### 

****

**数字图像处理实验报告**

**实验1 边缘检测**



**学 院 智能与计算学部**

**专 业 计算机科学与技术**

**学 号 3022206045**

**姓 名 陆子毅**

**一、实验内容与要求**

使用课上所讲的边缘检测算法对自选图像进行边缘检测处理，要求图像不少于三张，从个人相册或其他来源选取，不能全选取经典图像，至少包含一张实验者本人的照片。

结合算法原理，对比检测结果，讨论各算法的优缺点。

讨论为何FDoG可能产生宽度不同的线条，FDoG是否可以产生宽度为1的线条，如何实现。

**二、实验目的**

理解边缘检测的基本概念和原理。

掌握几种常见的边缘检测算法，如Sobel、Prewitt、Canny等。

学会使用图像处理软件或编程语言实现边缘检测。

分析不同边缘检测算法的优缺点及其适用场景。

**三、实验环境**

matplotlib==3.6.2

numpy==1.23.5

opencv-python==4.6.0.66

opencv-python-headless==4.1.2.30

**四、实验结果**

**4.1 实验效果分析**

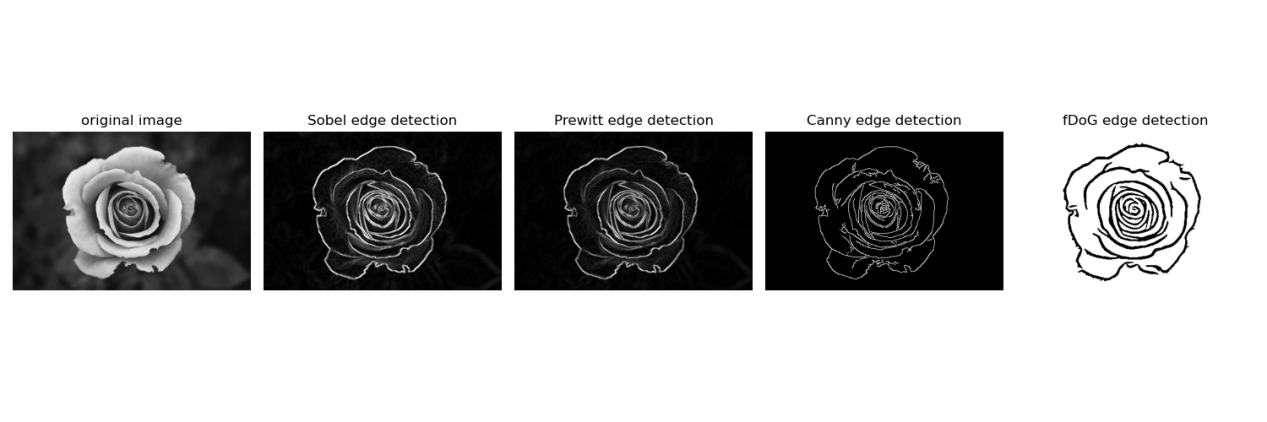


图1

对于图1而言，四种边缘提取算法的效果都比较好，Sobel和Prewitt会出现一些非边缘特征，将花瓣的纹路也当作边缘特征给提取出来了，边缘的粗细不一，深浅不一。Canny相较于前两种算法能够清晰地提取出花瓣的边缘特征，但是边缘比较杂乱，图像的明暗条件干扰了算法对于边界的判断。fDoG算法提取的边缘相较于前三种算法在准确性和清晰度上都表现不错，但是提取得到边缘线太粗，并且边缘线不平滑。

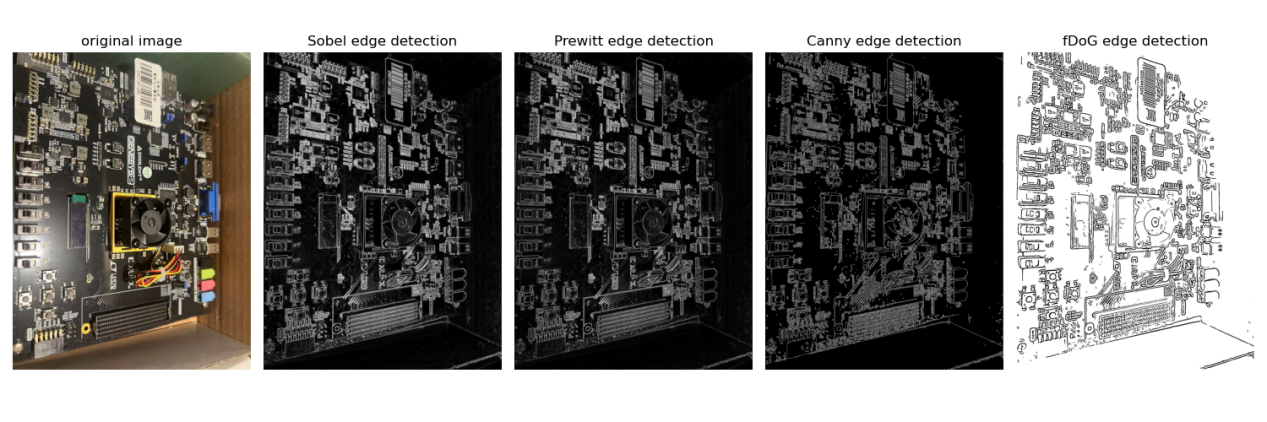
****

图2

图2 拍摄的是电路板，可以看到在细节较多的情况下，Canny的表现不如其他几种算法，Sobel和Prewitt的边缘提取结果相似，但是相比较而言，Sobel提取出来的边缘要更加清晰一点，fDoG提取出了大部分的边缘特征，但是相比较于其他几种算法，对于细节部分的边缘特征丢失比较多。而且边缘线条也相对较粗。

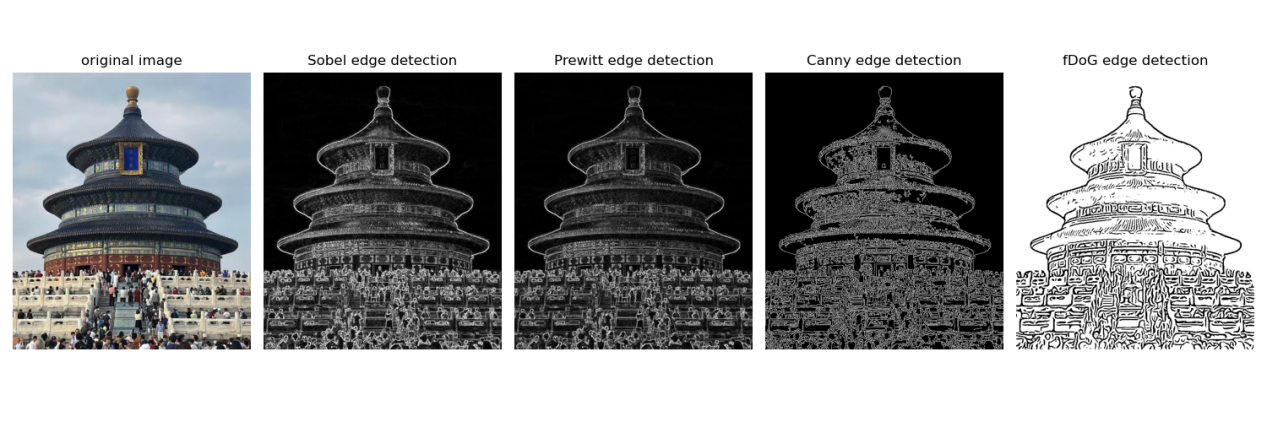


图3

图3是之前拍摄的天坛的图片，中式建筑具有比较多的纹理细节，首先可以看到，在处理非常多的纹理细节的时候，Canny提取出来的细节部分比较杂乱，并且由于使用的是同一明度的线条，使得具有很多细节的图像边缘看上去会比较杂乱，对比fDoG的边缘提取，fDoG没有将所有的细节都保留下来，只保留了一些比较明显的边缘特征，对于细节部分，fDoG的处理不是特别好。对于Sobel和Prewitt而言，使用梯度的方式是这两种算法能够将几乎所有的边缘信息给保留下来，最完整地保留了原有图像的边缘信息。

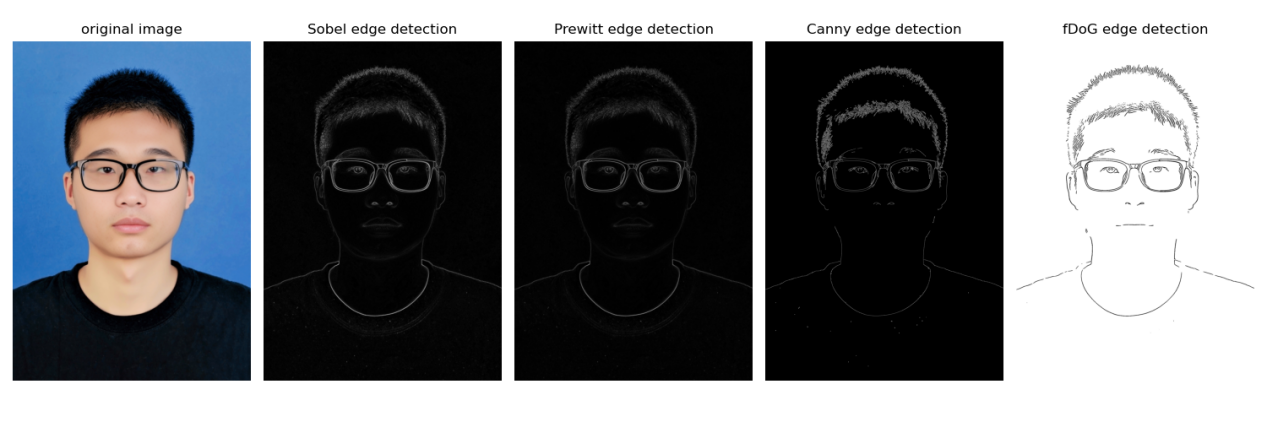
****

图4

本人照片，在图片色块区分比较明显的状况下，Canny将头发的边缘细节清晰的提取了出来，Soble和Prewitt还是利用梯度将所有的边缘信息都保留了下来，但是发现Sobel和Prewitt的边缘会出现不清晰以及双重边缘的情况，尤其是在头发这种本身就比较细的地方。fDoG利用图像流的方式，较为准确信息地提取出了边缘信息。同时，由于照片的尺寸比较大，在使用fDoG的时候，处理时间相较于其他算法显著变长。

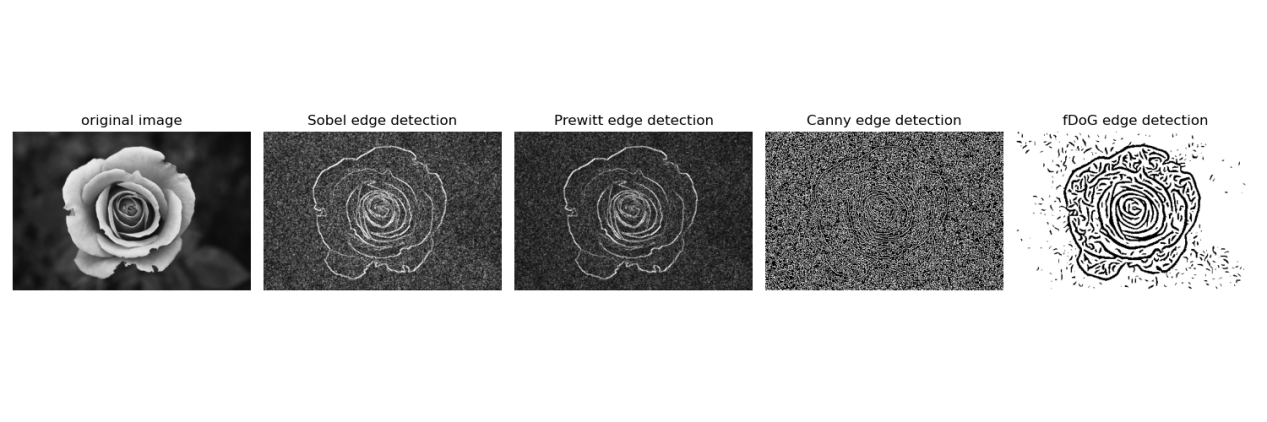
****

图5

在原图像的基础上添加噪声，可以发现噪声对Canny提取边缘的结果影响较大，所以一般在Canny提取边缘之前使用高斯滤波去除一定噪声。下面是去除了噪声之后的边缘提取结果。

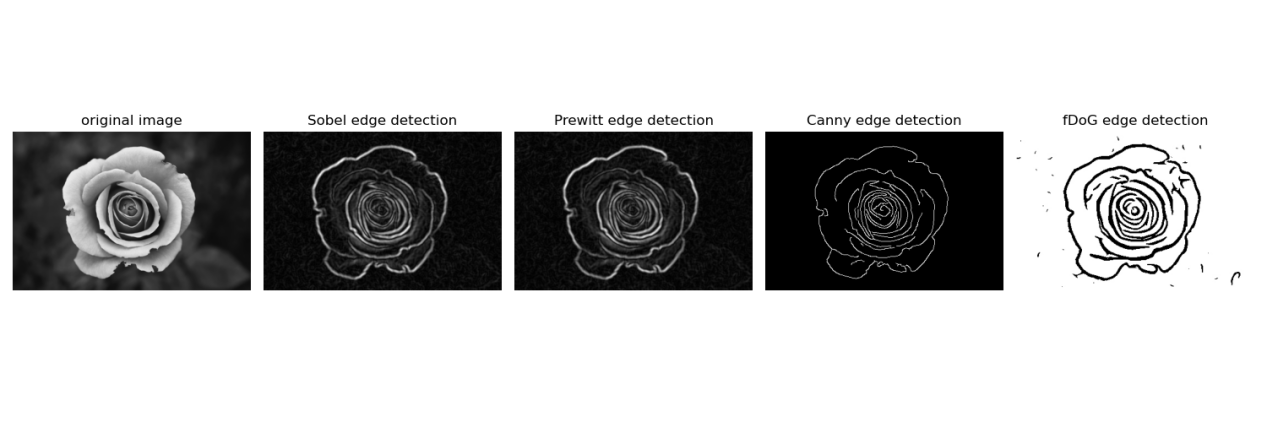


图6

可以看到在使用高斯滤波器对图像进行平滑处理之后，四种算法对边缘的提取结果都有所改善，其中Canny取得了和没有加入噪声时几乎相同的效果，对于Sobel和Prewitt算子而言，虽然效果比没有对图像进行平滑处理更好了，但是依然受到了噪声的影响，图像平滑操作将这一影响降低了。

**4.1 边缘提取算法分析**

首先，对于各个算法进行时间复杂度分析。

在不对原图像添加噪声的情况下，除去定义算子等时间，对几种边缘提取算法的计算进行时间分析。得到下面表格。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Algorithm | Picture1 | Picture2 | Picture3 | Picture4 | Picture5 | Average |
| Prewitt | 000.00765 | 00.01849 | 0.00107 | 00.00187 | 000.01270 | 000.008356 |
| Sobel | 000.00600 | 00.00516 | 0.00000 | 00.00000 | 000.01548 | 000.005238 |
| Canny | 000.00548 | 00.00257 | 0.00104 | 00.00187 | 000.00755 | 000.018510 |
| fDoG | 102.71790 | 12.15505 | 4.56277 | 20.25718 | 526.25974 | 133.190528 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Picrute1 | Picture2 | Picture3 | Picture4 | Picture5 |
| 1279 ×1076 | 590×578 | 427 ×285 | 674 ×786 | 1772 ×2532 |

实验结果显示，在四种算法种Sobel算子的速度是最快的，其次是Prewitt算子，因为二者都是使用一阶梯度提取图像边界信息，运算量较少且运算比较简单。

**4.2 关于fDoG的讨论**

**4.2.1 为何FDoG可能产生宽度不同的线条？**

FDoG 的原理是首先沿着图像梯度方向构建流场，使得滤波器可以适应图像的局部结构特征，然后根据流场方向提取线条。通过两个高斯函数的差值提取边缘，增强结构特征，同时抑制低频噪声。所以如果流场方向变化较快或复杂，生成的线条宽度可能会受到影响。检测过程中通常会使用非极大值抑制（NMS）和阈值化操作。不同的阈值可能会使较弱的边缘被忽略，进而影响线条的宽度。较低的阈值会保留更多的线条信息，但可能使线条看起来较粗。

**4.2.2 FDoG是否可以产生宽度为1的线条？如何实现？**

fDoG可以产生宽度为1的线条。首先让高斯核的尺度接近于像素大小，这样可以捕捉到单像素宽的细节。NMS是关键步骤，通过抑制梯度方向上的非极大值，仅保留最显著的响应，从而实现细化的线条。设置合适的双阈值（高低阈值结合），以剔除低强度的响应，同时保留精确的边缘。

**五、总结**

Prewitt算子，在3x3的窗口内计算水平和垂直方向的梯度，通常适用于检测直线边缘，利用Prewitt算子得到的结果比Roberts算子更平滑，能够检测到更多的边缘信息。Prewitt算子的计算效率也比较高。但是从图2可以看出，Prewitt对边缘的定位和细节处理能力不如Roberts，所以相比较于Roberts算子，Prewitt更加适用于对精度要求不高的使用场景。

Sobel 算子改进了 Prewitt 算子，引入了带权重的卷积核（如 [1,2,1][1,2,1]），增强对中心像素的响应，从而更准确地提取边缘信息。在一定程度上减少了噪声的干扰。适合大面积边缘提取。对细节不敏感：难以提取细小或模糊的边缘。复杂场景表现一般：在背景复杂的图像中可能产生伪边缘。我选取的图像基本都是比较真实比较复杂的，所以从实验结果中可以看出，Sobel算子基本上只能检测出图像的大致轮廓，对于精细的边缘无法识别。

Canny 算子是目前广泛使用的边缘检测算法。包含高斯平滑降低噪声，计算梯度幅值和方向，非极大值抑制精确定位边缘等多个步骤，并通过双阈值区分强弱边缘，避免边缘断裂。通过高斯滤波显著减少噪声的影响，抗噪声能力强。非极大值抑制能有效去除虚假的边缘，边缘定位精准。双阈值处理能连接弱边缘，生成更完整的边缘图。但是，高斯核大小和阈值选择对结果有较大影响。需要多步处理，速度较慢。

FDoG 算法在 DoG 的基础上，结合图像的结构主方向，对每个像素进行流方向上的滤波计算，以更好地提取线条和细长边缘，对细长的边缘（如纹理、线条）表现出色，结合方向信息，有效减少虚假边缘，但是需要估计主方向并进行方向相关滤波，计算较为复杂，流导向计算需要准确调整方向参数。

完成实验后，对于各种边缘提取算法的一些特性有了更加形象量化的认知，比如噪声对于图像提取边缘特征的影响具体是怎样的，多少的噪声能够产生会对边缘提取造成影响等等。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **算法** | **计算复杂度** | **抗噪性能** | **检测精度** | **连续性** | **应用场景** |
| Sobel | 低 | 中 | 中 | 差 | 快速、简单边缘检测 |
| Prewitt | 低 | 低 | 中 | 差 | 基础边缘检测 |
| Canny | 高 | 高 | 高 | 好 | 高精度边缘检测 |
| FDOG | 高 | 中 | 高 | 好 | 复杂纹理分析，曲线边缘检测 |