

캡스톤 디자인 I 종합설계 프로젝트

프로젝트 명	장애인용 시선추적 기반 키오스크
팀 명	Eye can do it
문서 제목	중간보고서

Version	1.2
Date	2019-04-18


팀원	이용훈 (조장)
	한민인
	윤진교
	최휴영
	우리젠
지도교수	주용수 교수

CONFIDENTIALITY/SECURITY WARNING


이 문서에 포함되어 있는 정보는 국민대학교 전자정보통신대학 컴퓨터공학부 및 컴퓨터공학부 개설 교과목 캡스톤 디자인 수강 학생 중 프로젝트 “장애인용 시선추적 기반 키오스크”를 수행하는 팀 “Eye can do it”의 팀원들의 자산입니다. 국민대학교 컴퓨터공학부 및 팀 “Eye can do it”의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다.

문서 정보 / 수정 내역

Filename	중간보고서-장애인용 시선추적 기반 키오스크.doc
원안작성자	이용훈, 한민인, 최휴영
수정작업자	이용훈, 한민인, 윤진교, 최휴영, 우리젠


 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	장애인용 시선추적 기반 키오스크	
	팀 명	Eye can do it!	
	Eye can do it!	Version 1.3	2019-APR-18

수정날짜	대표수정자	Revision	추가/수정 항목	내 용
2019-04-15	이용훈	1.0	최초 작성	초안 작성
2019-04-16	최휴영	1.1	내용 수정	수정된 연구내용 추가
2019-04-17	한민인	1.2	내용 수정	향후 추진 계획 수정
2019-04-18	윤진교	1.3	내용 수정	최종 내용 보충 및 수정 마무리

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	장애인용 시선추적 기반 키오스크	
	팀 명	Eye can do it!	
	Eye can do it!	Version 1.3	2019-APR-18

목 차

1	프로젝트 목표	4
2	수행 내용 및 중간결과	5
2.1	계획서 상의 연구내용	5
2.1.1	얼굴, 눈 영역 검출에 따른 동공 영역 검출	5
2.1.2	검출된 영역의 좌표값을 실제 화면의 좌표값으로 변환	6
2.1.3	포인터 출력 및 클릭 기능 구현	6
2.1.4	동공의 떨림 보완, 정확도 향상	6
2.1.5	키오스크 웹 구현	7
2.2	수행내용	8
2.2.1	시선 추적 기술	8
2.2.2	Web Front-end	8
2.2.3	Web Back-end	8
3	수정된 연구내용 및 추진 방향	9
3.1	수정사항	9
4	향후 추진계획	10
4.1	향후 계획의 세부 내용	10
4.1.1	시선 추적 기술	10
4.1.2	Web Front-end	10
4.1.3	Web Back-end	10
5	고충 및 건의사항	11


 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	장애인용 시선추적 기반 키오스크	
	팀 명	Eye can do it!	
	Eye can do it!	Version 1.3	2019-APR-18

1 프로젝트 목표

본 프로젝트는 최근 많은 곳에서 사용되고 있는 키오스크를 사용하기 어려운, 손을 자유롭게 사용하지 못하는 장애인들을 위하여 시선추적 기술을 기반으로 한 키오스크를 제작하는 것이다. 시선추적 기술을 사용하여 장애인들은 눈으로 손의 기능을 대신 할 수 있으며, 키오스크를 통하여 원하는 정보를 얻거나 상품을 주문할 수 있게 된다.

또한, 실제 키오스크 시스템처럼 Web Front-End와 Back-End를 구현한다. Web Front-End에는 상품의 가격, 사진, 정보, 장바구니등을 구현한다. Web Back-End에는 상품이나 정보를 데이터 베이스에 저장하고, 사용자가 선택한 상품을 장바구니에 담아 주고 주문한 상품을 데이터베이스에 저장한 후에 Owner에게 전달해주어 실제 키오스크를 사용하는 것과 같이 구현할 예정이다.

시선추적 기술을 적용하여 손을 사용하지 않고도 시선의 움직임, 얼굴의 움직임을 통하여 원하는 상품정보를 얻을 수 있으며 눈을 깜빡이는 특정한 행동을 통하여 상품을 선택하고 주문할 수 있다. 기존 키오스크에 add-on 형식으로 웹캠과 해당 소프트웨어를 실행하기만 하면 사용할 수 있도록 간편한 서비스를 제공하는 것을 목표로 한다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	장애인용 시선추적 기반 키오스크	
	팀 명	Eye can do it!	
	Eye can do it!	Version 1.3	2019-APR-18

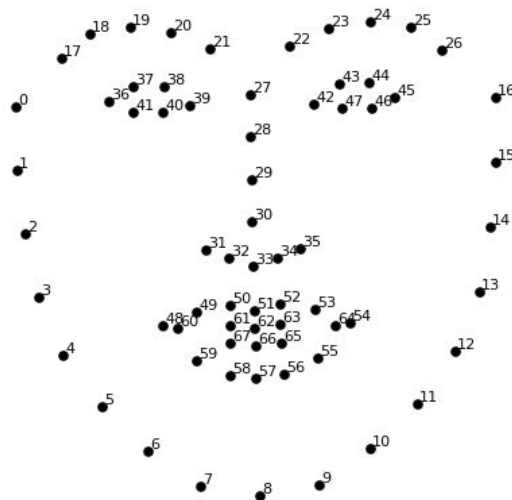
2 수행 내용 및 중간결과

2.1 계획서 상의 연구내용


시선추적 기술을 이용하여 Web에 구현되어 있는 키오스크 시스템을 이용한다. 일정한 거리에서 웹캠으로 사용자의 얼굴을 인식할 수 있게 한 후, 눈 영역과 동공 영역을 검출하여 시선을 추적하여 원하는 상품을 주문할 수 있게 한다. 이 때 Owner는 웹과 서버의 통신을 통하여 사용자가 선택한 상품의 정보를 정확하게 전달 받는다.

2.1.1 얼굴, 눈 영역 검출에 따른 동공 영역 검출

OpenCV를 사용하여 웹캠으로 사용자의 영상을 가져오게 되며 dlib을 사용하여 사용자의 얼굴영역을 검출하게 된다. 이 때 학습된 랜드마크 모델 데이터인 'shape_predictor_68_face_landmarks.dat' 을 사용한다. 이 데이터는 [그림 1]과 같이 사람의 얼굴에 68개의 점을 찍어 얼굴 영역을 인식하게 된다. 그 중에서도 얼굴 전체 영역에서 눈 영역을 찾아 그 범위 안에서 동공 영역을 검출하여 점을 찍어준다.



[그림 1] landmarks points

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	장애인용 시선추적 기반 키오스크	
	팀 명	Eye can do it!	
	Eye can do it!	Version 1.3	2019-APR-18

2.1.2 검출된 영역의 좌표값을 실제 화면의 좌표값으로 변환

2.1.1에서 검출해낸 영역을 영상 프레임마다 2차원 좌표값으로 변환해준다. 그 후 비례식 혹은 중심에서 이동한 거리를 계산하여 실제 화면에 포인터를 위치할 수 있도록 한다. 실제 화면의 좌표값은 사용자의 컴퓨터 해상도에 따라 달라지게 되는데 실제 사용할 화면의 해상도에 맞춰 설정한 후에 모든 해상도에서 사용할 수 있는 방법도 고려할 예정이다.


2.1.3 포인터 출력 및 클릭 기능 구현

2.1.2에서 실제 화면의 좌표값으로 변환해주면 Python 라이브러리인 Pymouse 혹은 PyAutoGUI를 사용하여 마우스를 제어한다.

클릭 기능을 구현하기 위해서는 사용자의 특정한 행동을 정의해야 하는데, 단순히 눈을 깜빡이는 행동을 하면 클릭이 되도록 구현하게 될 경우 일반적인 눈 깜빡임에도 반응하여 무의미한 클릭이 방해요소가 될 수 있을 것이다. 따라서 눈을 연속으로 두 번 깜빡이거나 한쪽 눈을 감게 될 경우 클릭이 되도록 할 예정이다. 이 때 눈의 깜빡임은 2.1.1에서 사용한 랜드마크 모델 데이터에서 눈 윗부분의 y축 좌표와 눈 아랫부분의 y축 좌표를 추출하여 두 좌표 사이의 거리가 일정 수준 미만으로 작아질 경우에 눈을 깜빡였다고 설정할 예정이다.

2.1.4 동공의 떨림 보완, 정확도 향상

위의 2.1.1~3을 수행하면서 동시에 진행해야 할 문제이다. 한 곳을 정확하게 응시하고 있더라도 동공의 중심은 미세하게 떨리고 있기 때문에 검출되어 표시한 점도 지속적으로 흔들리게 되어 불안정한 모습을 보여주고 있다. 이러한 방법을 해결하기 위하여 주파수의 경우 대역 제한에서 아이디어를 얻어 일정 범위 안에서 동공의 움직임을 감지하여 미세한 떨림을 방지할 수 있다. 예를 들어 x, y좌표의 동공의 움직임 값이 각각 최소 1~50, 1~30이라고 가정하면 1~9의 값은 무시하고 10이상의 움직임이 입력될 경우에만 동공이 움직였다고 인식하는 것이다. 그렇게 할 경우 frame 단위로 받아오는 이미지에서 동공의 떨림을 조금이라도 줄일 수 있을 것이다.


 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	장애인용 시선추적 기반 키오스크	
	팀 명	Eye can do it!	
	Eye can do it!	Version 1.3	2019-APR-18

2.1.5 키오스크 웹 구현

아이트래킹 키오스크가 실생활에서도 충분히 문제없이 사용될 수 있음을 보여주기 위하여 데모환경을 최대한 실제상황과 유사하게 구현해야한다.

현재 사회 곳곳에서 사용되는 대부분의 키오스크들은 우리가 익숙한 윈도우 환경에서 운영되고 있으며, 이러한 실제 키오스크와 최대한 유사하게 구현하기 위하여 실시간으로 고객들과 주문정보를 빠르게 주고 받을 수 있는 웹서버를 사용한다.

키오스크 데모는 일상에서 흔히 볼 수 있는 카페의 커피주문 키오스크를 모델로 했으며, 실제 키오스크 처럼 뜨거운 음료, 차가운 음료, 기타 음료 등을 고를 수 있으며 수량도 조정할 수 있게 구현하였다. 또한 물품을 선택하고 장바구니에 담았다가 일괄 주문하는 식의 방법을 선택하였다. 이후 사용자가 주문을 완료하면 주문정보가 서버로 넘어가고 DB에서 주문 정보를 확인 후 판매자가 제품을 제공하는 식으로 구현한다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	장애인용 시선추적 기반 키오스크	
	팀 명	Eye can do it!	
	Eye can do it!	Version 1.3	2019-APR-18


2.2 수행내용

2.2.1 시선 추적 기술

현재 OpenCV를 사용하여 웹캠으로 사용자의 실시간 영상을 입력 받으며 각 프레임마다 이미지를 전달해준다. 전달 받은 이미지는 Dlib에서 학습된 랜드마크 모델 데이터를 사용하여 얼굴 및 눈, 동공영역을 검출하였다. 검출된 눈 영역과 동공의 중심을 각각 선과 점으로 표시해주었으며, 동공의 중심을 좌표로 출력하였다. Pymouse와 PyAutoGUI를 라이브러리 사용하여 마우스를 움직이거나 특정 좌표를 클릭할 수 있다. 현재 포인터를 이동시킬 수 있지만 그 움직임이 부자연스럽다. 추후 떨림을 방지하는 알고리즘 혹은 필터를 사용하여 정확도를 개선시킬 예정이다.

2.2.2 Web Front-end


Frontend 구현의 시간 단축을 위하여 HTML, CSS, JavaScript 오픈소스 템플릿을 사용하였다. 또한 웹 디자인 쪽에 신경을 더 쓰기 위하여 부트스트랩을 사용하여 기존의 구현되어있는 키오스크와 비슷하게 디자인하였다. 상품페이지와 구매정보 페이지를 따로 구현할 계획이었으나 키오스크 특성상 큰화면에서 실행되기 때문에 하나의 페이지로 작성하였다. 조작의 용이성을 위해 앵커를 활용하여 상품 섹션과 주문정보 섹션의 이동을 용이하게 하였고, 포인터의 정확도를 고려하여 상품 및 제어를 담당하는 부분을 크게 디자인하였다..

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	장애인용 시선추적 기반 키오스크	
	팀 명	Eye can do it!	
	Eye can do it!	Version 1.3	2019-APR-18

3 수정된 연구내용 및 추진 방향

3.1 수정사항

- 시선추적을 하기 위해 오픈소스 라이브러리인 Pygaze를 사용할 예정이었으나 라이브러리에 대한 설명 부족과 설치 및 사용이 까다로워 보다 자료의 양도 많고 습득하기 쉬운 OpenCV와 Dlib을 사용하는 방향으로 수정하였다.
- 기존에는 새로운 키오스크 시스템을 만드려고 생각하였으나 보다 많은 곳에서 사용될 수 있도록 키오스크에 소프트웨어와 웹캠만 설치하면 시선추적을 할 수 있도록 add-on 형식으로 만들 예정이며, 키오스크가 도입되어 있지 않은 곳에서 새롭게 도입할 수 있도록 키오스크 UI나, 데이터베이스, 서버까지 제작하여 하나의 시스템으로도 사용할 수 있게 만들 예정이다.
- 장애인을 위한 키오스크인 만큼 단순히 키오스크 환경을 조성하기 보다는 장애인이 용이하게 사용하기 위해서 메뉴 선택이나 색상 등을 조작하기 쉽고 직관적인 UI를 만들기로 하였다.

 <div> <p>국민대학교 컴퓨터공학부 캡스톤 디자인 I</p> </div>	중간보고서		
	프로젝트 명	장애인용 시선추적 기반 키오스크	
	팀 명	Eye can do it!	
	Eye can do it!	Version 1.3	2019-APR-18

4 향후 추진계획

4.1 향후 계획의 세부 내용

4.1.1 시선 추적 기술

현재 해당 영역 검출 및 표시, 좌표를 출력하는 단계까지는 마무리 되어 있다. 중간발표 이후에는 본격적으로 실제로 사용할 수 있을 정도로 시선을 추적하여 특정 위치에 포인터를 이동시키고 클릭을 할 수 있는 기능까지 구현 할 예정이다. 또한 정확한 위치로 포인터를 이동시키기 위하여 비례식이나 방정식을 구하여 시선의 움직임에 따라 포인터를 이동시킬 수 있도록 할 것이다. 또한 동공의 떨림을 보완하기 위하여 low pass filter처럼 주파수 영역에서의 범주 이내의 주파수만 통과시키는 방법 혹은 손떨림 방지 기술 등 여러 분야에서 떨림이나 특정 값을 제외하는 방식을 도입하여 가장 적합한 방법을 찾을 계획이다.

4.1.2 Web Front-end

상품 목록

번호	✓	상품명(옵션)	판매가	회원 할인	수량	주문금액 (적립 예정)	주문관리
1	✓	 <div> <p>【유니스디자인】퍼펙트 핏 크롭 슬렉스.. 차콜</p> <p>옵션: L / 재고 5개 이상</p> <p>* 적립금 전환인 미적용 상품(적립만 가능)</p> </div>	59,000 47,200	-940	1 + - 수정	46,260 (1,416)	삭제하기

품종식제


선택삭제

[그림 2] 장바구니 시안

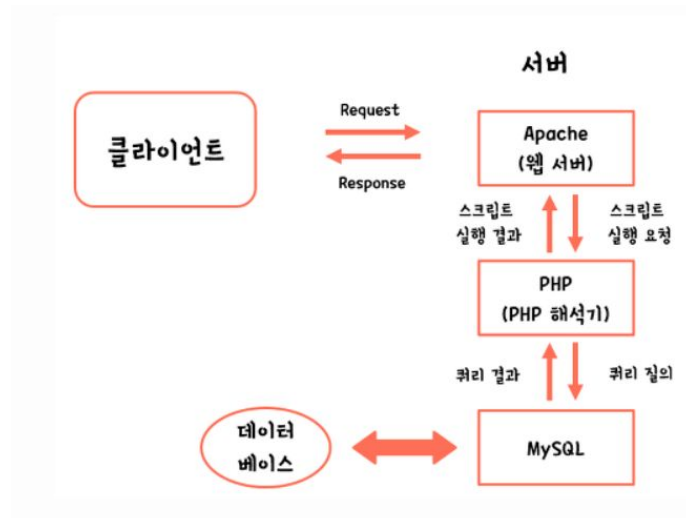
(출처 : <https://store.musinsa.com/app/cart>)

장바구니 환경 구현과 UI 및 미흡한 디자인을 보충하는 것으로 프론트엔드 구현을 마무리할 예정이다. 장바구니는 상품명, 수량, 가격을 담고 주문하면 테이블 아래쪽에 계속 스택되는 형태로 만들 것이고 가장 하단에 주문하기 버튼으로 서버와 통신할 예정이다.

단, 아이트래킹의 정밀도 향상을 위하여 모든 버튼은 크게 구현할 예정이다.

 <div> 국민대학교 컴퓨터공학부 캡스톤 디자인 I </div>	중간보고서		
	프로젝트 명	장애인용 시선추적 기반 키오스크	
	팀 명	Eye can do it!	
	Eye can do it!	Version 1.3	2019-APR-18

4.1.2 Web Back-end




[그림 3] APM 운영 방식

(출처 : <https://wodonggun.github.io>)

사용자가 원하는 제품을 장바구니에 담고 일괄주문하면 웹서버(Apache)는 PHP에게 스크립트 실행을 요청하고, PHP는 미리 작성된 프로그램을 통해 MySQL에 쿼리를 질의하여 DB에 저장된 제품정보를 가져와 관리자의 Client에 다시 전송한다. 관리자는 주문정보를 확인하고 제품을 사용자에게 제공한다.

관리자가 주문정보를 확인할 때, 주문정보를 확인하는 페이지에서 비동기식 연동인 Ajax를 활용하여 관리자는 리다이렉트없이 실시간으로 주문을 확인하게 한다. 주문정보를 확인하는 페이지에서는 주문시각도 출력하여 어디까지가 주문의 끝인지 알 수 있게 하고 더 나아가서 어떤 음료가 얼마나 판매가 되었는지 또한 추가적으로 구현할 계획이다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	장애인용 시선추적 기반 키오스크	
	팀 명	Eye can do it!	
	Eye can do it!	Version 1.3	2019-APR-18

5 고충 및 건의사항

- 동공의 떨림이나 시선 추적의 정확도를 높이기 위해 동공이 어떤식으로 인식되는지 기본적인 원리를 다시 확인하고, 다른 분야에서 특정 떨림현상을 방지하기 위해 사용되는 기술을 찾아보고 그 중에서 현재 진행하고 있는 프로젝트와 적합한 방식을 찾아 도입하려고 한다.
- 초반에 Eye tracking 시스템을 오로지 파이썬 오픈소스 라이브러리인 Pygaze로만 해결하려 했으나 Pygaze에 대한 정보가 너무 부족할 뿐만아니라 설치 및 사용이 까다로워서, 보다 보편화되어있고 자료 습득이 용이한 OpenCV와 Dlib을 사용하기로 방향을 수정하였는데 이 과정에서 Pygaze에 대한 정보를 획득하려고 많은 시간을 소요한 것 같다.