CONTRIBUTION TO STATISTICAL ACOUSTIC-TO-EMA MAPPING

Asterios Toutios[1] *and Konstantinos Margaritis*[2]

[1] LORIA, Nancy, France
phone: + 33 383 59 30, fax: + 33 383 55 25 73, email: asterios.toutios@loria.fr
[2] Department of Applied Informatics, University of Macedonia, Thessaloniki, Greece
phone: + 30 2310 891 891, fax: + 30 2310 891846 , email: kmarg@uom.gr

ABSTRACT

The availability of large corpora of parallel acoustic and articulatory data has enabled the use of statistical, data-driven, methods in the context of the speech inversion problem. This paper explores the use of Support Vector Regression for the mapping from acoustic parameters to electromagnetic articulograph trajectories and compares the outcoming results against those presented in other studies on the same problem and dataset.

1. INTRODUCTION

Acoustic-to-articulatory speech inversion, that is, the recovery of articulatory information from the corresponding acoustic speech signal is a problem that has drawn considerable attention in the speech processing community for several reasons. A successful solution may prove beneficial for automatic speech recognition and synthesis, be of great potential interest for phonetic theory and speech science, offer articulatory feedbacks for purposes of speech therapy and language acquisition, and enable cheap visualizations of speech in communication and entertainment applications. The problem is challenging, due to the nonlinearity and nonuniqueness of the mapping.

Older studies on speech inversion [1] usually relied on articulatory synthesis models, that were built using sparse data and a large degree of intuition regarding the process of speech production. However, the recent development and improvement of articulatory data acquisition techniques, like Electromagnetic Articulography (EMA) [2], and the resulting availablility of large amounts of acoustic and articulatory data recorded in parallel, has created a new option: the training of statistical learning functions to map acoustic onto articulatory information.

In this paper, the inversion mapping from acoustic parameters to EMA information is addressed. As a mapping method, Support Vector Regression (SVR), a relatively new nonlinear method which has been shown to produce state of the art results for several other supervised regression learning problems [3], is proposed. The mapping addresses only the static components of the EMA parameters; that is, no attempt is done to estimate and incorporate dynamical features of the EMA data. Also, as opposed to several other works on speech inversion (e.g. [4]), no constraints on phonetic information are used.

This system is compared against those presented by Richmond et al. [5] (also Richmond [6]), Toda et al. [7] and Richmond [8]. It is demonstrated that the system performs better than the corresponding baseline systems (which address exactly the same problem of estimating static components of the EMA data) and comparably to the final systems (which introduce further constraints to the mapping). The main reason for choosing these particular works to compare with is that they use the same dataset; namely, data from the MOCHA database.

It should be noted beforehand that the method presented herein does not explicitly deal with the nonuniqueness property of the speech inversion problem. In contrast with other speech inversion methods that produce several articulatory hypotheses for a single speech segment (e.g [9]), the method presented here leads to a single estimation of the articulatory state. An implicit assumption we make is that the articulatory strategy employed by each speaker in the MOCHA database may not change significantly during the recording session.

The rest of the paper is organized as follows: Section 2 briefly describes Support Vector Regression, in particular the ε-SVR algorithm. Section 3 describes the data and their processing in order to derive input-output vectors suitable for the regression algorithm. Section 4 presents results and compares them against those found in the aforementioned studies. Section 5 presents our conclusions.

2. SUPPORT VECTOR REGRESSION

Based on n real d-dimensional training input vectors $\mathbf{x}_i \in R^d, i = 1,\ldots,n$, and associated real output scalar values $y_i \in R, i = 1,\ldots,n$, the basic ε-SVR algorithm [3] seeks to estimate a linear function

$$f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b \tag{1}$$

(where $\mathbf{w}$ is a d-dimensional real vector, b is a real scalar, and $\langle .,. \rangle$ denotes the inner product), such that for previously unseen data $(\mathbf{x}, y) \in R^d \times R$, generated from the same underlying process as the training data, the value $f(\mathbf{x})$ approximates the value y as precisely as possible. In other words, the function f should be able to generalize well to previously unseen data, as long as they apply to the same (unknown) probability distribution $P(\mathbf{x}, y)$ as the training data. In order to achieve this, the ε-SVR algorithm attempts to minimize the quantity

$$\frac{1}{2}\|\mathbf{w}\|^2 + C \cdot \frac{1}{n} \sum_{i=1}^{n} |y_i - f(\mathbf{x}_i)|_\varepsilon \tag{2}$$

where

$$|y_i - f(\mathbf{x}_i)|_\varepsilon = \max\{0, |y_i - f(\mathbf{x}_i)| - \varepsilon\} \tag{3}$$

is the so-called ε-insensitive loss function, with ε being a small positive real scalar, while C is a positive real scalar and $\|.\|$ denotes the norm. The parameter C determines the trade-off between the regularization factor $\|\mathbf{w}\|^2$ and the mean training error $\frac{1}{n}\sum_{i=1}^{n} |y_i - f(\mathbf{x}_i)|_\varepsilon$.

It can be shown [10] that the minimization of (2) is equivalent to the following quadratic optimization problem:

$$\begin{gathered}\text{maximize} \\ -\varepsilon\sum_{i=1}^{n}(a_i^*+a_i)+\sum_{i=1}^{n}(a_i^*-a_i)y_i \\ -\frac{1}{2}\sum_{i,j=1}^{n}(a_i^*-a_i)(a_j^*-a_j)\left\langle \mathbf{x}_i,\mathbf{x}_j\right\rangle \qquad (4) \\ \text{subject to} \\ 0\le a_i,a_i^*\le\frac{C}{n},i=1,\ldots,n \text{ and } \sum_{i=1}^{n}(a_i^*-a_i)=0,\end{gathered}$$

where the a,a^* are Lagrange multipliers. The estimated function is then

$$f(\mathbf{x})=\sum_{i=1}^{n}(a_i^*-a_i)\left\langle \mathbf{x}_i,\mathbf{x}\right\rangle+b \qquad (5)$$

where a,a^* are the solutions to the optimization problem (4) and b is easily calculated from the KKT conditions for the problem [10]. There are three cases for the optimal values of a,a^*, for every i: it will be either $a_i=0,a_i^*\neq 0$, or $a_i\neq 0,a_i^*=0$ or $a_i=a_i^*=0$. For the first two cases, the corresponding training input vectors are called the support vectors. Apparently, the estimated function (5) depends only on these. Usually the support vectors are only a fraction of the training input vectors, and so, the solution that the ε-SVR algorithm leads to is sparse on the training data.

As presented so far, the ε-SVR algorithm leads to a linear regression function. It is extended to the nonlinear case with the introduction of a nonlinear mapping $\mathbf{x}\mapsto\Phi(\mathbf{x})$ of the input vectors from their original space to a new one, called the feature space. Practically, this is achieved by substituting the inner products in Equations (4) and (5), with a kernel function $k(\mathbf{x},\mathbf{x}')$.

A usual choice for the kernel function is the gaussian kernel

$$k(\mathbf{x},\mathbf{x}')=\exp(-\gamma\|\mathbf{x}-\mathbf{x}'\|^2), \qquad (6)$$

where the parameter γ is to be selected by the user (and so are the parameters C and ε in the optimization problem).

3. DATA DESCRIPTION AND PROCESSING

The MOCHA database [11] includes four data streams recorded concurrently: the acoustic waveform (that is later labeled at the phonemic level), laryngograph, electropalatograph and electromagnetic articulograph data. The speakers are recorded reading a set of 460 British TIMIT-style sentences, which are designed to provide phonetically diverse material and capture with good coverage the connected speech processes in English. The original plan was that the database would feature up to 40 speakers with a variety of regional accents, but at the time of conducting the experiments presented in this paper only from two speakers only where checked and available.

MOCHA includes electromagnetic articulography (EMA) information for the coils shown in Figure 1. The two coils at the bridge of the nose and the upper incisors are used for the normalization of the data from the rest. Seven coils, located at the lower incisors (*li*), upper lip (*ul*), lower lip (*ll*), tongue tip (*tt*), tongue blade (*tb*), tongue dorsum (*td*) and velum (*v*), offer useful location information, namely trajectories of the projections of their position on two axes on the midsagittal plane: one with direction from the front to the back of the head (x-axis) and one with direction from the bottom to the top of the head (y-axis). In the rest of this paper, the information flows from individual coils on individual axes will be referred to as *EMA channels* and will be designated with the initial letters of the corresponding articulator, using the the axis name as a subscript; e.g. channel li_x will refer to the projection of the position of the coil placed on the lower incisors on the horizontal axis.

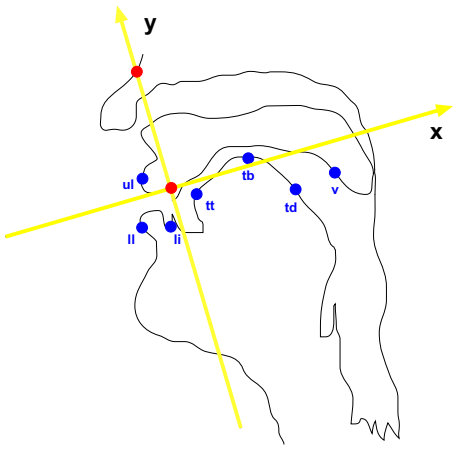

Figure 1: EMA sensor coils in the MOCHA database.

We process these data in a similar way as described in [6]. First, based on the label files, all EMA data corresponding to silent parts from the beginning and end of the utterances are omitted. During silent stretches the articulators can possibly take any configuration, something that could pose serious difficulties to the task at hand. The EMA trajectories are then resampled from 500 Hz to 200 Hz. Since the articulators move relatively slowly, crucial information is not lost. At the same time the trajectories are smoothed, using a low-pass filter in order to lessen the effect of measurement noise.

The mean values of the EMA trajectories calculated for every utterance vary considerably during the recording process. There are two kinds of variation: rapid changes, due to the phonemic content of each utterance, and slowly moving trends, mainly due to the fact that the subject's articulation adapts in certain ways during the recording session [6]. It is useful to remove from the EMA data the second type of variation, while keeping the first, which is achieved by subtracting a low-passed filtered version of the channel means from the EMA data.

Finally, the values of the channels are centered at zero and scaled by four times their standard deviations so that their vast majority falls in the interval $(-1,1)$ (this is a detail relevant to the ε-SVR software implementation used).

Regarding the acoustic speech signal silent parts from the beginning and end of the utterances are again omitted. Perceptual Linear Predictive analysis [12] is performed on the acoustic signal with the HTK Toolkit [13], using a Hamming window of 16ms (256 points – the speech signal is sampled at 16 kHz) with a shift of 5ms (to match the 200 Hz sampling rate of the EMA trajectories). 12 cepliftered MF-PLPs [14]

plus the logarithmic energy of the signal comprise the vector of parameters extracted from every speech frame. Those parameters are then normalized across the whole dataset so that they have zero mean and unity standard deviation.

Input vectors spanning over a large number of acoustic frames are constructed. These vectors include the acoustic parameters of 17 frames: the frame in question, plus 8 previous ones, plus 8 following ones. The time shift between adjacent frames for this construction is 10ms (that is, one in two of the previously derived vectors of parameters are used for this construction). Thus, every 221-dimensional input vector includes information corresponding to roughly 160 ms of speech. Small scale experiments indicated that this was an optimal construction for the task at hand.

4. EXPERIMENTS AND RESULTS

Considering every EMA channel as a separate, independent case, the problem of mapping the acoustic vectors derived from the speech signal onto EMA information becomes a series of fourteen distinct regression problems. The ε-SVR algorithm is called upon in order to solve them.

First, only data from the fsew0 speaker of the MOCHA database, a female with a southern English accent, are considered. Out of the 460 utterances, 368 are chosen to constitute the training set. These correspond to 198,730 input-output examples for every EMA channel. In order to reduce training times, a smaller practical training set (39,746 examples) is constructed by selecting the first out of every five consecutive candidate training examples (in a way, an amount of redundancy present in the information among neighboring input-output patterns is assumed). This latter set is actually used for training the ε-SVR algorithm.

46 utterances (25,022 examples) constitute the test set, and 46 are put aside. Care is taken, so that the utterances selected for each set correspond exactly with the ones used by Richmond [6], in order to have a direct comparison between the approaches (the 46 sentences that are put aside actually correspond to Richmond's development set).

The gaussian kernel is used with the ε-SVR algorithm, with $\gamma = 1/221$ (221 is the dimensionality of the input vectors). The other parameters of the algorithm are chosen as $C = 0.5$, $\varepsilon = 0.05$. Again, small scale experiments indicated that these are near optimal choices ("near" meaning that a deeper search would not lead to significant improvement of the results). The LibSVM software [15] is employed for the implementation of the method.

At testing, after both the actual values of the channels y_i and the corresponding values of the estimate function $f(\mathbf{x}_i)$ are scaled back by multiplying by four times the standard deviation of the corresponding channel, the RMS error over the whole test set is calculated as:

$$E_{RMS} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(f(\mathbf{x}_i) - y_i\right)^2} \tag{7}$$

where m is the number of examples in the test set. The RMS error measures the overall distance between the original and estimated trajectories. The Pearson correlation, calculated as:

$$r = \frac{\sum_{i=1}^{m}\left(f(\mathbf{x}_i) - \overline{f(\mathbf{x})}\right)(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{m}\left(f(\mathbf{x}_i) - \overline{f(\mathbf{x})}\right)^2 \sum_{i=1}^{m}(y_i - \overline{y})^2}} \tag{8}$$

channel	E_{RMS}	r
li_x	0.87	0.603
li_y	1.13	0.820
ul_x	0.93	0.621
ul_y	1.09	0.744
ll_x	1.17	0.637
ll_y	2.34	0.816
tt_x	2.21	0.834
tt_y	2.24	0.875
tb_x	2.07	0.831
tb_y	1.88	0.871
td_x	1.94	0.812
td_y	1.97	0.798
v_x	0.36	0.847
v_y	0.34	0.834
average	1.47	0.782

Table 1: Cumulative results on the test set for speaker fsew0. RMS values are measured in millimeters.

channel	E_{RMS}	r
li_x	0.49	0.711
li_y	0.88	0.821
ul_x	0.63	0.642
ul_y	1.08	0.755
ll_x	1.06	0.747
ll_y	1.62	0.849
tt_x	2.48	0.794
tt_y	2.84	0.834
tb_x	2.19	0.788
tb_y	1.91	0.858
td_x	2.13	0.768
td_y	1.85	0.824
v_x	0.44	0.766
v_y	0.97	0.674
average	1.47	0.774

Table 2: Cumulative results on the test set for speaker msak0. RMS values are measured in millimeters.

(where overlines denote mean values over the test set), quantifies the similarity in shape and synchrony between the trajectories.

These results are presented in Table 1. Figures 2 and 3 show the actual and estimated trajectories for a single utterance from the test set. The figures include corresponding phonemes (with IPA symbols) based the phonemic transcriptions provided with the MOCHA database. Care should be taken however: the phonemic labeling of MOCHA is the result of an automatic alignment process and considered prone to errors.

The process described so far is repeated using, both in training and testing, data from the msak0 speaker of the MOCHA database, a male with a northern English accent. Table 2 presents these results.

The results in Tables 1 and 2 may be compared against those reported in previous attempts on the mapping using the same dataset. Richmond et al. in [5] (also Richmond [6]) used a Multilayer Perceptron to map from filterbank coefficients to EMA information for the fsew0 speaker. They sampled the EMA trajectories at 100 Hz, as opposed to the 200 Hz sampling rate used in this paper. They reported an aver-

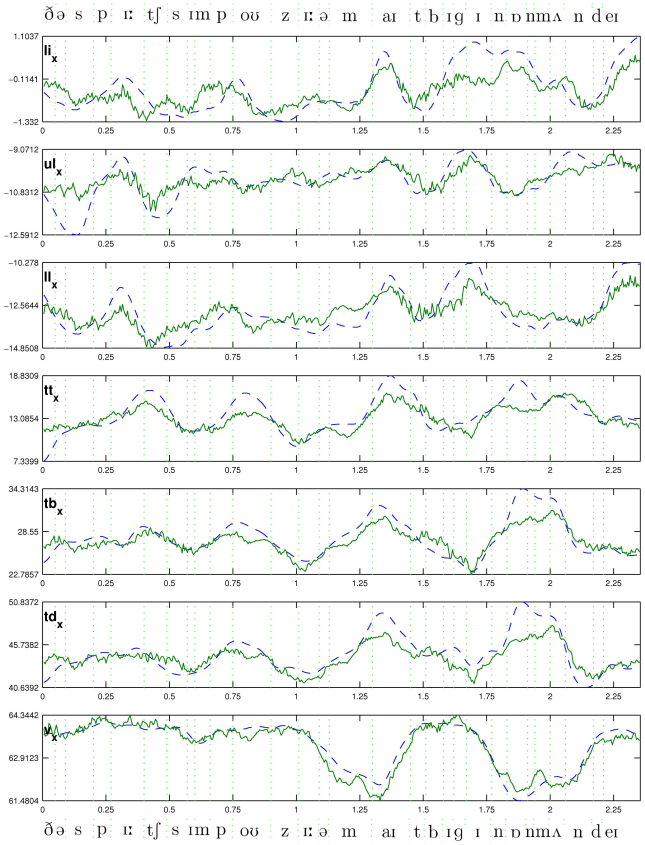

Figure 2: Actual (dashed lines) and estimated (solid lines) EMA trajectories (X-coordinates) when speaker fsew0 utters the phrase: "The speech symposium might begin on Monday". Vertical axes: value in millimeters of X-coordinate of coil in question using the mean position of coil on upper incisors as origin. Horizontal axes: time in seconds. Channel names are shown in the top-left corner of the boxes.

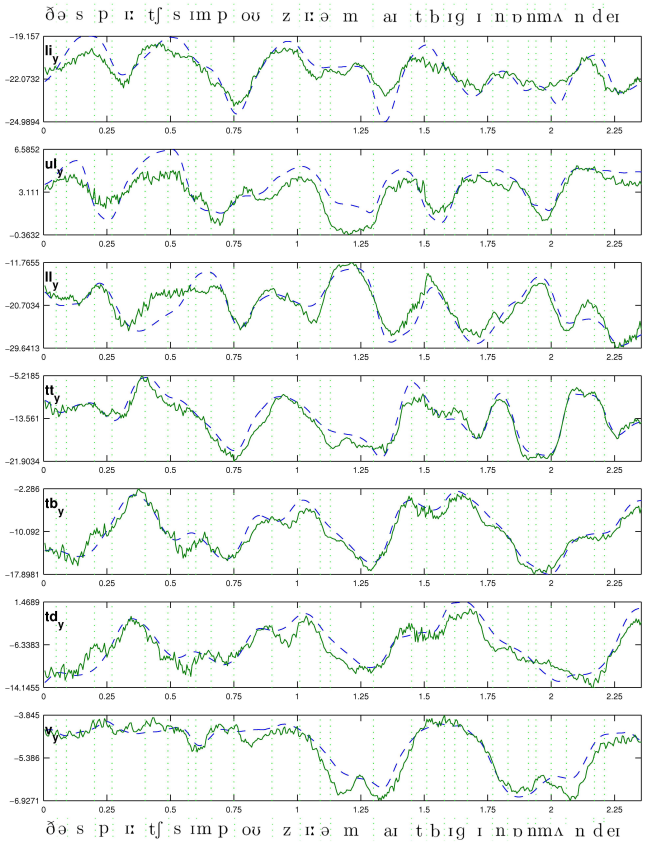

Figure 3: Actual (dashed lines) and estimated (solid lines) EMA trajectories (Y-coordinates) when speaker fsew0 utters the phrase: "The speech symposium might begin on Monday". Vertical axes: value in millimeters of Y-coordinate of coil in question using the mean position of coil on upper incisors as origin. Horizontal axes: time in seconds. Channel names are shown in the top-left corner of the boxes.

age (over the fourteen channels) RMS error of 1.62 mm and an average Pearson correlation of 0.739. As a second step of their approach they employed Mixture Density Networks reporting a relative increase of 9.3% on a measured average likelihood score over the MLP. As already mentioned, the test set used in this paper consists of the exact same utterances as in Richmond et al.

Toda et al. [7] also sampled the EMA trajectories at 100 Hz. As their baseline experiment they used a Gaussian Mixture Model based mapping algorithm in order to map mel-cepstral coefficients to the static EMA features. They reported average RMS errors of 1.63 mm for speaker fsew0 and 1.54 mm for speaker msak0. They did not choose a specific test set; they rather reported cross validation results.

Both Richmond (in [6]) and Toda et. al smoothed the estimated trajectories using a series of low-pass filters with incremental cutoff frequencies and, independently for each EMA channel, selected the cutoff frequency that minimized the RMS error. This lead Richmond to an average error of 1.57 mm (average correlation was 0.758) and Toda et. al to 1.49 mm, for the fsew0 case. Application of the same strategy to the results of the ε-SVR algorithm used in this paper leads to the results presented in Table 3, for speaker fsew0.

Toda et al. went on to incorporate dynamic features in their approach, via a parameter generation algorithm based on Maximum Likelihood Estimation. They achieved average RMS errors of 1.45 mm before smoothing, and 1.44 mm after smoothing on speaker fsew0.

Quite recently, Richmond [8] reported the introduction of dynamic features to his MDN setup. He reported results for EMA channels tt_x (RMS error 2.22, correlation 0.84), tt_y (2.31, 0.87), tb_x (2.13, 0.82), tb_y (1.93, 0.86), td_x (1.91, 0.82) and td_y (1.92, 0.81).

5. CONCLUSION

This paper demonstrated that the application of Support Vector Regression to the task of estimating EMA trajectories from the speech signal in a speaker-dependent setup is promising, based on the comparison of the results against those achieved in other studies in the literature using other statistical learning methods. The method considered only static features of the EMA information. The results presented might be improved with a further introduction of dynamic or phonetic constraints.

In our experimental setup, the training parameters (C, ε,

channel	cutoff	E_{RMS}	r
li_x	2.0	0.86 (2.01%)	0.622 (3.16%)
li_y	3.7	1.11 (1.37%)	0.825 (0.66%)
ul_x	1.7	0.90 (2.86%)	0.651 (4.86%)
ul_y	2.6	1.07 (2.62%)	0.759 (1.98%)
ll_x	1.9	1.15 (2.16%)	0.657 (3.08%)
ll_y	3.7	2.31 (1.36%)	0.822 (0.71%)
tt_x	2.9	2.17 (2.03%)	0.842 (1.07%)
tt_y	4.3	2.20 (1.80%)	0.881 (0.65%)
tb_x	3.2	2.03 (1.80%)	0.839 (0.94%)
tb_y	3.2	1.84 (2.29%)	0.877 (0.74%)
td_x	3.3	1.91 (1.70%)	0.820 (0.95%)
td_y	2.8	1.92 (2.36%)	0.808 (1.32%)
v_x	9.2	0.36 (0.63%)	0.849 (0.30%)
v_y	3.8	0.34 (1.05%)	0.838 (0.49%)
average		1.44 (1.86%)	0.792 (1.49%)

Table 3: Cumulative results on the test set for speaker fsew0, after smoothing. The "cutoff" column shows the cutoff frequency of the "best" low-pass filter in Hz. The numbers in the parentheses are the relative improvements of the results over the ones presented in Table 1.

γ and the size of the input context window) were chosen so as to optimize performance on the fourteen EMA channels in total. It might be the case (and more recent experiments indicate) that a channel-specific optimization of these parameters improves channel-specific results. Nonetheless, such a behavior should be thoroughly tested (perhaps using data from more speakers) to check whether it is systematic or not.

Another point (which might contardict the previous one) is that we believe that future attempts on the problem should take more explicitly into account its temporal and spatial structure. What we (and to our knowledge, the other methods presented here) do is treat the problem as a series of relatively independent static mapping problems. The concatenation of input vectors or the a posteriori introduction of dynamical constraints does not, to our belief, fully account for temporal structure. The spatial inter-correllations among articulatroy trajectories are not exploited. Learning problems involving structured spaces is the subject of many recent studies in the machine learning field (e.g. [16]).

Yet, the problem stated as: "How can a set of intercorrelated time-series be predicted from another set of (intercorrelated) time-series?" is open. A definitive answer to it may prove beneficial not only for the speech inversion field but also to scientific areas extending far beyond speech processing.

Acknowledgment

This work was supported by the Greek research program "HRAKLEITOS", which is co-funded by the European Social Fund (80%) and National Resources (20%).

REFERENCES

[1] J. Schroeter and M. M. Sondhi. Techniques for estimating vocal tract shapes from the speech signal. *IEEE Transactions on Speech and Signal Processing*, 2(1):133–150, January 1994.

[2] P. W. Schönle, K. Gräbe, P. Wenig, J. Höhne, J. Schrader, and B. Conrad. Electromagnetic articulography: Use of alternating magnetic fields for tracking movements of multiple points inside and outside the vocal tract. *Brain and Language*, 31:26–35, 1987.

[3] A. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, 14(3):199–222, August 2004.

[4] S. V. Dusan. *Statistical Estimation of Articulatory Trajectories from the Speech Signal Using Dynamical and Phonological Constraints*. PhD thesis, University of Waterloo, Ontario, Canada, 2000.

[5] K. Richmond, S. King, and P. Taylor. Modeling the uncertainty in recovering articulation from acoustics. *Computer Speech and Language*, 17:153–172, 2003.

[6] Korin Richmond. *Estimating Articulatory Parameters from the Speech Signal*. PhD thesis, The Center for Speech Technology Research, Edinburgh, UK, 2002.

[7] T. Toda, A. W. Black, and K. Tokuda. Acoustic-to-articulatory mapping with gaussian mixture model. In *INTERSPEECH 2004 - ICSLP, 8th International Conference on Spoken Language Processing*, Jeju Island, Korea, 2004.

[8] K. Richmond. A trajectory mixture density network for the acoustic-articulatory inversion mapping. In *INTERSPEECH 2006 - ICSLP, 9th International Conference on Spoken Language Processing*, Pittsburgh, USA, 2006.

[9] S. Ouni and Y. Laprie. Modeling the articulatory space using a hypercube codebook for acoustic-to-articulatory inversion. *Journal of the Acoustical Society of America*, 118(1):444–460, July 2005.

[10] B. Schölkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, December 2001.

[11] A. A. Wrench and W. J. Hardcastle. A multichannel articulatory database and its application for automatic speech recognition. In *5th Seminar on Speech Production: Models and Data*, pages 305–308, Kloster Seeon, Bavaria, 2000.

[12] H. Hermansky. Perceptual linear predictive (PLP) analysis of speech. *Journal of the Acoustical Society of America*, 87(4):1738–1752, April 1990.

[13] S. Young, D. Ollason, V. Valtchev, and P. Woodland. *The HTK Book (for HTK Version 3.3)*. Cambridge University Engineering Department, 2005.

[14] P. Woodland, M. Gales, D. Pye, and S. Young. Broadcast news transcription using htk. In *ICASSP '97: Proceedings of the 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '97)-Volume 2*, page 719, Washington, DC, USA, 1997. IEEE Computer Society.

[15] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[16] Ioannis Tsochantaridis, Thomas Hofmann, Thorsten Joachims, and Yasemin Altun. Support vector machine learning for interdependent and structured output spaces. In *ICML*, 2004.