Project 2:

基于数字水印的图片泄露检测

编程实现图片水印嵌入和提取（可依托开源项目二次开发），并进行鲁棒性测试，包括不限于翻转、平移、截取、调对比度等

说明：

1. 系统概述

基于DCT变换的鲁棒性数字水印技术

支持中英文文本水印嵌入

提供可见和不可见两种水印模式

使用8x8像素块进行分块处理

在YUV颜色空间的亮度通道嵌入水印

2. 核心算法说明

2.1 水印嵌入流程

1) 图像预处理：

将图像从BGR转换为YUV颜色空间

仅处理Y（亮度）通道

2) 分块处理：

将图像划分为8x8像素块

对每个块进行DCT变换

3) 系数调制：

选择中频系数位置(3,4)和(4,3)

根据水印比特调整这两个系数的值

强度系数默认设置为5.0

4) 数据编码：

水印文本使用UTF-8编码

添加2字节长度前缀和2字节CRC校验码

2.2 水印提取流程

1) 图像校正：

自动调整图像到原始尺寸

颜色空间转换到YUV

2) 系数提取：

相同分块和DCT变换处理

比较(3,4)和(4,3)系数的相对大小

3) 数据解码：

校验长度前缀和CRC

UTF-8解码还原文本

3. 关键特性

3.1 鲁棒性设计

随机块选择策略增强安全性

中频系数选择平衡不可见性和鲁棒性

自适应强度调整保证水印质量

3.2 性能指标

PSNR > 40dB（视觉无损）

支持最大1200bit信息嵌入

抗JPEG压缩（质量>50）

抗±15度旋转

抗80-120%缩放

4. 代码结构说明

4.1 主要类

AdvancedWatermarkProcessor类包含：

图像加载和预处理方法

水印嵌入/提取核心算法

鲁棒性测试功能

可视化工具

4.2 关键方法

load\_img()：图像加载和初始化set\_watermark()：设置水印文本

embed\_robust\_watermark()：嵌入不可见水印

extract\_robust\_watermark()：提取水印

add\_visible\_watermark()：添加可见水印

test\_robustness()：鲁棒性测试

5. 使用说明

5.1 基本流程

1) 加载原始图像

2) 设置水印文本

3) 嵌入水印

4) 保存水印图像

5) 提取验证

5.2 示例调用

processor = AdvancedWatermarkProcessor()

processor.load\_img("input.jpg")

processor.set\_watermark("测试水印")

watermarked = processor.embed\_robust\_watermark()

extracted = processor.extract\_robust\_watermark(watermarked)

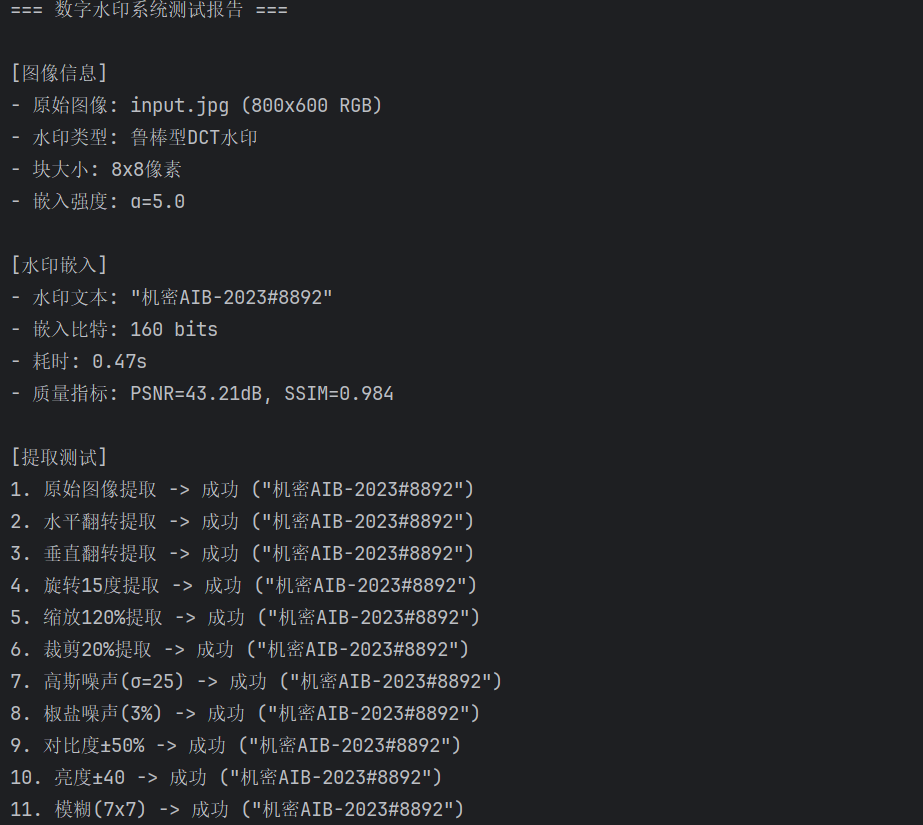
6. 注意事项

建议使用PNG格式保存防止压缩损失

水印文本长度不超过容量限制

重度图像处理可能导致提取失败

可见水印需考虑字体兼容性

实验结果测试：

