SmartFall: An Automatic Fall Detection System Based on Subsequence Matching for the SmartCane

Mars Lan[‡], Ani Nahapetian[*‡], Alireza Vahdatpour[‡], Lawrence Au[§]
William Kaiser[*§], Majid Sarrafzadeh[*‡]
[‡]Computer Science Department [§]Electrical Engineering Department [*]Wireless Health Institute
University of California Los Angeles
Los Angeles, CA 90095
{marslan,ani,alireza,majid}@cs.ucla.edu, {au,kaiser}@ee.ucla.edu

ABSTRACT

Fall-induced injury has become a leading cause of death for the elderly. Many elderly people rely on canes as an assistive device to overcome problems such as balance disorder and leg weakness, which are believed to have led to many incidents of falling. In this paper, we present the design and the implementation of SmartFall, an automatic fall detection system for the SmartCane system we have developed previously. SmartFall employs subsequence matching, which differs fundamentally from most existing fall detection systems based on multi-stage thresholding. The SmartFall system achieves a near perfect fall detection rate for the four types of fall conducted in the experiments. After augmenting the algorithm with an assessment on the peak impact force, we have successfully reduced the false-positive rate of the system to close to zero for all six non-falling activities performed in the experiment.

Keywords

Wireless health, fall detection, subsequence matching, geriatrics

1. INTRODUCTION

In recent years, fall-induced injury has become one of the leading causes of death among elderly people [18]. About one third of people aged over 65 in America fall every year [38], and the statistics is almost three times higher for those living in nursing homes [34]. Common injuries sustained from falls include soft and connective tissue damages, bone fractures, and head injuries [25, 36, 39]. Not only does fall pose a serious threat to the health and well-being of the elderly population, fall-induced injuries can also incur an average annual health care cost of nearly $20,000 for a person's first fall [32], which amounts to an estimated lifetime cost of $12.6b for persons aged 65 and over [31, 35].

Researchers have indicated that gait or balance disorder and leg weakness being the major causes of falling [33]. As a result, a walking stick or a cane is often prescribed as an assistive device to help elderly people overcome these problems [8]. In fact, there are more than 4 million cane users in the United States alone [6]. The evident popularity and relative low cost of the cane make it an ideal candidate for pervasive healthcare. To this end, we have previously developed the SmartCane platform [41], which comprises a traditional-looking cane embedded with numerous sensors and a wireless interface to relay sensor data to a personal device for further processing. The platform has been extended to provide active guidance for self-training of proper cane use [4]. In this paper, we present the design and the implementation of SmartFall—an automatic fall detection system built on top of the SmartCane platform. The goal of this system is to automatically alert the care givers or emergency service when the cane user falls and is unable to reach for help, for example, in the case when the user becomes unconscious.

To our best knowledge, SmartFall is the first fall detection system in its class that employs subsequence matching algorithm, instead of the commonly used thresholding-based method. Thus, the main contribution of this paper is the introduction and evaluation of such a fall detection system. Furthermore, because SmartFall does not require the user to wear or strap sensors to the body directly, it epitomizes a wireless health system that is Pervasive yet Non-Invasive (PNI). At the same time, SmartFall has a significantly higher detection rate and lower cost compared to other forms of PNI fall detection systems, such as infra-red inactivity sensor array [37], vibration-sensing floor [3], and video/audio capture analysis [40]. More importantly, SmartFall does not require specially setup monitoring areas and thus places little restriction on the user's mobility.

The remainder of this paper is organized as follows. Section 2 summaries related work in fall detection algorithms and subsequence matching. Section 3 gives an overview of the SmartCane system, including its hardware and software architectures. In Section 4, we present the design and the implementation details of the SmartFall detection algorithm, which is then evaluated experimentally in terms of fall detection rate and false-positive rate in Section 5. Finally, in Section 6 we conclude the paper and outline possible improvements and future extensions for the SmartFall system.

BodyNets '09 Los Angeles, California USA

2. RELATED WORK

2.1 Thresholding-Based Fall Detection

Many existing fall detection systems are based on thresholding, which involves a continuous comparison of raw or transformed sensor data against pre-defined thresholds [15, 14, 16, 9, 10, 24, 26, 12]. A multi-stage thresholding system uses more than one threshold, and all thresholds must be exceeded in a particular sequence over a certain time period to trigger the final alarm. This kind of system is well understood to outperform its single-stage counterpart by producing lower false-positive rate [15]. For example, in [14] the authors embed a tri-axial accelerometer into a wrist watch and use a three-stage acceleration-impact-inactivity algorithm to detect the falling of the watch-bearer. The norm of the accelerations in all three directions is computed and compared against the threshold defined for each stage. Although the system has an extremely low false-positive, it is found that the detection rate degrades when the falling direction changes, or when rotation is introduced during the fall. A similar system is presented by Hwang and colleagues in [16], where a sensor box is attached to the chest of the testing subject. Apart from the accelerometer, they also use tilt sensor and gyroscope to improve accuracy. It has also been demonstrated in [9, 10] that falls can be distinguished from Activities of Daily Living (ADL) by setting an upper and lower thresholds on either the angular velocity or peak acceleration gathered from sensors attached to the body. Other examples of multi-stage simple threshold-based fall detection algorithm can be found in [24, 26, 12].

One of the major problems of thresholding-based algorithms is illustrated in Figure 1. The two solid lines, Fall and Swing, in the graph are X-axis accelerometer signal acquired using the SmartCane system during a forward fall and a back-and-forth swinging motion respectively. The two dotted lines, TH1 and TH2, denote the two thresholds for stage 1 and 2 respectively. A typical fall signal consists of a sudden acceleration during the free fall period, which triggers TH1, followed shortly by a sudden deceleration during the impact stage, which is detected by TH2. As can be seen in the graph, despite a visually perceivable difference in shape, both signals would be regarded as a valid fall signal using multi-stage thresholding. In this paper, we propose an algorithm based on subsequence matching that focuses on the overall shape of the signal instead of the instantaneous values in the hope to reduce false alarms similar to the case shown here.

2.2 Subsequence Matching

Subsequence matching is a technique commonly employed in data mining and discrete time series analysis to find exact or closely matched segments of a given subsequence (a query) in a much longer sequence (a candidate) [2]. A query of length k is defined as

$$Q = Q_1, Q_2, \ldots, Q_k$$

whereas a candidate of length n is defined as

$$C = C_1, C_2, \ldots, C_n$$

The matching process involves sliding Q along C in the time axis direction and computing a distance metric that is proportional to the dissimilarity between Q and the corresponding segment of C at time t. A commonly used distance metric is Euclidean Distance [19], which is defined as the square of the difference square-rooted,

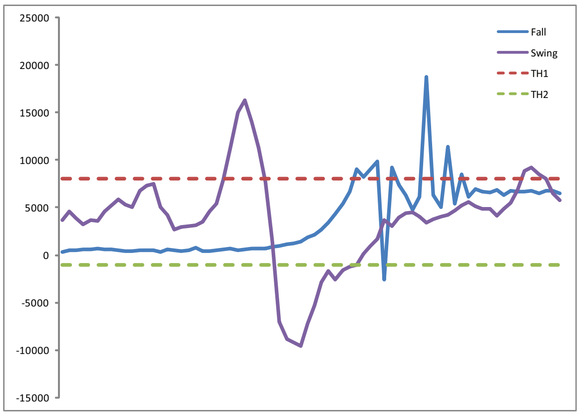

Figure 1: X-axis accelerometer signals for fall & swing acquired on SmartCane

$$D(Q, C)_t = \sqrt{\sum_{i=1}^{k} (Q_i - C_{t+i})^2}$$

Note that both the query and the corresponding candidate segment must be normalized to have a mean of zero and a standard deviation of one before computing Euclidean Distance to produce meaningful results [21].

Euclidean Distance is really a special case of another popular metric known as Dynamic Time Warping (DTW) [7]. DTW maps a point in the query to its closest neighbor in the candidate segment to minimize the effect of phase shifting, data misalignment, and speed difference. While DTW has been successfully applied to speech recognition [17], bioinformatics [1], and fingerprint verification [22], it is not suitable for our application due to the high $O(n^3)$ overall time complexity and the relatively poor performance in discriminating signals that are subtly different.

In the field of data mining and time series analysis, there are often many query and candidate sequences of extremely long length to match, which can easily cause an explosion in computation time. Techniques such as Discrete Fourier Transform (DFT) [13], Discrete Wavelet Transform (DWT) [11], Piecewise Aggregated Approximation (PAA) [42], Singular Value Decomposition [20], and Symbolic Aggregated Approximation (SAX) [23] have been proposed to reduce the data dimensionality. However, in the SmartFall system we have only a limited number of candidate sensor signals. Furthermore, the query sequence of interest, i.e. the actual falling motion, is relatively short in duration. Consequently, we decided to keep the full data precision without applying any dimensionality reduction.

3. THE SMARTCANE SYSTEM

Figure 2 illustrates the system architecture of the SmartCane system. The system consists of 1) a set of low-cost sensors that output signals related to motion, force, and pressure, 2) an acquisition unit that samples the sensor signal and communicates to external devices via a wireless link, and 3) a personal device that collects and processes the data sent from the acquisition unit.

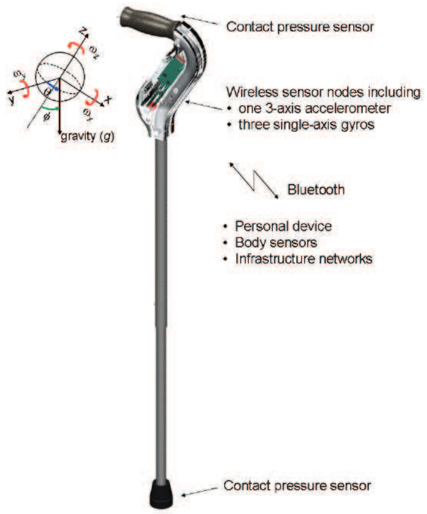

Figure 2: The SmartCane System

3.1 Sensors

The sensors onboard the SmartCane system includes a tri-axial accelerometer [27], three signal-axis gyroscopes [28], and two pressure sensors [30]. The gyroscopes, which are placed perpendicular to each other to measure angular rate in 3D, and the accelerometer are mounted near the handle of the cane with a 30°-slant from the direction of the gravity. The two pressure sensors are fixed at the handle and the tip of the cane, measuring the grip and the downward-pushing force respectively.

3.2 Acquisition Unit

The acquisition unit comprises a MicroLEAP [5] processor and a Bluetooth [29] interface board. Each sensor input channel can be sampled at a rate of up to 300Hz. For our application, a sampling rate of 26Hz is chosen. The unit is very power-efficient and supports over 20 hours of continuous sampling and Bluetooth data streaming using six 2200mAh AA-size batteries.

3.3 Personal Device

The personal device can be any mobile device that support Bluetooth. Although we choose a tablet PC for ease of programming and data visualization, the algorithm can be easily ported to cellphone or PDA. The incoming data is received and logged to a file by a data logging daemon. The SmartFall software then reads directly from the file and performs the detection algorithm in a near real-time fashion.

4. DETECTION ALGORITHM

4.1 A Model for Fall

The mechanism of a fall needs to be first understood in order to develop an effective fall detection algorithm. A typical fall for a cane user consists of a three stages process: 1) collapse, 2) impact, and 3) inactivity. During the collapse stage, the user loses balance and falls towards the ground in an accelerated motion. It is assumed that the cane should experience a similar free-fall process even if the user loses control over the cane. We model this free-fall motion as depicted in Figure 3(a). In this side view, the cane starts from a near upright orientation, topples under the force of gravity, and, just before hitting the ground, changes to a horizontal orientation. The acceleration perpendicular to the cane, denoted by the vector n in the figure, is the norm of the X- and Y-acceleration of the cane (see the top view in Figure 3(b)). Thus, n can be calculated as

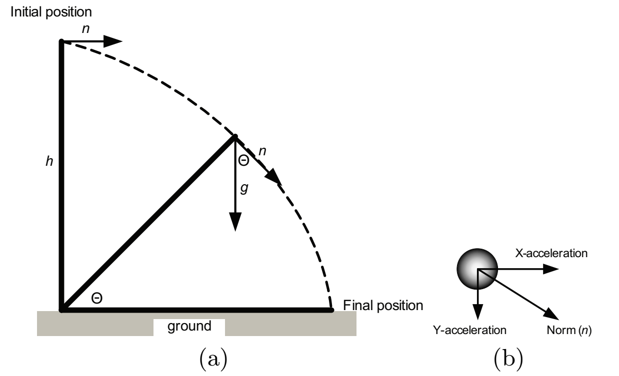

Figure 3: The (a) side view and (b) top view of the typical falling motion of a cane

$$n = g \cdot \cos(\theta) \quad (1)$$

where g is the gravitational acceleration, and θ is the angle between the cane and the ground ($0° \leq \theta \leq 90°$). Given the initial height of the accelerometers h, θ can be expressed as a function of time t

$$\theta = \arcsin(1 - \frac{g}{2h}t^2) \quad (2)$$

Since the accelerometers onboard the SmartCane is tilted by 30°, the actual norm, n' observed is therefore

$$n' = g \cdot \cos(\arcsin(1 - \frac{g}{2h}t^2)) \cdot \cos(30°) \quad (3)$$

which can be simplified to

$$n' = g \cdot \cos(30°) \cdot \sqrt{1 - (1 - \frac{g}{2h}t^2)^2} \quad (4)$$

The impact stage begins when the cane first makes contact with the ground and finishes when the cane becomes motionless. The impact exerts a counter force on the cane that results in a quick deceleration in the opposite direction of the gravity. Depending on the reaction force, the cane may bounce a few times until the energy is completed dissipated. An exact mathematical model for this stage is extremely difficult to derive as the motion depends on many factors, such as the stiffness of the ground, material of the cane, shape of the impacting surface, just to name a few. Consequently, we model the impact stage in an empirical way by collecting and averaging several experiment data.

Assuming the impact has caused a serious injury, both the user and the cane should lie still on the ground for a prolonged period. If the ground surface is flat,

$$n' = g \cdot \cos(30°) \quad (5)$$

for the inactivity stage. This is obviously the easiest stage to model, and the duration of the stage is the only concern. We choose a period of 1 second, which is comparable to that of

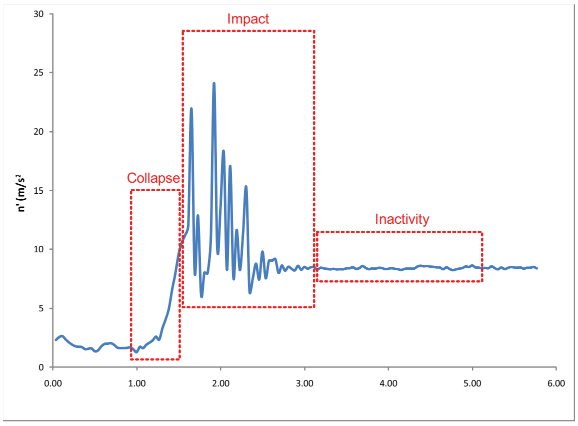

Figure 4: The normal acceleration n' based on the model

the impact stage. This is to ensure that the inactivity stage does not become the dominating factor in the subsequence matching process.

Figure 4 shows the resulting signal for n' based on the model presented here, with the aforementioned three stages highlighted in the figure. The signal is sampled at 26Hz to match the real sensor signal.

4.2 Algorithm Flow

The actual flow of the SmartFall algorithm is depicted in Figure 5. The raw input signals, namely the X- and Y-acceleration from the sensor are first calibrated before the norm is computed. This is then fed through by a signal conditioning module, followed by normalization. The normalized signal is continuously matched against a generated query signal. Based on the calculated distance, the decision module raises the fall alarm if the distance is sufficiently small.

4.2.1 Calibration & Norm Computation

The X- and Y-accelerometer data, shown in Figure 7(a) and 7(b) respectively, have a constant offset that needs to be calibrated in this module. The offset values are obtained from the accelerometers reading when the SmartCane is in a steady up-right orientation.

After the offsets are factored in, the normal acceleration, n' (Figure 7(c)), is calculated using the calibrated X- and Y-accelerations, a_x and a_y as

$$n' = \sqrt{(a_x)^2 + (a_y)^2}$$

4.2.2 Signal Conditioning

The calibrated normal acceleration is then passed through a 6th order biquad IIR low-pass filter with 5Hz cut-off to remove noise and high frequency components. From the filtered signal in Figure 7(d), it is clear that the impacting stage, being highly unpredictable and difficult to model as described earlier, has been effectively smoothen into a single peak. The peak follows closely to the envelope of the original transient signal. The filtering process has been shown experimentally to have slightly decreased the distance of genuine matches and greatly increased the distance of non-matches, which essentially improves the overall signal-to-noise ratio.

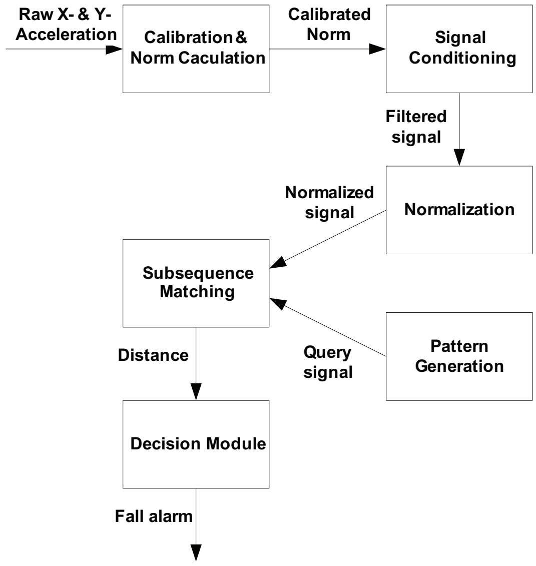

Figure 5: The flow of the SmartFall detection algorithm

4.2.3 Normalization

To facilitate the subsequence matching process, the normalization module buffers the filtered signals and computes a normalized segment of the same length as the query pattern. This basically performs the sliding action described in Section 2.2.

The standard normalization process involves mean shifting, i.e. subtracting the mean from all values, and autoscaling, i.e. diving all values by the standard deviation. However, mean shifting is not necessary in this case because the input signal has already been calibrated for the particular SmartCane in the first module. In fact, mean shifting may even have an adverse effect on the end result, since the mean of the short segment buffered by the normalization module is not a true representative of the mean for the overall input signal. Consequently, we decide to only perform autoscaling with a mean value of zero for the normalization process.

4.2.4 Query Generation

The query is generated statically based on the model presented in 4.1. To facilitate a meaningful comparison with the candidate, the query needs to go through the same filtering and normalization process. Figure 6 shows the filtered and normalized query signal.

4.2.5 Subsequence Matching

The subsequence matching module continuously outputs the Euclidean Distance between the incoming signal and the matching query. An example of the distance computed from this module is plotted as a solid line in Figure 7(e). Typically the distance dips quickly towards zero when a matching incoming signal and the query start to overlap, reaches a local minimum when the two perfectly lined up, and climbs up again as the two signals slide pass each other. The resulting trough is often fairly consistent and salient as can be seen from Figure 7(e).

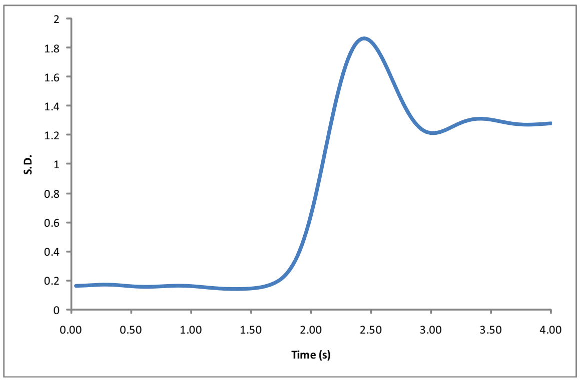

Figure 6: The filtered and normalized query signal

Table 1: Profile of the testing subjects

Subject	1	2	3
Age	35	28	25
Gender	Male	Male	Female
Height	1.86 m	1.60 m	1.65 m
Weight	109.3 kg	63.5 kg	50.8 kg

4.2.6 *Decision*

The decision module is in charge of raising the fall alarm based on the distance value computed from the subsequence matching module. While it is possible to employ sophisticated classification methods such as Artificial Neural Network (ANN) and Supporting Vector Machine (SVM), we are interested if a simple decision tree with one decision node can be as effective. Figure 8 shows the simple decision tree used in this module. The cut-off level is represented by the dashed line in Figure 7(e).

5. RESULTS AND EVALUATION

The performance of SmartFall is measured through a series of experiments. These experiments are designed to evaluate the fall detection rate (true-positive) as well as the ability of the algorithm to discriminate fall from other daily activities (false-positive).

5.1 Experiment Setup

We have selected three healthy test subjects to perform the experiments. Their profiles, including age, gender, height, and weight, are listed in Table 1. Subjects with dissimilar body builds are deliberately chosen to study the potential effect of weight and height on the end results.

A falling platform has been set up for the experiment. The platform is made up of a soft cushion for the subject to fall on, and a hard surface for the cane to hit, simulating what often happens in real life. The position and orientation of the cushion are adjusted to accommodate different types of falls. The sensor data are transmitted via Bluetooth and recorded on a tablet PC. The recoding is not interrupted throughout the process even when the subject completes the instructed action and returns to the starting position.

5.2 Fall Detection Results

Four types of falls, which have been experimented to gauge the fall detection rate of SmartFall, are listed below,

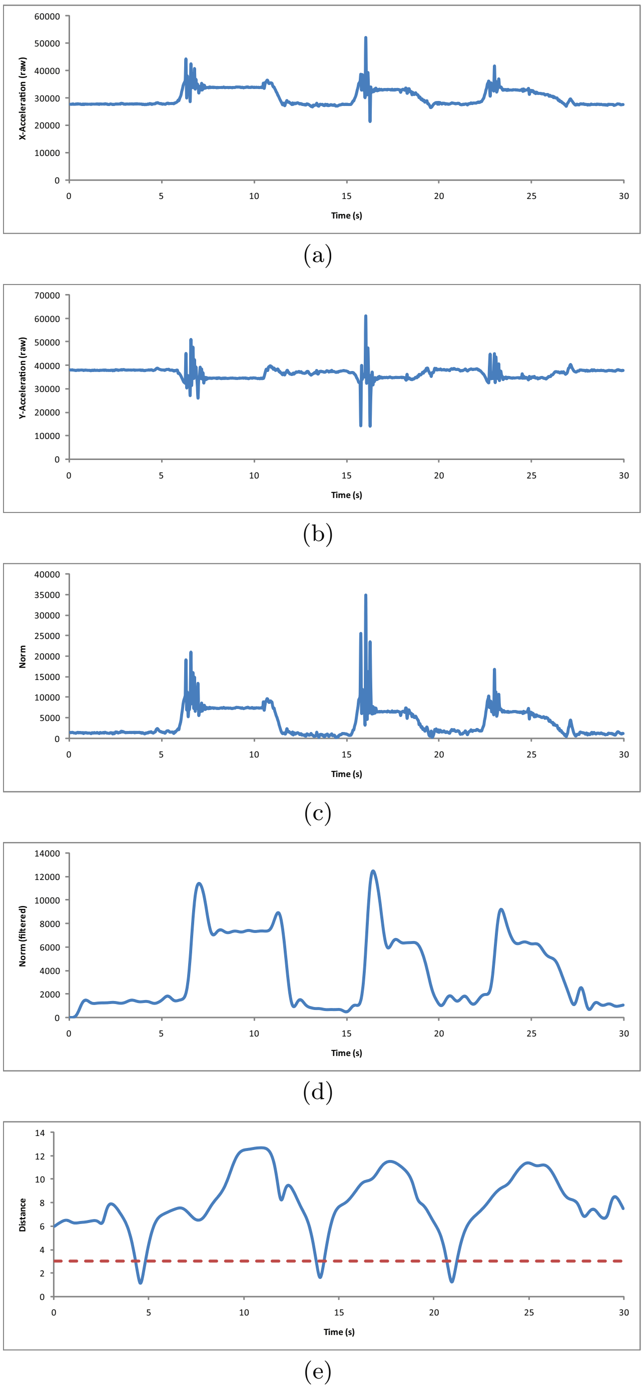

Figure 7: Signal output at various stages: (a) raw X-acceleration (b) raw Y-acceleration (c) Calibrated normal acceleration (d) Filtered normal acceleration (e) Computed Euclidian distance

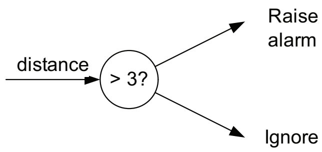

Figure 8: The simple decision tree used in SmartFall

Table 2: Fall detection rate for different type of falls. Figure pairs in the bracket correspond to success/failure counts.

Subject	1	2	3
Forward	100% (30/0)	100% (30/0)	100% (30/0)
Backward	100% (30/0)	100% (30/0)	93.3% (28/2)
Side	100% (30/0)	96.7% (29/1)	100% (30/0)
Free-fall	100% (30/0)	100% (30/0)	100% (30/0)

- Forward: simulates a faced down fall due to trip over
- Backward: simulates fall on the back or bottom due to a slip
- Side: simulates a sideward fall due to loss of balance
- Free fall: simulates an unobstructed topple of the cane due to loss of control. The subject does not fall with the cane in this case.

Each type of fall is performed 30 times, and the subjects are allowed to use their hands to brace against the ground when falling. The subjects can also choose to have a firm grip of the cane throughout the falling process or let go the cane at any time. The results include a fair mix of both situations.

The results listed in Table 2 shows a near 100% detection rate for all four types of fall performed by the three subjects. The difference in weight and height between subjects appears to have little effect on the end results. In the non-detecting case of sideward fall for subject 2, the cane actually hits an obstacle midway through the falling, and the impact is large enough to alter the shape of the signal beyond recognition. As for the two instances when SmartFall fails to pick up subject 3's backward falling, the distance is only about 8% higher than the cut-off level, which can be easily accommodated by tuning decision module slightly without hurting the overall performance. Nonetheless, this suggests a more sophisticated decision may be warranted. In addition, the Free fall experiment shows a limitation of the SmartFall algorithm where it is not possible to determine if the cane user has actually fallen along with the cane. Such shortcoming can be addressed by enforcing a sufficiently long period of inactivity before alarm raised or by allowing the user to cancel the alarm manually.

5.3 Initial Discrimination Results

Discrimination of fall from ordinary daily activities is critical. Frequent false alarms can seriously undermine the user's willingness to adopt a system. The discrimination power of SmartFall is evaluated by using six different experiments made up of commonly performed actions with a cane:

- Slow walk: walking with the cane at a pace less than one step per second
- Fast walk: walking with the cane at a pace around two steps per second
- Sit & stand: standing up with the help of the cane from a sitting position
- Stand still: standing still with a small portion of weight offloaded to the cane

Table 3: False-positive rate for various activities. Figures in the bracket correspond to false-positive counts.

Subject	1	2	3
Slow walk	0% (0)	0% (0)	3.3% (1)
Fast walk	0% (0)	0% (0)	0% (0)
Sit & stand	16.7% (5)	3.3% (1)	16.7% (5)
Stand still	0 in 30s	0 in 30s	1 in 30s
Swing	0% (0)	0% (0)	0% (0)
Lay on lap	96.7% (29)	100% (30)	90% (3)

Table 4: False-positive rate for various activities after applying peak impact assessment

Subject	1	2	3
Slow walk	0% (0)	0% (0)	3.3% (1)
Fast walk	0% (0)	0% (0)	0% (0)
Sit & stand	0% (5)	0% (1)	0% (0)
Stand still	0 in 30s	0 in 30s	0 in 30s
Swing	0% (0)	0% (0)	0% (0)
Lay on lap	10% (3)	3.3% (1)	3.3% (1)

- Swing: swinging the cane back-and-forth at around 1Hz with an angle less than 30° from the vertical axis
- Lay flat: picking up the vertically oriented cane and laying it flat on the lap while seated

All experiments, with the exception of *Stand still*, are performed for 30 complete cycles by the test subjects. The *Stand still* experiment consists of a 30-second period of static standing. The resulting false-positive rates are listed in Table 3.

In general the SmartFall algorithm can discriminate most non-falling actions from fall. For *Fast walk* and *Swing*, the fall alarm has not been triggered at all for all three subjects. However, SmartFall really struggles to tell the difference between laying the cane on lap and a fall, both involved changing the orientation of the cane from vertical to horizontal quickly.

5.4 Improved Discrimination Results

The inability of SmartFall to distinguish laying the cane flat from a genuine fall prompts us to further investigate the problem. As we have found the two actions do produce a very similar profile for the normal acceleration. However, the only difference between the two is the large impact deceleration that exists solely in the case of falling. Unfortunately, this short-living peak deceleration is filtered out by the signal conditioning module. After bringing the unfiltered signal directly to the decision module, the decision tree is modified with an additional condition that a peak normal acceleration in excess of 21 m/s^2 must be also detected within the matching window.

While the detection rate of the system remains unchanged, the false-positive rate, as listed in Table 4, has been dramatically reduced after the addition of peak impact assessment. The SmartFall system is able to achieve zero false-positive for the first five activities. Although false alarms for *Lay on lap* cannot be completely eradicated, they are kept a minimal level in general.

6. CONCLUSIONS AND FUTURE WORK

In this paper, we present SmartFall, an automatic fall detection system based on subsequence matching, for the SmartCane system. SmartFall uses data from the accelerometers embedded closely to the handle of the SmartCane to make inferences of current status. The detection algorithm differs fundamentally from most existing thresholding-based fall detection solutions in that the overall shape of the sensor signal is considered. Several experiments, which simulate various types of fall as well as other commonly performed activities, have been conducted to evaluate the performance of SmartFall. The results have indicated that the algorithm is able to detect almost all cases of falling in the experiment while achieving extremely low false-positive rates for most non-falling activities. However, the subsequence matching algorithm has real difficulty distinguishing an activity that produces a fall-like pattern, such as in the case of rotating the cane from a vertical to a horizontal orientation. To overcome the limitation, we have augmented the algorithm with peak impact assessment. This results in a dramatic decrease in false-positive rates to a near zero level for all non-falling activities while exerting no negative effect on the fall detection rate.

Since the SmartCane has other types of sensors, such as pressure and gyroscope, one of the immediate future research directions is to study the possibility of more accurate inference based on these sensors. We also plan to carry out full scale clinical trial to further validate the effectiveness of SmartFall. Ultimately, we would like to extend SmartFall not only to detect falling, but also to identify patterns and causes that may lead to an imminent fall.

7. REFERENCES

[1] J. Aach and G. Church. Aligning gene expression time series with time warping algorithms. *Bioinformatics*, 17:495–508, |2001|.

[2] R. Agrawal, C. Faloutsos, and A. Swami. Efficient similarity search in sequence databases. In *proceedings of the 4th International Conference on Foundations of Data Organization and Algorithms*, pages 69–84, Chicago, IL, |1993|.

[3] M. Alwan, P. J. Rajendran, S. Kell, D. Mack, S. Dalal, M. Wolfe, and R. Felder. A smart and passive floor-vibration based fall detector for elderly. In *Information and Communication Technologies, 2006. ICTTA '06. 2nd*, volume 1, pages 1003–1007, |2006|.

[4] L. K. Au, W. H. Wu, M. A. Batalin, and W. J. Kaiser. Active guidance towards proper cane usage. In *BSN 2008*, Hong Kong, |2008|.

[5] L. K. Au, W. H. Wu, M. A. Batalin, D. H. McLntire, and W. J. Kaiser. Microleap: energy-aware wireless sensor platform for biomedical sensing applications. In *Biomedical Circuits and Systems Conference, 2007. BIOCAS 2007. IEEE*, pages 158–162, |2007|.

[6] H. Bateni and B. Maki. Assistive devices for balance and mobility: benefits, demands, and adverse consequences. *Archives of Physical Medicine and Rehabilitation*, 86:134–145, |2005|.

[7] D. Berndt and J. Clifford. Using dynamic time warping to find patterns in time series, |1994|.

[8] W. Blount. Don't throw away the cane. *Journal of Bone & Joint Surgery*, 38:695–708, |1956|.

[9] A. Bourke and G. Lyons. A threshold-based fall-detection algorithm using a bi-axial gyroscope sensor. *Medical Engineering & Physics*, 30:84–90, |2006|.

[10] A. Bourke, J. O'Brien, and G. Lyons. Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm. *Gait & Posture*, 26(2):194–199, |2004|.

[11] K. Chan and F. A. W. Efficient time series matching by wavelets. In *proceedings of the 15th IEEE International Confernce on Data Engineering*, pages 126–133, Sydney, Australia, |1999|.

[12] J. Chen, K. Kwong, D. Chang, J. Luk, and R. Bajcsy. Wearable sensors for reliable fall detection. In *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the*, pages 3551–3554, |2005|.

[13] K. Chu and M. Wong. Fast time-series searching with scaling and shifting. In *proceedings of the 18th ACM Symposium on Principles of Database Systems*, pages 237–248, Philadelphia, PA, |1999|.

[14] T. Degen, H. Jaeckel, M. Rufer, and S. Wyss. Speedy: a fall detector in a wrist watch. In *Proceedings of the 7th IEEE international Symposium on Wearable Computers7th IEEE international Symposium on Wearable Computers*, page 184, Washington, D. C., |2003|.

[15] K. Doughty, R. Lewis, and A. McIntosh. The design of a pratical and reliable fall detector for community and institutional telecare. *Journal of Telemedicine and Telecare*, 6:S1:150–S1:154, |2000|.

[16] J. Hwang, J. Kang, Y. Jang, and H. Kim. Development of novel algorithm and real-time monitoring ambulatory system using bluetooth module for fall detection in the elderly., |2004|.

[17] F. Jelinek. *Statistical methods for speech recognition.* The MIT Press, Cambridge, MA, |1998|.

[18] P. Kannus, J. Parkkari, S. Niemi, and M. Palvanen. Fall-induced deaths among elderly people. *American Journal of Public Health*, 95(3):422–434, |2005|.

[19] E. Keogh, K. Chakrabati, M. Pazzani, and S. Mehrota. Dimensionality reduction for fast similarity search in large time series database. *KAIS Journal*, 3:263–286, |2001|.

[20] E. Keogh, K. Chakrabati, M. Pazzani, and S. Mehrota. Locally adaptive dimensionality reduction for indexing large time series databases. In *proceedings of ACM SiGMOD conference on management of data*, pages 151–162, Santa Barbara, CA, |2001|.

[21] E. Keogh, S. Lonardi, and C. Ratanamahatana. On the need for time series data mining benchmarks: A survey and empirical demonstration. In *proceeding of SIGKDD*, pages 102–111, |2004|.

[22] Z. Kovacs-Vajna. A fingerprint verification system based on triangular matching and dynamic time warping. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1266–1276, |2000|.

[23] J. Lin, E. Keogh, S. Lonardi, and B. Chiu. A symbolic representation of time series, with implications for streaming algorithms, |2003|.

[24] U. Lindemann, A. Hock, M. Stuber, W. Keck, and C. Becker. Evaluation of a fall detector based on accelerometers: A pilot study. *Medical and Biological*

Engineering and Computing, 43(5):548–551, |2005|.

[25] S. Lord, C. Sherrington, and H. Menz. *Falls in older people: risk factors and strategies for prevention.* Cambridge University Press, |2001|.

[26] S. Luo and Q. Hu. A dynamic motion pattern analysis approach to fall detection, |2004|.

[27] Analog Devices, Inc. ADXL330: small, low power, 3-axis $\pm$3g imems accelerometer. `http://www.analog.com/en/mems-and-sensors/imems-accelerometers/adxl330/products/product.html`.

[28] Analog Devices, Inc. ADXRS150: $\pm 150^\circ$/s single chip yaw rate gyro with signal conditioning. `http://www.analog.com/en/mems-and-sensors/imems-gyroscopes/adxrs150/products/product.html`.

[29] Roving Networks. BlueSentry-XPert direct connect installation guide. `http://www.rovingnetworks.com/documents/BlueSentry-install.pdf`.

[30] TekScan, Inc. Flexiforce force sensors. `http://www.tekscan.com`.

[31] D. Rice, E. MacKenzie, A. JONES, S. Kaufman, G. deLissovoy, W. Max, E. McLoughlin, T. Miller, L. Robertson, D. Salkever, and G. Smith. *Cost of injury in the United States: a report to congress.* San Francisco, CA: Institute for Health and Aging, University of California, San Francisco and Baltimore, MD: Injury Prevention Center, The Johns Hopkins University, San Francisco, CA and Baltimore MD, |1989|.

[32] J. A. Rizzo, R. Friedkin, C. S. Williams, J. Nabors, D. Acampora, and M. E. Tinetti. Health care utilization and costs in a medicare population by fall status. *Medical Care*, 36(8):1174–1188, |1998|.

[33] L. Rubenstein. Falls in older people: epidemiology, risk factors and strategies for prevention. *Age and Ageing*, 35:ii37–ii41, |2006|.

[34] L. Rubenstein, K. Josephson, and A. Robbins. Falls in the nursing home. *Annals of Internal Medicine*, 121:442–451, |1994|.

[35] J. Runge. The cost of injury. *Emergency Medicine Clinics of North Ameraica*, 11(1):241–253, |1993|.

[36] A. Salva, I. Bolibar, G. Pera, and C. Arias. Incidence and consequences of falls among elderly people living in the community. *Med Clin (Barc)*, 122(5):172–176, |2004|.

[37] A. Sixsmith and N. Johnson. A smart sensor to detect the falls of the elderly. *Pervasive Computing, IEEE*, 3(2):42–47, |2004|.

[38] M. Tinetti and M. Speechley. Prevention of falls among the elderly. *New England Journal of Medicine*, 320:1055–1059, |1989|.

[39] M. Tinetti and C. Williams. Falls, injuries due to falls, and the risk of admission to a nursing home. *New England Journal of Medicine*, 337(18):1279–1284, |1997|.

[40] B. U. Toreyin, Y. Dedeoglu, and A. E. Cetin. Hmm based falling person detection using both audio and video. In *Signal Processing and Communications Applications, 2006 IEEE 14th*, pages 1–4, |2006|.

[41] W. Wu, L. Au, B. Jordan, T. Stathopoulos, M. Batalin, W. Kaiser, A. Vahdatpour, M. Sarrafzadeh, M. Fang, and J. Chodosh. The smartcane system: an assistive device for geriatrics, |2008|.

[42] B. Yi and C. Faloutsos. Fast time sequence indexing for arbitrary lp norms. In *proceedings of the 26th International Conference on Very Large Databases*, pages 385–394, Cairo, Egypt, |2000|.