



南开大学
Nankai University

南 开 大 学

网络空间安全学院
信息隐藏技术实验报告

实验 9：变换域隐藏法

姓名：2113203 付政烨

年级：2021 级

专业：信息安全、法学

指导教师：李朝晖

2024 年 4 月 30 日

目录

一、 实验内容	1
(一) 实验目的	1
(二) 实验环境	1
(三) 实验要求	1
二、 实验原理	1
(一) 变换域技术	1
(二) 信息隐藏算法	2
(三) 修改系数法	2
三、 实验步骤	2
(一) 图像的预处理	2
(二) 水印嵌入	3
(三) 水印提取	3
(四) 结果展示	4
四、 扩展实验	4
(一) 不同的水印强度对提取效果的影响	4
(二) 增加加密步骤提高水印的安全性	5
五、 实验心得体会	6

一、实验内容

(一) 实验目的

DCT 域的信息隐藏包括：

- 修改系数方法；
- 系数比较方法。

以上两种方法任选一种，实现变换域中的信息隐藏与提取。

(二) 实验环境

- 运行系统：Windows11
- 实验工具：Matlab2022a
- 数据：BMP 格式图像

(三) 实验要求

- 在 MATLAB 中调试完成
- 编写实验代码和报告，并给出截图
- QQ 群提交作业

二、实验原理

(一) 变换域技术

在载体的显著区域隐藏信息，比 LSB 方法能够更好地抵抗攻击，而且保持了对人类感观的不可察觉性。**常用的变换域方法：**离散余弦变换 DCT，离散小波变换 DWT，离散傅里叶变换 DFT

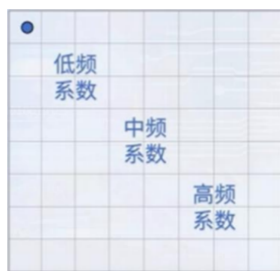


图 1: 二维 DCT 变换

图像压缩标准（JPEG）的核心：二维 DCT 变换。在 DCT 域中的信息隐藏，可以有效地抵抗 JPEG 有损压缩。

二维 DCT 变换：图像分为 8×8 的像素块，进行二维 DCT 变换，得到 8×8 的 DCT 系数。最左上角的系数是直流系数其余是交流系数。左上角部分是直流和低频，右下角部分是高频，中间区域是中频。中低频系数包含了图像的大部分能量，是对人的视觉最重要的部分。

(二) 信息隐藏算法

以一定方式选择一些中频系数，在这些中频系数中叠加秘密信息：

- **算法一**：在选出的中频系数中叠加秘密信息 $x(i, j)' = x(i, j) + a \cdot m_i$
- **算法二**：在选出的中频系数中叠加秘密信息 $x(i, j)' = x(i, j) \cdot (1 + a \cdot m_i)$
- **算法三**：不需要原始载体，直接利用载体中两个特定数的相对大小来代表隐藏的信息。
- **算法四**：算法三的扩展，利用 DCT 中频系数中的三个特定系数的相对关系来对秘密信息进行编码。

【注】如果选定位置的两个系数的下观察太大，则对图像的影响较大。应选择相近的值 (如中频系数)。对图像进行 DCT 变换，利用每一块特定位置的系数关系或系数大小中判断隐藏的信息是“1”“0”还是无效块，这样就可以恢复出秘密信息。

(三) 修改系数法

算法二原理：在选中的中频系数中叠加秘密信息，成比例修改 DCT 系数。

$$x(i, j)' = x(i, j) \cdot (1 + a \cdot m_i)$$

其中，a 是可调参数，控制嵌入强度。与算法一相比，每个系数上嵌入的强度大小会有所不同，和原始系数比例相关。

三、 实验步骤

(一) 图像的预处理

```
1 originalImage = imresize(originalImage, [256, 256]);  
2 watermarkImage = imresize(~watermarkImage, [64, 64]);
```

代码调整了图像的尺寸。原始图像调整到 256x256 像素，水印图像调整到 64x64 像素，并进行了取反操作 \sim ，这是为了更好的视觉效果或根据特定的水印逻辑。

```
1 originalImage = double(originalImage) / 256;  
2 watermarkImage = double(watermarkImage);
```

将图像的像素值转换为双精度浮点数，方便后续处理。原始图像除以 256 是为了进行归一化处理，将像素值限制在 0 到 1 之间。

```
1 imageSize = 256;  
2 blockSize = 4;  
3 numBlocks = imageSize / blockSize;  
4 newImage = zeros(imageSize);
```

定义了图像的尺寸和处理块的大小，并计算了总共的块数。然后初始化一个新图像，用于存放嵌入水印后的结果。

(二) 水印嵌入

水印嵌入的主要目的是将水印信息隐秘地嵌入到原始图像中,本次实验是通过修改图像的离散余弦变换(DCT)系数来实现的。

```
1 for i = 1:numBlocks
2     for j = 1:numBlocks
3         ...
4         blockDCT = dct2(block);
5         modFactor = (1 + 0.01 * (2 * watermarkImage(i, j) - 1));
6         blockDCT(1, 1) = blockDCT(1, 1) * modFactor;
7         blockIDCT = idct2(blockDCT);
8         ...
9     end
10 end
```

1. **图像分块**: 整个图像被分成大小为 4x4 的小块。对于一个 256x256 的图像,这意味着总共有 $64 \times 64 = 4096$ 个这样的块。
2. **应用 DCT**: 对每个块应用二维离散余弦变换(dct2)。DCT 能将图像从空间域(直接的像素值)转换到频率域。在频率域中,图像的大部分能量都集中在变换后系数矩阵的左上角(低频部分),这使得高频部分(右下角)的信息可以用于嵌入水印,而不过分影响图像质量。
3. **修改 DCT 系数**: 修改每个块的 DC 系数(DCT 矩阵的第一个元素,即块的平均亮度)。修改的幅度和水印图像对应位置的像素值相关:如果水印像素为 1,则略微增加 DC 系数;如果为 0,则略微减少。这个修改通过 modFactor 实现,其中 0.01 是一个小的调节因子,用于确保图像的视觉质量不会因修改太显著而受损。
4. **逆 DCT 变换**: 修改完 DCT 系数后,使用逆 DCT(idct2)将处理过的块变换回空间域。这样,每个块都被重新构造包含水印信息的新图像块。

(三) 水印提取

提取水印的过程是嵌入的逆过程,通过比较处理后和未处理的图像来检测水印信息。

```
1 for i = 1:numBlocks
2     for j = 1:numBlocks
3         ...
4         if newImage(xStart, yStart) > originalImage(xStart, yStart)
5             extractedImage(i, j) = 1;
6         else
7             extractedImage(i, j) = 0;
8         ...
9     end
10 end
```

1. **图像分块遍历**: 和嵌入过程相同,按块遍历整个图像。对于每个块,取其左上角的像素(即经过逆 DCT 变换后块的第一个像素)进行比较。

2. **比较像素值**：检查每个块的左上角像素。如果处理后的图像在该位置的像素值大于原始图像的相应值，推断此处的水印像素为 1；否则为 0。这种方法假设水印嵌入会导致相应位置像素值的轻微增加或减少。
3. **构建提取的水印图像**：通过上述比较，为每个块确定水印像素值，从而重建整个水印图像。

(四) 结果展示



图 2: 代码运行结果

可以观察到，在嵌入水印后的图像与原始图像之间几乎无可辨识的变化，表明嵌入过程对图像的视觉质量影响极小，主要体现在色调深度上的极细微差别。通过对原始水印图像与从加水印图像中提取出的水印图像进行比较，结果显示两者完全一致，证明水印的提取过程成功。

四、 扩展实验

(一) 不同的水印强度对提取效果的影响

调整水印嵌入的强度（modFactor 中的系数），探究水印强度对原始图像质量和水印提取效果的影响。进而找到最佳的水印强度，以实现水印的不可见性和抗干扰能力的平衡。



图 3: 提取图像



图 4: 提取图像



图 5: 取图像后反色与原图对比

过修改嵌入过程中调整 DC 系数的参数, 实验测试了一系列水印强度参数, 范围从较小的值 (0.001) 到较大的值 (0.9)。结果显示 (详见图3, 4, 4), 较小的水印强度导致水印在图像中的嵌入不够明显, 而较大的水印强度可能会导致水印变得过于明显, 影响图像的可视质量。通过观察和分析不同强度下的嵌入结果, 可以得出最佳的水印强度范围为 0.005 到 0.1 之间, 这样保证水印不可见性的前提下提高水印的鲁棒性和提取准确性。

(二) 增加加密步骤提高水印的安全性

在嵌入水印之前对水印图像进行加密处理, 增强水印的安全性。水印图像可以通过简单的加密算法 (如 XOR 操作等) 进行加密, 然后再进行嵌入。提取时, 需要对提取出的水印进行同样的解密操作。

dct_extend2.m 代码实现了一种数字图像水印嵌入和提取的过程 (详见源码)。首先, 将原始图像和水印图像读取并调整到合适的尺寸, 然后将它们转换为双精度浮点数。接着, 生成一个随机的加密密钥, 并使用异或操作对水印图像进行加密。在嵌入水印时, 对原始图像进行离散余弦变换 (DCT), 根据加密后的水印图像调整 DCT 系数, 再将其逆变换回空间域。在提取水印时, 通过比较处理后的图像与原始图像的像素值, 提取出水印信息。最后, 再次使用加密密钥对

提取的水印图像进行解密，得到最终的水印信息。整个过程中，加密密钥的引入增强了水印的安全性。

XOR 加密实现的关键代码

```
1 % 定义加密密钥
2 encryptionKey = randi([0, 1], size(watermarkImage));
3 % 加密水印图像
4 encryptedWatermarkImage = xor(watermarkImage, encryptionKey);
5 ...
6 % 解密水印图像
7 decryptedExtractedImage = xor(extractedImage, encryptionKey);
```



图 6: 异或加密后的实验结果

五、 实验心得体会

通过实验，我深入理解了数字图像水印技术的原理和实现方法。在实验步骤中，通过调整图像尺寸、应用离散余弦变换、修改 DCT 系数以及逆变换等步骤，成功地嵌入了水印，并且能够有效地提取出水印信息。通过调整水印嵌入的强度，我发现水印强度对图像质量和提取效果有明显影响。

在扩展实验中，我进一步增加了加密步骤，通过 XOR 加密算法对水印图像进行加密，提高了水印的安全性。实验结果表明，在选择水印强度时需要平衡水印的不可见性和抗干扰能力。过小的水印强度会导致水印不够明显，而过大的水印强度则会影响图像的视觉质量。通过观察和分析不同强度下的嵌入结果，确定了最佳的水印强度范围，从而提高了水印的鲁棒性和提取准确性。在增加加密步骤后，我成功地提高了水印的安全性。加密密钥的引入使得水印图像在嵌入和提取过程中都需要经过解密操作，增加了攻击者破解的难度，从而保护了水印的完整性和安全性。

通过本次实验我不仅学习了数字图像水印技术的原理和实现方法，还深入探讨了水印强度对图像质量和安全性的影响，为今后的数字水印技术研究和应用提供了重要参考。