

# 음성기술을 이용한 십자말 게임

유 일 수<sup>†</sup> · 김 동 주<sup>†</sup> · 홍 광 석<sup>††</sup>

## 요 약

본 논문에서는 음성으로 동작하는 십자말 게임을 구현하였다. 십자말 게임에 사용되는 문제의 배열은 본 논문에서 제안한 CAA(Cross Array Algorithm)에 의해 생성된다. CAA는 영역별 사전을 이용하여 십자말 배열을 매번 랜덤하게 자동으로 생성한다. CAA에 의한 배열 생성을 위해 본 논문에서는 7개 영역에 대한 사전을 구축하였다. 구현된 십자말 게임은 마우스나 키보드뿐만 아니라 음성으로 동작하도록 설계되었다. 음성에 의한 인터페이스는 음성인식 및 합성 기술이 사용되었으며, 사용자에게 보다 편리한 기능을 제공한다. CAA의 성능평가는 십자말 배열을 생성하는데 소요되는 연산시간의 측정과, 십자말 배열의 단어 생성율을 측정함으로써 수행되었다. CAA의 성능 평가 결과, 모든 창에 대하여 연산시간은 약 10ms 내외였으며, 단어 생성율은 약 50%를 보였다. 또한, 음성인식 실험 결과는 각 창의 크기가 "7×7", "9×9", "11×11" 일 때, 각각 98.5%, 97.6%, 96.2%의 인식률을 보였다.

## Crossword Game Using Speech Technology

Il-Soo Yu<sup>†</sup> · Dong-Ju Kim<sup>†</sup> · Kwang-Seok Hong<sup>††</sup>

## ABSTRACT

In this paper, we implement a crossword game, which operate by speech. The CAA (Cross Array Algorithm) produces the crossword array randomly and automatically using a domain-dictionary. For producing the crossword array, we construct seven domain-dictionaries. The crossword game is operated by a mouse and a keyboard and is also operated by speech. For the user interface by speech, we use a speech recognizer and a speech synthesizer and this provide more comfortable interface to the user. The efficiency evaluation of CAA is performed by estimating the processing times of producing the crossword array and the generation ratio of the crossword array. As the results of the CAA's efficiency evaluation, the processing times is about 10ms and the generation ratio of the crossword array is about 50%. Also, the recognition rates were 98.5%, 97.6% and 96.2% for the window sizes of "7×7," "9×9," and "11×11" respectively.

키워드 : 십자말 게임(Crossword game), 음성인식(Speech recognition), 음성합성(Speech synthesis), 배열 알고리즘(Array algorithm)

### 1. 서 론

사용자와 컴퓨터의 보다 편리한 인터페이스를 위하여 음성인식 및 합성을 이용한 사용자 인터페이스는 최근 널리 연구되고 있다. 인간과 컴퓨터의 인터페이스로 키보드나 마우스가 많이 사용되고 있지만, 보다 자연스럽게 편리한 음성을 이용한 인터페이스 기술이 적극적으로 개발되고 활용되고 있다. 음성을 이용한 인터페이스 기술은 여러 분야에 적용되어 서비스되고 있지만, 응용분야의 부족 및 많은 개선점이 필요하다. 이에 본 논문에서는 음성 기술의 한 응용 분야로 일반인들이 널리 즐길 수 있는 십자말 게임을 선택하였다. 기존의 십자말 게임은 신문, 잡지, 인터넷 분야[1,2]에 응용되어 활용되고 있다. 하지만 기존의 십자말 게임의 경우, 게임에 필요한 문제의 배열을 사람이 모두 만들어야

하는 노력이 필요하다. 이점은 사용자가 다양한 형태의 문제를 접할 수 있는 기회를 줄어줄게 한다. 또한 게임을 위해서 키보드와 마우스를 이용한 텍스트 위주의 인터페이스를 사용하기 때문에 다소 불편한 점이 존재한다.

본 논문에서는 기존의 십자말 게임의 기능을 향상시키기 위하여 영역별 사전을 이용하여 십자말 배열을 자동 생성하는 기능과 음성을 이용한 인터페이스 기술을 구현하였다. 기존의 십자말 게임에서 문제 배열을 수작업으로 구성해야 하는 어려운 점을 개선하기 위하여, 본 논문에서는 영역별 사전의 구축을 통하여 십자말 배열을 자동 생성하는 알고리즘을 제안하여 구현하였다. 십자말 배열을 자동으로 생성하는 알고리즘은 영역별 사전이 이용되며, 이를 위해 본 논문에서는 7개 분야에 대한 영역별 사전을 구축하였다.

기존의 십자말 게임 인터페이스는 신문이나 잡지의 경우 텍스트가, 인터넷의 경우에는 마우스나 키보드가 주로 사용되고 있다. 본 논문에서는 기존의 십자말 게임이 갖는 인터페이스 기능에 음성 기술을 추가하여 이를 구현하였다.

<sup>†</sup> 준 회 원 : 성균관대학교 대학원 정보통신공학부

<sup>††</sup> 종 신 회 원 : 성균관대학교 정보통신공학부 교수

논문접수 : 2002년 6월 7일, 심사완료 : 2003년 3월 13일

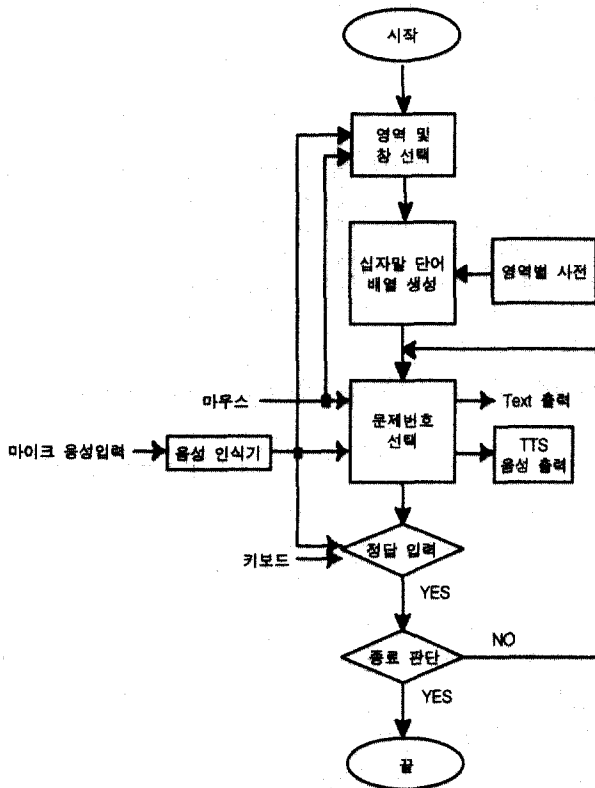
음성에 의한 십자말 게임의 동작은 음성인식[4] 및 합성 시스템[5]에 의하여 구현된다. 음성인식 시스템은 십자말 프로그램의 메뉴 명령어, 문제 선택, 정답 선택 등에 이용되며, 음성합성 시스템은 십자말 배열 내에 선택된 문제에 대한 설명을 음성으로 출력하여 들려준다.

본 논문에서 제안한 십자말 배열 알고리즘의 성능평가는 십자말 배열을 생성하는데 소요되는 연산 시간을 측정하여 이루어졌으며, [3]에서 제안한 배열 알고리즘과 비교하였다. 또한 음성기술이 적용된 십자말 게임에서 중요한 것은 음성에 의해 게임이 얼마나 잘 동작하는가 하는 것이다. 이에 본 논문에서는 음성인식기의 인식 성능 평가를 수행하였다.

## 2. 십자말 게임의 구성

### 2.1 십자말 게임의 구성

(그림 1)은 본 논문에서 구현한 십자말 게임의 구성을 보여준다.

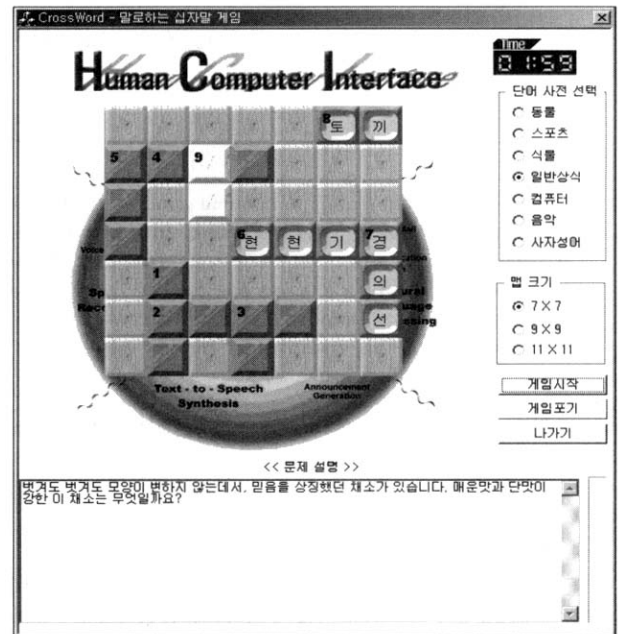


(그림 1) 십자말 게임의 흐름

십자말 게임은 미리 준비된 영역별 사전의 선택과 선택된 영역에 대한 문제 배열로부터 시작된다. 십자말 문제 배열은 선택된 영역별 사전과 창의 크기 정보를 이용하여 생성된다. 영역별 사전과 십자말 배열의 창 크기는 마우스나 음성으로 선택 할 수 있다. 음성에 의한 메뉴 제어를 위해서 십자말 프로그램의 초기화 과정에서는 명령어에 대한

인식 모델이 미리 로딩된다. 마우스 또는 음성 명령에 의하여 영역별 사전 및 십자말 배열의 창 크기 정보가 선택되고, “게임시작” 명령어가 실행되면 십자말 문제 배열이 생성되기 시작한다. 십자말 문제 배열은 랜덤하게 자동적으로 생성된다. 십자말 문제 배열이 완성되면 문제들에 대한 단어 인식 모델이 가변어휘 인식기에 의해 만들어지고 메모리에 로딩된다. 음성인식을 위한 기준 모델은 새로운 십자말 배열이 만들어질 때마다 동적으로 생성 및 로딩 되므로 불필요한 메모리 낭비를 가져오지 않는다. 본 논문에서 구축한 영역별 사전은 “동물”, “스포츠”, “식물”, “일반상식”, “컴퓨터”, “음악”, “사자성어” 로 구성되었으며, 새로운 영역에 대한 사전 구축 및 추가만으로 쉽게 다른 분야에 대한 십자말 게임을 쉽게 즐길 수 있다.

(그림 2)는 십자말 배열이 완성된 십자말 응용 프로그램을 보여준다. 십자말 응용 프로그램은 Window 2000 기반의 Visual C++의 MFC를 이용하여 구현하였다.



(그림 2) 십자말 게임 화면

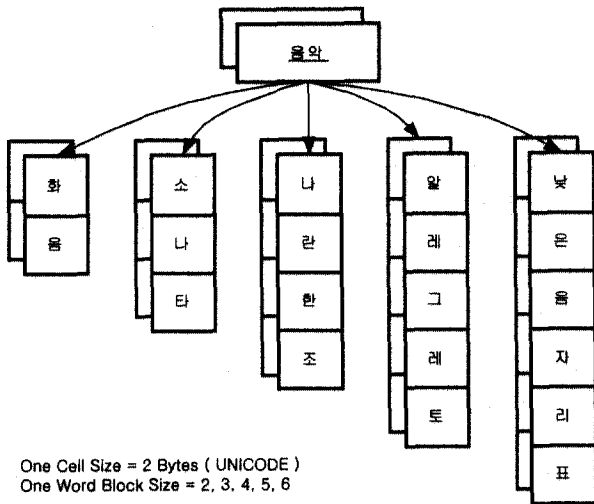
(그림 2)의 십자말 게임 프로그램은 “일반상식” 영역에 대한 “7×7” 십자말 창의 크기를 갖는 십자말 배열로 구성되어 있으며, 사용자는 마우스나 음성으로 번호를 선택하여 만들어진 문제들 중에 하나를 선택 할 수 있다. 임의의 문제가 선택되면 문제에 대한 설명은 텍스트와 음성합성 시스템을 통하여 음성출력으로 주어진다. 선택된 문제에 대한 정답은 키보드나 음성인식을 통해 맞출 수 있으며, (그림 2)에 문자로 표시되어 있는 것을 알 수 있다. 십자말 창에는 현재 “9번” 문제가 선택되어 문제에 대한 설명이 텍스트로 보여지고 있으며, 앞으로 풀어야 할 문제는 문제번호만이 표시되어 있다. “게임포기” 메뉴 명령어는 현재 영역에

대한 게임을 그만두고 다른 영역으로의 게임을 할 때 사용되며, "나가기" 메뉴 명령어는 십자말 게임 프로그램을 종료할 때 사용된다.

2.2 영역선택 및 문제배열

십자말 게임을 위해 사용한 영역별 사전은 "동물", "스포츠", "식물", "일반상식", "컴퓨터", "음악", "고사성어" 로 구성되어 있다. 영역별 사전은 각각 600여 개의 단어들로 구성되어 있으며, 게임을 시작하기 전에 사용자는 마우스나 음성으로 단어 배열에 사용될 영역을 선택 할 수 있다. 십자말 창 크기는 "7×7", "9×9", "11×11"로 주어지며 창이 커질수록 단어의 생성 개수가 늘어난다.

자동으로 십자말 배열을 만들기 위하여 본 논문에서는 CAA(Cross Array Algorithm)를 제안하여 구성하였다. 십자말 배열에 사용되는 단어들은 선택된 창의 크기에 맞도록 랜덤하게 생성되어 배열된다. 영역별 사전의 단어는 2음절 이상 6음절 이하의 단어들로 구성되어 있으며, 영역별로 음절의 개수에 따라 각각 따로 분류해 놓았다. 이것은 트리 구조의 형태를 갖으며, (그림 3)에 "음악" 영역에 대한 분류를 보여주고 있다. 이러한 음절 개수에 의한 단어의 분류 과정은 CAA가 적용되기 전에 모든 영역별 사전에 대해 미리 수행된다.



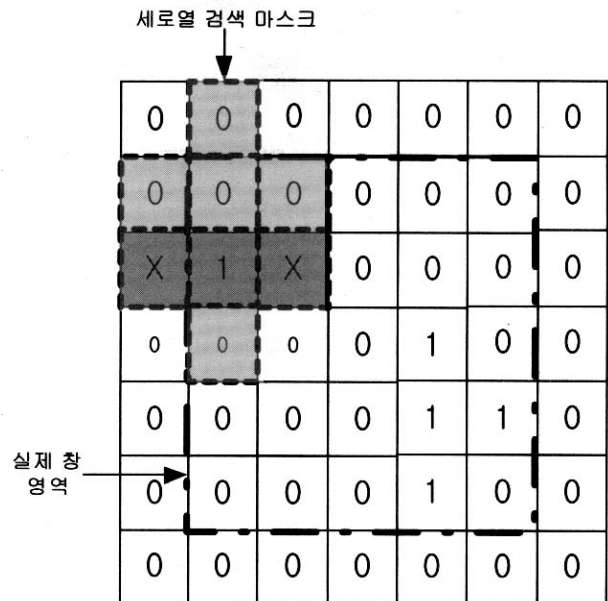
One Cell Size = 2 Bytes ( UNICODE )  
One Word Block Size = 2, 3, 4, 5, 6

(그림 3) 십자말 게임의 단어열 분류

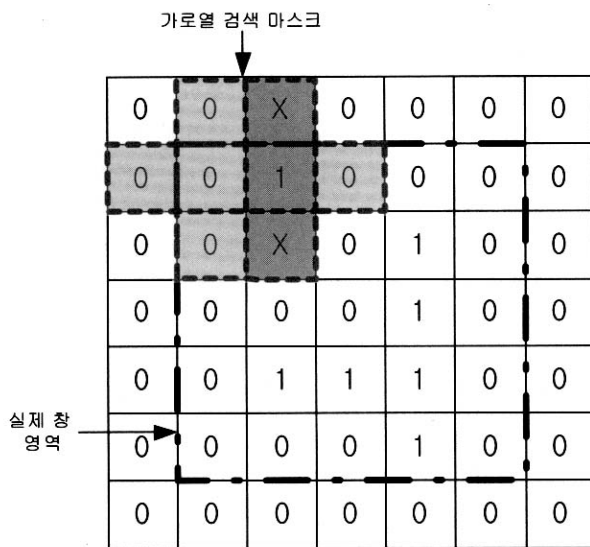
CAA에 의한 십자말 단어 배열은 가로 단어와 세로 단어가 서로 교차하여 이루어진다는 간단한 논리에 의하여 수행되며, 다음은 십자말 배열의 생성 과정을 나타낸다.

- ① 십자말 창 영역의 모든 셀 정보를 "0"으로 초기화한다.
- ② 처음에 놓이게 되는 단어가 가로 또는 세로 열에 위치할 것인지를 결정하고, 놓여지는 단어의 길이를 결정한다. (그림 4)와 (그림 5)에서는 3음절의 단어가 놓여지는 경우를 보여주고 있다.

- ③ 이전에 위치한 단어가 가로 열에 놓일 경우, 다음 단어는 세로열에 놓여야 한다. 세로열에 놓일 단어는 최소 한 음절은 이전의 가로열에 놓인 단어와 동일해야 하며, 인접한 셀과 겹치지 않는 조건을 만족해야 한다. 조건에 맞는 세로열의 단어를 찾기 위하여 CAA는 (그림 4)와 같은 가상의 마스크를 이용한다. CAA는 가상의 마스크를 왼쪽에서 아래쪽의 방향으로 한 셀 단위로 이동하면서 조건에 맞는지를 판단한다. 마스크 내부의 "1"은 이전의 가로열에 놓인 단어와 겹치게 되는 부분이며 "X"는 가로열에 놓여진 단어의 일부분 또는 아직 채워지지 않은 공백의 셀로 볼 수 있다. "0"은 공백의 셀을 의미한다. CAA는 마스크를 이동하면서 "0", "1", "X"의 조건이 모두 일치하는지를 판단하고, 만약 모든 조건이 만족하면 세로열의 단어를 위치시킨다. 여기서 "X"는 조건에 맞는지를 판단 할 때, 전혀 영향을 주지 않는 셀 정보임을 의미한다. 세로열의 단어가 결정되면, 결정된 위치를 정보 "1"로 채워지며, 다음에 수행할 열을 가로열로 한다.
- ④ 세로열에 단어가 놓이면, 다음에 연결할 단어를 가로 방향으로 하고 과정 ③과 같은 방법으로 단어를 검색한다. (그림 5)는 가로열의 단어를 검색하기 위한 마스크를 보여준다.
- ⑤ 연결되는 단어가 없을 때까지 과정 ③과 ④를 반복한다. 더 이상 연결단어가 없으면 새로운 빈 공간을 탐색하여 과정 ③과 ④를 반복한다.
- ⑥ 빈 공간 탐색이 실패할 때까지 과정 ④, ⑤, ⑥을 반복한다. 만약 빈 공간이 더 이상 없으면 십자말 배열 생성 작업을 종료한다.



(그림 4) CAA의 세로열 검색



(그림 5) CAA의 가로열 검색

CAA에 의한 십자말 단어 배열은 매번 게임을 할 때마다 랜덤하게 자동 생성된다. (그림 6)은 “일반상식” 영역에 대한 “7×7” 단어 배열이 생성된 예를 보여주고 있다.



(그림 6) “일반상식” 영역에 대한 “7×7” 단어배열 예

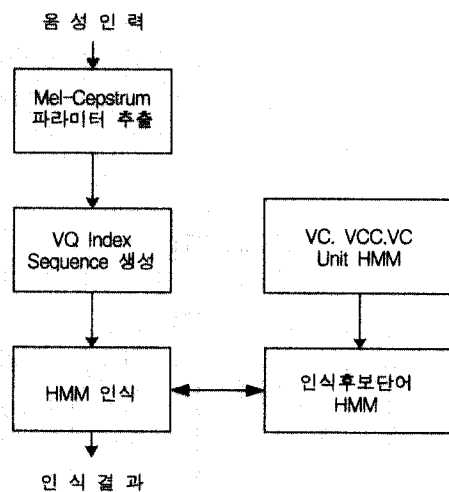
### 3. 음성인식 및 합성

#### 3.1 음성인식 시스템

십자말 단어의 배열이 이루어지고 게임이 시작되면 진행의 모든 부분은 키보드나 마우스뿐만 아니라 음성으로 제어할 수 있게 된다. 음성에 의한 동작은 명령어, 문제선택, 정답선택 등의 제어에 이용된다. 상황에 따라 인식되는

명령어는 다르게 동작한다. 예를 들어 단어 배열이 아직 구성되지 않는 초기에는 메뉴의 명령어만이 인식 후보가 되며, 단어 배열의 구성이 완료되면 상태에 따른 명령어 일부와 문제 번호, 배열에 사용된 단어들을 인식 후보로 갖는다.

(그림 7)은 본 논문의 십자말 게임에 사용된 음성인식 시스템의 구조를 보여준다. 음성인식 시스템의 인식 단위는 비교적 분할작업을 수월하게 할 수 있고, 인식시 분할작업에서 발생하는 약간의 오류를 수용할 수 있는 CV(Consonant Vowel), VCCV, VC를 사용하였다[4]. 음성 데이터 베이스로는 본 연구실에서 구축한 118개의 성이 포함된 성명데이터 1145개, 단음절 521개, PBW데이터가 포함된 1001개, FDB 4088개를 사용하였다. 위의 훈련데이터로부터 CV 단위 308개, VCCV단위 3319개, VC단위 56개를 추출하여 기준 모델을 구성하였다. 인식 단어의 모델을 생성할 때, VCCV 모델로 구성되지 않는 모델은 VC와 CV 모델을 연결하여 구성하도록 음성 시스템을 설계하였다.



(그림 7) 음성인식 시스템

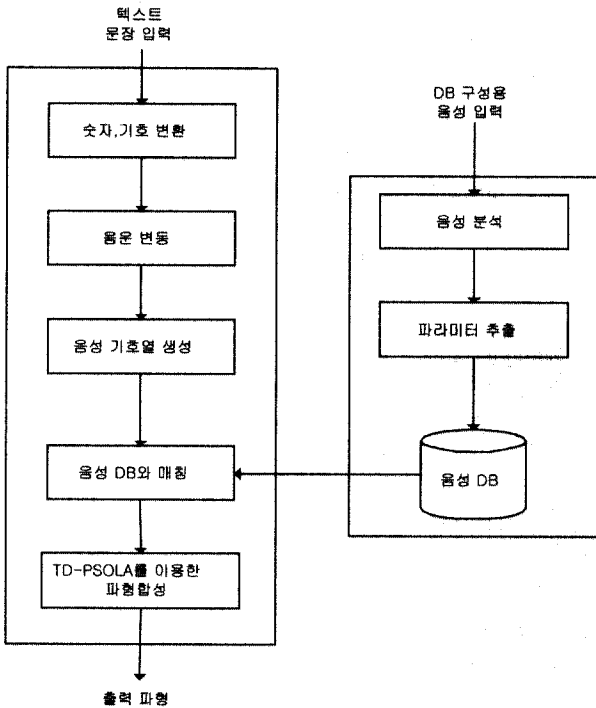
인식단계에서는 입력된 음성으로부터 16차 멜켑스트럼 특징 파라미터를 추출하고, 코드 북을 통과시켜 벡터 코드 열로 이산화 시킨다. HMM 알고리즘은 입력 음성으로부터 얻은 벡터 코드 열과 기준 모델들을 이용하여 인식 결과를 출력한다.

프로그램 초기에는 메뉴 명령어만을 인식 후보로 두고, 게임이 시작되면 일부의 메뉴 명령어와 문제번호, 배열된 단어만이 인식후보로 제한된다. 예를 들어 “게임시작” 명령이 실행되면 메뉴 명령어들 중에 인식후보는 단지 “게임포기”만이 남게 된다. 또한 게임 진행 중에 “게임포기” 명령이 수행되면 인식 후보는 단지 메뉴 명령어만으로 제한된다.

#### 3.2 음성합성 시스템

음성합성 시스템은 텍스트로 주어진 문장을 음성으로 변환하여 출력하는 TTS(Text-to-Speech)를 이용한다. 사용

한 TTS 시스템은 데이터베이스를 구축하는 부분과 음성합성을 수행하는 부분으로 나눌 수 있다. 음성 데이터 베이스는 남녀 화자 각각 한 명씩 빈도수가 높은 어절 1000여 개와 반음절 452개를 구축하였다. 합성 방식은 TD-PSOLA 방법을 적용시켰으며, 합성 단위는 반음절과 어절을 사용하였다. (그림 8)은 본 논문에서 사용된, TTS 음성 합성 시스템을 보여준다.



(그림 8) 음성 합성 시스템

우선 텍스트 문장이 입력되면 숫자나 기호, 영문자 등에 대한 변환과정을 거친 후, 한국어로 소리나는 대로 변환시키는 음운 변동 과정을 거친다. 음운 변동 과정으로부터 음성 기호 열이 생성되면, 사전에 구축되어진 음성 DB로부터 해당하는 소리를 가져와서 TD-PSOLA 방식의 음성 합성을 하게 된다[5].

TTS는 프로그램 수행시에 사용자에게 도움말과 같은 역할을 한다. 예를 들어 명령어가 인식되면 어떤 명령어가 선택되었는지 들려주며, 게임 중에 몇 번 문제를 선택하였는가의 여부, 선택한 문제에 대한 설명, 답을 입력하였을 때 정답의 여부 등을 들려주는 역할을 한다.

#### 4. 실험 및 결과

##### 4.1 십자말 배열 알고리즘(CAA)의 성능평가

본 논문에서는 십자말 배열을 효율적으로 생성하기 위하여 CAA를 고안하여 구현하였다. CAA는 랜덤한 십자말 배열을 영역별 사전을 이용하여 자동으로 생성하는 역할을

한다. 이에 CAA의 성능 평가는 십자말 배열을 생성하는 연산시간의 측정과 전체 창 의 셀의 개수와 십자말 배열내의 단어들이 차지하는 셀 개수의 비를 측정함으로써 이루어졌다.

연산 시간의 측정은 MFC의 "CTIME" 클래스를 사용하여 ms단위로 측정하였다. <표 1>은 본 논문에서 제안한 CAA에 의한 십자말 배열을 생성하는데 소요되는 연산시간을 측정한 결과이다. 연산시간의 측정은 각 창의 크기별로, 각각 다섯 가지의 십자말 배열의 경우에 대하여 수행되었다. 실험 결과는 모든 창에 대하여 십자말 배열을 생성하는데 소요되는 연산시간이 10ms 내외가 소요됨을 알 수 있으며, 이는 실시간으로 생성된다고 볼 수 있다.

<표 1> 제안된 십자말 배열 연산시간

실험	창 크기	7×7	9×9	11×11
경우 1		10 ms	11 ms	11 ms
경우 2		11 ms	10 ms	11 ms
경우 3		10 ms	12 ms	10 ms
경우 4		9 ms	10 ms	12 ms
경우 5		10 ms	10 ms	10 ms
평균 연산시간		10 ms	10.6 ms	10.8 ms

CAA의 두 번째 성능 평가는 십자말 단어 생성율을 측정함으로써 수행되었다. 여기서 단어 생성율은 전체 창이 차지하는 셀의 개수와 생성된 단어들이 차지하는 셀의 개수의 비율로 정의하였다. CAA의 단어 생성율은 각 창에 대하여 십자말 배열을 100번동안 생성하여 측정되었다. <표 2>는 각 창에 대한 단어 생성율을 보여주고 있으며, 대략 50% 내외의 비율을 나타내었다.

<표 2> CAA의 단어 생성율

단어 생성율	창 크기	7×7	9×9	11×11
단어의 셀 개수		2381	3943	6239
전체 셀 개수		4900	8100	12100
백분율(%)		48.5%	48.6%	51.5%

##### 4.2 음성인식 시스템의 성능평가

십자말 게임에 사용된 음성 인식기의 성능은 음성으로 게임을 진행하는데 중요한 역할을 한다. 이에 본 논문에서는 음성인식기의 성능 평가를 수행하였다. 실험에 사용된 입력 음성은 16bits, 11.025kHz로 녹음하였으며, 인식 후보는 각각의 창의 크기에 따라 랜덤하게 생성된 십자말 단어 배열을 사용하였다. 이때 인식 실험에 사용된 십자말 단어 배열은 각 화자마다 다르게 구성하였다. 인식 실험에는 5명의 남성화자가 참여하였으며, <표 3>은 인식 실험 결과를 보여준다. <표 3>의 인식 실험 결과는 각 창의 크기가 "7

×7”, “9×9”, “11×11”일 때 각각 98.5%, 97.6%, 96.2%의 인식률을 보였다.

〈표 3〉 음성인식기의 성능평가

창 크기 화 자	7×7	9×9	11×11
1	7/7	15/16	17/18
2	5/5	8/9	14/15
3	5/5	14/14	16/17
4	7/7	11/11	21/22
5	9/9	14/15	19/19
6	6/6	12/12	18/19
7	7/7	13/14	19/20
8	9/10	11/11	16/16
9	8/8	12/12	18/18
10	5/5	14/15	20/21
평균인식률	98.5(%)	97.6(%)	96.2(%)

〈표 3〉으로부터 창 의 크기가 커질수록 배열된 인식 후보의 개수가 많아지므로 인식률이 조금씩 저하됨을 볼 수 있다. 음성인식 실험 결과 십자말 게임이 음성인식에 의해 효과적으로 동작함을 알 수 있었다.

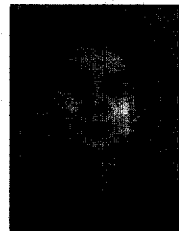
### 5. 결 론

본 논문에서는 음성을 이용한 십자말 게임의 구현에 대해 다루었다. 십자말 게임에 사용되는 십자말 배열을 위해 7개 영역에 대한 영역별 사전을 구축하였으며, 십자말 배열을 랜덤하게 자동 생성하는 CAA를 제안하였다. 이 기능은 새로운 분야에 대한 십자말 게임을 위해서 영역별 사전만을 구축하면 쉽게 적용할 수 있는 장점을 갖는다. CAA의 성능평가 실험에서 십자말 배열이 실시간으로 생성됨을 보여주었다. 또한 본 논문에서는 십자말 게임을 보다 쉽게 즐길 수 있도록 음성 인터페이스를 적용하였다. 음성 인터페이스에 의한 자연스러운 십자말 게임의 진행을 위해 음성인식기의 성능 평가를 하여 효율성을 확인하였다.

십자말 배열의 자동 생성기능과 음성 인터페이스 기술을 적용한 십자말 게임은 일반상식 및 다양한 정보를 게임을 통하여 좀더 쉽게 습득할 수 있으리라 사료되며, 음성을 이용한 게임 및 어학학습, 인터넷 및 모바일 등의 활용분야에 이용될 수 있을 것이라고 생각된다. 또한 십자말을 위한 음성인식의 성능향상을 위하여 단어 확인 기법의 도입도 고려해야 할 것으로 생각된다.

### 참 고 문 헌

- [1] <http://hokisimbox.com>, 호기심 십자말.
- [2] <http://www.netmarble.net>, 넷마블.
- [3] 김동주, 윤재선, 이용주, 김동환, 홍광석, “음성인식 및 합성을 이용한 십자말 게임의 구현”, 한국음향학회 추계학술발표대회논문집, 제20권 제2호, 2001.
- [4] 윤재선, 홍광석, “VCCV 단위를 이용한 어휘독립음성인식 시스템의 구현”, 한국음향학회지, 제21권 제2호, 2002.
- [5] 양원렬, 윤재선, 홍광석, “영한 음차 변환을 이용한 무제한 음성인식 및 합성기의 구현”, 한국 신호처리·시스템학회논문집, 제1권 제1호, 2000.
- [6] L. Rabiner and B. H. Juang, Fundamentals of speech recognition, Prentice Hall, 1993.
- [7] S. Furui, Digital speech processing, synthesis and recognition, Marcel Dekker, 2001.



### 유 일 수

e-mail : gildda@nate.com  
 2002년 강릉대학교 제어계측공학과(학사)  
 2002년~현재 성균관대학교 정보통신공학부 석사과정  
 관심분야 : 음성인식, 신호처리



### 김 동 주

e-mail : radiogyu@korea.com  
 1998년 충북대학교 전자공학과(학사)  
 2000년 충북대학교 전자공학과(공학석사)  
 2001년~현재 성균관대학교 정보통신공학부 박사과정  
 관심분야 : 음성인식, 음성 코딩, 신호처리



### 홍 광 석

e-mail : kshong@skku.ac.kr  
 1985년 성균관대학교 전자공학과(학사)  
 1988년 성균관대학교 전자공학과(공학석사)  
 1992년 성균관대학교 전자공학과(공학박사)  
 1990년~1993년 서울보건전문대학전산정보처리과 전임강사  
 1993년~1995년 제주대학교 정보공학과 전임강사  
 1995년~현재 성균관대학교 정보통신공학부 부교수  
 관심분야 : 음성인식 및 합성, HCI