

온라인 수강 집중도 감지 시스템

멀티캠퍼스 AI 엔지니어 취업캠프 2회차
2조 이공오공



2050

이공오공

멀티캠퍼스 AI 엔지니어 취업캠프 2회차

목차

01

프로젝트 개요

02

프로젝트 설계

03

개발 내용

04

개발후기 및 느낀점



01

프로젝트 개요

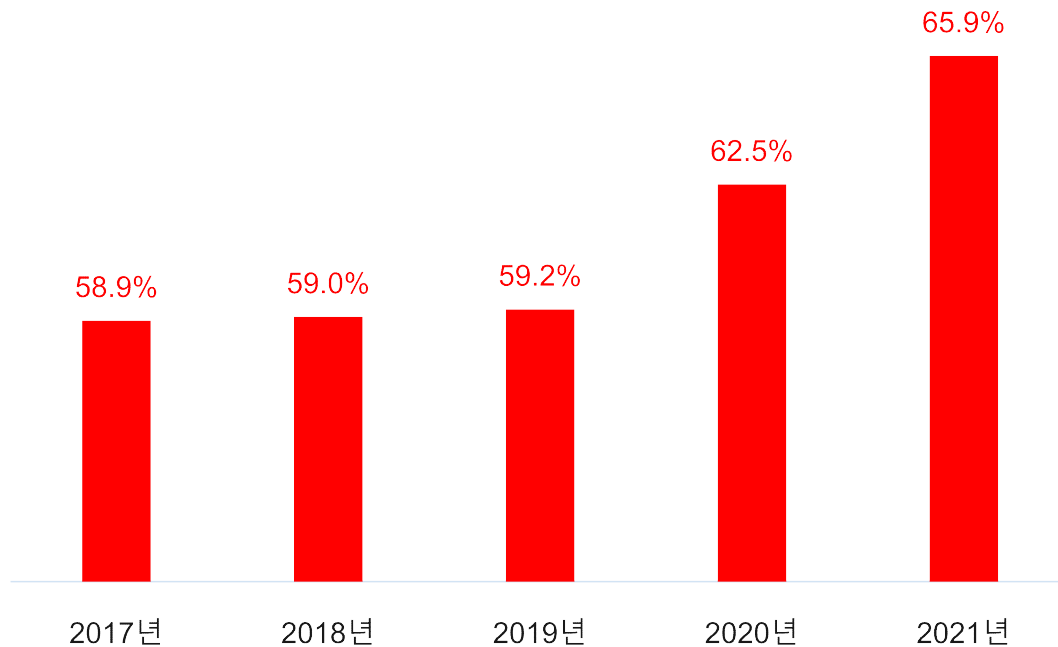
프로젝트 기획 배경 및 목표

구성원 및 역할

강지운 역할 세부사항

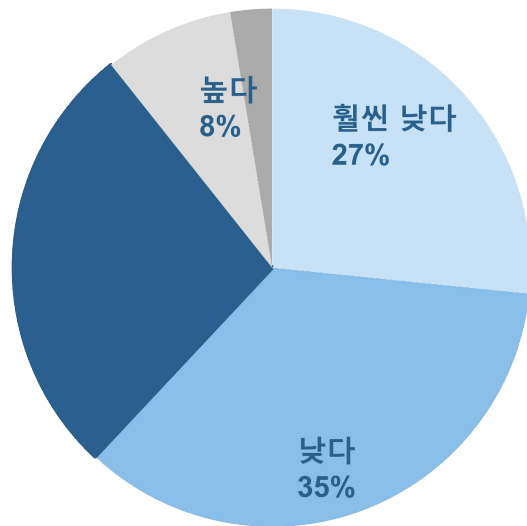
Work-flow

개인 이러닝 이용률



출처: 소프트웨어정책연구소

대면 수업 대비 온라인 강의 집중도

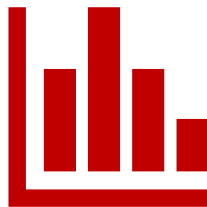


출처: 브런치 블로그 설문조사 (900명)

기획 배경



집중도 하락
온라인 강의 문제점



모니터링 도입
집중력 향상 시스템

프로젝트 목표

집중도 향상

강의 집중도 향상 기대

평가 자료

강의 평가 근거 자료
마련

피드백

강의 개선 피드백으로
활용

자가 평가

수강 과목에 대한 자가 평가
가능



구성원 및 역할



강지윤

모델링



김가현

모델링



김건우

프로젝트 관리



여은정

프론트/백엔드



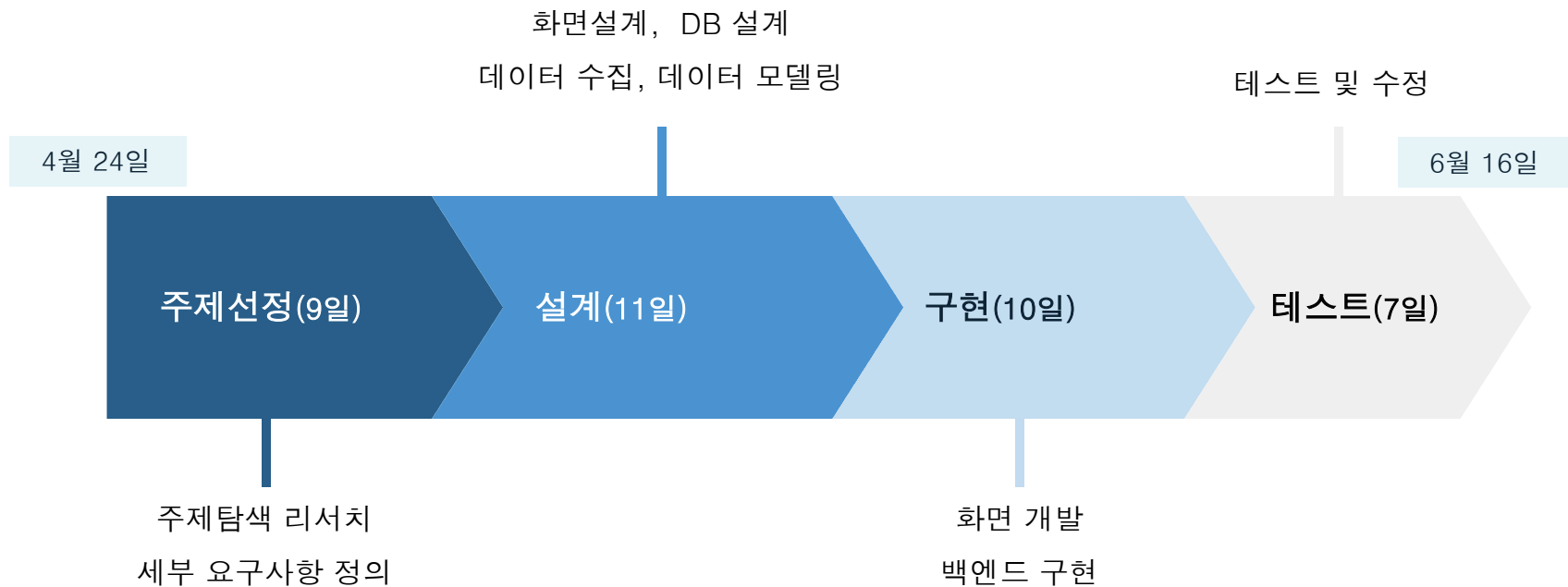
이종현

프론트/백엔드

강지윤 역할 세부사항

	세부 사항
기간	2023.04 ~2023.06
기여도	프론트/백엔드의 30% 참여, AI 모델링 90% 참여
주요업무	1. 강의 수강자의 얼굴을 통해 얼굴 탐지 bounding box와 깨어있는지 졸린지에 대한 classification 예측 2. 만든 모델을 django에 연결 3. 차트를 그릴 웹페이지 제작 Django로부터 데이터를 받아 차트 출력
사용 기술	html css javascript django python tensorflow Yolo opencv

WBS (Work Breakdown Structure)





02

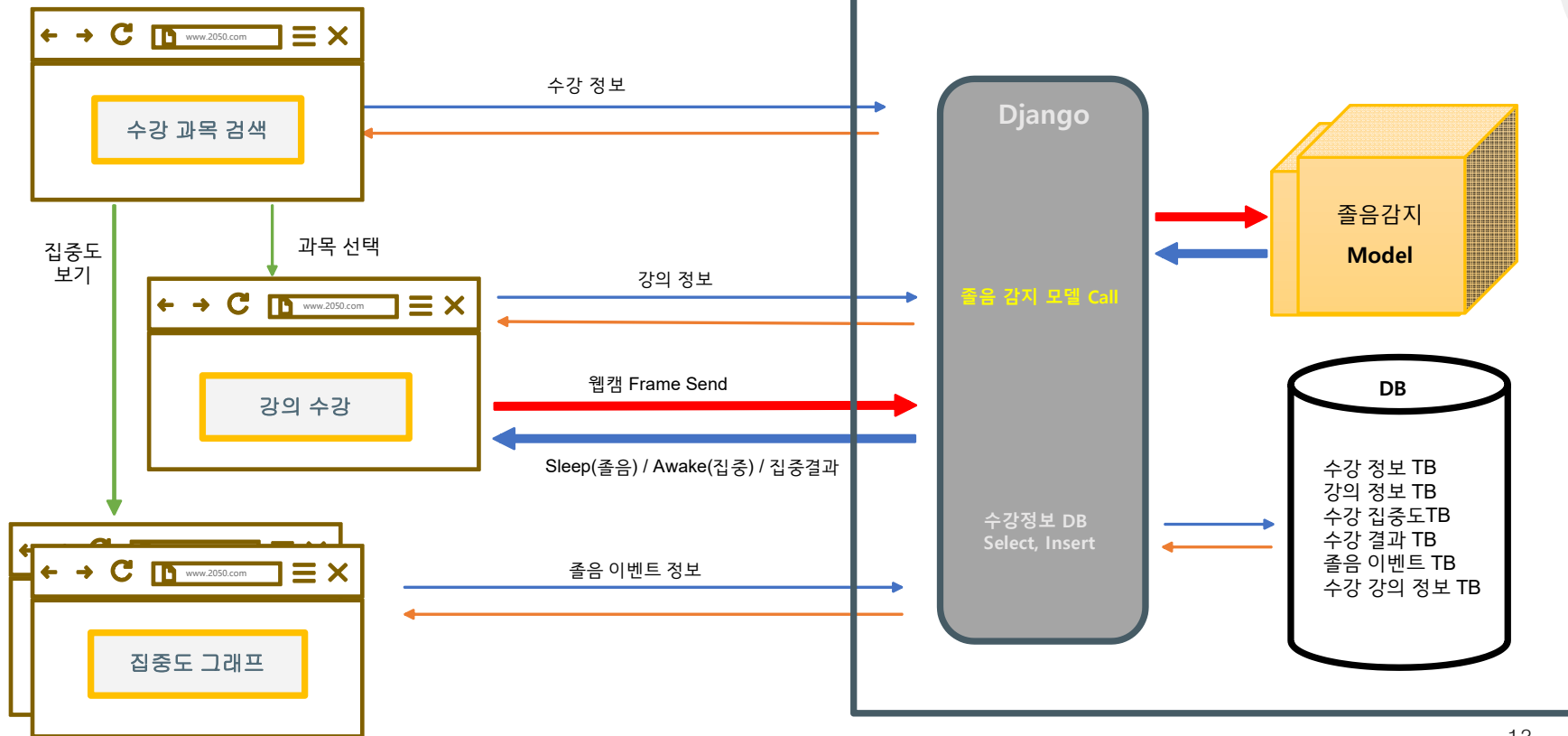
프로젝트 설계

시스템 설계

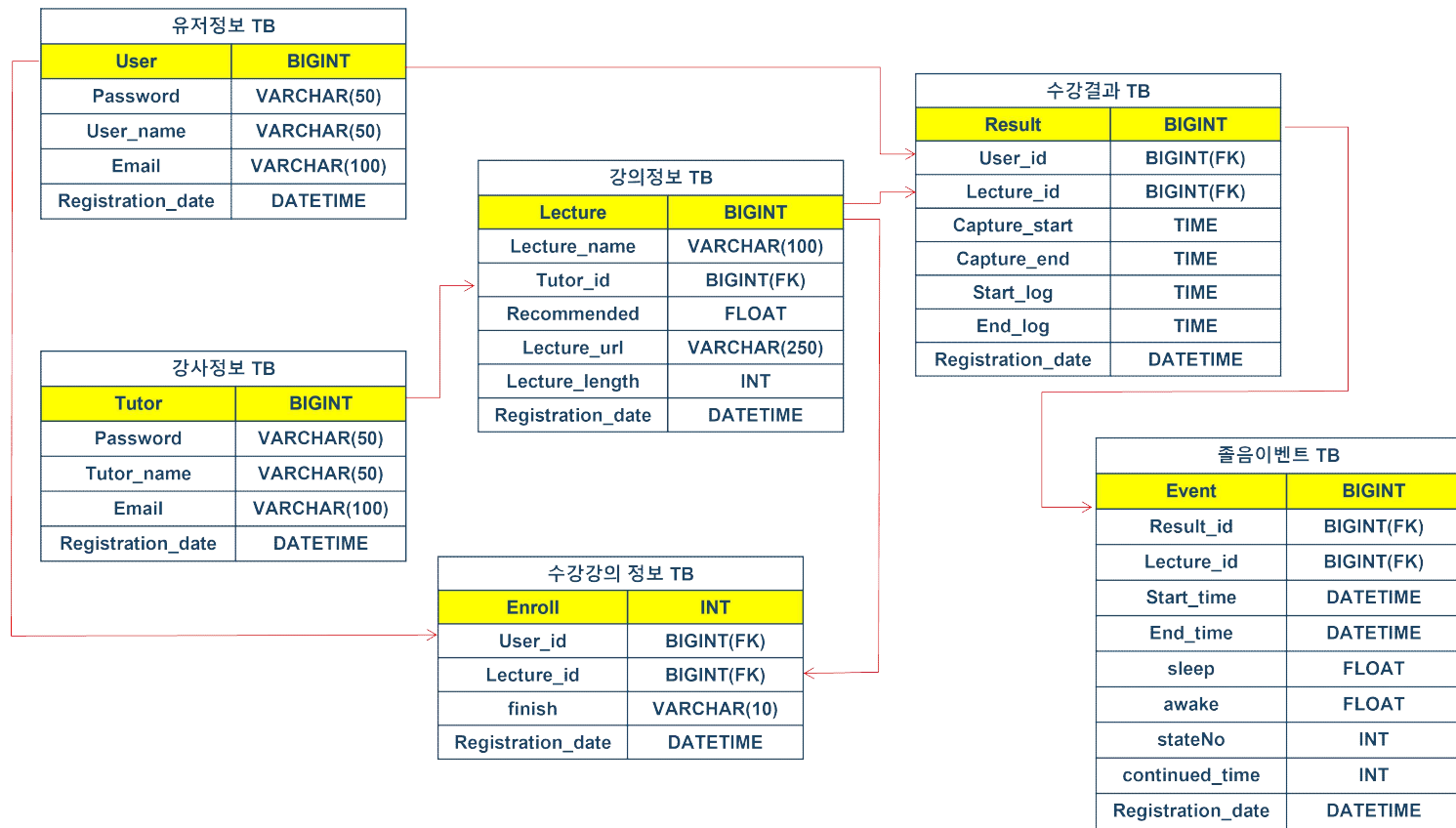
DB 설계

개발일정 및 개발 규칙

전체 시스템 구성도



ERD (DB 관계도)



개발 일정

	5월 2주	5월 3주	5월 4주	5월 5주	6월 1주	6월 2주
AI 모델링	데이터 수집 모델 구축	데이터 수집 모델 학습 모델 성능 비교	모델 선정 성능 개선	최종 모델 선정 django와 연결	테스트 및 수정 보완	발표 자료 정리
Front/Backend	화면 설계서 작성	화면 설계서 수정	시스템 구성도 템플릿 작성	템플릿 기능 구현	수강목록 보여주기 화면 구현 그래프 화면 구현	Html 디자인 적용 그래프 화면 수정
서버	접속 테스트	개발환경 구현	모델링 테스트 django 테스트	환경설정 및 테스트	버전 충돌 이슈 해결	발표자료 정리
DB	DB 설계서 작성	ERD 작성	DB 이식	Data 조회 테스트 Data 저장 테스트	모델 보완	모델 보완



03

개발 내용

적용기술
프론트/백엔드
모델링
시스템 시연

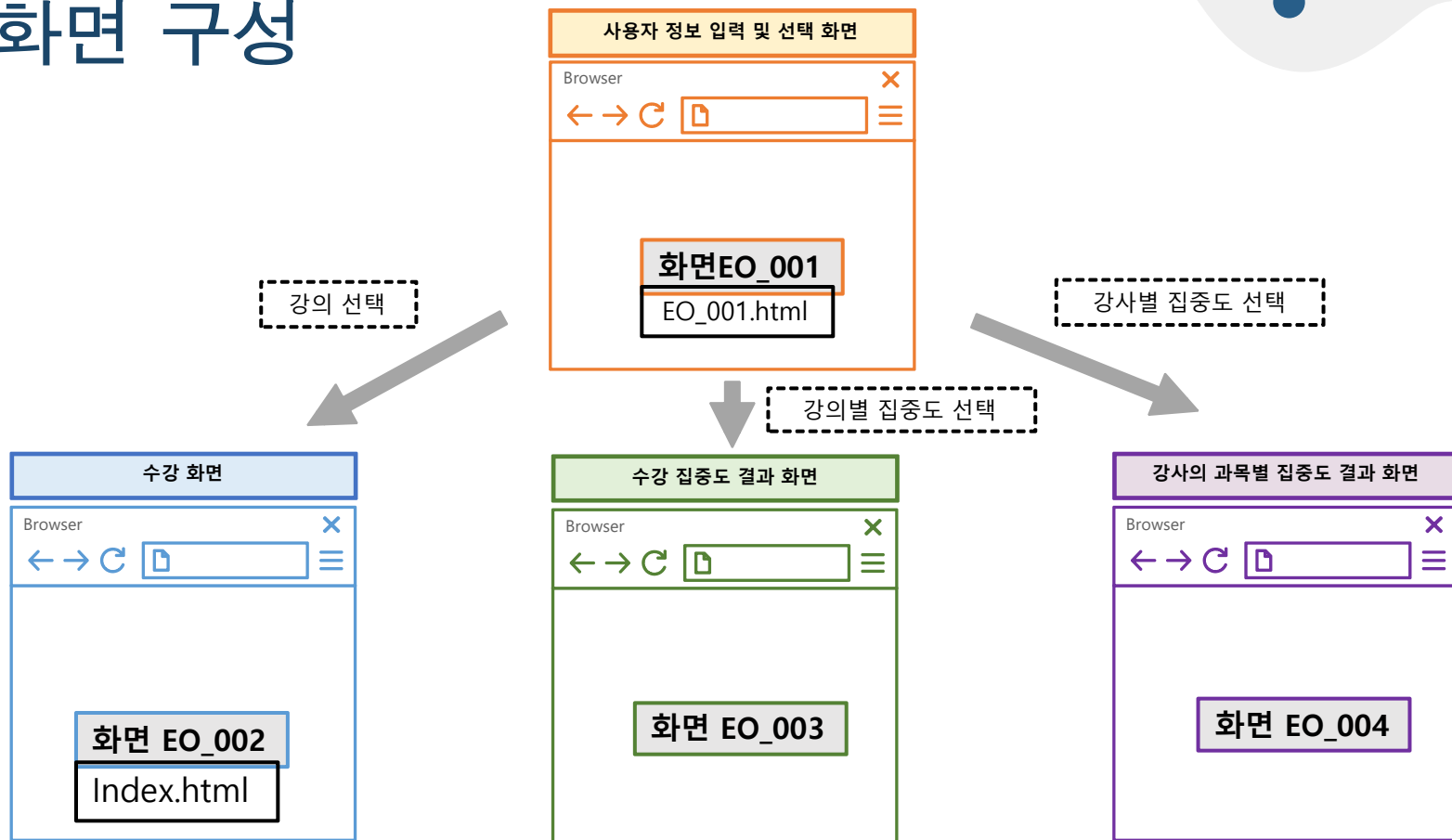
적용 기술

- Python
- Numpy
- Keras
- Matplotlib
- Opencv
- PIL
- dlib
- YOLO Model
- Torch
- Django
- MariaDB
- HTML/CSS
- Javascript

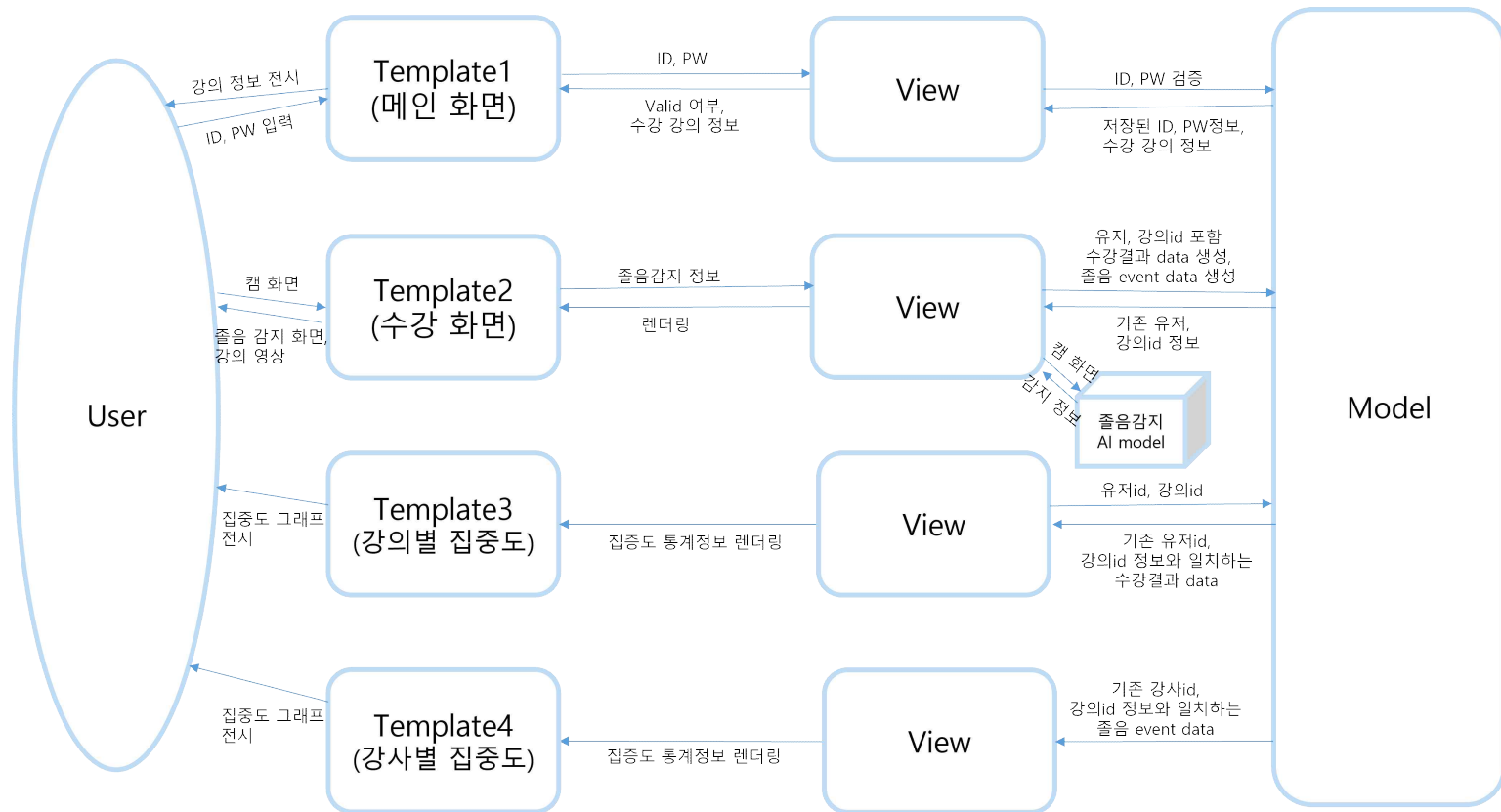
Server

- Python 3.9.16
- bazel 6.2.0, CUDA 11.2, cuDNN 8.1.1
- dlib 19.24.2
- tensorflow 2.11, tensorrt 8.0 GA Update 1
- Ubuntu 18.04

화면 구성



데이터 흐름도



데이터 수집

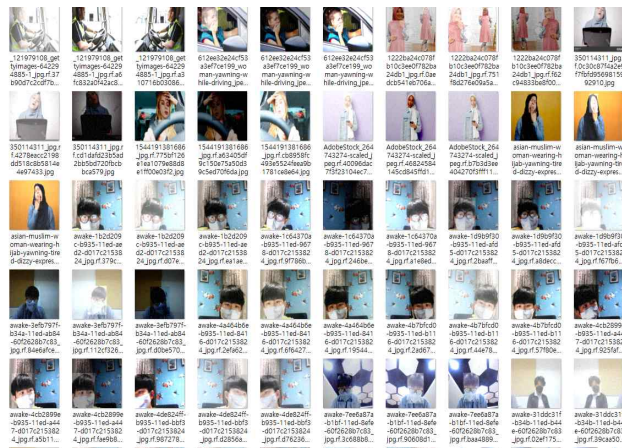
Roboflow & Kaggle

- 웹캠 얼굴 이미지
- YOLOv5 버전 Label

roboflow
kaggle

- 눈 감은 이미지
- 눈 뜬 이미지

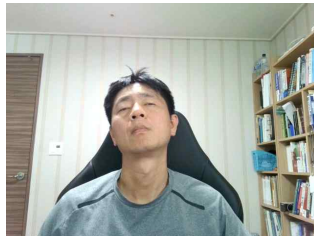
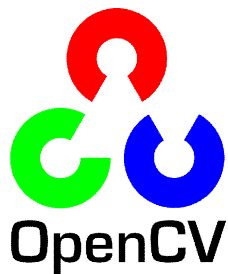
- 딥러닝 데이터셋 제공 사이트를 이용해 데이터 수집



데이터 수집

OpenCV

- 실시간 이미지 프로세싱 라이브러리
- 부족한 데이터 (600장)
- OpenCV로 300장의 이미지 제작

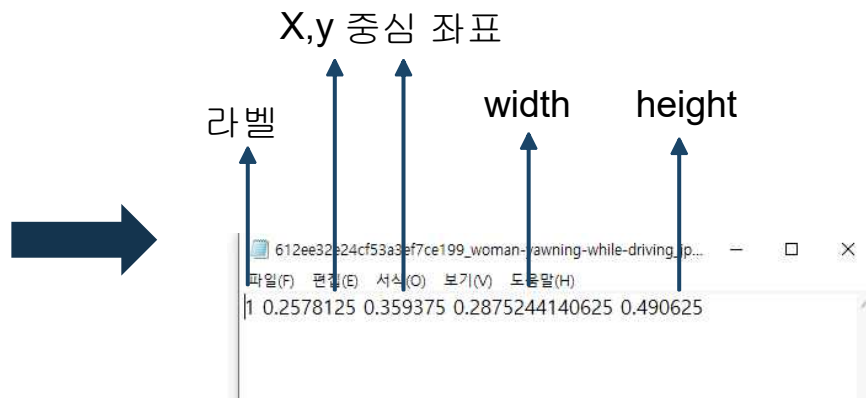
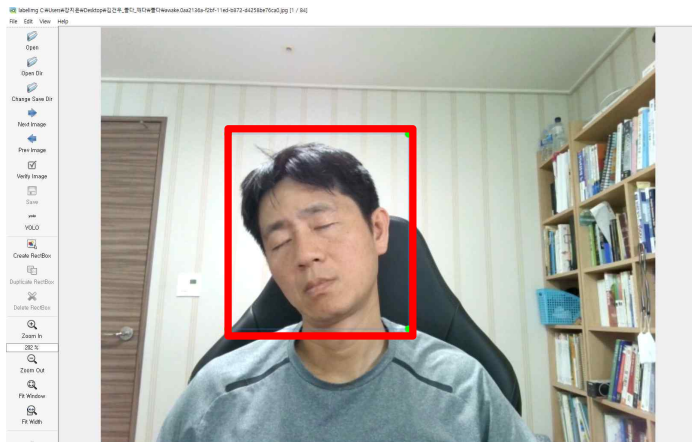


데이터 전처리

Labeling 작업

Tool : Labellmg

- 모델 생성과 학습을 위해 Label 필요
- YOLOv5 pytorch 버전 사용

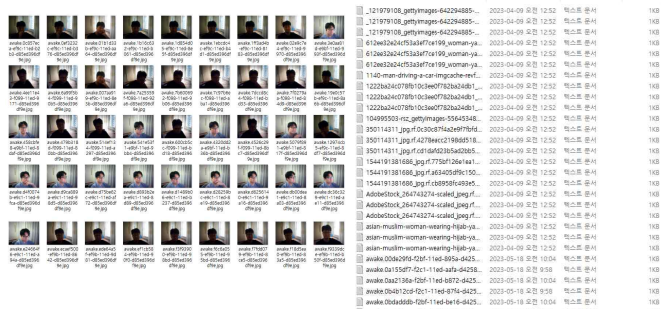


데이터 전처리

용량 축소

Tool : tfrecord

- 900장의 이미지와 Label 파일
- Tensorflow에서 tfrecord 제공
 - 작은 용량의 tfr 파일 제작



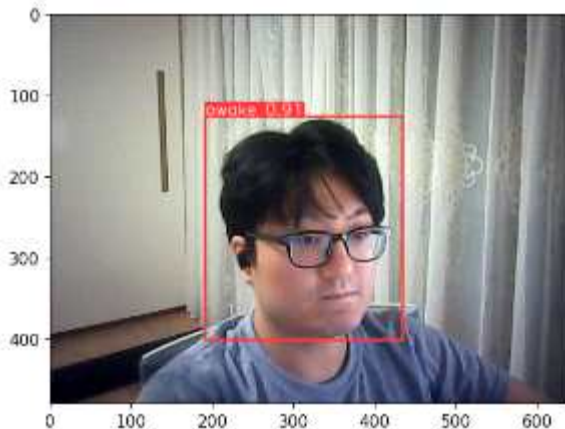
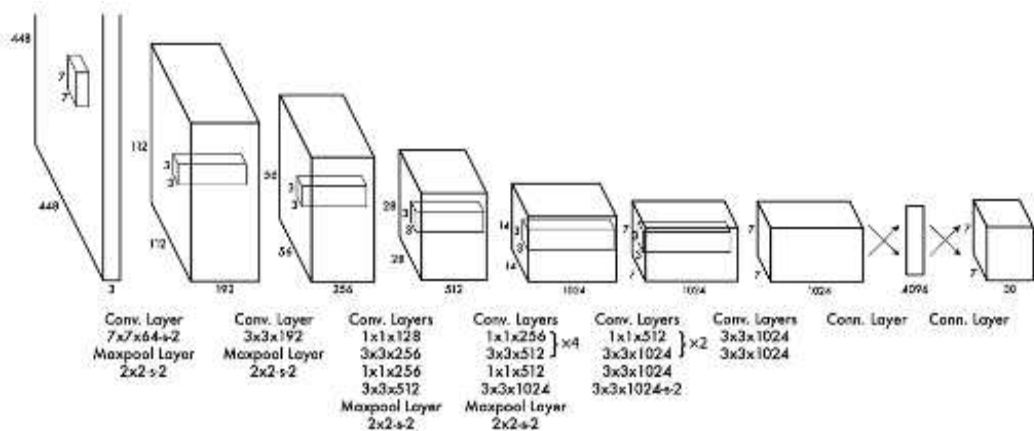
이름	수정된 날짜	유형	크기
Trian.tfr	2023-05-15 오후 4:46	TFR 파일	87,831KB
Val.tfr	2023-05-15 오후 4:46	TFR 파일	22.068KB

모델 생성

사용 모델

Yolo

- 빠르고 보편화된 모델 Yolo (Yolov5 Ultratic) 사용
- Yolov5s의 weight 사용해 학습시간 단축



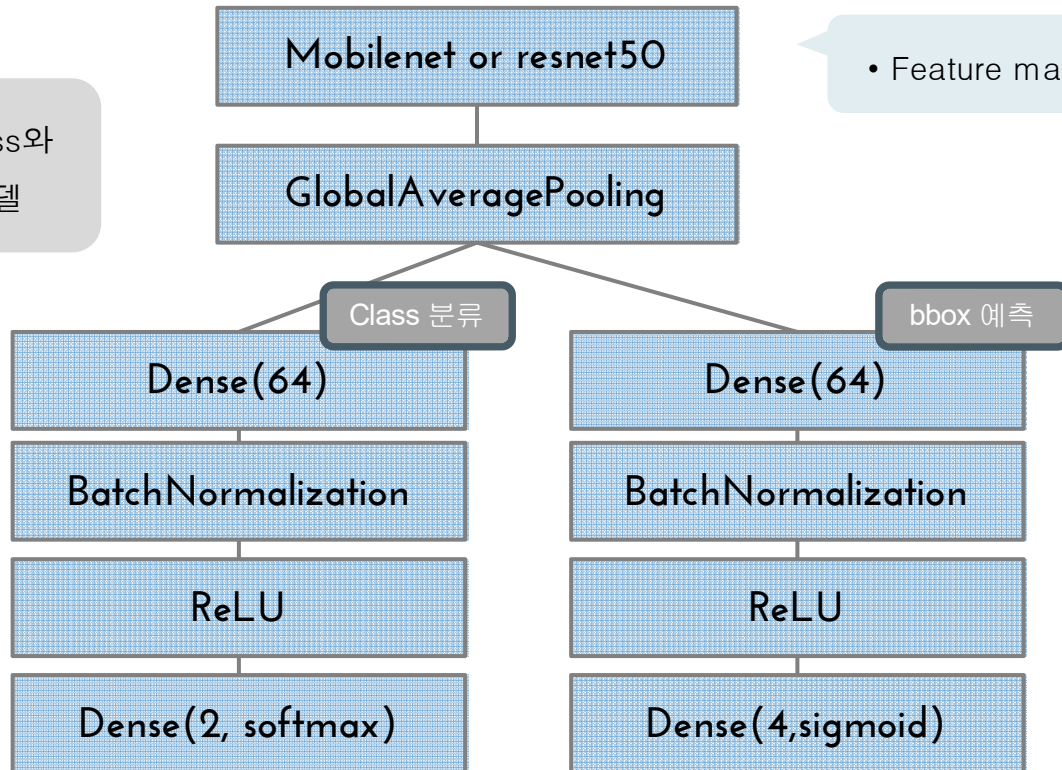
CNN 모델 생성

CNN Model

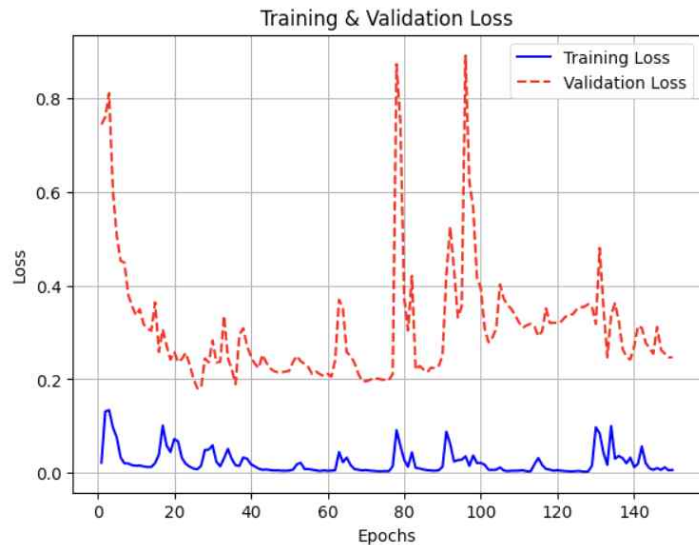
- 이미지(Input)의 class와 bbox 예측하는 모델

- Feature map 추출

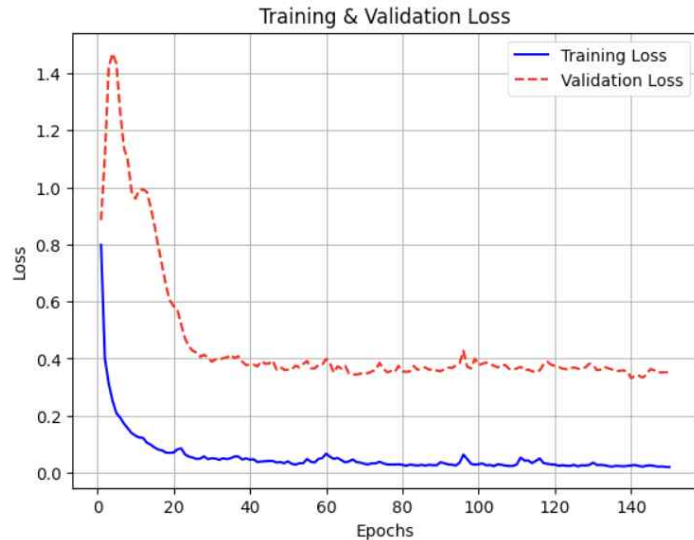
- Overfitting 방지



resnet 모델 결과

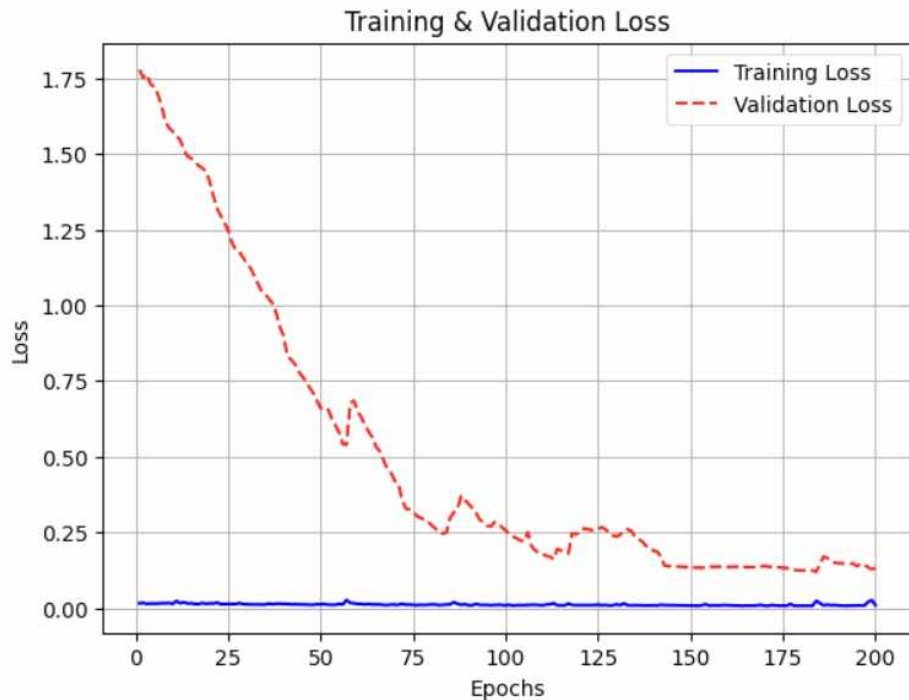


- Learning_rate : $1e-4$
- IOU : 0.699



- Learning_rate : $1e-5$
- IOU : 0.541

Mobilenet 모델 결과



- 최종모델 선택
- 용량이 적고 성능이 좋음
- IOU : 0.75

모델 학습

Loss function

- 모델의 예측 값과 실제 값과의 차이
- Optimizer : Adam

Dense(2, softmax)

Dense(4, sigmoid)

분류
loss_fn

+
x

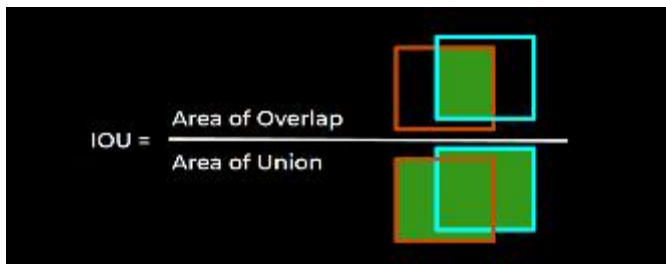
2

bbox의 loss_fn

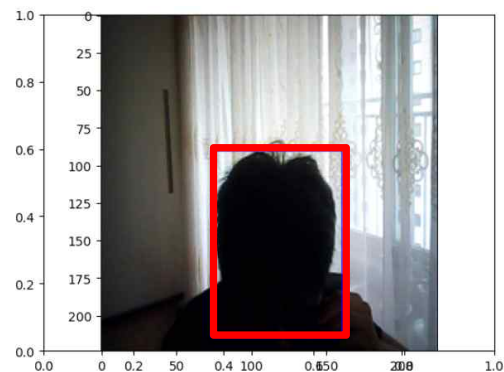
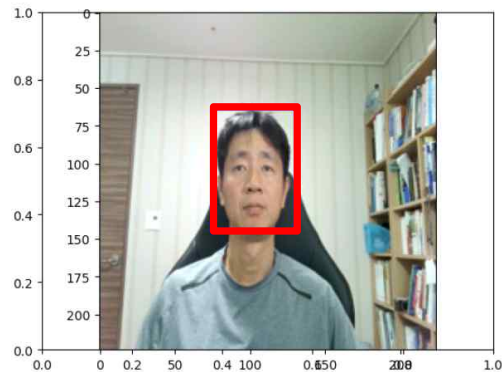
가중치 적용

모델 학습 결과

IOU 측정



- Ground Truth box와 모델 예측 값의 IOU 계산
- 평균 IOU 값 : **0.75**



모델 학습 결과

Accuracy

• Class 분류 Accuracy 측정

• Accuracy : 0.97

```

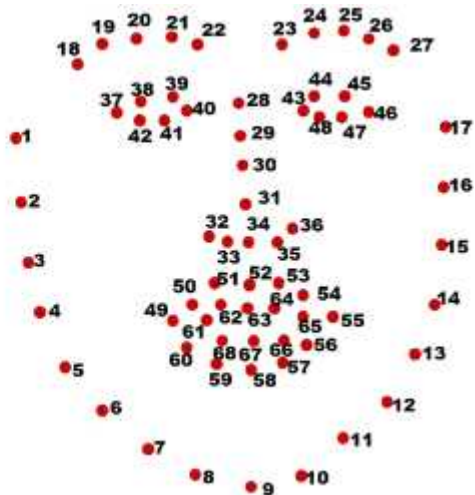
3/3 [=====] - 0s 43ms/step
[0 0 1 1 0 1 0 0 1 1 1 1 1 1 1 0 0 1 1 0 0 1 1 1 1 1 0 0 1 1 0 0 0 1 0 1
 0 0 1 0 1 0 0 0 1 0 1 0 1 1 1 1 1 1 1 0 1 0 1 0 1 1 0 0 1 1 1 0 0 0 1 1
 0 1 1 1 0 0 0 1 0 1 1 0 1 1 1 0]
tf.Tensor(
[0. 0. 1. 1. 0. 1. 0. 0. 1. 1. 1. 1. 1. 1. 1. 0. 0. 1. 1. 0. 0. 1. 1. 1.
 1. 1. 1. 0. 0. 1. 1. 0. 0. 0. 1. 0. 1. 0. 0. 1. 0. 1. 0. 0. 0. 0. 1.
 0. 1. 1. 1. 1. 1. 1. 1. 1. 0. 1. 1. 1. 0. 1. 1. 0. 0. 1. 1. 1. 0. 0. 0.
 1. 1. 0. 1. 1. 1. 0. 0. 0. 1. 0. 1. 1. 0. 1. 1. 1. 0.], shape=(90,), dtype=float32)
0.9777778
3/3 [=====] - 0s 39ms/step
[1 1 0 1 1 1 0 1 1 0 1 1 1 0 0 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 0 0 1 1 1 1 0
 1 1 0 1 0 1 0 0 1 0 0 0 0 1 0 0 0 1 0 1 1 0 1 1 0 1 0 0 0 0 1 1 1 1 0 0 0
 1 1 1 1 1 1 1 0 1 1 0 0 0 1 1 0]
tf.Tensor(
[1. 1. 0. 1. 1. 1. 0. 1. 1. 0. 1. 1. 1. 0. 0. 1. 0. 1. 1. 1. 0. 1. 1. 1.
 1. 1. 1. 0. 1. 0. 0. 1. 1. 1. 1. 1. 0. 0. 1. 0. 1. 0. 1. 0. 0. 1. 0. 0.
 0. 0. 1. 0. 0. 0. 1. 0. 1. 1. 0. 1. 1. 0. 1. 0. 0. 0. 0. 1. 1. 1. 1. 0.
 0. 0. 1. 1. 1. 1. 1. 1. 1. 0. 1. 1. 0. 0. 0. 1. 1. 0.], shape=(90,), dtype=float32)
0.9777778

```

눈 감지 모델

dlib

- 얼굴 탐지 라이브러리
- Landmarks 방식 사용
- 탐지된 얼굴을 68개의 점으로 반환



눈을 정밀하게 감지하기 위해
dlib 사용



EAR 알고리즘을 이용해
눈 상태(open/close) 분류

시연 동영상





04

개발후기 및 느낀점

향후 개선 사항 및 기대효과

향후 프로젝트 보완 계획

느낀점

활용 및 확장 가능성

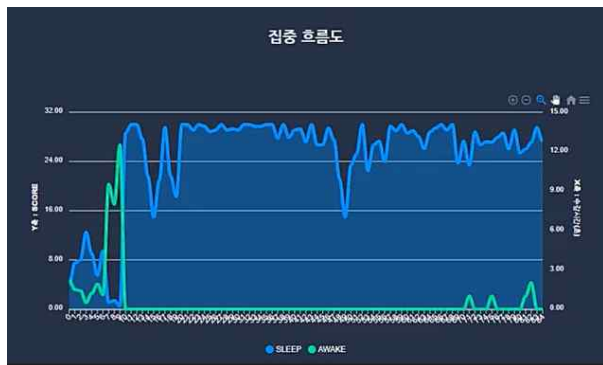


온라인 강의 사이트와 학원에서 수강자들의 집중도 분석 가능

- 정부에서 진행하는 필수 시청 강의에서 집중도에 따른 재수강 여부 판단 가능
- 법 위반자에게 적용 가능



개선 사항 및 기대효과



+

다양한 시각화 방법 적용

집중도 분석 효율성 향상

트래픽 관리

+

실시간 강의

동시 접속 서버 관리하며
신속한 피드백 제공 가능

프로젝트 보완 계획

회원가입

로그인 정보

이메일

이름

아이디 ☒

휴대전화

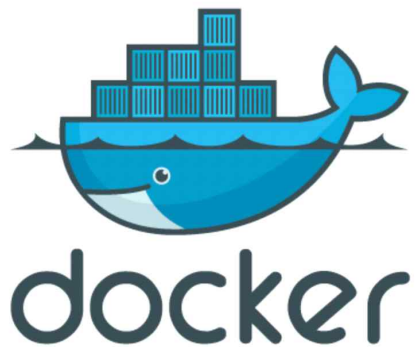
재발송

인증 번호 6자리를 입력해 주세요. 09:51

회원가입 기능 추가



관리자 기능 추가
(회원, 강사, 강의 등록)



도커를 통한 배포

조원 소감

다 만들 수 있어서 다행입니다.

- 강지윤 -

프로젝트 진행중에 어려움도 있었지만 팀원들의 노력으로 완성할 수 있었습니다. 다들 수고하셨습니다.

- 김가현 -

자발적으로 최선을 다해준 팀원들에게 감사하고 부족한 팀장으로서 미안합니다.

- 김건우 -

부족함에 대한 성찰과 지난 경험에 대한 자신감을 다져보는 시간이었습니다. 편견없이 대해준 조원들께 감사

드립니다. - 여은정 -

짧은 시간 내에 배우고 고민해 이 정도의 결과를 낼 수 있었던 팀원들이 자랑스럽습니다.

- 이종현 -

감사합니다