Vocal Tract Contour Tracking in rtMRI Using Deep Temporal Regression Network

Sasan Asadiabadi, *Student Member, IEEE*, Engin Erzin, *Senior Member, IEEE*

Abstract—Recent advances in real-time Magnetic Resonance Imaging (rtMRI) provide an invaluable tool to study speech articulation. In this paper, we present an effective deep learning approach for supervised detection and tracking of vocal tract contours in a sequence of rtMRI frames. We train a single input multiple output deep temporal regression network (DTRN) to detect the vocal tract (VT) contour and the separation boundary between different articulators. The DTRN learns the non-linear mapping from an overlapping fixed-length sequence of rtMRI frames to the corresponding articulatory movements, where a blend of the overlapping contour estimates defines the detected VT contour. The detected contour is refined at a post-processing stage using an appearance model to further improve the accuracy of VT contour detection. The proposed VT contour tracking model is trained and evaluated over the USC-TIMIT dataset. Performance evaluation is carried out using three objective assessment metrics for the separating landmark detection, contour tracking and temporal stability of the contour landmarks in comparison with three baseline approaches from the recent literature. Results indicate significant improvements with the proposed method over the state-of-the-art baselines.

Index Terms—Real-time magnetic resonance imaging (rtMRI), vocal tract, speech production, appearance model, contour detection, deep neural network

I. INTRODUCTION

KNOWLEDGE on vocal tract (VT) dynamics is the basis to understand the human speech production system and plays a significant role in many speech processing applications. Recent developments in the real-time Magnetic Resonance Imaging (rtMRI) provide rich information on the dynamics of the entire midsagittal vocal tract [1], [2]. In this context, development of automatic algorithms to detect the landmarks defining the contours of the vocal tract is necessary. To name a few applications, VT contour estimation is used as a pre-processing step to obtain and study the time evolution of the vocal tract cross sectional area function in [3]. In another work, Toutios et al. [4] propose a text-to-speech synthesis system using the estimated VT contours in rtMRI. Analyzing the time-varying dynamic changes of the vocal tract is crucial for several other applications such as identification and treatment of speech disorders [5], [6], speech production research [7], [8], [9] and articulatory synthesis [10].

Vocal tract contour detection is challenging due to the poor resolution of rtMRI, grainy noise, lack of textural variations, complex anatomy and the rapidly varying shapes of the vocal

tract. Two widely used approaches for the vocal tract airway boundary estimation using rtMRI are: (i) unsupervised approaches, which utilize computer vision methods to detect the VT boundaries, and (ii) supervised approaches, which learn to estimate VT boundaries from data. In the VT contour detection literature, the supervised approaches are mainly addressed using either landmark based or segmentation based contour estimation schemes. In the following, we review the recent literature of the VT boundary estimation based on these two lines of approaches.

1) *Landmark-based contour estimation*: In landmark based VT boundary estimation, a series of anatomical landmarks on the VT, separating the tissue from the airway, are identified in each image. Figure 1 presents sample landmarks for separating the articulator of interest (AoI) regions. Using unsupervised computer vision and/or supervised machine learning algorithms, the VT landmarks are extracted from the image features in the MRI frames.

In an early approach on the estimation of the VT contours in rtMRI, Bresch et al. [11] employ active contour model (ACM), aka SNAKE algorithm [12], with optical flow [13] to track the VT contours in rtMRI sequences. They manually initialize the first frame in a given image sequence and the contour is then tracked in the subsequent images. The major drawback of this method is that, when the articulators are in contact, they cause the estimation error to propagate through the image sequence, requiring frequent manual correction.

Bresch and Narayanan present a novel approach for segmentation of upper airway rtMRI in the frequency domain [14]. They use an anatomically informed model for the vocal tract that fits to an observed image through a gradient descent optimization procedure. Despite the visually accurate results, their approach suffers from high computational complexity. To get well-captured contours for the vocal tract, 650 gradient descent optimization steps are carried out and the process takes several minutes for each image on a modern desktop computer as reported in [14].

A semi-automatic rapid VT airway tissue boundary segmentation is proposed by Proctor et al. [15]. In this work, the image intensity profile along the Ohman's grid line system [16] is investigated. A graph is constructed by connecting all the local minima in the intensity profile on all of the grid lines. Obtaining the optimal path through the constructed graph, using the Dijkstra algorithm, yields the central airway path. The lower and upper VT boundaries are then estimated by locating the first point on the grid lines from either side of the center-line, which crosses a specified intensity threshold.

Kim et al. [17] propose an enhanced vocal tract contour

S. Asadiabadi is with the Electrical & Electronics Engineering Department, Koç University, Istanbul, Turkey (e-mail: sabadi15@ku.edu.tr).

E. Erzin is with the KUIS AI Lab, Computer Engineering Department and Electrical & Electronics Engineering Department, Koç University, Istanbul, Turkey (e-mail: eerzin@ku.edu.tr).

tracking in MRI videos. Their approach is quite similar to Proctor's [15] in the sense that the vocal tract center-line is estimated by minimizing an objective score using the Viterbi algorithm. The constructed graph in this method is not just limited to the local minima along the grid lines, but using all of the pixels on each line, which results in a quite high run time of the algorithm. Later, at each grid-line, the VT boundaries are determined by finding the first pixels crossing a fixed intensity threshold, from either side of the center-line. A potential shortcoming of this approach is to ignore certain regions such as velum opening/closing and the sublingual cavity, in the VT boundary representation. Furthermore this method does not provide identification of different regions of interest to allow automatic extraction of tract variables.

An active shape model (ASM) based estimation of VT boundaries is proposed by Silva et al. [18]. In this study, two shape models (oral and nasal) are constructed from a set of manually segmented contours. Then, for the VT contour detection, a model selection is performed with the analysis of the neighborhood around the velar port to check if the velum is open (nasal model) or closed (oral model). The main limitation of this study is the size of the training data. Building a shape and an appearance model from a small dataset makes it hard to generalize the model to different speakers and various configurations of the VT. Furthermore, VT contour detection is executed independently for each frame that yields VT contour tracking on a frame-by-frame basis.

Labrunie et al. [19] propose an articulator-specific multiple linear regression (MLR) method to perform the VT contour estimation. Their model predicts the contour coordinates as a linear combination of pixel intensities in the region of each articulator of interest and further refines the prediction using a modified ASM method. However, as in [18], their approach utilizes a small training dataset and suffers to generalize for different speakers and various configurations of the tract.

In another study, Asadiabadi and Erzin [20] propose a combination of ACM and ASM to overcome the shortcomings of the SNAKE algorithm in [11], when the articulators are in contact. In this study, an appearance model is built for the VT and is utilized as a contributing term to the energy minimization in the curve evolution process. A motion energy term is introduced as well to model the temporal nature of the articulatory movements.

Recently, Takemoto et al. [21] utilize the Dlib machine learning library [22] to estimate the VT contours of five speech organs. Their approach performs an accurate tracking of articulatory movements on unseen data using a few manually annotated images in the training set. Major shortcoming of this study is that their model training and experimental results are presented over a single subject only, which makes it hard to generalize over other subjects.

2) *Segmentation-based contour estimation*: Segmentation based VT boundary estimation approaches are typically carried out by pixel level segmentation of an MRI image, where the tissue and background class labels are assigned to each pixel in the image. The VT contours are then retrieved from the segmented MRI frame at a post-processing stage. Segmentation based approaches are mainly accomplished via supervised

Fig. 1. A sample vocal tract contour C is shown as continuous curve, highlighted in different colors describing different articulator of interest (AoI) regions. Separating landmark set $L = [u^1, u^2, \dots, u^6]$, are marked with the white points.

learning methods utilizing deep neural networks.

Somandepalli et al. [23] introduce a semantic edge detection algorithm for air-tissue boundary detection using convolutional neural networks with an encoder-decoder architecture. In this method, the VT contour estimation task is formulated as a multi-class pixel labeling problem, where the positive class comprises pixels forming the outline contour of each articulator of interest and the remaining tissue/air pixels form the negative class. A greedy algorithm is then utilized to draw the contours from the probability map of the positive class pixels predicted by the network.

Asadiabadi and Erzin [24] propose a heatmap regression based VT landmark localization using fully convolutional networks (FCN). In their algorithm, a heatmap containing the image intensity information around a landmark, is generated for each VT landmark using a Gaussian kernel. The proposed FCN comprises a regressor, estimating the heatmap associated with each landmark and a classifier, predicting the articulator label of each landmark. The landmarks are then reconstructed from the estimated heatmaps and refined in a spatio-temporal stabilization scheme using the Kalman filter.

Several other methods on semantic segmentation based estimation of vocal tract in MRI frames are proposed using fully convolutional networks [25] and convolutional encoder-decoder architectures [26], [27]. In these works, an MR image is segmented into three separate tissue regions and the VT boundary is obtained in a post-processing stage via an edge detection scheme. In a recent study, a convolutional LSTM network is trained to detect and label the articulatory contours using the temporal dynamics together with the spatial structure [28]. Utilizing the temporal context of the articulatory movements is a key factor to obtain a robust vocal tract contour detection, and includes challenging problems to be addressed in the impending literature.

In this study, we propose a robust VT contour detection and tracking system. Unlike the previous approaches, which perform the contour estimation on frame basis, the proposed system utilizes a sequence-to-sequence estimation scheme. In this regard, the key contributions of this study can be summarized as: i) A new deep temporal regression network

Fig. 2. Block diagram of the proposed VT contour tracker. The system is fed with a sequence of consecutive MRI frames and predicts the ZCA representation of the landmark locations and VT contour at the output. Note the highlighted index in the output stream.

is proposed for learning both the local and temporal contexts from a sequence of MR images to estimate the VT contour and the separation boundary between different articulators; ii) a blend of overlapping contour estimates defines the detected VT contour; and iii) an appearance model based contour refinement enhances the tracking precision of the detected VT contours.

The reminder of this paper is organized as follows. In section II, we provide a detailed description of the methodology proposed for vocal tract contour estimation in rtMRI data. Experimental evaluations are given and discussed in section III. Finally, section IV presents the conclusions.

II. METHODS

In this study, we approach the VT contour estimation as a sequence-to-sequence learning problem, where the aim is to estimate the coordinates of the airway tissue boundary from a sequence of rtMRI frames capturing the vocal tract region. This task is performed by training a mapping function from an overlapping fixed length sequence of MRI frames to the corresponding articulatory output streams. In this framework, we consider two output streams: i) a sequence for encoding the coordinates of VT contour in a frame, and ii) a sequence for representing the separation points of different articulators along the VT. Hence, we model the problem as a single input multiple output multivariate regression. The proposed VT contour estimation utilizes a Deep Temporal Regression Network (DTRN) to estimate a sequence of articulatory locations from the given MRI sequence. We introduce a novel Appearance Model based Contour Refinement (AMCR) scheme as a post-processing to the DTRN to further improve the quality of the contour estimation.

A. Dataset

In this study, the USC-TIMIT database [1] is used in training and testing of the models for vocal tract contour detection and tracking. The USC-TIMIT is a large audio-visual dataset containing 10 speakers (5F/5M) quoting the

phonetically diverse TIMIT sentences. The videos are mid-sagittal MRI frames, recorded at a frame rate of 23.18 fps and a spatial resolution of 68×68 pixels over 20×20 cm (approximately 2.9 mm pixel width). Synchronized audio is recorded at 20 kHz sampling rate. Phonetic transcriptions of the sentences are available in the dataset. The USC-TIMIT includes 92 videos for each subject with average duration of 24 seconds, resulting in approximately 6 hours of recorded data. The USC-TIMIT database also provides a set of ground-truth labels of the VT contours for three subjects (F1, F5 and M3), which are generated using the spatial frequency domain-based segmentation method introduced by Bresch and Narayanan [14].

Despite the availability of more recent rtMRI databases, such as the dataset discussed in [2], [23], which attains higher spatial and temporal resolutions, we choose to use the USC-TIMIT database in this study. Our main motivation for this choice is to utilize phonetically diverse speech utterances of the USC-TIMIT to better model the temporal nature of VT contours during the speech articulation.

B. Feature Representation

A key factor in training a neural network is the choice of features representing the inputs and outputs of the model. In this study we utilize a sequence of raw MR images as input visual features for training a regression network to estimate the articulatory streams.

1) *Visual Features*: In the proposed framework, VT contour tracking task is modeled as a multivariate regression problem, where the VT contour coordinates are estimated from the image intensities. The mapping function between the image intensity space and the VT contour coordinates can be conducted in a variety of ways. One approach, as presented in [19], is the MLR, which maps a PCA representation of the image intensities for an articulator to a PCA representation of the vocal tract contour of that articulator, using a linear regression.

In this study, we utilize the raw MR images as the input to the regression network. The visual input features at the j^{th}

frame are set as the raw image intensities, normalized between $[0, 1]$, and represented as $\mathbf{h}_j \in \mathbb{R}^{68 \times 68}$. Unlike the state of the art methods, which are performing the detection on a frame-by-frame basis, in this study we carry out the VT contour detection over a group of consecutive frames to capture the temporal context of the rtMRI sequence. A visual feature is formed by stacking K_v consecutive frames depth-wise as, $\mathbf{H}_j = [\mathbf{h}_{j-\Delta_v}, \dots, \mathbf{h}_j, \dots, \mathbf{h}_{j+\Delta_v}]$, where $\mathbf{h}_j \in \mathbb{R}^{68 \times 68}$ is the normalized image intensity map and j is the frame index. The resulting visual feature $\mathbf{H}_j^v$ is $68 \times 68 \times K_v$ dimensional with $K_v = 2\Delta_v + 1$.

2) *Articulatory Features*: We define the boundary between the airway and the tissue along the vocal tract as the VT contour. Vocal tract contour is described with a set of $M = 80$ equidistant points along the tract's upper and lower profile, spanning the VT from glottis to the lips, as highlighted in different colors in Figure 1, and represented as $\mathbf{C}_j = [\mathbf{v}_j^1, \mathbf{v}_j^2, \dots, \mathbf{v}_j^M]$, where $\mathbf{v}_j^m = [x_j^m, y_j^m]$ is the contour point coordinates, j is the frame index and m is the landmark index in each frame.

A compact VT representation is defined by removing the strong spatial correlation between VT contour points using a shape model. For this purpose, we build a statistical shape model over the set of training contours $\mathbf{C} = \{\mathbf{C}_j\}_{j=1}^J$ by applying Zero Component Analysis (ZCA) to represent the variation of data around the mean contour. The ZCA projection minimally distorts the original phase of the data and targets to preserve the orientation of the original data points. Hence, ZCA projects highly correlated contour space to an uncorrelated parameter space by keeping the transformed data as similar as to the original data. This makes ZCA a global normalization tool.

We define a projection from the contour space $\mathbf{C}_j$ to the parameter feature space as,

$$\mathbf{f}_j^c = (\mathbf{C}_j - \bar{\mathbf{C}})\mathbf{W}_c, \quad (1)$$

where $\mathbf{f}_j^c \in \mathbb{R}^{1 \times 2M}$ is the contour feature vector at frame j , $\mathbf{W}_c = \mathbf{V}^\top \text{diag}(\lambda_1^{-1/2}, \lambda_2^{-1/2}, \dots, \lambda_{2M}^{-1/2})\mathbf{V}$ is called the *whitening matrix*, $\mathbf{V} = [\mathbf{V}_1, \dots, \mathbf{V}_{2M}]$ is the eigenvectors of the covariance matrix $\Sigma = \mathbf{C}^\top \mathbf{C}$, λ_i 's are the corresponding eigenvalues and $\bar{\mathbf{C}}$ is the mean contour. Given the contour feature vector $\mathbf{f}_j^c$, the corresponding VT contour $\mathbf{C}_j$ in the original space is reconstructed as,

$$\hat{\mathbf{C}}_j = \bar{\mathbf{C}} + \mathbf{f}_j^c \mathbf{W}_c^{-1}, \quad (2)$$

where $\mathbf{W}_c^{-1} = \mathbf{V} \text{diag}(\lambda_1^{1/2}, \lambda_2^{1/2}, \dots, \lambda_{2M}^{1/2})\mathbf{V}^\top$ is the *de-whitening matrix*.

Vocal tract contour can be segmented based on articulators to construct articulator of interest (AoI) regions, which would be valuable for the extraction of articulatory features. We as well target to segment VT contour into several AoI regions, which are considered as A1: *lower lips*, A2: *tongue*, A3: *epiglottis and trachea*, A4: *pharyngeal wall*, A5: *velum*, A6: *hard palate* and A7: *upper lips*. Note that several tract variables can be measured given the precise contour coordinates of each AoI; for example Lip Aperture (LA) can be calculated as the minimum distance between A1 and

Algorithm 1: VT Contour and AoI Tracking

Input: Sequence of MR frames: $\{\mathbf{h}_j\}$ for $j = 1, 2, \dots$

Output: Estimated VT contour coordinates for each

AoI: $\{\mathbf{A}_j^1, \mathbf{A}_j^2, \dots, \mathbf{A}_j^N\}$ for $j = 1, 2, \dots$

VT contour and landmark regression

for $j = 1, 2, \dots$ **do**

Construct the temporal visual feature

$\mathbf{H}_j \leftarrow [\mathbf{h}_{j-\Delta_v}, \dots, \mathbf{h}_j, \dots, \mathbf{h}_{j+\Delta_v}] \in \mathbb{R}^{68 \times 68 \times K_v}$

Perform the regression

$\hat{\mathbf{F}}_j^c, \hat{\mathbf{F}}_j^l \leftarrow \text{DTRN}(\mathbf{H}_j)$

end

VT AoI contour tracking

for $j = 1, 2, \dots$ **do**

Temporal estimate of contour and landmark

$\hat{\mathbf{f}}_j^c \leftarrow \{\hat{\mathbf{F}}_{j-\Delta_a}^c, \dots, \hat{\mathbf{F}}_{j+\Delta_a}^c\}$ using (4)

$\hat{\mathbf{f}}_j^l \leftarrow \{\hat{\mathbf{F}}_{j-\Delta_a}^l, \dots, \hat{\mathbf{F}}_{j+\Delta_a}^l\}$ using (4)

Projection from contour parameters to coordinates

$\hat{\mathbf{C}}_j = [\hat{\mathbf{v}}^1, \dots, \hat{\mathbf{v}}^M] \leftarrow \bar{\mathbf{C}} + \hat{\mathbf{f}}_j^c \mathbf{W}_c^{-1}$

$\hat{\mathbf{L}}_j = [\hat{\mathbf{u}}^1, \dots, \hat{\mathbf{u}}^N] \leftarrow \bar{\mathbf{L}} + \hat{\mathbf{f}}_j^l \mathbf{W}_l^{-1}$

Apply appearance based contour refinement

$\hat{\mathbf{C}}_j \leftarrow \text{AMCR}(\hat{\mathbf{C}}_j)$

Apply articulatory separation

Initialize $m^- = 1$

for $n = 1$ to $N - 1$ **do**

$m^+ = \arg \min_{m^- \leq i \leq M} \|\hat{\mathbf{v}}^i - \hat{\mathbf{u}}^n\|_2$

$\hat{\mathbf{A}}_j^n \leftarrow [\hat{\mathbf{v}}^{m^-}, \dots, \hat{\mathbf{v}}^{m^+}]$

$m^- = m^+ + 1$

end

$\hat{\mathbf{A}}_j^N \leftarrow [\hat{\mathbf{v}}^{m^-}, \dots, \hat{\mathbf{v}}^M]$

end

A7 curves. We represent these 7 AoIs with the separating landmark location feature, defined as $\mathbf{L}_j = [\mathbf{u}_j^1, \mathbf{u}_j^2, \dots, \mathbf{u}_j^N]$, where $N = 6$ and $\mathbf{u}_j^n = [x_j^n, y_j^n]$ are the separating landmark coordinates. Note that the separating landmark location $\mathbf{L}_j$ is a subset of the VT contour $\mathbf{C}_j$. A sample sketch of the landmark locations is given in Figure 1.

Similar to the contour model, a statistical shape model is generated for the separating landmarks using the ZCA algorithm. The landmark feature vector $\mathbf{f}_j^l \in \mathbb{R}^{1 \times 2N}$ is extracted as in (1) using the corresponding whitening matrix $\mathbf{W}_l$, which is defined over the separating landmark locations $\mathbf{L} = \{\mathbf{L}_j\}_{j=1}^J$.

The temporal dependency within the VT contours and the landmark locations is modeled over an overlapping sliding window of features. The temporal feature vector of the VT contour is defined as, $\mathbf{F}_j^c = [\mathbf{f}_{j-\Delta_a}^c, \dots, \mathbf{f}_j^c, \dots, \mathbf{f}_{j+\Delta_a}^c]^\top \in \mathbb{R}^{2MK_a \times 1}$, where $K_a = 2\Delta_a + 1$ is the length of the articulatory temporal window. Similarly, the temporal feature vector of the landmark locations is represented as $\mathbf{F}_j^l \in \mathbb{R}^{2NK_a \times 1}$.

C. Deep Temporal Regression Network Architecture

As given in Figure 2, we construct a deep temporal regression network (DTRN), which receives temporal visual feature $\mathbf{H}_j$ and maps this input to the desired targets as the temporal VT contour and landmark location features $(\mathbf{F}_j^c, \mathbf{F}_j^l)$.

Fig. 3. Positive and negative sample extraction in the appearance model training procedure. A) Positive class sampling around an on the VT boundary point, v . B) Negative class sampling around an off the VT boundary point, w . C) Normalized gradient intensity vector Γ , for the sampled points around v and w . Best viewed in color.

At the early stage of the DTRN, a CNN network is applied to extract features from the input images. Any off-the-shelf deep convolutional network such as VGG [29] or ResNet [30] architecture can be used at the feature extraction stage. In this work, a ResNet based architecture, ResNet18, is used as the feature extractor, since the ResNet architectures are used successfully as a feature extraction network in many computer vision and image recognition tasks. The stacked temporal visual feature, $\mathbf{H}_j$, is fed to the ResNet18 based feature extraction architecture, which generates 512 feature maps, each with dimension 2×2 . A global average pool (GAP) layer follows the ResNet18 and reduces the size of the representation to $(1, 1, 512)$ by taking the average of each feature map. The Flatten layer follows the GAP layer just to flatten the input without changing the contained information. The ResNet18, GAP and Flatten layers builds a backbone network for the feature extraction. Then, the target VT contour and landmark location features $\mathbf{F}_j^c$ and $\mathbf{F}_j^l$ are estimated separately throughout two distinct fully connected layers, which are connected to the output of the backbone network. During the training, weights of the fully connected layers and the backbone network are updated simultaneously employing the mini-batch stochastic gradient descent to minimize the sum of Mean Squared Error (MSE) loss for the two output sequences. Each fully connected layer uses a hyperbolic tangent activation function followed by dropout with 25% probability to counteract overfitting. The model is trained for 100 epochs with early stopping enabled and a batch size of 128. An adaptive learning rate reduction method is employed to further reduce the MSE loss.

D. VT Contour and Landmark Estimation

The deep temporal regression network receives the temporal visual feature $\mathbf{H}_j$, which spans K_v MRI frames, and performs regression of the target VT contour and landmark location features $\hat{\mathbf{F}}_j^c$ and $\hat{\mathbf{F}}_j^l$, which are both temporal estimates of K_a

Algorithm 2: Appearance Model based Contour-Refinement Algorithm (AMCR)

Input: Estimated VT contour $\hat{\mathbf{C}}_t = [\hat{\mathbf{v}}^1, \dots, \hat{\mathbf{v}}^M]$

Output: Refined VT contour $\tilde{\mathbf{C}}_t = [\tilde{\mathbf{v}}^1, \dots, \tilde{\mathbf{v}}^M]$

Initialize:

for $m = 1, \dots, M$ **do**

$\hat{\mathbf{v}}_0^m = \hat{\mathbf{v}}^m$

$\mathbf{V}^m = \{\hat{\mathbf{v}}_{-n_s}^m, \dots, \hat{\mathbf{v}}_0^m, \dots, \hat{\mathbf{v}}_{n_s}^m\}$

end

Refine:

for $iter = 1, \dots, I$ **do**

for $m = 2, \dots, M - 1$ **do**

for $l_m = -n_s, \dots, n_s$ **do**

$E_{l_m}^{min} = \min_{-n_s \leq l_{m-1} \leq n_s} \{E_{l_{m-1}, l_m}\}$

$L_{l_m}^{min} = \arg \min_{-n_s \leq l_{m-1} \leq n_s} \{E_{l_{m-1}, l_m}\}$

end

end

Back-trace initialization: $l_1^* = l_M^* = 0$

Back-trace: **for** $m = M - 1, \dots, 2$ **do**

$l_m^* = L_{l_{m+1}^*}^{min}$

end

$E^{min} = E_{l_M^*}^{min}$

Update: **for** $m = 1, \dots, M$ **do**

$\hat{\mathbf{v}}_0^m = \hat{\mathbf{v}}_{l_m^*}^m$

$\mathbf{V}^m = \{\hat{\mathbf{v}}_{-n_s}^m, \dots, \hat{\mathbf{v}}_0^m, \dots, \hat{\mathbf{v}}_{n_s}^m\}$

$\tilde{\mathbf{v}}^m = \hat{\mathbf{v}}_0^m$

end

end

frames centered at frame j . Let us represent the frame level estimated VT contour feature as,

$$\hat{\mathbf{F}}_j^c = [\hat{\mathbf{f}}_{j,j-\Delta_a}^c, \dots, \hat{\mathbf{f}}_{j,j-1}^c, \hat{\mathbf{f}}_{j,j}^c, \hat{\mathbf{f}}_{j,j+1}^c, \dots, \hat{\mathbf{f}}_{j,j+\Delta_a}^c], \quad (3)$$

where the frame level estimated landmark location feature can also be defined similarly.

Note that regression is performed on the temporal visual feature for each MRI frame and the two consecutive estimates $\hat{\mathbf{F}}_j^c$ and $\hat{\mathbf{F}}_{j+1}^c$ overlap $K_a - 1$ frames. Frame level contour feature vector $\hat{\mathbf{f}}_{j,j}^c$ is the estimate of the contour vector $\mathbf{f}_j^c$ at frame j associated with the temporal visual input feature $\mathbf{H}_j$, whereas $\hat{\mathbf{f}}_{j+1,j}^c$ represents the estimate of the same contour vector $\mathbf{f}_j^c$ at frame j associated with the temporal visual input feature $\mathbf{H}_{j+1}$. Hence, both $\hat{\mathbf{f}}_{j,j}^c$ and $\hat{\mathbf{f}}_{j+1,j}^c$ constitute the estimate of the same contour vector $\mathbf{f}_j^c$ that are extracted using the input visual features $\mathbf{H}_j$ and $\mathbf{H}_{j+1}$, respectively. In the proposed configuration, we have K_a estimates for the contour vector $\mathbf{f}_j^c$ from all possible overlaps at the output of the deep temporal regression network. We construct the final estimate of the contour vector $\hat{\mathbf{f}}_j^c$ as the mean of overlapping estimates as,

$$\hat{\mathbf{f}}_j^c = \frac{1}{K_a} \sum_{k=-\Delta_a}^{\Delta_a} \hat{\mathbf{f}}_{j+k,j}^c. \quad (4)$$

The final estimate of the VT landmark location can be extracted similarly and defined as $\hat{\mathbf{f}}_j^l$. Frame level mean

Fig. 4. Duration distributions of the Arpabet phonemes at video frame level over the 270 videos from subjects F1, F5 and M3.

blending block in Figure 2 presents a visualization for the mean calculation over the overlapping estimates.

The VT contour $\hat{C}_j$ and the landmark locations $\hat{L}_j$ are then reconstructed respectively from the estimated contour $\hat{f}_j^c$ and landmark $\hat{f}_j^l$ features using (2). As a final stage for the proposed VT contour tracking system, we apply a post-processing refinement based on an appearance model to the estimated VT contours as described in the following section. Having computed the refined estimate of VT contour, the closest point of the contour to each landmark coordinate is calculated to locate the AoI boundaries, as described in Algorithm 1.

E. Appearance Model based VT Contour Refinement

The data-driven deep VT regression network may fail to properly capture the rapid articulatory movements in some highly deformable regions, such as tongue and velum. We introduce an appearance model based contour refinement (AMCR) post-processing to enhance the tracking precision of the estimated VT contours. For this purpose, we train a statistical deformable appearance model for the vocal tract, based on the grey level intensity textures around each contour point. The appearance model training is inspired from traditional Active Appearance Model (AAM) [31]. Similar to the classical AAMs, we introduce an iterative statistical model refinement, given a current estimate of the model parameters.

Appearance model is a generative model, which aims to recover parametric description of a certain object through optimization. We setup the VT appearance model as a binary classification problem; any point in the vocal tract region either belongs to the positive class ϕ_p indicating that the point lays on the vocal tract boundary, or the negative class ϕ_n stating that it is located off the boundary. In this scheme, all the points located on a VT contour belong to the positive class and any other pixel on the image plane belongs to the negative class. Given an initial estimate of the VT contour, coordinates of the points are optimized iteratively, such that each contour point satisfies to be in the positive class with maximum probability during the iterative fitting process.

The image intensity structure around a point is utilized as a discriminating feature between the positive and negative classes. Image intensity values are computed on $2\kappa + 1$ points,

which are sampled equidistantly along the perpendicular to the m -th VT contour point at $\mathbf{v}^m$. The points on the perpendicular of $\mathbf{v}^m$ are sampled at a unit pixel distance from each other denoted as $\mathbf{P}^m = \{\mathbf{p}_{-\kappa}^m, \dots, \mathbf{p}_0^m, \dots, \mathbf{p}_{\kappa}^m\}$, where the center point coincides with the VT contour point such that $\mathbf{p}_0^m = \mathbf{v}^m$ as illustrated in Figure 3. The image intensity vector at the sampled points over the frames is computed using the cubic interpolation and labeled as $+1$, indicating the positive class. A negative intensity profile is as well extracted on $\mathbf{w}^m$, which is chosen as a nearby point to $\mathbf{v}^m$ and located off the VT boundary, with $2\kappa + 1$ sampled points along the perpendicular of $\mathbf{w}^m$. To create a balanced positive and negative intensity profile, $\mathbf{w}^m$ is chosen randomly from the set $\mathbf{P}^m - \{\mathbf{p}_0^m\}$. The intensity vector for $\mathbf{w}^m$ is labeled as -1 , indicating the negative class.

Intensity vector, which is representing positive or negative classes for the VT contour point $\mathbf{v}^m$, is represented as $\Gamma^m = [\mathbf{dh}_{-\kappa}^m, \dots, \mathbf{dh}_0^m, \dots, \mathbf{dh}_{\kappa}^m]$ and labeled as y^m , where $\mathbf{dh}_l^m$ is the normalized derivative image intensity at point $\mathbf{p}_l^m$ and $y^m \in \{+1, -1\}$. A binary classifier is trained for each VT contour point $\mathbf{v}^m$ over a sequence of training data, $\{(\Gamma_j^m, y_j^m)\}_{j \in \mathcal{V}}$, where j is the sequence index of the training data. Hence, the binary classifier receives a $2\kappa + 1$ dimensional input intensity vector and outputs the probability of the positive class.

Given the estimated VT contour points $\hat{\mathbf{C}} = [\hat{\mathbf{v}}^1, \hat{\mathbf{v}}^2, \dots, \hat{\mathbf{v}}^M]$ at the reconstructed output of the DTRN as the initial contour position, the contour is evolved utilizing the trained appearance model through several iterations. For each contour point $\hat{\mathbf{v}}^m$, a nearby search region is selected as $2n_s + 1$ candidate locations, sampled at a unit pixel distance from each other, centered to and along the perpendicular of $\hat{\mathbf{v}}^m$. We denote the set of candidate locations for contour point $\hat{\mathbf{v}}^m$ as $\mathbf{V}^m = \{\hat{\mathbf{v}}_{-n_s}^m, \dots, \hat{\mathbf{v}}_0^m, \dots, \hat{\mathbf{v}}_{n_s}^m\}$. Note that $\mathbf{V}^m$ represents the set of possible refinement locations for the current point $\hat{\mathbf{v}}^m$. For each candidate point $\hat{\mathbf{v}}_l^m$, the $(2\kappa + 1)$ -length image intensity vector is computed as Γ_l^m , centered to the candidate point. The intensity vectors are then evaluated with the trained binary classifier and the probability of belonging to the positive class is assigned to each candidate point. The contour evolution is then executed by minimizing

Fig. 5. Validation loss curve for different visual feature extraction backbones.

an energy function defined between consecutive vertices as,

$$E_{l_{m-1}, l_m} = -E_{app}(\hat{\mathbf{v}}_{l_m}^m) + \alpha E_{con}(\hat{\mathbf{v}}_{l_{m-1}}^{m-1}, \hat{\mathbf{v}}_{l_m}^m), \quad (5)$$

where l_m and l_{m-1} are the candidate location indices selected from $\mathbf{V}^m$ and $\mathbf{V}^{m-1}$, respectively. The energy term E_{app} is the appearance energy, which moves the contour points toward the maximum positive class probability locations and E_{con} is the continuity energy, which prevents the independent movements of contour points by keeping the points at a constant distance from each other. These two energy functions are defined as,

$$E_{app}(\hat{\mathbf{v}}_{l_m}^m) = p(\hat{\mathbf{v}}_{l_m}^m \in \phi_p) \quad (6)$$

$$E_{con}(\hat{\mathbf{v}}_{l_{m-1}}^{m-1}, \hat{\mathbf{v}}_{l_m}^m) = \left| D_v - \left\| \hat{\mathbf{v}}_{l_m}^m - \hat{\mathbf{v}}_{l_{m-1}}^{m-1} \right\|_2 \right|, \quad (7)$$

where $E_{app}(\hat{\mathbf{v}}_{l_m}^m)$ is the probability of $\hat{\mathbf{v}}_{l_m}^m$ being in the positive class, $E_{con}(\hat{\mathbf{v}}_{l_{m-1}}^{m-1}, \hat{\mathbf{v}}_{l_m}^m)$ is the difference of separations on each side of the contour point $\hat{\mathbf{v}}_{l_m}^m$ and D_v is the mean distance between all contour points, given as,

$$D_v = \frac{1}{M-1} \sum_{m=1}^{M-1} \left\| \hat{\mathbf{v}}^{m+1} - \hat{\mathbf{v}}^m \right\|_2. \quad (8)$$

Minimization of the energy function in (5) is solved using dynamic programming to find a set of contour points globally minimizing the energy function at each iteration. At each contour vertex $\hat{\mathbf{v}}^m$ and for all candidate positions $\hat{\mathbf{v}}_{l_m}^m$ of the vertex, the optimal position of the predecessor is computed. The optimized contour locations at each iteration is then computed by back-tracing. A detailed description of the contour refinement is given in Algorithm 2. The optimal set $\hat{\mathbf{V}}$ is the final estimate of the VT contour points.

III. EXPERIMENTAL EVALUATIONS

We investigate the performance of the proposed VT contour tracking system against the ground-truth labeled contours. Detecting the VT contour in a data driven manner requires a sufficiently large training set. Annotating the ground labeled VT contours in such a scale is extremely time consuming and

Fig. 6. Exhaustive grid search for the optimum input-output sliding window lengths K_v and K_a in terms of the MSD loss of reconstructed VT contours from the validation set. Smaller window lengths tend to have higher estimation error.

requires great patience. To avoid the manual annotation procedure, we utilize the ground-truth labels of the VT contours which are provided by the USC SAIL group. The ground-truth labels are provided for three subjects (F1, F5 and M3) and are generated using the spatial frequency domain-based segmentation method introduced by Bresch and Narayanan [14]. Figure 1 illustrates an example of VT contour ground-truth label. As seen in this sample figure, the generated labels are nearly as well as a manual annotation. However, it should be noted that the spatial frequency domain-based segmentation method is computationally expensive and requires several minutes per frame in its original implementation.

We as well create a small set of manually labeled VT contours over three selected videos, from three subjects, that includes 966 frames in total. Performance evaluations over the manually labeled contours are as well presented in comparison to the large scale tests for the ground-truth labeled data. The manually labeled set is publicly available at¹.

To train the DTRN and AMCR networks, we utilize around 2 hours of data with available ground truth vocal tract contours, for subjects F1, F3 and M3 in the USC-TIMIT database. The selected data consists of 270 videos (90 videos per subject) with approximately 150K frames in total, containing high phonetic variability as shown by the phoneme distribution in Figure 4. For tuning hyper-parameters of the models, we randomly split the video data into training and validation sets with a validation split factor of 10% and run the experiments to tune the models with the best fitting parameters. Later, experimental evaluations are performed under subject-dependent and subject-independent settings. The subject-dependent experiments are carried out with 5-fold cross-validation, where the whole video data is split into 5 equal splits. At each fold, models are trained on 4 splits (80% of the video data) with the tuned parameters, and evaluated on the left-out split. Training the models with 80% of the data at each fold, ensures

¹<https://github.com/sanasadiabadi/rtMRI-track>

Fig. 7. An example of vocal tract contour and landmark detection, refinement and articulatory separation: A) Estimated VT separating landmarks vs the ground labeled landmarks; B) Estimated VT contour using DTRN; C) Refined VT contour after AMCR for tongue region; D) Articulatory separation.

the availability of samples from each subject for a subject-dependent evaluation. The subject-independent experiments are carried out with the leave-one-subject-out (LOSO) cross-validation, where at each fold of the LOSO, models are trained on data from two subjects and tested on the third subject's data. All regression and classification models are implemented in Keras [32] with Tensorflow [33] backend and trained on a NVIDIA GeForce GTX 1660 Ti GPU.

A. Baseline Methods

We compare our proposed VT landmark tracking algorithm against three state-of-the-art baselines as described below.

Stabilized Fully Convolutional Network (S-FCN) is proposed in [24] and performs the VT landmark localization via heatmap regression using a FCN. In this approach, each VT contour coordinate is transformed into a heatmap using a Gaussian kernel and the FCN learns the probability of a landmark presence at each pixel of a given image. The algorithm ignores the spatial and temporal correlation between the VT landmarks. However, the authors introduce a landmark spatio-temporal stabilization using the Kalman filter as a post-processing.

Dlib [34] is widely used in facial landmark localization tasks, where it is an implementation of Ensemble of Regression Trees (ERT) presented in [35]. A recent study successfully uses Dlib to detect vocal tract landmarks [21]. During the training of the Dlib, we observe that it is important to initialize the VT landmarks carefully. As in the conventional facial detection where Dlib finds a bounding box around the face prior to landmarking, the algorithm requires detecting a rectangle area around the vocal tract to perform a decent vocal tract contour localization. We utilize Python implementation of the Dlib² for the VT bounding box detection and contour estimation tasks.

²<https://github.com/davisking/dlib>

Fig. 8. Average MSD of the 5-fold cross validation over the defined AoIs; A1: lower lips, A2: tongue, A3: epiglottis and trachea, A4: pharyngeal wall, A5: velum, A6: hard palate and A7: upper lips. Error bars indicate the standard deviation.

Multiple Linear Regression (MLR) is presented in [19] to detect vocal tract landmarks in a cascade of three linear models. Separate shape models are generated for MR image intensities and the VT contours using PCA. A linear regressor is then trained to learn the mapping between image shape parameters and the VT contour parameters. Unlike the original approach in which separate shape and regression models are trained for each articulator, we train a single MLR for the VT. However we utilize the Dlib-based VT detector to isolate the tract area in the image for a more compact shape model of image intensities and hence more accurate results. Further details of the MLR can be found in [19]. We use the scikit-learn library [36] in Python to implement the PCA and the linear regression model.

B. Assessment Metrics

For the objective evaluations of the proposed system, we define three assessment metrics for the separating landmark detection, contour tracking and temporal stability of the contour landmarks.

The performance of the VT separating landmark detection network is evaluated using a point-to-point error metric. Let's consider the reference separating landmark location vector as $\mathbf{L} = [\mathbf{u}^1, \mathbf{u}^2, \dots, \mathbf{u}^N]$ and its estimate as $\hat{\mathbf{L}} = [\hat{\mathbf{u}}^1, \hat{\mathbf{u}}^2, \dots, \hat{\mathbf{u}}^N]$. Then we define the point-to-point error metric E_{p2p} as,

$$E_{p2p}(\mathbf{L}, \hat{\mathbf{L}}) = \frac{1}{N} \sum_{n=1}^N \|\mathbf{u}^n - \hat{\mathbf{u}}^n\|_2. \quad (9)$$

The performance of the proposed VT contour tracking system is evaluated using the Mean Sum of Distance (MSD) error metric, which is calculated across the estimated and the ground-truth contours. MSD measures the distance between two contours by computing the distance between closest vertices of each contour. Given two set of contours $\mathbf{U} =$

Fig. 9. Snapshots of estimated VT contours using the proposed DTRN+AMCR model on a sample test video in the subject dependent experiment. It is observed that the proposed model generalizes well for all tract regions and under different articulatory configurations.

$[\mathbf{u}^1, \mathbf{u}^2, \dots, \mathbf{u}^M]$ and $\mathbf{V} = [\mathbf{v}^1, \mathbf{v}^2, \dots, \mathbf{v}^M]$, the MSD error metric E_{MSD} is defined as,

$$E_{MSD}(\mathbf{U}, \mathbf{V}) = \frac{1}{2M} \sum_{i=1}^M \min_j \|\mathbf{v}^i - \mathbf{u}^j\|_2 + \min_j \|\mathbf{u}^i - \mathbf{v}^j\|_2. \quad (10)$$

Note that the ground-truth and estimated contours can have different number of vertices, and the MSD error metric can be calculated separately for each AoI region. The MSD for the n -th AoI is computed as $E_{MSD}(A^n, \hat{A}^n)$.

The third metric is defined as the temporal stability error, which measures the frame-to-frame change in each estimated VT contour landmark with respect to the true landmark changes. For a test video of length T frames, given the ground label contours at frame t as $\mathbf{V}_t = [\mathbf{v}_t^1, \mathbf{v}_t^2, \dots, \mathbf{v}_t^M]$ and their estimated values as $\hat{\mathbf{V}}_t = [\hat{\mathbf{v}}_t^1, \hat{\mathbf{v}}_t^2, \dots, \hat{\mathbf{v}}_t^M]$, the temporal stability error, E_{TSE} , is defined as,

$$E_{TSE} = \frac{1}{TM} \sum_{t=1}^T \sum_{m=1}^M \|(\mathbf{v}_t^m - \mathbf{v}_{t-1}^m) - (\hat{\mathbf{v}}_t^m - \hat{\mathbf{v}}_{t-1}^m)\|_2. \quad (11)$$

C. Hyper-parameter Tuning for the Regression and Classification Networks

1) *DTRN Tuning*: A key factor in the proposed VT tracking system is the feature extraction backbone at the early stage of the regression network. To justify the selection of the ResNet backbone, we compare the performance of the ResNet-based tracking system with the VGG19 feature extraction backbone. Figure 5 shows the validation MSE loss on the ZCA parameter space during the training for different feature extraction backbones. We observe that the ResNet-based feature extraction is performing remarkably better than the VGG19 in our VT tracking setup. The ResNet also obtains a much better inference run-time performance as presented in Table I.

Other important tuning parameters are the sliding window lengths, K_v and K_a . The input and output window sizes should be large enough to capture the salient information in the visual features and contour parameter spaces. We perform a grid search over specified parameter values to tune the sliding

window lengths. Figure 6 presents the MSD loss as defined in (10) on the VT contours, reconstructed from the output of the DTRN, for various values of K_v and K_a . We observe that smaller window sizes tend to have higher estimation error and as we increase the window length, the error starts decreasing. However with very large window sizes the estimation error rises again. With an exhaustive grid search, we observe that $(K_v, K_a) = (5, 5)$ gives the best performance on the validation data.

2) *AMCR Tuning*: The main tuning factors in the proposed contour refinement scheme are the normalized gradient intensity vector length $2\kappa+1$ and the number of candidate locations for each contour point $2n_s+1$. Parameter κ should be large enough to capture the discriminating texture around the on-and-off contour points. Parameter n_s controls the maximum pixel movement per iteration. A small value of n_s prevents the contour to move toward the airway tissue boundaries while a large value could cause divergence. With a grid search optimization, $\kappa = 8$ is observed to yield the maximum binary classification accuracy and $n_s = 3$ obtains the minimum contour estimation error over the validation data.

A multilayer perceptron (MLP) network is trained to classify an image pixel into positive and negative classes. The MLP network consists of two fully connected layers with 1024 hidden neurons each, followed by a *Relu* activation function. Outputs of the classifier are generated using a *sigmoid* function at the final layer. The trained classifier achieves above 98% accuracy on the validation set, for all contour points.

The energy combination weight in (5) is selected as $\alpha = 0.1$ and we consider the number of iterations as $I = 10$.

D. Subject Dependent Error Assessment

1) *Separating Landmark Detection*: Figure 7 illustrates an example of the detected separating landmark locations in comparison to the ground-truth reference landmarks. The landmark estimation errors, E_{p2p} , for the proposed network architectures are given in Table I. The ResNet based regression system delivers 1.97 point-to-point error, which is significantly lower as compared to the VGG19 based backbone. Moreover, the AMCR post-processing further reduces the point-to-point error to 1.83.

TABLE I

OBJECTIVE ASSESSMENT RESULTS FOR THE SUBJECT-DEPENDENT EXPERIMENTS OF THE PROPOSED DTRN REGRESSION NETWORKS AND THE AMCR POST-PROCESSING IN COMPARISON WITH THE BASELINES. RUN-TIME DEFINES THE AVERAGE VT CONTOUR ESTIMATION TIME PER MRI FRAME. THE MSD, TSE AND P2P COLUMNS PRESENT AVERAGE ERROR VALUES AND THEIR STANDARD DEVIATIONS OVER THE 5-FOLD CV IN THE SUBJECT-DEPENDENT EXPERIMENT.

	Method	Run Time [ms]	$E_{MSD}[mm]$	$E_{TSE}[mm]$	$E_{p2p}[mm]$
Backbone	DTRN+AMCR	135	2.78 ± 0.15	4.24 ± 0.13	1.83 ± 0.08
	DTRN-ResNet18	60	3.24 ± 0.17	4.36 ± 0.16	1.97 ± 0.10
	DTRN-VGG19	110	4.82	5.93	3.45
Baseline	S-FCN	118	3.55 ± 0.25	4.52 ± 0.12	2.43 ± 0.08
	DLIB	25	4.23 ± 0.24	8.78 ± 0.21	-
	MLR	40	5.46 ± 0.30	7.74 ± 0.23	-

2) *VT Contour Detection*: In order to investigate the effect of the AMCR, we evaluate the performance of the proposed DTRN and the refined DTRN (DTRN+AMCR) separately. Figure 8 illustrates the MSD over the seven defined AoI regions. We observe that the proposed ResNet based DTRN with the AMCR post-processing outperforms the baselines for all AoIs. The AMCR reduces the estimation error for all articulatory regions. However, improvement due to the AMCR is mainly remarkable for the tongue region.

An example of the estimated VT contour, before and after the appearance model based refinement, is given in Figure 7. As depicted in this figure, AMCR improves the quality of contour estimation in the regions that the DTRN fails to capture the contours accurately. The improvement with the AMCR is particularly noticeable for the tongue region in Figure 7.

Several snapshots of the VT articulatory contours, which are estimated with the proposed DTRN+AMCR pipeline, are given in Figure 9. While these snapshots indicate that the proposed DTRN+AMCR model well captures the tract boundaries in all regions and under different articulatory configurations, a possible AoI classification error is also visible in the first snapshot due to a challenging calculation for the closest point of the contour to the landmark coordinate. Video samples of the tracked MRI contours are also available for online demonstration³.

Table I presents the objective error evaluations for the subject dependent experiments, for the proposed DTRN architecture and the AMCR post-processing in comparison with the state-of-the-art baselines. The E_{MSD} results in this table are given as the average MSD over the AoIs. We observe that the proposed DTRN with ResNet backbone and the AMCR post-processing achieves significantly lower error values as compared to the baselines. As observed from Table I, the proposed DTRN+AMCR model achieves the lowest temporal stability error among the baselines. The temporal regression based VT contour estimation yields promising results in terms of utilizing temporal coherency of the MRI data as it achieves even higher temporal stability than the S-FCN, which utilizes a Kalman filter based temporal stabilization post-processing.

The inference run time is reported as the average time elapsed to perform the contour detection for each frame. The proposed refined DTRN+AMCR tracker evaluates the contour estimation at ~ 7 frames-per-second on a GPU. While the ResNet-based DTRN's speed is comparable to the baselines, the post processing AMCR is the bottleneck and increases the overall run time of the proposed algorithm. Since the AMCR is significantly contributing to the tongue region contour refinement, one can only choose to use the AMCR for the tongue contour to attain a faster inference.

Beside the performance evaluation over different regions of the VT, we investigate the performance of the proposed DTRN+AMCR tracker on several selected phones. The list of phones used for evaluation is [B, F, T, G, IY, AA, UW], which involves the sound production in different places of articulation with highly constricted or fully closed articulatory gestures. The average MSD error assessments for the selected phones in the speaker-dependent experiments are visualized in Figure 10. The evaluation is performed only on the highly deformable AoIs: A1 lower lip, A2 tongue, A5 velum and A7 upper lip, in which the contour estimation errors tend to be

Fig. 10. Average MSD on the test MRI videos in the 5-fold speaker-dependent cross validation, for several selected phones over the highly deformable AoIs, A1: lower lip, A2: tongue, A5: velum, and A7: upper lip. Articulators involved in the sound production tend to have higher estimation error.

³Sample videos of the tracked MRI contours are available at <http://mvgl.ku.edu.tr/demos/>

TABLE II
THE AVERAGE MSD ERROR VALUES ON THE MANUALLY LABELED
CONTOURS AND THE USC PROVIDED GROUND-TRUTH LABELS.

Method	Manual Labels	USC Labels
DTRN+AMCR	2.17	2.35
S-FCN	3.06	3.28
DLIB	3.95	4.21
MLR	5.11	5.28

higher. The illustrated heatmap values in this figure suggest that the higher MSD errors are observed for the bilabial /B/ at the lips, for the labiodental /F/ at the lower lip, tongue and velum, for the alveolar /T/ and the velar /G/ phones at the tongue and velum, for the front vowel /IY/ at the tongue, and for the back vowel /UW/ at the tongue and velum. For these selected set of phones, the AoIs with the high MSD errors are observed to be inline with the places of articulation of the phones, where we expect the highest deviations to occur during the articulation.

3) *Evaluations on the manually labeled contours:* The manually labeled VT contours are extracted for three selected videos, one per subject, among the all available 270 videos of the subjects F1, F5 and M3. The VGG Image Annotation (VIA)⁴ tool is utilized for the contour annotations. The resulting manually labeled data includes 966 video frames in total. For the evaluations of the models over the manually labeled contours, the models are trained using all the data except the manually labeled ones, and the contour detection performances are extracted over the three selected videos. Performance evaluations of the proposed method and baselines in terms of the average MSD errors are given in Table II, over the manually labeled contours and as well over the USC ground-truth annotations for the same three videos. The ordering and tendency of the MSD performances of the proposed and baseline methods are consistent with the subject-dependent evaluations in Table I. Furthermore, we observe a slight decrease in the MSD performances for the manually labeled contours compared to the USC ground-truth labels.

E. Subject Independent Error Assessment

Although it is challenging to setup subject-independent evaluations over only three subjects, we choose to present LOSO cross-validation evaluations as compared to the well performing subject-dependent results. Average MSD error values over the LOSO cross-validation in subject-independent experiments are given in Table III. The proposed DTRN+AMCR approach attains a lower average MSD as compared to the baselines. As we compare the subject dependent and independent evaluations in Table I and Table III, we clearly observe a level of degradation in the average MSD error values of the subject-independent evaluations for all methods. However we expect that including more subjects in the network training would help the model to be more generalizable across subjects, hence

TABLE III
THE AVERAGE MSD ERROR VALUES FOR THE SUBJECT-INDEPENDENT
EXPERIMENTS.

Test Subject	DTRN+AMCR	S-FCN	DLIB	MLR
F1	4.55	4.81	5.88	6.20
F5	4.68	4.92	5.53	6.85
M3	4.96	5.12	6.05	6.96
ALL	4.73	4.95	5.82	6.67

making subject-independent results more consistent with the subject-dependent ones.

IV. CONCLUSION

In this paper, we introduce a data driven vocal tract contour detection using a pioneering deep temporal regression network. Our approach utilizes the spatial and temporal context in the rtMRI sequence as well as a novel contour refinement algorithm to perform the vocal tract contour tracking. We train the network to learn the non-linear temporal mapping from an overlapping fixed length sequence of MRI frames to the corresponding sequence of articulatory movements. The estimated overlapping articulatory information is then blended using a temporal averaging scheme to obtain the vocal tract contour and articulatory separating landmarks. Finally, the proposed appearance model based contour refinement technique further improves quality of the vocal tract contour detection. The proposed approach generalizes well across various sub-regions of the tract and under different articulatory configurations. The subject-dependent experiments indicate that the proposed method is more generalizable across subjects against the baselines. It should be noted that the performance of the proposed method is also limited to the quality of the ground-truth VT contour labels, which are generated utilizing a highly computational automated algorithm. Future work may target training larger networks on richer data with more subjects, richer phonetic diversity and more accurate ground-truth labels. The robustness of the method to the potential scaling, rotation and shift problems across various subjects can be further investigated and assessed. We as well will focus on optimizing the proposed network by integrating the appearance model based refinement scheme into the temporal regression network to obtain an end-to-end VT contour detection scheme.

ACKNOWLEDGMENT

The authors would like to thank USC SAIL group for providing the annotated vocal tract contours in the USC-TIMIT database.

⁴<http://www.robots.ox.ac.uk/~vgg/software/via/>

REFERENCES

- [1] S. Narayanan, A. Toutios, V. Ramanarayanan, A. Lammert, J. Kim, S. Lee, K. Nayak, Y. Kim, Y. Zhu, L. Goldstein, D. Byrd, E. Bresch, P. Ghosh, A. Katsamanis, and M. Proctor, "Real-time magnetic resonance imaging and electromagnetic articulography database for speech production research (TC)," *The Journal of the Acoustical Society of America*, vol. 136, no. 3, p. 1307, 2014.
- [2] S. G. Lingala, Y. Zhu, Y.-C. Kim, A. Toutios, S. S. Narayanan, and K. S. Nayak, "A fast and flexible mri system for the study of dynamic vocal tract shaping," *Magnetic Resonance in Medicine*, vol. 77, p. 112–125, 2017.
- [3] S. Asadiabadi and E. Erzin, "A deep learning approach for data driven vocal tract area function estimation," *2018 IEEE Spoken Language Technology Workshop (SLT)*, pp. 167–173, 2018.
- [4] A. Toutios, T. Sorensen, K. Somandepalli, R. Alexander, and S. S. Narayanan, "Articulatory synthesis based on real-time magnetic resonance imaging data," in *InterSpeech*, 2016.
- [5] B. L. McMicken, F. F. Salles, S. V. Berg, M. Vento-Wilson, K. L. Rogers, A. Toutios, and S. S. Narayanan, "Bilabial substitution patterns during consonant production in a case of congenital aglossia," *Journal of Communication Disorders, Deaf Studies & Hearing Aids*, vol. 2017, pp. 1–6, 2017.
- [6] M. Guzman, G. Miranda, C. Olavarria, S. Madrid, D. Munoz, M. Leiva, L. G. López, and C. Bortnem, "Computerized tomography measures during and after artificial lengthening of the vocal tract in subjects with voice disorders," *Journal of voice : official journal of the Voice Foundation*, vol. 31 1, pp. 124.e1–124.e10, 2017.
- [7] A. Serrurier, P. Badin, L. Lamalle, and C. Neuschaefer-Rube, "Characterization of inter-speaker articulatory variability: A two-level multi-speaker modelling approach based on mri data," *The Journal of the Acoustical Society of America*, vol. 145 4, p. 2149, 2019.
- [8] M. Proctor, R. Walker, C. Smith, T. Szalay, L. Goldstein, and S. Narayanan, "Articulatory characterization of english liquid-final rimes," *Journal of Phonetics*, vol. 77, 2019.
- [9] T. Sorensen, A. Toutios, L. Goldstein, and S. S. Narayanan, "Task-dependence of articulator synergies," *The Journal of the Acoustical Society of America*, vol. 145 3, p. 1504, 2019.
- [10] R. Alexander, T. Sorensen, A. Toutios, and S. S. Narayanan, "A modular architecture for articulatory synthesis from gestural specification," *The Journal of the Acoustical Society of America*, vol. 146 6, p. 4458, 2019.
- [11] E. Bresch, J. Adams, A. Pouzet, S. Lee, D. Byrd, and S. Narayanan, "Semi-automatic processing of real-time MR image sequences for speech production studies," in *7th International Seminar on Speech Production (ISSP7)*, 2006.
- [12] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, Jan 1988. [Online]. Available: <https://doi.org/10.1007/BF00133570>
- [13] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. Int. Joint Conf. Artificial Intelligence*, pp. 1674–679, 1981.
- [14] E. Bresch and S. Narayanan, "Region segmentation in the frequency domain applied to upper airway real-time magnetic resonance imaging," *IEEE Transaction on Medical Imaging*, vol. 28, no. 3, 2009.
- [15] M. I. Proctor, D. Bone, A. Katsamanis, and S. Narayanan, "Rapid semi-automatic segmentation of real-time magnetic resonance images for parametric vocal tract analysis," in *InterSpeech*, Makuhari, Japan, September 2010, pp. 1576–1579.
- [16] S. E. G. Ohman, "Numerical model of coarticulation," *Journal of the Acoustical Society of America*, vol. 41, no. 2, pp. 310–320, 1967.
- [17] J. Kim, N. Kumar, S. Lee, and S. Narayanan, "Enhanced airway-tissue boundary segmentation for real-time magnetic resonance imaging data," in *10th international seminar on Speech Production (ISSP10)*, Cologne, May 2014, pp. 222–225.
- [18] S. Silva and A. Teixeira, "Unsupervised segmentation of the vocal tract from real-time MRI sequences," *Computer Speech and Language*, vol. 33, no. 1, pp. 25–46, September 2015. [Online]. Available: <http://dx.doi.org/10.1016/j.csl.2014.12.003>
- [19] M. Labrunie, P. Badin, D. Voit, A. Joseph, J. Frahm, L. Lamalle, C. Vilain, and L. Boë, "Automatic segmentation of speech articulators from real-time midsagittal MRI based on supervised learning," *Speech Communication*, vol. 99, pp. 27–46, 2018.
- [20] S. Asadiabadi and E. Erzin, "Vocal tract airway tissue boundary tracking for rtMRI using shape and appearance priors," in *InterSpeech*, August 2017, pp. 636–640.
- [21] H. Takemoto, T. Goto, Y. Hagihara, S. Hamanaka, T. Kitamura, Y. Nota, and K. Maekawa, "Speech organ contour extraction using real-time MRI and machine learning method," in *Interspeech*, 2019, pp. 904–908. [Online]. Available: <http://dx.doi.org/10.21437/Interspeech.2019-1593>
- [22] D. E. King, "Dlib-ml: A machine learning toolkit," *J. Mach. Learn. Res.*, vol. 10, pp. 1755–1758, 2009.
- [23] K. Somandepalli, A. Toutios, and S. Narayanan, "Semantic edge detection for tracking vocal tract air-tissue boundaries in real-time magnetic resonance images," in *InterSpeech*, August 2017.
- [24] S. Asadiabadi and E. Erzin, "Automatic vocal tract landmark tracking in rtMRI using fully convolutional networks and kalman filter," in *Proc. of the 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.
- [25] C. A. Valliappan, R. Mannem, and P. K. Ghosh, "Air-tissue boundary segmentation in real-time magnetic resonance imaging video using semantic segmentation with fully convolutional networks," in *InterSpeech*, 2018, pp. 3132–3136. [Online]. Available: <http://dx.doi.org/10.21437/Interspeech.2018-1939>
- [26] C. A. Valliappan, A. Kumar, R. Mannem, G. R. Karthik, and P. K. Ghosh, "An improved air tissue boundary segmentation technique for real time magnetic resonance imaging video using Segnet," in *Proc. of the 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 5921–5925.
- [27] R. Mannem and P. K. Ghosh, "Air-tissue boundary segmentation in real time magnetic resonance imaging video using a convolutional encoder-decoder network," in *Proc. of the 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 5941–5945.
- [28] S. A. Hebbar, R. Sharma, K. Somandepalli, A. Toutios, and S. S. Narayanan, "Vocal tract articulatory contour detection in real-time magnetic resonance images using spatio-temporal context," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7354–7358, 2020.
- [29] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *International Conference on Learning Representations*, 2015.
- [30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2015.
- [31] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 681–685, 2001.
- [32] F. Chollet et al., "Keras," <https://keras.io>, 2015.
- [33] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, "Tensorflow: A system for large-scale machine learning," in *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation*, ser. OSDI'16. Berkeley, CA, USA: USENIX Association, 2016, pp. 265–283. [Online]. Available: <http://dl.acm.org/citation.cfm?id=3026877.3026899>
- [34] D. E. King, "Dlib-ml: A machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.
- [35] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.
- [36] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

Sasan Asadiabadi (S'18) is a Ph.D. candidate in Department of Electrical and Electronics Engineering at Koç University, Istanbul, Turkey. He received his B.Sc. in Electrical Engineering from Sharif University of Technology, Tehran, Iran, and M.Sc. in Electrical Engineering from Koç University Istanbul, Turkey, in 2014 and 2017 respectively. His current research interests include multimodal signal processing, medical image analysis, computer vision and deep learning for multimedia.

Engin Erzin (S'88-M'96-SM'06) received his Ph.D. degree, M.Sc. degree, and B.Sc. degree from the Bilkent University, Ankara, Turkey, in 1995, 1992 and 1990, respectively, all in Electrical Engineering. During 1995-1996, he was a postdoctoral fellow in Signal Compression Laboratory, University of California, Santa Barbara. He joined Lucent Technologies in September 1996, and he was with the Consumer Products for one year as a Member of Technical Staff of the Global Wireless Products Group. From 1997 to 2001, he was with the Speech

and Audio Technology Group of the Network Wireless Systems. Since January 2001, he has been with the Electrical & Electronics Engineering and Computer Engineering Departments of Koç University, Istanbul, Turkey. Engin Erzin is currently a member of the IEEE Speech and Language Processing Technical Committee and Associate Editor for the IEEE Transactions on Multimedia, having previously served as Associate Editor of the IEEE Transactions on Audio, Speech & Language Processing (2010-2014). His research interests include speech-audio-visual signal processing, affective computing, human-computer interaction, and machine learning.