

LoadNet

September 16, 2018

```
In [1]: # config
from lib.models.pose_resnet import get_pose_net
from lib.core.config import config
from lib.core.config import update_config
config.TEST.FLIP_TEST = True
config.TEST.MODEL_FILE = 'pose_resnet_50_256x256.pth.tar'
update_config('experiments/mpii/resnet50/256x256_d256x3_adam_lr1e-3.yaml')
model = get_pose_net(config, is_train=False)

import torch
import torchvision.transforms as transforms
mean = [0.485, 0.456, 0.406]
std = [0.229, 0.224, 0.225]
toTensor = transforms.Compose([transforms.ToTensor(),
                               transforms.Normalize(mean, std)])

def getpoint(mat):
    height, width = mat.shape
    mat = mat.reshape(-1)
    idx = np.argmax(mat)
    return idx % width, idx // width

In [2]: # load image and predict
import cv2
import numpy as np
img = cv2.imread('0.png', cv2.IMREAD_COLOR | cv2.IMREAD_IGNORE_ORIENTATION)
img = cv2.resize(img, (256, 256))
x = toTensor(img).unsqueeze(0)
with torch.no_grad():
    res = model.forward(x)
res = np.array(res.detach().squeeze())
print(img.shape)
print(res.shape)
```

(256, 256, 3)

(16, 64, 64)

```
In [4]: # plot
image = cv2.resize(img, (64, 64))
print(image.shape)
for mat in res:
    x, y = getpoint(mat)
    print(x, y)
    cv2.circle(image, (x, y), 2, (255, 0, 0), 2)
import matplotlib.pyplot as plt
plt.imshow(image)
```

```
(64, 64, 3)
```

```
10 46
```

```
8 37
```

```
27 29
```

```
13 37
```

```
33 7
```

```
30 7
```

```
25 18
```

```
17 31
```

```
31 22
```

```
29 21
```

```
15 32
```

```
12 51
```

```
23 15
```

```
36 18
```

```
13 40
```

```
12 41
```

```
Out[4]: <matplotlib.image.AxesImage at 0x7f14625c1160>
```

