

5.4. Milestone 4: Semantic Segmentation

```
1. importing the dataset

In [ ]:

!wget http://images.cocodataset.org/zips/train2014.zip
!unzip -q train2014.zip
!wget http://images.cocodataset.org/zips/val2014.zip
!wget http://images.cocodataset.org/annotations/annotations_trainval2014.zip
!unzip -q val2014.zip
!unzip -q annotations_trainval2014.zip

! pip install 2to3
!git clone https://github.com/cocodataset/cocoapi.git
!cd cocoapi
!2to3 -w
!cd PythonAPI
!python3 setup.py install

~2022-12-06 22:17:41-- http://images.cocodataset.org/zips/train2014.zip
Resolving images.cocodataset.org (images.cocodataset.org)... 3.5.125, 52.216.244.164, 54.231.233.57, ...
Connecting to images.cocodataset.org (images.cocodataset.org) [54.231.138.89]:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 13510573713 (12.6G) [application/zip]
Saving to: 'train2014.zip'

train2014.zip  100%[=====] 12.58G  69.8MB/s   in 2m 59s

~2022-12-06 22:20:40 (72.0 MB/s) - 'train2014.zip' saved [13510573713/13510573713]

~2022-12-06 22:22:50-- http://images.cocodataset.org/zips/val2014.zip
Resolving images.cocodataset.org (images.cocodataset.org)... 54.231.225.105, 52.217.79.20, 52.216.81.48, ...
Connecting to images.cocodataset.org (images.cocodataset.org) [54.231.138.89]:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 212872794 (241K) [application/zip]
Saving to: 'val2014.zip'

val2014.zip  100%[=====] 6.19G  66.9MB/s   in 1m 53s

2022-12-06 22:24:43 (56.2 MB/s) - 'val2014.zip' saved [6645013297/6645013297]

~2022-12-06 22:24:44-- http://images.cocodataset.org/annotations/annotations_trainval2014.zip
Resolving images.cocodataset.org (images.cocodataset.org)... 54.231.225.105, 52.217.79.20, 52.216.81.48, ...
Connecting to images.cocodataset.org (images.cocodataset.org) [54.231.225.105]:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 212872794 (241K) [application/zip]
Saving to: 'annotations_trainval2014.zip'

annotations_trainval2014 100%[=====] 241.16K  84.9MB/s   in 2.8s

2022-12-06 22:24:47 (84.9 MB/s) - 'annotations_trainval2014.zip' saved [252872794/252872794]

Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting 2to3
  Downloading 2to3-0.0.py3-none-any.whl (1.7 kB)
Installing collected packages: 2to3
Successfully installed 2to3-1.0
Closing into 'cocoapi'
remote: Enumerating objects: 975, done.
remote: Total 975 (delta 0), reused 0 (delta 0), pack-reused 975
Receiving objects: 100% (975/975), 11.12 MiB | 23.59 MiB/s, done.
Resolving deltas: 100% (576/576), done.
/content/cocoapi
RefactoringTool: Skipping optional fixer: buffer
RefactoringTool: Skipping optional fixer: idioms
RefactoringTool: Skipping optional fixer: set_literal
RefactoringTool: Skipping optional fixer: ws_comma
RefactoringTool: No changes to ./PythonAPI/setup.py
RefactoringTool: No changes to ./PythonAPI/pycocotools/_init_.py
RefactoringTool: Refactored ./PythonAPI/pycocotools/coco.py
--- ./PythonAPI/pycocotools/coco.py      (original)
+++ ./PythonAPI/pycocotools/coco.py      (refactored)
@@ -58,7 +58,7 @@
 import sys
 PYTHON_VERSION = sys.version_info[0]
- if from urllib import urretrieve
+ from urllib.request import urretrieve
 elif PYTHON_VERSION == 3:
- from urllib.request import urretrieve

@@ -83,7 +83,7 @@
 tic = time.time()
 dataset = json.load(open(annotation_file, 'r'))
 assert type(dataset)==dict, 'annotation file format (type(dataset))
- print('Done (t={:0.2f}s)'.format(time.time()- tic))
+ print('Done (t={:0.2f}s)'.format(time.time()- tic))
 self.dataset = dataset
 self.createIndex()

@@ -123,8 +123,8 @@
     Print information about the annotation file.
     :return:
     """
-    for key, value in self.dataset['info'].items():
-        print('{}: {}'.format(key, value))
+    for key, value in list(self.dataset['info'].items()):
+        print('{}: {}'.format(key, value))

def getAnnIds(self, imgIds=[], catIds=[], areaRng=[], iscrowd=None):
    """
@@ -187,7 +187,7 @@
 catIds = catIds if _isArrayLike(catIds) else []

 if len(imgIds) == len(catIds) == len(areaRng) == 0:
     ids = self._getImgIds()
 else:
     ids = list(self._getImgIds(keys))
     for i, catId in enumerate(catIds):
@@ -300,7 +300,7 @@
 elif datasetType == 'captions':
     for ann in anns:
         print(ann['caption'])
+
def loadRes(self, resFile):
    """
@@ -313,7 +313,7 @@
 print('Loading and preparing results...')
 tic = time.time()
- if type(resFile) == str and (PYTHON_VERSION == 2 and type(resFile) == unicode):
+ if type(resFile) == str or (PYTHON_VERSION == 2 and type(resFile) == str):
     anns = json.load(open(resFile))
     if type(resFile) == np.ndarray:
         anns = self.loadNumpyAnnotations(resFile)
@@ -357,7 +357,7 @@
 self._ann['area'] = (x1-x0)*(y1-y0)
 ann['bbox'] = [x0,y0,x1-x0,y1-y0]
- print('DONE (t={:0.2f}s)'.format(time.time()- tic))
+ print(('DONE (t={:0.2f}s)'.format(time.time()- tic))
+
res.dataset['annotations'] = anns
res.createIndex()

@@ -374,7 +374,7 @@
 print('Please specify target directory')
 return -1
 if len(imgIds) == 0:
     imgs = self._imgIds()
+
     imgs = list(self._imgIds.values())
 else:
     imgs = self._loadImgs(imgIds)
     N = len(imgs)
@@ -385,7 +385,7 @@
 os.path.join(tarDir, img['file_name'])
 if not os.path.exists(filename):
     urretrieve(img['coco_url'], filename)
- print('DONE (t={:0.2f}s)'.format(time.time()- tic))
+ print('DONE (t={:0.2f}s)'.format(time.time()- tic))
+
print('Downloaded {}({}) images (t={:0.1f}s)'.format(i, N, time.time()- tic))

def loadNumpyAnnotations(self, data):
    """
@@ -395,13 +395,13 @@
 print('Converting ndarray to lists...')
 assert type(data) == np.ndarray
- print((data.shape))
+ print((data.shape))
assert data.shape[1] == 7
N = data.shape[0]
ann = []
for i in range(N):
    if i % 1000000 == 0:
        print('({}/{})'.format(i, N))
+
        'image_id' : int(data[i, 0]),
        'bbox' : [ data[i, 1], data[i, 2], data[i, 3], data[i, 4] ],
        'category': PythonAPI/pycocotools/cocoapi.py (original)
+++ ./PythonAPI/pycocotools/cocoapi.py (refactored)
@@ -129,8 +129,8 @@
 # add backward compatibility if useSegm is specified in params
 if not p.useSegm is None:
     p.iouType = 'segm' if p.useSegm == 1 else 'bbox'
- print('useSegm (deprecated) is not None. Running {} evaluation'.format(p.iouType))
+ print('Evaluate annotation type "{}"'.format(p.iouType))
- print('DONE (t={:0.2f}s)'.format(time.time()- tic))
+ print('DONE (t={:0.2f}s)'.format(time.time()- tic))
+
p.imgIds = list(np.unique(p.imgIds))
if p.useCats:
    p.catIds = list(np.unique(p.catIds))
@@ -158,7 +158,7 @@
 self.paramsEval = copy.deepcopy(self.params)
 tic = time.time()
- print('DONE (t={:0.2f}s)'.format(toc-tic))
+ print(('DONE (t={:0.2f}s)'.format(toc-tic))
+
def computeIOU(self, imgId, catId):
    p = self.params
    # get inds to evaluate
    k_list = [ m for n, k in enumerate(p.catIds) if k in set(k) ]
    m_list = [ m for n, m in enumerate(p.maxDetIds) if m in set(k) ]
- list = [ m for n, m in enumerate(lambda x: tuple(x), p.areaRng) if a in set(a) ]
+ list = [ m for n, a in enumerate((tuple(x) for x in p.areaRng)) if a in set(a) ]
    a_list = [ n for n, a in enumerate(p.imgIds) if i in set(i) ]
    I0 = len(p.imgIds)
    A0 = len(p.areaRng)
@@ -417,7 +417,7 @@
 'scores': scores,
 }
 toc = time.time()
- print('DONE (t={:0.2f}s)'.format(toc-tic))
+ print(('DONE (t={:0.2f}s)'.format(toc-tic))
+
def summarize(self):
    """
@@ -453,7 +453,7 @@
 mean_s = -1
 else:
     mean_s = np.mean(scores[1:])
- print('t={:0.2f}s: mean_s={}'.format(toc-tic, mean_s))
+ print('t={:0.2f}s: mean_s={}'.format(toc-tic, mean_s))
+
return mean_s

def summarizeDet(self):
    stats = np.zeros((12,))
RefactoringTool: No changes to ./PythonAPI/pycocotools/mask.py
RefactoringTool: Files that were modified:
RefactoringTool: ./PythonAPI/setup.py
RefactoringTool: ./PythonAPI/pycocotools/_init_.py
RefactoringTool: ./PythonAPI/pycocotools/coco.py
RefactoringTool: ./PythonAPI/pycocotools/cocoapi.py
RefactoringTool: ./PythonAPI/pycocotools/mask.py
/content/cocoapi/PythonAPI
running install
running build_ext
creating pycocotools.egg-info
writing pycocotools.egg-info/PKG-INFO
writing dependency links to pycocotools.egg-info/dependency_links.txt
writing requirements to pycocotools.egg-info/requirements.txt
writing top-level names to pycocotools.egg-info/top_level.txt
writing manifest file 'pycocotools.egg-info/SOURCES.txt'
writing manifest file 'pycocotools.egg-info/SOURCES.txt'
Installing library code to build/lib.linux-x86_64/egg
running install_lib
running build_py
creating build
creating build/lib.linux-x86_64-3.8
creating build/lib.linux-x86_64-3.8/pycocotools
copying pycocotools/coco.py -> build/lib.linux-x86_64-3.8/pycocotools
copying pycocotools/cocoapi.py -> build/lib.linux-x86_64-3.8/pycocotools
copying pycocotools/_init_.py -> build/lib.linux-x86_64-3.8/pycocotools
copying pycocotools/mask.py -> build/lib.linux-x86_64-3.8/pycocotools
creating build/lib.linux-x86_64-3.8/pycocotools/mask
creating build/lib.linux-x86_64-3.8/pycocotools/mask.c
cythoning pycocotools/mask.pyx to pycocotools/mask.c
./usr/local/lib/python3.8/dist-packages/Cython/Compiler/Main.py:369: FutureWarning: Cython directive 'language_level' not set, using '2' (Py2). This will change in a later release! File: /content/cocoapi/PythonAPI/pycocotools/mask.pyx
tree = ParsingModule(p, pxd, full_module_name)
creating 'pycocotools' 'mask' extension
creating build/compile
creating build/temp.linux-x86_64-3.8
creating build/lib.linux-x86_64-3.8/pycocotools
x86_64-linux-gnu-gcc -pthread -Wno-unused-result -Wsign-compare -DNDEBUG -g -fwrapv -O2 -Wall -g -fstack-protector-strong -Wformat -Werror=format-security -g -fwrapv -O2 -g -fstack-protector-strong -Wformat -Werror=format-security -Wdate-time -D_FORTIFY_SOURCE=2 -fPIC -I/usr/local/lib/python3.8/dist-packages/numpy/core/include -I..common -I/usr/include/python3.8 -c ..common/maskApi.c -o build/temp.linux-x86_64-3.8/pycocotools/mask.o -Wno-cpp -Wno-unused-function -std=c99
../common/maskApi.c:461:7: warning: this 'for' clause does not guard... [-Wmisleading-indentation]
    for (k=0; k<R[i].cnts[j]; k++) { (M+=)w+= v*lv; }
                                ^
../common/maskApi.c:461:49: note: ...this statement, but the latter is misleadingly indented as if it were guarded by the 'for'
    for (k=0; k<R[i].cnts[j]; k++) { (M+=)w+= v*lv; }
                                ^
../common/maskApi.c: In function 'rleToPoly':
../common/maskApi.c:221:13: warning: this 'if' clause does not guard... [-Wmisleading-indentation]
    for(j=0; j<k; j++) x[j]=int(scale*x[j]*2+0.5); x[k]=0;
                    ^
../common/maskApi.c:166:54: note: ...this statement, but the latter is misleadingly indented as if it were guarded by the 'for'
    for(j=0; j<k; j++) x[j]=int(scale*x[j]*2+0.5); x[k]=0;
                    ^
../common/maskApi.c:167:3: warning: this 'for' clause does not guard... [-Wmisleading-indentation]
    for(j=0; j<k; j++) y[j]=int(scale*y[j]*2+0.5); y[k]=0;
        ^
../common/maskApi.c:167:54: note: ...this statement, but the latter is misleadingly indented as if it were guarded by the 'for'
    for(j=0; j<k; j++) y[j]=int(scale*y[j]*2+0.5); y[k]=0;
        ^
../common/maskApi.c: In function 'rleToStr':
../common/maskApi.c:212:7: warning: this 'if' clause does not guard... [-Wmisleading-indentation]
    if(more) c |= 0x20; c+=48; s[p]=c;
        ^
../common/maskApi.c:212:27: note: ...this statement, but the latter is misleadingly indented as if it were guarded by the 'if'
    if(more) c |= 0x20; c+=48; s[p]=c;
        ^
../common/maskApi.c:220:13: warning: this 'while' clause does not guard... [-Wmisleading-indentation]
    while( s[m] ) m+=cnts[malloc(sizeof(int)*n)]; m=0;
                    ^
../common/maskApi.c:221:22: warning: this 'if' clause does not guard... [-Wmisleading-indentation]
    if(m>n2) x+=long(cnts[m-2]); cnts[m]=uint x;
        ^
../common/maskApi.c:228:34: note: ...this statement, but the latter is misleadingly indented as if it were guarded by the 'if'
    if(m>n2) x+=long(cnts[m-2]); cnts[m]=uint x;
        ^
../common/maskApi.c: In function 'rleToBox':
../common/maskApi.c:141:13: note: 'xp' may be used uninitialized in this function [-Wmaybe-uninitialized]
    if(j%2==0) xp+=1; else if (j%2==0) xp+=0;
                    ^
x86_64-linux-gnu-gcc -pthread -Wno-unused-result -Wsign-compare -DNDEBUG -g -fwrapv -O2 -Wall -g -fstack-protector-strong -Wformat -Werror=format-security -g -fwrapv -O2 -g -fstack-protector-strong -Wformat -Werror=format-security -Wdate-time -D_FORTIFY_SOURCE=2 -fPIC -I/usr/local/lib/python3.8/dist-packages/numpy/core/include -I..common -I/usr/include/python3.8 -c ..common/maskApi.c -o build/temp.linux-x86_64-3.8/pycocotools/mask.o -Wno-cpp -Wno-unused-function -std=c99
x86_64-linux-gnu-gcc -pthread -shared -Wl,-O1,-Wl,-Bsymbolic-functions -Wl,-Bsymbolic-functions -g -fwrapv -O2 -Wl,-Bsymbolic-functions -g -fwrapv -O2 -g -fstack-protector-strong -Wformat -Werror=format-security -Wdate-time -D_FORTIFY_SOURCE=2 -fPIC -I/usr/local/lib/python3.8/dist-packages/numpy/core/include -I..common -I/usr/include/python3.8 -c ..common/maskApi.c -o build/temp.linux-x86_64-3.8/pycocotools/mask.o -Wno-cpp -Wno-unused-function -std=c99
creating build/lib.linux-x86_64
creating build/lib.linux-x86_64/egg/pycocotools
copying build/lib.linux-x86_64/egg/pycocotools
copying build/lib.linux-x86_64-3.8/pycocotools/mask.cpython-38-x86_64-linux-gnu.so -> build/lib.linux-x86_64-3.8/pycocotools
copying build/lib.linux-x86_64-3.8/pycocotools/coco.py -> build/lib.linux-x86_64/egg/pycocotools
copying build/lib.linux-x86_64-3.8/pycocotools/cocoapi.py -> build/lib.linux-x86_64/egg/pycocotools
copying build/lib.linux-x86_64-3.8/pycocotools/mask.py -> build/lib.linux-x86_64/egg/pycocotools
byte-compiling build/lib.linux-x86_64/egg/pycocotools/coco.py to coco.cpython-38.pyc
byte-compiling build/lib.linux-x86_64/egg/pycocotools/cocoapi.py to cocoapi.cpython-38.pyc
byte-compiling build/lib.linux-x86_64/egg/pycocotools/mask.py to mask.cpython-38.pyc
creating stub loader for pycocotools/mask.cpython-38-x86_64-linux-gnu.so
byte-compiling build/lib.linux-x86_64/egg/pycocotools/mask.py to _mask.cpython-38.pyc
creating build/bdist.linux-x86_64/egg/EGG-INFO
copying pycocotools.egg-info/PKG-INFO -> build/bdist.linux-x86_64/egg/EGG-INFO
copying pycocotools.egg-info/dependency_links.txt -> build/bdist.linux-x86_64/egg/EGG-INFO
copying pycocotools.egg-info/requirements.txt -> build/bdist.linux-x86_64/egg/EGG-INFO
copying pycocotools.egg-info/top_level.txt -> build/bdist.linux-x86_64/egg/EGG-INFO
writing build/bdist.linux-x86_64/egg/EGG-INFO/native_libs.txt
zip_safe flag not set; analyzing archive contents...
pycocotools._pycachaes._mask.cpython-38: module references file__
creating dist
creating 'dist/pycocotools-2.0-py3.8-linux-x86_64.egg' and adding 'build/bdist.linux-x86_64/egg' to it
removing 'build/bdist.linux-x86_64/egg' (and everything under it)
Processing pycocotools-2.0-py3.8-linux-x86_64.egg
Searching for matplotlib==3.2.2
Best match: matplotlib 3.2.2
Adding matplotlib 3.2.2 to easy-install.pth file

Using /usr/local/lib/python3.8/dist-packages
Searching for Cython==0.29.32
Searching for Cython==0.29.32
Adding Cython 0.29.32 to easy-install.pth file
Installing cygdb script to /usr/local/bin
Installing cythonize script to /usr/local/bin
Installing cythonize script to /usr/local/bin

Using /usr/local/lib/python3.8/dist-packages
Searching for setuptools==57.4.0
Best match: setuptools 57.4.0
Adding setuptools 57.4.0 to easy-install.pth file

Using /usr/local/lib/python3.8/dist-packages
Searching for kiwisolver==1.4.4
Best match: kiwisolver 1.4.4
Adding kiwisolver 1.4.4 to easy-install.pth file

Using /usr/local/lib/python3.8/dist-packages
Searching for cycle==0.11.0
Best match: cycle 0.11.0
Adding cycle 0.11.0 to easy-install.pth file

Using /usr/local/lib/python3.8/dist-packages
Searching for pyarsing==3.0.9
Best match: pyarsing 3.0.9
Adding pyarsing 3.0.9 to easy-install.pth file

Using /usr/local/lib/python3.8/dist-packages
Searching for numpy==1.21.6
Best match: numpy 1.21.6
Adding numpy 1.21.6 to easy-install.pth file
Installing f2py script to /usr/local/bin
Installing f2py3.8 script to /usr/local/bin

Using /usr/local/lib/python3.8/dist-packages
Searching for python-dateutil==2.8.2
Best match: python-dateutil 2.8.2
Adding python-dateutil 2.8.2 to easy-install.pth file

Using /usr/local/lib/python3.8/dist-packages
Searching for six==1.15.0
Best match: six 1.15.0
Adding six 1.15.0 to easy-install.pth file

Using /usr/local/lib/python3.8/dist-packages
Finished processing dependencies for pycocotools==2.0

In [ ]:

!cd /content/
/content

1. importing the required library files

In [ ]:

# Importing Data From COCO
from pycocotools import COCO, cocoEval, mask
from pycocotools import mask as maskUtils
import array
import numpy as np
import skimage.io as io
import matplotlib.pyplot as plt
import pylab
import os
pylab.rcParams['figure.figsize'] = (8.0, 10.0)
matplotlib inline

1. splitting the training and validating dataset in the dataset

In [ ]:

CATEGORY_NAMES = 'person'

ANNOTATION_FILE_VAL = '/content/annotations/instances_val2014.json'
ANNOTATION_FILE_TRAIN = '/content/annotations/instances_train2014.json'

coco_train = COCO(ANNOTATION_FILE_TRAIN)
catIds_train = coco_train.getCatIds(catIds=CATEGORY_NAMES)
imgIds_train = coco_train.getImgIds(catIds=catIds_train)
imgDict_train = coco_train.loadImgs(imgIds_train)
len(imgIds_train), len(catIds_train)

coco_val = COCO(ANNOTATION_FILE_VAL)
catIds_val = coco_val.getCatIds(catIds=CATEGORY_NAMES)
imgIds_val = coco_val.getImgIds(catIds=catIds_val)
imgDict_val = coco_val.loadImgs(imgIds_val)
len(imgIds_val), len(catIds_val)

loading annotations into memory...
Done (t=15.12s)
creating index...
index created!
loading annotations into memory...
Done (t=5.86s)
creating index...
index created!
(21634, 1)

Out [ ]:

from random import shuffle

shuffle(imgIds_train)
shuffle(imgIds_val)

imgIds_train = imgIds_train[0:6000]
imgIds_val = imgIds_val[0:600]

train_images_person = ["COCO_train2014_0:012d.jpg".format(ids) for ids in imgIds_train]
val_images_person = ["COCO_val2014_0:012d.jpg".format(ids) for ids in imgIds_val]

print(len(train_images_person), len(val_images_person))

6000 600

train_images_person = ["COCO_train2014_0:012d.jpg".format(ids) for ids in imgIds_train]
print(train_images_person)
file_name = os.path.join('/content/train2014') - set(train_images_person)
for file_name in set(train2014) + file_name
if os.path.exists(file_name):
    os.remove(file_name)
```



```
5500
5501
5502
5503
5504
5505
5506
5507
5508
5509
5510
5511
5512
5513
5514
5515
5516
5517
5518
5519
5520
5521
5522
5523
5524
5525
5526
5527
5528
5529
5530
5531
5532
5533
5534
5535
5536
5537
5538
5539
5540
5541
5542
5543
5544
5545
5546
5547
5548
5549
5550
5551
5552
5553
5554
5555
5556
5557
5558
5559
5560
5561
5562
5563
5564
5565
5566
5567
5568
5569
5570
5571
5572
5573
5574
5575
5576
5577
5578
5579
5580
5581
5582
5583
5584
5585
5586
5587
5588
5589
5590
5591
5592
5593
5594
5595
5596
5597
5598
5599
5600
5601
5602
5603
5604
5605
5606
5607
5608
5609
5610
5611
5612
5613
5614
5615
5616
5617
5618
5619
5620
5621
5622
5623
5624
5625
5626
5627
5628
5629
5630
5631
5632
5633
5634
5635
5636
5637
5638
5639
5640
5641
5642
5643
5644
5645
5646
5647
5648
5649
5650
5651
5652
5653
5654
5655
5656
5657
5658
5659
5660
5661
5662
5663
5664
5665
5666
5667
5668
5669
5670
5671
5672
5673
5674
5675
5676
5677
5678
5679
5680
5681
5682
5683
5684
5685
5686
5687
5688
5689
5690
5691
5692
5693
5694
5695
5696
5697
5698
5699
5700
5701
5702
5703
5704
5705
5706
5707
5708
5709
5710
5711
5712
5713
5714
5715
5716
5717
5718
5719
5720
5721
5722
5723
5724
5725
5726
5727
5728
5729
5730
5731
5732
5733
5734
5735
5736
5737
5738
5739
5740
5741
5742
5743
5744
5745
5746
5747
5748
5749
5750
5751
5752
5753
5754
5755
5756
5757
5758
5759
5760
5761
5762
5763
5764
5765
5766
5767
5768
5769
5770
5771
5772
5773
5774
5775
5776
5777
5778
5779
5780
5781
5782
5783
5784
5785
5786
5787
5788
5789
5790
5791
5792
5793
5794
5795
5796
5797
5798
5799
5800
5801
5802
5803
5804
5805
5806
5807
5808
5809
5810
5811
5812
5813
5814
5815
5816
5817
5818
5819
5820
5821
5822
5823
5824
5825
5826
5827
5828
5829
5830
5831
5832
5833
5834
5835
5836
5837
5838
5839
5840
5841
5842
5843
5844
5845
5846
5847
5848
5849
5850
5851
5852
5853
5854
5855
5856
5857
5858
5859
5860
5861
5862
5863
5864
5865
5866
5867
5868
5869
5870
5871
5872
5873
5874
5875
5876
5877
5878
5879
5880
5881
5882
5883
5884
5885
5886
5887
5888
5889
5890
5891
5892
5893
5894
5895
5896
5897
5898
5899
5900
5901
5902
5903
5904
5905
5906
5907
5908
5909
5910
5911
5912
5913
5914
5915
5916
5917
5918
5919
5920
5921
5922
5923
5924
5925
5926
5927
5928
5929
5930
5931
5932
5933
5934
5935
5936
5937
5938
5939
5940
5941
5942
5943
5944
5945
5946
5947
5948
5949
5950
5951
5952
5953
5954
5955
5956
5957
5958
5959
5960
5961
5962
5963
5964
5965
5966
5967
5968
5969
5970
5971
5972
5973
5974
5975
5976
5977
5978
5979
5980
5981
5982
5983
5984
5985
5986
5987
5988
5989
5990
5991
5992
5993
5994
5995
5996
5997
5998
5999
6000
```

```
In [ ]:
rm -rf annotations/
rm -rf train2014.zip
rm -rf val2014.zip
rm -rf annotations_trainval2014.zip
```

```
In [ ]:
import os
import sys
import random

import numpy as np
import cv2
import tensorflow as tf
from tensorflow.keras.layers import *
from tensorflow.keras.models import *
from tensorflow.keras.optimizers import *

seed = 2019

random.seed = seed
np.random.seed = seed
```

```
In [ ]:
class DataGen(tf.keras.utils.Sequence):

    def __init__(self, path_input, path_mask, batch_size = 8, image_size = 128):
        self.ids = os.listdir(path_input)
        self.path_input = path_input
        self.path_mask = path_mask
        self.batch_size = batch_size
        self.image_size = image_size
        self.on_epoch_end()

    def __load__(self, id_name):
        image_path = os.path.join(self.path_input, id_name)
        mask_path = os.path.join(self.path_mask, id_name)
        image = cv2.imread(image_path, 1) # 1 specifies RGB format
        image = cv2.resize(image, (self.image_size, self.image_size)) # resizing before inserting to the network
        mask = cv2.imread(mask_path, -1)
        mask = cv2.resize(mask, (self.image_size, self.image_size))
        mask = mask.reshape((self.image_size, self.image_size, 1))

        #normalize image
        image = image / 255.0
        mask = mask / 255.0

        return image, mask

    def __getitem__(self, index):
        if (index + 1)*self.batch_size > len(self.ids):
            self.batch_size = len(self.ids) - index + self.batch_size

        file_batch = self.ids[index * self.batch_size : (index + 1) * self.batch_size]
        images = []
        masks = []

        for id_name in file_batch:
            img, _mask = self.__load__(id_name)
            images.append(img)
            masks.append(_mask)

        images = np.array(images)
        masks = np.array(masks)

        return images, masks

    def on_epoch_end(self):
        pass

    def __len__(self):
        return int(np.ceil(len(self.ids) / float(self.batch_size)))
```

```
In [ ]:
def down_block(
    input_tensor,
    no_filters,
    kernel_size=(3, 3),
    strides=(1, 1),
    padding="same",
    kernel_initializer="he_normal",
    max_pool_window=(2, 2),
    max_pool_stride=(2, 2)
):
    conv = Conv2D(
        filters=no_filters,
        kernel_size=kernel_size,
        strides=strides,
        activation=None,
        padding=padding,
        kernel_initializer=kernel_initializer
    )(input_tensor)

    conv = BatchNormalization(scale=True)(conv)

    conv = Activation("relu")(conv)

    conv = Conv2D(
        filters=no_filters,
        kernel_size=kernel_size,
        strides=strides,
        activation=None,
        padding=padding,
        kernel_initializer=kernel_initializer
    )(conv)

    conv = BatchNormalization(scale=True)(conv)

    conv = Activation("relu")(conv)

    # conv for skip connection
    conv = Activation("relu")(conv)

    pool = MaxPooling2D(pool_size=max_pool_window, strides=max_pool_stride)(conv)

    return conv, pool
```

```
In [ ]:
def bottle_neck(
    input_tensor,
    no_filters,
    kernel_size=(3, 3),
    strides=(1, 1),
    padding="same",
    kernel_initializer="he_normal"
):
    conv = Conv2D(
        filters=no_filters,
        kernel_size=kernel_size,
        strides=strides,
        activation=None,
        padding=padding,
        kernel_initializer=kernel_initializer
    )(input_tensor)

    conv = BatchNormalization(scale=True)(conv)

    conv = Activation("relu")(conv)

    conv = Conv2D(
        filters=no_filters,
        kernel_size=kernel_size,
        strides=strides,
        activation=None,
        padding=padding,
        kernel_initializer=kernel_initializer
    )(conv)

    conv = BatchNormalization(scale=True)(conv)

    conv = Activation("relu")(conv)

    return conv
```

```
In [ ]:
def up_block(
    input_tensor,
    no_filters,
    skip_connection,
    kernel_size=(3, 3),
    strides=(1, 1),
    upsampling_factor = (2,2),
    max_pool_window = (2,2),
    padding="same",
    kernel_initializer="he_normal"):
    conv = Conv2D(
        filters=no_filters,
        kernel_size=kernel_size,
        strides=strides,
        activation = None,
        padding=padding,
        kernel_initializer=kernel_initializer
    )(Upsampling2D(size = upsampling_factor)(input_tensor))

    conv = BatchNormalization(scale=True)(conv)

    conv = Activation("relu")(conv)

    conv = concatenate([skip_connection, conv], axis = -1)

    conv = Conv2D(
        filters=no_filters,
        kernel_size=kernel_size,
        strides=strides,
        activation=None,
        padding=padding,
        kernel_initializer=kernel_initializer
    )(conv)

    conv = BatchNormalization(scale=True)(conv)

    conv = Activation("relu")(conv)

    return conv
```

```
In [ ]:
def output_block(input_tensor,
    padding="same",
    kernel_initializer="he_normal"
):
    conv = Conv2D(
        filters=2,
        kernel_size=(3,3),
        strides=(1,1),
        activation="relu",
        padding=padding,
        kernel_initializer=kernel_initializer
    )(input_tensor)

    conv = Conv2D(
        filters=1,
        kernel_size=(1,1),
        strides=(1,1),
        activation="sigmoid",
        padding=padding,
        kernel_initializer=kernel_initializer
    )(conv)

    return conv
```

UNET MODEL

```
In [ ]:
def UNet(input_shape = (128,128,3)):

    filter_size = (64,128,256,512,1024)

    inputs = Input(shape = input_shape)

    d1, p1 = down_block(input_tensor=inputs,
                        no_filters=filter_size[0],
                        kernel_size = (3,3),
                        strides=(1,1),
                        padding="same",
                        kernel_initializer="he_normal",
                        max_pool_window=(2,2),
                        max_pool_stride=(2,2))

    d2, p2 = down_block(input_tensor=p1,
                        no_filters=filter_size[1],
                        kernel_size = (3,3),
                        padding="same",
                        kernel_initializer="he_normal",
                        max_pool_window=(2,2),
                        max_pool_stride=(2,2))

    d3, p3 = down_block(input_tensor=p2,
                        no_filters=filter_size[2],
                        kernel_size = (3,3),
                        strides=(1,1),
                        padding="same",
                        kernel_initializer="he_normal",
                        max_pool_window=(2,2),
                        max_pool_stride=(2,2))

    d4, p4 = down_block(input_tensor=p3,
                        no_filters=filter_size[3],
                        kernel_size = (3,3),
                        strides=(1,1),
                        padding="same",
                        kernel_initializer="he_normal",
                        max_pool_window=(2,2),
                        max_pool_stride=(2,2))

    b = bottle_neck(input_tensor=p4,
                    no_filters=filter_size[4],
                    kernel_size = (3,3),
                    strides=(1,1),
                    padding="same",
                    kernel_initializer="he_normal")

    u4 = up_block(input_tensor = b,
                  no_filters = filter_size[3],
                  skip_connection = d4,
                  kernel_size=(3, 3),
                  padding="same",
                  upsampling_factor = (2,2),
                  max_pool_window = (2,2),
                  kernel_initializer="he_normal")

    u3 = up_block(input_tensor = u4,
                  no_filters = filter_size[2],
                  skip_connection = d3,
                  kernel_size=(3, 3),
                  padding="same",
                  upsampling_factor = (2,2),
                  max_pool_window = (2,2),
                  kernel_initializer="he_normal")

    u2 = up_block(input_tensor = u3,
                  no_filters = filter_size[1],
                  skip_connection = d2,
                  kernel_size=(3, 3),
                  padding="same",
                  upsampling_factor = (2,2),
                  max_pool_window = (2,2),
                  kernel_initializer="he_normal")

    u1 = up_block(input_tensor = u2,
                  no_filters = filter_size[0],
                  skip_connection = d1,
                  kernel_size=(3, 3),
                  padding="same",
                  upsampling_factor = (2,2),
                  max_pool_window = (2,2),
                  kernel_initializer="he_normal")

    output = output_block(input_tensor=u1,
                          padding = "same",
                          kernel_initializer= "he_normal")

    model = Model(inputs = inputs, outputs = output)

    return model
```

```
In [ ]:
model = UNet(input_shape = (128,128,3))
model.compile(optimizer = Adam(lr = 1e-4), loss = 'binary_crossentropy', metrics = ['accuracy'])
```

```
!usr/local/lib/python3.8/dist-packages/keras/optimizers/optimizer_v2/adam.py:110: UserWarning: The 'lr' argument
is deprecated, use 'learning_rate' instead.
  super(Adam, self).__init__(name, **kwargs)
```

```
In [ ]:
image_size = 128
epochs = 1
batch_size = 8
```

```
In [ ]:
train_gen = DataGen(path_input = "/content/train2014", path_mask = "/content/mask_train_2014", batch_size =
val_gen = DataGen(path_input = "/content/val2014", path_mask = "/content/mask_val_2014", batch_size = batch_s
train_steps = len(os.listdir("/content/train2014"))/batch_size

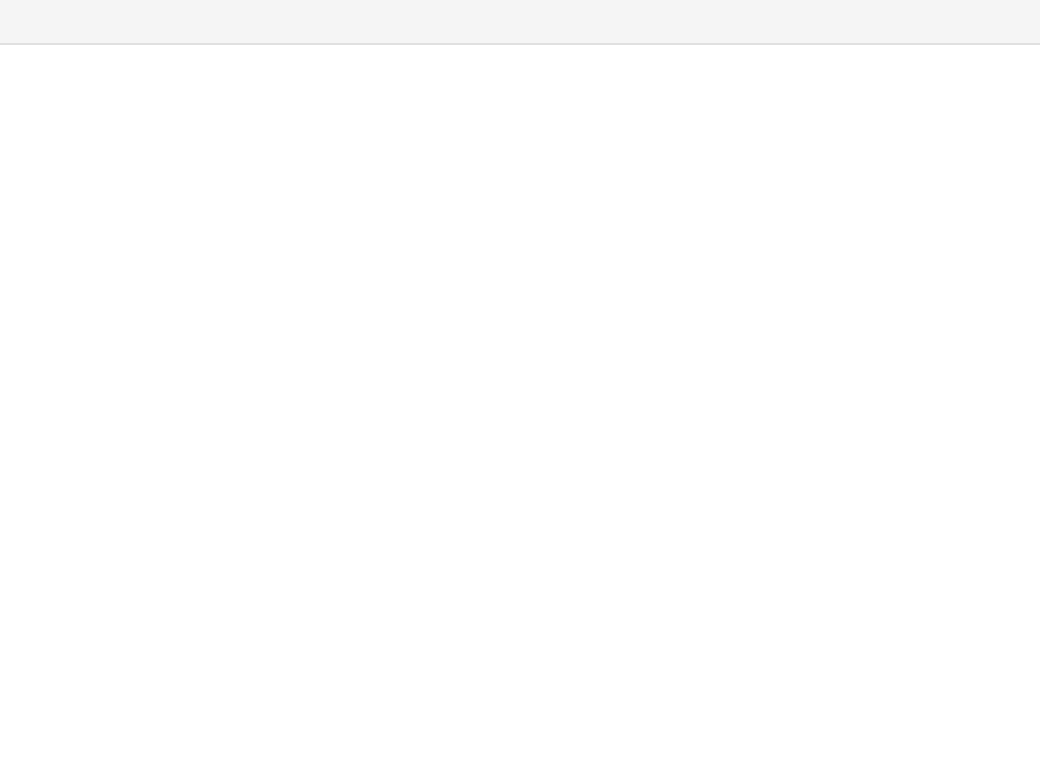
model.fit_generator(train_gen, validation_data = val_gen, steps_per_epoch = train_steps, epochs=epochs)
```

```
!python-input-27-303d4a120fbd:8: UserWarning: 'Model.fit_generator' is deprecated and will be removed in a f
1.0.0 version. Please use 'Model.fit', which supports generators.
model.fit_generator(train_gen, validation_data = val_gen, steps_per_epoch = train_steps, epochs=epochs)
546/750 [=====>.....] - ETA: 55:56 - loss: 0.3621 - accuracy: 0.8184
```



```
In [31]: import cv2
from matplotlib import pyplot as plt
fig = plt.figure(figsize=(10, 7))
rows = 2
columns = 2
Image1 = cv2.imread('/content/drive/MyDrive/imagesof hammer1.jpeg')
Image2 = cv2.imread('/content/drive/MyDrive/imagesof hammer2.jpeg')
fig.add_subplot(rows, columns, 1)
plt.imshow(Image1)
plt.axis('off')
plt.title("input image")
fig.add_subplot(rows, columns, 2)
plt.imshow(Image2)
plt.axis('off')
plt.title("masked image")

Out[31]: Text(0.5, 1.0, 'masked image')
```



In []: