

ขนาดข้อมูลที่ใช้ในการฝึก เทรน Models

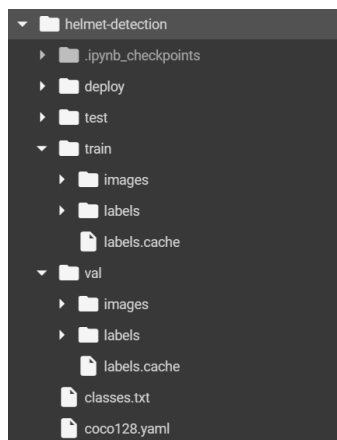
แหล่งที่มา: <https://www.kaggle.com/datasets/aneesarom/rider-with-helmet-without-helmet-number-plate>

1. ไฟล์รูปภาพ 102 รูปภาพ



2. ไฟล์ข้อความ 102 ไฟล์

Directory File / Folder



วิธีการดำเนินการ

Installing the Dependencies

!pip install ultralytics # เป็น Framework ของ YOLO (You only look once)

!pip install torch

!pip3 install torch torchvision torchaudio --index-url <https://download.pytorch.org/whl/cu118>

```
!git clone https://github.com/ultralytics/yolov5 # clone repo
```

```
%cd yolov5
```

```
!pip install -r requirements.txt # install dependencies
```

ติดตั้ง Library ที่เกี่ยวข้อง

```
import cv2
```

```
import pandas
```

```
import numpy as np
```

```
import matplotlib.pyplot as plt # พล็อตรูปภาพ
```

```
import matplotlib.image as mpimg
```

```
from pathlib import Path
```

```
from collections import Counter
```

```
import os
```

ติดตั้ง Configuration ของ YOLO5 ภายในไฟล์

```
yaml_path = '/content/drive/MyDrive/helmet-detection/coco128.yaml'
```

```
with open(yaml_path, 'r') as file:
```

```
    yaml_contents = file.read()
```

```
train: /train/images
val: /val/images

nc: 4

names: ["with helmet", "without helmet", "rider", "number plate"]
```

```
def count_class_instances(folder_path): # นับจำนวนข้อมูล
```

```
    labels_folder = Path(folder_path) / 'labels'
```

```
    class_counts = Counter()
```

```
    for label_file in labels_folder.glob('*.txt'):
```

```
        with open(label_file, 'r') as file:
```

```
            annotations = file.readlines()
```

```
            for annotation in annotations:
```

```
                class_id = int(annotation.split()[0])
```

```
                class_counts[class_id] += 1
```

```
    return class_counts
```

```
train_class_counts = count_class_instances('/content/drive/MyDrive/helmet-detection/train')
```

```
val_class_counts = count_class_instances('/content/drive/MyDrive/helmet-detection/val')
```

```
Train Counts: Counter({2: 117, 3: 113, 1: 89, 0: 64})
Validation Counts: Counter({2: 23, 3: 22, 1: 15, 0: 13})
```

1. with helmet : 117

2. without helmet : 3

3. rider : 89

4. number plate : 64

ปรับแต่งรูปภาพ

def augment_image(image):

```
flipped = cv2.flip(image, 1)

hsv = cv2.cvtColor(image, cv2.COLOR_BGR2HSV)

h, s, v = cv2.split(hsv)

v = cv2.add(v, np.random.randint(30, 70))

final_hsv = cv2.merge((h, s, v))

bright_adjusted = cv2.cvtColor(final_hsv, cv2.COLOR_HSV2BGR)

return flipped, bright_adjusted
```

def demonstrate_augmentation(folder_path, class_id=0):

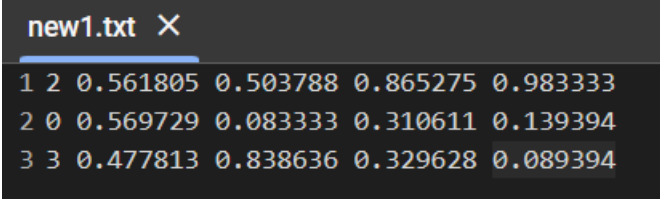
labels_folder = Path(folder_path) / 'labels' # สร้าง Path ตาม Parameter folder_path และ Concat ด้วย Subfolder 'labels'

images_folder = Path(folder_path) / 'images' # สร้าง Path ตาม Parameter folder_path และ Concat ด้วย Subfolder 'images'

for label_file in labels_folder.glob('*.txt'): # ไฟล์ที่มีนามสกุลไฟล์ .txt ใน Folder Labels

with open(label_file, 'r') as file:

annotations = file.readlines()



new1.txt ✕						
1	2	0.561805	0.503788	0.865275	0.983333	
2	0	0.569729	0.083333	0.310611	0.139394	
3	3	0.477813	0.838636	0.329628	0.089394	

ตัวอย่างข้อมูลภายในไฟล์ Labels/new1.txt

if any(int(ann.split()[0]) == class_id for ann in annotations):

image_name = label_file.stem

image_path = images_folder / f'{image_name}.jpg'

```

img = cv2.imread(str(image_path))

flipped, bright_adjusted = augment_image(img) # เรียกใช้ Method augment_image

return img, flipped, bright_adjusted

return None, None, None

original, flipped, bright_adjusted = demonstrate_augmentation('/content/drive/MyDrive/helmet-
detection/train')

fig, axes = plt.subplots(1, 3, figsize=(15, 5)) # Set ขนาดรูปภาพให้มีความยาว 15 นิ้วและความสูง 5 นิ้ว

axes[0].imshow(cv2.cvtColor(original, cv2.COLOR_BGR2RGB)) # ทำการ Convert สีรูปภาพจากเดิม BRG เป็น RGB

axes[0].set_title('Original')

axes[0].axis('off')

axes[1].imshow(cv2.cvtColor(flipped, cv2.COLOR_BGR2RGB))

axes[1].set_title('Flipped')

axes[1].axis('off')

axes[2].imshow(cv2.cvtColor(bright_adjusted, cv2.COLOR_BGR2RGB)) # ปรับความสว่างของรูปภาพ

axes[2].set_title('Brightness Adjusted')

axes[2].axis('off')

plt.show()

```



```
# เตรียมข้อมูล สำหรับใช้ในการ Train Models
```

```
%cd /content/yolov5
```

```
# Create a YAML file for your dataset ตาม Format Document YOLO5
```

```
data_yaml_content =
```

```
"""
```

```
train: /content/drive/MyDrive/helmet-detection/train/images
```

```
val: /content/drive/MyDrive/helmet-detection/val/images
```

```
nc: 4
```

```
names: ['with helmet', 'without helmet', 'rider', 'number plate']
```

```
"""
```

train: /content/drive/MyDrive/helmet-detection/train/images	Path ไปสู่ Folder ที่เก็บภาพสำหรับการเทรนข้อมูล
val: /content/drive/MyDrive/helmet-detection/val/images	Path ไปสู่ Folder ที่เก็บภาพสำหรับการตรวจสอบ (validation) โมเดล
nc: 4	Classification 4 ประเภท
names: ['with helmet', 'without helmet', 'rider', 'number plate']	คลาสที่ต้องจำแนกมี 4 คลาส ได้แก่ 'with helmet', 'without helmet', 'rider' และ 'number plate'

```
# เขียนคำสั่ง Configuration สำหรับเทรนข้อมูลในไฟล์ data.yaml
```

```
with open('data.yaml', 'w') as file:
```

```
    file.write(data_yaml_content)
```

```
!python train.py --img 640 --batch 16 --epochs 50 \
    --data data.yaml --cfg yolov5s.yaml \
    --weights yolov5s.pt --name yolov5_rider_safety
```

--img 640	ขนาดของรูปภาพ 640 x 640 pixel
--batch 16	ข้อมูลจำนวน 16 ข้อมูลต่อการ Train ใน 1 ครั้ง
--epochs 50	จำนวนรอบที่ใช้ในการเทรนโมเดล
-data data.yaml	ไฟล์ Payload Configuration สำหรับใช้ในการเทรนโมเดล
--cfg yolov5s.yaml	ไฟล์ Configuration ของ Model YOLO5
--weights yolov5s.pt	น้ำหนักของแต่ละข้อมูล เพื่อเพิ่มความแม่นยำของแต่ละข้อมูล
--name yolov5_rider_safety	ชื่อของการ Train ครั้งนี้มีชื่อว่า yolov5_rider_safety

Command for testing the model on new data

```
!python detect.py --weights /content/yolov5/runs/train/yolov5_rider_safety/weights/best.pt \
    --img 640 --conf 0.4 --source /content/drive/MyDrive/helmet-detection/val/images
```

--weights /content/yolov5/runs/train/yolov5_rider_safety/weights/best.pt	น้ำหนักความแม่นยำที่ดีที่สุด สำหรับการเทรนข้อมูล
--img 640	ขนาดของรูปภาพ
--conf 0.4	ค่าความเชื่อมั่นขั้นต่ำที่ 40 %
--source /content/drive/MyDrive/helmet-detection/val/images	แหล่งข้อมูลสำหรับการตรวจสอบ

```

detect: weights=['/content/yolov5/runs/train/yolov5_rider_safety/weights/best.pt'], source=/content/drive/MyDrive/helmet-detection/val/images, data=data/coco128.yaml
YOLOv5 v7.0-296-gae4ef3b2 Python-3.10.12 torch-2.2.1+cu121 CPU

Fusing layers...
YOLOv5s summary: 157 layers, 7020913 parameters, 0 gradients, 15.8 GFLOPs
image 1/20 /content/drive/MyDrive/helmet-detection/val/images/new0.jpg: 640x576 2 without helmets, 1 rider, 1 number plate, 482.4ms
image 2/20 /content/drive/MyDrive/helmet-detection/val/images/new113.jpg: 640x576 2 with helmets, 1 without helmet, 1 rider, 1 number plate, 392.1ms
image 3/20 /content/drive/MyDrive/helmet-detection/val/images/new115.jpg: 576x640 1 with helmet, 1 rider, 1 number plate, 376.1ms
image 4/20 /content/drive/MyDrive/helmet-detection/val/images/new127.jpg: 640x640 1 with helmet, 3 without helmets, 2 riders, 2 number plates, 429.3ms
image 5/20 /content/drive/MyDrive/helmet-detection/val/images/new137.jpg: 640x352 1 without helmet, 1 rider, 1 number plate, 230.8ms
image 6/20 /content/drive/MyDrive/helmet-detection/val/images/new26.jpg: 384x640 2 without helmets, 1 rider, 1 number plate, 245.6ms
image 7/20 /content/drive/MyDrive/helmet-detection/val/images/new28.jpg: 640x480 1 with helmet, 1 rider, 1 number plate, 355.8ms
image 8/20 /content/drive/MyDrive/helmet-detection/val/images/new4.jpg: 640x480 1 with helmet, 1 rider, 1 number plate, 314.2ms
image 9/20 /content/drive/MyDrive/helmet-detection/val/images/new41.jpg: 480x640 1 with helmet, 4 without helmets, 2 riders, 1 number plate, 336.1ms
image 10/20 /content/drive/MyDrive/helmet-detection/val/images/new47.jpg: 640x480 1 with helmet, 1 rider, 1 number plate, 333.1ms
image 11/20 /content/drive/MyDrive/helmet-detection/val/images/new49.jpg: 640x256 2 with helmets, 2 without helmets, 1 rider, 1 number plate, 181.0ms
image 12/20 /content/drive/MyDrive/helmet-detection/val/images/new56.jpg: 640x448 1 without helmet, 1 rider, 1 number plate, 306.9ms
image 13/20 /content/drive/MyDrive/helmet-detection/val/images/new73.jpg: 384x640 2 without helmets, 1 rider, 1 number plate, 254.8ms
image 14/20 /content/drive/MyDrive/helmet-detection/val/images/new74.jpg: 640x320 1 with helmet, 2 without helmets, 1 rider, 1 number plate, 229.0ms
image 15/20 /content/drive/MyDrive/helmet-detection/val/images/new80.jpg: 640x544 1 with helmet, 1 rider, 1 number plate, 347.8ms
image 16/20 /content/drive/MyDrive/helmet-detection/val/images/new86.jpg: 640x384 1 with helmet, 1 rider, 1 number plate, 259.7ms
image 17/20 /content/drive/MyDrive/helmet-detection/val/images/new89.jpg: 448x640 1 with helmet, 1 rider, 1 number plate, 305.1ms
image 18/20 /content/drive/MyDrive/helmet-detection/val/images/new9.jpg: 640x544 1 with helmet, 1 rider, 1 number plate, 382.5ms
image 19/20 /content/drive/MyDrive/helmet-detection/val/images/new94.jpg: 608x640 1 with helmet, 1 rider, 1 number plate, 414.0ms
image 20/20 /content/drive/MyDrive/helmet-detection/val/images/new98.jpg: 640x480 1 with helmet, 1 rider, 1 number plate, 306.4ms
Speed: 1.2ms pre-process, 324.1ms inference, 0.9ms NMS per image at shape (1, 3, 640, 640)
Results saved to runs/detect/exp2

```

เก็บข้อมูลหลังจากทำการ detect.py

```
output_directory = '/content/drive/MyDrive/helmet-detection/test/output'
```

```
output_files = os.listdir(output_directory)
```

```
if output_files:
```

```
    first_image_path = os.path.join(output_directory, output_files[0]) # ต่อ Path Images
```

```
    img = mpimg.imread(first_image_path)
```

```
    plt.imshow(img)
```

```
    plt.axis('off')
```

```
    plt.show()
```

```
else:
```

```
    print("No output files found.")
```



```
from PIL import Image
```

```
model = torch.hub.load('ultralytics/yolov5', 'custom',  
path='/content/yolov5/runs/train/yolov5_rider_safety/weights/best.pt') # Models หลังจากฝึก Train ข้อมูลเสร็จ
```

```
def detect_helmets(image_path):
```

```
    image = cv2.imread(image_path)
```

```
    results = model(image) # ทำนาย Predictions โดยใช้ Models YOLO5 หลังจากทำการฝึกเสร็จ
```

```
    output_image = cv2.cvtColor(np.squeeze(results.render()), cv2.COLOR_RGB2BGR)
```

```
# แปลงประเภทรูปภาพเป็นประเภท Numpy ใช้ส่งค่าออกไปใช้ใน plt.imread()
```

```
    return output_image
```

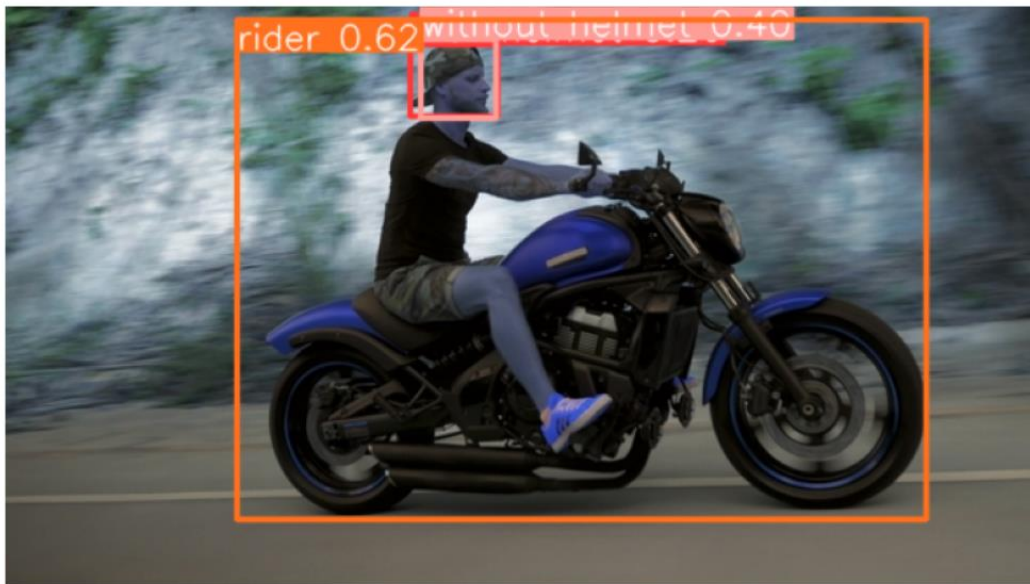
```
output_image = detect_helmets('/content/drive/MyDrive/helmet-detection/deploy/rider_with_hat.jpg')
```

```
plt.figure(figsize=(10, 10))
```

```
plt.imshow(cv2.cvtColor(output_image, cv2.COLOR_BGR2RGB))
```

```
plt.axis('off')
```

```
plt.show()
```



เพิ่มรูปอีกหลายกรณี หากรูปคนขี่มอเตอร์ไซด์กะป๋ายทะเบียนไม่เจอ

แหล่งที่มา

<https://docs.ultralytics.com>

<https://www.kaggle.com/code/pratul007/rider-helmet-no-helmet-yolo/notebook>

<https://www.kaggle.com/datasets/aneesarom/rider-with-helmet-without-helmet-number-plate/data>