Fusion of Data from Multiple Cameras for Fall Detection

Jana MACHAJDIK [a,1], Sebastian ZAMBANINI [a] and Martin KAMPEL [a]

[a] *Computer Vision Lab, Institute of Computer Aided Automation, Vienna University of Technology, Austria*

Abstract. In the context of ambient assisted living, camera-based fall detectors at elderly homes leads to immediate alarming and helping. In this paper we propose a novel approach for the detection of falls based on multiple cameras. Based on semantic driven features fall detection is done in 2D and each camera decides on its own, if a fall has occurred. 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 where the individual decisions are fused to an overall decision. Emphasis is given on simplicity, low computational effort and fast processing. We demonstrate the method and give results on 73 video sequences.

Keywords. Data Fusion, Smart Home, Fall Detection

1. Introduction

In the European Union about 30 % of people older than 65 live alone [1]. For these people, falls at homes are one of the major risks and an immediate alarming and helping is essential to reduce the rate of morbidity and mortality [13]. 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 emerged [14]. A basic 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 Hidden Markov Models (HMMs) are trained with simple features like projection histograms [5] or the aspect ratio of the bounding box surrounding the detected human [3,12]. However, the applicability of this methods in real-life scenarios is limited due to the high diversity of fall actions and the high number of different 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

[1] Corresponding Author; E-mail: jana@caa.tuwien.ac.at

from normal actions like sitting on a chair or lying on a bed. In this manner, various features have been used for camera-based fall detection, including the aspect ratio of the bounding box [11] or orientation of a fitted ellipse [10] for posture recognition and head tracking [9] or change rate of the human's centroid [7] for motion velocity. Apart from the features used, the methods also differ in the way how the final decision is obtained from the features. Besides parametric classifiers like Neural Networks [6], primarily empirically determined rules are applied [4,7,9,10]. If the method is vulnerable to 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 believe 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 position 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 various data streams is performed in the overall detection process. Using an early fusion scheme, detected motion can be fused together from calibrated cameras to obtain a 3D reconstruction and posture estimation of the human, e.g. by shape-from-silhouette [4]. Thereby, the human posture can be robustly estimated and the extracted features are independent of camera position. However, these methods need camera calibration and demand high computational effort which restrains the required real-time processing of the data.

In this paper we propose a late fusion approach in which every camera detects falls individually. 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 where the individual decisions are fused to an overall decision. Therefore, the drawback of view-dependence of the extracted features is overcome by a fusion strategy that needs no camera calibration and less computational effort.

The remainder of this paper is structured as follows. Section 2 describes the methodology of our fall detection approach. Experiments on a data set of 73 video sequences are reported in Section 3. Conclusions are finally given in Section 4.

2. Methodology

In this work we make use of a late fusion approach, performing analysis of the scene on each camera individually and then combining the individual results to get an overall decision. In contrast to other works (e.g. [2]), our posture estimation does not rely on image features such as color and texture. It only needs silhouettes of the human as input which makes the system less vulnerable to low-quality images. We define empirical, semantic driven rules to make the decisions using features with fuzzy boundaries to analyze the scene. 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. Our main condition on the performance is the usability as a real-time application. Another desirable property is that the construction/structure of the decision making part of the software is understandable to humans, so that more rules can be added easily to adapt or expand the system as new, unforeseen challenges are encountered in the real world.

Our approach is implemented as follows: first, motion detection is performed on the video to segment the person from the background. In our case simple background subtraction with a slowly adapting background model is used. To remove noise from the motion mask image we make use of morphological operations. First, dilating with a short vertical line as structuring element is performed to connect the blob of the upper body with the lower body of the person, as in our test videos it was often disconnected due to camouflaging effects arising from dark horizontal elements in our testing environment. Further we perform a series of opening operations to remove all blobs that are smaller than a set threshold. Since in our test videos there is only one person in the room, we simply choose the largest blob (above a certain threshold) to mark the region representing the person. After this procedure we have a mask with a rough silhouette of the person in the video. This is used to compute the features.

We implement a collection of straightforward semantic driven features, many of which have already been used before [4,7,10,11]. These can be divided into two groups, the intra-frame features which are computed within each frame and which focus on describing the character of the object area, i.e. the posture, and inter-frame features which express the character of the change that happens between consecutive frames and usually measure the motion velocity. As intra-frame features, we compute the area of the object of interest, the aspect ratio of the height and width of the bounding box, the orientation of the major axis of a fitted ellipse as well as the aspect ratio of the major and minor axis of the ellipse (as illustrated in Figure 1). Since we work in the 2D space with uncalibrated cameras, we can not take advantage of absolute measures such as the height of the bounding box, as these are highly dependent on the distance of the object to the camera and also the viewpoint of the camera.

Moreover, we have to introduce some corrective measure to account for mistakes due to the narrow field of view of a camera. It can happen that a person goes partly out of the frame, so the blob of the motion frame gets distorted and does not fulfill the expectation of a body in the proper posture anymore. In our case we introduce such a correction by simply defining a field of ignorance around the lower border of the camera image. When the person moves too near to the border we just ignore the blob and count on the other cameras that see the spot “properly” to assess the situation. With a more sophisticated implementation of object tracking with person detection, this step will either become superfluous or can be solved more elegantly.

To assess the character of the motion of the object we compute several inter-frame features. We compute the distance between centroids of the blobs of consecutive frames, the difference between the consecutive moving object areas relative to the area of the later frame motion area, and the change of orientation.

Further, we define 4 postures/states in which a person may reside: standing, lying, “in-between” (i.e. sitting, kneeling, bending, etc.) and falling. Sets of primarily empirically determined fuzzy thresholds in the form of trapezoidal functions are assembled to interpret the features and relate them to the postures. Each feature’s 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/state which is determined by a weighted sum of all feature confidences. For instance, during a fall event the transition between postures is usually accompanied by

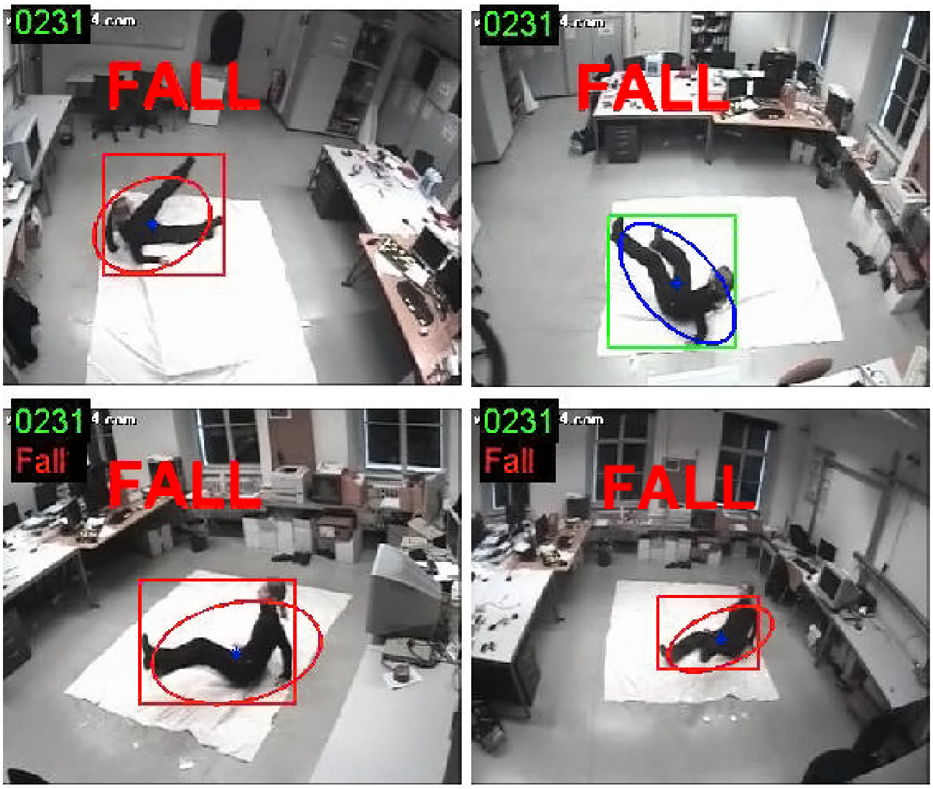

Figure 1. The four camera views of the test sequences with a person, showing the bounding box and fitted ellipse.

fast motion, which in our case is expressed as a combination of large centroid distance, high motion speed, and large change in orientation. A lying position combines a low bounding box and ellipse axis ratio with an major orientation about 0 degrees relative to the x-axis, etc. Therefore, we get a set of 4 confidence values for the 4 postures/states for each frame, similar to [4].

The final decision about the current posture for a single camera is made by taking the posture with the maximum confidence. At this point, we start analyzing the sequence of postures with the goal to send an alarm when the person falls down and can not get up again. It is especially important to prevent cases with a “long lie”, i.e. when an old person remains on the floor for more than one hour because he or she can not get up again. Statistics [13] show that half of such cases end in death of the victim within six months, mostly due to complications, such as dehydration or lung infection from a cold floor. To recognize the event of falling down and not getting up, a set of empirically derived semantic rules are applied, similarly to the works in [4,7,9,10]. In our case the rules are relatively simple and straightforward. An analysis of last n frames is conducted. If at least k (where $k < n$) of them (this introduces robustness against noise or errors) represent a lying or a falling posture, and the confidence of the posture is above a threshold τ, then an alarm is initiated. The confidence of the alarm is computed by averaging the confidences of the lying and falling postures within the active n frames. We also have the possibly to check the duration of the lay, if the previous state was in-between or standing or, if really a falling event occurred at the beginning of the lay, to increase or decrease the confidence of the alarm. The occurrence of the falling event is not a compulsory condition, since people may, in some cases, "fall" slowly.

Finally, we combine the input of all cameras with overlapping field of view to generate an overall decision. In our work this is done by averaging the posture confidences from all the cameras and then applying the same decision rules described above. However, there is a weakness to this democratic voting strategy. There are cases when only one camera sees the fall. This can happen if, e.g. 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 ($> 90\%$) even in a single camera, this one camera get "the right to over-vote the others", i.e. the alarm will still be initiated by the single strong vote.

3. Experiments

In order to thoroughly evaluate our fall detection method, test sequences were acquired that follow the scenarios described by Noury et al. [8] and added some more of our own scenarios. Hence, a challenging test set 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 IP cameras with a resolution of 288×352 were placed in a room at a height of approx. 2.5 meters. The four camera views are shown in Figure 1. Five different actors simulated the scenarios resulting in a total of 49 positive fall sequences and 24 negative sequences as categorized by Noury et al. [8]. However the definition of fall by [8] does not fully coincide with the goal of our application, neither is it a complete account of fall situations, as it misses, e.g. falls from sitting etc. In our field of application we do not want to recognize only the explicit falling event, but rather the scenario where a person falls and cannot get up again. Such "falls" can also happen slowly, e.g. the person starts feeling faint, losing their strength and slowly sliding down to the ground, possibly trying to stabilize themselves by touching the wall or a nearby piece of furniture, or person with dementia, going to sleep on kitchen floor. Such cases are defined as negative by Noury et al. [8], but in our case they should be positive, and on the other hand falls ending on the knees leave the person able to move (if they are not able to stand up, they can still lie down to initiate the alarm) so in our opinion they should not initiate an alarm. Therefore, we consider these disputable cases as a kind of gray area scenarios (there are 10 such videos). However, to present a statistic about the performance, we need to assign them a clear alarm/non-alarm category, so after our re-categorization we have 47 "alarming" (fall) sequences, and 26 normal/non-alarming activity sequences, whereby numerous normal activities are done in each sequence. Statistically, our algorithm performs as shown in Table 1.

	CORRECT	FALSE/MISSED ALARM	SUM
ALARMING (FALL)	40	7	47
NON-ALARMING	25	1	26
SUM	65	8	73

Table 1. Experimental results statistics.

However, for a better understanding we need to look beyond the statistic and at least discuss the specific scenarios where the method failed. To further enhance the understanding of the issues, the videos we are discussing can be viewed at *http://www.cogvis.at/mubisa/fallscenarios.html*

First, let us mention the mistakes that happened due to our primitive person detection. In several scenes a different object, such as a moved chair was chosen to be the object of interest instead of the person. Such errors will have to be corrected by implementing a better person tracking algorithm which will be our next step in the future. A second kind of reason why the alarm did not initialize are cases where the person got up immediately after the fall and therefore the time of the "lay" was shorter than the set threshold. With our goal, this might not even be considered a fail. A more challenging scenario, however, occurs when the fallen person stays on the floor in a curled up or in a too spread out positions, which no longer fit the expected flat and long figure with a clearly horizontal orientation. A scene which was falsely classified as alarming adds to this group of challenges. In that case a person crawling on the floor on all four set off the alarm. These cases present a challenge that will have to be examined more closely and tackled by extending our set of rules to cover such possibilities.

A special challenge that we solved was already mentioned: from our 4 cameras in a room, two were on the same wall, another on the opposite wall, looking approximately along the same optical axis are the first two, and one camera was on the third wall with a view axis perpendicular to the others. Several scenes in the set contain a fall in the direction of the optical axis of 3 of the cameras, so that only one camera saw the fall clearly.

4. Conclusions

We have proposed a method for the detection of falls in the context of ambient assisted living. Our approach for identifying a person's fall is based on motion detection followed by the computation of semantic driven features. We have presented a late fusion approach in which every camera detects falls individually. In a final voting step the individual decisions are fused to an overall decision. In order to allow a comparison to an early fusion scheme further research is directed towards experiments with calibrated cameras to obtain a 3D reconstruction.

Acknowledgements

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

References

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

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

[3] D. Anderson, J.M. Keller, M. Skubic, X. Chen, and Z. He. Recognizing falls from silhouettes. In *International Conference of the Engineering in Medicine and Biology Society*, 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. *Computer Vision and Image Understanding*, 113(1):80–89, 2009.

[5] R. Cucchiara, C. Grana, A. Prati, and R. Vezzani. Probabilistic posture classification for human-behavior analysis. *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, 35(1):42–54, 2005.

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

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

[8] N. Noury, A. Fleury, P. Rumeau, AK Bourke, GO Laighin, V. Rialle, and JE Lundy. Fall detection–Principles and methods. In *International Conference of the Engineering in Medicine and Biology Society*, pages 1663–1666, 2007.

[9] C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau. Monocular 3D head tracking to detect falls of elderly people. In *International Conference of the Engineering in Medicine and Biology Society*, pages 6384–6387, 2006.

[10] C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau. Fall detection from human shape and motion history using video surveillance. In *21st International Conference on Advanced Information Networking and Applications Workshops*, volume 2, pages 875–880, 2007.

[11] J. Tao, M. Turjo, M.F. Wong, M. Wang, and Y.P. Tan. Fall incidents detection for intelligent video surveillance. In *Fifth International Conference on Information, Communications and Signal Processing*, pages 1590–1594, 2005.

[12] B.U. Toreyin, Y. Dedeoglu, and A.E. Çetin. HMM based falling person detection using both audio and video. *Lecture Notes in Computer Science*, 3766:211–220, 2005.

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

[14] J. Willems, G. Debard, B. Bonroy, B. Vanrumste, and T. Goedemé. How to detect human fall in video? An overview. *International Conference on Positioning and Context-Awareness*, 2009.