20

Video Surveillance for Fall Detection

Caroline Rougier[1], Alain St-Arnaud[2], Jacqueline Rousseau[3]
and Jean Meunier[4]
[1,4] *Department of Computer Science and Operations Research, University of Montreal*
[2,3] *Research Center of the Geriatric Institute, University of Montreal*
Canada

1. Introduction

1.1 Context

Developed countries have to face the growing population of seniors. In Canada for example, while one Canadian out of eight was older than 65 years old in 2001, this proportion will be one out of five in 2026 (PHAC, 2002), due in particular to the "baby boomers" post-world war II and the increase of life expectancy. Several studies (Chappell et al., 2004; Senate, 2009) have shown that helping elderly people staying at home is interesting from a human perspective, but also from a financial perspective. Hence the interest to develop new healthcare systems to ensure the safety of elderly people at home.

Falls are one of the major risk for seniors living alone at home, often causing severe injuries. The risk is amplified if the person cannot call for help. Usually, wearable fall devices are used to detect falls. For example, an elderly person can call for help using a push button (DirectAlert, 2010), but it is useless if the person is immobilized or unconscious after the fall. Automatic wearable devices are more interesting as no human intervention is required. Some are based on accelerometers (Kangas et al., 2008; Karantonis et al., 2006) which detect the magnitude and the direction of the acceleration. Others are based on gyroscopes (Bourke & Lyons, 2008) which measure the body orientation. A combination of an accelerometer and a gyroscope was used by (Nyan et al., 2008) to detect falls at an earlier stage. The major drawback of these technologies is that these sensors are often embarrassing to wear, and require batteries which need to be replaced or recharged regularly for adequate functioning. Floor vibration-based fall detector (Alwan et al., 2006) can also be used to detect falls but depends on the floor dynamics. This idea has been successfully improved by (Zigel et al., 2009) by adding a sound sensor. They obtained high detection rates, but they admitted that low-impact real human falls may not be detected. Video surveillance offers a new and promising solution for fall detection, as no body-worn devices are needed. For this purpose, a (possibly miniaturized) camera network is placed in the elderly apartment to automatically detect a fall to prevent an emergency center or the family.

1.2 Fall detection problem

1.2.1 General fall detection problem

The main fall detection problem is to recognize a fall among all the daily life activities, especially sitting down and crouching down activities which have similar characteristics to

falls (especially a large vertical velocity). A fall event can be decomposed in four phases (Noury et al., 2008) as shown in Fig. 1:

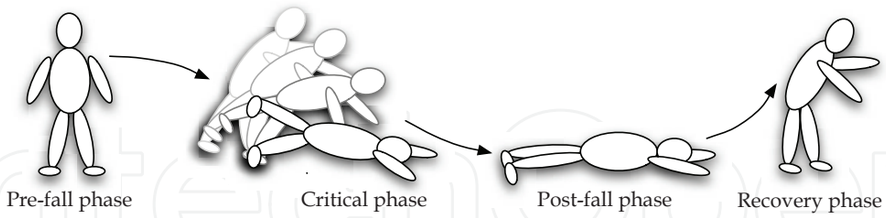


Fig. 1. The different phases of a fall event.

The pre-fall phase corresponds to daily life motions, with occasionally sudden movements directed towards the ground like sitting down or crouching down. These activities should not generate alarm with a fall detection system.

The critical phase, corresponding to the fall, is extremely short. This phase can be detected by the movement of the body toward the ground or by the impact shock with the floor.

The post-fall phase is generally characterized by a person motionless on the ground just after the fall. It can be detected by a lying position or by an absence of significative motion.

A recovery phase can eventually occur if the person is able to stand up alone or with the help of another person.

1.2.2 Specific video surveillance problems

Video surveillance systems need to be robust to image processing difficulties. One of them comes from the camera choice. With inexpensive cameras, the video sequences will contain a *high video compression* (MPEG4 for example) which can generate artifacts in the image. Sometimes, a *variable illumination* can be observed, which must be taken into account during the background updating process. The lighting can also be a source of problems with the appearance of *reflections* in the scene (colors brighter than usual) or *shadows* from the moving person (colors darker than usual). The problem with *reflections* and *shadows* is their detection erroneous as moving objects with a basic segmentation method. *Occlusions* are also a well-known source of errors, mainly due to furniture (chairs, sofa, etc) or entry/exit from the field of view. *Carried objects*, like bags or clothes, can also generate occlusions. Moving objects of no interest (e.g. chair moved) can cause "phantoms" in the image and must be finally integrated in the background image somehow. The silhouette of the person can also be disturbed by the action of *putting on/taking off a coat*. *Clothes with different textures and colors* need to be tested to evaluate their influence on the algorithms, as well as realistic *cluttered and textured backgrounds*.

Robust fall detection systems using video surveillance should not generate alarms because of image processing problems. Some precautions can be taken to limit these sources of problems. Beyond the choice of the camera, the placement of the cameras is important. They need to be placed highly in the room to limit occluding objects and to have a larger field of view. The use of infrared lights can also be considered for lighting problems or for use at night. For our experiments, we have acquired in our laboratory a realistic video data set (Auvinet et al., 2010) of simulated falls and normal daily activities with a multi-camera system. It is composed of inexpensive cameras with a wide angle to cover all the room. This video data set contains all

types of problems described previously, and has been made publicly available for the scientific community to test their fall detection algorithms.

1.3 User perception and receptivity of video surveillance systems

We have conducted a research project (Londei et al., 2009), funded by the Social Sciences and Humanities Research Council of Canada, to explore the perception and the receptivity of the potential users of the Intelligent Videomonitoring System. The study specifically focuses on two objectives:

- to explore their receptivity towards the system (cameras, computer at home) and
- to explore their perception related to the data transmitted (eg. images) and the transmission modes (eg. cell phone).

The study uses a mixed-methods design (Creswell & Clark, 2007). Participants (potential users) include: professionals from the health care and social system (n=31) (nurses, occupational therapists, physiotherapists, social workers and managers), elderly living at home who have fallen during the last year (n=30) and caregivers (n=18). Focus group technique (Krueger, 1994) and structured interviews (Mayer & Ouellet, 1991) were used for data collection. Data analyses were performed with (SPSS, 2007) and (QSR, 2002) softwares. The results of the three main questions are presented here:

1. *What do you think about the Intelligent Videomonitoring System?*
   Fifteen caregivers (83,3%) are in favor of this system as well as 26 seniors (86,7%). Advantages of the system are:
   a) security and quickness of intervention for the seniors,
   b) a relief from stress, for caregivers, related to their fear that the elder falls and stays a long time without assistance while hurt, and
   c) for the professionals, images videotaped a few seconds before the fall occurrence would be a valuable source of information to document fall events in a way to improve security and interventions.
2. *Would you actually use a system such as the Intelligent Videomonitoring System?*
   a) Most of the caregivers (n=15, 83,3%) would like to use the system.
   b) For the elderly, results show that a little less than fifty percent would use the system. The explanation of these results is that elders mention that they don't want it because they don't need it at this time. When they will be "old enough", they would certainly agree to have one in order to stay at home as long as they could.
   c) For the professionals and the managers, this system allows new opportunities for home care: 1) to improve security for elderly living at home focusing on the quickness of the emergency intervention and 2) to document the fall events in a way to better understand the origins of falls and to improve interventions.
3. *What is your choice of images to be transmitted?*
   Figure 2 shows the images presented to the participants. The original image (a) is preferred by all participants: 25 elders (92,6%), 14 caregivers (82,4%) and 5 groups of professionals. In accordance with the professionals, the silhouette images (g-h-i) seem to be more appropriate for videotaping in the bathroom but the original image (a) remains the first choice for elders and caregivers.

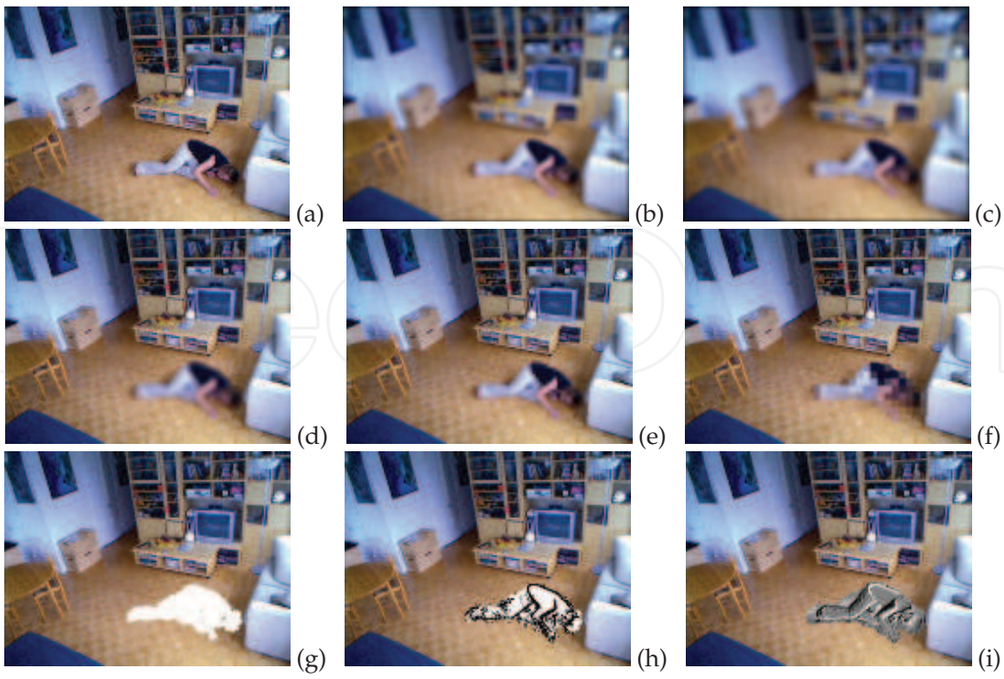


Fig. 2. Images presented to the participants: original image (a) and eight processed images (images (b)-(i) with blurring, pixelization or silhouette extraction)

To summarize: the results show receptivity from the potential users (for example: safety and quick intervention) but some concerns about safety of the transmission system (eg. Images). Intelligent Videomonitoring System is a very promising technology to support the elderly living at home respecting their privacy.

2. Related works on fall detection using video surveillance

The reader can find a good study on fall detection techniques using wearable devices or video surveillance in a recent article by (Noury et al., 2007). In this section, an overview of fall detection methods using video surveillance is proposed.

2.1 Using monocular systems

A commonly used method to detect falls consists of analyzing the person bounding box in the image (Anderson et al., 2006; Tao et al., 2005; Töreyin et al., 2005). This simple method works well with a camera placed sideways, but can fail because of occluding objects. In a more realistic way, other researchers (Lee & Mihailidis, 2005; Nait-Charif & McKenna, 2004) have placed the camera higher in the room for a larger field of view and to avoid occluding objects. The person silhouette and the 2D image velocity were analyzed by (Lee & Mihailidis, 2005) to detect falls with special thresholds for usual inactivity zones like the bed or the sofa (manually initialized). An ellipse representing the person was tracked with a particle filter by (Nait-Charif & McKenna, 2004) to obtain the trajectory used to detect abnormal inactivities outside usual inactivity zones (automatically learned). The vertical velocity is an interesting way to detect falls, either with the 2D vertical image velocity (Sixsmith & Johnson, 2004) or the 3D vertical velocity (Wu, 2000). In this chapter, some new monocular methods will be shown based on human shape change (see Sections 4 and 5) or on 3D head trajectory (see Section 6).

2.2 Using multi-camera systems

A calibrated multi-camera system is useful to reconstruct a three-dimensional representation of the human shape as done by (Anderson et al., 2009) in the voxel space from foreground silhouettes. Their fall detection step was performed by analyzing the states of the voxelized person with a fuzzy hierarchy. For different heights relative to the ground, the homographic transformations of the foreground silhouettes were fused by (Auvinet et al., 2008) in a plane parallel to the ground to reconstruct the 3D human blob. An analysis of the volume distribution along the vertical axis is performed to detect abnormal events like a person lying on the ground after a fall. An alarm is triggered when the major part of this distribution is concentrated near the floor during a predefined period of time. Without reconstructing the 3D human blob, a Layered Hidden Markov Model (LHMM) was used by (Thome et al., 2008) to distinguish falls from walking activities. Their method was based on motion characteristics extracted from a metric image rectification in each view. With two uncalibrated cameras, a Principal Component Analysis (PCA) was performed by (Hazelhoff et al., 2008) on the human silhouette to obtain the direction of the principal component and the variance ratio used for fall detection. A head tracking module was used to improve their recognition results.

3. Our fall detection system

Concretely, a camera network would be placed in the apartment of the person in order to automatically detect a fall. Figure 3 shows an overview of our fall detection system. The images acquired from the video cameras are processed by the local workstation to automatically detect a fall. When a fall is detected, a message could be sent to an emergency center or to the family through a secure Internet connection.

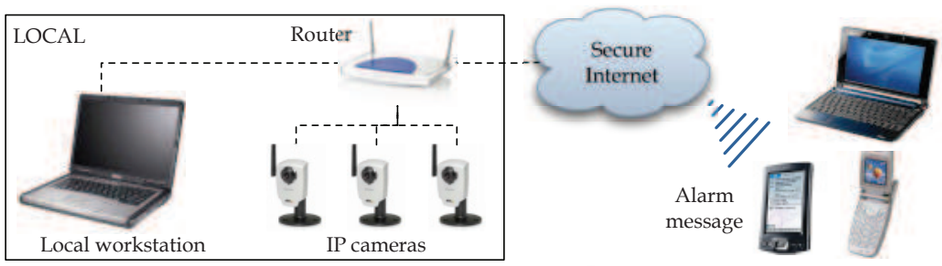


Fig. 3. Our fall detection system.

To limit the cost of our system, inexpensive cameras (IP cameras or webcams) are used and their number is limited to only one per room for cost effectiveness and for simplicity. Indeed, a multi-camera system is more difficult to implement than a monocular one, as reliable 3D information can only be computed if the system is well synchronized and calibrated.
The next sections will describe some of our solutions for fall detection using monocular systems. First, 2D information for fall detection can be used by detecting a fall as a large motion along with changes in the human shape (described in Section 4) or by analyzing the deformation of the human silhouette during and after the fall (described in Section 5). However, some 3D information can be very useful for fall detection as it becomes possible to recover the localization of the person relative to the ground. Usually a multi-camera system is required to have 3D information, but we will show in Section 6 that it is possible to compute the 3D head trajectory of the person from a monocular system. Then, a fall can be detected when the 3D vertical head velocity is too high or when the head is too close to the ground.

Notice that all our algorithms are implemented in C++ using the OpenCV library (Bradski & Kaehler, 2008) and can run in quasi-real-time.

4. 2D information for fall detection: human shape and Motion history image

A fall is characterized by a *large motion* combined with a *change in the human shape*. The idea in this work was to detect and analyze these two characteristics (Rougier et al., 2007).

4.1 Human shape change

The moving person is first extracted from the image with a background subtraction method (Kim et al., 2005) taking into account the problem of shadows, highlights and high image compression. Using moments (Jain, 1989; Pratt, 2001), the person is then approximated by an ellipse defined by its center $(\bar{x}, \bar{y})$, its orientation θ and the length a and b of its major and minor semi-axes. The approximated ellipse gives us information about the shape and orientation of the person in the image. Some examples of background subtraction results and ellipse approximation are shown in Fig. 5 and 6.
Two features are computed for a $1s$ duration to analyze the human shape change:

The orientation standard deviation σ_θ of the ellipse If a person falls perpendicularly to the camera optical axis, then the orientation will change significantly and σ_θ will be high. If the person just walks, σ_θ will be low.

The a/b ratio standard deviation $\sigma_{a/b}$ of the ellipse If a person falls parallelly to the camera optical axis, then the ratio will change and $\sigma_{a/b}$ will be high. If the person just walks, $\sigma_{a/b}$ will be low.

4.2 Motion history image

A serious fall generally occurs with a large movement which can also be quantified with the Motion History Image (Bobick & Davis, 2001). The Motion History Image (MHI) is an image representing the recent motion in the scene, and is based on a binary sequence of motion regions $D(x, y, t)$ from the original image sequence $I(x, y, t)$ using an image-differencing method. Then, each pixel of the Motion History Image H_τ is a function of the temporal history of motion at that point, occurring during a fixed duration τ (with $1 \leq \tau \leq N$ for a sequence of length N frames):

$$H_\tau(x, y, t) = \begin{cases} \tau & if\ D(x, y, t) = 1 \\ \max\left(0, H_\tau(x, y, t-1) - 1\right) & otherwise. \end{cases} \quad (1)$$

The more recent moving pixels are seen brighter in the MHI image. Then, to quantify the motion of the person, we compute a coefficient C_{motion} based on the motion history (accumulation of motion during $500ms$) within the blob representing the person using:

$$C_{motion} = \frac{\sum_{Pixel(x,y) \in blob} H_\tau(x, y, t)}{\#\ pixels \in blob} \quad \text{with} \begin{cases} blob \text{ the person silhouette pixels} \\ H_\tau \text{ the Motion History Image} \end{cases} \quad (2)$$

Only the largest blob is considered here. This coefficient is then scaled to a percentage of motion between 0% (no motion) and 100% (full motion). Some examples of MHI images and corresponding coefficients C_{motion} are shown in Fig. 5 and 6.

4.3 Fall detection

Our complete fall detection algorithm, shown in Fig. 4, is composed of three steps:

1. **Motion quantification**
   A large suspicious motion is detected when the coefficient C_{motion} is higher than 65%. However, a walking person moving perpendicularly to the camera optical axis can also generate a large movement in the MHI image. Thus, we need to analyze further this abnormal motion to discriminate a fall from a normal movement.
2. **Human shape analysis**
   A large motion is considered as a possible fall if σ_θ is higher than 15 *degrees* or if $\sigma_{a/b}$ is higher than 0.9 (sufficient to be insensitive to little ellipse variations due to image segmentation problems or variation in human gait).
3. **Lack of motion after a fall**
   The last step is used to check if the person is immobile on the ground just a few seconds after the fall (during 5 seconds). An unmoving ellipse must respect all these criteria:
   • $C_{motion} < 5\%$
   • $\sigma_{\bar{x}} < 2$ pixels and $\sigma_{\bar{y}} < 2$ pixels, with $\sigma_{\bar{x}}$ and $\sigma_{\bar{y}}$ the standard deviations of the centroid position.
   • $\sigma_a < 2$ pixels, $\sigma_b < 2$ pixels and $\sigma_\theta < 15$ degrees, with σ_a, σ_b and σ_θ the standard deviations of the ellipse parameters.

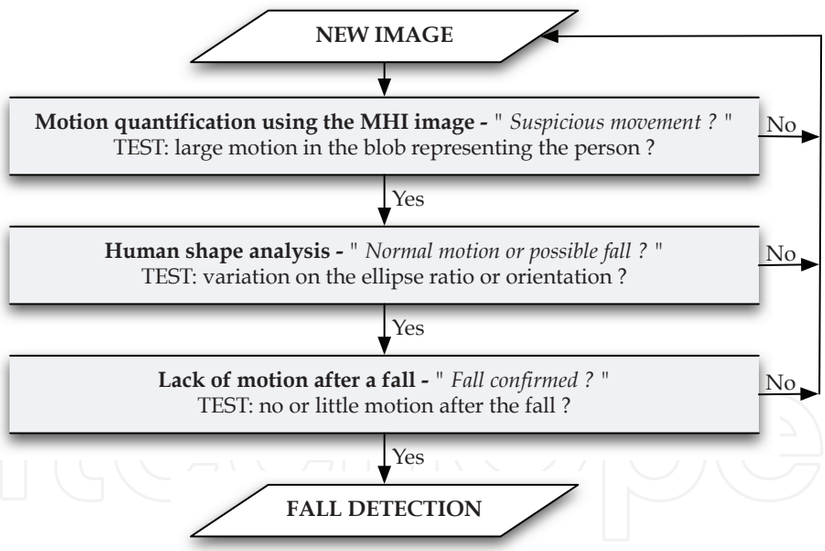


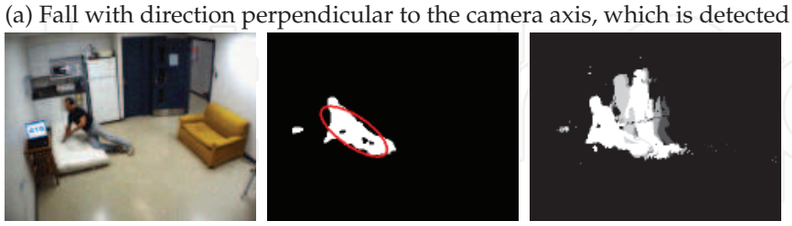
Fig. 4. Our fall detection algorithm based on the Motion History Image and human shape.

4.4 Experimental results

For a low-cost system, our video sequences were acquired using a USB webcam with a wide angle of more than 70 degrees to cover all the room (model Live! Ultra from Creative Technology Ltd). Our system works with a single uncalibrated camera (image size 320x240 pixels) and runs in real-time (computational time of less than 80 *ms* which is adequate for our application as 10 fps is sufficient to detect a fall).

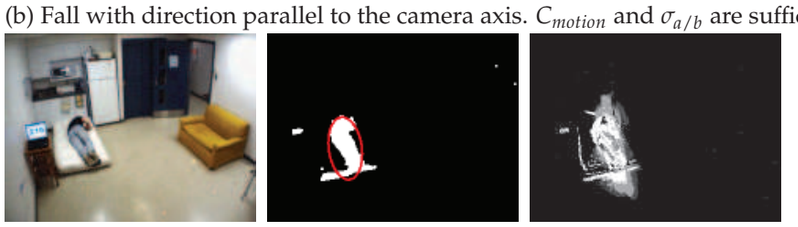
Some examples of falls are shown in Fig. 5 and normal activities in Fig. 6. The human silhouette, extracted from the background, is approximated by an ellipse shown in red. This figure shows also the MHI image obtained and the coefficient values used for fall detection. When a fall occurs, a large motion appears (high C_{motion}) with a significant change in orientation and/or scale (high σ_θ and/or $\sigma_{a/b}$).

(a) Fall with direction perpendicular to the camera axis, which is detected as C_{motion} and σ_θ are high.



$C_{motion} = \mathbf{93.3}\%$
$\sigma_\theta = \mathbf{15.98}$
$\sigma_{a/b} = 0.74$

(b) Fall with direction parallel to the camera axis. C_{motion} and $\sigma_{a/b}$ are sufficiently high to detect the fall.



$C_{motion} = \mathbf{74.3}\%$
$\sigma_\theta = 3.26$
$\sigma_{a/b} = \mathbf{0.97}$

For our experiments, our data set was composed of realistic video sequences representing 24 daily normal activities (walking, sitting down, standing up, crouching down) and 17 simulated falls (forward falls, backward falls, falls when inappropriately sitting down, loss of balance). We obtained a good fall detection rate with a sensitivity of 88% and an acceptable false detection rate with a specificity of 87.5%, in spite of the bad video quality and the fluctuant frame rate of the webcam. We have demonstrated that the combination of motion and change in the human shape gives crucial information on human activities. Some thresholds were experimentally defined in this work, but could be learned from training data. An automatic method based on the human shape deformation is proposed in the next section.

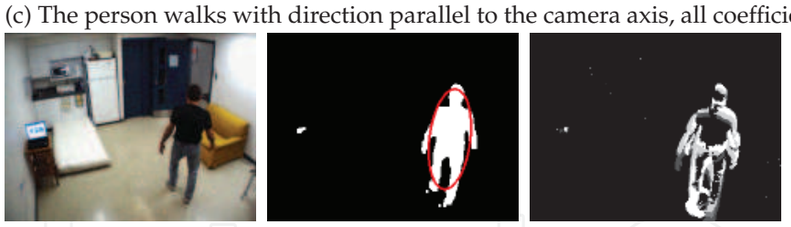
5. 2D information for fall detection: human shape deformation

As seen previously, the human shape is useful for fall detection. In this section, we describe our method to quantify the human shape *deformation* and automatically detect falls (Rougier et al., 2008; 2010b). The idea is that the human shape changes drastically and rapidly during a fall, while during usual activities, this deformation is more progressive and (relatively) slow. In this section, the human shape deformation is quantified to discriminate real falls from normal activities. First, some edge points are extracted from the human silhouette by combining a foreground segmentation with a Canny edge detection in the image. Then, two consecutive silhouettes can be matched using Shape Context to quantify the human shape deformation. Finally, a GMM classifier based on shape analysis is used to detect falls.

5.1 Silhouette edge point matching using shape context

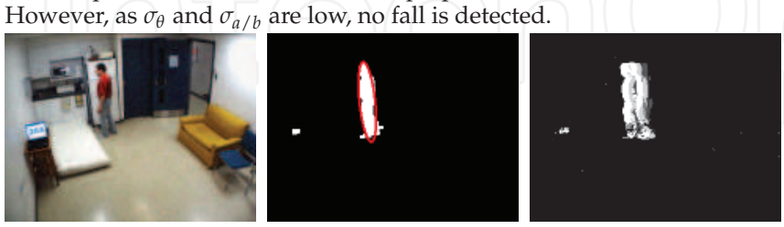
The shape descriptor "Shape Context" (Belongie et al., 2002) is used to match two consecutive sets of edge points. As Shape Context is sensitive to background edges, we improve the method by only considering moving silhouette edge points. They are extracted by combining

(c) The person walks with direction parallel to the camera axis, all coefficients are low.



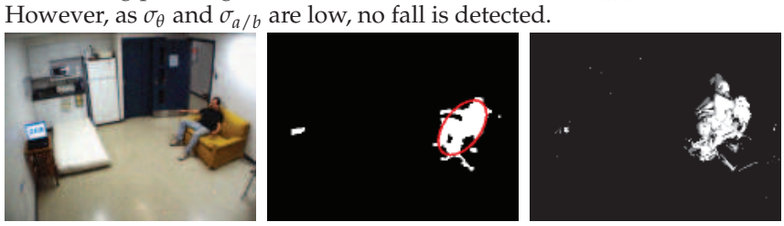
$C_{motion} = 39.6\%$
$\sigma_\theta = 1.64$
$\sigma_{a/b} = 0.26$

(d) The person walks with direction perpendicular to the camera axis, which generates a large C_{motion}. However, as σ_θ and $\sigma_{a/b}$ are low, no fall is detected.



$C_{motion} = \mathbf{74.6}\%$
$\sigma_\theta = 0.69$
$\sigma_{a/b} = 0.32$

(e) A sitting person generates a moderate coefficient C_{motion} which can be sufficient for a suspicious fall. However, as σ_θ and $\sigma_{a/b}$ are low, no fall is detected.



$C_{motion} = \mathbf{72.3}\%$
$\sigma_\theta = 10.4$
$\sigma_{a/b} = 0.8$

(f) The person, doing housekeeping, only moves the arm which do not increase the different coefficients.

$C_{motion} = 9.3\%$
$\sigma_\theta = 4.02$
$\sigma_{a/b} = 0.02$

Fig. 6. Examples of normal activities.

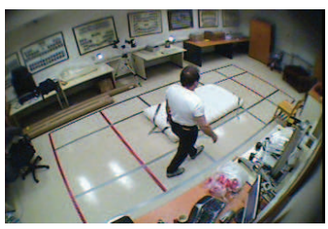
the foreground silhouette, obtained from a background subtraction method (Kim et al., 2005), with an edge image of the scene, obtained from a Canny edge detector (Canny, 1986), to provide additional shape information. For real-time purpose, N landmarks, regularly-spaced, are selected for each silhouette ($N = 250$ for our experiment).
For each point p_i of the first shape, the best corresponding point q_j of the second shape needs to be find. A log-polar histogram h_i is used to encode local information about each point relative to its neighbours. h_i is centered on each point p_i and contains the relative coordinates of the remaining $n - 1$ points:

$$h_i(k) = \#\{q \neq p_i : (q - p_i) \in bin(k)\}, \qquad h_i \text{ contains 5 bins for } \log r \text{ and 12 bins for } \theta \quad (3)$$

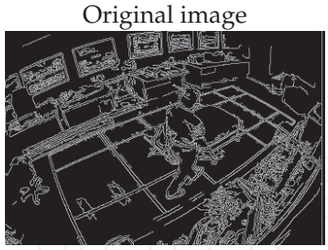
Similar points on the two shapes can be found using the matching cost computed with the χ^2 statistic. This matching cost C_{ij} is computed for each pair of points (p_i, q_j):

$$C_{ij} = C(p_i, q_j) = \frac{1}{2} \sum_{k=1}^{K} \frac{\left[h_i(k) - h_j(k)\right]^2}{h_i(k) + h_j(k)} \tag{4}$$

where $h_i(k)$ and $h_j(k)$ denote the K-bin histograms respectively for p_i and q_j.
Using the resulting cost matrix, the best corresponding points are obtained by minimizing the total matching cost $H(\pi) = \sum_i C(p_i, q_{\pi(i)})$ given a permutation $\pi(i)$. The Hungarian algorithm (Kuhn, 1955) for bipartite matching is used by the authors of (Belongie et al., 2002) to find corresponding points, but this algorithm is time consuming and some bad matching points can appear in spite of the inclusion of dummy points. As we want to keep only reliable points for the shape deformation quantification, we find those that have their cost minimal for the row and the column of the matrix ($\min_i C_{ij} = \min_j C_{ij}$). To discard some bad landmarks which may still remain, the set of matching points is also cleaned based on the motion of the person, by computing the mean motion vector $\bar{v}$ and the standard deviation σ_v from the set of matching points. Only the vectors within 1.28 standard deviation from the mean, which corresponds to 80% of the motion vectors, are kept. The *mean matching cost* $\bar{C}$ is then obtained by averaging all the best matching points costs. An example of Shape Context matching is shown in Fig. 7. While the foreground silhouette is not clean enough to be used for shape analysis, due to segmentation problems, the moving edge points are perfect to match the two consecutive silhouettes.


Original image


Foreground silhouette


Canny edge detection


Moving edge points

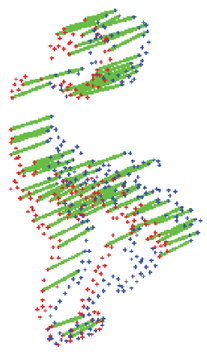

Selected edge points from two consecutive silhouettes and the corresponding matched points obtained with Shape Context

Fig. 7. Example of silhouette matching with Shape Context

5.2 Shape analysis

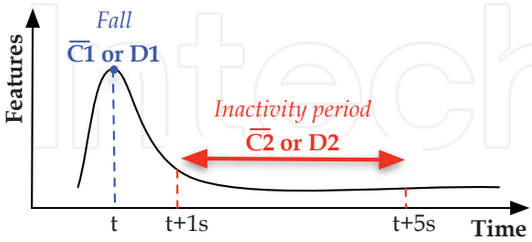
When a fall occurs, the human shape will drastically change during the fall ($\bar{C}_1$ or D_1) and finally will remain motionless just after ($\bar{C}_2$ or D_2) as shown in Fig. 8. These features are analyzed in two ways:

With the mean matching cost i.e. features $(\bar{C}_1, \bar{C}_2)$

$\bar{C}_1$ should be high during the fall as the human shape change drastically in a short period of time, while just after, $\bar{C}_2$ should be low as the person remains unmoving on the ground.

With the full Procrustes distance i.e. features (D_1, D_2)

The Procrustes shape analysis (Dryden & Mardia, 1998) is used to quantify the shape deformation, which consists in comparing the shapes once translational, rotational and scaling components are removed to normalize them. The full Procrustes distance should increase in case of a fall (feature D_1), and should be low just after the fall (feature D_2).



Feature $\bar{C}_1$ or D_1 represents the fall at time t.

Feature $\bar{C}_2$ or D_2 represents the lack of movement after the fall. This feature was computed between $t + 1s$ and $t + 5s$, this time interval was determined experimentally.

Fig. 8. The features $(\bar{C}_1, \bar{C}_2)$ and (D_1, D_2).

5.3 Fall detection using GMM

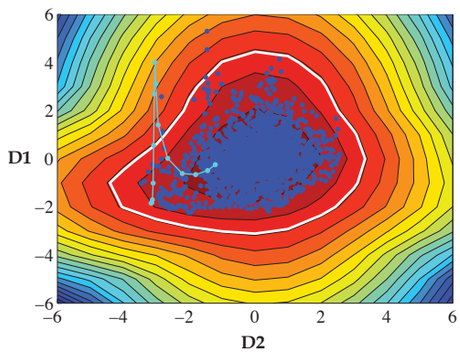
The fall detection problem consists in detecting an abnormal event from a training data set of normal activities, which is known as novelty detection methods (Hodge & Austin, 2004). For our experiment, our normal activities are modeled by a GMM (Gaussian Mixture Model) which is defined by a weighted sum of Gaussian distributions (Nabney, 2001). The GMM parameters are determined using the EM (Expectation-Maximisation) algorithm by maximizing the data likelihood.

Specifically in our case, the parameters of the GMM are estimated from a training data set of daily normal activities (walking, sitting down, crouching down, housekeeping, etc) with the GMM features $(\bar{C}_1$ *or* $D_1, \bar{C}_2$ *or* $D_2)$ described previously. For training and testing, a leave-one-out cross-validation is used. The data set is divided into N video sequences which contain some falls and/or normal activities (including lures). For testing, one sequence is removed from the data set, and the training is done using the $N - 1$ remaining sequences (where falls are deleted because the training is only done with normal activities). The removed sequence is then classified with the resulting GMM. This test is repeated N times by removing each sequence in turn. The sensitivity and the specificity of the system give an idea of the classifier performance. Considering the number of falls correctly detected (*True Positives, TP*) and not detected (*False Negatives, FN*), and the number of normal activities (including lures) detected as a fall (*False Positives, FP*) and not detected (*True Negatives, TN*), the sensitivity is equal to $Se = TP/(TP + FN)$ and the specificity $Sp = TN/(TN + FP)$. An efficient fall detection system will have a high sensitivity (a majority of falls are detected) and a high specificity (normal activities and lures are not detected as falls).

5.4 Experimental results

Our method was tested on our video data set of simulated falls and normal daily activities (Auvinet et al., 2010) taken from 4 different camera points of view. The acquisition frame rate was 30 fps and the image size was 720x480 pixels. The shape matching is implemented in C++ using the OpenCV library (Bradski & Kaehler, 2008) and the fall detection step is done with Matlab using the Netlab toolbox (Nabney, 2001) to perform the GMM classification. The computational time of the shape matching step is about $200ms$ on an Intel Core 2 Duo processor (2.4 GHz), which is adequate for our application as a frame rate of 5 fps is sufficient

to detect a fall. A 3-component GMM was used in our experiment as we have shown (Rougier et al., 2010b) that it was the best compromise between a low classification error rate, a good repeatability of the results and a reasonable computation time. Figure 9 shows a log-likelihood example obtained with a 3-component GMM for the full Procrustes distance features, and a fall event in light blue superimposed on the graphic. The input features are normalized to unit standard deviations and zero means.



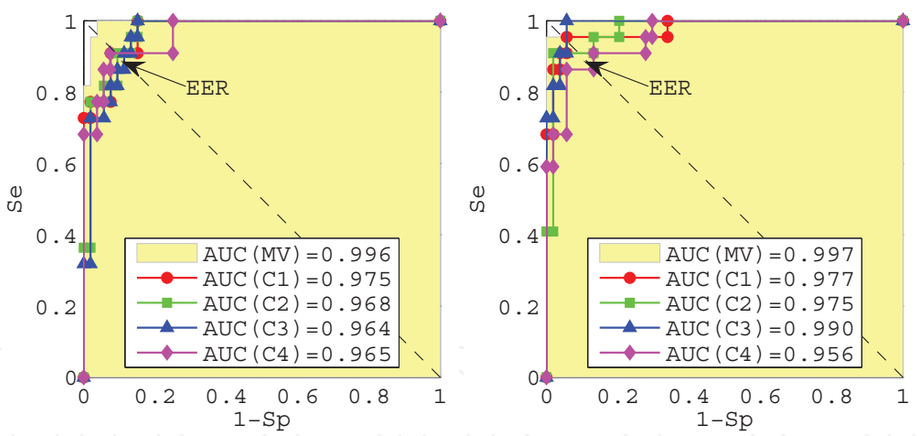
The dark blue points represent the normalized training data set (normal activities).

The white boundary represents the boundary for the chosen log-likelihood threshold.

The light blue points corresponds to a sequence where a fall occurs. The fall is detected when the points are outside the boundary.

Fig. 9. Example of log-likelihood obtained with a 3-component GMM and a fall event.

A ROC analysis was performed for each camera independently and for a majority vote (fall detected if at least 3 of 4 cameras returned a fall detection event). Figure 10 shows the curves obtained for the full Procrustes distance (a) and mean matching cost features (b).



(a) For the full Procrustes distance (b) For the mean matching cost

Fig. 10. ROC curves (log-likelihood threshold ranging from -50 to -1) obtained for each camera independently (C1, C2,C3, C4) and for a majority vote (MV, at least 3 of 4 cameras).

Table 1 shows our recognition results for the *full Procrustes distance* and the *mean matching cost* regarding several evaluation tests:

1. *Using the best matching points*
   Our results are quite good for each camera independently and increase with a majority vote. The similar ROC curves prove that our method is view-independent for the two features. The *full Procrustes distance* and the *mean matching cost* gave similar results with, respectively, an Equal Error Rate of 3.8% and 4.6% with a majority vote.

2. *Using the Hungarian matching*
   The results obtained with the Hungarian matching are not statistically different from those obtained with our methodology. However, Hungarian matching is more time consuming, requires to choose the percentage of dummy points (a parameter that affect considerably the quality of the results) and can leave bad matching points.
3. *Using normal inactivity zones*
   A solution to increase the recognition results could be to define normal inactivity zones (Lee & Mihailidis, 2005) like the bed or the sofa, where the detection thresholds should be less sensitive. Normal inactivity zones were defined manually in our video sequences, and when the person centroid was localized inside one of these zones, the detection threshold was fixed at 1.5 times the normal threshold. As shown in Table 1, the use of normal inactivity zones can really increase the recognition results. These inactivity zones could be automatically learned before installing the system.

Camera	Features	Best matching* points	Hungarian† matching	Inactivity‡ zones
Camera 1	(D_1, D_2)	9.1% (0.978)	13.2% (0.963)	5.7% (0.983)
Camera 2		9.4% (0.968)	9.4% (0.965)	9.1% (0.979)
Camera 3		11.3% (0.964)	7.6% (0.988)	7.6% (0.971)
Camera 4		9.1% (0.966)	13.6% (0.930)	9.1% (0.983)
Majority vote		3.8% (0.996)	9.1% (0.907)	0% (1)
Camera 1	$(\bar{C}_1, \bar{C}_2)$	5.7% (0.977)	11.3% (0.953)	4.6% (0.984)
Camera 2		9.1% (0.975)	9.1% (0.979)	0% (1)
Camera 3		5.7% (0.990)	9.4% (0.979)	5.7% (0.988)
Camera 4		13.2% (0.956)	13.6% (0.935)	9.4% (0.972)
Majority vote		4.6% (0.997)	0% (1)	1.9% (0.999)

* *Our matching method considering only the best matching points.*
† *The Hungarian algorithm (Kuhn, 1955) for bipartite matching with 20% of dummy points.*
‡ *Results obtained when normal inactivity zones are added for classification (best matching points).*

Table 1. EER and AUC values obtained for the full Procrustes distance (D_1, D_2) and the mean matching $(\bar{C}_1, \bar{C}_2)$ features.

In conclusion, the human shape deformation is a useful tool for fall detection, as the full Procrustes distance and the mean matching cost are really discriminant features for classification. By using only reliable landmarks, our silhouette matching using Shape Context is robust to occlusions and other segmentation difficulties (the full Procrustes distance or the mean matching can be sensitive to bad matching points). Our GMM classification results are quite good with only one uncalibrated camera, and the performance can increase using a majority vote with a multi-camera system. Detection errors generally occur when the person sits down too brutally which generates a high shape deformation detected as a fall, or with a slow fall which do not generate a sufficiently high shape deformation to be detected. With such cases, it becomes difficult to chose the best detection threshold. A solution is the use of known inactivity zones which increases the results as shown in this work.

6. 3D information for fall detection

The head trajectory can be very useful for activity recognition and video surveillance applications. A new method is shown here to compute the 3D head trajectory of a person

in a room with only one calibrated camera (Rougier et al., 2006; 2010a). The head, represented by a 3D ellipsoid, is tracked with a hierarchical particle filter based on color histograms and shape information. The resulting 3D trajectory is then used to detect falls.

6.1 Related works in 3D head tracking

The head has been widely used to track a person as it is usually visible in the scene and its elliptical shape is simple. The head can be track by a 2D ellipse in the image plane, for example, using gradient and/or color information with a local search (Birchfield, 1998) or with a particle filter (Charif & McKenna, 2006; K. Nummiaro & Gool, 2003). However, a 3D head trajectory gives more information about the localization and the movement of a person in a room. The easy way to recover some 3D information is to use several cameras. For example, the 3D head trajectory has been extracted using stereo cameras (Kawanaka et al., 2006; Mori & Malik, 2002) or multi-camera systems (Kobayashi et al., 2006; Usabiaga et al., 2007; Wu & Aghajan, 2008). However, tracking the head to recover a 3D trajectory in real-time with only one camera is a real challenge. One attempt by (Hild, 2004) was to compute the top head 3D trajectory of a walking person. However, his assumptions are that the person is standing and that the camera optical axis is parallel to the (horizontal) ground plane, which is not practical in video surveillance applications. Indeed, the camera must be placed higher in the room for a larger field of view and to avoid occluding objects, and the person is not always standing or facing the camera. In our previous work (Rougier et al., 2006) with a single calibrated camera, the head was tracked with a 2D ellipse which was used to compute the 3D head localization by knowing the 3D model of the head. The resulting 3D trajectory for a standing person was well estimated, but some errors occurred with a falling person (need to deal with oriented head). An improvement is shown here using an oriented 3D ellipsoid to represent the head which is tracked with a particle filter through the video sequence.

6.2 Head model projection

The head, represented by a 3D ellipsoid, is projected in the image plane as an ellipse (Stenger et al., 2001). The 3D head model will be tracked in the world coordinate system attached to the XY ground plane as shown in Fig. 11. The projection of the 3D head model in the image plane is possible by knowing the camera characteristics (intrinsic parameters) and the pose of the XY ground plane relative to the camera (extrinsic parameters).

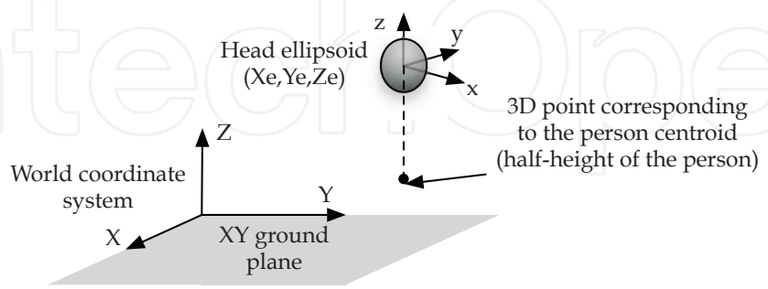


Fig. 11. The 3D head ellipsoid model.

- **Camera parameters**

  The intrinsic parameters were computed using a chessboard calibration pattern and the camera calibration toolbox for Matlab (Bouguet, 2008). The focal length (f_x, f_y) and the

optical center (u_0, v_0) in pixels define the camera's intrinsic matrix K. Notice that image distortion coefficients (radial and tangential distortions) are also computed to correct the images for distortion before processing. From a set of ground points in the real world and the corresponding image points, the plane-image homography is computed to obtain the extrinsic parameters (Zhang, 2000). The extrinsic matrix M_{ext} is defined by R and T which are respectively a 3D rotation matrix and a 3D translation vector.

$$K = \begin{pmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \qquad M_{ext} = \begin{pmatrix} R & T \\ 0\,0\,0 & 1 \end{pmatrix} \tag{5}$$

- **Ellipsoid projection**
  An ellipsoid is described by a positive definite matrix Q_C in the camera coordinate system, such that $[x, y, z, 1]^T Q_C [x, y, z, 1] = 0$, with (x, y, z) a point belonging to the ellipsoid. The ellipsoid Q_C is then projected in the image plane, using the projection matrix P, as a conic C (Hartley & Zisserman, 2004; Stenger et al., 2001):

$$C = Q_{C_{44}} Q_{C_{1:3,1:3}} - Q_{C_{1:3,4}} Q_{C_{1:3,4}}^T \tag{6}$$

  From the conic, the ellipse is described by $[u, v, 1]^T C [u, v, 1] = 0$ for a point (u, v) in the image plane.

- **From the head coordinate system to the ellipse in the image plane**
  Our 3D head ellipsoid model expressed in the head coordinate system, has the form:

$$Q_H = \begin{pmatrix} \frac{1}{B^2} & 0 & 0 & 0 \\ 0 & \frac{1}{B^2} & 0 & 0 \\ 0 & 0 & \frac{1}{A^2} & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} \qquad \begin{array}{c} \text{with the semi-major } A \text{ and} \\ \text{the semi-minor } B \text{ ellipsoid head axes} \end{array} \tag{7}$$

  The projection matrix $P = KM_{ext}M_{Head/World}$, which represents the transformation from the head ellipsoid coordinate system to the image plane, is used to project the head ellipsoid in the camera coordinate system such that $Q_C = P^{-1T} Q_H P^{-1}$. The translation and rotation of the head in the world coordinate system, which corresponds to the matrix $M_{Head/World}$, will be defined by the head tracking (see Section 6.3). Finally, the parameters of the ellipse representing the head in the image plane are obtained from the conic defined in eq. 6.

6.3 3D head tracking with particle filter

Tracking with particle filters has been widely used, for example, to track the head with an ellipse (K. Nummiaro & Gool, 2003; Rougier et al., 2006) or a parametric spline curve (Isard & Blake, 1998) using color information or edge contours. Their particularity is that they allow abrupt trajectory variations and can deal with small occlusions.
Particle filters are used to estimate the probability distribution $p(S_t|Z_t)$ of the state vector S_t of the tracked object given Z_t, representing all the observations. This probability can be approximated from a set $S_t = \{s_t^n, n = 1, \ldots, N\}$ of N weighted samples (also called particles) at time t. A particle filter is composed of three steps:

1. **Selection**
   N new samples are selected from the previous sample set by favoring the best particles to create a new sample set S'_t.
2. **Prediction**
   A stochastic dynamical model is used to propagate the new samples $s_t^n = A_l s_t'^n + B_l w_t^n$, where w_t^n is a vector of standard normal random variables, and A_l and B_l are, respectively, the deterministic and stochastic components of the dynamical model.
3. **Measurement**
   The new weights $\pi_t^n = p(z_t|s_t^n)$ are computed and normalized so that $\sum_n \pi^n = 1$.

The final step corresponds to the mean state estimation of the system at time t using the N final weighted samples i.e. $E[S_t] = \sum_{n=1}^{N} \pi_t^n s_t^n$
Our implementation of the particle filter is similar to the annealed particle filter (Deutscher et al., 2000) in a hierarchical scheme with several layers. Each layer is composed of the three main particle filter steps, and at the end of the layer, the stochastic component is reduced for the next layer: $B_{l+1} = B_l/2$ (see Section 6.5). Our ellipsoid particles are represented by the state vector:

$$s_t^n = [X_e, Y_e, Z_e, \theta_{Xe}, \theta_{Ye}]_t^n \tag{8}$$

where (X_e, Y_e, Z_e) is the 3D head ellipsoid centroid expressed in the world coordinate system (translation component of the matrix $M_{Head/World}$), and $(\theta_{X_e}, \theta_{Y_e})$ are respectively the rotation around the X and the Y axes (rotation component of the matrix $M_{Head/World}$) [1]. No motion is added in our dynamical model as the previous velocity between two successive centroids is already added to the particles to predict the next 3D ellipsoid localization before propagating the particles (i.e. A_l is an identity matrix).

6.4 Particles weights

The particle weights are based on foreground, color and body coefficients:

- **Foreground coefficient** C_F

  Role The 3D pose precision is obtained with the foreground coefficient when the ellipsoid is well matched to the head contour

  Definition The foreground silhouette of the person is extracted with a background subtraction method which consists in comparing the current image with an updated background image (Kim et al., 2005). The foreground coefficient is computed by searching for silhouette contour points along N_e line segments normal to the ellipse, distributed uniformly along the ellipse and centered on its contour. An example of foreground coefficient is shown in Fig. 12.

$$C_F = \frac{1}{N_e} \sum_{n=1}^{N_e} \frac{D_e(n) - d_e(n)}{D_e(n)}, \quad C_F \in [0 \ldots 1] \tag{9}$$

  where d_e is the distance from the ellipse point to the detected silhouette point and D_e, the half length of the normal segment, is used to normalize the distances.

[1] Notice that two angles (instead of three) are sufficient to define the position and orientation of the ellipsoid since its minor axes have both the same length in our model.

For an ellipse defined by the semi-axis parameters a and b, each segment begins from the inner ellipse (axes $a/2$ and $b/2$) to the outer ellipse (axes $a + a/2$ and $b + b/2$).

Fig. 12. Foreground segmentation and foreground coefficient computation examples.

- **Color coefficient** C_C

  Role The color coefficient is used to prevent the ellipsoid from hanging on something else inside the silhouette when large movement occurs.

  Definition The color coefficient is based on a normalized 3D color histogram of the head (K. Nummiaro & Gool, 2003). The histogram H is computed in the RGB color space inside a rectangular zone included in the head ellipse and composed of $N_b = 8 \times 8 \times 8$ bins. The updated color head model and the target model are compared by calculating the normalized histogram intersection:

$$C_C = \sum_{i=1}^{N_b} \min \left(H(i), H_{ref}(i) \right), \quad C_C \in [0 \ldots 1] \tag{10}$$

- **Body coefficient** C_B

  Role The body coefficient is used to link the head to the body through the body center, to avoid unrealistic 3D ellipsoid rotation.

  Definition The distance between the projection of the 3D point corresponding to the centroid of the person (see Fig. 11) and the 2D silhouette centroid (distance d_b compared to the half-major axis of the bounding box D_b) should be small. This coefficient is only used when the bounding box is valid (and thus not used in case of occlusion for example).

$$C_B = \frac{D_b - d_b}{D_b}, \quad C_B \in [0 \ldots 1] \tag{11}$$

The final ellipsoid coefficient is an amplified combination of these three coefficients to give larger weights to the best particles ($\sigma = 0.15$):

$$C_{final} = \frac{1}{\sqrt{2\pi}\sigma} \exp^{(C_F C_C C_B)/2\sigma^2} \tag{12}$$

As the mean state of the particle filter is a weighted combination of all particles, the weights amplification is important to obtain a more precise 3D localization.

6.5 Initialization and tracking

The ellipsoid size is calibrated from a manually initialized 2D ellipse representing the head. With this ellipse and by knowing the body height and the ellipse aspect ratio (The ratio of a human head ellipse is fixed at 1.2 (Birchfield, 1998)), the ellipsoid proportion can be computed.

Our system is automatically initialized with a head detection module which consists in testing several 2D ellipses from the top head point of the foreground silhouette. The one which has the biggest foreground coefficient C_F is kept, and if $C_F > 0.7$, the ellipse is supposed sufficiently reliable to begin the tracking with the particle filter.
An initial 3D head centroid localization can be computed from this 2D detected ellipse, by knowing the ellipsoid proportion and the camera calibration parameters using the iterative algorithm POSIT (Dementhon & Davis, 1995). This algorithm returns the relative position of the head in the camera coordinate system $P_{Head/Cam}$, which can be transformed in the world coordinate system attached to the XY ground plane using $P_{Head/World} = M^{-1}_{World/Cam} P_{Head/Cam}$ with the matrix $M_{World/Cam}$ representing the known position of the world coordinate system in the camera coordinate system. The head localization $P_{Head/World}$ is used to initialize the tracking and is then refined with the particle filter.
For a reliable 3D head localization, the head projection in the image need to be well adjusted to the head contour. With a conventional particle filter, a lot of particles are needed for precision which severely affects the computational performance and is incompatible with real-time operation. With several layers, a better precision can be reached in a shortest time, a good compromise between performance and computational time can be obtained with 250 particles and 4 layers. The stochastic component $B_l = \left[B_{X_e}, B_{Y_e}, B_{Z_e}, B_{\theta_{X_e}}, B_{\theta_{Y_e}}\right]$ for the model propagation is different for each layer, sufficiently large for the first layer and decreasing for the next layers, such as $B_{l+1} = B_l/2$ with l the current layer and $l+1$ the next layer. As the person is supposed to be standing up at the beginning, Z_e is approximately known and, θ_{X_e} and θ_{Y_e} are close to zero. Thus, our initial values are fixed to $B_l = [0.5\ 0.5\ 0.3\ 0\ 0]$ which corresponds to a large diffusion for the X and Y components ($\pm 50cm$) on the horizontal plane and a moderate one ($\pm 30cm$) for the Z component. For the next images, the current velocity is used to reinitialize B_l such that the particles spread towards the 3D trajectory direction (minimum of $0.1m$ or $0.1rad$ for B_l). Recall that A_l is an identity matrix (see Section 6.3). Figure 13 shows the usefulness of the hierarchical particle filter for a large motion. By considering only the first layer, the ellipsoid is badly estimated, while with the four layers, the ellipsoid is finally well adjusted.

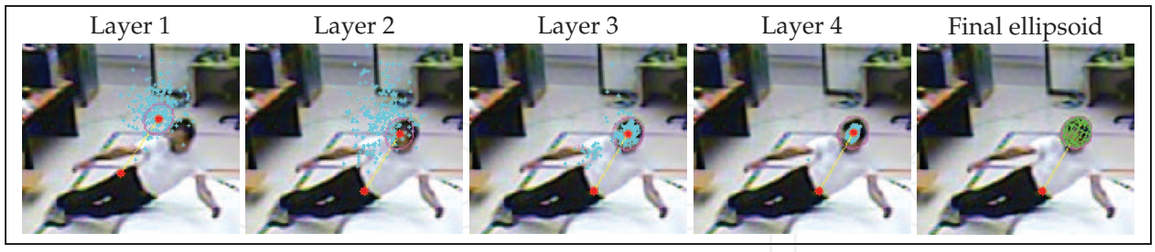


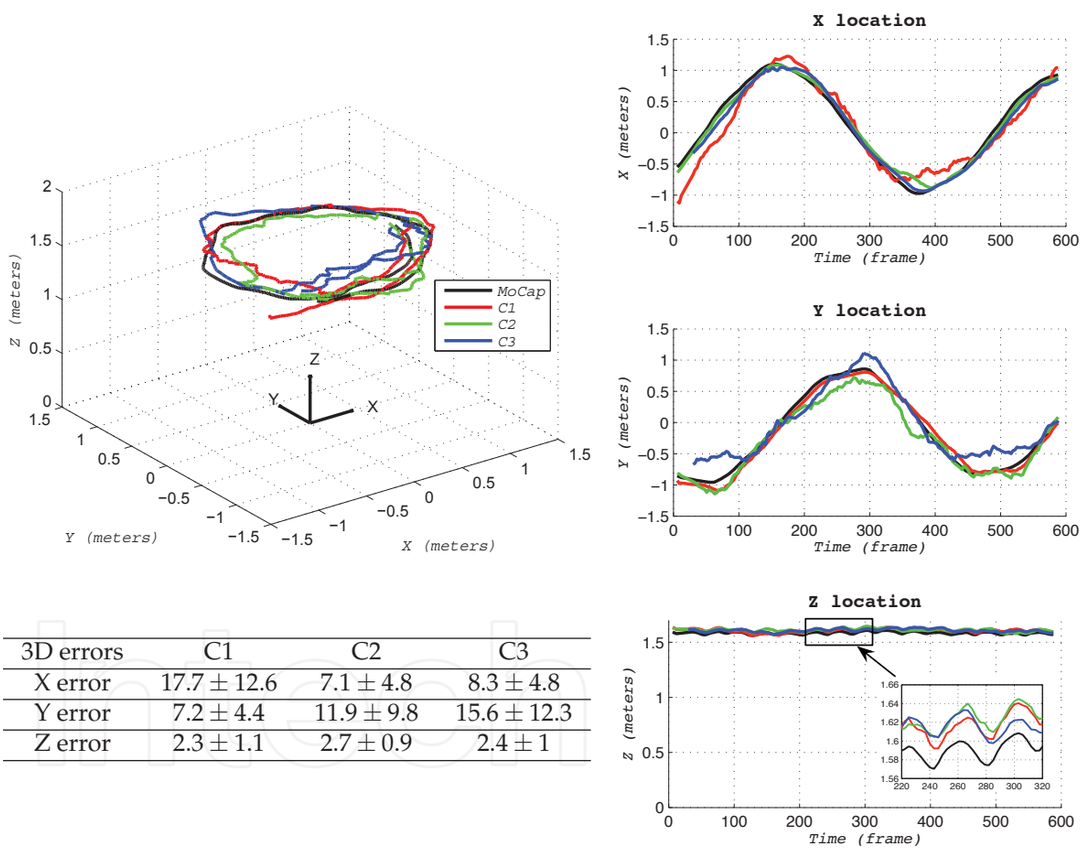
Fig. 13. Example of a large motion during a fall. The images show the particles and the mean state ellipsoid for each layer, and the resulting ellipsoid.

6.6 Experimental results

Our 3D head tracker is implemented in C++ using the OpenCV library (Bradski & Kaehler, 2008) and can run in quasi-real time (130ms/frame on an Intel Core 2 Duo processor (2.4 GHz), non optimized code and image size of 640x480).

6.6.1 3D localization precision using HumanEva data set

The 3D localization precision is evaluated with the HumanEva-I data set (Sigal & Black, 2006) which contains synchronized multi-view video sequences and corresponding MoCap data (ground truth 3D localizations). The results were obtained for each camera independently (color cameras C_1, C_2 and C_3) using the video sequences of 3 subjects (S1, S2 and S3). The motion sequences "walking" and "jogging" were used to evaluate our 3D trajectories at 30Hz, 20Hz and 10Hz. The resulting 3D head trajectories (top head point) obtained from different view points are similar to the MoCap trajectory as shown in Fig. 14. The small location error for the Z axis and the cyclical movement of the walking person visible on the curve prove that the head height is quite well estimated. The depth error (X and Y location) is a little higher due to the ellipsoid detection (noisy foreground images, artifacts on the silhouette) and depending on the orientation of the person relative to the camera (for simplicity, the frontal/back and lateral views of the head are considered identical in our 3D ellipsoid model).



3D errors	C1	C2	C3
X error	17.7 ± 12.6	7.1 ± 4.8	8.3 ± 4.8
Y error	7.2 ± 4.4	11.9 ± 9.8	15.6 ± 12.3
Z error	2.3 ± 1.1	2.7 ± 0.9	2.4 ± 1

Fig. 14. 3D head trajectories for a walking sequence of subject S1 (20Hz). The table shows the mean 3D errors (in cm) for X, Y and Z location.

The 3D mean errors obtained for each subject and each camera are shown in Table 2. The mean error was about 5% at a 4 to 6 meters distance. As expected, the error tended to be slightly higher when the movement was larger, but the head still continues to be well tracked with the 3D ellipsoid.

Camera, Frame rate	Walking sequences S1	S2	S3	Jogging sequences S1	S2	S3
C1, 30Hz	20.6 ± 13.6	20.5 ± 7.3	21.3 ± 12.9	19.2 ± 9.5	24.1 ± 16.3	25.9 ± 15.3
C2, 30Hz	17.1 ± 13	21.3 ± 7.8	23.6 ± 10.7	20.6 ± 12.4	24.5 ± 10.6	17.4 ± 9.7
C3, 30Hz	17.3 ± 11.6	21.4 ± 8.4	25.9 ± 17	21.5 ± 13.6	23.7 ± 11.6	28.8 ± 17.2
C1, 20Hz	20 ± 12.4	21.2 ± 7.4	19.7 ± 11.5	16.6 ± 10.3	25.4 ± 17.4	25.5 ± 16.7
C2, 20Hz	15.1 ± 9.6	22.8 ± 8.5	22.8 ± 11.1	22.8 ± 10.1	26.3 ± 12	16.9 ± 10.3
C3, 20Hz	19 ± 11.5	20.6 ± 8.4	28.8 ± 19.4	15.3 ± 9.8	23 ± 11.5	30 ± 19
C1, 10Hz	24 ± 12.8	22.8 ± 8.2	21 ± 13	21 ± 11.4	25.9 ± 16.2	23.2 ± 16.1
C2, 10Hz	18.3 ± 12.6	22.5 ± 9.9	18.3 ± 14	22.1 ± 13.8	29.2 ± 13.7	22.8 ± 16.8
C3, 10Hz	22.6 ± 14	19.5 ± 8.5	28.8 ± 17.7	22 ± 10.5	24.1 ± 14.2	33.7 ± 20.7

Table 2. Mean 3D errors (in cm) obtained from walking and jogging sequences for different subjects (S1, S2, S3), several view points (C1, C2, C3) and several frame rates.

6.6.2 3D head trajectory for fall detection

A biomechanical study with wearable markers (Wu, 2000) showed that falls can be distinguished from normal activities using 3D velocities. In Fig. 13, we have shown that our 3D head tracker was efficient with an oriented person and large motion. We propose here to use the 3D head trajectory, obtained without markers, for fall detection. Two fall detection methods are explored:

The vertical velocity V_v of the head centroid is computed as a height difference for a 500 ms duration [2]: $V_v = Z_e(t) - Z_e(t - 500\ ms)$

The head height Z_e, corresponding to the centroid head height relative to the ground, should be small at the end of the fall as the person is supposed to be near the ground.

For our experiment, ten falls (forward falls, backward falls, loss of balance) from our video data set (Auvinet et al., 2010) were used with two cameras. The falls were done in different directions with respect to the two camera points of view, which were separated by 9 meters (deep field of view), placed at the entrance of the space and on the opposite wall. The acquisition frame rate was 30 fps, but 10 fps was sufficient to detect a fall. The image size was 720x480 pixels. Due to the wide angle, the images needed to be corrected for distortion before processing as shown in Fig. 15.

Fig. 15. Examples of images from the two viewpoints before and after distortion correction.

Figure 16 shows the vertical velocity V_v and the head height Z_e obtained for a video sequence of a fall viewed from the two points of view. The 3D head tracking was performed in spite of the deep field of view, and even if the person was not entirely in the image (camera 1 near the entry). Our tracker was automatically initialized when the head was correctly detected. As seen previously with the HumanEva-I data set, the head height was rather precise giving

[2] Duration of the fall critical phase (Noury et al., 2008)

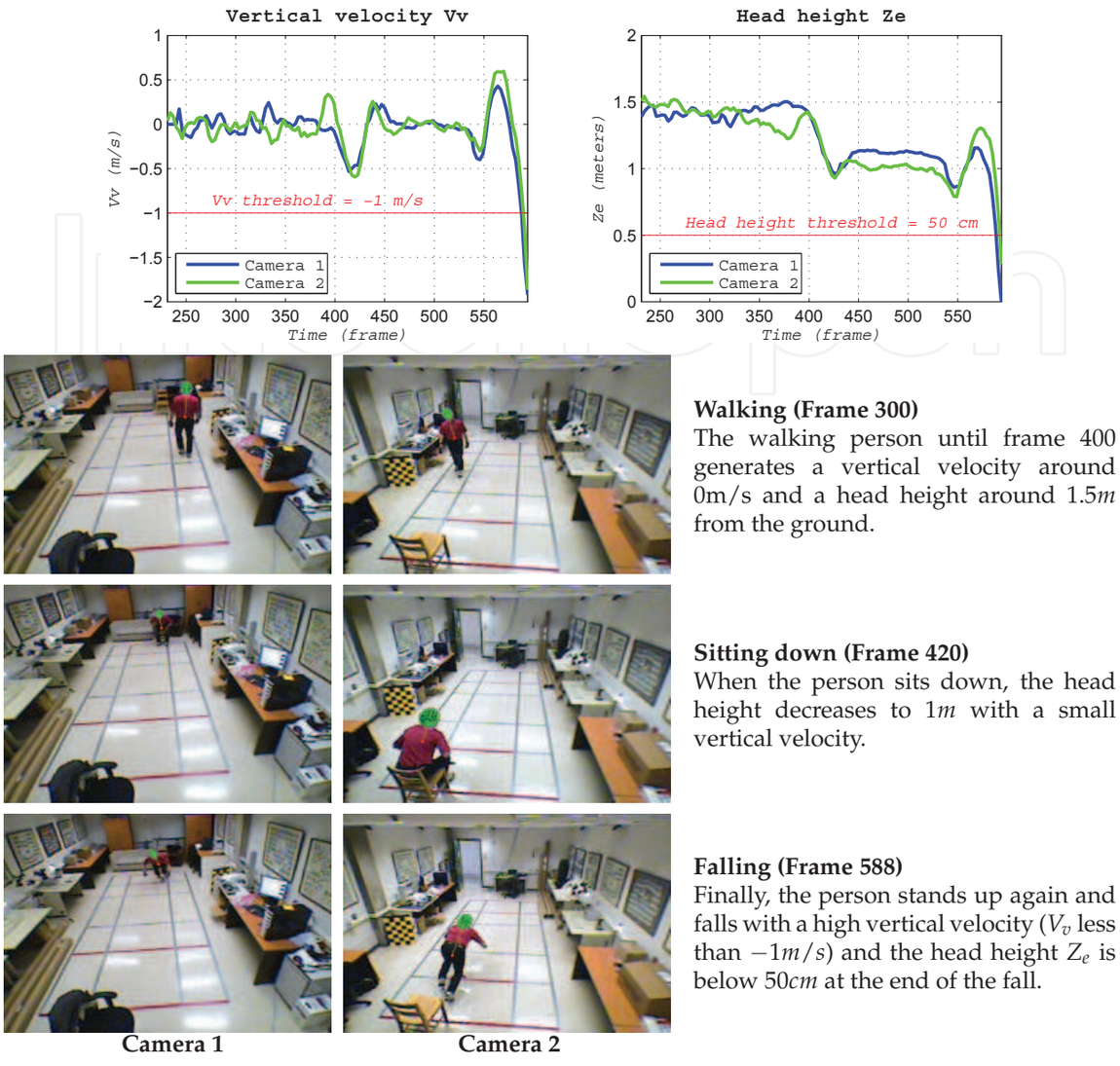


Fig. 16. Vertical velocity and head height obtained for a video sequence of a fall.

similar head heights from the two views although 9 meters separates the two cameras. In spite of the low image quality, the 10 falls from the two views were successfully detected with the vertical velocity V_v (with a threshold at $-1m/s$). A person sitting down abruptly is also shown in Fig. 16 producing a vertical velocity equal to $-0.53m/s$ which was not sufficient to be detected as a fall. A head localization near the ground can be considered as a suspicious event for an old person. Thus, a fall can be detected when the head height Z_e is below 50cm. In this case, only one fall was not detected because of a tracking failure due to a noisy silhouette. However, this fall was detected with vertical velocity ($V_v = -1.17m/s$). Notice in Fig. 16 that the head height was about 1m when the person was seated.

To summarize, a 3D head trajectory can be extracted with only one calibrated camera. Our tracker was able to give similar results for different viewpoints, different frame rates and different subjects as shown with the HumanEva-I data set. These tests showed that the 3D locations were estimated with a mean error of around 25cm (5% at 5 meters) which

is sufficient for most activity recognition based on trajectories. One important point is that our 3D head tracker is automatically initialized with a well-detected 2D head ellipse. The hierarchical particle filters with 4 layers is useful for the head tracking precision in a reasonable computational time. Our method can deal with body occlusions (for example with chairs or occlusion due to entry into the scene), however the head need to be appropriately visible to have a reliable 3D pose, as the 3D localization is inferred from the head foreground detection. For example, our 3D head tracker sometimes fails at the end of a fall towards the camera. Indeed, the head tends to be merged with the body of the person which can give some 3D errors. However, even if the 3D pose is not well estimated, a high vertical velocity generally occurs at the beginning of a fall. Thus, the vertical velocity is a better criterion for fall detection than the location of the head because head height can lead to failure because of occlusion or tracking problem when the head is near the ground.

7. Conclusion and future work

An overview of fall detection techniques using video surveillance has been proposed in this chapter. Several fall detection methods using a single camera have been shown and have shown that monocular video surveillance systems are a good solution for fall detection with high detection rates. A robust method for fall detection is the analysis of the human shape motion and deformation. Even with realistic and difficult video data sets, such system are able to discriminate falls from normal daily activities automatically (Sections 4 and 5). The 3D localization of the person is also a useful tool for fall detection, and we have demonstrated that it is feasible with only one calibrated camera. All these methods are view-independent, automatically initialized and can run in real-time, considering that 5 to 10 fps is sufficient for fall detection.

When developping such systems, we must ensure the privacy of the person, which can be satisfied here, as our systems are entirely automated and access to the images could be forbidden except in case of emergency. For instance, the system will send an alarm signal toward an outside resource (e.g. via a cell phone or Internet) if and only if an abnormal event is detected (e.g. falling). Moreover, recall that this technology do not hamper the movement of the person as no devices are required and no button needs to be pushed.

How to improve the robustness

To reduce the risk of false alarms, a hybrid method combining 2D and 3D information could be considered. Considering only the human shape deformation, slow falls are sometimes more difficult to discriminate from a person sitting down brutally. By using the 3D head velocities or the 3D head localization, these two events can be discriminated. Inversely, when the 3D tracking is not sufficiently reliable (for example when the head is occluded), the human shape deformation could help to detect falls.

Multi-camera systems could also be used to improve the recognition results by combining information from several cameras to take a decision. However, these systems are more expensive and difficult to implement requiring an accurate calibration and synchronization. Some stereo systems entirely calibrated and directly usable e.g. (PointGrey, 2010) could be used to provide a more reliable depth information than a monocular system. Although these systems are still expensive, with the renewed interest in 3D technologies, some 3D digital cameras and webcams are now proposed for general public (Fujifilm, 2010; Minoru, 2010) suggesting that stereo systems will become more affordable in the future. **Next challenges for**

healthcare video surveillance systems

Beyond fall detection, gait analysis could help to identify persons at risk with unstable gait patterns requiring reeducation to reduce the risk of falling. Moreover, a video surveillance system can provide a large amount of information about the person, but also his/her interaction with the environment. A computer vision system could be used to check other daily activities like medication intake (Valin et al., 2006), or meal/sleep time and duration. Information about his/her environment could also be analyzed for fire detection, forgotten oven or running faucet and other home hazards.

Healthcare video surveillance systems are a new and promising solution to improve the quality of life and care for elderly, by preserving their autonomy and generating the safety and comfort needed in their daily lives. This corresponds to the hopes of the elderly themselves, their families, the caregivers and the governments. The positive receptivity for video surveillance systems suggests that this technology has a bright future for healthcare and will advantageously complement other approaches (e.g. fixed or wearable sensors, safer home modifications, etc) by overcoming many of their limitations.

8. Acknowledgements

This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

9. References

Alwan, M., Rajendran, P., Kell, S., Mack, D., Dalal, S., Wolfe, M. & Felder, R. (2006). A smart and passive floor-vibration based fall detector for elderly, *2nd Information and Communication Technologies*, Vol. 1, pp. 1003–1007.

Anderson, D., Keller, J., Skubic, M., Chen, X. & He, Z. (2006). Recognizing falls from silhouettes, *International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 6388–6391.

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

Auvinet, E., Reveret, L., St-Arnaud, A., Rousseau, J. & Meunier, J. (2008). Fall detection using multiple cameras, *30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 2554–2557.

Auvinet, E., Rougier, C., Meunier, J., St-Arnaud, A. & Rousseau, J. (2010). Multiple cameras fall data set, *Technical Report 1350*, University of Montreal, Canada.
URL: *http://vision3d.iro.umontreal.ca/fall-dataset*

Belongie, S., Malik, J. & Puzicha, J. (2002). Shape matching and object recognition using shape context, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24(4): 509–522.

Birchfield, S. (1998). Elliptical head tracking using intensity gradients and color histograms, *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pp. 232–237.

Bobick, A. & Davis, J. (2001). The recognition of human movement using temporal templates, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23(3): 257–267.

Bouguet, J.-Y. (2008). Camera calibration toolbox for matlab.
URL: *http://www.vision.caltech.edu/bouguetj/calib_doc*

Bourke, A. & Lyons, G. (2008). A threshold-based fall-detection algorithm using a bi-axial gyroscope sensor, *Medical Engineering & Physics* 30(1): 84–90.

Bradski, G. & Kaehler, A. (2008). *Learning OpenCV: Computer Vision with the OpenCV Library*, O'Reilly.
URL: *http://opencv.willowgarage.com/wiki*

Canny, J. (1986). A computational approach to edge detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 8(6): 679–698.

Chappell, N. L., Dlitt, B. H., Hollander, M. J., Miller, J. A. & McWilliam, C. (2004). Comparative costs of home care and residential care, *The Gerontologist* 44: 389–400.

Charif, H. N. & McKenna, S. J. (2006). Tracking the activity of participants in a meeting, *Machine Vision and Applications* 17(2): 83–93.

Creswell, J. & Clark, V. P. (2007). *Designing and Conducting Mixed Methods Research*, Thousands Oaks, CA: SAGE.

Dementhon, D. & Davis, L. (1995). Model-based object pose in 25 lines of code, *International Journal of Computer Vision* 15(1-2): 123–141.

Deutscher, J., Blake, A. & Reid, I. (2000). Articulated body motion capture by annealed particle filtering, *Proc. IEEE Computer Vision and Pattern Recognition*, Vol. 2, pp. 126–133.

DirectAlert (2010). Wireless emergency response system.
URL: *http://www.directalert.ca/emergency/help-button.php*

Dryden, I. & Mardia, K. (1998). *Statistical Shape Analysis*, John Wiley and Sons, Chichester.

Fujifilm (2010). 3d digital camera finepix real 3d w3.
URL: *http://www.fujifilm.com*

Hartley, R. I. & Zisserman, A. (2004). *Multiple view geometry in computer vision*, 2nd edn, Cambridge University Press.

Hazelhoff, L., Han, J. & de With, P. H. N. (2008). Video-based fall detection in the home using principal component analysis, *Advanced Concepts for Intelligent Vision Systems*, Vol. 1, pp. 298–309.

Hild, M. (2004). Estimation of 3d motion trajectory and velocity from monocular image sequences in the context of human gait recognition, *International Conference on Pattern Recognition (ICPR)*, Vol. 4, pp. 231–235.

Hodge, V. J. & Austin, J. (2004). A survey of outlier detection methodologies, *Artificial Intelligence Review* 22: 85–126.

Isard, M. & Blake, A. (1998). Condensation – conditional density propagation for visual tracking, *International Journal of Computer Vision* 29(1): 5–28.

Jain, A. (1989). *Fundamentals of digital image processing*, 2nd edn, Prentice Hall, Englewood Cliffs, New Jersey.

K. Nummiaro, E. K.-M. & Gool, L. V. (2003). An adaptive color-based particle filter, *Image and Vision Computing* 21(1): 99–110.

Kangas, M., Konttila, A., Lindgren, P., Winblad, I. & Jämsä, T. (2008). Comparison of low-complexity fall detection algorithms for body attached accelerometers, *Gait & Posture* 28(2): 285–291.

Karantonis, D., Narayanan, M., Mathie, M., Lovell, N. & Celler, B. (2006). Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring, *IEEE Transactions on Information Technology in Biomedicine* 10(1): 156–167.

Kawanaka, H., Fujiyoshi, H. & Iwahori, Y. (2006). Human head tracking in three dimensional voxel space, *International Conference on Pattern Recognition (ICPR)*, Vol. 3, pp. 826–829.

Kim, K., Chalidabhongse, T., Harwood, D. & Davis, L. (2005). Real-time foreground-background segmentation using codebook model, *Real-Time Imaging* 11(3): 172–185.

Kobayashi, Y., Sugimura, D., Hirasawa, K., Suzuki, N., Kage, H., Sato, Y. & Sugimoto, A. (2006). 3d head tracking using the particle filter with cascaded classifiers, *Proc. of British Machine Vision Conference (BMVC)*, pp. 37–46.

Krueger, R. (1994). *Focus group: a practical guide for applied research*, 2nd edn, Thousands Oaks, CA: SAGE.

Kuhn, H. W. (1955). The hungarian method for the assignment problem, *Naval Research Logistic Quarterly* 2: 83–97.

Lee, T. & Mihailidis, A. (2005). An intelligent emergency response system: preliminary development and testing of automated fall detection, *Journal of telemedicine and telecare* 11(4): 194–198.

Londei, S. T., Rousseau, J., Ducharme, F., St-Arnaud, A., Meunier, J., Saint-Arnaud, J. & Giroux, F. (2009). An intelligent videomonitoring system for fall detection at home: perceptions of elderly people, *Journal of Telemedicine and Telecare* 15(8): 383–390.

Mayer, R. & Ouellet, F. (1991). *Méthodologie de recherche pour les intervenants sociaux*, Boucherville: Gaëtan Morin.

Minoru (2010). Webcam minoru 3d.
URL: *http://www.minoru3d.com*

Mori, G. & Malik, J. (2002). Estimating human body configurations using shape context matching, *European Conference on Computer Vision LNCS 2352*, Vol. 3, pp. 666–680.

Nabney, I. T. (2001). *NETLAB - Algorithms for Pattern Recognition*, Springer.

Nait-Charif, H. & McKenna, S. (2004). Activity summarisation and fall detection in a supportive home environment, *Proceedings of the 17th International Conference on Pattern Recognition (ICPR)*, Vol. 4, pp. 323–326.

Noury, N., Fleury, A., Rumeau, P., Bourke, A., Laighin, G., Rialle, V. & Lundy, J. (2007). Fall detection - principles and methods, *29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS 2007)*, pp. 1663–1666.

Noury, N., Rumeau, P., Bourke, A., ÓLaighin, G. & Lundy, J. (2008). A proposal for the classification and evaluation of fall detectors, *IRBM* 29(6): 340–349.

Nyan, M., Tay, F. E. & Murugasu, E. (2008). A wearable system for pre-impact fall detection, *Journal of Biomechanics* 41(16): 3475–3481.

PHAC (2002). Canada's aging population, Public Health Agency of Canada, Division of Aging and Seniors.

PointGrey (2010). Stereo vision camera system - bumblebee2.
URL: *http://www.ptgrey.com*

Pratt, W. (2001). *Digital Image Processing*, 3rd edn, John Wiley & Sons, New York.

QSR (2002). QSR international pty. QSR N'Vivo (2nd version for IBM). Melbourne, Australia.

Rougier, C., Meunier, J., St-Arnaud, A. & Rousseau, J. (2006). Monocular 3d head tracking to detect falls of elderly people, *International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 6384–6387.

Rougier, C., Meunier, J., St-Arnaud, A. & Rousseau, J. (2007). Fall detection from human shape and motion history using video surveillance, *IEEE 21st International Conference on Advanced Information Networking and Applications Workshops*, Vol. 2, pp. 875–880.

Rougier, C., Meunier, J., St-Arnaud, A. & Rousseau, J. (2008). Procrustes shape analysis for fall detection, *ECCV 8th International Workshop on Visual Surveillance (VS 2008)*.

Rougier, C., Meunier, J., St-Arnaud, A. & Rousseau, J. (2010a). 3d head tracking using a single calibrated camera, *Image and Vision Computing (Submitted)* .

Rougier, C., Meunier, J., St-Arnaud, A. & Rousseau, J. (2010b). Robust video surveillance for fall detection based on human shape deformation, *IEEE Transactions on Circuits and Systems for Video Technology (Accepted)* .

Senate (2009). Canada's aging population: Seizing the opportunity, *Technical report*, Special Senate Committee on Aging, Senate Canada.

Sigal, L. & Black, M. J. (2006). Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion, *Technical Report CS-06-08*, Brown University, Department of Computer Science, Providence, RI.

Sixsmith, A. & Johnson, N. (2004). A smart sensor to detect the falls of the elderly, *IEEE Pervasive Computing* 3(2): 42–47.

SPSS (2007). SPSS 15.0. Statistical Package for the Social Sciences Inc. Chicago.

Stenger, B., Mendonça, P. & Cipolla, R. (2001). Model-based hand tracking using an unscented kalman filter, *Proc. BMVC*, Vol. 1, pp. 63–72.

Tao, J., Turjo, M., Wong, M.-F., Wang, M. & Tan, Y.-P. (2005). Fall incidents detection for intelligent video surveillance, *Fifth International Conference on Information, Communications and Signal Processing*, pp. 1590–1594.

Thome, N., Miguet, S. & Ambellouis, S. (2008). A real-time, multiview fall detection system: A lhmm-based approach, *IEEE Transactions on Circuits and Systems for Video Technology* 18(11): 1522–1532.

Töreyin, B., Dedeoglu, Y. & Çetin, A. (2005). Hmm based falling person detection using both audio and video, *Proc. IEEE International Workshop on Human-Computer Interaction*, pp. 211–220.

Usabiaga, J., Bebis, G., Erol, A., Nicolescu, M. & Nicolescu, M. (2007). Recognizing simple human actions using 3d head movement, *Computational Intelligence* 23(4): 484–496.

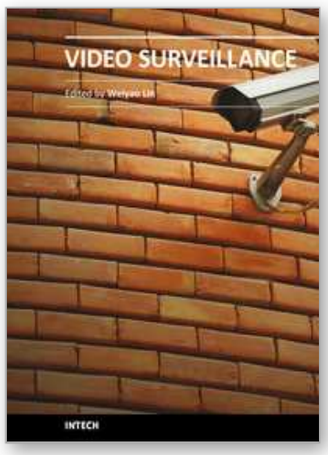
Valin, M., Meunier, J., St-Arnaud, A. & Rousseau, J. (2006). Video surveillance of medication intake, *International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 6396–6399.

Wu, C. & Aghajan, H. (2008). Head pose and trajectory recovery in uncalibrated camera networks - region of interest tracking in smart home applications, *ACM/IEEE International Conference on Distributed Smart Cameras*, pp. 1–7.

Wu, G. (2000). Distinguishing fall activities from normal activities by velocity characteristics, *Journal of Biomechanics* 33(11): 1497–1500.

Zhang, Z. (2000). A flexible new technique for camera calibration, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22(11): 1330–1334.

Zigel, Y., Litvak, D. & Gannot, I. (2009). A method for automatic fall detection of elderly people using floor vibrations and sound - proof of concept on human mimicking doll falls, *IEEE Transactions on Biomedical Engineering* 56(12): 2858–2867.



Video Surveillance
Edited by Prof. Weiyao Lin

ISBN 978-953-307-436-8
Hard cover, 486 pages
Publisher InTech
Published online 03, February, 2011
Published in print edition February, 2011

This book presents the latest achievements and developments in the field of video surveillance. The chapters selected for this book comprise a cross-section of topics that reflect a variety of perspectives and disciplinary backgrounds. Besides the introduction of new achievements in video surveillance, this book also presents some good overviews of the state-of-the-art technologies as well as some interesting advanced topics related to video surveillance. Summing up the wide range of issues presented in the book, it can be addressed to a quite broad audience, including both academic researchers and practitioners in halls of industries interested in scheduling theory and its applications. I believe this book can provide a clear picture of the current research status in the area of video surveillance and can also encourage the development of new achievements in this field.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen