

Face Tracking

using ~~OpenCV~~, ~~dlib~~, ~~face_recognition~~,
~~cvlib~~ modules

2021년 2학기

서경대학교 김진헌

얼굴 추적

1

- Object Tracking
 - ▣ 목표(target)이 주어졌을 때 이를 비디오 시퀀스 영상에서 찾아 내 위치 정보를 반환해 주는 알고리즘을 통칭한다.
- 여기서는 dlib에서 지원하는 discriminative correlation filter (DCF)-based visual trackers를 다루고자 한다.
 - ▣ **visual object tracking (VOT)** 2014 Challenge에서 우승
 - ▣ DCF trackers are currently a very popular method of choice for bounding boxbased tracking.
 - ▣ 얼굴뿐만 아니라 일반적인 사물 추적에 적합.
 - ▣ 이론적 기반: **Discriminative Scale Space Tracker (DSST)**
- 처음 시작할 때 추적 대상만 지정해 주는 것만이 요구사항

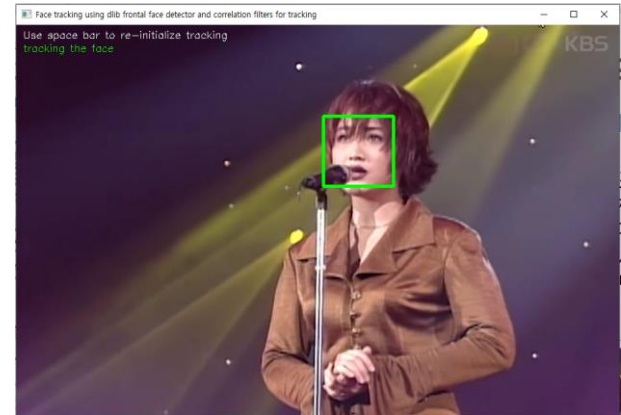
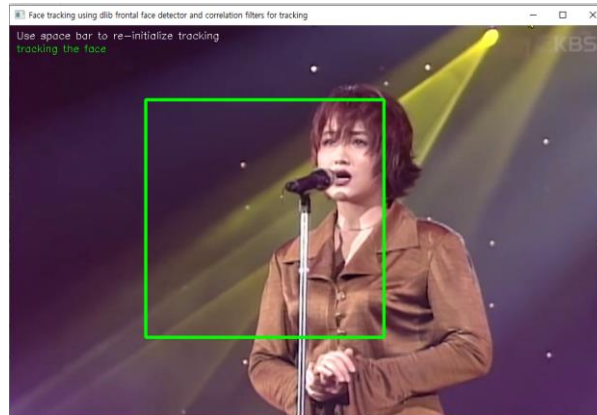
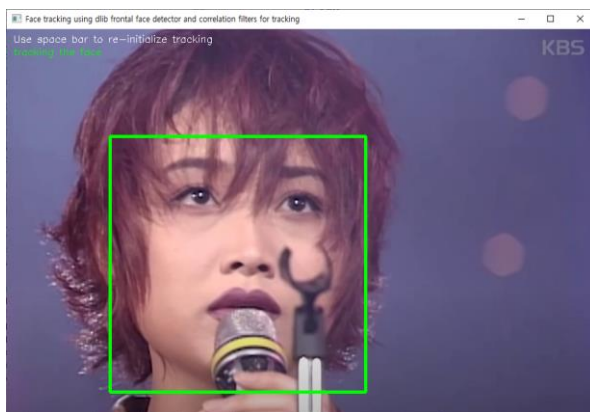
USB 캠 혹은 비디오 파일의 얼굴 추적하기

2

1_face_tracking_DCF.py

□ 동작 설명

- 처음 얼굴이 인식되는 사람을 목표로 추적을 시작한다.
 - 첫 화면에서 dlib face detector가 얼굴을 자동 검출한다.
 - 검출된 얼굴들의 첫번째를 추적 대상으로 선정한다.
- 추적 대상은 녹색 사각형으로 표시한다.
- 다시 사람을 인식하게 하고 싶으면 스페이스바를 입력한다.
- 종료할 때는 esc 키를 입력한다.



프로그램 수행 절차

3

1_face_tracking_DCF.py

- 1) dlib 함수로 frontal face detector를 생성한다.
 - `detector = dlib.get_frontal_face_detector()`
- 2) dlib 함수로 correlation tracker를 생성한다.
 - `tracker = dlib.correlation_tracker()`
- 3.1) 얼굴 추적 중이 아니라면...
 - 영상을 모노로 바꾸어 얼굴을 다시 검출하여 검출된 얼굴이 있으면 이를 목표 영상으로 설정한다.
 - `rects = detector(gray, 0)`
 - `if len(rects) > 0:`
 - `tracker.start_track(frame, rects[0])`
- 3.2) 얼굴 추적 중이라면...
 - 추적기를 업데이트하고 검출 신뢰 점수를 출력하고, 위치를 반환받아 해당 위치에 4각형을 그린다.
 - `score = tracker.update(frame)`
 - `pos = tracker.get_position()`
 - `cv2.rectangle(frame, (int(pos.left()), int(pos.top())), (int(pos.right()), int(pos.bottom()))) ...`
- 4) 키를 점검하여 재 탐색 여부와 중지를 결정한다.
- 5) 영상을 출력하고, 파일이라면 잠시(1/30 초) 출력지연시간을 유지한다.
 - `cv2.waitKey(1000//30 - 3)`
 - 이 지연시간 때문에 키보드 입력을 놓치는 경우가 많다.

USB 캠 영상에서 얼굴 추적하기

4

2_object_tracking_correlation_filters.py

□ 동작 설명

- 프로그램이 시작되면 마우스로 추적하고자 하는 물건을 드래그하여 4각영역을 선택한다.
- 키 1을 누르면 트래킹을 시작한다.
- 키 2를 누르면 중지한다.
- q를 누르면 종료한다.
- 참고: 마우스로 영역선택할 때 추적 대상을 얼굴로 추적하면 얼굴을 추적한다.

