**pillow实例 | 生成随机验证码**

时间：2015-02-24 17:26:29      阅读：417      评论：0      收藏：0      [点我收藏+]

标签：[算法](http://www.bubuko.com/so/1/%e7%ae%97%e6%b3%95)   [class](http://www.bubuko.com/so/1/class)   [style](http://www.bubuko.com/so/1/style)   [代码](http://www.bubuko.com/so/1/%e4%bb%a3%e7%a0%81)   [com](http://www.bubuko.com/so/1/com)   [log](http://www.bubuko.com/so/1/log)   [使用](http://www.bubuko.com/so/1/%e4%bd%bf%e7%94%a8)   [src](http://www.bubuko.com/so/1/src)   [问题](http://www.bubuko.com/so/1/%e9%97%ae%e9%a2%98)

1 PIL(Python Image Library)

PIL是Python进行基本图片处理的package，囊括了诸如图片的剪裁、缩放、写入文字等功能。现在，我便以生成随机验证码为例，讲述PIL的基本用法。

PIL库似乎已经被人抛弃，就为更新，上次使用时竟然不能用show()直接将图片，在系统默认的图片管理器中打开。好在pillow，一个PIL的方言，将PIL继续维护了下去。

生成验证码一般需要对写入的文字进行旋转、扭曲、变色等一系列操作，才能避免计算机算法的识别。

因此定义一个生成验证码的类是一个不错的选择，旋转、扭曲、变色就作为这个类的一系列方法。

2 初始化Captcha类

Captcha是Completely Automated Public Turing Test to Tell Computers and Humans Apart (全自动区分计算机和人类的图灵测试)的简称，是世界上最为著名的验证码项目，后来被google受够，现在它们开始用图片以外的其他方式来区分人与计算机

初始化需要设置验证码图片的大小、以及字体的大小，默认已选择了比较合适的值。

需要用到Draw子包的new方法新建一张图片，要给定mode、size和color。这就像是现实画画中的幕布。

mode与不同的图片表示格式有关，

彩色图片是可以通过红R、绿G、蓝B三原色合成而来，存储在计算机里的便是3个矩阵，这种最常见的模式叫做‘RGB‘；

黑白图片则只需要一层，用‘1‘表示；

在计算机屏幕显示的彩色图片在三原色的基础上，进而出现一个透明层，表示能够透过多少下一层图片，这种模式叫做‘RGBA‘，A代指alpha。

from PIL import Image  
  
class Captcha(object):

def \_\_init\_\_(self,size=(100,40),fontSize=30):

self.font = ImageFont.truetype(‘C:\Windows\Fonts\Arial.ttf‘,fontSize)

self.size = size

self.image = Image.new(‘RGBA‘,self.size,(255,)\*4)

self.texts = self.randNum(5)

ImageFont包用于处理字体，需要具体给定使用字体的路径以及字体的大小，Windows系统下字都在C:\Windows\Fonts\路径下。

 3 生成随机的数字及颜色

随机数字

使用random包的randint(start,end)方法，随机产生0~9中的一个数字，需要注意的是包含start和end

def randNum(self,bits):

return ‘‘.join(str(random.randint(0,9)) for i in range(bits))

随机颜色

颜色一般用8位表示，十进制下便是0~255的一个整数，随机生成一个三元组即可

def randColor(self):

self.fontColor = (random.randint(0,250),random.randint(0,250),random.randint(0,250))

 这都非常简单

4 写入文字

写入文字需要用到ImageDraw包

def write(self,text,x):

draw = ImageDraw.Draw(self.image)

draw.text((x,4),text,fill=self.fontColor,font=self.font)

 text方法需要指定，写入文字的位置，以左上角为原点，当然以像素为单位；写入的文字；写入文字的颜色；最后说字体

5 旋转文字

旋转一般不是直接旋转文字，而是旋转幕布，在这里需要随机旋转一个角度。

旋转幕布会产生一个问题，旋转后边边角角会伸到画板的外面，而最终我们需要一个方方正正的验证码图片。

伸到画板外面的幕布需要剪裁掉，因为旋转空余的地方又需要补上。

Image.composite(img1,img2,img1)就可以解决这个问题，它会把img1透明/空白的地方，用img2补上。在这里应该是纯白的背景。

def rotate(self):

rot = self.image.rotate(random.randint(-10,10),expand=0) #默认为0，表示剪裁掉伸到画板外面的部分

fff = Image.new(‘RGBA‘,rot.size,(255,)\*4)

self.image = Image.composite(rot,fff,rot)

 6 形成验证码

有了上面的方法作铺垫，以一定的间隔，写上几个数字，就得到验证码了

def writeNum(self):

x = 10

xplus = 15

for text in self.texts:

self.randColor()

self.write(text, x)

self.rotate()

x += xplus

return self.texts

 此方法会返回验证码的文字，这样计算机才能判断人是否输入正确。不像上面的几个方法，这个方法是面向用户的。

最终的效果是这样子的

附：完整的代码

class Captcha(object):

def \_\_init\_\_(self,size=(100,40),fontSize=30):

self.font = ImageFont.truetype(‘C:\Windows\Fonts\Arial.ttf‘,fontSize)

self.size = size

self.image = Image.new(‘RGBA‘,self.size,(255,)\*4)

self.texts = self.randNum(5)

def rotate(self):

rot = self.image.rotate(random.randint(-10,10),expand=0)

fff = Image.new(‘RGBA‘,rot.size,(255,)\*4)

self.image = Image.composite(rot,fff,rot)

def randColor(self):

self.fontColor = (random.randint(0,250),random.randint(0,250),random.randint(0,250))

def randNum(self,bits):

return ‘‘.join(str(random.randint(0,9)) for i in range(bits))

def write(self,text,x):

draw = ImageDraw.Draw(self.image)

draw.text((x,4),text,fill=self.fontColor,font=self.font)

def writeNum(self):

x = 10

xplus = 15

for text in self.texts:

self.randColor()

self.write(text, x)

self.rotate()

x += xplus

return self.texts

def save(self):

self.image.save(‘captcha.jpg‘)

img = Captcha()

num = img.writeNum()

#img.image.show()

#img.save()

[pillow实例 | 生成随机验证码](http://www.bubuko.com/infodetail-641617.html)

标签：[算法](http://www.bubuko.com/so/1/%e7%ae%97%e6%b3%95)   [class](http://www.bubuko.com/so/1/class)   [style](http://www.bubuko.com/so/1/style)   [代码](http://www.bubuko.com/so/1/%e4%bb%a3%e7%a0%81)   [com](http://www.bubuko.com/so/1/com)   [log](http://www.bubuko.com/so/1/log)   [使用](http://www.bubuko.com/so/1/%e4%bd%bf%e7%94%a8)   [src](http://www.bubuko.com/so/1/src)   [问题](http://www.bubuko.com/so/1/%e9%97%ae%e9%a2%98)

原文：http://www.cnblogs.com/meelo/p/4298579.html