Speaker-Adaptive Acoustic-Articulatory Inversion Using Cascaded Gaussian Mixture Regression

Thomas Hueber, Laurent Girin, Xavier Alameda-Pineda, and Gérard Bailly

***Abstract*—This paper addresses the adaptation of an acoustic-articulatory model of a reference speaker to the voice of another speaker, using a limited amount of audio-only data. In the context of pronunciation training, a virtual talking head displaying the internal speech articulators (e.g., the tongue) could be automatically animated by means of such a model using only the speaker's voice. In this study, the articulatory-acoustic relationship of the reference speaker is modeled by a gaussian mixture model (GMM). To address the speaker adaptation problem, we propose a new framework called *cascaded Gaussian mixture regression* (C-GMR), and derive two implementations. The first one, referred to as Split-C-GMR, is a straightforward chaining of two distinct GMRs: one mapping the acoustic features of the source speaker into the acoustic space of the reference speaker, and the other estimating the articulatory trajectories with the reference model. In the second implementation, referred to as Integrated-C-GMR, the two mapping steps are tied together in a single probabilistic model. For this latter model, we present the full derivation of the exact EM training algorithm, that explicitly exploits the *missing data* methodology of machine learning. Other adaptation schemes based on maximum-*a posteriori* (MAP), maximum likelihood linear regression (MLLR) and direct cross-speaker acoustic-to-articulatory GMR are also investigated. Experiments conducted on two speakers for different amount of adaptation data show the interest of the proposed C-GMR techniques.**

***Index Terms*—Acoustic-articulatory inversion, EM algorithm, Gaussian mixture regression, pronunciation training, speaker adaptation, speech production, talking head.**

I. Introduction

Acoustic-articulatory inversion consists in the estimation of the movements of the speech articulators (e.g. tongue, lips, jaw, velum) from the speech audio signal. The underlying articulatory structure of speech can be exploited in different areas of speech technology, such as automatic speech recognition [1], low bit-rate speech coding [2], and speech synthesis [3]. Acoustic-to-articulatory inversion can also be used to animate a virtual talking head displaying the internal speech articulators using augmented reality [4]. Such a tool provides a complete and intuitive visual feedback that is useful for speech therapy [5], [6] and second language learning [7]. This is the applicative context of the present study.

Acoustic-to-articulatory inversion has been addressed in many studies using different techniques: codebook-based approaches [8]–[10], artificial neural networks [11], [12], support vector machines (SVM) [13], Gaussian mixture models (GMM) [14], [15], or hidden Markov models (HMM) [16], [17].

In most studies, acoustic-articulatory models are trained in a speaker-dependent way, using simultaneous recordings of audio and electromagnetic articulography (EMA) data. To design a pronunciation training system based on a virtual talking head, a *speaker adaptation* framework is required for two reasons. First, the user, referred here to as the *source speaker*, is generally different from the *reference speaker* for whom the model was trained. Because of the inter-speaker variability, feeding the acoustic-articulatory model of the reference speaker with data from the source speaker, is expected to yield poor articulatory trajectories (as confirmed by our experiments). Second, a practical usage scenario excludes the use of invasive devices (such as EMA) on the source speaker. Hence no articulatory data of the source speaker is assumed to be available for adaptation. Consequently, the research question addressed in this study is: How to adapt an acoustic-articulatory model of a reference speaker to a different speaker, using acoustic data only?

To the best of our knowledge, only a few studies addressed the problem of recovering articulatory movements from speech signals produced by a new (source) speaker, using a model trained for another (reference) speaker. In [18], Dusan and Deng proposed a vocal tract length normalization procedure to compensate the morphological differences between the two speakers. In [19], Hiroya *et al.* proposed to adapt an HMM-based acoustic-to-articulatory model [16]. However, this adaptation technique aims at adjusting not only the acoustic-articulatory relationships of the reference model to the source speaker, but also the geometry of the reference speaker's vocal tract. Given the targeted application, the goal of the present study is slightly different. We do not aim at representing the estimated articulatory gestures in the articulatory space of the source speaker, but rather in the articulatory space of the reference speaker (in other words, we do not want to modify the geometry of a talking head associated to this model).

Manuscript received February 09, 2015; revised June 19, 2015; accepted July 30, 2015. Date of publication August 05, 2015; date of current version September 16, 2015. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Junichi Yamagishi.

T. Hueber and G. Bailly are with the CNRS/GIPSA-lab and University of Grenoble Alpes/GIPSA-lab, 38400 Saint Martin D'Hères, France (e-mail: thomas.hueber@gipsa-lab.grenoble-inp.fr; gerard.bailly@gipsa-lab.grenoble-inp.fr).

L. Girin is with the University of Grenoble Alpes/ GIPSA-lab and INRIA Grenoble Rhône- Alpes, 38330 Montbonnot-Saint-Martin, France (e-mail: laurent.girin@gipsa- lab.grenoble-inp.fr).

X. Alameda-Pineda is with the University of Trento, 38122 Trento, Italy, and also with INRIA Grenoble Rhône-Alpes, 38330 Montbonnot-Saint-Martin, France (e-mail: xavier.alamedapineda@unitn.it).

This paper has supplemental downloadable multimedia material available at http://ieeexplore.ieee.org, provided by the authors. The Supplementary Materials contain an MPEG-4 movie showing an articulatory talking head, modeling a reference speaker and animated automatically from the speech audio signal of another (source) speaker using SC-GMR and IC-GMR techniques. This material is 13.7 MB in size.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASLP.2015.2464702

In the present study, the acoustic-articulatory inversion is addressed in the GMM framework and its associated regression technique called Gaussian mixture regression (GMR). Different strategies are investigated to adapt an acoustic-articulatory GMM trained on a reference speaker to a source speaker, using a limited amount of audio-only data. First, we investigate the use of standard techniques such as the maximum-a-posteriori (MAP) method [20] and the maximum likelihood linear regression (MLLR) [21] to adapt the acoustic part of the acoustic-articulatory GMM. We also consider a direct cross-speaker model, i.e. a model trained on source speaker's audio data aligned with the reference speaker's articulatory data. Finally, we introduce another approach called *cascaded Gaussian mixture regression* (C-GMR).

Two versions of the C-GMR are proposed, motivated and evaluated. The first one, referred to as *split cascaded GMR* (SC-GMR) is a straightforward chaining of two *distinct* GMRs. The main principle is here to map the acoustic features of the source speaker into the acoustic space of the reference speaker, similarly to a voice conversion system, before estimating the articulatory trajectories with the reference model. In the second version, referred to as the *integrated cascaded GMR* (IC-GMR), acoustic conversion and acoustic-to-articulatory inversion are completely *tied* and *integrated* in a single probabilistic model. For this model, we derive the exact expectation-maximization (EM) [22] algorithm that jointly optimizes the complete set of model parameters during adaptation (i.e. the model parameters related to the acoustic data of source and reference speakers and to the articulatory data of the reference speaker). Importantly, this algorithm is intended to deal with small adaptation datasets using the *missing data* methodology of machine learning [23], [24]. As for the inference, we use both "frame-by-frame" estimation based on the mean squared error (MSE) criterion and "utterance-by-utterance" estimation based on the maximum likelihood parameter generation (MLPG) algorithm. This latter algorithm was proposed by Tokuda *et al.* for HMM-based speech synthesis [25] and adapted to GMR by Toda *et al.* [26]. Note that a preliminary version of the IC-GMR technique was initially proposed in [27] but with incomplete theoretical foundations and no training algorithm.

This paper is organized as follows. Section II recalls the basics of GMR techniques. Section III formalizes MAP and MLLR-based adaptation schemes, as well as the direct cross-speaker GMR. Section IV presents the SC-GMR. Section V presents the IC-GMR. The associated EM algorithm is derived in Section VI. Experiments conducted to assess the performance of the proposed techniques are reported and discussed in Section VII. Section VIII provides conclusions and perspectives.

II. Gaussian Mixture Regression

In this section, we first recall the theoretical aspects and set the notations of the GMR which is the foundation for both SC-GMR and IC-GMR techniques.

A. Gaussian Mixture Model

Let us consider $\mathbf{X}$ and $\mathbf{Y}$ two random (column) vectors, of dimension D_X and D_Y respectively. Let us denote by $\mathbf{J}$ the concatenation of $\mathbf{X}$ and $\mathbf{Y}$ into a column vector, i.e. $\mathbf{J} = [\mathbf{X}^\top, \mathbf{Y}^\top]^\top$, where $^\top$ denotes the transpose operator. Let $p(\mathbf{x}|\mathbf{\Theta_X})$[1] denote the probability density function (PDF) of $\mathbf{X}$, parametrized by the set of parameters $\mathbf{\Theta_X}$. Let $\mathcal{N}(\mathbf{x}|\mu_\mathbf{X}, \mathbf{\Sigma_{XX}})$ denote the Gaussian distribution on $\mathbf{X}$ with mean vector $\mu_\mathbf{X}$ and covariance matrix $\mathbf{\Sigma_{XX}}$. Let $\mathbf{\Sigma_{XY}}$ denote the cross-covariance matrix between $\mathbf{X}$ and $\mathbf{Y}$. A GMM on $(\mathbf{X}, \mathbf{Y})$ consists of a weighted sum of Gaussian PDFs:

$$p(\mathbf{j}|\mathbf{\Theta_J}) = \sum_{m=1}^{M} \pi_m \mathcal{N}\left(\mathbf{j}|\mu_{\mathbf{J},m}, \mathbf{\Sigma}_{\mathbf{JJ},m}\right), \tag{1}$$

where M is the number of components of the mixture. For each component m, $\pi_m = p(m)$ is the prior probability satisfying $\sum_{m=1}^{M} \pi_m = 1$, $\mu_{\mathbf{J},m} = [\mu_{\mathbf{X},m}^\top \mu_{\mathbf{Y},m}^\top]^\top$ is the mean vector and $\mathbf{\Sigma}_{\mathbf{JJ},m}$ is the covariance matrix given by:

$$\mathbf{\Sigma}_{\mathbf{JJ},m} = \begin{bmatrix} \mathbf{\Sigma}_{\mathbf{XX},m} & \mathbf{\Sigma}_{\mathbf{XY},m} \\ \mathbf{\Sigma}_{\mathbf{YX},m} & \mathbf{\Sigma}_{\mathbf{YY},m} \end{bmatrix}. \tag{2}$$

All these parameters are estimated using the classical EM algorithm for GMM [28(ch. 9)].

It is well-known that if $\mathbf{J}$ follows a Gaussian distribution, the marginal distribution of $\mathbf{X}$ and the conditional distribution of $\mathbf{Y}$ given $\mathbf{x}$ are also Gaussian. These results extend to Gaussian mixtures and we have:

$$p(\mathbf{y}|\mathbf{x}, \mathbf{\Theta_J}) = \sum_{m=1}^{M} p(m|\mathbf{x}, \mathbf{\Theta_X}) \mathcal{N}(\mathbf{y}|\mu_{\mathbf{Y}|\mathbf{x},m}, \mathbf{\Sigma}_{\mathbf{YY}|\mathbf{x},m}), \tag{3}$$

with

$$\mu_{\mathbf{Y}|\mathbf{x},m} = \mu_{\mathbf{Y},m} + \mathbf{\Sigma}_{\mathbf{YX},m}\mathbf{\Sigma}_{\mathbf{XX},m}^{-1}(\mathbf{x} - \mu_{\mathbf{X},m}), \tag{4}$$

$$\mathbf{\Sigma}_{\mathbf{YY}|\mathbf{x},m} = \mathbf{\Sigma}_{\mathbf{YY},m} - \mathbf{\Sigma}_{\mathbf{YX},m}\mathbf{\Sigma}_{\mathbf{XX},m}^{-1}\mathbf{\Sigma}_{\mathbf{XY},m}, \tag{5}$$

$$p(m|\mathbf{x}, \mathbf{\Theta_X}) = \frac{\pi_m \mathcal{N}\left(\mathbf{x}|\mu_{\mathbf{X},m}, \mathbf{\Sigma}_{\mathbf{XX},m}\right)}{\sum_{i=1}^{M} \pi_i \mathcal{N}\left(\mathbf{x}|\mu_{\mathbf{X},i}, \mathbf{\Sigma}_{\mathbf{XX},i}\right)}. \tag{6}$$

The conditional distribution (3) can be rewritten as a mixture of linear-Gaussian forms:

$$p(\mathbf{y}|\mathbf{x}, \mathbf{\Theta_J}) = \sum_{m=1}^{M} w_m \mathcal{N}\left(\mathbf{y}|\mathbf{A}_m^*\mathbf{x} + \mathbf{b}_m^*, \mathbf{U}_m^*\right), \tag{7}$$

with $w_m = p(m|\mathbf{x}, \mathbf{\Theta}_{\mathbf{X},m})$, $\mathbf{A}_m^* = \mathbf{\Sigma}_{\mathbf{YX},m}\mathbf{\Sigma}_{\mathbf{XX},m}^{-1}$, $\mathbf{b}_m^* = \mu_{\mathbf{Y},m} - \mathbf{A}_m^*\mu_{\mathbf{X},m}$, and $\mathbf{U}_m^* = \mathbf{\Sigma}_{\mathbf{YY}|\mathbf{x},m}$ (this choice of notation will become clear in Section V-B).

B. GMR-MSE

The conditional GMM (3) (or (7)) can be used to map $\mathbf{x}$ into an estimated value $\hat{\mathbf{y}}$ of $\mathbf{y}$. When the mapping is done to mini-

[1] $p(\mathbf{x}|\mathbf{\Theta_X})$ is an abuse of notation, meaning $p(\mathbf{X} = \mathbf{x}|\mathbf{\Theta_X})$. Upper-case letters $\mathbf{X}$ denote random vectors, and lower-case letters $\mathbf{x}$, realizations.

mize the mean squared error (MSE) independently for each observation vector $\mathbf{x}_t$, we obtain the well-known result:

$$\hat{\mathbf{y}}_t = \mathrm{E}[\mathbf{Y}_t|\mathbf{x}_t, \mathbf{\Theta_J}] = \sum_{m=1}^{M} p(m|\mathbf{x}_t, \mathbf{\Theta_X})\mu_{\mathbf{Y}_t|\mathbf{x}_t,m} = \sum_{m=1}^{M} w_m(\mathbf{A}_m^*\mathbf{x}_t + \mathbf{b}_m^*). \quad (8)$$

This mapping is referred to as a gaussian mixture regressor [29] (of $\mathbf{x}_t$ into $\hat{\mathbf{y}}_t$) based on MSE criterion (GMR-MSE).

C. GMR-MLPG

Alternatively, [26] proposed a joint estimator for a sequence of T vectors $[\mathbf{y}_1, \dots, \mathbf{y}_t, \dots, \mathbf{y}_T]$ given an input vector sequence $[\mathbf{x}_1, \dots, \mathbf{x}_t, \dots, \mathbf{x}_T]$. This estimator has the following expression:

$$\tilde{\mathbf{y}}_{\text{seq}} = \left(\mathbf{W}^\top\mathbf{D}^{-1}\mathbf{W}\right)^{-1}\mathbf{W}^\top\mathbf{D}^{-1}\mathbf{E}, \quad (9)$$

where $\tilde{\mathbf{y}}_{\text{seq}} = [\tilde{\mathbf{y}}_1^\top, \dots, \tilde{\mathbf{y}}_t^\top, \dots, \tilde{\mathbf{y}}_T^\top]^\top$ is a $D_Y T$ column vector[2], $\mathbf{W}$ is a matrix encoding the linear dependencies between static features and their derivatives, $\mathbf{E}$ is built from the MSE estimation for each input vector computed with (8) and $\mathbf{D}$ is a block-diagonal matrix built from the conditional covariance matrices (5) and posteriors (6), for the whole considered sequence. The reader is referred to [26] for more details. This approach will be referred to as GMR-MLPG since it is an adaptation of the maximum likelihood parameter generation algorithm (MLPG) proposed in [25] for HMM-based synthesis, to the GMM-based mapping. By imposing a consistent relationship between static and dynamic features, this mapping generates smooth trajectories.

III. GMR Adaptation: Principles and Baseline Methods

A. Principles

Let first define the following three random vectors: $\mathbf{X}$ and $\mathbf{Y}$ which are respectively acoustic and articulatory feature vectors of the reference speaker, and $\mathbf{Z}$ which is a corresponding acoustic vector from the source speaker. In practice, $\mathbf{X}$ and $\mathbf{Z}$ are composed of MFCC and Δ MFCC coefficients, and $\mathbf{Y}$ are EMA articulatory vectors (see Section VII). As illustrated in Fig. 1, let us assume that we have an extensive set of N joint observations $\{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N = \{\mathbf{x}_{1:N}, \mathbf{y}_{1:N}\}$ for the training of the reference speaker GMM (using the EM algorithm for GMM). In the adaptation stage, the source speaker is asked to pronounce a subset of the above dataset (typically a few minutes of speech). We note N_0 the number of acoustic features vectors in the adaptation dataset which is noted here $\{\mathbf{z}_n\}_{n=1}^{N_0} = \mathbf{z}_{1:N_0}$ with indeed $N_0 < N$ (in practice we can have $N_0 \ll N$).

B. MAP and MLLR

Maximum-a-posteriori (MAP) [20] and maximum likelihood linear regression (MLLR) [21] are two state-of-the-art techniques used in automatic speech and speaker recognition to adapt a GMM (or HMM) using a new set of observations. In the present study, the goal is to adapt the acoustic-articulatory (i.e. $\mathbf{X}$-to-$\mathbf{Y}$) GMR of the reference speaker using acoustic-only (i.e. $\mathbf{Z}$) observations of the source speaker. Therefore, we propose to apply the MAP and MLLR methodology to adapt the "acoustic part" of the $\mathbf{X}$-$\mathbf{Y}$ GMM, i.e. $\mu_{\mathbf{X},m}$ and $\mathbf{\Sigma}_{\mathbf{XX},m}$ for each component m.

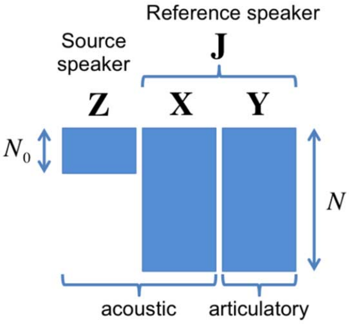

Fig. 1. Schematic representation of the key variables in the C-GMR framework with missing data.

The basic principle of the MAP adaptation is to find the model parameter set $\mathbf{\Theta}^{\mathbf{Z}}_{MAP}$ that maximizes the posterior probability $p(\mathbf{\Theta}^{\mathbf{Z}}_{MAP}|\mathbf{z})$ considering $\mathbf{\Theta_X}$ as prior knowledge over model parameters. The parameter set $\mathbf{\Theta}^{\mathbf{Z}}_{MAP}$ is determined using an EM algorithm with the following re-estimation equation (to be concise, we recall only the equation for the mean vectors; see [20] for the update equations of priors and covariance matrices):

$$\mu^{MAP}_{\mathbf{X},m} = \frac{\tau\mu_{\mathbf{X},m} + \sum_{n=1}^{N_0} p(m|\mathbf{z}_n, \mathbf{\Theta_Z})\mathbf{z}_n}{\tau + \sum_{n=1}^{N_0} p(m|\mathbf{z}_n, \mathbf{\Theta_Z})}, \quad (10)$$

where τ is a heuristic hyperparameter shared across all GMM components, controlling the balance between the prior knowledge and the adaptation data.

In MLLR, the model parameters are adapted using an affine transform: $\mu^{MLLR}_{\mathbf{X},m} = \mathbf{G}\mu_{\mathbf{X},m} + \mathbf{q}$ and $\mathbf{\Sigma}^{MLLR}_{\mathbf{XX},m} = \mathbf{H}\mathbf{\Sigma}_{\mathbf{XX},m}\mathbf{H}^\top$. The adaptation data likelihood is maximized with respect to the transform parameters $(\mathbf{G}, \mathbf{q}, \mathbf{H})$ using an EM algorithm. In our implementation, these transform parameters are shared across all GMM components. Therefore, MLLR imposes the same affine transformation to all GMM components, whereas MAP updates each component separately.

C. Cross-Speaker Acoustic-Articulatory GMR

Another straight-forward way to address the considered problem is to directly model the statistical relationships between the source speaker's acoustics $\mathbf{Z}$ and the reference speaker's articulation $\mathbf{Y}$ with a $\mathbf{Z}$-$\mathbf{Y}$ GMM (and directly derive the corresponding $\mathbf{Z}$-to-$\mathbf{Y}$ GMR). This approach is referred to as the "direct" cross-speaker acoustic-articulatory GMR (D-GMR). Importantly, training this model requires to associate the $\mathbf{z}_{1:N_0}$ adaptation data with "corresponding" $\mathbf{y}_{1:N_0}$ articulatory data. This is done by time-aligning each adaptation sentence pronounced by the source speaker, with the same sentence pronounced by the reference speaker, using a dynamic time warping (DTW) algorithm. After this procedure, the adaptation data $\mathbf{z}_{1:N_0}$ are assumed to be aligned with observations $\mathbf{x}_{1:N_0}$, and thus with corresponding articulatory data $\mathbf{y}_{1:N_0}$

[2] Notice that ~ refers to the MLPG estimator and ^ refers to the MSE estimator.

(reordering of the vectors is arbitrary). The EM algorithm for GMM is then applied to the set $\{\mathbf{z}_{1:N_0}, \mathbf{y}_{1:N_0}\}$.

IV. Split Cascaded Gaussian Mixture Regression

A. Motivation

One problem with the MAP and MLLR approaches is that the adaptation of the acoustic parameters of the $\mathbf{X}$-to-$\mathbf{Y}$ GMR is done independently of the joint acoustic-articulatory and articulatory parameters, leading to a potential mismatch. As for the (cross-speaker) D-GMR, the estimated acoustic-articulatory model relies on a limited number N_0 of articulatory observations of the reference speaker (among the N available observations). Because of the complexity of acoustic-articulation relationships, this may lead to poor inversion performances, especially when considering small adaptation datasets.

These two limitations motivated the development of the proposed Cascaded-GMR framework for the considered speaker adaptation problem. Indeed, the first core motivation of this framework is to develop an adaptation technique which benefits from *all the available articulatory data* of the reference speaker. In other words, the reference model should remain at the center of the adaptation process, while exploiting the acoustic adaptation data. In addition to that, the C-GMR aims at avoiding potential mismatch between adapted and original model parameters. This can be achieved in two ways:

- either by keeping the complete $\mathbf{X}$-to-$\mathbf{Y}$ reference GMR intact and making the acoustic observation $\mathbf{z}$ compatible with this model. This is the general idea of the *split cascaded GMR* (SC-GMR) that we define in the next subsection.
- or by jointly modeling the statistical relationships of the three vectors $\mathbf{Z}$, $\mathbf{X}$ and $\mathbf{Y}$. This is the spirit of the *integrated cascaded GMR* (IC-GMR), that will be described in Section V.

B. Definition

The split cascaded GMR (SC-GMR) consists of chaining two separate GMRs: a $\mathbf{Z}$-to-$\mathbf{X}$ spectral conversion module (similarly to [30]) followed by a $\mathbf{X}$-to-$\mathbf{Y}$ acoustic-to-articulatory inversion module. As illustrated by Fig. 2, those two GMRs are separated in the sense that the two successive mappings are independent: the output of the first one is calculated before being injected as input of the second one. In other words, we have $\hat{\mathbf{y}} = \mathrm{E}[\mathbf{Y}|\hat{\mathbf{x}}, \mathbf{\Theta_J}]$ with $\hat{\mathbf{x}} = \mathrm{E}[\mathbf{X}|\mathbf{z}, \mathbf{\Theta_I}]$ (being $\mathbf{I} = [\mathbf{Z}^\top, \mathbf{X}^\top]^\top$), where both expectations follow (8) with their respective parameters. Note that the two GMRs may have a different number of mixture components. The $\mathbf{X}$-to-$\mathbf{Y}$ GMR parameters are estimated with the EM algorithm for GMM applied on the complete $\{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^{N}$ reference dataset, “as usual”. The $\mathbf{Z}$-to-$\mathbf{X}$ GMR parameters are estimated with the EM algorithm for GMM applied on the aligned adaptation dataset $\{(\mathbf{z}_n, \mathbf{x}_n)\}_{n=1}^{N_0}$ (see Section III-C).

Compared to the D-GMR, one key-point of the SC-GMR is that the reference acoustic-articulatory model is trained from all the N available acoustic-articulatory observations of the reference speaker. The limited amount of N_0 data is used to model the statistical relationships between source and reference acoustic spaces. This spectral mapping is assumed to be simpler, or say, “better-posed,” than acoustic-articulatory mapping, and may therefore require less training data.

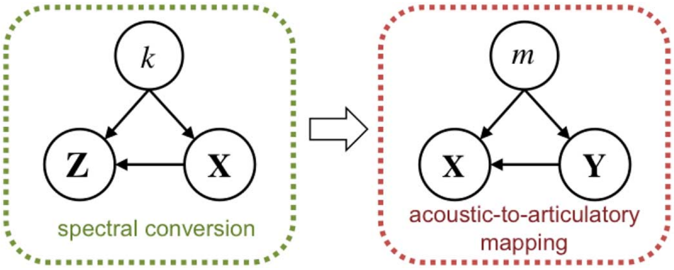

Fig. 2. Graphical representation of SC-GMR.

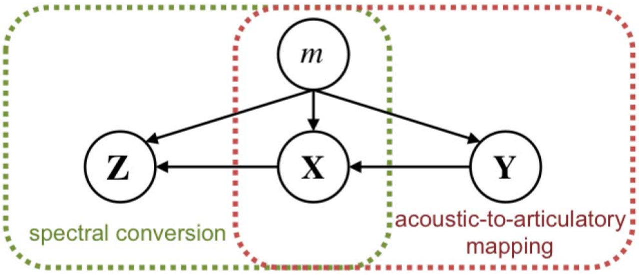

Fig. 3. Graphical representation of IC-GMR.

V. Integrated Cascaded GMR

We now present the integrated cascaded GMR (IC-GMR) model, that we propose to address the present speaker adaptation problem. Then, we discuss the specific way the EM algorithm is to be used in this context. The technical derivation of this algorithm is given in the next section.

A. Definition of the Mixture Model

The core idea of the IC-GMR model is to combine spectral conversion and acoustic-articulatory inversion into a single GMR-based mapping process. Very importantly, this is made at the component level of the GMR, i.e. *within the mixture*, as opposed to the SC-GMR of Section IV. In other words, the plugged “conversion + inversion” components share the same component assignment variable m, as illustrated by the graphical model shown in Fig. 3. The goal is to benefit from the partitioning of the acoustic-articulatory space of the reference speaker (i.e. $\mathbf{X}$-$\mathbf{Y}$) which is assumed to be well estimated, when proceeding to the source speaker adaptation. Contrary to the SC-GMR, the structure of the $\mathbf{Z}$-to-$\mathbf{X}$ conversion process is thus here constrained by the structure of the $\mathbf{X}$-to-$\mathbf{Y}$ GMR.

The statistical dependencies between $\mathbf{X}$, $\mathbf{Y}$ and $\mathbf{Z}$ are here defined as:

$$p(\mathbf{x}, \mathbf{y}, \mathbf{z}|\mathbf{\Theta}) = \sum_{m=1}^{M} p(m) p(\mathbf{y}|m, \mathbf{\Theta}_{\mathbf{Y},m}) \times p(\mathbf{x}|\mathbf{y}, m, \mathbf{\Theta}_{\mathbf{X}|\mathbf{Y},m}) \times p(\mathbf{z}|\mathbf{x}, m, \mathbf{\Theta}_{\mathbf{Z}|\mathbf{X},m}), \quad (11)$$

with

$$p(m) = \pi_m, \quad (12)$$

$$p(\mathbf{y}|m, \mathbf{\Theta}_{\mathbf{Y},m}) = \mathcal{N}(\mathbf{y}|\mathbf{e}_m, \mathbf{R}_m), \quad (13)$$

$$p(\mathbf{x}|\mathbf{y}, m, \mathbf{\Theta}_{\mathbf{X}|\mathbf{Y},m}) = \mathcal{N}(\mathbf{x}|\mathbf{A}_m\mathbf{y} + \mathbf{b}_m, \mathbf{U}_m), \quad (14)$$

$$p(\mathbf{z}|\mathbf{x}, m, \mathbf{\Theta}_{\mathbf{Z}|\mathbf{X},m}) = \mathcal{N}(\mathbf{z}|\mathbf{C}_m\mathbf{x} + \mathbf{d}_m, \mathbf{V}_m). \quad (15)$$

For each component, π_m still represents the prior distribution, $\mathbf{e}_m$ and $\mathbf{R}_m$ are respectively the mean vector and covariance matrix of the marginal Gaussian distribution of $\mathbf{Y}$, $\mathbf{A}_m$, $\mathbf{b}_m$ and $\mathbf{U}_m$ are respectively the transition matrix, constant vector and covariance matrix of the linear-Gaussian conditional pdf model in $(\mathbf{X}, \mathbf{Y})$, and the same for $\mathbf{C}_m$, $\mathbf{d}_m$ and $\mathbf{V}_m$ with $(\mathbf{Z}, \mathbf{X})$.

B. Inference Equation

Similarly to Section II, the minimum MSE estimation $\hat{\mathbf{y}}$ of $\mathbf{y}$ given $\mathbf{z}$ is given by its posterior mean[3]:

$$\hat{\mathbf{y}} = \mathrm{E}[\mathbf{Y}|\mathbf{z}] = \int_{\mathbb{R}^{D_Y}} \mathbf{y} p(\mathbf{y}|\mathbf{z}) \mathrm{d}\mathbf{y}, \tag{16}$$

with

$$p(\mathbf{y}|\mathbf{z}) = \int_{\mathbb{R}^{D_X}} \sum_{m=1}^{M} p(\mathbf{x}, \mathbf{y}, m|\mathbf{z}) \mathrm{d}\mathbf{x}. \tag{17}$$

In the IC-GMR case we have:

$$\begin{aligned} p(\mathbf{x}, \mathbf{y}, m|\mathbf{z}) &= p(m|\mathbf{z}) p(\mathbf{y}|\mathbf{x}, \mathbf{z}, m) p(\mathbf{x}|\mathbf{z}, m) \\ &= p(m|\mathbf{z}) p(\mathbf{y}|\mathbf{x}, m) p(\mathbf{x}|\mathbf{z}, m), \end{aligned} \tag{18}$$

since $\mathbf{Y}$ is independent of $\mathbf{Z}$ conditionally on $\mathbf{X}$ and m [28–(Section VIII.2]). Therefore, we have:

$$p(\mathbf{y}|\mathbf{z}) = \sum_{m=1}^{M} p(m|\mathbf{z}) \int_{\mathbb{R}^{D_X}} p(\mathbf{y}|\mathbf{x}, m) p(\mathbf{x}|\mathbf{z}, m) \mathrm{d}\mathbf{x}. \tag{19}$$

At this point, we can insert (19) into (16). But to go further, we face a problem: the model is expressed in terms of the distributions $p(\mathbf{y}|m)$, $p(\mathbf{x}|\mathbf{y}, m)$, $p(\mathbf{z}|\mathbf{x}, m)$ and not the "inverse" distributions $p(\mathbf{z}|m)$, $p(\mathbf{x}|\mathbf{z}, m)$, $p(\mathbf{y}|\mathbf{x}, m)$ as required in (19)[4]. Fortunately, a linear-Gaussian model is "invertible": knowing the Gaussian PDFs $p(\mathbf{y})$ and $p(\mathbf{x}|\mathbf{y})$, the PDFs $p(\mathbf{x})$ and $p(\mathbf{y}|\mathbf{x})$ are derived easily and form a linear-Gaussian model [28(p. 93)]. In the present case, we can chain the inversion across $\mathbf{Y}$, $\mathbf{X}$ and $\mathbf{Z}$ to obtain:

$$p(\mathbf{y}|\mathbf{x}, m, \boldsymbol{\Theta}_{\mathbf{Y}|\mathbf{X},m}) = \mathcal{N}\left(\mathbf{y}|\mathbf{A}_m^* \mathbf{x} + \mathbf{b}_m^*, \mathbf{U}_m^*\right), \tag{20}$$

$$p(\mathbf{x}|m, \boldsymbol{\Theta}_{\mathbf{X},m}) = \mathcal{N}\left(\mathbf{x}|\mathbf{e}_m^*, \mathbf{R}_m^*\right), \tag{21}$$

$$p(\mathbf{x}|\mathbf{z}, m, \boldsymbol{\Theta}_{\mathbf{X}|\mathbf{Z},m}) = \mathcal{N}\left(\mathbf{x}|\mathbf{C}_m^* \mathbf{z} + \mathbf{d}_m^*, \mathbf{V}_m^*\right), \tag{22}$$

$$p(\mathbf{z}|m, \boldsymbol{\Theta}_{\mathbf{Z},m}) = \mathcal{N}\left(\mathbf{z}|\mathbf{f}_m^*, \mathbf{P}_m^*\right), \tag{23}$$

with

$$\begin{aligned} \mathbf{U}_m^* &= (\mathbf{R}_m^{-1} + \mathbf{A}_m^\top \mathbf{U}_m^{-1} \mathbf{A}_m)^{-1}, \\ \mathbf{A}_m^* &= \mathbf{U}_m^* \mathbf{A}_m^\top \mathbf{U}_m^{-1}, \mathbf{b}_m^* = \mathbf{U}_m^* (\mathbf{R}_m^{-1} \mathbf{e}_m - \mathbf{A}_m^\top \mathbf{U}_m^{-1} \mathbf{b}_m), \\ \mathbf{R}_m^* &= \mathbf{U}_m + \mathbf{A}_m \mathbf{R}_m \mathbf{A}_m^\top, \mathbf{e}_m^* = \mathbf{A}_m \mathbf{e}_m + \mathbf{b}_m, \\ \mathbf{V}_m^* &= (\mathbf{R}_m^{*-1} + \mathbf{C}_m^\top \mathbf{V}_m^{-1} \mathbf{C}_m)^{-1}, \\ \mathbf{C}_m^* &= \mathbf{V}_m^* \mathbf{C}_m^\top \mathbf{V}_m^{-1}, \\ \mathbf{d}_m^* &= \mathbf{V}_m^* (\mathbf{R}_m^{*-1} \mathbf{e}_m^* - \mathbf{C}_m^\top \mathbf{V}_m^{-1} \mathbf{d}_m), \\ \mathbf{P}_m^* &= \mathbf{V}_m + \mathbf{C}_m \mathbf{R}_m^* \mathbf{C}_m^\top, \mathbf{f}_m^* = \mathbf{C}_m \mathbf{e}_m^* + \mathbf{d}_m. \end{aligned}$$

Now we can calculate (16) as:

$$\begin{aligned} \hat{\mathbf{y}} &= \sum_{m=1}^{M} p(m|\mathbf{z}) \int_{\mathbb{R}^{D_X}} \left(\int_{\mathbb{R}^{D_Y}} \mathbf{y} p(\mathbf{y}|\mathbf{x}, m) \mathrm{d}\mathbf{y} \right) \\ &\quad \times p(\mathbf{x}|\mathbf{z}, m) \mathrm{d}\mathbf{x} \\ &= \sum_{m=1}^{M} p(m|\mathbf{z}) \int_{\mathbb{R}^{D_X}} (\mathbf{A}_m^* \mathbf{x} + \mathbf{b}_m^*) p(\mathbf{x}|\mathbf{z}, m) \mathrm{d}\mathbf{x} \\ &= \sum_{m=1}^{M} p(m|\mathbf{z}) (\mathbf{A}_m^* (\mathbf{C}_m^* \mathbf{z} + \mathbf{d}_m^*) + \mathbf{b}_m^*), \end{aligned} \tag{24}$$

and finally:

$$\hat{\mathbf{y}} = \sum_{m=1}^{M} p(m|\mathbf{z}) (\mathbf{A}_m^* \mathbf{C}_m^* \mathbf{z} + \mathbf{A}_m^* \mathbf{d}_m^* + \mathbf{b}_m^*). \tag{25}$$

Similarly to (7), it can be shown that $\mathbf{C}_m^* = \boldsymbol{\Sigma}_{\mathbf{XZ},m} \boldsymbol{\Sigma}_{\mathbf{ZZ},m}^{-1}$, $\mathbf{d}_m^* = \mu_{\mathbf{X},m} - \mathbf{C}_m^* \mu_{\mathbf{Z},m}$. Therefore, (25) is equivalent to:

$$\begin{aligned} \hat{\mathbf{y}} = \sum_{m=1}^{M} p(m|\mathbf{z}) (\mu_{\mathbf{Y},m} \\ + \boldsymbol{\Sigma}_{\mathbf{YX},m} \boldsymbol{\Sigma}_{\mathbf{XX},m}^{-1} \boldsymbol{\Sigma}_{\mathbf{XZ},m} \boldsymbol{\Sigma}_{\mathbf{ZZ},m}^{-1} (\mathbf{z} - \mu_{\mathbf{Z},m})). \end{aligned} \tag{26}$$

The component weights $p(m|\mathbf{z})$ are obtained by applying the classical formula (6) with distribution (23).

Equation (26) was initially proposed in [27], but without theoretical support. It exhibits the chaining of $\mathbf{Z}$-to-$\mathbf{X}$ and $\mathbf{X}$-to-$\mathbf{Y}$ linear regressions at the mixture component level. This results into a $\mathbf{Z}$-to-$\mathbf{Y}$ GMR with a specific form of the covariance matrix $\boldsymbol{\Sigma}_{\mathbf{YZ},m} = \boldsymbol{\Sigma}_{\mathbf{YX},m} \boldsymbol{\Sigma}_{\mathbf{XX},m}^{-1} \boldsymbol{\Sigma}_{\mathbf{XZ},m}$. Note that these parameters depend on the joint distribution of $(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$, and in practice they are estimated from all available $(\mathbf{x}, \mathbf{y}, \mathbf{z})$ data (as we will see below). Even if their inference equation has the same general form, this makes the IC-GMR quite different from the D-GMR of Section III-C: Remind that this latter was obtained from a limited set of $N_0(\mathbf{z}, \mathbf{y})$ data only.

Similarly to Section II-C, we can derive the MLPG form for the IC-GMR. The mean matrix $\mathbf{E}$ is here constructed from (25), and the covariance matrix $\mathbf{D}$ is constructed from the conditional covariance matrices:

$$\boldsymbol{\Sigma}_{\mathbf{YY}|\mathbf{z},m} = \boldsymbol{\Sigma}_{\mathbf{YY},m} - \boldsymbol{\Sigma}_{\mathbf{YZ},m} \boldsymbol{\Sigma}_{\mathbf{ZZ},m}^{-1} \boldsymbol{\Sigma}_{\mathbf{ZY},m}, \tag{27}$$

which can be shown to be equal to

$$\boldsymbol{\Sigma}_{\mathbf{YY}|\mathbf{z},m} = \mathbf{R}_m - \mathbf{A}_m^* \mathbf{C}_m^* \mathbf{C}_m \mathbf{A}_m \mathbf{R}_m. \tag{28}$$

The sequence $\tilde{\mathbf{y}}_{\mathrm{seq}}$ is then estimated from sequence $\mathbf{z}$ by applying (9) with the constructed $\mathbf{E}$ and $\mathbf{D}$ matrices.

C. IC-GMR and EM for Speaker Adaptation

In order to infer the articulatory trajectory $\mathbf{y}$ from the acoustic features of the source speaker $\mathbf{z}$ by means of (25), the parameters of the joint model (11) need to be estimated from the data. Since (11) is a mixture model, this naturally leads to an EM algorithm [22], [28], whose derivation is given in the next section. In general, the initialization of EM algorithms is known to be a

[3] In this subsection we omit the parameter set in PDF notation for clarity of presentation.

[4] $p(m|\mathbf{z})$ can be deduced from $p(\mathbf{z}|m)$ using Bayes formula (6).

crucial phase. In the present study, we propose the following strategy:

- First, the reference GMR is obtained from an extensive set of $(\mathbf{x}, \mathbf{y})$ data, using the EM algorithm for GMMs just as in Section IV-B.
- Second, we note that the joint marginal distribution of $(\mathbf{X}, \mathbf{Y})$ obtained by integrating (11) over $\mathbf{z}$ is given by:

$$p(\mathbf{j}|\mathbf{\Theta_J}) = \sum_{m=1}^{M} \pi_m p(\mathbf{y}|m, \mathbf{\Theta}_{\mathbf{Y},m}) p(\mathbf{x}|\mathbf{y}, m, \mathbf{\Theta}_{\mathbf{X}|\mathbf{Y},m}). \quad (29)$$

  Since for each m, both the marginal distribution of $\mathbf{Y}$ and the conditional distribution of $\mathbf{X}|\mathbf{y}$ are Gaussian, (29) is equivalent to the standard GMM on $(\mathbf{X}, \mathbf{Y})$ given in (1). Therefore, the parameters of (29), i.e. $\{\pi_m, \mathbf{e}_m, \mathbf{R}_m, \mathbf{A}_m, \mathbf{b}_m, \mathbf{U}_m\}_{m=1}^{M}$ are computed from the parameters of the reference GMR[5].
- Third, $\{\mathbf{C}_m, \mathbf{d}_m, \mathbf{V}_m\}_{m=1}^{M}$, i.e., the parameters involving $\mathbf{Z}$, are initialized using the N_0 aligned $(\mathbf{z}, \mathbf{x})$ data.
- Finally, after the initialization is done, both the N_0 aligned $(\mathbf{z}, \mathbf{x}, \mathbf{y})$ data and the remaining $N - N_0(\mathbf{x}, \mathbf{y})$ data are used to train the IC-GMR. Most importantly, all data are used to *jointly update all IC-GMR parameters*, as opposed to the SC-GMR adaptation, where the reference model remains unchanged, i.e. its parameters are not influenced by the adaptation data $\mathbf{z}_{1:N_0}$.

VI. EM Algorithm for IC-GMR

In this section, we derive the exact EM algorithm associated to the IC-GMR model presented in the previous section. The aim of the EM algorithm is to maximize the *expected complete-data log-likelihood*, denoted by Q. At each iteration, the E-step computes Q and the M-step maximizes Q with respect to the parameters $\mathbf{\Theta}$. The EM algorithm alternates between the E and M steps until convergence.

A. E-step

At iteration $i+1$, $Q(\mathbf{\Theta}, \mathbf{\Theta}^{(i)})$ is defined as the expected value of the complete data log-likelihood with parameter set $\mathbf{\Theta}$. The expectation is taken accordingly to the posterior distribution of latent variables given the observed data and the parameter set at the previous iteration, $\mathbf{\Theta}^{(i)}$. In order to derive the Q function we follow the general methodology given in, e.g., [28]–(Section 9.4) and [23]. This leads to (30), where all pdfs are defined in Section V-A, $\mathbf{j}_n = [\mathbf{x}_n^\top \mathbf{y}_n^\top]^\top$, $\mathbf{o}_n = [\mathbf{x}_n^\top \mathbf{y}_n^\top \mathbf{z}_n^\top]^\top$ (see the details in Appendix A). For $n \in [1, N_0]$,

$$\begin{aligned} Q\left(\mathbf{\Theta}, \mathbf{\Theta}^{(i)}\right) &= \sum_{n=1}^{N_0} \sum_{m=1}^{M} \gamma_m^{(i+1)}(\mathbf{o}_n) \log p(\mathbf{o}_n, m|\mathbf{\Theta}_m) \\ &+ \sum_{n=N_0+1}^{N} \sum_{m=1}^{M} \frac{1}{p\left(\mathbf{j}_n|\mathbf{\Theta}_{\mathbf{J}}^{(i)}\right)} \int_{\mathbb{R}^{D_Z}} p\left(\mathbf{o}_n, m|\mathbf{\Theta}_m^{(i)}\right) \\ &\times \log p(\mathbf{o}_n, m|\mathbf{\Theta}_m) \mathrm{d}\mathbf{z}_n \end{aligned} \quad (30)$$

$$\gamma_m^{(i+1)}(\mathbf{o}_n) = \frac{p(\mathbf{o}_n, m|\mathbf{\Theta}_m^{(i)})}{p(\mathbf{o}_n|\mathbf{\Theta}^{(i)})} \quad (31)$$

are the so-called *responsibilities* (of component m explaining observation $\mathbf{o}_n$) [28]. Note that (30) is valid for any trivariate mixture model on $(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$ (or any bivariate mixture model on $(\mathbf{J}, \mathbf{Z})$) with partially missing $\mathbf{z}$ data and i.i.d. observations. If we now extend the definition of responsibilities for $n \in [N_0+1, N]$ with:

$$\gamma_m^{(i+1)}(\mathbf{j}_n) = \frac{p(\mathbf{j}_n, m|\mathbf{\Theta}_{\mathbf{J},m}^{(i)})}{p(\mathbf{j}_n|\mathbf{\Theta}_{\mathbf{J}}^{(i)})}, \quad (32)$$

and we use the IC-GMR definition (11)–(15), (30) becomes (see Appendix A for details):

$$\begin{aligned} Q(\mathbf{\Theta}, \mathbf{\Theta}^{(i)}) &= \frac{1}{2} \sum_{n=1}^{N_0} \sum_{m=1}^{M} \gamma_m^{(i+1)}(\mathbf{o}_n) \big(2 \log \pi_m - \log |\mathbf{R}_m| \\ &- \log |\mathbf{U}_m| - \log |\mathbf{V}_m| - \|\mathbf{y}_n - \mathbf{e}_m\|_{\mathbf{R}_m}^2 \\ &- \|\mathbf{x}_n - \mathbf{A}_m \mathbf{y}_n - \mathbf{b}_m\|_{\mathbf{U}_m}^2 - \|\mathbf{z}_n - \mathbf{C}_m \mathbf{x}_n - \mathbf{d}_m\|_{\mathbf{V}_m}^2\big) \\ &+ \frac{1}{2} \sum_{n=N_0+1}^{N} \sum_{m=1}^{M} \gamma_m^{(i+1)}(\mathbf{j}_n) \\ &\times \Big(2 \log \pi_m - \log |\mathbf{R}_m| - \log |\mathbf{U}_m| \\ &- \log |\mathbf{V}_m| - \|\mathbf{y}_n - \mathbf{e}_m\|_{\mathbf{R}_m}^2 - \|\mathbf{x}_n - \mathbf{A}_m \mathbf{y}_n - \mathbf{b}_m\|_{\mathbf{U}_m}^2 \\ &- \|\mathbf{C}_m^{(i)} \mathbf{x}_n + \mathbf{d}_m^{(i)} - \mathbf{C}_m \mathbf{x}_n - \mathbf{d}_m\|_{\mathbf{V}_m}^2 - \mathrm{Tr}[\mathbf{V}_m^{-1} \mathbf{V}_m^{(i)}]\Big), \end{aligned} \quad (33)$$

where $\|\mathbf{x}\|_{\mathbf{R}}^2 = \mathbf{x}^\top \mathbf{R}^{-1} \mathbf{x}$ denotes the Mahalanobis distance of $\mathbf{x}$ with matrix $\mathbf{R}$ and Tr stands for the trace operator. The sum in the range $[1, N_0]$ is a direct match of [28–(9.40)], i.e. the classical EM for GMM, while the sum in $[N_0+1, N]$ results from the expectation over the missing data $\mathbf{z}_n$.

For $n \in [N_0+1, N]$, let us denote the expected value of $\mathbf{Z}_n$ given $\mathbf{x}_n$ for the m-th model component by $\mathbf{z}'_{nm} = \mathbf{C}_m^{(i)} \mathbf{x}_n + \mathbf{d}_m^{(i)} = \mu_{\mathbf{Z}|\mathbf{x}_n,m}^{(i+1)}$. This amounts to replace the missing data with their conditional mean given $\mathbf{x}_n$ and the current model parameters. For convenience, let us extend the notation $\mathbf{z}'_{nm}$ to the interval $n \in [1, N_0]$ with $\mathbf{z}'_{nm} = \mathbf{z}_n$ (which does not depend on m here). If, in addition, we denote $\gamma_{nm}^{(i+1)} = \gamma_m^{(i+1)}(\mathbf{o}_n)$ for $n \in [1, N_0]$ and $\gamma_{nm}^{(i+1)} = \gamma_m^{(i+1)}(\mathbf{j}_n)$ for $n \in [N_0+1, N]$, then $Q(\mathbf{\Theta}, \mathbf{\Theta}^{(i)})$ can be rewritten as:

$$\begin{aligned} Q(\mathbf{\Theta}, \mathbf{\Theta}^{(i)}) &= \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{M} \gamma_{nm}^{(i+1)} \big(2 \log \pi_m - \log |\mathbf{R}_m| \\ &- \|\mathbf{y}_n - \mathbf{e}_m\|_{\mathbf{R}_m}^2 - \log |\mathbf{U}_m| - \|\mathbf{x}_n - \mathbf{A}_m \mathbf{y}_n - \mathbf{b}_m\|_{\mathbf{U}_m}^2 \\ &- \log |\mathbf{V}_m| - \|\mathbf{z}'_{nm} - \mathbf{C}_m \mathbf{x}_n - \mathbf{d}_m\|_{\mathbf{V}_m}^2\big) \\ &- \sum_{m=1}^{M} \left(\sum_{n=N_0+1}^{N} \gamma_{nm}^{(i+1)} \right) \mathrm{Tr}[\mathbf{V}_m^{-1} \mathbf{V}_m^{(i)}]. \end{aligned} \quad (34)$$

B. M-step

In this subsection, we provide the M-step updates for the IC-GMR parameters. The details of the derivations are given in Appendix B. Three important properties of the update rules appear. First, they are all closed-form expressions, thus yielding to an intrinsically efficient EM algorithm. Second, the dependencies between the update rules do not form a loop. In other

[5] The one-to-one correspondence between the parameters of the "compact" GMM formulation (1) and the parameters of the "developed" formulation (29) is similar to the one given with (7).

words, we first update the parameters that are independent, to later on estimate the rest of them. Third, several auxiliary quantities are shared between different updates, so that calculating these quantities once for all saves computational power. Additionally, this allows to present the update rules more clearly, as follows.

Auxiliary variables are weighted sums of the observations and their outer-products:

$$S_m^{(i+1)} = \sum_{n=1}^N \gamma_{nm}^{(i+1)}, \quad S_{\mathbf{X},m}^{(i+1)} = \sum_{n=1}^N \gamma_{nm}^{(i+1)} \mathbf{x}_n,$$
$$\text{and} \quad S_{\mathbf{XX},m}^{(i+1)} = \sum_{n=1}^N \gamma_{nm}^{(i+1)} \mathbf{x}_n \mathbf{x}_n^\top. \tag{35}$$

The definition of the variables $S_{\mathbf{Y},m}^{(i+1)}$, $S_{\mathbf{Z}',m}^{(i+1)}$, $S_{\mathbf{XY},m}^{(i+1)}$, $S_{\mathbf{YY},m}^{(i+1)}$, $S_{\mathbf{Z}'\mathbf{X},m}^{(i+1)}$ and $S_{\mathbf{Z}'\mathbf{Z}',m}^{(i+1)}$ follows the same principle.

Priors: Maximization of $Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(i)})$ with respect to the priors is trivial, since it is identical to the GMM case [28] (with of course the responsibilities being calculated from the IC-GMR's PDF). For $m \in [1, M]$, we have:

$$\pi_m^{(i+1)} = \frac{1}{N} S_m^{(i+1)}. \tag{36}$$

Constant vectors and transition matrices: For $m \in [1, M]$, we have:

$$\mathbf{e}_m^{(i+1)} = \frac{1}{S_m^{(i+1)}} S_{\mathbf{Y},m}^{(i+1)}, \tag{37}$$

and $\mathbf{A}_m$, $\mathbf{b}_m$, $\mathbf{C}_m$ and $\mathbf{d}_m$ are updated with (38) and (39). Note that $\mathbf{A}_m^{(i+1)}$ and $\mathbf{b}_m^{(i+1)}$ have the form of the standard weighted-MSE estimates of $\mathbf{A}_m$ and $\mathbf{b}_m$ given the ($\mathbf{x}$, $\mathbf{y}$) dataset and using the responsibilities as weights. $\mathbf{C}_m^{(i+1)}$ and $\mathbf{d}_m^{(i+1)}$ have a similar form but take into account partially missing $\mathbf{z}$ data.

$$\begin{aligned}\mathbf{A}_m^{(i+1)} &= \left(S_{\mathbf{XY},m}^{(i+1)} - \frac{1}{S_m^{(i+1)}} S_{\mathbf{X},m}^{(i+1)} S_{\mathbf{Y},m}^{(i+1)\top} \right) \\ &\quad \times \left(S_{\mathbf{YY},m}^{(i+1)} - \frac{1}{S_m^{(i+1)}} S_{\mathbf{Y},m}^{(i+1)} S_{\mathbf{Y},m}^{(i+1)\top} \right)^{-1}, \\ \mathbf{b}_m^{(i+1)} &= \frac{1}{S_m^{(i+1)}} \left(S_{\mathbf{X},m}^{(i+1)} - \mathbf{A}_m^{(i+1)} S_{\mathbf{Y},m}^{(i+1)} \right)\end{aligned} \tag{38}$$

$$\begin{aligned}\mathbf{C}_m^{(i+1)} &= \left(S_{\mathbf{Z}'\mathbf{X},m}^{(i+1)} - \frac{1}{S_m^{(i+1)}} S_{\mathbf{Z}',m}^{(i+1)} S_{\mathbf{X},m}^{(i+1)\top} \right) \\ &\quad \times \left(S_{\mathbf{XX},m}^{(i+1)} - \frac{1}{S_m^{(i+1)}} S_{\mathbf{X},m}^{(i+1)} S_{\mathbf{X},m}^{(i+1)\top} \right)^{-1}, \\ \mathbf{d}_m^{(i+1)} &= \frac{1}{S_m^{(i+1)}} \left(S_{\mathbf{Z}',m}^{(i+1)} - \mathbf{C}_m^{(i+1)} S_{\mathbf{X},m}^{(i+1)} \right)\end{aligned} \tag{39}$$

Covariance matrices: For $m \in [1, M]$, we have:

$$\begin{aligned}\mathbf{R}_m^{(i+1)} &= \mathbf{e}_m^{(i+1)} \mathbf{e}_m^{(i+1)\top} \\ &\quad + \frac{\left(S_{\mathbf{YY},m}^{(i+1)} - S_{\mathbf{Y},m}^{(i+1)} * \mathbf{e}_m^{(i+1)} \right)}{S_m^{(i+1)}},\end{aligned} \tag{40}$$

$$\begin{aligned}\mathbf{U}_m^{(i+1)} &= \mathbf{b}_m^{(i+1)} \mathbf{b}_m^{(i+1)\top} \\ &\quad + \frac{1}{S_m^{(i+1)}} \Big(S_{\mathbf{XX},m}^{(i+1)} + \mathbf{A}_m^{(i+1)} S_{\mathbf{YY},m}^{(i+1)} \mathbf{A}_m^{(i+1)\top} \\ &\quad - S_{\mathbf{XY},m}^{(i+1)} * \mathbf{A}_m^{(i+1)} \\ &\quad - \left(S_{\mathbf{X},m}^{(i+1)} - \mathbf{A}_m^{(i+1)} S_{\mathbf{Y},m}^{(i+1)} \right) * \mathbf{b}_m^{(i+1)} \Big),\end{aligned} \tag{41}$$

$$\begin{aligned}\mathbf{V}_m^{(i+1)} &= \mathbf{d}_m^{(i+1)} \mathbf{d}_m^{(i+1)\top} \\ &\quad + \frac{1}{S_m^{(i+1)}} \Big(S_{\mathbf{Z}'\mathbf{Z}',m}^{(i+1)} + \mathbf{C}_m^{(i+1)} S_{\mathbf{XX},m}^{(i+1)} \mathbf{C}_m^{(i+1)\top} \\ &\quad - S_{\mathbf{Z}'\mathbf{X},m}^{(i+1)} * \mathbf{C}_m^{(i+1)} \\ &\quad - \left(S_{\mathbf{Z}',m}^{(i+1)} - \mathbf{C}_m^{(i+1)} S_{\mathbf{X},m}^{(i+1)} \right) * \mathbf{d}_m^{(i+1)} \Big) \\ &\quad + \mathbf{V}_m^{(i)} \sum\nolimits_{n=N_0+1}^N \gamma_{nm}^{(i+1)},\end{aligned} \tag{42}$$

where $\mathbf{P} * \mathbf{Q} = \mathbf{P}\mathbf{Q}^\top + \mathbf{Q}\mathbf{P}^\top$ denotes the symmetrized outer product of $\mathbf{P}$ and $\mathbf{Q}$. Interestingly, (37), (40), (38) and (41) correspond to the classical two-variable GMM, whereas (39) and (42) encode the effect of the missing data. Indeed, all statistics related to $\mathbf{Z}$ are computed using the actually observed $\mathbf{z}_n$ for $n \in [1, N_0]$ and the expected value $\mu_{\mathbf{Z}|\mathbf{x}_n,m}^{(i+1)}$ for $n \in [N_0{+}1, N]$.

C. Complete EM Algorithm

The main steps of the initialization of the EM algorithm have been given in Section V-C. We complete this description here by formalizing the initialization of the subset of parameters related to $\mathbf{Z}$, i.e. $\{\mathbf{C}_m, \mathbf{d}_m, \mathbf{V}_m\}_{m=1}^M$. Basically, this is done by evaluating (39) with the auxiliary variables involving $\mathbf{Z}$ being calculated using observed $\mathbf{z}$ data only, i.e. $\mathbf{z}_{1:N_0}$, corresponding $\mathbf{x}$ data, i.e. $\mathbf{x}_{1:N_0}$, and responsibilities for $n \in [1, N_0]$ given by (31). The complete EM algorithm for the IC-GMR, including the initialization step, is schematized in Algorithm 1. The E-step boils down to the calculation of the responsibilities (31) and (32). The M-step computes the auxiliary quantities defined in (35) that speed up the update of the parameters (36)–(42).

VII. Experiments

In this section, we describe the experiments we conducted to evaluate and compare the five adaptation techniques considered in this study (MAP-based and MLLR-based adaptation, D-GMR, SC-GMR and IC-GMR), and we discuss the results.

A. Data

The articulatory data of the reference speaker were recorded synchronously with the audio signal using the Carstens 2D EMA system (AG200). Six coils were glued on the tongue tip, blade, and dorsum, and on the upper lip, the lower lip and the jaw (the position of the velum was not recorded). 2D positions of the 6 coils were concatenated in 12-dimensional feature vectors. The sequences of articulatory feature vectors $\mathbf{y}$ were recorded at 200 Hz and downsampled to 100 Hz. The recorded database consists of two repetitions of 224 VCVs (Vowel-Consonant-Vowel sequences such as [apa], [ata], etc.), two repetitions of 109 pairs of CVC real French words (such "balle," "pomme," etc.), and 88 sentences. Silence and long pauses were removed from the data set, which finally contains 16 minutes of speech.

Algorithm 1 EM algorithm for integrated cascaded-GMR (IC-GMR) with partially missing data $\mathbf{Z}$.

```
$\boxed{\textbf{Initialization}}$
Use EM for GMM over $\{\mathbf{x}_n, \mathbf{y}_n\}_{n=1}^N$ to compute $\boldsymbol{\Theta}_{\mathbf{J}}{}^{\text{in}}$:
$\{\pi_m^{\text{in}}, \mathbf{e}_m^{\text{in}}, \mathbf{R}_m^{\text{in}}, \mathbf{A}_m^{\text{in}}, \mathbf{b}_m^{\text{in}}, \mathbf{U}_m^{\text{in}}\}_{m=1}^M$.
for $n := 1 : N_0$ do
    $\mathbf{z}'_{nm} = \mathbf{z}_n, \forall m.$
end
for $m := 1 : M$ do
    for $n := 1 : N_0$ do
        Set $\gamma_{nm}^{\text{in}}$ using (32) with $\boldsymbol{\Theta}_{\mathbf{J}}^{\text{in}}$.
    end
    $S_{\mathbf{Z}',m}^{\text{in}} = \sum_{n=1}^{N_0} \gamma_{nm}^{\text{in}} \mathbf{z}'_{nm}$
    $S_{\mathbf{Z}'\mathbf{X},m}^{\text{in}} = \sum_{n=1}^{N_0} \gamma_{nm}^{\text{in}} \mathbf{z}'_{nm} \mathbf{x}_n^\top$
    $S_{\mathbf{Z}'\mathbf{Z}',m}^{\text{in}} = \sum_{n=1}^{N_0} \gamma_{nm}^{\text{in}} \mathbf{z}'_{nm} (\mathbf{z}'_{nm})^\top$
end
Use (39), (42) with the previous auxiliary variables to
compute $\{\mathbf{C}_m^{\text{in}}, \mathbf{d}_m^{\text{in}}, \mathbf{V}_m^{\text{in}}\}_{m=1}^M$.
Set $\boldsymbol{\Theta}^{(0)} = \boldsymbol{\Theta}^{\text{in}}$ and $i = 1$.
while Not convergence do
    $\boxed{\textbf{E} - \textbf{step}}$
    for $n := 1 : N_0$ do
        $p(\mathbf{o}_n|\boldsymbol{\Theta}^{(i)}) = \sum_{m=1}^M p(\mathbf{o}_n, m|\boldsymbol{\Theta}_m^{(i)})$, using (11).
        for $m := 1 : M$ do
            $\gamma_{nm}^{(i)} = \frac{p(\mathbf{o}_n, m|\boldsymbol{\Theta}_m^{(i)})}{p(\mathbf{o}_n|\boldsymbol{\Theta}^{(i)})}$.
        end
    end
    for $n := N_0 + 1 : N$ do
        $p(\mathbf{j}_n|\boldsymbol{\Theta}^{(i)}) = \sum_{m=1}^M p(\mathbf{j}_n, m|\boldsymbol{\Theta}_{\mathbf{J},m}^{(i)})$, using (29).
        for $m := 1 : M$ do
            $\gamma_{nm}^{(i)} = \frac{p(\mathbf{j}_n, m|\boldsymbol{\Theta}_{\mathbf{J},m}^{(i)})}{p(\mathbf{j}_n|\boldsymbol{\Theta}_{\mathbf{J}}^{(i)})}$.
            $\mathbf{z}'_{nm} = \mathbf{C}_m^{(i)} \mathbf{x}_n + \mathbf{d}_m^{(i)}$.
        end
    end
    $\boxed{\textbf{M} - \textbf{step}}$
    for $m := 1 : M$ compute
        Auxiliary variables using (35)
        $\pi_m^{(i+1)}$ and $\mathbf{e}_m^{(i+1)}$ using (36) and (37)
        $\mathbf{A}_m^{(i+1)}$ and $\mathbf{b}_m^{(i+1)}$ using (38)
        $\mathbf{C}_m^{(i+1)}$ and $\mathbf{d}_m^{(i+1)}$ using (39)
        $\mathbf{R}_m^{(i+1)}$, $\mathbf{U}_m^{(i+1)}$ and $\mathbf{V}_m^{(i+1)}$ using (40), (41) and (42)
    end
    $i++$
end
```

In order to evaluate and compare the five adaptation techniques considered in this study, a second database of audio data only was recorded by two other speakers: one male (M1) and one female (F1). Both were asked to pronounce the same corpus as the one recorded by the reference speaker. All audio speech signals (from the reference speaker and the two source speakers) were sampled at 16 kHz. Audio feature vectors ($\mathbf{x}$ and $\mathbf{z}$ data) consisted of 13 MFCC coefficients extracted from 25 ms-frames (weighted with the Hamming window) every 10 ms (therefore, audio and articulatory vectors are synchronized at 100 Hz). To capture the dynamics of speech articulation, both acoustic and articulatory feature vectors were completed by their first derivative. This resulted in 26-dimensional acoustic vectors and 24-dimensional articulatory vectors).

B. *Evaluation Protocol*

A 5-fold cross-validation technique was employed for evaluation. For each source speaker (M1 and F1), the acoustic-articulatory database was divided into 5 subsets of approximate equal size, each one representing about 3 min of speech. At each trial, 4 subsets were used for training the reference speaker model (i.e. the $\mathbf{x}$-to-$\mathbf{y}$ inversion), and the remaining subset was used for test. Also at each trial, 10 experiments were conducted by varying the amount of adaptation data. Depending on the experiment, an adaptation subset was extracted randomly from the training set, with a size equal to $k/20$ of the size of the training set with $k \in [1, 10]$, i.e. approximately 0.7, 1.4, 2.1, 2.8, 3.4, 3.9, 4.5, 4.9, 5.3 and 6 min of speech signals. For all adaptation schemes, no significant improvement was observed for larger datasets. Therefore, we report here only the results obtained when using less than 6 min of adaptation data, out of the 16 min available for each source speaker. This results in 50 experiments, for each of the two source speakers M1 and F1 (i.e. 100 experiments in total).

C. *Metrics*

In a practical context, the articulatory movements estimated from the source speaker's acoustics occur in the vocal tract space of the reference speaker (displayed via a virtual talking head). For each test sequence, the original articulatory movements recorded on the reference speaker were therefore considered as the *target*.

Two metrics were used to evaluate the accuracy of the estimated articulatory trajectories. The first one is the Root Mean Squared Error (RMSE) between the articulatory feature vectors of the reference speaker and those estimated from the source speaker's acoustics. A paired t-test was used to estimate a 95%-confidence interval for each RMSE measure. For each experiment (i.e. for each size of adaptation dataset), paired t-test were also used to determine if the performances given by two adaptation techniques were significantly different from each other. Before calculating the RMSE, a DTW-based procedure was used to align the signals of reference and source speakers. In contrast to the training step, we here warped the audio signals produced by the reference speaker onto the audio signals produced by the source speaker, and then we warped the articulatory movements of the reference speaker accordingly. This way, the articulatory movements of the reference speaker matched the speech rate of the source speaker. This enables the generation of audiovisual sequences of the talking head displaying the estimated articulatory trajectories synchronously with the voice of the source speaker. Such sequences are provided with this paper as supplementary material.

The second metric used for evaluation was derived from the so-called *articulatory recognition* paradigm [7], [31], [32]. This metric aims at evaluating the estimated trajectory at the phonetic level and can be summarized as follows. First, a HMM-based phonetic decoder was trained on the articulatory data of the reference speaker. A standard training procedure based on triphone modeling was used (with a 3-state left-to-right HMM, and a tree-based state-tying strategy, using the HTK toolkit [33]). For each test utterance, the Viterbi algorithm was then used to decode at phonetic level the articulatory trajectories estimated from the source speaker's acoustics. In order to alleviate the problem of insertion/deletion errors due to the absence of a language model, this evaluation procedure was used only on VCV and CVC sequences (the decoder being forced to recognize VCV and CVC only). The phone error rate (PER) was used as a measure of the accuracy of the estimated articulatory trajectories. It is here defined as $\text{PER} = 100{*}((N_p - S_p)/N_p)$, where N_p is the total number of phones in the test set, and S_p is the number of substitution errors. The 95% -confidence interval of each PER measure was defined as the Wilson score interval. For each experiment, statistical significance between two adaptation methods in term of PER was assessed using the non-parametric Wilcoxon test (which was preferred to paired t-test because of the non-Gaussian form of errors distribution).

D. Implementation Details

The number of mixture components of the different models were optimized at each trial of the 5-fold cross-validation, using a subset of the training set. As for the reference acoustic-articulatory GMR, we tested $M \in \{16, 32, 64, 128, 256\}$. In most experiments, the optimal number was found to be 64 or 128. Since the performance obtained with these two values were very close, we selected $M = 64$ to limit the number of parameters for the different models (we recall that all GMR and C-GMR models involve full covariance matrices). As for the first stage of the SC-GMR (i.e. the spectral conversion GMR) and the D-GMR, and for each size of the adaptation dataset, a cross-validation procedure was used to determine the best number of mixture components K among 8,16,32 and 64. For D-GMR, SC-GMR and IC-GMR, the number of EM iteration was fixed empirically to 50. In practice, we observed no significant evolution of model parameters with more iterations. As for the MLLR-based adaptation scheme, we adopted the formulation of [34]. Best performance was obtained when adapting $\mu_{\mathbf{X},m}$ and keeping the original value of $\mathbf{\Sigma}_{\mathbf{XX},m}$ and prior π_m, for each component m. As for the MAP-based adaptation scheme, both $\mu_{\mathbf{X},m}$ and $\mathbf{\Sigma}_{\mathbf{XX},m}$ were adapted, but best performance was obtained when keeping the original value of π_m.

E. Results

First, we report the performance of the acoustic-to-articulatory inversion by the reference $\mathbf{X}$-$\mathbf{Y}$ GMR, using reference speaker speech signals as inputs. This provides an upper bound for the five adaptation schemes considered in this study (i.e. the best possible result). In terms of RMSE, we obtained an error of 1.8 mm using GMR-MSE, and 1.5 mm using GMR-MLPG. These results are consistent with the literature on acoustic-articulatory mapping, e.g. [12], [15]. In terms of PER, we obtained 6% for the GMR-MSE, and 3.1% for the GMR-MLPG.

On the other extreme side, we report the performance obtained by the reference speaker $\mathbf{X}$-$\mathbf{Y}$ GMR (in MLPG implementation) when processing speech inputs $\mathbf{z}$ from source speakers F1 and M1, with no adaptation. This provides a lower bound for the five adaptation schemes (i.e. the worst possible results). As expected, the performance decreased drastically compared with the upper bound, with $\text{RMSE} = 3.8$ mm/$\text{PER} = 67\%$ for speaker M1, and $\text{RMSE} = 4.4$ mm/$\text{PER} = 80\%$ for speaker F1 (we recall that speaker M1 and reference speaker are male, whereas speaker F1 is a female). These results confirm the strong need to adapt the reference speaker model.

Let us now present the results obtained for the source speakers M1 and F1, for all adaptation schemes (MLLR, MAP, D-GMR, SC-GMR and IC-GMR). Fig. 4 and 5 show the RMSE and PER obtained for speakers M1 and F1 respectively, as a function of the amount of adaptation data, and for both MSE and MLPG implementations. These results can be discussed from different perspectives.

First of all, the adaptation of the reference model to the source speaker's acoustics drastically reduces both RMSE and PER. As an example, let us consider the IC-GMR technique in MSE implementation when considering 2 min of adaptation data. Compared to the lower bound, the relative improvement for speaker M1 is 29% for RMSE and 50% for PER (with 2.7 mm RMSE and 32% PER). For speaker F1, it is 34% for RMSE and 57% for PER (with 2.9 mm RMSE and 34% PER). This shows the global efficiency of the proposed C-GMR framework.

As expected, the MLPG implementation of each adaptation scheme systematically outperforms the corresponding MSE implementation (solid vs. dashed curves), for both source speakers, and all sizes of adaptation dataset. As mentioned in [15], the statistical smoothing of MLPG is of particular interest for the acoustic-to-articulatory inversion given the slow-varying nature of the EMA data. Using MLPG, the mapping is achieved utterance-by-utterance and not frame-by-frame as in MSE implementation. All acoustic feature vectors of an input sequence $\mathbf{z}$ contribute to the estimation of each output vector $\mathbf{y}_t$. As a consequence, the mapping can benefit from contextual information, which is also important to tackle the ill-posed nature of acoustic-articulatory inversion. However, in its standard implementation, MLPG estimation can not be done in real-time, which remains an important issue for the present system.

Let us now describe in more details the RMSE results for the five adaptation schemes (Fig. 4). In all experiments, MAP-based adaptation significantly outperforms MLLR. As a possible explanation, let us recall that MLLR imposes the same transformation to all GMM components, whereas MAP updates each component separately, leading potentially to better flexibility and accuracy. Experimental results show very distinct error patterns between MAP and MLLR on the one hand, and D-GMR, SC-GMR and IC-GMR on the other hand. Surprisingly, the performance of MAP and MLLR are quite stable with the size of the

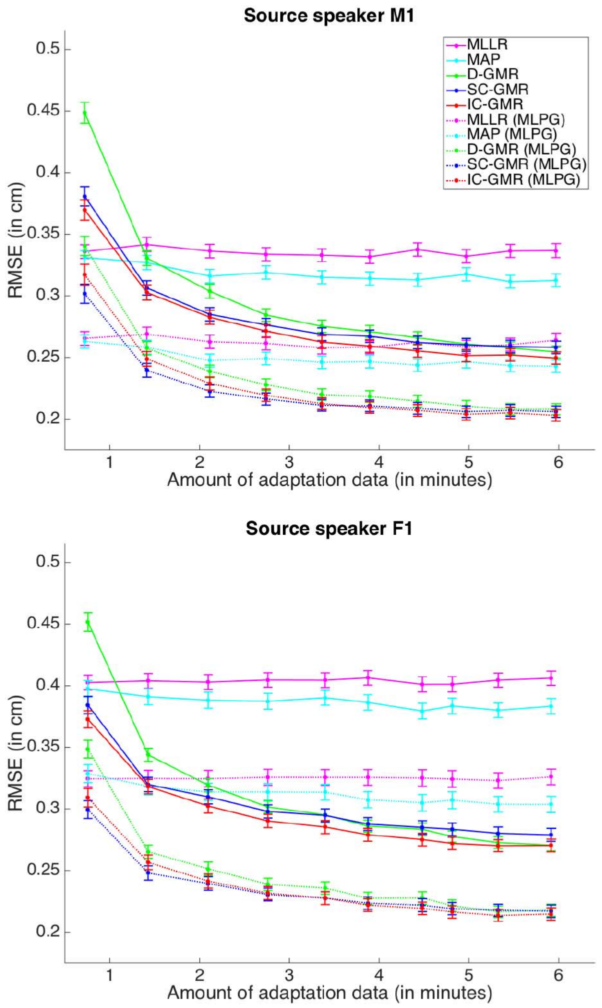

Fig. 4. Performance obtained with MLLR, MAP, D-GMR, SC-GMR and IC-GMR (for both MSE and MLPG implementations) as a function of the amount of adaptation data, in terms of RMSE, for speaker M1 (top) and F1 (bottom). Error bars represent 95% confidence intervals of RMSE.

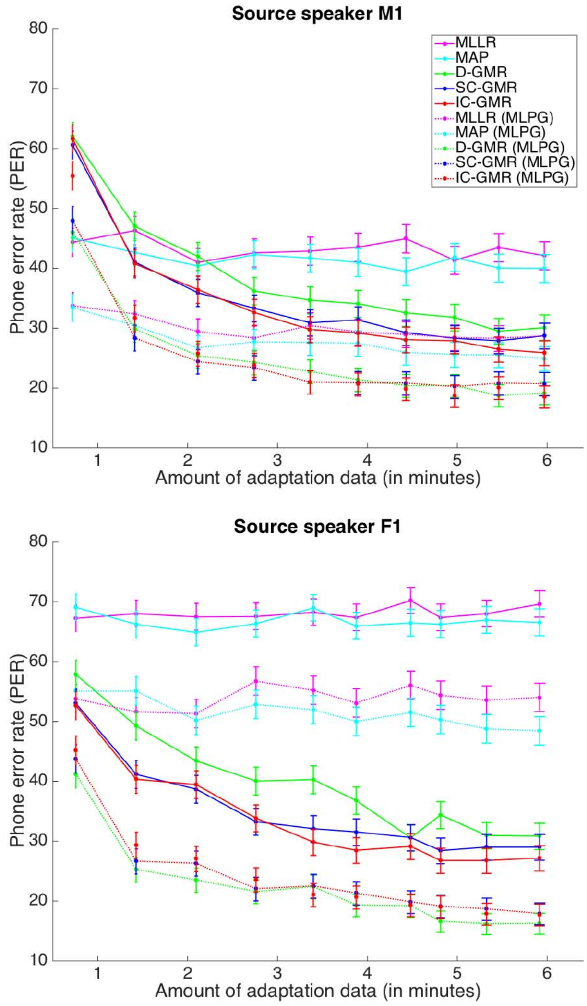

Fig. 5. Performance obtained with MLLR, MAP, D-GMR, SC-GMR and IC-GMR (for both MSE and MLPG implementations) as a function of the amount of adaptation data, in terms of phone error rate, for speaker M1 (top) and F1 (bottom). Error bars represent 95% confidence intervals of PER.

adaptation dataset. Best performance is almost already obtained with less than 1 min of adaptation data (training with a smaller size was not feasible). As already mentioned in Section IV-A, this may be explained by a potential mismatch between the adapted acoustic parameters and the original acoustic-articulatory parameters of the $\mathbf{X}$-$\mathbf{Y}$ reference GMR. In contrast, performances of D-GMR, SC-GMR and IC-GMR increase with the size of the adaptation dataset (for both MSE and MLPG implementations).

Now we detail the differences between D-GMR, SC-GMR and IC-GMR. In MSE implementation, SC-GMR and IC-GMR significantly outperform D-GMR for less than 2.1 min of adaptation data for speaker M1 and 2.7 min for speaker F1. For larger adaptation datasets, D-GMR, SC-GMR and IC-GMR give similar performances. As already mentioned in Section IV-A, D-GMR exploits only the reference speaker's articulatory data that can be associated with the source speaker's audio data. Hence, training the D-GMR on a limited dataset leads to low performances. This result tends to validate the proposed C-GMR models (SC-GMR and IC-GMR). We recall that these techniques exploit all available data from the reference speaker when training the acoustic-articulatory model. Regarding the MLPG implementations, the differences between D-GMR on the one hand and SC-GMR/IC-GMR on the other hand are smaller compared to the MSE implementation (for both source speakers). The smoothing effect of MLPG seems to alleviate the impact of errors on the estimated articulatory targets.

We now compare the performance of SC-GMR and IC-GMR. As shown in Fig. 4, IC-GMR outperforms slightly SC-GMR for MSE implementation in RMSE terms (i.e. the solid red curve is most often slightly below the solid blue curve). In order to test if the difference between the two techniques was statistically significant over all experiments (i.e. not for a specific size of adaptation dataset), we conducted a 3-way ANOVA test based on a mixed model (using the *lme* package of the R software). The variable to explain was the $RMSE$ whereas the explicative variables were: a 2-level categorical factor *method* (SC-GMR/IC-GMR), another 2-level categorical factor *speaker* (M1/F1), a 10-level categorical factor for the size of the adaptation corpus, and a random effect of the *test sentence* to account for data pairing. For both MSE and MLPG implementations, the factor *method* did not interact significantly with any of the two other factors. For MSE implementation, its effect was significant on the RMSE measure which is lower for IC-GMR compared to SC-GMR, by 0.008 with $F(1, 21976) = 221$ and $p < 0.001$. Therefore, IC-GMR outperforms slightly, but significantly, SC-GMR for MSE implementation. An opposite tendency was observed for MLPG implementation. Here, SC-GMR outperforms slightly IC-GMR by 0.002, but this main effect is weakly significant ($F(1, 21976) = 10$ and $p = 0.0016$). Therefore, performances of SC-GMR and IC-GMR in MLPG implementation should be considered as equivalent.

Similar general trends are observed for the PER (Fig. 5). As expected, the curves are less regular than for RMSE since a difference in terms of RMSE does not necessarily lead to a phonetic decoding error (in other words, PER metric is less fine-grained than RMSE). For MSE implementation, SC-GMR and IC-GMR outperform D-GMR almost always significantly. Similarly to RMSE results, IC-GMR outperforms slightly SC-GMR. However, the difference is here almost never significant. Again, the MLPG implementations of D-GMR, SC-GMR and IC-GMR (dashed curves) lead to similar performances (with 20% PER in average).

F. Discussion

To summarize the previous results, the proposed C-GMR framework most often outperforms the other adaptation techniques when considering the conventional MSE estimator. This is true for both RMSE and PER metrics, for both source speakers M1 and F1, and more importantly for small adaptation datasets. A good trade-off between the amount of adaptation data and the performance can be found between 1.5 and 2.5 min, depending on the speaker and on the implementation (MSE or MLPG). The performance for small datasets validates the benefit of introducing an intermediate spectral conversion stage (i.e. $\mathbf{Z}$-to-$\mathbf{X}$) in the acoustic-articulatory inversion process. This allows the model to rely on a well-estimated acoustic-articulatory model of the reference speaker.

For both SC-GMR and IC-GMR, the performances decrease significantly for very small datasets (say, less that 1 min). For both techniques, we can conjecture different explanations for this result. As concern the SC-GMR, one characteristic of this model is its flexibility, in the sense that the number of components of the $\mathbf{Z}$-to-$\mathbf{X}$ GMR can be set independently from the $\mathbf{X}$-to-$\mathbf{Y}$ GMR. Consequently, the $\mathbf{Z}$-to-$\mathbf{X}$ model can adapt to the structure of the adaptation set. This is suitable when dealing with a dense adaptation set. However, it can be problematic for a very small and potentially too sparse dataset. Indeed, some regions of source and reference speaker's acoustic space which are not represented in the adaption dataset may not be correctly covered by the model.

This problem is theoretically tackled by the IC-GMR. As an example, let us consider a component of the $\mathbf{X}$-to-$\mathbf{Y}$ GMR for which no corresponding $\mathbf{z}$ observation is available in the adaptation dataset. Thanks to the integrated structure of the IC-GMR, this component should be preserved and even slightly adapted using the estimated missing data (as detailed in Section VI). Therefore, the accuracy of the mapping in this area of the acoustic space will depend only on the distance of the $\mathbf{z}$ observation to the mean of the considered component. Therefore, the IC-GMR should achieve better generalization than the SC-GMR. However, our experiments did not confirm such property for the smallest adaptation datasets. Among the possible limitations of the IC-GMR, we can conjecture two. For very small adaptation datasets (i.e. $N_0 \ll N$), the amount of data available to correctly bootstrap the proposed IC-GMR training algorithm may not be sufficient: in the initialization stage, the statistics $S^{\text{in}}_{\mathbf{Z}',m}$, $S^{\text{in}}_{\mathbf{Z}'\mathbf{X},m}$ and $S^{\text{in}}_{\mathbf{Z}'\mathbf{Z}',m}$ are calculated from available adaptation data $\mathbf{z}_{1:N_0}$ and $\mathbf{x}_{1:N_0}$. This may deliver poorly reliable estimations of parameters $\mathbf{C}^{\text{in}}_m$, $\mathbf{d}^{\text{in}}_m$ and $\mathbf{V}^{\text{in}}_m$, and thus poorly estimated missing data $\mathbf{z}'$ and so forth in the following EM iterations. In other words, there may exist a limit on the amount of adaptation data above which the proposed EM algorithm and associated initialization work as a virtuous circle. A second limitation could be related to the ratios between data amount and number of model parameters. The number of free parameters that have to be estimated in any of the considered training processes is significantly lower for the SC-GMR than for the IC-GMR. Indeed, the SC-GMR is a chain of two independent 2-vector GMRs, whereas the IC-GMR is basically a 3-vector GMR. When the amount of training/adaptation data gets too limited, models with a larger number of parameters are generally penalized. This is a remaining issue of the proposed IC-GMR framework which should be addressed in future work.

VIII. Conclusion

This article addresses the problem of how to adapt an acoustic-articulatory GMR trained on a reference speaker, to another (source) speaker, using a limited amount of audio-only speech data. First, we investigated standard adaptation techniques for GMM such as MLLR and MAP, to modify the acoustic component of the acoustic-articulatory GMR. We tested also the performance of a standard GMR, which models directly the statistical relationships between the source speaker's acoustics, and the reference speaker's articulation (referred to as D-GMR). Then, we introduced a new general framework called cascaded Gaussian mixture regression. This approach aims notably at exploiting all information available about the acoustic-articulatory relationship of the reference speaker. To that purpose, it decomposes the conversion process in two steps, 1) a spectral mapping step which models the statistical relationships between source and reference speaker's acoustics, and 2) an acoustic-articulatory inversion step. Two versions of the C-GMR have been proposed. The first one is a straightforward chaining of two GMRs (SC-GMR), achieving respectively the spectral mapping and the acoustic-articulatory inversion. The second one (IC-GMR) integrates the two regressions in a single joint probabilistic model. The EM algorithm associated to the IC-GMR has been derived within the framework of missing data to deal with limited adaptation datasets. In line with the existing literature on conventional GMR, we derived two mapping procedures for the IC-GMR, based respectively on MSE estimator and MLPG algorithm (the latter including an explicit smoothness constraint).

Experiments have shown that both SC-GMR and IC-GMR outperform MAP, MLLR and D-GMR, and were able to recover phonetically consistent articulatory trajectories, from as few as 3 min of adaptation data. Besides, IC-GMR outperformed slightly (but significantly) SC-GMR in MSE implementation.

Further experiments should be conducted in order to validate the proposed system in a realistic applicative context. To that purpose, the performance of the C-GMR framework when adapting to disordered or non-native speech should be evaluated. In such cases, the pronunciation differences between source and reference speakers can be notable. This may occur when the reference speaker's language contains a phoneme which does not exist in the source speaker's language, or when the source speaker's production is altered by an articulatory disorder. To address this challenge, the proposed training algorithm of IC-GMR should be extended to the case of non-parallel corpus as considered in [35]. Moreover, the use of the C-GMR framework (and notably the IC-GMR approach) could be envisioned in other speech processing areas, such as *silent speech interfaces* [36] which are devices converting speech-related biosignals (e.g. articulatory movements, electrical activity of face muscles, etc.) into audible speech. The C-GMR techniques could be used to adapt an articulatory-to-acoustic GMR trained with vocalized data but used with silent speech and possible altered articulation [37].

IX. Supplementary Material

A video showing the articulatory talking head animated automatically from the speech audio signal of speakers F1/M1, for different VCV sequences, using SC-GMR and IC-GMR techniques is provided as supplementary material. The MATLAB source code of IC-GMR training and mapping algorithms is available at http://www.gipsa-lab.fr/~thomas.hueber/cgmr/.

Appendix A
Derivation of $Q(\mathbf{\Theta}, \mathbf{\Theta}^{(i)})$

Q is classically computed by taking the expectation of the complete-data log-likelihood with respect to the posterior distribution of the hidden variables given the observations (and the parameters at previous iteration):

$$\begin{aligned} Q\left(\mathbf{\Theta}, \mathbf{\Theta}^{(i)}\right) &= \sum_{n=1}^{N_0}\sum_{m=1}^{M} p\left(m|\mathbf{o}_n, \mathbf{\Theta}^{(i)}\right) \\ &\times \log p(\mathbf{o}_n, m|\mathbf{\Theta}_m) \\ &+ \sum_{n=N_0+1}^{N}\sum_{m=1}^{M}\int_{\mathbb{R}^{D_Z}} p\left(\mathbf{z}_n, m|\mathbf{j}_n, \mathbf{\Theta}^{(i)}\right) \\ &\times \log p(\mathbf{o}_n, m|\mathbf{\Theta}_m)\mathrm{d}\mathbf{z}_n. \end{aligned}$$

With definition (31) and multiplying and dividing the terms of the second double sum by $p(\mathbf{j}_n, \mathbf{\Theta}^{(i)})$, (30) follows immediately. Injecting (11)–(15) into the first double sum of (30) leads to the first double sum of (33). As for the second double sum, we remark that:

$$\begin{aligned} &\int_{\mathbb{R}^{D_Z}} p\left(\mathbf{o}_n, m|\mathbf{\Theta}_m^{(i)}\right) \log p(\mathbf{o}_n, m|\mathbf{\Theta}_m)\mathrm{d}\mathbf{z}_n \\ &= p\left(\mathbf{j}_n, m|\mathbf{\Theta}_m^{(i)}\right)\left[\log p\left(\mathbf{j}_n, m|\mathbf{\Theta}_m\right)\right. \\ &\quad + \int_{\mathbb{R}^{D_Z}} p\left(\mathbf{z}_n, m|\mathbf{j}_n, \mathbf{\Theta}_m^{(i)}\right) \\ &\quad \left.\times \log p\left(\mathbf{z}_n, m|\mathbf{j}_n, \mathbf{\Theta}_m\right)\mathrm{d}\mathbf{z}_n\right]. \end{aligned}$$

Factor $p(\mathbf{j}_n|\mathbf{\Theta}^{(i)})$ together with $p(\mathbf{j}_n, m|\mathbf{\Theta}_m^{(i)})$ form the responsibilities (32), and the integral term is responsible for the term $-\|\mathbf{C}_m^{(i)}\mathbf{x}_n + \mathbf{d}_m^{(i)} - \mathbf{C}_m\mathbf{x}_n - \mathbf{d}_m\|^2_{\mathbf{V}_m} - \mathrm{Tr}[\mathbf{V}_m^{-1}\mathbf{V}_m^{(i)}]$ of (33), that is equivalent in the case of missing data to the term $-\|\mathbf{z}_n - \mathbf{C}_m\mathbf{x}_n - \mathbf{d}_m\|^2_{\mathbf{V}_m}$ present in the first double sum of (33).

Appendix B
Maximization of $Q(\mathbf{\Theta}, \mathbf{\Theta}^{(i)})$

In this appendix we present the derivations for the M-step. All formulas start by taking the derivative of Q as expressed in (34).

Constant vectors and transition matrices: For $m \in [1, M]$, we have:

$$\frac{\partial Q(\mathbf{\Theta}, \mathbf{\Theta}^{(i)})}{\partial \mathbf{e}_m} = \mathbf{R}_m^{-1}\sum_{n=1}^{N}\gamma_{nm}^{(i+1)}(\mathbf{y}_n - \mathbf{e}_m).$$

Setting this expression to zero leads to:

$$\mathbf{e}_m = \frac{\sum_{n=1}^{N}\gamma_{nm}^{(i+1)}\mathbf{y}_n}{\sum_{n=1}^{N}\gamma_{nm}^{(i+1)}}, \tag{43}$$

from which we obtain (37). This expression is very similar to the classical GMM case (see [28]), except for the specific definition of the responsibilities for $n \in [N_0 + 1, N]$. In the same line, taking the derivative of $Q(\mathbf{\Theta}, \mathbf{\Theta}^{(i)})$ with respect to $\mathbf{b}_m$ and setting the result to zero leads to:

$$\mathbf{b}_m = \frac{\sum_{n=1}^{N}\gamma_{nm}^{(i+1)}(\mathbf{x}_n - \mathbf{A}_m\mathbf{y}_n)}{\sum_{n=1}^{N}\gamma_{nm}^{(i+1)}}. \tag{44}$$

Besides, for $m \in [1, M]$, we have:

$$\frac{\partial Q(\mathbf{\Theta}, \mathbf{\Theta}^{(i)})}{\partial \mathbf{A}_m} = \mathbf{U}_m^{-1}\sum_{n=1}^{N}\gamma_{nm}^{(i+1)}(\mathbf{x}_n - \mathbf{A}_m\mathbf{y}_n - \mathbf{b}_m)\mathbf{y}_n^\top.$$

Setting this expression to zero leads to:

$$\begin{aligned} \mathbf{A}_m = &\left(\sum_{n=1}^{N}\gamma_{nm}^{(i+1)}(\mathbf{x}_n - \mathbf{b}_m)\mathbf{y}_n^\top\right) \\ &\times \left(\sum_{n=1}^{N}\gamma_{nm}^{(i+1)}\mathbf{y}_n\mathbf{y}_n^\top\right)^{-1}. \end{aligned} \tag{45}$$

With the notation introduced in (35), Equ. (44) and (45) write:

$$\mathbf{b}_m = \frac{1}{S_m^{(i+1)}}\left(S_{\mathbf{X},m}^{(i+1)} - \mathbf{A}_m S_{\mathbf{Y},m}^{(i+1)}\right) \tag{46}$$

and

$$\mathbf{A}_m = \left(S_{\mathbf{XY},m}^{(i+1)} - \mathbf{b}_m S_{\mathbf{Y},m}^{(i+1)\top}\right) S_{\mathbf{YY},m}^{(i+1)-1}. \tag{47}$$

Replacing (46) into (47) we can deduce the final result for $\mathbf{A}_m$ and $\mathbf{b}_m$ given by (38)[6]. The optimal expression for $\mathbf{C}_m$ and $\mathbf{d}_m$ in (39) are obtained in the same manner.

[6] Alternately one can solve for $\mathbf{A}_m$ first and place the result in (46) to obtain $\mathbf{b}_m$. The two solutions are equivalent, including in terms of computational cost.

Covariance matrices: For $m \in [1, M]$, we have:

$$\frac{\partial Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(i)})}{\partial \mathbf{R}_m^{-1}} = \frac{1}{2}\sum_{n=1}^{N} \gamma_{nm}^{(i+1)} \times \left(\mathbf{R}_m - (\mathbf{y}_n - \mathbf{e}_m)(\mathbf{y}_n - \mathbf{e}_m)^\top\right).$$

Setting this expression to zero leads to:

$$\begin{aligned}\mathbf{R}_m &= \frac{1}{\sum_{n=1}^{N}\gamma_{nm}^{(i+1)}} \sum_{n=1}^{N} \gamma_{nm}^{(i+1)} (\mathbf{y}_n - \mathbf{e}_m)(\mathbf{y}_n - \mathbf{e}_m)^\top \\ &= \frac{1}{S_m^{(i+1)}} \left(S_{\mathbf{YY},m}^{(i+1)} - S_{\mathbf{Y},m}^{(i+1)} * \mathbf{e}_m + \mathbf{e}_m \mathbf{e}_m^\top \right).\end{aligned}$$

We recall that $\mathbf{P} * \mathbf{Q} = \mathbf{P}\mathbf{Q}^\top + \mathbf{Q}\mathbf{P}^\top$ denotes the symmetrized outer product of $\mathbf{P}$ and $\mathbf{Q}$. From these equations the result in (40) follows immediately. In the same line, taking the derivative of $Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(i)})$ with respect to $\mathbf{U}_m^{-1}$ and setting the result to zero leads to:

$$\mathbf{U}_m = \frac{1}{S_m^{(i+1)}} \sum_{n=1}^{N} \gamma_{nm}^{(i+1)} (\mathbf{x}_n - \mathbf{A}_m \mathbf{y}_n - \mathbf{b}_m) \times (\mathbf{x}_n - \mathbf{A}_m \mathbf{y}_n - \mathbf{b}_m)^\top,$$

which drives us to (41). These expressions are of course empirical covariance matrices weighted by specific responsibilities. As for the maximization of $Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(i)})$ with respect to $\mathbf{V}_m$, we have the additional contribution of the Trace term due to the missing data. Setting the corresponding derivative to zero yields:

$$\begin{aligned}\mathbf{V}_m = \frac{1}{S_m^{(i+1)}} \Bigg(\left(\sum_{n=N_0+1}^{N} \gamma_{nm}^{(i+1)} \right) \mathbf{V}_m^{(i)} &+ \sum_{n=1}^{N} \gamma_{nm}^{(i+1)} (\mathbf{z}'_{nm} - \mathbf{C}_m \mathbf{x}_n - \mathbf{d}_m) \\ &\times (\mathbf{z}'_{nm} - \mathbf{C}_m \mathbf{x}_n - \mathbf{d}_m)^\top \Bigg).\end{aligned}$$

The second term on the right side is an empirical covariance matrix and, again, it is similar to the classical GMM case [28] except for the specific definition of observation vectors and responsibilities for $n \in [N_0 + 1, N]$. The first term accounts for the missing data, i.e. $\mathbf{z}_n$ for $n \in [N_0 + 1, N]$. From this last equation (42) follows easily.

Acknowledgment

The authors would like to thank Pierre Badin and Frédéric Elisei for their work on the articulatory talking head and Christophe Savariaux for his help on the EMA recordings.

References

[1] K. Livescu, “Feature-based pronunciation modeling for automatic speech recognition,” Ph.D. dissertation, MIT Dept. of Elect. Eng. and Comput. Sci., Mass. Inst. of Technol., , 2005.

[2] S. Chennoukh, D. Sinder, G. Richard, and J. Flanagan, “Articulatory based low bit-rate speech coding,” *J. Acoust. Soc. Amer.*, vol. 102, no. 5, pp. 3163–3163, 1997.

[3] Z.-H. Ling, K. Richmond, J. Yamagishi, and R.-H. Wang, “Integrating articulatory features into HMM-based parametric speech synthesis,” *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, no. 6, pp. 1171–1185, Aug. 2009.

[4] P. Badin, F. Elisei, G. Bailly, and Y. Tarabalka, “An audiovisual talking head for augmented speech generation: Models and animations based on a real speaker articulatory data,” in *Articulated Motion and Deformable Objects*, ser. Lecture Notes in Computer Science, F. Perales and R. Fisher, Eds. Berlin/Heidelberg, Germany: Springer , 2008, vol. 5098, pp. 132–143.

[5] D. W. Massaro, S. Bigler, T. H. Chen, M. Perlman, and S. Ouni *et al.*, “Pronunciation training: The role of eye and ear,” in *Proc. Interspeech*, Brisbane, Australia, 2008, pp. 2623–2626.

[6] S. Fagel and K. Madany, “A 3-D virtual head as a tool for speech therapy for children,” in *Proc. Interspeech*, Brisbane, Australia, 2008, pp. 2643–2646.

[7] O. Engwall, “Analysis of and feedback on phonetic features in pronunciation training with a virtual teacher,” *Comput. Assisted Lang. Learn.*, vol. 25, pp. 37–64, 2012.

[8] B. S. Atal, J. Chang, M. V. Mathews, and J. W. Tukey, “Inversion of articulatory-to-acoustic transformation in the vocal tract by a computer-sorting technique,” *J. Acoust. Soc. Amer.*, vol. 63, no. 5, pp. 1535–1555, 1978.

[9] J. Hogden, A. Lofqvist, V. Gracco, I. Zlokarnik, P. Rubin, and E. Saltzman, “Accurate recovery of articulator positions from acoustics: New conclusions based on human data,” *J. Acoust. Soc. Amer.*, vol. 100, no. 3, pp. 1819–1834, 1996.

[10] S. Ouni and Y. Laprie, “Modeling the articulatory space using a hypercube codebook for acoustic-to-articulatory inversion,” *J. Acoust. Soc. Amer.*, vol. 118, no. 1, pp. 444–460, 2005.

[11] M. Rahim, W. Keijn, J. Schroeter, and C. Goodyear, “Acoustic to articulatory parameter mapping using an assembly of neural networks,” in *Proc. ICASSP*, 1991, pp. 485–488.

[12] K. Richmond, “Estimating articulatory parameters from the acoustic speech signal,” Ph.D. dissertation, Univ. of Edinburgh, Edinburgh, U.K., 2002.

[13] A. Toutios and K. G. Margaritis, “A support vector approach to the acoustic-to-articulatory mapping,” in *Proc. Interspeech*, Lisbon, Portugal, 2005, pp. 3221–3224.

[14] G. Ananthakrishnan and O. Engwall, “Mapping between acoustic and articulatory gestures,” *Speech Commun.*, vol. 53, no. 4, pp. 567–589, 2011.

[15] T. Toda, A. W. Black, and K. Tokuda, “Statistical mapping between articulatory movements and acoustic spectrum using a gaussian mixture model,” *Speech Commun.*, vol. 50, no. 3, pp. 215–227, 2008.

[16] S. Hiroya and M. Honda, “Estimation of articulatory movements from speech acoustics using an HMM-based speech production model,” *IEEE Trans. Speech Audio Process.*, vol. 12, no. 2, pp. 175–185, Mar. 2004.

[17] H. Zen, Y. Nankaku, and K. Tokuda, “Continuous stochastic feature mapping based on trajectory HMMs,” *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 2, pp. 417–430, Feb. 2011.

[18] S. Dusan and L. Deng, “Vocal-tract length normalization for acoustic-to-articulatory mapping using neural networks,” in *Proc. 138th Meeting Acoust. Soc. Amer.*, 1999, vol. 106, no. 4, p. 2181.

[19] S. Hiroya and M. Honda, “Speaker adaptation method for acoustic-to-articulatory inversion using an HMM-based speech production model,” *IEICE Trans. Inf. Syst.*, vol. 87, no. 5, pp. 1071–1078, 2004.

[20] J.-L. Gauvain and C.-H. Lee, “Maximum *a posteriori* estimation for multivariate gaussian mixture observations of Markov chains,” *IEEE Trans. Speech Audio Process.*, vol. 2, no. 2, pp. 291–298, Apr. 1994.

[21] C. J. Leggetter and P. Woodland, “Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models,” *Comput. Speech Lang.*, vol. 9, no. 2, pp. 171–185, 1995.

[22] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the EM algorithm,” *J. Roy. Statist. Soc., Ser. B (Methodological)*, pp. 1–38, 1977.

[23] Z. Ghahramani and M. I. Jordan, “Learning from incomplete data,” Cambridge, MA, USA, Tech. Rep., 1994.

[24] Z. Ghahramani and M. I. Jordan, “Supervised learning from incomplete data via an EM approach,” in *Advances in Neural Information Processing Systems*. Burlington, MA, USA: Morgan Kaufmann, 1994, vol. 6, pp. 120–127.

[25] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, “Speech parameter generation algorithms for HMM-based speech synthesis,” in *Proc. IEEE ICASSP*, Istanbul, Turkey, 2000, vol. 3, pp. 1315–1318.

[26] T. Toda, A. W. Black, and K. Tokuda, “Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory,” *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 8, pp. 2222–2235, Nov. 2007.

[27] T. Hueber, G. Bailly, P. Badin, and F. Elisei, "Speaker adaptation of an acoustic-articulatory inversion model using cascaded gaussian mixture regressions," in *Proc. Interspeech*, Lyon, France, 2013, pp. 2753–2757.

[28] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. New York, NY, USA: Springer-Verlag, 2006.

[29] Z. Ghahramani and M. I. Jordan, "Supervised learning from incomplete data via an EM approach," in *Advances in neural information processing systems*, J. D. Cowan, G. Tesauro, and J. Alspector, Eds. San Fransico, CA, USA: Morgan Kaufmann, 1994, vol. 6, pp. 120–127, Ed..

[30] Y. Stylianou, O. Cappé, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Trans. Speech Audio Process.*, vol. 6, no. 2, pp. 131–142, Mar. 1998.

[31] T. Hueber, A. Ben Youssef, G. Bailly, P. Badin, and F. Elisei, "Cross-speaker acoustic-to-articulatory inversion using phone-based trajectory HMM for pronunciation training," in *Proc. Interspeech*, Portland, OR, USA, 2012.

[32] A. Ben Youssef, P. Badin, G. Bailly, and P. Heracleous, "Acoustic-to-articulatory inversion using speech recognition and trajectory formation based on phoneme hidden Markov models," in *Proc. Interspeech*, Brighton, U.K., 2009, pp. 2255–2258.

[33] S. Young, P. Woodland, G. Evermann, and M. Gales, *The HTK toolkit 3.4.1*. Cambridge, U.K.: Cambridge Univ. Eng Dept., 2013 [Online]. Available: http://htk.eng.cam.ac.uk

[34] M. J. Gales and P. C. Woodland, "Mean and variance adaptation within the MLLR framework," *Comput. Speech Lang.*, vol. 10, no. 4, pp. 249–264, 1996.

[35] Y. Ohtani, T. Toda, H. Saruwatari, and K. Shikano, "Non-parallel training for many-to-many eigenvoice conversion," in *Proc. IEEE ICASSP*, Dallas, TX, USA, Mar. 2010, pp. 4822–4825.

[36] B. Denby, T. Schultz, K. Honda, T. Hueber, J. Gilbert, and J. Brumberg, "Silent speech interfaces," *Speech Commun.*, vol. 52, no. 4, pp. 270–287, 2010, 4.

[37] T. Hueber and G. Bailly, "Statistical conversion of silent articulation into audible speech using full-covariance HMM," *Comput. Speech Lang.*, 2015, to be published.

Thomas Hueber received an engineering degree in electronics, telecommunication and computer science from CPE Lyon (France) and an M.Sc. in image processing from University of Lyon in 2006. He worked towards his Ph.D. in computer science on *silent speech interfaces* and obtained it from Pierre and Marie Curie University (Paris) in 2009. In 2010, he joined GIPSA-lab (Grenoble, France) as a Post-Doctoral Researcher and became a tenured CNRS researcher in 2011. His research activities deal with multimodal speech processing (recognition, synthesis, conversion), with a special interest in speech biosignals (such as the articulatory movements, muscle and brain activities), their modeling using machine learning techniques, and their use in assistive technologies. In 2011, he received the 6th Christian Benoit Award (ACB/ISCA/AVISA) for his work on a real-time silent speech interface driven by ultrasound and video imaging.

Laurent Girin received the M.Sc. and Ph.D. degrees in signal processing from the Institut National Polytechnique de Grenoble (INPG), Grenoble, France, in 1994 and 1997, respectively. In 1999, he joined the Ecole Nationale Supérieure d'Electronique et de Radioélectricité de Grenoble (ENSERG), as an Associate Professor. He is now a Professor at Phelma (Physics, Electronics, and Materials Department of Grenoble-INP), where he lectures signal processing theory and applications to audio. His research activity is carried out at GIPSA-Lab (Grenoble Laboratory of Image, Speech, Signal, and Automation). It deals with different aspects of speech and audio processing (analysis, modeling, coding, transformation, synthesis), with a special interest in joint audio/visual speech processing and source separation. Prof. Girin is also a regular collaborator at INRIA (French Computer Science Research Institute), Grenoble, as an associate member of the Perception Team.

Xavier Alameda-Pineda received the M.Sc. in mathematics, in telecommunications from Barcelona Tech; and in computer science from Grenoble-INP. He worked towards his Ph.D. in mathematics and computer science in the Perception Team at INRIA, and obtained it from Université Joseph Fourier in 2013. He is currently a Post-Doctoral Fellow at the University of Trento. His research interests are multimodal machine learning and signal processing for scene analysis.

Gérard Bailly is a Senior CNRS Research Director at GIPSA-Lab, Grenoble-France. He was Deputy Director of the lab in 2007–2012. He has been working in the field of speech communication for 30 years, supervised 27 Ph.D. thesis, authored 40 journal papers, 24 book chapters and more than 170 papers in major international conferences. He coedited *Talking Machines: Theories, Models and Designs* (Elsevier, 1992), *Improvements in Speech Synthesis* (Wiley, 2002) and *Audiovisual Speech Processing* (CUP, 2012). His current interests include the conception and evaluation of interactive systems, in particular social robots and virtual conversational agents.