Data Augmentation

SEUNGWOO LEE

Table Of Contents

Introducing augmentation

Convolutional Neural Networks in TensorFlow

deeplearning.ai

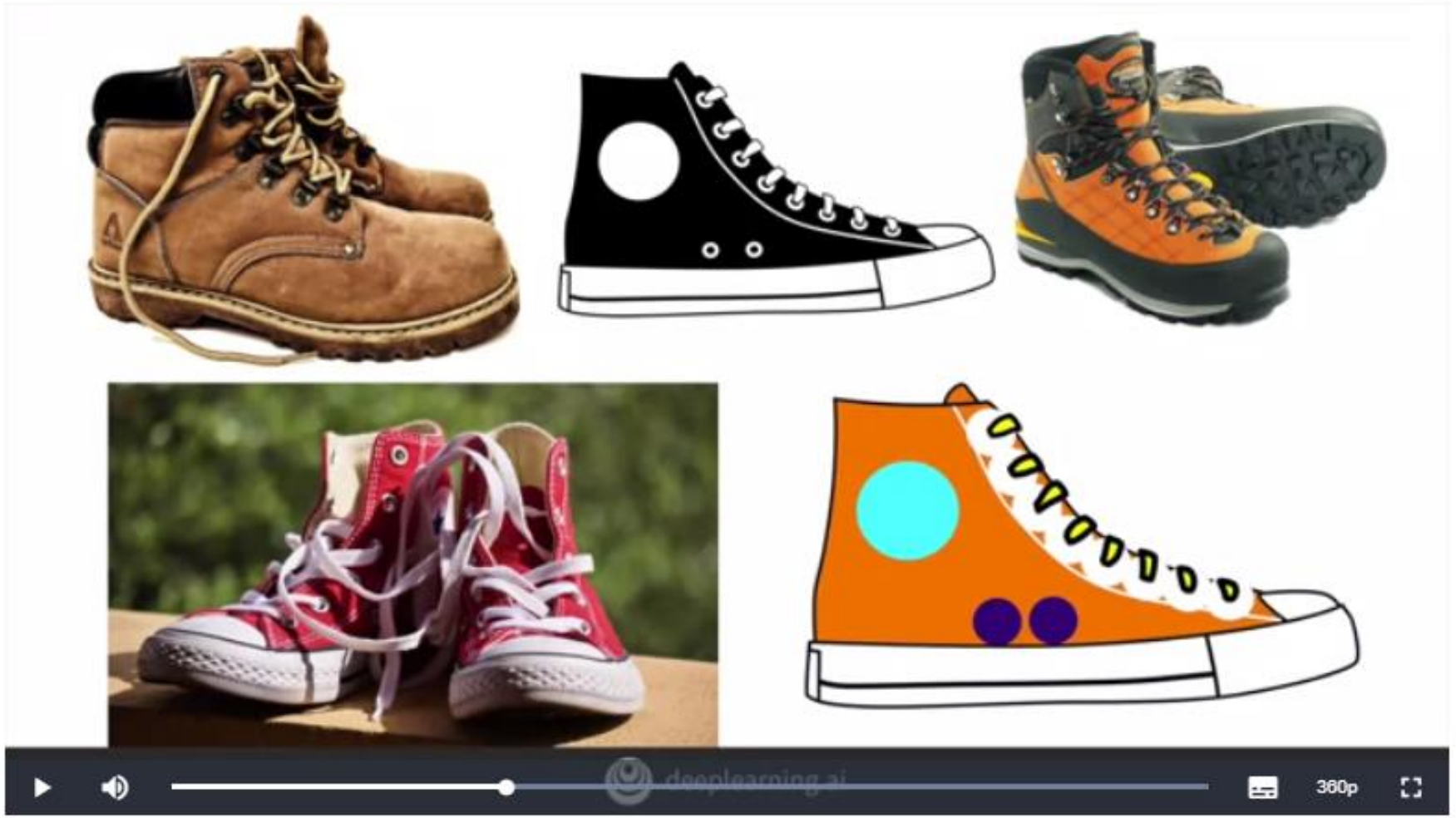
★★★★½ 4.8(694개의 평가) | 18K명의 학생이 등록함

TensorFlow in Practice 전문 분야에서 4의 강좌 2

무료로 등록

Dataset들 중에서 신발이라는 레이블을 가지고

다음과 같은 이미지들을 학습시켰다



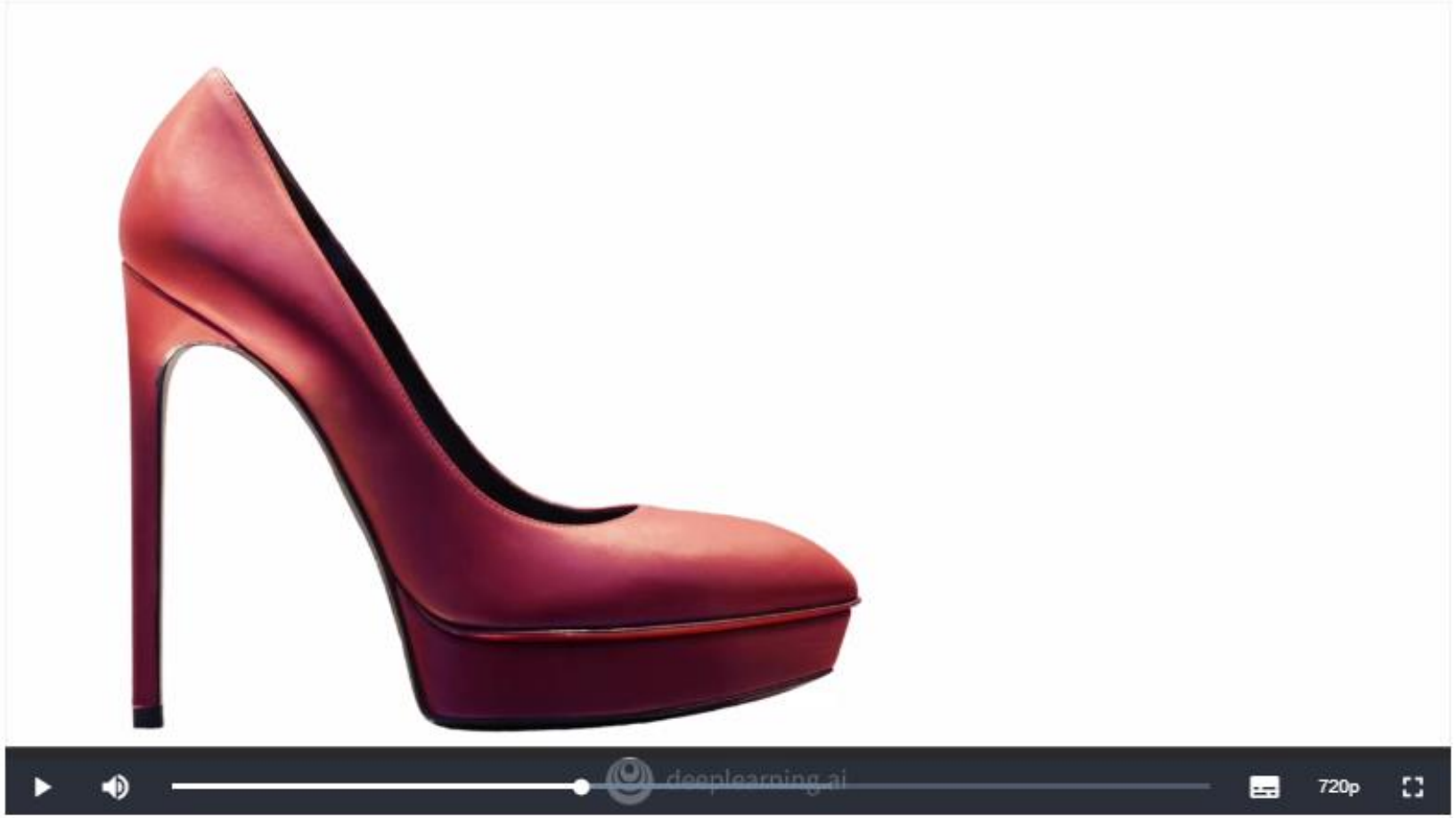
그리고 학습한 데이터를 토대로 이미지를 `.predict` 하면

신발이라고 나올 것이다.

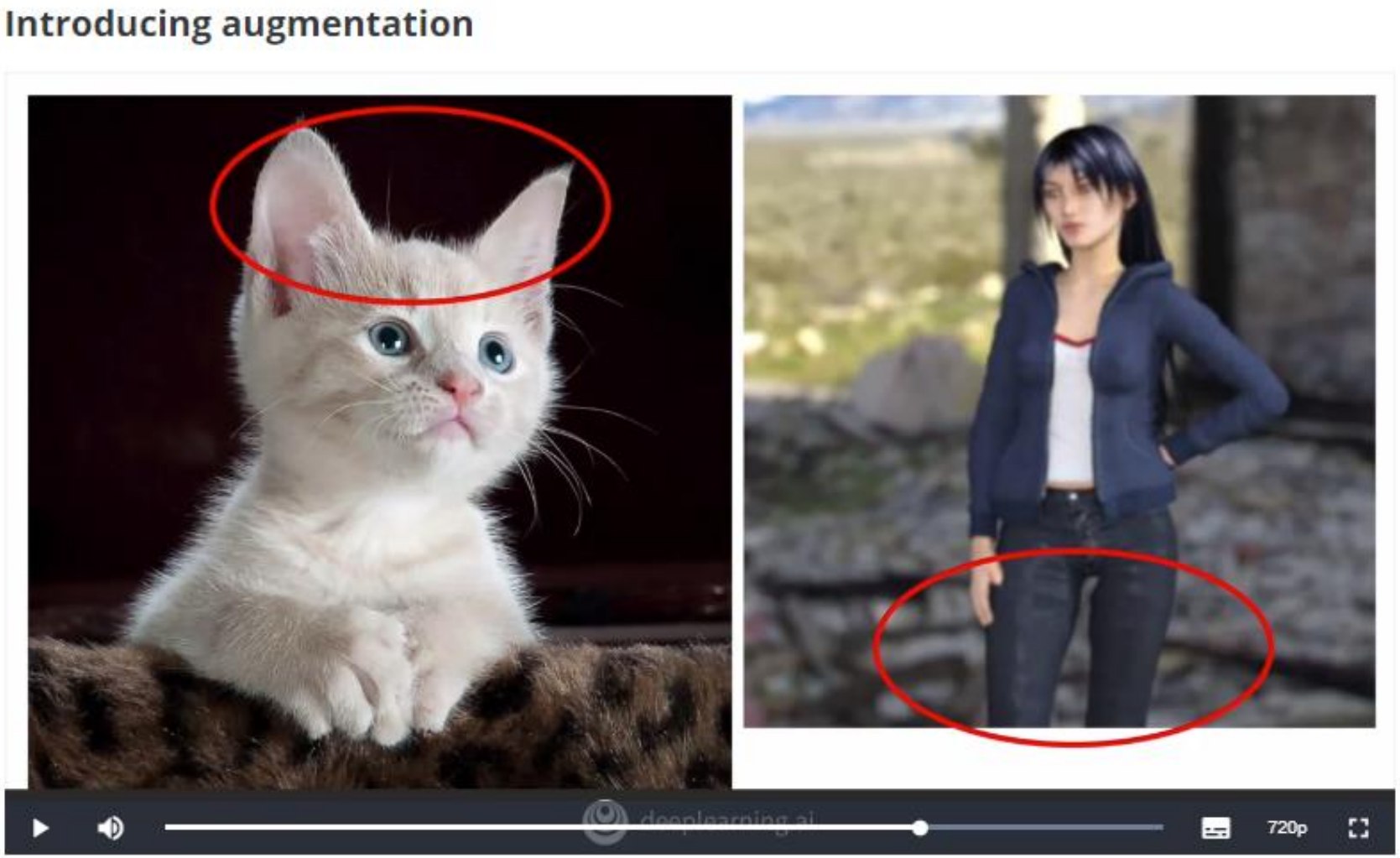
Introducing augmentation

그런데 하이힐을 predict 라고 하면 신발이라고 나오지 않을 것이다.
왜냐하면 하이힐에 대한 데이터가 주어지지 않았기 때문이다.

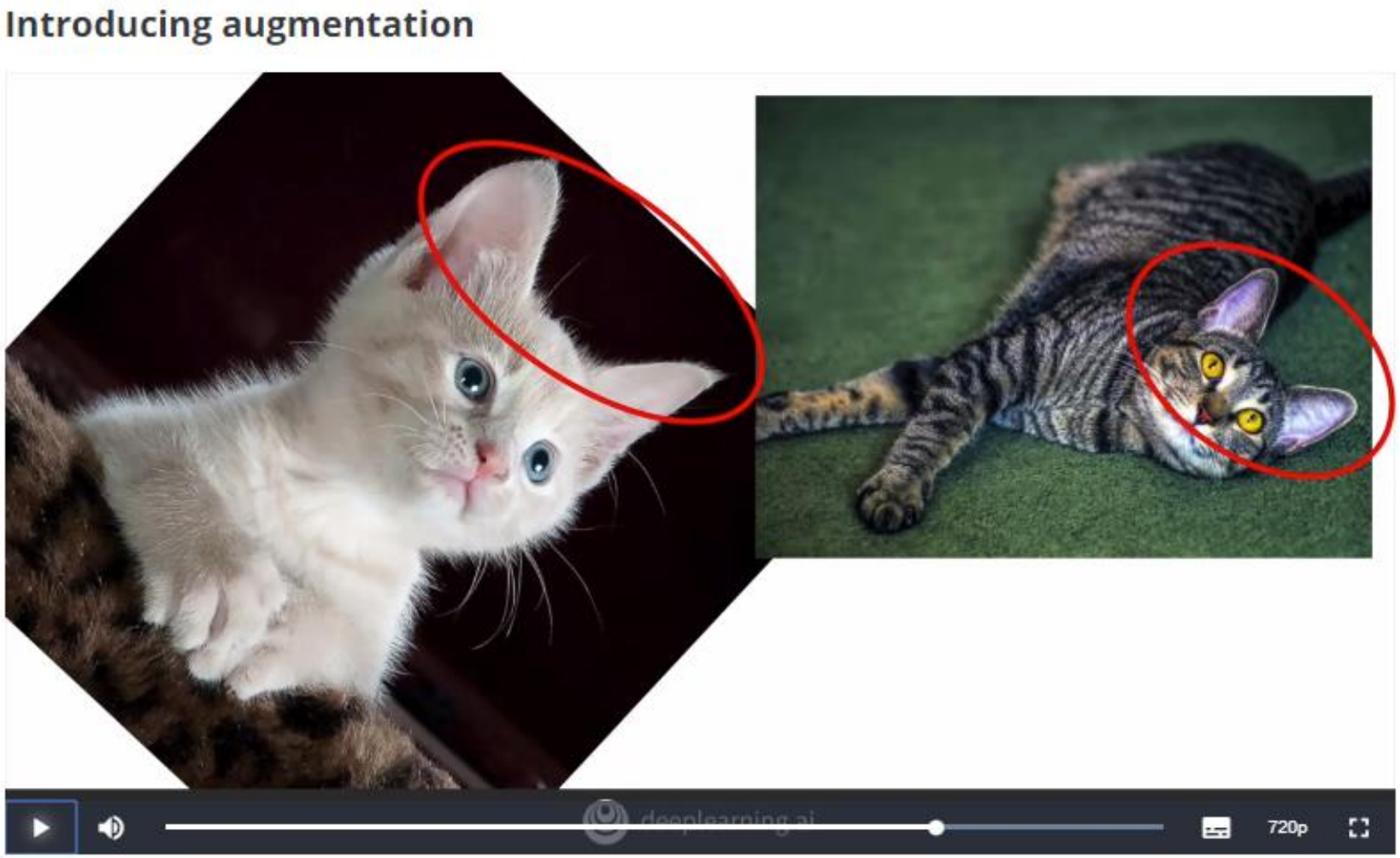
Introducing augmentation



다음과 같은 이미지들을 학습을 시켜보려고 하는데

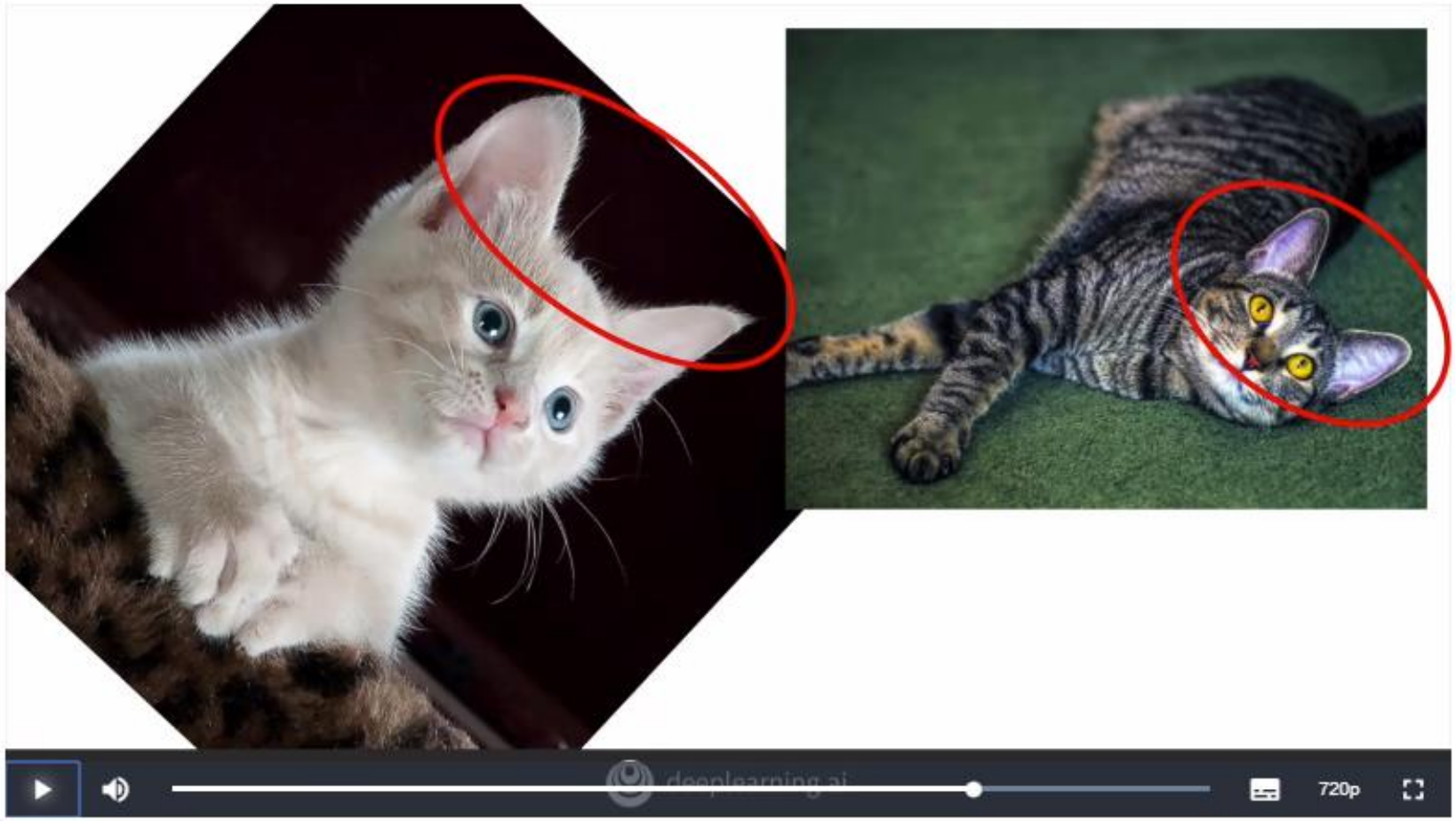


이렇게 이미지를 틀면 어떻게 될까?

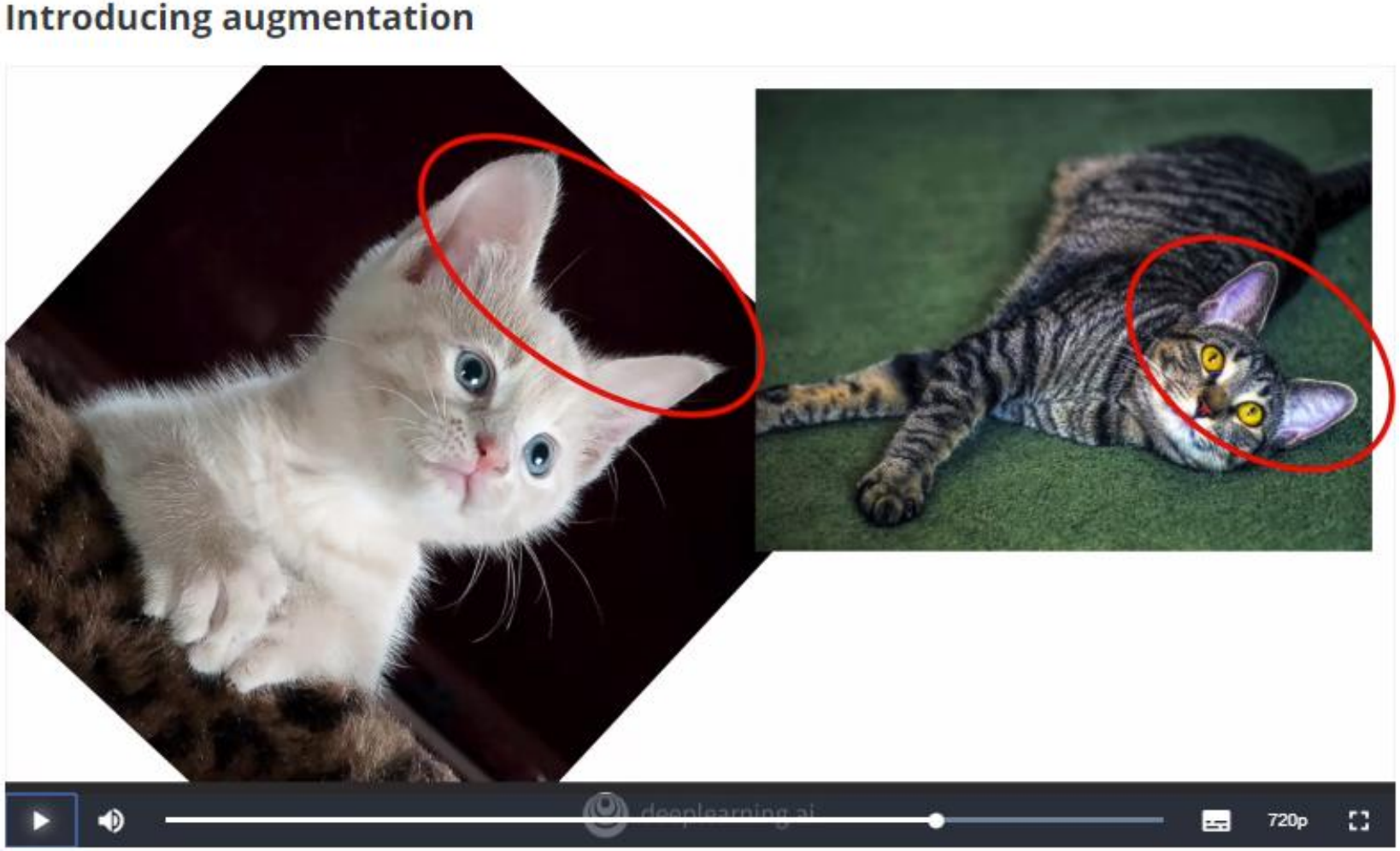


틀어버린 이미지에 대한 정보가 없기 때문에 Cat이라고 레이블을 하지 못할 것이다.

Introducing augmentation



바로 여기에서 "data augmentation" 이라고 하는 방법을 사용하면 이미지를 효과적으로 학습시킬 수 있다.

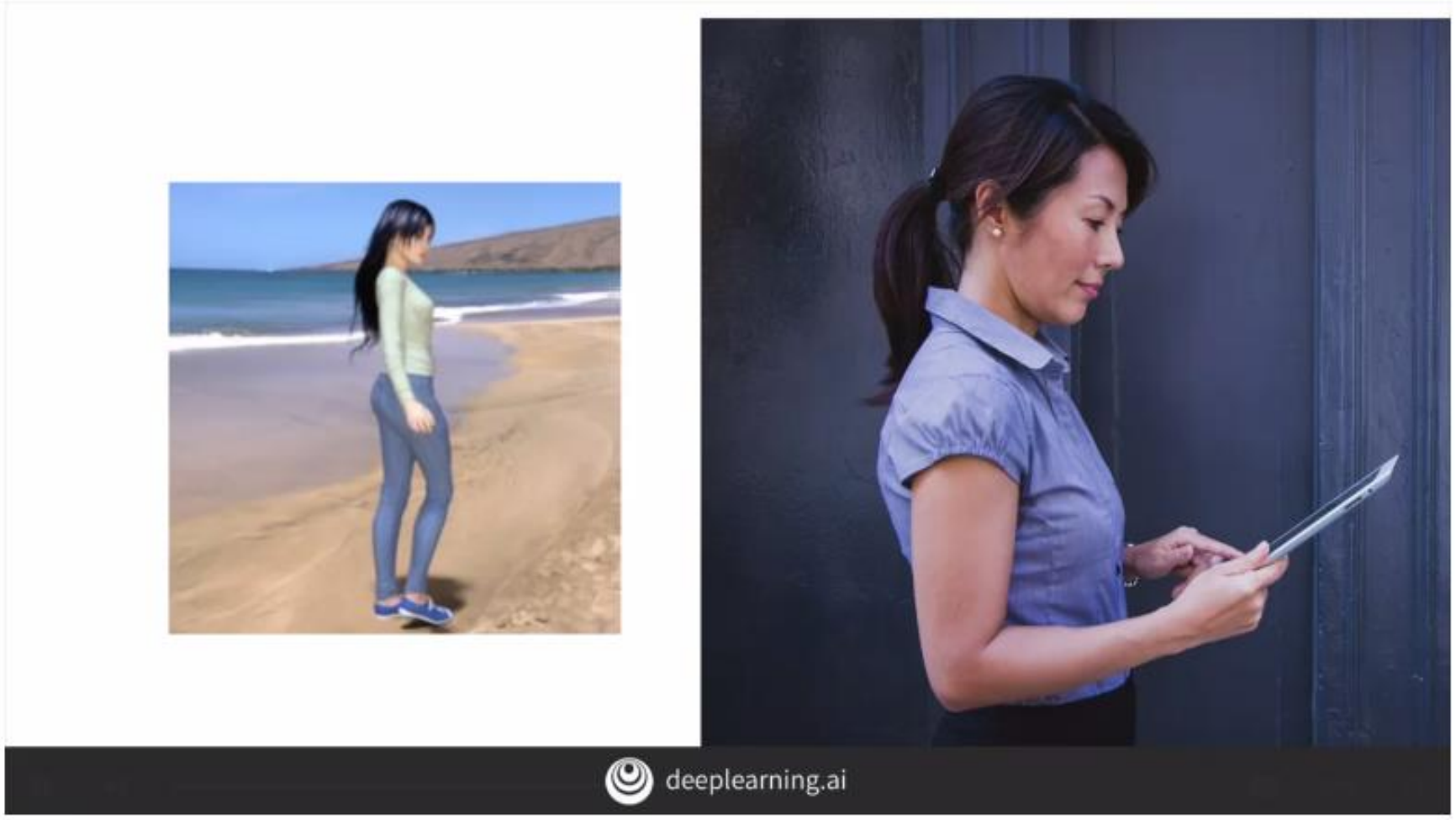


```python
import tensorflow as tf
from tf.keras.preprocessing.image import ImageDataGenerator

train_datagen = ImageDataGenerator(
    rescale = 1./255
)

train_datagen = ImageDataGenerator(
    rescale= 1./255,         # 이미지 Rescale
    rotation_range=40,       # 얼마나 이미지를 Rotation 시킬 것이냐 (0~40도)
    width_shift_range=0.2, # Width에서 얼만큼 이미지를 shift 시킬 것이냐
    height_shift_range=0.2 # Height에서 얼만큼 이미지를 shift 시킬 것이냐
    shear_range=0.2,         # 이미지를 20%까지 Shear 함
    zoom_range=0.2,          # 이미지를 20%까지 Zoom 함
    horizontal_flip=True,  # 이미지 뒤집기
    fill_mode='nearest'    # 인근의 픽셀을 가지고 와서 비어있는 픽셀을 Filling
)
```

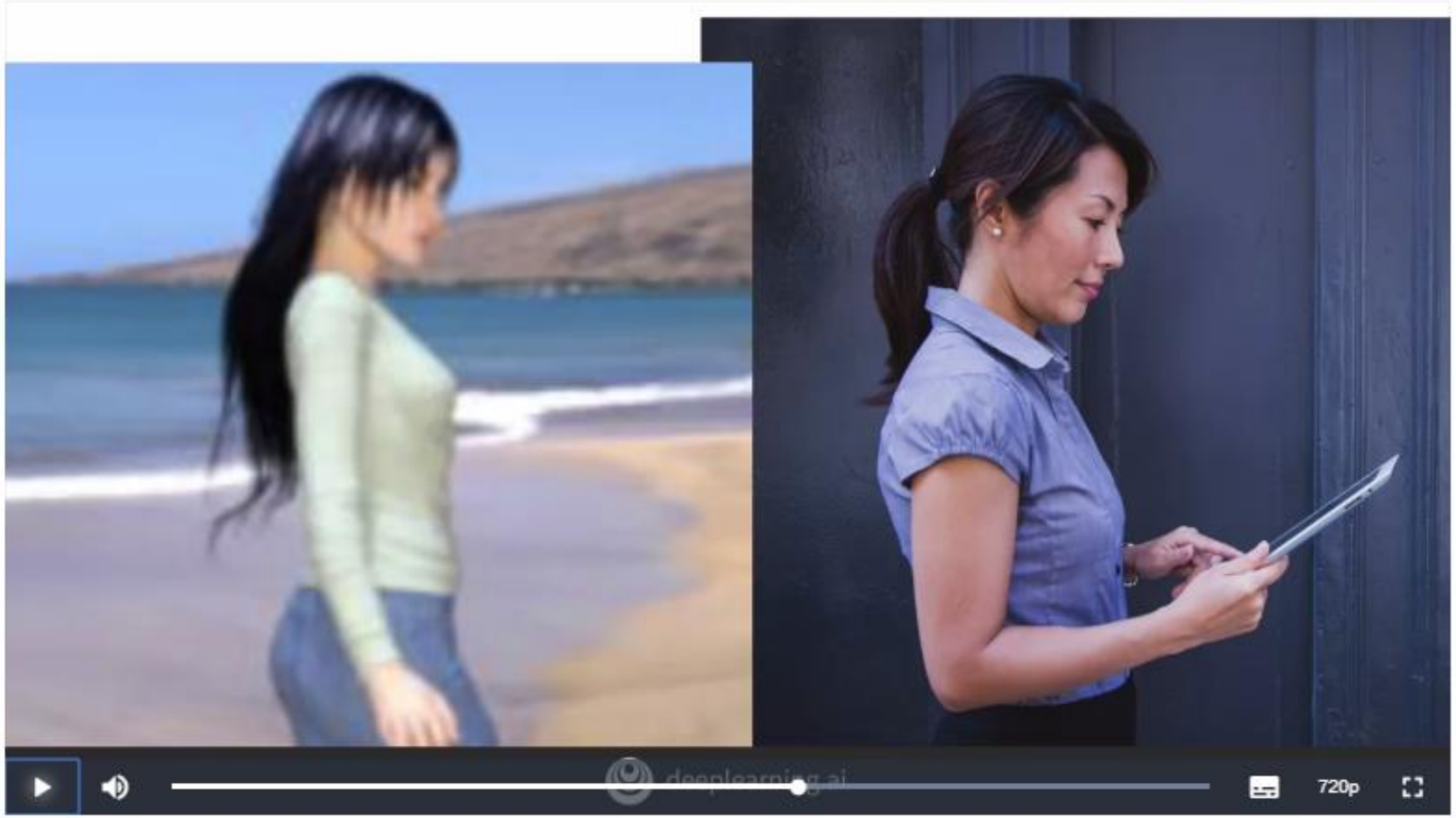
Shearing

Shearing

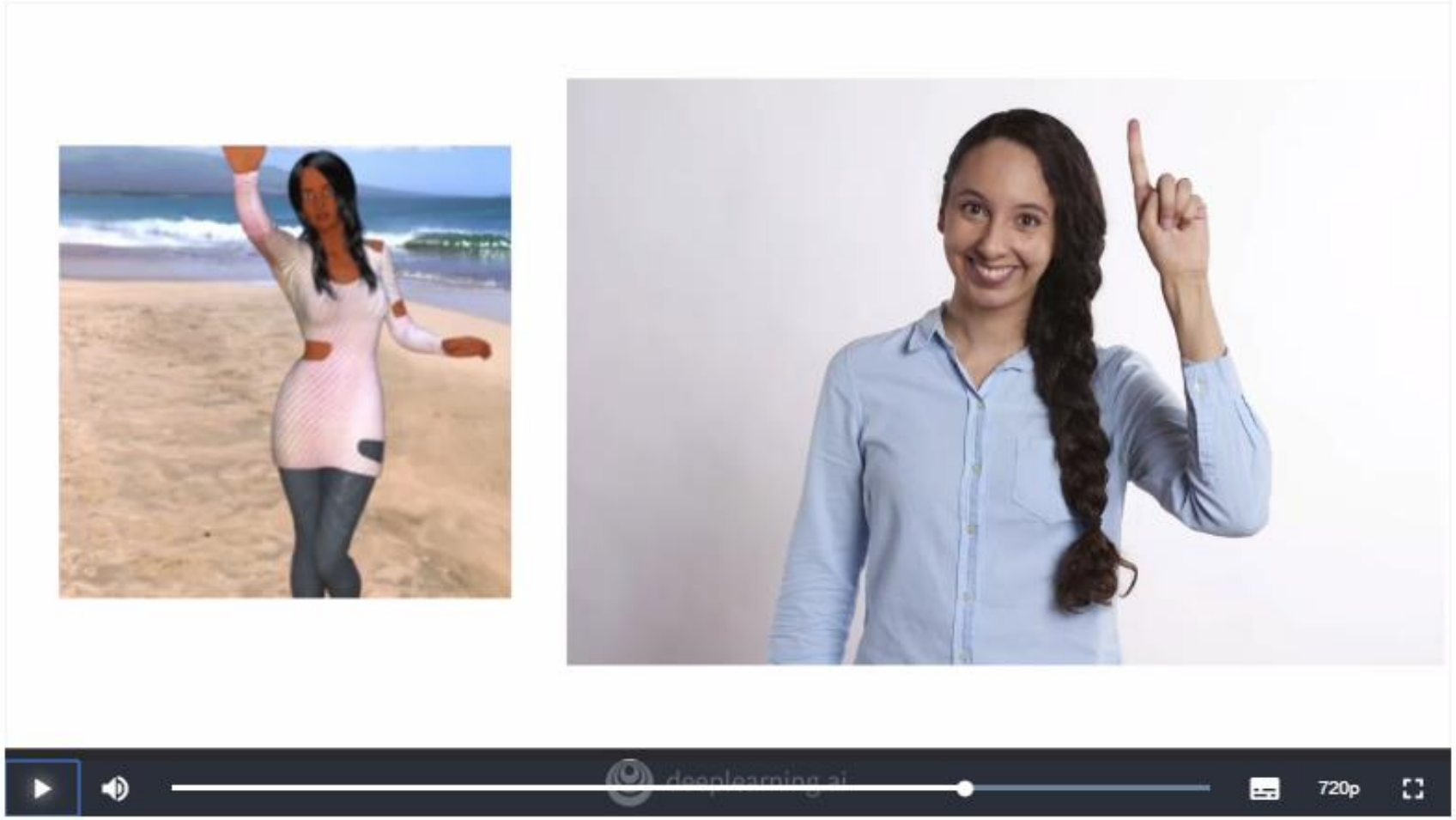
Zoom Range



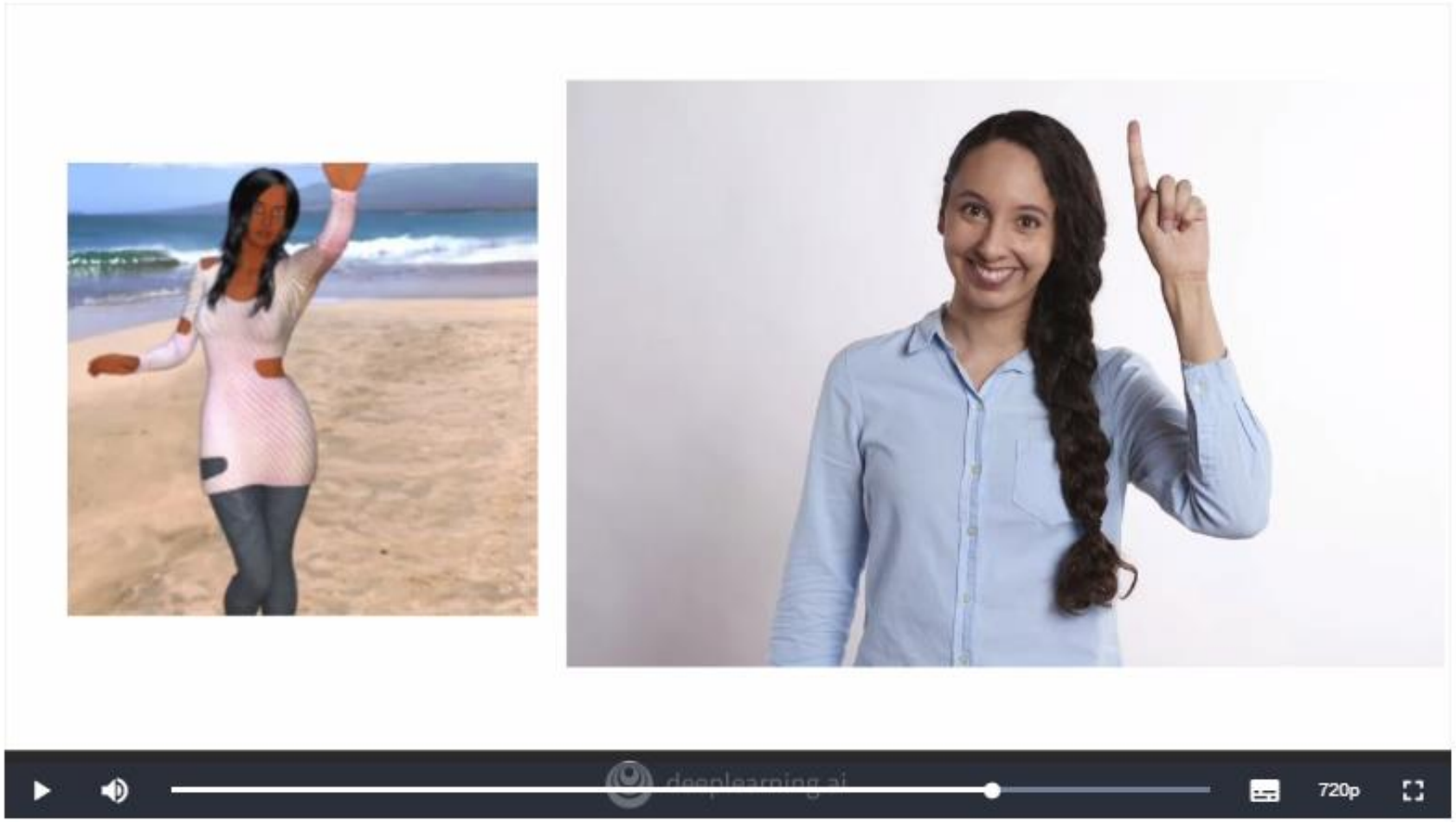
Zoom Range



Horizontal Flip



Horizontal Flip



Cats-v-Dogs-Augmentation.ipynb

File Edit View Insert Runtime Tools Help

COMMENT SHARE

CODE TEXT CELL CELL BUSY EDITING

```
validation_generator = test_datagen.flow_from_directory(
        validation_dir,
        target_size=(150, 150),
        batch_size=20,
        class_mode='binary')

history = model.fit_generator(
      train_generator,
      steps_per_epoch=100,  # 2000 images = batch_size * steps
      epochs=100,
      validation_data=validation_generator,
      validation_steps=50,  # 1000 images = batch_size * steps
      verbose=2)
```

```
--2019-02-12 07:05:37--  https://storage.googleapis.com/mledu-datasets/cats_and_dogs_filtered.zip
Resolving storage.googleapis.com... 2607:f8b0:4001:c1b::80, 209.85.234.128
Connecting to storage.googleapis.com|2607:f8b0:4001:c1b::80|:443... connected.
WARNING: cannot verify storage.googleapis.com's certificate, issued by 'CN=Google Internet Authority G3,O=Google Trust Services,C=US':
  Unable to locally verify the issuer's authority.
HTTP request sent, awaiting response... 200 OK
Length: 68606236 (65M) [application/zip]
Saving to: '/tmp/cats_and_dogs_filtered.zip'

/tmp/cats_and_dogs_ 100%[===================>]  65.43M   165MB/s   in 0.4s

2019-02-12 07:05:38 (165 MB/s) - '/tmp/cats_and_dogs_filtered.zip' saved [68606236/68606236]

Found 2000 images belonging to 2 classes.
Found 1000 images belonging to 2 classes.
Epoch 1/100
100/100 - 6s - loss: 0.6914 - acc: 0.5345 - val_loss: 0.6872 - val_acc: 0.5290
Epoch 2/100
100/100 - 5s - loss: 0.6538 - acc: 0.6040 - val_loss: 0.6354 - val_acc: 0.6500
Epoch 3/100
100/100 - 5s - loss: 0.6069 - acc: 0.6675 - val_loss: 0.5961 - val_acc: 0.6960
Epoch 4/100
100/100 - 5s - loss: 0.5658 - acc: 0.6995 - val_loss: 0.5799 - val_acc: 0.7120
Epoch 5/100
100/100 - 5s - loss: 0.5358 - acc: 0.7305 - val_loss: 0.5753 - val_acc: 0.7060
```

Cats-v-Dogs-Augmentation.ipynb

File Edit View Insert Runtime Tools Help

COMMENT SHARE

CODE TEXT CELL CELL

CONNECTED EDITING

```
plt.title('Training and validation accuracy')

plt.figure()

plt.plot(epochs, loss, 'bo', label='Training Loss')
plt.plot(epochs, val_loss, 'b', label='Validation Loss')
plt.title('Training and validation loss')
plt.legend()

plt.show()
```

학습한 데이터의 정확도 100%

Validation 정확도 70%

https://devrant.com/rants/1406998/always-use-analogy-when-explaining-things

Cats-v-Dogs-Augmentation.ipynb

File Edit View Insert Runtime Tools Help

COMMENT SHARE

CODE TEXT CELL CELL BUSY EDITING

```
model.compile(loss='binary_crossentropy',
              optimizer=RMSprop(lr=1e-4),
              metrics=['acc'])

# This code has changed. Now instead of the ImageGenerator just rescaling
# the image, we also rotate and do other operations
# Updated to do image augmentation
train_datagen = ImageDataGenerator(
      rescale=1./255,
      rotation_range=40,
      width_shift_range=0.2,
      height_shift_range=0.2,
      shear_range=0.2,
      zoom_range=0.2,
      horizontal_flip=True,
      fill_mode='nearest')

test_datagen = ImageDataGenerator(rescale=1./255)

# Flow training images in batches of 20 using train_datagen generator
train_generator = train_datagen.flow_from_directory(
        train_dir,  # This is the source directory for training images
        target_size=(150, 150),  # All images will be resized to 150x150
        batch_size=20,
        # Since we use binary_crossentropy loss, we need binary labels
        class_mode='binary')

# Flow validation images in batches of 20 using test_datagen generator
validation_generator = test_datagen.flow_from_directory(
        validation_dir,
        target_size=(150, 150),
        batch_size=20,
        class_mode='binary')

history = model.fit_generator(
      train_generator,
      steps_per_epoch=100,  # 2000 images = batch_size * steps
      epochs=100,
      validation_data=validation_generator,
      validation_steps=50,  # 1000 images = batch_size * steps
      verbose=2)
```

똑같은 코드에 Image Augmentation 만 적용

```
--2019-02-12 07:34:07--  https://storage.googleapis.com/mledu-datasets/cats_and_dogs_filtered.zip
Resolving storage.googleapis.com... 2607:f8b0:4001:c00::80, 64.233.191.128
Connecting to storage.googleapis.com|2607:f8b0:4001:c00::80|:443... connected.
WARNING: cannot verify storage.googleapis.com's certificate, issued by 'CN=Google Internet Authority G3,O=Google Trust Services,C=US':
  Unable to locally verify the issuer's authority.
HTTP request sent, awaiting response... 200 OK
```

Cats-v-Dogs-Augmentation.ipynb

File Edit View Insert Runtime Tools Help

COMMENT SHARE

CODE TEXT CELL CELL BUSY EDITING

```
        validation_dir,
        target_size=(150, 150),
        batch_size=20,
        class_mode='binary')

history = model.fit_generator(
      train_generator,
      steps_per_epoch=100,  # 2000 images = batch_size * steps
      epochs=100,
      validation_data=validation_generator,
      validation_steps=50,  # 1000 images = batch_size * steps
      verbose=2)
```

```
--2019-02-12 07:34:07--  https://storage.googleapis.com/mledu-datasets/cats_and_dogs_filtered.zip
Resolving storage.googleapis.com... 2607:f8b0:4001:c00::80, 64.233.191.128
Connecting to storage.googleapis.com|2607:f8b0:4001:c00::80|:443... connected.
WARNING: cannot verify storage.googleapis.com's certificate, issued by 'CN=Google Internet Authority G3,O=Google Trust Services,C=US':
  Unable to locally verify the issuer's authority.
HTTP request sent, awaiting response... 200 OK
Length: 68606236 (65M) [application/zip]
Saving to: '/tmp/cats_and_dogs_filtered.zip'

/tmp/cats_and_dogs_ 100%[===================>]  65.43M  81.1MB/s    in 0.8s

2019-02-12 07:34:08 (81.1 MB/s) - '/tmp/cats_and_dogs_filtered.zip' saved [68606236/68606236]

Found 2000 images belonging to 2 classes.
Found 1000 images belonging to 2 classes.
Epoch 1/100
100/100 - 14s - loss: 0.6922 - acc: 0.5235 - val_loss: 0.6906 - val_acc: 0.5070
Epoch 2/100
100/100 - 13s - loss: 0.6807 - acc: 0.5695 - val_loss: 0.6687 - val_acc: 0.5840
Epoch 3/100
100/100 - 13s - loss: 0.6613 - acc: 0.6060 - val_loss: 0.6237 - val_acc: 0.6600
Epoch 4/100
100/100 - 13s - loss: 0.6408 - acc: 0.6320 - val_loss: 0.5980 - val_acc: 0.6920
Epoch 5/100
100/100 - 12s - loss: 0.6240 - acc: 0.6430 - val_loss: 0.5722 - val_acc: 0.7150
Epoch 6/100
100/100 - 12s - loss: 0.6077 - acc: 0.6700 - val_loss: 0.5621 - val_acc: 0.7020
Epoch 7/100
100/100 - 12s - loss: 0.5954 - acc: 0.6740 - val_loss: 0.5549 - val_acc: 0.7200
```

처음에는 Accuracy가 낮음

→ 이미지를 Augment. 하여 학습하기 때문

Cats-v-Dogs-Augmentation.ipynb

File Edit View Insert Runtime Tools Help

COMMENT SHARE

CODE TEXT CELL CELL

CONNECTED EDITING

```
Epoch 88/100
100/100 - 13s - loss: 0.3679 - acc: 0.8380 - val_loss: 0.4491 - val_acc: 0.7810
Epoch 89/100
100/100 - 13s - loss: 0.3595 - acc: 0.8370 - val_loss: 0.4169 - val_acc: 0.8120
Epoch 90/100
100/100 - 13s - loss: 0.3597 - acc: 0.8405 - val_loss: 0.4379 - val_acc: 0.8140
Epoch 91/100
100/100 - 13s - loss: 0.3611 - acc: 0.8325 - val_loss: 0.4223 - val_acc: 0.7920
Epoch 92/100
100/100 - 13s - loss: 0.3622 - acc: 0.8315 - val_loss: 0.3903 - val_acc: 0.8190
Epoch 93/100
100/100 - 13s - loss: 0.3677 - acc: 0.8295 - val_loss: 0.3752 - val_acc: 0.8090
Epoch 94/100
100/100 - 13s - loss: 0.3604 - acc: 0.8345 - val_loss: 0.4889 - val_acc: 0.7790
Epoch 95/100
100/100 - 13s - loss: 0.3557 - acc: 0.8480 - val_loss: 0.4221 - val_acc: 0.7970
Epoch 96/100
100/100 - 13s - loss: 0.3540 - acc: 0.8485 - val_loss: 0.4615 - val_acc: 0.7820
Epoch 97/100
100/100 - 13s - loss: 0.3454 - acc: 0.8455 - val_loss: 0.4118 - val_acc: 0.8010
Epoch 98/100
100/100 - 13s - loss: 0.3369 - acc: 0.8505 - val_loss: 0.3930 - val_acc: 0.8170
Epoch 99/100
100/100 - 12s - loss: 0.3580 - acc: 0.8340 - val_loss: 0.4085 - val_acc: 0.8210
Epoch 100/100
100/100 - 12s - loss: 0.3293 - acc: 0.8615 - val_loss: 0.4549 - val_acc: 0.8080
```

이미지가 점차 학습되면서

Accuracy 가 증가

```
[ ] import matplotlib.pyplot as plt
    acc = history.history['acc']
    val_acc = history.history['val_acc']
    loss = history.history['loss']
    val_loss = history.history['val_loss']

    epochs = range(len(acc))

    plt.plot(epochs, acc, 'bo', label='Training accuracy')
    plt.plot(epochs, val_acc, 'b', label='Validation accuracy')
    plt.title('Training and validation accuracy')

    plt.figure()

    plt.plot(epochs, loss, 'bo', label='Training Loss')
    plt.plot(epochs, val_loss, 'b', label='Validation Loss')
```

Cats-v-Dogs-Augmentation.ipynb

File Edit View Insert Runtime Tools Help

COMMENT SHARE

CODE TEXT CELL CELL

CONNECTED EDITING

```
plt.plot(epochs, loss, 'bo', label='Training Loss')
plt.plot(epochs, val_loss, 'b', label='Validation Loss')
plt.title('Training and validation loss')
plt.legend()

plt.show()
```

학습 시 Accuracy가 점차 증가하는 것을 확인할 수 있고

Validation시 Loss가 점차 낮아지는 것을 확인할 수 있음

이를 통해서 Overfitting을 해결했다고 할 수 있음

더 많은 Epoch를 통해서 Accuracy를 높일 수 있음

Horse-or-Human-WithAugmentation.ipynb

File Edit View Insert Runtime Tools Help

COMMENT SHARE

CODE TEXT CELL CELL

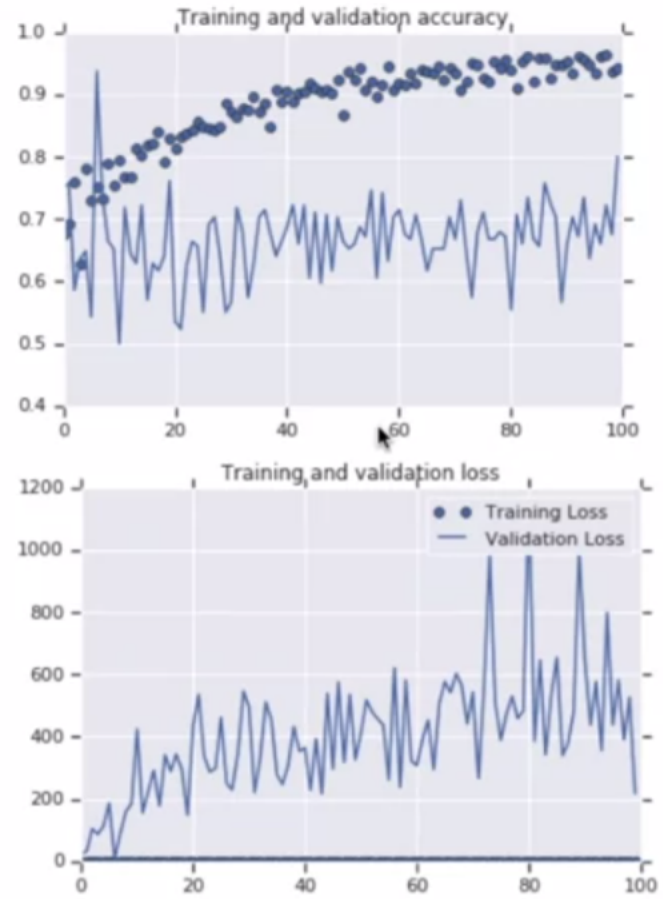
CONNECTED EDITING

Table of contents Code snippets

Building a Small Model from Scratch

SECTION

```
plt.plot(epochs, loss, 'bo', label='Training Loss')
plt.plot(epochs, val_loss, 'b', label='Validation Loss')
plt.title('Training and validation loss')
plt.legend()

plt.show()
```



주의할 점은 Training상태에서만 적용한 후 Test 상태에서 사용하지 않으면 다음과 같이 Validation Graph가 미친듯이 널뛰기 하는 것을 볼 수 있다