In search of a better fall detection algorithm

Andreea Catrina[1] & Vlad Stroia[2]
[1]University of Bath
[2]Université Libre de Bruxelles

October 2021

Abstract

Fall detection algorithms are an unquestionable necessity of our society, enabling us to better care for our loved ones. In this paper, we venture to find a reliable and secure solution.

Keywords— Fall detection algorithm, Open-Source

1 Introduction

The idea is to have the fall detection algorithm, henceforth denoted by FDA, run on the user's smartphone, in the intent of notifying said user's contact person in case of a detected emergency.

2 Implementation

We have decided to build our project with the open-source software package Flutter Toolkit[3], using the Dart programming language[2]. This allows us to create apps for all our target devices (Android and IOS smartphones and smartwaches) from a single codebase. The source code is freely distributed[4].

2.1 First Approach

We began by accesing the phone's integrated *acceleromter*. This is the main source of data that we plan to use to determine the state of the app. (the 2 possible states being either *At rest* or *Triggered*)

We also plan to use the phone's integrated *gyroscope* sensor.

2.2 Simple mathematical model

$$a = \sqrt{a_x^2 + a_y^2 + a_z^2} - g \tag{1}$$

Equation (1) gives us the phone's instant acceleration, disregarding the effects of Earth's gravitational field.

As a first proof of concept, we compared the acceleration a with a certain predefined *threshold acceleration* a_t. Whenever $a \geq a_t$ the app state changes to *Triggered*. As

expected, if a_t is low enough, the app detects virtually all falls, but the rate of false positives is too high for the verdict of the algoirthm to hold any weight.

We come to the conclusion that the propper identification of false positives is to be the real challenge in this project. A very fine balance must be found, for if the application fails to detect falls it may not be deemed secure, but if the rate of false positives is too high, the app is rendered useless just as quickly.

2.3 Going forward

To better understad the nuances of this problem, we consulted [1], were we found a wealth of knowledge in the form of well structured databases of sensor readings during falls.

We appreciate that with this data we would be able to further train our model.

References

[1] Eduardo Casilari, José-Antonio Santoyo-Ramón, and José-Manuel Cano-García. "Analysis of Public Datasets for Wearable Fall Detection Systems". In: *Sensors* 17.7 (2017). ISSN: 1424-8220. DOI: 10.3390/s17071513. URL: https://www.mdpi.com/1424-8220/17/7/1513.

[2] *Dart is a client-optimized language for developing fast apps on any platform.* URL: http://www-https://dart.dev/.

[3] *Google's UI toolkit for building natively compiled applications for mobile, web, desktop, and embedded devices from a single codebase.* URL: http://www-https://flutter.dev/.

[4] *Source code of our application.* URL: https://github.com/vladstroia/fall_detection_algorithm.