# Unsupervised Body Scheme Learning through Self-Perception

Jürgen Sturm        Christian Plagemann        Wolfram Burgard

*Abstract*— In this paper, we present an approach allowing a robot to learn a generative model of its own physical body from scratch using self-perception with a single monocular camera. Our approach yields a compact Bayesian network for the robot's kinematic structure including the forward and inverse models relating action signals and body pose. We propose to simultaneously learn local action models for all pairs of perceivable body parts from data generated through random "motor babbling." From this repertoire of local models, we construct a Bayesian network for the full system using the pose prediction accuracy on a separate cross validation data set as the criterion for model selection. The resulting model can be used to predict the body pose when no perception is available and allows for gradient-based posture control. In experiments with real and simulated manipulator arms, we show that our system is able to quickly learn compact and accurate models and to robustly deal with noisy observations.

## I. INTRODUCTION

Kinematic models are widely used in robotics, in particular for prediction and control of robotic manipulators [1], [2]. Such models are typically derived analytically by an engineer [3] and usually rely heavily on prior knowledge about the robot's geometry and kinematic parameters. As robotic systems become more complex and versatile, however, or are delivered in a completely reconfigurable way, there is a growing demand for techniques allowing a robot to automatically learn its current body scheme with no or minimal human intervention. Such a capability would not only facilitate the deployment and calibration of new robotic systems but also allow for autonomous re-adaptation when the body scheme changes, e.g., through regular wear-and-tear over time or even intended reconfiguration as for example in the case of temporary tool use.

Neuro-physiological evidence indicates that humans as well as higher primates learn and adapt their internal models continuously and autonomously using self-perception [4]. Brain scan studies on monkeys that have been trained to use tools revealed that the tool itself even gets integrated into their body schemes over time [5]. Mirror neurons as found in brain area F5 map proprioceptive sensations to tactile and visual ones and thereby seem to serve as a neurological representation of the body scheme [6]. Moreover, they seem to translate external visual stimuli, for example from a demonstrator, into proprioceptive ones, and thereby play an important role in imitation and imitation learning.

In this paper, we investigate ways of realizing such capabilities on artificial systems, in particular on robotic
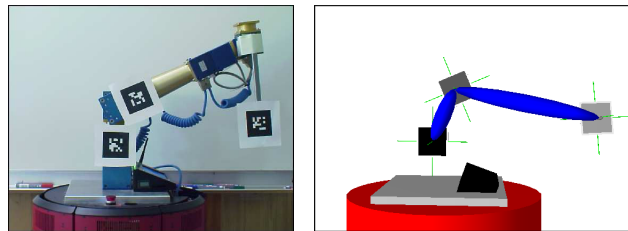
The authors are with the University of Freiburg, Department of Computer Science, Georges-Koehler-Allee 79, 79110 Freiburg, Germany
E-mail: {sturm, plagem, burgard} @ informatik.uni-freiburg.de

Fig. 1. Experimental setup: the robot issues random commands ("motor babbling") to its joints and perceives the resulting movements of its body parts using a monocular camera. From this self-perception, it learns a compact Bayesian network that it can then use for prediction and control. The right picture shows a visualization of the robot's self-model after learning.
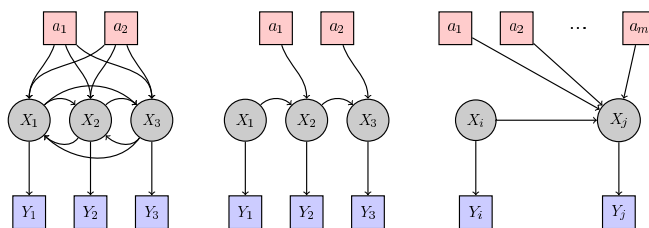


Fig. 2. **Left:** Initially, the Bayesian network representing the robot's body scheme is fully-connected. **Middle:** After training, only the local models most consistent with the observed data are retained to form a sparse kinematic model for the whole system. **Right:** Template of a local model for a body part $X_j$ which depends on its predecessor $X_i$ in the kinematic chain and all available action signals $a_1, \ldots, a_m$. Observed variables are depicted as rectangles, while hidden variables are depicted as circles.

manipulators in conjunction with visual self-perception. We propose to learn a Bayesian network for the robot's kinematic structure including the forward and inverse models relating action signals and body pose. More precisely, we start with a fully connected network containing all perceivable body parts and available action signals, to perform random "motor babbling," and to iteratively reduce the network complexity by analyzing the perceived body motion. At the same time, we learn non-parametric regression models for all dependencies in the network, which can later be used to predict the body pose when no perception is available or to allow for gradient-based posture control. In experiments with real and simulated manipulator arms, we show that our approach is able to quickly learn compact and accurate models and to robustly deal with noisy observations.

## II. RELATED WORK

Several approaches for learning and adapting body schemes at different levels of complexity have been proposed in the past. Self-calibration, for instance, can be understood as

a subproblem of body scheme learning. When the kinematic model is known up to a number of parameters, they can in certain cases be efficiently estimated by maximizing the likelihood of the model given the data [7]. Genetic algorithms have been used for parameter optimization when no closed form is available [8]. To a certain extend, such methods can also be used to calibrate a robot that is temporarily using a tool [9]. However, such approaches require a parameterized kinematic model of the robot.

There have also been approaches on learning sensor-motor maps when no such model is available. For example, [10] applied Hebbian networks to discover the body scheme from self-occlusion or self-touching sensations and later learned classifiers for body/non-body discrimination from visual data [11]. Other approaches used nearest-neighbor interpolation [12] or neural networks [13]. By considering body scheme learning as a problem of function approximation, such approaches are applicable in cases even where little prior knowledge is available. Without knowing the underlying structure, however, these approaches scale poorly with an increasing number of free variables.

This problem can be tackled by reducing the dimensionality of the learning problem. Principal component analysis (PCA), for example, has been used successfully for learning efficiently high-dimensional walking gaits for humanoid robots [14]. Whereas such approaches effectively remove the redundancy in the body scheme for a particular motion sequence, they run the risk of losing important information in the projection step. Our approach makes the factorized nature of the body scheme explicit and, thus, reduces the dimensionality of the learning problem in a more direct way. In [15] the structure of the body scheme is made explicit, by formulating a model selection problem between different Bayesian networks. Here, the qualitative relation between actions and observations is optimized to describe the observed data well. By using the structural information, the robot can infer motor commands to imitate the movements of a human demonstrator. To our knowledge, the approach only qualitatively describes the effects of actions and, thus, does not allow for precise and continuous actuation or prediction.

In contrast to the above mentioned approaches, we present an algorithm that both learns the kinematic structure as well as accurate functional mappings for all local connections.

## III. A PROBABILISTIC MODEL FOR KINEMATIC CHAINS

In this work, we seek to enable a robotic system to autonomously learn the relationship between action signals $a_1, \ldots, a_m$ and body part configurations $X_1, \ldots, X_n$, which can be (partially) observed as $Y_1, \ldots, Y_n$. In our concrete scenario, in which we learn the kinematic model of a robotic manipulator arm, the action signals $a_i$ are real-valued variables corresponding to the rotation angles of the joints. These are measured using joint encoders. The $X_i \in \mathbb{R}^{4 \times 4}$ are homogeneous transformation matrices, each encoding the 6-dimensional pose of a body part relative to a reference coordinate frame. On the real robotic platform used in our experiments, the observations $Y_i$ are obtained

by tracking visual markers in 3D space including their 3D orientation [16]. Note that these observations are inherently noisy, especially in the $z$ dimension, which is the distance of the marker from the camera, and we also consider markers that are cannot be detected at all, i.e. due to self-occlusion.

Our central idea is to model the kinematic structure of the robot as a Bayesian network. The action signals $a_1, \ldots, a_m$, the (hidden) true body part configurations $X_1, \ldots, X_n$, and their observed counterparts $Y_1, \ldots, Y_n$ are modeled as individual nodes in the network, see Fig. 2. The action nodes have only outgoing edges, corresponding to their interpretation as *causing* the observed movements. The body pose nodes have in- and outgoing edges, indicating that they *depend* on certain actions and on the pose of their predecessor (the body part to which they are attached to), and that they *influence* the pose of their succeeding body parts. The sub-graph containing only the body pose nodes and their connections corresponds to the kinematic chain(s).

We denote the local transformation function of pose $X_i$ to $X_j$ given the current joint configurations $a_1, \ldots, a_m$ by $\Delta_{ij}(\mathbf{A}_{ij}) = X_i^{-1} X_j$, where $\mathbf{A}_{ij} \subset \{a_1, \ldots, a_m\}$ corresponds to the relevant subset of the complete action signal. Learning the body scheme now means

1) finding the correct network topology (which parts are directly connected?) and
2) learning the local transformation models $\Delta_{ij}(\mathbf{A}_{ij})$ for this topology

by observing the joint angles $a_1, \ldots, a_m$ and the resulting body poses $Y_1, \ldots, Y_n$.

### A. Finding the Network Topology

We are looking for a compact Bayesian network for $p(X_1, \ldots, X_n | a_1, \ldots, a_m)$ that is composed of local models of the form $p(\Delta_{ij} | \mathbf{A}_{ij})$ with $\mathbf{A}_{ij} \subset \{a_1, \ldots, a_m\}$ and $\Delta_{ij} = X_i^{-1} X_j$. A trivial solution would be to assume that all local models depend on all action signals available. Such a model, however, would not generalize well over the training data as it could not take advantage of the intrinsic redundancy of the body structure. Additionally, since the individual models would be of high dimensionality, considerably more training examples would be required than for a sparse composition of low-dimensional local models. The upper arm of a robot, for example, only depends on the position of the trunk and the shoulder joints, while the lower arm would only depend on the position of the elbow and the remaining joints. If, however, intermediate body parts are not observable – a situation that we evaluated in the experimental section – higher order local models are necessary for describing the full system. In order to find the optimal dimensionality of local models, we formulate a search problem over all possible dependencies and model dimensionalities. The search space has an upper bound of $n_{localmodels} = \sum_{k=1}^{n} \binom{m}{k} \binom{n}{2}$ local models that would have to be considered, since the ordering and dependencies of joints and observed body parts is initially unknown to the robot. In order to connect all $n$ body poses in the Bayesian network, exactly $(n-1)$ local models
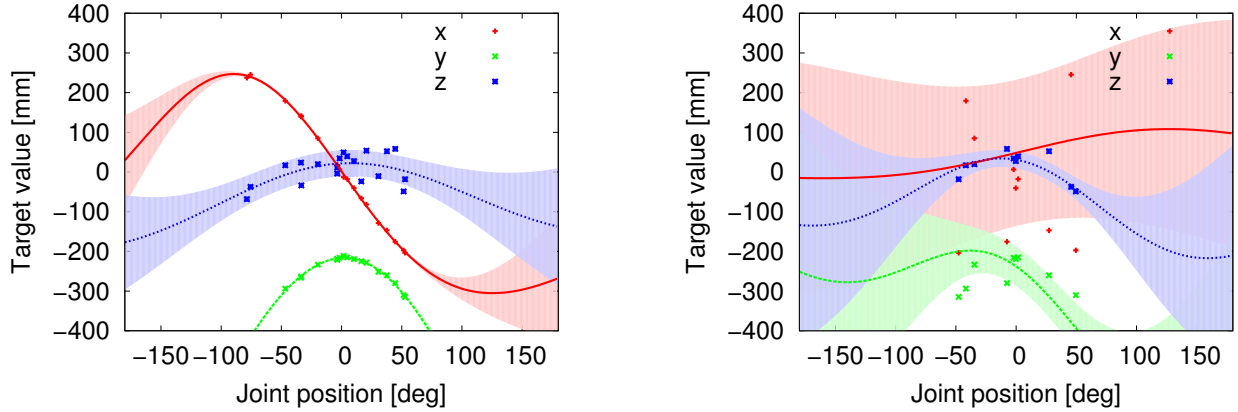
Fig. 3. **Left:** Example of an accurate local model learned for two body parts and an action variable. Note the low predictive variance for the $x$- and $y$ components as well as the higher noise in the $z$ dimension, which is due to higher measurement uncertainty in this direction. **Right:** Less accurate model learned for the same body parts but a different action variable. Such a local model is less likely to be part of the Bayesian network describing the full kinematic chain of the robot since, on average, its predictions are less accurate.

need to be selected, thus yielding the astronomical number of $n_{structures} = \binom{n_{localmodels}}{n-1}$ network structures to be considered.

In practice, the number of local models $n_{localmodels}$ can be reduced drastically by using simple search heuristics, such as evaluating the local models ordered by their complexity $|\mathbf{A}_{ij}|$ and to stop the search when a certain level of model accuracy is attained. The number of network structures $n_{structures}$ reflects the total number of possible spanning trees of the network; fortunately, it is sufficient to find the minimal spanning tree. The cost of each edge is determined by the residual sum of squares (RSS) of the corresponding local model $p(\Delta_{ij}|\mathbf{A}_{ij})$ on a validation data set that has been sampled during training but has not been included in the training set for the local models. Note, that model selection based on the RSS measure worked particularly well in our experiments, but other selection criteria like the Bayesian information criterion (BIC) or the data likelihood can be used likewise.

As a result, the recovered network is indeed Bayesian, i.e., it is a directed, acyclic graph, and factorizes the body scheme into the more compact representation

$$p(X_1, \ldots, X_n | a_1, \ldots, a_m) = P(X_t) \prod_{<i,j> \in E} p(X_i | X_j, \mathbf{A}_{ij})$$
$$= P(X_t) \prod_{<i,j> \in E} p(\Delta_{ij} | \mathbf{A}_{ij}) , \quad (1)$$

where $X_t$ is the root node and $E$ is the edge list of the recovered minimal spanning tree corresponding to the kinematic chain(s).

### B. Learning Local Kinematic Models

In order to learn an arbitrary local model $p(\Delta_{ij}|\mathbf{A}_{ij})$, we need to find the non-linear mapping from a vector of action signals $\mathbf{A}_{ij}$ to the relative transformations $\Delta_{ij} = X_i^{-1}X_j$. Remember, that $\Delta_{ij} \in \mathbb{R}^{4 \times 4}$ is parameterized as a homogeneous 4x4 matrix. For simplicity, we assume

all 12 free components $\delta_{ij}^k$ of $\Delta_{ij}$ being independent of each other and thus consider the functional mapping for each component separately. Due to this simplification, we cannot guarantee that the prediction corresponds to a valid, homogeneous transformation matrix. In practice, however, invalid transformations arise only rarely and they typically lie close to similar, valid transformations, such that a simple normalization step resolves the problem.

As the true relative transformations $\Delta_{ij} = X_i^{-1}X_j$ are only observable through the noisy observations $\overline{\Delta}_{ij} = Y_i^{-1}Y_j$, we assume additive white noise on each component $\overline{\delta}_{ij}^k \sim \mathcal{N}(\delta_{ij}^k, \sigma_m)$. A flexible model for learning such non-linear functions directly from noisy observations are the popular Gaussian processes. Due to space constraints, we only give the main characteristics of this framework here and refer to [17] for details. The main feature of the Gaussian process framework is, that the observed data points are explicitly included in the model and, thus, no parametric form of $f$ needs to be specified. Moreover, the dependencies between data points is specified in an interpretable way using a parameterized covariance function $k$ and predictions yield not only the most likely function value but also the corresponding predictive uncertainty. We parameterize the covariance function $k$ using the often used square exponential formulation

$$k(\delta_{ij}, \delta_{rs}) = \sigma_f^2 \cdot \exp\left(-\frac{1}{2\ell^2}|\delta_{ij} - \delta_{rs}|\right) , \quad (2)$$

which depends on the Euclidian distance between points $\mathbf{x}_p$ and $\mathbf{x}_q$ as well as on the amplitude parameter $\sigma_f^2$ and the length-scale $\ell$. This covariance function is particularly well suited to model sinusoidal dependencies as they arise in our setting, where we wish to infer components of harmonic transformations.

The diagrams of Fig. 3 depict several typical regression results, both generated using real data of our manipulator. In the left diagram, the regression results for an accurate local model are shown. It can be seen that the training data has
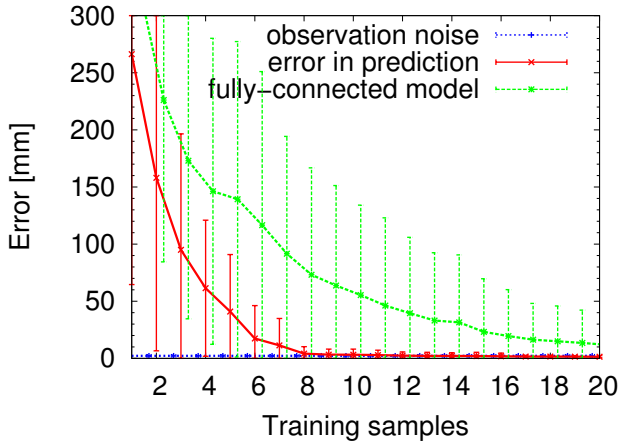
Fig. 4. Forward model evaluation on a simulated robot with low noise. The prediction error of the learned forward model quickly converges towards zero. It can also been seen that the compact Bayesian network – using a decomposition into small local models – converges much faster than the fully-connected Bayesian model which requires higher-dimensional regression models.
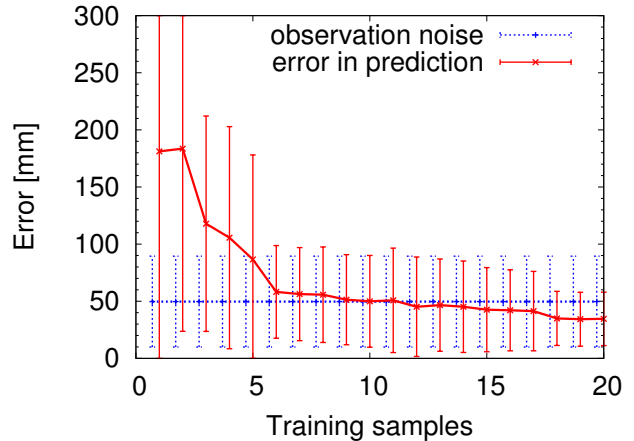


Fig. 5. Forward model evaluation on a real robot with noisy perception. Already after a few samples, the prediction accuracy of the forward model becomes better than the direct but noisy perceptions of the camera.

low noise in the x-y-components and slightly higher noise rates in the z-component. In the right diagram, we depict an alternative local model for the same transformation that is conditioned on a different action signal (x-axis). It is clearly visible that a much higher prediction error (RSS) on the validation set is to be expected for this model. Therefore, the latter model is less likely to be part of the kinematic chain for the full system.

### C. Using the Learned Body Scheme for Prediction

The learned Bayesian network can now be used as a predictive forward model. Given action signals $a_1, \ldots, a_m$, the relative transformations $\Delta_{ij}$ can be inferred from the local models $p(\Delta_{ij}|\mathbf{A}_{ij})$ of the kinematic chain. If one absolute body position, e.g., $X_1$ is known or assumed, then the absolute coordinates of all other body positions can be computed by re-arranging Equation 1.

In particular, the GPs underlying each local model yield the mean and the variance for a given joint configuration. While the mean corresponds to the maximum likelihood estimate, the variance can be used as a measure of uncertainty. In order to propagate Gaussian beliefs through the kinematic chain, we approximate the result of Gaussian multiplication again as Gaussians [18]. Note that this variance estimates can be used by the robot for active exploration, or to generate action commands that minimize the expected sensor and/or motor noise.

### D. Using the Learned Body Scheme for Control

In order to grasp an object or to imitate the actions of a human demonstrator, the robot needs an inverse model that maps from a given target position $X_{target}$ to an action command $\mathbf{a} = [a_1, \ldots, a_m]^T$ that, if executed, brings the manipulator into this position. Formally, we seek to continuously minimize the distance function $f(\mathbf{a}) = \|X_n(\mathbf{a}) - X_{target}\|$, where $X_m(\mathbf{a})$ refers to the predicted position of

body part $m$ given a motor command $\mathbf{a}$. This quantity can be estimated using Equation 1 as described in the previous subsection. As this distance function is continuous, also its Jacobian $\nabla f(\mathbf{a})$ can be evaluated, i.e.,

$$\nabla f(\mathbf{a}) \quad = \quad \left[ \frac{\partial f(\mathbf{a})}{\partial a_1}, \ldots, \frac{\partial f(\mathbf{a})}{\partial a_m} \right]^T . \qquad (3)$$

Given $\nabla f(\mathbf{a})$, it is straight forward to implement a gradient descent-based algorithm that continuously minimizes the distance function and, thus, controls the manipulator towards the target configuration. While such a "greedy" controller may get trapped in local minima of the distance function and fails to plan around obstacles, it nevertheless solves many important control tasks and it builds the basis for higher-level path-planning algorithms, such as probabilistic road-maps.

## IV. EVALUATION

We have tested our approach in a series of experiments, both on a simulated manipulator robot and a real one [1]. The experiments described in this section have been designed to demonstrate that

1) our approach yields a close to optimal and compact model when no noise is present and all quantities can be fully observed,
2) our approach is robust w.r.t. the noisy perception of a monocular camera on a real robot,
3) our approach can deal with unobserved body parts (in which case higher-order local models are needed),
4) our approach allows for online control of a real manipulator, even when self-perception is disabled after the training process.

For each experiment, 400 random action commands were generated ("motor babbling") and sent to the motors. After each action request was completed, the robot recorded the

---

[1]For a demonstration video of a real robot bootstrapping its body scheme, visit http://www.informatik.uni-freiburg.de/~sturm/media/bodyscheme-2x.avi
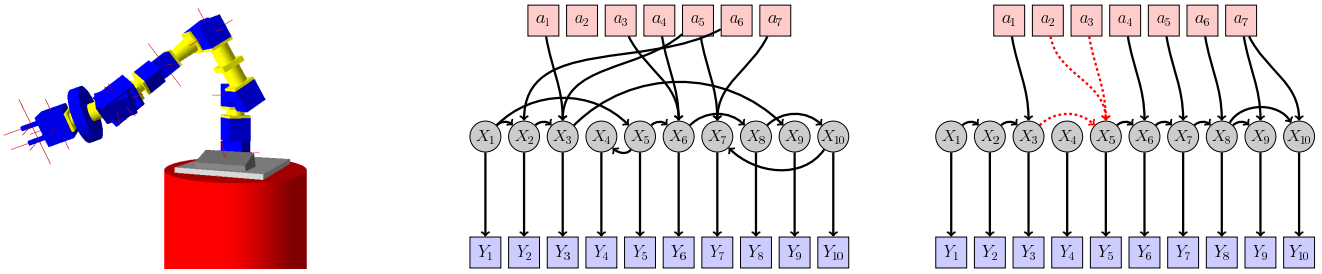
Fig. 6. Experiments with a simulated 7-DOF-manipulator consisting of 10 body parts. Body part $X_4$ was hidden and, thus, was never observed. **Left:** Screen-shot from the simulated robot. **Middle:** One of the potential Bayesian networks after the first training sample: the correct kinematic chain is not yet recovered. **Right:** Bayesian network after 10 training samples, the kinematic structure has converged to the true solution. Note that in this situation, a higher-order model (dashed, red arrows) is necessary to characterize the robot since one of the joints cannot be observed directly.

measurements from the joint encoders $a_1, \ldots, a_m$ and the observed positions $Y_1, \ldots, Y_n$ of its body parts. These data sets were then used for learning, testing, and validation. The training samples were added incrementally to the local models, in order to investigate the learning behavior. After each training sample, a test set of 40 data samples was used to measure the average accuracy of both prediction (forward model) and control (inverse model).

### A. Full Observability in Simulation

For our first experiment, we used a simulated manipulator robot with 2 rotational joints, similar to our real robot as shown in Figure 1. We added small amounts of white noise to both the measurements from the joint encoders ($\sigma_{joints} = 0.02°$) and the observations of the body positions ($\sigma_{markers} = 1$ mm). From the simulator, noise-free ground truth information was available for evaluation.

Fig. 4 gives the prediction errors of the learned model as a function of the number of training samples. Remember that a single training sample here corresponds to a pair of actions $\langle a_1, \ldots, a_m \rangle$ and visual observations $\langle X_1, \ldots, X_n \rangle$. It can be seen that the prediction error quickly converges towards zero; after only 10 training samples, the error is in the magnitude of millimeters. For comparison, a full model $p(\Delta_{13}|a_1, a_2)$ was learned directly from the training data using a 2-dimensional GP. The resulting prediction error is also given in Fig. 4. It can be seen that the compact Bayesian network composed of two local models converges much faster than the fully-connected model.

### B. Body Scheme Learning with a Real Robot

The robot used to carry out the experiments is equipped with a manipulator composed of Amtec (Schunk) PowerCube modules. With nominal noise values of $\sigma_{joints} = 0.02°$, the joint positions measured by the encoders were considered to be sufficiently accurate to compute the ground truth positions of the body parts from the known geometrical properties of the robot. Visual self-perception was implemented using a Sony DFW-SX900 FireWire-camera at a resolution of 1280x960 pixels. We attached black-and-white markers to the robot's joints (see Fig. 1), which are localized automatically using the ARToolkit vision module [16]. Per image,

the system perceives the unfiltered 6-dimensional poses of all detected markers.

The standard deviation of the camera noise was measured to $\sigma_{markers} = 44$ mm in 3D space, which is acceptable considering that the camera was located two meters apart from the robot. In the near future, we plan to develop a body part tracker similar to [19] that uses more natural visual features, such that the 6D trajectories can be recovered directly from images without the need for artificial markers.

On the real system, the measured noise levels were considerably higher than in simulation (around 44 mm). Still, as documented in Fig. 5, the body scheme converged within the first 10 observations. After about 15 training samples, the accuracy of the predicted body part positions even outperformed the accuracy of the direct observations. The latter is a remarkable result as it means that, although all local models are learned from noisy observations, the resulting model is able to blindly predict positions that are more accurate than immediate perception.

### C. Partial Observability and Long Kinematic Chains

We conducted an experiment with a simulated manipulator with 7 joints and 10 visible body parts, with a total length of 1300 mm. The manipulator has been assembled as follows (compare to Fig. 6):

- Body parts $X_1$ and $X_2$ were firmly connected to each other.
- Two fingers $X_9$ and $X_{10}$ were mounted on the 1-DOF gripper $a_7$ as the end-effector.
- The remaining body constituted a chain of visible body parts $X_2, \ldots, X_8$ and 1D rotary joints $a_1, \ldots, a_6$.

The learned forward model converges after around 10 samples, similar as in the earlier experiments. The average prediction error after convergence was lower than 1 mm.

We then analyzed the effects of partial observability. This was realized by covering body part $X_4$ completely, such that no observations of that body part could be made. As a result, none of the 1-dimensional local models is able to represent this particular local transformation well. In such a situation, the search algorithm also considers higher-dimensional local models until a satisfactory kinematic chain is found, i.e., the predictive error over the test set becomes sufficiently
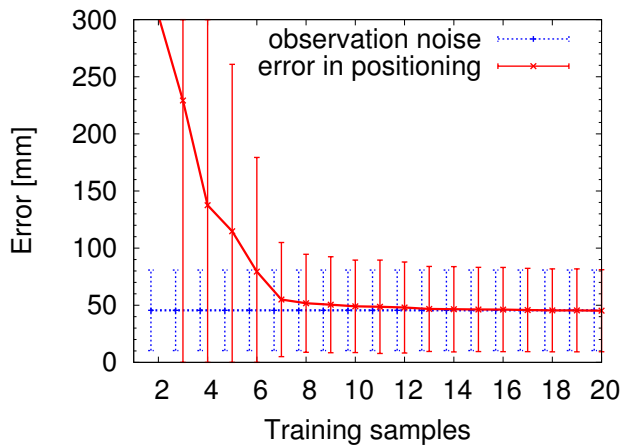
Fig. 7. Inverse model evaluation on a real robot with noisy perception. A gradient-descent algorithm is used to find the action command that minimizes the predicted distance to the desired target position. The positioning error reaches after a few samples the magnitude of the sensor noise.

small. The higher order (2-dimensional) model which was automatically incorporated into the Bayesian network is highlighted in Fig. 6 using red, dashed lines.

### D. Blind Control Using the Learned Body Scheme

Finally, we evaluated the control accuracy of our system after the body scheme had been learned from scratch. Fig. 7 shows the results on the real robot with noisy perception. The average positioning error converges after 10 training samples approximately at the level of the observation noise. This result is slightly worse than the prediction accuracy of the forward model. As a possible reason for this, the robot here has to deal with the observation noise twice: first, the model was learned from noisy data, and second, for testing against the desired target position the robot again received noisy information.

### V. CONCLUSIONS

We presented an approach that allows a robot to learn its own sensorimotor model from scratch using "motor babbling" and self-perception. The fundamental idea is to decompose the problem of learning a large and complex kinematic model into smaller pieces. Based on this decomposition, we formulate a search problem over network structures and independent learning problems for local forward models. As a result, the robot no longer relies on a model supplied by an engineer. In real-world experiments, we showed that our approach can deal with noisy perceptions. In a simulated experiment with a much larger robot, we also showed how a robot can use the local models to search efficiently for the kinematic structure in the case of partially observable body parts.

For this paper, we clearly chose the experimental setup in such a way that the problem becomes tractable. In the future, we therefore want to address problems like ambiguities of data association, partial pose observations, and efficient methods for heuristically searching through the space of alternative explanatory networks. We think that robotic self-perception and body scheme acquisition have a large potential. Therefore, we also want to address questions like how a robot can update small parts of its Bayesian network locally, for example by extending it temporarily when using a tool, or adapting it in case of hardware failures or deformation.

### REFERENCES

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

[2] H. Choset, K. M. Lynch, S. Hutchinson, G. A. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun, *Principles of Robot Motion: Theory, Algorithms, and Implementations*. MIT Press, June 2005.

[3] E. M. Rosales and Q. Gan, "Forward and inverse kinematics models for a 5-dof pioneer 2 robot arm," University of Essex, UK, Tech. Rep. CSM-413, 2004.

[4] A. N. Meltzoff and M. K. Moore, "Explaining facial imitation: A theoretical model," *Early Development and Parenting*, vol. 6, pp. 179–192, 1997.

[5] A. Maravita and A. Iriki, "Tools for the body (schema)," *Trends in Cognitive Sciences*, vol. 8, no. 2, pp. 79–86, 2004.

[6] N. P. Holmes and C. Spence, "The body schema and the multisensory representation(s) of peripersonal space," *Cognitive processes*, vol. 5, no. 2, pp. 94–105, 2004.

[7] N. Roy and S. Thrun, "Online self-calibration for mobile robots," in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 1999.

[8] J. C. Bongard, V. Zykov, and H. Lipson, "Automated synthesis of body schema using multiple sensor modalities," in *Proc. of the International Conference on the Simulation and Synthesis of Living Systems (ALIFEX)*, 2006, pp. 220–226.

[9] C. Nabeshima, M. Lungarella, and Y. Kuniyoshi, "Timing-based model of body schema adaptation and its role in perception and tool use: A robot case study," in *Proc. of the IEEE International Conference on Development and Learning (ICDL2005)*, 2005, pp. 7–12.

[10] Y. Yoshikawa, K. Hosoda, and M. Asada, "Binding tactile and visual sensations via unique association by cross-anchoring between double-touching and self-occlusion," in *Proc. of the International Workshop on Epigenetic Robotics*, Genoa, Italy, 2004, pp. 135–138.

[11] Y. Yoshikawa, Y. Tsuji, K. Hosoda, and M. Asada, "Is it my body? – body extraction from uninterpreted sensory data based on the invariance of multiple sensory attributes –," in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, Sendai, Japan, 2004, pp. 2325–2330.

[12] P. Morasso and V. Sanguineti, "Self-organizing body-schema for motor planning," *Journal of Motor Behavior*, vol. 26, pp. 131–148, 1995.

[13] L. Natale, "Linking action to perception in a humanoid robot: A developmental approach to grasping," Ph.D. dissertation, Univerity of Genoa, Italy, May 2004.

[14] D. Grimes, R. Chalodhorn, and R. Rao, "Dynamic imitation in a humanoid robot through nonparametric probabilistic inference," in *Proc. of the Robotics: Science and Systems (RSS)*, Philadelphia, USA, August 2006.

[15] A. Dearden and Y. Demiris, "Learning forward models for robots," in *Proc. of the Int. Conf. on Artificial Intelligence (IJCAI)*, Edinburgh, Scotland, 2005, pp. 1440–1445.

[16] M. Fiala, "Artag revision 1, a fiducial marker system using digital techniques," National Research Council Canada, Tech. Rep. NRC/ERB-1117, 2004.

[17] C. E. Rasmussen and C. K. Williams, *Gaussian Processes for Machine Learning*. Cambridge, Massachusetts: The MIT Press, 2006.

[18] R. Ware and F. Lad, "Approximating the distribution for sums of products of normal variables," University of Canterbury, England, Tech. Rep. UCDMS 2003/15, 2003.

[19] J. Yan and M. Pollefeys, "Automatic kinematic chain building from feature trajectories of articulated objects," in *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 712–719.