

一、Blur

1. MediumBlur

```
MediumBlur(radius).convolve(img, mode)
```

函数作用：中值模糊，对每个像素采用周围像素中值代替

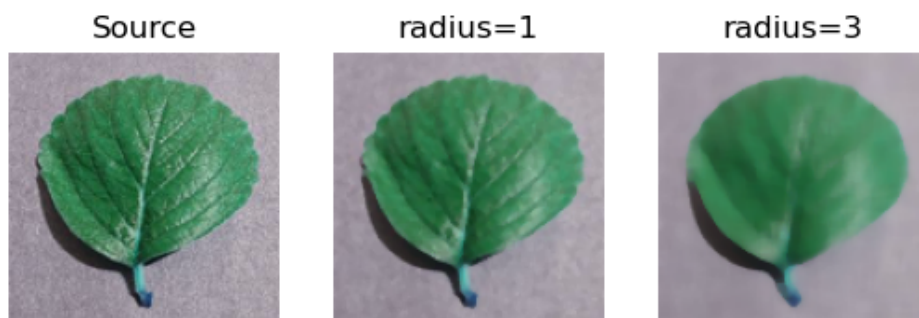
参数说明：

- radius --模板半径，决定计算时核半径，int类型
- img, 图片地址
- mode, 色彩模式 默认same 单色

举例

- `MediumBlur(1).convolve(imgPath, "fill")`
- `MediumBlur(1).convolve(imgPath, "fill")`

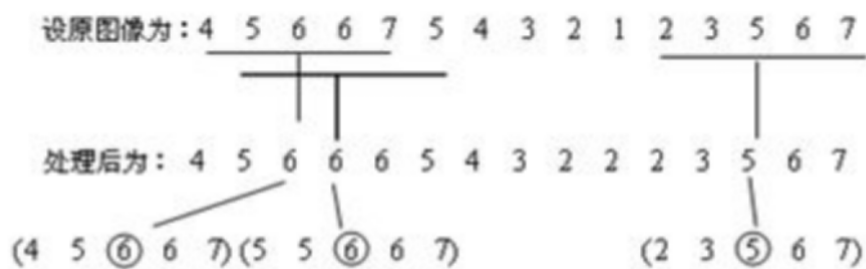
例图：



数学过程：



图9.9 一维窗口



2.AverageBlur

`AverageBlur(radius).convolve(img, mode, dtype)`

函数作用：均值模糊，对每个像素采用周围像素平均值代替

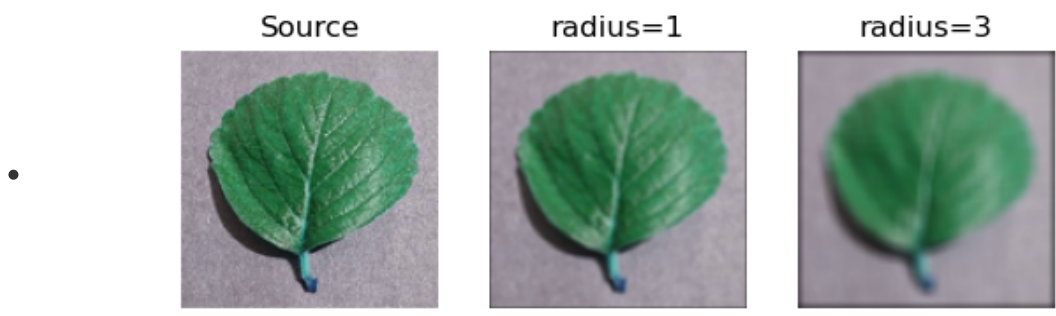
参数说明：

- radius --模板半径，决定计算时核半径，int类型
- img, 图片地址
- mode, 色彩模式 默认same 单色
- dtype, 数据类型

举例:

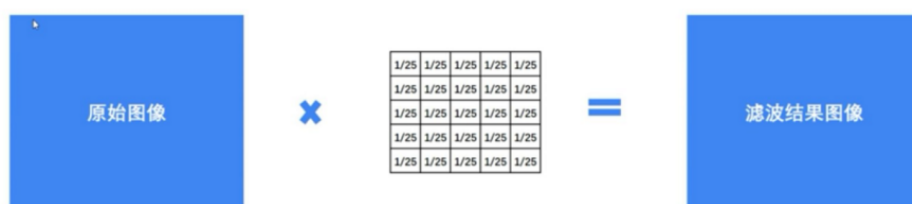
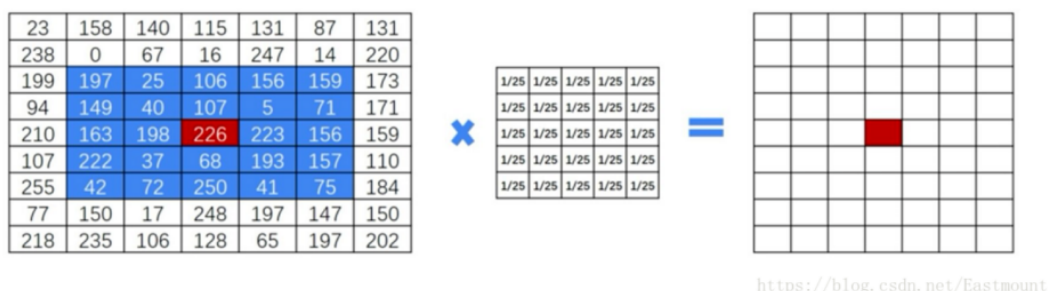
- `AverageBlur(1).convolve(imgPath, "fill")`
- `AverageBlur(3).convolve(imgPath, "fill")`

例图:



- 可以看出随着radius的变大，模糊程度加大

数学原理：



核中心像素等于以中心像素为中心的矩阵*半径为radius的卷积核

$$x_{\frac{n}{2}, \frac{n}{2}}^* = \frac{1}{n * n} \times \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,n} \end{pmatrix} \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{pmatrix}$$

3.BilateralBlur

```
BilateralBlur(radius).convolve(img,mode)
```

函数作用：双边模糊，相对于传统的高斯blur来说很重要的一个特性即可可以保持边缘（Edge Perseving）

参数说明：

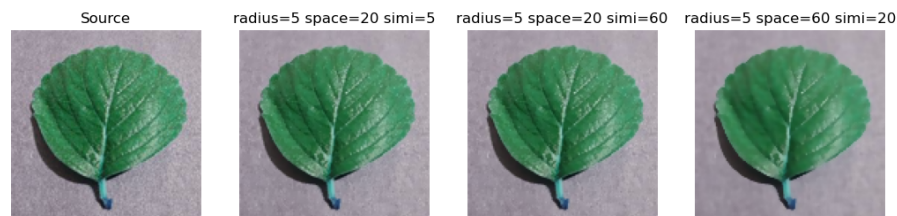
- radius --模板半径，决定计算时核半径，int类型
- img, 图片地址
- mode, 色彩模式 默认same 单色
- dtype, 数据类型

举例：

- `BilateralBlur(1,20,5).convolve(imgPath, "fill")`
- `BilateralBlur(5,20,60).convolve(imgPath, "fill")`
- `BilateralBlur(5,60,20).convolve(imgPath, "fill")`

例图：

•



• 数学原理：

◦ 空间距离的权重因子：

◦

$$c(\xi, \mathbf{x}) = e^{-\frac{1}{2}(\frac{d(\xi, \mathbf{x})}{\sigma_d})^2} = e^{-\frac{1}{2}(\frac{\|\xi - \mathbf{x}\|}{\sigma_d})^2}$$

◦ 相似度的权重因子：

◦

$$s(\xi, \mathbf{x}) = e^{-\frac{1}{2}(\frac{\sigma(f(\xi), f(\mathbf{x}))}{\sigma_r})^2}$$

◦

权重因子 = $\frac{s(\xi, x) + c(\xi, x)}{2}$

◦

4.GaussianBlur

```
GaussianBlur(radius,sigema).convolve(img,mode,dtype)
```

函数作用：双边模糊，相对于传统的高斯blur来说很重要的一个特性即可以保持边缘（Edge Perseving）

参数说明：

- radius 模板半径，决定计算时核半径，int类型
- sigema
- img, 图片地址
- mode, 色彩模式 默认same 单色
- dtype, 数据类型

数学原理：

正态分布的密度函数叫做"[高斯函数](#)"（Gaussian function）。它的一维形式是：

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} * e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$

其中， μ 是x的均值， σ 是x的方差。因为计算平均值的时候，中心点就是原点，所以 μ 等于0。

在二维空间中扩展为：

$$f(u,v) = \frac{1}{\sqrt{2\pi}\sigma} * e^{\frac{-(u^2+v^2)}{2\sigma^2}}$$

假定 $\sigma=1.5$,则模糊半径为1的权重矩阵如下：

0.0453542	0.0566406	0.0453542
0.0566406	0.0707355	0.0566406
0.0453542	0.0566406	0.0453542

中心像素计算过程如下

14	15	16	×	0.0947416	0.118318	0.0947416	=	14x0.0947416	15x0.118318	16x0.0947416
24	25	26		0.118318	0.147761	0.118318		24x0.118318	25x0.147761	26x0.118318
34	35	36		0.0947416	0.118318	0.0947416		34x0.0947416	35x0.118318	36x0.0947416

结果是：将这9个值加起来，就是中心点的高斯模糊的值

1.32638	1.77477	1.51587
2.83963	3.69403	3.07627
3.22121	4.14113	3.4107

5.MotionBlur

`MotionBlur(radius).convolve(img,mode,dtype)`

函数作用：运动模糊，在已知模糊运动核的前提下，通过核线性卷积的形式对图像添加运动模糊

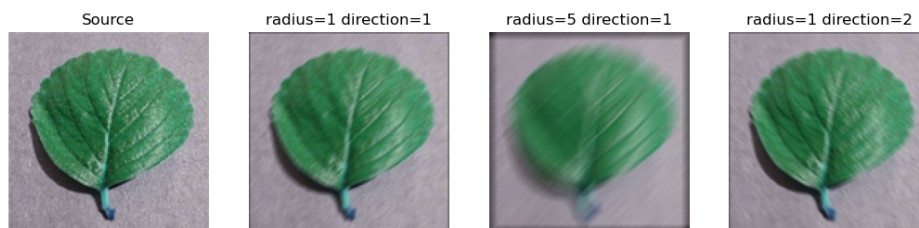
参数说明：

- radius --模板半径，决定计算时核半径，int类型
- img, 图片地址
- mode, 色彩模式 默认same 单色
- dtype, 数据类型

举例：

- `MotionBlur(1,20,5).convolve(imgPath, "fill")`
- `MotionBlur(5,20,60).convolve(imgPath, "fill")`
- `BilateralBlur(5,60,20).convolve(imgPath, "fill")`

例图：



数学原理：

$$x_{\frac{n}{2}, \frac{n}{2}}^* = \frac{1}{n} \times \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,n} \end{pmatrix} \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}$$

6.MeanShiftBlur

```
MeanShift(hs, hr, maxIter).convolve(img, mode='same', dtype='uint8')
```

函数作用：均值迁移模糊是图像边缘保留滤波算法中一种，经常用来在对图像进行分水岭分割之前去噪声，可以大幅度提升分水岭分割的效果。

参数说明：

- hs, 模板半径，决定计算核半径，int类型
- hr, 颜色差值的阈值
- maxIter, 最大迭代次数
- img, 图片地址
- mode, 色彩模式 默认same单通道 可为fill彩色三通道图片
- dtype, 数据类型

举例:

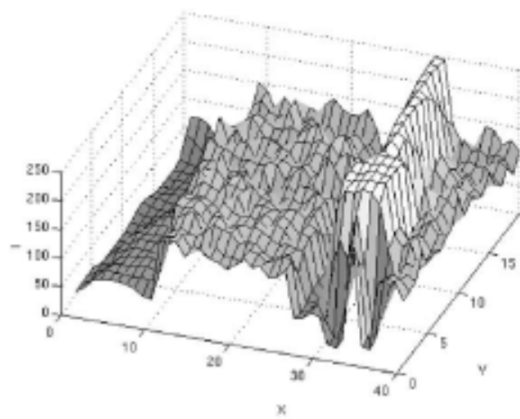
- `MeanShiftBlur(5,5,20).convolve(imgPath, "fill")`

例图：

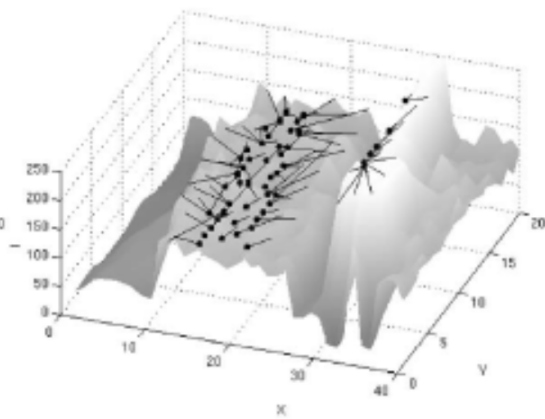
左图为原图，右图为在hs=5, hr=5, maxIter=20的情况面的图片



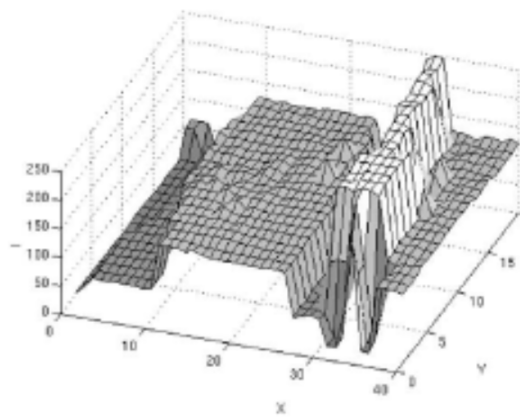
数学原理：



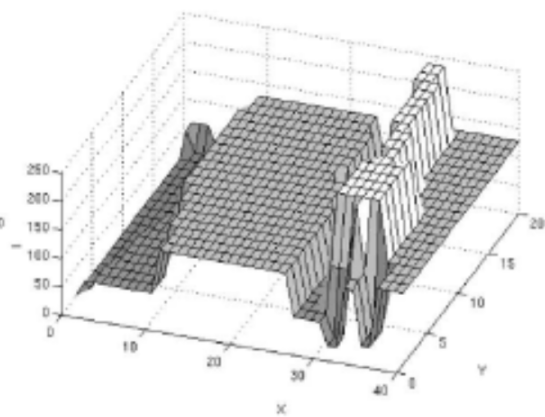
(a)



(b)



(c)



(d)

问题：时间复杂度过高 $O(N^5)$