Early versus Late Fusion in a Multiple Camera Network for Fall Detection*

Sebastian Zambanini, Jana Machajdik and Martin Kampel

Institute of Computer Aided Automation, Vienna University of Technology, Austria
{*zamba,jana,kampel*}*@prip.tuwien.ac.at*

Abstract
Falling and not being able to stand up is one of the major risks for elderly people who live alone. Camera based fall detectors represent one of the solutions to this problem. In this paper we compare two approaches for the detection of falls based on multiple cameras, the early fusion approach and the late fusion approach. In the early fusion approach, multiple camera views are combined to reconstruct the 3D voxel volume of the human. Based on semantic driven features fall detection is done on this 3D volume, whereas in the late fusion fall detection is done in 2D and each camera decides on its own, if a fall has occurred. These individual decisions are then combined into an overall decision. Fuzzy logic is both used to estimate confidence values for a fall/no fall in the single cameras as well as in the final voting step. We describe and evaluate both methods and give results on 73 video sequences.

1 Introduction

In the European Union about 30 % of people older than 65 live alone [1]. For these people, falls at home are one of the major risks and an immediate alarming and helping is essential to reduce the rate of morbidity and mortality [15]. Hence, there is a great need for reliable alarm systems. In this context, camera-based fall detectors are well suited since video cameras are passive and flexible sensors.

In the last five years a growing interest and number of publications for camera-based fall detection has been shown [16]. A general classification of proposed methods can be made by whether a fall is detected by modeling the fall action itself or by a frame-by-frame classification using different features measuring human posture and motion velocity. In the former type of methods parametric models like Hidden Markov Models (HMMs) are applied using simple features, e.g. projection histograms [5] or the aspect ratio of the bounding box surrounding the detected human [3, 14]. However, the applicability of these methods in real-life scenarios is limited due to the high diversity of fall actions and the high number of different negative actions which the system should not classify as fall. As also stated by Anderson et al. [4], such a model-based detection is able to detect different actions modeled by different HMMs but is not capable of dealing with unknown actions. The latter type of methods basically measures two types of features: the human posture and the motion velocity. The underlying assumption is that a fall is characterized by a transition from a vertical to a horizontal posture with an unusually increased velocity, i.e. to discern falls from normal actions like sitting on a chair or lying on a bed. In this manner, in the past various features have been used for camera-based fall detection, including the aspect ratio of the bounding box [13] or orientation of a fitted ellipse [12] for posture recognition and head tracking [11] or change rate of the human's centroid [8] for motion velocity. Apart from the features used, the methods also differ in the way how the final decision is derived

*This work was supported by the Austrian Research Promotion Agency (FFG) under grant 819862 (MuBisA).

from the features. Besides parametric classifiers like Neural Networks [7], primarily empirically determined rules are applied [4, 8, 11, 12]. In order to reduce false alarms, a final verification step can be performed which measures if the person was able to move and stand up again in a given period of time.

In the context of ambient assisted living, i.e. detecting falls at elderly homes, we argue that the use of a multiple camera network is inevitable. Multiple cameras allow for the monitoring of multiple rooms and the resolving of occlusions. Furthermore, we have to state that features for posture and motion velocity used in the above papers are highly dependent on the viewpoint of the camera, e.g. consider a fall along the optical axis of the camera versus a fall perpendicular to it. Therefore, in a real-life scenario robustness is highly increased when multiple cameras are used.

When using multiple cameras, a key question is when the fusion of the video data streams is performed in the overall detection process. In this paper we compare two basically different detection schemes which we label as *early fusion* and *late fusion*. Both schemes are illustrated in Figure 1.

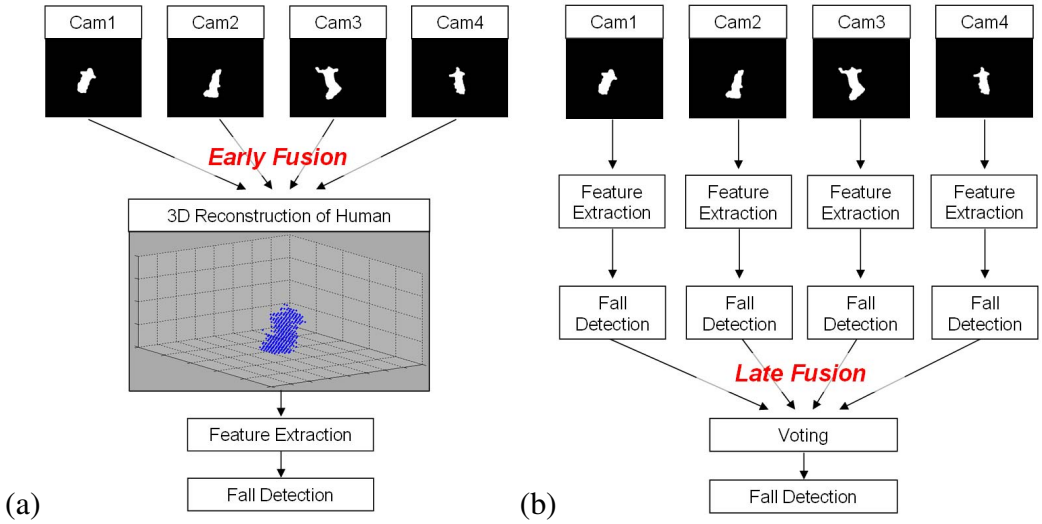

Figure 1. Fall detection with (a) early and (b) late fusion of the detected motion in four camera views.

In the early fusion scheme, detected motion in calibrated cameras is fused to obtain a 3D reconstruction of the human. This scheme follows previous works where Shape-from-Silhouette [4] and Particle Swarm Optimization [2] were used for reconstruction. These methods offer a robust estimation of the human posture and thus view-invariant features for fall detection. However, the drawback of 3D reconstruction is that it needs camera calibration and demands higher computational effort which handicaps the required real-time processing of the data.

In the late fusion scheme, feature extraction and fall detection is performed individually in each camera. In a final voting step the individual decisions are fused to an overall decision. This scheme tries to overcome the drawback of view-dependence of the extracted features by a well-adapted fusion strategy that needs no camera calibration and less computational effort.

In this paper we present two concrete approaches for fall detection which follow the early and late fusion scheme, respectively. The contribution of the paper is not to propose an overall vision system capable of detecting falls at home but rather to evaluate and discuss both fusion strategies as part of such a system. Whereas the presented early fusion scheme is basically an adaptation of the work proposed by Anderson et al. [4], we consider the late fusion scheme as a proposal to reduce its

computational efforts by exchanging the 3D reconstruction with a voting step.

The remainder of this paper is structured as follows. Section 2 describes the methodology of both fall detection approaches in detail. Experiments on a dataset of 73 video sequences are reported in Section 3. Conclusions are finally given in Section 4.

2 Methodology

In our methodology we focus on simplicity, low computational effort and therefore fast processing without the need of high-end hardware since the system has to be as cheap as possible to be affordable for the elderly. These design goals render, for instance, sophisticated model-based approaches for posture recognition infeasible. For both the early and late fusion approach, posture recognition is kept simple and estimates basically the general orientation of the human body, i.e. standing/vertical or lying/horizontal. This is achieved by combining the extracted features to confidence values for different posture states. Similar to [4], fuzzy logic [17] is used to compute the confidence values. In the late fusion approach, fuzzy logic is also used to fuse the confidence values of the various cameras to a final estimation of the state and for final fall detection.

In the following, the individual steps of our fall detection approaches are described. The steps of person detection (Section 2.1), feature extraction (Section 2.2) and estimation of posture and fall confidence values (Section 2.3) are nearly identical in both approaches. The only difference is on which kind of motion information the feature extraction is performed (2D pixels for late fusion and 3D voxels for early fusion) and whether posture and fall confidence values are estimated once (early fusion) or individually for each camera (late fusion).

2.1 Person Detection

Segmentation of the person from the background is the first step in our fall detection process. In the current state, person detection is kept simple since it is not our major concern. We apply simple background subtraction with a slowly adapting background model to detect motion. To remove noise from the motion image we make use of several morphological operations. Since in our test videos there is only one person in the room, we simply choose the largest connected component to mark the region representing the person.

2.2 Feature Extraction

We use a set of straightforward semantic driven features which was inspired by previous works [4, 8, 12, 13] and chosen based on empirical experiments. We discern between the intra-frame features which are computed within each frame and which focus on describing the character of the object, i.e. the posture, and a inter-frame feature which expresses the character of the change that happens between consecutive frames.

In particular, the following features are extracted at every frame with index i for early fusion (ef) and late fusion (lf) from the set of pixels ($\mathcal{P}_i$) or voxels ($\mathcal{V}_i$) representing the person:

- Intra-frame features
  - **Bounding Box Aspect Ratio** (B_i^{ef} and B_i^{lf}): The height of the bounding box surrounding the person divided by its width (for B_i^{lf}) or the mean of both its widths (for B_i^{ef}).

- **Orientation** (O_i^{ef} and O_i^{lf}): The orientation of the major axis of the ellipse fitted to the person, specified as the angle between the major axis and the x-axis (for O_i^{lf}) or the groundplane (for O_i^{ef}).
- **Axis Ratio** (A_i^{ef} and A_i^{lf}): The ratio between the lengths of the major axis and the minor axis of the ellipse fitted to the person. For A_i^{ef}, there are three axes and the ratio is computed between the longest axis and second longest axis.

- Inter-frame feature
  - **Motion Speed** (M_i^{ef} and M_i^{lf}): The relative number of new motion pixels/voxels in the current frame compared to the previous frame: $M_i^{ef} = |\mathcal{V}_i \backslash (\mathcal{V}_i \cap \mathcal{V}_{i-1})| \,/\, |\mathcal{V}_i|$ and $M_i^{lf} = |\mathcal{P}_i \backslash (\mathcal{P}_i \cap \mathcal{P}_{i-1})| \,/\, |\mathcal{P}_i|$.

2.3 Fuzzy-Based Estimation of Posture and Fall Confidence Values

In conformity with Anderson et al. [4], we define 3 posture states in which the person may reside: *"standing"*, *"in between"* and *"lying"*. Sets of primarily empirically determined fuzzy thresholds in the form of trapezoidal functions are assembled to interpret the intra-frame features and relate them to the postures. Thus, each feature value results in a confidence value in the range $[0, 1]$ on each posture, where the confidences of one feature sum up to 1 for all postures. These are then combined to assign a confidence value for each posture which is determined by a weighted sum of all feature confidences. The membership functions for the orientation O_i are exemplarily shown in Figure 2.

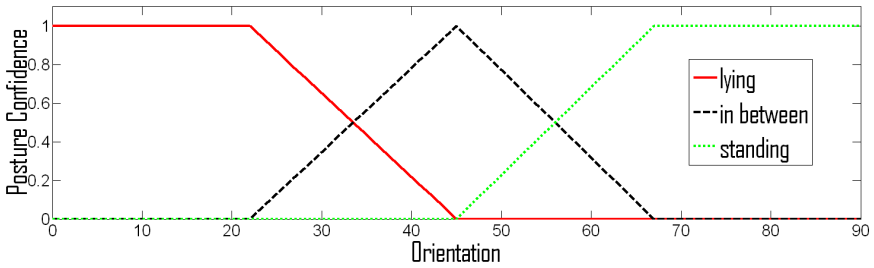

Figure 2. Membership functions for the three postures and the intra-frame feature O_i.

From the computed confidence values for the different postures, for every frame a confidence value for a fall event is computed. Therefore, we combine the intra- and inter-frame features with the assumption that a fall is defined by a relatively high motion speed, followed by a period with a *"lying"* posture. Thus, the confidence for a fall event at frame i is computed as the motion speed M_i multiplied by the confidence values for the posture *"lying"* for the next k frames.

2.4 Early Fusion

In the early fusion approach, a 3D reconstruction is computed from the set of motion pixels $\mathcal{P}_{c,i}$, where c is the camera index and i is the frame index. For this purpose Shape-from-Silhouette [6] is used, since we are able to apply this technique directly to the motion images from calibrated cameras and the achieved rough reconstruction is sufficient for our task of rough posture estimation, i.e. to differentiate between a lying and a standing posture. From all camera views c we have to find the intersection of the visual rays going through the points in $\mathcal{P}_{c,i}$. In order to keep the processing time within reasonable limits, a preprocessing step is applied which constructs a voxel list $L_c(m, n)$ for all image points (m, n) and all cameras c. The voxel list $L_c(m, n)$ stores all voxels $v(x, y, z)$ in the scene

that are intersected by the visual ray going through the image point (m, n) in the c-th camera. Once this voxel list has been build, every camera c defines a set of voxels $\mathcal{V}_{c,i} = \cup L_c(m, n)$ for all (m, n) in $\mathcal{P}_{c,i}$. The reconstruction $\mathcal{V}_i$ is finally obtained by an intersection test, i.e. $\mathcal{V}_i = \cap \mathcal{V}_{c,i}$ for all c.

2.5 Late Fusion

In the late fusion approach, the outputs of all cameras are combined to generate an overall decision. In our work, this is done by averaging the fall confidences from all the cameras. However, there is a weakness of this democratic voting strategy. There are cases when only one camera actually "sees" the fall. This can happen when the person falls in the direction of the optical axis of some of the cameras. In such a case cameras that are positioned along that axis will not recognize the fall at all, whereas a camera that is positioned in a perpendicular direction will have a clear view of the scene. A similar situation could arise in the presence of occlusions. Our solution in such case is that if the confidence of the alarm is very high (above a defined threshold) even in a single camera, this one camera gets "the right to over-vote the other" and the overall fall confidence is determined by this camera only.

3 Experiments

In order to thoroughly evaluate our fall detection methods, test sequences were acquired that follow the scenarios described by Noury et al. [10]. Hence, a testset consisting of various types of falls (forward and backward falls, falls from chairs etc.) as well as various types of normal actions (picking something up, sitting down etc.) was created. Four cameras with a resolution of 288×352 and frame rate of 25 fps were placed in a room at a height of approx. 2.5 meters. The four camera views are shown in Figure 3. Five different actors simulated the scenarios resulting in a total of 43 positive (falls) and 30 negative sequences (no falls).

In contrast to the definition given in [10], we consider falls ending on the knees as negative instances which the system should not detect as fall. The reason is that in this case people are whether still able to move, i.e. they would stand up, or would consequently lie down and thus the alarm would be initiated.

Figure 3 illustrates the process of fall detection for both fusion approaches and a given fall sequence. The particular sequence consists of a person simulating a fall from a chair. The graphs on the left show the confidence values for the posture *"lying"* and for a fall event over time. For the late fusion approach, the confidence values of the individual cameras as well as the fused confidence values are shown. For the early fusion approach, the confidence values extracted from the 3D reconstruction are shown. The fall starts approximately at frame number 540, and thus a peak in the fall confidence can be spotted at frame number 790, i.e. 250 frames later (please note that this "delay" is caused by $k = 250$). On the right side of Figure 3 some of the features extracted at frame number 790 are visualized: the bounding box (green) and fitted ellipse (blue) on the individual cameras and the bounding box on the 3D reconstruction of the person.

3.1 Results

Tests were performed on the overall dataset using the early fusion approach as well as the late fusion approach. Initial tests turned out that the whole temporal resolution is not needed in our case, hence a reduced frame rate of 5 fps was used. The parameter k defining the considered time period of the

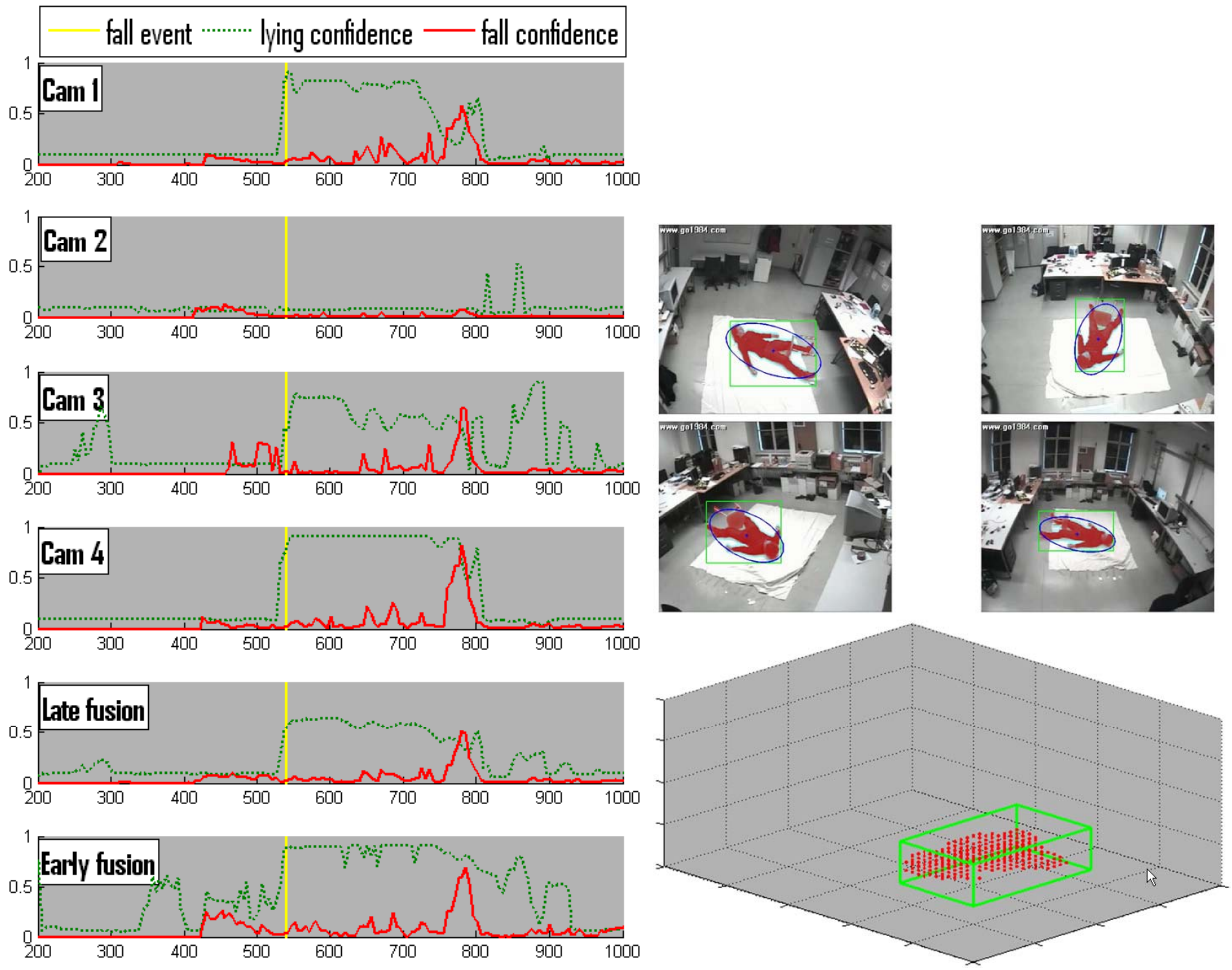

Figure 3. (a) Left: Confidence values for "lying" posture and a fall event over time; right: extracted features from the single cameras and 3D reconstruction of the person.

"lying" posture for fall detection (see Section 2.3) was set to 50 which corresponds to 10 seconds. Please note that in a real scenario this parameter has to be set to a higher value. In our simulated falls the lying periods are considerably shorter than they would be in case of a real fall event, for obvious reasons.

Since our methods result in confidence values for a fall event in every tested frame, we report true positive and false positive rate in the form of ROC curves in Figure 4. True positives and false positives were counted as the number of positive and negative sequences, respectively, where a fall confidence above the threshold could be found. It can be clearly seen that the early fusion approach outperforms the late fusion approach for the given setup. More precisely, early fusion comes to an *area under curve* of 0.935 whereas late fusion reaches only 0.876.

3.2 Discussion

The results show that the discriminative power of the chosen features is high enough to correctly classify the majority of the sequences using our fuzzy-based estimation of fall confidence values. Early fusion provides a more robust estimation of the person's posture than late fusion. The performance of late fusion is more dependent on the positioning of the cameras and inspection of the results reveals that for the given data misclassifications could have been avoided by a more optimal camera

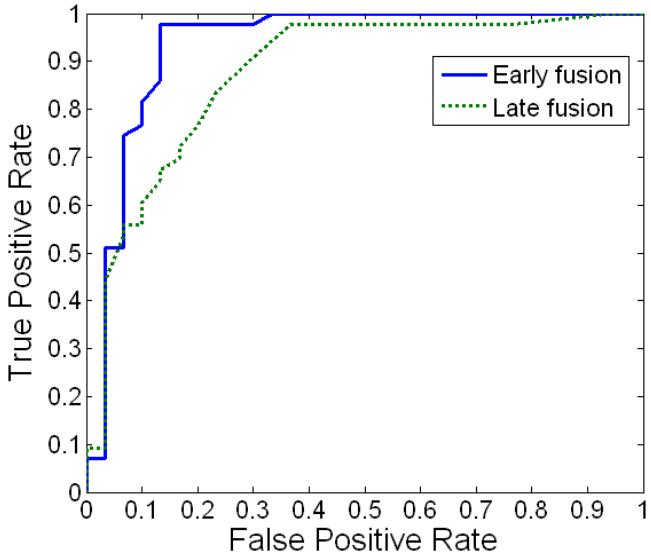

Figure 4. ROC curves for early and late fusion on the given data.

positioning. Another cause for misclassifications of both approaches is the limited overlap in the field-of-view of the cameras, as actions which are not totally visible in all four cameras renders the feature extraction and 3D reconstruction less robust.

4 Conclusions and Outlook

We have proposed and compared two methods for the detection of falls using multiple cameras in the context of ambient assisted living. In the early fusion method a 3D reconstruction is obtained using calibrated cameras, whereas in the late fusion method every camera detects falls individually and a final voting step leads to the overall decision.

Despite the better results achieved by the early fusion approach, we finally consider the late fusion as more appropriate for applying fall detection in multiple camera network at home. Late fusion needs less computational power and is easier to handle and implement (no camera calibration, possibility of parallel processing).

For this given test data, the experimental results have shown the general applicability of the proposed features and fusion approaches. However, especially the late fusion approach still bears potential for improvements. For instance, late fusion can be made more robust by considering the camera view for the definition of the membership functions for posture recognition, e.g. an orientation of $45°$ should contribute more to "lying" than to "standing". Also the rule for fall detection is rather simple at the moment and there is large space for extending the fall detection towards a more sophisticated reasoning.

In the future, prototype installations will show the real challenges of the very various environments and life styles of the elderly (overfilled flats, pets, dementia, active life style (e.g. exercising), visitors, etc.). Arguably, the manual or automatic definition of inactivity zones [9] will be necessary to make the system more robust against normal sitting and lying actions. Special care has to be taken towards the arrangement of the cameras in the rooms. Future research will also go into this direction, e.g. to

determine the necessary number of cameras and their optimal positioning.

References

[1] *The Life of Women and Men in Europe : A Statistical Portrait*. Eurostat, 2008 edition, 2008.

[2] H. Aghajan, C. Wu, and R. Kleihorst. Distributed Vision Networks for Human Pose Analysis. *Signal Proc. Techniques f. Knowledge Extraction and Inf. Fusion*, pages 181–200, 2008.

[3] D. Anderson, J.M. Keller, M. Skubic, X. Chen, and Z. He. Recognizing falls from silhouettes. In *Proc. of EMBS*, pages 6388–6391, 2006.

[4] D. Anderson, R.H. Luke, J.M. Keller, M. Skubic, M. Rantz, and M. Aud. Linguistic summarization of video for fall detection using voxel person and fuzzy logic. *CVIU*, 113(1):80–89, 2009.

[5] R. Cucchiara, C. Grana, A. Prati, and R. Vezzani. Probabilistic posture classification for human-behavior analysis. *SMC-A*, 35(1):42–54, 2005.

[6] C.R. Dyer. Volumetric scene reconstruction from multiple views. *Foundations of Image Understanding*, pages 469–489, 2001.

[7] C. Huang, E. Chen, and P. Chung. Fall detection using modular neural networks with back-projected optical flow. *BME*, 19(6):415–424, 2007.

[8] C.W. Lin, Z.H. Ling, Y.C. Chang, and C.J. Kuo. Compressed-domain Fall Incident Detection for Intelligent Homecare. *VLSISP*, 49(3):393–408, 2007.

[9] H. Nait-Charif and S.J. McKenna. Activity summarisation and fall detection in a supportive home environment. In *Proc. of ICPR*, volume 4, pages 323–326, 2004.

[10] N. Noury, A. Fleury, P. Rumeau, AK Bourke, GO Laighin, V. Rialle, and JE Lundy. Fall detection–Principles and methods. In *Proc. of EMBS*, pages 1663–1666, 2007.

[11] C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau. Monocular 3D head tracking to detect falls of elderly people. In *Proc. of EMBS*, pages 6384–6387, 2006.

[12] C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau. Fall detection from human shape and motion history using video surveillance. In *Proc. of AINAW*, volume 2, 2007.

[13] J. Tao, M. Turjo, M.F. Wong, M. Wang, and Y.P. Tan. Fall incidents detection for intelligent video surveillance. In *Proc. of ICICS*, pages 1590–1594, 2005.

[14] B.U. Toreyin, Y. Dedeoglu, and A.E. Çetin. HMM based falling person detection using both audio and video. *LNCS*, 3766:211, 2005.

[15] D. Wild, U.S. Nayak, and B. Isaacs. How dangerous are falls in old people at home? *Br Med J*, 282(6260):266–268, 1981.

[16] J. Willems, G. Debard, B. Bonroy, B. Vanrumste, and T. Goedemé. How to detect human fall in video? An overview. *Proc. of PoCA*, 2009. to be published.

[17] LA Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.