

## 还记得这些痛点吗？

- ▶ cv2.imread() 返回 None
- ▶ 图像显示颜色异常
- ▶ 数组越界、维度不匹配
- ▶ API 参数太多，记不住

## 典型错误代码

```
[language=Python, basicstyle=]  
死循环! while img is not None:  
    cv2.imshow('image', img) if cv2.waitKey(1) ==  
    'q': break
```

## 传统解决方式：

1. 翻文档 ([docs.opencv.org](https://docs.opencv.org))
2. 搜索 StackOverflow
3. 问同学/老师
4. 试错（耗费大量时间）

## AI 辅助的新方式

- ▶ 直接问 AI
- ▶ 解释错误原因
- ▶ 给出修改建议
- ▶ **降低学习门槛！**

大模型写代码的本质

# Token 预测

+

# 模式匹配

为什么 AI 擅长语法但不一定懂逻辑？

- ▶ **擅长：**基于海量代码库的模式复刻
- ▶ **不擅长：**理解你的具体业务逻辑

# 幻觉 (Hallucination) 专题

## 什么是 AI 幻觉?

AI 自信满满地生成看似合理但实际上错误或不存在的信息。

## 案例：AI 发明了一个不存在的 OpenCV 函数

### AI 的“创意”代码

```
[language=Python, basicstyle=] import cv2  
img = cv2.imread('exam.jpg')  
  
不存在的函数! fixed = cv2.auto_fix_exposure(img)
```

## 如何识别幻觉?

- ▶ 查官方文档 ([docs.opencv.org](https://docs.opencv.org))
- ▶ 在 Python 中 `dir(cv2)` 查看所有可用函数
- ▶ 多问一句：“这个函数真的存在吗？”

# 编程范式的演进

[draw, fill=blue!20] (v1) at (0,0) 1.0 阶段  
查阅文档; [draw, fill=green!20] (v2) at (4,0) 2.0 阶段  
AI 辅助; [draw, fill=orange!20] (v3) at (8,0) 3.0 阶段  
AI 原生? ;  
[->, thick] (v1) – (v2); [->, thick] (v2) – (v3);

核心结论：程序员的核心竞争力转变

从“记忆语法”转向“问题定义”与“代码审查”