

数字图像处理 作业 3

朱文杰 220320623 自动化 6 班 | 2024.9.23

3.29

讨论用一个 3×3 低通滤波器核反复滤波一幅图像的极限效果。可以忽略边界效应。

解：如果忽略边界效应，由于低通滤波器的核内所有元素和为 1，所以每次滤波后，图像的像素灰度值总和不变。

然而，由于低通滤波器的核内所有元素都是正数，所以每次滤波后，图像的像素灰度值会变得更加平滑，即图像的细节会逐渐消失。最后，整幅图像会变得非常模糊，只剩下一个平均灰度值。

3.31

使用系数之和为 1 的一个核对图像进行滤波。证明原图像中的像素值之和，与滤波后的图像中的像素值之和相等。

证明：设原图像为 $f(x, y)$ ，滤波核为 $h(x, y)$ ，滤波后的图像为 $g(x, y)$ 。

由于滤波核的系数之和为 1，所以滤波后的图像中的像素值之和为

$$\begin{aligned}\sum_{x,y} g(x,y) &= \sum_{x,y} \sum_{i,j} f(x-i, y-j)h(i,j) \\ &= \sum_{i,j} h(i,j) \sum_{x,y} f(x-i, y-j) \\ &= \sum_{i,j} h(i,j) \sum_{x,y} f(x,y) \\ &= 1 \cdot \sum_{x,y} f(x,y) \\ &= \sum_{x,y} f(x,y)\end{aligned}$$

所以原图像中的像素值之和与滤波后的图像中的像素值之和相等。

3.37

完成如下工作。

- 开发一个程序，计算一个 $n \times n$ 邻域的中值。
- 提出一种技术，更新邻域中心逐个像素移动时的中值。

解：(a) 整体思路为先对整个邻域内像素值进行快速排序，然后取中间值作为中值。使用 MATLAB 实现如下：

```
function median = median_filter(image, n)
    [m, n] = size(image);
    median = zeros(m, n);
    for i = 1:m
        for j = 1:n
            % 计算邻域
            neighborhood = image(max(1, i-n):min(m, i+n), max(1, j-n):min(n, j+n));
            % 排序
            sorted = sort(neighborhood(:));
            % 取中值
            median(i, j) = sorted(ceil(numel(sorted)/2));
        end
    end
end
```

(b) 对于邻域中心逐个像素移动时的中值，可以使用快速排序的方法，每次移动时只需要将离开邻域的像素值从排序中删除，将进入邻域的像素值插入排序中，然后取中间值即可。

3.40

图 3.46 表明，与中心是-4 的拉普拉斯相比，中心是-8 的拉普拉斯的结果要清晰一些，说明原因。

解：中心是-8 的拉普拉斯算子的解析式为

$$g_8(x, y) = f(x+1, y+1) + f(x-1, y-1) + f(x+1, y-1) + f(x-1, y+1) \\ + f(x, y+1) + f(x, y-1) + f(x+1, y) + f(x-1, y) - 8f(x, y)$$

注意到中心是-4 的拉普拉斯算子

$$g_4(x, y) = f(x, y+1) + f(x, y-1) + f(x+1, y) + f(x-1, y) - 4f(x, y)$$

所以

$$g_8(x, y) = g_4(x, y) + f(x+1, y+1) + f(x-1, y-1) + f(x+1, y-1) + f(x-1, y+1)$$

$g_8(x, y)$ 相比 $g_4(x, y)$ 增加了是对角线上的像素值与本像素值的微分，所以增强了锐化的效果。