PROJECT: Visual Perception System for a Social Robot (VISOR)
TYPE OF PROJECT: Topical Research Studies (TRS)

DELIVERABLE 3:

HUMAN MOTION CAPTURE

Juan Pedro Bandera
Grupo de Ingeniería de Sistemas Integrados
Dpto. Tecnología Electrónica , Universidad de Málaga
Campus de Teatinos 29071 Málaga (Spain)
www.grupoisis.uma.es

Due date of deliverable: June 15th, 2006
Actual submission date: June 15th, 2006
Start date of project: September 15th, 2005
Duration: 12 months

Organizational name of responsible for this deliverable:
Grupo de Ingeniería de Sistemas Integrados
Revision: 0.2
Dissemination level: PU

Dept. of Electronic Technology, University of Málaga (Spain)
Institute of Systems and Robotics, Coimbra (Portugal)

Contents

List of Figures

1 Introduction

The analysis of human actions by a computer is gaining more and more interest. An important part of this task is to register the motion, a process known as human motion capture. This process can be formally defined as [1]:

> The process of capturing the large scale body movements, of a subject, at some resolution.

In this work, a real-time human motion capture system based on computer vision is presented. The goal of this work is to extract the upper body movements of a person without using any beacons or markers, using only two stereo cameras. The use of markers is normally intrusive, it often necessitates the use of expensive specialized hardware and it can only be used on footage taken specially for that purpose [2]. Several markerless approaches to human motion capture have been recently proposed [2, 3]. All these approaches present the same problem: they take several seconds to process one frame.

The proposed human motion capture approach [4, 5, 6] is based on a novel hierarchical tracking system [7]. Since such a system is unstable and can only acquire partial information because of self-occlusions and depth ambiguity, a model-based pose estimation method based on inverse kinematics has been also employed. The resulting system can estimate upper body human postures with limited perceptual cues, such as centroid coordinates and disparity of head and hands.

The key idea behind this system is the assumption that in order to track the global upper human body motion, it is not necessary to capture with precision the motion of all its joints. Particularly, in this work only the movement of the head and hands of the human are tracked, because they are the most significant items involved in the human-to-human interaction processes. These are modelled by weighted templates that are updated and tracked at each frame using the previously mentioned hierarchical tracking approach. The pose of the joints is then extracted through the use of a kinematic model of the human to track. It is also assumed that the human motion speed is bounded and that the pose of the different items to track is related to its last detected pose. By assuming this important constraints, the proposed system can estimate upper-body human motion at 25 frames per second.

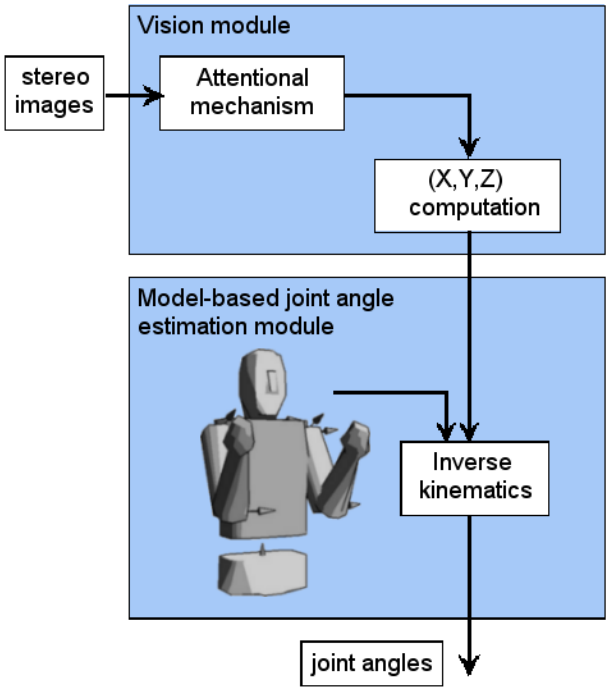


Figure 1: Overview of the proposed human motion capture system

An overview of the proposed system is shown in Fig. 1. The system has two main modules: a vision module and a joint angle extraction module. The vision module extracts the 3D coordinates of the head and hands of the human using the attentional mechanism previously explained in Deliverable 2 [8] which includes the hierarchical tracking algorithm. These (X, Y, Z) coordinates are used by the model-based joint angle extraction module to compute the pose of the upper-body joints by means of a kinematic model and a inverse kinematics algorithm.

2 Vision module

The main stage of the vision module is the attentional mechanism previously explained in Deliverable 2 [8] which has been slightly modified as shown below. The output of this attentional mechanism are the 2D coordinates of the head and the hands and their disparities. This information is combined using the calibration parameters of the cameras to obtain the 3D coordinates

of the head and hands. These 3D coordinates are the outputs of the vision module.

2.1 Attentional mechanism

The general purpose attentional mechanism proposed in Deliverable 2 has been slightly modified in this human motion capture system in order to extract as relevant information the position of the head and hands of the human whose movements are being tracked. In order to do that, the skin colour and the disparity are the only used low level features. The disparity computation method has been changed. In this work the Small Vision System (SVS) provided by Videre Design (www.videredesign.com) has been used to extract a more accurate disparity map. SVS is a set of library functions which implement the stereo algorithms. The disparity map is computed using a correlation-based algorithm.

The obtained disparity map is processed in order to extract the silhouette of the person. To do that, the face detection algorithm presented in Deliverable 1 [9] is used to determine the position of the human face in the input image. The mean value of the disparity of the localized face is used as threshold to reduce the number of disparity values in the disparity map. That is, only a certain number of disparities over this reference and below it are taken into account, the rest of values are removed from the map. This filtering is based in the fact that the maximum distance between the head and one hand is determined by the length of a stretched arm. We consider this length not to be superior to one meter. Thus, all disparities over this threshold are discarded. The result of this first filtering process is shown in Fig. 2.c.

Once this new map is obtained, the silhouette of the person is extracted using connected components (Fig. 2.d). The hands of the person are determined as the biggest skin colour regions located inside of the silhouette. These hands and the face are the extracted salient regions which are tracked by the hierarchical tracking algorithm included in the attentional mechanism. Therefore, the attentional mechanism is able to compute in each frame the 2D position of the head and the hands and their disparity values, as shown in Fig. 2.d.

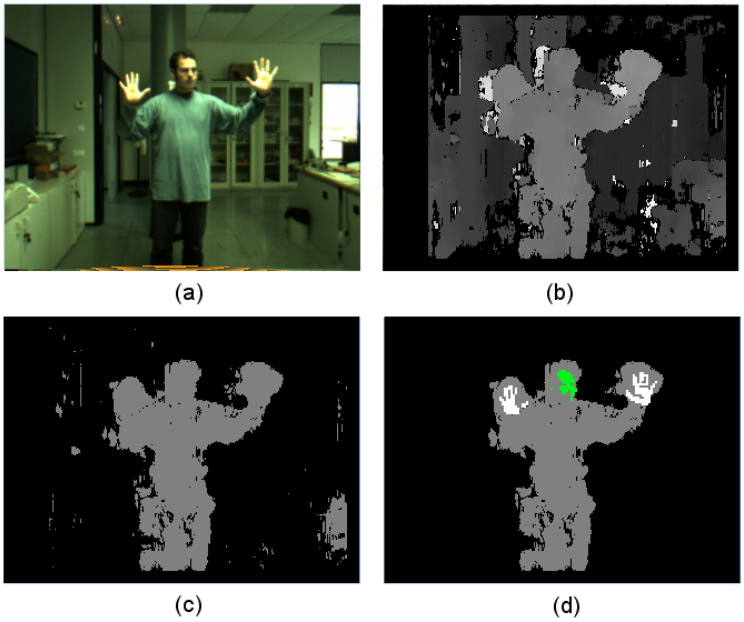


Figure 2: a) Left image of an input stereo pair; b) Disparity map; c) Relevant disparities (grey); and d) Extracted silhouette (grey), tracked face (green) and tracked hands (white).

3 Model-based pose generator

Our approach is exclusively based on the information obtained from the stereo vision system of the robot imitator. Thus, it is related to other experiments, e.g. the mimicking experiments show by Sauser and Billard [10], but in our case external color marks are not employed. As explained above, the information extracted for each frame is restricted to 3D positions of head and hands. Wren and Petland already developed a system to recover human motion from these limited cues, using physical constraints and probabilistic influences [11]. They also use a model to help in the tracking process by projecting 3D virtual blobs into 2D images taken with the stereo pair and improve pose estimation in a recursive scheme. The resulting system allows to track human upper-body movements at 30 fps, but it has to be manually initialized and requires several computers working on parallel due to its complexity.

Our system also uses a kinematic human model to translate 3D head and hands positions to a correct pose. But we base the translation in a fast analytic inverse kinematics algorithm running over a model that avoids incorrect poses. This model filters tracked movements and provides, in real-time, a set of joint angles that conforms a valid human pose and preserves perceived 3D positions.

3.1 Model

We have restricted ourselves to capture upper body motion. Thus, the geometric model contains parts that represent hips, head, torso, arms and forearms of the human to be tracked. Each of these parts is represented by a fixed mesh of few triangles, as depicted in Fig. 3. This representation has the advantage of allowing fast computation of collisions between parts of the model, which will help in preventing the model from adopting erroneous poses due to tracking errors.

Each mesh is rigidly attached to a coordinate frame, and the set of coordinate frames is organized hierarchically in a tree. The root of the tree is the coordinate frame attached to the hips, and represents the global translation and orientation of the model. Each subsequent vertex in the tree represents the three-dimensional rigid transformation between the vertex and its parent. This representation is normally called a skeleton or kinematic chain [12] (Fig. 3). Each vertex, together with its corresponding body part attached is called

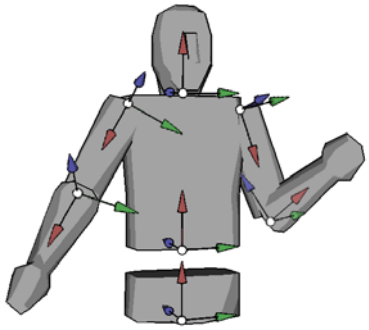


Figure 3: Illustration of the human upper-body kinematic model

a bone. Each bone is allowed to rotate –but not translate– with respect to its parent around one or more axes. Thus, at a particular time instant t, the pose of the skeleton can be described by $\Phi^{(t)} = (R^{(t)}, \vec{s}^{\,(t)}, \phi^{(t)})$, where $R^{(t)}$ and $\vec{s}^{\,(t)}$ are the global orientation and translation of the root vertex, and $\phi^{(t)}$ is the set of relative rotations between successive children. For upper-body motion tracking, it is assumed that only ϕ needs to be updated –this can be seen intuitively as assuming that the tracked human is seated on a chair.

Fig. 3 shows the 3D kinematic model used in this system. It has four degrees of freedom (DOF) in each arm. Three of them are located in the shoulder, and one in the elbow. Model proportions and dimensions have been set to average human values.

3.2 Inverse kinematics

As shown in Fig. 4, each arm is modelled with a two-bone kinematic chain. The parent bone corresponds to the upper arm and is allowed to rotate around three perpendicular axes. This provides a simplified model of the shoulder joint. $T({}^{w}_{1}R)$ is the local transformation between the upper-arm reference frame O_1 and a coordinate frame attached to the torso and centered at the shoulder joint w. The bone representing the lower arm is allowed to rotate around a single axis, corresponding to the elbow joint. $T({}^{2}_{1}R, {}^{1}\vec{l}_1)$ denotes the local transformation between the upper-arm reference frame O_1 and the lower-arm reference frame O_2, where ${}^{1}\vec{l}_1 = (0, 0, l_1)^T$, being l_1 the length of the upper-arm, and ${}^{2}_{1}R$ corresponds to the rotation θ_e about the elbow axis.

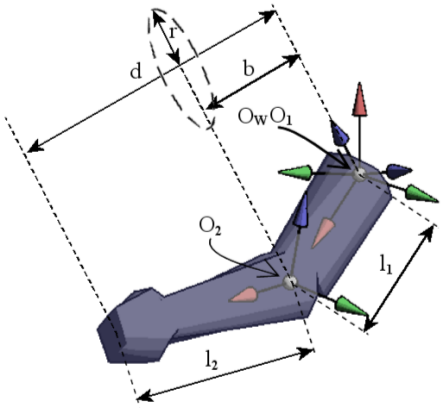


Figure 4: Kinematic model of the arm showing local coordinate frames and elbow circle (see text).

Given a desired position for the end-point of the arm at time instant $t+1$, ${}^w\vec{p}_d^{\,(t+1)}$, and given the rotation matrices ${}^w_1R^{(t)}$ and ${}^1_2R^{(t)}$ at the previous time instant t, the problem is then to find the updated matrices ${}^w_1R^{(t+1)}$ and ${}^1_2R^{(t+1)}$. A simple geometric method is summarized here that can solve such problem. See [13] for further details.

1. *Bring* ${}^w\vec{p}_d^{\,(t+1)}$ *within reach of the arm:*

$$\text{if } |{}^w\vec{p}_d^{\,(t+1)}| > (l_1 + l_2) \quad \text{then } {}^w\vec{p}_d^{\,(t+1)} \leftarrow {}^w\vec{p}_d^{\,(t+1)} \frac{l_1 + l_2}{|{}^w\vec{p}_d^{\,(t+1)}|}$$

2. *Compute elbow circle:* . Posing the model arms is an under-constrained problem, as four degrees of freedom must be specified from only three constraints, corresponding to the co-ordinates of the desired end-point position ${}^w\vec{p}_d^{\,(t+1)}$. The elbow circle is defined as the set of positions that the elbow is free to adopt when the end-point of the arm reaches ${}^w\vec{p}_d^{\,(t+1)}$. It has a radius r and it is contained in a plane perpendicular to the vector ${}^w\vec{p}_d^{\,(t+1)}$ at a distance b to the shoulder joint.

$$r^2 = \frac{(d + l_1 + l_2)(-d + l_1 + l_2)(d - l_1 + l_2)(d + l_1 - l_2)}{2d}$$

$$b = \sqrt{l_1^2 - r^2}$$

where $d = |{}^w\vec{p}_d^{(t+1)}|$

3. *Choose updated elbow axis* ${}^w\vec{x}_2^{(t+1)}$ *and location* ${}^w\vec{l}_1^{(t+1)}$ *:* We chose the elbow axis at time instant $t+1$ to be the closest to the one at the previous time instant, ${}^w\vec{x}_2^{(t)}$:

$$ {}^w\vec{x}_2^{(t+1)} = ({}^w\vec{p}_d^{(t+1)} \wedge {}^w\vec{x}_2^{(t)}) \wedge {}^w\vec{p}_d^{(t+1)} $$

$$ {}^w\vec{l}_1 = b\frac{{}^w\vec{p}_d^{(t+1)}}{|{}^w\vec{p}_d^{(t+1)}|} + r\frac{{}^w\vec{x}_2^{(t+1)} \wedge {}^w\vec{p}_d^{(t+1)}}{|{}^w\vec{x}_2^{(t+1)} \wedge {}^w\vec{p}_d^{(t+1)}|} $$

4. *Fill updated rotation matrices* ${}^w_1R^{(t+1)} = ({}^w\vec{x}_1\ {}^w\vec{y}_1\ {}^w\vec{z}_1)$ *and* ${}^1_2R^{(t+1)} = ({}^1\vec{x}_2\ {}^1\vec{y}_2\ {}^1\vec{z}_2)$ *with:*

$$ \begin{array}{ll} {}^w\vec{x}_1 = {}^w\vec{x}_2 & {}^1\vec{x}_2 = (1,0,0) \\ {}^w\vec{z}_1 = {}^w\vec{l}_1/|{}^w\vec{l}_1| & {}^1\vec{z}_2 = {}^w R_1({}^w\vec{p}_d - {}^w\vec{l}_1) \\ {}^w\vec{y}_1 = {}^w\vec{z}_1 \wedge {}^w\vec{x}_1 & {}^1\vec{y}_2 = {}^1\vec{z}_2 \wedge {}^1\vec{x}_2 \end{array} $$

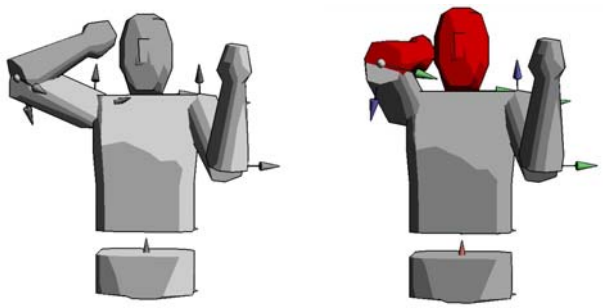
3.3 Enforcement of joint limits and collision avoidance

The proposed inverse kinematics method can obtain an arm pose that will put the hand of the model in the required position. The resulting pose must be analyzed in order to determine if it corresponds with a valid and natural body configuration. In this work we consider two limitations: a valid pose must respect joint limits and cannot produce a collision between different links.

- *Detection of joint limit violations.* Given the updated shoulder and elbow rotation matrices, it is necessary to extract joint angles from these matrices that correspond to the DOFs of the human model.

  This process is made by applying a parameterization change to rotation matrices. There is a direct correspondence between Denavith-Hartenberg (DH) [14] parameters and model joint angles, so the local axes referred angles are converted to DH parameters. The shoulder conversion can be done applying the following parameterization to the rotation matrix ${}^w_1R$:

$$ {}^w_1R = \begin{pmatrix} c\theta_2 c\theta_3 & -c\theta_2 s\theta_3 & s\theta_2 \\ s\theta_1 s\theta_2 c\theta_3 + c\theta_1 s\theta_3 & -s\theta_1 s\theta_2 s\theta_3 + c\theta_1 c\theta_3 & -s\theta_1 c\theta_2 \\ -c\theta_1 s\theta_2 c\theta_3 + s\theta_1 s\theta_3 & c\theta_1 s\theta_2 s\theta_3 + s\theta_1 c\theta_3 & c\theta_1 c\theta_2 \end{pmatrix} $$



(a) (b)

Figure 5: RAPID collision detection: (a) Valid pose. (b) Collision.

where θ_1, θ_2 and θ_3 are the real DOFs of the model arm, $c\theta_i$ is $cos\theta_i$ and $s\theta_i$ is $sin\theta_i$.

The elbow angle is much easier to obtain: as there is only one DOF in the elbow, the local rotation angle is equal to model θ_4 angle.

Once the model DOFs are computed, the system can directly check if any of them lies beyond its limits.

- *Collision detection.* We use RAPID [15] as the base of the collision detection module. This library provides functions that can quickly and efficiently check collisions between meshes composed by triangles, such as the ones attached to the links in our model (Fig. 5).

Once the system detects an incorrect position (i.e. joint limit or collision), it follows these steps:

1. The system looks for alternative poses (i.e. different arm configurations). Imitation requires to place hands in certain coordinates, but the elbow is free to move in the circle presented in Fig. 4. Thus, alternative poses will preserve hand positions, but will move the elbow in this circle.

2. The motion of the arm should be as smooth as possible. Thus, alternatives should be more densely searched near the current elbow location. This is implemented by exponentially distributing the alternatives around the initial incorrect elbow position, as shown below:

$$\theta_{2i} = \pi \frac{1}{100^{\frac{(n-i)}{n}}}$$
$$\theta_{2i+1} = -\theta_{2i}$$
$$i = 0, 1, 2, ...(n-1) \quad (1)$$

where θ_{2i} and θ_{2i+1} correspond to two symmetric alternatives on the elbow circle with respect to the current pose, and $n = \frac{N}{2}$, being N the number of alternative poses checked when current pose is erroneous. This distribution places alternative poses on the elbow circle (Fig. 4). As required, alternatives are more deeply distributed near the current elbow position.

3. The system chooses the nearest valid alternative.

4. If there is no valid alternative, the arm remains in the last valid position.

The speed of the process depends on the number of alternatives it needs to check. A system using a correct number of alternatives should produce smooth movements and work in real-time even in the case in which all of them need to be checked.

The alternative evaluation module has been also used when the system is in a valid pose: in these cases, the two nearest alternatives to current pose are checked. If one of them locates the elbow in a lower vertical position, and do not produce limits violation nor collisions, then the elbow is moved to that position. This allows the model to adopt more natural poses when possible.

3.4 Scaling the model to fit the human

In order to coherently follow the movements of the human, the 3D model will be scaled to match demonstrator's height. The scale ratio will be the following:

$$ratio = \frac{height_{human}}{height_{model}} \quad (2)$$

In our implementation, the model height is 170 cm. The human height is determined by the 3D position of the human head, provided by the vision module.

Imitated motions are then easily normalized by simply re-scaling the model to its original size while preserving the joint angles sequence. In this way, motions of very different people can be analyzed and compared.

4 Experimental Results

The proposed system has been tested using a STH-DCSG-VARX stereo system and the Small Vision System software, provided by Videre Design (www.videredesign.com). This architecture captures and preprocesses stereo pairs. The size of left and right images is 320x240. The disparity map has also a size of 320x240.

The face of the demonstrator is detected using a cascade detector based on the scheme by Viola and Jones (see [16] for details). The particular implementation of this scheme for the proposed system is deeply explained in Deliverable 1 [9]. The 3D virtual model used to reproduce perceived gestures is rendered and animated using OpenSceneGraph, an open source graphic engine available at www.openscenegraph.org.

The whole system runs on a 2 GHz. Pentium IV computer using Linux operating system.

The experiments performed to test the Human Motion Capture System involved different demonstrators moving their hands in a non-controlled environment. The only imposed requisite was to wear long sleeves. Besides, as the stereo system has a limited range, the demonstrator was told to stay at more than 1.50 meters from the cameras. Fig. 6 show the results obtained by the proposed system at an average rate of more than 25 fps.

As shown in Fig. 6, the generated pose closely resembles the pose of the human demonstrator. The figure also shows that the disparity map computed by the Videre system presents some noise. This noise will introduce errors to the perceived depth of the head and the hands. The first versions of the proposed system tried to reduce those errors by averaging the disparity values of the skin pixels in the regions of interest. This lead to better results, but still the errors in some pixels, specially those located in the borders, tended to distort the results.

The current version of the system takes into account the confidence value for the disparity of each pixel provided by the SVS software to reduce the disparity noise. After several tests, it was decided that the best option was to simply take as disparity value for each region the one associated with

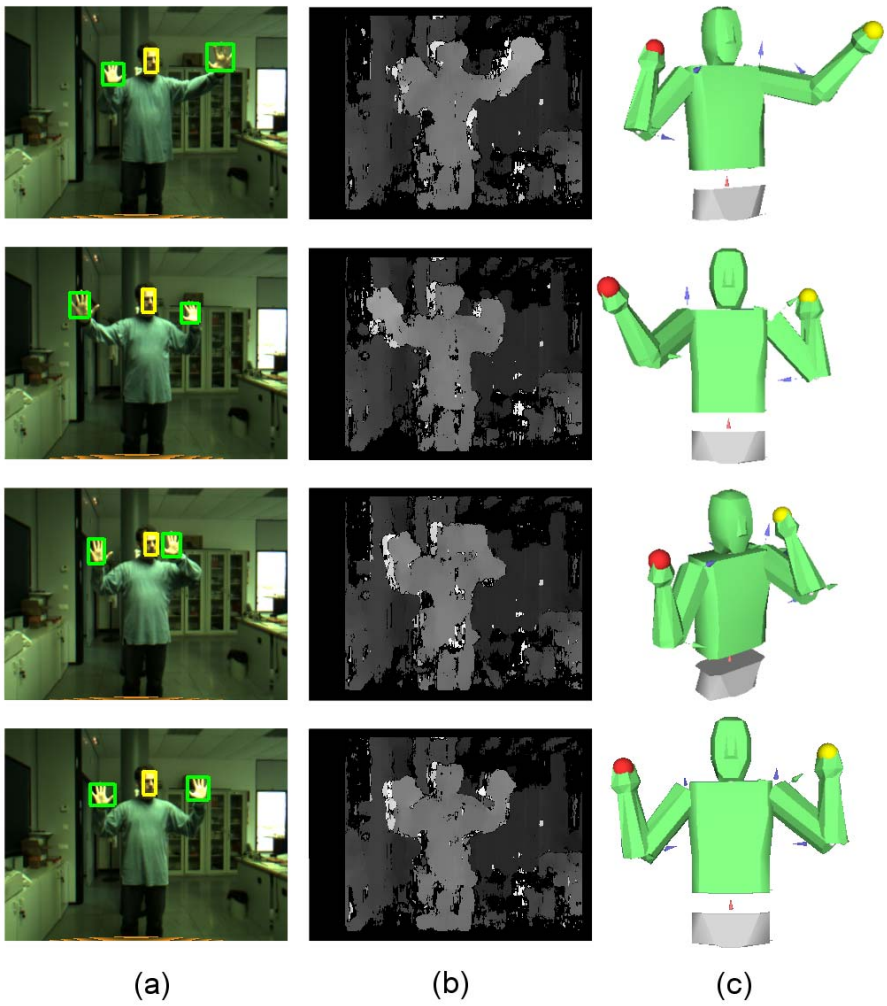


Figure 6: Human Motion Capture system: a) Left image of the stereo pair with head (yellow) and hands (green) regions marked; b) Disparity map; and c) 3D model showing generated pose.

the highest confidence into that region. Then, the disparity value and the pixel coordinates of the region centroid are used in combination with camera parameters to extract (X, Y, Z) coordinates of the head and the hands. In our system, the distance between the human and the cameras is around 170 cm. For these values, the theoretical depth resolutions for the SVS are under 1 cm - more precisely, 7 mm for a distance of 165 cm. But, as commented above, there are different sources that introduce errors in the disparity map and in the tracking algorithm. These errors reduce the effective resolution of the stereo system. In any case, the average error is less than 5 centimeters. This is a good enough result as the 3D model will help correcting incorrect poses.

Figs. 7 and 8 show frames of an imitation sequence involving a different demonstrator. While in Fig. 7 the movements of the human are fairly smooth and are performed in an easily reachable area (hands quite separated from the body), Fig. 8 shows more problematic frames. In this part of the sequence the demonstrator is performing fast and large movements. The hands, also, move near the body in some occasions, as depicted in the frames in Fig. 8. The model is able to find a valid and natural pose in these situations, although sometimes the position of the elbow differs respect to human pose, as in the second frame.

The third frame of Fig. 8 shows a situation in which the pose error is larger than the average due to a substantial reduction in the size of the tracked region. The region is, in fact, reduced to only one point. Thus, the disparity of the region is computed using the value for this pixel. In this case, this was not an accurate value, and the resulting pose is quite different to the demonstrated one. Our future work will have to focus on this issue and improve results in this situation, using information about previous frames to filter the current position.

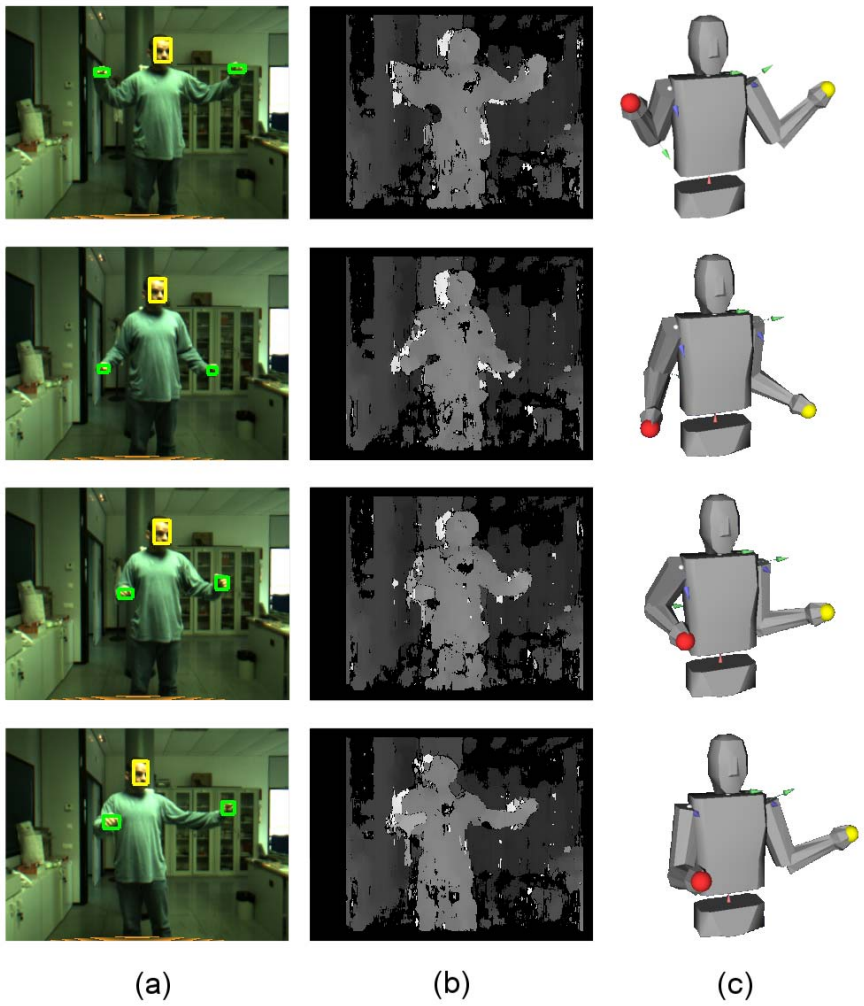


Figure 7: Human Motion Capture system: a) Left image of the stereo pair with head (yellow) and hands (green) regions marked; b) Disparity map; and c) 3D model showing generated pose.

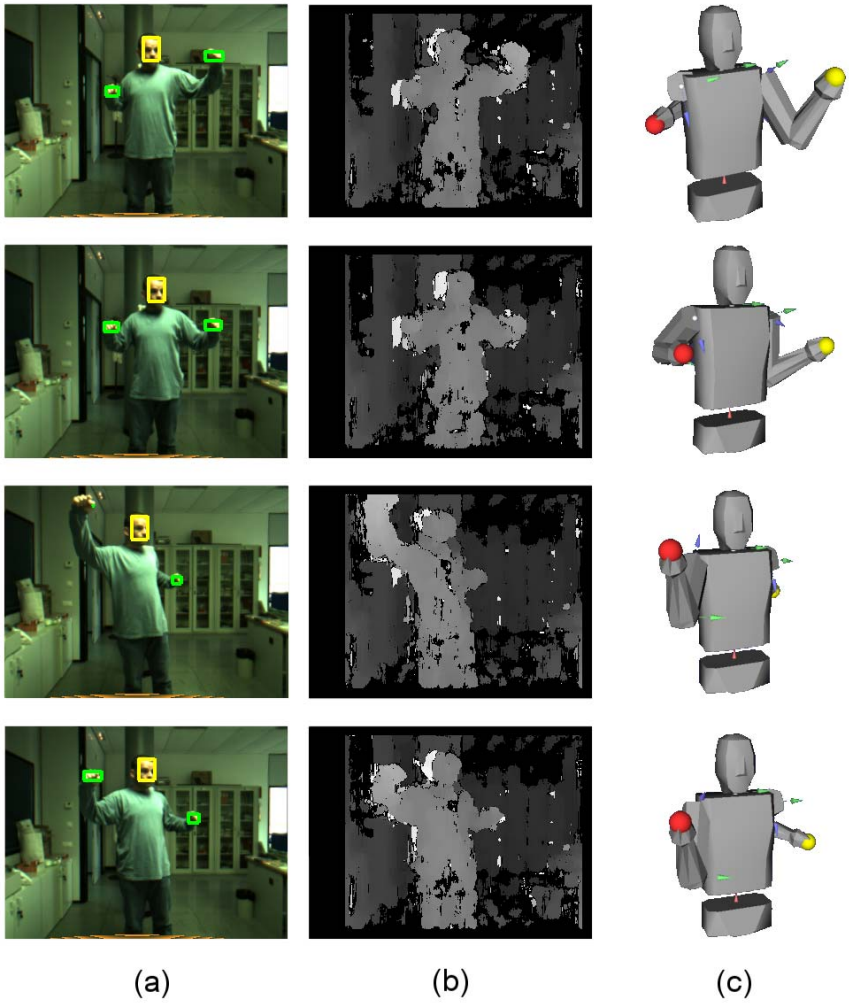


Figure 8: Human Motion Capture system: a) Left image of the stereo pair with head (yellow) and hands (green) regions marked; b) Disparity map; and c) 3D model showing generated pose.

References

[1] T. B. Moeslund and E. Granum, "A survey of computer vision-based human motion capture," *Computer Vision and Image Understanding*, vol. 81, no. 3, pp. 231–268, 2001.

[2] J. Deutscher, A. Davison, and I. Reid, "Automatic partitioning of high dimensional search spaces associated with articulated body motion capture," *Proc. of the 2001 IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 2, pp. 669–676, 2001.

[3] P. Tresadern and I. Reid, "Uncalibrated and unsynchronized human motion capture: A stereo factorization approach," *Proc. of the 2004 IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 1, pp. 128–134, 2004.

[4] J. Bandera, L. Molina-Tanco, R. Marfil, and F. Sandoval, "A model-based humanoid perception system for real-time human sotion imitation," *Proc. of the IEEE Conference on Robotics, Automation and Mechatronics*, pp. 523–129, 2004.

[5] J. Bandera, R. Marfil, L. Molina-Tanco, A. Bandera, and F. Sandoval, "Model-based pose estimator for real-time human-robot interaction," *Third International Conference on Computational Intelligence, Robotics and Autonomous Systems*, 2005.

[6] L. Molina-Tanco, J. Bandera, R. Marfil, and F. Sandoval, "Real-time human motion analisys for human-robot interaction," *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1808–1813, 2005.

[7] R. Marfil, "Tracking objects with the bounded irregular pyramid," Ph.D. dissertation, Dpto. Tecnologia Electronica, Universidad de Malaga, Spain, 2006.

[8] R.Marfil, "Attentional mechanism," " Visual Perception System for a Social Robot (VISOR) Project, Deliverable 2, Grupo de Ingenieria de Sistemas Integrados, University of Malaga, Spain, 2006.

[9] R.Marfil and J.Rett, "Skin colour detection, face detection and face recognition," " Visual Perception System for a Social Robot (VISOR)

Project, Deliverable 1, Institute of Systems and Robotics, Coimbra, Portugal, 2005.

[10] E. Sauser and A. Billard, "View sensitive cells as a neural basis for the representation of others in a self-centered frame of reference," *Proceedings of the Third International Symposium on Imitation in Animals and Artifacts, Hatfield-UK*, pp. 119–127, April 2005.

[11] C. Wren and A. Pentland, "Dynaman: Recursive modeling of human motion," " Technical Report TR-451, MIT, 1999.

[12] Y. Nakamura and K. Yamane, "Dynamics computation of structure-varying kinematic chains and its application to human figures," *IEEE Trans. on Robotics and Automation*, vol. 16, no. 2, pp. 124–134, 2000.

[13] J. Mitchelson, "Multiple-camera studio methods for automated measurement of human motion," Ph.D. dissertation, CVSSP, School of Electronics and Physical Sciences, Univ. of Surrey, UK, 2003.

[14] J.J.Craig, *Introduction to Robotics.* Addison-Wesley, 1986.

[15] M. L. S. Gottschalk and D. Manocha, "Obb-tree: A hierarchical structure for rapid interference detection," " Technical Report TR96-013, Department of Computer Science, University of N. Carolina, 1996.

[16] P. Viola and M. Jones, "Robust real-time face detection," *Int. Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.