

컴퓨터 비전을 활용한 공부 관리 플랫폼 개발
착수 보고서



뉴뉴뉴갈매기
201924480 박준형
201924431 김상유
201924530 이상준

목차

1. 연구 배경 및 목표
 - a. 연구 배경
 - b. 과제 목표
2. 요구조건 분석
3. 연구 방향
 - a. Eye tracking
 - b. YOLO
 - c. GPT API
 - d. 전체 구성도
 - e. 사용 기술
4. 현실적 제약 사항
5. 개발 일정 및 역할 분담
 - a. 개발 일정
 - b. 역할 분담
6. 참고문헌

1. 연구 배경 및 목표

a. 연구 배경

최근 공부에 집중해야하는 학생들은 유튜브, 틱톡 등의 디지털 매체로 쉽게 산만해지며 공부에 집중하기 어려워하는 문제가 두드러지고 있다. 기존의 공부 타이머는 강제성 부족으로 인해 효과가 떨어지는 문제점이 존재한다. 예시로 2018년 11월 11일 (주) 팔로에서 출시된 “열정품은타이머”는 자신의 공부 시간을 측정하고 이를 서로 공유함으로써 공부하고 있는 사람들을 보고 자극을 받아 공부할 수 있지만, 집계되는 시간동안 사용자가 실제로 공부에 집중하고 있는지 여부 판단은 실행하고 있지 않기 때문에 타이머를 켜놓고 자거나 놀며 공부시간을 말도 안 되게 늘린다는 단점이 존재한다. 이에 컴퓨터 비전 기술을 **Eye tracking**을 활용하여 사용자의 시선을 파악하고, 이를 통해 현재 공부에 집중하고 있는지 여부를 판단하여 타이머를 진행함으로써 사용자의 효율적인 공부 집중을 유도하고 동기부여를 촉진할 수 있도록 돕기 위해 해당 프로젝트를 제안한다.

b. 과제 목표

우리는 웹캠 화면을 이용하여 실시간으로 사용자 시선 추적을 통해 화면의 어느 부분을 보고 있는지 알아내는 아이트래커를 연구하고 성능을 향상 시킬 것이다. 이어서 사용자가 보고 있는 화면이 학업과 관련된 내용인지 분석하여 공부화면인지 판단하는 모델을 개발할 것이다. 마지막으로 이러한 모든 내용을 웹 서비스로 구현하여 편리하게 이용할 수 있도록 할 것이다. 더 나아가 사용자의 공부 습관과 학업 성취도를 평가하여 학업 방법 피드백을 주는 것이 최종 목표이다.

2. 요구조건 분석 및 설계

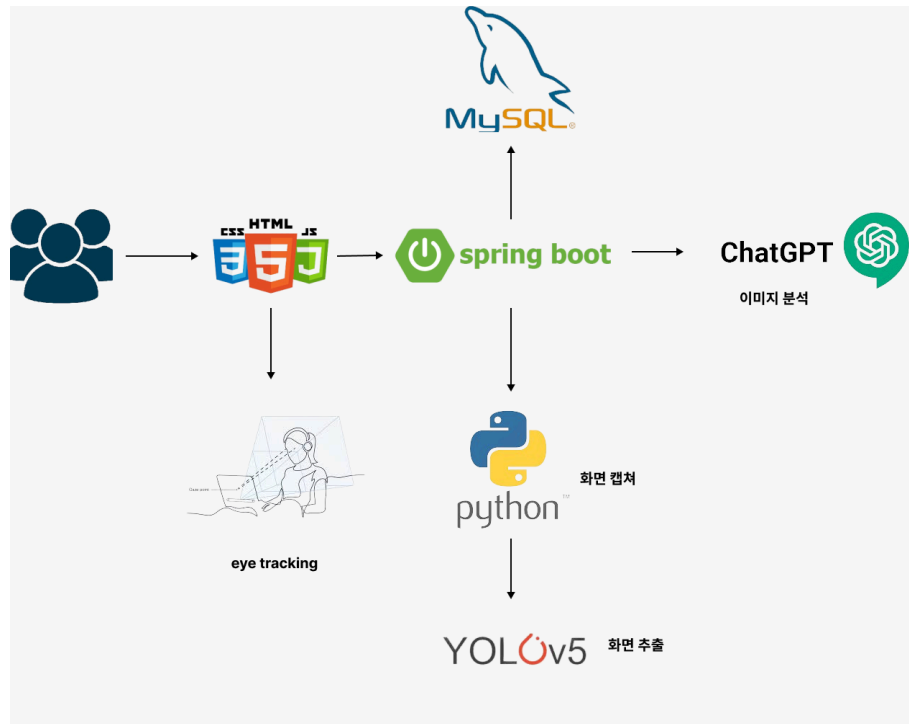
a. 조건 분석

- i. 활용할 사용자의 데이터, 웹캠, 사용자의 화면 정보
- ii. **face-landmark-detection**으로 사용할 모델
- iii. 화면 분할 모델, YOLO

b. 개발 환경

- i. Python
- ii. Spring boot
- iii. HTML, CSS, JS
- iv. mySQL
- v. AWS
- vi. Docker

c. 전체 구상도



1. 프론트엔드 JS에서 웹캠을 통한 eye tracking으로 화면 속 사용자의 시선 좌표를 추출한다.
2. 사용자의 시선 좌표를 서버로 보낸다.
3. python 프로그램으로 사용자의 모니터 전체 화면을 캡처한다.
4. YOLOV5를 이용해 사용자의 화면 분할을 추출한다.
5. 사용자의 시선 좌표가 어느 화면에 속해 있는지 판단한다.
6. 5번에서 추출한 화면을 ChatGPT API를 이용해서 공부와 관련된 내용인지 판단한다.

3. 연구 방향

a. eye tracking

하드웨어 아이트래커는 모든 조명 조건에서 눈 움직임을 포착하도록 설계 및 최적화된 전용 카메라 하드웨어이며 머리 움직임과 눈 영역의 광범위한 생리적 변화를 측정할 수 있다.

웹캠 기반 시선 추적은 각막 반사를 캡처하기 위해 적외선 다이오드를 사용하지 않으며 소위 말하는 순전히 비디오 기반 시선 추적이다. 이는 가시 스펙트럼의 빛을 감지하는 일반 카메라로부터 시선 추적 데이터를 얻는다는 것을 의미한다.

eye tracking은 아래의 과정을 통해 작동한다.

i. 얼굴 위치 감지

facd-landmarks-detection 모델을 활용하여 얼굴의 랜드마크 위치 정보를 파악한다.

ii. 얼굴 내에서 눈의 위치 감지

얼굴 부위의 랜드마크 정보를 이용하여 눈의 위치를 얼굴 좌표에 투영시킨다.

iii. 왼쪽, 오른쪽 눈의 방향 감지

동공이 쳐다보는 방향을 알아내어 바라보는 방향을 알아낸다.

iv. 눈의 방향을 화면 좌표계에 매핑

b. YOLO

i. YOLO(You Only Look Once)는 객체 감지(Object Detection) 및 분류를 수행하는 딥러닝 알고리즘 중 하나이다. YOLO는 이미지나 비디오에 포함된 여러 객체를 식별하고, 객체의 경계 상자(bounding box)와 클래스 레이블을 예측한다. YOLO의 특징은 다음과 같다.

1. 실시간 객체 감지: YOLO는 실시간으로 객체를 감지할 수 있는 속도를 가지고 있다. 이는 한 번의 순방향 전파(forward pass)로 이미지 전체를 처리하고 모든 객체를 감지할 수 있기 때문에 가능하다.
2. 한 번의 추론으로 감지 및 분류: YOLO는 이미지를 그리드로 나누고, 각 그리드 셀에서 여러 개의 바운딩 박스와 해당 객체에 대한 확률을 예측한다. 따라서 객체 감지와 분류를 동시에 수행한다.
3. 다양한 객체 탐지: YOLO는 다양한 클래스의 객체를 탐지할 수 있다. COCO 데이터셋에서 사용되는 80가지 클래스에 대해 미리 훈련된 YOLO 모델을 사용할 수 있다.
4. 단일 네트워크 구조: YOLO는 단일 신경망 네트워크 구조로 이루어져 있다. 이는 다른 객체 감지 알고리즘에 비해 구현이 간단하고 효율적이라는 장점을 제공한다.
5. 객체 위치 정보: YOLO는 바운딩 박스의 중심 좌표와 크기를 예측한다. 이는 객체의 위치와 크기를 정확하게 파악할 수 있도록 도와준다.

c. GPT API

i. GPT (Generative Pre-trained Transformer) API는 OpenAI에서 제공하는 인공지능 기반의 텍스트 생성 및 처리 서비스이다. 이 API를 통해 개발자들은 GPT 모델을 활용하여 다양한 언어 처리 작업을 수행할 수 있다. GPT 모델은 대규모의 언어 데이터를 사전 학습하여 자연스러운 언어 이해 및 생성 능력을 갖추고 있으며, 이를 API 형태로 제공함으로써 사용자가 쉽게 접근하고 활용할 수 있도록 한다.

1. 텍스트 생성: 사용자가 입력한 텍스트에 기반하여 관련성 높고 자연스러운 텍스트를 생성한다. 이를 통해 기사 작성, 스토리 텔링, 자동 답변 생성 등에 활용할 수 있다.
2. 언어 이해: 질문에 대한 답변을 제공하거나 텍스트 요약, 감정 분석 등의 작업을 수행한다.
3. 번역: 다양한 언어 간의 텍스트 번역을 지원한다.
4. 코드 생성 및 수정: 프로그래밍 언어에 대한 이해를 바탕으로 코드 조각을 생성하거나 기존 코드를 수정하는 데 도움을 준다.

4. 제약사항 (추가)

a. 웹캠 해상도에 따른 눈 영역 픽셀 수 부족 문제

최신 웹캠은 높은 해상도를 지원하는 경우가 대부분이지만 대기 시간과 프레임의 균형을 이루어야 합니다. 해상도가 낮으면 실시간성을 보장할 수 있지만 눈 영역을 나타내는 픽셀 수가 적어지는 문제가 발생합니다. 픽셀 수가 적으면 시선 추적에 있어서 필요한 정보가 부족하고 정확도와 신뢰성이 낮아진다는 문제가 발생합니다.

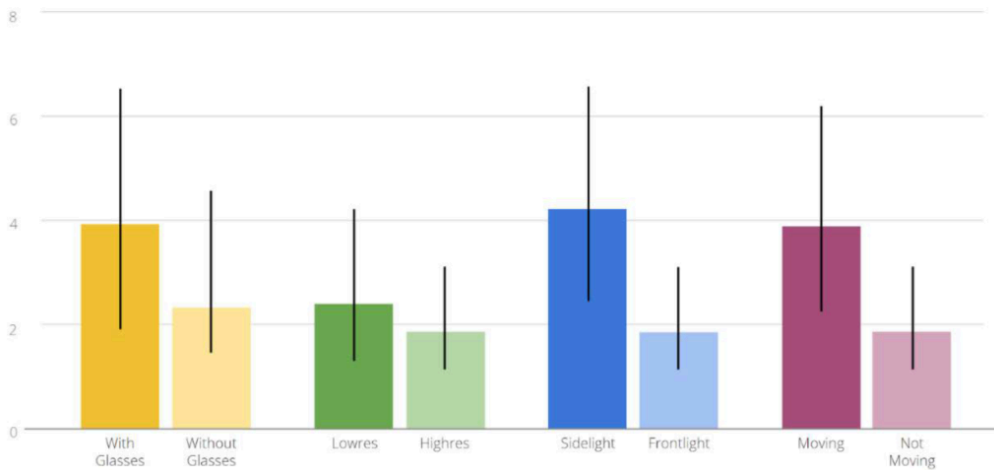
b. 어두운 조명 조건에서의 정확도 문제

웹캠 기반 시선 추적은 전용 하드웨어 장치와 다르게 적외선을 사용하지 않고 가시광선 영역의 데이터만으로 시적 추적을 진행합니다. 그러므로 주변 조명 조건에 따라 데이터가 변경되는 문제가 발생합니다. 일반적으로 대비가 낮을 경우 얼굴 배경에 대한 눈 움직임을 감지하는 정도가 약해 정확도가 낮아질 수 있습니다.

c. 안경, 얼굴 각도, 움직임 등 외부 조건에 따른 정확도 문제

사용자의 안경 등의 착용 물품, 카메라 각도, 시선 추적 도중 머리의 움직임 또한 얼굴 영역으로부터 눈 영역을 추출하는 모델의 성능을 떨어뜨릴 수 있는 주요 원인으로 볼 수 있습니다. 이러한 외부 환경이 적은 이상적인 조건에서 모델의 예측이 정확하게 수행된다는 한계점이 존재합니다.

Median accuracy (dva) and interquartile ranges for test conditions and their control



5. 개발 일정 및 역할 분담

a. 개발 일정

개발구분	세부항목	5	6	7	8	9	10
기획	주제 선정 및 고도화						
	사전 조사						
인공지능 모델 개발	Eyetracking 구현						
	YOLO 기반 화면 분할 구현						
	GPT API 연동						
DB 설계	DB 설계 및 구축						
REST API 설계	DB와 모델 연동						
	기능 개발 및 배포						
애플리케이션 개발	REST API 연동						
	기능 개발 및 배포						
배포 및 수정	보완사항 수정 및 배포						

b. 역할 분담

이름	담당
박준형	YOLO 기반 화면 분할 REST API 설계 및 구현
김상유	Eyetracking 구현 FE 담당
이상준	GPT API 연동 DB 설계 및 구축
공통	관련 논문 분석 데이터 전처리 모델 테스트 및 오차 분석 보고서 작성

6. 참고문헌

How accurate is Webcam Eye Tracking? -Morten Pedersen, 04/17/2024

<https://imotions.com/blog/learning/product-news/how-accurate-is-webcam-eye-tracking-and-what-is-the-best-way-to-use-it/>

시선추적(eye tracking)이란? varison|March 10th, 2023

<https://varison.co.kr/eye-tracking/introduction-to-eye-tracking/>