2000년도 한국음향학회 학술발표대회 논문집 제19권 제1(s)호

Car Navigation용 음성합성시스템 최저가 구현

나지훈*, 성정모**, 양윤기*

* 수원대학교 정보통신공학과, ** (주) 알파텔레콤

Low-cost implementation of text to speech(TTS) system for car navigation

Ji Hoon Na*, Jung Mo Sung**, Yoon Gi Yang*

* The University of Suwon, ** ALPHATELECOM CO.

na@csp.suwon.ac.kr, jmomo@alphatelecom.co.kr, ygyang@mail.suwon.ac.kr

요 약 문

최근에 무선통신망을 이용한 데이터 서비스가 폭넓게 제공되면서, 이동체(MS:mobile station)에 대한 위치정보나 교통상황 등의 부가 정보 서비스가 제공되고 있다. 이와 같이 이동체가 자동차와 같은 운행수단일 때 사용자가 디스플레이 되는 문자정보를 확인하게 되면 운전의 안정성이 저하되어 실용적이지 못하다. 이를 위해서 문자를 음성으로 전환하여 주는 문자-음성변환기(text to speech : TTS)가 필요하다. 본 논문은 car navigation용 '한국어 무제한 어휘 음성합성기' 를 저가의 DSP chip(ADSP-2185)과 저용량의 4M bits ROM을 사용하여 low-cost system으로 하드웨어를 구성하였다. 본 연구에서 개발된 실시간 한국어 음성 합성기는 저가의 통신 단말기로서 사용 될 수 있으나, 반음절 연결부분의 연결이 불완전한 경우가 많았다. 그러나 종성이 없는 음절에 대해서는 명료도가 비교적 우수 하였다.

I. 서 론

한국어 무제한 어휘 음성합성기는 멀티미디어 기술의 비약적인 발전에 힘입어 꾸준한 발전을 거듭하고 있다. 음성합성의 방법을 분류하면 몇 개의 단어를 녹음하는 녹음편집, 녹음 편집된 음성을 파라미터로 저장하여 출력하는 분석합성, 음소나 음절 등의 언어의 기본 파라미터를 이용하여 언어 규칙으로부터 합성하는 규칙합성의 세가지 방식으로 대별할 수 있다. 그런데 본 연구에서는 DSP보드에 제한된 크기의 ROM을 사용하면서도 무제한 어휘를 합성해야 하는 까다로운 조건이 있다. 이러한 요구를 충족시킬 수 있는 방식으로는 LPC, LSP를 사용하는 규칙합성이 있다. 이 방식은 낮은 bit rate를 사용하고 반음절들을 연결하는 방식이어서 합성음질의 품질이 그다지 우수하지는 못하다. 그러나 합성품질의 명료성은 크게 나쁘지 않아서 본 연구와 같은 응용에 사용될 수 있을 것이라 판단된다.

일반적으로 한국어 음성합성에 사용되는 합성단위는 음소, 반음절, 음절, diphone등이 있다. 본 논문에서는 반음절 단위의 합성방식을 택하였다. 반음절 합성의 경우 한국어 음절은 초성자음+중성모음+종성자음 으로 구분되는데 중성모음의 경우 상당히 일정기간 stationary 한 특성을 유지하므로 이를 중심으로 음절을 두 부분으로 분해하여 저장하면 매우 효과적이다. 즉, 음절핵(syllable core)을 중심으로 해서 그 전반부와 음절 끝의 자음부로 된 suffix 라고 부르는 단위를 사용한다. 한국어의 경우 총 567개의 반음절로 구성하면 매우 효과적임이 알려져 있다.

II. 시스템의 구성

본 논문에서 구현한 TTS를 사용하여 Car navigation용 음성합성시스템의 모델을 그림1에 제안하였다. 그림2에는 개발된 문자-음성변환기(TTS)의 블록다이어그램

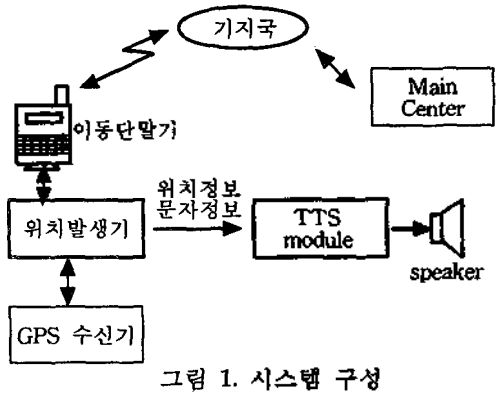


그림 1. 시스템 구성

이 제시되어 있다.

본 연구에서 개발하고자 하는 하드웨어는 위치발생기나 GPS수신기 등의 통신장비와 인터페이스 할 수 있는

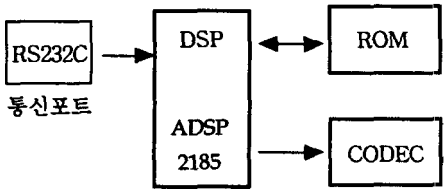


그림 2. 개발된 음성합성기(TTS)

시스템을 기본으로 한다. 즉 하드웨어의 형태는 상업용 저가의 DSP를 장착한 독립적인 보드를 기본 골격으로 하며 serial 통신을 통하여 한국어 단어 명령을 전달받아 이를 음성으로 전환하여 스피커로 전환하는 시스템을 구현하는 것을 목표로 한다. 개발된 TTS는 크게 CPU, ROM, Codec, RS232c interfacing 으로 대별할 수 있다. CPU로는 analog device 사의 DSP(digital signal processor) 인 ADSP-2185를 사용한다. ROM은 반음절 한글 database의 크기를 고려하여 볼 때 약 3 M-bits가 필요하므로 4 M-bits 인 27c040을 사용한다. Codec은 analog device 사의 AD73311을 사용하였다. DSP와 Codec 사이의 통신은 ADSP-2185의 serial 통신포트를 통하여 이루어 진다. Speaker 부분은 LM386을 사용한 저전력 회로이다. RS232를 위한 interfacing 으로는 ADM232을 사용하여 RS232 신호를 생성한다. 회로의 확장을 위하여 3개의 커넥터를 연결하였다. 첫 번째 커넥터는 9 핀의 D-type 커넥터로 주 프로세서와 +5V 의 전력과 RS232 신호를 전달받는데 사용된다. 10-pin 커넥터는 ADSP-2185 의 general purpose I/O 인 PF0-PF7을 외부로 보내기 위한 커넥터이며, 14-pin 커넥터는 ADSP 의 in circuit emulator 인 EZ-ICE를 시험하기 위한 포트이며, 이 핀의 배열은 EZ-ICE에 준한다.

ADSP-2185는 address space가 14 bits 이므로 총 16K words의 data addressing 이 가능하다. 따라서, 4-M bits의 외부 ROM을 access 하기 위해서는 특별한 주의가 필요하다. ROM의 역할은 power up이 발생될 때 program이 download 되는 boot ROM 의 역할과 다량의 음성 데이터 베이스를 저장하는 두가지 역할을 한다. 이를 위해서는 ROM의 address를 page 기법을 사용하여 분할하여 access 하여야 하는데, ADSP-2185의 address pin이 14 bits 이므로 16384 bytes 영역을 ROM의 한 page라 정의하면 편리하다.

먼저, boot-ROM 으로서는 ADSP-2185 의 내부 program memory의 word-length 가 24bits 이므로 총 16384개의 program memory를 boot ROM 으로 채우려면, 3개의 ROM page 가 필요하다. 따라서, 음성 데이터 베이스는 4번째 page부터 저장 되는데, 자세한 음성 데이터 베이스의 크기와 저장 방법은 IV장에서 서술하기로 한다. ADSP-2185 에서는 ROM을 access 하는 방법으로 BDMA(byte memory direct memory access)를 사용한다. 이 중에서 program memory로 전달될 page0-page2 외 3 page는 power-up mode에서 자동으로 내부 program 메모리에 BDMA port를 사용하여 download 된다. 나머지 음성데이터 베이스는 프로그램에서 필요에 따라서 적절한 위치의 데이터를 BDMA port를 사용하여 access 한다. DSP assembler 로 컴파일된 프로그램 코드의 길이는 24 비트이므로 이를 ROM에 저장하려면 한 word를 3개의 연속된 byte로 분리 하여주는 PROM-splitter 가 사용된다.

이와같이 구현된 시스템의 board 사진을 그림 3에 제시하였다.

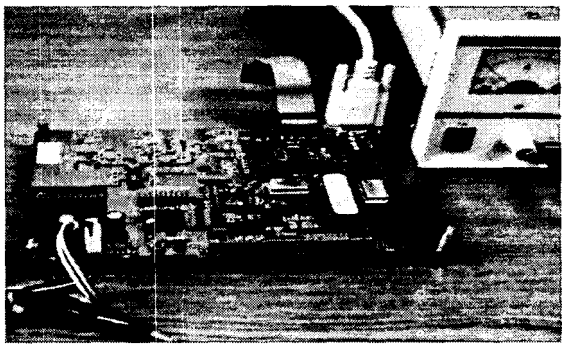


그림 3. 개발된 TTS board 사진

III. 실시간 음성합성

음성 합성을 방식별로 분류 하여보면 첫째로 음성파형을 이용한 방식, 둘째로 선형예측 파라미터를 사용한 방식, 셋째 포만트를 이용한 방식 그리고 조음파라미터를 사용한 방식을 들 수 있다. 이 중에서 비교적 구현이 용이하고 하드웨어의 복잡성이 적은 것이 선형예측 방식이다. 따라서, 메모리의 제한이 있는 본 연구에서는 선형예측 파라미터방식을 사용하기로 하였다. 선형예측 파라미터를 사용한 방식은 LPC(Parcor), LSP(line spectrum pair), cepstrum 방식이 있다. 이 방식은 비교적 간단한 구조와 일정 수준의 음질확보가 용이한 반면 합성음의 자연성이 취약한 점으로 지적받고 있다.

본 시스템에서는 음성 데이터 베이스가 ROM에 저장되어 있어야 한다. 따라서 LPC 계수를 off-line 으로 추출하여 저장 하여야 한다. 본 연구에서는 Matlab을 사용하여 LPC 계수를 구하였다. 음성 데이터는 8bits, 8000Hz로 샘플링된 546개의 반음절 데이터를 사용한다. 한 반음절의 길이는 0.6초로 총 5600개의 샘플이 사용된다. LPC는 한 프레임에서 음성 신호가 stationary 하다고 가정하고 이 구간에서 신호를 선형으로 모델링하고 이 계수를 전송하는 것이다.

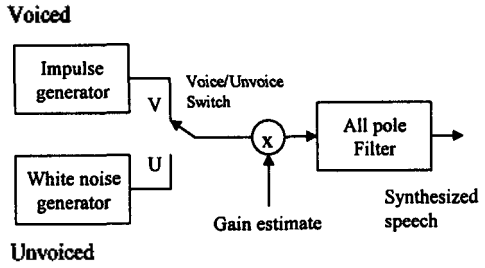


그림 3. speech production model

그림 3 에 LPC에 사용되는 모델이 제시되어 있다. 그림에서 프레임별로 유성음/무성음을 구별하고 유성음이면 피치의 주기와 동일한 impulse 신호를 입력으로 하고 무성음이면 백색잡음을 all-pole filter 의 입력으로 한다. 입력과 gain이 곱해져서, LPC 계수로 구성된 all-pole filter를 통과한 신호가 합성된 신호가 된다. 즉, LPC 계수는 피치, gain, all-pole filter의 계수등으로 구성된다. All-pole filter의 차수는 보통 10차 정도를 사용하는데, 이는 선형 예측기법에서 도출되는 것이다.

본 연구에서는 180개의 샘플을 한 프레임으로 정의하였다. 따라서 프레임의 길이는 180*0.125 = 22.5 msec 이다. 그런데, 보통은 Hamming window 등과 같이 windowing 기법을 사용하는데, 그 이유는 한 프레임에서 충분히 stationary 한 신호를 추출하기 위함이다. 본 연구에서는 window의 길이를 240샘플, 즉 30msec 으로 정하였고, window의 중심에서 180개에 해당하는 LPC 계수를 추출하였다. 사용된 LPC 의 차수는 10 차이며, 피치 검출을 위하여 최대 12.5 msec 까지의 범위에서 피치를 검출하였다.

반음절의 길이는 약 0.56초이므로 25프레임 길이의 LPC 계수를 추출하였다. 물론 음절별로 길이가 다르기 때문에 실제적인 음절의 길이를 별도로 저장 할 필요가 있으나, 데이터를 access 하기 쉽게 하기 위해서 모든 반음절에서 25프레임의 LPC 계수를 구하여 ROM에 저장하였다. 총 546 개의 반음절이 있으므로 전체 반음절 데이터베이스의 크기는 546*16*12*25 = 2,620,800 bits 이므로 27c040 ROM을 사용하면 충분하다. 실시간 음성합성을 위하여 프레임 당 LPC 계수를 찾아와서 22.5msec 단위의 합성을 하는데, 기본적인 원리는 그림 3 와 같다. 여기서 주의할 점은 연속적인 부드러운 음을 얻기 위하여 반사계수는 인접한 프레임들의 반사계수를 보간 하여 얻어진다는 점이다. 또한 두 프레임간이 피치가 연속적으로 부드럽게 연결되게 하기 위해서 pitch - synchronization 기법을 사용한다. 이를 실시간으로 처리하는 것은 DSP program에서 인터럽트를 사용하는데, DSP가 Codec 과 직렬로 연결되어 있으므로 직렬포트를 샘플링 주파수에 맞추어 실기간으로 합성음을 출력 시킨다.

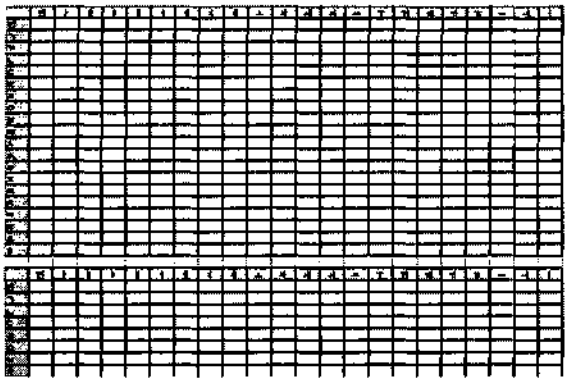
Ⅳ. 한국어 음성합성

본 연구에서는 조합형 문자로 입력된 한글을 분석하여 두 개의 반음절을 분석한다. 조합형 한글은 2-byte 로 구성되는데 MSB 는 항상 1 이고, 1aaaaabbbbbccccc 와 같이 aaaaa, bbbbb, ccccc 는 각각 초성, 중성, 종성의 5 bit 이다. 한국어 합성은 위와 같은 조합형 한글의 분석을 통하여 해당 반음절을 연결 시키는 방식으로 요약될 수 있다. 그러나, 실제로 발음되는 음운은 문자와 서로 다를 수 있다. 즉 음운현상들에 의해 실제 발음되는 반음절을 찾아야 할 필요성이 있다. 음운 현상으로서는 구개음화, 끝소리 규칙, 자음축약, 연음규칙, 자음동화, 경음화등 있다. 이를 위한 발음변환표가 구성되어 있으나, 이는 앞 뒤 음절의 성분에 따라 바뀌는 비교적 복잡한 양상을 보인다. 본 연구에서는 기본적으로 처리할 끝소리 규칙을 구현하였는데, 끝소리 규칙은 음절의 끝소리는 'ㄱ,ㄴ,ㄷ,ㄹ,ㅁ,ㅂ,ㅇ' 의 7개의 자음만을 발음되는 것을 말한다 [10]. 따라서, 전체적인 음성합성은 조합형 한글을 분석하여 초성, 중성, 종성을 분리한 다음 19개의 초성, 21개의 중성으로 구성된 첫 반음절과 21개의 중성과 7개의 대표 종성인 'ㄱ,ㄴ,ㄷ,ㄹ,ㅁ,ㅂ,ㅇ' 인 두 번째 반음절 database를 구한후 두 반음절을 연결하여 합성하게 된다.

반음절 database의 작성은 indexing을 편리하게 하기 위하여 한국어 반음절 database 표를 작성하였다. 총 399개의 첫 번째 반음절과 147개의 두 번째 반음절이 있으므로 이를 ROM에서 쉽게 접근할 수 있도록 indexing 하였다. 표 1 의 반음절와 indexing은 첫번째 열부터 행 순서로 하였는데, 첫번째 반음절의 경우 첫 번째 열의 반음절인 '가, 까, ... 하' 의 번호는 각각 '1, 2, ... , 19' 가 되고, 두 번째 열의 반음절인 '개, 깨, ... 해' 의 번호는 각각 '20, 21, ... , 38' 이 된다. 따라서, 첫 번째 반음절의 마지막 원소인, '히' 의 번호는 '399' 가 된다.

두 번째 반음절의 번호는 '나 =400, 난 =401,...' 과 같이하여 마지막 열의 'ㅇ=546' 과 같이 하였다. 이러한 반음절 indexing에 대한 LPC 계수는 반음절 database 1-546 까지 순서대로 정렬되어 있다. 앞에서 설명한 대로 LPC계수는 피치, 이득, 반사계수로 구성되어 있다. 녹음실에서 저장한 546개의 반음절 데이터 베이스는 약 0.6초 정도 각 반음절로 나누어지고, 25 프레임의 LPC 계수가 저장된다. LPC 계수의 저장순서는 이득, 피치, 반사계수의 순서이다. 각각의 반음절의 절대적인 길이 역시 데이터베이스화 된다.

표 1. 한국어 반음절 database 표



한국어의 음성합성은 조합형 한글의 분석, 음운법칙의 적용, 반음절 LPC 데이터베이스의 수집, 그리고 음절합성 등의 순서로 이루어진다. 보다 개선된 음질을 얻으려면 보다 많은 음운법칙의 적용, 피치값의 부드러운 연결등의 기법들이 필요하다. 그림 5 에 한글 음성합성에 대한 개요도가 제시되어 있다.

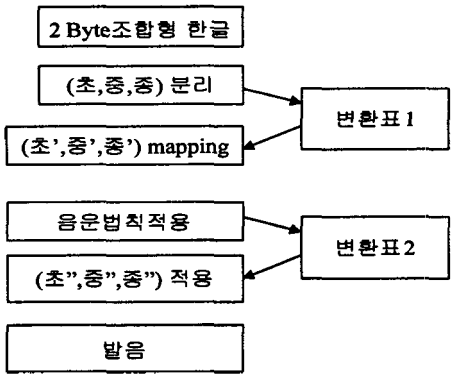


그림 5. 한글 음성합성 처리의 개요도

합성의 과정을 살펴보면 먼저 상용 조합형 한글에서 (초,중,종) 세개의 음소를 추출한다. 텍스트 문장에서 한글 이외의 문장부호, 문단부호 등을 추출하여 이 부분을 묵음처리 한다. 그 다음 표 1와 같은 반음절 데이터베이스의 index를 추출하기 위하여 두 번의 변환표를 거치는데, 이는 '끝소리규칙' 의 음운법칙을 적용하는 과정이다. 이와 같이 구하여진 두 개의 반음절은 같이 각각의 반음절의 길이를 데이터베이스에서 구하여 이를 연결 시키는데, 종성이 있는 경우는 중성을 보다 많이 중복하여 결합시키고, 없는 경우는 첫 번째 반음절을 바로 출력시킨다. 위와 같은 일련의 과정은 DSP program에서 일괄적으로 실시간으로 처리된다.

V. 평가 및 결론

본 연구에서는 실시간으로 동작하는 한국어 음성합성기를 작은 메모리를 사용하여 구현하기 위해서 PC와 같은 고성능 장비에서 적용되는 합성기에서 적용되는 수많은 음운처리 기법들을 잘 사용하지 못한 단점이 있다. 합성기를 통하여 합성한 음질을 고찰하여 보면 반음절 연결부분의 연결이 불완전하여 자연도에서 미흡한 결과를 얻었다. 이를 위해서는 보다 다양한 반음절 연결방식에 대한 결과를 구현할 필요가 있다. 그러나, 종성이 없는 음절에 대해서는 명료도가 비교적 우수 하였다.

한국어 반음절 데이터를 8kbps 의 LPC 계수로 저장하여 약 2.5 Mbits 정도의 메모리가 사용되었다. 개발된 한국어 음성합성기는 저가의 DSP를 사용하였으므로 위치정보발생기 등의 문자정보 통신단말기에 사용하여 저가의 단말기로서 사용될 수 있을 것이다.

참 고 문 헌

[1] J. R. Deller et. al, **Discrete-Time Processing of Speech Signals**, Prectice-Hall, 1987.

[2] Jan P. H, Van Santen et. al, **Progress in Speech Synthesis**, Springer 1995.

[3] G. Bailly and C. Benoit, **Talking Machine, Thoery, Models and Design**, North-Holland, 1992.

[4] T. Dutoit, **An Introduction to Text-To-Speech Synthesis**, Kluwer Academic Publishers, 1997.

[5] Oppenheim and Shafer, **Discrete Time Signal Processing**, Prentice-Hall, 1989.

[6] **ADSP-2100 Family User's Manual**, Analog Device, 1995

[7] **Digital signal processing applications: Using the ADSP-2100 family**, Prentice-Hall, 1995.

[8] **ADSP-2100 Family EZ-kit reference manual**, Analog Device

[9] "한국어 음성합성 및 인식 시스템의 구현", 현대전자산업, 1992

[10] 양윤기, "ADSP2181을 사용한 음성합성기의 구현", 내부보고서, part, 1,2,3 지오텔리콤, 1998.

[11] "한국어 실시간 음성합성기의 구현", 연구보고서, 수원대학교산업기술연구소,지오텔리콤, 1998.

[12] 안승권, "한국어 문자-음성변환 시스템에서 합성음의 자연도향상기법에 관한 연구," 서울대학교 박사학위 논문, 1992년