VOICESAUCE: A PROGRAM FOR VOICE ANALYSIS

Yen-Liang Shue[a], *Patricia Keating*[b], *Chad Vicenik*[b] *& Kristine Yu*[b]

[a]Dept. of Electrical Engineering; [b]Dept. of Linguistics, UCLA, Los Angeles, CA, USA
yshue@ee.ucla.edu; keating@humnet.ucla.edu;
cvicenik@humnet.ucla.edu; krisyu@humnet.ucla.edu

ABSTRACT

VoiceSauce is a new application, implemented in Matlab, which provides automated voice measurements over audio recordings. VoiceSauce computes many voice measures, including those using corrections for formant frequencies and bandwidths. It outputs values as text or for Emu database, and incorporates output from a separate program for automatic analysis of electroglottographic signals. VoiceSauce is available online for free download.

Keywords: voice, phonation, acoustic analysis, software

1. INTRODUCTION

The study of voice quality is part of several subfields of speech research, including linguistic phonetics, prosody, and sociophonetics. VoiceSauce (or VS) is an easy-to-use tool for researchers interested in multiple voice measures over running speech. It is implemented, and can run, in Matlab, but it is also available as a freestanding program for PCs.

VS runs on directories of .wav files, automatically producing measurements for every audio file. If Praat [1] textgrids are available for the files, analysis for many measures can be limited to labeled intervals on any tier, which greatly speeds up computations.

Figure 1: VoiceSauce home screen.

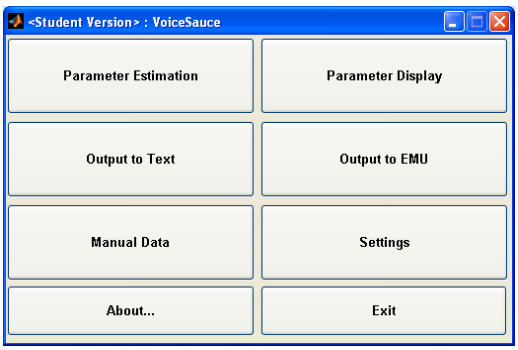


Figure 1 shows VS's streamlined home screen. The first step in a VS analysis is Parameter Estimation, the calculations of the various measures. In the Parameter Estimation window, the user can select which parameters to calculate; in the Settings window, default settings for these calculations can be changed.

2. ALGORITHMS AND PARAMETERS

The following sections describe the parameters that can be estimated, and the algorithms used to do so.

2.1. F0 and harmonic spectra magnitudes

One of the critical measurements made by VS is the fundamental frequency, F0. VS uses this measurement to estimate the location of harmonics. VS can make measurements of F0 using any of three different programs: STRAIGHT [9], the Snack Sound Toolkit [13], or Praat [1], which offers a choice of autocorrelation and cross-correlation algorithms. By default, the STRAIGHT algorithm is used to find F0 at 1 ms intervals. All the F0 detection algorithms rely on user specifications of the Max F0 and Min F0 to constrain their estimations.

Harmonic spectra magnitudes are computed pitch-synchronously, by default over a 3-cycle window. (Larger windows are recommended in some cases.) This method eliminates much of the variability in spectra computed over a fixed time window. The harrmonic magnitudes are found by using a maximum search algorithm around the spectral locations as estimated by the F0. The total search range is set to 10% of the estimated F0 value. This is equivalent to using a very long FFT window and enables a much more accurate measure without relying on large FFT calculations.

2.2. Formants and corrections

The Snack Sound Toolkit is used by default to find the frequencies and bandwidths of the first four formants, using as defaults the covariance method, pre-emphasis of .96, window length of 25 ms, and frame shift of 1 ms (to match STRAIGHT). Praat's Burg algorithm can also be used to estimate the

formants (always over whole files, never over labeled intervals in a textgrid). When Praat is used for formant (or F0) estimation, the user can set any of its parameters.

In previous work, Hanson [4] and Iseli and colleagues (e.g. [8]) developed an algorithm that estimates the voice source parameters H1*-H2* and H1*-A3*, where the asterisk is used to denote that the corresponding spectral magnitudes (H1, H2, A3) are corrected for the effect of formants (frequencies and bandwidths). In VS, the harmonic amplitudes for all measures of spectral magnitude can be corrected for every frame using the measured formant frequencies, plus bandwidths estimated by formula from those frequencies [6]. (In VS outputs, corrected measures are indicated by "c", uncorrected measures by "u".) For H1*-H2*, only F1 and F2 are used in the correction; for H1*-A3*, F1 through F3 are used. Finally, these measures are smoothed with a moving average filter with a default length of 20 samples.

2.3. SHR

The Subharmonic-to-Harmonic Ratio (SHR) is a measure proposed by Sun [15] that quantifies the amplitude ratio between subharmonics and harmonics. It is implemented using Sun's algorithm and code [14]. SHR may be especially relevant for characterizing speech with alternating pulse cycles [3] and is derived from the summed subharmonic and harmonic amplitudes calculated in the log domain using spectrum shifting.

2.4. Energy

Root Mean Square (RMS) energy is calculated at every frame over a variable window equal to five pitch periods. The variable window effectively normalizes the energy measure with F0 to reduce the correlation between them.

2.5. Cepstral measures

2.5.1. CPP

Cepstral Peak Prominence (CPP) calculations are based on the algorithm described by Hillenbrand, et al. [7]. A variable window length equal to five pitch periods is used for the calculations. After multiplying the data with a Hamming window, the data is then transformed into the real cepstral domain. The CPP is found by performing a maximum search around the frequency of the pitch period. This peak is normalized to a linear regression line which is calculated between 1 ms and the maximum frequency.

2.5.2. HNR

Harmonic-to-Noise Ratio (HNR) measures are derived by de Krom's algorithm [10]. Using a variable window length equal to five pitch periods, the HNR measurements are found by liftering the pitch component of the cepstrum and comparing the energy of the harmonics with the noise floor. HNR05 measures the HNR for 0-500Hz, HNR15 measures the HNR for 0-1500Hz and HNR25 measures the HNR for 0-2500Hz. Note that, in contrast, CPP covers the entire frequency range.

2.6. Summary of measures

The full set of measures that can be computed is:

- F0 from STRAIGHT
- F0 from Snack
- F0 from Praat
- F1-F4 and B1-B4 from Snack
- F1-F4 and B1-B4 from Praat
- H1, H2, H4
- A1, A2, A3
- Cepstral Peak Prominence
- Harmonic-to-Noise Ratios (3 frequency bands)
- Subharmonic-to-Harmonic Ratio
- Energy
- H1-H2(*)
- H1-A1(*)
- H1-A2(*)
- H1-A3(*)
- H2-H4(*)

All harmonic measures come both corrected (*) and uncorrected.

2.7. Manual correction of measures

The reliability of these measures depends on the successful estimation of their component parameters. If the F0 is not well-tracked, then all the measures that include H1 will be problematic. Similarly, if one or more formants are not well-tracked, then the corresponding measures will be problematic. Thus if the estimate of F1 is wrong, then A1 and H1-A1 will be wrong too, even for the uncorrected measures. Obviously, all the amplitude corrections also crucially depend on accurate formant estimation. Errors in F1 estimation are especially likely for breathy, nasal, or high-pitched vowels. Therefore it is recommended that the F0

and formant estimates be checked to verify the integrity of the voice measures derived from them.

VS allows some manual overriding of problematic measures. This function is particularly useful for phonations which contain pitch doubling or creakiness as these effects will often result in inaccuracies with the F0 estimator. It is common in these cases to hand-correct the measures in a new data file which is loaded into VS; the new values can then be used to recalculate other measures.

3. OUTPUTS

The initial output from VS's Parameter Estimation is a set of binary Matlab MAT-files, one per input file. The Parameter Display window allows the user to display (multiple) parameters with the waveform of a single audio file. This is not intended as a measurement facility, but rather for quick visual checks of sample outputs to verify that the estimations were successful. For most users, it is useful to perform a further output step in VS to extract results from the Matlab files.

3.1. Output to text

From Praat textgrids, VS identifies all labeled intervals on a tier (called segments) and writes out the results for them; if there are no textgrids, then results are given over the entire file. The user requests either:

- all values (at the frame shift rate, which by default is 1 ms), with each value of a given measure on a separate row, or
- averages over N sub-segments (where N is a number specified by the user) within a labeled segment, with each segment on a separate row and each sub-segment a new column.

In either case, each measured parameter is one or more columns. The first option creates text files that are very long; the second option creates text files that can be very wide.

The user specifies which parameters to output, and whether they should be written to one large text file with all parameters, or separate smaller text files with subsets of parameters.

3.2. Output to Emu

VS can output its data in SSFF format for use in Emu speech databases [5]. Users must direct VS to the measured data that are to be converted into Emu-readable format and select which measured parameters to write, but there are no options controlling textgrid labels or dividing data into sub-segments. The output is in the form of one track file per parameter per audio file. These track files can be viewed, queried, or further analyzed in Emu, or in R using the Emu library.

3.3. Including EGG measurements

VS can include in its output file the outputs from analysis of corresponding EGG signals, if the outputs are in the appropriate format and at the appropriate frame rate. Henry Tehrani's EGG analysis program EggWorks (available along with VS) produces text files with such outputs at 1 ms intervals. VS adds these measures to its output file.

4. COMPARISON WITH OTHER METHODS

We have compared results from VS for the most common measure in the literature, H1-H2, to those from two common sources of harmonic magnitude measures: "by-hand" from FFT spectra, and automated from Praat. For this comparison, the beginnings of tokens of the low vowel [a] after voiced, voiceless aspirated, and ejective stops from five speakers of Georgian [16] was analyzed by the three methods. In this language, the different stops audibly affect the voice quality at vowel onset.

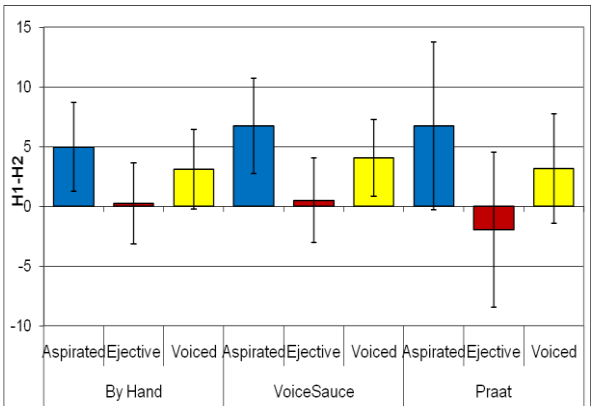
The "by-hand" measurements were made in Scicon R&D's PCQuirer, from FFT spectra made with a 21 Hz bandwidth and a 40 ms window, positioned immediately after vowel onset, so covering about the first third of the vowels. The amplitudes of the first and second harmonics were manually marked and logged using a cursor, as in several previous studies of voice quality in the literature, e.g. [2] and [11]. This method is relatively unreliable and is not taken as a benchmark, but rather simply as one standard practice. Many of the files could not be analyzed by this method because the FFT did not show a clear harmonic structure.

The Praat measurements were made using a new script based on one by Remijsen [12]. H1-H2, H1-A1, H1-A2, and H1-A3 can be measured for each labeled interval on a tier. These measures are neither pitch-synchronous nor corrected for formants. With this script, a file can not be analyzed (is discarded by the script, with no measurements) if Praat cannot detect an F0 and all three formants. Here, only the H1-H2 measure was needed, so the script was modified to compute only that, over the first third of each vowel. None of the files were discarded under this modified script.

The VoiceSauce measurements included here were also, for comparison's sake, H1-H2 over the first third of each vowel, not corrected for formants, and again no files were discarded by the program.

The results are shown in Figure 2. Overall, the results from the three methods are similar. The "by-hand" measurements show the smallest mean differences across the categories, while the Praat measurements show the largest differences. The Praat measurements also show the greatest within-category variability, much greater than either of the other two methods, with about twice as much variability for the less modal phonations. The greater variability is due to greater variability of both H1 and H2 separately.

Figure 2: H1-H2 by three methods for vowels after three categories of stops. Colored bars distinguish the categories of stops. Error bars = standard deviations.



We suggest that VS's measurements could be less variable because (1) the STRAIGHT pitchtracker is very good when there is little creaking or pitch doubling, (2) having F0 values every ms avoids discontinuities, producing a smooth pitchtrack which makes the harmonic amplitude estimation likewise smoother, and (3) the optimization method for finding harmonic amplitudes is equivalent to using a very long FFT window. The "by-hand" and Praat methods both give one value of F0 and of each harmonic amplitude for the entire analysis window.

In sum, the measures from VS appear to maximize the number of files that can be analyzed, enhance the distinctions between categories, and minimize the within-category variability. VS also provides corrections for formants, plus some measures not currently implemented in Praat.

5. CONCLUSIONS

VoiceSauce is available in Matlab and freestanding PC versions for free download from [17]. We hope that researchers in speech, and in other areas who use speech data, will find it useful. The version described here is that of March 2011.

6. ACKNOWLEDGMENTS

This work was supported by NSF grant BCS-0720304 to P. Keating, A. Alwan, and J. Kreiman. We thank code contributors H. Tehrani and M. Iseli, and beta users C. Esposito, M. Garellek, S. Khan, J. Kuang, and H. Pan.

7. REFERENCES

[1] Boersma, P.,Weenink, D. 2008. Praat: doing phonetics by computer (Version 5.0.13). *http://www.praat.org/*

[2] Esposito, C.M. 2010. Variation in contrastive phonation in Santa Ana Del Valle Zapotec. *JIPA* 40, 181-198.

[3] Gerratt, B.R., Kreiman, J. 2001. Toward a taxonomy of nonmodal phonation. *J. Phon.* 29, 365-381.

[4] Hanson, H. 1997. Glottal characteristics of female speakers: Acoustic correlates. *J. Acoust. Soc. Am.* 101, 466-481.

[5] Harrington, J. 2010. *Phonetic Analysis of Speech Corpora*. Wiley-Blackwell.

[6] Hawks, J.W., Miller, J.D. 1995. A formant bandwidth estimation procedure for vowel synthesis. *J. Acoust. Soc. Am.* 97, 1343-1344.

[7] Hillenbrand, J., Cleveland, R., Erickson, R. 1994. Acoustic correlates of breathy vocal quality. *J. Sp. Hear. Res.* 37, 769-778.

[8] Iseli, M., Shue, Y.L., Alwan, A. 2007. Age, sex, and vowel dependencies of acoustic measures related to the voice source. *J. Acoust. Soc. Am.* 121, 2283-2295.

[9] Kawahara, H., Masuda-Katsuse, I., de Cheveigne, A. 1999. Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency based F0 extraction. *Sp. Comm.* 27, 187-207.

[10] de Krom, G. 1993. A cepstrum-based technique for determining a harmonic-to-noise ratio in speech signals. *J. Sp Hear. Res.* 36, 254-266.

[11] Lee, C. 2009. Identifying isolated, multispeaker Mandarin tones form brief acoustic input: A perceptual and acoustic study. *J. Acoust. Soc .Am.* 125, 1125-1137.

[12] Remijsen, B. Bert Remijsen's Praat scripts. *http://www.ling.ed.ac.uk/~bert/praatscripts.html*

[13] Sjölander, K. 2004. Snack sound toolkit. KTH Stockholm, Sweden. *http://www.speech.kth.se/snack*

[14] Sun, X. 2002. Pitch determination algorithm. *http://www.mathworks.com/matlabcentral/fileexchange/1230*

[15] Sun, X. 2002. Pitch determination and voice quality analysis using subharmonic-to-harmonic ratio. *Proc. ICASSP* '02, 333-336.

[16] Vicenik, C. 2010. An acoustic study of Georgian stop consonants. *JIPA* 40, 59-92.

[17] VoiceSauce download site: *http://www.ee.ucla.edu/~spapl/voicesauce/*