

基本概念

• 智能机器

- 模拟人类的功能

让计算机具有视觉、听觉和说话等能力。这时计算机必须具有逻辑推理和决策的能力。具有上述能力的计算机就是智能计算机。

• 感知系统

- 人类感知外部世界主要是通过视觉、触觉、听觉和嗅觉等感觉器官
- 其中约80%的信息是由视觉获取的

• 计算机视觉(图像分析与理解)

- 研究用计算机来模拟生物外显或宏观视觉功能的科学和技术
- **计算机视觉首要目标：用图像创建或恢复现实世界模型，然后认知现实世界**

相关学科

计算机视觉可以而且应该根据计算机系统的特点来进行视觉信息的处理。

• 图像处理与图像分析

研究对象主要是**二维图像**，实现图像的转化，尤其针对像素级的操作，例如，提高图像对比度、边缘提取、去噪声和几何变换。

相关学科

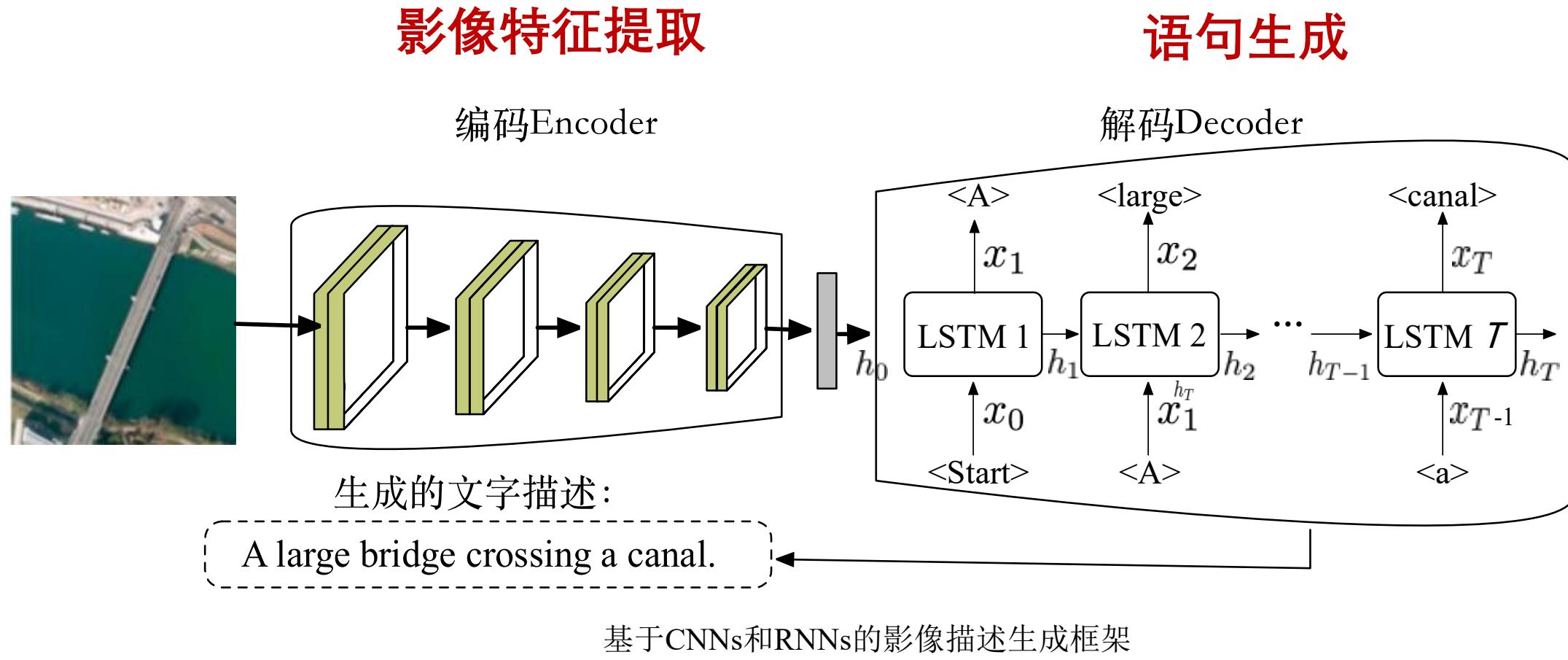
模式识别

- 根据从图像中抽取的**统计特性或结构信息**，把图像分成预定的类别。例如，文字识别或指纹识别。在计算机视觉中模式识别技术经常用于对图像中的某些部分，例如分割区域，进行识别和分类。

物理学、光学

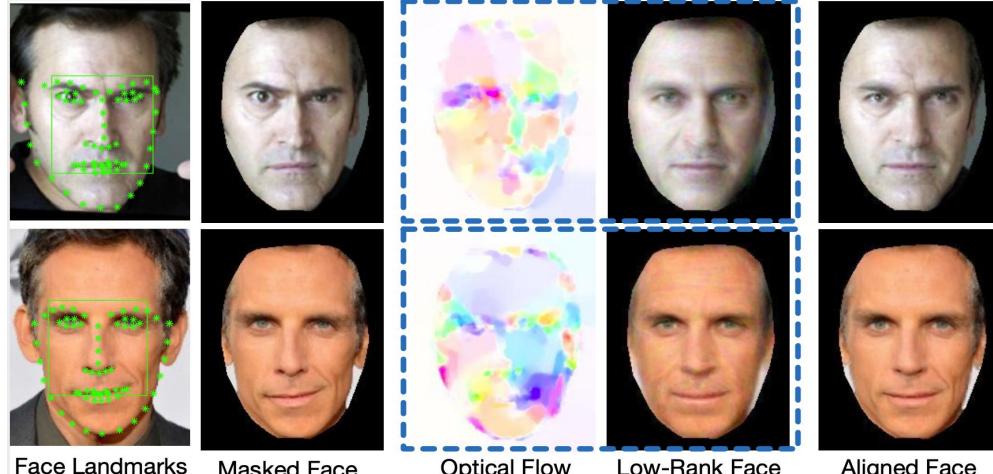
神经生物学（生物视觉）

计算机视觉的应用：影像描述生成 (Image Captioning)

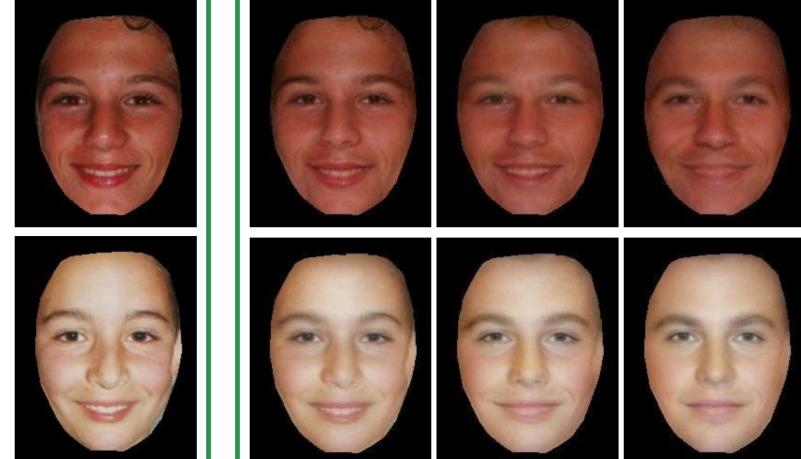


计算机视觉的应用：人脸检测

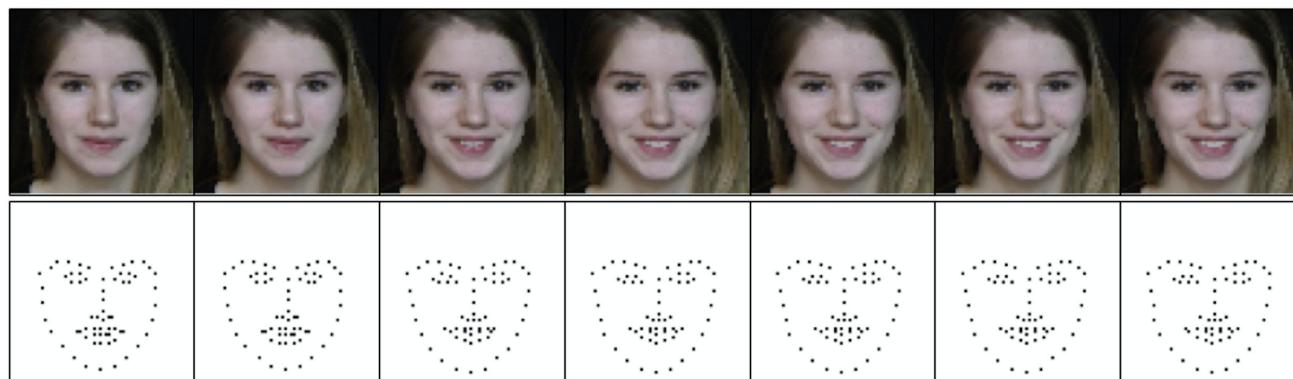
深度学习的应用->跨年龄人脸识别，计算机视觉的难点和热点



人脸角点检测



人脸老化 (孩童->青年)



人脸表情合成 (无表情->微笑)

计算机视觉的应用：物体识别（超市）



LaneHawk by EvolutionRobotics

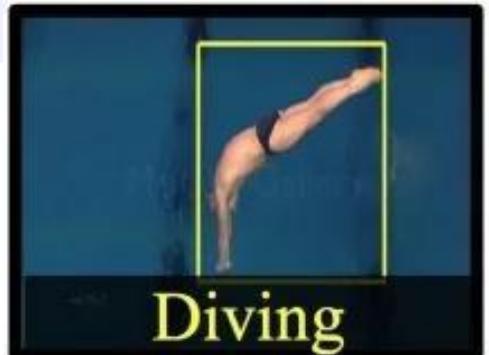
“A smart camera is flush-mounted in the checkout lane, continuously watching for items. When an item is detected and recognized, the cashier verifies the quantity of items that were found under the basket, and continues to close the transaction. The item can remain under the basket, and with LaneHawk, you are assured to get paid for it...”

- **主要任务：**判定一组图像数据中是否包含某个特定的物体、图像特征或运动状态。
- **到目前为止，还没有某个单一的方法能够广泛的对各种情况进行判定：**
- **在任意环境中识别任意物体。**现有技术仅能够很好地解决特定目标的识别：简单几何图形识别、人脸识别、印刷或手写文件识别、车辆识别.....
- **这些识别需要在特定的环境中，具有指定的光照，背景和目标姿态要求。**

计算机视觉的应用：图像恢复



计算机视觉的应用：体育赛事分析



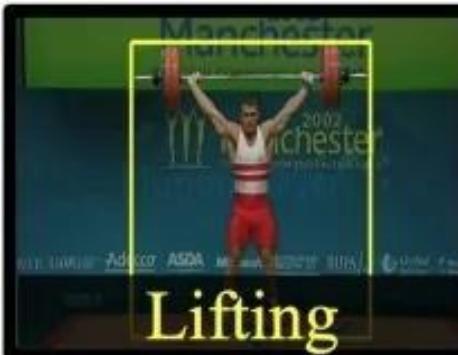
Diving



Golf Swing



Kicking



Lifting



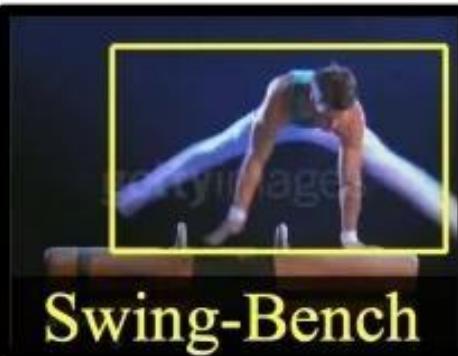
Riding Horse



Running



Skateboarding



Swing-Bench



Swing-Side



知乎 @vision
Walking

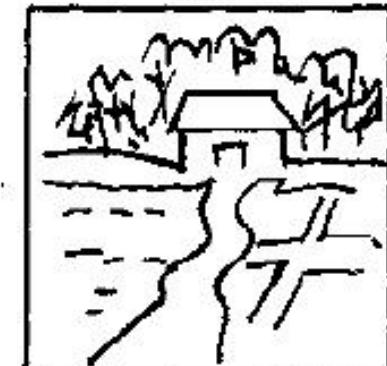
例如，视频分类，Central Florida大学的研究人员采集了一个包含10种运动（潜水、打高尔夫球、跑步等）150个视频的数据集UCF sports。

计算机视觉过程

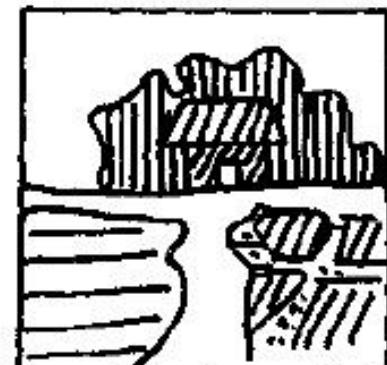
2025-11-27



(a)



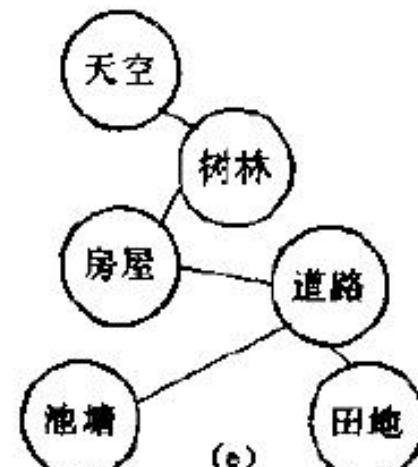
(b)



(c)



(d)



(e)



(f)

视觉的过程

分为三个阶段

□ 特征提取和区域分割

基于颜色, 轮廓, 纹理, 形状...

□ 建模与模式表达

基于各种物体的抽象化模型

□ 描述和理解

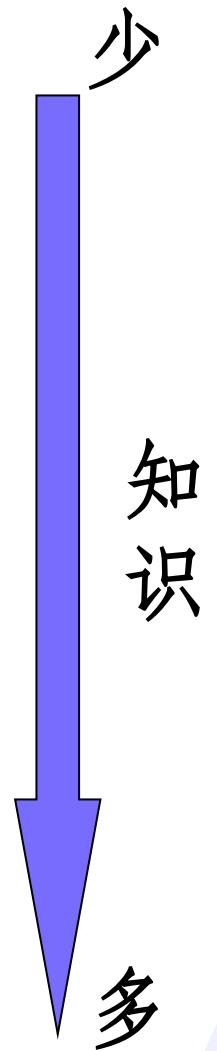
基于景物的结构知识



低层处理

中层处理

高层处理



传统计算机视觉

1. 基本概念和原理

- 通路、距离
- 仿射变换
- 直方图、直方图均衡化
- 颜色空间：RGB、HSI、HSV、伪彩色图像

2. 滤波

- 基本操作：协相关correlation、卷积 convolution
- 空间滤波
- 频域滤波

3. 特征提取

- 颜色
- 纹理
- 形状 (基元检测：角点、边缘)

HoG、LBP、Harr、SIFT、SURF

4. 立体视觉

5. 光流场

6. 工程实践：环境 + 任务

像素间通路

邻接

对两个像素 p 和 q 来说，如果 q 在 p 的邻域中，则称 p 和 q 满足邻接关系

连接

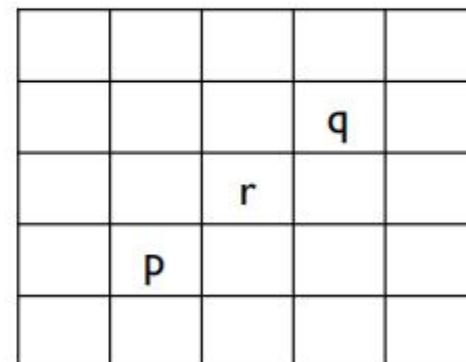
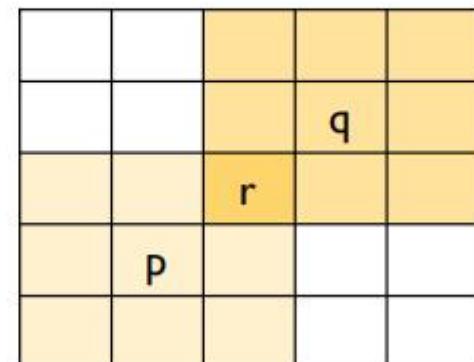
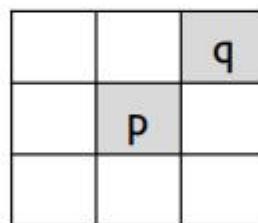
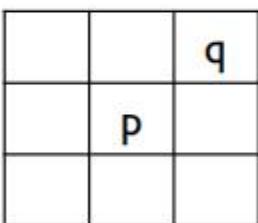
p 和 q 邻接且灰度值均满足某个特定的相似准则

连通

不（直接）邻接，但均在另一个像素的相同邻域中，且这3个像素的灰度值均满足某个特定的相似准则

通路

两个像素 p 和 q 间有一系列依次连接的像素使得 p 和 q 是连通的，则这些像素构成 p 和 q 间的通路



像素间距离

1. 曼哈顿距离(city-block-distance)

当 $p = 1$ 时, 即为曼哈顿距离或城市距离、街区距离, 是指两个向量之间的距离, 在计算距离时不涉及对角线移动。像素 $p(x,y)$ 和 $q(s,t)$ 之间的距离公式:

$$D_4(p, q) = |x - s| + |y - t|$$

表示从像素 p 向像素 q 出发, 每次能走的点必须是在当前像素点的 4 邻域中。一步一步走到 q 点后, 一共经过的像素点数就是曼哈顿距离。 **(D4距离)**

2. 欧式距离 (Euclidian distance)

当 $p = 2$ 时, 即为欧式距离, 就是直角坐标系的距离。像素 $p(x,y)$ 和 $q(s,t)$ 之间的距离公式:

$$D_e(p, q) = \sqrt{(x - s)^2 + (y - t)^2}$$

3. 切比雪夫距离 (chess-board-distance)

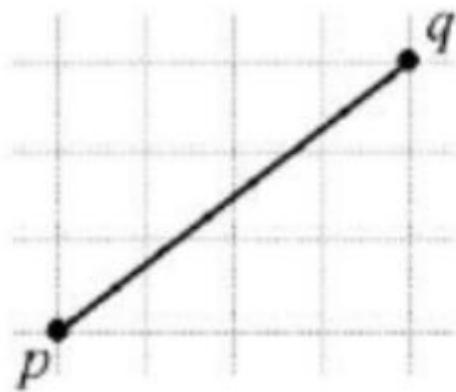
当 $p = \infty$ 时, 即为切比雪夫距离或棋盘距离, 像素 $p(x,y)$ 和 $q(s,t)$ 之间的距离公式:

$$D_8(x, y) = \max(|x - s|, |y - t|)$$

表示从像素 p 向像素 q 出发, 每次能走的点必须是在当前像素点的 8 邻域中。一步一步走到 q 点后, 一共经过的像素点数就是切比雪夫距离。 **(D8距离)**

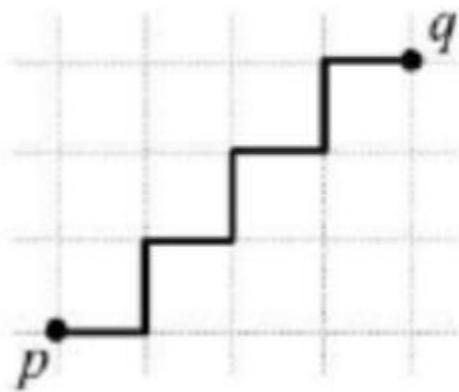
像素间距离

欧氏距离



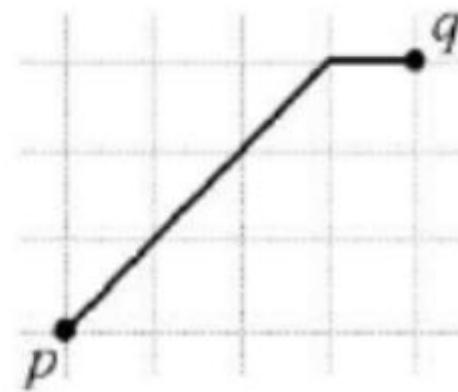
(a)

城区距离



(b)

棋盘距离



(c)

像素间距离

对于像素 $p(x,y)$ 、 $q(s,t)$ 和 $z(u,v)$ ，如果满足：

- 非负性： $D(p,q) \geq 0$
- 同一性： $D(p,q)=0$ ，当且仅当 $p=q$ 时
- 对称性： $D(p,q) = D(q,p)$
- 直递性： $D(p,z) \leq D(p,q) + D(q,z)$

则称 D 是距离的度量函数。

(Minkowski Distance)

在欧几里得空间中，点 (x_1, x_2, \dots, x_n) 和点 (y_1, y_2, \dots, y_n) 之间的闵可夫斯基距离：

$$D(x, y) = \left(\sum_{i=1}^n |x_i - y_i|^p \right)^{1/p}$$

图像直方图

图像直方图是图像像素强度分布的图形表示，对于灰度图像，直方图显示了每个灰度级（0 到 255）在图像中出现的频率，对于彩色图像，我们可以分别计算每个通道（如 R、G、B）的直方图。

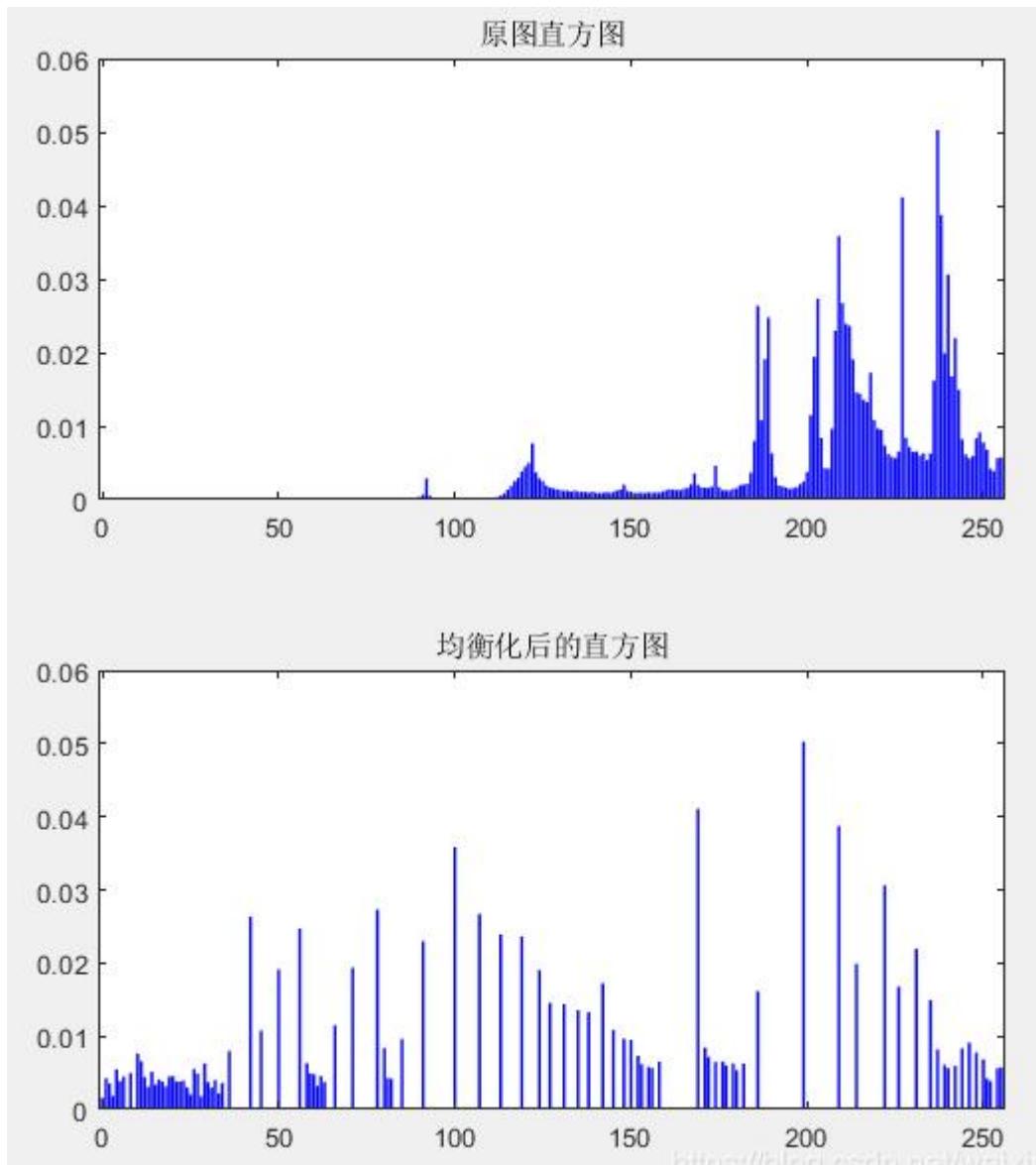
- 直方图：表示图像中像素强度的分布情况。
 - 横坐标 (X轴)：通常表示像素的强度值（例如，对于 8 位灰度图，就是 0 到 255）。
 - 纵坐标 (Y轴)：表示具有该强度值的像素数量。
 - **灰度直方图：针对灰度图像的直方图，表示每个灰度级的像素数量。**
 - 颜色直方图：针对彩色图像的直方图，分别表示每个颜色通道（如 BGR）的像素强度分布。
- 简单来说，直方图是一个统计图表，它显示了图像中每个像素强度值（比如亮度值0-255）出现的频率（即像素个数）。

直方图有什么用？

- **图像亮度与对比度：**
 - 如果直方图的峰值主要集中在左边（低亮度值），**说明图像偏暗。**
 - 如果峰值集中在右边（高亮度值），**说明图像偏亮。**
 - 如果峰值分布很窄，**说明图像对比度较低。**
 - 如果峰值分布很广，**说明图像对比度较高。**
- **图像内容分析：**通过直方图的形状，可以大致推断图像的内容。例如，一张大部分是天空的图片，其直方图的蓝色通道可能会在某个区域有高峰。
- **图像阈值化：**选择合适的阈值，用于将图像分割成前景和背景。
- **图像均衡化：**通过拉伸直方图，使其分布更均匀，可以增强图像的对比度。
- **图像匹配与检索：**比较不同图像的直方图，可以作为一种简单的图像相似性度量方法。

直方图均衡化

- 通常认为，图像统计直方图分布越均匀，图像的质量越好，直方图均衡化可以提升图像的对比度和质量。例如，若图像整体较暗，说明直方图中低像素值分布较多，高像素值分布较少，这时可以使用直方图均衡化来平衡其直方图分布，即减少低像素值，增加高像素值。从直观上看，均衡化的效果就是图像的部分区域相对原来变亮了，所以提升了图像的对比度。



- **直方图均衡化**：是一个所有像素值重映射的过程。例如，像素值 x 的直方图均衡化之后的值可按下式计算，

$$H(x) = (2^n - 1) * P(x)$$

其中， n 为图像的每一个像素点的位宽，对于灰度图通常 $n=8$ ，
 $P(x)$ 为像素值 x 的累加概率。

- $P(x)$ 的取值范围是0~1, $H(x)$ 与 $P(x)$ 具有线性关系。
- 为了进一步提升图像对比度, 可以对 $P(x)$ 作一个非线性的S型变换 $T(P(x))$, 并保证变换之后 $T(P(x))$ 的取值范围还在0~1之间。
- 例如, 可构造 T_1 和 T_2 这两个非线性函数用于非线性变换, 同时可以把原本 $H(x)$ 与 $P(x)$ 的线性关系看成函数 T_0 , 于是有以下三种变换:

$$T_0(x) = x$$

$$T_1(x) = 0.5 * (\sin(\pi x - \pi / 2) + 1)$$

$$T_2(x) = \frac{1}{1 + e^{3-x*6}}$$



T0



T1



T2

卷积与滤波

图像处理的方法基本上可分为空间域法和频域法两大类。

(1) 空域法：

在原图像上直接进行数据运算，对像素的灰度值进行处理。

- **点运算**：对图像作逐点运算；
- **局部运算**：在与处理像点邻域有关的空间域上进行运算。

(2) 频域法：

在图像的变换域上进行处理，增强感兴趣的频率分量，然后进行反变换，得到频域处理过的图像。

空域滤波及滤波器的定义：

- 使用**空域模板**进行的图像处理，被称为**空域滤波（局部运算）**。模板本身被称为**空域滤波器**。
- 空域滤波：卷积与模板
 - **模板操作**是数字图像处理中常用的一种运算方式，图像的平滑、锐化、细化、边缘检测等都要用到模板操作。
 - 例如，一种常见的平滑算法是将原图中的一个像素的灰度值和它周围邻近8个像素的灰度值相加，然后将求得的平均值作为新图像中该像素的灰度值。可用如下方法来表示该操作：

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

- 上式类似于矩阵，通常称之为**模板 (Template)**，带星号的数据表示该元素为**中心元素**，即这个元素是将要处理的元素。

- 如果模板为： 
- 则该操作的含义是：将原图中一个像素的灰度值和它右下相近的8个像素值相加，然后将求得的平均值作为新图像中该像素的灰度值。

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

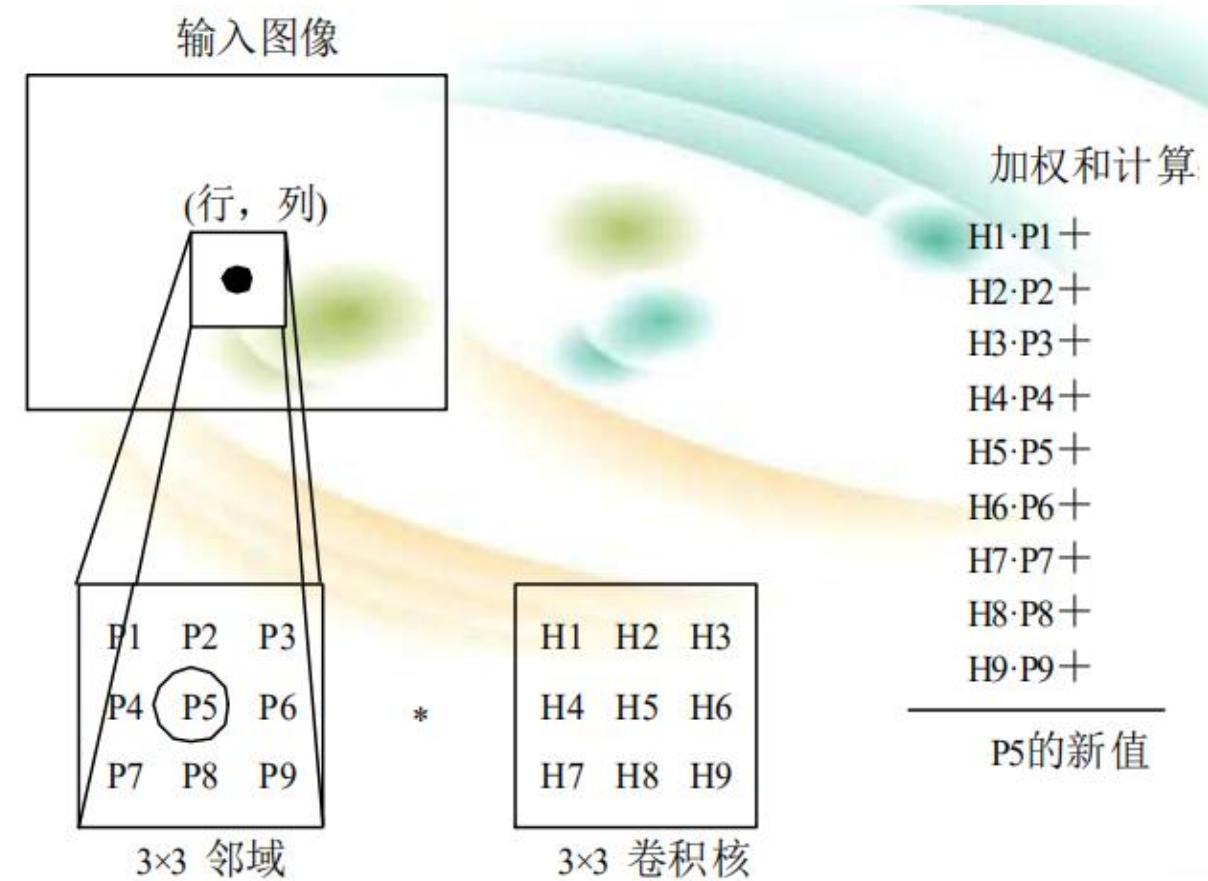
- 模板操作**实现了一种**邻域运算**，即某个像素点的结果不仅和本像素灰度有关，而且和其邻域点的值有关。模板运算的数学含义是**卷积**（或互相关）运算。

卷积与滤波

卷积是一种用途很广的算法，可用卷积来完成各种空域的图像处理：

卷积运算中的**卷积核**就是模板运算中的**模板**：

- **卷积**就是作**加权求和**的过程。邻域中的每个像素（假定邻域为 3×3 大小，卷积核大小与邻域相同），分别与卷积核中的每一个元素相乘，**乘积求和所得结果即为中心像素的新值**。
- **卷积核中的元素称作加权系数**（卷积系数），卷积核中的系数大小及排列顺序，决定了对图像进行处理的类型。改变卷积核中的加权系数，会影响到总和的数值与符号，从而影响到所求像素的新值。
- **卷积核 = 模板 = 空域滤波器**



滤波的主要目的两个：

1. 通过滤波来提取图像特征，简化图像所带的信息作为后续其它的图像处理
2. 为适应图像处理的需求，通过滤波消除图像数字化时所混入的噪声

$$R_5 = \frac{1}{n} \sum_{i=0}^n R_i G_i$$

其中， R_5 表示当前像素点， $R_i G_i$ 表示当前像素与滤波器对应值相乘的值， n 为滤波器大小，举例来说如若此滤波器值全为1，则此公式计算的是当前像素点的8连通像素点的平均值，因此滤波完后的图像应表现为模糊的效果，模糊程度取决于滤波器的大小，滤波器大小(size)越大，模糊效果越明显。

卷积边界处理

1. 零填充(Zero Padding)

- 在原始图像周围填充零值像素，使其在进行卷积操作时能够保持原始图像大小不变，保留边界信息，使得卷积核能够正确地处理边缘像素。

2. 同样填充(Same Padding):

- 在输入特征图的边缘添加适当数量的像素，使得卷积操作后的输出与原始输入图像具有相同的大
小，可以保留边缘信息，同时避免了输出尺寸的缩小。

3. 边缘复制填充(Edge Padding)

- 将图像边界的像素值复制到填充区域，使得边缘像素的信息得到充分利用，能够保留图像边缘的
特征。

4. 反射填充(Reflection Padding):

- 基于图像边界的像素值，将像素值进行镜像或反射填充到填充区域，可以减少边界效应，并提供
更平滑的填充。

5. 不做处理(No Padding):

- 在进行卷积操作时，直接忽略边界像素。可能导致边界信息丢失或在特征提取中产生不良影响。

选择合适的边界处理方式取决于任务的性质和数据集的特点。