Identification of Aedes Aegypti Eggs

Image classification using Keras. This project uses the Keras Fast RCNN model to identify Aedes Aegypti eggs in images taken from traps spread around Recife and Region.

The dataset was provided by Pickcells.

Authors: Tácio Nery / Gustavo Soares

Preprocessing

To perform Image Classification, the first thing to do is Label a sample of images. To do so, I used LabelImg. It will generate a XLM file with the coordinates of the objects marked. The sample images are on `data/training` .

After label the images, it's time to generate a CSV with the data includes on XML. The python script `utils/xml_to_csv.py` will do the job.

Visualizing Data

The below script demonstrates how the labels work. Simply load the CSV file and chose one image to display rectangles on the marked coordinates, which has the objects to detect.

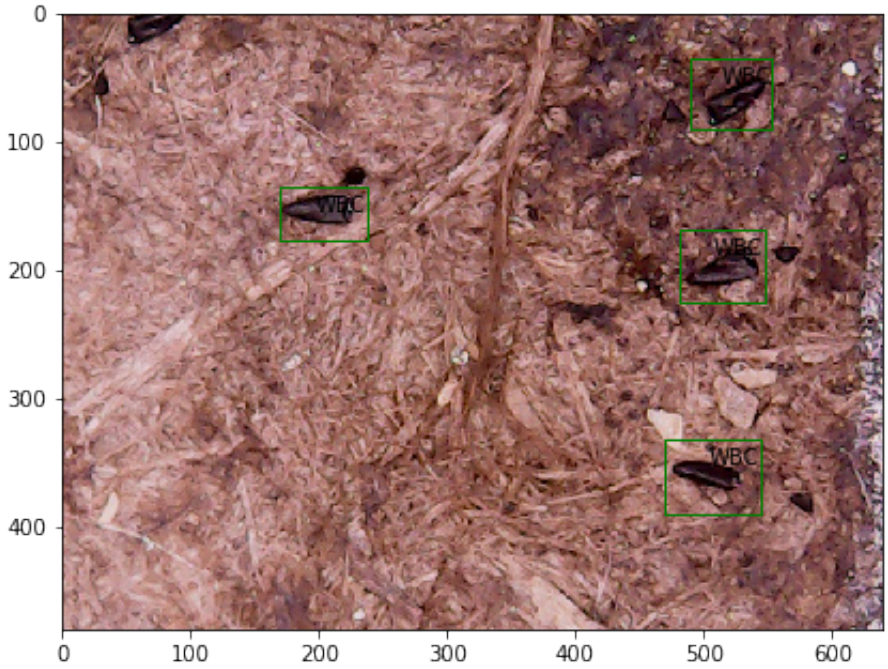
```
import pandas as pd
import matplotlib.pyplot as plt

%matplotlib inline

from matplotlib import patches

print("Pandas Version:", pd.__version__)
```

```
Pandas Version: 0.24.1
```

```
# Load the Tranining Set

train = pd.read_csv('aedes_labels.csv')
train.head(10)
```

	filename	width	height	class	xmin	ymin	xmax	ymax
0	0b57cceb-4d17-417c-a3c5-ffb0b62d8b59.jpg	640	480	full	169	134	237	177
1	0b57cceb-4d17-417c-a3c5-ffb0b62d8b59.jpg	640	480	full	481	168	547	225
2	0b57cceb-4d17-417c-a3c5-ffb0b62d8b59.jpg	640	480	full	470	332	544	389
3	0b57cceb-4d17-417c-a3c5-ffb0b62d8b59.jpg	640	480	full	489	34	553	89
4	0ca5874c-22db-4016-8d5c-5d2626dda567.jpg	640	480	full	576	221	607	280
5	0ca5874c-22db-4016-8d5c-5d2626dda567.jpg	640	480	full	250	319	311	360
6	0ca5874c-22db-4016-8d5c-5d2626dda567.jpg	640	480	full	212	335	268	370
7	0ca5874c-22db-4016-8d5c-5d2626dda567.jpg	640	480	full	173	271	223	316
8	0ca5874c-22db-4016-8d5c-5d2626dda567.jpg	640	480	full	288	213	344	237
9	0ca5874c-22db-4016-8d5c-5d2626dda567.jpg	640	480	full	279	233	355	258

```
# Mark the Aedes egg using one of the training set images

fig = plt.figure()
ax = fig.add_axes([0,0,1,1])

image = plt.imread('data/training/' + train.filename[0])
plt.imshow(image)

for _,row in train[train.filename == '0b57cceb-4d17-417c-a3c5-ffb0b62d8b59.jpg'].i
terrows():
    xmin = row.xmin
    xmax = row.xmax
    ymin = row.ymin
    ymax = row.ymax

    width = xmax - xmin
    height = ymax - ymin

    edgecolor = 'g'

    ax.annotate('WBC', xy=(xmax-40,ymin+20))

    rect = patches.Rectangle((xmin,ymin), width, height, edgecolor = edgecolor, fa
cecolor = 'none')

    ax.add_patch(rect)
```



Training

The Keras Faster RCNN needs a txt file with the same format the CSV file generated before.

```
# Convert the CSV file to TXT

data = pd.DataFrame()
data['format'] = train['filename']

for i in range(data.shape[0]):
    data['format'][i] = '../../data/training/' + data['format'][i]

for i in range(data.shape[0]):
    data['format'][i] = data['format'][i] + ',' + str(train['xmin'][i]) + ',' + st
r(train['ymin'][i]) \
        + ',' + str(train['xmax'][i]) + ',' + str(train['ymax'][i]) \
        + ',' + train['class'][i]

data.to_csv('annotate.txt', header = None, index = None, sep = ' ')
```

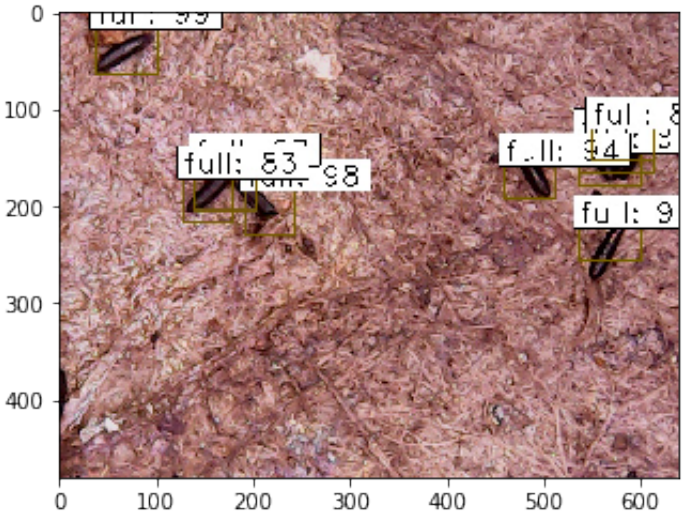
Now the model is has enough resources to be trained. The model can be found in `model/keras-frcnn` directory. To perform the training just run in terminal: `python train_frcnn.py -o simple -p ../../annotate.txt --num_epochs=50`. The model was edited to perform 50 epochs with 10 iterations each due to hardware limitations, it could take too long to train.

Once the traning finishes, the weights file will be generated into `model/kearas-frcnn/model_frcnn.hdf5`. Now the test can be performed with the command `python test_frcnn.py -p ../../data/testing`. The result of the testing will be saved in `data/output` directory.

Here's one sample of the output generated by the test.

```
test_output = plt.imread('data/output/13.jpg')
plt.imshow(test_output)
```

```
<matplotlib.image.AxesImage at 0x109bf92e8>
```


full: 83
0
100
200
300
400
0
100
200
300
400
500
600