

年轻即出发 ,

不努力 , 拿什么说明天

目录视图

摘要视图

RSS 订阅

## 个人资料



watersink



访问 : 231136次

积分 : 3013

等级 : 

排名 : 第12198名

原创 : 71篇 转载 : 1篇

译文 : 1篇 评论 : 576条

## 文章搜索

异步赠书 : 9月重磅新书升级 , 本本经典

程序员9月书讯

每周荐书 : ES6、虚拟现实、物联网 ( 评论送书 )

## MTCNN ( Multi-task convolutional neural networks ) 人脸对齐

2016-09-27 15:23

15731人阅读

评论(0)

分类 : 人脸对齐 ( 8 )

版权声明 : 本文为博主原创文章 , 未经博主允许不得转载。

该MTCNN算法出自深圳先进技术研究院 , 乔宇老师组 , 是今年2016的ECCV。(至少我知道的今年已经一篇eccv了)。

进入正题

理论基础 :

关闭

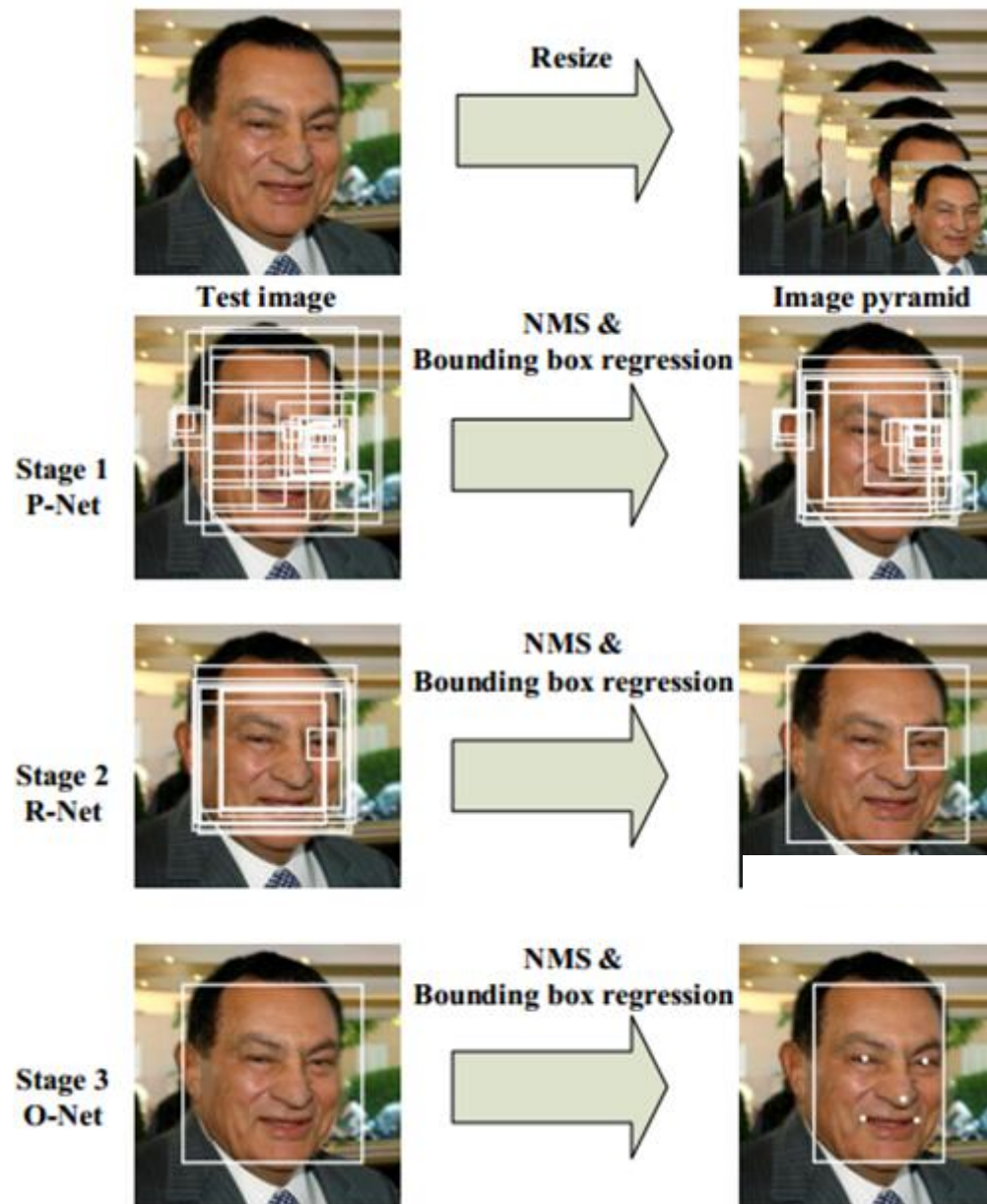
## 文章分类

人脸检测 (8)  
行人检测 (3)  
图像数据库 (3)  
机器学习 (18)  
深度学习 (38)  
人脸对齐 (9)  
开发工具 (1)  
人脸识别 (4)  
CUDA加速 (1)  
人群密度估计 (2)  
姿态估计 (2)  
车辆识别 (1)  
人脸增强 (1)  
行人重识别 (1)  
Torch7 (1)  
caffe (2)  
TensorFlow (1)  
mxnet (1)

## 文章存档

2017年09月 (4)  
2017年08月 (5)  
2017年07月 (3)  
2017年06月 (3)  
2017年05月 (2)

展开



正如下图所示，该MTCNN由3个网络结构组成（P-Net,R-Net,O-Net）。

Proposal Network (P-Net)：该网络结构主要获得了人脸区域的候选窗口和边界框的回归向量。并用该边界框做回归，对候选窗口进行校准，然后通过非极大值抑制（NMS）来合并高度重叠的候选框。

关闭

## 阅读排行

<a href="#">windows下的python+ op</a>	(24771)
<a href="#">Windows下caffe安装详解</a>	(22633)
<a href="#">MTCNN ( Multi-task con</a>	(15721)
<a href="#">YOLO v2之总结篇 ( linu</a>	(13336)
<a href="#">SeetaFace大总结</a>	(11742)
<a href="#">图像数据库</a>	(10329)
<a href="#">SSD(Single Shot MultiBc</a>	(9812)
<a href="#">人脸识别之caffe-face</a>	(7254)
<a href="#">DPM ( Deformable Part</a>	(7021)
<a href="#">windows下让自己的程序</a>	(5707)

## 评论排行

<a href="#">SeetaFace大总结</a>	(110)
<a href="#">Windows下caffe安装详解</a>	(67)
<a href="#">人脸识别之caffe-face</a>	(52)
<a href="#">YOLO v2之总结篇 ( linu</a>	(49)
<a href="#">windows下让自己的程序</a>	(47)
<a href="#">人群密度估计之MCNN</a>	(39)
<a href="#">LeNet识别自己的手写数:</a>	(27)
<a href="#">MTCNN ( Multi-task con</a>	(18)
<a href="#">人脸识别之SphereFace</a>	(18)
<a href="#">YOLO v1之总结篇 ( linu</a>	(15)

## 推荐文章

Refine Network (R-Net)：该网络结构还是通过边界框回归和NMS来去掉那些false-positive区域。

只是由于该网络结构和P-Net网络结构有差异，多了一个全连接层，所以会取得更好的抑制false-positive的作用。

Output Network (O-Net)：该层比R-Net层又多了一层卷基层，所以处理的结果会更加精细。作用和R-Net层作用一样。但是该层对人脸区域进行了更多的监督，同时还会输出5个地标（landmark）。

详细的网络结构如下图所示：

关闭

- \* CSDN新版博客feed流内测用户征集令
- \* Android检查更新下载安装
- \* 动手打造史上最简单的Recycleview 侧滑菜单
- \* TCP网络通讯如何解决分包粘包问题
- \* SDCC 2017之大数据技术实战线上峰会
- \* 快速集成一个视频直播功能

## 最新评论

### 人群密度估计之MCNN

sdsfby: 楼主你好, 相请问下文章里的shanghaiAB数据集的gt图像是怎么做的, 貌似上面的代码是针对于的ma...

### YOLO v1之总结篇 ( linux+windo

jiangqiangguo4180: 博主您好, 为什么我训练成功后, 输入任意图片后, 没有显示检测结果, 而显示的是"pottedplant:...

### 人群密度估计之MCNN

芝麻麻麻: 感谢博主细致的讲些, 受益匪浅。caffe新手在理解密度图标签转LMDB格式的时候遇到问题, 希望博主给...

### windows下让自己的程序调用cafi

watersink: @LZWXAZ:在预处理里面加个, CPU\_ONLY, 试试

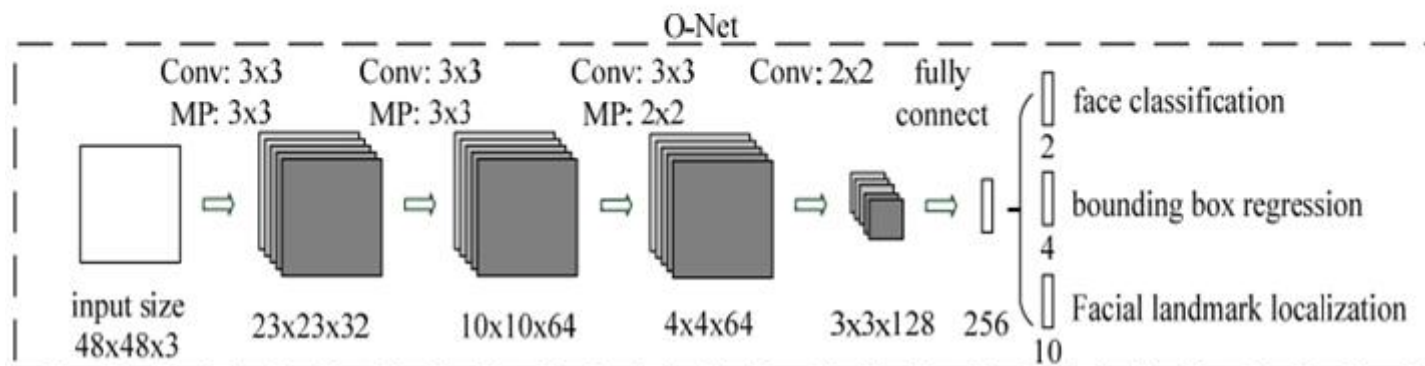
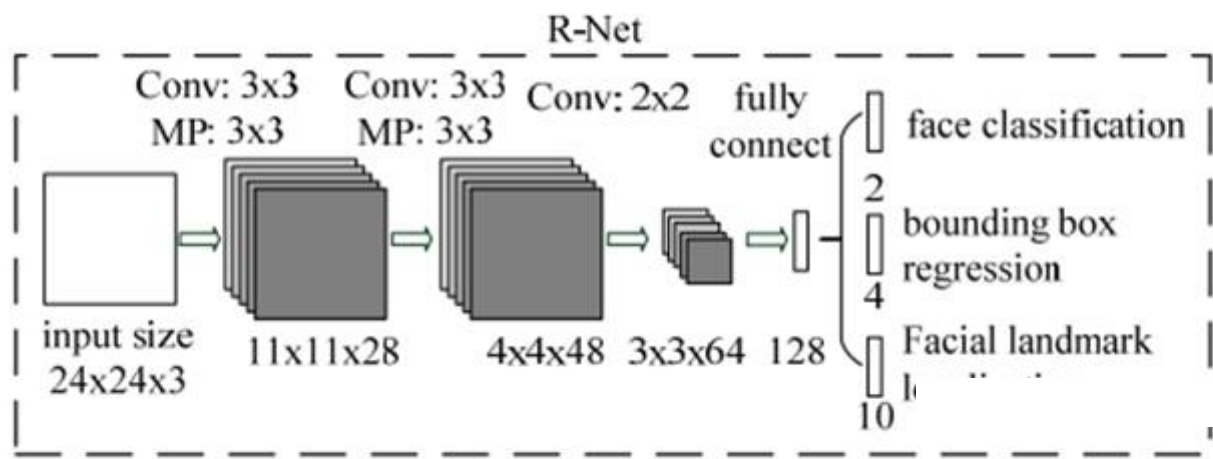
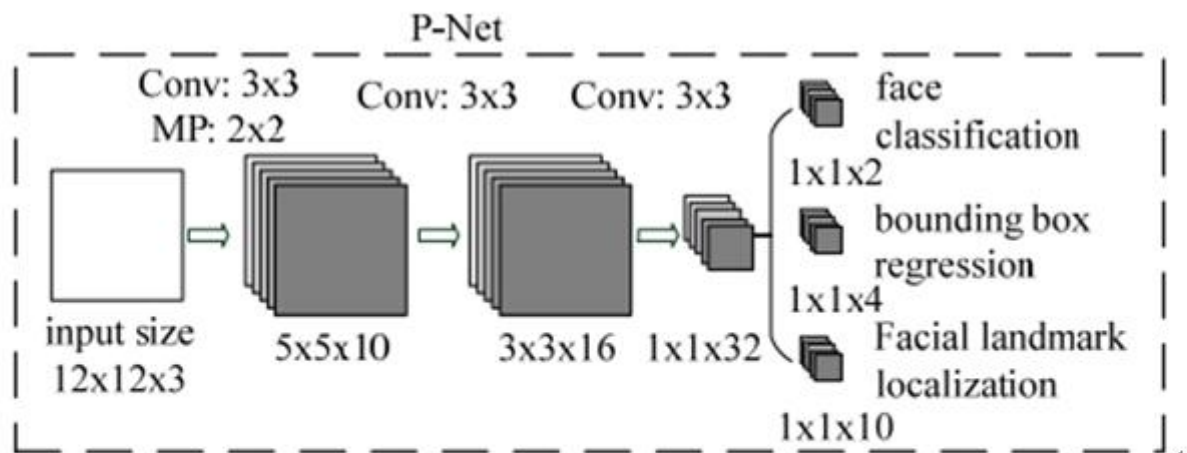
### windows下让自己的程序调用cafi

LZWXAZ: @qq\_14845119:是的, 博主, 我用的就是CPU版的, 电脑里没有GPU

### 图像数据库

weixin\_32528779: 楼主想问一下!! imagenet下载下来图片咋看啊? 解压包打开是没有后缀的文件

caffe模型weights&featureMap 可



prototxt的更加详细的网络结构如下: 分别为det1,det2,det3。

关闭

watersink: @touqiuyan8418:加上这2句

```
#include<lt;caffe/layers/decon...
```

caffe模型weights&featureMap 可

touqiuyan8418: 博主, 您好, 请问像可视化fcn的网络, 其中运行的时候说是没有反卷积层, 只有conv,softmax,...

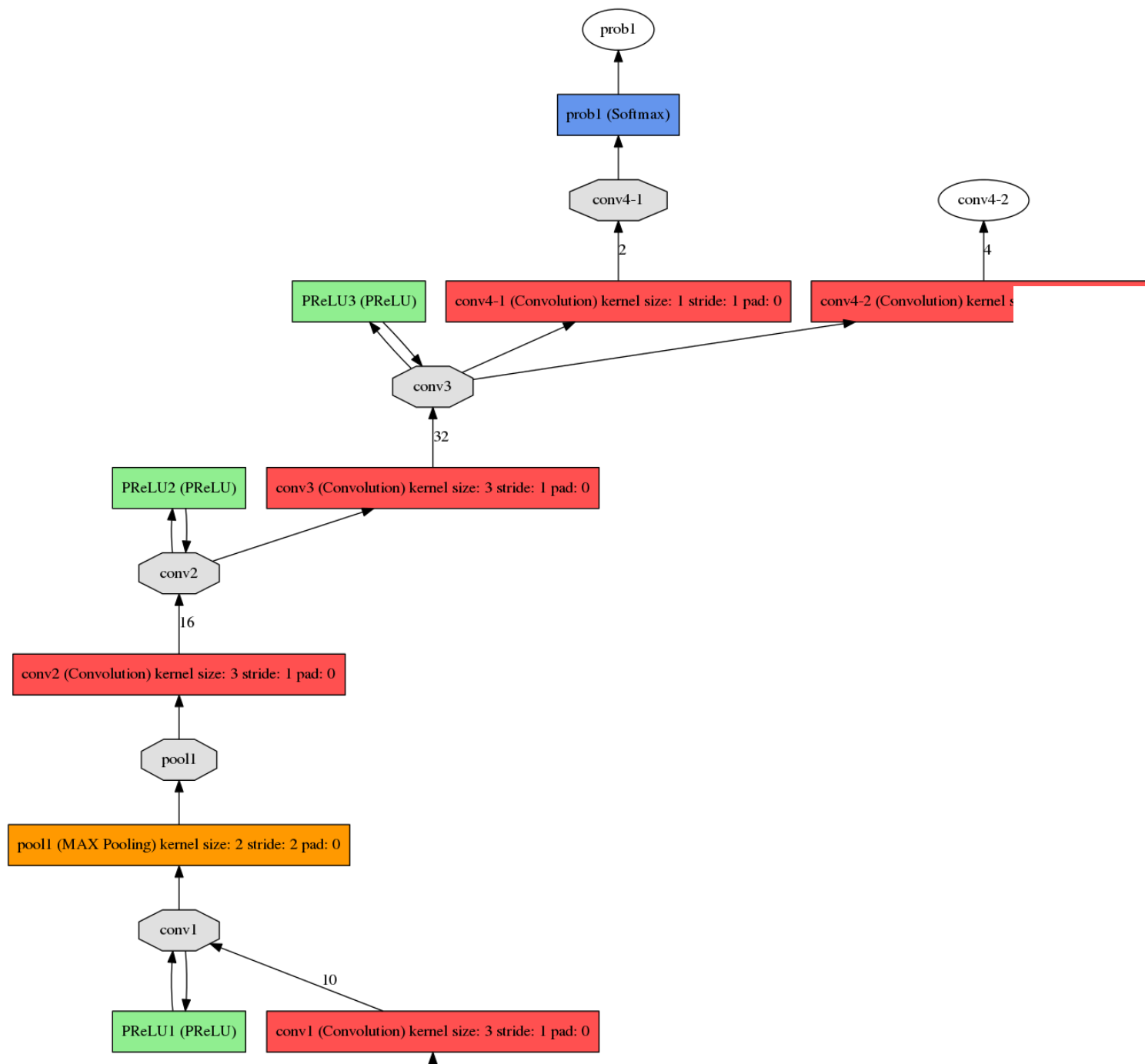
windows下让自己的程序调用cafi

纸巾盒: @qq\_38131712:是引用dll的地方出错了, 是要64位的

windows下让自己的程序调用cafi

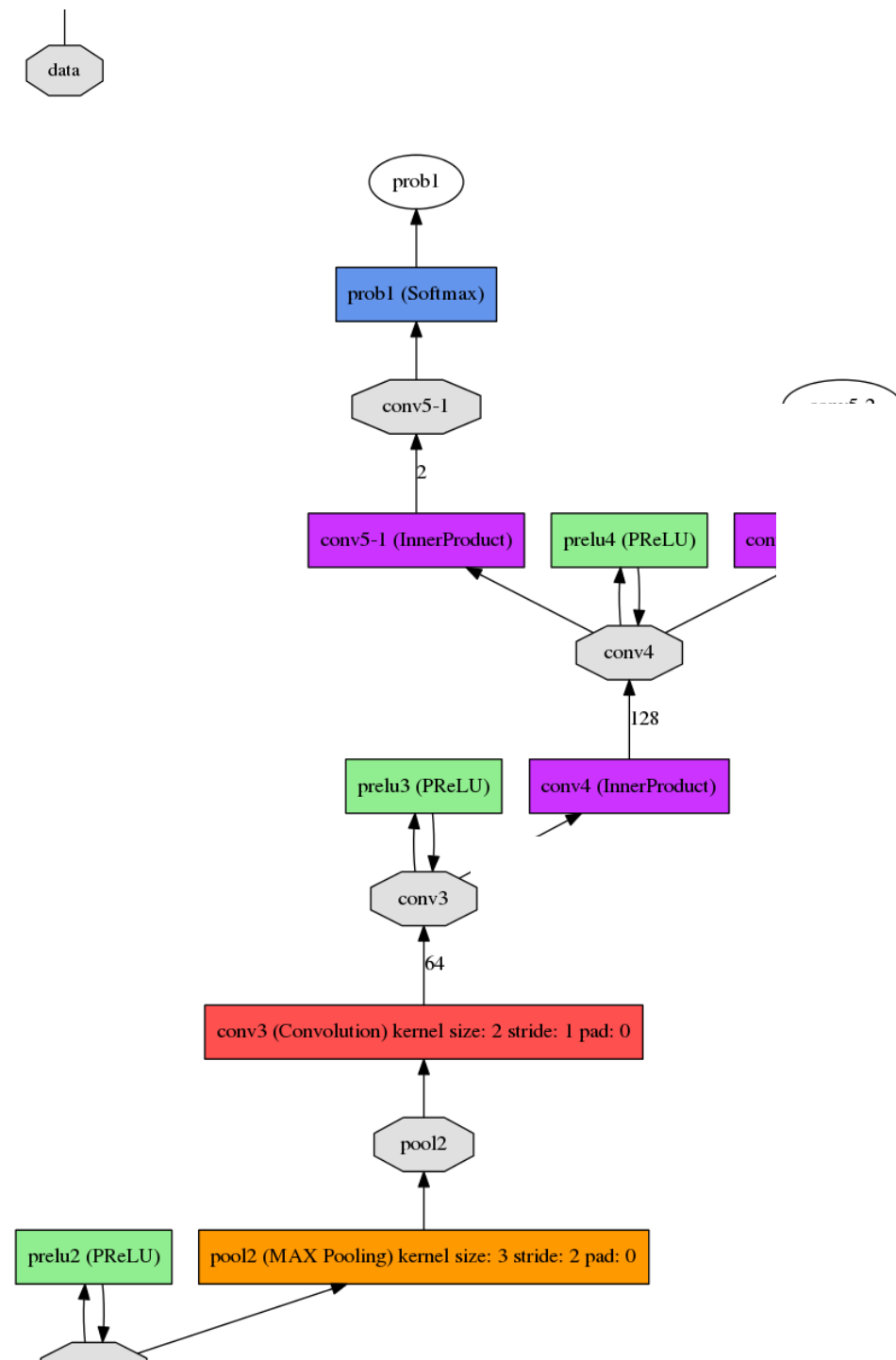
纸巾盒: @he5688:原来是dll引用的不对, 引用成32位的了。。。。。

det1.prototxt结构:

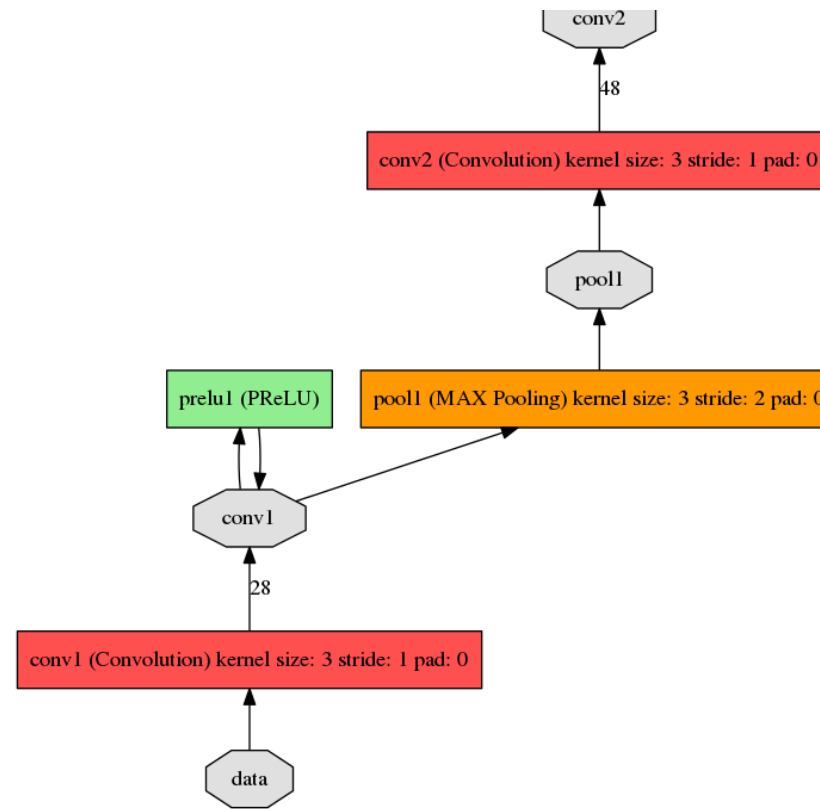


关闭

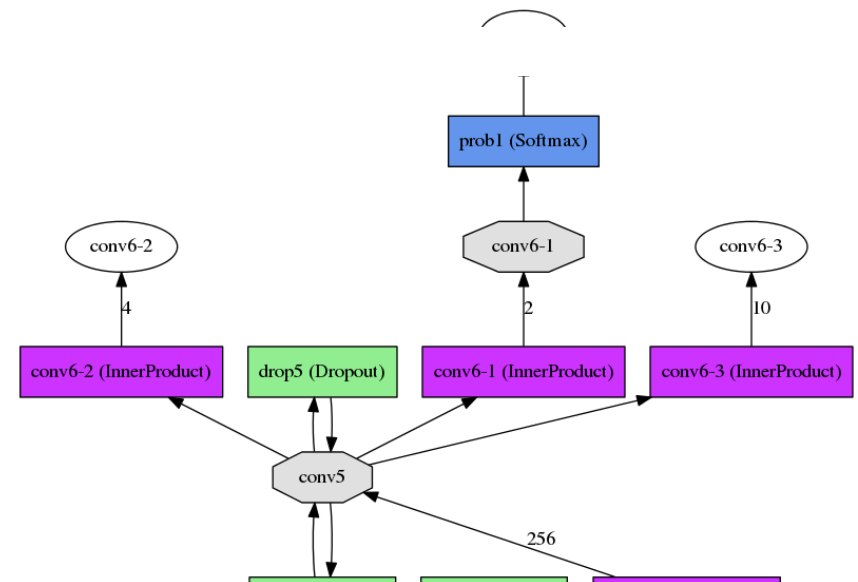
det2.prototxt结构：



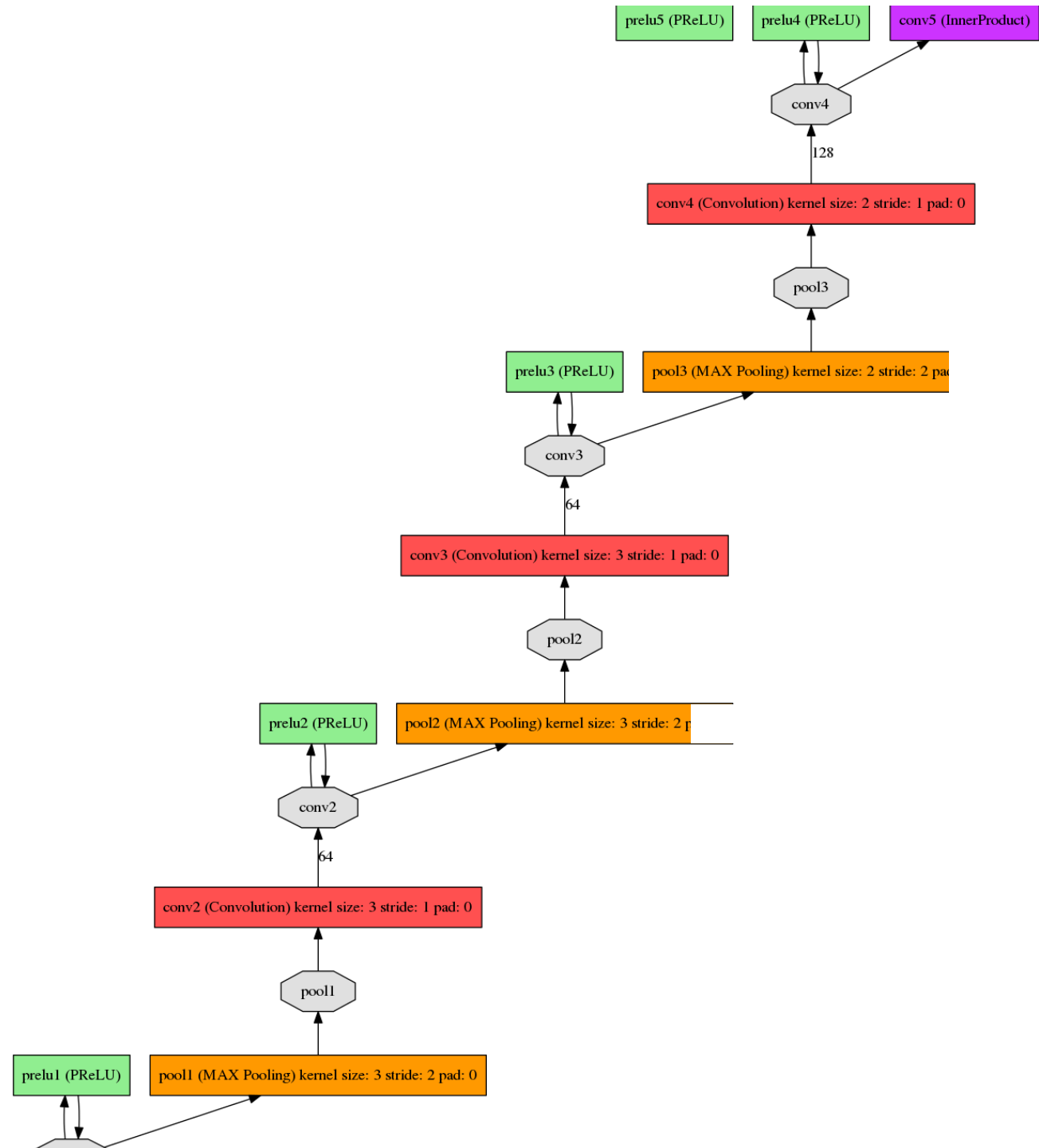
关闭



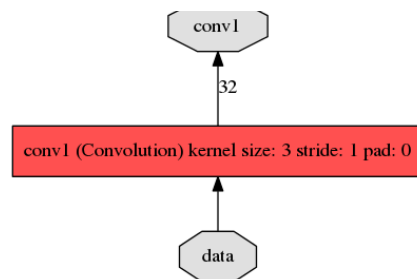
det3.prototxt结构：



关闭





**训练：**

MTCNN特征描述子主要包含3个部分，人脸/非人脸分类器，边界框回归，地标定位。

人脸分类：

$$L_i^{det} = -(y_i^{det} \log(p_i) + (1 - y_i^{det})(1 - \log(p_i)))$$

$$y_i^{det} \in \{0,1\}$$

上式为人脸分类的交叉熵损失函数，其中， $p_i$ 为是人脸的概率， $y_i^{det}$ 为背景的真实标签。

边界框回归：

$$L_i^{box} = \|\hat{y}_i^{box} - y_i^{box}\|_2^2$$

$$y_i^{box} \in \mathbb{R}^4$$

上式为通过欧氏距离计算的回归损失。其中，带尖的 $y$ 为通过网络预测得到，不带尖的 $y$ 为实际的真实的目标坐标。其中， $y$ 为一个（左上角 $x$ ，左上角 $y$ ，长，宽）组成的四元组。

地标定位：

$$L_i^{landmark} = \|\hat{y}_i^{landmark} - y_i^{landmark}\|_2^2$$

$$y_i^{landmark} \in \mathbb{R}^{10}$$

和边界回归一样，还是计算网络预测的地标位置和实际真实地标的欧式距离，并最小化该距离。其中，带尖的 $y$ 为通过网络预测得到，不带尖的 $y$ 为实际的真实的地标坐标。由于一共5个点，每个点2个坐标，所以， $y$ 属于十元组。

多个输入源的训练：

[关闭](#)

$$\min \sum_{i=1}^N \sum_{j \in \{det, box, landmark\}} \alpha_j \beta_i^j L_i^j$$

$$\beta_i^j \in \{0, 1\}$$

P-Net R-Net ( $\alpha_{det} = 1, \alpha_{box} = 0.5, \alpha_{landmark} = 0.5$ )

O-Net ( $\alpha_{det} = 1, \alpha_{box} = 0.5, \alpha_{landmark} = 1$ )

整个的训练学习过程就是最小化上面的这个函数，其中，N为训练样本数量， $\alpha_j$ 表示任务的重要性， $\beta_j$ 为样本标签， $L_j$ 为上面的损失函数。

在训练过程中，为了取得更好的效果，作者每次只后向传播前70%样本的梯度，这样来保证传递的都是有效类似latent SVM，只是作者在实现上更加体现了深度学习的端到端。

在训练过程中，y尖和y的交并集IoU ( Intersection-over-Union ) 比例：

0-0.3：非人脸

0.65-1.00：人脸

0.4-0.65：Part人脸

0.3-0.4：地标

训练样本的比例，负样本:正样本:part样本:地标=3:1:1:2

**安装步骤：**

caffe-windows的安装：

[http://blog.csdn.net/qq\\_14845119/article/details/52415090](http://blog.csdn.net/qq_14845119/article/details/52415090)

Pdollar toolbox的安装：

Pdollar toolbox由UCSD的Piotr Dollar编写，侧重物体识别 ( Object Recognition ) 检测相关的特征提取和分类算法。这个工具箱属于专而精的类型，主要就是Dollar的几篇物体检测的论文的相关算法，如果做物体识别相关的研究，应该是很好用的。同时它的图像操作或矩阵操作函数也可以作为Matlab图像处理工具箱的补充，功能主要包括几个模块：

\* **channels**模块，图像特征提取，包括HOG等，Dollar的研究工作提出了一种Channel Feature的特征[2]，因此这

关闭

个channels主要包括了提取这一特征需要的一些基本算法梯度、卷及等基本算法

\* **classify**模块, 一些快速的分类相关算法, 包括random ferns, RBF functions, PCA等

\* **detector**模块, 与Channel Feature特征对应的检测算法1

\* **filters**模块, 一些常规的图像滤波器

\* **images**模块, 一些常规的图像、视频操作, 有一些很实用的函数

\* **matlab**模块, 一些常规的Matlab函数, 包括矩阵计算、显示、变量操作等, 很实用

\* **videos**模块, 一些常规的视频操作函数等

下载链接: <https://github.com/pdollar/toolbox>

下载到Toolbox后, 将其解压到任意目录下, 如E:\MATLAB\MATLAB Production Server\toolbox

在Matlab命令行中输入

```
addpath(genpath('toolbox-masterROOT'));savepath;
```

将解压目录加入Matlab路径。其中toolbox-masterROOT为解压目录路径, 如解压到E:\ MATLAB\MATLAB Production Server\toolbox时, 则命令为

```
addpath(genpath('E:\ MATLAB\MATLAB ProductionServer\toolbox')); savepath;
```

这样Piotr's Image & VideoMatlab Toolbox就安装好了。

关闭

path里面加入caffe的库目录,例如本人的path加入如下的路径



打开demo.m,修改其中的caffe\_path, pdollar\_toolbox\_path, caffe\_model\_path。  
。同时由于本人电脑没有GPU,因此对其做如下修改。

```
%path of toolbox
caffe_path='E:\caffe\Build\x64\release\matcaffe';
pdollar_toolbox_path='E:\MATLAB\MATLAB Production Server\toolbox'
caffe_model_path=.\model
addpath(genpath(caffe_path));
addpath(genpath(pdollar_toolbox_path));

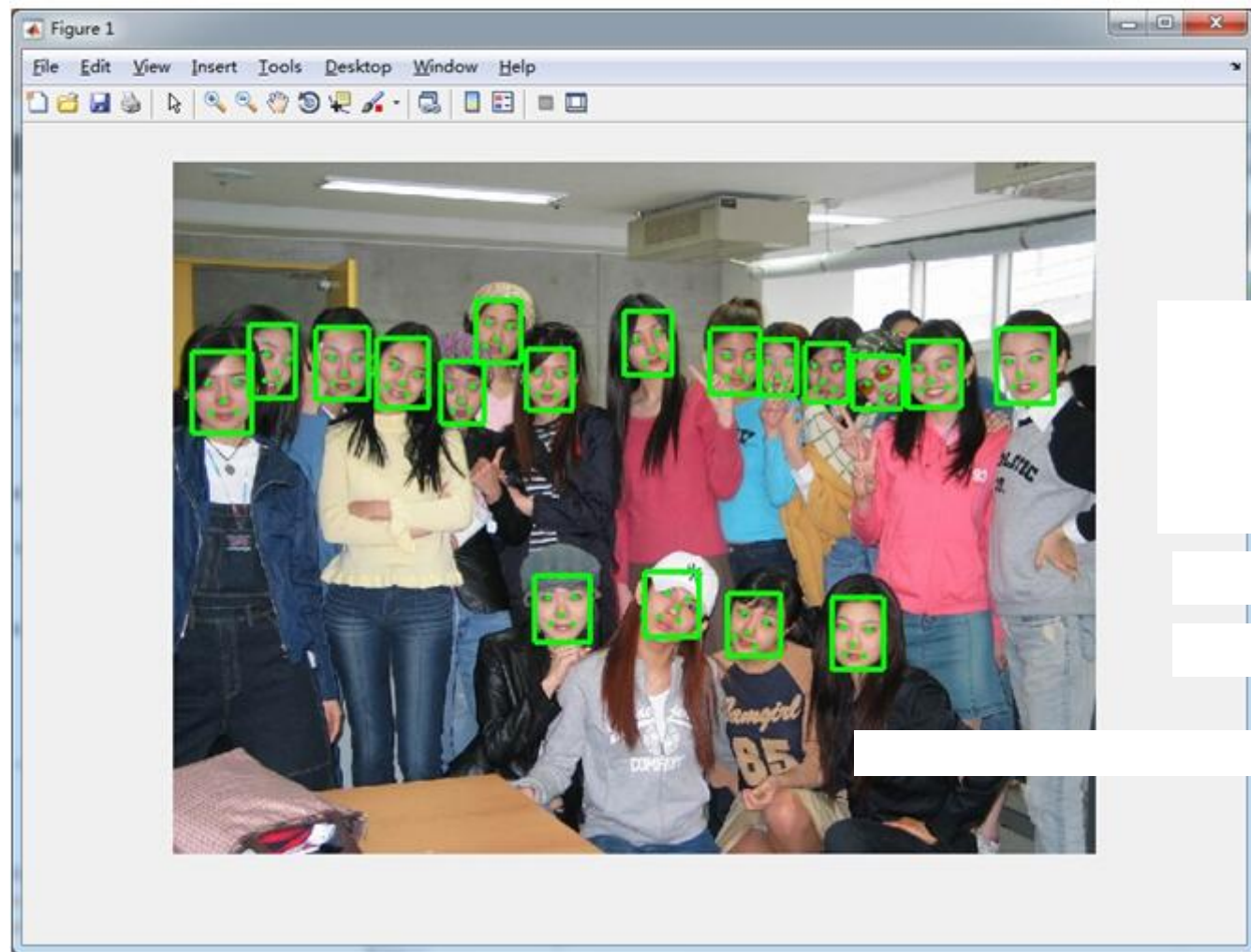
%use cpu
caffe.set_mode_cpu();
%gpu_id=0;
%caffe.set_mode_gpu();
%caffe.set_device(gpu_id);
```

### 实验结果：

运行时间1.2S,按照检测出18个脸算,平均一个66MS,运行版本为release版本。从实验结果来看,不管是检测还是对齐都是空前的好。以我的经验来看,face++的对齐是最好的,剩下的开源的里面这篇MTCNN算是最好的了,然后才是SDM。

关闭

Elapsed time is 1.228995 seconds.



从下面的作者在FDDB+WIDERFACE+AFLW上验证的正确性来看，基本95%的准确度。可见该MTCNN的性能和效率都是很给力的。

关闭

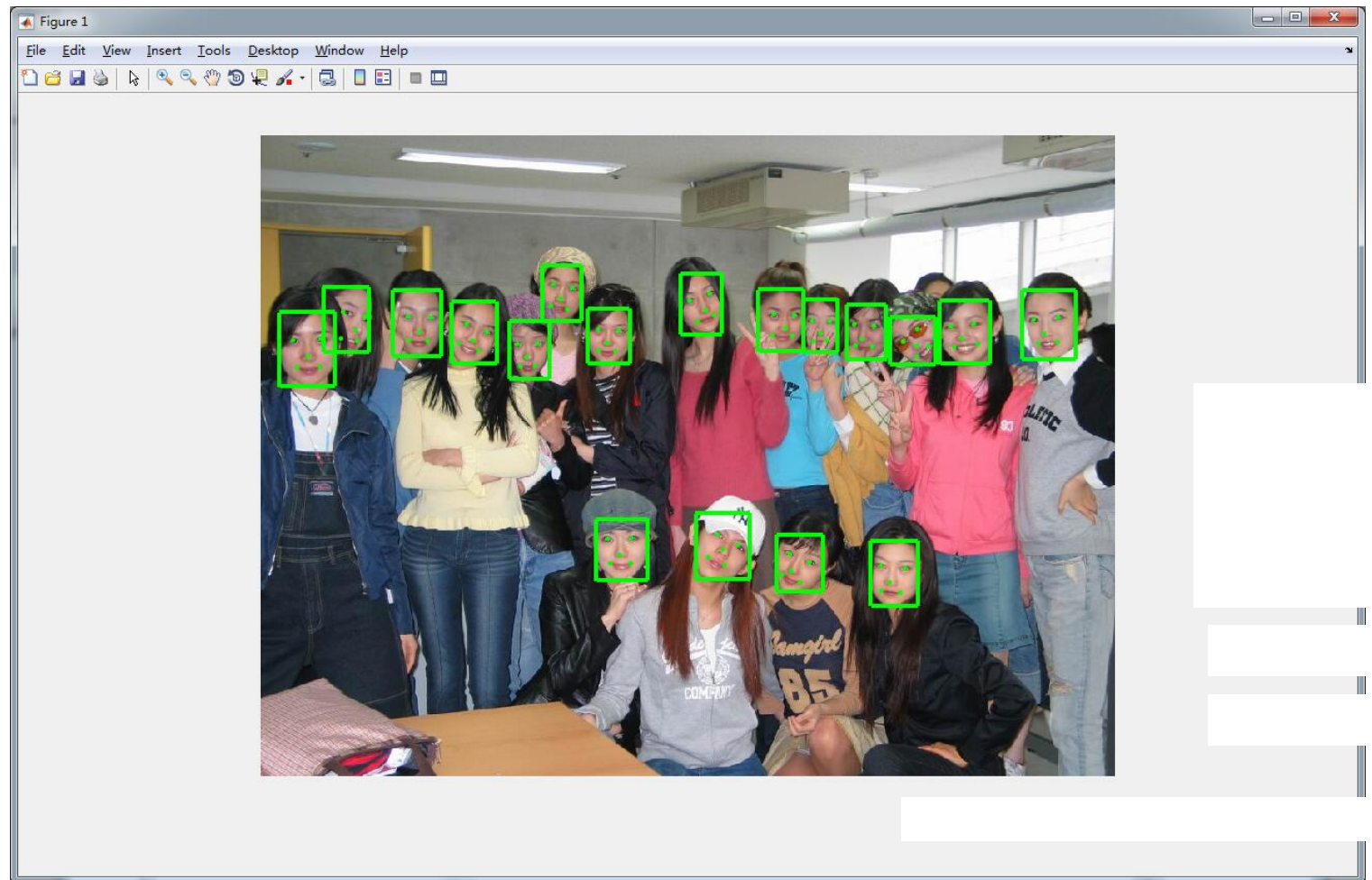
Group	CNN	300 × Forward Propagation	Validation Accuracy
Group1	12-Net [19]	0.038s	94.4%
	P-Net	0.031s	94.6%
Group2	24-Net [19]	0.738s	95.1%
	R-Net	0.458s	95.4%
Group3	48-Net [19]	3.577s	93.2%
	O-Net	1.347s	95.4%

从实验结果可以看出，上图的第二行的第二个对齐的出了问题，因此，本人对其程序进行了微小改动，实际图，时间和效果上都有了提升。

```
Elapsed time is 0.672338 seconds.
```

关闭

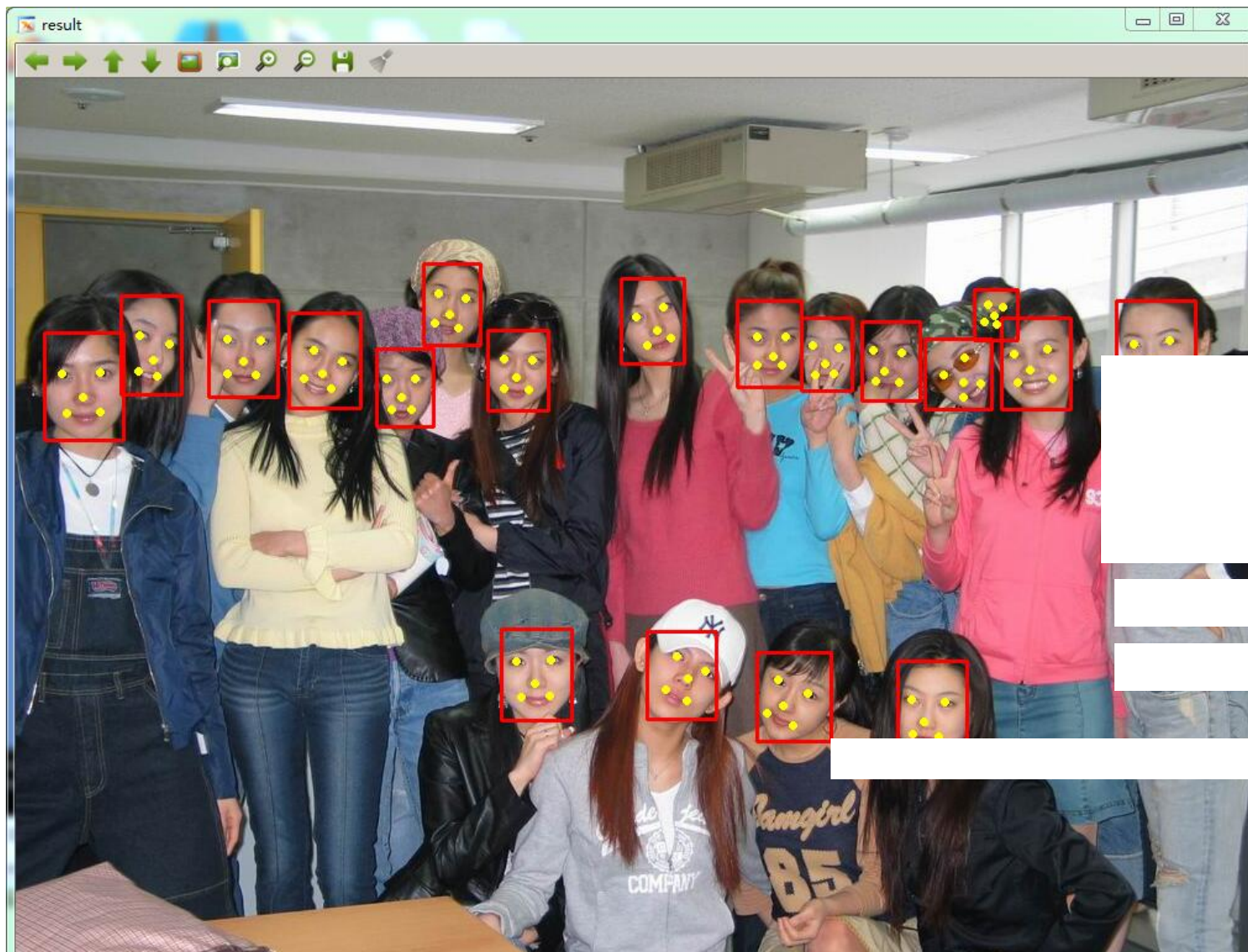




关闭

下载链接：[http://download.csdn.net/detail/qq\\_14845119/9653138](http://download.csdn.net/detail/qq_14845119/9653138)

组后感谢公司大牛的帮忙，c语言版本终于改出来了。老实说，真心不容易，走了好多坑。贴个效果图，纪念那些苦了笑了的时光。



关闭

#### References:

- [1] [https://kpzhang93.github.io/MTCNN\\_face\\_detection\\_alignment/index.html](https://kpzhang93.github.io/MTCNN_face_detection_alignment/index.html)
- [2] [https://github.com/kpzhang93/MTCNN\\_face\\_detection\\_alignment/tree/master/code/codes](https://github.com/kpzhang93/MTCNN_face_detection_alignment/tree/master/code/codes)
- [3] ZhangK, Zhang Z, Li Z, et al. Joint Face Detection and Alignment using Multi-taskCascaded Convolutional Networks[J]. arXiv preprint arXiv:1604.02878, 2016.



顶 踩  
8 0

上一篇 SeetaFace大总结

下一篇 visual studio 插件篇(适合CV方向的开发者)

#### 相关文章推荐

- 我读FaceNet
- 自然语言处理在“天猫精灵”的实践应用--姜飞俊
- 【深度学习论文笔记】FaceNet: A Unified Embed...
- 蚂蜂窝大数据平台架构及Druid引擎实践--汪木铃
- 人脸识别之FaceNet
- Retrofit 从入门封装到源码解析
- python2.7与python3.5中不同的地方 ( 自己踩过的...
- 程序员如何转型AI工程师
- 64位ubuntu16.04安装tensorflow的方方面面
- 深入探究Linux/VxWorks的设备树
- YOLO v2之总结篇 ( linux+windows )
- 使用QEMU搭建u-boot+LINUX+NETS嵌入式开发环境
- 在cuda8.0+faster-rcnn ( python版 ) 下使用kitti数据..
- Tensorflow Ubuntu16.04上安装及CPU运行Tensor...
- mtcnn分析
- Caffe 开发流程简介

#### 查看评论

7楼 零下275度 2017-09-13 15:29发表



把这个m文件加载到l i n u x 版本的m a t l a b , 如何进行添加啊? 且没有r o o t 权限, 求指导下

Re: [watersink](#) 2017-09-13 18:23发表



回复零下275度: 我不理解你说的加载是什么意思, 那个m文件直接用matlab就可以运行的

6楼 [Saber-alter](#) 2017-09-06 22:07发表



博主辛苦了, 请教一个问题, 既然第二个网络R-Net输入的是P-Net输出的bbox, 那么它是如何训练进行bounding box regression? 输入不是整张图片不能进行box regression吧, 毕竟最后label的坐标标签和输入图片对不上啊

5楼 [李菲儿](#) 2017-08-22 08:43发表



你好, 我想问调用CPU的那个代码, 总出现+caffe\private\caffe\_.mexw64&#39;; 找不到指定的模块。我看了private目录, 怎么办呀?

Re: [watersink](#) 2017-08-22 09:29发表



回复李菲儿: 默认编译完了matcaffe之后, 就会在caffe/matlab/+caffe/private目录下生成, caffe\_.mexa64, 题, 导入的话, 使用, addpath(genpath('caffe/matlab/+caffe/private'));就可以导入了

Re: [李菲儿](#) 2017-08-23 14:32发表



博主, 我用的是vs2013和MATLAB 2015b, 会是MATLAB2015b识别不了编译器吗?

>> mex -setup C++

Error using mex

No supported compiler or SDK was found. You can install the freely available MinGW-w64 C/C++ compiler; see Install MinGW-w64 Compiler. For more options, see

<http://www.mathworks.com/support/compilers/R2015b/win64.html>.

MATLAB出现这种个问题。

Re: [李菲儿](#) 2017-08-23 13:55发表



博主你好, 还是那个问题。我生成的caffe\_.mexa64是在/Build/x64/Release/matcaffe/+caffe/private里面。这个路径('caffe/matlab/+caffe/private')下没有生成, 我把它拷贝过来还是不管用。那这个问题该怎么解决?

Re: [watersink](#) 2017-08-23 14:49发表



关闭

回复李菲儿：这个应该就是你没生成Matlab的接口库，应该个matlab版本没啥关系吧，我的linux用的是2016b，windows是2015a都可以支持，应该是你的matlab没检测到vs2013的编译器。mex 是混合编译的命令，所以，要不是Matlab的问题，要不是vs的问题

Re: 李菲儿 2017-08-24 09:41发表



我的vs原来在MATLAB中可以编译，后来改了路径就打不开了，我重装之后就不会编译了。

Re: watersink 2017-08-24 10:41发表



回复李菲儿：所以说，看来事你电脑环境的问题，实在不行就重装吧

4楼 kunanzong5266 2017-05-07 11:19发表



其实Dlib库的68点检测只需2ms

3楼 redcp 2016-10-20 09:11发表



博主，你好，我是在校研究生一枚。最近在学习研究MTCNN。博主可以发份你改好的C代码给我么？我的QQ415152252@qq.com  
如果不方便恳请博主加我，我请教一些问题。谢谢你了

Re: watersink 2016-10-20 11:21发表



回复redcp：在说，如果你是研究生，那就是搞数学的了，研究网络结构才是你的重点，有matlab 的不是已经足够了么，你考虑C的干嘛。

Re: watersink 2016-10-20 09:16发表



回复redcp：很遗憾，没有开源的意向。

2楼 CodenameNet 2016-10-16 23:53发表



csdn要10分，没有那么多积分，能放到百度网盘不？

1楼 小四掰 2016-10-09 21:42发表



你好，请问如果我想在视频中做人脸识别，这个MTCNN或者是SDM、3000FPS哪个更适合一些呢？

Re: watersink 2016-10-09 23:02发表

关闭



回复小四掰：老实说，这个问题我也不知道怎么回答，因为我也还是学习阶段，做视频的话，应该得50MS每帧的样子，还要取决你的视频中平均每帧多少人脸，所以单纯的检测是跟不上速度的，必须得靠跟踪辅助，我还是建议SDM。

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

\* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

[公司简介](#) | [招贤纳士](#) | [广告服务](#) | [联系方式](#) | [版权声明](#) | [法律顾问](#) | [问题报告](#) | [合作伙伴](#) | [论坛反馈](#)

网站客服

杂志客服

微博客服

webmaster@csdn.net

400-660-0108

| 北京创新乐知信息技术有限公司 版权所有 | 江苏知之为计算机有限公司 | 江苏乐知网络科技有限公司

京 ICP 证 09002463 号 | Copyright © 1999-2017, CSDN.NET, All Rights Reserved



关闭