

# OpenCV-python 学习笔记 OpenCV图像平滑

## 1. 2D卷积

同一维信号一样，可以对2D图像实施低通滤波（LPF）和高通滤波（HPF）。LPF用于去除噪音，模糊图像，HPF用于找到图像的边缘。

OpenCV提供的函数 `cv.filter2D()` 可以对一幅图像进行卷积操作。练习一幅图像使用平均滤波器。举例下面是一个5X5的平均滤波器核：

$$K = \frac{1}{25} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

将核放在图像的一个像素A上，求与核对应的图像上25（5x5）个像素的和(卷积)，再取平均数（平均池化），用这个平均数代替像素A的值。

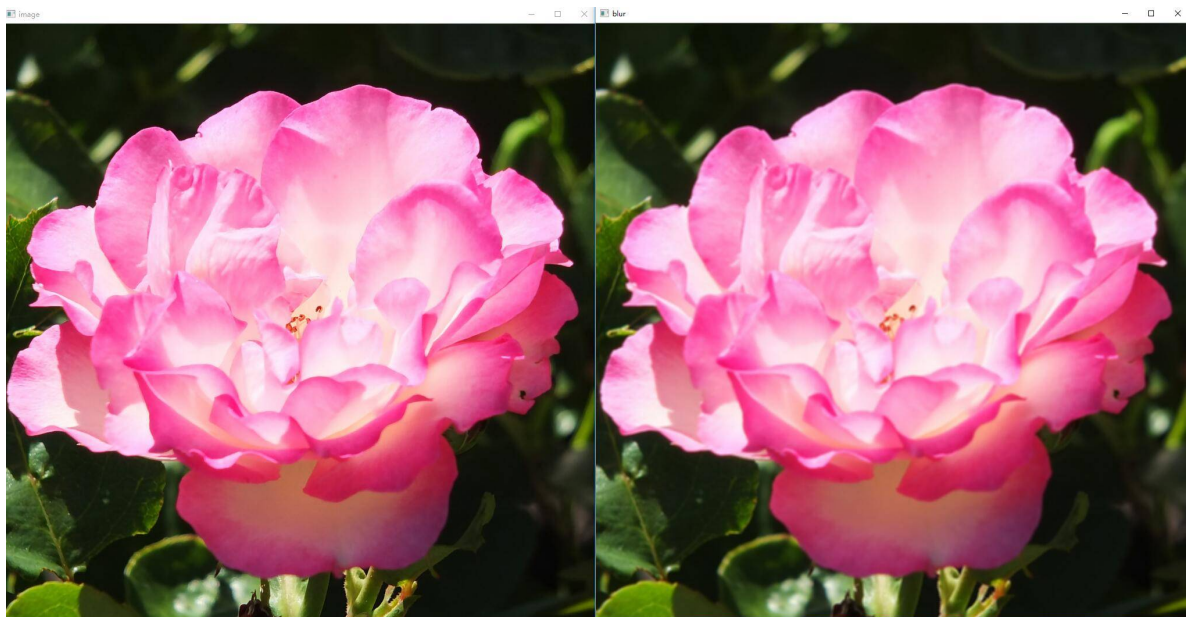
```
1 import cv2
2 import numpy as np
3 from matplotlib import pyplot as plt
4
5 img = cv2.imread('1024.jpg') # 读取图片
6
7 kernel = np.ones((5,5),np.float32)/25 # 卷积核
8 dst = cv2.filter2D(img,-1,kernel) # 2D卷积操作
9
10 plt.subplot(121),plt.imshow(img),plt.title('original')
11 plt.xticks([],plt.yticks([]))
12 plt.subplot(122),plt.imshow(dst),plt.title('averaging')
13 plt.xticks([],plt.yticks([]))
14 plt.show()
```

## 2. 平均

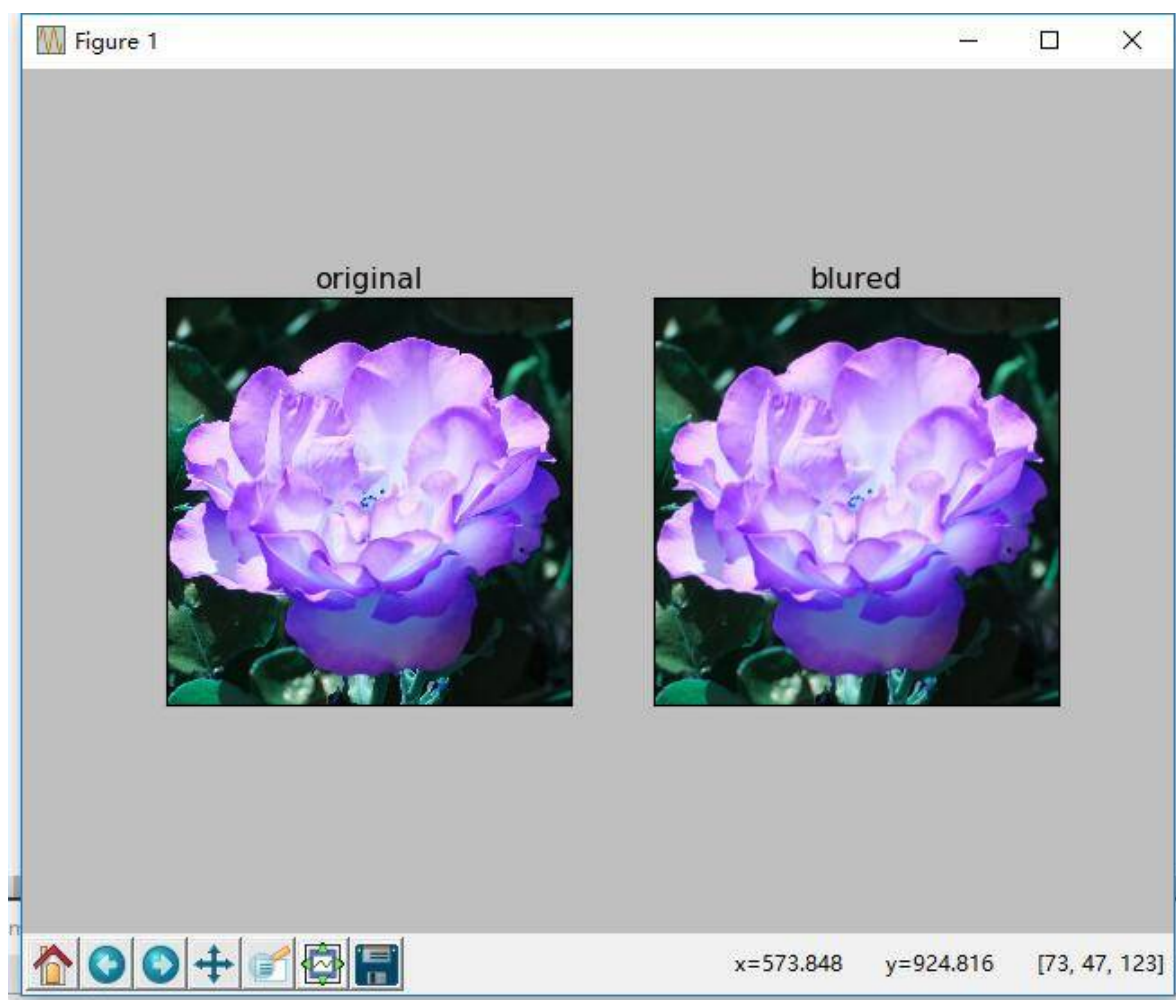
这是由一个归一化卷积框完成的，他只是用卷积框覆盖区域所有像素的平均值来代替中心元素。可以使用 `cv2.blur()` 和 `cv2.boxFilter()` 来实现，我们需要设定卷积框的宽和高。同样是一个矩阵。

例如：

```
1 import cv2
2 import numpy as np
3 from matplotlib import pyplot as plt
4
5 img = cv2.imread('1024.jpg')
6
7 blur = cv2.blur(img,(5,5))
8
9 while(1):
10     cv2.imshow('image',img)
11     cv2.imshow('blur',blur)
12     k=cv2.waitKey(1)
13     if k == ord('q'):#按q键退出
14         break
15 cv2.destroyAllWindows()
```



显示结果：



### 3. 高斯模糊

将卷积核换成高斯核（卷积核中的值符合高斯分布，方框中心的值最大，其余方框根据距离中心元素的距离递减，原先求平均变为加权平均，权重为方框中的值），使用 `cv2.GaussianBlur()` 函数。

使用该函数时，需要指定高斯核的宽和高（必须是奇数），以及高斯函数沿X,Y方向的标准差。如果我们只指定了X方向的标准差，Y方向也会取相同值，如果两个标准差都是0.那么函数会根据核函数的大小自己计算，高斯滤波可以有效的从图像中去除高斯噪音。

也可以使用 `cv2.getGaussianKernel()` 自己构建一个高斯核。

代码实现将上面的卷积操作改为：

```
1 # 0是指根据窗口大小（5,5）来计算高斯函数标准差
2 blur = cv2.GaussianBlur(img,(5,5),0)
```

### 4. 双边滤波

函数 `cv2.bilateralFilter()` 能在保持边界清晰的情况下有效的去除噪音，但比较慢。

这种高斯滤波器只考虑像素之间的空间关系，而不会考虑像素值之间的关系（像素的相似度），所以这种方法不会考虑一个像素是否位于边界，因此边界也会被模糊掉。

双边滤波在同时使用空间高斯权重和灰度值相似性高斯权重。空间高斯函数确保只有邻近区的像素对中心点有影响，灰度值相似性高斯函数确保只有与中心像素灰度值相近的才会被用来做模糊运算。所以能保证边界不会被模糊，因此边界处的灰度值变化比较大。

代码实现：

```
1 #cv2.bilateralFilter(src, d, sigmaColor, sigmaSpace)
2 #d - Diameter of each pixel neighborhood that is used during
   filtering. # If it is non-positive, it is computed from
   sigmaSpace
3 # 9 邻域直径，两个 75 分别是空间高斯函数标准差，灰度值相似性高斯函数标准差
4 blur = cv2.bilateralFilter(img,9,75,75)
```

## 5. 中值模糊

利用 `cv2.medianBlur()` 实现

就是用与卷积框对应像素的中值来替代中心像素的值，这个滤波器经常用来去除椒盐噪声。

前面的滤波器都是用计算得到的一个新值来取代中心像素的值，而中值滤波是用中心像素周围或者本身的价值来取代他，他能有效去除噪声。卷积核的大小也应该是一个奇数。

需要给原始图像加上50%的噪声，然后用中值模糊。

```
1 median = cv2.medianBlur(img,5)
```