Domotics for disability: smart surveillance and smart video server

R. Cucchiara, A. Prati, R. Vezzani

Dipartimento di Ingegneria dell'Informazione, Università di Modena e Reggio Emilia
Via Vignolese, 905 - 41100 Modena – Italy
{cucchiara.rita, prati.andrea, vezzani.roberto}@unimore.it - http://imagelab.ing.unimo.it

Abstract. In this paper we address the problem of human posture classification, in particular focusing to an indoor surveillance application. The approach was initially inspired to a previous works of Haritaoglou et al. [6] that uses histogram projections to classify people's posture. Projection histograms are here exploited as the main feature for the posture classification, but, differently from [6], we propose a supervised statistical learning phase to create probability maps adopted as posture templates. Moreover, camera calibration and homography is included to resolve prospective problems and improve the precision of classification. Furthermore, we make use of a finite state machine to detect dangerous situations as falls and to activate a suitable alarm generator. The system works on line on standard workstation with network cameras.

1. Introduction

Home automation, environmental control and tele-assistance are three hot keywords in current research activities of the computer engineering community. The emerging technologies can offer a very interesting contribution in improving the quality of the life of people in the house, and especially of people with some forms of disability.

The research in computer vision on people surveillance joint with the research in efficient remote multimedia access make feasible a complex framework where the people in the home could be monitored in their daily activity in a fully automatic way, therefore in total agreement with privacy policies. A well formalized set of alarm situations can be defined and can be used as the trigger of some actions such as the communication to remote users, control center or private person. Finally, only in such a situation, remote users can connect also with low-cost devices, such as GPRS phones and PDAs.

In this context, we are developing an integrated framework that aims at exploiting the potentiality of automatic video annotation for detecting dangerous events and, therefore, reacting by sending an alarm to a control center. A set of computer vision and motion analysis techniques are used to extract objects and events from the scene, according with a previously defined ontology. In this specific case, moving people are the main objects of interest, while, through body modeling and posture recognition, a

state transition-based high-level reasoning module is able to extract interesting events, such as the fall of the monitored person.

The most innovative part of our proposal is the definition of a machine learning phase to construct very simple and reliable posture models used in an on-line posture model free classifier.

In this paper, after a brief presentation of related works in Section 2, the proposed framework is presented in Section 3. Section 4 gives details of our proposal of probabilistic templates of projection histograms and Section 5 shows some tests on real videos. Conclusions end the paper in Section 6.

2. Related Works

Recently an increasing number of computer vision projects dealing with detection and tracking of human posture have been proposed and developed. An exhaustive review of proposals addressing this field was written by Moeslund and Granum in [7], where about 130 papers are summarized and classified according with several taxonomies. In particular, they consider three different application fields: video surveillance, control and pure analysis. Our proposal can be included in the first class.
According with [7], we can classify most of them into two basic approaches to the problem.

From one side, some systems (like Pfinder [10] or W4 [5]) use a direct approach and base the analysis on a detailed human body model: an effective example is the Cardboard Model [6]. In many of these cases, an incremental predict-update method is used, retrieving information from every body parts. Nevertheless these systems are generally too sensitive, when loosing information about features, needing often a reboot phase. For this reason, the segmentation process has to be as accurate as possible using specific human cues (e.g., skin detection). Often this can contribute to system instability because some of these features could not be found in every frame (due to overlapping, for example).

In order to bypass these drawbacks, where no body parts control is necessary, many researchers deal with the problem in a indirect way using less, but more robust, information about the body. Many of these approaches are based on human body silhouette analysis. The work of Fujiyoshi et. Al. [3.] uses a synthetic representation (Star Skeleton) composed by outmost boundary points. In [4] Haritaoglu et al. add to W4 framework [5] some techniques for human body analysis using only information about silhouette and its boundary. They first use hierarchical classification in main and secondary postures, processing vertical and horizontal histogram profiles (or *projections*) from the body silhouette. Then, they locate body parts on the silhouette boundary's corners.

Our approach is similar to [4], as well as concerning projections features, but, differently from it, it is not based on a priori defined model. In fact, the main strength of our approach is a machine learning phase, exploited to create feature models, further used in the classifier.

3. The Proposed Framework

In order to be as flexible as possible, we constraint the problem's dimensionality to the 2D case, reducing the analysis to single camera images. Therefore, we suppose to have a point of view where the human posture can be easily perceived without ambiguities also in the image plane. Nevertheless, we exploit a 2D ½ space, by computing homography after camera calibration in order to have an approximate position of the person in the 3D space.

We consider video from indoor environment with a calibrated fixed camera. The method can be also exploited in outdoor environment, but we exclude high luminance variation and other artifacts that could add difficulties in moving object detection. Moreover, in this paper we do not address problems of people overlapping, neither problems of occlusion of human parts, and we consider only one person at a time in the room. The extension to more than one people is straightforward. The problem of occlusion is, instead, critical: we adopted an approach based on tracking with probabilistic and appearance model as in [9]. This approach is not discussed in this paper because it is beyond its scopes.

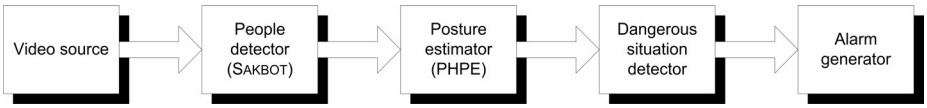

Fig. 1. A simplified scheme showing the main modules of the overall framework

The overall framework is composed by several modules, as shown in Fig. 1. In particular three internal processing modules are in pipeline:

- a *people detector*; it takes as input the frame sequence from a video source and produces for each frame the list of people present in the scene; the appearance, the extent coordinates, and some other features are extracted for each people element;
- a *posture estimator*; each people element detected in the previous step is analyzed to identify the current posture;
- a *dangerous situation detector*; this module is able to identify some anomalous situations only by means of temporal analysis of the people posture. When a dangerous event is detected, this module generates a suitable alarm.

The *people detector* module is based on SAKBOT (Statistical And Knowledge Based Object deTector) system [1], that is able to extract moving objects in many different conditions exploiting a complex background suppression algorithm.

The background is computed by means of an adaptive temporal median and a feedback from extracted knowledge of the objects in the previous frame is used to improve background model reactivity. Moreover, SAKBOT is able to detect and remove shadows reliably. At the end of this step we can detect a MVO (Moving Visual Object), discriminating it from shadows and other artifacts (as "ghost" or something with only an apparent motion due to background errors).

In this structured indoor environment we suppose that the moving object with an elongated shape and a size larger than a fixed parameter could be classified as people. Objects differing from peoples, noise, ghosts and shadows concur for a reliable background update.

A MVO detected and classified as a person is saved, its status (motion/stopped/static) is stored together with many visual features that are computed on its appearance blob.

The *posture estimator module* provides posture classification frame by frame. No past information are exploited to identify the current posture. We consider only four main posture (see Fig. 2): standing (a), sitting (b), crawling (c), and lying down (d). This module will be accurately described in the next section.

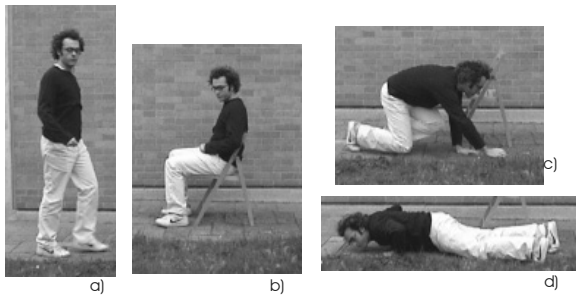

Fig. 2. The four postures considered by the system

Even if tracking is not directly exploited for each single posture classification, the posture history is used in the following module: the *dangerous situation detector*. In particular, in Fig. 3 the finite state machine used for the recognition of a fall is shown. A person is assumed to be detected in standing position at the beginning (in an indoor environment the people typically enter in the rooms standing). The transition between *moving* and *static* states (see Fig. 3) depends on the average motion of the tracked blob, while the transition between *moving* states is guided by changes in the posture classification. After a too long permanence in the laying down static state an alarm is generated. Note that there is no connection from standing to laying states: this transition can produce a warning because probably the system is failing.

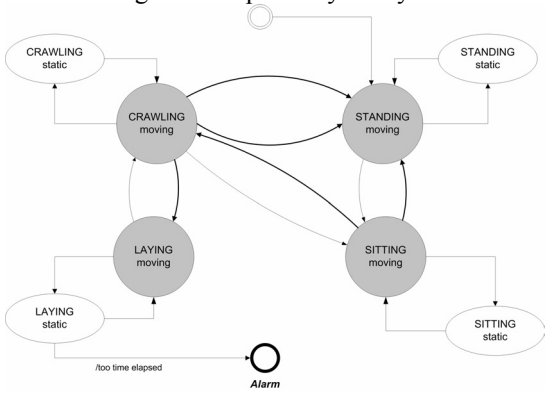

Fig. 3. Finite State Machine that allows the identification of a fall

The alarms corresponding to dangerous situations (in our case the falls) can be managed in several ways. For example, a control center can be advised and connected through a video-audio link with the assisted person. Besides, a vocal message or a SMS can be sent to a relative or a neighbor on their cell phone or PDA, and, in this last case, provide a link for a low-bandwidth video connection to assert person's conditions. Finally, all the events can be saved on a database for further processing.

We have developed a specific server for PDA clients that is able to manage the communication allowing the PDA to access to some semantically valid part of the video only [2].

4 Projection Histograms based Posture Estimator

In order to recognize human body posture, we used a knowledge classification inspired to the one proposed by [4]. As aforementioned, we discriminate four main postures which represent our classifier's states: *standing, crawling, laying down* and *sitting*. Since the silhouette of people sitting with a frontal, left or right view are very different, internally the system splits each state in three view-based subclasses: *front-view, left-view and right-view*. Then, the total set *RP* of the *K recognized posture* is:

$$RP = \{ Standing\_F, Standing\_L, Standing\_R, Crawling\_F, Crawling\_L, Crawling\_R, Sitting\_F, Sitting\_L, Sitting\_R, Laying\_F, Laying\_L, Laying\_R \} \quad (1)$$

The block diagram of the *Projection Histograms-based Posture Estimator* (PHPE) is reported in Fig. 4. Differently from [4] we use camera calibration to scale detected blobs in order to have a posture classifier invariant to people position in the room. Moreover, we adopt a supervised statistical learning phase to create the probability maps.

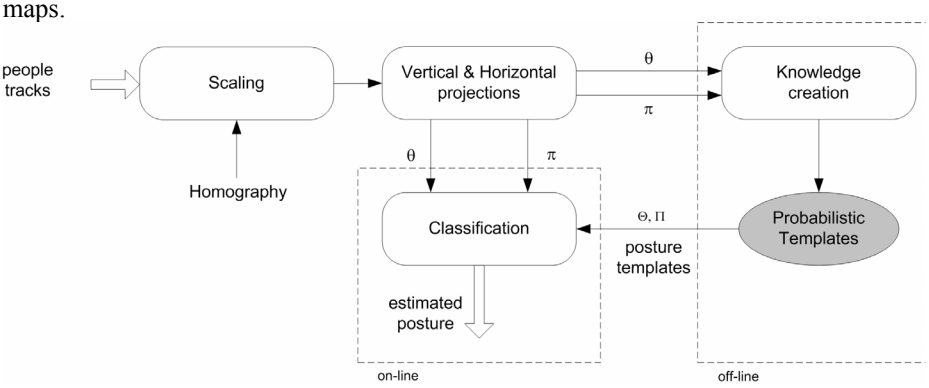

Fig. 4. Block diagram of the PHPE module

Let us describe the different modules.

4.1 Track Scaling

The main lack of the direct use of the silhouette of the people to estimate the posture is the dependence on the distance between people and camera. The work [4] suggests a normalization phase after the computation of the projection histograms, but, in this way, without a relation with the 3D space, many errors can arise: for instance, the two postures *standing_F* and *crawling_F* can be easily confused. To this aim we exploit camera calibration to compute the distance d of the people from the camera and we use this value to scaling the tracks. Choosing a suitable normalization distance D, we can use the rate sf as a scaling factor to remove the abovementioned problem, where:

$$sf = d / D \tag{2}$$

Applying the scale factor sf to the people tracks is analogous to "shift" all the people detected at a fixed distance D from the camera.

The d measure depends on the position of the support point SP (i.e., the contact point of the person with the Z=0 ground plane). Normally it corresponds with the foot position, but it is not necessarily true in the case of person laying down on the floor. If the camera's point of view is frontal, SP could be easily computed as the point with the maximum y coordinate (see Fig. 5). If more points present the same y-coordinate, SP could be randomly selected or computed as the middle point.

Once we obtained the SP image coordinates with this simple algorithm, we can compute the world coordinates of this point by using the projection equations of the pin-hole model and assuming Z_{SP}=0 (i.e., the support point lies on the floor).

Fig. 5 reports two schemes useful to understand the equation (3) exploited to convert between image coordinates and real ones.

$$\beta = arctg\left(\frac{y_{SP}}{f_y}\right) \quad X_{SP} = \frac{x_{SP}}{f_x} \cdot Y_{SP} \quad d = \sqrt{X_{SP}^{\ 2} + Y_{SP}^{\ 2}}$$
$$\alpha = \frac{\pi}{2} - \tau - \beta \qquad Y_{SP} = h \cdot tg(\alpha) \tag{3}$$

where h is the height of the camera with respect to the floor, SP the support point having image coordinates (x_{SP},y_{SP}) and real coordinates $(X_{SP},Y_{SP},Z_{SP}=h)$, f_x and f_y the focal lengths acquired by camera calibration.

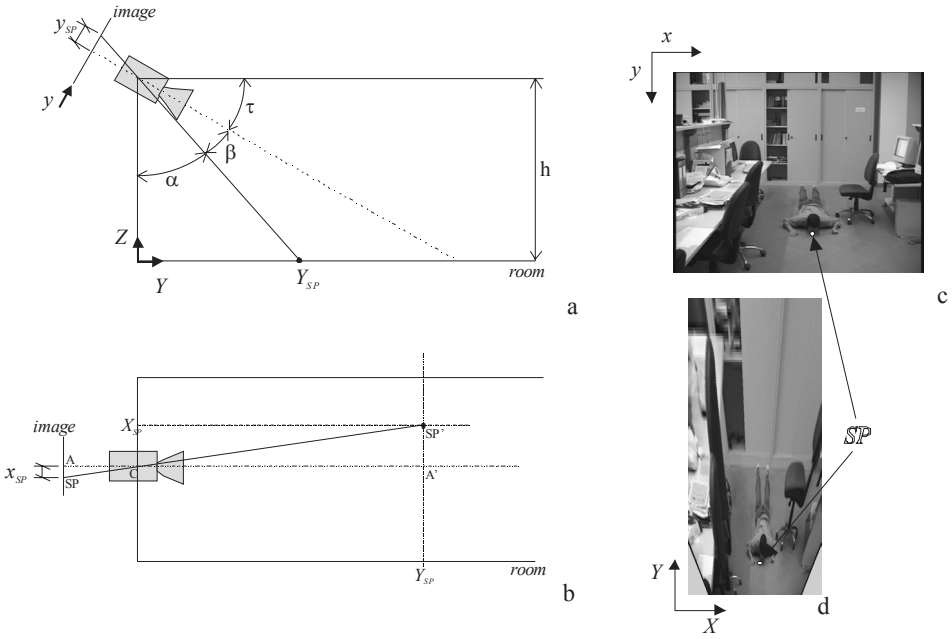

Fig. 5. a) lateral view and b) upper view of the pin-hole camera model. c) original frame and d) frame distorted through homography on plane Z=0.

4.2 Projection histograms and probabilistic templates

We start with a blob *B* as a cloud of Let 2D points contained into a bounding box with size *(Bx,By),* representing the points internal to the person silhouette. As in [4] we define horizontal and vertical projections (respectively π and θ) the following cardinality (indicated with #) of horizontal and vertical sub-sets:

$$\theta(x) = \#\left\{(x_p, y_p) \in B \middle| x_p = x\right\} \text{ where } x \in [0, Bx-1] \quad \textbf{(4)}$$

$$\pi(y) = \#\left\{(x_p, y_p) \in B \middle| y_p = y\right\} \text{ where } y \in [0, By-1] \quad \textbf{(5)}$$

Fig. 6 shows an example of histograms computed.

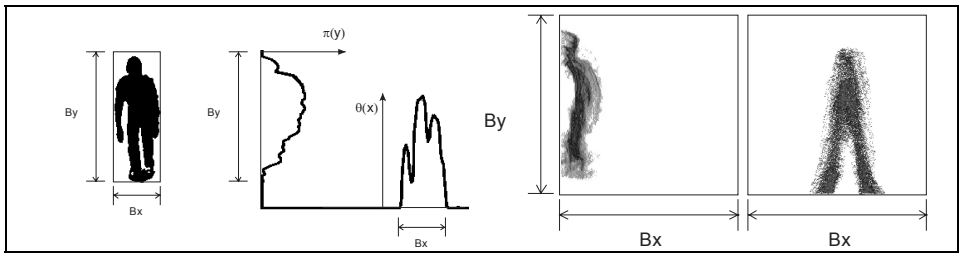

Fig. 6. Example of projection histograms and probability maps for *Standing_F* posture. Darker color correspond to higher probability

We defined a learning phase where the system constructs two probability maps (horizontal and vertical one) for every posture (class), using the respective silhouette projections. A training set of T_i 2D blobs referred to the *i*-th class is given for each *k* posture class: $B_t^i = \{(x,y), x \in [0, Bx_t^i - 1], y \in [0, By_t^i - 1]\}, t = 1..T_i$.

For each B_i^t the couple $P_i^t = (\theta_i^t, \pi_i^t)$ of projection histograms is computed as in equations (4) and (5). We construct the couple $\Theta_i(x,y)$ and $\Pi_i(x,y)$, (with $x \in [0, Bx-1]$, $y \in [0, By-1]$) of 2D *probability density maps* of the *i-th* state as follow:

$$\Theta_i(x,y) = \frac{1}{T_i}\sum_{t=1}^{T_i} g(\theta_i^t(x), y) \quad \Pi_i(x,y) = \frac{1}{T_i}\sum_{t=1}^{T_i} g(x, \pi_i^t(y)) \tag{6}$$

where

$$g(s,t) = \begin{cases} 1 & \text{if } s = t \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

Due to the different size of the silhouettes, Bx^i and By^i are not constant. As a consequence, the sizes (*Bx,By*) of $\Theta_i(x,y)$ and $\Pi_i(x,y)$ must be chosen to contain all the projection histograms P_i^t. Furthermore, $\Theta_i(x,y)$ and $\Pi_i(x,y)$ must be centred to the maps before using the equations (6). The density functions $\Theta_i(x,y)$ and $\Pi_i(x,y)$ describe a priori conditional probability: $\Theta_i(x,y)$ is the probability (conditionned to the fact of being in the status i) that the histogram θ has $\theta(x) = y$. That means, in other words, to have *y* points in the blob at the coordinate *x*. Similarly for $\Pi_i(x,y)$. Fig. 6 shows two probability maps computed for the posture *standing_F*.

4.3 Classification phase

In the classification phase the computed projection histograms are compared with the probability maps stored into the system. For each posture *i,* a measure of similarity S_i is extracted. In [4] the reference templates are histograms instead of maps, and S_i is computed with a logarithmic likelihood formula. By making that, all

the point of the template histograms have the same weight. Differently, with the adoption of the probability maps, the most reliable parts of the histograms are highlighted. Let $\tau_i = (\Theta_i, \Pi_i)$ be a probability map couple for the class i and $P = (\theta, \pi)$ the projection silhouette of the track to be classify. We consider the two similarity values S_i^θ and S_i^π obtained as:

$$S_i^\theta = \frac{1}{B_x}\sum_{x=0}^{B_x-1}\Theta_i\left(x,\theta(x)\right) \quad S_i^\pi = \frac{1}{B_y}\sum_{y=0}^{B_y-1}\Pi_i\left(\pi(y),y\right) \tag{8}$$

The final score S_i is computed as the correlation between the two scores and the posture estimator module selects as final posture the one with the maximum final score S_i.

$$S_i = S_i^\theta \cdot S_i^\pi \qquad P = posture(B) \Leftrightarrow p = \arg\max_{i=1..K}\{S_i\} \tag{9}$$

4.4 Dense Probabilistic Templates

The previous algorithm can be improved. If the training set is not large enough, the probability maps obtained could be sparse. As a consequence, also if the current histogram is very similar to those used during the learning phase, the similarity between histogram and template could be very low. The problem can be solved using a dense probabilistic template. To this aim, it is possible to use a "fuzzy" function, for example exchanging the definition of g in equation (7) with the one reported in equation (10).

$$g(s,t) = \frac{1}{|s-t|+1} \tag{10}$$

Each point of a map is increased according with its distance from the histogram. The number 1 at the denominator is inserted to avoid dividing by zero. To better understand refer to Fig. 7, showing the horizontal probability map obtained for the standing posture after 1, 2 and 20 training tracks.

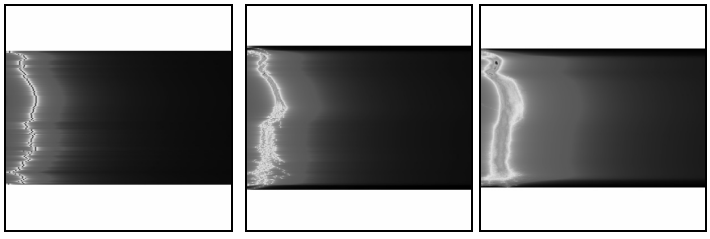

Fig. 7. Dense horizontal maps for the *Standing_F* posture after 1, 20 and 200 frames

5. Experimental results

The system has been designed to meet real-time constraints and to process a sufficient number of frame per second to be reactive and adaptive enough for possible alarm. Using standard workstations connected with a network camera we are able to process about 10 fps. Here we report some results on videos acquired in different contexts (Table 1). Similar tests have been proposed in [8] that describes a very preliminary phase of the project without calibration neither dense templates.

Luca1	**Luca2**	**Roberto1**	**Roberto2**
			
1209 frames	735 frames	294 frames	416 frames
320 x 240 pixel	320 x 240 pixel	360 x 288 pixel	384 x 288 pixel

Table 1. Frames from the videos adopted

In particular we describe three tests:

1. the *efficacy* in posture detection *over the same video where the training phase has been performed*: these results could be interesting for a domotic surveillance application, supposing that an initial training is done in the specific context on the specific people, as a sort of initial calibration of the system (Table 2) ;
2. the *efficacy* in posture detection by using a *different training set* obtained from the *same camera system* (Table 3, first row);
3. the *efficacy* and the generality of the model in posture detection *on different videos* (different camera, different scene, different actors) w.r.t. the training set (table Table 3, second row).

Video	Frames	Correct	Wrong	Efficacy %
Luca1	1209	1167	42	96,53%
Luca2	735	723	8	98,37%
Roberto1	294	289	1	98,30%
Roberto2	416	412	4	99,04%

Table 2. Efficacy rate for test 1

Video	Frame	Correct	Wrong	Efficacy %
Luca1	1209	1197	13	99,01%
Roberto2	416	387	29	93,03%

Table 3. Efficacy rate for tests 2 and 3; the template exploited in the test is obtained with Luca2

The system exhibits a quite robustness (about 90%) in every test. To better understand the main reasons of errors, we report in Table 4 the confusion matrix of

the posture estimator. Actually, the confusion matrix should be discussed on 12 postures (Eq. 1), but for brevity we have aggregated the result showing only the four main postures. Principally, the two mistaken postures are *standing* and *crawling*, because the transitions between them are very difficult to classify also for human observer.

Video: Luca1		Ground Truth			
		Standing	Sitting	Laying down	Crawling
Classifier	Standing	**1048**	0	0	0
	Sitting	0	**66**	0	1
	Laying down	0	0	**0**	0
	Crawling	39	2	0	**53**

Video: Luca2		Ground Truth			
		Standing	Sitting	Laying down	Crawling
Classifier	Standing	**349**	0	0	4
	Sitting	0	**55**	0	0
	Laying down	0	0	**100**	2
	Crawling	0	2	0	**219**

Video: Roberto1		Ground Truth			
		Standing	Sitting	Laying down	Crawling
Classifier	Standing	**113**	0	0	1
	Sitting	0	**43**	0	0
	Laying down	0	0	**93**	0
	Crawling	0	0	0	**40**

Video: Roberto2		Ground Truth			
		Standing	Sitting	Laying down	Crawling
Classifier	Standing	**122**	0	0	0
	Sitting	0	**0**	0	0
	Laying down	1	0	**219**	0
	Crawling	1	0	2	**71**

Table 4. The confusion matrixes obtained with test 1.

6. Conclusions and Acknowledgments

The paper discusses initial results of detecting human posture for surveillance and behavior monitoring in domotic applications. The discussed approach proved to be reliable and robust if the working constraints are satisfied.

In conclusion, in the project we have defined the universe of objects and events that are interesting for the application of autonomous domotic surveillance, and the system is able to detect in real-time. This kind of smart surveillance, improved by an initial learning phase is able to create a contact with the people in the house without the need of a remote human controller that will be involved only in case of dangerous situation. Only in this case the architecture of smart video server will be exploited to allow a full and low-cost connection and remote access to all available information. The system will be capable to communicate with audio output with the people in the home in order to have a feedback of the recognized situation. This can be the basis for

future paradigms of home-human interactions that will improve autonomy of people with some disabilities, increasing their safety and at the same time allowing programs of tele-rehabilitation.

This work is supported by the project "Domotics for disability" of "Fondazione Cassa di Risparmio di Modena" (Italy). The authors are also thankful to Luca Panini for his valuable help in ground-truth tests.

Bibliography

1. Cucchiara, R., Grana, C., Piccardi, M., Prati, A.: Detecting Moving Objects, Ghosts and Shadows in Video Streams. In press on IEEE Transactions on Pattern Analysis and Machine Intelligence.
2. Cucchiara, R., Grana, C., Prati, A.: Semantic Video Transcoding using Classes of Relevance. In International Journal of Image and Graphics, vol. 3, n. 1, January (2003) 145-169.
3. Fujiyoshi, H., Lipton, A.J.: Real-Time Human Motion Analysis by Image Skeletonization. In Fourth IEEE Workshop on Applications of Computer Vision, (1998).
4. Haritaoglu, I., Harwood, D., Davis, L.S.: Ghost: A Human Body Part Labeling System Using Silhouettes. In Proceedings of Fourteenth International Conference on Pattern Recognition, Brisbane, Aug (1998)
5. Haritaoglu, I., Harwood, D., Davis, L.S.: W^4: Real-Time Survelliance of People and Their Activities. IEEE Trans. On Pattern Analysis and Machine Intelligence,22(8) Aug (2000) 809-830.
6. Ju, S.X, Black, M.J., Yacob, Y.: Cardboard People: A Parameterized Model of Articulated Image Motion. In 2° International Conf. on Automatic Face & Gesture Recognition, (1996)
7. Moeslund, T.B., Granum, E.: A Survey of Computer Vision-Based Human Motion Capture. Computer Vision and Image Understanding, vol. 81, Elsevier Science Pubs., North Holland, (2001) 231-268.
8. Panini, L., Cucchiara, R.: A machine learning approach for human posture detection in domotics applications. In press on Proceedings of International Conference on Image Analysis and Processing (ICIAP 2003), Mantova, Italy, Sep. 17-19, (2003)
9. Senior, A., Hampapur, A., Tian, Y.-L., Brown, L., Pankanti, S., and Bolle, R.: Appearance models for occlusion Handling. In *Proceedings of Second International workshop on Performance Evaluation of Tracking and Surveillance systems,* (2001).
10. Wren, C.R., Azarbayejani, A., Darrel, T., Pentland, A.P.: Pfinder: Real-Time Tracking of the Human Body. IEEE Trans. On Pattern Analysis and Machine Intelligence,19(7), Jul (1997) 780-785.