

简介

文件描述：my_watermark.py 和 README.md.md。前者是程序代码文件，后者则给出了程序的使用示例。

程序作用和功能：本程序提供了一种对图像进行水印嵌入和提取的方法，旨在保护数字图像的版权和完整性。程序实现了图像的分块 FFT、加嵌水印、分块逆 FFT 等核心功能，并通过命令行参数控制随机种子、小方块的边长和嵌入水印时的参数，具有一定的灵活性和扩展性。

语言和平台要求：本程序为 Python 程序，需要在已经安装了 Python3 和以下库的环境下运行：numpy, opencv-python, matplotlib。

版本号和修改记录：1.0.0 (2023/06/08) 第一版发布

作者和联系方式：李铭辉，2011266@mail.nankai.edu.cn。

主要贡献

这段代码的主要贡献在于提供了一个 Python 程序，实现了图像水印的嵌入和提取功能，方便用户进行水印保护和提取。此外，该程序还利用了分块 FFT 和分块逆 FFT 的技术对图像进行处理，使得程序的水印嵌入和提取过程更加高效。同时，通过命令行参数控制随机种子、分块大小和嵌入水印时的参数，使得该程序具有一定的灵活性和扩展性。

关键代码分析

关键的代码过程包含两个部分：

第一部分是嵌入水印。

1. 先后使用 cv2 模块的函数 imread、cvtColor 来读取图像、转换颜色空间（目标类型：YUV）；
2. 利用指定的随机种子，使用 numpy 模块的函数 randn 生成两个随机数组（辅助水印嵌入，起到类似编码的作用）；
3. 使用函数 fft_block 对图像的 U 层进行分块 fft；
4. 使用函数 embed 对分块 fft 的结果嵌入水印；
5. 使用函数 ifft_block 将嵌入水印的结果分块逆 fft 成完整 U 层；

6. 将 U 层与原来的 Y、V 层合并，使用 cv2 模块的函数 cvtColor 来转换颜色空间（目标类型：RGB）；

第二部分是水印提取。

1. 先后使用 cv2 模块的函数 imread、cvtColor 来读取图像、转换颜色空间（目标类型：YUV）；
2. 使用 numpy 模块的函数 randn 生成两个随机数组（指定 seed）；
3. 使用函数 fft_block 对图像的 U 层进行分块 fft；
4. 使用函数 get_watermark 从 U 层中提取水印；

关键函数如下：

1. fft_block(image): 将图像分块，并进行 FFT 变换。

输入：图像 image；

输出：分块 FFT 变换后的结果。

代码设计：通过调用 numpy 模块的函数 vsplit、hsplit，可以很方便地完成图像的二维划分；通过调用 numpy 模块的函数 fft2，对每个小块做二维快速傅里叶变换。

2. embed(blocks, watermark): 将水印嵌入到 FFT 变换后的结果中。

输入：FFT 变换后的小块数组 blocks 和水印图像 watermark；

输出：嵌入水印后的 FFT 变换结果。

代码设计：（这里到时候画一张图可能会比较好理解）由于小块的数目与水印的大小完全一致（前面有相关的调整以确保这一点），所以将水印的每一个像素信息嵌入到对应位置的小块中。因为水印是二值的，如果该像素点是 1，那就把第一个随机数组按照指定位置添加到小块中；如果是 0，则把第二个随机数组按照指定位置添加到小块中。

3. ifft_block(blocks): 对 FFT 变换后的各小块进行逆 FFT 变换，合并各小块，得到图像。

输入：FFT 变换后的小块数组 blocks；

输出：逆变换后得到的图像。

代码设计：通过调用 numpy 模块的函数 ifft2，对每个小块做二维快速傅里叶逆变换。通过调用 numpy 模块的函数 vstack、hstack，二维划分重新拼接成完整图像。

4. get_watermark(embed_U_image, watermark_size): 从 U 层中提取水印。

输入：图像的 U 层 embed_U_image 和原水印大小 watermark_size；

输出：提取出的水印。

代码设计：首先调用函数 fft_block。然后在每个小块的指定位置处，提取数组，检查它与两个随机数组中的哪一个更像。如果它与第一个随机数组相关程度更高，则将水印图像对应位置的像素赋值 255，否则赋值为 0。

总结

我编写了简易的 Python 程序，实现了图像的嵌入和提取功能，方便用户进行水印保护和提取。此外，该程序还利用了分块 FFT 和分块逆 FFT 的技术对图像进行处理，使得程序的水印嵌入和提取过程更加高效。同时，通过命令行参数控制随机种子、小方块的边长和嵌入水印时的参数，使得该程序具有一定的灵活性和扩展性。在我的 readme.md 文件上，可以看到

我的程序在对原图改变不大的基础上很好地嵌入了水印（肉眼几乎看不到差别），并且这样的带水印图像可以抵御个别种类的破坏。