Interspeech 2018
2-6 September 2018, Hyderabad

A novel normalization method for autocorrelation function for pitch detection and for speech activity detection

Qiguang Lin[1,2] *and Yiwen Shao*[1,3]

[1] Baihu Technology, Co., Ltd, China
[2] ZIBOT, Inc., Santa Clara, California, USA
[3] Dept Computer Science, Johns Hopkins University, USA
qlin@agilespeech.com, yshao18@jhu.edu

Abstract

Autocorrelation functions (ACF) have been used in various pitch detection algorithms (PDA) and voicing-feature based speech activity detection (SAD) techniques. Speech is assumed to be stationary over a short-term window, and a Hanning window is typically applied in the calculation of ACF. As a result of windowing, the ACF tapers as the autocorrelation lags increase. Boersma demonstrated that the tapering effect could be compensated for by dividing the ACF of the windowed signal by the autocorrelation of the windowing function itself, referred to as wACF hereafter. We recently found that wACF could cause overcompensation and therefore, result in errors in pitch detection. In this paper, a novel normalization method, eACF, is proposed that can both mitigate the tapering effect and minimize the overcompensation. The new method is evaluated on synthetic speech and on the TIMIT database with various types of additive noise at different signal-to-noise (SNR) ratios. The results show that the new method leads to better performance both in terms of pitch detection and speech activity detection. In this paper, we also investigate the scenarios where applying the wACF method is advantageous and where it is not.

Index Terms: autocorrelation function, windowing effects, pitch detection algorithms, and speech activity detection

1. Introduction

Short-term autocorrelation functions (ACF) play an important role in speech processing, especially in pitch detection algorithms (PDA) and in voicing-feature based speech activity detection (SAD).

Pitch detection, the problem of determining the fundamental frequency of acoustic signals, is a significant component in large speech processing. Similarly, speech activity detection, i.e. the discrimination of the speech or nonspeech segments in an audio input, is another important part of speech applications. Efficient PDA and SAD methods can considerably improve the performance of large speech processing systems, such as speech recognition, speaker identification and speech coding systems. For this reason, advanced algorithms have been proposed for robust PDA and SAD in adverse acoustic environments [1, 2, 3].

The autocorrelation function based algorithms are well known to be comparatively robust against noise [1, 4, 5]. Assume that x(j) is the speech signal, and w(j) is the Hanning window of 32-ms long. The ACF of the windowed signal is then given by:

$$r_{xx}(t,k) = \sum_{j=0}^{N-1} [x(j)w(j)][x(j+k)w(j+k)], \qquad (1)$$

where N = 256 for a 32-ms window size and a sampling frequency of 8 kHz, j is the sample index, and t and k are the frame and autocorrelation lag indices, respectively.

$r_{xx}(t,k)$ has several important features including, for example, $r_{xx}(t,0) \geq r_{xx}(t,k)$, for all integers k and for any input signal. Consequently, eq. (1) is usually normalized by its value at k = 0:

$$r'_{xx}(t,k) = \frac{r_{xx}(t,k)}{r_{xx}(t,0)}. \qquad (2)$$

It is noted that, by definition, $r'_{xx,w}(t,k) \leq 1$. Another feature is that, for periodic signals, $r_{xx}(t,k)$ has (local) maximal values at $k = nT_0$, where n=1, 2, 3,… and T_0 denotes the period of the signal. The latter feature is utilized in speech and signal processing to determine if a signal is periodic, and if yes, what the corresponding period is [1, 4]. However, due to the windowing effect, $r_{xx}(t,k)$ tapers as the lag k increases. As a result, eq. (1) becomes less effective for detecting larger periods (i.e., lower pitches).

Boersma [6] demonstrated that the undesired tapering effect could be accounted for by dividing eq. (1) by the ACF of the window function itself, namely:

$$r_{ww}(t,k) = \sum_{j=0}^{N-1} w(j)w(j+k), \qquad (3)$$

$$r'_{ww}(t,k) = \frac{r_{ww}(t,k)}{r_{ww}(t,0)}, \qquad (4)$$

$$r'_{xx,w}(t,k) = \frac{r'_{xx}(t,k)}{r'_{ww}(t,k)}. \qquad (5)$$

Eq. (5) is referred to as the wACF method for simplicity. It is implemented in the popular open source Praat program for speech analysis [7], and is used in many recent studies of SAD [8, 9]. In addition, eq. (5) can mitigate strong oscillation of formants, which contributes to accurate pitch detection.

We recently found that eq. (5) would overcompensate the tapering effect. This overcompensation was recognized in [6], and the following remedy was suggested: First, compute the value of $r'_{xx,w}(t,k)$ in eq. (5). If the computed $r'_{xx,w}(t,k)$ is greater than 1, it will be replaced by its reciprocal. Note that this overcompensation tends to yield pitch detection errors also for higher pitches (larger lags).

10.21437/Interspeech.2018-45

In this paper, we propose a novel normalization method for the ACF by modifying the algorithm in [6]. We will show that eqs. (1) and (5) are two special cases of the new method, and that the new method, once optimized, has the combined advantages of combating the windowing effect without overcompensation. The new method is referred to as eACF.

This paper is organized as follows. In section 2, the new normalization method is introduced, together with a brief description about the synthetic speech used for evaluation experiments. In section 3, the eACF method is first evaluated for the PDA task using synthetic speech dataset, and in section 4, the eACF method is evaluated for the SAD task using the TIMIT dataset. Finally the paper is concluded with a summary in section 5.

2. New normalization method

In our recent study of using pitch continuity for robust SAD, we found that eq. (5) could cause overcompensation, and the overcompensation could not be completely undone by taking the reciprocal of eq. (5). Because the quantity $r'_{ww}(t,k)$ is always in the interval of (0, 1], and we decide to introduce an exponent to Boersma's algorithm [6] to effectively mitigate overcompensation (hence the name eACF). Mathematically, the eACF is given by:

$$r'_{xx,e}(t,k) = \frac{r'_{xx}(t,k)}{r'_{ww}(t,k)^{\beta}}, \qquad (6)$$

where $0 \le \beta \le 1$. It can been that eq. (6) is a generalized form of eqs. (2) and (5); That is, eq. (6) degenerates to eq. (2) when $\beta = 0$, and it is equivalent to eq. (5) when $\beta = 1$. By tuning the value of β, it is hoped that one can achieve the optimal trade-off between tapering and overcompensation so as to improve the performance of PDA and of SAD.

Figure 1 illustrates several ACFs for the waveform shown on the top. The waveform is produced by the LF model [10] as the excitation source. The vocal tract load is approximated by one formant, F1. In b) the long term ACF (over 50 periods) is depicted. No tapering is observed because of no need of windowing for long-term inputs. It represents the ideal scenario, and it accurately detects the fundamental period at 6.6 ms (corresponding to the pitch of 150 Hz), as marked by the location of the square box in b). On the other hand, c), d), and e) are all for short-term ACFs, and they are obtained by eqs. (2), (5), and (6), respectively. Although visual inspection easily reveals the tapering effect in c), the overcompensation in d) is less obvious. However, both b) and c) give wrong period values. Curve e) of the eACF method is seen to give right estimate of the period. It can also seen that estimation errors often pertain to octave errors, that is, the estimated period is either $\frac{T_0}{n}$ or nT_0, where n is a positive integer (practically, n = 2 or 3.)

Figure 2 gives another example of comparing different ACF curves for the same, short-term segment of speech input. It can clearly be seen that reciprocal-substitution occurs around the lag index of 120 for both β=1 and β=0.8. The superimposed curves in Figure 2 also facilitate to reveal the tapering impact (β=0 has the local maximum at lag of 17).

2.1. Data of synthetic speech

The β value used in Figure 1 is 0.9. In order to determine optimal value of β, a database of synthetic speech is prepared, where, again, the excitation source is based on the LF model, and one formant load is used to represent vocal tract response.

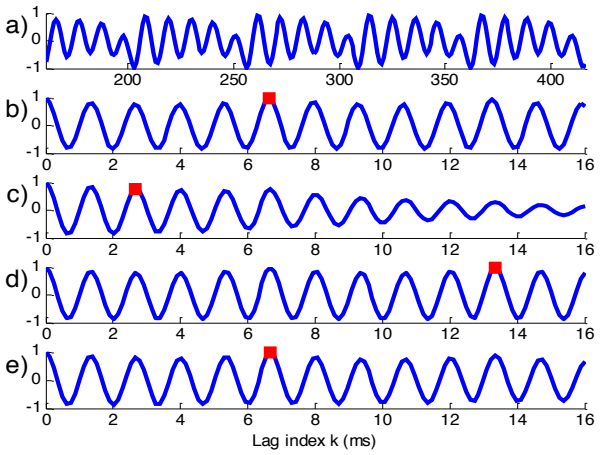

Figure 1: *Waveform and different autocorrelation functions of a synthetic speech (F0 = 150 Hz and F1 = 750 Hz). a) waveform shown as a function of sample index; b) the long-term ACF (50 fundamental periods, to avoid windowing need; c) short-term ACF computed by eq. (2); d) short-term ACF computed by eq. (5), the wACF method; and e) short-term ACF computed by eq. (6), eACF with β = 0.9. The red squares mark where the ACFs exhibit a peak in the plausible pitch range (from 62.5 Hz to 500 Hz).*

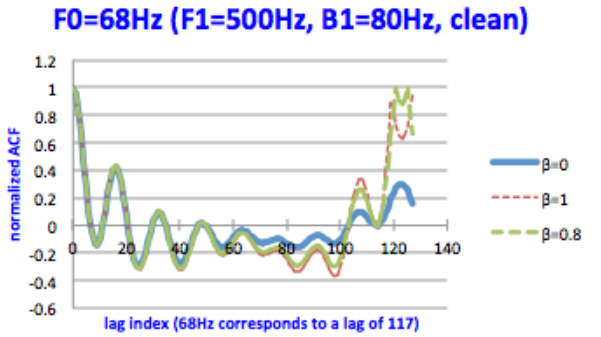

Figure 2: *Superimposed* ACF curves for the same speech segment, but for different *β values, to illustrate tapering, overcompensation, and reciprocal-substitution.*

The LF model has several timing parameters, see Figure 2. Without losing generality, T_a is set to 0, and T_e and T_p are made to covary T_0. T_0 is determined by the pitch values of interest: 65, 75, 100, 125, 150, 175, and 200 Hz. To examine the impact of formant oscillation, different frequencies and bandwidths of the first formant are considered: F1 varies from 250 Hz to 850 Hz with the step size of 50 Hz, while two bandwidths alternate between 100 and 50 Hz.

Finally, the clean, synthetic speech is added with different type of noise at various levels of SNR, with the help of the open-source software, FaNT [11]. The following noise types of noise are available: white, factory, volvo, babble (from the NOISEX-92 database [12]), subway noise (from the FaNT distribution), and finally an office background noise that we recorded [13]. The other details in the experiments are as follows:

- Sampling rate: 8000 Hz;
- Window size: 32 ms;
- Shift size: 10 ms; and
- SNR levels: 0, 5, 10, and 20 (dB).

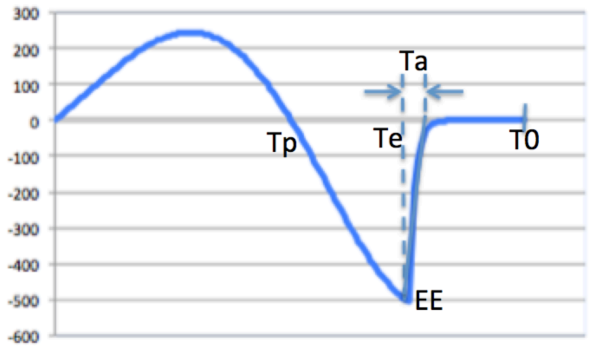

Figure 3: *The LF glottal source model with its timing parameters T_a, T_e, T_p, and T_0, and the relative excitation strength parameter EE at the instant of T_e. See [10] for more details.*

3. Performance evaluation of PDA

The synthetic speech is used to tune the parameter β. One of the advantages of using synthetic speech is of course the fact that the true period is known. As a result, we are able to be more accurate in evaluating the results than the gross error of 20% commonly used in the discussion of PDAs. If the estimated period is 3 lags or more away from the truth, it is regarded as an error.

We vary β from 0 (equivalent to eq. (2)) to 1 (equivalent to eq. (5) with a delta of 0.05. The experimental results are presented in Table 1 and Figure 4 below.

Table 1: PDA error rate as function of pitch, F0. For each F0, error rates for 3 values of β are presented: β = 0, β =1, and the optimal β with the lowest error rate.

F0 (Hz)	β=0 (%)	β=1 (%)	eACF (%)	Optimal β
65	79.79	30.45	26.07	0.8
75	56.78	20.91	17.40	0.85
100	33.37	17.31	17.31	1
125	23.73	50.00	18.72	0.55
150	15.58	51.23	15.42	0.15
175	15.46	51.92	15.13	0.25
200	14.42	59.13	14.42	0
Aggregate	34.15	40.13	19.9	0.7

In Figure 4, Curve a) is the overall error rate distribution. It is the sum of Curves b) and c), which, respectively, denote the errors with underestimated periods and with overestimated periods. It is seen that with β = 1, the PDA errors are dominated by overestimates: the detected periods are longer than the input period. This is expected since overcompensation gets more and more serious as the autocorrelation lags increase, It is noted that we have not considered pitch continuity in the above result analysis. This continuity feature can of course easily be incorporated to enhance the accuracy of pitch detection.

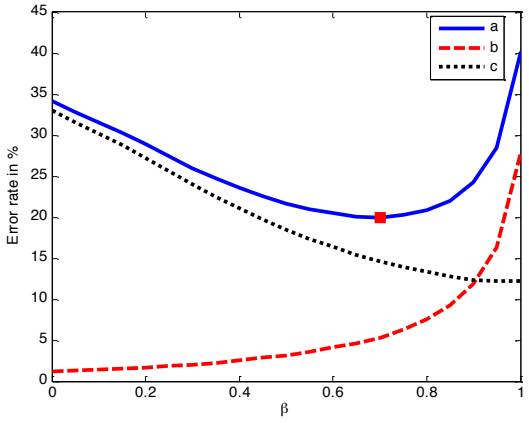

Figure 4: *pitch detection error rates in % by varying β. a: sum of curves b and c; b: error rates of underestimated pitch; and c: error rates of overestimated pitch. The red square indicates the minimum error rate, with β = 0.7.*

4. Performance evaluation of SAD

4.1. Testing data based on TIMIT

The New England subset of the TIMIT database [14] is used as clean data for experiments on SAD. It has a total of 38 speakers, with 10 utterances per speaker. The data is first down-sampled to 8 kHz, and then added with 3 noise types (volvo, factory and office) at 4 different SNR levels (0, 5, 10, and 20 dB).

4.2. Voicing features for SAD

Voicing features, those related to pitch and harmonics, have been utilized successfully for SAD, especially under the interference of ambient additive noise [8, 9].

Two popular voicing features are (i) harmonicity and (ii) clarity [8], with both being dependent on ACFs:

- Harmonicity (or harmonics-to-noise ratio): the relative height of the maximum autocorrelation peak in the plausible pitch range:

$$H(t) = \frac{r'_{xx,w}(t, k_{max})}{r'_{xx,w}(t,0) - r'_{xx,w}(t, k_{max})}. \qquad (7)$$

- Clarity: the relative depth of the minimum average magnitude difference function (AMDF) valley in the plausible pitch range, where AMDF is approximated by [15]:

$$AMDF(t,k) \approx 0.8\times\sqrt{2r_{xx,w}(t,0) - r_{xx,w}(t,k)}, \qquad (8a)$$

$$C(t) = 1 - \frac{AMDF(t, k_{min})}{AMDF(t, k_{max})}. \qquad (8b)$$

The two features are used to study how SAD performance varies as β of eq. (6) changes. We follow the approach in [8] for feature fusion based on principal component analysis, and for setting the threshold for decision-making. The resultant one-dimension feature is the final feature for SAD.

Error rates are estimated from the amount of time that is misclassified by the system, in the way as specified in the official NIST OpenSAD [16]. The Detection Cost Function (DCF) in [16] is used as the criterion for performance evaluation, which is given by:

$$\mathrm{DCF} = P_{Miss}\times 0.75 + P_{FA}\times 0.25, \qquad (9)$$

where P_{Miss} is the missing rate, and P_{FA} is the false alarm rate. The results are presented in Table 2 and Figure 5.

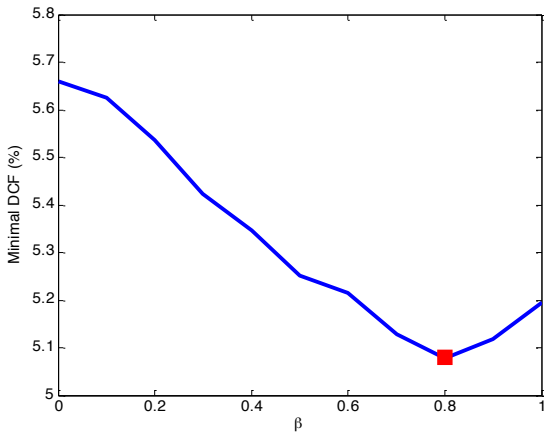

Figure 5: *Minimal DCF in % with varying β. The red square marks the minimum in the plot, at β = 0.8.*

Table 2: *Minimal DCF in % of different* β.

β	MinDCF (%)
0	5.66
1	5.19
0.8 (eACF)	**5.08**

As shown in Table 2 and Figure 5, the new method achieves the best performance at the vicinity of β = 0.8. But this time the difference between β = 0.8 and β = 1.0 (wACF) is marginal. Recall from Figure 4 that for pitch detection the wACF is notably inferior to the eACF (with β = 0.7). In other words, β=1.0 for SAD is more effective than for PDA. One explanation for such contrast is as follows. For PDA, it is the location of ACF's peaks that matters; while for SAD using harmonicity and clarity, both the location and its absolute amplitude value of ACF that matter jointly, especially the latter. An error in the location can mean an error in PDA. However, the location error may probably be offset by the minor difference between the amplitude of ACF at the true location and detected one. In such cases, comparatively good SAD performance is obtained even if the detected location is off when β = 1 [17]. On the other hand, the DCF at β = 0 is, however, significantly worse, explaining the popularity of using wACF for SAD.

5. Conclusion

A novel normalization method for the autocorrelation function is proposed in this paper, based on the algorithm outlined in [6]. In [6], the ACF of windowed signal is normalized by the ACF of the window function, to compensate for tapering effect resulting from windowing. But it is found that overcompensation may take place. In the new method, an exponent is introduced to minimize overcompensation, and hence the name of eACF.

Using a synthetic speech input, it is illustrated how short-term ACF differs from its long-term counterpart. (We are not aware of any attempt where humans can sustain identical phonation and articulation for relatively a long time.) In Figure 1, tapering effect, overcompensation, and effectiveness of eACF are all shown.

Using the LF source model we prepare a set of speech data where controlled changes to T_0, and the frequency and bandwidth of the first formant. The clean set of the data is next augmented by the noisy version by adding noise of different types and at different SNR levels. This set of data is used to determine the optimal value of β for PDA. The result shows that a minimum PDA error rate of 19.9% is achieved at β = 0.7. This error rate represents a 50% reduction of the error rate comparing to that of β = 1 or 0.

The eACF is then evaluated in term of robust SAD. The New England subset of the TIMIT corpus is used for this purpose. Again, the clean version is augmented with the noisy versions by adding noise. The voicing features, harmonicity and clarity [8], are used in the experiment. The result shows that the eACF is able to produce a lowered minimal DCF, see Figure 5.

The optimal value for β differs depending the task: PDA versus SAD. But the difference is reasonably small. We would suggest that a range of β from 0.7 to 0.8 be used.

In addition, it can be seen from Table 1 that different ranges of pitch are favored by different values of β. For instance, β =1 works better for low pitches, while β = 0 better for high pitches. It is therefore possible to dynamically adjust the value of β once we know the range of pitches of the incoming speech.

In the near future, we plan to evaluate our new method using real speech data (preferably with pitch information already manually marked.)

6. Acknowledgements

This work was in part supported by research grant from Ministry of Science and Technology, China, Contract# 14C-26213201061.

7. References

[1] W. Hess, *Pitch Determination of Speech Signals – Algorithms and Devices* (Springer, Berlin, Heidelberg), 1983.

[2] Shimamura, Tetsuya, and Hajime Kobayashi, "Weighted autocorrelation for pitch extraction of noisy speech," IEEE transactions on speech and audio processing Vol. 9 (7), pp. 727-730, 2001.

[3] Abdullah-Al-Mamun, K., F. Sarker, and G. Muhammad, "A high resolution pitch detection algorithm based on AMDF and ACF," Journal of scientific research, Vol. 1 (3), pp. 508-515, 2009.

[4] L. R. Rabiner, “On the use of autocorrelation analysis for pitch detection,” IEEE Trans. Acoustics, Speech, Signal Processing, vol. ASSP-25, no. 1, pp. 24–33, 1977.
[5] K. A. Oh and C. K. Un, “A performance comparison of pitch extraction algorithms for noisy speech,” in Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing, pp. 18B4.1–18B4.4, 1984.
[6] P. Boersma, “Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of the sampled sound,” in Proc. Inst. Phonetic Sci., Vol. 17, pp. 97-110, 1993.
[7] P. Boersma, “Praat, a system for doing phonetics by computer,” *Glot International* 5:9/10, 341-345. 2001.
[8] O. Sadjadi and J. Hansen, “Unsupervised speech activity detection using voicing measures and perceptual spectral flux,” IEEE Signal Processing Letters, Vol. 20, pp. 197-200, 2013.
[9] E. Chuangsuwanich and J. Glass, “Robust voice activity detector for real world applications using harmonicity and modulation frequency,” InterSpeech, pp. 2645-2648, 2011.
[10] G. Fant, J. Liljencrants, and Q. Lin, “A four-parameter model of the glottal flow”, STL-QPSR, KTH, Stockholm, pp. 1-13, 1985.
[11] H. Hirsch, “FaNT - filtering and noise adding tool”. Online available: http://dnt.kr.hs-niederrhein.de/download.html.
[12] A. Varga, H. Steeneken, “Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems,” Speech Communication, Vol. 12, pp. 247-251, 1993.
[13] Y. Liu, J. Wang, Q. Lin, and S. Wang, “A novel speech activity detection algorithm based on the fusion of time domain and frequency domain features,” J. Jiangsu University of Science & Technology (in Chinese). Vol. 31 (01): 73-78, 2017.
[14] J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallett, N. Dahlgren, and V. Zue. “Timit acoustic-phonetic continuous speech corpus,” [online]. Available: http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC93S
[15] L. Rabiner and R. Schafer, Theory and Applications of Digital Speech Processing, Prentice-Hall, 2010.
[16] “Evaluation Plan for the NIST Open Evaluation of Speech Activity Detection (Open-SAD15),” NIST, USA. [online] Available: https://www.nist.gov/sites/default/files/documents/itl/iad/mig/Open_SAD_Eval_Plan_v10.pdf
[17] Y. Shao and Q. Lin, “Use of pitch continuity for robust speech activity detection.” Proc. IEEE-ICASSP, pp. 5534-5537, Calgary, Canada, 2018.