Inferring Guidance Information in Cooperative Human-Robot Tasks

Erik Berger[1], David Vogt[1], Nooshin Haji-Ghassemi[2], Bernhard Jung[1], and Heni Ben Amor[2]

***Abstract*— In many cooperative tasks between a human and a robotic assistant, the human guides the robot by exerting forces, either through direct physical interaction or indirectly via a jointly manipulated object. These physical forces perturb the robot's behavior execution and need to be compensated for in order to successfully complete such tasks. Typically, this problem is tackled by means of special purpose force sensors which are, however, not available on many robotic platforms. In contrast, we propose a machine learning approach based on sensor data, such as accelerometer and pressure sensor information. In the training phase, a statistical model of behavior execution is learned that combines Gaussian Process Regression with a novel periodic kernel. During behavior execution, predictions from the statistical model are continuously compared with stability parameters derived from current sensor readings. Differences between predicted and measured values exceeding the variance of the statistical model are interpreted as guidance information and used to adapt the robot's behavior. Several examples of cooperative tasks between a human and a humanoid NAO robot demonstrate the feasibility of our approach.**

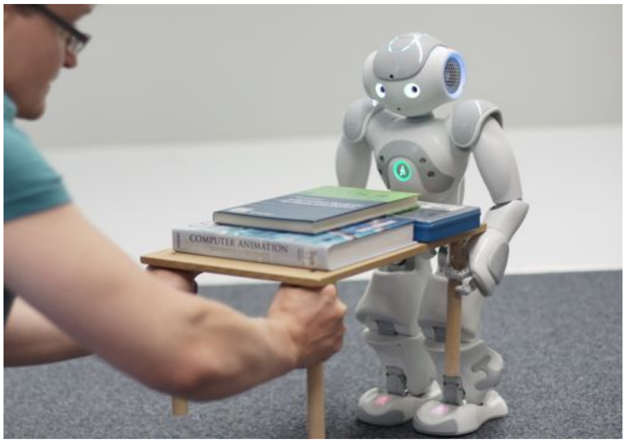

Fig. 1: A robot is following the human guidance during a cooperative transportation task. The intended walking direction is inferred using probabilistic machine learning methods.

I. INTRODUCTION

An important vision of research in robotics and artificial intelligence is the development of robot assistants that can support humans in physically demanding tasks. A robot assistant can hand over a distant tool or help in lifting a heavy object. However, close-contact human-robot cooperation of this kind requires significant sensing capabilities in order to ensure safe and meaningful physical interactions between humans and robots.

Humans often rely on tactile and force feedback as a means of communication during joint physical activities. For example, during joint manipulation and transportation tasks the heading in which to carry an object is often communicated using a gentle push in that direction. In general, we note that activities which are based on the cooperation of pairs or teams of persons, e.g., sports or dancing, make intensive use of such guidance information which is communicated through touch and force information. In order for a robot to engage in similar cooperative activities, it needs to be able to access and interpret human guidance through the analysis of externally exerted tactile forces. However, this requires special purpose sensors, which are often expensive, heavy and prone to error. In practice, many sensors return non-zero readings when the robot executes a motor task, e.g., walking. As a result, it is often difficult to distinguish external perturbations of the sensor readings, which are caused by the human interaction partner, from sensor values that are caused by the execution of the robot behavior itself.

In this paper, we present a machine learning approach that can be used to identify and interpret external perturbations caused by human interaction and guidance. In this approach, we create a statistical model of the regular pattern of sensor readings during the execution of different robot behaviors. In turn, this is used to identify irregular deviations in the sensor readings, which are caused by the human interaction partner. An important advantage of this approach is that it can be used with low-cost sensor technologies. Using machine learning methods we can combine readings from different types of sensors, e.g., accelerometers and pressure sensors, in order to generate a robust statistical model.

We will show how such statistical models can be used to produce intuitive physical interaction between humans and robots in cooperative tasks as shown in Figure 1. In particular, we will show how these models are used to allow a NAO robot to identify and interpret human forces during a joint transportation task.

II. RELATED WORK

Over the last twenty years, human-robot interaction has become a major research topic within robotics, with dedicated conferences, books and journal issues. Many researchers have focused on speech and gesture as a means of establishing communication protocols between robots and humans. In recent years, however, various researchers have focused on using cues from direct, physical interactions in order

[1] Institute of Computer Science, Technical University Bergakademie Freiberg, Bernhard-von-Cotta-Str. 2, 09599 Freiberg, Germany

[2] Department of Computer Science, Technical University Darmstadt, Hochschulstr. 10, 64289 Darmstadt, Germany

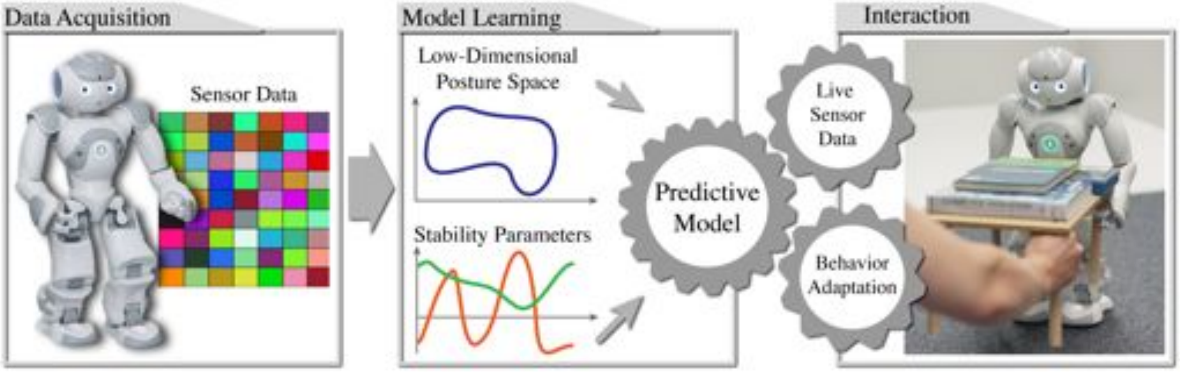

Fig. 2: An overview of the presented machine learning approach. Left: Sensor data during regular behavior execution is acquired. Center: After pre-processing the data, we learn a probabilistic model of the evolution of stability parameters. Right: At runtime, we compare measured to predicted sensor data and use the deviation to control the robots behavior.

to realize non-verbal communication between robots and humans.

Wang et al. [1] present a robot that can adapt its dancing steps based on the external forces exerted by a human dance partner. Ben Amor et al. [2] uses touch information to teach new motor skills to a humanoid robot. In this approach, the robot is not actively involved in any joint task with the human. Touch information is, therefore, only used to collect data for subsequent learning of a robotic motor skill. Robot learning approaches that are based on such kinesthetic teach-in have gained considerable attention in the literature, with similar results reported in [3] and [4]. Following a similar scheme, Lee et al. [5] use impedance control and force-torque sensors in order realize human-robot interaction during programming by demonstration tasks. A different approach aiming at joint physical activities between humans and robots has been reported in [6]. Ikemoto et al. use Gaussian mixture models to adapt the timing of a humanoid robot to that of a human partner in close-contact interaction scenarios. The parameters of the interaction model are updated using binary evaluation information obtained from the human. The approach significantly improves physical interactions, but is limited to learning timing information.

Stückler et al. [7] present a cooperative transportation task where a robot follows the human guidance using arm compliance. In doing so, the robot recognizes the desired walking direction through visual observation of the object being transported. A similar setting has been investigated by Yokoyama et al. [8]. They use a HRP-2P humanoid robot equipped with a biped locomotion controller and an aural human interface to carry a large panel together with a human. Forces measured with sensors on the wrists are utilized to derive the walking direction. Similarly, Bussy et al. [9] also use force-torque sensors on the wrists to adapt the robot behavior during object transportation tasks. Lawitzky et al. [10] also shows how load sharing and role allocation can be used to balance the contribution of each interaction partner depending on the current situation.

The main drawback of the above approaches is that they require special aural and visual input devices or force sensors which are not present on many robot platforms. Additionally, none of the approaches utilizing force-torque sensors addresses the problem of uncertainty in the provided measurements. As a result, all of these approaches assume high-quality sensing capabilities and low-speed execution of the joint motor task. In contrast to the above approaches, we estimate the human forces using probabilistic machine learning methods. These methods allow us to naturally model uncertainties in sensor readings.

III. APPROACH

In our approach the human applies forces to the robot via direct touch or via a carried object. The human guides the robot, which consecutively has to appropriately react to the exerted forces, i.e., stand-up, walk forward, walk backwards or perform side steps. Measured forces are treated as indications for the guidance of the human, e.g., a push backwards signal the humans intention to move backwards. In this paper, we assume that the human forces *cannot* be directly measured using, for example, force-torque sensors. Instead, we estimate these forces using the internal sensors of the robot.

An overview of the approach can be seen in Figure 2. First, we execute each of the behaviors, e.g., walking or side steps, of the robot and collect sensor data at each time step. For each behavior, we then learn a predictive model of stability parameters. Using this model we can calculate the deviation between the predicted and the currently calculated stability parameters. If the deviation cannot be explained by the predictive uncertainty of the learned model, the robot attributes this effect to an external perturbation caused by the human interaction partner. The sign and size of the deviation are then used to infer the human guidance and control the robot adequately.

In the following, we will discuss each step of our approach in more detail.

A. Data Acquisition

In order to learn a predictive model of sensor values, we first collect for each motor behavior of the robot a

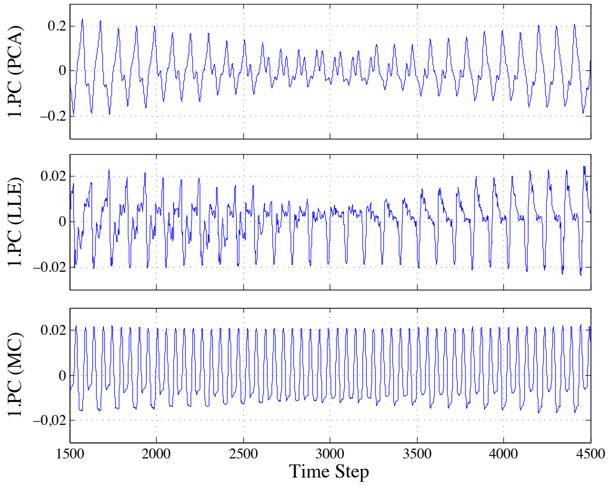

Fig. 3: Particularly in the case of PCA we see that the amplitude of the first principal component is proportional to the robot's walking velocity.

representative data set. The goal of this process is to gather information about the change of the robot parameters when *no* interaction with a human partner is realized. The recorded data includes the joint angles for all degrees-of-freedom, the raw sensor reading of the internal sensors, as well as the current parameters of the executed behavior, e.g., the walking velocity.

To increase the robustness of the later model learning method, we calculate higher-level stability parameters from the recorded raw sensor readings. In particular, we compute the *center-of-mass* (C_M) of the robot, using the position $\mathbf{p}_i$ and the mass m_i of each body part.

$$C_M = \frac{1}{M}\sum_{i=1}^{n}(\mathbf{p}_i m_i),$$

Where n is the number of body parts and M is the total mass of the robot. Additionally, we use acceleration $\mathbf{a}$ which is measured by a three-axis accelerometer, as well as the mass m, and the gravity constant $\mathbf{g}$ to calculate the *ground reaction vector* (C_G).

$$C_G = (m\mathbf{a}) - (m\mathbf{g}).$$

Finally, we also approximate the *center of pressure* (C_P) of the robot using the position $\mathbf{p}_i$ and measurement r_i of the foot pressure sensors.

$$C_P = \frac{1}{R}\sum_{i=1}^{n}(\mathbf{p}_i r_i).$$

Here, n reflects the number of pressure sensors and R is the sum of the measured pressure values.

To reduce the dimensionality of the joint angle values of the robot, we use manifold learning methods. All vectors representing the joint angle configuration of the robot at a specific time step are processed using linear or nonlinear dimensionality reduction methods. The result is a set of low-dimensional points, which reflect the dynamics of the executed motor skill. More specifically, we used *Principal Component Analysis* (PCA) [11], *Locally Linear Embedding* (LLE) [12] and *Manifold Charting* (MC) [13] for the low-dimensional embedding. Figure 3 shows the first principal component of a robot's walking gait recorded with 100Hz, where the walking speed is first decreased and then increased again. When using PCA, the amplitude of the principal component is proportional to the robot's walking velocity. A similar effect, although less salient, can be observed when using other dimensionality reduction methods too. Additionally, we observe in Figure 3 that the executed motor behavior exhibits a clear periodic behavior. We will later see that this property can be exploited in order to realize faster and more accurate learning of predictive models. After dimensionality reduction, the joint angle data is replaced by its low-dimensional projection. This has the benefit of significantly reducing the complexity of the learning task, while at the same time reducing the effect of noise and outliers, as these are often cut out from the first principal components.

B. Model Learning

Subsequently, the acquired data is used to learn a mapping from low-dimensional postures and behavior parameters to stability parameters. Input to the mapping is a vector which holds the low dimensional posture and the desired behavior parameters. The mapping produces a vector which predicts stability parameters, e.g., the C_P as illustrated in Figure 4.

For learning such a mapping, we use *Gaussian process regression* (GPR) [14]. GPR is a powerful method for learning a probabilistic, non-linear mapping between two data sets. A *Gaussian process* (GP) models a distribution $p(f)$ over functions, where f is a function mapping some input space to an output space $\mathcal{R}$.

Given a data set $\mathcal{D} = (X, y)$ with noisy output y, a nonlinear regression can be written as:

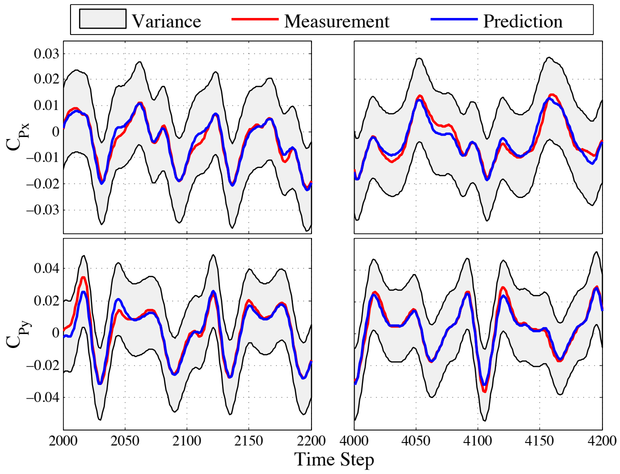

Fig. 4: The learned model is used to predict the C_{Px} and C_{Py}. The deviation between measured values and the learned prediction is located inside the allowed variance which is given by the predictive uncertainty of the Gaussian process.

$$y = f(\mathbf{x}) + \epsilon, \epsilon \sim N(0, \sigma^2)$$

where σ^2 is the variance of the noise. The prior of f is a Gaussian Process:

$$p(f) = GP(\mu, K)$$

which can be used to perform Bayesian regression:

$$p(f|\mathcal{D}) = \frac{p(f)p(\mathcal{D}|f)}{p(\mathcal{D})}$$

The prediction of output y_* at new entry point $\mathbf{x}_*$ has Gaussian probability distribution with mean $\mu(\mathbf{x}_*)$ and variance σ^2 as follows:

$$\begin{aligned} \mu(\mathbf{x}_*) \quad &= K(\mathbf{x}_*, X)[K + \sigma^2 I]^{-1} y \\ \sigma^2(\mathbf{x}_*) \quad &= K(\mathbf{x}_*, \mathbf{x}_*) - K(\mathbf{x}_*, X)[K + \sigma^2 I]^{-1} K(X, \mathbf{x}_*). \end{aligned}$$

A GP prior can be specified by the mean μ and covariance function k. Since offsets and simple trends can be subtracted out from data, we can use a zero mean prior of the GP. Thus, the key quantity is the covariance matrix K, whose elements are the output of the covariance function or kernel $k(x, x')$. A kernel encodes implicit assumptions about the underlying function f, such as smoothness or periodicity assumption. The most frequent kernel in the GP literature is the squared exponential kernel:

$$k(\mathbf{x}, \mathbf{x}') = \theta_1 \exp\left(-\frac{1}{2}\sum_{d=1}^{D}\left(\frac{x_d - x'_d}{l_d}\right)^2\right)$$

where D is the number of input dimensions. Parameters of the covariance function (θ_1, l_d) are called hyper-parameters and can be calculated based on the recorded training data. These parameters govern the properties of the kernel $k(\mathbf{x}, \mathbf{x}')$. θ_1 controls the amplitude of the function, while l_d is called characteristic length-scale and reflects the relative importance of each input dimension.

The squared exponential kernel has been shown to produce good results for a large number of application domains. Yet, for some tasks better results can be achieved by using different types of kernels. In human-robot interaction scenarios, in particular in cooperative transportation tasks, we are often faced with periodic movements. Walking, sidestepping or rotations are all examples for rhythmic movements, which repeatedly execute similar motor patterns, such as a left-right sequence of steps. Given this knowledge, we can devise a kernel function that incorporates periodicity information. In this paper, we will, therefore, use a novel periodic kernel, which is defined as:

$$k(\mathbf{x}, \mathbf{x}') = \theta_1 \exp\left(-\frac{1}{2}\sum_{d=1}^{D}\left(\frac{\sin\left(\frac{\pi}{\lambda_d}(x_d - x'_d)\right)}{l_d}\right)^2\right).$$

In addition to the hyper-parameters of the Squared Exponential kernel, our periodic kernel has a periodicity hyper-parameter λ. The similar fashion to the other hyper-parameters, λ can be learned based on the acquired training data.

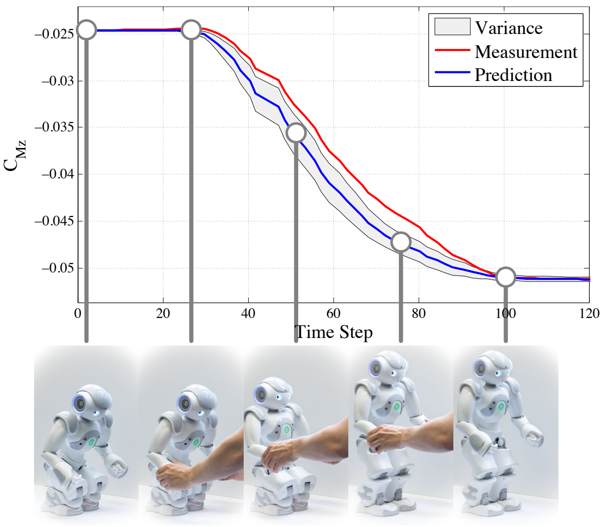

Fig. 5: The robot stands up while inferring that the human is pulling at its arms. The execution of the stand-up behavior is driven by the difference in the predicted and measured center-of-mass C_{Mz}.

C. *Human-Robot Interaction*

During interaction we measure the current angles of the robot and project them into the low dimensional posture space. The resulting low dimensional point and the current behavior parameters are mapped to the desired stability parameter via the predictive model. This predicted reference value is compared with the measured stability parameter. If the measurement is located outside the predictive uncertainty of the Gaussian process we assume that the human is guiding the robot and react by continuously adapting the behavior parameters or changing the robot's pose. For instance, the robot increases his walking velocity when the human pulls it or the robot stands up while lifting it. In our feedback loop we use a simple proportional-derivative controller to adapt the robot's behavior based on the difference between the predicted and measured stability parameters. Figure 5 illustrates the measured and predicted C_{Mz} coordinate when lifting the robot. As a result the robot can react to human guidance in real-time, e.g., the robot is standing up as long as the human is lifting it.

In the following section several experiments focusing a cooperative transportation task are explained in more detail.

IV. EXPERIMENTS

In the following experiments we use learned predictive models in order to realize close-contact cooperative tasks between a human and a robot. All experiments are carried out on the NAO robot from *Aldebaran Robotics*. As illustrated in Figure 5, we can use a model of the robot's C_{Mz} for up and down movements which can also be used for lifting and placing of objects.

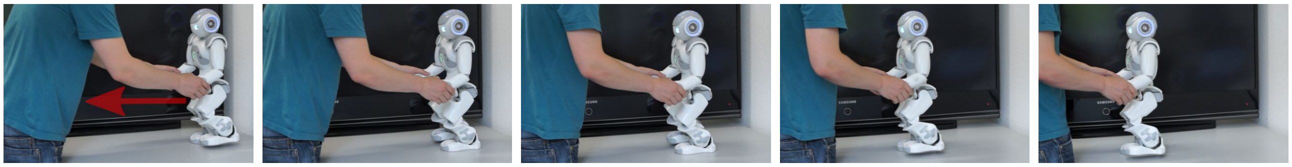

Fig. 6: The human directly guides the robot through longitudinal physical interaction. The robot infers the intended walking direction and adapts the walking parameters, i.e., heading and velocity, accordingly.

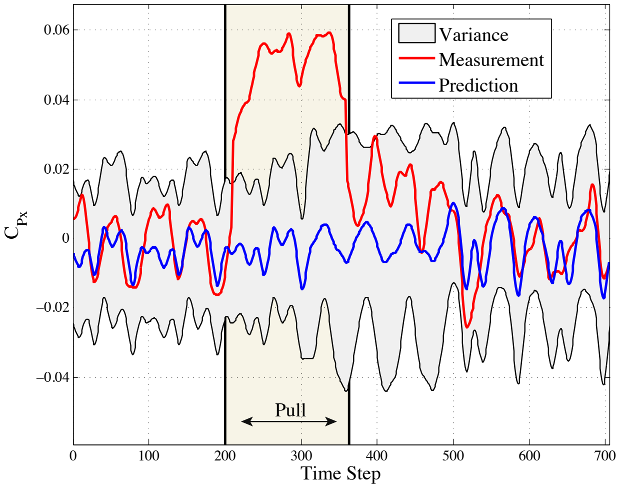

Fig. 7: The predicted and measured center-of-pressure C_{Px} during interaction. The highlighted region indicates the time period with human perturbation.

However, after lifting, we transport the object by adapting the direction and velocity of the robots walking behavior by using a predictive model for lateral and longitudinal walking/side stepping. For this, we again acquire data from the robots walking behavior. Here, the robot starts with the maximum forward walk velocity and decreases its speed until walking backward with the maximum speed. While doing so we capture the robot's pose, stability parameters and the walk velocity with 100Hz resulting in 6000 measurements. Next, we learn two different predictive models. The first model learns the mapping from the current posture and velocity to the C_{Px}, while the second model performs a mapping onto the C_{Py} parameter. While this can also be realized using a multivariate output GP, we have often experienced a better accuracy when using separate models. A video showing the results of the experiments in this section can be found under the following link: `http://youtu.be/48y0hEix2fY`.

A. Longitudinal Walking Model

In this experiment, we apply our approach to realize a direct, physical interaction between a human and robot through touch. As shown in Figure 6, the human guides the robot by applying forces to his hands. In order to respond to these forces, the speed of the walking behavior is adapted. For this, we measure the robot's current posture, C_{Px} and walking velocity. As explained above, the posture is reduced using the low dimensional posture model and then mapped to the center-of-pressure parameter C_{Px} via the predictive model. Figure 7 shows predicted and measured values during human-robot interaction. Throughout the experiment the walk velocity remains the same as long as the measured C_{Px} is located within the area of uncertainty, i.e., within the envelope spanned by the variance. However, the walking velocity is increased whenever the measured C_{Px} is outside the area of uncertainty. Up to time step 200 the velocity remains constant because the measurement is located inside the envelope. Between time step 200 and 350 the measured C_{Px} is higher than the allowed deviation, which triggers an increase in the velocity. We can also see that the prediction is adapted to the new walking velocity after time step 350. Since the human permanently interacts with the robot, we also measure small variations of the C_{Px} throughout the entire interaction.

B. Lateral Walking Model

Next, we apply our lateral walking model to a cooperative human-robot manipulation task in which the human applies forces to the robot via a carried object as shown in Figure 8. Since the hands and arms are not completely stiff, any forces of the human induce slight changes to their position thereby affecting the C_{My}. The resulting measurements and predictions are shown in Figure 9. These are due to the fact that the model does not include the weight of the table or the carried object. However, for small weights the deviation is still inside the allowed variance.

C. Periodic Kernel

We also evaluated the benefit of using our periodic kernel for learning the predictive model. Table I reports the Root Mean Squared (RMSE) error for the squared exponential kernel and the periodic kernel as performed on test data that has not been used for training. The evaluation was performed using a five-fold cross validation. The results clearly indicate an improvement when using a periodic kernel instead of a squared exponential kernel. We also noticed that, when using a periodic kernel, less training data is needed to generate a good predictive model. While not conclusive, these results demonstrate that choosing a periodic kernel can significantly improve the quality and speed of learning, when generating predictive models for rhythmic motor skills.

V. CONCLUSION AND FUTURE WORK

In this paper we proposed a behavior adaptation approach for cooperative human-robot interaction tasks. Using a GPR and a novel periodic kernel, we learn a model of the regular

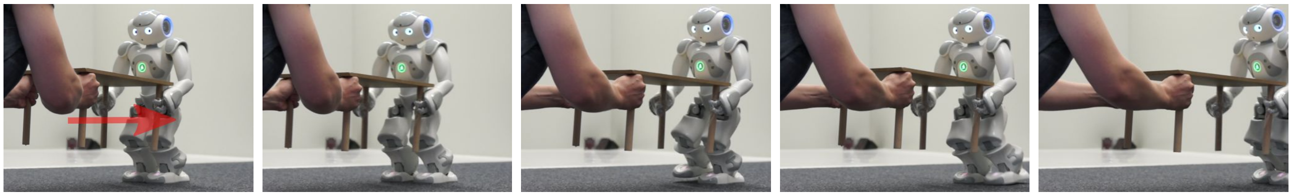

Fig. 8: A cooperative human-robot transportation task. The human applies lateral forces on the robot via a carried object. The robot reacts by side-stepping.

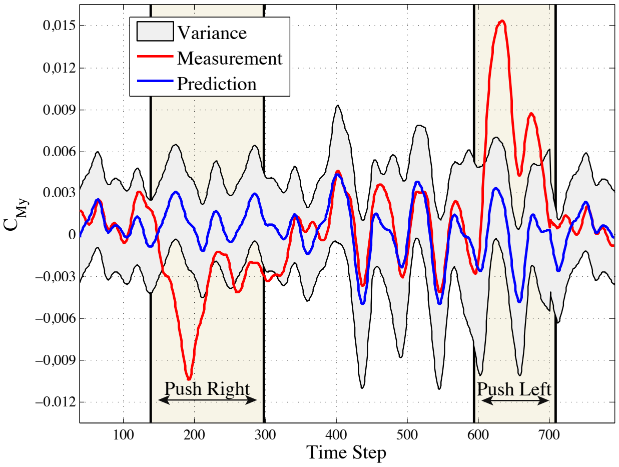

Fig. 9: The predicted and measured center-of-mass C_{My} during a cooperative transportation task. Whenever the table is pushed to the left or to the right, the robot responds with adequate side-steps to stabilize the C_{My}.

TABLE I: Comparison of Squared Exponential and Periodic kernels

Kernels	Root Mean Squared Error
Squared Exponential	0.0196
Periodic	0.0005

sensory consequences of a motor skill. At runtime, we can analyze deviations from this regular pattern to infer information about human guidance during physical interaction. We have shown that this approach can be used to realize interesting cooperative tasks involving a human and a robot. By using a probabilistic approach, we account for the inherent uncertainty in the sensor readings of the used robot. We are currently extending our framework to scenarios that additionally involve interactions with the environment. In such scenarios the robot needs to differentiate between the different sources of external perturbations, e.g., the slope of the ground vs. the force of the human partner. To this end, we want to investigate the use of online blind-source separation algorithms. Furthermore, we want to increase the efficiency of our regression model by using sparse approximation algorithms. More specifically, we want to analyze the required amount of training data needed to learn sufficiently good predictive model. Finally, we want to investigate how different predictive models can be used in combination and in a hierarchy in order to generalize learned stability information to entirely new situations and behaviors.

VI. Acknowledgment

The work presented here is funded through the EC's Seventh Framework Programme under grant agreement n ICT-600716.

References

[1] H. Wang and K. Kosuge, "Control of a robot dancer for enhancing haptic human-robot interaction in waltz," *IEEE Trans. Haptics*, vol. 5, no. 3, pp. 264–273, Jan. 2012.

[2] H. Ben Amor, E. Berger, D. Vogt, and B. Jung, "Kinesthetic bootstrapping: Teaching motor skills to humanoid robots through physical interaction," in *KI 2009: Advances in Artificial Intelligence*. Springer Berlin Heidelberg, 2009, pp. 492–499.

[3] S. Calinon, *Robot programming by demonstration: A probabilistic approach*. EPFL Press, 2009.

[4] J. Kober and J. Peters, "Policy search for motor primitives in robotics," *Machine Learning*, vol. 84, no. 1, pp. 171–203, 2011.

[5] D. Lee and C. Ott, "Incremental kinesthetic teaching of motion primitives using the motion refinement tube," *Autonomous Robots*, vol. 31, no. 2-3, pp. 115–131, 2011.

[6] S. Ikemoto, H. Ben Amor, T. Minato, B. Jung, and H. Ishiguro, "Physical human-robot interaction: Mutual learning and adaptation," *IEEE Robotics & Automation Magazine*, vol. 19, no. 4, pp. 24–35, 2012.

[7] J. Stückler and S. Behnke, "Following human guidance to cooperatively carry a large object," in *Humanoids'11*, 2011, pp. 218–223.

[8] K. Yokoyama, H. Handa, T. Isozumi, Y. Fukase, K. Kaneko, F. Kanehiro, Y. Kawai, F. Tomita, and H. Hirukawa, "Cooperative works by a human and a humanoid robot," in *Proceedings. ICRA '03. IEEE International Conference on Robotics and Automation 2003.*, vol. 3, 2003, pp. 2985–2991 vol.3.

[9] A. Bussy, P. Gergondet, A. Kheddar, F. Keith, and A. Crosnier, "Proactive behavior of a humanoid robot in a haptic transportation task with a human partner," in *RO-MAN, 2012 IEEE*. IEEE, 2012, pp. 962–967.

[10] M. Lawitzky, A. Mortl, and S. Hirche, "Load sharing in human-robot cooperative manipulation," in *RO-MAN, 2010 IEEE*, 2010, pp. 185–191.

[11] I. T. Jolliffe, *Principal Component Analysis*, 2nd ed. Springer, 2002.

[12] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, pp. 2323–2326, 2000.

[13] M. Brand and M. Brand, "Charting a manifold," in *Advances in Neural Information Processing Systems 15*. MIT Press, 2003, pp. 961–968.

[14] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.