总结

小米
京东
Talkweb
小天才
安克创新
多益网络 测评

talk web

卷积移动的公式

$$w' = \frac{(w + 2p - k)}{s} + 1$$

1. 池化是为了什么

（1）首要作用，下采样（downsamping）

（2）降维、去除冗余信息、对特征进行压缩、简化网络复杂度、减小计算量、减小内存消耗等等。各种说辞吧，总的理解就是减少参数量。

（3）实现非线性（这个可以想一下，relu函数，是不是有点类似的感觉？）。

（4）可以扩大感知野。

（5）可以实现不变性，其中不变形性包括，平移不变性、旋转不变性和尺度不变性。

2. 连续子数组和

- 前n项和，遍历前n项和求解

3. 数组翻转90 180 270 度

```python
def rotate_90(matrix):
    return [list(row) for row in zip(*matrix[::-1])]

def rotate_180(matrix):
   return [row[::-1] for row in matrix[::-1]]

def rotate_270(matrix):
    return [list(row) for row in zip(*matrix)][::-1]
```

小米

1. 翻转两个相邻的数
2. 判断字符串长度

```python
a = 'a\\\abc'
b = r'a\\\abc'
print(len(a),a)
print(len(b),b)
```

字符串 `a = 'a\\\abc'`：

这里的 `\\\` 会被解释为两个部分：前两个反斜杠 `\\` 作为转义字符表示一个反斜杠，第三个反斜杠 `\` 用于转义字符 a。由于 `\a` 是转义序列，在 Python 中表示 ASCII 控制字符“响铃”（Bell），而不是普通字符。这样，字符串实际被解释为 `a\bc`，因为控制字符不会输出内容。
因此，`a` 的实际长度为 `5`，但因为控制字符没有打印，输出为 `a\bc`。

字符串 `b = r'a\\\abc'`：
原始字符串不会解析反斜杠，因此字符串 `a\\\abc` 会被按原样解释。
这里不会有转义的效果，所有反斜杠都被保留，字符串的实际长度是 `7`。
输出为 `a\\\abc`。

小天才

各种算子

完全二叉树叶子节点的计算

完全二叉树的性质

1. **完全二叉树**：完全二叉树是一种特殊的二叉树，除了最后一层外，所有的层都被完全填满，并且最后一层的节点从左到右连续排列。
2. **节点数**：对于高度为 `h` 的完全二叉树，节点总数可以用公式计算：

$$n = 2^h - 1$$

其中，`h` 是树的高度（从0开始计数），`n` 是节点总数。

叶子节点数量计算

前序遍历和后续遍历的数组排列一样

所有节点都没有子节点或只有一个子节点的树。这种树的结构非常特殊，被称为**退化树**（Degenerate Tree）或**链式树**（Skewed Tree）。

霍夫变换编码压缩公式

Linux DTS文件配置

Linux内核锁

频域空间域滤波有哪些

常见的空间域滤波器：

- **均值滤波**：用邻域内像素的平均值来替换中心像素，主要用于图像去噪，但会模糊图像细节。
- **中值滤波**：用邻域内像素的中值来替换中心像素，能够很好地去除椒盐噪声。
- **高斯滤波**：用邻域内像素按高斯分布加权平均来替换中心像素，常用于平滑处理以去除噪声，同时保留较多细节。
- **拉普拉斯滤波**：是一种二阶导数滤波器，用于检测图像中的边缘，强调图像中的变化区域。
- **Sobel滤波**：通过计算图像梯度用于边缘检测，分为水平方向和垂直方向滤波器。

优点：

- 直观、易于实现。
- 适合在空间分辨率要求较高的场景中使用。

缺点：

- **在去噪过程中容易造成图像细节的丢失和模糊化。**
- 有时对周期性噪声（如条纹噪声）处理效果较差。

频域和空间域滤波是图像处理中的两大基本技术，分别在不同的域（频域或空间域）对图像进行处理。

频域滤波

频域滤波是通过对图像进行傅里叶变换，将其从空间域转换到频域，然后在频域内对频率分量进行操作。常见的频域滤波器会在高频或低频部分进行滤波。

常见的频域滤波器：

- 低通滤波器
  （Low Pass Filter, LPF）：去除图像的高频分量，保留低频分量，实现图像的平滑和去噪。
  - 例如：理想低通滤波器、巴特沃斯低通滤波器、Gaussian低通滤波器。
- 高通滤波器
  （High Pass Filter, HPF）：去除图像的低频分量，保留高频分量，通常用于增强图像的边缘和细节。
  - 例如：理想高通滤波器、巴特沃斯高通滤波器。
- **带通滤波器**（Band Pass Filter）：允许一定范围内的频率通过，同时滤除高频和低频分量，适用于特定频率段的提取。
- **带阻滤波器**（Band Stop Filter）：滤除一定范围内的频率，保留其余频率，常用于去除周期性噪声。

优点：

- 频域滤波可以精确地对频率分量进行处理，适合去除周期性噪声和增强图像的特定特征。
- 能够在高频和低频分量间灵活选择，以更好地保留图像的细节或去噪。

缺点：

- 需要对图像进行傅里叶变换和逆傅里叶变换，计算复杂度相对较高。
- 不如空间域滤波直观，实施较为复杂。

空间域与频域滤波的对比

- **计算复杂度**：频域滤波需要进行傅里叶变换，计算量大；而空间域滤波则可以通过简单的卷积实现，计算量较小。
- **效果**：频域滤波对周期性噪声处理效果更好，而空间域滤波对随机噪声处理效果较好。
- **应用场景**：频域滤波适用于特征提取或增强特定频率段，空间域滤波适用于去噪

栈和队列的区别和使用方式

- **操作顺序**：
  - 栈：后进先出（LIFO）。

    - 队列：先进先出（FIFO）。
- **使用场景**：
    - 栈：用于递归问题、括号匹配、逆序输出等场景。
    - 队列：用于广度优先搜索、任务调度、数据流缓冲等场景。
- **实现方式**：
    - 栈：通常使用数组或链表，通过在末端插入和删除元素来实现。
    - 队列：通常使用链表或双端队列（`deque`），通过在头部删除、在尾部插入元素来实现。

内存分配方式有哪些

- 动态内存分配

动态内存分配在程序运行时，根据需求动态分配和释放内存。它允许程序在运行时根据实际需求调整内存的使用量。

- 静态内存分配

静态内存分配在编译时完成，即程序在编译时就确定了所有变量的内存需求。分配的内存大小在程序整个生命周期内保持不变，直到程序终止时由系统自动回收。

- 内存池内存分配

内存池是一种特殊的内存分配方式，它预先分配一块大内存区域，然后在该区域内进行内存的划分和管理。内存池分配通常用于提高内存分配效率和减少内存碎片。

- 栈内存分配

栈内存分配主要用于存储函数的局部变量和函数调用的上下文信息。当函数被调用时，局部变量的内存从栈中分配，函数返回时这些内存会自动释放。

边缘提取方法有哪些和区别

各方法的区别总结

方法	特点	抗噪能力	计算复杂度	边缘检测精度
Sobel算子	简单的一阶微分，适合简单边缘检测	较差	低	中
Prewitt算子	类似Sobel，计算稍简单	较差	低	中
Roberts算子	二阶差分，计算快，适合实时系统	差	非常低	较低
Laplacian算子	二阶微分，适合细节检测	差	低	高
Canny算法	首先对图像进行高斯平滑，再计算梯度幅值和方向，通过非极大值抑制去除冗余边缘，最后通过双阈值处理得到较精确的边缘。边缘精度高，抗噪能力强	强	高	非常高
傅里叶变换法	频域边缘检测，适合特定频率分析	中等	高	中
LoG算子	平滑+Laplacian，抗噪能力较好	强	中	较高

低通滤波 高通滤波 维也纳滤波

低通滤波

低通滤波是一种允许**低频**信号通过，阻止或衰减**高频**信号的滤波方法。在图像处理中，低频分量通常对应图像的**平滑区域**，而高频分量对应图像的**细节和噪声**。因此，低通滤波器常用于图像的去噪和平滑。

高通滤波（High-Pass Filtering）

主要功能：

- **允许高频信号通过**，削弱或去除低频信号。
- 在图像处理中，高频成分包含边缘和细节信息，因此高通滤波器常用于边缘检测和图像增强。

维纳滤波（Wiener Filtering）

主要功能：

- **自适应滤波**：通过**最小化均方误差（MSE）来估计噪声**，适用于在噪声环境中恢复信号或图像。
- 维纳滤波考虑了信号和噪声的统计特性，能够在去噪的同时保留细节。

应用：

- **图像去噪和复原**：维纳滤波能够有效去除噪声，同时尽可能保留图像的细节。
- **信号恢复**：当信号被噪声干扰时，维纳滤波可以有效恢复原始信号

除大颗粒噪声保存更多细节的方法

步骤1：**中值滤波**（去除大颗粒噪声）

中值滤波非常适合处理大颗粒或椒盐噪声，能有效去除孤立的噪声点，同时对图像细节的破坏较小。

步骤2：**双边滤波**（平滑图像，保留边缘）

双边滤波可以平滑噪声较多的区域，同时保留图像的边缘信息。这个步骤用于进一步消除噪声，同时保持细节。

步骤3：**总变差去噪**（减少细微噪声，保留细节）

总变差去噪用于处理图像中剩余的细小噪声，并通过能量最小化来保留图像的边缘和重要细节。这个步骤可以进一步改善图像的整体质量。

方案优点：

- **中值滤波**用于初步去除大颗粒噪声。
- **双边滤波**保留了图像边缘，确保在去噪过程中不会破坏关键的结构信息。
- **总变差去噪**用于细节恢复，减少细微噪声，同时保留更多图像细节。

数组中的重复元素　只保留三个

京东

随机森林

希尔排序

**dic

```python
def a(a,b):
    return a + b
b = {"a":1,"b":2}
print(a(**b)) #3
```

a = 1,2 a*=2 a = (1,2,1,2)

B树

LSTM三门

三门一状态

- 输入门

决定哪些新信息将被加入到单元状态中。它包括两个部分：一个sigmoid函数决定要更新哪些信息，另一个tanh函数生成一个新的候选值向量，用于更新状态

- 输出门

决定从单元状态中输出哪些信息。它也通过sigmoid激活函数生成一个值，然后将单元状态通过tanh函数处理，最后与输出门的输出相乘

- 遗忘门

决定从单元状态中丢弃哪些信息。它通过一个sigmoid激活函数，输出一个介于0和1之间的值，表示每个信息的保留程度，0表示完全丢弃，1表示完全保留。

- 单元状态

虽然不是一个门，但它是LSTM的重要组成部分，负责携带信息在网络中流动。它通过遗忘门和输入门进行更新。

logistic是什么算法

逻辑回归（Logistic Regression）是一种用于二分类问题的统计学习方法，尽管名称中包含“回归”，但它实际上是一种分类算法。逻辑回归的基本思想是使用逻辑函数（Sigmoid函数）将线性组合的输入特征映射到0和1之间的概率值，进而用于进行分类。

牛牛的和谐神器

区间第k小

小天才2

模型Dropout后推理时间变吗

- 模型dropout后，在推理时时间并没有明显的变化

下面结构体

```
分析下面结构体占用了多少字节{
    long a,
    int b,
    static int c,
    char d,
    char e
}

16 # static 不占用。char 要对齐。8 + 4 + 1 + 1 + 2（对齐）
```