智能医学数字图像处理实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 学号 | 19084127 | 班级 | 190841 |
| 姓名 | 周亚诺 | 专业 | 生物医学工程 |
| 实验  名称 | 实验6：医学图像分割 | | |
| 实验  目的  和  内容 | **实验目的和要求：**  掌握传统基于边缘检测的图像分割⽅法  **实验内容：**  python实现基于边缘检测的图像分割 | | |
| 实验  结果  与  分析 | 1. 原图 2. 灰度图 3. 高斯平滑 4. Roberts算子 5. Prewitt算子 6. Sobel算子 7. Laplace算子 8. Canny算子 | | |
| 实验  代码 | import cv2 import numpy as np  if \_\_name\_\_ == '\_\_main\_\_':  # original input  img = cv2.imread('./Lena.png')  cv2.imwrite('./out/origin.png', img)   # need gray image not rgb  gray\_img = cv2.cvtColor(img, cv2.COLOR\_RGB2GRAY)  cv2.imwrite('./out/gray\_img.png', gray\_img)   # gauss smoothed  gaussian\_img = cv2.GaussianBlur(gray\_img, (3, 3), 0)  cv2.imwrite('./out/gaussian\_img.png', gaussian\_img)   # roberts operator  kernel\_x = np.array(  [  [-1, 0],  [0, 1]  ],  dtype=int  )  kernel\_y = np.array(  [  [0, -1],  [1, 0]  ],  dtype=int  )  x = cv2.filter2D(gaussian\_img, cv2.CV\_16S, kernel\_x)  y = cv2.filter2D(gaussian\_img, cv2.CV\_16S, kernel\_y)  absX = cv2.convertScaleAbs(x)  absY = cv2.convertScaleAbs(y)  roberts = cv2.addWeighted(absX, 0.5, absY, 0.5, 0)  cv2.imwrite('./out/roberts.png', roberts)   # prewitt operator  kernel\_x = np.array(  [  [1, 1, 1],  [0, 0, 0],  [-1, -1, -1]  ],  dtype=int  )  kernel\_y = np.array(  [  [-1, 0, 1],  [-1, 0, 1],  [-1, 0, 1]  ],  dtype=int  )  x = cv2.filter2D(gaussian\_img, cv2.CV\_16S, kernel\_x)  y = cv2.filter2D(gaussian\_img, cv2.CV\_16S, kernel\_y)  absX = cv2.convertScaleAbs(x)  absY = cv2.convertScaleAbs(y)  prewitt = cv2.addWeighted(absX, 0.5, absY, 0.5, 0)  cv2.imwrite('./out/prewitt.png', prewitt)   # sobel operator  x = cv2.Sobel(gaussian\_img, cv2.CV\_16S, 1, 0)  y = cv2.Sobel(gaussian\_img, cv2.CV\_16S, 0, 1)  absX = cv2.convertScaleAbs(x)  absY = cv2.convertScaleAbs(y)  sobel = cv2.addWeighted(absX, 0.5, absY, 0.5, 0)  cv2.imwrite('./out/sobel.png', sobel)   # laplace operator  dst = cv2.Laplacian(gaussian\_img, cv2.CV\_16S, ksize=3)  laplace = cv2.convertScaleAbs(dst)  cv2.imwrite('./out/laplace.png', laplace)   # canny operator  canny = cv2.Canny(gaussian\_img, 100, 200, 5)  cv2.imwrite('./out/canny.png', canny) | | |
| 成绩  评定 | 教师签名：  年 月 日 | | |