# Motion Recognition and Recovery from Occluded Monocular Observations

Dongheui Lee[a,*], Yoshihiko Nakamura[b]

[a]*Institute of Automatic Control Engineering, Technical University of Munich, Germany*
[b]*Department of Mechano-Informatics, The University of Tokyo, Japan*

**Abstract**

This paper proposes a method for 3D whole-body motion recovery and motion recognition from a sequence of occluded monocular camera images based on statistical inference using a motion database. In the motion database, each motion primitive (e.g., walk, kick, etc) is represented in an abstract statistical form. Instead of extracting rich information by expensive computation of image processing, we propose an inference mechanism from low level image features (e.g., optical flow), inspired by psychological research on how humans perceive motion. The proposed inference mechanism recovers the 3D body configuration and finds the closest motion primitive in the motion database. Observations in 2D camera image space can be recognized even though the motion database is prepared in a different space (such as joint space) by coordinate transformation of the statistical motion representation. The approach is view invariant since the demonstrator's baselink position and orientation with respect to camera coordinates are tracked using an extended particle filter. Finally, an experimental evaluation of the presented concepts using a 56-degree-of-freedom articulated human model is discussed.

*Keywords:* statistical inference, motion recognition, motion recovery, motion capturing, optical flow, particle filter, monocular vision

## 1. Introduction

Motion understanding of human movements from a camera system which is mounted on a robot is important for realizing smooth and practical human-robot interaction. Although a studio-type motion capture system with several cameras provides good tracking accuracy, the system is expensive and requires a large set up in the environment. Also, human subjects have to wear optical markers on their body and motions can be captured only in the studio. Although a wearable type of a motion capturing system can eliminate space restriction, subjects still have to wear sensors on their bodies. Thereafter, it is inconvenient to use them in daily life environments. Therefore, a new technology for human motion understanding using onboard camera systems seems beneficial for seamless human robot interaction.

Perception of human motion has been studied in psychology [1][2][3][4][5] in the framework of moving light display (MLD). The moving light display is an experimental setup to show a human motion by lights attached to various parts of the body. These studies report that human can recover and understand three-dimensional human movements from the video while a single static image of the lights is insufficient to find the human shape. The experiments show that humans have high sensitivity to human motion perception and can recover 3D motion from a temporal sequence of images without any structural information. Human motion perception includes spatial and temporal understanding. This suggests that humans use the temporal information and the memory of human motions to recover missing spatial information.

With the final goal of capturing three-dimensional human motions and recognizing action classes from an onboard camera system, we focus on an inference mechanism from 2D optical flow without structure information[1]. An approach is proposed to use a human motion database of 3D motion to solve lack of depth information of a monocular camera and occlusion problems. Even a stereo camera system may suffer from depth insensitivity[2]. Although the authors assume a single onboard camera system composed of a monocular camera, the approach can be extended to an onboard stereo vision system. While recently 3D cameras became popular, the proposed technology has benefits for recovering 3D information from 2D video images like film archives as well as smart surveillance systems.

The main contribution of this paper is 3D whole body motion recovery from an occluded monocular image sequence, which includes not only self occlusion but also occlusion by obstacles. The following paragraphs summarize the technical characteristics of the proposed method.

(1) *Coordinate transformation of the statistical database*: In this work, human motion patterns in the database are represented by a time sequence of joint variables and the 3D position/orientation of the basebody[3] to allow for easy control of

*Corresponding author
*Email addresses:* dhlee@tum.de (Dongheui Lee), nakamura@ynl.t.u-tokyo.ac.jp (Yoshihiko Nakamura)

[1]The kinematic structure of human is invisible.

[2]Even an onboard stereo camera may not achieve complete 3D information of an object far in the distance because of its fixed baseline.

[3]To be precise, our motion database is represented in joint angles, joint ve-

articulated body motions. For the human motion database, the hidden Markov model (HMM) is adopted [6][7] because it uses a concise representation of spatiotemporal patterns and has well established computational methods. In order to recognize human motions (2D image observations from onboard camera) without the need of a database with many different views, we propose a method to transform the statistical database to an appropriate coordinate. By the coordinate transformation, the HMMs in joint space can be compared with 2D images from any view point without the need of depth information.

(2) *Concurrent motion recovery*[4] *and motion recognition*[5]: One can find many publications of motion recovery [8][9][10] and motion recognition [11][12][13] as independent problems. In contrast, our algorithm emphasizes that recovery and motion recognition are tightly coupled in a single framework, where recovery assists action recognition and vice versa. The inference cost for motion recognition in a next time step is significantly reduced by closing the computational loop using recovered motion. Computational concurrency of motion recovery and motion recognition is similar to that of localization and mapping in SLAM (Simultaneous Localization and Mapping) [14][15].

(3) *Inference from optical flow of feature points*: The appearance of people in images varies due to different clothing and lighting conditions [16]. Often used image descriptors include silhouettes [8][17], edges [18][19], color [20], and motion [21][22]. A large computation for image processing of the 2D image sequence would maximally extract information for 3D recognition. Instead, this paper focuses on development of an inference method from low level image features (e.g., optical flow [23] of unlabeled features) without shape and structure information, inspired by human's high perception ability shown in the MLD experiments [1]. Note that the main objective of this paper lies on the inference mechanism from partial monocular observations. In contrast, the reliable feature selection and robust optical flow calculation from blurred images are not the focus of this research. Such methods for image processing (optical flow estimation) can be found in [24][25]. Therefore, to separate these problems in our experiments, we attach artificial markers to the subject as distinctive feature points. Note also that the markers are placed at arbitrary points and neither labeled nor tracked, in contrast to optical markers in conventional motion capturing. Thanks to these properties of random placement of markers, and no need of tracking and labeling, the synthetic observations can be easily replaced with the optical flows from real images. Therefore this allows that the proposed inference method can be directly integrated with 2D optical flows processed from real images.

(4) *Mimesis model*: The basic framework used in this work is the *mimesis model* [6], which was inspired by the mimesis theory [26] and the mirror neurons [27] in cognitive and neuro science. The mimesis model was proposed for imitation learning from human demonstrations, which consists of three components: motion learning, recognition, and generation. This model has been selected because the use of the mimesis model for 3D recovery of human motion patterns may be natural if we recall the fact that our skill of human motion perception is based on tightly connected cognitive activity with learning and reproduction.

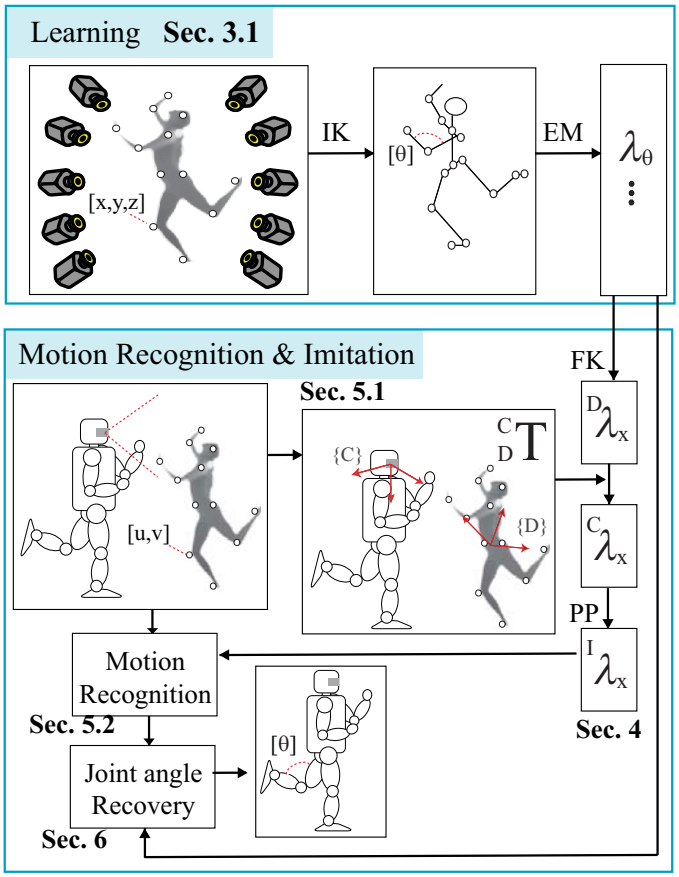


Figure 1: Overall architecture of the proposed method.

The overall data flow is shown in Fig. 1, consisting of the learning procedure and the 3D whole body motion recovery from 2D images. First, during the learning stage, a human performs multiple demonstrations for each motion primitive using conventional motion capturing system. The observed three dimensional Cartesian marker position data [x,y,z] on the human body is converted to joint angle data for a chosen kinematic model[6] using inverse kinematics. The observations in the joint angle space are embodied into the parameters of an HMM (Section 3). The transformation matrix ${}^{C}_{D}T \in SE(3)$ between the camera coordinates and the demonstrator baselink coordinates is found by applying the extended particle filtering algorithm (Section 5.1). In Section 4, the coordinate transformation of the statistical database is described. Motion primitives $\lambda$ are converted from the demonstrator's joint coordinates $\lambda_\theta$ into the

locities, and baselink velocities.

[4]Motion recovery denotes estimation of the sequence of joint angles and basebody position/orientation from the 2D image sequence.

[5]Motion recognition denotes the search for the closest HMM (e.g. walk, run, jump, etc.) to the 2D image sequence.

[6]The kinematic model is chosen depends on an application: for example, a humanoid robot kinematic model for robot imitation of human motions and a human skeleton kinematic model for human motion reconstruction.

demonstrator's Cartesian coordinates ${}^{D}\lambda_x$ by forward kinematics, into the camera coordinates ${}^{C}\lambda_x$ by a transformation matrix ${}^{C}_{D}T$, and into 2D image Cartesian coordinates ${}^{I}\lambda_x$ by perspective projection. Finally, both proto-symbols ${}^{I}\lambda_x$ and observations ${}^{I}o_x$ are represented in the 2D image Cartesian coordinates. When all the markers are not visible, motion recognition from partial observations are carried out as described in Section 5.2. Section 6 explains how to recover 3D whole body motion close to the 2D observed motion.

Note that there are two stages of motion recovery in this work: one for human baselink position and orientation ${}^{C}_{D}T$ (6DOF) and the other for joint angles (50DOF). The particle filter represents a probabilistic distribution of ${}^{C}_{D}T$ and it influences coordinate transformation and thereafter motion recognition. Motion recognition results affect the prediction of particles at the next step and recovery of joint angles. In this regard, concurrent motion recovery and recognition is implemented in this work.

An earlier version of this work was presented in [28]. This work is extended by in depth explanations of methodology and new experimental results. A method to reproduce a motion sequence by manipulating proto-symbols in different coordinates is newly proposed. While the previous work showed a recovery result of only one occluded motion sequence, this paper provides statistical analysis under different conditions, such as multiple runs with different initialization, multiple motions for a kind motion type, different numbers of particles, etc.

## 2. Related Research

### *2.1. Learning from human demonstrations*

Imitation is considered as the most primitive and fundamental element of intelligence development for human beings [26]. Donald defined mimesis as the ability to produce conscious, self-initiated, representational acts that are intentional but not linguistic. Mimesis is the basis of human communication skills [26]. Moreover, there is evidence that mimicry and imitation play significant roles in the developmental stages of animals and human beings. Neuroscientists [27][29] reported the mirror neuron system in primates' brains that activates both during observation of other's motions and self execution of similar tasks. The neuroscientific evidence of motor primitives and mirror neurons inspired technical studies of imitation learning in robotics.

Imitation learning in robotics, often referred as *Programming by Demonstration*, provides a means of automatic programming of complex systems such as dexterous anthropomorphic robots without extensive trials or complex programming [30][31]. Bentivegna and Atkeson [32][33] used the idea of primitives for motor learning to play air hockey and marble maze. Billard and Matarić [34] used connectionist-based approaches to represent movements. Inamura et al. [6] proposed the HMM based mimesis model. The mimesis model encodes the time-series motion patterns as proto-symbol representations and decodes motion primitives from the proto-symbols. The authors [7] extended the mimesis model in order to recognize and imitate whole body motions from partial observations. A theory of human-robot communication was developed based on the mimesis model [35][36].

In research on imitation learning for humanoid robots, motion capturing systems are widely used [6][7][37][38][39] to acquire reference motion patterns, such as human beings' motion patterns. Most motion capturing systems use optical devices, consisting of reflective markers and multiple cameras [40]. Some imitation research [38] adopts wearable motion capturing systems. In both studio- and wearable-type systems, subjects have to wear optical markers or sensors on their bodies. Thereafter, they are inconvenient to use in daily life environments. In order to realize intuitive and practical interaction, a mimesis model using a simple onboard vision system on the humanoid robot is beneficial.

One technical challenging issue in such applications is how to deal with partial observations (i.e., incomplete depth information, self-occlusion and occlusion by surrounding objects). Ghahramani and Jordan [41] proposed using the expectation-maximization (EM) algorithm to fill in missing feature values of examples when learning from incomplete data for a classification problem. Humans show high robustness against incompleteness in sensing information: Although human eyes have low depth sensitivity, specially for objects at a distance, human beings can recognize and imitate other's motion, even when a part of his/her body is occluded; Humans can recognize 3D information from two-dimensional images, films and video archives. The uniqueness of our work lies on an inference mechanism based on mimesis model, which can recognize and reconstruct high dimensional human articulated movements from incomplete sensing data of an onboard vision system. In contrast to our previous work [42] where we proposed full body imitation from partial observations, this paper deals with a challenging problem where the input observation (images from an arbitrary view) and the output reconstruction (joint angles for an articulated body) are not in the same space. Also, we do not assume that the observation data is structured. A complete observation is not necessarily defined as a vector with one specific dimensionality, but varying dimensionality. For example, in the case we got a full body image without occlusion by external objects, the full body can be represented as 100 or 50 image features. This implies also that a complete observation representation vector cannot be uniquely divided into an observed and missing subvector. Therefore the algorithm should be able to handle a time-varying size of an observation.

### *2.2. 3D Motion Understanding from 2D Images*

There has been a great deal of research on vision-based human motion capture [43][44][8][45]. A good review of early works on motion understanding was made by Cedras and Shah [46]. A detailed discussion of shape matching can be found in [47]. See [48][45][16] for recent surveys on vision-based human motion analysis. Due to the wide range of studies in computer vision on this topic and the limited space, we focus our survey to the most relevant works.

Full 3D pose reconstruction from single view images is a considerably difficult and ill-posed problem, compared to the

problem of 2D pose estimation or 3D pose estimation from multiple views. To resolve the inherent ambiguity in monocular images, additional constraints on kinematics and movement are typically employed [21][49][50][51][52]. Taylor [51] and Barron & Kakadiaris [52] proposed reconstruction methods from a single uncalibrated image by considering the foreshortening of body segments in the image under assumption of scaled orthographic projection. Our method resolves the problem by having a predefined 3D human articulated model and a motion database. In our motion database, each motion primitive embodies temporal and spatial variability of the motion. Namely, kinematics constraints, and spatiotemporal constraints of learned motions are applied to resolve the depth ambiguities of monocular images. We design the human articulated model by an average of human body data, and scaled it with the ratio between the real human's height and the model's height so that the both have the same height. The articulated model does not need to be exactly same as the real human demonstrator.

The learning based methods [53][54][55][56][8] estimate poses using exemplars. Brand [54] modeled a dynamical manifold of human body configurations with an HMM, learned it using entropy minimization, and used it to infer 3D body pose from 2D shadows. Rosales and Sclaroff [57] employed a learned mapping function from silhouettes to 2D poses by a neural network. Agarwal and Triggs [56] proposed a learning based method for 3D human pose recovery from a single 2D image, using silhouettes. The relationship between the training dataset of silhouette histograms and 3D poses is learned by the relevance vector machine regressor on Kernel base. Rather than learning the human motion, Sidenbladh et al. [9] developed an implicit probabilistic model, searching binary trees among exemplar, based on the coefficients of a low-dimensional approximation to human motion data. Sminchisescu and Triggs [10] recovered 3D human body motion from monocular video sequences based on an image matching metric and a sample-and-refine search strategy. The image matching cost metric is designed carefully combining optical flow, edge energy, and motion boundaries. In Ramanan et al. [18], under the assumption that people tend to take on certain canonical poses, a human detector is built by using a pictorial structure model on an edge-based representation in lateral view of walking.

Among research for motion recognition, Yamato et al. [12] proposed a motion recognition method from silhouettes using HMMs. Davis and Bobick [11] proposed a motion recognition method using two kinds of synthesized images, namely binary motion energy image and motion history image. This method is viewpoint dependent. In the work of Chomat and Crowley [58], action recognition is processed statistically according to the conditional probability that a measure of the local spatio-temporal appearance is occurring for a given action. The measure of spatio-temporal structure is computed based on Gabor energy filters. Multi-dimensional histograms of these measures are used to estimate the probability of an action. Yang et al. [59] proposed a method for gesture spotting and recognition problem using stereo cameras. Manually segmented gestures are trained into Gesture HMMs. The rest motions besides gestures are modeled as transition gesture HMM. Given the trajectories of 3D body parts positions, gesture spotting and recognition is solved by using HMM classifier. In [60] [59] [61], the 3D reconstruction (or tracking) problem and the recognition problem are completely separated: After solving the former completely, the latter starts. In [59], 3D position of labeled body parts are estimated from stereo cameras by applying a reconstruction method in [62]. In [60] and [61], head and hands are tracked from stereo images by using 3D colored blob tracking method. Afterwards the trajectories of labeled body parts are used for detecting pointing gesture [61] and for recognizing hand gestures [60] and whole body motion [59].

The above mentioned works successfully reported either 3D pose recovery or motion recognition as independent problems. In some research, the two problems are treated in a sequence. However, if the algorithm for the first problem fails, the following problem also cannot be successful as a consequence. Although the two problems are strongly interconnected to each other, it is hard to find previous works solving both problems simultaneously. In contrast, we aim to solve the both problems simultaneously, where recovery assists action recognition and vice versa, because in most cases recovery is difficult if we do not know activity clustering (recognition) and also activity recognition is difficult if we do not know 3D whole body information.

Similar approaches to ours are carried out in [63][20][22]. Lu et al. proposed methods for tracking and action recognition which are coupled. In [63], an athlete is represented by the PCA-HOG descriptor and tracked by a particle filter. Based on the tracking result, action is recognized by the forward-backward algorithm of an HMM and a new template for tracking is updated based on the Viterbi algorithm. In [20], their method is extended for multiple people tracking. However, their method is limited to 2D tracking, not 3D recovery.

Fathi and Mori [22] developed a motion-exemplar approach for tracking human figures. Similar to ours, they infer the pose of the human figure by finding a sequence of exemplars which matches a given input sequence, and then estimating body joint positions using these exemplars. Similarity between an exemplar sequence and an input image sequence is calculated by the motion consistency measure introduced in [64]. However, the method requires manual labeling, and cannot solve the action recognition problem.

## 3. Mimesis Model from Partial Observation

### *3.1. Mimesis Model*

The mimesis model as shown in Fig. 2 was introduced by Inamura et. al [6], inspired by the mirror neuron system [27]. It is a bidirectional computational model which performs three functions; motion learning, motion recognition, and motion generation with the concept of proto-symbols. The proto-symbols are defined through the HMM parameters.

The advantages to have proto-symbols are summarized as follows. The compactness of the representation is an efficient computational strategy for large information spaces in the real world. The learned proto-symbols are easy to reuse for recognition of other's motion and generation of self motion.

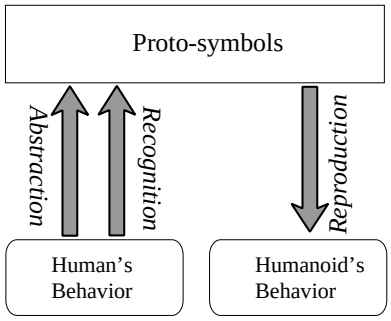


Figure 2: The conceptual diagram of the mimesis model. It is a bidirectional model which performs learning, recognition and generation functions through proto-symbols. The proto-symbols are defined through the HMM parameters.

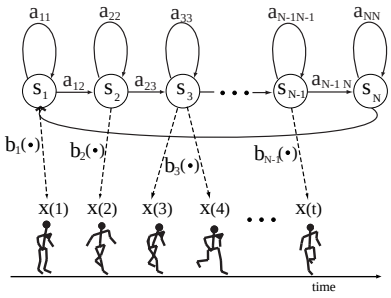


Figure 3: Periodic continuous HMM

*Motion Learning*: The training data is provided from the human demonstrator's motions. Motion learning means proto-symbol acquisition that involves segmentation of observed motion data and stochastic modeling of each segment. In this work, manually segmented motions are used in the learning. Nevertheless, the segmentation process can be easily automated by recent segmentation algorithms [65][66][67][68][69]. In Takano and Nakamura's method [67], primitives are specified by the designer a-priori, and segmentation is based on the comparison between the known motions and the incoming data. In Fod et al. [66], a segmentation point is recognized when a zero velocity crossing is detected in a sufficient number of dimensions. Janus, Kulić, and Nakamura [68][69] investigated the use of the Kohlmorgen and Lemm algorithm [65] for unsupervised segmentation of on-line human motion data. After the segmentation process, the inherent dynamics of the segmented motion is modeled by an HMM, which is known as an efficient stochastic model for spatiotemporal motion data.

A proto-symbol corresponds to a set of parameters of an HMM $\lambda = \{A, B, \pi\}$. The vector $\pi = \{\pi_i\}$ is the initial state probability vector, where $\pi_i$ is the probability for the initial state to be state $i$. The matrix $A$ is the state transition probability matrix $A = \{a_{ij}\}$, where $a_{ij}$ is the probability of transition from state $i$ to state $j$. $B = \{b_i\} = \{c_{ij}, \mu_{ij}, \Sigma_{ij}\}$, where $b_i(o)$ is the probability density function for the output of continuous vector $o$ at state $i$, is represented with a mixture of Gaussian distributions. The function $b_i$ consists of the weight $c_{ij}$, mean vector $\mu_{ij}$, and covariance matrix $\Sigma_{ij}$ for the $j$-th mixture component at state $i$. For simplicity, we write $B = \{c, \mu, \Sigma\}$. Since many human motions are cyclic, periodic continuous HMMs as shown in Fig. 3 are used. The proto-symbols are obtained via the Baum-Welch algorithm [70]. The motion database consists of a set of acquired proto-symbols.

*Motion Recognition*: Motion recognition is to identify a proto-symbol from the motion database that has the highest likelihood for generating the observed motion. Unlikely to the learning process, motion segmentation is not prerequisite for motion recognition. Instead, a continuously incoming motion stream is observed through a fixed time window. Let $O = \{o_{t-w}, \cdots, o_t\}$ represent an observed motion through the $w + 1$ width window at each sampling time $t$. The likelihood $P(O|\lambda)$ to generate the fixed time window sequence $O$ for each proto-symbol $\lambda$ is calculated. The proto-symbol that provides the largest $P(O|\lambda)$ is the result of the motion recognition.

*Motion Generation*: During motion generation, a motion pattern is decoded from a selected proto-symbol by using the expectation operator in the stochastic model. The motion generation has two processes: generation of the state sequence and generation of the output motion. The state sequence is decoded from the state transition probability matrix $A$ and the initial state probability vector $\pi$. Once the state sequence has been generated, the output motion at each state is decoded from the observation emission probability distribution $B$. In order to eliminate the artifacts caused by discrete state switching in the HMM, a Monte Carlo sampling based technique [6, 42] and Gaussian regression based technique [71, 72] have been proposed. Then, the generated smooth motion trajectory can be used as a command input for a humanoid robot or a human figure in animation.

The fundamental imitation mechanism for a humanoid robot (or a human figure) using the mimesis model is described below. A humanoid robot has the motion database which consists of hidden Markov models (HMMs). When the robot observes a human's motion, it compares the observation to each HMM in the database and finds the best matching HMM. By generating its own motion from the best HMM, the imitation of the observed motion is realized. The mechanism enables the robot to classify others' motions and to generate its own motions using knowledge called the motion database. The imitation mechanism based on mimesis model is different from simple mimicry, where human motions are simply mapped to the robot kinematic model without an inference mechanism. In our imitation mechanism based on mimesis model, a different representation of proto-symbols is chosen depending on the application. When the HMMs in the robot kinematics are used, the imitation inference of a robot can be achieved. When the HMMs in the human skeleton model is used, the reconstruction of the human motion can be realized.

### *3.2. Mimesis Model from Partial Data [7]*

In the original mimesis model [6], a motion is generated only from the HMM with the highest likelihood. The generated motion pattern is simply one of the memorized motion patterns by the proto-symbols.

Active use of stochastic models can enrich the repertoire of the motion patterns [73][7]. One extension can be the *proto-*

*symbol based motion duplication* method [7] which can perform close motion imitation of the partially occluded observation. The method consists of two step procedures: *motion recognition from partial observations* and *proto-symbol based duplication of an observed motion*. When the human motion is partially visible, only visible parts are compared to the corresponding parts of HMMs in the motion database. By estimating the optimal state sequence corresponding to the observation using the Viterbi algorithm, the imitated motion pattern is temporally synchronized with the observation. Different motion patterns corresponding to the same proto-symbol can be imitated with different temporal sequences. This allows situated motion generation by temporal synchronization.

### *3.3. Mimesis Model from Monocular Observation*

Understanding of human movements from an onboard camera system is relevant to imitation from partial observations. In general, imitation from partial observations is more complicated than that of section 3.2 because observed sensory data may not be in a desired form.

**Observations**: Being inspired by human's highly sensitive perception, as shown in the moving light display experiments [1], we focus on an inference algorithm based on the 2D optical flow, which shows the velocities of distinctive feature points by arrows computed from two succeeding images. Calculation of robust optical flow from blurred images of a moving camera are not our main interest in this work. Thus, we assume that 2D optical flow is calculated in advance and focus on inference mechanism for 3D motion recovery from 2D optical flow. The observation can be written as ${}^{I}o_x$, where $\{I\}$ indicates "2D Image" and $x$ denotes "Cartesian space."

**Motion Database**: A human motion pattern is represented by a time sequence of joint angles, joint velocities, and velocities of the basebody position/orientation. A segmented human motion sequence in joint space is embodied into a proto-symbol which is represented by the parameters of an HMM $\lambda_\theta = \{A_\theta, B_\theta, \pi_\theta\}$. Note that $\lambda_\theta$ is a proto-symbol in the joint space, where $\theta$ denotes "joint space." The humanoid robot uses the motion database, consisting of a set of proto-symbols, in order to solve depth and occlusion problems.

The overview of the proposed method is described in Fig. 1. Because an observed human motion of a monocular image sequence cannot be compared to the motion database in joint space directly, the motion database is converted to images as seen from camera viewpoint. The basebody position/orientation of the demonstrator is estimated by a particle filter and used for the coordinate conversion. The partial observation is compared to the converted HMMs and the best matching HMM is found. Then, by using the *proto-symbol based duplication of observed motion*, 3D full-body motion is imitated. This whole-body imitation from a monocular image sequence can be interpreted as the 3D motion recovery from the 2D image sequence. The algorithm is summarized as follows:

(1) $\lambda_\theta \rightarrow {}^{D}\lambda_x$: The proto-symbols are converted from the demonstrator's joint coordinates into the demonstrator's Cartesian coordinates by forward kinematics. (Sec. 4.2)

(2) The demonstrator's basebody position/orientation ${}^{C}_{D}T \in SE(3)$, which is the homogeneous transformation matrix between the camera coordinates and the demonstrator coordinates, is estimated via a particle filter. (Sec. 5)

(3) ${}^{D}\lambda_x \rightarrow {}^{C}\lambda_x \rightarrow {}^{I}\lambda_x$: The proto-symbols in the demonstrator's Cartesian coordinate ${}^{D}\lambda_x$ are transformed into the camera coordinates ${}^{C}\lambda_x$ by the transformation matrix ${}^{C}_{D}T$ (section 4.3). By perspective projection, the proto-symbols in the camera Cartesian coordinates ${}^{C}\lambda_x$ are converted to proto-symbols in the image Cartesian coordinates ${}^{I}\lambda_x$ in the same way as in section 4.3.

(4) After converting the proto-symbols into the same space as the observations, the optical flow in the 2D-transformed proto-symbol ${}^{I}\lambda_x$, corresponding to the observed optical flow, is calculated (Sec. 5.3.3). The computed optical flow in the converted proto-symbol and the observed optical flow are compared to calculate the likelihood $P({}^{I}o_x|{}^{I}\lambda_x)$. The particle filter for the demonstrator's basebody position/orientation is updated based on the likelihood and the motion is recognized as the HMM which has the highest likelihood.

(5) By adopting the *proto-symbol based motion duplication* [74], a temporally synchronized motion pattern to the observed motion pattern is generated. Section 6 describes how to extend the algorithm for the 3D motion recovery in joint space from a 2D image sequence.

## 4. Coordinate Transformation of Proto-Symbols

### *4.1. Coordinate Transformation*

Proprioception is an internal sense of the relative position of neighboring parts of the body and exteroception sense is the perception of the outside world through, for example sight, taste, smell, touch, and hearing. One can learn a motion from other people by relating exteroception to proprioception and vice versa. Previous works of imitation research in robotics [6][75][76][77][7] have not considered such conversions and assumed same modality for simplicity.

This section proposes *coordinate transformation of proto-symbols*, which implies the conversion between proprioception and exteroception of the human. A proto-symbol is a probabilistic form of a motion pattern and is represented by HMM parameters $\lambda = (A, B, \pi)$, as explained in section 3.1. When converting a proto-symbol $\lambda = \{A, B, \pi\}$ into different coordinate spaces, the main difference among the proto-symbols in the different spaces is the representation of motion patterns. Thus, the state transition probability matrix $A$ and the initial state probabilities vector $\pi$ can stay unchanged. Only the output probability distribution $B = \{c, \mu, \Sigma\}$ is to be transformed, where $c$, $\mu$, and $\Sigma$ are the weight scalar and the mean vector and the covariance matrix for each Gaussian respectively. It is assumed that the output probability is represented by a mixture of Gaussians even after coordinate transformation.

Linear conversion by homogeneous transformation matrix is detailed in section 4.3. Nonlinear conversion by kinematics is explained in section 4.2. Conversion by perspective projection is represented by the homogeneous transformation matrix discussed in section 4.3, if camera distortion is ignored. When

considering camera distortion, this perspective projection is carried out as a nonlinear conversion similarly to section 4.2. Here, the camera is supposed to be calibrated.

### *4.2. Proto-Symbol Conversion by Kinematics*

This section considers how to convert a proto-symbol $\lambda$ from the joint space $\lambda_\theta = \{A_\theta, B_\theta, \pi_\theta\}$, $B_\theta = \{c_\theta, \mu_\theta, \Sigma_\theta\}$ to the Cartesian space $\lambda_x = \{A_x, B_x, \pi_x\}$, $B_x = \{c_x, \mu_x, \Sigma_x\}$ by forward kinematics. Note that the proto-symbol in joint space $\lambda_\theta$ contains information of joint angles and joint angular velocities. The converted proto-symbol $\lambda_x$ includes information (mean vector and covariance matrix) of feature positions and their translational velocities. In the following, two complementary approaches for conversion by kinematics are given in the case of small and large covariance.

#### *4.2.1. Monte Carlo method*

When the covariance $\Sigma_\theta$ is large, the mean vector $\mu_x$ and covariance matrix $\Sigma_x$ are calculated using the Monte Carlo method. The Monte Carlo method estimates a continuous probability distribution function by using discrete samples. Samples $o_\theta = \{o_{\theta i}\}$ are generated from the probability distribution $B_\theta = \{c_\theta, \mu_\theta, \Sigma_\theta\}$ where $o_{\theta i}$ denotes the $i$-th sample of $o_\theta$. Each sample is converted from the joint space (joint angles and joint angular velocities) to the Cartesian space (feature positions and their translational velocities) by forward kinematics $f$ and the uncertainty model of kinematics $\epsilon(f)$ by

$$o_{xi} = f(o_{\theta i}) + \epsilon(f), \ \forall i = 1, ..., N_s \tag{1}$$

where $N_s$ is the number of samples. The kinematic uncertainty model $\epsilon(f)$ due to computational errors is designed as a zero-mean Gaussian distribution.

In eq. (1), $o_{xi}$ denotes the $i$-th sample which is converted into the Cartesian space. Then, the converted mean vector and covariance matrix are calculated from a set of samples $o_x = \{o_{xi}\}$. For a single Gaussian model, the Gaussian distribution becomes

$$c_x = c_\theta = 1 \tag{2}$$

$$\mu_x = \frac{1}{N_s}\sum_i o_{xi} \tag{3}$$

$$\Sigma_x = \frac{1}{N_s}\sum_i (o_{xi} - \mu_x)(o_{xi} - \mu_x)^T. \tag{4}$$

For a Gaussian mixture model [78], a clustering method [79] is applied. The number of Gaussians becomes the number of clusters and the weighting scalar of each Gaussian is proportional to the number of samples in the corresponding cluster. For each Gaussian, the mean vector and the covariance matrix are calculated from the samples in the corresponding cluster. Calculation via the Monte Carlo method is simple. Its computational cost is proportional to the desired accuracy, namely the number of samples.

#### *4.2.2. Linear approximation method*

Although the kinematic model is a nonlinear function, if the covariance $\Sigma_\theta$ is small enough, it can be approximated as a linear function. In such a case, the mean vector $\mu_x$ is calculated from $\mu_\theta$ by eq. (5). The covariance matrix is converted by eq. (6) using the Jacobian matrix of the forward kinematics at the mean vector.

$$\mu_x = f(\mu_\theta) \tag{5}$$

$$\Sigma_x = J(\mu_\theta)\Sigma_\theta J(\mu_\theta)^T \tag{6}$$

where

$$J(\theta) = \frac{\partial f(\theta)}{\partial \theta} \tag{7}$$

Most human motions in our experimental dataset show that the standard deviation of proto-symbols $\lambda_\theta$ are less than 0.122 rad. This would justify applying the linear approximation method to the data.

### *4.3. Proto-Symbol Conversion by Homogeneous Transformation Matrix*

In order to handle the cases where a robot (an onboard camera) and/or a human subject moves, the algorithm should be view-point independent. Therefore, the proto-symbol conversion by homogeneous transformation matrix is proposed.

With the homogeneous transformation matrix between the demonstrator's basebody Cartesian coordinates and the camera Cartesian coordinates[7], a proto-symbol ${}^D\lambda_x = \{{}^DA_x, {}^DB_x, {}^D\pi_x\}$, ${}^DB_x = \{{}^Dc_x, {}^D\mu_x, {}^D\Sigma_x\}$ is converted to ${}^C\lambda_x = \{{}^CA_x, {}^CB_x, {}^C\pi_x\}$, ${}^CB_x = \{{}^Cc_x, {}^C\mu_x, {}^C\Sigma_x\}$. Parameter $A$ and $\pi$ stay unchanged, ${}^CA_x = {}^DA_x$ and ${}^C\pi_x = {}^D\pi_x$. If $B$ is composed of one Gaussian, the weight scalar for the Gaussian becomes 1, ${}^Cc_x = {}^Dc_x = 1$. Because conversion by the homogeneous transformation matrix ${}^C_DT$ is linear, the mean vector and the covariance matrix are converted as follows.

$$\begin{bmatrix} {}^C\mu_{x_i} \\ 1 \end{bmatrix} = {}^C_DT \begin{bmatrix} {}^D\mu_{x_i} \\ 1 \end{bmatrix} \tag{8}$$

$${}^C\Sigma_{x_i} = {}^C_DR\,{}^D\Sigma_{x_i}\,{}^C_DR^T \tag{9}$$

where $\mu_{x_i} \in R^3$ and $\Sigma_{x_i} \in R^{3\times3}$ are the mean vector and the covariance matrix of the $i$-th feature's 3D Cartesian coordinates. Because the covariance matrix is not related to the translation vector, the covariance matrix is calculated by considering the rotation parts ${}^C_DR \in SO(3)$ of ${}^C_DT$.

## 5. Motion Recognition and Basebody Position/Orientation Estimation

### *5.1. Particle Filter for Baselink Pose Estimation*

In order to estimate the demonstrator's basebody position/orientation ${}^C_DT \in SE(3)$, a particle filter is implemented. Based on Markov assumption, the particle filter is represented as

$$p(s_t|o_{1:t}, a_{1:t}) = \eta p(o_t|s_t) \int p(s_t|a_{t-1}, s_{t-1}) p(s_{t-1}) ds_{t-1} \ . \tag{10}$$

[7] Calculation of this transformation will be given in section 5.

- $s_t$: Demonstrator's baselink position/orientation, ${}^{C_t}_{D_t}T$, with respect to the onboard camera at time $t$, is represented in a probabilistic way. Its probabilistic distribution is represented with a particle set $s_t = \{s_{j,t}\}, \forall j = 1..N_p$, where $N_p$ is the number of particles. Each particle $s_{j,t}$ is a 6 dimensional vector which represents the basebody position and orientation.

- $a_{t-1}$: $a_{t-1}$ is composed of the camera's movement ${}^{C_t}_{C_{t-1}}T$ and demonstrator's action ${}^{D_{t-1}}_{D_t}T$. Here $C_t$ denotes camera coordinate at time $t$ and $D_t$ denotes demonstrator coordinate at time $t$.

- $o_t$: An observation at time $t$ is represented as $o_t$.

In the paper, we call the case that particles are converged locally *local tracking* and the case when the particles are spread widely *global estimation* [15]. At the initial step, if demonstrator's initial basebody position/orientation is known, particles are located near the given position with small noises. Otherwise, particles are spread globally.

#### 5.1.1. Motion Model $p(s_t|a_{t-1}, s_{t-1})$

The motion model of the demonstrator's position/orientation with respect to the humanoid position/orientation represents the motion uncertainty $a_{t-1}$ based on the measurement of the camera's movement ${}^{C_t}_{C_{t-1}}T$ and the expectation of demonstrator's action ${}^{D_{t-1}}_{D_t}T$.

$$p(s_t|a_{t-1}, s_{t-1}) \longleftarrow {}^{C_t}_{D_t}T = {}^{C_t}_{C_{t-1}}T\,{}^{C_{t-1}}_{D_{t-1}}T\,{}^{D_{t-1}}_{D_t}T \tag{11}$$

The camera's movement can be roughly estimated from the motor command of the robot[8]. Strictly speaking, the demonstrator action ${}^{D_{t-1}}_{D_t}T$ is not observable. However, when the demonstrator's motion is known, the next movement can be predicted. Therefore, during local tracking, the demonstrator's action is modeled as the human's basebody motion of the recognized proto-symbol, which has the highest likelihood.

$${}^{D_{t-1}}_{D_t}T = {}^{D_{q(t-1)}}_{D_{q(t)}}T \tag{12}$$

Herein $q(t)$ denotes the state of the recognized proto-symbol (HMM) at time $t$. On the other hand, when the demonstrator's motion is unknown, the demonstrator's action ${}^{D_{t-1}}_{D_t}T$ is hard to predict. Therefore, during global estimation, it is modeled as a possible random movement within a certain time period.

#### 5.1.2. Sensor Model $p(o_t|s_t)$

Belief $p(s_{j,t})$ is the probability that the current demonstrator position/orientation is the $j$-th particle $s_{j,t}$ at time $t$. For each particle, the belief $p(s_{j,t})$, $\forall j = 1..N_p$ is calculated by

$$p(s_{j,t}) = p(o_t|s_{j,t}) \tag{13}$$

$$p(o_t|s_{j,t}) \propto \max P(O|{}^I\lambda_{x,ij}), \quad \forall i = 1..N_\lambda \tag{14}$$

where $O = \{o_{t-w}, \cdots, o_t\}$ is a time-sequence of observation through the window, whose width is $w + 1$ frames, at time $t$. The belief $p(o_t|s_{j,t})$ is the maximum likelihood among all proto-symbols by eq. (14), where $i$ is the index of a proto-symbol and $N_\lambda$ is the number of proto-symbols. The $i$-th proto-symbol $\lambda_{\theta,i}$ is converted to the image Cartesian coordinates by forward kinematics, the $j$-th particle, and perspective projection. Namely, ${}^I\lambda_{x,ij}$ represents the converted $i$-th proto-symbol by the $j$-th particle. In the rest of this section, ${}^I\lambda_{x,ij}$ is written as $\lambda_{ij}$ for the simplicity reason. The term $P(O|\lambda_{ij})$ is the likelihood to generate observation $O$ from the converted proto-symbol $\lambda_{ij}$. The detailed calculation of $P(O|\lambda_{ij})$ is explained in section 5.3. Equation (13) is substituted into eq. (10). The belief $p(s_{j,t})$ is normalized by $\eta$ so that

$$\sum_{j=1}^{N_p} p(s_{j,t}) = 1 \ . \tag{15}$$

### 5.2. Motion Recognition

The humanoid robot recognizes the demonstrator's motion by identifying the best proto-symbol in the motion database. Time series of motion data $O = \{o_{t-w}, \cdots, o_t\}$ are observed through a fixed width window. The best matching proto-symbol for the observation is found by calculating eq. (16) for all proto-symbols and all particles.

$$\lambda^* = \arg\max_{\lambda_{ij}} P(O|\lambda_{ij}) \tag{16}$$

where $i$ is the index of a proto-symbol and $j$ is the index of a particle.

### 5.3. Calculation of $P(O|\lambda_{ij})$

#### 5.3.1. Time-sequence of Demonstrator's Position/Orientation

For the demonstrator's position/orientation estimation (eq. (13)) and motion recognition (eq. (16)), $P(O|\lambda_{ij})$ should be calculated for each particle ($\forall j = 1..N_p$) and each proto-symbol ($\forall i = 1..N_\lambda$). Note that, in order to calculate $P(O|\lambda_{ij})$, the time-sequence of observation $O = \{o_{t-w}, \cdots, o_t\}$ and the time-sequence of each particle $S_j = \{s_{j,t-w}, \cdots, s_{j,t}\}$ are necessary. In the conventional particle filter, the particles represent only a probabilistic distribution of the current demonstrator position/orientation, namely $s_t = \{s_{j,t}\}, \forall j = 1..N_p$. Also, the sensor model is updated based on observations at the current time step. In the proposed approach, the time-sequence of the $j$-th particle $S_j = \{s_{j,t-w}, \cdots, s_{j,t}\}$ is calculated backward from time $t$ to time $t - w$ using camera movement and demonstrator action of the proto-symbol.

During **local tracking**, past time-sequences of all particles are set to the time-sequence of the champion particle in the past. The champion particle denotes the particle with the highest likelihood among all particles.

$$s^*_{j,m} = \arg\max_{s_{k,m}} p(s_{k,m}), \forall k = 1..N_p, \forall m = t-w, ..., t-1 \tag{17}$$

where $N_p$ is the number of particles. Therefore, all particles have the same past trajectory apart from current time $t$.

[8] We assume that the onboard camera coordinates on the robot with respect to the robot's base link coordinates are known.

During **global estimation**, the time-sequence of the each particle $S_j = \{s_{j,t-w}, \cdots, s_{j,t}\}$ is calculated from each proto-symbol $\lambda_i$ by the following initialization and induction process.

*Initialization*: The proto-symbol $\lambda_i$ is converted into 2D Cartesian space by using the particle $s_{j,t}$ at time $t$ and the converted proto-symbol is represented as $\lambda_{ij}$. Let ${}^I b_k(o_t)$ denote the probability to generate $o_t$ from the output probability distribution of the $k$-th state in the 2D-converted $\lambda_{ij}$. Thereafter, the best matching state $q(t)$ at time $t$ is found by

$$q(t) = \arg\max_k {}^I b_k(o_t), \forall k = 1..N \ , \tag{18}$$

where $N$ is the number of states in HMM $\lambda_{ij}$.

*Induction*: From the best matching state $q(t)$, the basebody position/orientation at one-step prior $s_{j,t-1}$ is estimated using the demonstrator motion ${}^{D_{q(t-1)}}_{D_{q(t)}}T$ in state $q(t)$[9].

$$p(s_{t-1}|a_{t-1}, s_t) \longleftarrow {}^{C_{t-1}}_{D_{q(t-1)}}T = {}^{C_t}_{C_{t-1}}T^{-1} {}^{C_t}_{D_{q(t)}}T {}^{D_{q(t-1)}}_{D_{q(t)}}T^{-1} \tag{19}$$

The particle at one-step prior $s_{j,t-1}$ is calculated based on the mean values of motions of the camera and the demonstrator, which is ${}^{C_t}_{C_{t-1}}T$ and ${}^{D_{q(t-1)}}_{D_{q(t)}}T$. After estimating $s_{j,t-1}$, the $i$-th proto-symbol is converted into 2D Cartesian space by using the particle $s_{j,t-1}$. Let ${}^I b_k$ denote the output probability distribution of the $k$-th state in the 2D-converted proto-symbol. Then, the best matching state $q(t-1)$ at time $t-1$ is calculated by

$$q(t-1) = \arg\max_k {}^I b_k(o_{t-1}) a_{kq(t)}, \quad \forall k = 1..N \tag{20}$$

where $N$ is the number of states in the proto-symbol. This induction phase is iterated until obtaining $s_{j,t-w}$.

### 5.3.2. Proto-symbol Conversion

Once the time-sequence of the demonstrator's relative position/orientation $S_j = \{s_{j,t-w}, \cdots, s_{j,t}\}$ is obtained, the proto-symbol can be converted into the camera coordinates and subsequently into image coordinates. Since the demonstrator's basebody position/orientation with respect to the camera is time-varying, we convert the proto-symbol at each time-step. At time $t$, we focus on the state at time $t$, namely $q(t)$. The demonstrator's basebody position/orientation is represented as a probability distribution by using particles. The proto-symbol is converted at each time-step by each particle. At time $t$, the proto-symbol is converted into camera coordinates $\{C_t\}$ by each particle $s_{j,t}$, which is a candidate of ${}^{C_t}_{D_t}T$. The conversion by the homogeneous transformation matrix is carried out as described in section 4.3. Then, the proto-symbol is converted into the image coordinates $\{I_t\}$ at time $t$ by the perspective projection with calibrated camera parameters.

### 5.3.3. Optical flow in Proto-symbol

This section explains the calculation of ${}^I b_{q(t)}({}^I o_t)$, the probability density function to generate an observed motion ${}^I o_t$ from a state $q(t)$ of the proto-symbol at time $t$. Let ${}^I o_t$ be an observed motion, which consists of makers' 2D pixel positions and their optical flow, at time $t$, as shown on the right sub-figure of Fig. 4.

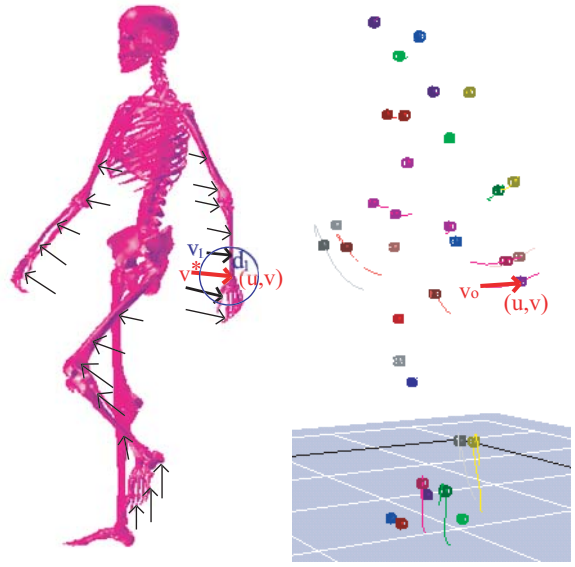


Figure 4: Computation of a corresponding optical flow $v*$ in a proto-symbol (left) to the observed optical flow $v_o$ at pixel position $(u, v)$ (right). If the proto-symbol does not have the optical flow at the position $(u, v)$, the corresponding optical flow $v*$ at $(u, v)$ in the proto-symbol is calculated by interpolation of nearby optical flows.

Let $v_o$ be an optical flow at a pixel position $p_o = (u, v)$ of the observed image. The observed optical flow $v_o$ is compared with an optical flow $v*$ at the same position $p_o = (u, v)$ in the converted proto-symbol ${}^I\lambda_x$. The optical flow $v*$ is calculated by interpolation of nearby optical flow information as shown on the left sub-figure of Fig. 4 and eq. (21).

$$v* = \frac{1}{\sum_l \frac{1}{d_l}} \sum_{l=1}^{L} \frac{v_l}{d_l} \tag{21}$$

Let $v_l$ be a nearby optical flow, $\forall l = 1..L$ at its pixel position $p_l$, whose distance $d_l$ from $p_o = (u, v)$ is smaller than a predefined threshold. Here, $L$ is the number of the optical flows in the local region.

Please remind that converted proto-symbols ${}^I\lambda_x$ include the mean vector and covariance matrix of both feature positions and their optical flows. Feature points correspond to the acting point $p_l$ and their optical flows correspond to $v_l$, $\forall l = 1..L$. In the same way as eq. (21), which is simple linear interpolation, the mean and covariance for the corresponding optical flow are calculated.

With the calculated mean and covariance of the output Gaussian distribution function, output probability ${}^I b_{q(t)}({}^I o_t)$ is calculated in terms of optical flows. The likelihood $P(O|\lambda_{ij})$ is computed by the forward algorithm [80]. Only observed optical flows are compared with the corresponding ones of the proto-symbol, when calculating the output probability ${}^I b_{q(t)}({}^I o_t)$. In this way (using only visible parts), the method can estimate

[9] Please remind that proto-symbols contain information of the baselink velocity as well as joint position and joint velocity.

whole-body motion even under occlusion. Note that the proposed method does not require the solution of the feature labeling problem or continuous tracking of the features.

## 6. Motion Recovery using Proto-symbols in Multiple Coordinates

Motion patterns are decoded using the expectation operator in the stochastic model. The motion generation is a two-stage stochastic process: state transition generation and motion output generation.

Active use of stochastic models enriches the repertoire of the motion patterns [73][7]. One approach is interpolation approach in [73]. This enables to generate motions which corresponds to a mixture of existing proto-symbols. Using mixing coefficients of multiple proto-symbols, a new proto-symbol is calculated in the proto-symbol space and a corresponding motion is reproduced from the new proto-symbol. Another approach is *proto-symbol based motion duplication* [7], which can perform complete motion generation which is close to the partial observation using database.

The motion $y$ for a humanoid robot or a human articulated figure is generated by applying the best proto-symbol $\lambda^*$ and current observations $o$ by eq. (22).

$$y = g(\lambda^*, o) \tag{22}$$

It follows a two-step procedure: to estimate hidden state sequences in HMM for a partial observation sequence, and then to reproduce a full motion sequence from the estimated states. The original algorithm is modified in order to cope with the motion recovery of a human articulated figure from an image sequence.

Note that observation of a motion pattern at each time step is an occluded 2D monocular image, $o = {}^I o_x$. The generated motion $y$ is desired to be represented in joint space, $y = y_\theta$, because of the easy control for the articulated movements.

The state sequence is obtained by applying the Viterbi algorithm [80], which finds the single best state sequence $Q = \{q_t\}, 1 \leq t \leq T$ for the given observation sequence ${}^I O_x$. As a result, the recovered motion pattern can be temporally synchronized with the observation.

$$Q = \max_{q_1,\cdots,q_T} P({}^I o_x | {}^I \lambda_x^*) \tag{23}$$

Since the observation ${}^I o_x$ is represented in 2D image coordinates, the transformed proto-symbol in 2D image space ${}^I \lambda_x^*$ is used for the Viterbi algorithm. This optimal state transition generation enables us to generate a motion pattern similar to the observed target motion pattern. For the invisible motion elements, either eq. (24) or eq. (25) is substituted into the output probability density function,

$$\{x_k\}_t - \mu_{ij} = * \tag{24}$$

$$\Sigma_{ij} = \infty \tag{25}$$

so that the invisible motion elements do not affect the output probability density function. In eq. (24), $*$ indicates a constant value and it is set to zero[10] in the experiments (Sec. 7).

After the optimal state sequence $Q = \{q_t\}$ is obtained, the output observation sequence $y$ is decoded according to $Q$ by the output probability distribution $B_\theta$ of the proto-symbol in joint space. Note that $B_\theta$ is used in order to generate joint angles $y_\theta$. In other words, the proto-symbols in 2D image coordinate is used for the former step (estimating hidden states) and the proto-symbols in joint space is used for the latter step (reproducing complete human motions). Since the state transition probability matrix $A$ and the initial state probability vector $\pi$ are not changed for coordinate transformation of a proto-symbol, the optimal state transition $Q = Q_x = Q_\theta$ is the same for both ${}^I \lambda_x$ and $\lambda_\theta$.

## 7. Experiments

### 7.1. Experimental Setting

As shown in Fig. 1, the proposed architecture consists of two main steps: learning and recognition & recovery. During the learning step, in order to acquire the 3D proto-symbols for training data, accurate human motion patterns including basebody position/orientation are captured by an optical motion capture system, which is composed of ten cameras. The 3D positions of the optical markers, which are attached to the subject, are obtained by the capturing system (left in Fig. 5). Then, the attachment points of the markers' labels are specified by a labeling procedure[11] (middle in Fig. 5). Once the labeled markers' 3D positions are acquired, the subject's motion can be mapped into a humanoid robot model or a human skeleton model by inverse kinematics (right in Fig. 5). In the experiment, we use a skeleton model. The skeleton model has 56 DOF in total: 50 DOF for joint angles $\theta$ and 6 DOF for base body position/orientation. Table 1 shows the details of the joint configurations. The position/orientation of the base body can be represented as the homogenous transformation matrix ${}^G_D T$. From the time-sequence of $\theta$ and ${}^G_D T$, joint angular velocities $\dot{\theta}(t)$ and basebody position/orientation velocity ${}^{D_{t-1}}_{D_t} T$ are calculated.

With the inverse kinematics results, proto-symbols are trained with joint angles ${}^{D_t}\theta$ (50 DOF), joint angular velocities ${}^{D_t}\dot{\theta}$ (50 DOF) and basebody position/orientation velocity ${}^{D_{t-1}}_{D_t} T$ (6 DOF). Six motions are trained as proto-symbols a-priori: (1) STEP, (2) CHEER, (3) KICK, (4) SQUAT, (5) BOW, and (6) RUN in a circle. The sampling time of motion data is 30ms and the length of each motion pattern is 1000 frames (30 seconds) except for the KICK. The length of the KICK motion pattern is 802 frames (24 seconds). Since people do not perform the same motion twice in the exactly same fashion, it is useful to train proto-symbols from multiple exemplars. Each motion pattern

[10] The value $*$ is not necessary to be zero, but any constant number.

[11] Labeling is needed only during the learning step. Neither the labels of markers nor marker tracking is needed for motion recovery and recognition.

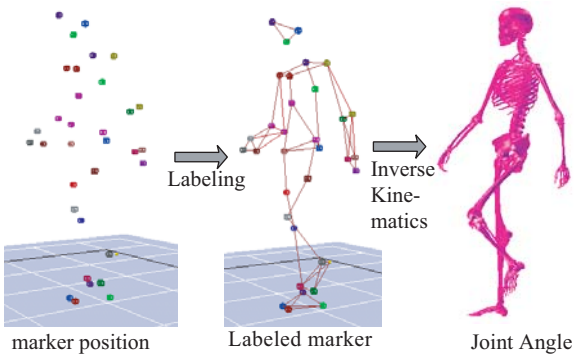


Figure 5: Computation flow to estimate motion from 3D marker position for training data

Table 1: Degrees of Freedom : 50 DOF for joint angles

| Joint name | DOF | Joint name | DOF |
|---|---|---|---|
| Lumbar Vertebra | 3 | Left Hand | 3 |
| Rib Vertebra | 3 | Neck Vertebra7 | 3 |
| Right Shoulder Clavicle | 3 | Head | 3 |
| Right Upper Arm | 3 | Right Leg Thigh | 3 |
| Right Fore Arm Elbow | 3 | Right Leg Shank | 1 |
| Right Hand | 3 | Right Foot | 3 |
| Left Shoulder Clavicle | 3 | Left Leg Thigh | 3 |
| Left Upper Arm | 3 | Left Leg Shank | 1 |
| Left Fore Arm Elbow | 3 | Left Foot | 3 |

contains several periodic movements. The STEP motion pattern contains 26 steps. The CHEER motion pattern contains 15 cheers. The KICK contains 14 kicks. The SQUAT contains 19 squats. The BOW motion pattern contains 12 bows. The RUN motion pattern contains 35 steps.

In order to validate the proposed recognition and recovery algorithm, six motions are demonstrated as test data: STEP, CHEER, KICK, SQUAT, BOW, and RUN in a circle. Some of the demonstration motions of the real human subject are shown in Fig. 6. Time series of motion data are observed through a fixed width window and the width is set to 70 frames. Each motion length is 300 frames (9 sec). Some of the motions are partially occluded.

The observed data from a monocular camera consist of pixel positions of arbitrary feature points on the subject, as shown in Fig. 7. In the experiments, a motion model of the camera is not considered. The camera is not located on a real moving robot, but at a static position, since optical flow calculation from blurred image of a moving camera is beyond our interests in this paper. The human's baselink positions/orientations for the initial 70 frames are given roughly by adding Gaussian noises. 1000 particles are used for the particle filter estimating the demonstrator position/orientation (6DOF). In order to evaluate the accuracy of 3D motion recovery from 2D images, the true motion is computed by inverse kinematics from 3D position of labeled markers, which are captured by ten cameras.

In order to separate image processing part from our main focus on the inference mechanism, the monocular observation data are calculated in an artificial way in the experiments. Thirty-four reflective markers are attached on the human body. From the labeling, the linkage structure between the markers is known. Virtual markers are added between two linked markers in order to gather more features than the 34 attached markers. The artificially obtained markers are displayed in Fig. 7. After generating synthetic observation data we discard labeling information. For motion recognition and morion recovery, features do not need to be labeled nor tracked. Also, they can be located at arbitrary positions. Thus, these artificially prepared 2D features can be substituted with processed optical flows from real images.

### *7.2. Evaluation of Motion Recovery*

Here we show five recovery examples: (1) CHEER motion without occlusion, (2) KICK motion with small occlusion on a swinging right leg from time to time (22% occlusion), (3) RUN around making a CIRCLE motion with occlusion of two lower legs (36% occlusion), (4) STEP motion with occlusion of the left half of the body (36% occlusion), and (5) BOW motion with occlusion of the left half (51% occlusion). The observed data are pixel positions of features on the subject. The observations of the five motions at selected frames are shown in Fig. 7. From the 2D monocular images, the proposed method recognizes the subject's motion by finding the proto-symbol with the highest likelihood. A set of 1000 particles is used in the particle filter for estimation of the subject's position/orientation.

The recovered motion at selected frames is shown in Fig. 8. The 3D motion recovery (blue, dark color) from 2D unlabeled marker data is compared with the inverse kinematics results (magenta, light color) from 3D labeled marker data, which are assumed to be ground true. Figure 8 shows that the recovered 3D motions fit well to the true motions.

The estimated baselink position and orientation are shown in Fig. 9. The light colored bold curves represent the true values. The dark colored thin curves indicate the estimated poses from 2D partial observation of optical flows. The figure shows that the particle filter estimates the human's baselink position and orientation generally well.

In addition, an analysis of how close the recovered motion to the motion database is carried out. Please remind that the motion database is not represented by static motion patterns, but by stochastic forms (HMMs). In order to compare them, first, each recovered motion is trained to a new HMM. Then, the new HMMs are compared to HMMs in the database by Kullback Leibler distance [81]. The Kullback Leibler distance $D(\lambda_i, \lambda_j)$ is a dissimilarity measure between two HMMs ($\lambda_i$ and $\lambda_j$). It is calculated by

$$D^*(\lambda_i, \lambda_j) = \ln P(O_i|\lambda_i) - \ln P(O_i|\lambda_j) \tag{26}$$

$$D(\lambda_i, \lambda_j) = \frac{D^*(\lambda_i, \lambda_j) + D^*(\lambda_j, \lambda_i)}{2} \tag{27}$$

where $O_i$ denotes the motion patterns generated by the HMM $\lambda_i$. The calculated Kullback Leibler distance is shown in Table 2. Each motion among five recovered results is compared to the six HMMs in the database. From the table, each motion has the smallest dissimilarity to the corresponding HMM in the database.


Figure 6: Subject's demonstration. (a) KICK: at frame 80, 90, 100, $\cdots$, 170. (b) RUN: at frame 80, 90, 100,$\cdots$, 170. (c) BOW: at frame 70, 80, 90, $\cdots$, 160.

Table 2: Comparison recovered motion to HMMs in database by Kullback Leibler Distance

| Motion | STEP HMM | CHEER HMM | KICK HMM | SQUAT HMM | BOW HMM | RUN HMM |
|---|---|---|---|---|---|---|
| STEP | $\mathbf{2.9\times10^2}$ | $4.0\times10^3$ | $2.5\times10^3$ | $2.8\times10^3$ | $3.8\times10^3$ | $3.7\times10^3$ |
| CHEER | $6.5\times10^3$ | $\mathbf{1.7\times10}$ | $2.3\times10^3$ | $1.7\times10^3$ | $3.1\times10^3$ | $4.1\times10^3$ |
| KICK | $1.1\times10^4$ | $1.2\times10^4$ | $\mathbf{1.4\times10^3}$ | $8.7\times10^3$ | $1.2\times10^4$ | $7.5\times10^3$ |
| BOW | $1.2\times10^4$ | $6.4\times10^3$ | $7.0\times10^3$ | $4.0\times10^3$ | $\mathbf{2.9\times10^2}$ | $1.1\times10^4$ |
| RUN | $6.8\times10^3$ | $6.4\times10^3$ | $3.1\times10^3$ | $6.8\times10^3$ | $1.1\times10^4$ | $\mathbf{8.5\times10}$ |

Table 3: Mean errors of basebody position/orientation. Maximum errors are shown in parentheses. (unit: meter and radian)

| Motion | x | y | height | roll | pitch | yaw |
|---|---|---|---|---|---|---|
| CHEER | 0.002 (0.011) | 0.004 (0.017) | 0.002 (0.008) | 0.014 (0.098) | 0.003 (0.020) | 0.014 (0.085) |
| KICK | 0.010 (0.060) | 0.018 (0.147) | 0.013 (0.067) | 0.043 (0.286) | 0.051 (0.302) | 0.092 (0.627) |
| RUN | 0.010 (0.054) | 0.015 (0.069) | 0.008 (0.033) | 0.0194 (0.080) | 0.019 (0.080) | 0.001 (0.029) |
| STEP | 0.003 (0.012) | 0.007 (0.036) | 0.006 (0.029) | 0.029 (0.175) | 0.012 (0.053) | 0.021 (0.081) |
| BOW | 0.003 (0.019) | 0.012 (0.062) | 0.008 (0.046) | 0.019 (0.139) | 0.005 (0.028) | 0.014 (0.065) |

Finally a statistical analysis for estimation of human's baselink position and orientation is given. From five runs for each motion type with different initialization parameters, the mean and maximum errors of the estimated position and orientation are shown in table 3. From the table, it can be shown that the position and orientation are well estimated for various motions including a dynamic occluded motion. The highly dynamic KICK motion has the maximum position error of 0.067 [m] and maximum angular errors of 0.627 [rad]. The BOW motion with 51% occlusion has 0.062 [m] and 0.139 [rad] errors in the worst case.

### *7.3. Evaluation of Motion Recognition*

Recognition rate is surveyed from multiple runs with different initial conditions. Sixty runs are carried out as follows.

- for 6 motion types
- 5 exemplars for each motion type (the range of occlusion levels is between 0 and 50%.)
- two cases of the number of particles in the particle filter: 100 particles and 1000 particles

The overall rate for successful motion recognition is 100 [%]. Since currently a small number (6) of motion types are considered, it is rather easy to classify the motion type out of the six. However, as the number of motion types increases, successful recognition ratio may decrease. In our previous work [42], we evaluated recognition robustness with respect to partial observations. The average recognition success was 99.85% under 50% occlusion for 8 motion types. More detailed analysis for recognition with respect to different occlusion levels, please refer to [42].

### *7.4. Evaluation of Computational Cost*

From 30 runs (6 motion types and 5 exemplars for each motion type) with different initial conditions, the overall rate for computation time is calculated in the case of 1000 particles and in the case of 100 particles. Since the recognition and recovery is tightly coupled, the integrated computation time for both procedures is measured. With 1000 particles, the total computation time takes 2769 [s] on an average. With 100 particles, the averaged total computation time for motion recognition and recovery for a 9 [sec] motion is 538 [s] without the computational optimization of code and parallel processing. The specification of the computer used for the experiments is Intel(R) Core(TM)2 CPU 6700 @2.66GHz, 3.00 GB RAM. Real-time computation


Figure 7: Unlabeled 2D markers in image coordinates. (1) CHEER without occlusion: at frame 80, 90, 100, $\cdots$, 170. (2) KICK with small occlusion on a swinging right leg from time to time: at frame 80, 90, 100, $\cdots$, 170. (3) RUN with occlusion of two legs: at frame 80, 90, 100,$\cdots$, 170. (4) STEP with occlusion of the left half: at frame 70, 80, 90, $\cdots$, 160. (5) BOW with occlusion of the left half: at frame 70, 80, 90, $\cdots$, 160.

for the application to the human robot interaction is beyond the scope of this study.

For reduction of computational cost, parallel computation using a PC cluster can be considered. The particle filter is suitable for parallel computation since the computation for each particle is done independently. The GPGPU computation as reported in our previous work [82] reduced the computational time by an order of ten. Further reduction and realtime computation will be within the scope as we see the recent advance of GPU boards.

### *7.5. Discussions*

In the current implementation, we assume that a proto-symbol corresponding to the observed data exists. This does not mean the exactly same motion pattern is observed during online recognition as training data, because the same subject may perform the same gesture in a slightly different manner. Handling a completely unknown motion outside of the database (learned proto-symbols ) is a difficult problem. A new motion types can be handled by linear interpolation between the known proto-symbols in the symbol space [73]. Another approach may be the unsupervised incremental learning scheme [83] where a completely unknown observation can be learned incrementally in real-time. In the latter case, an important issue is how to classify unknown movements. One way is to use a ratio of maximum likelihood and the second highest likelihood, called *recognition ratio*. Using this recognition ratio, the algorithm can judge "whether to learn a new proto-symbol" and act as a trigger for online incremental proto-symbol acquisition, so that proto-symbol acquisition can be performed automatically. This approach for handling unknown motions by incremental learning using the recognition ratio has been proposed in our previous work [42].

Note that markers do not need to be placed at exact locations for motion recognition in our proposed method. Same markers were used for motion training only for comparison purpose with the inverse kinematics solution. There is no need of to fix the markers in specific locations in principle. As shown in Sec. 5.3.3, feature points at any locations can be compared with the corresponding ones of the proto-symbol.

Motion recognition and recovery have been tested with different motions and occlusion levels. As we expected, the results (Fig. 8 and Table 3) highly dynamic motions and large portion of body occlusion make the recovery harder. In our previous work [42], we analyzed the effect of different occlusion levels where used motion types (e.g. walk, kick, raise arms, punch, bow, squat, etc) were similar to those in this work. The statistical analysis (using 672 observations) showed the 99.85% recognition success rate on average, when half of body was occluded. When more than 75% information is missing, the recognition becomes unstable and recovery error increases rapidly. For the detailed experimental results and analysis, we referred the readers to [42].

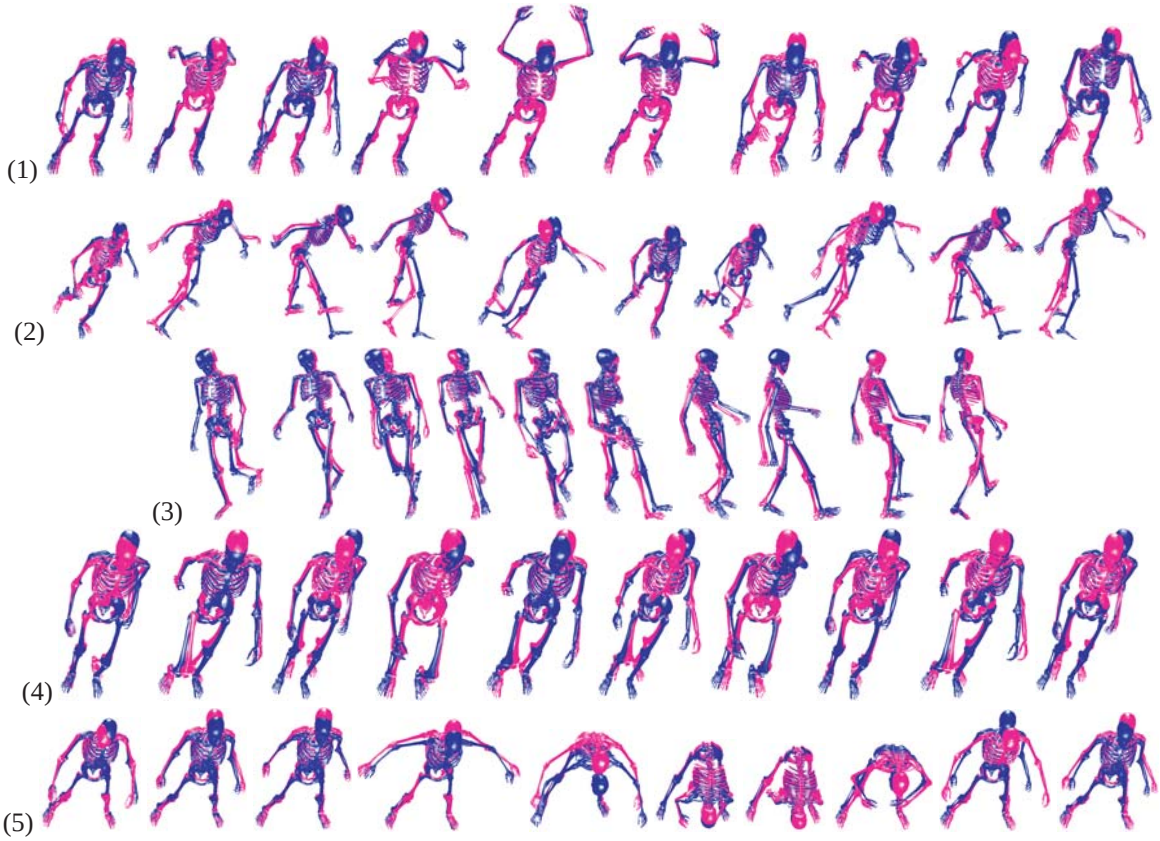


Figure 8: Motion recovery in 3D space. The recovered motion (blue, dark color) from 2D occluded and unlabeled marker data is compared with the inverse kinematics results (magenta, light color) from 3D labeled markers data. (1) CHEER: at frame 80, 90, 100, $\cdots$, 170. (2) KICK: at frame 80, 90, 100,$\cdots$, 170. (3) RUN: at frame 80, 90, 100,$\cdots$, 170. (4) STEP: at frame 70, 80, 90, $\cdots$, 160. (5) BOW: at frame 70, 80, 90, $\cdots$, 160.

## 8. Conclusion

An approach for 3D human motion capturing and recognition from 2D partial observations of optical flow based on statistical inference using a database of 3D motion is proposed. The motion database is composed of proto-symbols. Each proto-symbol is defined through the HMM parameters. The proposed inference mechanism solves both 3D motion recovery and motion recognition problems simultaneously. Instead of extracting rich information by expensive computation of image processing, this paper focuses on an inference mechanism from low level image features (e.g., optical flow), inspired by human's high perception ability shown in the moving light display experiments. Optical flows of unlabeled features are used for characterizing the motion. The human basebody position/orientation with respect to camera coordinates is estimated by the particle filter. A 3D whole body motion is recovered by using the motion database. The proposed method is validated on a human motion dataset with a 56DOF human articulated model. Experimental results show successful motion recognition and 3D recovery from occluded 2D optical flows of unlabeled features. Motion recognition is carried out 100 [%] successfully among six motion types. A demonstrator's basebody position and orientation is estimated with an acceptable range of errors. The maximum position and angular error is 0.067[m] and 0.627[rad] in the case of the highly dynamic KICK motion. The recovered 50 DOF joint angles are reasonably well matched to the true values.

The main contributions are summarized as follows.

1. Coordinate transformations of the HMM parameters, which relate exteroception (monocular observation) to proprioception (motor control), are proposed. Based on these transformations the compact motion database in joint space can be used for comparison with 2D images from any view point without the need of depth information.
2. 3D whole body motion can be recovered from an *occluded monocular* image sequence, which includes not only self occlusion but also occlusion by obstacles.
3. In contrast to the conventional particle filter, the extended particle filter can estimate explicitly a time series of the human basebody position/orientation with respect to camera coordinates. This enables view invariant method for motion recognition and recovery.

While the current paper focuses on the inference mechanism for 3D human motion capturing and recognition from 2D partial observations of optical flow, in the future work the proposed method will be extended to realize a complete system by addressing the following issues: (1) extraction of feature points [84] and optical flows [85][24][25] from texture image sequences of onboard moving cameras on a humanoid robot, and (2) real-time computation with parallel computation.

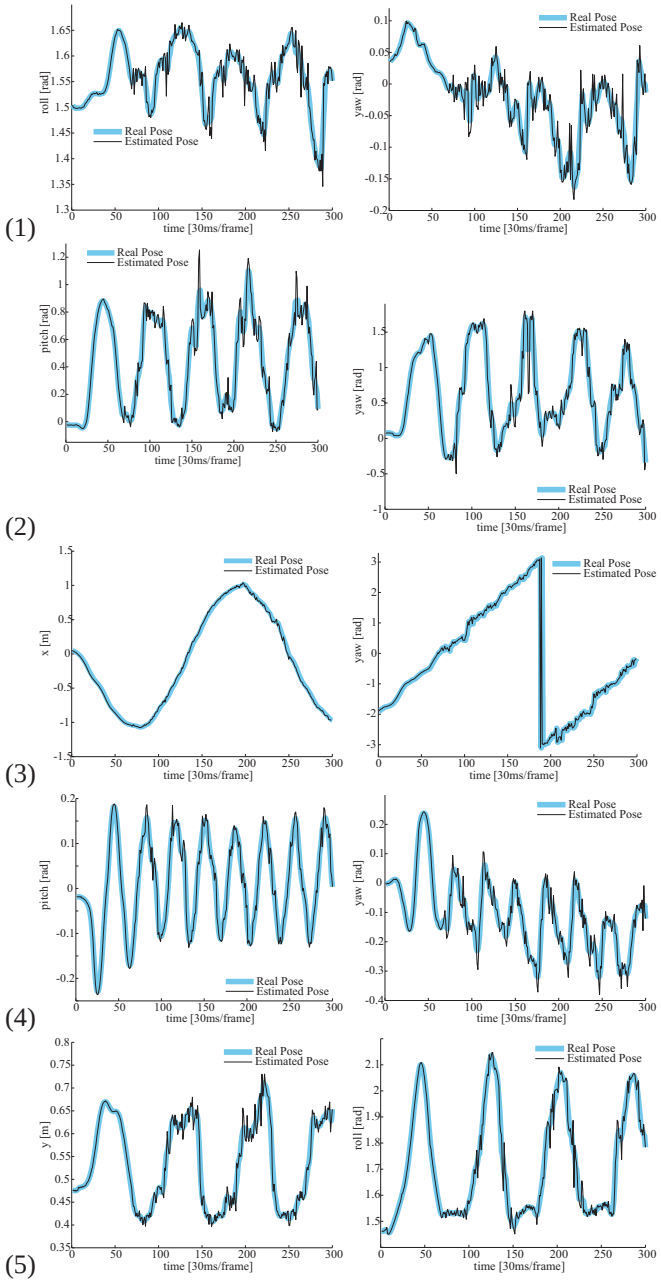


Figure 9: Real Position/Orientation and Estimated Position/Orientation. The horizontal axis is time (frame) and the vertical axis is base-link position and orientation (unit: meter and radian). Light colored bold curves indicate real positions and orientations. Dark colored thin curves indicate estimated positions and orientations. For each motion, the two axes with the largest changes are displayed. (1) CHEER: roll and yaw, (2) KICK: pitch and yaw, (3) RUN: x and yaw, (4) STEP: pitch and yaw, (5) BOW: y and roll.

Further, application for human-humanoid interaction using onboard cameras will be handled, for example motion retargeting to a humanoid robot by adopting the method in [86]. Although there is the kinematic difference between a real human and a humanoid robot (usually 20~40 DOF), some recent methods [86] [87] showed reasonable retargeting performance from humans to humanoid robots.

## Acknowledgment

This research is partially supported by Special Coordination Funds for Promoting Science and Technology, "IRT Foundation to Support Man and Aging Society" and Technical University Munich, Institute for Advanced Study, funded by the German Excellence Initiative.

[1] G. Johansson, Visual motion perception, Scientific American (1975) 76–88.
[2] S. Sumi, Upside-down presentation of the johansson moving light-spot pattern, Perception 13 (3) (1984) 283 – 286.
[3] J. Cutting, L. Kozlowski, Recognizing friends by their walk: Gait perception without familiarity cues, Bulletin Psychonometric Society 9 (1977) 353–356.
[4] W. Dittrich, T. Troscianko, S. Lea, D. Morgan, Perception of emotion from dynamic point-light displays represented in dance, Perception 25 (1996) 727–738.
[5] G. Mather, L. Murdoch, Gender discrimination in biological motion displays based on dynamic cues, Proc. R. Soc. Lond. B 259 (1994) 273–279.
[6] T. Inamura, Y. Nakamura, I. Toshima, Embodied symbol emergence based on mimesis theory, Int. Journal of Robotics Research 23 (4) (2004) 363–377.
[7] D. Lee, Y. Nakamura, Mimesis from partial observations, in: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2005, pp. 1911–1916.
[8] A. Agarwal, B. Triggs, Recovering 3d human pose from monocular images, IEEE Trans. on Pattern Analysis and Machine Intelligence 28 (1) (2006) 44–58.
[9] H. Sidenbladh, M. J. Black, L. Sigal, Implicit probabilistic models of human motion for synthesis and tracking, in: European Conf. on Computer Vision, 2002, pp. 784–800.
[10] C. Sminchisescu, B. Triggs, Estimating articulated human motion with covariance scaled sampling, Int. Journal of Robotics Research 22 (6) (2003) 371–392.
[11] J. Davis, A. Bobick, The representation and recognition of action using temporal templates, in: Proceedings Computer Vision and Pattern Recognition, 1997, pp. 928–934.
[12] J. Yamato, J. Ohya, K. Ishii, Recognizing human action in time-sequential images using hidden markov model, in: IEEE Conf. on Computer Vision and Pattern Recognition, 1992, pp. 379–385.
[13] D. Weinland, E. Boyer, R. Ronfard, Action recognition from arbitrary views using 3d exemplars, in: IEEE Int. Conf. on Computer Vision, 2007, pp. 1–7.
[14] G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, M. Csorba, A solution to the simultaneous localization and map building (slam) problem, IEEE Transactions on Robotics and Automation 17 (2001) 229–241.
[15] D. Lee, W. Chung, Discrete status based localization for indoor service robots, IEEE Transactions on Industrial Electronics 53 (5) (2006) 1737–1746.
[16] R. Poppe, Vision-based human motion analysis: An overview, Computer Vision and Image Understanding 108 (1-2) (2007) 4–18.
[17] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, IEEE Trans. on Pattern Analysis and Machine Intelligence 24 (4) (2002) 509 – 522.
[18] D. Ramanan, D. A. Forsyth, A. Zisserman, Strike a pose: Tracking people by finding stylized poses, in: IEEE Conf. on Computer Vision and Pattern Recognition, 2005, pp. 271–278.
[19] J. Deutscher, I. D. Reid, Articulated body motion capture by stochastic search, International Journal of Computer Vision 61 (2) (2005) 185–205.
[20] W.-L. Lu, K. Okuma, J. J. Little, Tracking and recognizing actions of multiple hockey players using the boosted particle filter, Image and Vision Computing 27 (2009) 189–205.
[21] C. Bregler, J. Malik, Tracking people with twists and exponential maps, in: IEEE Conf. on Computer Vision and Pattern Recognition, 1998, pp. 8–15.
[22] A. Fathi, G. Mori, Human pose estimation using motion exemplars, in: IEEE Int. Conf. on Computer Vision, 2007, pp. 1–8.
[23] D. Fleet, Y. Weiss, Optical flow estimation, 2005, chapter in book edited by N. Paragios, Y. Chan and O. Faugeras and titled Mathematical Models in Computer Vision: the handbook.

[24] H. Spies, H. Scharr, Accurate optical flow in noisy image sequences, in: IEEE Int. Conf. on Computer Vision, 2001, pp. 587–592.
[25] Y. Song, L. Goncalves, E. D. Bernardo, P. Perona, Monocular perception of biological motion in Johansson displays, Computer Vision and Image Understanding 81 (3) (2001) 303–327.
[26] M. Donald, Origins of the Modern Mind, Harvard University Press, Cambridge, 1991.
[27] G. Rizzolatti, L. Fadiga, V. Gallese, L. Fogassi, Premotor cortex and the recognition of motor actions, Cognitive Brain Research 3 (1996) 131–141.
[28] D. Lee, Y. Nakamura, Motion capturing from monocular vision by statistical inference based on motion database: Vector field approach, in: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2007, pp. 617–623.
[29] V. Gallese, L. Fadiga, L. Fogassi, G. Rizzolatti, Action recognition in the premotor cortex, Brain 119 (1996) 593–609.
[30] G. Hayes, J. Demiris, A robot controller using learning by imitation, in: Intl. Symp. on Intelligent Robotic Systems, 1994, pp. 198–204.
[31] Y. Kuniyoshi, M. Inaba, H. Inoue, Learning by watching: Extracting reusable task knowledge from visual observation of human performance, IEEE Transaction on Robotics and Automation 10 (6) (1994) 799–822.
[32] D. C. Bentivegna, C. G. Atkeson, Using primitives in learning from observation, in: IEEE-RAS Int. Conf. on Humanoid Robots, 2000.
[33] D. C. Bentivegna, C. G. Atkeson, G. Cheng, Learning similar tasks from observation and practice, in: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2006, pp. 4994–5000.
[34] A. Billard, M. J. Matarić, Learning human arm movements by imitation: Evaluation of biologically inspired connectionist architecture, Robotics and Autonomous Systems 37 (2001) 145–160.
[35] W. Takano, K. Yamane, T. Sugihara, K. Yamamoto, Y. Nakamura, Primitive communication based on motion recognition and generation with hierarchical mimesis model, in: IEEE Int. Conf. on Robotics and Automation, 2006, pp. 3602–3609.
[36] D. Lee, C. Ott, Y. Nakamura, Mimetic communication model with compliant physical contact in human-humanoid interaction, Int. Journal of Robotics Research 29 (13) (2010) 16841704.
[37] S. Nakaoka, A. Nakazawa, F. Kanahiro, K. Kaneko, M. Morisawa, K. Ikeuchi, Task model of lower body motion for a biped humanoid robot to imitate human dances, in: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2005, pp. 2769–2774.
[38] T. Inamura, N. Kojo, T. Sonoda, K. Sakamoto, K. Okada, M. Inaba, Intent imitation using wearable motion capturing system with on-line teaching of task attention, in: IEEE-RAS Int. Conf. on Humanoid Robots, 2005, pp. 469–474.
[39] E. Demircan, L. Sentis, V. D. Sapio, O. Khatib, Human motion reconstruction by direct control of marker trajectories, in: Advances in Robot Kinematics, 2008, pp. 263–272.
[40] K. Kurihara, S. Hoshino, K. Yamane, Y. Nakamura, Optical motion capture system with pan-tilt camera tracking and realtime data processing, in: IEEE Int. Conf. on Robotics and Automation, Vol. 2, 2002, pp. 1241–1248.
[41] Z. Ghahramani, M. I. Jordan, Supervised learning from incomplete data via an EM approach, in: Advances in Neural Information Processing Systems, Vol. 6, 1994, pp. 120–127.
[42] D. Lee, Y. Nakamura, Mimesis model from partial observations for a humanoid robot, Int. Journal of Robotics Research 29 (1) (2010) 60–80.
[43] J. K. Aggarwal, Q. Cai, Human motion analysis: A review, Computer Vision and Image Understanding 73 (3) (1999) 428–440.
[44] L. Ren, G. Shakhnarovich, J. Hodgins, P. Viola, H. Pfister, Learning silhouette features for control of human motion, in: Technical Report CMU-CS-04-165, Carnegie Mellon University, 2004.
[45] T. B. Moeslund, A. Hilton, V. Kruger, A survey of advances in vision-based human motion capture and analysis, Computer Vision and Image Understanding 104 (2-3) (2006) 90–126.
[46] C. Cedras, M. Shah, Motion-based recognition: A survey, IVC 13 (2) (1995) 129–155.
[47] R. Veltkamp, M. Hagedoorn, State-of-the-art in shape matching, Tech. Rep. UU-CS-1999-27, Utrecht University, the Netherlands (1999).
[48] D. A. Forsyth, O. Arikan, L. Ikemoto, J. O'Brien, D. Ramanan, Computational studies of human motion: Part 1, tracking and motion synthesis, Foundations and Trends in Computer Graphics and Vision 1(2) (2005) 77–254.
[49] C. Bregler, J. Malik, K. Pullen, Twist based acquisition and tracking of animal and human kinematics, International Journal of Computer Vision 56 (3) (2004) 179–194.
[50] S. Wachter, H.-H. Nagel, Tracking persons in monocular image sequences, Computer Vision and Image Understanding 74 (3) (1999) 174 – 192.
[51] C. J. Taylor, Reconstruction of articulated objects from point correspondences in a single uncalibrated image, Computer Vision and Image Understanding 80 (2000) 677–684.
[52] C. Barrón, I. A. Kakadiaris, Estimating anthropometry and pose from a single uncalibrated image, Computer Vision and Image Understanding 81 (3) (2001) 269–284.
[53] N. R. Howe, M. Leventon, W. Freeman, Bayesian reconstruction of 3d human motion from single-camera video, Advances in Neural Information Processing Systems 12 (1999) 820–826.
[54] M. Brand, Shadow puppetry, in: IEEE Int. Conf. on Computer Vision, 1999, pp. 1237–1244.
[55] G. Mori, J. Malik, Estimating human body configurations using shape context matching, in: European Conf. on Computer Vision, 2002, pp. 666–680.
[56] A. Agarwal, B. Triggs, 3d human pose from silhouettes by relevance vector regression, in: IEEE Conf. on Computer Vision and Pattern Recognition, 2004, pp. 882–888.
[57] R. Rosales, S. Sclaroff, Inferring body pose without tracking body parts, in: Computer Vision and Pattern Recognition, 2000, pp. 721–727.
[58] O. Chomat, J. Martin, J. L. Crowley, A probabilistic sensor for the perception and recognition of activities, in: European Conf. on Computer Vision, 2000, pp. 487–503.
[59] H. Yang, A. Park, S. Lee, Gesture spotting and recognition for human-robot interaction, IEEE Trans. on Robotics 23 (2) (2007) 256–270.
[60] A. Just, S. Marcel, O. Bernier, Hmm and iohmm for the recognition of mono and bi-manual 3d hand gestures, in: British Machine Vision Conf., 2004.
[61] K. Nickel, R. Stiefelhagen, Visual recognition of pointing gestures for human-robot interaction, Image and Vision Computing 3 (12) (2006) 1875–1884.
[62] H. Yang, A. Park, S. Lee, Reconstruction of 3d human body pose for gait recognition, Biometrics. New York: Springer-Verlag 3832 (2006) 619–625.
[63] W.-L. Lu, J. J. Little, Simultaneous tracking and action recognition using the pca-hog descriptor, in: Canadian Conference on Computer and Robot Vision, 2006, p. 6.
[64] E. Shechtman, M. Irani, Space-time behavior based correlation or how to tell if two underlying motion fields are similar without computing them?, IEEE Trans. on Pattern Analysis and Machine Intelligence 29 (11) (2007) 2045–2056.
[65] J. Kohlmorgen, S. Lemm, A dynamic hmm for on-line segmentation of sequential data, NIPS 2001: Advances in Neural Information Processing Systems 14 (2002) 793–800.
[66] A. Fod, M. Matarić, O. Jenkins, Automated derivation of primitives for movement classification, Autonomous Robots 12 (1) (2002) 39–54.
[67] W. Takano, Y. Nakamura, Humanoid robot's autonomous acquisition of proto-symbols through motion segmentation, in: IEEE-RAS International Conference on Humanoid Robotics, 2006, pp. 425–431.
[68] B. Janus, Y. Nakamura, Unsupervised probabilistic segmentation of motion data for mimesis modeling, in: the 12th IEEE Int. Conf. on Advanced Robotics, 2005, pp. 411–417.
[69] D. Kulić, Y. Nakamura, Scaffolding on-line segmentation of fully body human motion patterns, in: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2008, pp. 2860–2866.
[70] J. A. Blimes, A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models, Tech. Rep. ICSI-TR-97-021, University of Berkeley (1997).
[71] S. Calinon, F. D'halluin, E. Sauser, D. Caldwell, A. G. Billard, Learning and reproduction of gestures by imitation: An approach based on hidden markov model and gaussian mixture regression, IEEE Robotics and Automation Magazine 17 (2) (2010) 44–54.
[72] D. Lee, C. Ott, Incremental kinesthetic teaching of motion primitives using the motion refinement tube, Autonomous Robots 31 (2) (2011) 115131.

[73] T. Inamura, H. Tanie, Y. Nakamura, From stochastic motion generation and recognition to geometric symbol development and manipulation, in: IEEE-RSJ Int. Conf. on Humanoid Robots, 2003, pp. 1b–02.
[74] D. Lee, Y. Nakamura, Stochastic model of imitating a new observed motion based on the acquired motion primitives, in: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2006, pp. 4994–5000.
[75] K. Tokuda, H. Zen, T. Kitamura, Trajectory modeling based on hmms with the explicit relationship between static and dynamic features, in: European Conference on Speech Communication and Technology, 2003, pp. 1–4.
[76] W. Takano, H. Tanie, Y. Nakamura, Key feature extraction for probabilistic categorization of human motion patterns, in: the 12th IEEE Int. Conf. on Advanced Robotics, 2005, pp. 424–430.
[77] A. P. Shon, J. J. Storz, R. P. Rao, Towards a real-time bayesian imitation system for a humanoid robot, in: IEEE Int. Conf. on Robotics and Automation, 2007, pp. 2847–2852.
[78] D. Lee, Y. Nakamura, Stochastic theory for motion capturing from onboard monocular vision of humanoid robots, in: 12th Robotics Symposia, 2007, pp. 424–429.
[79] H. Kadone, Y. Nakamura, Symbolic memory for humanoid robots using hierarchical bifrurcations of attractors in nonmonotonic neural networks, in: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2005, pp. 2900–2905.
[80] L. R. Rabiner, A tutorial on hidden markov models and selected applications in speech recognition, Proc. IEEE 77(2) (1989) 257–286.
[81] B. H. Juang, L. R. Rabiner, A probabilistic distance measure for hidden markov modeling, AT&T Tech. J., vol. 64, no. 2 64(2) (1985) 391–408.
[82] L. Zhang, J. Sturm, D. Cremers, D. Lee, Real-time human motion tracking using multiple depth cameras, in: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2012, pp. 2389–2395.
[83] D. Kulić, W. Takano, Y. Nakamura, Incremental learning, clustering and hierarchy formation of whole body motion patterns using adaptive hidden markov chains, Int. Journal of Robotics Research 27 (7) (2008) 761–784.
[84] J. Shi, C. Tomasi, Good features to track, in: IEEE Conf. on Computer Vision and Pattern Recognition, 1994.
[85] B. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision, in: Proceedings of Imaging Understanding Workshop, 1981, pp. 121–130.
[86] C. Ott, D. Lee, Y. Nakamura, Motion capture based human motion recognition and imitation by direct marker control, in: IEEE-RAS Int. Conf. on Humanoid Robots, 2008, pp. 399–405.
[87] B. Dariush, M. Gienger, B. Jian, C. Goerick, K. Fujimura, Whole body humanoid control from human motion descriptors, in: IEEE Int. Conf. on Robotics and Automation, 2008, pp. 2677–2684.