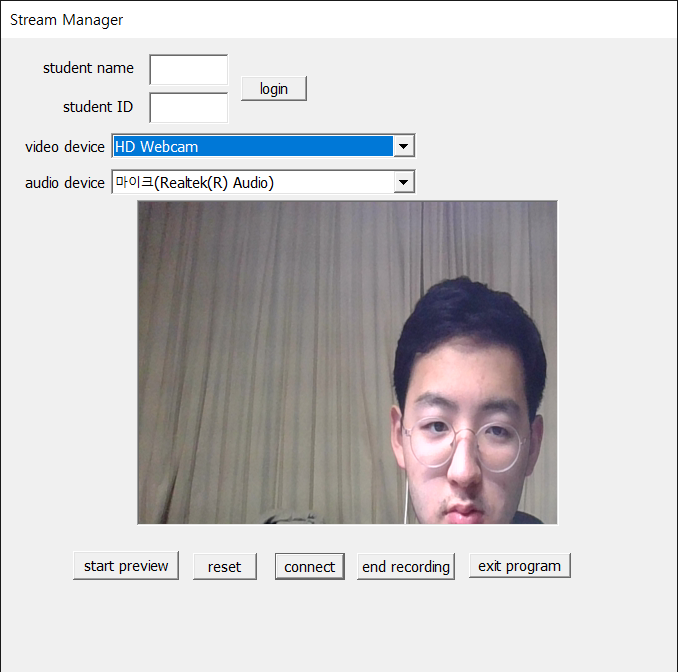
학생 클라이언트 매뉴얼/정리 2021/02/23

황진우 (010-9409-3984)에게 문의하면 됩니다

Part 1 클라이언트



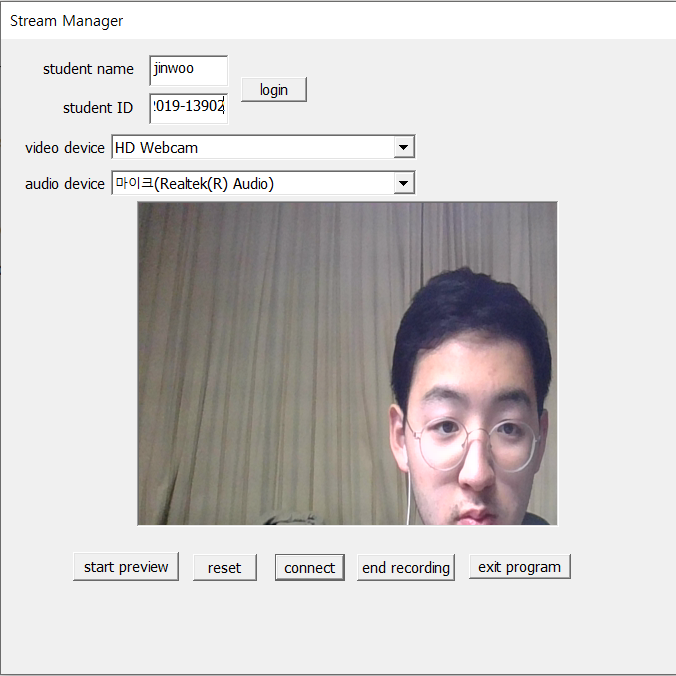
클라이언트는 sln 혹은 exe 파일을 통해서 사용이 가능합니다

전자의 경우에는 그냥 F5나 shift+F5 같은 방법으로 컴파일하면 됩니다

후자는 배포된 버전이 사용되며, 이는 제가 2/23 배포하겠습니다

해당 창에 유저가 사용할 수 있는 기능은 다음과 같습니다.

1. 이름, 학번 입력
2. 해당 정보를 바탕으로 “로그인” (서버에 시험 과목 코드, 테이블 이름을 받아옴)
3. 비디오 디바이스 선택 (webcam, camera 등의 이름이 들어간 기본 웹캠)
4. 오디오 디바이스 선택 (통상적으로는 마이크 등의 이름이 들어간 기본 오디오)
5. 프리뷰 (현재 웹캠에 비춰지는 모습)
6. 프리뷰 시작 버튼 (디바이스를 바꾸는 경우 재시작 및 종료용)
7. 리셋 버튼 (비디오, 오디오 디바이스 설정을 디폴트 (각 list의 1번째 멤버)로 되돌림)
8. 연결 버튼 (서버에 송출을 시작)
9. 녹화 중지 버튼 (시험이 종료되는 시점에 학생 컴퓨터에서 송출된 영상을 저장하고 DB로 넘기기 위해서 녹화 중지를 요청
10. 프로그램 종료 버튼 (프로그램을 종료시킴)

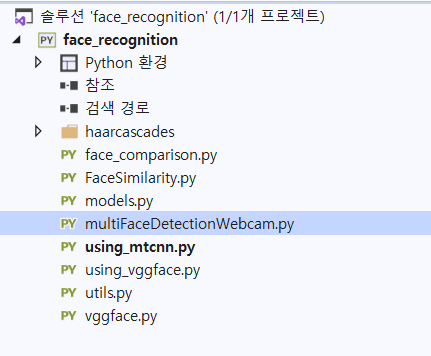


\*\*디바이스를 잘못 설정한 경우🡪start preview 프리뷰 중지🡪디바이스 변경🡪start preview 다시\*\*

\*\*이 경우에 영상이 이상하게 보이면 reset 누르고 디바이스만 다시 선택해주면 됩니다\*\*

\*\*연결 버튼을 송출 중에 누르면 에러 발생해서 송출 중에는 비활성화됩니다\*\*

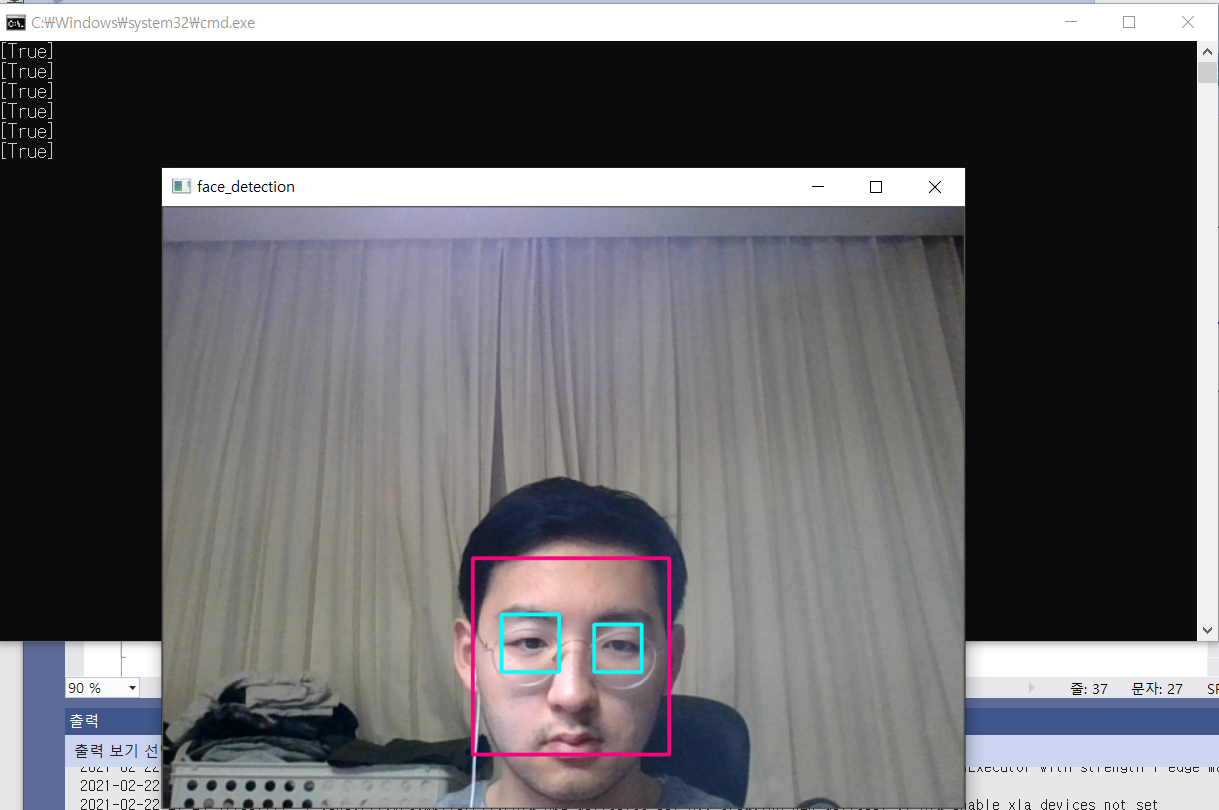
Part 2 얼굴인식



프로젝트를 열면 위와 같습니다

여기서 multiFaceDetectionWebcam.py를 오른쪽 클릭하고, 디버깅 없이 시작 단추를 누릅니다

(해당 프로그램은 현재 openCV로 웹캠을 액세스하기에 클라이언트와 동시 사용이 불가합니다)



위와 같이 얼굴, 그리고 눈을 잡아냅니다

왼쪽 상단에 보이는 건 실제로는 그냥 카운트만 될 True/False 값이지만, 테스트 목적으로 True가 출력됩니다.

\*\*해당 프로그램은 처음 컴파일할 때 시간이 좀 소요됩니다 (20초)\*\*

\*\*해당 프로그램의 threshold는 0.6\*\*

Threshold가 낮을수록 엄격하게 채점하며, cmd에서는 멀티코어를 활용해서 연산을 병렬화하거나 해당 threshold를 변화시키는 게 나와있습니다.

해당 부분은 Python의 내장된 face\_recognition 라이브러리를 활용했습니다.

<https://pypi.org/project/face-recognition/>

현재는 얼굴이 잡히면

1. 얼굴을 둘러싼 붉은 사각형을 사진으로 저장
2. 해당 저장된 사진과 레퍼런스 사진 (학생 데이터베이스에 나와있는 사진 등)을 비교
3. True/false 값을 반환

얼굴이 잡히지 않는다면 그냥 계속 OpenCV 창으로 웹캠 모습이 재생됩니다.

\*\*현재 문제는 얼굴이 잡히면 1초 정도 지연되어서 영상이 계속 지연되는 것입니다\*\*

현재 본인 확인 성능 자체는 좋습니다 (처리속도 이슈는 다른 방식을 채택해야 할 것 같습니다)

\*\*제 생각에는 DB에 저장된 영상을 분석하는 데에 쓰여도 좋지 않을까 합니다\*\*

해당 얼굴인식은 xml 파일 형태로 미리 훈련이 된 모델로 얼굴의 feature를 잡아냅니다.

