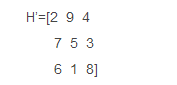
# **卷积算子计算方法(卷积运算)**

卷积操作是对图像处理时，经常用到的一种操作。它具有增强原信号特征，并且能降低噪音的作用。 那么具体是如何计算的呢？

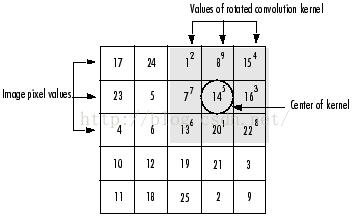


步骤：

  1）将算子围绕中心旋转180度



2）  滑动算子，使其中心位于输入图像g的（i，j）像素上



3）  利用公式求和，得到输出图像的（i,j）像素值

（2,4）元素值=  1\* 2+  8\*  9+15\* 4

                          +  7\* 7+14\* 5+16\* 3

                          +13\* 6+20\* 1+22\* 8=575

  4）重复2）,3），直到求出输出图像的所有像素值

注意：

scipy的signal模块经常用于信号处理，卷积、傅里叶变换、各种滤波、差值算法等。

二维的卷积需要用上面的signal.convolve2d()。



实现代码：

#-\*- coding: utf-8 -\*-  
# @Time : 2017/12/5 9:59  
# @Author : Z  
# @Email : S  
# @File : convolve.py  
def conv(lst1,lst2):  
 result=[]  
 #翻转第一个列表  
 lst1.reverse()  
 length1=len(lst1)  
 length2=len(lst2)  
 #移动翻转后的第一个列表，直到完全移入  
 for i in range(1,length1+1):  
 t=lst1[length1-i:]  
 #计算重叠的面积  
 v=sum((item1\*item2 for item1,item2 in zip(t,lst2)))  
 result.append(v)  
 #继续移动翻转后的第一个列表，直到完全移出  
 for i in range(1,length2):  
 t=lst2[i:]  
 v=sum((item1\*item2 for item1,item2 in zip(lst1,t)))  
 result.append(v)  
 return result  
print conv([1,2,3],[4,5])  
  
import numpy as np  
import scipy.signal  
x=np.array([1,2,3])  
y=np.array([4,5])  
print scipy.signal.convolve(x,y)