

문서 편집 접근성 향상을 위한 음성 명령 기반 모바일 어플리케이션 개발

박주현[†], 박세아^{**}, 이무늬^{***}, 임순범^{****}

Voice Activity Detection Algorithm using Wavelet Band Entropy Ensemble Analysis in Car Noisy Environments

Joo Hyun Park[†], Seah Park^{**}, Muneui Lee^{***}, Soon-Bum Lim^{****}

ABSTRACT

Voice Command systems are important means of ensuring accessibility to digital devices for use in situations where both hands are not free or for people with disabilities. Interests in services using speech recognition technology have been increasing. In this study, we developed a mobile writing application using voice recognition and voice command technology which helps people create and edit documents easily. This application is characterized by the minimization of the touch on the screen and the writing of memo by voice. We have systematically designed a mode to distinguish voice writing and voice command so that the writing and execution system can be used simultaneously in one voice interface. It provides a shortcut function that can control the cursor by voice, which makes document editing as convenient as possible. This allows people to conveniently access writing applications by voice under both physical and environmental constraints.

Key words: Voice Recognition, Voice Command System, Voice Writing Application, Accessibility, Dual Voice Mode, Physical Disabled User, Mobile Accessibility

1. 서 론

사물인터넷 기술이 점차적으로 우리 생활에 익숙해지면서, 거동이 불편한 장애 사용자들에게도 사물인터넷 기술은 새로운 생활 수단이자 희망으로 발전해 가고 있다. 이러한 사물 인터넷을 이루는 다양한 기술 중 음성 인식 및 음성 명령 기술은 요리할 때, 운전할 때 등과 같이 양손 사용이 자유롭지 않은 상

황이나, 장애로 인해 손사용이 자유롭지 못하는 사람들에게 디지털 기기 접근성을 확보해 주는 중요한 수단이다[1, 2]. 음성 인식 시스템이란 기계로부터 음성에서 의미가 분석되어 동작이 수행되는 시스템을 의미하며, 음성 명령 시스템은 이러한 음성 인식 기술을 바탕으로 음성 명령으로 시스템이 제공하는 다양한 기능들을 실행 시키는 기술이다. 모바일 환경에서도 음성 인식 및 명령 기술은 S보이스, Siri 등의

* Corresponding Author: Soon-Bum Lim, Address: (140-742) Sookmyung Women's Univ. Cheongpa-dong 2-ga, Yongsan-gu, Seoul, Korea, TEL: +82-2-710-9424, FAX: +82-2-710-9703, E-mail: sblim@sm.ac.kr
Receipt date: May 9, 2018, Revision date: Sep. 20, 2018
Approval date: Oct. 19, 2018

[†] Dept of IT Engineering, Sookmyung Women's University (E-mail: park.joohyun5@gmail.com)

^{**} Dept of IT Engineering, Sookmyung Women's University (E-mail: joli9706@naver.com)

^{***} Dept of IT Engineering, Sookmyung Women's University (E-mail: ansmls2@sm.ac.kr)

^{****} Research Institute of ICT Convergence, Dept of IT Engineering, Sookmyung Women's University

* This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2015R1D1A1A09060170, NRF-2018R1A4A1025559)

다양한 형태로 우리 생활 속에서 쉽게 접근 할 수 있게 되었고 이에 발맞춰 음성으로 문자를 보내거나 사진 촬영을 하는 등 음성 인식 기능이 적용된 다양한 어플리케이션들 또한 쏟아지고 있다[3, 4].

하지만 이러한 음성 인식 기능이 탑재된 어플리케이션 대부분은 음성 명령 기능만을 수행하거나 STT (Speech-To-Text) 기능만을 수행하는 등 한가지의 역할만을 수행하는 어플리케이션으로 완전한 음성 인터페이스를 사용하기 어려운 것이 현실이다. 따라서 이 두 가지의 역할이 동시에 수행되어야 하는 다양한 어플리케이션 특히, 일상생활의 기본 활동이 되는 기록 활동을 위한 음성 편집 메모장과 같은 어플리케이션은 찾아보기 어렵다. Google Docs[5]를 비롯해 다수의 음성 인식기능을 탑재한 모바일 기록 어플리케이션은 단순히 받아 적고 읽어 주는 기능만을 제공할 뿐 음성 명령을 통해 편집과 같은 기능을 지원하지 않고 있어 음성 인식 기능을 제대로 활용하고 있지 못하고 있다. 안드로이드 마켓에서 가장 많은 다운로드를 기록한 Well source의 Speech Note[6]는 음성 입력으로 작성된 메모는 음성 입력 즉시 텍스트로 변환 되어 저장되는 기능과 ‘new paragraph’, ‘period’와 같이 지정 된 특정 단어가 입력 될 경우 줄 바꿈과 마침표를 입력해주는 간단한 음성 명령 기능이 혼합되어 제공한다. 하지만 음성 인식의 수행 체계가 구분되지 않아 텍스트로 기록하고 싶은 단어를 음성 명령어로 인식하여 해당 명령어에 할당 된 기능을 수행하는 등 음성 명령 기능에 한계가 있다. 또한 실제 사용자들이 작성이 완료된 텍스트를 음성 명령을 통해 자유롭게 편집하는 기능

과 음성 명령을 통해 어플리케이션 메뉴를 제어하는 기능은 찾아볼 수 없다.

따라서, 본 논문에서는 음성 인식 기능의 수행 체계를 음성 텍스트 입력 모드와 음성 명령 모드로 구분 설계하여 신체 및 환경적 제약 상황에서 사용자의 음성을 통해 기록 및 편집하고 어플리케이션을 제어할 수 있는 음성 명령 기록 어플리케이션을 개발한다. 또한 각 모드 간의 전환을 음성 명령으로 자유롭게 전환할 수 있도록 하였으며, 문서의 편집을 위한 커서의 역할을 슛컷 기능으로 제공하여 기록된 문서에서 자유롭게 커서를 이동할 수 있도록 하여 모바일 환경에서의 기록 활동의 접근성을 향상 시키는 것을 목표로 한다. 본 어플리케이션은 음성 인식 기능을 통해 받아들여진 텍스트를 검사하여 해당하는 모드를 판별하고, 모드 별 제공 기능을 수행하는 절차로 동작한다.

본 논문의 구성은 다음과 같다. 2장에서 본 논문의 기본 기술이 되는 음성인식 기술에 대해 설명하고, 이를 이용한 응용프로그램에 대하여 설명한 다음, 3장에서 제안하는 음성 인식 편집 시스템에 대해 설명한다. 여기에서는 제안한 시스템의 전체적인 개요와 음성으로 메모 선택 및 종료, 기록, 메모 편집, 슛컷 기능에 대해 순서로 서술한다. 그리고 4장에서는 제안한 시스템의 구현 과정과 결과를 제시하고 5장에서 시스템에 대한 사용성 평가 후 6장에서 결론을 맺는다.

2. 관련 연구

2.1 음성 인식 기술



Fig. 1. Screenshot of (a) Speechnotes (b) Google Docs.

음성 인식은 사람이 일상생활 속에서 마우스나 키보드 등을 사용하지 않고 목소리를 통해 원하는 기기 및 정보 서비스를 제어할 수 있는 기술이다. 이러한 음성 인식 기술은 2000년대 중반까지 낮은 음성 인식률로 대중화되지 못했지만, 최근 휴대용 컴퓨팅 단말의 확산과 클라우드 인프라의 확충에 맞물려 최근 IT업계의 'Next Big Thing'으로 부상하고 있다[7].

음성 인식은 친숙하고 편리하며, 이동 및 작업 중에도 상시 입력이 가능하고, 화자의 고유 정보를 전달할 수 있으며, 입력속도가 빠르다는 장점 때문에 다양한 IT기기 및 서비스에 결합되고 있다[8]. 운전 중 모바일 기기 조작 시 안전성과 효율성 향상부터 화자의 음성을 통한 신원, 심리, 건강상태, 언어 능력 등을 파악하는 개인별 서비스 제공과 시각장애 및 운동장애 사용자의 디지털 기기 접근을 위한 서비스에 이르기 까지 다양하게 활용되고 있다[9].

2.2 음성 인식 활용 어플리케이션

구글 Docs는 구글 문서 환경에서의 문서 편집기로 음성으로 문서 작성을 지원한다. 구글의 음성 인식 기능을 활용하기에 다른 어플리케이션에 비해 음성 인식률이 좋으며, 음성 편집 기능을 지원하여 문자 입력 도중 편집에 관한 음성 명령이 발생하면 구글 문서 시스템이 이를 판단하여 복사, 붙이기, 삭제 등의 명령어를 수행한다. 그러나 현재 음성 편집은 영어만 지원하며, 문서 작성도중에 수정이 필요한 경우 마우스를 이용해 해당 부분으로 이동 후 키보드로 수정해야 한다. 또한, 복잡한 편집명령어로 아직 실용적인 수준은 아니며 웹 기반으로 사용 시 환경적 제약이 있다.

Speechnotes는 안드로이드 버전과 웹 버전으로 제공하는 음성 인식 기록 어플리케이션으로, 음성으로 메모 작성이 가능하고 작성한 메모를 음성으로 확인 할 수 있다. 음성 입력 즉시 텍스트로 변환 되어 화면에 입력되며, 지정 된 특정 단어가 입력 될 경우 문자 기록 활동에서 자주 사용 되는 특수문자(마침표, 쉼표, 물음표, 콜론, 세미콜론, 느낌표, 하이픈)와 줄 바꿈(새 문장, 새 문단)의 기능을 수행한다. 이는 음성 텍스트 입력과 음성 명령이 결합 된 형태의 어플리케이션으로 본 연구의 목표와 유사하지만, 모드 구분 없이 음성 텍스트 입력 모드에서 특정 단어에 역할을 지정하기에는 명령어 수의 한계가 있다. 어플리케이션의 실제 명령어 중 'period'의 경우 사용자가 텍스트로 period라고 입력 하고 싶지만 입력을 받는 즉시 특수문자인 마침표가 화면에 표시되는 불편함이 있다. 또한 입력이 완료된 텍스트를 수정하기 위해 커서를 위/아래 등으로 움직일 수 있는 음성 명령 기능은 제공되지 않아 입력이 완료된 문서를 음성 명령을 이용하여 수정하기 어렵다. 또한, 기본 언어가 영어로 설정 되어 있어 한국어로 설정을 변경 후 사용할 경우, 한국어로 문자 기록은 가능하지만, 'Period'와 같은 특수 문자 명령어를 입력하면 한국어 '퍼리어드'로 인식되어 기능이 수행되지 않는다. 또한 음성으로 메뉴 제어가 불가능하고 저장, 수정, 삭제 등의 메모편집이 불가능 하다는 단점이 있다.

리케이션의 실제 명령어 중 'period'의 경우 사용자가 텍스트로 period라고 입력 하고 싶지만 입력을 받는 즉시 특수문자인 마침표가 화면에 표시되는 불편함이 있다. 또한 입력이 완료된 텍스트를 수정하기 위해 커서를 위/아래 등으로 움직일 수 있는 음성 명령 기능은 제공되지 않아 입력이 완료된 문서를 음성 명령을 이용하여 수정하기 어렵다. 또한, 기본 언어가 영어로 설정 되어 있어 한국어로 설정을 변경 후 사용할 경우, 한국어로 문자 기록은 가능하지만, 'Period'와 같은 특수 문자 명령어를 입력하면 한국어 '퍼리어드'로 인식되어 기능이 수행되지 않는다. 또한 음성으로 메뉴 제어가 불가능하고 저장, 수정, 삭제 등의 메모편집이 불가능 하다는 단점이 있다.

3. 음성인식 편집 시스템 설계

3.1 시스템 개요

본 시스템은 요리, 운동, 운전 중과 같은 환경 제약적인 상황과 장애로 인해 팔을 사용이 자유롭지 못한 상황의 사용자가 화면 터치를 최소화 하여 음성 기록 어플리케이션을 사용하여 입력뿐만 아니라 메뉴와 커서 제어를 통해 문서 편집까지 가능하게 하는 것이 목표이다.

본 논문에서 제안하는 시스템은 입력 받은 음성 명령어를 분석하여 명령어에 부여된 특정 기능을 수행하도록 설계하였으며, 시스템의 이름을 Talkie Takie라고 정하였다. 메모를 작성하는 화면에서 메모 작성을 위한 음성 인식과 명령어 수행을 위한 음성 인식이 필수적으로 구분되어야 하므로, 음성 인식 모드를 '라이팅 모드'와 '커맨드 모드'로 구분하여 구성하였다. 결과적으로 모드 체인지를 통해 STT와 명령어 수행을 구분하였다. Fig. 2는 Talkie Takie의 상태전이도이다. 어플리케이션 메인 화면이 실행되면 새로운 파일을 생성하거나 기존의 파일을 선택할 수 있는 음성 명령 모드가 실행된다. 음성 명령으로 선택 상태로 진입하면 읽기, 삭제, 수정 명령을 수행하고 각 선택, 메인, 쓰기 상태로 진입한다. 음성 텍스트 입력 모드인 쓰기 상태로 진입하면 음성 텍스트 기록을 수행한다. "커맨드 모드", "라이팅 모드" 명령어 호출로 쓰기 상태와 파일 제어 음성 명령 모드인 커맨드 상태 간 상호 전환이 이루어지며, 각 상태에서 슛컷 명령어를 호출하여 수행한다. 커맨드 상태에서

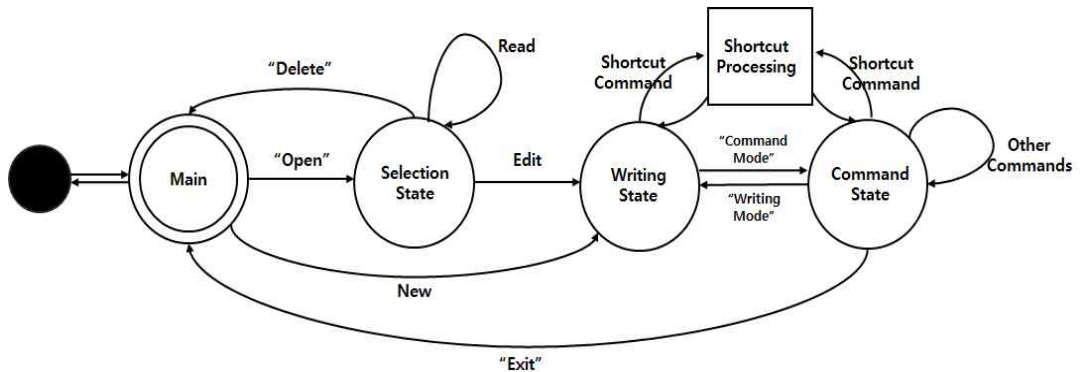


Fig. 2. Transition Diagram of the System.

서 파일 수정 및 작성을 종료하면 어플리케이션의 메인 화면으로 돌아가도록 설계하였다.

3.2 시스템 주요 기능

본 시스템의 주요 기능은 크게 ‘파일 선택’, ‘음성 기록’, ‘메모 편집’, ‘숏컷 기능’으로 구성된다. 먼저, ‘파일 선택’ 기능에서는 어플리케이션 실행 후 저장된 메모 목록 화면에서 메모 번호를 포함한 명령어를 통해 저장된 메모를 불러올 수 있다. 화면에 메모 목록은 최대 10개 나열되고, 그 이상 메모가 저장된 경우 목록 페이지가 추가되며, 페이지 이동 명령어를 통해 자유로운 페이지 이동이 가능하다. 또한 음성 명령어로 기존 메모를 불러온 후 수정 및 삭제가 가능하며, 새 메모 생성과 어플리케이션 종료 기능을 수행할 수 있다. 메모 선택 화면에서 사용 가능 명령어 리스트 확인도 가능하다.

두 번째, ‘음성 기록’ 기능은 본 연구에서 설계한 음성 인식 모드의 ‘라이팅 모드’를 의미하며, 새 메모 생성 후 음성 기록의 초기 상태는 라이팅 모드로 설정된다. 음성 입력 기능으로 받아들인 음성은 모두 메모로 입력되며, 메모의 편집 기능을 수행하기 위해서는 ‘커맨드 모드’로 모드 전환 후 수행 가능하다. 메모 작성 초기 화면에서 커서는 항상 내용 텍스트 맨 뒤에 위치하며, 커서 관련 명령어인 숏컷 명령어를 통해 커서를 자유롭게 이동 시키며 실제 문서 작성 화면에서와 같이 메모 작성이 가능하다.

세 번째 기능인 ‘메모 편집’ 기능은 ‘커맨드 모드’로 메모에 대한 특정 편집 기능을 수행한다. 작성한 메모를 음성으로 확인하고 저장할 수 있으며, 메모 작성 시 내용에 포커스 되어있는 커서를 이동시킬 수

있다. 라이팅 모드와 같이 숏컷 명령어를 사용 할 수 있으며, ‘라이팅 모드’ 명령어 호출로 라이팅 모드로 상태를 전환한다.

마지막으로 ‘숏컷 기능’은 모드에 상관없이 항상 수행 가능하여 문서작성 및 편집 중 모드를 매번 체인지 하지 않아도 된다. 다양한 숏컷 명령어를 통해 커서를 자유자재로 이동하여 원하는 위치에서 글을 입력 또는 수정 가능하며, 문장 삭제 기능도 수행한다. 다음 Table 1은 본 시스템에서 사용되는 명령어와 명령어의 기능을 나타낸 표이다.

4. 음성 인식 시스템 구현 및 결과

4.1 구현 과정

본 시스템은 안드로이드 환경에서 구현하였다. 먼저 마이크를 활성화하기 위한 SpeechRecognizer 클래스를 생성하고, 입력 받은 음성을 텍스트로 변환하여 ‘mResult’ 변수에 저장한다[10]. 새 메모를 작성하거나, 수정하는 화면에서는 현재 음성 인식으로 받아들여진 텍스트가 메모 콘텐츠를 위한 텍스트인지, 명령어인지를 구분할 필요가 있다. 따라서 ‘isTalkied’ 함수를 생성 후 현재 상태가 ‘라이팅 모드’인지, ‘커맨드 모드’ 인지 판별 후, 해당 모드에 따라 입력되는 명령어 체계를 구분하였다. 사용자가 현재의 모드를 확인할 수 있도록 화면의 상단에 색깔과 텍스트로 시각화 하였다.

모드 체크 후 ‘라이팅 모드’에서는 현재 커서가 포커스 되어 있는 EditText에 ‘mResult’의 값이 입력되도록 설정하였고, ‘커맨드 모드’에서는 Switch문을 이용하여 입력 받은 ‘mResult’의 값이 정해진 명령어

Table 1. Functions and corresponding commands Provided by the System

Main Functions	Commands	Detailed Features
File Selecting	“1번 메모” ... “10번 메모”	Open Memo
	“메모고치기	Edit Memo
	“메모삭제”	Delete Memo
	“다음” “이전”	Next Page, Previous Page
	”메모”	New Memo
	”종료”	Exit
Voice Recording (Writing Mode)	“커맨드 모드”	Switch to Command Mode
Edit (Command Mode)	“읽기”	Read the Memo Contents
	“저장”	Save the Memo Contents
	“제목”, “내용”	Move Cursor Focus to Title and Contents
	“라이팅 모드”	Switch to Writing Mode
	”종료”	Exit
Shortcut	“커서 앞으로”, “커서 뒤로”, “커서 위로”, “커서 아래로”, “커서 스페이드”, “커서 엔터”	Cursor Move

와 매치 판단 후 해당 명령어의 기능을 수행할 수 있도록 하였다.

커서의 움직임은 숏컷 기능으로 설정하여 모드와 상관없이 별도의 기능으로 설정하였다. 이는 모든 모드에서 사용 되는 기능이기 에 모드 변환의 번거로움을 줄이기 위함이다. 따라서 현재의 모드를 체크하기 이전에 먼저 입력 값이 숏컷 명령어와 매치하는지를 체크한다. 숏컷 명령어는 커맨드 모드 명령어와 다르게 짧은 간격으로 자주 발생하기 때문에 마이크가 활성화 된 상태에서 명령어 텍스트를 지속적으로 입력 받아 숏컷 기능을 처리한다. 이때 명령어와 명령어 사이에 공백 시간을 기준으로 앞, 뒤 입력된 명령어를 각각 하나의 값으로 간주하여 해당 기능을 실행시킨다.

‘라이팅 모드’와 ‘커맨드 모드’를 구분할 필요가 없는 어플리케이션의 메인 화면에서는 모드 판별 없이 해당 명령어와의 매치 여부만을 확인한다. 매치되는 명령어가 없을 시에는 사용자에게 수행 가능 명령이 아님을 알려준다. 메인 화면에는 저장 된 메모의 목록이 보이도록 구성하였으며, arrayAdater 객체를 이용하여 한 페이지에 10개씩 묶음으로 메모를 보여준다. 마지막으로 마이크를 비활성화 시키고 일반 메모장으로 활용하고 싶을 때에는 버튼을 이용하여 자유롭게 마이크의 on/off를 제어할 수 있다. 사용자가

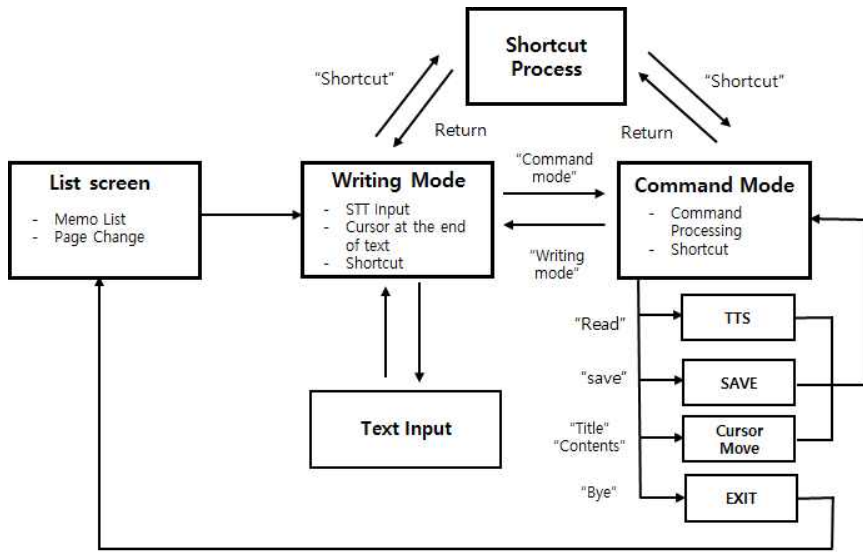
즉각적으로 마이크의 on/off상태를 확인할 수 있도록 마이크 버튼을 빨강과 회색으로 활성/비활성 상태를 시각화 하였다.

4.2 시스템 구현 결과

음성 명령어 처리는 목록 화면, 라이팅 모드, 커맨드 모드 에서 각각 이루어진다. 라이팅 모드와 커맨드 모드에서는 숏컷 처리 후 다시 결과를 리턴 해주며, 커맨드 모드에서도 마찬가지로 특정한 편집 기능을 수행하는 명령어 처리 후, 다시 현재 모드로 되돌아온다. Fig. 3은 전체 시스템의 흐름을 보여준다. 명령어는 모두 한국어로 수행되면, 라이팅 모드에서 커맨드 모드로의 모드 전환 명령어와 커서 사용을 위한 커서 명령어는 연속하여 두 번씩 발음하게 설정하였다. 이는 입력 받은 명령어가 기록을 위한 음성 텍스트로 입력되는 것을 방지하기 위함이다.

4.2.1 어플리케이션 목록 화면

목록 화면은 어플리케이션 시작 또는 저장 후 기본화면으로 ‘x번 메모’ 로 해당 번호의 메모를 선택하여 메모 열기를 수행한다. 한 화면에는 최대 10개의 메모가 나열되며, 10개 이상의 메모는 페이지를 넘겨 해당 메모를 선택할 수 있다. ‘넘겨’와 ‘이전’ 명령어



* The words in all double quotes are in Korean.
 * In Writing mode, the word "command mode" and "cursor" for text input speak twice.

Fig. 3. System Flow Diagram.

로 해당 페이지를 제어할 수 있다.

4.2.2 메모 선택 후 화면(memo read)

첫 화면에서 메모를 선택 후에 화면에서는 메모를 수정하는 ‘메모 수정’과 해당 메모를 읽어주는 ‘읽기’ 명령어가 존재한다. 메모 선택 후 화면은 기본적으로 ‘라이팅 모드’가 설정되며, 음성 인식으로 입력되어진 텍스트가 작성된다. 모드 변환을 원할 때는 ‘커맨드 모드’와 ‘라이팅 모드’ 명령어를 입력하면 된다. 커맨드 모드에서는 ‘저장’, ‘제목’(제목으로 커서의 포

커스 이동)/‘내용’(내용으로 커서의 포커스 이동), ‘읽기’, ‘종료’ 등의 명령어로 다양한 편집을 수행할 수 있다.

숏컷 명령어로는 ‘커서 위로’, ‘커서 아래로’, ‘커서 뒤로’, ‘커서 앞으로’, ‘커서 스페이드’, ‘커서 엔터’ 등의 커서 제어 명령어로 커서를 원하는 위치로 배치시킬 수 있다.

5. 사용성 평가 및 결과

본 연구의 사용성 평가 목적은 개발된 Talkie Takie가 제공하는 음성 모드의 기능들과 모드 변환이 정상적으로 작동하는지 검증하고, 문서 수정 편집에서 숏컷 기능으로 제공하는 커서의 효율성을 검증하기 위함이다. 사용성 평가를 위한 태스크를 수행할 피실험자는 총 10명의 대학생으로 구성하였다. 이들 모두 문서 작성 프로그램을 사용해 본 경험이 있는 스마트폰 사용자들로 구성되었으며, 실험 전 손을 사용하지 않고 실험에 참가하도록 안내하였다.

5.1 음성 명령 기능 검증

시스템의 음성 명령 기능 검증은 명령어 카테고리 별 전체 실험 참가자의 음성 명령 기능의 그룹 별 수행 성공률로 평가하였다. 검증에 사용되는 태스크

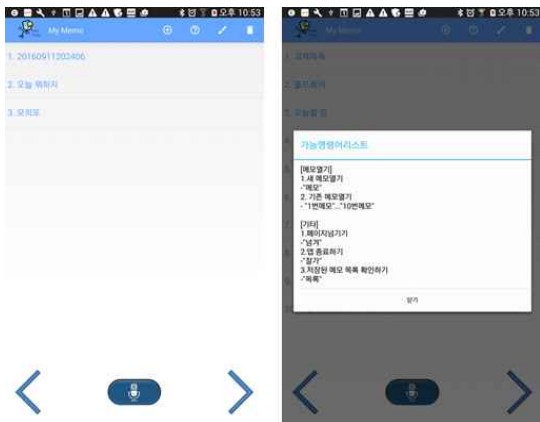


Fig. 4. First page of Application and Command List.

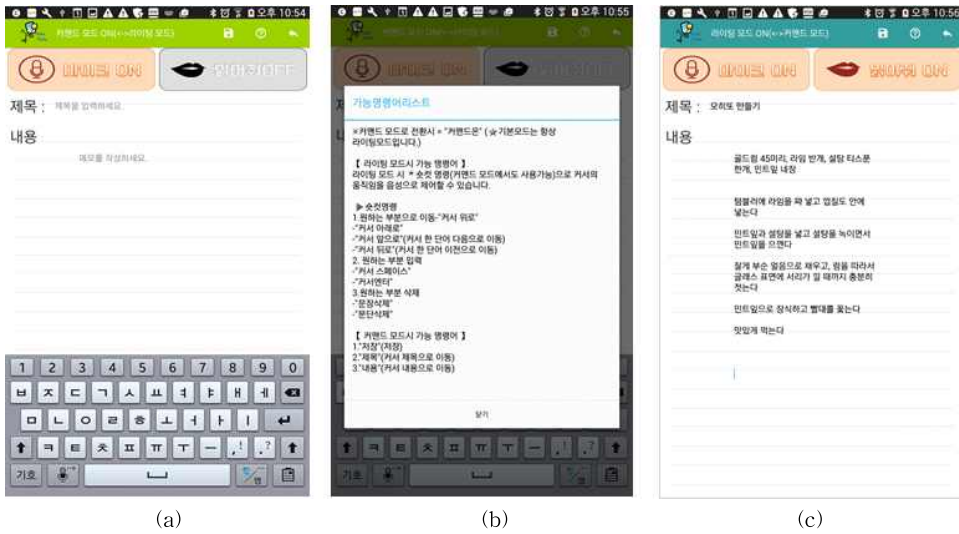


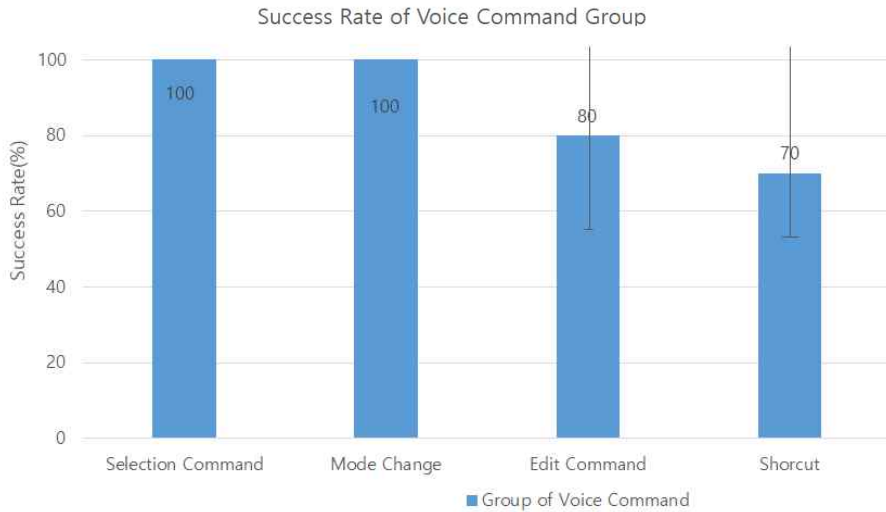
Fig. 5. Screenshot of Application (a)Memo Writing (b) Command List (c) Shortcut Screen,

는 일반적으로 문서를 작성 및 수정 할 때 사용하는 행위를 구분하여 설정되었으며, 태스크 별 본 시스템에서 제공하는 음성 명령 그룹의 기능이 적절하게 포함되어 있다. 평가에서 각 사용자들은 어플리케이션으로 메모를 작성하게 하였으며, 작성된 메모를 선택, 읽기, 수정, 삭제 하는 태스크를 수행하였다. 평가가 진행 되는 동안 성공률 측정은 음성 명령어를 호출하고 해당 명령어에 할당된 기능이 실행되기까지의 과정을 측정하였다. 이 측정에서의 성공은 첫 번째 음성 명령어 호출 시도로 해당 기능이 정상적으로

로 수행 될 경우로 정의하였으며, 음성 명령어를 1회 초과하는 시도부터는 태스크 실패로 간주하였다. 평가에 참여하는 모든 사용자들은 동일한 장소에서 실험을 수행하였으며, 태스크 수행 시 사용자의 모습, 디바이스의 상태와 주변 환경 등이 평가자에 의해 관찰되었다. 평가를 위한 태스크의 구성은 다음 Table 2와 같으며, 각 태스크에 수행되어야 하는 음성 명령어 및 호출 횟수를 표시하였다. 제시한 호출 횟수는 태스크가 정상적으로 수행 되었을 때의 명령어 호출의 횟수이며, 태스크 성공 횟수의 기준이 된다.

Table 2. Configuration of Tasks for Evaluation

Command Category	Task 1 (Create New Document and Writing)	Task 2 (Open and Read Written Document)	Task 3 (Open and Delete Written Document)	Task 4 (Edit Written Document)	Total Number of Commands
Selection Command	- New Memo 1 time - Exit 1 time	- Next page 1 time - Open Memo 1 time - Exit 1 time	- Open Memo 1 time - Delete Memo 1 time - Exit 1 time	- Open Memo 1 time - Edit Memo 1 time - Exit 1 time	11 times
Mode Change	- Command Mode 1 time - Writing Mode 1 time	- Command Mode 1 time	- Command Mode 1 time	- Command Mode 3 times - Writing Mode 2 times	9 times
Edit Command	- Title 1 time - Read 1 time - Save 1 time - Exit 1 time	- Content 1 time - Read 1 time	- Read 1 time - Exit 1 time	- Save 1 time - Title 1 time - Content 1 time - Exit 1 time	12 times
Shortcut	- Cursor Move 6 times	-	-	- Cursor Move 12 times	18 times



	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	Average
Selection	1	1	1	1	1	1	1	1	1	1	100%
mode change	1	1	1	1	1	1	1	1	1	1	100%
edit command	1	1	0	1	1	1	0	1	1	1	80%
shortcut	0	1	1	1	0	0	1	1	1	1	70%

Fig. 6. Success Rate of Voice Command Group, 95% Confidence Interval.

음성 명령 기능 그룹 별 수행 성공률 평가 결과 Fig. 6과 같은 결과를 얻었다. 선택 명령어와 모드 체인지의 결과 모든 사용자가 태스크 별 예상된 명령어 실행 횟수대로 수행하여 성공률 100%를 얻었지만, 모드 체인지와 슛컷 명령어에서는 예상된 횟수보다 초과되게 수행하여 각 80%, 70%의 결과를 얻었다. 이 수치는 2가지 측면으로 원인을 파악할 수 있었다. 첫 번째 원인은 편집 명령어와 슛컷 명령어의 경우, 사용자의 자율도가 높은 기능이기때 사용자의 습관(짧은 저장 빈도, 습관적 커서의 이동)에 따라 명령어의 횟수가 증가되었음이 관찰되었다. 두 번째 초과 시도의 원인은 명령어 입력 당시 주변의 소음이 발생으로 인한 오류와 명령어 입력 당시 네트워크 연결이 해제되어 명령어를 초과 입력하였다. 이는 제안하는 시스템의 기능 문제가 아닌 사용자의 문서 작성 습관과 사용 환경에서의 문제로 본 시스템의 문제가 아님으로 판단되며, 본 시스템에서 제공하는 다양한 명령어들이 무리 없이 제 기능을 수행함을 확인하였다.

5.2 커서 기능의 효율성 검증

커서 기능의 효율성 검증은 두 가지 실험을 통해 확인하였다. 첫 번째 실험에서는 문서 작성 및 편집

시 사용되는 편집 기능을 태스크로 제시하여 각 어플리케이션 별 편집 가능 여부를 통해 문서 편집의 가능성을 확인하였다. 두 번째 실험에서는 작업 지시서에 따라 문서 편집에 걸리는 시간과 편집의 완성도를 성공률로 측정하여 효율성을 확인하였다.

첫 번째 실험에서는 간단한 문서 편집 기능을 태스크로 제시하였으며, 각 어플리케이션 별 수행이 가능한지 커서 조작의 가능성을 확인하였다.

Table 3은 제시한 태스크에 대한 어플리케이션 별 수행 가능 여부를 나타낸 표이다. 표에서 알 수 있듯이 본 시스템인 Talkie Takie가 음성 명령어로 문서의 편집이 가능함을 알 수 있다. 이는 본 시스템이 커서의 움직임을 기능으로 제공하고 있어, 기존 Speech-

Table 3. A table showing whether the task can be performed in each application

Functions	Talkie Takie	Speechnotes
• Indent	Y	N
• Outdent	Y	N
• Alignment	Y	N
• line-break	Y	Y
• Word Spacing	Y	N

notes를 비롯한 음성 입력 문서 어플리케이션에서 불가능했던 간단한 문서 편집이 가능함을 의미한다.

두 번째 실험에서는 작업 지시서에 따른 문서 편집 태스크를 어플리케이션 별 수행에 걸리는 평균 시간과 태스크 성공률 측정으로 커서 조작의 효율성을 확인하였다. 태스크 수행에 걸리는 시간은 총 10명의 피실험자가 각 어플리케이션 별 3회의 편집을 수행 후 평균 시간으로 설정하였으며, 태스크 성공률은 편집의 완성도 즉, 작업 지시서에 표시된 총 25개의 교정 부호와 3개의 숫자 삽입에 따라 사용자가 성공 수행한 교정 부호 개수와 숫자 삽입 개수로 측정하였다. 또한, 각각의 커서 명령어가 정확한 기능으로 호출되었는지도 확인하였다. 피실험자에게 편집이 되지 않고 텍스트로만 구성된 문서와 편집점을 포함하고 있는 완성된 문서를 각각 제공하고, 슛컷 명령어를 사용하여 교정 부호 편집을 수행하도록 하였다. 제공하는 모든 슛컷 명령어가 고루게 사용 되도록 28개의 편집점을 설정하였다. 28개의 편집점을

수정하기 위해서는 총 176회의 음성 커서 명령어(스컷)가 발생된다. 커서 명령어에는 “커서 스페이드”가 18회, “커서 뒤로” 140회, “커서 엔터” 10회, “커서 위로” 3회, “커서 앞으로” 3회, “커서 아래로” 2회가 포함된다. 먼저 교정 부호가 편집이 끝나면, 마지막 3줄의 각 텍스트 시작지점에 숫자를 삽입하도록 안내하였다. Fig. 7은 피실험자에게 제공한 편집 문서 화면이다.

커서 기능의 효율성 평가 결과 Talkie Takie가 28개 작업을 모두 완성하는데 평균 5분 46초의 수행 시간에 100%의 태스크 성공률을 보였다. 또한 피실험자 모두 커서 명령어를 자유롭게 사용하는 것을 확인할 수 있었다. 반면, Speechnotes는 평균 7분 46초의 수행 시간에 34%의 태스크 성공률을 보여 Talkie Takie의 효율성이 좋은 것으로 확인하였다 (Fig. 8). 이 결과 또한 음성 명령으로 커서의 기능을 제공하고 있어 문서 편집이 가능하며 편리함을 의미한다. Speechnotes의 경우 평균 34%의 교정 부호만

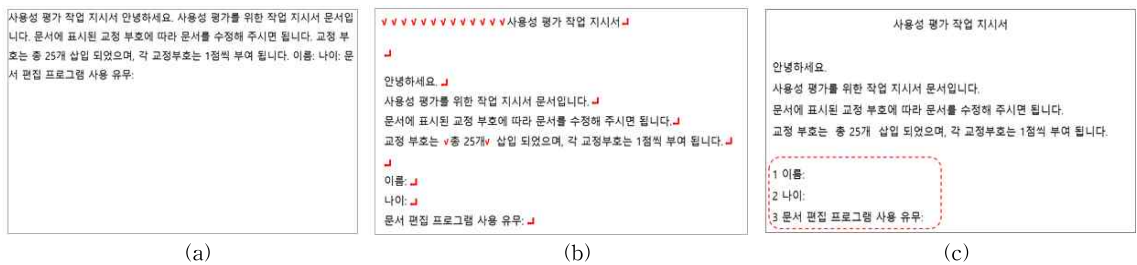


Fig. 7. Document for Editing (a) Unedited Document (b) Document with Calibration Marks (c) Final Edited Document
* The red mark indicated where the edit should be performed.

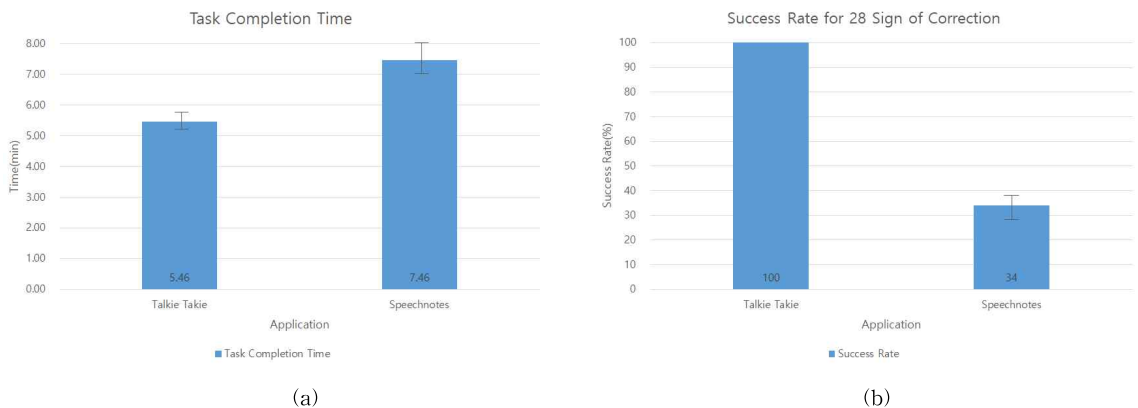


Fig. 8. Evaluation Results for Cursor Function, All Charts Include Standard Deviation. (a) Task Completion Time by Applications (b) Success Rate for 28 Sign of Correction by Applications

을 편집 하였다. 이는 Speechnotes에서 커서의 움직임에 관한 음성 명령이 줄 바꿈에 관한 명령어만 제공하여 편집이 어려웠기 때문이다.

6. 결 론

본 논문에서는 음성 인식 기능의 수행 체계를 음성 텍스트 입력 모드와 음성 명령 모드로 구분 설계하여 신체 및 환경적 제약 상황에서 사용자의 음성을 통해 기록 및 편집하고 어플리케이션을 제어할 수 있는 음성 명령 기반 모바일 기록 어플리케이션 Talkie Takie를 구현 하였다. 본 시스템은 안드로이드 환경에서 구현 되었으며, 음성 인식 API를 사용해 입력 받은 음성 명령어를 분석하여 특정 기능을 수행하도록 하였다. 문서 작성을 위한 음성 인식과 명령어 수행을 위한 음성 인식이 구분되어야 하므로 음성 인식 모드를 구분하여 '라이팅 모드'와 '커맨드 모드'를 제공하였다. 또한, 문서의 편집을 위한 커서의 역할을 쏘킷 기능으로 제공하여 기록된 문서에서 자유롭게 커서를 이동할 수 있도록 하여 모바일 환경에서의 기록 활동의 접근성 및 문서 편집의 편의성을 향상 시켰다. 10명의 사용자를 대상으로 사용성 평가를 통해 음성 모드의 기능들과 모드 변환이 정상적으로 작동하는지 검증하였으며, 안드로이드 마켓에서 가장 많은 다운로드를 기록한 어플리케이션 Speechnotes와의 비교 실험을 통해 문서 수정 편집에서 커서 조작의 기능성과 효율성을 검증하였다.

본 연구에서 제안한 음성 인터페이스 설계를 활용한 모드 체인지를 통해 음성 인식을 단순 문장입력 뿐 아니라, 편집 명령으로도 편리하게 사용될 수 있음을 확인하였다. 이는 운전, 운동, 요리중과 같이 두 손을 온전하게 사용하지 못하는 환경에서 안전하게 중요한 내용을 기록 및 수정이 가능하게 한 것이다. 또한, 손사용이 불편한 장애 사용자들이 일반 사용자들과 같은 음성을 통해 스마트폰에 문자 기록할 수 있게 함으로써 그들의 기억을 기록하며, 다양한 메모 작성을 가능하게 하였다. 마지막으로 음성 인식의 한계점들이 극복된 후 본 논문에서 제안한 음성 인식 설계를 활용하면 모드 체인지를 통해 음성 인식을 한 가지 기능에 머물지 않고 다수의 수행 체계를 설계할 수 있을 것으로 기대되며, 다양한 분야에 활용될 수 있을 것으로 기대한다.

REFERENCE

- [1] J.H. Park, S.B. Lim, J.H. Yook, and J.W. Lee, "An Analysis on the Disability Types and Requirements for Developing Daisy Reading Assistive Devices," *Journal of Special Education and Rehabilitation Science*, Vol. 56, No. 3, pp. 503-520, 2017.
- [2] H.Y. Kim and S.B. Lim, "Accessibility Automatic Inspector Library for EPUB and its Components," *Journal of Korea Multimedia Society*, Vol. 20, No. 2, pp. 330-335, 2017.
- [3] Voice Recognition System, <http://blog.lgcns.com/711> (accessed Mar., 15, 2018).
- [4] J.R. Choi, J.S. Hwang, E.J. Sin, and S.B. Lim, "A Feedback Clue Model for Dynamically Updating e-book Content from User Feedback," *Journal of Korea Multimedia Society*, Vol. 20, No. 2, pp. 313-321, 2017.
- [5] Google Docs, <https://docs.google.com/> (accessed Mar., 5, 2018).
- [6] SpeechNotes, <https://play.google.com/store/apps/details?id=co.speechnotes.speechnotes> (accessed Mar., 12, 2018).
- [7] Strabase, *Platform Big 3's Voice Recognition UI Competitive Landscape Analysis*, Strabase Issue Alert, 2011.
- [8] J.H. Park, S.B. Lim, and J.W. Lee, "A Voice Annotation Browsing Technique in Digital Talking Book for Reading-disabled People," *Journal of Korea Multimedia Society*, Vol. 16, No. 4, pp. 510-519, 2013.
- [9] D.G. Jeong, "Trend on Artificial Intelligence Technology and Its Related Industry," *Korea Institute of Information Technology Magazine*, Vol. 15, No. 2, pp. 21-28, 2017.
- [10] Android Speech API, <https://developer.android.com/reference/android/speech/package-summary.html> (accessed Mar., 20, 2018).



임 순 범

1982년 서울대학교 계산통계학과 (학사)
1983년 한국과학기술원 전산학과 (석사)
1992년 한국과학기술원 전산학과 (박사)

1989~1992년 (주)휴먼컴퓨터 창업 (연구소장)
1992~1997년 (주)삼보컴퓨터 프린터개발부 부장
1997~2001년 건국대학교 컴퓨터과학과 교수
2001년~현재 숙명여자대학교 IT공학과 교수
2006년 University of Colorado 방문교수
2014년~ 한국멀티미디어학회 회장, 명예회장
관심분야: 컴퓨터 그래픽스, 웹/모바일 멀티미디어 응용, 디지털 방송, 전자출판(폰트, 전자책, XML 문서), User Interface



박 세 아

2017년 숙명여자대학교 IT공학과 (학사)
관심분야: 웹/모바일 멀티미디어 응용, 웹 개발, User Interface 등



이 무 닉

2017년 숙명여자대학교 IT공학과 (학사)
관심분야: 웹/모바일 멀티미디어 응용, 웹 개발, User Interface 등



박 주 현

2010년 2월 숙명여자대학교 멀티미디어학과(이학사)
2012년 8월 숙명여자대학교 멀티미디어학과 (이학석사)
2013년 3월~현재 숙명여자대학교 IT공학과 박사 과정 수료

관심분야: 멀티모달 인터페이스, UI/UX, 접근성, 장애인 접근성, HCI, 웹/모바일 멀티미디어 응용 등