A Pitch Detection Algorithm for Continuous Speech Signals Using Viterbi Traceback with Temporal Forgetting

J. Bartošek

Abstract

This paper presents a pitch-detection algorithm (PDA) for application to signals containing continuous speech. The core of the method is based on merged normalized forward-backward correlation (MNFBC) working in the time domain with the ability to make basic voicing decisions. In addition, the Viterbi traceback procedure is used for post-processing the MNFBC output considering the three best fundamental frequency (F0) candidates in each step. This should make the final pitch contour smoother, and should also prevent octave errors. In transition probabilities computation between F0 candidates, two major improvements were made over existing post-processing methods. Firstly, we compare pitch distance in musical cent units. Secondly, temporal forgetting is applied in order to avoid penalizing pitch jumps after prosodic pauses of one speaker or changes in pitch connected with turn-taking in dialogs. Results computed on a pitch-reference database definitely show the benefit of the first improvement, but they have not yet proved any benefits of temporal modification. We assume this only happened due to the nature of the reference corpus, which had a small amount of suprasegmental content.

Keywords: PDA, fundamental frequency, Viterbi, temporal forgetting, MNFBC, speech processing.

1 Introduction

Almost every audible sound tends to have a fundamental frequency. This is the lowest frequency on which the signal is periodic, and we sense this frequency as the height (pitch) of the sound. Human speech perception is partly based on intonation (changes of pitch), which is an aspect of prosody. Thanks to this we can distinguish whether a person is making a statement or a question [1]. Prosodic information also enables us to recognize the emotions of a speaker. A motivation for finding a precise and robust PDA could be to track the intonation contour in continuous speech. This is a crucial step for the proper function e.g. of a punctuation detector [2] or an emotion classifier of the speaker.

There are nowadays several known pitch detection methods. They can generally be divided according to the domain in which they operate (time, frequency, cepstrum, etc.) An overview of some basic methods can be found in [12]. The most widely used methods are probably various forms of autocorrelation algorithms (time and frequency domain), based on similarity of the signal itself after some time period (time domain) or periodicity in the spectrum (frequency domain). AMDF [5] (time domain), the cepstral method [4] (modification of the spectrum domain) and sub-harmonic summation (SHS) [3] are well described and widely used methods.

2 A description of PDA using MNFBC

A critical aspect of PDAs used for speech analysis is that there are fast transitions between articulated phonemes. For this reason, the best result for a speech signal (in contrast to a singing signal) is obtained by methods that work in the time domain.

The PDA that is used and improved in this paper emerges from the very complex PDA described in detail in [9]. The core of the pitch-detection method is a merged normalized forward-backward correlation. Equation (1) presents the basic correlation term that is used in equations (2) and (3) to compute forward and backward correlations, where the constant MAX_PER refers to the time period of the lowest detectable frequency. These two correlations combined together lead to the MNFBC function (4), which is then half-way rectified. Combining two opposed correlations into a final correlation should improve the precision for frames with problematic content in terms of the different nature of the beginning and ending parts (transitions).

$$\langle x_{w_k}[n], x_{w_l}[n]\rangle = \sum_{n=0}^{2*MAX\_PER-1} x_w[n+k]x_w[n+l] \quad (1)$$

$$NFC[t] = \frac{\langle x_{w_0}[n], x_{w_t}[n]\rangle}{\sqrt{\langle x_{w_0}[n], x_{w_0}[n]\rangle\langle x_{w_t}[n], x_{w_t}[n]\rangle}} \quad (2)$$

$$NBC[t] = \frac{\langle x_{w_{2MAX\_PER}}[n], x_{w_{2MAX\_PER-t}}[n]\rangle}{\sqrt{\langle x_{w_{2MAX\_PER}}[n], x_{w_{2MAX\_PER}}[n]\rangle \langle x_{w_{2MAX\_PER-t}}[n], x_{w_{2MAX\_PER-t}}[n]\rangle}} \quad (3)$$

$$MNFBC[t] = \frac{\langle x_{w_0}[n]], x_{w_0}[n]]\rangle (NFC'[t])^2 + \langle x_{w_{2MAX\_PER}}[n], x_{w_{2MAX\_PER}}[n]\rangle (NBC'[t])^2}{\langle x_{w_0}[n], x_{w_0}[n]\rangle + \langle x_{w_{2MAX\_PER}}[n], x_{w_{2MAX\_PER}}[n]\rangle} \quad (4)$$

In contrast to [9], there is neither a signal preprocessing section in our algorithm nor a special block determining whether the segment of speech is voiced or unvoiced. This decision is made by thresholding the MNFBC value itself. There is also no special block ensuring correct pitch with a sub-harmonic summation (SHS) function [3] to prevent halving errors. With the improvements suggested in this paper, this is not needed (see section 3 for details).

The computational requirements are determined by the complexity of the correlation operation done in the time domain, which is N^2, where N is the length of the processed window in the samples. This is worse than the complexity $N \log(N)$ of faster PDAs operating in the frequency domain and using FFT. Although real-time use in the final implementation is wanted, we assume that the algorithm will not be used in any time-critical or embedded system where computation complexity is a critical issue.

3 Viterbi post-processing

Post-processing using the Viterbi algorithm [6] is applied to find the optimal track of the pitch. The power of the Viterbi procedure lies in its ability to apply user-defined rules for comparing the candidates. However, for proper use of the algorithm some requirements have to be met. Each candidate needs to have assigned its "emission" probability b_k (the probability that candidate k is F0 for the current frame, without considering any history) and also its "transition" probability a_{kl}, which denotes how probable it is that candidate k in the current frame will be followed by candidate l in the next frame. Having these context-independent values, we can gradually compute the values of function $\delta_{m,l}$ which tells the final probability of candidate l being F0 for frame m considering the results from the previous frames. Function $\psi_{m,l}$ designates the index of the most probable candidate in frame $m-1$. The equations can then be expressed as (5) and (6).

$$\delta[m,l] = \max_k [\delta[m-1,k]a[k,l]]b[l] \quad (5)$$

$$\psi[m,l] = \arg\max_k [\delta[m-1,k]a[k,l]] \quad (6)$$

The algorithm starts by assigning for the first frame $\delta_{1,i} = b_i$ and $\psi_{1,i} = 0$. In the current implementation, the three best candidates (the three highest peaks) come from the MNFBC function. Note that these candidates often (but not always) correspond to the harmonic content of a speech signal [11]. This means that in most cases the candidate with the highest MNFBC value is really F0, and the two other highest values are harmonics of F0 (its natural multiples). However, there could also be cases when the harmonics are "stronger" than fundamental. In this case, the Viterbi procedure should prevent halving or doubling errors. The basic scheme of the algorithm is depicted in Figure 1.

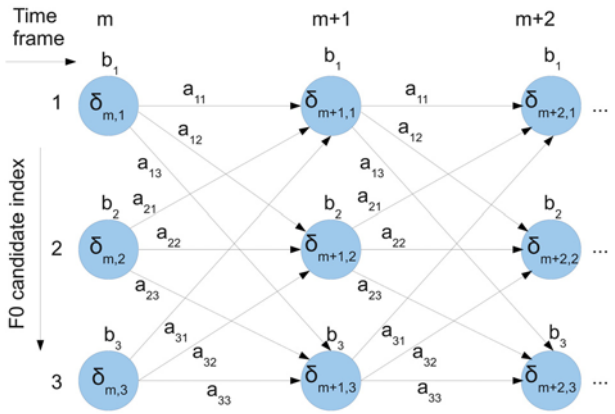

Fig. 1: The trellis of the Viterbi algorithm

The emission probability b_k is implemented directly as the value of the MNFBC function for each F0 candidate. The transition probability can be computed according to [9] as the decreasing exponential of the frequency difference. This could work well for some range of low fundamental frequencies with a suitable multiplying constant for the difference. However, our perception of pitch is not linear with growing frequencies, but is logarithmic [11]. This means that the same difference in frequency in a lower frequency band causes a greater pitch change perception than the same difference in a higher frequency band. For this reason, the difference in frequency in Hz is converted in our algorithm to the difference in pitch in musical manners — semitones and cents.

Let variable x be the difference of frequencies for consequent candidates converted to musical cents (100 cents = 1 semitone) according to equation (7). The resulting transition probability function $a(x)$ is then expressed as (8), and the function is visualized

in Figure 2. Multiplying constant 0.001 2 was experimentally found to give the best results. Overall results that improve the precision of the algorithm with this modification can be found in section 4.

$$x = 1\,200 \left| \log_2 \left(\frac{f1}{f2} \right) \right| \tag{7}$$

$$a(x) = \frac{1}{\mathrm{e}^{0.001\,2x}} \tag{8}$$

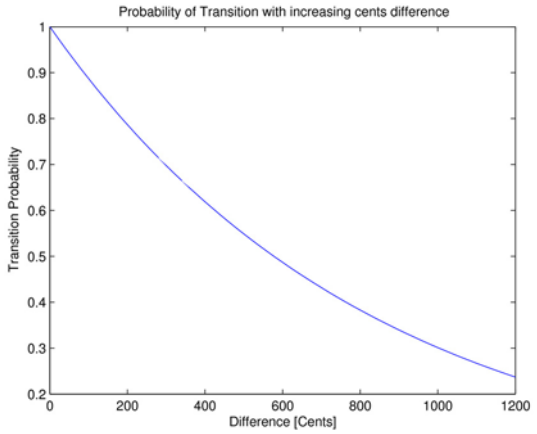

Fig. 2: Transition probabilities function depending only on the difference in cents

Now let us consider a situation when it is possible to have a jump in the pitch of speech in the place of the border of neighbouring prosodic units [1] (with unvoiced segments between them, so that the first voiced segment of the new prosodic unit is the next voiced segment for the last prosodic unit in terms of the Viterbi algorithm). The previous probability function will not allow the change to be immediately applied to the pitch track, and needs some time to "adopt" the new pitch level.

Most utterances take place in the range of the musical fourth (which is 5 semitones = 500 cents in terms of explicit musical distance). This is not the overall pitch range, but it is the common range that we use across prosodic units, sometimes referred to in the literature as the "pitch sigma". It is probable that the biggest jump in pitch of 5 semitones will not occur very often, and only on the boundaries of prosodic units. However, we permit this jump to be possible without any penalization after a long enough prosodic pause. For this reason, a difference of 500 cents is a limit, and higher differences will be penalized by a linear decrease.

The behaviour of the temporal probability function of two variables (cent difference x and time t) can thus be expressed in the cent difference interval $x \in \langle 0, 500 \rangle$ as:

$$a(x,t) = \mathrm{e}^{-0.0012x} + \frac{(1 - \mathrm{e}^{-0.001\,2x})t}{T_{thr}} \tag{9}$$

and on the interval $x \in \langle 500, 1\,200 \rangle$ as:

$$a(x,t) = \mathrm{e}^{-0.001\,2x} + \frac{\left(\frac{1\,200-x}{700} - \mathrm{e}^{-0.001\,2x}\right) t}{T_{thr}} \tag{10}$$

where T_{thr} is the time forgetting threshold. When the prosodic pause length reaches this value, all pitch changes in the range of 500 Hz have a transition probability of 1.

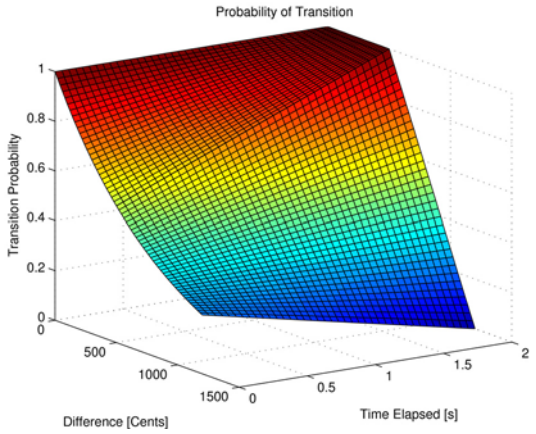

Fig. 3: Transition probability function depending on difference in cents and on time

4 Results

4.1 Test conditions

All the results were computed using a manually labeled pitch-reference database as a part of Spanish SPEECON [10], with the use of a pitch evaluation framework [7]. All parts of the proposed algorithm were implemented in the MATLAB environment.

4.2 Evaluation criteria

The results section uses the evaluation criteria suggested in [7]. The voiced error VE (unvoiced error UE) rate is the proportion of voiced (unvoiced) frames misclassified as unvoiced (voiced). The gross error high GEH (gross error low GEL) is the rate of F0 estimates (correctly classified as voiced) which does not meet the 20 % upper (lower) tolerance of frequency in Hz. The GEH and GEL 20 % tolerance range is quite broad, and thus cannot distinguish clearly between two precise PDAs. For this reason, GEH10 and GEL10 were established by analogy with GEH and GEL, but with only 10 % tolerance ranges. These new criteria are also expected to result in higher error rates than the older criteria, but might be useful in applications where precision matters. UE+VE and GEH+GEL criteria are sometimes used to summarize PDA errors. Halving errors (HE — the estimated frequency is half of the refe-

Table 1: Channel 0 overall results

PDA	VE [%]	UE [%]	VE+UE [%]	GEH [%]	GEL [%]	GEH10 [%]	GEL10 [%]	DE [%]	HE [%]
ACF freq	44.4	23.5	31.6	1.2	0.1	1.5	0.18	0.4	0.06
DFE	26.6	15.5	20.4	8.4	4.2	16.5	8.9	0.2	1.3
MNBFCv1	22	12.7	16.3	0.4	21.2	1.5	22.1	0.06	19.5
MNBFCv2	22	10.7	15	0.4	1.1	1.8	2.3	0.03	0.8
MNBFCv3	18.5	13.3	15.3	0.5	1.2	1.9	2.6	0.05	0.8
MNBFCv4	15.6	16.3	16	0.6	1.3	2	2.8	0.05	0.9
MNBFCv5	22	10.7	15	0.7	1.7	2.1	2.9	0.14	1

Table 2: Gross errors (GEL+GEH [%]) in 2/3 octave frequency bands on Channel 0

PDA	57–88 [Hz]	88–141 [Hz]	141–225 [Hz]	225–353 [Hz]	353–565 [Hz]
ACF freq	92.7	5.2	0.6	1.1	17.7
DFE	26.4	12.1	12.3	13.0	53.4
MNBFCv1	2	1.9	28.3	49.7	73.7
MNBFCv2	1.1	0.7	1.7	2.2	32.1
MNBFCv3	1.7	0.9	2	2.3	33
MNBFCv4	2.3	1	2.3	2.7	34.4
MNBFCv5	2.2	1.6	2.8	2.9	32.1

rence) and doubling errors (DE) were also brought in with a tolerance of 1 semitone range from half or double the reference F0. Errors of this kind are a special type of gross errors and often occur on real PDA outputs for noisy signals or transitions from voiced to unvoiced speech elements. We may sometimes need to observe the errors not in the entire frequency band but e.g. within 5 smaller frequency sub-bands individually (2/3 octave bands were used to cover the range from 60 to 560 Hz).

4.3 Results and discussion

Table 1 shows the overall results for the highest signal-to-noise (SNR) ratio channel 0 of the reference database. MNBFCv1 is the basic variant with the voiced/unvoiced (V/UV) decision threshold set to value 0.5 and with the transition probability of the Viterbi procedure computed from the direct frequency difference. MNBFCv2 improves the first variant with the conversion difference to cents. MNBFCv3 is almost the same as MNBFCv2, but has the V/UV threshold set to 0.45, whereas MNBFCv4 has the threshold value set to 0.4. The final MNBFCv5 involves adding the temporal domain to the transition probability function with the time forgetting threshold set to 2 seconds. Table 2 presents a comparison of the precision over five distinct frequency bands. To compare our method with other widely used methods, we added the results for autocorrelation in the frequency domain (ACF freq, a very good method for tracking singing) and the Direct Frequency Estimation method (DFE) [8], which is currently used for evaluating Parkinson's disease at FEE CTU in Prague.

The results show that MNFBC is better than DFE in V/UV detection and also in precision. The VE+UE parameter is the best for MNBFCv2, but we can achieve the best VE ratio for MNBFCv4 (but with a worse UE rate). The choice of variant depends on the target application — whether we need to minimize voiced errors or unvoiced errors. For example, in the case of the planned punctuation detector we are trying to minimize the unvoiced error rate in order to obtain only confident F0 estimates. The results also show a big increase in precision with frequency difference (MNBFCv1) to cent conversion (MNBFCv2). Progress can be seen mainly in GEL and in the halving error rate. Table 2 shows that most errors for MNBFCv1 occur in the highest band, where the dif-

ferences from the current frequency are much greater in Hz units than for lower bands. Thus these transitions are evaluated with very low probability, leading to these errors. The table also shows that the ACF method can provide the best results for the highest frequency band, but is very poor in the lowest band. MNBFCv5 with temporal forgetting could probably not show its strength on the reference corpus due to lack of suprasegmental prosodic phrases (the corpus consists mainly of isolated words). In comparison with MNBFCv2, however, there is only a slightly higher GEH rate. Globally, MNFBC with the addition of the Viterbi traceback procedure outperforms DFE on close talk channel 0. Note that it has much lower GEH and GEL even for lower VE. This is not easy to achieve for PDA, because lower VE means that more uncertain segments (which other PDAs with higher VE have considered as unvoiced) pass to computation of the precision of F0 detection (gross errors).

Other results not presented in this paper have also shown a noticeable decrease in precision on channel 1 with the algorithm presented here. To get good results even in noisy environments, higher noise robustness is needed. This could be accomplished by adding a pre-processing stage with noise reduction (not implemented yet).

5 Conclusion

We have described a pitch-detection algorithm purely based on merged normalized forward-backward correlation (MNFBC) with an advanced Viterbi post-processing procedure for finding the most probable pitch track. The optimal range of the voicing threshold was found for the MNFBC function. The results confirm that computing the transition probabilities with the pitch difference measured in semitones significantly improves the gross error rates (especially frequency halving) over the case of direct difference of frequencies. We have also tried to extend the transition probability function with a temporal dimension. This enhancement should lead to fewer errors occurring on the edges of prosodic pauses, but this has not been proven in experiments performed on a pitch reference database. This could be due to the very limited presence of supra-segmental prosodic pauses in the corpus. More experiments on suitable utterances need to be performed in order to evaluate this hypothesis.

Acknowledgement

The research presented in this paper was supervised by Ing. Václav Hanžl, FEE CTU in Prague. It has been supported by the Czech Grant Agency under grant No. 102/08/0707 "Speech Recognition under Real-World Conditions" and by grant No. 102/08/H008 "Analysis and modelling biomedical and speech signals".

References

[1] Palková, Z.: *Fonetika a fonologie češtiny.* Praha : Karolinum, 1994.

[2] Kim, J., Woodland, P. C.: The use of prosody in a combined system for punctuation generation and speech recognition. *In Proceedings Eurospeech* (2001), 2 757–2 760.

[3] Hermes, D. J.: Measurement of pitch by subharmonic summation. *J. Acoust. Soc. Am.* **83** (1988), 257–264.

[4] Noll, A. M.: Cepstrum pitch determination. *J. Acoust. Soc. Am.* **41** (2) (August 1966), 293–309.

[5] Ross, M. J., Shaffer, H. L., Cohen, A., Freudberg, R., Manley, H. J.: Average magnitude difference function pitch extractor. *IEEE Trans. Acoust. Speech Signal Process.* ASSP-**22** (5) (October 1974), 353–361.

[6] Viterbi, A. J.: Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Trans. Inform. Theory*, vol. IT-13, (April 1967), 260–269.

[7] Bartošek, J.: Pitch detection algorithm evaluation framework. *20th Czech-German Workshop on Speech Processing, Prague* (2010), 118–123.

[8] Bořil, H., Pollák, P.: Direct time domain fundamental frequency estimation of speech in noisy conditions. *In the Proceedings of EUSIPCO 2004 (European Signal Processing Conference*, Vol. 1) (2004), 1 003–1 006.

[9] Kotnik, B., et al.: Noise robust F0 determination and epoch-marking algorithms. *Signal Processing* **89**(2009), 2 555–2 569.

[10] Kotnik, B., Höge, H., Kacic, Z.: Evaluation of pitch detection algorithms in adverse conditions. *Proc. 3rd International Conference on Speech Prosody, Dresden, Germany* (2006), 149–152.

[11] Syrový, V.: *Hudební akustika*, 2nd ed. Praha : HAMU, 2008.

[12] Uhlíř, J.: *Technologie hlasových komunikací.* Praha : ČVUT, 2007.

About the author

Jan BARTOŠEK was born in 1984 in Litoměřice (CZ) and attended the primary and the secondary school there. He completed his bachelor degree (2007 — Informatics and Computer Science) and his master degree (2009 — Software Engineering) at the department of Computer Science, Faculty of Electrical Engineering, CTU in Prague. He is now a student on the doctoral study programme at CTU Prague, Dept. of Circuit Theory, and is interested in voice and music technologies.

Jan Bartošek
E-mail: bartoj11@fel.cvut.cz
Dept. of Circuit Theory
Faculty of Electrical Engineering
Czech Technical University in Prague
Technická 2, 166 27 Praha, Czech Republic