Project no.:	**027657**
Project full title:	**Perception, Action & Cognition through learning of Object-Action Complexes**
Project Acronym:	**PACO-PLUS**
Deliverable no.:	**D8.2.3**
Title of the deliverable:	**Real-time imitation learning system using an active humanoid head**

Contractual Date of Delivery to the CEC:	31. January 2009	
Actual Date of Delivery to the CEC:	03 Febrauary 2009	
Organisation name of lead contractor for this deliverable:	UniKarl	
Author(s):	Tamim Asfour, Aleš Ude, Pedram Azad, and Rüdiger Dillmann	
Participant(s):	UniKarl, JSI	
Work package contributing to the deliverable:	WP8.2	
Nature:	R/D	
Version:	Draft	
Total number of pages:	5	
Start date of project:	1st Feb. 2006	**Duration:** 48 month

Project co-funded by the European Commission within the Sixth Framework Programme (2002–2006)
Dissemination Level

PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Abstract:

In this report, we present a real-time system for imitation of human motion on the humanoid robot ARMAR-III and techniques for reaching using 3-D vision on the humanoid robot CBi. The imitation system can perceive articulated upper body motion in real-time and reproduce captured movements on a humanoid robot system online. For this purpose, captured joint angle trajectories are mapped via the Master Motor Map to the kinematics of the robot using non-linear optimization. The processing rate for perception and reproduction amounts to approx. 15 Hz in total. For reaching, the parameters of the robot's eyes under motion are updated to enable the use of 3-D vision to actively track objects using the head and eye degrees of freedom. The reaching behavior is encoded as a dynamic movement primitive, which was learned from a single demonstration.

Keyword list: Markerless human motion capture, Imitation on a humanoid robot, Reaching using 3-D active vision

Table of Contents

1. Executive Summary

In this report, we present a real-time system for imitation of human motion combined with 3-D active vision on humanoid robots. Note that the associated scientific contributions are part of the work done in WP2 and WP3. In this deliverable, experimental results of the developed techniques are presented.

1.1 Imitation Learning System on ARMAR-III

The imitation system system is based on two approaches, which are presented in the following:

- Stereo-based markerless human motion capture
- Online reproduction of observed movements

The system can capture and reproduce human motion online with a processing rate of approx. 15 Hz. References to related external work are given in the references section of each attached paper.

1.1.1 Stereo-based Markerless Human Motion Capture

The attached paper [A] presents an extension of the approach presented in [2] (see also D3.1.1, Section 4, from the last report period). The focus was on the acquisition of more accurate and smoother human trajectories while increasing robustness, in particular for online application at lower frame rates. For this purpose, the underlying problems for typical noisy estimation and local minima when using particle filters for human motion capture were examined. In the work presented in [A], these problems and their solutions are presented, which are summarized in the following:

- **Prioritized fusion method:** The gradient cue and the distance cue, as presented in [2], often hinder each other in practice, when being fused with the conventional fusion method. Therefore, a prioritized fusion method has been developed, which gives the distance cue the higher priority and activates the gradient cue only in case the distance constraint is fulfilled.
- **Adaptive noise:** Often a static amount of noise is applied in the sampling step of the particle filter. Instead we apply, for each arm, independent adaptive noise that depends on the overall edge error for that arm. Experiments reveal that not only the estimated trajectories become smoother, but also a finer search can be performed in the vicinity of the true configuration, leading to more accurate results.
- **Adaptive shoulder position:** For the purpose of markerless human motion capture, the state-of-the-art approach is to use a simplified 3-D human model with a single ball joint for each shoulder and a static shoulder position. However, such a simplified model is far away from reflecting the nature of the human shoulder joint and thus does not allow proper alignment for the arms in many cases. Therefore, the estimation of the shoulder position is incorporated into the particle filter for estimating the arm configuration.
- **Incorporation of redundant inverse kinematics problem:** A typical problem with tracking approaches are local minima and recovery once tracking has got lost. A related problem is the robust application of tracking approaches at low frame rates, where the de-facto benefits of temporal information is limited. To tackle these problems, the redundant inverse kinematics problem, where the redundant degree of freedom is the elbow, is incorporated into the sampling step of the particle filter.

1.1.2 Online Imitation of Observed Movements

The attached paper [B] presents a method for imitation of human motion on a humanoid robot. For this purpose, motion trajectories are first mapped to an intermediate representation: the Master Motor Map (MMM), as specified in [1]. Using the MMM representation, a unifying interface is implemented, which allows the connection between various human capture systems and a humanoid robot. In order to enable goal-directed movements, the MMM representation is extended by a designated TCP target position corresponding to the already specified joints. However, due to differences in the kinematic structure of the human and the robot, a one-to-one mapping from the MMM to the robot would lead to an inaccurate reproduction of the captured motion. The first major issue is due to the fact that not all joints described by the MMM model are available on the robot. Therefore, a sole mapping of the existing robot joints, while disregarding the missing ones, would fail. Furthermore, additional constraints given by the mechanics of the robot must be taken into account in order to enable generating feasible robot motion.

In order to achieve feasible reproduction of observed human motion, that features goal-directedness as well as human-likeness an approach was developed that transforms the motion trajectories given in the MMM representation to the kinematics of ARMAR III. By solving a constrained optimization problem, which incorporates the designated TCP position and the joint angle configuration of the observed human motion, a compensation of the missing joints and prevention of joint constraint violations are attained. As a result, one obtains robot joint angle configurations that allow to place the robot's end-effector close to given target positions, while achieving maximum similarity between the robot motion and the human motion. The optimization problem is solved using the Levenberg-Marquardt algorithm with respect to the above-mentioned requirements.

1.2 Reaching Using Active 3-D Vision

In D2.1.4 we described a computational process that can be utilized to update the parameters of the robot's eyes under motion, which enables the use of 3-D vision on an active humanoid head. This system was utilized to realize reaching using active 3-D vision. In the attached video the reaching behavior was encoded as a dynamic movement primitive, which was learned from a single demonstration. By introducing an appropriate scaling into the nonlinear part of the movement primitive, the system could control the trade-off between reaching the goal position and fitting the original example movement. In this way the robot executed a reaching movement close to the original movement for goal positions in the neighborhood of the example goal position, but could still reach the final position in a given time for reaching positions further away from the example position. Based on this research we developed a motion generalization system which enables the learning of reaching movements from a larger library of example movements. This system is explained in D2.3.1.

2. Attached Videos

2.1 Online imitation of human motion on ARMAR-III

The video **HumanMotionImitiationOnARMAR.avi** first shows the visualization of the improved markerless human motion capture system (see Section 1.1.1), which is running in real-time. In the second part of the video, the reproduced motion after applying the presented mapping approach (see Section 1.1.2) can be seen. Note that the complete chain, from acquisition to imitation, is running online in the video at a processing frame rate of 15 Hz.

2.2 Reaching Using Active 3-D Vision

In the movie **CBiGrasp.mpg**, the robot is actively tracking the object using the head and eye degrees of freedom. The object position is estimated in the robot base coordinate system and the dynamic movement primitive is instantiated using this estimate. Grasping is implemented as a power grasp and is initiated when the robot arm reaches its final configuration.

Attached Papers

[A] P. Azad, T. Asfour, and R. Dillmann. Robust Real-time Stereo-based Markerless Human Motion Capture. In *IEEE/RAS International Conference on Humanoid Robots*, Daejeon, Korea, 2008.

[B] M. Do, P. Azad, T. Asfour, and R. Dillmann. Imitation of Human Motion on a Humanoid Robot using Nonlinear Optimization. In *IEEE/RAS International Conference on Humanoid Robots*, Daejeon, Korea, 2008.

References

[1] P. Azad, T. Asfour, and R. Dillmann. Toward an Unified Representation for Imitation of Human Motion on Humanoids. In *IEEE International Conference on Robotics and Automation*, pages 2558–2563, Roma, Italy, 2007.

[2] P. Azad, A. Ude, T. Asfour, and R. Dillmann. Stereo-based Markerless Human Motion Capture for Humanoid Robot Systems. In *IEEE International Conference on Robotics and Automation*, pages 3951–3956, Roma, Italy, 2007.

Robust Real-time Stereo-based Markerless Human Motion Capture

Pedram Azad, Tamim Asfour, Rüdiger Dillmann

University of Karlsruhe, Germany azad@ira.uka.de, asfour@ira.uka.de, dillmann@ira.uka.de

***Abstract*— The main problem of markerless human motion capture is the high-dimensional search space. Tracking approaches therefore utilize temporal information and rely on the pose differences between consecutive frames being small. Typically, systems using a pure tracking approach are sensitive to fast movements or require high frame rates, respectively. However, on the other hand, the complexity of the problem does not allow real-time processing at such high frame rates. Furthermore, pure tracking approaches often only recover by chance once tracking has got lost. In this paper, we present a novel approach building on top of a particle filtering framework that combines an edge cue and 3D hand/head tracking in a distance cue for human upper body tracking, as proposed in our earlier work. To overcome the mentioned deficiencies, the solutions of an inverse kinematics problem for a – in the context of the problem – redundant arm model are incorporated into the sampling of particles in a simplified annealed particle filter. Furthermore, a prioritized fusion method and adaptive shoulder positions are introduced in order to allow proper model alignment and therefore smooth tracking. Results of real-world experiments show that the proposed system is capable of robust online tracking of 3D human motion at a frame rate of 15 Hz. Initialization is accomplished automatically.**

I. Introduction

Markerless human motion capture means to capture human motion without any additional arrangements required, by operating on image sequences only. Commercial human motion capture systems such as the VICON system (`www.vicon.com`), which are popular in the film industry as well as in the biological research field, require reflective markers and time consuming manual post-processing of captured sequences. In contrast, a real-time markerless human motion capture system using the image data acquired by the robot's head would allow online imitation-learning in a natural way. Another application for the data computed by such a system is the recognition of human actions and activities, serving as a perception component for human-robot interaction.

For application on an active head of a humanoid robot, a number of restrictions has to be coped with. In addition to the limitation to two cameras positioned at approximately eye distance, one has to take into account that an active head can move. Furthermore, computations have to be performed in real-time, and most importantly for practical application, the robustness of the tracking must not depend on a high frame rate or slow movements, respectively.

In the following, a short overview of approaches to markerless human motion capture that are relevant for application on humanoid robot systems is given. Approaches operating on 3D data either extend the ICP algorithm for application to articulated object tracking ([1], [2]) or utilize an optimization method based on 3D-3D correspondences [3]. The 3D point clouds used as input are either acquired by disparity maps or a 3D sensor is used such as the SwissRanger (`www.mesa-imaging.ch`). Image-based approaches are either search-based ([4], [5]), utilize an optimization approach based on 2D-3D correspondences [6], [7], [8] (resp. [9] for articulated hand tracking), or are based on particle filtering. In [10], it was shown that human motion can be successfully tracked with particle filtering, using three cameras positioned around the scene of interest. In [11], it was shown that with the same principles, 3D human motion can be estimated from monocular image sequences to some degree, when learning a motion model. Recently, we have proposed the incorporation of stereo-based 3D hand/head tracking for an additional distance cue in [12]. In [13], in addition, a certain percentage of the particles is sampled with a Gaussian distribution around a single solution computed by an analytical inverse kinematics method for the purpose of re-initialization. Taking into account *all* relevant solutions of the inverse kinematics problem is not considered.

In Section II, the basic components of the used particle filtering framework are introduced, namely the utilized 3D human model and the used visual cues. The proposed approach consisting of the components hierarchical search, a prioritized fusion method, adaptive noise in sampling, adaptive shoulder positions, and the incorporation of the solutions of an inverse kinematics problem is presented in the Sections III–VII. The results of real-world experiments are presented in Section VIII, ending with a conclusion in Section IX.

II. Basic Components

A. Human Upper Body Model

In the proposed system, a kinematics model of the human upper body consisting of 14 DoF is used, not modeling the neck joint. The shoulder is modeled as a ball joint with 3 DoF, and the elbow as a hinge joint with 1 DoF. Additional 6 DoF are used for the base transformation. With this model, rotations around the axis of the forearm cannot be modeled. Capturing the forearm rotation would require tracking of the hand, which is regarded as a separate problem.

The shoulder joints are implemented with an axis/angle representation in order to avoid problems with singularities, which can occur when using Euler angles. The base rotation is modeled by Euler angles to allow a better imagination so that joint space restrictions can be defined easily. For the geometric model, the body sections are fleshed out by sections of a cone with circular cross-sections.

B. Image Processing Pipeline

The image processing pipeline transforms each input image pair into a binarized skin color image pair $I_{s,l}, I_{s,r}$ and a gradient image pair $I_{g,l}, I_{g,r}$, which are used by the likelihood functions presented in Section $II-C$. In Fig. 1, the input and outputs for a single image are illustrated. This pipeline is applied twice: once for the left and once for the right camera image. In order to allow for ego-motion, figure-ground segmentation is performed by shirt color segmentation, which is only needed for distinguishing edges that belong to the person's contour from edges belonging to the background. Details are given in [14].

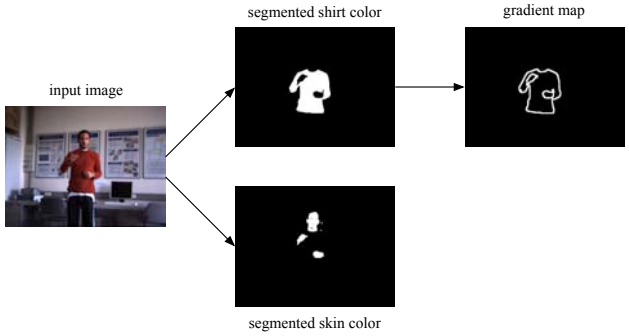

Fig. 1. Illustration of the input and outputs of the image processing pipeline.

C. Cues

In the following, the cues that are used in the proposed system are presented. The formulations are given for a single image; their application to stereo image pairs is explained in Section IV.

1) Edge Cue: According to [10], for the edge cue, the gradient values from the gradient image I_g are summed up along the projected model contours. Assuming that the gradient image has been remapped to the interval $[0,\,1]$, the evaluation function is defined as:

$$w_g(I_g, P) = 1 - \frac{1}{|P|}\sum_{i=1}^{|P|} I_g(\boldsymbol{p}_i)\,, \tag{1}$$

where P denotes the set of sampled 2D contour points. Note that compared to [10], squaring is omitted, which turned out not to have any significant effect. The likelihood function reads:

$$p_g(I_g \,|\, \boldsymbol{s}) \propto \exp\left\{-\frac{1}{2\sigma_g^2} w_g(I_g, f_g(\boldsymbol{s}))\right\}\,, \tag{2}$$

where the function f_g computes the set of sampled 2D points P for a given model configuration $\boldsymbol{s}$.

2) Distance Cue: According to [12], the distance cue evaluates the squared distances between distinct model points and their absolute 3D measurements in the current stereo pair. In the proposed system, the hands and the head of the person are used as such points, which are tracked by a separate hand/head tracking system. The evaluation function of the distance cue is defined as follows:

$$w_d(I_d, P) = \sum_{i=1}^{|P|} |\boldsymbol{p}_i - \boldsymbol{p}_i'(I_d)|^2\,, \tag{3}$$

where $P = \{\boldsymbol{p}_i\}$ denotes a set of transformed model points and $\boldsymbol{p}_i'(I_d)$ their measured positions that have been computed on the basis of the observations I_d. In the proposed system, these observations are the skin color segmentation results $I_{s,l}, I_{s,r}$ (see Section II-B). In order to apply this evaluation function for tracking, model points must be transformed into the coordinate system the measurements are accomplished in, yielding the point set P. For this purpose, the transformation $f_{d,i} : R^{\dim(\boldsymbol{s})} \rightarrow R^{\dim(\boldsymbol{p}_i)}$ is used, which maps a certain model point $\boldsymbol{p}_{m,i}$ to the coordinate system of the corresponding measured point $\boldsymbol{p}_i'$, given a model configuration $\boldsymbol{s}$. The function f_d performs this transformation $f_{d,i}$ for each desired model point and thereby computes the point set P. Finally, the likelihood function p_d can be formulated as follows:

$$p_d(I_d \,|\, \boldsymbol{s}) \propto \exp\left\{-\frac{1}{2\sigma_d^2} w_d(I_d, f_d(\boldsymbol{s}))\right\}\,. \tag{4}$$

III. Hierarchical Search

The most general approach is to use one particle filter for estimating all degrees of freedom of the model, as done in our earlier work ([12], [14]). The advantage is that by estimating all degrees of freedom simultaneously, potentially the orientation of the torso can be estimated as well. In practice, however, the human model is not precise enough to benefit from this potential – if the sensor system is restricted to a single stereo camera system.

To reduce the number of particles, a hierarchical search is performed i.e. the search space is partitioned explicitly. Since the head is tracked for the distance cue anyway, the head's position can be used as the root of the kinematic chain. By doing this, only 3 DoF of the base transformation remain to be estimated. If not modeling the neck joint, these degrees of freedom describe the orientation of the torso. Since the torso orientation can hardly be estimated on the basis of 2D measurements only, it is regarded as a separate problem. In order to achieve robustness to small changes of the body rotation without actually knowing it, the shoulder positions are modeled to be adaptive, as will be described in Section VI.

With static shoulder positions (relative to the head), the final estimation problem for the particle filter would consist of 4 DoF for each arm; the 3 DoF of the base translation are

estimated directly by a separate particle filter used for head tracking. Intuitively, estimating the 4 DoF of one arm with a separate particle filter sounds simple and one would assume that this approach would lead to an almost perfect result – given the restriction of a more or less frontal view of the person. However, various extensions are necessary to allow smooth and robust tracking of arm movements, which will be introduced in the following sections.

IV. Prioritized Fusion

The conventional approach for combining several cues within a particle filtering framework is to multiply the results of the respective likelihood functions. The quality and accuracy achieved by such an approach strongly depends on the cues agreeing on the way to the target configuration. In practice, however, different cues have different characteristics. While the likelihood functions of different cues in general have their global maximum in the vicinity of the true configuration, i.e. agree on the final goal, they often exhibit totally different local maxima. This circumstance often causes the likelihood functions to fight against each other, resulting in a typically noisy estimation.

In the proposed system, the edge cue and the distance cue have to be fused. Since the distance cue is the more reliable cue due to the explicit measurement of the 3D head and hand position, the idea is to introduce a prioritization scheme: If the distance error of the hand for the current estimation is above a predefined threshold, then the error of the edge cue is ignored by assigning the maximum error of 1; otherwise the distance error is set to zero. By doing this, the particle filter rapidly approaches configurations in which the estimated hand position is within the predefined minimum radius of the measured hand position – without being disturbed by the edge cue. All configurations that satisfy the hand position condition suddenly produce a significantly smaller error, since the distance error is set to zero and the edge error is < 1. Therefore, within the minimum radius, the edge cue can operate undisturbedly. Applying this fusion approach allows the two cues to act complementary instead of hindering each other.

Algorithm 1 ComputeLikelihoodArm($I_{g,l}$, $I_{g,r}$, $\boldsymbol{p}_h$, $\boldsymbol{s}$) $\rightarrow \pi$

1) $e_g := \dfrac{w_g(I_{g,l}, f_{g,l}(\boldsymbol{s})) + w_g(I_{g,r}, f_{g,r}(\boldsymbol{s}))}{2}$
2) $e_d := |\boldsymbol{p}_h - f_d(\boldsymbol{s})|^2$
3) If $e_d < t_d^2$ then set $e_d := 0$ else set $e_g := 1$.
4) $e_d := \dfrac{s_d \cdot e_d}{e_d^{(t-1)}}$
5) If $e_d > 50$ then set $e_d := 50$.
6) $\pi := \exp\{-(e_d + s_g \cdot e_g)\}$

In addition, the range of the distance error is limited by division by the distance error $e_d^{(t-1)}$ for the estimated configuration of the previous frame. Otherwise the range of the distance error could become very large in some cases, potentially leading to numerical instabilities. Finally, the argument to the exponential function is cut off when it exceeds the value 50. The final likelihood function fusing the errors calculated by the edge cue and the distance cue is summarized in Algorithm 1. The inputs to the algorithm are the gradient map stereo pair $I_{g,l}, I_{g,r}$, the measured hand position $\boldsymbol{p}_h$, and the configuration $\boldsymbol{s}$ to be evaluated. For the weighting factors $s_g := \frac{1}{2\sigma_g^2}$ and $s_d := \frac{1}{2\sigma_d^2}$, $s_g = s_d = 10$ is used. As the minimum radius, $t_d = 30\,\text{mm}$ is used. The function f_d computes the 3D position of the hand for a given joint configuration $\boldsymbol{s}$ using the forward kinematics.

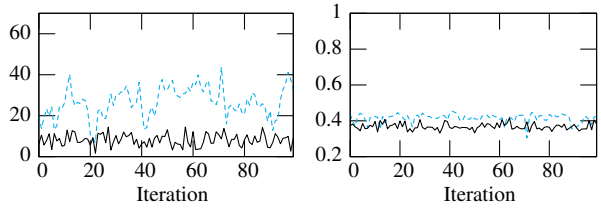

Fig. 2. Illustration of the effect of the proposed fusion method on the overall edge and distance error for a typical situation. The solid line indicates the result computed using the proposed fusion method, the dashed line using conventional fusion. Left: Euclidean distance error in [mm]. Right: edge error.

In Fig. 2, the results of 100 iterations of the particle filter are plotted, after the particle filter has already converged. As can be seen, using prioritized fusion does not only lead to smaller edge and distance errors, but the variances are also considerably smaller. The reason is that the cues do not agree on the same goal and thus cannot find the optimal configuration when using the conventional fusion method.

V. Adaptive Noise

In [15], the idea was raised to not apply a constant amount of noise for sampling new particles, but to choose the amount to be proportional to the variance of each parameter. Since the variance of a parameter is not necessarily related to an error of the parameter itself, we choose the amount for all degrees of freedom of an arm to be proportional to the current overall edge error of that arm. In Fig. 3, the overall errors are plotted for the same example as used for Fig. 2, comparing the application of adaptive noise to constant noise. In both cases, prioritized fusion was applied (see Section IV).

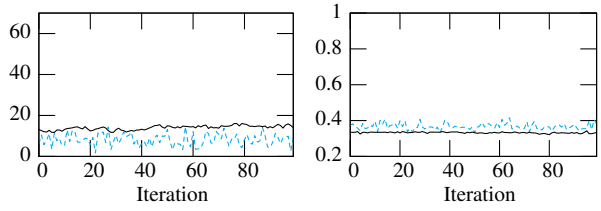

Fig. 3. Illustration of the effect of adaptive noise on the overall edge and distance error for a typical example. The solid line indicates the result computed using adaptive noise, the dashed line using a constant amount of noise. Left: Euclidean distance error in [mm]. Right: edge error.

Note that not only the standard deviation of the estimated trajectory is lower by a factor of approx. 2–3 – which is reasonable when reducing the amount of noise – but also the edge error exhibits a lower magnitude, compared to the application of a constant amount of noise. This means that the particle filter could find a better goal configuration when applying adaptive noise. The reason is that when applying a constant amount of noise, the amount must be chosen to be relatively high in order to cope with (unpredictable) motion. In the vicinity of the true configuration, however, this amount is too high to allow a fine search, whereas adaptive noise allows to search with a higher resolution in a smaller subspace. The reason for the slightly higher distance error is that the prioritized fusion method with $t_d = 30\,\text{mm}$ in Algorithm 1 gives the configurations the freedom to produce any distance error smaller than 30 mm. If desired, t_d could be chosen to be smaller. However, this would lead to less robustness to the effects of clothing, and in particular to loose sleeves.

VI. Adaptive Shoulder Position

In general, one of the main problems with real image data is that the model does not perfectly match the observations. In the case of motion capture of the upper body, the problem often occurs for the shoulder joint, which is usually approximated by a single ball joint, the glenohumeral joint. In reality, however, the position of this ball joint depends on two other shoulder joints, namely the acromioclavicular joint and the sternoclavicular joint. When not modeling these joints, the upper body model is too stiff to allow proper alignment; an exemplary situation is shown for the person's right arm in Fig. 4. Even more problematic situations occur, when the arm is moved to the back.

Fig. 4. Illustration of the effect of adaptive shoulder positions. The main difference can be observed for the person's right arm; the model edges cannot align with the image edges when using a static shoulder position, since the shoulder position is too much inside. The white dots indicate joint positions of the model, black dots mark the positions of the head and the hands of the model, and red dots mark the respective measured positions. Left: static shoulder position. Right: adaptive shoulder position.

In the proposed system, this problem becomes even more severe, since the shoulder positions are inferred by the head position, assuming a more or less frontal view. Our solution is to estimate the shoulder position within the particle filter of the arm, i.e. going from 4 DoF to 7 DoF. As it turns out, the higher dimensionality does not lead to any practical problems, whereas the freedom of the shoulder positions for aligning the model results in a significantly more powerful system.

The three additional degrees of freedom define a translation in 3D space. The limits are defined as a cuboid, i.e. by $[x_{min},\, x_{max}] \times [y_{min},\, y_{max}] \times [z_{min},\, z_{max}]$. The right image from Fig. 4 shows the improvement in terms of a better alignment of the person's right arm achieved by the adaptive shoulder position. As can be seen, the right shoulder has been moved slightly outwards in order to align the contour of the model with the image edges. Furthermore, the shoulder has been moved downwards so that the distance error is within the minimum radius, allowing the edge cue to operate undisturbedly.

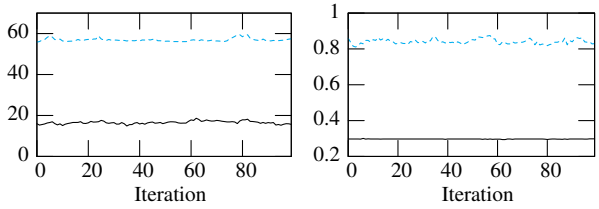

Fig. 5. Illustration of the effect of adaptive shoulder positions on the overall edge and distance error by the example of the person's right arm shown in Fig. 4. The solid line indicates the result computed using an adaptive shoulder position, the dashed line using a static shoulder position. Left: Euclidean distance error in [mm]. Right: edge error.

In Fig. 5, the overall errors are plotted for the person's right arm shown in Fig. 4, comparing a static to an adaptive shoulder position. As can be seen, both errors are significantly lower when modeling the shoulder position to be adaptive. The reason for the lower distance error is that the shoulder joint could move downwards so that the hand of the model can approach the hand in the image. The lower edge error is more significant: In the case of a static shoulder position, the edge error could not be minimized at all, while the adaptive shoulder position allows practically perfect alignment.

VII. Incorporating Inverse Kinematics

The system which has been presented so far performs well and can acquire smooth and accurate trajectories. The success of the tracker, however, depends on the speed of the person's movements with respect to the frame rate of the camera. This is typical for all pure tracking approaches, since they rely on the differences between consecutive frames being small. This leads to the main problem that once tracking has got lost, in general, tracking systems only recover by chance. The inclusion of the measured head and hand positions in the proposed system already leads to a considerable improvement, since the distance cue allows comparatively fast and reliable recovery.

One problem that remains are local minima. A typical situation is the automatic initialization of the tracking system. Here, the configuration must be found without the aid of temporal information. An example of such a local minimum is shown for the person's right arm in Fig. 6. Another problematic situation occurs when the arm is almost fully extended. In this

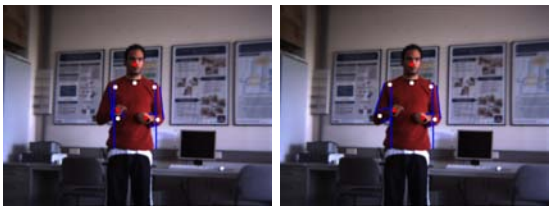

Fig. 6. Illustration of the effect of incorporating inverse kinematics. Left: without inverse kinematics. Right: with inverse kinematics.

case, one of the 3 DoF of the shoulder – namely the rotation around the upper arm – cannot be measured due to the lack of available information. Problems now occur when the person starts to bow the elbow, since the system cannot know at this point, in which direction the hand will move to. If the guess of the system is wrong, then the distance between the true configuration and the state of the particle filter can suddenly become very large and tracking gets lost.

In order to overcome these problems, the redundant inverse kinematics of the arm are incorporated into the sampling step of the particle filter. Given a 3D shoulder position $\boldsymbol{s}$, a 3D hand position $\boldsymbol{h}$, the length of the upper arm a, and the length of the forearm b, the set of all possible arm configurations is described by a circle on which the elbow can be located. The position of the elbow on this circle can be described by an angle α. Algorithm 2 analytically computes for a given angle α the joint angles $\theta_1, \theta_2, \theta_3$ for the shoulder and the elbow angle θ_4. The rotation matrix R_b denotes the base rotation from the frame the shoulder position $\boldsymbol{s}$ was measured in. Since the computations assume that the base rotation is zero, the shoulder position $\boldsymbol{s}$ and the hand position $\boldsymbol{h}$ are rotated back with the inverse rotation R_b at the beginning. The underlying geometric relationships are illustrated in Fig. 7.

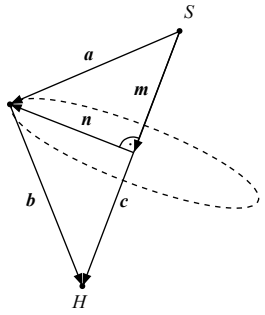

Fig. 7. Illustration of the geometric relationships for the inverse kinematics computations.

The general idea of the used inverse kinematics method is as follows. The starting point is the calculation of the vector $\boldsymbol{m}$, which points from the shoulder position to the center of the circle. Subsequently, for each α a vector $\boldsymbol{n}$ is calculated that points from the center to the position of the elbow. Then, one possible rotation matrix R_e for the shoulder joint is calculated that moves the elbow to the computed position. For this rotation matrix, the rotation matrix $R_y(\varphi)$ for the rotation around the upper arm is calculated that satisfies the hand constraint. The final rotation matrix R for the shoulder joint satisfying both the elbow and the hand constraint is composed of the rotations R_e and $R_y(\varphi)$. The elbow angle θ_4 is given by $\gamma - \pi$, where γ is the angle between $-\boldsymbol{a}$ and $\boldsymbol{b}$ (see Fig. 7), since $\theta_4 \leq 0$ and for a fully extended arm, it is $\theta_4 = 0$.

In order to take into account joint constraints, not all possible vectors $\boldsymbol{n}$ are considered, but only a subset. For this, the shoulder rotation that is necessary for bringing the hand to the target position $\boldsymbol{c}$ is reproduced in a defined way. Since the computation of this rotation is ambiguous and a defined elbow position is desired, the rotation is decomposed into two single rotations. The first rotation moves the hand to the proper position in the sagital plane (yz), the second rotation finally moves the hand to the target position. By applying the same two rotations to the vector $(0,\, 0,\, -a \sin\beta)^T$, which defines $\boldsymbol{n}$ in a canonical way, the vector $\boldsymbol{n}_0$ is calculated as a reference. For this $\boldsymbol{n}_0$, according to human-like joint constraints, plausible values for the bounds of $\alpha \in [\alpha_{min},\, \alpha_{max}]$ are $\alpha_{min} = -0.2,\ \alpha_{max} = \pi$ for the left arm, and $\alpha_{min} = -\pi,\ \alpha_{max} = 0.2$ for the right arm, respectively.

Algorithm 2 ComputeInverseKinematics(R_b, $\boldsymbol{s}$, $\boldsymbol{h}$, a, b, α) $\rightarrow$ $\theta_1, \theta_2, \theta_3, \theta_4$

1) $\boldsymbol{c} := R_b^T(\boldsymbol{h} - \boldsymbol{s})$
2) If $|\boldsymbol{c}| > 0.95\,(a+b)$ then set $\boldsymbol{c} := 0.95\,(a+b)\dfrac{\boldsymbol{c}}{|\boldsymbol{c}|}$.
3) If $|\boldsymbol{c}| < |a-b|$ then set $\boldsymbol{c} := |a-b|\dfrac{\boldsymbol{c}}{|\boldsymbol{c}|}$.
4) $c := |\boldsymbol{c}|$
5) $\beta := \arccos\dfrac{a^2 + c^2 - b^2}{2ac}$
6) $\gamma := \arccos\dfrac{a^2 + b^2 - c^2}{2ab}$
7) $\boldsymbol{u}_1 := (0,\, c,\, 0)^T$
8) $\boldsymbol{u}_2 := (0,\, c_y,\, \mathrm{sign}(c_z)\sqrt{c_x^2 + c_z^2})^T$
9) $\boldsymbol{n}_0 := \mathrm{Rotate}((0,\, 0,\, -a\sin\beta)^T,\, (1,\, 0,\, 0)^T,\, \mathrm{Angle}(\boldsymbol{u}_1,\, \boldsymbol{u}_2,\, (1,\, 0,\, 0)^T))$
10) $\boldsymbol{n}_0 := \mathrm{Rotate}(\boldsymbol{n}_0,\, (0,\, 1,\, 0)^T,\, \mathrm{Angle}(\boldsymbol{u}_2,\, \boldsymbol{c},\, (0,\, 1,\, 0)^T))$
11) $\boldsymbol{n} := \mathrm{Rotate}(\boldsymbol{n}_0,\, \boldsymbol{c},\, \alpha)$
12) $\boldsymbol{m} := \dfrac{\boldsymbol{c}}{|\boldsymbol{c}|} a\cos\beta$
13) $\boldsymbol{a} := \boldsymbol{m} + \boldsymbol{n}$
14) $\boldsymbol{b} := \boldsymbol{c} - \boldsymbol{a}$
15) $\boldsymbol{u}_1 := (0,\, 1,\, 0)^T$
16) $\boldsymbol{u}_2 := \dfrac{\boldsymbol{a}}{|\boldsymbol{a}|}$
17) $R_e := \mathrm{RotationMatrixAxisAngle}(\boldsymbol{u}_1 \times \boldsymbol{u}_2,\, \mathrm{Angle}(\boldsymbol{u}_1,\, \boldsymbol{u}_2,\, \boldsymbol{u}_1 \times \boldsymbol{u}_2))$
18) $\varphi := \mathrm{Angle}(R_e \cdot R_x(\gamma - \pi) \cdot (0,\, b,\, 0)^T,\ \boldsymbol{b},\ \boldsymbol{a})$
19) $R := R_e \cdot R_y(\varphi)$
20) $(\theta_1,\, \theta_2,\, \theta_3) := \mathrm{GetAxisAngle}(R)$
21) $\theta_4 := \gamma - \pi$

Finally, the inverse kinematics method must be incorporated into the sampling step of the particle filter. For this purpose, the general idea of annealed particle filtering [10] is exploited,

which is running the particle filter several times on the same frame while adapting the parameters for each run in a suitable way in order to support faster convergence. In [10], the adapted parameter was the weighting factor for the evaluation function, with which the broadness of the resulting probability distribution can be modified.

A naive approach would be to apply the inverse kinematics for sampling all particles of the first run. Doing this would reset the complete state of the particle filter, including the elimination of all hypotheses, which are stored in the probability distribution. To keep the characteristics and benefits of a particle filter, only a certain percentage of the particles is sampled according to the inverse kinematics; all other particles are sampled in the conventional way. By doing this, new particles created by the inverse kinematics sampling get the chance to establish themselves, while particles with great likelihoods from the last generation, i.e. frame, can survive according to the particle filtering principle. For each frame, we use one such mixed run, followed by three normal runs of the particle filter. These additional runs allow the particle filter to sort out weak particles from the inverse kinematics sampling and to converge to a representative probability distribution. In the first run, 60% of the particles are sampled according to the inverse kinematics, while the other 40% are sampled in the conventional way. In Algorithm 2, the hand position $\boldsymbol{h}$ is measured by hand tracking, and for the shoulder position $\boldsymbol{s}$, the estimated shoulder position offset from the previous frame is applied to the measured head position.

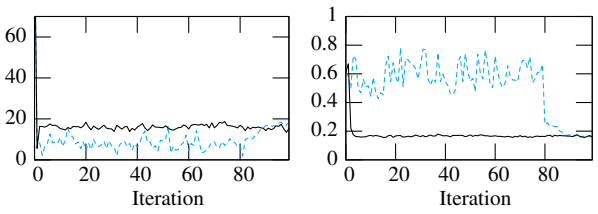

Fig. 8. Illustration of the effect of inverse kinematics sampling on the overall edge and distance error by the example of the person's right arm shown in Fig. 6. The solid line indicates the result computed with inverse kinematics sampling, the dashed line without. Left: Euclidean distance error in [mm]. Right: edge error.

In Fig. 8, the overall errors are plotted for the person's right arm shown in Fig. 6, comparing conventional sampling to sampling taking into account inverse kinematics. As can be seen, conventional sampling searches for 80 frames within the minimum distance radius until the true configuration is found and thus the edge error decreases. The corresponding joint angle trajectories are shown in Fig. 9. The proposed combined inverse kinematics sampling leads to almost immediate convergence, in contrast to sampling without inverse kinematics. To allow comparison of the results, the particle filter was run four times in one iteration of the conventional sampling method.

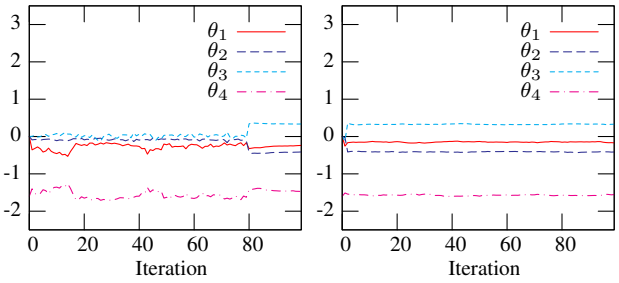

Fig. 9. Illustration of the effect of inverse kinematics sampling on the trajectory of the person's right arm shown in Fig. 6. Left: without inverse kinematics. Right: with inverse kinematics. The standard deviations for the iterations 3–99 for the angles $\theta_1, \theta_2, \theta_3, \theta_4$ are 0.011, 0.0070, 0.0076, 0.015, when using inverse kinematics sampling. The units are in radians.

VIII. Experimental Results

A. Runtime

In Table I, the runtimes for the different processing stages are given for the proposed system. The runtimes have been measured on a 3 GHz single core CPU for a test sequence consisting of 840 24 bit RGB stereo images with a resolution of 640×480 each. For arm motion tracking, 150 particles with four runs were used. The total processing time of 66 ms yields a processing rate of 15 Hz.

	Time [ms]
Skin color segmentation	4
Shirt color segmentation	20
Edge image calculation	6
Particle filters for hand/head tracking	6
Particle filters for arm motion tracking	30
Total	**66**

TABLE I

Processing times for the proposed system.

B. Real-world Experiments

For the results presented in this section, an exemplary sequence consisting of 840 frames captured at a frame rate of 30 Hz was processed and analyzed. The sequence was processed with the proposed system once on all 840 frames and once using every second frame only. By doing this, the degradation of the accuracy with lower frame rates can be observed. As will be shown, the proposed system operates robustly on lower frame rates as well, which is crucial for robust online application. The system proved to be applicable for online reproduction of movements on the humanoid robot ARMAR III, as presented in [16].

The estimated trajectories of the left and right arm are plotted in the Fig. 10 and Fig. 11, respectively. The angles θ_1–θ_4 are the direct output of the particle filter. The angles θ_1–θ_3 define a vector whose direction represents the rotation axis and whose magnitude the rotation angle. As can be seen, the trajectories acquired at 15 Hz and 30 Hz mostly equal. The greatest deviations can be observed for the first 100 frames

of the left arm in Fig. 10. However, the magnitude of the deviation is not representative for the actual error. The elbow angle for these frames is near zero, and the different values result from the uncertainty of the estimation of the upper arm rotation – a problem that is not related to the frame rate. Due to the small elbow angle, the projections of both trajectories look similar. The deviation for the angle θ_2 of the right arm for the frames 670–840 in Fig. 11 is due to the same ambiguity; again, the elbow angle is near zero. Judging from the visualized model configurations in 2D and 3D, both alternatives are plausible. For stable recognition or reproduction of such configurations with a humanoid robot system, the trajectories must post-processed in order to ensure continuity and uniqueness. This post-processing can be performed online at run-time, as applied for reproduction of movements on the humanoid robot ARMAR III presented in [16].

Finally, in Fig. 12 snapshots of the state of the tracker are given for the test sequence. Each snapshot corresponds to a frame $1+k\cdot 60$ from the Fig. 10 and Fig. 11, respectively. Note that not only the projection of the human model configuration to the left camera image is plausible, but also the estimated 3D pose illustrated by the 3D visualization of the human model is correct.

IX. Discussion and Outlook

We have presented a stereo-based markerless human motion capture system that is capable of robust real-time tracking of upper body motion. The processing rate amounts to 15 Hz on a 3 GHz single core CPU operating on stereo color image pairs with a resolution of 640×480. We introduced a prioritized fusion method for combining the edge cue and the distance cue, the latter operating on 3D positions acquired by a 3D hand/head tracking system. It was shown that this fusion method together with adaptive noise leads to substantially smoother and more accurate trajectories. Accurate model alignment is accomplished by modeling the shoulder position to be adaptive – in contrast to conventional models using a stiff ball joint for the shoulder. The introduced incorporation of the solutions of an inverse kinematics problem with a redundancy degree of one into particle sampling reduces the problem of local minima drastically, allowing for immediate recovery and automatic initialization.

In the current system, 3D hand/head tracking is performed separately in a pre-processing step for each frame. In the near future, we plan to resolve ambiguities that can occur throughout hand/head tracking by utilizing the evaluation function of the particle filters used for arm tracking. In this way, hand/head tracking and arm tracking can mutually support each other, rather than arm tracking benefitting from hand/head tracking only.

Acknowledgment

The work described in this paper was conducted within the EU Cognitive Systems project PACO-PLUS (FP6-2004-IST-4-027657) and funded by the European Commission.

References

[1] D. Demirdjian, T. Ko, and T. Darrell, "Constraining Human Body Tracking," in *International Conference on Computer Vision (ICCV)*, Nice, France, 2003, pp. 1071–1078.

[2] 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.

[3] D. Grest, , J. Woetzel, and R. Koch, "Nonlinear Body Pose Estimation from Depth Images," *Lecture Notes in Computer Science*, vol. 3663, pp. 285–292, 2005.

[4] D. Gavrila and L. Davis, "3-D Model-based tracking of humans in action: a multi-view approach," in *Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, San Francisco, USA, 1996, pp. 73–80.

[5] K. Rohr, "Human Movement Analysis based on Explicit Motion Models," *Motion-Based Recognition*, pp. 171–198, 1997.

[6] C. Bregler and J. Malik, "Tracking People with Twists and Exponential Maps," in *Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, Santa Barbara, USA, 1998, pp. 8–15.

[7] S. Wachter and H.-H. Nagel, "Tracking Persons in Monocular Image Sequences," *Computer Vision and Image Understanding*, vol. 74, no. 3, pp. 174–192, 1999.

[8] D. Grest, D. Herzog, and R. Koch, "Monocular Body Pose Estimation by Color Histograms and Point Tracking," in *DAGM-Symposium*, Berlin, Germany, 2006, pp. 576–586.

[9] T. E. de Campos, B. J. Tordoff, and D. W. Murray, "Recovering Articulated Pose: A Comparison of Two Pre and Postimposed Constraint Methods," *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 28, no. 1, pp. 163–168, 2006.

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

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

[12] 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.

[13] M. Fontmarty, F. Lerasle, and P. Danes, "Data Fusion within a modified Annealed Particle Filter dedicated to Human Motion Capture," in *International Conference on Intelligent Robots and Systems (IROS)*, San Diego, USA, 2007, pp. 3391–3396.

[14] P. Azad, A. Ude, T. Asfour, and R. Dillmann, "Stereo-based Markerless Human Motion Capture for Humanoid Robot Systems," in *International Conference on Robotics and Automation (ICRA)*, Roma, Italy, 2007, pp. 3951–3956.

[15] J. Deutscher, A. Davison, and I. Reid, "Automatic Partitioning of High Dimensional Search Spaces associated with Articulated Body Motion Capture," in *Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, Kauai, USA, 2001, pp. 669–676.

[16] M. Do, P. Azad, T. Asfour, and R. Dillmann, "Imitation of Human Motion on a Humanoid Robot using Nonlinear Optimization," in *International Conference on Humanoid Robots (Humanoids)*, Daejeon, Korea, 2008.

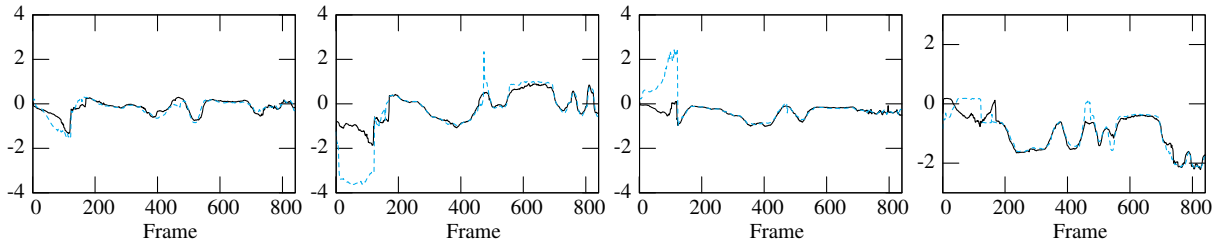

Fig. 10. Exemplary arm trajectory for the left arm acquired by the proposed human motion capture system. The solid line indicates the tracking result acquired at the full temporal resolution of 30 Hz; for the dashed line every second frame was skipped, i.e. 15 Hz. The angles θ_1–θ_4 are plotted from left to right. The units are in radians.

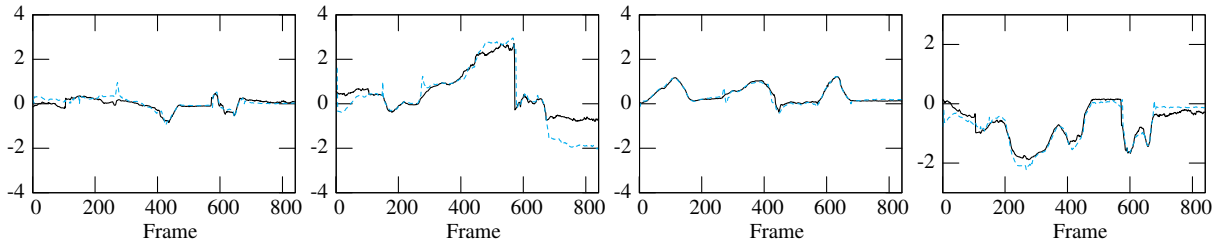

Fig. 11. Exemplary arm trajectory for the right arm acquired by the proposed human motion capture system. The solid line indicates the tracking result acquired at the full temporal resolution of 30 Hz; for the dashed line every second frame was skipped, i.e. 15 Hz. The angles θ_1–θ_4 are plotted from left to right. The units are in radians.

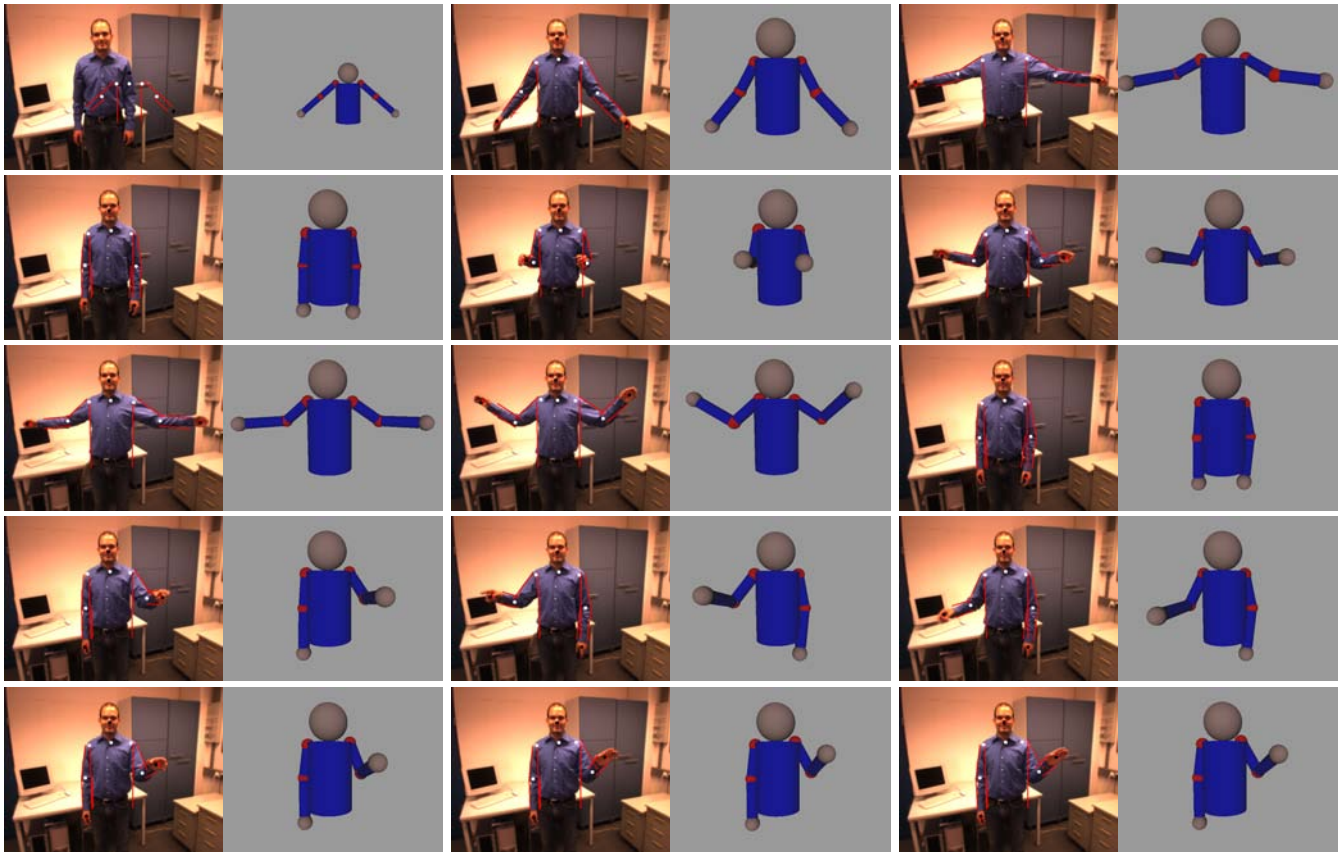

Fig. 12. Snapshots of the results computed for a test sequence consisting of 840 frames, which were captured at a frame rate of 30 Hz. Every 60th frame is shown; the frames are ordered row-wise from top left to bottom right. The red dots mark the measured positions computed by the hand/head tracking system. The black dots mark the corresponding positions according to the estimated model configuration. The first frame illustrates the initial state of the particle filter.

Imitation of Human Motion on a Humanoid Robot using Non-Linear Optimization

Martin Do, Pedram Azad, Tamim Asfour, Rüdiger Dillmann

University of Karlsruhe, Germany $do@ira.uka.de$, $azad@ira.uka.de$, $asfour@ira.uka.de$, $dillmann@ira.uka.de$

***Abstract*—In this paper, we present a system for the imitation of human motion on a humanoid robot, which is capable of incorporating both vision-based markerless and marker-based human motion capture techniques. Based on the so-called Master Motor Map, an interface for transferring motor knowledge between embodiments with different kinematics structure, the system is able to map human movement to a human-like movement on the humanoid while preserving the goal-directed characteristics of the movement. To attain an exact and goal-directed imitation of an observed movement, we introduce a reproduction module using non-linear optimization to maximize the similarity between the demonstrated human movement and the imitation by the robot. Experimental result using markerless and marker-based human motion capture data are given.**

I. Introduction

The interaction between robots and humans is one of the main goals in humanoid robotics research. A successful interaction depends on various factors like the acceptance of a humanoid robot by society, its capabilites to act in unconstrained human-centered environments and its communication skills. As a consequence, to raise its acceptance in society, a robot needs to adapt human characteristics to its actions and skills. Especially, human-like motion and gestures of a robot are main contributions to its appearance, which has a strong influence on a user. Hence, under these circumstances controlling the motion of a robot is a very challenging task and still a major topic in humanoid robotics research. The most intuitive solution for this problem lies in imitation, where the user adopts the role of a teacher by demonstrating how to perform a certain action, while the robot tries to repeat this action on the basis of the observation. The benefit of exploiting demonstration is clearly revealed in [1], where an anthropomorphic arm is capable of balancing a pole in the first trial after observing a human. The concept of imitation can be understood in many ways. In [2], imitation of humans in the field of robotics is divided into two categories: imitation learning and motion imitation.

Imitation learning sets the focus on the understanding of actions. Following this scheme, which underlies imitation learning methods, first, data is collected from multiple observations of a demonstrated action. From this data collection features are extracted allowing the robot to draw conclusions on the humans behaviour. Based on the learned behaviour, the robot should be able to reproduce a generalized version of the demonstrated action.

In [3], a neuroscientific inspired approach is presented, which solves imitation learning of cyclic motion with a set of basic motor primitives. These are learned by clustering and dimensionality reduction of visually acquired human motion data. For reproduction, a movement is classified into motor primitives which are played back sequentially.

In [4] and [5] methods are introduced, where Hidden Markov Models are trained with a collection of observations of a demonstrated movement. To reproduce a newly observed movement, the observation is recognized based on a set of trained models. With the complying model, a generalization of the recognized movement is generated.

Imitation learning approaches emphasize the learning and understanding of human behaviour by its interpretation by the humanoid. These methods require offline processing and due to the loss of accuracy as a result of generalization, they are often limited to simple movements.

The imitation of a complex motion requiring high precision and stability is addressed by approaches dealing with the pure imitation of motion. In contrast to imitation learning, the learning of any kind of behaviour is disregarded. Instead, the focus is on finding a trajectory, which corresponds exactly to the data, that a humanoid obtains from a human motion capture system. [6], [7], and [8] present methods for motion imitation, which make use of artificial markers on the humanoid robot as well as the demonstrator. For the reproduction of motion, corresponding marker positions between both subjects are minimized leading to similar postures. Instead of exploiting marker positions, [9] and [10] calculate the joint angles of a demonstrators posture, which are transferred to the robot for execution. Due to joint and velocity constraints, a scaling and transformation process must be performed to obtain a feasible joint angle configuration for the robot. In contrast to the mentioned motion imitation approaches mentioned above, a more natural way of imitation using the humanoid robots own stereo vision system to record human trajectories by exploiting color markers on the demonstrators clothing is presented in [11].

Each of these approaches is focused on a specific human motion capture technique. Since every technique has its advantages and drawbacks, in our approach, we propose a system for the imitation of motion within a framework that allows integration of various marker-based and markerless human motion capture systems and the reproduction on a robot. This compability leads to a high level of flexibility and versatility, which opens the system to a wide range of different applications from motion analysis to imitation of highly complex motions in real-time.

Concerning the reproduction of object manipulation actions, one desires a module that produces trajectories that keep the goal-directedness of the observed movement while keeping the human-like charactericstics of the motion. The term goal-directedness refers to the pose of the end effector relative to the object of interest. Since the pose of the object relative to the robot will always differ from the observed situation, one needs the possibility to incorporate the currently desired end effector pose which can be derived from the currently observed object pose into the transformation procedure.

However, due to severe constraints of mechanical systems and unknown environments, it becomes very difficult to satisfy all requirements. Inspired by the previous works, a reproduction module is developed based on a non-linear optimization problem, which incorporates the robots hand in the task space as well as the joint angles. Similar to the previous imitation solutions, we focus on the optimization of the humanoids posture in each frame.

The paper is organized as follows. Section II describes the proposed imitation system and the human robot used in the experiments. In Section III, an overview of markerless and marker-based human motion capture is given. The extension of the Master Motor Map is described in Section V. The generation of human-like movements from captured human motion using non-linear optimization techniques is presented in Section VI. Finally, experimental results are given in VII.

II. System Overview

As depicted in Fig. 1, the proposed system consists of three major components, which are coupled in consecutive processing stages: the acquisition of human motion, the Master Motor Map (MMM) interface [12], and the motion generation and reproduction.

As mentioned before, the proposed system allows data input from different human motion capture systems. For applications requiring highly accurate data, marker-based motion capture systems are more suitable. In contrast, for online imitation in a natural way, markers cannot be used. For the experiments performed in the context of this paper, the Vicon system [13] was used for marker-based motion capture (see Section III-B) and the stereo-based markerless motion capture system presented in [14] (see Section III-A) for natural imitation.

In both cases, the acquired trajectories are first translated to the unifying MMM format. In order to enhance both, human-likeness and accuracy, the MMM joint angle configuration runs through an optimization procedure, which fits the configuration to the kinematical structure and constraints of the robot. By interpolation between the consecutive posture frames, a smooth imitated movement is generated. If communciation between the single modules becomes necessary, e.g. when using an external Vicon system, UDP is used to establish the connection.

A. ARMAR-IIIb

The humanoid robot ARMAR-IIIb, which serves as the experimental platform in this work, is a copy of the humanoid

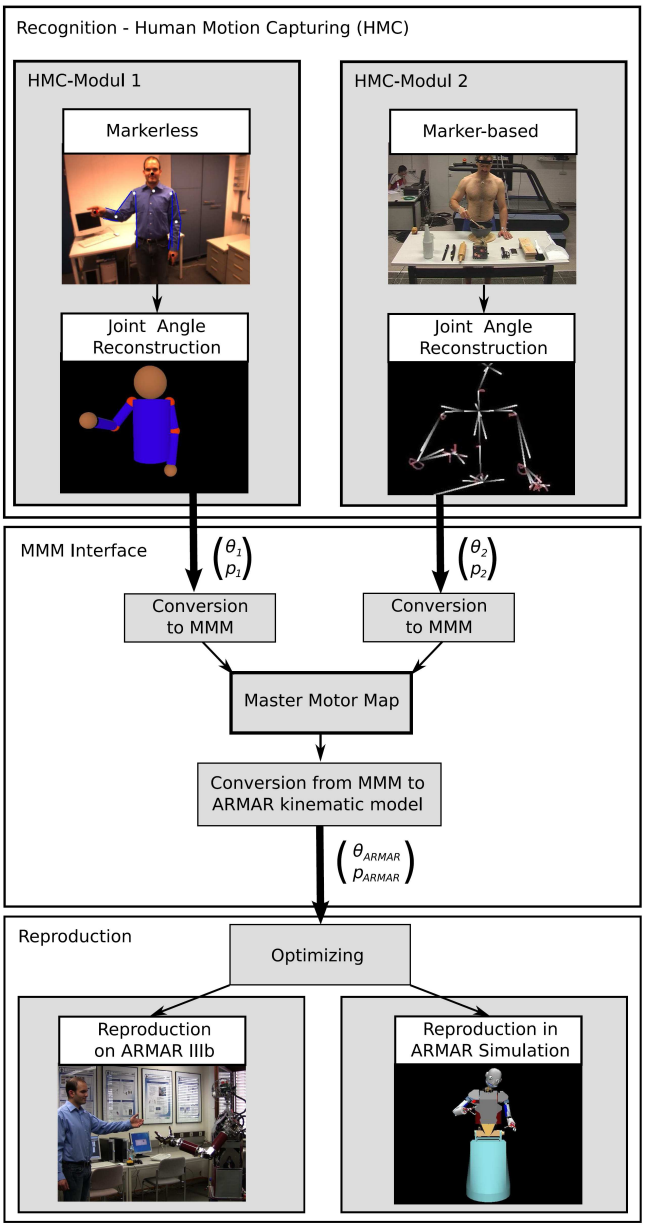

Fig. 1. Overview of the proposed system. θ_x denotes the joint angles, while p_x describe the hand position in the Cartesian space.

robot ARMAR-IIIa [15]. From the kinematics point of view, the robot consists of seven subsystems: head, left arm, right arm, left hand, right hand, torso, and a mobile platform. The head has seven DoF and is equipped with two eyes, which have a common tilt and can pan independently. Each eye is equipped with two digital color cameras, one with a wide-angle lens for peripheral vision and one with a narrow-angle lens for foveal vision. The upper body of the robot provides 33 DoF: 2·7 DoF for the arms and three DoF for the torso. The arms are designed in an anthropomorphic way: three DoF for each shoulder, two DoF in each elbow and two DoF in each wrist. Each arm is equipped with a five-fingered hand with

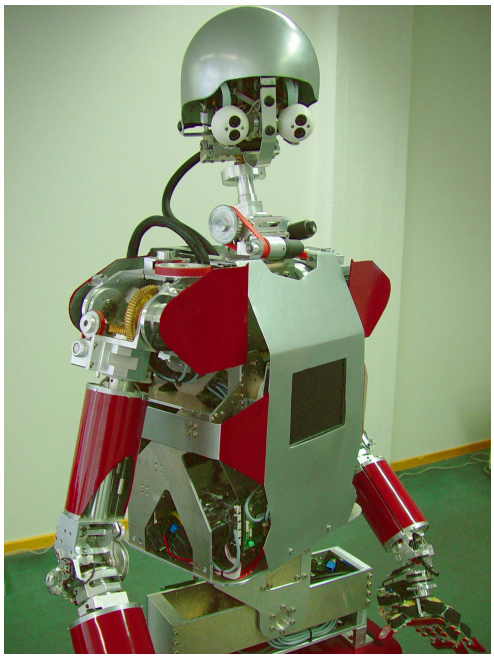

Fig. 2. The humanoid robot ARMAR-IIIb.

eight DoF. The locomotion of the robot is realized using a wheel-based holonomic platform.

III. Human Motion Capture

In this section, a short outline of the integrated markerless and marker-based human motion capture methods is given. In addition to the brief descriptions of the techniques, the advantages as well as the drawbacks are discussed. Furthermore, possible applications are pointed out.

A. Markerless Human Motion Capture

In the following, our real-time stereo-based human motion capture system presented in [14] will be summarized briefly. The input to the system is a stereo color image sequence, captured with the built-in wide-angle stereo pair of the humanoid robot ARMAR-IIIb, which can be seen in Fig. 2. The input images are pre-processed, generating output for an edge cue and a so-called distance cue, as introduced in [16]. The image processing pipeline for this purpose is illustrated in Fig. 3. Based on the output of the image processing pipeline, a particle filter is used for tracking the movements in joint angle space. For tracking the movements, a 3D upper body model with 14 DoF (6 DoF for the base transformation, 2·3 for the shoulders, and 2·1 for the elbows) consisting of rigid body parts is used, which provides a simplified description of the kinematic structure of a human. The model configuration is determined by the body properties like the limbs length of the observed human subject. The core of the particle filter is the likelihood function that evaluates how well a given model configuration matches the current observations, i.e. stereo image pair. For this purpose, an edge cue compares the projected model contours to the edges in the image. On the basis of an additional 3D hand/head tracker, the distance cue evaluates the distance between the measured positions and the corresponding positions inferred by the forward kinematics of the model. Various extensions are necessary for robust real-time application such as a prioritized fusion method, adaptive shoulder positions, and the incorporation of the solutions of the redundant arm kinematics. The system is capable of online tracking of upper body movements with a frame rate of 15 Hz on a 3 GHz single core CPU. Details are given in [14].

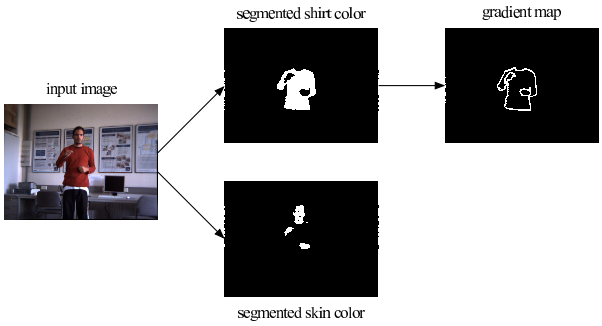

Fig. 3. Illustration of the image processing pipeline.

B. Marker-based Human Motion Capture

Marker-based human motion capture frameworks are widespread systems in the robotics research community as well as in the industry. One of the most popular commercially available systems is provided by Vicon [13]. The technique, which is used here, relies on infrared cameras and artificial reflective markers. The markers are placed on predefined body parts of a human subject. In a defined workspace, the subject is surrounded by a set of infrared cameras. Each camera is equipped with a infrared strobe, emitting a light signal, which is reflected by the markers. The reflected light, which distinguishes itself from the background, is registered by the cameras. The data from each camera consisting of 2D coordinates of each recognized marker position, is merged in a data station, which computes the 3D position by triangulation and the label of each visible marker. Besides the hardware, the system contains a comprehensive software package, which facilitates the calibration and handling of the system. Due to the high-speed and high-resolution properties of the cameras, the Vicon system provides an accurate method for capturing human motion at high frame rates. Furthermore, since the use of numerous markers allows capturing of barely visible motion of unobvious joints, complex kinematic models are applicable for the processing and representation of the motion data. The problem of occlusion of body parts is reduced to a minimum, since multiple cameras are used, which deliver multiple views of the same subject. However, the enormous equipment needs cause high costs. Furthermore, a time and space-consuming preparation is essential to provide the necessary setup for proper human motion capture. For our purpose, the joint angles are reconstructed by optimization of a human model based on the computed 3D marker positions. Details are given in [17].

IV. Extended Master Motor Map

Since each human motion capture system produces data in terms of its own specific model and format, respectively, one has to deal with a variation of different data formats. Likewise, for reproduction of movements, each robot system requires data in terms of its own kinematics. One possible solution could be the definition of an interface for each combination of a sensing system and a robot. However, doing so would restrict the robot and the utilization of the data. To overcome this difficulties, in this work, a standardized interface is established by using the MMM, which features a high level of flexibilty and combability. The MMM is introduced in [12] and provides a reference kinematic model by defining the maximum number of DoF, that can be used by a human motion capture module and a robot. A trajectory in the original MMM file format consists of 52-dimensional vectors, each vector describing a joint angle configuration with a floating point number for every single DoF. Since we are able to recognize the human finger movements using the Vicon system, the MMM is extended by three DoF for coupled finger flexion, thumb flexion and thumb abduction. The reference kinematic model of the extended MMM is illustrated in Fig. 4. As a result of the extension, one obtains a 58-dimensional vector for the description of a joint angle configuration of the model. Due to differences in the Euler conventions, active joint sets, which can be controlled, and the order of the joint angle values between the modules, a conversion module has to be implemented for each of the systems in order to provide a proper connection via the MMM. This conversion module transforms the module specific data into the MMM file format and vice versa. As depicted in Fig. 1, for the proposed system, one conversion module is implemented for each human motion capture system, converting the motion capture data to the MMM format. A third conversion module is implemented for mapping the MMM data to the kinematics of ARMAR-IIIb. The focus of this paper lies on this third module and is presented in the following. Further details on the MMM are given in [12].

V. Reproduction of Trajectories

Concerning the imitation of humanoid motion, the simplest and most desired way to reproduce a movement from given joint angles consists of a one-to-one mapping between an observed human subject and the robot. Unfortunately, due to the differences in the kinematic structures of a human and the robot e.g. differing joints and limb measurements, only in rare cases a one-to-one mapping shows acceptable performance regarding the functionality as well as the human-like appearance of the reproduced movement. In this work, we address this problem by applying a postprocessing procedure in joint angle space. In two stages, the joint angles, given in the MMM format, are optimized concerning the tool center point (TCP) position and the kinematic structure of the robot. First, a feasible solution is estimated, which serves as an intial solution for an optimization step in the second stage. Following this scheme, one obtains a human-like motion on the robot, while preserving its goal-directed characteristics.

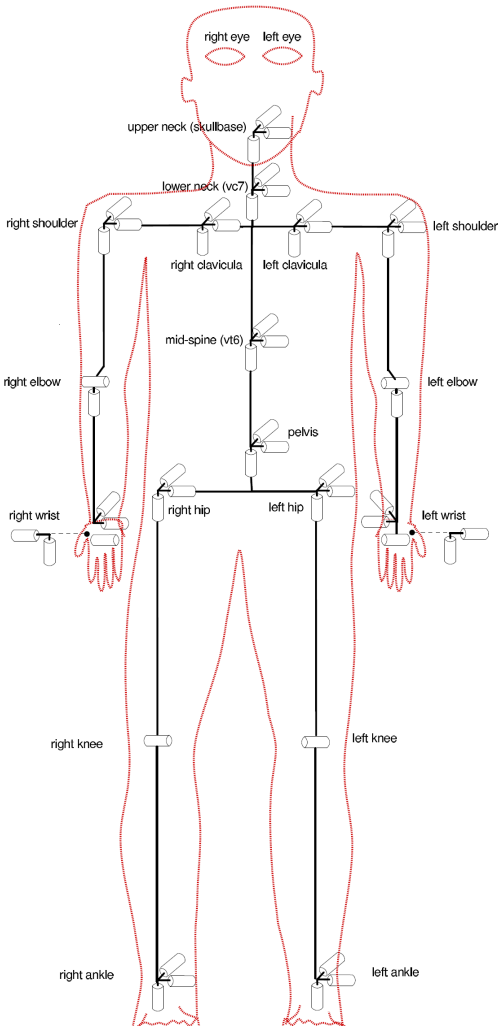

Fig. 4. Reference kinematic model of the Extended Master Motor Map.

A. Similarity Measure

One of the most crucial factors in the reproduction of human motion is the measure for rating the similarity between the imitated and the demonstrated movement. For the online reproduction of a human movement, one is more interested in comparing the current postures at the time t than in investigating a whole trajectory. In [7], it is proposed to determine the distance between the postures of the robot and the human by exploiting point correspondences between specified points on both bodies. To infer useful statements concerning the similarity, accurate localization and identification of the limbs are required, which makes the use of physical markers inevitable. In [9], a similarity measure is introduced, which only considers the joint angle relations. However, it disregards structural differences between human and robot like differing limb lengths, which one has to take into account in order to preserve the goal of a movement when mapped on the robot. Combining both, the joint angle configuration and key point correspondences, for a joint angle configuration $\boldsymbol{\sigma} \in \mathbb{R}^n$ with n joints, we define the similarity measure as follows:

$$S(\boldsymbol{\sigma}) = 2 - \frac{\frac{1}{n}\sum_{i=1}^{n}\left(\hat{\sigma_i}^t - \sigma_i\right)^2}{\pi^2} - \frac{\frac{1}{3}\sum_{i=1}^{3}\left(\hat{p_k}^t - p_k\right)^2}{\left(2 \cdot l_{arm}\right)^2} \quad (1)$$

with $\sigma_i, \hat{\sigma_i}^t \in [0, \pi]$ and $p_k, \hat{p_k}^t \in [-l_{arm}, l_{arm}]$, whereas l_{arm} describes the robots arm length. The reference joint angle configuration is denoted by $\hat{\boldsymbol{\sigma}} \in \mathbb{R}^n$, while $\hat{\boldsymbol{p}} \in \mathbb{R}^3$ stands for the desired TCP position. The current TCP postion $\boldsymbol{p}$ can be determined by applying the forward kinematics of the robot to the joint angle configuration $\boldsymbol{\sigma}$.

B. Estimation of an Initial Solution

To obtain a posture, which bears a high resemblance to the one of the demonstrator and at the same time meets all the mechanical constraints of the robot, the original joint angle configuration is optimized regarding the similarity measure as specified in Eq. 1. An optimal solution is found by applying a numerical optimization algorithm, namely the Levenberg-Marquardt (LM). However, the efficiency of most of the numerical optimization algorithms strongly depends on the initial estimation of the parameters to be optimized. An initial estimation within the neighbourhood of the optimal solution leads to a high chance that the algorithm converges fast directly towards the optimum without being trapped in local extrema. In this work, an initial estimation is determined from a preselection of candidate initial joint angle configurations, which are generated and evaluated by means of the similarity measure. To generate a candidate initial estimation $\boldsymbol{\sigma}^j$, the reference joint angle configuration $\hat{\boldsymbol{\sigma}}^t$ computed at time t is mapped into the robot joint angle space and projected on the bound constraints:

$$\hat{\sigma}_i^t = \begin{cases} C_{i_{min}} & \text{if } \hat{\sigma}_i^t \leq C_{i_{min}} \\ \hat{\sigma}_i^t & \text{if } C_{i_{min}} \leq \hat{\sigma}_i^t \leq C_{i_{max}} \\ C_{i_{max}} & \text{if } \hat{\sigma}_i^t \geq C_{i_{max}} \end{cases} \quad (2)$$

where $C_{i_{min}}$ and $C_{i_{max}}$ denote the lower and upper joint angle bounds of joint i. If the value of $\hat{\sigma}_i^t$ exceeds the given bounds, the joint i is fixed at the closest of the two boundaries. A candidate is obtained by altering each non-fixed joint angle of the mapped configuration by means of a vector $\boldsymbol{\delta}^t \in \mathbb{R}^n$ with $\delta_i^t = \hat{\sigma}_i^t - \hat{\sigma}_i^{t-1}$. Thus, $\boldsymbol{\delta}^t$ describes the changes between two consecutive frames. As a result, a candidate initial estimation can be described as:

$$\sigma_i^j = \hat{\sigma}_i^t + \alpha_i \beta_i \quad (3)$$

with

$$\alpha_i = \begin{cases} 1 & \text{if } C_{i_{min}} \leq \hat{\sigma}_i^t \leq C_{i_{max}} \\ 0 & \text{else} \end{cases} \quad (4)$$

$$\beta_i \in \{-\delta_i^t, 0, \delta_i^t\} \quad (5)$$

Given n joints to control, in the worst case $M = 3^n$ candidates need to be calculated and evaluated. The best initial estimation satisfies the following equation:

$$\boldsymbol{\sigma}_{init} = \underset{j=1,\ldots,M}{\arg\max}\ S(\boldsymbol{\sigma}^j) - \|\hat{\boldsymbol{\sigma}}^t - \boldsymbol{\sigma}^j\| \quad (6)$$

Finding the best initial estimation causes some overhead regarding processing time, but it is necessary to ensure that the LM algorithm will provide an optimal solution.

C. Optimization Problem

For optimization of a reference joint angle configuration regarding the similarity measure, one can use the Levenberg-Marquardt algorithm. The algorithm, which was first introduced in [18], provides a standard technique for solving non-linear least squares problems by iteratively converging to a minimum of function expressed as sum of squares. Combinating the Gauss-Newton and the steepest descent method, the algorithm unites the advantages of both methods. Hence, using the LM method, a more robust convergence behaviour is achieved at points far from a local minimum, while a faster convergence is gained close at a minimum. Due to its numerical stability, the LM method has also become a popular tool for solving inverse kinematics problems as demonstrated in [19]. For our problem, where, given the reference joint angle configuration $\hat{\boldsymbol{\sigma}}^t$, we seek a $\boldsymbol{\sigma}^t$, which maximizes Eq. 1. To interpret Eq. 1 as a function of sum of squares to be minimized, we define a function $s(\boldsymbol{\sigma}) : \mathbb{R}^n \rightarrow \mathbb{R}^m$ with $n < m$ as follows:

$$s(\boldsymbol{\sigma}) = \begin{pmatrix} \frac{1}{\sqrt{3}\cdot 2\cdot l_{arm}}\ p_1 \\ \frac{1}{\sqrt{3}\cdot 2\cdot l_{arm}}\ p_1 \\ \frac{1}{\sqrt{3}\cdot 2\cdot l_{arm}}\ p_1 \\ \frac{1}{\sqrt{n}\cdot \pi}\ \sigma_1 \\ \vdots \\ \frac{1}{\sqrt{n}\cdot \pi}\ \sigma_n \end{pmatrix} \quad (7)$$

The corresponding optimization problem can be written in the following form:

$$\min S'(\boldsymbol{\sigma}) = 2 - S(\boldsymbol{\sigma}) \quad (8)$$

$$\text{subject to} \quad C_{i_{min}} \leq \hat{\sigma}_i \leq C_{i_{max}} \quad (9)$$

which is equivalent to the maximization of Eq. 1. Similar to the Gauss-Newton method, in the LM method a Taylor expansion of s is performed around $\boldsymbol{\sigma}$. For a small $\boldsymbol{\rho}$, s can be approximated by the following equation:

$$s(\boldsymbol{\sigma} + \boldsymbol{\rho}) \approx s(\boldsymbol{\sigma}) + J_s \boldsymbol{\rho} \quad (10)$$

where J_s denotes the Jacobian of s. Based on the initial guess $\boldsymbol{\sigma}_{init}$, a sequence of estimations $\boldsymbol{\sigma} + \boldsymbol{\rho}$ is calculated that converges to a solution of Eq. 9. Therefore, in each iteration, the optimization problem is reduced to finding a $\boldsymbol{\rho}$, that minimizes $\|s(\hat{\boldsymbol{\sigma}}^t) - s(\boldsymbol{\sigma}) + J_s\boldsymbol{\rho}\|$. For an adequate $\boldsymbol{\rho}$ the following condition must hold true:

$$\left(s(\hat{\boldsymbol{\sigma}}^t) - s(\boldsymbol{\sigma}) + J_s\boldsymbol{\rho}\right) J^T = 0 \quad (11)$$

Solving the least squares problem of Eq. 11 yields the sought $\boldsymbol{\rho}$. Based on Eq. 11, the LM algorithm solves following slightly modified equation:

$$I\mu + J_s^T J_s \boldsymbol{\rho} = J_s^T s(\boldsymbol{\sigma} + \boldsymbol{\rho}) \quad (12)$$

which includes a dampening term μ. If reduction of S' concerning $\boldsymbol{\rho}$ can be accomplished, then for the next iteration $\boldsymbol{\sigma} := \boldsymbol{\sigma} + \boldsymbol{\rho}$ holds and a smaller value is assigned to μ to achieve faster convergence. If reduction fails, μ is set to a higher value, which slows down the convergence. Furthermore, μ prevents meeting singularites in the Jacobian. To obtain a feasible joint angle configuration, after each iteration, $\boldsymbol{\sigma}$ is projected onto the bound constraints according to Eq. 2. The algorithm terminates if $S'(\boldsymbol{\sigma}) < \epsilon_1$ or $\|\boldsymbol{\rho}\| < \epsilon_2$, and one can set $\boldsymbol{\sigma}^t = \boldsymbol{\sigma}$. More practical details on the algorithm can be found in [20].

VI. Experimental Results

In this section, results of experiments of the imitation system with the two human motion capture systems introduced in Section III are demonstrated. The approach was evaluated by comparison to an inverse kinematics method based on the Jacobian transpose and a one-to-one mapping of the captured joint angles onto the robot. The results were generated with the humanoid robot platform ARMAR-IIIb in real-world as well as in simulation.

A. Marker-based Motion Capture Data

The hardware setup which was used to capture the human motion consists of ten Vicon cameras. Since using a marker-based approach allows to capture a large set of degrees of freedom, the number of active joint angle adds up to 24 DoF, ten for each arm, three DoF for the head and one DoF for the hip rotation. Concerning the arm, three DoF are assigned to the shoulder rotation, two for the elbow, two for the wrist and three DoF describe the finger movements. The experiments focused on the reproduction of actions in a kitchen scenario. The data was generated within the work of [21]. The kitchen actions included movements like stirring, cutting with a knife, sweeping, grinding coffee beans, grating, and pouring. Fig. 8 shows screenshots of cutting sequence, which was reproduced on ARMAR in simulation. The results using the optimization as proposed in this work on marker-based captured motion data are illustrated in Fig. 5. The left plot of Fig. 5 shows the joint angle error of a reproduced joint angle configuration on the robot and the reference configuration. Due to redundancy, the inverse kinematics method produces results with higher error, while a one-to-one mapping naturally leads to a minimal error. In the center plot of Fig. 5, by the right arm TCP, the deviation of the TCP positioning is illustrated. Here, given a TCP destination, the inverse kinematics method leads to an exact positioning of the TCP, while using the one-to-one mapping the destinated position is not reached. In both plots, it is shown, that the application of the optimization procedure as proposed in Section V, a tradeoff is attained, which results in a quite accurate TCP positioning with an maximum error of 25 mm and an acceptable mean joint angle error of 2.0 degrees for each DoF. One of the most crucial joints which has a huge impact on the style of a trajectory is the shoulder joint. Therefore, the right plot of Fig. 5 shows the joint angle error for this joint in particular.

B. Markerless Motion Capture Data

For the online reproduction and imitation of the observed human motion, the stereo camera system of ARMAR-IIIb was used to capture the upper body movements with the method described in Section III-A. Using the onboard cameras allows to perform a more natural way of imitation, but limits the number of DoF, which can be measured, since the system is more sensitive to noise and occlusion. A total number of eight DoF is used, four for each arm, three DoF for the shoulder joint and one for elbow flexion. The under arm rotation, the wrist, and finger movements cannot be recognized with this system. The online reproduction was tested with simple movements like reaching, waiving, and approaching certain postures. Some sample images showing the online imitation of human motion can be seen Fig. 7. Similar to the results achieved with the Vicon system, applying the proposed motion imitation system leads to a tradeoff between the accuracy of the TCP position and the joint angle error. However, due to the reduced number of measured joints, one obtains results with a mean joint angle error of 2.7 degrees for each DoF, as shown in the left plot of Fig. 6, and a maximum deviation of 65 mm in the TCP position of the right arm, as shown in the center plot of Fig. 6. The reason for the relatively large deviation is that the utilized vision-based motion capture system is not yet capable of measuring the torso rotation. This lack information leads to a decreased flexibility throughout the reproduction, assuming the hip joint angles to be fixed. One solution would be to incorporate the hip rotation into the optimization procedure in order to allow for the missing flexibility even if the torso rotation cannot be measured.

VII. Conclusions

In this work, we have presented a system for motion imitation with the goal of attaining a human-like motion, but without loss of functionality. Based on the Master Motor Map, a system was developed, which is capable of incorporating various human motion capture techniques. In particular, it was dealt with the marker-based Vicon system and a markerless vision-based approach. Their output is transformed to the structure of the robot platform ARMAR-IIIb by using a non-linear optimization technique in form of the LM algorithm. A natural way of motion imitation is demonstrated by applying the system successfully for the online reproduction of observed motion. Furthermore, with the system, reproduction of complex kitchen actions was achieved based on high-resolution Vicon data. In the near future, the involvement of objects is planned to enable imitation of manipulation tasks. The proposed system provides a solid basis for further studies towards human motion analysis. By incorporating machine learning methods, the system can be extended imitation learning tasks.

VIII. Acknowledgment

The work described in this paper was partially conducted within the EU Cognitive Systems projects GRASP (FP7-215821) and PACO-PLUS (FP6-027657) funded by the European Commission.

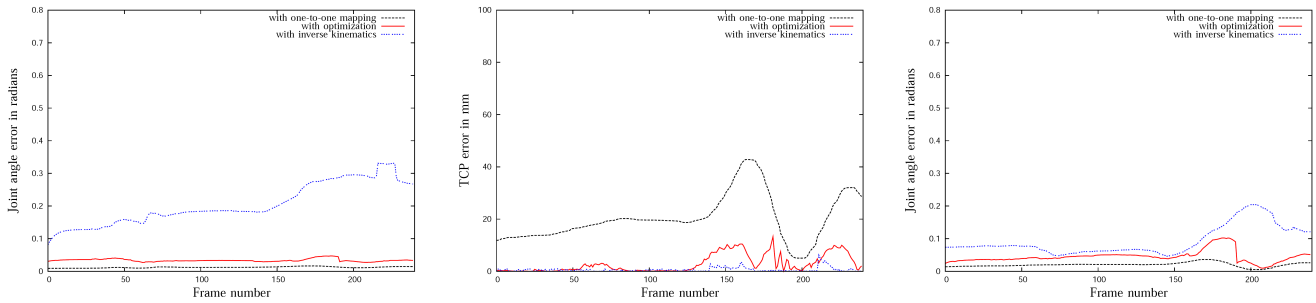

Fig. 5. Evaluation results for the reproduction of motion captured by the Vicon system. Left: Mean joint angle error over all active joints in radians. Center: Deviation of right arm TCP of the robot and a predefined destination in mm. Right: Mean joint angle error over the shoulder joint in radians.

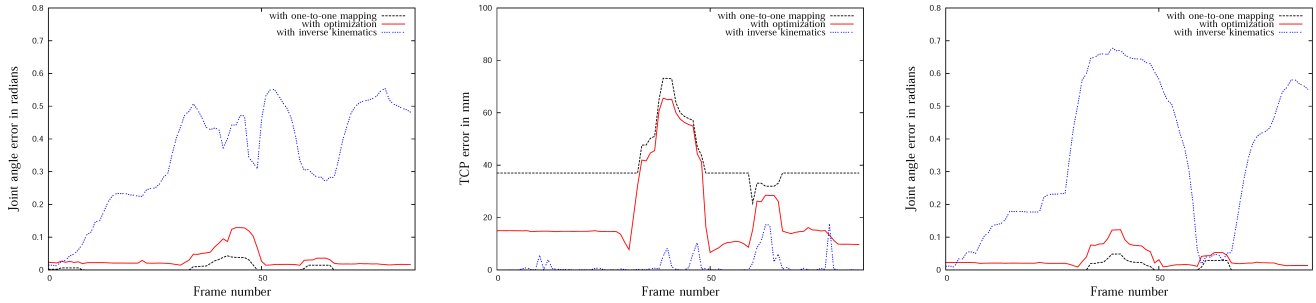

Fig. 6. Evaluation results for the reporduction of vision-based captured motion. Left: Mean joint angle error over all active joints in radians. Center: Deviation of right arm TCP of the robot and a predefined destination in mm. Right: Mean joint angle error over the shoulder joint in radians.

REFERENCES

[1] S. Schaal, "Learning from Demonstration," in *Advances in Neural Information Processing Systems 9*, Denver, USA, December 1997, pp. 1040–1046.

[2] P. Bakker and Y. Kuniyoshi, "Robot see, Robot do: An Overview of Robot Imitation," in *AISB96 Workshop: Learning in Robots and Animals*, Brighton, UK, 1996.

[3] M. J. Mataric, "Getting Humanoids to Move and Imitate," *IEEE Intelligent Systems*, vol. 15, no. 4, pp. 18–24, 2000.

[4] S. Calinon and A. Billard, "Learning of Gestures by Imitation in a Humanoid Robot," in *Imitation and Social Learning in Robots, Humans and Animals: Behavioural, Social and Communicative Dimensions*. Cambridge University Press, 2007, pp. 153–177.

[5] T. Asfour, F. Gyarfas, P. Azad, and R. Dillmann, "Imitation Learning of Dual-Arm Manipulation Tasks in Humanoid Robots," in *IEEE-RAS International Conference on Humanoid Robots*, Genova, Italy, December 2006, pp. 40–47.

[6] C. Kim, D. Kim, and Y. Oh, "Adaption of Human Motion Capture Data to Humanoid Robots for Motion Imitation using Optimization," *Integrated Computer-Aided Engineering*, vol. 13, no. 4, pp. 377–389, 2006.

[7] D. Matsui, T. Minato, K. F. MacDorman, and H. Ishiguro, "Generating Natural Motion in an Android by Mapping Human Motion," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Edmonton, Alberta, Canada, August.

[8] A. Ude, C. Atkeson, and M. Riley, "Programming Full-Body Movements for Humanoid Robots by Observation," *Robotics and Autonomous Systems*, vol. 47, no. 2-3, pp. 93–108, 2004.

[9] X. Zhao, Q. Huang, Z. Peng, and K. Li, "Humanoid Kinematics Mapping and Similiarity Evaluation based on Human Motion Capture," in *IEEE International Conference on Information Acqusition*, Hefei, China, June 2004, pp. 426–431.

[10] N. Pollard, J. Hodgins, M. Riley, and C. Atkeson, "Adapting Human Motion for the Control of a Humanoid Robot," in *IEEE International Conference on Robotics and Automation*, Washington, DC, USA, May 2002, pp. 1390–1397.

[11] M. Riley, A. Ude, K. Wade, and C. Atkeson, "Enabling Real-Time Full-Body Imitation: A Natural Way of Transferring Human Movement to Humanoids," in *IEEE International Conference on Robotics and Automation*, Taipei, Taiwan, September 2003, pp. 2368–2374.

[12] P. Azad, T. Asfour, and R. Dillmann, "Toward an Unified Representation for Imitation of Human Motion on Humanoids," in *IEEE International Conference on Robotics and Automation*, Rome, Italy, April 2007.

[13] "Vicon Peak," Website, available online at http://www.vicon.com.

[14] P. Azad, T. Asfour, and R. Dillmann, "Robust Real-time Stereo-based Markerless Human Motion Capture," in *submitted to International Conference on Humanoid Robots*, Daejeon, Korea, December 2008.

[15] T. Asfour, K. Regenstein, P. Azad, J. Schröder, N. Vahrenkamp, and R. Dillmann, "ARMAR-III: An Integrated Humanoid Platform for Sensory-Motor Control," in *IEEE/RAS International Conference on Humanoid Robots*, 2006.

[16] 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.

[17] H. Köhler, M. Pruzinec, T. Feldmann, and A.Wörner, "Automatic Human Model Parametrization from 3D Marker Data for Motion Recognition," in *International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*, Plzen, Czech Republic, February 2008.

[18] K. Levenberg, "A Method for the Solution of Certain Non-Linear Problems in Least Squares," *The Quarterly of Applied Mathematics*, vol. 2, pp. 164–168, 1944.

[19] C. W. Wampler, "Manipulator Inverse Kinematic Solutions based on Vector Formulations and Damped Least Squares Methods," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 16, pp. 93–101, 1986.

[20] M. L. A. Lourakis and A. A. Argyros, "Is Levenberg-Marquardt the Most Efficient Optimization Algorithm for Implementing Bundle Adjustment ?" in *IEEE International Conference on Computer Vision*, vol. 2, Beijing, China, October 2005.

[21] T. Stein, A. Fischer, I. Boesnach, H. Köhler, D. Gehrig, and H. Schwameder, *Kinematische Analyse menschlicher Alltagsbewegungen für die Mensch-Maschine-Interaktion*. V. Aachen: Shaker, 2007, in press.

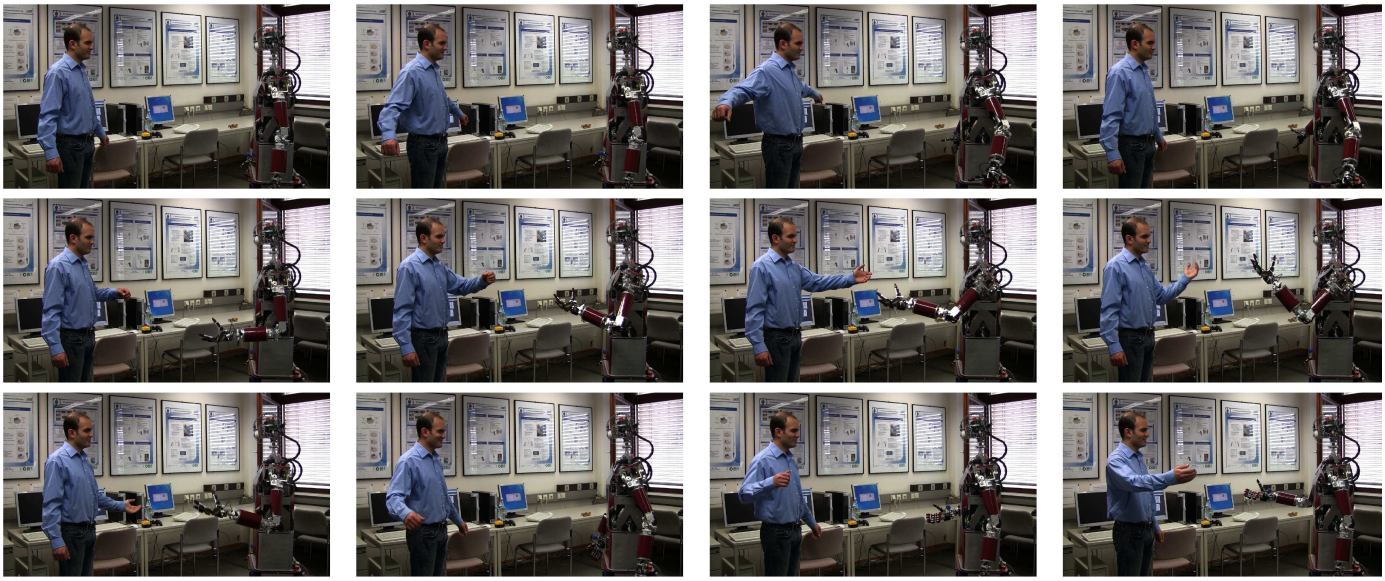

Fig. 7. Image samples of the online imitation of human motion by the humanoid ARMAR-IIIb.

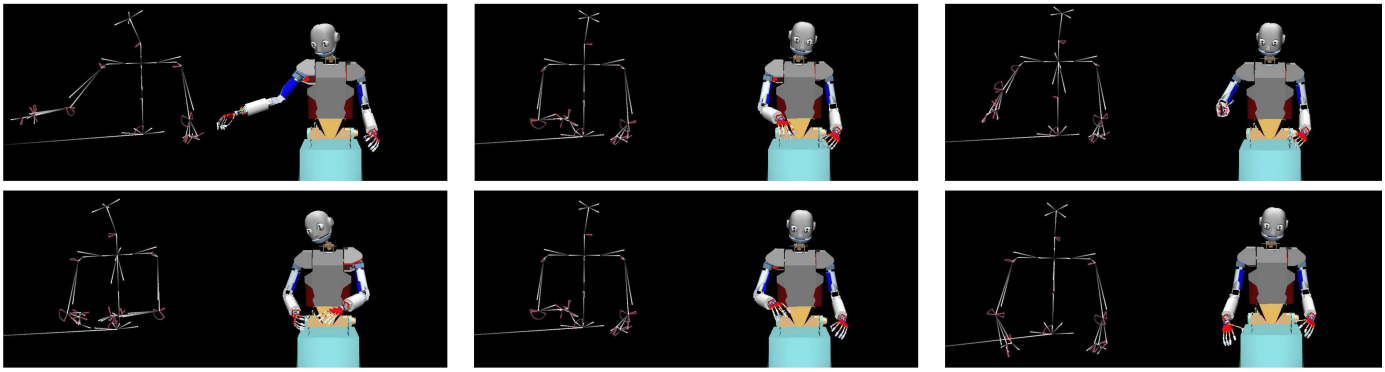

Fig. 8. Image sequence of a cutting trajectory captured by the Vicon system and the reproduction in the ARMAR III simulation.