

學 士 學 位 論 文

스마트미러를 활용한 면접 대비 프로그램

충남대학교

공과대학 컴퓨터공학과

201000000 홍 길 동

201000000 성 춘 향

201000000 이 순 신

지도교수 김 기 일

2020 年 2 月

스마트미러를 활용한 면접 대비 프로그램

지도교수 김 기 일

이 논문을 공학사학위
청 구 논 문 으 로 제 출 함

2019 年 11 月

충 남 대 학 교

공과대학 컴퓨터공학과

201000000 홍 길 동

201000000 성 춘 향

201000000 이 순 신

목 차

I. 서 론	1
1. 연구 필요성	1
1.1 취업난으로 인한 면접의 중요성 증가	1
1.2 면접 준비과정에서의 부담	2
1.3 기존 연구의 단점	3
2. 관련 연구	5
2.1 음성 인식	5
2.1.1 Google Cloud Speech-to-Text API	5
2.2 모션 인식	5
2.2.1 OpenCV	5
2.2.2 Harr-like feature	6
2.2.3 AdaBoost 알고리즘	6
2.2.4 Cascade Classifier	6
II. 본 론	7
1. 설 계	7
1.1 하드웨어 구성	7
1.1.1 스마트 미리.	7
1.2 기능	7
1.2.1 면접 연습	7
1.2.2 결과 확인	8
1.2.2 사용법 확인	8
2. 구현	8
2.1 구현 환경 및 사용 언어	8
2.2 UI 구성	9
2.2.1 시작 화면	9

2.2.2 면접 연습 화면	10
2.2.3 분석 결과 확인 화면	11
2.2.4 사용법 설명 화면	13
2.3 면접 연습 기능 구현	13
2.4 분석 기능 구현	14
2.5 스마트 미러 구현	15
2.6 시스템 분석 및 문제점	15
2.6.1 시스템 분석	15
2.6.2 문제점	16
III. 결 론	18
1. 구현 사항	18
2. 활용방안 및 기대 효과	18
2.1 사용자 측면	18
2.2 비즈니스 측면	19
참고문헌	20

그림목차

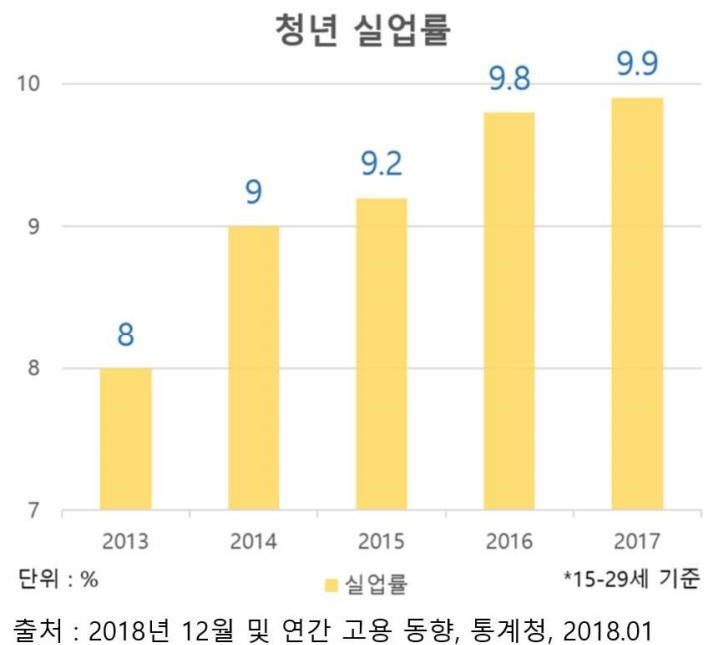
< 그림 1> 청년 실업률	1
< 그림 2 > 면접 비용 지출 부담 조사 결과	3
< 그림 3> ‘꿈날개’ 모의 면접 예시	4
< 그림 4 > Harr-like feature	6
< 그림 5 > 시작화면	9
< 그림 6 > 면접 연습	10
< 그림 7 > 결과	11
< 그림 8 > 결과 목록	12
< 그림 9 > 결과 삭제	12
< 그림 10 > 사용법	13
< 그림 11 > 스마트 미러 예시	15

I. 서 론

1. 연구 필요성

1.1 취업난으로 인한 면접의 중요성 증가

우리나라의 청년 실업률이 갈수록 증가하고 있다. 통계청 자료에 따르면 2018년 청년 실업률이 9.5%에 육박했다. 해당 자료는 통계청에서 15-29세 사이의 청년을 대상으로 조사한 전국 청년 실업률 자료이다.[1] 청년 실업률이 8%였던 2013년에 비해 5년 사이 약 2% 증가했다. 또한 한국경영자총협회 노동 통계에 따르면 2018년 연간 실업률은 3.8%로 2001년(4.0%) 이후 17년 만에 최고치를 기록했다. 2018년 한 해 동안 실업자는 107만 3천명으로 이 또한 2000년 이후 최고치이다. 이를 통해 갈수록 취업난이 심각해지고 있음을 알 수 있다.



<그림 1> 청년 실업률

취업난이 발생하는 원인 중 하나는 인구에 비해 일자리가 턱없이 부족하기 때문이다. 제한된 일자리에 많은 사람들이 지원하고 고용주는 그들 중 제한된 인원을 선발해야한다. 그러므로 더 우수한 인재를 선발하기 위한 다양한 시도의 결과로 서류, 인적성 평가, 필기시험, 면접 등 많은 절차들이 생겨났다.

그 중 가장 중요하고 보편적인 절차는 면접이다. 한국경영자총협회 조사에 따르면 최종 합격 결정 시 면접이 100점 만점에 48점에서 61점 정도를 차지한다고 한다. 대부분의 기업에서는 인재 채용 최종단계에서 면접을 포함시키고 있다. 일부 기업은 서류전형 이후 면접 절차만을 포함시키기도 한다. 그만큼 면접은 채용절차 중 많은 부분을 차지한다. 그러므로 면접에서 좋은 인상을 남기는 것이 중요하다. 본 연구는 사용자가 면접 연습을 할 수 있는 프로그램 개발을 목표로 한다.

1.2 면접 준비과정에서의 부담

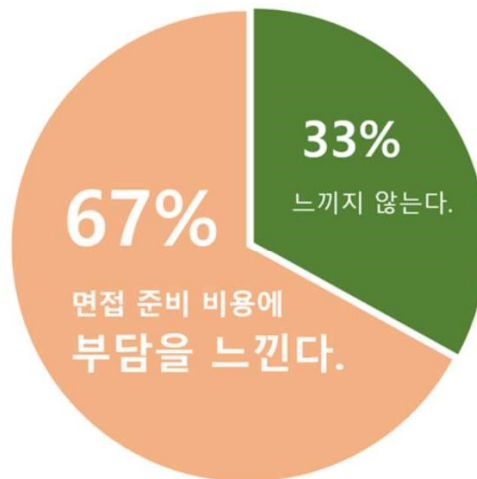
대부분의 기업에서 면접 절차를 필수적으로 포함시키게 되면서 취업 준비생들에게 면접 준비는 취업 준비과정에서 필수 요소가 되었다. 면접 준비 과정에서는 여러 요소에 의해 비용이 발생한다. 면접 준비 비용에는 면접 의상 구입비, 교통비, 헤어 및 메이크업 비, 면접 관련 도서 구입비, 컨설팅 등 사교육비 등이 포함된다.

많은 취업 준비생들은 이러한 면접 준비 비용에 부담을 느끼고 있다. 국내 구인구직 매칭플랫폼 ‘사람인’이 구직자 1,904명을 대상으로 실시한 면접 준비 비용 조사에 따르면 응답자의 67%가 면접 비용에 부담을 느낀다고 답했다. 면접 비용에서 가장 부담되는 면접 지출은 면접 의상 구입비(37.6%), 교통비(29.3%), 컨설팅등 사교육비(15.1%), 헤어 및 메이크업 비(6%), 면접 관련 도서 구입비(4.9%) 순이다.

정부에서는 이러한 취업 준비생들의 경제적 부담을 덜어주기 위한 다양한 지원을 하고 있다. 한 예를 들자면 정부에서는 청년희망아카데미를 통해 구직활동을 하는 대한민국의 만 34세 이하의 구직청년을 대상으로 면접 준비 비용을 지원하는 정책을 실시하고 있다. 지원 금액은 1인당 60만원으로 신청한 날부터 6개월 간 정장대여료, 이력서 사진촬영 비용, 숙박비, 교통비 등을 지원받을 수 있다.

이와 같은 정책들 덕분에 복장이나 교통비 등의 면접 준비 비용은 줄일 수 있지만 컨설팅 등 사교육 비 또는 면접 관련 도서 구입비와 같이 면접을 연습하는데 드는 비용은 줄이기 어렵다. 따라서 취업 준비생들의 면접 연습 비용을 절감할 필요가 있다.

면접 비용 지출 부담



자료제공 : 사람인

[사람인 뉴스]"구직자 67%, 면접 비용 지출 부담된다! 면접 1회당 6.5만원 지출", 취업스펙뉴스, 2019.10.01

< 그림 2 > 면접 비용 지출 부담 조사 결과

본 연구는 면접 준비과정에서 사용자가 부담하게 될 면접 준비 비용 중 컨설팅과 같은 사교육비 절감을 목표로 사용자가 스스로 면접 연습을 할 수 있는 스마트 미리 개발을 목표로 한다.

1.3 기존 연구의 단점

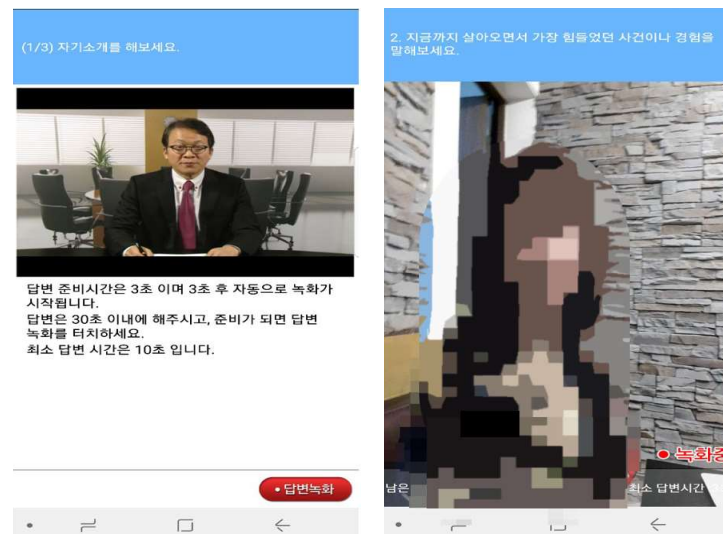
면접의 중요성이 강조되면서 면접 준비를 도와주는 선행 기술들과 이를 사용한 여러 프로그램들이 존재한다. 여러 프로그램 중 적용 기술에 따라 두 가지로 분류하였다.

첫 번째는 ‘꿈날개 모바일 면접 시스템’과 같이 가상의 면접관이 사용자에게 랜덤으로 선정한 질문을 제시하고 사용자는 이에 대답을 하는 식으로 면접 연습을 진행하는 방식의 프로그램이다.

면접관이 랜덤으로 선정한 질문을 제시하는 것은 대부분의 면접 연습 프로그램의 공통 사항이다. 대부분의 면접 준비 프로그램들은 면접관이 제시하는 질문에 대답하는 사용자의 모습을 녹화하여 제공하는 방식으로 면접 연습을 도와준다. 하지만 사용자가 자신의 모습을 스스로 점검하는 것은 사용자가 자신의 모습을 스스로 판단하고 분석하는 것이기 때문

에 객관성이 떨어질 수 있다. 사용자들은 자신의 모습에 익숙해져 있기 때문에 자신의 녹화 영상을 보고 과대평가하거나 과소평가할 여지가 있다.

이러한 기존 연구의 단점을 보완하기 위해 본 연구는 사용자의 면접 연습 평가에 관한 객관적 평가 기준을 정하고 그 기준에 맞추어 사용자의 행동을 분석하는 기능을 포함한다.



<그림 3> ‘꿈날개’ 모의 면접 예시

두 번째는 ‘인헨스유’와 같이 사용자의 면접 태도에 대한 분석 기능을 제공하는 프로그램이다. 첫 번째 프로그램들이 면접 연습을 하는 사용자에게 대한 녹화기능만 제공했다면 두 번째 프로그램들은 분석 기능이 추가된 형태이다. ‘인헨스유’ 같은 경우에는 사용자의 표정과 음성을 기반으로 사용자의 면접 태도를 분석한다.

면접을 볼 때 좋은 인상을 주기 위해서는 얼굴 표정과 음성도 중요하지만 자세도 매우 중요하다. 면접 평가 요소는 각 기업의 인재 상에 따라 차이가 나지만 공통적으로 면접관에게 나쁜 인상을 줄 수 있는 안 좋은 습관들이 있다. 예를 들어 한쪽으로 기운 자세나 과장되거나 산만한 동작 등이 안 좋은 습관에 포함된다. 본 연구는 모션 인식 기술을 통하여 사용자의 행동을 분석하고 안 좋은 습관을 탐지하여 분석 결과를 제공하는 기능을 포함한다.

2. 관련 연구

2.1 음성 인식

2.1.1 Google Cloud Speech-to-Text API

Google Cloud Speech-to-Text API [1] 는 구글의 머신러닝 기술을 이용하여 음성을 분석해 오디오를 텍스트로 변환하는 기술이다. 120개 이상의 언어를 인식할 수 있으며 일상 대화를 잘 인식하도록 설계되어 고유명사 변환 및 언어 형식 지정 기능을 포함한다. 길이에 상관없이 음성을 텍스트로 변환해주며 이 API를 통해 음성 명령 및 제어 기능을 구현할 수 있다.

본 연구는 음성인식 기술을 기반으로 사용자의 음성을 인식하여 프로그램을 진행하도록 설계되었다. 따라서 본 연구는 구글 음성인식 API의 기능 중 실시간 스트리밍 기능을 활용하여 프로그램이 진행되는 동안, 특히 면접 연습이 진행되는 동안 사용자의 명령어, 면접 연습 중 사용자의 답변을 변환하여 활용한다. 하지만 클라우드(Cloud) 기반으로 인터넷 통신을 통한 시간지연 및 통신오류가 있다는 단점이 존재한다.

2.2 모션 인식

2.2.1 OpenCV

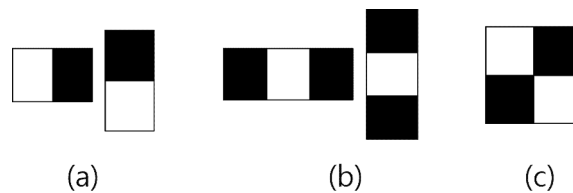
OpenCV(Open Source Computer Vision) [2][3] 는 인텔에서 c/c++ 언어를 이용해 실시간 영상처리, 컴퓨터 비전 등을 목적으로 개발한 라이브러리이다. 리눅스, window 등 다양한 플랫폼을 지원하며, python, Java 등 프로그래밍 언어로도 구현할 수 있다. open-source BSD 라이선스 기반으로 연구 또는 상업적 용도로 자유롭게 사용가능하다.

OpenCV 는 컴퓨터 비전과 기계학습과 관련된 알고리즘을 가지고 있으며 물체 식별 및 추적, 객체 분류 등 영상 처리를 지원 라이브러리와 함께 제공하기 때문에 프로그래밍하기 편하고 기업, 연구단체, 정부 기관에서 광범위하게 사용되고 있다.

본 연구에서는 사용자를 인식하고 사용자의 행동을 분석할 목적으로 사용되었으며 python으로 구현한다.

2.2.2 Harr-like feature

이미지에서 특징을 찾는 방법 중 하나인 Harr-like feature [4][5][6] 은 Viola 와 Jones 가 제안한 방법으로 영역과 영역의 밝기 차를 이용한 것으로 아래 사각형 영역에 따른 기본 feature 들에 따라 조합하여 특정 위치 및 분포에 따라 물체에 대한 특징을 추출한다. 각 feature 들의 특징 값은 아래 그림과 같이 서로 인접한 사각형 영역들의 밝기 차이로 계산되며 임계값 이상인 것을 찾아 파악하려는 대상을 추측한다.



< 그림 4 > Harr-like feature

2.2.3 AdaBoost 알고리즘

AdaBoost [6] 는 여러 개의 약 분류기를 결합해 강 분류기를 구성하는 알고리즘이다. 학습을 진행하는 과정에서 맞게 분류된 데이터들의 가중치는 낮추고, 잘못 분류된 데이터들의 가중치는 높여 다음 분류기에 반영한다. 따라서 최종 분류기는 약 분류기들에 대한 가중치 합이다.

2.2.4 Cascade Classifier

이미지에 윈도우를 사용하여 여러 개의 Harr-like feature 적용하면 모든 특징을 적용하지 않고 여러 단계의 그룹으로 묶어 사용하는 방식이다. 첫 번째 단계의 특징에서 측정하고자 하는 객체의 영역이 아니라는 판정이 나면 다음 위치로 윈도우를 이동하고, 객체의 영역이라는 판정이 나면 현재 윈도우가 위치한 곳에 다음 단계의 특징을 적용한다.

II. 본 론

1. 설 계

본 연구는 사용자가 타인의 도움 없이 혼자서도 면접 준비를 할 수 있으며 면접 연습 과정에서 사용자의 비언어적 요소에 대한 분석을 제공하는 면접 대비용 스마트 미러에 대한 내용이다, 이에 본 연구는 사용자에게 면접 예상 질문을 하는 인터페이스를 제공하며 프로그램 진행을 위한 음성인식 기능, 사용자의 안 좋은 습관을 분석하는 기능을 가진다.

1.1 하드웨어 구성

1.1.1 스마트 미러.

스마트 미러는 거울(mirror) 와 디스플레이(display) 의 결합으로 차세대 디스플레이이다. 평소에는 거울 역할을 하면서 뒷면에 부착된 모니터로 원하는 내용을 출력할 수 있는 장치이다. 본 연구는 아크릴판에 하프 미러 필름을 부착해 스마트 미러로 사용하며 뒷면에 모니터를 부착해 면접 연습에 필요한 과정을 출력한다.

1.2 기능

본 프로그램은 다음과 같은 과정으로 진행된다. 각 과정은 사용자의 음성으로 수행된다.

1.2.1 면접 연습

1. 사용자가 ‘면접 연습’ 선택지를 선택하면 모의 면접을 시작한다.
2. 면접은 사용자의 음성 인식을 통해 진행되며 면접 영상은 카메라로 녹화되고 질의응답 내용은 채팅형식으로 기록된다.
3. 면접 종료 후 기록된 면접 연상이 분석되고 사용자는 분석 결과를 보고서 형태로 확인 가능하다.

1.2.2 결과 확인

1. 사용자가 ‘결과 보기’ 선택지를 선택하면 저장된 분석 결과 보고서 목록을 화면에 출력한다.
2. 사용자는 확인하고 싶은 결과 보고서를 선택하고 본인 확인 후 열람 가능하다.

1.2.2 사용법 확인

1. 사용자가 ‘사용법’ 선택지를 선택하면 사용법에 대한 설명을 화면에 출력한다.
2. 사용자는 각 단계에 대한 설명을 확인할 수 있다.

2. 구현

2.1 구현 환경 및 사용 언어

본 프로그램은 Window10 환경에서 개발되었다. 전체 프로그램은 python3.7을 사용하여 구현했고 HTML, CSS, 자바스크립트를 활용하여 프로그램의 UI를 구현 하였다. 프로그램에서 사용되는 데이터들은 local Storage 에 데이터를 저장하는 방식으로 관리하였다.

2.2 UI 구성

2.2.1 시작 화면

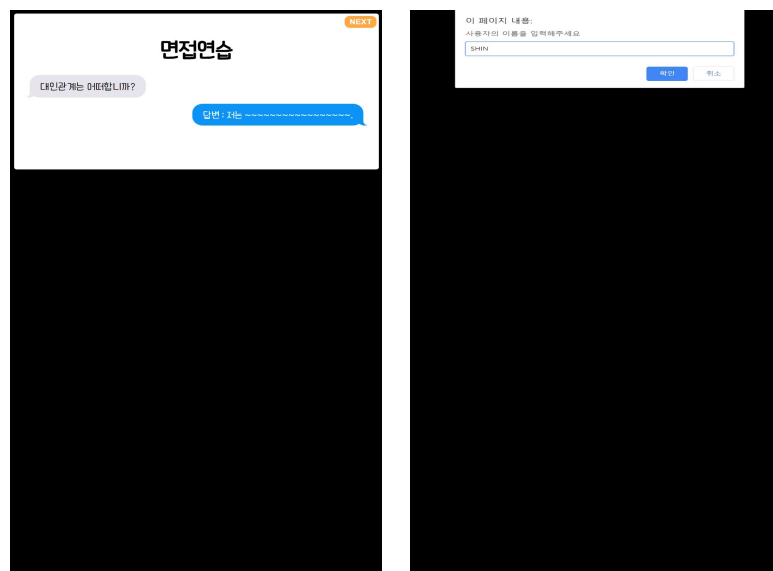
프로그램을 시작하면 아래 그림과 같이 화면이 출력된다. 시작화면에는 ‘면접연습’, ‘결과보기’, ‘사용법’의 세 가지 선택지가 존재하고, 사용자가 ‘면접연습’을 선택하면 사용자는 면접연습을 수행할 수 있다. ‘결과보기’를 선택하면 기존에 수행했던 면접들의 결과 목록과 그에 해당하는 내용을 확인 할 수 있다. 마지막으로 ‘사용법’을 선택하면 프로그램의 사용법을 확인할 수 있다.



< 그림 5 > 시작화면

2.2.2 면접 연습 화면

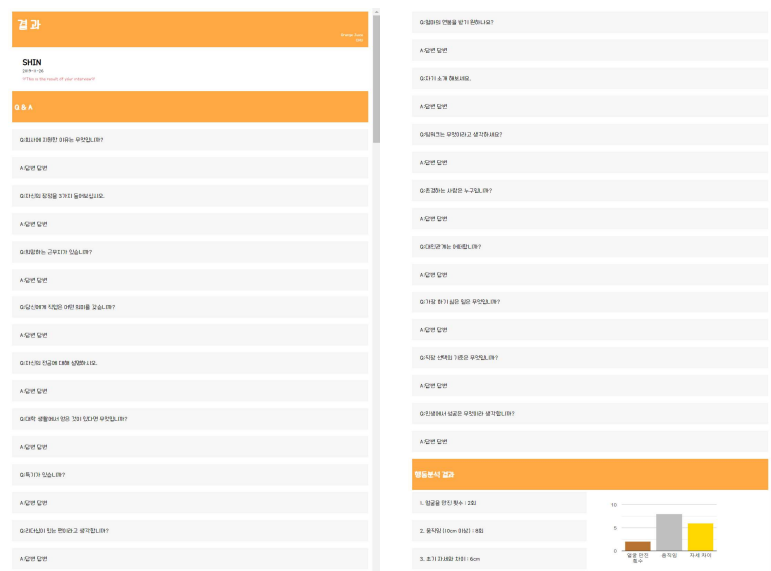
사용자가 시작화면에서 ‘면접연습’을 선택하면 면접 연습 화면으로 이동한다. ‘면접 연습’이라고 되어있는 아래 그림의 상단 부분에는 질문과 답변이 뜨며 일정한 시간이 지나면 다음 질문으로 변경된다. 그림의 아래에 존재하는 검은 화면은 스마트 미러를 이용해 사용자가 본인의 모습을 확인하며 면접을 진행할 수 있는 부분이다. 면접이 종료되면 이후 사용자가 본인의 면접 결과를 보기 위해 확인할 요소로서 사용자의 이름을 입력하라는 메시지가 뜬다. 사용자는 사용자의 이름을 입력한다.



< 그림 6 > 면접 연습

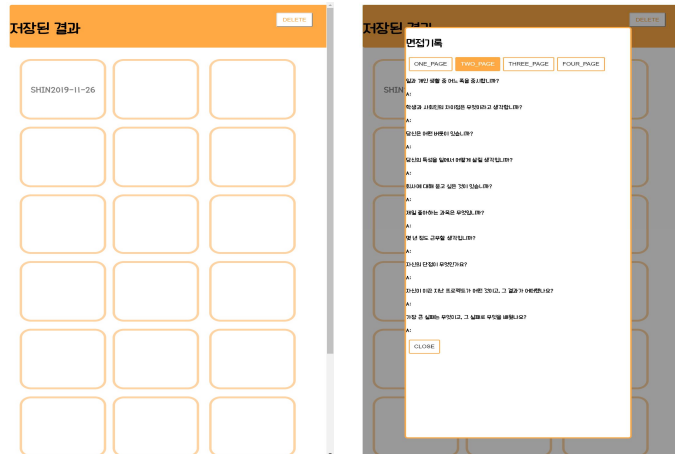
2.2.3 분석 결과 확인 화면

사용자의 면접 연습이 종료된 이후 그 결과는 사용자가 확인할 수 있도록 보고서 형태로 사용자가 입력한 이름과 연습을 진행한 날짜가 아래 그림의 상단에 출력되고 그 아래 부분에는 기록된 사용자가 받았던 질문들과 그에 대한 사용자의 답변을 볼 수 있다. 행동 분석 결과에는 면접 연습을 하던 중 사용자의 특정 행동 패턴에 대해 분석한 결과가 표와 그래프로 표시된다.



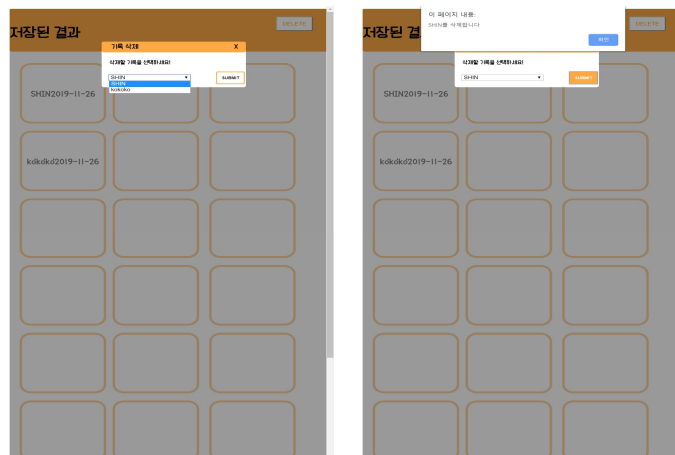
< 그림 7 > 결과

시작화면에서 결과보기를 선택하면 지금까지 저장된 면접 연습 결과들을 볼 수 있다. 연습 결과는 사각형 모양에 사용자의 이름과 연습을 한 날짜와 같이 기록되어 있다. 사각형 모양을 클릭하면 면접 질문과 그에 해당하는 답변을 확인 할 수 있다. 총 4 영역으로 구성되어 있으며 버튼을 클릭하면 페이지가 이동하고 닫기 버튼을 누르면 창이 닫힌다.



< 그림 8 > 결과 목록

저장된 결과들 중 삭제하고 싶은 결과가 있는 경우 삭제 버튼을 누르면 결과를 삭제 할 수 있는 페이지가 뜬다. 목록 중 삭제하고 싶은 결과를 선택하고 확인 버튼을 누르면 해당 결과가 삭제된다.



< 그림 9 > 결과 삭제

2.2.4 사용법 설명 화면

시작화면에서 사용법을 선택하면 사용법 설명 화면으로 이동한다. 해당 화면에서는 프로그램의 전체적인 사용방법에 대해 3가지 단계로 나누어 설명한다. 자동 슬라이드 기능으로 구현되어 있어 사용자는 별다른 행동 없이도 사용방법에 대해 볼 수 있다.



< 그림 10 > 사용법

2.3 면접 연습 기능 구현

본 연구에서 구현한 프로그램은 음성인식 기술을 기반으로 사용자의 음성을 인식하여 프로그램을 진행한다. 구글 음성인식 API를 사용하여 사용자의 음성을 text 로 변환하고 해당 text 에서 특정 단어를 추출한다. 추출한 단어에 해당하는 기능을 수행하고 일부 단어는 시작 화면 또는 전 화면으로 돌아가는 기능을 수행한다.

시작화면에서 사용자가 '면접연습' 이라는 단어가 포함된 말을 할 경우 면접 연습 메뉴를 수행한다. 면접 연습 메뉴에서 랜덤하게 선정된 질문을 일정 시간 간격으로 제시하고 사용자는 해당 질문에 대한 답변을 제한된 시간동안 대답한다. 사용자의 답변은 결과화면 제공 시 사용하기 위하여 별도의 장소에 저장된다.

또한 면접 연습을 진행하는 동안 사용자의 모습은 분석을 하기 위해 녹화된다. 사용자가 ‘다음’ 이라는 단어가 포함된 말을 할 경우 면접 연습을 중단하고 결과화면 출력 화면으로 넘어간다. 사용자의 이름을 입력받은 후 사용자의 이름, 면접 연습을 진행하는 동안 저장된 질문 및 답변, 면접 연습 진행 중 저장된 영상에 대한 분석 결과가 포함된 결과 화면을 출력한다. 면접 연습 중간에 ‘처음으로’ 라는 단어가 포함된 말을 할 경우 시작 화면으로 돌아간다.

2.4 분석 기능 구현

본 연구에서 구현한 프로그램은 면접이 종료된 이후 사용자의 행동을 분석하기 위해 카메라로 녹화한 사용자의 면접 과정이 담긴 영상을 가져온다. 이 영상으로 사람의 상반신과 얼굴을 찾는데 사용자의 얼굴, 상반신을 검출하기 위해서 위에서 언급한 Harr-like feature 을 이용한다. 본 프로그램에서는 OpenCV 를 사용하는데 OpenCV 에서는 이 Harr-like feature 을 적용해 얼굴, 눈 등 판별에 필요한 미리 학습시켜 놓은 분류기를 제공한다. 위의 분류기는 xml 파일로 제공되는데 본 프로그램은 이 xml 파일을 사용해 분석에 필요한 사람의 상반신과 얼굴 인식을 수행한다.

처음 필요한 xml 분류기를 로드한 후 입력된 사용자의 면접 영상을 흑백으로 변환한다. 위의 Harr-like feature 를 이용하기 때문에 물체를 인식하는데 있어서 색상 데이터는 필요하지 않고, 영상처리와 물체 인식할 때의 계산을 빠르게 하기 위해서이다.

또한 조명, 방향 등의 차이를 없애고 인식을 잘 수행하도록 히스토그램 균등화 과정 [7] 을 진행한다. 히스토그램은 영상 안에서 픽셀들에 대한 명암 값의 분포를 나타내는 것인데 균등화 과정으로 그 값들을 넓게 분포해 고른 정보를 가질 수 있도록 한다. 이 과정을 통해 어두운 영상은 밝게, 너무 밝은 영상을 어둡게 하는 등 적당한 명도 값을 유지해 영상의 질을 향상시킨다. 영상 속의 물체를 구분하지 못하거나 인식을 진행할 때의 어려움을 최소화하기 위해서이다.

영상의 각 프레임마다 사용자의 상반신 인식 후 얼굴을 인식한다. 이때 사용자의 움직임은 측정하기 위해서 각 상반신과 얼굴의 인식된 범위의 좌표를 저장한다. 이 좌표를 활용해 전 프레임에 비해 움직인 정도, 각 좌표들의 차이가 임계값보다 큰지 작은지 확인한 후 임계값보다 클 때마다 움직임 횟수를 증가시킨다. 영상 종료 후 각 움직임 횟수들을 출력한다.

2.5 스마트 미러 구현

아래의 사진은 실제로 제작한 스마트미러로 사용자는 본인의 모습을 확인하며 면접을 진행하는 한편 면접에 대한 안내 사항을 확인할 수 있다. 더불어 프로그램의 각 단계를 볼 수 있다.



< 그림 11 > 스마트 미러
예시

(스마트미러 실물 사진 첨부)

2.6 시스템 분석 및 문제점

2.6.1 시스템 분석

본 연구에서 구현한 프로그램은 사용자에게 면접 예상 질문을 제공하는 인터페이스 구축, 사용자의 면접 연습 영상 녹화, 프로그램 진행을 위한 음성 인식 기능, 사용자의 안 좋은 습관 분석 기능 구현을 목표로 했다.

본 연구에서 구현한 프로그램은 구글 음성 인식 기술을 기반으로 사용자의 음성을 text로 변환하여 각 기능을 수행한다. 사용자의 음성을 문장 단위로 그룹화 하여 문장 내의 특정 단어가 포함되어있는지를 판단한다. 구글에서 제공하는 음성인식 API를 사용하기 때문에 해

당 API 의 알고리즘을 정확히 알기 어렵다. 또한 구글 음성인식 API 는 cloud 기반으로 구동하기 때문에 반드시 인터넷 통신이 되어야 한다.

본 연구에서 구현한 프로그램은 html 파일과 jsp 파일을 python 파일에서 구동하는 방식으로 구성되어 있다. python 파일 내에는 사용자의 음성을 text 로 변환하고 기능 구동을 위한 단어가 포함되어 있는지에 따라 각 기능에 해당하는 html 파일 또는 jsp 파일을 호출하는 코드가 포함되어 있다. 또한 python 파일에는 사용자가 면접 연습을 하는 동안 답변을 저장하는 코드가 포함되어 있다.

본 연구에서 구현한 사용자 행동 분석 기능은 python에서 제공하는 OpenCV 를 사용하였다. OpenCV에서는 Haar-like feature를 적용해 얼굴, 눈 등 판별에 필요한 미리 학습시켜 놓은 분류기를 제공한다. 분류기는 xml 파일로 제공되는데 본 프로그램은 이 xml 파일을 사용해 분석에 필요한 사람의 상반신과 얼굴 인식을 수행한다. 이때 조명, 방향 등의 차이를 없애고 인식을 잘 수행하도록 히스토그램 균등화 과정을 진행한다. 영상의 각 프레임마다 사용자의 상반신 인식 후 얼굴을 인식하는데 영상처리 작업은 많은 연산을 해야 하므로 연산 속도가 느려져 영상 속도 또한 느려지는 상황이 발생한다.

2.6.2 문제점

본 연구는 인터넷 통신 없이 사용가능한 스마트 미러를 활용한 면접 대비 프로그램을 구현하도록 설계하였다. 본 프로그램을 구현하는 과정에서 여러 요소를 고려한 결과 본 프로그램에 가장 적합하다고 판단한 구글 음성인식 API 를 사용하였고, 이 API 는 cloud 기반으로 구동하기 때문에 인터넷 연결이 가능한 환경에서만 구동 가능하다. 따라서 초기 설계와 다르게 인터넷 통신이 필요한 프로그램을 구현하게 되었다. 또한 인터넷 속도에 따라 음성을 인식하는 속도가 달라져 인터넷 속도에 따라 프로그램의 진행 속도에 차이가 발생했다. 또한 인터넷 통신으로 시간 지연 및 통신 오류가 발생할 수 있는 단점이 존재하는데 향후 인터넷 통신 또는 인터넷 속도에 영향을 받지 않는 다른 음성인식 기술을 적용하는 것이 바람직 할 것으로 예상된다.

본 연구는 사용자의 음성만으로 프로그램이 진행되는데 python 파일에서 사용자의 음성을 text 로 변환한 후 해당 기능에 해당하는 html 파일 또는 jsp 파일을 호출하기 때문에 웹 자체에서 동작을 할 수 없다. 따라서 프로그램이 음성인식 기술에 종속적으로 구동된다.

본 연구는 면접을 진행할 때 머리, 얼굴 등을 만지는 행위 또는 과한 손동작 등이 부정적인 영향을 미친다는 설문 조사 결과[9]를 참고해 손과 얼굴을 기준으로 사용자의 행동을 판단할 수 있도록 설계하였다. 하지만 손을 인식하는 xml 분류기를 OpenCV에서 제공하지 않아 따로 로드하였지만 OpenCV에 불러오지 못해 적용하지 못하였다.

이에 본 연구는 나쁜 자세인 ‘구부정한 자세’와 ‘자세를 자주 고쳐 앉는 태도’로 초기 설계와 달리 초점을 바꾸어 얼굴과 상반신을 기준으로 사용자의 행동을 판단하였다. 위의 초기의 설계와 현재의 기준을 함께 적용한다면 더 정확하고 객관성 있는 결과가 나올 수 있을 것이라고 생각한다.

또한 본 연구는 스마트 미러라는 물리적인 제약이 존재하는데 웹캠이 아닌 카메라를 필요로 한다. 이 문제를 해결하기 위해 분석 기능은 라즈베리파이의 운영체제 중 하나인 라즈비안에서 구동되고 그 외의 기능은 window 10 에서 구동되는데 분석 결과를 사용자에게 출력하기 위해서는 그 결과를 파일로 저장해 전송하는 인터페이스가 필요하다. 따라서 두 호스트 사이에서 파일 전송을 담당하는 인터페이스를 구현하고자 하였으나 실패하였다. 각각의 호스트에서 서로 다른 운영체제가 아닌 하나의 라즈베리파이에서 window 10 운영체제를 사용하여 이 문제를 해결할 수 있을 것이라고 생각하며 본 연구에서는 각 문제에 대해 적절한 해결책들을 바로 적용시키지 못한 것이 가장 큰 문제이다.

Ⅲ. 결 론

1. 구현 사항

본 연구는 사용자 스스로 면접 연습을 할 수 있는 스마트 미러를 활용한 면접 대비 프로그램 구현을 목표로 한다. 본 연구는 사용자에게 면접 예상 질문을 제공하는 인터페이스 구축, 사용자의 면접 연습 영상 녹화, 프로그램 진행을 위한 음성 인식 기능, 사용자의 안 좋은 습관 분석 기능을 가진다.

본 프로그램은 사용자가 면접 연습 기능을 실행하는 동안 사용자에게 면접 예상 질문을 10초 간격으로 사용자에게 제공한다. 해당 질문은 빈출도가 높은 면접 질문 중 임의로 선정한 질문들이다. 본 프로그램은 구글 음성인식 API 를 사용하여 사용자의 음성을 text 로 변환한 후 프로그램 진행에 사용한다.

사용자가 스마트 미러에 비친 자신의 모습을 보며 면접 연습을 하는 동안 카메라를 통해 사용자가 면접 연습을 진행하는 모습을 녹화한 후 별도의 장소에 저장한다. 저장된 영상에서 기준에 따라 좋지 않은 습관을 분석하여 사용자에게 제공한다. 사용자의 머리와 상반신을 인식하여 사용자가 움직이는 정도를 측정한 후 지나친 움직임을 안 좋은 습관으로 판단하여 사용자에게 그래프와 표로 분석 결과를 제공한다.

2. 활용방안 및 기대 효과

2.1 사용자 측면

본 연구에서 구현한 스마트미러를 활용한 면접 대비 프로그램은 면접 연습이 필요한 사용자를 대상으로 한다. 면접 연습 기능을 사용하는 사용자는 기존의 질문 위주의 면접 연습에서 탈피해 무의식적 행동, 즉 안 좋은 습관을 탐지하고 평가 기준에 따라 객관적으로 사용자의 행동을 확인하고 개선할 수 있다.

또한 본 프로그램은 제한된 공간에서 외부의 도움 없이 스스로 면접 연습을 할 수 있다는 점에서 시간적, 물리적 제약이 감소하며 경제적 비용 부담이 감소한다.

2.2 비즈니스 측면

본 연구에서 구현한 스마트 미러를 활용한 면접 대비 프로그램은 면접 연습이 필요한 취업 준비생 또는 입시생을 대상으로 한다. 따라서 향후 특정 기업 및 학교와 연계하여 특정 기업 및 학교에 맞춤형 프로그램으로 발전시킬 수 있다.

참고문헌

- [1] Google Cloud, Cloud Speech-to-Text, <https://cloud.google.com/speech-to-text/>, 2019
- [2] Kari Pulli, Anatoly Baksheev, Kirill Korniyakov, Victor Eruhimov, Realtime Computer Vision with OpenCV, Communications of the ACM Volume 55 Issue 6, pages 61 - 69, 2012
- [3] OpenCV, <https://en.wikipedia.org/wiki/OpenCV>, 2019
- [4] Alexander Mordvintsev, Abid K., Face Detection using Haar Cascades, https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_objdetect/py_face_detection/py_face_detection.html, 2013
- [5] Paul Viola, Michael Jones, Rapid Object Detection using a Boosted Cascade of Simple Features, IEEE Computer Society Conference on Computer Vision and Pattern Recognition vol. 1, 2001, pages 511 - 518
- [6] 정병우, 박기영, 황선영, 객체 검출을 위한 빠르고 효율적인 Haar-Like 피쳐선택 알고리즘, 한국통신학회논문지 제 35권 제6호, 2013, pages 486 - 491
- [7] Robert Krutsch, David Tenorio, Histogram Equalization, Freescale Semiconductor Inc., 2011
- [8] midas HRi, 기업별/직무별 면접 질문 500개 모음, <https://www.midashri.com/blog/500interviewlist>, 2019
- [9] 김민진, 빵빵한 스펙을 무너뜨리는 ‘면접시 행동’ 5가지, https://news.chosun.com/site/data/html_dir/2016/01/15/2016011502457.html, 2016