

project2 实验说明

网安一班-202200460017-张睿琰

1、实验思路

大体实现流程：

预处理——>水印嵌入——>水印提取——>质量评估——>
鲁棒性测试

2、实验原理

采用频域嵌入方法实现，一些关键注意点：

2.1 DCT 变换：将图像分块之后再进行离散余弦变换。

2.2 中频系数嵌入：选择中频位置嵌入水印，平衡不可见性和鲁棒性。

2.3 嵌入规则：修改选定 DCT 系数值来代表水印信息。

3、实验具体实现的一些说明

3.1. DCTWatermark 类

```
__init__(self, strength=25, block_size=8, pattern=[(4,1), (3,2)])
```

用于初始化水印参数，包含的参数有：

strength: 水印嵌入强度

block_size: DCT 块大小

pattern: 用于嵌入水印的 DCT 系数位置

```
_get_dct_blocks(self, img)
```

将图像分割为块并进行 DCT 变换,注意要保证图片大小是块大小的整数倍。之后返回包含块位置和 DCT 系数的列表。

```
_merge_blocks(self, blocks, img_shape)
```

用于从 DCT 块重建图像。

```
embed(self, host_img, watermark)
```

将水印嵌入图像, 加入一些 `host_img`、`watermark` 等参数。其中要注意, 调整水印大小, 并将水印值调整到一定的范围之内。

```
extract(self, watermarked_img, watermark_shape)
```

使用此函数从图像中提取水印。

```
evaluate_quality(self, original, watermarked)
```

可以评估图像质量, 最后返回 PSNR 和 SSIM 值。

3.2 RobustnessTester 类 - 鲁棒性测试

在这个类中定义一些可以对有水印图像进行各种攻击并测试水印提取效果的功能, 满足我们测试的需要。

`apply_rotation(self, img, angle)` 旋转攻击

`apply_translation(self, img, dx, dy)` 平移攻击

`apply_cropping(self, img, crop_percent)` 裁剪攻击

`apply_contrast(self, img, alpha)` 对比度调整攻击

`apply_brightness(self, img, beta)` 亮度调整攻击

`apply_gaussian_noise(self, img, mean=0, sigma=25)` 高斯噪声攻击

`apply_resizing(self, img, scale_factor)` 缩放攻击

`apply_blur(self, img, kernel_size=5)` 模糊攻击

`calculate_nc(self, original_wm, extracted_wm)` 计算归一化相关系数(NC)

`test_robustness(self, watermarked_img, original_wm,
watermark_extractor)`

可以执行完整的鲁棒性测试，返回包含攻击结果和 NC 值的字典。

3.3 辅助函数

`plot_results(original_img, watermarked_img, original_wm,
extracted_wm, robustness_results)`

对水印镶嵌结果进行一些可视化分析，可视化所有结果，并且得到两个结果图像文件。