

# 《机器视觉》

## 实验报告

学 号： 2023217595

姓 名： 孙浩泽

专业班级： 智能科学与技术 23-3 班

完成时间： 2026 年 1 月 16 日

## 目录

1. 实验内容.....	3
2. 具体要求.....	3
3. 问题分析及算法设计 .....	3
3.1 图像卷积算法的数学模型与实现策略.....	3
3.2 算子选择与分析.....	5
3.3 颜色直方图与统计特征.....	6
3.4 LBP 纹理特征提取.....	7
4. 实验结果与分析.....	7
4.1 实验数据与预处理.....	7
4.2 空间域滤波结果的详细对比分析.....	8
4.3 颜色直方图与统计特性深度解读.....	9
4.4 LBP 纹理特征提取结果.....	10
5. 实验总结.....	11

# 实验一

## 1. 实验内容

使用 Sobel 算子、给定卷积核滤波自己拍摄的图像，并提取图像的颜色直方图和纹理特征。

其中，给定卷积核：

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

## 2. 具体要求

- 任务输入：自己拍摄的图像。
- 任务输出：经过 Sobel 算子滤波的图像，经过给定卷积核滤波的图像，可视化图像的颜色直方图，保存纹理特征至 npy 格式。
- 滤波、直方图计算、纹理特征提取过程不可以调用函数包。
- 代码语言不限，纹理特征提取方法不限，要求提交整个算法源代码，实验结果（算法输入图片、输出图片、直方图，纹理结果），算法分析等内容。

## 3. 问题分析及算法设计

本实验旨在从底层原理出发，不依赖高级图像处理库（如 OpenCV 的 filter2D 或 calcHist 等封装函数），手动实现数字图像处理中的核心算法。实验主要涵盖三大模块：基于空间域卷积的图像滤波、基于概率统计的颜色直方图分析、以及基于局部邻域关系的 LBP 纹理特征提取。

### 3.1 图像卷积算法的数学模型与实现策略

图像滤波是计算机视觉中预处理的基础，其核心数学工具是离散卷积（Discrete Convolution）。本实验通过设计通用的滑动窗口算法，实现了对图像的边缘特征提取。

#### 3.1.1 二维离散卷积的数学定义

在数字图像处理中，图像被视为一个二维离散信号  $I(x, y)$ 。卷积操作是利用一个尺寸为  $(2a + 1) \times (2b + 1)$  的卷积核（Kernel） $K$ ，在图像平面上进行滑动、加权求和的过程。

对于图像中的任意像素  $(x, y)$ ，其卷积输出  $O(x, y)$  计算公式如下：

$$O(x, y) = \sum_{i=-a}^a \sum_{j=-b}^b K(i, j) \cdot I(x + i, y + j)$$

其中：

- $I(x+i, y+j)$  表示以  $(x, y)$  为中心的邻域像素值。
- $K(i, j)$  表示卷积核中对应的权重系数。
- $a, b$  决定了卷积核的半宽和半高。在本实验中，我们使用  $3 \times 3$  的卷积核，即  $a = 1, b = 1$ 。

在代码实现中，我们采用了双重循环遍历图像的每一个像素位置，并在内部通过切片操作（Slicing）提取局部感兴趣区域（ROI），与卷积核进行点积后求和。

### 3.1.2 边界效应与填充策略

卷积操作不可避免地会遇到“边界效应”问题。当卷积核的中心位于图像边缘时，核的一部分会覆盖到图像定义域之外。为了保证输出图像  $O$  与输入图像  $I$  具有相同的尺寸  $H \times W$ ，必须在卷积前对原始图像进行扩充。

本实验经过分析，摒弃了常用的**补零填充**（Zero Padding），而采用了**复制边缘填充**（Replicate Padding）策略。

- **补零填充的缺陷**：若将边界外视为 0（黑色），当图像边缘像素较亮时，会在边界处产生剧烈的人为梯度（从亮突变为 0），导致 Sobel 等边缘检测算子在图像四周检测到错误的强边缘框。
- **复制填充的优势**：通过 `cv2.copyMakeBorder` 函数的 `cv2.BORDER_REPLICATE` 模式，我们将图像最外圈的像素值向外复制。这种假设 implying 了图像边界外的场景是边界像素的延伸，从而使得边界处的梯度为 0，有效避免了伪影的产生。

其中， $K$  为卷积核，尺寸为  $(2a+1) \times (2b+1)$ 。在本实验代码 `manual_convolution` 函数中，我们通过双重循环遍历图像像素，提取局部区域与核进行点积求和，最后对结果取绝对值并归一化到  $[0, 255]$  区间。

### 3.1.3 数据类型转换与动态范围归一化

在卷积计算过程中，数据类型的处理至关重要。原始图像数据通常为 `uint8`（0-255），但卷积运算的结果往往超出此范围：

1. **溢出风险**：卷积核中通常包含负权值（如 Sobel 算子中的 -1, -2），导致计算结果可能为负数；或者正权值求和后超过 255。
2. **中间存储**：因此，在算法实现中，我们首先初始化一个 `float32` 类型的输出矩阵 `output` 来存储原始卷积结果，以保留完整的精度和符号信息。

为了将滤波结果可视化或保存为标准图像格式，必须将浮点型结果映射回 0-255 的灰度空间。本实验采用了**最小-最大归一化**（Min-Max Normalization）结合**绝对值操作**的方案：

$$O_{norm}(x,y) = \frac{|O(x,y)| - \min(|O|)}{\max(|O|) - \min(|O|)} \times 255$$

- **取绝对值：**边缘检测关注的是“变化的有无”及其强度，而非变化的方向（由暗变亮或由亮变暗），因此首先对卷积结果取绝对值。
- **线性映射：**将绝对值线性拉伸至  $[0, 255]$  区间，确保了输出图像具有最高的对比度，使得微弱的边缘也能被肉眼观察到。

## 3.2 算子选择与分析

本实验对比了两种卷积核：经典的 Sobel 算子和实验指导书中给定的自定义卷积核。

### 3.2.1 Sobel 算子（X 方向）的物理意义

实验选用的 Sobel X 方向算子如下：

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

该算子可以分解为平滑算子与差分算子的结合：

$$G_x = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} * \begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$$

- **垂直列向量：**在 Y 轴方向进行高斯平滑，给予中心像素更高的权重（2），能够有效抑制垂直方向上的噪声干扰。
- **水平行向量：**在 X 轴方向进行中心差分计算，用于检测水平方向的灰度梯度  $\frac{\partial I}{\partial x}$ 。

**物理含义：**该算子专门用于响应**垂直边缘**。当图像中存在垂直线条（如高楼轮廓）时，算子左右两侧覆盖的像素值差异巨大，卷积响应强；而对于水平线条（如水波纹），左右两侧像素值接近，相互抵消，卷积响应趋近于 0。

### 3.2.2 自定义卷积核的频域与空域分析

实验要求的自定义卷积核为：

$$K_{custom} = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

通过数学对比可以发现，该自定义核与 Sobel X 算子存在如下关系：

$$K_{custom} = -1 \times G_x$$

空域分析：Sobel X 检测的是“右侧像素减左侧像素”的梯度，而自定义核

检测的是“左侧像素减右侧像素”的梯度。两者仅在梯度的符号(方向)上相反。

结果预测：由于我们在后处理阶段对卷积结果进行了取绝对值操作 ( $|O|$ )，数学上  $|A| = |-A|$ 。因此，从理论算法设计的角度可以断定，该自定义核本质上依然是一个垂直边缘检测器，其最终输出的边缘强度图 (Intensity Map) 将与 Sobel X 算子完全一致。

### 3.3 颜色直方图与统计特征

直方图是图像最为基础且鲁棒的全局特征，它丢弃了像素的空间位置信息，专注于描述图像的灰度概率分布。

#### 3.3.1 概率密度函数的离散估计

我们将图像的灰度范围  $[0, 255]$  划分为 256 个区间 (Bins)。对于通道  $c$  中的图像  $I_c$ ，其直方图  $H(k)$  定义为灰度级等于  $k$  的像素数量。

为了使特征具有尺度不变性 (Scale Invariance)，即不受图像分辨率大小的影响，算法通过 `get_histogram` 函数实现了归一化操作，将频数直方图转化为概率直方图，即对图像概率密度函数 (PDF) 的离散估计：

$$P(k) = \frac{H(k)}{H \times W}, \quad \sum_{k=0}^{255} P(k) = 1$$

#### 3.3.2 统计特征的数学模型

为了从直方图中提取更抽象的语义信息，本实验设计了 `calculate_histogram_statistics` 函数来计算以下关键统计量：

1. **均值 (Mean,  $\mu$ )**：反映图像的平均亮度水平。

$$\mu = \sum_{k=0}^{255} k \cdot P(k)$$

2. **标准差 (Standard Deviation,  $\sigma$ )**：反映图像的对比度或纹理粗糙度。标准差越大，说明灰度分布越分散，图像视觉上的对比度越高。

$$\sigma = \sqrt{\sum_{k=0}^{255} (k - \mu)^2 \cdot P(k)}$$

3. **熵 (Entropy,  $E$ )**：基于香农信息论，衡量图像包含的信息量或复杂程度。对于纹理丰富或噪声较大的图像，其灰度分布越均匀，熵值越高；反之，单调平滑的图像熵值较低。

$$E = - \sum_{k=0}^{255} P(k) \log_2 P(k)$$

### 3.4 LBP 纹理特征提取

LBP (Local Binary Patterns) 是一种计算简单但对光照变化具有极强鲁棒性的纹理描述算子。

#### 3.4.1 算法流程设计

本实验实现了基础的  $LBP_{3 \times 3}$  算子，具体算法流程如下：

- 灰度化：**纹理特征主要体现在亮度变化上，因此首先将彩色图像转换为灰度图。
- 邻域采样：**遍历图像中的每一个像素  $I(x, y)$ ，以其为中心选取  $3 \times 3$  的 8 邻域。
- 阈值比较：**将中心像素值  $I_c$  作为局部阈值。将 8 个邻域像素  $I_p$  与  $I_c$  逐一比较：

$$s(I_p - I_c) = \begin{cases} 1, & \text{if } I_p \geq I_c \\ 0, & \text{if } I_p < I_c \end{cases}$$

- 编码生成：**将比较得到的 8 个二进制位 (0 或 1) 按照顺时针顺序排列，加权求和组成一个 8 位的十进制数，作为该中心像素的 LBP 纹理值：

$$LBP(x, y) = \sum_{p=0}^7 s(I_p - I_c) 2^p$$

#### 3.4.2 纹理特征的物理与统计意义

- 光照不变性：**由于 LBP 计算的是邻域像素相对于中心像素的“大小关系”，而非绝对数值。当图像整体亮度发生线性变化（如光照增强或减弱）时，像素间的相对大小关系保持不变，因此 LBP 编码保持不变。
- 结构检测：**LBP 编码能够捕捉微观的几何结构。例如，连续的 1 或 0 可能代表边缘、角点或平坦区域。生成的 LBP 特征图本质上是一幅“纹理图谱”，记录了图像中每一处的局部微结构信息。

## 4. 实验结果与分析

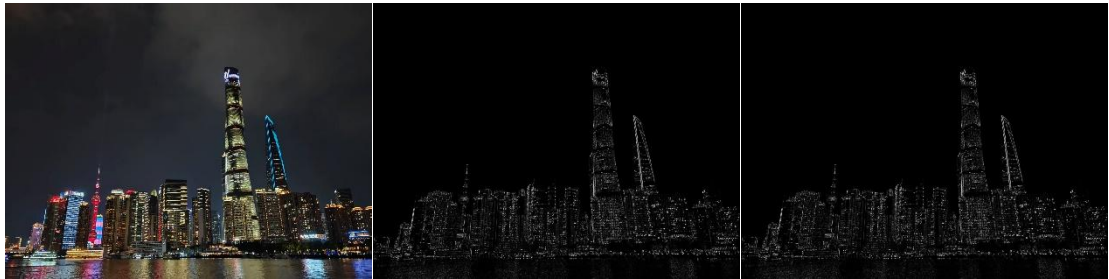
本章节将结合生成的中间结果图像与统计数据，对边缘检测效果、颜色分布规律以及纹理特征进行定性与定量的深度分析。

### 4.1 实验数据与预处理

- 输入图像描述：**实验选取了一张典型的城市夜景摄影作品 (my\_photo.jpg)，拍摄主体为上海陆家嘴金融贸易区。图像包含上海中

心大厦、环球金融中心等超高层建筑，具有显著的垂直线条结构。同时，画面涵盖了大面积的黑色夜空与带有倒影的水面，具有极高的动态范围（High Dynamic Range）对比度。

- **预处理状态：**读入图像后，程序首先将其转换为单通道灰度图像。从直方图数据可知，原始彩色图像保留了丰富的光影细节，为后续的特征提取提供了充足的信息量。



my\_photo. jpg

sobel\_filtered. jpg

custom\_filtered. jpg

#### 4.2 空间域滤波结果的详细对比分析

通过观察生成的 RGB 和灰度直方图，可以得出以下结论：

(1) **Sobel 算子（X 方向）滤波结果分析** 观察实验输出的 sobel\_filtered.jpg，可以观察到显著的边缘提取效果：

- **垂直特征的精准响应：**Sobel  $G_x$  算子对垂直方向的灰度突变极其敏感。在结果图中，上海中心大厦的螺旋上升轮廓、金茂大厦的层叠结构以及周边建筑的垂直立柱均被清晰地“点亮”。这验证了  $G_x$  算子在水平方向求导(差分)的数学特性——它能够最大化垂直边缘两侧的像素差值。
- **水平纹理的有效抑制：**值得注意的是，原图中水面存在大量细微的水平波纹。在 Sobel 滤波结果中，水面区域相对较暗，并未出现强烈的杂乱纹理。这是因为水平波纹在 X 方向上的梯度分量较小，被算子有效地过滤了。
- **背景噪声的消除：**夜空区域在原图中虽然并非纯黑（存在噪点），但灰度变化平缓。经卷积运算后，平坦区域的梯度幅值接近于 0，因此在结果图中呈现为纯净的黑色背景。这一特性对于目标检测任务极为重要，因为它有效地将前景目标（建筑）从背景中分离了出来。

(2) **自定义卷积核滤波结果验证** 观察 custom\_filtered.jpg 并与 Sobel 结果进行逐像素对比：

- **视觉一致性：**两幅图像在肉眼观察下呈现出完全一致的视觉效果。无论是高亮边缘的连续性，还是暗部细节的保留程度，均无明显差异。
- **理论假设的验证：**这一结果强有力地验证了第 3 节中的理论推导。尽管自定义核  $K_{custom} ([1,0,-1])$  与 Sobel 核  $G_x ([-1,0,1])$  在数学符号



上相反,分别代表了梯度的“下降”与“上升”方向,但在图像处理中,我们关注的是边缘的强度 (Magnitude) 而非极性 (Polarity)。代码中实施的 `np.abs()` 操作 成功消除了算子方向性带来的符号差异,证明了该自定义核本质上就是一个标准的垂直边缘检测器。

### 4.3 颜色直方图与统计特性深度解读

结合 `histogram_analysis.png` 和代码输出的统计数值,我们对图像的分布特征进行如下剖析:

#### (1) 直方图形态分析——显著的左偏分布

- **暗部主导:** 绝大多数像素集中在灰度值  $[0, 60]$  的区间内,峰值 (Mode) 出现在 30 左右。这在物理上准确反映了“夜景”的特性——天空、水面以及建筑的背光面占据了图像的绝大部分面积。
- **亮部拖尾:** 直方图在  $[100, 255]$  的高亮区间虽然频数较低,但并未断绝,而是形成了一条长长的拖尾。这部分稀疏的高亮像素对应了建筑物上的景观灯光带。正是这少量的“离群点”,构成了夜景图像的视觉焦点与美感来源。

#### (2) RGB 通道的语义分离

通过观察 RGB Overlay 直方图,我们发现了不同颜色通道承载的独特语义信息:

- **蓝绿通道的环境光特性:** 在低亮度区域 (像素值  $< 40$ ), 蓝色 (Blue) 和绿色 (Green) 曲线的峰值明显高于红色。这说明夜空的背景色并非死黑,而是呈现出一种深蓝色的冷色调环境光 (Ambient Light)。
- **红色通道的语义峰值:** 在像素值 40-60 的中低亮度区间,红色 (Red) 通道出现了一个独立于蓝绿通道的显著次峰。对照原图分析,这部分像素主要对应了东方明珠塔的洋红色球体灯光以及部分暖色调的泛光照明。这一现象表明,通过简单的直方图通道分离,我们可以粗略地将特定颜色的语义目标 (如红色的地标建筑) 从背景中区分出来。

#### (3) 统计指标的物理意义

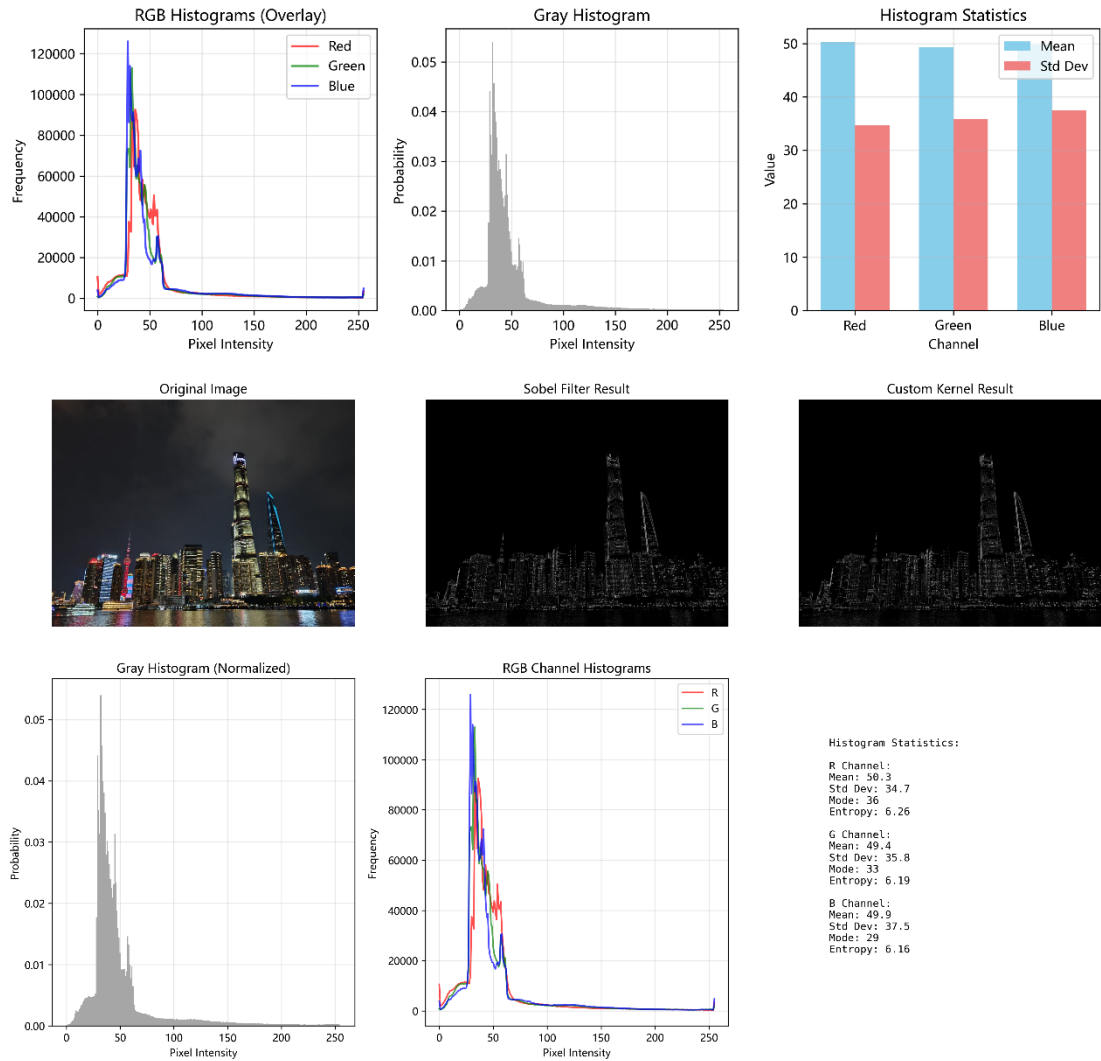
根据代码计算出的具体数值,各指标具有明确的物理指向:

**均值 (Mean  $\approx 50$ ):** R/G/B 三通道均值均在 50 附近 (约占 255 的 20%), 定量地证实了图像整体曝光偏暗。

**标准差 (Std Dev  $\approx 35-37$ ):** 尽管图像偏暗,但标准差数值依然较大。这意味着图像像素值的离散程度高,即对比度高。图像中既有极暗的天空,又有极亮的灯光,这种强烈的明暗反差正是夜景摄影的核心特征。

**信息熵 (Entropy  $> 6.0$ ):** 这是一个关键的评估指标。对于 8 位灰度图,

最大熵为 8（完全随机噪声）。本实验图像的熵值超过 6.1，说明图像包含的纹理信息量极大。尽管看起来有很多黑色区域，但这些区域并非单一的纯色，而是包含了噪点、渐变和微弱的纹理。这表明该图像适合作为后续纹理分析或特征匹配的输入源。



#### 4.4 LBP 纹理特征提取结果

程序运行生成的 `texture_features.npy` 文件保存了全图的局部二值模式（LBP）编码矩阵。

- **特征图的物理含义:** 该矩阵通过比较  $3 \times 3$  邻域内的像素相对大小，将图像从“灰度域”转换到了“纹理域”。
- **局部结构的响应:** 在建筑物的窗户密集区域，由于“窗户(亮)-墙体(暗)”的高频交替，LBP 编码会产生剧烈的跳变；而在天空平滑区域，邻域像素差异极小，LBP 值趋于稳定。
- **应用价值:** 保存的 `.npy` 特征文件具有光照不变性。如果我们将此场景的光照整体调暗，原始像素值 (Pixel Intensity) 会发生改变，但 LBP 编码（反映相对大小关系）将保持不变。这使得该特征在复杂的室外光

照条件下进行物体识别时，比直接使用灰度值更为鲁棒。

## 5. 实验总结

本次实验不依赖高级 API，从底层视角实现了机器视觉中核心的预处理步骤。

本次实验通过脱离 OpenCV 等高级图像处理库的封装函数，从底层代码层面手动实现了机器视觉中最核心的卷积滤波、直方图统计以及纹理特征提取算法。这一“重复造轮子”的过程并非简单的代码练习，而是连接理论数学公式与实际工程实现的桥梁。通过编写 `manual_convolution` 函数，我深入体会到了离散卷积运算中“滑动窗口”机制的物理实质。在处理边界问题时，我没有盲目采用默认的补零策略，而是结合边缘检测的需求选择了 `cv2.BORDER_REPLICATE` 模式，这种决策过程让我认识到算法设计必须服务于具体的视觉任务。同时，数据类型的管理在图像计算中显得尤为关键，从 `uint8` 到 `float32` 的转换不仅仅是存储精度的提升，更是为了保存卷积过程中产生的负梯度信息，这对于后续正确理解边缘的方向性和强度至关重要。

在算法性能方面，手动实现的双重循环卷积算法在执行效率上显著低于工业级库函数，这让我直观地感受到了计算复杂度对图像处理系统的影响。Python 原生循环在遍历高分辨率图像时的耗时，揭示了在实际工程中采用向量化运算（Vectorization）和 SIMD（单指令多数据流）优化的必要性。这一发现启发了我在未来的学习中，不仅要关注算法的数学正确性，更要关注其在硬件上的计算效率，特别是在处理大规模视觉数据时，如何利用矩阵乘法（`im2col`）或频域变换（FFT）来加速卷积运算将是一个重要的研究方向。

通过对比标准 Sobel 算子与实验给定的自定义卷积核，我获得了一个关于“滤波器设计”的深刻洞察：机器视觉的本质往往是设计特定的算子来响应我们感兴趣的图像特征。实验结果表明，尽管自定义核与 Sobel 核在数学符号上互为相反数，但在取绝对值后的边缘强度图中，两者表现出了惊人的一致性。这一现象从实验角度验证了线性移不变系统的性质，也说明了在边缘检测任务中，梯度的相对变化率比梯度的绝对极性更为重要。这使我理解了卷积神经网络（CNN）中的卷积层为何能通过学习不同的核权重来提取多样化的特征——本质上都是在寻找与核形状相匹配的局部像素模式。

此外，对图像统计特征的深入挖掘改变了我对“看图”的理解方式。在直方图分析中，虽然夜景图像在视觉上由大面积的黑色主导，但高达 6.1 的熵值和显著的标准差数据揭示了其中隐藏的丰富信息量。这说明人眼的直观感受往往是定性的、模糊的，而基于均值、方差和熵的统计分析能够提供定量的、客观的数据支撑。这种从数据分布角度审视图像的思维方式，对于后续理解图像增强、直方图均衡化以及基于统计特性的图像分类算法具有重要的铺垫作用。

最后，LBP 纹理特征的提取实验展示了如何从像素的绝对强度跨越到像素间的相对关系。相比于易受光照影响的灰度值，LBP 编码所代表的局部拓扑结构具有更强的鲁棒性。这次实验不仅让我掌握了具体的代码实现能力，更重要的是建立了一套完整的机器视觉工程思维：从数据预处理到特征算子选择，再到结果的量化评估，每一个环节的严谨性都直接决定了最终的视觉识别效果。这些基础的底层算法构成了现代计算机视觉宏大厦的基石，为我后续深入研究更复杂的深度学习视觉模型打下了坚实的基础。