Towards Unsupervised Measurement of Assembly Work Cycle Time by Using Wearable Sensor

Daisuke Nakai, Takuya Maekawa, Yasuo Namioka
Osaka University, Toshiba Corporation
{daisuke.nakai, maekawa}@ist.osaka-u.ac.jp, yasuo.namioka@toshiba.co.jp

***Abstract*—This paper presents a new method for monitoring and managing factory workers of assembly works by using a wearable sensors. In line production systems, a worker repetitively performs predefined operation processes. Since the delay of a worker's process influences the entire line production system, measuring and managing cycle time of each worker's operation process is important. This study tries to measure cycle time of a worker using a wearable sensor in unsupervised manner.**

I. Introduction

Daily activity recognition using sensor data obtained from body-worn sensors is currently one of the most active topics in the pervasive and wearable computing research communities. The activity recognition techniques are expected to be applied to industrial applications such as work analysis of factory workers [4] as well as daily applications such as healthcare, elder care, and lifelogging. This paper focuses on assembly works of factory workers and attempts to analyze the works by using wearable sensors.

Many factories have applied a line production system where each product passes through the same sequence of operation processes. Assembly works by factory workers still constitute the core of the production system and improvement of the assembly works is one of the most important tasks for increasing productivity. In the line production system, a worker repetitively performs predefined operation processes, and each operation process consists of a sequence of operations such as screwing and attaching a part. Because, when a worker completes his/her operation process related to a product, the worker passes the product to the next worker, the worker has to complete one cycle of his/her process within the predefined duration. When the delay of a worker's process occurs, the delay influences the entire line production system. While it is possible to manage and maintain the entire assembly line by measuring cycle time of each worker's operation process, installing a time clock for each worker and making the worker operate the clock during his/her work are costly and burdensome. While working devices with timing measurement functions do exist, such devices are not applied to all workers and all processes.

Therefore, an easy and unobtrusive way to automatically measure a worker's cycle time is required by line managers. One possible way to measure the cycle time is using a wearable sensor and machine learning techniques. By learning and detecting a unique sensor data segment that appears during the operation process, we can measure the cycle time based on the frequency (occurrence interval) of the segment. For example, when a screwing action occurs only once during one cycle of the operation process, we can measure cycle time of the process by learning and detecting a sensor data segment of the action. However, this approach requires training data collected from workers in advance. Since an operation process depends on each worker and it can differ from day to day, training data collection in advance requires substantial costs.

Therefore, this study proposes an unsupervised cycle time measurement method using a wearable sensor. In a line production system, a worker repetitively performs his/her 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," and measures cycle time based on the occurrence intervals of the motif. So we should automatically find a motif that occurs once in each operation cycle. In this method, an operation process model is prepared in advance based on knowledge about the predefined standard duration of the operation process, and our method finds a motif suitable for measuring cycle time that appears in the sensor data sequence in accordance with the operation process model. Because this method permits us to automatically measure cycle time of an operation process by solely using the predefined standard duration of the operation process, we can reduce installation costs of the work management system.

There are several simple ways to measure cycle time such as using the autocorrelation of sensor data or calculating dominant frequencies by analyzing the entire sensor data sequence using the fast Fourier transformation. However, because the duration of one cycle of an operation process of actual factory works fluctuates, using the autocorrelation to calculate the cycle time is difficult. Also, the Fourier analysis cannot provide cycle time of *each* operation process. Furthermore, since each operation process of a worker sometimes includes different operations, these simple methods do not work well. For example, if a worker replaces a part only when the part is broken, the cycle time depends on a test result of the part. We deal with such variations of cycle time in the same operation process by employing a particle filter [1], which is usually used to estimate the states of non-linear systems. A particle filter permits us to track a motif that appears non-linearly.

In this method, an operation process model is first created from the predefined standard duration of a worker's operation process of interest. The method then compares the model with the first t_{ms} minutes sensor data from the beginning of the worker's work, and find a motif that is suitable for measuring cycle time 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

978-1-5090-1941-0/16/$31.00

sensor data from the beginning of the worker's work, 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 occurrence intervals for the operation process model. We compute the score for each randomly selected motif and a motif with the largest score will be used to measure cycle time. That is, after t_{ms}, we track the motif with the largest score using the particle filter.

To track randomly generated motifs in the particle filtering, we calculate the similarity between each motif and each data segment extracted from the entire sensor data. Because sensor data of a worker's operation may vary in time or speed, we employ the dynamic time warping (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, and its computation cost is high. Because our method should compute the similarity between each motif and each data segment extracted from the entire sensor data, it takes long time to find a suitable motif from the sensor data. Therefore, before running the particle filter based on DTW, we first select several motif candidates with small computation costs by using the first t_{ms} minutes sensor data from the beginning of the worker's work. 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 use the discretized data to find several motif candidates that occur in accordance with the operation process model. After that, for only the selected motif candidates, we track them in detail based on DTW and find the best motif.

II. Related Work

We introduce studies on monitoring and analyzing factory works using wearable sensors. Koskimaki et al. [2] 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 the needed operations are performed. The study recognizes such activities as hammering and screwing by using the kNN search. Ward et al. [6] obtain acceleration and sound sensor data using a wrist worn device to recognize wood working activities by using hidden Markov models (HMMs) and a linear discriminative classifier. Stiefmeiner et al. [5] focus on assembly works of bicycles and use inertial sensors attached to several body parts to classify a sensor data segment by computing the distance between the segment and sensor data templates prepared in advance using discretized sensor data. The above all methods for analyzing factory works rely on supervised machine learning approaches and require training data collection.

III. Proposed Method

A. Overview

The proposed method finds a motif that repeatedly occurs in accordance with an operation process model, and measures cycle time of an operation process based on the occurrences of the motif. The overview of our method is illustrated in Fig. 1.

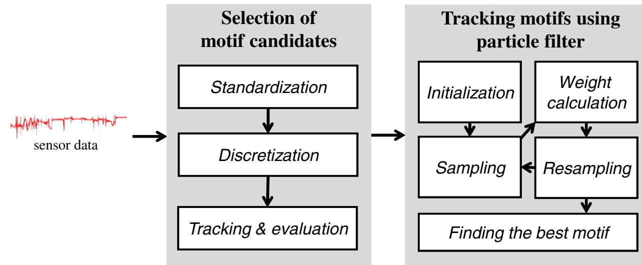

Fig. 1. Overview of proposed method.

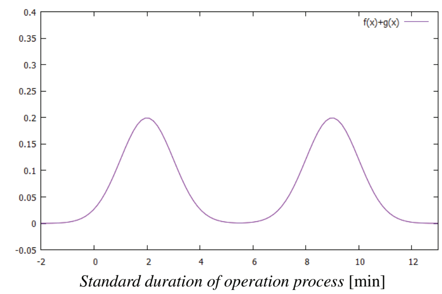

Fig. 2. Example operation process model consisting of two Gaussians.

Our method randomly extracts various motifs from the first t_{init} minutes sensor data from the beginning of the worker's work, and then finds the best motif that occurs in accordance with an operation process model. Here, because we track the occurrences of a motif based on the DTW distance (similarity) between the motif and each sliding window extracted from the entire sensor data, our method requires substantial computational cost. Therefore, as shown in Fig. 1, we first select several motif candidates with small computation costs by using discretized sensor data. We then track the selected candidates in detail based on DTW by using original sensor data, and find the best motif. We measure cycle time of each operation process based on the occurrences of the best motif.

B. Operation process model

Our method employs an operation process model that defines the standard duration of the operation process of interest. In this study, a Gaussian distribution is used to represent the standard duration of the process. Note that, since an operation process sometimes includes additional operations and consequently its standard duration varies, an operation process model is represented as a mixture of Gaussian distributions, each of their means corresponds to each case of standard duration. For example, when an operation process has two possible cases of standard duration, its operation process model consists of two Gaussian distributions as shown in Fig. 2. 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.

C. Selection of motif candidates

We first select few motif candidates that occur in accordance with an operation process model from randomly generated many motifs by using t_{ms} minutes of short sensor data. To select motif candidates with small computation costs, we standardize and then discretize (symbolize) t_{ms} minutes of time-series sensor data according to [3]. 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 an representative value of the frame. 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 **c**, and a value of a frame falls into the area, the value is mapped into **c**. 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 III-D. Note that, since this process deals with the discretized time-series, we employ the Hamming distance instead of the DTW distance to compute the data segment similarity, which permits us to compute the similarities with small computation costs. Then, each tracked motif is evaluated 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.

D. Tracking motifs using particle filter

As mentioned above, cycle time of an operation process varies depending on operations included in the process. Also, the cycle time fluctuates due to the degree of tiredness of a worker and the habituation of the work. Therefore, we achieve motif tracking robust against the fluctuation of the cycle time and the variation of the cycle time 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 non-linear 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.

1) Initialization: In the initialization process, we randomly extract motifs with random lengths from the first t_{init} minutes of the sensor data. We assume an extracted motif as a particle and a timestamp of the first data sample of the motif as time when the motif first occurred. Then we track the subsequent occurrences of the particle (motif). Note that this initialization is executed only when we select top-k motif candidates by 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.

2) 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 $i+1$th occurrence of x_n (i.e., $t(x_n, i+1)$) is determined according to the operation process model as follows:

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

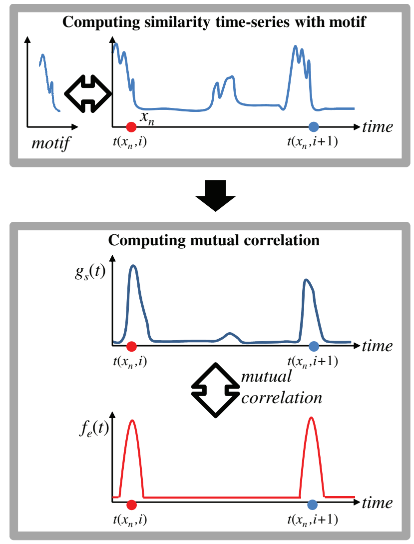

Fig. 3. Weight calculation of particle using mutual correlation.

where Δt is an estimated interval of the occurrence of x_n that is randomly sampled from the operation process model. We sample n_s particles from x_n as the estimated $i+1$th occurrences of x_n, i.e., prior estimations.

3) 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., $t(x_n, i+1)$, and 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 sensor data. Because the similarity value is computed for each sliding window, we can obtain time-series of the similarity values with the motif as shown in $g_s(t)$ of Fig. 3. Note that we compute the time-series $g_s(t)$ between $t(x_n, i) - \sigma$ and $t(x_n, i+1) + \sigma$. When the ith and $i+1$th occurrences of x_n are actually at $t(x_n, i)$ and $t(x_n, i+1)$, respectively, the similarity values around $t(x_n, i)$ and $t(x_n, i+1)$ become large as shown in $g_s(t)$ of Fig. 3. Ideally, only the similarity values around $t(x_n, i)$ and $t(x_n, i+1)$ are large and the similarity values other than them are small when the motif is suitable for measuring cycle time of the operation process. (A motif that occurs only once during an operation process cycle is suitable for measuring cycle time. In other words, a motif that occurs many times in each operation cycle is not suitable.) In this study, we define a function $f_e(t)$ consisting of a mixture of a Gaussian function whose center of the peak is $t(x_n, i)$ and that whose center of the peak is $t(x_n, i+1)$ as shown in Fig. 3 and represented as follows:

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

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

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

where $\overline{f_e}$ and $\overline{g_s}$ are the means of $f_e(t)$ and $g_s(t)$, respectively. For example, when the estimated $i+1$th occurrence of the motif $t(x_n, i+1)$ is close to the actual occurrence of the motif, the second peak of $f_e(t)$ and that of $g_s(t)$ overlap with each other and thus the computed r value becomes large. In contrast, when $t(x_n, i+1)$ is not close to the actual occurrence of the motif, the computed r value becomes small. We assume the computed r value as the weight of the particle ($i+1$th occurrence of x_n).

4) 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 process, we resample only one particle from the n_s particles according to their weights. That is, the time associated with the resampled particle corresponds to the posterior estimation of the $i+1$th occurrence of x_n.

5) Finding 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 suitable for measuring cycle time of the operation process of interest from the motifs. 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. We also prepare time-series of the similarity values with the motif computed for each sliding window extracted from sensor data before 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 find a motif with the highest score to track the subsequent occurrences of the motif after t_{ms}, i.e., estimate cycle time, by using the above mentioned particle filter. (In the selection of motif candidates process, we only select the top-k discretized motifs.)

IV. Evaluation

A. Data set and methodology

In a real factory of the author's company, we collected 3-axis acceleration data from a worker of an assembly work using Sony SmartWatch3 SWR50 attached to her right wrist. The process of the worker is a final assembly and test process which consists of several operations such as placing a printed circuit board (PCB) inside a body of a product, checking the PCB using a function tester, and putting a covering on the PCB. Because the process sometimes includes a box packing work, the process has two types of standard durations; 140 and 550 seconds. From the worker, we collected about 40 minutes sensor data corresponding to eleven cycles. Note that, because we use three-axis sensor data, the DTW distance (similarity) used in our method corresponds to the average distance over the three axes.

TABLE I. Computation times (seconds) of our method and the naive method.

methods	select motif candidates	particle filter	tracking after t_{ms}	total
naive	n/a	1765.3	63.6	1828.9
proposed	6.0	76.1	64.1	146.2

To investigate the accuracy of our method, this evaluation employs the mean absolute error of estimated cycle time by comparing ground truth cycle time obtained from video recordings. Also, to evaluate the effectiveness of our method, we prepare a method that does not employ the motif candidate selection using the discretization (naive method). That is, the naive method finds the best motif by using particle filtering based only on DTW.

In our implementation, $t_{init} = 140$ seconds and $t_{ms} = 20$ minutes. Also, $k = 3$ and $n_s = 60$.

B. Results

The mean absolute errors of our method and the naive method were 4.3 seconds and 4.6 seconds, respectively. The errors are very small and we confirmed that our architecture achieves sufficient accuracies without using any training data. Also, our method could achieve almost the same accuracy as the naive method that tracks many randomly selected motifs using DTW (15 motifs in our implementation).

Table I shows computation times of our method and the naive method for each process. Our method could find the best motif for only 82.1 seconds (6.0+76.1) using about 20 minutes sensor data. In contrast, it took about 30 minutes for the naive method to output the best motif. By discretizing sensor data, we could find motif candidates for only 6 seconds using about 20 minutes sensor data. The best motif extracted by our method was an action of checking a PCB using a tester.

References

[1] A. Doucet, *Sequential Monte Carlo methods*. Wiley Online Library, 2001.

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

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

[4] P. Lukowicz, J. Ward, H. Junker, M. Stager, G. Tröster, A. Atrash, and T. Starner, "Recognizing workshop activity using body worn microphones and accelerometers," in *Pervasive 2004*, 2004, pp. 18–32.

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

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