



Project. FDC

CONTENTS

01 Project Overview

- 배경 및 목표
- 수행일정
- 기술스택
- 데이터

02 EDA & Data Preprocessing

- EDA
- 데이터 전처리

03 Data Modeling

- 이미지 인식
- 성별 분류 모델
- 나이 분류 모델

04 Web Service

- 웹 사용자
인터페이스
- 시연

05 Conclusion

- 향후 과제
- 느낀 점

Part 01

Project Overview

Background

최근 딥러닝 및 머신러닝 기술
발전에 따라 보안, 헬스케어,
마케팅 등 다양한 분야에서
얼굴 인식 및 판별 기술이
요구되고 있음

대부분의 모델이 서양인 얼굴
기반으로 개발되어 있으므로,
한국인 안면 데이터셋을 활용해
학습모델을 구현해보고자 함



Goal

한국인의 얼굴 특성에
최적화된 얼굴 인식 모델 개발
(성별 및 나이 분류)

사용자 친화적인 UI/UX 제공

Part 01 프로젝트 일정

2023. 9. 18. ~ 2023. 10. 13. (4주)

Sep							Oct						
S	M	T	W	T	F	S	S	M	T	W	T	F	S
17	18	19	20	21	22	23	1	2	3	4	5	6	7
	주제선정 및 데이터수집				데이터 탐색			모델학습 및 검증					
24	25	26	27	28	29	30	8	9	10	11	12	13	14
	모델링 및 모델학습							웹 서비스 구현		PPT작성			



TensorFlow



데이터 전처리 및 모델링: python, tensorflow, pytorch, jupyter

웹서비스 구현: django

데이터 관리: dbeaver

버전 관리 및 협업: git, github

구분	PC	Google Colab
Processor	Intel(R) Core(TM) i5-9400F CPU @ 2.90GHz 2.90 GHz	Intel(R) Xeon(R) CPU @ 2.20GHz
RAM	16.0GB	12.7GB
GPU	NVIDIA GTX1050 2.0GB	NVIDIA Tesla T4 15.0GB

안면 인식 에이징(aging) 이미지 데이터

데이터 구축년도	2022년	데이터셋	45,200건 (train: 40,150건, test: 5050건)
Contributor	894명	연령 범위	0~82세



구분	유형	데이터 형식	데이터 내용
사진데이터	이미지	png	참여자 1인당 유아~현재까지 연령대별 사진
라벨링데이터	바운딩박스, 키포인트	json	참여자 출생년도, 성별, 현재나이, 사진상 나이, 원본사진 촬영기기 등

Part 02

EDA & Data Preprocessing

Part 02 데이터셋 (image)

얼굴이 찍힌 사진파일
(png, 크기 다양)

Part 02 데이터셋 (image)



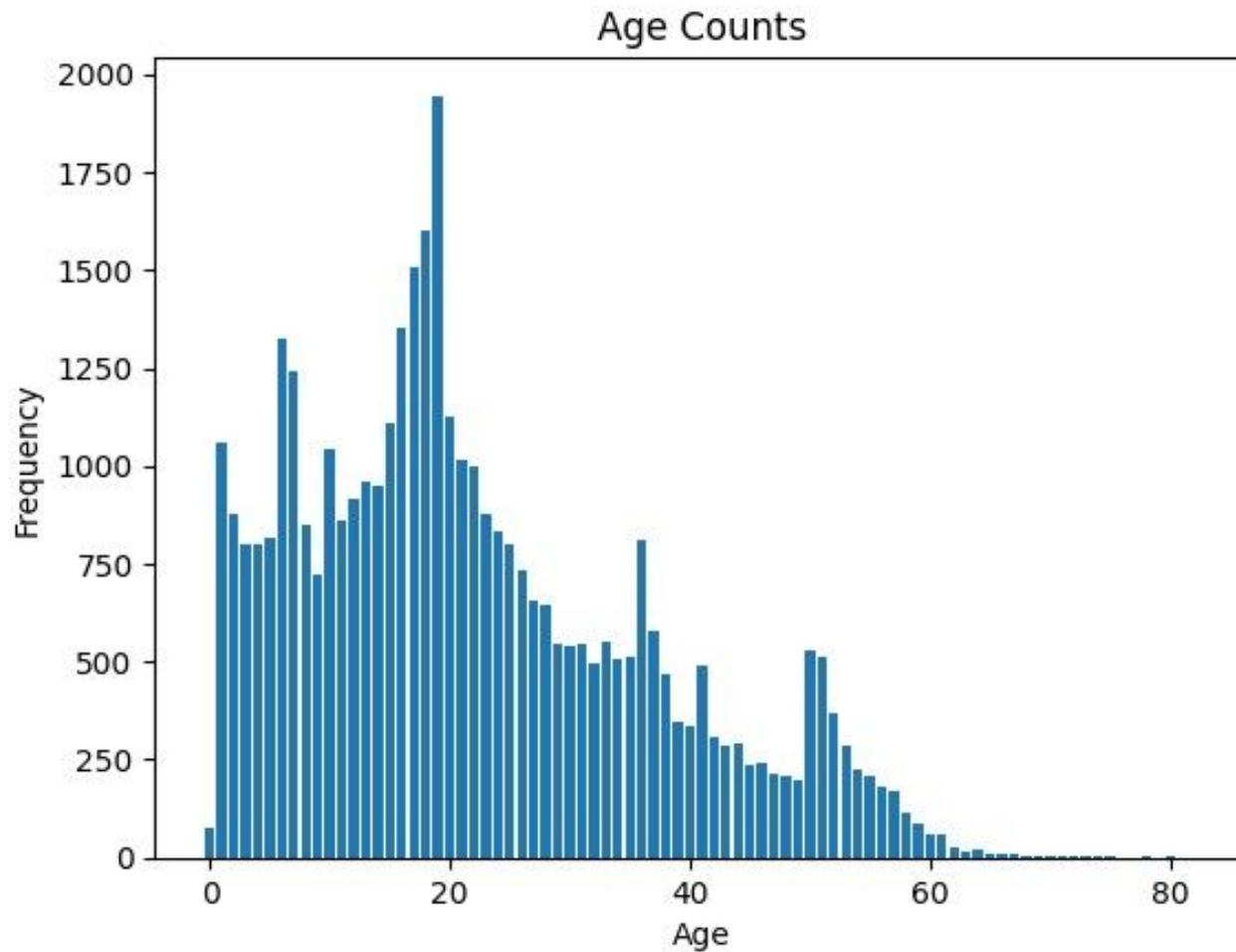
Part 02 데이터셋 (json)

생년, 현재나이, 촬영당시나이,
데이터형식, 데이터크기,
촬영장비, 바운딩박스, 랜드마크



```
{'filename': '0001_1992_06_00000020_F',
 'id': 1,
 'birth': 1992,
 'age_now': 31, (현재나이)
 'age_past': 6, (촬영당시나이)
 'format': 'png',
 'imgsize': '1.557MB',
 'width': 918,
 'height': 1629,
 'device': 'film',
 'gender': 'male', (성별) (바운딩박스)
 'annotation': [ {'box': {'x': 323.7489602657746,
   'y': 611.1985474670377,
   'w': 407.58051163597787,
   'h': 394.8600372545264}, (랜드마크)
 'landmark': [[526.4206368425632, 786.4665429221211],
 [445.48921818322185, 725.1151635275899],
 [611.2035522460938, 711.3058672248031],
 [473.55859375, 894.9515380859375],
 [589.5057373046875, 886.8681030273438]]}]}  
©Saebyeol Yu. Saebyeol's PowerPoint
```

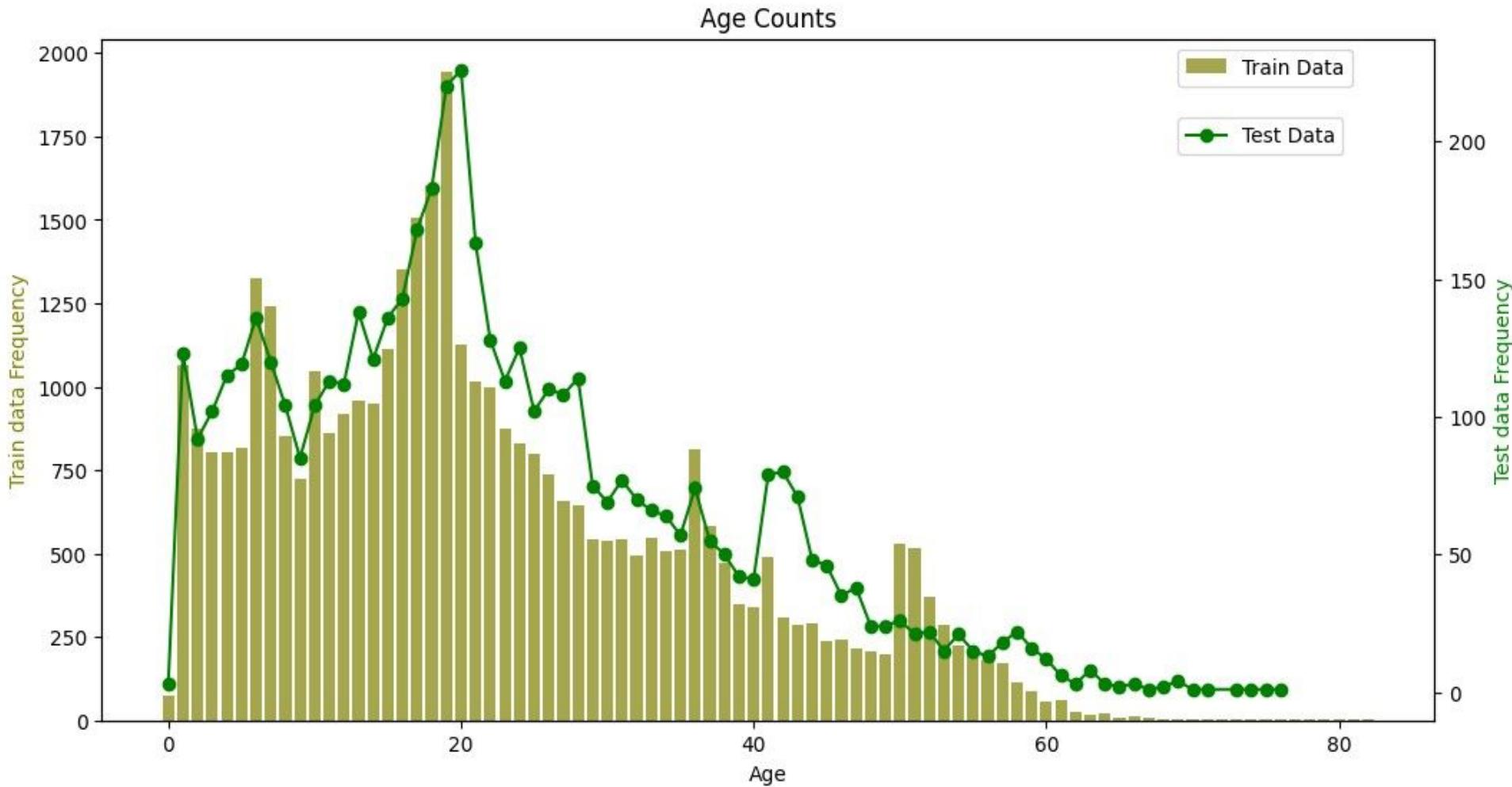
Age 분포



전체 데이터셋 연령 분포

10~20대 위주로 데이터가 많은 상태

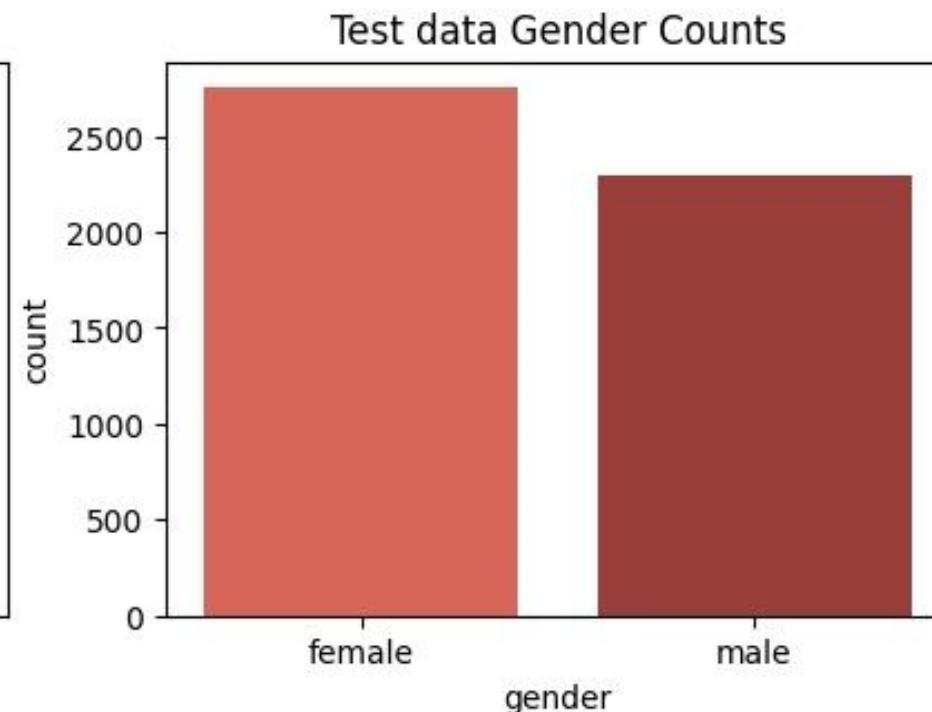
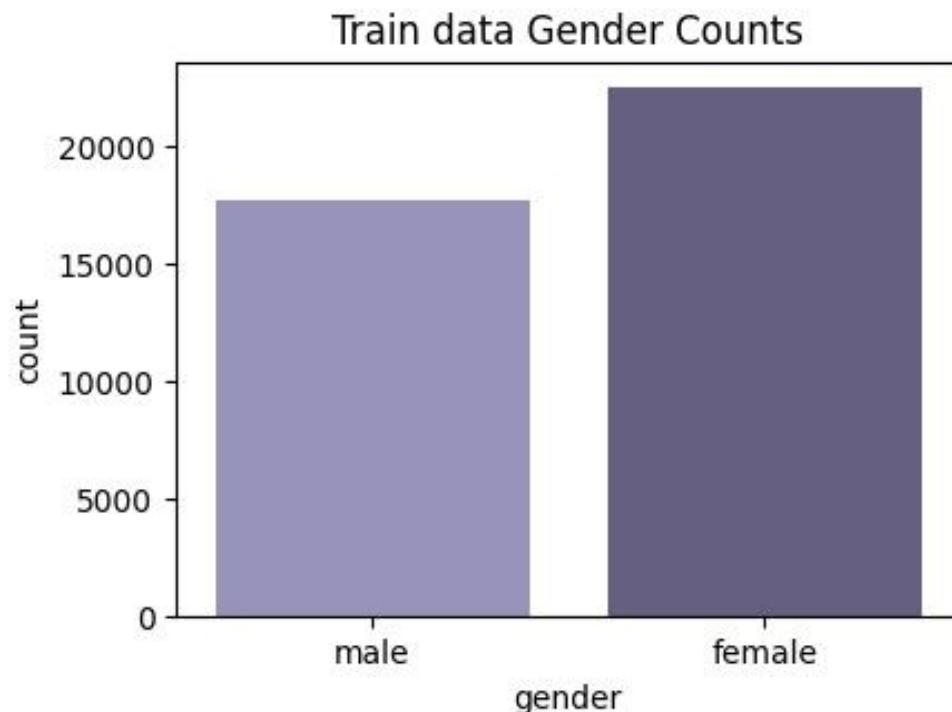
Age 분포



Train set / Test set 유사: Train/Test set은 균등하게 나누어진 것 확인할 수 있음

Gender 분포

- (train set) female보다 male보다 많음 : female로 오버피팅될 수 있음 (계층적 동일하게 학습 필요)

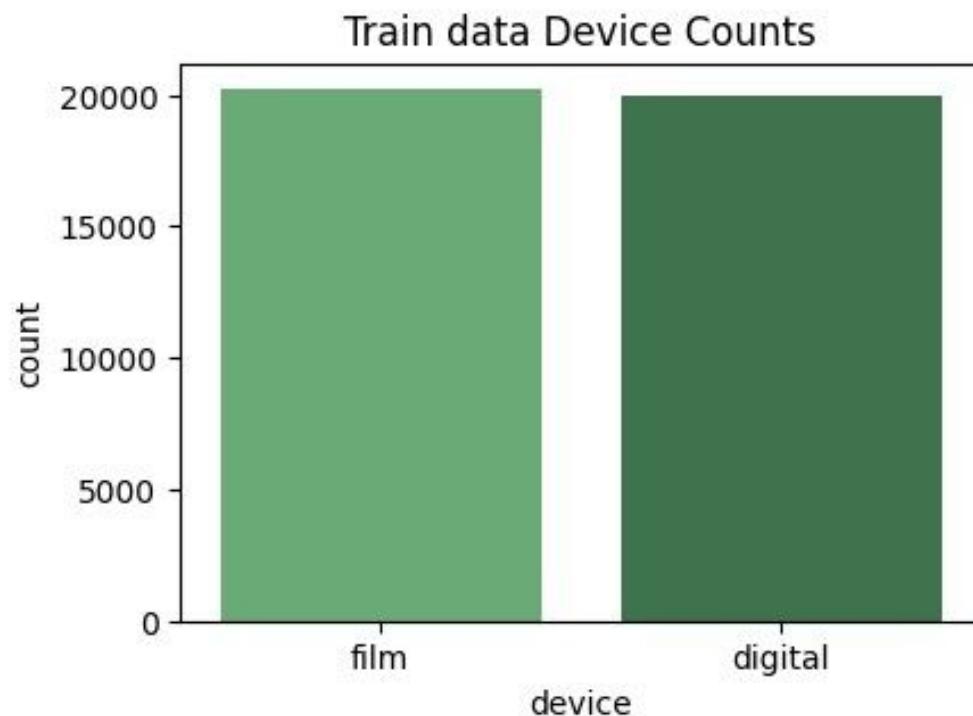


```
gender  
female    22450  
male      17700  
Name: count, dtype: int64
```

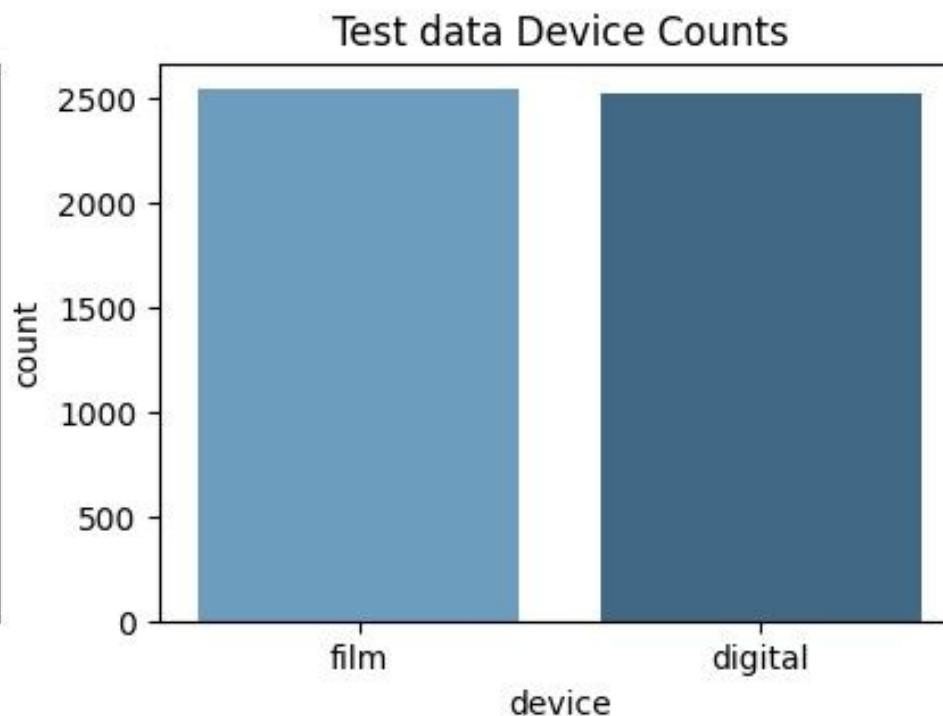
```
gender  
female    2750  
male      2300  
Name: count, dtype: int64
```

Device 분포

- Film과 Digital간 거의 차이 없음



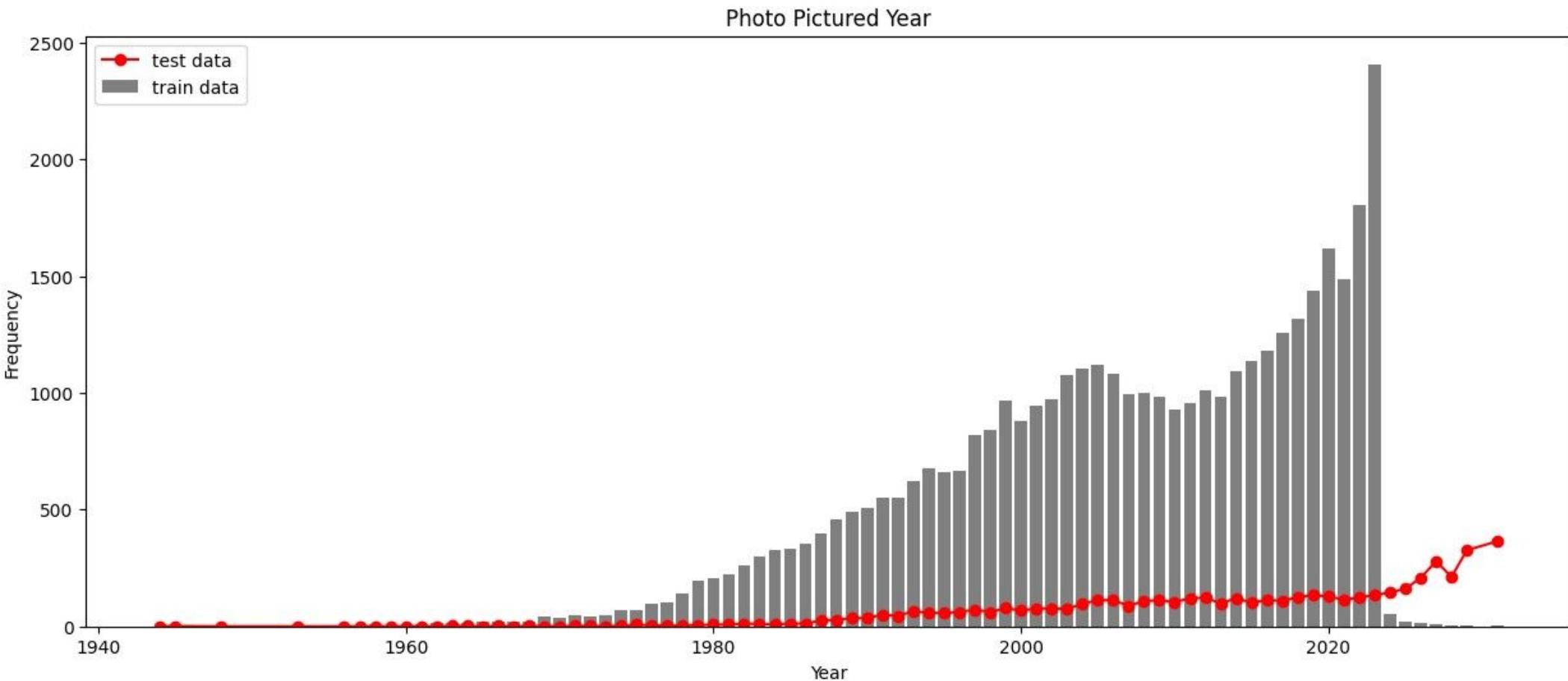
```
device
film      20186
digital   19964
Name: count, dtype: int64
```



```
device
film      2536
digital   2514
Name: count, dtype: int64
```

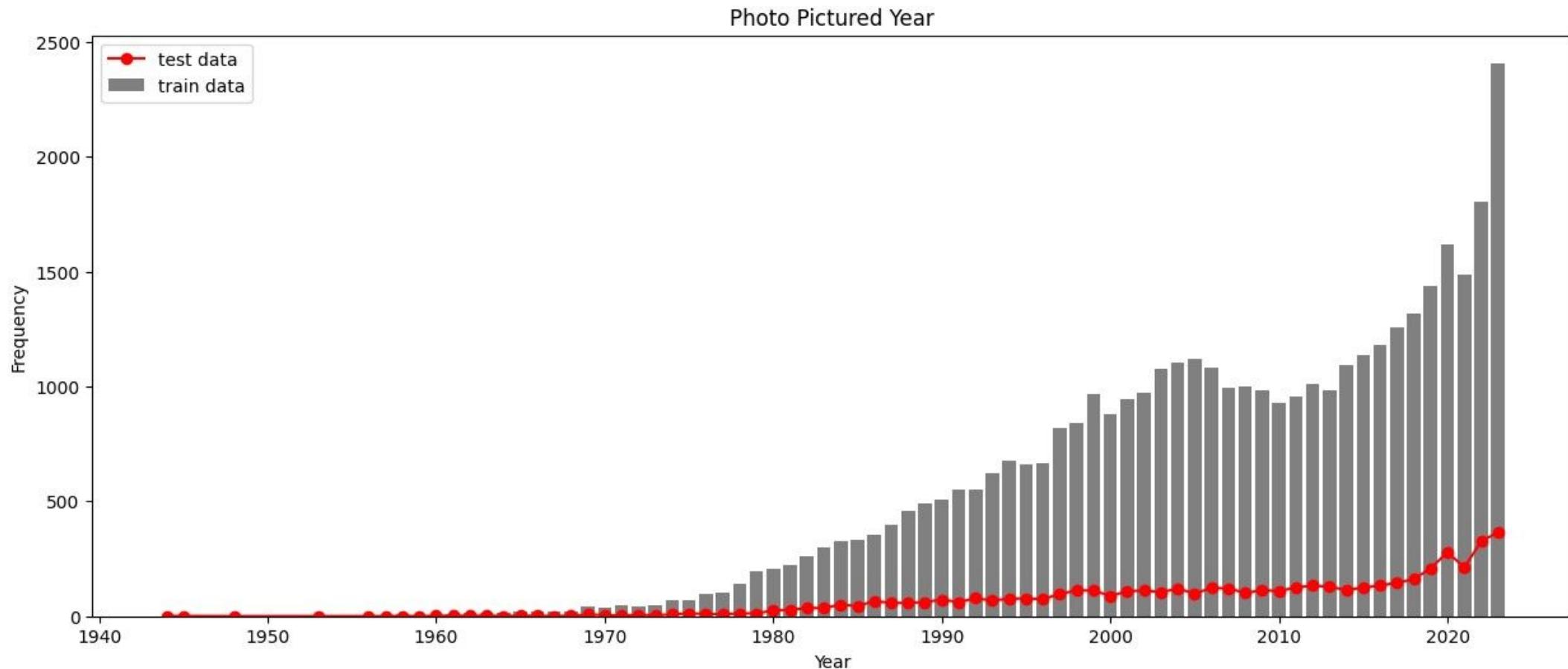
Pictured_year (사진촬영연도) 분포

- 2023년 이후로 그래프가 그려지는 것을 보아 데이터가 일부 오염된 것을 확인할 수 있음



Pictured_year (사진촬영연도) 분포

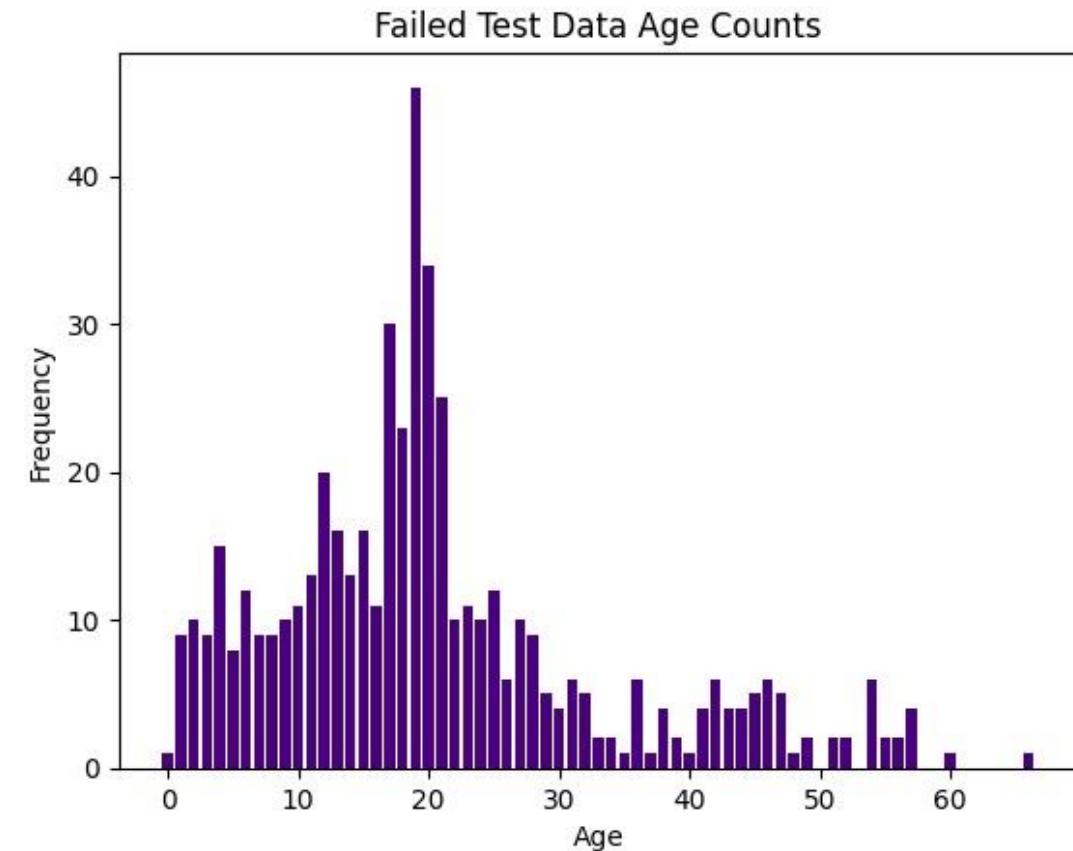
→ 2023년 이후 데이터 삭제: 2023년까지만 출력되는 것 확인



이미지 detecting이 잘 되지 않는 데이터

: train, test 데이터 비율은 0.1% 이하로 낮음

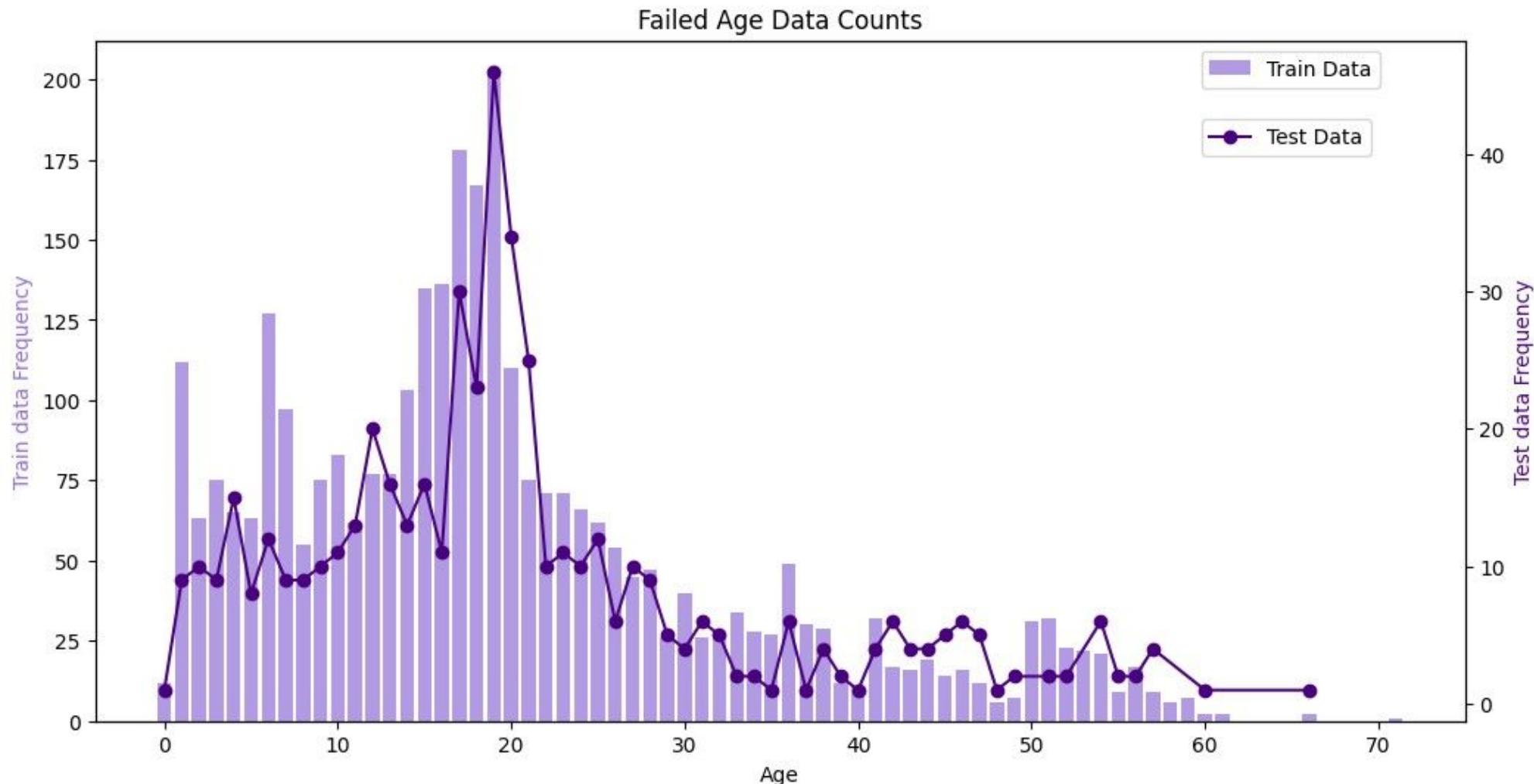
```
# train failed list: 3227 (0.08%)  
# test failed list: 514 (0.10%)
```



이미지 detecting이 잘 되지 않는 데이터

: train, test 데이터 비슷한 분포

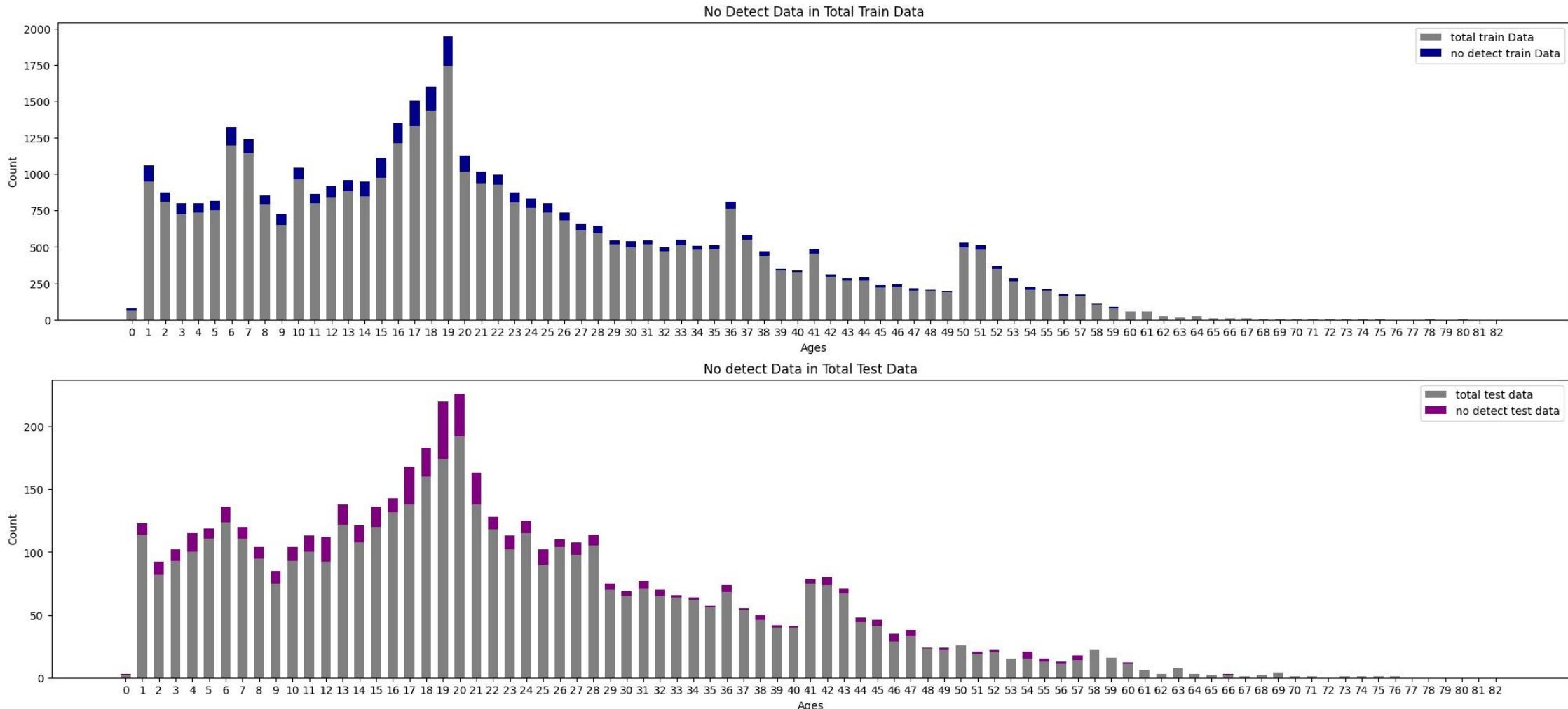
train failed list: 3227 (0.08%)
test failed list: 514 (0.10%)



이미지 detecting이 잘 되지 않는 데이터

: train, test 전체 데이터 대비 비율이 비슷함 → 제외

train failed list: 3227 (0.08%)
test failed list: 514 (0.10%)



[정제한 데이터]

Whole dataset: 41,410건

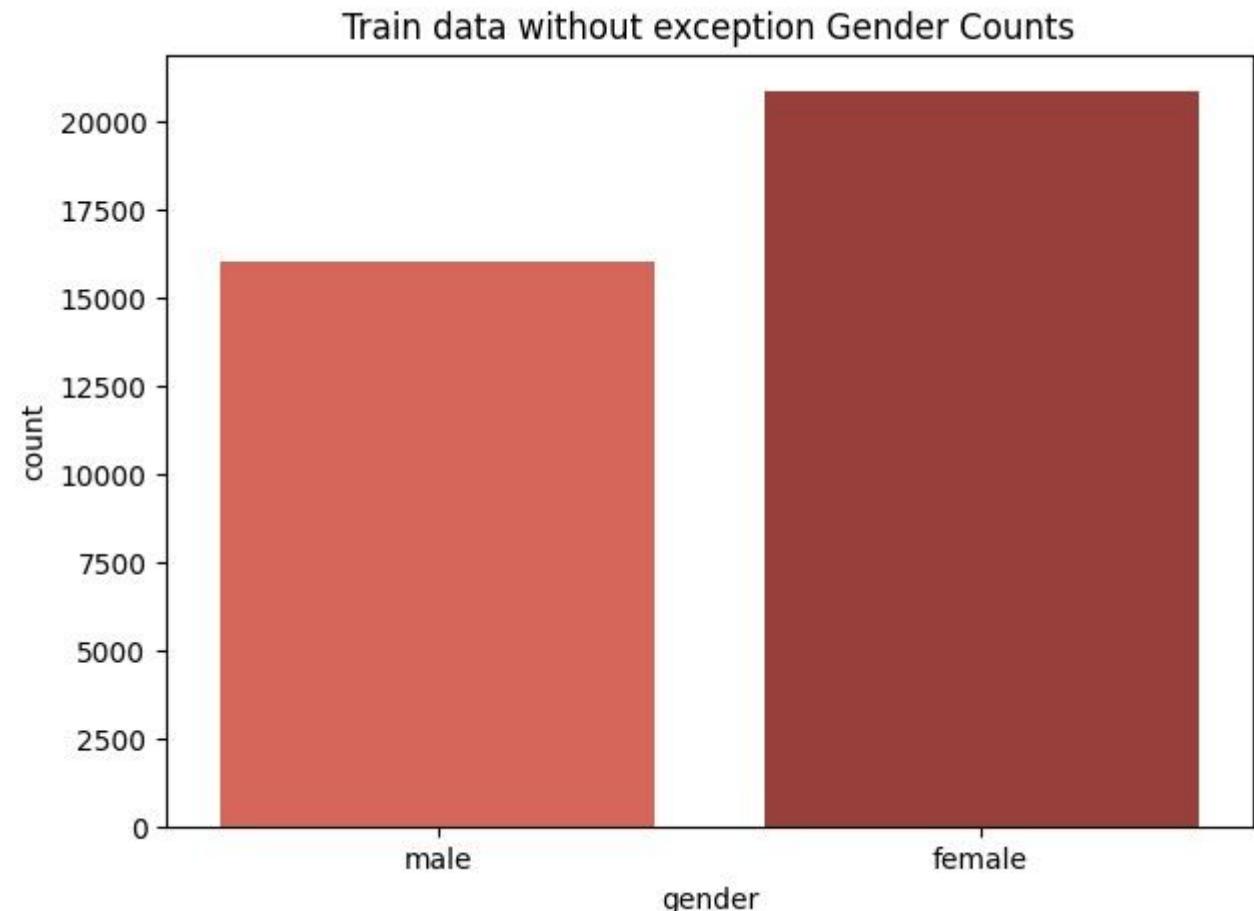
Train dataset: 36,839건

Test dataset: 4,571건

```
# Dataset after preprocessing  
Whole dataset: 41410  
Train dataset: 36839  
Test dataset: 4571
```

엑스를 이곳에 입력해
주세요

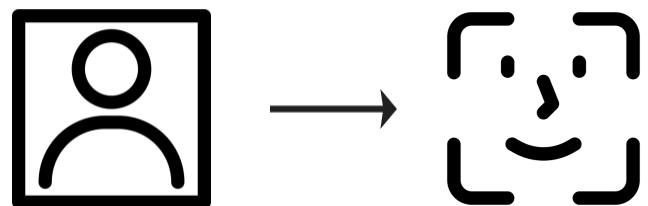
※ 성별데이터 클래스 불균형 → Balancing 필요



Part 03

Data Modeling





안면인식

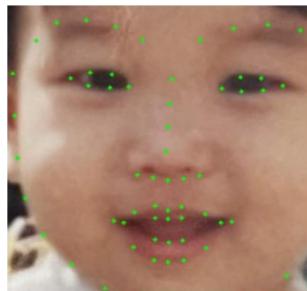
Face Image Detection

- 이미지 pixel 데이터
→ (224, 224)로 size 변환
- 얼굴 부위 안면 인식
오픈되어 있는 model 활용
(shape_predictor_68_face_landmarks.dat)

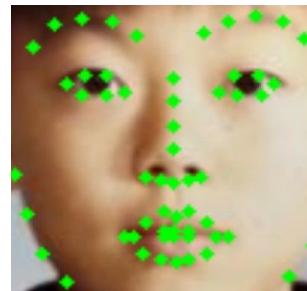
Image Classification

안면인식 (Face image Detection)

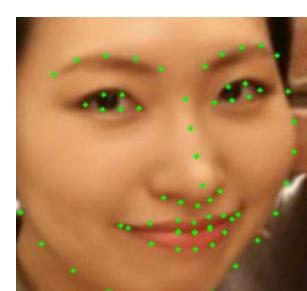
- 알려져 있는 shape_predictor_68_face_landmarks.dat 를 활용해 입력된 사진의 안면인식
- 모든 나이대에서 안면인식이 잘 되며, 화소가 낮은 이미지도 인식이 잘 되고 있음



0~9세



10~19세



20~29세



30~39세

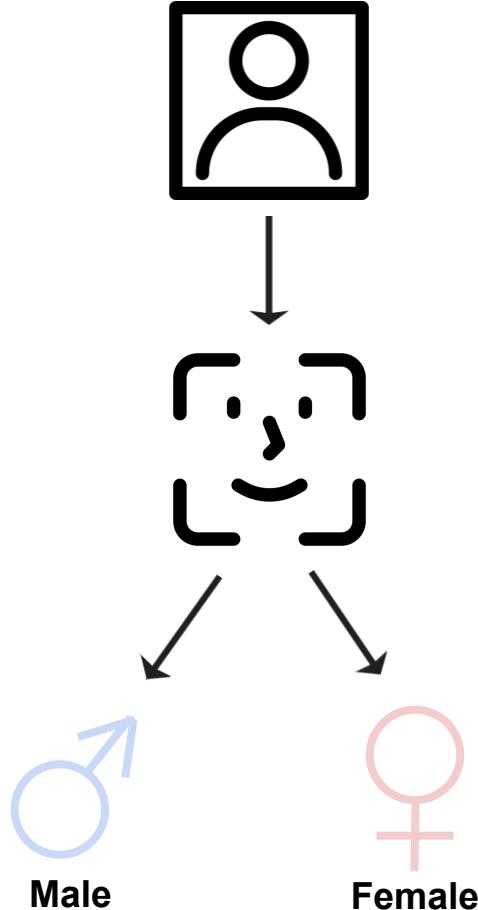


40~49세



50~59세

Part 03 Image Classification

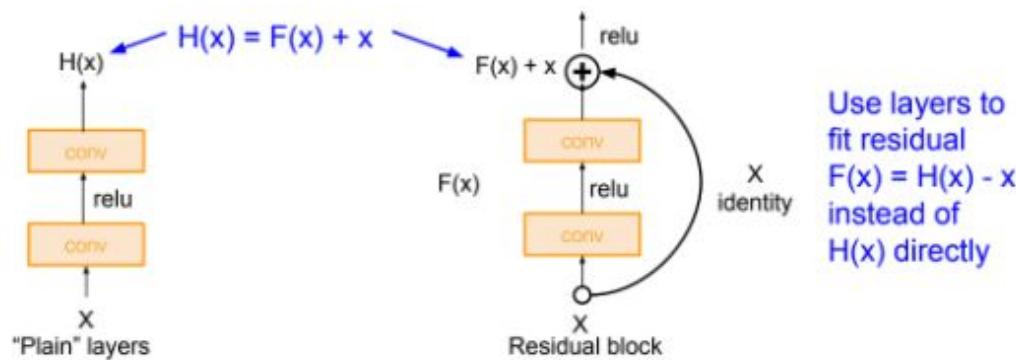


성별 분류 Gender Classification

- 인식된 얼굴을 바탕으로 성별 분류
(남성/여성)

ResNet 모델

- Plain 신경망에 Residual Block 을 추가한 모델
(Plain 신경망의 문제: 신경망층이 깊어질수록 학습이 어려워짐
기울기소실, network가 깊어질수록 모델의 accuracy가 향상되지 않는 문제)
- 네트워크 깊이를 깊게 만들면서 accuracy를 향상시킴: Residual Block 추가
- $F(x) = H(x) - x$: Residual(잔차), input_value가 0으로 수렴하도록 만들어줌



- VGG-19 모델에 3x3 ConV layer를 2개씩 쌓고, Residual block을 끼워줌
주기적으로 필터를 2배씩 늘림
- feature map 크기 줄일 때, stride=2로 다운샘플링
- Residual Block : (ResNet-34) Basic Block 사용, (ResNet-50 이상) Bottleneck Block 사용
 $1 \times 1 \text{ conv} \rightarrow 3 \times 3 \text{ conv} \rightarrow 1 \times 1 \text{ conv}$
- 마지막에 FC층 없고 global average pooling

ResNet34

- (layer) 34층까지 설계, 모든 ConV layer 후 Batch Normalization
- (Hyperparameters) Batch size: 32, Epoch: 10, Learning rate: 0.001
- (option) dropout은 쓰지 않음

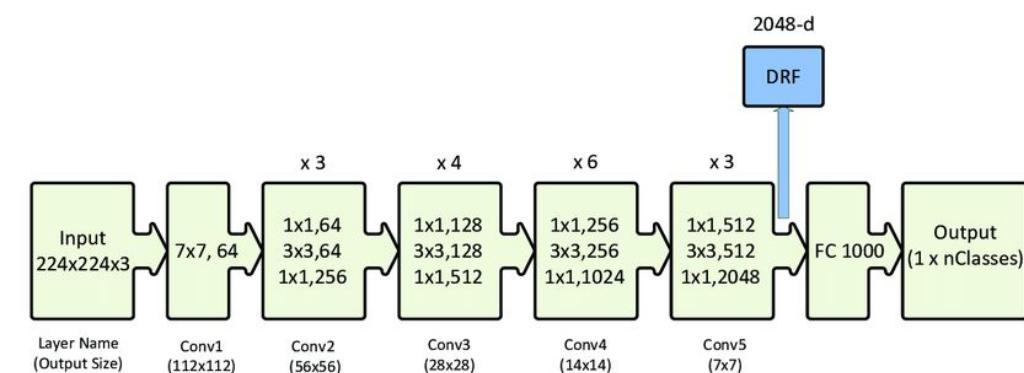
test accuracy: 92.93%

```
Using device: cuda
C:\Python\Python39\lib\site-packages\torchvision\models\_utils.py:208: UserWarning
  warnings.warn(
C:\Python\Python39\lib\site-packages\torchvision\models\_utils.py:223: UserWarning
  warnings.warn(msg)
Test accuracy: 92.93%
```

ResNet50

- (layer) 50층까지 설계, 모든 ConV layer 후 Batch Normalization
- (Hyperparameters) Batch size: 8, Epoch: 10, Learning rate: 0.001
- (option) dropout은 쓰지 않음

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112			7×7, 64, stride 2		
				3×3 max pool, stride 2		
conv2_x	56×56	$\left[\begin{array}{l} 3 \times 3, 64 \\ 3 \times 3, 64 \end{array} \right] \times 2$	$\left[\begin{array}{l} 3 \times 3, 64 \\ 3 \times 3, 64 \end{array} \right] \times 3$	$\left[\begin{array}{l} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{array} \right] \times 3$	$\left[\begin{array}{l} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{array} \right] \times 3$	$\left[\begin{array}{l} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{array} \right] \times 3$
conv3_x	28×28	$\left[\begin{array}{l} 3 \times 3, 128 \\ 3 \times 3, 128 \end{array} \right] \times 2$	$\left[\begin{array}{l} 3 \times 3, 128 \\ 3 \times 3, 128 \end{array} \right] \times 4$	$\left[\begin{array}{l} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{array} \right] \times 4$	$\left[\begin{array}{l} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{array} \right] \times 4$	$\left[\begin{array}{l} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{array} \right] \times 8$
conv4_x	14×14	$\left[\begin{array}{l} 3 \times 3, 256 \\ 3 \times 3, 256 \end{array} \right] \times 2$	$\left[\begin{array}{l} 3 \times 3, 256 \\ 3 \times 3, 256 \end{array} \right] \times 6$	$\left[\begin{array}{l} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{array} \right] \times 6$		
conv5_x	7×7	$\left[\begin{array}{l} 3 \times 3, 512 \\ 3 \times 3, 512 \end{array} \right] \times 2$	$\left[\begin{array}{l} 3 \times 3, 512 \\ 3 \times 3, 512 \end{array} \right] \times 3$	$\left[\begin{array}{l} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{array} \right] \times 3$		
	1×1			average pool, 1000-d fc.		
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9		



ResNet50

- (layer) 50층까지 설계, 모든 ConV layer 후 Batch Normalization
- (Hyperparameters) Batch size: 8, Epoch: 10, Learning rate: 0.001
- (option) dropout은 쓰지 않음

test accuracy: 91.58% (ResNet34 test accuracy: 92.93%)

```
Using device: cuda
C:\Python\Python39\lib\site-packages\torchvision\models\_utils.py:208: UserWarning
  warnings.warn(
C:\Python\Python39\lib\site-packages\torchvision\models\_utils.py:223: UserWarning
  warnings.warn(msg)
Test accuracy: 91.58%
```

→ ResNet34 사용!

Part 03 Image Classification



나이 분류 Age Classification

- 인식된 얼굴을 바탕으로 나이 범위 분류
(0~9세/10~19세/ 20-29세/
30-39세/40~49세/50~59세/60대 이상)

Part 03 Age Classification

데이터셋 나누어 학습

- json데이터의 past_age(촬영당시 나이)를 기반으로 나이대를 나눠서 7개의 클래스로 이미지 데이터를 분류함

Class	Age Range
0	0~9세 (10대 미만)
1	10~19세 (10대)
2	20~29세 (20대)
3	30~39세 (30대)
4	40~49세 (40대)
5	50~59세 (50대)
6	60세~ (60대 이상)

Part 03 Age Classification

Models

구분	VGG	ResNet	EfficientNet
등장 배경	Network 깊이가 CNN 성능에 어떤 영향을 주는지 알아보기 위해 층을 깊이 쌓는 것에 초점	Network 깊이가 깊어짐에 따라 Gradient vanishing, Overfitting의 문제 발생	Network 깊이, filter 개수, resolution(해상도)에 따라 모델의 성능이 달라짐
특징	3x3 filter, stride=1 반복 적용 max pooling을 stride=2로 적용	Residual Learning (이전의 결과를 추가해 학습시킴)	width, depth, resolution 적절히 선택해 Compound Scaling Method를 사용

Part 03 Age Classification

VGG-16

```
100%|██████████| 37311/37311 [22:56<00:00, 27.12it/s]
train Loss: 6.2070 Acc: 0.2457
100%|██████████| 3124/3124 [01:48<00:00, 28.66it/s]
val Loss: 7.4199 Acc: 0.3092

Training complete in 54m 41s
Best val Acc: 0.309219
```

accuracy: ≈ 30%

VGG-19

```
100%|██████████| 9328/9328 [22:48<00:00, 6.82it/s]
0%|          | 0/781 [00:00<?, ?it/s] train Loss: 2.2441 Acc: 0.3251
100%|██████████| 781/781 [01:49<00:00, 7.11it/s]
val Loss: 1.9625 Acc: 0.3444

Training complete in 49m 49s
Best val Acc: 0.353073
```

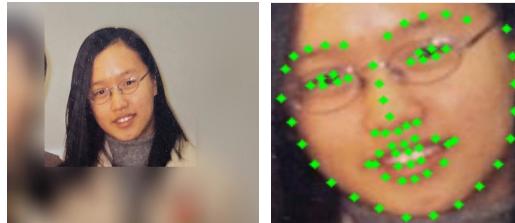
accuracy: ≈ 34%

→ accuracy가 너무 낮음

Part 03 Age Classification

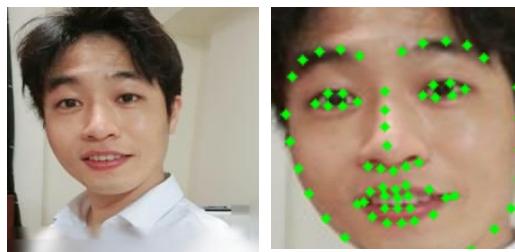
ResNet152

- Accuracy: 약 53%



```
Predicted age range: 1  
Process finished with exit code 0
```

10대 이미지를 10대로 분류



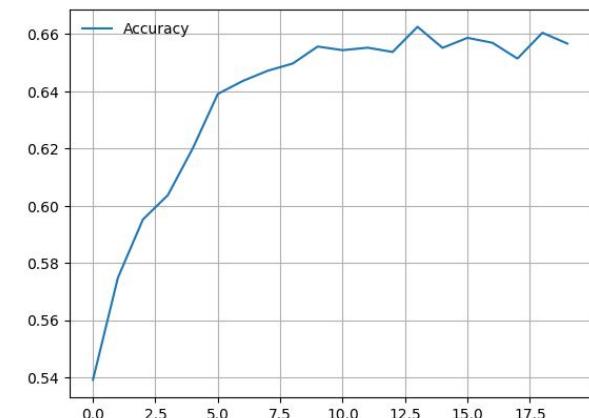
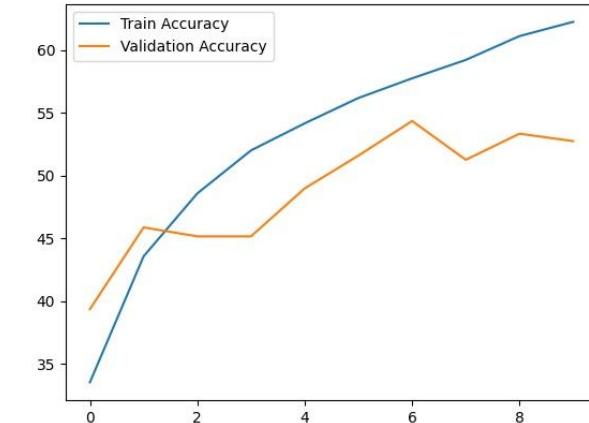
```
Predicted age range: 0  
Process finished with exit code 0
```

30대 이미지를 10대로 분류

EfficientNet

- Accuracy: 약 66%

```
Epoch 12: train_loss: 1.2745, val_loss: 1.4433, val_acc: 0.6537  
100%|██████| 502/502 [02:42<00:00, 3.08it/s]  
Saved PyTorch Model State to /content/drive/MyDrive/3rdProject/best_ep_13_0.66252.pt  
Epoch 13: train_loss: 1.2651, val_loss: 1.4405, val_acc: 0.6625
```



Part 03 Age Classification

ResNet101

- VG

Part 03 Age Classification

모델 선택

구분	VGG-16	VGG-19	ResNet50	ResNet101	EfficientNet
Batch Size	16	4	16	8	64
Epochs	2	1	10	2 → 20	20
Optimizer	SGD	SGD	SGD	SGD	Adam
Learning Rate	0.001	0.001	0.001	0.001	0.001
Test Accuracy	39.95%	34.44%	46.13%	91.3%	66.25%

→ ResNet101 사용!

Part 03 Age Classification

데이터셋 클래스 줄여서 재학습

- 60세 이상 데이터셋 갯수가 적으므로(244개) 50세 이상으로 편입 후 재학습

Class	Age Range	Number of datasets
0	0~9세 (10대 미만)	8,574
1	10~19세 (10대)	12,252
2	20~29세 (20대)	8,228
3	30~39세 (30대)	5,359
4	40~49세 (40대)	2,805
5	50세~ (50대 이상)	2,932 (2,688 + 244)

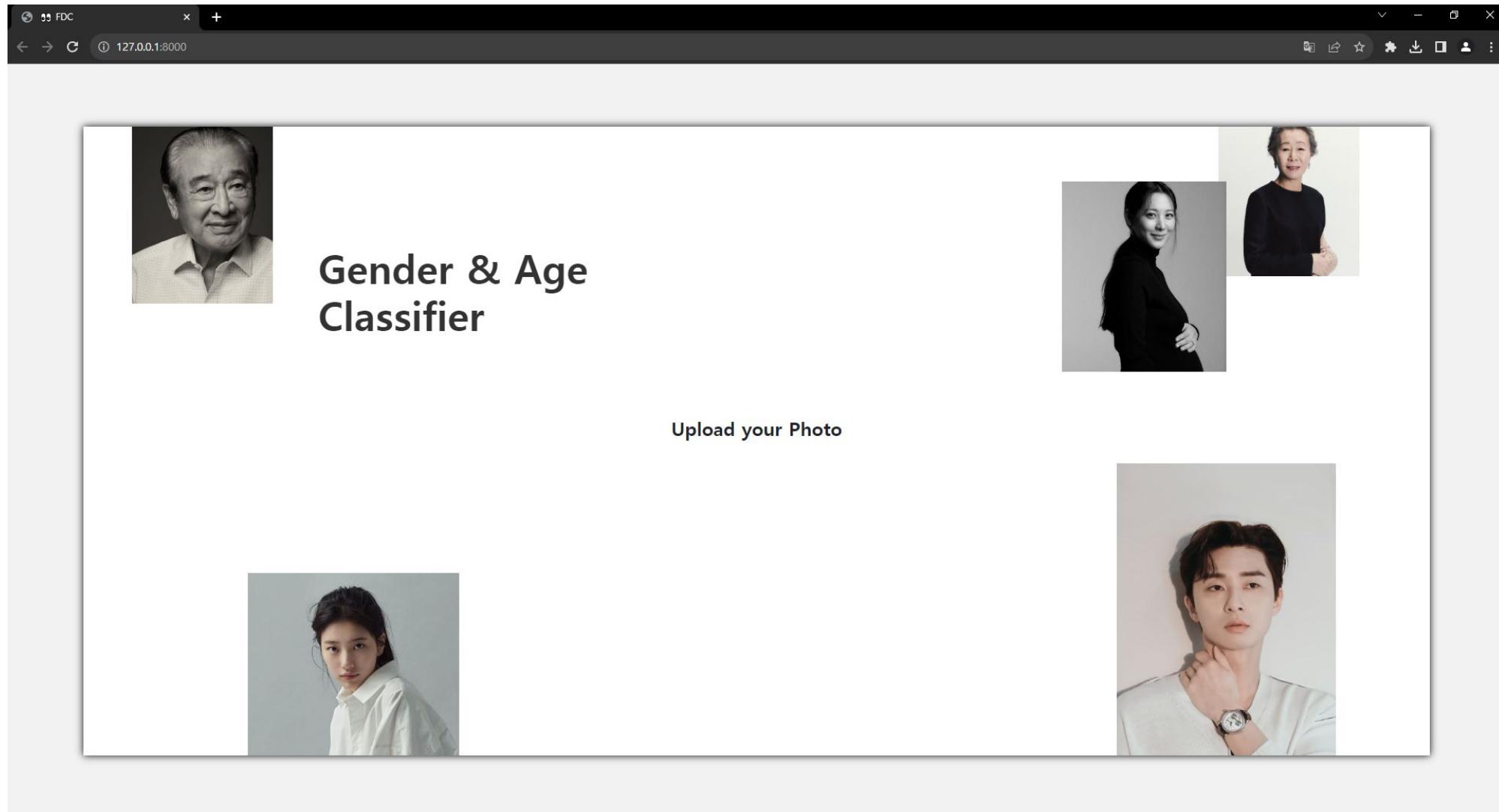
Part 04

Web Service



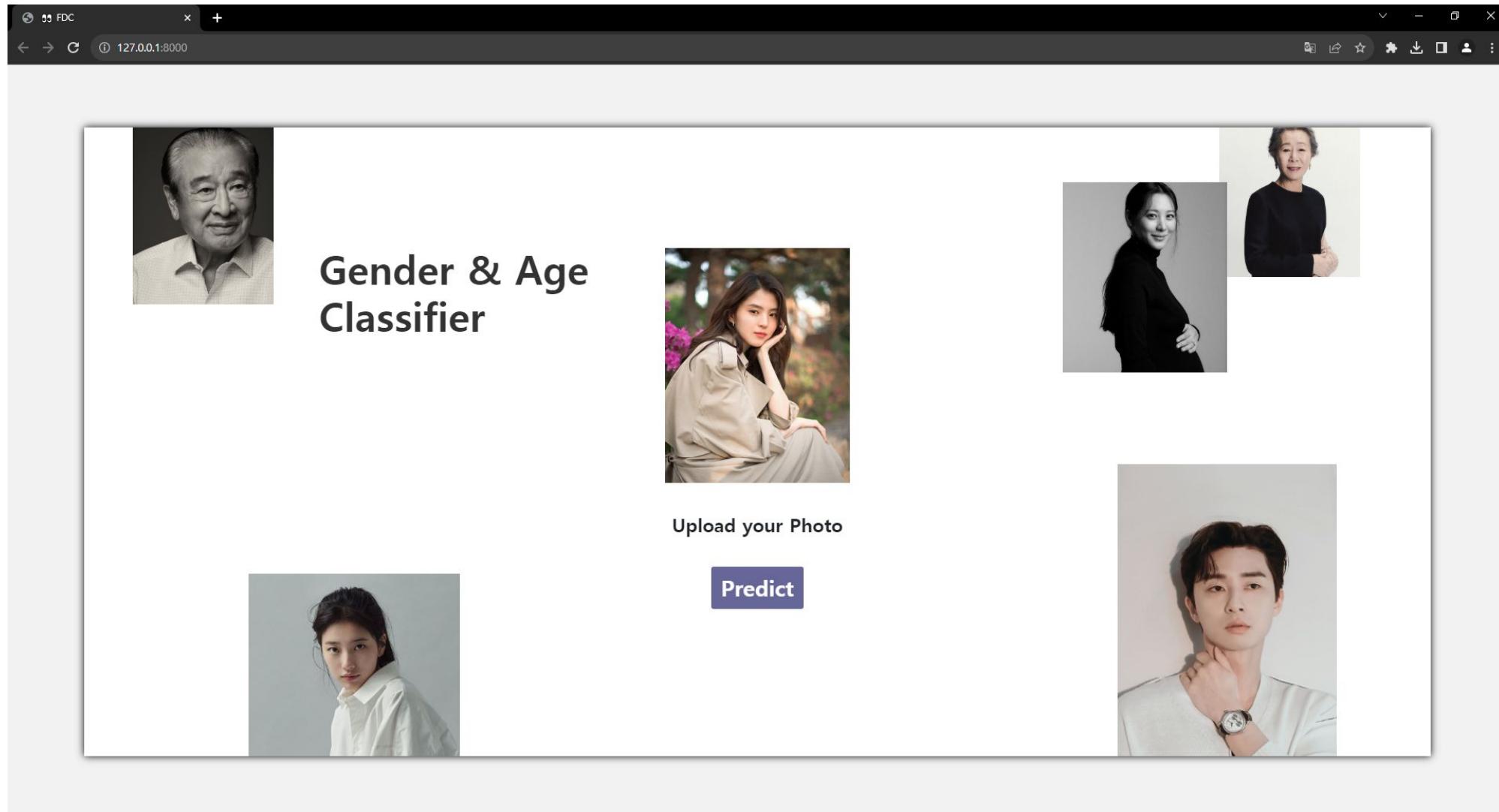
Part 04 Web Service

Web page



Part 04 Web Service

Web page



Part 04 Web Service

Web page: test

The screenshot shows a web browser window with the URL `127.0.0.1:8000`. The page displays a "Gender & Age Classifier" interface. On the left, there is a small portrait of an elderly man. In the center, the main title "Gender & Age Classifier" is displayed above a photograph of a young woman sitting outdoors. Below this image, a text box contains the Korean text "당신은 아마도" and the English text "Female 이고 20대 입니다." To the right of the central image, there are two smaller photographs: one of a young woman in a black top and another of an elderly woman in a black top. At the bottom left, there is a small portrait of a young woman. At the bottom center, a blue "HOME" button is visible. At the bottom right, there is a portrait of a young man.

Gender & Age Classifier

당신은 아마도
Female 이고 20대 입니다.

HOME

Part 04 Web Service

Web page: test

The screenshot shows a web browser window with the URL `127.0.0.1:8000`. The page displays a "Gender & Age Classifier" interface. On the left, there is a small portrait of an elderly man. In the center, the main title "Gender & Age Classifier" is displayed above a larger portrait of a man with dark hair and a mustache. Below this central image, Korean text reads "당신은 아마도 Male이고 30대입니다." (You are probably a Male in your 30s). At the bottom left, there is a portrait of a young woman. At the bottom right, there is a portrait of a young man. A "HOME" link is located at the bottom center. The browser interface includes standard navigation buttons and a status bar.

Part 04 Web Service

Web page: test

The screenshot shows a web browser window with the URL `127.0.0.1:8000` in the address bar. The page itself is a gender and age classifier interface. On the left, there is a small portrait of an elderly man. In the center, the text "Gender & Age Classifier" is displayed above a portrait of a woman with short dark hair, smiling. To the right of this central image are two smaller portraits: one of a young woman in profile and another of an elderly woman. At the bottom left is a large, mostly white rectangular area containing a portrait of a young woman. At the bottom right is a portrait of a young man. In the center bottom of the page, the text "당신은 아마도 Female 이고 50대 입니다." (You are likely a Female in your 50s) is displayed in Korean and English. Below this text is a blue "HOME" link.

Web page: test

The screenshot shows a web browser window with the URL `127.0.0.1:8000`. The main content is a web page titled "Gender & Age Classifier". On the left, there is a portrait of an elderly man. In the center, there is a photograph of an elderly man wearing glasses and a vest, holding a pen. To the right, there are two side-by-side portraits of women. At the bottom left, there is a portrait of a young woman. At the bottom right, there is a portrait of a young man. Below the central image, there is Korean text: "당신은 아마도 Male 이고 60대 이상입니다." (You are probably a Male in your 60s or older). In the bottom center, there is a blue "HOME" link.

Gender & Age Classifier

당신은 아마도
Male 이고 60대 이상입니다.

HOME

Part 05

Conclusion



Final Models

Gender Classification Model: ResNet34 (Acc: 92%)
Age Classification Model: ResNet101 (Acc: 91%)

한계

- 50대 이상인 사람들의 데이터수가 적어서 50대 이상 분류의 정확도가 떨어짐. 50대+60대 이상을 하나의 클래스로 편입해서 학습도 해보았으나, 성능이 개선되지 않음. 더 정교한 설계의 모델이 필요할 것으로 보임.

향후 과제

- 동일한 데이터(한국인 안면 aging 데이터)로 생성모델을 구현하여, 사용자의 사진을 입력하면 사용자가 원하는 나이대의 얼굴이미지를 생성하는 서비스 제공
- 미아, 기억 상실에 의한 행방불명 등으로 장기 실종 상태인 대상자의 현재 얼굴과 실종 신고 시 사용되었던 당시 얼굴 사진으로 본인 여부 확인에 활용

배운 점 및 아쉬운 점

1 데이터셋

데이터셋의 형태와 특징을 이해하여 전처리하는 것이 중요하다는 점을 깨달음. **outlier**와 같은 데이터를 학습 중에 발견해 재학습을 여러번 수행함.

2 데이터모델링

레이어를 무조건 많이 쌓거나 복잡하다고 모델 성능이 향상되는 것은 아니라는 점을 확인함.

같은 모델을 사용하더라도 하이퍼파라미터에 따라 성능이 변한다는 점도 알 수 있었음.

3 데이터학습

PC의 저장 공간 및 GPU자원(2GB)이 한정되어 있어 주제에 맞는 데이터셋 로드 및 모델 학습에 어려움이 컸음. 컴퓨터 자원의 중요성을 깨달음.

4 프로젝트 전반

프로젝트 초기에는 나이대별로 변해가는 안면 이미지 생성을 하고 싶었으나 한정된 시간과 자원 등으로 인해 분류 모델을 구현하는 데 그쳐 아쉬움.



Q & A

EXIT



Fin.

참고자료

PseudoLab PyTorch guide

<https://pseudo-lab.github.io/pytorch-guide/docs/ch03-1.html>

CNN Architectures

<https://inovation97.tistory.com/25>

CNN 사전학습 모델

<https://yeong-jin-data-blog.tistory.com/entry/%EC%82%AC%EC%A0%84%ED%95%99%EC%8A%B5-%EB%AA%A8%EB%8D%B8-CNN>

CNN의 등장과 발전 과정

<https://dotiromooook.tistory.com/21>

VGGNet 개념 정리

<https://daechu.tistory.com/10>

