

AI 辅助调试 (Debug) 三部曲

第 1 步

提供 Traceback

第 1 步：提供 Traceback

完整复制报错信息：

- 错误类型
- 错误位置（行号）
- 完整的堆栈跟踪

示例：

```
cv2.error: ...
```

```
Invalid number of  
channels
```

第 2 步

说明环境

第 2 步：说明环境

提供运行环境信息：

- 操作系统
- Python 版本
- OpenCV 版本
- 其他相关库版本

示例：

Windows 11

Python 3.9.7

OpenCV 4.8.0

第 3 步

贴出数据

第 3 步：贴出输入数据

提供输入数据信息：

- 数据类型和格式
- 数据的维度/大小
- 期望的输出格式
- 实际的输出（如果有错误）

示例：

图像尺寸：1920x1080

通道： 3 (RGB)



调试 Prompt 模板

完整的调试 Prompt 示例

我的 OpenCV 代码报错了，请帮我分析：

错误信息：

cv2.error: OpenCV(4.8.0) ...

Invalid number of channels in input image

运行环境：

Windows 11, Python 3.9.7, OpenCV 4.8.0

相关代码：

```
img = cv2.imread('test.jpg')
```

```
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
```

输入数据：

图像文件 test.jpg, 尺寸 1920x1080

重构：把 AI 当成导师

重构的意义

代码能运行只是第一步，优雅的代码才能长期维护。

重构 Prompt 模板：

重构请求

请帮我重构这段代码，提高代码质量：

[粘贴代码]

要求：

- ① 提高运行效率
- ② 添加类型提示
- ③ 改进命名规范

AI 可能的重构建议：

① 性能优化：

- 向量化运算替代循环
- 避免重复计算
- 使用生成器替代列表

② 可读性提升：

- 变量名更有意义
- 函数拆分更合理
- 添加类型提示



安全与伦理：红线警告

重要提醒

使用 AI 辅助编程时，必须注意以下安全与伦理问题！

1. 数据安全

- 不要上传：
 - API Key、密码
 - 个人隐私数据
 - 商业机密代码
 - 学生个人信息

- 安全做法：

- 使用脱敏示例数据

本节课更多精彩内容

2. 版权与责任

- 代码归属：

- AI 生成代码的版权复杂
- 检查所用工具的许可协议
- 商业项目需谨慎

- 责任界定：

- AI 生成的代码可能有错误
- 最终责任人是你自己
- 务必测试验证

3. 学术诚信

