GENERALIZED F0 MODELLING WITH ABSOLUTE AND RELATIVE PITCH FEATURES FOR SINGING VOICE SYNTHESIS

S. W. Lee [1], *Shen Ting Ang* [1,2], *Minghui Dong* [1], *and Haizhou Li* [1]

[1] Human Language Technology Department, Institute for Infocomm Research, A*STAR, Singapore 138632
[2] Department of Statistics, University of Warwick, Coventry CV4 7AL, United Kingdom
{swylee, mhdong, hli}@i2r.a-star.edu.sg, S.T.Ang@warwick.ac.uk

ABSTRACT

Natural pitch fluctuations are essential to human singing. To effectively synthesize singing voice, the generation of these pitch fluctuations is necessary. Previous synthesis methods classify and reproduce them individually. These fluctuations, however, are found to be dependent and vary under different contexts. This paper proposes a generalized framework for F0 modelling to learn and generate these fluctuations on a note basis. Context-dependent hidden Markov models, representing the possible fluctuations observed in particular musical contexts, are built. To capture the pitch fluctuation and the voicing transitions in human singing, we employ both absolute and relative pitch as the modelling features. Results of our experiments on pitch accuracy and quality of synthesized singing showed that the proposed framework achieves accurate pitch generation and better naturalness of synthesized outputs.*

Index Terms— singing, synthesis, pitch, modelling, HMM

1. INTRODUCTION

Singing voice synthesis has been one of the emerging and popular research topics in recent years [1]-[3]. There is a growing number of related applications in the market, such as entertainment development, computer-assisted vocal training and music production [4]-[6].

This study of F0 modelling is based on speech-to-singing synthesis, which is a popular approach in singing voice generation [7]. Given a lyrics-reading speech input, this approach converts the input speech to a singing voice output by manipulating the pitch and spectrum with an input melody. Vocal characteristics of the input speech are hence easily preserved. Furthermore, pleasant singing voice is possibly made from any individual's input speech, even if he or she is not good at singing.

Pitch variation in a singing voice is essential to the perceived quality [8]. Some of the unique features in the fundamental frequency (F0) of a singing voice, e.g. vibrato and overshoot, are shown to give the strongest contributions [9]. Vibrato is a quasi-periodic frequency modulation; whereas overshoot is a deflection exceeding the target note after a note change. Consequently, these fluctuations together with some others, for example, the preparation (similar to overshoot, but a deflection in the opposite direction observed just before the note change) and the fine fluctuation [9], are enriched during synthesis. Specifically, a second-order parametric system [10] has been widely used to explicitly modify these fluctuations individually [7].

In actual singing, these fluctuations are singer-dependent, correlated with each other and presenting different behaviour in different contexts. For instance, vibrato is present more often in long notes rather than short notes. The overshoot in low frequency notes is different from that in high frequency notes. This work proposes a generalized framework for F0 modelling and generation of singing voice. In particular, the F0 behaviour is learnt and modelled by context-dependent note hidden Markov models (HMMs), also known as note HMMs. Various F0 fluctuations are implicitly modelled using the same HMM representation. Decisions of which F0 features and where they occur are implicitly made, by maximizing the likelihood of the generated F0 contour (similar to [11]). In contrast, each note always has overshoot, vibrato or preparation, plus the fine fluctuation in [7].

Although note HMMs have been used in a similar manner for modelling singing style parameters [12], our method models all the F0 fluctuations while [12] focuses on vibrato and power dynamics. Hence, our modelling is generalized in that (1) all fluctuations in F0 are captured; (2) the representation is non-parametric such that the exact shapes of the fluctuations are solely determined by the training data and the input note sequence for generation; and (3) the modelling is note-based, rather than being based on fluctuation type (as in [7]). Different F0 representations, absolute [3] and relative [12] ones, have been used. In our method, the absolute and relative F0 are modelled under different streams. Information from the generated absolute and relative contours is then fused into a final F0 estimate for synthesis. Our experiments have shown that the proposed method achieves satisfactory performance in F0 generation and natural singing synthesis.

2. REVIEW ON SPEECH-TO-SINGING SYNTHESIS

Before introducing our F0 modelling method, we first briefly review the speech-to-singing synthesis approach (as shown in Fig. 1). It is helpful to illustrate how to use the generated F0 for singing voice synthesis in our experiments.

Given a lyrics-reading speech input $x(n)$, Tandem-STRAIGHT [13] is used to decompose $x(n)$ into spectral envelope, F0 and aperiodicity for analysis. The timing information for each syllable in $x(n)$ is found by forced alignment. Our note models are built by learning the singing F0 behaviour with reference to the melody. These note models will be used to generate the singing F0 with an input melody later. The modelling and generation will be discussed further in the next section. During synthesis, the vocal-

[*] This work was conducted when Shen Ting Ang was doing his research attachment at Institute for Infocomm Research in 2011.

timing process constructs the target sequences of spectral envelope and aperiodicity for singing output $y(n)$ (similar to [7]). Specifically, spectral envelopes and aperiodicity functions of syllables in $x(n)$ are replicated according to the input melody. Tandem-STRAIGHT will be finally used to combine the sequences of the converted spectral envelope and aperiodicity with the generated F0 and produce the singing voice output.

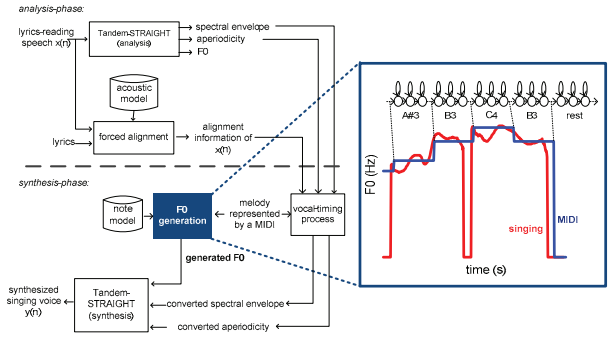

Fig. 1. Block-diagram of the speech-to-singing method used.

3. NOTE-BASED F0 MODELLING & GENERATION

In this work, the F0 of singing voice is modelled on a note, context-dependent basis, as shown in Fig. 1. The F0 fluctuations on identical note context are assumed to be consistent. These models are initialized with mono-note models (analogous with the monophone models in speech recognition, covering the complete singing range 80-1100 Hz) and one rest model for rest notes. Under the equal-tempered scale with the A4 note at 440 Hz, these mono-note models spanned from A1 (55 Hz) to C6 (1046.5 Hz) and are one semitone apart from each other. Depending on the desired range of generated F0s and the training data used, it is possible to use a different span of notes.

The singing recordings are first cut into segments, each of which contains a line of lyrics. The corresponding MIDI files are segmented as well and will be converted into the note labels. The HMM-based Speech Synthesis System (HTS) is used as the platform for model training and generation [14].

3.1. Feature Extraction

Absolute F0 and relative F0 are used as the features. Let $p(n)$ and $r(n)$ be the corresponding feature values, which are defined as

$$p(n) = 1200 \log_2 \left(p_{\mathrm{Hz}}(n) \right) \tag{1}$$

$$r(n) = 1200 \log_2 \left(\frac{p_{\mathrm{Hz}}(n)}{m(n)} \right) \tag{2}$$

where $p_{\mathrm{Hz}}(n)$ and $m(n)$ are the pitch frequency and the known MIDI note frequency respectively. $p_{\mathrm{Hz}}(n)$ and $m(n)$ are in Hertz and $p(n)$ and $r(n)$ are in cents. $p_{\mathrm{Hz}}(n)$ is extracted by Tandem-STRAIGHT with a 5 ms frame shift.

MIDI	singing	
	voiced	unvoiced
pitched	①: $p(n)$ real, $r(n)$ real	③: $p(n)$ undefined, $r(n)$ undefined
rest	②: $p(n)$ real, $r(n)$ undefined	④: $p(n)$ undefined, $r(n)$ undefined

Table 1. Four conditions of $p(n)$ and $r(n)$.

The feature vector consists of these two static features, together with the delta and the delta-delta features. These six elements are put under two streams during modelling. There are some conditions under which $p(n)$ and $r(n)$ are undefined, as shown in Table 1. Hence, the method of multi-space probability distribution HMM (MSD-HMM) [15] is used. For all the undefined $p(n)$ and $r(n)$, they are treated as the same, assigned to the zero-dimensional space.

Given $m(n)$, there exists a one-to-one mapping between $p(n)$ and $r(n)$. It may seem to be redundant in using both features. Nevertheless, $p(n)$ and $r(n)$ have their own merits and weaknesses in F0 modelling. $p(n)$ directly represents the pitch, without relating to $m(n)$. Any undefined $p(n)$ (Condition ③ and ④) essentially means to unvoiced singing. On the other hand, there are several conditions (②, ③ and ④) giving undefined $r(n)$, including unvoiced singing, rest MIDI or both. $r(n)$ is effective for modelling the F0 interval to $m(n)$ with insufficient training data. Similar distributions for r(n) are expected for notes under similar context. This is especially critical for training with tree-based context clustering. Consequently, both features are used for modelling.

3.2. Model Training

Our note HMMs are N-state single-mixture left-to-right hidden semi Markov models (HSMMs) [16]. Although all F0 fluctuations are modelled with the same model structure, different states are responsible for certain types of fluctuations in practice. The transient fluctuations, namely overshoot and preparation, are expected to be located at earlier states; whereas vibrato will probably occur at the middle towards the end of the model. The fine fluctuation will be spread throughout all the states.

The training procedure is similar to the standard HTS process for speaker-dependent systems [14]. Context-dependent note models are initialized by the mono-note models and rest model with context information. We apply the following context information: (1) note identity (previous, current and next note); (2) note interval relative to the current note in the unit of semitones (previous and next note); (3) note duration (previous, current and next note); and (4) tempo class of the song. Based on the tempo stored in the MIDI, a song is classified as one of five tempo classes: slow (below 76 beats per minute (bpm)), slightly slow (76-90 bpm), moderate (91-105 bpm), moderately fast (106-120 bpm) and fast (above 120 bpm). The context-dependent note models are clustered by a decision tree, using questions regarding the four context information above and the minimum description length (MDL) criterion, and then re-estimated again.

3.3. F0 Generation

With the context-dependent note models, the F0 contour for a given segment can be generated. The corresponding note sequence is first converted to note labels with context information. The parameter generation algorithm for Case 3 in [11] is adopted to estimate the F0 contour using the absolute F0 stream. In this generation algorithm, the hidden state sequence and the $p(n)$ sequence are estimated. We use the time alignment of the note sequence to specify the model durations. Global variance (GV) is considered in the generation. Let $\tilde{p}(n)$ be the resultant absolute F0 sequence. The relative F0 sequence is estimated ($\tilde{r}(n)$) in a similar manner. It is further converted to the absolute F0 domain

by using $m(n)$. Let $\tilde{q}(n)$ be this resultant F0 sequence. Any undefined value in $\tilde{p}(n)$ and $\tilde{q}(n)$ refers to an unvoiced frame.

In the following, we propose a method to fuse $\tilde{p}(n)$ and $\tilde{q}(n)$ and generate a final F0 contour $f(n)$. $f(n)$ is determined as below.

- If both $\tilde{p}(n)$ and $\tilde{q}(n)$ are real, $f(n) = \left(\tilde{p}(n) + \tilde{q}(n)\right)/2$;
- If both $\tilde{p}(n)$ and $\tilde{q}(n)$ are undefined, both estimates are consistent, suggesting an unvoiced frame. $f(n)$ is set to undefined;
- If $\tilde{p}(n)$ is real and $\tilde{q}(n)$ is undefined, $\tilde{p}(n)$ and $\tilde{q}(n)$ are contradictory. In this case, $m(n)$ will be considered. $f(n)$ is set to $\tilde{p}(n)$, if $m(n)$ is real. Otherwise, $f(n)$ is set to undefined. This is essentially a voting between $\tilde{p}(n)$, $\tilde{q}(n)$ and $m(n)$;
- If $\tilde{p}(n)$ is undefined and $\tilde{q}(n)$ is real, $\tilde{r}(n)$ must be real for real $\tilde{q}(n)$. This is another contradictory result, since $\tilde{p}(n)$ must be real for real $\tilde{r}(n)$. Voting with $m(n)$ is not applicable for this case, as this is always biased towards $\tilde{q}(n)$. Instead, a random number with equal probabilities on {0, 1} is drawn. $f(n)$ is set to $\tilde{p}(n)$, if this number is one. Otherwise, $f(n)$ is set to undefined.

With $\tilde{p}(n)$, $\tilde{q}(n)$ and $f(n)$, synthesis of the singing voice $y(n)$ can be made by using either one of them.

4. EXPERIMENTS

The performance of the proposed F0 modelling is reported in the following. A F0 modelling is effective if the generated F0 contour is close to human singing F0 contour and the synthesized singing is natural with high quality. We evaluated our system performance by two indices: (1) objective measurement of the errors in generated F0 contours; and (2) subjective listening test on the naturalness of $y(n)$. A collection of solo singing recordings from a male professional singer was used for the studies below. There were altogether 17 Mandarin Chinese pop songs. Each song lasted about four minutes, totalling 66 min 37 sec. These songs were selected by the singer, based on his singing skills, rhythms and pitch ranges. Besides, the MIDI files for these songs were used as the input for F0 modelling and generation. There were 685 segments for training and 33 segments for testing in total. These testing segments were unseen from training.

With this collection of singing recordings, a set of 31 mono-note models and one rest model were trained, together with the associated context-dependent models. These models spanned from D2# (77.82 Hz) to B4 (493.88 Hz) and each contained five states.

Several modelling methods were compared. They are: (A) recorded singing voice with Tandem-STRAIGHT analysis and synthesis (no modification on singing F0); (B) the proposed F0 modelling; (C) MIDI F0; and (D) the second-order damping system for singing F0 modelling [7]. The spectral envelope and aperiodicity extracted from the recorded singing voice remained unchanged (as the converted spectral envelope and the converted aperiodicity in Fig. 1) for all methods.

4.1. Objective Measurements Of F0 Generation Accuracy

Fig. 2 shows an example of the generated F0 contours. The measurements on various types of error of F0 generation are given in Table 2.

Comparing the generated F0 contour $f(n)$ with the reference singing F0, the human F0 fluctuations were mostly learnt and generated. For example, the overshoots at the beginnings of notes, fluctuations within note periods, early stop at the end of lyrics, etc. Some of the unvoiced singing transitions at the note ends were similar to the reference singing, while some were not. Hence, our generalized F0 modelling was shown to be capable of learning various F0 fluctuations under the same premise.

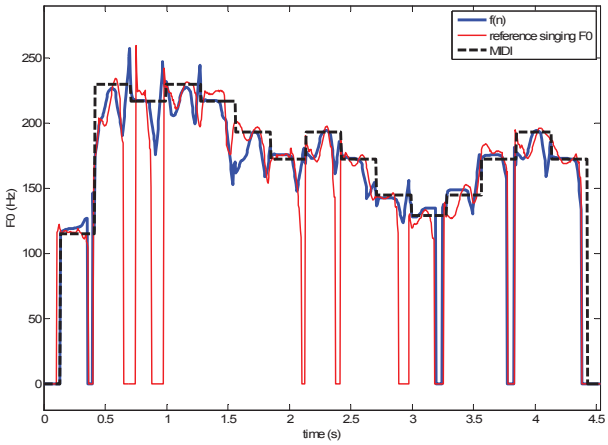

Fig. 2. Comparison between a generated F0 contour $f(n)$, with the reference singing F0 and the MIDI input.

system	error type		
	RMSE (cents)	E_{10} (%)	E_{01} (%)
$f(n)$	140.27	5.20	12.40
$\tilde{p}(n)$	**137.99**	7.83	**9.64**
$\tilde{q}(n)$	161.15	3.38	13.65
MIDI F0	141.84	**2.69**	15.12
2nd order damping system [7]	141.88	**2.69**	15.12

Table 2. Measurements of F0 accuracy from various systems.

The root-mean-square-error (RMSE) measurement was applied on voiced singing frames only with correct voicing in the generated F0. If there is any voicing error, i.e. a voiced frame treated as unvoiced (E_{10}), or an unvoiced frame treated as voiced (E_{01}), the corresponding voicing error is counted. The best results under individual error types are bolded. It was found that systems from the proposed modelling methods ($f(n)$, $\tilde{p}(n)$ and $\tilde{q}(n)$) achieved the lowest RMSE and E_{01}, compared to the second-order damping system and the one from MIDI F0. Concerning E_{10}, the second-order damping system and the MIDI F0 system achieved the lowest percentage. This is expected, as the two systems always generate voiced frames, except during rests. It was found that using both the absolute F0 and the relative F0 streams, $f(n)$ achieved satisfactory performance in terms of various types of error measurements. The merits of $\tilde{p}(n)$ and $\tilde{q}(n)$ were inherited.

4.2. Subjective Listening Tests

Two subjective listening tests were conducted. The first one examined the performance of the proposed F0 modelling and generation on the naturalness of synthesized singing. The system ($f(n)$) fusing both the absolute and the reference F0 estimates was used as a representative of our method. Listeners were asked to compare and rate the naturalness of the four systems (A, B, C and D) by mean opinion score (MOS). Possible MOS ranged from 1

(bad) to 5 (excellent). This test consisted of 20 questions, taken from the testing set. Each contained one line of lyrics. Listeners could play the stimuli as many times as they wished. A total of 26 subjects participated, contributing 520 responses.

Fig. 3 (left figure) depicts the box plot of the MOS result. On each box, the central mark is the median. The edges are the 25th and 75th percentiles. Outliers are indicated by plus symbols. The experimental results showed that the naturalness of the proposed F0 modelling (Method B) was significantly better than generation using MIDI F0 (Method C) or the second order damping system (Method D) with 95% confidence. Compared with the recorded singing voice (Method A), the naturalness of the proposed modelling was not as high as the recorded singing.

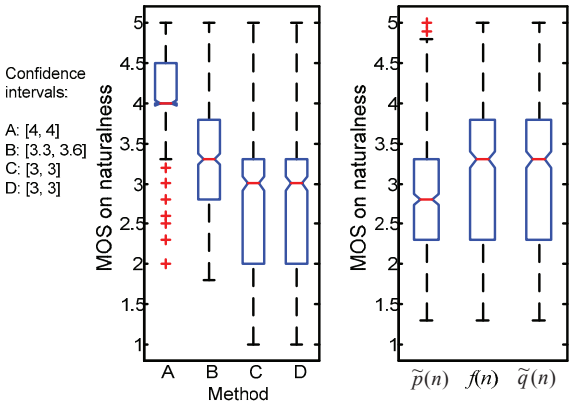

Fig. 3. Box plots of the MOS results. (left) comparison between the proposed F0 modelling ($f(n)$ as Method B) and other methods; (right) comparison between the systems using the absolute F0 stream, the relative F0 stream and both streams for synthesis.

The second listening test examined the relative effectiveness of using the absolute, relative and both streams for F0 modelling and generation, on singing synthesis. There were 20 questions in total. Similar to the first listening test, listeners were asked to rate the naturalness. 26 subjects participated, contributing 517 responses. Three other responses were found to be invalid and excluded. The box plot is depicted in Fig. 3 (right figure). Comparing the MOS results, it was found that $f(n)$ and $\tilde{q}(n)$ achieved the highest naturalness. Their performances were more or less the same; whereas $\tilde{p}(n)$ was inferior to the other two. Referring to Table 2, $\tilde{p}(n)$ had the lowest RMSE and E_{01}, but the highest E_{10}. $\tilde{q}(n)$ had low E_{10}, but the highest E_{01}. This revealed that the preservation of voiced singing plays an important role in the naturalness perceived by listeners, while F0 accuracy and the preservation of unvoiced singing are less dominant.

5. CONCLUSIONS

Human singing voice possesses natural fluctuations in pitch. These fluctuations have been traditionally classified as vibrato, overshoot, preparation and fine fluctuation, and are modelled individually. Knowing that they are correlated and vary under different contexts, a generalized F0 fluctuation modelling and generation method is proposed in this paper. Various kinds of F0 fluctuations are jointly modelled using an identical HMM representation, without an explicit decision on the present types of fluctuations. The proposed modeling makes use of the merits of absolute and relative F0 features. Our experiments confirmed that this generalized modelling achieves satisfactory F0 generation accuracy and captures the natural fluctuations in pitch, leading to improved synthesized singing.

6. REFERENCES

[1] *Synthesis of Singing Challenge (Special Session)*, *Proc. Interspeech*, Aug. 2007.

[2] M. Akagi, "Rule-based voice conversion derived from expressive speech perception model: How do computers sing a song joyfully?" in *Proc. ISCSLP*, Tutorial 01, Nov. 2010.

[3] K. Saino, H. Zen, Y. Nankaku, A. Lee, and K. Tokuda, "HMM-based singing voice synthesis system," in *Proc. Interspeech*, pp. 1141-1144, Sep. 2006.

[4] H. Kenmochi and H. Ohshita, "VOCALOID – Commercial singing synthesizer based on sample concatenation," in *Proc. Interspeech*, Aug. 2007.

[5] P. Kirn (2009, Oct. 6). iPhone Day: LaDiDa's Reversed Karaoke composes Accompaniment to Singing [Online]. Available: http://createdigitalmusic.com/2009/10/iphone-day-ladidas-reverse-karaoke-composes-accompaniment-to-singing/

[6] L. Regnier and G. Peeters, "Singing voice detection in music tracks using direct voice vibrato detection," in *Proc. ICASSP*, pp. 1685-1688, Apr. 2009.

[7] T. Saitou, M. Goto, M. Unoki, and M. Akagi, "Speech-to-singing synthesis: Converting speaking voices to singing voices by controlling acoustic features unique to singing voices," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 215-218, Oct. 2007.

[8] M. B. Dayme, *Dynamics of the Singing Voice*, 5th ed. New York: Springer, 2009.

[9] T. Saitou and M. Goto, "Acoustic and perceptual effects of vocal training in amateur male singing," in *Proc. Interspeech*, pp. 832-835, Sep. 2009.

[10] Y. Ohishi, H. Kameoka, D. Mochihashi, H. Nagano, K. Kashino, "Statistical modeling of F0 dynamics in singing voices based on Gaussian processes with multiple oscillation bases," in *Proc. Interspeech*, pp. 2598-1601, Sep. 2010.

[11] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in *Proc. ICASSP*, pp. 1315-1315, Jun. 2000.

[12] K. Saino, M. Tachibana, and H. Kenmochi, "A singing style modeling system for singing voice synthesis," in *Proc. Interspeech*, pp. 2894-2897, Sep. 2010.

[13] H. Kawahara, M. Morise, T. Takahashi, R. Nisimura, T. Irino, and H. Banno, "TANDEM-STRAIGHT: A temporally stable power spectral representation for periodic signals and applications to interference-free spectrum, F0, and aperiodicity estimation," in *Proc. ICASSP*, pp. 3933-3936, Mar. 2008.

[14] H. Zen, K. Oura, T. Nose, J. Yamagishi, S. Sako, T. Toda, T. Masuko, A. W. Black, and K. Tokuda, "Recent development of the HMM-based speech synthesis system (HTS)," in *Proc. APSIPA ASC*, pp. 121-130, Oct. 2009.

[15] K. Tokuda, T. Masuko, N. Miyazaki, and T. Kobayashi, "Multi-space probability distribution HMM," *IEICE Trans. Inf. & Syst.*, vol. E85-D, pp. 455-464, Mar. 2002.

[16] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "A hidden semi-Markov model-based speech synthesis system," *IEICE Trans. Inf. & Syst.*, vol. E90-D, pp. 533-543, May 2007.