A Real-Time, Multiview Fall Detection System: A LHMM-Based Approach

Nicolas Thome, Serge Miguet, *Member, IEEE*, and Sébastien Ambellouis

***Abstract*—Automatic detection of a falling person in video sequences has interesting applications in video -surveillance and is an important part of future pervasive home monitoring systems. In this paper, we propose a multiview approach to achieve this goal, where motion is modeled using a layered hidden Markov model (LHMM). The posture classification is performed by a fusion unit, merging the decision provided by the independently processing cameras in a fuzzy logic context. In each view, the fall detection is optimized in a given plane by performing a metric image rectification, making it possible to extract simple and robust features, and being convenient for real-time purpose. A theoretical analysis of the chosen descriptor enables us to define the optimal camera placement for detecting people falling in unspecified situations, and we prove that two cameras are sufficient in practice. Regarding event detection, the LHMM offers a principle way for solving the inference problem. Moreover, the hierarchical architecture decouples the motion analysis into different temporal granularity levels, making the algorithm able to detect very sudden changes, and robust to low-level steps errors.**

***Index Terms*—Fall detection, layered hidden Markov model (LHMM), metric rectification, multiview pose classification.**

I. Introduction

With the population growing older and the increasing number of people living alone, supportive home environments able to automatically monitor human activities are flourishing due to their promising ability to help elderly people living alone and to reduce healthcare costs. In particular, fall detection is becoming an emergent field of research, by the increase each year in deaths and injuries entailed by falls. At the moment, existing solutions can be classified into personal embedded sensors, low-level sensors and video sensors. Worn sensors such as fall detectors may produce false alarms. Simple remote sensors produce low-level data that are crude and difficult to interpret. On the other hand, cameras offer a semantic information. Unfortunately, data processing requires advanced computer vision techniques that are prone to errors and computationally expensive. Moreover, cameras are limited to their field of view. Finally, the main issue of using computer vision techniques is related to the acceptability and privacy surrounding it. In this paper we propose a video-based method for monitoring human activities, with a particular interest to the problem of fall detection.

Manuscript received February 17, 2008; revised July 01, 2008. First published September 26, 2008; current version published October 29, 2008. This work was supported by the SAS Foxstream (http://www.foxstream.fr). This paper was recommended by Associate Editor D. Schonfeld..

N. Thome and S. Ambellouis are with the LaboratoireÉlectronique, Ondes et Signaux pour les Transports, 59666 Villeneuve d'Ascq Cédex, France (e-mail: nicolas.thome@inrets.fr;sebastien.ambellouis@inrets.fr).

S. Miguet is with the Laboratoire d'InfoRmatique en Images et Systèmes d'information, Université Lumière Lyon 2.5, 69576 Bron Cedex, France (e-mail: serge.miguet@liris.cnrs.fr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2008.2005606

II. State of the Art

Providing robust solutions for detecting falls requires the solution of two main challenges. First, we have to propose algorithms for posture classification that are robust to large changes in viewpoint, that can efficiently deal with partial occlusions and cover a maximal field of view. Second, the proposed solution must capture and recognize motion features, that are very discriminative in the fall detection context. More generally, we aim at providing solutions compatible with real time purpose in a multiview setting.

A. Posture Classification

Regarding posture classification, existing approaches can be classified depending on the use of a model. 3-D models are by essence view-points independent, and 3-D tracking approaches have been investigated by directly minimizing an image to model measurement (generative approaches) [1], or by learning the features to pose mapping from exemplars (discriminative approaches) [2]. However, 3-D approaches are mostly not able to achieve real-time, and mainly require manual initialization. Many methods modeling the body part assembly with 2-D models have been proposed, for example cardboard models or pictorial structures [3]. Once the body parts are properly labeled, model-based approaches can generally recover posture easily, and are robust to partial occlusions. On the other hand, model-free method try to directly estimate the pose using generic image feature. Haritaoglu *et al.* propose in [4] to classify the pose between a set of predefined ones using silhouette projection histograms. Although the approach is computationally efficient, it is not clear how well the algorithm will generalize with large variations in viewpoint.

B. Event Detection

Event modeling and recognition relates to building a semantic description of human activity. Model-free methods [5] aim at automatically clustering different kinds of events. Model-based methods explicitly describe a given type of movement. The first attempt to perform this task relies on building temporal templates [6]. Shortcomings of this approach are related to viewpoint and time variability dependence as well as sensitivity to noise in the observations. Alternatively, hidden Markov models

(HMMs) [7] have been widely used for tackling simple behaviours such as gestures or gait recognition. Due to the Markovian assumption, HMM are limited to model simple motions for one single human. Thus, other extensions to the basic HMM have been used such as the Coupled Hidden Markov Models [8], and variable length Markov models [9]. Finally, requirements for scalable systems for high level understanding and semantically rich behaviour recognitions led to study frameworks that use the inherent hierarchical structure of motion. In that sense, sophisticated stochastic methods have been used to model the combination between elementary behavioral pattern detected by the previous methods, leading to the highest level of the interpretation module. It has been accomplished by the development of abstract hidden Markov models, hierarchical hidden Markov models and layered hidden mMarkov models [10].

C. Previous Works on Fall Detection

The Simbad project [11] uses infra-red sensors, making the people detection and feature extraction easier. Fall detection is performed by using a neural network classifying a vertical velocity descriptor. Nevertheless, the requirement of fast movements recognition may lead to a sensitivity to noise tending to send false alarms. The Ubisense project [12] proposes to classify human poses by computing the orientation of each detected blob. However, no motion modeling and recognition is proposed for analyzing the pose sequences.

Nait-Charif and McKenna [13] propose a method for automatically extracting motion trajectory and providing human-readable summarization of activity and detection of unusual inactivity. Tracking is performed with an omnidirectional camera by means of a particle filter estimating ellipse parameters describing human posture. Fall is detected as a deviation to usual activity. However, no information about the pose of the person or his motion dynamic is taken into account.

Töreyin *et al.* [14] suggest a method for fall detection by making use of an HMM using both audio and video. For the vision part of the approach, the aspect ratio of the bounding box of the moving region detected with a standard camera is analyzed by the motion model. More precisely, its wavelet transform is used as input feature for the HMM. Using conjointly video and audio cues seems to be well founded. Defining HMM states in the frequency domain is interesting because it makes explicit use of motion features. However, the viewpoint robustness of the bounding box aspect ratio feature for discriminating standing and lying postures is not discussed, and the evaluation is mainly limited to frontal views. It is clear, for example, that the aspect ratio observed in the image corresponding to a standing posture will sensitively vary between a vertically-oriented optical axis and an horizontally-oriented one. The problem remains for the wavelet coefficients, making the motion recognition efficiency only limited to some specific viewpoint configurations.

Recently, Cucchiara *et al.* [15] propose a multiview solution dedicated to fall detection. They make use of histogram projections to classify the silhouette between the standing and lying poses. Interestingly, warping people's silhouette between the different views makes it possible to detect partial occlusions, and to compensate it. A HMM-based approach is proposed for making the pose recognition more robust. However, the motion is only taken into account at very small time scales, to disambiguate the pose estimation, and no explicit modeling of the motion in terms of pose sequence is proposed.

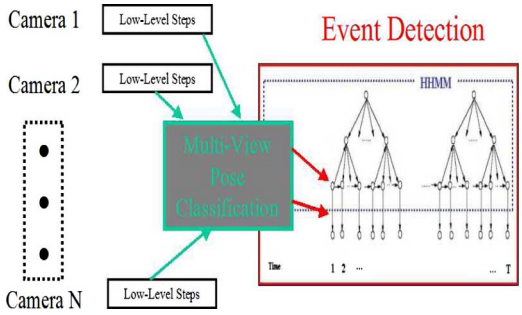


Fig. 1. Overview of the proposed multiview fall detection system.

III. Approach Overview

In the previous Fall Detection approaches (Section II-C), either the solutions concentrate on the posture classification, or they focus on the high-level reasoning. In this paper, we propose the following approach, schemed in Fig. 1, that addresses both problems conjointly. We use a multiple view setting, where the low-level steps are (mainly) performed independently in each view, leading to the extraction of simple image features compatible with real-time achievement. Then, a fusion unit merges the output of each camera to provide a multiview pose classifier efficient in unspecified conditions (viewpoints), as explained in Section IV. The motion analysis is performed by means of a layered hidden Markov model (LHMM), that is well adapted to the fall detection context (Section V).

Our main contributions for providing a robust fall detection system state are as follows.

- First, we derive theoretical properties making it possible to determine the domain validity of the chosen detector dedicated to classifying the 3-D pose of the person in each view. This sets up the camera placement to provide a combined pose classifier with 100% detection rate. Importantly, the fusion unit is performed in a fuzzy logic context, making it possible to output a combined classifier likelihood.
- Second, the hierarchical architecture of the LHMM offers an intuitive way for representing falls, and provides an interesting trade-off between temporal sensitivity and robustness. Moreover, the event detection is formulated by the HMM formalism as an inference problem in a principled way.

IV. Body Pose Analysis

A. People Detection and Tracking

The first step of the system consists in detecting people. This is achieved by using a background subtraction algorithm, using a variant of the Stauffer mixture of Gaussians modeling [16], that is robust to shadows by using a color space invariant in luminance. Then, human classification and tracking is carried out by building a robust appearance model. This feature is used to identify people in difficult situations, such as occlusions, and

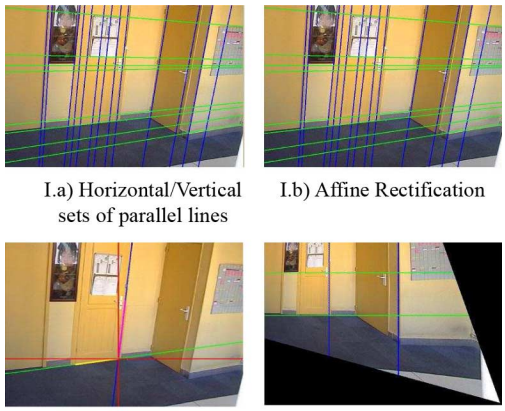


Fig. 2. Metric rectification performing.

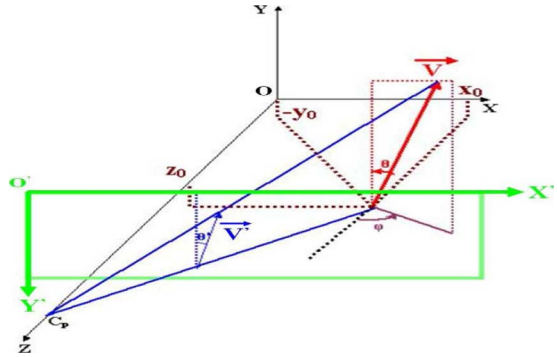


Fig. 3. Three-dimensional deviation between the vertical direction θ and image deviation θ'.

maintain the tracking. For further details about this part of the approach, the reader can refer to [17].

B. Feature Extraction

Regarding people being detected and tracked as explained in Section IV-A, we propose a static analysis of their body pose dedicated to classifying the silhouette between standing and lengthened postures. First, we compute the silhouette best fitting ellipse, using a minimal bounding rectangle (MBR) algorithm. Thus, we compute the angle between the MBR length and the vertical direction, constituting the input feature for our body pose analysis algorithm. The image descriptor is not actually computed in each view independently. Similarly to [15], we match the silhouettes between the different sensors by a re-projection step using the calibration parameters. If a large gap between the registered views is observed, a geometrical reasoning is carried out to infer the more reliable silhouette. Eventually, this makes it possible to detect occlusions or segmentation errors, and to compensate them, making the multiview feature extraction more robust.

1) Applying a Metric Rectification: However, we have to face geometrical issues when directly using the MBR angle for posture classification. Indeed, strong perspective effects might be observed in our context, due to the fact that small focal distances are used in indoor conditions. Thus, the 3-D vertical direction projects into the image plane on a pencil of lines with potential large variations (we observe deviations larger than 10° in our experiments). We apply a metric rectification to overcome that shortcoming. For a general overview of metric rectification of perspective images of planes, the reader is referred to [18]. In our context, we choose the following strategy for performing that step, illustrated in Fig. 2. An affine rectification is applied by determining the vanishing line, computed by identifying two orthogonal vanishing points, as illustrated in Fig. 2(a) and (b). To perform the metric part of the rectification, we enforce two additional constraints to preserve orthogonality and aspect ratio, as illustrated in Fig. 2(c) and (d).

C. Single View Pose Classification

Let us denote π the plane where the previously described rectification has been applied. By choosing π containing the vertical direction, we can certify that each vertical line in the scene is projected in a vertical line in the image. In addition, the rectification makes it possible to refer to a generic configuration, from which we can establish the needed properties to classify the feature between standing and lengthened poses. Thus, once the rectification is applied, everything occurs as if the camera were facing the rectification plane π. Fig. 3 illustrates the problem formulation, where π corresponds to the $(0, X, Y)$ plane. The red vector $\vec{V}$ represents the principal axis of the person in the 3-D world. Its origin V_o is located at $(x_0, -y_0, z_0)$ and corresponds to the feet of the person, and its extremity V_e gives the position of the head. The θ angle corresponds to the 3-D deviation from the vertical axis (OY), and φ is the angle between the (OZ) axis and the projection of $\vec{V}$ on the (OXZ) plane. The image plane intersects the (OZ) axis in O' (at $z = d$), and the center of projection Cp coordinates are $(0, 0, (d + f))^T$, where f is the camera focal length. In the image plane (O', X, Y'), the blue vector $\vec{V}'$ is the image of $\vec{V}$ after projection. The θ' angle corresponds to the deviation from the vertical axis (OY') on the image plane.

As a first step for labeling the silhouette between standing and lengthened poses, We propose to relate θ to θ'. Assuming pinhole model for the camera, we get (see Appendix A)

$$\tan(\theta') = \tan(\theta)\frac{(d + f - z_0)\sin(\varphi) + x_0\cos(\varphi)}{(d + f - z_0) - y_0\tan(\theta)\cos(\varphi)}. \quad (1)$$

1) Feature Classification: Equation (1) constitutes the basis of our posture classification. We can derive the following Property (see Appendix B).

Property 4.1: Provided that

$$\begin{cases} z_0 < S \\ x_0 \in [-L; L] \\ \theta \in [0; \theta_{\max}] \text{ and } \theta_{\max} < \arctan(d + f - S/|y_0|) \end{cases}$$
$$|\tan(\theta')| \leq \frac{\tan(\theta_{\max})(d + f + L)}{d + f - S - |y_0|\tan(\theta_{\max})}. \quad (2)$$

Equation (2) makes it possible to limit the angular error introduced by the image formation process. S and L are positive thresholds. Qualitatively, the Property 4.1 may be reformulated as follows: if someone is in an *approximate upright standing position* ($\theta \in [0; \theta_{\max}]$) and if they are *not too close to the camera* (meaning $z_0 < S$), then they will be seen with a *small deviation with respect to the vertical direction* in the camera image plane.

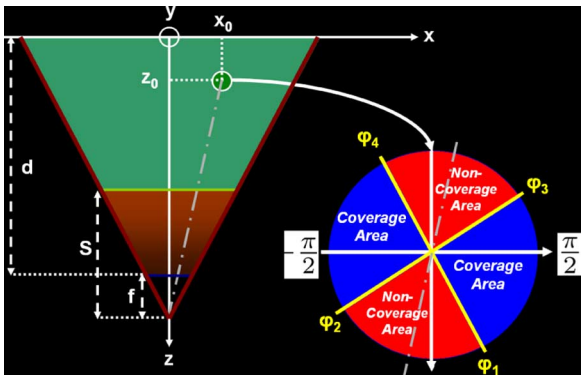


Fig. 4. Validity domain over φ for a lengthened pose in (x_0, y_0, z_0).

2) Nonverticality Detector: Property 4.1 enables us to label the silhouette of the tracked person between standing or lengthened. Let us define θ'_T as follows:

$$|\tan(\theta'_T)| = \frac{\tan(\theta_{\max})(d+f+L)}{d+f-S-y_0\tan(\theta_{\max})}. \tag{3}$$

Taking advantage of Property 4.1 contrapose, we can state that if $|\theta'| > \theta'_T$, then $\theta > \theta_{\max}$. Thus, we can build a ***nonverticality*** detector by thresholding the image feature θ'.

D. Multiple View Posture Classification

Let us denote a positive as a lengthened pose detected with our nonverticality detector, and a negative as a standing detected pose. Provided that the low level parts of the system leading to the computation of the angle are properly carried out, we can certify that the nonstanding poses labeled with the proposed detector correspond to nonstanding in the world, i.e., we can reach a 100% true positive detection rate. There are, however, some nonstanding poses in the world that our detector fails to identify, and we can not guarantee a detection rate of 100% true negatives. For example, if $\varphi = 0$ and $x_0 = 0$ in (1), $\theta' = 0\,\forall\theta$. However, we can derive the following Property (see Appendix C).

Property 4.2:

If $\theta = \pi/2$ and under Property 4.1 assumptions

$$\tan(\theta') > \tan(\theta'_T) \Leftrightarrow \begin{cases} \varphi > \varphi_1 = \arctan\left(\frac{-x_0+\tan(\theta'_T)|y_0|}{d+f-z_0}\right) \\ \text{or} \\ \varphi < \varphi_2 = \arctan\left(\frac{-x_0-\tan(\theta'_T)|y_0|}{d+f-z_0}\right) \end{cases}. \tag{4}$$

Property 4.2 holds for $\varphi \in [-\pi/2; \pi/2]$, but similar conditions exist for $\varphi \in [\pi/2; 3\pi/2]$, with thresholds such as $\varphi_4 = \varphi_1 + \pi$ and $\varphi_3 = \varphi_2 + \pi$.

The Property 4.2 is very interesting for our purpose, because it specifies that the limited recall of our detector is directly related to the falling direction, i.e., to φ. Thus, we can define a range value for φ, in each point (x_0, y_0, z_0), to remove false negatives, i.e., someone lengthened on the world will necessary be detected as lengthened with our image detector. Fig. 4 illustrates the computed validity domain on φ in each point (x_0, y_0, z_0). As we are interested in the φ and as $\theta = \pi/2$, the figure represents a view from above, i.e., in the $y < 0$ direction.

In addition, we prove that a value exists for φ_i, $i \in [1, 4]$ that makes it possible to remove all false negatives whatever the (x_0, z_0) position.

Property 4.3:

If $\theta = \pi/2$, and under Property 4.1 assumptions

$$\forall(x_0, z_0)\ \tan(\theta') > \tan(\theta'_T) \Leftrightarrow \begin{cases} \varphi > \varphi_S = \arctan\left(\frac{L+\tan(\theta'_T)|y_0|}{d+f-S}\right) \\ \text{or} \\ \varphi < -\varphi_S = -\arctan\left(\frac{L+\tan(\theta'_T)|y_0|}{d+f-S}\right) \end{cases}.$$

Property 4.3 (proved in Appendix D) states that it is possible to define an area of the φ space for which we can properly identify all lengthened poses, independently to the position. This result is capitalized on for providing a **multiple view pose detector** that reaches 100% true positives and negatives. Thus, we define ρ_c as the *coverage area*, i.e.,

$$\rho_c = 1 - \frac{2\varphi_S}{\pi}. \tag{5}$$

Then, we propose a simple strategy for providing a 100% true negative detection rate in a multiview context. The minimal number of cameras as well as their placement for properly detecting a person lengthened in at least one camera can be determined in the following way.

- $N_c = \lceil 1/1-\rho_c \rceil$, where $\lceil X \rceil$ corresponds to the upper rounding of the real X.
- A way to place the $N_c - 1$ additional cameras consists in orienting the optical axis of the ith camera ($i \in [1; N_c - 1]$) with an angle $\Psi_i = \pm i\varphi_S/N_c - 1(\textit{modulo}\ \pi)$.

As an example, if $N_c = 2$, placing the two cameras in orthogonal viewing directions makes it possible to remove all false negatives, i.e., to compensate for all the recognition errors for the lengthened poses.

1) Fuzzy Combination of Single-view Pose Classifiers: The camera placement being set up, the **multiple view pose detector** behaves as a fusion information unit. We explain now more formally how the multiview fusion is performed. We recall that our nonverticality detector is reliable: a lengthened pose detected in a given view must be related to a lengthened pose in the world, since a standing detected pose might be wrongly labeled. Thus, the fusion unit has to perform a logical OR between the lengthened pose detectors. In addition, we aim at performing a fuzzy logic fusion to take advantage of the likelihood of each detection θ'_i ($i \in \{1; N_c\}$), N_c being the number of cameras. Thus, we use a Gaussian modeling, the standing and lengthened states in the ith view being represented with Gaussian distributions $\eta(\mu_s, \sigma_s^i)$ and $\eta(\mu_l, \sigma_l^i)$, with mean value μ_s, μ_l and standard deviation σ_s^i, σ_l^i, respectively. This is an important specificity of our approach, because these parameters are not learned from training data, but are derived from (3). Thus, the mean values for the standing and lengthened states are $\mu_s = 0$ and $\mu_l = \pi/2$,

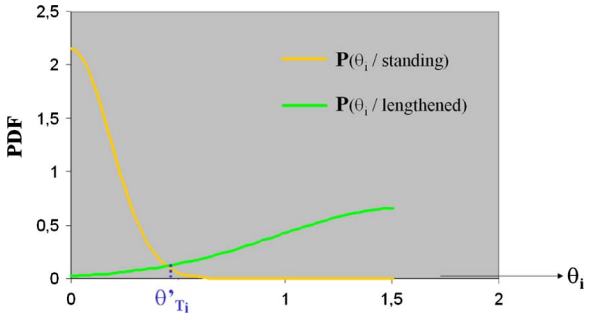


Fig. 5. Conditional states probability density functions.

respectively. The standard deviations are set up so that the probability density functions of the two states reach the same value at $\theta'_i = \theta'_{T_i}$, as illustrated in Fig. 5.

As we have $P(\text{lengthened}/\theta'_i) \sim P(\theta'_i/\text{lengthened})$, we can compute the lengthened likelihood $P(\text{lengthened}/\theta'_i)$ for a given detected angle θ'_i in the ith view. We then transpose the problem into the fuzzy logic domain, and we define the fuzzy set L_i corresponding to the lengthened state, with its associated membership function $\mu(L_i) = P(\text{lengthened}/\theta'_i)$. We then perform the fuzzy logical OR by determining the view c such as: $c = \arg\limits_i \left(\max\{\mu(L_i)\}_{i\in\{1;N_c\}}\right)$, i.e., we select the view maximizing the lengthened pose likelihood. The combined decision for the multiple view pose classification is performed as follows: if $\theta'_c < \theta'_{T_c}$, the multiview classifier output a standing pose, and a lengthened pose otherwise. In addition, we come back to the Bayesian domain, and compute the combined standing and lengthened pose probabilities as: $P(\text{lengthened}/\theta'_c) = \mu(L_c)$ and $P(\text{standing}/\theta'_c) = \mu(S_c)$.

V. Motion Analysis

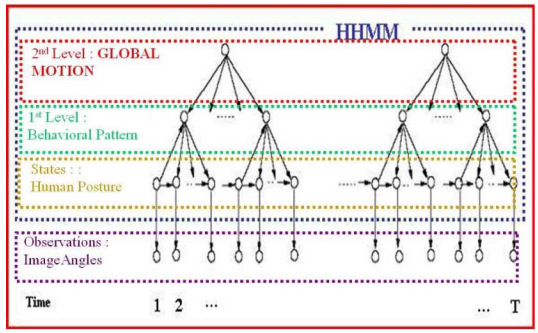
In the previous section, we explained our multiview pose classification strategy. Now, we detail how the pose sequence is analyzed, in order to recognize people motions, in particular, to detect people falling.

A. Motion Model Architecture

We propose to use a LHMM as a generative model for describing and identifying events. LHMMs are a special case of HHMM, where a decoupling between logical levels is carried out. In LHMMs, each layer of the architecture is connected to the next layer via its inferential results. This representation segments the problem into distinct layers that operate at different temporal granularities, as illustrated in Fig. 6.

The general model architecture is shown in Fig. 6(a). The observation vector corresponds to θ'_c, i.e., the combined deviation between the different views (see Section IV-D1). The model states are directly related to the human postures, and there are two states corresponding to standing and lengthened poses. In the first level of the LHMM, we define "elementary motions" or "behavioral pattern." These models are dedicated to representing and identifying sudden changes. For our fall detection purpose, we use three different motions models: "Is Walking," "Is Falling," and "Is Lying." The second hierarchy level represents global motions, that we denote BEHAVIOR. In

(a) General Architecture of the two-level LHMM



(b) Example: Modeling a FALL

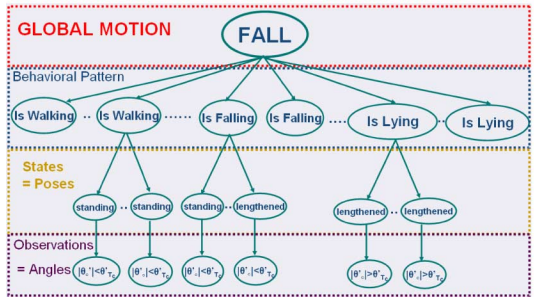


Fig. 6. Layered hidden Markov model architecture.

that sense, the LHMM corresponds to a Hidden Markov Model of Hidden Markov Model (meta-model). Indeed, in the last hierarchy level, the states of the model are Hidden Markov Models themselves. The global motions correspond to behavioral pattern sequences, and have a larger time extent. We use three kinds of such motions: a WALK model, a FALL model, and a LENGTHENING model. The global motion models have similar meaning as the elementary motion models, except that they operate at a larger temporal granularity level. For the sake of clarity, we will from now always refer to global motions with capital letters, and to behavioral pattern using quotes. Fig. 6(b) presents the overall FALL motion model, and illustrates how it is supposed to generate its corresponding behavioral pattern sequence, itself supposed to produce a given state sequence. We insist here on the strength of hierarchical architecture for our event detection purpose. First, as pointed out in [10], LHMM are superior to standard HMM because they encode prior knowledge about the problem, are less prone to over-fitting, and are no more difficult to set up. In addition, the states definition in our application is close to the human concepts, making the parameters setting easy to validate by a simple reasoning about the model semantics. For example, a behavioral pattern "Is Falling" must correspond to a transition from a standing state to a lengthened state, an elementary motion model "Is Walking" must correspond to a transition from a standing state to a standing state, etc.

B. LHMM Parameters

A standard HMM is characterized by the following set of parameters, $\lambda_k = (\pi_j, A_j, B_j)$, as follows.

- k is the temporal index and j corresponds to the index of a given state model, N_j being the total number of states.
- π_j is the initial probability of the jth state.

- A_j is the $N_j \times N_j$ transition matrix specifying the transition from a given state to any other state.
- B_j is the conditional probability of the jst state, given the observations.

As the LHMM is a hidden Markov model of hidden Markov models, the motion model parameters of the ist level of the hierarchy λ_k^i ($i \in \{1;3\}$) can be denoted $(\pi_j^i, A_j^i, B_j^i)^k$.

The layered formulation of LHMMs makes it feasible to decouple different levels of analysis for training. At the first level, i.e., for the "behavioral pattern," we recall that the observations correspond to the angle θ'_c determined by the fuzzy multiview fusion (see Section IV-D-I). The conditional states probability functions are thus fixed by the Gaussian distribution of the angle θ'_c in the cth view, as illustrated in Fig. 5. For the remaining parameters (i.e., transition matrices $(A_j^i)_k$ and initial state probabilities $(B_j^i)_k$), they are learned using the standard HMM technique, i.e., the Baum–Welch algorithm [19], an extension of the expectation-maximization (EM) [20]. Formally, considering m motion models at the ith hierarchy level, we define the forward $\alpha_{m,k}^i(j)$ and backward $\beta_{m,k}^i(j)$ variables as in [19], whose normalized product leads to $\gamma_{m,k}^i(j) = P\left(q_j^i = S_j^i | O(1:k)\right)$, i.e., the conditional likelihood of a particular state S_j^i at level i and time k, given the observations $O(1:k)$ up to time k. The log-likelihood of a sequence of observations for the mth motion model is given by

$$\mathcal{L}_{m,k}^i = \log P\left(O(1:k)\right) = \sum_{j=1}^{N_j} \alpha_{m,k}^i(j). \tag{6}$$

In particular, the expressions of $\alpha_{m,k}^i(j)$ and $\beta_{m,k}^i(j)$ can be iteratively computed using a dynamic programming technique. Finally, the transition matrices $(A_j^i)_k$ and initial state probabilities $(B_j^i)_k$ are directly obtained from α and β (see [19]).

C. LHMM Inference

At each hierarchy level, we aim at recognizing a given motion from the observation sequences, i.e., estimating the likelihood of each generative model, and identifying the one maximizing the *a posteriori* probability. This task, known as inference, can be exactly performed in the Hidden Markov Model formalism.

For inference, the LHMM again decouples the different levels of analysis. Thus, at each hierarchy level i, we attempt at determining the motion model maximizing the likelihood $\mathcal{L}_k^i$ defined in (6), i.e., finding $m_b = \underset{m}{arg}\left(\max\left\{\mathcal{L}_{m,k}^i\right\}_{m\in\{1;M\}}\right)$. For that, we first use the Viterbi algorithm [21]. This dynamic programing technique makes it possible to output the best state decoding sequence. For example, at the first level of the hierarchy, we estimate the likelihood of each elementary motion model ("Is Walking," "Is Falling," and "Is Lying") for generating the observed image angle sequences. With LHMM, there are two kinds of strategies for providing the observations when going up in the hierarchy (see [10]): the **maxbelief** and the **distributional** approach. In the former, only the motion model m_b with the largest likelihood $\left\{\mathcal{L}_{m_b,k}^i\right\}$ is provided, while in the later, the overall sequence of models with their associated likelihood $\left\{\mathcal{L}_{m,k}^i\right\}_{m\in\{1;M\}}$ is used. Contrary to [10], we notice a significant improvement in using the distributional approach and, therefore, use this strategy.

Thus, at the last level of the hierarchy, the likelihoods of the global motion models WALK, FALL, and IS LYING are estimated, using the overall set of behavioral patterns with their associated likelihood computed at the lower hierarchy level. Finally, if the most probable motion likelihood is below a given threshold, we define an "Unknown Model," that we identify as the recognized event.

VI. Performances Evaluation

A. Single View Evaluation

1) Posture Classification: The theoretical study proposed Section IV-C relates the deviation θ' with respect to the rectified vertical direction in the image plane and the deviation θ with respect to the vertical direction in the 3-D world. However, the threshold values for θ' (3) and φ (Property 4.3) actually correspond to upper bounds. They have been determined by coarse bounding that have been sufficient for deriving the properties, but that are not reached. In this section, we propose to numerically simulate (1) to estimate the threshold θ'_T, making the classification between standing and lengthened poses more accurate. The first step for the simulation consists in calibrating the camera (see [22]).

Numerical Simulation Results: As the height of the camera is a calibration result, we have $y_0 = h_c$ ($2390\ mm$). Thus, y_0 is considered constant in the simulation, and only $(x_0, z_0, \theta, \varphi)$ are varying. We present the simulation results with a top view, i.e., in the $-y$ direction of Fig. 3. We make (x_0, z_0) vary in their definition ranges, ensuring that they are inside the field of view. The simulation goal is two-fold. On the one hand, we want to validate the fact that every standing posture in the world is detected as a standing posture in the image, and estimate the maximum θ' value (small angles validation). On the other hand, we want to numerically determine the range of values for φ making it possible to properly identify all lengthened poses (large angles validation), and infer the number of cameras required to reach 100% true negative.

For the small angles validation, we define the standing postures in the 3-D world as those fulfilling $\theta < \theta_{\max} = \Pi/10$. In this setting, we have $\mathrm{S} = 2000$ mm, $\mathrm{d} = 4260$ mm and $\mathrm{L} = 1395$ mm.

For each simulation step, we sample θ' using (1). The maximum value θ'_T for properly identifying the standing poses as theoretically derived in (3) is 0.85 in our configuration. The nondetected standing poses (i.e., $\theta' > \theta'_T$) are represented by using false color coding, from blue to red. This is illustrated in Fig. 7, the value $z = S$ from which (3) can not be violated is shown with a red horizontal line. Fig. 7(a) illustrates the fact that the upper bound on θ' defined by (3) is actually not reached: no labeling error occurs for $z < S'$, with $S' > S$, S' being represented by a green horizontal line. We thus determine the maximal reached θ'_E value for θ' with our simulation, and we find $\theta'_E = 0.37$. We can now guarantee that if a person has a deviation with respect to the vertical direction in the world smaller than $\Pi/10$, the angle measured in the image plane must be smaller than 0.37. Fig. 7(b) illustrates the simulation results

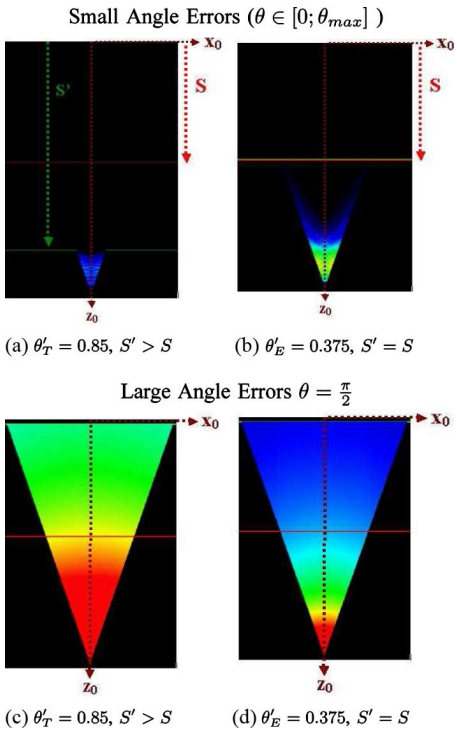


Fig. 7. Numerical simulation results.

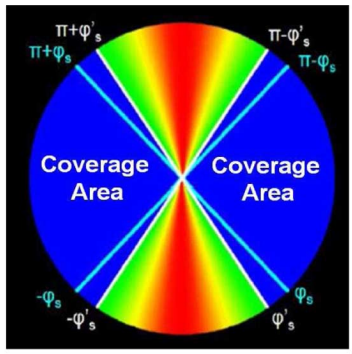


Fig. 8. φ Validity domain for the lengthened pose detection. A two-view system can manage to detect all lengthened poses. See text.

Fig. 9. FALL detection with the LHMM.

using the threshold θ'_E, and we can verify that the first labeling error occurs for $z = S$.

The large angle validation processes in the same manner, except that θ is fixed to $\Pi/2$. We again evaluate θ' by sampling the projection formula defined at (1), and we check if $\theta' > \theta'_T$ (good detection i.e., true positive), or $\theta' \leq \theta'_T$ (false negative). Fig. 7(c) illustrates the pose labeling errors for $\theta'_T = 0.85$, and Fig. 7(d) illustrates the pose labeling errors for $\theta'_E = 0.37$. Again, we can notice that the labeling errors are sensitively decreased in the latter case.

Fig. 8 illustrates the simulation results related to the φ influence on the lengthened pose detection, by presenting the false negative repartition ($\varphi \in [-\pi/2; \pi/2]$). The blue region indicates the absence of errors (false negative = 0%), while a false color coding from yellow to red is used to represent the error rate magnitude. From the Property 4.3, we find a theoretical value $\varphi_S = 44°$ from which we can guarantee a 100% true negative rate. It corresponds to a coverage area $\rho_c \approx 51\%$. Thus, we are just on the borderline case where two cameras are sufficient for detecting 100% of the lengthened poses whatever the falling direction. Moreover, we can again notice that the limit expermimental value is smaller than the theoretical one. We find a limit angle $\varphi'_S = 37°$, corresponding to a coverage area of $\rho'_c \approx 63\%$. In our configuration, two cameras are thus widely sufficient for removing all false negatives.

2) Single View Motion Analysis Performances: We present here the performances of the motion recognition performed in a single view context.

Fig. 9 illustrates the motion analysis by means of the Layered Hidden Markov Model. From the Frame 10, the elementary motion model "Is Walking" is recognized, meaning that it maximizes the *a posteriori* probability of observing the image angle sequences. The most probable state sequence, computed by means of the Viterbi algorithm, corresponds thus to a sequence of standing poses. However, at Frame 40, the behavioral pattern "Is Falling" becomes the motion model with the maximum likelihood, indicating that the system detects a transition from a set of standing poses to a set of lengthened poses. Finally, from the Frame 60 and until the Frame 260, the elementary motion model "Is Lying" is recognized as a sequence of lengthened postures. The motion analysis at a largest time scale, carried out by the second level of the hierarchy, leads to properly detect the global motion FALL, as the most probable for generating a sequence of "Is Walking," "Is Falling," and "Is Lying" behavioral pattern.

Fig. 10 illustrates the robustness of the Layered architecture to low-level motion recognition errors. Falls being extremely sudden motions, the elementary motion pattern are supposed to have a sufficient fine temporal sensitivity to detect them. In counterpart, the detection accuracy is paid in terms of robustness, and wrongly identifying a behavioral pattern in real situations is unavoidable. Indeed, neither the motion segmentation nor the feature extraction are perfect, and these low-level steps affect the inference algorithm for recognizing a given elementary motion pattern. Thus, although the "Is Walking" behavioral pattern is properly identified between Frames 25 to 266 of Fig. 10, an elementary motion model "Is Falling" is wrongly recognized around Frame 278. In that case, this corresponds to a motion segmentation error, which is altered by the shadow that is not completely removed although the use of an color space invariant in luminance. As a consequence, the angle computation is disturbed, lengthened states become more probable than standing ones, and the motion model "Is Falling" is occasionally the generative model that best explains the observed data. Thus, the requirement for systems with a fine temporal sensitivity would lead to the emission of a significant number of

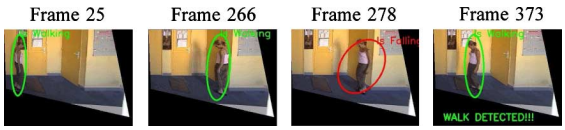


Fig. 10. Robustness of the recognition to low-level errors.

false alarms, with a single level Hidden Markov Model. However, we explain how the second level of the hierarchy makes it possible to filter out these false detections. Indeed, although the "Is Falling" behavioral pattern is recognized at Frame 278, the global motion WALK is properly identified, as illustrated in Frame 373. At a larger time scale, the WALK motion remains the most probable to have generated the elementary motion sequence, because a single false detection of the "Is Falling" behavioral pattern occurred. Its likelihood is decreased, but remains larger than the FALL motion likelihood, which would have required a sequence of "Is Walking," "Is Falling," and "Is Lying" elementary motion models. Thus, the second level of the hierarchy can be interpreted as a way to perform Top/Down verifications, and wrongly identifying a global motion due to low-level errors is heavily reduced when performing the inference on the LHMMl.

B. Multiple View Evaluation

The single view motion analysis system is mainly limited by the pose classification step, that might fail at detecting people falling in a direction "too close" to the optical axis. The multiple view pose classifier proposed in Section IV-D offers a solution to overcome that shortcoming. In our experiments, we notice that only two cameras are needed for providing a system able to properly label lengthened poses whatever the falling directions. In Fig. 11, a 2-view system for fall detection is presented. In Fig. 11, a person is falling, and the 2-view system proves to be able to detect it. It can be pointed out that the fall takes place in the optical axis direction of the camera 1 (first row). This is thus the worst case for this viewpoint. As we can notice in Frames 139, 167, and 258, the principal axis of the tracked person is very close to the vertical rectified direction in the three cases, and the pose classification would fail at detecting a lengthened pose in a single-view context. This is illustrated in the first row, where the green best ellipse indicates that "Is walking" elementary motion model would be the most probable if only relying on the camera 1 output. However, the pose variation from standing to lengthened can easily be detected in the second camera, as illustrated in the second row. Thus, the fuzzy logical fusion at the pose detection step explained in Section IV-D makes the 2-view system able to properly identity the lengthened poses from Frames 167 to 258. Therefore, the elementary motion pattern "Is Walking," "Is Falling," and "Is Lying" are properly identified when combining the two views, and the global motion FALL is identified at Frame 258.

C. Experimental Validation

In order to validate the overall system performances, including segmentation, tracking and recognition, we propose to perform an experimental evaluation of the proposed fall detection algorithm. Thus, we test our approach on a sample of fifty

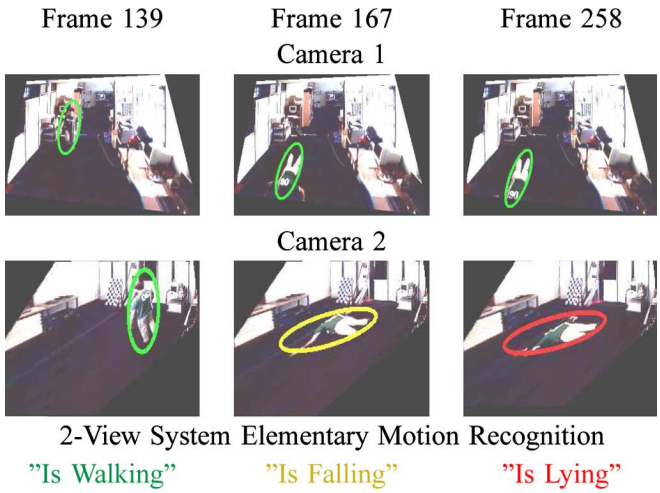


Fig. 11. Multiple view performances. The 2-view system can successfully detect the FALL by combining the pose classification outputs. See text.

TABLE I
EXPERIMENTAL RESULTS

Truth \ Detection	Single-View			Two-View		
	FALL	WALK	?	FALL	WALK	?
FALL	41	8	1	49	0	1
WALK	0	49	1	0	50	0

cases of falls and fifty cases of walks. We try to use sequences that are relevant to evaluate the robustness of the system. Results are presented in Table I. The ? represents the case where the unknown event has been detected as the most probable. We can notice that the single-view system already almost never sends false alarms: for WALK motions we have a detection rate of 98%. For real falling cases we obtain a rate of 82% correct detections (and then 18% of false negatives). As we can notice, the performances obtained for the overall system are comparable to those corresponding only to the posture detection, and evaluated by the numerical simulation (see Section VI-A1a). It does not mean that the low level steps of the system provide perfect results, but rather demonstrates the capacity of the hierarchical motion model to perform Top/Down verifications that are able to incorporate *prior* knowledge able to filter out the low-level errors. The single-view false negatives that remain after the analysis at the second level of the LHMM are mainly a consequence of mis-identifications between standing and lengthened poses. Indeed, the evaluation carried out with the two-view system proves that they can be removed by the fusion unit, that is able to efficiently combine the pose classification.

D. Processing Time

Table II gathers the complexity of the main steps of the proposed algorithm. The experiments have been carried out on a Pentium IV at 2.66 GHz, with 512 MB of RAM. The software has been built in C++ using Microsoft Visual studio 2005 (version 7.0). The video sequences were composed of image sequence with CIF resolution (i.e., of size 320×240).

As we can see, the background subtraction is the most computationally demanding step. The pose estimation includes feature extraction and (multiview) classification, and these two steps

TABLE II
COMPUTATION TIME FOR THE MAIN STEPS OF THE MOTION ANALYSIS ALGORITHM

	Motion Detection	Pose Estimation	Motion Analysis	Total
Time(ms)	13	2	10	36

are very fast. Similarly, the multiview fusion unit dedicated to providing a fuzzy logical OR between the lengthened pose detectors can be performed very quickly. The motion analysis by means of the Layered Hidden Markov Model requires about 8 ms. It is relatively demanding because it has been set up with a maximum accuracy. Indeed, we choose to run the inference process at each time step, and we use a distributional approach for providing the observations at the second hierarchy level (see Section V-C). The background subtraction is performed in each camera independently, so that the processing time is given per view. Contrarily, the processing times for the other steps are given in a 2-view context. Thus, the overall required time for fall detection with our 2-view system is about $2 * 13 + 2 + 8 = 36$ ms.

VII. CONCLUSION AND FUTURE WORKS

In this paper, we propose a multiview system dedicated to fall detection, that is compatible with real-time purpose and that explicitly addresses the posture classification and motion modeling issues. The proposed algorithm detect, track, and extract features independently in each view. Then, a fusion unit merges the posture analysis to provide a standing/lengthened pose classifier that is efficient in unspecified viewpoints and falling directions. From the pose likelihood estimation, the inference is performed regarding all the cameras jointly, and is managed by using a LHMM. This association deals with sudden changes and is robust to low-level errors. The direction for future works mainly include a more important cooperation between views for the low-level steps of the algorithm to improve robustness.

APPENDIX A
DERIVATION OF EQUATION 1

Regarding vector $\vec{V}$ defined in Fig. 3, we use homogeneous coordinates so that the origin of $\vec{V}$ is $V_o = (x_0, -y_0, z_0, 1)^T$ and its extremity is $V_e = \begin{pmatrix} x_0 + r * \sin(\theta)\sin(\varphi) \\ -y_0 + r * \cos(\theta) \\ z_0 + r * \sin(\theta)\cos(\varphi) \\ 1 \end{pmatrix}$ where $r = \left\|\vec{V}\right\|$.

Assuming pinhole model for the camera, and given the notations of Section IV-C, the projection relating the 4-D coordinates of a point in the world (in the $(0xyz)$ coordinate system) and its 4-D coordinates after the projection on the image plane (in the $(0x'y'z)$ coordinate system) can be expressed by the following 4×4 matrix H_p (see [23])

$$H_p = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{d}{f} & \frac{d^2}{f} + d \\ 0 & 0 & -\frac{1}{f} & \frac{d}{f} + 1 \end{pmatrix}. \tag{7}$$

Applying H_p to V_o and V_e, we compute their coordinates after projection, that we denote V'_o and V'_e, respectively.

Both V'_o and V'_e lie on the image plane ($z = d$). Let us denote $\vec{V}' = \overrightarrow{0V'_e} - \overrightarrow{0V'_o}$ the image of $\vec{V}$ in the image plane after projection, as illustrated in Fig. 3. We have (8), shown at the bottom of the page.

We can compute the angle θ' between $\vec{V}'$ and the vertical direction on the image plane in the following way:

$$\tan(\theta') = \frac{V'_x}{V'_y} = \tan(\theta)\frac{(d + f - z_0)\sin(\varphi) + x_0\cos(\varphi)}{(d + f - z_0) - y_0\tan(\theta)\cos(\varphi)} \tag{9}$$

where V'_x and V'_y are the coordinates of $\vec{V}'$ in the $(O'x')$ and $(O'y')$ directions, respectively.

APPENDIX B
PROOF OF PROPERTY 4.1

Let us denote

$$\tan(\theta') = \frac{N}{D} = \frac{[(d + f - z_0)\sin(\varphi) + x_0\cos(\varphi)] * \tan(\theta)}{(d + f - z_0) - y_0\tan(\theta)\cos(\varphi)}.$$

Provided that $\begin{cases} z_0 < S \\ x_0 \in [-L; L] \\ \theta \in [0; \theta_{\max}] \end{cases}$, we have[1]

$$\begin{aligned} d + f - S < d + f - z_0 < d + f^* \\ -(d + f) < (d + f - z_0)\sin(\varphi) < d + f \\ -L < x_0\cos(\varphi) < L, \text{ so that} \\ -\tan(\theta_{\max})(d + f + L) < N < \tan(\theta_{\max})(d + f + L). \end{aligned} \tag{10}$$

Similarly, we have

$$d + f - S - |y_0|\tan(\theta_{\max}) < D < d + f + |y_0|\tan(\theta_{\max}). \tag{11}$$

Let us denote $m = d + f - S - |y_0|\tan(\theta_{\max})$ and

[1]We recall that the coordinate system $(OXYZ)$ illustrated in Fig. 3 is defined so that its origin O is the point that bounds the filed of view in the $-z$ direction. Thus, it is implicitly required that $z_0 \geq 0$.

$$\vec{V}' \begin{pmatrix} \frac{f * x_0 * r * \sin(\theta)\cos(\varphi)}{(d + f - z_0 - r * \sin(\theta)\cos(\varphi))(d + f - z_0)} + \frac{f * r\sin(\theta)\sin(\varphi)}{d + f - z_0 - r * \sin(\theta)\cos(\varphi)} \\ \frac{-f * y_0 * r * \sin(\theta)\cos(\varphi)}{(d + f - z_0 - r * \sin(\theta)\cos(\varphi))(d + f - z_0)} + \frac{f * r\cos(\theta)}{d + f - z_0 - r * \sin(\theta)\cos(\varphi)} \\ 0 \\ 0 \end{pmatrix} \tag{8}$$

$M = d + f + |y_0| \tan(\theta_{\max})$. We can bound $1/D$ iff m and M are the same sign. As M is necessary positive, it requires that $m > 0$, i.e., $\theta_{\max} < \arctan(d + f - S/|y_0|)$. In that case, we have $D > 0$ and

$$\frac{1}{d + f + |y_0| \tan(\theta_{\max})} < \frac{1}{D} < \frac{1}{d + f - S - |y_0| \tan(\theta_{\max})} \tag{12}$$

and

$$-\frac{\tan(\theta_{\max})(d + f + L)}{d + f - S - |y_0| \tan(\theta_{\max})} < \frac{N}{D} < \frac{\tan(\theta_{\max})(d + f + L)}{d + f - S - |y_0| \tan(\theta_{\max})} \tag{13}$$

which proves Property 4.1.

Appendix C
Proof of Property 4.2

We consider here the case $\theta = \pi/2$. Thus, we have

$$\vec{V}' \begin{pmatrix} \frac{f*x_0*r*\cos(\varphi)}{(d+f-z_0-r*\cos(\varphi))(d+f-z_0)} + \frac{f*r\sin(\varphi)}{d+f-z_0-r*\cos(\varphi)} \\ \frac{-f*y_0*r*\cos(\varphi)}{(d+f-z_0-r*\cos(\varphi))(d+f-z_0)} \\ 0 \\ 0 \end{pmatrix} \tag{14}$$

and

$$\tan(\theta') = \frac{V'_x}{V'_y} = -\frac{x_0 + \tan(\varphi)(d + f - z_0)}{y_0}. \tag{15}$$

We want to find conditions on φ so that $|\theta'| > \theta'_T$ in the case of $\theta = \pi/2$, i.e., our detector properly detect a Lying pose

$$|\tan(\theta')| > \tan(\theta'_T) \Leftrightarrow |-x_0 - \tan(\varphi)(d + f - z_0)| > \tan(\theta'_T)\,|y_0|\,. \tag{16}$$

1) If $(-x_0 - \tan(\varphi)(d + f - z_0)) > 0$ we get

$$\tan(\varphi) < \frac{-x_0 - \tan(\theta'_T\,|y_0|)}{d + f - z_0}. \tag{17}$$

2) Else if $(-x_0 - \tan(\varphi)(d + f - z_0)) < 0$ we get

$$\tan(\varphi) > \frac{-x_0 + \tan(\theta'_T\,|y_0|)}{d + f - z_0}. \tag{18}$$

If $\varphi \in [-\pi/2; \pi/2]$, (17) leads to $\varphi_2 = \arctan\left(-x_0 - \tan(\theta'_T)\,|y_0|/d + f - z_0\right)$ and (18) leads to $\varphi_1 = \arctan\left(-x_0 + \tan(\theta'_T)\,|y_0|/d + f - z_0\right)$, as defined in Property 4.2.

If $\varphi \in [\pi/2; 3\pi/2]$, the solution to (17) is $\varphi_3 = \varphi_2 + \pi$, and to (18) is $\varphi_4 = \varphi_1 + \pi$, because the tangent functional is π periodical.

Appendix D
Proof of Property 4.3

Property 4.2 defines the coverage area with respect to φ at a given (x_0, y_0, z_0) location. Let us define $P(x_0, z_0) = -x_0 - \tan(\theta'_T)\,|y_0|/d + f - z_0$ and $H(x_0, z_0) = -x_0 + \tan(\theta'_T)\,|y_0|/d + f - z_0$. As $z_0 < S$ and $x_0 \in [-L; L]$, we can derive

$$P(x_0, z_0) < \frac{L + \tan(\theta'_T)\,|y_0|}{d + f - S}$$
$$H(x_0, z_0) > -\frac{L + \tan(\theta'_T)\,|y_0|}{d + f - S}. \tag{19}$$

Thus, we prove that if $\tan(\varphi) > L + \tan(\theta'_T)\,|y_0|/d + f - S$ or $\tan(\varphi) < -L + \tan(\theta'_T)\,|y_0|/d + f - S$, then $|\tan(\theta')| > \tan(\theta'_T)$. Again, if $\varphi \in [-\pi/2; \pi/2]$, $\varphi_S = \arctan\left(L + \tan(\theta'_T)\,|y_0|/d + f - S\right)$ defines the coverage area, i.e., the lying pose is properly detected if $\varphi < -\varphi_S$ or $\varphi > \varphi_S$. If $\varphi \in [\pi/2; 3\pi/2]$, the coverage area is defined by the values $\{-\varphi_S + \pi; \varphi_S + \pi\}$.

References

[1] C. Sminchisescu and B. Triggs, "Estimating articulated human motion with covariance scaled sampling," *IJRR*, vol. 22, no. 6, pp. 371–391, Jun. 2003.

[2] G. Mori, "Recovering 3D human body configurations using shape contexts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 7, pp. 1052–1062, Jul. 2006.

[3] P. F. Felzenszwalb and D. P. Huttenlocher, "Pictorial structures for object recognition," *Int. J. Comput. Vis.*, vol. 61, no. 1, pp. 55–79, 2005.

[4] I. Haritaoglu, D. Harwood, and L. Davis, "W4: Real-time surveillance of people and their activities," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 8, pp. 809–830, Aug. 2000.

[5] H. Zhong, J. Shi, and M. Visontai, "Detecting unusual activities in video," *Comput. Vis. Pattern Recognit.*, 2004.

[6] J. W. Davis and A. F. Bobick, "The representation and recognition of action using temporal templates," *Comput. Vis. Pattern Recognit.*, 1997.

[7] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, pp. 257–286, 1989.

[8] N. M. Oliver, B. Rosario, and A. Pentland, "A bayesian computer vision system for modeling human interactions," *Pattern Anal. Mach. Intell.*, pp. 831–843, 2000.

[9] A. Galata, N. Johnson, and D. Hogg, "Learning variable length Markov models of behaviour," *CVIU*, pp. 398–413, 2001.

[10] N. Oliver, E. Horvitz, and A. Garg, "Layered representations for human activity recognition," in *Proc. 4th IEEE Int. Conf. Multimodal Interfaces*, 2002, pp. 3–8.

[11] A. Sixsmith and N. Johnson, "Simbad: Smart inactivity monitor using array-based detector," *Gerontechnology*, 2002.

[12] B. Lo, J. Wang, and G. Yang, "From imaging networks to behavior profiling: Ubiquitous sensing for managed homecare of the elderly," *Pervasive*, 2005.

[13] H. Nait-Charif and S. McKenna, "Activity summarisation and fall detection in a supportive home environment," *ICPR*, pp. 323–326, 2004.

[14] B. U. Töoreyin, Y. Dedeoglu, and A. E. Çetin, "Hmm based falling person detection using both audio and video," *ICCV-HCI*, pp. 211–220, 2005.

[15] R. Cucchiara, H. Rita, A. Prati, O. Andrea, R. Vezzani, and C. Roberto, "A multi-camera vision system for fall detection and alarm generation," *Expert Syst.*, vol. 24, no. 5, pp. 334–345, Nov. 2007.

[16] C. Stauffer and W. Grimson, "Learning patterns of activity using realime tracking," *Pattern Anal. Mach. Intell.*, 2000.

[17] N. Thome and S. Miguet, "A robust appearance model for tracking human motions," *AVSS*, pp. 528–533, 2005.

[18] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed ed. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[19] J. R. Deller, J. G. Proakis, and J. H. L. Hansen, *Discrete-Time Processing of Speech Signals*, 2nd ed ed. New York: Macmillian, 1993.

[20] A. Dempster, N. Laird, and D. Rubin, "Maximum likelylihood from incomplete data via the em algorithm," *J. Roy. Statist. Soc.*, vol. 39, no. 1, pp. 1–38, 2003.
[21] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inf. Theory*, vol. IT-13, no. 2, pp. 260–269, Feb. 1967.
[22] F. Lv, T. Zhao, and R. Nevatia, "Self-calibration of a camera from video of a walking human," *ICPR*, 2002.
[23] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes, *Computer Graphics: Principles and Practice in C*, 2nd ed ed. Reading, MA: Addison Wesley, 1995.

Nicolas Thome received the diplome d'Ingénieur from the École Nationale Supérieure de Physique de Strasbourg, France, the DEA (M.Sc.) degree from the University of Grenoble, France, in 2004, and the Ph.D. degree in computer science from the University of Lyon, France, in 2007.

He is currently a Postdoctoral Associate at INRETS, Villeneuve d'Ascq, France. His main research focuses on video analysis, particularly on human detection, tracking, and motion interpretation.

Serge Miguet (M'94) graduated from the École Nationale Supérieure de Mathématiques appliquées de Grenoble, France, and received the Ph.D. degree in computer science from the École Normale Supérieure, Lyon, France.

Since 1996, he has been a Full Professor at the University of Lyon. He is specialized in image and video processing and is interested in fundamental aspects of image description with discrete geometry tools, as well as distributed computing for managing large databases.

Śebastien Ambellouis received the Ph.D. degree in computer science and control from the University of Science and Technology, Lille, France.

He is an Engineer with the ESIGELEC School of Rouen, France. Since 2001, he has been a Researcher in the LEOST Laboratory, INRETS, Villeneuve d'Ascq, France, and is working on computer vision and signal processing for transport application.