

Detección de caídas mediante un acelerómetro de tres ejes ubicado en la muñeca en personas de tercera edad

Armando Collado Villaverde, María D. R-Moreno, David F. Barrero y Daniel Rodríguez

Universidad de Alcalá, Departamento de Automática / CC. Computación
Crta. Madrid-Barcelona, Alcalá de Henares, Madrid, Spain
armando.collado@edu.uah.es, mdolores@aut.uah.es, david@aut.uah.es, daniel.rodriguez@uah.es

Resumen Uno de los grandes problemas del envejecimiento es la pérdida de motricidad y el consecuente aumento de riesgo de caídas. Las consecuencias prácticas de las caídas en tercera edad son severas. Dada la popularización de los relojes inteligentes, planteamos un mecanismo para la detección de caídas mediante el uso de dichos relojes. En particular, mostramos cómo la utilización de técnicas de Aprendizaje Automático sobre datos provenientes del acelerómetro de un reloj inteligente nos permite detectar caídas con un muy alto grado de fiabilidad. Para ello construimos un conjunto de datos con aceleraciones medidas en caídas simuladas, que agrupamos en ventanas de tiempo, y a continuación aplicamos Aprendizaje Supervisado para clasificar la señal de aceleración. Los resultados muestran una precisión superior al 98% detectando las caídas.

Palabras Clave: Caídas, acelerómetro, Aprendizaje Automático, Clasificación, Aprendizaje Supervisado, Dependencia, Tercera Edad

1. Introducción

En los últimos años se han producido numerosos avances en el campo del reconocimiento de actividades humanas. Esto, unido al envejecimiento de la población ha permitido el uso del reconocimiento de actividades con fines asistenciales, principalmente orientado a personas mayores y dependientes. Del amplio espectro de situaciones que pueden ser detectadas y reconocidas mediante el uso de acelerómetros, la detección de caídas en personas mayores ocupa una de las primeras posiciones en cuanto a importancia práctica. Las caídas son una fuente muy significativa de problemas asociados a la tercera edad, tanto por sus consecuencias (tales como traumatismos) como por ser ellas mismas una consecuencia de otras dolencias (desvanecimientos, infartos, etc). Por lo tanto, las caídas son una de las mayores fuentes de preocupación tanto de profesionales como de familiares de personas vulnerables.

Se han desarrollado numerosos proyectos e investigaciones sobre el reconocimiento de actividades mediante acelerómetros, dado que es un nuevo campo que

presenta retos aún no explorados. Podemos mencionar el trabajo de Bulling *et al.* [1], basado en el reconocimiento de actividades usando tres acelerómetros a lo largo del brazo; Mannini & Sabatini [2] y Zheng [3] centrados en el reconocimiento básico de actividades (estar de pie, sentarse, levantarse y tumbarse) usando también acelerómetros. Concretamente Kwapisz *et al.* [4] usan móviles para el reconocimiento de actividades. Trabajos similares son los de Ravi *et al.* [5], o el de Gibson *et al.* [6] en el cual se tratan de detectar caídas, así como su dirección, mediante un acelerómetro situado en el pecho. En otros trabajos también se ha mostrado la preocupación por la detección de caídas orientados a personas mayores, como el trabajo de Lustrek & Kaluza [7], Noury *et al.* [8], o de Albert *et al.* [9] y Zhang *et al.* [10] los cuales están basados en el uso de móviles para la detección de las caídas.

Entre los sistemas de detección de caídas no basados en acelerómetros, tal vez los más numerosos son los basados en imágenes. Podemos destacar los trabajos de Shaou-Gang Miao [11], Rita Cucchiara [12] y Edouard Awinet [13]. También se ha abordado este problema con sensores de sonido y vibraciones [14]. De cara a la atención a la dependencia podemos identificar dos problemas básicos en esta aproximación. Por una parte la complejidad de la instalación de equipos de imagen, susceptibles además de tener zonas sin cobertura, pero sobre todo existe un factor psicológico de rechazo a convivir con cámaras de video o micrófonos, a pesar de que no exista una intervención humana que pueda comprometer la privacidad del dependiente.

Este trabajo presenta un sistema automático de detección de caídas en base al tratamiento de los datos recogidos de un acelerómetro tri-axial situado en la muñeca del sujeto, como los que se pueden encontrar habitualmente en los relojes inteligentes, utilizando Aprendizaje Automático. Dicho dispositivo se puede integrar con facilidad en la vida del dependiente. Siendo el reloj un dispositivo ya conocido y comúnmente utilizado, minimiza el riesgo de rechazo psicológico. Desde una óptica técnica, la existencia en los relojes inteligentes de sensores, capacidad de comunicación e interfaces de programación hacen de estos una tecnología prometedora en atención a la dependencia.

A diferencia de otros trabajos relacionados, hemos tomado los datos sobre las caídas usando un maniquí de enfermería; haciéndolo caer de forma similar a como caería una persona mayor en caso de pérdida de consciencia, que es el tipo de caída más común en tercera edad. Esta diferencia en la recogida de datos es crucial, ya que en caso de recogerlos de personas con entrenamiento previo la forma en la que ocurrirían las caídas no sería la misma. Simular las caídas con personas (habitualmente jóvenes y entrenadas) introduce un sesgo que tiene una repercusión notable [15]. La utilización del maniquí introduce otros sesgos nuevos, pero una captura de datos real implica utilizar caídas reales, con el potencial riesgo para el sujeto. En la práctica, los pocos estudios [15] que utilizan datos reales de caídas utilizan una población en riesgo de parecer caídas, lo cual también sesga el resultado, además de ser muy costoso y dilatado en el tiempo.

También hay que considerar que en la tercera edad la dinámica de la caída es distinta que en otros segmentos de población. La caída más común es por

desvanecimiento, en la que la persona cae verticalmente. Las personas mayores tienden a separar las piernas y levantar menos los pies al andar, produciendo un desplazamiento del centro de gravedad hacia un punto más bajo y desplazado hacia delante. Como consecuencia, rara vez se producen caídas laterales o hacia atrás; en una caída típica primero se hincan las piernas y luego cae el torso del cuerpo. Afortunadamente esta situación se puede simular adecuadamente mediante un maniquí de enfermería.

El trabajo está estructurado como sigue. En la sección 2 se presenta el procedimiento de captura de datos, tanto los de las caídas como los que ausentan este evento. En la sección 3 se trata la metodología de análisis de los datos, especificando herramientas y algoritmos utilizados. Posteriormente, en la sección 4, se presentan una serie de pruebas de tratamiento de datos de actividades de la vida diaria. En la sección 5 se analizan los datos de las caídas, realizando posteriormente una selección de atributos en la sección 6. Se finaliza con unas conclusiones y trabajo futuro.

2. Procedimiento de captura de datos

Se obtuvieron dos conjuntos de datos, uno conteniendo simulaciones de caídas y otro sin caídas¹. En todos los casos se midió la aceleración en los tres ejes, X, Y y Z, con un reloj inteligente. El periodo de muestreo fue 20 milisegundos, velocidad de muestreo máxima permitida por el reloj.

Un problema a considerar es la clase base con la que comparar las caídas para poder plantear el entrenamiento del clasificador. Decidimos capturar datos mientras se practicaba un deporte (baloncesto), que contiene una cantidad notable de movimiento vertical de la muñeca. Así evitamos que la detección de la caída sea excesivamente fácil, aunque sea una situación poco realista en una persona de tercera edad en situación de dependencia.

La simulación de las caídas se realizó con un maniquí de enfermería, cuyas características físicas son similares a las humanas en cuanto a peso, tamaño y movilidad de las articulaciones. El objetivo era simular la caída lo más realísticamente posible, imitando la caída típica de una persona de tercera edad. Con este fin todo el proceso de simulación se encontró supervisado por un experto en Geriátrica, que descartaba todas las caídas que, a su juicio, no resultaban naturales.

El procedimiento de simulación de la caída fue el siguiente: una persona sujetaba de pie al maniquí y lo soltaba con un ligero empuje hacia delante, con ello el maniquí cae casi verticalmente, primero de rodillas y después con el resto del cuerpo. El resultado así es más natural que si simplemente se deja caer en vertical, en cuyo caso el maniquí se queda sobre las rodillas. El empuje imita a efectos prácticos el desplazamiento del centro de gravedad típico en una persona mayor. La Figura 1 muestra la secuencia típica de las caídas simuladas. Dos

¹ Los datos se encuentran publicados en <http://atc1.aut.uah.es/~david/caepia2016>



Figura 1: Secuencia del desarrollo de una caída: situación de partida, con el maniquí sujetado manteniendo las piernas ligeramente flexionadas, primer impacto con las piernas y finalmente el impacto del tronco.

expertos en geriatría supervisaron el procedimiento, descartando aquellas caídas simuladas que no parecieran naturales.

La lentitud del procedimiento limitó la capacidad de recogida de datos. En total se simularon 42 caídas, de las cuáles los expertos consideraron válidas 30. Las 12 caídas que se desecharon no eran realistas por diversos motivos: en algunos casos el maniquí no cayó de rodillas (lo hizo directamente con todo el cuerpo), se quedó de rodillas sin llegar a caer el tronco, o tras caer de rodillas se desplazó lateralmente o hacia atrás en lugar de hacia delante. La duración de las caídas era en torno a medio segundo, aunque se toman muestras durante los instantes previos y posteriores a la caída. El hardware impuso una velocidad de muestreo de 20ms. Dado que el acelerómetro ofrece tres valores de aceleración, cada caída genera entre 150 y 300 muestras.

3. Preprocesamiento de los datos

Las muestras del acelerómetro fueron agrupadas en ventanas de tiempo, de manera que todas las muestras consecutivas dentro de la ventana de tiempo alimentan al clasificador en paralelo. Indirectamente, mediante este procedimiento, se considera un histórico en el clasificador con la misma duración que el tamaño de la ventana. Todas estas ventanas se etiquetaron manualmente, indicando cuáles de ellas contienen la caída y cuáles no. Para construir la clase base se definieron ventanas del mismo tamaño que las caídas, y se muestrearon etiquetándolas como 'No Caída'.

La captura de datos presentó una asimetría: resultó mucho más fácil capturar datos deportivos que de las caídas, resultando en un conjunto de datos desbalanceado. Para evitar que esta situación afectara negativamente al entrenamiento del clasificador se submuestreó las clases deportivas, igualando el número de instancias al de las caídas.

Una vez las ventanas de tiempo fueron creadas y etiquetadas, se pudieron utilizar para entrenar y evaluar a los clasificadores. Dado el estado inicial de esta investigación, se consideraron algunos algoritmos de clasificación básicos integrados en Weka: el árbol de clasificación C4.5 [16] y el vecino más cercano 1-NN y 2-NN [17], regresión logística [18], *random forest* [19], Naïve Bayes [20] y PART [21]. Para evitar el sesgo que la selección del conjunto de entrenamiento pueda inducir, la evaluación utilizó validación cruzada con diez subconjuntos.

4. Detección de caídas

La detección de caídas se plantea como un problema de clasificación binaria: clasificar las aceleraciones contenidas en ventanas de tiempo como correspondientes a una caída o no. El conjunto de datos se construyó tal y como se describió en la sección 2, y preprocesaron siguiendo el procedimiento descrito en la sección 3.

Antes de realizar el entrenamiento del clasificador debemos establecer el tamaño de la ventana. De cara a evitar arbitrariedad en la decisión se hizo un pequeño estudio exploratorio.

4.1. Determinación del tamaño de la ventana

El tamaño de la ventana es un parámetro básico a la hora de desarrollar el clasificador. Un tamaño de ventana excesivo podría introducir información irrelevante (aceleraciones fuera del rango de tiempo en el que se produce la caída), que podría degradar el rendimiento del clasificador en términos de precisión y sobrecarga computacional. Hay que considerar que el tamaño de ventana está directamente relacionado con el número de atributos que se introducen al clasificador, por lo que conviene mantenerlo lo más acotado posible. Sin embargo, si el tamaño de la ventana es demasiado pequeño, el clasificador podría carecer de información necesaria, reduciendo la capacidad de detección.

En primer lugar exploramos la influencia del tamaño de la ventana en la precisión del clasificador. Para ello calculamos la precisión de los clasificadores con ventanas de distinto tamaño. Partiendo de un periodo de muestreo de 20 ms, se varió el número de muestras contenidas en la ventana, desde 10 (tamaño de la ventana 100ms) hasta 40 (tamaño de la ventana 400ms).

La influencia del tamaño de la ventana en el rendimiento del clasificador se muestra en el Cuadro 1. El rendimiento de los clasificadores, pese a ser bastante alto en casi todos los casos, con ventanas inferiores a 20 muestras tiene un rendimiento considerablemente más bajo que con 20-30, por otra parte, a partir de 30 muestras por ventana el rendimiento baja en algunos clasificadores como el 1-NN, mientras que en el árbol C4.5 sigue aumentando ligeramente. Esto, unido al coste computacional, descarta el uso de ventanas iguales o superiores a 35 muestras.

Se optó por usar ventanas de 25 muestras (500ms). El Cuadro 1 sugiere que una ventana de 500ms ofrece una buena precisión en la mayoría de los clasificadores utilizados a la vez que mantiene el número de atributos limitados.

Cuadro 1: Efecto del tamaño de ventana. Precisión con muestreo cada 20 ms y ventanas con número de muestras variable desde 10 a 40.

Muestras	C4.5	1-NN	2-NN	Reg. Log.	Random Forest	Naive Bayes	PART
10 (200ms)	88.11 %	95.36 %	95.69 %	77.45 %	94.89 %	73.56 %	88.72 %
15 (300ms)	88.44 %	95.81 %	96.01 %	79.42 %	95.73 %	73.03 %	89.73 %
20 (400ms)	89.12 %	95.26 %	95.75 %	79.69 %	95.51 %	72.50 %	91.89 %
25 (500ms)	89.22 %	95.60 %	95.8 %	79.22 %	97.14 %	72.29 %	92.59 %
30 (600ms)	90.10 %	93.67 %	94.60 %	79.28 %	97.726 %	72.91 %	92.45 %
35 (700ms)	90.25 %	92.46 %	93.62 %	78.98 %	98.24 %	74.05 %	92.73 %
40 (800ms)	90.54 %	91.45 %	92.93 %	78.85 %	98.55 %	75.42 %	93.47 %

Cuadro 2: Rendimiento de los algoritmos C4.5, 1-NN y PART incluyendo la media y desviación estándar. La ventana contiene 25 muestras separadas por 20 ms.

Algoritmo	C4.5	1-NN	PART
Precisión	0.953	0.972	0.955
Recall	0.953	0.971	0.955
F-Measure	0.953	0.971	0.955
%Instancias Correctas	95.291	97.076	95.528

Conviene destacar que en todos los casos el rendimiento de los clasificadores es bastante alto, sugiriendo que la dificultad del problema es moderada.

4.2. Efecto de la media y desviación estándar

Con el objeto de mejorar el rendimiento de los clasificadores, a los datos de aceleración en bruto se les añadió dos nuevos atributos: la media y desviación estándar de las aceleraciones contenidas en la ventana. Ambos atributos se calcularon para las aceleraciones en cada uno de los tres ejes. Para agilizar la experimentación se volvieron a evaluar los clasificadores C4.5, 1-NN y PART, ya que eran los que tenían un mayor rendimiento en comparación con el resto, además de no presentar una gran carga computacional como en el caso del *Random Forest*; excluyendo el resto de algoritmos. El resultado es una mejora significativa del rendimiento, tal y como se puede apreciar en el Cuadro 2. Hay una clara mejora en ambos clasificadores, en el caso del C4.5 y PART el rendimiento aumenta un 2% mientras que 1-NN mejora en un 3%. También se probaron más atributos sintéticos como la derivada de la función en cada eje, pero el rendimiento apenas aumentaba en la mayoría de los casos, llegando incluso a disminuirlo.

Un aspecto todavía no considerado, y que puede tener un impacto notable en el clasificador resultante, es la utilidad de los atributos incorporados de cara a realizar la clasificación. Por eso en la siguiente sección se aborda el problema de la selección de los atributos.

5. Detección de caídas con selección de atributos

El número de atributos utilizados en la clasificación es relativamente alto, 81 atributos. En una ventana de tiempo se realizan 25 muestras, cada una arrojando tres valores de aceleración (X, Y, Z), generando en total 75 atributos. A estos hay que añadir las características sintéticas aplicadas a cada uno de los ejes, totalizando 81 atributos. El alto número de atributos hace que el entrenamiento y ejecución de los clasificadores tenga un mayor coste computacional, y eventualmente también puede influir negativamente en su rendimiento.

Se realizó una selección de atributos con el objetivo de reducir la complejidad del clasificador y, eventualmente, mejorar la clasificación. Se aplicó el método *Correlation Feature Selection Subset Evaluation* [22], que evalúa la correlación de cada atributo con la clase, así como la redundancia de los atributos entre sí, puntuando mejor aquellos atributos con una alta correlación con la clase y poca entre sí.

Cuadro 3: Valoración de los atributos según el método *Correlation Feature Selection Subset Evaluation*. Los atributos se encuentran ordenados de mayor a menor correlación.

%Inf.	Atributo	%Inf.	Atributo	%Inf.	Atributo	%Inf.	Atributo
0.342	MediaX	0.22	AcelX8	0.213	AcelX24	0.169	AcelZ9
0.329	MediaZ	0.22	AcelX10	0.199	DesvX	0.168	AcelZ11
0.233	DesvY	0.219	AcelX12	0.176	DesvZ	0.168	AcelZ15
0.225	AcelX0	0.219	AcelX15	0.171	AcelZ0	0.168	AcelZ17
0.224	AcelX1	0.219	AcelX18	0.171	AcelZ1	0.166	AcelZ23
0.224	AcelX4	0.218	AcelX20	0.17	AcelZ2	0.166	AcelZ24
0.223	AcelX2	0.213	AcelX22	0.17	AcelZ4	0.165	AcelZ20
0.222	AcelX6	0.213	AcelX23	0.169	AcelZ7	0.135	MediaY

El Cuadro 3 muestra la aportación de cada atributo. Se puede apreciar que los atributos mejor correlados con la clase son algunos valores de los ejes X y Z, principalmente los iniciales y finales de la ventana, así como las medias y desviaciones de las aceleraciones. Por el contrario, el eje Y apenas tiene capacidad predictiva, ya que debido a la dinámica de las caídas no hay aceleraciones considerables a lo largo de dicho eje. De hecho, considerar las aceleraciones en Y perjudican el rendimiento del clasificador; eliminando completamente los atributos asociados al eje Y induce una ligera mejora en el rendimiento tanto de C4.5 como de 1-NN y PART, tal y como se puede apreciar en el Cuadro 4.

En el Cuadro 4 se muestra la precisión tras la selección de atributos. Los atributos utilizados son los mostrados en el Cuadro 3, 32 en total. En todos los casos el rendimiento aumentó. En el caso de C4.5 y de PART el aumento es de aproximadamente un 0,6% y el de 1-NN, un 1,5%. Adicionalmente, el tiempo de evaluación y construcción de los clasificadores se redujo en gran medida, aproximadamente a la mitad.

Cuadro 4: Rendimiento de los algoritmos C4.5, 1-NN y PART tras llevar a cabo la selección de atributos. Los atributos considerados son los mostrados por el Cuadro 3.

	C4.5	1-NN	PART
Precisión	0.958	0.986	0.961
Recall	0.958	0.986	0.961
F-Measure	0.958	0.986	0.961
%Instancias Correctas	95.842	98.614	96.126

6. Selección de atributos específica al algoritmo de clasificación

Un problema del método de selección de atributos mostrado anteriormente es que considera exclusivamente las propiedades del conjunto de datos, sin tener en cuenta la relación existente entre el clasificador y los datos. Una adecuada explotación de esta interrelación puede generar una mejora en el clasificador, cuestión que exploraremos en esta sección.

Usamos un *wrapper*, el cuál realiza una búsqueda de atributos evaluando su rendimiento dentro del clasificador con validación cruzada. Específicamente utilizamos *WrapperSubSetEval* [23], que busca el mejor subconjunto de atributos mediante un algoritmo de ascenso de colina.

Cuadro 5: Rendimiento de los algoritmos C4.5, 1-NN y PART después de realizar una selección de atributos utilizando un *WrapperSubSetEval* con una búsqueda de ascenso de colina.

	J48	1-KNN	PART
Atributos	8	6	8
Precisión	0.967	0.997	0.959
Recall	0.967	0.997	0.959
F-Measure	0.967	0.997	0.959
%Instancias Correctas	96.715 %	99.658 %	96.126 %

El Cuadro 5 muestra el rendimiento de los algoritmos C4.5, PART y 1-NN después de aplicar el procedimiento de selección de atributos descrito. Al usar dicho algoritmo sobre el árbol C4.5, el número de atributos se ve reducido enormemente, únicamente a 8 atributos, los 6 atributos post-procesado, una aceleración en el eje X al principio de la ventana y otra aceleración en X al final de la ventana. En el caso de PART, pese a que el número de atributos también se reduce a 8, se mantienen los 6 atributos post-procesado, una aceleración en el eje Z al final de la ventana y una aceleración al principio de ventana en X. En 1-NN la reducción es aún más destacable, ya que únicamente se mantienen los seis atributos post-procesado, media y desviación típica de los 3 ejes.

7. Conclusiones y trabajo futuro

En este artículo se ha presentado una aplicación del Aprendizaje Automático para detectar caídas a partir de un acelerómetro triaxial colocado la muñeca de una persona. El objetivo final es implementar un sistema de detección de caídas en un reloj inteligente orientado a la atención de la dependencia en tercera edad.

La detección de la caída se plantea como un problema de clasificación binario, que requiere de datos para alimentar el aprendizaje. Primero se midió la aceleración en la muñeca de un maniquí de enfermería, con tamaño, peso y movilidad similar a un ser humano. El maniquí simuló una caída vertical imitando el tipo de caída más común en tercera edad: una caída en vertical en dos fases, primero las rodillas se posan sobre el suelo, y después el tronco cae hacia delante hasta el suelo. El caso opuesto se realizó mediante una actividad deportiva.

La capacidad de distinguir las dos clases con algoritmos básicos como C4.5 y 1-NN ha resultado ser muy alta. Sin realizar un esfuerzo considerable en la optimización del clasificador, se ha obtenido una precisión de alrededor del 90-95 %, precisión que aumenta cuando se eliminan los datos de aceleración en el eje Y. Aplicando una selección de atributos basada en *wrappers*, la precisión de C4.5 y 1-NN aumenta hasta un 97 % y 99 %, respectivamente. Además, el número de atributos se reduce significativamente hasta 8 y 6.

En el futuro se desea considerar otros tipos de caídas habituales en personas de tercera edad. Para ello se pretenden desarrollar nuevos clasificadores especializados en cada tipo de caída, y en última instancia construir un único clasificador como agrupación de clasificadores.

8. Agradecimientos

Los autores agradecen a Isabel Pascual Benito, Francisco López Martínez y Helena Hernández Martínez, profesores del Departamento de Enfermería y Fisioterapia de la Universidad de Alcalá, por su ayuda diseñando y supervisando el procedimiento de simulación de caídas. Este trabajo ha sido financiado por la UAH (2015/00297/001), JCLM (PEII-2014-015-A) y MINECO (TIN2014-56494-C4-4-P).

Referencias

1. Bulling, A., Blanke, U., Schiele, B.: A tutorial on human activity recognition using body-worn inertial sensors. *ACM Computing Surveys (CSUR)* **1** (2014) 1–33
2. Mannini, A., Sabatini, A.M.: Machine learning methods for classifying human physical activity from on-body accelerometers. *Sensors* **10** (2010) 1154–1175
3. Zheng, X.: (Development of a human movement monitoring system based on wearable devices)
4. Kwapisz, J.R., Weiss, G.M., Moore, S.a.: Activity Recognition using Cell Phone Accelerometers. *ACM SIGKDD Explorations Newsletter* **12** (2010) 74–82
5. Ravi, N., Dandekar, N., Mysore, P., Littman, M.L.: (Activity Recognition from Accelerometer Data) 1541–1546

6. Gibson, R.M., Amira, A., Ramzan, N., Casaseca-de-la higuera, P., Pervez, Z.: Multiple comparator classifier framework for accelerometer-based fall detection and diagnostic. *Applied Soft Computing Journal* **39** (2016) 94–103
7. Luštrek, M., Kaluža, B.: Fall Detection and Activity Recognition with Machine Learning. **33** (2008) 205–212
8. Noury, N., Fleury, a., Rumeau, P., a.K. Bourke, Laighin, G.O., Rialle, V., Lundy, J.E.: Fall detection - Principles and Methods. 2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (2007) 1663–1666
9. Albert, M.V., Kording, K., Herrmann, M., Jayaraman, A.: Fall classification by machine learning using mobile phones. *PLoS ONE* **7** (2012) 3–8
10. Zhang, T., Wang, J., Liu, P., Hou, J.: Fall Detection by Embedding an Accelerometer in Cellphone and Using. *Journal of Computer Science* **6** (2006) 277–284
11. Miaou, S.G., Sung, P.H., Huang, C.Y.: A customized human fall detection system using omni-camera images and personal information. *Conference Proceedings - 1st Transdisciplinary Conference on Distributed Diagnosis and Home Healthcare, D2H2 2006* **2006** (2006) 39–42
12. Cucchiara, R., Prati, A., Vezzani, R., Emilia, R.: and Alarm Generation. *Expert Systems* **24** (2007) 334–345
13. Auvinet, E., Multon, F., Saint-Arnaud, A., Rousseau, J., Meunier, J.: Fall detection with multiple cameras: an occlusion-resistant method based on 3-D silhouette vertical distribution. *IEEE transactions on information technology in biomedicine : a publication of the IEEE Engineering in Medicine and Biology Society* **15** (2011) 290–300
14. Zigel, Y., Litvak, D., Gannot, I.: A method for automatic fall detection of elderly people using floor vibrations and soundProof of concept on human mimicking doll falls. *IEEE Transactions on Biomedical Engineering* **56** (2009) 2858–2867
15. Bagalà, F., Becker, C., Cappello, A., Chiari, L., Aminian, K., Hausdorff, J.M., Zijlstra, W., Klenk, J.: Evaluation of accelerometer-based fall detection algorithms on real-world falls. *PLoS one* **7** (2012) e37062
16. Learning, M., Learning, M., Publishers, K.A., Publishers, K.A., Sciences, A.C., Sciences, A.C., August, R., August, R.: Induction of Decision Trees. *Expert Systems* (2007) 81–106
17. Cover, T., Hart, P.: Nearest neighbor pattern classification. *IEEE Transactions on Information Theory* **13** (1967) 21–27
18. Le Cessie, S., Van Houwelingen, J.C.: Ridge estimators in logistic regression. *Applied statistics* (1992) 191–201
19. Tin Kam Ho: Random decision forests. *Proceedings of 3rd International Conference on Document Analysis and Recognition* **1** (1995) 278–282
20. John, G.H., Langley, P.: Estimating continuous distributions in bayesian classifiers. In: *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, Morgan Kaufmann Publishers Inc. (1995) 338–345
21. Frank, E., Witten, I.H.: Generating accurate rule sets without global optimization. In: *ICML. Volume 98.* (1998) 144–151
22. Hall, M.A.: Correlation-based Feature Subset Selection for Machine Learning. PhD thesis, University of Waikato, Hamilton, New Zealand (1998)
23. Kohavi, R., John, G.H.: Wrappers for feature subset selection. *Artificial intelligence* **97** (1997) 273–324