

Research Article

Fall-Detection Algorithm Using 3-Axis Acceleration: Combination with Simple Threshold and Hidden Markov Model

Dongha Lim,¹ Chulho Park,¹ Nam Ho Kim,^{1,2} Sang-Hoon Kim,¹ and Yun Seop Yu¹

¹ Department of Electrical, Electronic and Control Engineering and IITC, Hankyong National University, 327 Chungang-no, Anseong, Gyeonggi-do 456-749, Republic of Korea

² Laon People Co. Ltd., B-402 Bundang Technopark, 255 Yatapnam-ro, Bundang-gu, Seongnam, Gyeonggi-do 463-760, Republic of Korea

Correspondence should be addressed to Yun Seop Yu; ysyu@hknu.ac.kr

Received 10 February 2014; Accepted 19 August 2014; Published 17 September 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 Dongha Lim et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Falls are a serious medical and social problem among the elderly. This has led to the development of automatic fall-detection systems. To detect falls, a fall-detection algorithm that combines a simple threshold method and hidden Markov model (HMM) using 3-axis acceleration is proposed. To apply the proposed fall-detection algorithm and detect falls, a wearable fall-detection device has been designed and produced. Several fall-feature parameters of 3-axis acceleration are introduced and applied to a simple threshold method. Possible falls are chosen through the simple threshold and are applied to two types of HMM to distinguish between a fall and an activity of daily living (ADL). The results using the simple threshold, HMM, and combination of the simple method and HMM were compared and analyzed. The combination of the simple threshold method and HMM reduced the complexity of the hardware and the proposed algorithm exhibited higher accuracy than that of the simple threshold method.

1. Introduction

In the past decade, the population in the world has been increasingly aging [1]. Korea, for example, is rapidly changing into an “aging society.” The elderly, especially those above the age of 65, are exposed to falls owing to the deterioration of their physical functions [2]. When an elder person falls and becomes unconscious or is unable to move his/her body, he/she may succumb to the injuries that caused the fall [3]. Thus, research and development of a system that can automatically detect falls in the elderly or other patients has been actively studied [4–7].

Because of the expansion of the Internet in the 90s, it is now commonly referred to as the Internet of Things (IoT). The pervasive and seamless interaction among objects, sensors, and computing devices is an important concern of the IOT [8]. Smart embedded objects such as a fall-detection sensor with wireless communication [9] will also become an important part of the IoT.

The identified fall-detection systems can be classified primarily into two categories: context-aware systems and wearable devices [10–21]. Context-aware systems use devices such as cameras, floor sensors, infrared sensors, microphones, pyroelectric infrared (PIR) sensors, and pressure sensors, deployed in the environment, to detect falls [10–13]. Their principal advantage is that a person is not required to wear any special equipment. Wearable device-based approaches rely on clothing with embedded sensors to detect the motion and location of the body of the subject [14–27]. The advantages of wearable devices are the cost efficiency, ease of installation, setup, and operation of the design.

There are two main approaches (algorithms) to detect falls: simple threshold and machine learning methods. In the simple threshold method, threshold values of specific parameters calculated from sensor data such as 3-axial acceleration are used to detect a fall [14–20]. Automatic fall detection using a threshold-based method of single parameters, calculated using acceleration measured by an accelerometer, has a high



FIGURE 1: Photograph of the sensor node (80 mm \times 50 mm) for fall detection. (a) Front side and (b) back side.

sensitivity (about 100%); however it has a relatively low specificity [14, 15]. Automatic fall detection using multiple parameter combinations has a relatively high sensitivity (85.7%) and specificity (90.1%) [16]. Automatic fall detection using angular velocities measured using a gyroscope has a high sensitivity (100%) and specificity (97.5%) [18]. Further, automatic fall detection using multiple parameters that are calculated using the acceleration and angular velocities measured by an accelerometer and a gyroscope, respectively, has a high sensitivity (91%) and specificity (92%) [19]. They are simple to implement and their computation effort is minimal. However, they have a problem with the tolerance of individual behavior and are less accurate for detecting falls that occur. In the machine learning method, various types of fall and activity of daily living (ADL) patterns are trained by a learning algorithm and then an event is classified as a fall or ADL by applying it to an evaluation algorithm [21–27]. The machine learning methods include support vector machine (SVM) [21, 22], Gaussian distribution of clustered knowledge [23], decision tree [24], and hidden Markov model (HMM) [25–27]. The machine learning method is more sophisticated and leads to better detection rates with accuracy of over 95%. Unfortunately, it is difficult to implement the machine learning approach due to the heavy computational and resource requirements [4]. The combination of the two approaches for fall detection has not yet been investigated.

In this paper, a fall-detection algorithm using 3-axis acceleration is proposed. The fall-feature parameters, calculated from the 3-axis acceleration, are applied to a simple threshold method [20]. Then, the falls that are determined from the simple threshold are applied to the HMM [25–27] to distinguish between falls and ADLs. The results from a simple threshold, HMM, and the combination of the simple method and HMM are compared and analyzed.

2. Materials and Methods

A novel fall-detection algorithm using an acceleration sensor node is presented. Because the chest of the subject is near the body's center of gravity [28], the sensor node is attached with an elastic belt on the chest of the subject. The sensor

node, as shown in Figure 1, measures sensor data and sends them to the gateway (portable computer (PC)) using a ZigBee network processor. The software environment used in the experiment was Visual Studio 2008 and fall-detection code was written in the C language on a Windows XP PC.

2.1. Subjects and Testing Activities. Intentional falls were performed by six healthy volunteers: four male and two female subjects whose ages ranged from 20 to 50, height from 160 to 185 cm, and weight from 50 to 85 kg. The falls were performed using a mattress (thickness: 20 cm). Each subject performed 7 types of activity (three types of fall and four types of ADL) as follows:

- (i) ADL-a: walking,
- (ii) ADL-b: running,
- (iii) ADL-c: standing jumping,
- (iv) ADL-d: lying down and standing up from a bed,
- (v) Fall-a: falling forward over something,
- (vi) Fall-b: falling laterally by losing balance,
- (vii) Fall-c: sliding and falling backward.

A total of 320 ADLs and 240 falls were tested. The total number of each activity for subjects A, B, C, and D (age: 20s) was 15 and for subjects E (age: 50s) and F (age: 40s) was 10. The ADLs used in this study were activities that could cause high impact or abrupt changes in a person's movement.

2.2. Hardware Description. The fall-detection system implemented in this paper consisted of a sensor node with a 3-axis accelerometer ± 8 g triaxial accelerometer (BMA150, Bosch) [30] and wireless communication module (CC2530, Texas Instrument) [31], a gateway to collect the information from multiple wireless sensor nodes, and a server to determine falls by applying the parameters from the 3-axis acceleration to the proposed fall-detection algorithm. The sensor was controlled by the ZigBee network processor. The sampling rate was set to 100 Hz, a bandwidth exceeding the characteristic response of human movement. Each triaxial acceleration was statistically

calibrated in order to correct any possible axis tilt due to the orientation of the device on the subject or lower back tilt of the subject. To execute the algorithm that detects a fall suffered by an elder person, the gateway monitors the accelerations sent from the sensor nodes and calculates several parameters including the directions, magnitudes, and angles of the elder person's motion from the sensing data. The algorithm stores the measured data and calculates the parameters.

2.3. Fall-Feature Parameters. To detect a fall, five types of parameters are used in the analyses [27]. The fall-feature parameters of sum vector magnitude (SVM) A_{SVM} , differential SVM (DSVM) A_{DSVM} of acceleration, angle θ , gravity-weighted SVM (GSVM) A_{GSVM} , and gravity-weighted DSVM (GDSVM) A_{GDSVM} are calculated using the following equations [27]:

$$A_{SVM}(i) = \sqrt{A_x^2(i) + A_y^2(i) + A_z^2(i)},$$

$$\theta(i) = \tan^{-1} \left(\frac{\sqrt{A_y^2(i) + A_z^2(i)}}{A_x(i)} \right) \times \frac{180}{\pi},$$

$$A_{DSVM}(i) = \left((A_x(i) - A_x(i-1))^2 + (A_y(i) - A_y(i-1))^2 + (A_z(i) - A_z(i-1))^2 \right)^{1/2},$$

$$A_{GSVM}(i) = \frac{\theta(i)}{90} \times A_{SVM}(i),$$

$$A_{GDSVM}(i) = \frac{\theta(i)}{90} \times A_{DSVM}(i),$$
(1)

where i denotes the sample number and $A_x(i)$, $A_y(i)$, and $A_z(i)$ denote the x -axial, y -axial, and z -axial accelerations of the i th sample, respectively. The Euler angle θ denotes the tilted angle between the accelerometer y -axis and the vertical direction.

2.4. Fall-Detection Algorithm. Figure 2 shows the flow diagram for the fall-detection system. Real-time 3-axis accelerations are sent from the sensor handset of the subject to the server through the ZigBee network and then the five types of fall-feature parameters are calculated from the sample data in the learning and evaluating range.

The fall-feature parameters are applied to the simple threshold method to determine whether a parameter is above a certain threshold within a time interval. If any parameter is above a threshold, the sample is determined to be a possible fall indicating a subject fall event or ADL similar to a fall event. The simple threshold algorithm for multiple parameters using double parameters is shown in Algorithm 1. The thresholds are determined from the receiver operating characteristics (ROC) curve from which both true positive

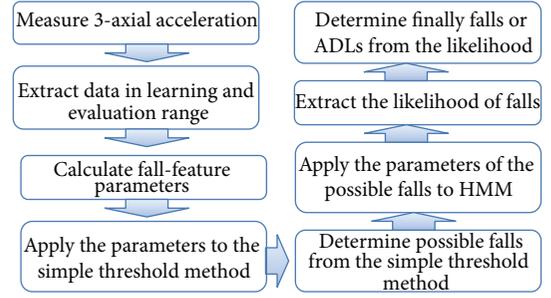


FIGURE 2: Flow diagram for falls detection system.

and false positive rates are calculated for all parameter values. The threshold values are determined when the specificity is best with a sensitivity of 100%. Instead of using all the events, only the fall-feature parameters of the events determined to be possible falls from the simple threshold method are applied to the HMM algorithm [29, 32–34] as shown in Figure 3 and Algorithm 2. First, the learning process of the HMM is performed for four types of ADL and three types of fall; all the values in the model matrices $\lambda_i = (A_i, B_i, \pi_i)$ of all the single parameters for i activities of four types of ADL and three types of fall are calculated using Baum-Welch learning algorithm [29]. A_i , B_i , π_i , M , and N denote the state transition probability distribution, observation emission probability distribution, initial state distribution, number of invisible states, and number of observation values for i activities, respectively. In this paper, M and N are used as 4 and 8, respectively. Then, based on the leaning database, an activity is evaluated by applying the HMM with the parameter. The likelihood of all the single parameters for the observation sequences is calculated using the HMM evaluation algorithm [29]. Finally the maximum probability among the four types of ADL and three types of fall is determined. If the selected activity with the maximum probability is among the three types of fall, the fall is alarmed; otherwise, it is determined to be an ADL.

3. Experimental Results

Figure 4 shows the ROC curve of the fall detection obtained from the simple threshold using a single parameter. The single parameter $A_{GSVM} = 2.5$ g has the best fall detection with sensitivity, specificity, and accuracy of 92.92, 81.56, and 88.05%, respectively. The best specificity with 100% sensitivity is 68.13% when $\theta = 60^\circ$. This is chosen as the threshold value.

Figures 5, 6, 7, and 8 show the ROC curves of the fall detection obtained from the simple threshold using the double parameters of A_{SVM} and θ , A_{DSVM} and θ , A_{GSVM} and θ , and A_{GDSVM} and θ , respectively, as shown in Algorithm 1. Each best fall detection using double parameters is shown in the captions of Figures 5, 6, 7, and 8. As the false positive rate is decreased, the true positive rate is abruptly decreased. Among them, the best fall detection is of sensitivity 98.75%, specificity 94.38%, and accuracy 96.25% when $A_{SVM} = 2.5$ g and $\theta = 65^\circ$, as shown in Figure 5. The best specificity with 100% sensitivity is 91.56% and accuracy is 95.18% when

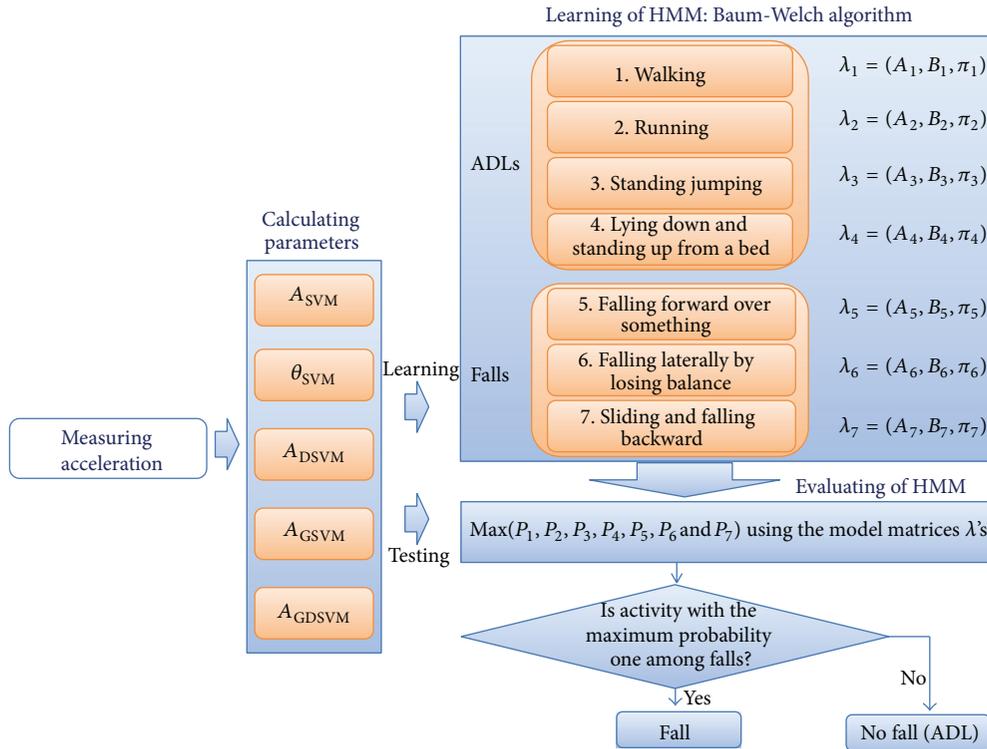


FIGURE 3: Configuration of fall-detection system applying HMM.

- ```

(1) if the parameter > threshold value of the parameter
 then
(2) if θ > threshold value of θ
 (among 100 samples after satisfying the condition in Line 1)
 then
(3) return fall detection
(4) return no fall detection

```

ALGORITHM 1: Simple threshold algorithm using double parameters for fall detection.

$A_{GSVM} = 2.5g$  and  $\theta = 55^\circ$ . These are chosen as threshold values. Table 1 shows falls and ADLs determined with these threshold values. In this table, the total numbers of each activity for subjects A, B, C, and D and subjects E and F are 15 and 10, respectively. All events of the three types of fall are detected as a fall, but 27 events of ADLs are detected as falls instead of ADLs. Subjects C, D, E, and F especially failed to detect several lying-down events (ADL-4) as ADLs, and subjects E and F of over 40 years old also failed to detect a few running and standing jump events as ADLs. It shows that the simple threshold method has a limitation to detect lying-down events of all subjects and running and jumping events of relatively old subjects.

Table 2 shows the falls and ADLs in which the fall events (267 events) chosen from the simple threshold method using double parameters as shown in Table 1 are evaluated by applying the parameter  $\theta$  to the HMM as shown in Algorithm 2, which is the best of fall and ADL detection results applying 5 types of parameters to HMM as shown in Table 3. Instead

of evaluating all 560 events, only the fall events (267 events) chosen from the simple threshold method using double parameters are applied only to the HMM. Computing effort and resources can be saved, compared to applying all the events to the HMM. The sensitivity, specificity, and accuracy obtained from applying the parameter  $\theta$  to the HMM are 99.17%, 99.69%, and 99.5%, respectively. One lying-down event of subject E (over 40 years old) is detected as a fall instead of an ADL, and one forward fall and one backward fall events of subject F (over 50 years old) are detected as ADLs instead of falls. The experimental results of combining the simple threshold with the HMM are higher than those with the simple threshold method only.

#### 4. Conclusions

To detect falls, the fall-detection algorithm combining a simple threshold method and an HMM with 3-axis acceleration was proposed. To apply the proposed fall-detection algorithm

- (1) Calculate all the values in model matrices  $\lambda_i = (A_i, B_i, \pi_i)^*$  of all single parameters for  $i$  activities of four types of ADL and three types of fall by the Baum-Welch learning algorithm [29]
- (2) Calculate all the likelihoods of all single parameters for observation sequences using the evaluation algorithm [29]
- (3) Find a maximum probability among four types of ADL and three types of fall
- (4) **If** the activity with the maximum probability is among three types of fall **then** fall detection
- (5) **Else** no fall (ADL) detection

\*  $A_i, B_i,$  and  $\pi_i$  denote the state transition probability distribution, observation emission probability distribution, and initial state distribution for  $i$  activities, respectively.

ALGORITHM 2: HMM algorithm for fall detection.

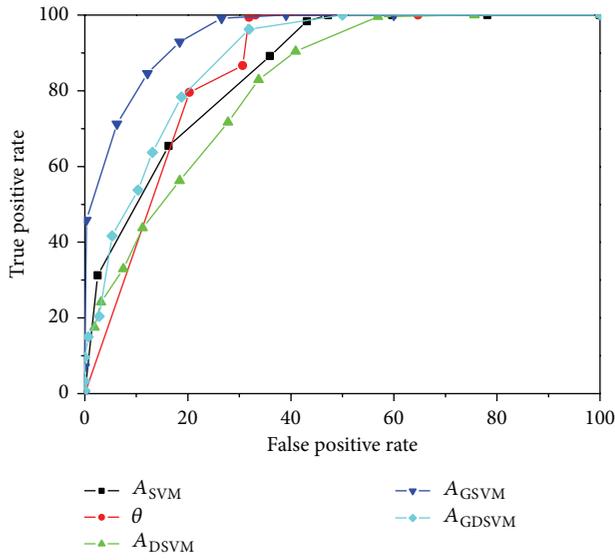


FIGURE 4: ROC curve of fall detection obtained from the simple threshold using the single fall-feature parameter. The best fall detection is of sensitivity 92.92%, specificity 81.56%, and accuracy 88.05% when  $A_{GSVM} = 2.5$  g.

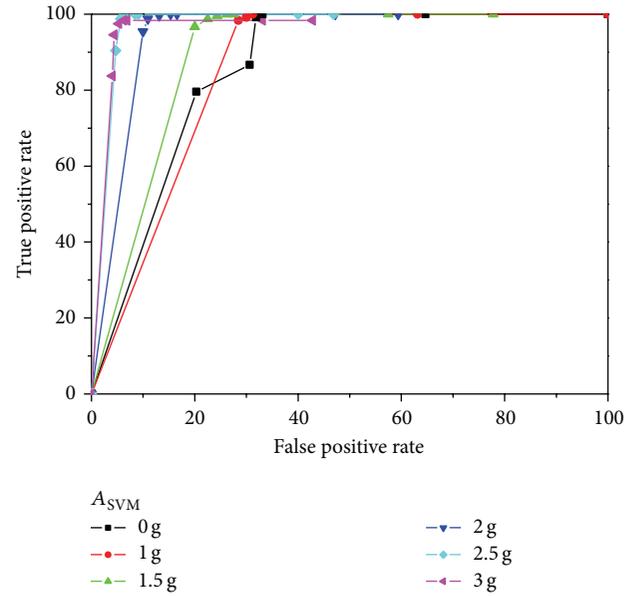


FIGURE 5: ROC curve of fall detection obtained from the simple threshold using the double fall-feature parameters of  $A_{SVM}$  and  $\theta$ . The best fall detection is of sensitivity 98.75%, specificity 94.38%, and accuracy 96.25% when  $A_{SVM} = 2.5$  g and  $\theta = 65^\circ$ .

and detect falls, a wearable fall-detection device was designed and produced. The several fall-feature parameters of the 3-axis acceleration were introduced and applied to the simple threshold method. Possible falls were chosen through the simple threshold and then applied to the HMM to solve the problems such as deviation of interpersonal falling behavioral patterns and similar fall actions. The double parameters  $A_{SVM} = 2.5$  g and  $\theta = 65^\circ$  showed the best fall detection with sensitivity, specificity, and accuracy of 98.75%, 94.38%, and 96.25%, respectively. The best fall detection combining the simple threshold and HMM was of sensitivity 99.17%, specificity 99.69%, and accuracy 99.5% when the threshold values for the simple threshold method were  $A_{SVM} = 2.5$  g and  $\theta = 55^\circ$  and the parameter  $\theta$  was applied to the HMM. These results are higher than those with the simple threshold method using double parameters. Applying only the fall events determined from the simple threshold method to the HMM reduced the computing effort and resources, compared to those of using all the events applied to the HMM. Because the proposed algorithms are simple, they can be implemented into an

TABLE 1: Best fall and ADL detection results obtained from the simple threshold method with the double threshold values  $A_{SVM} = 2.5$  g and  $\theta = 55^\circ$  (sensitivity = 100% and specificity = 91.56% and accuracy = 95.18%).

| Subjects (ages) | ADL-a | ADL-b | ADL-c | ADL-d | Fall-a | Fall-b | Fall-c |
|-----------------|-------|-------|-------|-------|--------|--------|--------|
| A (20s)         | 15    | 15    | 15    | 15    | 15     | 15     | 15     |
| B (20s)         | 15    | 15    | 15    | 15    | 15     | 15     | 15     |
| C (20s)         | 15    | 15    | 15    | 11    | 15     | 15     | 15     |
| D (20s)         | 15    | 15    | 15    | 11    | 15     | 15     | 15     |
| E (40s)         | 10    | 10    | 5     | 7     | 10     | 10     | 10     |
| F (50s)         | 10    | 4     | 6     | 9     | 10     | 10     | 10     |

embedded system such as an 8051-based microcontroller with 128 Kbyte ROM. In the future, if the proposed algorithms are implemented to the embedded system, its performance will be tested in a real time.

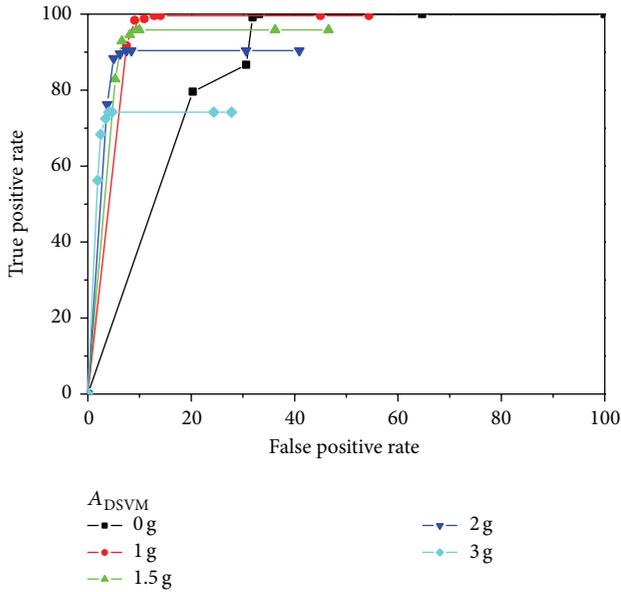


FIGURE 6: ROC curve of fall detection obtained from the simple threshold using the double fall-feature parameters of  $A_{DSVM}$  and  $\theta$ . The best fall detection is of sensitivity 98.33%, specificity 90.94%, and accuracy 95.16% when  $A_{DSVM} = 1\text{ g}$  and  $\theta = 65^\circ$ .

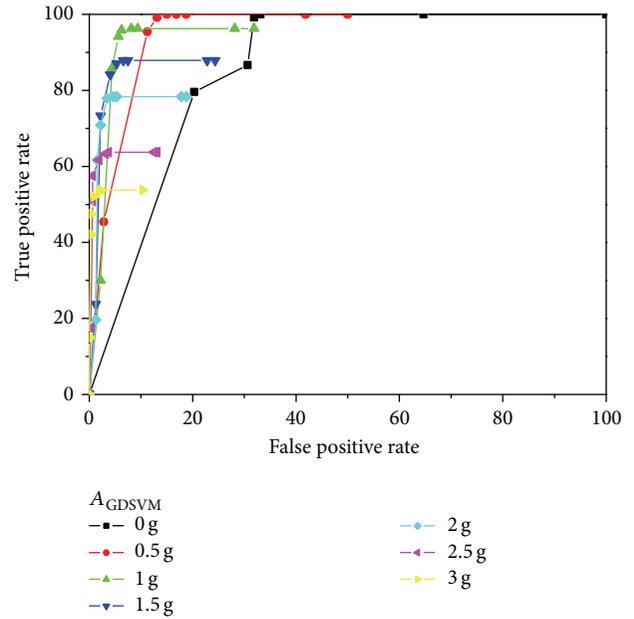


FIGURE 8: ROC curve of fall detection obtained from the simple threshold using the double fall-feature parameters of  $A_{GDSVM}$  and  $\theta$ . The best fall detection is of sensitivity 95.83%, specificity 93.75%, and accuracy 94.94% when  $A_{GDSVM} = 1\text{ g}$  and  $\theta = 60^\circ$ .

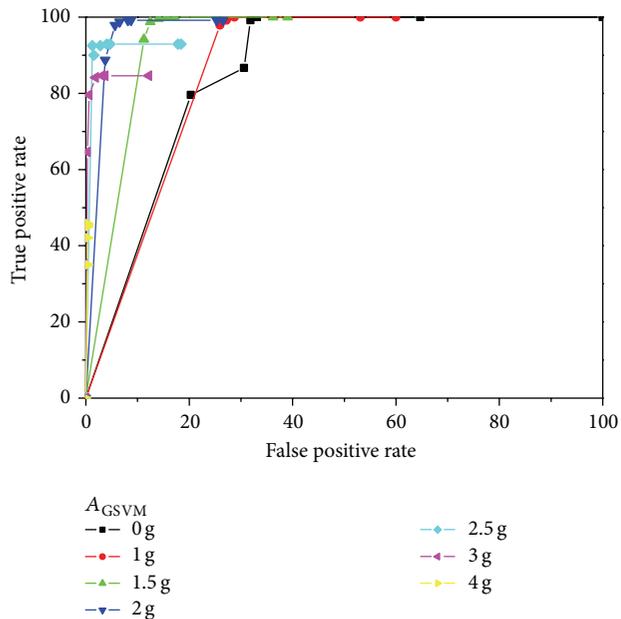


FIGURE 7: ROC curve of fall detection obtained from the simple threshold using the double fall-feature parameters of  $A_{GSVM}$  and  $\theta$ . The best fall detection is of sensitivity 97.92%, specificity 94.38%, and accuracy 96.4% when  $A_{GSVM} = 2\text{ g}$  and  $\theta = 65^\circ$ .

One limitation of this study is that the fall-detection algorithm was tested on subjects in the age group 20–50, falling under simulated conditions. Further research is required to test the performance of the proposed algorithm for detecting actual falls suffered by the elderly.

TABLE 2: Best fall and ADL detection results evaluated by applying the parameter  $\theta$  of the observation samples determined with fall events shown in Table 1 to the HMM algorithm shown in Algorithm 2 (sensitivity = 99.17% and specificity = 99.69% and accuracy = 95.89%).

| Subjects (ages) | ADL-a | ADL-b | ADL-c | ADL-d | Fall-a | Fall-b | Fall-c |
|-----------------|-------|-------|-------|-------|--------|--------|--------|
| A (20s)         | 15    | 15    | 15    | 9     | 15     | 15     | 15     |
| B (20s)         | 15    | 15    | 15    | 4     | 15     | 15     | 15     |
| C (20s)         | 15    | 15    | 15    | 15    | 15     | 15     | 15     |
| D (20s)         | 15    | 15    | 15    | 12    | 15     | 15     | 15     |
| E (40s)         | 10    | 10    | 10    | 9     | 10     | 10     | 10     |
| F (50s)         | 10    | 10    | 10    | 10    | 9      | 10     | 9      |

TABLE 3: Fall detecting results evaluated by combination of the simple threshold method with double parameters and HMM.

|             | $\theta$     | $A_{SVM}$ | $A_{DSVM}$ | $A_{GSVM}$ | $A_{GDSVM}$ |
|-------------|--------------|-----------|------------|------------|-------------|
| Sensitivity | <b>99.17</b> | 97.5      | 99.6       | 99.17      | 99.17       |
| Specificity | <b>99.69</b> | 95.63     | 97.81      | 96.88      | 97.5        |
| Accuracy    | <b>99.5</b>  | 96.43     | 98.57      | 97.86      | 98.21       |

### Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

### Acknowledgment

This work was supported by the GRRRC program of Gyeonggi Province [(GRRRC HANKYONG 2012-B02), Development of

Vision Inspection algorithm and Wireless and Wired Integrated Control System for Intelligent Logistics Inspection].

## References

- [1] C. A. Werner, "The Older Population: 2010," Census Briefs U.S. Bureau of the Census, 2010, <http://www.census.gov/prod/cen2010/briefs/c2010br-09.pdf>.
- [2] B. M. H. Park, J. C. Ha, I. H. Shin et al., "Senior survey 2008: life and welfare service needs of the elderly in Korea," Ministry for Health and Welfare, 2009.
- [3] B. Kaluža and M. Luštrek, "Fall detection and activity recognition methods for the confidence project: a survey," in *Proceedings of the 12th International Multiconference Information Society*, vol. A, pp. 22–25, 2008.
- [4] R. Igual, C. Medrano, and I. Plaza, "Challenges, issues and trends in fall detection systems," *BioMedical Engineering Online*, vol. 12, no. 1, article 66, 2013.
- [5] M. Mubashir, L. Shao, and L. Seed, "A survey on fall detection: principles and approaches," *Neurocomputing*, vol. 100, pp. 144–152, 2013.
- [6] R. Hegde, B. G. Sudarshan, S. C. P. Kumar, S. A. Hariprasad, and B. S. Satyanarayana, "Technical advances in fall detection system—a review," *International Journal of Computer Science and Mobile Computing*, vol. 2, no. 7, pp. 152–160, 2013.
- [7] S.-R. Ke, H. L. U. Thuc, Y.-J. Lee, J.-N. Hwang, J.-H. Yoo, and K.-H. Choi, "A review on video-based human activity recognition," *Computers*, vol. 2, pp. 88–131, 2013.
- [8] L. Atzori, A. Iera, and G. Morabito, "The internet of things: a survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [9] J. K.-Y. Ng, "Ubiquitous healthcare: healthcare systems and applications enabled by mobile and wireless technologies," *Journal of Convergence*, vol. 3, no. 2, pp. 15–20, 2012.
- [10] B. Töreyn, Y. Dedeoglu, and A. Cetin, "HMM based falling person detection using both audio and video," in *Computer Vision in Human-Computer Interaction*, pp. 211–220, Springer, Berlin, Germany, 2005.
- [11] Y. Li, K. C. Ho, and M. Popescu, "A microphone array system for automatic fall detection," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 5, pp. 1291–1301, 2012.
- [12] X. Luo, T. Liu, J. Liu, X. Guo, and G. Wang, "Design and implementation of a distributed fall detection system based on wireless sensor networks," *Eurasip Journal on Wireless Communications and Networking*, vol. 2012, article 118, 2012.
- [13] H. Rimminen, J. Lindström, M. Linnavuo, and R. Sepponen, "Detection of falls among the elderly by a floor sensor using the electric near field," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 6, pp. 1475–1476, 2010.
- [14] M. Kangas, I. Vikman, J. Wiklander, P. Lindgren, L. Nyberg, and T. Jämsä, "Sensitivity and specificity of fall detection in people aged 40 years and over," *Gait & Posture*, vol. 29, no. 4, pp. 571–574, 2009.
- [15] P.-K. Chao, H.-L. Chan, F.-T. Tang, Y.-C. Chen, and M.-K. Wong, "A comparison of automatic fall detection by the cross-product and magnitude of tri-axial acceleration," *Physiological Measurement*, vol. 30, no. 10, pp. 1027–1037, 2009.
- [16] A. Weiss, I. Shimkin, N. Giladi, and J. M. Hausdorff, "Automated detection of near falls: algorithm development and preliminary results," *BMC Research Notes*, vol. 3, article 62, 2010.
- [17] A. K. Bourke, J. V. O'Brien, and G. M. Lyons, "Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm," *Gait and Posture*, vol. 26, no. 2, pp. 194–199, 2007.
- [18] A. K. Bourke and G. M. Lyons, "A threshold-based fall-detection algorithm using a bi-axial gyroscope sensor," *Medical Engineering & Physics*, vol. 30, no. 1, pp. 84–90, 2008.
- [19] Q. Li, J. A. Stankovic, M. A. Hanson, A. T. Barth, J. Lach, and G. Zhou, "Accurate, fast fall detection using gyroscopes and accelerometer-derived posture information," in *Proceedings of the 6th International Workshop on Wearable and Implantable Body Sensor Networks (BSN '09)*, pp. 138–143, Berkeley, Calif, USA, June 2009.
- [20] Y. J. Yi and Y. S. Yu, "Emergency-monitoring system based on newly-developed fall detection algorithm," *Journal of Information and Communication Convergence Engineering*, vol. 11, no. 3, pp. 147–154, 2013.
- [21] T. Zhang, J. Wang, L. Xu, and P. Liu, "Fall detection by wearable sensor and one-class SVM algorithm," in *Intelligent Computing in Signal Processing and Pattern Recognition*, vol. 345 of *Lecture Notes in Control and Information Science*, pp. 858–863, 2006.
- [22] C. Doukas, I. Maglogiannis, P. Tragas, D. Liapis, and G. Yovanof, "Patient fall detection using support Vector Machines," *International Federation for Information Processing*, vol. 247, pp. 147–156, 2007.
- [23] M. Yuwono, B. D. Moulton, S. W. Su, B. G. Celler, and H. T. Nguyen, "Unsupervised machine-learning method for improving the performance of ambulatory fall-detection systems," *BioMedical Engineering Online*, vol. 11, article 9, 11 pages, 2012.
- [24] H. Kerdegari, K. Samsudin, A. R. Ramli, and S. Mokaram, "Evaluation of fall detection classification approaches," in *Proceedings of the 4th International Conference on Intelligent and Advanced Systems (ICIAS '12)*, pp. 131–136, Kuala Lumpur, Malaysia, June 2012.
- [25] J. Cheng, X. Chen, and M. Shen, "A framework for daily activity monitoring and fall detection based on surface electromyography and accelerometer signals," *IEEE Journal on Biomedical and Health Informatics*, vol. 17, no. 1, pp. 38–45, 2013.
- [26] L. Tong, Q. Song, Y. Ge, and M. Liu, "HMM-based human fall detection and prediction method using tri-axial accelerometer," *IEEE Sensors Journal*, vol. 13, no. 5, pp. 1849–1856, 2013.
- [27] N. H. Kim and Y. S. Yu, "Fall recognition algorithm using gravity-weighted 3-axis accelerometer data," *Journal of the Institute of Electronics and Information Engineers*, vol. 50, no. 6, pp. 254–259, 2013.
- [28] M. Kangas, A. Konttila, P. Lindgren, I. Winblad, and T. Jämsä, "Comparison of low-complexity fall detection algorithms for body attached accelerometers," *Gait & Posture*, vol. 28, no. 2, pp. 285–291, 2008.
- [29] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.
- [30] "BMA150 Triaxial acceleration sensor Data sheet," Bosch Sensortec, [http://ae-bst.resource.bosch.com/media/products/dokumente/bma150/bma150\\_flyer\\_rev13\\_14jan2008\\_redlich.pdf](http://ae-bst.resource.bosch.com/media/products/dokumente/bma150/bma150_flyer_rev13_14jan2008_redlich.pdf).
- [31] CC2530: A True System-on-Chip Solution for 2.4-GHz IEEE 802.15.4 and ZigBee Applications, Texas Instruments Incorporated.
- [32] M. A. Fattah, "The use of MSVM and HMM for sentence alignment," *Journal of Information Processing Systems*, vol. 8, no. 2, pp. 301–314, 2012.

- [33] W. O. Odoyo, J.-H. Choi, I.-K. Moon, and B.-J. Cho, "Silhouette-edge-based descriptor for human action representation and recognition," *Journal of Information and Communication Convergence Engineering*, vol. 11, no. 2, pp. 124–131, 2013.
- [34] A. Hussain, A. R. Abbasi, and N. Afzulpurkar, "Detecting & interpreting self-manipulating hand movements for student's affect prediction," *Human-Centric Computing and Information Sciences*, vol. 2, article 14, 2012.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

