Real-time and interactive tools for vocal training based on an analytic signal with a cosine series envelope

Hideki Kawahara*, Ken-Ichi Sakakibara†, Eri Haneishi‡ and Kaori Hagiwara‡

* Wakayama University, Wakayama, Japan

E-mail: kawahara@wakayama-u.ac.jp

† Health Science University of Hokkaido, Sapporo, Japan

E-mail: quesokis@gmail.com

‡ Showa University of Music, Kawasaki, Japan

E-mail: haneishi@tosei-showa-music.ac.jp, k-hagiwara@tosei-showa-music.ac.jp

Abstract—We introduce real-time and interactive tools for assisting vocal training. In this presentation, we demonstrate mainly a tool based on real-time visualizer of fundamental frequency candidates to provide information-rich feedback to learners. The visualizer uses an efficient algorithm using analytic signals for deriving phase-based attributes. We start using these tools in vocal training for assisting learners to acquire the awareness of appropriate vocalization. The first author made the MATLAB implementation of the tools open-source. The code and associated video materials are accessible in the first author's GitHub repository.

I. INTRODUCTION

Adequate vocalization, such as speaking and singing, plays an essential role in maintaining a good quality of daily human life. For old adults, adequate vocalization helps their quality of life by promoting social communication activities and, in some cases, even reducing the risk of aspiration pneumonia. For vocal professionals, such as singers, teachers, and music therapists, proper vocalization technique is crucial for preserving their professional lives[1]. To maintain proper vocalization skills, we introduce tools for assisting vocal training. Real-time tools provide visual feedback of voicing attributes. The supporting tools enable detailed analysis and study the effects of articulatory changes. The tools use an extractor of f_o[1] (fundamental frequency) candidates in real-time and audio sampling rate using analytic signals with a six-term cosine series[3]. It also uses other interactive tools for visualization of auditory information and speech production processes[4]. The body part of this article focuses on the introduction and use of the tools. We placed technical details in the appendix for making the body text less confusing.

II. REAL-TIME AND INTERACTIVE TOOLS

This section mainly focuses on the tool based on a real-time extractor of f_o candidates. In a training session, the instructor uses this tool interactively to demonstrate learners how to control voicing by visualizing their voices in real-time and playback the reference samples and the learners' voices.
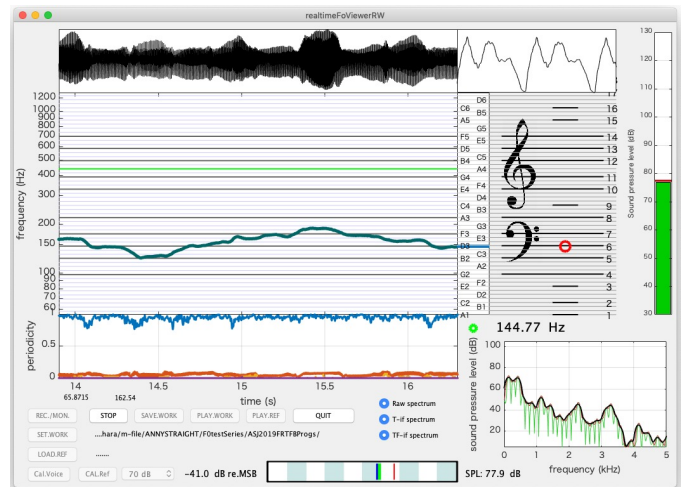

Fig. 1. GUI of the tool with a real-time f_o candidates extractor.

The tool has the following functions; a)f_o candidates' trajectory visualization with the salience of periodicity, b)musical note display, c)waveform display (long and short duration), d)sound pressure level, e)spectrum, f)recording with calibration, and g)playback of working and reference recordings.

Figure 1 shows a snapshot of the GUI of the tool.[2] The left three panels are continuously scrolling to the left to show trajectories of the waveform, f_o candidates, and the periodicity salience (from top to bottom). The top right panel shows the stabilized waveform of several cycles of the fundamental period. The middle right panel shows staves with a treble clef and a bass clef. We introduced this panel to help learners to be aware of voice pitch because sometimes learner has difficulty in finding what "high" or "low" pitch is. The red circle represents the corresponding f_o position. Each stave has a number at the right side. The instructor can use the red circle and the number to advise learners. Just below the musical

---

[1]Using "f_o" instead of "F0" for representing fundamental frequency is based on a recommendation[2]

[2]The first author's GitHub repository[5] has the latest version of the accompanying video, which introduces how to use this tool.

panel, a green circle and a number representing the frequency of the best f_o candidate appear when the signal periodicity is salient. The small panel between the scrolling display and the musical display shows corresponding note names. Thin horizontal lines span these three panels. They represent the chromatic scale.

The rightmost bar graph shows the calibrated sound pressure level at the reference position (30 cm in front of the lip center[6]) using C-weighting[7]. The green bar shows the fast response, and the horizontal red line shows the slow response.

The bottom right panel shows spectral information using three lines. The three lines are a)power spectrum (a green line), b)temporally interference-free spectrum (a red line), and c)spectrum without temporal and frequency interferences (a bold black line)[8]. The vertical axis represents the sound pressure level when calibrated.

The bottom center bar panel is a level indicator in dB scale spanning from -100 dB to 0 dB, where 0 dB corresponds to the MSB (Most Significant Bit, the maximum input level). This indicator has three cursors. The red cursor represents the peak value, and the blue cursor represents the RMS value. Finally, the green cursor represents the temporally smoothed RMS value. The user can calibrate this tool by using any stable test signals (including the learner's sustained vowel) and a sound level meter together with this level indicator. Please refer[9], [6] for the recommended procedure and microphone settings.

This tool has ten push buttons and a popup menu. They have the following functions. "REC. START" button restarts the tool. The "SAVE.WORK" button writes the contents of the input buffer to a file providing a unique name. The saved file is a WAVE format with 44,100 Hz sampling frequency and 24-bit resolution. The "STOP" button stops the continuous monitoring of the input. "PLAY.WORK" button outputs the signal in the audio input buffer. "PLAY.REF" button outputs the signal in the reference buffer. The "QUIT" button terminates the tool. "SET.WORK" button assigns the directory for writing work files. "LOAD.REF" button reads an audio file as the reference. "Cal.Voice" and "CAL.Ref" buttons calibrate the microphone input level. Only adequate buttons are active and highlighted.

The popup menu showing "70 dB" selects the calibration sound pressure level. Our GitHub repository[5] has links to videos showing how to calibrate this tool and several operation examples.

A. Supporting interactive tools

In addition to this tool, we developed two supporting tools and applied previously developed tools[4]. Developed tools are a real-time phase-attributes visualizer and a detailed interactive inspector.

1) Real-time phase-attributes visualizer: Figure 2 shows example visualizations of three phase-related attributes. From top to bottom, the panels show the phase map, the normalized instantaneous frequency map, and the normalized group delay map.

Descriptions of the components in each panel are as follows. This description is for the top panel. The top left panel
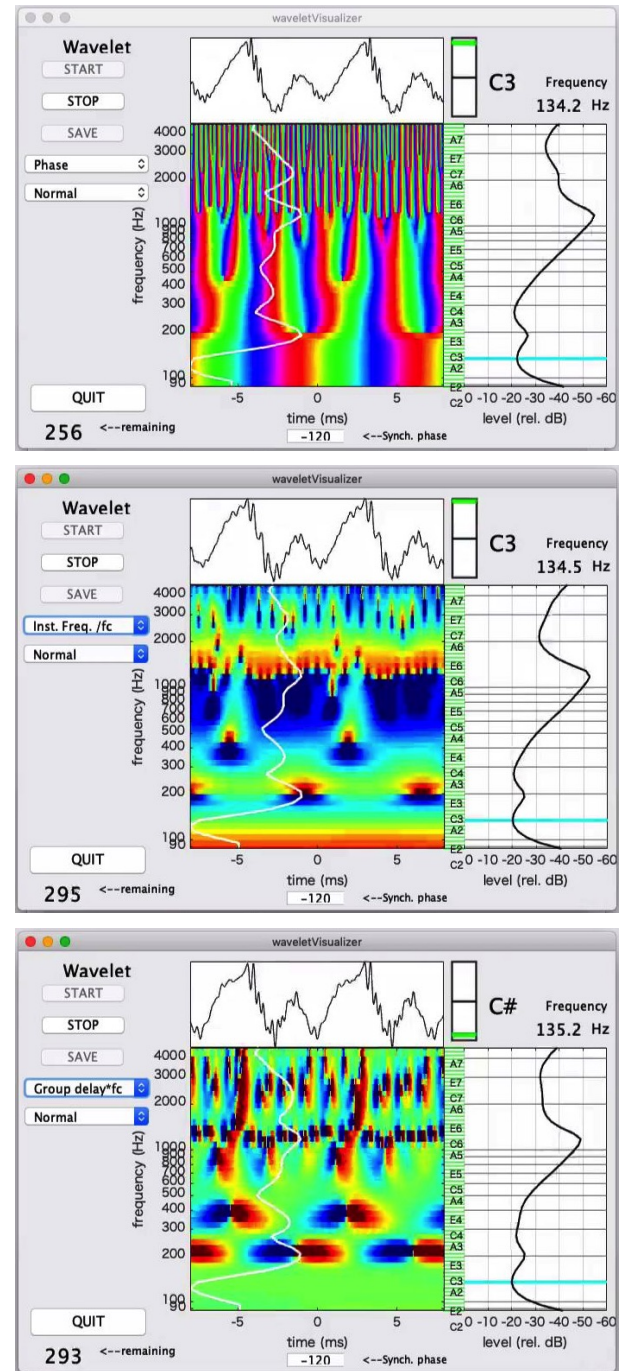

Fig. 2. Snapshots of a real-time phase-attributes visualizer.

shows the pseudo color image of the phase of the analysis results in the middle. The line plot above the image shows the time-aligned input waveform based on the phase of the fundamental component. The cyan line in the right plot indicates the fundamental component. The rightmost line plot shows the RMS (root mean squared) average of each output. The top right corner displays the frequency of the best f_o candidate and the closest musical note name with a tuning display (The black center line shows the target note for the
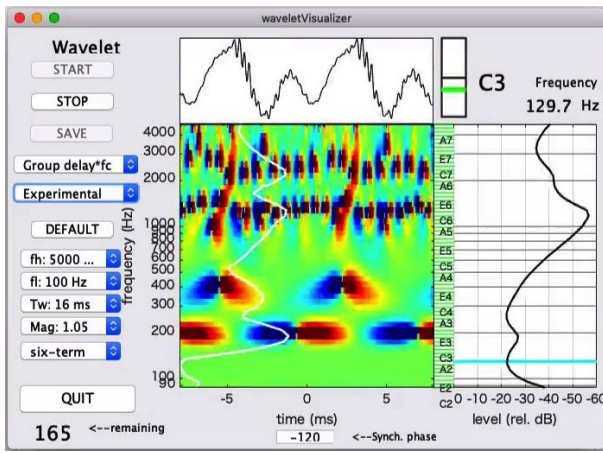
Fig. 3. Snapshot of the "Experimental" mode of a real-time phase-attributes visualizer.

Fig. 4. GUI of an interactive detailed inspector.

Fig. 5. GUI of a voice production simulator in the previous tools[4].

green line pitch monitor). The main image of this panel shows the pseudocolor representation of the phase value itself. We use the colormap which yields cyclic impression because phase has a circular topology.

The second panel shows the relative output instantaneous frequency normalized by the center frequency. The relative instantaneous frequency display has a rainbow-colored horizontal bar which corresponds to the fundamental component.

The bottom panel shows the relative output group delay normalized by the nominal period of each filter response. The relative group delay displays have (gently bending) rainbow-colored vertical bars which correspond to the GCIs (Glottal Closure Instances).

Figure 3 shows the "Experimental" mode, which allows an extended parameter setting.

2) Detailed interactive inspector: Figure 4 shows a snapshot of the detailed inspector. The analyzed file is the contents of the real-time tool of the session shown in the accompanying video. The top panel shows the waveform
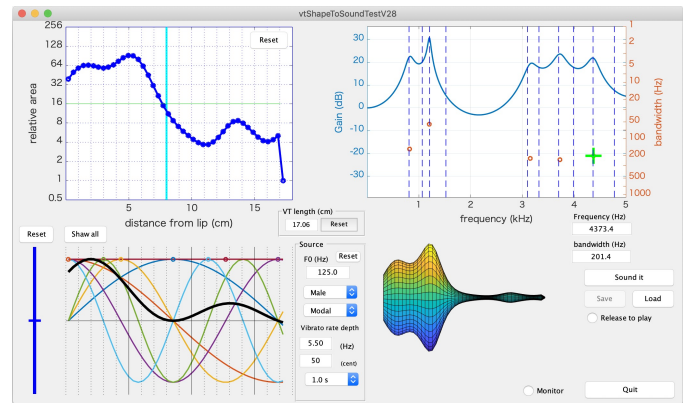
for about 4.5 seconds. The middle panel shows the f_o candidates trajectories. The thickness and the darkness of each trajectory represent salience of the periodicity. The bottom panel represents periodicity salience. It uses the estimated SNR for the salience. The user can interactively check details by "zoom in," "zoom out," and "panning" in each panel.

3) SparkNG: previous tools: Figure 5 shows the snapshot of the GUI of a voice production simulator in our previous tools[4]. The tool provides interactive manipulation of voicing attributes and provide results by synthesizing voiced sounds. The attributes are the vocal tract area function, the first five formant frequencies and bandwidths, vocal tract length, and the shape parameters of the glottal source waveform. The results of parameter modifications directly change the synthesized sounds and enabling users to acquire intricate relations between various voicing attributes.

III. CONCLUSIONS

We introduced real-time and interactive tools for assisting vocal training. The tools are implemented using MATLAB and open-sourced. The introduced tools and the other assistive tools, SparkNG are available in the first author's GitHub repository[5].

IV. ACKNOWLEDGEMENTS

REFERENCES

[1] E. Haneishi, "Effects of a music therapy voice protocol on speech intelligibility, vocal acoustic measures, and mood of individuals with parkinson's disease," *Journal of music therapy*, vol. 38, no. 4, pp. 273–290, 2001.

[2] I. R. Titze, R. J. Baken, K. W. Bozeman, S. Granqvist, N. Henrich, C. T. Herbst, D. M. Howard, E. J. Hunter, D. Kaelin, R. D. Kent, J. Kreiman, M. Kob, A. Löfqvist, S. McCoy, D. G. Miller, H. Noé, R. C. Scherer, J. R. Smith, B. H. Story, J. G. Švec, S. Ternström, and J. Wolfe, "Toward a consensus on symbolic notation of harmonics, resonances, and formants in vocalization," *The Journal of the Acoustical Society of America*, vol. 137, no. 5, pp. 3005–3007, 2015.

[3] H. Kawahara, K.-I. Sakakibara, M. Morise, H. Banno, T. Toda, and T. Irino, "A new cosine series antialiasing function and its application to aliasing-free glottal source models for speech and singing synthesis," in *Proc. Interspeech 2017*, Stocholm, August 2017, pp. 1358–1362.

[4] H. Kawahara, "SparkNG: Interactive MATLAB tools for introduction to speech production, perception and processing fundamentals and application of the aliasing-free L-F model component," in *Interspeech 2016*, 2016, pp. 1180–1181.

[5] ——, "GitHub projects of Hideki Kawahara," GitHub, (Last access: 2019-04-26). [Online]. Available: https://github.com/HidekiKawahara

[6] R. R. Patel, S. N. Awan, J. Barkmeier-Kraemer, M. Courey, D. Deliyski, T. Eadie, D. Paul, J. G. Švec, and R. Hillman, "Recommended protocols for instrumental assessment of voice: American speech-language-hearing association expert panel to develop a protocol for instrumental assessment of vocal function," *American Journal of Speech-Language Pathology*, vol. 27, no. 3, pp. 887–905, 2018.

[7] International Electrotechnical Commission, "Electroacoustics: sound level meters–Part 1: Specification (IEC 61672-1:2013)," Geneva, Switzerland, 2013, tC 29.

[8] H. Kawahara, M. M., and K. Hua, "Revisiting spectral envelope recovery from speech sounds generated by periodic excitation," in *APSIPA ASC 2018, Hawaii*, 2018, pp. 1674–1683.

[9] J. G. Švec and S. Granqvist, "Guidelines for selecting microphones for human voice production research," *American Journal of Speech-Language Pathology*, vol. 19, no. 4, pp. 356–368, 2010.

[10] J. L. Flanagan and R. M. Golden, "Phase Vocoder," *Bell System Technical Journal*, vol. 45, no. 9, pp. 1493–1509, Nov 1966.

[11] Intel, "Vector Mathematics (VM): Performance and Accuracy Data," 2018, (Access date: 2018-10-12). [Online]. Available: https://software.intel.com/sites/products/documentation/

[12] F. J. Harris, "On the use of windows for harmonic analysis with the discrete Fourier transform," *Proceedings of the IEEE*, vol. 66, no. 1, pp. 51–83, 1978.

[13] J. Kaiser and R. W. Schafer, "On the use of the I_0-sinh window for spectrum analysis," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 28, no. 1, pp. 105–107, 1980.

[14] A. H. Nuttall, "Some windows with very good sidelobe behavior," *IEEE Trans. Audio Speech and Signal Processing*, vol. 29, no. 1, pp. 84–91, 1981.

APPENDIX

A. Real-time f_o candidates extractor

This appendix briefly describes the procedures used to implement the proposed tools.

1) Simplified implementation of phase-related attributes: Flanagan introduced an equation to calculate the instantaneous frequency, which does not rely on phase unwrapping[10]. Recent advances in multimedia processing made CPU have efficient instructions for calculating related functions (for example[11]).

These modern CPUs calculates the following discrete implementation of instantaneous frequency $\omega_i[n]$ and group delay $\tau_g[k]$ efficiently.

$$\omega_i[n] = \angle \left[\frac{x[n+1]}{x[n]} f_s \right], \qquad (1)$$

$$\tau_g[k] = -\frac{1}{\Delta\omega} \angle \left[\frac{X[k+1]}{X[k]} \right], \qquad (2)$$

where $x[n]$ represents an analytic discrete-time signal, and k represents the discrete frequency.

2) Analytic signal with the six-term cosine series envelope: Filtering an input signal using a filter having an analytic signal impulse response yields an analytic signal output. Multiplying a time windowing function and a carrier complex exponential function makes such analytic signal. We found that commonly used time windowing functions[12], [13], [14] are not relevant for calculating phase-related attributes[3]. The
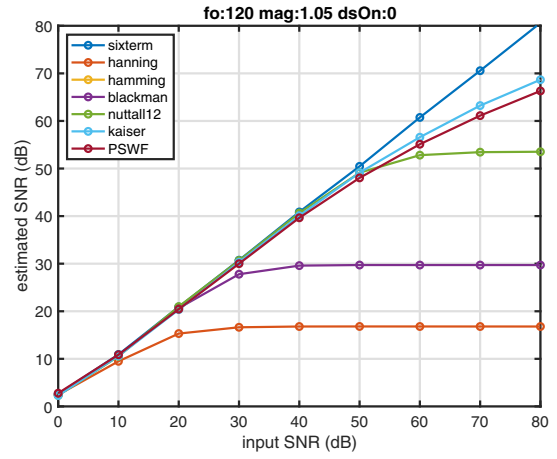

Fig. 6. Estimated SNR for each envelope functions. The stretching factor is $c_{\mathrm{mag}} = 1.05$.

following function $w_e(t; f_c, c_{\mathrm{mag}})$ provides practically the best results.

$$w_e(t; f_c, c_{\mathrm{mag}}) = \sum_{k=0}^{K} a_k \cos\left(\frac{2\pi k f_c t}{K c_{\mathrm{mag}}} \right), \qquad (3)$$

where c_{mag} represents a tuning parameter which determines the relation between the bandwidth and the carrier frequency f_c. We used the following cosine series with $K = 5$. The optimized coefficients are the following.

$$\{a_k\}_{k=0}^{5} = \{0.2624710164, 0.4265335164, 0.2250165621,$$
$$0.0726831633, 0.0125124215, 0.0007833203\} \qquad (4)$$

The following equation provides the desired impulse response.

$$w(t) = w_e(t; f_c, c_{\mathrm{mag}}) \exp(j2\pi f_c t) \quad \text{where} \quad j = \sqrt{-1}, \quad (5)$$

3) SNR estimation: When the dominant component is a stable sinusoid located in the pass-band of a filter, the instantaneous frequency of the filter output is temporally constant. The group delay calculated using adjacent channel outputs is also temporally constant. Mixing temporal variations of the instantaneous frequency and the group delay provides the estimate of the SNR of the filter output.

Figure 6 shows the relation between the given SNR and the estimated SNR using commonly used windowing functions and the proposed six-term cosine series. The proposed envelope yields accurate estimates from 10 dB to 80 dB SNR. This SNR estimation executes about 300 times faster than real-time for a single filter output for 44,100 Hz sampled signals. This speed test used MATLAB implementation on MacBook Pro (2019 13" 2.7 GHz Intel Corei7, with 16 GB memory).

4) Extractor of f_o candidates: We placed six SNR estimators for each octave covering from 80 Hz to 5 kHz. The initial candidates are detected as fixed points from the filter center frequency to the output instantaneous frequency. The extractor selects the initial four candidates based on the SNR. These are the candidates displayed in the proposed tools.