Aravinth R - 19MIC0053 0. Setup Paths In [1]: WORKSPACE PATH = 'Tensorflow/workspace' SCRIPTS PATH = 'Tensorflow/scripts' APIMODEL PATH = 'Tensorflow/models'

IMAGE PATH = WORKSPACE PATH+'/images' MODEL PATH = WORKSPACE PATH+'/models'

In [3]:

Out[3]:

ANNOTATION PATH = WORKSPACE PATH+'/annotations' PRETRAINED MODEL PATH = WORKSPACE PATH+'/pre-trained-models' CONFIG PATH = MODEL_PATH+'/my_ssd_mobnet/pipeline.config' CHECKPOINT PATH = MODEL PATH+'/my ssd mobnet/' 1. Create Label Map

In [2]: labels = [{'name':'ID', 'id':1}, {'name':'No ID', 'id':2}] [{'name': 'ID', 'id': 1}, {'name': 'No ID', 'id': 2}] for label in labels: f.write('item { \n') f.write('\tname:\'{}\'\n'.format(label['name'])) f.write('\tid:{}\n'.format(label['id'])) f.write('}\n') 2. Create TF records

CSI1006 - MINIPROJECT

Title - ID Card Detection Using Computer Vision

Team Members

Priyanshi Premkumar - 19MIC0051

Ravuri Yashaswini Tejaswi - 19MIC0052

In [4]: with open(ANNOTATION PATH + '/label map.pbtxt', 'w') as f: sorflow io plugins.so'] vs/tensorflow/lib/python3.9/site-packages/tensorflow_io/python/ops/libtensorflow io plugins.so'"]

In [5]: | !python {SCRIPTS PATH + '/generate tfrecord.py'} -x {IMAGE PATH + '/train'} -1 {ANNOTATION PATH + '/label map.p. !python {SCRIPTS PATH + '/generate tfrecord.py'} -x{IMAGE PATH + '/test'} -1 {ANNOTATION PATH + '/label map.pbt /Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow_io/python/ops/__init__.py:98: UserWarning: unable to load libtensorflow_io_plugins.so: unable to open file: libtensorflow_io_plugins.so, from warnings.warn(f"unable to load libtensorflow_io_plugins.so: {e}")

paths: ['/Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow io/python/ops/libten caused by: ["[Errno 2] The file to load file system plugin from does not exist.: '/Users/aravinth/miniconda3/en /Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow io/python/ops/ init .py:10 4: UserWarning: file system plugins are not loaded: unable to open file: libtensorflow_io.so, from paths: ['/Us $ers/aravinth/miniconda 3/envs/tensor flow/lib/python 3.9/site-packages/tensor flow_io/python/ops/libtensor flow_io.set and the state of the state$ caused by: ["dlopen(/Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow io/pytho n/ops/libtensorflow io.so, 0x0006): tried: '/Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packa ges/tensorflow_io/python/ops/libtensorflow_io.so' (no such file), '/System/Volumes/Preboot/Cryptexes/OS/Users/a ravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow io/python/ops/libtensorflow io.so' (n o such file), '/Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow io/python/ops/ libtensorflow io.so' (no such file)"] warnings.warn(f"file system plugins are not loaded: {e}") Successfully created the TFRecord file: Tensorflow/workspace/annotations/train.record /Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow io/python/ops/ init .py:98: UserWarning: unable to load libtensorflow_io_plugins.so: unable to open file: libtensorflow_io_plugins.so, from paths: ['/Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow io/python/ops/libten sorflow io plugins.so'] caused by: ["[Errno 2] The file to load file system plugin from does not exist.: '/Users/aravinth/miniconda3/en vs/tensorflow/lib/python3.9/site-packages/tensorflow io/python/ops/libtensorflow io plugins.so'"] warnings.warn(f"unable to load libtensorflow io plugins.so: {e}") /Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow io/python/ops/ init .py:10 4: UserWarning: file system plugins are not loaded: unable to open file: libtensorflow io.so, from paths: ['/Us ers/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow_io/python/ops/libtensorflow_io.s

caused by: ["dlopen(/Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow io/pytho n/ops/libtensorflow io.so, 0x0006): tried: '/Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packa ges/tensorflow_io/python/ops/libtensorflow_io.so' (no such file), '/System/Volumes/Preboot/Cryptexes/OS/Users/a ravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow io/python/ops/libtensorflow io.so' (n o such file), '/Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow io/python/ops/ libtensorflow_io.so' (no such file)"] warnings.warn(f"file system plugins are not loaded: {e}") Successfully created the TFRecord file: Tensorflow/workspace/annotations/test.record

3. Download TF Models Pretrained Models from Tensorflow Model Zoo

!cd Tensorflow && git clone https://github.com/tensorflow/models fatal: destination path 'models' already exists and is not an empty directory. 4. Copy Model Config to Training Folder In [7]: CUSTOM MODEL NAME = 'my ssd mobnet'

!mkdir {'Tensorflow/workspace/models/'+CUSTOM MODEL NAME+'/'} !cp {PRETRAINED_MODEL_PATH+'/ssd_mobilenet_v2_fpnlite_320x320_coco17_tpu-8/pipeline.config'} {MODEL PATH+'/'+CU mkdir: Tensorflow/workspace/models/my ssd mobnet/: File exists

/Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow_io/python/ops/__init__.py:98: UserWarning: unable to load libtensorflow_io_plugins.so: unable to open file: libtensorflow io plugins.so, from paths: ['/Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow_io/python/ops/libten

caused by: ["[Errno 2] The file to load file system plugin from does not exist.: '/Users/aravinth/miniconda3/en

/Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow io/python/ops/ init .py:10 4: UserWarning: file system plugins are not loaded: unable to open file: libtensorflow io.so, from paths: ['/Us ers/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow_io/python/ops/libtensorflow_io.s

caused by: ["dlopen(/Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow_io/pytho n/ops/libtensorflow io.so, 0x0006): tried: '/Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packa ges/tensorflow io/python/ops/libtensorflow io.so' (no such file), '/System/Volumes/Preboot/Cryptexes/OS/Users/a ravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow io/python/ops/libtensorflow io.so' (n o such file), '/Users/aravinth/miniconda3/envs/tensorflow/lib/python3.9/site-packages/tensorflow_io/python/ops/

vs/tensorflow/lib/python3.9/site-packages/tensorflow_io/python/ops/libtensorflow_io_plugins.so'"]

5. Update Config For Transfer Learning

warnings.warn(f"unable to load libtensorflow_io_plugins.so: {e}")

warnings.warn(f"file system plugins are not loaded: {e}")

In [10]: CONFIG PATH = MODEL PATH+'/'+CUSTOM MODEL NAME+'/pipeline.config'

In [11]: config = config util.get_configs_from_pipeline_file(CONFIG_PATH)

from object detection.utils import config util from object detection.protos import pipeline pb2

from google.protobuf import text format

libtensorflow io.so' (no such file)"]

In [9]: import tensorflow as tf

In [12]: config

Out[12]:

{'model': ssd {

}

num classes: 90 image resizer {

> height: 320 width: 320

feature extractor {

min depth: 16 conv hyperparams { regularizer {

initializer {

batch norm {

scale: true

use depthwise: true

faster_rcnn box coder {

matched threshold: 0.5 unmatched threshold: 0.5 ignore thresholds: false

use_matmul_gather: true

similarity calculator { iou similarity {

> conv_hyperparams { regularizer {

> > initializer {

batch norm {

depth: 128

anchor generator {

post processing {

loss {

min level: 3 max level: 7 anchor scale: 4.0 aspect_ratios: 1.0 aspect_ratios: 2.0 aspect_ratios: 0.5 scales per octave: 2

kernel size: 3

use depthwise: true

scale: true

mean: 0.0

activation: RELU 6

12 regularizer {

box predictor {

negatives lower than unmatched: true

weight shared convolutional box predictor {

weight: 3.9999998989515007e-05

random normal initializer {

decay: 0.996999979019165

num layers before predictor: 4

share prediction tower: true

multiscale anchor generator {

batch non max suppression {

score converter: SIGMOID

localization loss { weighted smooth 11 {

classification loss {

gamma: 2.0 alpha: 0.25

weighted sigmoid focal {

classification weight: 1.0 localization weight: 1.0

encode background as zeros: true normalize_loc_loss_by_codesize: true

inplace batchnorm update: true

'train config': batch size: 128 data augmentation options { random horizontal flip {

freeze batchnorm: false

data augmentation options { random crop image {

overlap thresh: 0.0

min area: 0.75 max area: 1.0

sync replicas: true

momentum optimizer { learning_rate {

cosine_decay_learning_rate {

total steps: 50000

warmup steps: 1000

use moving average: false

unpad groundtruth tensors: false

fine_tune_checkpoint_version: V2,

startup_delay_steps: 0.0 replicas to aggregate: 8 max number of boxes: 100

tf record input reader {

use moving averages: false,

tf record input reader {

tf record input reader {

proto str = f.read()

f.write(config text)

6. Train the model

10000

In [17]: import os

In [14]: pipeline config.model.ssd.num classes = 2

num steps: 50000

shuffle: false num epochs: 1

shuffle: false num epochs: 1

],

learning rate base: 0.07999999821186066

momentum optimizer value: 0.8999999761581421

fine tune checkpoint: "PATH TO BE CONFIGURED"

fine_tune_checkpoint_type: "classification"

input path: "PATH TO BE CONFIGURED"

input path: "PATH TO BE CONFIGURED"

input_path: "PATH_TO_BE CONFIGURED"

In [13]: pipeline config = pipeline pb2.TrainEvalPipelineConfig() with tf.io.gfile.GFile(CONFIG PATH, "r") as f:

pipeline config.train config.batch size = 4

In [15]: config text = text format.MessageToString(pipeline config) with tf.io.gfile.GFile(CONFIG PATH, "wb") as f:

7. Load Train Model From Checkpoint

from object detection.utils import visualization utils as viz utils

configs = config util.get configs from pipeline file(CONFIG PATH)

ckpt.restore(os.path.join(CHECKPOINT PATH, 'ckpt-6')).expect partial()

detections = detection model.postprocess(prediction dict, shapes)

ckpt = tf.compat.v2.train.Checkpoint(model=detection model)

image, shapes = detection model.preprocess(image)

Device (device: 0, name: METAL, pci bus id: <undefined>)

{1: {'id': 1, 'name': 'ID'}, 2: {'id': 2, 'name': 'No_ID'}}

num detections = int(detections.pop('num detections')) detections = {key: value[0, :num_detections].numpy()

viz_utils.visualize_boxes_and_labels_on_image_array(image np with detections, detections['detection boxes'],

detections['detection scores'],

use normalized coordinates=True,

detections['num_detections'] = num_detections

image_np_with_detections = image_np.copy()

category index,

if cv2.waitKey(1) & 0xFF == ord('q'):

max boxes to draw=5, min score thresh=.5, agnostic mode=False)

for key, value in detections.items() }

detections['detection classes']+label id offset,

width = int(cap.get(cv2.CAP_PROP_FRAME_WIDTH)) height = int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT))

detections = detect_fn(input_tensor)

detection classes should be ints.

ret, frame = cap.read() image_np = np.array(frame)

label id offset = 1

cap.release()

break

prediction dict = detection model.predict(image, shapes)

detection model = model builder.build(model config=configs['model'], is training=False)

from object detection.utils import label map util

In [18]: # Load pipeline config and build a detection model

Restore checkpoint

def detect_fn(image):

return detections

Metal device set to: Apple M1

8. Detect in Real-Time

@tf.function

MA support.

import numpy as np

In [32]: cap = cv2.VideoCapture(1)

In [19]: import cv2

In [21]: category index

In [33]: while True:

Out[21]:

from object detection.builders import model builder

text_format.Merge(proto_str, pipeline_config)

warmup learning rate: 0.026666000485420227

'train input config': label map path: "PATH TO BE CONFIGURED"

'eval input configs': [label map path: "PATH TO BE CONFIGURED"

'eval input config': label map path: "PATH TO BE CONFIGURED"

pipeline config.train config.fine tune checkpoint type = "detection"

pipeline config.train input reader.label map path= ANNOTATION PATH + '/label map.pbtxt'

pipeline_config.eval_input_reader[0].label_map_path = ANNOTATION_PATH + '/label_map.pbtxt'

pipeline config.train config.fine tune checkpoint = PRETRAINED MODEL PATH+'/ssd mobilenet v2 fpnlite 320x320 co

pipeline config.train input reader.tf record input reader.input path[:] = [ANNOTATION PATH + '/train.record']

pipeline config.eval input reader[0].tf record input reader.input path[:] = [ANNOTATION PATH + '/test.record']

In [57]: print("""python {}/research/object detection/model main tf1.py --model dir={}/{} --pipeline config path={}/{}/p

python Tensorflow/models/research/object detection/model main tf1.py --model dir=Tensorflow/workspace/models/my ssd mobnet --pipeline config path=Tensorflow/workspace/models/my ssd mobnet/pipeline.config --num train steps=

2023-02-17 22:37:45.237521: I tensorflow/core/common runtime/pluggable device/pluggable device factory.cc:306] Could not identify NUMA node of platform GPU ID 0, defaulting to 0. Your kernel may not have been built with NU

2023-02-17 22:37:45.237542: I tensorflow/core/common runtime/pluggable device/pluggable device factory.cc:272] Created TensorFlow device (/job:localhost/replica:0/task:0/device:GPU:0 with 0 MB memory) -> physical Pluggable

In [20]: category_index = label_map_util.create_category_index_from_labelmap(ANNOTATION_PATH+'/label_map.pbtxt')

input_tensor = tf.convert_to_tensor(np.expand_dims(image_np, 0), dtype=tf.float32)

detections['detection_classes'] = detections['detection_classes'].astype(np.int64)

cv2.imshow('object detection', cv2.resize(image np with detections, (800, 600)))

'eval config': metrics set: "coco detection metrics"

optimizer {

min_object_covered: 0.0 min aspect ratio: 0.75 max aspect ratio: 3.0

normalize loss by num matches: true

score threshold: 9.9999993922529e-09 iou threshold: 0.6000000238418579 max_detections_per_class: 100 max total detections: 100 use static shapes: false

epsilon: 0.0010000000474974513

class prediction bias init: -4.599999904632568

stddev: 0.009999999776482582

force match for each row: true

y scale: 10.0 x scale: 10.0 height scale: 5.0 width scale: 5.0

argmax matcher {

min level: 3 max level: 7

fpn {

box coder {

matcher {

mean: 0.0

activation: RELU 6

fixed shape resizer {

depth multiplier: 1.0

12 regularizer {

type: "ssd mobilenet v2 fpn keras"

random normal initializer {

decay: 0.996999979019165

additional layer depth: 128

epsilon: 0.0010000000474974513

override base feature extractor hyperparams: true

weight: 3.9999998989515007e-05

stddev: 0.009999999776482582

sorflow_io_plugins.so']