

# Opencv图像处理

导师: Zhang

---





deepshare.net

深度之眼

# 第四课：图像特征与目标检测

---

# 知识树

## 图像特征与目标检测

图像特征理解

形状特征

Harris

HOG原理

SIFT

纹理特征

LBP

模板匹配

人脸检测

行人检测



# 目录

1/ 图像特征理解

2/ 形状特征



# 1. 图像特征理解

---



# 图像特征

图像特征是图像中独特的，易于跟踪和比较的特定模板或特定结构。



特征就是有意义的图像区域，该区域具有独特性或易于识别性！

# 图像特征

---

- 图像特征提取与匹配是计算机视觉中的一个关键问题，在目标检测、物体识别、三维重建、图像配准、图像理解等具体应用中发挥着重要作用。
- 图像特征主要有图像的颜色特征、纹理特征、形状特征和空间关系特征。

# 颜色特征

---

- 颜色特征是一种全局特征，描述了图像或图像区域所对应的景物的表面性质
- 颜色特征描述方法：
  - 颜色直方图
  - 颜色空间
  - 颜色分布

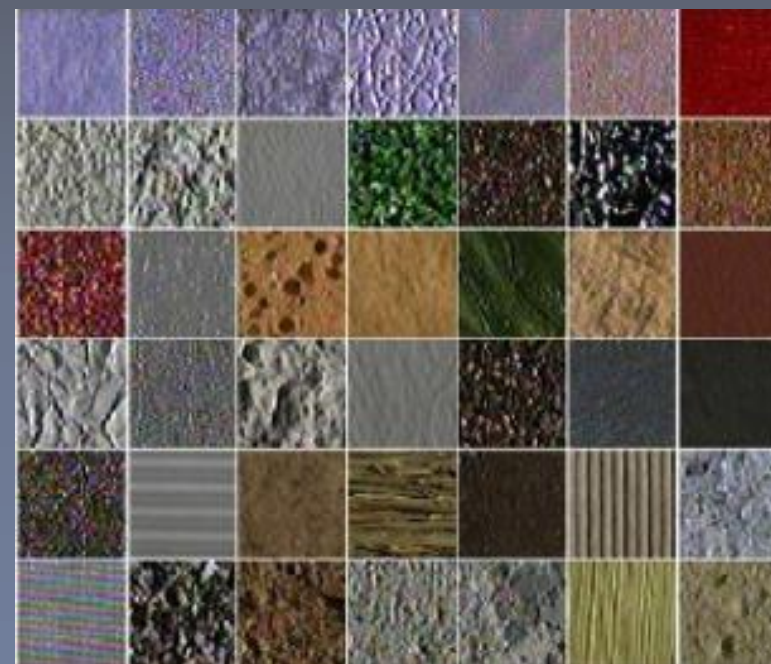




# 纹理特征

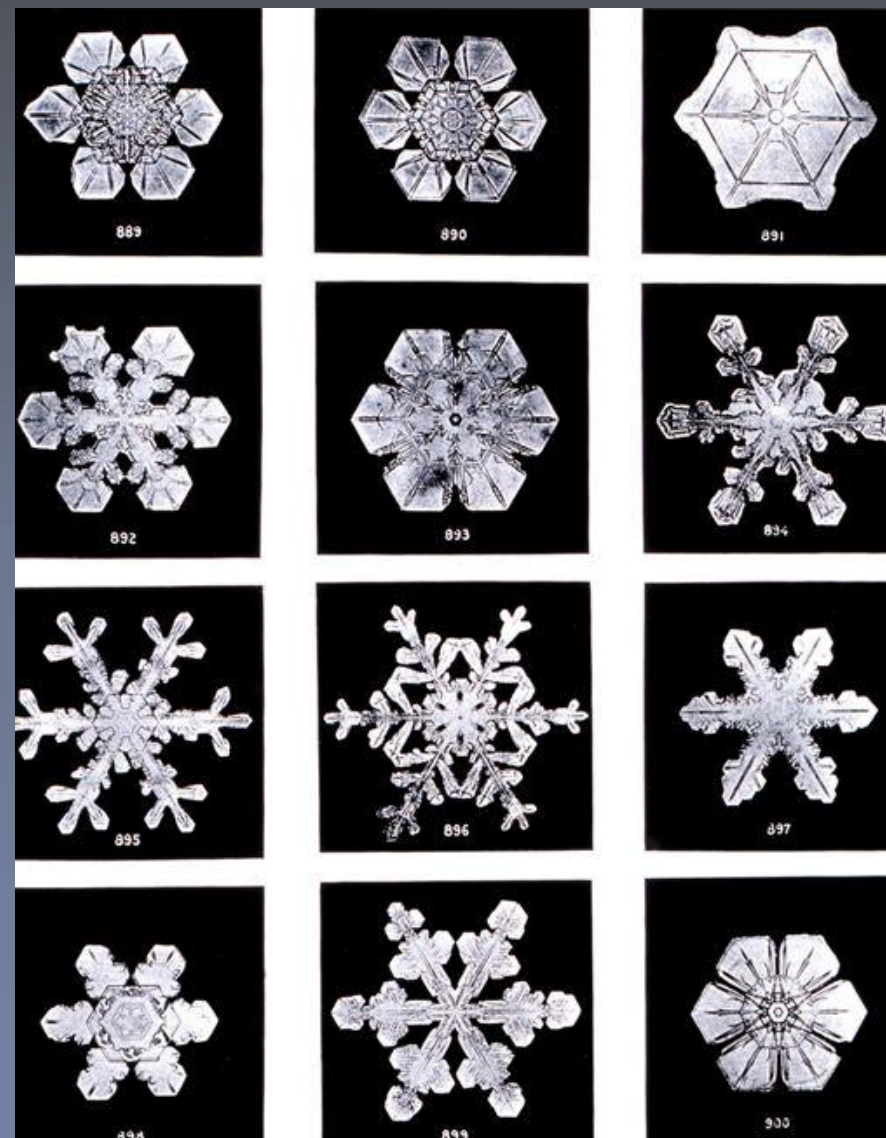
---

- 纹理特征也是一种全局特征，它也描述了图像或图像区域所对应景物的表面性质。但由于纹理只是一种物体表面的特性，并不能完全反映出物体的本质属性，所以仅仅利用纹理特征是无法获得高层次图像内容的。



# 形状特征

- 形状特征：形状特征有两类表示方法，一类是轮廓特征，另一类是区域特征。图像的轮廓特征主要针对物体的外边界，而图像的区域特征则描述了是图像中的局部形状特征。







# 空间关系特征

- 空间关系特征，是指图像中分割出来的多个目标之间的相互的空间位置或相对方向关系
- 这些关系也可分为连接/邻接关系、交叠/重叠关系和包含/独立关系等。





## 2. 形状特征描述

---



# HOG特征提取

---

- 方向梯度直方图 (Histogram of Oriented Gradient, HOG) 特征是一种在计算机视觉和图像处理中用来进行物体检测的特征描述子。
- 它通过计算和统计图像局部区域的梯度方向直方图来构成特征。
- Hog特征结合SVM分类器已经被广泛应用于图像识别中，尤其在行人检测中获得了极大的成功。
- 主要思想：在一副图像中，目标的形状能够被梯度或边缘的方向密度分布很好地描述。

# HOG实现过程

---

- 灰度化（将图像看做一个 $x, y, z$ （灰度）的三维图像）；
- 采用Gamma校正法对输入图像进行颜色空间的标准化（归一化）；
- 计算图像每个像素的梯度（包括大小和方向）；
- 将图像划分成小cells；
- 统计每个cell的梯度直方图（不同梯度的个数），得到cell的描述子；
- 将每几个cell组成一个block，得到block的描述子；
- 将图像image内的所有block的HOG特征descriptor串联起来就可以得到HOG特征，该特征向量就是用来目标检测或分类的特征。



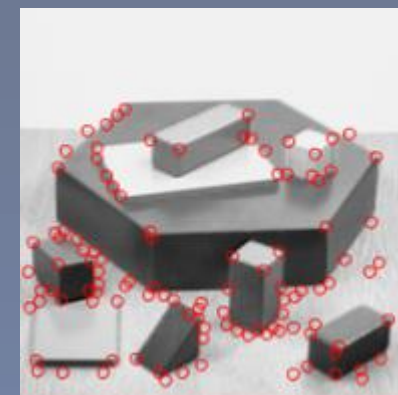
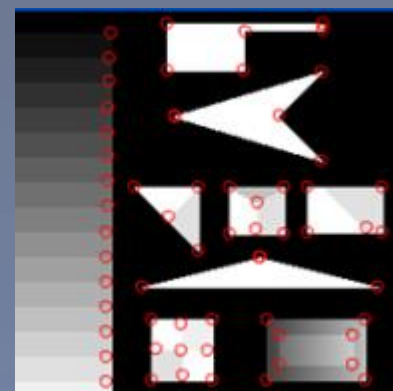
# 角点概念

---

- 角点：在现实世界中，角点对应于物体的拐角，道路的十字路口、丁字路口等。  
从图像分析的角度来定义角点可以有以下两种定义：
  - 角点可以是两个边缘的交点；
  - 角点是邻域内具有两个主方向的特征点；
- 角点计算方法：
  - 前者通过图像边缘计算，计算量大，图像局部变化会对结果产生较大的影响；
  - 后者基于图像灰度的方法通过计算点的曲率及梯度来检测角点；

# Harris角点检测

- 角点所具有的特征：
  - 轮廓之间的交点；
  - 对于同一场景，即使视角发生变化，通常具备稳定性质的特征；
  - 该点附近区域的像素点无论在梯度方向上还是其梯度幅值上有着较大变化；
- 性能较好的角点：
  - 检测出图像中“真实”的角点
  - 准确的定位性能
  - 很高的重复检测率
  - 噪声的鲁棒性
  - 较高的计算效率





# Harris实现过程

---

- 计算图像在X和Y方向的梯度;

$$I_x = \frac{\partial I}{\partial x} = I \otimes (-1 \ 0 \ 1), \quad I_y = \frac{\partial I}{\partial y} = I \otimes (-1 \ 0 \ 1)^T$$

- 计算图像两个方向梯度的乘积;

$$I_x^2 = I_x \cdot I_x, \quad I_y^2 = I_y \cdot I_y, \quad I_{xy} = I_x \cdot I_y$$

- 使用高斯函数对三者进行高斯加权, 生成矩阵M的A, B, C;

$$A = g(I_x^2) = I_x^2 \otimes w, \quad C = g(I_y^2) = I_y^2 \otimes w, \quad B = g(I_{xy}) = I_{xy} \otimes w$$

- 计算每个像素的Harris响应值R, 并对小于某一阈值t的R置为零;
- 在 $3 \times 3$ 或 $5 \times 5$ 的邻域内进行非最大值抑制, 局部最大值点即为图像中的角点;



# Harris 代码应用

Open 中的函数 `cv2.cornerHarris()` 可以用来进行角点检测。参数如下：

`img` - 数据类型为 `float32` 的输入图像

`blockSize` - 角点检测中要考虑的领域大小

`ksize` - Sobel 求导中使用的窗口大小

`k` - Harris 角点检测方程中的自由参数, 取值参数为 `[0.04, 0.06]`

```
import cv2
import numpy as np

filename = 'harris2.png'
img = cv2.imread(filename)
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
gray = np.float32(gray)

# 输入图像必须是 float32 , 最后一个参数在 0.04 到 0.06 之间
dst = cv2.cornerHarris(gray, 2, 3, 0.06)

# 结果进行膨胀, 可有可无
dst = cv2.dilate(dst, None)

# 设定阈值, 不同图像阈值不同
img[dst>0.01*dst.max()]=[0, 0, 255]
```

代码讲解





# SIFT算法

---

- SIFT，即尺度不变特征变换算法（Scale-invariant feature transform, SIFT），是用于图像处理领域的一种算法。SIFT具有尺度不变性，可在图像中检测出关键点，是一种局部特征描述子。
- 其应用范围包含物体辨识、机器人地图感知与导航、影像缝合、3D模型建立、手势辨识、影像追踪和动作比对。

# SIFT实现过程

---

- SIFT特性:

- 独特性, 也就是特征点可分辨性高, 类似指纹, 适合在海量数据中匹配。
- 多量性, 提供的特征多。
- 高速性, 就是速度快。
- 可扩展, 能与其他特征向量联合使用。

- SIFT特点:

- 旋转、缩放、平移不变性
- 解决图像仿射变换, 投影变换的关键的匹配
- 光照影响小
- 目标遮挡影响小
- 噪声景物影响小



# SIFT算法步骤

---

- SIFT算法步骤:

- 尺度空间极值检测点检测
- 关键点定位: 去除一些不好的特征点, 保存下来的特征点能够满足稳定性等条件
- 关键点方向参数: 获取关键点所在尺度空间的邻域, 然后计算该区域的梯度和方向, 根据计算得到的结果创建方向直方图, 直方图的峰值为主方向的参数
- 关键点描述符: 每个关键点用一组向量(位置、尺度、方向)将这个关键点描述出来, 使其不随着光照、视角等等影响而改变
- 关键点匹配: 分别对模板图和实时图建立关键点描述符集合, 通过对比关键点描述符来判断两个关键点是否相同



# SIFT代码实现

- 返回的关键点是一个带有很多不用属性的特殊结构体，属性当中有坐标，方向、角度等。
- 使用sift.compute()函数来进行计算关键点描述符
  - `kp, des = sift.compute(gray, kp)`
- 如果未找到关键点，可使用函数  
`sift.detectAndCompute()` 直接找到关键点并计算。
- 在第二个函数中，kp为关键点列表，des为numpy的数组，为关键点数目 $\times$ 128

```
import cv2
import numpy as np

img = cv2.imread('harris2.png')
gray= cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
sift = cv2.xfeatures2d.SIFT_create()

kp = sift.detect(gray, None) #找到关键点

img=cv2.drawKeypoints(gray, kp, img) #绘制关键点
cv2.imshow('sp', img)
cv2.waitKey(0)
```





# 3. LBP纹理特征

---



# LBP 介绍

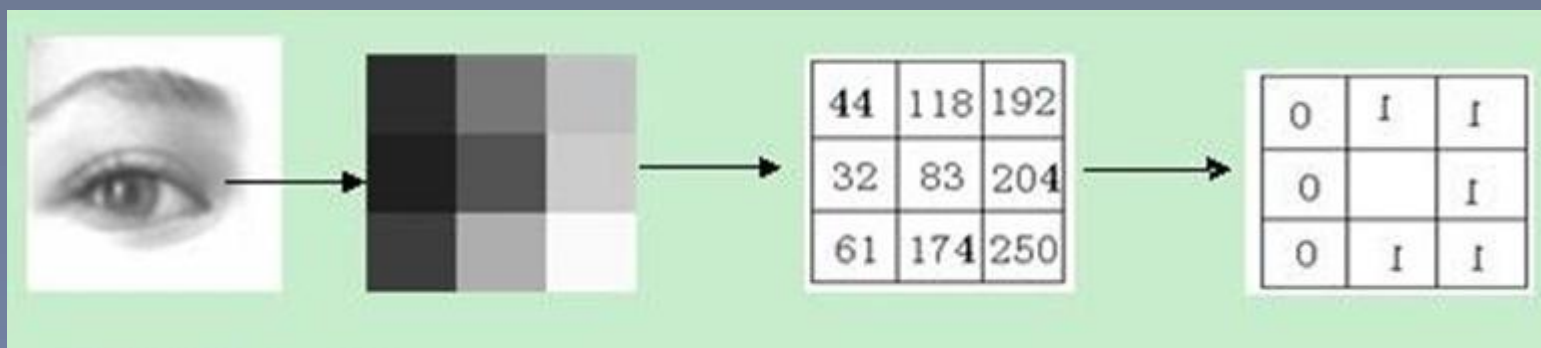
---

- LBP (Local Binary Pattern, 局部二值模式), 是一种用来描述图像局部纹理特征的算子; 它具有旋转不变性和灰度不变性等显著的优点;
- 作者: T. Ojala, M. Pietikäinen, 和 D. Harwood ;
- 提出时间: 1994年



# LBP原理

- LBP算子定义在一个 $3 \times 3 \times 3$ 的窗口内，以窗口中心像素为阈值，与相邻的8个像素的灰度值比较，若周围的像素值大于中心像素值，则该位置被标记为1；，否则标记为0。如此可以得到一个8位二进制数（通常还要转换为10进制，即LBP码，共256种），将这个值作为窗口中心像素点的LBP值，以此来反应这个 $3 \times 3$ 区域的纹理信息。



# LBP原理

---

- 数学公式：

$$LBP(x_c, y_c) = \sum_{p=1}^8 s(I(p) - I(c)) * 2^p$$

- 其中，p表示3×3窗口中除中心像素点外的第p个像素点；
- I(c)表示中心像素点的灰度值，I(p)表示领域内第p个像素点的灰度值；
- s(x)公式如下：

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & otherwise \end{cases}$$

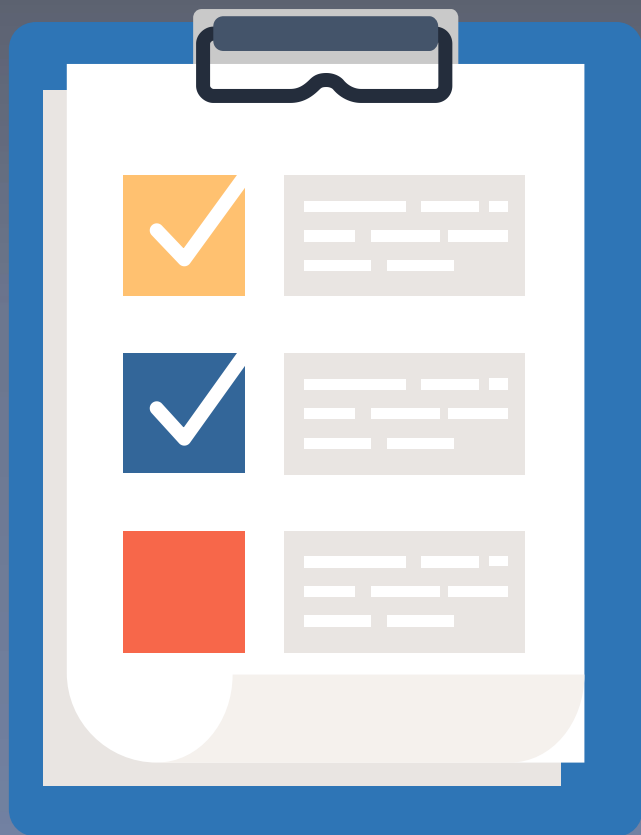
# LBP 原理

---

- LBP记录的是中心像素点与领域像素点之间的差值
- 当光照变化引起像素灰度值同增同减时，LBP变化并不明显
- LBP对与光照变化不敏感，LBP检测的仅仅是图像的纹理信息；



# 本课总结



## 核心知识

- ✓ 图像特征的理解
- ✓ HOG原理及使用HOG进行人体检测
- ✓ 图像角点检测原理和实现



重点 重点来了!

# 本节作业

Preview of next lesson

---



## 01 本节代码复现和理解

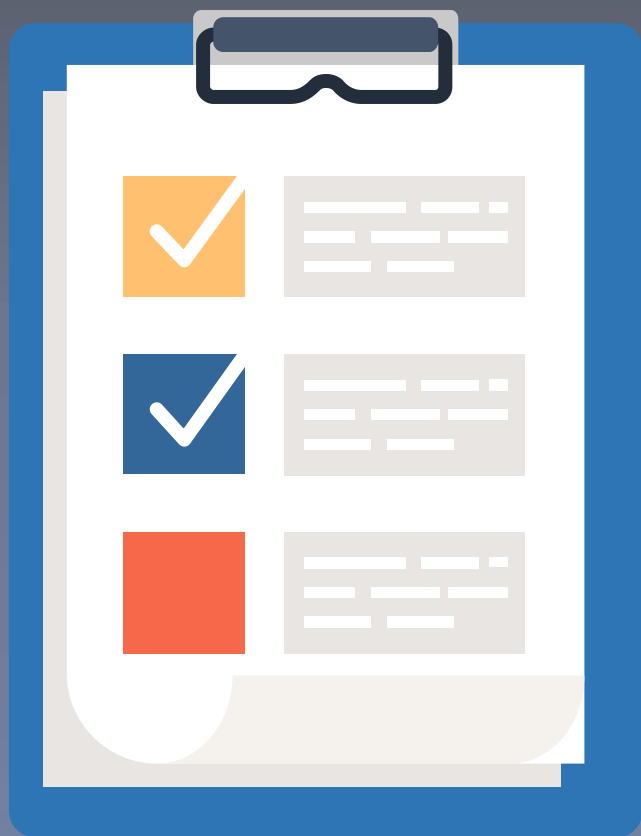
回顾本节知识点，将本节列举的例子进行代码复现，关键代码手动输入，加深印象



## 02 使用Hog和harris算法对图像进行特征点提取

# 下节预告

Preview of next lesson



01 图像分割中阈值分割

---

02 边缘检测算法

---

03 连通域分析/图像轮廓

---

04 其他区域生长算法





深度之眼  
deepshare.net

联系我们：

电话：18001992849

邮箱：[service@deepshare.net](mailto:service@deepshare.net)

QQ：2677693114



公众号



客服微信

