A NEW FUNDAMENTAL FREQUENCY MODIFICATION ALGORITHM WITH TRANSFORMATION OF SPECTRUM ENVELOPE ACCORDING TO F_0

Kimihito TANAKA *Masanobu ABE*
NTT Human Interface Laboratories
1-1 Hikarinooka, Yokosuka-shi, Kanagawa, 239 JAPAN
E-mail: tanaka@nttspch.hil.ntt.co.jp

ABSTRACT

This paper proposes a new speech modification algorithm which makes it possible to change the fundamental frequency (F_0) while preserving high quality. One novel point of the algorithm is that the spectrum envelope is transformed according to amount of F_0 modification. Based on a codebook mapping formulation, transformation rules are generated using speech data uttered in a different F_0 range. The rules have two purposes: one is transforming the spectrum envelope of the low frequency band and the other is adjusting the balance between low band power and high band power. The proposed algorithm is applied to a text-to-speech system based on waveform concatenation, and good performance is confirmed by listening tests.

1. INTRODUCTION

It is well known that fundamental frequency (F_0) modification is one of the most difficult problems in developing high quality speech synthesis-by-rule based on speech segment concatenation. Although the waveform-based speech synthesis approach offers good performance for moderate F_0 modification, the quality of the synthesized speech is degraded when F_0 is modified strongly as in synthesizing emotive speech. Similar degradation also occurs when the target F_0 contours are quite different from the F_0 contours of the synthesis units, such as when generating a rapidly raising F_0 contour using a synthesis unit whose F_0 contour gradually decreases. Based on the speech production theory[1], one possible reason for these degradations in speech quality is that the desirable speech spectrum itself has different characteristics from the spectrum of the synthesis units; i.e., the characteristics of a natural speech spectrum vary with the voice source. For example, voicing source studies[2] report that the low frequency band characteristics are, for a glottal source model, affected by the open quotient parameter; this seems also true for the amplitudes of the first and second harmonics. These facts suggest the necessity of controlling the speech spectrum when F_0 is modified strongly. However, almost all of the conventional F_0 modification algorithm did not concerned with the phenomena. This paper proposes a new F_0 modification algorithm wherein the spectrum envelope is also transformed according to the amount of F_0 modification. This makes it possible to strongly change F_0 while preserving high quality.

2. F_0 MODIFICATION ALGORITHM

2.1. Outline of a proposed algorithm

The relationship between spectrum envelope and F_0 is extracted from speech samples uttered in three F_0 ranges: high, middle, and low. This extraction was performed based on a codebook mapping formulation[3]; i.e., three spectrum envelope codebooks are generated, and the code vectors of the three codebooks have a one-to-one correspondence. Based on the codebooks, spectrum envelope modification can estimated for any F_0 value. Figure 1 shows the basic idea of the estimation method. This example shows transformation when the target F_0 is higher than the F_0 of the speech segment providing the synthesis units. Here, it is assumed that speech segments providing the synthesis units are uttered in the middle F_0 range. In Fig.1, the lower and upper circles show the codebooks of the middle and high F_0 ranges, respectively. The solid arrows indicate the differential vectors (i.e., spectrum envelope differences) between middle and high F_0 ranges. First, a differential vector (the dashed arrow) against the input spectrum is estimated as the linear combination of the k-nearest neighbor differential vectors using weights obtained by fuzzy vector quantization. The differential vector is then stretched using a rate determined by the target F_0 value and F_0 value of the speech segment providing the synthesis unit. Finally, the resulting differential vector (the bold solid arrow) is added to the spectrum envelope of the input spectrum.

The proposed algorithm consists of both off-line and on-line procedures. In the off-line procedure, codebooks are generated for spectrum envelope transformation. These codebooks are stored in a text-to-speech system. Using the codebooks, the on-line procedure transforms the spectrum

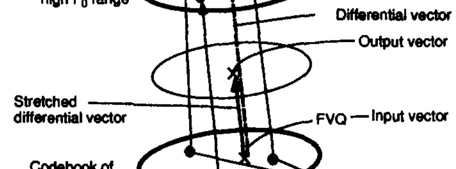

Figure 1. A basic idea of the spectrum envelope modification

envelope of the synthesis units according to target F_0. The procedure is implemented within the signal processing module for text-to-speech conversion.

Another unique point of the proposed algorithm is spectrum envelope extraction. According to the source model theory[2], the F_0 value might strongly influence the low frequency spectrum such as the amplitudes of the first and second harmonics. To precisely extract the low frequency spectrum envelope, we propose a new spectrum envelope extraction approach based on the PSE (Power Spectrum Envelope analysis) algorithm[4]. The proposed algorithm offers stable spectrum envelope estimation even for the transient parts of the speech signal. We call it IPSE (Improved Power Spectrum Envelope analysis).

In the following sections we explain the IPSE algorithm and the off-line and on-line procedures for spectrum envelope transformation.

2.2. IPSE algorithm

The spectrum envelope is extracted pitch-synchronously. The algorithm approximates the spectrum envelope by interpolating the local peaks of the harmonics of the log-power spectrum using the cosine function. IPSE analysis proceeds as follows.

(1) The speech signal is cut out with a five-fundamental-period-length Hamming window and its log-power spectrum is calculated by FFT.
(2) The local-maximum value of the log-power spectrum is sampled at f_n ($nF_0 - F_0/2 < f_n < nF_0 + F_0/2$, where n is an integer).
(3) If the interval between f_n and f_{n+1} is larger than 1.5 times F_0, local-peaks of the log-power spectrum within the interval are added to the sampled series.
(4) The sampled series are linearly interpolated.
(5) The interpolated lines are re-sampled using F_0/n intervals(here, n is the integer that gives the maximum value of F_0/n while satisfying $F_0/n < 50Hz$), and are approximated by the cosine model shown in equation (1) to minimize the mean square error between the model and re-sampled points.

$$Y(\lambda) = \sum_{i=0}^{M} A_i \cos i\lambda, \qquad (0 \le \lambda \le \pi) \qquad (1)$$

We call A_i the IPSE (Improved Power Spectrum Envelope) cepstrum. Taking perceptual sensitivity into account, A_i is converted in the mapping stage to mel-cepstrum by Oppenheim's recursion[5].

Figure 2 shows an example of the spectrum envelopes extracted by IPSE analysis. It clearly shows that IPSE precisely approximates the local peaks of the harmonics.

2.3. Codebook generation for spectrum envelope transformation (off-line procedure)

Three codebooks are generated using word utterances pronounced in three F_0 ranges (high, middle and low F_0 range). The code vectors of three codebooks have a one-to-one correspondence to the other codebook's vector. A codebook of deferential vectors is also generated by calculating the difference between the corresponding code vectors of the middle and other F_0 range codebook. The codebooks are generated as follows. The following algorithm is applied only to voiced speech segments. Here, we explain the flow of high F_0 range codebook generation. A codebook for low F_0 range can be generated in the same way. Figure 3 shows a block diagram of the codebook generation procedure. Numbers in the following explanation correspond to the block numbers in Fig.3.

(1) Extract the spectrum envelope (IPSE cepstrum) from the voiced speech segments.
(2) Convert the IPSE cepstrum to IPSE mel-cepstrum.
(3) Generate a codebook of the IPSE cepstrum for the middle F_0 range using the LBG algorithm[6].
(4) Encode the IPSE cepstrum of the middle F_0 range by the codebook.
(5) Realize time alignment by phoneme-to-phoneme linear time warping between the same words uttered in the middle and high F_0 ranges.
(6) Referring to the output of step 4 and 5, the IPSEs of high F_0 range are classified into the clusters generated in step 3.
(7) Average the IPSEs in each cluster, which generates the code vectors of the high F_0 range codebook.
(8) Generate the codebook of the differential vectors by calculating the difference between the corresponding code vectors of middle and high F_0 range codebooks.

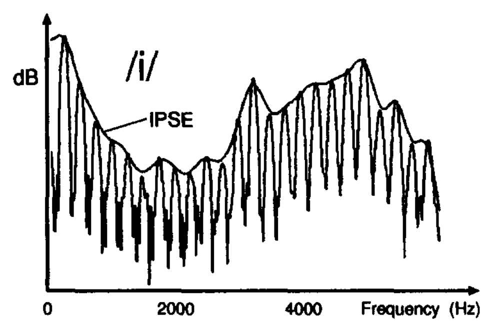

Figure 2. An example of spectrum envelopes extracted by IPSE analysis

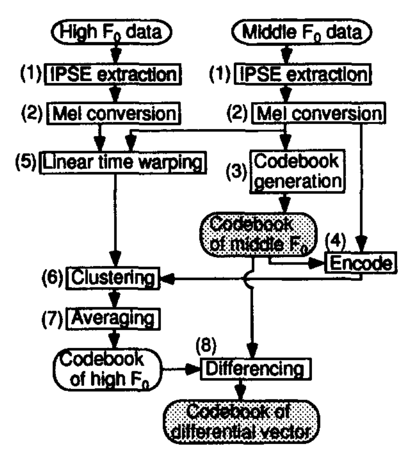

Figure 3. Block diagram of codebook generation procedure

2.4. F_0 modification Algorithm (on-line procedure)

Figure 4 shows a block diagram of the spectrum envelope transformation algorithm. The algorithm modifies input speech for the low frequency and high frequency bands in different ways. The IPSE transformation according to F_0 is performed only in the low frequency band, and the original signal of input speech is used in the high frequency, because the effect of F_0 appears most strongly in the low frequency band. All procedures are applied to the voiced speech segments pitch-synchronously. The procedures are as follows, they start after synthesis unit selection and setting the target (or desirable) F_0 value, numbers in the following explanation correspond to the block numbers in Fig.4.

(1)-(3) Extract the spectrum envelope (IPSE cepstrum) by the algorithm explained in section 2.2.

(4) Convert the IPSE cepstrum to IPSE mel-cepstrum.

(5) Fuzzy vector quantize the IPSE mel-cepstrum using the codebook of middle F_0 range, and calculate fuzzy membership functions μ_k for k-nearest-neighbors by equation (2).

$$\mu_k = \frac{1}{\sum_{j \in K} (d_k/d_j)^{1/(f-1)}} \qquad (2)$$

(6) Estimate the differential vector V against the input spectrum envelope using middle and low/high F_0 range codebooks by equation (3). The low or high F_0 range codebook is selected according to the target F_0 and F_0 of the selected synthesis unit.

$$V = \frac{\sum_{j \in K} \mu_j V_j}{\sum_{j \in K} \mu_j} \qquad (3)$$

where V_i is the differential vector between code vectors of middle and low/high F_0 ranges, and μ_j is the fuzzy membership function of k-nearest-neighbor vectors.

(7) Calculate the stretching rate r of the differential vector by equation (4).

$$r = \frac{F_{target} - F_{unit}}{F_{low/high} - F_{middle}} \qquad (4)$$

where the denominator of equation (4) denotes the F_0 difference between the target F_0 and F_0 of the selected synthesis unit, and the numerator denotes the F_0 difference between the low/high and middle F_0 ranges.

(8) Stretch differential vector V by multiplying it by stretching rate r. The intent is to obtain the transformed IPSE (mel-cepstrum) whose F_0 range is different from the ranges of the three codebooks.

(9) Add the differential vector V to the IPSE mel-cepstrum extracted from the synthesis unit waveform.

(10) Convert the IPSE mel-cepstrum to IPSE cepstrum.

(11) Obtain a waveform by IFFT from the transformed IPSE using zero-phase.

(12) Obtain a low frequency band signal using a low pass filter.

(13) Obtain a high frequency band signal using a high pass filter.

(14) Cut out waveform with a two-fundamental-period-length Hamming window.

(15) Obtain a high frequency band signal of the synthesis unit using a high pass filter.

(16) Adjust the power level of high frequency band signal of the synthesis unit to power level of the transformed IPSE.

(17) Add the low frequency band signal to high frequency band signal.

(18) Pitch-synchronous-overlap-add (PSOLA) the previous and current signals.

3. PERFORMANCE EVALUATION BY LISTENING TEST

Three codebooks were generated using 520 phoneme-balanced word utterances pronounced by one female speaker in three F_0 ranges. To minimize the VQ distortion in low and high F_0 ranges, codebook size was set at 512. The listening tests were carried out to evaluate the performance of the proposed algorithm. The experimental conditions are shown in table 1.

3.1. Evaluation by ABX tests

ABX listening tests were carried out to determine if the proposed algorithm could synthesize more natural speech

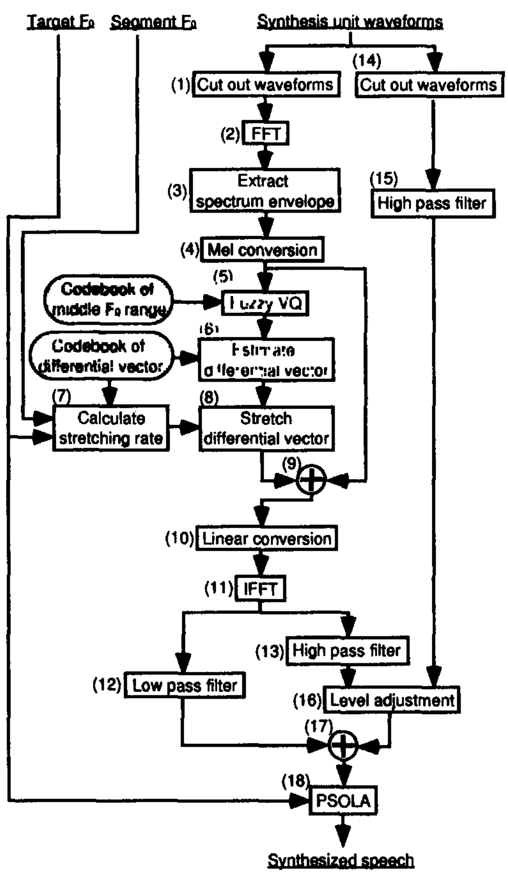

Figure 4. Block diagram of spectrum envelope transformation algorithm

than the PSOLA algorithm. Three versions of five words were created as follows;

(S1) Speech whose F_0 was modified by the PSOLA algorithm (conventional algorithm).

(S2) Speech whose F_0 was modified by the proposed algorithm.

(S3) Natural speech uttered in the target F_0 range.

For S1 and S2, the target values of F_0 contours, power patterns and duration of phonemes were extracted from S3. The F_0 modifications were from middle to low F_0 range and from middle to high F_0 range. S1 and S2 were tagged as stimulus A and B in the ABX tests. S1, S2, and S3 were used for stimulus X. The test subjects judged whether stimulus A or B was closest to stimulus X.

Figure 5 illustrates the test results. For modification toward the lower range, 85% of the subjects judged that the proposed algorithm (S2) yielded more natural speech (S3). On the other hand, for F_0 modification toward the upper range, 59% of the subjects judged that S2 was closer to S3. These results indicate that the proposed algorithm is superior to the PSOLA algorithm and the proposed algorithm has especially good performance in F_0 modification toward a lower F_0 range.

3.2. Evaluation by preference tests

To evaluate overall performance, preference tests were carried out. Two versions of five sentences were synthesized by a text-to-speech (TTS) system[7] with low, middle, and high F_0 ranges. One was the speech synthesized by the PSOLA algorithm and the other by the proposed algorithm. The F_0 contours, power pattern, and duration of phonemes of both versions were the same. The differences arose from the signal processing module of the TTS system. Figure 6 shows the results of preference tests.

In terms of low and high F_0 ranges, 94% and 85% of the subjects preferred the speech synthesized by the proposed algorithm, respectively. However, the speech synthesized by the PSOLA algorithm was preferred by 74% in the high F_0 range. Judging from an informal listening test and analysis, this poor performance might be caused by the relatively larger power of the high frequency band in the speech synthesized by the proposed algorithm. Thus the subjects judged the speech to be noisy. In conclusion, the proposed algorithm is superior to PSOLA in the low and middle F_0 ranges.

Table 1. Experimental conditions

Sampling frequency	12kHz
Boundary frequency between low and high band	500Hz
Order of IPSE cepstrum	30
Average of low F_0 range Average of middle F_0 range Average of high F_0 range	172.4 215.9 309.6
Codebook size	512
Number of k-nearest-neighbor	12
Fuzziness	1.5
Number of Subjects	10 persons

4. CONCLUSION

We proposed a new fundamental frequency modification algorithm with spectrum envelope transformation according to F_0. Listening tests confirmed that the proposed algorithm is superior to the conventional method. We have a plan to analyze male speech and to establish a method of high quality male speech synthesis. In the future, we will apply this algorithm to various speech synthesis goals, such as the synthesis of emotive speech.

ACKNOWLEDGMENT

We are grateful to the members of the Speech Processing Department for their helpful discussions. We also thanks Dr. Kitawaki, director of the Speech and Acoustics Lab., and Mr. Nakajima, leader of the Speech Processing Group, for their continuous support of this work.

REFERENCES

[1] G.Fant, "Acoustic Theory of Speech Production," Mouton, The Hague (1960).

[2] D.H.Klatt and L.C.Klatt, "Analysis, synthesis, and perception of voice quality variations among female and male talkers," J. Acoust. Soc. Am., Vol.87, No.2, pp.820-857 (1990-02).

[3] M.Abe, S.Nakamura, K.Shikano and H.Kuwabara, "Voice conversion through vector quantization," ICASSP'88, pp.655-658 (1998).

[4] T.Nakajima and T.Suzuki, "Power Spectrum Envelope (PSE) Analysis based on Pitch Frequency Interval Sampling of Short Term Power Spectrum," The First Symposium on Advanced Man-Machine Interface through Spoken Language, pp.155-164 (1988).

[5] A.V.Oppenheim and R.W.Schafer, "Digital Signal Processing," pp.480-531 (1975).

[6] Linde, Buzo and Gray, "An algorithm for vector quantizer design," IEEE Trans., COM-28, 1, pp.84-95 (1980).

[7] K.Hakoda, T.Hirokawa, H.Tsukada, Y.Yoshida and H.Mizuno, "Japanese Text-to-Speech Software based on Waveform Concatenation Method," AVIOS'95, pp.65-72 (1995).

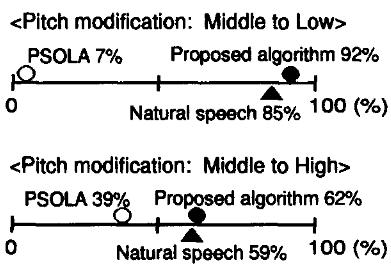

Figure 5. Results of ABX tests

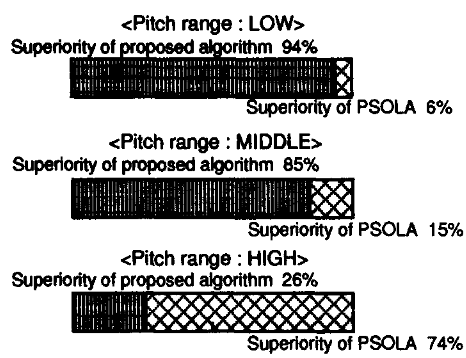

Figure 6. Results of preference tests