Clustering of Human Actions using Invariant Body Shape Descriptor and Dynamic Time Warping

Massimiliano Pierobon, Marco Marcon, Augusto Sarti and Stefano Tubaro
Image and Sound Processing Group
Dipartimento di Elettronica e Informazione - Politecnico di Milano
Piazza Leonardo da Vinci 32, 20133 Milano, Italy
Email: massimiliano.pierobon@poste.it, marcon@elet.polimi.it, sarti/tubaro@elet.polimi.it

Abstract

We propose a human action clustering method based on a 3D representation of the body in terms of volumetric coordinates. Features representing body postures are extracted directly from 3D data*, making the system inherently insensitive to viewpoint dependence, motion ambiguities and self-occlusions. An* Invariant Shape Descriptor *of human body is obtained in order to capture only posture-dependent characteristics, despite possible differences in translation, orientation, scale and body size. Frame-by-frame descriptions, generated from a gesture sequence, are collected together in matrices. Clustering of action matrices is eventually performed, and through a* Dynamic Time Warping *(while computing the distance metric), we gain independence from possible temporal nonlinear distortions among different instances of the same gesture.*

1. Introduction

Systems that are able to recognize human gestures and actions, without any invasive device, have recently raised a great deal of interest not only in the research community, but also for industrial applications. All these techniques could have direct applications to video surveillance problems [1], human-computer gestural interaction projects, robot skill learning and to all fields in which activity recognition is needed.

Multi-camera systems are considered nowadays among the most promising techniques used in computer vision. 3D reconstructions derived from different views are inherently able to solve ambiguities and viewpoint-dependencies, which are unavoidable in systems based on monocular views (see [2] and [3]). In this paper we consider volumetric 3D reconstruction of a moving human body, in terms of voxel occupancy in an assigned voxelset. This is the starting point for developing a reliable set of features representing an actor performing a natural gesture.

A good selection of salient features from voxels coordinates of the "actor's body" has a great importance for the overall performance of the recognition system. In order to succeed in obtaining a robust representation of an action, we developed a feature extraction method similar to [4], based on a spherical *Shape Descriptor* obtained from a sampled shape function, a cylinder, adapted on the fly to the size of the body. Features are invariant to scale, translation and rotation and constitute a meaningful representations of body postures. These features vary continuously with body motions.

The recognition stage is then performed through a clustering of different instances of gestures formed by a collection of shape distributions, one for each considered time instance. Distance metric between sequences of features is computed through the use of *Dynamic Time Warping*, a method that accounts for possible nonlinear distortions in action delivery speed.

1.1. Previous Work

In the past few years a great deal of research has been done in the field of activity recognition with 3D data, for examples see [5] and [6]. Major effort has been put into the research of invariant features with respect to viewpoint and trajectory variations (see [7]).

Another interesting direction, aimed at discovering synthetical and meaningful representations of human postures, has been taken by researchers. The main idea beneath these studies is to find similarities between actions and speech recognition, considering postures as the atoms of gestures in the same way as phonemes are often considered as the bricks that form words. In the field of posture estimation we can find methods concerning the use of body part displacement coordinates (see [8]), or regarding the computation of a global body shape descriptor (see [4]). We considered the last one as a starting point for our feature selection.

The classifier design is an important part in recognition system projects, but it cannot be considered separately from the evaluation of features. Many recognition methods have been proposed, most of them based on HMMs (Hidden

0-7803-9385-6/05/$20.00 ©2005 IEEE

Markov Models) theory, such as in [8], [9] and in [10]. At the moment we decided to adopt a simpler recognition algorithm, which is more computationally efficient and is able to exploit the discrimination properties of our features. *Dynamic Time Warping* (DTW), even if it presents some limitations (see [11]), is an efficient way to compute a distance metric between two sequences of same postures performed in the same order but with different speeds. References on DTW method can be found in [11] and in [12].

2. Data Acquisition and Feature Extraction

2.1. 3D Volumetric Reconstruction

In order to have a 3D reconstruction of the moving body into the scene, we apply the so called *Volumetric Intersection* method (see [13], [14]). Starting from eight different viewpoints, represented by eight synchronized cameras, we compute, frame by frame, the extraction of body silhouettes using a *Chroma Keying* algorithm [15]. Then, in a virtual 3D environment, we build the generalized cones starting from the optical center of each camera and intercepting each respective silhouette. The volumetric intersection of these cones, called *Visual Hull*, approximates the 3D reconstruction of the actor and, sampling its convolution with a smoothing filter, can be transformed in a 3D representation compound of voxels coordinates (Fig. 1).

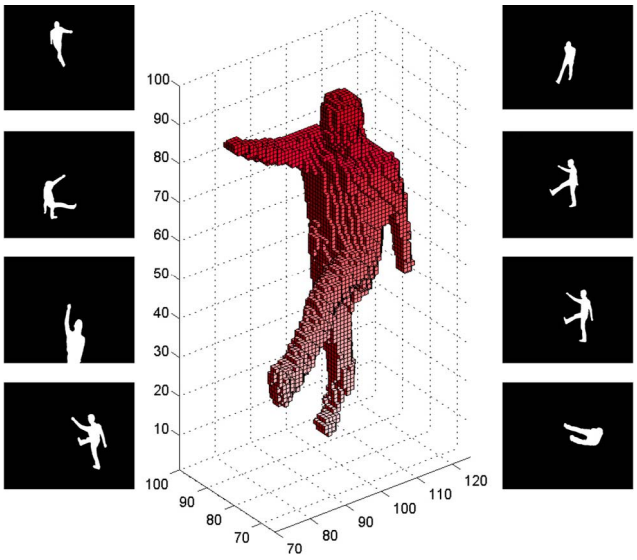

Figure 1: *Volumetric intersection. Example of voxelsets creation by 3D intersection of Visual Hulls projected from segmented edges. This method is performed for each frame of a gesture action sequence.*

2.2. Requisites of Features Representing Human Body Postures

The voxelset occupancy represents a redundant description of a body posture since it contains information like body position, orientation and size of the actor into the scene. Our target is to obtain a description of a body with a good generalization power, but at the same time with a great discriminatory capabilities in terms of different postures. Besides, our representation should vary continuously as the gesture evolves. The last property is essential not only using DTW and *Dynamic Programming*, as explained later, but also applying an HMM action modeling approach, which is a natural extension of our system built so far.

2.3. Invariant Shape Descriptor Method

The set of features that we extract is an extension of the *Shape Descriptor* explained in [4], already used to infer a body posture in a static environment. In this work we propose an adaptation of the method to a dynamic context: a collection of postures across time.

Let us describe the general *Shape Descriptor* applied to a volumetric voxelset:

- *Shape Descriptor* describes a 3D volumetric object with regard to a *reference shape*, Θ: normally a surface like a cylinder or a sphere is used.
- The surface of the reference shape is sampled regularly in a sufficient number N of points, called *control points*, according to some empiric criteria.
- For each control point, P_n:
  - Each voxel is encoded in a spherical frame of reference centered in P_n with dimensions ρ (from 0 to a suitable value), θ (from 0 to π rad) and φ (from 0 to 2π rad).
  - Each polar coordinate is uniformly sampled into ten parts, obtaining a set $\{(\rho_i, \theta_j, \varphi_k) : 0 \leq i, j, k \leq 9\}$ of 1000 elements.
  - For each volume in spherical coordinates, defined by a particular $(\rho_i, \theta_j, \varphi_k)$, we count the voxels contained and build a *spherical histogram* $f_n(i, j, k)$ containing these values (for more details see [4]).
- A spherical *Shape Descriptor* $F(i, j, k)$ is computed summing up all the corresponding values in the histograms of the control points and normalizing all to the maximum value:

$$F(i,j,k) = \sum_{n=1}^{N} \frac{f_n(i,j,k)}{\max_{\bar{i},\bar{j},\bar{k}} \left(\sum_{l=1}^{N} f_l\left(\bar{i},\bar{j},\bar{k}\right)\right)}$$

Obviously reference shapes chosen among simple 3D surfaces comply with general geometrical properties without filtering out peculiar information about the described object. For example, using a sphere centered in the body centroid with a radius that is proportional to body's main direction, we obtain a description of the body shape with complete loss of information about position in space, actor's height and 3D orientation. In our project we use, as suggested in [4], a cylinder with the axis crossing the centroid, vertically oriented and fitting the body's height. In our approach, instead of inscribing the body inside the reference shape, we optimized the cylinder radius using a suitable value. The used value is the radius of the major circle inscribed inside the projection of the entire voxelset on the floor (Fig. 2 right). This way we obtain a representation that is independent from position, size, scale, body proportions and, possibly, invariant to rotations on its own axis. We call it *Invariant Body Shape Descriptor*. The main idea beneath these choices is that gestural instance-dependent information tends to be filtered out by these adaptive reference shape (Fig. 2 left), while important data on body postures are captured by the *Shape Descriptor* algorithm, computed on sort of a normalized object.

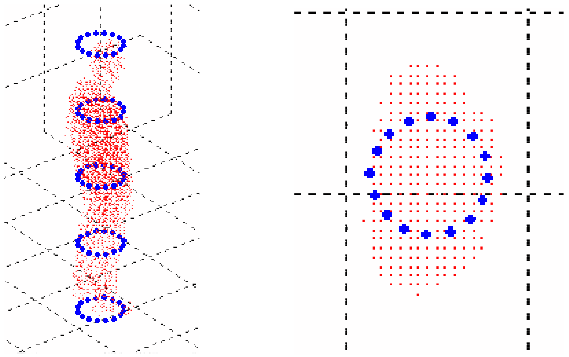

Figure 2: *Cylindrical Reference Shape.* Left*: Example of a cylindrical reference shape adapted to body proportions. The voxelset is here sub-sampled by a rate of 4 and each voxel is represented only with its center in order to make internal points visible.* Right*: Cylinder radius is adapted to the major circle inscribed inside planar projection.*

We would like to point out an important aspect that confirms the rotational invariance of the shape descriptor: for each polar reference frame, centered in its respective control point, we assume as zero-elevation and zero-azimuth the direction of the segment lying on the horizontal plane (zero-elevation) projecting the control point on the cylinder axis.

The final normalization computed in the *Shape Descriptor* algorithm is important in order to create a representation that is independent of the voxel size and, quite obviously, of the different proportions that the reference shape and, consequently, the derived volumes in spherical coordinates could have.

An example of *Invariant Body Shape Descriptor* can be seen in Fig. 3.

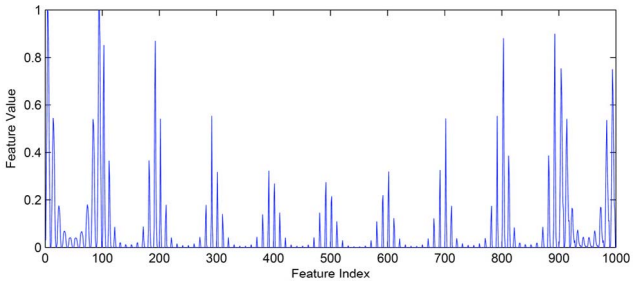

Figure 3: *Invariant Body Shape Descriptor. Example of an Invariant Body Shape Descriptor computed on the voxelset shown in Fig. 2. This is a vector filled scanning the spherical coordinates of the shape descriptor in this order: for every* θ_k *{for every* φ_j *{for every* ρ_i *{read* $F(\rho_i, \varphi_j, \theta_k)$*}}}.*

In a dynamic context problems arise while readapting the reference shape at each frame of the sequence: these are mainly due to our constraint of obtaining features that vary continuously throughout the motion. In fact slight variations of the reference cylinder may cause heavy temporal discontinuities in the obtained features. We fixed this problem by using the cylinder computed in the first frame for the entire duration of a sequence. The cylinder follows the motion of the body's centroid but its size remains unchanged.

Following the described method, we compute an *Invariant Body Shape Descriptor* for each frame and the collection of these 1000×1 vectors throughout a sequence is the data set that we use to represent a gesture (six examples are shown in Fig. 5).

3. Action Clustering Stage

In order to evaluate the discriminatory abilities of the extracted features we use one of the simplest template matching algorithms. The *DTW* is a definition of a distance metric for measuring similarity between a known reference pattern and a test pattern. This method accounts for the non-linear distortions that could affect two sequences of features. If we take two gestures, a direct comparison between two feature vectors at a given time is clearly impossible: this is mainly due to the different duration of the gesture's steps. It follows that the whole action length has to be considered (Fig. 5). Through DTW we are able to find optimal correspondences between feature vectors of different matrices according to

an agreed cost function. In other words, we can compare sequences of similar body postures in two actions independently from their time index.

DTW is based on the *Dynamic Programming* theory. If we have a reference pattern, say $r_i, i = 0, \cdots, I$, and a test pattern $t_j, j = 0, \cdots, J$, where, in the general situation, $I \neq J$, we can find a distance measure between the two sequences building a 2D grid with points on respective axis assigned to their feature vectors. Each node (i, j) is associated with a specific value of a cost function $c(i, j)$ measuring the "distance" between the respective elements of the strings, r_i and t_j. We are now looking for a path through the grid from an initial node (i_0, j_0) to a final one (i_F, j_F) that minimize the overall cost C defined as:

$$C = \sum_{k=0}^{F} c(i_k, j_k)$$

In order to obtain the optimal path with the overall minimum cost, according to the *Bellman's Optimality Principle* [11], the overall optimal path from (i_0, j_0) to (i_F, j_F) through (i, j) is the concatenation of the optimal path from (i_0, j_0) to (i, j) and the optimal path from (i, j) to (i_F, j_F). In other words, when we have the optimal path from the beginning to a certain point, we only need to search for the optimal path starting from this point in order to reach the final node in an optimal fashion. For each node of the grid (i_k, j_k), though, we only have to find a node (i_{k-1}, j_{k-1}), from a list of possible predecessor, that leads to minimum cost:

$$C_{min}(i_k, j_k) = \min_{i_{k-1}, j_{k-1}} [C_{min}(i_{k-1}, j_{k-1}) + \\ + c(i_k, j_k | i_{k-1}, j_{k-1})]$$

Using this formula we can compute the so-called Minimum Distance Grid (Fig. 4-left), in which every node is now associated to the minimum cost from the initial node. This matrix is computed incrementally in such a way that its node (i_F, j_F) contains the minimum cost $C_{min}(i_F, j_F)$ to reach the final node starting from the initial one, (i_0, j_0). Besides, we can take into account each optimal predecessor for each node of the grid in order to be able to construct the optimal path backtracking from (i_F, j_F) (Fig. 4-right).

In this work we consider:

$$(i_0, j_0) = (0, 0) \qquad (i_F, j_F) = (I, J)$$

which means that we are searching for the optimal path from the initial node to the node corresponding to final feature vectors of both sequences. Note that each sequence is composed of an isolated instance of a single action. Moreover we assume:

$$c(i_k, j_k | i_{k-1}, j_{k-1}) = c(i_k, j_k) = L_2(i_k - j_k)$$

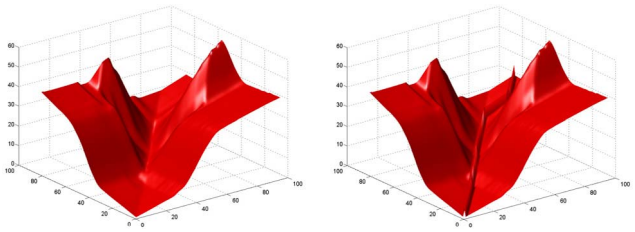

Figure 4: *Dynamic Time Warping. Left: Minimum Distance grid computed using* Bellman's Optimally Principle *between the two "KICK" sequences of Fig. 5. Right: the same grid with the overall optimal path (the one across the valley from* $(0, 0)$ *to* (I, J)*)*

considering the cost function not associated with a specific transition to the node from a predecessor. Instead, we compute only a Euclidian Norm of the distance between the two corresponding feature vectors. In other words, we allow all possible transitions from a node predecessor without any extra cost. We define a predecessor in this way:

$$(i_{k-1}, j_{k-1}) = \begin{cases} (i_k - 1, j_k) \\ (i_k - 1, j_k - 1) \\ (i_k, j_k - 1) \end{cases}$$

therefore there is no limit in the rate of expansion/compression, as successive horizontal or vertical transitions could occur.

4. Experimental Results

We tested the system with different instances, performed differently by the same person or by another one, of three simple actions: "POINTING AT", "CROUCHING DOWN" and "KICK". With the word "simple" we refer to actions that are not repeated for a random number of times, therefore different instances must contain corresponding feature vectors. For each gesture we collected at least two different realizations. This constraint avoids problems due to the low-level comparison made by the DTW. Only by computing a statistical model of a gesture we can get rid of this limitation.

The first recognition can be made as shown in Fig. 5, where similarities between instances of the same action are quite apparent.

Using the DTW algorithm we built a matrix in which each element (n, m) has the distance value from the sequence n to the sequence m (Fig. 6). We computed these distances with two data subsets. In Fig. 6(left) elements 1, 2, 3 correspond to "POINTING AT" actions: we can see that the minimum distances between each one of these sequences and another one (notice that the distance of a

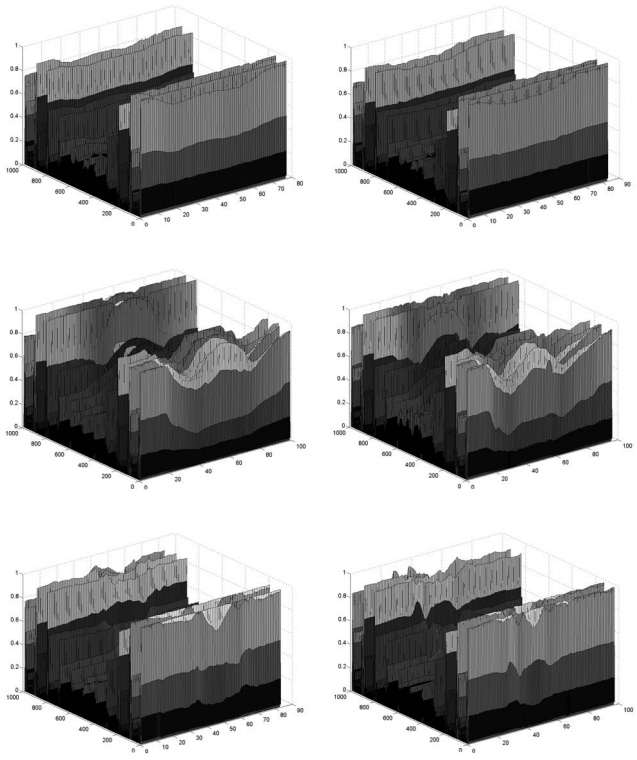

Figure 5: *Examples of feature matrices. The upper two matrices are instances of "POINTING AT" gestures, the middle ones are two "CROUCHING DOWN" actions while the lower graphs correspond to two "KICK" sequences.*

sequence from itself is zero, hence the black main diagonal) are concentrated inside the "POINTING AT" cluster (3×3 dark upper-left sub-matrix). The farthest ones from these sequences are the "CROUCHING DOWN" actions (white and light grey columns or rows) while the "KICK" gestures are a bit closer (grey sub-matrices). The same behavior is underlined by the other two clusters represented by the elements 4, 5 for "CROUCHING DOWN" action (note the central dark square) and the elements 6, 7 for "KICK"(lower-right corner square). The only difference among distances from "CROUCHING DOWN" is that "KICK" gestures are a bit closer (grey columns or rows) than "POINTING AT" ones. In conclusion "KICK" has an intermediate position between "POINTING AT" and "CROUCHING DOWN" according to DTW-computed distance. In the second data subset we have added two more "POINTING AT" sequences, this time performed by another person, and another two "KICK" actions, the last of these concerning another actor. This time the distances matrix computed draws our attention to a problem (Fig. 6-right). It is, in fact, noticeable that the new "POINTING AT" sequences tend to fall outside clusters boundaries. More precisely, the fourth action has the same distance (fourth white-light grey column or row) from each of the others and the fifth seems to be closer to "KICK" instances. On the other hand the new "KICK" actions, the tenth and the eleventh, are close to each other but are farther from the other "KICK" sequences. An explanation of these phenomena could be twofold: the adaptive technique performed on the reference shape (in order to make the features person-independent) could still be non optimal, and, at the same time, each actor could probably perform the same action in a dramatically different fashion. For example, watching carefully to the video recording of the fourth sequence, we notice that the actor moves his arm only, while the person in the other sequences moves arm, shoulder and bends his back in order to point at something.

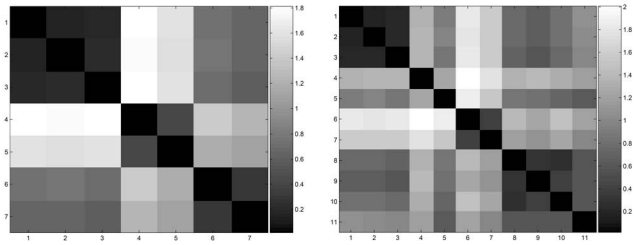

Figure 6: *Distances between sequences. Left: Comparison among 7 sequences: {1, 2, 3} = "POINTING AT"; {4, 5} = "CROUCHING DOWN"; {6, 7} = "KICK". Right: The same with 11 sequences: {1, 2, 3, 4, 5} = "POINTING AT"; {6, 7} = "CROUCHING DOWN"; {8, 9, 10, 11} = "KICK".*

5. Summary and Conclusions

In this paper we proposed an action-clustering system based on volumetric 3D data. Features have been represented by a *Shape Descriptor* computed frame-by-frame and adapted in order to be independent from position, size, scale, body proportions and, possibly, be invariant to rotations. We used a rather simple, but robust, pattern recognition algorithm, *Dynamic Time Warping*, to compute distances among gestural actions.

The performance shown by the experiments have highlighted the abilities of this system based on *Shape Descriptor* not only to recognize postures, as shown in [4], but also to be tuned up in a dynamic context. The simulations that have been carried out have demonstrated the ability of the proposed method in classifying the different considered actions. Moreover the algorithm can be parallelized and, in our opinion, after an optimization procedure, it will reach real-time performance.

The system introduced in this paper is only the first step

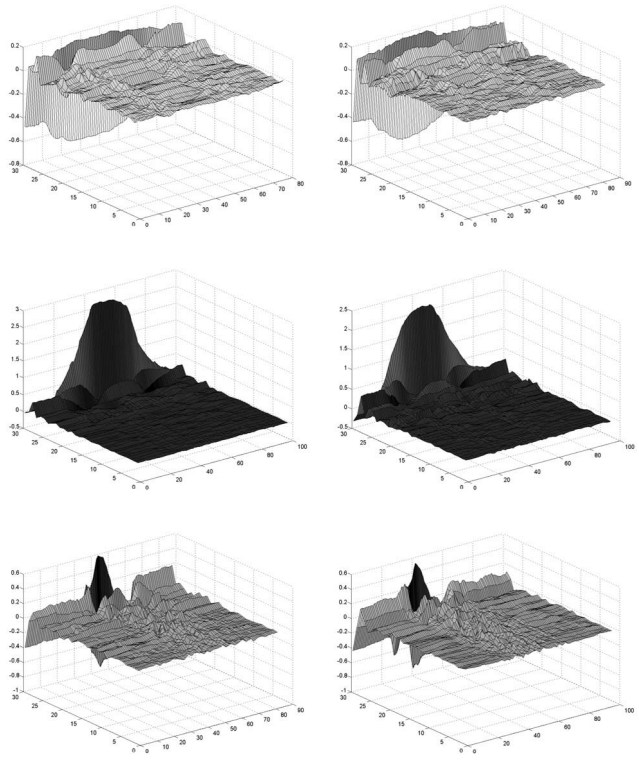

Figure 7: *Examples of feature matrices reduced using PCA techniques. These are the same feature matrices as Fig. 5, reduced by linear transformation using the first 30 eigenvectors of the covariance matrix.*

towards a more complex gesture recognition system. There are different ways through which we intend to proceed. The implementation of a recognition machine with a better generalization performance is obviously one of the most important. Hidden Markov Models could be a solution, but in order to deal with these features in a more complex system we should reduce feature dimensionality by means of effective techniques. We have already tried that with Principal Component Analysis (Fig. 7) but we think that approaches like Isomap or, even, Independent Component Analysis would better exploit the complex statistics under Shape Descriptor features.

References

[1] Y. Ivanov, C. Stauffer, A. Bobick and W. E. L. Grimson, "Video surveillance of interactions," *In IEEE Proceedings of the CVPR'99 Workshop on Visual Surveillance*, pp. 82-89, June 1998.

[2] D. DiFranco, T. Cham and J. Rehg, "Reconstruction of 3D figure motion from 2D correspondences," *In IEEE Proceedings of CVPR'01 International Conference on Computer Vision and Pattern Recognition*, December 2001.

[3] C. Sminchisescu and B. Triggs, "Covariance scaled sampling for monocular 3D body tracking," *In IEEE Proceedings of CVPR'01 International Conference on Computer Vision and Pattern Recognition*, December 2001.

[4] I. Cohen and H. Li, "Inference of human postures by classification of 3D human body shape," *In IEEE Proceedings of AMFG'03 International Workshop on Analysis and Modeling of Faces and Gestures*, pp. 74-81, October 2003.

[5] L. Campbell, D. Becker, A. Azarbayejani, A. Bobick and A. Pentland, "Invariant features for 3-D gesture recognition," *In IEEE Proceedings of FG'96 Second International Conference on Face and Gesture Recognition*, pp. 157-162, October 1996.

[6] S. Seitz and C. Dyer, "View-invariant analysis of cyclic motion," *International Journal of Computer Vision*, vol. 25, num. 3, pp. 231-251, 1997.

[7] C. Rao, A. Yilmaz and M. Shah, "View-invariant representation and recognition of actions," *International Journal of Computer Vision*, vol. 50, num. 2, pp. 203-226, 2002.

[8] F. Cuzzolin, A. Sarti, S. Tubaro, "Invariant action classification with volumetric data", *In IEEE MMSP'04 Workshop on Multimedia Signal Processing*, pp. 395-398, September 2004.

[9] A. D. Wilson and A. F. Bobick, "Parametric Hidden Markov Models for gesture recognition," *In IEEE Transactions on PAMI'99 Pattern Analysis and Machine Intelligence*, vol. 21, num. 9, pp. 884-900, September 1999.

[10] M. Brand, N. Oliver, and A. Pentland, "Coupled HMM for complex action recognition," *In IEEE Proceedings of CVPR'97 International Conference on Computer Vision and Pattern Recognition* vol. 29, pp. 213-244, June 1997.

[11] S.Theodoridis, K.Koutroumbas, *Pattern recognition*, Elsevier Academic Press, 2003.

[12] S. C. Brofferio, *Riconoscimento dei segnali: teorie e tecniche*, Libreria Clup, 2003.

[13] Z. Yue, L. Zhao, R. Chellappa, "View synthesis of articulating humans using visual hull," *In IEEE Proceedings of ICME'03 International Conference on Multimedia and Expo*, vol. 1, pp. 489-92, July 2003.

[14] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*, Cambridge University Press, 2000-2003.

[15] O. Grau, T. Pullen, G.A. Thomas, "A combined studio production system for 3-D capturing of live action and immersive actor feedback," *In IEEE Transactions on CSVT'04 Circuits and Systems for Video Technology*, Vol. 14, Issue 3, pp. 370-380, March 2004.