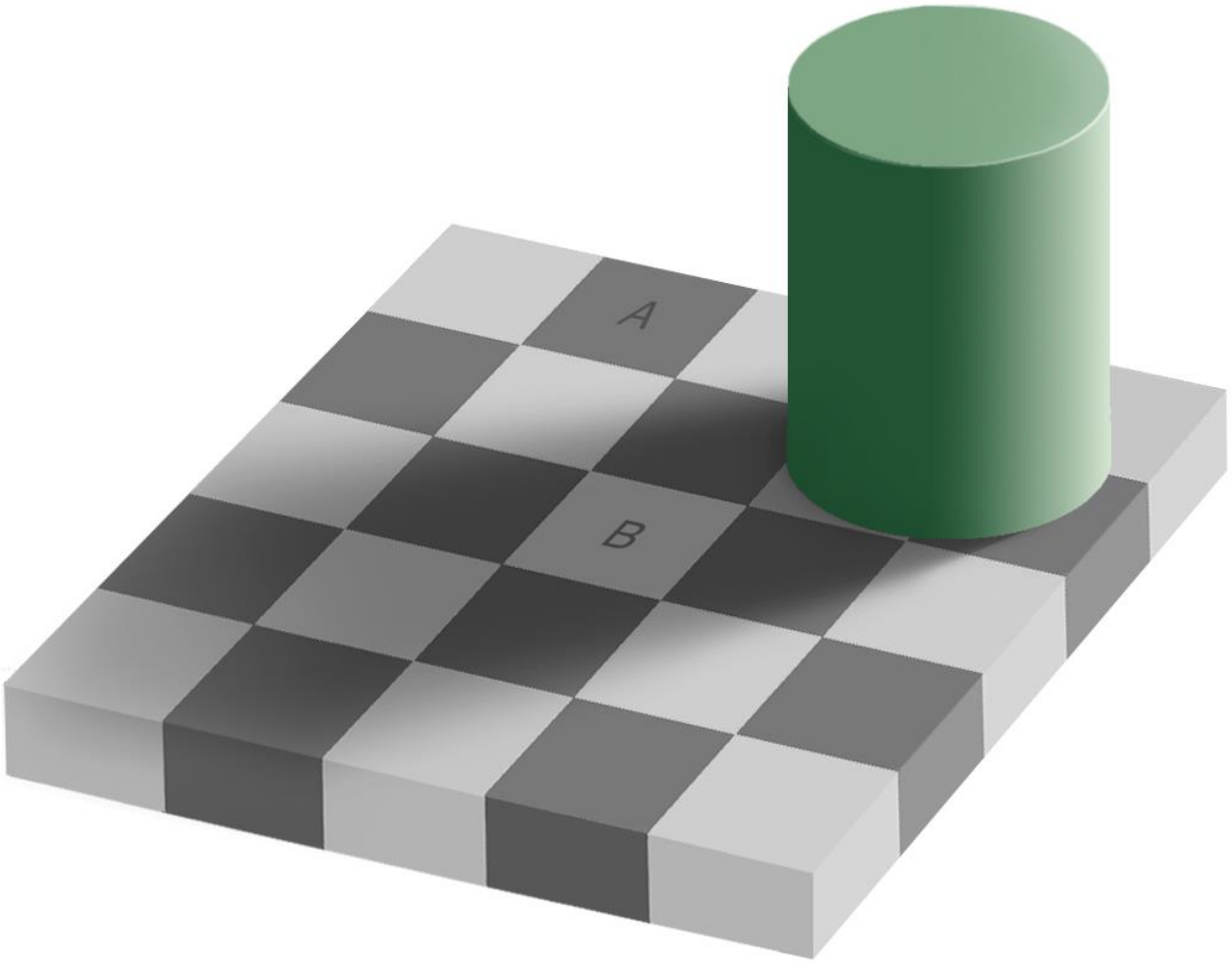
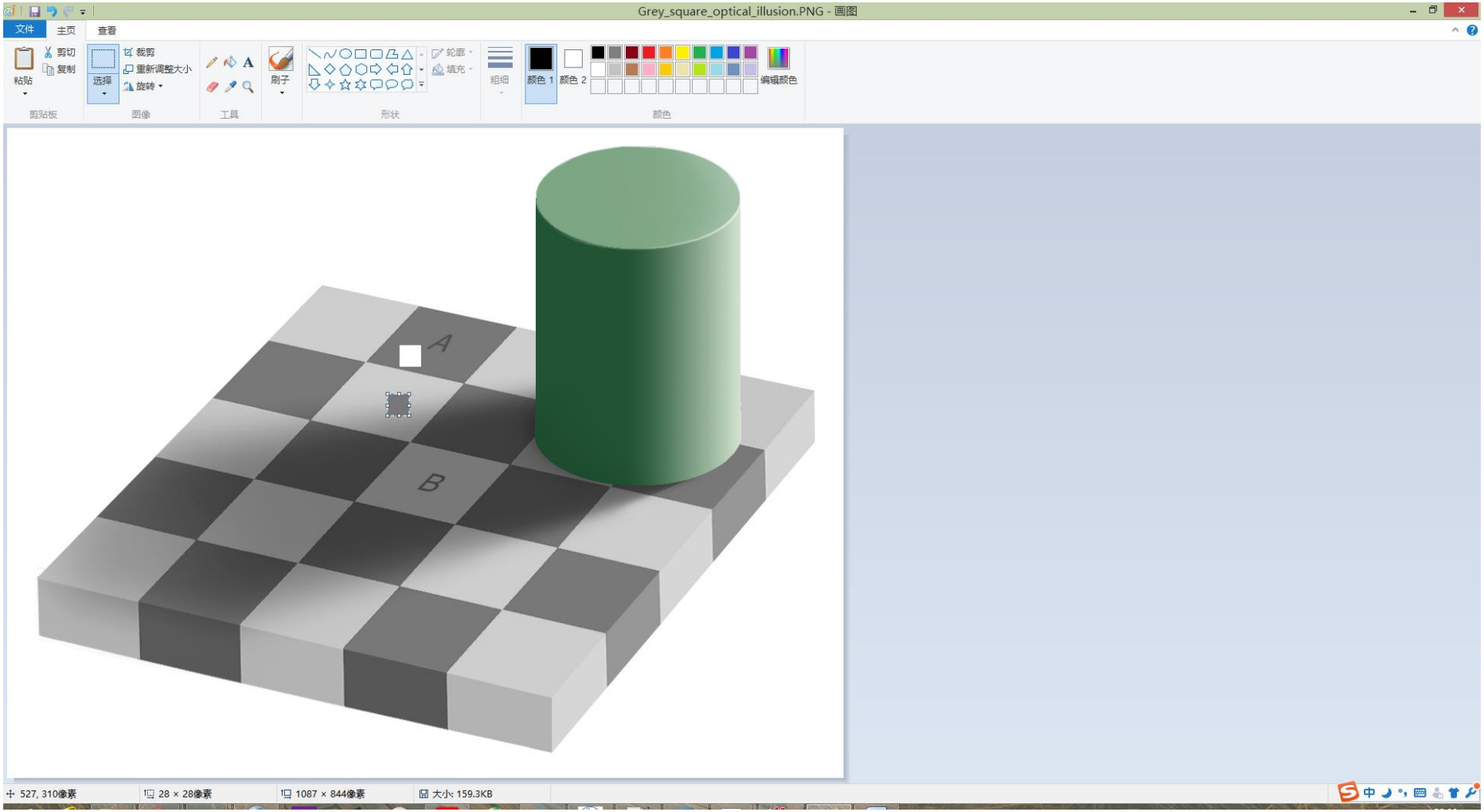
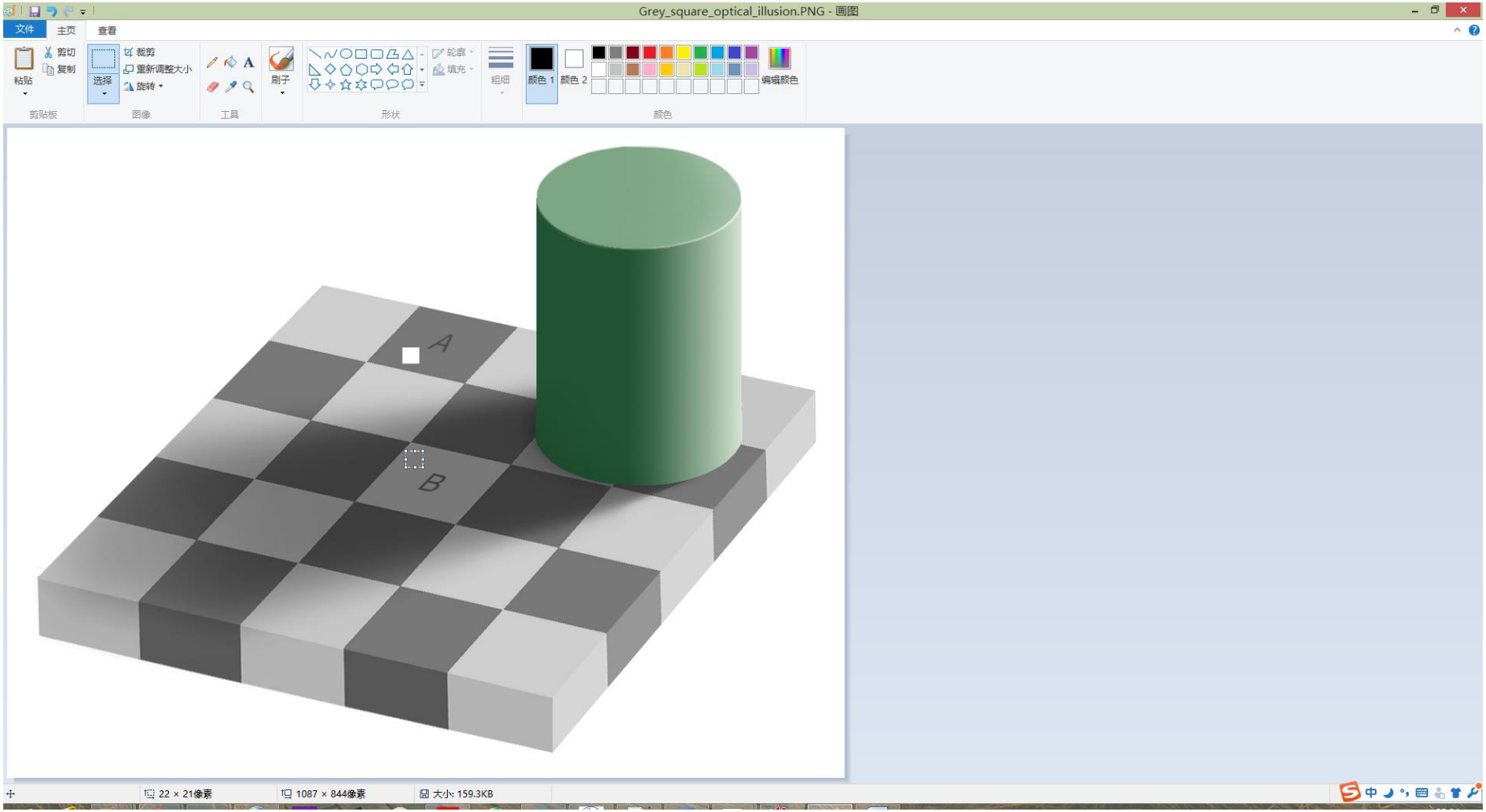
图像的数字化、颜色空间、滤波与频域变换

章国锋

A、B块中的颜色哪个亮？




Grey_square_optical_illusion.PNG - 画图
文件
主页
查看
粘贴
剪切
复制
选择
裁剪
重新调整大小
旋转
刷子
轮廓
填充
粗细
颜色 1
颜色 2
编辑颜色
剪贴板
图像
工具
形状
颜色
A
B
527, 310像素
28 × 28像素
1087 × 844像素
大小: 159.3KB

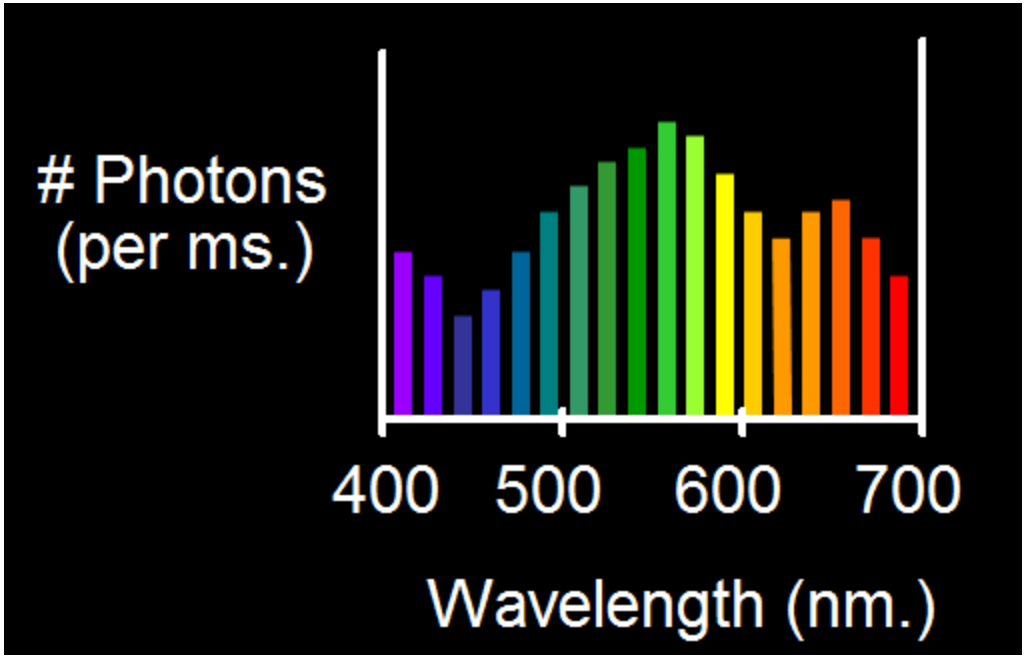

Grey_square_optical_illusion.PNG - 画图
文件
主页
查看
粘贴
剪切
复制
选择
裁剪
重新调整大小
旋转
刷子
轮廓
填充
粗细
颜色 1
颜色 2
编辑颜色
剪贴板
图像
工具
形状
颜色
A
B
22 × 21像素
1087 × 844像素
大小: 159.3KB

颜色的产生

- 定义
  - 颜色是视觉系统对可见光的感知结果。
  - 人的视网膜有对红，绿，蓝颜色敏感程度不同的三种椎体细胞，三种椎体细胞对不同频率的光的感知程度不同，对不同亮度的感知程度不同。
  - 任何颜色都可以通过这三种颜色混合而成。

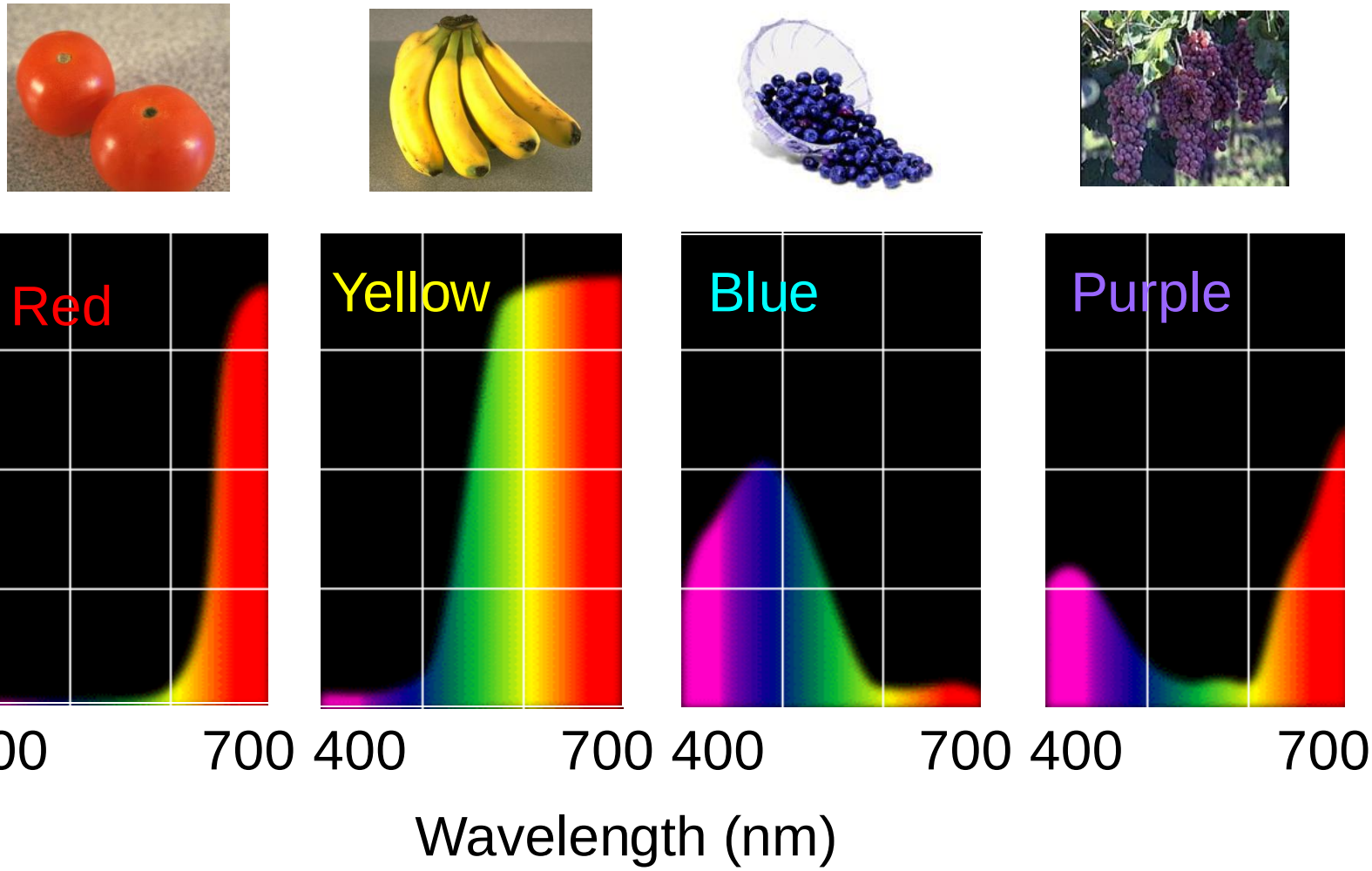
光的物理描述

任何一束光都可以在物理上用它的光谱描述：波长在400nm-700nm之间单位时间的光子数量.



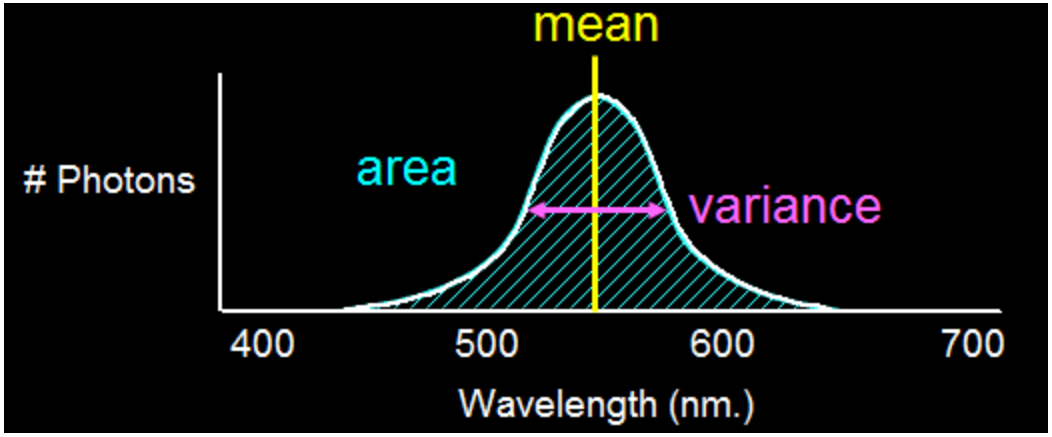
光的物理描述

一些表面反射的光谱的例子



物理光谱的描述

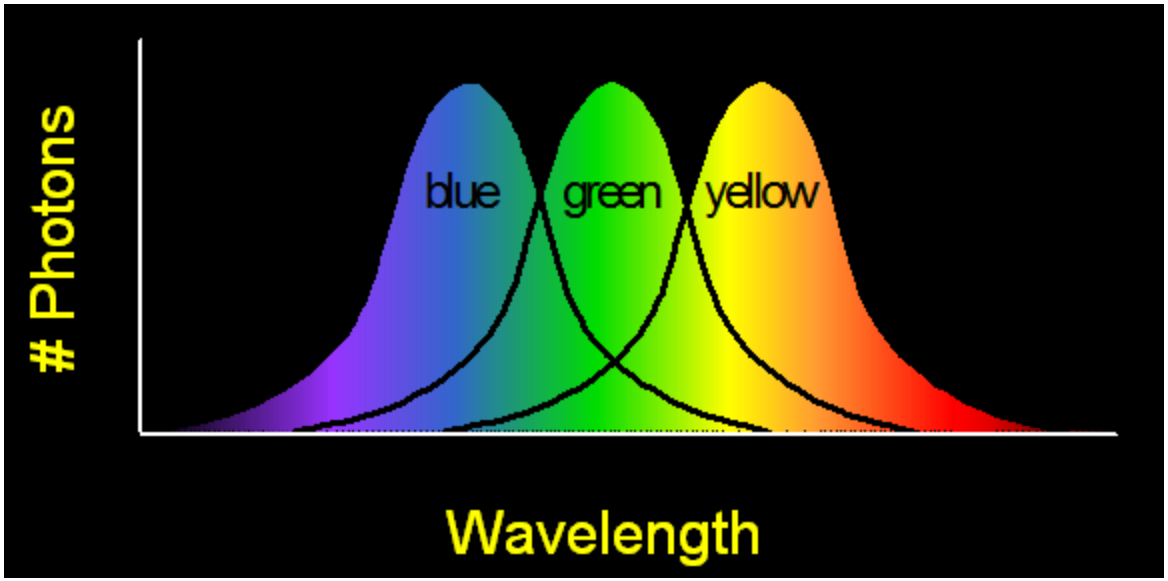
没有简单的函数用来描述在所有观察条件下的光的颜色
但是有一个很有用的限制：用正态分布描述物理光谱



颜色的基本属性

1. 色调（hue），指的是颜色外观，用于区别颜色的种类。

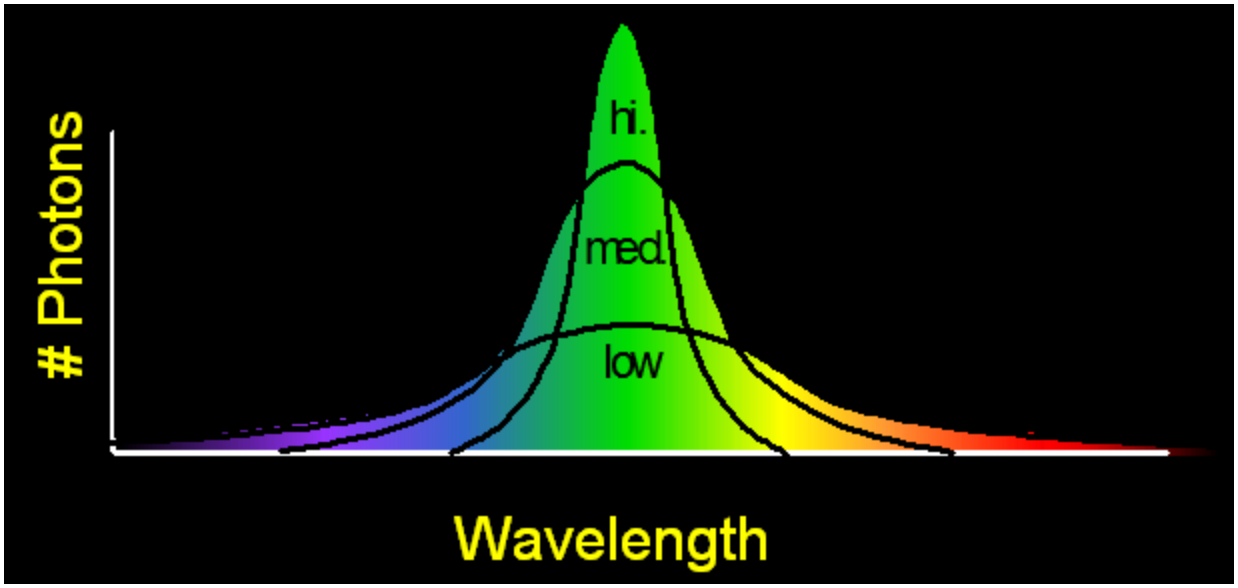
Mean ⟷ **Hue**



颜色的基本属性

2. 饱和度（saturation），指的是颜色的纯洁性，用于区分颜色的明暗程度，完全饱和的颜色是指没有渗入白光所呈现的颜色，单一波长组成的光谱色就是完全饱和的颜色。

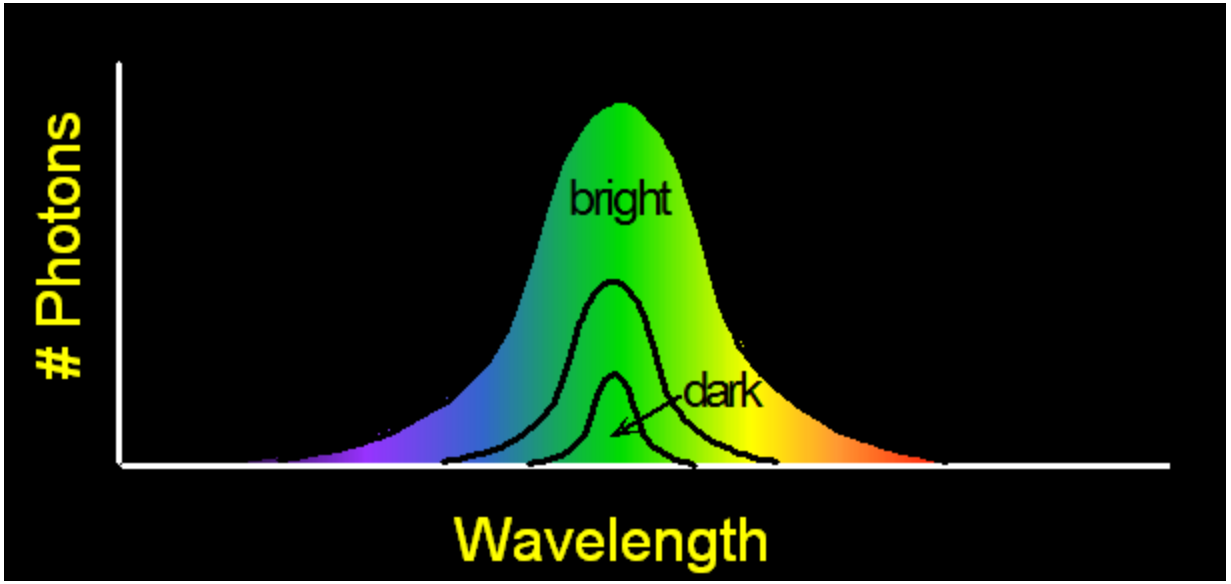
Variance ⟺ **Saturation**



颜色的基本属性

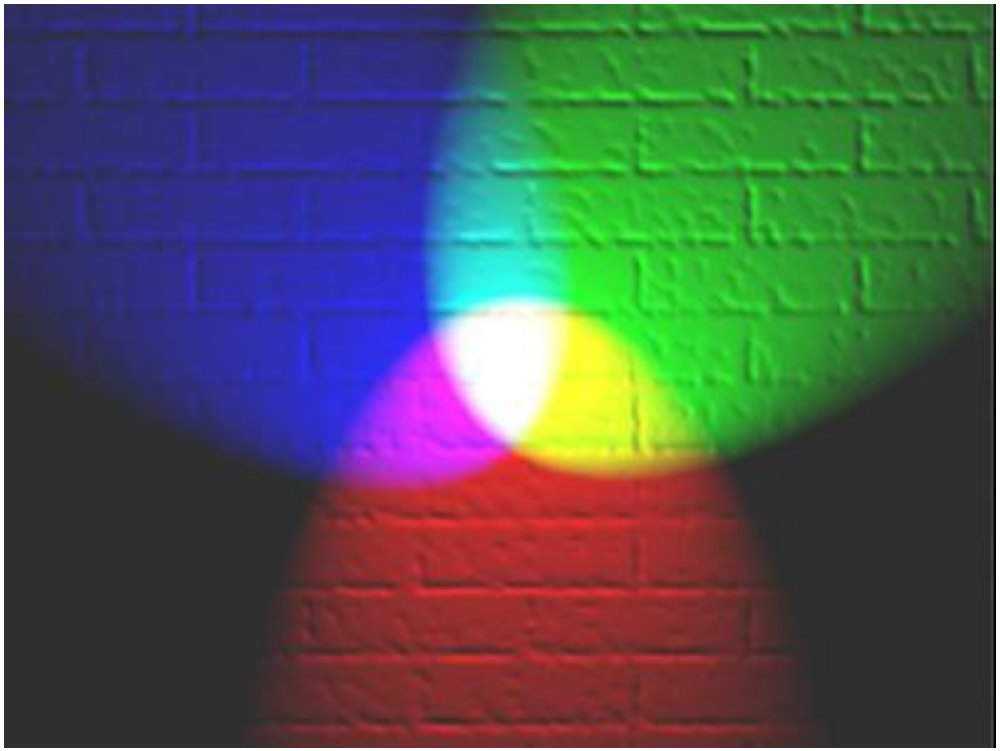
3. 亮度（brightness）指的是视觉系统对可见物体辐射或者发光多少的感知属性。

Area ⟺ **Saturation**



颜色空间

怎样表示颜色



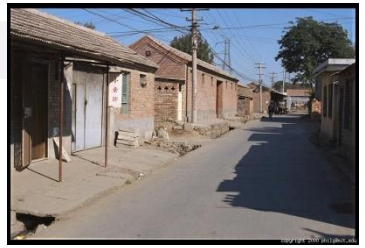
颜色空间

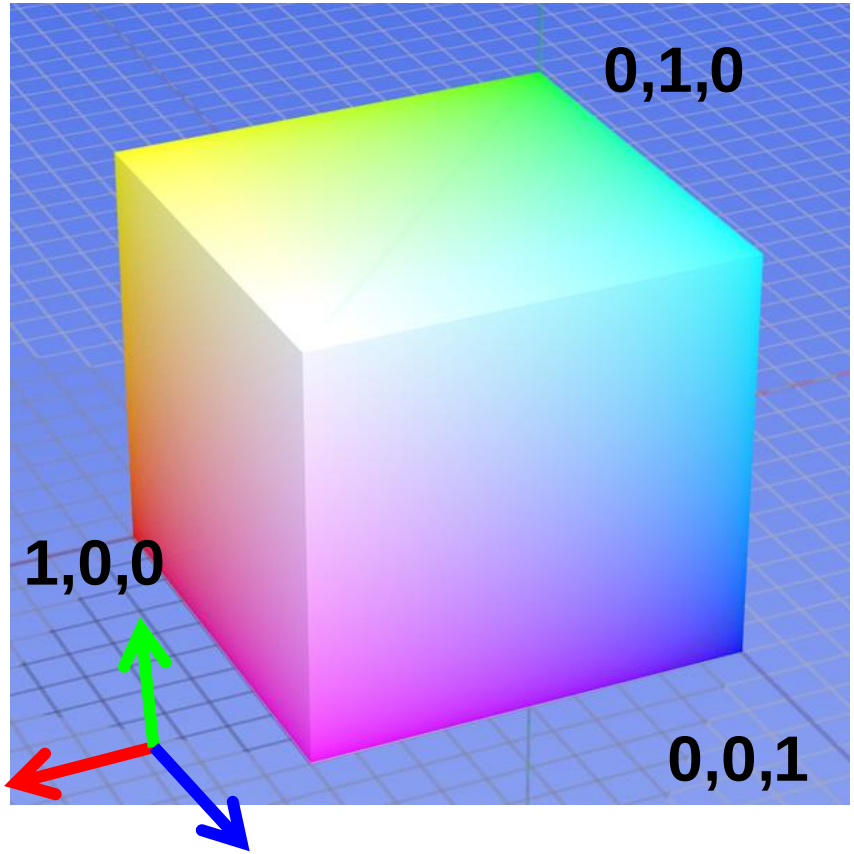
按用途分类：

- 面向诸如视频监视器、 彩色摄像机或打印机之类的硬件设备的RGB模型
- 面向以彩色处理为目的的HSI模型
- 在印刷工业上和电视信号传输中，经常使用CMYK和YUV色彩系统

颜色空间：RGB

默认的颜色空间





R

(G=0,B=0)

G

(R=0,B=0)

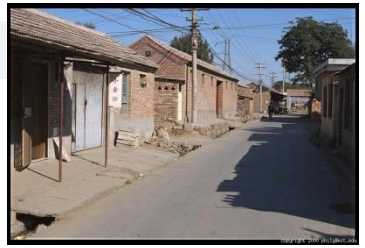
B

(R=0,G=0)

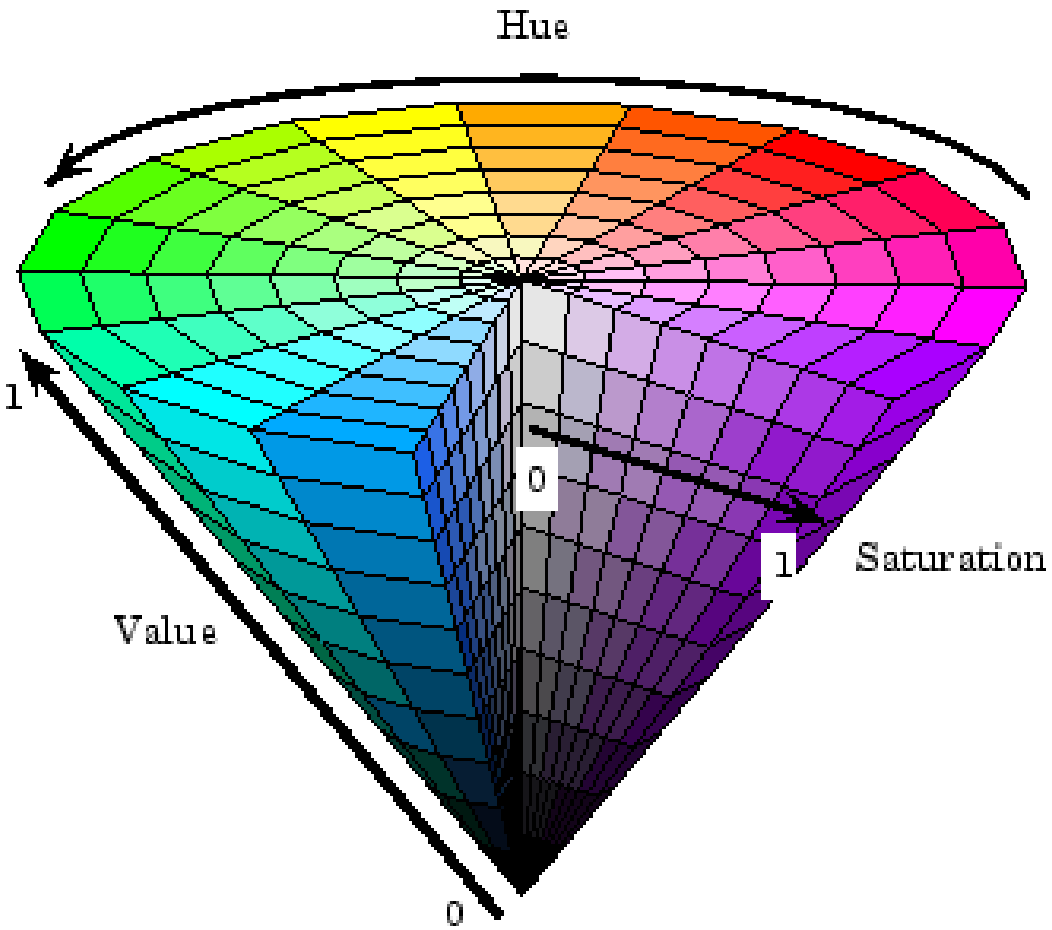
缺点

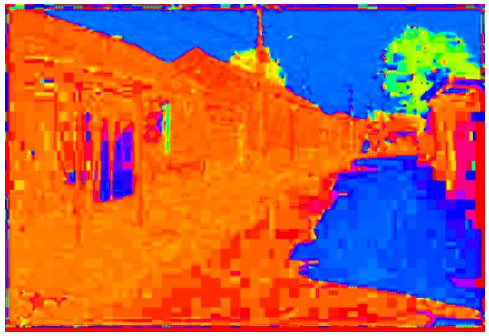
- channels间强烈的相互关联
- Non-perceptual

颜色空间：HSV



直观的颜色空间





H

(S=1,V=1)

S

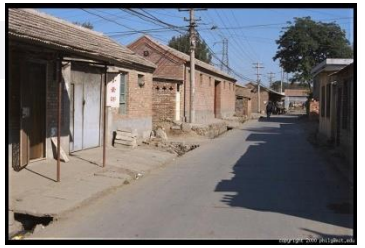
(H=1,V=1)

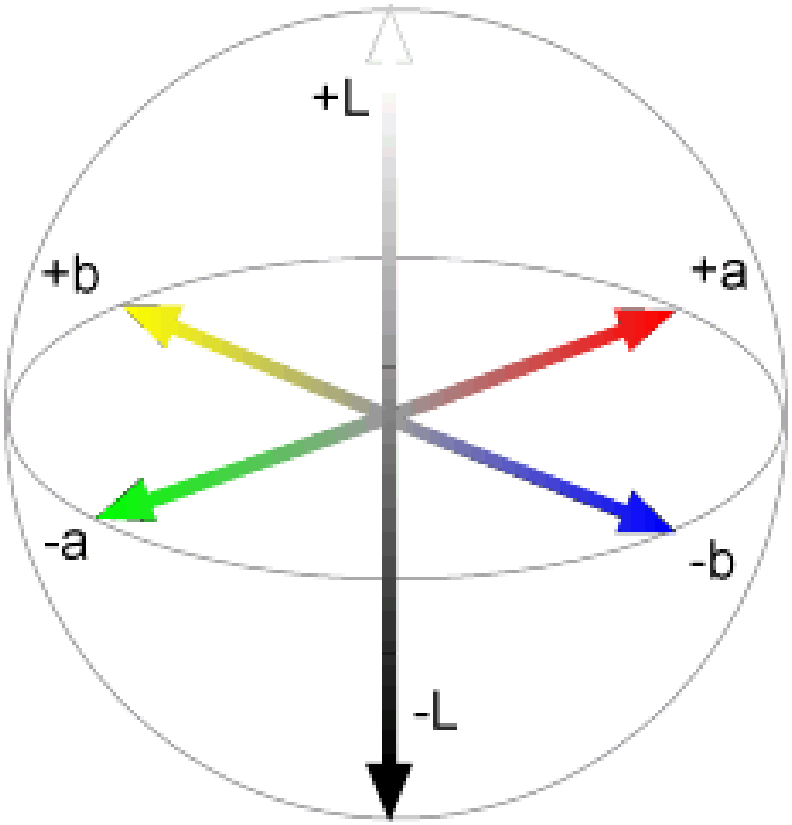
V

(H=1,S=0)

颜色空间：Lab

感知统一(Perceptually uniform)的颜色空间





L

(a=0,b=0)

a

(L=65,b=0)

b

(L=65,a=0)

颜色空间：Lab

- Lab 颜色由亮度或光亮度分量L 和a、b两个色度分量组成。其中a在的正向数值越大表示越红，在负向的数值越大则表示越绿；b在的正向数值越大表示越黄，在负向的数值越大表示越蓝。
- Lab颜色与设备无关， 无论使用何种设备（如显示器、打印机、计算机或扫描仪）创建或输出图像，这种模型都能生成一致的颜色。

颜色空间：CMYK

CMYK颜色模型时减色颜色模型被用在彩印中，CMYK指的是在彩印中使用的四种墨水：cyan，magentra，yellow，key(black)

单独使用每种颜色墨水打印的同一张图片

颜色空间：YUV

被欧洲电视系统所采用的一种颜色编码方法，多用于优化彩色视频信号的传输

分别使用Y， U，V通道表示的图片

颜色空间：YUV

YUV彩色电视信号传输时，将*R*、*G*、*B*改组成亮度信号和色度信号。PAL制式将*R*、*G*、*B*三色信号改组成*Y*、*U*、*V*信号，　其中*Y*信号表示亮度，*U*、*V*信号是色差信号。

*RGB*与*YUV*之间有一定的对应关系

颜色空间转换：Y‘UV444 和 RGB888

$$Y' = 0.299 \times R + 0.587 \times G + 0.114 \times B$$
$$U = -0.147 \times R - 0.289 \times G + 0.436 \times B$$
$$V = 0.615 \times R - 0.515 \times G - 0.100 \times B$$

$$C = Y' - 16$$
$$D = U - 128$$
$$E = V - 128$$

$$R = \text{clamp}((298 \times C + 409 \times E + 128) >> 8)$$
$$G = \text{clamp}((298 \times C - 100 \times D - 208 \times E + 128) >> 8)$$
$$B = \text{clamp}((298 \times C + 516 \times D + 128) >> 8)$$

颜色空间转换：HSV to RGB

$$C = V \times S_{HSV}$$

$$H' = \frac{H}{60^\circ}$$

$$X = C(1 - |H' \bmod 2 - 1|)$$

$$(R_1, G_1, B_1) = \begin{cases} (0,0,0) & \text{if } H \text{ is undefined} \\ (C,X,0) & \text{if } 0 \le H' < 1 \\ (X,C,0) & \text{if } 1 \le H' < 2 \\ (0,C,X) & \text{if } 2 \le H' < 3 \\ (0,X,C) & \text{if } 3 \le H' < 4 \\ (X,0,C) & \text{if } 4 \le H' < 5 \\ (C,0,X) & \text{if } 5 \le H' < 6 \end{cases}$$

$$m = V - C$$

$$(R, G, B) = (R_1 + m, G_1 + m, B_1 + m)$$

$H \in [0^\circ, 360^\circ)$

$S_{HSV} \in [0, 1]$

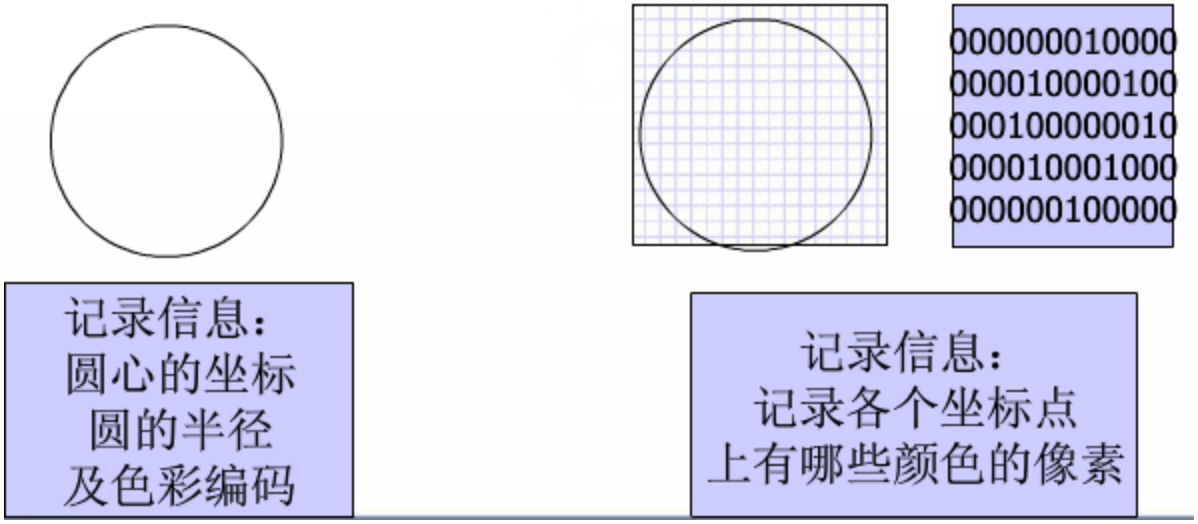
$V \in [0, 1]$

颜色空间转换：RGB和CMYK到Lab

在RGB或CMYK值与Lab 之间没有转换的简单公式，因为RGB和CMYK色彩空间是设备依赖的。RGB或CMYK值首先必须被变换到特定绝对色彩空间中，比如sRGB或Adobe RGB。这种调整将是设备依赖的，但是变换的结果数据是设备无关的，允许把数据变换成CIE 1931色彩空间并接着变换成Lab

图像种类：灰度图与彩色图

图像种类：矢量图与位图


记录信息：
圆心的坐标
圆的半径
及色彩编码
000000010000
000010000100
000100000010
000010001000
000000100000
记录信息：
记录各个坐标点
上有哪些颜色的像素

图像种类：矢量图

矢量图是用一系列绘图指令来表示一幅图，其本质是用数学(更准确地说是几何学)公式描述一幅图像。图像中每一个形状都是一个完整的公式，称为一个对象。

对象是一个封闭的整体，所以定义图像上对象的变化和对象与其他对象的关系对计算机来说是简单的， 所有这些变化都不会影响到图像中的其他对象。

图像种类：矢量图

优点：

- 文件数据量很小；
- 图像质量与分辨率无关。

缺点：

- 不易制作色调丰富或色彩变化太多的图像；
- 绘出来的图像不是很逼真；
- 不易在不同的软件间交换文件。

图像种类：位图

位图是通过许多像素点表示一幅图像，每个像素具有颜色属性和位置属性。

优点：

具有丰富的层次和色彩 。

缺点：

缺少灵活性，分辨率固定；

文件存储量大。

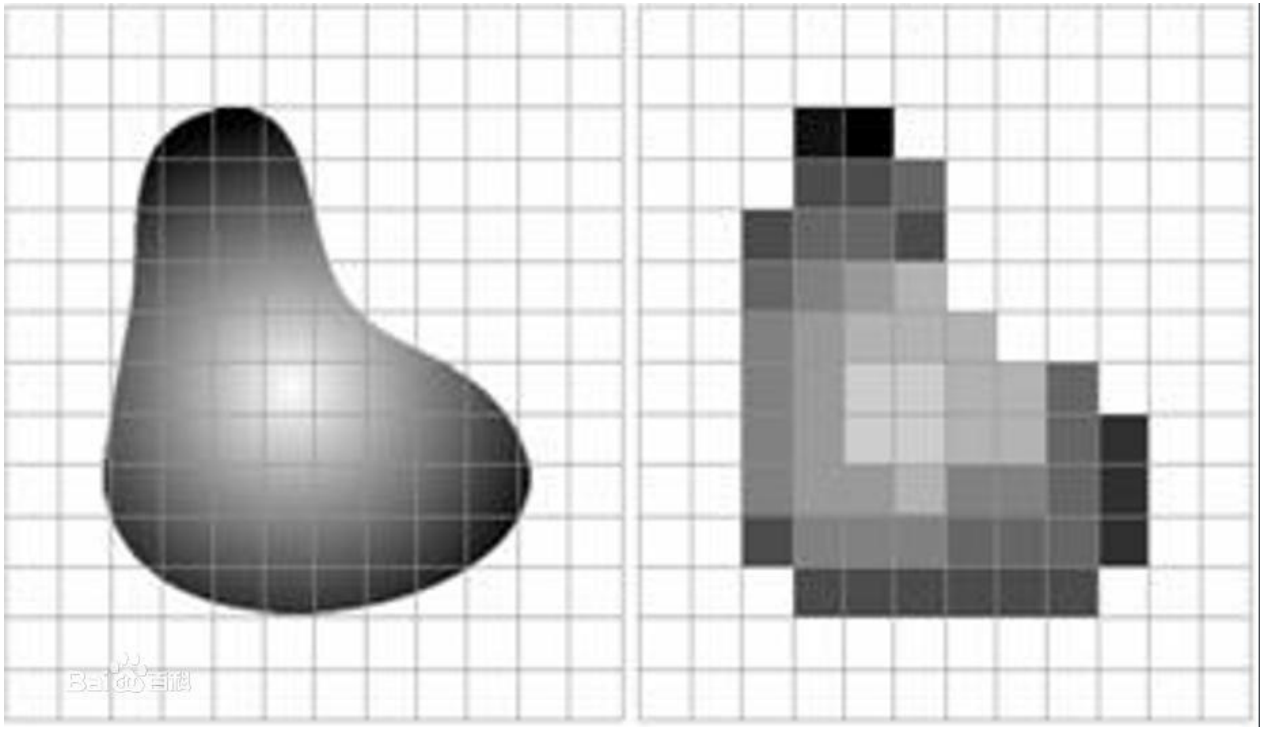
图像数字化

图像数字化是将连续色调的模拟图像经采样量化后转换成数字影像的过程。数字化过程主要分为以下三个步骤：

- 采样
- 量化
- 编码

图像数字化：采样

采样的实质就是要用多少点来描述一幅图像，采样结果质量的高低就是用前面所说的图像分辨率来衡量。



图像数字化：量化

量化是指要使用多大范围的数值来表示图像采样之后的每一个点。量化的结果是图像能够容纳的颜色总数，它反映了采样的质量。

例如：如果以4位存储一个点，就表示图像只能有16种颜色；若采用16位存储一个点，则有2^{16}=65536种颜色。所以，量化位数越来越大，表示图像可以拥有更多的颜色，自然可以产生更为细致的图像效果。

图像数字化：量化原则

- 等间隔量化（均匀量化）—适合像素灰度值在黑白范围较均匀分布的图像。
- 非等间隔量化（非均匀量化）—对图像中像素灰度值频繁出现的灰度值范围，量化间隔取小一些，而对那些像素灰度值极少出现的范围，则量化间隔取大一些。

图像数字化：压缩编码

数字化后得到的图像数据量十分巨大，必须采用编码技术来压缩其信息量。在一定意义上讲，编码压缩技术是实现图像传输与储存的关键。已有许多成熟的编码算法应用于图像压缩。常见的有图像的预测编码、变换编码、分形编码、小波变换图像压缩编码等。

图像格式：BMP格式

- BMP是Bitmap的缩写，扩展名是.bmp，是微软特为windows环境图像设计
- 图像是静止的
- 可以多种彩色模式保存图像，如16色，256色，24bit真彩色，最新版本的BMP格式允许32bit真彩色
- 从图像的左下角为起点存储图像

图像格式：BMP格式

BMP格式的图像文件结构

- 文件头的长度固定是54字节
- 调色板数据用于描述所有不超过256色的图像模式，图像采用24bit真彩色模式或者更高模式，该图像文件中的调色板数据就不再描述相关信息
- BMP格式的图像文件既可以采用压缩算法对其进行处理，也可以不压缩

图像格式：TIFF格式

- TIFF是Tag Image File Format的缩写
- TIFF格式的图像文件由Aldus公司和与微软联手开发
- 扩展名.tif
- 支持从单色模式到32bit真彩色模式的所有图像
- 不针对某一个特定的操作平台
- 数据结构是可变的，文件具有可改写性，使用者可向文件写入相关信息
- 具有多种数据压缩方式

图像格式：TIFF格式

TIFF格式的图像文件结构

- IFH由8个字节组成
- IFD标识的内容包括指示标识信息的代号，数据类型说明，数据值，文件数据量等
- 数据区用来存放数据并指明压缩方法，数据排列，数据分割的信息

图像格式：TGA格式

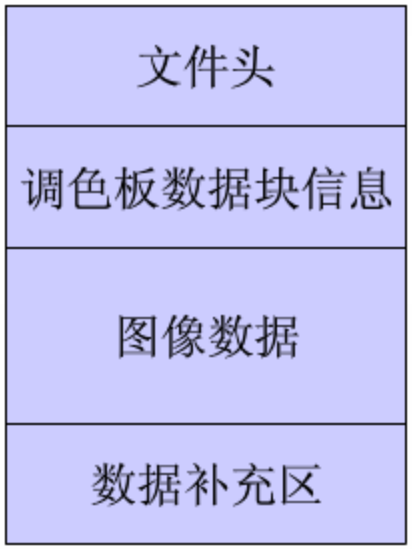
- TGA是Target Image Format的缩写
- 由Truevision公司开发，最初的目的是支持本公司生产的Targa图形卡
- 目前成为世界的通用格式，被应用到多个专门的领域。（动画制作，模拟显示，影视画合成）
- 扩展名是.tag
- 支持任意尺寸的图像
- 支持1bit单色到32bit真彩色模式的所有图像，特别适合影视广播级的动画制作
- 图像的存储具有可选择性，图像数据既可以按照从上到下，从左到右的顺序进行存储，也可以按照相反的顺序存储

图像格式：TGA格式

- 图像的存储具有可选择性，图像数据既可以按照从上到下，从左到右的顺序进行存储，也可以按照相反的顺序存储
- TGA格式的图像对硬件的依赖性强，如果显卡不具备24bit或32bit的显示能力，该格式的图像将不能正确显示

图像格式：TGA格式

TGA格式的图像文件结构



- 文件头主要用于说明TGA文件的出处，颜色映象表类型，图像数据存储类型，图像数据存储顺序等内容
- 调色板数据信息是可选择部分，包括TGA图像文件格式的调色板数据块构成方式，图像数据的组织方式等
- 图像数据区用于存储大量的图像数据
- 数据补充区用于表明当前文件是新版本文件，分为开发者目录区和扩充数据区

图像格式：GIF格式

- GIF是Graphics Interchange Format的缩写
- 由Compuserve公司于1987年推出，主要为了网络传输和BBS用户使用图像文件而设计
- 目前GIF格式的图像文件已经是网络传输和BBS用户使用最频繁的文件格式
- 扩展名是.gif
- GIF格式是世界通用的图像格式，特别适用于动画制作，网页制作以及演示文稿制作等方面
- GIF格式的图像文件适用于各种个人计算机和许多UNIX工作站，并且可以再不同输入，输出设备之间方便的传送

图像格式：GIF格式

- 对于灰度图像表现最佳
- 采用改进的LZW压缩算法处理图像数据
- 不支持24bit彩色模式，最多存储256色
- 有GIF87a和GIF89a两个版本，GIF89a是89年推出的允许一个文件存多个图像，可实现动画功能

图像格式：JPEG格式

- 全称是Joint Photograph Coding Experts Group
- 由ISO和CCITT共同制定，并在1992年后被广泛采纳成为国际标准。可以压缩到原图像的百分之一（100:1）
- 是一种基于DCT的静止图像压缩和解压缩算法
- 在存储容量有限的条件下进行携带和传输，如网络传输
- JPEG 2000是基于小波变换的图像压缩标准，由Joint Photographic Experts Group组织创建和维护。JPEG 2000通常被认为是未来取代JPEG（基于离散余弦变换）的下一代图像压缩标准。

图像格式：PNG格式

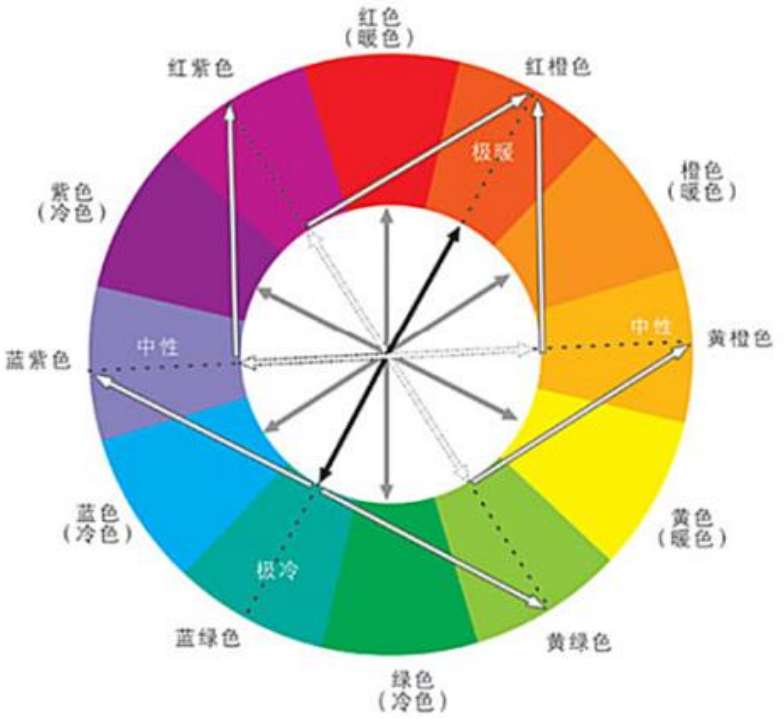
- 全称是Portable Network Graphics
- 是一种无损压缩的位图图形格式
- 支持256色调色板技术以产生小体积文件
- 最高支持48位真彩色图像以及16位灰度图像
- 支持Alpha通道的透明/半透明特性
- 支持图像亮度的Gamma校准信息
- 支持存储附加文本信息，以保留图像名称、作者、版权、创作时间、注释等信息。
- 渐近显示和流式读写，适合在网络传输中快速显示预览效果后再展示全貌

图像格式：PNG格式

- 使用CRC防止文件出错
- 开发目标是改善并取代GIF作为适合网络传输的格式而不需要专利许可，被广泛用于互联网及其他方面
- 最新的PNG标准允许在一个文件内存储多幅图像

色彩的神秘魔力

- 为什么保险柜多为黑色？
- 为什么冰箱多为白色？
- 为什么被子多为白色和淡蓝色？
- 什么颜色让你容易入睡？
- 什么颜色又能让你保持清醒？



色彩可以使人的时间感发生混淆

- 人看着红色，会感觉时间比实际长，看着蓝色，则感觉时间比实际时间短。
- 这个现象在日常生活中非常常见
  - 休闲潜水：大多数潜水者将一个可以持续40~50分钟的氧气瓶的氧气用光后，却感觉在水中只下潜了20分钟左右。
  - 灯光照明：青白色的荧光灯下，人会感觉时间过得很快，而在温暖的白炽灯下，就会感觉时间过得很慢。

快餐店不适合等人

- 快餐店的装潢以桔黄色或红色为主
  - 这两种颜色有使人心情愉悦、兴奋和增进食欲的作用；
  - 使人感觉时间漫长，如果在这样的环境中等人，会越来越烦躁。
- 比较适合约会、等人的场所
  - 色调偏冷的咖啡馆

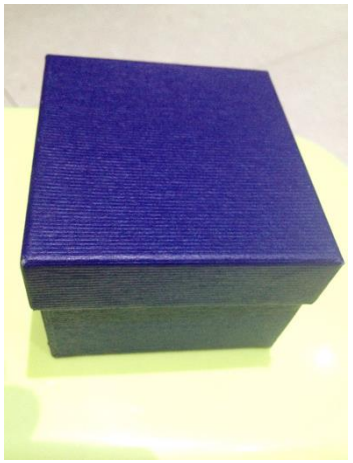
运用色彩心理学营造理想的会议环境

- 长时间的会议是所有公司职员挥之不去的烦恼
- 公司的会议室最好以蓝色为基调进行室内装潢
  - 蓝色会让人觉得时间过得很快；
  - 蓝色可以使人放松。

颜色的“重量”

- 色彩是有重量的
  - 同等重量的白色箱子、黄色箱子、蓝色箱子相比哪个感觉更重？





- 不同的颜色“重量”差别有多大？
  - 实验结果表明黑色的箱子与白色的箱子相比，前者看上去要重 1.8倍。
  - 明亮度低的颜色比明亮度高的颜色感觉重。

颜色的“重量”

- 为什么保险柜多为黑色？
  - 使用黑色可以大大增加保险柜的心理重量，从而有效防止被盗的发生。
- 为什么包装纸箱多为浅褐色？
  - 利用再生纸制造而成的，保持了纸浆的原色；
  - 浅褐色可以使人感觉包装纸箱的重量比较轻。

颜色的“冷暖”

- 颜色有让人心理上感觉暖与感觉冷之分
  - 暖色：红色、橙色、粉色
  - 冷色：蓝色、绿色、蓝绿色
- 冷暖程度会受到颜色亮度的影响
  - 明度高的颜色，会使人感觉寒冷或凉爽；明度低的颜色，会使人感觉温暖。
- 在心理上因人而异
  - 由不同的成长环境和个人经验造成的。

物理上反光不吸热的颜色与吸光吸热的颜色

- 颜色的反光与吸热
  - 有些颜色可以反射光线而不吸热量，使物体实际温度比较低，而有些颜色吸收光线的同时还会吸收热量，使物体实际温度比较高；
  - 白色、浅色（如浅蓝、浅黄）等明亮的颜色可以反射光线，但却不容易吸收热量；
  - 黑色、深色（如深蓝、深红、深绿）等颜色容易吸收光线和热量。
- 应用：建筑与设计、服装、太阳能设备

颜色可以将物体放大或缩小

- 膨胀色
  - 红色、橙色和黄色等暖色系颜色
- 收缩色
  - 蓝色、蓝绿色等冷色系颜色
- 明亮度的影响
  - 明亮度高的颜色为膨胀色，可以将物体放大
- 应用
  - 穿衣、室内装修

颜色的凹凸效果

- 前进色与后退色
  - 前进色：看起来显得凸出的颜色，包括红色、橙色和黄色等暖色，主要为高彩度的颜色；
  - 后退色：看起来显得凹陷的颜色，包括蓝色和蓝紫色等冷色，主要为低彩度的颜色。
- 广泛应用
  - 广告牌颜色
  - 应用于工厂，提高工人工作效率
  - 让房间看起来更加宽敞

颜色会影响胃口

- 让人胃口打开的颜色
  - 红色、橙色和黄色等鲜艳的颜色都有增进食欲的效果。
- 使人食欲减退的颜色
  - 蓝色、紫色、紫黑色等颜色。
- 颜色与照明的配合
  - 盛食物的器皿以白色居多，这是因为白色可以更好地突出食物颜色的缘故。

催人入眠、使人清醒的颜色

- 具有催眠作用的颜色
  - 蓝色：可以降低血压，消除紧张感，从而起到镇定的作用
  - 绿色：可以使人从心理上得到放松
- 使人清醒的颜色
  - 红色：可以增强人的紧张感，使血压升高
- 应用
  - 为什么被子多为白色和淡蓝色？

睡眠与照明的关系

- 照明的颜色与睡眠有着紧密的关系
  - 照明的颜色会对人体内一种叫做褪黑激素的荷尔蒙的分泌产生影响。
  - 暖光促进褪黑激素分泌，冷光抑制褪黑激素分泌。
- 褪黑激素
  - 促使人自然入睡的荷尔蒙。
  - 改善人体机能、提高免疫力和抵抗力的功能。
  - 通常在夜间分泌，而青白色的荧光灯有抑制褪黑激素分泌的作用。

图像滤波

- 图像滤波: 在图像上的每个位置利用邻域信息计算函数
- 非常重要!
  - 增强图像：去噪(denoise)，调整大小(resize)，增加对比度(increase contrast)等
  - 从图像中提取信息：纹理(texture)，边缘(edges)，区分点(distinctive points)等
  - 检测模式：模板匹配(template matching)

图像滤波: 均值滤波

$g[\cdot,\cdot]$

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

图像滤波: 均值滤波

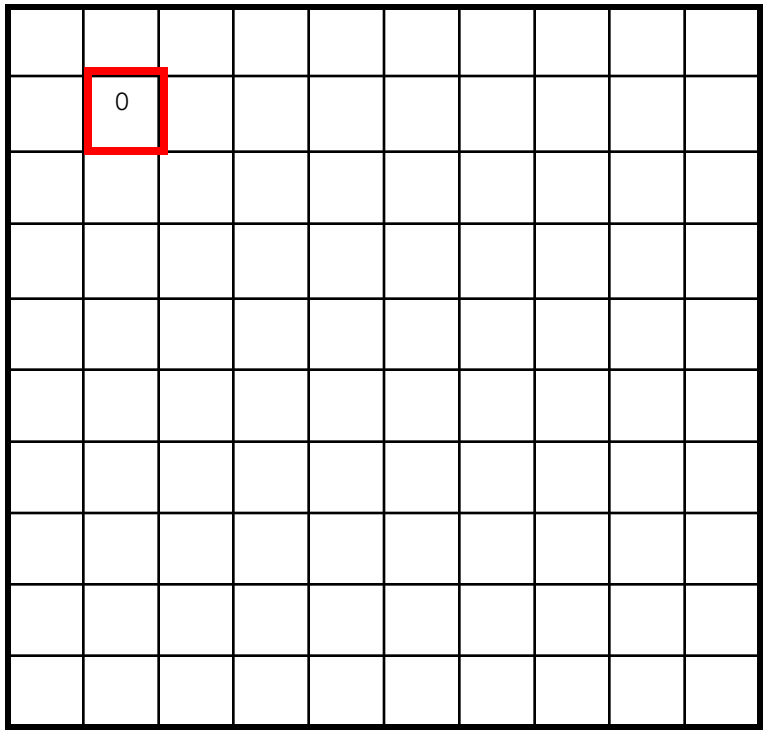
$g[\cdot,\cdot]$ $\frac{1}{9}$

1	1	1
1	1	1
1	1	1

$f[.,.]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

$h[.,.]$



$$h[m,n] = \sum_{k,l} g[k,l]\ f[m+k,n+l]$$

图像滤波: 均值滤波

$g[\cdot,\cdot]$ $\frac{1}{9}$

1	1	1
1	1	1
1	1	1

$f[.,.]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

$h[.,.]$

	0	10							

$$h[m,n] = \sum_{k,l} g[k,l]\ f[m+k,n+l]$$

图像滤波: 均值滤波

$g[\cdot,\cdot]$ $\frac{1}{9}$

1	1	1
1	1	1
1	1	1

$f[.,.]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

$h[.,.]$

	0	10	20						

$$h[m,n] = \sum_{k,l} g[k,l]\ f[m+k,n+l]$$

图像滤波: 均值滤波

$g[\cdot,\cdot]$ $\frac{1}{9}$

1	1	1
1	1	1
1	1	1

$f[.,.]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

$h[.,.]$

	0	10	20	30					

$$h[m,n] = \sum_{k,l} g[k,l]\ f[m+k,n+l]$$

图像滤波: 均值滤波

$g[\cdot,\cdot]$

$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

$f[.,.]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

$h[.,.]$

	0	10	20	30	30				

$$h[m,n] = \sum_{k,l} g[k,l]\ f[m+k,n+l]$$

图像滤波: 均值滤波

$g[\cdot,\cdot]$ $\frac{1}{9}$

1	1	1
1	1	1
1	1	1

$f[.,.]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

$h[.,.]$

	0	10	20	30	30				
				?					

$$h[m,n] = \sum_{k,l} g[k,l]\ f[m+k,n+l]$$

图像滤波: 均值滤波

$g[\cdot,\cdot]$ $\frac{1}{9}$

1	1	1
1	1	1
1	1	1

$f[.,.]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

$h[.,.]$

	0	10	20	30	30				
						?			
				50					

$$h[m,n] = \sum_{k,l} g[k,l]\ f[m+k,n+l]$$

图像滤波: 均值滤波

$g[\cdot,\cdot]$ $\frac{1}{9}$

1	1	1
1	1	1
1	1	1

$f[.,.]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

$h[.,.]$

	0	10	20	30	30	30	20	10	
	0	20	40	60	60	60	40	20	
	0	30	60	90	90	90	60	30	
	0	30	50	80	80	90	60	30	
	0	30	50	80	80	90	60	30	
	0	20	30	50	50	60	40	20	
	10	20	30	30	30	30	20	10	
	10	10	10	0	0	0	0	0	

$$h[m,n] = \sum_{k,l} g[k,l]\ f[m+k,n+l]$$

图像滤波: 均值滤波

均值滤波做了什么?

- 把每个像素用它的邻域的均值进行替换
- 达到平滑的效果

$g[\cdot,\cdot]$

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

图像滤波: 均值滤波

图像滤波: 线性滤波练习

原始图

0	0	0
0	1	0
0	0	0

?

图像滤波: 线性滤波练习

原始图

0	0	0
0	1	0
0	0	0

滤波后(没有改变)

图像滤波: 线性滤波练习

原始图

0	0	0
0	0	1
0	0	0

?

图像滤波: 线性滤波练习

原始图

0	0	0
0	0	1
0	0	0

向左平移一个像素

图像滤波: 线性滤波练习

原始图

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad ?$$

(注意filter的总和是1)

图像滤波: 线性滤波练习

原始图

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

-锐化滤波器

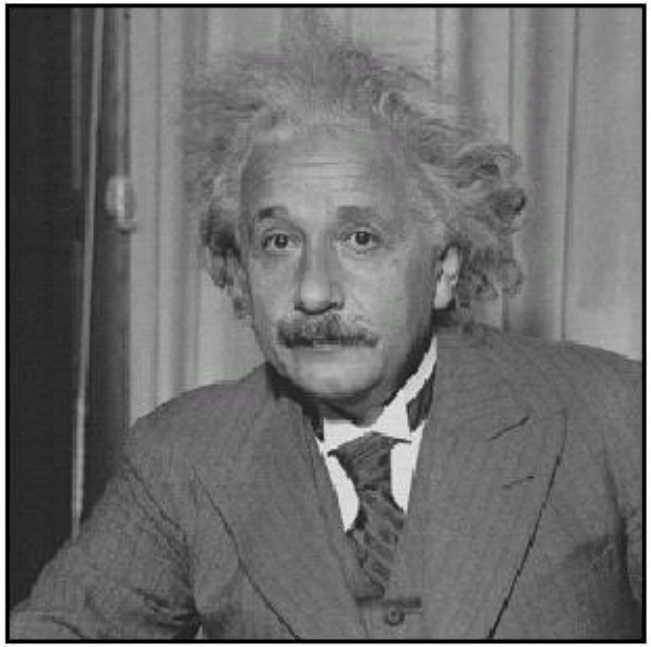
图像滤波: 线性滤波练习

原始图

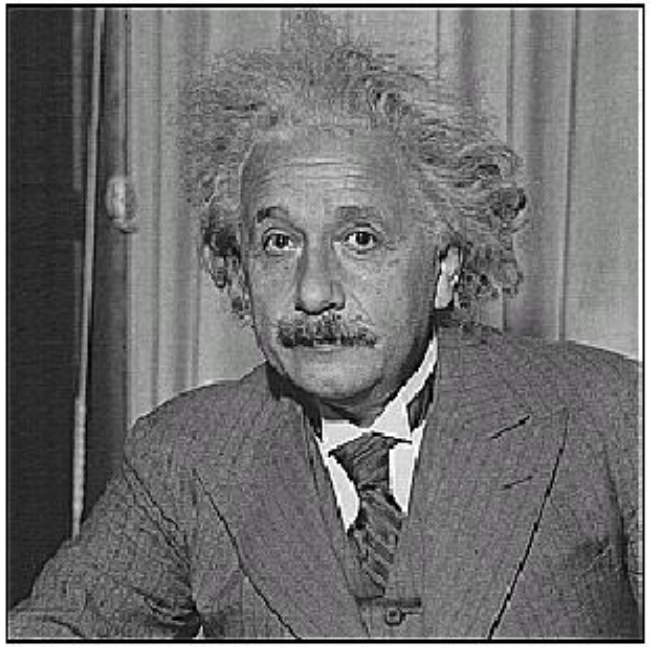
$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

-锐化滤波器

图像滤波: 锐化

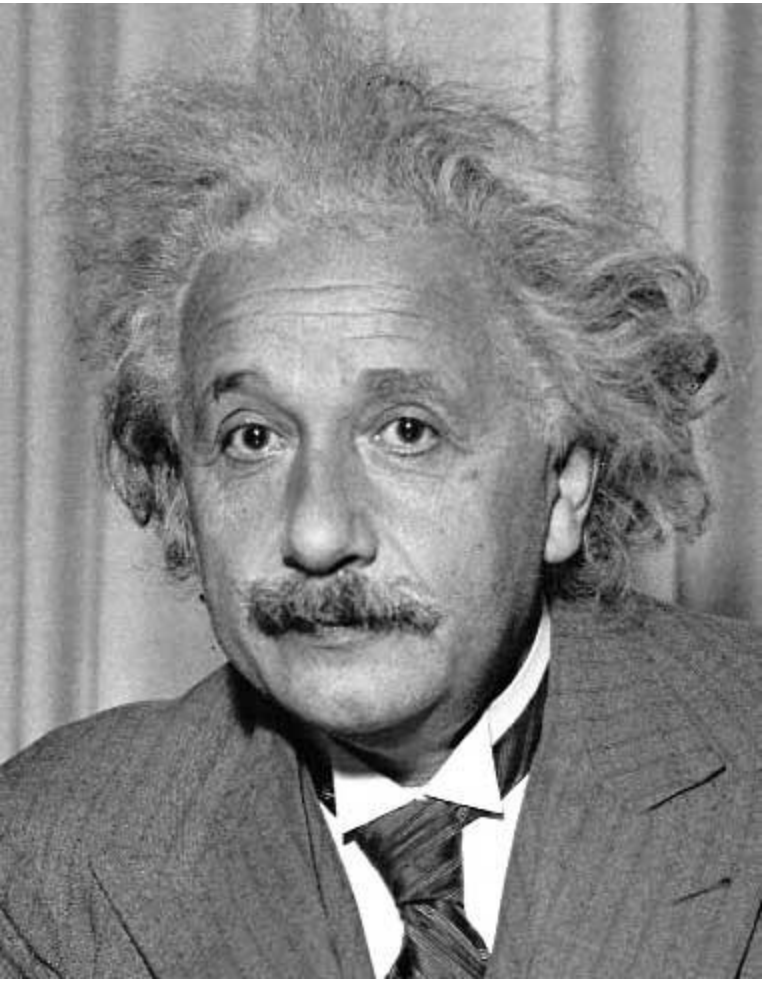


before



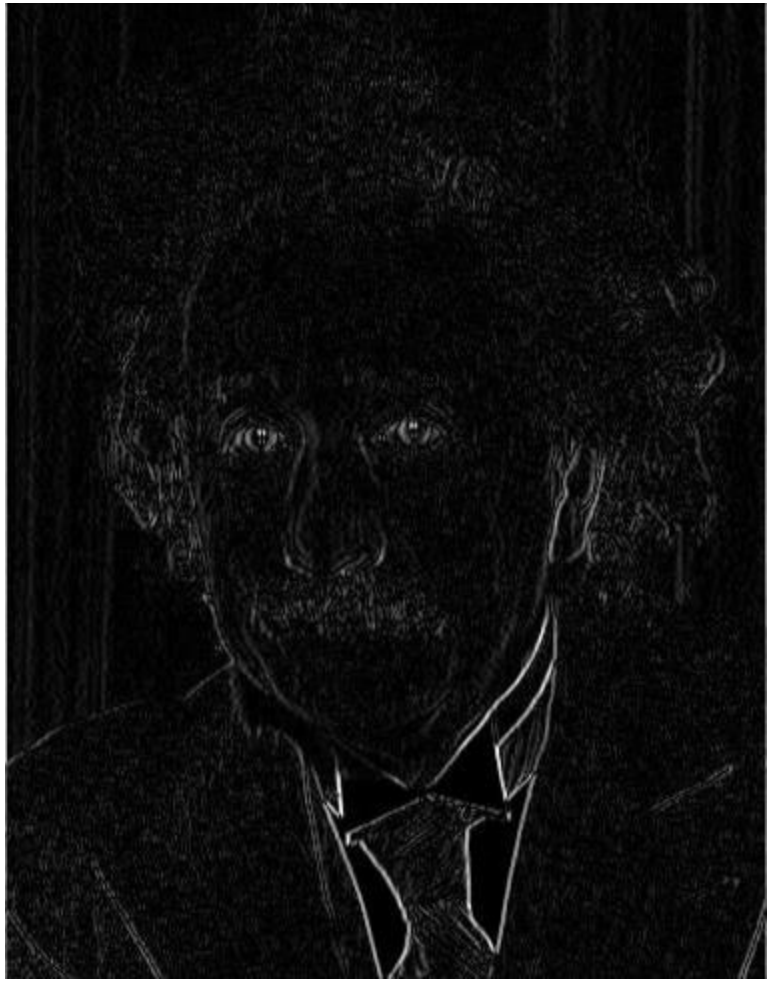
after

图像滤波: 其它滤波器



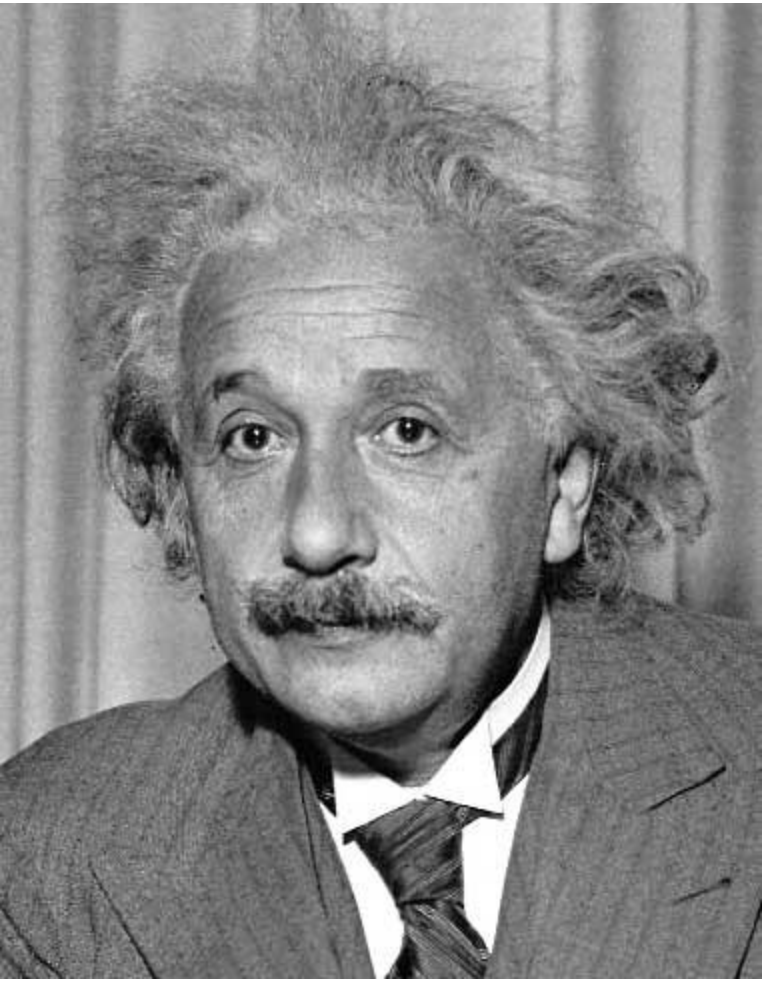
1	0	-1
2	0	-2
1	0	-1

Sobel



垂直边缘(绝对值)

图像滤波: 其它滤波器



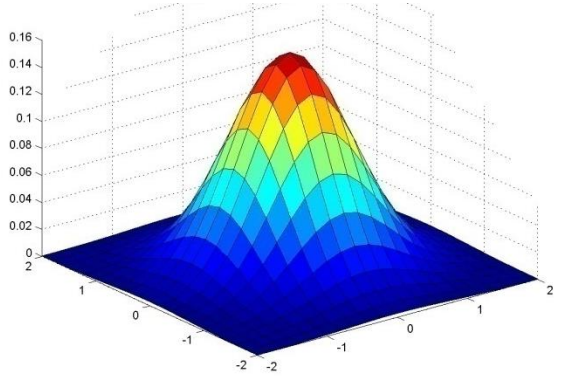
1	2	1
0	0	0
-1	-2	-1

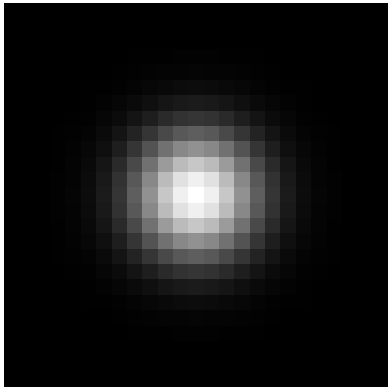
Sobel

水平边缘(绝对值)

图像滤波: 高斯滤波

- 离当前像素点越近对权重贡献越大



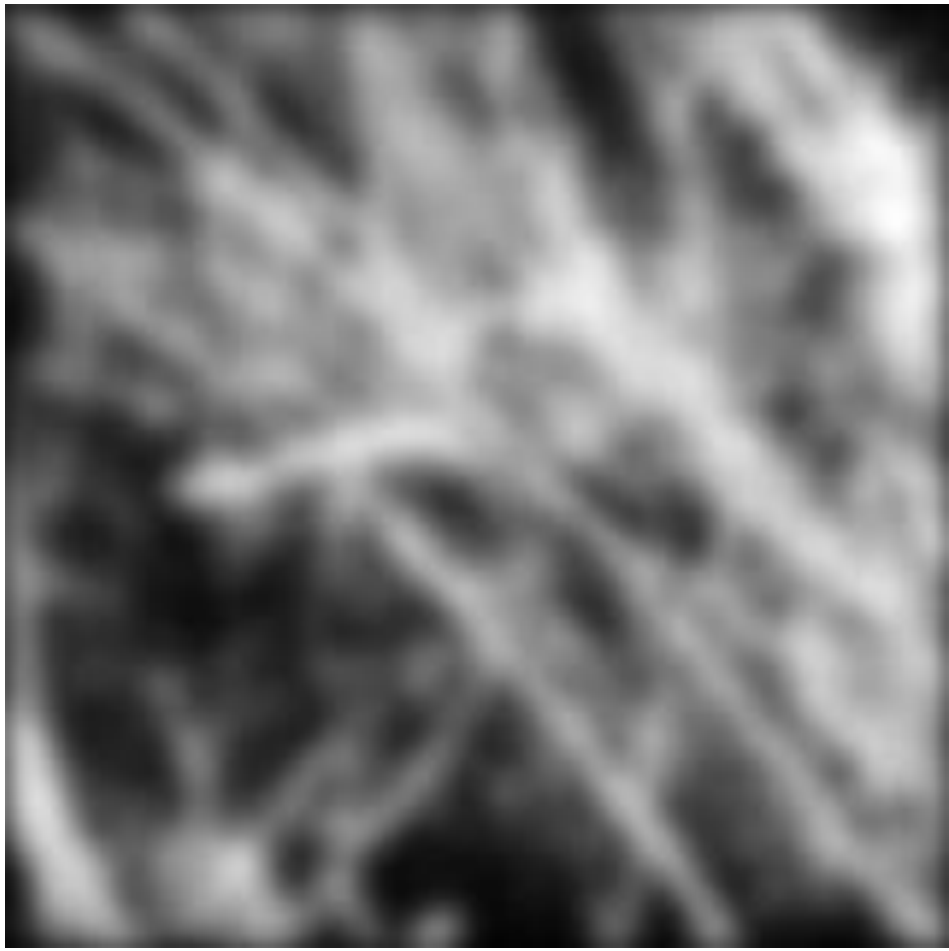


0.003	0.013	0.022	0.013	0.003
0.013	0.059	0.097	0.059	0.013
0.022	0.097	0.159	0.097	0.022
0.013	0.059	0.097	0.059	0.013
0.003	0.013	0.022	0.013	0.003

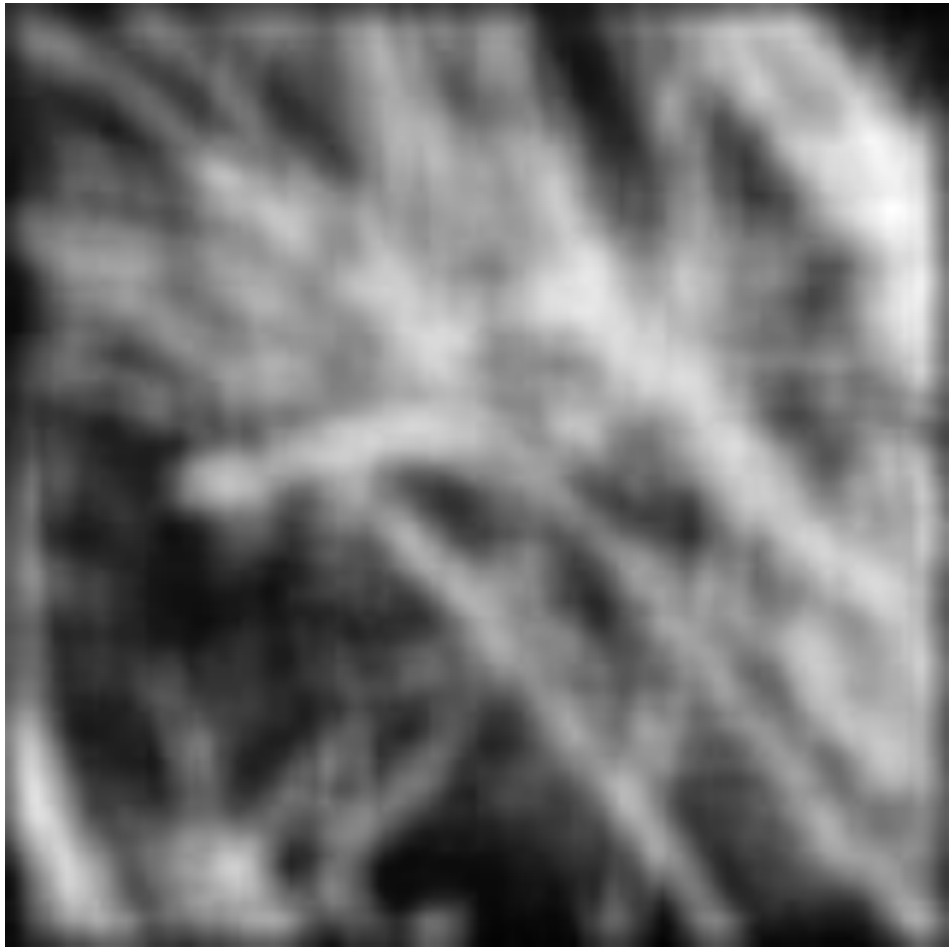
5 x 5, $\sigma = 1$

$$G_\sigma = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

图像滤波: 高斯滤波进行平滑



图像滤波: 均值滤波进行平滑

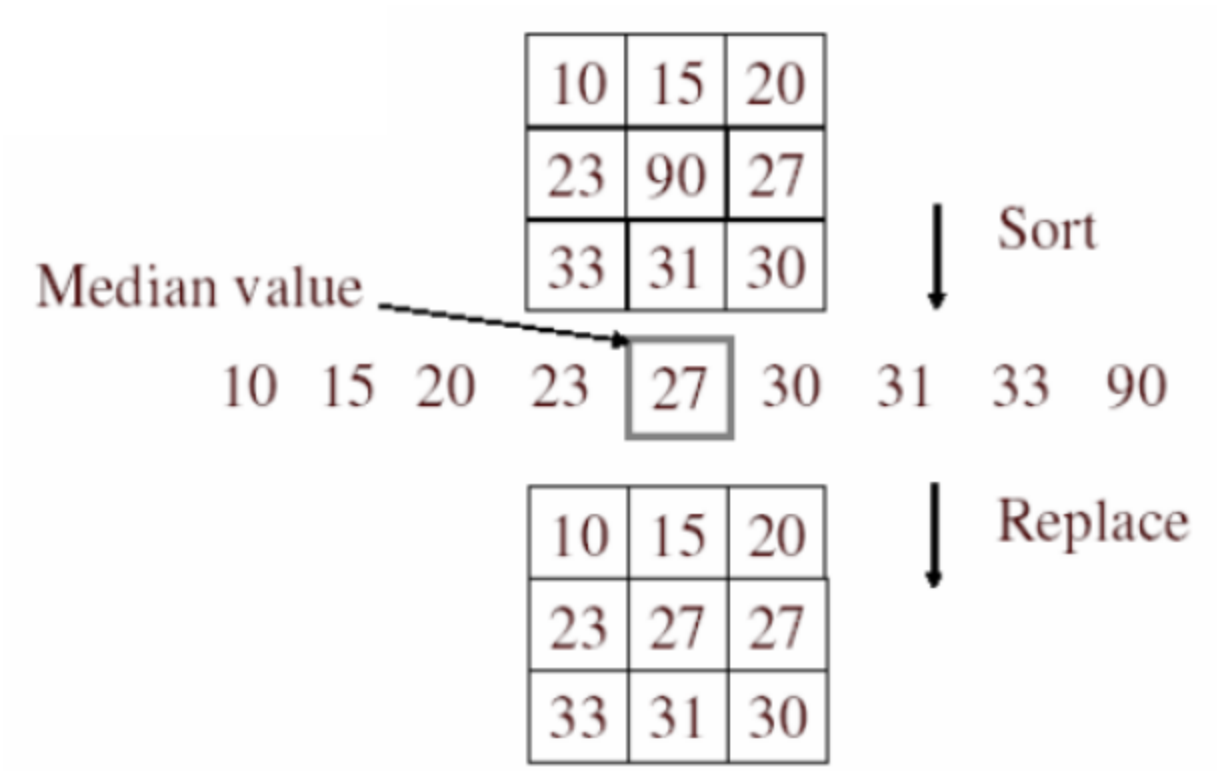


图像滤波:高斯滤波

- 从图像中移除高频部分（低通滤波），所以图像变得更加平滑
- 和自身的卷积（convolution）是另外一个高斯，用宽度是σ的高斯核卷积两次和用宽度是sqrt(2) σ进行一次卷积的效果是一样的
- 可分的核，可以把2D的高斯分解成两个1D的高斯

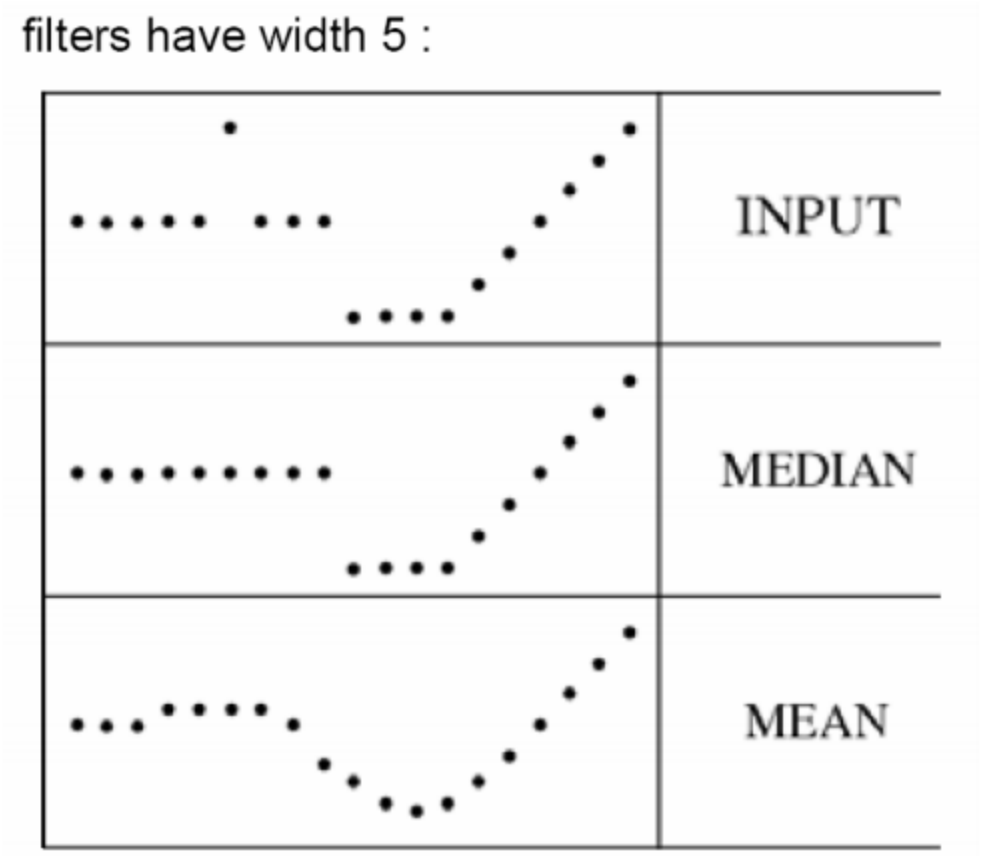
图像滤波: 中值滤波

- 从邻域窗口中选择中值强度的像素值



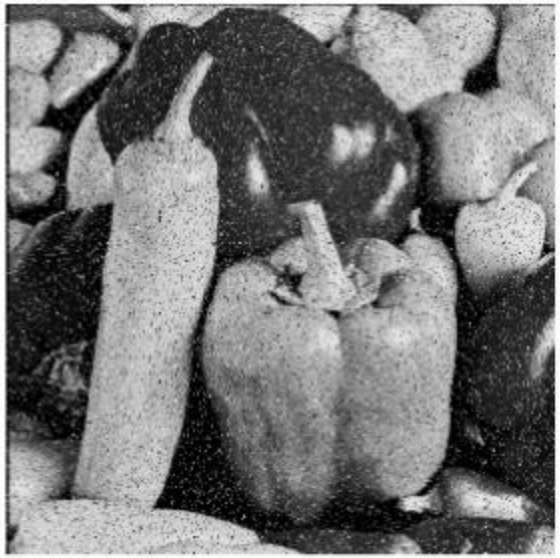
图像滤波: 中值滤波

- 中值滤波的优势是对outliers鲁棒

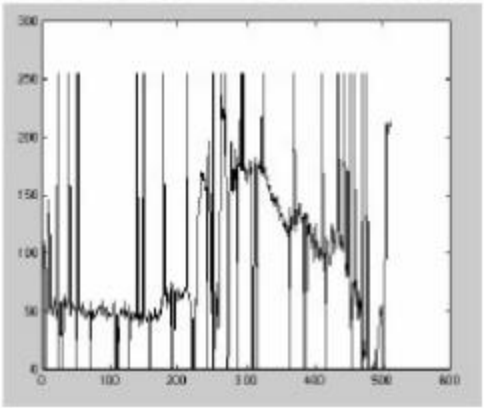


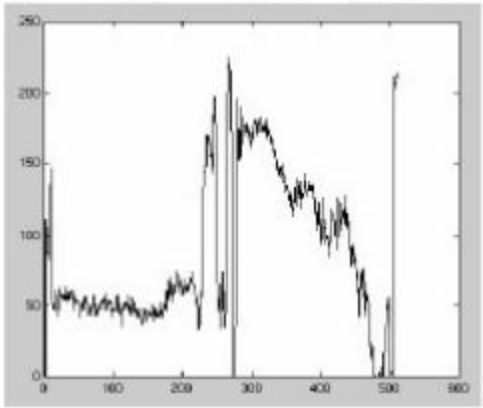
图像滤波: 中值滤波

Salt-and-pepper noise



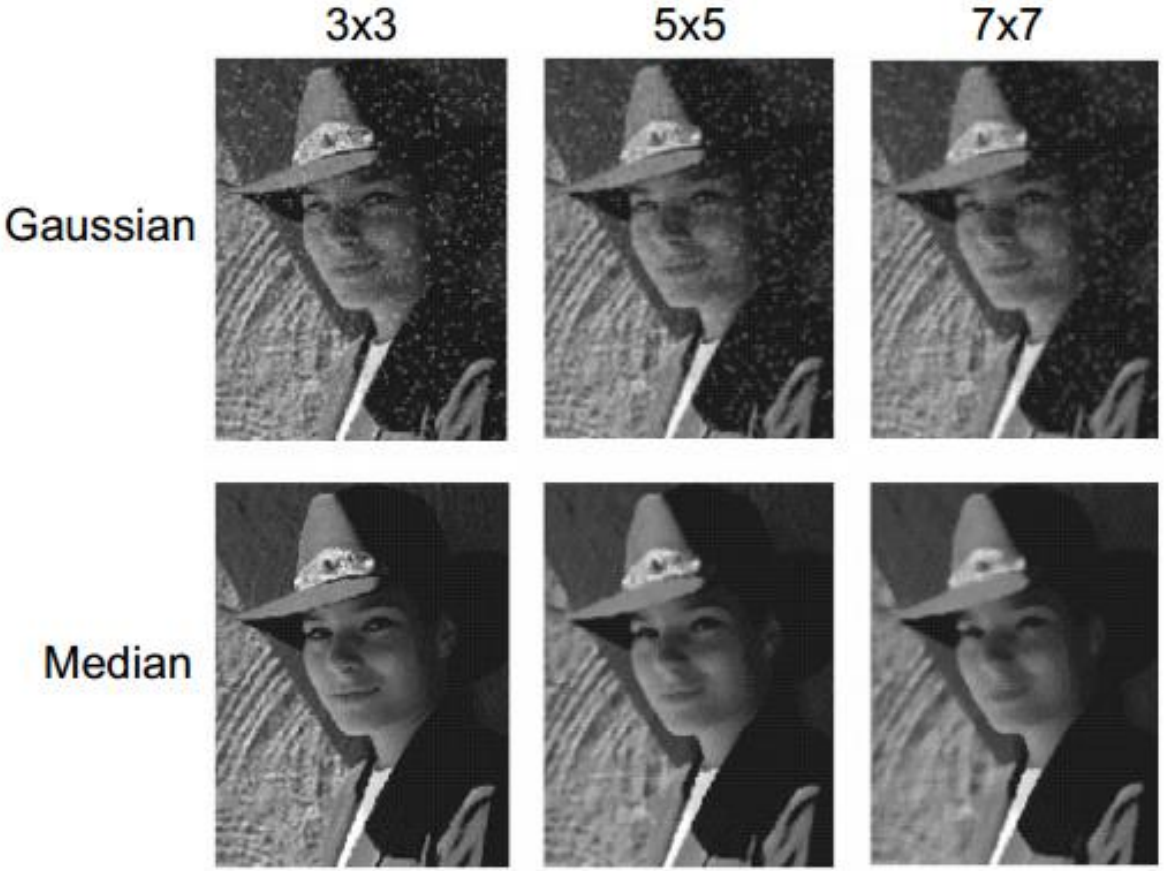
Median filtered





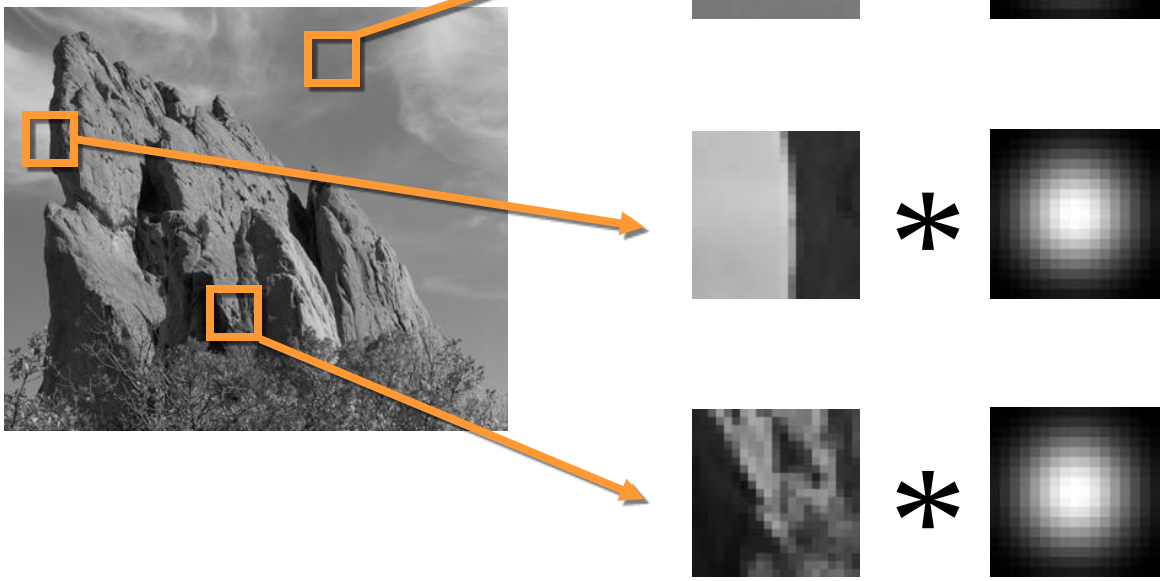
MATLAB: medfilt2(image, [h w])

图像滤波: 中值滤波 vs 高斯滤波



图像滤波: 双边滤波

输入

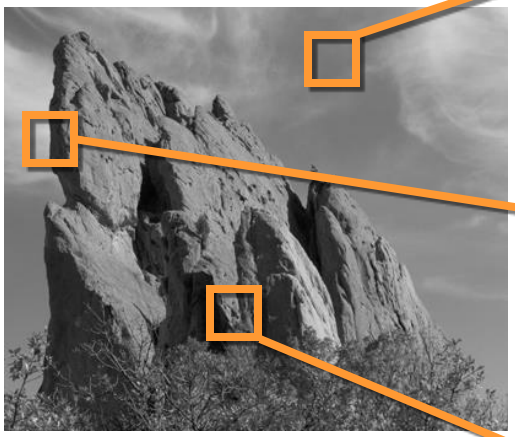


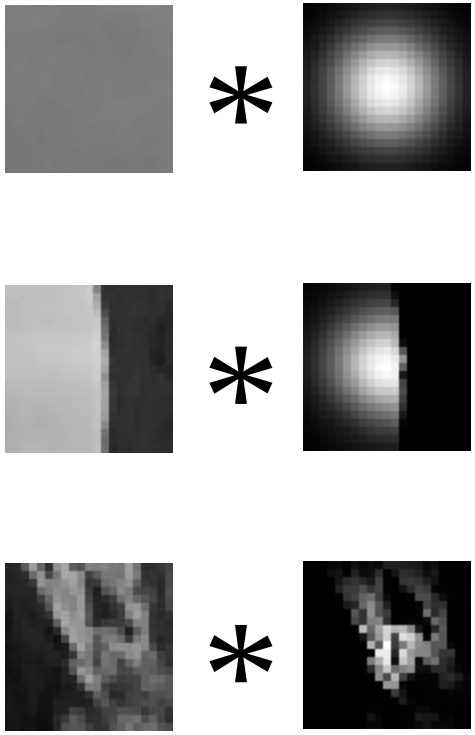
输出

每个像素相同的高斯核

图像滤波: 双边滤波

输入





输出

核的形状根据图像内容

图像滤波: 双边滤波

平滑并保持边界

new

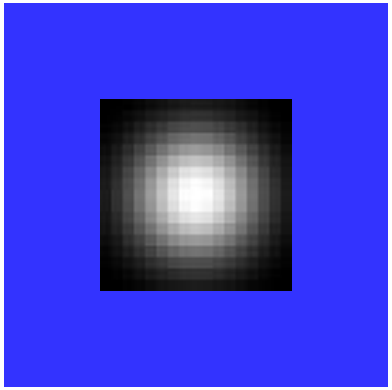
not new

new

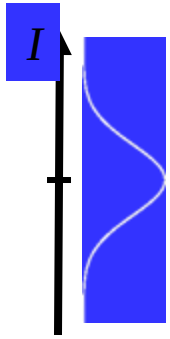
$$BF[I]_{\mathbf{p}} = \frac{1}{W_{\mathbf{p}}} \sum_{\mathbf{q} \in S} G_{\sigma_s}\left(\|\mathbf{p}-\mathbf{q}\|\right) G_{\sigma_r}\left(|I_{\mathbf{p}}-I_{\mathbf{q}}|\right) I_{\mathbf{q}}$$

normalization factor

space weight

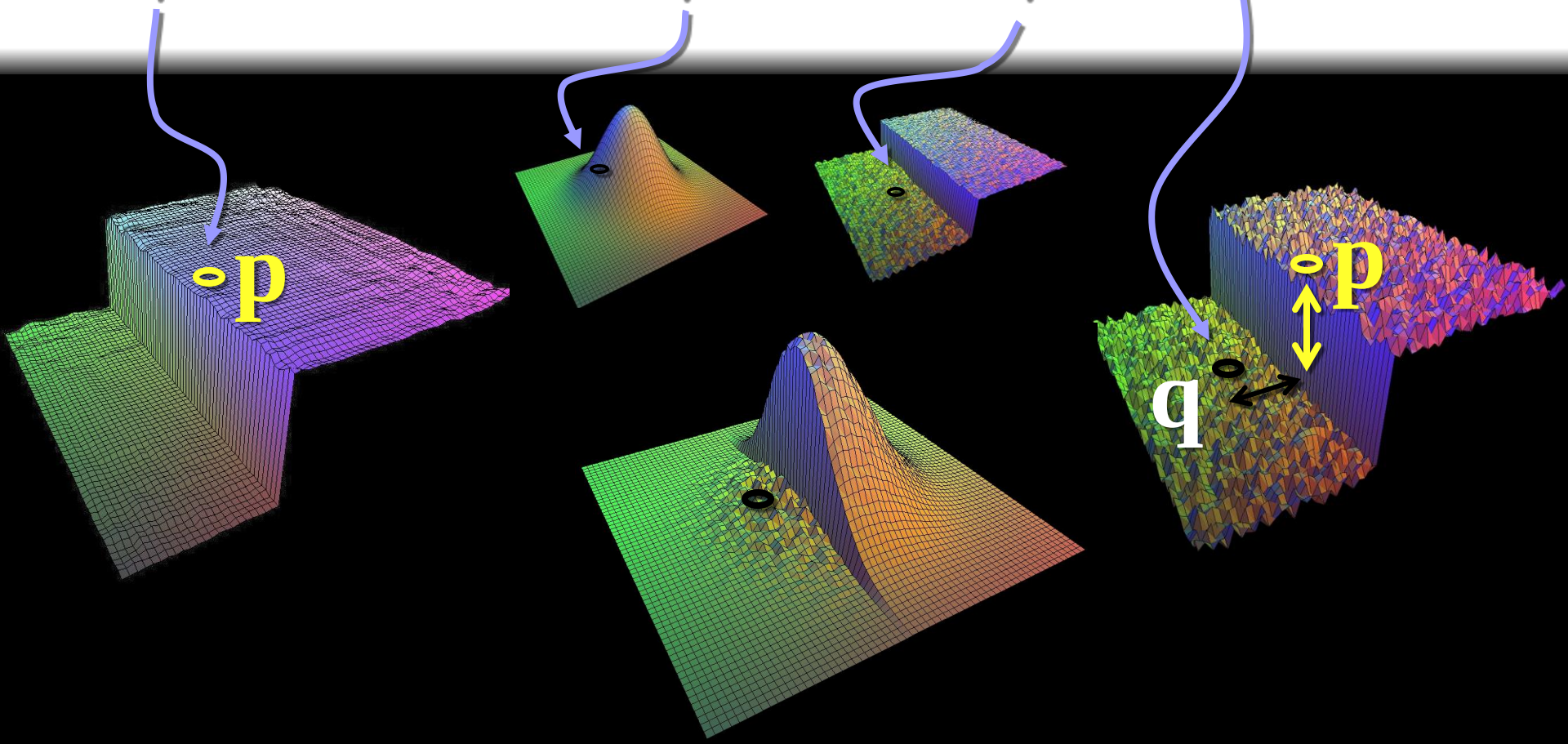


range weight



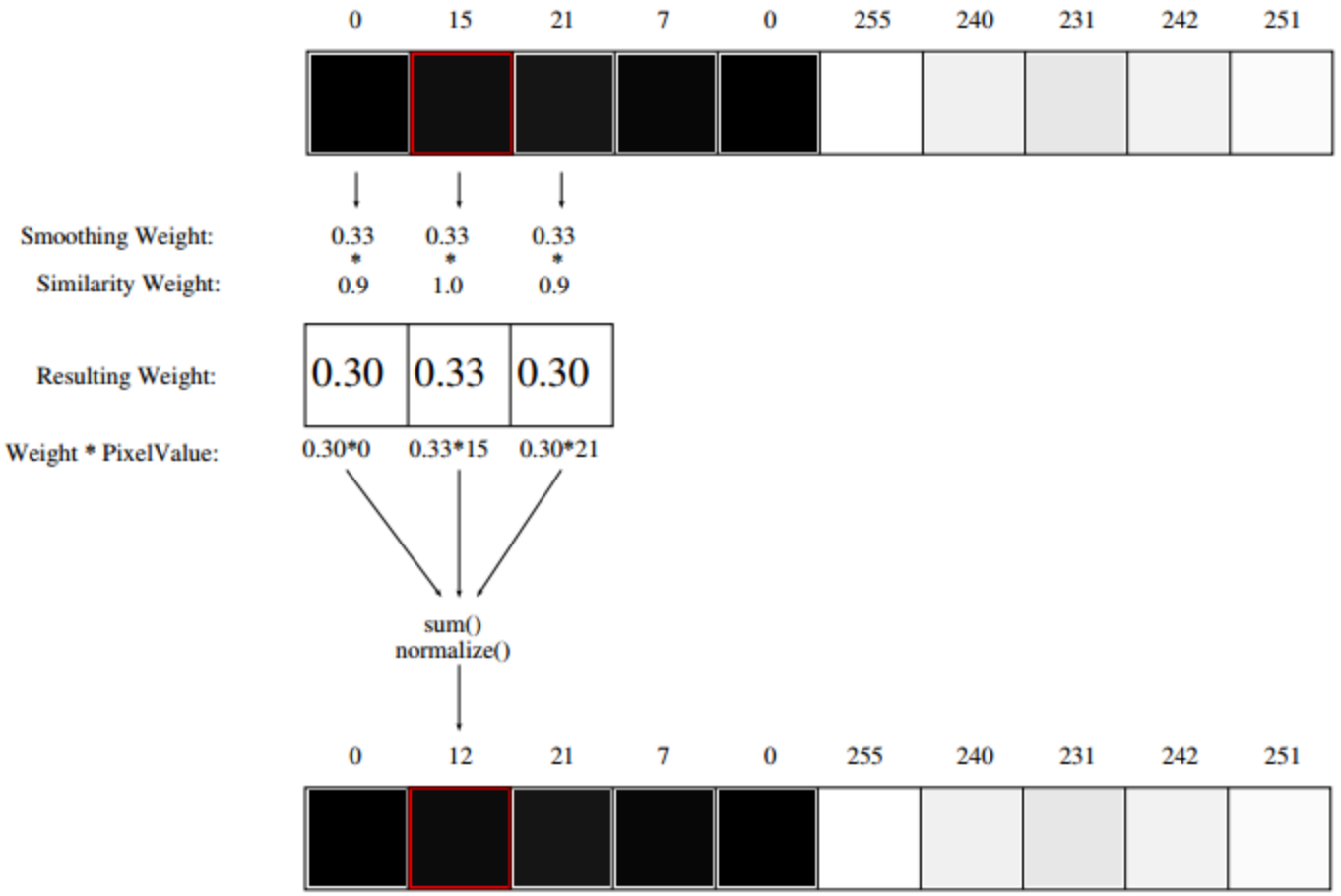
图像滤波: 双边滤波

$$BF[I]_{\mathbf{p}} = \frac{1}{W_{\mathbf{p}}} \sum_{\mathbf{q} \in S} G_{\sigma_s}(\|\mathbf{p}-\mathbf{q}\|)\, G_{\sigma_r}(|I_{\mathbf{p}}-I_{\mathbf{q}}|)\, I_{\mathbf{q}}$$

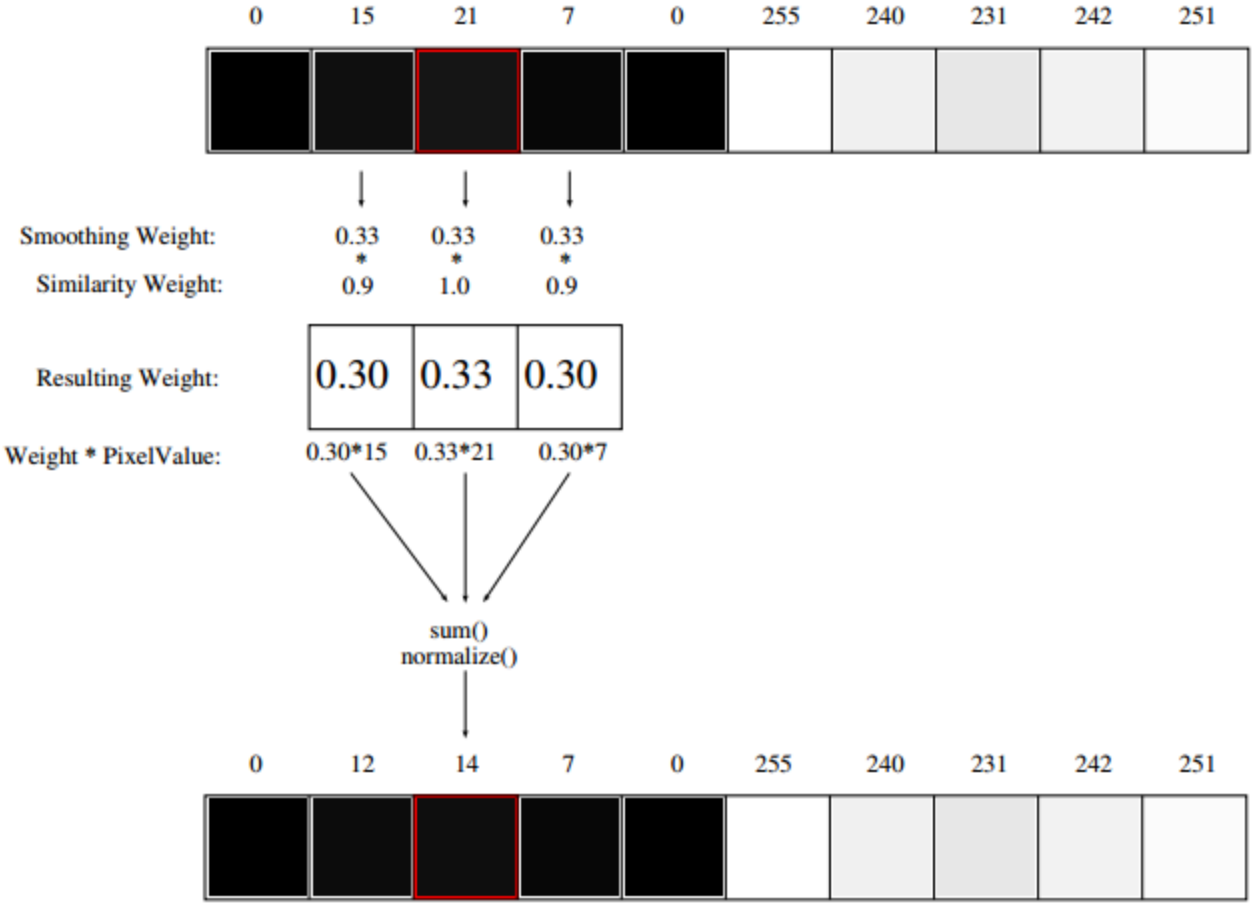


reproduced from [Durand 02]

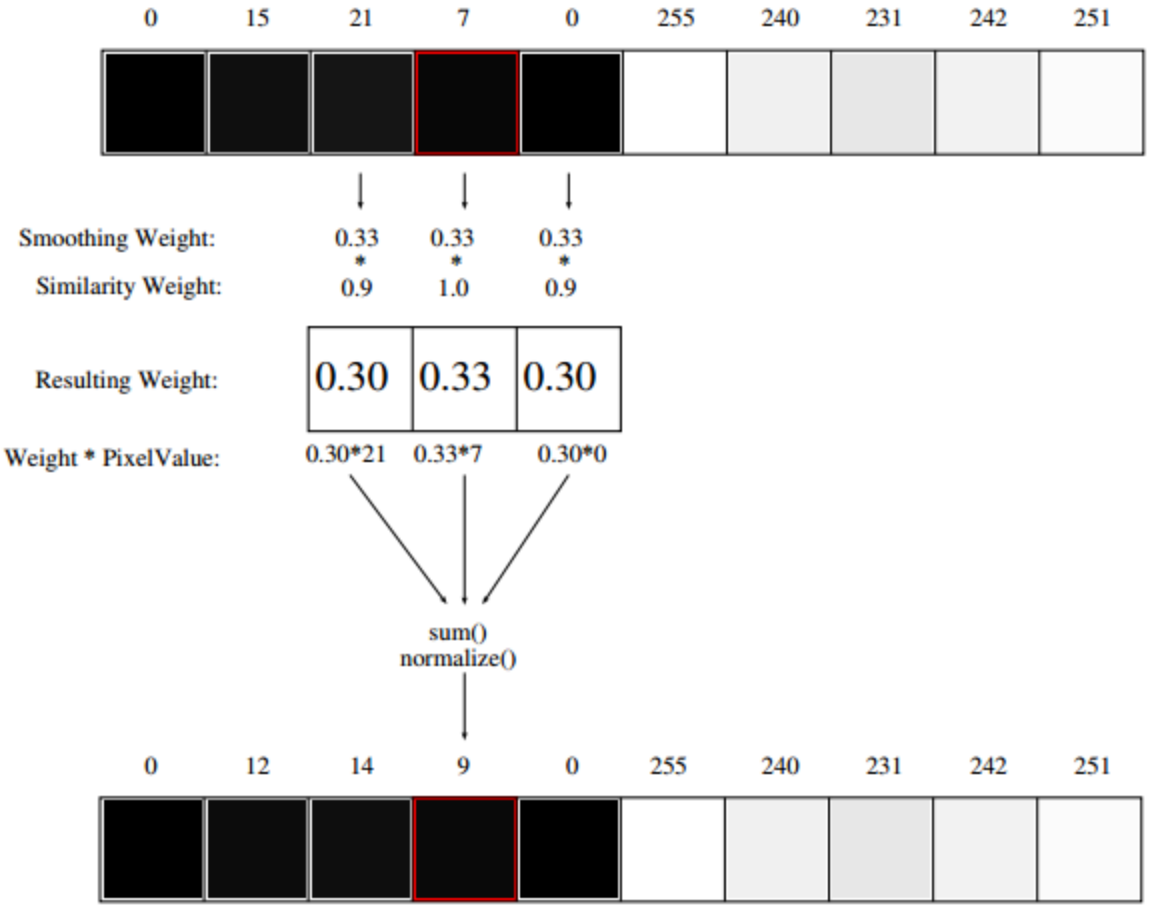
图像滤波: 双边滤波1D例子



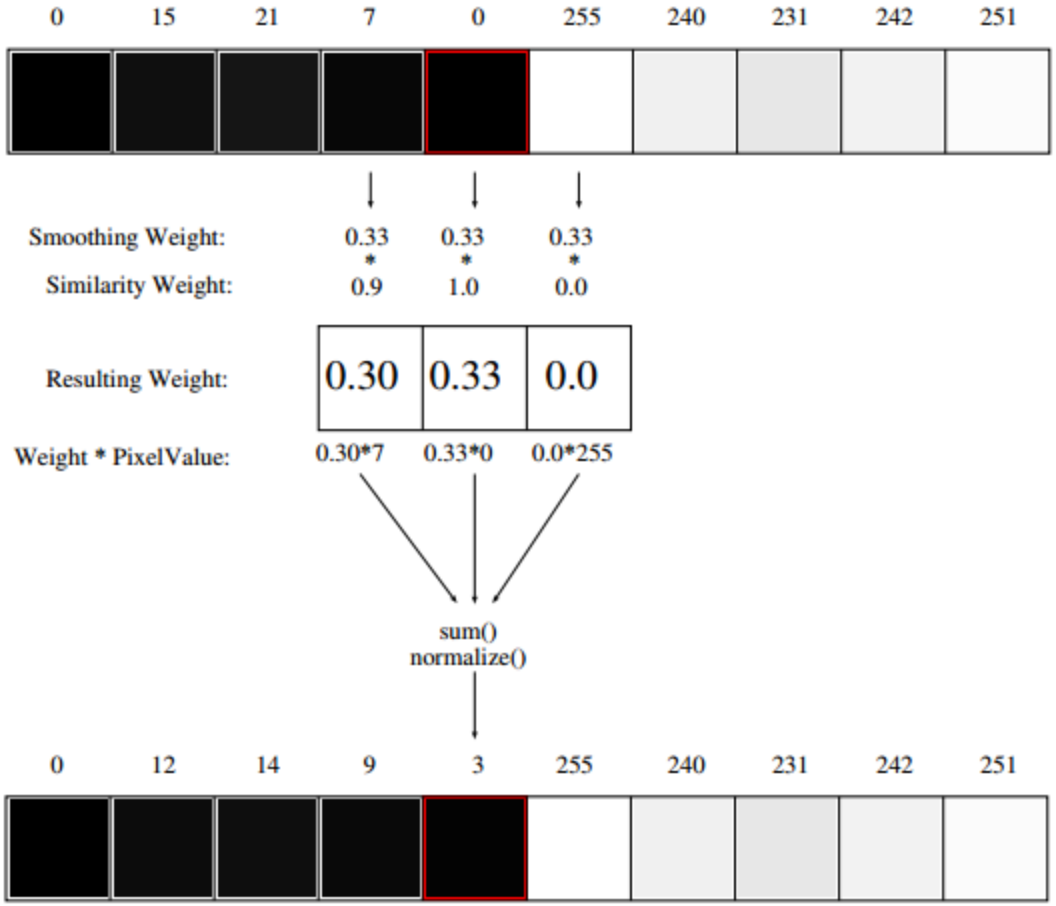
图像滤波: 双边滤波1D例子



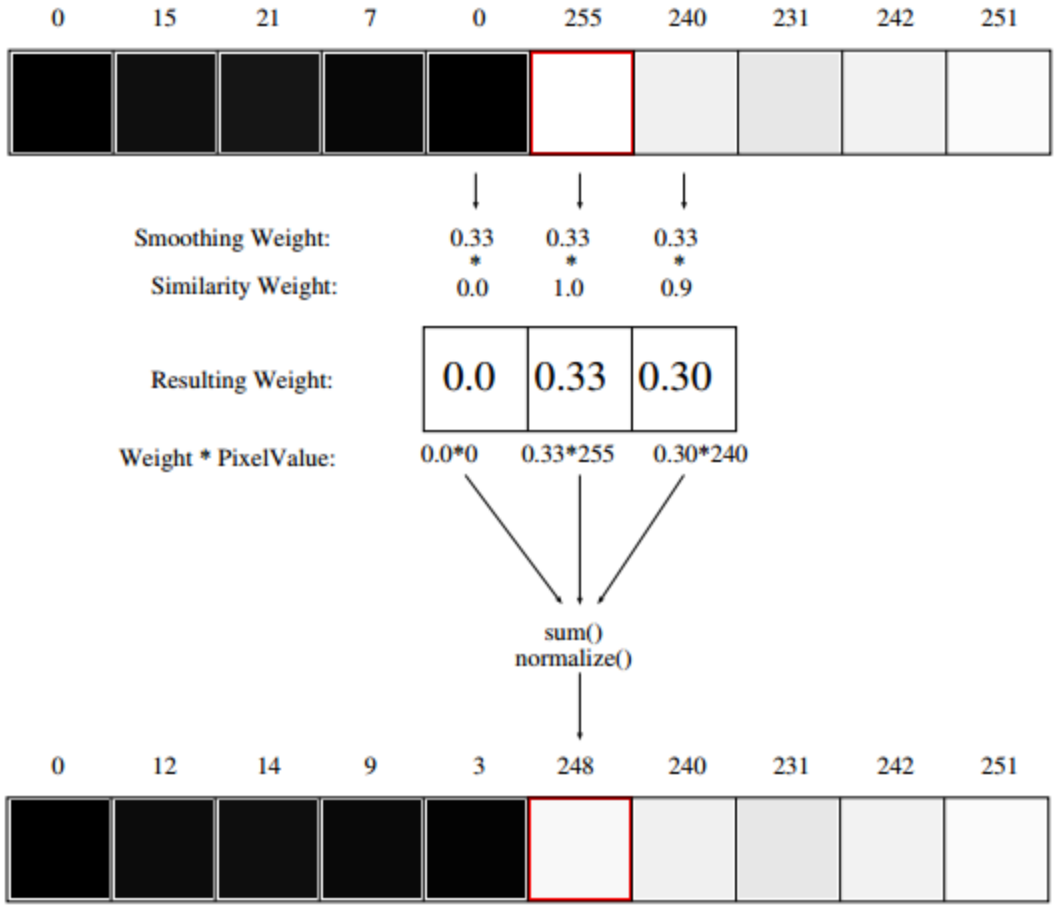
图像滤波: 双边滤波1D例子



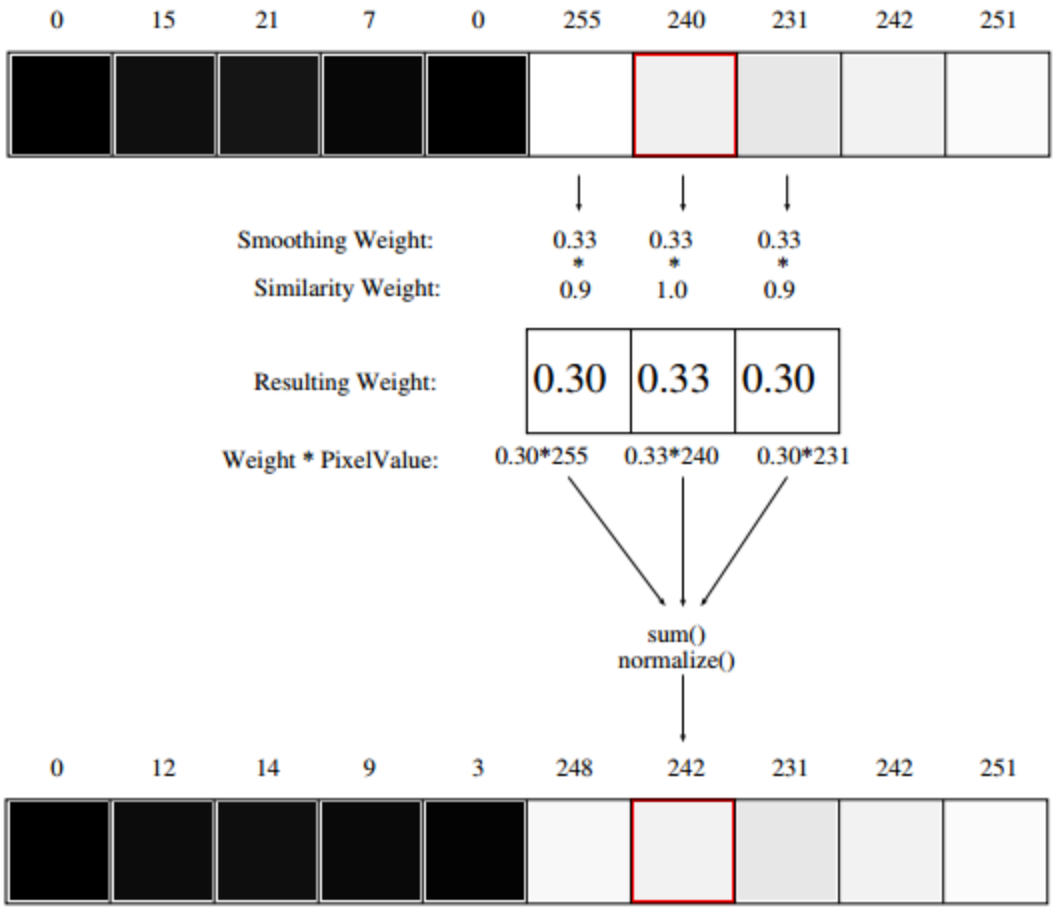
图像滤波: 双边滤波1D例子



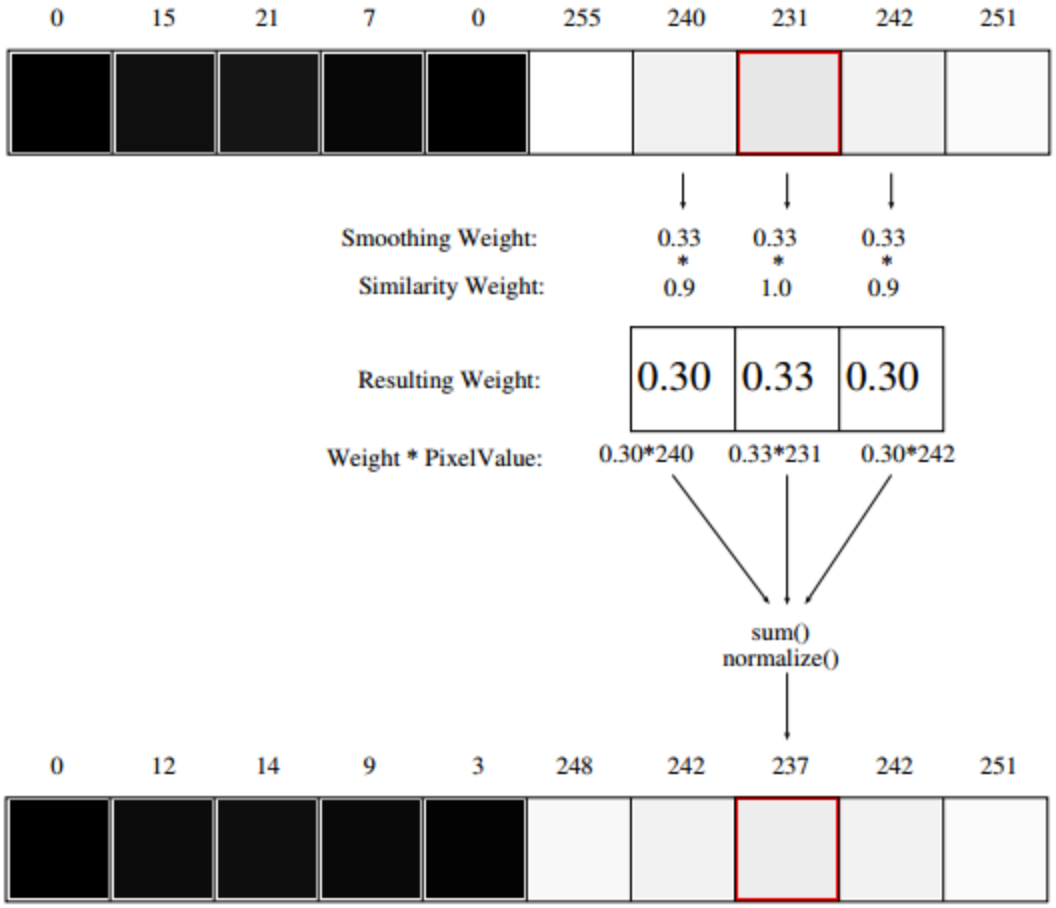
图像滤波: 双边滤波1D例子



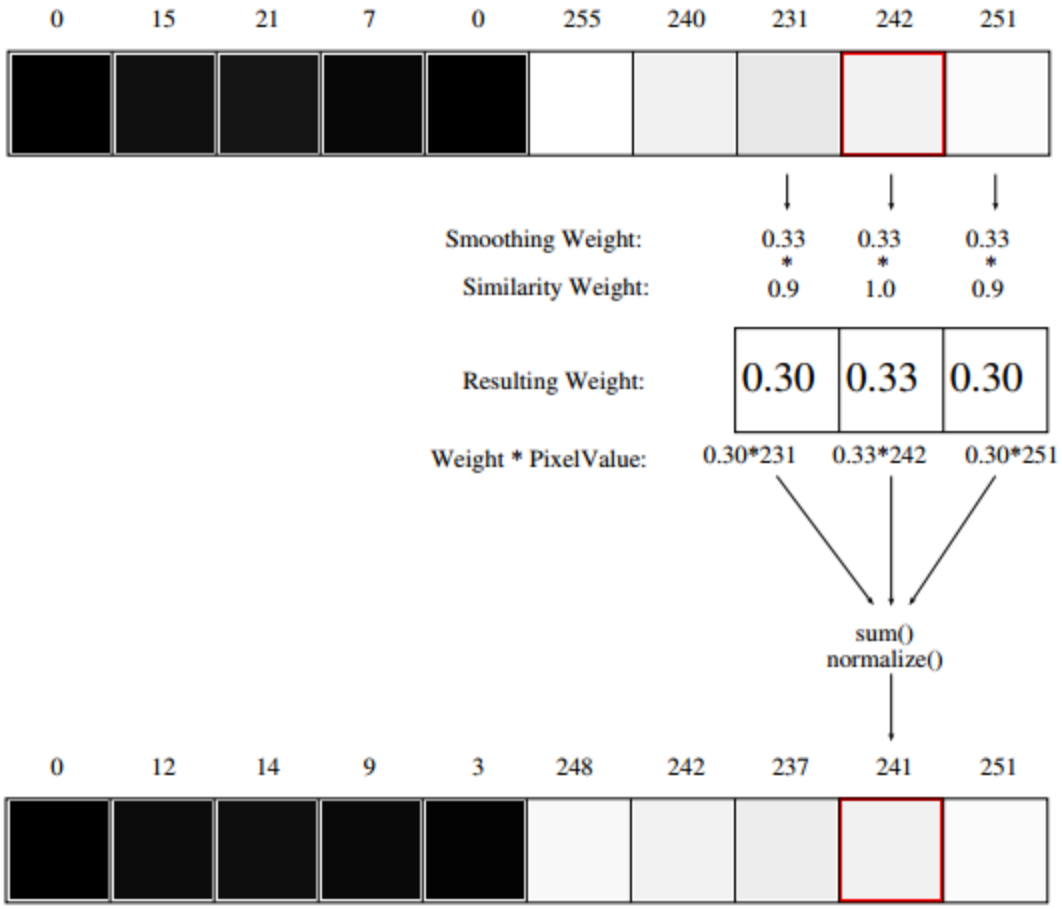
图像滤波: 双边滤波1D例子



图像滤波: 双边滤波1D例子



图像滤波: 双边滤波1D例子



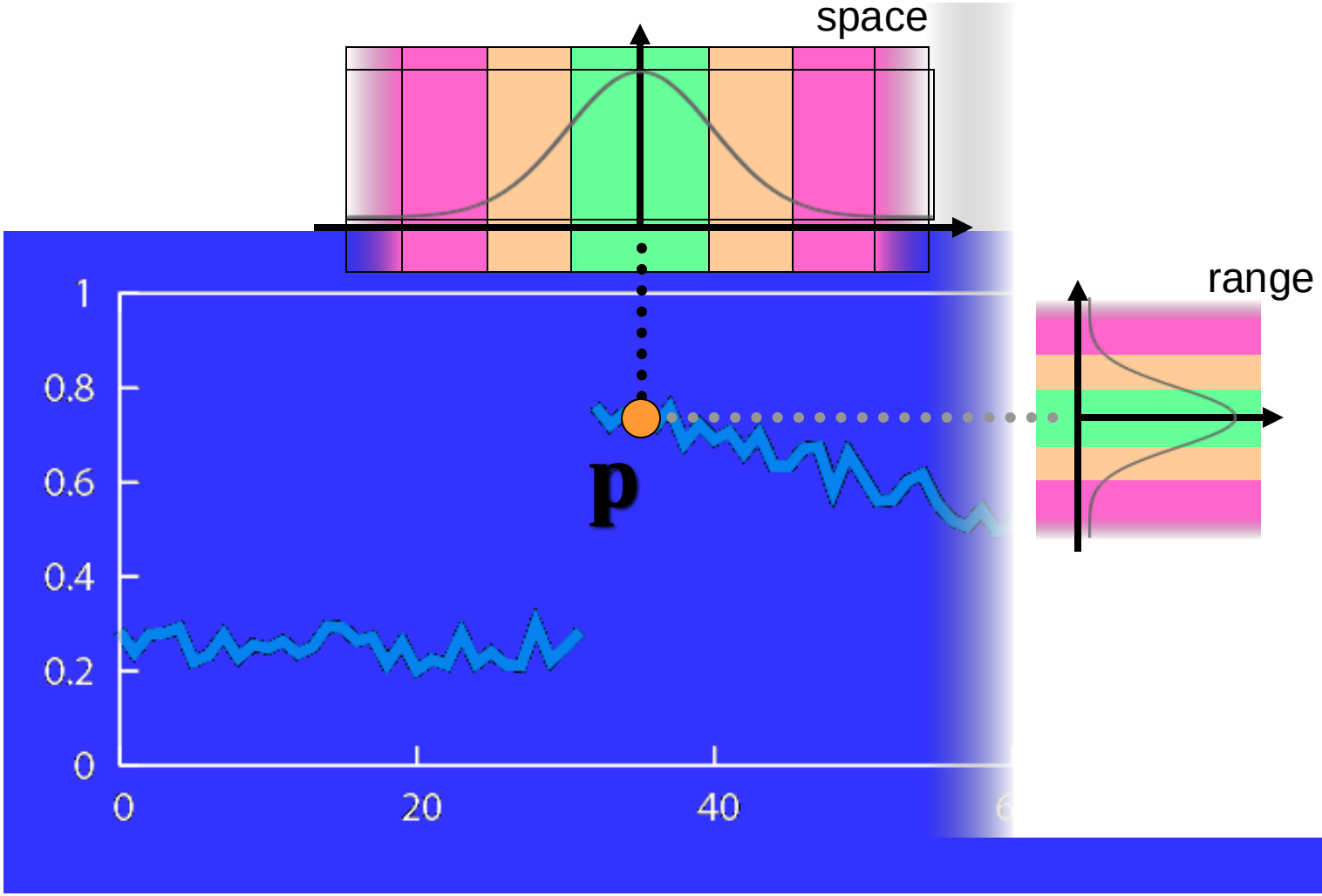
图像滤波: 双边滤波

$$BF[I]_{\mathbf{p}} = \frac{1}{W_{\mathbf{p}}} \sum_{\mathbf{q} \in S} G_{\sigma_s}(\|\mathbf{p}-\mathbf{q}\|) G_{\sigma_r}(|I_{\mathbf{p}} - I_{\mathbf{q}}|) I_{\mathbf{q}}$$

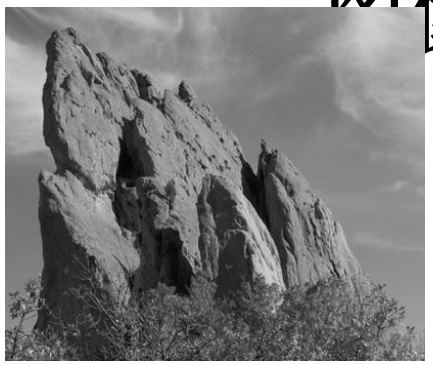
- space σ_s： 核的空间范围, 考虑的邻域的范围。
- range σ_r： 边界的最小梯度

图像滤波: 双边滤波

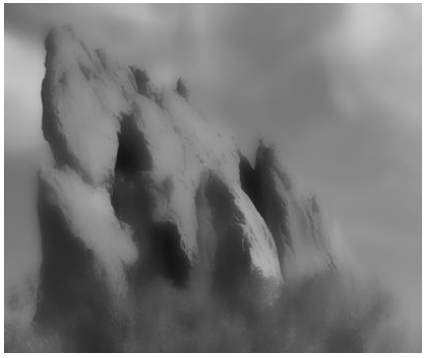
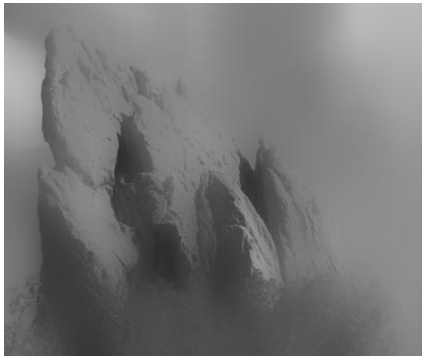
只考虑在space和range都相近的像素



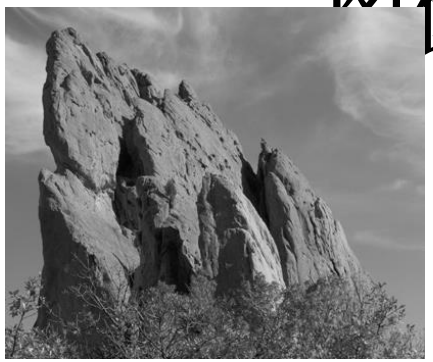
图像滤波: 双边滤波调整参数结果



输入

	$\sigma_r = 0.1$	$\sigma_r = 0.25$	$\sigma_r = \infty$ (Gaussian blur)
$\sigma_s = 2$			
$\sigma_s = 6$			
$\sigma_s = 18$			

图像滤波: 双边滤波调整参数结果



输入

	$\sigma_r = 0.1$	$\sigma_r = 0.25$	$\sigma_r = \infty$ (Gaussian blur)
$\sigma_s = 2$			
$\sigma_s = 6$			
$\sigma_s = 18$			

图像滤波: 双边滤波

设置双边滤波参数

根据应用。例如:

- 空间参数：和图像大小成比例
  - 例如：图像对角线的2%
- 范围参数: 和图像边界成比例
  - 例如： 图像梯度的中值或者均值

图像滤波: 双边滤波

双边滤波可以进行迭代

$$I_{(n+1)} = BF[I_{(n)}]$$

- 产生更加的分段平滑图像
- 不是经常能用到

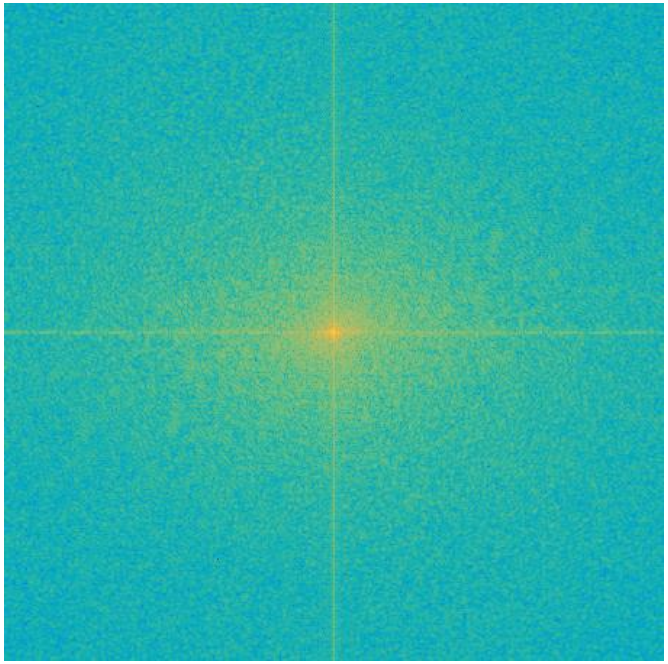

1 iteration


2 iterations


4 iterations

傅里叶变换

傅里叶变换



傅里叶变换

- 傅里叶变换将一个信号分解成一系列频率不同的正弦波函数的加权和

傅里叶变换

- 傅里叶变换将一个信号分解成一系列频率不同的正弦波函数的加权和

$$f_n = \frac{1}{M} \sum_{m=0}^{M-1} \hat{f}_m e^{j2\pi nm/M}$$

原始信号　　权重　　正弦波函数

这里的信号是离散信号列，长度是 M ，注意这个式子是反变换。

离散傅里叶变换

- 由原始信号计算各频率分量称为离散傅里叶变换(DFT)
- 反之由各频率分量合并为原始信号称为反离散傅里叶变换(IDFT)
- DFT: $\hat{f}_m = \sum_{n=0}^{M-1} f_n e^{-j2\pi nm/M}$
- IDFT: $f_n = \frac{1}{M} \sum_{m=0}^{M-1} \hat{f}_m e^{j2\pi nm/M}$

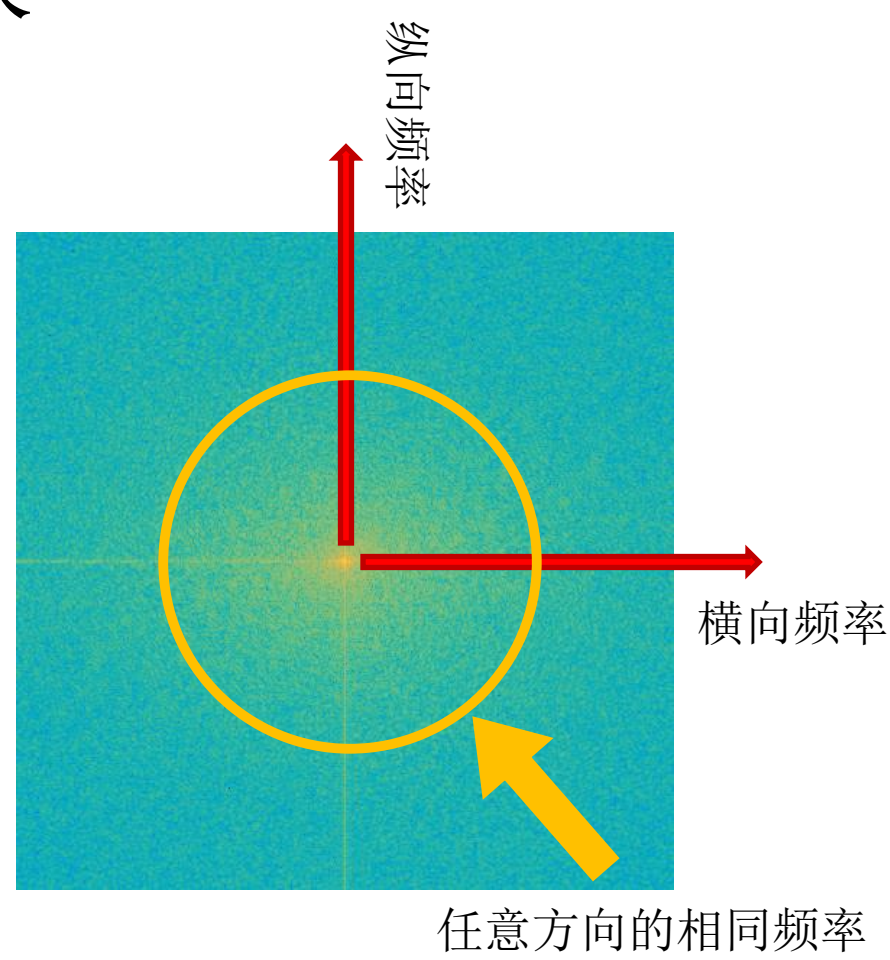
时域/频域变换

- 傅里叶变换又称为时域/频域变换
  - 原始信号对应时间域上的采样
  - 变换后函数对应于频率域上的分量

图像的傅里叶变换

- 二维 DFT

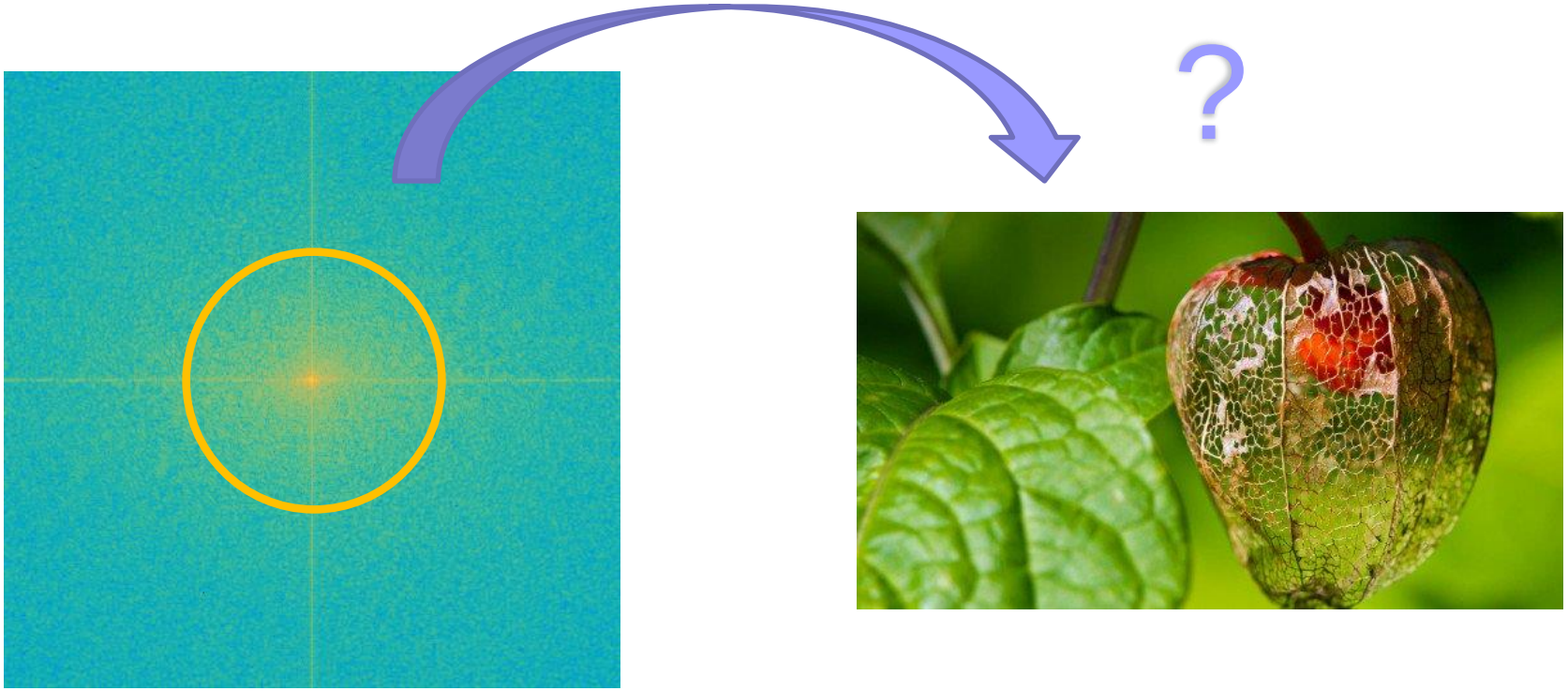
时域



频域

图像的频率分量

- 每个频率的分量是什么样子？



图像的频率分量

图像的频率分量

- 低频：区域等变化较小的内容
- 高频：边缘等变化较大的内容

低通/高通滤波

低频：区域

高频：边界

图像的频率分量

- 频率为 0 的分量是什么？

$$\hat{f}_0 = \sum_{n=0}^{M-1} f_n$$

  - 图像所有像素的值的和

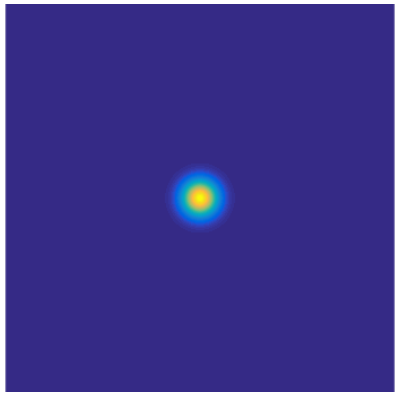
卷积特性

- 傅里叶变换有如下卷积特性

$$\mathrm{F}(f \star h) = \mathrm{F}(f) \times \mathrm{F}(h)$$

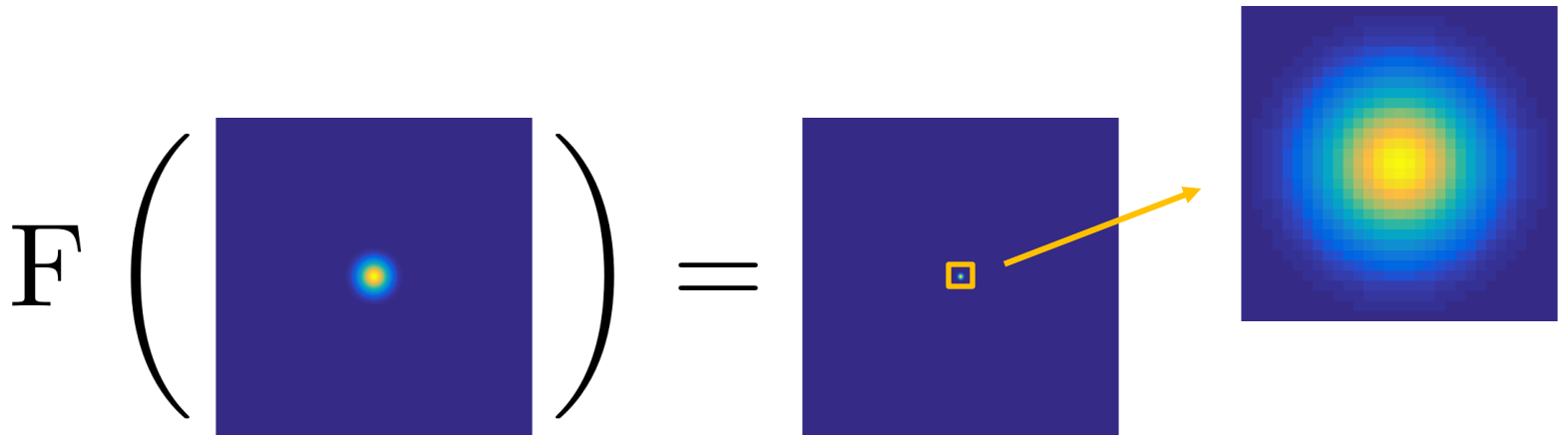
  - 时域卷积的傅里叶变换等于傅里叶变换在频域的乘积

高斯滤波

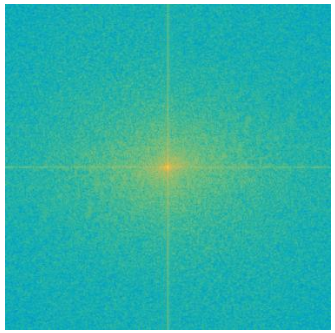


卷积特性

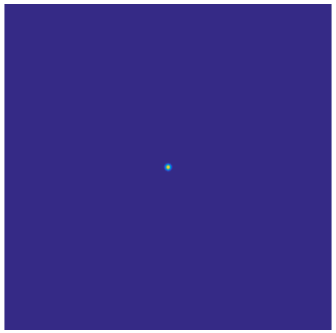
- 高斯核的傅里叶变换恰好是另一个高斯核
  - 时域的高斯滤波 = 频域用高斯核直接乘积



高斯滤波

$$\mathrm{F}\left(\right.$$  $$\left.\right)$$  

$$\times$$

$$\mathrm{F}^{-1}$$ 

卷积特性

- 可以看到高斯滤波在频域上是一个以高斯核加权的低通滤波
  - 代表边缘的高频部分被抑制，图像会变平滑
- 为什么不直接卷积？

离散傅里叶变换

- 工程上用来计算离散傅里叶变换的算法是快速傅里叶变换（FFT），在它发明以前，计算长信号的傅里叶变换是困难的。

- FFT复杂度: $O(M \log_2 M)$
- 按照定义计算DFT的复杂度呢？

离散傅里叶变换

- 用一个 $P \times Q$ 大小的图像对一个 $M \times N$ 的图像进行卷积的复杂度是 $O(MNPQ)$
- 利用卷积特性在频域进行滤波的复杂度是

$$O(PQ \log_2 PQ) + 2O(MN \log_2 MN) + O(\max\{MN, PQ\})$$

- 在卷积核很大的时候可以采用FFT加速。

谢谢！