THE JOURNAL OF THE ACOUSTICAL SOCIETY OF KOREA Vol.20, No.4E 2001. 12. pp. 31~38

A New Pruning Method for Synthesis Database Reduction Using Weighted Vector Quantization

Sanghun Kim*, Youngjik Lee*, Keikichi Hirose**

*Spoken Language Processing Team, Human Interface Department, Electronics and Telecommunication Research Institute
**Dept. of Frontier Informatics, School of Frontier Sciences, University of Tokyo
(Received 28 May 2001; revised 4 September 2001; accepted 24 November 2001)

Abstract

A large-scale synthesis database for a unit selection based synthesis method usually retains redundant synthesis unit instances, which are useless to the synthetic speech quality. In this paper, to eliminate those instances from the synthesis database, we proposed a new pruning method called weighted vector quantization (WVQ). The WVQ reflects relative importance of each synthesis unit instance when clustering the similar instances using vector quantization (VQ) technique. The proposed method was compared with two conventional pruning methods through the objective and subjective evaluations of the synthetic speech quality: one to simply limit maximum number of instance, and the other based on normal VQ-based clustering. The proposed method showed the best performance under 50% reduction rates. Over 50% of reduction rates, the synthetic speech quality is not seriously but perceptibly degraded. Using the proposed method, the synthesis database can be efficiently reduced without serious degradation of the synthetic speech quality.

Keywords: *Speech synthesis, Text-to-Speech, Pruning, Unit selection, Corpus based synthesis*

I. Introduction

Recently, many Koran TTS systems have been adopted the unit selection based synthesis method[1]. This method generates highly natural synthetic speech without prosodic modifications. However, it is necessary to construct a large-scale speech database. In the synthesis database, each synthesis unit has many instances (or candidates) with different prosodic characteristics. To select the most appropriate synthesis unit instance, a cost function is applied to minimize spectral and prosodic mismatches while concatenating synthesis units. With the cost function, the synthesis unit instance is dynamically selected during synthesis process.

The synthetic speech quality of the unit selection based synthesis method is roughly proportional to the speech database size, and the necessary size usually reaches to a few hundred mega-byte. The huge database requires a large memory size and slows down the computational speed. Though an optimized sentence set was designed to reject the redundant synthesis unit instances as much as possible, the synthesis database still retains similar synthesis unit instances in the sense of prosodic and spectral characteristics. Those redundant synthesis unit instances should be pruned for the efficiency of TTS system.

In Microsoft, whistler system selected a small number

Corresponding author: Sanghun Kim (ksh@etri.re.kr)
Electronics and Telecommunications Research Institute, Daejeon 305-350 Korea

of instances based on HMM matching scores[2]. It was reported that very high concatenating quality was achieved by choosing instances with the highest HMM score. However, they only considered phonetic contexts except prosodic contexts. Black and Taylor[3] clustered phonetic and prosodic contexts using a decision tree. They pruned synthesis units by discarding 1~4 instances locating furthest from each cluster center. Reduction rates of 20% to 50% were realized without serious degradation in synthetic speech quality. In CHATR, Campbell and Black[4] selected the most representative instances from prosodic viewpoint for each unit using VQ clustering technique. The cluster number (i.e. codebook size) was determined according to the number of instances for each unit.

In fact, pruning methods have not been intensively studied for synthesis database reduction, which is an important issue for unit selection based synthesis method. In this paper, we propose a new pruning method, where the relative importance of the synthesis unit instances is reflected. In section 2 and 3, we introduce our synthesis system and the speech database preparation. Section 4 describes the proposed pruning method. We will discuss the experimental results in section 5. Finally, we will conclude this paper in section 6.

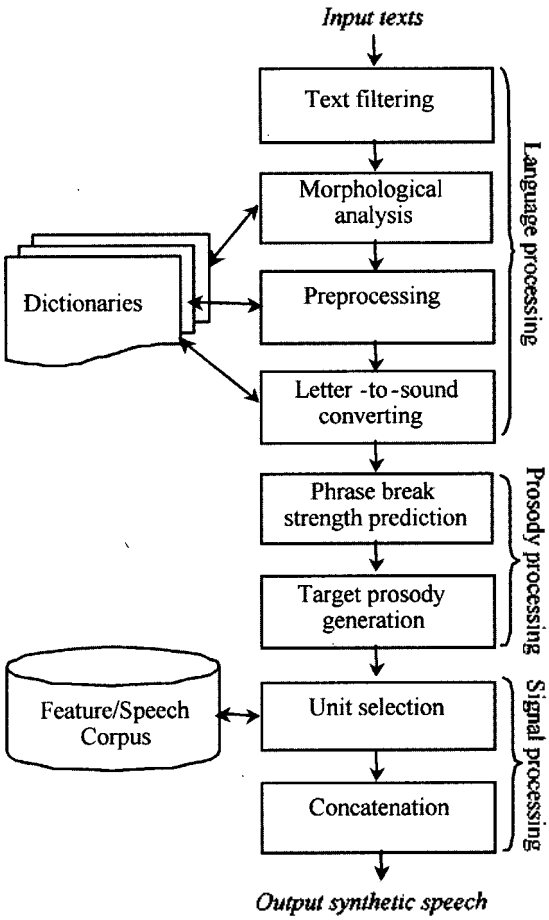


Figure 1. Block diagram of the synthesis system.

II. TTS System: Geulsori

The '*Geulsori*' has successfully adopted the unit selection based synthesis method in recent years. As shown in Figure 1, the synthesis system is composed of language processing, prosody processing, and signal processing.

Language processing

The language processing performs text filtering, morphological analysis, text preprocessing, and letter-to-sound converting. In text filtering module, the texts are filtered out undesired symbols (i.e. two byte graphic characters, control characters and non-sense symbol). It converts typical text forms (i.e. date, telephone number, e-mail address, and URL address) into the appropriate reading styles. In morphological analysis module, HMM-based statistical method has been utilized to analyze the morpheme boundaries and the best POS (Part-of-Speech) tag sequence. For letter-to-sound converting, the meaningful symbols, numerals, Chinese characters, and English characters are converted into the proper Korean alphabets in advance. Then, the graphemes are converted into the phonemes after applying several phonological rules.

Prosody processing

Prosodic phrase boundaries play an important role to realize natural and understandable synthetic speech. It breaks a sentence to several chunks of meaningful units[5]. To reflect the prosodic phrase boundaries, we introduced multi-level phrase break strength. The phrase boundaries were classified four kinds of break strength (no break, minor, medium, and major break) as pause length. The pause length must be a decisive feature to characterize the

phrase strength, though the prosodic phrase boundaries involve in the various prosodic features such as intonation, duration, and energy. To predict phrase break strength on texts, we adopted HMM-like POS sequence model[6]. Given POS sequence ($W_1^n = W_1, W_2, W_3, \ldots, W_N$), maximum probability of break strength sequence $Q(B^*)$ can be written as the following formula (1).

$$Q(B^*) = \max_{B^*} P(B_1^n \mid W_1^n) \tag{1}$$

where B_1^n is break strength sequence $B_1, B_2, B_3, \ldots, B_N$.. The performance shows 73.5% in 4-level of break strength prediction[7].

Signal processing

As feature vectors for unit selection, the spectral and prosodic features (i.e. LPC-based cepstrum coefficients, energy, pitch, and phoneme duration) are extracted and then normalized using Z-score (i.e. $z = \frac{\mu - x}{\sigma}$). To select the best combination of synthesis unit (i.e. triphone) instances, Viterbi beam search is utilized to find the best path with minimal accumulated distortion. In forward path, the Euclidean distortion between current triphone instance and the following triphone instance is accumulated in each state. When it arrives at the final state, we compare the accumulated distortion between instances in the final state and then decide which instance has minimal distortion. Beginning from that instance, we go to backward path and finally select the minimal distortion path. In particular, the unit selection was conducted only using concatenation cost (C^c).

$$C^c(u_{i-1}, u_i)_{path} = \sum_{j}^{Q} w_j^c C_j^c(u_{i-1}, u_i)_{path} \tag{2}$$

$$Best\ path = \arg\min_{path} \{C^c(u_{i-1}, u_i)_{path}\} \tag{3}$$

where Q is a number of the state. Concatenation cost C^c for unit u_{i-1} and u_i is represented where u_i is a phonetic and prosodic feature and w_j^c is a weight. Here, w_j^c is adjusted by perception. While concatenating, the phase mismatches at the concatenating boundaries cause perceptible glitches. The synthetic speech is smoothed by overlap-and-add method.

III. Database Preparation

To construct triphone-based synthesis database, triphone coverage respecting the phonetic/prosodic contexts should be considered. For the phonetic aspect, there are over fifty thousand triphones in Korean: [{V, C_i, #}+V+{C_i, C_f, V, #}], [{V, C_f, #}+C_i+{V}], and [{V}+C_f+{C, #}], where V, C_i, C_f, and # are 21 vowels, 19 syllable initial consonants, 7 syllable final consonants, and silence, respectively. For the prosodic aspect, a triphone should have enough instances to cover natural prosody of source utterance. In the result, the number of necessary triphone instances may reach over several millions. However, it is impractical to get all the necessary triphones in real situation.

In Korean, the function words (i.e. particles or inflections) play an important role to demarcate the syntactic boundaries. The major prosodic variations mostly occur at the function words. Accordingly, it had better get more triphone instances of function words. Thus a sentence corpus retaining a few hundred thousand of well-formed sentences was selected from the textbooks and newspapers. To avoid the redundant synthesis units in the resulted synthesis database, an iterative algorithm was introduced.

Iterative algorithm

i) Compute frequency of triphone occurrence *f(j)*, where *f(j)* represents the number of triphone *j* in the sentence corpus.

ii) For each sentence, calculate $C_l = \sum_{k \in S_l} f(k)$, where S_l is the l^{th} sentenceor.

iii) Choose S_l with the largest C_l.

iv) Delete triphone $k \in S_l$ from the sentence corpus.

v) Go to step ii) and iterate until the triphone coverage reaches a threshold.

The sentence set consists of 3,600 sentences and contains 14,882 unique triphones. The total number of triphone instances amounts to 410,000, approximately. With this sentence set, a female announcer naturally pronounced the source utterances.

To get phoneme-sized segments, the source utterances were automatically segmented into phones using HMM-

based continuous speech recognizer. The performance has shown that more than 80% of phonemes are correctly placed within 30msec deviation respect to manual segmentation results. In order to adapt to target speaker, the speaker recognizer has been trained using target speaker's database. With the adapted distribution and codebook weight parameters, Viterbi alignment has been conducted to segment an utterance into the correspondent phonetic symbols. Then, we manually corrected the phoneme boundaries.

With the phoneme segments, we constructed the synthesis database that the concatenating cost should be zero if two synthesis units are consecutively placed in the source utterances. It reduces the number of concatenating points that may produce spectral mismatches. The triphone units are further splitted as 4-level phrase break strength. The phrase break strengths mark sentence/clause boundary, phrase boundary, and word boundary. The number of unique triphone after reflecting the phrase break strength is 24,407. The distribution of the number of instances in the splitted triphone is as shown in Figure 2.

The resulting synthesis database reaches to 600Mbyte~1Gbyte. To reduce the database size, the original speech (16 kHz, 16 bits) has been compressed using waveform coding, i.e. u-law PCM (8 kHz, 8 bits) and ADPCM (16 kHz or 11 kHz, 4 bits).

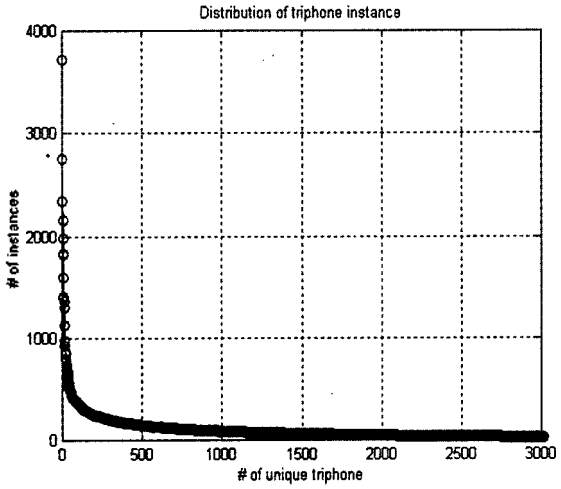


Figure 2. Distribution of number of instances (limited to 3,000 of unique triphones).

IV. Pruning Method: Weighed VQ

The unit selection process dynamically determines the best instance sequence by scoring the accumulated distance within words or phrases. Selecting the best instances is usually affected by the preceding and following unit instances. However, the conventional pruning methods ignored the mutual effects of adjacent units to be concatenated. Generally, some instances are more frequently selected during synthesis process. It means that the frequently selected instances are more important and contributive to the synthetic speech quality. Thus, we should differentiate the frequently selected instances with other rarely selected instances. To reflect this fact, we introduced a weight, which indicates the relative importance of the instances. We combined the weight with VQ for the new pruning method. To implement WVQ algorithm, the Lloyd algorithm was modified[8].

Weighted VQ algorithm

Step 1: Choose randomly initial N codeword $c_n^{(i)}(i=0)$.

Step 2: For each training vector (M), find the codeword that is nearest, and assign that vector to the corresponding cell.

$$q(x_m)=\min_{c_n^{(i)}}||x_m-c_n^{(i)}||^2 \quad m=1,2,\ldots,M \tag{4}$$

where $||e||^2=e_1^2+e_2^2+\ldots+e_n^2$.

Step 3: Find the centroid vector reflecting the frequency of the selected instances in each cluster.

$$c_n^{(i+1)}=\frac{\sum Q(x_m)=c_n^{(i)}x_m\times freq_m}{\sum Q(x_m)=c_n^{(i)}freq_m} \quad n=1,2,\ldots,N \tag{5}$$

where $freq_m$ is the frequency of the selected m^{th} instance.

Step 4: Set $i=i+1$ and calculate the average distance.

$$Dist^{(i)}=\frac{\sum_{m=1}^{M}||x_m-Q(x_m)||^2\times freq_m}{k\times\sum_{m=1}^{M}freq_m} \tag{6}$$

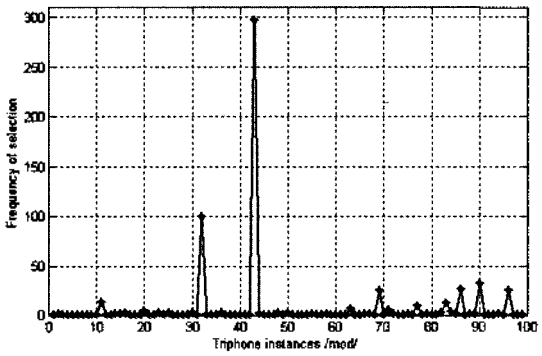
where k is a dimension of vector.

Step 5: Repeat steps 2~5 until the average distance is

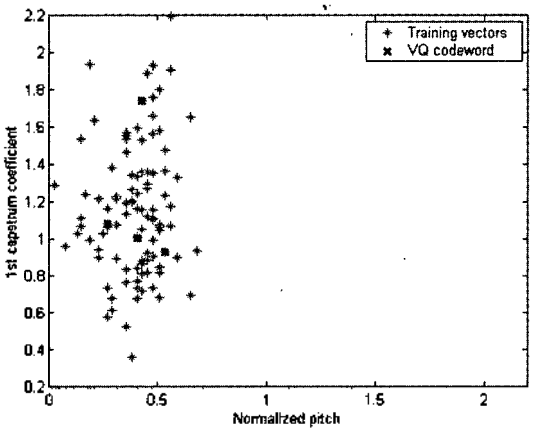
less than a preset threshold (ϵ)

$$\text{if}\left(\frac{Dist^{(i-1)} - Dist^{(i)}}{Dist^{(i-1)}} < \epsilon\right) \text{ Stop; } else \text{ go to Step 2.} \quad (7)$$

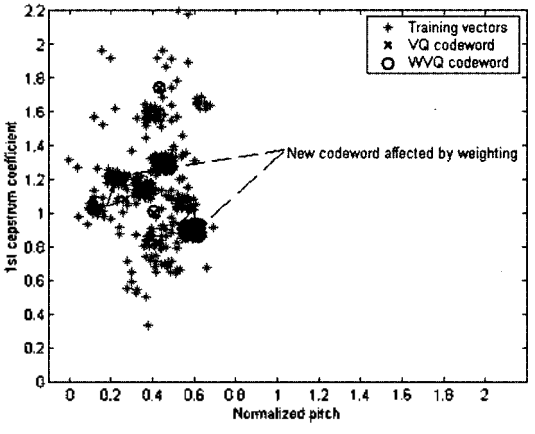
To get the relative frequency of the selected instances,



(a) Occurrence of the triphone /mod/ instances selected by unit selection module



(b) VQ



(c) WVQ

Figure 3. VQ/WVQ results of clustering triphone /mod/ instances.

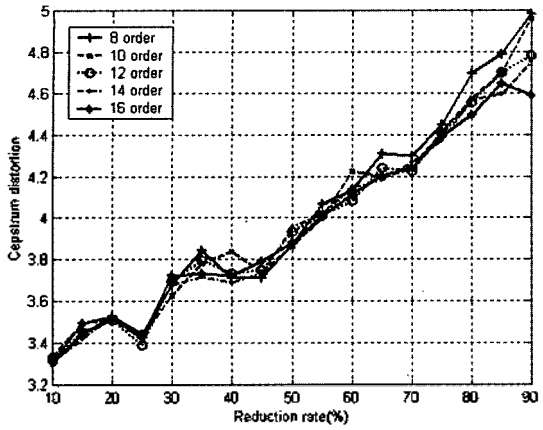
20,000 sentences with the various domains (textbook, broadcasting news, dialogue, and so on) were synthesized. As a result, two million triphone instances were occurred, approximately. The number of occurrence of the selected instances was counted to obtain the weight for each instance.

To verify WVQ algorithm, the result of WVQ and VQ was compared. Figure 3-(a) shows the relative frequency of triphone /mod/ instances. Here, /m/ and /d/ are the preceding and following phonetic contexts of phone /o/, respectively. In Figure 3-(b), all the training vectors and VQ codewords of triphone /mod/ in the pitch-cepstrum (1st coefficient) two-dimensional space is shown. To investigate the effect of weight, we intentionally created the $freq_m$ of additional vectors by adding small random values to original training vectors and overlaid them as shown in Figure 3-(c).

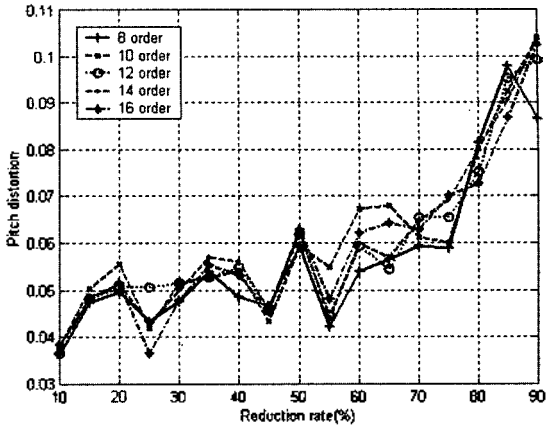
The densely distributed regions indicate that there are the frequently selected instances. The result shows that two of VQ codewords are influenced by the weight and moved to the densely distributed regions.

V. Experimental Results

We have experimented three kinds of pruning methods: *'Limit'* (the number of maximum instances is simply limited), VQ (the conventional method), and weighted VQ (the proposed method: WVQ). At present, our synthesis system has been restricting the maximum number of instances for real-time synthesis. The *'Limit'* version is a baseline performance of our synthesis system. To decide the dimension of feature vectors, we analyzed the performance of VQ as varying feature dimensions (i.e. cepstrum order). In the synthetic speech, the mismatch of pitch and cepstrum are audibly perceptible than that of duration and power. To reduce the feature dimension, the duration and power were excluded. In the results as shown in Figure 4, there are nearly no differences in the performance regarding the cepstrum and pitch distortion. To perform VQ clustering process, 12 order prosodic and spectral feature vectors (i.e. 2 pitch values, 10 order cepstrum

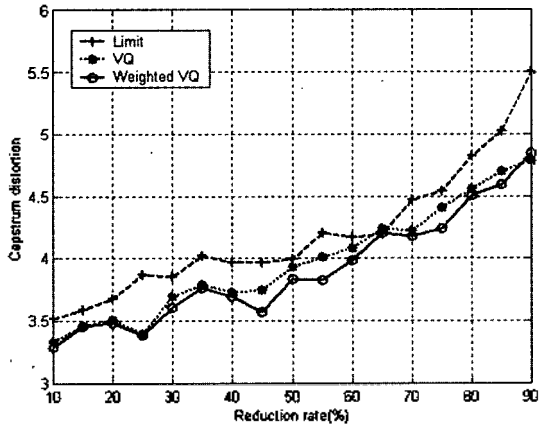


(a) cepstrum distortion

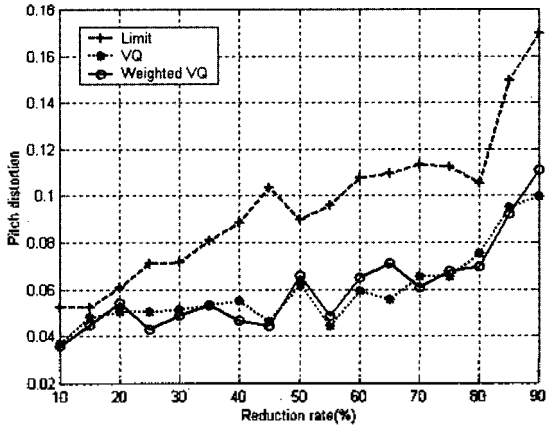


(b) pitch distortion

Figure 4. Performance as varying the dimension of feature vectors.



(a) cepstrum distortion



(b) pitch distortion

Figure 5. Distortion as reduction rate.

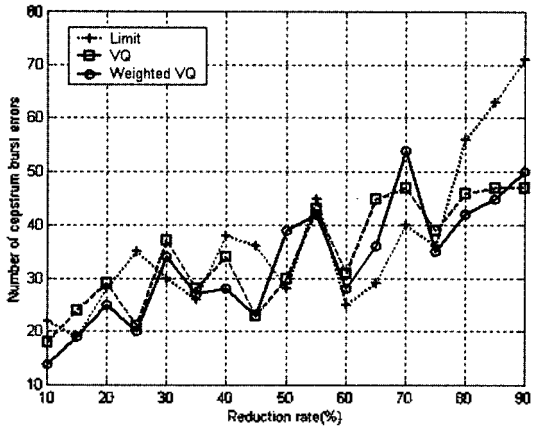
coefficients) were extracted. Then VQ/WVQ process was performed.

To evaluate the performance of the pruning methods, we selected 20 test sentences from 589 phonetically balanced sentences. Then, the synthesis process was conducted to compute the accumulated cepstrum and pitch distortion, which is used for objective evaluation results. The accumulated distortion was averaged as total number of concatenating boundaries. In addition, the number of burst error was considered. The burst error occurs at the drastic mismatches of unit concatenation. Human perception feels badly at burst error. Those objective evaluations were investigated as reduction rate of the synthesis database size.

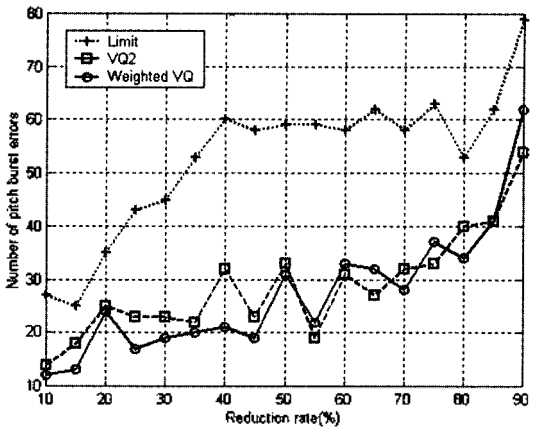
In Figure 5, the averaged distortion of cepstrum and pitch are presented. The WVQ shows better performance than the other methods with regard to cepstrum distortion. In pitch distortion, the WVQ and VQ outperform *'Limit'* but WVQ is roughly the same as VQ.

The results of burst error are shown in Figure 6. The WVQ might be slightly better than the other two methods with regard to cepstrum burst error. The WVQ and VQ show significantly better performance than the method *'Limit'* with regard to pitch burst errors. However, the WVQ shows better results than VQ under 50% reduction rates.

To evaluate subjectively the performance of the pruning methods, informal listening test with respect to 45%

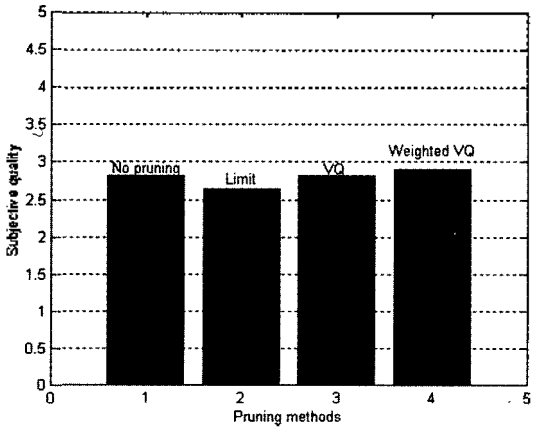


(a) cepstrum burst error

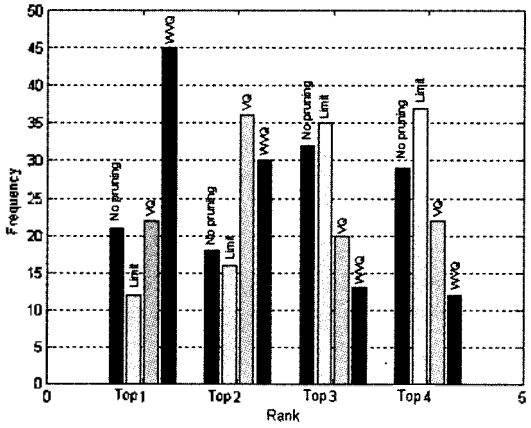


(b) pitch burst error

Figure 6. Burst error as reduction rate.



(a) MOS score



(b) frequency of rank

Figure 7. Subjective evaluation results.

reduction rates was conducted and compared to the full search performance, i.e. '*no pruning*'. The test material was the same as that of the used for the objective evaluation. 5 peoples were participated in MOS (Mean Opinion Score) test. All the participants scored a number ranged from 1(worst) to 5(best). The results of the subjective evaluation are shown in Figure 7-(a). Surprisingly, the WVQ shows slightly better results than '*no pruning*' even if a large reduction rate is adopted. The VQ shows the similar results with '*no pruning*'. The '*Limit*' method results in the worst method that is currently applied to our TTS system. In Figure 7-(b), it shows the number of rank frequency when the scores are sorted. Top 1 means that the participant gave the highest score to the one of the pruning methods. The WVQ shows the highest frequency with regard to Top 1. The '*Limit*' still shows the worst performance.

VI. Conclusion

In this paper, we proposed the weighted VQ pruning method to eliminate the redundant synthesis unit instances from the large-scale synthesis database. It reflected the relative importance of instances in addition to prosodic and spectral contexts. To investigate performance of the pruning methods, we have experimented three kinds of pruning methods (i.e. '*Limit*', VQ, and WVQ) and

compared them by the subjective/objective evaluation. The '*Limit*' version is a baseline performance of our synthesis system. In the results of 45% reduction rates, the weighted VQ shows better results than the conventional VQ, and outperforms the '*Limit*' method. Over 50% reduction rate, the new pruning method doesn't seriously deteriorate the synthetic speech quality.

As further works, we will try to use auditory-based feature (i.e. Mel-cepstrum, Perceptual Linear Prediction) and reduce the feature vector dimension using LDA (Linear Discriminant Analysis). It enables to use the duration, power and high order cepstrum coefficients. The VQ usually needs large training samples. Hence, it may not find good representatives in small training samples. We will manage this drawback of VQ.

References

1. A. Hunt and A. W. Black, "Unit Selection in a Concatenative Speech Synthesis System Using a Large Speech Database," *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, vol. **1**, pp. 373-376, 1996.
2. X. Huang, A. Acero, and J. Adcock, "WHISTLER: A Trainable Text-to-Speech System," *Proceedings of International Conference on Spoken Language Processing*, Philadelphia, vol. **4**, pp. 2387-2390, 1996.
3. A. W. Black, and P. A. Taylor, "Automatically Clustering Similar Units for Units Selection in Speech Synthesis," *Proceedings of Eurospeech97*, vol. **2**, pp. 601-604, 1997.
4. N. Campbell, and A. W. Black, "Prosody and Selection of Source Units for Concatenative Synthesis," A Collection of Technical Publications, ATR-ITL, pp. 45-58, 1996.
5. M. Ostendorf and N. Veilleux, "A Hierarchical Stochastic Model for Automatic Prediction of Prosodic Boundary Location," *Computational Linguistics*, vol. **20**, No. 1, pp. 27-54, 1994.
6. P. A. Taylor and A. W. Black, "Assigning Phrase Breaks from Part-of-Speech Sequences," *Computer Speech and Language*, vol. **12**, pp. 99-117, 1998.
7. S. H. Kim, Y. J. Lee, and K. Hirose, "A New Korean Corpus-based Text-to-Speech System," submitted to *International Journal of Speech Technology*.
8. R. M. Gray, "Vector Quantization," *IEEE ASSP Magazine*, pp. 4-28, 1984.

[Profile]

• Sanghun Kim

Sanghun Kim received the B.S. degree in Electrical Engineering from Yonsei University, Seoul, Korea in 1990 and the M.S. degree in Electrical Engineering from Korea Advanced Institute of Science and Technology, Daejon, Korea in 1992. Since 1992, he has been with Research Department and Spoken Language Processing Section of ETRI, Daejon, Korea. Currently, he is a senior researcher in Speech Database Development Team, Speech Information Technology Research Center of ETRI. His interests include speech synthesis, speech recognition, and speech signal processing.

• Youngjik Lee

Youngjik Lee received the B.S. degree in Electronics Engineering from Seoul National University, Seoul, Korea in 1979, the M.S. degree in Electrical Engineering from Korea Advanced Institute of Science, Seoul, Korea in 1981, and the Ph.D. degree in Electrical Engineering from Polytechnic University, Brookly, New York, U.S.A. From 1981 to 1985 he was with Samsung Electronics Company, Suwon, Korea where he was involved in the development of video display terminal. From 1985 to 1988 his research topic was concentrated on the theories and applications of sensor array signal processing. Since 1989, he has been with Research Department and Spoken Language Processing Section of ETRI, Daejon, Korea pursuing interests in theories, implementations, and applications of spoken language translation, speech recognition and synthesis, and neural network.

• Keikichi Hirose

Keikichi Hirose received the B. E. degree in electrical engineering in 1972, and the M. E. and Ph. D. degrees in electronic engineering respectively in 1974 and 1977 from the University of Tokyo. From 1977, he is a faculty member at the University of Tokyo, and was a Professor of the Department of Electronic Engineering from 1994. In 1995, the University of Tokyo re-organized its graduate school to make it as the main body of the University. From March 1987 until January 1988 he was a Visiting Scientist of the Research Laboratory of Electronics at the Massachusetts Institute of Technology. Although his research interests widely cover the field of speech information processing, such as analysis, synthesis, perception, and recognition, he has major interest on prosody. He is a member of the Institute of Electrical and Electronics Engineers, the Acoustical Society of America, the European Speech Communication Association, the Institute of Electronics, Information and Communication Engineers, the acoustical Society of Japan, the Japan Society of Applied Physics, the Information Processing Society of Japan and the Japanese Society for Artificial Intelligence.