Customisable Fall Detection: A hybrid approach using physics based simulation and machine learning

Georgios Mastorakis[1]
g.mastorakis@ieee.org
Xavier Hildenbrand[2]
hildenbrand.xavier@gmail.com
Kevin Grand[2]
kevin.grand43@hotmail.fr
Dimitrios Makris[1]
d.makris@kingston.ac.uk

[1]Human Body Motion Group
Digital Information Research Centre
School of Computer Science and Mathematics
Kingston University London
[2]Télécom Physique Strasbourg
Université de Strasbourg

Abstract

This study proposes an approach for a customisable fall detection methodology using physics-based simulation, to address the issue of scarcity of recorded data required for training. Information such as the height and the pre-fall orientation of a human body is derived by a calibrated depth camera and then used to initialise a myoskeletal model used in simulation. Such a customised simulated fall can then be combined with a training dataset to improve the accuracy of modelling fall events for the specific person and situation. Our approach is evaluated using a single feature, the vertical velocity of the centre of mass in a dataset of 132 videos. The results show a close correlation between actual and simulated falls.

1 Introduction

Plenty of studies on fall detection have been published in recent years, driven by the need of monitoring for independent living applications and detecting accidents as well as the availability of cheap and easy to use depth cameras. Usually, fall detection methods are tuned either by ad-hoc empirical approaches or by machine learning algorithms. Both approaches require video recordings of falls that are normally staged and performed by volunteers or actors. Nevertheless, human subjects may hesitate to perform a fall and also the acting of a fall might be directed in such a way that is not realistic, or similar to an actual fall event, e.g. fainting [4]. Another issue is the fact that every human has different anatomical build and as a result the dynamics of the fall may differ accordingly. In a trainable algorithm the requirement is to have a dataset sufficiently large, not only to cover the data requirements of the machine learning algorithm but also to properly cover a range of people's anatomies and fall types. Somehow, existing datasets [6] for fall detection are based on a small number of human subjects with limited body variability in sex, age, height and weight distribution. Therefore it is questionable whether the results can be generalised to the wider population. On the other side, a physics-based simulation provides the opportunity for customising the activity based on the body characteristics and/or the environment [1].

Our aim is to address the issue of scarcity of training data and improve existing fall detection algorithms by customising fall events using physics simulations. The proposed methodology extracts the person's height and pre-fall body orientation from calibrated depth cameras to customise a simulated fall that is then used to improve the accuracy of the fall event model.

2 Physics-Based Simulation

A rough approximation of a falling person may be provided by simulating a falling stick of height L with uniform mass distribution. The following formulas show the angular velocity ω and the velocity of the centre of mass V_y of the stick, where the subscript $_{pre}$ is used to indicate the previous observation (either ω or θ):

$$\omega = \sqrt{\omega_{pre}^2 + \frac{3g(\sin(\theta_{pre}) - \sin(\theta))}{L}} \tag{1}$$

$$V_y = \frac{L}{2}\omega\cos(\theta) \tag{2}$$

A more precise approximation is derived by OpenSim [2], an open-source simulation software initiated by Stanford University. A number of different pre-defined myoskeletal models are available for OpenSim. In Figure 1 the simulated model is observed to be comparable with the falling behaviour of a person.

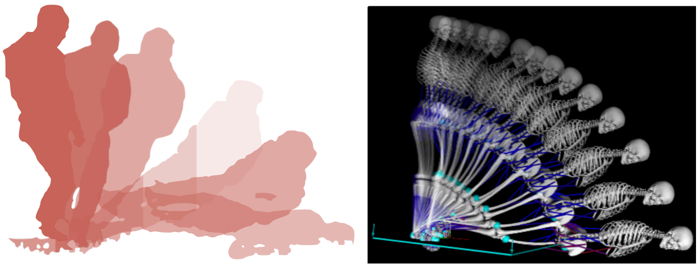

Figure 1: Phases of an actual fall event as captured by a depth camera and of a fall simulated by OpenSim

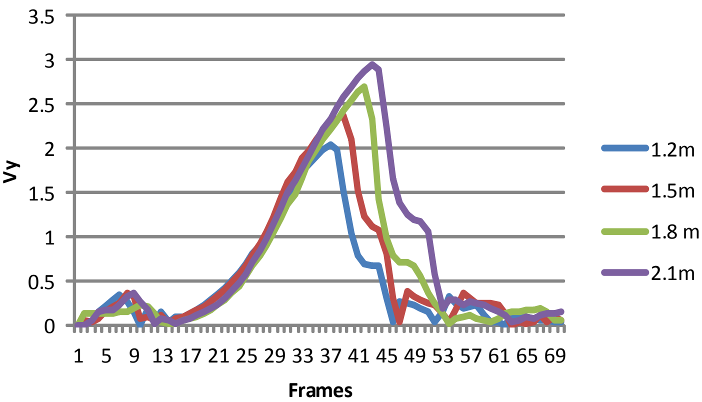

Figure 2: Vy profiles of an OpenSim model performing a backward fall: Notice the analogy between height - velocity

The dynamics of the fall are quantified by the vertical velocity of the centre of mass V_y. A number of simulations assuming different model heights are shown in Figure 2. Both Eq. 2 and these results agree that the taller the human body, the higher the maximum of the final vertical fall velocity.

Figure 3 demonstrates the dynamics of different falls, based on the body/myoskeletal model orientation of the fall (forward, backward, sideways), as derived by OpenSim. These dynamics are related to the balancing forces, affected mainly by the support of the feet which differs depending on the fall orientation, e.g. maximal feet support in forward falls and minimal in backward falls. Another observation is that the backward and sideways falls have a steady velocity profile, while in the forward fall the velocity increases halfway between zero and maximum velocity.

3 Detection Algorithm

The proposed detection algorithm considers the height of the person as estimated by the calibrated depth camera and the orientation [3] of the person before the fall. An OpenSim simulation is performed for each type of fall and for the specific model height as discussed previously and the $V_y(t)_{(pol)}$ velocity profile is fitted using a polynomial regression algorithm (Eq. 3). After fitting the simulated data, the measured vertical velocity $V_y(t)_{(meas)}$ of the centre of mass is considered to estimate the error ε (Eq. 4).

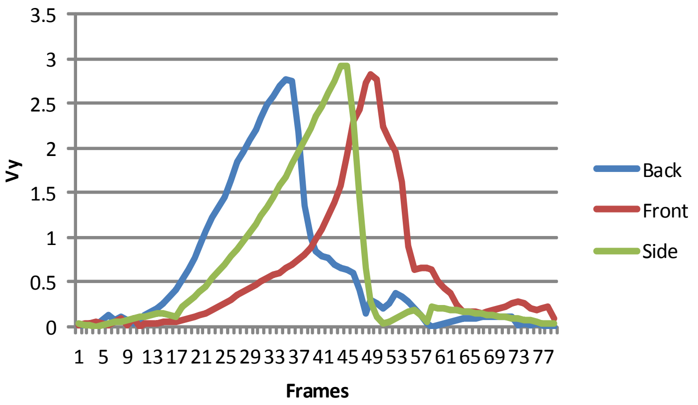

Figure 3: Types of falls simulated by OpenSim. A standard model represents a typical male body of 1.78m height and 78Kg mass was used for all three simulations. Notice the time-to-fall difference due to balancing constraints caused by the feet

Approach / Fall type	Customised	Trained based
Back	4.93	5.08
Front	4.82	5.12
Side	4.84	4.98

Table 1: Mean error from Physics and trained based approaches.

$$V_y(t)_{(pol)} = \sum_i a_i t^i \quad (3)$$

$$\varepsilon = \log[\sum_t (V_y(t)_{(pol)} - V_y(t)_{(meas)})^2] \quad (4)$$

We consider a training dataset that consists of positive and negative examples of falls and apply linear SVM on the error values to specify the decision boundary between the two classes.

4 Experimental Results

The dataset from [5] is used to evaluate our approach, which consists of 51 fall and 81 non fall examples. For each sample from the dataset, the height of the person is measured and the body orientation is estimated to select the closest simulation model and properly fit a 3rd degree polynomial.

In Figure 5, a simulated and an example of an actual fall are depicted to demonstrate their similarity. This particular pair has a Pearson product-moment correlation coefficient at 0.92, indicating a high degree similarity. The average Pearson product-moment correlation coefficient between simulated and actual falls for the backward, forward and sideways types are 0.89, 0.82 and 0.87 respectively, indicating again high degree similarity. For each sample, the fitting error is measured considering the difference between the customised fitted simulated model from the actual observed event (fall or non-fall). The algorithm has 100% classification accuracy and the measured error can be seen in Figure 4, where falls and non falls are linearly separable for the particular dataset. The red line represents the decision boundary, as specified by linear SVM.

Our approach is compared against a similar baseline method, which also uses SVM applied on the error values derived from Eq. 4, but lacks the customisation, provided by the simulation. Specifically, for the baseline approach, $V_y(t)_{(pol)}$ is estimated by polynomial (3rd degree) regression using all actual fall data from the dataset. The mean error for the proposed customised approach is lower for all different types of falls as seen in Table 1.

5 Conclusion and Future Work

This paper presented a method for customising the expected human body falls, using their height and body orientation, based on physics-based simulations, derived by the OpenSim software. Customised simulation seems to provide a better approximation of real fall events than only considering a polynomial regression of recorded fall events. The simulated model and its synergy with existing machine learning algorithms could be further investigated, with the aim to further customise and therefore optimise

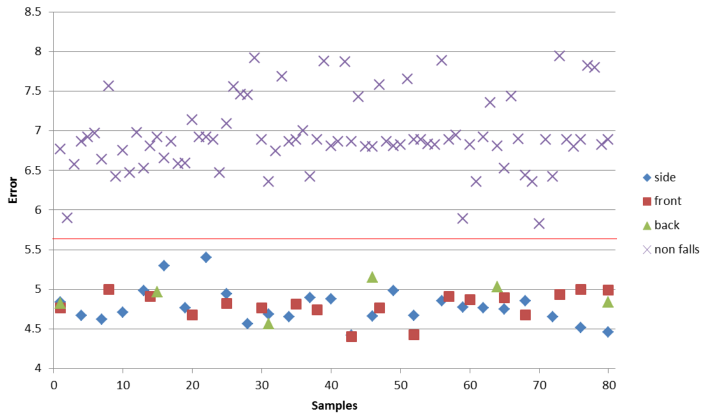

Figure 4: Fitting error of falls and non-falls: Notice how well separated falls and non falls appear

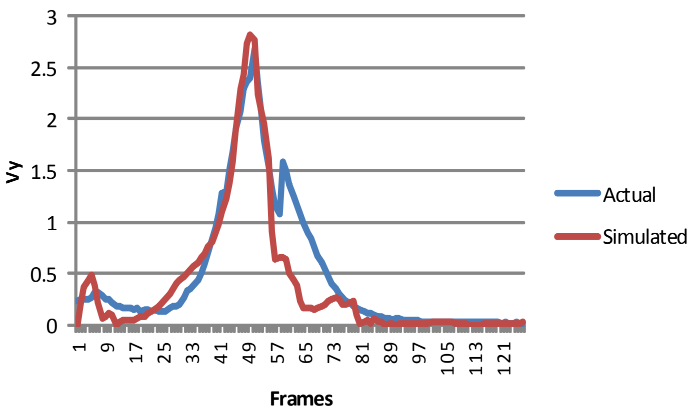

Figure 5: Graph comparing actual with simulated fall. X axis: frames, Y axis: velocity m/sec

fall detection algorithms, but also to address the issue of the scarcity of realistic fall recordings.

[1] Marcus A. Brubaker, David J. Fleet, and Aaron Hertzmann. Physics-based person tracking using the anthropomorphic walker. *International Journal of Computer Vision*, 87(1):140–155, 2009. ISSN 1573-1405. doi: 10.1007/s11263-009-0274-5. URL `http://dx.doi.org/10.1007/s11263-009-0274-5`.

[2] S. L. Delp, F. C. Anderson, A. S. Arnold, P. Loan, A. Habib, C. T. John, E. Guendelman, and D. G. Thelen. Opensim: Open-source software to create and analyze dynamic simulations of movement. *IEEE Transactions on Biomedical Engineering*, 54(11):1940–1950, Nov 2007. ISSN 0018-9294. doi: 10.1109/TBME.2007.901024.

[3] W. Liu, Y. Zhang, S. Tang, J. Tang, R. Hong, and J. Li. Accurate estimation of human body orientation from rgb-d sensors. *IEEE Transactions on Cybernetics*, 43(5):1442–1452, Oct 2013. ISSN 2168-2267. doi: 10.1109/TCYB.2013.2272636.

[4] G. Mastorakis and D. Makris. Fall detection: types, solutions, weaknesses. a review of computer vision based fall detection systems. In *IET Human Motion Analysis for Healthcare Applications*, 2016.

[5] Georgios Mastorakis and Dimitrios Makris. Fall detection system using kinect's infrared sensor. *Journal of Real-Time Image Processing*, 9(4):635–646, 2012. ISSN 1861-8219. doi: 10.1007/s11554-012-0246-9. URL `http://dx.doi.org/10.1007/s11554-012-0246-9`.

[6] Zhong Zhang, Christopher Conly, and Vassilis Athitsos. A survey on vision-based fall detection. In *Proceedings of the 8th ACM International Conference on PErvasive Technologies Related to Assistive Environments*, PETRA '15, pages 46:1–46:7, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3452-5. doi: 10.1145/2769493.2769540. URL `http://doi.acm.org/10.1145/2769493.2769540`.