论文题目　　基于 Hill 矩阵与 Haar 域序列的图像加密算法研究

姓　　名　　马林海

学　　院　　软件学院

专　　业　　信息安全

指导教师　　谭振华　秦瑶

备　　注

2014 年 6 月 10 日

基于 Hill 矩阵与 Haar 域序列的图像加密算法研究

作者姓名：　　马林海

校内指导教师：　谭振华　　副教授

校外指导教师：　秦瑶　　高级工程师

单位名称：　　东软集团

专业名称：　　信息安全

东 北 大 学

2014 年 6 月

Research of Image Encryption Algorithm Based on Hill Matrix and Haar Domain Sequence

by Ma Linhai

Supervisor: Associate Professor Tan Zhenhua
Associate Supervisor: Senior Engineer Qin Yao

Northeastern University

June 2014

毕业设计（论文）任务书

毕业设计（论文）题目：

基于 Hill 矩阵与 Haar 域序列的图像加密算法研究

基本内容：

研究 Hill 矩阵与 Haar 域序列加密算法，学习 Visual Studio2008 集成开发环境，以及 c#.net Framework 3.5。分析图像加密要求，把 Hill 矩阵与 Haar 域序列的图像加密算法进行整合，设计基于 Hill 矩阵与 Haar 域序列的图像加密算法，将图像通过“Hill 矩阵”加密后，再进行“Haar 域序列置乱”加密，并开发完成一个完整的系统。之后会选取多种图像测试与评估的方法对已制作的加密系统进行对比性质的评估测试。在此基础上得出研究成果。翻译一篇与毕设内容相关的外文资料，译文汉字字数不少于 4000 字。

毕业设计（论文）专题部分：

题目：

基本内容：

学生接受毕业设计（论文）题目日期

第 1 周

指导教师签字：

2014 年 3 月 7 日

基于 Hill 矩阵与 Haar 域序列的图像加密算法研究

摘 要

随着信息科学的不断发展，图像类文件经常传播于网络之上，图像内容的安全已经逐渐引起了人们的注意，对图像内容加解密的讨论与研究也逐渐升温。本文基于 Hill 矩阵加密算法、Haar 域置乱加密算法对图像加密展开了研究。

本文首先对图像加密算法的相关原理进行了阐述，重点分析了 Hill 矩阵加密及 Haar 域置乱算法。在此基础上，基于自可逆矩阵对 Hill 加密算法进行了改进，并结合 Haar 域置乱提出了一种混合的图像加密算法。为了减少图像失真，本文使用 Haar 小波的提升变换及二维超混沌映射系统来对算法进行实现。最后，基于直方图、相关性、信息熵等对加密算法进行了性能评估设计。

测试及分析结果表明，本文所提加密系统在应对针对密文类的统计分析类的攻击时，有较好的对抗性，但是在面对针对明文类的差分攻击时，在 NPCR 和 UACI 两个指标数值都很低的本加密系统抵抗能力较弱。

关键词：图像加密，Hill 矩阵，自可逆矩阵，Haar 小波，混沌映射

Research of Image Encryption Algorithm Based on Hill Matrix and Haar Domain Sequence

Abstract

With the development of information science, and image files are often spread on the network. Image security content has gradually attracted people's attention. Discussion and research on image encryption and decryption is gradually warming. . This paper is a research paper based on image encryption algorithm， which consists of two, one is Hill matrix image encryption algorithm, the other is Haar domain scrambling image encryption algorithm.

This paper firstly describes the principles of encryption of Hill matrix,, analyzes the Hill matrix encryption algorithm and Haar matrix domain scrambling encryption algorithm. Then, using since reversible matrix to improve the Hill encryption algorithm, this research combined it with Haar domain scrambling algorithm to propose a new image encryption algorithm. In order to reduce image distortion, this time, enhancing the Haar wavelet transform and two-dimensional mapping system are used to achieve the final encryption system. Finally, based on the histogram, correlation and information entropy, this program creates encryption algorithm performance evaluation function.

The testing for this encryption system showed that, the encryption system is good at resisting attacking based on analysis of cipher text. But this algorithm with a low NPCR and a low UACI is weak to differential attacking.

Key words: Image Encryption, Hill Matrix, the self-reversible matrix, Haar wavelet, Chaotic Mapping

目 录

第 1 章 绪 论

1.1 研究背景

1.1.1 总体背景

随着互联网的兴起，图像视频等多媒体信息需要在网络上传输，于是产生了图像信息安全的问题。图像加密是首要的解决方案，优势相对独立的信息安全分支。在国内外的琳琅满目的科研论文与期刊之中，图像加密也是被广泛关注的热点，是信息安全之中的一处重要领域。

1.1.2 技术层面

从技术层面来说，图像加密是图像信息安全的核心技术，是隐藏信息和数字水印的必须技术[1]。图像加密既属于数字图象处理领域，也属于图像通信学科；与此同时，他属于信息安全的领域，处于密码学的新领域，与计算机密码学的很多算法和技术有着密切的联系，并在此基础之上又发展了图像加密特有的方案和算法。数字加密的特有的方案和算法除了涉及计算机密码学和数字图象处理的理论与算法之外，还设计了混沌与分型、编码理论、数据的压缩、智能算法、小波与傅立叶分析、神经网络、计算方法与近代数学等很多的学科和领域。图像信息安全是集数学、密码学、信息论、概率论、计算复杂度理论、计算机网络等其他计算机科学在内于一体的多学科交叉研究课题[2][3]。因此，掌握图形加密的理论需要非常多的知识与技能。

1.1.3 经济与社会效益

从经济与社会效益方面来看，图形加密它既然从属于信息安全领域。凡是信息安全的问题，事无巨细，每一个问题都是极其的重要，每一个分支都是极其重要的领域。因为这些都关系到了国家与人民的安全。图像数据的获取、传输、处理遍及数字时代的各个角落。安全问题也日益严重。很多图像数据需要进行保密传输和存储，例如军用卫星拍摄的图片、新型武器图纸、金融机构建筑图等，还有些图像信息根据法律必须要在网络上加密传输，例如在远程医疗系统中，患者的病历和医学影像。由于这些图像数据的特殊性，使其经常遭受一些别有用心的剽窃与盗取，从而对于相关的人员和团体的工作甚至日常生活造成难以想象的困难与损失。图像加密技术可以将这些重要的图像信息处理为杂乱无章的类似噪音的图像，使未授权者无法浏览或修改这些信息。从而保证了人

们赖以生存的基本利益。

1.2 国内外研究现状

与古典的文本信息加解密技术所不同，图像数据有其自身的特点，比如像素之间有很高的冗余度和相关性，而且，一般来说数据包含的数据量很大。除此之外，数据加密还有些特别的要求，如实时性的加解密、保真度、图像格式的一致性和为快速传输进行的图像压缩等等。传统的加密算法如果强行运用于图像加密，不仅加密速度让人难以接受，而且高度安全和高质量地进行加解密也是个需要解决的问题。因而，在国内外图像加解密算法的发展历程之中，加解密方法的比较测试一直都是主线。通过比较测试，人们总是可以找到各自加密法的优点与不足，而图像加解密算法也是这样的发展着。

1.2.1 国内外图形加密算法的发展历史

1989 年，英国数学家 Mattlews 首次提出混沌加密[4]。

1991 年，Schwartz[5]提出了用置乱的方法加密图像首先在原图之上生成随机点序列，然后，在序列的每两个相邻点之间画出线条。进而，以相反的模式画图，即将白色改为黑色，在原图上花了很多相反的线条后，原图就被加密了。但是，这些随机点是由随机数发生器的种子确定，这种子就是这个加密算法的密钥，这种方法简单而且快速，但是存在的问题在于安全性不高，对于图像的保护不够。

之后，在 1992 年，使用 SCAN[6]语言进行图像加密的方法被提出。这种方法是把二维的图像转变为一维的序列，并用 SCAN 语言描述转变的结果。在确定了 SCAN 语言的字母组合之后，，就产生了一个 SCAN 串，这个串中蕴含的信息就是对原始图像的扫描次序。按照这个次序进行扫描，用 SCAN 串进行加密，从而达到加密的目的。非法用户无法得到原 SCAN 串，因而原图是安全的。但是这个方法没有对原图像的压缩，因而并不是很高效的直接加密法。

在之后，又接连产生了好多特色各异的图像加解密算法。其中比较有影响力的是混沌加密法。混沌加密法整体上的分为模拟混沌加密系和离散混沌加密系[7]。模拟混沌加密系主要是以混沌同步技术为核心的混沌保密通信系统，主要基于模拟混沌电路系统。这与本文所研究的方向有一定的差距，故而不再多说。

离散混沌加密主要基于计算机有限精度下实现的数字化混沌系统，细分为流密码和分组密码。混沌流密码主要使用到 Logistic 映射，这种算法的优点在于算法简单、加密速度快、安全性较高，缺点在于没有考虑图像的特点。而分组密码，是通过对图像的折

叠和拉伸，产生二维混沌映射，而后，通过迭代映射置乱图像中的像素，进行加密。常见的二维混沌映射有 Baker Map、Cat Map、Standard Map 和 Tent Map 等，其中加密效果好、应用广的是 Baker Map。这种加密方法的优点在加密速度快、安全性高，加密过程无信息损失；缺点在于密钥受图像大小限制，对加密对象要求较严。

除此之外，还有很多很多的图形加密算法。比较经典的是 1983 年 Wolfman[8]奠定了自动细胞机理论，被后人所用，大量的关于此的算法在 2002 年之后相继被提出。还有本文本次毕设中所用到的带自可逆矩阵的“Hill 矩阵”加密法，既可用于灰度加密，又可用于彩色的加密[9]。他们各有各的优缺点，但总体来说都是朝着一个大致的趋势。

1.2.2 国内外图像加解密算法的发展趋势

有关图像加密的算法太多太多，自 1991 年到如今可圈可点的就有几十个。纵观图像加密算法的发展，本文大概可以总结出图像加密算法的发展方向。一个完善的加密算法不仅在安全机制方面是柔韧的，而且总是可以高效地高度安全地运行；该算法应该尽可能的面面俱到，并且在尽量小的影响系统性能的情况下，尽量的使图像数据精简的的要求下，保证图像的安全性。

1.3 研究内容

通过以上的叙述，本文大致知道了现在图像加解密的领域的发展与趋势。最后也点到了，尽管图像加密的算法层出不穷，形式各异，但是其最基本的要求是恒定的，那就是：尽量的保证图像的安全性。而图像安全性的保障在于对于一些攻击的防范上。而本文此次的算法设计类的毕业设计也是围绕这一点来的。

本文主要内容，在于将 Hill 矩阵与 Haar 域序列的图像加密算法进行整合，也就是将图像通过“Hill 矩阵”加密后，再进行“Haar 域序列置乱”加密，并形成一个完整的系统。之后选取多种图像测试与评估的方法对已制作的加密系统进行评估测试。在此基础上得出研究成果。

本文所设计的图像加密算法，来自于两种基本的原始加密算法，Hill 加密算法[10]和 Haar 离散小波分解并置乱的加密[11]算法。原始的 Hill 矩阵加密算法是由 Hill 在 1929 年提出的[12]，其基本的思想是将 n 个明文字母通过线性变换转换为 n 个密文字母，也就是用一次矩阵的相乘；解密只需要求得该矩阵的逆矩阵即可，也就是进行一次逆变换。而密钥，就是矩阵的本身。本毕设中将会将此经典加密算法稍作改动，并将之作为一部分，加入到一个整体的图像加解密的系统之中。

本次毕业设计牵扯到的另外一种加密算法是 DWT 域成对 Logistic 混沌置乱加密方案[11]，大致思想是将图像进行多层的小波分解，在每一层进行频域置乱操作。当然本次试验并未采用原始的成对 Logistic 映射作为对每一层的置乱映射，而使用的是具有较高复杂度的二维超混沌系统。关于此混沌系统，本文之后会有详细的介绍。

而对于之后的对加密法的测试与评估的方法，本次毕设会使用统计分析和扩散性测试的分析与测试方法。统计分析的测试方法将使用像素直方图与方差、相邻像素的相关性分析、以及加密图像的信息熵的计算来评估算法混乱与扩散性能方面抵抗统计攻击的能力。而扩散性测试中，本文将运用像素改变率和一致平均改变强度的指标来测试加密图的原图一个像素改变后对加密图像的影响。两种指标的详细的计算方式将在第二章中进行详细介绍。

本次毕业设计会使用统计分析和扩散性测试的分析与测试方法对所设计的图像加密算法进行测试评估。其中并以此来证明，本次的毕业设计所设计的算法，将在混乱和扩散性能方面抵抗统计攻击的能力，以及对抗差分攻击的能力之上将会有较为不错的表现。

1.4 论文组织结构

本论文主体部分共分 5 章，每章的组织结构安排和内容如下：

第 1 章是绪论。本章简要介绍本次毕业设计所牵扯到的图像加密的背景、国内外相关专业的研究现状与发展趋势、毕业设计所牵扯到的基本内容以及其创新点与意义，和本论文的结构。

第 2 章是相关工作与理论基础。本章介绍了本次算法设计所涉及的理论层面的问题和所用到的数学公式。主要包括 Hill 矩阵加密算法中自可逆矩阵的数学求法以及 Haar 域置乱加密的详细数学步骤。之后还会提及统计分析中所用到的方差、相关性系数以及熵的概念的详细阐述。

第 3 章是算法的实现与分析。本章中会写出本论文所设计的算法的流程，并用算法描述语言来描述之。

第 4 章是加密算法的评估与测试。本章主要会写出通过算法实现的程序的运行得出图表，所设计的算法在统计分析与扩散性测试之下的评估结果。

第 5 章是结论，将根据上一章所得出的数据来得出结果，对已设计的算法做出评鉴，并对全文经过判断、归纳、推理等过程，将研究结果升华成新的总观点。

第 2 章 相关理论基础

本次毕业设计所完成的图像加密算法主要由两大块组成。这两大块都是在经典密码学或者之前的参考文献的基础思路上，加以实现、创新与融合最终得出的，最后再将这两大块融合成为一个共同的混合加密系统。本章对所涉及到的相关理论基础进行阐述，为后续算法设计做理论铺垫。包括 Hill 矩阵灰度加密中觉有实际意义的自可逆矩阵的数学求解，还有另一种算法中的具体的离散小波分解过程。

2.1 Hill 灰度自可逆矩阵

1929 年 Hill 提出了最原始版本的 Hill 矩阵加密算法[12]。Hill 算法的基本思想，是将 l 个明文字母通过线性变换转换为 l 个密文字母，解密只需要做一次逆变换即可。而密钥就是矩阵本身。用式子表示如下：

设明文 $M = m_1 m_2 ... m_l$；

密文 $E(M) = c_1 c_2 \cdots c_l$；

这时，本文不妨设

$$C = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_l \end{bmatrix}, \quad M = \begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_l \end{bmatrix}, \mathrm{K} = \begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1l} \\ k_{21} & k_{22} & \cdots & k_{2l} \\ \vdots & \vdots & \ddots & \vdots \\ k_{l1} & k_{l2} & \cdots & k_{ll} \end{bmatrix} \tag{2.1}$$

这之后，加密公式就是

$$C = KM(\mathrm{modN}) \tag{2.2}$$

解密公式为

$$K^{-1}KM = M = K^{-1}C(\mathrm{mod}\,26) \tag{2.3}$$

也就是

$$M = K^{-1}C(\mathrm{modN}) \tag{2.4}$$

以上所提及的式子，都是最先用于英文字母的加解密，所以其中的 N 都是等于 26。

以上可见，必须 K 有逆矩阵，解密才有可能，所以加解密时主要采用可逆矩阵 K。而众所周知，一个方阵的可逆性取决于其行列式的值，一个实矩阵有逆元，当且仅当它的行列式为 0.以上的所有运算都是在模 26 下进行的。于是，可知，矩阵 K 有模 26 的逆元，当且仅当 K 的行列式与 26 的最大公约数为 1，即 K 的行列式与 26 互质，如此情况下才可以。

如此看来，本算法实则即使一个矩阵的相乘。如果将之应用在图像加解密的算法中，其实十分的方便，只需将图像以矩阵的形式提取出来。矩阵里的每一个数字都代表源图像的一个像素的灰度值。用之前的矩阵作为密钥对此矩阵进行置乱，就得到了加密图像。而解密的过程实则一个求逆矩阵的过程。

上述 Hill 矩阵加密从数学的角度看是颇为简单，但是从实际的应用角度来看，有些地方显得颇为繁琐。一个重要的地方，就是，每一次的解密，都需要一次矩阵的可逆变化。

设 P 为待加密的图像矩阵；

C 为加密后的图像矩阵；

K 是作为密钥的那个置换矩阵。

那么，具体的加解密算法过程如下：

加密过程：

$$C = E_k(\mathrm{P}) = \mathrm{KP} \tag{2.5}$$

解密过程：

$$P = D_k(\mathrm{C}) = \mathrm{K}^{-1}C = K^{-1}KP = P \tag{2.6}$$

式中的密钥 K 是个可逆矩阵，如果设计生成一个满足

$$A = A^{-1}$$

的矩阵 A 作为密钥矩阵 K，则不必再求逆矩阵，这时解密就会方便很多。本文称具有这种性质的矩阵为自可逆矩阵，但是怎么来求这种自可逆的矩阵呢？

设

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1l} \\ a_{21} & a_{22} & \cdots & a_{2l} \\ \vdots & \vdots & \ddots & \vdots \\ a_{l1} & a_{l2} & \cdots & a_{ll} \end{bmatrix} \tag{2.7}$$

为 n×n 自可逆矩阵，其中 n 为偶数，设 n 为 2，则将其写作

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \tag{2.8}$$

式子中，A11、A12、A21、A22 都是(n/2)×(n/2)的矩阵。由 A 的自可逆性可得：

$$A.A = A.A^{-1} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}.\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = I \tag{2.9}$$

由上式可得

$$A_{12}.A_{21} = I - A_{11}^{\ 2} = (I - A_{11})(I + A_{11}) \tag{2.10}$$

由上式可以知道，若$|A_{12}|$ 是$|I-A_{11}^{2}|$ 的一个因子，则$|A_{21}|$ 是另外一个因子，从而存在常数 k，使得：

$$A_{12}=k(I-A_{11}) \tag{2.11}$$

则有：

$$A_{21}=k(I+A_{11}) \tag{2.12}$$

由

$$A.A=A.A^{-1}=\begin{bmatrix}A_{11} & A_{12}\\ A_{21} & A_{22}\end{bmatrix}.\begin{bmatrix}A_{11} & A_{12}\\ A_{21} & A_{22}\end{bmatrix}=I \tag{2.13}$$

还可以得出

$$A_{11}A_{12}+A_{12}A_{22}=0 \tag{2.14}$$

当

$$A_{12}=k(I-A_{11}) \tag{2.15}$$

成立时，A11 和 A12 可交换，即有

$$A_{11}A_{12}=A_{11}.k(I-A_{11})=k(I-A_{11}).A_{11}=A_{12}A_{11} \tag{2.16}$$

将上式代入 $A_{11}A_{12}+A_{12}A_{22}=0$ ，可得：

$$A_{12}A_{11}+A_{12}A_{22}=A_{12}.(A_{11}+A_{22})=0 \tag{2.17}$$

当 $A_{12}\neq 0$ 时，有

$$A_{11}+A_{22}=0 \tag{2.18}$$

综上所述，可以得出自可逆矩阵的生成算法：

① 选择任意的（n/2）×（n/2）矩阵 A_{22}；

② 计算 $A_{11}=-A_{22}$ ；

③ 取 $A_{12}=k(I-A_{11})$ 或 $k(I+A_{11})$ ，其中 k 是一个素数；

④ 计算 $A_{21}=(I+A_{11})/k$ 或 $(I-A_{11})/k$

⑤ 合并成一个完整的矩阵

2.2 DWT 域成对 Logistic 混沌置乱

Gu 和 Han 提出的 DWT 域成对 Logistic 混沌置乱加密方案[11]。

这个加密方案的大致思想，就是，

（1）首先将待加密的图像以矩阵的形式提取出来，与之上的方式相同，矩阵中的每一个数字都代表的是该图像的一个像素的灰度值。

（2）之后将这个图像运用小波进行多层分解，分解的层数由用户指定。本次试验

所用的分解用的小波是提升整数 Haar 小波，这种小波在只进行一层分解之后可以做到无损，这在频域置乱加密算法中是很少见的。即使是 2 层或是 3 层，它的失真也是较低的。该小波的详细介绍会在本章的下一节“基础理论”中详细介绍。以下是该小波的基本波形[13]，仅供参考：

$$\psi_H(t)=\begin{cases}1, & 0\le t<\frac{1}{2}\\ -1, & \frac{1}{2}\le t<1\\ 0, & \text{其他}\end{cases} \tag{2.19}$$

（3）再之后，使用二维混沌映射，以用户给出的(x_0,y_0)作为密钥，生成混沌序列。

混沌，是指发生在确定系统中的貌似随机的不规则运动，一个确定性理论描述的系统，其行为却表现为不确定、不可重复和不可预测，这就是混沌现象。进一步研究表明，混沌壶是非线性动力系统的固有的特有性质，是非线性系统普遍存在的现象。混沌序列的对初始值的敏感依赖性和非周期性正是密码学中密钥和密钥流需要具备的特性。因而混沌映射在信息加密中得到了广泛的应用。

而所谓的二维混沌映射，本文可以从以下两个混沌映射之中进行选择：

第一种，二维成对 Logistic 映射[14][11]，简单且性能优良，公式如下：

$$\begin{cases}x_{n+1}=4\mu_1 x_n(1-x_n)+\gamma y_n\\ y_{n+1}=4\mu_2 y_n(1-y_n)+\gamma x_n\end{cases} \tag{2.20}$$

其中，$\mu_1=\mu_2=0.89$，$\gamma=0.1$。

前述参数都只有在一定范围内才可以使这映射为混沌映射，详细原因与本毕设无关，故不赘述。而在此时，本文所选定的这两个值，就可以让这映射属于混沌映射。

第二种，二维超混沌系统，公式如下[11]：

$$\begin{cases}x_{n+1}=1.55y_n-1.3y_n^{\,2}\\ y_{n+1}=0.1y_n-1.1x_n\end{cases} \tag{2.21}$$

以上两种混沌映射的用法，就是用密钥(x_0,y_0)不断地套用，得出一个足够长度的混沌序列，为以后的加密做准备。本次试验，将使用第二种，即二维超混沌系统作为本次试验的混沌映射。

混沌序列得到之后，通过排序，得到一个原下标序列，这就是本文之后在每一层进行置乱所用的置乱序列。

（4）用置乱序列对每一层的数据进行分块置乱，这样有利于减少失真。之后将各

层合并，并进行小波的逆变换，得到置乱后的图像，也就是加密图像。

2.3 Haar 小波分解

小波分解是个复杂的过程，先从一维 Haar 小波变换开始说起。

一维 Haar 小波变换的具体步骤是[15][13]：

① 输入信号集 S_n，它由 2^n 个样本值 $S_{n,l}$ 组成，记为 $S_n=\left\{s_{n,l}\middle|0\le l\le 2^n-1\right\}$ ；

② 将信号分成 $a=s_{n,2l}$,$b=s_{n,2l+1}(l=0,1\ldots,2^{n-1}-1)$,共有 2^{n-1} 对，然后对每一对样本进行变换，得到均值集 S_{n-1} 和差值集 D_{n-1}，即

$$\begin{cases} s_{n-1,l}=\dfrac{1}{2}(s_{n,2l+1}+s_{n,2l}) \\ d_{n-1,l}=s_{n,2l+1}-s_{n,2l} \end{cases} \tag{2.22}$$

③ 对 S_{n-1} 再进行第二步的变换，得到均值集 S_{n-2} 和差值集 D_{n-2}，它们各有 2^{n-2} 个样本；

④ 重复上述的步骤，直到 S_0 中只有一个 $s_{0,0}$ 为止。

整个分解的过程如图 2.1 所示：

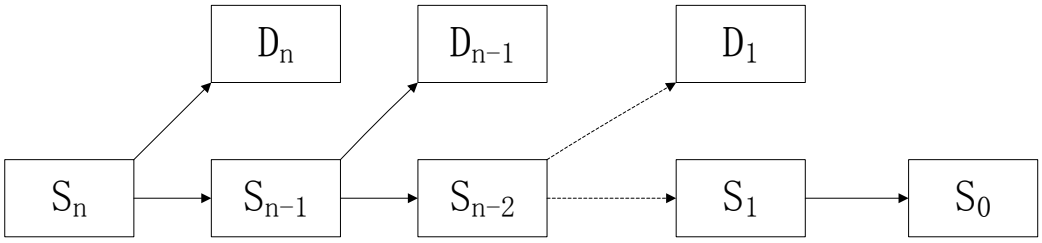

图 2.1 Haar 小波分解过程

以上就是一维的 Haar 小波分解，但是本实验牵扯图像的加解密，一维的显然是不够用。二维的 Haar 小波变换是通过两次一维 Haar 小波变换得到的。先对每一列进行一维小波变换，这个叫列变换，然后对列变换后的矩阵每一行进行一维小波变换，这个叫行变换。通过列变换和行变换得到的就是二维的 Haar 小波变换的结果。

2.4 本章小结

本章对 Hill 矩阵加密的经典矩阵相乘的置乱算法进行了详细的说明，并在此基础上引入了自可逆矩阵的概念。在这一部分，详细的解释了自可逆矩阵的求解过程。自可逆矩阵的加入，使这种 Hill 矩阵加密的算法实现变得更方便实现，也更高效。

与此同时，本章第二节对 Haar 域置乱加密的数学流程作了较为详细地阐述和说明，本次试验在图像的分解时使用的是提升整数 Haar 小波，而在每个层次的以块为单位的

置乱之中使用的是二维超混沌系统，这个组合可以很大程度地减少图像的失真。

总之，本章主要对本次毕业设计所设计的算法所牵扯到的基本定义、概念、公式以及一些重要的定理的推导进行了详细的说明，为本次论文在理论层次上奠定了基础。

第 3 章　算法设计与实现

上一章，对于本次试验所涉及的基础知识与公式进行了详细的说明。本章，会详细地介绍本次算法设计与测试评估的具体实现。本次算法实现所使用的编程语言是 C#。

3.1　算法整体描述

本次试验所设计的是一个图像加解密系统，当然其中也自带该算法的统计分析和扩散性测试与评估的功能。不过算法的测试与评估相对于主题程序较为独立，故具体的实现过程将在本章的第三节将会详细叙述。

本图像加解密系统主体部分的流程如图 3.1 所示。

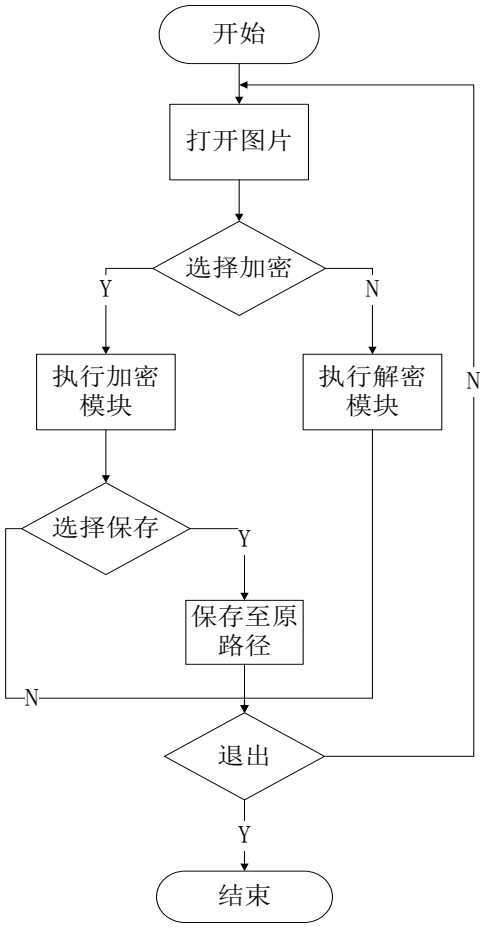

图 3.1 主体流程图

主体部分的步骤形式的算法描述如下：

初始化：二维数组 P，二维数组 E1，二维数组 E2

① 以矩阵的形式打开指定位置的图片，将之赋值给 P，图片必须是 256×256 的 bmp

类型的；

② 选择是要加密还是要解密，加密执行步骤 3，解密执行步骤 4；

③ 将矩阵 P 代入加密模块，得出加密后的矩阵，赋值于 E1，显示出来。若选择保存，则会保存至打开该图像的地址；不管保存与否，执行步骤 5；

④ 将矩阵 P 代入解密模块，得出解密后的矩阵，赋值于 E2，显示出来，执行步骤 5；

⑤ 若选择退出，则程序结束，否则返回步骤 1.

下面详细介绍上面所说的各个模块的详细实现过程。

3.2 加密算法设计与实现

本图像加密系统的解密算法就是加密算法一个逆过程，并在加密算法的基础之上添加了与原图比较后计算出像素差异的算法，故而，解密算法在下文中就不再赘述，重点介绍加密算法。

加密算法主要分为两个模块，一个是 Hill 矩阵加密模块，另一个是 Haar 域置乱加密模块。加密算法的流程图如图 3.2 所示。

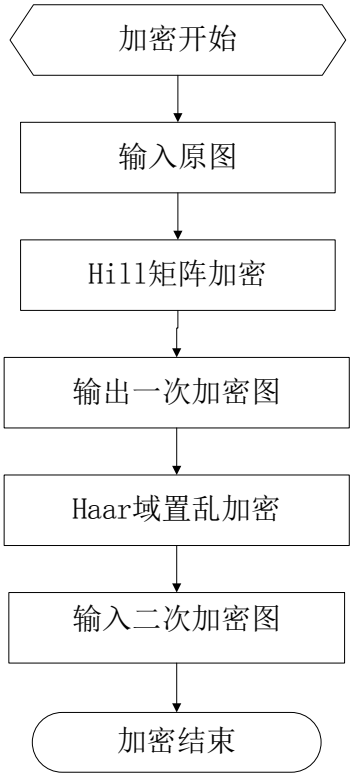

图 3.2 加密流程图

整个加密过程的伪代码如下所示：

```
BEGIN:  //过程开始
(1)if  (iw == N && ih == N )检查所打开的图像是否是265×265的图像
            (1.1) comm.getPixel()  从窗口1读入图像
            (1.2)HillEnc()  进行Hill矩阵加密模块,结果赋值于二维矩阵E
            (1.3) SuperChaosED()  进行Haar域置乱加密模块，结果为E2
            (1.4)comm.showImage()在界面上显示加密之后的图
END  //过程结束
```

整个加密过程大致流程就如上面的伪代码所示，下面，会详细的介绍两种加密算法的具体实现过程。

3.2.1 Hill 矩阵灰度加密算法

3.2.1.1 Hill 矩阵加密算法设计

Hill 矩阵加密算法的步骤形式的算法描述如下：

初始化：二维数组 A，整形变量 m，整型变量 i；

① 系统生成 m×m 自可逆矩阵 A；

② 将原图像分成 m×m 块图像；

③ 将每个图像的第 i 块集合于一个图像块，最后组合成一个临时图块 A_i，共有 m×m 尺寸；

④ 用矩阵 A 对临时图像 A_i 加密；

⑤ 用步骤 4 所得的图像 A_i 进行转置，为 B_i；

⑥ 用矩阵 A 对 B_i 加密；

⑦ 将所得到的放在最后加密所得矩阵的第 i 个位置；

⑧ 重复步骤 4、5、6、7，将所有 A_i 和 B_i 进行处理，得出最后的加密图。

Hill 矩阵加密算法的流程图形式的描述如图 3.3。

Hill 矩阵加密算法的伪代码形式描述如下。

```
BEGIN:  //自可逆 Hill 矩阵加密模块
        (1)输入图像矩阵 int[,] P,和图像大小  int N
        (2) mtrx.keyMatrix()生成自可逆矩阵密钥
        (3) comm.dividBlocks()将图像矩阵分成M×N块
        (4) hillCipher()将块矩阵tem进行hill加密
        (5) mtrx.mtrTrans()转置矩阵
        (6) hillCipher()将矩阵再次进行Hill加密
        (7)comm.combinBlocks()各块归为，形成并输出加密图
END  //算法结束
```

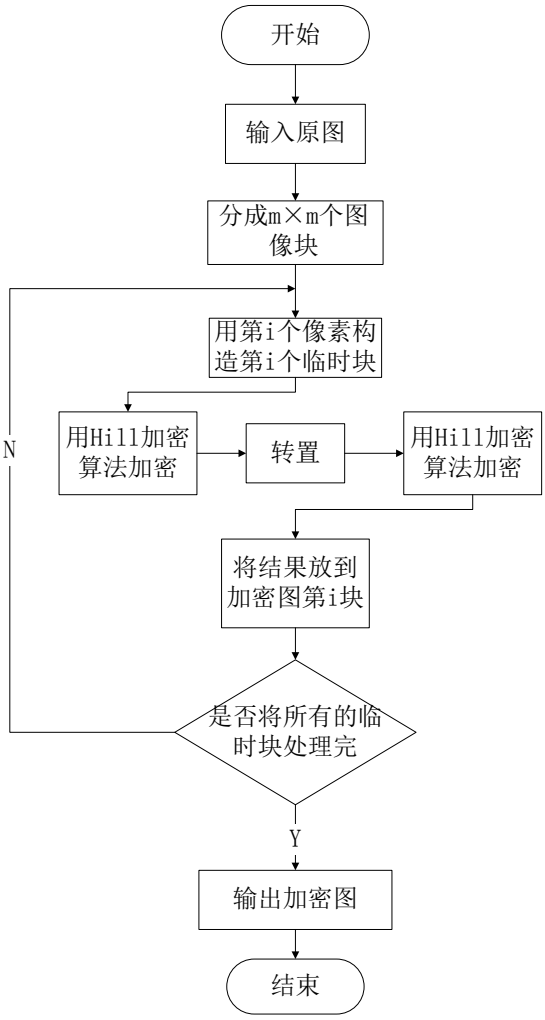

图 3.3 Hill 矩阵加密流程图

Hill 矩阵加密具体计算部分 C#代码实现，包括上述自可逆 Hill 矩阵加密模块的伪代码部分的 hillCipher()函数的具体实现。这个算法模块的主要意义，在于使用自可逆矩阵对导入的图像矩阵进行加密的具体实现，是 Hill 矩阵加解密的核心算法。这个算法伪代码下：

```
BEGIN:// 自可逆密钥矩阵与图像分块矩阵的加密过程
        （1）输入int[][,] tem, int[,] A, int M,和int N
        （2）定义变量int[][,] ctem=new int[N][,]
        （3）for (0<= i < N)
              （3.1）ctem[i]=new int[M,M]
        （4）for (0<= l < N,0<=j<M,0<=i<M)
              （4.1）ctem[l][i, j] = 0
              （4.2）for (0<= p < M)
                    （4.2.1）分块进行矩阵乘法ctem[l][i,j] += (A[p, i] * tem[l][j, p]) % N
              （4.3）ctem[l][i,j] = ctem[l][i, j] % N
        （5） 输出 ctem
END
```

Hill 矩阵加密的主体代码实现，也就是本节前伪代码部分的 HillEnc()函数的具体实现，这部分实现的主要功能就在于完整的进行带有自可逆矩阵的 Hill 矩阵加密算法。它的伪代码如下所示。

```
BEGIN:// 自可逆Hill矩阵加密模块具体实现
        （1） 输入int[,] oP和int N
        （2） 定义变量int M = 16;
              int k = 127;
              int[,] cP=new int[N,N];
        （3） mtrx.keyMatrix()生成密钥int[,] A
        （4） comm.dividBlocks()将原图分块储存在int[][,] tem之中
        （5） hillCipher()加密图像矩阵，结果存在int[][,] ctem中
        （6） mtrx.mtrTrans()转置矩阵，结果放入int[][,] tem中
        （7） hillCipher()加密图像，结果存入ctem中
        （8） comm.combinBlocks()整合成为加密图像，并输出
    END
```

其中，mtrx.keyMatrix(M,k,N)代表生成自可逆矩阵密钥;comm.dividBlocks(oP,M,N)将图像矩阵分成 M×N 块；hillCipher(tem,A,M,N) 将块矩阵 tem 进行 hill 加密;mtrx.mtrTrans(ctem,N,M) 转置矩阵;hillCipher(tem,A,M,N) 将块矩阵 tem 进行 hill 加密;comm.combinBlocks(ctem,M,N)表示各块归位，形成加密图;Hill 矩阵加密的具体实现过程上边已经讲完，但是中间涉及一些算法不得不提，下面会详细说明这些算法。

3.2.1.2 自可逆矩阵生成算法

自可逆矩阵的生成是 Hill 矩阵加密的关键过程，算法设计如下：

初始化：（n/2）×（n/2）的二维数组 A_{22}，空的（n/2）×（n/2）的二维数组 $A_{11}A_{12}A_{21}$，整形的变量 k

① 使用随机算法生成的数值代入矩阵 A_{22};

② 赋值 $A_{11} = -A_{22}$ ；

③ 计算，并赋值 $A_{12} = k(I - A_{11})$ 或 $k(I + A_{11})$ ，其中 k 是一个素数;

④ 计算，并赋值 $A_{21} = (I + A_{11}) / k$ 或 $(I - A_{11}) / k$

⑤ 将步骤 1、2、3、4 中所得的小矩阵合成最终的自可逆密钥矩阵，合并成一个完整的矩阵 。

自可逆矩阵的生成算法大致代码实现如过程 3.5 所示。

3.2.1.3 最终的加密图像合成算法

在 hill 矩阵加密模块的最后，需要将之前经过处理得像素块重新拼接起来，形成一

个完整的加密图像。只要这样，才是一个完整的加密过程，这的算法相对简单，伪代码实现如下所示。

```
BEGIN://混合加密算法
    (1) 输入 int[][,] ctem, int M, int N
    (2) 定义变量 int[,] cP = new int[N, N]
    (3) for (0<=v < M ,0<= u < M ,0<= j < M ,0<= i < M)
        (3.1)cP[u * M + i, v * M + j] = ctem[v * M + u][i, j]将蕴含数组平铺到二维数组之上，得出最终加密图像
    (4)  输出 cP;
END
```

3.2.1.4 像素差异检测系统

本文之前提到过，Hill 加解密模块之中，解密算法只比加密算法多出来一个比较系统。这个比较系统的作用，是为了比较解密之后的图像相对于原图像的像素改变了多少，以此来确定本文所设计的算法的准确性。这个算法的实现其实就是比较下各个位置的像素是否有变化，它的伪代码本文也会给出，这些代码如下所示。

```
BEGIN:// 像素差异计算算法
    (1) 输入string str1, string str2, int[,] img1, int[,] img2, int n
    (2) 定义变量int q = 0为像素差异数量
    (3) for (0<=j < n , 0<=i < n)
        (3.1)if (img1[i, j] != img2[i, j])
            (3.1.1) q++
    (4) MessageBox.Show()显示结果
END
```

如此一来，整个 Hill 矩阵加密模块的实现已经完全的阐述完毕。整体来说，这一模块包括将图像转化为灰度矩阵的模块、自行生成自可逆图像加密矩阵算法模块、密钥矩阵和图像灰度矩阵的相乘的加解密模块、矩阵本身的转置模块以及计算解密图与原图的像素差异的模块。

Hill 矩阵加密系统是本次所涉及的混合加解密系统中的重要内容，对整个加密系统的性能的提高有着重要的意义。

3.2.2 Haar 域置乱加密算法

3.2.2.1 Haar 域置乱加密算法设计

Haar 域置乱图像加密模块涉及小波分解和频域变换，相对较为复杂，本节将会详细

的讲述本模块的实现。需要一提的是，之前本文提到过二维 haar 小波的分解的常规步骤，但是，由于常规的 Haar 小波变换中的类似 $S_{n-1,l}=\frac{1}{2}(S_{n,2l+1}+S_{n,2l})$ 的变换可能会产生非整数，进而使加解密的图像出现一定程度地失真。因此，本文选择在此基础之上，使用二维提升 Haar 小波分解，它的具体的步骤会在本节之后详细说明。

Haar 域置乱加密算法的步骤形式算法描述：

初始化：整型变量 L，整型变量 x，整型变量 y，蕴含二维整形数组 h[,][,]，蕴含二维整形数组 s[,][,]，待加密的由上一模块得到的二维矩阵 P[,]，二维数组 H[,]，一维数组 a[]，一维数组 b[]，一维数组 a0[]，一维数组 b0[]；

① 用户输入整数，为 L 赋值，确定欲分解的层数，0<L<4；

② 将待加密的 P 进行 L 层二维提升小波变换，赋值给 H；

③ 将二维的 H 转化为四维的表示形式 h；

④ 用户输入密钥 x，y；

⑤ 对步骤 4 得到的密钥进行二维超混沌系统的运算，形成两个长度为图片像素长度的混沌序列 a，b；

⑥ 将 a，b 各自进行排序，形成两个原下标序列 a0，b0；

⑦ 用 a0b0 对 h 每一层的每一块进行置乱，得到 s；

⑧ 将 s 转化为二维的 H；

⑨ 对 H 进行 L 层的二维提升 Haar 小波的重构，得到加密图像。

Haar 域置乱加密算法的流程图形式算法描述如图 3.4 所示。

3.2.2.2 提升小波分解算法设计

与之前一样，先来介绍一下一维提升 Haar 小波的具体分解步骤：设 Split()为分裂算子；P()为预测算子，即由括号中的数所预测的；U()为更新算子；Even 代表信号中的偶数下标；Odd 代表奇数下标；J 为分解层数。

假定原始信号为 c_j，小波分解的结果同一般小波分解一样为 c_{j+1} 和 d_{j+1}，那么他们是这样分解的[16]：

① 分裂

信号按照奇偶性化为 $c_{2l,}$和 c_{2l+1} 两个部分，每个信号集合的长度为原信号的一半。这一步又被称之为懒惰变换，表达式如下：

$$Split(c_j)=Even_{j+1}+Odd_{j+1} \tag{3.1}$$

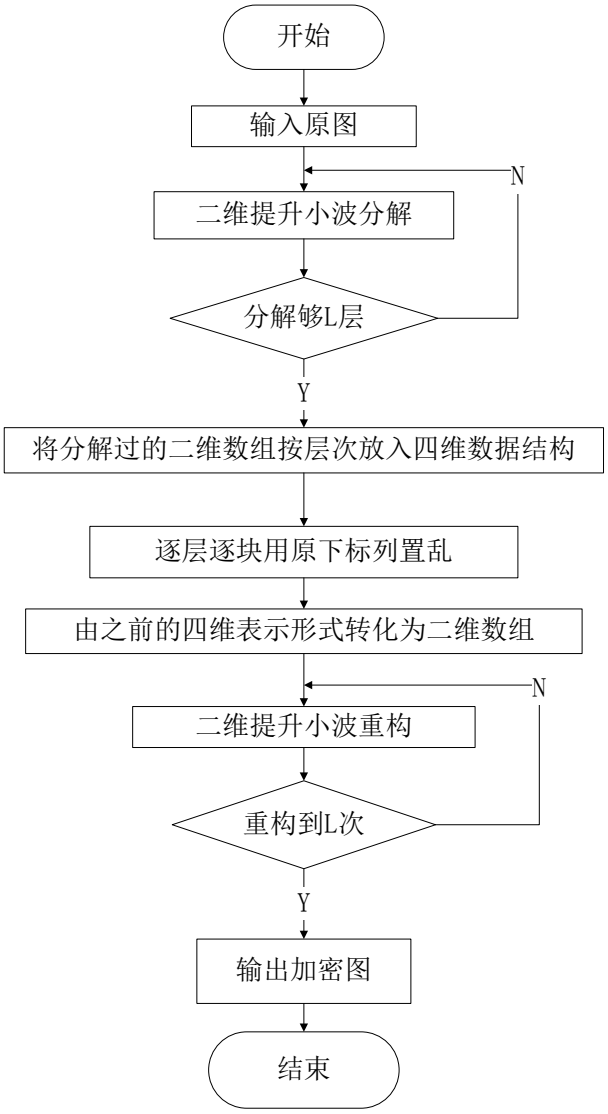

图 3.4 Haar 域置乱加密算法流程图

② 预测

奇数和偶数下标交叉分布在原始信号集之中，根据信号的相关性，可以由奇数下标的点预测偶数下标的点，用 d_{j+1} 表示下表为偶数的点与其预测值的偏差，表达是如下：

$$d_{j+1} = Even_{j+1} - P(Odd_{j+1}) \tag{3.2}$$

本次试验所选的预测算子就是奇数集合本身，也就是

$$P(Odd_{j+1}) = Odd_{j+1} \tag{3.3}$$

③ 更新

用偏差 d_{j+1} 修正奇数下标的点，使其保持原信号的一些特征，这一步称之为主要提升：

$$c_{j+1} = Odd_{j+1} + U(d_{j+1}) \tag{3.4}$$

本次实现中，所选的更新算子为[17]：

$$U(d_{j+1}) = \left\lfloor 0.5 + \frac{1}{2} d_{j+1} \right\rfloor \tag{3.5}$$

常规的提升小波变换主要经历的步骤为分裂、预测和更新，它的简要图示如图 3.5 所示。

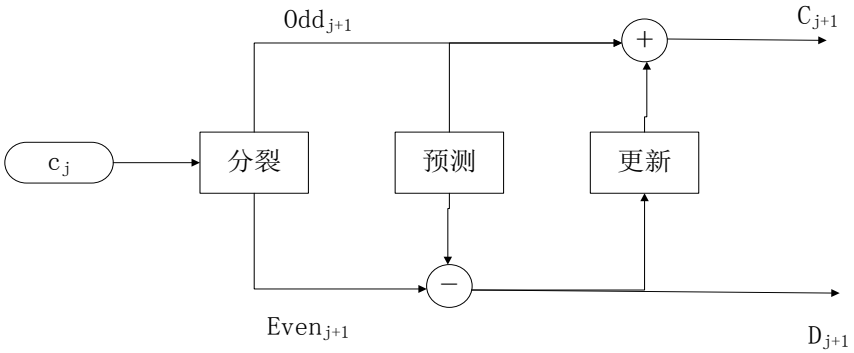

图 3.5 提升框架分解变换

重构过程就是之上分解的逆过程，步骤如下：

① 反更新，表达式为：

$$Odd_{j+1} = c_{j+1} - U(d_{j+1}) \tag{3.6}$$

② 反预测，表达式为：

$$Even_{j+1} = d_{j+1} + P(Odd_{j+1}) \tag{3.7}$$

③ 合并，表达式为：

$$c_j = merge(Even_{j+1} + Odd_{j+1}) \tag{3.8}$$

常规的提升小波变换经历的步骤为反分裂、反预测和合并，，重构变换的框架图示如图 3.6。

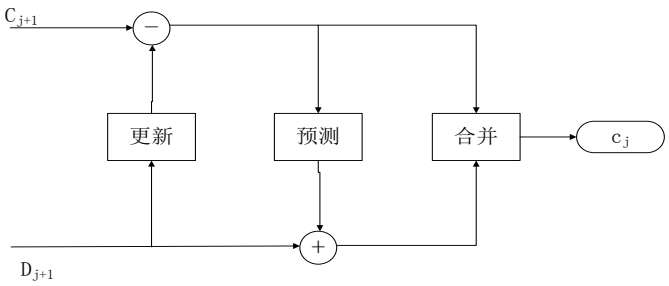

图 3.6 提升框架合并变换

以上本文讨论的都是一维的提升 Haar 小波变换，而本次实验操作的都是二维矩阵，因而有必要提一下二维提升 Haar 小波变换的用法。

二维的提升小波变换是通过两次一维小波变换得到的。先对每一列进行一维小波变换，这个叫列变换，然后对列变换后的矩阵每一行进行一维小波变换，这个叫行变换。通过列变换和行变换得到的就是二维小波变换的结果。这个与基本的 Haar 小波变换类似。

以上，解释了 Haar 域置乱加密模块的核心内容，也就是 Haar 小波变换的步骤，接下来，本文来看一下单次小波分解的详细实现过程。

首先，需要的数据和参数，实现代码所需数据结构如下所示。

```
BEGIN:// Haar 提升小波变换算法
    (1) 输入图像矩阵 int[,] image, int N
    (2) 定义int T = N / 2;
        int[,] f = new int[N, T]行方向分裂的一部分
        int[,] f2 = new int[N, T]行方向分裂的另一部分
        int[,] g = new int[T, N]列方向分裂
        int[,] g2 = new int[T, N]列方向分裂的另一部分
        int[,] hcol = new int[N, T]行方向对奇提升
        int[,] lcol = new int[N, T]行方向主要提升，低频区
        int[,] hrow = new int[T, N]列方向对奇提升
        int[,] lrow = new int[T, N]列方向低频区
        int[,] hwdec = new int[N, N]最终结果
```

从此可以看出，一次小波分解只需要待加密的矩阵和该矩阵（图像）的宽度。

下面本文详细说明一下二维提升小波分解的实现过程。

第一步，行方向分裂:

```
(3) for (0<=i < T, 0<= j < N )
      (3.1)   f[j, i] = image[j, 2 * i]
      (3.2)   f2[j, i] = image[j, 2 * i + 1]
```

第二步，行方向预测:

```
(4)for (0<= k < T ,0<= l < N)
      (4.1) hcol[l, k] = f[l, k] - f2[l, k]得出的是D，对奇数提升
```

第三步，行方向主要提升:

```
(5)for (0 <= m < T, 0<= n < N)
     (5.1) lcol[n, m] = f2[n, m] + (int)Math.Floor(0.5 * (double)hcol[n, m] + 0.5)
```

第四步，一维分解结果赋值于原矩阵:

```
(6)for (0<= j2 < T , 0<= i2 < N)
     (6.1) hwdec[i2, j2] = lcol[i2, j2]
     (6.2) hwdec[i2, T + j2] = hcol[i2, j2]
```

第五步，列方向分裂:

```
(7)for (0<= j3 < N , 0<= i3 < T)
      (7.1) g[i3, j3] = hwdec[2 * i3, j3]
      (7.2) g2[i3, j3] = hwdec[2 * i3 + 1, j3]
```

第六步，列方向预测:

```
(8)for ( 0 <= j4 < N , 0 <= i4 < T)
     (8.1) hrow[i4, j4] = g[i4, j4] - g2[i4, j4]
```

第七步，列方向主要提升:

```
(9)for (0 <= j5 < N , 0 <= i5 < T)
      (9.1)lrow[i5, j5] = g2[i5, j5] + (int)Math.Floor(0.5 * (double)hrow[i5, j5] +
0.5);
```

第八步，另一方向的一维分解结果赋值:

```
(10)for (0 <= j6 < N , 0 <= i6 < T)
    (10.1)      hwdec[i6, j6] = lrow[i6, j6];
    (10.2)      hwdec[T + i6, j6] = hrow[i6, j6];
(11)输出  hwdec;

  END
```

合并的过程其实就是分解过程的逆过程，因而本文中不再给出详细的实现代码。

3.2.2.3 原下标序列的形成

Haar 域置乱加密算法中，起到置乱序列作用的是原下标序列。因而，原下标序列的生成对于整个加密系统来说也是至关重要。原下标序列的生成分为以下 2 个步骤：

第一步，使用二维超混沌序列将输入的密钥进行运算，生成两个长度等于图像长度的数组。主要的步骤包括：1、接受外界输入的密钥 x、y，以他们为参数进行二维超混沌映射运算，得出新的 x_i 和 y_i；2、将这些 x_i 和 y_i 放入各自的准备好的数组中，直到达到要求的长度，最终得到两个长度相同的数组。

第二步，对第一步的两个混沌序列进行处理，得到原下标序列。主要步骤包括：1、对第一步得到的两个数组各自进行从大到小排序；2、用他们排序后的数组的下标作为原下标序列，即本加密算法的置乱序列。

经过以上的两步，本文得到了本加密算法的置乱序列，也就是本算法的核心序列。

3.2.2.4 分层分块的置乱

在经过上一步的原下标序列生成之后，就可以进入 Haar 域置乱加密的最核心的一步，即，对小波分解之后的矩阵进行分层分块的置乱，以获得加密后的分块矩阵。本节主要介绍对一块分解后的矩阵进行 Haar 置乱的实现过程。

本步骤的实现思路，主要在于将两个原下标序列的数值，按次序地做待置乱的块的

点作下标，并将这些点转移到下表所指示的点的位置处，从而完成原图像 Haar 小波分解之后的每一层每一块的置乱，从而达到加密的效果。一块数据的置乱的伪代码如下所示。

```
BEGIN:
        （1）  输入int[,] pix, int L, int[] xind, int[] yind, int pm
        （2）  定义变量int[,] rpix = new int[L, L]
        （3）  for (0 <= i < L)
            (3.1) 定义变量int v = yind[i];
            (3.2)  for (int j = 0; j < L; j++)
                (3.2.1)  int u = xind[j];
                (3.2.2)  if (pm == 1)
                    (3.2.2.1)  rpix[u, v] = pix[j, i];
                (3.2.3)  Else if (pm == -1)
                    (3.2.3.1)  rpix[j, i] = pix[u, v];
        （4）  输出 rpix;
END
```

代码中的 int[] xind 和 nt[] yind 都是由上一步所获得的原下标序列，即为这里的置乱序列。至于分层，这个可以由每块置乱的函数所在的循环实现，但是需要注意的就是每一层之间，待置乱的矩阵的宽度的大小会随着分解层数的增加而以两倍的速度缩小，因而在置乱的时候需要将矩阵的宽度的动态变化处理好。置乱每深入一层，所使用的 M 参数就应该缩小为原先的二分之一，这样才能保证置乱的准确性。

3.2.2.5 数据结构的分与合

在最初得到 L 层小波分解后的二维矩阵时，如果直接用这个二维矩阵进行分块置乱，在实现上会变得很是繁琐。因此，本次试验采用的是将之转化为四维数据结构的方式进行保存，之后在此四维结构上进行置乱操作。

这种四维数据结构样例为：int[,][,] s = new int[Lev, 4][,]。转换为这种蕴含数组的格式之后，其中的第一个二维数组中，Lev 表示该块所属的层数，另一个参数则表示的是该块的位置，即，用 0,1,2,3 分别表示本层中的左上, 右上, 左下, 右下块；而被蕴含的二维数组，则表示具体的数据块。

转换为这种格式之后，我们可以直接通过操纵蕴含数组下标的方式对小波分解后的图像矩阵进行逐层逐块的操作。这样做的好处，在于可以省去繁琐的矩阵行数查询步骤以及一些列的数据处理过程，并且更加的直观，更加便于理解。进行详细的转化算法的实现过程如下所示。

```
BEGIN:
        (1)输入int[,] H, int Lev, int N
        (2)for (0 <= j < N , 0 <= i < N)
                h[0, 0][i, j] = H[i, j];
        (3)定义变量M = N;
        (4)for (int k = 1; k < Lev; k++)
          (4.1)  M = M / 2;
          (4.2)  for (int j = 0; j < M; j++)
          (4.3)  for (int i = 0; i < M; i++)
                (4.3.1)     h[k, 0][i, j] = h[k - 1, 0][i, j];          //LL双低频区
                (4.3.2)     h[k, 1][i, j] = h[k - 1, 0][i, j + M];      //LH半高频区
                (4.3.3)     h[k, 2][i, j] = h[k - 1, 0][i + M, j];      //HL另一个半高频区
                (4.3.4)     h[k, 3][i, j] = h[k - 1, 0][i + M, j + M];  //HH双高频区
        (5) 输出 h;

END
```

这是一部分代码的伪代码，其中 H 是已经被小波分解后得到的二维数组，h 是将要从 H 转化而来的四维数据结构。

3.3　解密算法设计与实现

本图像加密系统的解密算法的过程整体来说就是加密算法的逆过程，大致的流程就是在获取被加密的图像之后，分别对之先后进行 Haar 域置乱解密算法和 Hill 矩阵解密算法。而这两种所谓的解密算法，其实就是 Haar 域置乱加密算法和 Hill 矩阵加密算法的逆过程。解密算法的流程图如图 3.7 所示。

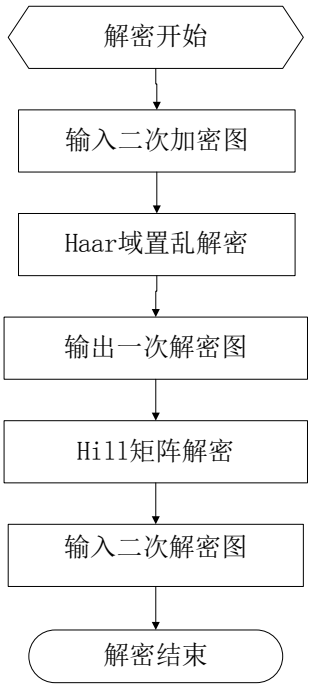

图 3.7 解密流程图

由于解密算法与加密算法类似，就是加密算法的逆过程，因此所有解密算法在此不再赘述。到此为止，这个图像加密系统的加解密模块的具体实现思路和过程都已经描述完毕，下面本文将叙述一下测试模块的具体实现。

3.4 本章小结

本章主要详细叙述了本加密系统的详细实现过程，而这之中又主要包括自可逆 Hill 矩阵加密和 Haar 域置乱加密的具体实现步骤。

自可逆 Hill 矩阵加密算法的叙述中，本文主要介绍了 Hill 矩阵加密的主体实现过程，也就是图像的引入、与加密矩阵的相乘以及整理成为加密图片的过程。在此基础上，本章又更加深入地对 Hill 矩阵加密的一些关键实现步骤的实现进行了详细阐述。这里主要包括自可逆矩阵的生成，这是本文所提及的 Hill 矩阵加密算法的核心步骤，主要是用来生成用来加密图像的密钥矩阵。而自可逆矩阵作为密钥，可以大大的简化 Hill 矩阵加密的执行步骤，并且不会影响它的具体性能。

Haar 域置乱加密的叙述之中，本章着重介绍了本加密算法的详细实现过程和其中涉及的提升 Haar 小波的原理与实现过程。Haar 域置乱加密的具体步骤，就是对一对所给出的密钥执行二维超混沌映射，并通过排序得出置乱序列，再用此序列对进行了指定层数的 Haar 小波分解的图像矩阵逐层逐块进行置乱加密的过程。而这之中，最重要的就是 Haar 小波的分解，本文使用的提升 Haar 小波的分解方式，可以减少图像还原后的失真情况。

第 4 章 加密评估与测试

4.1 实验环境与成果预期

4.1.1 实验环境与配置

本次实验所使用的开发环境和机器配置

（1） 硬件配置：

电脑型号：Acer Aspire 4750g；

处理器：Inter(R) Core(TM) i5-2410M CPU @2.30GHz 2.30GHz；

安装内存：4.00GB（3.85GB 可用）。

（2） 操作系统： Windows7 旗舰版。

（3） 开发工具： Visual Studio 2008。

（4） 开发语言： C#；.net Framework 3.5。

4.1.2 成果预期

将基于 Hill 矩阵与 Haar 域序列置乱的图像加密算法进行整合，图像通过“Hill 矩阵”加密后，再进行“Haar 域序列置乱”加密，并形成系统。

之后选取至少两种图像加密与评估的方法（本次选择的是统计分析、扩散性测试）对已制作的加密系统进行基于测试模块的评估与测试，测试本系统对于对抗基于分析统计类攻击和差分攻击的能力。所谓差分攻击，一种针对明文的攻击，其实就是，通过分析特定明文差分对相对应密文差分影响来获得尽可能多的密钥。差分分析涉及带有某种特性的密文对和明文对比较，其中分析者寻找明文有某种差分的密文对。这些差分中有一些有较高的重现概率，差分分析用这些特征来计算可能密钥的概率，最后定为最可能的密钥[18]。

在此基础上形成算法类的论文。

4.2 图像加密评估方法

测试评估模块的存在，是为了检测算法在一些方面的性能，本次试验所采用的是统计分析和扩散性测试两种评估测试方法。

4.2.1 统计分析法

本文对所设计的图像加密算法的评估测试所用的分析统计的方法中，主要使用了 3 种指标，他们分别是：直方图与方差、相邻像素的相关性和图片信息熵的计算。本节，会把对算这些指标所用的算法做出详细的说明。使用统计分析来测试算法可以有效地测出算法在混乱和扩散性方面抵抗统计攻击的能力。

对于许多种加密方法，用统计分析的方法是可能解密的[19]。以为了挫败其基于统计分析的强力攻击，可以采用混沌和扩散两种方法所谓混沌，是指发生在确定性系统中的貌似随机的不规则运动，一个确定性理论描述的系统，其行为却表现为不确定、不可重复和不可预测，这就是混沌现象。所谓扩散性，是加密算法中的一个重要性质，意指当改变一个图像的一个比特的时候，其加密图将以不可预测的方式进行改变。

本实验所涉及的统计分析的方法有三个，他们分别是直方图与方差、相邻像素的相关性系数和图片的信息熵。

4.2.1.1 直方图的描绘和方差的计算

方差表示直方图与其平均值之间的分散程度，直方图一致分布的程度可以由方差来度量，方差值越小表明待检测图像的直方图分布越一致，对于此类明文攻击的抵抗能力越强。

用 histi（i=0,1…，255）来表示图片中像素某一方向（即 RGB 中本实验选定的某一方向）的灰度值，用 aver 来表示所有像素灰度值的平均值，在此基础上可以得出图像方差的计算公式：

$$S = \frac{1}{256}\sum_{i=0}^{255}(\text{hist}_i - \text{aver}) \tag{4.1}$$

而其中的 aver，可以由以下公式算出：

$$aver = \frac{1}{256}\sum_{i=0}^{255} hist_i \tag{4.2}$$

计算方差[11]，需要所有的待测数据所组成的序列以及所有数据的平均值，有了这两点，计算起来较为简单，具体步骤如下：

设 v[]为储存所有待测数据的数组，L 为数据个数，sqsum 为方差

① 计算 v 中所有数据的和；

② 将第一步得出的结果除以 L 得出平均值；

③ 将 v[i]减去第二步得出的结果，所得数据进行平方运算；

④ 将第三步所得数据加在 sqsum 之上；

⑤ 重复第三第四步，直到所有的 v[i]都执行完毕。

当方差计算出来之后，直方图以图像中每个像素为横轴，以像素的灰度值为纵座标。

4.2.1.2 相邻像素的相关性计算

关于相邻像素的相关性分析，本次试验是如此执行的。

相关性系数的测试是为了测试出图像水平相邻、垂直相邻和对角线方向相邻的两个像素的相关性。测试方法是：从图中随机的分别选出水平、垂直和对角线方向的大量的成对像素，然后分别用以下公式计算其相关性：

设 x,y 是图像中两个相邻像素的灰度值；

rxy 是所求的相关性系数；

E（x）是像素 x 灰度值的数学期望；

D（x）是像素 x 灰度值的方差；

Cov（x，y）是像素 x 和 y 的灰度值的协方差；

则相关性系数为：

$$r_{xy} = \frac{\mathrm{cov}(x, y)}{\sqrt{D(x)}\ \sqrt{D(y)}} \tag{4.3}$$

灰度值的数学期望为：

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{4.4}$$

灰度值的方差为：

$$D(x) = \frac{1}{N}\sum_{i=1}^{N} (x_i - E(x))^2 \tag{4.5}$$

灰度值的协方差为：

$$\mathrm{cov}(x, y) = \frac{1}{N}\sum_{i=1}^{N} (x_i - E(x_i))(y_i - E(y_i)) \tag{4.6}$$

为测试垂直相邻，水平相邻和对角线相邻的两个像素的相关性，本文会遵照以下步骤来实现。

首先，从图像中随机地选择 3000 对水平、垂直、对角线方向的相邻点，将它们的像素 RGB 三方向的灰度值放在各自准备好的一维数组里；

其中的 RGB 是本次试验所选择的色彩数据标准，相关性测试本文会用色彩坐标上的三个方向来完成。在这里不得不解释一下，本次试验所使用的像素的灰度值，都是建立在 RGB 系统之上的。RGB 色彩模式是工业界的一种颜色标准，是通过对红(R)、绿(G)、蓝(B)三个颜色通道的变化以及它们相互之间的叠加来得到各式各样的颜色的，RGB 即是代表红、绿、蓝三个通道的颜色，这个标准几乎包括了人类视力所能感知的所有颜色，

是目前运用最广的颜色系统之一[20]。R、G、B 分别代表红、绿、蓝，可以将这三个值理解为颜色空间的 3 个基向量，可以按一定序列组成几乎所有颜色。之前和之后的测试都只需要一个方向的，但是在相关性方面，最好还是将三个方向都给计算一下。

然后，利用公式计算出相关性系数，画出相关性图示。

其中，计算相关性系数分为几个步骤：

① 计算待测的两组数的各自的数学期望；

② 计算各自的方差；

③ 用步骤 1 中得出的结果来计算两组数的协方差；

④ 用前三部的结果计算相关性系数。

4.2.1.3 信息熵的计算

一个图像如果有 L 种灰度值 mi（i=0,1，...L-1），并且各个灰度值出现的频率分别为 $p(\mathrm{m}_i)(\mathrm{i}=0,1,\cdots,\mathrm{L}-1)$ 。根据香农定理，图像的信息量为：

$$H(m)=-\sum_{i=0}^{L-1}p(m_i)\cdot\log_2 p(m_i) \tag{4.7}$$

其中

$$\sum_{i=0}^{L-1}p(m_i)=1 \tag{4.8}$$

从公式可以推导出，当图中各灰度值出现的概率相等时，$H(m_i)$ ，也就是图像的信息熵，可以取到最大值，推导过程如下：

证明：

主要依据在于数学中的幂平均不等式[21]：

$$a_o{}^{p_0}a_1{}^{p_1}\cdots a_{L-1}{}^{p_{L-1}}\le p_0a_0+p_1a_1+\cdots+p_{L-1}a_{L-1} \tag{4.9}$$

式子中，$0<p_i<1(i=0,1,\cdots,L-1)$,并且

$$\sum_{i=0}^{L-1}p_i=1$$

式子（4.9）的等号成立，当且仅当 $a_0=a_1=\cdots a_{L-1}$ 时。

式子（4.7）根据对数的性质，可以变换为：

$$\begin{aligned}H(m)&=\sum_{i=0}^{L-1}\log_2\left(\frac{1}{p(m_i)}\right)^{p(m_i)}\\&=\log_2\left(\frac{1}{p(m_0)}\right)^{p(m_0)}\left(\frac{1}{p(m_1)}\right)^{p(m_1)}\cdots\left(\frac{1}{p(m_{L-1})}\right)^{p(m_{L-1})}\end{aligned} \tag{4.10}$$

对于上式，本文应用幂平均不等式（4.9），可以得出：

$$H(m) \le \log_2\left(p(m_1)\cdot\frac{1}{p(m_1)} + p(m_2)\cdot\frac{1}{p(m_2)} + \cdots + p(m_{L-1})\cdot\frac{1}{p(m_{L-1})} \right) = \log_2 L \tag{4.11}$$

本实验所牵扯的灰度级 L 为 28，即 256.

综上所述，当且仅当

$$p(m_i) = \frac{1}{256}(i = 0,1,\cdots,255) \tag{4.12}$$

时，H（m）可以取到最大值，也就是：

$$H(m) = \log_2 L = 8 \tag{4.13}$$

成立。

从上面推出的结论，本文不难看出，图像灰度分布越是一致，其信息熵越大，对于一个理想图像，其信息熵可以达到最大值 8。而一个有效地加密算法应该使加密图像的信息熵接近于 8。这样，可以有效地抵挡统计分析的攻击的入侵。

具体的计算方法，可以运用数学变形将式(4.1)变换为：

$$H(m) = \frac{\lg N^2}{\lg 2} - \frac{\sum_{i=0}^{L-1} p_a(m_i)\cdot\lg p_a(m_i)}{N^2\cdot\lg 2} \tag{4.14}$$

其中 $P_a(m_i)$ 就是灰度值 m_i $(i = 0,1,\cdots,L-1)$ 在这个图上的个数。这样的变换，会使计算起来更为方便。

如此，图片信息熵求出，输出之后的便是本文变换之后的得出的结果。

4.2.2 扩散性分析法

在一个加密算法之中，扩散，是一个很重要的性质。一个优秀的加密系统应该有良好的扩散性[17]。扩散性的意义在于当原图改变一个像素的时候，其加密图会以不可预测的方式进行改变。图像加密算法的扩散特性指加密图的输出像素应该以非常复杂的方式依赖于原图的输出像素，从而可以抵抗攻击者对算法的分析攻击。

攻击者经常会对原图仅仅修改一个像素，然后观察有什么变化，比较两者的加密图，看看两者有什么关系，这类攻击称之为差分攻击。使用差分攻击的方法，攻击者有可能找到原图与加密图之间一些有意义的关系。但是，如果原图的微小变化可以使加密图在扩散和混沌效应中引起重大的变化，那么差分攻击的效率就会很低。

而扩散性测试，顾名思义，测试的是加密算法的扩散性能。扩散性好的加密算法，加密图的输出像素应该以非常复杂的方式依赖于原图的输入像素，这样可以有效地抵挡

入侵者对算法的分析。

为了测试改变一个像素对于加密图的影响，本文会用两个指标来测量，他们分别是像素改变率（NPCR）和一致平均改变强度(UACI)。

4.2.2.1 计算平均改变率（NPCR）

像素改变率定义如下：

用 C_1C_2 表示两幅加密图像，他们的原图片只有一个像素不同。设 C_1C_2 在(I,j)处的灰度值为 C_1（I,j）C_2（I,j），则有以下的定义[22]：

$$D(i,j)=\begin{cases} D(i,j)=0, & C_1(i,j)=C_2(i,j) \\ D(i,j)=1, & C_1(i,j)\neq C_2(i,j) \end{cases} \tag{4.15}$$

则像素改变率（NPCR）的定义为：

$$NPCR=\frac{\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}D(i,j)}{M\times N}\times 100\% \tag{4.16}$$

式子中，MN 分别为 CC 的宽度和高度，而 NPCR 的意义是计算两幅图像不同像素所占的百分比。

平均改变率的计算主要是比较两个图像的所有像素的灰度值，相同的就过去，不相同的就在计数器中加 1，最后将计数器中的数字除以图像总的像素数，就得出数值了。

4.2.2.2 计算一致平均改变强度(UACI)

一致平均改变强度定义如下：

一致平均改变强度（UACI）的定义为[18]：

$$UACI=\frac{\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}\left|C_1(i,j)-C_2(i,j)\right|}{255\times M\times N}\times 100\% \tag{4.17}$$

这个 UACI 存在的意义在于计算两个图像之间的差异的平均强度。

该算法的也较为简单，直接套用公式计算即可。本测试数据的作用是检测铭文图片做轻微更改后，相应的加密图的像素的变化的强度。

4.3 实验结果

本次测试与评估将主要从五个方面，对本次设计的图像加解密系统进行评估，以测试本加密系统的性能。这五个方面分别是：直方图与方差的统计分析、相邻像素相关性系数的统计分析、图像信息熵的统计分析、图像的像素改变率的计算分析以及一致平均改变强度的计算与分析。这五个方面的数据之中，前三个统计分析所获得的数据对于应对基于统计的密文攻击的能力有测试作用，而后两个扩散性测试之中得到的数据，则是

用来测量本文加密算法应对差分类明文攻击的能力

4.3.1 直方图与方差统计分析

本部分的测试方法，就是将一个 256×256 的图片，和其进行加密所得到的图片，分别进行直方图的绘制和方差的运算，以测量其分布的一致性程度。图 4.1 和图 4.2 就是本次试验用来测试的原始图片。

图 4.1 实验用图 1

图 4.2 实验用图 2

以上的图片都是 256×256 的 bmp 格式的图片。

图 4.3 和图 4.4 是图 4.1 的原图以及其直方图与方差的测试：

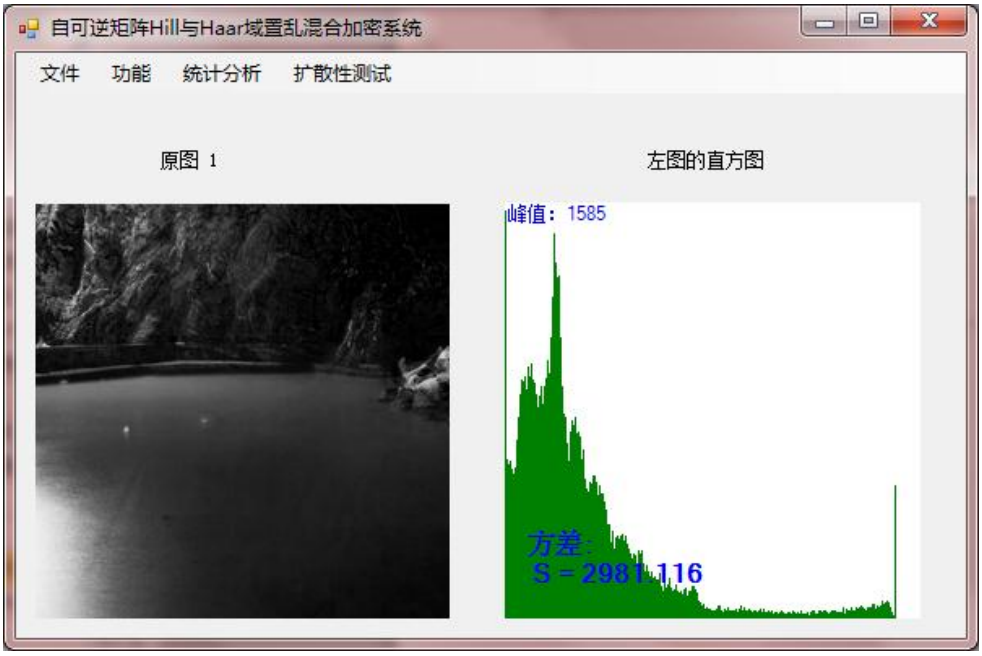

图 4.3 原图直方图

由图 4.3 可以看出，原始图片图 4.1 的方差是 S=2981.116，分布一致性比较差。图 4.4 中的加密所用的密钥为：x=0.311，y=0.722。

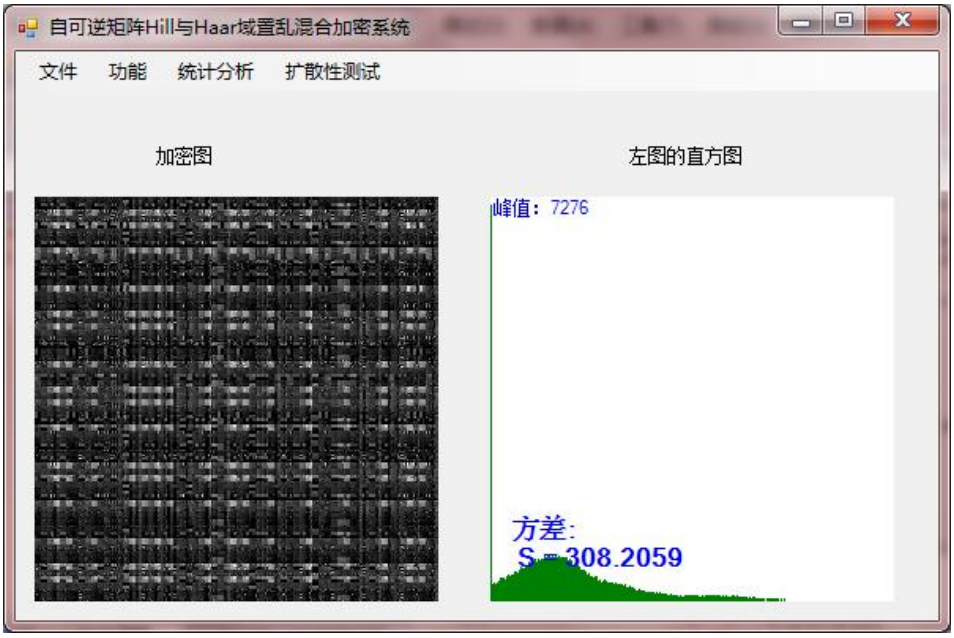

图 4.4 加密直方图

这次加密使用了 3 层 Haar 提升小波分解。所加密的图像方差为 S=308.2059. 由图 4.5 和前几幅图片比较可以看出，整合后的混合图像加密算法加密出的图像的一致性比单纯的 Hill 矩阵加密要好一些。

图 4.5 是图 4.1 的 Hill 矩阵加密的测试与评估。

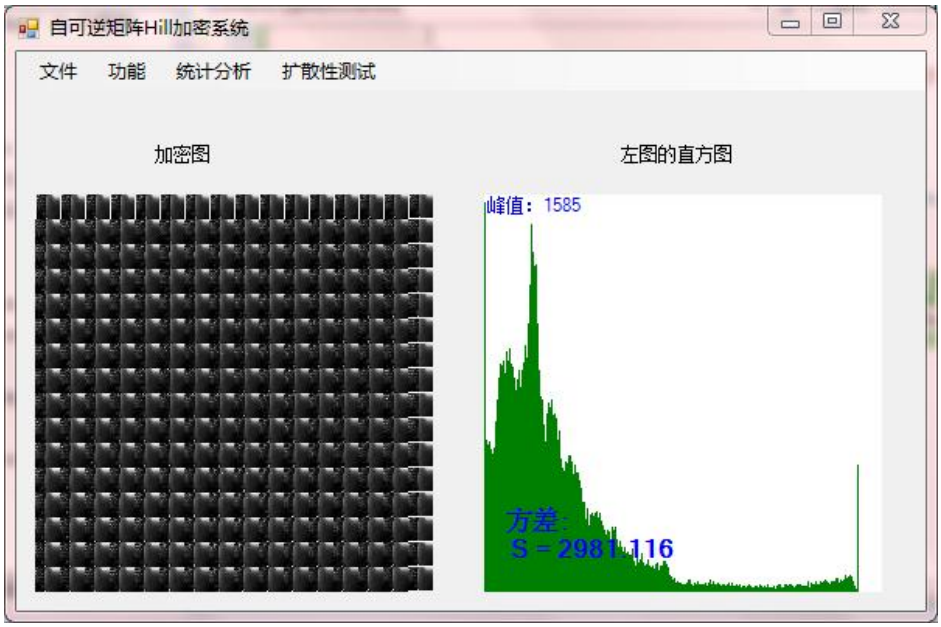

图 4.5 Hill 矩阵直方图

图 4.5 中显示的是，Hill 矩阵加密后的图像的灰度方差为 S=2981.116，与加密之前没有区别。该图像分布一致性也比较差。

4.3.2　相关性统计分析

本部分的测试方法，就是将一个 256×256 的图片，和其进行加密所得到的图片，分别进行水平，竖直，对角线相邻像素的随机选取和相关性的运算，并画出相关性图示，以测量其加密前后的相关性是否分开。

本次的加密所使用的密钥是：x=0.311,y=0.722；选择的分解层数是 2.

图 4.7、图 4.8、图 4.9 和图 4.10 是实验图 4.1 的原图的相关性测量数据，这四幅图片分别是 RGB 系统下的三个方向的水平，竖直，对角线相邻像素的相关性系数、水平方向相邻像素的相关性图示、垂直方向的相邻像素的相关性图示和对角线方向的相邻像素的相关性。

图 4.11、图 4.12、图 4.13 和图 4.14 是实验图 4.1 的加密之后的图像的相关性测量数据，分别是 RGB 系统下的三个方向的水平，竖直，对角线相邻像素的相关性系数、水平方向相邻像素的相关性图示、垂直方向的相邻像素的相关性图示和对角线方向的相邻像素的相关性。

图 4.6 是实验图 4.1 的 Hill 矩阵加密后的相关性系数比较，整体比混合加密系统的的高。从原图和加密图的相关性系数和图示上，本文可以看出图 4.1 的原图和加密图的相关性是比较分开的。

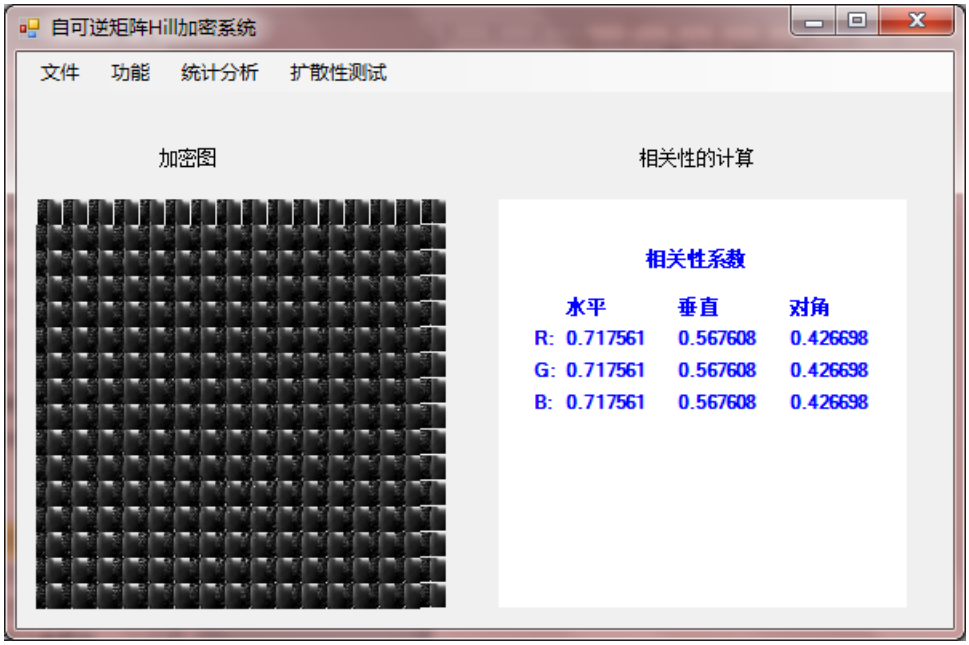

图 4.6 Hill 相关性系数

如图 4.6 所示，原图像经过单纯的 Hill 矩阵加密之后，所得出的加密图由水平方向大量相邻点求出的相关性系数为 0.717561，在 RGB 三色素方向的都是一样的；同样的，垂直方向的相关性系数为 0.567608；对角线方向的相关性系数为 0.426698。相对于原图来说，相关性系数整体有些下降。

图 4.7 原图相关性系数

如图 4.7 所示，从原图像所得出的由水平方向大量相邻点求出的相关性系数为 0.923629，在 RGB 三色素方向的都是一样的；同样的，垂直方向的相关性系数为 0.95286；对角线方向的相关性系数为 0.877881。

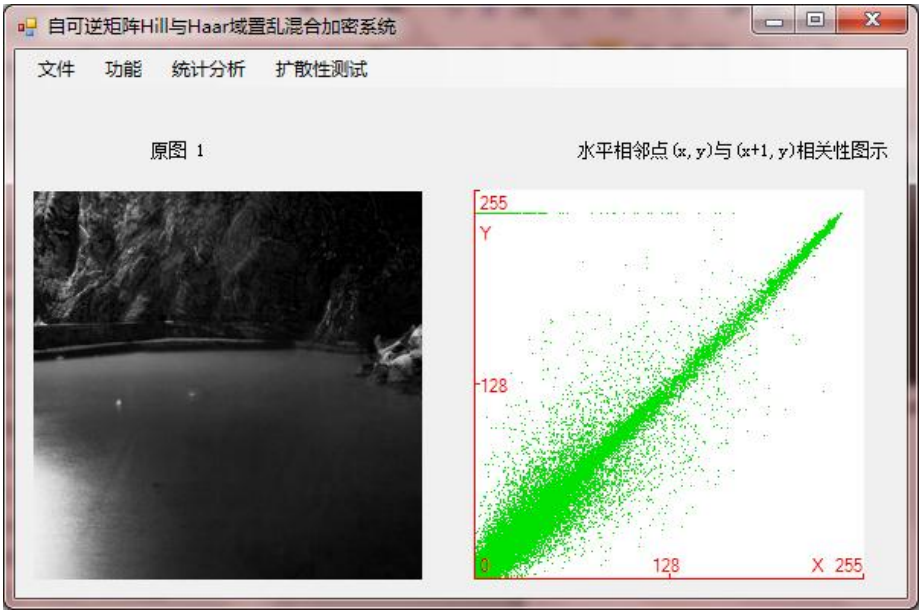

图 4.8 水平相关性图示

图 4.8 所示的是原图 4.1 的水平相邻点的相关性图示。从这里，可以看出，原图的水平相邻像素的相关性还是比较强的。

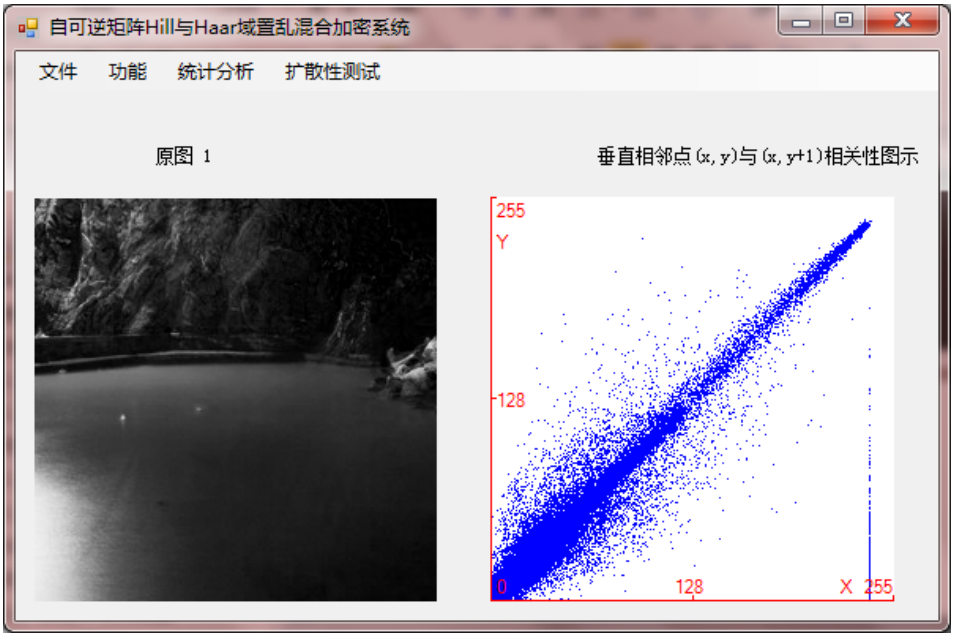

图 4.9 垂直相关性图示

图 4.9 所示的是原图 4.1 的垂直相邻点的相关性图示。从这里，可以看出，原图的垂直的相邻像素的相关性还是比较强的。

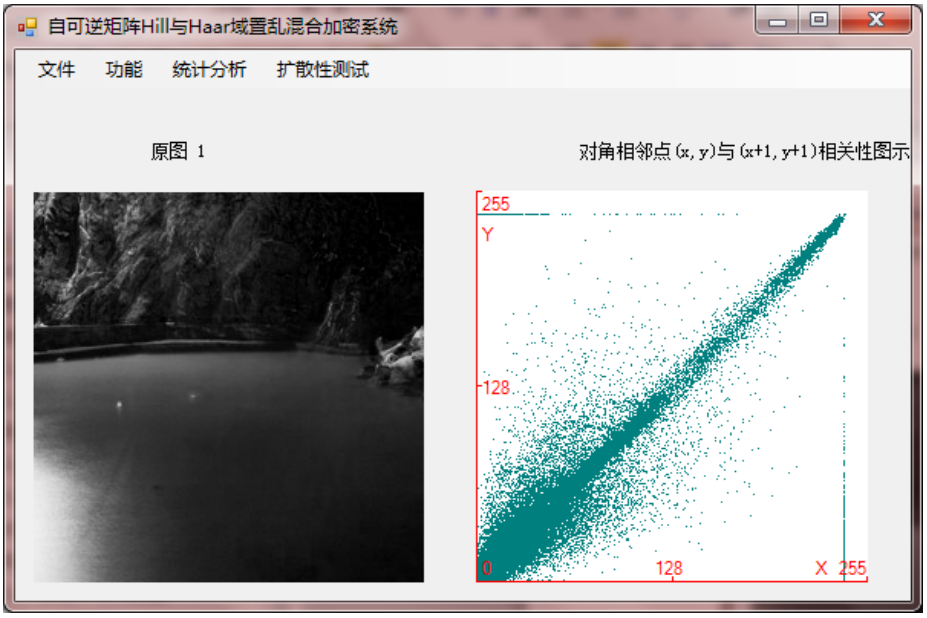

图 4.10 对角线相关性图示

图 4.10 所示的是原图 4.1 的对角线相邻点的相关性图示。从这里，可以看出，原图的对角线方向的相邻像素的相关性还是比较强的。

图 4.11 加密图相关性系数

如图 4.11 所示，本文的自可逆矩阵 Hill 矩阵加密和 haar 域置乱加密所得出的由水平方向大量相邻点求出的相关性系数为 0.564207，在 RGB 三色素方向的都是一样的；同样的，垂直方向的相关性系数为 0.515496；对角线方向的相关性系数为 0.321969。

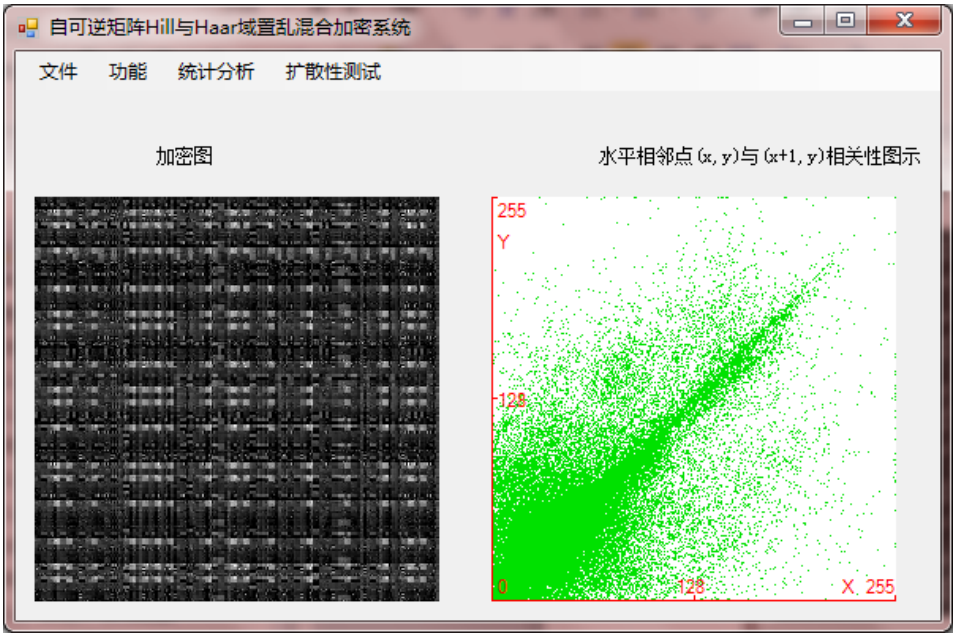

图 4.12 加密图水平相关性图示

图 4.12 所示的是原图 4.1 的加密后的图像水平相邻点的相关性图示。从这里，可以看出，加密后的水平方向的相邻像素的相关性减弱了很多。

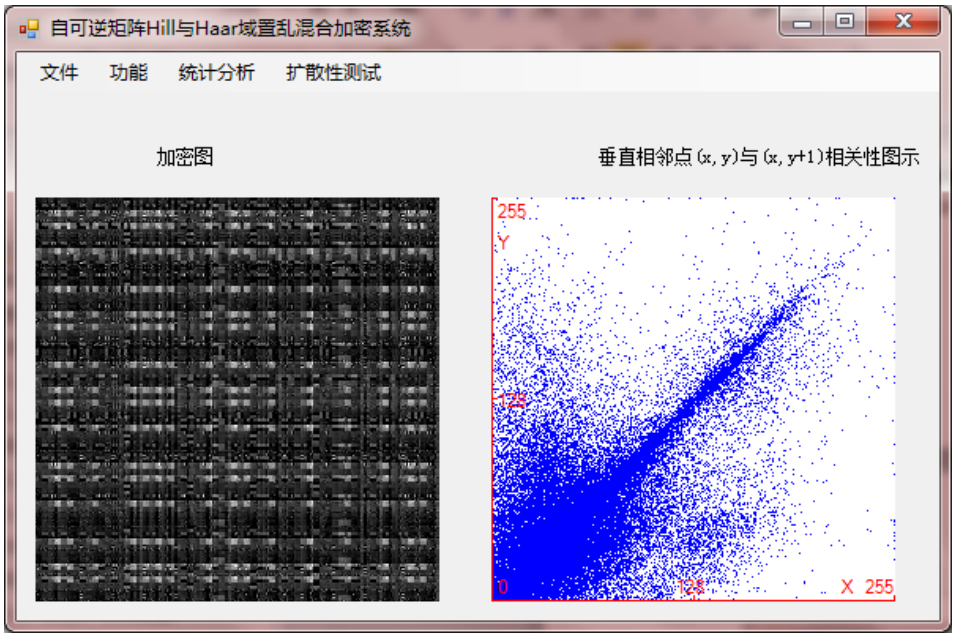

图 4.13 加密图垂直相关性图示

图 4.13 所示的是原图 4.1 的加密后的图像垂直相邻点的相关性图示。从这里，可以看出，加密后的垂直方向的相邻像素的相关性减弱了很多。

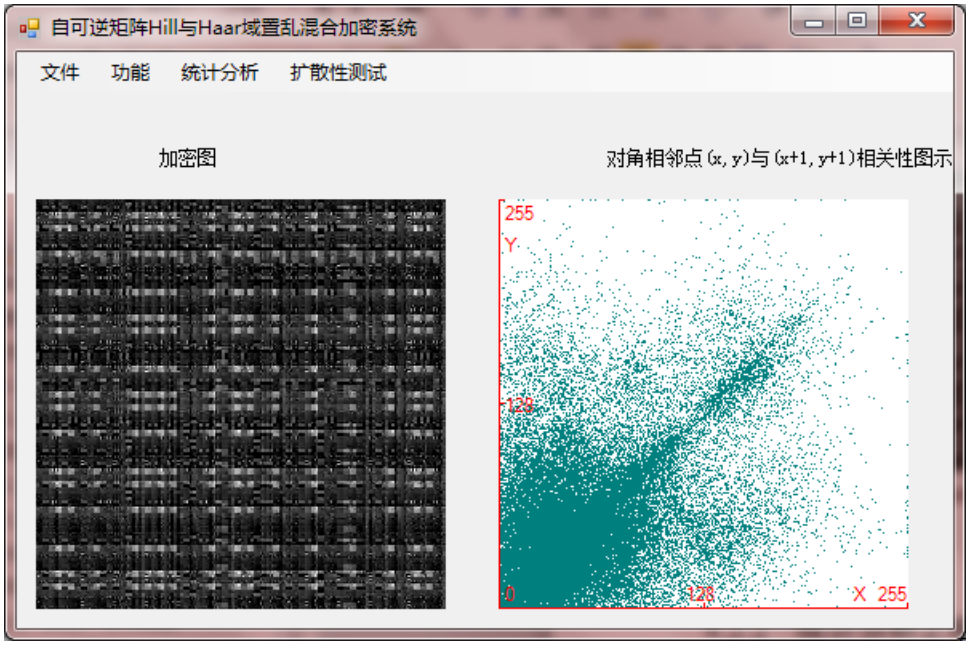

图 4.14 加密图对角相关性图示

图 4.14 所示的是原图 4.1 的加密后的图像对角位置相邻点的相关性图示。从这里，可以看出，加密后的对角位置方向的相邻像素的相关性减弱了很多。

4.3.3　图像信息熵统计分析

图像的信息熵可以度量一幅图像的所包含的信息的多少。一幅图像的信息熵如果越大，它的像素的灰度分布越一致，一个理想的图片，它的信息熵应该无限接近于 8。

之前在第二章时，本文已经证明过，图像信息熵的公式所得的结果是趋近于 8 的，因而，一个有效的加密算法，它的加密图像的信息熵应该接近于 8.

图 4.15 和图 4.16 分别是图 4.2 的原图和加密之后的图的图像信息熵的计算结果，它们都被列在图的右边。

本次的加密所使用的密钥是：x=0.311,y=0.722；选择的分解层数是 2.

从图 4.15 可以看出，原始图像的图像熵是 H=7.756689；而加密之后的图像熵变为了 H=7.678362。

经过测试，原始图像经过 Hill 矩阵加密之后，信息熵为 H=7.756689；

大致来看，本加密系统的信息熵也比较接近于 8，所以也算是比较有效的图像加密系统的。但是加密之后的图像信息熵比之前的图像信息熵数值要小，这说明了本图像系统在加密图片的过程中可能会造成微量图片信息的丢失。

图 4.15 原图信息熵

从图 4.15 可以看出，原图的图像信息熵为 7.756689.

图 4.16 加密图信息熵

从图 4.16 可以看出，加密后的图像的信息熵为 7.678362。相对于原图的信息熵有些下降。

4.3.4 扩散性测试：像素改变率

在扩散性测试之中，像素改变率（NPCR）的测量方式，就是将原图中的一个像素改变，得到一个伪原图，然后将原图和伪原图进行加密，测量加密后的两个图像的不同的像素点的所占的百分比。

本文依然使用图 4.2 进行测试。本次加密使用的密钥是：x=0.311，y=0.722；选的加密层数是 1 层。

经过测试，本算法的一次 NPCR=0.00152587890625%

4.3.5 扩散性测试：一致平均改变强度

在扩散性测试之中，一致平均改变强度（UACI）的测量方式，就是将原图中的一个像素改变，得到一个伪原图，然后将原图和伪原图进行加密，测量加密后的两个图像的不同的像素点的差异强度的平均值。

本文依然使用图 4.2 进行测试。本次加密使用的密钥是：x=0.311，y=0.722；选的加密层数是 1 层。

经过测试，本算法的一次 UACI= 2.99×10^{-5} %

4.3 实验结果分析

本次试验主要使用了两种测试与评估的方法，运用五种指标对本文中的图像加密系统进行了测试与评估，最终得出了实验的结果。

4.3.1 统计分析测试评估结果

在统计分析的测试与评估之中，本文中的 Hill 矩阵与 Haar 域序列混合加解密系统表现还不错。

在原图和加密图的方差计算和直方图的绘画之中，本文可以清楚地看到，两个图片在加密之前的方差很高，一致性并不是太好；但是在加密之后，它们的一致性有了很显著的提高，方差降低了都将近有 10 倍。同时再与单纯的 Hill 矩阵加密相比，加密后的方差也低好多，证明混合系统加密后一致性略强于单纯 Hill 矩阵加密。

这是一个较为不错的数据，加密图像的分布较为一致，他们抵抗此类统计类攻击有较强的抵抗能力。

在原图和加密图的相邻像素的相关性系数的计算之中，本文也可以从原图和加密图的相关性图示可以看到，加密后与加密之前，相关性图示差异还是比较大的；如果从计算出来的数据上看，可以轻易的看出相关性系数的差异。加密之后的图像的相关性，无论哪个方向，都降低了两倍甚至 3 倍。Hill 加密后的图像此系数也减少了，但是没有混合系统的低。

由此可见，原图和加密图相关性完全的分开，并且加密图的相关性系数也较低，抵

抗此类统计类攻击能力较强。

在原图和加密图的信息熵的计算之中，本文可以看到，加密后图像的信息熵比较接近于 8.本文曾经在第二章推导过，图像熵是接近于 8 的，而且，加密图像熵接近于 8，也是一个加密算法是否有效的一项指标。因而，加密图像熵接近于 8，从此可以看出本文中的混合加密系统还是比较有效地。

4.3.2 扩散性测试评估结果

在扩散性测试之中，此混合图像加解密系统的表现较为平庸。为了比较本文中的图像加密系统的扩散性能类的参数的大小，本文可以拿另 3 种加密算法的 NPCR 和 UACI 来做比较，从表 4.1 可以看出一些问题。

表 4.1 三种图像加密算法的扩散性能比较

算法列表	**NPCR**	**UACI**
基于置换信息熵的加密算法[23]	0.35	0.12
Huang 的加密算法[24]	9.16e-5	6.22e-6
Ye 的加密算法[25]	6.10e-5	1.60e-5
本文中的加密算法	1.53e-5	2.99e-7

从表格中可以看出，与其他三种加密方法相比，本文中的加密算法的 NPCR 和 UACI 都处于较低的水准，也就是说，这种算法的扩散性能比较弱。因而在对抗差分类明文攻击的时候，本算法会显得较为乏力。

第 5 章 结论

本次毕业设计论文是关于两种加密算法，Hill 矩阵加密和 Haar 域置乱加密，的研究。以及，在此基础之上将两种算法用 C#代码实现、略微改动并整合在一起形成一个完整的图像加密系统，并同时对之进行评估与测试的过程。

本图像加密系统之中的 Hill 矩阵加密，加密和解密的过程实则都是矩阵的乘法运算。加密的时候以置乱矩阵为密钥，进行图像矩阵和密钥矩阵的相乘，得出加密的矩阵，也就是加密图像。而在解密的时候，将密钥矩阵求逆，也到解密矩阵；再将解密矩阵与已加密图像的矩阵相乘，得出解密的图像。

但是在实现的过程中，代码实现在原始的 Hill 矩阵加密的基础之中加入自可逆矩阵的随机生成的实现。自可逆矩阵，顾名思义，就是可逆并且逆矩阵就是自己本身的方阵。将这种矩阵加入传统的 Hill 矩阵加密之中，则加密解密都是同一个矩阵作为密钥，这样一来，该做法不但消除了解密时候对矩阵求逆的繁琐过程，提高了算法效率，而且，为这种算法的使用提供了便利。

至于另一个加密算法模块，Haar 域置乱加密模块，则使用的是 Haar 小波分解加二维混沌映射置乱的流程。先将待加密矩阵进行 k 层的 Haar 小波分解；再利用二维混沌系统将输入的两个整数密钥生成两个序列，并将这两个序列进行排序得出两个下标序列，作为置乱序列；利用置乱序列对分解后的矩阵进行逐行逐列的置乱；最后用小波的分解的逆向过程得出加密图像。

本算法在实现的过程中，小波分解的过程选择了提升二维 Haar 小波分解，减少了加解密过程中的图像损失，使用一层小波分解进行加解密时，可以达到无损。实现中选择的二维映射是二维超混沌系统。

本次试验将以上两个算法进行了整合并实现，形成了一个完整的图像加密系统。并且在此基础上对这个加密系统进行了统计分析测试和扩散性测试。

统计分析测试得出的结果显示，本图像加密系统在加密后的图像一致性分布性上表现不错；加密前后的相邻像素的相关性系数的差异较大，加密图像之后的相邻点相关性系数相比于原图较小；还有，加密后图像可以保持较接近于 8 的信息熵。这些数据与单纯的 Hill 矩阵加密相比，除了信息熵上略有降低，其他方面的表现都比单纯 Hill 矩阵加密要好。从以上提到的测试结果可以得出，本图像加密系统在应对针对密文的统计分析类攻击的时候，相对于单纯 Hill 矩阵加密，其抵抗能力还是比较好的。

而在扩散性测试当中，从第四章最后的表格中比较看出，本图像加密系统的加密图

像的像素改变率和一致平均改变强度都低于正常的水平。由此可见，本图像加密系统在面对差分攻击这类明文攻击时，抵抗能力较差。

以上是本次毕业设计研究的基本内容，本次试验也有一些遗漏问题。

首先，本次试验使用了提升小波分解代替了一般的 Haar 小波分解，一定程度上减少了整个加密系统的对于图像加解密过程中的损失，但是在使用 1 层之上的分解之后，解密后的图像还是会出现一些程度的失真，这个问题暂时没有找到有效地解决方法，有待解决；

其次，本次算法中的测试部分中，关于扩散性测试的系统，对于 NPCR 好 UACI 都只进行了第一层的计算，并未进行持续的迭代计算。也许在本图像加密系统多次迭代的情况下，可以大大的提高像素改变率和一支平均改变强度，这个暂时不得而知；

再次，本次对于加密系统所使用的测试与评估的方法较少，对于本加密系统面对其他的类型的攻击时的抵抗能力，没有测试到。

参考文献

1.李晖,杜永爱. 基于混沌的图像信息安全技术的研究[A].扬州：扬州大学，2010,1

2.俞银燕,汤织. 数字版权保护技术综述[J]. 计算机学报.2005.28（12）：1957-1968

3.张焕国,郝艳军,王丽娜. 数字水印、密码学比较研究[J]. 计算机工程与应用.2003.（9）：63-67.

4.Mattlews R . On the derivation of a chaotic encryption algorithm[J], Cryptologia 1989 8:29-42.

5.Schwartz C.A new graphical method for encryption of computer data[J]. Cryptologia,1991,15(1):43-46.

6.Bourbakis N, Alexopoulos C.Picture data encryption using SCAN pattern[J].Pattern Recongnition,1991,25(6):567-581.

7.黄峰.现代图像加密技术发展概况[A]. 湘潭：湖南工程学院，2007,38-40

8.Wolfram S.Statistical mechanics of cellular automata[J].Rev Mod Phys,1983,55:601-644.

9.Srivastava A.A survey report on different techniques of image enpryption[J].International Journal of Emerging Technology and Advanced Engineering,2012,2(6):163-167.

10. Acharya B, Panigrahy S K, Patra S K, et al. Image Encryption Using Advanced Hill Cipher Algorithm[J]. International Journal on Signal & Image Processing, 2010, 1(1):663-667.

11. Gu G S, Han G Q. The application of chaos and DWT in image scrambling[C]//2005 International Conference on Machine Learning and Cybernetics. 2006: 3729-3733.

12.朱文余,孙琦.计算机密码应用基础[M].北京:科学出版社，2000.

13.倪琳.小波变换与图像处理[M].合肥:中国科学技术大学出版社，2010.

14. Wang X, Shi Q. New type crisis: hysteresis and fractal in coupled logistic map[J]. Chin. J. Appl. Mech, 2005, 4: 501-506.

15. 孙燮华. 数字图像处理—原理与算法[M]. 北京:机械工业出版社,2010.

16. 潘泉. 小波滤波方法及应用[M]. 北京: 清华大学出版社有限公司, 2005.

17. 左超. 小波提升（续）[EB/OL] . http://blog.csdn.net/henhen2002/article/details/6315451.

18. 差分攻击[EB/OL]. http://baike.baidu.com/view/5857754.htm?fr=aladdin

19.Shannon C E. Communication Theory of Secrecy Systems[J]. Bell system technical journal, 1949, 28(4): 656-715.

20. RGB[EB/OL]. http://baike.baidu.com/view/17423.htm?fr=aladdin

21.Beckenbach E F, Bellman R. Inequalities[M]. Berlin:Springer-Verlag,1983.

22.Chen G R, Mao Y B, Chui C K. A symmetric image encryption scheme based on 3D chaotic cat maps[J]. Chaos Solution and Fractals, 2004, 21(3): 749-761.

23. Guang-hui C A O, Kai H, He Y, et al. Algorithm of Image Encryption based on Permutation Information Entropy[C]//proceedings of 2010 3rd International Conference on Computer and Electrical Engineering (ICCEE 2010 no. 2). 2012.

24. C. K. Huang, H. H. Nien. Multi chaotic systems based pixel shuffle for image encryption[J]. Optics Communications, 2009, 282(11): 2123-2127.

25. Guodong Ye. Scrambling encryption algorithm of pixel bit based on chaos map[J]. Pattern Recognition Letters,2010, 31(5): 347-354.

致　谢

感谢东北大学软件学院，为我提供了 4 年学习的平台和机会，本次研究成果与此四年的学习是分不开的。

感谢我的本次毕业设计的指导老师，谭振华老师。本次研究便是在这位老师的悉心指导下完成的，他严肃的科学态度和精益求精的工作作风在本研究的进行过程中给予了我很大的帮助，同时，他也以严谨的治学精神使我在思想上得到了很大的提升。

感谢与我在一起愉快地度过毕设研究的同学们，正是有他们的支持和帮助，我才可以克服一个又一个困难和疑惑，直到本论文顺利完成。

感谢父母，焉得谖草，言树之背，寸草春晖，无以回报，在此祝愿他们永远健康快乐。

最后，再次对所有关心和帮助过我的人表示衷心地感谢！