



A Time-Warping Pitch Tracking Algorithm considering fast f_0 changes

Simon Stone*, Peter Steiner*, Peter Birkholz

Institute of Acoustics and Speech Communication, Technische Universität Dresden, Germany

simon.stone@tu-dresden.de, peter.steiner@mailbox.tu-dresden.de,
peter.birkholz@tu-dresden.de

Abstract

Accurately tracking the fundamental frequency (f_0) or pitch in speech data is of great interest in numerous contexts. All currently available pitch tracking algorithms perform a short-term analysis of a speech signal to extract the f_0 under the assumption that the pitch does not change within a single analysis frame, a simplification that introduces errors when the f_0 changes rather quickly over time. This paper proposes a new algorithm that warps the time axis of an analysis frame to counteract intra-frame f_0 changes and thus to improve the total tracking results. The algorithm was evaluated on a set of 4718 sentences from 20 speakers (10 male, 10 female) and with added white and babble noise. It was comparative in performance to the state-of-the-art algorithms RAPT and PRAAT to Pitch (ac) under clean conditions and outperformed both of them under noisy conditions.

Index Terms: fundamental frequency estimation, pitch detection, pitch tracking.

1. Introduction

Fundamental frequency (f_0) and the related pitch are relevant for various research fields in the context of speech processing [1]. Intonation as part of linguistics for example directly refers to the pitch: According to Taylor [2, pp. 122], a question is expressed by an increasing pitch at the end of the phrase and a short time increase of the pitch is used to stress a specific word or syllable, which can be very important to resolve ambiguities. Also, the intentions as well as the emotions of a speaker influence the f_0 contour [3], a fact which is for example used in [4] to synthesize and recognize emotional speech and, in conjunction with other features, exploited in text-independent speaker recognition [5].

But the seemingly simple task of extracting the time-varying f_0 from an utterance is in practice complicated by numerous confounding factors: Unvoiced sounds do not contain an f_0 , thus making a voiced-unvoiced classification necessary. Within voiced segments, the f_0 is not necessarily the spectrally most pronounced frequency (due to the formants of the vocal tract), which eliminates simple global search algorithms. Instead, more elaborate strategies are necessary to yield a truly robust pitch tracking algorithm. But even with highly tuned algorithms, background noise (especially colored or even babble noise) can greatly decrease the recognition accuracy and although some de-facto standards have emerged, the problem of pitch tracking is by no means definitively solved.

In [6], Strömbergsson reviewed studies employing the most frequently used methods for f_0 estimation and tracking (in descending order): PRAAT To Pitch (ac) (PRAAT for short) [7], RAPT [8], STRAIGHT [9], and YIN [10]. Although PRAAT's autocorrelation method for f_0 estimation [11] is, according to

the review, by far the most frequently cited algorithm, its popularity is arguably due to its availability since it is not generally superior to the other techniques. In fact, depending on the applied error measure, YIN (the least commonly used algorithm) can outperform PRAAT and RAPT (see [6]). Since it is currently not part of any speech analysis software, however, it remains fairly unused. STRAIGHT is actually a vocoder that offers the possibility for pitch tracking but is not commonly used for analytical purposes outside of synthesis applications.

RAPT is based on the normalized cross-correlation function, which only uses a very short segment (assumed to contain roughly one glottal pulse) and correlates it with a longer segment from the original signal. PRAAT and YIN are both based on the autocorrelation function and thus need a longer correlation window. In addition, the analysis window length for PRAAT depends on the lower frequency of the expected f_0 band, which further increases the computational cost for a less supervised approach. Both PRAAT and RAPT estimate several “best guesses” for the actual f_0 within a frame (called f_0 candidates) in a first stage. In a second stage, the sequence of f_0 candidates over all frames of the analyzed signal spans a Dynamic Programming grid in which the optimal path is found using a Viterbi search. In this stage, several factors like voiced-unvoiced transitions and octave jumps are taken into account (see below).

YIN uses a periodicity function for the candidate search, which is based on the autocorrelation function and comes at about the same computational cost. The periodicity function is more robust against low formant frequencies (which may be easily mistaken for the f_0) and the window, which has a constant length of 0.025 s, leads to a good frequency resolution. However, as presented in [10], YIN only identifies a set of likely f_0 candidates for each frame and does not contain a second stage to find the actual f_0 contour over time. This leads to a lot of voiced-unvoiced decision errors and thus the results of YIN reported by [6] were generally worse than the results of PRAAT (which performed best overall) or RAPT (which came in second).

All algorithms make one common assumption: that the analyzed speech signal and, consequently, the f_0 is stationary within an analysis frame, which is true for very short time periods (strictly speaking, only for the duration of one glottal pulse). Contrary to that and due to the uncertainty principle in signal processing, the analysis frame should be as long as possible to ensure a suitable frequency resolution. Several studies (e.g., [12, 13, 14]) examined the change of the f_0 and found pitch changes as fast as 4.16 oct s^{-1} [14]. Considering the default analysis frame length of 40 ms for PRAAT and 27.5 ms¹ for

¹RAPT uses a shorter segment with a default length of 7.5 ms as a user-adjustable parameter, but this segment is crosscorrelated with a longer segment of this length, which is the relevant length when considering intra-frame f_0 changes.

*The first two authors contributed equally to this work.

RAPT, this means changes of the f_0 within a single analysis frame of up to approx. 2 st for PRAAT and 1.4 st for RAPT. Therefore, many pitch detection errors may be related to these fast changes that undermine the basic assumption behind those algorithms.

To account for these rapid f_0 changes, Birkholz [15] presented an idea based on warping the time axis with a non-linear function to normalize the instantaneous frequencies within a single frame. In this paper, we built upon this basic idea and developed it into a Time-Warping Pitch Tracking Algorithm (TPTA).

2. The Time-Warping Pitch Tracking Algorithm

The outline of the Time-Warping Pitch Tracking Algorithm is visualized in Figure 1. The pre- and post-processing of the TPTA is similar to the established state-of-the-art algorithms: the speech signal is split up into short frames of the length L (in samples), f_0 candidates are estimated for each frame, and eventually the best path through the sequence of candidates through time is determined by a Viterbi search. However, in addition to the procedures of state-of-the-art algorithms, a time-warping step is inserted before the determination of the f_0 candidates.

2.1. Windowing and time-warping

The signal is divided into single frames $s[n]$ using a rectangular window of length L (in samples), where n is the discrete time index. In this frame, the oscillation with the fundamental frequency $f_0 = \frac{\omega_0}{2\pi}$ can be described by a (without loss of generality continuous) sine wave with a phase $\varphi_0(t)$:

$$x_0(t) = x_0 \sin(\varphi_0(t)), \quad (1)$$

The phase $\varphi_0(t)$ is defined as:

$$\varphi_0(t) = \int \omega_0(t) dt. \quad (2)$$

In the stationary case, ω_0 is constant and thus leads to a linear phase. However, ω_0 is in fact not necessarily constant within the analysis frame but can be approximated as a linear function over time. The result of (2) therefore becomes non-linear:

$$\varphi_0(t) = \int (\omega_0(0) + mt) dt = \omega_0(0)t + \frac{m}{2}t^2 + \varphi_0(0) \quad (3)$$

To linearize the phase $\varphi_0(t)$ in the analyzed frame and thus force a (seemingly) constant frequency ω_0 , the time axis has to be warped to compensate the quadratic term in (3). This warping has the boundary constraints that the phase at the beginning and end of the frame should remain the same, thus ensuring a continuous phase between frames. Additionally, the frequency within a frame is assumed to linearly increase or decrease from $f_0(0)$ to $f_0(T)$ (where T is the window length in seconds) and the constant frequency in warped time is set to the mean frequency in linear time. Equating the quadratic phase function in linear time t (3) to a linear target phase function in warped time t' yields a warping function $t(t')$:

$$t(t') = \frac{T}{1-r} \cdot \left(1 - \sqrt{1 - \frac{t'}{T} (1-r^2)} \right), \quad (4)$$

where $r = \frac{f_0(T)}{f_0(0)}$ describes the (unknown) change of the f_0 between the beginning and the end of the frame.

When this relationship is used to obtain a discrete warped signal $s_w(n\Delta t(t'))$ from a discrete linear-time signal $s(n\Delta t)$, the sampling times become non-uniformly distributed along the time axis because the effective sampling period Δt is no longer constant. Therefore, the signal is equidistantly resampled using linear interpolation to obtain the final signal $s_w[n\Delta t]$ (shortened to $s_w[n]$ for the remainder of the discussion).

To parameterize the warping in a more intuitive fashion, the ratio r is converted to the dimension oct s^{-1} and yields the user-adjustable warping factor w :

$$w = \frac{\log_2 r}{L}, \quad (5)$$

where $r = \frac{f_0(L-1)}{f_0(0)}$ is the frequency ratio of the discrete signal frame of length L in samples.

In the TPTA, nine warped signals $s_w[n]$ are calculated using nine integer values on the interval $\pm 4 \text{ oct s}^{-1}$ for w . If, for example, the f_0 increased by 3 oct s^{-1} in the original frame, then the signal $s_{-3}[n]$ would be “most periodic” with a quasi-constant f_0 , because the f_0 increase was exactly compensated. Even if the unknown true ratio is not exactly matched by any of the nine values for w , the signal still becomes “more periodic” the closer w gets to the true value.

2.2. Periodicity function and identification of optimal warping parameters

For each of the nine warped signals $s_w[n]$, the periodicity function (PF) by [10] is calculated analogously to YIN:

$$p_w[k] = \begin{cases} 1 & \text{if } k = 0 \\ \frac{1 - \phi_w[k]}{\frac{1}{k} \sum_{i=1}^k (1 - \phi_w[i])} & \text{else,} \end{cases} \quad (6)$$

where $\phi_w[k]$ is the non-stationary autocorrelation function (ACF):

$$\phi_w[k] = \frac{\sum_{n=0}^{N-1} s_w[n] s_w[n+k]}{\sqrt{E_w[0] E_w[k]}}, \quad (7)$$

and E_k is the energy of the shifted signal:

$$E_w[k] = \sum_{n=0}^{N-1} s_w^2[k+n], \quad (8)$$

where N is the integration length and k denotes the lag. The main advantage of the periodicity function over the ACF (which is used in PRAAT) is that the PF does not have an extremum at lag 0 and that it is more robust against low first formant frequencies.

To illustrate the effect of the time-warping of a signal with a linear decreasing f_0 , Figure 2 shows a synthesized vowel /a/ and the corresponding ACF and PF for no warping and a warping that matches the f_0 change. However, in a real-world setting, the exact f_0 change will be unknown. Therefore, TPTA calculates the PF $p_w[k]$ of all nine signals $s_w[n]$ and searches for the global minimum of all $p_w[k]$. Because the minimum value of a $p_w[k]$ becomes smaller the better the corresponding w approximates the actual f_0 change in the signal, the PF $p_{w_{opt}}[k]$ that contains the global minimum of all PF is determined and used for the further calculations.

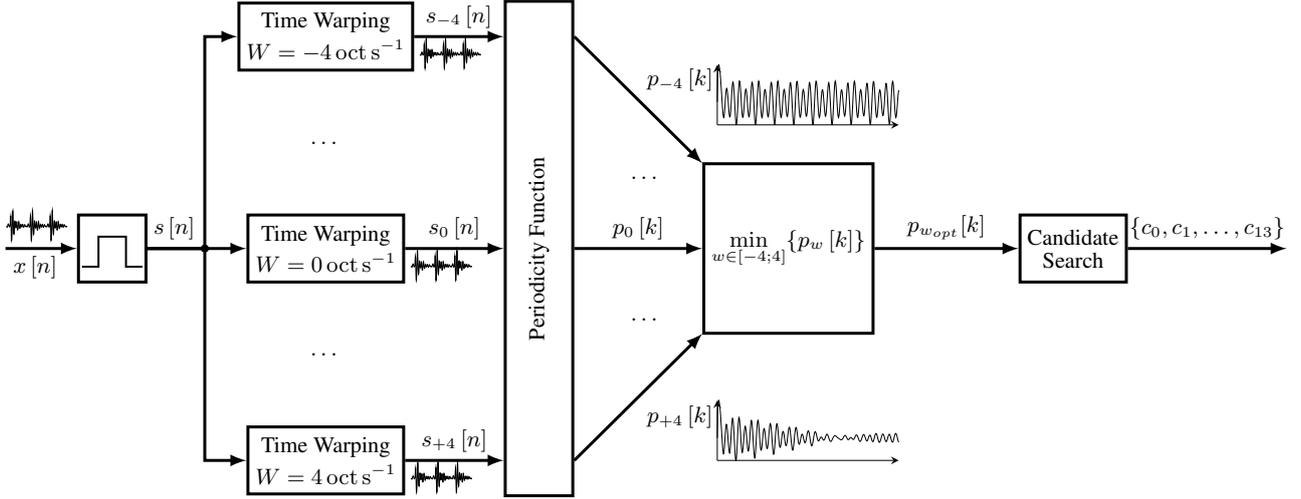


Figure 1: Calculation of f_0 candidates for one frame: The windowed signal is warped with nine different warping factors w . For each warped signal s_w , the periodicity function is calculated and the function with the global minimum determined. In this function, the frequencies corresponding to the first 14 local minima are identified as the potential f_0 candidates in this frame. These 14 candidates are calculated for every frame and the sequence of all candidate sets is passed to a Viterbi search to identify the final f_0 -contour over time.

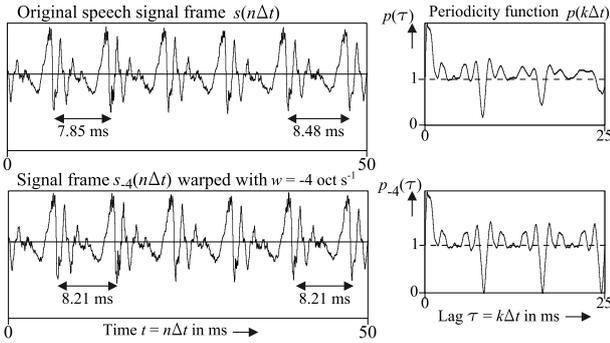


Figure 2: Example of the warping effect: A frame from the speech signal that contains a fast f_0 increase of 4 oct s^{-1} (top left) leads to measurable differences of the fundamental period at the beginning and the end of the frame. This corresponds to a hardly periodic periodicity function with a large first minimum (top right). If this frequency change is counteracted by time-warping with the matching factor $w = -4 \text{ oct s}^{-1}$, the fundamental period within the frame becomes constant (bottom left) and the periodicity function is truly periodic with minimum values very close to 0 (bottom right).

2.3. Candidate Extraction

Due to the influence of lower formant frequencies and other harmonic interferences, the lag corresponding to a global minimum of a PF is not necessarily the glottal period. Instead, the 14 smallest local minima in $p_{w_{opt}}[k]$ and their corresponding lags k_{c_i} ($i = 0, \dots, 13$) are extracted as a set of 14 voiced f_0 -candidates, each of which is a tuple $c_i = \{k_{c_i}, p_{w_{opt}}[k_{c_i}]\}$

2.4. F0 contour estimation using Dynamic Programming

Since the f_0 -contour across an utterance is unlikely to change abruptly, information from previous frames and following frames can be incorporated in the identification of the final f_0 es-

timate using a Viterbi search, analogously to PRAAT and RAPT: After analyzing every frame of the original speech signal, the result is a sequence of f_0 -candidate sets. Each of these sets is a column in a Dynamic Programming grid with the local costs calculated analogously to PRAAT and RAPT [11] (but modified to reflect the use of the PF instead of the ACF):

$$C_{l,v} = p_{w_{opt}}[k_{c_i}] + C_{oct} \cdot \log_2(\min(p_{w_{opt}}) \cdot k_{c_i}), \quad (9)$$

with the minimum value $p_{w_{opt}}[k_{c_i}]$, the global minimum $\min(p_{w_{opt}})$, the corresponding lag k_{c_i} of the i -th candidate, and the user-adjustable constant C_{oct} .

To allow a frame to be unvoiced, an additional unvoiced candidate is added with the local costs given by:

$$C_{l,u} = V_B + (1 - p_{w_{opt},min}), \quad (10)$$

where $\min(p_{w_{opt}}[k])$ is the global minimum of the PF and V_B is a user-defined bias term.

The transition costs were adopted from [11] without modifications:

$$C_t = \begin{cases} C_{oct,J} \left| \log_2 \left(\frac{k_{c_i}}{k_{c_{i-1}}} \right) \right| & \text{V-V} \\ C_{VU} & \text{V-U or U-V} \\ 0 & \text{U-U,} \end{cases} \quad (11)$$

where k_{c_i} and $k_{c_{i-1}}$ are the lags of the current and the previous candidate and $C_{oct,J}$ and C_{VU} are used-defined constants.

All user-adjustable parameters of the TPTA are summarized in Table 2. The default values for the octave jump cost $C_{oct,J}$ and the bias for high frequencies C_{oct} was determined empirically by minimizing the octave errors in a set of 10 test sentences. The voicing bias V_B and the voiced-unvoiced transition cost C_{VU} were found using an exhaustive search to find the optimal pair (V_B, C_{VU}) that minimizes the relative f_0 estimation error in a subset of the evaluation data (see below) consisting of 100 randomly selected sentences. The search interval was $[-0.5, 0.1] \times [0.1, 0.6]$ with a step size of 0.01.

Table 1: Results of PRAAT, RAPT, and TPTA from 4718 sentences of read speech by 10 male and 10 female speakers.

SNR dB	With white noise												With babble noise											
	PRAAT				RAPT				TPTA				PRAAT				RAPT				TPTA			
	GPE %	FPE cents	VDE %	FFE %	GPE %	FPE cents	VDE %	FFE %	GPE %	FPE cents	VDE %	FFE %	GPE %	FPE cents	VDE %	FFE %	GPE %	FPE cents	VDE %	FFE %	GPE %	FPE cents	VDE %	FFE %
clean	6.73	42.29	12.82	14.25	3.22	62.60	6.10	6.74	7.97	51.57	5.67	7.34	6.73	42.29	12.82	14.25	3.22	62.60	6.10	6.74	7.97	51.57	5.67	7.34
20 dB	6.93	42.81	13.07	14.56	2.94	60.95	4.94	5.12	8.02	49.35	4.65	6.32	7.10	43.04	13.36	14.88	3.26	62.47	10.99	11.64	8.20	51.31	5.34	7.06
15 dB	7.26	43.96	20.96	22.54	2.77	59.47	4.78	5.32	7.88	47.76	4.63	6.24	8.18	45.26	21.18	22.97	3.24	62.72	11.32	11.96	8.08	51.44	5.40	7.04
10 dB	8.88	47.71	64.91	66.96	2.54	56.95	4.96	5.44	7.00	45.59	5.13	6.49	11.85	50.74	52.72	55.44	3.46	63.69	11.61	12.30	8.00	51.84	5.80	7.39
5 dB	10.88	50.47	76.91	79.48	2.28	53.14	6.03	6.42	4.94	43.14	7.27	8.10	18.51	58.85	74.41	78.77	4.55	65.62	12.67	13.55	7.78	53.32	7.52	8.93
0 dB	14.95	54.43	77.08	80.62	1.94	48.81	9.48	9.74	2.40	42.16	13.59	13.81	29.15	70.90	77.01	83.88	9.61	70.63	15.04	16.76	8.55	55.62	12.37	13.50

Table 2: Parameters of the TPTA and their default values.

Parameter	Value
Length of frame	L 0.05 s
Minimum pitch	f_{\min} 50 Hz
Maximum pitch	f_{\max} 500 Hz
Time step	Δt 0.01 Hz
Max. changing speed	w_{\max} $\pm 4 \text{ oct s}^{-1}$
Bias for voiced decision	V_B -0.29
Number of f_0 candidates	N_{cand} 14
Favouring high or low f_0 values	C_{oct} 0.05
Octave jump cost	$C_{\text{oct}J}$ 0.4
Transition cost for changed voicing state	C_{VU} 0.42

3. Evaluation

To evaluate the performance of the TPTA, a corpus of speech data was analyzed using TPTA and the two most cited reference algorithms (according to [6]) RAPT and PRAAT.

3.1. Speech Data

The speech data used in the evaluation was taken from the Pitch Tracking Database provided freely by the Graz University of Technology [16], which is one of the largest freely available corpora with microphone and laryngograph data. It consists of 4720 sentences from 20 speakers (10 male and 10 female) of read speech recorded in an audio recording studio. Two sentences (by speaker F07) were not included in the archive obtained from <https://www.spssc.tugraz.at/tools/ptdb-tug> (accessed: March 10, 2017) and thus not part of the evaluated speech material. For every sentence, a high-quality audio recording (48 kHz, 16 Bit WAV) was available, accompanied by a text file containing the ground truth f_0 at a sampling rate of 100 Hz, which was derived from the corresponding laryngograph signal. To analyze a more realistic scenario than the high-quality recordings suggest, white and babble noise from [17] with Signal-to-Noise-Ratios (SNR) of 0 dB, 5 dB, 10 dB, 15 dB and 20 dB were added to the speech data.

3.2. Algorithm setups

RAPT [8] was used in its implementation in the command line tool Snack [18] and was run with default settings except for the frequency range, which was set to 50 Hz to 500 Hz. PRAAT was also used with default settings except for the frequency range (which was also set to 50 Hz to 500 Hz) and the step size, which was set to 0.01 s to align with the ground truth sampling rate. TPTA was used with the settings shown in Table 2.

3.3. Analysis

The same error measures as in [6] were used here to evaluate the algorithms:

- Gross Pitch Error (GPE): Ratio of the number of voiced frames with a relative error higher than 20% of the ground truth to the number of all frames identified as voiced by both the pitch tracker and the ground truth.
- Fine Pitch Error (FPE): Standard deviation of the difference between the estimated f_0 in correctly recognized voiced frames and the true f_0 .
- Voicing Decision Error (VDE): Ratio of the number of frames with wrong voicing-state decisions to the total number of frames.
- f_0 Frame Error (FFE): Ratio of the number of all erroneous frames to the total number of frames.

3.4. Results

The results from all three algorithms (PRAAT, RAPT, TPTA) and for all SNR with both kinds of noise are shown in Table 1. Both RAPT and TPTA achieved better total accuracy in all settings in terms of GPE, VDE, and FFE compared to PRAAT. RAPT was still slightly superior to TPTA for clean speech, because it more accurately estimated the f_0 in frames that it recognized as voiced. However, TPTA made less voicing decision errors. When white noise was added to the speech, the VDE of RAPT was better for an SNR of 10 dB or lower and the FFE stayed below the FFE of TPTA. When babble noise was added, however, the VDE of RAPT drastically deteriorated with decreasing SNR and resulted in an overall better FFE of TPTA for even mildly noisy speech (SNR of 20 dB).

4. Conclusions and Outlook

Compared to the state-of-the-art, TPTA yielded similar results as RAPT and was superior to PRAAT, if the starting parameters are kept mostly generic (i.e., no very narrow f_0 band adapted to the speaker's gender). The warping of the time axis as implemented by TPTA apparently had a significant positive impact on the VDE, which in turn led to good results even in very noisy conditions, where both RAPT and PRAAT were outperformed.

Currently, TPTA needs audio files in the WAV-format with an arbitrary sampling rate and a length of at least two times the integration length, i.e. 50 ms. The audio file should contain only one speaker because multi-pitch tracking is currently not possible. TPTA is currently implemented as a MATLAB script and will be ported to a C++ implementation for future works to improve the calculation time. The candidate search in TPTA should also be improved to lower TPTA's GPE to get closer to the results with RAPT, especially for clean speech. Another improvement could be to use more finely quantized time-warping factors w to better approximate the actual f_0 change within a frame and thus potentially further lower the GPE.

5. References

- [1] W. Hess, *Pitch determination of speech signals: Algorithms and devices*. Berlin Heidelberg, Germany: Springer-Verlag, 1983, vol. 1.
- [2] P. Taylor, *Text-to-Speech Synthesis*. Cambridge, UK: Cambridge University Press, 2009.
- [3] I. Siegert, K. Hartmann, D. Philippou-Hübner, and A. Wendemuth, "Human behaviour in HCI: Complex emotion detection through sparse speech features," in *Proc. of the 4th International Workshop on Human Behavior Understanding*, vol. 8212. New York, NY, USA: Springer-Verlag New York, Inc., 2013, pp. 246–257.
- [4] M. Akagi, X. Han, R. Elbarougy, Y. Hamada, and J. Li, "Toward affective speech-to-speech translation: Strategy for emotional speech recognition and synthesis in multiple languages," in *Asia-Pacific Signal and Information Processing Association, 2014 Annual Summit and Conference (APSIPA)*. IEEE, 2014, pp. 1–10.
- [5] M. Bashar, M. T. Ahmed, M. Syduzzaman, P. J. Ray, and A. T. Islam, "Text-Independent Speaker Identification System Using Average Pitch and Formant Analysis," *International Journal on Information Theory (IJIT)*, vol. 3, no. 3, pp. 23–30, 2014.
- [6] S. Strömbergsson, "Today's most frequently used F0 estimation methods, and their accuracy in estimating male and female pitch in clean speech," in *Proc. of Interspeech*, 2016, pp. 525–529. [Online]. Available: <http://dx.doi.org/10.21437/Interspeech.2016-240>
- [7] P. Boersma, "Praat, a system for doing phonetics by computer," *Glot International*, vol. 5, pp. 341–345, 2002.
- [8] D. Talkin, "A robust algorithm for pitch tracking (RAPT)," *Speech coding and synthesis*, vol. 495, p. 518, 1995.
- [9] H. Kawahara, J. Estill, and O. Fujimura, "Aperiodicity extraction and control using mixed mode excitation and group delay manipulation for a high quality speech analysis, modification and synthesis system STRAIGHT," in *Proc. of the Second International Workshop on Models and Analysis of Vocal Emissions for Biomedical Applications (MAVEBA)*, 2001, pp. 59–64.
- [10] A. De Cheveigné and H. Kawahara, "YIN, a fundamental frequency estimator for speech and music," *The Journal of the Acoustical Society of America*, vol. 111, no. 4, pp. 1917–1930, 2002.
- [11] P. Boersma, "Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound," in *Proc. of the Institute of Phonetic Sciences*, vol. 17, no. 1193, Amsterdam, 1993, pp. 97–110.
- [12] J. Sundberg, "Maximum speed of pitch changes in singers and untrained subjects," *Journal of Phonetics*, vol. 7, no. 2, pp. 71–79, 1979.
- [13] Y. Xu and X. Sun, "How fast can we really change pitch? maximum speed of pitch change revisited," in *Proc. of the Interspeech*, 2000, pp. 666–669.
- [14] —, "Maximum speed of pitch change and how it may relate to speech," *The Journal of the Acoustical Society of America*, vol. 111, no. 3, pp. 1399–1413, 2002.
- [15] P. Birkholz, "Grundfrequenzbestimmung unter Berücksichtigung linearer Frequenzveränderungen," *Fortschritte der Akustik*, vol. 29, pp. 768–769, 2003.
- [16] G. Pirker, M. Wohlmayr, S. Petrik, and F. Pernkopf, "A pitch tracking corpus with evaluation on multipitch tracking scenario," in *Proc. of the Interspeech*, 2011, pp. 1509–1512.
- [17] S. P. S. (SPS). (1993) Signal processing information base (spib). [Online]. Available: spib.linse.ufscr.br/index.html
- [18] K. Sjölander and J. Beskow, "Wavesurfer—an open source speech tool," in *Proc. of the Interspeech*, 2000, pp. 464–467.