高斯滤波器的递归实现

IanＴ.Yound,Lucas J.van Vliet

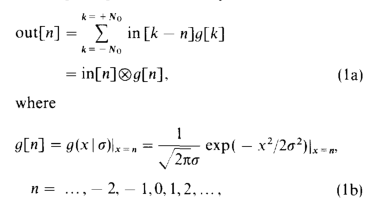
摘要：在本文中我们提出了一种高斯滤波器的递归实现，采用无限脉冲响应滤波器（IIR数字滤波器），该滤波器包含6个MADDs(multiplications and additions)在一个方向上，独立于高斯核的方差的值，与Rachid Deriche - "[Recursively implementing the Gaussian and its derivatives](http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.35.5904)", 1993.（这里写成了1987年）的实现对比，我们的迭代滤波器的参数更加简单、有所需要的高斯滤波器方差的封闭解。我们的实现总体上比用高斯模板直接进行卷积要快，比多次进行均值滤波，比高斯滤波的FFT实现都要快。

**关键词：**高斯滤波，IIR滤波器（无限脉冲响应滤波器），迭代滤波器，多维滤波器，平滑衍生算子

**前言**

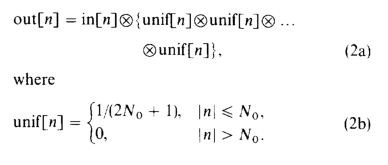
高斯滤波器被认为是图像滤波领域最核心的部分，因为在人类视觉模型的研究、在尺度空间中的结果和基于数字数据的模拟量的精确测量。

高斯滤波器在一维或者二维的实现是通过和需要的高斯模板做卷积来实现的（如公式1所示Eq.(1)）。因为多次利用更简单的滤波器（如均值滤波）进行卷积，和因为用递归滤波器模拟高斯滤波需要复杂的计算步骤来决定滤波器的参数，多次卷积根据中心极限定理可知，在这个极限中，多次的卷积与简单的均值模糊滤波器可以等效于高斯滤波器。

一维的与高斯模板的离散卷积可以表示为：  
  
，

这里是实数为整数，被典型为整数，通常取5左右，当取这个值的时候，g(x)的输出将变为g(x=0)的3.7\*10-6倍。

通过多次均值核的卷积实现高斯模糊每一维定义为：



通常使用三次均值模糊来逼近高斯模糊，对于一个需要的，则这里需要N0

≈[],也就是取整数，只有有限的高斯滤波器可以通过这种方式重建，这是因为我们需要使用整形的变量N0。

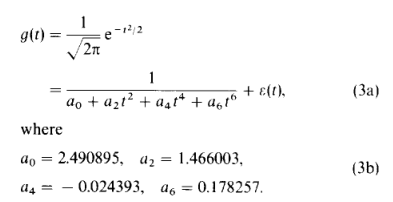
Deriche提出了一种复杂的设计步骤对于每一个,基于这样的一个原因，Deriche设计出了一种非高斯的递归滤波器，这个滤波器在每一维都有脉冲响应,随着这类滤波器有让人印象深刻的属性，但是他们也有一些缺点，它是具有各向异性的（在二维的非对称圆中），它不是值域滤波器，而高斯滤波器是值域滤波器。

因为多维的高斯滤波器具有可分离性（就是exp(-(n2+m2)=exp(-n2)exp(-m2))），对于我们来说是恰当的，来计算每种算法的计算复杂度每次只计算一维的计算复杂度。对于每一维直接使用卷积（公式（1a））,意味着（2N0+1）次MADDs(乘法和加法)。利用高斯滤波器的对称性，可以将计算复杂度降低为2N0次加法，和N0+1次乘法。使用公式（2a）中的意味着2N0次加法，Detiche逼近高斯的每一维都有12次乘加运算，而替代它的递归滤波器（上面提到的）需要8次MADDS在每一维。

在本文中，我们提出了一种高斯滤波器的替代实现，这种实现建立在递归结构上的，这种实现采用有限脉冲滤波器（IIR），这种滤波器具有6次MADDS在每一维中，并且独立于高斯核，实验的结果一般高于上述的实现，并且不损失精度。

2.规范在拉普拉斯域

我们的方法基于对高斯函数的合理逼近，这是由文献[1]的公式26.2.20说明。



这个误差可以限定到,相对应着g(t)的最大值，即当t=0时，这个最大的误差是一个更复杂的逼近g(t)的多项式，这个多项式为十阶多项式在文献[1]中被提出来了，但是提出来的版本显示已经证明是足够的了。

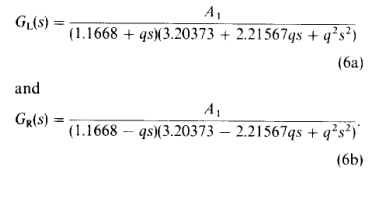
我们处理这种近似并不是高斯脉冲响应而是作为傅里叶变换的近似。这里我们有著名的结果，

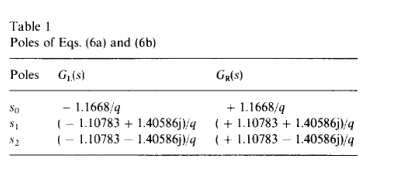
为什么这样做后面会变得更加明显，我们将使用q代替来实现合理的逼近。



，相当于

表达可以分解为两项的乘积，，的极点在左半平面，的极点在右半平面。





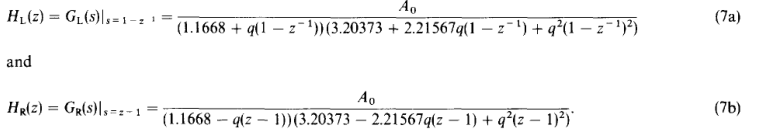
等式的极点如表1所示。

独立于,两个滤波器在复平面上都有极点。

**3.在Z-域上的表示：**

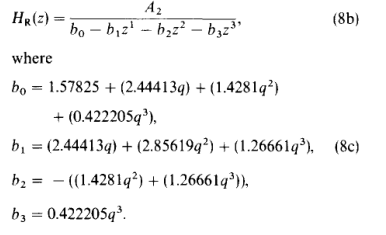
****是因果稳定的微分方程，可以转换为因果稳定的差分方程。代表非因果稳定的微分方程，可以别转换为稳定的差分方程。标准的将微分方程转换为差分方程的方法是采用双线性变换。这项技术会产生传递函数的零点，这是我们想避免的副作用，另外双线性插值还有其他的副作用，看附录A。

相反，我们选择使用后向差分**(y[n]- y(n - 1))/ T**近似导数**dy / dt,**从而代替s=(1-z-1)/T。和双线性变换一样，这种方法具有这样的属性将因果稳定的映射到因果稳定的，而且，没有零点引入到或者中，用于产生，我们使用前向差分方程来逼近复频率可以被替换为另外，这种方法具有这样的属性，可以将反因果、稳定映射到反因果稳定的假设T=1,则可推导出下面的公式：



这两个公式都可以重写为包含z-1和Z的标准多项式，即



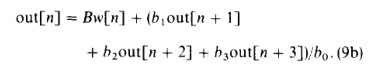


这个实现显示了下面的滤波策略，输入的数据首先通过前向的滤波，正如公式对应的差分方程，这个结果的输出，我们称之为，然后通过后向滤波器滤波，对应的公式为，差分方程是：

前向：



后向：



这两个滤波器都具有相同的归一化常数B，它可以确定通过下面这样的约束，就是滤波器的传递函数的值应该为1.0对于频率，这就导出了



**例子：**让我们看一下这个过程的使用，我们首先选择使用公式，这将导出B=0.01543。高斯滤波器的递归实现的脉冲响应如图所示，与之对比的是真实的高斯连续曲线，如图像所示，一个增大的尺度以增强高斯的尾部在中。

通过观察这个例子可以得到一些有用的信息。以及相关的极点（如表1中列出的）说明了每个分离项（如前向和后向）的脉冲响应将会是震荡的。高斯迭代滤波器的脉冲响应没有这种问题（使用双线性插值的情况并非如此，见附录A）。

递归高斯滤波器的脉冲响应的标准差并不是值(在公式4,5（a）,5(b)),从上面的例子中q = 5.0,根据高斯脉冲响应的最小均方误差匹配可以得到值为的均方根误差为最大绝对误差为