2009 IEEE International Conference on Robotics and Automation
Kobe International Conference Center
Kobe, Japan, May 12-17, 2009

Model Adaptation with Least-Squares SVM for Adaptive Hand Prosthetics

Francesco Orabona, Claudio Castellini, Barbara Caputo, Angelo Emanuele Fiorilla and Giulio Sandini

***Abstract*— The state-of-the-art in control of hand prosthetics is far from optimal. The main control interface is represented by surface electromyography (EMG): the activation potentials of the remnants of large muscles of the stump are used in a non-natural way to control one or, at best, two degrees-of-freedom. This has two drawbacks: first, the dexterity of the prosthesis is limited, leading to poor interaction with the environment; second, the patient undergoes a long training time. As more dexterous hand prostheses are put on the market, the need for a finer and more natural control arises. Machine learning can be employed to this end. A desired feature is that of providing a pre-trained model to the patient, so that a quicker and better interaction can be obtained.**

To this end we propose *model adaptation* with least-squares SVMs, a technique that allows the automatic tuning of the degree of adaptation. We test the effectiveness of the approach on a database of EMG signals gathered from human subjects. We show that, when pre-trained models are used, the number of training samples needed to reach a certain performance is reduced, and the overall performance is increased, compared to what would be achieved by starting from scratch.

I. Introduction

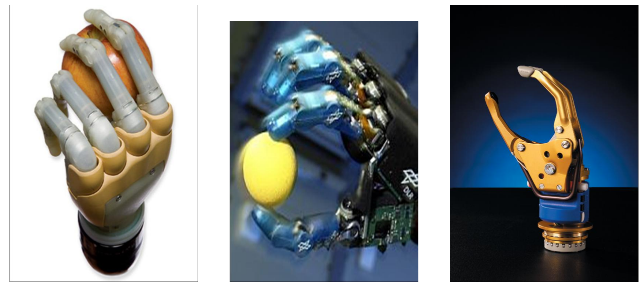
In the framework of advanced, active hand prosthetics we are witnessing a technology transfer from robotics and mechatronics. Touch Bionics's i-LIMB [1] prosthetic hand, with its five degrees-of-freedom, is a real breakthrough with respect to the previous state-of-the-art, Otto Bock's SensorHand Speed [2], which is essentially an open-close mechanism. Dexterity of hand prostheses is still far from that of state-of-the-art non-prosthetic mechanical hands, such as, e.g., the DLR II [3] (not to mention a human hand, of course), but things are getting better thanks to the afore-mentioned inter-disciplinary exchange (see Figure 1). Several EU-funded projects (e.g., CyberHand [4] and SmartHand [5]) testify the enthusiasm in the field.

The simplest, cheapest and therefore most used technique for interfacing the patient with the prosthesis is surface electromyography (EMG): activation potentials of the patient's stump residual muscles are detected to move the hand to predefined positions. But the control schema employed is rather poor, using two or three electrodes to issue an "open/close" command or, in the more advanced case of the i-LIMB, to choose among a predefined set of grasp shapes.

F. Orabona *(corresponding author)* and B. Caputo are with Idiap Research Institute, rue Marconi, 19, Case Postale 592, CH-1920 Martigny, Switzerland. e-mail: francesco.orabona@idiap.ch, barbara.caputo@idiap.ch

C. Castellini is with the LIRA-Lab, University of Genova, viale F. Causa, 13, 16145 Genova, Italy. e-mail: claudio.castellini@unige.it

A.E. Fiorilla and G. Sandini are with the DIST, University of Genova, viale F. Causa, 13, 16145 Genova, Italy. and the Italian Institute of Technology, via Morego, 30, 16163 Genova, Italy. e-mail: emanuele.fiorilla@iit.it, giulio.sandini@iit.it



(a) (b) (c)

Fig. 1. (a) Touch Bionics's i-LIMB prosthetic hand (reproduced from [1]); (b) the DLR-II mechanical hand; (c) Otto Bock's SensorHand Speed (reproduced from [2]).

In order to control the prosthesis in a more natural way, machine learning can be used to better interpret the standard EMG signals. In the typical case, the patient is asked to imagine, e.g., a pinch grip; the related EMG pattern is then used to obtain a pinch grip with the required force from the prosthesis. A degree of control unknown so far can be thus obtained, improving the patient's life and shortening the training time. We envision *adaptive prosthetics*, a framework where a patient enters a virtuous loop of *reciprocal learning*, whereas so far (s)he has to learn how to control the prosthesis from scratch.

To further improve this loop it would be desirable to *pre-train* the prosthesis with a model which will be then refined and adapted on-line to the patient. In machine learning, this is called *model adaptation*: a system that adapts to a data distribution as it shifts over time. In our proposed method model adaptation works by constraining at each step a new model to be *close* to one of a set of pre-trained models stored in the memory of the prosthesis. The degree of *closeness* and the choice of the pre-trained model to use are done automatically by estimating the generalization power, using the leave-one-out error.

To check whether this idea works we apply it to a set of EMG data collected from 10 healthy subjects. Each subject was asked to grasp a force sensor using three different grips; meanwhile, we recorded the electrical activity of the muscles that are most involved in the hand/wrist movements and the force exerted by the subject on an off-the-shelf force sensor. The experimental results show that our intuition is correct: the proposed adaptation method, on average, shortens the training time and also gives better overall performances, with less training samples. This is true for the classification of the grasp types and for the prediction of the force applied.

978-1-4244-2789-5/09/$25.00 ©2009 IEEE

The paper is structured as follows: after a brief review of related work, we describe our method (Section II) and the EMG database (Section III). Section IV shows the experimental results and lastly Section V contains the conclusions.

A. Related Work

The use of surface forearm EMG to control active hand prostheses dates back to the Fifties and was brought to the market by Otto Bock Orthopaedic Industry, Inc. [6]. EMG works by detecting a muscle's activation potential, a fast oscillating signal whose root-mean-square is non-linearly related to the force exerted by the muscle [7]. Since amputees are usually left with little of their forearm, it has so far been necessary to carefully detect the patient's residual muscles with the strongest activity. These muscles are used, still nowadays, to control one, or at best two degrees-of-freedom. For example, the usual control schema of Otto Bock's SensorHand Speed prosthetic hand maps wrist flexion to hand closing and wrist extension to hand opening.

The situation hasn't changed for a long time because the EMG signal is badly conditioned, being influenced by sweat, muscular fatigue, inter-arm differences and non-hand-related muscular activity (supination/pronation, walking, raising one's arms and so on; see [8] for a survey). Only in the 1990s it became apparent that machine learning could be used to classify hand postures via the EMG. In their seminal work, Bitzer and van der Smagt [9] used a Support Vector Machine (SVM) to robustly classify six different hand postures. Neural networks and LWPR [10] have been used to the same end (see, e.g., [11], [12], [13]) to classify up to 11 hand/finger postures and movements, and to approximate the force involved in the grasp. As long as it is trained for a sufficient time, that is it explores a relevant portion of the input space, a well-employed machine learning method will be able to take into account all of the EMG signal's problems.

As far as we know, there is no EMG database present in the machine learning community, which could serve our purpose. In some of the aforementioned papers, analogous data sets (but most likely smaller than ours) are reported about, but there is no mention of their availability. Regarding model adaptation, several approaches has been proposed. In [14] many of them are compared and benchmarked, however most of them are computationally inefficient because a re-training over new and old data is needed. An approach that does not use re-training, based on SVM has been proposed in [15]. As far as we know this work is the first attempt to use model adaptation in the domain of EMG prosthetics.

II. Mathematical Framework

This section describes our mathematical framework. We first introduce the basic notation (Section II-A), then we present our algorithm for online model adaptation (Section II-B).

A. Background

Assume $\mathbf{x}_i \in \mathbb{R}^m$ is an input vector and $y_i \in \mathbb{R}$ is its associated output. Given a set $\{\mathbf{x}_i, y_i\}_{i=1}^{l}$ of samples drawn from an unknown probability distribution, we want to find a function $f(\mathbf{x})$ such that it determines best the corresponding y for any future sample $\mathbf{x}$. This is a general framework that includes both regression and classification. The problem can be solved in various ways. Here we will use kernel methods and in particular Least-Squares Support Vector Machines (LS-SVM) [16]. In LS-SVM the function $f(\mathbf{x})$ is built as a linear model $\mathbf{w} \cdot \phi(\mathbf{x}) + b$, where $\phi(\cdot)$ is a non-linear function mapping input samples to a high-dimensional (possibly infinite-dimensional) Hilbert space called *feature space*. Rather than being directly specified, the feature space is usually induced by a *kernel function* $K(\mathbf{x}, \mathbf{x}')$ which evaluates the inner product of two samples in the feature space itself, i.e. $K(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x}) \cdot \phi(\mathbf{x}')$. A common kernel function is the Gaussian kernel

$$K(\mathbf{x}, \mathbf{x}') = \exp(-\gamma ||x - x'||^2) \tag{1}$$

that will be used in all our experiments.

The parameters of the linear model, $\mathbf{w}$ and b, are found by minimizing a regularized least-squares loss function [16]. This approach is similar to the well-known formulation of Support Vector Machines (SVMs), the difference being that the loss function is the square loss. While this does not induce a sparse solution, it makes it possible to write the leave-one-out error in closed form and with a negligible additional computational cost [17]. This is known to be approximately an unbiased estimator of the classifier generalization error [18]. This property is useful to find the best parameters for learning (e.g. γ in (1)) and it will be used in our adaptation method. Note that we use the same formulation to solve both regression and classification problems.

B. Model Adaptation

Let us assume we have N pre-trained models stored in memory, trained off-line on data acquired on N different subjects. When the prosthetic hand starts to be used by subject $N+1$, the system begins to acquire new data. Given the differences among the subjects' arms and as well in the placement of the electrodes, these new data will belong to a new probability distribution, in general different from the N previously modeled and stored. Still, as all subjects perform the same grasp types, it is reasonable to expect that the new distribution will be *close* to at least one of those already modeled; then, it should be possible to use one of the pre-trained model as a *starting point* for training using the new data. We expect that, by doing so, learning should be faster than using the new data alone. To solve this problem we generalize the framework for adaptation proposed in [15] for SVMs: the basic idea is to slightly change the regularization term of the SVM cost functional, so that the solution will be "close" to the pre-trained one. The optimization problem is [15]

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w} - \mathbf{w}'\|^2 + C\sum_{i=1}^{l}\xi_i$$
$$\text{subject to} \;\; \xi_i \geq 0, \;\; y_i\mathbf{w} \cdot \phi(\mathbf{x}_i) + b \geq 1 - \xi_i \tag{2}$$

where $\mathbf{w}'$ is a pre-trained model. In order to tune the closeness of $\mathbf{w}$ to $\mathbf{w}'$, we introduce a scaling factor β weighing the pre-trained model; also, we use the square loss and therefore resort to the LS-SVM formulation. This way the leave-one-out error can be evaluated in closed form, enabling automatic tuning of β. The optimization problem reads now like this:

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}-\beta\mathbf{w}'\|^2 + \frac{C}{2}\sum_{i=1}^{l}\xi_i^2$$
$$\text{subject to } \; y_i = \mathbf{w}\cdot\phi(\mathbf{x}_i) + b + \xi_i \qquad (3)$$

and its solution is

$$\mathbf{w} = \beta\mathbf{w}' + \sum_{i=1}^{l}\alpha_i\phi(\mathbf{x}_i), \; \alpha_i \in \mathbb{R} \; . \qquad (4)$$

Hence, the adapted model is given by the sum of the pre-trained model $\mathbf{w}'$ (weighted by β) and a new model $\mathbf{w}$ obtained from the new samples. (Note that when β is 0 we recover the original LS-SVM formulation without any adaptation to previous data.) As far as the leave-one-out error is concerned, we have that

$$\boldsymbol{\alpha} = \boldsymbol{R}(\boldsymbol{Y} - \beta\hat{\boldsymbol{Y}}) \qquad (5)$$

where $\boldsymbol{\alpha}$ is the vector of the α_i's in (4), $\boldsymbol{Y}$ is the vector of the y_i, $\hat{\boldsymbol{Y}}$ is the vector of the predictions of the previous model, and $\boldsymbol{R} = (\boldsymbol{K} + 1/C)^{-1}$ with $\boldsymbol{K}$ denoting the kernel matrix, i.e., $K_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$. Let $\boldsymbol{\alpha}' = \boldsymbol{R}\boldsymbol{Y}$ and $\boldsymbol{\alpha}'' = \boldsymbol{R}\hat{\boldsymbol{Y}}$; from the equation above, and using the same steps in [19], we have that the prediction on sample i, when removed from the training set, is

$$y_i - \frac{\alpha_i'}{R_{ii}} + \beta\frac{\alpha_i''}{R_{ii}} \qquad (6)$$

from which the leave-one-out-error is easily evaluated, according to the required measure of accuracy for the problem at hand. Notice that, in the above formula, β is the only parameter; hence, it is possible to set it optimally in order to minimize the leave-one-out error while at the same time choosing the best pre-trained model for adaptation.

Notice also that the complexity of the algorithm is dominated by the evaluation of the matrix $\boldsymbol{R}$, which must anyway occur while training; thus, the computational complexity of evaluating the leave-one-out error is negligible, if compared to the complexity of training. As a last remark, we underline that the pre-trained model $\mathbf{w}'$ can be obtained by any training algorithm, as far as it can be expressed as a weighted sum of kernel functions. The framework is therefore very general.

III. Database

A. Subjects and setup

We acquired data from ten healthy subjects, two women and eight men, nine right-handed and one left-handed, of an average age of 30.9 ± 8.45 years. The subjects were generally naïve with respect to the recording procedure. We placed on each subject's dominant forearm 7 surface EMG electrodes. The number of electrodes and their positions were chosen, visually and by palpation, according to the medical literature [20]. This procedure allowed us to identify the most relevant flexor and extensor muscles of the forearm, and to record their EMG activity from the spots that should be least affected by signal cross-talk[1]. The chosen locations were:

- on the forearm ventral side: near the wrist, above the *flexor pollicis longus*; centrally, above the *flexor digitorum superficialis*; near the elbow, above the *flexor digitorum profundus*; and near the wrist, above the *flexor digitorum superficialis* again;
- on the forearm dorsal side: near the wrist, above the *extensor pollicis brevis/abductor pollicis longus*; centrally, above the *extensor digitorum communis* and *extensor digiti minimi*.

We employed the electrodes Aurion ZeroWire wireless EMG electrodes [21]. Moreover the subjects were given a FUTEK LMD500 Hand Gripper force sensor [22] in order to measure the force applied by her/his hand during the recording.

We used a standard National Instruments data acquisition board (NI-USB6211) connected to the receiver of the EMG wireless device and to the force sensor, in order to record the sensors' signals and the exerted force. We set the sampling rate of the board at 2kHz, since it is known that the raw EMG relevant bandwidth lies between 15 and 500Hz. See Figure 3 for an example.

B. Data acquisition and pre-processing

We first considered a rest condition, so to define the baseline of the EMG activity. We then proceeded with the data recording: the subject kept her/his arm still and relaxed on a table, and was asked to grasp the force sensor using, in turn, three different grips (Figure 2).

The subject freely repeated each grasping action for 100 seconds, resting for 30 seconds in between grasps. In order to gather more data and diminish the effect of local errors, the whole procedure was repeated twice. As a whole, each subject's recording resulted in about 2.4×10^6 samples.

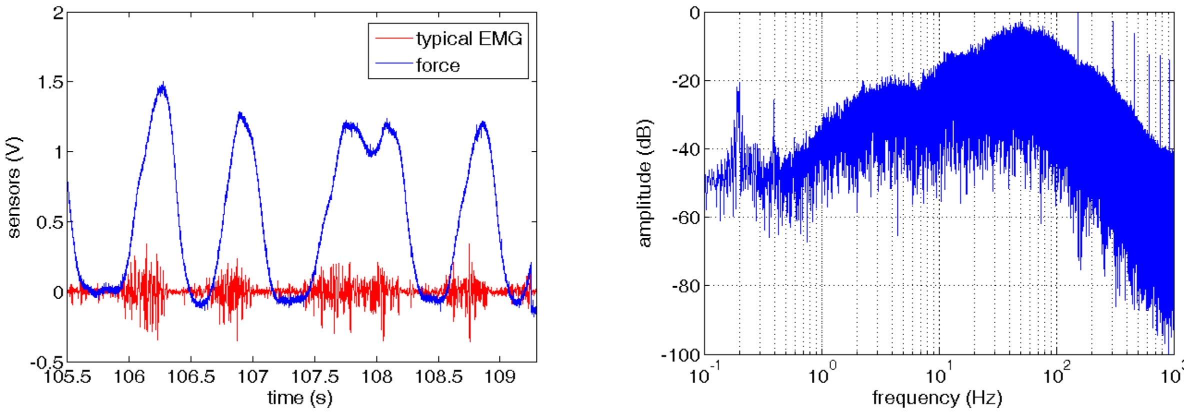
Unlike commercial EMG electrodes, such as, e.g., Otto Bock's MyoBock electrodes [23] that return the on-board computed Root-Mean Square (RMS) of the EMG signal, the electrodes employed here return the "raw" EMG signal. Nevertheless, it is well-known [7], [8] that the force exerted by a muscle is strongly related to the RMS of the EMG signal, rather than to the raw signal. For this reason, in order to have a signal that is as similar as possible to a *control signal*, we decided to evaluate the RMS, electrode by electrode.

For a given mono-variate discrete time-varying signal, the RMS is defined as the mean of the squares of the signal values, evaluated over a certain time-window T_{RMS}. Roughly

[1]But notice that some of the aforementioned muscles are deep into the forearm, so that muscle cross-talk cannot be completely avoided.

(*a*) (*b*) (*c*)

Fig. 2. The three different grips employed in the experiment: (a) index precision grip; (b) other fingers precision grip; (c) power grasp.



(*a*) (*b*)

Fig. 3. (a) typical raw EMG and force signals; (b) frequency diagram of the EMG signal.

speaking, the RMS acts like an envelope extraction plus a low-pass filter, whose cutoff frequency grows smaller as the time-window grows larger (i.e., as T_{RMS} becomes higher). For this reason, high values of T_{RMS} imply an ostensible delay in the resulting signal that is due to *responsiveness* of the synthesized output signal. It becomes slower and slower as the T_{RMS} value increases, since more "samples" are averaged to obtain a significant value. The choice of T_{RMS} is therefore crucial to produce a signal which is maximally related to the force signal, unaffected by high-frequency noise, and with an acceptable lag. However, it must be noted here that the EMG signal, being directly related to the muscle *activation potentials*, happens to *anticipate* the muscle movements[2]. Therefore, in practical applications, it can be considered as acceptable a wider lag than what one would expect. This is useful since it allows us to increase T_{RMS}, if necessary.

We are not aware of any systematic way of setting a good value of T_{RMS} in such a framework. Therefore we found T_{RMS} heuristically, according to some initial experiments.

Figure 3 (panel (a)) shows a few seconds of typical force/EMG behavior: it is apparent that the EMG signal starts oscillating when the force signal starts increasing. It is also quite clear that the amplitude of the envelope of the EMG is related to the force, as indicated in the literature. Panel (b) shows the frequency analysis of the same EMG signal: as one can see, the meaningful bandwidth lies in the interval known from the literature.

This enables us to safely sub-sample the EMG signal after having applied the RMS. Assuming that T_{RMS} is not too small, we subsampled both the EMG and force signal at 25Hz, taking one sample every 80 of the original sequence. This considerably reduced the amount of data to be processed, namely to about 30.000 samples for each subject.

As a last data pre-processing step, we removed from the sample set those samples for which the applied force was lower than a specific threshold, in order to get a clearer representation of the activation potentials. This threshold was chosen in order to remove a minimal fraction of the samples. Of course, we fully retained the samples corresponding to the baseline rest condition. This is why we chose to record this condition before the data acquisition.

IV. Experimental Results

We conducted two different experiments, one to predict the force measured by the force sensor and another one to classify the grasp type.

As already mentioned in Section II-B, our working assumption is to have N pre-trained models stored in memory;

[2] The electromechanical delay (EMD) of a muscle is defined as the interval between the onset of the electrical activity of the muscle (EMG) indicating its activation by the neural system and the onset of the resulting change in the mechanical variable observed. The delays reported range from 25 to 100ms for different muscles and tasks [24].

new data comes from subject $N+1$ and the system starts training, to build the $N+1$'th model. The performance is then evaluated using unseen data from subject $N+1$. To simulate this scenario and to have a reliable estimation of the performance, we use a leave-one-out approach: out of the 10 subjects for which we have data recordings, we train 9 models off-line. These correspond to the N stored models in memory, while data from the remaining subject are used for the adaptive learning of the $N+1$'th model. The training sequences are random subsets from the entire dataset, that is they are taken without considering the order in which they were acquired. This procedure is repeated 10 times, using in turn all the recorded subjects for the adaptive learning of the model.

To assess the performance of the proposed adaptation method we compared it to two baseline methods. The first one, that we call *Prior*, consists in using only the pre-trained models without updating them with the new training data. Therefore we consider only the best performance obtained by one of the 9 pre-trained models, corresponding to the best-case scenario. The second one, *NoAdapt*, is plain LS-SVM using only the new data for training, as it would be in the standard scenario without adaption.

As a measure of performance, for classification we use the standard classification rate; for regression, the performance index is the correlation coefficient evaluated between the predicted force signal and the real one. The choice of the correlation coefficient, as opposed to the more standard Mean-Square Error, is suggested by a practical consideration: when driving a prosthesis, or even a non-prosthetic mechanical hand, we are not interested in the absolute force values desired by the user/subject, since mechanical hands usually cannot apply as much force as human hands do, for obvious safety reasons[3]. We are rather concerned about getting a signal which is *strongly correlated* with the user/subject's will.

To build the pre-trained models we used the standard SVM algorithm. All the parameters to be set during training (C and γ of the gaussian kernel) were chosen by cross-validation. In the following Figures, error bars (when present) denote ± 1 standard deviations with respect to the average values.

We have considered a maximum of 360 training samples and the final performance, averaged across the subjects, for *NoAdapt* is 89.2% and 0.811, respectively for classification and regression. With our adaptation method we get instead 90.2% and 0.85. Figure 4 shows in more details the difference in classification (panel (a)) and regression (panel (b)) performance obtained by our method with respect to *NoAdapt*. As one can see, adaptation uniformly obtains a better performance, with the exception of classification when the number of samples is below 150: in that case a slight loss of about 1% can appear. In the worst cases, the performance of *NoAdapt* is re-obtained. Notice also that standard deviations are rather large when training is done on too few samples. This is due to the high variance of the leave-one-out error technique when too few training samples are considered.

Depending on the subject, the improvement can be quite large: up to about 15% higher rate for classification and about 0.15 points stronger correlation for regression. In average, for classification, the gain is almost 5% when there are only 30 training samples. It settles to around 1% with smaller standard deviation, as the number of training samples increases. For regression, we have a correlation coefficient about 0.04 points uniformly stronger in average.

To get a more detailed idea of the results obtained, consider now Figure 5, concerning the classification experiment. Panel (a) shows the performance obtained on the best-case subject, that is, a subject for whom a very good match has been found among the pre-trained models, while panel (b) shows the performance for the worst-case subject. In the best case the gain is about 3% after 360 samples, while in the worst case we basically re-obtain the performance of *NoAdapt*, as soon as enough samples from the new distribution are considered. Essentially, our method improves things if a good match can be found, and does no harm if none exists. Similar observations can be done for the regression task (Figure 6). In the best case, the correlation is about 0.06 points uniformly stronger, whereas in the worst case *NoAdapt*'s performance is obtained. Note also the performance of *Prior*, constantly inferior to *NoAdapt* and our method.

The worst-case subjects represent the paradigmatic case of no previous models matching the current distribution; as a consequence, the parameter β was automatically set to a very small value. In this case, there is essentially no transfer of prior knowledge. But it is reasonable to claim that the overall performance of the method would increase along with the number of stored models, since this would mean a larger probability of finding a matching pre-trained model.

In the long run, a large database of pre-trained models, possibly categorized in order to avoid too hard a computational burden, would definitely help getting uniformly better performances.

V. CONCLUSIONS

The model adaptation method presented in this paper stems from a problem in adaptive hand prosthetics, namely: is it possible to help a patient to learn to use a dexterous hand prosthesis by exploiting the common features found in models trained upon other patients? The answer, at least as far as healthy subjects are concerned, is yes: we have hereby presented a novel method for model adaptation in machine learning, using Least-Squares SVMs; the idea is to build a SVM solution which is *close* to one of a set of pre-stored models. The choice of which model to use among the pre-trained ones, as well as the parameter β, determining the degree of closeness to start the training from, are completely automatic, as we use an estimation of the generalization error.

We tested our method on a database built with EMG and force data from 10 healthy subjects, trying to improve the

[3]Or, e.g., in teleoperation scenarios, they could be able to apply *much more* force than a human hand can.

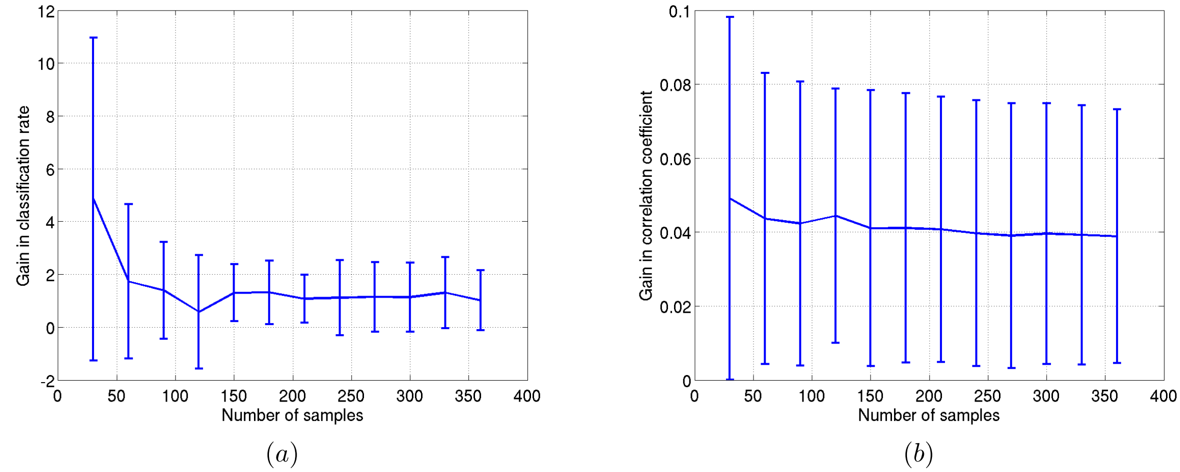


Fig. 4. Classification (a) and regression (b) performance difference obtained by our method with respect to *NoAdapt*.

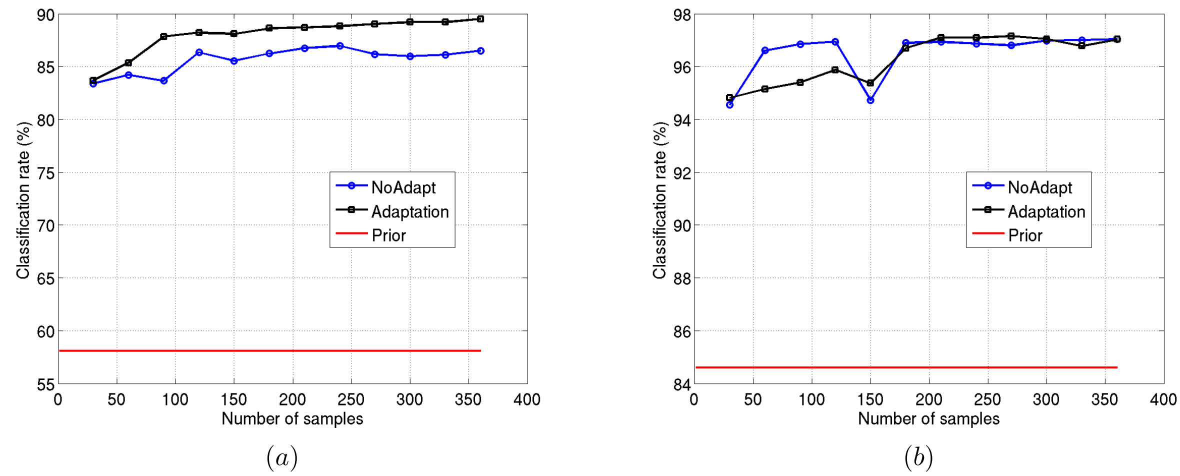


Fig. 5. Classification: (a) classification rate gain of the adapted model compared to *NoAdapt* and *Prior* on the best-case subject; (b) classification rate gain for the worst-case subject.

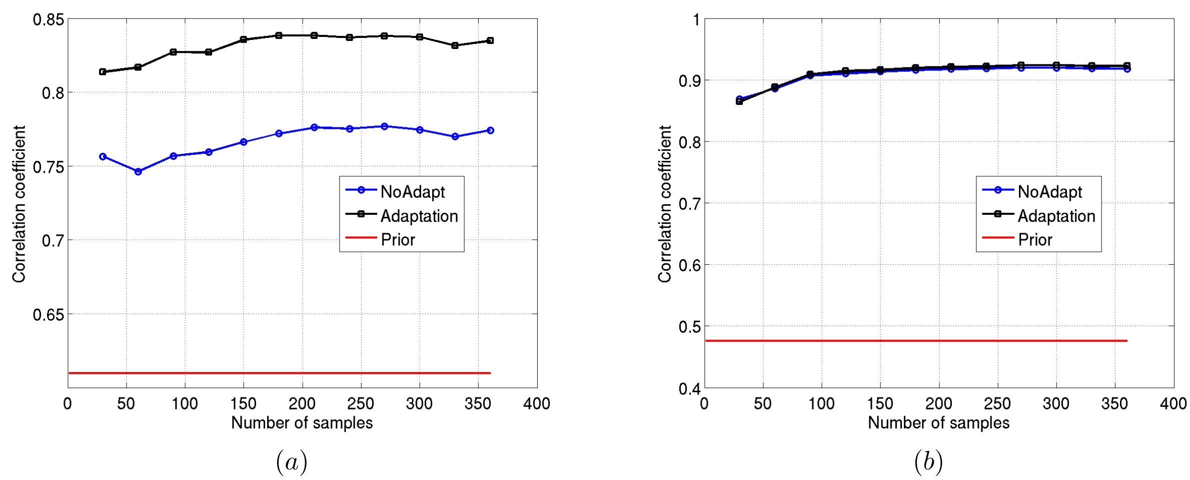


Fig. 6. Regression: (a) correlation strength gain of the adapted model compared to *NoAdapt* and *Prior* for the best-case subject; (b) correlation gain for the worst-case subject.

training times and asymptotic performance of one subject by pre-training on other subjects. The outcome of the experiment is positive: our method gains consistently both in the classification and regression tasks in the best and average cases, and it resorts to the non-adaptive performance in the worst.

Therefore, it is apparent that a large amount of knowledge stored in LS-SVM models is common to all subjects, which is obviously due to the analogies among the tasks performed by the subjects, as well as to the anatomical similarities among the arms and the careful positioning of the electrodes on the subjects' forearms. A further interesting point is that, almost uniformly, models obtained by adaptation from a pre-trained model obtain a *better* performance than those trained from scratch. This result is somehow surprising, although very encouraging, and subject of future research.

Notice that we present no results on a real prosthetic/robotic hand so far — this is subject of immediate-future research. We successfully applied a similar system to the DLR-II mechanical hand (see [11], [12]), and since the accuracy of the system presented here is analogous to that of the one therein, there is no reason why the results presented here should not apply as well in the practical case. One interesting possibility is that of using this system to speed up the adaptation of an already existing dexterous hand prostheses, such as, e.g., Touch Bionics's i-LIMB [1] prosthetic hand, as already mentioned in the introduction.

Lastly, let us consider the fact that, most likely, the overall performance of the method will increase when more subjects are available, since this would mean a larger probability of finding a matching pre-trained model. In a clinical setting, this means that after an experimental phase, adaptive prostheses employing this method could actually be built. It remains, of course, to discover whether this idea can be transferred to amputees: amputations are, obviously, non-controlled, traumatic events (except in some cases), and therefore stumps exhibit much more variability than healthy forearms. This is the subject of ongoing as well as future research.

ACKNOWLEDGMENTS

This work is partially supported by the project NEURObotics, FP6-IST-001917 (CC) and the project DIRAC, FP6-IST-027787 (FO,BC). We thank Thierry Pozzo and Marco Jacono for their support during the experiment and Elisa Ricci for the useful discussions on the paper.

REFERENCES

[1] "The i-LIMB hand prosthesis," 2007. [Online]. Available: http://www.touchbionics.com

[2] "The Otto Bock SensorHand hand prosthesis," 2008. [Online]. Available: http://www.ottobockus.com/products/upper_limb_prosthetics/myoelectric_hands_sensorhand.asp

[3] H. Huang, L. Jiang, D. Zhao, J. Zhao, H. Cai, H. Liu, P. Meusel, B. Willberg, and G. Hirzinger, "The development on a new biomechatronic prosthetic hand based on under-actuated mechanism," in *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 3791–3796.

[4] "The CyberHand project," 2007. [Online]. Available: http://www.cyberhand.org

[5] "The SmartHand project," 2007. [Online]. Available: http://www.elmat.lth.se/~smarthand

[6] J. A. B. Wilson, "History of amputation surgery and prosthetics," in *Atlas of Limb Prosthetics: Surgical, Prosthetic, and Rehabilitation Principles*, 2nd ed., H. K. Bowler and J. W. Michael, Eds. C. V. Mosby, 1981, pp. 3—23, available at http://www.oandplibrary.org/alp/chap01-01.asp.

[7] C. J. De Luca, "The use of surface electromyography in biomechanics," 1997. [Online]. Available: http://www.health.uottawa.ca/biomech/courses/apa4311/biomec~1.htm

[8] M. Zecca, S. Micera, M. C. Carrozza, and P. Dario, "Control of multifunctional prosthetic hands by processing the electromyographic signal," *Critical Reviews in Biomedical Engineering*, vol. 30, no. 4–6, pp. 459–485, 2002.

[9] S. Bitzer and P. van der Smagt, "Learning EMG control of a robotic hand: Towards active prostheses," in *Proceedings of ICRA, International Conference on Robotics and Automation, Orlando, Florida, USA*, may 2006, pp. 2819–2823.

[10] S. Vijayakumar, A. D'Souza, and S. Schaal, "Incremental online learning in high dimensions," *Neural Computation*, vol. 17, pp. 2602–2634, 2005.

[11] C. Castellini, P. van der Smagt, G. Sandini, and G. Hirzinger, "Surface EMG for force control of mechanical hands," in *Proceedings of ICRA-08 - International Conference on Robotics and Automation*, 2008, to appear.

[12] C. Castellini and P. van der Smagt, "Surface EMG in advanced hand prosthetics," *Biological Cybernetics*, 2008, to appear.

[13] F. C. P. Sebelius, B. N. Rosen, and G. N. Lundborg, "Refined myoelectric control in below-elbow amputees using artificial neural networks and a data glove," *J Hand Surg [Am]*, vol. 30, no. 4, pp. 780–789, Jul 2005.

[14] H. Daumé III, "Frustratingly easy domain adaptation," in *Conference of the Association for Computational Linguistics (ACL)*, Prague, Czech Republic, 2007.

[15] J. Yang, R. Yan, and A. G. Hauptmann, "Adapting SVM classifiers to data with shifted distributions," in *ICDMW '07: Proceedings of the Seventh IEEE International Conference on Data Mining Workshops*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 69–76.

[16] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines (and Other Kernel-Based Learning Methods)*. Cambridge University Press, 2000.

[17] R. Rifkin and R. Lippert, "Notes on regularized least-squares," Massachusetts Institute of Technology, Cambridge, MA, Tech. Rep. CBCL Paper #268/AI Technical Report #2007-019, May 2007.

[18] A. Luntz and V. Brailovsky, "On estimation of characters obtained in statistical procedure of recognition (in russian)," *Techicheskaya Kibernetica*, vol. 3, 1969.

[19] G. C. Cawley, "Leave-one-out cross-validation based model selection criteria for weighted LS-SVMs," in *In Proceedings of the International Joint Conference on Neural Networks (IJCNN-2006)*, Vancouver, BC, Canada, 2006, pp. 1661–1668.

[20] F. P. Kendall, E. K. McCreary, P. G. Provance, M. M. Rodgers, and W. Romani, *Muscles: Testing and Function, with Posture and Pain*. 530 Walnut St.Philadelphia, PA 19106-3621: Lippincott Williams & Wilkins, 2005.

[21] "Aurion zerowire EMG electrodes." [Online]. Available: http://www.aurion.it/

[22] "Futek LMD500 medical load cell (hand)." [Online]. Available: http://www.futek.com/product.aspx?stock=FSH00125

[23] "Otto Bock MYOBOCK 13E200=50 surface emg electrodes," 2008. [Online]. Available: http://www.ottobockus.com/products/upper_limb_prosthetics/myoelectric_hands_myobockr.asp

[24] W. Wolf, C. Staude, and U. Appel, "Enhanced onset detection accuracy "reduces" the electromechanical delay of distal muscles," in *Proc. 16th Annual International Conference of the IEEE Engineering Advances: New Opportunities for Biomedical Engineers Engineering in Medicine and Biology Society*, 3–6 Nov. 1994, pp. 392–393.