

# 线框图质量自动检测算法（SketchChecker）

By 乐信产学研

## 1. 运行环境

Python 3.7

## 2. 需要的库

PIL

OpenCV-Python

Scikit-image

PyMuPDF (用于处理 Pdf 文件的读取)

Fitz (用于处理 pdf 文件的读取)

## 3. 主程序

Main.py

该文件通过首先读取指定文件夹下的所有 PDF 文件，对于每一个 pdf 文件，转换为用于后续处理的 PNG 图像，并送入线框质量检测程序进行细线化检测和断点检测，最终返回标记了细线区域和断点位置的结果图。

默认程序会在 PDF 文件所在的文件夹创建 result 目录，并在该目录中创建 broken\_line\_result 和 thin\_line\_result 子文件夹，分别保存断点和细线检测结果图。以断点检测为例，每个结果文件以原文件命名，并在原文件名后面加有以数字表示的检测到的断点个数。

另外，在 main\_process.py 文件中，ENABLE\_UNCLOSED\_LINE 和 ENABLE\_THIN\_LINE 两个变量，控制了是否开启断点检测和细线检测功能。

## 4. 细线检测

默认检测线框图中少于 2 pixel 宽度的细线。算法对线框的粗细进行了分类，一般来说，标记了红色的线框比粉色的更细。对于检测结果，可能存在如下情况：

1) 可能线段中间有比较细的部分，或者两端比较细，甚至中间有可能存在断裂，也会被检测出，如下所示：

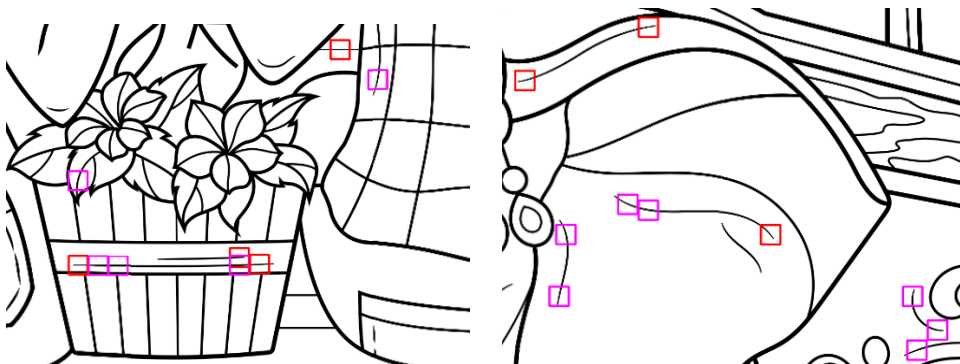


图 1 中间或者两端比较细，也会被检测到

2) 有些图中可能存在很小很细的圆，可能会被漏检

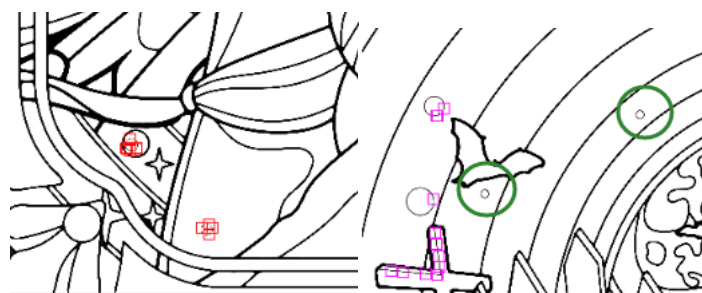


图 2 可能会被忽略的小圆圈。左图为检测到的两个细线圆，右图绿色圆圈表示的是被忽略掉的两个很小的细线圆

3) 一般只需要看那种连成长线的，中间或者两端比较细检测出来的，可以忽略。但是要注意部分比较短的线可能会被遗漏。

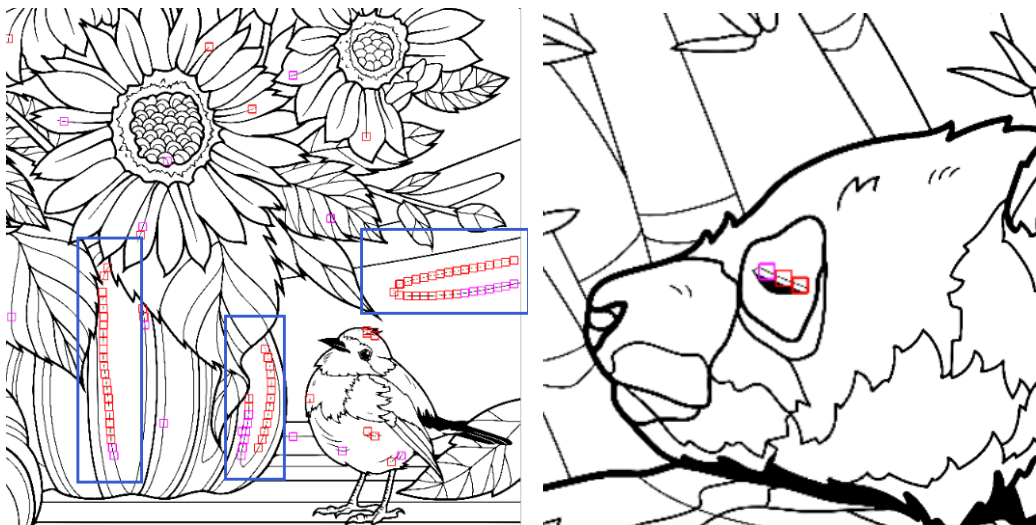


图 3 重点关注连成线的部分。当然，对于某些图，要注意短线区域，如右图所示

## 5. 断点检测

默认检测断开部分相距小于等于 10 个像素左右的断点区域。有些段开的地方，相距较远，不会被检测到。因为相距较远，一般是设计师有意为之。对于断点检测，需要注意的是：可能存在因为线非常细而导致误检为断点的。

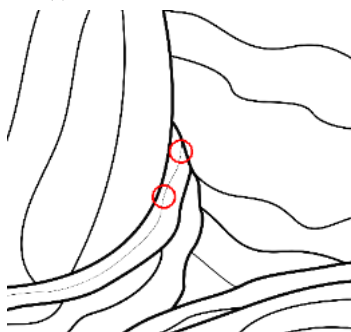


图 4 因为线段过细造成的断点误检

## 6. 其它问题

1) 基于我们自己跑的数据，细线检测和断点检测的准确率是非常高的，但是不能保证 100%都检测到。主要是为了去除很多假阳性的点，加入了深度学习模型。

该模型能去除一半以上的假阳性，但仍然存在很小的概率，将可能的真阳性点排除。这点需要注意。尽管如此，我们依然发现，现有的算法能检测到一些设计师都漏检的点。

2) 正方形的图，pdf 中原始大小多为 2048，比较固定，问题不大。对于墙纸（长方形）图，pdf 中原始大小可能是 750，也可能是 1500，似乎画师对于这个并不统一。我们在程序中输出了 pdf 中线框图的原始大小，如图 5 所示。大部分是 750，但是我们也发现少数图是 1500 的。由于分辨率差了近 1 倍，因此在细线检测时，可能会有存在一定问题。需要进一步观察。

```
start to convert pdf to png image  
pdf content size = Rect(0.0, 0.0, 2048.0, 2048.0)  
当前正在处理 32/34 :复杂5.png
```

图 5 程序中输出了 pdf 中线框图的原始大小