

# Machine Learning Assignment

이 해 민

2021312394

인공지능융합전공

## 1. Problem Definition

주어진 6가지의 요가 자세가 담긴 이미지 데이터셋을 통해 요가 자세를 분류하는 인공지능 모델을 설계한다,

## 2. Data

### 2.1. Data Description

각 자세의 label을 제목으로 하는 6개의 sub-folder 내부에 각 class에 해당하는 이미지 데이터셋이 존재한다. 총 데이터의 개수(구글 드라이브 업로드 후 모델 학습에 사용한 데이터셋 개수)는 2495개로 0번 381개, 1번 404개, 2번 453개, 3번 320개, 4번 432개, 5번 455개의 이미지 데이터가 있다. 라벨 별 불균형 문제는 크지 않은 것으로 파악했다. 이미지 데이터의 file type은 jpg, jpeg, png 형식이다.

### 2.2. Data Pre-processing

Train dataset이 각 label 별로 폴더 내부에 정리되어 있기 때문에 ImageData generator를 통해 이미지를 전처리하는 것이 유리하다고 판단하였다. 또한 학습 데이터를 증강하면 학습 데이터가 증가함과 동시에 조금 더 일반화된 모델을 만들 수 있기에 이미지 데이터 증강을 실시하였다 [1].

먼저 픽셀값을 정규화하여 학습을 안정화시켰다. 다음으로 밝기 조정, 랜덤 확대, 회전, 좌우상하 이동, shear 연산 등을 통해 이미지를 증강하여 모델의 정확도를 높이고자 하였다. 또한 모델 검증을 위한 validation dataset의 크기는 전체 데이터셋의 20%로 설정하여 각각 train datagenerator와 val datagenerator를 정의하였다.

```
train_datagen = ImageDataGenerator(  
    rescale=1./255, #이미지 rescaling  
    zoom_range = 0.1, # 이미지를 랜덤 확대/축소  
    horizontal_flip=True, #좌우반전  
    brightness_range=[0.2,1.0], # 밝기 조절  
    rotation_range=30, # 회전제한 각도 30도  
    width_shift_range=0.2, # 좌우이동 20%  
    height_shift_range=0.2, # 상하이동 20%  
    shear_range=0.15, #shear 연산  
    validation_split=0.2)  
  
val_datagen = ImageDataGenerator(rescale=1./255, validation_split=0.2)
```

Fig1. 이미지 증강 및 전처리 코드

```

train_generator = train_datagen.flow_from_directory(
    directory=train_data_dir,
    batch_size = batch_size,
    target_size=(img_height, img_width),
    class_mode='sparse',
    shuffle = True,
    subset='training')

validation_generator = val_datagen.flow_from_directory(
    directory=train_data_dir,
    batch_size=batch_size,
    target_size=(img_height, img_width),
    class_mode='sparse',
    subset='validation')

```

Fig2. 데이터 분할

이후, `flow_from_directory` 함수를 호출하여 경로의 이미지들을 불러와 크기를 딥러닝 학습에 주로 사용되며 후에 학습에 사용된 ResNet152-v2의 입력값인 (224,224) 크기로 조정하고 batch size는 64로 설정하였다. 또한, 현재 폴더명이 정수형으로 라벨링 되어 있기에 class mode는 “sparse”로 설정하였다.

### 3. Analysis

이미지 분류를 위해서는 각 이미지들의 특징을 잘 추출해낼 수 있는 architecture가 필요하다. 특히, 본 프로젝트의 목적은 요가 자세를 분류해내는 것으로 각 클래스별 이미지를 확인하고 그에 맞는 데이터 전처리를 실시하는 것이 중요했다. 예를 들어, 상하반전을 실시할 시에 완전히 다른 자세로 인식되므로 이미지 증강과정에서 생략하였다. 또한, 회전 각도도 자세 분류에 영향이 가지 않을 만큼 작은 변동만 주었다.

또한, 딥러닝 모델을 학습시키기에 약 2500개의 이미지 데이터는 적은 양이므로 적은 양이라는 것을 파악했다. 따라서, Imagenet이라는 대량의 데이터를 통해 학습된 모델을 불러와 전이학습을 실시하는 것이 유리하다고 판단하였다.

## 4. Model

### 4.1. Classifier

요가자세 분류를 위한 모델로는 ResNet152-V2를 전이학습 후 fine tuning한 모델을 사용하였다. ResNet152-v2는 마이크로소프트 리서치에서 개발한 resnet의 변형이다. 총 152개의 층으로 이루어져 있으며, skip connection을 통해 gradient vanishing 문제를 완화하였다. V2는 기존의 ResNet보다 배치 정규화 등을 재조정하여 학습을 더 효율적으로 만든 모델이다. 이 모델은 케라스에서 Imagenet으로 사전 학습된 가중치를 제공하므로 전이학습에 유리하다고 판단했다. [2]

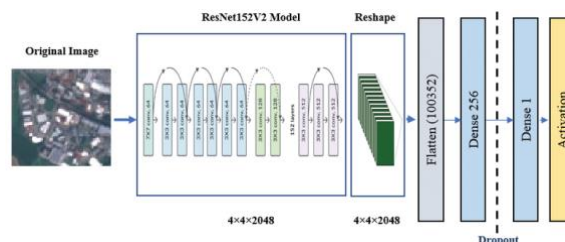


Fig3 ResNet152-v2 구조 [3]

## 4.2. Design Consideration

ResNet152는 ImageNet으로 사전학습된 가중치를 제공하므로, 전이학습 시 높은 정확도를 보여준다. 하지만, 작은 데이터셋에 대한 과적합 위험이 있고, 새로운 데이터를 구분하는 특징에 대한 학습이 잘 이루어지지 않은 상태이다. 따라서, fine tuning을 통해 base model의 동결을 풀고 낮은 learning rate로 가중치를 조정한다. 전이학습된 모델의 구조는 다음과 같다. 마지막 출력층에는 공간적인 특징을 도출해주는 global average pooling layer와 클래스 분류에 사용할 softmax dense 층을 삽입하였다. 추가한 모델의 layer와 파라미터 수는 다음과 같다.

global_average_pooling2d (GlobalAveragePooling2D)	(None, 2048)	0	['post_relu[0][0]']
dense (Dense)	(None, 6)	12294	['global_average_pooling2d[0][0]']
=====			
Total params: 58,343,942			
Trainable params: 12,294			
Non-trainable params: 58,331,648			

Fig 4. Model Design

또한, 이 과정에서 모델의 과적합을 방지하기 위해 callback으로 early stopping과 model checkpoint를 도입하였다.

## 5. Experiments

### 5.1. Settings

ResNet152, Xception, EfficientNet 세가지 모델로 전이학습을 진행하며 실험을 실시하였다. Train: validation = 8:2 로 설정하였으며, 이미지 크기와 batch size는 (224,224), 64로 고정하였다. Optimizer는 Adam을 사용하였으며 learning rate는 base 모델 동결 시에는 0.01, fine tuning 시에는 0.0001로 고정하였다.

### 5.2. Performance Metrics

Kaggle의 public test accuracy를 최종 모델 선정 지표로 활용하였다. Validation accuracy보다 test accuracy가 높게 나오는 것을 파악하여, 위의 지표를 기준으로 평가하는 것이 유리하다고 판단하였다.

### 5.3. Results

Resnet152-V2 모델은 0.97427, Xception 모델은 0.95819, EfficientNet 모델은 0.94533의 정확도를 보여주었다. 따라서 Resnet 152-V2 모델을 최종 모델로 선정하였다. Resnet 152-V2 모델의 학습과정에 따른 validation loss와 validation accuracy 변화는 다음과 같다.

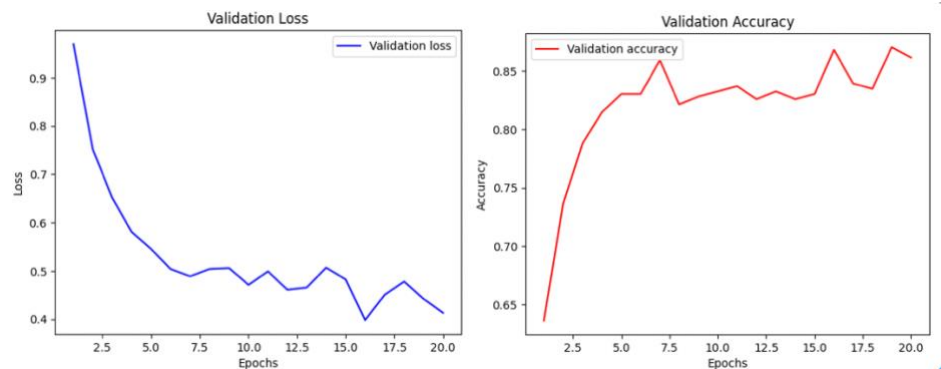


Fig 4. ResNet152\_V2 validation loss & accuracy

## 6. Discussion and Limitation

케라스에서 ImageNet이라는 대량의 데이터로 학습된 모델의 가중치를 제공해주므로, 이를 전이학습하여 다양한 이미지 분류 문제에 활용할 수 있다. 다양한 모델을 전이학습 시켜보며 데이터 특징 별로 분류에 적합한 모델이 존재한다는 사실을 파악할 수 있었다.

이 프로젝트는 이미지 증강과 fine tuning을 통해 요가 자세 분류라는 목적에 맞는 설계를 진행했다는 것에서 의의가 있다. 특히, 이미지 증강의 과정에서도 무조건 많은 데이터로 증강하는 것에 목적을 두지 않고, 자세 분류라는 데이터의 특성에 맞춰 그 변형 정도를 조정했다. 또한, 기존 베이스 모델의 layer를 동결하고 학습시켰을 때 약 86%정도였던 정확도가, fine tuning을 통해 약 97%까지 상승했다는 점에서 fine tuning과정의 중요성을 파악할 수 있었다.

하지만, 배경의 요소가 관여될 수 있다는 점, 더 다양한 모델을 이용하여 전이 학습 시켜보지 않았다는 점에서 한계가 존재한다. 이후에 Segmentation기법을 통해 사람의 모습만 추출하여 모델을 학습시킨다면 더 좋은 결과를 얻을 수 있을 것이라고 판단한다. 또한, 위에서 실험에 사용한 3가지 모델 외에도 mobilenet, Densenet 등 다양한 모델 등을 실험한 후 앙상블 기법을 실시한다면 더 나은 결과를 도출할 수 있을 것이라고 기대한다.

## 7. References

- [1] “[딥러닝] Keras ImageDataGenerator를 이용한 이미지 데이터 증강 (Data Augmentation)”, <https://velog.io/@ym980118/%EB%94%A5%EB%9F%AC%EB%8B%9D-Keras-ImageDataGenerator%EB%A5%BC-%EC%9D%B4%EC%9A%A9%ED%95%9C-%EC%9D%B4%B%AF%B8%EC%A7%80-%EB%8D%B0%EC%9D%B4%ED%84%B0-%EC%A6%9D%EA%B0%95-Data-Augmentation> (접근일 05.31.23)
- [2] Microsoft Research (2016). Identity mappings in deep residual networks. Computer Vision – ECCV 2016, 630–645. [https://doi.org/10.1007/978-3-319-46493-0\\_38](https://doi.org/10.1007/978-3-319-46493-0_38)
- [3] Kittusamy, Kousalya & Krishnakumar, B. & Aswath, A. & Gowtham, P. & Vishal, S.. (2021). Terrain identification and land price estimation using deep learning. AIP Conference Proceedings. 2387. 140030. 10.1063/5.0068625.