

# 졸업프로젝트 최종보고서

- 딥러닝과 라즈베리파이를 이용한 홈 IoT 보안 시스템 -

집지김이

201502062 백우진

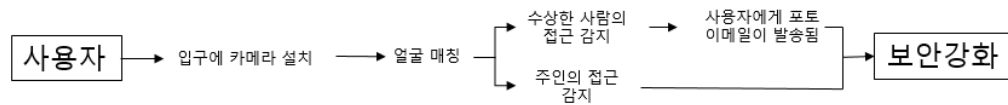
201502112 전현석

# 목차

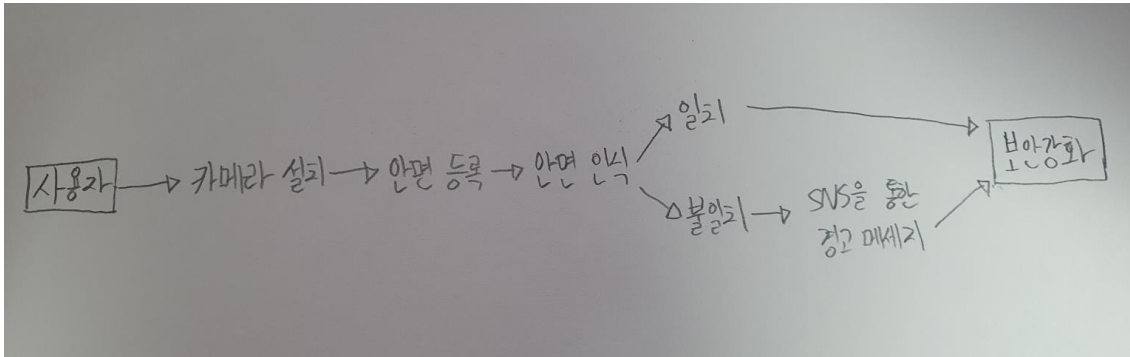
- 디자인 스프린트
- SE 문서
- 멘토링
- 설문
- 프로토타입 데모

# 디자인 스프린트

## 1. Map

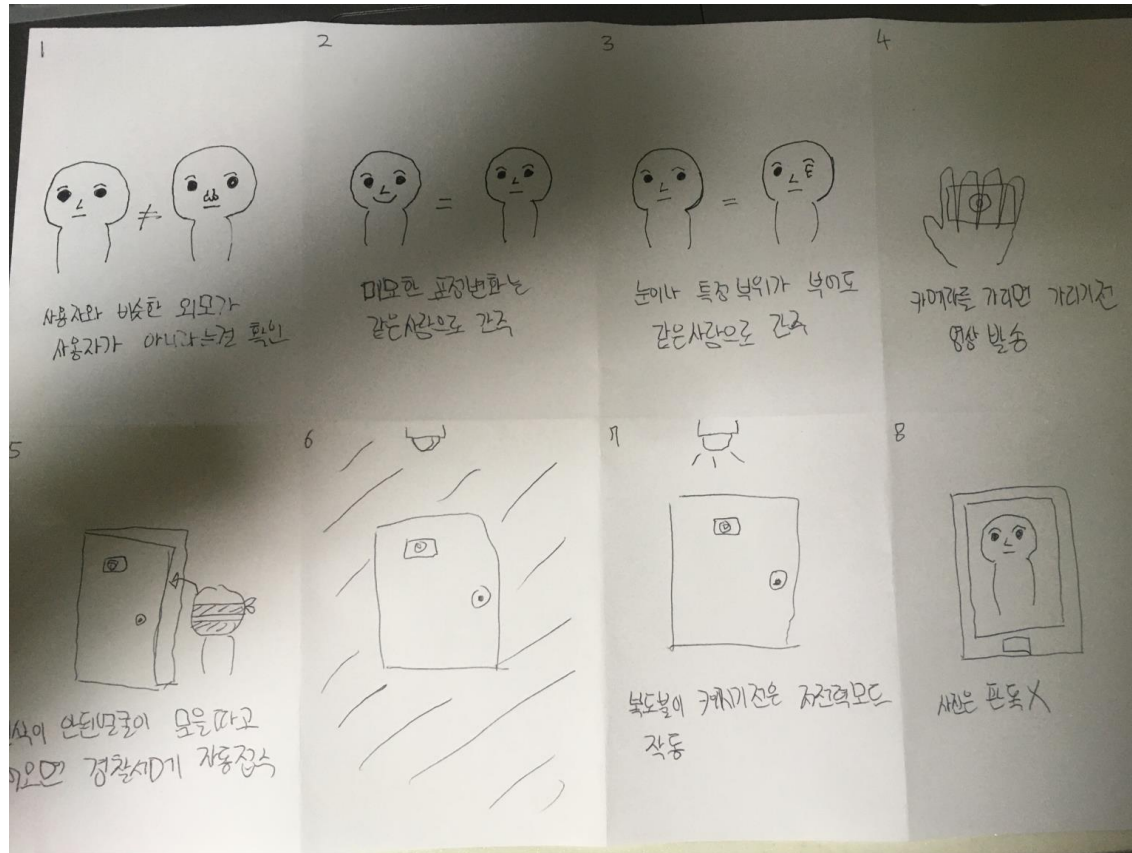


행위자(왼쪽)와 결말(오른쪽)을  
사용자와 보안강화로 동일하게  
설정 후 진행

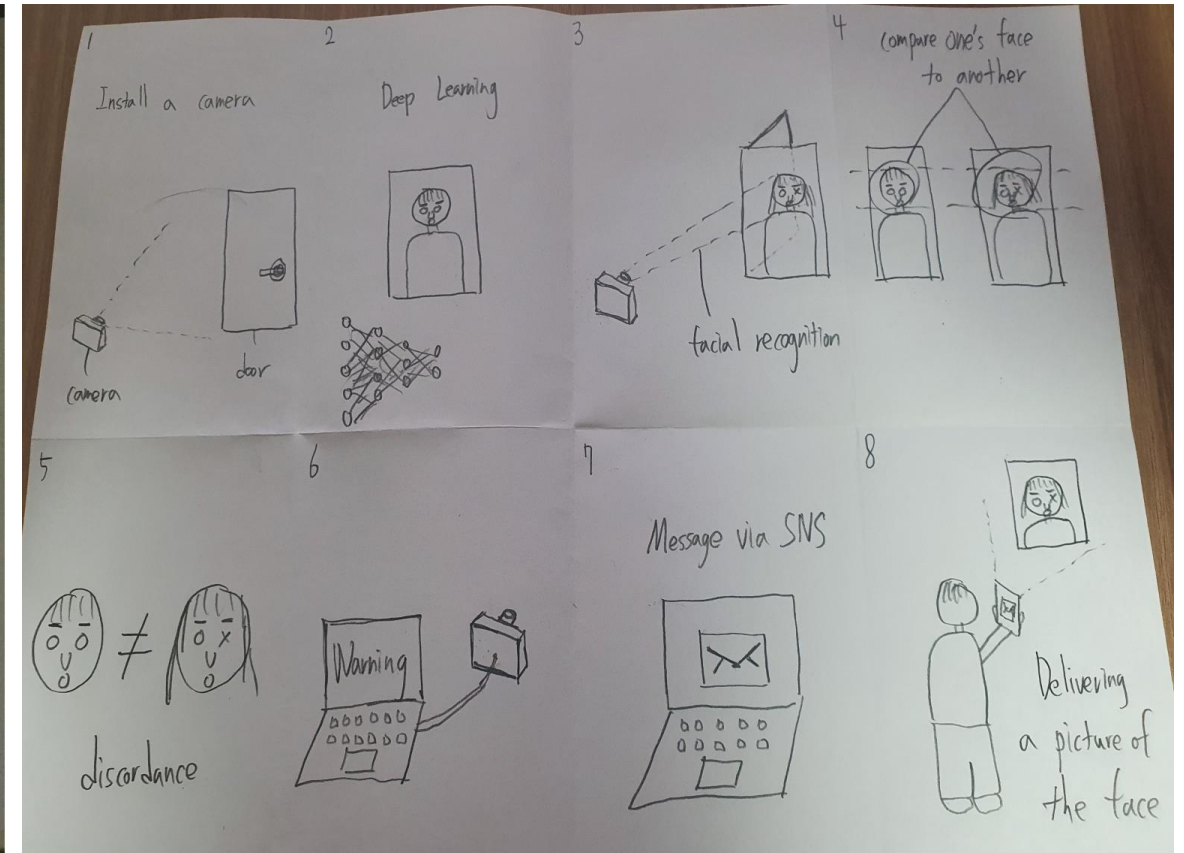


# 디자인 스프린트

## 2. Sketch



- 백우진 -



- 전현석 -

# 디자인 스프린트

## 3. Solution

- 1) 정면 얼굴로 출입을 하지 않을 시에는 안면인식을 어떻게 할 수 있을까?
  - 지문등록을 할 때 측면 부분까지 스캔하는 것처럼 얼굴도 정면뿐만 아니라 옆모습까지 딥러닝 시켜서 안면 등록을 한다.
- 2) 사진과 실물을 어떻게 비교 할 수 있을까?
  - 안면등록을 할 때 웃는 모습이나 모션 등의 옵션을 추가하여서, 안면인식을 할 때 그 옵션까지 하게 해서 보안을 강화한다.
- 3) 집지킴이 어플 or 웹의 보안을 좀 더 강화 할 수 있는 방법이 있을까?
  - 안면인식을 통해 불일치되면 SNS를 통해 사진이 전송되므로, 신고 기능을 추가하여 사진을 첨부하고 담당 기구에 신고할 수 있게 한다.
- 4) 여성은 화장 전후가 달라서 안면인식이 안될 수도 있는데 어떻게 하면 될까?
  - 딥러닝을 통한 안면인식뿐만 아니라 목소리 등록 기능을 추가한다.
- 5) 집지킴이에 추가하면 좋은 기능이 뭐가 있을까?
  - 주로 범죄는 했던 사람이 또 하므로, 경찰에서 등록한 범죄자 사진과 연동하여 집에 범죄자가 들어올 때 바로 경찰서에 신고가 되게 한다.

**\*투표를 통해 3번 솔루션을 적용\***

# SE 문서

1. 문제 정의서 (첨부)
2. 요구사항 명세서 (첨부)
3. 유스케이스 (첨부)
4. 클래스 다이어그램 (첨부)
5. 시퀀스 다이어그램 (첨부)

# 멘토링

## 1. 피드백

-졸업프로젝트의 방향성

도중에 주제를 한번 바꾸는 바람에 교수님께서 졸업프로젝트 진행이 힘든 줄 알았지만 어느정도 틀을 잡아가고 있으니 앞으로 좀 더 자주 만나며 졸업프로젝트를 진행하라고 하셨고, 현재 3명의 팀원 중 두명만 졸업프로젝트를 진행하고있지만 나중에 한명이 더 투입될 것을 감안하여 주제의 크기도 좀 키우는 것이 좋을 것이라고 말씀해주었습니다.

-공부 방식에 대한 조언

현재 인프라에 파이썬 데이터시각화, 텐서플로2마라톤예측, 케라스, 텐서플로2 최신기법, opencv, iot딥러닝 등의 강의를 구입하여 졸업프로젝트 개발을 위한 선수 학습을 하고있다고 말씀드리니, 현재 잘하고 있으며 만약 부족한 지식이 있을 때는 언제든지 물어보시라고 말씀해주셨다. 현재 일주일에 한번 씩 목요일 아침마다 교수님과 회의를 진행 중입니다.

# 멘토링

## 1. 피드백

### -SNS 알람에 대한 피드백

수상한 사람의 접근 시 SNS 알람을 Push하는 방식에 대해 카카오톡을 사용할 것이라 말씀드리니 카카오톡을 통한 알람 Push보다는 독자적인 어플을 개발하여 그 어플에 알람을 Push하는 방법을 고안해 보라고 말씀해주셨습니다.

### -안면인식에 대한 피드백

유스케이스 다이어그램에 안면인식에 대한 세부 내용을 표정인식과 화장인식으로 나뉘었는데 화장인식은 오차율이 높을 것으로 예상되어 키 같은 신체 특성을 파악할 수 있는 다른 센서 방안을 추천해주었습니다.

그리고 카메라 촬영을 통한 2D 안면 인식 기술은 얼굴의 굴곡을 인식하지 못해 사진으로 무력화되기 쉽고, 어두운 밤에는 인식이 불가능한 문제점이 있기 때문에 레이저 및 적외선 등의 전용 하드웨어 사용을 통한 3D 안면 인식 기술을 추천해주었습니다.



# 멘토링

## 1. 피드백

-데이터수집에 대한 피드백

오늘 찍은 사진과 며칠 뒤에 찍는 사진은 분명히 다르기 때문에, 사진을 하루에 많이 찍는 것보다는 여러 날짜에 걸쳐 사진을 찍어두는 것을 추천해주었습니다.

## 2. 소감

생체인식 분야는 보안이 중요하기 때문에 최신 기술 동향을 잘 파악해야 한다는 걸 알았고, 오프라인을 통한 실시간 피드백이 없어서 아쉬웠습니다. 또한, 문제 정의서와 요구사항명세서를 제대로 작성하지 않아서 수정할 점이 많아진 걸 보고 처음 틀 잡는 거에 대한 중요성을 깨닫게 되었습니다.

# 설문

## 홈안서비스 프로토타입 설문조사

안녕하세요 컴퓨터공학과 15학번 백우진, 전현석입니다.

종합설계1 과제로인해 설문을 진행하게 되었습니다.  
단톡방에 보여드린 프로토타입을 보고 설문을 해주시면 감사하겠습니다.

1. SNS알람(이전)과 자체 어플(현재)을 통한 알람 중 어떤 알람 방식이 더 좋다고 생각하십니까?

- ☐ 1. 자체 어플(2번으로)
- ☐ 2. SNS(3번으로)

2. 자체 어플을 통한 알람이 더 좋다고 생각하시는 이유는 무엇입니까?

내 답변

3. SNS를 통한 알람이 더 좋다고 생각하시는 이유는 무엇입니까?

내 답변

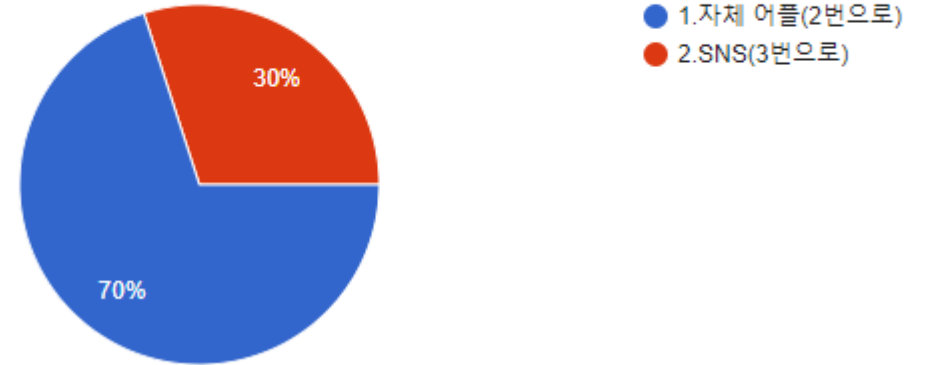
4. 현재 프로토타입에서 개선되어야 할 점이 무엇이라고 생각하십니까?

내 답변

제출

1. SNS알람(이전)과 자체 어플(현재)을 통한 알람 중 어떤 알람 방식이 더 좋다고 생각하십니까?

응답 10개



# 설문

- SNS 알람이 더 좋은 이유

자체 어플은 설치해야하는 번거로움이 있다.

카카오톡을 이용한 알람이 확인하기 더 쉽다.

어플 개발에 들어가는 비용을 SNS를 이용함으로써 축소시킬수 있을 것 같다.

# 설문

- 자체 어플 통한 알람이 더 좋은 이유

SNS를 통한 알람은 기능이 제한되어있지만 자체적으로 어플을 개발한다면 여러가지 기능을 추가할 수 있을것 같다.

SNS를 이용한 알람은 기존의 홈 IOT 보안서비스와 크게 다르지 않다고 생각한다. 자체 어플 개발을 통한 확장성을 생각해보면 좋겠다.

# 설문

- 현재 프로토타입에서 개선되어야 할 점

기존 안면인식 기술과 큰 차별성이 없는것 같다.

기존의 홈 보안서비스와 다른점이 없다.

카메라 촬영을 통한 2D 안면 인식 기술은 얼굴의 굴곡을 인식하지 못해 사진으로 무력화되기 쉽고 어두운 밤에는 인식이 불가능한 문제점 있다고 생각한다.

# 설문

- 디자인 스프린트 대비 개선점

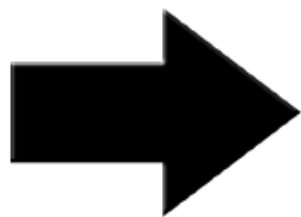
기존의 집지킴이 서비스

SNS를 통한 알람

얼굴만 인식

라즈베리파이 기본 카메라

휴대폰에 사진 저장



개선된 집지킴이 서비스

자체 어플을 통한 알람

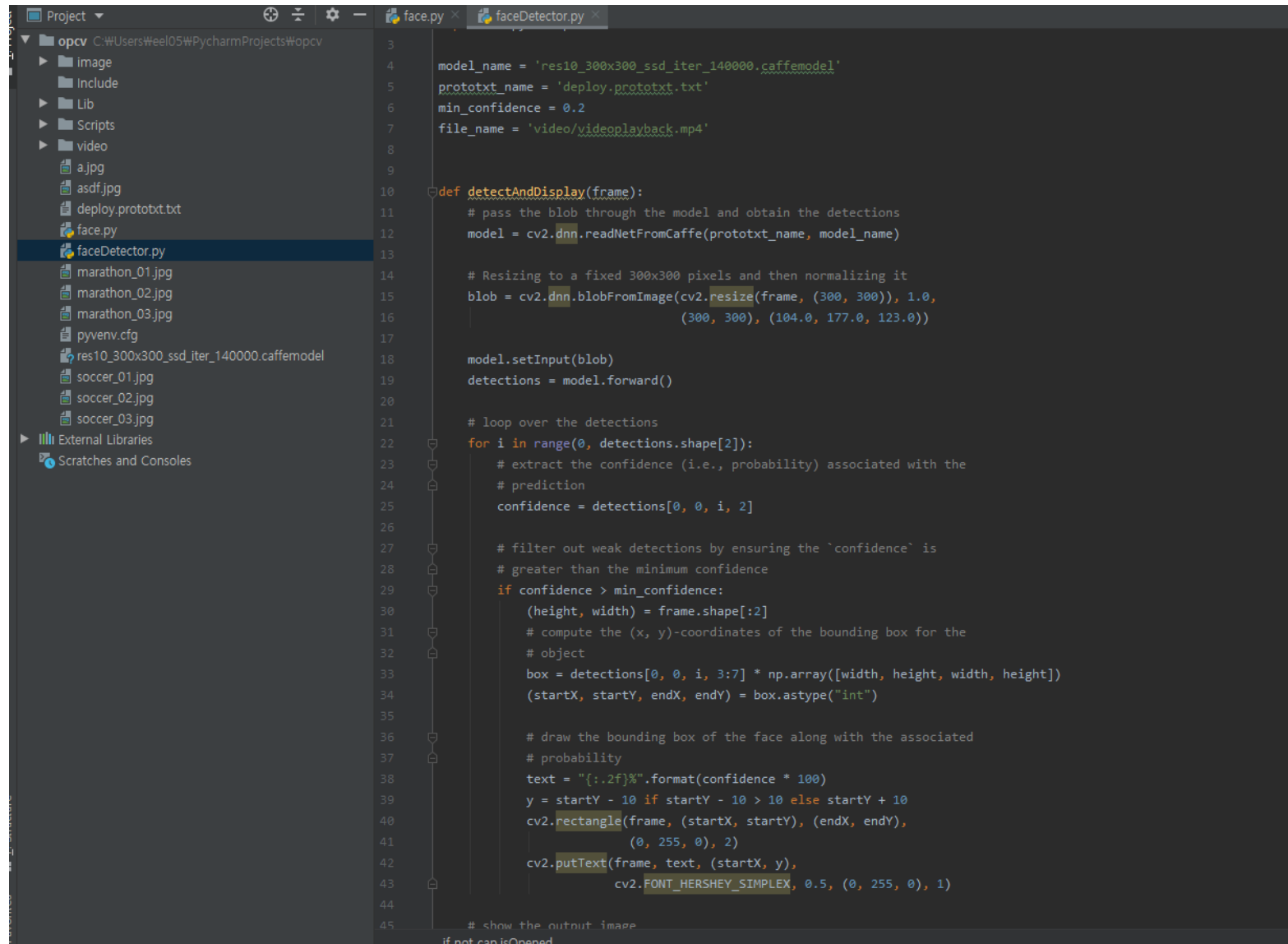
얼굴+신체적 특징까지 인식

적외선 or 레이저 카메라

클라우드에 사진 저장

# 프로토타입 데모

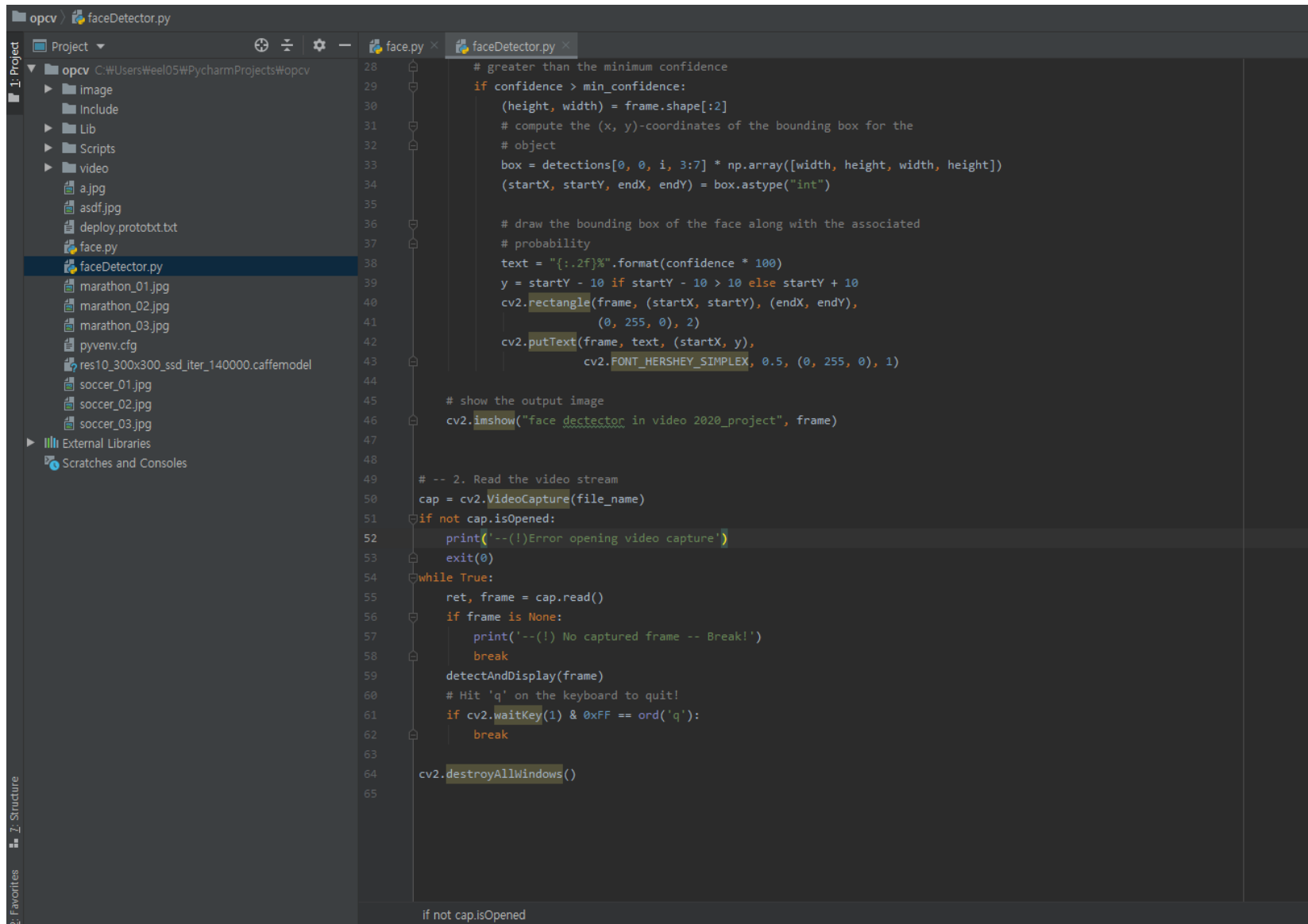
## 1. 코드



```
1  # Project: opcv C:\Users\Weel05\PycharmProjects\Wopcv
2
3  model_name = 'res10_300x300_ssd_iter_140000.caffemodel'
4  prototxt_name = 'deploy.prototxt.txt'
5  min_confidence = 0.2
6  file_name = 'video/videoplayback.mp4'
7
8
9
10 def detectAndDisplay(frame):
11     # pass the blob through the model and obtain the detections
12     model = cv2.dnn.readNetFromCaffe(prototxt_name, model_name)
13
14     # Resizing to a fixed 300x300 pixels and then normalizing it
15     blob = cv2.dnn.blobFromImage(cv2.resize(frame, (300, 300)), 1.0,
16                                   (300, 300), (104.0, 177.0, 123.0))
17
18     model.setInput(blob)
19     detections = model.forward()
20
21     # loop over the detections
22     for i in range(0, detections.shape[2]):
23         # extract the confidence (i.e., probability) associated with the
24         # prediction
25         confidence = detections[0, 0, i, 2]
26
27         # filter out weak detections by ensuring the 'confidence' is
28         # greater than the minimum confidence
29         if confidence > min_confidence:
30             (height, width) = frame.shape[:2]
31             # compute the (x, y)-coordinates of the bounding box for the
32             # object
33             box = detections[0, 0, i, 3:7] * np.array([width, height, width, height])
34             (startX, startY, endX, endY) = box.astype("int")
35
36             # draw the bounding box of the face along with the associated
37             # probability
38             text = "{:.2f}%".format(confidence * 100)
39             y = startY - 10 if startY - 10 > 10 else startY + 10
40             cv2.rectangle(frame, (startX, startY), (endX, endY),
41                           (0, 255, 0), 2)
42             cv2.putText(frame, text, (startX, y),
43                        cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 255, 0), 1)
44
45     # show the output image
46     if not cap.isOpened:
```

# 프로토타입 데모

## 1. 코드

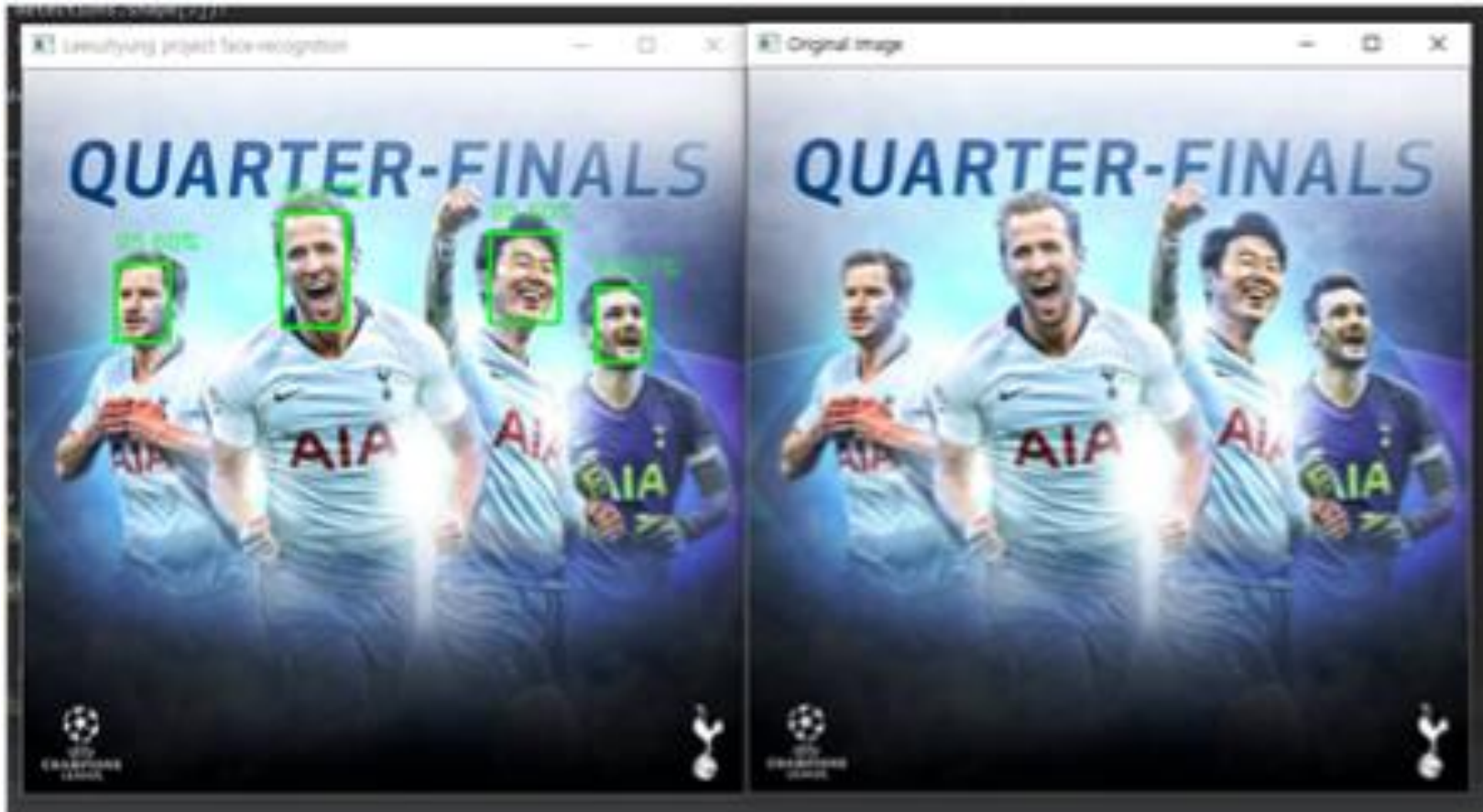


```
28 # greater than the minimum confidence
29 if confidence > min_confidence:
30     (height, width) = frame.shape[:2]
31     # compute the (x, y)-coordinates of the bounding box for the
32     # object
33     box = detections[0, 0, i, 3:7] * np.array([width, height, width, height])
34     (startX, startY, endX, endY) = box.astype("int")
35
36     # draw the bounding box of the face along with the associated
37     # probability
38     text = "{:.2f}%".format(confidence * 100)
39     y = startY - 10 if startY - 10 > 10 else startY + 10
40     cv2.rectangle(frame, (startX, startY), (endX, endY),
41                   (0, 255, 0), 2)
42     cv2.putText(frame, text, (startX, y),
43                 cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 255, 0), 1)
44
45 # show the output image
46 cv2.imshow("face detector in video 2020_project", frame)
47
48
49 # -- 2. Read the video stream
50 cap = cv2.VideoCapture(file_name)
51 if not cap.isOpened:
52     print('--(!)Error opening video capture')
53     exit(0)
54 while True:
55     ret, frame = cap.read()
56     if frame is None:
57         print('--(!) No captured frame -- Break!')
58         break
59     detectAndDisplay(frame)
60     # Hit 'q' on the keyboard to quit!
61     if cv2.waitKey(1) & 0xFF == ord('q'):
62         break
63
64 cv2.destroyAllWindows()
65
```



# 프로토타입 데모

## 2. 실행



- 얼굴 인식을 위한 여러가지 딥러닝 오픈소스를 활용하여 얼굴인식 기초 제작

# 프로토타입 데모

## 2. 실행



- 개인 보안용으로 사용하기 위해서는 마스크 등 변장의 상황에서도 어느정도 인식이 되어야하지만 현재 얼굴인식만으로는 사실상 힘들

# 프로토타입 데모

## 2. 실행



- DNN을 활용하여 영상에 대한 얼굴 인식 제작
- 일반적으로 사용하기에는 무리없는 인식 품질

# [집지킴이 팀 Github & 영상]

## 1. Github

- [https://github.com/eel0511/g\\_project](https://github.com/eel0511/g_project)

## 2. 영상

-

[https://drive.google.com/file/d/1\\_9pi0EyLj1SrJuBvdEOxxKODj5YozspW/view?usp=sharing](https://drive.google.com/file/d/1_9pi0EyLj1SrJuBvdEOxxKODj5YozspW/view?usp=sharing)