Project using Deep Neural Network

Group 2

IMEN468 - Deep Learning for Industrial Engineering:
Group Project
second term
1 December 2021

Member

2013270064 Yoon Tae Joon, 윤태준

2016131547 Lee Jee Hyun, 이지현

2017170859 Choi Jeong eun, 최정은

2017380322 Ha Eun gyeom, 하은겸

2017390348 Kim Jong Min, 김종민

2018170865 Park Jae Hyun, 박재현

2019170345 Park Seok Hyun, 박석현

Contents

1. Data Description

- Source : Medical mask dataset(Humans-in-the-loop)
- About 6,000 images
- Consist of 20 classes of different accessories on faces
  - face with mask, face with no mask, face with other coverings, face with mask incorrect
  - glasses, hijab, hats, etc.
- The images were annotated manually by the refugee workforce of Humans in the Loop in Bulgaria.

Medical mask dataset

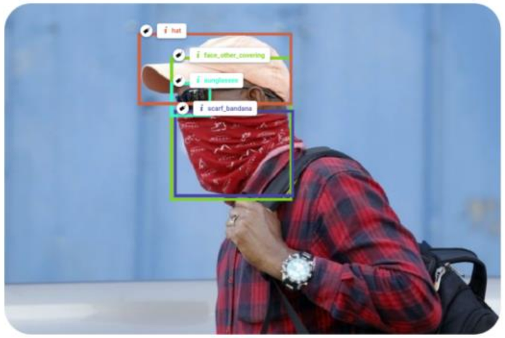
More than 6000 images for detecting masks and accessories

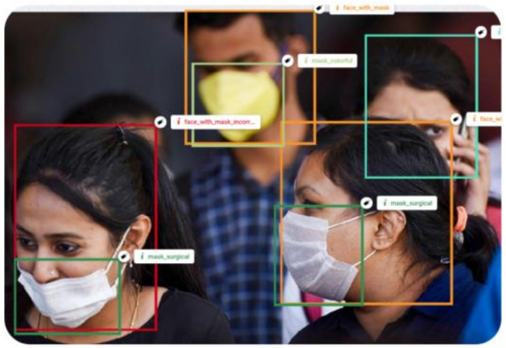
Humans in the Loop is publishing an open access dataset annotated as a contribution to the worldwide fight against COVID-19.

The dataset consists of 6k images acquired from the public domain with an extreme attention to diversity, featuring people of all ethnicities, ages, and regions.

In addition, the datset covers 20 classes of different accessories as well as a classification of faces with a mask, without a mask, or with an incorrectly worn mask.

The images were collected and annotated by the refugee workforce of Humans in the Loop in Bulgaria.





2. How to Classification

- Definition: Detect people wearing masks properly or not
- Objective: Protecting the health of individuals against COVID-19
- Classifying instances into four class labels
    - face with mask
    - face no mask
    - face other covering
    - face with mask incorrect

face with mask

face with no mask*
hood

* Original data was labeled 'face with other covering', so we are modified it.

face with mask

face with no mask*
hood

* Original data was labeled ‘face with other covering’, so we are modified it.

face other covering

face with mask incorrectly

How can we classify them?

face other covering

face no mask

3. Data Pre-processing

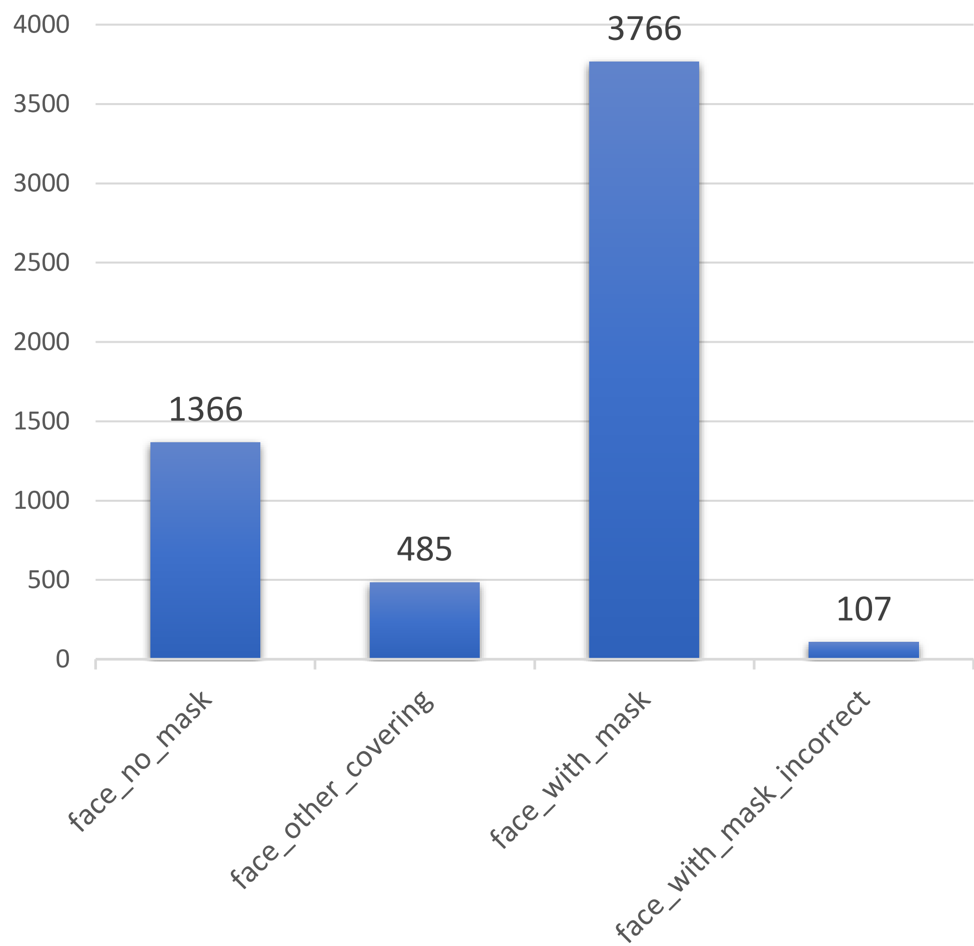
- Re-labeled Images
- Eliminate noise data
  - Group pictures
  - Dog with mask
- Data augmentation using image generator
- Scaling

Eliminate high noise data

Group pictures

Dog with mask

Imbalanced Data



Data augmentation

- ImageDataGenerator
- The number of data for each class is similar.
  - face with mask: 2605
  - face with no mask: 975 → 2590
  - face other covering: 346 → 2581
  - face with mask incorrect: 79 → 2211
    - i의 의미는 반복횟수 인가요?

```
in_corr=79(batch:32, i>80)->2211

no_mask=975(batch:32, i>50)->2590

other=346 (batch:32, i>70)->2581

mask=2605
```

Gray-scale vs. RGB-scale

- 마스크는 하얀색이 많고 색상이 중요하지 않을 수 있다고 생각되어, computation을 줄일 수 있는 Gray scale로 변환을 해보았다.
- 하지만, Gray scale과 RGB scale 결과를 비교해봤을 때, RGB scale이 2배 가량 좋았다. 따라서 색상 정보가 중요하게 작용한다는 것을 알 수 있다.
- 우리는 보통 마스크 생각하면 하얀 마스크만 생각하는데, 전세계적으로 보면 아니다

4. Introduction to Models

- CNN
- Pre-trained Model
  - VGG16
  - Inception ResNetV2
  - InceptionV3
  - EfficientNet

CNN

- 간단한 설명
  - base model :
- Hyperparameter 설명
  - base : max pooling, 3x3 kernels, ReLU, Softmax(분류시), dropout = 0.5

  - RGB scale을 사용
  - 레이어가 깊어질 수록 모델 성능이 좋아졌다.

CNN Model Comparison

Name	ConvLayer	Kernel size	Batch size	Total parameters	Test acc
CNN_3_1					
CNN_3_2					
CNN_5_1	5				
CNN_5_3			32		

VGG16, Inception ResNetV2, Inception V3

- 간단한 설명
- Hyperparameter 설명
- GAP(global avg pooling) 적용

Model Comparison

	GAP	Batch size	Drop out	Valid acc	Test acc
inceptionV3_1	○	-	-	0.7752	86.29
inceptionV3_2	○	-	-	0.8007	89.99
inceptionV3_3	○	-	-	0.7977	88.78
vgg16	○	-	-	0.6350	74.78
incresv2	○	-	-	0.8087	89.24

EfficientNet

- 간단한 설명
  - compound scaling 이용
    - 네트워크의 width, depth, resolution 사이의 balance를 맞춘다
- Hyperparameter 설명
  - 성능이 높아지는 경우 거의 없었고,
  - 성능이 낮아지는 경우 : 배치 사이즈 낮게 하면

Model Comparison

	Drop out	Batch size	Optimizer	Valid acc	Test acc
EfficientNet B0	-	-	-	0.7236	88.03
	-	64	-	0.8076	89.59
	0.2	64	-	0.7812	90.34
	0.6	64	-	0.8555	90.40
	-	64	GAP	0.7490	89.54
	-	64	Swish	0.7529	89.99
	-	32	Swish	0.7717	87.74
	-	64	SGD(0.9)	0.8408	89.99
	-	-	Adagrad	0.7734	87.10
	-	-	Sigmoid	0.7832	90.40

5. Model Selection

Name	Hyperparameter	Test accuracy(%)
CNN	RMS_5_3	83.89
inceptionV3_2	Global average pooling	89.99
EfficientNet B0	Dropout = 0.6, batchsize = 64	90.40

EfficientNet(batchsize = 64, dropout = 0.6) is best!

6. Conclusion

- 데이터 전처리가 중요하다 : 문제 정의에 알맞게 data를 구성하고, 올바르게 라벨링하였다
- Hyperparameter를 잘 조정하는 게 매우 중요하다
  - dropout : overfitting을 피하기 위해, 적용 유무가 정확도에 큰 차이를 보였다.
    - vanishing Gradient Problem을 피하기 위해
  - batch-size
- long model을 사용하는 게 좋다 : 레이어가 깊어질 수록 성능 향상

Limitation

End of Document

&

Thank you