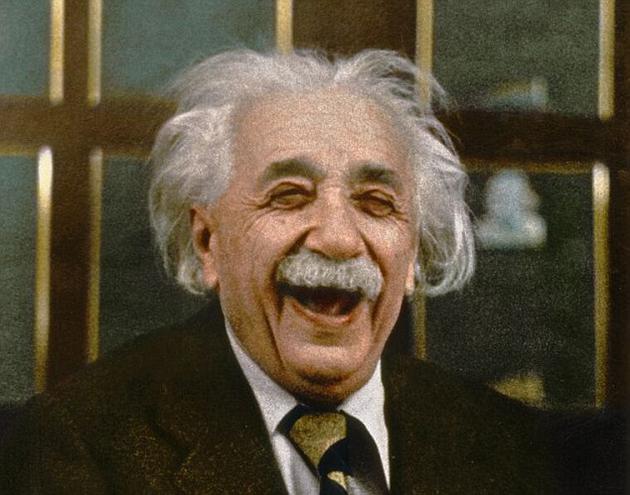
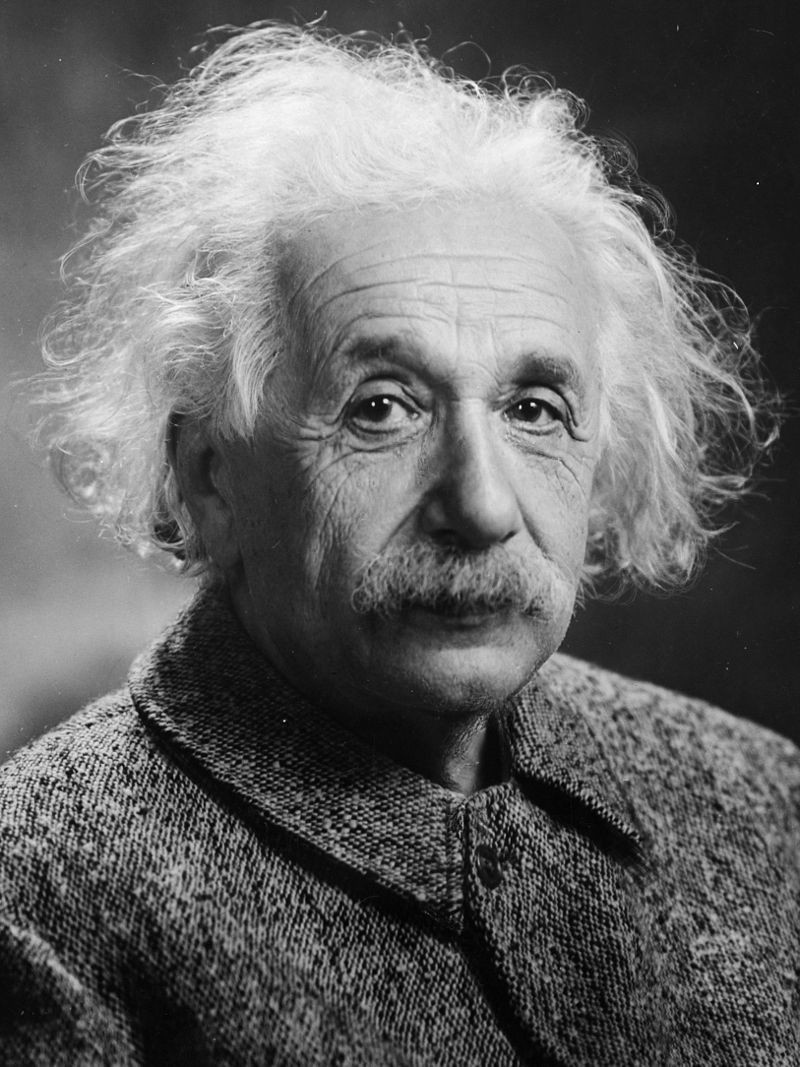
# 如何用墙画机画爱因斯坦

第一步：Google一张爱因斯坦的照片下载下来。



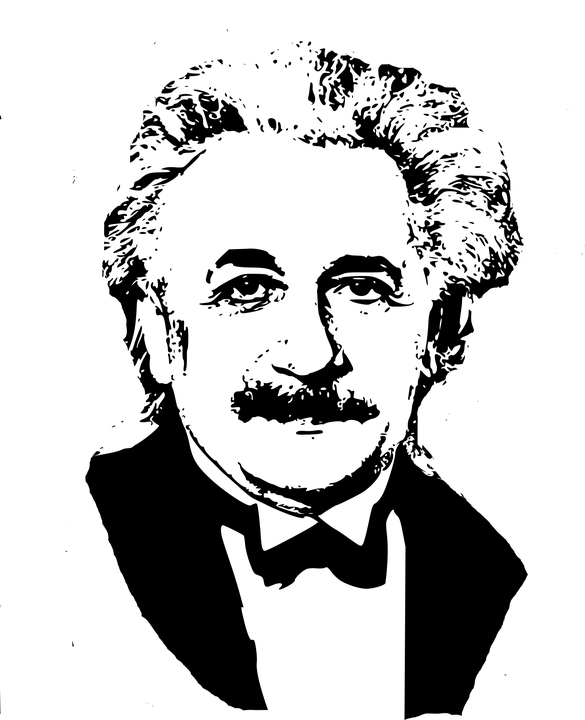
遇到了障碍，我们只有一直笔，虽然是彩色笔，但是依然只能绘制一种颜色，所以我们需要找一张单色的照片才行。

第二步：重新Goolge一张单色的爱因斯坦

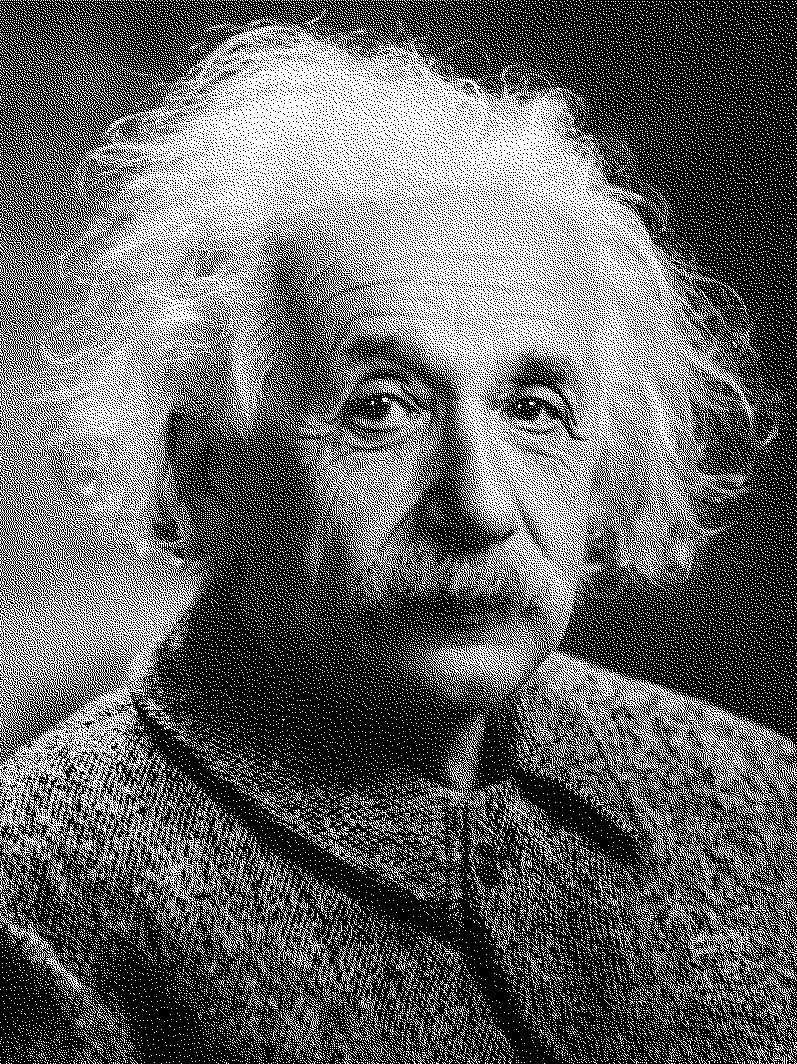


又遇到了困难，虽然照片是黑白的了，但是照片里依然有很多个不同的灰色，深灰，浅灰，深深灰，浅浅灰。。。。我们的笔，显然还做不到不同深度的笔触。所以我们需要一张纯黑白的不要灰的照片。

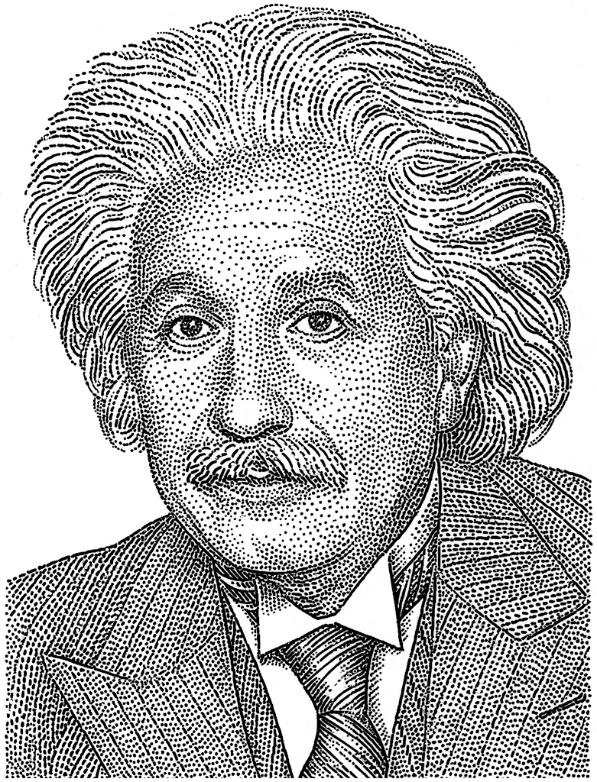
第三步：继续Google一张黑白的爱因斯坦。



好了，找到了完美的黑白爱因斯坦。等等，好像哪里不对？这个人好像是Godfather。继续寻找。



我要是会PS就不用这么麻烦了，想想中应该是这样的，可惜不会P继续找。

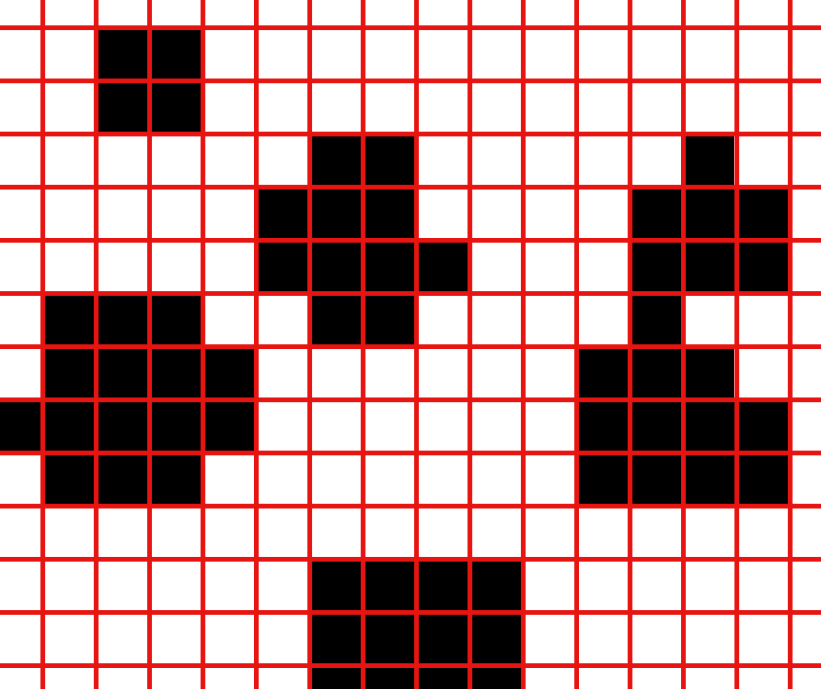


完美地找到了一个黑白的爱因斯坦。

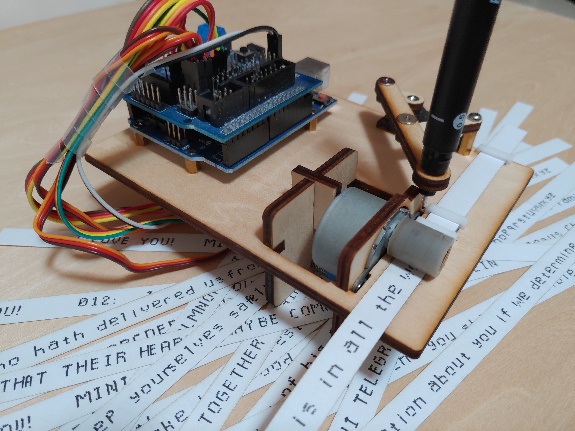
第四步，图片是点阵图，需要转换成矢量图才能绘制。

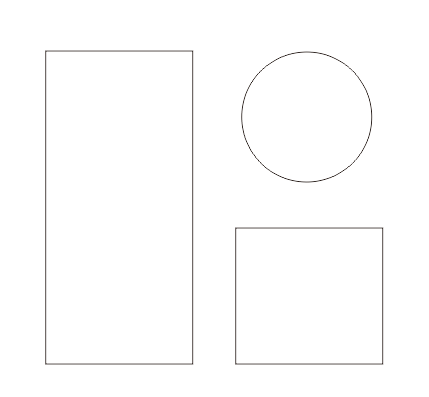


什么是点阵图，放大了看就是由一个一个点组成的图像。



再放大，每个格子是一种颜色，由很多很多个格子组合在一起拼成一个图案。上面这个简单的爱因斯坦是由横781\*竖1024一共799744个黑白格子组成的。黑色点大约是23.1万个。我们可以知道每个黑色点的位置，然后用笔在纸张上打点就可以了。每个点1秒就需要63个小时完成，似乎有点慢。说到打点可以看看我们店里的电报机<https://item.taobao.com/item.htm?spm=a1z0d.6639537.1997196601.237.154c7484xanMSf&id=576633734176>，那可是专业打点的机器。

[](https://item.taobao.com/item.htm?spm=a1z10.1-c.w4004-1266050943.8.37833c6eJ5we2P&id=576633734176)显然打点不适合我们的拉线墙画机，它是绘制矢量图的神器。那啥是矢量图呢？

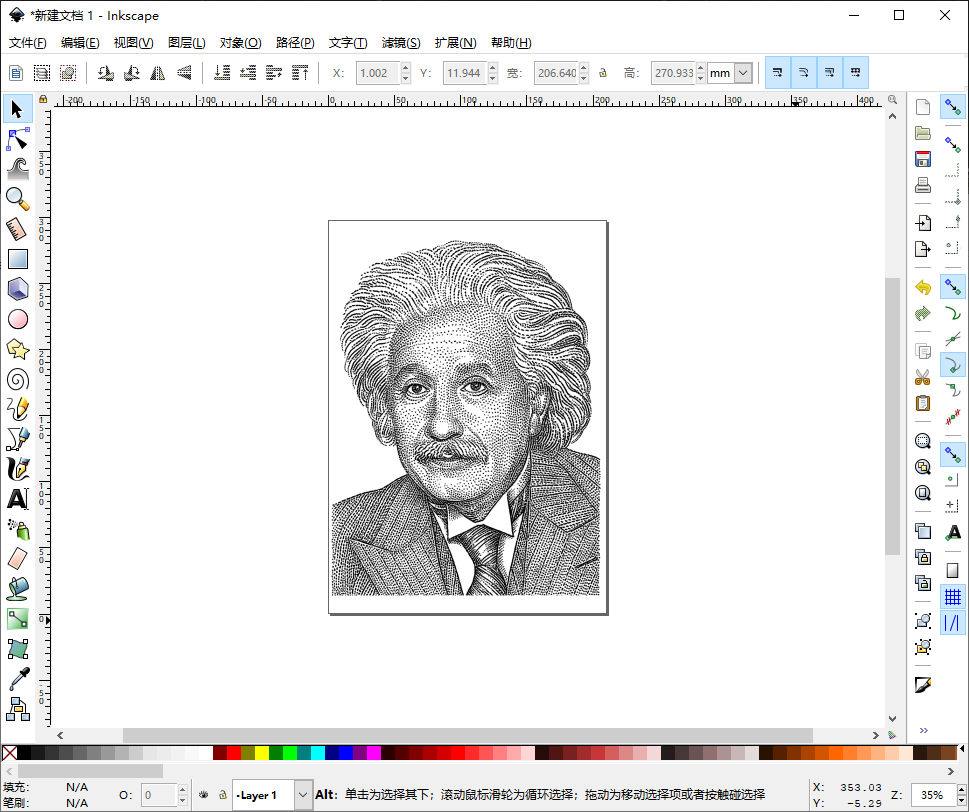


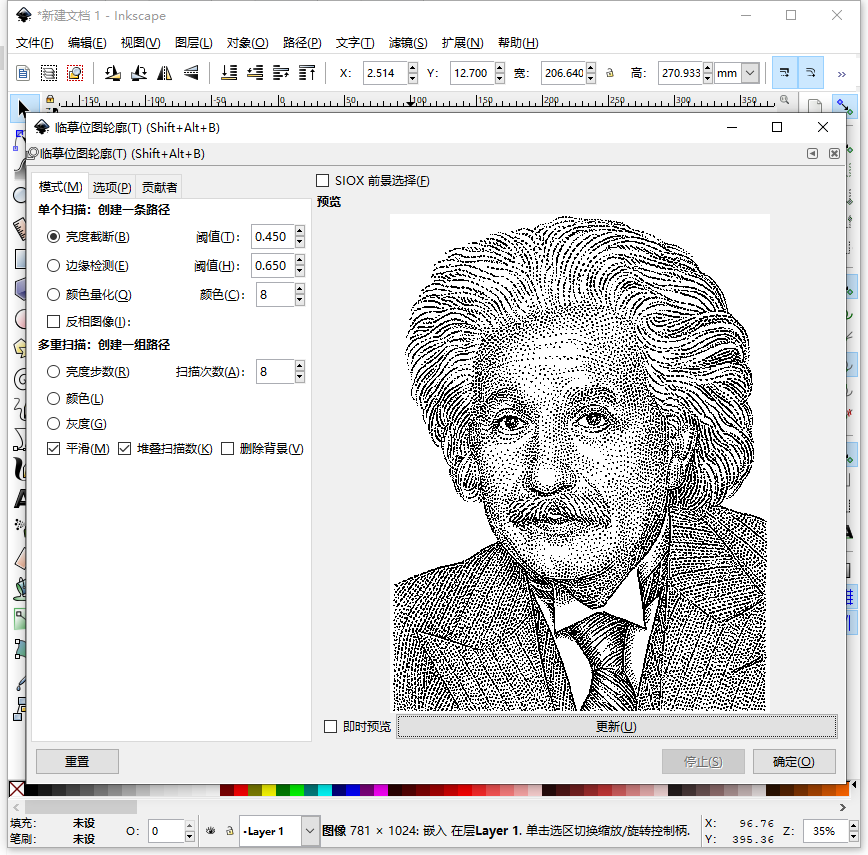
一些图形或者线条，用数字记录该图形的位置，大小，颜色，线条粗细等信息，每次看到的图，都是由软件计算出来并且绘制的图（它们在屏幕上最终还是以点阵图的形式显示出来）。这种图的好处是可以任意放大，无论多大都不会产生点阵图中的那种方块点。坏处是只能表现相对简单的图案。



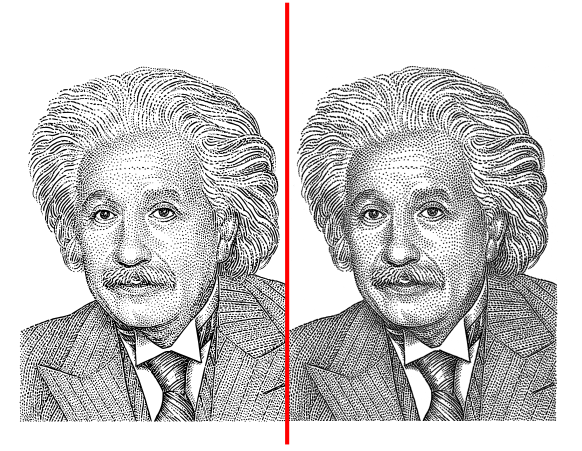
打脸了，这个也是矢量图。还由大神用Excel徒手画变形金刚呢。这些都是大神非人类。补一句，矢量图最擅长曲线，可以秒杀点阵图。上图中每一缕头发都是一个曲线，再加上不同的颜色可以表现出惊人的细节。

拉回到爱因斯坦，如何把点阵图变成矢量图呢？答，可以，但不是所有图都可以。有很多种软件都可以实现这个功能。我们讲一个免费软件Inkscape。

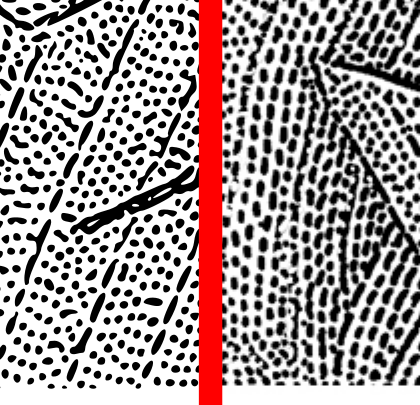




操作很简单，拖爱因斯坦进去，选择爱因斯坦，路径->临摹位图轮廓，确定。



几乎无法区分谁是位图，谁是矢量图。



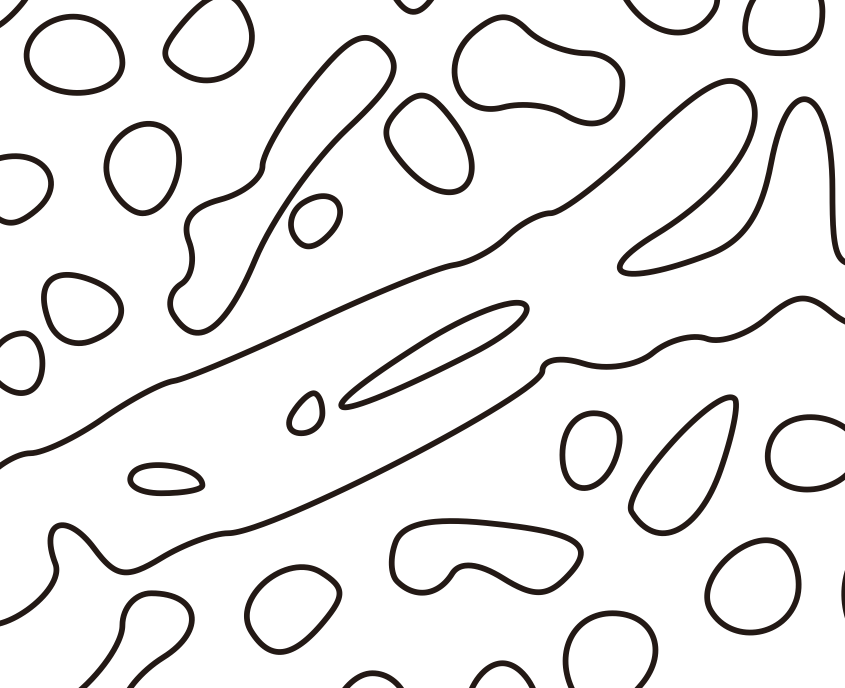
放大到200%就可以看出一些端倪了。



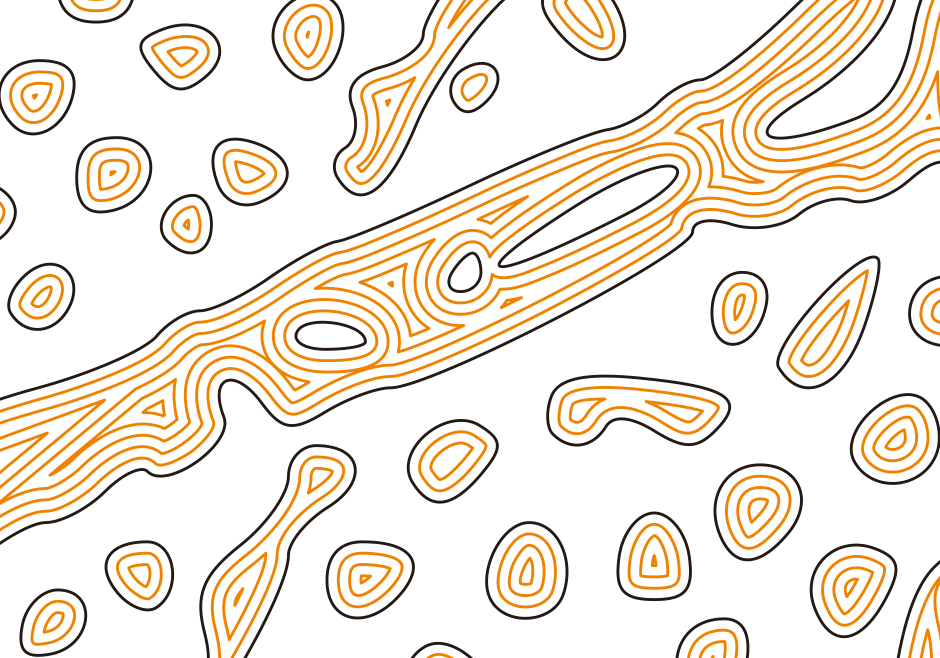
800%的时候，点阵图已经惨不忍睹了。



矢量图还可以继续毫无压力的放大，再放大。



如果我们忽略填色的问题，图形就是这样了。我们让笔沿着这些路径走一下，就可以画好轮廓，需要填色的部分，再在里面多走几趟就像小朋友画画一样，就搞定了。理论上的确是这样操作的。只是还需要一些转换步骤，第一是把轮廓告诉墙画机，让它也沿着轮廓的路径开动，第二在需要填色的地方来回的走。

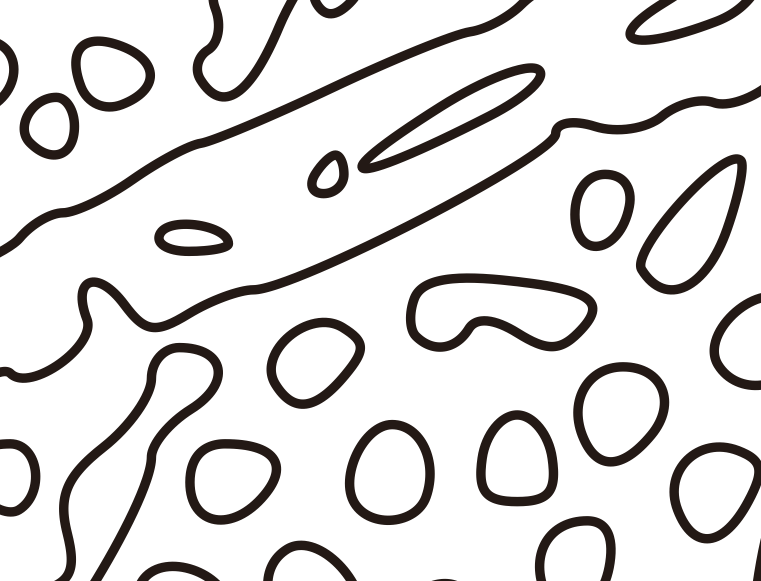


黑色勾边，黄色涂实就，因为笔尖还有个粗细的问题，只要线够粗，就会变成实色的图案了。



可怜我的密集恐惧症啊！！！！

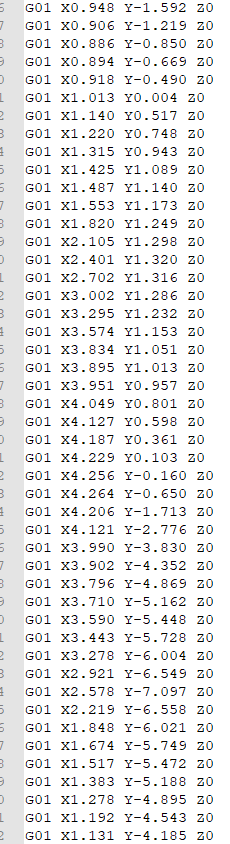
实际上这2个步骤没有太大的本质区别，都是让墙画机走一个特定的路径。所以合并同类项，忽略第二步。我们只先搞一个轮廓就好了。主要是为了我的密集恐惧症否则就不能继续了。



接下来，需要搞一个叫做G-code的东西，就是将图案中的路径，转换成一大串位置坐标，让机器再沿着这些坐标一点一点移动就可以加工（G-code是国际上最广泛使用的数控编程语言）出我们需要的图案了。通俗点说，就是告诉机器从坐标xy移动到x1,y1,x2,y2.....Xn,Yn。是不是觉得哪里不对？横平竖直的直线还可以理解，曲线是如何描述的？这里好像又变成了点阵图吧？没错，G-code就是点阵图。只是有小数点而已。在gcode里没有曲线，任何曲线都被拆分成了小的直线线段。



就连这种德国的5轴数控加工中心也是用的gcode，各种小直线！毁三观么？实际上gcode的计算也是一种非常复杂的处理，他需要计算各种电机的加速，减速，惯性，被加工工件的材质，刀具的尺寸等等等因素。精密的加工中心可以达到微米级的移动控制，还需要什么曲线。。。

拉回来，我们到毫米级的精度就ok了。（实际上我们的理论分辨率可以达到0.027mm，即使实际精度误差1个数量级也可以达到0.27mm的分辨率相当惊人了）画个爱因斯坦而已，扯远了。

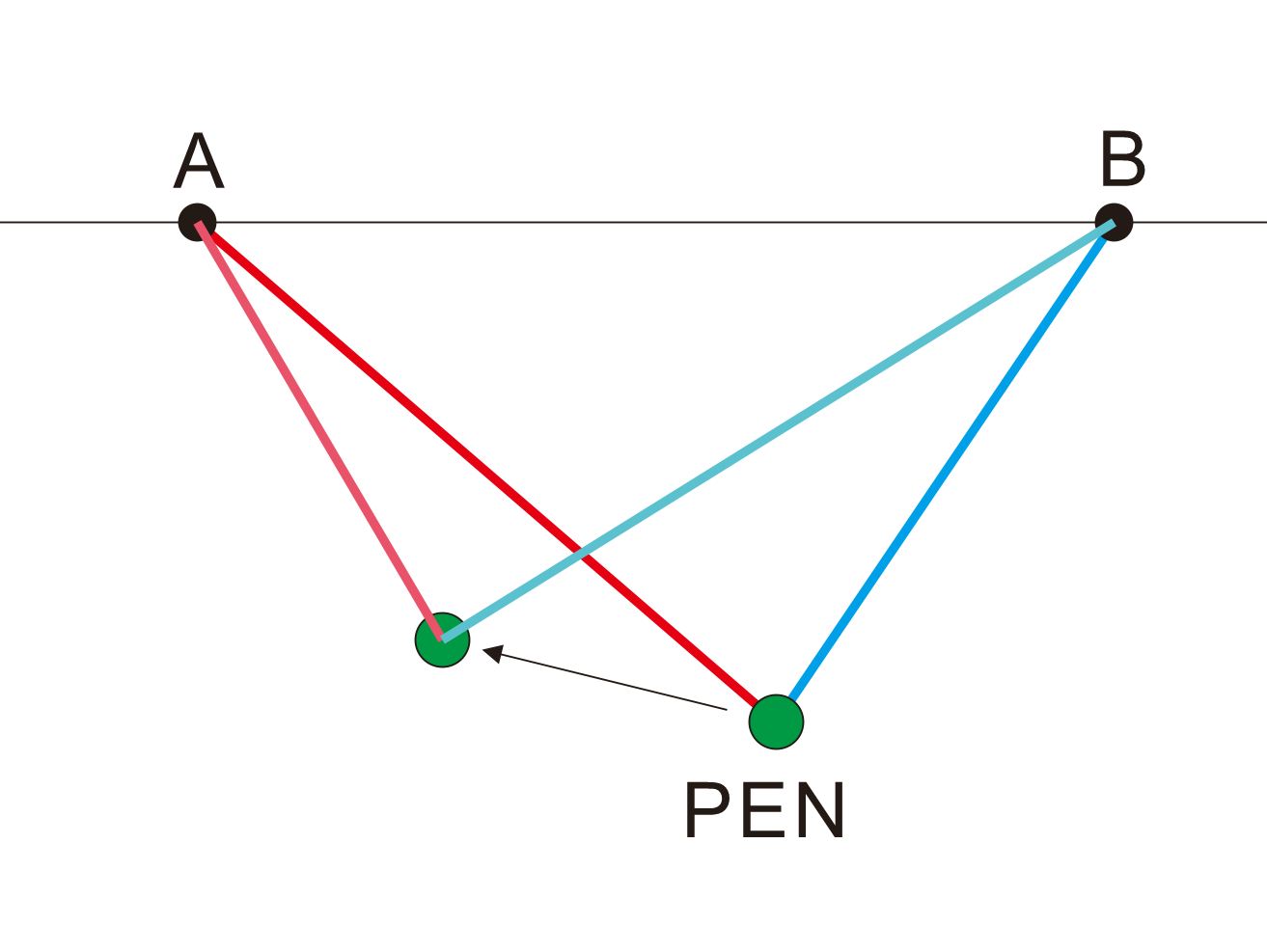
Inkscape可以将矢量图制作成gcode保存，方法请Google一下很多，实在不行百度也凑合了。此处略。

如上我们已经将一张照片，完全变成了，一大串位置坐标。我们只要可以让我们的设备，准确的走到我们指定的位置，然后下一个位置，再下一个，就OK了。

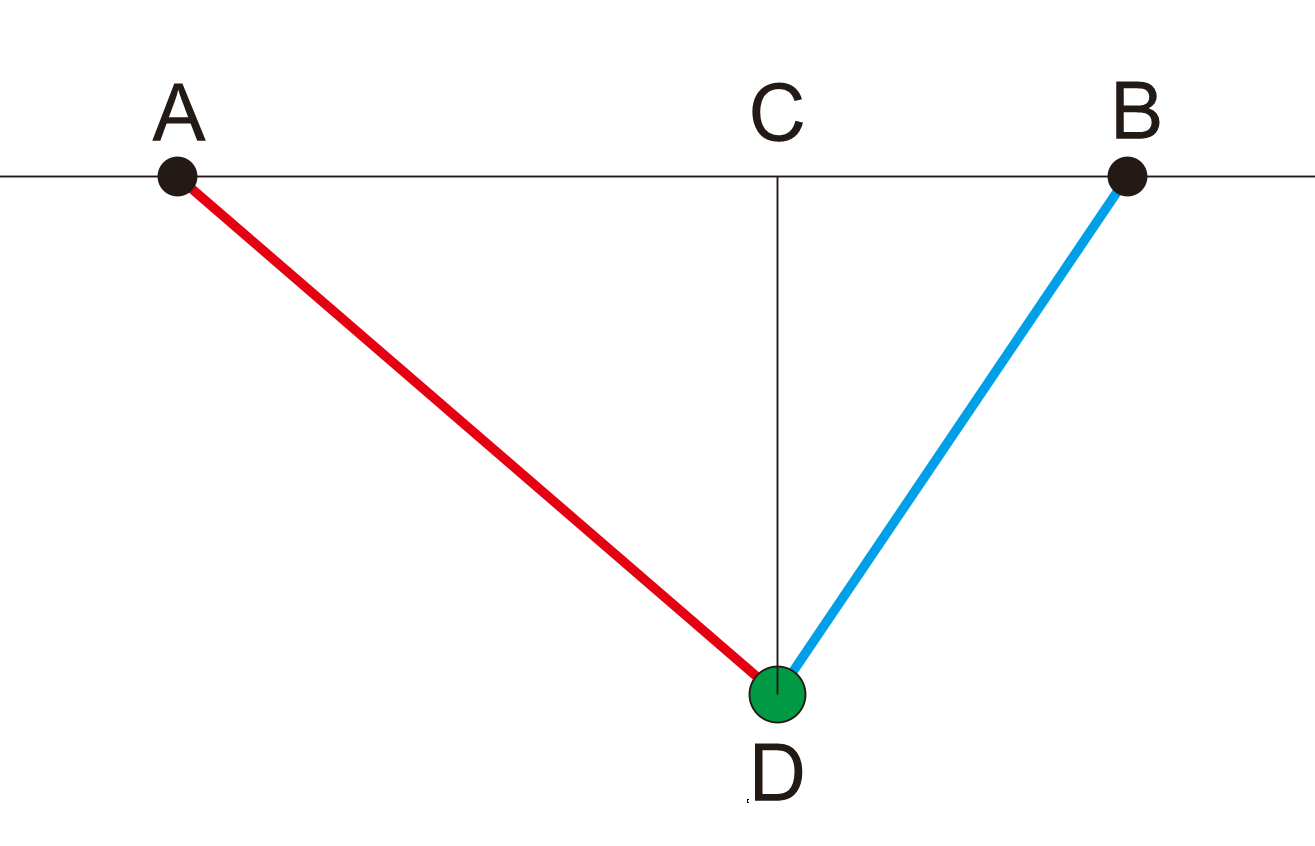
如何走到一个预定的位置呢？我们可以来研究一下主题了吧。



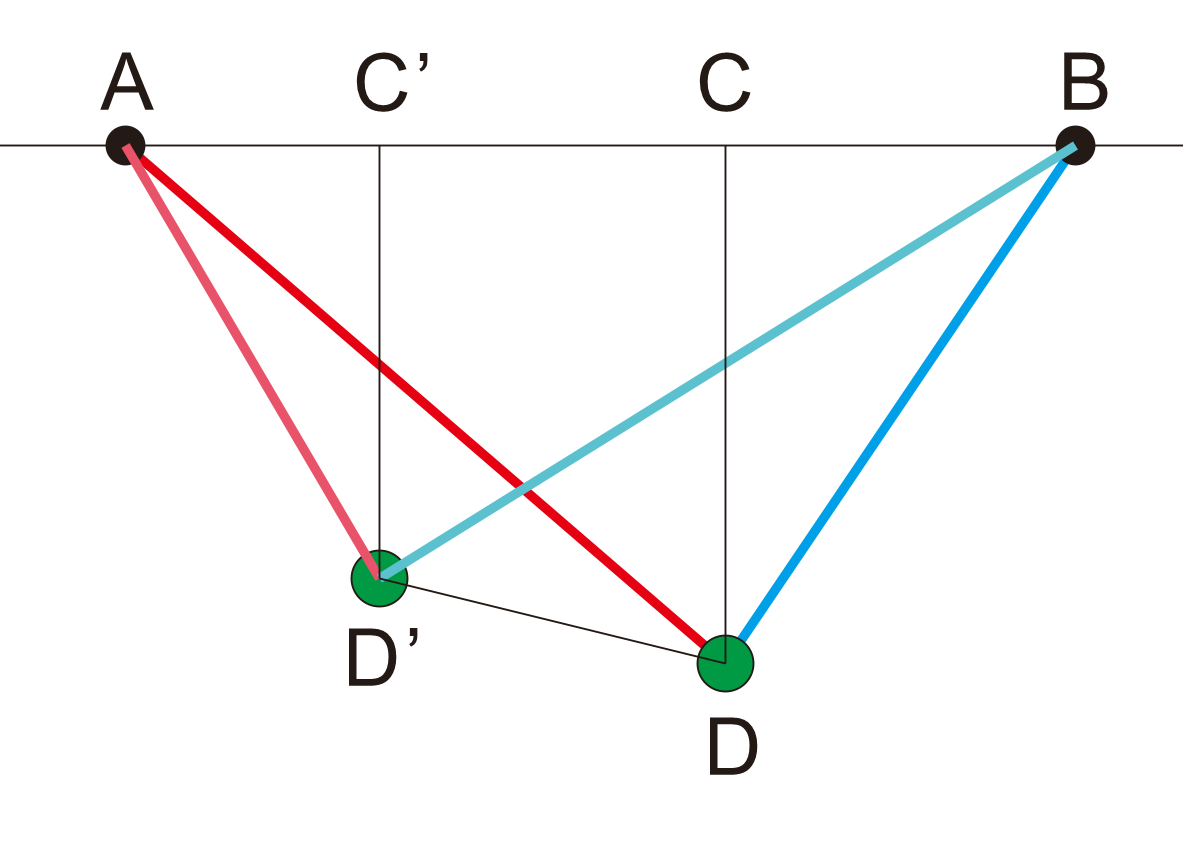
步进电机，请Google百度一下。简单说，可以旋转的电机，可以精确控制旋转角度的电机。普通电机通电就转，转了多少，只能估算，不能控制，停电后不能保持，有大负载会被拉动。而步进电机不但可以精确控制依什么速度转多少角度（工业级步进电机角度最小可以达到0.9°，加减速齿轮的可以达到更高的精度），而且在不转的时候，还可以有一定的力道保持不动（负载超过还是可能被转动）。我们使用的是28BYJ的步进电机，自带减速齿轮。大部分空调的导风板都是用这种电机拉动的。接好线我们用程序就可以控制它转。无需外接驱动器，电源等一大堆莫名其妙的东西。更不需要你了解步进电机的细分、距角。。。等莫名其妙的属于。一行程序 m1. moveRelativeInSteps (1)，转一下。循环2048次，转一圈。如果用35mm的线轴，转一步，可以拉动0.027mm的线。步进电机就算讲完了。



接下来看看两条线是如何拉动笔尖的。两个步进电机在AB点，电机转动线轴，收放两条线，笔就会跟着移动。只要让A电机拉线，B电机放线，笔就可以移动到要的位置。线需要收多少，放多少，也很简单，三角函数了解一下。



以知线段AB，CD的长度，求AD，BD？忘了的请找个15岁的骚年问一问。因为我也忘了。反正可以算。



要从D移动到D’的位置，只需要计算出AD和AD’的差，BD和BD’的差，然后正转或反转两个电机就OK了。

然后我们只要读Gcode，一行一行的按坐标移动笔，一下一下的就可以把马龙.白兰度画出来了。

