

关于数字水印和信息隐写的技术报告

一、 技术背景

随着信息时代的到来，人们对于信息技术的依赖程度呈现出指数式的增长。在信息技术给人们生活带来极大便捷的同时，人们在信息安全领域遇到的威胁也随着信息爆炸而变得越来越严峻。实现具有完整性、保密性、可用性、可控性和不可否认性的信息系统，成为人们日益迫切的需求。

图像是最重要的信息载体之一。一方面，图像本身传递的视觉信息就具有极高的价值，例如照片、艺术作品等；另一方面，利用人类视觉特性和图像信息存储格式的特殊结构，可以将额外的（视觉信息以外的）信息以一种不易察觉的方式“嵌入”图像之中，以实现信息传递的保密性。

对于图像的视觉信息，人们较为关注的是图像的完整性和不可否认性，比如图像的防伪、图像的版权保护、图像的修改检测等等。而对于通过图像传递额外信息，人们关注的重点集中在信息传递的保密性上。

二、 技术介绍

数字水印技术，主要是针对上文所述的第一类信息，即图像的视觉信息。数字水印技术通过一种不易察觉的形式将特殊信息插入图像中，达到诸如版权保护、图像完整性鉴定等目的。而信息隐写技术主要是针对第二类信息，通过图像为载体，达到隐秘传递信息的目的。

数字水印技术的主要要求是：透明性（不易被肉眼察觉差别）、鲁棒性（不易通过某种改动而丢失水印信息）、安全性（信息隐藏在数据中而非文件头等位置）、无歧义性（恢复出的水印能明确表明所有权）和通用性（适用多种图像尺寸、格式等）。如今主流实现方法可以分为空间域水印和频率域水印，前者直接在空间域中对采样点的幅度值做出改变，具有简单、高效的特点，后者通过傅立叶变换等手段对变换域中的系数做出改变，具有较高的鲁棒性，但算法复杂度极高。

信息隐写技术的主要要求是：不可感知性（不易被肉眼察觉差别），高安全性（能抵御攻击者对所隐写信息的破解），大容量，高效。实现方法与数字水印原理接近。而对于鲁棒性的要求相对数字水印较低。

三、 Jupiter 技术简介

本人根据嵩天老师网络信息安全课程大作业要求，使用脚本语言 Python 开发了基于空间域信息嵌入算法的图像信息嵌入和提取程序 Jupiter。该程序有以下功能：

1. 基于空间域的图像水印嵌入和提取

Jupiter 采用空间域图像水印嵌入算法，可以在原始图片中嵌入另一张（二值）图片。嵌入的基本步骤是：

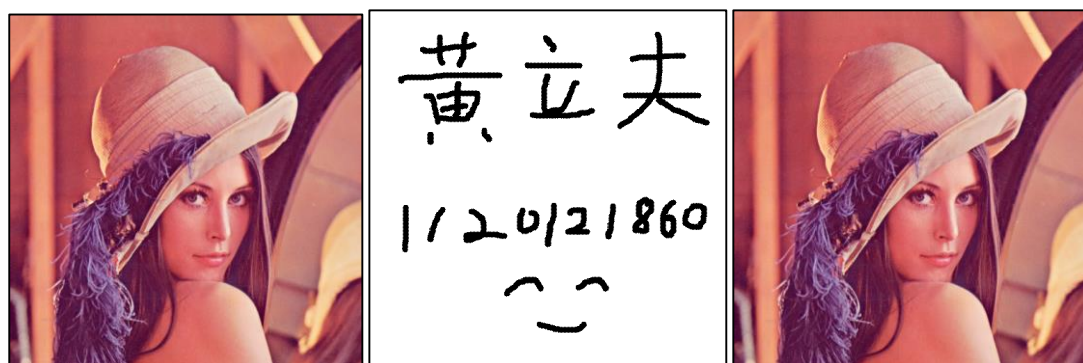
1) 根据原始图像特征对用户给定的水印图像进行预处理

由于算法要求水印图像只能是二值图像，所以用户提供的是 multi-band 图像（如 RGB 图像），或梯度灰度图像，将首先使用 PIL 库转换成二值图像。然后对水印大小等参数进行调整，以满足后续算法需求。

2) 逐像素嵌入

逐像素将预处理后的水印图像的每个点（二值 0 或 255）保存在原始图像第一条 band（如 RGB 图像中的 R）的 LSB（Least Significant Bit）中，分别保存为 0 和 1。

考虑到本次大作业的实验性质，该版本 Jupiter 只使用了编程复杂度和算法复杂度更低但不具有鲁棒性的空间域算法。但实际上通过使用现有第三方库，可以轻易替换成频率域算法。



（示例：上方左图为未加水印的原始图片，中图为用“画图”创建的水印图片文件，经过 Jupiter 的水印嵌入后，得到右图目标图片）

提取水印步骤本质上只是嵌入步骤的逆向过程，可以从右图目标图片中提取出水印图片，和中图完全一致，此处不再赘述。

2. 文本信息的嵌入和提取

1) 文本进行预处理

对文本编码统一为 utf-8，并通过根据用户提供的密钥对文本进行加密（见功能 3）。最后，将加密后的二进制串使用 base64 编码。

2) 将预处理后文本嵌入和相关尺寸信息嵌入图片

将预处理后的文本编码逐位保存在原始图像第二条 band（如 RGB 图像中的 G）的 LSB（Least Significant Bit）中。将诸如编码总长度等元信息以二进制数的形式逐位嵌入到原始图像三条 band（如 RGB 图像中的 B）的 LSB（Least Significant Bit）中。

3. 信息的加密和解密

除了基本的信息传递功能外，为了保证信息的保密性，Jupiter 对于嵌入图片中的文本还提供了密文传递功能。对于用户提供的文本，首先将加密算法（为了简单起见，本版本只使用了自己实现的异或加密，对于商业用途可方便替换成第三方库如 Crypto 提供的 AES 等成熟对称加密算法）。然后将密文再次经过 Base64 编码后存入图像文件中。

四、 Jupiter 使用说明

1. 配置环境：

3) 解释器：Python 3.4 及以上

4) 第三方库：PIL (Python Image Library)，推荐 Pillow 2.0 及以上

（Pillow 安装方法：<http://pillow.readthedocs.org/en/latest/installation.html>）

2. 使用说明：

1) 在控制台界面下启动 Jupiter.py

2) 输入操作命令

>>> image in	——在图片中嵌入用户指定水印
>>> image out	——从图片中提取水印，并将水印文件保存到指定路径中
>>> text in	——在图片中嵌入（加密）隐藏文本信息
>>> text out	——从图片中提取嵌入的隐藏文本信息，并解密显示

3) 根据提示输入相应参数（如原始文件路径名，目标文件路径名，密码等）