Mask R-CNN - Train on Shapes Dataset

This notebook shows how to train Mask R-CNN on your own dataset. To keep things simple we use a synthetic dataset of shapes (squares, triangles, and circles) which enables fast training. You'd still need a GPU, though, because the network backbone is a Resnet101, which would be too slow to train on a CPU. On a GPU, you can start to get okay-ish results in a few minutes, and good results in less than an hour.

The code of the *Shapes* dataset is included below. It generates images on the fly, so it doesn't require downloading any data. And it can generate images of any size, so we pick a small image size to train faster.

In [1]:

```
import os
import sys
import random
import math
import re
import time
import numpy as np
import cv2
import matplotlib
import matplotlib.pyplot as plt
from utils.raw_data import *
import skimage

# Root directory of the project
ROOT_DIR = os.path.abspath("../")

# Import Mask RCNN
sys.path.append(ROOT_DIR)  # To find local version of the library
from mrcnn.config import Config
from mrcnn import utils
import mrcnn.model as modellib
from mrcnn import visualize
from mrcnn.model import log

%matplotlib inline

# Directory to save logs and trained model
MODEL_DIR = os.path.join(ROOT_DIR, "logs")

# Local path to trained weights file
COCO_MODEL_PATH = os.path.join(ROOT_DIR, "mask_rcnn_coco.h5")
# Download COCO trained weights from Releases if needed
if not os.path.exists(COCO_MODEL_PATH):
    utils.download_trained_weights(COCO_MODEL_PATH)
```

```
/home/paperspace/anaconda3/lib/python3.6/site-packages/h5py/__init__.
py:36: FutureWarning: Conversion of the second argument of issubdtype
from `float` to `np.floating` is deprecated. In future, it will be tr
eated as `np.float64 == np.dtype(float).type`.
  from ._conv import register_converters as _register_converters
Using TensorFlow backend.
```

Configurations

In [2]:

```
class PedConfig(Config):
    """Configuration for training on the toy shapes dataset.
    Derives from the base Config class and overrides values specific
    to the toy shapes dataset.
    """
    # Give the configuration a recognizable name
    NAME = "ped"

    # Train on 1 GPU and 8 images per GPU. We can put multiple images
 on each
    # GPU because the images are small. Batch size is 8 (GPUs * image
s/GPU).
    GPU_COUNT = 1
    IMAGES_PER_GPU = 2

    # Number of classes (including background)
    NUM_CLASSES = 1 + 3  # background + 3 shapes

    # Use small images for faster training. Set the limits of the sma
ll side
    # the large side, and that determines the image shape.
    IMAGE_MIN_DIM = 480
    IMAGE_MAX_DIM = 640
    IMAGE_CHANNEL_COUNT = 3
    USE_MINI_MASK = False

    # Use smaller anchors because our image and objects are small
    RPN_ANCHOR_SCALES = (8, 16, 32, 64, 128)  # anchor side in pixels

    # Reduce training ROIs per image because the images are small and
 have
    # few objects. Aim to allow ROI sampling to pick 33% positive ROI
s.
    TRAIN_ROIS_PER_IMAGE = 32

    # Use a small epoch since the data is simple
    STEPS_PER_EPOCH = 100

    # use small validation steps since the epoch is small
    VALIDATION_STEPS = 5

config = PedConfig()
config.display()
```

```
Configurations:
BACKBONE                       resnet101
BACKBONE_STRIDES               [4, 8, 16, 32, 64]
BATCH_SIZE                     2
BBOX_STD_DEV                   [0.1 0.1 0.2 0.2]
COMPUTE_BACKBONE_SHAPE         None
DETECTION_MAX_INSTANCES        100
DETECTION_MIN_CONFIDENCE       0.7
DETECTION_NMS_THRESHOLD        0.3
FPN_CLASSIF_FC_LAYERS_SIZE     1024
GPU_COUNT                      1
GRADIENT_CLIP_NORM             5.0
IMAGES_PER_GPU                 2
IMAGE_CHANNEL_COUNT            3
IMAGE_MAX_DIM                  640
IMAGE_META_SIZE                16
IMAGE_MIN_DIM                  480
IMAGE_MIN_SCALE                0
IMAGE_RESIZE_MODE              square
IMAGE_SHAPE                    [640 640   3]
LEARNING_MOMENTUM              0.9
LEARNING_RATE                  0.001
LOSS_WEIGHTS                   {'rpn_class_loss': 1.0, 'rpn_bbox_los
s': 1.0, 'mrcnn_class_loss': 1.0, 'mrcnn_bbox_loss': 1.0, 'mrcnn_mask
_loss': 1.0}
MASK_POOL_SIZE                 14
MASK_SHAPE                     [28, 28]
MAX_GT_INSTANCES               100
MEAN_PIXEL                     [123.7 116.8 103.9]
MINI_MASK_SHAPE                (56, 56)
NAME                           ped
NUM_CLASSES                    4
POOL_SIZE                      7
POST_NMS_ROIS_INFERENCE        1000
POST_NMS_ROIS_TRAINING         2000
PRE_NMS_LIMIT                  6000
ROI_POSITIVE_RATIO             0.33
RPN_ANCHOR_RATIOS              [0.5, 1, 2]
RPN_ANCHOR_SCALES              (8, 16, 32, 64, 128)
RPN_ANCHOR_STRIDE              1
RPN_BBOX_STD_DEV               [0.1 0.1 0.2 0.2]
RPN_NMS_THRESHOLD              0.7
RPN_TRAIN_ANCHORS_PER_IMAGE    256
STEPS_PER_EPOCH                100
TOP_DOWN_PYRAMID_SIZE          256
TRAIN_BN                       False
TRAIN_ROIS_PER_IMAGE           32
USE_MINI_MASK                  False
USE_RPN_ROIS                   True
VALIDATION_STEPS               5
WEIGHT_DECAY                   0.0001
```

Notebook Preferences

In [3]:

```
def get_ax(rows=1, cols=1, size=8):
    """Return a Matplotlib Axes array to be used in
    all visualizations in the notebook. Provide a
    central point to control graph sizes.

    Change the default size attribute to control the size
    of rendered images
    """
    _, ax = plt.subplots(rows, cols, figsize=(size*cols, size*rows))
    return ax
```

Dataset

Create a synthetic dataset

Extend the Dataset class and add a method to load the shapes dataset, `load_shapes()`, and override the following methods:

- load_image()
- load_mask()
- image_reference()

In [4]:

```python
class PedDataset(utils.Dataset):
    """Generates the shapes synthetic dataset. The dataset consists of simple
    shapes (triangles, squares, circles) placed randomly on a blank surface.
    The images are generated on the fly. No file access required.
    """

    def load_ped(self, load_path="../Datasets/", istrain=True, st_index=None, batch_size=20):
        """Generate the requested number of synthetic images.
        count: number of images to generate.
        height, width: the size of the generated images.
        """
        # Add classes
        self.add_class("ped", 1, "bike")
        self.add_class("ped", 2, "car")
        self.add_class("ped", 3, "person")

        if istrain is True:
            self.X = np.load(load_path+'X_train.npy')
            self.X_mask = np.load(load_path+'X_mask_train.npy')
            self.y = np.load(load_path+'y_train.npy')


        if istrain is False:
            self.X = np.load(load_path+'X_val.npy')
            self.X_mask = np.load(load_path+'X_mask_val.npy')
            self.y = np.load(load_path+'y_val.npy')

        self.N = self.X.shape[0]

        if istrain:
            self.X = self.X[st_index:st_index+batch_size]
            self.X_mask = self.X_mask[st_index:st_index+batch_size]
            self.y = self.y[st_index:st_index+batch_size]

        for i in range(self.X.shape[0]):
            self.add_image("ped", image_id=i, path=None)

    def load_image(self, image_id, istrain=True):
        """Generate an image from the specs of the given image ID.
        Typically this function loads the image from a file, but
        in this case it generates the image on the fly from the
        specs in image_info.
        """
        image = self.X[image_id]
        if image.ndim != 3:
            image = skimage.color.gray2rgb(image)
        return image

    def image_reference(self, image_id):
        """Return the shapes data of the image."""
        return self.X[image_id].shape

    def load_mask(self, image_id):
        """Generate instance masks for shapes of the given image ID.
        """
        mask = np.zeros((self.X_mask.shape[1], self.X_mask.shape[2],
1))
        mask[:, :, 0] = np.logical_not(self.X_mask[image_id]).astype(
np.bool_)
#         print('masks are bool!')
        class_ids = np.array([self.y[image_id]+1])
        return mask, class_ids.astype(np.int32)


    def prepare(self):
        """Prepares the Dataset class for use.

        TODO: class map is not supported yet. When done, it should ha
ndle mapping
              classes from different datasets to the same class ID.
        """

        self.num_classes = int(max(self.y[:]) + 2)
        self.class_ids = np.arange(self.num_classes)
        self.class_names = ["BG", "BIKE", "CAR", "PERSON"]
        self.num_images = self.X.shape[0]
        self._image_ids = np.arange(self.num_images)

        # Mapping from source class and image IDs to internal IDs
        self.class_from_source_map = {"{}.{}".format(info['source'],
info['id']): id
                                      for info, id in zip(self.class_
info, self.class_ids)}
        self.image_from_source_map = {"{}.{}".format(info['source'],
info['id']): id
                                      for info, id in zip(self.image_
info, self.image_ids)}

        # Map sources to class_ids they support
        self.sources = list(set([i['source'] for i in self.class_info
]))
        self.source_class_ids = {}
        # Loop over datasets
        for source in self.sources:
            self.source_class_ids[source] = []
            # Find classes that belong to this dataset
            for i, info in enumerate(self.class_info):
                # Include BG class in all datasets
                if i == 0 or source == info['source']:
                    self.source_class_ids[source].append(i)
```

In [5]:

```
# Training dataset
dataset_train = PedDataset()
dataset_train.load_ped(istrain=True, st_index=0)
dataset_train.prepare()
```

In [6]:

```
# Validation dataset
dataset_val = PedDataset()
dataset_val.load_ped(istrain=False)
dataset_val.prepare()
```

Ceate Model

In [7]:

```
# Create model in training mode
model = modellib.MaskRCNN(mode="training", config=config,
                          model_dir=MODEL_DIR)
```

In [10]:

```
# Train Network

num_epoch = 10
batch_size = 20
N = dataset_train.N / batch_size
lr = config.LEARNING_RATE
for e in range(num_epoch):
    for b in range(int(N)):
        dataset_train.load_ped(istrain=True, st_index=b*batch_size)
        dataset_train.prepare()
        print("Epoch %d: Training batch %d out of %d" % (e, b, int(N
)))
        # Which weights to start with?
        if b == 0:
            model.load_weights(COCO_MODEL_PATH, by_name=True,
                          exclude=["mrcnn_class_logits", "mrcnn_bbox
_fc",
                                   "mrcnn_bbox", "mrcnn_mask"])
        else:
            # Load the last model you trained and continue training
            model.load_weights(model.find_last(), by_name=True)

        model.train(dataset_train, dataset_val,
                    learning_rate=lr,
                    epochs=1,
                    layers='heads')

        lr = lr * 0.90
```

```
Epoch 0: Training batch 0 out of 40
WARNING:tensorflow:From /home/paperspace/anaconda3/lib/python3.6/site
-packages/tensorflow/contrib/learn/python/learn/datasets/base.py:198:
retry (from tensorflow.contrib.learn.python.learn.datasets.base) is d
eprecated and will be removed in a future version.
Instructions for updating:
Use the retry module or similar alternatives.

Starting at epoch 0. LR=0.001

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
```

```
/home/paperspace/anaconda3/lib/python3.6/site-packages/tensorflow/pyt
hon/ops/gradients_impl.py:100: UserWarning: Converting sparse Indexed
Slices to a dense Tensor of unknown shape. This may consume a large a
mount of memory.
  "Converting sparse IndexedSlices to a dense Tensor of unknown shap
e. "
```

```
Epoch 1/1
100/100 [==============================] - 111s 1s/step - loss: 2.785
9 - rpn_class_loss: 0.0184 - rpn_bbox_loss: 1.6422 - mrcnn_class_los
s: 0.2197 - mrcnn_bbox_loss: 0.5045 - mrcnn_mask_loss: 0.4011 - val_l
oss: 1.9911 - val_rpn_class_loss: 0.0130 - val_rpn_bbox_loss: 1.2895
- val_mrcnn_class_loss: 0.0740 - val_mrcnn_bbox_loss: 0.3562 - val_mr
cnn_mask_loss: 0.2584
Epoch 0: Training batch 1 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.0009000000000000001

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 105s 1s/step - loss: 1.330
5 - rpn_class_loss: 0.0151 - rpn_bbox_loss: 0.7286 - mrcnn_class_los
s: 0.0598 - mrcnn_bbox_loss: 0.2592 - mrcnn_mask_loss: 0.2678 - val_l
oss: 2.2950 - val_rpn_class_loss: 0.0051 - val_rpn_bbox_loss: 1.5279
- val_mrcnn_class_loss: 0.0554 - val_mrcnn_bbox_loss: 0.3730 - val_mr
cnn_mask_loss: 0.3335
Epoch 0: Training batch 2 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.0008100000000000001

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
```

```
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 101s 1s/step - loss: 1.083
1 - rpn_class_loss: 0.0087 - rpn_bbox_loss: 0.6066 - mrcnn_class_los
s: 0.0522 - mrcnn_bbox_loss: 0.1964 - mrcnn_mask_loss: 0.2191 - val_l
oss: 2.5264 - val_rpn_class_loss: 0.0174 - val_rpn_bbox_loss: 1.6280
- val_mrcnn_class_loss: 0.1139 - val_mrcnn_bbox_loss: 0.5140 - val_mr
cnn_mask_loss: 0.2532
Epoch 0: Training batch 3 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.000729

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
```

```
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 101s 1s/step - loss: 1.211
1 - rpn_class_loss: 0.0117 - rpn_bbox_loss: 0.7559 - mrcnn_class_los
s: 0.0425 - mrcnn_bbox_loss: 0.1573 - mrcnn_mask_loss: 0.2438 - val_l
oss: 2.3194 - val_rpn_class_loss: 0.0105 - val_rpn_bbox_loss: 1.5851
- val_mrcnn_class_loss: 0.1528 - val_mrcnn_bbox_loss: 0.3942 - val_mr
cnn_mask_loss: 0.1769
Epoch 0: Training batch 4 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.0006561000000000001

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
```

```
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 105s 1s/step - loss: 0.760
1 - rpn_class_loss: 0.0066 - rpn_bbox_loss: 0.3075 - mrcnn_class_los
s: 0.0477 - mrcnn_bbox_loss: 0.1107 - mrcnn_mask_loss: 0.2876 - val_l
oss: 1.3385 - val_rpn_class_loss: 0.0037 - val_rpn_bbox_loss: 0.8166
- val_mrcnn_class_loss: 0.0713 - val_mrcnn_bbox_loss: 0.2829 - val_mr
cnn_mask_loss: 0.1639
Epoch 0: Training batch 5 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.00059049

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 102s 1s/step - loss: 1.106
8 - rpn_class_loss: 0.0083 - rpn_bbox_loss: 0.6501 - mrcnn_class_los
s: 0.0653 - mrcnn_bbox_loss: 0.1591 - mrcnn_mask_loss: 0.2240 - val_l
oss: 2.2187 - val_rpn_class_loss: 0.0064 - val_rpn_bbox_loss: 1.4020
- val_mrcnn_class_loss: 0.1336 - val_mrcnn_bbox_loss: 0.4555 - val_mr
cnn_mask_loss: 0.2212
Epoch 0: Training batch 6 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.000531441

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
```

```
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 102s 1s/step - loss: 1.055
9 - rpn_class_loss: 0.0075 - rpn_bbox_loss: 0.5578 - mrcnn_class_los
s: 0.0813 - mrcnn_bbox_loss: 0.1656 - mrcnn_mask_loss: 0.2437 - val_l
oss: 2.0538 - val_rpn_class_loss: 0.0042 - val_rpn_bbox_loss: 1.3202
- val_mrcnn_class_loss: 0.0902 - val_mrcnn_bbox_loss: 0.3837 - val_mr
cnn_mask_loss: 0.2556
Epoch 0: Training batch 7 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.0004782969

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
```

```
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 102s 1s/step - loss: 1.289
0 - rpn_class_loss: 0.0052 - rpn_bbox_loss: 0.7337 - mrcnn_class_los
s: 0.0909 - mrcnn_bbox_loss: 0.1743 - mrcnn_mask_loss: 0.2850 - val_l
oss: 2.2413 - val_rpn_class_loss: 0.0051 - val_rpn_bbox_loss: 1.5917
- val_mrcnn_class_loss: 0.0846 - val_mrcnn_bbox_loss: 0.3105 - val_mr
cnn_mask_loss: 0.2493
Epoch 0: Training batch 8 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.00043046721

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
```

```
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 102s 1s/step - loss: 1.182
9 - rpn_class_loss: 0.0069 - rpn_bbox_loss: 0.7495 - mrcnn_class_los
s: 0.0728 - mrcnn_bbox_loss: 0.1570 - mrcnn_mask_loss: 0.1967 - val_l
oss: 1.8006 - val_rpn_class_loss: 0.0048 - val_rpn_bbox_loss: 1.0078
- val_mrcnn_class_loss: 0.1290 - val_mrcnn_bbox_loss: 0.4357 - val_mr
cnn_mask_loss: 0.2233
Epoch 0: Training batch 9 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.000387420489

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 103s 1s/step - loss: 0.968
1 - rpn_class_loss: 0.0048 - rpn_bbox_loss: 0.5819 - mrcnn_class_los
s: 0.0636 - mrcnn_bbox_loss: 0.1300 - mrcnn_mask_loss: 0.1878 - val_l
oss: 2.0970 - val_rpn_class_loss: 0.0079 - val_rpn_bbox_loss: 1.4279
- val_mrcnn_class_loss: 0.0921 - val_mrcnn_bbox_loss: 0.3069 - val_mr
cnn_mask_loss: 0.2621
Epoch 0: Training batch 10 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.0003486784401
```

```
Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 101s 1s/step - loss: 0.924
0 - rpn_class_loss: 0.0036 - rpn_bbox_loss: 0.5368 - mrcnn_class_los
s: 0.0471 - mrcnn_bbox_loss: 0.1306 - mrcnn_mask_loss: 0.2059 - val_l
oss: 1.1579 - val_rpn_class_loss: 0.0046 - val_rpn_bbox_loss: 0.7345
- val_mrcnn_class_loss: 0.0465 - val_mrcnn_bbox_loss: 0.1494 - val_mr
cnn_mask_loss: 0.2228
Epoch 0: Training batch 11 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.00031381059609000004

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 103s 1s/step - loss: 1.127
8 - rpn_class_loss: 0.0053 - rpn_bbox_loss: 0.7540 - mrcnn_class_los
s: 0.0651 - mrcnn_bbox_loss: 0.1647 - mrcnn_mask_loss: 0.1387 - val_l
oss: 2.2728 - val_rpn_class_loss: 0.0063 - val_rpn_bbox_loss: 1.4662
- val_mrcnn_class_loss: 0.1221 - val_mrcnn_bbox_loss: 0.4408 - val_mr
cnn_mask_loss: 0.2373
Epoch 0: Training batch 12 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.00028242953648100003

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
```

```
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 104s 1s/step - loss: 0.671
7 - rpn_class_loss: 0.0038 - rpn_bbox_loss: 0.3141 - mrcnn_class_los
s: 0.0454 - mrcnn_bbox_loss: 0.0984 - mrcnn_mask_loss: 0.2100 - val_l
oss: 1.6837 - val_rpn_class_loss: 0.0053 - val_rpn_bbox_loss: 1.0427
- val_mrcnn_class_loss: 0.1356 - val_mrcnn_bbox_loss: 0.3423 - val_mr
cnn_mask_loss: 0.1578
Epoch 0: Training batch 13 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.00025418658283290005

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 102s 1s/step - loss: 0.888
2 - rpn_class_loss: 0.0047 - rpn_bbox_loss: 0.4392 - mrcnn_class_los
s: 0.0585 - mrcnn_bbox_loss: 0.1537 - mrcnn_mask_loss: 0.2322 - val_l
oss: 1.4106 - val_rpn_class_loss: 0.0023 - val_rpn_bbox_loss: 0.9446
- val_mrcnn_class_loss: 0.0393 - val_mrcnn_bbox_loss: 0.2372 - val_mr
cnn_mask_loss: 0.1873
Epoch 0: Training batch 14 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.00022876792454961005
```

```
Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 103s 1s/step - loss: 0.798
2 - rpn_class_loss: 0.0040 - rpn_bbox_loss: 0.4177 - mrcnn_class_los
s: 0.0479 - mrcnn_bbox_loss: 0.1370 - mrcnn_mask_loss: 0.1915 - val_l
oss: 1.6841 - val_rpn_class_loss: 0.0057 - val_rpn_bbox_loss: 1.1825
- val_mrcnn_class_loss: 0.1030 - val_mrcnn_bbox_loss: 0.2261 - val_mr
cnn_mask_loss: 0.1668
Epoch 0: Training batch 15 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.00020589113209464906

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 104s 1s/step - loss: 0.850
7 - rpn_class_loss: 0.0049 - rpn_bbox_loss: 0.4889 - mrcnn_class_los
s: 0.0535 - mrcnn_bbox_loss: 0.1215 - mrcnn_mask_loss: 0.1818 - val_l
oss: 1.3925 - val_rpn_class_loss: 0.0031 - val_rpn_bbox_loss: 0.9574
- val_mrcnn_class_loss: 0.0445 - val_mrcnn_bbox_loss: 0.1888 - val_mr
cnn_mask_loss: 0.1987
Epoch 0: Training batch 16 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.00018530201888518417

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
```

```
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 104s 1s/step - loss: 1.030
2 - rpn_class_loss: 0.0044 - rpn_bbox_loss: 0.6393 - mrcnn_class_los
s: 0.0537 - mrcnn_bbox_loss: 0.1373 - mrcnn_mask_loss: 0.1955 - val_l
oss: 1.6983 - val_rpn_class_loss: 0.0034 - val_rpn_bbox_loss: 1.0574
- val_mrcnn_class_loss: 0.0905 - val_mrcnn_bbox_loss: 0.2787 - val_mr
cnn_mask_loss: 0.2684
Epoch 0: Training batch 17 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.00016677181699666576

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 104s 1s/step - loss: 0.950
9 - rpn_class_loss: 0.0039 - rpn_bbox_loss: 0.5277 - mrcnn_class_los
s: 0.0452 - mrcnn_bbox_loss: 0.1260 - mrcnn_mask_loss: 0.2481 - val_l
oss: 1.3124 - val_rpn_class_loss: 0.0034 - val_rpn_bbox_loss: 0.7526
- val_mrcnn_class_loss: 0.0516 - val_mrcnn_bbox_loss: 0.2609 - val_mr
cnn_mask_loss: 0.2438
Epoch 0: Training batch 18 out of 40
Re-starting from epoch 1
```

```
Starting at epoch 0. LR=0.0001500946352969992

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 103s 1s/step - loss: 1.170
0 - rpn_class_loss: 0.0054 - rpn_bbox_loss: 0.7688 - mrcnn_class_los
s: 0.0536 - mrcnn_bbox_loss: 0.1187 - mrcnn_mask_loss: 0.2235 - val_l
oss: 2.1357 - val_rpn_class_loss: 0.0061 - val_rpn_bbox_loss: 1.4862
- val_mrcnn_class_loss: 0.1555 - val_mrcnn_bbox_loss: 0.2789 - val_mr
cnn_mask_loss: 0.2090
Epoch 0: Training batch 19 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.0001350851717672993

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
```

```
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 104s 1s/step - loss: 0.768
3 - rpn_class_loss: 0.0045 - rpn_bbox_loss: 0.3980 - mrcnn_class_los
s: 0.0444 - mrcnn_bbox_loss: 0.1144 - mrcnn_mask_loss: 0.2069 - val_l
oss: 2.2610 - val_rpn_class_loss: 0.0044 - val_rpn_bbox_loss: 1.5530
- val_mrcnn_class_loss: 0.0882 - val_mrcnn_bbox_loss: 0.3971 - val_mr
cnn_mask_loss: 0.2182
Epoch 0: Training batch 20 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.00012157665459056936

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
```

```
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 103s 1s/step - loss: 1.053
8 - rpn_class_loss: 0.0046 - rpn_bbox_loss: 0.5862 - mrcnn_class_los
s: 0.0476 - mrcnn_bbox_loss: 0.1650 - mrcnn_mask_loss: 0.2505 - val_l
oss: 1.1941 - val_rpn_class_loss: 0.0026 - val_rpn_bbox_loss: 0.7229
- val_mrcnn_class_loss: 0.0360 - val_mrcnn_bbox_loss: 0.2442 - val_mr
cnn_mask_loss: 0.1884
Epoch 0: Training batch 21 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=0.00010941898913151243

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 105s 1s/step - loss: 0.885
3 - rpn_class_loss: 0.0032 - rpn_bbox_loss: 0.5354 - mrcnn_class_los
s: 0.0374 - mrcnn_bbox_loss: 0.1121 - mrcnn_mask_loss: 0.1973 - val_l
oss: 1.3984 - val_rpn_class_loss: 0.0052 - val_rpn_bbox_loss: 0.8502
- val_mrcnn_class_loss: 0.0857 - val_mrcnn_bbox_loss: 0.2623 - val_mr
cnn_mask_loss: 0.1949
Epoch 0: Training batch 22 out of 40
Re-starting from epoch 1
```

```
Starting at epoch 0. LR=9.847709021836118e-05

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 105s 1s/step - loss: 0.801
7 - rpn_class_loss: 0.0051 - rpn_bbox_loss: 0.3857 - mrcnn_class_los
s: 0.0433 - mrcnn_bbox_loss: 0.1242 - mrcnn_mask_loss: 0.2434 - val_l
oss: 2.1358 - val_rpn_class_loss: 0.0071 - val_rpn_bbox_loss: 1.4229
- val_mrcnn_class_loss: 0.1670 - val_mrcnn_bbox_loss: 0.3696 - val_mr
cnn_mask_loss: 0.1693
Epoch 0: Training batch 23 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=8.862938119652506e-05

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
```

```
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 104s 1s/step - loss: 1.188
9 - rpn_class_loss: 0.0048 - rpn_bbox_loss: 0.7858 - mrcnn_class_los
s: 0.0406 - mrcnn_bbox_loss: 0.1369 - mrcnn_mask_loss: 0.2208 - val_l
oss: 1.2412 - val_rpn_class_loss: 9.9534e-04 - val_rpn_bbox_loss: 0.8
403 - val_mrcnn_class_loss: 0.0347 - val_mrcnn_bbox_loss: 0.1667 - va
l_mrcnn_mask_loss: 0.1985
Epoch 0: Training batch 24 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=7.976644307687256e-05

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
```

```
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 104s 1s/step - loss: 1.148
8 - rpn_class_loss: 0.0055 - rpn_bbox_loss: 0.7121 - mrcnn_class_los
s: 0.0488 - mrcnn_bbox_loss: 0.1810 - mrcnn_mask_loss: 0.2014 - val_l
oss: 0.9611 - val_rpn_class_loss: 0.0021 - val_rpn_bbox_loss: 0.4818
- val_mrcnn_class_loss: 0.0428 - val_mrcnn_bbox_loss: 0.2446 - val_mr
cnn_mask_loss: 0.1898
Epoch 0: Training batch 25 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=7.17897987691853e-05

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 109s 1s/step - loss: 1.307
5 - rpn_class_loss: 0.0054 - rpn_bbox_loss: 0.8429 - mrcnn_class_los
s: 0.0614 - mrcnn_bbox_loss: 0.2055 - mrcnn_mask_loss: 0.1921 - val_l
oss: 1.0380 - val_rpn_class_loss: 0.0031 - val_rpn_bbox_loss: 0.4595
- val_mrcnn_class_loss: 0.0794 - val_mrcnn_bbox_loss: 0.2228 - val_mr
cnn_mask_loss: 0.2732
Epoch 0: Training batch 26 out of 40
```

```
Re-starting from epoch 1

Starting at epoch 0. LR=6.461081889226677e-05

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
100/100 [==============================] - 112s 1s/step - loss: 0.869
4 - rpn_class_loss: 0.0044 - rpn_bbox_loss: 0.4330 - mrcnn_class_los
s: 0.0482 - mrcnn_bbox_loss: 0.1438 - mrcnn_mask_loss: 0.2400 - val_l
oss: 2.0533 - val_rpn_class_loss: 0.0051 - val_rpn_bbox_loss: 1.4719
- val_mrcnn_class_loss: 0.1173 - val_mrcnn_bbox_loss: 0.2459 - val_mr
cnn_mask_loss: 0.2132
Epoch 0: Training batch 27 out of 40
Re-starting from epoch 1

Starting at epoch 0. LR=5.81497370030401e-05

Checkpoint Path: /home/paperspace/Documents/PedNet/logs/ped20181117T1
835/mask_rcnn_ped_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
```

```
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
Epoch 1/1
  3/100 [..............................] - ETA: 5:37 - loss: 0.5065 -
rpn_class_loss: 0.0016 - rpn_bbox_loss: 0.2603 - mrcnn_class_loss: 0.
0039 - mrcnn_bbox_loss: 0.0558 - mrcnn_mask_loss: 0.1849
```

```
---------------------------------------------------------------------
------
ResourceExhaustedError                    Traceback (most recent call
 last)
~/anaconda3/lib/python3.6/site-packages/tensorflow/python/client/sess
ion.py in _do_call(self, fn, *args)
   1326     try:
-> 1327       return fn(*args)
   1328     except errors.OpError as e:

~/anaconda3/lib/python3.6/site-packages/tensorflow/python/client/sess
ion.py in _run_fn(feed_dict, fetch_list, target_list, options, run_me
tadata)
   1311       return self._call_tf_sessionrun(
-> 1312           options, feed_dict, fetch_list, target_list, run_me
tadata)
   1313 

~/anaconda3/lib/python3.6/site-packages/tensorflow/python/client/sess
ion.py in _call_tf_sessionrun(self, options, feed_dict, fetch_list, t
arget_list, run_metadata)
   1419             self._session, options, feed_dict, fetch_list, ta
rget_list,
-> 1420             status, run_metadata)
   1421 

~/anaconda3/lib/python3.6/site-packages/tensorflow/python/framework/e
rrors_impl.py in __exit__(self, type_arg, value_arg, traceback_arg)
    515             compat.as_text(c_api.TF_Message(self.status.statu
s)),
--> 516             c_api.TF_GetCode(self.status.status))
    517     # Delete the underlying status object from memory otherwi
se it stays alive

ResourceExhaustedError: OOM when allocating tensor with shape[2,512,1
60,160] and type float on /job:localhost/replica:0/task:0/device:GPU:
0 by allocator GPU_0_bfc
         [[Node: training_27/SGD/gradients/rpn_model/rpn_bbox_pred/co
nvolution_grad/Conv2DBackpropInput = Conv2DBackpropInput[T=DT_FLOAT,
 _class=["loc:@rpn_model/rpn_bbox_pred/convolution"], data_format="NC
HW", dilations=[1, 1, 1, 1], padding="VALID", strides=[1, 1, 1, 1], u
se_cudnn_on_gpu=true, _device="/job:localhost/replica:0/task:0/devic
e:GPU:0"](training_27/SGD/gradients/rpn_model/rpn_bbox_pred/convoluti
on_grad/ShapeN, rpn_bbox_pred/kernel/read, training_27/SGD/gradients/
rpn_model/rpn_bbox_pred/convolution_grad/Conv2DBackpropInput-2-Transp
oseNHWCToNCHW-LayoutOptimizer)]]
Hint: If you want to see a list of allocated tensors when OOM happen
s, add report_tensor_allocations_upon_oom to RunOptions for current a
llocation info.

         [[Node: training_27/SGD/gradients/mrcnn_mask_conv1/convoluti
on_grad/Conv2DBackpropInput-0-0-TransposeNCHWToNHWC-LayoutOptimizer/_
41253 = _Recv[client_terminated=false, recv_device="/job:localhost/re
plica:0/task:0/device:CPU:0", send_device="/job:localhost/replica:0/t
ask:0/device:GPU:0", send_device_incarnation=1, tensor_name="edge_766
7_...tOptimizer", tensor_type=DT_FLOAT, _device="/job:localhost/repli
ca:0/task:0/device:CPU:0"]()]]
```

```
Hint: If you want to see a list of allocated tensors when OOM happen
s, add report_tensor_allocations_upon_oom to RunOptions for current a
llocation info.


During handling of the above exception, another exception occurred:

ResourceExhaustedError                    Traceback (most recent call
 last)
<ipython-input-10-be59f87163bb> in <module>()
     22                     learning_rate=lr,
     23                     epochs=1,
---> 24                     layers='heads')
     25 
     26         lr = lr * 0.90

~/Documents/PedNet/mrcnn/model.py in train(self, train_dataset, val_d
ataset, learning_rate, epochs, layers, augmentation)
   2312             max_queue_size=100,
   2313             workers=workers,
-> 2314             use_multiprocessing=False,
   2315         )
   2316         self.epoch = max(self.epoch, epochs)

~/anaconda3/lib/python3.6/site-packages/keras/legacy/interfaces.py in
 wrapper(*args, **kwargs)
     89                 warnings.warn('Update your `' + object_name +
     90                               '` call to the Keras 2 API: ' +
 signature, stacklevel=2)
---> 91             return func(*args, **kwargs)
     92         wrapper._original_function = func
     93         return wrapper

~/anaconda3/lib/python3.6/site-packages/keras/engine/training.py in f
it_generator(self, generator, steps_per_epoch, epochs, verbose, callb
acks, validation_data, validation_steps, class_weight, max_queue_siz
e, workers, use_multiprocessing, shuffle, initial_epoch)
   2222                     outs = self.train_on_batch(x, y,
   2223                                                sample_weight=
sample_weight,
-> 2224                                                class_weight=c
lass_weight)
   2225 
   2226                     if not isinstance(outs, list):

~/anaconda3/lib/python3.6/site-packages/keras/engine/training.py in t
rain_on_batch(self, x, y, sample_weight, class_weight)
   1881             ins = x + y + sample_weights
   1882         self._make_train_function()
-> 1883         outputs = self.train_function(ins)
   1884         if len(outputs) == 1:
   1885             return outputs[0]

~/anaconda3/lib/python3.6/site-packages/keras/backend/tensorflow_back
end.py in __call__(self, inputs)
   2476         session = get_session()
   2477         updated = session.run(fetches=fetches, feed_dict=feed
_dict,
-> 2478                                **self.session_kwargs)
   2479         return updated[:len(self.outputs)]
   2480 

~/anaconda3/lib/python3.6/site-packages/tensorflow/python/client/sess
ion.py in run(self, fetches, feed_dict, options, run_metadata)
    903     try:
    904       result = self._run(None, fetches, feed_dict, options_pt
r,
--> 905                          run_metadata_ptr)
    906       if run_metadata:
    907         proto_data = tf_session.TF_GetBuffer(run_metadata_ptr
)

~/anaconda3/lib/python3.6/site-packages/tensorflow/python/client/sess
ion.py in _run(self, handle, fetches, feed_dict, options, run_metadat
a)
   1138     if final_fetches or final_targets or (handle and feed_dic
t_tensor):
   1139       results = self._do_run(handle, final_targets, final_fet
ches,
-> 1140                              feed_dict_tensor, options, run_m
etadata)
   1141     else:
   1142       results = []

~/anaconda3/lib/python3.6/site-packages/tensorflow/python/client/sess
ion.py in _do_run(self, handle, target_list, fetch_list, feed_dict, o
ptions, run_metadata)
   1319     if handle is None:
   1320       return self._do_call(_run_fn, feeds, fetches, targets,
 options,
-> 1321                            run_metadata)
   1322     else:
   1323       return self._do_call(_prun_fn, handle, feeds, fetches)

~/anaconda3/lib/python3.6/site-packages/tensorflow/python/client/sess
ion.py in _do_call(self, fn, *args)
   1338         except KeyError:
   1339           pass
-> 1340       raise type(e)(node_def, op, message)
   1341 
   1342   def _extend_graph(self):

ResourceExhaustedError: OOM when allocating tensor with shape[2,512,1
60,160] and type float on /job:localhost/replica:0/task:0/device:GPU:
0 by allocator GPU_0_bfc
         [[Node: training_27/SGD/gradients/rpn_model/rpn_bbox_pred/co
nvolution_grad/Conv2DBackpropInput = Conv2DBackpropInput[T=DT_FLOAT,
 _class=["loc:@rpn_model/rpn_bbox_pred/convolution"], data_format="NC
HW", dilations=[1, 1, 1, 1], padding="VALID", strides=[1, 1, 1, 1], u
se_cudnn_on_gpu=true, _device="/job:localhost/replica:0/task:0/devic
e:GPU:0"](training_27/SGD/gradients/rpn_model/rpn_bbox_pred/convoluti
on_grad/ShapeN, rpn_bbox_pred/kernel/read, training_27/SGD/gradients/
rpn_model/rpn_bbox_pred/convolution_grad/Conv2DBackpropInput-2-Transp
oseNHWCToNCHW-LayoutOptimizer)]]
```

```
Hint: If you want to see a list of allocated tensors when OOM happen
s, add report_tensor_allocations_upon_oom to RunOptions for current a
llocation info.

         [[Node: training_27/SGD/gradients/mrcnn_mask_conv1/convoluti
on_grad/Conv2DBackpropInput-0-0-TransposeNCHWToNHWC-LayoutOptimizer/_
41253 = _Recv[client_terminated=false, recv_device="/job:localhost/re
plica:0/task:0/device:CPU:0", send_device="/job:localhost/replica:0/t
ask:0/device:GPU:0", send_device_incarnation=1, tensor_name="edge_766
7_...tOptimizer", tensor_type=DT_FLOAT, _device="/job:localhost/repli
ca:0/task:0/device:CPU:0"]()]]
Hint: If you want to see a list of allocated tensors when OOM happen
s, add report_tensor_allocations_upon_oom to RunOptions for current a
llocation info.


Caused by op 'training_27/SGD/gradients/rpn_model/rpn_bbox_pred/convo
lution_grad/Conv2DBackpropInput', defined at:
  File "/home/paperspace/anaconda3/lib/python3.6/runpy.py", line 193,
 in _run_module_as_main
    "__main__", mod_spec)
  File "/home/paperspace/anaconda3/lib/python3.6/runpy.py", line 85,
 in _run_code
    exec(code, run_globals)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/ipyker
nel_launcher.py", line 16, in <module>
    app.launch_new_instance()
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/traitl
ets/config/application.py", line 658, in launch_instance
    app.start()
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/ipyker
nel/kernelapp.py", line 486, in start
    self.io_loop.start()
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tornad
o/platform/asyncio.py", line 112, in start
    self.asyncio_loop.run_forever()
  File "/home/paperspace/anaconda3/lib/python3.6/asyncio/base_events.
py", line 421, in run_forever
    self._run_once()
  File "/home/paperspace/anaconda3/lib/python3.6/asyncio/base_events.
py", line 1426, in _run_once
    handle._run()
  File "/home/paperspace/anaconda3/lib/python3.6/asyncio/events.py",
 line 127, in _run
    self._callback(*self._args)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tornad
o/platform/asyncio.py", line 102, in _handle_events
    handler_func(fileobj, events)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tornad
o/stack_context.py", line 276, in null_wrapper
    return fn(*args, **kwargs)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/zmq/ev
entloop/zmqstream.py", line 450, in _handle_events
    self._handle_recv()
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/zmq/ev
entloop/zmqstream.py", line 480, in _handle_recv
    self._run_callback(callback, msg)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/zmq/ev
entloop/zmqstream.py", line 432, in _run_callback
    callback(*args, **kwargs)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tornad
o/stack_context.py", line 276, in null_wrapper
    return fn(*args, **kwargs)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/ipyker
nel/kernelbase.py", line 283, in dispatcher
    return self.dispatch_shell(stream, msg)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/ipyker
nel/kernelbase.py", line 233, in dispatch_shell
    handler(stream, idents, msg)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/ipyker
nel/kernelbase.py", line 399, in execute_request
    user_expressions, allow_stdin)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/ipyker
nel/ipkernel.py", line 208, in do_execute
    res = shell.run_cell(code, store_history=store_history, silent=si
lent)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/ipyker
nel/zmqshell.py", line 537, in run_cell
    return super(ZMQInteractiveShell, self).run_cell(*args, **kwargs)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/IPytho
n/core/interactiveshell.py", line 2662, in run_cell
    raw_cell, store_history, silent, shell_futures)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/IPytho
n/core/interactiveshell.py", line 2785, in _run_cell
    interactivity=interactivity, compiler=compiler, result=result)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/IPytho
n/core/interactiveshell.py", line 2903, in run_ast_nodes
    if self.run_code(code, result):
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/IPytho
n/core/interactiveshell.py", line 2963, in run_code
    exec(code_obj, self.user_global_ns, self.user_ns)
  File "<ipython-input-10-be59f87163bb>", line 24, in <module>
    layers='heads')
  File "/home/paperspace/Documents/PedNet/mrcnn/model.py", line 2314,
 in train
    use_multiprocessing=False,
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/keras/
legacy/interfaces.py", line 91, in wrapper
    return func(*args, **kwargs)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/keras/
engine/training.py", line 2080, in fit_generator
    self._make_train_function()
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/keras/
engine/training.py", line 990, in _make_train_function
    loss=self.total_loss)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/keras/
legacy/interfaces.py", line 91, in wrapper
    return func(*args, **kwargs)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/keras/
optimizers.py", line 173, in get_updates
    grads = self.get_gradients(loss, params)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/keras/
optimizers.py", line 78, in get_gradients
    grads = K.gradients(loss, params)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/keras/
backend/tensorflow_backend.py", line 2515, in gradients
    return tf.gradients(loss, variables, colocate_gradients_with_ops=
True)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tensor
flow/python/ops/gradients_impl.py", line 488, in gradients
    gate_gradients, aggregation_method, stop_gradients)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tensor
flow/python/ops/gradients_impl.py", line 625, in _GradientsHelper
    lambda: grad_fn(op, *out_grads))
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tensor
flow/python/ops/gradients_impl.py", line 379, in _MaybeCompile
    return grad_fn()  # Exit early
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tensor
flow/python/ops/gradients_impl.py", line 625, in <lambda>
    lambda: grad_fn(op, *out_grads))
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tensor
flow/python/ops/nn_grad.py", line 514, in _Conv2DGrad
    data_format=data_format),
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tensor
flow/python/ops/gen_nn_ops.py", line 1224, in conv2d_backprop_input
    dilations=dilations, name=name)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tensor
flow/python/framework/op_def_library.py", line 787, in _apply_op_help
er
    op_def=op_def)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tensor
flow/python/framework/ops.py", line 3290, in create_op
    op_def=op_def)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tensor
flow/python/framework/ops.py", line 1654, in __init__
    self._traceback = self._graph._extract_stack()  # pylint: disable
=protected-access

...which was originally created as op 'rpn_model/rpn_bbox_pred/convol
ution', defined at:
  File "/home/paperspace/anaconda3/lib/python3.6/runpy.py", line 193,
 in _run_module_as_main
    "__main__", mod_spec)
[elided 22 identical lines from previous traceback]
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/IPytho
n/core/interactiveshell.py", line 2963, in run_code
    exec(code_obj, self.user_global_ns, self.user_ns)
  File "<ipython-input-7-7928c4edfc77>", line 3, in <module>
    model_dir=MODEL_DIR)
  File "/home/paperspace/Documents/PedNet/mrcnn/model.py", line 1794,
 in __init__
    self.keras_model = self.build(mode=mode, config=config)
  File "/home/paperspace/Documents/PedNet/mrcnn/model.py", line 1901,
 in build
    layer_outputs.append(rpn([p]))
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/keras/
engine/topology.py", line 619, in __call__
    output = self.call(inputs, **kwargs)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/keras/
engine/topology.py", line 2085, in call
    output_tensors, _, _ = self.run_internal_graph(inputs, masks)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/keras/
engine/topology.py", line 2236, in run_internal_graph
    output_tensors = _to_list(layer.call(computed_tensor, **kwargs))
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/keras/
layers/convolutional.py", line 168, in call
    dilation_rate=self.dilation_rate)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/keras/
backend/tensorflow_backend.py", line 3335, in conv2d
    data_format=tf_data_format)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tensor
flow/python/ops/nn_ops.py", line 782, in convolution
    return op(input, filter)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tensor
flow/python/ops/nn_ops.py", line 870, in __call__
    return self.conv_op(inp, filter)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tensor
flow/python/ops/nn_ops.py", line 522, in __call__
    return self.call(inp, filter)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tensor
flow/python/ops/nn_ops.py", line 206, in __call__
    name=self.name)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tensor
flow/python/ops/gen_nn_ops.py", line 953, in conv2d
    data_format=data_format, dilations=dilations, name=name)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tensor
flow/python/framework/op_def_library.py", line 787, in _apply_op_help
er
    op_def=op_def)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tensor
flow/python/framework/ops.py", line 3290, in create_op
    op_def=op_def)
  File "/home/paperspace/anaconda3/lib/python3.6/site-packages/tensor
flow/python/framework/ops.py", line 1654, in __init__
    self._traceback = self._graph._extract_stack()  # pylint: disable
=protected-access

ResourceExhaustedError (see above for traceback): OOM when allocating
 tensor with shape[2,512,160,160] and type float on /job:localhost/re
plica:0/task:0/device:GPU:0 by allocator GPU_0_bfc
         [[Node: training_27/SGD/gradients/rpn_model/rpn_bbox_pred/co
nvolution_grad/Conv2DBackpropInput = Conv2DBackpropInput[T=DT_FLOAT,
 _class=["loc:@rpn_model/rpn_bbox_pred/convolution"], data_format="NC
HW", dilations=[1, 1, 1, 1], padding="VALID", strides=[1, 1, 1, 1], u
se_cudnn_on_gpu=true, _device="/job:localhost/replica:0/task:0/devic
e:GPU:0"](training_27/SGD/gradients/rpn_model/rpn_bbox_pred/convoluti
on_grad/ShapeN, rpn_bbox_pred/kernel/read, training_27/SGD/gradients/
rpn_model/rpn_bbox_pred/convolution_grad/Conv2DBackpropInput-2-Transp
oseNHWCToNCHW-LayoutOptimizer)]]
Hint: If you want to see a list of allocated tensors when OOM happen
s, add report_tensor_allocations_upon_oom to RunOptions for current a
llocation info.

         [[Node: training_27/SGD/gradients/mrcnn_mask_conv1/convoluti
on_grad/Conv2DBackpropInput-0-0-TransposeNCHWToNHWC-LayoutOptimizer/_
41253 = _Recv[client_terminated=false, recv_device="/job:localhost/re
plica:0/task:0/device:CPU:0", send_device="/job:localhost/replica:0/t
ask:0/device:GPU:0", send_device_incarnation=1, tensor_name="edge_766
```

```
7_...tOptimizer", tensor_type=DT_FLOAT, _device="/job:localhost/repli
ca:0/task:0/device:CPU:0"]()]]
Hint: If you want to see a list of allocated tensors when OOM happen
s, add report_tensor_allocations_upon_oom to RunOptions for current a
llocation info.
```

In [ ]:

```
# Save weights
# Typically not needed because callbacks save after every epoch
# Uncomment to save manually
# model_path = os.path.join(MODEL_DIR, "mask_rcnn_shapes.h5")
# model.keras_model.save_weights(model_path)
```

Detection

In [12]:

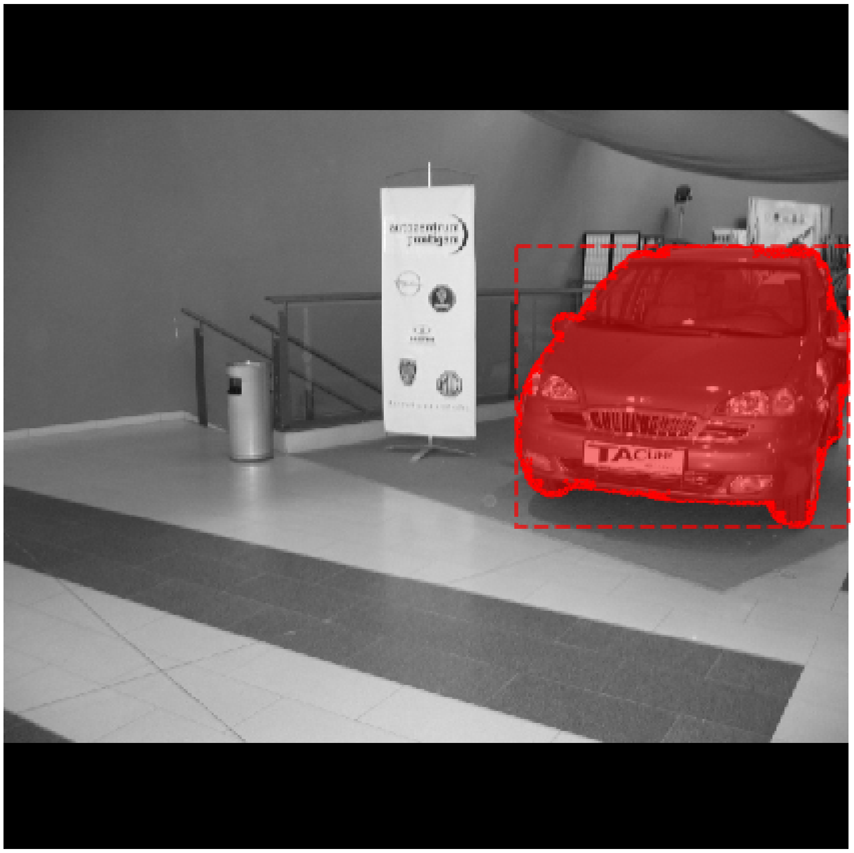
```
class InferenceConfig(PedConfig):
    GPU_COUNT = 1
    IMAGES_PER_GPU = 1

inference_config = InferenceConfig()

# Recreate the model in inference mode
model = modellib.MaskRCNN(mode="inference",
                          config=inference_config,
                          model_dir=MODEL_DIR)

# Get path to saved weights
# Either set a specific path or find last trained weights
# model_path = os.path.join(ROOT_DIR, ".h5 file name here")
# model_path = model.find_last()

# Load trained weights
print("Loading weights from ", "/home/paperspace/Documents/PedNet/log
s/ped20181117T1835/mask_rcnn_ped_0001.h5")
model.load_weights("/home/paperspace/Documents/PedNet/logs/ped2018111
7T1835/mask_rcnn_ped_0001.h5", by_name=True)
```

```
Loading weights from  /home/paperspace/Documents/PedNet/logs/ped20181
117T1835/mask_rcnn_ped_0001.h5
Re-starting from epoch 1
```

In [13]:

```
# Test on a random image
# image_id = random.choice(dataset_val.image_ids)
image_id = 2
original_image, image_meta, gt_class_id, gt_bbox, gt_mask =\
    modellib.load_image_gt(dataset_val, inference_config,
                           image_id, use_mini_mask=False)

log("original_image", original_image)
log("image_meta", image_meta)
log("gt_class_id", gt_class_id)
log("gt_bbox", gt_bbox)
log("gt_mask", gt_mask)

visualize.display_instances(original_image, gt_bbox, gt_mask, gt_clas
s_id,
                            dataset_train.class_names, figsize=(8, 8
))
```

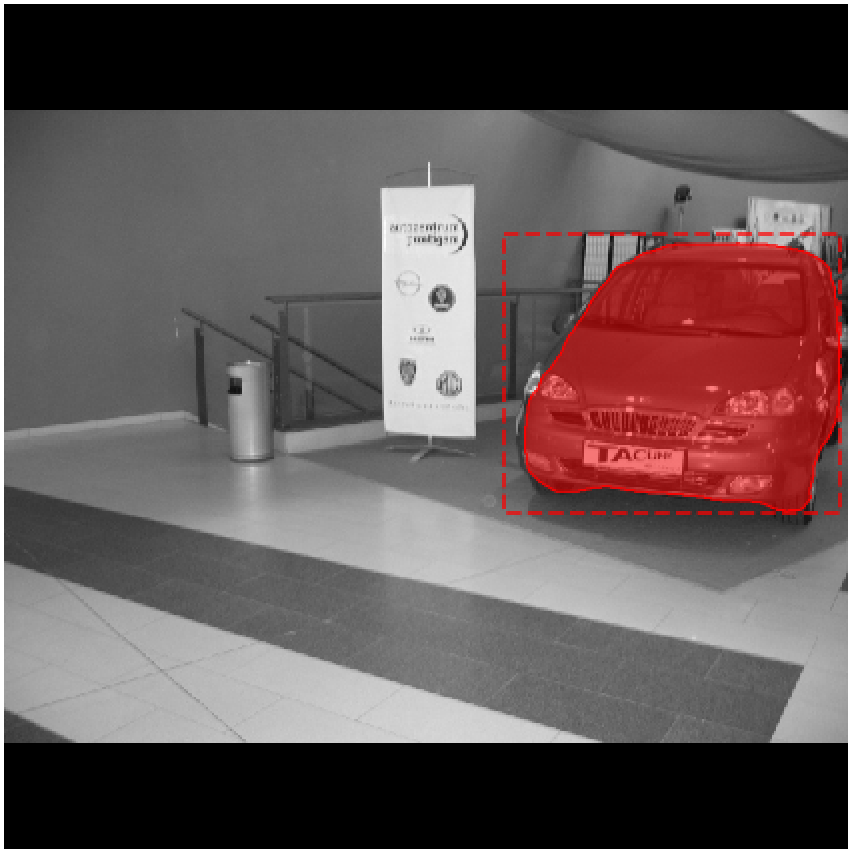
```
original_image           shape: (640, 640, 3)         min:    0.00000
  max:  255.00000  float64
image_meta               shape: (16,)                 min:    0.00000
  max:  640.00000  int64
gt_class_id              shape: (1, 1)                min:    2.00000
  max:    2.00000  int32
gt_bbox                  shape: (1, 4)                min:  183.00000
  max:  640.00000  int32
gt_mask                  shape: (640, 640, 1)         min:    0.00000
  max:    1.00000  float64
```



In [14]:

```
results = model.detect([original_image], verbose=1)

r = results[0]
visualize.display_instances(original_image, r['rois'], r['masks'], r[
'class_ids'],
                            dataset_val.class_names, r['scores'], ax=
get_ax())
```

```
Processing 1 images
image                    shape: (640, 640, 3)        min:    0.00000
  max:  255.00000  float64
molded_images            shape: (1, 640, 640, 3)     min: -123.70000
  max:  151.10000  float64
image_metas              shape: (1, 16)              min:    0.00000
  max:  640.00000  int64
anchors                  shape: (1, 102300, 4)       min:   -0.14164
  max:    1.04149  float32
```



Evaluation

In [16]:

```
# Compute VOC-Style mAP @ IoU=0.5
# Running on 10 images. Increase for better accuracy.
image_ids = np.random.choice(dataset_val.image_ids, dataset_val.N)
APs = []
for image_id in image_ids:
    # Load image and ground truth data
    image, image_meta, gt_class_id, gt_bbox, gt_mask =\
        modellib.load_image_gt(dataset_val, inference_config,
                               image_id, use_mini_mask=False)
    molded_images = np.expand_dims(modellib.mold_image(image, inferen
ce_config), 0)
    # Run object detection
    results = model.detect([image], verbose=0)
    r = results[0]
    # Compute AP
    AP, precisions, recalls, overlaps =\
        utils.compute_ap(gt_bbox, gt_class_id, gt_mask,
                         r["rois"], r["class_ids"], r["scores"], r['m
asks'])
    APs.append(AP)

print("mAP: ", np.mean(APs))
```

```
mAP:  0.9833333333333333
```

Data Visualization

In [5]:

```
config = PedConfig()

if config.NAME == 'ped':
    dataset = PedDataset()
    dataset.load_ped(istrain=True, st_index=0)


# Must call before using the dataset
dataset.prepare()

print("Image Count: {}".format(len(dataset.image_ids)))
print("Class Count: {}".format(dataset.num_classes))
for i, info in enumerate(dataset.class_info):
    print("{:3}. {:50}".format(i, info['name']))
```

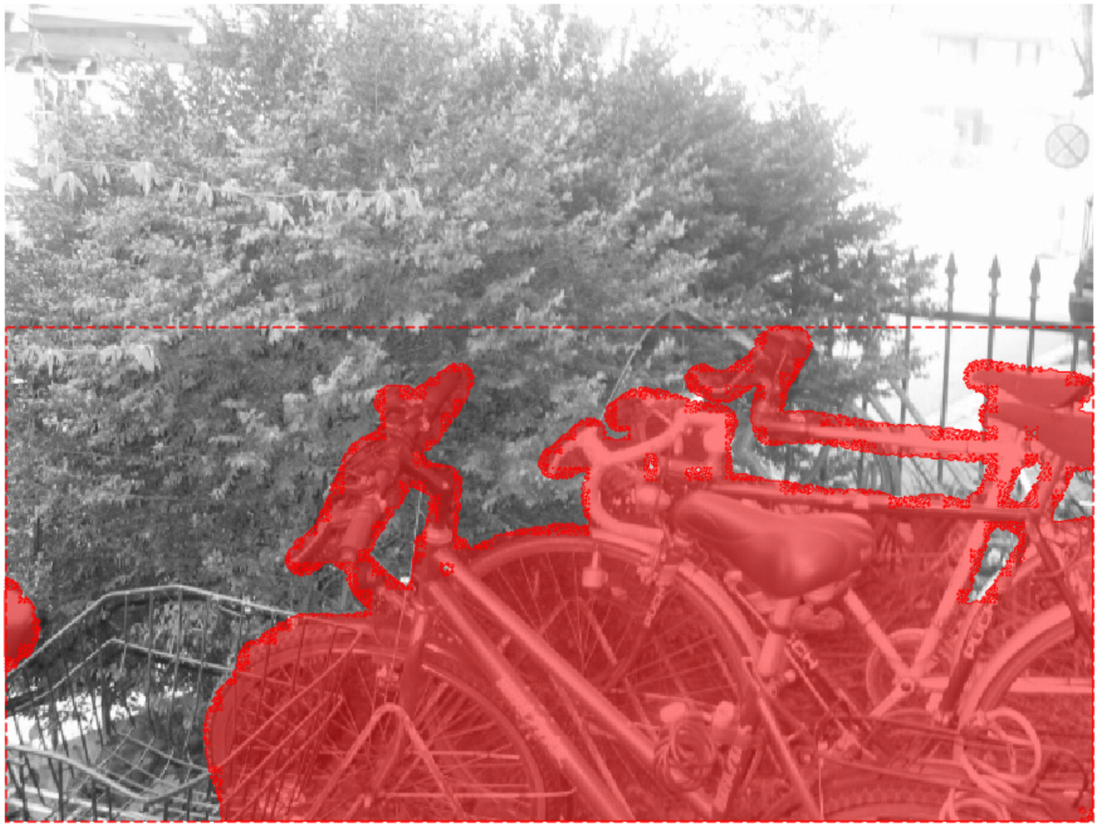
```
Image Count: 20
Class Count: 4
  0. BG
  1. bike
  2. car
  3. person
```

In [6]:
```
# Load and display random samples
image_ids = np.random.choice(dataset.image_ids, 4)
for image_id in image_ids:
    image = dataset.load_image(image_id)
    mask, class_ids = dataset.load_mask(image_id)
    visualize.display_top_masks(image, mask, class_ids, dataset.class
_names)
```

In [7]:

```
# Load random image and mask.
image_id = random.choice(dataset.image_ids)
image = dataset.load_image(image_id)
mask, class_ids = dataset.load_mask(image_id)
# Compute Bounding box
bbox = utils.extract_bboxes(mask)

# Display image and additional stats
print("image_id ", image_id, dataset.image_reference(image_id))
log("image", image)
log("mask", mask)
log("class_ids", class_ids)
log("bbox", bbox)
# Display image and instances
visualize.display_instances(image, bbox, mask, class_ids, dataset.cla
ss_names)
```

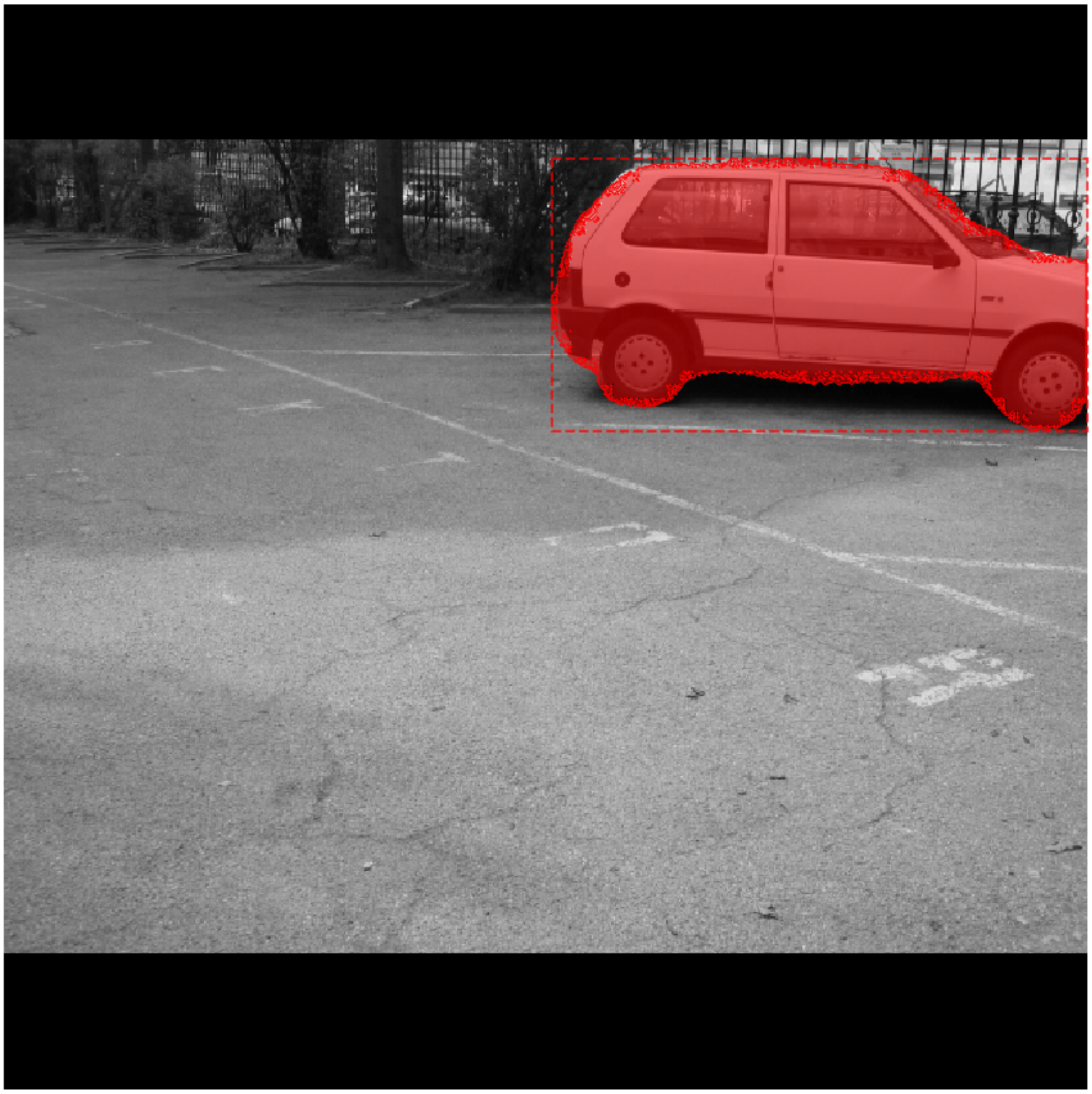
```
image_id  0 (480, 640)
image                    shape: (480, 640, 3)         min:   33.00000
  max:  255.00000  float64
mask                     shape: (480, 640, 1)         min:    0.00000
  max:    1.00000  float64
class_ids                shape: (1, 1)                min:    1.00000
  max:    1.00000  int32
bbox                     shape: (1, 4)                min:    0.00000
  max:  640.00000  int32
```



In [8]:

```
# Load random image and mask.
image_id = np.random.choice(dataset.image_ids, 1)[0]
image = dataset.load_image(image_id)
mask, class_ids = dataset.load_mask(image_id)
original_shape = image.shape
# Resize
image, window, scale, padding, _ = utils.resize_image(
    image,
    min_dim=config.IMAGE_MIN_DIM,
    max_dim=config.IMAGE_MAX_DIM,
    mode=config.IMAGE_RESIZE_MODE)
mask = utils.resize_mask(mask, scale, padding)
# Compute Bounding box
bbox = utils.extract_bboxes(mask)

# Display image and additional stats
print("image_id: ", image_id, dataset.image_reference(image_id))
print("Original shape: ", original_shape)
log("image", image)
log("mask", mask)
log("class_ids", class_ids)
log("bbox", bbox)
# Display image and instances
visualize.display_instances(image, bbox, mask, class_ids, dataset.cla
ss_names)
```

```
image_id:  11 (480, 640)
Original shape:  (480, 640, 3)
image                    shape: (640, 640, 3)         min:    0.00000
  max:  244.00000  float64
mask                     shape: (640, 640, 1)         min:    0.00000
  max:    1.00000  float64
class_ids                shape: (1, 1)                min:    2.00000
  max:    2.00000  int32
bbox                     shape: (1, 4)                min:   90.00000
  max:  640.00000  int32
```



In [9]:

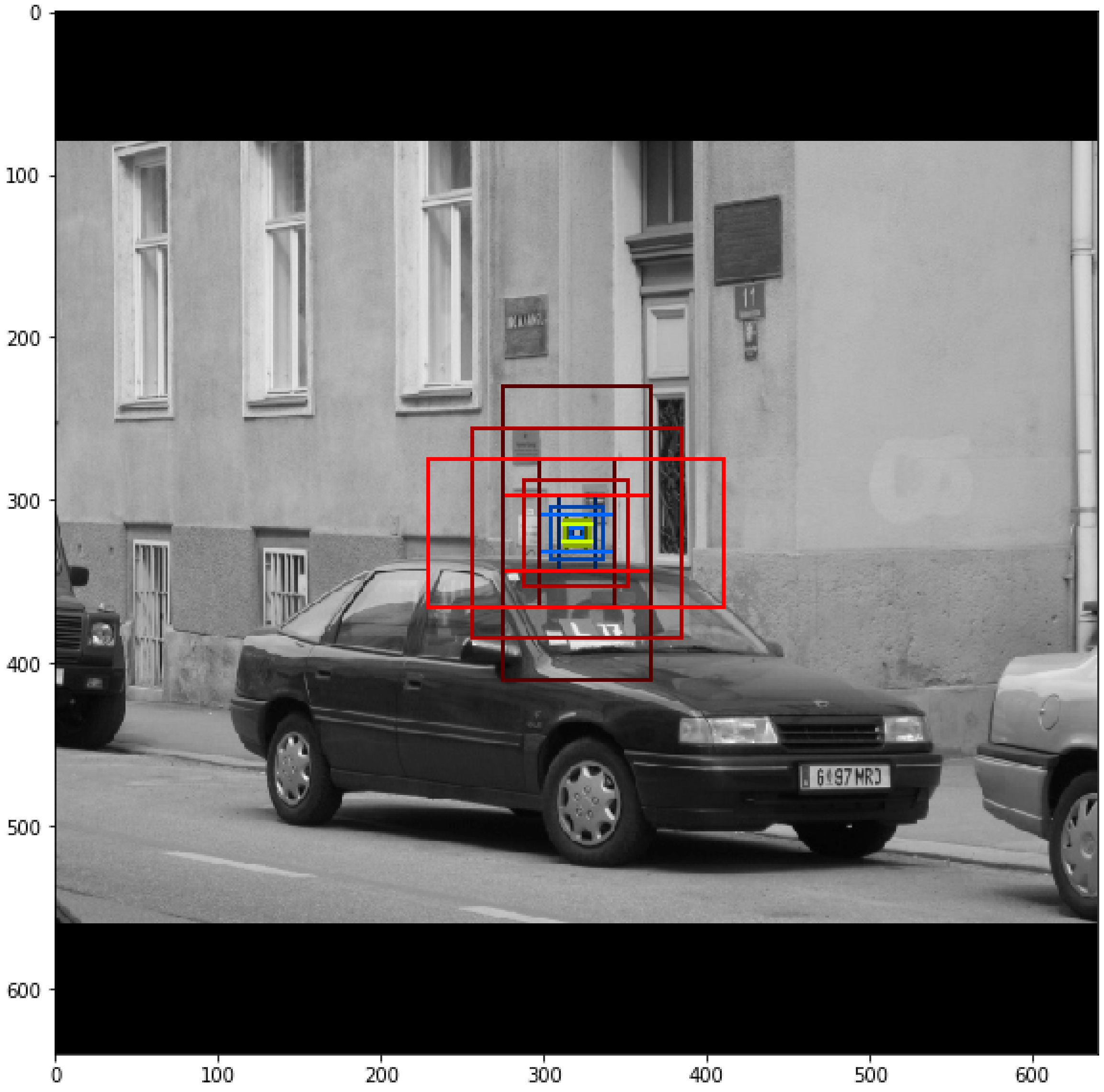
```
# Generate Anchors
backbone_shapes = modellib.compute_backbone_shapes(config, config.IMAGE_SHAPE)
anchors = utils.generate_pyramid_anchors(config.RPN_ANCHOR_SCALES,
                                          config.RPN_ANCHOR_RATIOS,
                                          backbone_shapes,
                                          config.BACKBONE_STRIDES,
                                          config.RPN_ANCHOR_STRIDE)

# Print summary of anchors
num_levels = len(backbone_shapes)
anchors_per_cell = len(config.RPN_ANCHOR_RATIOS)
print("Count: ", anchors.shape[0])
print("Scales: ", config.RPN_ANCHOR_SCALES)
print("ratios: ", config.RPN_ANCHOR_RATIOS)
print("Anchors per Cell: ", anchors_per_cell)
print("Levels: ", num_levels)
anchors_per_level = []
for l in range(num_levels):
    num_cells = backbone_shapes[l][0] * backbone_shapes[l][1]
    anchors_per_level.append(anchors_per_cell * num_cells // config.RPN_ANCHOR_STRIDE**2)
    print("Anchors in Level {}: {}".format(l, anchors_per_level[l]))
```

```
Count:  102300
Scales:  (8, 16, 32, 64, 128)
ratios:  [0.5, 1, 2]
Anchors per Cell:  3
Levels:  5
Anchors in Level 0: 76800
Anchors in Level 1: 19200
Anchors in Level 2: 4800
Anchors in Level 3: 1200
Anchors in Level 4: 300
```

In [10]:

```
import matplotlib.patches as patches
import matplotlib.lines as lines
from matplotlib.patches import Polygon

## Visualize anchors of one cell at the center of the feature map of
 a specific level

# Load and draw random image
image_id = np.random.choice(dataset.image_ids, 1)[0]
image, image_meta, _, _, _ = modellib.load_image_gt(dataset, config,
image_id)
fig, ax = plt.subplots(1, figsize=(10, 10))
ax.imshow(image.astype(np.uint8))
levels = len(backbone_shapes)

for level in range(levels):
    colors = visualize.random_colors(levels)
    # Compute the index of the anchors at the center of the image
    level_start = sum(anchors_per_level[:level]) # sum of anchors of
 previous levels
    level_anchors = anchors[level_start:level_start+anchors_per_level
[level]]
    print("Level {}. Anchors: {:6}  Feature map Shape: {}".format(lev
el, level_anchors.shape[0],
                                                                  bac
kbone_shapes[level]))
    center_cell = backbone_shapes[level] // 2
    center_cell_index = (center_cell[0] * backbone_shapes[level][1] +
 center_cell[1])
    level_center = center_cell_index * anchors_per_cell
    center_anchor = anchors_per_cell * (
        (center_cell[0] * backbone_shapes[level][1] / config.RPN_ANCH
OR_STRIDE**2) \
        + center_cell[1] / config.RPN_ANCHOR_STRIDE)
    level_center = int(center_anchor)

    # Draw anchors. Brightness show the order in the array, dark to b
right.
    for i, rect in enumerate(level_anchors[level_center:level_center+
anchors_per_cell]):
        y1, x1, y2, x2 = rect
        p = patches.Rectangle((x1, y1), x2-x1, y2-y1, linewidth=2, fa
cecolor='none',
                              edgecolor=(i+1)*np.array(colors[level])
 / anchors_per_cell)
        ax.add_patch(p)
```

```
Level 0. Anchors:  76800  Feature map Shape: [160 160]
Level 1. Anchors:  19200  Feature map Shape: [80 80]
Level 2. Anchors:   4800  Feature map Shape: [40 40]
Level 3. Anchors:   1200  Feature map Shape: [20 20]
Level 4. Anchors:    300  Feature map Shape: [10 10]
```



Create data generator

```
In [11]: random_rois = 2000
         g = modellib.DataGenerator(
             dataset, config, shuffle=True, random_rois=random_rois,
             batch_size=4,
             detection_targets=True)
         g
```

```
Out[11]: <mrcnn.model.DataGenerator at 0x7f973101bc88>
```

In [13]:

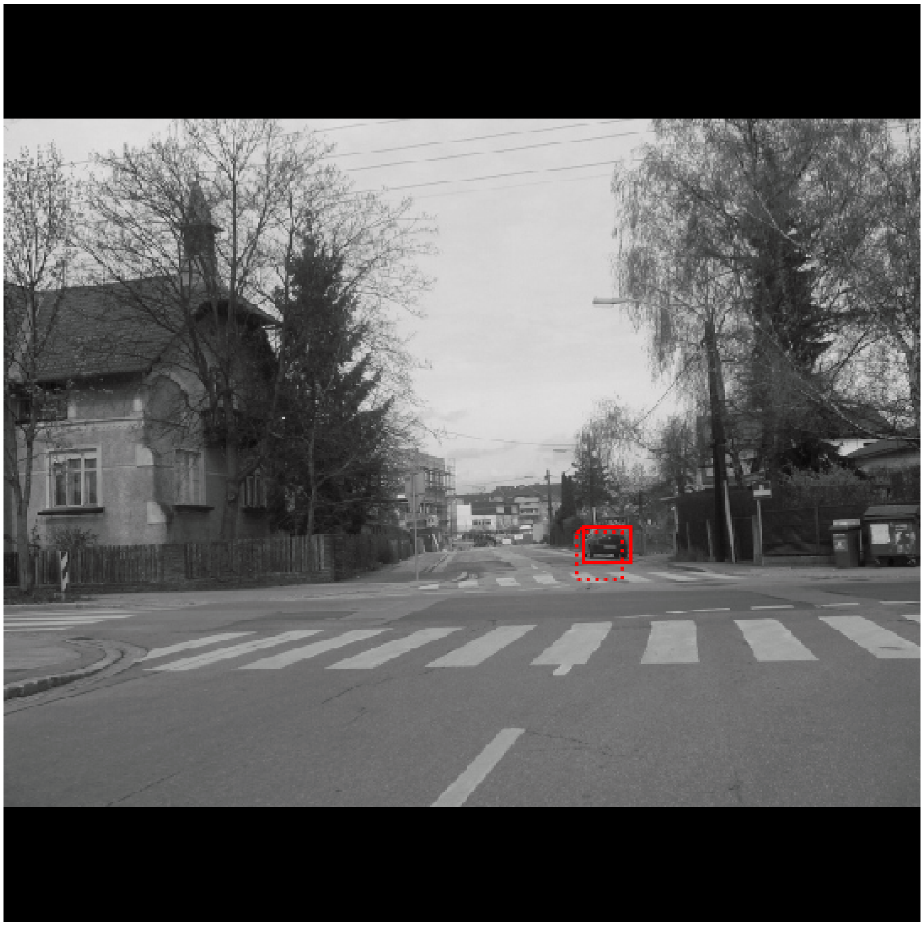
```
# Get Next Image
if random_rois:
    [normalized_images, image_meta, rpn_match, rpn_bbox, gt_class_ids
, gt_boxes, gt_masks, rpn_rois, rois], \
    [mrcnn_class_ids, mrcnn_bbox, mrcnn_mask] = g.__getitem__(0)

    log("rois", rois)
    log("mrcnn_class_ids", mrcnn_class_ids)
    log("mrcnn_bbox", mrcnn_bbox)
    log("mrcnn_mask", mrcnn_mask)
else:
    [normalized_images, image_meta, rpn_match, rpn_bbox, gt_boxes, gt
_masks], _ = next(g)

log("gt_class_ids", gt_class_ids)
log("gt_boxes", gt_boxes)
log("gt_masks", gt_masks)
log("rpn_match", rpn_match, )
log("rpn_bbox", rpn_bbox)
image_id = modellib.parse_image_meta(image_meta)["image_id"][0]
print("image_id: ", image_id, dataset.image_reference(image_id))

# Remove the last dim in mrcnn_class_ids. It's only added
# to satisfy Keras restriction on target shape.
mrcnn_class_ids = mrcnn_class_ids[:,:,0]
```

```
rois                     shape: (4, 32, 4)            min:    7.00000
  max:  639.00000  int32
mrcnn_class_ids          shape: (4, 32, 1, 1)         min:    0.00000
  max:    2.00000  int32
mrcnn_bbox               shape: (4, 32, 4, 4)         min:   -3.31551
  max:    2.68606  float32
mrcnn_mask               shape: (4, 32, 28, 28, 4)    min:    0.00000
  max:    1.00000  float32
gt_class_ids             shape: (4, 100)              min:    0.00000
  max:    2.00000  int32
gt_boxes                 shape: (4, 100, 4)           min:    0.00000
  max:  640.00000  int32
gt_masks                 shape: (4, 640, 640, 100)    min:    0.00000
  max:    1.00000  bool
rpn_match                shape: (4, 102300, 1)        min:   -1.00000
  max:    1.00000  int32
rpn_bbox                 shape: (4, 256, 4)           min:   -7.07107
  max:    7.41752  float64
image_id:  3 (480, 640)
```

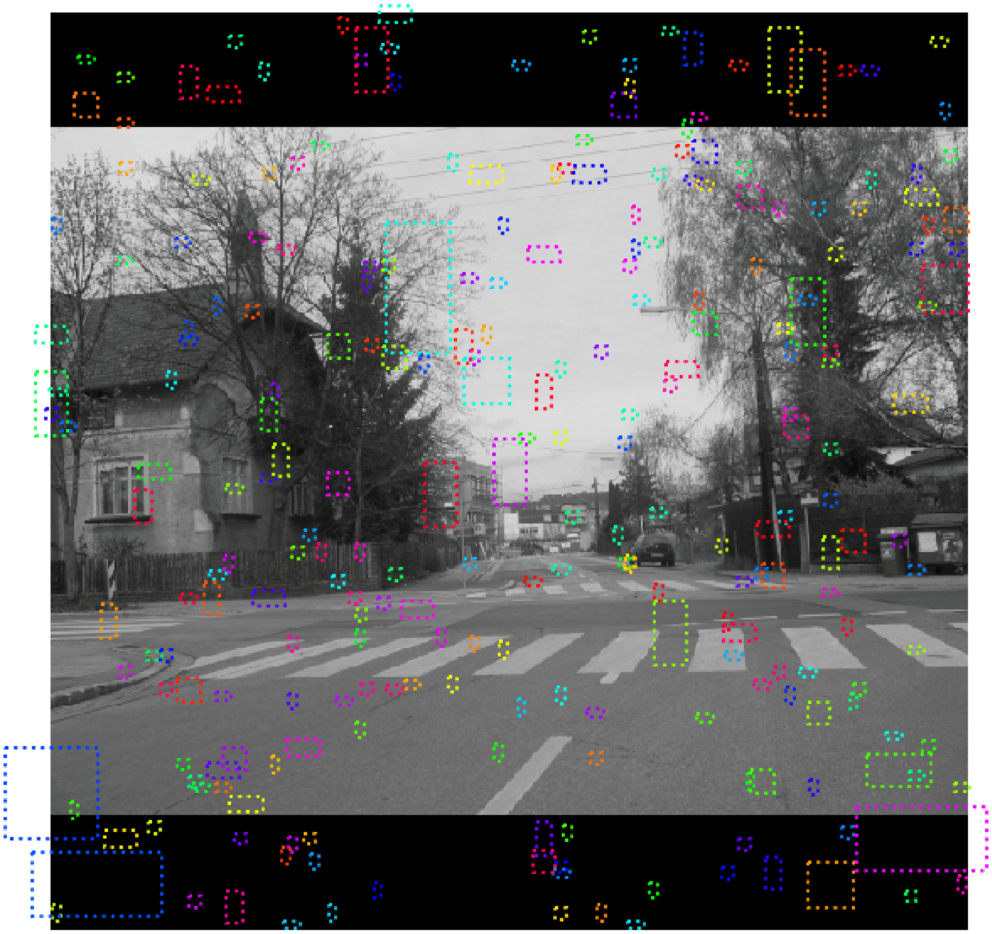
In [19]:

```
b = 0

# Restore original image (reverse normalization)
sample_image = modellib.unmold_image(normalized_images[b], config)

# Compute anchor shifts.
indices = np.where(rpn_match[b] == 1)[0]
refined_anchors = utils.apply_box_deltas(anchors[indices], rpn_bbox[b
, :len(indices)] * config.RPN_BBOX_STD_DEV)
log("anchors", anchors)
log("refined_anchors", refined_anchors)

# Get list of positive anchors
positive_anchor_ids = np.where(rpn_match[b] == 1)[0]
print("Positive anchors: {}".format(len(positive_anchor_ids)))
negative_anchor_ids = np.where(rpn_match[b] == -1)[0]
print("Negative anchors: {}".format(len(negative_anchor_ids)))
neutral_anchor_ids = np.where(rpn_match[b] == 0)[0]
print("Neutral anchors: {}".format(len(neutral_anchor_ids)))

# ROI breakdown by class
for c, n in zip(dataset.class_names, np.bincount(mrcnn_class_ids[b].f
latten())):
    if n:
        print("{:23}: {}".format(c[:20], n))

# Show positive anchors
visualize.draw_boxes(sample_image, boxes=anchors[positive_anchor_ids
],
                     refined_boxes=refined_anchors)
```

```
anchors                  shape: (102300, 4)          min:  -90.50967
  max:  666.50967  float64
refined_anchors          shape: (1, 4)               min:  364.00000
  max:  437.00000  float32
Positive anchors: 1
Negative anchors: 255
Neutral anchors: 102044
BG                     : 22
CAR                    : 10
```



In [20]:
```
visualize.draw_boxes(sample_image, boxes=anchors[negative_anchor_ids
])
```



In [21]:
```
visualize.draw_boxes(sample_image, boxes=anchors[np.random.choice(neu
tral_anchor_ids, 100)])
```