首先看下OpenCV 官方文档对于cvSmooth各个参数的解释：

Smooths the image in one of several ways.

C: void cvSmooth(const CvArr\* src, CvArr\* dst, int smoothtype=CV\_GAUSSIAN, int param1=3, int param2=0, double param3=0, double param4=0)

其对于每个参数的解释如下：

**param1** – The first parameter of the smoothing operation, the aperture width. Must be a positive odd number (1, 3, 5, ...)

**param2** – The second parameter of the smoothing operation, the aperture height. Ignored by CV\_MEDIAN and CV\_BILATERAL methods. In the case of simple scaled/non-scaled and Gaussian blur if param2 is zero, it is set to param1 . Otherwise it must be a positive odd number.

**param3** – In the case of a Gaussian parameter this parameter may specify Gaussian  \sigma  (standard deviation). If it is zero, it is calculated from the kernel size:

C:\Users\kangqingbo\Desktop\84729e19fed623015ebf4dc8b76897157612dc4d.png

Using standard sigma for small kernels( C:\Users\kangqingbo\Desktop\7b5343f2d9e22816e8de24262565ff2937695b2c.png to C:\Users\kangqingbo\Desktop\d82a7109054f6068dbb1152f48bd109090f2c73d.png ) gives better speed. If param3 is not zero, while param1 and param2 are zeros, the kernel size is calculated from the sigma (to provide accurate enough operation).

对于参数smoothtype为CV\_GAUSSIAN的高斯滤波来讲，param1和param2是高斯滤波核（Gaussian kernel）的尺寸，param3为高斯核的标准差（详见博文：<http://www.cnblogs.com/pegasus/archive/2011/05/20/2052031.html>）。文档里已经说了当param3或者param4为零的时候，如何根据核的大小算出标准差，那么就如本题的情况，在param1和param2为零即变换核的高宽都为零的时候，如何根据param3和param4算出param1和param2呢？在OpenCV的smooth.cpp文件中有个createGaussianFilter函数，下面的代码即为当只提供标准差（平行向或者竖直向）的时候，如何算法核的大小，代码：

    // automatic detection of kernel size from sigma

    if( ksize.width <= 0 && sigma1 > 0 )

        ksize.width = cvRound(sigma1\*(depth == CV\_8U ? 3 : 4)\*2 + 1)|1;

    if( ksize.height <= 0 && sigma2 > 0 )

        ksize.height = cvRound(sigma2\*(depth == CV\_8U ? 3 : 4)\*2 + 1)|1;

这里的sigma1即为param3，sigma2即为param4，可以看出，当图像的数据类型为8为无符号型时，核的大小（直径）为 6 \* sigma + 1，大概OpenCV认为半径为3\*sigma的窗口就是高斯函数能量最集中的区域。至于在图像数据类型不为CV\_8U的时候为什么核就变成了8\*sigma + 1 就不得而知了。

下面回到本题。

我用来做变换的原始图像为：

E:\OpenCV_Code\OpenCV_Code\练习题\第五章\Problem_3\Problem_3\Fig0333(a)(test_pattern_blurring_orig).tif

param1 = param2 = 9的结果图像（这时候的param3为1）：



param1 = param2 = 0的结果图像（这时候的param3为1）：



上面这两幅图的视觉效果差不多，事实是这两幅图像完全一样，其**峰值信噪比**为正无穷大。

为什么？

上面已经说了当核大小为零的时候，OpenCV会根据你提供的sigma大小来得出核的大小，这里的sigma为1，所以得到的核大小应该是7（即param1=param2=7）

那么，核大小为9和核大小为7的高斯滤波出来的结果为什么会一模一样？

在smooth.cpp文件中的createGaussianFilter函数中，有两个矩阵（数据类型为Mat）kx和ky，在size=param1=param2的时候，这两个矩阵一样，且矩阵的大小为size行1列，因此在param1=param2=9，param3=1的时候这两个矩阵为（前面的括号表示行列索引，后面为该位置的数值）：

(0, 0) - 0.000134

(1, 0) - 0.004432

(2, 0) - 0.053991

(3, 0) - 0.241971

(4, 0) - 0.398943

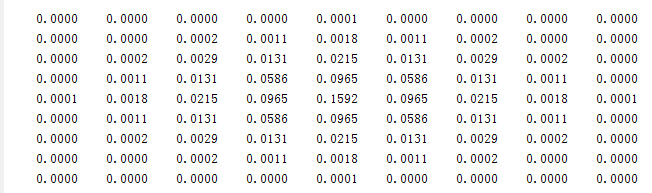
(5, 0) - 0.241971

(6, 0) - 0.053991

(7, 0) - 0.004432

(8, 0) - 0.000134

其对应的高斯变换核为：



在param1=param2=7，param3=1的时候这两个矩阵为：

(0, 0) - 0.004432

(1, 0) - 0.053991

(2, 0) - 0.241971

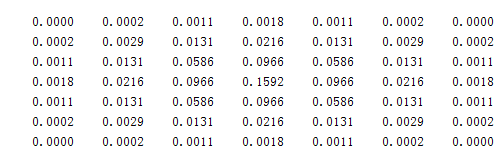
(3, 0) - 0.398943

(4, 0) - 0.241971

(5, 0) - 0.053991

(6, 0) - 0.004432

其对应的高斯变换核为：



可以看出，这两个矩阵除了第一个多了最外面的上面两行和左右两列外，里面是一模一样的，同时，多出来的行或者列除了一个数值为0.0001外，其余基本是零。这也就解释了核大小为9和核大小为7的高斯滤波出来的结果几乎是一模一样的（我这儿的例子是完全一模一样）。

param1 = param2 = 9的结果图像（这时候的param3为4）：



param1 = param2 = 0的结果图像（这时候的param3为4）：



可以看出这两个结果的视觉效果就不一样了，下面这个比上面那个药更模糊一点，事实也证明了这点，这两张图像的PSNR为26.565127。

在param1=param2=0，param3=4的时候，其参与运算的核大小是25（4\*6+1），因此与核大小为9的结果当然有差别。

param1 = param2 = 9的结果图像（这时候的param3为6）：



param1 = param2 = 0的结果图像（这时候的param3为6）：



这两副图像的PSNR为22.688340，也属于差距比较大的两张图像。

可以得出，第二副图像变换核大小为37，与核大小为9的变换核滤波出来的结果当然有很大区别。

总结：1. 在OpenCV所实现的高斯滤波中，param1或者param2为零（即变换核为0的时候），将根据param3或者param4的值算出变换核的大小；对于8位单通道图像，算法是size = 6\*sigma + 1；

2. 在产生高斯变换核的标准差（sigma）指定时，如果两个核大小差不多，很有可能出来的滤波结果图几乎完全一模一样；

3. 在标准差固定时，变换核越大，滤波出来的图像越模糊，丢失的细节也就越多；

4. 在变换核的尺寸固定时，标准差（即sigma，param3）越大，滤波出来的图像越模糊，丢失的细节也就越多。