2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

Toward an Unified Representation for Imitation of Human Motion on Humanoids

Pedram Azad, Tamim Asfour and Rüdiger Dillmann
Institute for Computer Science and Engineering
University of Karlsruhe,
Haid-und-Neu-Strasse 7, 76131 Karlsruhe, Germany
Email: azad|asfour|dillmann@ira.uka.de

Abstract— **In this paper, we present a framework for perception, visualization, reproduction and recognition of human motion. On the perception side, various human motion capture systems exist, all of them having in common to calculate a sequence of configuration vectors for the human model in the core of the system. These human models may be 2D or 3D kinematic models, or on a lower level, 2D or 3D positions of markers. However, for appropriate visualization in terms of a 3D animation, and for reproduction on an actual robot, the acquired motion must be mapped to the target 3D kinematic model. On the understanding side, various action and activity recognition systems exist, which assume input of different kinds. However, given human motion capture data in terms of a high-dimensional 3D kinematic model, it is possible to transform the configurations into the appropriate representation which is specific to the recognition module. We will propose a complete architecture, allowing the replacement of any perception, visualization, reproduction module, or target platform. In the core of our architecture, we define a reference 3D kinematic model, which we intend to become a common standard in the robotics community, to allow sharing different software modules and having common benchmarks.**

I. Introduction

In the recent past, research on visual perception and understanding of human motion has become of high interest. On the one hand, a large number of various approaches on the perception side exists, resulting in different human motion capture systems, producing output in terms of different models and stored in different formats. On the other hand, many action recognition and activity recognition systems exist, expecting input data specific to their own internal representation. Furthermore, any target platform for the reproduction of human motion, namely 3D models for animation and simulation purposes, and humanoid robots, expects human motion capture data in terms of its own 3D kinematic model. Because of this fact, currently it is not possible to exchange single modules in an overall infrastructure for a humanoid robot, including perception, visualization, reproduction, and recognition of human motion. Furthermore, having common benchmarks is only feasible when a common representation for human motion is shared.

Methods for modeling, generating of human motion and its reproduction on humanoid robots have been proposed in robotics and computer graphics ([1], [2], [3], [4]). These methods involve capturing full body motion of a human performer using human motion capture systems and the transformation of the motion to the kinematics of a humanoid robot or human figures. To answer the question of how to generate and represent motor primitives in imitation learning architectures, several approaches have been proposed. In most related researches, continuous Hidden Markov Models (HMM) are used, where the Viterbi algorithm plays an important role to generate a close motion to the observation. In [5], a stochastic model has been proposed that abstracts the whole body motion as symbols and integrates motion recognition, generation, and symbolization of motion patterns. Nakamura presents in [6] a stochastic mimesis model for the imitation of new observed motions without learning the motion and for the online acquisition of motion patterns.

Human motion capture systems can be generally divided into marker-based and markerless systems. Marker-based systems are widely used in the film and animation industry, and in the research field of human motion analysis. One of the most popular marker-based motion capture systems is the VICON system [7], which can track a set of reflective markers that are attached to the person to be tracked. Such systems can acquire a very high accuracy and a high temporal resolution; their output is a set of 3D positions of markers over time in first place. This output data has then to be post-processed manually, and if necessary translated into the configuration space of a given 3D human model by determining rotations on the base of adjacent markers, as done in [8].

Markerless systems are often used in the context of visual perception for humanoid robot systems. Recently, many approaches have been proposed, differing in the number, type, and arrangement of the sensors incorporated, real-time applicability, the ability to perceive 2D or real 3D motion, and the smoothness of the output trajectories. When using multiple cameras, i.e. three or more cameras located around the area of interest, two different systems have shown very good results. The one class of approaches is based on the calculation of 3D voxel data, as done by [9], [10]. The other approach is based on particle filtering and became popular by the work of Deutscher et al. [11] for multi-camera systems, and later by [12] for the specific case of monocular image sequences. Recently, we have started to adapt and extend this system for real-time application on a humanoid robot head ([13], [14]). Other approaches depend on incorporation of an additional 3D sensor and the *Iterative Closest Point*

1-4244-0602-1/07/$20.00 ©2007 IEEE.

(ICP) algorithm, such as the Swiss Ranger, as presented by [15].

The output data computed by the mentioned systems is intended to be used for different applications, namely action and activity recognition, and the reproduction of movements on a humanoid robot system. For visualization and evaluation purposes, the data is usually also applied to a 3D human model. The problem is that most human motion capture systems are intended for the mentioned applications, but in practice the transfer rarely happens. The reason is that each system is based on its own representation and therefore the output data is always given in terms of this specific human model with its own data format, which is in general not compatible with systems for action or activity recognition, or the kinematic model of an actual robot system. Since one research group cannot deal with all the mentioned issues, it is crucial to overcome this deficiency to allow for compatibility of any module developed for any of the mentioned purposes. In the following, we propose a framework, defining a reference 3D kinematic human model in its core, and presenting the architecture, with which all modules are connected together. As a practical example, we show how we built in a markerless human motion capture system developed at our institute and the humanoid robot ARMAR III for reproduction of movements.

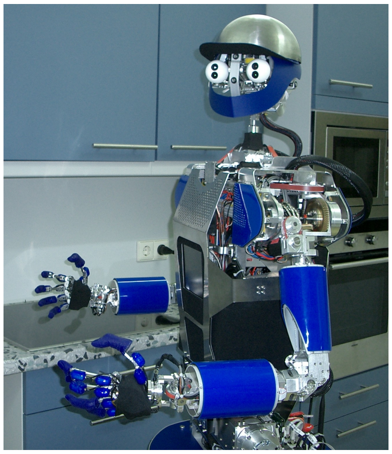

Fig. 1. ARMAR III with sensor head

II. Human Motion Capture

In this section, we want to give a short outline of marker-based and markerless human motion capture systems, in particular concentrating on the data input and output of the systems. In Section III, we show how their output can be mapped to one canonical representation.

A. Markerless Human Motion Capture

As mentioned in Section I, various approaches for markerless human motion capture exist. Here, we want to introduce a system intended for real-time application on an active head of a humanoid robot system, which has been developed at our institute ([13], [14]). The input of the system is a stereo color image sequence, captured with two calibrated Dragonfly cameras built-in into the head of the humanoid robot ARMAR III, which can be seen in Figure 1. The input images are preprocessed, generating output for the gradient cue, the distance cue, and an optional region cue, as described in [14]. The image processing pipeline for this purpose is illustrated in Figure 2.

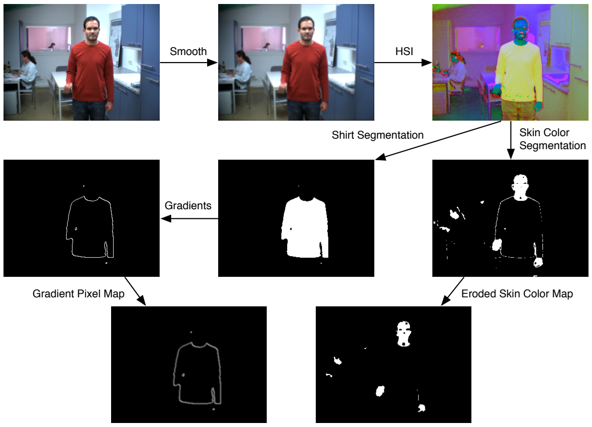

Fig. 2. Visualization of the image processing line

Based on the output of the image processing pipeline, a particle filter is used for tracking the movements in configuration space. The overall likelihood function to compute the a-posteriori probabilities is formulated as:

$$p(\mathbf{z}|\mathbf{s}) \propto \exp\left\{-\frac{1}{2}\left(\frac{1}{\sigma_d^2}\sum_{i=1}^{3} d_i(\mathbf{s},\mathbf{c}_i) + \frac{1}{\sigma_g^2}(1-\frac{1}{M_g}\sum_{m=1}^{M_g} g_m)\right)\right\} ,$$

where $\mathbf{s}$ is the configuration to be evaluated, $\mathbf{z}$ is a general denotation for the current observations i.e. the current input image pair, and $\mathbf{c}_i \in \mathbb{R}^3$ with $i \in \{1, 2, 3\}$ denotes the triangulated 3D position of the hands and the head. The function $d_i(\mathbf{s}, \mathbf{c})$ is defined as:

$$d_i(\mathbf{s},\mathbf{c}) := \left\{ \begin{array}{ccl} |f_i(\mathbf{s}) - \mathbf{c}|^2 & : & \mathbf{c} \neq \mathbf{0} \\ 0 & : & \text{otherwise} \end{array} \right. ,$$

where $n := \dim(\mathbf{s})$ is the number of DoF of the human model. The transformation $f_i : R^n \rightarrow R^3$ transforms the n-dimensional configuration of the human model into the 3D position of the left hand, right hand, or head respectively, using the forward kinematics of the human model.

The g_m with $m \in \{1, \ldots, M_g\}$ denote the intensity values in the gradient image (which is derived from the input images $\mathbf{z}$) at the M_g pixel coordinates of the projected contour of the human model for a given configuration $\mathbf{s}$. This process is performed for both input images using the calibration parameters of each camera.

A detailed description is given in [14]. For each image pair of the input sequence the output of the system is the estimation of the particle filter, given by the weighted mean $\bar{\mathbf{s}}$ over all particles. The format of the output configuration $\bar{\mathbf{s}}$ is:

$$\bar{\mathbf{s}} = (\mathbf{t}_{BT}\ \boldsymbol{\theta}_{BT}\ \boldsymbol{\theta}_{LS}\ \theta_{LE}\ \boldsymbol{\theta}_{RS}\ \theta_{RE})^T ,$$

where BT denotes the base transformation, LS and RS the shoulder joints for the left and right arm, and LE and RE the elbow joints. $\mathbf{t}_{BT} \in \mathbb{R}^{1\times 3}$ is the base translation, $\boldsymbol{\theta}_{BT}, \boldsymbol{\theta}_{LS}, \boldsymbol{\theta}_{RS} \in \mathbb{R}^{1\times 3}$ are rotations given in the Euler convention $R_{X'Z'Y'}(\alpha, \beta, \gamma)$, and $\theta_{RE}, \theta_{RE} \in \mathbb{R}$ are scalar values for the elbow angle α in the Euler convention $R_{X'Z'Y'}(\alpha, 0, 0)$. The Euler angle conventions can be found in the Appendix B of [16].

B. Marker-based Human Motion Capture

Marker-based human motion capture systems are commercially available and often used in the film and animation industry. The probably most popular one is the VICON system, which consists of a set of infrared cameras, with each having a diode array attached to it. The system offers a convenient calibration routine, which makes it possible to determine 3D positions for the reflective markers that come with the system. Since occlusions can occur, it is necessary to postprocess the output, which can be relatively time-consuming, depending on the number of markers used. In Figure 3, a typical marker setup for the acquisition of human upper body motion is illustrated. Having postprocessed the data, the human motion capture data, which consists of a temporal sequence of 3D marker position sets, can be used for various purposes.

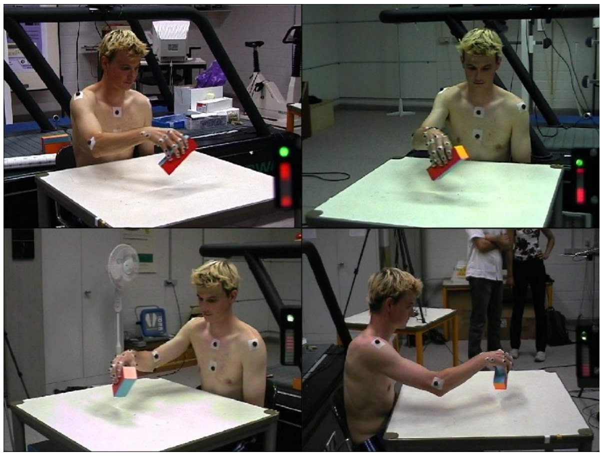

Fig. 3. Illustration of a marker-based human motion capture setup from [8]

For action recognition, it is possible either to use the plain 3D marker positions as input directly, or use trajectories in joint angle space. For the second approach, the trajectories of 3D marker position have to be transformed to joint angle trajectories by deriving rotations from the positions of adjacent markers, as described in [8]. Depending on the target kinematic model of the transformation, the joint angle trajectories can be used for visualization with an articulated 3D human model, or reproduced on an actual humanoid robot system. However, the problem is that usually this approach is a dead-end, since the target kinematic model is determined in advance, and therefore, the final data can be used only for the one desired purpose. Only with a lot of effort, human motion capture data can be shared within the robotics community, since an agreement on a common representation has been missing so far.

III. Master Motor Map

To overcome the deficiencies mentioned above, we propose a reference kinematic model, which we will call the *Master Motor Map* (*MMM*) in the following. The strategy is to define the maximum number of DoF that might be used by any visualization, recognition, or reproduction module, but not more than that. The H-Anim 1.1 specification [17] defines a joint for each vertebra of the spine, which is not suitable for the robotic applications mentioned. Moreover, the H-Anim 1.1 specification does only define relative joint positions in terms of a graph, but not the actual kinematic model including the joint angle conventions for each joint, which is crucial for any robotic application. Therefore, we have defined a subset of the H-Anim 1.1 specification and have specified the joint angle conventions for each joint.

The joints which build the subset of H-Anim 1.1 are *skullbase*, *vc7*, *vt6*, *pelvis*, *HumanoidRoot*, *l_hip*/*r_hip*, *l_knee*/*r_knee*, *l_ankle*/*r_ankle*, *l_sternoclavicular*/*r_sternoclavicular*, *l_shoulder*/*r_shoulder*, *l_elbow*/*r_elbow*, and *l_wrist*/*r_wrist*. The numbers of DoF and the Euler angle conventions are listed in Table I. The kinematic model for the MMM is illustrated in Figure 4.

	DoF	Euler angles
skullbase	3	$R_{X'Z'Y'}(\alpha,\beta,\gamma)$
vc7	3	$R_{X'Z'Y'}(\alpha,\beta,\gamma)$
vt6	3	$R_{X'Z'Y'}(\alpha,\beta,\gamma)$
pelvis	3	$R_{X'Z'Y'}(\alpha,\beta,\gamma)$
HumanoidRoot	3	$R_{X'Z'Y'}(\alpha,\beta,\gamma)$
l_hip / r_hip	$3+3$	$R_{X'Z'Y'}(\alpha,\beta,\gamma)$
l_knee / r_knee	$1+1$	$R_{X'Z'Y'}(\alpha,0,0)$
l_ankle / r_ankle	$3+3$	$R_{X'Z'Y'}(\alpha,\beta,\gamma)$
l_sternoclavicular / r_sternoclavicular	$3+3$	$R_{X'Z'Y'}(\alpha,\beta,\gamma)$
l_shoulder / r_shoulder	$3+3$	$R_{X'Z'Y'}(\alpha,\beta,\gamma)$
l_elbow / r_elbow	$2+2$	$R_{X'Z'Y'}(\alpha,\beta,0)$
l_wrist / r_wrist	$2+2$	$R_{X'Z'Y'}(\alpha,0,\gamma)$
Total	52	

TABLE I
NUMBER OF DEGREES OF FREEDOM AND EULER ANGLE CONVENTIONS FOR THE JOINTS OF THE MMM

The file format is specified as follows. The 52-dimensional configuration vectors are written sequentially to a text file, where each component is a floating point number formatted as readable text. All components are separated by whitespace. After one configuration, an additional floating point value specifies the associated timestamp in milliseconds. Since one configuration contains a fixed number of 53 numbers (including the timestamp), it is not necessary to introduce an explicit end of one configuration. For readability, it is recommended to put a line break after each timestamp instead of a space. The order of the 52 floating point numbers for the configuration vector is:

$$(\mathbf{t}_{RT}\ \boldsymbol{\theta}_{RT}\ \boldsymbol{\theta}_{SB}\ \boldsymbol{\theta}_{VC7}\ \boldsymbol{\theta}_{P}\ \boldsymbol{\theta}_{VT6}\ \boldsymbol{\theta}_{LSC}\ \boldsymbol{\theta}_{LS}\ \boldsymbol{\theta}_{LE}\ \boldsymbol{\theta}_{LW}$$
$$\boldsymbol{\theta}_{RSC}\ \boldsymbol{\theta}_{RS}\ \boldsymbol{\theta}_{RE}\ \boldsymbol{\theta}_{RW}\ \boldsymbol{\theta}_{LH}\ \theta_{LK}\ \boldsymbol{\theta}_{LA}\ \boldsymbol{\theta}_{RH}\ \theta_{RK}\ \boldsymbol{\theta}_{RA})^T$$

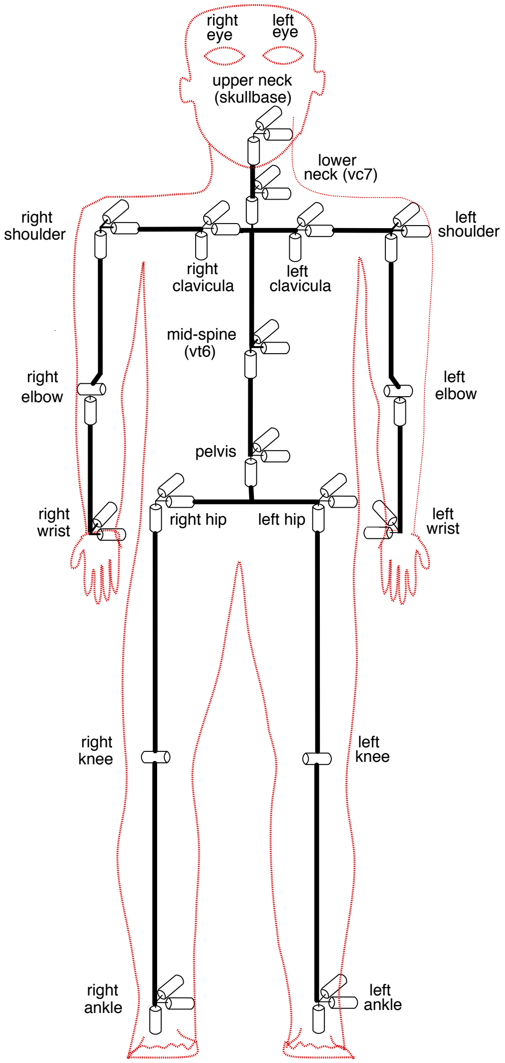

Fig. 4. Illustration of the MMM kinematic model

where RT denotes the root transformation, SB the skull base joint, P the pelvis joint, LSC/RSC the sternoclavicular joints, LS/RS the shoulder joints, LE/RE the elbow joints, LH/RH the hip joints, LK/RK the knee joints, and LA/RA the ankle joints. The length of each vector is given by the number of DoF, given in Table I.

IV. Framework and Converter Modules

In the following, we propose the framework which connects all mentioned modules, namely responsible for data acquisition, visualization, reproduction, and recognition. In the core of the framework is the MMM, as specified in Section III. All perceptive modules have an additionally implemented converter module, which transforms the output data to the MMM. Modules for visualization, reproduction, and recognition, which need motion capture data as input, implement an additional converter module, which transforms the data provided in the MMM format to the required input data format. This framework is illustrated in Figure 5.

The converter modules implement the transformation from one human motion representation to the MMM, or vice versa. In the case of marker-based human motion capture systems, this transformation is computed on the base of adjacent markers, as described in [8]. For all other modules, the converter module has to perform a transformation between two different kinematic models. There are five common basic types of adaptations which can occur in such a transformation:

1) Changing the order of values (all modules).
2) Setting zeroes for joint angles which are not captured (perception modules).
3) Ignoring joint angle values which can or are not to be used (reproduction and recognition modules).
4) Transformations between two different Euler angle conventions for a ball joint (all modules).
5) Adaptations that include more than one joint, if the target module does not offer the corresponding degrees of freedom (reproduction and recognition modules).

We will show three example converter modules, covering all five mentioned cases. One converter module is for mapping the output of our markerless human motion capture system to the MMM, the second for mapping the MMM to the kinematic model for the humanoid robot ARMAR, and the third for transforming 6D task space trajectories to the MMM. Cases 1, 2, and 3 are trivial. Case 4 can be solved by carefully calculating the conversion between two Euler angle conventions, as will be shown. Case 5 can not be generalized; we will show an example for the humanoid robot ARMAR. In the following, the notation $\mathbf{0}_i$ denotes the transposed zero vector of $\mathbb{R}^i$ i.e. $\mathbf{0}_i \in \{\nvdash\}^{1\times i}$.

A. Conversion Example 1

Here, we show how the output of our markerless human motion capture system is mapped to the MMM. The conversion covers the cases 1 and 2; case 4 does not occur, since the Euler angle conventions for the shoulder joints are both $R_{X'Z'Y'}(\alpha, \beta, \gamma)$, as is the base rotation for both models. Cases 3 and 5 are not of interest for perception modules. The transformation is formulated as follows:

$$f_1 : \mathbb{R}^{14} \rightarrow \mathbb{R}^{52}$$
$$(\mathbf{t}_{BT}\ \boldsymbol{\theta}_{BT}\ \boldsymbol{\theta}_{LS}\ \theta_{LE}\ \boldsymbol{\theta}_{RS}\ \theta_{RE})^T \rightarrow$$
$$(\mathbf{t}_{BT}\ \boldsymbol{\theta}_{BT}\ \mathbf{0}_{15}\ \boldsymbol{\theta}_{LS}\ \theta_{LE}\ \mathbf{0}_6\ \boldsymbol{\theta}_{RS}\ \theta_{RE}\ \mathbf{0}_{17})^T$$

B. Conversion Example 2

Here, we show how the MMM is mapped to the kinematic model of the humanoid robot ARMAR. The conversion covers the cases 1, 3, 4, and 5. Case 2 is only of interest for perception modules. The first problem is that ARMAR does not have the sternoclavicular joint, which is an example for case 5. One solution is to calculate the effective rotation matrix for the combination of the sternoclavicular and the shoulder joint. The effective rotation for the sternoclavicular and the shoulder joints is then formulated as:

$$\begin{aligned} R_{LS'} &= R_{X'Z'Y'}(\boldsymbol{\theta}_{LSC}) \cdot R_{X'Z'Y'}(\boldsymbol{\theta}_{LS}) \\ R_{RS'} &= R_{X'Z'Y'}(\boldsymbol{\theta}_{RSC}) \cdot R_{X'Z'Y'}(\boldsymbol{\theta}_{RS}), \end{aligned}$$

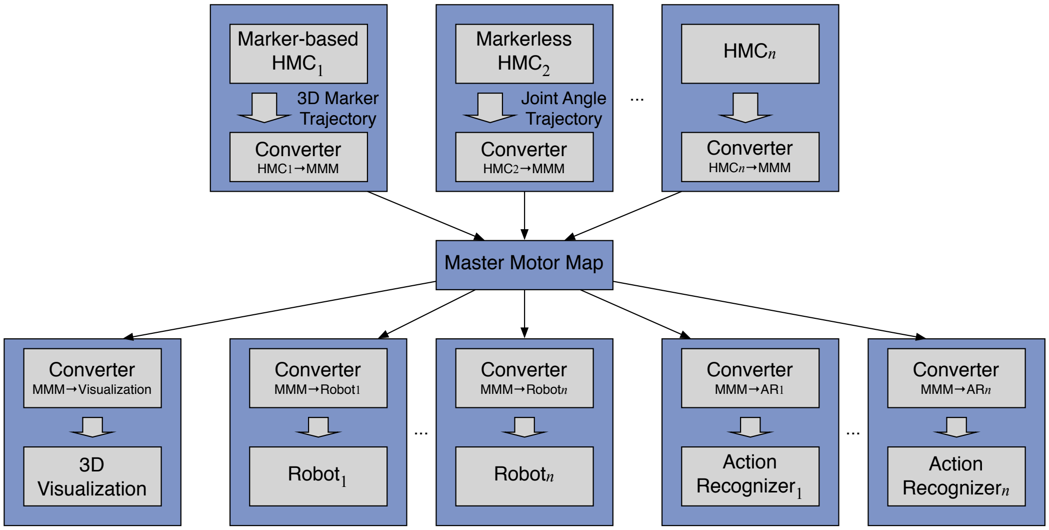

Fig. 5. Illustration of the proposed framework

where the notation $R(\boldsymbol{\theta})$ means $R(\alpha, \beta, \gamma)$ with $(\alpha\ \beta\ \gamma) = \boldsymbol{\theta}$. In the following, c denotes the side, which is necessary because ARMAR III has different coordinate systems for the two sides, in contrast to the MMM:

$$c := \begin{cases} 1 & : \text{ left} \\ -1 & : \text{ right} \end{cases}$$

Then, the kinematics for the shoulder joint of ARMAR III are defined as:

$$\begin{aligned} & R_{S,ARMAR}(c, \alpha, \beta, \gamma) \\ = \quad & R_y(c\tfrac{\pi}{6}) \cdot R_x(-\tfrac{\pi}{2}) \cdot R_z(-c\alpha) \cdot R_x(\beta) \cdot R_y(c\gamma) \\ = \quad & R_y(c\tfrac{\pi}{6}) \cdot R_x(-\tfrac{\pi}{2}) \cdot R_{Z'X'Y'}(-c\alpha, \beta, c\gamma) \end{aligned}$$

The problem of calculating the transformed rotation can be formulated as finding the angles α, β, γ so that:

$$\begin{aligned} R_{LS'} &= R_{S,ARMAR}(1, \alpha, \beta, \gamma) \\ R_{RS'} &= R_{S,ARMAR}(-1, \alpha, \beta, \gamma) \end{aligned}$$

and furthermore:

$$R_L := R_x(-\frac{\pi}{2})^T \cdot R_y(\frac{\pi}{6})^T \cdot R_{LS'} = R_{Z'X'Y'}(-\alpha, \beta, \gamma)$$
$$R_R := R_x(-\frac{\pi}{2})^T \cdot R_y(-\frac{\pi}{6})^T \cdot R_{RS'} = R_{Z'X'Y'}(\alpha, \beta, -\gamma)$$

In the following, we determine the solution for the left shoulder; the solution for the right shoulder can be determined analogously. First, we need the rotation matrix $R_{Z'X'Y'}(-\alpha, \beta, \gamma)$:

$$\begin{aligned} & R_{Z'X'Y'}(-\alpha, \beta, \gamma) \\ = & \begin{pmatrix} s\alpha s\beta s\gamma + c\alpha c\gamma & s\alpha c\beta & -s\alpha s\beta c\gamma + c\alpha s\gamma \\ c\alpha s\beta s\gamma - s\alpha c\gamma & c\alpha c\beta & -c\alpha s\beta c\gamma - s\alpha s\gamma \\ -c\beta s\gamma & s\beta & c\beta c\gamma \end{pmatrix} \\ = & \begin{pmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{pmatrix} = R_L \end{aligned}$$

The two possible solutions can then be calculated by:

$$\begin{aligned} \alpha &= \operatorname{atan2}(\pm r_2, \pm r_5) && (1) \\ \beta &= \operatorname{atan2}(r_8, \pm\sqrt{r_2^2 + r_5^2}) && (2) \\ \gamma &= \operatorname{atan2}(\mp r_7, \pm r_9) && (3) \end{aligned}$$

These can be disambiguated by taking into account maximum joint angle constraints. To formulate the complete conversion, we define for the left shoulder:

$$g_l : \mathbb{R}^6 \to \mathbb{R}^3 : \quad (\boldsymbol{\theta}_{LSC}\ \boldsymbol{\theta}_{LS}) \ \to (\alpha\ \beta\ \gamma)^T,$$

as given in the equations (1)-(3); g_r is defined analogously. The final conversion is then given by:

$$f_2 : \mathbb{R}^{52} \to \mathbb{R}^{14}$$
$$(\mathbf{t}_{RT}\ \boldsymbol{\theta}_{RT}\ \boldsymbol{\theta}_{SB}\ \boldsymbol{\theta}_{VC7}\ \boldsymbol{\theta}_{P}\ \boldsymbol{\theta}_{VT6}\ \boldsymbol{\theta}_{LSC}\ \boldsymbol{\theta}_{LS}\ \boldsymbol{\theta}_{LE}\ \boldsymbol{\theta}_{LW}$$
$$\boldsymbol{\theta}_{RSC}\ \boldsymbol{\theta}_{RS}\ \boldsymbol{\theta}_{RE}\ \boldsymbol{\theta}_{RW}\ \boldsymbol{\theta}_{LH}\ \theta_{LK}\ \boldsymbol{\theta}_{LA}\ \boldsymbol{\theta}_{RH}\ \theta_{RK}\ \boldsymbol{\theta}_{RA})^T \to$$
$$(g_l \begin{pmatrix} \boldsymbol{\theta}_{LSC}^T \\ \boldsymbol{\theta}_{LS}^T \end{pmatrix}^T \boldsymbol{\theta}_{LE}\ \boldsymbol{\theta}_{LW}\ g_r \begin{pmatrix} \boldsymbol{\theta}_{RSC}^T \\ \boldsymbol{\theta}_{RS}^T \end{pmatrix}^T \boldsymbol{\theta}_{RE}\ \boldsymbol{\theta}_{RW})^T$$

C. Conversion Example 3

This example is important for the programming and execution of manipulation tasks, which are specified in terms of object trajectories. Using a magnetic tracking system (Fasttrak, www.polhemus.com), both the position and the orientation of the hand $(x\ y\ z\ \alpha\ \beta\ \gamma)^T$ is tracked in Cartesian space. The mapping to the MMM is provided by a closed-form inverse kinematics algorithm, which computes the transformation for an arm:

$$f_3 : \mathbb{R}^6 \to \mathbb{R}^7 :$$
$$(x\ y\ z\ \alpha\ \beta\ \gamma)^T \to \ (\boldsymbol{\theta}_{LS}\ \boldsymbol{\theta}_{LE}\ \boldsymbol{\theta}_{LW})^T$$

In solving the inverse kinematics problem of the arm, the arm redundancy is used for the generation of human-like arm postures. The arm joint angles are reconstructed using a sensorimotor transformation model, which was found in physiological observation of human arm movements [18], [19]. The model maps the Cartesian wrist position to a natural arm posture using a set of representation parameters, which are the upperarm elevation, the forearm elevation, the upperarm yaw, and the forearm yaw. Once these parameters are obtained, the shoulder and elbow joint angles are calculated to match the hand position whereas the forearm and wrist joint angles are calculated to match the hand orientation (see [20]).

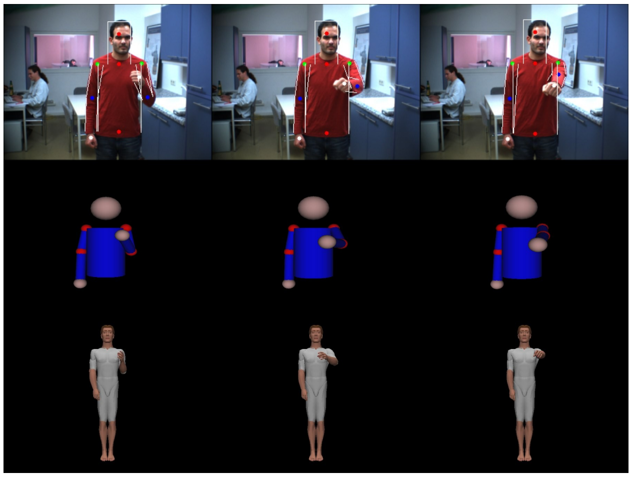

Fig. 6. Example frames. Top: Projected result from the HMC system. Middle: 3D Visualization with the HMC model. Bottom: 3D Visualization of the MMM representation.

V. Conclusion

We have presented a framework for perception, visualization, reproduction, and recognition of 3D human motion. In the core of our framework, we have defined a reference kinematic model – the Master Motor Map. We have showed that our approach performs well in practice by having built in a markerless human motion capture system, a 3D visualization of the MMM, and a module for converting trajectories given in the MMM to the kinematic model of the humanoid robot ARMAR III. Although the proposed framework is intended primarily to query the kinematics of 3D human motion, it can be also augmented with dynamic parameters of the body parts. Currently, we are working on reproducing movements on ARMAR III not only in simulation but on the real robot. Herefore, it is crucial to incorporate a self-collision detection module to avoid configurations which are outside the robot's working space. Furthermore, we are building up a database of movements in the MMM format, consisting of both markerless and marker-based human motion capture data, which will serve as a solid data source for applications related to imitation learning within the EU project PACO-PLUS, and hopefully also within in the entire robotics community.

Acknowledgment

The work described in this paper was partially conducted within the EU Cognitive Systems project PACO-PLUS (FP6-2004-IST-4-027657) and funded by the European Commission and the German Humanoid Research project SFB588 funded by the German Research Foundation (DFG: Deutsche Forschungsgemeinschaft).

References

[1] A. Safonova, N. S. Pollard, and J. K. Hodgins, "Optimizing human motion for the control of a humanoid robot." in *Symposium on Adaptive Motion of Animals and Machines (AMAM03)*, Kyoto, Japan, March 2003.

[2] N. S. Pollard and J. K. Hodgins, "Generalizing demonstrated manipulation tasks." in *Workshop on the Algorithmic Foundations of Robotics (WAFR02)*, Nice, France, December 2002.

[3] A. Ude, C. G. Atkeson, and M. Riley, "Programming Full-Body Movements for Humanoid Robots by Observation," *Robotics and Autonomous Systems*, vol. 47, pp. 93–108, 2004.

[4] M. Ruchanurucks, S. Nakaoka, S. Kudoh, and K. Ikeuchi, "Generation of humanoid robot motions with physical constraints using hierarchical b-spline," in *IEEE/RSJ International. Conference on Intelligent Robots and Systems*, Edmonton, Alberta, Canada, 2005, pp. 1869 – 1874.

[5] T. Inamura, Y. Nakamura, and I.Toshima, "Embodied symbol emergence based on mimesis theory," *International Journal of Robotics Research*, vol. 23, no. 4, pp. 363–377, 2004.

[6] L. Dongheui and Y. Nakamura, "Stochastic model of imitating a new observed motion based on the acquired motion primitives," in *IEEE/RSJ International. Conference on Intelligent Robots and Systems*, Beijing, China, 2006, pp. 4994 – 5000.

[7] "Vicon Peak," http://www.vicon.com.

[8] T. Beth, I. Boesnach, M. Haimerl, J. Moldenhauer, K. Bös, and V. Wank, "Characteristics in Human Motion – From Acquisition to Analysis," in *International Conference on Humanoid Robots (Humanoids)*, Karlsruhe/München, Germany, 2003.

[9] F. Caillette and T. Howard, "Real-Time Markerless Human Body Tracking with Multi-View 3-D Voxel Reconstruction," in *British Machine Vision Conference*, vol. 2, Kingston, UK, 2004, pp. 597–606.

[10] I. Mikic, M. Trivedi, E. Hunter, and P. Cosman, "Human Body Model Acquisition and Tracking using Voxel Data," *International Journal of Computer Vision*, vol. 53, no. 3, pp. 199–223, 2003.

[11] J. Deutscher, A. Blake, and I. Reid, "Articulated Body Motion Capture by Annealed Particle Filtering," in *Computer Vision and Pattern Recognition (CVPR)*, Hilton Head, USA, 2000, pp. 2126–2133.

[12] H. Sidenbladh, "Probabilistic Tracking and Reconstruction of 3D Human Motion in Monocular Video Sequences," Ph.D. dissertation, Royal Institute of Technology, Stockholm, Sweden, 2001.

[13] P. Azad, A. Ude, T. Asfour, G. Cheng, and R. Dillmann, "Image-based Markerless 3D Human Motion Capture using Multiple Cues," in *International Workshop on Vision Based Human-Robot Interaction*, Palermo, Italy, 2006.

[14] P. Azad, A. Ude, T. Asfour, and R. Dillmann, "A Full Body Human Motion Capture System using Particle Filtering and On-The-Fly Edge Detection," in *International Conference on Robotics and Automation (ICRA)*, Rome, Italy, 2007.

[15] S. Knoop, S. Vacek, and R. Dillmann, "Modeling Joint Constraints for an Articulated 3D Human Body Model with Artificial Correspondences in ICP," in *International Conference on Humanoid Robots (Humanoids)*, Tsukuba, Japan, 2005.

[16] J. J. Craig, *Introduction to Robotics: Mechanics and Control*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1989.

[17] H. A. W. Group, "H-anim 1.1 specification," http://h-anim.org.

[18] J. F. Soechting and M. Flanders, "Sensorimotor Representations for Pointing to Targets in Three-Dimensional Space," *Journal of Neurophysiology*, vol. 62, no. 2, pp. 582–594, 1989.

[19] ——, "Errors in Pointing are Due to Approximations in Targets in Sensorimotor Transformations," *Journal of Neurophysiology*, vol. 62, no. 2, pp. 595–608, 1989.

[20] T. Asfour and R. Dillmann, "Human-like Motion of a Humanoid Robot Arm Based on Closed-Form Solution of the Inverse Kinematics Problem." in *The IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)*, Las Vegas, USA, 27-31 October, 2003.