

Manual for Cloud type Detection by Deep learning

1. เริ่มจากโหลดไฟล์ที่ [GitHub link here!](#)
2. เตรียมแพลตฟอร์ม (Software & Hardware)
ติดตั้งไลบรารีที่จำเป็นได้แก่
 - ☐ conda install -c conda-forge cudnn=8.1
 - ☐ pip install tensorflow-gpu==2.8.*
 - ☐ pip install protobuf==3.20.*
 - ☐ pip install pillow
 - ☐ pip install lxml
 - ☐ pip install Cython
 - ☐ pip install opencv-python
 - ☐ pip install opencv-contrib-python
 - ☐ pip install Flask
 - ☐ pip install Flask-Navigation
 - ☐ pip install shapely

```
import tensorflow as tf
print(tf.__version__)
print(tf.config.list_physical_devices('GPU'))
```

✓ 0.0s

2.8.1

[PhysicalDevice(name='/physical_device:GPU:0', device_type='GPU')]

```
import cv2
import shapely
import flask
import importlib.metadata

print('opencv version',cv2.__version__)
print('shapely version',shapely.__version__)
print('flask version',importlib.metadata.version('flask'))
```

✓ 0.4s

opencv version 4.10.0
shapely version 2.0.6
flask version 3.1.0

3. ไปที่ไฟล์ setting.ipynb ที่อยู่ในหน้าแรก แล้วแก้ไข path ไฟล์ทั้ง 2 ที่ดังตาราง

```
import re
with open(r"your path +
Cloud-type-Detection-by-Deep-learning-main/models-master/research/object_detection/
packages/tf2/setup.py") as f:
```

```
s = f.read()
```

```
with open(r"your path +  
Cloud-type-Detection-by-Deep-learning-main/models-master/research/setup.py", 'w')  
as f:  
    s = re.sub('tf-models-official>=2.5.1',  
               'tf-models-official==2.8.*', s)  
    f.write(s)
```

```
import re  
with open(r"D:\TNI-works\CPE-431\Cloud-type-Detection-by-Deep-learning-main\Cloud-type-Detection-by-Deep-learning-main\models-master\research\object_detection\packages\tf2setup.py") as f:  
    s = f.read()  
  
with open(r"D:\TNI-works\CPE-431\Cloud-type-Detection-by-Deep-learning-main\Cloud-type-Detection-by-Deep-learning-main\models-master\research\setup.py", 'w') as f:  
    s = re.sub('tf-models-official>=2.5.1',  
               'tf-models-official==2.8.*', s)  
    f.write(s)
```

จากนั้นเปิด Terminal (ให้ตรวจสอบว่าอยู่ใน env ที่ต้องการด้วยหรือไม่) และให้ติดตั้งด้วยคำสั่ง
ดังนี้ (ในโฟลเดอร์ **models-master/research** จะต้องพบไฟล์ **setup.py**)

บน terminal: cd models/research

pip install -e .

pip install protobuf==3.20

protoc object_detection/protos/*.proto --python_out=.

```
(smartcctv_0039) G:\TNIWorks\CPE-431\Cloud-type  
in\models-master\research>pip install -e .
```

หลังจากขั้นตอนนี้ ผู้ใช้จะสามารถใช้โมเดลที่ผมเทรนไว้ก่อนแล้ว โดยสามารถกดไปที่ไฟล์
test.ipynb ในหน้าแรก แล้วทำการเปลี่ยน **path** ใน Cell สุดท้าย และสามารถเปลี่ยนรูปทดลองได้
ใน **path** สุดท้าย (# Path to the test image)

******ใน Cell สุดท้ายของไฟล์ test.ipynb จะเป็นโค้ดที่มีต้นแบบมาจากไฟล์ obj_detection.py แล้ว
ผมนำมาดัดแปลงเพิ่มเติมนิดหน่อย

```
# Load the saved model
```

```
SAVED_MODEL_PATH = "your path +
```

```
Cloud-type-Detection-by-Deep-learning-main/exported_model_v13/saved_model"
```

```
# Load the label map
```

```
LABEL_MAP_PATH = "your path +
```

```
Cloud-type-Detection-by-Deep-learning-main/models-master/ssd_mobilenet_v2/train/l  
abel_map.pbtxt"
```

```
# Path to the test image
```

```
IMAGE_PATH = r"your path +
```

```
Cloud-type-Detection-by-Deep-learning-main/Cloud_image_classification_Dataset/dat  
a/test/132.jpg'
```

```
# Load the saved model
SAVED_MODEL_PATH = "G:/TNIWorks/CPE-431/Cloud-type-Detection-by-Deep-learning-main/exported_model_v13/saved_model"

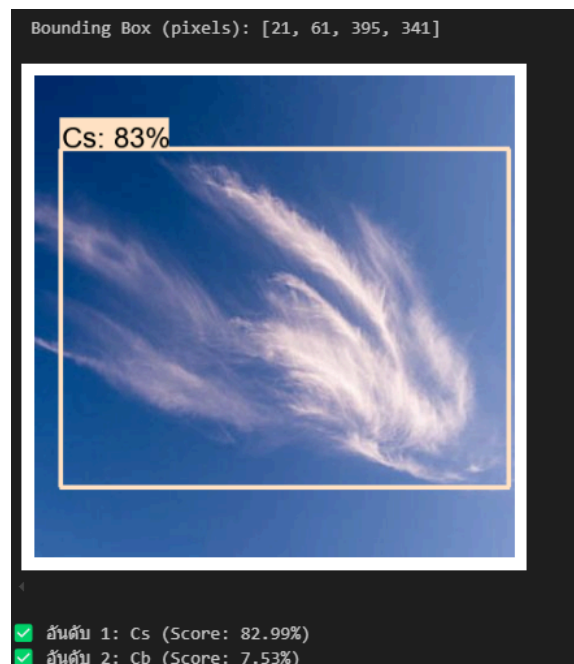
detect_fn = tf.saved_model.load(SAVED_MODEL_PATH)

# Load the label map
LABEL_MAP_PATH = "G:/TNIWorks/CPE-431/Cloud-type-Detection-by-Deep-learning-main/models-master/ssd_mobilenet_v2/train/label_map.pbtxt"
category_index = label_map_util.create_category_index_from_labelmap(LABEL_MAP_PATH, use_display_name=True)

# Function to load and preprocess an image
def load_image_into_numpy_array(image_path):
    image = cv2.imread(image_path)
    image_rgb = cv2.cvtColor(image, cv2.COLOR_BGR2RGB) # Convert BGR to RGB
    return image_rgb

# Path to the test image
IMAGE_PATH = r'G:/TNIWorks/CPE-431/Cloud-type-Detection-by-Deep-learning-main/Cloud_image_classification_Dataset/data/test/132.jpg'
image_np = load_image_into_numpy_array(IMAGE_PATH)
```

ตัวอย่างผลลัพธ์






- หลังจากนี้เป็นขั้นตอนที่ผู้ใช้สามารถทำการเทรนโมเดลเอง โดยเริ่มไปที่ไฟล์ `generate_tfrecord.py` ซึ่งอยู่ใน `models-master/research/generate_tfrecord.py` แล้วทำการเปลี่ยน **path** ไฟล์ 6 ที่ตรงบรรทัดที่ 84 และ 85 เมื่อแก้ไขเสร็จแล้วทำการกด Run เพื่อสร้างไฟล์ Record

```
create_tf_record('your path +  
Cloud-type-Detection-by-Deep-learning-main/models-master/research/data3/tr  
ain/annotations', 'your path +  
Cloud-type-Detection-by-Deep-learning-main/models-master/research/data3/tr  
ain/images', 'your path +  
Cloud-type-Detection-by-Deep-learning-main/models-master/ssd_mobilenet_v  
2/train/train.record', label_map)
```

```
create_tf_record('your path +  
Cloud-type-Detection-by-Deep-learning-main/models-master/research/data3/va  
l/annotations', 'your path +  
Cloud-type-Detection-by-Deep-learning-main/models-master/research/data3/va  
l/images', 'your path +  
Cloud-type-Detection-by-Deep-learning-main/models-master/ssd_mobilenet_v  
2/train/val.record', label_map)
```

```
69 # Example usage  
70 label_map = {  
71     'Ac': 1,  
72     'As': 2,  
73     'Cb': 3,  
74     'Cc': 4,  
75     'Ci': 5,  
76     'Cs': 6,  
77     'Ct': 7,  
78     'Cu': 8,  
79     'Ns': 9,  
80     'Sc': 10,  
81     'St': 11,  
82 }  
83  
84 create_tf_record('D:/TNI-works/CPE-431/lab6/models-master/research/data3/train/annotations', 'D:/TNI-works/CPE-431/lab6/models-master/res  
85 create_tf_record('D:/TNI-works/CPE-431/lab6/models-master/research/data3/val/annotations', 'D:/TNI-works/CPE-431/lab6/models-master/resea  
86
```

เมื่อรันเสร็จในโฟลเดอร์ \models-master\ssd_mobilenet_v2\train จะมี 3 ไฟล์ดังรูป

 label_map.pbtxt	2/11/2025 9:33 AM	PBTEXT File	1 KB
 train.record	2/23/2025 4:07 AM	RECORD File	501,722 KB
 val.record	2/23/2025 4:07 AM	RECORD File	30,026 KB

5. ไปที่ไฟล์ pipeline.config ซึ่งอยู่ใน models-master\ssd_mobilenet_v2\pipeline.config แล้วทำการเปลี่ยน path ไฟล์ 5 ที่ ตรงบรรทัดที่ 142, 200, 202, 212 และ 216 โดยในไฟล์นี้เราสามารถเปลี่ยนการตั้งค่าได้หลายอย่าง เช่น

- batch_size ตรงบรรทัดที่ 144
- num_step ตรงบรรทัดที่ 148, 186 และ 188
- learning_rate ตรงบรรทัดที่ 185 และ 187
- data_augmentation_options ตั้งแต่บรรทัดที่ 149 ลงไป

เมื่อแก้ไขเสร็จแล้วอย่าลืมทำการกดเซฟ

```
fine_tune_checkpoint: "your path +  
Cloud-type-Detection-by-Deep-learning-main/models-master/ssd_mobilenet_v2/check  
point/ckpt-0"
```

```
train_input_reader: {  
  label_map_path: "your path +  
Cloud-type-Detection-by-Deep-learning-main/models-master/ssd_mobilenet_v2/train/l
```

```

abel_map.pbtxt"
  tf_record_input_reader {
    input_path: "your path +
Cloud-type-Detection-by-Deep-learning-main/models-master/ssd_mobilenet_v2/train/t
rain.record"
  }
}

```

```

eval_input_reader: {
  label_map_path: "your path +
Cloud-type-Detection-by-Deep-learning-main/ssd_mobilenet_v2/train/label_map.pbtxt
"
  shuffle: false
  num_epochs: 1
  tf_record_input_reader {
    input_path: "your path +
Cloud-type-Detection-by-Deep-learning-main/ssd_mobilenet_v2/train/val.record"
  }
}

```

```

140 train_config: {
141   fine_tune_checkpoint_version: V2
142   fine_tune_checkpoint: "G:/TNIWorks/CPE-431/Cloud-type-Detection-by-Deep-learning-main/models-master/ssd_mobilenet_v2/checkpoint/ckpt-0"
143   fine_tune_checkpoint_type: "detection"
144   batch_size: 8
145   sync_replicas: true
146   startup_delay_steps: 0
147   replicas_to_aggregate: 8
148   num_steps: 5000
149   data_augmentation_options {
150     random_horizontal_flip {
151     }
152   }
153   data_augmentation_options {
154     random_adjust_brightness {
155       max_delta: 0.2
156     }
157   }

```

```

198
199 train_input_reader: {
200   label_map_path: "G:/TNIWorks/CPE-431/Cloud-type-Detection-by-Deep-learning-main/models-master/ssd_mobilenet_v2/train/label_map.pbtxt"
201   tf_record_input_reader {
202     input_path: "G:/TNIWorks/CPE-431/Cloud-type-Detection-by-Deep-learning-main/models-master/ssd_mobilenet_v2/train/train.record"
203   }
204 }
205
206 eval_config: {
207   metrics_set: "coco_detection_metrics"
208   use_moving_averages: false
209 }
210
211 eval_input_reader: {
212   label_map_path: "G:/TNIWorks/CPE-431/Cloud-type-Detection-by-Deep-learning-main/ssd_mobilenet_v2/train/label_map.pbtxt"
213   shuffle: false
214   num_epochs: 1
215   tf_record_input_reader {
216     input_path: "G:/TNIWorks/CPE-431/Cloud-type-Detection-by-Deep-learning-main/ssd_mobilenet_v2/train/val.record"
217   }
218 }

```

6. เมื่อตั้งค่า pipeline.config เสร็จแล้ว ให้เปิด terminal ใน directory ของ Cloud-type-Detection-by-Deep-learning-main แล้วพิมพ์คำสั่ง

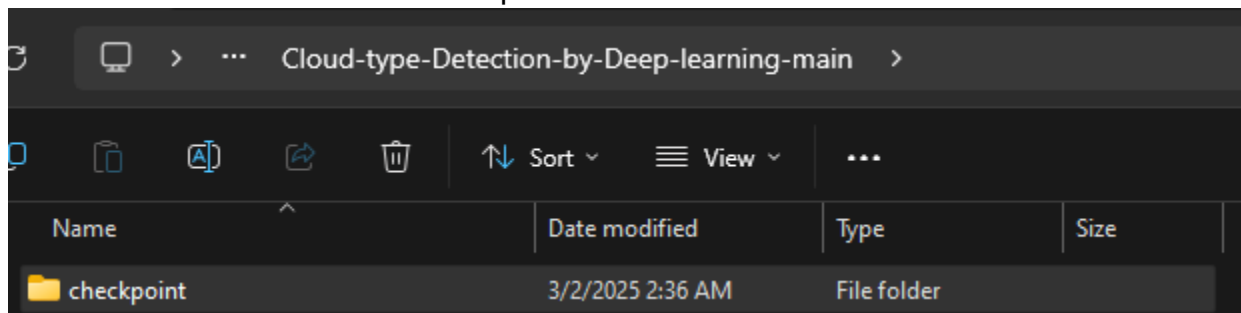
```
python models-master/research/object_detection/model_main_tf2.py
--pipeline_config_path="models-master/ssd_mobilenet_v2/pipeline.config"
--model_dir="checkpoint" --alsologtostderr
--checkpoint_exclude_scopes="FeatureExtractor"
```

ส่วนที่ผู้ใช้จะต้องเปลี่ยน (หากพิมพ์คำสั่งอยู่ใน directory อื่น)

- python **path** + models-master/research/object_detection/model_main_tf2.py
- --pipeline_config_path="**path** + models-master/ssd_mobilenet_v2/pipeline.config"
- --model_dir="**path ของ checkpoint ที่ผู้ใช้อยากตั้ง**" --alsologtostderr
- โดยในตัวอย่าง model_dir="checkpoint" จะอยู่ใน directory ของ Cloud-type-Detection-by-Deep-learning-main เพราะพิมพ์คำสั่งใน directory นี้
- ผู้ใช้จะต้องมีการเปลี่ยนชื่อ checkpoint เช่น checkpoint_v2 ถ้าเป็นการเทรนรอบที่ 2 หรือมากกว่านั้นเพื่อเปรียบเทียบผลลัพธ์

```
(smartcctv_0039) G:\TNIWorks\CPE-431\Cloud-type-Detection-by-Deep-learning-main\Cloud-type-Detection-by-Deep-learning-main>python models-master/research/object_detection/model_main_tf2.py --pipeline_config_path="models-master/ssd_mobilenet_v2/pipeline.config" --model_dir="checkpoint" --alsologtostderr --checkpoint_exclude_scopes="FeatureExtractor"
```

เมื่อเทรนเสร็จแล้วจะมีโฟลเดอร์ checkpoint ออกมา



จากนั้นแล้วให้พิมพ์คำสั่งต่อไปใน terminal เดิม

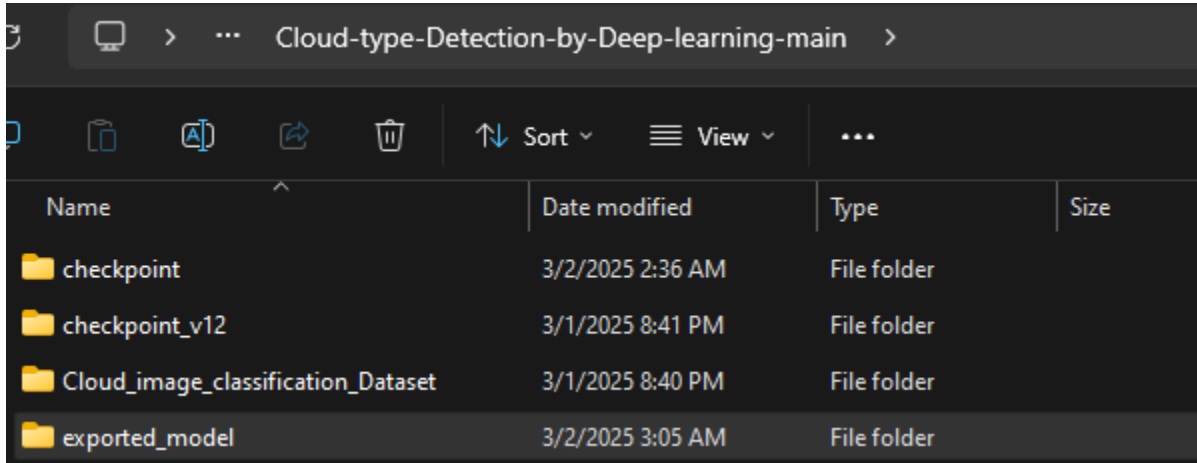
```
python models-master/research/object_detection/exporter_main_v2.py --input_type
image_tensor
--pipeline_config_path="models-master/ssd_mobilenet_v2/pipeline.config"
--trained_checkpoint_dir="checkpoint" --output_directory="exported_model"
```

ส่วนที่ผู้ใช้จะต้องเปลี่ยน (หากพิมพ์คำสั่งอยู่ใน directory อื่น)

- python **path** + models-master/research/object_detection/exporter_main_v2.py
- --pipeline_config_path="**path** + models-master/ssd_mobilenet_v2/pipeline.config"
- --trained_checkpoint_dir="**path ของ checkpoint ที่ผู้ใช้เทรนออกมาจากคำสั่งก่อนหน้า**"
- --output_directory="**path ของ exported_model ที่ผู้ใช้อยากตั้ง**"
- ผู้ใช้จะต้องมีการเปลี่ยนชื่อ checkpoint เช่น checkpoint_v2 ถ้าเป็นการเทรนรอบที่ 2 หรือมากกว่านั้นเพื่อเปรียบเทียบผลลัพธ์
- ผู้ใช้จะต้องมีการเปลี่ยนชื่อ exported_model เช่น exported_model_v2 ถ้าเป็นการเทรนรอบที่ 2 หรือมากกว่านั้นเพื่อเปรียบเทียบผลลัพธ์และให้สอดคล้องกับ checkpoint

```
(smartcctv_0039) G:\TNIWorks\CPE-431\Cloud-type-Detection-by-Deep-learning-main\Cloud-type-Detection-by-Deep-learning-main>python models-master/research/object_detection/exporter_main_v2.py --input_type image_tensor --pipeline_config_path="models-master/ssd_mobilenet_v2/pipeline.config" --trained_checkpoint_dir="checkpoint" --output_directory="exported_model"
```

เสร็จแล้วจะมีโฟลเดอร์ exported_model โผล่มา



เมื่อผู้ใช้ได้ exported_model มาแล้ว ผู้ใช้ควรจะต้องไปทดสอบประสิทธิภาพของโมเดลที่ไฟล์ evaluate.py โดยในไฟล์นี้ จะมี path ไฟล์ให้ผู้ใช้เปลี่ยนอยู่ 3 จุด

```
# Path configurations
label_map_path = "path +
Cloud-type-Detection-by-Deep-learning-main/models-master/ssd_mobilenet_v2/train/label_map.pbtxt"
val_record_path = "path +
Cloud-type-Detection-by-Deep-learning-main/models-master/ssd_mobilenet_v2/train/val.record"
model_dir = "path +
Cloud-type-Detection-by-Deep-learning-main/exported_model/saved_model"
```

- exported_model/saved_model เป็น path ที่ได้จากการทำในข้อที่ 4 ผู้ใช้อาจมีการเปลี่ยนชื่อ exported_model ถ้าทำข้อ 4 หลายรอบ

```
6
7 # Path configurations
8 label_map_path = "G:/TNIWorks/CPE-431/Cloud-type-Detection-by-Deep-learning-main/models-master/ssd_mobilenet_v2/train/label_map.pbtxt"
9 val_record_path = "G:/TNIWorks/CPE-431/Cloud-type-Detection-by-Deep-learning-main/models-master/ssd_mobilenet_v2/train/val.record"
10 model_dir = "G:/TNIWorks/CPE-431/Cloud-type-Detection-by-Deep-learning-main/exported_model/saved_model"
11
```

ตัวอย่างผลลัพธ์

```
Accuracy: 97.51%
Precision: 97.63%
Recall: 97.51%
F1 Score: 97.55%
mAP: 87.48%
```

- หากผู้ใช้ยังไม่พอใจในประสิทธิภาพของโมเดล ผู้ใช้สามารถย้อนกลับไปทำตั้งแต่ข้อที่ 4 ได้ โดยการเปลี่ยนแปลงการตั้งค่าใน pipeline.config

7. เมื่อผู้ใช้พอใจกับประสิทธิภาพของโมเดลแล้ว ผู้ใช้จะสามารถใช้โมเดลที่เทรนไว้ โดยกดไปที่ไฟล์ test.ipynb ในหน้าแรก โดยโค้ดจะอยู่ใน Cell สุดท้ายของไฟล์ แล้วทำการเปลี่ยน **path** ใน Cell และสามารถเปลี่ยนรูปทดลองได้ใน **path** สุดท้าย(# Path to the test image)

******ใน Cell สุดท้ายของไฟล์ test.ipynb จะเป็นโค้ดที่มีต้นแบบมาจากไฟล์

obj_detection.py แล้วผมนำมาดัดแปลงเพิ่มเติมเล็กน้อย

```
# Load the saved model
```

```
SAVED_MODEL_PATH = "your path +
```

```
Cloud-type-Detection-by-Deep-learning-main/exported_model/saved_model"
```

```
# Load the label map
```

```
LABEL_MAP_PATH = "your path +
```

```
Cloud-type-Detection-by-Deep-learning-main/models-master/ssd_mobilenet_v2/train/label_map.pbtxt"
```

```
# Path to the test image
```

```
IMAGE_PATH = r'your path +
```

```
Cloud-type-Detection-by-Deep-learning-main/Cloud_image_classification_Dataset/data/test/132.jpg'
```

```
# Load the saved model
SAVED_MODEL_PATH = "G:/TNIWorks/CPE-431/Cloud-type-Detection-by-Deep-learning-main/exported_model/saved_model"

detect_fn = tf.saved_model.load(SAVED_MODEL_PATH)

# Load the label map
LABEL_MAP_PATH = "G:/TNIWorks/CPE-431/Cloud-type-Detection-by-Deep-learning-main/models-master/ssd_mobilenet_v2/train/label_map.pbtxt"
category_index = label_map_util.create_category_index_from_labelmap(LABEL_MAP_PATH, use_display_name=True)

# Function to load and preprocess an image
def load_image_into_numpy_array(image_path):
    image = cv2.imread(image_path)
    image_rgb = cv2.cvtColor(image, cv2.COLOR_BGR2RGB) # Convert BGR to RGB
    return image_rgb

# Path to the test image
IMAGE_PATH = r'G:/TNIWorks/CPE-431/Cloud-type-Detection-by-Deep-learning-main/Cloud_image_classification_Dataset/data/test/132.jpg'
image_np = load_image_into_numpy_array(IMAGE_PATH)
```

ตัวอย่างผลลัพธ์

