Fall Detection

Periklis Korkontzelos (pk1913), Imperial College London

I. Abstract

Fall detection is an area of high interest due to the increased number of accidents over the years. While various techniques have been optimized for elder use, there is not much literature on cycling falls. This report outlines how representative cycling data will be extracted, processed and classified for the development of a smart helmet. A metric is proposed to assess the performance of different algorithms and future work and improvements are listed.

II. Introduction

Cycling is becoming an increasingly popular mode of transport for people, but at the same time accidents induced on them become more frequent [1]. According the Royal Society for Prevention of Accidents, about 22,000 accidents occur in the UK every year, 0.5% of which result in fatality [2]. In certain cases, the nature of the accident leads to instantaneous death, in others however, it is the result of a late response and complications that arise thereafter. It has been shown that mortality following a fall can significantly be reduced if an alarm system is in place notifying a caregiver that can provide immediate assistance [3].

Despite the extended research performed in the area of fall detection, an ideal system has not been developed due to several challenges involved in the acquisition of data and training of algorithms. A fall is a rare event that can occur in diverse ways [4]. Thus, it is not easy to collect a sufficiently large dataset covering all possible cases. The most common approach is to gather data from volunteers performing activities of daily life (ADL) [5]. As illustrated by the developers of Sisfall [6], a refined dataset of ADLs, there can be significant error caused due to the variation of training and true data. In the context of Sisfall, this was shown in the different data generated from young and elder individuals; yet this distinction is also present in this case, between ADLs and falls by cyclists and pedestrians. In addition to variation error, there might be some error introduced during the acquisition of training data [5]. This is mainly because the laboratory setting in which data are gathered may not be representative of true events. For instance, most fall data are acquired by a volunteer dropping on a mattress, which has different properties than the ground and is likely to affect readings during impact. Nevertheless, there have been developed several fall detection systems that overcome these challenges and achieve accuracies higher than 90%.

III. Related Work

Most fall detection systems can be identified as a wearable device, an ambience device or a vision based system. While wearable and ambient devices offer low cost and complexity, vision based solutions are more robust for detecting a fall [7].

The most common implementation includes the use of accelerometer and gyroscope, yet this is not always the case. Xiang and Gong (2005) developed a system that detects abnormal behaviours, such as a fall, through image processing [8]. Zigel, Litvak and Gannot (2009), revised a method of automatic fall detection using sounds and vibration sensors placed on the floor [9].

Some systems use multiple sensors to improve accuracy. Su, Liu and Wu (2016), managed to improve fall detection accuracy by combining data from two accelerometers, on the waist and thigh [10]. Different sensors could also be used, such as an ECG monitor to provide information about the user's vital signs [11].

IV. Implementation

As the solution being developed will be integrated on a helmet, the most sensible choice is an accelerometer-based solution. It has been shown that in order to distinguish a fall from activities of daily life (ADL) it is necessary to use both a motion (accelerometer) and posture (gyroscope) data [12]. The simplest way to implement a fall detection algorithm is through threshold detection which is computationally inexpensive. A more refined solution allows for programmable thresholds, depending on age, weight, height and gender [13]. However, it can be hard to generalize threshold parameters for different individuals and generally, threshold detection algorithms do not offer a good trade-off between false positive and false negative readings [14]. Therefore, machine learning techniques performing data acquisition, feature extraction and event classification will be used instead, as they have been used successfully to distinguish between falls and ADLs [15].

Data Acquisition

As explained earlier, acquiring data that are representative of real world situations can be problematic. One of the most important design considerations is the placement of the sensor module. The body parts yielding the most consistent data are the waist, the chest and head due to their relative lack of motion during ADLs [16]. Wrist-worn devices are user friendly but are not suitable for detecting falls due to the wide variety of movements of the arm. Therefore, for this project the sensors will be placed on the back of the head which is close to the neck that does not experience sudden motion patterns during ADLs.

To facilitate the development of the fall detection algorithms, several databases were used to test basic functionalities. There are several available databases such as the Sisfall, tFall and MobiFall [17]. These datasets include accelerometer and gyroscope datasets gathered from individuals performing ADLs and falls. While these data are useful in identifying the necessary features and prototyping classification algorithms, they do not suffice to fully train the algorithms as they are not representative of the conditions applied while cycling. The motions and posture of a cyclist are different to that of a person walking so a variation is expected in the accelerometer and gyroscope readings both during ADLs and falls. Also, due to the higher momentum of a cyclist, the acceleration pattern tends to have a higher magnitude and more noise.

While fall detection is popular topic, there does not exist much literature on motion and posture data while cycling. To gather more representative data of normal cycling operation and falls from a bicycle, a prototype was developed. A 9-axis motion processing unit (MPU-9250) was connected to an Arduino Mega which was externally power by a 9V battery for wireless operation. Data were communicated from the sensor to the processor using I2C and at a flick of a switch an interrupt triggered the writing of data for 20 seconds to a file in a micro SD card, which was also connected to the Arduino. This allowed the acquisition of more representative cycling data during activities such as riding straight, turning, starting/stopping suddenly and going up or down a pavement. To gather fall data the prototype was attached to a dummy for health and safety reasons. Someone was cycling holding the dummy and dropped it to simulate a fall. The apparatus used for custom data acquisition is shown in Figure 1.

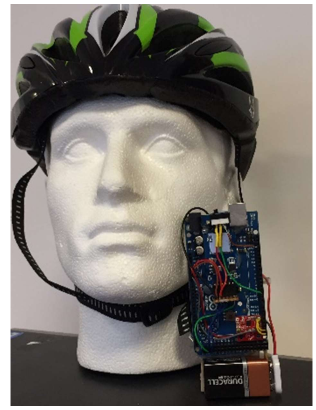

Figure 1 - Cycling Fall Apparatus

Feature Extraction

In order to extract features that can accurately distinguish a fall from ALDs, it is important to gain an understanding over the nature of the problem. A fall is an event that last for 1-2 seconds [18]. Thus, it is important to select an appropriate sampling rate and window to ensure no event is missed.

In related literature, there are a lot of different features that have been used with varying accuracy, sensitivity, specificity and complexity. The most common feature which is found in all fall detection systems is the overall acceleration experienced by the user, $a(t)$, defined by equation (1), where $a_x(t)$, $a_y(t)$, and $a_z(t)$ are the acceleration components in x, y and z direction at time t, respectively.

$$a(t) = \sqrt{a_x(t)^2 + a_y(t)^2 + a_z(t)^2} \qquad (1)$$

More features that can be extracted include the following: maximum, minimum and peak-to-peak magnitude, absolute and change in orientation, jerk, statistical properties such as mean and standard deviation and area related features. However, the latter require integration and can be computationally expensive. A more detailed set of features can be found in [19] and [6].

Selecting the appropriate features is an important task in achieving the best performance possible. Since all features are extracted from one data source a high degree of correlation is expected between certain features. An algorithm is proposed in [19] to select the most appropriate features based on their information gain. The algorithm's procedures are the following: a) rank features by ascending gain ratio, b) for each feature compute its correlation with all other features and remove it if at least one coefficient is higher than 0.8 and c) Repeat b) until no features are removed.

Event Classification

Event classification consists of a binary problem with only two possible outcomes, fall or no-fall. Several machine learning algorithms can be implanted to distinguish between the two events such as Decision Trees, , Support Vector Machines and K-Nearest Neighbor [15]. Some more refined techniques include the use of neural networks [20], Kernel Principal Component Analysis (KPCA) and a method of improved KPCA [21].

V. Results

In assessing the performance of fall detection algorithms, 5 parameters will be used including accuracy, sensitivity, specificity, precision and the F-measure [22]-[23]. While all these characterize the performance of the algorithm their importance might vary, depending on the classification problem. For fall detection algorithms, false negatives imply that a fall has not not detected and are much more severe than false positives which only cause discomfort to the user.

Using the datasets available online, some initial testing was performed to gain understanding over the most relevant features and suitable classification technique. Results indicate that the most dominant features for detecting a fall include: raw accelerometer data, overall acceleration, overall angular speed and orientation. MATLAB was used to process these features and identified Quadratic SVM, Medium Gaussian SVM and Medium Tree as the best approaches, yielding the highest accuracies. However, more simulations should be performed using the true cycling data to reevaluate the choices of features and classification techniques.

VI. Improvements & Future Work

Current data are not sufficient to accurately distinguish between normal cycling operation and falls. Thus, the next stage involves the formation of a large dataset of cycling data which will be used to refine the fall detection algorithms.

There are techniques that can be implemented to improve performance. It has been shown that preprocessing data using a Kalman filter to reduce noise at the input improves most features related to fall detection but may deteriorate the performance of some [24].

The power requirements for the system can also be revised. Power can be saved by putting the processor in sleep mode during ADLs and trigger an interrupt in the case of a fall using a feature that wakes it up [25]. Also, the power requirement of a feature, which varies along computational complexity, could be considered in feature selection to relax the requirements [26]. However, as noted by [27] there tends to be a trade-off between power consumption and accuracy.

VII. Conclusion

This report outlines the different steps required in the identification of a fall. It underlines the significance and challenges involved and covers similar technologies. It also explains how data will be acquired, processed and classified and how the performance of the developed algorithms will be assessed. Finally, the next stages and possible future improvements are listed.

VIII. Bibliography

[1] Allan, C. (2017) Cycling UK's Cycling Statistics, Cycling UK, Available from: http://www.cyclinguk.org/resources/cycling-uk-cycling-statistics#Is cycle use increasing in Britain?

[2] ROSPA (2014) Cycling Accidents Facts & Figures. Royal Society for Prevention of Accidents, Available from: http://www.rospa.com/road-safety/advice/pedal-cyclists/facts-figures/

[3] Bagala, F., Becker, C., Cappello, A., Chiari, L., Aminian, K., Hausdorff J. M., (2012) Evaluation of

accelerometer-based fall detection algorithm in realworld falls. *PLoS ONE,* 7(5), pp. 1-9.

[4] Stone, E. and Skubic, M. (2015). Fall Detection in Homes of Older Adults Using the Microsoft Kinect. *IEEE Journal of Biomedical and Health Informatics*, 19(1), pp.290-301.

[5] Khan, S. and Hoey, J. (2017). Review of fall detection techniques: A data availability perspective. *Medical Engineering & Physics*, 39, pp.12-22.

[6] Sucerquia, A., López, J. and Vargas-Bonilla, J. (2017). SisFall: A Fall and Movement Dataset. *Sensors*, 17(1), p.198.

[7] Mubashir, M., Shao, L. and Seed, L. (2013). A survey on fall detection: Principles and approaches. *Neurocomputing*, 100, pp.144-152.

[8] Xiang, T., Gong, S., (2005) Video behaviour profiling and abnormality detection without manual labeling. *Procceedings of ICCV. IEEE Computer Society*. pp. 1238-1245

[9] Zigel, Y., Litvak, D., Gannot, I., (2009) A Method for Automatic Fall Detection of Elderly People Using Floor Vibrations and Sound – Proof of Concept on Human Mimicking Doll Falls. *IEE Transactions on Biomedical Engineering*, 56(12): pp2858-2867.

[10] Su, Y., Liu, D., Wu, Y. (2016) A Multi-sensor Based Pre-impact Fall Detection System With a Hierarchical Classifier. *9th International Congress in Image and Signal Processing*, BioMedical Engineering and Informatics.

[11] Horta, E. T., Lopes, I. C., Rodrigues, J. J. P. C., Proenca, M. L. (2013) A Mobile Health Application for Falls Detection and Biofeedback Monitoring. *IEE International Conference on e-Health Networking, Applications & Services,* pp36-40.

[12] Weng, W. X., Lo, S. C., (2016) Fall Detection Based on Tilt Angle and Acceleration Variations. *IEE Trustcom/BigDataSE/ISPA*.

[13] Zheng, J., Zhang, G., Wu, T., (2009) Automatic Fall Detector for Elderly Based on Triaxial Accelerometer. *IEEE International Conference on Bioinformatics and Biomedical Engineering*, pp. 1-4.

[14] Igual, R., Medrano, C. and Plaza, I. (2013). Challenges, issues and trends in fall detection systems. *BioMedical Engineering OnLine*, 12(1), p.66.

[15] Hsieh, C., Huang, C. Liu, K., Chu, W., Chan., C. (2016) A machine learning approach to fall detection algorithm using wearable sensor. *Proceeding of the IEE International Conference on Advanced Materials for Science and Engineering.*

[16] Bagnasco, A., Scapolla, A. M., Spasova, V. (2011) Design, implementation and experimental evaluation of a wireless fall detector. *Proceedings of the 4th International Symposium on Applied Sciences in Biomedical and Communication Technologies*. Article 65.

[17] Biomedical Informatics & eHealth Laboratory. (2017) The MobiFall & MobiAct Datasets. Available from: http://www.bmi.teicrete.gr/index.php/research/ mobiact.

[18] Yu, X., (2008) Approaches and principles of fall detection for elderly and patients. *IEE International Conference on e-Health Networking, Applications & Services*.

[19] Gjoreski, M., Gjoreski, H., Luštrek, M. and Gams, M. (2016). How Accurately Can Your Wrist Device Recognize Daily Activities and Detect Falls?. *Sensors*, 16(6), p.800.

[20] Abbate, S., Avvenuti, M., Bonatesta, F., Cola, G., Corsini, P. and Vecchio, A. (2012). A smartphone-based fall detection system. *Pervasive and Mobile Computing*, 8(6), pp.883-899.

[21] Wang, H., Chen, X., Chen, X., Zang, P., Liu, Y., Wang, L. (2016) Fall Detection Based on KPCA and 3D KPCA. *2016 IEEE Trustcom/BigDataSE/ISPA.*

[22] Powers, D.M. W., (2011). Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation. Journal of Machine Learning Technologies. 2 (1): 37–63.

[23] Chawla, N. V., (2009) Data mining for imbalanced datasets: An overview. Data mining and knowledge discovery handbook. *Springer*. pp. 875-886.

[24] He. J., Zhou, M., Wang, X., Han, Y. (2016) A Wearable Method for Autonomous Fall Detection Based on Kalman Filter and k-NN Algorithm. *2016 IEEE Biomedical Circuits and Systems Conference.*

[25] Jian Yuan, Kok Kiong Tan, Tong Heng Lee, and Koh, G. (2015). Power-Efficient Interrupt-Driven Algorithms for Fall Detection and Classification of Activities of Daily Living. *IEEE Sensors Journal*, 15(3), pp.1377-1387.

[26] Wang, C., Redmond, S., Lu, W., Stevens, M., Lord, S. and Lovell, N. (2017). Selecting Power-Efficient Signal Features for a Low-Power Fall Detector. *IEEE Transactions on Biomedical Engineering*, pp.1-1.

[27] Saeedi, R., Schimert B., Ghasemzadeh H. (2014) Cost-sensitive feature selection for on-body sensor localization. *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. pp. 833-842.