

合肥工业大学

2025 年《机器视觉》实验



实验内容: 车道线检测

姓 名: 折浩宇

学 号: 2023217896

完成时间: 2025/12/27

一、实验目的

1. 理解并掌握车道线检测这一典型机器视觉任务：

学习从自然场景道路图像中提取结构化目标（车道线）的基本思路与方法。
2. 熟悉经典图像处理流程：

掌握灰度化、滤波去噪、边缘检测（如 Canny）、形态学处理（可选）、ROI（感兴趣区域）提取等预处理步骤，并理解其对结果的影响。
3. 掌握霍夫变换直线检测原理与应用：

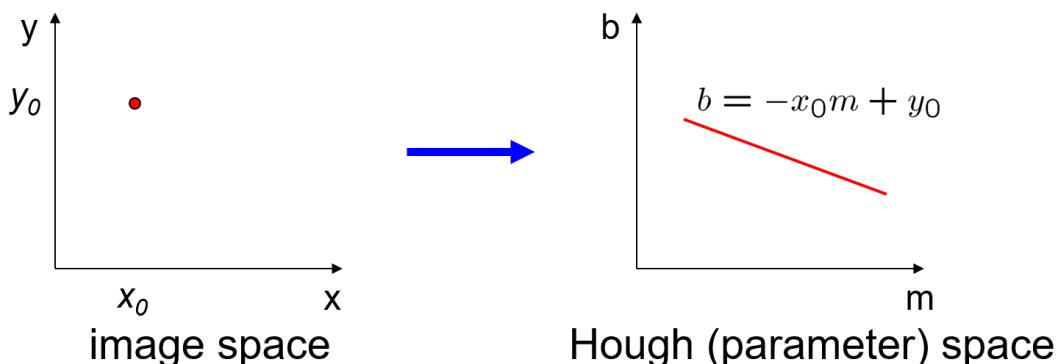
能够使用霍夫变换从边缘点中检测直线，并理解关键参数（阈值、步长、最小线段长度、最大间隔等）对检测效果的作用。
4. 实现车道线定位与可视化输出：

将检测到的直线进行筛选、分类与拟合（如左右车道线分离、斜率筛选、线段合并），最终在原图上标注车道线位置。
5. 培养实验分析与调参能力：

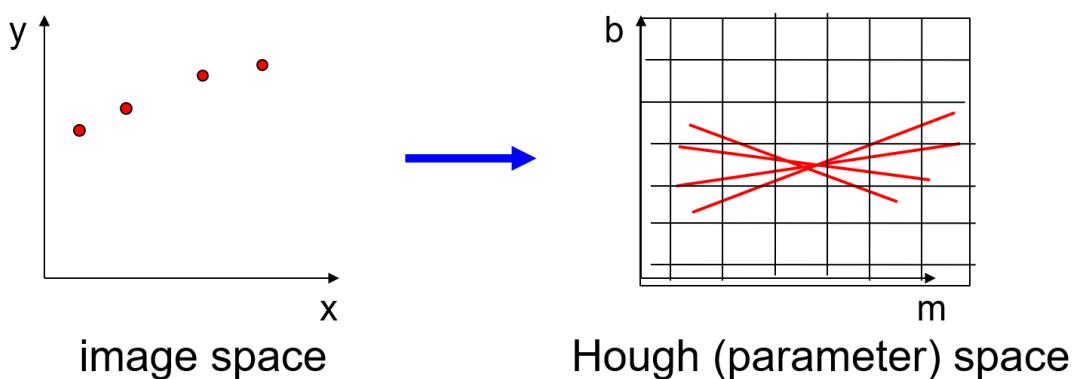
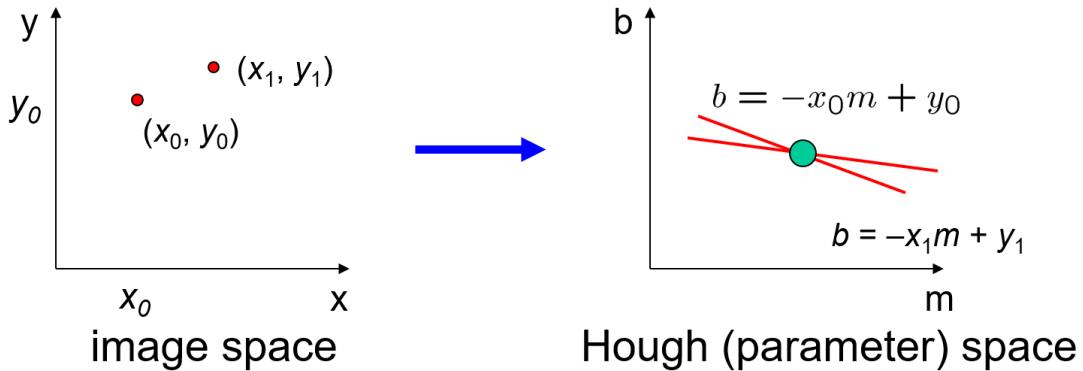
对不同光照、阴影、路面纹理、噪声等因素导致的误检/漏检进行总结，分析失败原因并提出改进方案（如自适应阈值、改进 ROI、增强边缘特征等）。

二、实验原理：霍夫变换

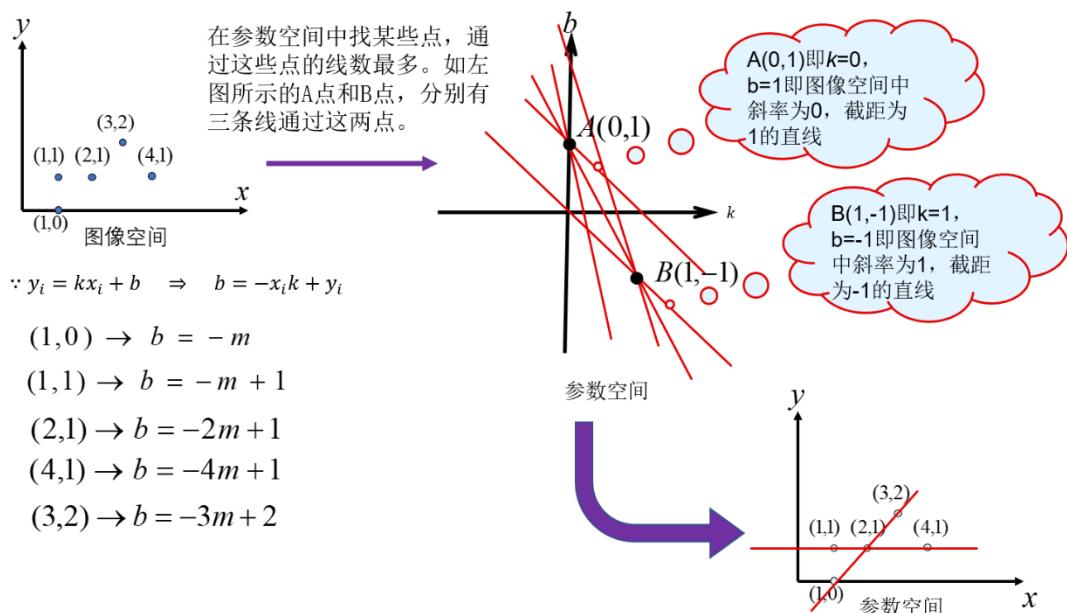
- 1) 图像空间中的点与参数空间中的直线一一对应。



- 2) 图像空间中的直线与参数空间中的点一一对应

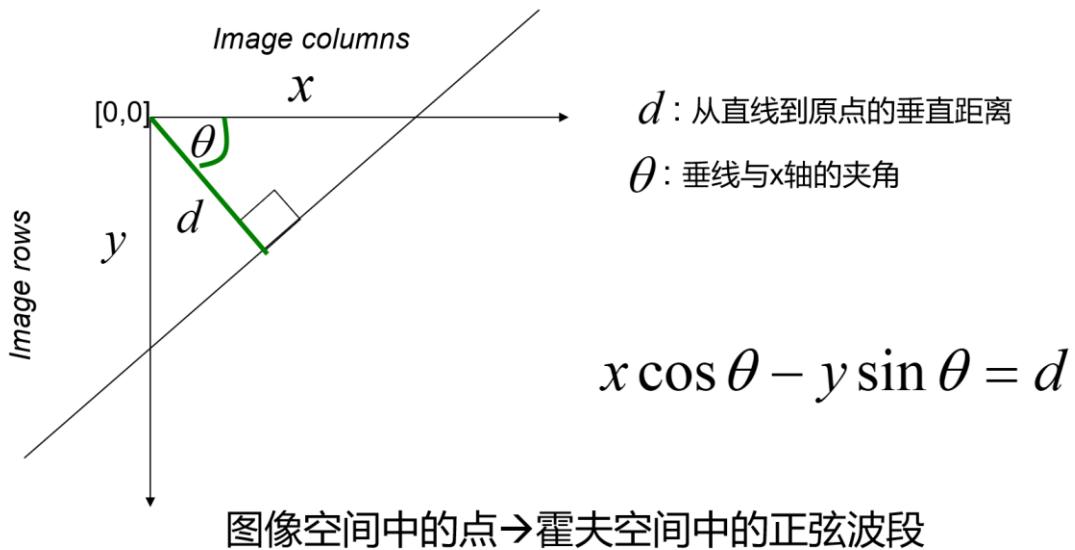


3) 把图像空间中的直线对应到参数空间中的点，最后通过统计特性来解决问题。



假如图像空间中有两条直线，那么最终在参数空间中就会对应到两个峰值点，依此类推。

4) 极坐标表示



5) 整体流程

使用极坐标参数化:

$$x \cos \theta - y \sin \theta = d$$

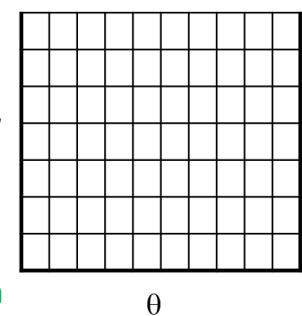
基本的霍夫变换算法

1. 初始话 $H[d, \theta] = 0$
2. for each edge point $I[x, y]$ in the image
 - for $\theta = [\theta_{\min} \text{ to } \theta_{\max}]$ // some quantization

$$d = x \cos \theta - y \sin \theta$$

$$H[d, \theta] += 1$$
3. 找出 $H[d, \theta]$ 的最大值为 (d, θ) 的值
4. 图像中检测到的直线由 $d = x \cos \theta - y \sin \theta$ 给出

H : accumulator array (votes)



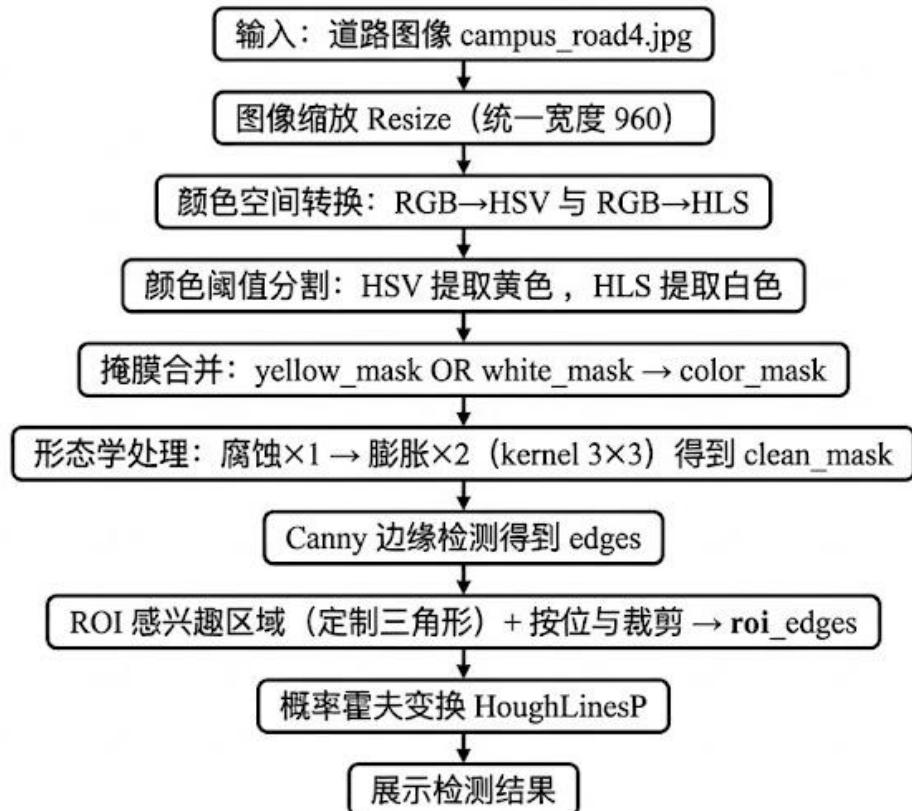
三、实验流程与解析

3.1 整体流程

本实验以道路图像为输入，首先对图像进行尺寸统一（Resize），然后分别在 HSV 与 HLS 颜色空间中通过阈值分割提取黄色车道线与白色车道线的候选区域，并将两类掩膜合并得到车道线二值图；接着对二值图进行腐蚀+膨胀等形态学处理以去除噪点并增强车道线连通性，再对处理后的结果执行 Canny 边缘

检测获得边缘点集合；随后构建梯形 ROI（感兴趣区域）并对边缘图进行裁剪，减少天空、路边结构等无关直线干扰；最后在 ROI 内使用概率霍夫变换检测直线线段，将检测到的线段叠加绘制回原图得到最终车道线检测可视化结果，同时展示霍夫空间可视化。

基于颜色阈值+边缘检测+霍夫变换的车道线检测流程



实验流程图

3.2 具体流程与结果

3.2.1. 读取图像并统一尺寸

由于原始手机拍摄的照片像素太高，导致细节过多后续识别效果很差，所以进行此操作：

1. 使用 OpenCV 读取道路原始图像。
2. 将图像按固定宽度（如 960）进行缩放，保持纵横比不变，以统一后续 ROI 位置与参数适用范围。

3.2.2. 颜色空间转换与车道线颜色阈值分割

为了能够识别车道中间的黄线和白线，我做了如下改进：

1. 将原图分别转换到 HSV 和 HLS 颜色空间。
2. 在 HSV 空间中用 `inRange()` 提取黄色车道线候选区域（如 $H \in [15, 40]$ ， S 、 V 设置下限抑制低饱和/低亮区域）。
3. 在 HLS 空间中用 `inRange()` 提取白色车道线候选区域（主要提升 L 通道下限以保留高亮白线）。
4. 将黄色掩膜与白色掩膜进行按位或运算，得到车道线候选二值图 `color_mask`。

3.2.3. 形态学去噪与连通（腐蚀 + 膨胀）

为了得到更干净，更连续的二值候选图，我做了如下操作：

1. 构造 3×3 结构元素 (`kernel`)。
2. 对 `color_mask` 执行 腐蚀 (1 次) 以去除孤立噪点、细碎毛刺。
3. 对腐蚀结果执行 膨胀 (2 次) 以恢复车道线宽度并连接断裂区域。
4. 得到更干净、更连续的二值候选图 `clean_mask`。

Clean Mask (High Threshold + Morphology)



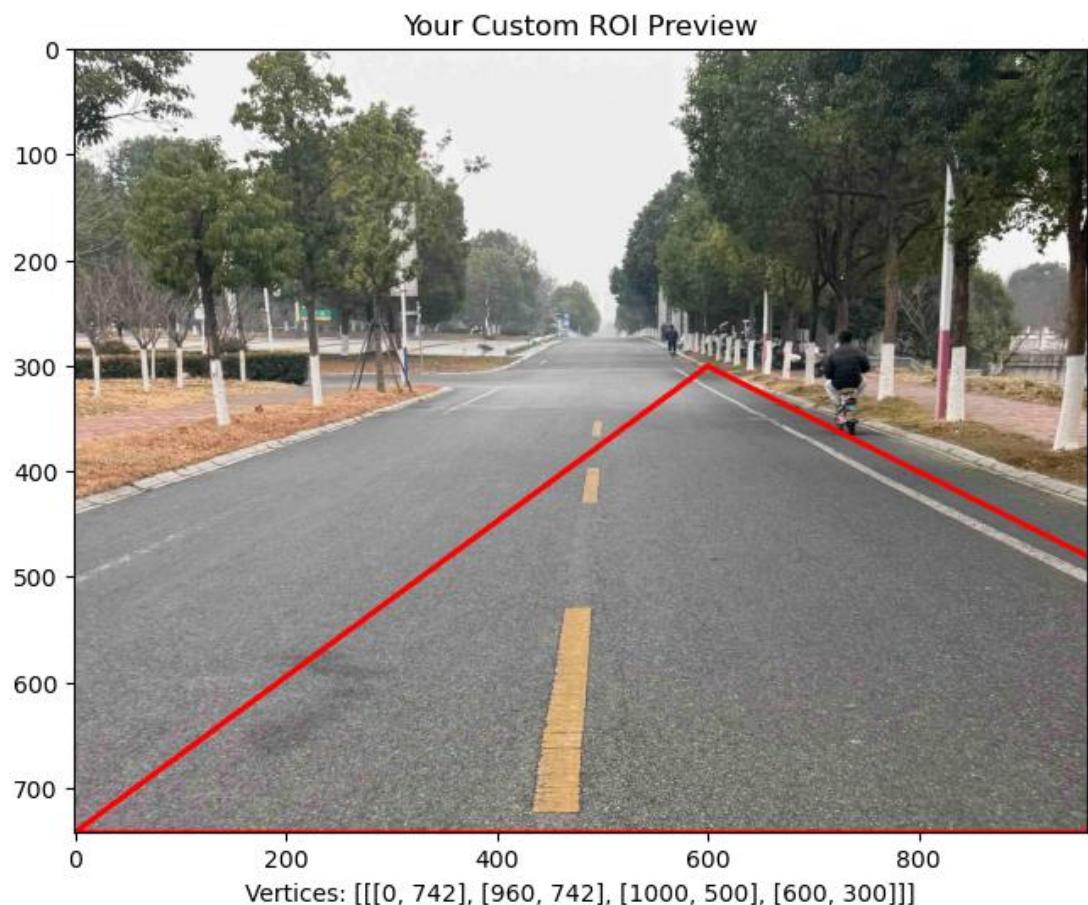
经过上面三步得到的结果

3.2.4. 边缘检测 (Canny)

1. 对 clean_mask 执行 Canny 边缘检测，得到边缘图 edges（阈值如 50/150）。
2. 该步骤将车道线候选区域转换为边缘点集合，便于后续霍夫变换进行直线检测。

3.2.5. 构建感兴趣区域 ROI 并裁剪边缘

1. 根据道路图像几何结构设定 ROI 多边形（**这里我做了一些改进，为了更好适应图像，我自己选取了 ROI 多边形的形状，为三角形**）
2. 使用 fillPoly() 生成 ROI 掩膜。
3. 将 edges 与 ROI 掩膜做按位与运算，得到只包含道路区域的边缘图 roi_edges，以减少天空、护栏等干扰直线带来的误检。



我自己选择的检测区域（三角红框）

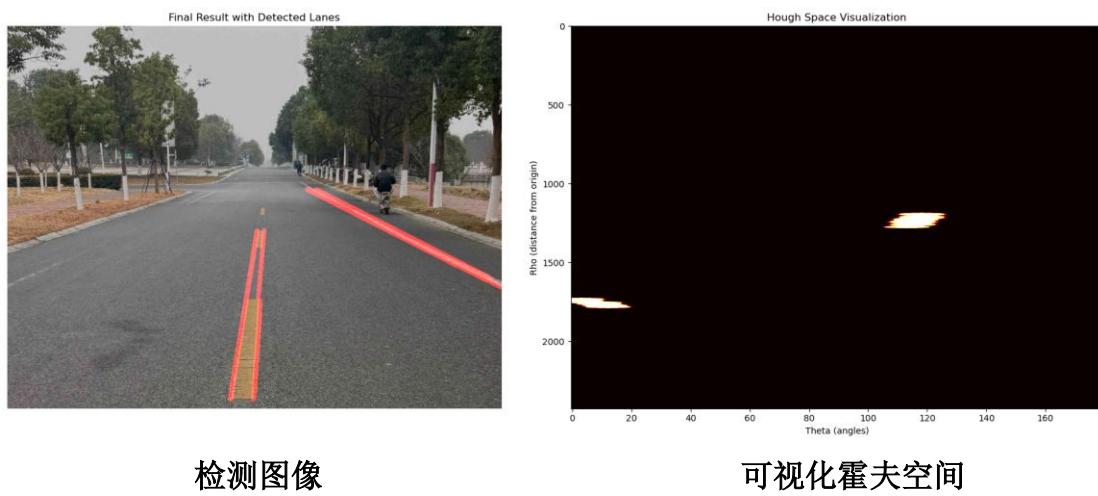


原始图像

经过 Canny 与 ROI 裁剪图像

3.2.6. 霍夫变换检测车道线（线段/直线）

1. 对 `roi_edges` 执行 概率霍夫变换 `HoughLinesP` 检测线段，获得若干线段端点 $((x_1, y_1, x_2, y_2))$ 。
 - 关键参数：
 - `rho`、`theta`: 累加器分辨率
 - `threshold`: 投票阈值
 - `minLineLength`: 最短线段长度（过滤噪声短线）
 - `maxLineGap`: 允许断裂间隙（连接同一直线上的分段）
2. 可同时调用 `HoughLines` 得到 $((\rho, \theta))$ 用于霍夫空间结果示意展示。



检测图像

可视化霍夫空间

可以明显看到断掉的黄线被连接上，而且右边的白线可以正常检测出来

四、心得体会

通过本次实验，我对机器视觉中“从原始图像到结构化结果”的整体思路有了更深的体会。之前对车道线检测的理解更多停留在“做边缘、做直线检测”，但实际动手后发现：**在真实道路场景里，图像信息非常复杂，路面纹理、阴影、反光、路边设施、标志线等都会产生大量干扰特征，如果不先对输入进行约束和筛选，后续任何检测算法都会被噪声淹没。**这次实验让我意识到，机器视觉的关键并不只是某一个“核心算法”，而是各个模块之间的配合与衔接，尤其是前面步骤产生的中间结果质量，会直接决定后续检测是否可靠。

实验过程中我最明显的收获是：对“先验”和“约束”的理解更具体了。比如把图像转换到不同颜色空间再做阈值分割，看似只是换了一种表示方法，但实际效果差异非常明显。**HSV 对颜色类别的表达更直观，适合把黄色车道线从路面背景里分离出来；而 HLS 里亮度信息更突出，用来提取白色车道线更稳。**通过对比掩膜结果我感受到，选择合适的颜色空间相当于在特征层面做了一次“降维筛选”，将大量与目标无关的信息提前过滤，这种“把问题变简单”的思路比盲目提升算法复杂度更有效。与此同时，我也体会到阈值并不是随便给一个范围就行，它其实隐含着对场景的假设：光照强弱、摄像头成像风格、路面颜色都会影响阈值分割的质量，因此参数设置必须结合图像实际效果来判断。

形态学处理也让我有了更直观的认识。以前只知道“腐蚀、膨胀用于去噪”，但这次真正看到二值掩膜在腐蚀后噪点被削弱、膨胀后断裂区域被连接起来，才理解它在流水线中的作用是“修复结构”，而不仅仅是“变粗/变细”。特别是在阈值分割后的掩膜里，经常会出现一些零散亮点或局部缺口，如果直接做 Canny，边缘会非常碎，霍夫检测会得到很多短线段甚至杂乱直线；形态学相当于先把候选区域整理成更像“线”的形状，为后续边缘与直线检测提供更干净的输入。这让我对“中间结果要可控、可解释”有了更强的意识：每一步做完都应该能说清楚它改变了什么、为什么这样更利于下一步。

在 Canny 边缘检测部分，我对“阈值的含义”理解更扎实了。以前只知道 Canny 需要两个阈值，但这次调参时能明显观察到：阈值偏低时边缘会变多，虽然车道线更容易被保留，但噪声边缘也会大量出现；阈值偏高时边缘更干净，但车道线在远处或较暗区域可能变得断断续续。更重要的是，我意识到边缘检测不

是目标本身，它只是把连续区域转成离散边界点的手段，边缘的“数量”并不等于“质量”。只有当边缘点集中反映目标轮廓、并且背景边缘得到抑制时，后面的霍夫投票才会稳定。因此，边缘检测效果好不好，要结合整体任务来看，而不是只看边缘图“是不是很清楚”。

ROI 的设置给我的感受也很强烈。实验中我能明显看到：当不做 ROI 或 ROI 过大时，画面中许多与车道线无关的结构（路缘、护栏、建筑边缘）都会产生直线，导致霍夫检测结果非常混乱；而 ROI 合理时，检测线段明显更集中、更贴近道路区域。这让我认识到 ROI 的本质是把“全局搜索”变成“局部搜索”，利用几何常识限制算法关注的区域，既能减少误检，也能提升效率。并且 ROI 的形状、顶点位置不是随意的，它实际上反映了摄像头视角下道路的透视结构：上边界通常靠近消失方向，下边界覆盖车辆前方区域。通过 ROI 的调整，我更能理解“场景先验”在传统视觉方法里占很大比重，很多时候不是算法不会，而是搜索范围不合理。

霍夫检测部分让我把“参数空间投票”这一概念真正落到了可观察的结果上。以前学到霍夫变换时，知道边缘点会在参数空间形成峰值，但这次结合检测输出，我更清楚它为什么适合做直线检测：它不是去拟合每一段边缘，而是通过投票把“分散的局部证据”整合成“全局一致的直线假设”。尤其是使用概率霍夫变换输出线段后，我能直观看到参数对结果的影响：阈值决定了投票强度要求，minLineLength 过滤掉短而不稳定的线段，maxLineGap 决定了断裂边缘能否被合并成一条更长的线段。这让我体会到机器视觉算法的“可控性”：每个参数对应一种偏好和假设，调参过程其实是在表达我们希望算法更相信什么、更忽略什么。

整体做完这次实验，我最大的变化是：看待机器视觉问题不再只盯着最终输出，而是更关注每个阶段的中间表示与信息流动。以前容易把实验当成“跑通代码、得到结果图”，但这次我逐步观察了 mask、形态学结果、edges、roi_edges、最终线段叠加图，发现一旦最终效果不好，往往能从上游某一步找到原因并定位问题。这种“分阶段诊断”的习惯对今后做任何视觉任务都很重要，因为它让实验不只是结果导向，而是过程可解释、可复现。此外，我也更理解传统视觉方法的优势：实现简单、计算开销低、每一步都有明确意义；同时也感受到它对场景

依赖较强，需要合理的先验约束与参数配置。总体来说，本次实验加深了我对机器视觉经典方法的理解，也提升了我用工程化思路分析问题、观察现象、解释结果的能力。