Monocular Camera Fall Detection System Exploiting 3D Measures: A Semi-supervised Learning Approach

Konstantinos Makantasis[1], Eftychios Protopapadakis[1], Anastasios Doulamis[1], Lazaros Grammatikopoulos[2], and Christos Stentoumis[3]

[1] Technical University of Crete, 73100 Chania, Greece
{konst.makantasis,eft.protopapadakis}@gmail.com, adoulam@cs.ntua.gr
[2] Technological Educational Institute of Athens, 12210, Athens, Greece
lazaros.pcvg@gmail.com
[3] National Technical University of Athens, 15773, Athens, Greece
cstent@mail.ntua.gr

Abstract. Falls have been reported as the leading cause of injury-related visits to emergency departments and the primary etiology of accidental deaths in elderly. The system presented in this article addresses the fall detection problem through visual cues. The proposed methodology utilize a fast, real-time background subtraction algorithm based on motion information in the scene and capable to operate properly in dynamically changing visual conditions, in order to detect the foreground object and, at the same time, it exploits 3D space's measures, through automatic camera calibration, to increase the robustness of fall detection algorithm which is based on semi-supervised learning. The above system uses a single monocular camera and is characterized by minimal computational cost and memory requirements that make it suitable for real-time large scale implementations.

Keywords: image motion analysis, semisupervised learning, self calibration, fall detection.

1 Introduction

According to medical records, traumas resulting from falls have been reported as the second most common cause of death for the elderly and as the most important problem that hinders these people's ability to live an independent life. Therefore, a major research effort has been conducted for automatically detecting persons' falls, either through the use of specialized equipment, (i.e. accelerometers, floor vibration sensors, gyroscope sensors, barometric pressure sensors, sound sensors) [1–4] or through visual cues by using cameras [5–12].

Techniques based on specialized equipment, require the use of wearable devices that should be attached to the human body and thus their efficiency relies on persons' ability and willingness to wear them. On the other hand, vision based systems, a more research challenging alternative due to the complexity of visual

A. Fusiello et al. (Eds.): ECCV 2012 Ws/Demos, Part III, LNCS 7585, pp. 81–90, 2012.

content and the fact that a fall should be discriminated than other ordinary humans' activities (i.e. sitting, lie down), present several advantages. They are installed on buildings and not worn by users, are able to detect multiple events simultaneously and the recorded video can be used for post verification analysis. Towards this direction, the works of [6, 7, 11] exploit foreground object's shape, as well as, its vertical motion velocity to detect a fall incident. The authors of [8] and [5] use a wider set of features along with SVM to detect falls. In system of [12] a classifier, capable to discriminate six indoor human activities, is trained based on knowledge derived from human anatomy body parts ratios. Although, experimental results of the above works show high detection rates none of these exploits 3D information to increase system robustness. A 3D active vision system based on Time of Flight cameras is proposed in [9]. However, the measures that are provided by this type of cameras could be affected by reflectivity objects properties and aliasing effects when the camera-target distance overcomes the non-ambiguity range. In [10] a multi camera system that exploits stereo vision is proposed. 3D processing, though more robust than a 2D image analysis in terms of fall detection and discrimination among falls and other daily activities; requires high computational cost that usually making these systems unsuitable for real-time, large scale implementations.

In this paper, a new innovative approach is presented that exploits, on the one hand, monocular cameras to detect falls in real-time and, on the other, it is capable to exploit actual 3D space's measures, through camera calibration and inverse perspective mapping (IPM), to increase system's robustness for a wider range of camera positions and mountings compared to other 2D fall detection methods ([7, 11]). The fall detection algorithm discriminates fall incidents by using a non-Linear Warning System (nLWS) based on semi-supervised learning. Our system, due to its minimal computational cost and memory requirements, let alone its low financial cost since ordinary low-resolution cameras are used, making it affordable for a large scale.

1.1 Problem Formulation

Fall incidents can be discriminated than other human activities by using human motion and posture analysis. Information about humans' posture can be derived by the height-width ratio; in a 3D space this ratio is smaller in value when a fall event occurs compared to humans in standing position. In addition, as explained in [8] and [5], the ellipse that bounds the foreground object and more specifically the angle of its major semi-axis can provide useful information about human posture. This angle is close to 90 ° when the human is in standing position and close to 0 ° after a fall incident. Considering on motion information, the most commonly used feature is vertical motion velocity, which during a sequence of frames can be expressed by Eq.(1)

$$V = \sum_{i=k-m}^{k} h_a(i) - h_a(i-1) \tag{1}$$

$h_a(k)$ stands for the actual height of a human in 3D space at the k^{th} image frame. Vertical motion velocity is calculated for a sequence m of frames and is an estimation of the speed of the motion and also an evident of how severe would be a fall. Instead of humans' projected height, we choose to use their actual height, measured in physical world units (e.g., cm, inches), since: (a) this yields a more robust performance not affected by cases where the human is far away or very close to the camera, (b) actual height can provide information about the moving object, making the system capable to discriminate if the moving object might be a human or something else and (c) system's performance improves for a wider range of camera positions and mountings.

In order to extract all these features, first of all, a foreground extraction algorithm has to be used to extract the foreground object, which initially is unknown (Section 2). Width-height ratio computation requires information about left-most q_{lm}, right-most q_{rm}, top-most q_{tm} and bottom-most q_{bm} points of foreground object, to calculate its projected height and width (Section 2.1), while the bounding ellipse can be approximated by using image moments for the foreground object's area (Section 2.2).

Representation of an object on camera's plane is presented in Fig.1(a). It appears that the actual height of foreground object can be given through Eq.(2), if cameras focal length f, distance Z between the camera and foreground object and foreground object's projected height h_p are known.

$$h_a = Z \frac{h_p}{f} \tag{2}$$

h_p can be obtained in the same way as width-height ratio, f can be automatically obtained through camera self-calibration (Section 3) and Z can be obtained through the construction of a reference plane Fig.1(c) that is the orthographic view of the floor Fig.1(b) (Section 3.1).

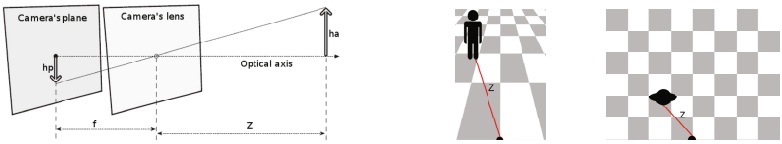
1.2 Proposed Contribution

The main contributions of the proposed system are:

3D measures exploitation using a single monocular camera. Using a single monocular camera along with the exploitation of 3D measures, give our system the opportunity to detect falls in real-time, like 2D fall detection methods, and to approximate the robustness of 3D ones.

Semi-supervised learning fall detection algorithm. A semi-supervised learning approach, let the fall detection algorithm that exploits a non Linear Warning System (nLWS), to be effectively trained by calling an expert to further refine an initially unsupervised created small subset of labeled samples.

Self calibrated fall detection system. The proposed fall detection system utilizes a self calibration technique, to estimate camera parameters, based on the detection of vanishing points. Using of self-calibration creates a *fully* automatic



(a) (b) (c)

Fig. 1. (a) object representation on camera's plane, (b) floor plane, (c) reference plane

system capable to operate properly without the need of any human user manual configuration.

The rest of this paper is organized as follows: in Section 2 the foreground extraction algorithm is presented along with 2D features extraction, Section 3 presents self calibration method and 3D features extraction, in Section 4 the fall detection algorithm is presented, Section 5 concerns on experimental results and, finally, Section 6 concludes this work.

2 Foreground Extraction and 2D Features

For foreground extraction, we used the technique described in [11], which is based on an iterative scene learning algorithm. This is a lightweight foreground extraction algorithm, with minimal computational cost and memory requirements, capable to operate properly in real-time and in complex, dynamic in terms of visual content and unexpected environments.

It uses the "pyramidal" Lucas-Kanade algorithm to estimate the intensity of motion vectors in a scene, along with their directions, in order to identify humans' movements. Motion information is used as a background updating mechanism, according to which the background is updated at every frame instance by using motion vectors within high motion information areas. If motion vectors' intensity exceeds a threshold then this area is denoted as foreground, otherwise it is denoted as background Fig.2(b).

2.1 Foreground Object's Projected Height-Width Ratio

The first step for width-height ratio computation, is the estimation of foreground object's projected height and width. These two measures can be estimated by the four corners of a minimum bounding box, Fig.2(b), that includes the foreground. The four corners of the minimum bounding box are associated with foreground object's q_{lm}, q_{rm}, q_{tm} and q_{bm} points and thus height width ratio can be expressed by Eq.(3)

$$R = \frac{h_p}{w_p} = \frac{q_{tm} - q_{bm}}{q_{rm} - q_{lm}} \tag{3}$$

where w_p and h_p stand for the projected width and height of foreground object.

(a) (b) (c) (d)

Fig. 2. (a) original frame, (b) minimum bounding box, (c) approximated ellipse for standing position, (d) approximated ellipse after a fall incident

2.2 Bounding Ellipse Major Semi-axis Angle

For the approximation of an bounding ellipse that includes the foreground object, we based on image moments that can successfully describe objects' properties after segmentation. An ellipse is defined by its center $(\bar{x}, \bar{y})$, its orientation, which is the angle θ of its major semi-axis and the lengths a and b of its major and minor semi-axes.

For a scalar image with pixel intensities *I(x,y)*, spatial image moments are given by

$$M_{ij} = \sum_x \sum_y x^i y^i I(x,y) \quad for\ i,j = 0,1,2,... \tag{4}$$

The center of the ellipse coincides with the center of mass of foreground object and can be obtained by

$$(\bar{x}, \bar{y}) = \{M_{10}/M_{00}, M_{01}/M_{00}\} \tag{5}$$

The orientation θ of the ellipse can be computed with the central moments of second order. Computation of central moments is based on the centroid $(\bar{x}, \bar{y})$ and is given by

$$\mu_{ij} = \sum_x \sum_y (x-\bar{x})^i (y-\bar{y})^j I(x,y) \quad for\ i,j = 0,1,2,... \tag{6}$$

and orientation θ is given by

$$\theta = \frac{1}{2} \arctan\left(\frac{2\mu_{11}}{\mu_{20} - \mu_{02}}\right) \tag{7}$$

Approximated ellipses for standing position and after a fall incident are shown in Fig.2(c)-(d).

3 Camera Self-calibration

Camera calibration is necessary to obtain camera's focal length, which is required for actual human's height approximation. Our system by using a single stationary monocular camera, only a single view of the scene is available. In order to use an

automatic calibration technique three finite vanishing points that correspond to three mutually orthogonal planes of the scene are necessary. This can be achieved when camera's plane is not parallel to any of these three planes.

The image plane is chosen as the accumulator space, while the intersections of all pairs of image line segments represent potential vanishing points. Since vanishing points in 3D scene are points at infinity, vanishing points in the 2D image cannot lie on line segments. All potential vanishing points that do not satisfy this constraint are removed. For the rest of them the contribution of every line segment is computed by means of a voting scheme. Next, all potential vanishing points are checked against certain geometrical criteria for the determination of the three dominant vanishing points of mutually orthogonal space directions. Only triplets of vanishing points forming acute triangles are considered (orthogonality criterion) and principal point, orthocenter of the triangle, as well as, computed camera constant should have "reasonable" values (camera criterion). Vanishing point triplets that satisfy these criteria are sorted according to their total score; that with the highest score is chosen as the final triplet of dominant vanishing points.

Radial lens distortion at any image point (x_d, y_d) can be modeled by the first two coefficients of a Taylor series around $r = 0$, where r is the distance between point (x_d, y_d) and principal point (x_o, y_o). Radial lens distortion is given by

$$x_u = x_d + x_d(k_1 r^2 + k_2 r^4) \quad and \quad y_u = y_d + y_d(k_1 r^2 + k_2 r^4) \tag{8}$$

(x_u, y_u) is the undistorted point corresponding to distorted point (x_d, y_d). The observed lines are constraint to converge to their corresponding vanishing point $V(x_v, y_v)$ according to the following equation [13]

$$[x-(x-x_o)(k_1 r^2+k_2 r^4)-x_v]cos\phi+[y-(y-y_o)(k_1 r^2+k_2 r^4)-y_v]sin\phi = 0 \tag{9}$$

where (x, y) are the image coordinates of an individual point on a line and (x_o, y_o) is the principal point. Coefficients k_1 and k_2 are computed so as the root mean square distance of points (x, y) from the fitted line is minimized. To estimate principal point (x_o, y_o) and camera constant (focal length) c, each pair of orthogonal vanishing points, $\mathbf{v_1}$ and $\mathbf{v_2}$, expressed in homogeneous coordinates, supplies a linear constraint on the entities of conic ω of the form

$$\mathbf{v_1}^{\mathbf{T}} \omega \mathbf{v_2} = 0 \tag{10}$$

by ignoring image aspect ratio and skewness, ω may be written as:

$$\omega = \begin{bmatrix} 1 & 0 & -x_o \\ 0 & 1 & -y_o \\ -x_o & -y_o & {x_o}^2 + {y_o}^2 + c^2 \end{bmatrix} \tag{11}$$

3.1 Reference Plane Construction - Height Approximation

For a projective space $\wp^n$ a projective homography is defined as a nonsingular matrix $\mathbf{H_{(n+1)x(n+1)}}$ with elements belonging to an affine space $\Re^n$, and defined

up to a certain scalar value. A point $\mathbf{x}$ is projectively transformed to $\hat{\mathbf{x}}$ as follows:

$$\hat{\mathbf{x}} = \mathbf{H}\mathbf{x}, \quad \mathbf{x}, \hat{\mathbf{x}} \in \wp^n \tag{12}$$

where $\mathbf{H}$ is the coordinate transformation matrix (homography matrix). According to the IPM algorithm, described in [14], $\hat{\mathbf{x}}$ and $\mathbf{x}$ can be expressed as:

$$\hat{\mathbf{x}} = [\hat{x}\ \hat{y}\ 1] \quad and \quad \mathbf{x} = [x\ y\ 1] \tag{13}$$

where $x, y, \hat{x}, \hat{y}$ represent Cartesian coordinates on image plane and reference plane respectively and homography matrix $\mathbf{H} = [h_{ij}]$ is a 3x3 matrix, normalized so to have $h_{33} = 1$. Eq.(12) requires at least four non collinear points in order to be solved. By using a known target or ground truth images, a larger set of points can be found and this equation can be solved in a least square sense. This equation represents a perspective transformation, any parallelogram can be transformed to any trapezoid and vice versa.

To approximate the distance Z between foreground object and camera, we use the q_{bm} point. On the reference plane the relation between camera's natural units (pixels) and the units of the physical world (cm) is linear and thus Z is straightforwardly calculated. This results in a simple model and a single solution in which a point in the 3D world (X, Y, W) with actual height h_a is projected on the image plane with projected height h_p in accordance with Eq.(14).

$$h_a = Z\frac{h_p}{c} \tag{14}$$

The appearance of errors during perspective transformations affects the h_a estimation (Fig.3), as it depends on distance estimation on created reference plane.

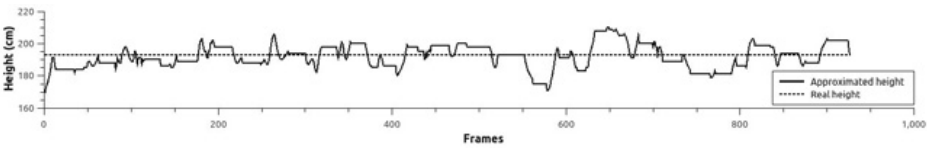


Fig. 3. Real height approximation

4 Fall Detection Algorithm

Fall detection algorithm utilizes a non Linear Warning System (nLWS). The nLWS is a feed forward neural network, topologically optimized (i.e. number of hidden layers, neurons, activation function), using an island genetic algorithm. The system's inputs are vectors of size 3x1. The first and second elements are calculated as the change at the angle (degrees) and at the height (cm) for the last four frames, respectively. The third input is the ratio between foreground

object's projected height and width (both in pixels). The target outputs were logical values $\in \{0, 1\}$, where 1 suggests that a fall has happened.

System initialization - The training procedure takes place offline. Initially the input vectors are separated in two classes. The separation is performed unsupervised using a similarity based classifier (i.e. kmeans) using various metric distances. A weighted average sum function of the results (for each of the metric distances) is used to decide in which class each frame should be placed. An assumption is made that the fewer-element class describes the possible falls. Subsequently, the following semi-supervised training procedure occurs: initially, the vectors are labeled according to the suggested class. At this stage the training sample for the nLWS is formed and training procedure takes place. Then, for the non-fall class, an expert is summoned to further refine it by removing at random non-fall describing vectors. Finally, the nLWS weights and biases are adapted at the new values. For any given input vector the nLWS output is expected at $[0, 1]$. Values close to 1 mean that it is more likely to observe a fall.

Evaluation Phase - The fall detection mechanism is based on threshold value. Initially, that value is set as the average of the n greatest output prices according to the training sample (n was set at $40 \mapsto 6\%$ of the training set). The vectors are fed one by one at the nLWS. If a score greater than the threshold is achieved then a fall tag is given at the specific input vector. For the following frames the same procedure is followed. If within a range of 10 sequent frames more than 3 positive-fall tags are observed then we have detected a fall (it is assumed that the duration of a fall incident is 3 to 4 frames for real time operation - at least 20fps). That range scheme is based on the input vector creation; similar vectors will be created around the fall time and will produce higher output values.

5 Experimental Results

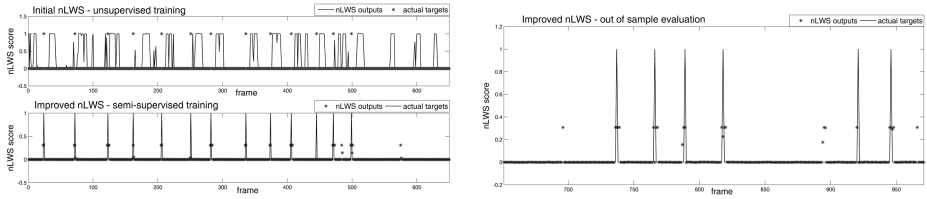
During the experiments one person simulated falls, in every direction according to the camera position and normal every day activities, that may look like falls but they are not, Fig.4. Self calibration method was compared to calibration method that comes with OpenCV and both algorithms present the same results for radial distortion and camera constant. The fall detection algorithm was tested in dynamically changing visual conditions, including illumination changes, cluttered background and occlusions at a martial arts school in Chania and at a demo room in Trikala municipality for 5 and 9 days respectively.

Our system operates in real-time at 20fps and as experimental results suggest that algorithm detects over 90% of fall incidents and presents very low false positive rate, which is not crucial when post verification video analysis is available. In sample and out of sample algorithm's performance is presented in Fig.5. The performance of fall detection scheme is presented in Table 1(a). Its performance is affected by foreground extraction, and this results to more robust performance for indoor environments, the impact of occlusions is being reduced as camera's height is being increased and in contrast to other approaches (e.g. [8]), its performance is not affected by the direction of falls. Table 1(b) presents false positive

rates for different activities. Lie on the floor presents the biggest false positive rate, however, this activity can't be thought as "normal".

(a) (b)

Fig. 4. (a) falls, (b) normal activities



(a) (b)

Fig. 5. (a) In sample performance, (b) Out of sample performance

Table 1. (a) Performance (b) Total false positive rate divided in regard to activities

(a)

Camera's height (cm)		Proposed system	Indoor No occlusion	Indoor with occlusion	Outdoor
40	*Falls detected*	90 %	98 %	76%	83 %
	Wrong detections	4	2	5	9
220	*Falls detected*	93 %	97 %	92%	82 %
	Wrong detections	6	4	7	8
260	*Falls detected*	97 %	97 %	94%	82 %
	Wrong detections	3	3	4	6

(b)

Activity	False positive
Lie down	62,5%
Sit on floor	25%
Other	12,5%

6 Conclusions

This paper presents a fall detection scheme that exploits 3D measures by using a single monocular camera through camera self-calibration and inverse perspective mapping. The proposed scheme operates in real-time and detects over 90% of fall incidents in complex and dynamically changing visual conditions, while it presents very low false positive rate. Besides the contribution to humans fall problem, significant measures of a 3D scene can be calculated that can reveal much more information which might be useful in different kind of applications.

Acknowledgements. The research leading to these results has been supported by European Union funds and national funds from Greece and Cyprus under the project "POSEIDON: Development of an Intelligent System for Coast Monitoring using Camera Arrays and Sensor Networks" in the context of the inter-regional programme INTERREG (Greece-Cyprus cooperation) - contract agreement K1 3 1017/6/2011.

References

1. Le, T., Pan, R.: Accelerometer-based sensor network for fall detection. In: Biomedical Circuits and Systems Conference, BioCAS, pp. 265–268. IEEE (November 2009)
2. Nyan, M., Tay, F.E., Murugasu, E.: A wearable system for pre-impact fall detection. Journal of Biomechanics 41(16), 3475–3481 (2008)
3. Bianchi, F., Redmond, S., Narayanan, M., Cerutti, S., Lovell, N.: Barometric pressure and triaxial Accelerometry-Based falls event detection. IEEE Trans. on Neural Systems and Rehabilitation Engineering 18(6), 619–627 (2010)
4. Zigel, Y., Litvak, D., Gannot, I.: A method for automatic fall detection of elderly people using floor vibrations and sound - proof of concept on human mimicking doll falls. IEEE Trans. on Biomedical Eng. 56(12), 2858–2867 (2009)
5. Debard, G., Karsmakers, P., Deschodt, M., Vlaeyen, E., Van den Bergh, J., Dejaeger, E., Milisen, K., Goedemé, T., Tuytelaars, T., Vanrumste, B.: Camera based fall detection using multiple features validated with real life video (July 2011)
6. Rougier, C., Meunier, J., St-Arnaud, A., Rousseau, J.: Robust video surveillance for fall detection based on human shape deformation. IEEE Trans. on CSVT (5), 611–622 (2011)
7. Fu, Z., Culurciello, E., Lichtsteiner, P., Delbruck, T.: Fall detection using an address-event temporal contrast vision sensor. In: ISCAS 2008, pp. 424–427 (2008)
8. Foroughi, H., Rezvanian, A., Paziraee, A.: Robust fall detection using human shape and multi-class support vector machine. In: ICVGIP 2008, pp. 413–420 (2008)
9. Diraco, G., Leone, A., Siciliano, P.: An active vision system for fall detection and posture recognition in elderly healthcare. In: DATE 2010, pp. 1536–1541 (2010)
10. Thome, N., Miguet, S., Ambellouis, S.: A Real-Time, multiview fall detection system: A LHMM-Based approach. IEEE Trans. on CSVT 18(11), 1522–1532 (2008)
11. Doulamis, N.: Iterative motion estimation constrained by time and shape for detecting person's falls. In: ACM 3rd Inter. Conference on Pervasive Technologies Related to Assistive Environments, Samos, Greece
12. Qian, H., Mao, Y., Xiang, W., Wang, Z.: Home environment fall detection system based on a cascaded multi-SVM classifier. In: ICARCV 2008, pp. 1567–1572 (2008)
13. Grammatikopoulos, L., Karras, G., Petsa, E.: An automatic approach for camera calibration from vanishing points. ISPRS 62(1), 64–76 (2007)
14. Bevilacqua, A., Gherardi, A., Carozza, L.: Automatic perspective camera calibration based on an incomplete set of chessboard markers. In: Sixth ICVGIP, Bhubaneswar, India, pp. 126–133 (2008)