首页　　问答　　面试题　　程序人生　　开源代码库　　关于

Hi, 请登录　我要注册

大话面试　　Android试题　　IOS试题　　Java试题　　前端试题　　架构师试题　　Python试题　　.NET试题　　PHP试题　　算法试题　　C++试题　　计算机基础试题

🏠 > 架构师 > 用tornado、tensorflow、opencv打造一个在线性别识别、年龄识别、颜值打分服务

用tornado、tensorflow、opencv打造一个在线性别识别、年龄识别、颜值打分服务

搜索 🔍

📖 架构师　　👤 来源:nanjingdreamfly　　🕐 2017-03-07 17:39　　👁 351℃　　💬 0评论

分享到　　　　　　　　百度一下

目录(?)　　　　　　　　[+]

架构

1.web服务用tornado；

2.模型训练和预测用tensorflow；

3.opencv做人脸检测

Prerequisites:

tensorflow0.12
opencv3.0
python2.7
训练好的模型，模型训练参考https://github.com/dpressel/rude-carnie

主代码

```
1  # code: UTF-8
2  import tornado.ioloop
3  import tornado.web
4  import uuid
5  #import Image
6  import StringIO
7  import os
```

IT问道推荐　　关闭

银行贷款频频被拒？

"Dr信用牛牛"让你远离信用污点 国内首家
信用健康管理平台免费为你提供信用修复方案

```python
 8  import commands
 9  import guess
10  import json
11
12
13  class PredictHandler(tornado.web.RequestHandler):
14
15      def get(self):
16          self.render("index.html")
17
18      def post(self):
19          if self.request.files:
20              file_name = "%s" % uuid.uuid1()
21              print 'file_name',file_name
22              file_raw = self.request.files["file"][0]["body"]
23              usr_home = os.path.expanduser('~')
24              file_name = usr_home+"/tensorflow/static/tmp/m_%s.jpg" % file_name
25              fin = open(file_name,"w")
26              print "success to open file"
27              fin.write(file_raw)
28              fin.close()
29              print "use tensorflow"
30              #首先检测是否是人类，如果不是，直接返回
31              isman = guess.detectface(file_name)
32              feeds_json = {}
33              if isman == 1:
34                  #预测年龄、颜值打分
35                  age,score = guess.guessAge(file_name)
36                  #预测性别
37                  gender = guess.guessGender(file_name)
38                  print 'guess age is ', age
39                  print 'gender is ',gender
40                  dic = {}
41                  dic['isman'] = True
42                  dic['age'] = age
43                  dic['score'] = score
44                  dic['gender'] = gender
45                  feeds_json = json.dumps(dic)
46              elif isman == 0 :
47                  dic = {}
```

```
48              dic['isman'] = False
49              feeds_json = json.dumps(dic)
50          self.set_header('Content-Type', 'application/json; charset=UTF-8')
51          self.write(feeds_json)
52          self.finish()
53
54
55
56  application = tornado.web.Application([
57      (r"/predict", PredictHandler),
58  ])
59
60  if __name__ == "__main__":
61      application.listen(8882)
62      tornado.ioloop.IOLoop.instance().start()
```

使用到了guess.py，代码如下

```
1   #code: UTF-8
2   from __future__ import absolute_import
3   from __future__ import division
4   from __future__ import print_function
5   import sys
6   reload(sys)
7   sys.setdefaultencoding('utf-8')
8   from datetime import datetime
9   import math
10  import time
11  from data import inputs
12  import numpy as np
13  import tensorflow as tf
14  from model import select_model, get_checkpoint
15  from utils import ImageCoder, make_batch, FaceDetector
16  import os
17  import csv
18  import random
19
20  RESIZE_FINAL = 227
21  GENDER_LIST =['M','F']
22  AGE_LIST = ['(0, 2)','(4, 6)','(8, 12)','(15, 20)','(25, 32)','(38, 43)','(48, 53)','(60, 100)']
```

```
23  AGE_MODEL_PATH = '/work/facescore/22801'
24  GENDER_MODEL_PATH = '/run-1322'
25  model_checkpoint_path = ''
26
27  tf.app.flags.DEFINE_string('model_dir', '',
28                             'Model directory (where training data lives)')
29
30  tf.app.flags.DEFINE_string('class_type', 'age',
31                             'Classification type (age|gender)')
32
33
34  tf.app.flags.DEFINE_string('device_id', '/cpu:0',
35                             'What processing unit to execute inference on')
36
37  tf.app.flags.DEFINE_string('filename', '',
38                             'File (Image) or File list (Text/No header TSV) to process')
39
40  tf.app.flags.DEFINE_string('target', '',
41                             'CSV file containing the filename processed along with best guess and score')
42
43  tf.app.flags.DEFINE_string('checkpoint', 'checkpoint',
44                             'Checkpoint basename')
45
46  tf.app.flags.DEFINE_string('model_type', 'default',
47                             'Type of convnet')
48
49  tf.app.flags.DEFINE_string('requested_step', '', 'Within the model directory, a requested step to re
50
51  tf.app.flags.DEFINE_boolean('single_look', False, 'single look at the image or multiple crops')
52
53  tf.app.flags.DEFINE_string('face_detection_model', '', 'Do frontal face detection with model specifi
54
55  FLAGS = tf.app.flags.FLAGS
56
57  def one_of(fname, types):
58      for ty in types:
59          if fname.endswith('.' + ty):
60              return True
61      return False
62
```

```python
def resolve_file(fname):
    if os.path.exists(fname): return fname
    for suffix in ('.jpg', '.png', '.JPG', '.PNG', '.jpeg'):
        cand = fname + suffix
        if os.path.exists(cand):
            return cand
    return None

def classify(sess, label_list, softmax_output, coder, images, image_file):

    print('Running file %s' % image_file)
    image_batch = make_batch(image_file, coder, not FLAGS.single_look)
    batch_results = sess.run(softmax_output, feed_dict={images:image_batch.eval()})
    output = batch_results[0]
    batch_sz = batch_results.shape[0]
    for i in range(1, batch_sz):
        output = output + batch_results[i]

    output /= batch_sz
    best = np.argmax(output)
    best_choice = (label_list[best], output[best])
    print('Guess @ 1 %s, prob = %.2f' % best_choice)

    #calculate face score
    score = scoreAge(output)

    nlabels = len(label_list)
    if nlabels > 2:
        output[best] = 0
        second_best = np.argmax(output)

        print('Guess @ 2 %s, prob = %.2f' % (label_list[second_best], output[second_best]))
    return label_list[best],score

def classifyGender(sess, label_list, softmax_output, coder, images, image_file):

    print('Running file %s' % image_file)
    image_batch = make_batch(image_file, coder, not FLAGS.single_look)
    batch_results = sess.run(softmax_output, feed_dict={images:image_batch.eval()})
    output = batch_results[0]
```

```python
103        batch_sz = batch_results.shape[0]
104        for i in range(1, batch_sz):
105            output = output + batch_results[i]
106
107        output /= batch_sz
108        best = np.argmax(output)
109        best_choice = (label_list[best], output[best])
110        print('Guess @ 1 %s, prob = %.2f' % best_choice)
111
112        #calculate face score
113        #score = scoreAge(output)
114
115        nlabels = len(label_list)
116        if nlabels > 2:
117            output[best] = 0
118            second_best = np.argmax(output)
119
120            print('Guess @ 2 %s, prob = %.2f' % (label_list[second_best], output[second_best]))
121        return label_list[best]
122
123    def scoreAge(prop_list):
124        #['(0, 2)', '(4, 6)', '(8, 12)', '(15, 20)', '(25, 32)', '(38, 43)', '(48, 53)', '(60, 100)']
125        #[90, 80, 80, 70, 60, 50, 40, 20]
126        score_list = [100, 100, 95, 90, 80, 50, 40, 20]
127        random_score_list = []
128        finalScore = 0
129        for score in score_list:
130            randScore = score + random.random() * 10
131            random_score_list.append(randScore)
132        j = 0
133        while(j < len(random_score_list)):
134            prop = prop_list[j]
135            score = random_score_list[j]
136            finalScore = finalScore + score * prop
137            j = j + 1
138        return finalScore
139
140    def batchlist(srcfile):
141        with open(srcfile, 'r') as csvfile:
142            reader = csv.reader(csvfile)
```

```
143            if srcfile.endswith('.csv') or srcfile.endswith('.tsv'):
144                print('skipping header')
145                reader.next()
146
147            return [row[0] for row in reader]
148
149    def detectface(filename):
150        files = []
151        #print('Using face detector %s' % FLAGS.face_detection_model)
152        face_detect = FaceDetector('haarcascade_frontalface_default.xml')
153        face_files, rectangles = face_detect.run(filename)
154        files += face_files
155        if (len(files)>0) :
156            return 1
157        else:
158            return 0
159
160    def guessAge(image_file):
161
162        #import!!!Fix the bug http://stackoverflow.com/questions/33765336/remove-nodes-from-graph-or-reset-entire-default-graph
163        tf.reset_default_graph()
164        with tf.Session() as sess:
165
166            age_label_list = AGE_LIST
167            agelabels = len(age_label_list)
168
169            # print('Executing on %s' % FLAGS.device_id)
170            model_fn = select_model('inception')
171
172            images = tf.placeholder(tf.float32, [None, RESIZE_FINAL, RESIZE_FINAL, 3])
173            logits_age = model_fn(agelabels, images, 1, False)
174            init = tf.initialize_all_variables()
175
176            requested_step = FLAGS.requested_step if FLAGS.requested_step else None
177
178            checkpoint_path = '%s' % (AGE_MODEL_PATH)
179            # update in 0.11 version
180
181            model_checkpoint_path, global_step = get_checkpoint("/22801/", requested_step, FLAGS.checkpo
182        #print 'model_checkpoint_path is', model_checkpoint_path
```

```python
183            #print model_checkpoint_path
184        saver = tf.train.Saver()
185            if not saver.last_checkpoints :
186                saver.restore(sess, model_checkpoint_path)
187
188
189        softmax_output = tf.nn.softmax(logits_age)
190
191        coder = ImageCoder()
192
193        files = []
194
195        # detect age
196        best_choice = classify(sess, age_label_list, softmax_output, coder, images, image_file)
197
198        sess.close()
199        return best_choice
200
201 def guessGender(image_file):
202     tf.reset_default_graph()
203     with tf.Session() as sess:
204
205        sess = tf.Session()
206        age_label_list = AGE_LIST
207        gender_label_list = GENDER_LIST
208        genderlabels = len(gender_label_list)
209
210        # print('Executing on %s' % FLAGS.device_id)
211        model_fn = select_model('')
212
213        images = tf.placeholder(tf.float32, [None, RESIZE_FINAL, RESIZE_FINAL, 3])
214        logits_gender = model_fn(genderlabels, images, 1, False)
215        init = tf.initialize_all_variables()
216
217        requested_step = 10000
218
219        checkpoint_path = '%s' % (GENDER_MODEL_PATH)
220
221        model_checkpoint_path, global_step = get_checkpoint(checkpoint_path, requested_step, FLAGS.c
222
```

```
223         saver = tf.train.Saver()
224         saver.restore(sess, model_checkpoint_path)
225
226         softmax_output = tf.nn.softmax(logits_gender)
227
228         coder = ImageCoder()
229
230         files = []
231
232         # detect gender
233         #try:
234         best_choice = classifyGender(sess, gender_label_list, softmax_output, coder, images, image_file)
235         return best_choice
236         #except Exception as e:
237         # print(e)
238         # print('Failed to run image %s ' % image_file)
```
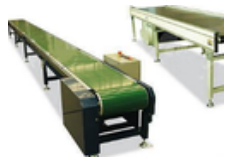
相关推荐

○ 用python、opencv快速实现人脸检测   ○ 一种ReactNative Android、IOS、Web融合的路由生成引擎

○ Recycleview封装Adapter增加头部尾部   ○ 博弈论游戏 - 信任的进化

○ 那些年刷过的题 - Runtime   ○ 制作表白动画

○ View draw流程分析   ○ 自定义一个和京东一样的效果的UIAlertView

○ 自定义UIDatePicker，以后不用第三方的，想怎么改就怎么改~~~   ○ 从源码角度分析SrcollView嵌套ListView显示不全的问题

○ Kotlin基本语法

登录

留下您的评论和问题吧

还没有评论，快来抢沙发吧！

IT问道网正在使用畅言

热评话题

先AES加密后MD5加密 - 移动开发 - IT问道

Xcode:用于管理多个 target 配置的 XCConfig 文件 - 移动开发 - IT问道

nodejs+redis应用 - 互联网 - IT问道

Swift 中的尾递归和蹦床 - 移动开发 - IT问道

Xcode 扩展 - 移动开发 - IT...

Sublime Text 3最好的功能、插件和设置 - 移动开

NodeJS与Django协同应用

IT问道推荐　　　　　　关闭

【Java集合源码剖析】Arr...

银行贷款频频被拒？

"Dr信用牛牛"让你远离信用污点 国内首家
信用健康管理平台免费为你提供信用修复方案