A Study on Design and Implementation of Embedded System for speech Recognition Process

*Jung Hoon Kim, **Sung In Kang, *Hong Suk Ryu, *Sang Bae Lee

* Department of Electronic Communication Eng, Korea Maritime University
** Department of Computer Eng, Tong-myong University of Information Technology

Abstract

This study attempted to develop a speech recognition module applied to a wheelchair for the physically handicapped. In the proposed speech recognition module, TMS320C32 was used as a main processor and Mel-Cepstrum 12 Order was applied to the pro-processor step to increase the recognition rate in a noisy environment.

DTW (Dynamic Time Warping) was used and proven to be excellent output for the speaker-dependent recognition part. In order to utilize this algorithm more effectively, the reference data was compressed to 1/12 using vector quantization so as to decrease memory. In this paper, the necessary diverse technology (End-point detection, DMA processing, etc.) was managed so as to utilize the speech recognition system in real time

Key words : Embedded System, DSP(TMS320C32) , DTW, VQ, Speech Recognition

1. Introduction

Recently, speech-processing technology has seen increasing investment with the development of HCI (Human Computer Interface), and the demand for speech recognition products have also increased. Speech recognition technology allows for the development of products which interpret human spoken language. The application range of this technology has increased in such areas as information technology, digital communication, electronic appliances, multimedia appliances, etc., and is applied to PCS phones, toys, telephones, the Internet, computers or car accessories (navigation system) to be continuously placed on the market. Also, this technology is increasingly being applied to products for the physically handicapped. This study attempted to develop a speech recognition module for motorized wheelchairs so as to provide more convenient functions to physically handicapped persons.[1] For speech recognition, DTW (Dynamic Time Warping) using the pattern matching method and HMM (Hidden Markov Model) using the statistical pattern recognition method were used. Also, the neural network was used for speech recognition, as DTW and HMM recognition algorithms are widely used in industry.[2] The recognition method can be divided into two types: the speaker-dependent type and speaker-independent type. The former can be applied to only one person while the latter can be applied to all persons. The speaker-dependent type is widely used for the DTW recognition algorithm; and the speaker-independent type is proper for the HMM recognition algorithm. The DTW recognition method does not require training and shows excellent recognition performance for isolated words in a limited vocabulary, and it can be used in small size or DSP applications. In this research, the DTW recognition algorithm was used, the most suitable type for the speech-dependent recognition algorithm, and VQ (Vector Quantization) was applied, typically used in data compression for more effective measures to compress words, which were to be saved in the reference, to 1/12. With this, more diverse words can be inputted with less memory, and the recognition rate as well as the cost-competitiveness was expected to improve.[3] MFCC was used in the feature extraction step in order to show strong features against the environment, improving the recognition rate in spite of noise.

Because this study had less white noise components, white noise elimination methods (spectrum subtraction, Wiener filtering, etc.)were not implemented considering the coding efficiency. TI's TMS320C32, a low-cost floating point digital signal processor, was used as the main processor for the system to implement the speaker-dependent speech recognition product using the DTW+VQ algorithm.

This paper is described as follows: In Section 2, the speech signal detection and MFCC is described; in Section 3, the DTW/VQ as a recognition algorithm is

접수일자 : 2003 년 5월 10 일
완료일자 : 2004 년 4월 6 일.

described; in Section 4, the real recognition system is described, and Section 5 shows conclusions found.

2. Endpoint Detection and Feature Extraction

2.1 Endpoint Detection

In the speech recognition phase, speech entered in real-time is distinguished as voice or voiceless sound. If a voice sound is recognized, the data value in each frame of the voice sound is calculated: if it is greater than the set up threshold value, it is considered a voice and is used as a real voice. Namely, speech is processed by analyzing short-time segments while two buffers are allocated in order to save voices entered into the DMA (Direct Memory Access) channel temporarily. Speech is entered into the first buffer, and the other buffer is designed to detect speech in real-time by calculating the energy. [4][5]

$$E_i = \sum_{n=0}^{N-1}[s_i(n)]^2 \qquad \text{Eq.(1)}$$

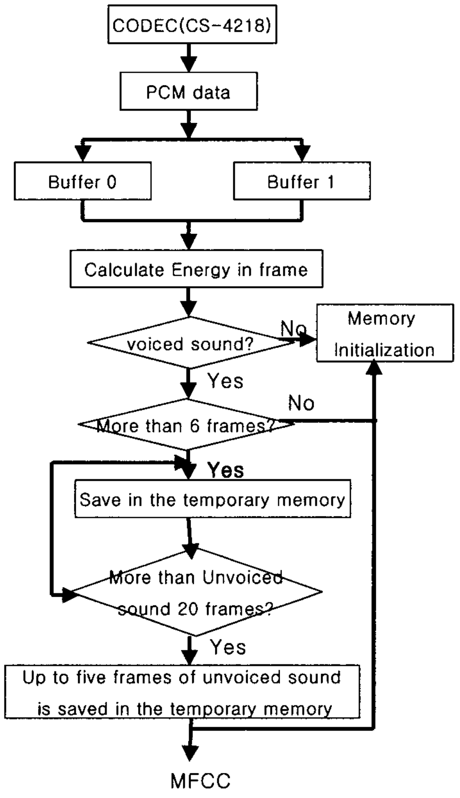

CODEC(CS-4218)
PCM data
Buffer 0
Buffer 1
Calculate Energy in frame
voiced sound?
No
Memory Initialization
Yes
No
More than 6 frames?
Yes
Save in the temporary memory
More than Unvoiced sound 20 frames?
Yes
Up to five frames of unvoiced sound is saved in the temporary memory
MFCC

Fig .1. End point detection for sensing the speech section

In this study, the frame size was set up as 100, and the frame blocking was set up as 30. Energy was calculated by using the absolute energy formula per frame. (Eq. (1) shows the absolute energy formula).

The threshold determines whether the entered value is speech or not, and the threshold value was obtained from the experiment data (here, ITU=50000).

Figure 1 expresses the flowchart of the end-point detection for sensing the speech section. If the speech frame is greater than 6 frames, the corresponding section is considered speech and saved in memory; Later, the voiceless section had more than 20 frames, so the first 5 frames were saved considering that the voiceless section exists in the end-point, and the other 15 frames were ignored. The recognized speech signal passed through the MFCC (Mel-Cepstrum coefficient), the feature-extraction step, and decreased to 12 coefficients per frame.

2.2 Feature Extraction

Among Cepstrum analysis methods, Mel-Cepstrum uses human hearing features. Mel is a unit of recognized pitch or frequency of a tone signal, and does not respond to the physical frequency of the tone signal linearly because the human hearing system does not recognize pitch linearly. Here, we set 1000Hz as 1000mel, and the pitch recognized by humans changed into double the standard frequency expressed as 2000mel. With this method, the responding relationship between the real physical frequency and the recognized frequency was obtained. The filter bank in this research used the filter bank consisting of the log scale, and the relationship displays linearity below the 1kHz range while the logarithm is above the 1kHz range. The relationship between Mel and frequency can approximately be expressed as in Eq. (2):

$$F_{mel} = 2595\log_{10}(1+\frac{F_{Hz}}{700}) \qquad \text{Eq.(2)}$$

Namely, once the pre-emphasis is implemented to the speech signal, the Hamming window is applied. Then, FFT (Fast Fourier Transform) is used to transform it into the frequency domain. If the transformed frequency-domain values pass through the pre-setup filter bank, and DCT (Discrete Cosine Transform), the inverse-transform, is applied, and 12 coefficients are obtained per frame. MFCC is implemented with this filter bank. Figure 2 displays the MFCC process procedure.

Eq. (3) to (5) are used in Figure 2 procedure.

Eq. (3) is the pre-emphasis step, and eliminates

DC elements; Eq. (4) is for the Hamming window; Eq. (5) is the DCT (Discrete Cosine Transformer) equation, and inversely transforms the frequency−domain into the time−domain. [6][7]

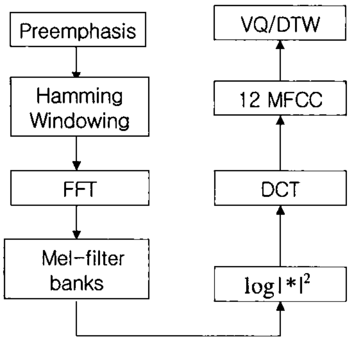


Fig .2 MFCC process procedure

$$\bar{s}(n) = s(n) - as(n-1)\ , \quad (0.9 < a < 1.0) \qquad \text{Eq.(3)}$$

$$w(n) = 0.54 - 0.46\cos(\frac{2\pi n}{N-1})\ , \quad (0 < n < N-1) \qquad \text{Eq.(4)}$$

$$c_n = \frac{1}{20}\sum_{k=1}^{20} X_k \cos\left[n(k-\frac{1}{2})\frac{\pi}{20}\right] \qquad \text{Eq.(5)}$$

3. Recognition Algorithm

DTW for the recognition algorithm in this study used the pattern matching method, and the feature values of the word to be recognized was saved in the reference.

If feature values of a newly entered word is entered, the distance value is obtained by features of the existing words and each DTW recognition algorithm. Among those obtained distances, the least value is determined as the recognition word. In this study, we used vector quantization to compress the feature values into 1/12 so as to decrease memory saved in the reference. Also, the less data there is per word, the faster the DTW distance calculation is; thus the speed is approximately 4 times faster.

In order to compensate for these two problems, vector quantization was used to implement the recognition algorithm effectively.

3.1 Vector Quantization Algorithm

Feature values per word were extracted by implementing speech detection and feature extraction procedures by sounding the words five times each using the DSP board. The feature values obtained here are combined and produced in one file.

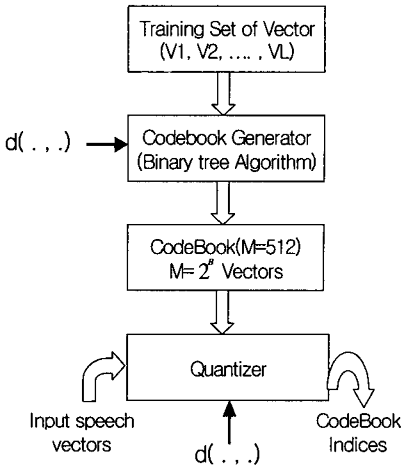

Training Set of Vector (V1, V2, ... , VL)
d(. , .)
Codebook Generator (Binary tree Algorithm)
CodeBook(M=512) M= 2^n Vectors
Quantizer
Input speech vectors
d(. , .)
CodeBook Indices

Fig. 3. General block diagram of vector quantization

With this data, 512 codebooks were created using the vector codebook generation training algorithm on the PC. Figure 3 shows the general block diagram of the vector quantization.

If newly entered feature vectors are input, 12 feature vectors per frame can be compared with the existing quantization table so as to produce the symbol of the closest quantization table. [1][3][6]

3.1.1 Quantization Table Generation Step

In order to generate the quantization table, the K−Means Algorithm or binary tree algorithm is mainly used; in this study, the binary tree algorithm was used to generate the quantization table. Figure 4 shows the flowchart of the binary tree algorithm. We set M (Vector quantization size)=512, the error rate= 0.01 and e (Splitting parameter)= 0.000001.

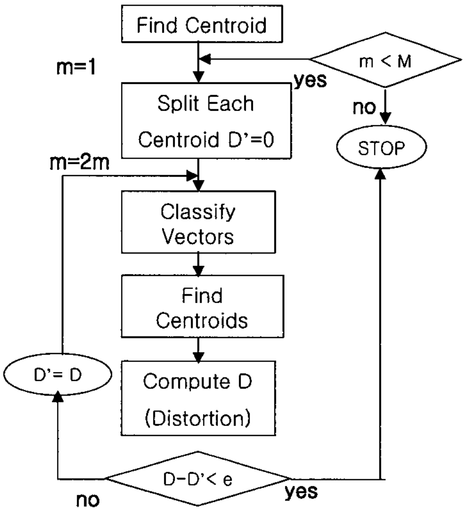


Fig. 4. Flowchart of binary tree algorithm

The binary tree algorithm is split from one vector codebook, and creates M codebooks. Figure 4 shows the sequence of the binary tree algorithm [8]

3.1.2 Vector Symbol Extraction Procedure

This step is to produce the symbol of the optimum codebook between the codebook generated in Figure 3 and the newly entered vectors. M is 512, and the symbol of the optimum codebook symbol is expressed in Eq. (6):

$$M^{*}=\arg\min_{1<m<M} d(v,ym)\qquad \text{Eq.(6)}$$

M−dimensional vector codebook: ym, (1<ym<512)
Feature−extracted entered vector value : v
Optimum vector index matrix: m*

3.2 DTW Recognition Algorithm

DTW obtains similarity between the standard speech signal pattern and the inputted speech signal by using dynamic programming. Namely, this method compensates for the difference on the time−axis. Let us say that the inputted speech pattern with length M is T=T(1),T(2),... ,T(M); the standard pattern with length N is R=R(1),R(2),R(3)..., R(N); then, the similarity between the two patterns can be described in Eq. (7):

$$D=\sum_{N}^{n=1} d(R(n),T(W(n)))\qquad \text{Eq.(7)}$$

Here, d(R(n), T(w(n))) is R's n^{th} and T's w(n) th local distance; DTW finds the optimum path m=w(n) of (m,n) plane optimizing the accumulated distance between the two patterns. When the optimum path is found using this method, the following five constraint conditions are added considering the speech signal features in order to reduce search time.

- Endpoint Constraints
- Monotonicity
- Local Path Constraints
- Global Path Constraints
- Slope weighting

If the start point is (1,1); the end point is (Tx,Ty), the optimum path can be obtained as below:

1. Initialization : Da(1,1)=d(1,1)m(1)
2. If 1≤ix≤Tx and 1≤iy≤Ty, then
   Da(ix,iy)=min(ix,iy) [Da(i'x,i'y)+ζ((i'x,i'y),(ix,iy))]
3. Termination : d(x,y)=Da(Tx,Ty) / Mϕ

In this paper, we used the Euclid method for local distance, and the ITAKURA method to obtain the local minimum.[1][5][9]

5. Configuration of Recognition System and Result of Examination

5.1 Recognition System

The overall configuration of the recognition system is shown in Figures 5 and 6. Figure 5 is the hardware block diagram and consists of 16−bit CODEC and TMS320C32 floating point DSP, SRAM(32K*2) and PPI 8255. The EPROM memory interface uses one of the Strb0 signals for 8−bit data access; RAM is designed to access the 32−bit data and uses four Strb1 signals.[10]

Total RAM capacity should be enough for speech, so we used 128Kb by using four 32kb. The 16−bit CODEC delivers an analog signal to the DSP by sampling speech into 16−bit and sending it to the DSP through serial communication. CS−4218 was used for the CODEC, 8kHz was the sampling frequency and the resolution was 16.

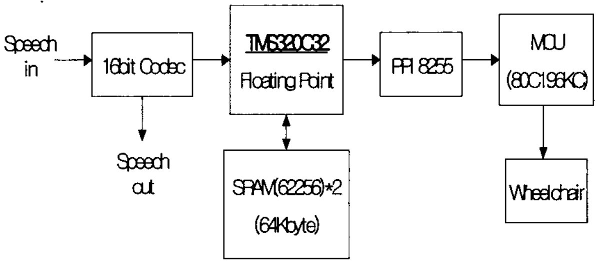


Fig .5. Speech recognition hardware configuration

According to the switch on the speech recognition board, it can be divided into recognition mode and learning mode: if S2 (S1: Reset) on the board is pressed, the learning mode is activated, and the patterns of the commands are inputted into the reference. Then, S3 is pressed to activate the recognition mode to save the speech, calculating each distance using already saved pattern words and the VQ/DTW recognition algorithm.

The total process steps are described in Figure 6. Here, among the compared values in the final decision logic, the least distance word is outputted as the recognition word. The actual figure of the implementation of Embedded System for speech recognition is shown in Figure 7

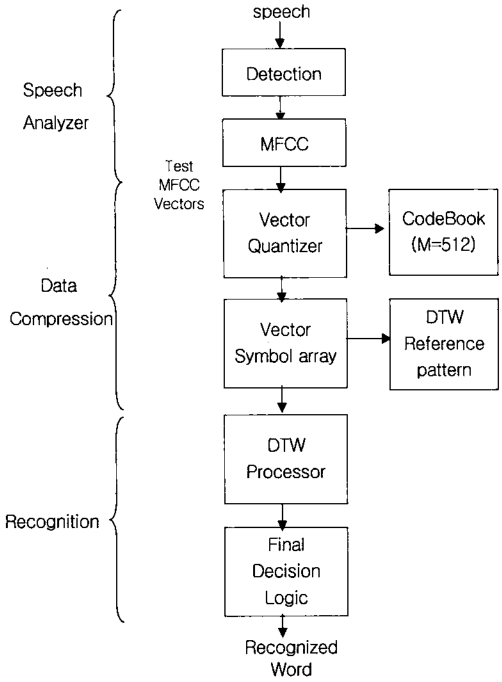


Fig .6. General software block diagram of speech recognition

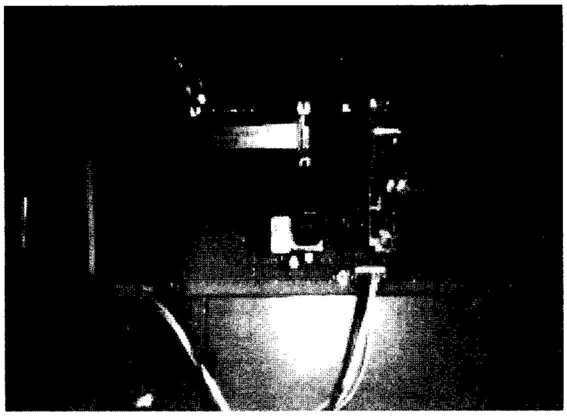


Fig. 7. Real picture of wheelchair system and speech recognition test

5.2. Experiment and Examination

The VQ/DTW recognition algorithm suggested in this research was compared with the traditional DTW recognition algorithm. Tables 1 and 2 show the recognition rate and the average execution speed of eight Korean wheelchair commands (Korean words for "Forward", "Backward", "To the Left", "To the Right", "Stop", "Turn", "Faster" and "Slower") to be recognized by the system after testing them 30 times each. The commands were voiced 10 times per word; thus a total of 80 words (8 x 10) were saved in the reference.

Table 1. Recognition rate and execution speed of traditional DTW algorithm

Word	Test number	Recognized number	Recognition rate	Speed (second)
Forward	30	30	100%	1.4
Backward	30	29	96.6%	1.1
Left	30	28	93.3%	1.0
Right	30	28	93.3%	1.1
Stop	30	30	100%	1.0
Fast	30	29	96.6%	1.1
Slow	30	30	100%	1.4
Rotate	30	30	100%	1.0

Table 2. Recognition rate and execution speed of VQ/DTW algorithm

Word	Test number	Recognized number	Recognition rate	Speed (second)
Forward	30	30	100%	0.4
Backward	30	28	93.3%	0.3
Left	30	28	93.3%	0.3
Right	30	28	93.3%	0.3
Stop	30	29	96.6%	0.3
Fast	30	29	96.6%	0.3
Slow	30	30	100%	0.4
Rotate	30	28	93.3%	0.3

- Forward : "A-p-ro", Backward : "Dui-ro"
- Left : "Jwa-ro", Right : "U-ro"
- Stop : "Jeong-ji", Fast : "Bbal-ri"
- Slow : "cheon- cheon-hi"
- Rotate : "Hoe-jeon"

6. Conclusion

In this study, a method was suggested to compress data with vector quantization in order to decrease the memory and the execution speed, utilizing the DTW recognition algorithm, one of the most commonly used recognition algorithms as speaker-dependent, more effectively. Also, we applied this recognition system to a wheelchair for the disabled. After testing the wheelchair-related commands in the recognition test, we obtained a 90% recognition rate. Furthermore, wheelchair driving performance and environment safety was confirmed.

For improvement, more study on noise, and the implementation of a recognition system with HMM and a neural network should be sought so as to improve the speaker-independent system. To improve wheelchair security, we expect more functions to sense obstacles as well as avoid them.

References

[1] L.R.Rabiner, B.H.Juang, "Fundamentals of Speech Recognition", Prentice Hall, 1993

[2] Lawrence Rabiner, "A Tutorial on Hidden Markov Models and Selected Application in Speech Recognition", Proc. IEEE, Vol.77, No. 2, February 1989.

[3] Lawrence Rabiner, "On the Application of Vector Quantization and Hidden Markov Models to Speaker Independent Isolated Word Recognition", Bell System Technical Joural, Vol. 62, No.4, April 1983.

[4] Ji Hong Lee & Seoil DSP Technology Research, "Applications of DSP chip", Seoil DSP Co., Ltd.

[5] Ik Joo Jeong, Hoon Jeong, "Implementation of Real-time Speaker-dependent Speech Recognition Hardware Module (VR32) using TMS320C32 DSP", Acoustical Society of Korea, Vol.17., No.4.14 22, 1998.

[6] Kang Joo Yoo, "A study on Data Fusion of DHMM-Based Korean Digits Recognition", Master's Thesis of Engineering Dept of Korea Maritime University, 1998.

[7] Tae Han Lee, "Improvement of performance of Speech Recognition System for Vehicle Navigation using Commonplace DSP", Master's Thesis of Engineering Dept of Yonsei University, 1999.

[8] www.data-compression.com

[9] Chang Keun Kim, Hak Yong Han,"Implementation of Real-time Speech Recognition Wireless Automobile using TMS320C32", KISPS, 2001.

[10] Jung Hoon Kim, Hong Suk Ryu, Sung In Kang, "Design and Implementation of Multi-functional WheelchairSystem with Speech Recognition", KFIS, Vol.12, No.1, pp1-5, 2002.

저자소개

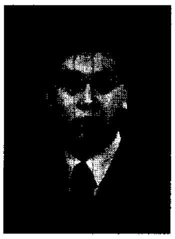
김 정 훈

2001 년 : 동명정보대학교 정보통신공학과(공학사).

2003 년 : 한국해양대학교 전자통신공학과 (공학석사).

현재 : 한국해양대학교 전자통신공학과 박사과정

관심분야 : 음성인식, DSP, 인공지능



강 성 인

1997 년 : 한국해양대학교 전자통신공학과(공학사).

1999 년 : 한국해양대학교 전자통신공학과(공학석사).

2004 년 : 한국해양대학교 전자통신공학과(공학 박사)

현재 : 동명정보대학교 강의전담 교수

관심분야 : DSP, 인공지능, 지능제어

류 홍 석

2002 년 : 동명정보대학교 정보통신공학과(공학사).

2004 년 : 한국해양대학교 전자통신공학과(공학석사).

현재 : ㈜임펙 근무 중.

관심분야 : 음성인식, DSP

이 상 배

1989 년 : 고려대학교(공학박사).

현재 : 한국해양대학교 전자통신공학과 정교수