The 31st International Technical Conference on Circuits/Systems,
Computers and Communications (ITC-CSCC 2016)

A Scheme of Motion Recognition For Wearable Device Gesture

Inhye Park[1], Sang-Yub Lee[2], and Jae-Jin Ko[3]
[1-3] Korea Electronics Technology Institute
Daewangpangyo-ro 712, Bundang-gu, Seongnam-si, Gyeonggi-do 13488, Republic of Korea
E-mail: {[1]ine.park, [2]syublee, and [3]jaejini}@keti.re.kr

Abstract: This paper presents a scheme of the motion recognition for wearable device gesture. To recognize motion of user's arm, we use 6-axis sensors, which consists of a 3-axis accelerometer, a 3-axis gyroscope. When user moves as predefined gesture with wearable devices, the several sensor data forms correlation relation between themselves over the certain level. So, we implement cross-correlation function between generated sensor data. And we filter these sensor data as a certain level to classify specific patterns. We show overall motion recognition architecture at the first. Then our a methodology including the generated sensor data and data processing are explained. Finally, we experiment our method to test platforms divided as wearable device and AVN platform and show the results.

Keywords— **MEMS sensors, wearable decive gesture, air gesture recognition**

1. Introduction

Due to increasingly bigger needs of convenience and useful services for user, the demand of small IT devices for the IoT(Internet of thing) including the wearable devices are increased. The small devices have typical advantages of lower cost, smaller size and lower power consumption [1-2]. So, small devices have been using at many industrial aspects. Especially, the wearable device is changing from simply supporting health services including heart rate and a step count to more advanced service. Gesture recognition using motion sensors of wearable device is one of these advanced operations [3-4].

Usually, the researchs of gesture recognition have been studied as mainly two parts, *vision-based* and *accelerometer and/or gyroscope based* [5]. The *vision-based* method used for 3D regocnition in big system such as image processing scope. The other is usally used to estimate the real-time gait cycle and to recognize hand gesture[3]. Since the image detection processing of *vision-based* method burdens in system cost and energy consumption, *accelerometer and/or gyroscope based* method are implemented in the embedded and small devices such as our system. Due to the limitations such as unexpected ambient optical noise, slower dynamic response, and relatively large data collections/processing of vision-based method [6], our recognition system is considered based on cross-correlation scheme between axis of MEMS(micro-electromechanical system) acceleration sensor.

Existing gesture recognition approaches include template-matching [7], dictionary lookup [8], statistical matching [9], linguistic matching [10], and neural network [11]. For sequential data such as measurement of time series and acoustic features at successive time frames used for speech recognition, HMM (Hidden Markov Model) is one of the most important models [12]. It is effective for recognizingpatterns with spatial and temporal variation [13].

This paper presents the gesture recognition method, which operates simply and correctly without additional HW function. We implement cross-correlation function to match with predefined gesture of specific motion. This paper is organized as follows. Section 2 presents the recognition method of wearable device gesture using 6-axis sensors. Section 3 describes the experiment and its results of method. The conclusion is expressed in last section.

2. The method

We represent the simple method about data processing and filtering for wearable gesture recognition. We assume the two part as separate devices. One is wearable device generating motion sensor data and the other is the AVN platform as processing and interpreting the raw sensor data. To explain architecture, first of all, we describe system architecture of gesture recognition as shown in Figure 1. The architecture in Figure 1 sequencially summarize this section. The key methods are cross-correlation and classifying function. More detailed represents will be explained.

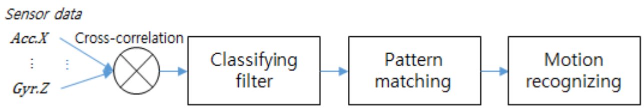

Figure 1. System architecture

2. 1 Generated sensor values

In wearable devices, accelerometer and gyroscope sensors generate sensor information as 6-aixs form. To measure accurate motion sensor data, we set an designated sampling period as t_s. Also we let wearable device transmit the sampling sensor data as a fixed duration, t_{tx}, for efficient data collection. Because wearable device collects sampled single sensor data, D_i, during t_{tx}, the AVN platform receives several sensor data from wearable device at a time. Therefore, the amount of received sensor data at a time in AVN platform D_s, is

$$D_s = 6 * \sum_{i=0}^{k} D_i , k = T_{tx}/T_s. \quad (1)$$

To support seamless delivering of sensor data, we adopt the fixed time cycle concept in the time domain. Also, to synchronize of gesture information, an initilization duration, T_{init}, is inserted in the cycle of transmission, T_{cycle}. The initializing duration contains dummy data like zero. Because an initializing duration is situated at the head of sensor data, a head of gesture information from wearable device be synchronized. The total valid data, D_{total}, during overall transmission time, $n * T_{cycle}$ is

$$D_{total} = \frac{T_{motion}}{T_{cycle}} * n * T_{cycle} * D_s, \quad (2)$$

at $T_{cycle} = T_{init} + T_{motion}$. (3)

We use only D_{total} to recognize gesture.

2. 2 Data processing and motion recogntion

We describe the method to process data and recognize gesture of wearable device using 6-axis sensor information D_{total}. The each 6-axis sensors contain separated data. But several sensor data correlated with some particular gestures. Follow predefined pattern feature of gesture data, we represent the correlation between sensor data as matrix form and do the filtering and matching method for gesture recognition.

Let $\boldsymbol{A}$ is cross-correlation matrix of D_{total}, and ρ_{ij} is the element meaning cross-correlation between sensor i and j data generated at same time.

$$\boldsymbol{A} = (\rho_{ij}) = \left(\frac{\sigma_{ij}}{\sigma_i\sigma_j}\right) = \begin{bmatrix} 1 & \rho_{A_xA_y} & \cdots & \rho_{A_xG_z} \\ \rho_{A_yA_x} & 1 & \cdots & \rho_{A_yG_z} \\ \vdots & & \ddots & \vdots \\ \rho_{G_zA_x} & \rho_{G_zA_y} & \cdots & 1 \end{bmatrix} \quad (4)$$

In Eq. 4, $\rho_{A_xA_y}$ means cross-correlation between x-axis and y-axis of accelerometer sensor. As same role, $\rho_{A_xG_z}$ means cross-correlation between accelerometer sensor's x-axis and gyroscope sensor's z-axis. Since correlation value of itself is 1, the diagonal value of $\boldsymbol{A}$ is 1. A property of cross-correlation is $\left(\frac{\sigma_{ij}}{\sigma_i\sigma_j}\right) = \left(\frac{\sigma_{ji}}{\sigma_j\sigma_i}\right)$, so the left-down side and the right-up side of $\boldsymbol{A}$ are contains exactly same values.

Next, $\boldsymbol{A}$ enters Classifying filter block(CFB) as the input. At this block, un-valuable elements ρ_{ij} are filtered which has lower correlation value than pre-optimized constant C_{opt}. So, $\boldsymbol{A}$ multiplies $f(x)$ of Eq. 5. to itself in CFB. Therefore only a valid elements set $\boldsymbol{A} = \{\rho_{ij}, \ldots\}$ is remained after CFB.

$$f(x) = \begin{cases} x, & if\ C_{opt} > |x| \\ 0, & else \end{cases} \quad (5)$$

At, $x = \rho_{ij}$, $\{i,j\} \subset \{A_x, A_y, A_z, G_x, G_y, G_z\}$, $i \neq j$

After then, in the Pattern matching block(PMB), the several maximum correlation elements are choose for motion matching. To get the union of the most valuable element's position, U_m, PMB iterates Eq. 6 at C_{iter} times, which this constant is pre-defined from administrator.

$$U_m = \max_{i,j} \arg\{|[\rho_{ij}]|\}, \ \boldsymbol{A} = \boldsymbol{A} - U_m, \ 0 \notin U_m \quad (6)$$

After acquisition of U_m, finally, pattern is recognized to a specific motion using comparison process of motion data union. The position of maximum correlation values are used in this process. New processed data is compared with predefined position of distinguishing correlation pattern in each gestures. If the features of pattern is similar with predefined data base, the new motion data is regard to specific gesture. We will explain this process using an example in detail at the continued session 3.

3. Experiments and results

3. 1 Experiment system architecture

To verify our method, we tested with 6-axis sensor in a manufactured wearable device. We chose EFM32 HG32 of Cortex-M series as MCU of wearable device. The AVN platform we chose is embedded application board called i.MX6 AI(automotive infotainment), which has a 4-CPU Cortex-A9 MCU of ARM and 2GB DDR3. For low-power consumption, we adopted BLE protocol to communicate sensor data between wearable device and AVN platform.

Overall system architecture of experiment are shown in Figure 2. First of all, the wearable device generate sensor value as the rule. Generated sensor values are transmitted to AVN platform via BLE. And then, the explained method including processing blocks is executed at the AVN platform. Finally, the AVN platform let users know about the recognized motion.

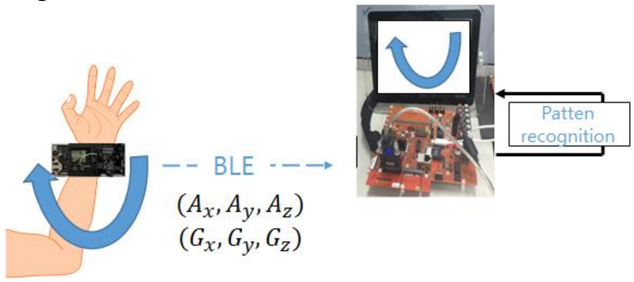

Figure 2. Test system for experiments

3. 2 Predefined pattern data and results

To match with generated sensor data, we set several predefined data as motion data. As shown in Table 1, we simply employ 6 motion, which consist of *twisting a wrist to left/right*, *bending an arm to up/down,* and *rotating a wrist to clock-wise/anticlock-wise.* The motion number 1, *TW left*, is means to twist the wearing wrist to left and returning at the beginning position, as represented as words and picture in the upper of Table 1. Every motion are described in Table 1 as same role from the motion number 1 to 6.

The several correlation data of predefined motion are shown in Table 2. The axis on top and left side mean sensor factor as you see in Table 2. Each value means cross-correlation between sensor data, so 3rd raw and 2nd column value(0.85) means $\rho_{A_xA_y}$. As shown in Table 2 (a) and (b), some factors has unexpectable high value like $\rho_{A_yA_z}$ of (a) and $\rho_{A_xG_y}$ of (b). Using these specific patterns, our system recognize the gesture.

Table 1. The gesture list used in this paper

Motion		Explain	
①	TW left	To twist the wearing wrist to left and return	
②	TW right	To twist the wearing wrist to right and return	
③	BD up	To bend the wearing arm to up and return	
④	BD down	To bend the wearing arm to down and return	
⑤	RT clock	To rotate the wearing arm to clockwise and return	
⑥	RT anti-clock	To rotate the wearing arm to anti-clockwise and return	

Table 2. The motion data table (ρ_{ij}). The left-down side contents is deleted because it contains same value with right-up side.

(a) An example motion data of ①

	A.x	A.y	A.z	G.x	G.y	G.z
A.x	1	0.85	0.74	0.37	-0.03	0.38
A.y		1	0.94	0.12	0.09	0.23
A.z			1	0.07	0.16	0.10
G.x				1	-0.74	0.44
G.y					1	-0.29
G.z						1

(b) An example motion data of ②

	A.x	A.y	A.z	G.x	G.y	G.z
A.x	1	0.78	-0.56	-0.22	0.8	-0.6
A.y		1	-0.70	-0.37	0.78	-0.49
A.z			1	-0.03	-0.61	0.31
G.x				1	-0.36	0.27
G.y					1	-0.47
G.z						1

To explain the results of test, the generated actual sensor data is represented at the Figure 3. The 6 vertical bars show 6-axis data from *A.x* to *G.z*, respectively. And the sections horizonally devided represent the motion number from ① to ④ in sequentially. As shown in the Figure 3, each sensor data in every motion has different pattern. From these patterns, we can see that every motion gesture has unique correlated relationship between sensor axis.

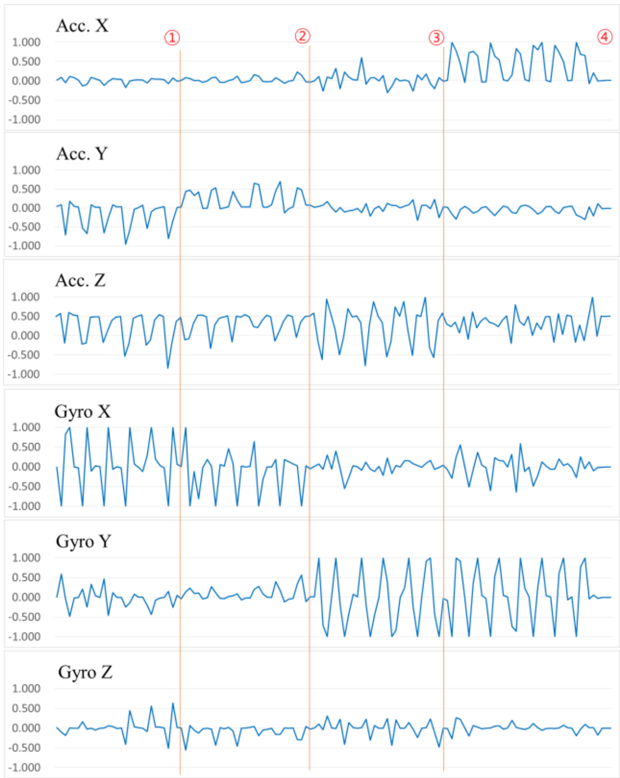

Figure 3. Generated sensor data of the motion①-④ (normalized).

The results of experiment are shown in Table 3. We set C_{opt} and C_{iter} to 4. We did specific gesture each 20 times and checked how much our system correctly recognizing. As shown in Table 3, average recognition rate is pretty high as 90.8%. From the results, we confirm our motion recognition system is operating with high reliability.

Table 3. Results of experiment. Successful motion recognition rate.

	①	②	③	④	⑤	⑥
%	90	95	85	85	95	95

4. Conclusion

This paper presented a scheme of the motion recognition for wearable device gesture. To recognize motion of user, we used 6-axis sensors at a scheme and experiments. When user moves as predefined gesture with wearable devices, the specific sensor data have correlation between themselves over the certain level. So, we implement cross-correlation between generated sensor data and filtering as a certain level to classify specific patterns. We shown overall motion recognition architecture at the first. Then our a method including the generated sensor data and data processing was explained. Finally, we experimented our method to test platforms divided as wearable device and AVN platform and show the results. As the results, we confirm our motion recognition system is operating with high reliability as 90.8%.

5. Acknolodgement

This research was supported by a grant from the IT R&D program of MOTIE/KEIT [10051106, Development of band type wearable device that contains the automotive smart key function]

References

[1] R. Noe-Nilssen and L. L. Helbostad, "Estimation of gait cycle characterestrics by trunk accelerometry," *Elsevier Journal of Biomechanics*, Vol. 37, pp. 121-126, 2004

[2] X. Yuan, S. Yu, S. Zhang, C. Liu, and S. Liu, "Modeling and Analysis of Wearable Low-Cost MEMS Inertial Measurement Units," In Proc. of *IEEE 15th International Conference on Electronic Packaging Technology*, 2014

[3] C. C. Yang, Y. L. Hsu, K. S. Shih, and J. M. Lu, "Real-Time Gait Cycle Parameter Recognition Using a Wearable Accelerometry System," *Journal of Sensors*, pp. 7314-7326, Nov. 2011

[4] M. Kusserow and O. Amft, "Modeling Arousal Phases in Daily Living Using Wearable Sensors," *Journal of IEEE Transaction on Affective Computing* Vol.4, No.1, pp. 93-105, Jan 2014

[5] R. Xu, S. Zhou, and W. J. Li, "MEMS Accelerometer Based Nonspecific-User Hand Gesture Recognition," *IEEE Sensors Journal* Vol.12, N0.5, May 2012

[6] S. Zhou, Q. Shan, F. Fei, W. J. Li, C. P. Kwong, and C. K. Wu, "Gesture recognition for interactive controllers using MEMS motion sensors," In Proc. of *IEEE international Conference Nano/Micro Engineered and Molecular Systems*, Jan. 2009, pp.935-940

[7] J. S. Lipscomb, "A trainable gesture recognizer," *Pattern. Recognit.*, vol. 24, no. 9, pp. 895–907, 1991.

[8] W. M. Newman and R. F. Sproull, *Principles of Interactive Computer Graphics*. New York: McGraw-Hill, 1979.

[9] D. H. Rubine, "The Automatic Recognition of Gesture," Ph.D dissertation, Computer Science Dept., Carnegie Mellon Univ., Pittsburgh, PA, Dec. 1991.

[10] K. S. Fu, *Syntactic Recognition in Character Recognition*. New York:Academic, 1974, vol. 112, Mathematics in Science and Engineering.

[11] S. S. Fels and G. E. Hinton, "Glove-talk: A neural network interface between a data glove and a speech synthesizer," *IEEE Trans. Neural Netw.*, vol. 4, no. l, pp. 2–8, Jan. 1993.

[12] C. M. Bishop, *Pattern Recognition and Machine Learning*, 1st ed.New York: Springer, 2006.

[13] T. Schlomer, B. Poppinga, N. Henze, and S. Boll, "Gesture recognition with a Wii controller," in *Proc. 2nd Int. Conf. Tangible and Embedded Interaction (TEI'08)*, Bonn, Germany, 2008, pp. 11–14.