# 人脸图像的裁剪与矫正说明文档

一、数据来源

saber.jpeg

二、算法说明

1、加载图片

2、通过人机交互获取到图像中眼睛中心和嘴巴中心的坐标

3、通过获取到的坐标和规定要转换到的坐标点解出仿射变换矩阵

4、创建出背景

5、让原图片每一个坐标点都乘以仿射变换矩阵

6、判断点是否符合要不符合着舍去，符合着通过近邻法灰度插值来讲对应的像素点插入相应的位置，执行到最后一个像素点后终止。

7、保存图像

三、代码实现

#读取图片

img = cv2.imread("saber.jpeg", cv2.IMREAD\_GRAYSCALE)

#对读入的图片进行展示

cv2.imshow("image", img)

#计算仿射变换模型参数

X = np.mat(weizhe)

U = np.mat([[24,25],[76,25],[50,75]])

M = np.array((X.T\*X)\*\*-1\*X.T\*U).T

#定义鼠标事件进行人机交互

on\_EVENT\_LBUTTONDOWN(event, x, y, flags, param):

#进行仿射变换，和灰度值插法（采用最近灰度值法），返回一张处理后的图片

def warpAffine(img, M):

参数：一张图片，和仿射矩阵

#保存图片

cv2.imwrite('test.png', img\_transformed)