# Improving the accuracy of existing camera based fall detection algorithms through late fusion

Greet Baldewijns[1,2,3], Glen Debard[1,3,5], Gert Mertes[1,2,3], Tom Croonenborghs[1,6,7] and Bart Vanrumste[1,2,3]

*Abstract*— **Fall incidents remain an important health hazard for older adults. Fall detection systems can reduce the consequences of a fall incident by insuring that timely aid is given. Currently fall detection algorithms however suffer a reduction in accuracy when introduced in real-life situations. In this paper a late fusion technique is proposed that will improve the accuracy of existing fall detection systems. It combines the confidence levels of different single camera fall detection systems. Four different aggregation methods are compared to each other based on the Area Under the Curve (AUC) of precision-recall curves. Calculating the median of the confidence levels of five cameras an increase of 218% in the AUC of the precision-recall-curves is achieved compared to the AUC of the single camera fall detector. These results show that significant improvements can be made to the accuracy of single camera fall detectors in a relatively easy way.**

## I. INTRODUCTION

Fall incidents cause more than 50% of injury-related hospitalizations among persons aged 65 and older [1]. Moreover, 30% of those who fall are unable to rise unaided after such a fall incident. Long periods of time on the floor however puts the fall victim at risk for further medical complications [2].

In addition previous research showed that the majority of older adults who are unable to rise from the floor do not use the emergency call system [2]. Automatic fall detection systems can thus be an important aid for older adults living at home. By triggering alarms when a person has fallen they ensure that timely aid is given.

Over the previous years a lot of research effort has been invested in the development of camera based automatic fall detection systems. When installed in real-life environments, however, these systems suffer a reduction in accuracy due to, among others, occlusions, shadows and ghost figures [3]. The influence of some noise sources can be reduced by fusing data from multiple cameras. Early fusion techniques are often used in which the video data from multiple cameras is combined to build a 3D silhouette of the person. After this, the 3D silhouette is used to calculate fall features [4], [5]. The different cameras, however need to remain calibrated at all times which can be difficult when installed in real-life.

In this paper a late fusion technique is proposed. The video from several individual cameras is used as input for a single camera fall detection algorithm. This algorithm in turn generates confidence levels, indicating the chance that a fall occurred within a certain time frame. These confidence levels from corresponding time frames are combined and are subsequently thresholded to decide if a fall has occurred in the given time-frame. Using this methodology no calibration of the separate cameras is needed.

## II. METHODOLOGY

### A. The dataset

*1) Real-life data:* During previous research six older persons were monitored in their home environment for several months. This resulted in the recording of a very extensive dataset including 26 fall incidents [3], [6]. The majority of these fall incidents, however, did not happen in full view of the cameras. Only eleven fall incidents were recorded with the person in full camera view. These fall incidents together with 268.7 hours of real-life data were used as the training dataset of the SVM-based single camera fall detector.

Ethical approval for the study was provided by the Medical Ethics Committee of the Leuven University Hospitals and a written informed consent was obtained from all participants

*2) Simulated data:* To provide additional fall data a realistic simulated fall dataset was recorded in a nursing home set-up with five different cameras. This was done by re-enacting the 26 real-life fall incidents which were recorded during the previously mentioned research. Some small variations were made in the scenarios by changing the ending of the scenario (person remains on the floor after the fall or gets back up again). In total 55 fall scenarios were recorded with an average length of 2:45 minutes each. Additionally 5:50:29 hours worth of normal activities were also recorded. The different scenarios include real-life challenges such as occlusions and light changes. The proposed fusion techniques were firstly evaluated using only the fall recordings, after this the combination of fall and ADL recordings were used for further evaluation. A more extensive description of the used simulated dataset can be found in [7].

### B. Fusion algorithm

A general overview of the different steps in the algorithm is given in Figure 1. Each of the steps is discussed more in

[1]KU Leuven, Campus Groep T, e-Media Lab, Belgium
[2]KU Leuven, ESAT-STADIUS, Leuven, Belgium
[3]IMEC Medical Information Technology department, Leuven, Belgium
[4]Thomas More Kempen, MOBILAB, Geel, Belgium
[5]KU Leuven, ESAT-PSI, Leuven, Belgium
[6]KU Leuven, Department of Computer Science, DTAI, Leuven, Belgium
[7]Program in Translational NeuroPsychiatric Genomics, Brigham and Womens Hospital, Harvard Medical School, Broad Institute of Massachusetts Institute of Technology and Harvard, Cambridge, MA.

978-1-5090-2809-2/17/$31.00 ©2017 IEEE


Authorized licensed use limited to: KU Leuven Libraries. Downloaded on May 11,2020 at 14:01:54 UTC from IEEE Xplore. Restrictions apply.

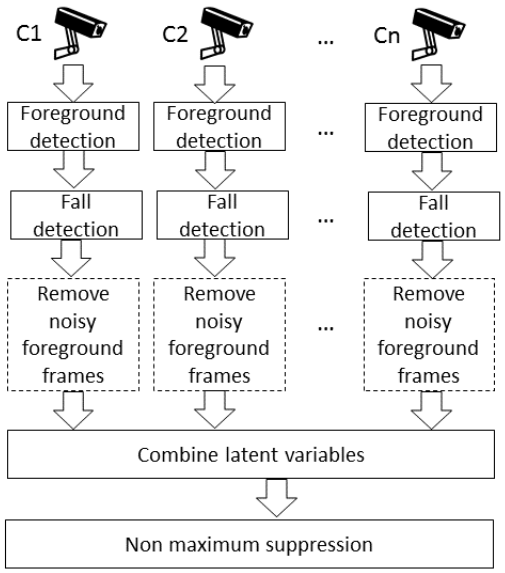



Fig. 1. Schematic overview of the algorithm (steps in the algorithm marked by a dotted line are optional steps). The symbol $n$ equals the number of cameras used.

detail in this section.

*1) Foreground detection:* The fall detection algorithm from Debard et al. was used as the basis of the developed methodology [8]. This algorithm uses a combination of an approximate median filter together with a particle filter to detect the person in each video frame.

*2) Fall detection:* After this, five fall detection features are calculated based on both the bounding box and the bounding ellipse: Aspect ratio (AR), Change in AR, Fall angle, the speed of the centre of mass, the speed of the head.

These fall features are combined into feature vectors for each second in the video stream. This vector thus contains information concerning the current time slot but also about previous and following time slots [8]. The feature vectors are used as input for a Support Vector Machine (SVM) with a linear kernel.

For each second in the simulated video fragments a latent variable is generated as output of the SVM. Using SVMs a latent variable is defined as the distance between the feature vector and the boundary surface. This latent variable is an indication of the confidence level of the prediction made by the SVM. When using a single camera fall detection algorithm these latent variables are thresholded to decide if a time slot contains a fall incident. However in our procedure they are further processed as discussed in section II-B.4.

*3) Remove noisy foreground frames:* In order to evaluate if the algorithm could be improved by eliminating the effect of noise in the preprocessing, two optional steps were added before combining the latent variables.

The first optional step consists of calculating the mean of the bounding ellipse over a four-second window. This window-size was chosen based on the feature vector from [8]. If the size of the bounding ellipse exceeds a 95% confidence interval, which was determined on the whole simulation dataset, the foreground detection was considered noisy. Consequently the validity of the corresponding latent variable was considered doubtful and it was thus discarded.

The second optional step was similar to the first. In this case, however, the standard deviation was used as a measure to determine if the foreground detection was noisy.

The fusion techniques suggested in the next section were evaluated using either or none of the proposed noise reduction steps.

*4) Combine latent variables:* For each second in each video fragment and for each camera viewpoint a latent variable is generated using the SVM-based single camera fall detector. As each scenario is recorded with five cameras, five latent variables are thus generated per second in one scenario. These latent variables were combined in four different ways: the minimum, maximum, average and median of the different latent variables were calculated per second.

For each combination method precision-recall curves were generated and Areas Under the Curve (AUC) were calculated to evaluate the accuracy of the different combinations. An alarm was deemed as a true positive (TP) when it was generated on the fall second which was observed through manual labelling of the data. It was also deemed a TP when it occurred within the feature-vector time frame surrounding the observed fall second.

Two operating points were chosen on the precision-recall curves and precision and recall of those points are also given in the results section.

Using these methods several tests were performed:

- Three-camera tests:
  Ten unique camera combinations can be made using a subset from three of the five cameras. In a first test the average AUC from these unique combinations is calculated.
  For the second test the number of seconds a person is detected in the foreground of the video segment is used as a camera selection criteria. If a video stream from a certain camera has a very low number of seconds with foreground detections this means that the person is not often in the field of view of the camera. The latent variables generated from this camera thus contain less accurate information as opposed to one where the person is often in the field of view of the camera. For this reason the three cameras with the highest number of seconds with foreground detections were combined and the AUC was calculated.
  Lastly, the camera combination with the highest AUC was selected.
- Four-camera tests:
  A similar approach is used for the four-camera set-up. In this case the average AUC for five unique camera combinations was calculated. This was followed by combining the best four cameras based on the number of seconds with a foreground detection. Lastly, the camera combination with the highest AUC was selected.
- Five-camera tests:
  Using five cameras only one combination could be made. As such only one AUC is calculated using five cameras.



Authorized licensed use limited to: KU Leuven Libraries. Downloaded on May 11,2020 at 14:01:54 UTC from IEEE Xplore. Restrictions apply.

TABLE I

NUMBER OF SECONDS IN WHICH A FOREGROUND IS DETECTED (SEC) AND AREA UNDER THE CURVE (AUC) FROM EACH CAMERA INDIVIDUALLY)

| | Fall dataset | | Fall and ADL dataset | |
|---|---|---|---|---|
| | sec | AUC | sec | AUC |
| cam 1 | 1386 | 0.657 | 13362 | 0.249 |
| cam 2 | 1357 | 0.502 | 15912 | 0.243 |
| cam 3 | 1455 | 0.687 | 14240 | 0.138 |
| cam 4 | 1791 | 0.660 | 19208 | 0.189 |
| cam 5 | 1280 | 0.525 | 15562 | 0.193 |

For further in-depth comparison of the results on the fall and ADL dataset two operating points were chosen on the precision-recall curves of the best three- and four-camera combinations as well as on the precision-recall curves of the best and worst single camera and the five-camera combination. The selection was again performed based on the number of seconds in which a person is detected in the foreground. The first operating point corresponds with approximately 70% precision and the second with 70% recall. When a 70% precision could not be reached the maximum is chosen as a first operating point.

*5) Non-maximum suppression:* After combining the latent variables from the different cameras non-maximum suppression is applied over a five-second window. This is again consistent with the algorithm of Debard et al. [8]. In this algorithm however the non-maximum suppression is done on the single camera latent variables. In our approach all latent variables are first combined after which non-maximum suppression is applied.

The different tests as described in this section were performed on both the fall dataset as well as on the combined dataset of fall and ADL activities.

## III. RESULTS

### A. Single camera results

To facilitate the comparison of the late fusion results with those of the single camera fall detection system the AUC for each individual camera on the given simulation datasets was calculated for the fall dataset as well as for the dataset combining fall and ADL videos. The results for both simulation datasets are given in Table I.

From this table we can observe a higher AUC for all cameras when only using fall videos in our dataset. Introducing normal activities to the fall detector will increase the number of possible false alarms and will therefore cause the AUC to drop.

### B. Multi-camera results

Table II gives an overview of the AUC results of the different camera combinations for both the fall dataset and the fall and ADL dataset. Several observations can be made from these results.

It was firstly observed that calculating the maximum of the latent variables yielded the best results in AUC on the fall dataset. In contrast, when using the fall and ADL dataset the mean of the latent variables yielded the best results if a small number of cameras was used. Additionally, if a larger number of cameras is used the median performs better.

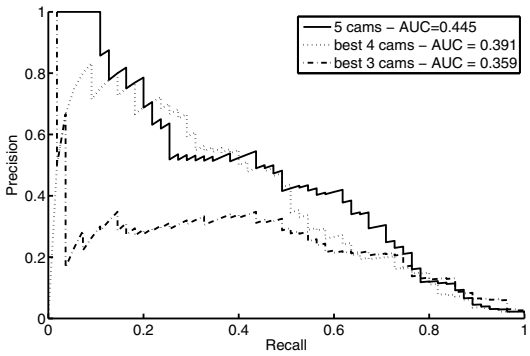



Fig. 2. Precision-recall curves calculated using both fall and ADL data from the five-camera combination and the best four-camera and three-camera combinations (based on the number of seconds in which a foreground is detected)

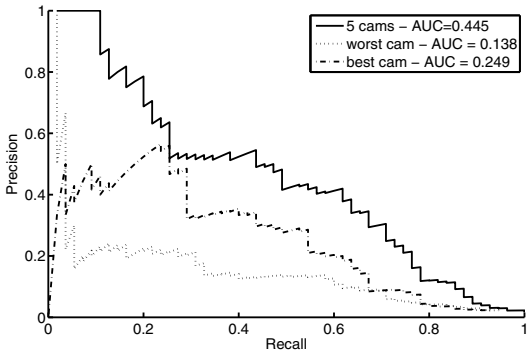



Fig. 3. Precision-recall curves calculated using both fall and ADL data from the five-camera combination and the best and worst camera

However, this table also shows that limited to no improvements were made by preprocessing the latent variables based on bounding ellipse size.

Improvements to the AUC results were made by selecting good camera combinations based on the number of seconds with foreground detections. It is worth noting that the maximum AUC that can be reached using four cameras corresponds with the AUC of the best camera combination selected based on the highest number of seconds with foreground detections.

Lastly, it was observed that although an increase in AUC can be reached by intelligently selecting the best camera combination, adding an extra camera is more fruitful.

A similar observation can be made from Figure 2. In this figure the precision-recall curves from the best three and four camera combinations are compared to the one calculated combining five cameras. This figure also shows that adding an extra camera will result in improvements in accuracy of the algorithm.

Figure 3 again demonstrates that the five-camera combination yields important improvements in the precision-recall-curve compared to those from the best camera and the worst camera.

From Table III it can be concluded that depending on the chosen operating point a different approach in camera combinations is needed. If we aim to detect at least 70% of the fall incidents, installing five cameras yields the lowest amount of false alarms. If, however, an operating point is



Authorized licensed use limited to: KU Leuven Libraries. Downloaded on May 11,2020 at 14:01:54 UTC from IEEE Xplore. Restrictions apply.

TABLE II

AREA UNDER THE CURVE (AUC) FOR THE DIFFERENT CAMERA COMBINATIONS FOR BOTH THE FALL AND THE FALL AND ADL DATASET. THE HIGHEST AUC IS MARKED IN BOLD.

| | | Fall dataset | | | | Fall and ADL dataset | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Aggregation method | | | | Aggregation method | | | |
| | | Min | Max | $\mu$[e] | Med.[e] | Min | Max | $\mu$ | Med. |
| | Filter[a] | | | | | | | | |
| **3 cams** | | | | | | | | | |
| Average | na | 0.54 | **0.77** | 0.73 | **0.77** | 0.23 | 0.25 | **0.30** | 0.30 |
| AUC[b] | AVG | 0.54 | **0.78** | 0.74 | 0.73 | 0.23 | 0.25 | **0.31** | 0.30 |
| | STD | 0.55 | **0.77** | 0.74 | 0.74 | 0.24 | 0.25 | **0.31** | 0.30 |
| Best | na | 0.51 | **0.88** | 0.79 | 0.77 | 0.24 | 0.28 | **0.34** | 0.34 |
| camera | AVG | 0.52 | **0.88** | 0.78 | 0.77 | 0.25 | 0.28 | 0.34 | **0.35** |
| comb.[c] | STD | 0.53 | **0.88** | 0.78 | 0.77 | 0.24 | 0.28 | 0.34 | **0.34** |
| Maximum | na | 0.62 | **0.88** | 0.78 | 0.78 | 0.32 | 0.30 | **0.36** | 0.34 |
| AUC | AVG | 0.63 | **0.88** | 0.78 | 0.78 | 0.32 | 0.30 | **0.36** | 0.35 |
| | STD | 0.62 | **0.88** | 0.78 | 0.78 | **0.37** | 0.30 | 0.37 | 0.34 |
| **4 cams** | | | | | | | | | |
| Average | na | 0.45 | 0.70 | 0.75 | **0.78** | 0.21 | 0.25 | **0.35** | 0.30 |
| AUC | AVG | 0.45 | 0.70 | 0.75 | **0.79** | 0.21 | 0.25 | **0.35** | 0.30 |
| | STD | 0.46 | 0.70 | 0.75 | **0.79** | 0.22 | 0.25 | **0.36** | 0.30 |
| Best | na | 0.50 | **0.81** | 0.77 | 0.75 | 0.18 | 0.23 | 0.32 | **0.39** |
| camera | AVG | 0.51 | **0.81** | 0.77 | 0.75 | 0.18 | 0.23 | 0.32 | **0.39** |
| comb.[d] | STD | 0.52 | **0.81** | 0.77 | 0.76 | 0.19 | 0.23 | 0.32 | **0.40** |
| Maximum | No | 0.50 | **0.87** | 0.77 | 0.77 | 0.35 | 0.29 | 0.37 | **0.39** |
| AUC | AVG | 0.51 | **0.88** | 0.77 | 0.77 | 0.31 | 0.29 | 0.37 | **0.39** |
| | STD | 0.52 | **0.87** | 0.77 | 0.77 | 0.35 | 0.29 | 0.38 | **0.40** |
| **5 cams** | | | | | | | | | |
| AUC | na | 0.37 | **0.83** | 0.73 | 0.78 | 0.18 | 0.26 | 0.38 | **0.44** |
| | AVG | 0.37 | **0.84** | 0.73 | 0.78 | 0.18 | 0.26 | 0.38 | **0.44** |
| | STD | 0.39 | **0.83** | 0.75 | 0.77 | 0.20 | 0.26 | 0.39 | **0.44** |

Notes:
[a]Filtering of the latent variables based on the average bounding ellipse size AVG ( or the standard deviation of the bounding ellipse size (STD). (na) no filtering is applied.
[b]Average AUC of all unique camera combinations
[c] Best camera combination based on the number of foreground detections. For the fall dataset cameras 4, 3 and 1 were combined, for the fall and ADL dataset cameras 5, 4 and 2.
[d] Best camera combination based on the number of foreground detections. For the fall dataset cameras 4, 3, 1 and 2 were combined, for the fall and ADL dataset cameras 5, 4, 2 and 3.
[e] $\mu$ = Mean and Med. = Median

chosen in which 70% of the alarms are correct, intelligently selecting three cameras yields the best results.

Figure 3 and Table III also confirm that installing additional cameras will result in improvements in both precision and recall when compared to the best and worst single camera.

TABLE III

PRECISION (PRC) AND RECALL (REC) FOR TWO OPERATING POINTS CALCULATED ON THE FALL AND ADL DATASET

| | Operating point 1 | | Operating point 2 | |
|---|---|---|---|---|
| | PRC | REC | PRC | REC |
| Best[a] 3-cam | 0.667 | 0.290 | 0.216 | 0.691 |
| Best 4-cam | 0.667 | 0.182 | 0.180 | 0.691 |
| 5-cam | 0.667 | 0.109 | 0.224 | 0.691 |
| Best single cam | 0.500 | 0.146 | 0.081 | 0.691 |
| Worst single cam | 0.565 | 0.236 | 0.086 | 0.691 |

[a] Best cameras selected based on the number of seconds during which a person is detected in the foreground

## IV. DISCUSSION

In this paper a late fusion technique is proposed that uses the output of multiple single camera fall detectors to allow for a more accurate fall detection system as would be the case when using only one single camera fall detector.

The results from the single camera fall detection system were improved by fusing the latent variables from each individual camera. Regarding the combination of fall and ADL dataset fusing the latent variables from three cameras yielded an improvement of 149% on the AUC. For each additional camera a further improvement was made. Four cameras led to an increase of 168% and five cameras 218%. Smaller improvements were made when regarding only the fall dataset. Fusing the latent variables of three cameras resulted in an increase of 127% in the AUC and four and five cameras yielded an increase of 129% and 137%. This smaller increase in AUC reached when regarding the fall dataset as opposed to the combined fall and ADL dataset is mainly due to the limited number of false alarms generated when using the fall dataset.

### A. The design process

Additional challenges were included in the simulated dataset compared to the simulation datasets used by other research groups [7]. The extra challenges resulted in a very



Authorized licensed use limited to: KU Leuven Libraries. Downloaded on May 11,2020 at 14:01:54 UTC from IEEE Xplore. Restrictions apply.

small Area Under the Curve (on average 0.606 on the fall dataset and 0.208 on the ADL and fall dataset) when using a single camera fall detector. Due to this challenging nature of our dataset it is also very difficult to compare our results to those reached by other research groups who are using other fusion techniques.

When combining latent variables from multiple cameras the best results for the fall dataset were achieved by calculating the maximum of the latent variables. In contrast, the best results on the fall and ADL dataset were reached when using the mean or the median of the latent variables. This is mainly due to the very limited number of non-fall activities in the fall dataset. When fusing the focus should thus lie on the detection of the falls which is done by selecting the highest latent variable. When adding a large number of non-fall samples to the dataset the potential for false alarms rises. Eliminating a number of these false alarms is done by using either the median or the mean of the latent variables.

Additionally, the results show that a smart camera selection algorithm can improve the accuracy of the fall detection algorithm. By for example selecting those three cameras with the highest number of seconds with foreground detections an AUC of 0.344 can be reached which is an important increase when compared to the average AUC of the three-camera combination (AUC = 0.305). However, a bigger improvement in AUC can be made by adding additional cameras. The fusion of five cameras for instance yields better results than the best possible combination of four cameras. It is thus preferable to add more cameras than to use prior knowledge to determine the placement of a limited number of cameras.

In order to reduce the noise from the imperfect preprocessing on the results two additional pre-processing steps were implemented. In these steps latent variables were discarded when either the average bounding ellipse size or the standard deviation exceeded a threshold. However, this only resulted in very small increments in the AUC.

As relatively low AUCs are reached by the different fusion techniques it is worth comparing the AUC with that of a random classifier. The AUC of a random classifier is directly proportional to the class imbalance. As we have a highly imbalanced dataset with only 1.5% of the data being positive examples the AUC is determined by a precision of 0.0154. When comparing this to the AUC of the different classifiers it is clear that even the single camera classifiers before fusion perform better on such a highly imbalanced dataset as a random classifier.

### *B. Strengths and weaknesses of the presented method*

It is expected that adding more cameras would further reduce the number of false alarms yielded by the system. A high number of cameras nonetheless increases the overall cost of the fall detection system. Additionally, further research should assess if people are willing to accept a system with a high amount of cameras in their home environment.

The main strength of the presented method is the robustness of the system towards external interference by the user. When using early fusion to boost the accuracy of the system, this system needs to remain calibrated at all times. This is not feasible in real life where several sources of interference to the installed system are present (e.g. the cameras could be moved slightly out of position when touching them while cleaning). Using the late fusion technique the system will work equally well when the position of one of the cameras is moved as long as the room that is monitored remains in the field of view of the camera.

## V. CONCLUSION

In this paper a late fusion technique is presented which can improve the accuracy of camera-based fall detectors. Calculating the median of the latent variables of five cameras yielded the best results with an AUC of 0.445 which is an improvement of 218% when compared to the average AUC of a single camera fall detection. The results show that significant improvements can be made to the accuracy of single camera fall detectors in a relatively easy way.

## ACKNOWLEDGMENTS

The authors would like to acknowledge the following projects: ingenieurs@WZC, iMinds FallRisk project, IWT-ERASME AMACS project, IWT Tetra Fallcam project, H2020-TWINN-2015 ALTHOUR project and the ProFouND network.

## REFERENCES

[1] World Health Organization. Ageing and Life Course Unit: 'Who global report on falls prevention in older age', World Health Organization, 2008.

[2] J. Fleming and C. Brayne: 'Inability to get up after falling, subsequent time on floor, and summoning help: prospective cohort study in people over 90', *BMJ*, 2008, 337, a2227.

[3] G. Debard, P. Karsmakers, M. Deschodt, E. Vlaeyen, E. Dejaeger, K. Milisen, T. Goedemé, B. Vanrumste, and T. Tuytelaars: 'Camera-based fall detection on real world data', *Outdoor and Large-Scale Real-World Scene Analysis*, vol.7474 of Lecture Notes in Computer Science, 356-375, 2012, Springer.

[4] E. Auvinet, F. Multon, A. Saint-Arnaud, J. Rousseau and J. Meunier: 'Fall detection with multiple cameras: An occlusion-resistant method based on 3-d silhouette vertical distribution',*Information Technology in Biomedicine, IEEE Transactions on*, 15(2), 290-300, 2011

[5] D. Anderson, R. H. Luke, J. M. Keller, M. Skubic, M. Rantz, M. Aud: 'Linguistic summarization of video for fall detection using voxel person and fuzzy logic', *Computer Vision and Image Understanding*, 113(1), 80-89, 2009

[6] E. Vlaeyen, M. Deschodt, G. Debard, E. Dejaeger, S. Boonen, T. Goedemé and K. Milisen: 'Fall incidents unraveled: a series of 26 video-based real-life fall events in three frail older persons', *BMC geriatrics*, 2013, **13(1)**, 103.

[7] G. Baldewijns, G. Debard, G. Mertes, B. Vanrumste and T. Croonenborghs: 'Bridging the gap between real-life data and simulated data by providing realistic fall dataset for evaluating camera-based fall detection algorithms', *Healthcare Technology Letters*, 2016, IET Digital Library

[8] G. Debard, G. Baldewijns, T. Goedemé, T. Tuytelaars, and B. Vanrumste: 'Camera-based fall detection using a particle filter', *Proceedings of the 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society* 2015, pp. 6947-6950.

[9] G. Debard, M. Mertens, M. Deschodt, E. Vlaeyen, E. Devriendt, E. Dejaeger, K. Milisen, J. Tournoy, T. Croonenborghs, T. Goedemé, T. Tuytelaars and B. Vanrumste: 'Camera-based fall detection using real-world versus simulated data: how far are we from the solution?, *Journal of Ambient Intelligence and Smart Environments*, 2015, Accepted for publication.



Authorized licensed use limited to: KU Leuven Libraries. Downloaded on May 11,2020 at 14:01:54 UTC from IEEE Xplore. Restrictions apply.