



# Face Guard

More easier More secure

도원결의

# 목차



프로젝트 소개



프로젝트 개요



시연영상



주요기술



문제해결

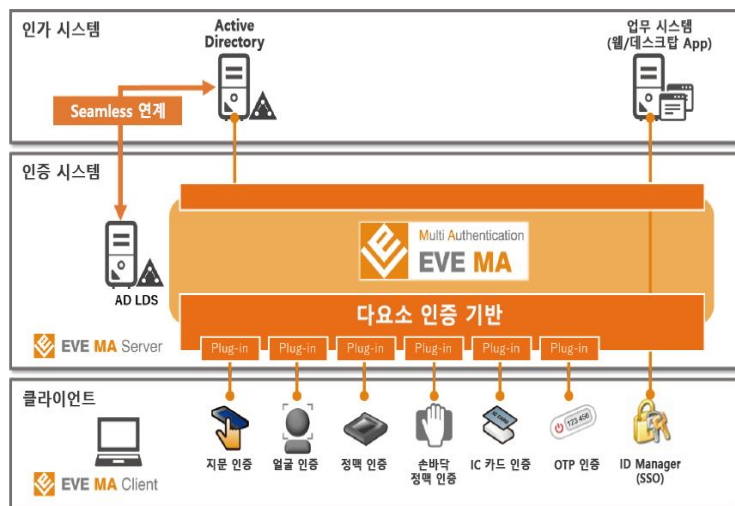


# 프로젝트 소개

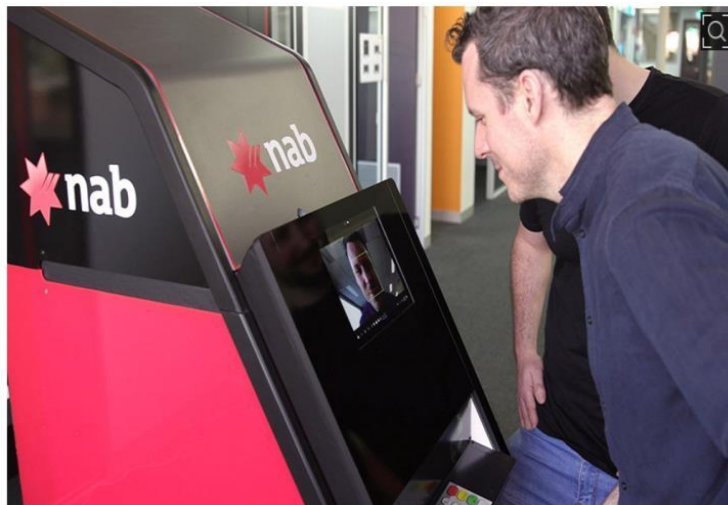
- 프로젝트 소개
- 프로젝트 동기
- 프로젝트 목적

# 프로젝트 소개

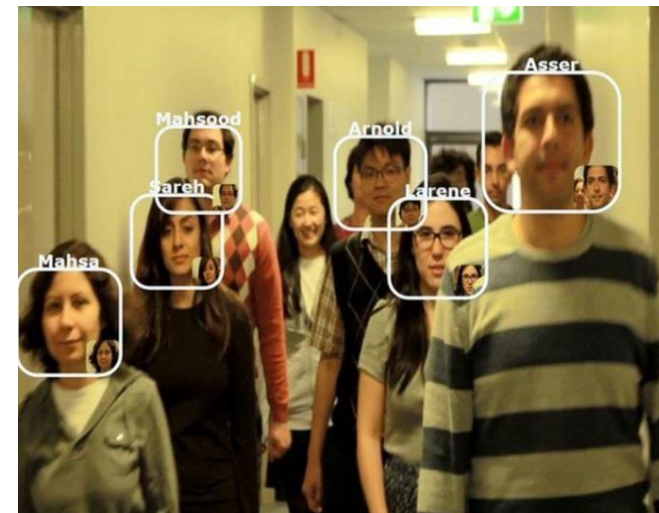
- 다중 인증 시스템의 보편화
- 생체인식을 기반으로 개인 인증을 위한 **얼굴인식 기술의 고도화**
- 스마트 디바이스의 **OTP 활용 서비스 증가**
- 비대면으로 인한 **새로운 인증방식의 필요성 증가**



<FIDO 인증 확대>



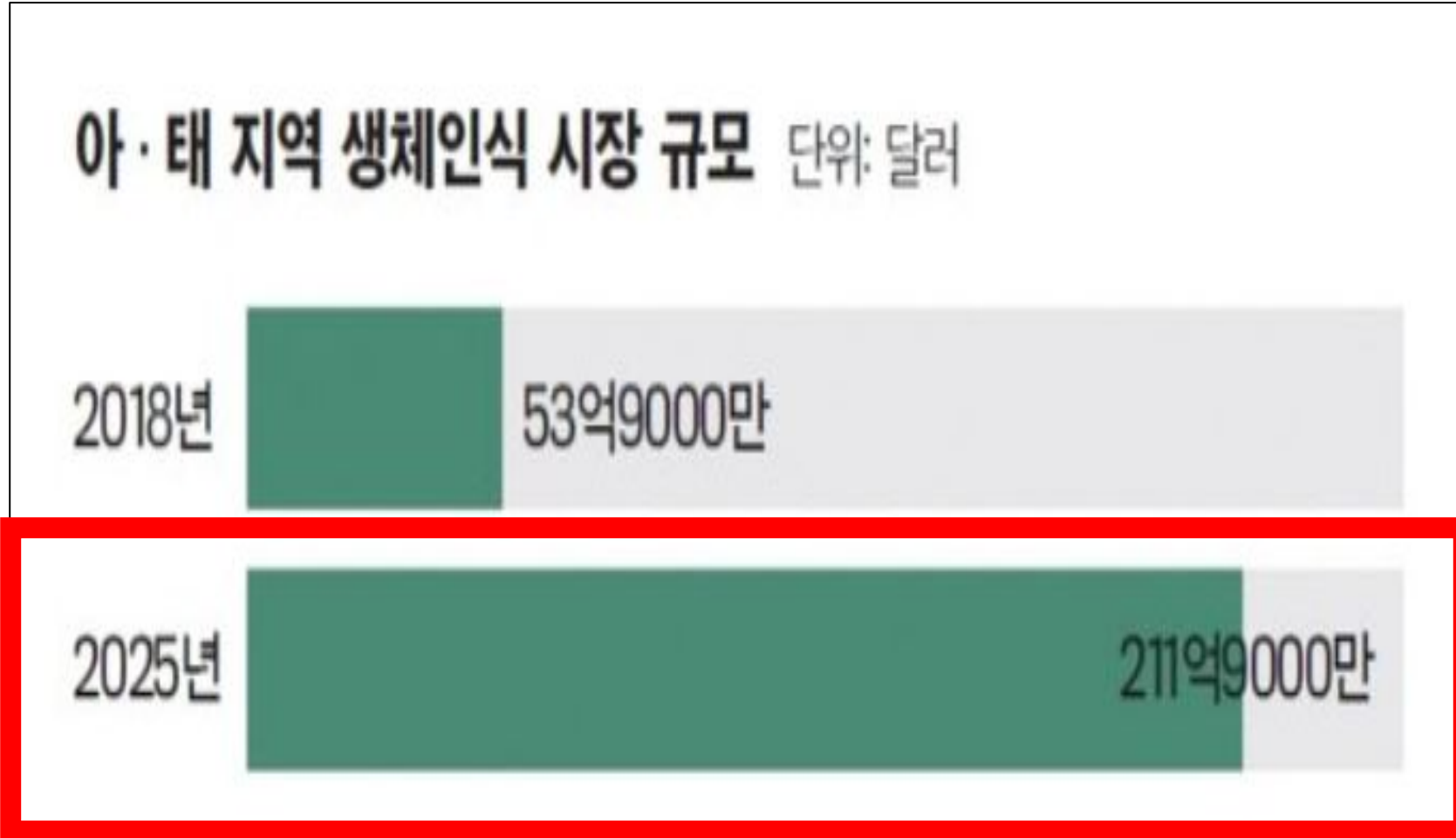
<NAB의 얼굴인식 인증>



<딥러닝 기반 얼굴인식>

# 프로젝트 동기

- 아시아·태평양 지역 생체인식 시장 규모의 증가



※ 2025년은 전망치.

자료: 프로스트앤드설리번

# 프로젝트 동기

- 생체인식 알고리즘 성능 고도화 필요성 증대

시기	주요 내용
2017년 4월	삼성 갤럭시S8, 실제 얼굴 아닌 얼굴 사진에도 잠금 해제
2018년 12월	삼성·LG·원플러스 스마트폰, 3D프린터로 복제된 얼굴인식
2019년 5월	원플러스 6T·7 프로, 한 유튜버에 의해 지문인식 무력화
2019년 10월	삼성 갤럭시S10·노트10, 초음파 지문인식 센서 오류
	구글 픽셀4, 눈 감고도 얼굴인식돼 잠금 해제 오류
	애플 아이폰X, 아들이 아버지 페이스ID 해제해 무단 결제
	텐센트, 20분 만에 주요 스마트폰 지문인식 해킹 시연 성공

# 프로젝트 동기

1. 생체인식 보안 수단이 완벽하지 않음
2. 생체정보의 위조와 유출
3. 얼굴 위·변조 공격 시도 증가

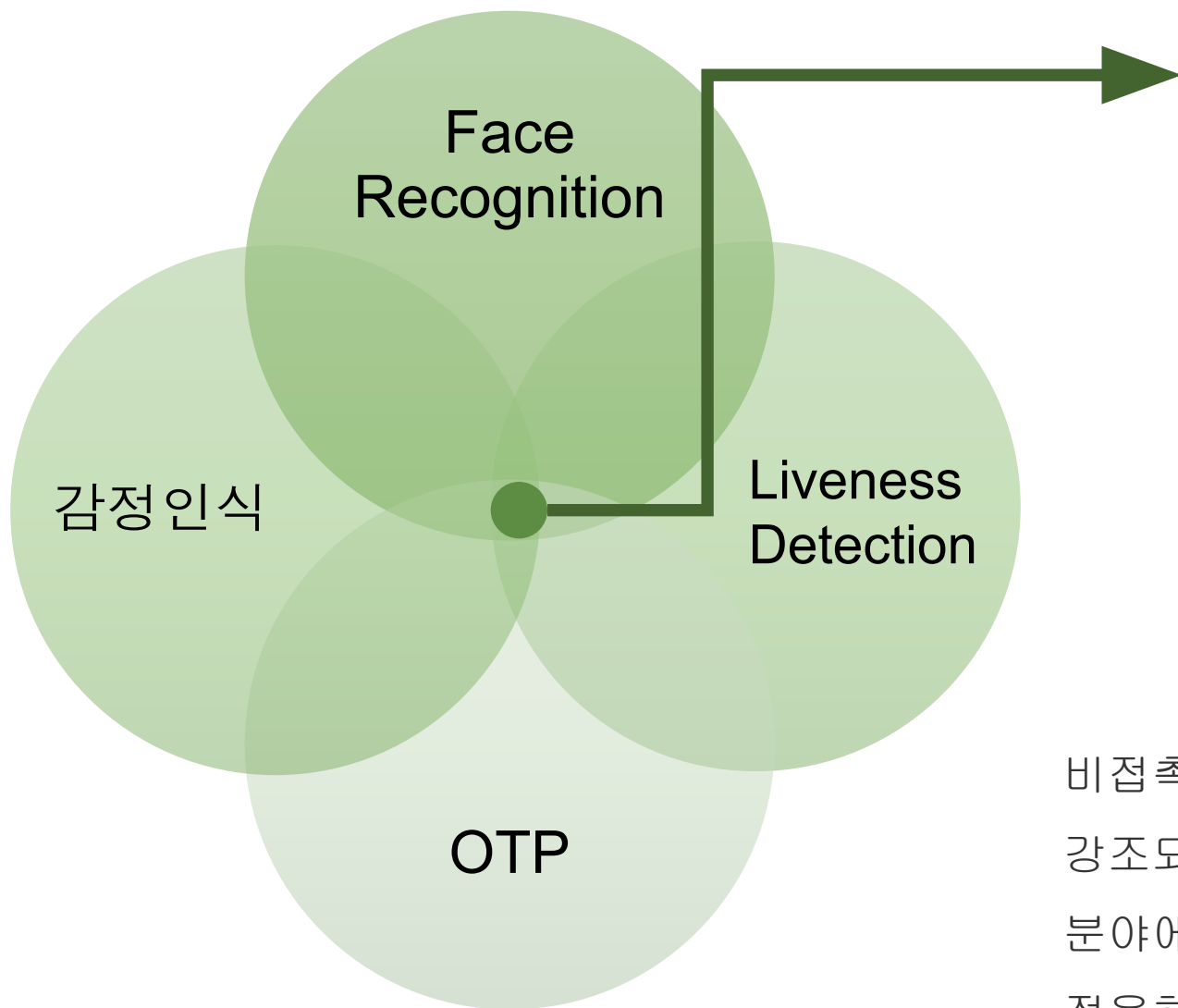


알고리즘 보완 & OTP 인증

안전한 생체인식 시스템



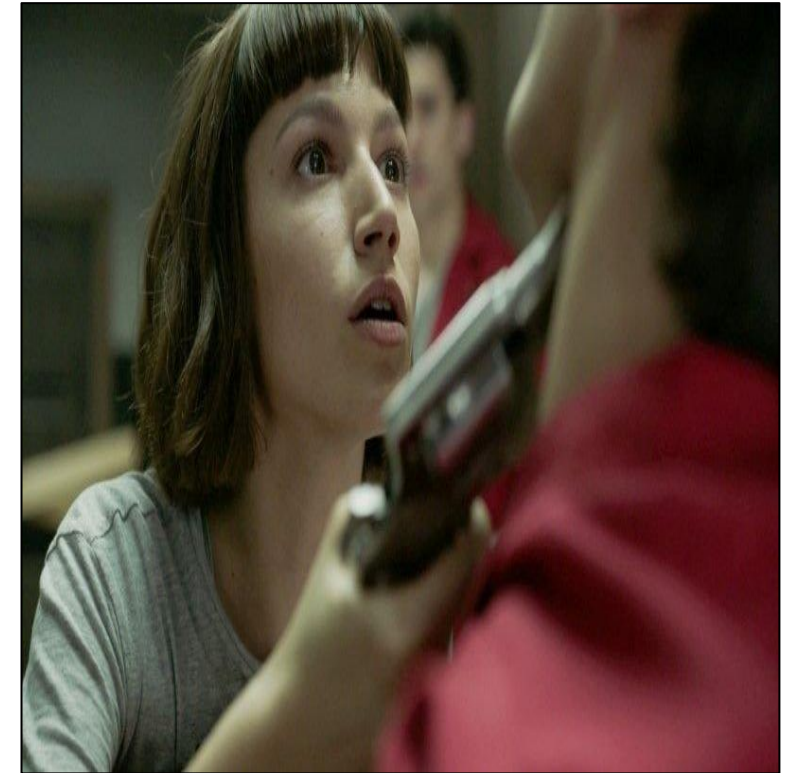
# 프로젝트 목적



비접촉 방식의 무인 인증 시스템뿐만 아니라 보안이 강조되는 스마트 금융, 얼굴인증 결제 시스템 등 다양한 산업 분야에 적용하여 얼굴 인식에 최적화된 서비스를 구현할 수 있습니다.

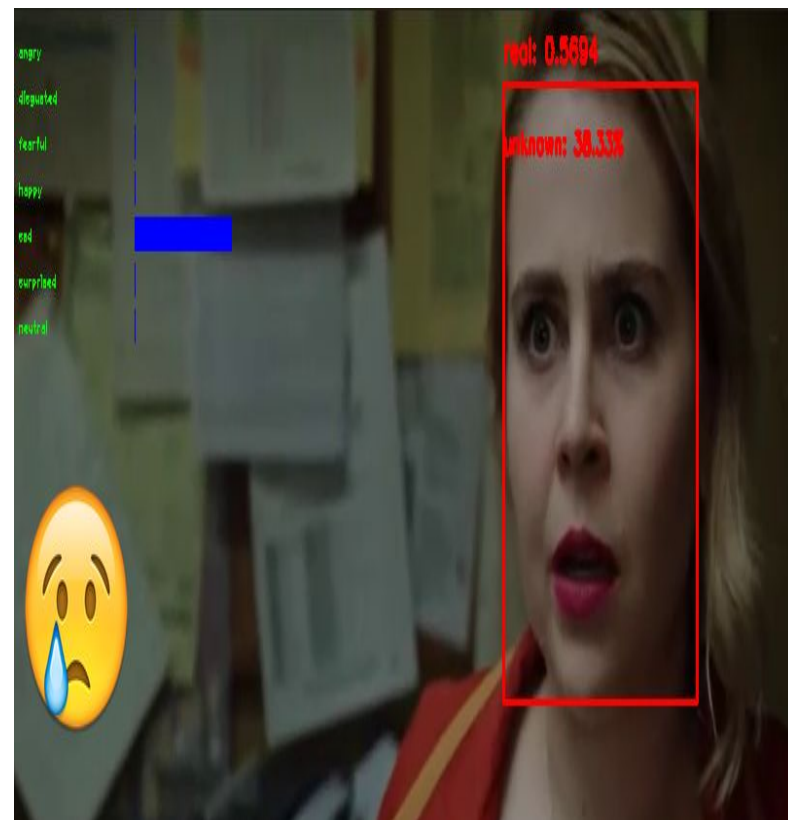
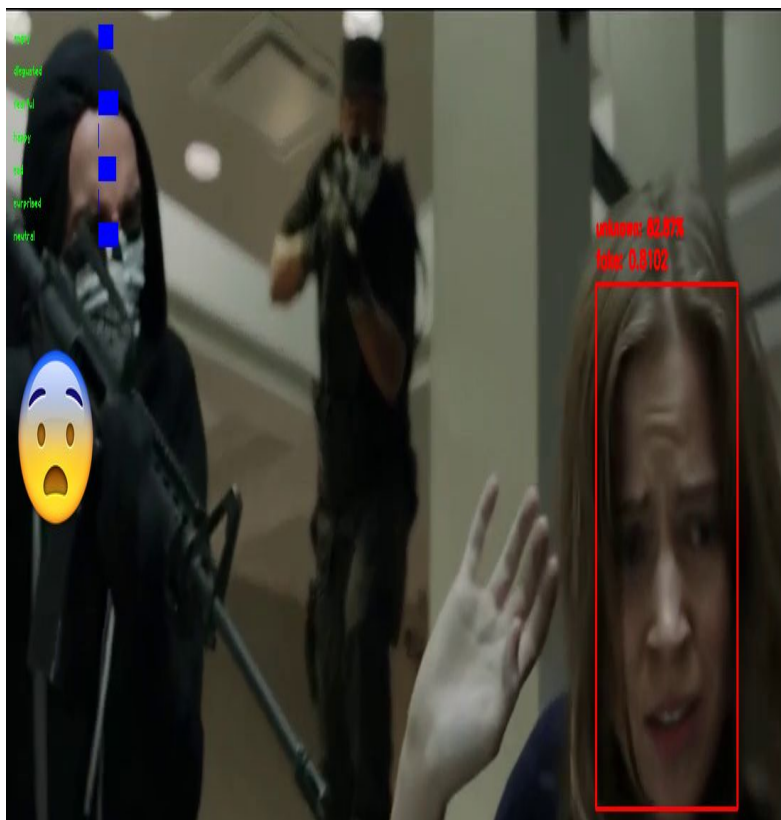


## 4. 범죄예방을 위한 감정인식 기능 추가(1)



**ATM 기기 이용객을 노린 범죄 발생**

## 4. 범죄예방을 위한 감정인식 기능 추가(2)



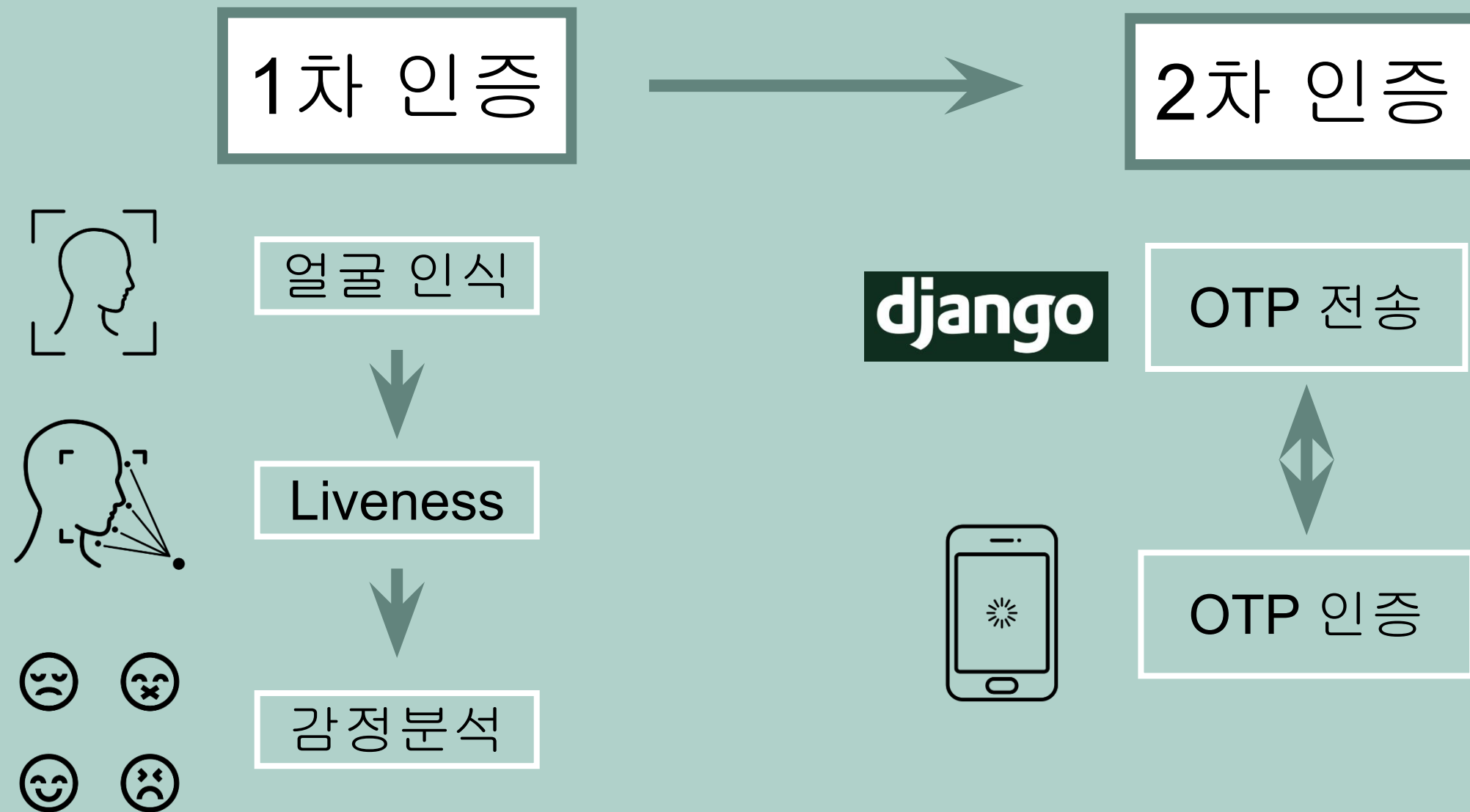
현재 감정상태를 추가로 분석하여 불안, 초조일 경우 거래중지.



# 프로젝트 개요

- 프로젝트 흐름도
- 개발방법론

# 프로젝트 흐름도



# 개발방법론

## 1. Agile 방법론

### ■ 데일리 스크럼(Daily Scrum)

- 매일 현재 상태를 업데이트하고 조율하는 것

1) 혜린  
1. 우분투 오류 나는 것을 확인한다. (딱 한시간 반)  
    ↳ 오류나는 것을 파악했을 시) 수정하고 codeSumUp2 에서 취합한 코드가 잘 돌아가게 한다.  
    ↳ 오류 수정을 해결하지 못했을 시 )  
        A) 하나하나 적용한다.  
        B) 그래도 실패시 다시 output을 뺀다  
2. 안드로이드 진짜 진짜 최종할거 준비해 (10시 ~ 12시)  
    ↳ 비디오  
    ↳ 사진  
3. 밑단에서 파베 깔짜거리기

-----

2) 훈  
1. 오전 문일좌 자료 이용 1차 피클 뽑기 (real -Fake)  
2. 와다다다 테스트  
    ↳ OK) TV, 빔, 핸드폰 촬영도 고려  
3. 케글에 있는 마스크 분류 사진 이용 fake에 넣는다  
    ↳ fake시 경고창 문구 수정

-----

3) 정하

오늘의 할일  
최종목표 - 목요일까지 ppt 에 들어갈 영상 및 자료 만들기 완성.  
양문일  
- ppt 목차 뽑아주기. ( 허, 홍, 황이 그에 맞는 캡처본을 주기위해.)  
허훈  
- 양문일 서버로 불러서 liveness에 관한 자료 생성 및 전달.  
- pickle 파일 전달.  
- 결과값 고도화....  
정하  
- 6번 인증수행 in Django 시작.  
혜린  
- 시연영상 (목요일 오전)

오후 12:21

<데일리 스크럼 적용 예시>

### ■ 칸반 보드(Kanban Board)

- 프로젝트의 진행 상황을 실시간으로 공유하기 위해 사용

할 일	진행중	끝
0/11개 업무에 포인트가 있음 0 포인트	0/1개 업무에 포인트가 있음 포인트	0/5개 업무에 포인트가 있음 0 포인트
<input type="checkbox"/> ③ Dataset 90%, 70%, 50%	<input type="checkbox"/> ② 광고-> DB 저장	<input type="checkbox"/> ① 등록 요청 100%
<input type="checkbox"/> ④ DL Training 90%, 70%, 50%	진행 중인 업무 1개	<input type="checkbox"/> django api 받아와서 사용자 정보 db 저장 기능 만들기
<input type="checkbox"/> ⑤ 인증 요청 90%		

NOV 12 업데이트 사항을 모두 확인하셨습니다 :) 새 알림 다시 알림 지움

△ 팀원들이 보낸 모든 알림을 받을 수 있는 곳입니다.

지난 주 모두 읽은 상태로 표시 • 모두 지우기

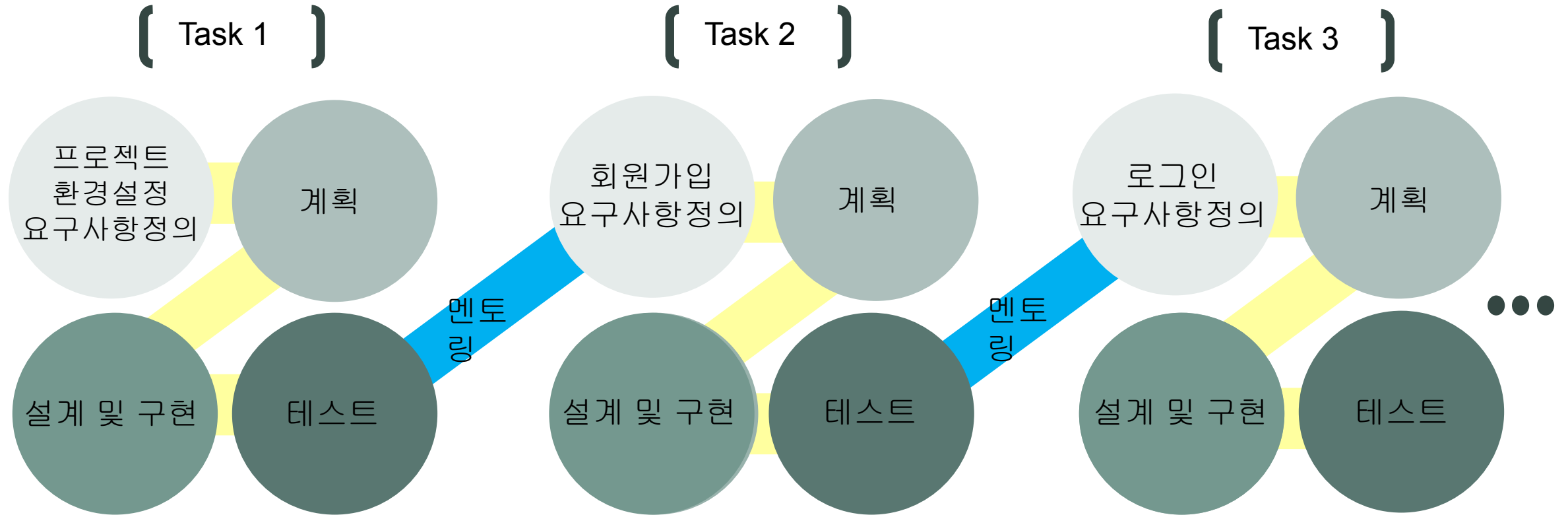
- 허훈 ① 등록 요청 100% 업무에 회원님을 배정했습니다. 11월 7일 15:09  
    ↳ 파일별 프로젝트 > 끝
- 혜린 홍 '강정분석' 에 대해서는 한번 회의 업무에 회원님을 배정했습니다. 11월 6일 22:05  
    ↳ 파일별 프로젝트 > 할 일
- 더 큰 활동들 (4)

<칸반 보드 적용 예시>



# 개발방법론

## 1. Agile 방법론



태스크 완료 시 멘토링으로 프로젝트 점검 및 다음 단계 진행

## 2. 간트 차트(Gantt chart)

- 프로젝트 일정관리를 위한 바(bar)형태의 도구로서, 각 업무별로 일정의 시작과 끝을 그래픽으로 표시

프로젝트 기간		2020.09 ~ 2020. 12. 04			
구분	추진내용	프로젝트기간			
		9월	10월	11월	12월
계획	아이디어 논의, 실현 가능성 파악, 세부 계획 수립, 각 단계 산출물 및 개발 방법론 정의, 개발 도구 선택, 역할분담				
분석	핵심 기능의 원리 탐색, 개발도구 세팅 및 파악				
설계	프로젝트 전체 흐름도, ERD 설계도 작성				
개발	Google OTP 사용 Django 연결 DB 연결				
	얼굴 인식을 이용한 Liveness Detection 모델링 감정 분석 모델링				
	웹 구현 및 하이브리드 앱 구현				
테스트	통합 테스트 완료 및 오류 수정				
종료	진행사항 보고 피드백, 최종 산출물(웹/앱, ppt)				

<간트 차트 적용 예시>

# 시연영상

- 회원가입
- 로그인
- 최종 로그인 결과



# 시연영상 - 회원가입



## <개인정보 수집 안내>

### 📄 개인정보 수집 동의서

◆수집항목◆ : 성명, 전화번호, 얼굴 정보

◆수집 및 이용 목적◆

- Facein 인증 진행

- 사용자 사진의 스푸핑 및 본인 식별

- 사용자 자원 관리

◆수집 기간◆

위 개인정보는 수집이용에 관한 동의일로부터 보유목적 달성 시 또는 정보주체가 개인정보 삭제를 요청할 경우 지체 없이 파기합니다.

단, 사업 종료일 후에는 향후 정부지원사업 신청 시의 이력관리만을 위하여 보유·이용되며 기간은 1년입니다.(공공기록물 관리에 관한 법률 시행령)

◆동의를 거부할 권리 및 동의를 거부할 경우의 불이익◆

위 개인정보 중 필수항목의 수집·이용에 관한 동의는 인증 수행을 위해 필수적이므로 이에 동의하셔야 이후 절차를 진행할 수 있습니다.

다만 동의하지 않으시는 경우 본 어플 인증이 불가능합니다.

거절      동의

2020/12/08 18:29:26  
동의하셨습니다.

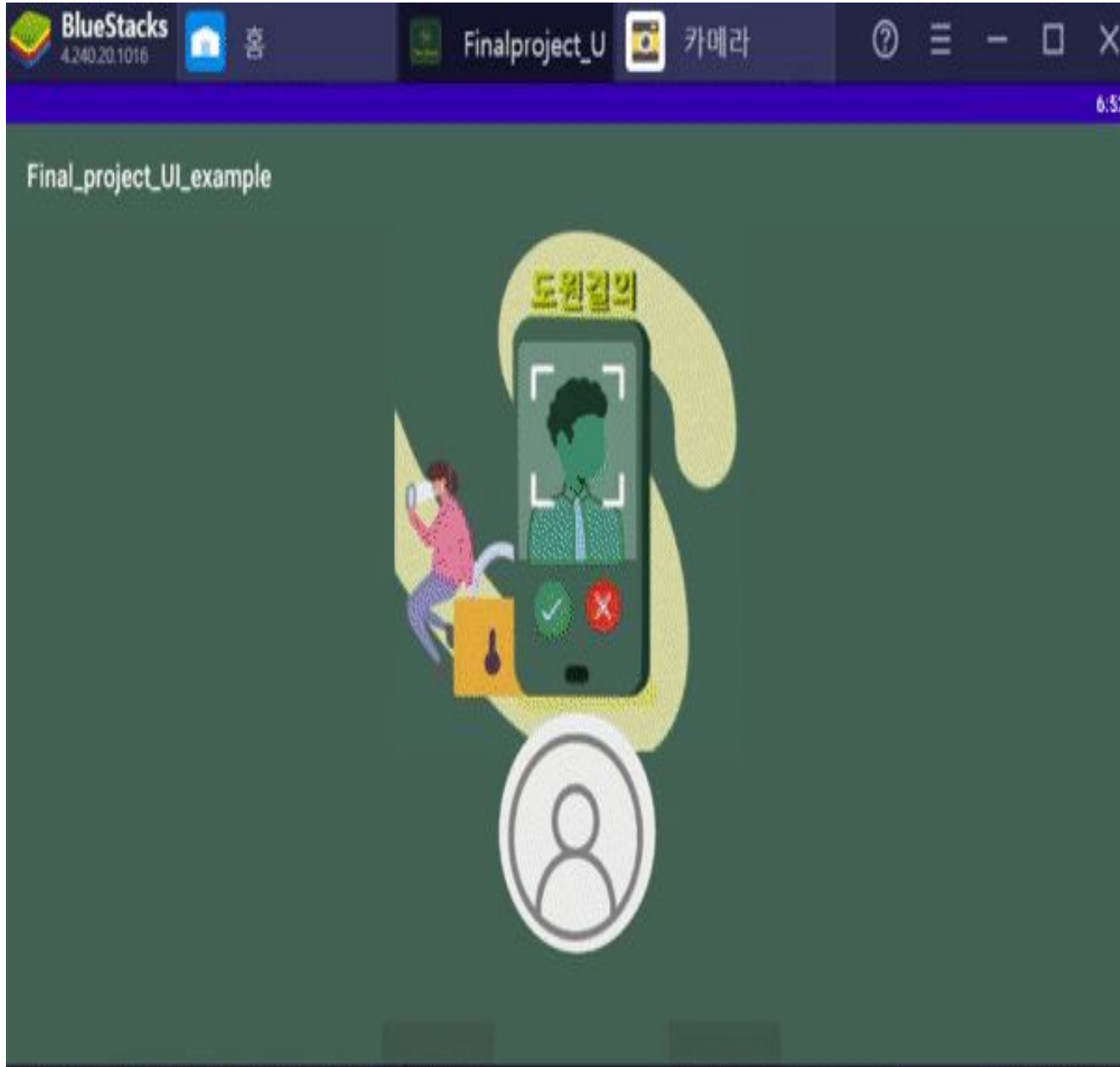
거절하셨습니다.  
Face Guard 회원가입이 어렵습니다.

## <회원가입 시 사용자의 전화번호 입력 후 중복확인 버튼으로 다중 회원가입 방지>

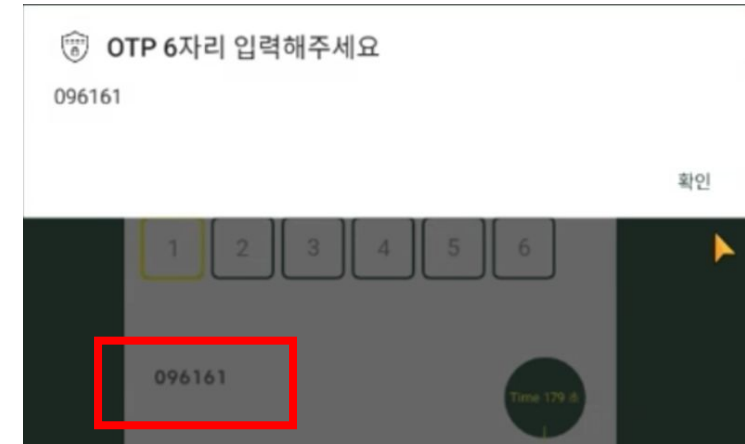


01037-■■■■■■■■■■  
사용가능한 전화번호입니다.

# 시연영상 - 로그인



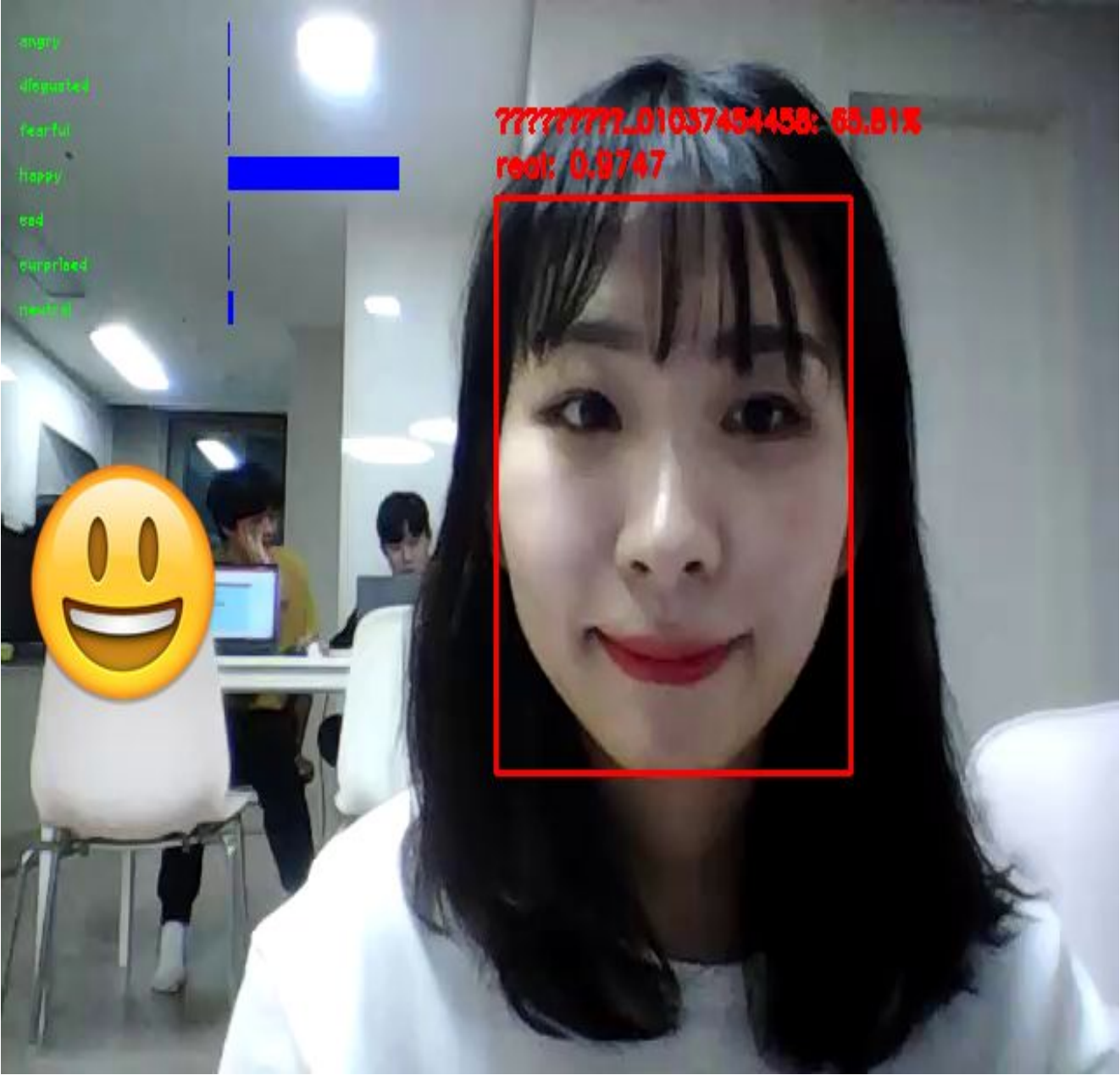
## <OTP 푸쉬알림 및 토스트 적용>



## <사용자의 얼굴인식 및 감정분석>



# 시연영상 - Final



<최종결과>

사용자 얼굴인식	Liveness	감정분석
홍혜린 (0103xxx4xxx)	Real	Happy
85.81%	0.9747	0.9747
‘홍혜린’님이 로그인에 성공하였습니다.		

## 주요기술 설명

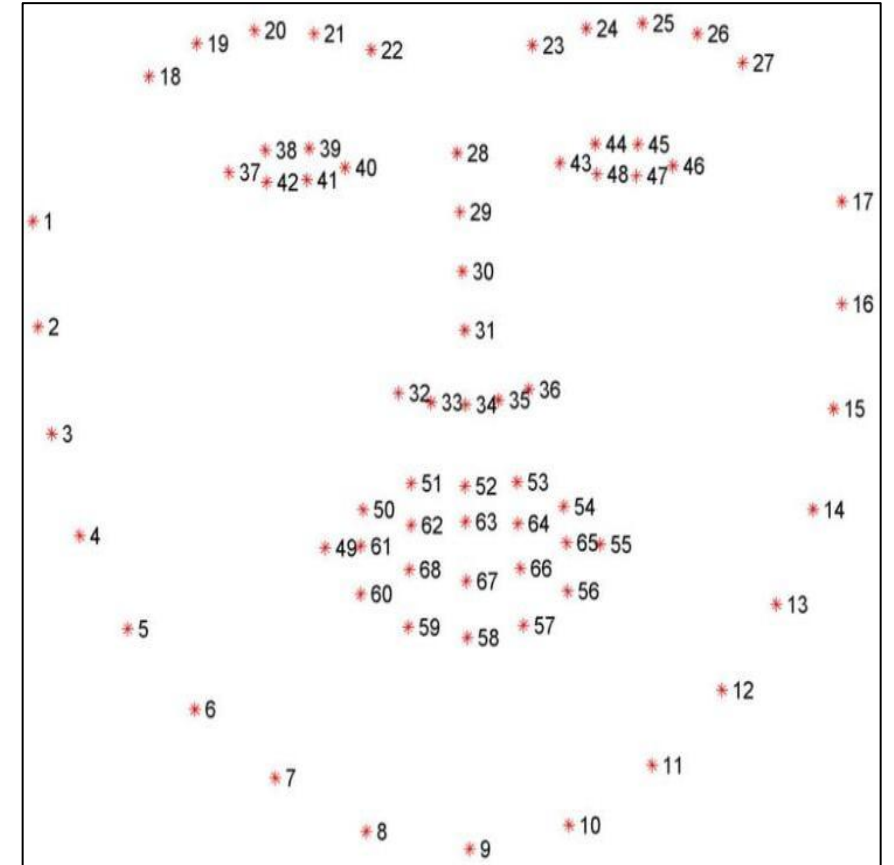
- Face landmark검출
- 얼굴인식 딥러닝
- **Liveness** 감지
- 감정인식
- 종합 얼굴 분석 결과
- OTP

# 주요 기술

## 1. Face landmark검출



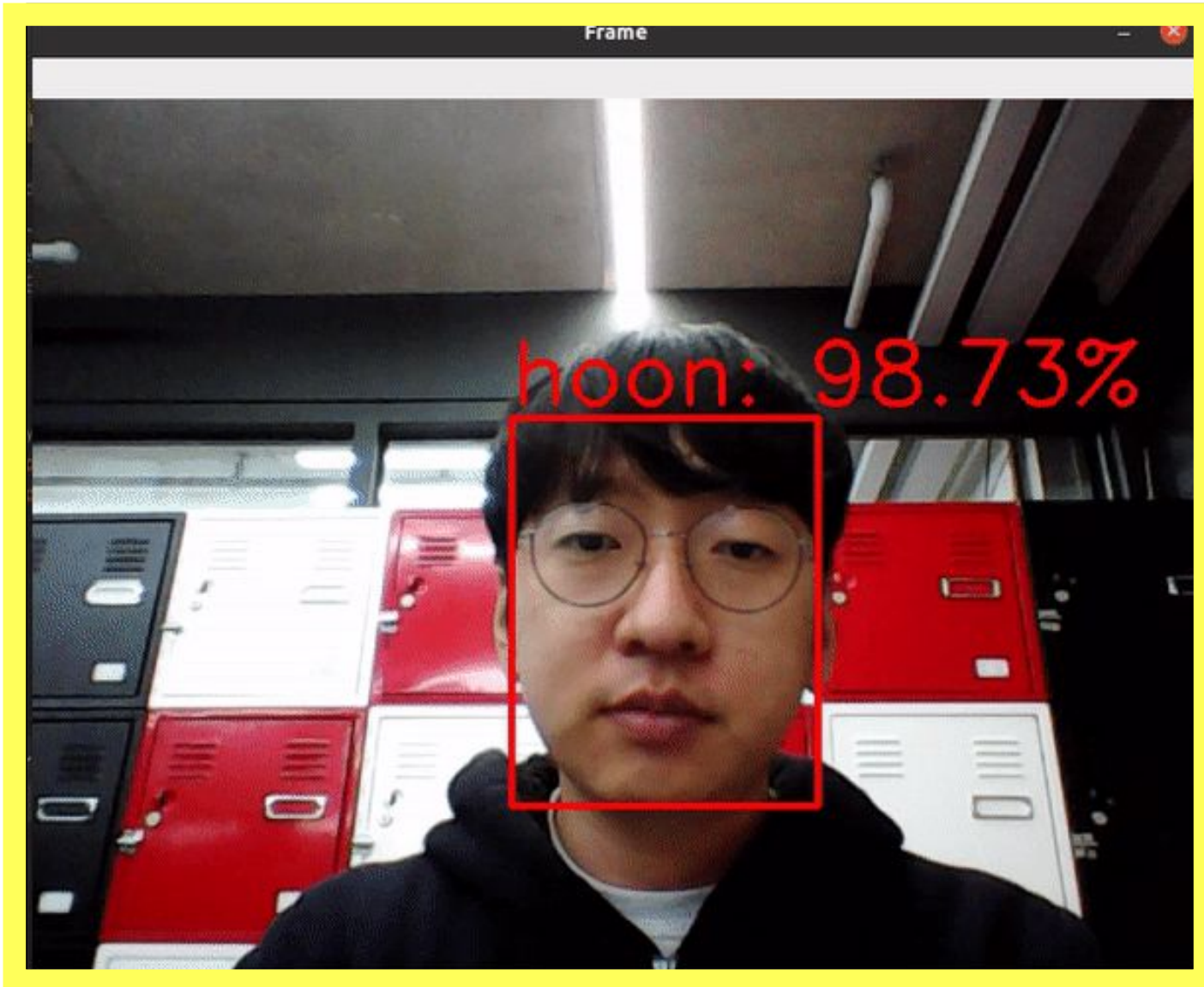
### <얼굴 랜드마크 시각화 예시>



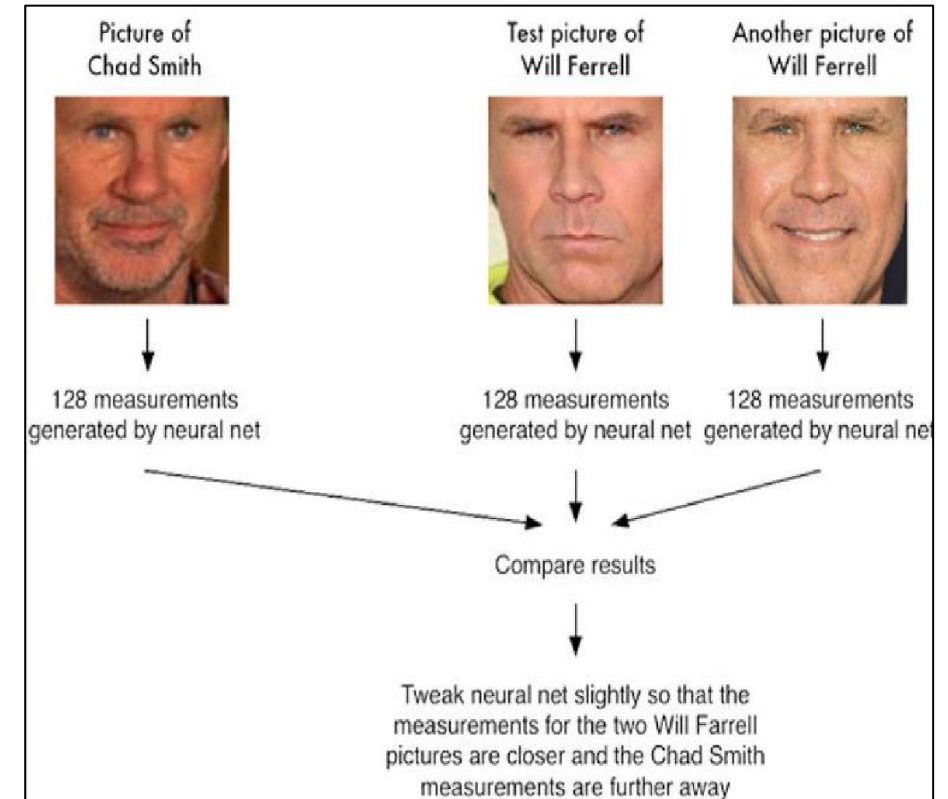


# 주요 기술

## 2. 얼굴 인식 딥러닝(dlib + face\_recognition)

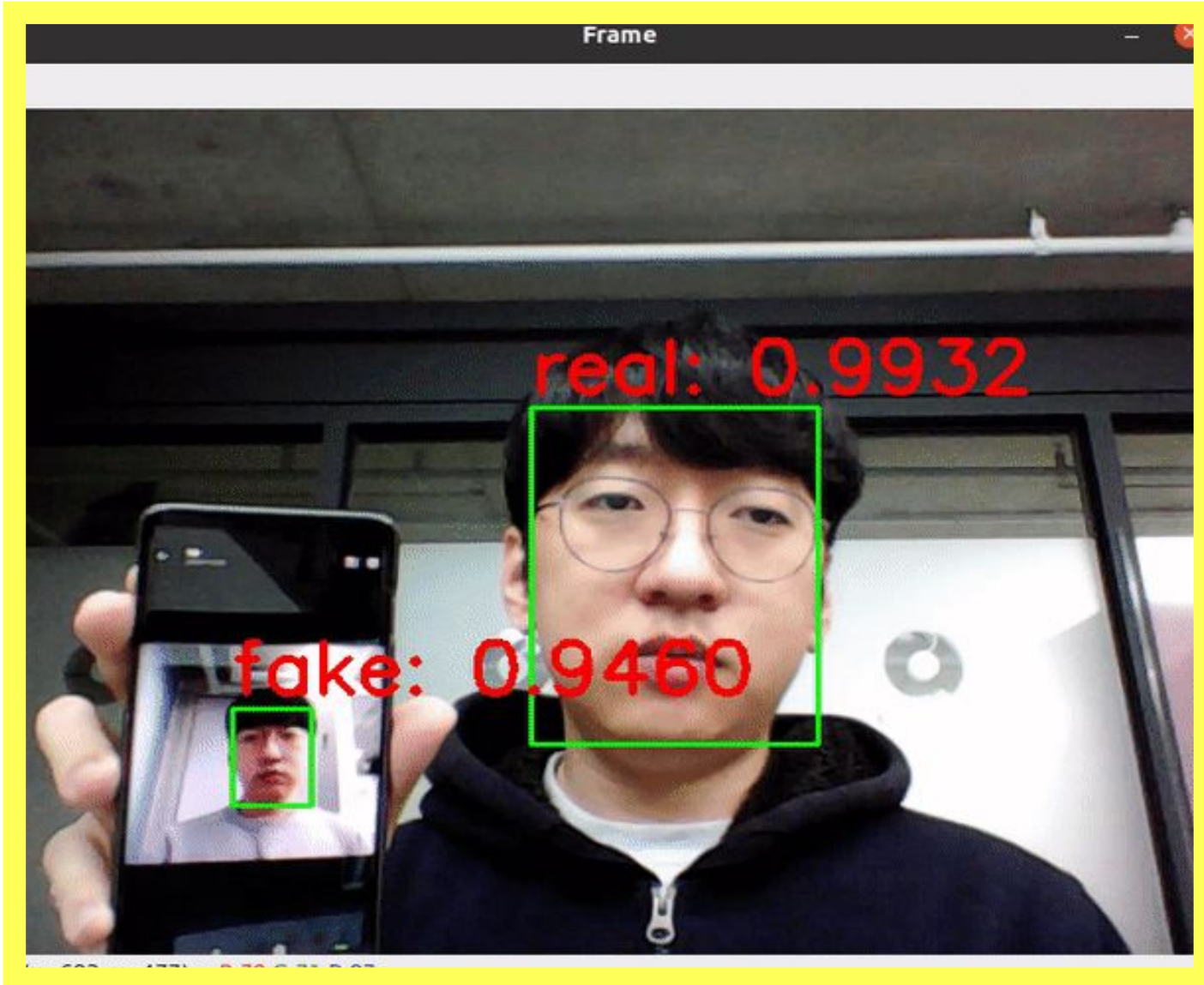


### <Triplet 학습 단계>

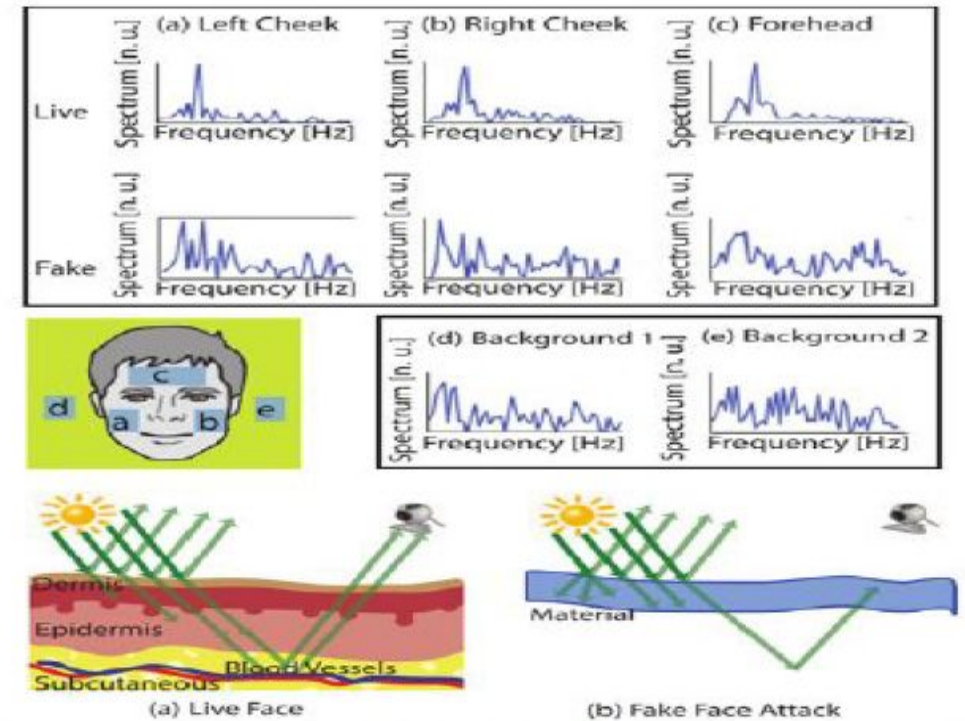


# 주요 기술

## 3. Liveness 감지 알고리즘 구현



<Live face와 Fake face의 주파수 비교>



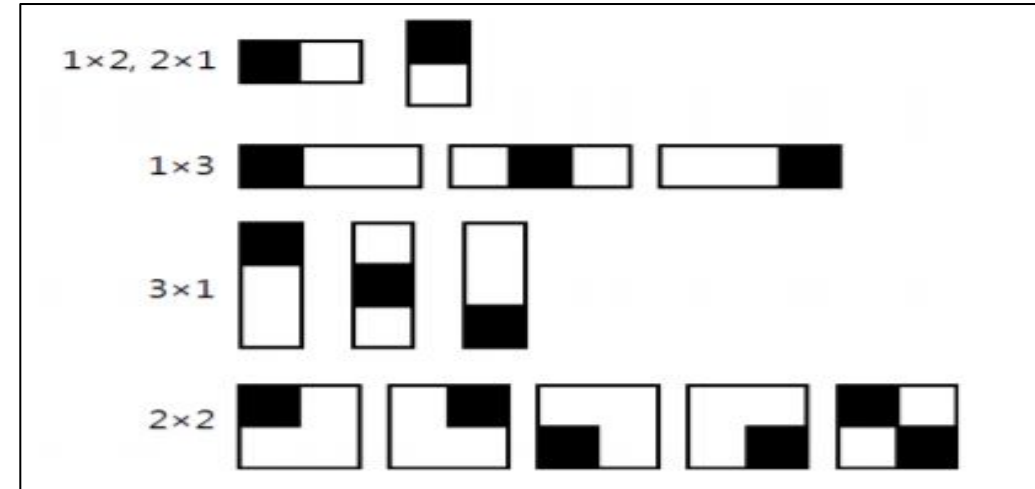
PPGSecure: Biometric Presentation Attack Detection Using Photoplethysmograms, IEEE FG 2017, 2017.



## 4. 감정인식



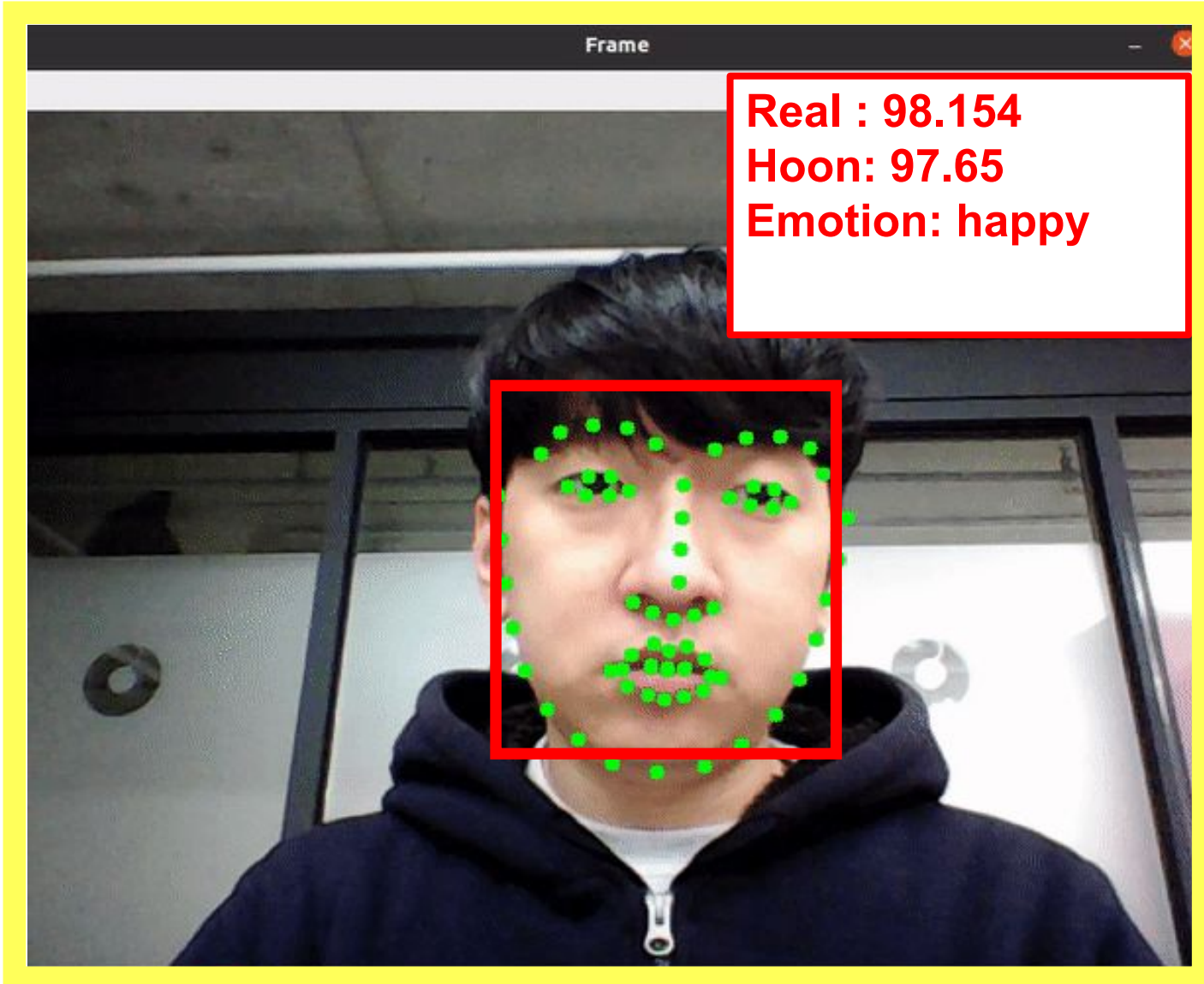
### <Haar-like Feature>



Happy



## 5. 종합 얼굴 분석 결과



## 5. OTP



# 문제 해결

- 팀원 소개
- Liveness 성능 고도화
- 장고 Restful Framework
- 딥러닝-서버 실시간 연동
- 안드로이드

# Face Guard 팀원소개



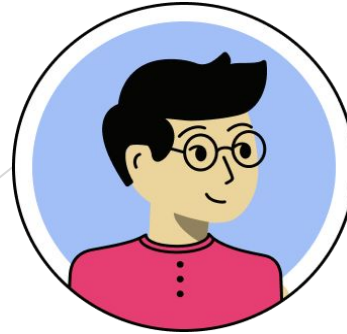
홍혜린(조장)

- ❑ 프론트엔드(App) 개발
- ❑ 딥러닝 코드 구현 및 분석
- ❑ REST API 활용 서비스 설계
- ❑ MySQL DB Schema 설계



양문일

- ❑ 선행 연구 분석 및 자료 수집
- ❑ 딥러닝 코드 구현
- ❑ 딥러닝 모델 분석
- ❑ Real / Fake 딥러닝 모델링



허 훈

- ❑ 본인 식별 딥러닝 모델링
- ❑ Real / Fake 딥러닝 모델링
- ❑ 데이터 수집
- ❑ 딥러닝 코드 구현

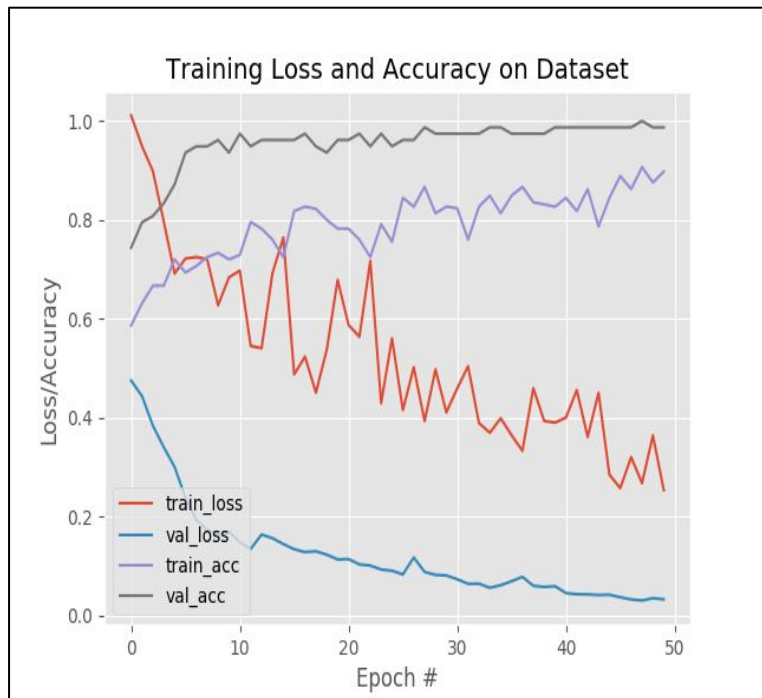


황정하

- ❑ 장고 서버 구축 및 구현
- ❑ OTP 기능 구현
- ❑ 딥러닝 코드 분석 및 구현
- ❑ REST API 활용 서비스 설계

## 1. Liveness 감지 알고리즘 성능 고도화

- 학습데이터의 수를 늘려 Real / Fake 의 분류성능을 고도화



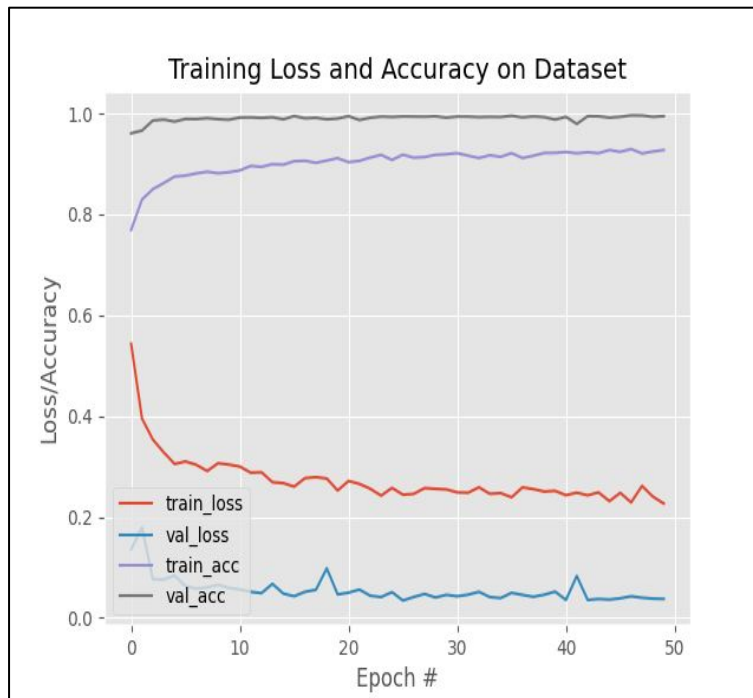
학습데이터

Real Face

300

Fake Face

300



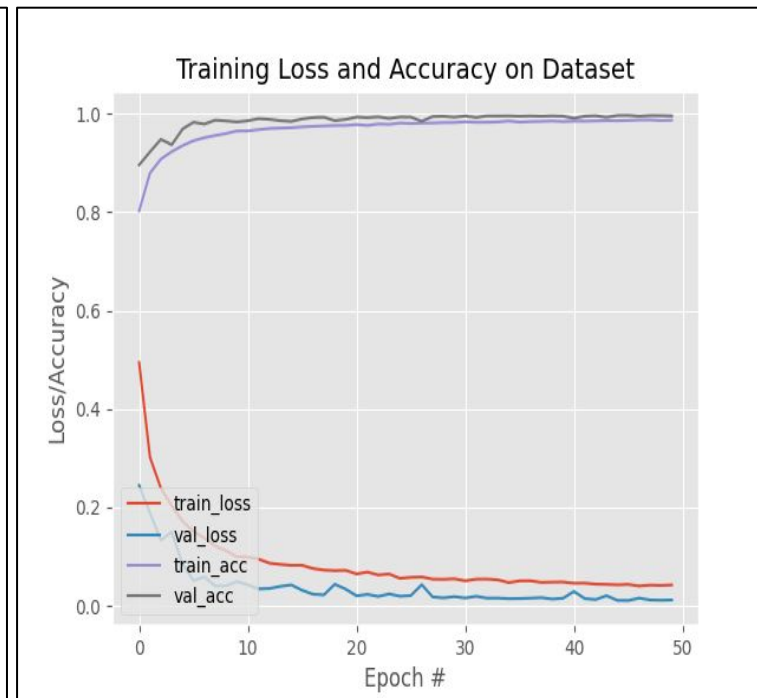
학습데이터

Real Face

5000

Fake Face

10000



학습데이터

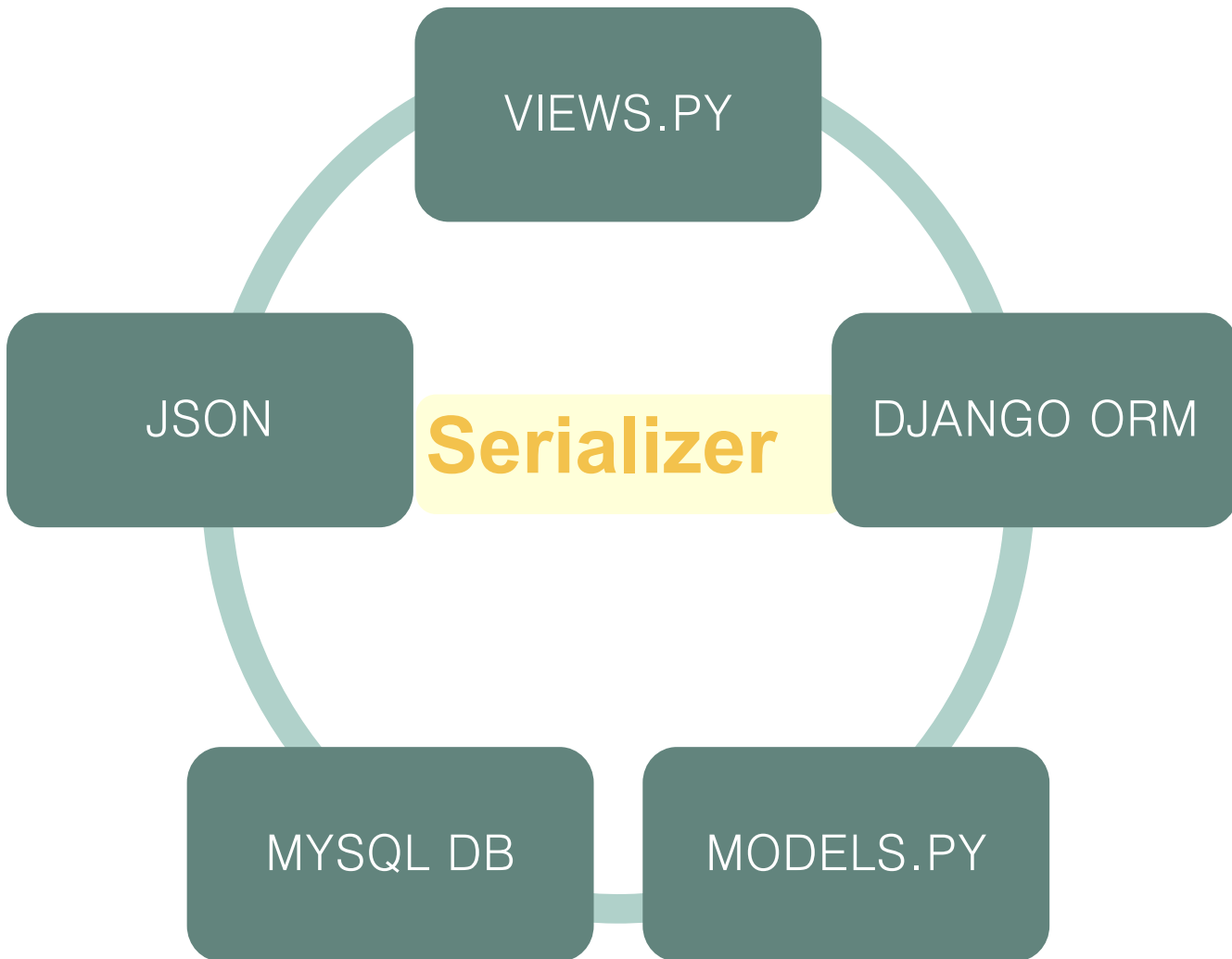
Real Face

16800

Fake Face

29222

# 문제 해결 – 장고 Restful Framework



```
from django.db import models

class Otp(models.Model):
    otp = models.OneToOneField('User', models.DO_NOTHING, primary_key=True, related_name='otp')
    sent_date = models.DateTimeField(blank=True, null=True)
    phone = models.ForeignKey('User', models.DO_NOTHING, db_column='phone', blank=True, null=True,
                             related_name='otpphone')
```

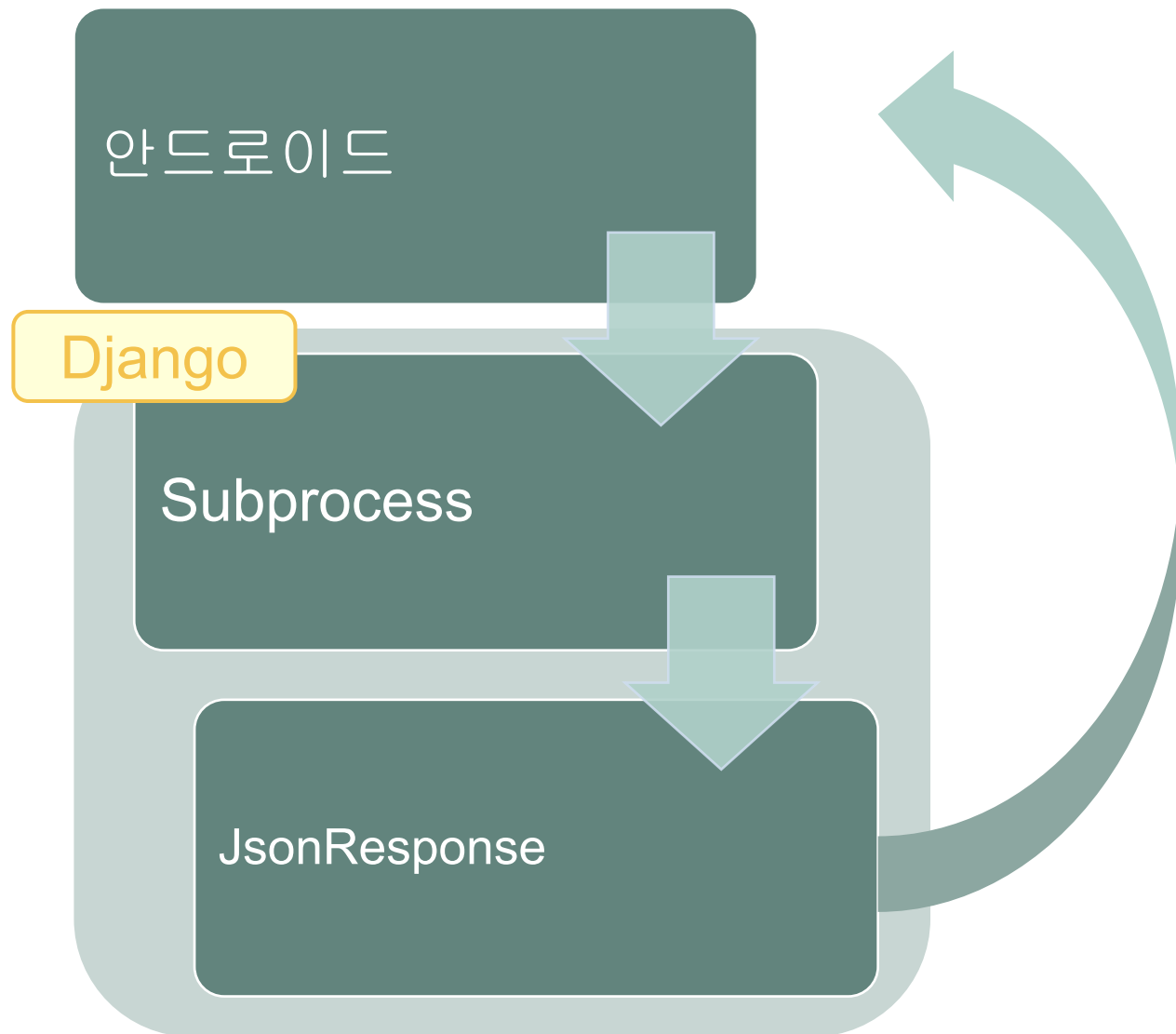
```
class Meta:
    managed = False
    db_table = 'otp'
```

```
class User(models.Model):
    user_id = models.AutoField(primary_key=True)
    account_number = models.CharField(max_length=30, blank=True, null=True)
    user_name = models.CharField(max_length=10, blank=True, null=True)
    created_date = models.DateTimeField(blank=True, null=True)
    phone = models.CharField(max_length=12)
    istrain = models.IntegerField(blank=True, null=True)
    otp_id = models.CharField(max_length=100, blank=True, null=True)
```

```
class Meta:
    managed = False
    db_table = 'user'
    unique_together = (('user_id', 'phone'),)
```



# 문제 해결 - 딥러닝 - 서버 실시간 연동



```
class RegisterView(APIView):
    parser_classes = (MultiPartParser, FormParser)

    def post(self, req, *args, **kwargs):

        file_name=str(req.data.get('upload'))
        file_object=req.data.get('upload')
        video_path = 'C:/Users/Playdata/PycharmProjects/restfulapi/restfulapiserver/addresses/train_data/' + file_name
        user_name=req.data.get('user_name')
        phone=req.data.get('phone')
        if not User.objects.filter(user_name=user_name, phone=phone):
            with open(video_path, 'wb+') as f:
                for chunk in file_object.chunks():
                    f.write(chunk)
            user = User(user_name=user_name, phone=phone, otp_id=pyotp.random_base32())
            user.save()
            new_face = Face(user=user, image_path=video_path)
            new_face.save()

            return JsonResponse({'message': '등록 완료'})

        else:
            return JsonResponse({'message': '중복된 사용자 입니다'})
```

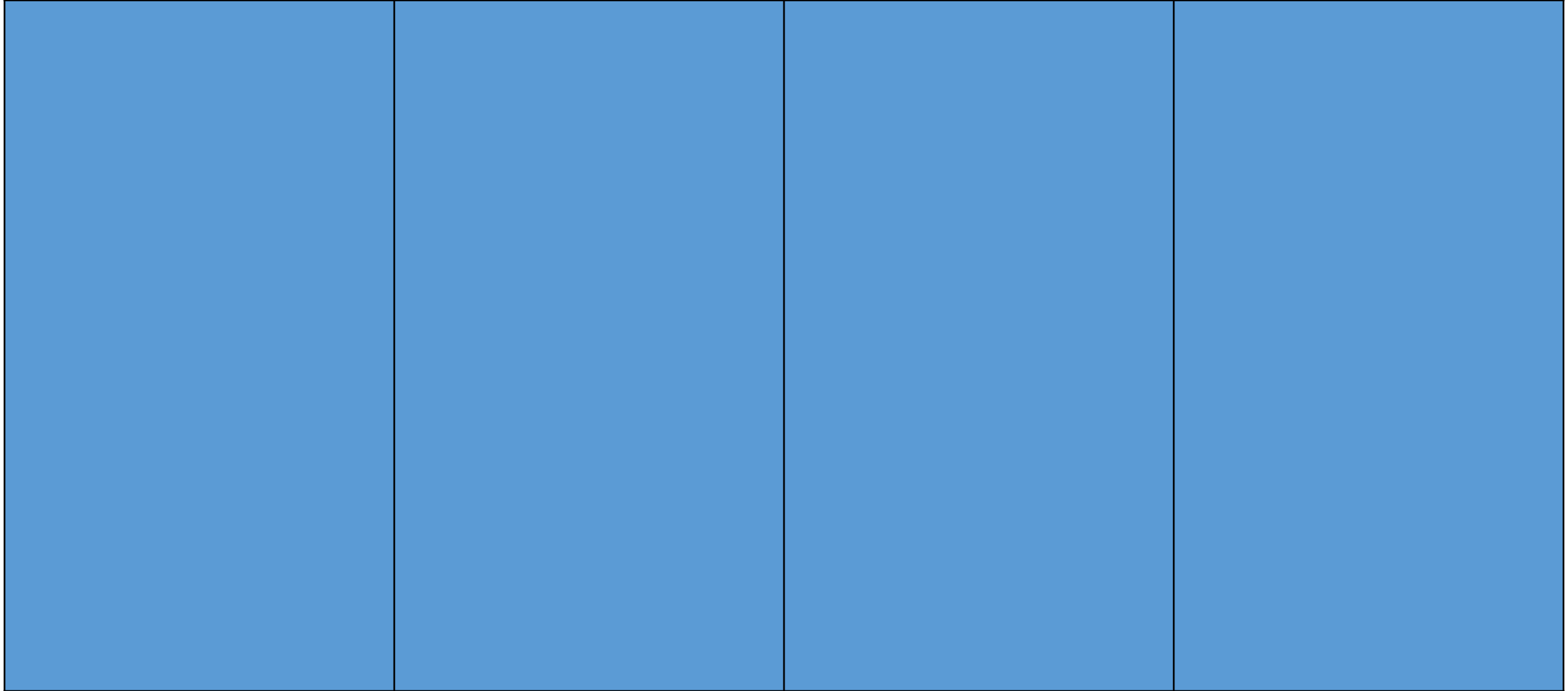
# 문제 해결 - 안드로이드

< 안드로이드 서버 >

< 디자인 >

< UI / UX >

< 개발 >





추후 업데이트 예정

1. **AWS** 서버 한계  
극복

2. 프레임워크 상품화

## 프레임워크 상품화



Q&A

감사합니다

A decorative graphic consisting of two overlapping circles. The top circle is white and the bottom circle is dark grey. Two horizontal lines, one white and one dark grey, pass through the circles. The white line is positioned higher and the dark grey line is positioned lower, creating a layered effect.

# APPENDIX

- 분야별 세부설명

산업별 생체인증 적용 가능 범위

[표] 생체인증 시장의 범위

부문	분야	세부분야
민간	금융	은행, 카드회사, 보험회사, ATM기기 등
	IT/통신	데이터센터, IT회사, 통신사업자, 인터넷서비스 제공회사
	헬스케어	병원, 의료기기, 제약사
	소비재	식음료, 패션기업, 귀금속, 전자기업
	산업재	제조, 철강, 자동차, 에너지, 석유가스, 광산, 화학, 원자력
	기타	홀도머락, 게임, 교육, 병원
공공	정부	정부의 각 부처 및 지방자치단체
	공기업	일반공기업, 금융공기업

# 주요 기술

## 1. 얼굴 랜드마크 검출

### ■ 얼굴 ROI(Region Of Image) 검출\*

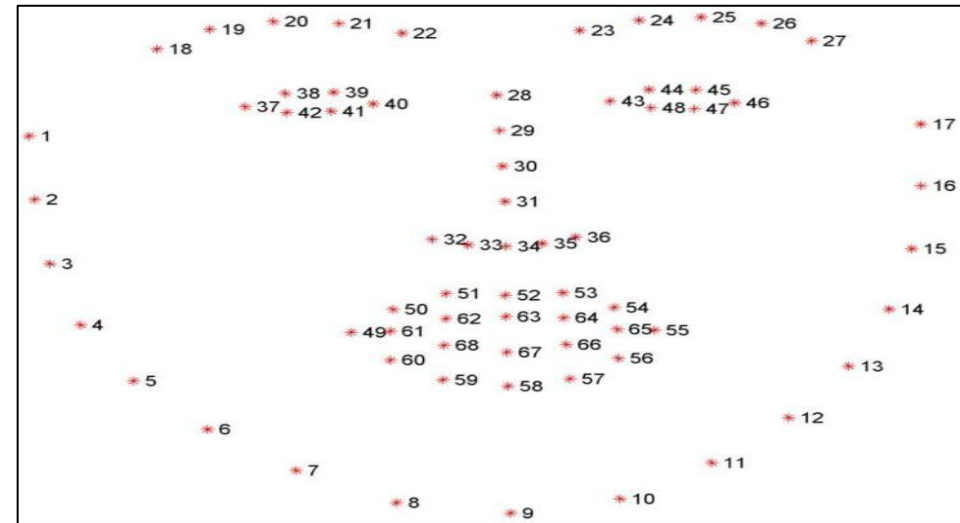
얼굴인식을 위해서는 판독 대상 이미지에서 얼굴영역을 국소화하고, 얼굴 영역내에서 주요 얼굴 구조를 감지하는 얼굴의 랜드마크를 검출해야 함.

\*얼굴 ROI 검출 : 눈, 눈썹, 코, 입, 턱 선 등의 (x,y) 좌표

### ■ 얼굴 랜드마크 감지를 위한 OpenCV와 dlib

1. 얼굴영역 국소화 및 탐지 방법
  - 1-1) OpenCV 내장된 Haar cascade 기법<sup>1)</sup>
  - 1-2) 얼굴 인식 작업을 위해 사전 훈련된 HOG + Linear SVM 객체 탐지 기법<sup>2)</sup>
  - 1-3) 얼굴 위치 파악을 위한 딥러닝 기반 알고리즘<sup>3)</sup>
2. 얼굴영역에서 얼굴 구조 감지
  - 2-1) dlib에 포함된 얼굴 랜드마크 검출 기법<sup>4)</sup>
  - 2-2) HELEN 데이터 세트의 194 포인트 모델 포함 얼굴 랜드마크 검출<sup>5)</sup>

### <얼굴 랜드마크 시각화 예시>



- 1) 머신러닝기반의 오브젝트 검출 알고리즘 - 특징(feature)을 기반으로 비디오 또는 이미지에서 오브젝트를 검출하기 위해 사용, Paul Viola and Michael Jones, Rapid Object Detection using a Boosted Cascade of Simple Features(2001)
- 2) Dalal and Triggs가 2005 년 논문에서 HOG (Histogram of Oriented Gradients) image descriptor와 Linear SVM (Support Vector Machine)을 사용하여 고도로 정확한 객체 분류기를 훈련할 수 있음을 입증, 사람 탐지에 많이 사용 <http://lear.inria.fr/people/triggs/pubs/Dalal-vpr05.pdf>
- 3) OpenCV의 딥러닝 지원, 2017년 8월에 OpenCV 3.3이 공식적으로 출시되어 개선된 심층 신경망(DNN) 모듈 제공 <https://github.com/opencv/opencv/wiki/Deep-Learning-in-OpenCV>
- 4) dlib 라이브러리에 포함된 얼굴 랜드마크 검출기는 Valid Kazemi와 Josephine Sullivan의 One millisecond face alignment with an ensemble of regression trees 구현(2014) <https://www.semanticscholar.org/paper/One-millisecond-face-alignment-with-an-ensemble-of-Kazemi-Sullivan/d78b6a5b0dca81b1fae5fb0000045a62513567> <http://blog.dlib.net/2014/08/real-time-face-pose-estimation.html>
- 5) HELEN 데이터 세트의 194 포인트 모델을 포함한 얼굴 랜드마크 검출기 <http://www.vision.ee.ethz.ch/~wongke2/helen/>
- 6) dlib 얼굴 랜드마크 검출기에 적용된 훈련된 68 포인트 BGLFW 데이터 세트의 일부 <https://lbugdocia.co.uk/resources/facial-point-annotations/>

# 주요 기술

## 2. 얼굴 인식 딥러닝(dlib + face\_recognition)

### <Triplet 학습 단계>

- 딥러닝 학습을 이용한 얼굴인식은 ‘딥 메트릭 학습’ 기술에 의해 구현

- 딥메트릭 학습을 통한 얼굴인식의 경우 임베딩(128-d 얼굴 특징 벡터) 추출

#### 1) 얼굴인식 딥러닝 학습을 위한 신경망 아키텍처는

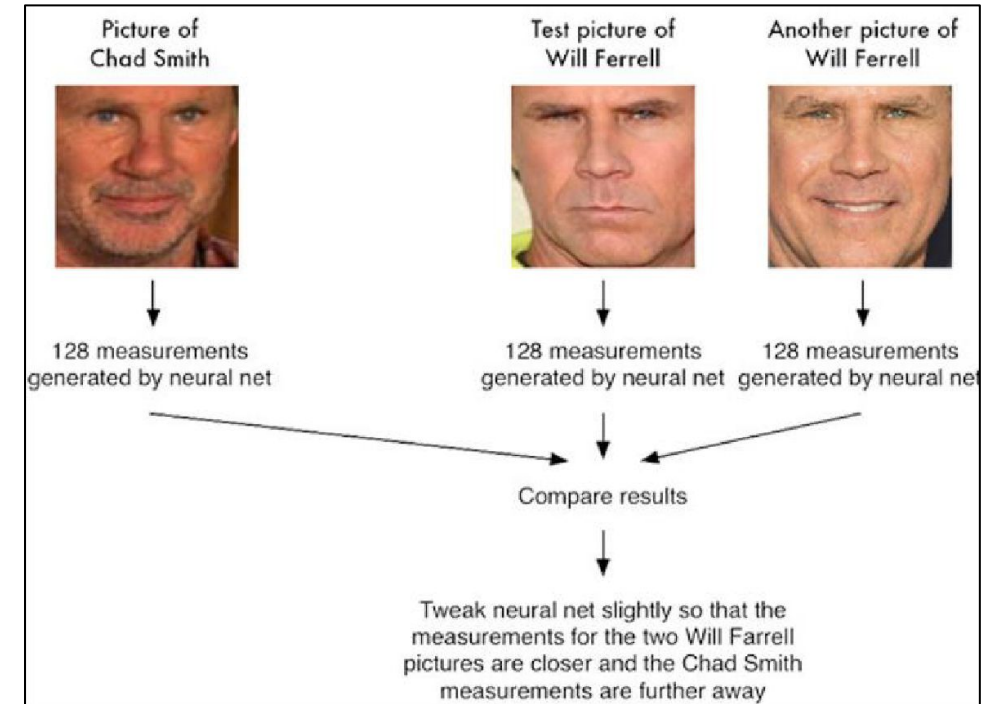
Deep Residual learning for Image Recognition 논문의 **ResNet-34**를 기반<sup>7)</sup>

- #### 2) 신경망 아키텍처는 Davis King에 의해 약 3 백만 개의 이미지 데이터<sup>8)</sup>를 학습하였고, LFW(Labeled Faces in the Wild) 데이터 세트<sup>8)</sup>에서 약 99.38%의

정확도를 보임

#### ■ Triplet training

- 딥 메트릭 학습을 통한 얼굴 인식에는 “트리플렛 훈련” 단계가 포함
- 트리플렛은 3개의 얼굴 이미지로 구성 (3명중 2명은 동일한 사람)
- Neural Network은 3개의 얼굴 이미지 각각에 대해 128-d 벡터를 생성
- 같은 사람의 두 얼굴 이미지의 경우 신경망 가중치를 조정하여 거리 매트릭스를 통해 벡터를 더 가깝게 만듦



- 각 이미지에서 신경망에 의해 **128-d 얼굴 특징 벡터를 추출하고 비교**한 결과 두 개의 이미지에 대한 측정치가 더 가까워지고 다른 하나의 이미지가 더 멀리 떨어지도록 **신경망 가중치를 조정**

7) Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Deep Residual Learning for Image Recognition (10 Dec 2015)  
<https://arxiv.org/abs/1512.03385>

8) 웹에서 수집한 13000개 이상의 얼굴 이미지가 포함된 데이터 세트 얼굴 검증을 위한 공개 벤치마크  
<http://vis-www.cs.umass.edu/lfw/>

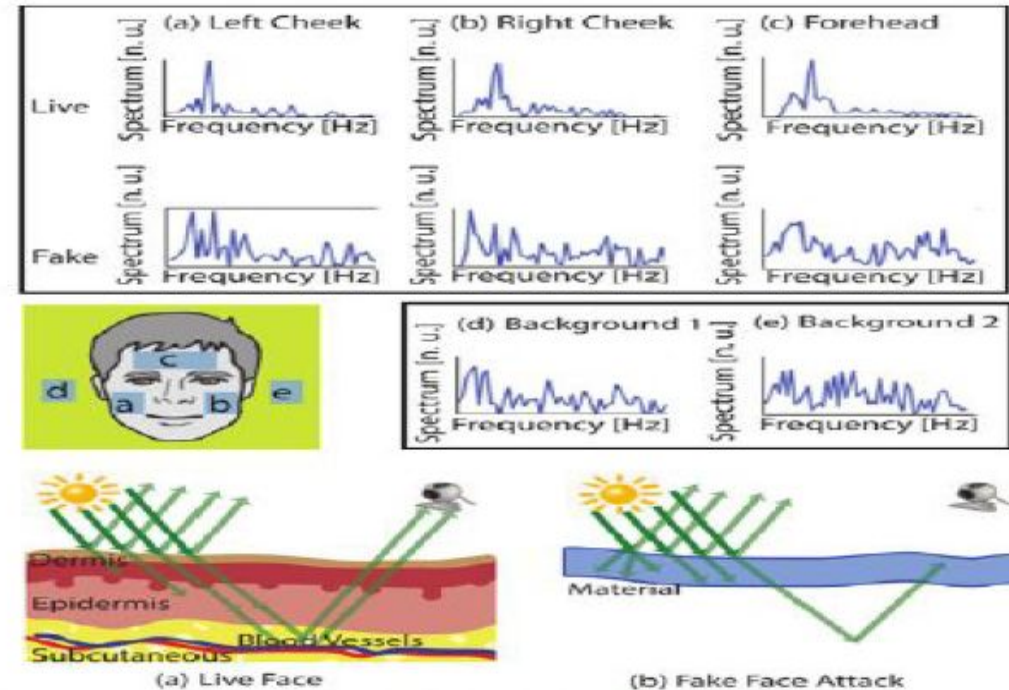
# 주요 기술

## 3. Liveness 감지 알고리즘

### ■ Frequency and Texture based analysis<sup>9)</sup>

- 얼굴 영역에서 로컬 이진 패턴 (LBP)을 계산
- SVM을 사용하여 얼굴을 실제 또는 스푸핑(Spoofing)으로 분류하는 Texture기반 분석
- 실제 얼굴과 가짜 얼굴 사이의 높은 주파수 정보를 통해 고주파의 불일치를 이용하는 기법

<Live face와 Fake face의 주파수 비교>



### ■ 광혈류 측정계 (Photoethysmogram, PPG)

- PPG는 실제 사람의 얼굴과 비디오 녹화 영상으로부터 혈류를 확인.
- 혈류로 인한 색의 변화를 감지하여 실제 얼굴과 스푸핑 공격을 구분.
- 실제 사람의 얼굴과 달리 동영상이나 인화된 사진의 경우 혈관이 보이지 않음.
- 때문에 색의 변화가 실제 사람의 얼굴과 다름을 이용하여 구분.

PPGSecure: Biometric Presentation Attack Detection Using Photoplethysmograms, IEEE FG 2017, 2017.

9) [G. Kim, S. Eum, J. K. Suh, D. I. Kim, K. R. Park, and J. Kim, Face liveness detection based on texture and frequency analyses, 5th IAPR International Conference on Biometrics (ICB), New Delhi, India, pp. 67-72, March 2012.]



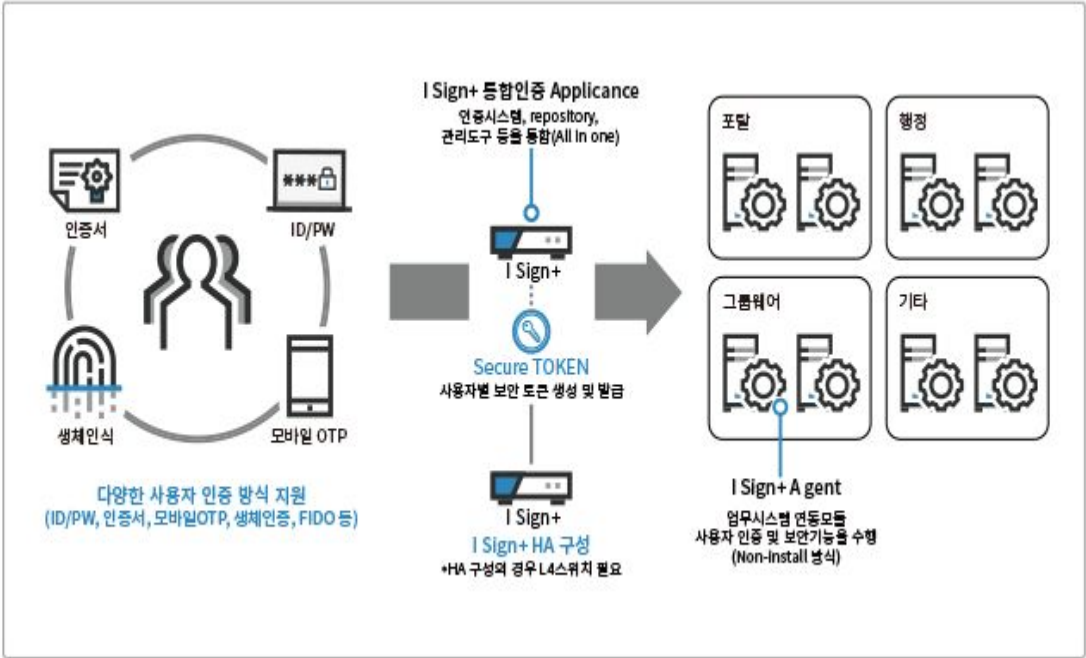
# 주요 기술

## 4. OTP(One Time Password)

- **OTP(One Time Password) 유형과 장\*단점<sup>10)</sup>**
  - 1회에 한해 사용할 수 있는 비밀번호로 매번 다른 비밀번호를 이용하여 사용자를 인증하는 방식

매체	장점	단점
난수표	<ul style="list-style-type: none"><li>기존 시스템을 활용한 구현 용이</li><li>난수표 발급 비용 저렴</li><li>카드 형태로 휴대 간편 및 사용 용이</li></ul>	<ul style="list-style-type: none"><li>난수표에 대한 복제 용이</li><li>난수표 코드표(PassPhrase)에 대한 경우의 수 제한적(ex.35개)</li><li>PassPhrase 유추 상대적 용이</li></ul>
클라이언트용 SW프로그램	<ul style="list-style-type: none"><li>PassPhrase 유추 불가</li><li>HW기기방식에 비해 구축 비용 저렴</li><li>개별 PC 설치에 따른 분실 위험 없음</li><li>PIN입력을 통한 추가 인증 구현(Two factor 인증)</li></ul>	<ul style="list-style-type: none"><li>PIN값을 알고 있을 경우 시스템 접속 가능</li><li>S/key방식의 경우 상대적으로 암호화 수준 낮음(MD4)</li><li>사용방법이 타 방식에 비해 어려움</li></ul>
HW Token	<ul style="list-style-type: none"><li>PassPhrase 유추 불가</li><li>사용자별 고유 인증기기 사용을 통한 추가 인증 구현(Two factor 인증)</li><li>PIN의 별도 입력이 필요 없음(PIN 필요 기기 존재)</li></ul>	<ul style="list-style-type: none"><li>HW기기 분실에 대한 위험 높음</li><li>보안카드에 비해 휴대 불편</li><li>HW기기 구입 및 시스템 구축 비용 고가</li><li>건전지 수명에 따른 기기 변경 필요</li></ul>
MOTP	<ul style="list-style-type: none"><li>PassPhrase 유추 불가</li><li>사용자별 고유 인증기기 사용을 통한 추가 인증 구현(Two factor 인증)</li><li>시간 동기화 용이</li><li>분실위험이 HW Token방식에 비해 상대적으로 적음</li></ul>	<ul style="list-style-type: none"><li>전용 휴대폰 필요(ex. SKT(SKVM,WIPI ), KTF(Brew), LGT(JavaStation)</li><li>시스템 구축 비용이 타 방식에 비해 높음</li></ul>
E-Mail	<ul style="list-style-type: none"><li>불특정 다수 사용자 이용 용이</li><li>OTP이용을 위한 별도의 모듈이 필요하지 않음</li><li>타 방식에 비하여 중앙집중 관리 용이 및 사용자 조작 용이</li></ul>	<ul style="list-style-type: none"><li>모든 사용자가 메일 전송/회신을 위한 인프라 환경이 구축되어야 함</li></ul>
SMS	<ul style="list-style-type: none"><li>불특정 다수 사용자 이용 용이 및 네트워크 환경에 영향을 받지 않음</li><li>OTP이용을 위한 별도의 모듈이 필요하지 않음</li><li>타 방식에 비하여 중앙집중 관리 용이 및 사용자 조작 용이</li></ul>	<ul style="list-style-type: none"><li>핸드폰 발송을 위한 시스템 구축 비용 및 문자 메시지 전송 시 지속적인 비용 발생</li></ul>

### ■ OTP 인증 시스템 구성



- FIDO와 모바일 OTP 인증 매커니즘 및 시스템 설계<sup>11)</sup>

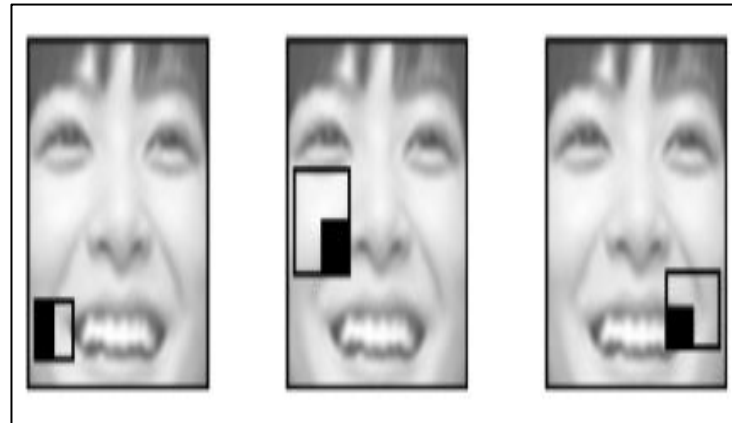
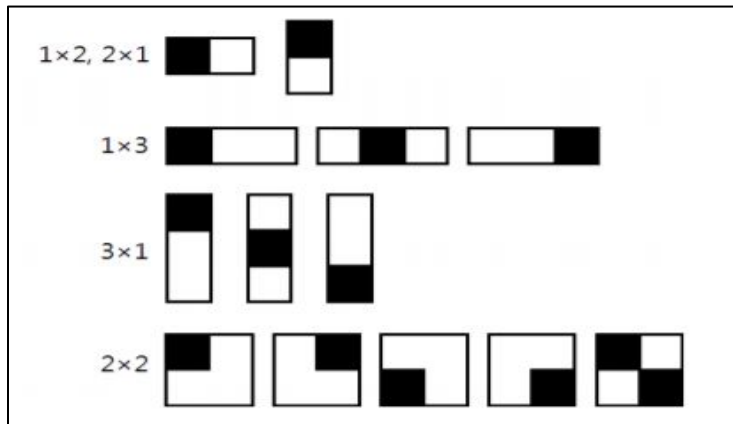
10) <http://wiki.wikisecurity.net/wiki.otp>  
11) <https://www.pentasecurity.co.kr/multi-factor-authentication/>

# 주요 기술

## 5. 감정인식(Emotion Ai)

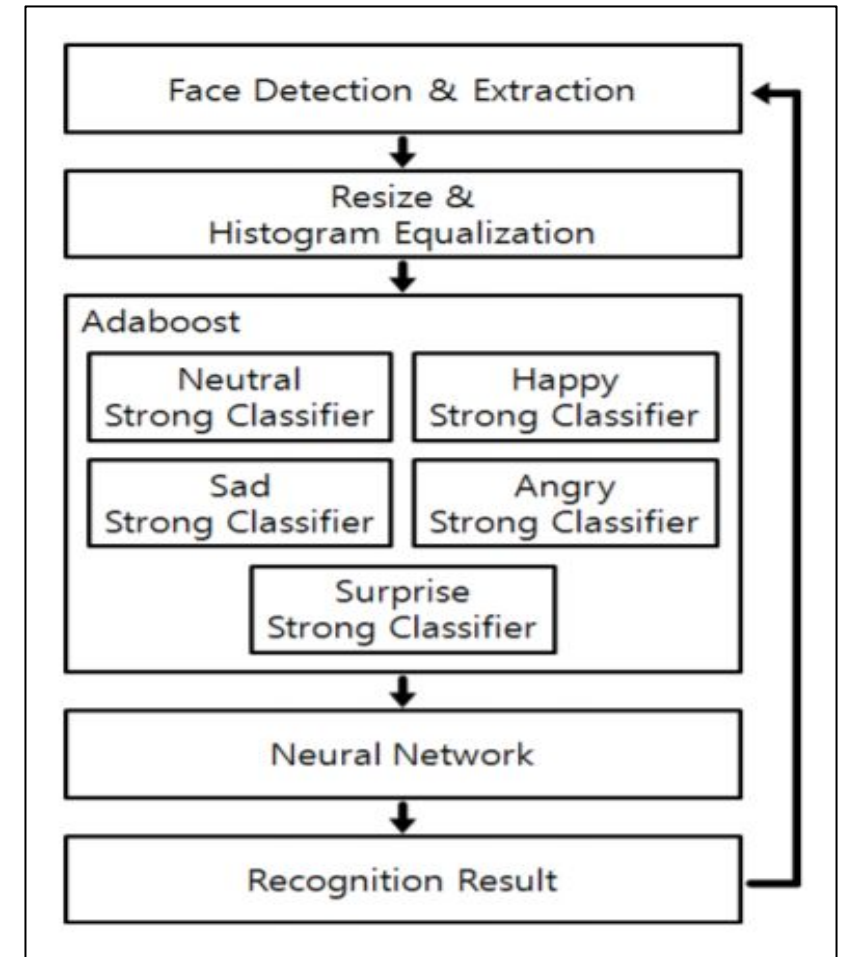
### ■ Haar-like Feature

- 얼굴의 명암을 이용하여 감정패턴을 구함.
- 사람 얼굴 위에 흑백의 사각형을 겹쳐 놓음.
- 밝은 영역에 속한 픽셀 값들의 평균에서 어두운 영역에 속한 픽셀값들의 **평균의 차이**를 구함.
- 그 차이가 임계점을 넘으면 얼굴에 대한 **Haar-like Feature**가 됨.



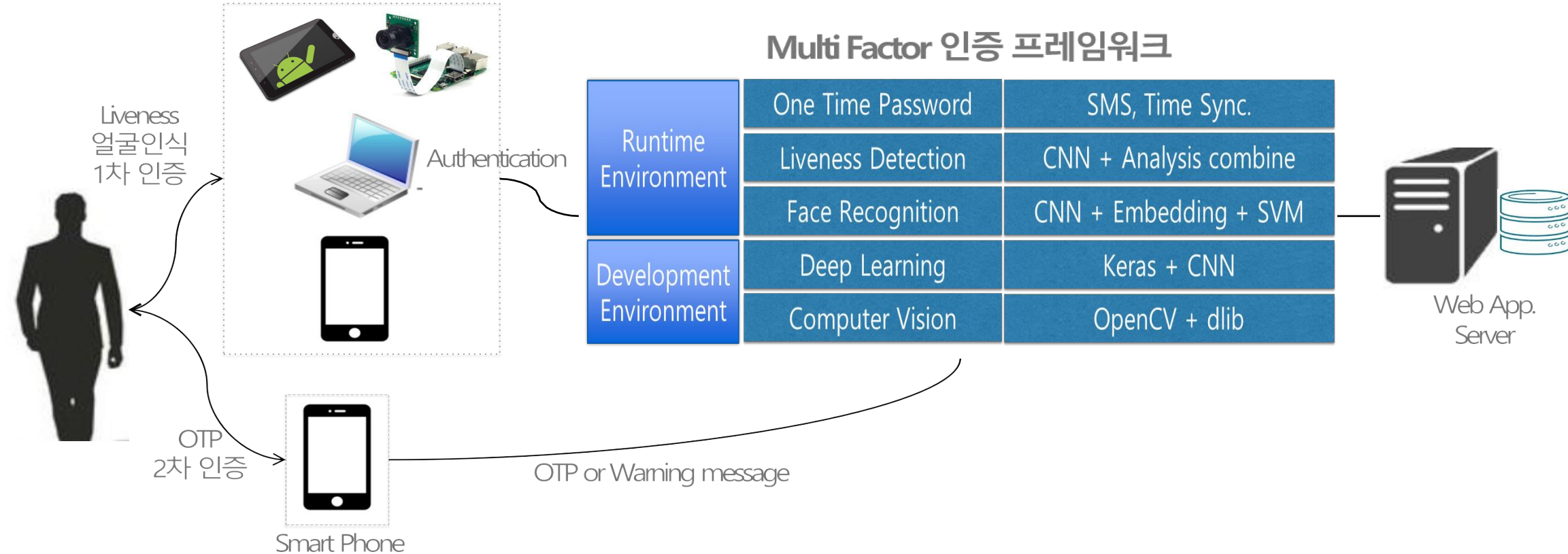
홍용희, 한영준, 한헌수. (2010). 에이다부스트와 신경망 조합을 이용한 표정인식. 한국지능시스템학회 논문지, 20(6), 806-813.

### ■ 표정인식 시스템



# 시스템 아키텍처

- 딥러닝 학습을 통한 얼굴인식, 스푸핑 방지 위한 Liveness 학습 알고리즘을 융합하여 1차 인증(Liveness 얼굴인식) 수행.
- 식별된 고객의 스마트가기를 활용하여 OTP발급 및 검증을 통해 2차 인증(OTP) 수행.



<Spoofing 대응 위한 얼굴인식 시스템 아키텍처>

THE END

감사합니다