

计算机视觉第四次作业

朱明杰 15331441

一. 实验任务

1. 输入普通 A4 打印纸，上面可能有手写笔记或者打印内容，但是拍照时可能角度不正。输出已经矫正好的标准普通 A4 纸（长宽比为 A4 纸的比例），并裁掉无用的其他内容，只保留完整 A4 纸张。
2. 输入两张图像，输出一系列渐变的图像。（附加题）

二. 实验工具

Visual studio 2015

Clmg Library

三. 算法流程

1. Canny 算子。
直接利用上次作业所实现的 Canny 算子代码即可。
2. Hough 变换。
直接利用上次作业所实现的 Hough 变换代码即可。
3. 局部极大值的寻找。
直接利用上次作业所实现的基于最小生成树的聚类代码即可。
4. 角点的求法。
直接利用上次作业所实现的 Cramer 法则即可。
5. 确定角点的方位。
只需判断角点与图像四角的距离即可。
对于一系列角点 (x_i, y_i) ，左上角点为

$$\arg \max_i (x_i - 0)^2 + (y_i - 0)^2$$

类似可以确定其他三个方位的角点。

结果可能会导致宽比高更大的情况。虽然按照老师的原意，这是不允许出现的情况。但是也不能排除那些喜欢把 A4 纸横放着写东西的人啊对不对？所以我觉得拍照所放置的位置。如果之后要进一步处理的话，可以利用线性变换构成群的性质，将之后的结果旋转 90° 即可（注意变换应用的次序，因为线性变换并不构成阿贝尔群！）。

6. 投影变换矩阵的求法。
至此，我们一共获得了四个角点（八个参量可以构成八个方程），需要求投影变换矩阵

$$\mathbf{A} = \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{pmatrix}$$

的八个参数。譬如，对于左上角点 (x, y) ，其是由摆正后的左上角点 $(1, 1)$ 的齐次坐标

经过左乘投影变换矩阵得到。注意我们这里用(1,1)而非(0,0)来表示左上角点，这是因为用(0,0)表示左上角点的话会得到退化的线性方程，从而可能会导致系数矩阵不满秩。而且正如我们上面已经提到过的，所有线性变换构成一个群，所以我们先应用平移变换再做投影变换肯定也对应着另一个投影变换。对于左上角点，我们可以依据下列方程组：

$$\begin{cases} \frac{a+b+c}{g+h+1} = x \\ \frac{d+e+f}{g+h+1} = y \end{cases}$$

经过恒等变形：

$$\begin{cases} a+b+c-xg-xh = x \\ d+e+f-yg-yh = y \end{cases}$$

写成矩阵形式：

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & -x & -x \\ 0 & 0 & 0 & 1 & 1 & 1 & -y & -y \end{pmatrix} \begin{pmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix}$$

还有三个点可以构成剩下的六个方程，拼到下面即得到了我们的最终的方程组。

值得注意的是，Clmg 提供了 `solve()` 函数，能够让我们便利地解出线性方程组的解。但是需要注意的是，Clmg 的矩阵行列需要调整一下，总之要先跑两个 demo 才知道这个坑。

7. 反向映射与双线性插值。

由于上面我们是求的从摆正后的点到摆正前的点的变换矩阵，所以这个本身就是反向映射矩阵了。至于双线性插值，Clmg 提供了一个叫做 `linear_atXY()` 的函数，直接拿来用就是。

8. 附加题：暂时先鸽一下。

四. 算法代码 (Clmg) 实现

见压缩包下代码。

五. 实验结果

见工程文件下的 Dataset/myresult 中的结果。

六. 分析与评价

需要指出的是，这一套模型对参数的要求是很高的。我们在 canny 算子的时候把阈值卡的非常高，这样带来的坏处是相机水印的梯度可能比 A4 纸的边缘的梯度还要更加显著。所以该模型要求相机不能开水印，或者是先通过 Inpaint 技术把水印去掉。