Accelerating the Centerline Processing of Vocal Tract Shapes for Articulatory Synthesis

Romain Karpinski, Vinicius Ribeiro, Yves Laprie

ICA 2022
24th INTERNATIONAL CONGRESS ON ACOUSTICS
October 24(Mon) - 28(Fri), 2022 Gyeongju, Korea
Extended abstract

PROCEEDINGS of the
24th International Congress on Acoustics

October 24 to 28, 2022 in Gyeongju, Korea

Accelerating the Centerline Processing of Vocal Tract Shapes for Articulatory Synthesis

Romain Karpinski, Vinicius Ribeiro, Yves Laprie

Université de Lorraine, CNRS, Inria, LORIA, Nancy, F-54000, France

ABSTRACT

Acoustic simulations used in the articulatory synthesis of speech take a series of vocal tract shapes as an input. Acoustic simulations assume a plane wave propagation, simplifying and limiting the calculation time. It is, therefore, necessary to split 2D vocal tract shapes into small tubes perpendicular to the centerline simulating the plane wave propagation. The algorithm developed previously used a time-consuming regularization step whose computation time was close to that of acoustic simulations. Therefore, we explored the possibility of using deep learning to perform this step and accelerate the whole synthesis process. We used a database with a large number of rt-MRI images (150 000) and our regularizing algorithm for training. Two architectures were tested, one using a regression strategy applied to the two curves defining the vocal tract and one exploiting the classification of pixels in 2D images of the vocal tract. The first turned out to be much faster, even if it requires checking that the center line is correct and, in some sporadic cases using the initial algorithm as a fallback solution.

Keywords: Speech, Vocal tract shape, Centerline, Articulatory synthesis

1 INTRODUCTION

The process of articulatory synthesis comprises generating a series of 2D or 3D vocal tract shapes corresponding to the target utterance and synthesizing the audio signal using numerical aero-acoustical simulations [1]. An important intermediate step is generating the acoustic parameters used as input for the simulations. In order to limit the computation time of the simulations, we used an approach that assumes the propagation of a plane wave in the vocal tract. The intermediate step consists of splitting the vocal tract into small tubes, which requires the determination of the centerline assumed to represent the propagation of the wave inside the vocal tract.

The determination of the centerline has therefore received sustained attention, which led to several algorithms [2], [3]. In [4], we presented a heuristic algorithm that relies on dynamic programming to generate a first guess of the centerline and then a regularization step inspired by active curves [5] to obtain a smooth and relevant curve. The first step was optimized to reduce the space explored by dynamic programming, but the second step requires more computation. The work reported here is intended to accelerate the determination of the centerline by using neural networks, which are increasingly used to solve problems and surrogate optimization algorithms in many engineering domains ([6] for instance).

2 ACCELERATING THE CENTERLINE DETERMINATION

The time required to determine the centerline appears comparable to that of the acoustic simulation in order of magnitude. The time required for this step is about 0.250 s on a recent 2021 laptop (11th Gen Intel(R) Core(TM) i7-11850H @ 2.50GHz), and it seems reasonable to reduce the time required to the determination of the centerline to accelerate the overall synthesis. Moreover, even if the shapes of the vocal tract present variability, they always have invariant characteristics (e.g., the location of the extremities, the fixed or quasi-fixed walls of the pharynx, the presence of one or more constrictions), which means that a machine learning

approach can be considered. We thus have experimented with two types of neural networks:

- The classification approach which converts the initial problem into a semantic segmentation task and aims to find the points (pixels) in the vocal tract corresponding to the center line.
- The regression approach which uses the vocal tract contours directly as an input and generates the desired centerline.

In both cases, the training uses the centerline determined by the heuristic algorithm [4].

2.1 Dataset

The centerline algorithm is intended to be used on synthetic vocal tract shapes. In [7] we developed a deep learning approach for generating vocal tract shapes for a given series of phonemes (corresponding to a target sentence). The training of this approach exploited a database composed of rt-MRI (real-time Magnetic Resonance Imaging) sequences recorded by one male French native speaker [8] at Max Plank Institute, Göttingen, Germany. The recordings have a frame rate of 55 `fps`, pixel spacing of 1.412 `mm`, and an image resolution of 136×136 pixels for the 2D images of the vocal tract in the mid-sagittal plane. The corpus contains 38 acquisitions, with a median acquisition time of 81.8 seconds, a minimum of 36.3 seconds, and a maximum of 90.1 seconds. The sentences were selected to provide a phonetically balanced coverage of French, and the whole dataset comprises 161 570 images. We developed a deep learning automatic tracking of the tongue [9] which was extended to all the speech articulators [10]. By concatenating those contours, the two edge contours of the vocal tract C_{int} and C_{ext} (for inner and outer) are obtained and give the complete 2D geometrical shape of the vocal tract from the glottis to the lips (see Figure 1).

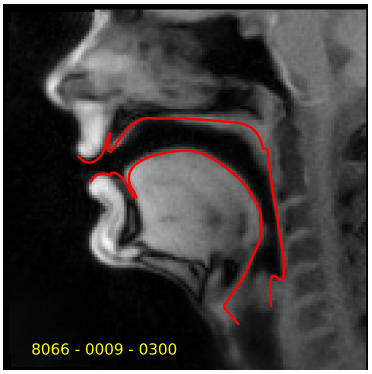

Figure 1. C_{int} and C_{ext} contours of the vocal tract.

Since the shapes used for articulatory synthesis derive from the contours of this dataset through training, we used them to train and test the acceleration of the centerline determination. Indeed, they cover a substantial variability of vocal tract shapes. The database is divided into three parts: 104 808 images for the training set, 19 307 images for the validation set, and 37 455 images for the test set. Some images do not correspond to speech since several images correspond to pauses, breaths, and swallowings.

2.2 Classification approach

This approach converts the centerline determination problem into a semantic segmentation task. The model aims to determine whether a pixel in a 2D image belongs to the centerline or the background. Once the segmentation is achieved, a second post-processing step transforms the network's output into an actual curve.

The network's inputs are binary images describing the two vocal tract walls (see Figure 1), while the target is a binary image describing the centerline calculated by the heuristics algorithm. To perform the semantic segmentation, we chose MobileNetV3 [11] as a backbone CNN to extract features and took advantage of pre-trained weights provided by the `PyTorch` [12] framework. However, the architecture of MobileNetV3 does

not enable semantic segmentation directly, and a decoder is thus needed to retrieve the original image size required to classify each pixel. We stripped the classification layer of MobileNetV3 and added upsampling blocks composed of an upsampling layer followed by a 2D convolution layer with `ReLU` activation. A total of three upsampling blocks are required to obtain the original image size. Finally, we added two convolutional layers, the last one performing classification. Figure 2 summarizes the neural network architecture used for this experiment.

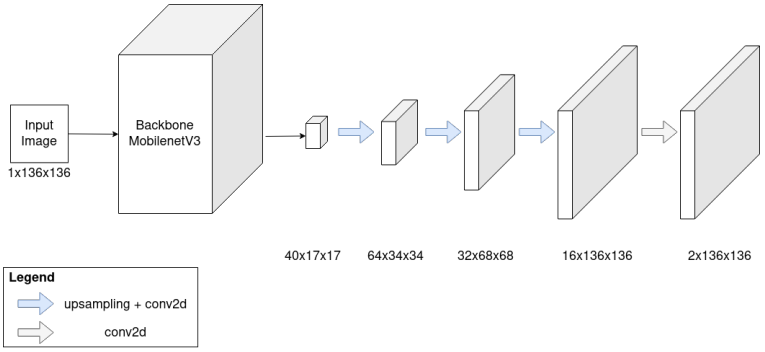

Figure 2. Architecture of the neural network used for the semantic segmentation.

The resulting probability map is then post-processed to find the centerline, which amounts to finding the shortest path between the extremities of the vocal tract (the middle points at the glottis and lips). Those two extremities are found from C_{int} and C_{ext} contours. The weight of each point equals $1 - p(j,i)$ where $p(j,i)$ is the probability that pixel j,i belongs to the centerline. Since the region corresponding to the vocal tract is given by the two edge contours, the graph can be drastically reduced, guaranteeing that the shortest path, i.e., the centerline, is inside the vocal tract.

2.3 Regression approach

The regression approach aims at estimating a function through a neural network. This function uses the contours of the vocal tract to extract the centerline. The goal is to obtain a function f' such that $f' \approx f$ with f being the existing function to replace. The method can be trained by minimizing the mean squared error (MSE) between f' and f.

The advantages of this solution are:

- Fast: the dimensionality being very low, the number of calculations is limited.
- Simple: the problem consists in approximating a function which is the expected result.
- Direct: there is no need for post-processing since the centerline is directly obtained.

The input corresponds to the two curves C_{int} and C_{ext} with their normalized coordinates in the interval $[0;1]$. These two curves are concatenated in a matrix $X \in \mathbb{R}^{4\times N}$ with N the number of points of both curves. It is important to note that the points of the two contours (inner and outer) are not synchronized, i.e., the *i-th* point of the inner contour is independent of the *i-th* point of the outer contour. The output curve corresponds to the centerline C_{center} in a form similar to inputs, i.e., a vector of N points.

Figure 3 shows the regression network's architecture. It uses convolutional layers that allow features to be extracted and fully connected layers that allow features to be related to each other. After each convolutional layer, and `Dense 1` and `Dense 2` layers, `ReLU` activation function is applied. This architecture enables the centerline to be obtained at a low computational cost.

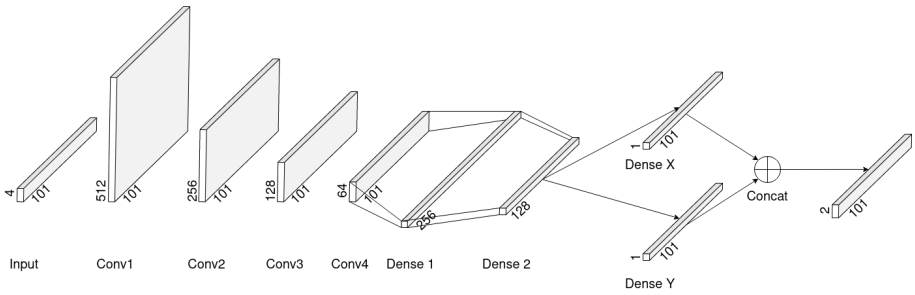

Figure 3. Architecture of the neural network used to perform regression.

2.3.1 Loss function

Let the predicted matrix $Y' \in \mathbb{R}^{2\times N}$. The is to minimize the MSE described by Equation 1 and Equation 2.

$$L(Y,Y') = \frac{1}{N} \times \sum_{i=0}^{N-1} L(Y_i, Y'_i) \tag{1}$$

$$L(Y_i, Y'_i) = \frac{1}{2} \times \sum_{j=0}^{1} (Y_{j,i} - Y'_{j,i})^2 \tag{2}$$

The objective described above does not guarantee the conformity of the predictions to the physical constraint, i.e., the predicted coordinates must lie between the vocal tract walls. An additional cost was introduced to penalize the predictions outside the vocal tract for overcoming this. If a point is inside the vocal tract, the sum of its distance with both contours should be equal to the distance of the two contours at the same position (illustrated by Figure 4) as written in Equation 3 and Equation 4.

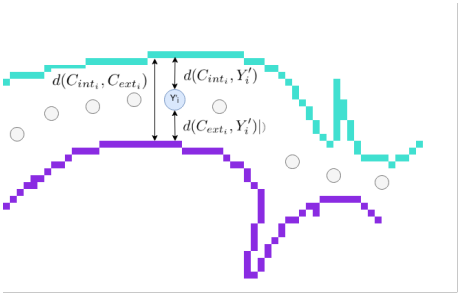

Figure 4. Illustration of the distance property between a centerline point and the nearest vocal tract contours points.

$$L_{out}(Y', C_{int}, C_{ext}) = \frac{1}{N} \times \sum_{i=0}^{N-1} L_{out}(Y'_i, C_{int_i}, C_{ext_i}) \tag{3}$$

$$L_{out}(Y'_i, C_{int_i}, C_{ext_i}) = |d(C_{int_i}, C_{ext_i}) - d(C_{int_i}, Y'_i) - d(C_{ext_i}, Y'_i)| \tag{4}$$

In addition, all centerline points are not equally important. For example, shifting a centerline point near a constriction gives rise to a more significant acoustic error than when the vocal tract is wider. Thus, the narrower the vocal tract, the more accurate the centerline at that point should be. The points are thus weighted according to the inverse of their distance with the vocal tract contours as shown by Equation 5 where ε was set to 10^{-4}.

$$Weight(C_{int_i}, C_{ext_i}) = \frac{1}{d(C_{int_i}, C_{ext_i}) + \varepsilon} \tag{5}$$

The complete updated loss is given by Equation 6.

$$Loss(C_{int}, C_{ext}, Y, Y') = \frac{1}{N} \times \sum_{i=0}^{N-1} Loss(C_{int_i}, C_{ext_i}, Y_i, Y_i') \tag{6}$$

$$Loss(C_{int_i}, C_{ext_i}, Y_i, Y_i') = (L(Y_i, Y_i') + L_{out}(Y_i', C_{int_i}, C_{ext_i})) \times Weight(C_{int_i}, C_{ext_i}) \tag{7}$$

2.3.2 Training parameters

The `PyTorch` library was used to create and train the neural network. Examples are also shifted during training by a small value to make the network more robust against translation. The Adam optimizer [13] was used with an initial learning rate of 10^{-3} which is decayed by a factor of 0.9 after five epochs without validation loss improvements. We used 500 epochs to train the network.

3 RESULTS

3.1 First results

Figure 5 illustrates some examples of the determination of the centerline by the Regression-B method and Table 1 shows assessment of both approaches. Regression-A experiment is the version of the regression without improving the loss function, and Regression-B with the improved loss function. A centerline is rejected as soon as it lies outside the vocal tract defined by the two contours C_{int} and C_{ext}. The Classification experiment has no rejection rate since it decodes the centerline within the vocal tract.

Table 1. Centerline distance and rejection rates for the three experiments.

	Regression-A	Regression-B	Classification
Centerline distance (in mm)	0.56	**0.50**	0.88
Rejection rate (in %)	36.30	40.75	0
Rejection rate for speech frames without extremities (in %)	10.25	8.86	0

It turns out that Regression-B gives the lowest distance between the reference and computed centerline but with a rejection rate of 23%. When looking at the results, it turns out that the high rejection rate is mainly due to the extremities slightly outside the vocal tract without changing the splitting of the vocal tract into tubelets. When the extremities are not considered, the rejection rate is only 8.86% for vocal tracts corresponding to speech and 23.07% for non-speech vocal tracts (silences, pauses, breaths, and swallowings). It should be noted that most rejections corresponding to speech are due to a small error. Concerning the Classification experiment, the lower accuracy is due to the low resolution of the rt-MRI images (136×136). The precision could be increased artificially by re-scaling images to a resolution of 1024×1024 but this would largely increase the time to perform inference and post-processing. We, therefore, decided to abandon this avenue.

4 Speed evaluation and improvements

Experiments were done using the following hardware with `PyTorch` framework:

- CPU: 11th Gen Intel(R) Core(TM) i7-11850H @ 2.50GHz
- GPU: NVIDIA T600

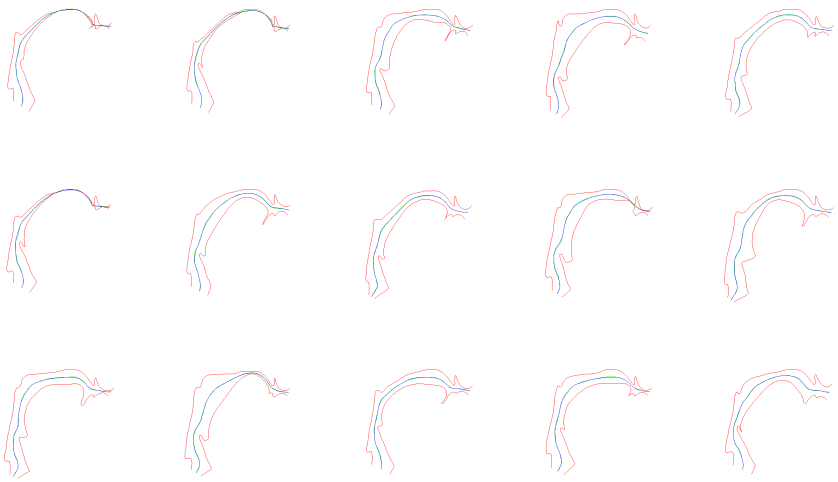

Figure 5. Results of the Regression-B on the test set. In red the vocal tract shape contours. In green the reference centerline and in blue the predicted centerline.

The Regression method is a lot faster (0.1044 ms with a batch size of 32, and the time used by the rejection test is negligible) than the Classification (38.57 ms on average, including post-processing). However, the time required by the Regression method has to consider the rejection cases. Indeed, the initial heuristic algorithm is used as a fallback solution in case of rejection. The average time required by the Regression method is thus 12.94 ms, approximately 20 times faster than the original solution.

Some of the hyperparameters used in the neural network architecture were set arbitrarily. This means that we may not have the best combination of feature maps and/or the number of layers. We thus searched for the best hyperparameters by providing the range of values for the convolution and dense networks in the Tree-Structured Parzen Estimator (TPE) [14] implemented in the Optuna framework [15].

We found that the best results on the validation test set were obtained with a network using 200 points for the vocal tract contours and 100 points for the centerline. Also, the optimal convolution sub-network uses 512 initial feature maps and a depth of 5. The optimal dense sub-network uses 256 neurons and a depth of 3. These values are not far from the original network since the new architecture only adds one depth to each sub-network.

With this network configuration, the rejection rate decreases to 6.18% for vocal tracts corresponding to speech with an average centerline error of 0.56 mm.

5 CONCLUSIONS

This work thus enables the determination of the centerline of the vocal tract to be drastically accelerated, which was the objective. Indeed, the results presented above show that the accelerated version is 20 times faster than the original algorithm while keeping the same level of precision. However, the training database relies on a heuristic algorithm using "common sense" acoustic criteria to express a cost function. There is no proof that this heuristic corresponds to the acoustical ground truth. The strength of this approach presented in this paper is that it can be applied to other centerline determination algorithms or even real data, provided that wave propagation in the vocal tract can be observed easily.

ACKNOWLEDGEMENTS

Authors acknowledge the CNRS for funding the engineer involved in this project and the ANR for funding the Full3DTalkingHead project in which this work takes place.

REFERENCES

[1] S. Stone, Y. Gao, and P. Birkholz, "Articulatory synthesis of vocalized /r/ allophones in german," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 879–889, 2022. DOI: 10.1109/TASLP.2021.3130969.

[2] A. Poznyakovskiy, A. Mainka, I. Platzek, and D. Mürbe, "Fast semiautomatic algorithm for centerline-based vocal tract segmentation," *Biomed Res Int.*, vol. Epub 2015 Oct 18. 2015. DOI: 10.1155/2015/906356.

[3] Z. I. Skordilis, A. Toutios, J. Töger, and S. Narayanan, "Estimation of vocal tract area function from volumetric magnetic resonance imaging," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 924–928. DOI: 10.1109/ICASSP.2017.7952291.

[4] Y. Laprie, M. Loosvelt, S. Maeda, E. Sock, and F. Hirsch, "Articulatory copy synthesis from cine x-ray films," in *Interspeech 2013 (14th Annual Conference of the International Speech Communication Association)*, Lyon, France, Aug. 2013.

[5] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1987.

[6] K. Singh and R. K. Kapania, "Accelerated optimization of curvilinearly stiffened panels using deep learning," *Thin-Walled Structures*, vol. 161, p. 107 418, 2021, ISSN: 0263-8231. DOI: https://doi.org/10.1016/j.tws.2020.107418. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0263823120312817.

[7] V. Ribeiro, K. Isaieva, J. Leclere, P.-A. Vuissoz, and y. Laprie, "Automatic generation of the complete vocal tract shape from the sequence of phonemes to be articulated," *Speech Communication*, vol. 141, pp. 1–13, Apr. 2022. DOI: 10.1016/j.specom.2022.04.004. [Online]. Available: https://hal.univ-lorraine.fr/hal-03650212.

[8] I. Douros, J. Felblinger, J. Frahm, K. Isaieva, A. Joseph, Y. Laprie, F. Odille, A. Tsukanova, D. Voit, and P. Vuissoz, "A multimodal real-time mri articulatory corpus of french for speech research," in *INTERSPEECH 2019-20th Annual Conference of the International Speech Communication Association*, 2019.

[9] K. Isaieva, Y. Laprie, N. Turpault, A. Houssard, J. Felblinger, and P. Vuissoz, "Automatic tongue delineation from mri images with a convolutional neural network approach," *Applied Artificial Intelligence*, vol. 34, no. 14, pp. 1115–1123, 2020.

[10] V. Ribeiro, K. Isaieva, J. Leclere, P. Vuissoz, and Y. Laprie, "Towards the Prediction of the Vocal Tract Shape from the Sequence of Phonemes to be Articulated," in *Proc. Interspeech 2021*, 2021, pp. 3325–3329. DOI: 10.21437/Interspeech.2021-184.

[11] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Q. V. Le, and H. Adam, *Searching for mobilenetv3*, 2019. DOI: 10.48550/ARXIV.1905.02244. [Online]. Available: https://arxiv.org/abs/1905.02244.

[12] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., Curran Associates, Inc., 2019, pp. 8024–8035. [Online]. Available: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[13] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[14] J. Bergstra, D. Yamins, and D. Cox, "Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures," in *International conference on machine learning*, PMLR, 2013, pp. 115–123.

[15] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," in *Proceedings of the 25rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.