第一部分　加解密

05.　信息隐藏

厚 德 健 行

目录

- 何为信息隐藏
- 信息隐藏模型
- 信息隐藏特点

信息保密技术的发展

古代（密码术和隐写术）

发展至今

- 密码术→现代密码学
- 隐写术→信息隐藏、数字水印、隐通道和匿名通信

隐写术

- 古希腊历史学家希罗多德记载(公元前约440年)
  - 剃光奴隶的头
  - 在头上写消息
  - 头发重新长出盖住消息
  - 奴隶被派出传递消息
  - 剃光头发来看消息（波斯入侵的警告）
- 贯穿整个军事历史，隐写术比密码更经常使用

七律•问缘

我常夜半询姻命，
与月为邻爱晚星。
秋槿含情风后落，
香獐有意谷间鸣。
天街雨过涤新树，
长路云收现旧亭。
地老皆缘蕃草木，
久愁比翼痛风铃。

施耐庵《水浒传》第61回
吴用诱使卢俊义将离合诗

卢花潭上有扁舟，
俊杰黄昏独自游。
义到尽头原有命，
反弓逃难必无忧。

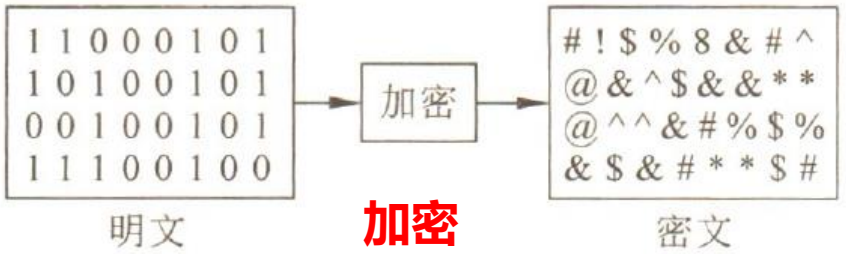
题于墙上，使卢俊义遭官府
迫害逼上梁山

信息隐藏的概念

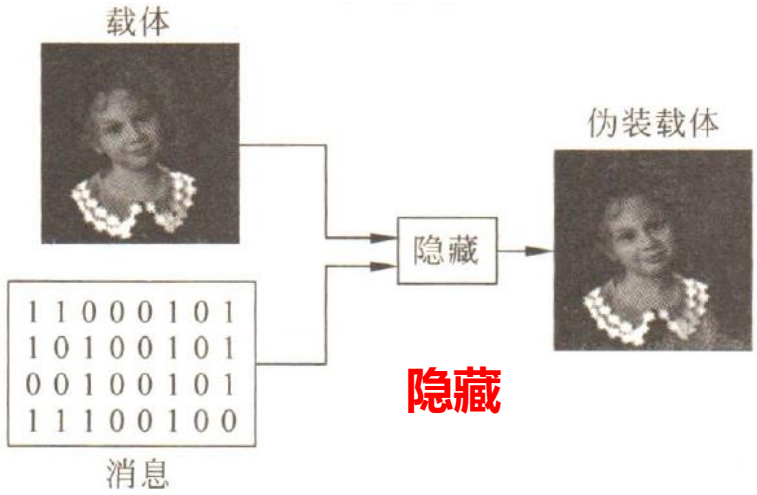
将关键信息秘密地隐藏于一般的载体中（图像、声音、视频或一般的文档），或发行或通过网络传递，达到秘密消息保护的目的。

何为信息隐藏

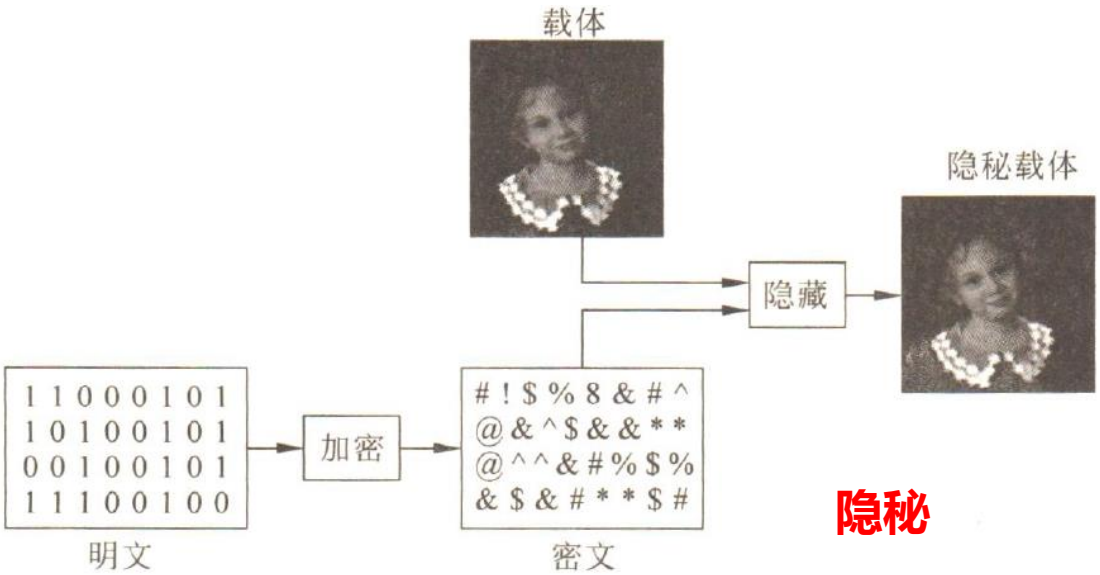
信息隐藏和加密的区别



加密



隐藏



隐秘

- 何为信息隐藏
- 信息隐藏模型
- 信息隐藏特点

信息隐藏的一般模型

秘密信息：被隐藏的信息，如版权信息、秘密数据、软件序列号等

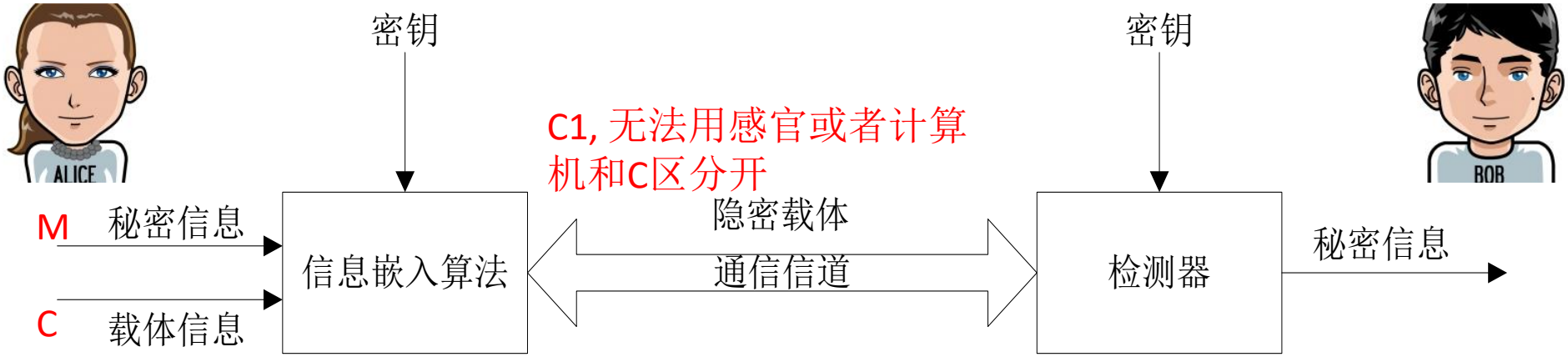
载体信息：公开的信息，如视频、图片、音频等

信息隐藏的一般模型

信息隐藏技术主要由下述两部分组成:

- **隐蔽信息嵌入**算法，利用密钥实现秘密信息隐藏。
- **隐蔽信息检测/提取**算法（检测器），利用密钥从隐蔽载体中检测/恢复出秘密信息。

信息隐藏的一般模型



在密钥未知的前提下，第三者很难从隐秘载体中得到或删除秘密信息，甚至不能发现秘密信息！

基于密钥的信息隐藏分类

1.无密钥的信息隐藏

对一个五元组，

Σ=〈C，M，C^h，D，E〉

其中: C是所有可能载体的集合; M是所有可能秘密消息的集合; C^h是所有可能伪装对象的集合;

E是嵌入函数，$C \times M \rightarrow C^h$;

D是提取函数， $C^h \rightarrow M$。

若满足性质: 对所有$m \in M$和$c \in C$，恒有D(E(c，m))=m，则称该五元组为无密钥信息隐藏系统。

系统的安全性完全依赖于隐藏算法和提取算法的保密性。

信息隐藏模型

基于密钥的信息隐藏分类

2.对称秘钥信息隐藏

对一个六元组，

$\Sigma=\langle C，M，K，C^h，D，E\rangle$

其中: C是所有可能载体的集合; M是所有可能秘密消息的集合; C^h是所有可能伪装对象的集合; K是所有可能密钥的集合;

E是嵌入函数，$C\times M\times K\rightarrow C^h$;

D是提取函数，$C^h\times K\rightarrow M$。

若满足性质: 对所有$m\in M$，$c\in C$和$k\in K$，恒有$D_K(E_K(c，m，k)，k)=m$，则称该六元组为对称秘钥信息隐藏系统。

基于密钥的信息隐藏分类

3.公钥信息隐藏

- 公钥信息隐藏类似于公钥密码
- 公开密钥存储在一个公开的数据库
- 私有密钥由通信各方自己保存
- 公开密钥用于信息的嵌入过程，私有密钥用于信息的提取过程
- 一个公钥信息隐藏系统的安全性完全取决于所选用的公钥密码体制的安全性

- 何为信息隐藏
- 信息隐藏模型
- 信息隐藏特点

■ 鲁棒性（Robustness)

不因图像文件的某种改动而导致隐藏信息丢失的能力，这里所谓“改动”包括传输过程中的信道噪音、滤波操作、重采样、有损编码压缩、D/A或A/D转换等。

■ 不可检测性（Undetectability)

隐蔽载体与原始载体具有一致的特性,如具有一致的统计噪声分布等，以便使非法拦截者无法判断是否有隐蔽信息。

■ 透明性(Invisibility)

利用人类视觉系统或人类听觉系统，经过一系列隐藏处理，使目标数据没有明显的降质现象，而隐藏的数据无法被看见或听见。

■ 安全性（Security）

隐藏算法有较强的抗攻击能力，即它必须能够承受一定程度的人为攻击，而使隐藏信息不会被破坏。

■ 自恢复性(Self-recovery)

由于经过一些操作或变换后，可能会使原图产生较大的破坏，如果只从留下的片段数据，仍能恢复隐藏信号，而且恢复过程不需要宿主信号，这就是所谓的自恢复性。

目录

- 图像的基本表示
- 空间域算法
- 变换（频）域算法

像素(Pixel)

灰度图像——每个像素点仅由灰度值表示

彩色图像——每个像素点由红、绿、蓝三基色组成

灰度图像的信息隐藏

二值图像——每个像素点的灰度值仅取0或1

如果灰度值的取值范围为0~255，每个像素点可用8 bit来表示，则记为(a7，a6，…，a0)，其中ai=0或1(i=0，…，7)。对于每个像素点来说，都取其中的某一位就构成了一幅二值图像。

思考：如何进行信息隐藏？

图像的基本表示

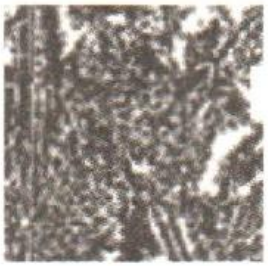
灰度图像

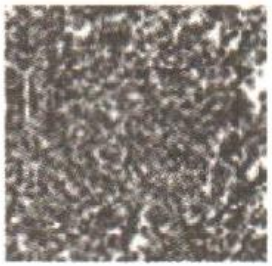

原始图像


位平面7


位平面6

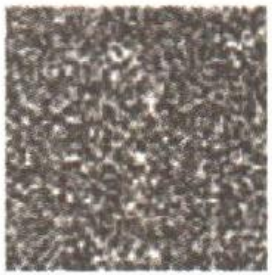

位平面5


位平面4


位平面3


位平面2


位平面1


位平面0

思考：如何进行信息隐藏？

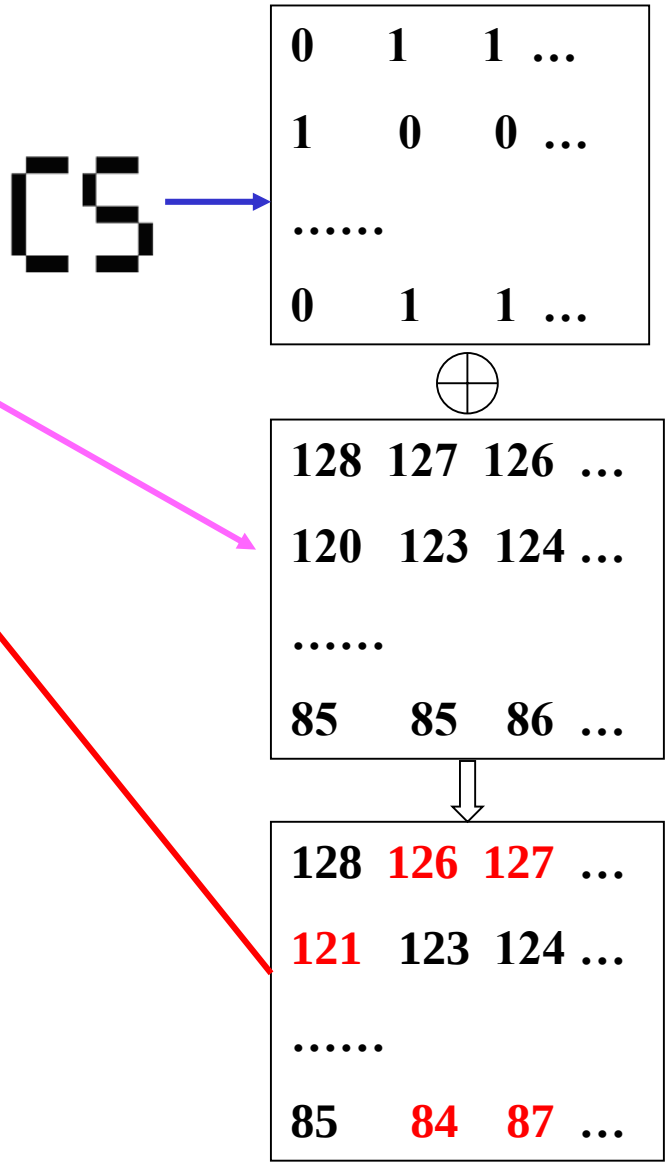
灰度图像的信息隐藏

原始图像

最低位为0

图像的基本表示

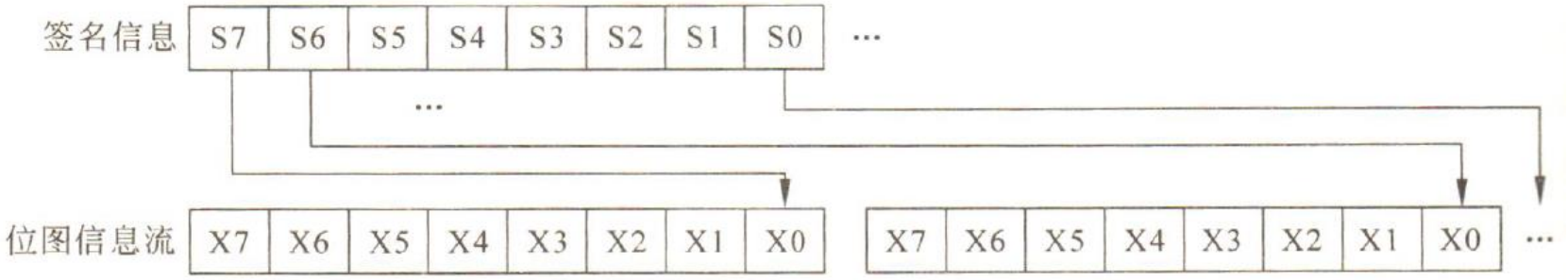
灰度图像的信息隐藏



彩色图像的信息隐藏

- 图像使用24位色: **RGB**
  - 8 位红色, 8 位绿色, 8 位蓝色
- 例如
  - **0x7E 0x52 0x90** is "this color"字体的颜色
  - **0xFE 0x52 0x90** is "this color"字体的颜色
- 然而
  - **0xAB 0x33 0xF0**是"this color"字体的颜色
  - **0xAB 0x33 0xF1** 也是"this color"字体的颜色
- 低位色是无关紧要的

空间域算法

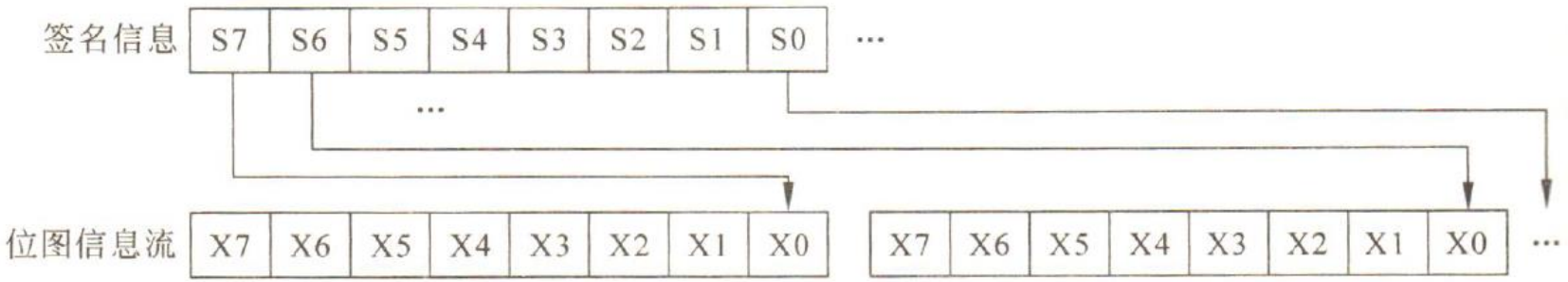


(1) 将待隐藏信息(称为签名信息)的字节长度写入BMP文件标头部分的保留字节中。

(2) 将签名信息转化为二进制数据码流。

(3) 将BMP文件图像数据部分的每个字节的最低位依次替换为上述二进制数码流的一个位。

空间域算法



(a) 未嵌入信息

(b) 嵌入信息后

(c) 被隐藏的信息

对图像数据进行某种变换，这种方法可以嵌入大量的比特而不引起可察觉的降质。

常用方法：

(1) DFT(离散傅里叶变换)

(2) DCT(离散余弦变换)

(3) DWT(离散小波变换)

(a) 原始图像　(b) DFT域　(c) DCT域　(d) DWT域

目录

- 数字水印提出的背景
- 数字水印的概念、特性及分类
- 图像数字水印的几种算法
- 常见的几种数字水印攻击方法
- 数字水印的应用

数字水印提出的背景

- 多媒体信息安全中传统的加解密系统并不能很好地解决版权保护、信息防伪问题。
  - 因为，虽然经过加密后只有被授权持有解密密钥的人才可以存取数据，但是这样就无法向更多的人展示自己的作品；而且数据一旦被解开，就完全置于解密人的控制之下，原创作者没有办法追踪作品的复制和二次传播。
  - 数字水印技术是信息隐藏在多媒体领域的重要应用。
- 数字水印技术已发展成为涉及数学、密码学、通信理论、编码理论、扩频技术、信号处理技术、数据压缩技术、躁声理论和视听觉感知理论等学科的综合技术。

数字水印的概念

- 数字水印（digital watermark）技术，是指在数字化的数据内容中嵌入不明显的记号。
- 被嵌入的记号通常是不可见或不可察的，但是通过一些计算操作可以检测或者提取。
- 水印与源数据（如图象、音频、视频数据）紧密结合并隐藏其中，成为源数据不可分离的一部分，并可以经历一些不破坏源数据使用价值或商用价值的操作而存活下来。

数字水印的概念

1 原始图像　　2 水印　　3 加入水印后的图像

数字水印的特性

- ✧透明性(隐藏性)：是指利用人类视觉系统或人类听觉系统属性，经过一系列隐藏处理 ，使目标数据没有明显的降质现象，在视觉或听觉上具有不可感知性。
- ✧鲁棒性：指不因图象文件的某种改动而导致隐藏信息丢失的能力，这些改动包括传输过程中的信道噪声、滤波、增强、有损压缩、几何变换、D/A或A/D转换等。
- ✧隐藏位置的安全性：指将水印信息藏于目标数据的内容之中，而非文件头等处，防止因格式变换而遭到破坏。
- ✧无歧义性：恢复出的水印或水印判决的结果应该能够确定地表明所有权，不会发生多重所有权的纠纷。
- ✧通用性：好的水印算法适用于多种文件格式和媒体格式。通用性在某种程度上意味着易用性。

数字水印的特性

- 鲁棒水印和脆弱水印。
  - 鲁棒水印要求嵌入的水印能够经受各种常用的编辑处理，主要用于在数字作品中标识著作权信息；
  - 脆弱水印对信号的改动很敏感，根据脆弱水印的状态就可以判断数据是否被篡改过，主要用于完整性保护 。
- 空间域水印和变换（频）域水印：
  - 直接在空间域中对采样点的幅度值作出改变，嵌入水印信息的称为空间域水印；
  - 对变换域中的系数作出改变，嵌入水印信息的称为频率域水印。
- 非盲水印和盲水印。
  - 非盲水印在检测过程中需要原始数据，而盲水印的检测只需要密钥，不需要原始数据。
- 可见水印与不可见水印。

数字水印的特性

盲水印

含水印图像

盲抽取算法
+密钥

抽取的水印

非盲（明文）水印

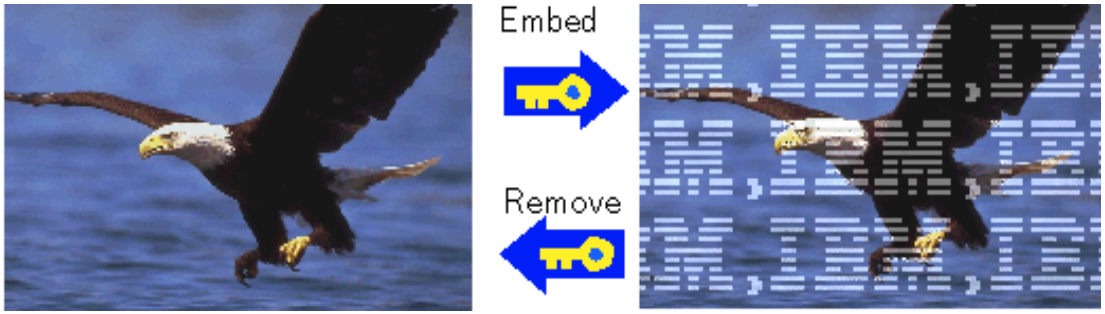
原始图像

含水印图像

抽取的
水印

数字水印的特性

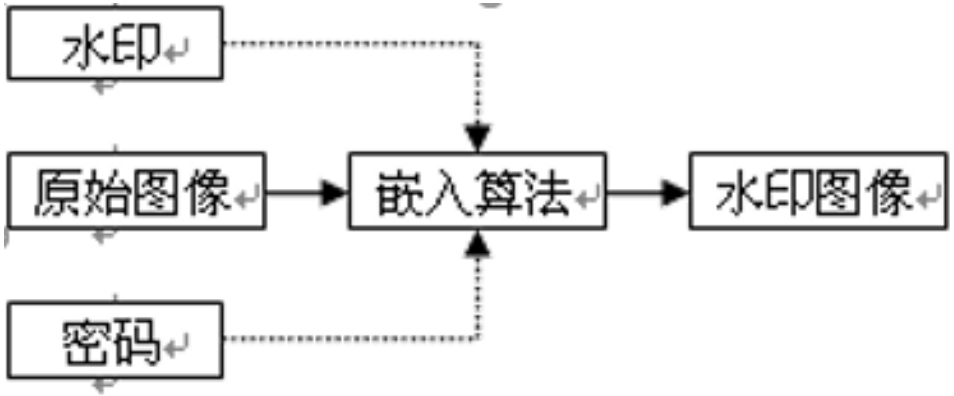
- 可见水印



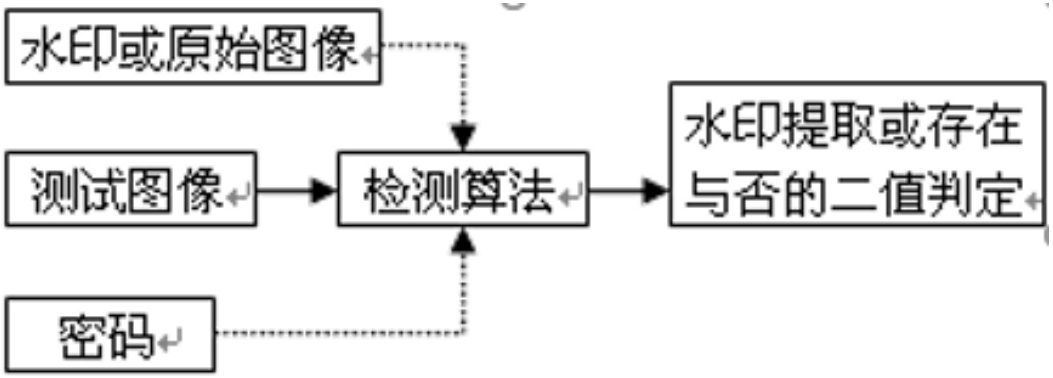
- 不可见水印

嵌入水印

数字水印系统框架



水印信号嵌入模型



水印信号检测模型

- 空间域算法：
  - 典型的算法(LSB,最低有效位算法)是将信息嵌入到随机选择的图像点中最不重要的像素位上，算法的鲁棒性差，水印信息很容易为滤波、图像量化、几何变形的操作破坏。
  - 另一个常用方法是利用像素的统计特征将信息嵌入像素的亮度值中，如Patchwork算法，随机选择N对像素点 (ai，bi)，然后将每个ai点的亮度值加 1，每个bi点的亮度值减 1，这样整个图像的平均亮度保持不变。检测时，计算

$$S = \sum_{i=1}^{n} (\tilde{a}_i - \tilde{b}_i)$$

如果这个载体确实包含了一个水印，就可以预计这个和为2n，否则它将近似为零。

LSB数字水印和健壮性问题

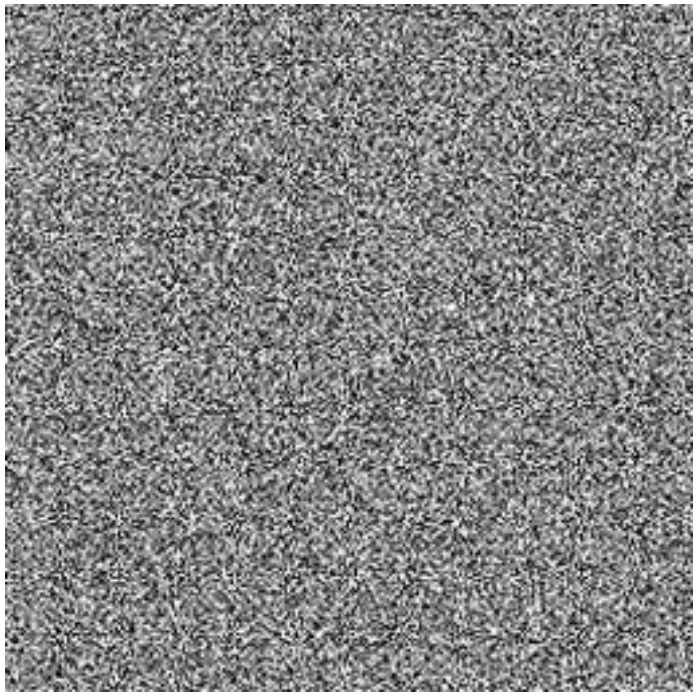
Watermark
(50 x 20 pixels)

厚德健行 取精用弘

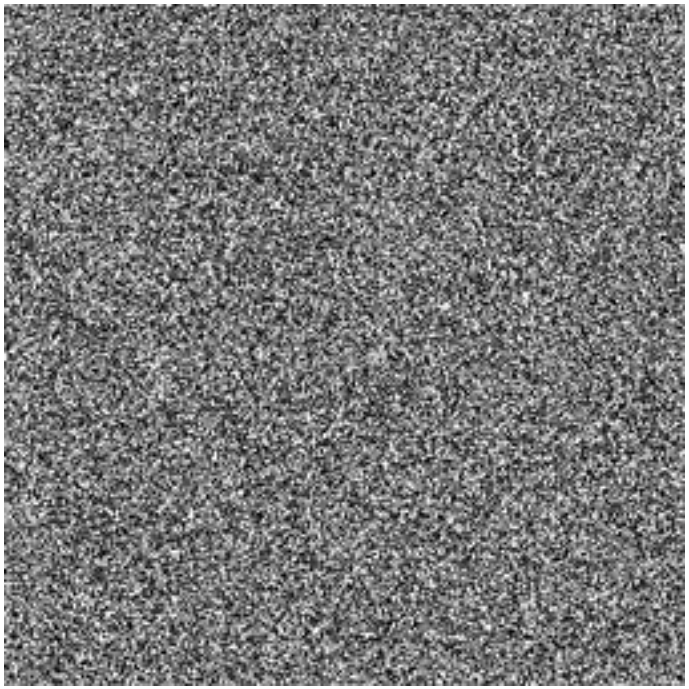
Watermarked Image

Recovered Watermark

LSB数字水印的健壮性很差



Recovered Watermark
after addition of 1% Gaussian Noise



Recovered Watermark
after JPEG Compression with Quality 95

- 变换（频）域算法：
  - 图象的频域空间中可以嵌入大量的比特而不引起可察的降质，当选择改变中频或低频分量（除去直流分量）来加入水印时，强壮性还可大大提高。
  - 频域水印技术可以利用通用的离散余弦变换（DCT），离散小波变换（DWT）和离散傅立叶变换（DFT）等变换方法。

基于DCT变换的水印技术

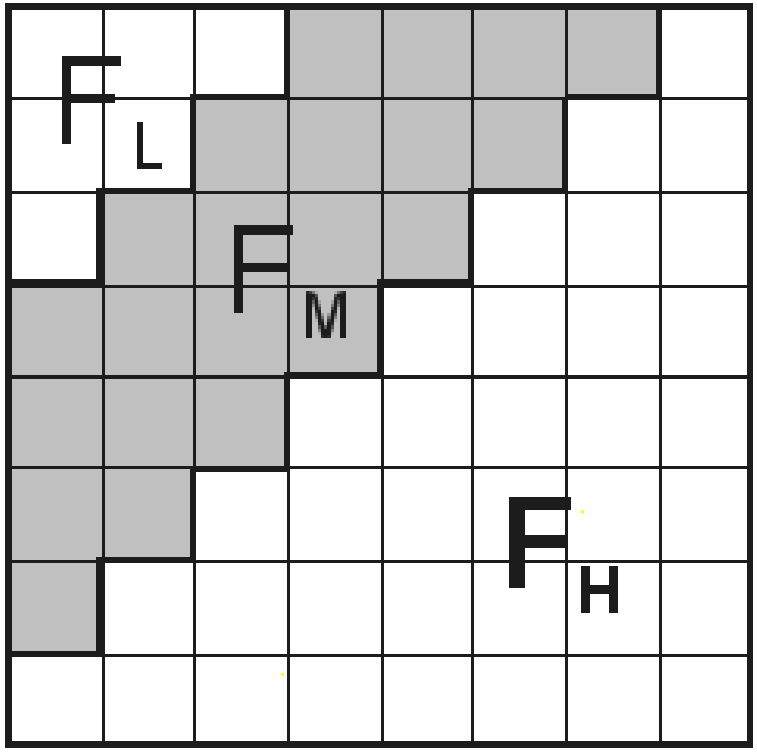
正变换：

$$F(u,v)=\frac{2}{\sqrt{MN}}c(u)c(v)\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\cos\frac{(2x+1)u\pi}{2M}\cos\frac{(2y+1)v\pi}{2N}$$

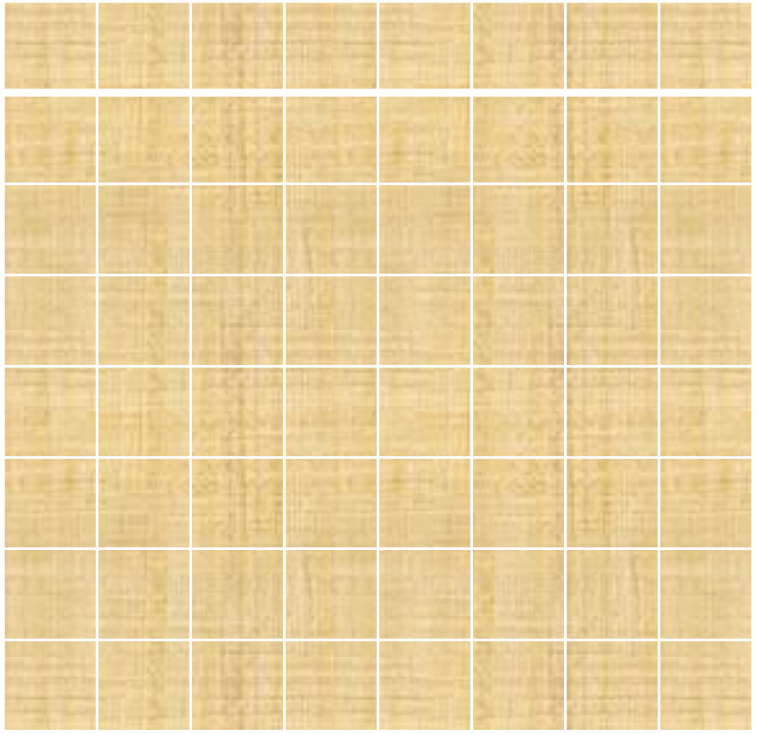
逆变换：

$$f(x,y)=\frac{2}{\sqrt{MN}}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}c(u)c(v)F(u,v)\cos\frac{(2x+1)u\pi}{2M}\cos\frac{(2y+1)v\pi}{2N}$$

其中：$c(x)=\begin{cases}\frac{1}{\sqrt{2}} & x=0\\ 1 & x=1,2,\ldots,N-1\end{cases}$



DCT变换域



基于DCT变换的水印技术

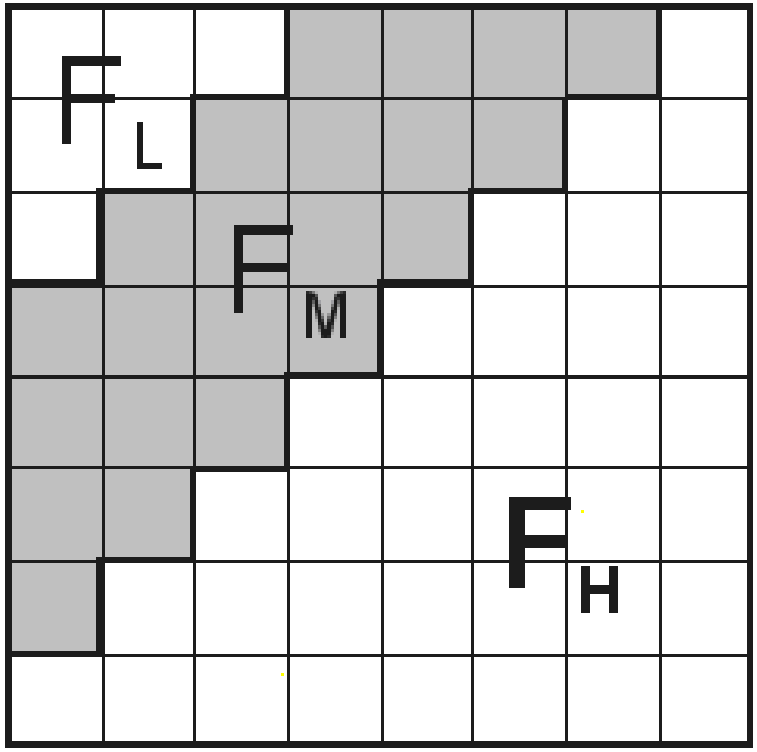
正变换：

$$F(u,v)=\frac{2}{\sqrt{MN}}c(u)c(v)\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\cos\frac{(2x+1)u\pi}{2M}\cos\frac{(2y+1)v\pi}{2N}$$

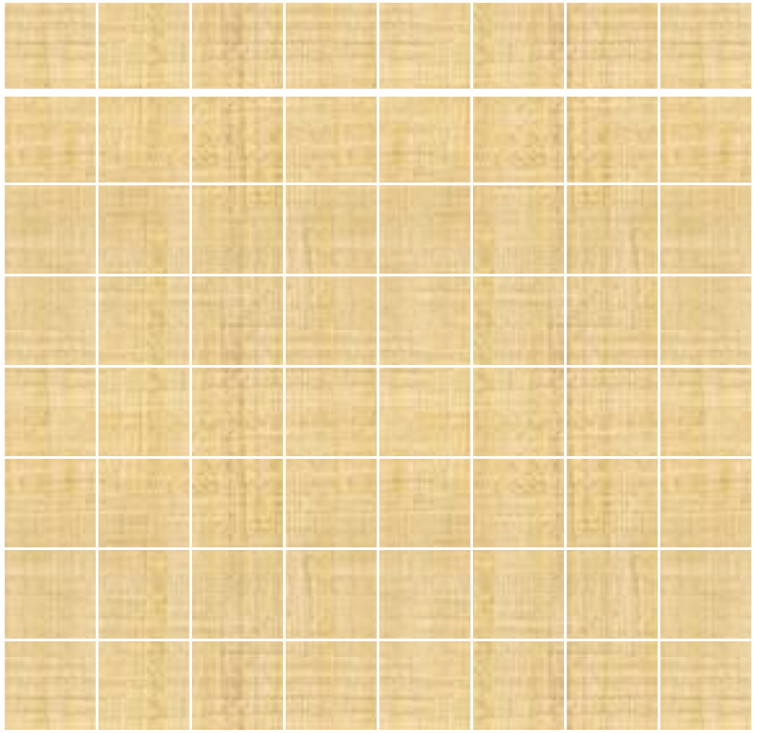
逆变换：

$$f(x,y)=\frac{2}{\sqrt{MN}}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}c(u)c(v)F(u,v)\cos\frac{(2x+1)u\pi}{2M}\cos\frac{(2y+1)v\pi}{2N}$$

其中：$c(x)=\begin{cases}\frac{1}{\sqrt{2}} & x=0\\ 1 & x=1,2,\ldots,N-1\end{cases}$



DCT变换域



基于DCT变换的水印技术

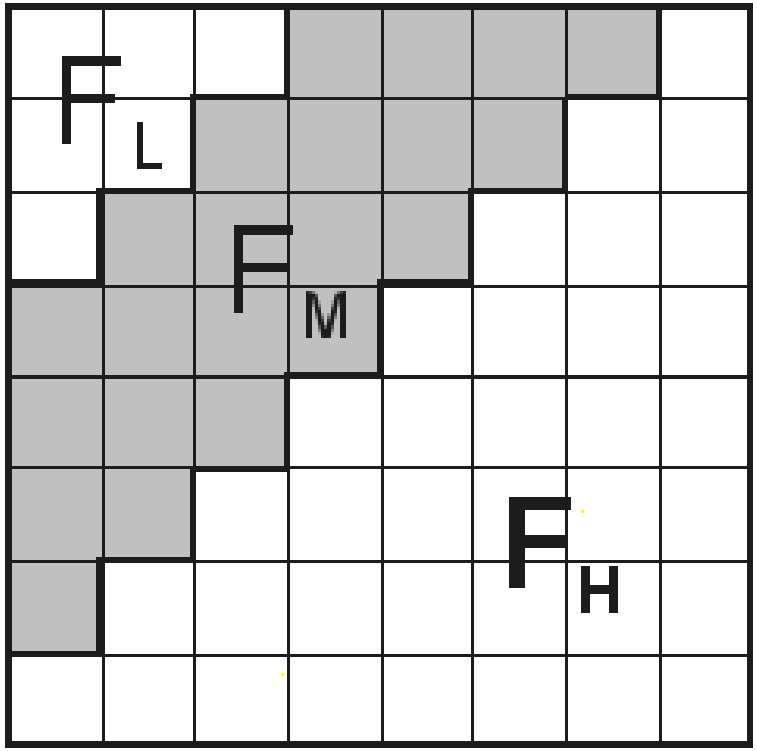
正变换：

$$F(u,v)=\frac{2}{\sqrt{MN}}c(u)c(v)\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\cos\frac{(2x+1)u\pi}{2M}\cos\frac{(2y+1)v\pi}{2N}$$

逆变换：

$$f(x,y)=\frac{2}{\sqrt{MN}}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}c(u)c(v)F(u,v)\cos\frac{(2x+1)u\pi}{2M}\cos\frac{(2y+1)v\pi}{2N}$$

其中：$c(x)=\begin{cases}\frac{1}{\sqrt{2}} & x=0\\ 1 & x=1,2,\ldots,N-1\end{cases}$



DCT变换域

低频：图像强度（亮度/灰度）变换平缓，也就是大片色块的地方

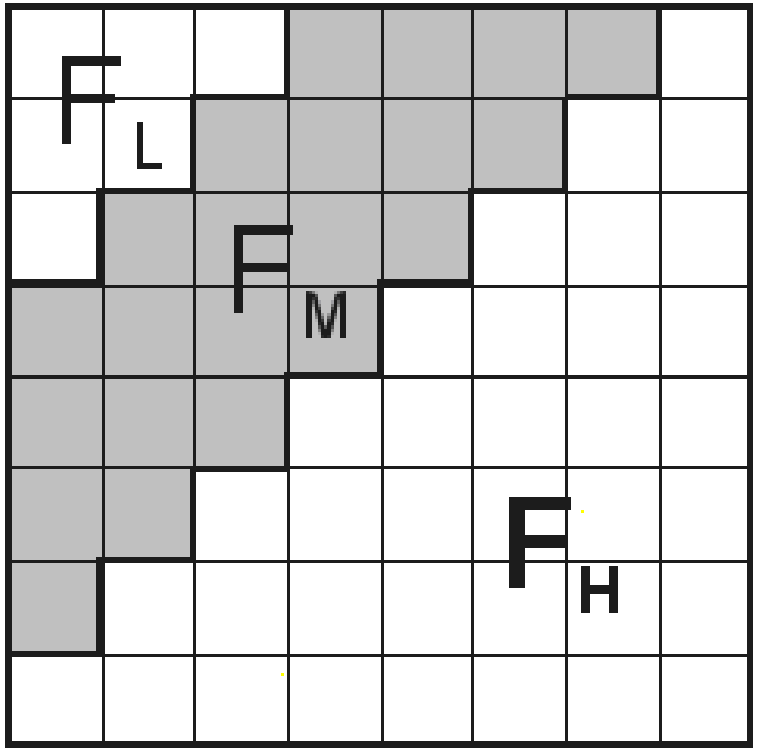
高频：图像强度（亮度/灰度）变换剧烈，也就是我们常说的边缘轮廓，纹理

基于DCT变换的水印技术

正变换：

逆变换：

其中：$c(x)=\begin{cases}\frac{1}{\sqrt{2}} & x=0\\ 1 & x=1,2,\ldots,N-1\end{cases}$



DCT变换域

低频分量（F_L）：主要成分是低频信息，形成了图像的基本灰度等级，对图像结构的决定作用较小

中频分量（F_M）：决定了图像的基本结构，形成了图像的主要边缘结构

高频分量（F_H）：形成了图像的边缘和细节，是在中频信息上对图像内容的进一步强化

思考：水印时选哪种频率部分进行嵌入，为什么？

基于DCT变换的水印技术

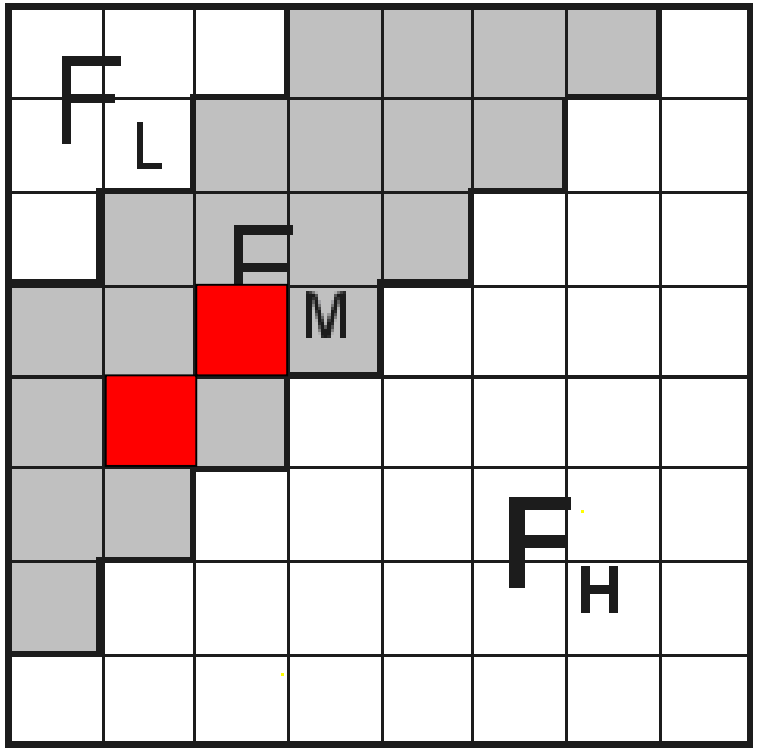
正变换：

$$F(u,v)=\frac{2}{\sqrt{MN}}c(u)c(v)\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\cos\frac{(2x+1)u\pi}{2M}\cos\frac{(2y+1)v\pi}{2N}$$

逆变换：

$$f(x,y)=\frac{2}{\sqrt{MN}}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}c(u)c(v)F(u,v)\cos\frac{(2x+1)u\pi}{2M}\cos\frac{(2y+1)v\pi}{2N}$$

其中：$c(x)=\begin{cases}\frac{1}{\sqrt{2}} & x=0\\ 1 & x=1,2,\ldots,N-1\end{cases}$



DCT变换域

若修改低频部分，也就是大片色块，那就容易看出变化，隐蔽性差

若修改高频部分，也就是边缘轮廓，不容易看出来，但是容易被多数处理高频的图像压缩算法破坏，鲁棒性差

折中的方案是修改中频部分，也就是图像的主要边缘结构

基于DCT变换的水印技术

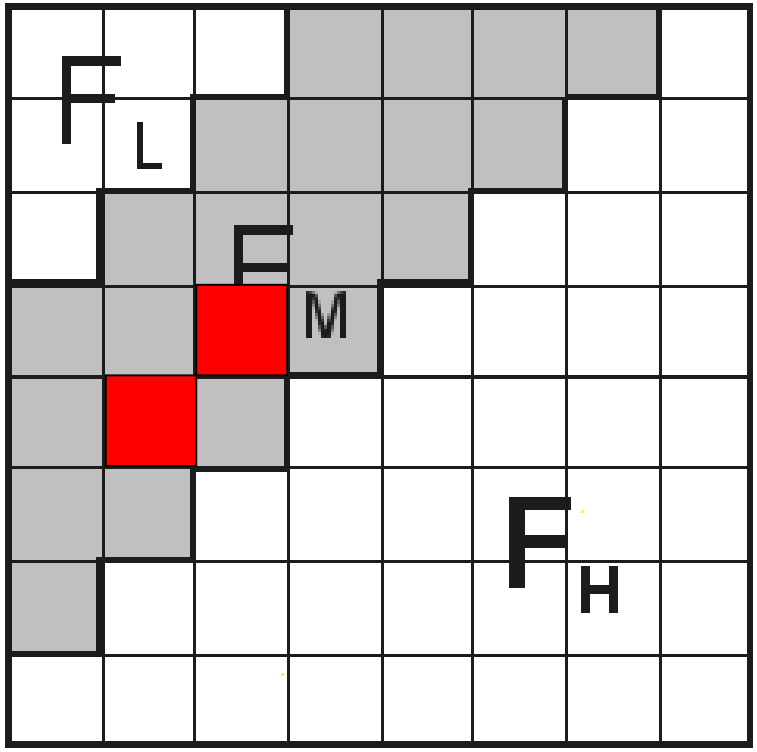
正变换：

$$F(u,v)=\frac{2}{\sqrt{MN}}C(u)C(v)\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\cos\frac{(2x+1)u\pi}{2M}\cos\frac{(2y+1)v\pi}{2N}$$

逆变换：

$$f(x,y)=\frac{2}{\sqrt{MN}}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}C(u)C(v)F(u,v)\cos\frac{(2x+1)u\pi}{2M}\cos\frac{(2y+1)v\pi}{2N}$$

其中：$C(x)=\begin{cases}\frac{1}{\sqrt{2}} & x=0\\ 1 & x=1,2,\ldots,N-1\end{cases}$



DCT变换域

9	11	12	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

像素频率矩阵

基于DCT变换的水印技术

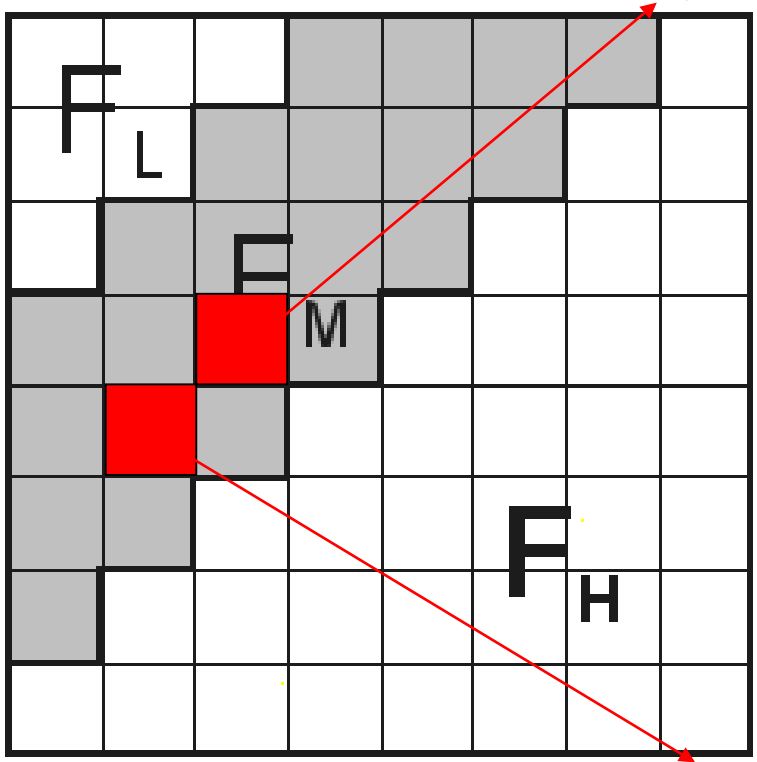
正变换：

$$F(u,v)=\frac{2}{\sqrt{MN}}C(u)C(v)\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\cos\frac{(2x+1)u\pi}{2M}\cos\frac{(2y+1)v\pi}{2N}$$

逆变换：

$$f(x,y)=\frac{2}{\sqrt{MN}}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}C(u)C(v)F(u,v)\cos\frac{(2x+1)u\pi}{2M}\cos\frac{(2y+1)v\pi}{2N}$$

其中：$C(x)=\begin{cases}\frac{1}{\sqrt{2}} & x=0\\ 1 & x=1,2,\ldots,N-1\end{cases}$



DCT变换域

9	11	12	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

像素频率矩阵

基于DCT变换的水印技术

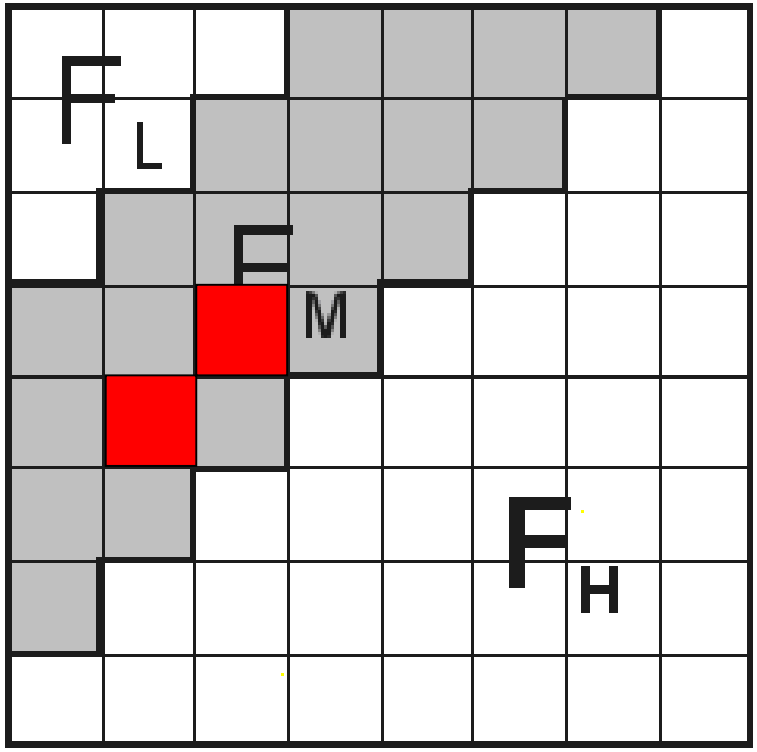
正变换：

$$F(u,v)=\frac{2}{\sqrt{MN}}c(u)c(v)\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\cos\frac{(2x+1)u\pi}{2M}\cos\frac{(2y+1)v\pi}{2N}$$

逆变换：

$$f(x,y)=\frac{2}{\sqrt{MN}}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}c(u)c(v)F(u,v)\cos\frac{(2x+1)u\pi}{2M}\cos\frac{(2y+1)v\pi}{2N}$$

其中：$c(x)=\begin{cases}\frac{1}{\sqrt{2}} & x=0\\ 1 & x=1,2,\ldots,N-1\end{cases}$



DCT变换域

如果Bi(u1,v1) - Bi(u2,v2) > k，则嵌入1，否则嵌入0

不满足关系式的系数值通过加入随机噪声进行修改

例：

初始值Bi(u1,v1) =22， Bi(u2,v2)=22

若k=1,想嵌入的值为1，则可以将Bi(u1,v1) 加上一个值，如2。

则Bi(u1,v1) =24， Bi(u1,v1) - Bi(u2,v2) =2 > k

如果Bi(u1,v1) - Bi(u2,v2) > k，则嵌入1，否则嵌入0
不满足关系式的系数值通过加入随机噪声进行修改

Watermarked Image

Recovered Watermark

DCT变换鲁棒性分析

5% Gaussian Noise

15% Gaussian Noise

JPEG Compression Q=50

JPEG Compression Q=20

对数字水印的攻击

- 鲁棒性攻击
  - 它包括常见的各种信号处理操作，如图象压缩、线性或非线性滤波、叠加噪声、图象量化与增强、图象裁剪、几何失真、模拟数字转换以及图象的校正等。

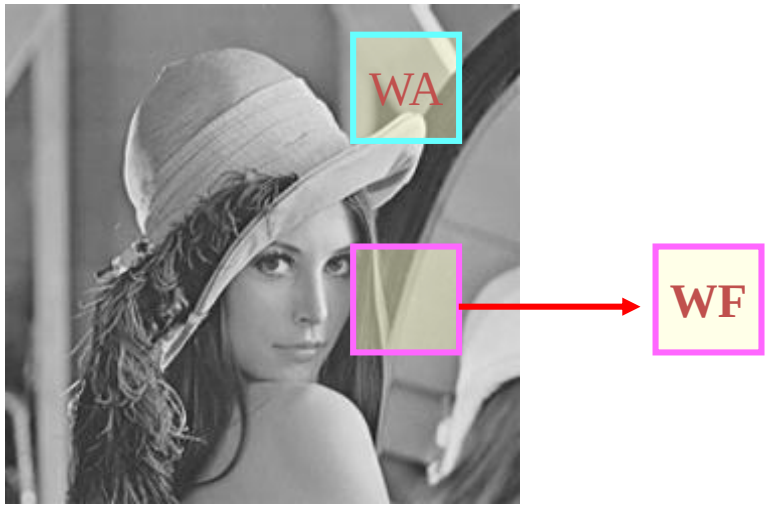
IBM攻击（解释攻击）

- 针对可逆、非盲（non-oblivious）水印算法而进行的攻击。
- 其原理为设原始图象为I，加入水印W_A的图象为$I_A=I+W_A$。攻击者首先生成自己的水印W_F，然后创建一个伪造的原图$I_F=I_A-W_F$，也即$I_A=I_F+W_F$。这就产生无法分辨与解释的情况。防止这一攻击的有效办法就是研究不可逆水印嵌入算法，如哈希过程。

原始图像

$I_A=I+W_A$

水印图像



$I_F=I_A-W_F$

伪造的原始图像

StirMark攻击

- Stirmark是英国剑桥大学开发的水印攻击软件，它采用软件方法，实现对水印载体图象进行的各种攻击，从而在水印载体图象中引入一定的误差，我们可以以水印检测器能否从遭受攻击的水印载体中提取/检测出水印信息来评定水印算法抗攻击的能力。如StirMark可对水印载体进行**重采样攻击**，它可模拟首先把图象用高质量打印机输出，然后再利用高质量扫描仪扫描重新得到其图象这一过程中引入的误差。

马赛克攻击

- 其攻击方法是首先把图象分割成为许多个小图象，然后将每个小图象放在HTML页面上拼凑成一个完整的图象。一般的Web浏览器都可以在组织这些图象时在图象中间不留任何缝隙，并且使其看起来这些图象的整体效果和原图一模一样，从而使得探测器无法从中检测到侵权行为。

串谋攻击

- 所谓串谋攻击就是利用同一原始多媒体数据集合的不同水印信号版本，来生成一个近似的多媒体数据集合，即同一图像加入不同水印后的多幅图像， 从各图像中分别截取一小部分，对应拼接成新的图像数据，以此来逼近和恢复原始数据。
- 其目的是使检测系统无法在这一近似的数据集合中检测出水印信号的存在。

跳跃攻击

- 跳跃攻击主要用于对音频信号数字水印系统的攻击，其一般实现方法是在音频信号上加入一个跳跃信号，即首先将信号数据分成500个采样点为一个单位的数据块，然后在每一数据块中随机复制或删除一个采样点，来得到499或501个采样点的数据块，然后将数据块按原来顺序重新组合起来。实验表明，这种改变对古典音乐信号数据也几乎感觉不到，但是却可以非常有效地阻止水印信号的检测定位，以达到难以提取水印信号的目的。类似的方法也可以用来攻击图象数据的数字水印系统，其实现方法也非常简单，即只要随机地删除一定数量的象素列，然后用另外的象素列补齐即可，该方法虽然简单，但是仍然能有效破坏水印信号存在的检验。

数字水印的应用

虽然**信息防伪**（所有者鉴别）和**版权保护**（所有权验证）是数字水印领域研究的主要驱动力，但目前还有许多其他应用：

- 所有者鉴别：嵌入代表作品版权所有者身份的水印。
- 所有权验证：在发生所有权纠纷时，用水印来提供证据。
- 广播监控：通过识别嵌入到作品中的水印来鉴别作品是何时何地被广播的。
- 拷贝跟踪：用水印来鉴别合法获得内容但非法重新发送内容的人。
- 内容认证：将签名信息嵌入到内容中以待日后检查内容是否被篡改。
- 拷贝控制：使用水印来告知录制设备不能录制什么内容。
- 设备控制：使用水印来制造设备，比如Digimarc公司的MediaBridge系统。

关注我，下节内容更精彩：第二章：访问控制