07.2
JPEG图像的隐写与隐写分析

钮心忻、杨榆、雷敏
北京邮电大学信息安全中心
yangyu@bupt.edu.cn

JPEG图像隐写软件

- Jsteg
- OutGuess
- F5
- 改变了载体图像的DCT直方图或分块效应

JPEG压缩过程

- 将原始图像分割为8×8的小块
- 每小块作二维DCT变换
  - 左上角为直流，zigzag扫描，对应频率从低到高
- 对DCT系数进行量化
  - 对不同频率成分采用不同的量化步长
  - 量化后的DCT系数是整数

JPEG压缩过程

- 标准量化表

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

Jsteg密写

- 将秘密信息嵌入在量化后的DCT系数的LSB上。但原始值为－1，0，＋1的DCT系数除外。
- 提取秘密信息时：将图像中不等于－1，0，＋1的量化DCT系数的LSB取出即可。

Jsteg密写——实例

- 原图像块的像素值为

$$\mathbf{I}=\begin{bmatrix}139 & 144 & 149 & 153 & 155 & 155 & 155 & 155\\ 144 & 151 & 153 & 156 & 159 & 156 & 156 & 156\\ 150 & 155 & 160 & 163 & 158 & 156 & 156 & 156\\ 159 & 161 & 162 & 160 & 160 & 159 & 159 & 159\\ 159 & 160 & 161 & 162 & 162 & 155 & 155 & 155\\ 161 & 161 & 161 & 161 & 160 & 157 & 157 & 157\\ 162 & 162 & 161 & 163 & 162 & 157 & 157 & 157\\ 162 & 162 & 161 & 161 & 163 & 158 & 158 & 158\end{bmatrix}$$

Jsteg密写——实例

- DCT变换后的系数矩阵为

$$\mathbf{Y}=\begin{bmatrix} 1260 & -1 & -12 & -5 & 2 & -2 & -3 & 1 \\ -23 & -17 & -6 & -3 & -3 & 0 & 0 & -1 \\ -11 & -9 & -2 & 2 & 0 & -1 & -1 & 0 \\ -7 & -2 & 0 & 1 & 1 & 0 & 0 & 0 \\ -1 & -1 & 1 & 2 & 0 & -1 & 1 & 1 \\ 2 & 0 & 2 & 0 & -1 & 1 & 1 & -1 \\ -1 & 0 & 0 & -1 & 0 & 2 & 1 & -1 \\ -3 & 2 & -4 & -2 & 2 & 1 & -1 & 0 \end{bmatrix}$$

Jsteg密写

- 以标准量化表量化后的系数矩阵为

$$\mathbf{Y}_Q = \begin{bmatrix} 79 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Jsteg密写

- 其中，只有两个系数可以隐藏秘密信息，如果需要隐藏的为01

$$\mathbf{Y}'_Q = \begin{bmatrix} 78 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -3 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Jsteg密写

- 用量化表进行逆量化，做逆DCT变换，得到密写图像块
- 对比发现，几乎每个像素都有变化

$$\mathbf{I}'=\begin{bmatrix}140&142&145&148&150&151&152&152\\144&146&148&151&152&153&152&152\\151&152&154&155&155&154&153&152\\158&158&159&159&158&156&154&153\\161&162&162&162&160&158&156&154\\162&162&163&163&162&160&157&156\\160&161&162&162&162&160&159&157\\158&159&161&162&162&161&159&158\end{bmatrix}$$

$$\mathbf{I}=\begin{bmatrix}139&144&149&153&155&155&155&155\\144&151&153&156&159&156&156&156\\150&155&160&163&158&156&156&156\\159&161&162&160&160&159&159&159\\159&160&161&162&162&155&155&155\\161&161&161&161&160&157&157&157\\162&162&161&163&162&157&157&157\\162&162&161&161&163&158&158&158\end{bmatrix}$$

Jsteg隐写

- 将秘密信息嵌入在量化后的DCT系数的LSB上。但原始值为－1，0，＋1的DCT系数除外。
- 提取秘密信息时：将图像中不等于－1，0，＋1的量化DCT系数的LSB取出即可
- Jsteg隐写就是对DCT系数进行LSB隐写，用χ^2分析可以进行隐写分析

基于量化表调整的隐写

- Jsteg隐写可嵌入的DCT系数非常少，因此隐写量较小
- 提出基于量化表调整的隐写：不使用标准量化表

- 不同之处在于：中频量化步长值为1

$$Q'=\begin{bmatrix} 16 & 11 & 10 & 16 & 1 & 1 & 1 & 1 \\ 12 & 12 & 14 & 1 & 1 & 1 & 1 & 55 \\ 14 & 13 & 1 & 1 & 1 & 1 & 69 & 56 \\ 14 & 1 & 1 & 1 & 1 & 87 & 80 & 62 \\ 1 & 1 & 1 & 1 & 68 & 109 & 109 & 77 \\ 1 & 1 & 1 & 64 & 81 & 104 & 104 & 92 \\ 1 & 1 & 78 & 87 & 103 & 121 & 121 & 101 \\ 1 & 92 & 95 & 98 & 112 & 100 & 100 & 99 \end{bmatrix}$$

基于量化表调整的隐写

- 有较多的量化DCT系数可以进行隐写
- 问题
  - 仍然是LSB隐写，用χ2分析可以进行隐写分析
  - 量化表中量化步长的异常，会引起分析者怀疑

F5隐写

- JPEG图像的DCT系数特点
  - DCT系数的绝对值越大，其出现的频率越低
  - 随着DCT系数绝对值的升高，其出现次数下降的幅度减小

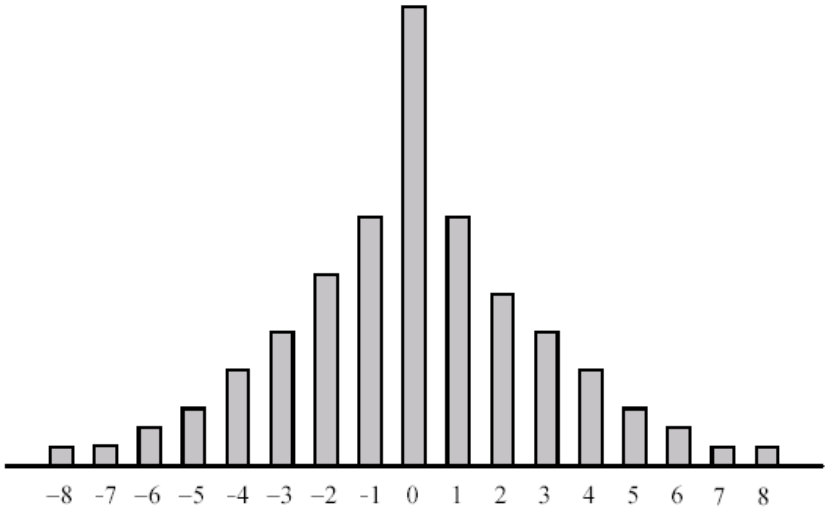

图 5.2.1 JPEG 图象 DCT 系数直方图特性示意

F3隐写

- F5隐写方法是由F3、F4发展而来
- F3隐写
  - 每个非0的DCT系数用于隐藏1比特秘密信息
  - 秘密信息与DCT系数的LSB相同，则不改动；不同则将DCT系数绝对值减1，符号不变
  - 如果原始值为＋1或－1，嵌入比特0时，变为0，此隐藏视为无效，在下一个系数上重新嵌入
  - 提取时：将不为0的DCT系数的LSB取出即可

F3隐写

- 例：
  - 已知JPEG图像DCT系数为： -9, -4, 0,0,0,1
  - 且
    - 已知负整数最低比特位与其奇偶性一致
    - 已知该图像使用了F3隐藏
  - 则从中能提取几比特秘密信息？秘密信息为？

F3隐写

- F3隐写的特点
  - 隐写是将绝对值减1，而不是LSB替换，因此可以抵抗χ2分析
  - 漏洞
    - 算法约定：“原始值为＋1或－1，嵌入比特0时，变为0，此隐藏视为无效，在下一个系数上重新嵌入”
    - 造成隐藏了更多的比特0
    - 因此隐写后DCT系数直方图中，偶数位置上的灰色柱比奇数位置上的要突出

F3隐写

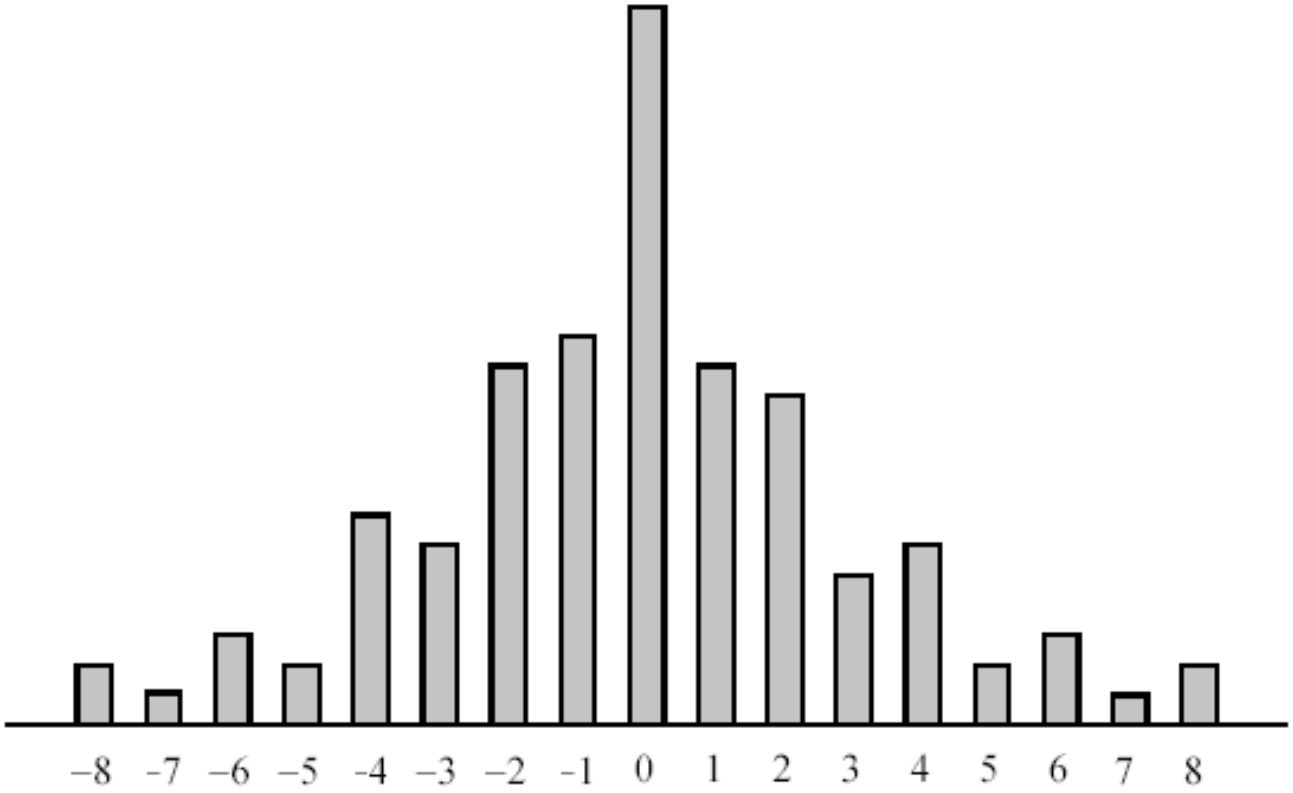

图 5.2.2 F3 密写后图象 DCT 系数直方图特性示意

F4隐写

- F4隐写
  - 用正奇数和负偶数代表秘密信息1
  - 用负奇数和正偶数代表秘密信息0
  - 值为0的DCT系数仍然不负载秘密信息
  - 当欲嵌入的比特与DCT系数代表的信息不同时，同样将绝对值减1，符号不变

F4隐写

- F4隐写
  - 如果嵌入时产生了0系数，同样无效，在下一个系数上重新嵌入
- 与F3的区别
  - 不仅嵌入比特0时可能产生无效隐藏，嵌入比特1时也会产生无效隐藏，需要重新嵌入
  - 所以偶数柱比奇数柱突出的特点不会出现

F4隐写

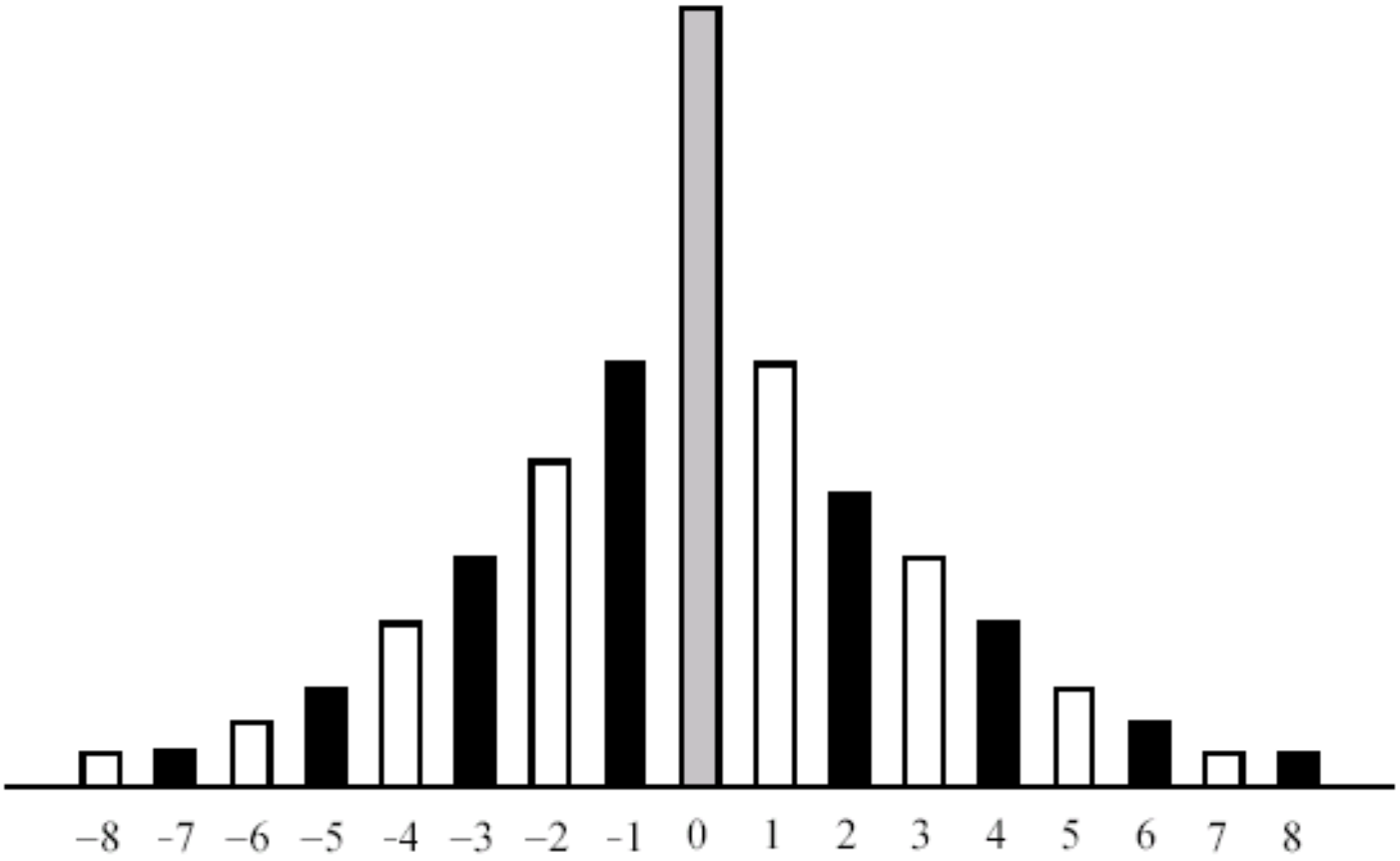

图 5.2.3 F4 密写图象的 DCT 系数直方图

F5隐写

- F5隐写
  - 在F4的基础上，添加了混洗技术和矩阵编码技术
- 混洗：使秘密信息分布在整个图像中
  - 如果可携带秘密信息的DCT系数个数大于秘密信息长度，采取混洗的方法，随机选择系数进行嵌入
- 矩阵编码
  - 优点：减少了LSB的修改量
  - 缺点：降低嵌入量

F5隐写：矩阵编码

- LSB隐写
  - 嵌入1比特可能修改也可能不修改原数据，概率为0.5
  - 则每个LSB的修改可以平均嵌入2比特信息
- 矩阵编码的目的是，使得每个LSB的修改可以嵌入更多的比特信息
  - 在2^k-1个原始数据的LSB中最多改动1比特达到嵌入k比特的效果
  - k=1：普通LSB隐写
  - k=2：在3个数据上，只修改1比特，代表嵌入2比特

矩阵编码

- 例如：k=2
- 设a1, a2, a3是三个载体数据的LSB
- 设x1, x2是要嵌入的秘密比特
- 如果 $x_1 = a_1 \oplus a_3,\quad x_2 = a_2 \oplus a_3$ 不改变原数据
- 如果 $x_1 \neq a_1 \oplus a_3,\quad x_2 = a_2 \oplus a_3$ 改变a1
- 如果 $x_1 = a_1 \oplus a_3,\quad x_2 \neq a_2 \oplus a_3$ 改变a2
- 如果 $x_1 \neq a_1 \oplus a_3,\quad x_2 \neq a_2 \oplus a_3$ 改变a3
- 提取：a1与a3异或为x1，a2与a3异或为x2

例

- 在（-24，32，-32）（-27，28,29）(20, -1,-5)上用F5算法隐藏101010，则隐藏后，载体变为:
- 根据F5定义（正奇负偶为1，负奇正偶为0），载体对应的比特为:
- (1,0,1)(0,0,1)(0,0,0)
- 要隐藏101010，根据矩阵编码
- 1~=1xor1,0~=0xor1,则修改a3(1)->0，即符号不变，数值减1，变为-31，第一组数据为：-24,32,-31
- 1==0xor1,0~=0xor1，则修改a2(0)->1，即符号不变，数值减1，变为27，第二组数据为：-27，27,29
- 1~=0xor0，0==0xor0，则修改a1(0)->1，即符号不变，数值减1，变为19，第三组数据为：(19, -1,-5)

矩阵编码

- 矩阵编码的特点
  - 嵌入效率：嵌入比特数/平均修改长度
  - 嵌入效率高：同样嵌入量，对图像的修改少，失真小
  - 载体数据利用率：嵌入比特数/所需像素数
  - 载体数据利用高：同样的嵌入量，所需的像素少

例——分析嵌入效率和数据利用率

- 当k=2时，按x1是否等于xor(a1,a3)以及x2是否等于xor(a2,a3)划分，有四种情况，即：
- a.x1=xor(a1,a3)，x2=xor(a2,a3)
- b. x1!=xor(a1,a3)，x2=xor(a2,a3)
- c. x1=xor(a1,a3)，x2!=xor(a2,a3)
- d. x1!=xor(a1,a3)，x2!=xor(a2,a3)
- 每种情况发生的概率都相同，即皆为1/4。而除了情况a下，不需要修改LSB以外，其它情况下都要修改1个LSB，所以平均修改长度为：1/4*0+1/4*1*3=3/4。此时，
- 嵌入效率=嵌入比特数/平均嵌入长度=2/(3/4)=8/3
- 而普通LSB算法中，
- 嵌入效率=嵌入比特数/平均嵌入长度
  =1/(1/2*0+1/2*1)=1/(1/2)=2<2.5=7.5/3<8/3

例——分析嵌入效率和数据利用率

- 嵌入效率和数据利用率比较
  - k= 2时，嵌入2比特平均修改3/4个LSB，
  - 普通LSB：嵌入1比特平均修改1/2个LSB，
  - 嵌入效率比普通LSB高。
  - k=2时，用3个数据负载2比特，
  - 普通LSB：1个数据负载1比特，
  - 载体数据利用率比普通LSB低。

矩阵编码

- k越大，载体数据利用率越低，嵌入效率越高

表 5.2.2 不同 k 值相应的矩阵编码性能

k	n	载体数据利用率	嵌入效率
1	1	100.00%	2.00
2	3	66.67%	2.67
3	7	42.86%	3.43
4	15	26.67%	4.27
5	31	16.13%	5.16
6	63	9.52%	6.09
7	127	5.51%	7.06
8	255	3.14%	8.03
9	511	1.76%	9.02

F5隐写

- 步骤
  - 进行JPEG压缩，量化DCT系数
  - 混洗DCT系数
  - 确定k，并计算 $n = 2^k - 1$
  - 实施矩阵编码嵌入
  - 逆混洗，产生隐写后的图像

F5隐写

- DCT系数直方图的原始特性不变
- P(X=x)表示原始图像中，DCT系数等于x的概率
- P(Y=y)表示隐写图像中，DCT系数等于y的概率
- a表示非0DCT系数被改动的概率，k=1时，a=1-a，k>1时，a<1-a

F5隐写

- 隐写后，直方图统计特性：DCT系数的绝对值越大，其出现的频率越低，得以保持
  - P(Y=1)=(1-a)P(X=1)+aP(X=2)
  - P(Y=2)=(1-a)P(X=2)+aP(X=3)
  - P(Y=3)=(1-a)P(X=3)+aP(X=4)
  - P(Y=1)-P(Y=2)
  - =(1-a)[p(X=1)-P(X=2)]+a[p(X=2)-P(X=3)]
  - >(1-a)[p(X=2)-P(X=3)]+a[p(X=2)-P(X=3)]*
  - =p(X=2)-P(X=3)
  - >0
  - P(Y=2)-P(Y=3)>0…

F5隐写

- 隐写后，直方图统计特性：随着DCT系数绝对值的升高，其出现次数下降的幅度减小，得以保持
  - P(Y=1)=(1-a)P(X=1)+aP(X=2)
  - P(Y=2)=(1-a)P(X=2)+aP(X=3)
  - P(Y=3)=(1-a)P(X=3)+aP(X=4)
  - L1-L2=(P(Y=1)-P(Y=2))-(P(Y=2)-P(Y=3))
  - =(1-a)[(p(X=1)-P(X=2))-(p(X=2)-P(X=3))]+a[(p(X=2)-P(X=3))-(p(X=3)-P(X=4))]
  - =(1-a)[d1-d2]+a[d2-d3]
  - >a[d1-d2]+a[d2-d3]*
  - =a[d1-d3]
  - >0
  - P(Y=1)-P(Y=2)>P(Y=2)-P(Y=3)

总结F5隐写可能的漏洞

- DCT系数绝对值减1
  - 直方图奇偶不均衡的特点不会出现
  - 直方图会由两端向中间收缩
- DCT系数量化是分块进行的
  - 不同小块之间会有一定的不连续性
  - 当压缩比较高时，人眼可以分辨出小块的界限；用高通滤波后，界限更明显
  - F5隐写后，小块间的不连续性更明显

JPEG图像隐写分析

- 直方图分析
- 分块特性分析

JPEG图像隐写分析

- 分析者无法得到原始图像，但是能构造一个统计特性相近的参考图像
  - 将待测图像删除前四行（或前四列），得到参考图像
  - 重新分块，DCT变换，量化
- 参考图像与原始图像有相近的内容，使用相同的量化表
- 参考图像的DCT直方图和分块特性作为原始图像的估计

JPEG图像隐写分析

- 分块特性

$$B = \sum_{i=1}^{\lfloor (M-1)/8 \rfloor} \sum_{j=1}^{N} \left| g_{8i,j} - g_{8i+1,j} \right| + \sum_{i=1}^{M} \sum_{j=1}^{\lfloor (N-1)/8 \rfloor} \left| g_{i,8j} - g_{i,8j+1} \right|$$

- M，N是图像的行数和列数
- g是像素灰度值
- 分块特性B表示相邻块的相邻像素灰度值之差的绝对值总和

JPEG图像隐写分析

- 计算参考图像的分块特性Be
- 计算待测图像的分块特性B1
- 如果B1明显大于Be
- 或，参考图像与待测图像的DCT系数直方图存在明显差异
- 则可以认为待测图像是经过隐写的
- 此方法可以察觉Jsteg、OutGuess、F5等方法嵌入的秘密信息

例1

- 512×512
- 质量因子70
- JPEG图像

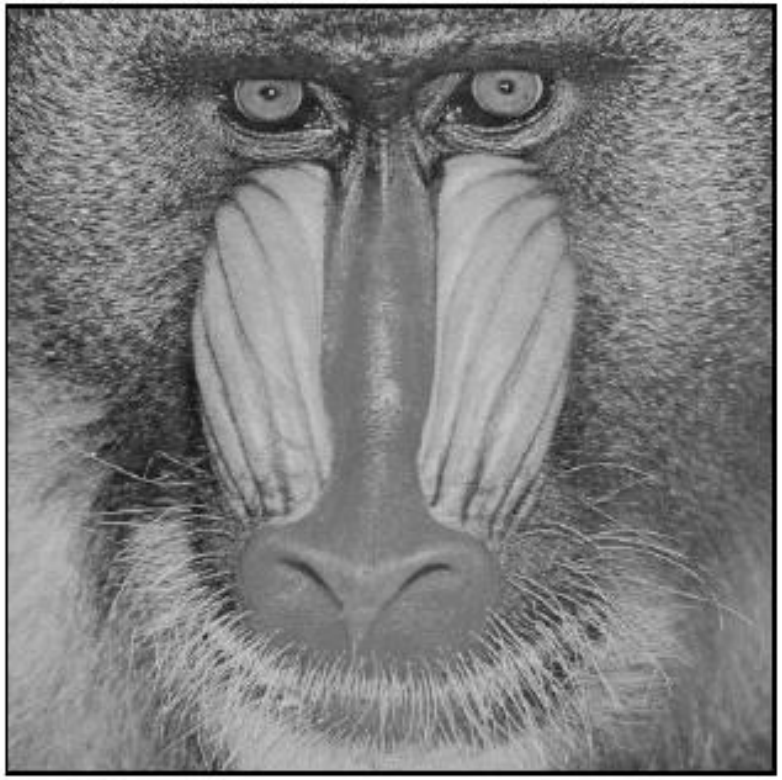

图 5.3.1 质量因子为 70 原始 JPEG 图象 Baboon

例1

- 原始直方图
- 参考图像直方图
- F5隐写后的直方图

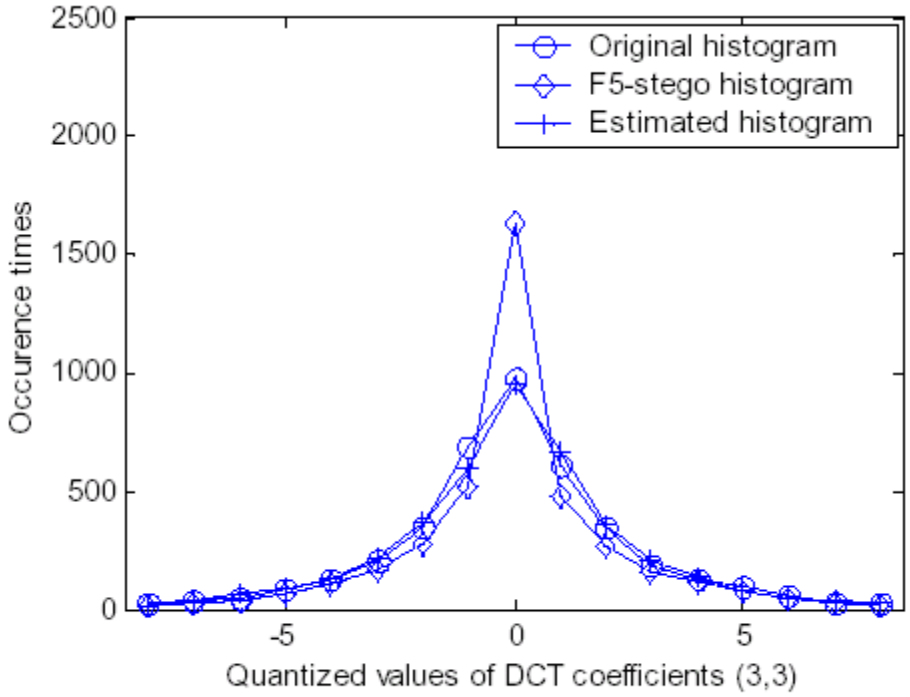

图 5.3.2 原始直方图、估计直方图与 F5 密写后的直方图

例2

- 512×512
- 质量因子70
- JPEG图像

图 5.3.3 质量因子为 70 的原始 JPEG 图象 Man

例2

- 原始图像分块特性值5.40E5
- 参考图像分块特性值5.51E5
- F5隐写后图像分块特性值6.56E5

安全的JPEG隐写

- 隐写时兼顾图像的统计特性
  - 不改变DCT直方图
  - 不改变分块特性

新隐写算法的原则

- 秘密信息嵌入在非零、非直流的DCT系数上，每个系数负载1比特
- 用正奇数和负偶数代表秘密信息1
- 用负奇数和正偶数代表秘密信息0

新隐写算法的步骤

- 计算原始图像DCT系数直方图，和分块特性
- 将秘密信息每比特对应一个非零、非直流DCT系数
  - 相同：不作任何改动
  - 不同：修改DCT系数
- 修改系数时：可以加1，也可以减1
  - 即：正向调整，负向调整

新隐写算法的步骤

- 在选择用加1还是减1的方法修改系数时，计算直方图的改变，使得调整后直方图与原始图像直方图近似不变
- 同时，计算分块特性，选择合适的修改（加1还是减1），使得分块特性不会大大偏离原始分块特性

新隐写算法的步骤

- 提取
  - 取出非零、非直流的量化后DCT系数
  - 正奇数或负偶数：1
  - 负奇数或正偶数：0

例：新隐写算法结果

- 可嵌入4.5E4个秘密信息比特
- 隐写后PSNR＝35.1dB

图 5.4.1 密写后的图象

例：新隐写算法结果

- 原始图像分块特性B0＝5.40E5
- 参考图像分块特性Be＝5.51E5
- 含密图像分块特性B1＝5.40E5

例：新隐写算法结果

- 直方图

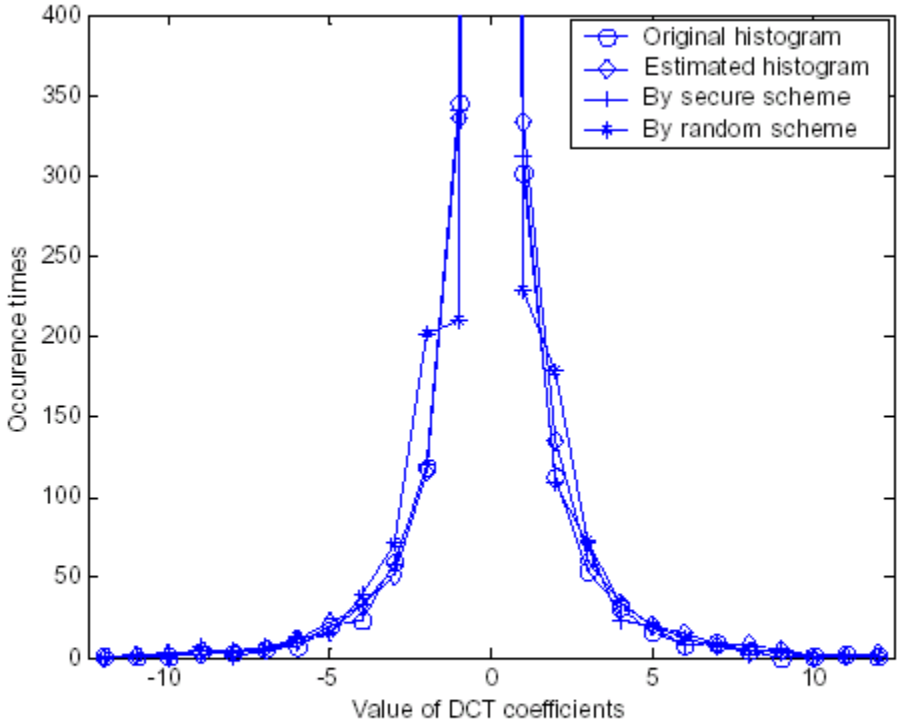

图 5.4.2　(3,3)位置 DCT 系数的原始直方图、估计直方图、使用安全方案和随机方案密写后的直方图

算法比较

- 用数字相机采集100幅人物、风景图片
- 以质量因子70压缩后作为原始图像集
- 用**随机修改的方法**和**新隐写算法**进行隐写

算法比较——分块特性比较

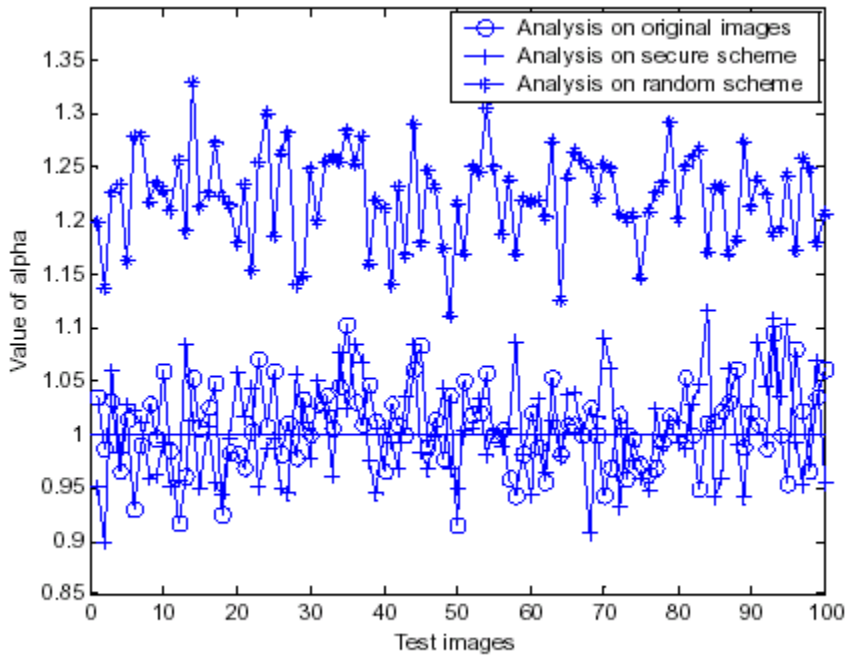

图 5.4.3 对原始图象、安全方案和随机方案产生的密写图象进行分块特性分析的结果

- 计算待测图像的分块特性B1
- 计算参考图像的分块特性Be
- 计算 α=B1/Be

算法比较——直方图比较

- 计算待测图像和参考图像的直方图差异

$$\rho_k = \frac{\sum_{m\neq 0}\left|h^{\mathrm{R}}{}_k(m) - h^{\mathrm{E}}{}_k(m)\right|}{\sum_{m\neq 0} h^{\mathrm{E}}{}_k(m)}$$

算法比较——直方图比较

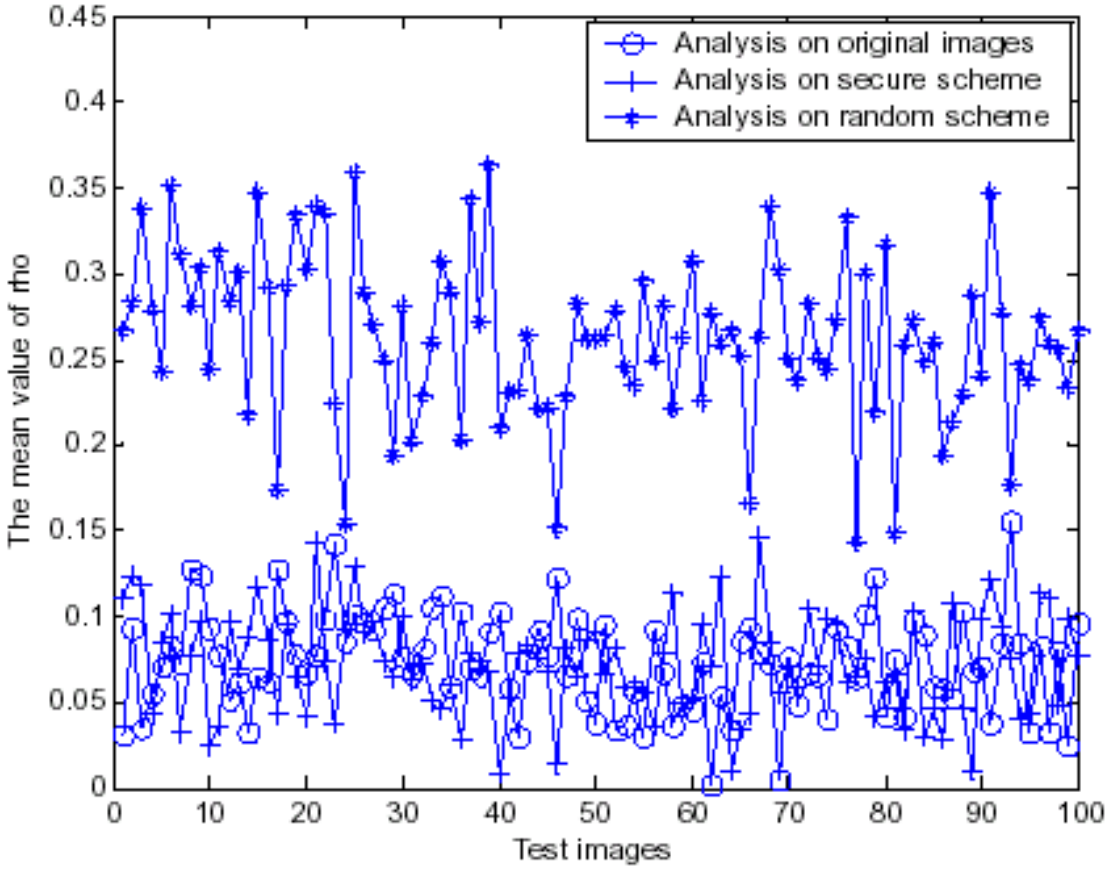

图 5.4.4 对原始图象、安全方案和随机方案产生的密写图象进行直方图分析的结果

结论

- 安全隐写方案
  - 分块特性
  - 直方图
  - 无明显差异
- 存在的问题
  - 质量因子越小时，DCT量化步长越大，隐写时不太容易同时保持直方图和分块特性不变