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3：图像空域增强 | | |
| 实验  目的  和  内容 | **实验目的和要求：**  1.掌握图像空域变换的⽅法和原理  1.掌握空域滤波中常用的平滑滤波⽅法和原理  2.掌握空域滤波中常用的锐化滤波⽅法和原理  **实验内容：**  1.实现图像的算术运算和逻辑运算的代码  2.实现以下常用平滑滤波⽅法的代码：  （1）图像均值滤波  （2）图像中值滤波  （3）图像⾼斯滤波  3.实现以下常⽤锐化滤波⽅法的代码：  （1）roberts算⼦  （2）prewitt算⼦  （3）sobel算⼦  （4）拉普拉斯算⼦ | | |
| 实验  结果  与  分析 | A: 原图  B：灰度图  C：降低50亮度  D：灰度反转  E：均值滤波  F：中值滤波  G：高斯滤波  H：roberts算子  I：prewitt算子  J：sobel算子  K：拉普拉斯算子 | | |
| 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)   # dim by 50  dim = cv2.subtract(  img,  np.ones(  img.shape,  dtype='uint8'  ) \* 50  )  cv2.imwrite('./out/dim50.png', dim)   # logical not  # need gray image not rgb  gray\_img = cv2.cvtColor(img, cv2.COLOR\_RGB2GRAY)  cv2.imwrite('./out/gray\_img.png', gray\_img)  gray\_not = cv2.bitwise\_not(img)  cv2.imwrite('./out/gray\_not.png', gray\_not)   # mean filtering  mean\_filtered = cv2.blur(gray\_img, (10, 10))  cv2.imwrite('./out/mean\_filtered.png', mean\_filtered)   # medium filtering  medium\_filtered = cv2.medianBlur(gray\_img, 3)  cv2.imwrite('./out/medium\_filtered.png', medium\_filtered)   # gauss filtering  gauss\_filtered = cv2.GaussianBlur(gray\_img, (7, 7), 1.8)  cv2.imwrite('./out/gauss\_filtered.png', gauss\_filtered)   # 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(gray\_img, cv2.CV\_16S, kernel\_x)  y = cv2.filter2D(gray\_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(gray\_img, cv2.CV\_16S, kernel\_x)  y = cv2.filter2D(gray\_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(gray\_img, cv2.CV\_16S, 1, 0)  y = cv2.Sobel(gray\_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(gray\_img, cv2.CV\_16S, ksize=3)  laplace = cv2.convertScaleAbs(dst)  cv2.imwrite('./out/laplace.png', sobel) | | |
| 成绩  评定 | 教师签名：  年 月 日 | | |