The Seventeenth International Symposium on Artificial Life and Robotics 2012 (AROB 17th '12),
B-Con Plaza, Beppu, Oita, Japan, January 19-21, 2012

A study on the use of tactile instructions for developing robot's motions

Fransiska Basoeki[1,2], Fabio DallaLibera[1,2], Takashi Minato[2,3], and Hiroshi Ishiguro[1,2]

[1] Graduate School of Engineering Science, Osaka University, Toyonaka, Osaka 560-8531, Japan
(fransiska.basoeki@irl.sys.es.osaka-u.ac.jp)
[2] JST ERATO Asada Project, Suita, Osaka 565-0871, Japan
[3] ATR Hiroshi Ishiguro Laboratory, Kyoto 619-0288, Japan

Abstract:
Developing motions for humanoid robots is a time consuming task. However, we note that sport or dance instructors can easily adjust their students' postures by simple touches. This suggests the possibility of exploiting touch for motion development, and allows us to propose a methodology based on this concept. This requires defining how the robot should interpret user's touches. We propose a supervised learning approach for coping this issue, and verify its feasibility experimentally. We then study the data collected by the algorithm, and show that the system is usable both for motion development and as a tool for studying human-robot tactile communication. In particular, considerations on the sparsity that characterize the whole process are presented, and policies for an efficient interpretation of tactile instructions are drawn.

Keywords: humanoid robot, human-robot interaction, touch, tactile communication

1 INTRODUCTION

Humanoid robots often have a high number of degrees of freedom, that makes developing motions challenging. Having robots to do automatic learning of the motion is often impossible. To compensate this, various ways of transferring human knowledge into the robot exists in literature. When the task is known in advance, the programmer may insert his knowledge in the robot's control algorithms, such as in the design of a Central Pattern Generator structure [1]. When the task is only partly known in advance, the programmer can provide modules or motion primitives, as in the Mimesis model [2], while the final user specifies the task motion by composing these elementary modules. Finally, the task may be unknown a priori and in such case the final user must be able to transfer its knowledge to the robot directly, as in Motion retargetting [3] and Kinesthetic demonstration [4].

Our proposed method, Teaching by Touching (TbT), aims to tackle the third case. The idea is to mimic the way sport or dance instructors use touch to correct their students' movements. In conventional approaches [4], robots are completely passive during the learning process. The user is required to force the posture of the robot to the desired one, by application of the necessary forces. Conversely, we propose to equip the robot with knowledge on the meaning of touch instructions, and give it an active role in interpreting the given touch for moving according to the estimated user's intention.

The main issue of such a system lies in the definition of how the touch instruction should be interpreted by the robot. Different approaches exist. The first is to force the user to learn and use a fixed protocol. In this case the robot interpretation algorithms can be very simple. It is sufficient to interpret the instructions according to the mapping. However, the user needs to remember which touch instruction brings the robot to his desired posture, and this may be difficult for inexperienced users. The other way is modeling the way human's communicate through touch. In this case the user can just intuitively provide a touch instruction and the robot will try to interpret the meaning based on all of the available information.

Our system takes the second approach, as explained in the following Section. Section 3 will present the hardware and the system we developed, which can be used both for motion development and for studying the mapping between instructions and desired robot responses. The ability of studying the mapping is important, because it allows extraction of general criteria for the interpretation of tactile instructions, that can be used for the future design of better tactile instruction interpretation algorithms. Section 4 will conclude the paper by summarizing the main results.

2 ALGORITHM

In order to build a model of the mapping between touch instructions and desired responses, data collection of how human interpret touch is necessary. Our system collects these data online, during the motion development itself. In this way no initial model is required, and the model can be progressively refined as the user interacts with the robot.

Fig.1 schematizes the data collection methodology. When the user applies a touch instruction and the robot takes a wrong interpretation, the user notifies the robot that the interpretation is wrong, and shows the correct interpretation by manually moving the robot to the pose as he expected.

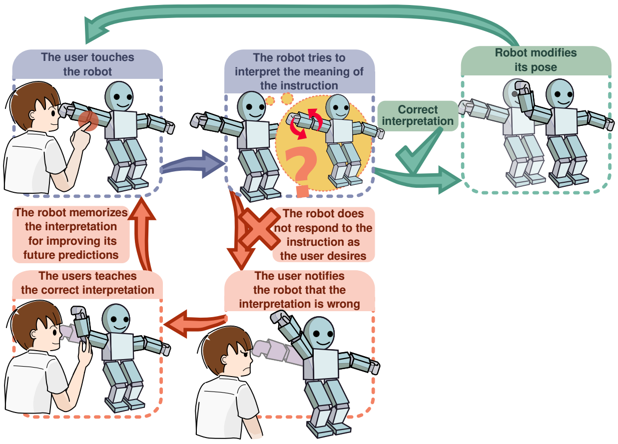

Fig. 1. Teaching by Touching motion development process.

The robot records this example of touch instruction and corresponding desired response into a database, and uses it for improving subsequent interpretations.

The database of collected data is taken as input by a Kernel regression algorithm, used for the computation of the expected meaning of the instructions received. Formally, the input $I \in \mathbb{R}^{n+m+o}$ consists of the touch sensor information $\bar{I} \in \mathbb{R}^n$, where n is the number of sensors, and the context of the robot $\check{I} \in \mathbb{R}^{m+o}$, included becaus it could influence the correct interpretation of the tactile instruction. In our implementation, the context is given by the current position of the m motors and $o = 2$ accelerometer values that represent the robot's spatial orientation. The output is a robot response, properly a rotation angle for each of the motors $M_* \in \mathbb{R}^m$.

The output of the algorithm is defined as $M_* = \sum_{i=1}^{E} \omega(I_*, I_i) M_i$ where E is the number of examples in the database, I_* is the current input from the sensors, I_i is the i-th example input, M_i is the i-th example posture change, and $\omega(I_*, I_i)$ is a kernel function, defined as

$$\omega(I_*, I_i) = \begin{cases} 0 & if\ \exists s : s \in \Psi_i \wedge s \notin \Psi_* \\ \frac{\prod_{s \in \Psi_i} \bar{I}_*^{(s)} / \bar{I}_i^{(s)}}{1+\sqrt{\|\check{I}_* - \check{I}_i\|_2^2 + \sum_{s:\notin \Psi_i} (\bar{I}_*^{(s)})^2}} & otherwise \end{cases} \tag{1}$$

with $\bar{I}_i^{(s)}$ denoting the input from the s-th tactile sensor and Ψ_i the set of sensors pushed in the i-th example. This kernel assures that when the sensors are pushed further the joint rotation angles are larger. Its denominator has the role of reducing the influence of the examples that present a context strongly different from the current context $\check{I}_*$. The condition on the set Ψ_* imposes to ignore all the examples relative to the pressure of touch sensors not being pressed in I_*.

The output of the algorithm M_* is used to modify the motion. In particular, the motion of the robot is defined as a set of keyframes F_k that specify the position of all the motors for a certain time t_k of the motion (positions for times $t_k < t < t_{k+1}$ are obtained by linear interpolation of F_k and F_{k+1}). After selecting a time instant t on a GUI, the users can touch the robot, and modify its posture at time t from F_t to to $F_t + M_*$ by simple touches.

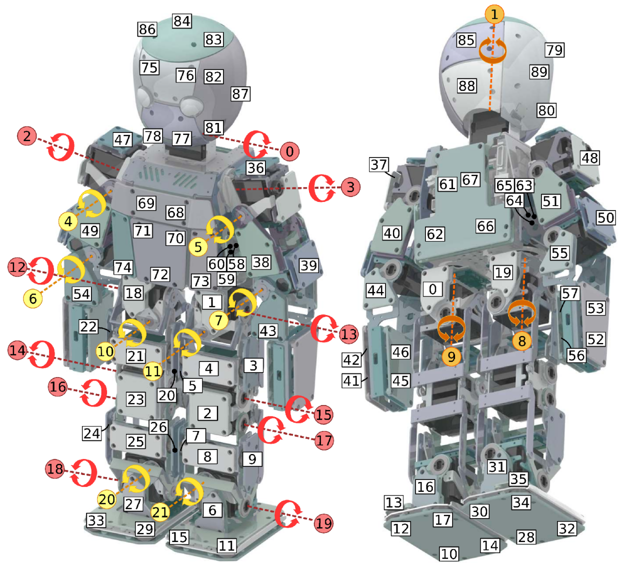

Fig. 2. Diagram of M3 Neony's motors and sensors.

3 EXPERIMENT

To verify the feasibility of the approach, we requested four people that never used the TbT system to develop a motion based on the first half of *Algorithm Exercise*, a dance from a Japanese TV show. The users had a reference motion video, but were allowed to decide when they are done developing the motion.

3.1 Hardware

The experiments were performed with a humanoid robot capable of recognizing touch, M3-Neony [5], a humanoid robot of small size equipped with a high number of tactile sensors developed as a suitable platform for tactile interaction. M3-Neony features 22 41KgF·cm servomotors, 90 tactile sensors (shown in Fig. 2), 3 accelerometers, 2 gyroscopes, 2 cameras and 2 microphones. The tactile sensors are composed of photointerruptors that translate the change of the force applied to the robot into a change in the light received by a phototransistor, as shown in Fig. 3. The experimental setup can be seen in Fig. 4. In particular, a pedal is used to notify the robot that its tactile interpretation is wrong, and allows the user to start providing the correct interpretation, that the robot stores as a (I_i, M_i) couple.

3.2 Results

All the users that attended the experiment performed the task with the proposed system without difficulties, or need for assistance, showing that the system reveals itself to be

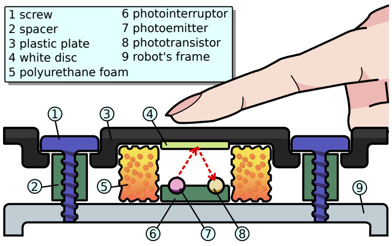

Fig. 3. Schema of tactile sensor structure of M3-Neony.

intuitive for first time users. The development time for the four users was, respectively, 221, 222, 176 and 476 minutes. Direct inspections of video recordings of the interaction show that users spent much time for robot stabilization, beyond our expectations. This confirms experimentally the importance of applying self stabilizing techniques, as the ones in [6], even for the development of quasi-static motions. Future works will introduce the concept of motion stability in the interpretation of tactile instructions, to allow the robot interpret the user's desire of making the robot balanced and let the robot compute the necessary adjustments.

One of the advantages of the systems is, as previously stated, the possibility of studying features of the way users employ touch for teaching. One of the main features that can be noticed by direct data inspection is the sparsity in the tactile interpretation taught by the users. In practice, users mainly touched few sensors and associated interpretations that involve the movement of few motors. As a more quantitative analysis, Table 1 reports the mean over the examples of the Gini index for the sensors ($G(\bar{I}_i)$) and for the motors ($G(Mi)$), $1 \leq i \leq E$.

This strong sparsity suggests us that the mapping between sensors inputs $I_i \in \mathbb{R}^n$ and motors $M_i \in \mathbb{R}^m$ could be sparse as well. To verify this hypothesis, we approximate the mapping with a linear function, and study the effect of enforcing sparsity in the mapping. More specifically, let us consider M_i, $1 \leq i \leq e$. Let us train a linear predictor $B_e \in \mathbb{R}^{m\times n+1}$ such that $M_i \approx B_e \begin{bmatrix} 1 & \bar{I}_i \end{bmatrix}^T$ by setting the k-th row of

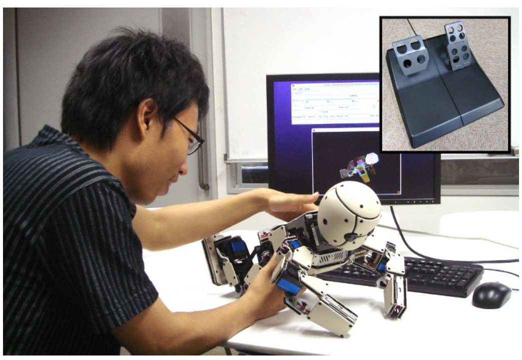

Fig. 4. Experiment environment, inset shows the pedal.

Table 1. Gini Index

	A	B	C	D
$avg_{1\leq i\leq E} < G(\bar{I}_i)$¿	0.96	0.90	0.98	0.92
$avg_{1\leq i\leq E} < G(Mi) >$	0.82	0.78	0.87	0.70

$B_e^{(k)}$ as $[b_e^{(k)}\beta_e^{(k)}]$, with $b_e^{(k)} \in \mathbb{R}$ and $\beta_e^{(k)} \in \mathbb{R}^n$ minimizers for the cost function

$$\frac{1}{2N}\sum_{i=1}^{e}\left(M_i^{(k)} - b_e^{(k)} - \beta_e^{(k)}\bar{I}_i\right)^2 + \lambda\left[(1-\alpha)\frac{1}{2}\|\beta_e^k\|_2^2 + \alpha\|\beta_e^k\|_1^2\right] \tag{2}$$

where $M_i^{(k)}$ is the rotation for the k-th motor in M_i, $\|.\|_2$ denotes the Euclidean norm, $\|.\|_1$ the ℓ_1 norm, $\alpha, \lambda \in \mathbb{R}$ are constants. We note that, as $\alpha \to 0$ the minimization resembles classic linear regression and as $\alpha \to 1$ the minimization favors sparsity [7]. Let us test the generalization capabilities of the predictor by computing the error on the $e+1$-th couple $(\bar{I}_{e+1}, M_{e+1})$, that is, compute the error

$$\varepsilon_e = \left\| M_{e+1} - B_e \begin{bmatrix} 1 \\ \bar{I}_{e+1} \end{bmatrix} \right\|_2 \tag{3}$$

for different values of α. The average error over $2 \leq e \leq E$ is reported in Fig. 5. The graph clearly shows that increasing the sparsity of the predictor improves the performances, providing support to our hypothesis of strong sparsity in the mapping as well. Previous works [8] showed that the desired responses M_i lie in a subspace of the whole motor subspace, analogously to the tendency for motions of lying in small subspaces of the whole command space [9]. We also found that the two spaces are strongly correlated, i.e. that if we know the frames F_k, $1 \leq k \leq K$ of the motion being developed we can compute the subspace $span < F_1, \dots F_K > \subsetneq \mathbb{R}^m$ where the M_i have high probability of being located. Motivated by the previous results, we can verify whether also in this case sparsity can improve our predictions. In detail, let us define the matrix

$$\Gamma_{e+1} = \begin{bmatrix} 1 & \dots & 1 \\ F_1 & \dots & F_{K'} \end{bmatrix} \tag{4}$$

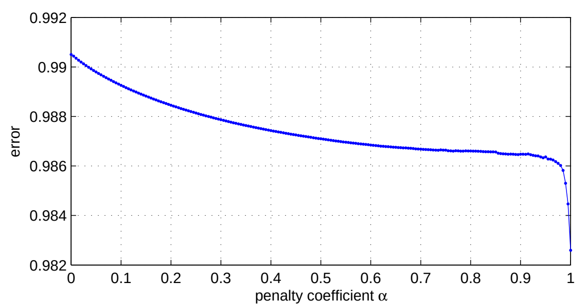

Fig. 5. Average error ε_e for the four users for different values of α.

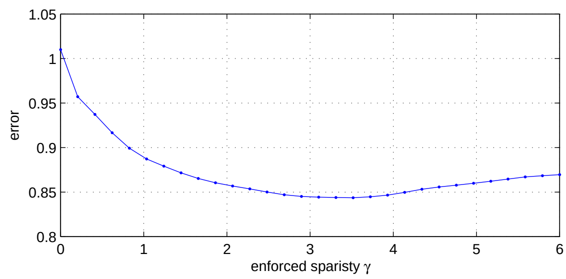

Fig. 6. Average error δ_e for the four users for different values of γ.

of the frames available before the couple $(\bar{I}_{e+1}, M_{e+1})$ is taught. Let us compute B_e and the prediction $p_{e+1} = B_e \left[\begin{array}{cc} 1 & \bar{I}_{e+1} \end{array} \right]^T$ as above, and define ρ_{e+1} as the minimizer of $p_{e+1} - \|\Gamma_{e+1}\rho_{e+1}\|_2^2 + \gamma\|\rho_{e+1}\|_1$. This new value $\rho_{e+1} \in \mathbb{R}^{m+1}$ is essentially a projection on Γ_{e+1} of the prediction p_{e+1}, whose sparsity is enforced increasing γ. Again, we compute the average error

$$\delta_e = \left\| M_{e+1} - \Gamma_{e+1} \left[\begin{array}{c} 1 \\ \bar{I}_{e+1} \end{array} \right] \right\|_2 \quad (5)$$

over the examples, and report the results in Fig. 6. We notice that the best predictions are obtained for $\gamma > 0$, confirming that also in this case sparsity of the projection coefficients can improve our estimates.

4 CONCLUSIONS AND FUTURE WORK

In this paper, we presented a system that allows to teach whole body motions to humanoid robots by physical interaction. Conceptual aspects of the exploitation of touch for motion development were briefly discussed, and a practical system implementation, comprising a small-sized robot equipped with 90 tactile sensors over the whole body was briefly introduced. This system has a two-fold role. On the one side, it allows inexperienced users to develop robot motions. On the other side, it allows studying the way users employ touch to intuitively communicate with robots.

As an example of the possible analysis that can be done on tactile instructions, this paper reports a study on the importance of sparsity in the mapping between pressed sensors and desired robot movements. First, quantitative measures on the sparsity of the input signal (touch sensors pressed) and the output signals (motors that should be moved) were reported. Successively, the possibility of improving the mapping from tactile instructions to motor movements by imposing sparsity was investigated. Finally, the idea of using the frames of the motion itself to improve the prediction on the robot movement was studied. In particular, it was shown that if we restrict the choice to a small subset of them (again, by imposing sparsity in a coefficient vector), then the prediction on the robot movement desired by the user can be improved.

This analysis provides us with important criteria for the design of new algorithms for tactile interpretation. In particular, it tells us that ensuring sparsity can give great performance improvements. For instance, if a neural network is employed for mapping tactile instructions to robot responses, then using rectifying neurons [10], that assure sparsity, appears to be a good choice. Future works will deal with the extension of the analysis reported here and the design of better, more performing algorithms for tactile interpretation based on this analysis.

REFERENCES

[1] L. Righetti and A. Ijspeert. Design methodologies for central pattern generators: an application to crawling humanoids. *Robotic Science and Systems 2006*, pp. 191–198, 2006.

[2] T. Inamura, Y. Nakamura, H. Ezaki, and I. Toshima. Imitation and Primitive Symbol Acquisition of Humanoids by the Integrated Mimesis Loop. *ICRA 2001*, pp. 4208–4213. Seoul, Korea, 2001.

[3] S. Nakaoka, A. Nakazawa, K. Yokoi, H. Hirukawa, and K. Ikeuchi. Generating whole body motions for a biped humanoid robot from captured human dances. *ICRA 2003*, pp. 3905–3910. Taipei, Taiwan, 2003.

[4] M. Hersch, F. Guenter, S. Calinon, and A. Billard. Dynamical System Modulation for Robot Learning via Kinesthetic Demonstrations. *IEEE Transactions on Robotics*, vol. 24(6):pp. 1463–1467, 2008.

[5] T. Minato, F. DallaLibera, S. Yokokawa, Y. Nakamura, H. Ishiguro, and E. Menegatti. A Baby Robot Platform for Cognitive Developmental Robotics. *Workshop on "Synergistic Intelligence" at IROS 2009*. St. Louis, MO, USA, 2009.

[6] Y. Kuroki, B. Blank, T. Mikami, P. Mayeux, A. Miyamoto, R. Playter, K. Nagasaka, M. Raibert, M. Nagano, and J. Yamaguchi. Motion Creating System for A Small Biped Entertainment Robot. *ICRA 2003*, pp. 1394–1399. Taipei, Taiwan, 2003.

[7] J. H. Friedman, T. Hastie, and R. Tibshirani. Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software*, vol. 33(1):pp. 1–22, 2010.

[8] F. DallaLibera, F. Basoeki, T. Minato, H. Ishiguro, and E. Menegatti. Teaching By Touching: Interpretation of Tactile Instructions for Motion Development. *IROS 2011*, pp. 3480–3487. San Francisco, CA, USA, 2011.

[9] B. Lim, S. Ra, and F. C. Park. Movement Primitives, Principal Component Analysis, and the Efficient Generation of Natural Motions. *ICRA 2005*, pp. 4630–4635. Barcelona, Spain, 2005.

[10] X. Glorot, A. Bordes, and Y. Bengio. Deep Sparse Rectifier Neural Networks. *Proc. of the 14th Int. Conf. on Artificial Intelligence and Statistics (AISTATS)*, vol. 15 (draft). Sardinia, Italy, 2010.