Pitch Contour Stylization by Marking Voice Intonation

Sakshi Pandey[1], Amit Banerjee[2], Subramaniam Khedika[3]
Computer Science Department
South Asian University
New Delhi, India

***Abstract*—The stylization of pitch contour is a primary task in the speech prosody for the development of a linguistic model. The stylization of pitch contour is performed either by statistical learning or statistical analysis. The recent statistical learning models require a large amount of data for training purposes and rely on complex machine learning algorithms. Whereas, the statistical analysis methods perform stylization based on the shape of the contour and require further processing to capture the voice intonations of the speaker. The objective of this paper is to devise a low-complexity transcription algorithm for the stylization of pitch contour based on the voice intonation of a speaker. For this, we propose to use of pitch marks as a subset of points for the stylization of the pitch contour. The pitch marks are the instance of glottal closure in a speech waveform that captures characteristics of speech uttered by a speaker. The selected subset can interpolate the shape of the pitch contour and acts as a template to capture the intonation of a speaker's voice, which can be used for designing applications in speech synthesis and speech morphing. The algorithm balances the quality of the stylized curve and its cost in terms of the number of data points used. We evaluate the performance of the proposed algorithm using the mean square error and the number of lines used for fitting the pitch contour. Furthermore, we perform a comparison with other existing stylization algorithms using the LibriSpeech ASR corpus.**

Keywords*—Pitch contour; pitch marking; linear stylization; straight-line approximation

I. Introduction

Speech prosody represents the pitch contour of a voice signal and can be used for the construction of linguistic models and their interaction with other linguistic domains, such as morphing and speech transformation [1]. In addition, the pitch contours are used for learning generative models for text-to-speech synthesis applications [2], language identification [3], emotion prediction and for forensics research [4]. Researchers have also used pitch and intensity of sound for predicting the mood of a speaker [5]. In order to remove the variability in the pitch contour, *stylization* is used to encode the contour into meaningful labels [6] or templates [7] for speech synthesis application. According to [8], stylization is a process of representing the pitch contour of the audio signal with a minimum number of line segments, such that the original pitch contour is auditorily indistinguishable from the re-synthesized pitch contour.

Broadly, the stylization of pitch contour either uses statistical learning or statistical analysis models. In *statistical analysis models*, the pitch contour is decomposed into a set of previously defined functions such as polynomial [9], [10], parabolic [11], and B-splines [12]. In addition, low-pass filtering is also used for preserving the slow time variations in the pitch contours [6]. Recently, researchers have studied the *statistical learning models*, using hierarchically structured deep neural networks for modeling the F0 trajectories [13] and sparse coding algorithm based on deep learning auto-encoders [14]. In general, the statistical learning models require a large amount of data and uses complex machine learning algorithms for training purposes [13], [14]. On the other hand, the statistical analysis models decompose the pitch contours as a set of functions based on the shape and structure of the contour that requires further processing to capture voice intonations of the speaker [9]–[12], [15]. Table I summarizes the algorithms proposed for the stylization of pitch contour. Many successful speech applications use piecewise stylization of the pitch, including the study of sentence boundary [16], dis-fluency [17], dialogue act [18], and speaker verification [19].

In this paper, we use statistical analysis for piecewise decomposition of the pitch contour using the instance of glottal closure or pitch marks to stylize the pitch contour as well as capture the intonation of the speaker's voice. As mentioned above, the previous works based on the statistical analysis approach [6], [9]–[12], mainly consider the shape and structure of the contour for stylization. For example, [12] use best-fit B-splines to define the segments of a pitch contour, and [11] uses parabolic functions to approximate the pitch contour. In contrary to these approaches, in this paper, we try to model the instances of glottal closure (pitch marks) of the source speaker. An advantage of the proposed approach is that the pitch marks can be used directly as templates for speech synthesis or speech morphing, making the approach suitable for various real-time applications.

The piecewise stylization approximates the pitch contour using K subset points. That is, if we let $\{y_n\}_{n=1}^{N}$ to be the pitch at each instant of time in a speech signal then the piecewise stylization can be defined using function 1, where $g(y)$ is the stylized pitch, a_i and b_i are the slope and intercept of each line at each y time instant and K is the subset size required for the stylization of the speech signal. In this paper, we select the pitch marks as a subset of points for the reconstruction of the pitch contour. These pitch marks are selected to fit the pitch contour for capturing large-scale variations. For this, we propose an algorithm using pitch marks as the subset points for the stylization of the pitch contour. The proposed algorithm can be used for retrieving the pitch marks from the voiced region of a pitch contour. In addition, it can

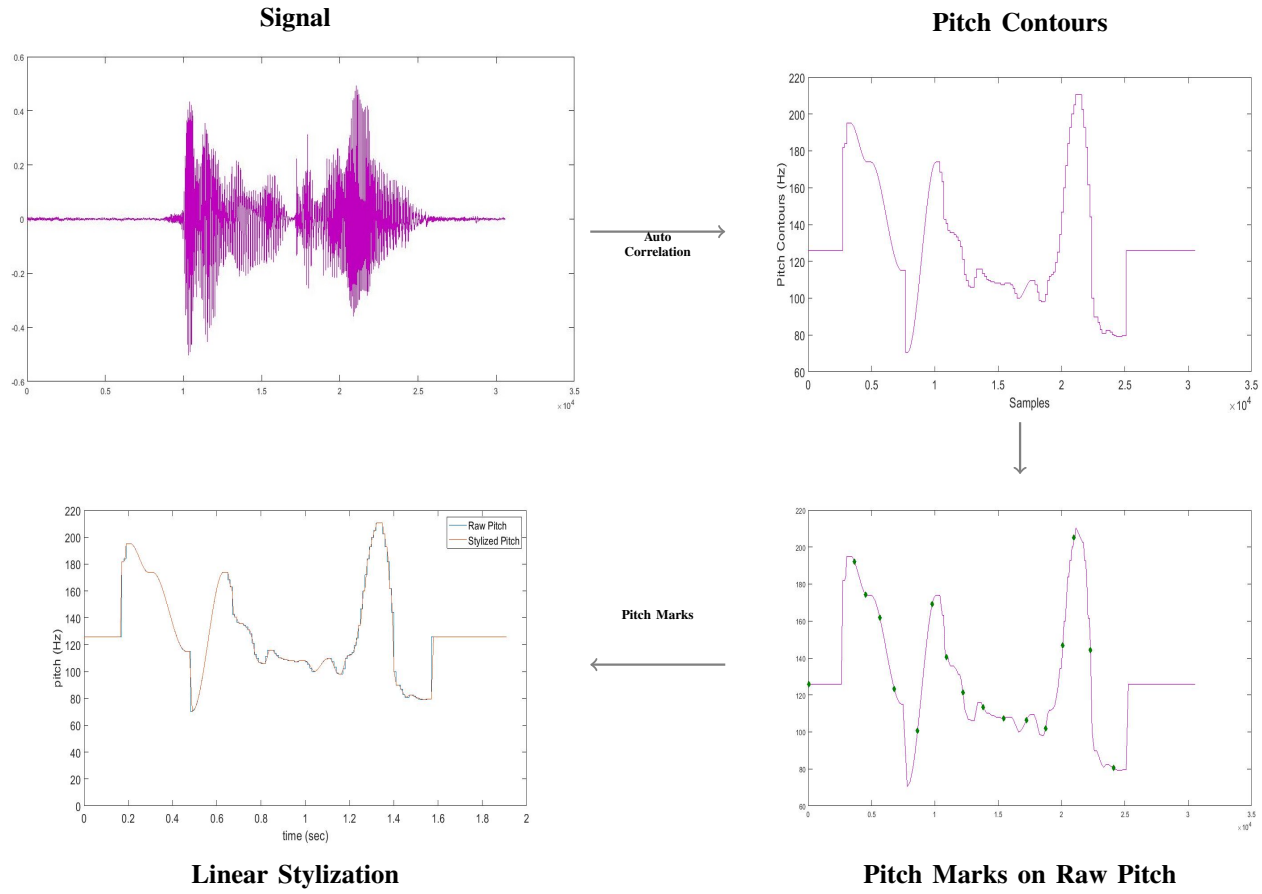


Fig. 1. Block Diagram of Proposed Method.

stylize the voiced and unvoiced region of the contours after pitch smoothing, which can be apt for applications mentioned above and for text-to-speech conversion [24], [25]. The general flow of the proposed methodology on a smoothed pitch contour is shown in Fig. 1. As shown in the figure, the approach uses auto-correlation to detect pitch and uses median filtering with length-3-window to remove sudden spikes to generate the corresponding pitch contour. This is used for extracting the pitch marks and to approximate the pitch contour using linear interpolation. The number of the linear segment depends on the number of pitch marks in the speech signal.

$$g(y) = \sum_{i=1}^{K-1} \sum_{j=i}^{i+1} (a_i y_j + b_i) \tag{1}$$

The proposed work is closely related to [10], [15]. In [10], the authors discuss a computationally efficient dynamic programming solution for the stylization of pitch contour. The approach calculates the MSE (mean square error) of the stylized pitch by predetermining the number of segments K using [15]. The authors in [15], use Daubechies wavelet (Db10) to perform a multilevel decomposition of the pitch contour and use third-level decomposition to extract the number of extremes (K) for the stylization. The choice for the third level is based upon the empirically tested results, which show the best result for 60% of the cases. However, for the same data 29% of the cases show better results for higher wavelet decompositions or fewer segments, and 11% of the cases have better performance for second level decomposition. On contrary, in our approach, the number of segments is determined by the intonation of the speaker's voice and no pre-determination is required for the same. That is, the number of segments required for pitch stylization is neither pre-determined nor depends on any empirical result. The algorithm computes the optimal number of segments based on the change in the pitch trajectory of the speaker.

To understand the performance of the proposed algorithm, we analyze matrices such as mean squared error (MSE) and the number of line segments (K) used for stylization. For our analysis, we use voice samples from the LibriSpeech ASR corpus [26] and the EUSTACE speech corpus [27] to compare the performance with [15]. The experimental results show that in comparison to [15], the proposed methodology uses less number of lines (K) to represent the pitch contour of a speech signal. Also, the proposed approach has a lower MSE, in comparison to stylization via wavelet decomposition [15].

The rest of the paper is organized as follows. Section II presents the related work. Section III presents the methodology of the proposed piecewise linear stylization approach. In Section IV, we discuss the experimental setup and simulation

TABLE I. SUMMARY ON EXISTING WORK ON PITCH CONTOUR STYLIZATION

Works	Approach	Algorithm	Application
Xiang Yin et.al. [2016] [13]	Stat. Learning	Hierarchically structured deep neural networks	Statistical parametric speech synthesis
Nicolas Obin et.al, [2018] [14]	Stat. Learning	Deep Auto-Encoders	Learning pitch templates for synthesis and voice conversion.
J't Hart el.at. [1991] [11]	Stat. Analysis	Piecewise stylization	Parabolas's adequate for F0 approximations
Daniel Hirst el.at. [1993] [12]	Stat. Analysis	Stylization using quadratic spline function	Coding and synthesis of curve used for different languages.
D'Alessandro el. at. [1995] [20]	Stat. Analysis	Perceptual model of intonation	Prosodic analysis and speech synthesis
Nygaard el.at. [1998] [9]	Stat. Analysis	Piecewise polynomial approximation	Electrocardiogram (ECG)
Dagen Wang el. at. [2005] [15]	Stat. Analysis	Piecewise stylization via wavelet analysis	Pitch stylization for spoken languages
Prashant K. Gosh el.at.[2009] [10]	Stat. Analysis	Polynomial approximation via dynamic programming	Pitch stylization
Origlia A. el.at.[2011] [21]	Stat. Analysis	Divide and conquer approach	Pitch stylization
Yadav O. P. el.at.[2019] [22]	Stat. Analysis	Piecewise approximation via Chebyshev polynomial	Electrocardiogram (ECG)
Yadav O. P. el.at.[2019] [23]	Stat. Analysis	Chebyshev nodes used for Lagrange interpolation	Electrocardiogram (ECG)
This paper	Stat. Analysis	Piecewise approximation via Pitch Marks	Pitch stylization

results. Finally, Section V concludes the paper.

II. RELATED WORKS

Pitch Stylization is the process of retrieving pitch contours of an audio signal using linear or polynomial functions, without affecting any perceptually relevant properties of the pitch contours. Broadly, the stylization of pitch contour either uses statistical learning or statistical analysis models. Table I, summarizes the stylization algorithms to show the current state-of-art. In the following, we discuss these approaches in detail.

A. Stylization using Statistical Learning

Recently, researchers used statistical learning models for pitch contour stylization. In [13], the author uses deep neural networks (DNN) to consider the intrinsic F0 property for modeling the F0 trajectories for statistical parametric speech synthesis. The approach embodies the long-term F0 property by parametrization of the F0 trajectories using optimized discrete cosine transform (DCT) analysis. Two different structural arrangements of a DNN group, namely cascade, and parallel, are compared to study the contributions of context features at different prosodic levels of the F0 trajectory. The authors in [14] propose a sparse coding algorithm based on deep-auto encoders for the stylization and clustering of the pitch contour. The approach learns a set of pitch templates for the approximation of the pitch contour. However, both these approaches use a large data set for training and may not be applicable for stylizing unknown audio samples.

B. Stylization using Statistical Analysis

In contrary to the previous approaches, statistical analysis models have low computational complexity and can be used for unknown audio samples. This is a well-studied technique for stylization and researchers are actively proposing newer methods for optimally approximating signals. In [11], authors introduce the concept of piecewise approximation of F0 curve using fragments of a parabola and perform stylization of the contour via rectilinear approximation. Similarly, authors in [12], propose a model for the approximation of fundamental frequency curves that incorporates both coding and synthesis of pitch contours using quadratic spline function. The model is applied for the analysis of fundamental frequency curves in several languages including English, French, Spanish, Italian and Arabic. The author in [20] discuss a new quantitative model of tonal perception for continuous speech. In this, the authors discuss automatic stylization of pitch contour with applications to prosodic analysis and speech synthesis.

In [9] the authors discuss piecewise polynomial approximation for the ECG signals. The paper uses second-order polynomials for reconstructing the signal with minimum error. The authors show that the method outperforms the linear interpolation method in various cases. The concept of polynomial interpolation is applied for the pitch contour stylization in [10]. The paper proposes an efficient dynamic programming solution for the pitch contour stylization with the complexity of $O(KN^2)$. It calculates the MSE (mean square error) of the stylized pitch by predetermining the number of segments K using [15]. The authors in [15], use Daubechies wavelet (Db10) to perform a multilevel decomposition of the pitch contour and use third-level decomposition to extract the number of extremes (K) for stylization. The choice for the third level is based upon the empirically testing, showing the best result for 60% of the cases. For remaining cases, 29% shows the better result on higher wavelet decompositions or fewer segments, and 11% of the cases have better performance for second level decomposition. The author in [21] proposes a divide and conquer approach for pitch stylization to balance the number of control points required for the approximation. Recently, in [22], authors used bottom-up time series for the segmentation of the signal, and the restoration is performed using the Chebyshev polynomials. An improvement to the approach

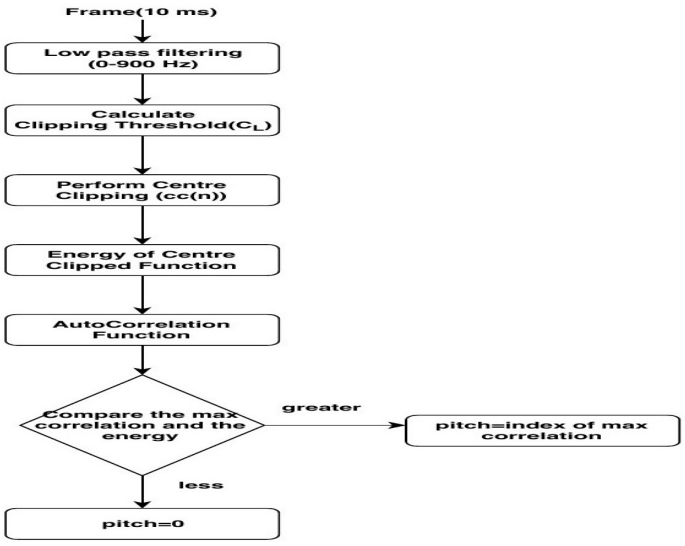


Fig. 2. Pitch Detection.

is proposed by the authors in [23], where the Chebyshev nodes are used for the segmentation of the signal and the approximation is performed using Lagrange interpolation.

C. Summary

In the proposed algorithm, we use statistical analysis for stylization. Unlike previous works, the number of segments is determined by the intonation of the speaker's voice and no pre-determination is required for the same. That is, the number of segments required for pitch stylization is neither pre-determined nor depends on any empirical result. The algorithm computes the number of segments based on the changes in the pitch trajectory of the speaker. The pitch marks are used for the linear stylization of the contour. The purpose of choosing pitch marks as the subset is to capture the intonation of the speaker in the pitch contour, which can further be used for various other applications like voice morphing, dubbing and can also act as an input to [9].

III. Proposed Methodology

The process of pitch stylization is divided into three steps: (1) pitch (F_0) determination, (2) pitch marking, and (3) linear stylization. In the following, we discuss these steps in detail.

A. Pitch Determination

Pitch determination is a process of determining the fundamental frequency or the fundamental period duration [28]. Pitch period is directly related to speaker's vocal cord and is used for speaker identification [4], emotion prediction [5], real-time speaker count problem [29]–[31]. This is one of the fundamental operations performed in any speech processing application. Researchers have proposed various algorithms for pitch determination, including YAAPT [32], Wu [33], SAcC [34]. However, in this paper, we are using the auto-correlation technique for the same.

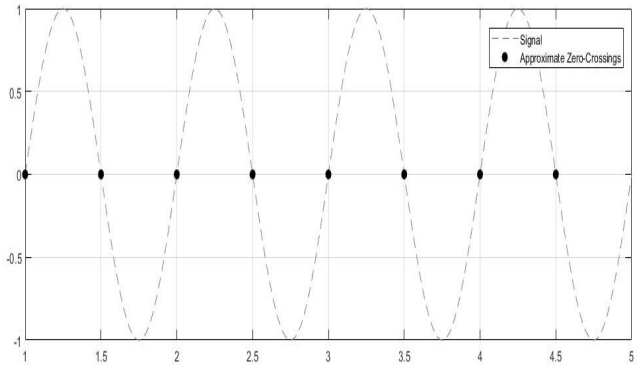


Fig. 3. Zero Crossing Points.

For pitch determination, we first perform low-pass filtering with a passband frequency of 900 Hz. As the fundamental frequency ranges between 80-500 Hz, the frequency components above 500 Hz can be discarded for pitch detection. In order to remove the formant frequencies in the speech signal and to retain the periodicity, center clipping is performed using a clipping threshold (C_L) [35]. We choose 30% of the max amplitude as C_L. We use equation 2 for center clipping, where $x(n)$ is speech signal and $cc(n)$ is the center clipped signal.

$$cc(n) = \begin{cases} x(n) - C_L & \text{if } x(n) > C_L \\ x(n) + C_L & \text{if } x(n) < -C_L \end{cases} \tag{2}$$

Furthermore, the energy of the center-clipped signal can be evaluated using equation 3. This can be used for determining the voiced and unvoiced regions in the pitch contour.

$$E_s = \sum_{n=1}^{N} |x(n)|^2 \tag{3}$$

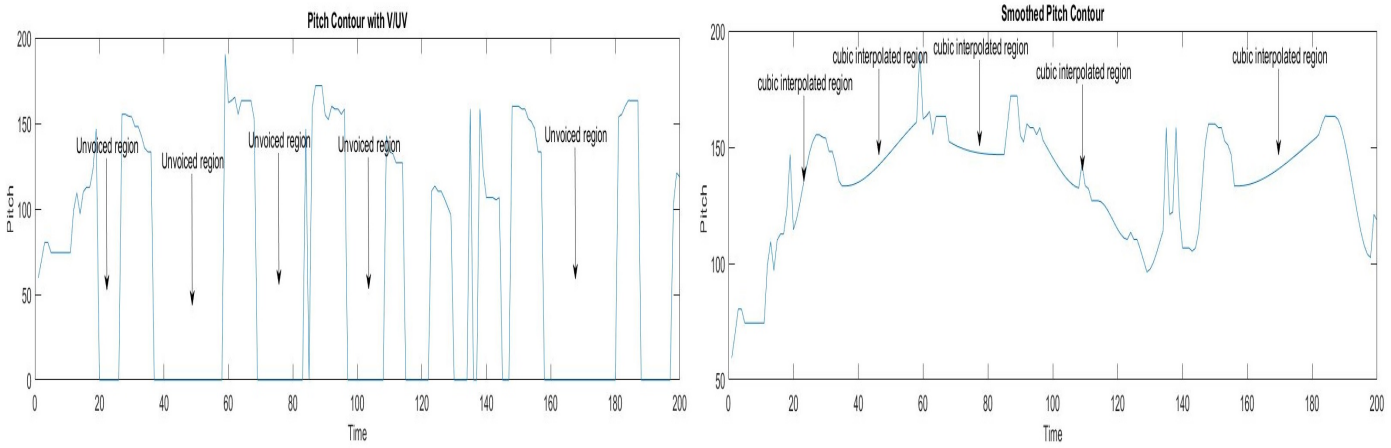
Finally, we use the autocorrelation method to detect the periodicity of a speech signal. The frame size used for pitch estimation is 10 ms. For a speech signal, autocorrelation measures the similarity of the signal with itself with a time lag. Given a discrete-time speech signal $x(n), n \in [0, N-1]$ of length N and τ as the time lag, the autocorrelation can be defined as the following.

$$R(\tau) = \sum_{n=0}^{N-1-\tau} x(n)x(n+\tau) \quad \tau \in [0, 1, \ldots, N-1] \tag{4}$$

We compare the energy E_s to the maximum correlation value, to determine the pitch of the frame. Fig. 2 gives the flowchart of the steps followed. This step generates the pitch contour $pcont$ corresponding to a speech signal.

B. Pitch Marking

A pitch mark can be defined as an instance of the glottal closures in a speech waveform. Previously, researchers have used pitch marks for various applications, such as voice transformation and pitch contour mapping [36]. However, in this paper, we are using pitch marks for pitch contour stylization. The following steps are used for generating the pitch marks.



(a) Pitch Contour

(b) Pitch Contour after Interpolation

Fig. 4. Smoothed Pitch Contour.

Algorithm 1 Extract Pitch Marks (p_{start}, p_{end})

1: Low pass filtering with cutoff frequency $500Hz$
2: Reverse the signal again perform low pass filtering
3: High pass filtering with cutoff frequency $150Hz$
4: Reverse the signal again perform high pass filtering
5: The delta function is used to differentiate the filtered signal.
6: The delta signal is again double low pass filtered to remove any noise or phase differences.
7: find the zero crossing points.

Algorithm 2 Pitch Marking for Voiced Region

1: Extract voiced (P_v) and unvoiced (P_{uv}) segments from the pitch contour
▷ For each ith segment $2i$ is the starting point and $2i+1$ is the end point
2: **for** each i-th voiced segment in (P_v) **do**
3: p_{start} = $get\_start\_point(i)$
4: p_{end} = $get\_end\_point(i)$
5: S_v = Extract pitch marks (p_{start}, p_{end})
6: **end for**
7: **for** each i-th unvoiced segment in (P_{uv}) **do**
8: p_{start} = $get\_start\_point(i)$
9: p_{end} = $get\_end\_point(i)$
10: $S_{uv} \leftarrow$ Append p_{start}, p_{end} to the list.
11: **end for**
12: $pitchMarks \leftarrow$ MERGE (S_v, S_{uv}) ▷ Merge two sorted lists in $O(n)$

Algorithm 1, is used for pitch marking. In the algorithm, we first perform low pass double filtering. It is a process where the first filtered waveform is reversed and fed again to the filter to diminish the phase difference between the input and output of the filter. Subsequently, double high pass filtering is performed to lessen the phase shifts, followed by the application of the delta function for differentiating the filtered signal. The delta signal is again passed through a double low pass filter to remove any noise or phase differences. The zero-crossing points are considered as the pitch marks. Zero crossings are points where the signal changes from positive to negative or vice-versa. Fig. 3 marks the zero-crossing points of a simple sine wave.

Algorithm 3 Pitch Marking after Smoothing

1: $smooth\_pcont \leftarrow ptch\_fix(pcont)$
2: $fsize \leftarrow$size of the frame, $f_s * t$
3: $nof \leftarrow$ number of frames of frame size $fsize$
4: $temp \leftarrow 0$
5: **for** $i \leftarrow 1$ to nof **do**
6: $range \leftarrow temp + 1 : temp + fsize$
7: $pitchMarks \leftarrow$ find pitch marks in each frame from $smooth\_pcont(range)$
8: $temp \leftarrow temp + fsize$
9: **end for**

The pitch marks are a compact representation of the pitch contour. By knowing the position of pitch marks, a very accurate estimation of f0 contour can be obtained, which can be further utilized for various speech analysis and processing methods [37]. Next, we use Algorithm 1 for determining the pitch marks from the pitch contour $(pcont)$ for the following two cases.

1) Pitch marking for voiced region: In this approach, we extract the pitch marks from the voiced regions. The classification of the voiced and unvoiced regions can be determined by using the values of $pcont$, as the unvoiced regions are marked by zero pitch values. Fig. 4 shows the voiced and the unvoiced regions in the pitch contour. The unvoiced region is marked by black arrows and has zero value. On the other hand, the non-zero values represent the voiced regions, where the pitch marking is performed. For each unvoiced region, we store the first and the last data points in the $pitchMarks$. The steps followed for pitch marking are shown in Algorithm 2. In the algorithm, for each i^{th} voice segment, we extract the pitch marks using Algorithm 1. The extracted pitch marks of the voiced region are stored in S_v (step 5). Similarly, the starting

Algorithm 4 Linear Stylization Algorithm

```
1: i ← 1
2: while i <= length(pitchMarks)-1 do
3:     slopes(i) ← slope of the points i and i + 1
4:     i ← i + 1
5: end while
6: for i ← 1 to length(slopes) do
7:     p ← pitchMarks(i)
8:     q ← pitchMarks(i + 1)
9:     k ←1
10:    for p to q do
11:        y = slope(i) ∗ k + p
12:        k ← k + 1
13:    end for
14:    y is the stylized pitch contours
15: end for
```

and end time instance of the unvoiced regions are stored in S_{uv} (step 10). Finally, the two lists, i.e., S_v and S_{uv} are merged. As the lists are sorted, the run-time complexity for merging is $O(n)$, where n is the maximum number of elements in both lists.

2) Pitch marking after smoothing: Above, the pitch marks are extracted only from voiced frames. As an extension, the unvoiced regions in the pitch contour are interpolated to generate a smoothed pitch contour. The shape-preserving piecewise cubic interpolation is performed in each segment and then median filtering is performed to get the new pitch contour. Fig. 5 shows the smoothed pitch contour. The generated pitch contours are segmented and pitch marks in each segment are stored. The steps followed for pitch marking are shown in the Algorithm 3. In the algorithm, we perform framing to extract the pitch mark from each frame, where t is the frame size. The main difference between the two approaches is that in the first approach the pitch marking is performed in each voiced region which is of variable length, on the other hand in the second approach the pitch marking is performed in fixed-size frames which gives a better approximation of the pitch contour as seen in the results.

The calculated $pitchMarks$ is the input for linear stylization, discussed below.

C. Linear Stylization

In this, we approximate the stylized pitch contour using linear functions. The linear stylization is done using $pitchMarks$. First, we calculate the slope between two consecutive pitch marks using equation 5, where m is the slope and (x_1, y_1) and (x_2, y_2) are coordinates of the two consecutive pitch marks. The number of slopes generated is equal to the number of straight lines (K) needed to approximate the pitch contours of a speech signal.

$$m = \frac{y_2 - y_1}{x_2 - x_1} \tag{5}$$

Next, the intermediate pitches, called stylized pitches, between two consecutive pitch marks are calculated using the straight-line equation. Algorithm 4 shows the detailed steps of Linear Stylization. In the algorithm, we use k to generate the intermediate points between two pitch marks.

TABLE II. MSE COMPARISON

Samples	Mean Squared Error (MSE)		
	Stylization via Wavelet [15]	Algorithm 2	Algorithm 3
1272-135031-0009.flac	3883.70	118.40	11.19
1272-135031-0010.flac	2999.80	89.30	2.60
1272-141231-0002.flac	1932.80	7192.80	1.17
1462-170138-0000.flac	2941.00	8451.40	19.64
2035-147961-0000.flac	1428.50	3124.50	32.12
422-122949-0025.flac	1669.20	6645.30	6.27
1673-143396-0004.flac	2911.50	17382.00	24.45
2035-152373-0013.flac	3055.50	3471.60	16.32
2803-161169-0009.flac	860.37	4766.20	0.67
7850-73752-0003.flac	1201.70	13129.00	6.64

IV. EXPERIMENT AND RESULTS

For the experimental evaluation, we use voice samples from the LibriSpeech ASR corpus [26]. LibriSpeech is a corpus of English speech containing approximately 1000 hours of audio samples of 16kHz, prepared by Vassil Panayotov with the assistance of Daniel Povey. The data is derived from audiobooks (part of LibriVox project) and is carefully segmented and aligned. We test the voice samples for both Algorithm 2, 3 and compare our results with the previously proposed methodology [15]. We use Edinburgh Speech Tools Library for pitch marking [38]. We use the *ptch_fix* function which is a part of YAAPT pitch tracking Algorithm [39], to perform the pitch smoothing.

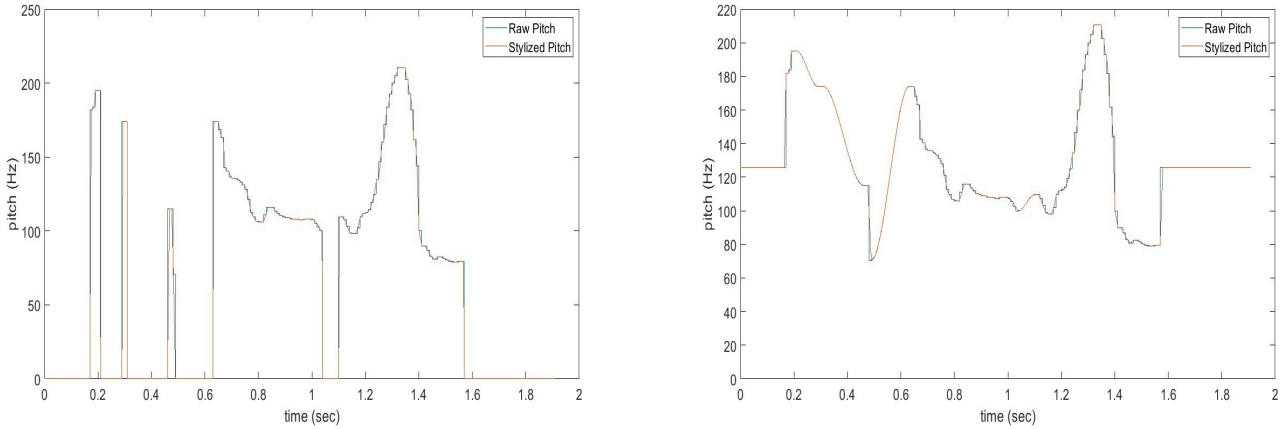
A. Comparison using MSE

Linear stylization approximates the original pitch contour using subset points, the parameter used to test the accuracy of the approximation is mean squared error (MSE). The lower values of MSE suggest a better approximation of the original pitch contours. The stylized pitch contour generated by the proposed algorithms is shown in Fig. 5. Fig. 5a, shows the pitch marks retrieved from the voiced region of the pitch contours. The pitch marks retrieved from smoothed pitch contour are shown in Fig. 5b.

Table II, shows a comparison between the three approaches. From the table, the MSE of Algorithm 2 is higher than the previously proposed speech stylization methodology using wavelet analysis [15]. This is because, in [15] the change points are extracted from each frame, whereas an Algorithm 2 the pitch marks are extracted from the complete signal, without framing of the pitch contour. However, for Algorithm 2, the MSE is considerably low compared to the [15], as the pitch marks are extracted for both voiced and unvoiced regions from each frame. The second approach of stylization yields better results than [15]. This gives a perception that the subset points extracted via pitch marks give better approximations. The average of the corpus is given in Fig. 6, in the figure we plot the values of MSE at log scale to give better representation.

B. Comparison using Subset Size(K)

The efficiency of the algorithm is tested using the number of segments (K), as K is directly proportional to the number of intermediate points generated. It is evident from the Algorithm



(a) Pitch Marking for Voiced Region

(b) Pitch Marking after Smoothing

Fig. 5. Original Pitch Contour and Stylized Pitch Contour for Audio Sample "*1272-135031-0009.flac* [26]".

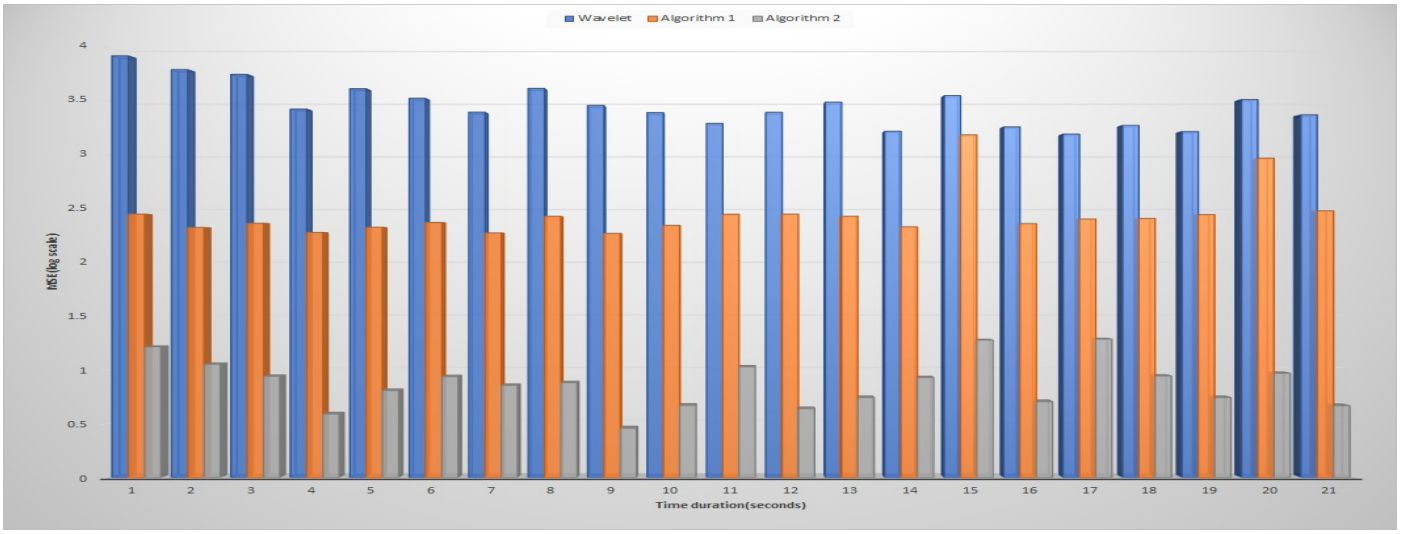


Fig. 6. The Average Mean Square Error by the Three Algorithms.

4 that the more the number of segments in the linear stylization process more is the time complexity. The number of segments K in the stylized pitch contours generated by the proposed algorithms is shown in Fig. 8. Fig. 8a and 8b, shows the segments obtained by using Algorithm 2 and 3, respectively.

Table III, shows the number of segments generated by the proposed algorithms and compares the same with [15]. The table shows that the proposed algorithms need less number of line segments for the stylized pitch contour in comparison to [15]. For all cases, we find that there is a significant difference in the number of line segments K generated by the proposed approach in comparison to [15]. The average result of the complete corpus is given in Fig. 7, the results show that on average 82.97% less is the subset size.

TABLE III. COMPARISON OF K

Samples	No. of lines		
	Stylization via Wavelet [15]	Algorithm 2	Algorithm 3
1272-135031-0009.flac	730	135	250
1272-135031-0010.flac	3656	725	1121
1272-141231-0002.flac	4454	920	1664
1462-170138-0000.flac	4574	1518	2714
2035-147961-0000.flac	4454	2300	3338
422-122949-0025.flac	5568	1159	2165
1673-143396-0004.flac	7225	2827	4316
2035-152373-0013.flac	5681	3144	4860
2803-161169-0009.flac	10189	1948	3007
7850-73752-0003.flac	10382	2873	4970

C. Comparison of the Proposed Algorithms

Finally, we compare the number of line segments (K) and the MSE of the proposed algorithms. The number of segments K, is significantly large when the pitch marks are retrieved from voiced and unvoiced regions after pitch smoothing, Fig. 9. The reason for this is framing, the segments are extracted from each frame which results in a better approximation of the original pitch contour. We can also see from Fig. 10 that mean

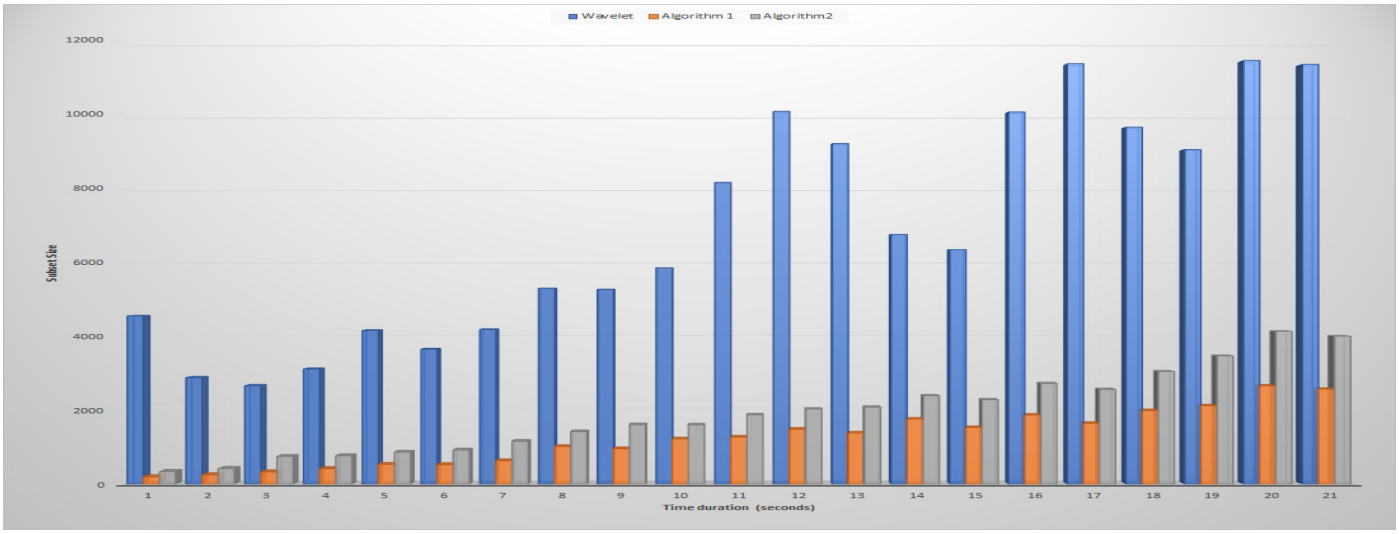
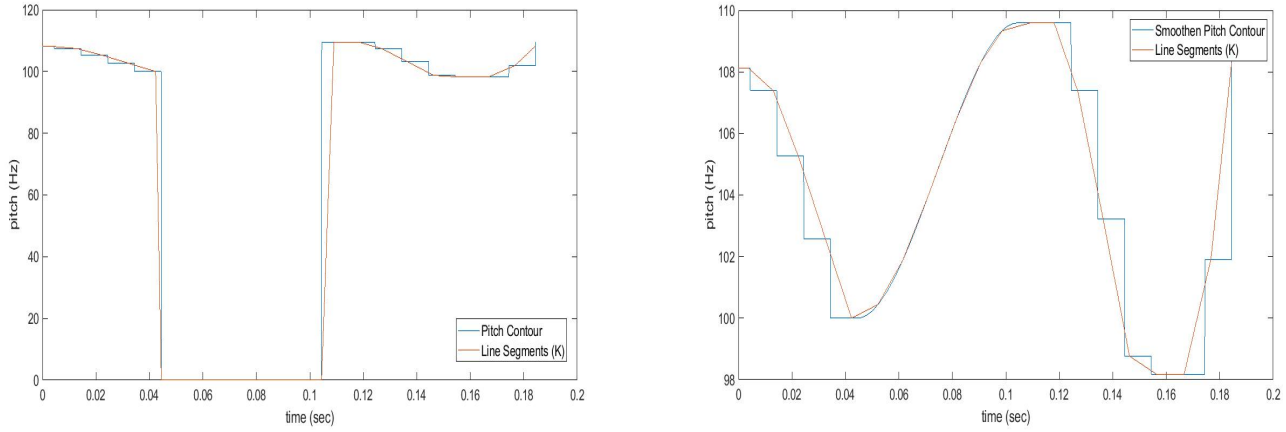


Fig. 7. The Average Subset Size by the Three Algorithms.



(a) Pitch Marking for Voiced Region

(b) Pitch Marking after Smoothing

Fig. 8. Number of Segments K for Audio Sample "*1272-135031-0009.flac* [26]".

square error reduces with an increase in the subset points. The results show that the approach that extracts the pitch marks both from voiced and unvoiced regions using framing is better in terms of MSE, but the complexity of the same is more.

V. Conclusion

The paper proposes two stylization approaches of pitch contour using linear functions. The subset of points used for stylization is the pitch marks on the pitch contour. The pitch marks capture the voice intonation of a speaker. The experimental results show that the proposed algorithms need fewer line segments (K) to approximate the stylized pitch contour with a low mean squared error. The results show a better approximation of the pitch contour using the pitch marks in comparison to the change points selected in the wavelet decomposition. First, the pitch marks are extracted from the voiced region of the pitch contours. Further, as an extension, we consider both voiced and unvoiced regions in the pitch contour to retrieve the pitch marks after performing pitch smoothing. The approximation result is better for the latter approach. In the future, we intend to test the proposed algorithm for more voice samples and apply it for real-time applications like voice morphing, templates to speaker recognition, etc.

References

[1] B. Gillett and S. King, "Transforming f0 contours," 2003.

[2] N. Obin, "Analysis and modelling of speech prosody and speaking style," Ph.D. dissertation, Ph. D. dissertation, IRCAM, Paris VI University, 2011.

[3] R. W. Ng, T. Lee, C.-C. Leung, B. Ma, and H. Li, "Analysis and selection of prosodic features for language identification," in *Asian Language Processing, 2009. IALP'09. International Conference on*. IEEE, 2009, pp. 123–128.

[4] P. Labutin, S. Koval, and A. Raev, "Speaker identification based on the statistical analysis of f0," *women*, vol. 16, no. 23.7, pp. 24–9, 2007.

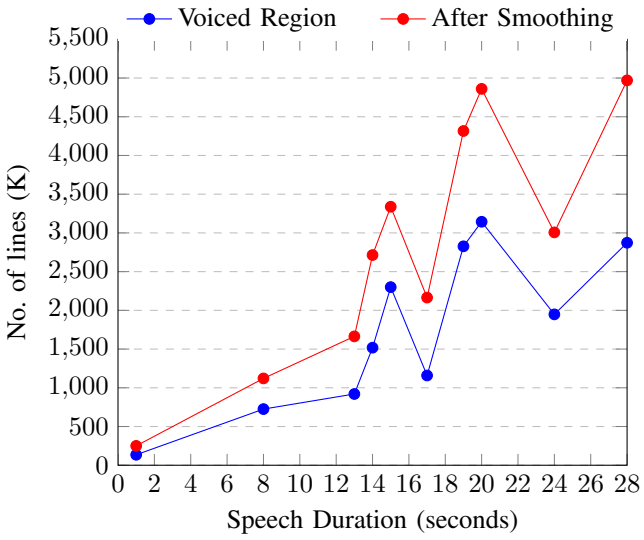


Fig. 9. Number of Segments(K) Vs Time.

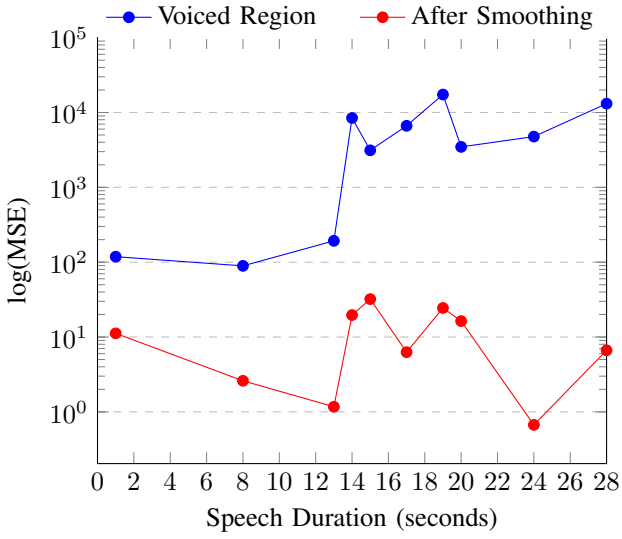


Fig. 10. MSE Vs Time.

[5] D. Guo, H. Yu, A. Hu, and Y. Ding, "Statistical analysis of acoustic characteristics of tibetan lhasa dialect speech emotion," in *SHS Web of Conferences*, vol. 25. EDP Sciences, 2016.

[6] N. Obin, J. Beliao, C. Veaux, and A. Lacheret, "Slam: Automatic stylization and labelling of speech melody," 2014.

[7] R. Dall and X. Gonzalvo, "Jndslam: A slam extension for speech synthesis," in *Proc. Speech Prosody*, 2016, pp. 1024–1028.

[8] J. Hart, R. Collier, and A. Cohen, *A perceptual study of intonation: an experimental-phonetic approach to speech melody*. Cambridge University Press, 2006.

[9] R. Nygaard and D. Haugland, "Compressing ecg signals by piecewise polynomial approximation," in *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181)*, vol. 3. IEEE, 1998, pp. 1809–1812.

[10] P. K. Ghosh and S. S. Narayanan, "Pitch contour stylization using an optimal piecewise polynomial approximation," *IEEE signal processing letters*, vol. 16, no. 9, pp. 810–813, 2009.

[11] J. 't Hart, "F 0 stylization in speech: Straight lines versus parabolas," *The Journal of the Acoustical Society of America*, vol. 90, no. 6, pp. 3368–3370, 1991.

[12] D. Hirst and R. Espesser, "Automatic modelling of fundamental frequency using a quadratic spline function." 1993.

[13] X. Yin, M. Lei, Y. Qian, F. K. Soong, L. He, Z.-H. Ling, and L.-R. Dai, "Modeling f0 trajectories in hierarchically structured deep neural networks," *Speech Communication*, vol. 76, pp. 82–92, 2016.

[14] N. Obin and J. Beliao, "Sparse coding of pitch contours with deep auto-encoders," 2018.

[15] D. Wang and S. Narayanan, "Piecewise linear stylization of pitch via wavelet analysis," in *INTERSPEECH*, 2005.

[16] E. Shriberg, A. Stolcke, D. Hakkani-Tur, and G. Tur, "Prosody-based automatic segmentation of speech into sentences and topics," *Speech Communication*, vol. 32, pp. 127–154, 09 2000.

[17] E. Shriberg, R. A. Bates, and A. Stolcke, "A prosody only decision-tree model for disfluency detection," in *EUROSPEECH*, 1997.

[18] A. Stolcke, N. Coccaro, R. Bates, P. Taylor, C. Van Ess-Dykema, K. Ries, E. Shriberg, D. Jurafsky, R. Martin, and M. Meteer, "Dialogue act modeling for automatic tagging and recognition of conversational speech," *Comput. Linguist.*, vol. 26, no. 3, pp. 339–373, Sep. 2000. [Online]. Available: https://doi.org/10.1162/089120100561737

[19] M. Sönmez, E. Shriberg, L. Heck, and M. Weintraub, "Modeling dynamic prosodic variation for speaker verification," in *ICSLP, Sydney, Australia*, August 1998.

[20] C. d'Alessandro and P. Mertens, "Automatic pitch contour stylization using a model of tonal perception," *Computer Speech and Language*, vol. 9, no. 3, pp. 257–288, 1995.

[21] A. Origlia, G. Abete, F. Cutugno, I. Alfano, R. Savy, and B. Ludusan, "A divide et impera algorithm for optimal pitch stylization," in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[22] O. P. Yadav and S. Ray, "Piecewise modeling of ecg signals using chebyshev polynomials," in *Computational Intelligence in Data Mining*. Springer, 2019, pp. 287–296.

[23] ——, "Ecg signal characterization using lagrange-chebyshev polynomials," *Radioelectronics and Communications Systems*, vol. 62, no. 2, pp. 72–85, 2019.

[24] R. Bakis, "Systems and methods for pitch smoothing for text-to-speech synthesis," Nov. 16 2006, uS Patent App. 11/128,003.

[25] X. Zhao, D. O'Shaughnessy, and N. Minh-Quang, "A processing method for pitch smoothing based on autocorrelation and cepstral f0 detection approaches," in *2007 International Symposium on Signals, Systems and Electronics*. IEEE, 2007, pp. 59–62.

[26] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[27] L. White and S. King, "The eustace speech corpus," 2003.

[28] W. J. Hess, "Algorithms and devices for pitch determination of speech signals," in *Automatic Speech Analysis and Recognition*. Springer, 1982, pp. 49–67.

[29] A. Banerjee, S. Pandey, and M. A. Hussainy, "Separability of human voices by clustering statistical pitch parameters," in *2018 3rd International Conference for Convergence in Technology (I2CT)*. IEEE, 2018, pp. 1–5.

[30] C. Xu, S. Li, G. Liu, Y. Zhang, E. Miluzzo, Y.-F. Chen, J. Li, and B. Firner, "Crowd++: Unsupervised speaker count with smartphones," in *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*. ACM, 2013, pp. 43–52.

[31] A. Agneessens, I. Bisio, F. Lavagetto, M. Marchese, and A. Sciarrone, "Speaker count application for smartphone platforms," in *Wireless Pervasive Computing (ISWPC), 2010 5th IEEE International Symposium on*. IEEE, 2010, pp. 361–366.

[32] K. Kasi, "Yet another algorithm for pitch tracking:(yaapt)," Ph.D. dissertation, Old Dominion University, 2002.

[33] M. Wu, D. Wang, and G. J. Brown, "A multipitch tracking algorithm for noisy speech," *IEEE Transactions on Speech and Audio Processing*, vol. 11, no. 3, pp. 229–241, 2003.

[34] B. S. Lee, "Noise robust pitch tracking by subband autocorrelation classification," Ph.D. dissertation, Columbia University, 2012.

[35] M. Sondhi, "New methods of pitch extraction," *IEEE Transactions on audio and electroacoustics*, vol. 16, no. 2, pp. 262–266, 1968.

[36] A. Banerjee, S. Pandey, and K. Khushboo, "Voice intonation transformation using segmental linear mapping of pitch contours," in *2018 IEEE 4th International Conference on Computer and Communications (ICCC)*. IEEE, 2018, pp. 1278–1282.

[37] M. Legát, J. Matoušek, and D. Tihelka, "A robust multi-phase pitch-mark detection algorithm," in *Eighth Annual Conference of the International Speech Communication Association*, 2007.

[38] P. Taylor, R. Caley, A. W. Black, and S. King, "Edinburgh speech tools library," *System Documentation Edition*, vol. 1, pp. 1994–1999, 1999. [Online]. Available: http://www.cstr.ed.ac.uk/projects/speech_tools

[39] H. H. Stephen A. Zahorian. Yaapt pitch tracking matlab function. [Online]. Available: http://www.ws.binghamton.edu/zahorian