Human Posture Recognition in Video Sequence

Bernard Boulay, François Bremond, Monique Thonnat

To cite this version:

Bernard Boulay, François Bremond, Monique Thonnat. Human Posture Recognition in Video Sequence. IEEE International Workshop on VS-PETS, Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2003, Nice, France. inria-00494249

HAL Id: inria-00494249
https://hal.inria.fr/inria-00494249

Submitted on 22 Jun 2010

Human Posture Recognition in Video Sequence

Bernard Boulay François Bremond Monique Thonnat

INRIA Sophia Antipolis, France
{Bernard.Boulay,Francois.Bremond,Monique.Thonnat}@sophia.inria.fr

Abstract

This paper presents a new approach to recognize human postures in video sequences comparing two methods. We first describe these two methods based on 2D appearances. The first one uses projections of moving pixels on the reference axis. The second method decomposes the human silhouette into blocks and learns 2D posture appearances through PCA. Then we use 3D model of posture to make the previous methods independent of the camera position. At the end we give some preliminary results and conclude on the effectiveness of this approach.

1. Introduction

This paper presents a new approach to recognize human postures. The recognition of posture is one step in the global process of analyzing human behavior. Behavior analysis is an important field dealing with many applications such as video surveillance or domotics. Usually, human behavior is recognized through the study of trajectories and positions of persons and using a priori knowledge about the scene (localization of doors, walls, areas of interest,...). This method is well adapted to a scene with large field of view observing the full trajectories of people, like in subway scene observed by top view cameras. But, when we consider a cluttered place where there is no continuous observation of people displacement like in office, we often do not have enough information to accurately determine behaviors. Recognizing posture then is a necessary step to recognize behaviors more accurately.

This work has been realized under two hypotheses. We want to recognize postures with only one static camera and in real time. These hypotheses are justified because in majority of applications, like video surveillance, only one static camera is used to observe the scene. The real time hypothesis is required in most applications. For example when we monitor a bank agency we want to trigger an alarm as soon as a hold-up is detected to prevent a crisis situation.

After a brief state of the art (section 2), we present (section 3) the video interpretation platform used to develop the posture recognition approach. We then describe the methods (section 4) and the results obtained (section 5). Finally we conclude on the generality of the approach (section 6).

2. State of Art

We classify existing methods in human posture recognition in two categories : methods based on 2D appearances and methods using 3D models.

2.1 2D Appearance-Based Methods

The majority of the methods based on 2D appearance use the same schema, [3], [4], [5] and [11]. First they detect the principal parts of the body such as head, hands and feet (the extremities of the body) and based on these detections they search for the secondary parts of the body such as shoulders, elbows and knees (the articulations).

The system Ghost, proposed by Haritaoglu et al, [3] segments the silhouette from the background. It computes the vertical and horizontal projections of the silhouette to determine the global posture of the person (standing, sitting, crawling-bending and laying) and his orientation relative to the camera (front view, left side view and right side view). To recognize posture the system computes the projections of the current silhouette and compares them to the model of projection realized for a set of predefined posture and point of view. Then it determines the body parts by analyzing the contour of the silhouette.

Iwasawa et al [5] have proposed a method in three steps. The first step consists in determining the center of gravity of the human silhouette. The second step computes the orientation of the upper half of the body. Then the significant points such as feet, hands, elbows, and knees are estimated by using a heuristic contour analysis of the human silhouette.

Pfinder [11] is a real time system which uses a multi-class statistical model of color and shape to obtain a 2D representation of head and hands in a wide range of viewing conditions.

Bobick and Davis [1] use temporal template build on a set of several consecutive frames. The method interprets human motion in an image sequence by using motion energy images (MEI) and motion history images (MHI). The motion images are calculated via subtracting successive frames thresholded into binary values. These motion images are accumulated in time and form MEI. The MEI are enhanced into MHI where each pixel value is proportional to the duration of motion at that position. Moment based features are extracted from MEI and MHI and employed for recognition using template matching.

The principal drawback of these methods is the dependency on the point of view. Indeed, if we consider a seated person, according to the position of the camera the person appears different on the screen. However these methods are not resource demanded (use only one camera) and they are adapted for real time problems.

2.2 3D Model-Based Methods

A 3D model is made of geometrical objects such as parallelepipeds, spheres or truncated cones. The model includes the parameters which define the relations between these objects. This model is defined in a high dimensional phase space (dimension depends on the degrees of freedom of the model).

Systems which use monocular vision are based on a priori knowledge in the form of a human model and the constraints related to it.

In [7], Moeslund et al propose an alternative representation of the phase space which supports a more efficient use of the different constraints. This method needs a frontoparallel torso with respect to the camera. Moreover to deal with real time constraint, one arm is processed.

To handle ambiguities in posture estimation and to estimate precisely the depth, several cameras may be used.

In [6], three cameras are used to observe the person from the top, front and side view. In each image, the significant points (head, hands, feet, ...) are located in 2D. Then two views are selected and the 3D coordinates of each significant point are calculated by triangulation.

In [2], Delamarre and Faugeras use three cameras all around the person. The method compares the projections of a 3D model of a person on an image with the detected silhouettes of the person (binary image). This process is iterated by computing a force that will move the 3D model towards the current silhouette. The final parameters of the 3D model constitute an estimation of the real posture.

In [9], Shimada et al propose a system estimating 3D human hand posture. The estimation is based on a 2D image retrieval. More than 16000 hand appearances was generated and stored in a database. The search area is reduced by using an adjacency map in the database. This method is implemented on a PC cluster system consisting of 6 PCs (Pentium III 600 MHz) to achieve real time. The use of these PCs is not adapted for video surveillance applications.

Tao Zhao et al [12] use a human 3D model to verify if the moving region detected is a person. The verification is done by walking recognition using an articulated human walking model.

Except for the two last, these methods have two drawbacks. The first drawback is the utilization of many parameters difficult to tune. A second drawback is the processing time. To recognize the posture in real time they just process some parts of the body. Moreover the majority of these methods use several cameras. However, since they use a 3D model they are independent on point of view. Indeed, if we consider a seated person, a 3D model gives a good projection according to the position of the camera.

3. Video Interpretation Platform

To determine the posture of a person in a video, we must first detect the person in this video. For this we use an interpretation platform VSIP (Visual Surveillance Intelligent Platform). This platform is able to detect moving pixels in a video. The detection is made by subtracting the current image with the reference image. The reference image is the empty scene and it is also called background image. After the difference, the image is thresholded to obtain a binary image. The moving pixels are grouped into connected regions which are called blobs. These blobs are classified into predefined classes (vehicles, persons for example) and they are tracked all along the video. The final step consists of recognizing the behaviors related to these tracked mobile objects. On each frame, to recognize postures we use the mobile objects detected and classified as person.

4. Methods

In the next section, we describe two methods based on 2D appearances. We recognize postures on the blobs (the binary images). First we have a learning phase which is made on several videos with different actors. We then compare the blob we study with the data obtained by the learning phase. Moreover, we describe an approach which uses 3D human model to make the previous methods independent of the camera position.

4.1 Chosen Postures

First we need to determine which postures we want to recognize. We have selected postures of every day work in office and easily to be recognized. The chosen postures can be classified into three categories : the standing postures (cf. figure 3) (standing with arms near the body, standing with arm to left, standing with arm to right and T-shape), the seated postures (cf. figure 4) (seated on a chair and seated on the floor) and the bending posture (cf. figure 5) (a person who tries to pick up something on the floor).

4.2 Projections

The first method uses horizontal and vertical (H. & V.) projections on the references axis on the image. The vertical (resp. horizontal) projection is obtained by counting the number of pixels of each column (resp. row) of the blob we study, which correspond to the person. We then normalize the projection by giving an arbitrary length to it.

The learning phase is made on different videos with different actors. For each detected blob, we compute its projections and we normalize them. We then compute the average vertical and horizontal projections of each posture. Then a posture is represented by a couple of projections.

To recognize the posture of a person in a blob, we compute its projections on the reference axis. We normalize them and compare them with the average projections using a SSD (Sum of Squared Differences).

$$S_i = \log\left(\sum_h \left(\hat{H}_h^i - H_h\right)^2 + \sum_v \left(\hat{V}_v^i - V_v\right)^2\right) \quad (1)$$

where $\hat{H}^i$ (resp. $\hat{V}^i$) is the average horizontal (resp. vertical) projections of the posture i and H (resp. V) is the horizontal (resp. vertical) projection of the blob we study.

4.3 Block Density

Another method consists in decomposing a given blob into blocks and to characterize its posture by the density of motion pixels inside each block. The characterization of the posture is based on PCA (Principal Component Analysis). PCA (or Karhunenn Loeve transform) is usually used in different applications like character recognition or particularly face recognition (eigenfaces). A drawback of PCA is the number of data it is used : for an image 100 by 100, a vector of 10000 numbers is used. Here we try to characterize a blob by decrease the length of the vector with a very simple idea. Since our aim is not representation or compression, that is to say with the vector we do not have to find the original blob, we will cut the blob in smaller blocks.

We compute the ratio of the blob corresponding to the ratio between the height and the width of the bounding box of the blob. So we can classify blob into three types : stretched blob, laid blob and squared blob. The method is able to classify blob in only one, two or three types. Then according to the type of the blob we normalize it by giving a new height and a new width to the bounding box of the blob. If the blob is a stretched blob the new height will be bigger than the new width. Whereas if the blob is a squared blob the new height and the new width will be equal. It helps us for the next stage. According to the type of the blob we decompose the blob in rectangular blocks. If the blob is a stretched blob we consider more blocks on the height than on the width. We associate to each block the density (percentage) of motion pixels which belongs to the person. With these values we compute a vector. The first values of the vectors are the values of the first line of blocks. So a blob is characterized by a type and a vector.

The learning phase is made on different videos with different actors. For each detected blob, we compute its type and its vector. So we have three classes of vectors : one for each type. Now we compute a PCA on each class. If we suppose there are M vectors Φ_i on a class : we compute the average vector Ψ :

$$\Psi = \frac{1}{M}\sum_{i=1}^{M} \Phi_i \quad (2)$$

We center the vectors :

$$\Gamma_i = \Phi_i - \Psi \quad (3)$$

We have the matrix A of centered vectors :

$$A = [\Gamma_1, \Gamma_2, \ldots, \Gamma_M] \quad (4)$$

We compute the covariance matrix C :

$$C = AA^T \quad (5)$$

We compute the eigen values μ_j and the eigen vectors v_j of the matrix C. We keep the K biggest eigen values and consider the base B with the K eigen vectors. To choose the value K, we compute the ratio :

$$\frac{\sum_{i=1}^{K} \mu_i}{\sum_{i=1}^{N} \mu_i} \quad (6)$$

where K is the number of eigen vectors we keep, N is the number of the components of vectors Γ_i and μ_i the eigen values. We choose K such that the ratio (equation 6) is superior to 90%, that means 90% of the information is kept. We project all the M vectors of the class on the base B :

$$\omega_k = v_k.\Gamma_i \text{ with } \omega_k \text{ the } k^{th} \text{ weight} \quad (7)$$
$$\text{and } 1 \leq k \leq K \text{ and } 1 \leq i \leq M$$

and we obtain M vectors with K values. In these vectors, there are several vectors which represent a same posture because different actors can have the same posture and a same actor in a same posture can be detected by blobs with different type of shape. So we compute the average vector for each posture. A posture is represented by an average vector.

To recognize the posture of a person in a blob, we compute its type and its vector. We project the vector on the base B and obtain the new vector P. Then according to the type we compare the vector with the average vectors by using Mahanalobis distance :

$$\sqrt{\sum_j \frac{1}{\mu_j}\left(\hat{P}_j^i - P_j\right)^2} \quad (8)$$

where μ_j are the eigen values, $\hat{P^i}$ is the average vector of posture i and P is the vector which characterizes the studied blob. As each axis has not the same significance components corresponding to smaller eigen values are weighted more heavily since our aim is discrimination rather than representation.

4.4 3D Model

To solve the problem of the dependence on the camera view point we propose to use a 3D model of a human being. This model is composed of three types of geometrical objects : parallelepipeds (4), spheres (9), truncated cones (10). The posture of the human model is defined by a set of 111 parameters corresponding to the position and orientation of all these geometrical objects. We have defined for each posture of interest a specific set of parameters. For example, we can see on figure 6 the 3D human model in a seated posture .

For each detected person, we first compute the position and orientation of this person in the 3D world. The 3D position is computed with VSIP which have different informations about the scene, in particular about the parameters of the camera. The orientation is based on the walk forward hypothesis : the previous and the current position of the person provide us the face of the person. It is a preliminary method, which must be improved to work on the most cases, in particularly when the person do not walk. Second, we project the 3D human model for each predefined posture in the image according to the position and orientation of the detected person. In figure 6, we can see the projection of the 3D human model in a seated posture. Finally we compare these projections of the 3D human model with the blob corresponding to the detected person. The comparison can be done either by both previous 2D methods (projections and block density).

We adapt the projection method by replacing the learn projection models by the horizontal and vertical (H. & V.) projections on the reference axis of the 3D model for each posture. The comparison between these (H. & V.) projections of the 3D model with the (H. & V.) projections of the detected blob has been realized by the SSD similarly to the 2D method. The block density method cannot also be used directly because this method is too expensive in processing time. The computation of the PCA matrix needs to be recomputed for each detected person. The block density method has been adapted to the 3D human model by using directly the vectors obtained by the projection of the 3D model on the image for each posture. These vectors constitute a new reference base. The comparison between this vector base $\{B^i\}$ and the vector corresponding to the detected person P has been done with an Euclidean distance :

$$d^i = \sqrt{\sum_j \left(P_j - B^i_j\right)^2} \tag{9}$$

where d^i is the distance between the i^{th} vector, which represents the i^{th}posture and the vector corresponding to the detected person.

Postures	Number of blobs	Number of blobs correctly detected	Success percentage
Standing	451	347	76
Standing with arm to left	106	96	90
Standing with arm to right	110	95	86
T-shape	79	49	62
Sitting on a chair	85	57	67
Sitting on the floor	92	66	71
Bending	189	142	75
Total	1112	852	76

Table 1: Recognition of chosen postures with the (H. & V.) projections

5. Results

The results are computed with several videos where the actors are the same than in the learning phase on a PC (Pentium III 866 MHz, 256 MO, Linux).

5.1 Projections

The (H. & V.) projections for each posture (7 postures) are stored. The time for the matching (i.e. the time to compute (≈ 1 ms), to normalize (≈ 80 μs) and to compare (≈ 330 μs) the projections of the blob we studied with the stored projections) is 1 ms and 410 μs by blob.

We can see in the table 1 the success percentage of recognition for each chosen posture. These results are obtained for a normalization of 128, that means the projections have 128 values. The total rate of recognition is 76%. The lower rate is for the postures T-shape (62%) and seated on a chair (67%). The bad result for posture T-shape is due to the lean of the person in the blob which is not centered implying a bias for the vertical projections. A better normalization will be made by aligning the median coordinate of the silhouette at the center to increase the results. Moreover, the average

Normalization length	Rate of Recognition (%)
8	70
16	74
32	74
64	76
128	76
256	76

Table 2: Normalization length of the projections

projections of the postures seated on a chair and seated on the floor are very similar so these postures are often mixed. To see the influence of the normalization on the results, we use different normalizations (table 2). If we use a length of 64 or above we have the best results. The same result is obtained in the case of using a superior length because we have enough information. If we use a length inferior to 64 the results decrease because we loose information.

5.2 Block Density

For each type of blob, the average vector (with K values) of each posture and the base (the K eigen vectors (with M values) and the K eigen values) are stored. The time for matching (i.e. the time to compute the type and the vector representing the blob (≈ 3 ms), to project the vector on the new base (≈ 250 μs) and to compare it with the average vectors (≈ 1 ms 500 μs)) is 4 ms and 750 μs by blob.
We can see in the table 3 the success percentage of recognition for each chosen posture. These results are obtained for the parameters in table 4. The total rate of recognition is 80%. The lower rate is for the T-shape posture (56%). We can see on fig. 2 a person in T-shape posture who is wrongly recognized. Indeed the point of view is different from the point of view of the learning phase (front of the camera). To see the influence of the number of type considered, we compute results for different classification (table 5). We can have one, two or three classes (different type of blob). We notice that the best result is obtained when only the two classes stretched and squared are considered. This situation gives best results, because we do not have laid postures like person laying on the floor.

5.3 3D Model

We use 3D human model to improve cases where the 2D methods do not give satisfying results (an example of wrong recognition for 2D methods is shown in fig. 2). First for each case of wrong recognition, the posture parameters of the 3D model are defined off line and applied. The result for T-shape posture with good orientation is shown in figure 8 for the previous example (the position is computed with VSIP and orientation is computed manually). Second

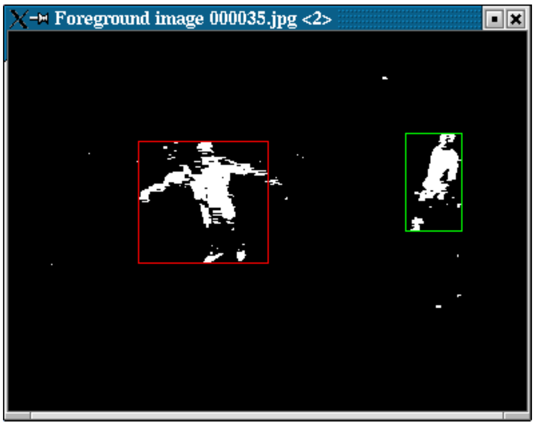

Figure 1: Foreground image

Postures	Number of blobs	Number of blobs correctly detected	Success percentage
Standing	451	378	83
Standing with arm to left	106	87	82
Standing with arm to right	110	78	70
T-shape	79	45	56
Sitting on a chair	85	75	88
Sitting on the floor	92	85	92
Bending	189	143	75
Total	1112	891	80

Table 3: Recognition of chosen postures with the block density method

Blob type $\tau = 1$	Normalization	cutting	Number of eigen vectors kept (K)
Squared	100 X 100	10 X 10	31
Laid	75 X 160	5 X 20	14
Stretched	160 X 75	20 X 5	37

Table 4: Parameters use with the block density method

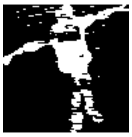

Figure 2: T-shape posture not recognized

Number of class	Rate of Recognition(%)
1 (sq)	70
2 (st,l)	69
2 (sq,st)	80
3 (sq,st,l)	72

Table 5: Rate of recognition for different classification (sq=squared, st=stretched, l=laid)

3D Human Posture	SSD
T-shape posture with correct orientation	12.31
T-shape posture with wrong orientation	13.20
Sitting on a chair	13.81
Standing with arm to left	14.15
Standing	14.27
Standing with arm to right	14.96
Bending	15.50

Table 6: Comparison between (H. & V.) projections of 3D human models and (H. & V.) projections of posture in fig.2

the projection of the 3D model on the image is computed (fig. 8) for each posture. Then the (H. & V.) projections on the reference axis of each blob corresponding to the predefined postures are compared (using the SSD defined in section 4.2) with the (H. & V.) projections of the current blob corresponding to the detected person. On table 6 we can see that the 3D model of the T-shape posture with correct orientation give the best result. We also use the block density method by representing the blobs by vectors and compare them with the Euclidean distance. The T-shape posture in figure 8 gives the best result similarly to the (H. & V.) projection method. Because the projection method requires less processing time than the block density method we propose to only use the first method.

6. Conclusions and Perspectives

We have described our work on human posture recognition in video sequence. Our approach improves traditional 2D methods ((H. & V.) projection and block density methods) by using a 3D model to be independent on the camera view point. 2D methods depend on the camera point of view. When the learning phase is made for a certain position of the camera, the methods give good results if the camera is about the same position. The 3D model improves results in the case where the 2D methods do not work correctly. Previous work has shown the interest of using a 3D model to recognize posture of a walking person [12]. This paper is an attempt to generalize the utilization of 3D model to recognize any type of postures.

This work can be improved in the following aspects. First, we need to compute automatically the orientation of the persons who do not walk. Second, we plan to determine primitive postures which we can combine to form all the postures we want to recognize. Indeed, if we consider only some key postures, intermediate postures will be badly recognized. A person with arms near the body does not have immediate transition to be in the T-shape posture. To use these primitive postures frame by frame tracking information becomes important. We can use information about the posture of a person in a frame to determine the posture of the same person in the next frame. On two consecutive frames the postures of the same person are not independent.

References

[1] A. F. Bobick and J. W. Davis, "The Recognition of Human Movement Using Temporal Templates," *IEEE Transactions on pattern analysis and machine intelligence*, Vol. 23, no 3, march 2001.

[2] Q. Delamarre and O. Faugeras, "3D Articulated Models and Multi-View Tracking with Silhouettes," *RobotVis Project INRIA*, 2001.

[3] I. Haritaoglu, D. Harwood and L. S. Davis, "Ghost : A Human Body Part Labeling System Using Silhouettes," *14th International Conference on Pattern Recognition*, 1998, Brisbane, Australia.

[4] I. Haritaoglu, D. Harwood and L. S. Davis, "W^4 : Who? When? Where? What? A Real time System for Detecting and Tracking People," *3. International Conference on Face and Gesture Recognition*, 1998, Nara, Japan.

[5] S. Iwasawa, K. Ebihara, J. Ohya and S. Morishima, "Real-Time Estimation of Human Body Posture from Monocular Thermal Images," *Conference on Computer Vision and Pattern Recognition*, 1997.

[6] S. Iwasawa, J. Ohya, K. Takahashi, T. Sakaguchi, S.Kawato, K. Ebihara and S. Morishima, "Real-time, 3D Estimation of Human Body Postures from Trinocular Images,' *Faculty of engineering, Seikei University*, 2000.

[7] T. B. Moeslund and E. Granum, "3D Human Pose Estimation using 2D Data and an Alternative Phase Space representation," *Procedure Humans 2000*, 2000, Hilton Head Island, South Carolina.

[8] N. Rota and M. Thonnat, "Video Sequence Interpretation for Visual Surveillance," *3rd IEEE International Workshop on Visual Surveillance*, 2000, Dublin, Ireland.

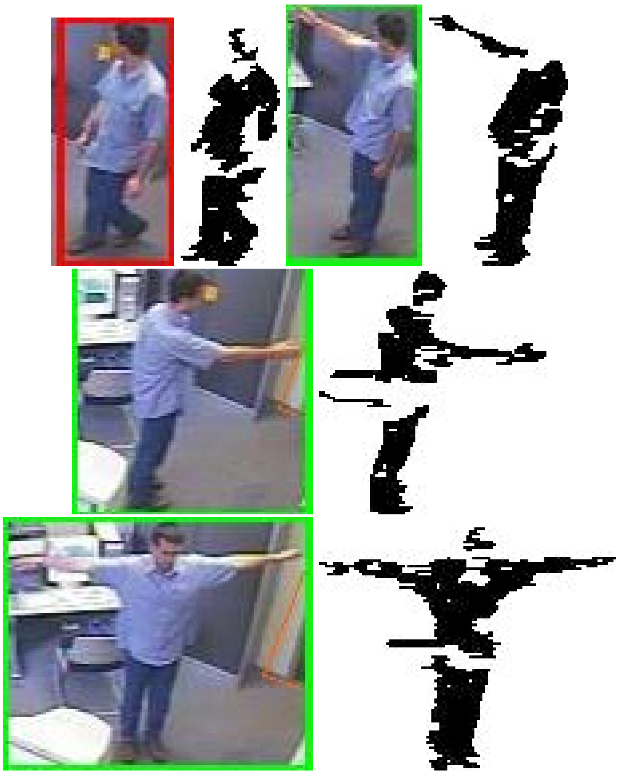

Figure 3: The standing postures and their blobs

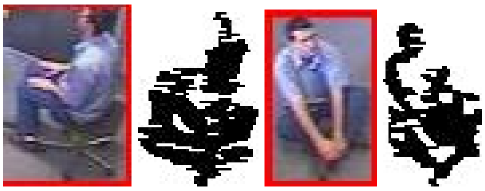

Figure 4: The seated postures and their blobs

[9] N. Shimada, K. Kimura and Y. Shirai, “Real-time 3D Hand Posture Estimation based on 2D Appearance Retrieval Using Monocular Camera,” *Proc. Int. WS on RATFG-RTS*, pp. 23-30, 2001.

[10] M. Thonnat and N. Rota, “Image Understanding for Visual Surveillance Applications,” *3rd IWCDV*, 1999, Kyoto, Japan.

[11] C. Wren, A. Azarbayejani, T. Darrell and A. Pentland, “Pfinder : Real-Time Tracking of the Human Body,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, no 7, pp. 780-785, 1997.

[12] T. Zhao, R. Nevatia and F. Lv, “Segmentation and Tracking of Multiple Humans in Complex Situations,” *Computer Vision and Pattern Recognition 2001*, 2001.

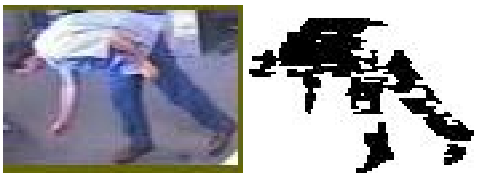

Figure 5: The bending posture and its blob

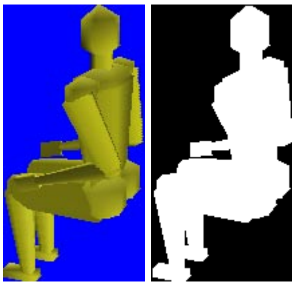

Figure 6: Seated posture of 3D model and its blob

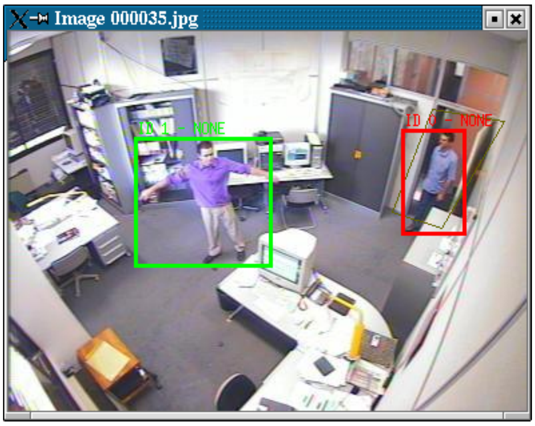

Figure 7: Current image

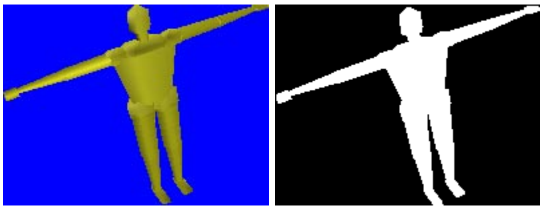

Figure 8: T-shape posture of 3D model and its blob