Appendix

(1). SingleAnnotation.py

```
import os
import xml.dom.minidom as minidom
import numpy as np
import cPickle
import subprocess

def readSingleAnno(filename):
   with open(filename) as f:
      data = f.read()

   import re
   objs = re.findall('\(\d+, \d+\)[\s\-]+\(\d+, \d+\)',
data)

   num_objs = len(objs)

   boxes = np.zeros((num_objs, 4), dtype=np.uint16)
   gt_classes = np.zeros((num_objs), dtype=np.int32)
   overlaps = np.zeros((num_objs, 2), dtype=np.float32)

         # "Seg" area here is just the box area
   seg_areas = np.zeros((num_objs), dtype=np.float32)

      # Load object bounding boxes into a data frame.
   for ix, obj in enumerate(objs):
      # Make pixel indexes 0-based
     coor = re.findall('\d+', obj)
```

```
        x1 = float(coor[0])
        y1 = float(coor[1])
        x2 = float(coor[2])
        y2 = float(coor[3])
        boxes[ix, :] = [x1, y1, x2, y2]
        seg_areas[ix] = (x2 - x1 + 1) * (y2 - y1 + 1)
    return boxes
```

(2). BatchAnnotations.py

```
import os
import numpy as np
import singleAnnotation as sann

#folder = "F:\INRIAPerson\Train\annotations"
#path = 'F:\INRIAPerson\Train\\annotations\\'
path = 'F:\INRIAPerson\Test\\annotations\\'
print path

imagelist = open('ImageList.txt', 'w')
annotations = open('annotations.txt', 'w')
for root, dirs, files in os.walk(path):
   for fn in files:
     #print (root + fn)
     imagelist.writelines(root + fn + '\n')
     boxes = sann.readSingleAnno(root + fn)
     #print len(boxes)
     annotations.writelines(root + fn + str())
                    #str(boxes[0][0]) + '\n')

   for i in range(0, len(boxes)):
      annotations.writelines(str(boxes[i][0]) + ' ' +
                             str(boxes[i][1]) + ' ' +
                             str(boxes[i][2]) + ' ' +
                             str(boxes[i][3]) + ' ')
   annotations.writelines('\n')

imagelist.close()
annotations.close()
```