

# README

## 实验报告：图片水印嵌入与提取及鲁棒性测试

### 一、实验目的

- 实现图片中嵌入水印信息（文字型水印）；
- 能够从嵌入水印的图片中正确提取出水印；
- 对加水印后的图像进行常见图像扰动操作，并测试水印的鲁棒性；
- 为后续图像版权保护、内容溯源等应用提供技术基础。

### 二、实验原理

#### 1. 水印嵌入方式

本实验采用**空域水印嵌入方法**，即直接在图像像素层面叠加水印图层。

实现方式如下：

- 使用 `cv2.putText()` 将文字绘制为水印图像；
- 使用 `cv2.addWeighted()` 对原图与水印图进行加权混合，得到叠加了水印的图像；
- 水印在视觉上明显，但在像素值上只是小幅修改。

#### 2. 水印提取方式

采用原图与水印图之间的**差异图提取 (diff)** 方式：

- 用 `cv2.absdiff(watermarked, original)` 提取两图的像素差；
- 得到的图像中，差异区域即为嵌入的水印；
- 再通过二值化增强提取结果的对比度。

#### 3. 鲁棒性测试方法

对加水印图像执行常见扰动操作，包括：

操作类型	操作说明
水平翻转	<code>cv2.flip()</code>
平移	<code>cv2.warpAffine()</code> 实现图像平移
裁剪	中心区域裁剪（1/4区域）
调整对比度	使用 <code>cv2.convertScaleAbs()</code> 增强对比度
原图无扰动	控制变量，用于对比提取效果

### 三、实验环境

---

- 开发语言：Python 3.x
- 依赖库：OpenCV (cv2)、NumPy、Pillow
- 测试图片：



- 操作系统：Windows / macOS / Linux 兼容

### 四、实验过程与核心代码

---

#### 1. 文字水印嵌入函数

```
def embed_watermark(image, watermark_text):  
    watermark = np.zeros_like(image)  
    cv2.putText(watermark, watermark_text, (30, 60), cv2.FONT_HERSHEY_SIMPLEX,  
1.5, (255,255,255), 2)  
    return cv2.addWeighted(image, 1.0, watermark, 0.4, 0)
```

## 2. 提取水印函数（增强后）

```
def extract_watermark(watermarked, original):  
    diff = cv2.absdiff(watermarked, original)  
    gray = cv2.cvtColor(diff, cv2.COLOR_BGR2GRAY)  
    _, binary = cv2.threshold(gray, 30, 255, cv2.THRESH_BINARY)  
    return binary
```

## 3. 鲁棒性扰动操作函数

```
def apply_robustness_tests(image):  
    results = {}  
    results["flipped"] = cv2.flip(image, 1)  
    rows, cols = image.shape[:2]  
    results["translated"] = cv2.warpAffine(image, np.float32([[1, 0, 30], [0, 1,  
30]]), (cols, rows))  
    h, w = image.shape[:2]  
    results["cropped"] = image[h//4:3*h//4, w//4:3*w//4]  
    results["contrast"] = cv2.convertScaleAbs(image, alpha=1.8, beta=20)  
    return results
```

---

## 五、实验结果展示

### 原图与加水印图像

- 原图：无任何嵌入
- 加水印图：左上角出现半透明 "WATERMARK"





提取效果示意图

图像操作	提取图示	结果评估
原图 → 水印图	<code>extracted_watermark.png</code>	水印完整清晰
翻转图 → 提取	<code>extracted_flipped_horizontal.png</code>	可辨识，略有镜像变形 ✓
平移图 → 提取	<code>extracted_translated.png</code>	水印部分缺失，中等 ✓
对比度调整 → 提取	<code>extracted_contrast_adjusted.png</code>	水印变浅，噪点增多 ⚠
裁剪图 → 提取	<code>extracted_cropped.png</code>	基本不可辨识 ✗

结论：空域水印对于裁剪和强对比变化较敏感，翻转和平移影响较小。

## 六、结论与展望

- 成功实现了基于 OpenCV 的文字水印嵌入与提取；
- 通过图像差异计算成功恢复水印内容；
- 实验验证了在部分图像扰动下水印可提取，但空域嵌入在鲁棒性上存在一定局限。

## 展望方向

改进项	说明
使用频域嵌入	如 DCT、DWT 可提升对压缩、对比度变化、裁剪等的鲁棒性
加密水印	对水印文字加密后嵌入，提高安全性
图像水印	支持 logo 或二维码水印嵌入
多区域冗余嵌入	提高鲁棒性（对抗裁剪）