

# 커스텀 ResNet50 모델을 이용한 한국인 표정 인식기

Facial Expression Recognition using custom ResNet50

---

송기태 ([kitae040522@gmail.com](mailto:kitae040522@gmail.com))

Dankook Univ. Software High School (AI Techniques)

# Content

Facial Expression Recognition Task에 대해서...

ResNet50 모델이란?

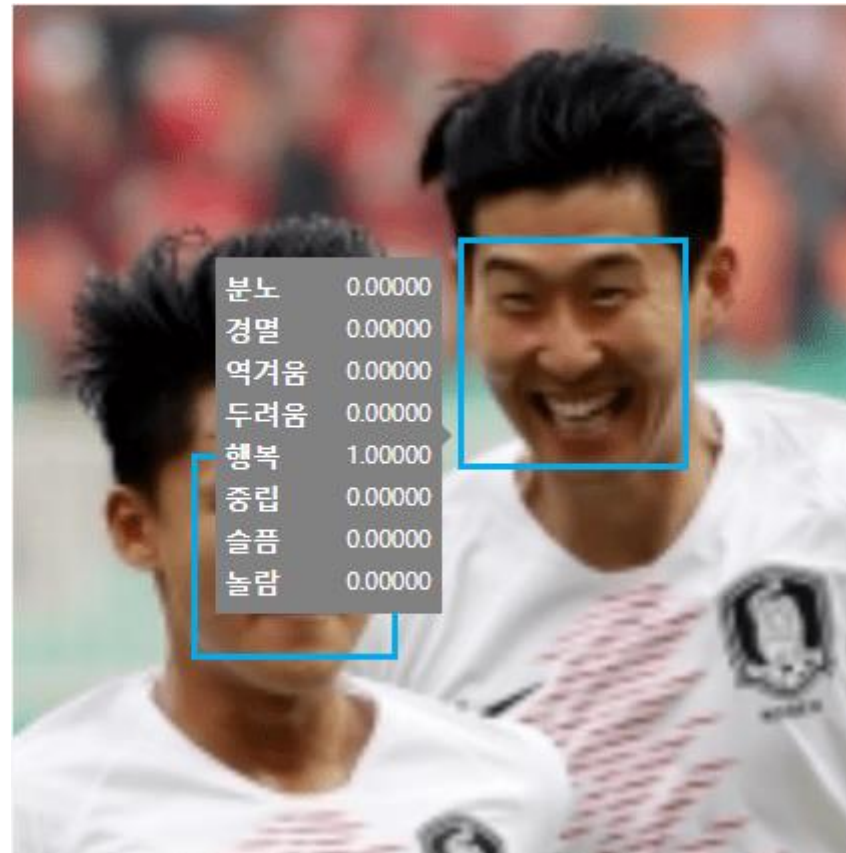
데이터셋 구성

커스텀 ResNet50 모델 구현

얻은 인사이트 공유

# Facial Expression Recognition Task에 대해서...

얼굴 이미지의 표정을 분노, 두려움, 놀람, 슬픔, 행복 등 다양한 범주로 Classification하는 Task



# Facial Expression Recognition Task에 대해서...

짜증



당황



행복



중립

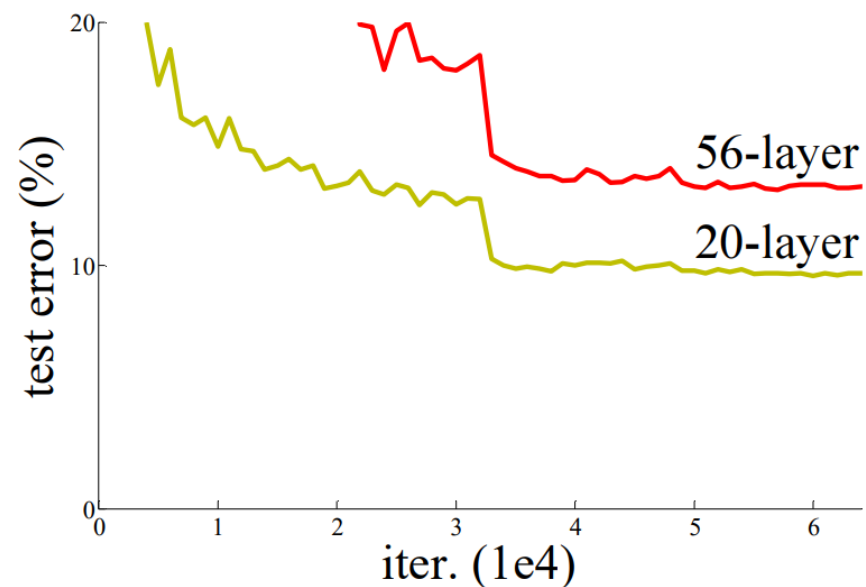
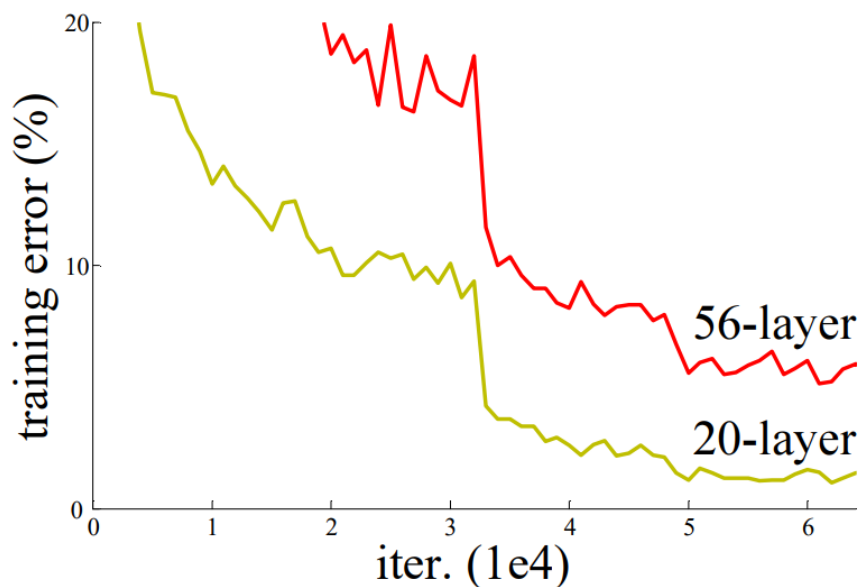


슬픔



# ResNet50 모델이란?

Q. Is learning better networks as easy as stacking more layers?



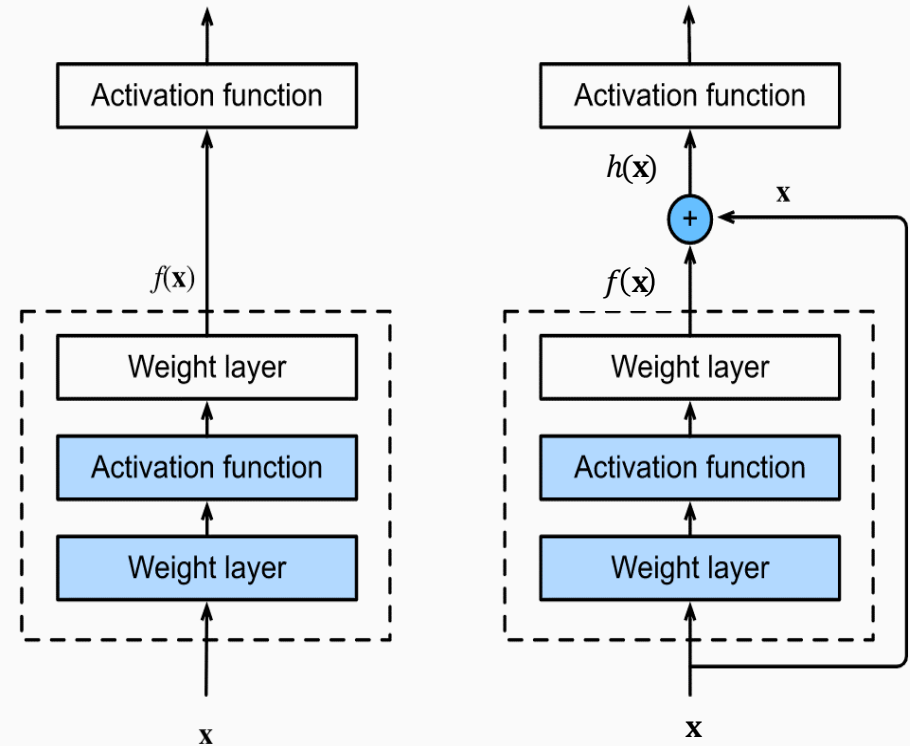
기존의 방식으로 Layer을 깊게 짤 때, 오히려 성능이 낮았음. → 새로운 알고리즘을 찾자.

# ResNet50 모델이란?

## 핵심 아이디어

- 깊은 네트워크로 학습하기 위해 잔류 학습(Residual Learning)을 제안함.
- 적당한  $h(x)$ 를 구하는 것이 아니라  $f(x) + x$ 를 학습하자.
  - Gradient Vanishing 문제를 해결하면서 레이어를 깊게.

Gradient Vanishir

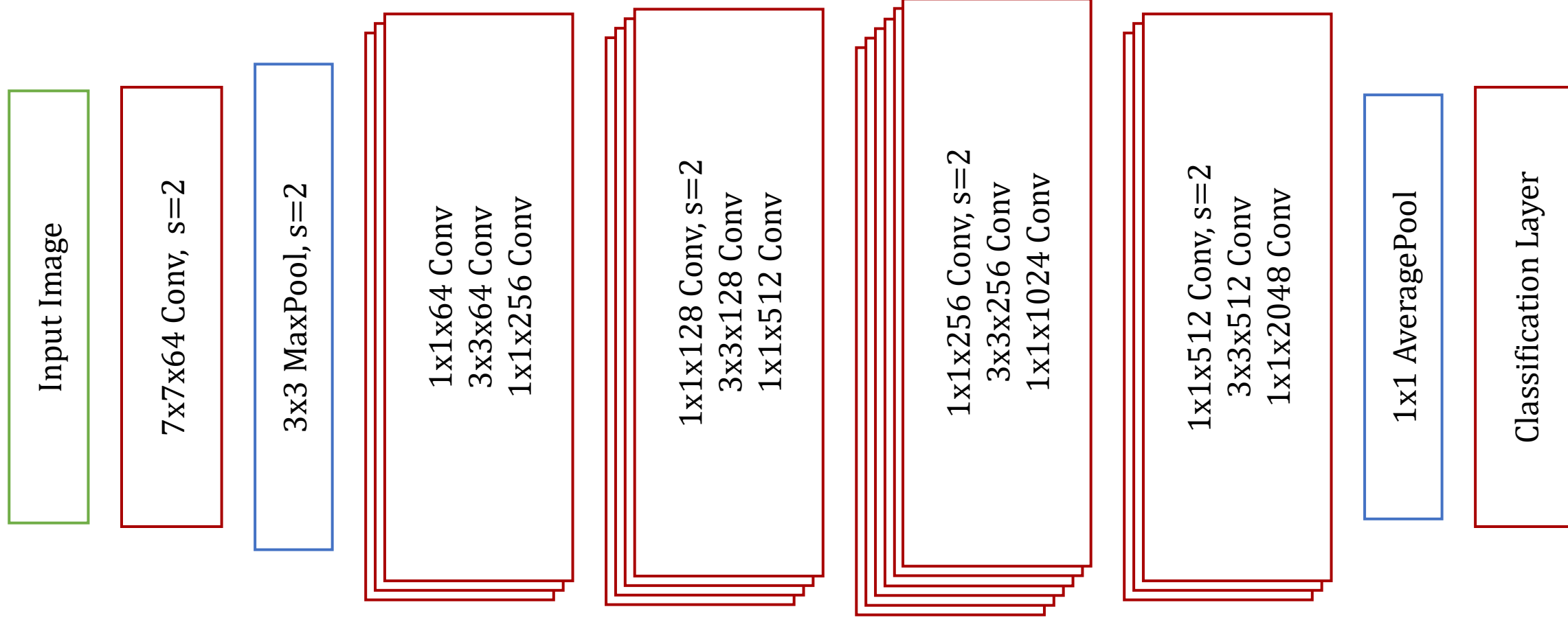


## ResNet50 모델이란?

Q. 어떻게 Residual Block을 사용해 Gradient Vanishing Problem을 해결할 수 있는가?

A.  $\frac{\partial f}{\partial x} \cong 0$  이라도, 1을 넘겨주기 때문에 Gradient Vanishing Problem을 해결할 수 있다.

# ResNet50 모델이란?





# 데이터셋 구성 ( 데이터셋 출처 )

## 개방 데이터

개방 데이터 > 비전 > 한국인 감정인식을 위한 복합 영상 소개

## 한국인 감정인식을 위한 복합 영상 소개

소개

다운로드

데이터셋명	한국인 감정인식을 위한 복합 영상		
데이터 분야	비전	데이터 유형	이미지
구축기관	한국과학기술원	데이터 관련 문의처	담당자명 박주용(한국과학기술원)
가공기관	메트릭스리서치, 소리자바, 데이터헌트		전화번호 042-350-2924
검수기관	메트릭스리서치, 소리자바, 데이터헌트		이메일 juyongp@kaist.ac.kr
구축 데이터량	50만	구축년도	2020년
버전	1.1	최종수정일자	2022.01.18
소개	장면의 맥락 정보를 통한 감정 분석을 위한 얼굴 표정 이미지 데이터		
주요 키워드	감정 분석, 표정 분석		
저작권 및 이용정책	본 데이터는 과학기술정보통신부가 주관하고 한국지능정보사회진흥원이 지원하는 '인공지능 학습용 데이터 구축사업'으로 구축된 데이터입니다. [데이터 이용정책 상세보기]		
데이터설명서	<a href="#">자료보기</a>	구축활용가이드	<a href="#">자료보기</a>
샘플데이터	<a href="#">다운로드</a>	교육활용동영상	<a href="#">영상보기</a>
저작도구	<a href="#">다운로드</a>	AI모델	<a href="#">다운로드</a>

인공지능 학습용 데이터  
다운로드 프로그램 설치

Windows & Mac용

> 간단 사용설명서  
> 맥용 설치&삭제 가이드

Ubuntu Ver.18.04용

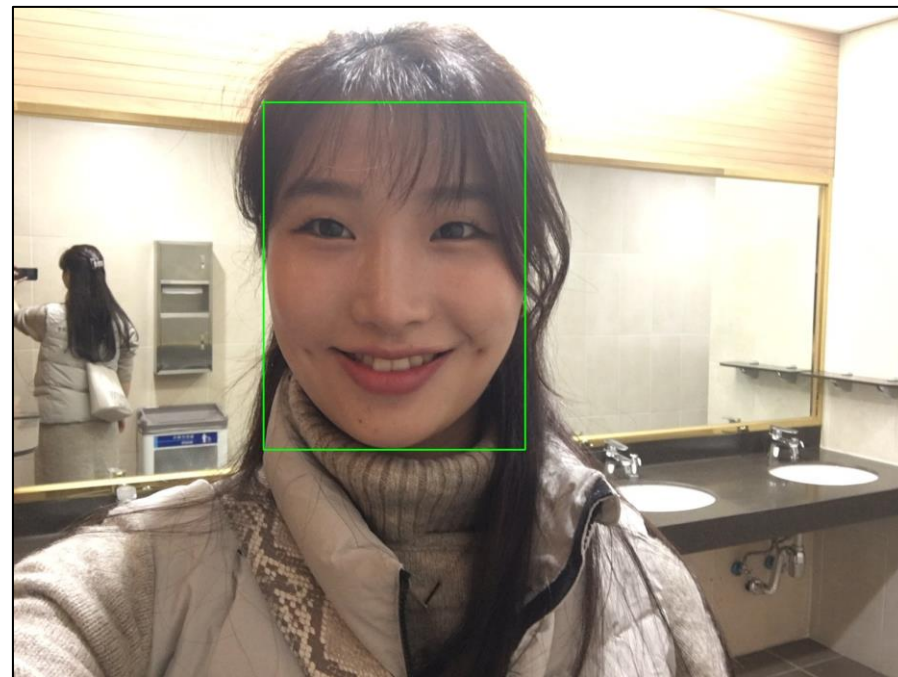
> 간단 사용설명서



(타이틀)\_(성별)\_(나이)\_(감정)\_(날짜).jpg

## 데이터셋 구성 ( Face Detect )

```
import cvlib as cv  
faces, conf = cv.detect_face(image)
```



## 데이터셋 구성 ( Face Crop & IMG Resize & Grayscale)

```
img = cv2.resize(roi, (224, 224), interpolation=cv2.INTER_CUBIC)  
img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
```



# 데이터셋 구성 ( IMG Augmentation )

기존의 데이터에 약간의 변형을 가해 새로운 데이터셋을 만듦 → 과잉적합 방지, 모델 정확도 상승

원본 데이터



Augmentation



# 데이터셋 구성 ( IMG Augmentation )

```
from keras.preprocessing.image import ImageDataGenerator

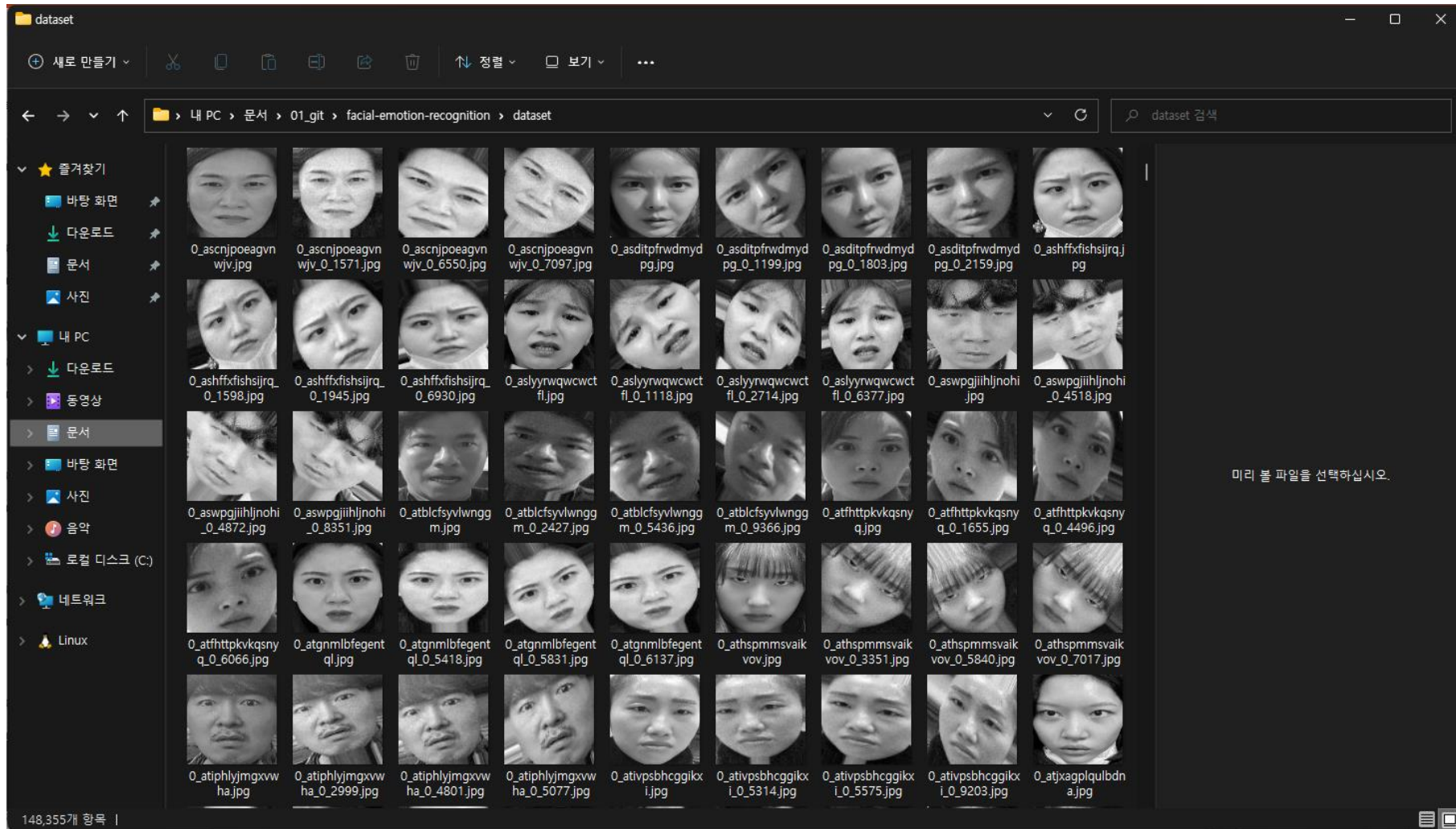
def random_noise(x):
    x = x + np.random.normal(size=x.shape) * np.random.uniform(1, 5)
    x = x - x.min()
    x = x / x.max()

    return x * 255.0

ImageDataGenerator(
    rotation_range=40,
    width_shift_range=0.07,
    height_shift_range=0.07,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True,
    fill_mode='nearest',
    preprocessing_function=random_noise
)
```



# 데이터셋 구성 ( Result )



## 커스텀 ResNet50 모델 구현 ( 가중치 초기화 알고리즘 선택 )

*glorot unifom:  $\text{unif}(-limit, +limit), limit = \sqrt{\frac{6}{fan\ in + fan\ out}}$*

*he normal:  $\text{normal}(mean=0, stddev), stddev = \sqrt{\frac{2}{fan\ in}}$*

Glorot은 tanh 활성화 함수를 위해 고안된 거라고 하는데, **ReLU에 적용해도 큰 차이는 없지 않을까?\***

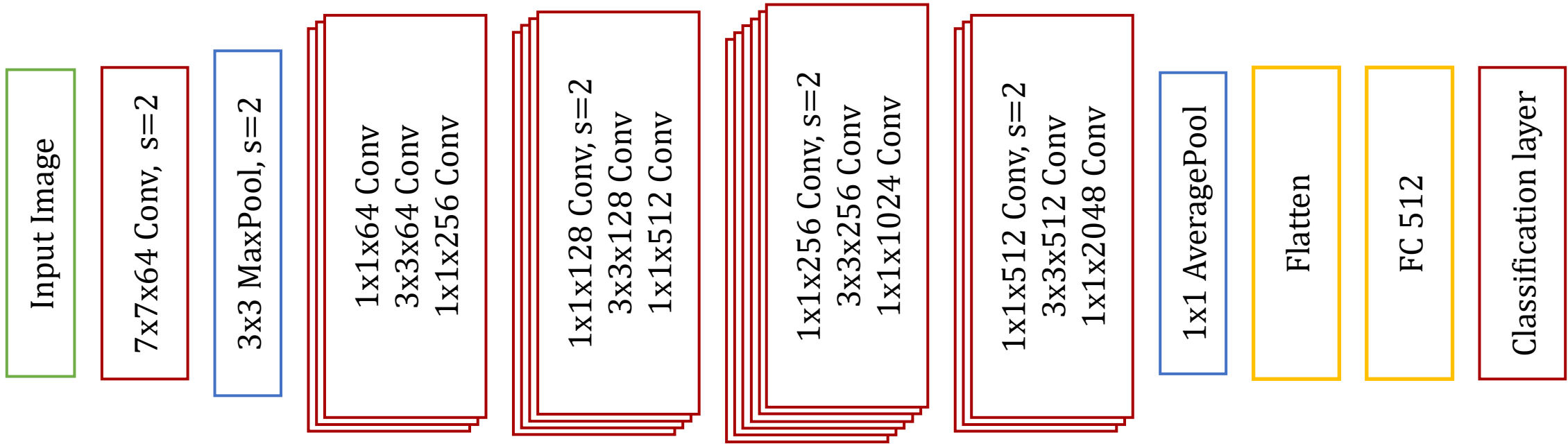


\*얼은 인사이트에서 계속...

# 커스텀 ResNet50 모델 구현 ( Linear Layer 추가 )

파라미터 개수가 늘어나 연산량이 많아진다고 해도, 필요하다고 생각함.

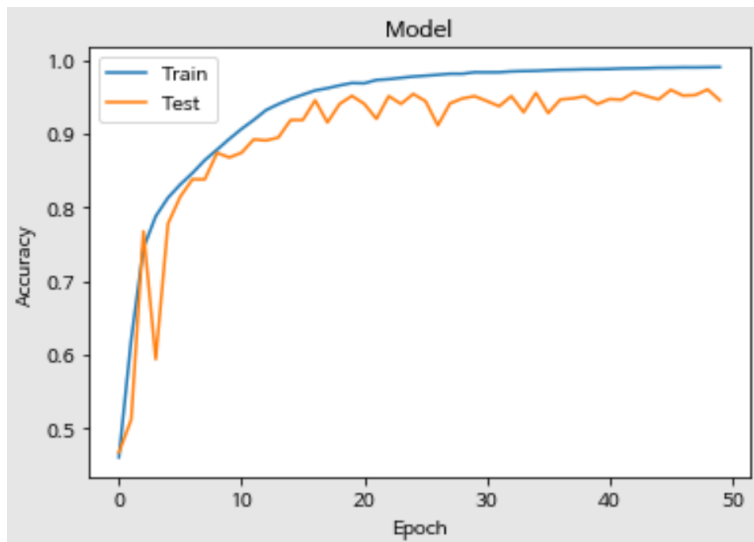
```
X = keras.layers.Flatten()(X)
X = keras.layers.Dense(units=512, activation='relu')(X)
```



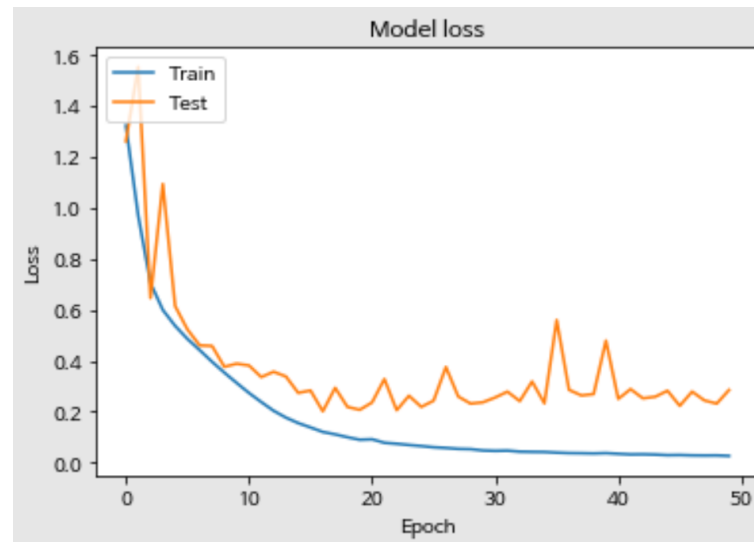


# 커스텀 ResNet50 모델 구현 ( 학습 결과 )

정확도



오차율



36 Epoch과 39 Epoch에서 과잉적합 현상을 보였지만, 다시 회복하는 모습을 보임.

최종: loss: 0.0303 - accuracy: 0.9900 - val\_loss: 0.2233 - val\_accuracy: 0.9598

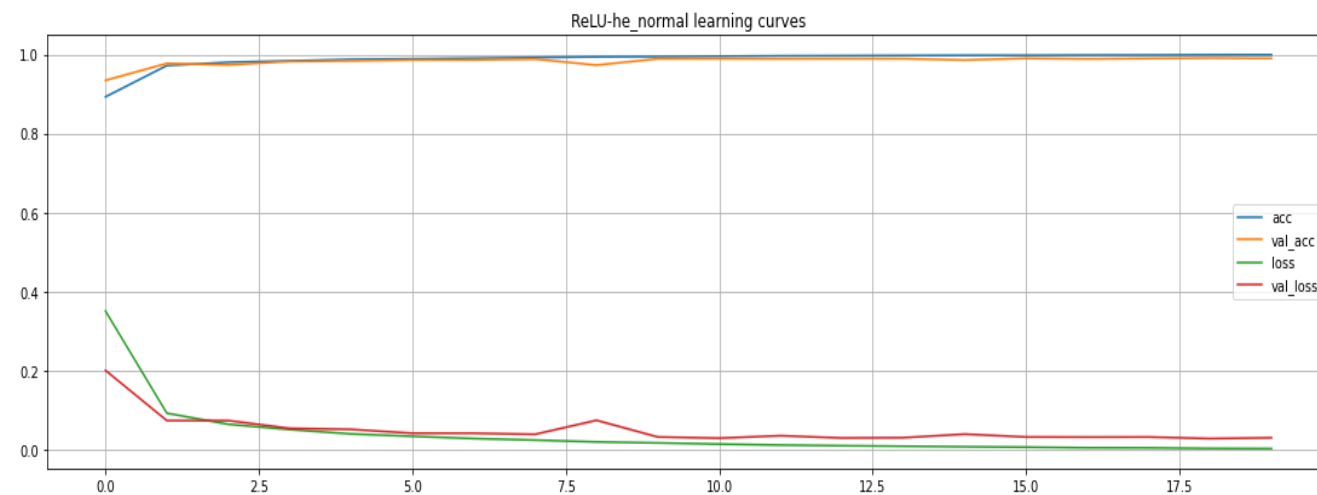
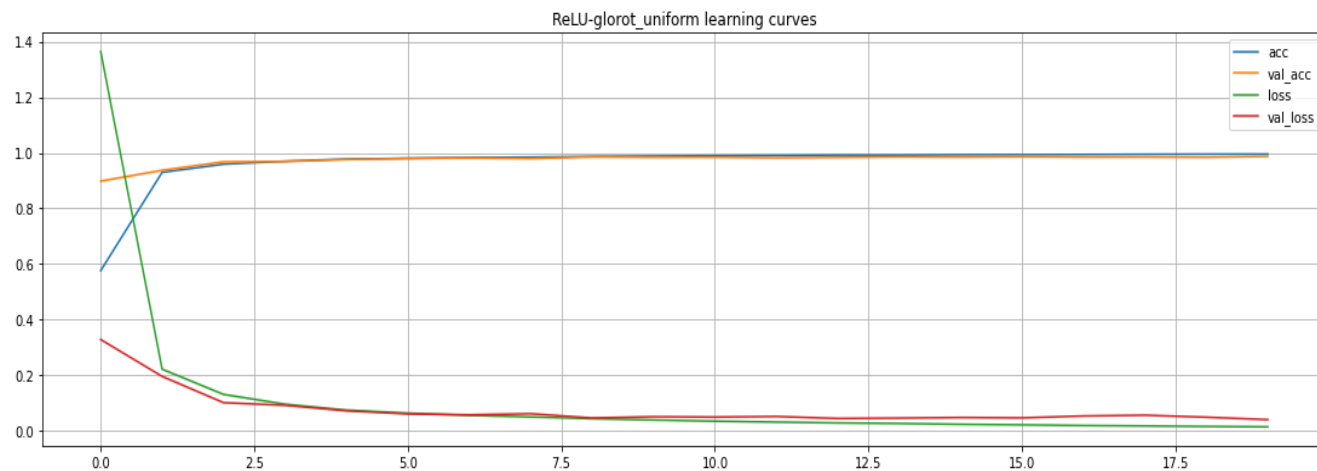
# 커스텀 ResNet50 모델 구현 ( 학습 결과 )



학습에 관여하지 않은 Test Dataset에서는  
acc\_0.88, loss\_0.6를 기록함

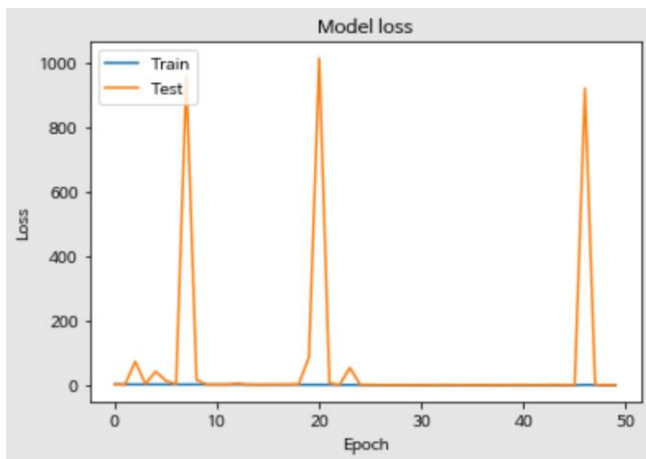
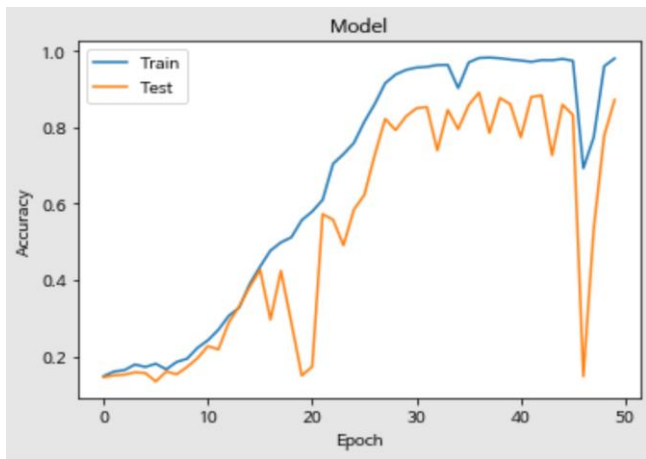
# 얻은 인사이트 공유 ( 가중치 초기화 알고리즘 )

논문을 의심하지 말자...  
앞으로 ReLU는 he\_normal로 고정



# 얻은 인사이트 공유 ( Linear Layer 추가에 대해 )

Layer 추가 전



Layer 추가 후

