensor. The goal of this method is to find a start time (and an end time) of each period in unsupervised manner. **Figure 1** shows an example of an acceleration data sequence obtained from a worker in a real factory. The worker first turned on a sensor data logger and then started her work. Because the second period included additional operations, its lead time is longer than the lead times of the other periods. By analyzing the acceleration sequence, our method detects a start time and end time of each period as shown in Fig. 1 in an unsupervised manner.

While the main focus of this paper is estimating the lead times, we should briefly mention unsupervised analysis of each period based on outputs of our method to clarify the usefulness of our method. Because we assume that training data are unavailable, we segment the time-series data of the period into meaningful states present in the time-series data in the operation period solely from the data. With the segmentation result, the line manager can easily understand the structure of the operation period in detail. When the lead time of the period is longer than the lead times of the other periods, for example, the line manager looks at a segmentation result of the period and can find the duration of which state in the period is longer than usual. **Figure 2** shows an example of a segmentation (clustering) result where a color shows an associated cluster (hidden state) of a data point. Our segmentation method is a Bayesian nonparametric version of the hidden Markov model (HMM) called the hierarchical Dirichlet process HMM (HDP-HMM) [12]. For more detail about the unsupervised analysis, refer to [13]. In the example, the lead time of the $(n+1)$st period is longer than that of the nth period. As shown in the segmentation result, the duration of the blue colored region of the $(n+1)$st period indicated by a curly bracket seems to be longer than that of the nth period (63 seconds vs 55 seconds). From the segmentation result, the line manager can estimate (1) whether or not necessary operations are missing, (2) whether or not unnecessary operations are included, (3) whether or not the duration of an operation is longer/shorter than usual, and (4) whether or not the order of operations is correct. Based on the estimation, the line

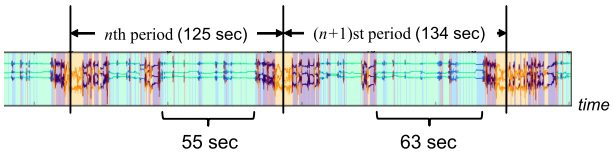


Fig. 2 Example of segmentation result of three-axis acceleration data obtained from a factory worker.

manager checks videos recorded by cameras that overlook the assembly line, and confirms whether or not the corresponding operations are correct. As discussed above, the lead time estimation and segmentation results are clues to identify outlying operations.

In the introduction section, we introduced an unsupervised segmentation result based on HDP-HMM. Here we introduce other unsupervised segmentation methods for time-series data. In AutoPlait [14], the authors used a multi-level chain model to segment the time-series data. Unlike HDP-HMM, they estimate its model parameters based on the combination of model description cost and coding cost of whole time-series. Similarly, in TrailMarker [15], the authors segment trajectory data as well as clustering the trajectories based on their similarities by minimizing model description cost and coding cost of describing data. In [16], pairwise distances among all subsequences in time-series data are efficiently computed. The authors segment time-series data based on the fact that the distance between subsequences in the same segment is small. Unlike these methods, HDP-HMM estimates model parameters of HMMs based on the Bayes estimation framework.

1.3 Research Methodology

For estimating a start time of each operation period, our proposed method requires only information about a predefined standard lead time of the operation process. The idea behind our proposed method is simple. In a line production system, a worker repetitively performs their operation process. Therefore, sensor data obtained from a wearable sensor attached to the worker also have repetitive patterns. Our proposed method finds a frequent sensor data segment as a "motif" that occurs once in each operation period. Red time windows in Fig. 1 indicate examples of occurrences of a motif. Based on the occurrences of the motif, we estimate the actual start time of each period. In this method, an operation process model is prepared in advance based on knowledge about the predefined standard lead time of the operation process, and our method finds a motif that appears in the sensor data sequence in accordance with the operation process model. When the standard lead time is two minutes, for example, we find a motif that occurs about every two minutes and then employ the motif to track operation periods and estimate the start time of each period.

Note that the following factors make this task difficult.

1) The lead time of one period of an operation process of actual factory work fluctuates. Therefore, simple existing methods for frequency analysis do not work well for estimating the lead time, e.g., using the autocorrelation of sensor data or calculating dominant frequencies by analyzing the entire sensor data sequence using fast Fourier transformation. Also, Fourier analysis cannot provide the lead time of *each* period.
2) Operations in a worker's operation process sometimes change depending on the situation. For example, if a worker replaces a part on a board only when the part is broken, the lead time of the operation process depends on a test result of the part. Also, the worker assembly tasks sometimes consist of several operation processes. For example, a worker performs operation process A and operation process B; operation process A corresponds to assembling parts and operation process B corresponds to boxing several assembled products. Therefore, operation process A is iterated several times and then operation process B is performed to box products assembled in preceding periods of operation process A. In such case, an operation process has two possible cases of standard lead time.
3) It is difficult to detect when the worker started operation processes. We assume that the worker runs a data logger on a wrist-worn sensor device by herself before she starts her work. So, the time when she starts sensor data collection does not strictly correspond to the time when she starts her work.

To cope with the first and second problems, we deal with such fluctuations and variations of the lead time in an operation process by employing a particle filter [17], which is usually used to estimate the states of non-linear systems. A particle filter permits us to robustly track a motif that appears non-linearly. To cope with the third problem, we utilize our idea that sensor data unrelated to operation processes, e.g., just after the logging start, are dissimilar to sensor data corresponding to operation processes. Based on this idea, we detect a sensor data segment collected before the first operation process and then find a start time of the first operation period. For the second or later periods, we find their start times using a sensor data segment corresponding to the start of the first operation period based on their similarities.

1.4 Research Contributions

The research contributions of this paper are as follows: (1) To the best of our knowledge, this is the first study that proposes an unsupervised method for measuring the lead times of operation periods and estimating start times of the operation periods of a factory worker. (2) To deal with fluctuating and varying operation periods, we design a robust motif tracking method based on particle filtering. (3) To reduce the computation cost of the particle filter based tracker, we quickly identify candidates of motifs by symbolizing time-series acceleration data. (4) We evaluate our method using sensor data obtained in real factories.

2. Related Work

Due to the recent growing interest in smart manufacturing and Industry 4.0 [18], [19], studies on recognizing and supporting factory work using sensor technologies [20], [21] have been attracting attention.

We introduce studies on monitoring and analyzing factory work using wearable sensors. Koskimäki et al. [22] obtain acceleration and gyro sensor data from a wrist-worn inertial sensor device and analyze operation processes in a line production system to ensure that all necessary operations are performed. The study recognizes such activities as hammering and screwing by using kNN search. Ward et al. [23] obtain acceleration and sound sensor data from a wrist-worn device to recognize woodworking activities by using hidden Markov models (HMMs) and a linear discriminative classifier. Stiefmeier et al. [24] focus on assembly work of automobiles and use inertial sensors attached to several body parts such as the upper and lower arms to classify a sensor data segment by computing the distance between the segment and sensor data templates prepared in advance using discretized sensor data. They attempt to classify activities such as opening the engine hood and opening the trunk. Stiefmeier et al. [25] also focus on work of bicycle repair and use motion sensors and ultrasonic hand tracking to recognize maintenance activities using HMMs. All the above methods for analyzing factory work rely on supervised machine learning approaches and require training data collection.

Here we introduce studies on unsupervised activity recognition. Huynh et al. [26] use topic models to cluster activity data in an unsupervised manner. Also, Khan et al. [27] symbolize acceleration data to discover the structure of surgical activities in an unsupervised manner.

3. Assumed Environment

3.1 Sensor Setting

We assume that a worker wears body-worn inertial sensors such as accelerometers. In our experiment, workers wore a smart watch with a three-axis accelerometer on their right wrists. The sensor sampling rate is about 60 Hz. We also assume that several cameras that overlook an assembly line are installed. Because the cost of the cameras is high and the data size of the video recordings is huge, small numbers of cameras are installed and each camera captures multiple workers. We assume that, when a line manager finds outlying operation processes from results of our methods, the manager checks video recordings of these outliers.

3.2 Work Instructions

Work instructions are prepared in advance for each worker assembly task. The work instructions specify a standard lead time of the operation process and a flow of operations included in the operation process, e.g., (1) placing a board on a work bench, (2) checking a mode of a tester, and (3) changing a mode of a signal monitor.

To estimate the lead time of an operation period, we apply the standard lead time included in the work instructions. As mentioned in the introductory section, the work instructions can have multiple standard lead times. For example, an assembly work consists of two kinds of operation processes: operation process A corresponding to assembling parts and operation process B corresponding to boxing several assembled products. Therefore, operation process A, which is the main operation process in the work, is iterated several times and then operation process B, which is

sub-operation process, is performed to box products assembled in preceding periods of operation process A. For each kind of operation process, the standard lead time is defined, e.g., 2 minutes for operation process A and 5 minutes for operation process B.

Here, our method estimates the lead time of an operation period by tracking a motif, which occurs once in each operation period. Therefore, when there are no motifs that commonly occur in all the kinds of operation processes, our method cannot measure the lead times of the processes. In the above example, because operation processes A and B are completely different, it is difficult to find a motif that occurs in both operation processes A and B. To cope with this problem, we introduce a new kind of operation process C, which consists of operation process A followed by operation process B, instead of operation process B. For example, when we observe a sequence of operation processes "AAAABAABAAAAB," we regard the sequence as "AAACACAAAC," i.e., "AB" is replaced by "C." The standard lead time of operation process C will be the sum of the standard lead times of operation processes A and B, i.e., 7 minutes. By doing so, we can find a motif that occurs in the first two minutes of operation process C. As discussed above, when sensor data (operation flows) of different kinds of operation processes are completely different, we should modify standard lead times described in the work instructions. This is a limitation of our method. However, in many cases, we can easily modify the standard lead times by just concatenating a main operation process and a sub operation process because a sub operation process is usually performed between iterations of a main operation process.

4. Detecting the Operation Period

The proposed method first finds a motif that repeatedly occurs in accordance with an operation process model. The method then tracks the found motif, which occurs once in each operation period. Based on the occurrences of the motif, we estimate the actual start time of each period. **Figure 3** shows an overview of finding the best motif. Also, Fig. 3 shows an overview of finding the start times of operation periods. Before explaining the methods, we introduce an operation process model.

4.1 Operation Process Model

Our method employs an operation process model that defines the standard lead time of the operation process of interest. In this study, a Gaussian distribution is used to represent the standard lead time of the process. Since periods of operation processes sometimes have different standard lead times, an operation process model is represented as a mixture of Gaussian distributions, each of whose means corresponds to each case of standard lead time. The probability with which the lead time of a period is t is described by

$$p(t|M) = \sum_{i=1}^{N} \pi_i \mathcal{N}(t, \mu_i, \sigma_i),$$

where N is the number of mixtures, π_i is the mixture weight of the ith multivariate Gaussian distribution of the GMM (Gaussian Mixture Model; $\pi_i = 1/N$ in our implementation), and μ_i and σ_i are the mean and variance of the Gaussian distribution, respectively. Also, the GMM parameters are collectively represented by M. For example, when an operation process has two possible cases of standard lead time, e.g., 2 and 5 minutes, its operation process model consists of two Gaussian distributions whose means are 2 and 5 minutes. By comparing the occurrences (intervals) of a motif and an operation process model, we judge whether or not the motif occurs in accordance with the model.

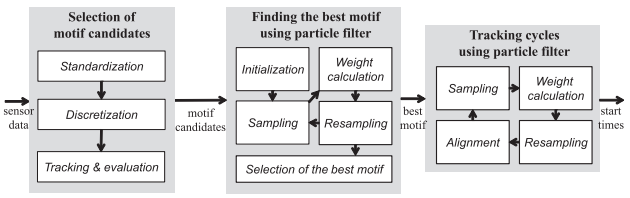


Fig. 3 Overview of finding the best motif from sensor data, tracking operation periods, and estimating their start times using the best motif.

4.2 Finding the Best Motif

4.2.1 Overview

The left portion of Fig. 3 shows an overview of finding the best motif. We assume that an operation process model is created from the predefined standard lead time(s) of a worker's operation process of interest in advance. Our method compares the model with the first t_{ms} minutes of sensor data from the beginning of the sensor data collection, and finds a motif that is suitable for measuring lead times of the operation processes by using a particle filter. In the initialize phase of the particle filtering, we randomly extract motifs (sensor data segments) with random durations from the first t_{init} minutes of sensor data from the beginning of sensor data collection, and we regard the motifs as particles. After that we successively track the occurrences of the motif (particle), and then calculate the likelihood (score) of the operation process model for the occurrence intervals. We compute the score for each randomly generated motif and the motif with the largest score is used to find operation periods and estimate start times of the periods. That is, after t_{ms}, we track the motif with the largest score using the particle filter.

To track randomly generated motifs using the particle filtering, we calculate the similarity between each motif and each sliding window segment extracted from the entire sensor data. Because sensor data of a worker's operation may vary in time or speed, we employ DTW to calculate the similarity (distance) between a motif and sensor data segment. DTW is designed to calculate the similarity between two temporal sequences with different lengths. Because our method should compute the similarity between each motif and each data segment extracted from the entire sensor data, it takes a long time to find a suitable motif from the sensor data. Therefore, before running the particle filter based on DTW, we select several motif candidates with small computation costs by using the first t_{ms} minutes of sensor data from the beginning of sensor data collection. In this study, we discretize (symbolize) the sensor data and compute the similarity between the discretized motif and sensor data segment by using the Hamming distance, which permits us to substantially reduce computation costs as regards similarity calculation. Therefore, we first use the discretized data to find several motif candidates that occur

in accordance with the operation process model. After that, we track only the selected motif candidates again in detail based on DTW and find the best motif.

4.2.2 Selection of Motif Candidates

We first select a few motif candidates (k candidates) that occur in accordance with an operation process model from many randomly generated motifs by using t_{ms} minutes of sensor data. Here, t_{ms} determines the duration of sensor data that are used to find the best motif. To select motif candidates with small computation costs, we standardize and then discretize (symbolize) t_{ms} minutes of time-series sensor data according to Ref. [28]. We briefly explain the procedure proposed in Ref. [28]. After the standardization, the time-series is represented as piece-wise approximations where the time-series is divided into equal-sized frames and the mean value of data within each frame becomes a representative value of the frame as shown in **Fig. 4**. Therefore, we reduce the length of the time-series to the number of frames.

We then convert the reduced time-series into a series of symbols such as $cbaaabb\cdots$. We set several breakpoints and map each value of a frame into a symbol. For example, when an area between breakpoints β_1 and β_2 corresponds to b, and a value of a frame falls into the area, the value is mapped into b. From the discretized time-series, we randomly extract motifs with random lengths from the first t_{init} minutes of the sensor data, and then track each motif by using the particle filter. The way of tracking a motif is almost identical to that described in Section 4.2.3, which tracks a motif in detail using DTW. Note that, since this process deals with the discretized time-series, we employ the Hamming distance instead of the DTW distance. Then, each tracked motif is evaluated as to whether or not it occurs in accordance with the operation process model, and we select the top-k motifs according to the evaluated scores. After that, we track the selected top-k motifs in detail by using original sensor data based on DTW and find the best motif.

4.2.3 Tracking Motifs Using a Particle Filter

We achieve motif tracking that is robust against fluctuations and variations in lead times by using the particle filter. Then we find the best motif that occurs in accordance with the operation process model. The particle filter estimates the states of a nonlinear system by iterating a three-step process: sampling, weight calculation, and resampling. Our method tracks a motif according to the procedures of the particle filter as follows.

Initialization: In the initialization process, we randomly extract n_{init} motifs with random lengths from the first t_{init} minutes of the sensor data. That is, t_{init} determines the duration of sensor data that are used to randomly extract motif candidates. We assume an extracted motif as a particle and a timestamp of the first data sample of the motif as the time when the motif first occurred. Then we track the subsequent occurrences of the particle (motif). Note that this random extraction is executed only when we select top-k motif candidates using discretized sensor data. By using original (non-discretized) sensor data of the top-k selected motifs, we track the motifs again in detail based on DTW.

Sampling: Based on the predefined operation process model, we randomly sample particles. When we assume that the time of the ith occurrence of the nth particle x_n is $t(x_n, i)$, the time of the i+1th occurrence of x_n (i.e., $t(x_n, i+1, j)$) is determined according to the operation process model as follows:

$$t(x_n, i+1, j) = t(x_n, i) + \Delta t,$$

where Δt is an estimated interval of the occurrence of x_n that is randomly sampled from the operation process model, i.e., $p(t|M)$, and j shows the jth particle generated from x_n according to the operation process model. We generate n_s particles from x_n as the estimated $i+1$th occurrences of x_n, i.e., prior estimations.

Weight calculation: We calculate a score of each particle that was sampled according to the operation process model as the weight of the particle. Specifically, we compare the prior estimated time of the $i+1$th occurrences of x_n, i.e., $t(x_n, i+1, j)$, with actual sensor data, and evaluate the estimation. To achieve this, we first calculate the similarity between the motif (x_n) and each sliding window segment extracted from the sensor data. The similarity value is computed by $c^{DTW(X,Y)}$ ($c < 1$), where $DTW(X, Y)$ shows the DTW distance between X and Y. (Since we use a three-axis accelerometer, we use the average distance of the three axes.)

Because the similarity value is computed for each sliding window in our method, we can obtain a time series of the similarity values with the motif as shown in $g_s(t)$ of **Fig. 5**. Note that we compute the time-series $g_s(t)$ between $t(x_n, i) - \sigma$ and $t(x_n, i+1, j) + \sigma$. When the ith and $i+1$th occurrences of x_n are actually at $t(x_n, i)$ and $t(x_n, i+1, j)$, respectively, the similarity values around $t(x_n, i)$ and $t(x_n, i+1, j)$ become large as shown in $g_s(t)$ of Fig. 5. Ideally, only the similarity values around $t(x_n, i)$ and $t(x_n, i+1, j)$ are large and the other similarity values are small when the motif is suitable for measuring the lead times of the operation process. (A motif that occurs only once during an operation period is suitable for measuring the lead time. In other words, a motif that occurs many times in each operation period is unsuitable.) To evaluate whether or not the motif occurs in accordance with the operation process model, we define a function $f_e(t, j)$ consisting of a mixture of a Gaussian function whose center of the peak is $t(x_n, i)$ and whose peak center is $t(x_n, i+1, j)$ as shown in **Fig. 6** and it is represented as follows:

$$f_e(t, j) = \exp\left\{-\frac{(t - t(x_n, i))^2}{2\sigma^2}\right\} + \exp\left\{-\frac{(t - t(x_n, i+1, j))^2}{2\sigma^2}\right\}.$$

We compute the mutual correlation between $f_e(t, j)$ and $g_s(t)$ to evaluate whether or not the motif occurs in accordance with the operation process model, i.e., whether or not the motif occurs only at $t(x_n, i)$ and $t(x_n, i+1, j)$ by using

$$r = \frac{\sum_t (f_e(t, j) - \overline{f_e})(g_s(t) - \overline{g_s})}{\sqrt{\sum_t (f_e(t, j) - \overline{f_e})^2}\sqrt{\sum_t (g_s(t) - \overline{g_s})^2}},$$

where $\overline{f_e}$ and $\overline{g_s}$ are the means of $f_e(t, j)$ and $g_s(t)$, respectively. For example, when the estimated $i+1$th occurrence of the motif $t(x_n, i+1, j)$ is close to the actual occurrence of the motif as shown in the left and bottom portions of Fig. 6, the second peak of $f_e(t, j)$

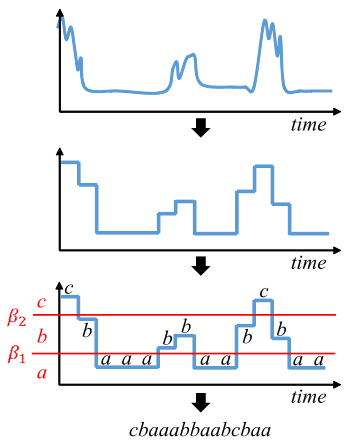


Fig. 4 Discretizing time-series acceleration data.

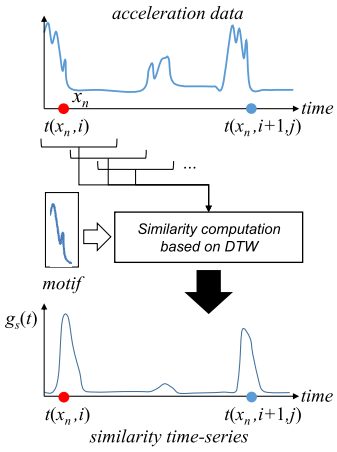


Fig. 5 Computing similarity of time series comparing acceleration data and motif.

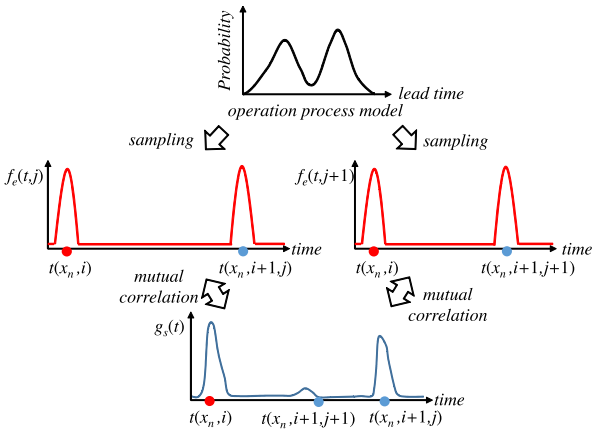


Fig. 6 Weight calculation of particle using mutual correlation.

and that of $g_s(t)$ overlap and thus the computed r value becomes large. In contrast, as shown in the right and bottom portions of Fig. 6, when $t(x_n, i+1, j+1)$ is not close to the actual occurrence of the motif, the computed r value becomes small. We assume the computed r value for $f_e(t, j)$ and $g_s(t)$ as the weight of the particle.

Resampling: We resample the sampled particles according to their computed weights. In the sampling process, we sampled n_s particles from one particle x_n. In this study, we resample only one particle from the n_s particles according to their weights. That is, the timestamp associated with the resampled particle corresponds to the posterior estimation of the $i+1$th occurrence of x_n, i.e., $t(x_n, i+1)$.

4.2.4 Selection of the Best Motif

By iterating the above procedures until time t_{ms}, we can track the occurrences of each motif randomly generated in the initialization phase (or selected by using the discretized sensor data). Finally, we determine the best motif (or top-k motifs) suitable for measuring lead times of the operation process of interest from the motifs in a similar way to the above weight calculation. Specifically, we prepare a function $f_b(t)$ similar to $f_e(t)$ used in the weight calculation process and evaluate a score (mutual correlation r) of each motif by using the function. The function $f_b(t)$ also consists of Gaussian functions where each of the Gaussian functions corresponds to the occurrence of the motif estimated by the above particle filter. For example, when the motif occurs n times, the function is a mixture of n Gaussian functions whose centers correspond to the times of the n occurrences. We also prepare time-series of the similarity values comparing the motif with sliding windows extracted from sensor data between time 0 and t_{ms}. We then compute the mutual correlation r between the time-series of the similarity values and $f_b(t)$. The computed mutual correlation value becomes a score of the motif. We employ the motif with the highest score to track the subsequent occurrences of the motif after t_{ms} by using the above-mentioned particle filter.

5. Tracking Operation Periods

5.1 Overview

Figure 3 also shows an overview of tracking each operation period using the best motif. After t_{ms}, we track the motif using the particle filter in almost the same way as the above method. Based on the found occurrence of the motif, we find the start time of the corresponding operating period.

5.2 Tracking with Particle Filter

The procedure is almost the same as that in the above method. Note that, in real sensor data, data collected during a short break or stoppage of the line caused by a sudden accident can be included. (For simplicity, we assume that such accidental events do not occur before t_{ms}.) To cope with the problem, when the maximum similarity value computed in the weight calculation phase is smaller than a threshold, i.e., no occurrences of the motif can be found, so we sample particles according to a uniform distribution.

5.3 Finding Start Time

After finding the occurrences of the motif, we find the start time based on our idea that sensor data collected before the start of working are dissimilar to sensor data collected during working. Therefore, we first find the start time of the first operation period based on the idea. We explain the procedures using **Fig. 7**. We first extract a sensor data segment between time 0 and time $t(x_b, 1)$, i.e., $s(0, t(x_b, 1))$, and a segment between $t(x_b, 1)$ and $t(x_b, 2)$, i.e., $s(t(x_b, 1), t(x_b, 2))$. Here, x_b shows the best motif. We then reverse the order of data points in each segment and obtain reversed segments $s(t(x_b, 1), 0)$ and $s(t(x_b, 2), t(x_b, 1))$.

Assume that the actual start time of the first period is s_1 and the actual start time of the second period is s_2. Here we can say that $0 \le s_1 \le t(x_b, 1)$ and $t(x_b, 1) \le s_2 \le t(x_b, 2)$ and, $s(t(x_b, 1), s_1)$ and $s(t(x_b, 2), s_2)$ are similar to each other. In contrast, $s(s_1, 0)$ and $s(s_2, t(x_b, 1))$ are not similar to each other because $s(s_1, 0)$ corresponds to sensor data collected before working. We compare $X : s(t(x_b, 1), 0)$ and $Y : s(t(x_b, 2), t(x_b, 1))$, and find $\hat{s_1}$ and $\hat{s_2}$ that maximize the similarity between $s(t(x_b, 1), \hat{s_1})$ and $s(t(x_b, 2), \hat{s_2})$.

We compare $s(t(x_b, 1), 0)$ and $s(t(x_b, 2), t(x_b, 1))$ to estimate $\hat{s_1}$ and $\hat{s_2}$ based on DTW. In the standard DTW algorithm, the cumulative distance $g(i, j)$ is computed at each cell. (DTW algorithm computes a cost matrix to find the optimal alignment between two time series. For more details about the DTW algorithm, see

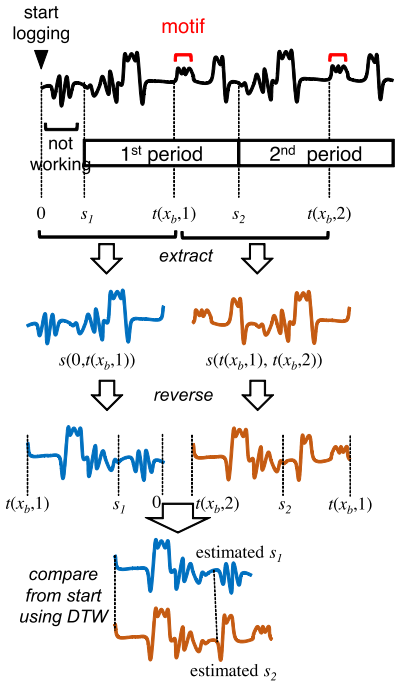


Fig. 7 Finding start time of the 1st period.

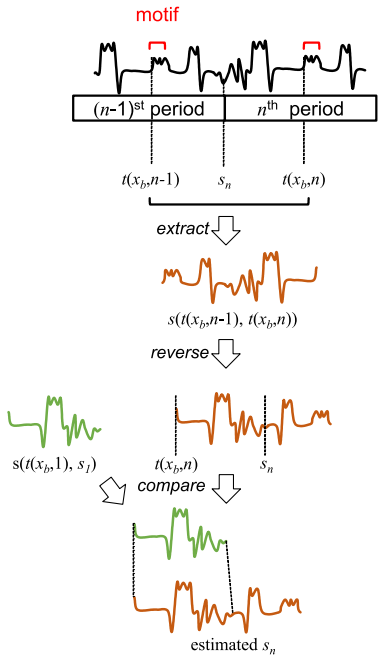


Fig. 8 Finding start time of the nth period.

[29], [30].) After completing the computation at all the cells, we compute the normalized cumulative distance for each cell by

$$g'(i, j) = g(i, j)/(i + j) - \ln(i + j),$$

where the second term is a penalty on small i and j. With the normalized cumulative distances, we can find similar sub-segments $s(t(x_b, 1), \hat{s_1})$ and $s(t(x_b, 2), \hat{s_2})$ regardless of the lengths of the sub-segments. We find a cell with the smallest $g'(i, j)$ and the time corresponding to i becomes $\hat{s_1}$ and the time corresponding to j becomes $\hat{s_2}$.

Because $\hat{s_1}$ is estimated, we can obtain a sensor data segment from the beginning of the first period to the occurrence of the best motif in the first period, i.e., $s(\hat{s_1}, t(x_b, 1))$, and we then reverse the segment. Using the reversed segment, we can estimate the start time of the nth period from the occurrence of the motif in the period by using DTW as shown in **Fig. 8**. We obtain $s(t(x_b, n-1), t(x_b, n))$ and reverse the segment. By comparing the two reversed segments, i.e., $X : s(t(x_b, n), t(x_b, n-1))$ and $Y : s(t(x_b, 1), \hat{s_1})$, we find $\hat{s_n}$ based on DTW ($t(x_b, n-1) \leq \hat{s_n} \leq t(x_b, n)$). To achieve this, we introduce edges with zero distances into the DTW computation as shown in **Fig. 9**. The edges permit us to go to the (N^{th}, M^{th}) cell without comparing whole elements in $s(t(x_b, n), t(x_b, n-1))$ with $s(t(x_b, 1), \hat{s_1})$. Therefore, we can find sub-segment $s(t(x_b, n), \hat{s_n})$ from $s(t(x_b, n), t(x_b, n-1))$ that minimizes the DTW distance between $s(t(x_b, n), \hat{s_n})$ and $s(t(x_b, 1), \hat{s_1})$, and the found $\hat{s_n}$ is the estimated start time of the nth period[*1]. The lead time of the $n-1$st period is computed by $|\hat{s_n} - \hat{s}_{n-1}|$.

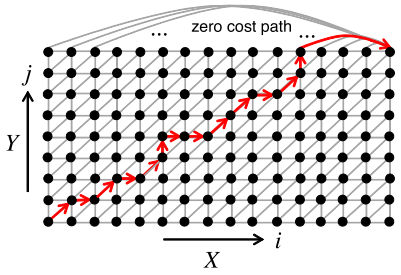


Fig. 9 Introducing zero cost paths to flexibly find the end of X.

6. Evaluation

6.1 Data Set

The aim of our research project is to optimize assembly work by factory workers with small burdens. Toshiba corporation operates many factories that manufacture electrical appliances and devices such as CCD cameras and personal computers, so optimizing the assembly tasks there is important. To investigate the feasibility of a system for automatically measuring lead time and visualizing rough structure of an operation process, we collected sensor data from eight different workers (workers A - H) in the factories. We used Sony SmartWatch3 SWR50 attached to their right wrists to observe their daily tasks. The sensor sampling rate is about 60 Hz and collected data were analyzed off-line. Their work was also video recorded to obtain the ground truth data. **Figure 10** shows an example of an image collected in the factory. We show overviews of work of workers A, B, and C in **Table 1**. Due to the page limitation, we only show a brief overview of work of workers D, E, F, G, and H in **Table 2**. The standard lead times of the work of workers B and C were modified because the work has two kinds of operation processes and they are completely different. Refer to Section 3.2 for that reason. Since workers D and E performed the same operation process, their standard lead times were identical. Also, workers F, G, and H performed the same operation process.

[*1] Assume an operation process consisting of the main operation process A and operation process B. When operation process B is performed (inserted) between the start time of A and that of a selected motif, our method cannot find the start time of A. In such case, we should find the end time of A. (We judge it using the DTW distances.) This is a limitation of our method.

Table 1 Overview of factory tasks by workers A, B, and C. Work of workers B and C has two types of operation processes.

	Worker A	Worker B		Worker C	
operation process	Testing board	Testing board (B-1)	Boxing (B-2)	packing (C-1)	Boxing (C-2)
standard lead time (sec)	130	140	420	50	70
modified standard lead time (sec)	n/a	140	560	50	120
data length (sec)	1,440	1,682	838	559	63
flow of operations (To keep a secret of a factory, we show only abstracted operation flows.)	1: Placing board 2: ABB 3: RBOffset 4: Setting bench 5: Setting head voltage 6: Checking AGC 7: Blooming 8: Flex filter 9: Removing board	1: Placing board 2: Checking image 3: Gamma characteristic 4: White balancing 5: Auto shutter 6: Modulation 7: Shading 8: Checking scratches 9: Preset 10: Removing board	1: Closing 2: Preparing box 3: Labeling 4: Preparing pack 5: Preparing AP 6: Preparing board	1: Configuration 2: Labeling 3: Packaging 4: AP 5: Packing 6: Scanning	1: Closing 2: Preparing box 3: Labeling 4: Preparing pack 5: Preparing AP 6: Preparing board

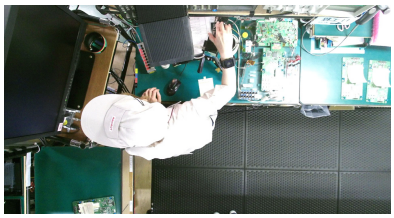


Fig. 10 Example of image captured during task.

6.2 Evaluation Methodology

To investigate the effectiveness of our method, we prepare the following methods.

- *Proposed*: This is our proposed method.

- *w/o disc*: This method is also based on our method. This method does not employ the motif candidate selection using the discretization. In other words, this method randomly extracts n_{init} motifs from raw sensor data and evaluates them using DTW.

- *SV*: This is a supervised method where a sensor data segment between s_1 and $s_1 + \Delta$ is given. By tracking the segment using the particle filter tracker based on DTW, this method finds the start time of each period. (Δ is 3 seconds.)

In addition, we test FFT and autocorrelation, which are commonly used methods for frequency analysis.

- *FFT*: We simply analyze the entire sensor data sequence using Fast Fourier transformation. This method provides only the amplitude values for frequencies and cannot estimate the lead time of each period.

- *Autocorrelation*: Similar to FFT, this method also reveals the frequency of the sensor data using the autocorrelation.

The evaluation criteria of this study are the mean absolute errors (MAEs) for the start time estimation and lead time estimation. When we calculate an error of lead time of a period, we find a period of the ground truth with the largest overlap with the estimated period and compare their lead times. When we calculate an error of a start time, we find a start time of the ground truth closest to the estimated start time and compute the difference. We employ this evaluation methodology because the methods do not always find all the operation periods. (Our method could find all the periods in the data.) Experimental parameters used in this study are shown in **Table 3**, which are determined based on our preliminary experiments.

Table 2 Overview of factory works of workers D, E, F, G, and H.

	Worker D	E	F	G	H
operation process	Screwing		Final test		
standard lead time (sec)	55		55		
data length (sec)	677	692	867	710	682

Table 3 Experimental parameters.

parameter	value
t_{init}	standard lead time [sec]
t_{ms}	5×standard lead time [sec]
n_{init}	0.5×standard lead time
n_s	60
c	0.6
k	10
# symbols	5
frame length for symbolization	5 [sample]

6.3 Results

6.3.1 Frequency Analysis

Here we show results of conventional frequency analysis methods. Since the *FFT* method cannot output the lead time of each period, we show the power spectrum for worker A in **Fig. 11**. As shown in the result, the amplitude corresponding to the standard lead time (130 seconds) is not high and the result captured longer periods (about 180 seconds). Note that the actual average lead time was 127 seconds. **Figure 12** shows a correlogram of *Autocorrelation* for worker B. The second period of her work started 133 seconds after the start of sensor data collection, the third period started 274 seconds after the start of the collection, and the fourth period started 828 seconds after the start of the collection. As shown in the figure, *Autocorrelation* could not output meaningful results due to the fluctuation and variation in lead times.

We also show results of local FFT and autocorrelation where five-period of sensor data were input. As shown in **Fig. 13**, this method could roughly capture the standard lead time of the work since short-term data were not greatly affected by global (long-term) signal fluctuation. However, capturing precise lead time using this approach is still difficult. Also, as shown in **Fig. 14**, the peak of the correlogram is different from the worker's standard lead time. (Actual average lead time for the first five periods was 138 seconds.)

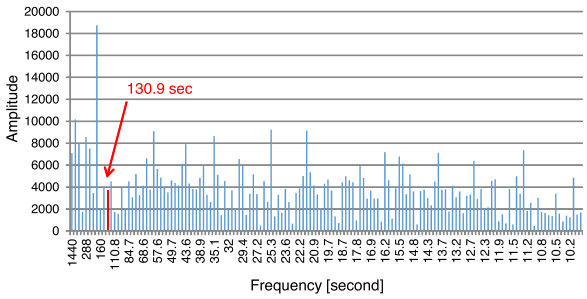


Fig. 11 Power spectrum for sensor data from worker A.

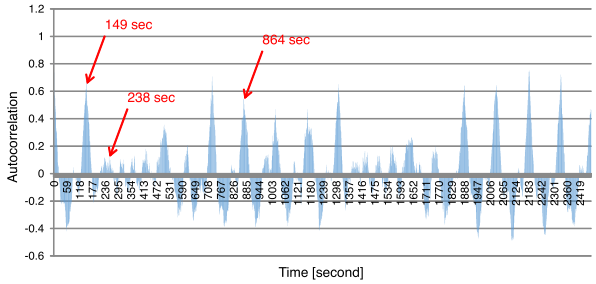


Fig. 12 Correlogram of autocorrelation for sensor data from worker B.

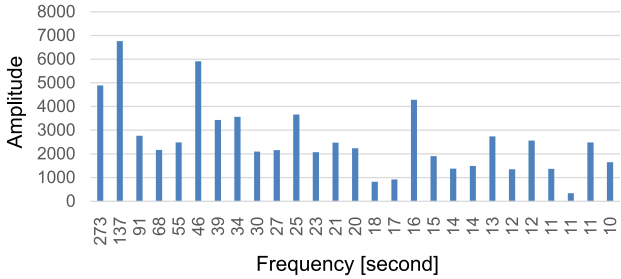


Fig. 13 Power spectrum for short-term sensor data from worker A.

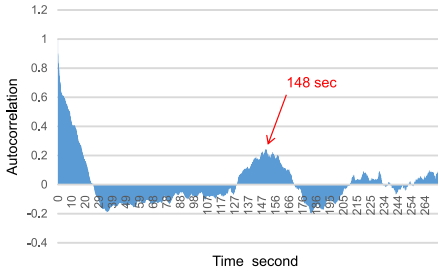


Fig. 14 Correlogram of autocorrelation for short-term sensor data from worker B.

6.3.2 Lead Time Estimation Accuracy

Table 4 shows the MAEs for lead time estimation. As shown in the results, *Proposed* achieved almost the same accuracy as *w/o disc*. The MAEs of *SV* were much poorer than those for our method even though *SV* uses a motif that corresponds to a sensor data segment of the beginning of an operation process. This is because the motif was not identified and the particle filter could not track the motif. Furthermore, since *SV* output only five periods of worker B while our sensor data contain twelve periods of the worker, the MAE for worker B was much larger than the MAEs for other workers. In contrast, our method can automatically find the best motif that occurs according to an operation process model. As above, the MAEs of our method were only about three seconds. We showed our estimation results to

Table 4 MAEs for lead time estimation (seconds).

Worker	A	B	C	D	E	F	G	H
Proposed	1.8	4.2	3.3	0.8	1.6	4.0	2.5	1.2
w/o disc	2.8	3.4	3.3	0.6	5.0	3.5	2.3	1.2
SV	3.2	35.0	6.1	0.8	5.9	7.7	4.4	7.0

Table 5 Error ratios for lead time estimation (%).

	A	B-1	B-2	C-1	C-2
Proposed	1.4	2.9	1.1	6.0	5.0
w/o disc	2.2	2.5	0.6	6.0	5.0
SV	2.5	5.0	21.9	11.0	9.2

	D	E	F	G	H	average
Proposed	1.5	2.9	7.3	4.5	2.2	3.5
w/o disc	1.1	9.1	6.4	4.2	2.2	3.9
SV	1.5	10.7	14.0	8.0	12.7	9.7

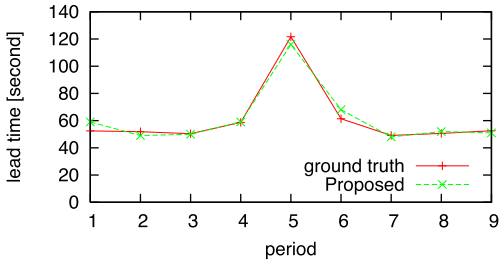


Fig. 15 Transitions of estimated and ground-truth lead times for worker C.

engineers of production management in the factory and they expressed the view that our lead time estimation has sufficient accuracy for finding outlying periods of operation processes in the real factory such as delays because the MAEs were much smaller than the lead times. Also, since the durations of many operations included in operation processes are longer than 10 to 20 seconds, and our method can easily detect the omission of necessary operations and addition of unnecessary operations.

Table 5 shows the error ratios of lead time estimation. As for data from workers A and B, *Proposed* and *w/o disc* achieved low error ratios about 3%. The error ratios for worker C were much higher than those for workers A and B. This is because operation process C-1 was similar to operation process C-2. **Figure 15** shows the transitions of estimated lead times by *Proposed* and actual lead times for worker C. The actual lead time of the fifth period, which corresponds to operation process C-2, was 121 seconds and the estimated lead time was 116 seconds. Also, the actual lead time of the sixth period, which corresponds to operation process C-1, was 61 seconds and the estimated lead time was 68 seconds. This is because our method could not find the correct start time of the sixth period. However, as shown in Fig. 15, our method could capture changes in lead times with small errors. Also, the error ratios for worker F were somewhat large as shown in Table 5. This is because the worker performed an operation in each period after t_{ms} that was not performed before t_{ms}.

6.3.3 Start Time Estimation Accuracy

Table 6 shows the MAEs for start time estimation. The MAEs for *Proposed* and *w/o disc* were small and *Proposed* achieved almost the same accuracy as *w/o disc*. Also, *Proposed* and *w/o disc* greatly outperformed *SV*. The MAEs of start time for workers A and B were larger than those of lead time for workers A and B. This is caused by the shift of estimated periods from periods of

Table 6 MAEs for start time estimation (seconds).

Worker	A	B	C	D	E	F	G	H
Proposed	4.6	4.9	1.8	1.5	2.9	2.6	4.2	1.0
w/o disc	7.1	2.6	1.8	3.1	7.0	2.4	4.2	1.0
SV	3.7	66.0	27.2	4.9	5.4	7.6	6.6	48.4

Table 7 Computation times of methods (seconds).

		select motif candidates	select best motif	track after t_{ms}	estimate start time	total
A	*Proposed*	2.0	223.6	38.6	3.9	268.1
	w/o disc	n/a	1434.8	39.8	0.3	1474.9
B	*Proposed*	1.3	350.3	66.9	37.9	456.4
	w/o disc	n/a	2303.3	66.0	58.6	2427.9
C	*Proposed*	0.3	97.0	17.3	2.0	116.6
	w/o disc	n/a	265.2	17.3	2.0	284.5

the ground truth.

6.3.4 Computation Time

Table 7 shows computation times of the methods run on Windows PC with Core i5 CPU and 8 GB memory. As shown in the results, the computation time of *Proposed* was much shorter than that of *w/o disc*. This is because the computation time for finding the best motif for *Proposed* was much shorter than that for *w/o disc*. By selecting motif candidates, we could reduce the number of motifs evaluated using DTW. As for sensor data from worker A, our method found the best motif within 225.6 seconds after t_{ms}. It corresponds to the duration of two periods of her operation process. After finding the best motif, our method can track the motif in real time.

For data from workers A and B, we could reduce the number of motifs evaluated by about 15% (from 65 to 10 and 70 to 10 motifs, respectively). As shown in Table 7, the computation time for selecting motif candidates was very short because motifs were evaluated using discretized data.

7. Conclusion

This study investigates the feasibility of unsupervised understanding of operation processes in line production systems. Our method achieves highly accurate lead time estimation of factory work in an unsupervised manner. The estimation error of our method was only about 3 seconds and our method has a sufficient accuracy for finding outlying periods. As part of our future work, we will endeavor to understand the structure of operation periods in detail using operation flows described in work instructions as prior knowledge. In addition, because data used in the evaluation section includes about ten operation periods for each participant, the data may not capture fluctuations of lead times caused due to worker fatigue. Investigation of our method with long-term sensor data is also one of the most important future work of this study.

References

[1] Korpela, J., Takase, K., Hirashima, T., Maekawa, T., Eberle, J., Chakraborty, D. and Aberer, K.: An energy-aware method for the joint recognition of activities and gestures using wearable sensors, *International Symposium on Wearable Computers* (*ISWC 2015*), pp.101–108 (2015).

[2] Maekawa, T. and Watanabe, S.: Unsupervised Activity Recognition with User's Physical Characteristics Data, *International Symposium on Wearable Computers* (*ISWC 2011*), pp.89–96 (2011).

[3] Maekawa, T., Yanagisawa, Y., Kishino, Y., Ishiguro, K., Kamei, K., Sakurai, Y. and Okadome, T.: Object-based activity recognition with heterogeneous sensors on wrist, *Pervasive 2010*, pp.246–264 (2010).

[4] Thomaz, E., Essa, I. and Abowd, G.D.: A practical approach for recognizing eating moments with wrist-mounted inertial sensing, *UbiComp 2015*, pp.1029–1040 (2015).

[5] Ranjan, J. and Whitehouse, K.: Object hallmarks: Identifying object users using wearable wrist sensors, *UbiComp 2015*, pp.51–61 (2015).

[6] Lukowicz, P., Ward, J.A., Junker, H., Stäger, M., Tröster, G., Atrash, A. and Starner, T.: Recognizing workshop activity using body worn microphones and accelerometers, *Pervasive 2004*, pp.18–32 (2004).

[7] Aehnelt, M., Gutzeit, E. and Urban, B.: Using activity recognition for the tracking of assembly processes: Challenges and requirements, *WOAR 2014*, pp.12–21 (2014).

[8] Blum, M., Pentland, A.S. and Tröster, G.: Insense: Interest-based life logging, *IEEE Multimedia*, Vol.13, No.4, pp.40–48 (2006).

[9] Maekawa, T., Yanagisawa, Y., Kishino, Y., Kamei, K., Sakurai, Y. and Okadome, T.: Object-blog system for environment-generated content, *IEEE Pervasive Computing*, Vol.7, No.4, pp.20–27 (2008).

[10] Mynatt, E.D., Rowan, J., Craighill, S. and Jacobs, A.: Digital family portraits: Supporting peace of mind for extended family members, *CHI 2001*, pp.333–340 (2001).

[11] Lotter, B. and Wiendahl, H.-P.: *Montage in der industriellen Produktion: Ein Handbuch für die Praxis*, Springer-Verlag (2013).

[12] Johnson, M.J. and Willsky, A.S.: Bayesian Nonparametric Hidden Semi-Markov Models, *Journal of Machine Learning Research*, Vol.14, No.1, pp.673–701 (2013).

[13] Maekawa, T., Nakai, D., Ohara, K. and Namioka, Y.: Toward practical factory activity recognition: Unsupervised understanding of repetitive assembly work in a factory, *UbiComp 2016*, pp.1088–1099 (2016).

[14] Matsubara, Y., Sakurai, Y. and Faloutsos, C.: Autoplait: Automatic mining of co-evolving time sequences, *Proc. 2014 ACM SIGMOD International Conference on Management of Data*, pp.193–204, ACM (2014).

[15] Honda, T.: TrailMarker: Automatic Mining of Geographical Complex Sequences, *Proc. 2016 on SIGMOD'16 PhD Symposium*, pp.22–26, ACM (2016).

[16] Yeh, C.-C.M., Zhu, Y., Ulanova, L., Begum, N., Ding, Y., Dau, H.A., Silva, D.F., Mueen, A. and Keogh, E.: Matrix Profile I: All Pairs Similarity Joins for Time Series: A Unifying View that Includes Motifs, Discords and Shapelets, *IEEE ICDM* (2016).

[17] Doucet, A.: *Sequential Monte Carlo methods*, Wiley Online Library (2001).

[18] Lucke, D., Constantinescu, C. and Westkämper, E.: Smart factory-a step towards the next generation of manufacturing, *Manufacturing Systems and Technologies for the New Frontier*, pp.115–118, Springer (2008).

[19] Radziwon, A., Bilberg, A., Bogers, M. and Madsen, E.S.: The Smart Factory: Exploring adaptive and flexible manufacturing solutions, *Procedia Engineering*, Vol.69, pp.1184–1190 (2014).

[20] Aehnelt, M. and Urban, B.: The knowledge gap: Providing situation-aware information assistance on the shop floor, *HCI in Business*, pp.232–243 (2015).

[21] Korn, O., Schmidt, A. and Hörz, T.: The potentials of in-situ-projection for augmented workplaces in production: A study with impaired persons, *CHI'13 Extended Abstracts*, pp.979–984 (2013).

[22] Koskimäki, H., Huikari, V., Siirtola, P., Laurinen, P. and Röning, J.: Activity recognition using a wrist-worn inertial measurement unit: A case study for industrial assembly lines, *17th Mediterranean Conference on Control and Automation* (*MED 2009*), pp.401–405 (2009).

[23] Ward, J.A., Lukowicz, P. and Tröster, G.: Gesture spotting using wrist worn microphone and 3-axis accelerometer, *The 2005 Joint Conference on Smart Objects and Ambient Intelligence: Innovative context-aware services: Usages and technologies*, pp.99–104 (2005).

[24] Stiefmeier, T., Roggen, D. and Tröster, G.: Fusion of string-matched templates for continuous activity recognition, *11th IEEE International Symposium on Wearable Computers* (*ISWC 2007*), pp.41–44 (2007).

[25] Stiefmeier, T., Ogris, G., Junker, H., Lukowicz, P. and Tröster, G.: Combining motion sensors and ultrasonic hands tracking for continuous activity recognition in a maintenance scenario, *10th IEEE International Symposium on Wearable Computers* (*ISWC 2006*), pp.97–104 (2006).

[26] Huynh, T., Fritz, M. and Schiele, B.: Discovery of activity patterns using topic models, *UbiComp 2008*, pp.10–19 (2008).

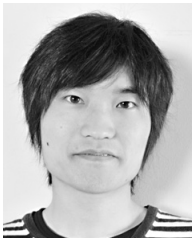
[27] Khan, A., Mellor, S., Berlin, E., Thompson, R., McNaney, R., Olivier, P. and Plötz, T.: Beyond activity recognition: Skill assessment from accelerometer data, *ACM International Joint Conference on Pervasive and Ubiquitous Computing* (*UbiComp 2015*), pp.1155–1166 (2015).

[28] Lin, J., Keogh, E., Lonardi, S. and Patel, P.: Finding motifs in time series, *The 2nd Workshop on Temporal Data Mining*, pp.53–68 (2002).

[29] Müller, M.: Dynamic time warping, *Information Retrieval for Music*

and Motion, pp.69–84 (2007).
[30] Berndt, D.J. and Clifford, J.: Using Dynamic Time Warping to Find Patterns in Time Series, *AAAI-94 Workshop on Knowledge Discovery in Databases*, Vol.10, No.16, pp.359–370 (1994).

Yasuo Namioka is a chief research scientist at Corporate Manufacturing Engineering Center, Toshiba Corporation, Japan. His research interests include factory data management and analysis for production and quality control. Namioka has a Ph.D. in Engineering from Osaka University. He is a member of IPSJ, DBSJ, JSAI, IEICE, and IEEE.



Daisuke Nakai was a student at Osaka University, Japan. He is now working for NTT West. His research interests include indoor positioning and elder-care technologies.

Kazuya Ohara is a first year Ph.D. student at Infomation Science and Technology, Osaka University, Japan. He received his B.S. degree in Division of Electronic and Information Engineering, Osaka University in 2015. His research interests lie in the areas of Activity Recognition and Indoor Localization, Ubiquitous Computing, Machine Learning. He is a student member of IPSJ.

Takuya Maekawa is an associate professor at Osaka University, Japan. His research interests include ubiquitous and mobile sensing. Maekawa has a Ph.D. in Information Science and Technology from Osaka University.