Acoustic VR in the Mouth: A Real-time Speech-driven Visual Tongue System

Ran Luo[1], Qiang Fang*[2], Jianguo Wei[3], Wenhuan Lu†[3], Weiwei Xu[4], and Yin Yang‡[1]

[1]Department of Electrical and Computer Engineering, The University of New Mexico, USA
[2]Institute of Linguistics, Chinese Academy of Social Sciences, China
[3]School of Software, Tianjin University, China
[4]State Key Laboratory of CAD & CG, Zhejiang University, China

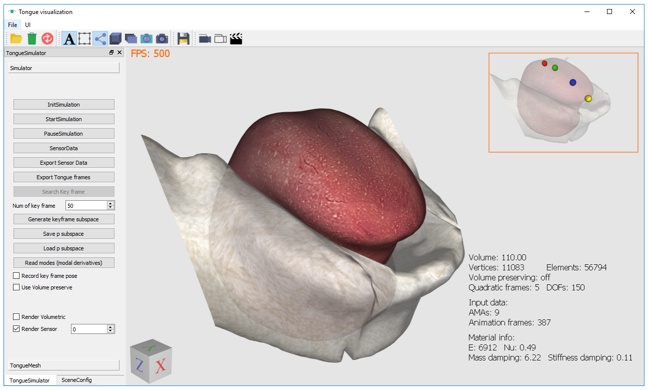
Abstract

We propose an acoustic-VR system that converts acoustic signals of human language (Chinese) to realistic 3D tongue animation sequences in real time. It is known that directly capturing the 3D geometry of the tongue at a frame rate that matches the tongue's swift movement during the language production is challenging. This difficulty is handled by utilizing the electromagnetic articulography (EMA) sensor as the intermediate medium linking the acoustic data to the simulated virtual reality. We leverage Deep Neural Networks to train a model that maps the input acoustic signals to the positional information of pre-defined EMA sensors based on 1,108 utterances. Afterwards, we develop a novel reduced physics-based dynamics model for simulating the tongue's motion. Unlike the existing methods, our deformable model is nonlinear, volume-preserving, and accommodates collision between the tongue and the oral cavity (mostly with the jaw). The tongue's deformation could be highly localized which imposes extra difficulties for existing spectral model reduction methods. Alternatively, we adopt a spatial reduction method that allows an expressive subspace representation of the tongue's deformation. We systematically evaluate the simulated tongue shapes with real-world shapes acquired by MRI/CT. Our experiment demonstrates that the proposed system is able to deliver a realistic visual tongue animation corresponding to a user's speech signal.

1 Introduction

The human tongue is a muscular organ that plays an essential role during speech production. A high-quality visual representation of the human tongue for specific speech sounds is of importance in the domain of speech research and has numerous potential applications. For example, in the rehabilitation of speech disorders [16], a realistic visualization of 3D tongue motion could provide a visible paradigm that helps an individual achieve the correct articulation of the tongue during the production of various speech sounds.

Unfortunately, the detailed mechanism that drives the deformable motion of the human tongue remains largely unknown to the research community – there exist many challenges, both practical and theoretical, that are still underexplored. Firstly, the tongue is an interior organ inside of the oral cavity. As such, ordinary optical sensors like video cameras are not suited to retrieve the motion data. Secondly, the tongue's movement during the production of speech is swift. For instance, a complete production of a single vowel-consonant-vowel (VCV) syllable takes only tenths of a second. Therefore, most 3D imaging modalities like computed tomography (CT) or magnetic resonance imaging (MRI) are not able to follow such quick movements. The ultrasound imaging (US), while widely used in many tongue related research, produces only 2D information and contains noise which requires post-processing. More importantly, the contour information at the tongue tip is frequently missed. Restoring or constructing the per-frame correspondence for a given US sequence is challenging and relies heavily on manual labelling, which is subjective and tedious. Lastly, the anatomical structure of the tongue is quite complex [14, 49], requiring several intrinsic and extrinsic muscles to be tightly coordinated during the speech sound production. An accurate mathematical description for the speech motor control is still beyond the knowledge for speech scientists [23, 35]. The inverse dynamics method turns out to be a promising solution to this problem [7, 17, 42, 48, 57]. This method does not require an active muscle-activation-driven motor to control the tongue's motion. Instead, it unitizes the pre-defined biomechanical parameters of the vocal tract to reconstruct the tongue motion by enforcing certain constraints during the simulation.



FPS: 500

Num of key frame 50

□ Record key frame pose
□ Use Volume preserve

□ Render Volumetric
☑ Render Sensor 0

Volume: 110.00
Vertices: 11083 Elements: 56794
Volume preserving: off
Quadratic frames: 5 DOFs: 150

Input data:
AMAs: 9
Animation frames: 387

Material info:
E: 6912 Nu: 0.49
Mass damping: 6.22 Stiffness damping: 0.11

Figure 1: A snapshot of the interface of the proposed system.

The proposed framework further advances the existing inverse dynamics models and advances the frontier of creating a realistic virtual reality representation of the invisible vocal tract. While our framework also employs the idea of inverse dynamics, many novel techniques have been developed which complement and are orthogonal to the state-of-the-arts. First of all, the tedious sensor setup is hidden in our system to the end user. To this end, we use deep learning to train a mapping mechanism that directly converts input acoustic signals to feature vectors (position information) of articulators. The training uses a database consisting of complete and meaningful sentences (in Chinese) instead of simple CV syllables (i.e. as in [57]). Secondly, our framework equips a dedicated

*Qiang Fang and Ran Luo are joint first authors as they contributed equally on the speech inversion and inverse dynamics.

†wenhuan@tju.edu.cn

‡yangy@unm.edu

nonlinear finite element method (FEM) simulator using a technique called *spatial reduction* and domain decomposition. Nonlinear deformations of the tongue can be simulated accurately in *real time*. This method allocates low-dimensional simulation degrees of freedom (DOFs) more effectively than standard modal reduction techniques [6, 9]. The nonlinear deformation pattern is well captured by quadratic DOFs associated with each domain and is smoothly blended across the entire tongue model. The nonlinear volume preserving constraint is fully addressed in our framework. At each simulation time frame, we compute a displacement and velocity correction that effectively suppresses the volume change. A data-driven method is used to create a pressure subspace to facilitate an efficient volume conservation at the simulation runtime. The simulated tongue shapes are compared to real-world MRI-CT data. The results show that our framework delivers a high-quality animated tongue dynamics which could be of great potential for a wide range of medical scenarios and clinic applications. In summary, some noteworthy technical features of our framework are:

- **Plug and play:** We leverage the deep learning method to train a speech inversion mechanism from acoustic signals to articulators' positions (§ 5). Hence, from an end user's point of view, tedious experiment setup is skipped, and the speech production can be performed in a comfortable and sensor-free environment, which drives the visual animated dynamic model in real time.
- **Nonlinearity in real time:** Our inverse dynamic simulator uses a series of novel numerical techniques that accurately capture local nonlinear deformations of the tongue while retaining the entire simulation algorithm within a low-dimensional configuration (§ 6 & § 7). Volume preservation is achieved by using displacement/velocity correction within the pressure subspace (§ 8).
- **High-quality:** Our framework is empowered by real-world and subject-specific data from various imaging modalities (§ 4). The model's quality at each step of the framework is systematically evaluated in an objective way (§ 9).

2 Related Work

The human tongue is a critical articulator for speech sound production. Investigating its behavior and contribution to speech production has been of interest to researchers in linguistics, phonetics and physiology. Due to the interdisciplinary nature of this work, we only briefly cover a few of the most relevant existing studies in acoustic signal processing, speech inversion and FEM tongue modeling in this section.

Acquisition of the tongue's geometry The human tongue is an interior organ and its motion is inaccessible to regular optical sensors like video cameras. The rapidly developing MRI systems have been used as an important data source [4, 5] for gathering 3D tongue shapes. To capture the tongue motion, three sagittal directions of MRI images [46] were used to record the 2D contours of tongue in three sagittal planes. However, the MRI acquisition frequency is too low for capturing the rapid tongue motion during real language production. Recent advances of high-speed MRI [26] have shown significant potentials of real-time shape acquisition [11, 20, 27]. However, they are still not yet able to capture intact 3D geometric information of the tongue. X-ray CT imaging systems have higher temporal resolutions [45]. However, they expose the speaker to radiation. Thus, they are not applicable for massive data collection. Ultrasound or US systems [38, 41] have also been widely used for modeling tongue movements at a very high frequency (100 *Hz* for instance). However, they often miss the tracking of the tongue tip [24, 28, 47] because of the surrounding air gaps, and the resulting images are always noisy. Hence, restoring the frame-to-frame correspondence over the tongue 2D contour for an US sequence is a challenging problem which requires significant overhead [1, 19, 25].

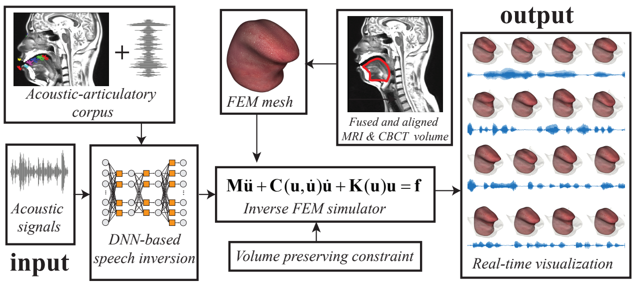

output
Acoustic-articulatory corpus
+
FEM mesh
Fused and aligned MRI & CBCT volume
Acoustic signals
input
DNN-based speech inversion
$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}(\mathbf{u},\dot{\mathbf{u}})\dot{\mathbf{u}} + \mathbf{K}(\mathbf{u})\mathbf{u} = \mathbf{f}$
Inverse FEM simulator
Volume preserving constraint
Real-time visualization

Figure 2: An overview of our acoustic-VR system.

Speech inversion Our framework is also related to speech inversion, a technique that estimates vocal tract shapes or articulators' positions based on input speech signals. Speech inversion has been performed by the codebook searching method by synthesizing sounds from the entire space of control parameters of an articulatory model [3, 33]. The problem with this approach is that the synthesis includes articulations that never occur in real human speech production. Clearly, the relationship between the acoustic and articulatory features is highly nonlinear and may not be bijective. Furthermore, the articulator's movements are not solely determined by the phoneme being pronounced, but also by the succeeding or preceding phonemes (the so called co-articulation phenomenon). In past decades with the increasing popularity of machine learning techniques, a number of methods have been proposed to tackle this problem using statistical learning such as the Hidden Markov model (HMM) [18, 58], the Gaussian mixture model (GMM) [51], the artificial neural network (ANN) [36], and the deep neural network (DNN) [52]. Wu and colleagues [56] tested the performance of the aforementioned techniques on the acoustic-articulatory English speech corpus MNGU0 (http://www.mngu0.org/). The results indicated that DNN-based acoustic-articulatory mapping tended to yield the best performance, and our framework also uses the DNN model to achieve a high-quality speech inversion.

FEM-based tongue dynamics Biomechanical models of the tongue using FEM methods are widely used [8, 13, 43]. An active biomechanical model takes the muscle activations as the input to simulate the speech motors. Interesting results have been reported using active models. For instance, Stavness and colleagues developed an algorithm that is able to automatically estimate the internal activation of a muscle group [44]. The limitations of this approach is that the state-of-the-art active model is only able to simulate general tongue movements like upward or lateral bending. Subtle and localized deformations on the tongue are still difficult to be directly generated. Conversely, the inverse dynamics technique or the passive model that builds the unknown motion based on pre-known constraints [7, 42, 57] have some notable advantages over the active methods. They allow us to restore the 3D motion of the tongue with partial information that is more accessible than full-scale activation control.

Real-time deformable model Simulating the human tongue is a problem well suited for FEM deformable models. Since the FEM simulation of nonlinear deformation is known to be time-consuming, a technique referred to as *model reduction* is widely used in the computer graphics and animation community [40], which is able to improve performance by orders-of-magnitude. The idea is to build a displacement subspace consisting of representative deformed shapes and restrict the nonlinear integration within the constructed subspace. Standard model reduction uses the modal analysis that decomposes the dynamics into a set of linear vibrations [34] which is only valid for small-scale linear elasticity. Yang and colleagues [57] used an extended modal analysis method called modal warping [9] to simulate the tongue's dynamics. Because this method is still based on linear elasticity, it is not able to produce plausible tongue dynamics for full words and sentences. Nonlinear elasticity can also be dealt with using modal analysis [6] however, subspace bases are global eigenvectors and less expressive for localized deformation.

3 System Overview

As sketched in Fig. 2, our framework takes real human speech signals (in Chinese) as input, and outputs realistic tongue animation sequences in real time corresponding to the speech being produced. The input speech signals are first mapped to articulator's positions through a DNN-based speech inversion. The DNN is trained using an acoustic-articulatory corpus consisting of 1,108 complete sentences (§ 5). The output of the speech inversion is the estimated position information corresponding to four electromagnetic articulography (EMA) sensors at the tongue's tip, blade, dorsum and rear on the mid-sagittal plane. This information serves a set of constraint equations for the FEM tongue simulator. Our simulator is geometrically nonlinear. The finite element mesh of the tongue is obtained by fusing the MRI and CBCT volumes of the same subject, whose contour is manually outlined by domain experts and triangularized afterwards (§ 4). We developed a novel reduced deformable simulator using blended quadratic domains. While this simulator is low-dimensional and model reduced, the nonlinear DOFs are assigned according to the location of EMA sensors so that local deformation can be well captured (§ 6). The global and general tongue's motion is calculated by smoothly blending the local domain-wise deformations in a material-aware manner (§ 7). Our simulator also addresses the nonlinear volume preserving constraint efficiently in a pressure subspace (§ 8). Collision detection and resolving is also handled in our system using the penalty method. As most FEM-related computation is done within a low-dimensional subspace, our system is real-time and able to produce plausible animations directly from human speech. The following sections describe each major technical component of our system in detail.

4 Data Acquisition

Our system is built upon real-world, subject-specific data including the tongue's geometry, feature positions from EMA coils, and the associated acoustic signals. This section elaborates on the data acquisition procedure.

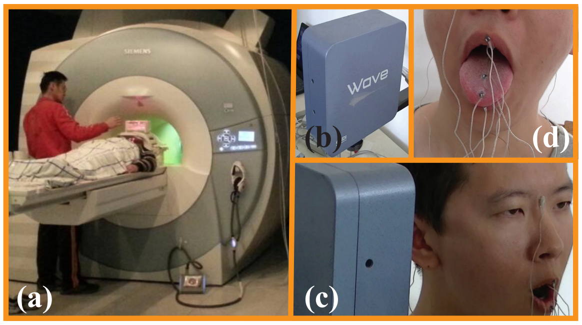


Figure 3: Experiment setting for data acquisition. (a) the SIEMENS MRI system. (b) NDI Wave EMA system. (c) Gathering the articulary movement data using EMA. (d) Placing EMA coils on the tongue.

Construction of the 3D tongue model Our 3D tongue model is built using both MRI (for soft tissues like the tongue, soft palate, and the pharyngeal wall) and cone beam CT (CBCT) (for bony structures) of an individual subject. The MRI data is recorded using the SIEMENS MAGNETOM Trio, a Tim system with 3 tesla magnetic field strength, 64 *ms* echo time, 340 *ms* repetition time, 31 sagittal slice planes, 3 *mm* slice thickness, 3.6 *mm* slice interval, 256 by 256 *mm* field of view, and 192 by 192 pixel resulting image size (Fig. 3 (a)). The rightmost and leftmost planes are located at 54 *mm* from the mid-sagittal plane. The "rest shape"of the tongue is defined as the averaged tongue shape for 36 Chinese vowels (9 vowels with 4 different tones) and 73 consonants in symmetric VCV syllables including fricative, stop, affricate, nasal, as well as lateral[1]. Detailed phonetic information is reported in Fig. 4.

Vowel	Fricative	Stop	Affricate	Nasal	Lateral

[a], [i],
[ʅ], [ʮ],
[u], [ɛ],
[ɤ], [o],
[y]
	[s] + [a], [i], [u];
	[ʂ] + [a], [i], [u];
	[ɕ] + [i], [y];
	[f] + [a], [ɛ], [u], [o];
	[x] + [a], [ɛ], [u];
		[t] + [a], [i], [u], [ɛ];
		[k] + [a], [i], [u], [ɛ];
		[p] + [a], [i], [u], [o];
		[p^h] + [a], [i], [u], [o];
		[t^h] + [a], [i], [u], [ɛ];
		[k^h] + [a], [i], [u], [ɛ];
			[ʦ] + [a], [i], [u], [ɛ];
			[tʂ] + [a], [i], [u], [ɛ];
			[ʨ] + [i], [y];
			[ʦ^h] + [a], [i], [u], [ɛ];
			[tʂ^h] + [a], [i], [u], [ɛ];
			[ʨ^h] + [i], [y];
				[m] + [a], [i],
				[u], [o], [ɛ];
				[n] + [a], [i],
				[u];
					[l] + [a],
					[i], [u],
					[y], [ɤ];
					Approximant
					[r] + [i];

Figure 4: VCV syllables used for MRI-based tongue shape retrieval.

During MRI data acquisition, the subject took the supine position and was asked to perform the required VCV syllables after a short period of warm-up practice. Each VCV sequence was produced with a consonant, surrounded by vowels, e.g. [a]–[t]–[a]. All articulations were artificially sustained during the ten-second acquisition time. For consonants, the subject made the initial VC transition before the acquisition, then held the articulation while breathing out very slowly (for fricatives) or holding his breath (for stops) and finally made the rest CV transition after the MRI scan. Other bony structures attached to the tongue such as teeth and jaw are acquired by a LargeV HiRes 3D dental CBCT. This device is primarily used for dental surgery and delivers much less radiation to the subject than regular CT systems. Afterwards, a rigid body registration is applied to align the MRI and CBCT volumes (Fig. 5 (b)). Finally, the volumetric tetrahedral mesh is built using Tetgen [39] based on the extracted surface information.

Acoustic-articulatory corpus The NDI Wave system (Fig. 3 (b)) is employed to record acoustic and articulator position recordings simultaneously. The articulators use electromagnetic transducer coils glued to the vocal-tract articulators to record precise measurements of their positions. There are 1,108 phonetically balanced Chinese sentences in total selected to serve as the recording prompts. In the EMA experiment, coils or sensors are attached to the Tongue Rear (TR), Tongue Dorsum (TD), Tongue Blade (TB), Tongue Tip (TT), Lower Incisor (LI), Lower Lip (LL) and Upper Lip (UL) in the mid-sagittal plane. Another two coils attached to the ridge of nose serve as a reference (as shown in Fig. 5 (a)). As a result, we can easily extract the global rigid body motion associated with head's movement as in [57]. The same subject participates in the EMA experiment. The acoustic signals and articulatory data are recorded simultaneously. The sampling frequencies are 16,000 *Hz* for acoustic signals and 100 *Hz* for the articulatory signal, respectively. A third-order *Savitzky-Golay* filter [29] with the frame size of 21 is applied to smooth the trajectory of coils attached to articulators to suppress their jittery motions.

5 DNN-based Speech Inversion

Speech inversion is the first step in our system. It refers to the procedure that estimates the vocal tract shape or articulators' positions based on speech signals. We develop a DNN-based mapping mechanism bridging the acoustic speech and the articulatory movement, taking the features representing the acoustic speech as the initial input and outputting the articulatory features at EMA sensors.

Traditional DNN is constructed by stacking a series of trained *Restricted Boltzmann Machines* (RBMs) [32], where a hidden layer of the preceding RBM serves as the visible layer of the following RBM. At the top, a regression layer with linear units is added to

[1] We note that there does not exist a well defined rest configuration of the tongue, and the most comfortable position of the tongue varies significantly by individuals. Therefore, the median shape of the tongue while performing a series of standard pronunciations is a more meaningful starting point.

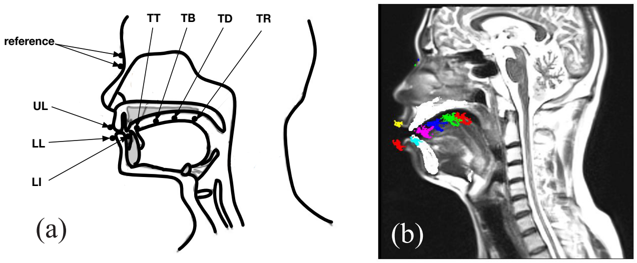
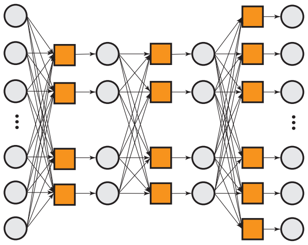


Figure 5: (a) EMA sensors' placement and (b) the aligned MRI-CBCT-EMA volume.

the RBM stacks. This method has been proven to be simple and effective in many applications. The activation of each unit can be formulated as: $I_{(n),i} = \sum_j w_{(n),ij} o_{(n-1),j} + b_{(n),j}$, where $I_{(n),i}$ is the input of the i^{th} unit in the n^{th} layer. $o_{(n-1),j}$ is the output of the j^{th} unit from the $(n-1)^{\text{th}}$ layer. $b_{(n),i}$ is the bias of the j^{th} unit in the n^{th} layer. The distribution of $I_{(n),i}$ varies during the training as the parameters of the previous layers change. This issue downgrades the learning rates, requires a more dedicated parameter initialization, and makes it hard to train models with saturating nonlinearities. To deal with this problem, we employ a batch normalization strategy as proposed in [21]. This method performs the normalization over a part of the model's architecture (orange blocks in the inset) as:



$$\widetilde{x}_{(n),i} = \frac{x_{(n),i} - \mu_{(n),i}}{\sqrt{\sigma^2_{(n),i} + \varepsilon}}, \quad x_{(n),i} = \sum_j w_{(n),ij} o_{(n-1),j} + b_{(n),j}, \tag{1}$$

and the final output becomes:

$$x_{(n),i} = \gamma_{(n),i} \widetilde{x}_{(n),i} + \beta_{(n),i}. \tag{2}$$

Here, $\mu_{(n),i}$ and $\sigma^2_{(n),i}$ are the mean and variance of $x_{(n),i}$. $\gamma_{(n),i}$ and $\beta_{(n),i}$ are scaling and shifting parameters applied to the normalized $\widetilde{x}_{(n),i}$. All the parameters are evolved using the iterative momentum gradient method as:

$$\begin{aligned}
\mathbf{W}^{i+1} &= \mathbf{W}^i + \Delta\mathbf{W}^{i+1} & \Delta\mathbf{W}^{i+1} &= d \cdot \Delta\mathbf{W}^i - \eta \cdot \frac{\partial L}{\partial \mathbf{W}} \\
\mathbf{b}^{i+1} &= \mathbf{b}^i + \Delta\mathbf{b}^{i+1} & \Delta\mathbf{b}^{i+1} &= d \cdot \Delta\mathbf{b}^i - \eta \cdot \frac{\partial L}{\partial \mathbf{b}} \\
\boldsymbol{\gamma}^{i+1} &= \boldsymbol{\gamma}^i + \Delta\boldsymbol{\gamma}^{i+1} & \Delta\boldsymbol{\gamma}^{i+1} &= d \cdot \Delta\boldsymbol{\gamma}^i - \eta \cdot \frac{\partial L}{\partial \boldsymbol{\gamma}} \\
\boldsymbol{\beta}^{i+1} &= \boldsymbol{\beta}^i + \Delta\boldsymbol{\beta}^{i+1} & \Delta\boldsymbol{\beta}^{i+1} &= d \cdot \Delta\boldsymbol{\beta}^i - \eta \cdot \frac{\partial L}{\partial \boldsymbol{\beta}} \\
\boldsymbol{\mu}^{i+1} &= \boldsymbol{\mu}^i + \Delta\boldsymbol{\mu}^{i+1} & \Delta\boldsymbol{\mu}^{i+1} &= d \cdot \Delta\boldsymbol{\mu}^i - \eta \cdot \frac{\partial L}{\partial \boldsymbol{\beta}} \\
\boldsymbol{\sigma}^{2^{i+1}} &= \boldsymbol{\sigma}^{2^i} + \Delta\boldsymbol{\sigma}^{2^{i+1}} & \Delta\boldsymbol{\sigma}^{2^{i+1}} &= d \cdot \Delta\boldsymbol{\beta}^i - \eta \cdot \frac{\partial L}{\partial \boldsymbol{\beta}},
\end{aligned} \tag{3}$$

where $\mathbf{W}$, $\mathbf{b}$, $\boldsymbol{\gamma}$, $\boldsymbol{\beta}$, $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}^2$ are the aggregated training parameters w, b, γ, β, μ and σ^2 for a certain layer in the matrix/vector form. L is the loss over the training set. The superscript $[\cdot]^i$ indicates the iteration index. The partial derivatives of $\partial L/\partial \mathbf{W}$, $\partial L/\partial \mathbf{b}$, $\partial L/\partial \boldsymbol{\gamma}$, $\partial L/\boldsymbol{\beta}$, $\partial L/\boldsymbol{\mu}$ and $\partial L/\boldsymbol{\sigma}^2$ can be calculated using the backpropagation algorithm as:

$$\begin{aligned}
&\frac{\partial L}{\partial \mathbf{W}_{(n)}} = \frac{1}{m}\sum_{l=1}^{m} \mathbf{o}^{(l)}_{(n-1)} \left(\frac{\partial L^{(l)}}{\partial \mathbf{x}^{(l)}_{(n)}}\right)^{\top}, \qquad \frac{\partial L}{\partial \mathbf{b}_{(n)}} = \frac{1}{m}\sum_{l=1}^{m} \left(\frac{\partial L^{(l)}}{\partial \mathbf{x}^{(l)}_{(n)}}\right)^{\top}, \\
&\frac{\partial L}{\partial \gamma_{(n),i}} = \sum_{l=1}^{m} \frac{\partial L^{(l)}}{\partial I^{(l)}_{(n),i}} \widetilde{x}^{(l)}_{(n),i}, \qquad \frac{\partial L}{\partial \beta_{(n),i}} = \sum_{l=1}^{m} \frac{\partial L^{(l)}}{\partial I^{(l)}_{(n),i}}, \\
&\frac{\partial L}{\partial \mu_{(n),i}} = -\frac{1}{m}\sum_{l=1}^{m} \frac{\partial L^{(l)}}{\partial \widetilde{x}^{(l)}_{(n),i}} \frac{1}{\sqrt{\sigma^2_{(n),i} + \varepsilon}} - \frac{2}{m}\sum_{l=1}^{m} \frac{\partial L^{(l)}}{\partial \sigma^2_{(n),i}} \left(x^{(l)}_{(n),i} - \mu_{(n),i}\right), \\
&\frac{\partial L}{\partial \sigma^2_{(n),i}} = \frac{1}{m}\sum_{l=1}^{m} \frac{\partial L^{(l)}}{\partial \sigma^2_{(n),i}},
\end{aligned} \tag{4}$$

where $\mathbf{o}^{(l)}_{(n)}$ is the output of the n^{th} layer. The summation index l iterates all the m samples in a mini-batch. Other intermediate partial derivatives can be computed as:

$$\begin{aligned}
&\frac{\partial L^{(l)}}{\partial \mathbf{x}^{(l)}_{(n)}} = \frac{\partial \mathbf{I}^{(l)}_{(n)}}{\partial x^{(l)}_{(n)}} \frac{\partial L^{(l)}}{\partial \mathbf{I}^{(l)}_{(n)}}, \qquad \frac{\partial I^{(l)}_{(n),i}}{\partial x^{(l)}_{(n),i}} = \frac{\gamma_{(n),i}}{\sqrt{\sigma^2_{(n),i} + \varepsilon}}, \\
&\frac{\partial L^{(l)}}{\partial \mathbf{I}^{(l)}_{(n-1)}} = \frac{\partial \mathbf{o}^{(l)}_{(n-1)}}{\partial \mathbf{I}_{(n-1)}} \mathbf{W}_{(n)} \frac{\partial \mathbf{I}^{(l)}_{(n)}}{\partial \mathbf{x}^{(l)}_{(n)}} \frac{\partial L^{(l)}}{\partial \mathbf{I}^{(l)}_{(n)}}, \qquad \frac{\partial L^{(l)}}{\partial \widetilde{x}^{(l)}_{(n),i}} = \frac{\partial L^{(l)}}{\partial I^{(l)}_{(n),i}} \gamma_{(n),i}, \\
&\frac{\partial L^{(l)}}{\partial \sigma^2_{(n),i}} = -\frac{1}{2} \frac{\partial L^{(l)}}{\partial \widetilde{x}^{(l)}_{(n),o}} \left(x^{(l)}_{(n),i} - \mu_{(n),i}\right) \left(\sigma^2_{(n),i} + \varepsilon\right)^{-\frac{3}{2}}.
\end{aligned}$$

The recorded speech signals are segmented into frames with a hanning window. Each frame contains a speech segment of 25 *ms*, and is encoded by the log-energy and 12-order Mel-frequency cepstral coefficients (MFCC) augmented with their delta and delta-deltas. The frame shift between consecutive frames is 10 *ms* to match the sampling rate of EMA sensors. The dataset is partitioned in three sets: a validation and a testing set of 110 utterances each, and a training set consisting of the other 880 utterances. Both EMA and MFCC feature vectors are normalized by subtracting their global mean and dividing by the standard deviation of each dimension, respectively.

6 Real-time Deformable Simulation of Tongue

The tetrahedral finite element mesh used for the 3D tongue model consists of 11,083 nodal points and 56,794 tetrahedral elements. Simulating such high-dimensional mesh with over 30K DOFs at the rate in sync with the acoustic input is challenging. Yang and colleagues [57] adopted a simplified dynamic model extending the linear elasticity using the modal warping technique [9] to alleviate this problem. Unfortunately, we found that the simulator was only able to generate plausible results for repeated CV trainings (e.g. [ta]–[ta]–[ta]). It often produced unnatural motion patterns for real speech production of complete words and sentences. The reasons are twofold. First, the modal warping method is still based on linear elasticity and its nonlinear deformation comes from a geometric warping correction, which is not physics-based. Second, modal analysis constructs global subspace basis vectors while during language production, the tongue's deformation could be highly nonlinear and localized. As reported in the previous study [12], the human tongue undergoes a compression up to ∼ 200% and an elongation up to ∼ 160% when producing certain speech sounds. In other words, we need a new numerical framework that is able to perform the deformation integration in real time and effectively capture the nonlinear local deformations. In this section, we detail a novel *spatial reduction* method that allocates nonlinear simulation DOFs via *quadratic domains*. Each domain houses 30 DOFs grouped into 3 translation DOFs, 9 affine DOFs, 9 quadratic homogenous DOFs, as well as 9 quadratic heterogenous DOFs. We assign each EMA sensor a domain and an additional one for the tongue's interior in order to capture local deformation nearby the sensor while keeping the overall simulation in a low-dimensional subspace. Our model also fully addresses the volume preserving constraint rather than relying on tweaking Poisson's ratio as in [57].

Kinematics For a given material point $\mathcal{P}$ on the tongue model, we denote $\mathbf{x} = [x_1, x_2, x_3]^\top$ and $\mathbf{u} = [u_1, u_2, u_3]^\top$ as its rest shape position and displacement. A nearby *domain* imposes a quadratic influence to its displacement components such that $u_i = \mathbf{x}^\top \mathbf{Q}_i \mathbf{x} + \mathbf{a}_i^\top \mathbf{x} + t_i$ for $i = 1,2,3$. $\mathbf{Q}_i \in \mathbb{R}^{3\times 3}$ is a symmetric tensor encoding the iso-quadratic DOFs. We put its three diagonal DOFs into a vector such that $\mathbf{q}_{o_i} = [Q_{11}, Q_{22}, Q_{33}]^\top$ and refer to it as *homogenous* DOFs. Similarly, the vector $\mathbf{q}_{e_i} = [2Q_{12}, 2Q_{23}, 2Q_{13}]^\top$ containing off-diagonal elements of $\mathbf{Q}_i$ is referred to as *heterogenous* DOFs. The *affine* DOFs $\mathbf{a} \in \mathbb{R}^3$ describes how u_i is linearly related to its rest position, and t_i is a *translation* DOF. Each type of deformable DOFs from different domains are convexly combined, and the displacement of $\mathcal{P}$ can be written as:

$$u_i = \sum_j w_t^j(\mathbf{x}) t_i^j + w_a^j(\mathbf{x}) \mathbf{a}_i^{j\top} \mathbf{x} + w_o^j(\mathbf{x}) \mathbf{q}_{o_i}^{j\top} \widetilde{\mathbf{x}} + w_e^j(\mathbf{x}) \mathbf{q}_{e_i}^{j\top} \widehat{\mathbf{x}}, \tag{5}$$

where w_t^j, w_a^j, w_o^j and w_e^j are location-dependent weight coefficients indicating how much domain j affects different types of deformable DOFs. $\widetilde{\mathbf{x}} = [x_1^2, x_2^2, x_3^2]^\top$ and $\widehat{\mathbf{x}} = [x_1 x_2, x_2 x_3, x_1 x_3]^\top$ are second-order homogenous and heterogenous vectors of $\mathcal{P}$. By stacking all the deformable DOFs from the j^{th} domain into a single vector $\mathbf{q}^j \in \mathbb{R}^{30}$ such that $\mathbf{q}^j = [\mathbf{t}^{j\top}, \mathbf{a}_1^{j\top}, \mathbf{a}_2^{j\top}, \mathbf{a}_3^{j\top}, \mathbf{q}_{o_1}^{j\top}, \mathbf{q}_{o_2}^{j\top}, \mathbf{q}_{o_3}^{j\top}, \mathbf{q}_{e_1}^{j\top}, \mathbf{q}_{e_2}^{j\top}, \mathbf{q}_{e_3}^{j\top}]^\top$, the displacement of $\mathcal{P}$ can be concisely expressed as a matrix-vector product:

$$\mathbf{u} = \mathbf{G}^j \mathbf{q}^j = \left[\mathbf{G}_t^j | \mathbf{G}_a^j | \mathbf{G}_o^j | \mathbf{G}_e^j\right] \mathbf{q}^j, \tag{6}$$

where

$$\begin{aligned} \mathbf{G}_t^j &= w_t^j \mathbf{I} & \mathbf{G}_a^j &= w_a^j \mathbf{I} \otimes \mathbf{x}^\top \\ \mathbf{G}_o^j &= w_o^j \mathbf{I} \otimes \widetilde{\mathbf{x}}^\top & \mathbf{G}_e^j &= w_e^j \mathbf{I} \otimes \widehat{\mathbf{x}}^\top. \end{aligned}$$

We call matrix $\mathbf{G}^j$ the *geometric displacement matrix* as it depends soley on the rest shape of the tongue mesh. The generalized coordinate $\mathbf{q}^j$ uniquely determines the kinematic of $\mathcal{P}$:

$$\dot{\mathbf{u}} = \sum_j \mathbf{G}^j \dot{\mathbf{q}}^j, \qquad \ddot{\mathbf{u}} = \sum_j \mathbf{G}^j \ddot{\mathbf{q}}^j. \tag{7}$$

Reduced dynamics Let $\mathbf{e}_i$ denote the canonical basis vectors of $\mathbb{R}^3$, and we drop the domain superscript $[\cdot]^j$ in this paragraph for the sake of a succinct formulation. Based on Eq. (5), each row of the deformation gradient tensor $\mathbf{F} = [\mathbf{F}_1, \mathbf{F}_2, \mathbf{F}_3]^\top \in \mathbb{R}^{3\times 3}$ can be written as $\mathbf{F}_i = \mathbf{F}_{t_i} + \mathbf{F}_{a_i} + \mathbf{F}_{o_i} + \mathbf{F}_{e_i} + \mathbf{e}_i$, where

$$\begin{aligned} \mathbf{F}_{t_i} &= \sum \nabla w_t t_i & \mathbf{F}_{a_i} &= \sum \mathbf{a}_i^\top \mathbf{x} \nabla w_a + w_a \mathbf{a}_i^\top \\ \mathbf{F}_{o_i} &= \sum \mathbf{q}_{o_i}^\top \widetilde{\mathbf{x}} \nabla w_o + w_o \mathbf{q}_{o_i}^\top \widetilde{\mathbf{X}} & \mathbf{F}_{e_i} &= \sum \mathbf{q}_{e_i}^\top \widehat{\mathbf{x}} \nabla w_e + w_e \mathbf{q}_{e_i}^\top \widehat{\mathbf{X}}, \end{aligned}$$

and

$$\widetilde{\mathbf{X}} = \begin{bmatrix} x_2 & x_1 & 0 \\ 0 & x_3 & x_2 \\ x_3 & 0 & x_1 \end{bmatrix}, \qquad \widehat{\mathbf{X}} = \begin{bmatrix} 2x_1 & 0 & 0 \\ 0 & 2x_2 & 0 \\ 0 & 0 & 2x_3 \end{bmatrix}.$$

Once we have computed $\mathbf{F}$, we can evaluate the nonlinear Green strain, $\mathbf{E} = \frac{1}{2}(\mathbf{F}^\top \mathbf{F} - \mathbf{I})$, and proceed to express the strain energy density Ψ as well as the first Piola-Kirchhoff stress tensor (PK1) based on the chosen material model. Previous research [12, 13, 53], indicates that an isotropic and homogenous material model for the tongue is applicable as the variation of Young's modulus at different parts of the tongue is very small. Accordingly we choose to use the St. Venant-Kirchhoff (StVK) model since it is capable of producing most desired nonlinear deformation effects of the tongue. The Young's modulus is set as $6{,}912$ and the Poisson's ratio is set as 0.49. Extending our method to accomodate other materials like Neo-Hookean is straightforward under our framework.

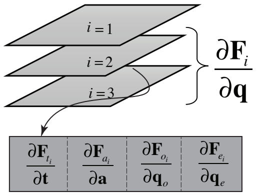
With the StVK material, the energy density and PK1 are formulated as: $\Psi = \mu \mathbf{E} : \mathbf{E} + \frac{\lambda}{2} \text{tr}^2(\mathbf{E})$ and $\mathbf{P} = \mathbf{F}[2\mu \mathbf{E} + \lambda \text{tr}(\mathbf{E}) \mathbf{I}]$, respectively, where λ and μ are the Lamé parameters. The per-domain reduced internal force $\widetilde{\mathbf{f}}_{int}$ and its gradient $\partial \widetilde{\mathbf{f}}_{int} / \partial \mathbf{q}$ are computed as:

$$\widetilde{\mathbf{f}}_{int} = -\int \left(\mathbf{P} \frac{\partial \mathbf{F}}{\partial \mathbf{q}} \right) \mathrm{d}V, \tag{8}$$

and

$$\frac{\partial \widetilde{\mathbf{f}}_{int}}{\partial \mathbf{q}} = -\int \left[\left(\frac{\partial \mathbf{P}}{\partial \mathbf{F}} \frac{\partial \mathbf{F}}{\partial \mathbf{q}} \right)^\top \frac{\partial \mathbf{F}}{\partial \mathbf{q}} \right] \mathrm{d}V. \tag{9}$$

Here, $\partial \mathbf{F} / \partial \mathbf{q} \in \mathbb{R}^{3\times 3\times 30}$ is a third-order block-sparse tensor, which can be understood as the superposition of three layers as shown at right. The i^{th} layer represents the matrix $\partial \mathbf{F}_i / \partial \mathbf{q}$ and it hosts four sub-matrices, namely $\partial \mathbf{F}_{t_i} / \partial \mathbf{t}$, $\partial \mathbf{F}_{a_i} / \partial \mathbf{a}$, $\partial \mathbf{F}_{o_i} / \partial \mathbf{q}_o$, and $\partial \mathbf{F}_{e_i} / \partial \mathbf{a}_e$. All of these sub-matrices are block-sparse as the partial derivative yields a nonzero block only when the subscript of the generalized coordinates agree. Each nonzero block can be easily calculated as:


$i=1$
$i=2$
$i=3$
$\frac{\partial \mathbf{F}_i}{\partial \mathbf{q}}$
$\frac{\partial \mathbf{F}_{t_i}}{\partial \mathbf{t}}$
$\frac{\partial \mathbf{F}_{a_i}}{\partial \mathbf{a}}$
$\frac{\partial \mathbf{F}_{o_i}}{\partial \mathbf{q}_o}$
$\frac{\partial \mathbf{F}_{e_i}}{\partial \mathbf{q}_e}$

$$\begin{aligned} \frac{\partial \mathbf{F}_{t_i}}{\partial t_i} &= \nabla w_t & \frac{\partial \mathbf{F}_{a_i}}{\partial \mathbf{a}_i} &= \nabla w_a \otimes \mathbf{x} + w_a \mathbf{I} \\ \frac{\partial \mathbf{F}_{o_i}}{\partial \mathbf{q}_{o_i}} &= \nabla w_o \otimes \widetilde{\mathbf{x}} + w_o \widetilde{\mathbf{X}}^\top & \frac{\partial \mathbf{F}_{e_i}}{\partial \mathbf{q}_{e_i}} &= \nabla w_e \otimes \widehat{\mathbf{x}} + w_e \widehat{\mathbf{X}}^\top. \end{aligned} \tag{10}$$

Applying temporal discretization using the implicit Euler integration leads to the final nonlinear system to be solved at each time step:

$$\left(\widetilde{\mathbf{M}} - h\widetilde{\mathbf{C}} - h^2 \frac{\partial \widetilde{\mathbf{f}}_{int}}{\partial \mathbf{q}} \right) \Delta \dot{\mathbf{q}} = h \widetilde{\mathbf{f}}_{ext} + h^2 \frac{\partial \widetilde{\mathbf{f}}_{int}}{\partial \mathbf{q}} \dot{\mathbf{q}}, \tag{11}$$

where $\widetilde{\mathbf{M}}$ is the reduced mass matrix, which can be evaluated blockwisely: $\widetilde{\mathbf{M}}^{ij} = \int \rho \mathbf{G}^{i\top} \mathbf{G}^j \mathrm{d}V$ ($\rho = 1$ as the tongue consists of mostly water); $\widetilde{\mathbf{f}}_{ext}$ is the generalized external force; h is the time step size; and $\widetilde{\mathbf{C}}$ is the reduced damping matrix.

7 Domains' Weight Coefficients

Analogous to shape functions used in the standard FEM that blend nodal quantities volumetrically within an element, weighting functions superimpose quadratic transformations from domains and yield global deformations of the tongue. The domain subspace should be material-customized. To this end, we present an efficient algorithm to calculate the weight distribution for each domain to accurately reflect the material properties of the tongue and augment the geometric displacement matrix (Eq. (6)). Our method is *fully* material-and-geometric-aware, and possesses important traits such as locality, smoothness and interpolation.

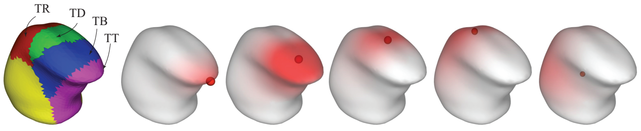


Figure 6: The domain partition of the input tongue mesh as well as the weight distribution for each domain.

Intuitively, the weighting function $w(\mathbf{x})$ ought to align with the visual impression of how the deformation fades away from the seed of the domain, where the maximum local displacement occurs. Apparently, a straightforward way to obtain such deformation dissipation is to solve a static equilibrium by imposing an external force $\mathbf{f}_s \in \mathbb{R}^3$ at the domain seed and anchoring all other seeds. Unfortunately, this problem is ill-defined as we have infinite choices of $\mathbf{f}_s$ – obviously they give different weighting distributions when used.

We resolve this ambiguity by restricting $\mathbf{f}_s$ along the *principle direction* $\mathbf{p}$ of a domain, which can be understood as the "most deformable direction" such that the domain undergoes the largest displacements when $\mathbf{f}_s$ aligns with it (i.e. $\mathbf{f}_s = \mathbf{p}$). Mathematically, it can be formulated as a quadratically constrained quadratic program (QCQP) problem as:

$$\begin{aligned} &\underset{\mathbf{p}}{\text{maximize}} && |\mathbf{u}|^2 \\ &\text{subject to} && \begin{bmatrix} \mathbf{K} & \mathbf{B}_a^\top \\ \mathbf{B}_a & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{B}_s^\top \mathbf{p} \\ \mathbf{0} \end{bmatrix}, \\ &\text{and} && |\mathbf{p}|^2 = 1, \end{aligned} \tag{12}$$

where $\mathbf{B}_s$ and $\mathbf{B}_a$ are two binary matrices picking the domain's seed on which $\mathbf{f}_s$ is applied, and anchor seeds to incorporate boundary conditions. λ is the unknown multipliers. In general, QCQP is NP-hard and a polynomial-time solution may not be available. Fortunately as Eq. (12) only activates equality constraints, it can be directly solved. To do so, we explicitly write down its inversion:

$$\begin{bmatrix} \mathbf{u}_p \\ \mathbf{u}_w \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{H}_{pp} & \mathbf{H}_{pw} & \mathbf{H}_{p\lambda} \\ \mathbf{H}_{pw}^\top & \mathbf{H}_{ww} & \mathbf{H}_{w\lambda} \\ \mathbf{H}_{p\lambda}^\top & \mathbf{H}_{w\lambda}^\top & \mathbf{H}_{\lambda\lambda} \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ \mathbf{0}_w \\ \mathbf{0}_\lambda \end{bmatrix}. \tag{13}$$

Here, the matrix $\mathbf{H}$ is the inverse of the constrained stiffness matrix in Eq. (12), which is often referred to as the *flexibility matrix*. We arrange all DOFs into groups such that subscript p is for 3 DOFs associated with the seed where the unit force $\mathbf{f}_s = \mathbf{p}$ is applied; subscript w is associated with all other nodal DOFs for which the weighting is to be calculated; and subscript λ is for multipliers' DOFs. Spanning the first rows in Eq. (13), the target function can be simplified:

$$\begin{aligned} |\mathbf{u}|^2 &= \mathbf{p}^\top (\mathbf{H}_{pp}\mathbf{H}_{pp} + \mathbf{H}_{pw}\mathbf{H}_{pw}^\top)\mathbf{p} \\ &\triangleq \mathbf{p}^\top \mathbf{A}\mathbf{p}. \end{aligned} \tag{14}$$

Note that $\mathbf{A}$ is SPD, and we diagonalize it using the eigen decomposition: $\mathbf{D} = \text{diag}(d_1, d_2, d_3) = \mathbf{\Phi}^\top \mathbf{A} \mathbf{\Phi}$, $d_1 \leq d2 \leq d3$ leading to:

$$\begin{aligned} |\mathbf{u}|^2 &= (\mathbf{\Phi}\mathbf{p})^\top \text{diag}(d_1, d_2, d_3)\,(\mathbf{\Phi}\mathbf{p})\,. \\ &= d_1 p_1^2 + d_2 p_2^2 + d_3 p_3^2 \\ &\leq d_3 \qquad\qquad (\text{by the fact that } |\mathbf{p}|^2 = p_1^2 + p_2^2 + p_3^2 = 1). \end{aligned}$$

The maximum value of $|\mathbf{u}|^2$ will be obtained when $\mathbf{p} = \boldsymbol{\phi}_3$, the eigenvector corresponding to the largest eigenvalue of $\mathbf{A}$.

After $\mathbf{p}$ is ready, the weighting function over the domain can be numerically computed by prescribing $\mathbf{p}$ as the constrained displacement. We notice that completely anchoring all the neighbor seeds as well as the domain boundary (i.e. fixing all of their x, y and z freedoms) produces an over-damped weighting. Accordingly, we lift up the boundary condition and only restrict their displacements along the principle direction, while tangential movements towards $\mathbf{p}$ are still allowed. In other words, each 3 by 3 sub-block of an identity matrix in $\mathbf{B}_a$ corresponding to a anchor node on the mesh is changed to $\mathbf{p}^\top$. Fig. 7 left shows the comparative results of a standard bending simulation of a load-end cantilever beam using different weighting functions. It can be seen that our method yields a natural and smooth nonlinear bending.

We subdivide the tongue mesh into five domains as shown in Fig. 6. Four of them are seeded at the corresponding EMA sensors. The fifth one is at the mass center of the tongue mesh. The domain's partition is obtained by performing flooding from the seed. The weight distribution of the domains is visualized using the red-white color map.

8 Preserving Volume within the Subspace

As a muscular organ, the human tongue consists of 99% of water, which preserves its volume during speech production. To achieve this effect, we introduce an auxiliary pressure variable which provides a volume adjustment of both displacement and velocity vectors to the simulated tongue mesh at each frame. This method has been explored by the computer graphics community [22, 50]. However, a fullspace displacment/velocity amendment is needed to solve the pressure terms for each element, which is $\mathbf{O}(n^2)$ at runtime and downgrades the performance of the real time simulation. We leverage the fact that the nonlinear tongue deformation driven by EMA sensors is of a low rank and efficiently handle the volume preserving constraint in a reduced space.

Let V_0 denote the volume of the original tongue mesh Ω. It can be computed as $V_0 = \int_\Omega d\mathbf{x}$. Its deformed volume V_t at time t can be calculated similarly as: $V_t = \int_{\Omega_t} d\mathbf{y} = \int_\Omega |\mathbf{F}(\mathbf{x})| d\mathbf{x}$, where $\mathbf{y} = \mathbf{x} + \mathbf{u}$ is the deformed nodal position. Noting that $|\mathbf{F}| = |\mathbf{I} + \nabla\mathbf{u}| \approx 1 + \mathtt{div}\,\mathbf{u}$, the volume change between V_0 and V_t can be first-order approximated as:

$$\Delta V = V_t - V_0 = \int_\Omega (|\mathbf{F}(\mathbf{x})| - 1) d\mathbf{x} \approx \int_\Omega \mathtt{div}\,\mathbf{u} d\mathbf{x}. \tag{15}$$

We introduce a virtual pressure term $\mathbf{p} \in \mathbb{R}^n$ to cancel ΔV which results in a "pressure force" of $-\nabla\mathbf{p}$. Inserting $-\nabla\mathbf{p}$ into the time integration yields:

$$\Delta V \approx \mathtt{div}\left(\mathbf{M}^{-1}\nabla\mathbf{p} - \mathbf{v}\right) \cdot h, \tag{16}$$

where h is the time step size, which is set as 0.01 in our system. Discretizing Eq. (16) at each tetrahedral element on the mesh allows us to solve $\mathbf{p}$ by inverting an n by n matrix. This matrix is constant and can be pre-factorized. Thus, the runtime evaluation of $\mathbf{p}$ requires a complete $\mathbf{O}(n^2)$ forward-backward substitution at each time step. The resulting $\mathbf{p}$ is used to obtain a displacement correction $\Delta\mathbf{u} = -h \cdot \mathbf{M}^{-1}\nabla\mathbf{p}$. We follow the same idea as Irving and colleagues [22] and apply another velocity correction $\Delta\mathbf{v}$ to make the velocity field as divergence-free as possible. Doing so effectively stabilizes potential oscillations under nonlinear constraints.

Data-driven pressure subspace Since the deformable motion of the tongue driven by the EMA coils is obviously of low rank, we further construct a reduced *pressure subspace* and solve Eq. (16) within the subspace. Our method is data-driven, based on the recorded articulatory corpus consisting of 1,108 complete Chinese sentences. Each EMA frame i includes a vector $\mathbf{s}_i$ of 3D positions of TT, TB, TD and TR sensors. We evaluate the finite difference acceleration as: $(\mathbf{s}_{i+1} + \mathbf{s}_{i-1} - 2\mathbf{s}_i)/2\Delta t^2$ to obtain the inflection points of each sensor's trajectory. Frames with small acceleration magnitude are chosen as key frames (Fig. 7 right). While there are hundreds of thousands of EMA frames, we found that using 100 key frames is sufficient to construct a high-quality subspace for the volume correction.

For each selected key frame k_i, we solve a nonlinear static equilibrium by imposing constraint forces at nodal points corresponding to EMA sensors under the volume preserving constraint, and record the associated correcting pressure vector $\mathbf{p}_{k_i}$ such that:

$$\begin{bmatrix} \mathbf{K}(\mathbf{u}) & \mathbf{C}^\top(\mathbf{u}) \\ \mathbf{C}(\mathbf{u}) & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{0} \end{bmatrix}, \tag{17}$$

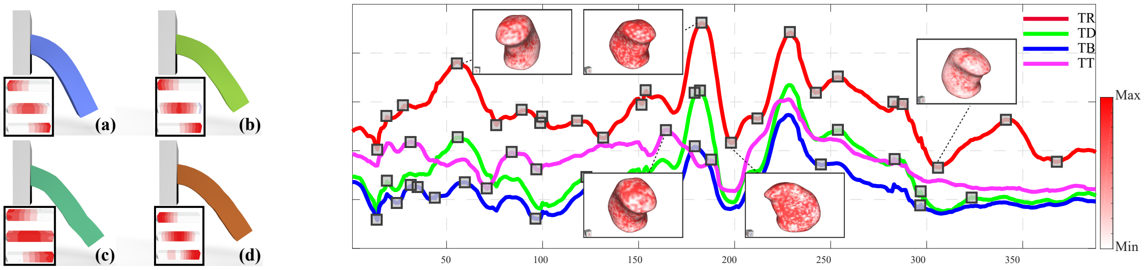


Figure 7: **Left:** In this illustrative example, we compare a simple bending simulation of a standard load-end cantilever beam (with three domains) using (a) our method, (b) weighting computed with a completely fixed boundary condition, (c) weighting computed along the direction perpendicular to the principle direction, and (d) using harmonic coordinates. **Right:** We pick key frames (grey blocks) by checking the finite difference acceleration magnitude of each EMA sensor and compute the corresponding pressure field. Selected pressure vectors and the corresponding tongue shapes are visualized as well using the red-white color map.

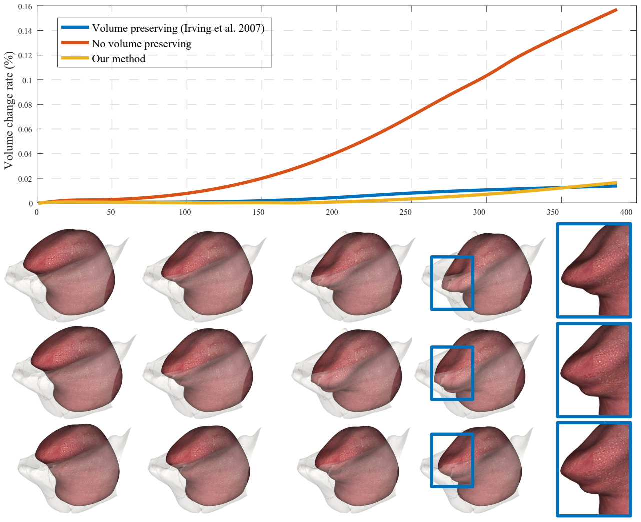


Figure 9: Applying the volume preserving constraint yields more natural tongue shapes, and our subspace volume preserving is able to effectively suppress volume change during tongue's deformation. The first row of snapshots is the shapes without volume preserving. The second and the third rows are the results using fullspace method and our method. It can be seen that our method is able to produce almost identical results compared to the fullspace volume correction. With our method, the volume change during the tongue simulation is always less than 2%.

where $\mathbf{C}(\mathbf{u})$ encodes the required nonlinear constraints for both position constraints at EMA sensors and volume preservation. Newton's method is used to solve this nonlinear problem, which requires the evaluation of the tangent stillness matrix $\mathbf{K}(\mathbf{u}^i)$ at each intermediate $\mathbf{u}^i$ of the i^{th} iteration as well as the Jacobi of the constraint matrix, which can be calculated as $\nabla\mathbf{C} = [\mathbf{B}_s^\top\ \mathbf{D}]^\top$. Here $\mathbf{B}_s$ is a constant binary matrix picking the nodal DOFs corresponding to the EMA sensors. $\mathbf{D} \in \mathbb{R}^{n\times 3n}$ is the discretized matrix representation of the `div` operator. In the dynamic integration, the first-order approximation of the volume change (e.g. Eq. (15)) is applied to an incremental displacement update occurring within a single time step. Together with the velocity correction, the volume preserving constraint can always be well satisfied. However, in our subspace construction a given key frame often corresponds to a deformed configuration of the tongue deviating significantly from the rest shape. Indeed, solving Eq. (16) is numerically equivalent to performing one Newton iteration to solve the nonlinear system. Typically, we need three to five iterations to fully suppress ΔV. Finally, a modified Gram-Schmidt process (MGS) is applied to all the computed $\mathbf{p}_{k_i}$, which serves as the basis vectors for the pressure subspace. In this manner the displacement and velocity corrections can be efficiently calculated within milliseconds and impose a nominal computation penalty to the simulator.

Other implementation details We employ Rayleigh damping and the mass and stiffness damping coefficients are set as 6.22 and 0.11 as reported in [12, 53]. In order to efficiently evaluate the reduced internal force and its gradient, we use the Cubature scheme proposed by An and colleagues [2]. The idea is to avoid evaluating $\widetilde{\mathbf{f}}_{int}$ and $\partial\widetilde{\mathbf{f}}_{int}/\partial\mathbf{q}$ at each element, which is a $\mathbf{O}(n)$ runtime procedure. Instead, the Cubature scheme selects a set of few key elements and approximates them as the weighted summation of per-element internal force and force gradient. We refer the reader to the related documents [2, 54] for a detailed exposition of the Cubature method. In our implementation, the training data for the Cubature is selected in a similar way as for constructing the pressure subspace, yet consists of 500 training poses. The DNN-based speech inversion feeds Eq. (11) the positional information of the TT, TB, TD and TR sensors based on the acoustic signals. We use the Lagrange multiplier method to deform the tongue mesh so that positional constraints can be precisely satisfied.

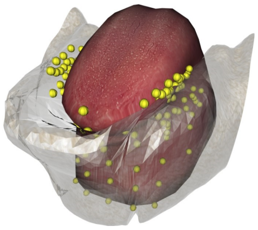


Figure 8: Collision detection is limited at few selected collision points.

We monitor the collisions between the tongue and the jaw. As the collision patterns between them are highly coherent, the collision detector simply tracks only selected collision points as shown in Fig. 8. If a collision is detected, a damped spring is applied to resolve it.

9 Experimental Result

In addition to the data acquisition equipment described in § 4, the numerical part of our framework was implemented using `Microsoft Visual C++ 2013` on a desktop PC with an `Intel i7-5960` CPU and 32GB of DDR4 RAM. The GUI was implemented using `QT`. All the numerical algorithms were implemented using the `Eigen C++` template (the Cholesky `LDLT` routine is used for solving Eq. (11)). Our simulation runs at 60+ FPS including collision and volume preservation.

Evaluation of the speech inversion During the speech inversion, in order to determine the number of hidden units of each layer, we first conduct an experiment on a neural network with one hidden layer. The number of hidden units varies from 50 to 1,600. The results indicate that the neural network with 400 hidden units should achieve a good performance. Therefore, we construct a deep neural network with 6 hidden layers. The momentum d (e.g. in Eq. (3)) is set to be 0.8. The initial learning rate is set to be 0.0004,

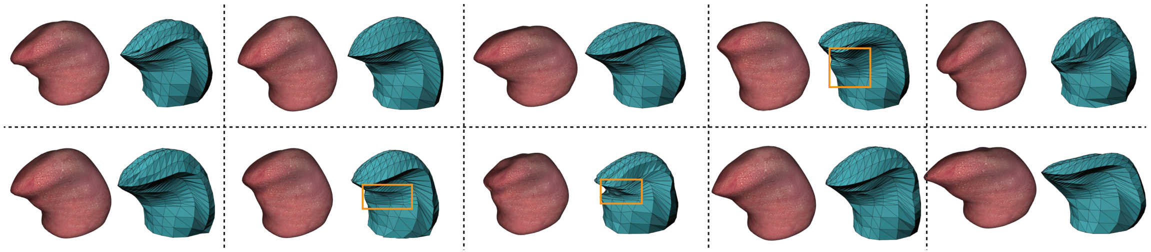
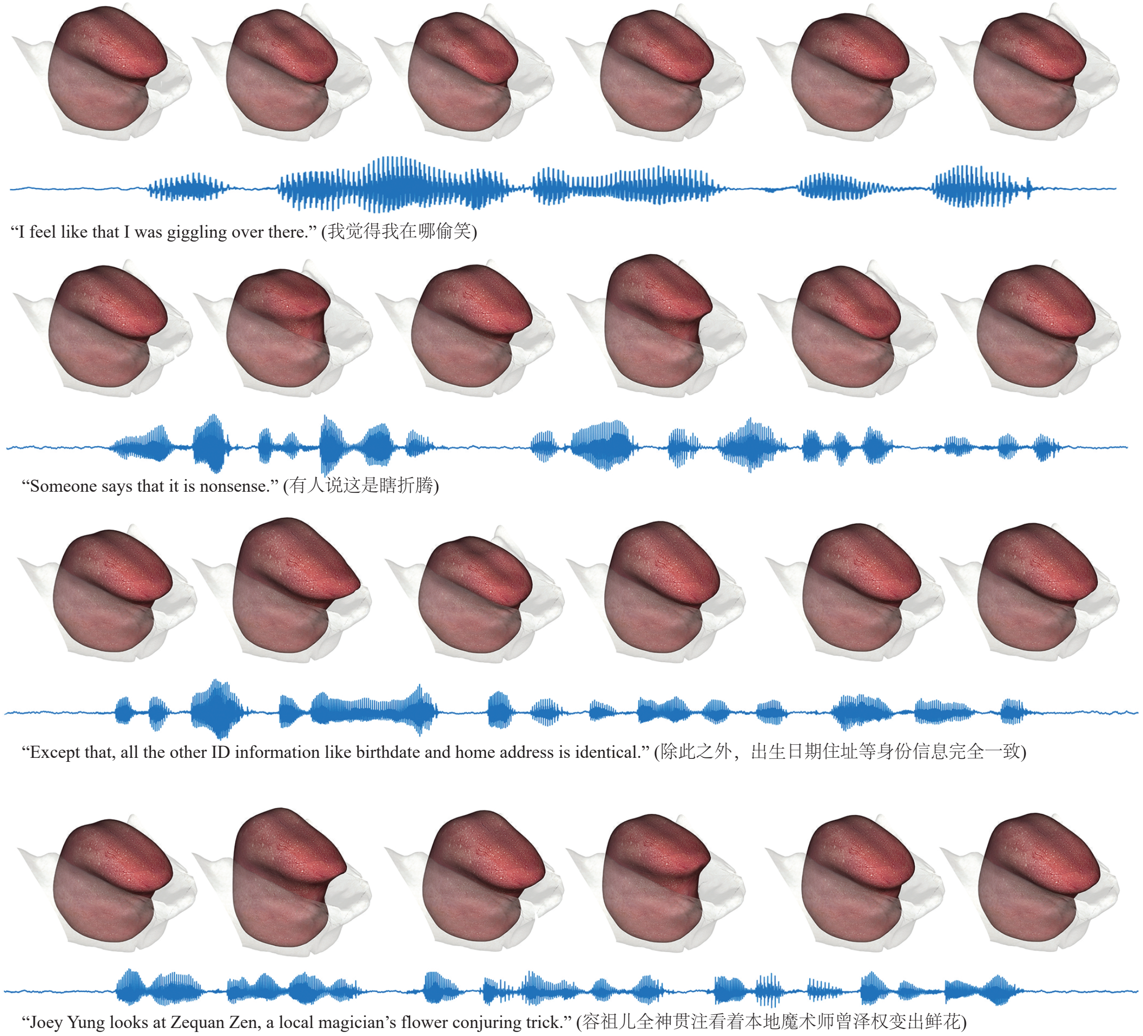


Figure 10: Side-by-side comparisons between simulated tongue shapes (textured) and real-world shapes extracted from MRI-CBCT fused images (in cyan)



“I feel like that I was giggling over there.” (我觉得我在哪偷笑)

“Someone says that it is nonsense.” (有人说这是瞎折腾)

“Except that, all the other ID information like birthdate and home address is identical.” (除此之外，出生日期住址等身份信息完全一致)

“Joey Yung looks at Zequan Zen, a local magician’s flower conjuring trick.” (容祖儿全神贯注看着本地魔术师曾泽权变出鲜花)

Figure 11: More snapshots of the tongue during speech production. The input acoustic signal waves are also provided.

and decays with the proportion of 0.9. Each mini-batch contains $1,024$ examples. The maximum number of training epoch is set to be 50. The evaluation of the DNN-based speech inversion is performed over the 110 utterances out of the collected corpus that do not participate in the DNN training. We compute the root mean-squared error (RMSE) defined as: $\varepsilon_{RMSE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}|\widetilde{\mathbf{x}}-\mathbf{x}|^2}$, for each EMA sensor to see how much deviation we have between the sensor’s position and DNN trained model. Here, $m = 110$. $\widetilde{\mathbf{x}}$ and $\mathbf{x}$ are the sensors’ positions as the output from the DNN and their observed coordinates. ε_{RMSE} for all the sensors is less than 3 *mm*. Specifically, the deviations are 2.93 *mm* for the TR sensor, 2.56 *mm* for the TD sensor, 1.2 *mm* for the TB sensor, and 0.87 *mm* for the TT sensor. We also compute the cross correlation coefficient to evaluate similarity of the motion trajectories. The correlation coefficient between the trained and real EMA sensors’ trajectory is 0.81. In addition, we further applied our DNN to the MOCHA database [55]. Our DNN model produces comparable results (average $\varepsilon_{RMSE} = 1.09$ *mm* and correlation is 0.89) as other paradigms for the inverse speech mapping (e.g. in [37]).

Evaluation of subspace volume preserving Next, we quantitatively evaluate the performance of the proposed subspace volume preserving method. Fig. 9 shows a representative example of a deformable tongue motion when the entire tongue mesh is moving downwards. Without enforcing the volume conservation constraint, the volume change of the entire mesh can be as high as 15%. Fullspace volume preservation as in [22] is able to correct this issue but takes $500-600$ *ms* to solve Eq. (16) in the fullspace. This correction must be calculated twice for both corrective displacement and velocity. Our subspace volume preserving algorithm can be completed less than 20 *ms* and the visual difference between the fullspace volume preservation and our method is indistinguishable.

Evaluation of the FEM simulator Evaluating the resulting deformed tongue shapes is crucial for us to understand the quality of the proposed simulator. However, there does not exist a "gold standard" that could serve as the ground truth to conclusively tell if a given tongue's shape is valid or not. Indeed, the current knowledge of tongue placement is quite limited even for certificated Speech-Language Pathologists (SLP) [30]. Previous work on the tongue modeling borrowed experiences from domain experts and conducted qualitative visual evaluations [57]. While such approach is able to more or less estimate the quality of the visual tongue model, it is highly subjective. Occasionally, even certificated SLPs are not able to tell if a motion looks "right" or not. In this work, thanks to the various medical imaging systems adopted, we are able to quantitatively evaluate the quality of the simulated tongue shapes. Our ground truth is obtained in a similar manner as for constructing a subject-specific tongue mesh (§ 4) by fusing and aligning both MRI and CBCT volumes. We compared the simulated shapes and the ones extracted from the 3D images for 10 representative poses. To the best of our knowledge, it is the first quantitative evaluation for FEM based tongue models that leverages full 3D real-world data. Fig. 10 reports the side-by-side comparison. We also computed the Hausdorff distance [10] between the simulated shape and captured ones. We first eliminate the shape differences induced by uniform scaling and rigid body transformation. To do so, a shape matching [31] is performed to find the optimal rotation between a pair of meshes (i.e. the simulated and captured ones). After that, a scaling factor s can be computed, and the Hausdorff distance is evaluated finally. The average shape difference is less than 5% of for all the 10 examples listed in Fig. 10. Visually, our simulator replicates real-world tongue shapes plausibly. However, it can also be seen from this comparison that localized denting deformation under the tongue tip is not well captured, which occurs due to the contraction of the underlaying muscle group. As our DOF assignment and domain decomposition are based on the placement of EMA sensors, and such localized sharp deformation is probably beyond our subspace expressivity. After all, we only use 150 DOFs to simulate the complex nonlinear motion of the tongue (over 30K fullspace DOFs). More results are reported in Fig. 11. The sound waves along with the original Chinese language and the corresponding English translations are also provided. We refer readers to the accompanying video for more animated results.

10 Limitation and Future Work

We propose a real-time system that creates a vivid VR representation of the human tongue in a sensor free manner based on input acoustic signals. To achieve this objective, we use a two-step inverse map: the first inverse map converts speech sounds to a predefined articulators' 3D trajectory, which is further utilized in an inverse dynamic based simulator. In order to achieve real-time performance, we carefully shape the simulator to find a good trade-off between the performance and accuracy. Our simulator is versatile and accommodates collision handling and volume preservation.

There are also several limitations of the current system, which leave us many exciting directions to explore in the near future. First of all, it is still unknown to us that using four EMA sensors at the mid-sagittal plane is the optimal setting for the follow-up inverse simulation or if extra sensors might improve our results. While putting quadratic domains according to the sensors' locations gives a satisfactory result in general, some local deformation is missed (highlighted in Fig. 10). Therefore, performing the domain partition in a way that better reflects the tongue's anatomy [15] may be a potential improvement. We will work closely with our collaborators and domain experts to find the answer to this fundamental question. We will also apply our system to other languages. Since our DNN-based method works well for the MOCHA database, it is expected that our system should perform well on the English language. Using an active model instead of a passive model to synthesize the articulation of the tongue is also an ambitious future work for us. Combing machine learning, physics-based modeling, and multi-modality data fusion seems to be a worthy idea.

11 Acknowledgement

Authors would like to thank anonymous reviewers for theirs constructive comments. Ran Luo and Yin Yang are partially supported by National Science Foundation (CHS−1464306 and CNS−1637092) as well as equipment support from UNM OVPR award. Qiang Fang is supported by the Natural Science Foundation of China (NSFC) (61175016, 61573254 and 61304250), Key Fund projects of 61233009, and the financial support from CASS Innovation Project "Articulatory model for pronunciation training". Jianguo Wei is supported in part by National Basic Research Program of China (2013CB329305), and NSFC (61175016). Weiwei Xu is supported by NSFC (61272392 and 61322204).

References

[1] Y. S. Akgul, C. Kambhamettu, and M. Stone. Automatic extraction and tracking of the tongue contours. *IEEE Transactions on Medical Imaging*, 18(10):1035–1045, 1999.

[2] S. S. An, T. Kim, and D. L. James. Optimizing cubature for efficient integration of subspace deformations. *ACM Trans. Graph.*, 27(5):165:1–165:10, Dec. 2008.

[3] B. S. Atal, J. J. Chang, M. V. Mathews, and J. W. Tukey. Inversion of articulatory-to-acoustic transformation in the vocal tract by a computer-sorting technique. *The Journal of the Acoustical Society of America*, 63(5):1535–1555, 1978.

[4] P. Badin, G. Bailly, L. Reveret, M. Baciu, C. Segebarth, and C. Savariaux. Three-dimensional linear articulatory modeling of tongue, lips and face, based on mri and video images. *Journal of Phonetics*, 30(3):533–553, 2002.

[5] T. Baer, J. Gore, S. Boyce, and P. Nye. Application of mri to the analysis of speech production. *Magnetic resonance imaging*, 5(1):1–7, 1987.

[6] J. Barbič and D. L. James. Real-time subspace integration for st. venant-kirchhoff deformable models. In *ACM transactions on graphics (TOG)*, volume 24, pages 982–990. ACM, 2005.

[7] R. W. Bisseling and A. L. Hof. Handling of impact forces in inverse dynamics. *Journal of biomechanics*, 39(13):2438–2444, 2006.

[8] S. Buchaillard, P. Perrier, and Y. Payan. A biomechanical model of cardinal vowel production: Muscle activations and the impact of gravity on tongue positioning. *The Journal of the Acoustical Society of America*, 126(4):2033–2051, 2009.

[9] M. G. Choi and H.-S. Ko. Modal warping: Real-time simulation of large rotational deformation and manipulation. *IEEE Transactions on Visualization and Computer Graphics*, 11(1):91–101, Jan. 2005.

[10] P. Cignoni, C. Rocchini, and R. Scopigno. Metro: measuring error on simplified surfaces. In *Computer Graphics Forum*, volume 17, pages 167–174. Wiley Online Library, 1998.

[11] M. Fu, M. S. Barlaz, J. L. Holtrop, J. L. Perry, D. P. Kuehn, R. K. Shosted, Z.-P. Liang, and B. P. Sutton. High-frame-rate full-vocal-tract 3d dynamic speech imaging. *Magnetic resonance in medicine*, 2016.

[12] J.-M. Gérard, J. Ohayon, V. Luboz, P. Perrier, and Y. Payan. Indentation for estimating the human tongue soft tissues constitutive law: ap-

plication to a 3d biomechanical model. In *Medical Simulation*, pages 77–83. Springer, 2004.

[13] J.-M. Gérard, P. Perrier, and Y. Payan. 3d biomechanical tongue modeling to study speech production. *Speech production: Models, phonetic processes, and techniques*, pages 85–102, 2006.

[14] K. GRENABO. Atlas of topographical and applied human anatomy. head and neck, 1965.

[15] B. P. Halpern. Functional anatomy of the tongue and mouth of mammals. In *Drinking behavior*, pages 1–92. Springer, 1977.

[16] M. N. Hegde. *Introduction to communicative disorders*. Pro Ed, 1995.

[17] M. Hirayama, E. Vatikiotis-Bateson, and M. Kawato. Inverse dynamics of speech motor control. In *Advances in neural information processing systems*, pages 1043–1050, 1994.

[18] S. Hiroya and M. Honda. Estimation of articulatory movements from speech acoustics using an hmm-based speech production model. *IEEE Transactions on Speech and Audio Processing*, 12(2):175–185, 2004.

[19] T. Hueber, G. Aversano, G. Cholle, B. Denby, G. Dreyfus, Y. Oussar, P. Roussel, and M. Stone. Eigentongue feature extraction for an ultrasound-based silent speech interface. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, volume 1, pages I–1245. IEEE, 2007.

[20] P. W. Iltis, J. Frahm, D. Voit, A. A. Joseph, E. Schoonderwaldt, and E. Altenmüller. High-speed real-time magnetic resonance imaging of fast tongue movements in elite horn players. *Quantitative imaging in medicine and surgery*, 5(3):374–381, 2015.

[21] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[22] G. Irving, C. Schroeder, and R. Fedkiw. Volume conserving finite element simulations of deformable models. *ACM Trans. Graph.*, 26(3), July 2007.

[23] R. D. Kent. Research on speech motor control and its disorders: A review and prospective. *Journal of Communication disorders*, 33(5):391–428, 2000.

[24] S. A. King and R. E. Parent. A 3d parametric tongue model for animated speech. *The Journal of Visualization and Computer Animation*, 12(3):107–115, 2001.

[25] M. Li, C. Kambhamettu, and M. Stone. Automatic contour tracking in ultrasound images. *Clinical linguistics & phonetics*, 19(6-7):545–554, 2005.

[26] S. G. Lingala, B. P. Sutton, M. E. Miquel, and K. S. Nayak. Recommendations for real-time speech mri. *Journal of Magnetic Resonance Imaging*, 43(1):28–44, 2016.

[27] S. G. Lingala, Y. Zhu, Y.-C. Kim, A. Toutios, S. Narayanan, and K. S. Nayak. A fast and flexible mri system for the study of dynamic vocal tract shaping. *Magnetic resonance in medicine*, 2016.

[28] A. J. Lundberg and M. Stone. Three-dimensional tongue surface reconstruction: Practical considerations for ultrasound data. *The Journal of the Acoustical Society of America*, 106(5):2858–2867, 1999.

[29] J. Luo, K. Ying, and J. Bai. Savitzky–golay smoothing and differentiation filter for even number data. *Signal Processing*, 85(7):1429–1434, 2005.

[30] S. Mcleod. Speech–language pathologists' knowledge of tongue/palate contact for consonants. *Clinical linguistics & phonetics*, 25(11-12):1004–1013, 2011.

[31] M. Müller, B. Heidelberger, M. Teschner, and M. Gross. Meshless deformations based on shape matching. In *ACM SIGGRAPH 2005 Papers*, SIGGRAPH '05, pages 471–478, New York, NY, USA, 2005. ACM.

[32] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 807–814, 2010.

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

[34] A. Pentland and J. Williams. Good vibrations: Modal dynamics for graphics and animation. *SIGGRAPH Comput. Graph.*, 23(3):207–214, July 1989.

[35] J. Perkell, M. Matthies, H. Lane, F. Guenther, R. Wilhelms-Tricarico, J. Wozniak, and P. Guiod. Speech motor control: Acoustic goals, saturation effects, auditory feedback and internal models. *Speech communication*, 22(2):227–250, 1997.

[36] K. Richmond. *Estimating articulatory parameters from the acoustic speech signal*. PhD thesis, University of Edinburgh, 2002.

[37] K. Richmond. A trajectory mixture density network for the acoustic-articulatory inversion mapping. In *Interspeech*, 2006.

[38] T. H. Shawker, B. Sonies, M. Stone, and B. J. Baum. Real-time ultrasound visualization of tongue movement during swallowing. *Journal of Clinical Ultrasound*, 11(9):485–490, 1983.

[39] H. Si. Tetgen, a delaunay-based quality tetrahedral mesh generator. *ACM Transactions on Mathematical Software*, 41(2):11, 2015.

[40] E. Sifakis and J. Barbic. Fem simulation of 3d deformable solids: A practitioner's guide to theory, discretization and model reduction. In *ACM SIGGRAPH 2012 Courses*, SIGGRAPH '12, pages 20:1–20:50, New York, NY, USA, 2012. ACM.

[41] B. C. Sonies, T. H. Shawker, T. E. Hall, L. H. Gerber, and S. B. Leighton. Ultrasonic visualization of tongue motion during speech. *The Journal of the Acoustical Society of America*, 70(3):683–686, 1981.

[42] I. Stavness, J. Lloyd, and S. Fels. Inverse-dynamics simulation of muscular-hydrostat finite-element models. In *23rd International Society of Biomechanics Congress (ISB)*, volume 933, 2011.

[43] I. Stavness, J. Lloyd, Y. Payan, and S. Fels. Towards speech articulation simulation with a dynamic coupled face-jaw-tongue model. In *ISSP'2011*, pages 1–4, 2011.

[44] I. Stavness, J. E. Lloyd, and S. Fels. Automatic prediction of tongue muscle activations using a finite element model. *Journal of biomechanics*, 45(16):2841–2848, 2012.

[45] M. Stone. A three-dimensional model of tongue movement based on ultrasound and x-ray microbeam data. *The Journal of the Acoustical Society of America*, 87(5):2207–2217, 1990.

[46] M. Stone, E. P. Davis, A. S. Douglas, M. N. Aiver, R. Gullapalli, W. S. Levine, and A. J. Lundberg. Modeling tongue surface contours from cine-mri images. *Journal of speech, language, and hearing research*, 44(5):1026–1040, 2001.

[47] M. Stone and A. Lundberg. Three-dimensional tongue surface shapes of english consonants and vowels. *The Journal of the Acoustical Society of America*, 99(6):3728–3737, 1996.

[48] S. Suzuki, T. Okadome, and M. Honda. Determination of articulatory positions from speech acoustics by applying dynamic articulatory constraints. In *ICSLP*, 1998.

[49] H. Takemoto. Morphological analyses of the human tongue musculature for three-dimensional modeling. *Journal of Speech, Language, and Hearing Research*, 44(1):95–107, 2001.

[50] J. Teran, S. Blemker, V. N. T. Hing, and R. Fedkiw. Finite volume methods for the simulation of skeletal muscle. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '03, pages 68–74, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.

[51] T. Toda, A. W. Black, and K. Tokuda. Statistical mapping between articulatory movements and acoustic spectrum using a gaussian mixture model. *Speech Communication*, 50(3):215–227, 2008.

[52] B. Uria, I. Murray, S. Renals, and K. Richmond. Deep architectures for articulatory inversion. In *INTERSPEECH*, pages 867–870, 2012.

[53] F. Vogt, J. E. Lloyd, S. Buchaillard, P. Perrier, M. Chabanas, Y. Payan, and S. S. Fels. Efficient 3d finite element modeling of a muscle-activated tongue. In *International Symposium on Biomedical Simulation*, pages 19–28. Springer, 2006.

[54] C. von Tycowicz, C. Schulz, H.-P. Seidel, and K. Hildebrandt. An efficient construction of reduced deformable objects. *ACM Trans. Graph.*, 32(6):213:1–213:10, Nov. 2013.

[55] A. Wrench. The mocha-timit articulatory database, 1999.

[56] Z. Wu, K. Zhao, X. Wu, X. Lan, and H. Meng. Acoustic to articulatory mapping with deep neural network. *Multimedia Tools and Applications*, 74(22):9889–9907, 2015.

[57] Y. Yang, X. Guo, J. Vick, L. G. Torres, and T. F. Campbell. Physics-based deformable tongue visualization. *IEEE transactions on visualization and computer graphics*, 19(5):811–823, 2013.

[58] L. Zhang and S. Renals. Acoustic-articulatory modeling with the trajectory hmm. *IEEE Signal Processing Letters*, 15:245–248, 2008.