IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 28 December 2022, accepted 23 January 2023, date of publication 27 January 2023, date of current version 1 February 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3240181

RESEARCH ARTICLE

A Pitch Estimation Algorithm for Speech in Complex Noise Environments Based on the Radon Transform

BAI LI AND XIANWU ZHANG
School of Information Science and Engineering, Xinjiang University, Xinjiang 830046, China

Corresponding authors: Xianwu Zhang (zxw@xju.edu.cn) and Bai Li (libai@stu.xju.edu.cn)

This work was supported in part by the Natural Science Foundation of Xinjiang Uygur Autonomous Region under Grant 2022D01C61, and in part by the ''Tianchi Doctoral Program'' Project of Xinjiang Uygur Autonomous Region under Grant TCBS202046.

ABSTRACT The pitch period as an essential feature is used in various speech-related works. Most actual projects collect speech signals in complex noise environments. Thus, the noise resistance of the algorithm for accurate pitch estimation has become more critical than ever. However, many state-of-the-art algorithms fail to obtain good results when dealing with noisy speech files at a low signal-to-noise ratio (SNR) value. This study presents a new noise-resistant pitch estimation algorithm based on the Radon transform and reduces the influence of formants with the modification of the classical equation. In addition, we use the difference between the pitch candidates of the consecutive frames as part of the criterion for the decoding of the Viterbi algorithm to strengthen the correlation of the pitch estimates and make the pitch contours smoother. We synthesized three noisy speech databases with 18 types of collected environmental noise and compared our algorithm with 7 state-of-the-art algorithms. The proposed algorithm has the best performance on CSTR and self-recorded databases and reduces Gross Pitch Error (GPE) rate by over 12% at 0 dB SNR against Bayesian Pitch Tracker. In particular, the GPE rate of our proposed algorithm can be maintained under 25% at 0 dB SNR, while BaNa only achieves 35%.

INDEX TERMS Pitch estimation, radon transform, Viterbi algorithm, noise resistance.

I. INTRODUCTION

The vocal folds vibrate when people speak, and the time taken for the vocal folds to open and close each time represents the pitch. According to the quasi-periodic nature of vocal fold vibration, the pitch can be called in terms of the pitch period, and its inverse is known as the fundamental frequency. People who speak with high-sounding voices tend to have high fundamental frequencies and vice versa. Each individual's fundamental frequency varies with different characteristics. Males can reach a low fundamental frequency of around 60 Hz, and children and females go up to 500 Hz. These different fundamental frequency ranges also provide a massive reference for our study.

The pitch period as a piece of vital information can be applied to various speech-related works. [1] increases the intelligibility of noisy speech by applying pitch enhancement in the frequency domain. [2] uses the pitch period to build both noise and speech long-term models for human speech enhancement. The pitch period also provides valid information for automatic speech recognition (ASR) systems. One study uses prosodic events in the form of pitch accents to improve speech recognition in a baseline ASR system [3]. Another study builds a pitch-adaptive speech recognition system for children by reducing pitch variation sensitivity [4]. In order to make the applications above more practical, we need to extract accurate pitch information from the speech.

However, many difficulties exist in extracting pitch from speech. First, the speech signals produced by the voiced sounds are not perfectly periodic [5]. Then, these signals pass through the vocal tract and produce formants, which cause significant changes in the structures of the speech signals [6]. Finally, environmental noise is one of the most critical factors

The associate editor coordinating the review of this manuscript and approving it for publication was Khursheed Aurangzeb.

affecting pitch estimation. In actual projects, a simple denoise method before pitch estimation algorithm can neither be adaptive to all types of noise nor maintain the original structure of speech signal. Additionally, we found by experiments in Section III that most dominant pitch estimation algorithms based on the autocorrelation function (ACF) [7], the average magnitude difference function (AMDF) [8], or Cepstrum [9] do not perform well at low SNR values because they do not have robust noise resistance.

The Radon transform proposed in [10] was first applied to reconstruct the spatial distribution of projection-based objects. Since then, it has been used mainly in subjects related to image processing, such as medicine and seismology. In [11], the Radon transform is used for seismic data processing to achieve significant noise resistance which is seldom applied to speech-related works. Other state-of-the-art transforms like Discrete Hahn polynomials (DHPs) and Discrete Tchebichef polynomials (DTPs) are also applied in image processing and speech processing and have remarkable performance. In [12], an operative method is proposed to compute the Hahn orthogonal basis for high orders and effectively reduced the computational cost. In [13], a fast and accurate algorithm for high-order DTPs is proposed which provided great inspiration for signal processing.

In this study, we propose to use its robust noise resistance to detect pitch instead of the previous basic algorithms. We use the Radon transform to generate pitch candidates and then use the Viterbi algorithm to find the most likely pitch contour which will be expounded on in Section III.

In order to compare the effect of our algorithm with other algorithms for pitch estimation in complex noise environments, we recorded environment sounds of 18 scenarios, such as airplane, restaurant, street, subway, night market, and so on. The clean speech database is derived from the Keele database [14], CSTR database [15], and several self-recorded speech files. We generate the noisy speech database by synthesizing the noise and clean speech files under SNR values from −10 dB to 20 dB. After comparing the proposed algorithm with several state-of-the-art noise-resilient algorithms, such as BaNa [16] and Robust Bayesian Pitch Tracking [17], we found that our proposed algorithm performs best in aggregate under SNR values from −10 dB to 20 dB. The Gross Pitch Error (GPE) rate of the proposed algorithm can be maintained under 25% when detecting noisy speech signals at 0 dB SNR. The contributions of this study are as follows:

1) The Radon transform is introduced into the pitch estimation method to cope with complex noise.
2) A modification of the Radon transform is proposed to reduce the influence of high-frequency components like formants.
3) Speech signals can be converted between 1D and 2D with the proposed pseudo-2D image and energy function.

The rest of the paper is as follows. In Section II, we discuss related works of pitch estimation algorithms. A comprehensive analysis of the proposed algorithm is in Section III. Our experimental settings and data comparison results are in Section IV and Section V. Finally, Section VI describes the conclusions.

II. RELATED WORKS

Hitherto, all pitch estimation algorithms can be classified into non-data-driven and data-driven approaches. Non-data-driven approaches, as the name implies, do not need previous data information and can be subdivided into two types: non-parametric and parametric methods [17]. On the other hand, data-driven approaches require previous information in different scenarios to be effective.

Non-parametric methods can be further divided into time-domain methods, frequency-domain methods, and hybrid methods [6]. The most applied pitch estimation algorithms are time-domain methods. ACF [7] calculates the autocorrelation between the original signal and the lagged signal to extract the pitch information. AMDF [8] reduces the computational complexity by changing the multiplication method of autocorrelation into a subtraction method. Circular AMDF (CAMDF) [18] and extended AMDF (EAMDF) [19] both improve detection accuracy by reducing the falling tendency of the original difference function. Praat [20] imports the Viterbi algorithm to find a pitch contour, which provides a new idea for pitch estimation. The RAPT algorithm [21], on the other hand, performs pitch estimation by measuring the normalized cross-correlation function (NCCF). YIN [22] proposed by Cheveigne and Kawahara dramatically improves pitch estimation accuracy by using the absolute threshold and parabolic interpolation, but its discontinuity between the previous and following frames makes the pitch contours not smooth. The probabilistic YIN (pYIN) [23] algorithm is an improvement of YIN, which obtains a large number of pitch candidates with probabilistic threshold distributions and uses probabilities as observations of the Hidden Markov Model (HMM) to decode a smoother pitch contour. However, pYIN may determine the voiced segments of noisy speech at low SNR values as unvoiced segments and fail to extract the complete pitch contour.

Pitch estimation can be implemented by analyzing frequency-domain features. Cepstrum [9] enables the pitch to emerge as a peak and reduces the influence of the noise at a certain level. Recently, a new Cepstrum method was proposed to compensate for octave errors and decreased the estimation errors of music signals [24]. The Harmonic Product Spectrum (HPS) algorithm [25] uses the theory of Schroeder's frequency histogram to obtain the pitch period by measuring each harmonic period and calculating their least common multiple, but it is susceptible to half-octave errors. SWIPE [26] uses the harmonic summation method with weights similar to sawtooth waves, effectively overcoming the half-octave errors in the frequency domain. Camacho improves SWIPE and proposes SWIPE' [26] by choosing only the candidate and prime frequencies in harmonic summation. However, their noise resistance needs further improvement.

Hybrid methods often integrate features in different domains. Pitch Estimation Filter with Amplitude Compression (PEFAC) [27] attenuates narrowband noise with amplitude compression and convolves the power spectral density of each speech frame, which suppresses the additive noise with a smooth power spectrum while aggregating the harmonic energy. YAAPT [28] calculates the path of the pitch in the frequency domain by using the Spectral Harmonics Correlation of the signal after nonlinear processing and selects the pitch candidates with NCCF in the time domain. BaNa [16] proposed by Yang and Ba integrates the fundamental frequency candidates selected from the harmonic frequencies with the candidate selected from Cepstrum and then uses the Viterbi algorithm to track the pitch contours. Reference [6] shows that BaNa has great noise resistance and detection abilities among non-parametric methods.

Parametric methods, such as harmonic model-based methods, always show high robustness to additive noise [29]. The fast NLS [30] as a computationally efficient pitch tracker is less affected by octave errors. A robust Bayesian harmonic model-based pitch tracker [17] as a state-of-the-art parametric method uses first-order Markov processes and information from previous frames to improve noise robustness.

In recent years, a large number of data-driven approaches have contributed to noise resistance. The TAPS algorithm [31] trains the peak spectrum exemplar set and uses the difference between temporal accumulations of clean and noisy speech data for pitch estimation. In [32], Chu and Alwan proposed SAFE to model the effects of noise on the locations and amplitudes of the peaks in the clean speech spectrum. Crepe [33] uses a convolutional neural network to train a synthetic database and produces a fundamental frequency estimate from the network. Self-Supervised Pitch Estimation (SPICE) [34] trains the constant Q transform of the signals and calibrates the trained data to achieve better results. DeepF0 [35] extends the receptive field of a network to capture pitches under various levels of noise. HarmoF0 [36] proposed recently outperforms DeepF0 by evaluating the multiple rates dilated causal convolution and other dilated convolutions in pitch estimation. In actual projects, different conditions mean different types of environmental noise. Data-driven approaches are trained with known noise types and specific noise levels, so the noise information needs to be used as input to the model. However, the type of noise and the noise level is hard to get in complex conditions [16].

Our proposed algorithm based on the Radon transform selects a non-data-driven approach that does not require any previous noise information and introduces the noise resistance of the Radon transform to pitch estimation. As a time domain method, our proposed algorithm reduces the influence of formants by adding a logarithmic function and a power function. According to Praat [20] and BaNa [16], we use the difference between the pitch candidates of the consecutive frames as part of the criterion for the decoding of the Viterbi algorithm in our algorithm. This strengthens the correlation of the pitch estimates of the consecutive frames in a speech segment and makes the pitch contours smoother.

III. DETECTING PITCH WITH THE RADON TRANSFORM AND THE VITERBI ALGORITHM

A. PREPROCESSING

Based on the short-time stability of speech, the proposed algorithm, like most algorithms, needs to frame the speech signal. The settings of frame length win and time step s are described in Section IV. The sampling rate of speech is denoted as Fs, N_{frame} is the total number of frames, and the number of samples per frame is denoted as N_{sample}. We also denote l_{max} as the pitch period's upper limit of the detection range corresponding to the minimum fundamental frequency and l_{min} as the lower limit. We use a certain frame to describe the process of the proposed algorithm in Section III-B and Section III-C.

B. FINDING PITCH LOCATIONS WITH THE RADON TRANSFORM

In prevailing studies, the classical Radon transform (RT) is used for processing images. [11] indicates the Radon transform can significantly increase the ratio of periodic signal to non-periodic noise of the image through the principle proposed in [10]. In the meantime, the pitch period is quasi-periodic, so it is possible using the Radon transform for speech pitch estimation. The principle of the classical linear Radon transform given in [10] presents as follows.

For a given image, we first need to specify the coordinates of a point in the image as (q, t). Then, in this $q - t$ plane, we assume that there exists a straight line l with a given gradient p and an intercept b. Thus, the slope-intercept form of the line l is defined to be

$$t = pq + b \tag{1}$$

The classical Radon transform is defined as a two-step process: first calculating the line integral of the intensity over the given line l in the original image, then considering each new integral value as the intensity of the point (p, b) in the new $p - b$ plane. A line with a different gradient or intercept will result in a different point in the $p - b$ plane. It will eventually form a whole Radon transformed image. The classical Radon transform equation is shown in (2):

$$R(p, b) = \int_l I(q, t)dl \tag{2}$$

where $I(q, t)$ denotes the intensity of the point (q, t) in the target image, and similarly, $R(p, b)$ represents the intensity of the point (p, b) in the Radon transformed image. It shows that the principle of the Radon transform is essentially a conversion of the image from the $q - t$ plane to a new $p - b$ plane.

Equation (2) indicates that if the target image has periodic changes with a period T in the vertical direction, the line integrals calculated from a set of lines with the gradient $p_0 = T$ will have some enhancing effects in the $p - b$ plane presenting a distinct spectral line at $p = p_0$ with alternating light and dark changes. In addition, spectral lines with the same characteristics will appear at the locations where p is equal to an integer multiple of the period $T(p = aT)$, where a indicates any of the non-negative integers. When p is not equal to an integer multiple of the period T, the line integrals will have adverse effects in the $p-b$ plane, making the values at $p \neq aT$ close to zero. The Radon transform can accentuate the periodic components and suppress the non-periodic noise of the image. Namely, it improves the SNR. Next, we will discuss applying the Radon transform to pitch estimation.

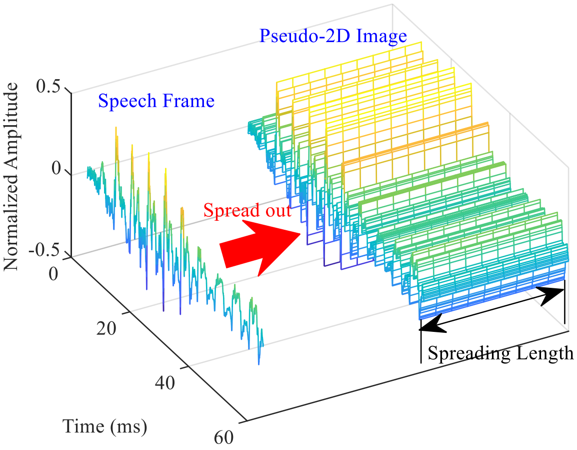


FIGURE 1. Spreading out the target speech frame to the pseudo-2D image with a spreading length of 100.

The easiest way to use the Radon transform on pitch estimation is to convert the speech frame into a pseudo-2D image. Here, we propose a simple idea to cope with this conversion: horizontally spreading out the speech frame with a spreading length N_{spread}. The variability of the data in the 2D image generated by this method is reflected only in the vertical direction but not in the horizontal direction, so we describe it as the pseudo-2D image. Meanwhile, the periodicity of speech is also reflected in the vertical direction.

According to the principle of the Radon transform [10], if we horizontally spread out a strictly periodic signal into an image and transform it to the Radon transformed image, we will definitely find spectral lines with alternating light and dark changes at $p = aT$. Therefore, we only need to find the most obvious spectral line in the $p-b$ plane generated by the pseudo-2D image to determine the pitch period of a speech frame. The spreading method is shown in Fig. 1.

The coordinates of a point in the pseudo-2D image can still be represented by (q, t), where q denotes the qth sample of the speech frame of the pseudo-2D image from left to right, and q will not exceed the spreading length N_{spread}. The vertical coordinate t represents the time.

Since the sampled speech data used in the experiments are discrete, using the classical Radon transform to process the data would be ineffective, so we use the discrete Radon transform (DRT) proposed in [37] to modify the original equation:

$$R(p, b) = \sum_{q=1}^{N_{spread}} I(q, u) \tag{3}$$

where

$$u = p(q-1) + b + 1 \tag{4}$$

In Equation (3), we replace the line integral with a discrete summation method and substitute the time t with the location of samples u, where $I(q, u)$ and $R(p, b)$ denote respectively the intensity in the discrete pseudo-2D image and the discrete Radon transformed image. Furthermore, in order not to increase the computational complexity, we do not use circular or extended methods to optimize the discrete Radon transform equation. In particular, we specify that once the location of samples u exceeds the number of samples per frame N_{sample} during the computation, the value of $I(q, u)$ is set to 0. To align the gradient p on the same scale with other time-domain methods, we compute Equations (3) and (4) from $p = 0$ to $p = l_{max}$. In addition, we define the max computation range of intercept b to be consistent with the number of samples per frame N_{sample}.

In the Radon transformed image, the computation with the above settings will inevitably yield a spectral line at $p = 0$ with tremendous absolute values, making us impossible to view the result of the pitch in the image effectively. Hence, we add a modification to post-process the discrete Radon transform.

Theoretically, the spectral line at $p = 0$ corresponds to the infinite frequency, which is useless to our algorithm, so we need to select a function of p to make all values of the spectral line at $p = 0$ become 0. In addition, since the fixed Radon transform is calculated, the number of computable points for the Radon transform becomes less when the gradient p increases to a condition that $p \times N_{spread}$ is larger than the number of vertical speech frame samples N_{sample} in the pseudo-2D image, resulting in a shallow high-period region on the right side of the Radon transformed image. Thus, this modification function needs to be monotonically increasing to complement the high-period region. We find that an arbitrary logarithmic function satisfies the above requirements.

In order to further weaken the low period region affected by formants and strengthen the pitch period region, we will introduce another function with the same monotonicity of the logarithmic function. Due to the limitations of the Radon transform itself, it will undoubtedly bring the radiation of the pitch to the spectral lines at multiples of the pitch period, leading to unpredictable influences at 1/2, 1/3 octaves, etc. Thus, we add a power function, of which second-order derivative is less than 0, to complement the logarithmic function dynamically and introduce less interference at the 1/2, and 1/3 octaves. The modified Radon transform function $R_r(p, b)$ is defined as

$$R_r(p, b) = R(p, b) \times \lg(p+1) \times (p+1)^{\frac{1}{3}} \tag{5}$$

where a decimal logarithm function and a power function are chosen.

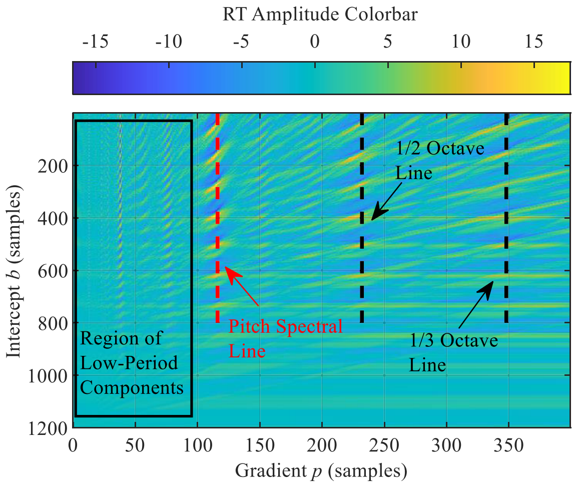


FIGURE 2. The modified radon transformed image.

Fig. 2 shows the result of the modified Radon transformed image. In Fig. 2, the black borders on the left side indicate the region of the low-period components, the red dashed line depicts the exact location of the pitch period, and the two black dashed lines represent the 1/2 octave line and the 1/3 octave line, respectively.

In the Radon transformed image, the gradient p represents the location of the pitch period. A spectral line with the most conspicuous alternating light and dark changes can be viewed at the pitch period. At the same time, the effects of the modification make both the spectral lines at submultiple octaves and the region of the low-period components look dim. The Radon transformed image obtained by the above steps builds a good foundation for the following works of extracting the pitch period.

C. GENERATING THE ENERGY FUNCTION AND COMPARISON OF NOISE RESISTANCE OF THE RADON TRANSFORM WITH OTHER BASIC ALGORITHMS

The modified Radon transformed image obtained in Section III-B visualizes the location of the pitch spectral line. To extract the exact value of the pitch period from the image, we propose the extraction method in Section III-C and Section III-D.

We find the location of the pitch period corresponds only to the gradient p. We define the pitch period as

$$PitchPeriod = \frac{p}{Fs} \times 10^3 \tag{6}$$

In the Radon transformed image, the most obvious spectral line indicates the most likely location of the pitch, where the points have the largest absolute values. We then use the sum of the squares of the values on each spectral line to reflect the

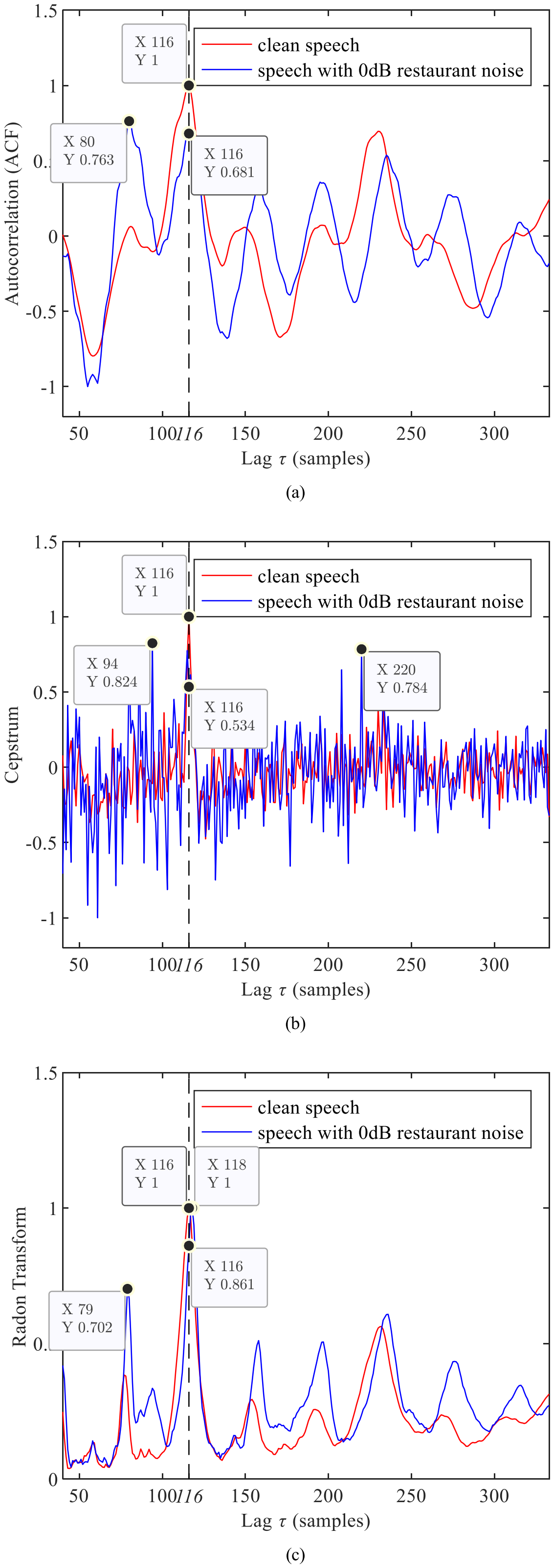


FIGURE 3. Results of a frame of clean speech and speech with 0 dB restaurant noise using three basic algorithms: (a) ACF, (b) Cepstrum, and (c) Radon transform. The location of each maximum value represents its pitch value.

obviousness. Thus, we create a function named *pline* using the gradient p as its variable, which is defined as

$$pline(p) = \sum_{b=0}^{N_{spread}-1} R_r(p, b)^2 \tag{7}$$

Equation (7) indicates that for each given gradient p_0, we define $pline(p_0)$ as the sum of squares of all points of the spectral line at $p = p_0$ in the modified Radon transformed image. The function *pline* represents the change of energy of the variable p, so we call it the energy function. We normalize the *pline* in preparation for the pitch tracking in Section III-D.

We use the energy function of the Radon transform to compare its noise resistance with other basic algorithms (ACF and Cepstrum). A clean speech frame and its noisy speech frame with 0 dB restaurant noise are chosen to be analyzed in the coming example. The results are shown in Fig. 3.

Figs. 3(a) and 3(b) show that both previous basic algorithms (ACF and Cepstrum) obtain the same pitch at 116 samples. However, their positive maximum values significantly change when calculating the noisy speech frame. Fig. 3(a) reaches the maximum ACF value of the noisy speech at 80 samples, and the value at 116 samples is much lower than the original one. In Fig. 3(b), the Cepstrum value has two high peaks at 94 and 220 samples, both of which have larger values than the value at 116 samples. Thus, we infer that neither ACF-based nor Cepstrum-based algorithms can easily extract the pitch from a noise-corrupted frame. In fact, noise like restaurant noise, which distorts frequencies of the entire speech spectrum, is prevalent in actual projects.

Fig. 3(c) shows the results with energy function of the proposed algorithm. The maximum value of the clean speech is still located at 116 samples. Notably, the maximum value of the noisy speech is at 118 samples, equivalent to the measured fundamental frequency changing within 3 Hz. Fig. 3(c) also shows that the value at 116 samples of the noisy speech is higher than the values of all interference peaks. This example demonstrates the robust noise resistance of the Radon transform in pitch estimation, which is the basis of our innovation.

D. GENERATING THE PITCH CANDIDATES

Next, we describe the selection of the pitch candidates from the energy function *pline*. Starting by partitioning *pline*, we choose the area of the pitch candidates based on the upper limit of pitch period l_{max} and the lower limit l_{min}. We first exclude the areas where gradient p is no more than l_{min} and more than l_{max} and then divide the remaining areas equally into intervals according to the given interval length N_{in}. When the number of samples in the remaining area is not divisible by N_{in}, we specify that the number of samples in each previous interval is kept as N_{in} except for the last interval. In Fig. 4, the red areas on the left and right sides represent the areas outside of $l_{min} \sim l_{max}$ in which we do not select the pitch candidates, and the middle area is divided into six intervals according to the given interval length N_{in}, with the first five intervals being of the same size and the last one being slightly smaller.

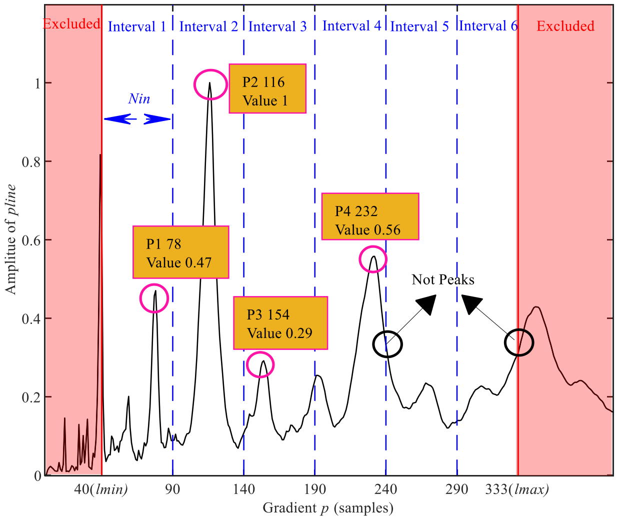


FIGURE 4. Selection of pitch candidates using the energy function with an interval length of 50.

The selection of the pitch candidates is performed as follows. We choose one pitch candidate in each interval. The range of each interval is left-open and right-closed. For each interval, we find the gradient p_m representing the location of the maximum value of the energy function *pline*. If $pline(p_m)$ is a peak, the gradient p_m will be noted as a pitch candidate of this speech frame, and the value of $pline(p_m)$ will be used for the Viterbi algorithm in Section III-D. Fig. 4 shows the process of selecting the pitch candidates. In Fig. 4, four pitch candidates, P_1, P_2, P_3, and P_4, are selected in the first four intervals, and their values are 0.47, 1, 0.29, and 0.56, while in the last two intervals, no pitch candidates are selected because the points where they have their maximum values are not peaks.

E. PITCH TRACKING WITH THE VITERBI ALGORITHM

Similar to BaNa [16] and Praat [20], our proposed algorithm also uses the Viterbi algorithm to select the pitch candidates of each frame to make a smooth and accurate pitch contour in one voiced segment. The Viterbi algorithm has an effect to make the pitch of the adjacent frames strongly related. We have already obtained the pitch candidates from one speech frame in Section III-C and will describe the pitch tracking method below.

We denote P_i^n as the ith pitch candidate of the nth frame of one voiced segment, and its energy function value is defined as V_i^n. Before calculating the path cost of the Viterbi algorithm, we need to calculate the *Cost* of each two pitch candidates between adjacent frames. We refer to BaNa using the same format to define the *Cost* function; the variables are substituted in our algorithm, as shown in Equation (8):

$$Cost(P_i^n, P_j^{n+1}) = \left| \log_2 \frac{P_i^n}{P_j^{n+1}} \right| + w \times \frac{1}{V_i^n} \tag{8}$$

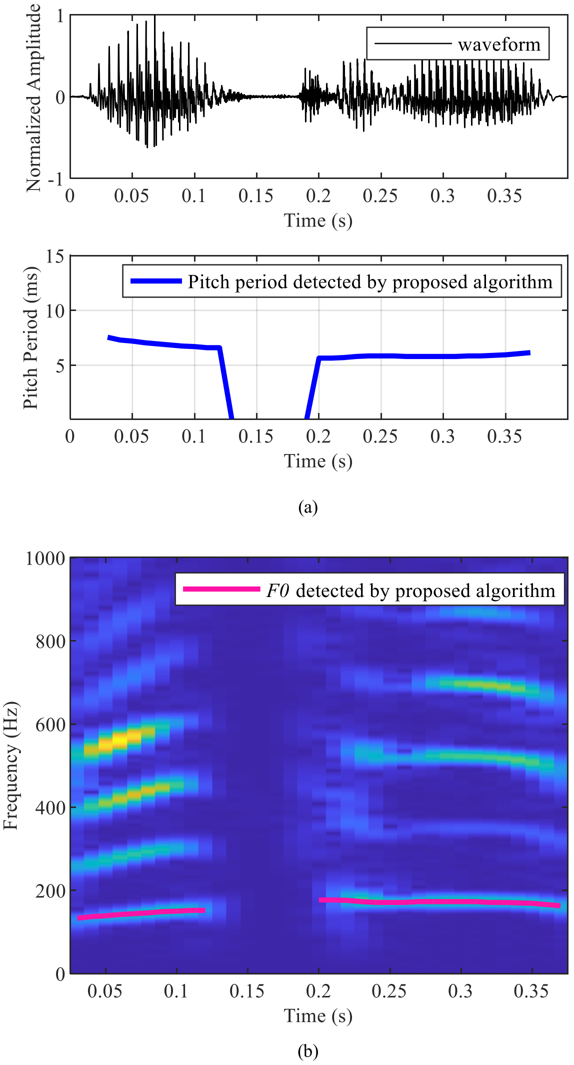


FIGURE 5. Detecting the pitch contours of a clean speech segment with the proposed algorithm: (a) shows the waveform and its pitch contours and (b) plots the fundamental frequency contours in the spectrogram.

where w represents the balance weight. The product of the weight w and the inverse of V_i^n denotes the *Cost* of P_i^n itself. A larger V_i^n deduces a smaller *Cost* of P_i^n, making it easier to pick out. The absolute value of the logarithm of the ratio of P_i^n to P_j^{n+1} indicates the similarity of the two pitch candidates between adjacent frames. The more similar these two candidates are, the less *Cost* their combination generates and the easier their combination will be selected. We replace the fundamental frequency candidates in BaNa with the pitch candidates and directly use the value of P_i^n to represent V_i^n, so we have a different choice of the weight w. We use the Viterbi algorithm provided in [38] to calculate the total path cost, as shown in Equation (9):

$$TCost(pa_k) = \sum_{n=1}^{N_{frame}-1} Cost(P_i^n, P_j^{n+1}) \tag{9}$$

where pa_k denotes the kth path found by the Viterbi algorithm and *TCost* denotes the sum of all values of the *Cost* on path pa_k. By comparing all paths, we select the one with the smallest *TCost* value as the path for the pitch tracking, which creates a smooth and accurate pitch contour and determines the final pitch choice for each frame individually.

Up to now, we have stated the generation of Radon transformed image, the energy function *pline*, the selection of the pitch candidates, and the generation of the pitch contour. In Fig. 5, we perform a complete pitch estimation process on a speech segment with the Radon transform and the Viterbi algorithm. Fig. 5(a) shows the smooth and accurate pitch contours we obtained. In Fig. 5(b), two red curves represent the fundamental frequency (F_0) contours. We compare F_0 contours with the ground-truth F_0 values in the spectrogram and find that the extracted F_0 contours can remain in the center of the ground-truth F_0 values, proving that our proposed algorithm is feasible and accurate.

IV. EXPERIMENTAL SETTINGS

A. PREPARATION OF THE DATABASES

Before the start of the experiment, we considered recording the noisy speech database in different scenarios directly. This approach is not feasible because we lack information about the ground-truth pitch values of each noisy speech, making it impossible to compare various algorithms adequately. To avoid this drawback, we collect the clean speech database and complex environmental noise database separately.

The clean speech database we collect is divided into three parts. The first part is the Keele database [14], which contains the speech and laryngograph information of fifteen speakers, from which we select five different male and five different female long sentences as part of our database, with a sampling rate of 16 kHz and a total duration of 337 s. The Keele database contains voiced/unvoiced information and ground-truth pitch values for reference. The second part is the CSTR database [15], which contains a large number of clean speech sentences and has information on the ground-truth pitch values estimated by seven algorithms, from which we select ten male sentences and ten female sentences with a sampling rate of 20 kHz and a total duration of 33.5 s. The third part is self-recorded speech files; we recorded four 48 kHz sampling rate clean male speech files in a quiet laboratory; each audio content is concise with no more than five words per sentence, and the total duration is 6 s.

To simulate actual projects, we choose the self-recorded environmental sounds to make our noise database instead of an existing noise database. We recorded the sounds with 18 scenarios in Hangzhou, China, including the restaurant, subway, airplane, street, canteen, night market, hospital, construction scenes, etc. The self-recorded noise database has a total duration of 773 s, which provides a sufficient guarantee for the subsequent generation of the noisy speech database. The collection of the databases is shown in Table 1.

We select six representative algorithms with excellent noise resistance among the non-data-driven algorithms according to [6] and [17] for comparison with the

TABLE 1. The collection of the databases with statistical characteristics.

Name	# of files	length	Sampling rate	Has ground-truth?	Relabeled?
Keele[14]	10	337 s	16 kHz	Yes	No
CSTR[15]	20	33.5 s	20 kHz	Yes	Yes
Recorded speech	4	6 s	48 kHz	No	Yes
Recorded noise	18	773 s	48 kHz		

proposed algorithm in this paper, which are Cepstrum [9], CAMDF [18], YIN [22], SWIPE' [26], BaNa [16] and Bayesian Pitch Tracking [17]. Then we choose HarmoF0 [36] as the representative of the state-of-the-art data-driven algorithm. We get the source code for Cepstrum and AMDF from [39] and modify the AMDF code to obtain the CAMDF algorithm code according to [18], and the source code for YIN, SWIPE', BaNa, Bayesian Pitch Tracking and HarmoF0 are derived from [40], [41], [42], [43], and [44] respectively.

B. OBTAINING GROUND-TRUTH PITCH VALUES AND NOISY SPEECH DATABASE

Both the Keele database [14] and the CSTR database [15] provide ground-truth pitch values, but due to algorithmic limitations, their selection of the voiced/unvoiced segments and calculation of the pitch periods are both deviated from the ones observed from the spectrogram.

In order to ensure the reliability of the experimental data, we only use the ground-truth pitch values provided by the Keele database, which has a considerable total duration, to benchmark against other algorithms. We align each valid ground-truth pitch value with the time step of each speech frame.

To enrich the experimental data and accurately reflect the impact of noise on the algorithms, we label the self-recorded speech files and relabel the CSTR database, which has a short total duration. First, accurate voiced/unvoiced information of clean speech is obtained using a method for voice activity detection that combines short-time energy and spectral entropy [45]. Then, we use SWIPE' and BaNa, which have the best performance in [6], to extract the pitch in the voiced segments. If these two algorithms extract the pitch period of a frame with an error within 10%, we take their average value as the ground-truth pitch value of this frame. Otherwise, we will hand-label the pitch period based on its fundamental frequency by measuring the spectrogram of the current frame.

Fig. 6 shows the spectrogram of a clean speech segment recorded in the laboratory, where the red and magenta circles represent the F_0 contours calculated by the proposed algorithm and BaNa, and the green circles represent the ground-truth F_0 contours after hand-labeling. In Fig. 6, the F_0 contours calculated by the proposed algorithm match most of the spectrogram. Table 1 also shows the condition of which database we hand-labeled.

Before conducting the experiments, we also need to produce the noisy speech database. According to the noise level in different scenarios, we divide the SNR from −10 dB to 20 dB into seven levels, namely −10 dB, −5 dB, 0 dB, 5 dB, 10 dB, 15 dB and 20 dB. We generate the noisy speech database by the following steps. First, we resampled 18 types of noise files and adjusted their length for each clean speech file. Then we synthesized the noisy speech database under seven SNR levels. Each clean speech file generated 126 noisy speech files, which formed the noisy speech database with a total of 4284 files.

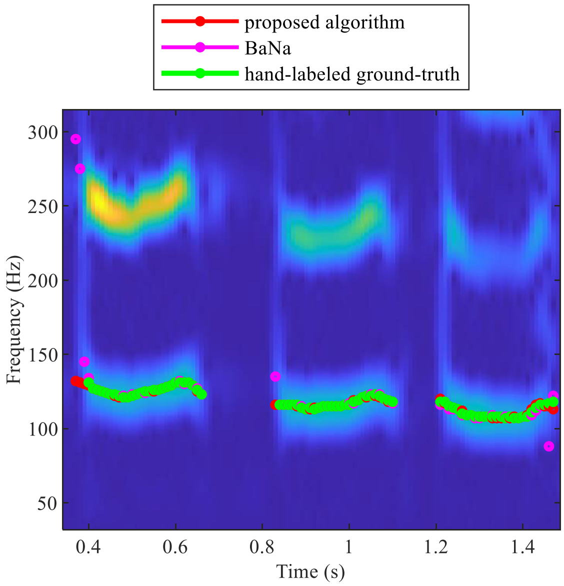


FIGURE 6. The ground-truth contours are labeled by combining SWIPE', BaNa, and the information of the spectrogram and compared with proposed algorithm.

C. THE EVALUATION METHOD FOR PITCH ESTIMATION

The Gross Pitch Error (GPE) rate, as a critical evaluation method for determining the accuracy of pitch estimation algorithms, has been applied in many state-of-the-art studies and is also applicable in this study. GPE rate reflects the detection rate of the algorithm by calculating the percentage of frames with false pitch periods in the voiced segment. As shown in Equation (10), the lower the GPE rate, the higher the algorithm's accuracy.

$$\mathrm{GPE} = \frac{N_{EV}}{N_V} \times 100\% \tag{10}$$

where N_V denotes the total number of frames to be evaluated in the voiced segment, and N_{EV} denotes the number of frames with false pitch periods. In addition, the voiced/unvoiced information is a key to the evaluation. Although the proposed algorithm does not involve voice activity detection, we have obtained the voiced/unvoiced information from Section IV-B. Therefore, it is appropriate to use this evaluation method for the algorithms in this study.

We need the information about the dissimilarity between the pitch value detected by the algorithm and the ground-truth pitch value to define an error frame. If their relative error exceeds the specified threshold, the measured frame is considered a frame with a false pitch period. Usually, the threshold is set to 20%; however, to demonstrate the noise resistance of the algorithms, we use a 10% threshold for the calculation, which is also used in the study of [16].

D. SETTINGS OF EXPERIMENTAL PARAMETERS

According to the effective range of pitch, we specify the minimum and maximum fundamental frequencies of the speech as 60 Hz and 500 Hz and calculate the upper limit l_{max} and lower limit l_{min} for each speech.

For all the non-parametric algorithms (Cepstrum, CAMDF, YIN, SWIPE', BaNa and proposed algorithm), we resample all noisy speech files to 16 kHz and set the frame length *win* to 60 ms to ensure each fragmented speech with short-time stability and each frame with the complete information of the pitch period. In addition, we make the time step s 10 ms to render the frames as continuous as possible.

For the parametric algorithm (Bayesian Pitch Tracking), we keep the default initializations to get the best performance [17]. The sampling rate of Bayesian Pitch Tracking in experiments is 16 kHz, the frame length *win* is 25 ms and the time step s is 10 ms. Considering that our noise database is made of self-recorded environmental sounds, Bayesian Pitch Tracking also needs a prewhitening step to deal with the inconsistency [17]. Thus, we compare Bayesian Pitch Tracking with and without prewhitening.

For the data-driven algorithm (HarmoF0), we use the default pre-trained checkpoint and the default initializations where the sampling rate is 16 kHz, the time step length is 160 points (10 ms) and the frame length is 1024 points (64 ms).

As shown in Table 2, we use the accuracy of the self-recorded database under SNR values from 0dB to 20dB to determine other experimental parameters. First, the spreading length N_{spread} must be set. After changing its value from 2 to 16, we find that when N_{spread} is 8, increasing its value will not improve the performance of the proposed algorithm, so we set the experimental value of N_{spread} to 8. Then, the interval length N_{in} is mentioned during the process of selecting pitch candidates. If N_{in} is long, the number of pitch candidates per frame is small, and the accuracy of the result is low. Conversely, the accuracy and computational complexity will be increased at the same time with a short N_{in}. We find that when N_{in} is set to about 1/2000 times the sampling rate Fs, the accuracy and computational complexity can achieve better results together; of course, we also need to ensure that N_{in} is an integer. Finally, a balance weight w is needed to calculate the *Cost* between the adjacent frames in the Viterbi algorithm. The smaller the w, the higher the continuity between the frames. However, a small w will not only increase the error rate of the proposed algorithm but also significantly degrade the rich variability of the expected results. We set the experimental value of w to 0.1 to get higher accuracy and enable the algorithm to obtain good results in most cases.

TABLE 2. Determination of parameter settings with self-recorded database.

Spreading length N_{spread}								
N_{spread}	2	4	6	**8**	10	12	14	16
Accuracy (%)	0.0	48.0	94.3	**95.9**	94.9	92.3	81.8	63.0
Interval length N_{in}								
N_{in}	$\frac{1}{250}Fs$	$\frac{1}{500}Fs$	$\frac{1}{1000}Fs$	$\frac{1}{2000}Fs$	$\frac{1}{3000}Fs$		$\frac{1}{4000}Fs$	
Accuracy (%)	93.1	94.6	95.3	**95.9**	95.9		95.9	
Balance weight w								
w	0.03	0.05	0.08	**0.1**	0.13	0.15	0.2	0.3
Accuracy (%)	94.6	95.3	95.7	**95.9**	95.7	95.7	95.3	95.0

V. PERFORMANCE FOR NOISY SPEECH

The performance of the proposed algorithm in complex noise environments is discussed by comparing the GPE rates of different algorithms under different speech databases, SNR values, and environmental noise conditions. Here we will elaborate on the two evaluation standards, SNR values and types of environmental noise.

A. CONTRIBUTION OF THE VARIOUS STEPS OF THE PROPOSED ALGORITHM

To represent the effects of the modified power function in Equation (5) and the Viterbi algorithm, we first compared the contributions of our proposed algorithm at various steps with the self-recorded noisy speech database under different SNR values.

Step 1: We modified the Radon transform only with the logarithmic function to ensure the achievability of the algorithm, while not using the power function for modification. Then, instead of selecting the pitch candidates with the Viterbi algorithm, we determine the pitches directly based on the location of the maximum value in the energy function for each frame. The result of Step 1 is represented by ”Proposed step1” in Fig. 7.

Step 2: Unlike Step 1, we add the power function modification, but again only determine the pitches based on the location of the maximum value in the energy function for each frame. The result of Step 2 is represented by ”Proposed step2” in Fig. 7.

Step 3: Based on Step 2, we use the method of selecting the pitch candidates with the Viterbi algorithm to find the pitch contours. This step performs the final result of our algorithm and the result of Step 3 is represented by the ”Proposed step3” in Fig. 7.

As shown in Fig. 7, after we added the power function modification, the GPE rates of our proposed algorithm under SNR values from −10dB to 20dB are significantly improved, with a reduction of nearly 20% at 0dB SNR. In addition, the GPE rates decrease further after we included the method of selecting pitch candidates with the Viterbi algorithm, which

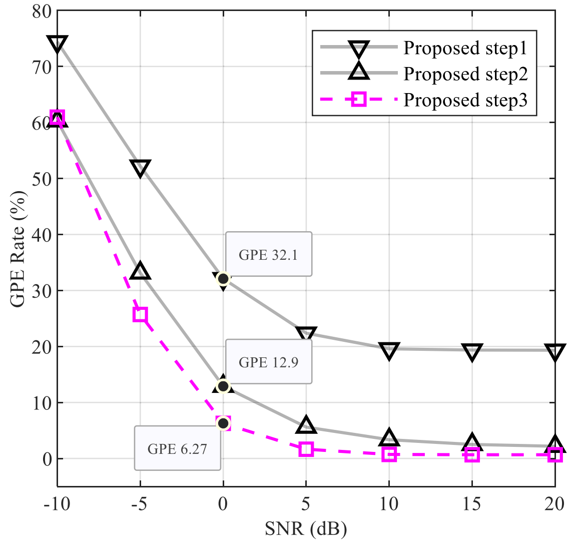


FIGURE 7. GPE rate of the three steps of the proposed algorithm under different SNR values.

proves that the effectiveness of our algorithm is improving step by step.

B. PERFORMANCE AT DIFFERENT SNR VALUES

Each clean speech corresponds to 18 noisy speech files under a given SNR value. To compare only the performance of algorithms under different SNR values, we bundle the GPE rates of each 18 noisy speech files to eliminate the impacts of noise.

Figs. 8 and 9 show the performance of the eight algorithms in detecting the pitch on the relabeled CSTR database and labeled self-recorded database, respectively. On the two databases we hand-labeled, CAMDF has the worst detection effect for the noisy speech under all SNR values; its GPE rate is up to 40% at 20 dB SNR. However, the stable performance of CAMDF under SNR values from 10 dB to 20 dB reflects its noise resistance to a certain degree. YIN and SWIPE' have good performance at 20 dB SNR. However, YIN and SWIPE' are very vulnerable to noise, and they have the same lousy detection rate when the SNR value is dropped, with the GPE rates around 85% and 80% on these two databases at −10 dB SNR. In addition, Cepstrum and BaNa, which have better noise resistance, perform better than CAMDF on these two databases. Cepstrum has higher overall accuracy on CSTR than YIN but lower overall accuracy on the self-recorded database. BaNa has a significant advantage over all previous non-parametric algorithms, both in terms of accuracy of pitch estimation and noise resistance, with a GPE rate of less than 4% at 20 dB SNR and less than 26% at 0 dB SNR. The parametric algorithm (Bayesian Pitch Tracking) with and without prewhitening are represented as Bayes-prew and Bayes-non in Figs. 8 and 9. It can be seen that this parametric algorithm has similar noise resistance to BaNa only when it has prewhitening. The data-driven algorithm HarmoF0 which has good performance on synthetic speech databases performs mediocrely on our real speech databases and is slightly inferior to BaNa and Bayes-prew. Figs. 8 and 9 show that our proposed algorithm outperforms the other six algorithms, with the lowest GPE rates under all SNR values. The GPE rate of the proposed algorithm is less than 10% at 0 dB SNR. Especially under SNR values from 5 dB to 20 dB, its GPE rate keeps under 4%. If we look at the rate of change of the curves, the proposed algorithm changes slowest under SNR values from 20 dB to 0 dB, proving its noise resistance is the best in this area.

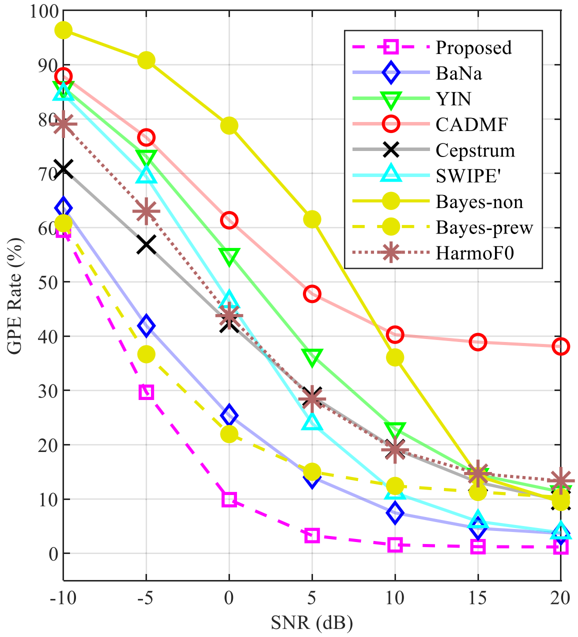


FIGURE 8. GPE rate on the relabeled CSTR database under different SNR values.

To make the experiment adequate and credible, we also need to compare the noise resistance performance of the seven algorithms on the non-relabeled database. As shown in Fig. 10, the GPE rates obtained from the non-relabeled Keele database are almost consistent with those from the hand-labeled databases. CAMDF remains the worst on this database, with Cepstrum, BaNa, HarmoF0, the proposed algorithm and Baysian Pitch Tracking with prewhitening performing in ascending order. It is worth mentioning that, by comparing the ground-truth pitch values provided by the Keele database with the spectrogram, we find that the Keele database not only labels the values of the unvoiced frames at the edges of voiced segments, but most of them are incorrect. We also find that some ground-truth pitch values of the voiced frames have 1/2-octave deviations. Therefore, it makes sense that the proposed algorithm, BaNa and Bayesian Pitch Tracking with prewhitening have higher GPE rates than YIN and SWIPE' at 20 dB SNR, as shown in Fig. 10. Conversely, even though SWIPE' has the lowest GPE rate at 20 dB SNR, its value is still higher than 8%, compared to less than 4% on the other two hand-labeled databases. This reflects the inaccuracy of the ground-truth pitch values

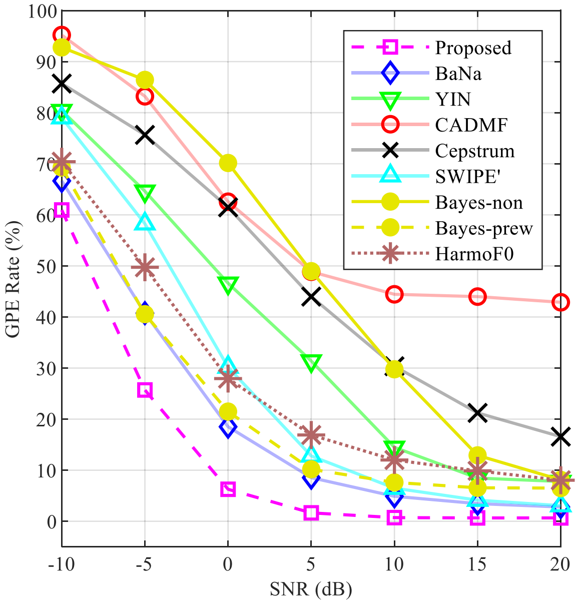


FIGURE 9. GPE rate on the labeled self-recorded database under different SNR values.

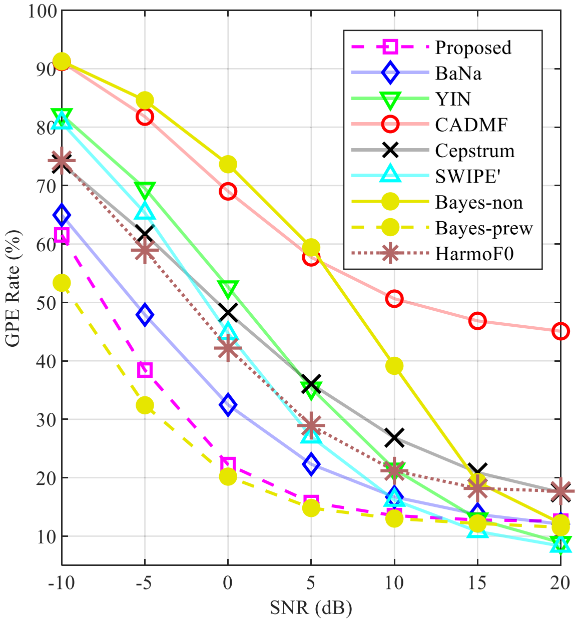


FIGURE 10. GPE rate on the non-relabeled Keele database under different SNR values.

provided by the Keele database. Although there are some problems with the non-relabeled Keele database, the robust noise resistance of our proposed algorithm can still be seen in Fig. 10, where it has the second lowest GPE rate under SNR values from −10 dB to 10 dB only higher than the parametric algorithm with prewhitening.

Although Bayesian Pitch Tracking with prewhitening is slightly better than the proposed algorithm on the non-relabeled Keele database under SNR values from −10 dB to 10 dB, its computational efficiency is much slower than the proposed algorithm. Table 3 gives the experimental results of the average running time comparison of Bayesian Pitch Tracking, BaNa, and the proposed algorithm for noisy speech files in Keele and CSTR database under the same conditions. Collectively, Fig. 10 and Table 3 show that for the Keele database, the proposed algorithm can achieve similar pitch estimation results with Bayesian Pitch Tracking with prewhitening with only one-tenth of its running time, which proves that our algorithm performs better in terms of computational efficiency and accuracy.

TABLE 3. Average running time comparison for noisy speech files in Keele and CSTR database.

CSTR: Average speech file time length = 1.68 s				
Algorithm	Proposed	BaNa	Bayes-non	Bayes-prew
Average running time (s)	1.74	3.10	5.33	16.44
Keele: Average speech file time length = 33.70 s				
Algorithm	Proposed	BaNa	Bayes-non	Bayes-prew
Average running time (s)	23.58	64.64	109.47	325.46

The values of each point in Figs. 8, 9, and 10 are shown in Table 4. By comparing the average running times and the pitch estimation results of each algorithm on the three databases under different SNR values, it can be concluded that our proposed algorithm performs best in aggregate under low SNR values, and its GPE rate can be maintained under 25% while BaNa achieves only 35% when detecting noisy speech at 0 dB SNR.

C. COMPARISON OF THE PROPOSED ALGORITHM WITH BaNa FOR DIFFERENT TYPES OF NOISE

In Section V-B, we can see that, except for the proposed algorithm, BaNa has the highest noise resistance and the best accuracy among non-parametric algorithms under SNR values from −10 dB to 5 dB. Therefore, in this section, we focus on comparing the proposed algorithm with BaNa for different types of noise at 0 dB SNR. Table 5 shows 18 types of noise and their characteristics. We use each of these 18 types of noise in our experiments to compare the noise resistance of the two algorithms.

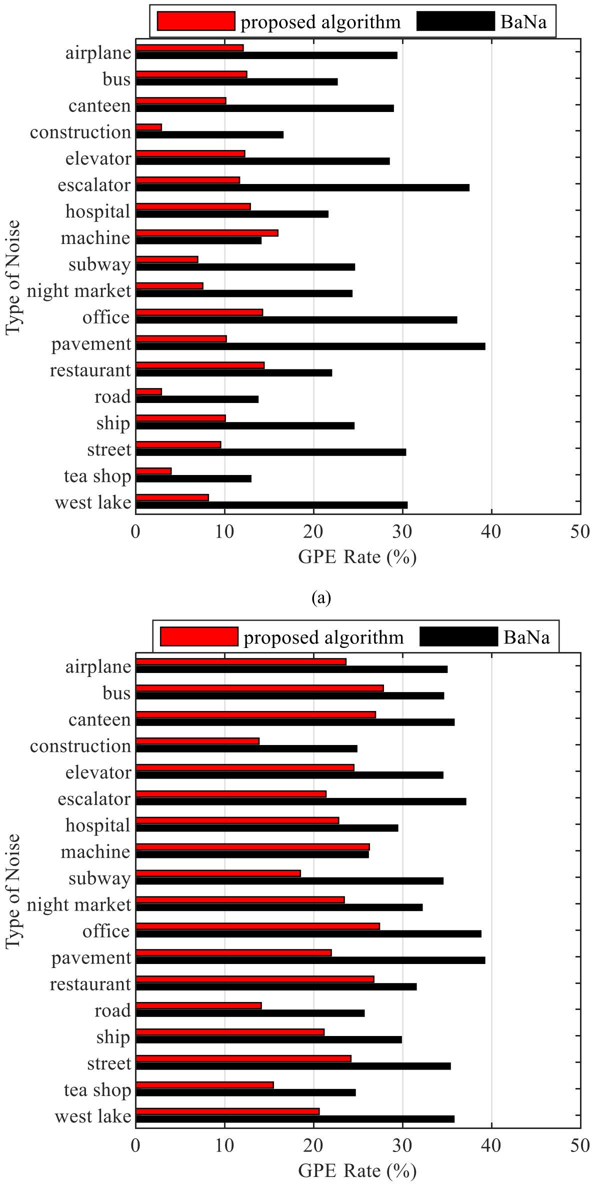
We conducted experiments with the relabeled CSTR database and the non-relabeled Keele database as a control group. Fig. 11(a) shows the performance of the proposed algorithm and BaNa on the CSTR dataset under different environmental noise conditions at 0 dB SNR. From the red bars in Fig. 11(a), we can see that the GPE rate of our proposed algorithm for all 18 types of noise can be kept under 20%, and its average GPE rate is 9.9%, while the GPE rate of BaNa for escalator, office and pavement maintenance scenarios exceeds 35%, with an average GPE rate of 25.4%. Only in the machine room scenario is the GPE rate of BaNa lower than that of the proposed algorithm; our algorithm has a significant advantage in the rest of the scenarios. The performance of both algorithms on the Keele database at 0 dB SNR is shown in Fig. 11(b). As the same results on the

TABLE 4. GPE rate (%) for three databases calculated by seven algorithms under different SNR conditions.

SNR (dB)	-10	-5	0	5	10	15	20
Relabeled CSTR database							
Proposed	59.5	29.7	9.9	3.3	1.6	1.2	1.2
BaNa	63.6	41.9	25.4	14.0	7.5	4.6	3.7
YIN	85.8	73.1	55.0	36.5	22.9	14.5	11.3
CAMDF	87.9	76.6	61.4	47.8	40.3	38.9	38.1
Cepstrum	70.8	56.9	42.4	28.9	19.2	13.1	9.8
SWIPE’	84.6	69.4	46.4	23.9	11.1	5.9	3.8
Bayes-non	96.4	90.8	78.8	61.5	36.1	14.2	9.5
Bayes-prew	60.8	36.7	22.0	15.0	12.4	11.3	10.3
HarmoF0	79.1	63.0	43.8	28.4	19.1	14.7	13.4
Labeled self-recorded database							
Proposed	61.0	25.7	6.3	1.7	0.7	0.7	0.7
BaNa	66.7	40.8	18.5	8.5	4.9	3.5	2.8
YIN	80.4	64.6	46.6	31.4	14.5	8.5	7.8
CAMDF	95.2	83.2	62.6	48.9	44.4	44.0	42.9
Cepstrum	85.7	75.7	61.5	44.0	30.4	21.2	16.6
SWIPE’	79.0	58.3	30.2	12.8	6.5	4.1	3.1
Bayes-non	92.8	86.4	70.2	49.0	29.8	12.9	8.1
Bayes-prew	69.3	40.6	21.5	10.2	7.6	6.6	6.5
HarmoF0	70.4	49.7	27.9	16.9	12.0	9.9	8.0
Non-relabeled Keele database							
Proposed	61.6	38.4	22.3	15.8	13.5	12.8	12.6
BaNa	65.0	47.9	32.5	22.3	16.7	13.7	12.1
YIN	82.1	69.5	52.6	35.3	21.5	12.9	8.9
CAMDF	91.2	81.8	69.0	57.7	50.6	46.8	45.1
Cepstrum	73.7	61.7	48.3	36.1	26.9	20.9	17.5
SWIPE’	80.8	65.3	44.7	27.0	16.2	10.8	8.3
Bayes-non	91.3	84.6	73.7	59.4	39.2	19.3	12.2
Bayes-prew	53.4	32.4	20.2	14.8	13.0	12.2	11.5
HarmoF0	74.3	59.0	42.2	28.9	21.2	18.2	17.7

TABLE 5. 18 types of noise and their characteristics.

Types	Characteristics
Airplane	Stationary noise, energy concentrated under 1000 Hz
Bus	Non-stationary noise, energy concentrated under 600 Hz, accompanied by 1000 Hz wind sounds
Canteen	Non-stationary noise, energy concentrated under 1000 Hz, with human voice interference
Construction Site	Intermittent full-band stationary noise
Elevator	Quasi-stationary noise, with energy concentrated under 300 Hz
Escalator	Quasi-stationary noise, with energy concentrated under 1200 Hz
Hospital	Non-stationary noise, energy concentrated under 1400 Hz, with music interference
Machine Room	Stationary noise, with energy concentrated at 250 Hz and 800 Hz and their harmonics
Subway	Stationary noise with energy concentrated under 1000 Hz
Night Market	Non-stationary noise, energy concentrated under 600 Hz, with human voice interference
Office	Non-stationary noise, energy concentrated at 200 Hz, with human voice interference
Pavement Maintenance	Continuous stationary noise under 500 Hz with intermittent full-band striking sound
Restaurant	Full-band non-stationary noise with human voice
Road	Quasi-stationary noise, with energy concentrated under 2600 Hz
Ship	Non-stationary noise, energy concentrated under 800 Hz, accompanied by 1000 Hz distinct noise
Street	Continuous stationary noise under 400 Hz with intermittent full band bump sound
Tea Shop	Continuous full-band stationary noise
West Lake	Non-stationary noise, energy concentrated under 300 Hz, with human voice and bird's sound



(a)

(b)

FIGURE 11. Performance of the two algorithms on two databases under different types of noise conditions at 0 dB SNR: (a) the relabeled CSTR database and (b) the non-relabeled Keele database.

CSTR database, the performance of the proposed algorithm in Fig. 11(b) is significantly more efficient than BaNa in all scenarios except for the machine room, and its GPE rate is kept less than 30% under all types of noise conditions, with an average GPE rate of 22.3%; in comparison, BaNa can only keep the GPE rate less than 40%, with its average GPE rate of 32.5%. At 0 dB SNR, the proposed algorithm has excellent results in pitch estimation under the influence of 18 types of noise and performs best for stationary environmental noise with uniform energy distribution like the construction site, road, and tea shop scenarios.

Through the performance of the proposed algorithm in Fig. 11, we need to focus on its weak noise resistance under the influence of some specific noise environments, including the airplane, bus, machine room, pavement maintenance scenarios, etc. As we know from Table 5, these types of noise have the characteristics of energy concentrated under 1000 Hz, indicating that there is still room to improve the resistance to low-frequency noise of the proposed algorithm.

VI. CONCLUSION AND FUTURE WORKS

In this paper, we apply the Radon transform to speech processing and propose a new pitch estimation algorithm combining the Viterbi algorithm to cope with the complex noise environments in actual projects. The Radon transform-based algorithm derives a more precise pitch from a noise-corrupted speech frame than other basic algorithms. We experimentally compare the performance of the proposed algorithm with Cepstrum, CAMDF, YIN, BaNa, SWIPE' and Bayesian Pitch Tracking on three clean speech databases and a self-recorded noise database. Results show that the proposed algorithm has the lowest GPE rate on the CSTR and self-recorded database under SNR values from −10 dB to 20 dB, its GPE rate changes minimally under SNR values from 5 dB to 20 dB. Additionally, the proposed algorithm can achieve similar detection results with Bayesian Pitch Tracking with prewhitening on the Keele database with only one-tenth of its running time, which proves that our algorithm performs best in aggregate by combining computational efficiency and accuracy.

In the future, we need further refinement for the proposed algorithm. Experiments show that our algorithm is susceptible to noise with energy concentrated under 1000 Hz, so we will focus on improving the resistance to low-frequency noise of the proposed algorithm under extremely low SNR values.

In summary, the proposed algorithm provides a novel and reliable pitch estimation algorithm for actual projects affected by complex noise environments and a fresh concept of analyzing noisy speech frames with the Radon transform.

REFERENCES

[1] H. Park, J.-Y. Yoon, J.-H. Kim, and E. Oh, "Improving perceptual quality of speech in a noisy environment by enhancing temporal envelope and pitch," *IEEE Signal Process. Lett.*, vol. 17, no. 5, pp. 489–492, May 2010.

[2] L. Buera, J. Droppo, and A. Acero, "Speech enhancement using a pitch predictive model," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Mar. 2008, pp. 4885–4888.

[3] S. Ananthakrishnan and S. Narayanan, "Improved speech recognition using acoustic and lexical correlates of pitch accent in a N-Best rescoring framework," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2007, pp. IV-873–IV-876.

[4] R. Sinha and S. Shahnawazuddin, "Assessment of pitch-adaptive front-end signal processing for children's speech recognition," *Comput. Speech Lang.*, vol. 48, pp. 103–121, Mar. 2018.

[5] B. Cardozo and R. Ritsma, "On the perception of imperfect periodicity," *IEEE Trans. Audio Electroacoustics*, vol. AE-16, no. 2, pp. 159–164, Jun. 1968.

[6] L. Sukhostat and Y. Imamverdiyev, "A comparative analysis of pitch detection methods under the influence of different noise conditions," *J. Voice*, vol. 29, no. 4, pp. 410–417, Jul. 2015.

[7] L. Rabiner, "On the use of autocorrelation analysis for pitch detection," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-25, no. 1, pp. 24–33, Feb. 1977.

[8] M. Ross, H. Shaffer, A. Cohen, R. Freudberg, and H. Manley, "Average magnitude difference function pitch extractor," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-22, no. 5, pp. 353–362, Oct. 1974.

[9] A. M. Noll, "Cepstrum pitch determination," *J. Acoust. Soc. Amer.*, vol. 41, no. 2, pp. 293–309, 1967.

[10] J. Radon, "Über die bestimmung von funktionen durch ihre integralwerte langs gewisser mannigfaltigkeiten," *Berichte Sachsische Acadamie der Wissenschaften, Leipzig, Math.-Phys. Kl.*, vol. 69, pp. 262–267, Apr. 1917.

[11] Q. Zhang, H. Wang, W. Chen, and G. Huang, "A local radon transform for seismic random noise attenuation," *J. Appl. Geophys.*, vol. 186, Mar. 2021, Art. no. 104264.

[12] B. M. Mahmmod, S. H. Abdulhussain, T. Suk, and A. Hussain, "Fast computation of Hahn polynomials for high order moments," *IEEE Access*, vol. 10, pp. 48719–48732, 2022.

[13] S. H. Abdulhussain, B. M. Mahmmod, T. Baker, and D. Al-Jumeily, "Fast and accurate computation of high-order tchebichef polynomials," *Concurrency Comput., Pract. Exper.*, vol. 34, no. 27, Dec. 2022, Art. no. e7311.

[14] F. Plante, G. F. Meyer, and W. A. Ainsworth, "A pitch extraction reference database," in *Proc. 4th Eur. Conf. Speech Commun. Technol. (Eurospeech)*, Sep. 1995, pp. 837–840.

[15] P. C. Bagshaw, S. Hiller, and M. A. Jack, "Enhanced pitch tracking and the processing of f0 contours for computer aided intonation teaching," in *Proc. 3rd Eur. Conf. Speech Commun. Technol. (Eurospeech)*, Sep. 1993, pp. 1003–1006.

[16] N. Yang, H. Ba, W. Cai, I. Demirkol, and W. Heinzelman, "BaNa: A noise resilient fundamental frequency detection algorithm for speech and music," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 22, no. 12, pp. 1833–1848, Dec. 2014.

[17] L. Shi, J. K. Nielsen, J. R. Jensen, M. A. Little, and M. G. Christensen, "Robust Bayesian pitch tracking based on the harmonic model," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 27, no. 11, pp. 1737–1751, Nov. 2019.

[18] X. Gang and T. Liang-Rui, "Speech pitch period estimation using circular AMDF," in *Proc. 14th IEEE Pers., Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2003, pp. 2452–2455.

[19] G. Muhammad, "Noise robust pitch detection based on extended AMDF," in *Proc. IEEE Int. Symp. Signal Process. Inf. Technol.*, Dec. 2008, pp. 133–138.

[20] P. Boersma, "Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound," *Proc. Inst. Phonetic Sci.*, vol. 17, pp. 97–110, Mar. 1993.

[21] D. Talkin, "A robust algorithm for pitch tracking (RAPT)," *Speech Coding Synth.*, vol. 495, p. 518, Nov. 1995.

[22] A. de Cheveigné and H. Kawahara, "YIN, a fundamental frequency estimator for speech and music," *J. Acoust. Soc. Amer.*, vol. 111, no. 4, pp. 1917–1930, 2002.

[23] M. Mauch and S. Dixon, "PYIN: A fundamental frequency estimator using probabilistic threshold distributions," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2014, pp. 659–663.

[24] J. Gauer, D. Kleingarn, and R. Martin, "Analysis and improvements of the cepstrum method for fundamental frequency estimation in music signals," in *Proc. 29th Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2021, pp. 371–375.

[25] M. R. Schroeder, "Period histogram and product spectrum: New methods for fundamental-frequency measurement," *J. Acoust. Soc. Amer.*, vol. 43, no. 4, pp. 829–834, Apr. 1968.

[26] A. Camacho and J. G. Harris, "A sawtooth waveform inspired pitch estimator for speech and music," *J. Acoust. Soc. Amer.*, vol. 124, pp. 1638–1652, Jun. 2008.

[27] S. Gonzalez and M. Brookes, "A pitch estimation filter robust to high levels of noise (PEFAC)," in *Proc. Eur. Signal Process. Conf.*, Barcelona, Spain, 2011, pp. 451–455.

[28] K. Kasi and S. A. Zahorian, "Yet another algorithm for pitch tracking," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, May 2002, pp. I-361–I-364.

[29] M. G. Christensen, "Accurate estimation of low fundamental frequencies from real-valued measurements," *IEEE Trans. Audio, Speech, Language Process.*, vol. 21, no. 10, pp. 2042–2056, Oct. 2013.

[30] J. K. Nielsen, T. L. Jensen, J. R. Jensen, M. G. Christensen, and S. H. Jensen, "Fast fundamental frequency estimation: Making a statistically efficient estimator computationally efficient," *Signal Process.*, vol. 135, pp. 188–197, Jun. 2017.

[31] F. Huang and T. Lee, "Pitch estimation in noisy speech using accumulated peak spectrum and sparse estimation technique," *IEEE Trans. Audio, Speech, Language Process.*, vol. 21, no. 1, pp. 99–109, Jan. 2013.

[32] W. Chu and A. Alwan, "SAFE: A statistical approach to f0 estimation under clean and noisy conditions," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 3, pp. 933–944, Mar. 2012.

[33] J. W. Kim, J. Salamon, P. Li, and J. P. Bello, "Crepe: A convolutional representation for pitch estimation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 161–165.

[34] B. Gfeller, C. Frank, D. Roblek, M. Sharifi, M. Tagliasacchi, and M. Velimirovic, "SPICE: Self-supervised pitch estimation," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 28, pp. 1118–1128, 2020.

[35] S. Singh, R. Wang, and Y. Qiu, "DeepF0: End-to-end fundamental frequency estimation for music and speech signals," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 61–65.

[36] W. Wei, P. Li, Y. Yu, and W. Li, "HarmoF0: Logarithmic scale dilated convolution for pitch estimation," in *Proc. IEEE Int. Conf. Multimedia Expo. (ICME)*, Jul. 2022, pp. 1–6.

[37] G. Beylkin, "Discrete radon transform," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, no. 2, pp. 162–172, Feb. 1987.

[38] P. van Alphen and D. van Bergem, "Markov models and their application in speech recognition," in *Proc. Inst. Phon. Sci. Univ. Amsterdam*, 1989, pp. 1–26.

[39] L. R. Rabiner and R. W. Schafer, *Theory and Application of Digital Speech Processing*. Upper Saddle River, NJ, USA: Prentice-Hall, 2010.

[40] *Source Code for the YIN Algorithm*. Accessed: Apr. 15, 2022. [Online]. Available: http://audition.ens.fr/adc/

[41] *Source Code for the SWIPE' Algorithm*. Accessed: Jun. 6, 2022. [Online]. Available: http://www.cise.ufl.edu/acamacho/publications/swipep.m

[42] *BaNa Source Code, WCNG Website*. Accessed: Apr. 20, 2022. [Online]. Available: http://www.ece.rochester.edu/projects/wcng/project_bridge.html

[43] *Source Code for the Bayesian Pitch Tracking Algorithm*. Accessed: Nov. 10, 2022. [Online]. Available: https://github.com/LimingShi

[44] *Source Code for the HarmoF0 Algorithm*. Accessed: Dec. 15, 2022. [Online]. Available: https://github.com/WX-Wei/HarmoF0

[45] T. H. Zaw and N. War, "The combination of spectral entropy, zero crossing rate, short time energy and linear prediction error for voice activity detection," in *Proc. 20th Int. Conf. Comput. Inf. Technol. (ICCIT)*, Dec. 2017, pp. 1–5.

BAI LI received the B.S. degree in communication engineering from the Dalian University of Technology, Dalian, China, in 2017. He is currently pursuing the M.S. degree in electronic and communication engineering with the School of Information Science and Engineering, Xinjiang University, Xinjiang, China. His current research interests include speech signal processing, speech recognition, and speech enhancement.

XIANWU ZHANG received the B.S. and M.S. degrees from the School of Earth Exploration Science and Technology, Jilin University, China, and the Ph.D. degree from the University of Chinese Academy of Sciences, China. Since 2020, he has been a Graduate Supervisor with the School of Information Science and Engineering, Xinjiang University, Xinjiang, China. His research interests include radar, seismic imaging technology, and speech signal processing.