

The Design and Implementation of HSB Information Hiding Algorithm*

Song Zhang¹, Mingyu Fan²

¹Department of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu

²Research Center of Information Security, University of Electronic Science and Technology of China, Chengdu

Email: cszdzs@189.cn, ff98@163.com

Received: Dec. 20th, 2011; revised: Jan. 10th, 2012; accepted: Jan. 16th, 2012

Abstract: This dissertation first introduces the structure and characteristic of BMP file and then the LSB algorithm. A lossy information hiding algorithm named HSB (Highest Significant Bits) is proposed in order to break the limit of the embedded image by the traditional LSB algorithm. The new algorithm is based on 24-bit BMP image and increases the size of the embedded image, which is applicable to both image and text. The development environment is VS2008 and it demonstrates the correctness and effectiveness of the new algorithm.

Keywords: Information Hiding; HSB; BMP; LSB

HSB 信息隐藏算法设计与实现*

张 淦¹, 范明钰²

电子科技大学计算机科学与工程学院, 成都

电子科技大学信息安全研究中心, 成都

Email: cszdzs@189.cn, ff98@163.com

收稿日期: 2011 年 12 月 20 日; 修回日期: 2012 年 1 月 10 日; 录用日期: 2012 年 1 月 16 日

摘 要: 本文首先介绍 BMP 文件结构及特点, 再介绍 LSB 算法, 针对传统 LSB 算法可嵌入图像大小的限制, 提出了一种有损信息隐藏算法 HSB(Highest Significant Bits), 此算法基于 24 位 BMP 图像文件, 有效的增加了嵌入图像的大小, 分别对于图像型和文字型图像进行实验证明。实验编程环境为 VS2008, 结果证明了此算法的正确性和有效性。

关键词: 信息隐藏; HSB; BMP; LSB

1. 引言

随着计算机技术、信息处理技术和网络通信技术的迅猛发展, 信息隐藏技术越来越受到各方面的重视。信息隐藏^[1,2]是一个集多学科理论与技术于一身的新兴信息安全方法, 简要过程为将一个信息(秘密信息)隐藏在另一个信息(载体)中, 由于在隐藏后外部表现的只是载体的外部特征, 从而在传输过程中, 非法者不知道这个载体中是否隐藏了其他的信息, 这样就不

易对信息进行窃取、修改和破坏。

在图像中应用的信息隐藏技术基本上可分为两大类: 空域法和频域法。空域法是直接改变图像元素的值, 一般是在图像元素的亮度和色带中加入隐藏的内容。频域法是利用某种数学变换, 将图像用频域表示, 通过更改图像的某些频域系数加入秘密信息, 然后再利用反变换来生成隐藏了秘密信息的图像。与频域法相比, 空域的优点为: 算法简单、使用方便以及隐藏信息量大。信息隐藏中所用的载体可以是文字、图像、声音及视频等, 其中, 以图像作为载体对象的

*资助信息: 本文受国家“863”项目资助(项目编号: 2009AA01Z435)。

信息隐藏最为常见。现阶段嵌入量较大的 LSB 算法^[3]是针对载体图像的改写, 秘密数据图像嵌入量有一定的限制, 针对这个不足, 本文提出一种有损压缩、仅嵌入秘密图像高位平面的算法, 有效地增加了可嵌入秘密图像的大小。

2. 理论基础

2.1. BMP 文件结构及特点^[4,5]

BMP(全称 Bitmap)是 Window 操作系统中的标准图像文件格式, 位图表示的是将一幅图像分割成栅格, 栅格的每一点称为像素, 每一个像素具有自己的 RGB 值, 即一幅图像是由一系列像素点构成的点阵。它采用位映射存储格式, 除了图像深度可选以外, 不采用其他任何压缩, 因此, BMP 文件所占用的空间很大。典型的 BMP 图像文件由四部分组成: 文件头、位图信息头、颜色信息和图形数据。24 位 BMP 图像文件的其独特结构特点是:

- 1) 每个文件只能非压缩地存放一幅彩色图像。
- 2) 非图像数据部分由最前面的文件头和位图信息头结构及一些其它字节组成, 有时也称整个非图像数据部分为文件头。在 Windows API 中, 非图像数据部分的文件头结构用一个名为 BITMAPFILEHEADER 的类型描述, 信息头结构用名为 BITINFOHEADER 的类型描述, 它们分别为 14 字节和 40 字节的数据类型, 这两个类型共占 54 字节。由这两个结构的成员值可以得到位图文件的一些重要信息, 其中包含有该位图文件的类型、大小、图像尺寸、每像素点所占的二进制位数等。

3) 从第 55 个字节开始, 是该 BMP 图像文件的像素数据部分, 每连续 3 个字节便描述图像一个像素点的颜色信息, 这 3 个字节分别代表 B(蓝)、G(绿)、R(红)三基色在此像素中的亮度, 若某连续 3 个字节为: FFH, 00H, 00H, 则表示该像素的颜色为纯蓝色。

2.2. LSB 算法

通常定义图像的第 0 个位平面到第 7 个位平面依次为最不important位平面到最重要位平面, 相应的位被称为最低有效位 LSB 和最高有效位 MSB。

对于图 1, 是一幅普通的 BMP 格式图像, 记为原始图, 图 2(a)是保留原图的最高位, 其他位清零后的效果; 图 2(b)保留原图的次高位, 其他位清零后的效



Figure 1. Original graph
图 1. 原始图

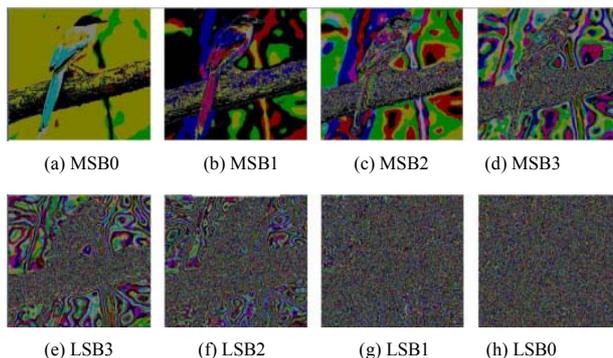


Figure 2. Bitplane clear graph
图 2. 各位平面清零图

果; 直到图 2(h)保留原图的最低位, 其他位清零后的效果。从图 2 可以看出, 随 MSB 到 LSB 的过程中, 图像清晰度逐渐降低, 前两位平面图还可以大概看待图像, 直到最后两位的图像已经完全看不清图像, 由此可以得出结论, MSB 对图像影响最大, 而 LSB 对图像的影响最小, 即位平面越低, 对图像的影响越小, 因此可通过修改低位平面上的位值, 实现在图像中隐藏信息。

LSB 算法^[3-5]是一种基于空域的最低有效位隐藏算法, 将秘密信息嵌入像素值的最低几位比特, 这样对载体的品质影响最小, 可以保证其外部特征不会发生大的变化。信息提取时只需知道嵌入比特数及位置便可以将信息提取出来。由于信息隐藏在最低几位, 只是相当于叠加一个能量微弱的信号, 因而在视觉上很难察觉, 但接收方只要知道秘密信息嵌入的位置就能够提取出隐藏的信息。

根据 RGB 颜色空间的亮度公式 $I = 0.3R + 0.59G + 0.11B$ ^[3]可知, 人眼对绿色最敏感, 对红色次之, 对蓝色最不敏感。因此我们可以在红色和蓝色像素上隐藏较多的信息。因此, 为了提高信息隐藏效率, 可以用待隐藏信息替换每个像素 RGB 亮度分量的不同比特位。实践证明, G 分量改变最低 1 个比特位, R 分量改变最低 2 个比特位, B 分量改变最低 3 个比特位,

这对图像的改变,是人眼不易察觉的。每1个像素(即3个字节: R, G, B)可以利用 $2 + 1 + 3 = 6$ 位比特位,则一个24位的BMP文件的隐藏效率是25%,而直接替换最低1位的LSB算法的隐藏效率只有 $(1 + 1 + 1)/24$, 即 12.5% ^[3]。由于此算法隐藏数据量很大,因此本文的隐藏算法即采用此改进的LSB算法。

2.3. 位平面层数对图像的影响

基于最高位对图像的影响最大的理论,定义位平面最高一位表示为 HSB0, 以此最高八位表示 HSB7(原图像), 从 HSB0 到 HSB7 图像清晰度逐渐增加, 对于对嵌入密文的清晰度要求不高的情况, 可以采用只嵌入密文的前几个位平面, 这样可以节约载体空间以嵌入更多的内容、更大的图片。

对于这个理论, 将于后文实验中进行验证。

3. 算法改进设计

基于以上理论, 提出改进算法 HSB(Highest Significant Bits): 将需隐藏的图像在嵌入载体时, 只嵌入最高几位信息, 提取的时候提取出最高4位(文字型的图片为1位), 其他位置零, 这样可以增大嵌入图像的嵌入量。

假设有需要隐藏的图像 B, 为其大小为 N 字节, 假设只存储 B 图像最高 M 位嵌入到图像 A 中, 传统的 LSB 算法需要加载字节空间为 N, HSB 算法需要字节空间为 $54 + (N - 54) \times M/8$, 大小约为原来的 M/8。

对于图像型的图像, M 取 4, 则需嵌入量为传统的 LSB 算法的 4/8, 对于文字型的图像, M 取 1, 则需嵌入量为传统的 LSB 算法的 1/8, 这样都有效地增加了载体可嵌入图像的大小。

举例: 载体图像 A, 大小为 200×200 (像素), 可嵌入数据量为 30,000 字节, 对于秘密图像 B, 大小为 100×100 (像素), 按传统 LSB 算法, 需嵌入信息量为 30,054 字节, 则不能嵌入图像 A。使用 HSB 算法, 嵌入高 4 位的情况, 需嵌入信息量为 15,054 字节, 可嵌入图像 A。

算法描述: (假设载体图像的图像信息部分的前 10 个字节存储隐写信息的大小)

3.1. 嵌入算法

1) 选择载体图像 A, 得到 A 的大小, 设为 M(字节)。根据 bmp 图像的结构及存储特点, 假设一个

RGB(3 个字节)可以嵌入位数为 R($0 \leq R \leq 24$), 则可以嵌入数据量为 $(M - 64) \times R/24$ (54 个字节的文件头和 10 个字节的隐写信息大小)个字节, 简记为 K。

2) 选择嵌入图片 B, 得到 B 的大小, 假设为 N(字节)。假设需嵌入位平面数为 T($0 \leq T \leq 8$), 根据公式可知需要嵌入保存的数据量为 $54 + (N - 54) \times T/8$ 个字节, 简记为 L。

3) 判断 $K \geq L$, 是则嵌入。载体图像的 55 到 64 个字节嵌入 L。载体图像从 65 字节开始, 嵌入图像 B 图像的前 54 个字节(文件头信息)。B 的图形信息部分, 每个字节只将其高 T 位嵌入图像 A, 循环 L - 54 次。

4) 如果 $K < L$, 不能嵌入。

3.2. 提取算法

1) 打开隐藏信息的图片 C, 定位到第 54 个字节。从第 55 到第 64 共 10 个字节中提取嵌入信息的大小 L。

2) 从第 64 个字节开始, 提取出 54 个字节的信息。根据公式 $(L - 54) \times 8/T$, 计算出循环的次数, 简记为 cnt。循环 cnt 次, 每取出 T 位数据后, 其后 $8 - T$ 位置为 0。

3) 把取出的数据写入文件, 显示秘密图片。

4. 仿真实验与分析

4.1. 图像型

4.1.1. HSB 实验证明

实验试用 VS2008 编程环境, 原始图像如图 1, 图 3 为实验结果, 分别显示 HSB0(其余位置置零)如图 3(a)、HSB3 图 3(b)以及 HSB6 图 3(c)的图像对比。

根据以上实验可以得出结论, 对于精度要求不高的情况, 最高 HSB4 即可显示图像。

4.1.2. 数据隐藏提取实验

信息隐藏与提取过程的界面如图 4 所示, 其中



Figure 3. Impact of the number of bitplane on the graph
图 3. 位平面数对图像的影响

图 4(a)是信息隐藏实验的界面，图 4(b)是信息提取实验的界面。

实验结果如图 5 所示。载体图像如图 5(a)，图 5(b)是待隐藏的图片，图 5(c)是隐藏后的图片，图 5(d)是提取之后的图片。

从实验的结果可以看出，载体图像与嵌入信息后的图像，肉眼是可接受的，提取出的图像与原图像差别也不太大，对于清晰度要求不高的情况完全能满足需要

4.2. 文字图像

对于文字型的图片，图 6 为原始图，图 7(a)是图 6 的 HSB0，图 7(b)为 HSB3，图 7(c)为 HSB6。

从实验结果可以看出，HSB0 就可以分辨出文字，由此将其嵌入载体时就可以根据清晰度需求只嵌入 HSB0 或更多的位平面，在提取的时候将其他位置零即可恢复到相应的 HSB 位面效果，这样有效提高了载体图像嵌入图片的大小。

对于嵌入信息过程与提取过程，与图像型的图片过程一致，这里就不再赘述。

4.3. 进一步思考

- 在图像嵌入时，对于清晰度要求更高的情况，可以再嵌入更高的 HSB。比如最高 5 位、6 位或 7 位，根据 LSB 理论，最低位对图像基本没有影响，所以不需要将所有位平面都嵌入载体。对于文字型的图像亦是如此，并且从实验可以看出，位平面对于文字型的图像的影响没有图像型的大，所以文字型的图像可以嵌入更少的位平面，以此提高利用率。



Figure 4. Information hiding and extracting
图 4. 信息隐藏与提取

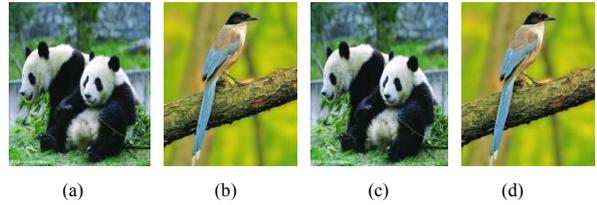


Figure 5. Comparison of the embedded hidden information
图 5. 嵌入隐藏信息的对比

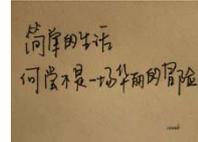


Figure 6. Original graph
图 6. 原始图

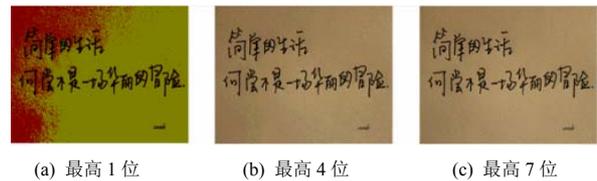


Figure 7. Impact of the number of bitplane on the graph
图 7. 位平面数对图像的影响

- 对于嵌入的信息还可以先进行加密，再嵌入，能有效提高安全性。
- 对于嵌入信息少的情况，可以使用一些编码协议，对信息进行嵌入，以提高抗检测性。

5. 结束语

本文提出的 HSB 算法，通过实验得到了预期的结果，从而验证了算法的正确性与可行性。理论分析和实践表明，该算法具有实际应用的可行性和较好的安全性，优点为嵌入数据量大，隐蔽性好，可变嵌入量等，是一种较高效的信息隐藏算法。缺点为图像秘密图像的清晰度不能保证达到最高，适用范围有一定的局限性。

参考文献 (References)

- [1] T. B. Jack, L. Steven and F. M. Nicholas. Copyright protection for the Eled Ronic distribution of text documents. IEEE, 1999, 87(7): 1181-1196.
- [2] 贾玉珍, 靳冰, 刘琮等. BMP 文件结构的信息隐藏方法与实现[J]. 江西理工大学学报, 2009, 30(1): 42-44.
- [3] 郑皓岚. 改进 LSB 算法以增加 BMP 图像信息隐藏量的方法[J]. 广州城市职业学院学报, 2010, 4(2): 58-61.
- [4] 王少杰, 严伍琼. 基于 BMP 位图的 LSB 信息隐藏算法的实现

- [J]. 电脑开发与应用, 2009, 22(10): 37-40.
- [5] 周文锦, 范明钰, 王光卫. 一种针对 BMP 格式图像的 LSB 数字隐藏方法[J]. 信息安全与通信保密, 2005, 6(7): 253-255.