



南開大學

Nankai University

网络空间安全学院
信息隐藏技术实验报告

变换域隐藏法

姓名：孙璐

学号：2112060

专业：信息安全

2024 年 5 月 6 日

目录

1 实验要求	2
2 实验原理	2
2.1 变换域技术	2
2.2 二维 DCT 变换	2
2.3 JPEG 压缩原理举例	3
2.4 基本隐藏算法	4
2.4.1 算法一	5
2.4.2 算法二	5
2.4.3 算法三	5
2.4.4 算法四	6
3 实验步骤	6
3.1 图像处理	6
3.2 秘密信息嵌入	7
4 实验结果展示	8
5 实验心得体会	9

1 实验要求

实验 9: 变换域隐藏法

内容:

DCT 域的信息隐藏包括: 1. 修改系数方法; 2. 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。

2 实验原理

2.1 变换域技术

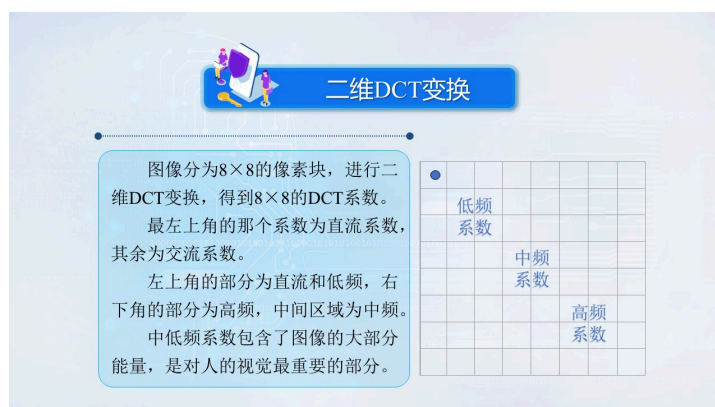
在载体的显著区域隐藏信息, 比 LSB 方法能够更好的抵抗攻击, 而且还保持了对人类感官的不可察觉性

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

图像压缩标准 JPEG 的核心是二维 DCT 变换。在 DCT 域中的信息隐藏, 可以有效的抵抗 JPEG 有损压缩。

2.2 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。

最左上角的那个系数为直流系数, 其余为交流系数。

左上角的部分为直流和低频, 右下角的部分为高频, 中间区域为中频。

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。

DCT 与 Zig-Zag 扫描顺序:

转换 2-D 系数区块成 1-D 系数, 系数按照 Zig-Zag 次序排列。



2.3 JPEG 压缩原理举例

JPEG 压缩原理举例如下：

图像的某个 8×8 图块的亮度值如下：

JPEG压缩原理举例

图像的某个 8×8 图块的亮度值如下：

52	55	61	66	70	61	64	73
63	59	55	90	109	85	69	72
62	59	68	113	144	104	66	73
63	58	71	122	154	106	70	69
67	61	68	104	126	88	68	70
79	65	60	70	77	68	58	75
85	71	64	59	55	61	65	83
87	79	69	68	65	76	78	94

将数值由 0~255 变换到-128~127

将数值由0~255变换到-128~127。

	x								
	-76	-73	-67	-62	-58	-67	-64	-55	
	-65	-69	-73	-38	-19	-43	-59	-56	
	-66	-69	-60	-15	16	-24	-62	-55	
$g =$	-65	-70	-57	-6	26	-22	-58	-59	
	-61	-67	-60	-24	-2	-40	-60	-58	
	-49	-63	-68	-58	-51	-60	-70	-53	
	-43	-57	-64	-69	-73	-67	-63	-45	
	-41	-49	-59	-60	-63	-52	-50	-34	
									y

接下来，根据 DCT 变换公式，计算得到临时结果

接下来，根据DCT变换公式，计算得到临时结果。

$$G = \begin{matrix} & \xrightarrow{u} & \\ \begin{matrix} \downarrow v \\ \end{matrix} & \begin{matrix} -415.38 & -30.19 & -61.20 & 27.24 & 56.12 & -20.10 & -2.39 & 0.46 \\ 4.47 & -21.86 & -60.76 & 10.25 & 13.15 & -7.09 & -8.54 & 4.88 \\ -46.83 & 7.37 & 77.13 & -24.56 & -28.91 & 9.93 & 5.42 & -5.65 \\ -48.53 & 12.07 & 34.10 & -14.76 & -10.24 & 6.30 & 1.83 & 1.95 \\ 12.12 & -6.55 & -13.20 & -3.95 & -1.87 & 1.75 & -2.79 & 3.14 \\ -7.73 & 2.91 & 2.38 & -5.94 & -2.38 & 0.94 & 4.30 & 1.85 \\ -1.03 & 0.18 & 0.42 & -2.42 & -0.88 & -3.02 & 4.12 & -0.66 \\ -0.17 & 0.14 & -1.07 & -4.19 & -1.17 & -0.10 & 0.50 & 1.68 \end{matrix} \end{matrix}$$

JPEG 亮度量化表

亮度量化表

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

JPEG亮度量化表

得到量化结果

获得量化结果：

$$B = \begin{matrix} \begin{matrix} -26 & -3 & -6 & 2 & 2 & -1 & 0 & 0 \\ 0 & -2 & -4 & 1 & 1 & 0 & 0 & 0 \\ -3 & 1 & 5 & -1 & -1 & 0 & 0 & 0 \\ -3 & 1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{matrix} \end{matrix}$$

可见，新的数据很小，而且很多是0。这么多的0，完全可以用游程编码，大大缩小数据量。

Zig-Zag排列：

`[-26,-3,0,-3,-2,-6,2,-4,1,-3, 1,1,5,1,2, -1,1,-1,2,0,0, 0,0,0,-1,-1,0,0,0,0,0,0,0.....]`

2.4 基本隐藏算法

以一定的方式挑选一些中频系数，在这些中频系数中叠加秘密信息。

- 所有中频系数
- 固定位置的中频系数
- 随机挑选中频系数
- 选择最大的几个中频系数

2.4.1 算法一

算法一：在选出的中频系数中叠加秘密信息

$$x'(i, j) = x(i, j) + \alpha m_i$$

$x(i, j)$: DCT 系数

$x'(i, j)$: 隐藏后的 DCT 系数

m_i : 第 i 个秘密信息比特

α : 可调参数, 控制嵌入强度

2.4.2 算法二

算法二：在选出的中频系数中叠加秘密信息

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

方法一：每个系数上嵌入的强度相同

方法二：根据稀疏的大小, 成比例的嵌入

上述的算法一和算法二, 此两算法的提取都需要原始图像

2.4.3 算法三

算法三：不需要原始载体的信息隐藏方法：利用载体中两个特定数的相对大小来代表隐藏的信息

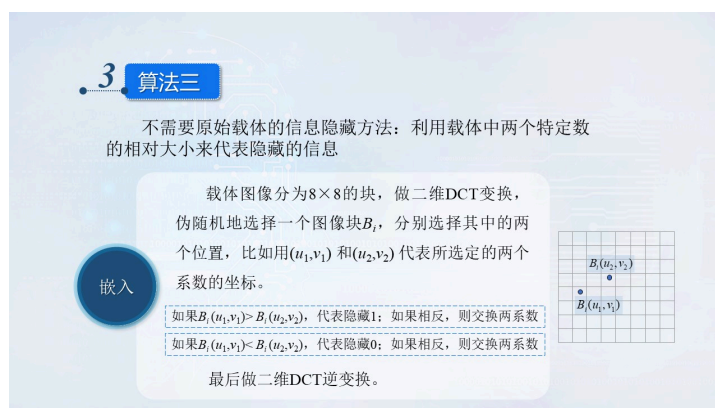
• 嵌入：

载体图像分为 8×8 的块, 做二维 DCT 变换, 伪随机地选择一个图像块 B_i , 分别选择其中的两个位置, 比如用 (u_1, v_1) 和 (u_2, v_2) 代表所选定的两个系数的坐标。

如果 $B_i(u_1, v_1) > B_i(u_2, v_2)$, 代表隐藏 1; 如果相反, 则交换两系数

如果 $B_i(u_1, v_1) < B_i(u_2, v_2)$, 代表隐藏 0; 如果相反, 则交换两系数

最后做二维 DCT 逆变换。



• 提取

接收者进行二维 DCT 变换, 比较每一块中约定位置的 DCT 系数值, 根据其相对大小, 得到隐藏信息的比特串, 从而恢复出秘密信息。

特点：不需原始图像。

注意：如果选定位置的两个系数相差太大，则对图像影响较大。应选择相近的值（如中频系数）

2.4.4 算法四

算法三的扩展：利用 DCT 中频系数中的三个系数之间的相对关系来对秘密信息进行编码

- 嵌入

选择 3 个位置 $(u_1, v_1), (u_2, v_2), (u_3, v_3)$

嵌入 1: 令 $B_i(u_1, v_1) > B_i(u_3, v_3) + D$, $B_i(u_2, v_2) > B_i(u_3, v_3) + D$

嵌入 0: 令 $B_i(u_1, v_1) < B_i(u_3, v_3) - D$, $B_i(u_2, v_2) < B_i(u_3, v_3) - D$

如果数据不符，则修改这三个系数值，使得它们满足上述关系。

其中参数 D 的选择要考虑隐藏的健壮性和不可察觉性之间的平衡，D 越大，隐藏算法对于图像处理就越健壮，但是对图像的改动就越大，越容易引起察觉。

如果需要做的修改太大，则放弃该块，将其标识为“无效”。

“无效”：对这三个系数做小量的修改使得它们满足下面条件之一：

$$B_i(u_1, v_1) \leq B_i(u_3, v_3) \leq B_i(u_2, v_2)$$

或

$$B_i(u_2, v_2) \leq B_i(u_3, v_3) \leq B_i(u_1, v_1)$$

- 提取

对图像进行 DCT 变换，比较每一块相应三个位置的系数，从它们之间的关系，可以判断隐藏的是信息“1”、“0”还是“无效”块，这样就可以恢复秘密信息。

3 实验步骤

3.1 图像处理

读取原始图像文件 Lena 图像和水印图像。水印图像转换成二值图像，原始 Lena 图像大小调整成 256×256 ，水印图像进行反转并将其大小调整成 64×64 。

Listing 1: encode

```

1
2 img = (imread('./raw.bmp'));
3 watermark = imbinarize(imread('./watermark.bmp'));
4 img = imresize(img, [256, 256]);
5 watermark = imresize(~watermark, [64,64]);

```

3.2 秘密信息嵌入

划分块 block 并且在每个块的某一特定位置修改 DCT 次数，嵌入秘密信息，逐块进行扫描。

将水印图像归一化到 0-1 之间，定义图像大小为 256×256 ，每个小块的大小是 4×4 。创建一个大小为 256×256 的全 0 矩阵，用于存储嵌入水印后的图像。

在两个 for 循环里，遍历图像中的每个小块。使用 DCT 变换将其转换成频域，然后根据水印图像的值，对小块的第一个系数进行微调，嵌入水印信息，最后使用 IDCT 变换将小块转换为空余，并将其存储在新的图像中。

Listing 2: encode

```
1
2 img = double(img)/256;
3 watermark = im2double(watermark);
4 size = 256; width = 4;
5
6 blocks = size / width;
7 new_image = zeros(size);
8 vec = ones(64);
9
10 for i = 1 : blocks
11     for j = 1 : blocks
12         x = (i - 1) * width + 1;
13         y = (j - 1) * width + 1;
14         cur = img(x:x+width-1, y:y+width-1);
15         cur = dct2(cur);
16
17         if watermark(i, j) == 0
18             a = -1;
19         else
20             a = 1;
21         end
22
23         cur(1, 1) = cur(1, 1) * (1 + .01 * a) + .01 * a;
24         cur = idct2(cur);
25         new_image(x: x + width - 1, y : y + width - 1) = cur;
26     end
27 end
28
29 for i = 1 : blocks
30     for j = 1 : blocks
31         x = (i - 1) * width + 1;
32         y = (j - 1) * width + 1;
33
```



```
34         if new_image(x, y) > img(x, y)
35             vec(i, j) = 1;
36         else
37             vec(i, j) = 0;
38         end
39     end
40 end
41
42 subplot(231); imshow(img);title("原始图像");
43 subplot(232); imshow(watermark);title("水印图像");
44 subplot(233);
45     imshow(imcomplement(watermark));title("反色之前的水印图像");
46 subplot(234); imshow(new_image,[]);title("嵌入水印");
47 subplot(235); imshow(vec,[]);title("提取图像");
48 subplot(236);
49     imshow(imcomplement(vec),[]);title("提取图像后反色与原图对比");
```

4 实验结果展示

原始 Lena 图像:



水印图像:





5 实验心得体会

本次实验不仅帮助我加深了对变换域隐藏法的理解，还提高了我在 MATLAB 编程方面的实践能力。通过实验，我深刻体会到了理论知识与实践操作的结合对于知识的巩固和应用的重要性，也更加深刻地认识到了信息安全在数字化时代的重要性。