PURETALK: A HIGH QUALITY JAPANESE TEXT-TO-SPEECH SYSTEM
Masayuki Yamada, Yasuo Okutani, Toshiaki Fukada, Takashi Aso and Yasuhiro Komori
Platform Technology Development Center, Canon Inc.
53, Imaikami-cho, Nakahara-ku, Kawasaki-shi, Kanagawa 211-8501, Japan
E-mail:masayuki@cis.canon.co.jp

ABSTRACT
This paper describes a high quality Japanese text to speech (TTS) system, PureTalk. This system is similar to the conventional diphone-based TTS using PSOLA except that PureTalk employs the following novel techniques which enable to produce more intelligible and natural-sounding speech: 1) two-stage duration modeling based on a linear regression technique, 2) F0 contour modeling using polynomial segment models, 3) sophisticated waveform unit selection, and 4) efficient waveform compression designed for TTS system. The result of the subjective hearing test shows that PureTalk achieves high quality under practical computation and memory requirement.

1. INTRODUCTION
Applying statistical methods is a promising way to achieve a high quality text to speech (TTS) system[1, 2, 3]. We have developed a Japanese TTS engine, PureTalk, which utilize statistical methods throughout its synthesis process as the other modern TTS systems.

Although the system is similar to the conventional diphone-based TTS using PSOLA[4] (pitch synchronous overlap adding), it employs the following novel techniques which enable to produce more intelligible and natural-sounding speech.

- Two-stage duration modeling based on a linear regression technique
- F0 contour modeling using polynomial segment models
- Sophisticated waveform unit selection
- Efficient waveform compression designed for TTS system

The two-stage duration modeling is introduced to reduce monotony of synthesized speech. We consider that the monotony is mainly because most of current duration modeling approaches lack capturing macroscopic duration characteristics. The first duration model captures macroscopic (e.g., phrase) duration characteristics and the second model captures microscopic (e.g., phoneme) characteristics.

Generation natural F0 contours is one of the key issues to produce high quality synthetic speech. We introduce here a precise F0 segment modeling where the F0 contours are represented by an arbitrary order polynomial. The segment modeling has been introduced to speech recognition to represent a segment better than Hidden Markov Model (HMM)[6].

The waveform unit selection algorithm is important when waveform concatenation method such as PSOLA is adopted. While the method realizes synthesized speech resembling original speaker, it seems to be sensitive to quality of the waveform units. Statistical approaches which we have adopted are effective for designing a proper size dictionary from a large speech corpus.

The waveform concatenation algorithm (more specifically TD-PSOLA) also has advantage that it requires slight computation while a waveform dictionary tends to become large. Our waveform compression algorithm reduces the dictionary size effectively with keeping the advantage in computation.

In the following sections, the algorithms introduced into PureTalk are described along the TTS flow (Fig.1). The results of the subjective test held on PureTalk and four commercial TTS systems are also mentioned.

2. TEXT ANALYSIS
At first, text analysis is carried out for determining

1. pronunciation (phoneme sequence)
2. 'pause phrase' boundary
3. 'accent phrase' boundary and accent position in the phrase

of an input sentence.

A pause (short silent term) is placed at every pause phrase boundary.
At first, duration of a pause phrase $D$ is estimated from the number of morae, the number of accent phrases contained in the phrase and the phoneme categories in the phrase as follows.

$$D = N_m / \left( A_{N_a} + B_{N_m} + \sum_{i=1}^{N_p} C_{P(i)} \right)$$

where

- $N_a$ : # of accent phrases
- $N_m$ : # of morae
- $N_p$ : # of phonemes
- $P(i)$ : phoneme type of $i$-th phoneme

and $A_{N_a}$, $B_{N_m}$ and $C_{P(i)}$ are constants assigned to $N_a$, $N_m$, $P(i)$ respectively. They are trained on a speech corpus using a linear regression based method.

In the second step, the duration of each phoneme $\hat{d}_i$ is estimated independently of the phrase duration $D$ as follows:

$$\hat{d}_i = \sum_{j=1}^{N_f} \sum_{k=1}^{C_j} a_{j,k} \delta_{j,k}$$

where

- $N_f$ : # of estimation factors
- $C_j$ : # of categories of the $j$-th factor
- $a_{j,k}$ : value assigned to the $k$-th category of the $j$-th factor
- $\delta_{j,k}$ : \begin{cases} 1 & \text{if } k\text{-th category of the } j\text{-th factor is satisfied} \\ 0 & \text{otherwise} \end{cases}

As estimation factors, we use the category of the $i$-th phoneme, the categories of the phonemes next to the $i$-th phoneme, position of the $i$-th phoneme in the accent phrase, etc. These factors are given as the results of the text analysis. The coefficients $a_{j,k}$ are trained on a speech corpus.

At last, the $\hat{d}_i$ is modified to satisfy

$$\sum_{i=1}^{N_p} \hat{d}_i = D .$$

It can be obtained by using ML estimation:

$$d_i = \hat{d}_i + \rho \frac{\sigma_{P(i)}^2}{\sum_{i=1}^{N_p} \sigma_{P(i)}^2}$$

where

- $\sigma_{P(i)}^2$ : variance of the duration of $P(i)$
- $\rho$ : scaling factor
- $\sum_{i=1}^{N_p} \sigma_{P(i)}^2$ : variance in a speech corpus.
3.2. F0 contour estimation using polynomial segment models

The F0 contour generation is carried out in the following two steps.

1. baseline estimation for each accent phrase based on quantification method type 1
2. F0 contour generation for each phoneme using segment models

These steps are shown in Fig. 3.

At first, a baseline of an F0 contour is estimated for each accent phrase. The results of text analysis such as the number of morae contained in the accent phrase, accent type, position in the sentence, etc. are used for the estimation factors.

The second step is based on the segment model where an F0 contour within a phoneme is represented by a second order polynomial curve:

\[ f(t) = a_{0,c(i)} + a_{1,c(i)}z + a_{2,c(i)}z^2 + f, \quad z = \frac{t - t_0}{t_1 - t_0} \]

where

- \( t_0, t_1 \): start and end time of the \( i \)-th phoneme
- \( c(i) \): category of the \( i \)-th phoneme
- \( f \): baseline determined in the first step.

The coefficients of the curve \((a_{0,c}, a_{1,c}, a_{2,c})\) are trained on a speech corpus[6]. The position in the phrase, the accent position of the phrase the phoneme types (voiced/unvoiced), etc. are used to determine the category of a phoneme \((c(i))\). After the every F0 contour has been generated, smoothing is carried out to compensate gaps between phonemes.

4. WAVEFORM GENERATION

4.1. Unit selection from speech corpus

At first in waveform generation, diphone units are retrieved from a waveform dictionary. The diphone units included in the dictionary are pre-selected from a speech corpus in advance (Fig. 4). We use the following three methods for the unit selection:

- HMM-based likelihood scoring
- heuristic filtering
- evaluation of concatenation distortion at unit boundaries.

In the HMM-based scoring, speech recognition using phoneme HMMs is carried out so that the mis-recognized units not be included in the dictionary. The units which have extraordinary prosodic features are also eliminated by the heuristic filtering.

In the concatenation distortion evaluation, LPC cepstrum distances are measured at unit boundaries. A very large text corpus is used for considering the frequency of the occurrence of the concatenation.

4.2. Prosody modification

Prosody of the diphone units are modified using TD-P SOLA (Time Domain Pitch Synchronous Overlap Adding) to fit the estimated prosody. Then, the units are concatenated.

4.3. Waveform compression designed for TTS system

Scalar quantization and sub-optimal predictive coding is a speech compression method which requires very few computation for decoding. It is a combination of sec-
second order predictive coding and sub-optimal codebook trained by the LBG algorithm. The prediction coefficient and the codebook are fixed and optimized for a waveform unit. This compression technique realizes comparable storage\(^1\) and quality to the ADPCM system with having one-tenth computation.

At first in the encoding, the prediction coefficients are computed to optimize the second order prediction model by minimizing prediction error \((\sum d_i^2)\) on the speech segment \(\{x_t\}\):

\[
x_t = a_1 x_{t-1} + a_2 x_{t-2} + d_t.
\]

Then, the codebook is trained on the residual \(\{d_t\}\) using the LBG algorithm.

Finally, the \(\{x_t\}\) are encoded by finding \(i\) which minimizes the coding distortion:

\[
|a_1 \hat{x}_{t-1} + a_2 \hat{x}_{t-2} + \hat{d}_t - x_t|
\]

where

\[
\hat{x}_t: \text{decoded waveform}
\]

\[
\{\hat{d}(i)\}: \text{residual codebook}.
\]

The decoding process:

\[
\hat{x}_t = a_1 \hat{x}_{t-1} + a_2 \hat{x}_{t-2} + \hat{d}_t
\]

requires equal or less computation than PSOLA.

5. EXPERIMENT

A subjective hearing test has been carried out to investigate the quality of our system. The specifications of the acoustic module of PureTalk are shown in Table 1. Four kinds of commercially-available Japanese TTS systems were also used for comparison. Eight sentences were synthesized for every system (i.e., 40 stimuli in total) and they were evaluated by a subjective opinion test of five grades with 20 listeners. The grades were “very good,” “good,” “fair,” “bad,” and “very bad” and the score points were 5, 4, 3, 2 and 1, respectively. The MOS (mean opinion score) is shown in Table 2.

For all sentences, PureTalk has achieved best score among the systems. Note that PureTalk got the best score among the systems for all sentences.

\(^1\)4 bits/sample in the current implementation.

<table>
<thead>
<tr>
<th>Program size</th>
<th>100KB</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAM requirement</td>
<td>300KB</td>
</tr>
<tr>
<td>Dictionary size</td>
<td>300KB</td>
</tr>
<tr>
<td># of diapason units</td>
<td>330</td>
</tr>
<tr>
<td>Sampling frequency</td>
<td>2kHz</td>
</tr>
</tbody>
</table>

Table 1: Spec. of the acoustic module of PureTalk.

<table>
<thead>
<tr>
<th>System</th>
<th>PT</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOS</td>
<td>3.9</td>
<td>2.3</td>
<td>1.5</td>
<td>3.1</td>
<td>2.4</td>
</tr>
</tbody>
</table>

Table 2: Result of subjective evaluation; the ‘PT’ denotes PureTalk and the ‘A’–’D’ denote commercial systems.

6. CONCLUSION

In this paper, we described our Japanese TTS engine PureTalk which utilize statistical methods in prosody generation and waveform unit selection. Efficient waveform compression designed for TTS system is also used. The result of the subjective hearing test show that PureTalk achieves high quality under practical computation and memory requirement.

In our future work, we will investigate methods for registering multiple waveform units into a waveform dictionary for a diaphone.

7. REFERENCES