Performance analysis of various fundamental frequency estimation algorithms in the context of pathological speech

Robin Vaysse, Corine Astésano, Jérôme Farinas

JASA ARTICLE

Performance analysis of various fundamental frequency estimation algorithms in the context of pathological speech

Robin Vaysse,[1,a),b)] Corine Astésano,[2,c)] and Jérôme Farinas[1]
[1]IRIT, Université de Toulouse, CNRS, Toulouse INP, UT3, Toulouse, France
[2]Laboratoire de NeuroPsychoLinguistique, Université Toulouse Jean-Jaurès, France

ABSTRACT:
Reliable fundamental frequency (f_0) extraction algorithms are crucial in many fields of speech research. The current bulk of studies testing the robustness of different algorithms have focused on healthy speech and/or measurements of sustained vowels. Few studies have tested f_0 estimations in the context of pathological speech, and even fewer on continuous speech. The present study evaluated 12 available pitch detection algorithms on a corpus of read speech by 24 speakers (8 healthy speakers, 8 speakers with Parkinson's disease, and 8 with head and neck cancer). Two fusion methods' algorithms have been tested: one based on the median of algorithms and one based on the fusion between the best algorithm for voicing detection and the algorithm that generates the most accurate f_0 estimations on voiced parts. Our results show that time-domain algorithms, like REAPER, are best for voicing detection while deep neural network algorithms, like FCN-f_0, yield better accuracy for the f_0 values on voiced parts. The combination of REAPER and FCN-f_0 yields the best ratio performance/implementation complexity, since it generates less than 4% errors on voicing detection and less than 5% of gross errors in the estimation of the f_0 values for all speaker groups.
 https://doi.org/10.1121/10.0015143

(Received 2 August 2022; revised 29 September 2022; accepted 26 October 2022; published online 29 November 2022)
[Editor: B. Yegnanarayana] Pages: 3091–3101

I. INTRODUCTION

The measurement of the fundamental frequency (f_0) is an essential element of automatic speech processing, particularly in the study of prosody. It is, therefore, crucial to have a good estimate of this parameter. Many algorithms have been developed for estimating the fundamental frequency of healthy speech recorded under good conditions (without noise), which provide very good f_0 approximations (see Sec. II). In the context of pathological speech, the calculation of precise f_0 variations is necessary because most pathologies have an impact on voice quality, more specifically on speakers' inability to maintain a stable fundamental frequency (jitter, shimmer) (Jiménez-Jiménez *et al.*, 1997). In addition, the dynamics of the fundamental frequency in a sentence defines the intonation, which corresponds to the voice "melody." Intonation provides main communicative functions and is a powerful tool for the illocutionary and structural interpretation of the speaker's message (Di Cristo, 2016). Yet, some pathologies can lead to a poor control of intonation that can induce confusion as to the type of sentence the speaker is trying to produce (Le Dorze *et al.*, 1994), which affects both his/her intelligibility and comprehensibility. When we want to model intonation or stress patterns from the f_0, these types of errors can lead to distorted interpretations over large time spans.

It is, therefore, crucial to use an f_0 extraction algorithm, which is as accurate as possible and avoids gross estimation errors (such as dividing by two or doubling the real value of the fundamental frequency) or errors in the detection of voiced or unvoiced areas. When working on large corpora of pathological voice recordings, such as Cesari *et al.* (2018), this issue is even more challenging because the amount of data does not allow for precise manual annotations. The objective of the present study is, therefore, to test several different algorithms in the particular context of pathological voice, such as those resulting from head and neck cancers (H&NC) or Parkinson's disease (PD).

Several performance evaluation studies of pitch detection algorithms have been designed on non-pathological speech (de Cheveigné and Kawahara, 2001; Strömbergsson, 2016) and it seems that auto-correlation function (ACF) from Praat (Boersma and Weenink, 2020) and the YIN algorithms (de Cheveigné and Kawahara, 2002) are good methods for typical, healthy voices. Some studies also looked into the evaluation of noisy speech, which best corresponds to real recording conditions (Jouvet and Laprie, 2017; Luengo *et al.*, 2007). These results show that, while all the evaluated algorithms provide comparable results on healthy speech, an increase in background noise results in a loss of algorithm performance, specifically with regard to the detection of voicing. More specifically, the robust algorithm for pitch tracking (RAPT) and the robust epoch pitch estimator (REAPER) algorithms seem to provide good results on

[a)]Also at: Laboratoire de NeuroPsychoLinguistique, Université Toulouse Jean-Jaurès, France
[b)]Electronic mail: robin.vaysse@irit.fr
[c)]Also at: UMR 5267 Praxiling - Université Paul Valéry Montpellier, France

noisy data while the ACF algorithm does not provide good results on noisy speech. Indeed, according to Jouvet and Laprie (2017), ACF generates an error rate of 4.6% for voicing detection on noise-free speech, but this rate increases to 16.2% with the addition of noise with a signal to noise ratio (SNR) level of 10 db, while the REAPER error rate only increases from 5% to 8.3% with the same SNR level. Pathological speech was evaluated marginally in some studies, such as Parsa and Jamieson (1999). The authors compared seven pitch detection algorithms on sustained vowels of pathological speech where the patients showed benign vocal lesions, such as polyps, nodules, and cysts. They showed that among the compared algorithms, the ACF and average magnitude difference function (AMDF) algorithms were good fits for pathological voices. Later on, Jang *et al.* (2007) also compared seven pitch detection algorithms on sustained vowels. This last study concluded that the ACF (Boersma and Weenink, 2020) performed best on their dataset. Tsanas *et al.* (2014) compared 10 pitch detection algorithms on a sustained vowel task and showed that the sawtooth waveform inspired pitch estimator (SWIPE) and the nearly defect-free (NDF) algorithms provide the best f_0 estimates on their dataset. They also proposed a new combination algorithm based on Kalman filters that was 16% more accurate than the best algorithm tested. According to this brief review of literature, the next step is to test pitch detection algorithms in connected pathological speech, which has, to our knowledge, never been addressed. The present study tackles this issue on two different pathologies: H&NC and PD. These pathologies have quite different impacts on f_0: H&NC present a variety of f_0 alterations (e.g., hoarseness, dysfluences interrupting coherent intonation groups) while PD does not so much impact the global linguistic features of f_0 but rather the dynamics of f_0 variations. These two pathologies, thus, allow for complementary insight on f_0 detection algorithms. Furthermore, a comparison between the classical algorithms of pitch detection and the new emerging methods (Ardaillon and Roebel, 2019; Kim *et al.*, 2018) based on deep neural networks could be interesting.

Closer to our present goals, the study by Jouvet and Laprie (2017) using speech in noise was of particular interest to help us determine which algorithms to test on our pathological speech corpora. Specifically, algorithms whose performances were highest were chosen, and they have been sorted according to their differences of implementation (spectral vs time domain; post- vs pre-processing). In addition to these algorithms, three additional algorithms have been integrated: pitch estimation filter with amplitude compression (PEFAC), an algorithm that is in the spectral domain with or without pre- or post-processing (Gonzalez and Brookes, 2014), that was designed to perform on noisy speech; convolutional representation for pitch estimation (CREPE) (Kim *et al.*, 2018), which uses a pre-trained convolutional neural network; and fully convolutional networks for f_0 detection (FCN- f_0) (Ardaillon and Roebel, 2019), which is a fully-convolutional neural network that has been trained to optimize both f_0 computation and voicing detection while the other deep neural network algorithm, CREPE, has been optimized for f_0 estimation only. Section II presents the different fundamental frequency detection algorithms that have been selected for this study. Section III describes the voice recordings and the method used to extract the real values of f_0 and also the evaluation metrics used to evaluate the performances of the algorithms. Finally, Sec. IV presents our results and proposes leads to understand the differences observed between healthy and pathological speech with the various tested algorithms.

II. FUNDAMENTAL FREQUENCY DETECTION ALGORITHMS

Speech researchers need reliable fundamental frequency detection programs and have a wide variety of choice (see Sec. II A). In the case of pathological speech research, it is much trickier to decide which f_0 algorithm is best suited for degraded voice quality. This section describes our selection process leading to the choice of the f_0 algorithms that will be used in our testing section. The programs were selected primarily on the basis of their availability and ease of access, to fit the ecological situation of most speech researchers interested in f_0 detection. Our selection was also motivated by the need to cover a large variety of algorithms based on temporal and frequency representations, and those based on deep learning methods. It is commonly acknowledged that most f_0 extraction algorithms can be decomposed in three steps:

- First, a pre-processing of the signal can be applied to remove unnecessary information. For example, yet another algorithm for pitch tracking (YAAPT) (Kasi and Zahorian, 2002) applies a bandpass filter between 100 Hz and 900 Hz to the signal, while some algorithms apply a low-pass filter on the raw signal [e.g., the YIN algorithm (de Cheveigné and Kawahara, 2001) or the REAPER algorithm from Google-Open-Source (2015)].
- Then, candidates' values of f_0 are extracted using, for example, temporal or spectral representations of the signal.
- Finally, a step of post-processing takes the pitch candidates and chooses those more likely to be good f_0 estimations. For example, CREPE (Kim *et al.*, 2018) applies a Viterbi smoothing to remove isolated or incoherent values (sudden drops or increases of f_0). Some algorithms also use dynamic programming (Bellman, 1954) like REAPER, RAPT (Ghahremani *et al.*, 2014), or YAAPT (Kasi and Zahorian, 2002), allowing sudden jumps in the final f_0 curve to be minimized.

Aside from these common prerequisites and even though their global architectures are generally similar, pitch detection algorithms implementations otherwise differ in many ways. We purposely used these programs with default parameters, as researchers commonly do when first using such algorithms. Section II A will present the different approaches underlying the f_0 extraction algorithms.

https://doi.org/10.1121/10.0015143

A. Pitch detection algorithms typology

The f_0 information can be retrieved using time-domain representations of signal or frequency domain representations. Time-domain algorithms generally use the autocorrelation algorithm which consists of extracting a signal window of a few milliseconds and searching whether the detected pattern is repeated in successive signal windows. This method computes the correlation between two windowed signals. The resulting f_0 value will consist of the delay τ between windows that present the best correlation. Some of them use a slightly different method called cross correlation which computes the correlation between the signal and a modified version of itself like a downsampled version of the signal.

The following methods use time-domain algorithms: ACF (Boersma, 2000), YIN (de Cheveigne and Kawahara, 2002), AMDF (Ross *et al.*, 1974) and REAPER are based on the autocorrelation algorithm, while RAPT (Talkin and Kleijn, 1995) and enhanced RAPT (Ghahremani *et al.*, 2014) use cross correlation to extract pitch candidates.

As far as frequency domain algorithms are concerned, the f_0 values are selected by looking at the occurrence of the f_0 harmonics in the spectrum. In the present study, the following algorithms were chosen: PEFAC (Gonzalez and Brookes, 2014) and SWIPE (Camacho and Harris, 2008), which are known to be robust even for noisy speech signals (Jouvet and Laprie, 2017).

Some algorithms can also use information from time and frequency domains to refine their selection of pitch values, such as NDF (Kawahara *et al.*, 2005) and YAAPT (Kasi and Zahorian, 2002) for which the combination allows selection of the most likely pitch candidates.

Finally, new kinds of algorithms use deep neural networks like CREPE (Kim *et al.*, 2018) or FCN-f_0 (Ardaillon and Roebel, 2019). These methods rely on machine learning techniques, where the algorithm is trained to compute f_0 values from a raw signal with no explicit procedure. It is, thus, difficult to know what kind of information from the signal those algorithms use and whether it works in the time domain, the frequency domain, or both. These methods, however, provide robust results. The list of algorithms selected for the present study is given in Table I.

B. Algorithm merging

In addition to the above algorithms, we decided to integrate techniques based on combinations of different algorithms to test whether these combinations can reduce common errors. Indeed, autocorrelation-based algorithms tend to produce halving f_0 errors (estimated f_0 two times smaller than the real value) while methods based on frequency domain produce more doubling errors (estimated f_0 two times bigger than the real value). It is, thus, interesting to mix time and frequency domain algorithms to compensate for their respective common errors. An example of previous mixing f_0 algorithms techniques can be found in Espesser (1999) with the toolkit MES-SignAix (Espesser, 1996), which used a majority vote between different algorithms.

Tsanas *et al.* (2014) also used a median vote between 10 different pitch f_0 algorithms as a baseline for algorithm combinations. They found no improvement over the NDF algorithm alone on the raw accuracy of f_0 estimation on a sustained vowel task. However, we believe that using a simple median filtering on top of complementary algorithms (which produce different kinds of f_0 estimations errors) can improve the reliability of f_0 measurements. They also used a method based on a Kalman filter to merge different algorithm results. This latter method provides promising results on their sustained vowels dataset. Unfortunately, the codebase of their algorithm is not publicly available.

With this in mind, two simple methods for merging algorithms were selected. The first method is a "majority vote" obtained by using the median between the different values of several selected algorithms (Hess, 2008; Soquet, 1994; Espesser, 1999). The choice of the median allows us to eliminate gross f_0 errors. At least three different algorithms were computed at a time: because each algorithm generates different results on the same file, the median of these values was used as the f_0 estimate. A 10 ms frame was chosen to comply with the pseudo-stationary property of the

TABLE I. List of algorithms tested in the present study, with a link to the chosen implementation. The last three columns indicate whether the algorithm works on the signal's time or spectral domain, or whether it uses deep learning.

Algorithm	Implementation	Time domain	Spectral	Neural network
ACF (Boersma, 2000)	Praat	X		
AMDF (Ross et al., 1974)	Snack Sound toolkit (Kåre, 2005)	X		
REAPER (Google-Open-Source, 2015)	https://github.com/google/REAPER	X		
RAPT (Talkin and Kleijn, 1995)	Snack Sound toolkit (Kåre, 2005)	X		
Enhanced RAPT (Ghahremani *et al.*, 2014)	Kaldi (Povey *et al.*, 2011)	X		
Yin (de Cheveigne and Kawahara, 2002)	https://github.com/patrice.guyot/Yin	X		
NDF (Kawahara et al., 2005)	STRAIGHT (Kawahara, 2018)	X	X	
YAAPT (Kasi and Zahorian, 2002)	MATLAB implementation (Zahorian and Hu, 2016)	X	X	
SWIPE (Camacho and Harris, 2008)	Speech signal processing toolkit (Tokuda *et al.*, 2017)		X	
PEFAC (Gonzalez and Brookes, 2014)	VOICEBOX (Brookes, 2018)		X	
CREPE (Kim et al., 2018)	https://github.com/marl/crepe			X
FCN-f_0 (Ardaillon and Roebel, 2019)	https://github.com/ardaillon/FCN-f0			X

TABLE II. Illustration of the merging of algorithms by the resulting f_0 median vote: The first column indicates the start time of the 10 ms frame, the next 3 columns are the estimated values by an example of 3 different algorithms, and the last column with boldface values is the f_0 median voting.

Time (s)	f_0 Yin	f_0 ACF	f_0 *SWIPE*	f_0 *Median*
0	0	140	0	**0**
0.01	0	189	181	**181**
0.02	170	173	169	**170**
—	—	—	—	—

speech signal (Hess, 2008). An illustration of the method is described in Table II.

The second merging method consists of the actual fusion/combination of two algorithms, by taking an algorithm (Algorithm A) that is particularly efficient on the calculation of the f_0 value and another algorithm that is very accurate on the voicing detection (Algorithm B). Algorithm B is then used to select the voiced time windows and Algorithm A gives the estimated f_0 values in those windows. Table III describes the method used.

III. EXPERIMENTAL SETUP

A. Recordings description

This work is part of the ANR project RUGBI 2018-2023 (RUGBI, 2018-2023) in which the main goal is to find specific speech markers that impact speakers' intelligibility. Two pathological speech corpora from this project were used.

The first corpus is taken from the Carcinologic Speech Severity Index (C2SI) project (see Acknowledgements), in which 127 speakers were recorded, consisting of 40 control subjects and 87 patients who had been treated for cancer of the oral cavity or pharynx. Speakers were recorded in several tasks (such as sustained /a/, non-words reading, short-text reading, picture description, prosodic functions encoding) For a complete description of the corpus, see Woisard *et al.* (2021).

Our second source of pathological speech is the Aix Hospital Neuro (AHN) corpus, with 209 recordings of PD patients (112 controls) described in Ghio *et al.* (2012). A subset of this corpus is used in the RUGBI project. Speakers also recorded several tasks, such as those described in the C2SI corpus plus additional tasks specifically designed for this pathology (e.g., diadochokinesis, singing, breathing). Both corpuses used the same short-text reading (Daudet, 1870), which was used in the present study to test the f_0 algorithms on connected speech. This task has been chosen to have consistent recordings between the different speakers (the first four sentences of the text are common to both corpora), but also because the recordings are relatively long (ranging from 20 to 70 s). A subset of 24 speakers composed of 8 healthy patients, 8 patients with H&NC, and 8 patients with PD (4 men and 4 women in each group) has been selected.

TABLE III. Illustration of the merging of algorithms by fusion/combination. The last column with bold values represents the final f_0 estimation based on the two previous columns. The voicing detection is based on A (if there is a 0 in A then it is reported on the resulting combination value) and the estimated f_0 values are taken from the Algorithm B.

Time (s)	Algorithm B	Algorithm A	*f0* Combination
0	0	140	**0**
0.01	173	189	**189**
0.02	170	180	**180**
0.03	0	0	**0**
—	—	—	—

The selection of patient files was based on perceptual analysis conducted by specialized clinicians to assess the quality of patients' voices on the reading task. Speakers with the most degraded voice quality were selected to test the algorithms under the most difficult conditions. In total, the corpus is composed of 120 sentences (40 per group) and represents roughly 13 min of recordings.

B. Manual f_0 annotations

1. Manual correction of period detection errors

Since the files were not recorded using an Electro-Glotto-Graph, it was necessary to fully annotate the f_0 manually. To do so, the Praat software (Boersma and Weenink, 2020) was used to perform a first automatic annotation using an algorithm based on the autocorrelation of the signal (Boersma, 2000). This annotation was then manually corrected at the signal level by indicating the boundaries of each pattern to obtain a fundamental frequency corresponding to the real value (illustration on Fig. 1). Once corrected, the fundamental frequency curve was extracted with values every 10 ms (Fig. 2).

Our resulting manual annotated f_0 constitutes our reference (gold standard) and was then compared to the outputs of all the algorithms described in Sec. II A.

2. Simultaneous f_0 zones (diplophonia)

When analyzing pathological speech, we encounter numerous instances where the fundamental frequency seems to drop abruptly, with periodic patterns that double in length in the signal. An example is shown in Fig. 3.

The length of the patterns at the beginning of the signal corresponds to a frequency of about 126 Hz; a new periodic pattern suddenly appears, which is much longer (66 Hz) and disappears after a few milliseconds. This phenomenon leads to simultaneous frequencies, one quite low and another usually an octave higher. From a perceptual point of view, this results in a hoarse voice and an indeterminate fundamental frequency (Keating *et al.*, 2015). Further studies need to be run to better encompass this phenomenon, but they go beyond the scope of our paper. These complex areas have been annotated to observe how the algorithms behave on these segments. In these zones, the annotation of the f_0 value was performed using a linear interpolation between the previous (stable) f_0 values and the following values. This choice is justified by the fact that these segments are relatively

https://doi.org/10.1121/10.0015143

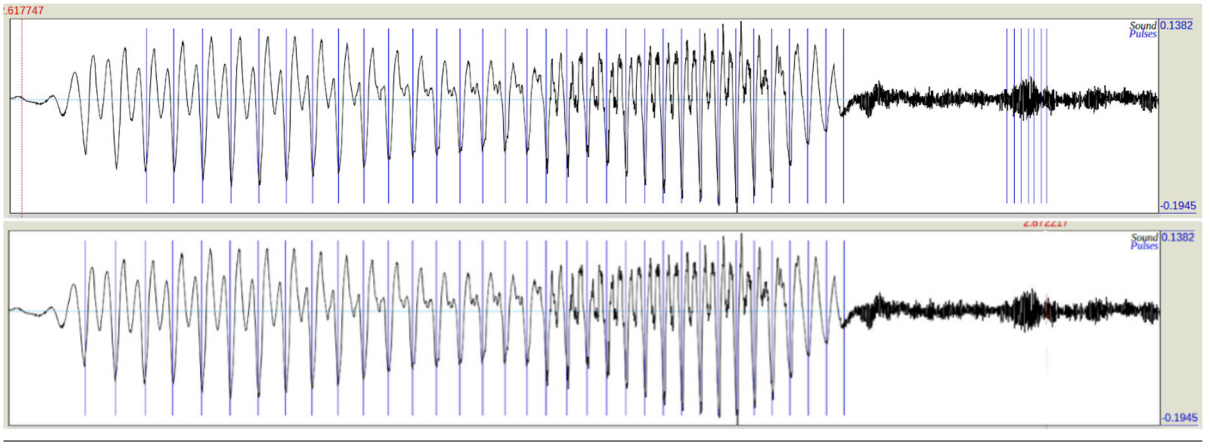

FIG. 1. (Color online) Example of annotation: the automatic marking of periodic pattern boundaries from Praat is at the top and the manually corrected annotation is at the bottom of the figure. Note that the first two periods on the left were not detected as voiced and an unvoiced segment at the end was detected as periodic.

short (less than 100 ms on average) and are, thus, hardly perceived as sudden f_0 drops. These complex periodic areas are actually quite rare. In total, in this corpus, simultaneous frequencies represent:

- 0.9% of voiced segments for healthy speech
- 4.5% of voiced segments for cancer patients (with a maximum of 13% for one speaker)
- 1.5% of voiced segments for PD patients.

C. Metrics

To have an objective evaluation of the quality of the different algorithms, three metrics and four sub-metrics classically used were computed to exhaustively evaluate the computation of f_0 (Jouvet and Laprie, 2017; Drugman and Alwan, 2011; Babacan *et al.*, 2013; Chu and Alwan, 2009). These metrics allow us to describe the different types of errors produced by the algorithms, whether they are voicing detection errors or errors in the estimation of the real value of f_0:

- Voicing detection error (VDE), which measures the 10 ms frame proportion containing errors in the detection of voicing; two sub-metrics for voicing detection were also added
- False negative rate (FNR), which computes the proportion of voiced frame detected as unvoiced by the algorithm
- False positive rate (FPR), which computes the proportion of unvoiced frame detected as voiced by the algorithm
- Gross pitch error (GPE), which measures the proportion of frames where the estimated value differs from the real value by more than 20%
- The proportion of frames where the estimated value is at least 20% higher than the reference value ($\times 2$)
- The proportion of frames where the estimated value is at least 20% lower than the reference value ($\div 2$)

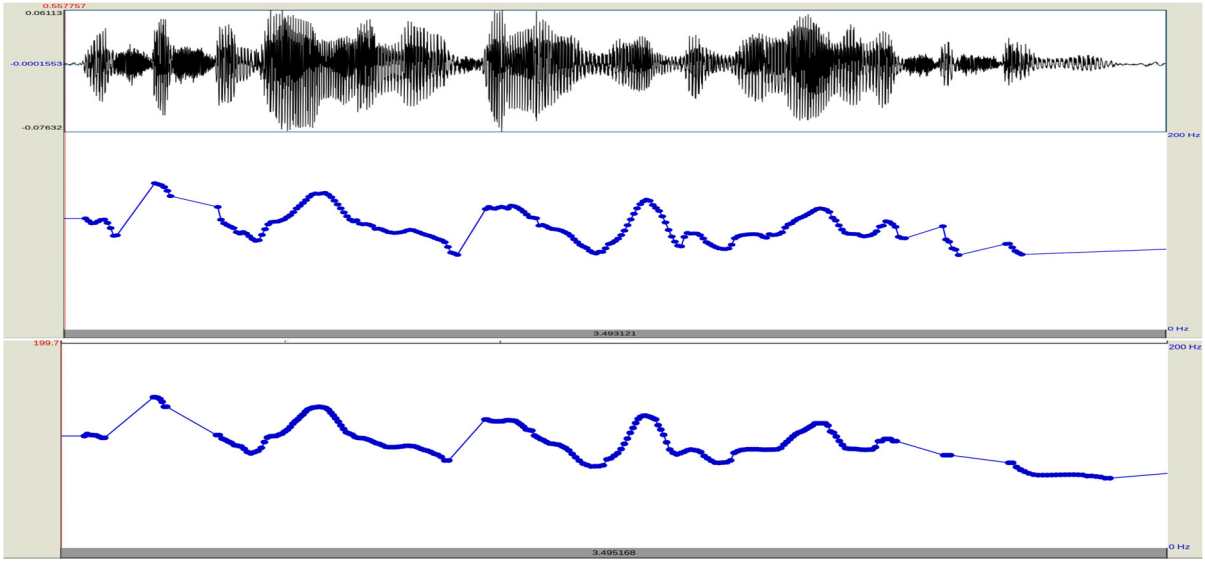

FIG. 2. (Color online) Example of file annotation from a healthy speaker on the sentence, "*Monsieur Seguin n'avait jamais eu de bonheur avec ses chèvres.*" Automatic f_0 from Praat is at the top and the manually corrected annotation is at the bottom. Each blue point corresponds to a f_0 value for a 10 .ms frame.

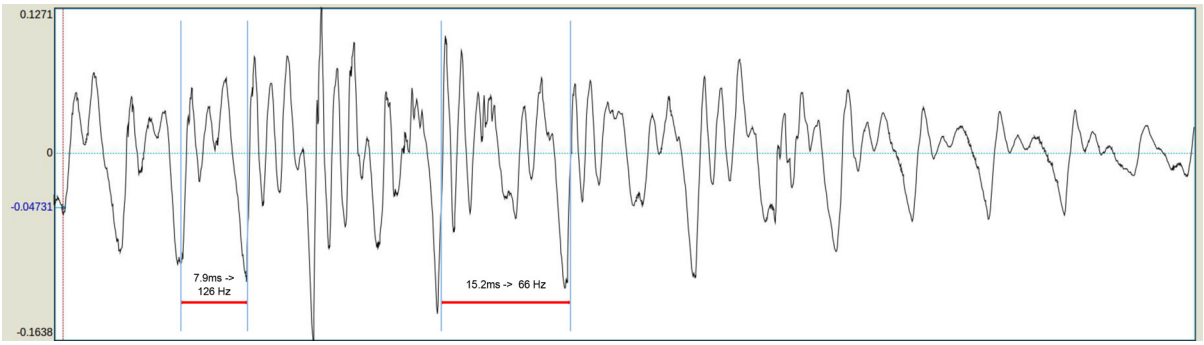

FIG. 3. (Color online) Example where the fundamental frequency seems to decrease suddenly by an octave.

- FFE (f_0 frame error), which measures the frame proportion where an error was detected, whether it is a voicing detection error or a gross pitch error.

D. Experimental protocol

Each of the 12 algorithms (see Table I) were then executed on all audio files, respecting the recommended sampling frequencies for each algorithm and using the default settings of the different implementations. Unfortunately, some algorithms are quite sensitive to parameters like pitch range, silence threshold and it is necessary to adjust them for each speaker. However, we purposely chose not to modify them, to evaluate which algorithms best adapt to the data quickly. The f_0 estimate was calculated on 10 ms windows (Hess, 2008), which corresponds to the generated annotations (see Sec. III B). Some algorithms generate a voicing probability score, which makes it possible to determine whether the analyzed window contains a fundamental frequency value. It is then possible to test different thresholds to determine whether the portion of the signal is voiced. For example, one can consider that if the probability is less than 80%, then the estimate of the f_0 is set to 0. To determine an optimal threshold, for each algorithm, the threshold that minimizes the VDE metric (see Sec. III C) was chosen.

If the algorithm used does not generate a voicing probability, then the unvoiced sections are set to 0 by the algorithm and the metrics are computed directly.

IV. RESULTS

We now present the results obtained with the different algorithms described in Table I, with the manual annotations described in Sec. III B as a reference (gold standard). In addition to these algorithms, the median vote described in 2.2 was computed on 5 methods: AMDF, Kaldi, NDF, FCN-f_0, and REAPER. The median vote was applied to these 5 methods because they give the best vote results, and they are based on various calculation methods. The results of the median vote will be referred to as "median" in Figs. 4–6.

Finally, the combination of two algorithms was also computed using REAPER as a basis for the detection of voicing, and the values estimation was given by the FCN-f_0 method. This choice is based on the fact that REAPER generates the best overall results concerning the voicing detection and FCN-f_0 provides the most accurate estimates of f_0 on our dataset. The results of the combination of those two algorithms will be referred to as ⟨⟨ Combi ⟩⟩ in the following figures.

First, the global results for the VDE were analyzed, then GPE, and the final step consists in analyzing mixed errors (FFE metric). These tests were done after excluding speech areas with simultaneous fundamental frequencies (Sec. III B 2). All the detailed results are included in the Sec. V presented in Table IV.

A. Voicing detection

Figure 4 shows the results obtained for different algorithms, on the metric VDE. The abscissa shows the different algorithms that have been tested, while the proportion (between 0 and 1) of analysis frame windows (10 ms) with a VDE is shown on the ordinate. The figure compares the results from the "Healthy group", the group with H&NC, and the group with PD.

It seems that algorithms based on the signal time domain give the best results for voicing detection. The RAPT, ACF, and AMDF algorithms give similar results with about 5% of windows containing a voicing error on the voices of speakers with H&NC and PD, and about 4% for control speakers. REAPER algorithm seems to be the best one for patients with H&NC and PD with 3.5% of errors on speech of patients with H&NC cancer, 3.3% on speech of patients with PD, and 3.6% for healthy speakers.

The majority of errors in these algorithms are found at the beginning and end of the voiced segments where detection usually starts slightly too early and ends too late. One can notice that the methods based on deep neural networks provide good results with about 8% of VDE, which is, however, not as good as the time-domain algorithms. This could be explained by the fact that the voicing decision has to be made by using a hard threshold on probabilities that can lead to isolated errors if the threshold is not strict enough. The results for the patients with PD also show the same trend with time-domain methods being more effective for voicing detection with about 4% of errors. The median

https://doi.org/10.1121/10.0015143

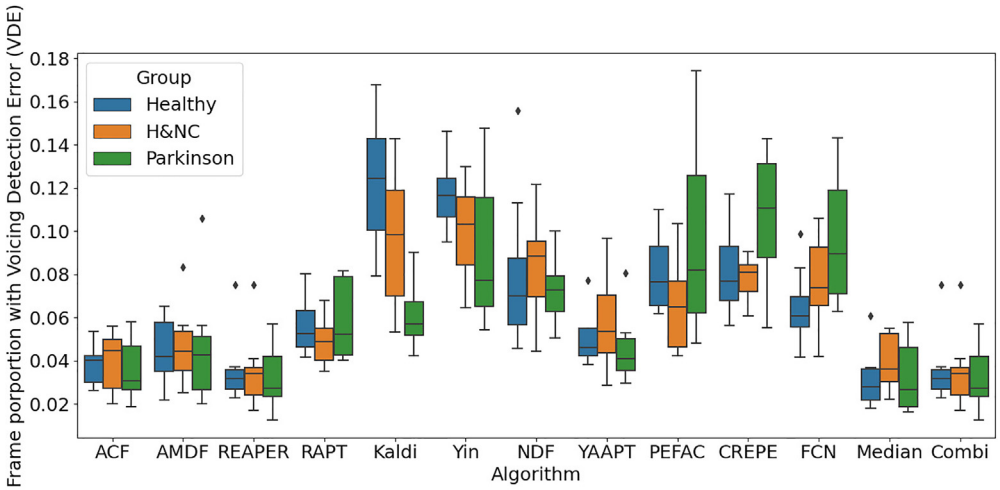

FIG. 4. (Color online) Results on voicing detection errors: Blue boxplots represent VDE for control speakers, orange boxplots are for speakers with H&NC, and green boxplots are for PD patients. Each boxplot represents an error percentage (lower percentages are optimal).

voting and YAAPT also gave similar results to time-domain algorithms.

It is noteworthy that the results obtained on the VDE metric are surprising because, for many algorithms, the performances are comparable in pathological and non-pathological speech.

B. Gross pitch errors

Figure 5 represents the results for the metric GPE, i.e., the percentage of voiced frames for which an algorithm has generated a value distanced by more than 20% from the real value of f_0.

For this metric, methods based on neural networks give good results with 1% errors for CREPE and 0.5% for FCN-f_0 on pathological voices. On the other hand, methods based on autocorrelation give good results on healthy speech, but produce more errors on pathological voice with much higher variations between speakers. A large part of these errors come from a one-octave estimation below the real value for time domain methods. Globally, GPE yields fewer good results for pathological speech compared to healthy speech, which confirms that the evaluation of the exact f_0 value on pathological speech is harder than for healthy, clean speech.

Surprisingly, the results are really good for PD patients, and are really close to the results for healthy speakers. Indeed, almost all algorithms give good results with less than 3% of mistakes. No difference in favor of one method or the other can be noticed. Overall, the performance of the algorithms on voices of speakers with H&NC show more errors on the determination of the f_0 value than for the two other groups.

C. f_0 frame errors

Figure 6 represents the results for the metric f_0 FFE, i.e., the percentage of frames where the algorithm has made a mistake, whether it is a voicing detection mistake or a gross pitch one.

Based on Fig. 6, it is difficult to choose one algorithm that outperforms the others. However, some of them provide relatively good results with little variation between speakers (small error bars). For example, YAAPT (which uses both time domain and spectral domain of the signal) provides

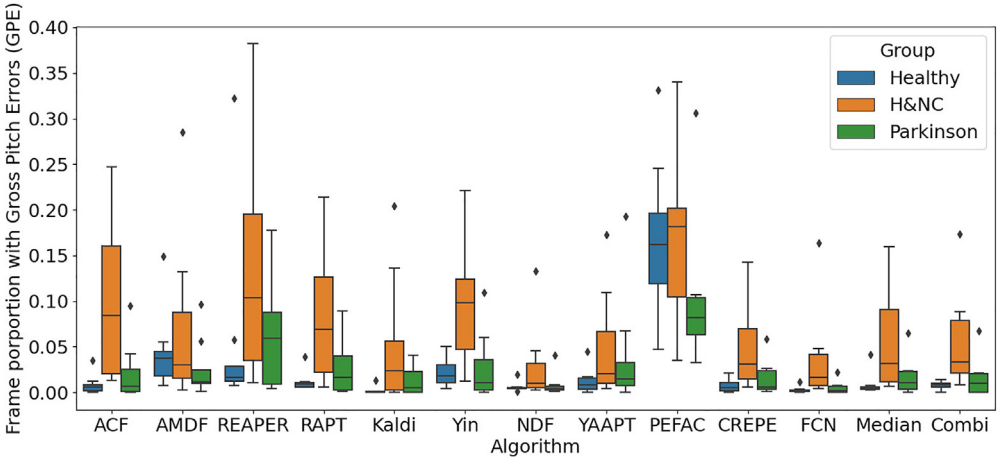

FIG. 5. (Color online) Results on gross pitch errors: Blue boxplots represent GPE for control speakers, orange boxplots are for speakers with H&NC, and green boxplots are for PD patients. Each boxplot represents an error percentage (lower percentages are better).

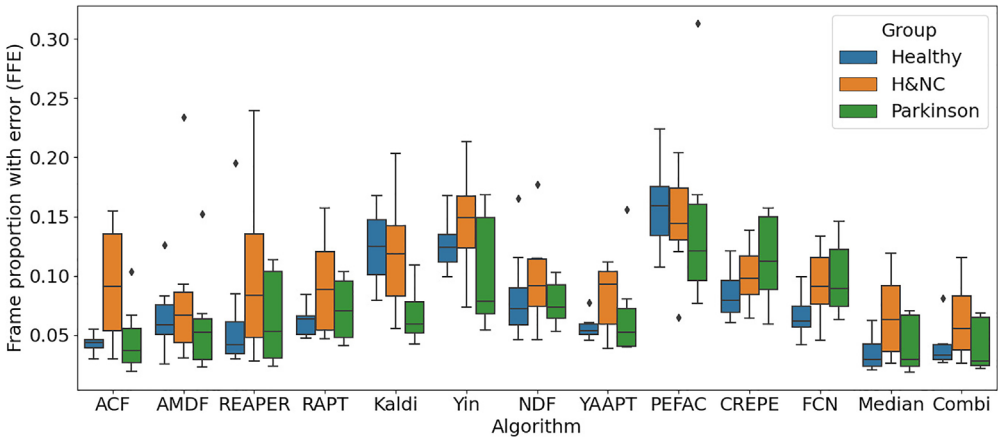

FIG. 6. (Color online) Results on f_0 frame errors for patients with H&NC and patients with PD: each boxplot represents an error percentage (lower percentages are better). Blue boxplots represent FFE for healthy speakers, orange boxplots are for speakers with H&NC, and green boxplots are for speakers with PD.

good results with only 7% of frames containing errors for pathological and healthy speech. FCN- f_0 also gives good and stable results with 7.3% of error for cancer patients and 6.2% for healthy speakers.

Finally, for PD patients, the results are again similar to those on healthy speakers. The algorithms with best detection performance are YAAPT, ACF, AMDF, and the merging methods.

D. Simultaneous f_0 zones

Concerning the simultaneous f_0 areas from the diplophonia phenomenon described in Sec. III B 2, we decided not to make a quantitative evaluation of the various pitch detection algorithms because it concerns too few segments (n = 35). Instead of computing statistics, a qualitative analysis were performed by looking more in detail at the outcomes for these particular zones. As indicated in Sec. III B 2, the f_0 values in these areas were annotated as an interpolation of the previous and next f_0 values. The reason behind this choice is that, for our future experiments, we will study stress and intonation patterns on the same corpus of pathological speech. To have a good estimation of the linguistic implementation of f_0, it is crucial to have a regular, precise f_0 curve and avoid sudden drops or increases. On the other hand, it is also valuable to have algorithmic estimations of f_0 sudden changes when searching to precisely characterize pathological speech. We we chose to compare different algorithms to our manual annotation (in black) for this reason.

After analyzing the results on the 35 overlapping f_0 intervals, a trend can be observed. The algorithms based on the time domain of the signal tend to show a sudden drop by an octave, as exhibited by the ACF curve in Fig. 7. The ACF algorithm generates a drop by an octave for 28 intervals (80%). Also, some algorithms avoid the drop by doing an interpolation between the previous and the following f_0 values. Typically, YAAPT seems to provide consistent results with 24 intervals (69%) in this particular case. Also, the neural network algorithms, like FCN- f_0, often give stable f_0 curves like YAAPT, but they also tend to tag these superposed f_0 patterns as unvoiced segments.

V. DISCUSSION AND CONCLUSION

This paper analyzed the performance from 12 algorithms based on time domain, frequency domain, or deep neural network techniques on clinical data including healthy speech, speakers with H&NC, and speakers with PD. Two methods of merging algorithms (combi and median) were also tested to determine whether the potential performance improvement induced by these merges was sufficient to alleviate its inherent technical constraints. The main objectives were to test the following:

(1) a large amount of widely used pitch detection algorithms for these two particular diseases, because they yield different f_0 detection problems
(2) these algorithms on connected speech, which is more ecological for automatic pitch detection tools but nevertheless never proposed in similar studies.

The experiments were run on a corpus composed of 24 French speakers (8 healthy, 8 for H&NC, 8 for PD) performing a reading task. The performance of the algorithms was tested through three metrics: VDE for voicing detection, GPE for estimation accuracy, and FFE for overall performance. A test were also computed based on a selection of three algorithms representative of time or frequency domain and deep neural network on specific speech areas with simultaneous f_0 typical of the diplophonia phenomenon. Indeed, although these simultaneous f_0 areas are relatively scarce, it is interesting to characterize the algorithms' behavior for future research.

Regarding the three metrics (VDE, GPE, FFE), our results indicate that algorithms based on the temporal study of the signal generate better results for voice detection.

https://doi.org/10.1121/10.0015143

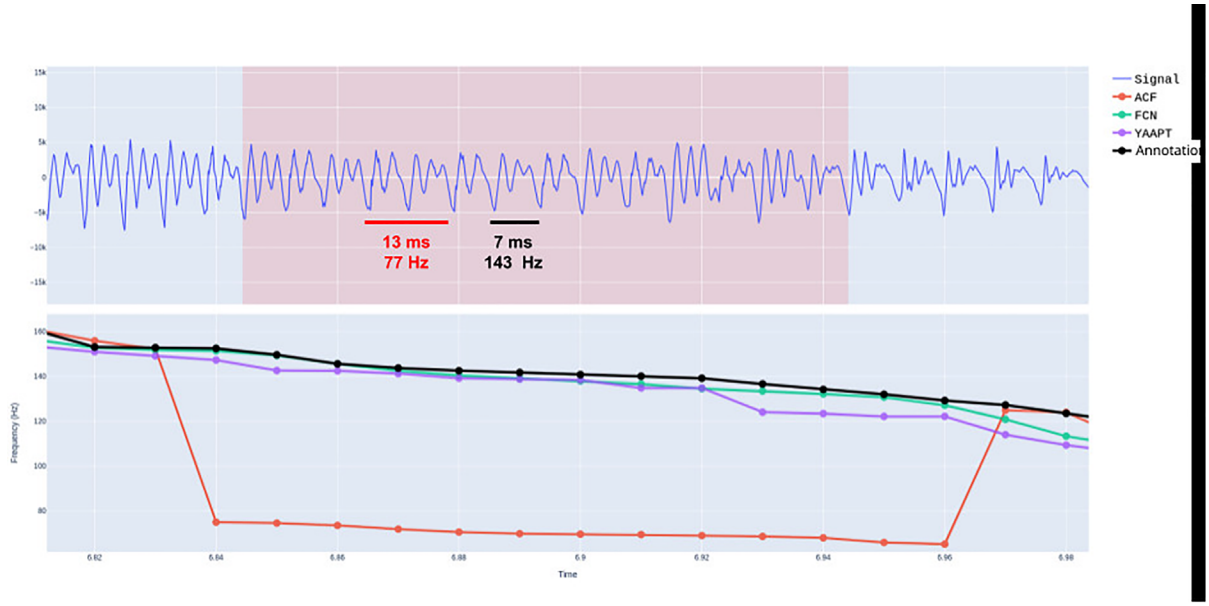

FIG. 7. (Color online) Example of results for three algorithms (ACF in red, FCN-F0 in green, and YAAPT in violet) on a particular zone with superposed f_0. The upper figure shows the raw signal with a superposed f_0 zone highlighted.

Indeed, ACF, REAPER, and AMDF algorithms offer the best VDE values with similar results for all speaker types (healthy or pathological) with an error rate below 5%. The algorithms using deep neural network however are most efficient to account for the accuracy of the estimates (GPE). Indeed, CREPE and FCN- f_0 produce less than 0.5% gross errors on healthy and Parkinsonian individuals. As a reminder, an estimation is considered as a gross error when it differs by more than 20% from the reference value. The results for speakers with H&NC generally yield worse GPE values, regardless of the algorithm chosen. Despite this, CREPE and FCN generate less than 1.5% error on average for this type of speech.

By running the three metrics on the fusion algorithm methods, our results indicate that the "median" algorithm fusion method using five algorithms (AMDF, Kaldi, NDF, FCN- f_0, and REAPER) generates overall better results than these methods taken individually. The median vote yields less than 4% of VDE for all speaker groups and less than 5% of GPE. However, the performance improvement does not seem to be significant enough to justify the technical cost (implementation and execution time) of running five algorithms. On the other hand, our results on the "combi" algorithm fusion method indicate that the simple combination of two methods (mixing REAPER for detecting voicing and FCN- f_0 for estimating the f_0 value) is efficient enough as it generates results at least similar to the "median" vote (around 5% of VDE and less than 3% of GPE), while executing only 2 algorithms.

To summarize, time-domain methods are best for voicing detection while deep neural networks generate more accurate f_0 estimations. Using a combination of just two algorithms (REAPER for good voicing detection and FCN-f_0 for accurate estimations) is a good compromise between performance and computational complexity.

From a clinical point of view, choosing the fittest f_0 detection algorithm is a crucial element depending on the studies one wishes to perform. However, this choice is rarely justified in clinical studies of pathological speech pitch. This study allowed us to highlight the strengths and weaknesses of different methods. It may, thus, help to best choose the relevant algorithm (or combination of algorithms) in future clinical studies, depending on the finality and purpose of the research. For example, if a study is particularly interested in the presence or absence of voicing, the use of an algorithm, such as ACF or AMDF, seems recommended as we did in Vaysse *et al.* (2022) where we measured the f_0 on a large corpora of Parkinsonian speech. Conversely, if one is mostly interested in retrieving the mean f_0 values of a corpus, it is more interesting to use methods, such as REAPER, FCN- f_0, or NDF, which generate robust estimates even if some voiced areas are missed. Interestingly, our results indicate that the same f_0 extraction algorithms (NDF and FCN- f_0 for accurate pitch estimation and REAPER or ACF for voicing detection) turn out to be the most reliable for speakers with PD and H&NC.

ACKNOWLEDGMENTS

This work has been carried out thanks to the French National Research Agency in 2018 as part of the RUGBI 2018 project entitled "Looking for relevant linguistic units to improve the intelligibility measurement of speech production disorders" (Grant No. ANR-18-CE45-0008). The first corpus is taken from the Carcinologic Speech Severity Index (C2SI) project (Grant No. 2014-135) from Institut National du Cancer (INCa).

APPENDIX: RAW RESULTS

The raw results of the tested algorithms are presented in Table IV.

TABLE IV. Raw results of the 14 tested algorithms on the different metrics and sub-metrics. C, controls (healthy) speakers; H, H&NC; P, Parkinson's disease patients; VDE, voicing detection errors; FNR, false negative rate of VDE; FPR, false positive rate of VDE; GPE, the gross pitch errors; ×2 and ÷2, rates of algorithm estimations above and below 20% from our annotations, respectively (see Sec. III B). Values in boldface are the best scores for each metric.

	VDE (%)			GPE (%)			FNR (%)			FPR (%)			×2 (%)			÷ 2 (%)		
Algorithm	C	H	P	C	H	P	C	H	P	C	H	P	C	H	P	C	H	P
ACF	3.8	4.0	3.6	0.9	10.0	2.1	**1.3**	2.4	2.4	2.5	1.7	1.1	0.1	**0.7**	0.2	0.8	9.3	1.9
AMDF	4.5	4.7	4.6	4.6	7.2	2.6	3.1	3.6	3.6	1.4	1.0	1.0	2.1	4.6	1.4	2.4	2.7	1.2
REAPER	3.6	**3.5**	**3.3**	5.8	13.7	6.3	1.5	**1.3**	**1.7**	2.0	2.2	1.6	0.2	1.4	0.2	5.6	12.3	6.2
RAPT	5.6	4.9	5.9	1.2	8.6	2.6	2.6	2.2	**1.7**	3.0	2.7	4.2	0.2	1.4	0.5	1.0	7.3	2.1
Enhanced RAPT	12.4	9.6	6.0	**0.2**	5.3	1.3	8.1	6.7	3.3	4.2	2.9	2.7	0.1	**0.9**	0.6	**0.1**	4.4	0.7
Yin	11.7	10.1	9.1	2.3	9.8	2.8	6.8	6.6	6.1	4.9	3.5	3.0	0.1	1.2	0.2	2.2	8.5	2.6
NDF	8.0	8.4	7.3	0.6	**3.0**	0.9	1.4	3.2	2.5	6.7	5.2	4.8	0.2	1.9	0.5	0.4	**1.0**	**0.3**
YAAPT	5.0	5.9	4.6	1.3	5.0	4.0	2.7	4.4	2.9	2.3	1.4	1.6	0.2	1.7	0.3	1.1	3.2	3.8
SWIPE	6.7	9.2	6.2	79.2	68.1	91.8	6.0	8.8	5.1	**0.7**	**0.4**	1.1	18.3	5.4	9.1	60.9	62.7	82.7
PEFAC	8.1	6.6	9.5	16.8	16.7	10.5	4.1	3.5	3.9	4.0	3.0	5.5	6.9	8.0	7.1	9.9	8.6	3.4
CREPE	8.2	7.8	10.6	0.8	4.8	1.6	2.9	4.4	4.7	5.3	3.4	5.9	0.4	2.1	0.7	0.4	2.7	0.9
FCN-F0	6.5	7.7	9.6	0.3	3.8	**0.5**	3.7	5.0	5.2	2.8	2.6	4.3	**0.0**	1.7	**0.1**	0.3	2.1	0.4
Median	**3.1**	3.9	**3.3**	0.9	5.7	1.7	1.9	2.8	2.1	1.2	1.1	1.2	**0.0**	1.5	0.3	0.9	4.2	1.4
Combi	3.6	**3.5**	**3.3**	0.7	5	1.6	2.6	4.0	3.6	1.4	1.2	**0.9**	0.1	1.9	0.5	0.3	**1.0**	**0.3**

Ardaillon, L., and Roebel, A. (**2019**). "Fully-convolutional network for pitch estimation of speech signals," in *Proc. Interspeech 2019*, pp. 2005–2009.

Babacan, O., Drugman, T., D'Alessandro, N., Henrich, N., and Dutoit, T. (**2013**). "A comparative study of pitch extraction algorithms on a large variety of singing sounds," in *38th International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2013)*, Vancouver, Canada, pp. 1–5.

Bellman, R., (**1954**). "The theory of dynamic programming," Bull. Am. Math. Soc. **60**(6) 503–515.

Boersma, P. (**2000**). "Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound," in *Proceedings of the Institute of Phonetic Sciences 17*.

Boersma, P., and Weenink, D. (**2020**). "Praat: Doing phonetics by computer (version 6.1.16) [computer program]," http://www.praat.org (Last viewed January 20, 2022).

Brookes, M. (**2018**). {VOICEBOX: Speech Processing Toolbox for MATLAB, available at http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html (Last viewed November 12, 2022).

Camacho, A., and Harris, J. (**2008**). "A sawtooth waveform inspired pitch estimator for speech and music," J. Acoust. Soc. Am. **124**, 1638–1652.

Cesari, U., De Pietro, G., Marciano, E., Niri, C., Sannino, G., and Verde, L. (**2018**). "A new database of healthy and pathological voices," Comput. Electr. Eng. **68**, 310–321.

Chu, W., and Alwan, A. (**2009**). "Reducing f0 frame error of f0 tracking algorithms under noisy conditions with an unvoiced/voiced classification frontend," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 3969–3972.

Daudet, A. (**1870**). *Lettres de mon moulin: Impressions et souvenirs (Letters from my Windmill)* (Hetzel, Paris).

de Cheveigné, A., and Kawahara, H. (**2001**). "Comparative evaluation of F0 estimation algorithms," in *Eurospeech*, NA, France, pp. 2451–2454.

de Cheveigne, A., and Kawahara, H. (**2002**). "YIN, a fundamental frequency estimator for speech and music," J. Acoust. Soc. Am. **111**(4), 1917–1930.

Di Cristo, A. (**2016**). *Les musiques du français parlé: Essais sur l'accentuation, la métrique, le rythme, le phrasé prosodique et l'intonation du français contemporain (The Music of Spoken French: Essays on Accentuation, Metrics, Rhythm, Prosodic Phrasing and Intonation of Contemporary French)* (de Gruyter, Berlin).

Drugman, T., and Alwan, A. (**2011**). "Joint robust voicing detection and pitch estimation based on residual harmonics," in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pp. 1973–1976.

Espesser, R. (**1996**). "MES : un environnement de traitement du signal" ("MES: A signal processing environment"), *XXIèmes Journées d'\'Etude sur la Parole* (*XXIst Study Days on the Word*), Avignon, France, p. 447.

Espesser, R. (**1999**). "Mes signaix package," Technical Report.

Ghahremani, P., BabaAli, B., Povey, D., Riedhammer, K., Trmal, J., and Khudanpur, S. (**2014**). "A pitch extraction algorithm tuned for automatic speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2494–2498.

Gonzalez, S., and Brookes, M. (**2014**). "PEFAC - A pitch estimation algorithm robust to high levels of noise," IEEE/ACM Trans. Audio. Speech. Lang. Process. **22**(2), 518–530.

Google-Open-Source (**2015**). "Reaper: Robust epoch and pitch estimator," https://github.com/google/REAPER (Last viewed September 20, 2020).

Hess, W. J. (**2008**). *Pitch and Voicing Determination of Speech with an Extension Toward Music Signals* (Springer Berlin Heidelberg, Berlin, Heidelberg), pp. 181–212.

Jang, S.-J., Choi, S,-H., Kim, H.-M., Choi, H.-S., and Yoon, Y.-R. (**2007**). "Evaluation of performance of several established pitch detection algorithms in pathological voices," in *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 620–623.

Jiménez-Jiménez, F. J., Gamboa, J., Nieto, A., Guerrero, J., Orti-Pareja, M., Molina, J. A., García-Albea, E., and Cobeta, I. (**1997**). "Acoustic voice analysis in untreated patients with Parkinson's disease," Parkinsonism Relat. Disord. **3**(2), 111–116.

Jouvet, D., and Laprie, Y. (**2017**). "Performance analysis of several pitch detection algorithms on simulated and real noisy speech data," in *2017 25th European Signal Processing Conference (EUSIPCO)*, pp. 1614–1618.

Kåre, S. (**2005**). The Snack Sound Toolkit (Version 2.2.10), available at https://www.speech.kth.se/snack/index.html (Last viewed November 11, 2022).

Kasi, K., and Zahorian, S. (**2002**). "Yet another algorithm for pitch tracking," in *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 1, p. 361.

Kawahara, H., Cheveigné, A., Banno, H., Takahashi, T., and Irino, T. (**2005**). "Nearly defect-free f0 trajectory extraction for expressive speech modifications based on straight," in *Ninth European Conference on Speech Communication and Technology*, pp. 537–540.

Kawahara, H. (**2018**). STRAIGHT, a speech analysis, modification and synthesis system, available at http://web.wakayama-u.ac.jp/~kawahara/STRAIGHTadv/index_e.html (Last viewed November 11, 2022).

Keating, P. A., Garellek, M., and Kreiman, J. (**2015**). "Acoustic properties of different kinds of creaky voice," in *Proceedings of the 18th International Congress of Phonetic Sciences, Glasgow*, Vol. 2015, pp. 2–7.

Kim, J. W., Salamon, J., Li, P., and Bello, J. P. (**2018**). "CREPE: A convolutional representation for pitch estimation," arXiv:1802.06182.

Le Dorze, G. L., Ouellet, L., and Ryalls, J. (**1994**). "Intonation and speech rate in dysarthric speech," J. Commun. Disorders **27**(1), 1–18.

Luengo, I., Saratxaga, I., Navas, E., Hernaez, I., Sanchez, J., and Sainz, I. (**2007**). "Evaluation of pitch detection algorithms under real conditions," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, Vol. 4, pp. IV-1057– IV-1060.

Parsa, V., and Jamieson, D. G. (**1999**). "A comparison of high precision F0 extraction algorithms for sustained vowels," J. Speech. Lang. Hear. Res. **42**(1), 112–126.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Silovsky, J., Stemmer, G., and Vesely, K. (**2011**). "The Kaldi Speech Recognition Toolkit", in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding* (IEEE, New York).

Ross, M., Shaffer, H., Cohen, A., Freudberg, R., and Manley, H. (**1974**). "Average magnitude difference function pitch extractor," IEEE Trans. Acoust. Speech, Signal Process. **22**(5), 353–362.

RUGBI (**2018–2023**). "Looking for relevant linguistic units to improve the intelligibility measurement of speech production disorders," https://www.irit.fr/rugbi (Last viewed November 10, 2022).

Soquet, A. (**1994**). "Approche coopérative de l'extraction de la fréquence fondamentale" ("A cooperative approach of f0 extraction"), in *XXèmes Journées D'Études Sur la Parole (XXth Study Days on the Word)*, Trégastel, France, pp. 229–234.

Strömbergsson, S. (**2016**). "Today's most frequently used F_0 estimation methods, and their accuracy in estimating male and female pitch in clean speech," in *Proc. Interspeech 2016*, pp. 525–529.

Talkin, D., and Kleijn, W. B. (**1995**). "A robust algorithm for pitch tracking (RAPT)," in *Speech Coding Synthesis*, edited by W. B. Kleijn and K. K. Paliwal (Elsevier Science B. V., Amsterdam), pp. 495–518.

Tsanas, A., Zañartu, M., Little, M. A., Fox, C., Ramig, L. O., and Clifford, G. D. (**2014**). "Robust fundamental frequency estimation in sustained vowels: Detailed algorithmic comparisons and information fusion with adaptive Kalman filtering," J. Acoust. Soc. Am. **135**(5), 2885–2901.

Tokuda, K., Oura, K., Yoshimura, T., Tamamori, A., Sako, S., Zen, H., Nose, T., Takahashi, T., Yamagishi, J., and Nankaku, Y. (**2017**). Speech Signal Processing Toolkit (Version 3.11), available at https://sp-tk.sourceforge.net/ (Last viewed Novembver 12, 2022).

Vaysse, R., Ghio, A., Astésano, C., Farinas, J., and Viallet, F. (**2022**). "Analyse macroscopique des variations et modulations de F0 en lecture dans la maladie de Parkinson: Données sur 320 locuteurs "Macroscopic analysis of f0 variations and modulations in read speech for Parkinson disease patients: Data from 320 speakers"), in *34e Journées D'Études Sur la Parole (JEP2022)*, [34th 740 Speech Study Days (JEP2022)] (Association Française de la Communication Parlée, Noirmoutier, France, to be published).

Woisard, V., Astésano, C., Balaguer, M., Farinas, J., Fredouille, C., Gaillard, P., Ghio, A., Giusti, L., Laaridh, I., Lalain, M., Lepage, B., Mauclair, J., Nocaudie, O., Pinquier, J., Pouchoulin, G., Puech, M., Robert, D., and Roger, V. (**2021**). "C2SI corpus: A database of speech disorder productions to assess intelligibility and quality of life in head and neck cancers," Lang. Resour. Eval. **55**(1), 173–190.

Zahorian, S. A., and Hu, H. (**2016**). YAAPT Pitch Tracking MATLAB Function, available at http://www.ws.binghamton.edu/zahorian/yaapt.htm (Last viewed November 11, 2022).