Computer Science and Application 计算机科学与应用, 2017, 7(10), 913-926
Published Online October 2017 in Hans. http://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2017.710104

Hans汉斯

An Information Hiding Algorithm Based on Pixel Difference and Index Function

Shuyuan Shen[1], Lihong Huang[2*], Songsen Yu[1]

[1]School of Software, South China Normal University, Foshan Guangdong
[2]Changsha University of Science and Technology, Changsha Hunan
Email: ssyuan02@126.com, *lhhuang@csust.edu.cn

Received: Sep. 23rd, 2017; accepted: Oct. 7th, 2017; published: Oct. 13th, 2017

Abstract

In the 2 × 2 pixel block, reference pixel values was randomly determined by the index function, And we calculated the pixel value difference between reference pixel value and the other three pixel value around it. The number of secret data was embedded according to the range of pixel value difference. Using complementary function embedded secret information to effectively reduce the load of image distortion degree. In the process of extracting information, the quantitative pixel values can accurately extract the secret information. Our proposed method improves the Jung and Yoo information hiding algorithm based on index function, and the proposed algorithm has more security.

Keywords

Pixel Value Difference, Index Function, Information Hiding

基于像素差和索引函数的信息隐藏算法

申淑媛[1]，黄立宏[2*]，余松森[1]

[1]华南师范大学，软件学院，广东 佛山
[2]长沙理工大学，湖南 长沙
Email: ssyuan02@126.com, *lhhuang@csust.edu.cn

收稿日期：2017年9月23日；录用日期：2017年10月7日；发布日期：2017年10月13日

摘　要

在2 × 2像素块中，参考像素值由索引函数随机确定，然后计算参考像素与其周围的其它三个像素值的差，

*通讯作者。

文章引用：申淑媛，黄立宏，余松森. 基于像素差和索引函数的信息隐藏算法[J]. 计算机科学与应用，2017，7(10): 913-926. DOI: 10.12677/csa.2017.710104

根据像素差所处的划分范围决定嵌入像素的秘密信息比特数。利用余函数嵌入秘密信息有效地减少载密图像的失真度，在提取信息的过程中，量化后的像素值能准确的提取秘密信息。改进了Jung和Yoo提出的基于索引函数信息隐藏算法，使算法更具有安全性。

关键词

像素差，索引函数，信息隐藏

1. 引言

由于图像的细微改变不会引起人类视觉系统的感知，因此利用图像的视觉冗余，把秘密信息嵌入到图像中，而不影响图像的视觉效果。例如，水印[1]、认证[2]、隐写[3] [4] [5]，水印是为了保护载体的所有权，认证是为了数据的完整性，而隐写是为了秘密信息的传输。

近年来，由于网络的迅速发展，给信息的传输带来了方便，而网络上的图像无处不在，在图像上隐藏秘密信息成了新的热门研究领域。最经典的图像信息隐藏方法之一是基于像素差的信息隐藏算法，基于像素差的信息隐藏算法[6] [7] [8] [9] [10]主要是考虑两个相邻像素的像素差，由于视觉系统对纹理丰富区域的像素的改变不太敏感，因此基于像素差的信息隐藏算法在纹理丰富区域嵌入的信息比平滑区域要多。

最初的基于像素差的信息隐藏算法是由 Wu 和 Tsai [6]于 2003 年提出。然而此算法使得图像的像素值的修改幅度较大，嵌入失真度较大，相对应的载密图像像素差的直方图偏离原始图像的直方图较大。为了改进基于像素差的信息隐藏算法，2008 年，Wang 等人[7]应用模函数在图像的边缘区域嵌入更多的秘密信息(MF-PVD)，不过此算法却使得像素差值直方图在零附近急剧变化，比较容易被基于直方图分析检测到。Yang 等人[10]也于 2008 年提出了一种改进的边缘区域的像素差的信息隐藏算法(AE-PVD)，这种算法优于最初的 PVD 算法。AE-PVD 选择两个像素作为一个嵌入单元，算出两个像素差的绝对值，嵌入的秘密信息比特数由像素差的绝对值决定，像素差的绝对值在嵌入前后属于相同的划分区，才能保证秘密信息的正确提取。与 PVD 隐藏算法相比，MF-PVD 和 AE-PVD 嵌入相同的秘密信息时，能保持更好的图像质量。然而，AE-PVD 算法采用最优像素调整算法(OPAP)嵌入秘密信息，比较容易被卡方分析检测到；另外，AE-PVD 算法引起像素差直方图的异常变化，因此也难以抵抗基于像素差的直方图分析。2010 年，Joo 等人[8]改进了 MF-PVD 算法，能抵抗直方图的分析，不过图像的质量比 AE-PVD 差，不适合于特殊的应用，如军事，医用图像等。同年，Yoo 等人[11]改进了 MF-PVD 算法，考虑图像的局部特征及人类视觉系统，他们首先把图像分成大小为 4 × 4 的像素块，然后获得每个像素块像素的高频部分，由于人类视觉对高频系数的变化不太敏感，所以在高频部分能嵌入更多的秘密信息。可是，他们也是应用 OPAP 算法嵌入信息，卡方分析能检测出图像中是否嵌入秘密信息。

2008 年，Chang 等人[12]提出了基于三个不同方向像素差的信息隐藏算法(TPVD)，此算法不仅提高了嵌入秘密信息的容量而且也能使嵌入信息后的图像具有很好的视觉掩蔽性。此外，Jung 等人[13]提出用多重像素差来估计每个像素的平滑程度。Balasubramanian 等人[14]提出了一种新的像素差信息隐藏算法，此算法考虑八个方向的像素差。Jung 和 Yoo [15]提出了一种基于索引函数，利用像素差嵌入信息的

算法，此算法中的索引函数提高了嵌入秘密信息的安全性，嵌入每个图像的容量也较大。然而，此算法中的索引函数要么所选择的像素块中的参考像素的位置在整个图像中是一样的，要么就得选择一定的存储空间来保存索引函数值。

虽然已经有很多关于像素差信息隐藏算法，但仍有一定的研究空间，考虑如果去提高嵌入容量，减少图像的失真。本文提出了一种基于像素差和索引函数的信息隐藏算法，改进了 Jung 和 Yoo [15]提出的索引函数，使得选择像素块中的参考像素更具有一定的随机性，从而嵌入信息后的图像更具有安全性。

2. 预备知识

把图像分成互不重叠的大小为 $n\times n$ 像素块 $A_t\left(t=1,2,\cdots,N\right)$，其中 N 表示像素块的总块数。每个像素块中含有 $n\times n$ 像素 $F_t\left(i,j\right)$，其中 $i,j\left(0\le i,j\le n-1\right)$ 分别表示像素 $F_t\left(i,j\right)$ 在像素块中所在的行与列。则索引函数定义如下：

$$F_t\left(i,j\right)=\left[n\times\left(i \bmod n\right)+\left(n-1\right)\times\left(j \bmod n\right)\right] \bmod n^2 \tag{1}$$

由于本文信息隐藏算法中参考像素是由索引函数决定的，攻击者如果没有索引函数，很难知道哪个像素是参考像素，信息隐藏算法更具有安全性。确定了参考像素，然后计算其与周围像素的像素差。

例如，$F_t\left(0,0\right),F_t\left(0,1\right),F_t\left(1,0\right),F_t\left(1,1\right)$ 是 2×2 像素块中的四个像素，有

$$F_t\left(0,0\right)=\left[2\times\left(0 \bmod 2\right)+\left(2-1\right)\times\left(0 \bmod 2\right)\right] \bmod 2^2=0$$

$$F_t\left(0,1\right)=\left[2\times\left(0 \bmod 2\right)+\left(2-1\right)\times\left(1 \bmod 2\right)\right] \bmod 2^2=1$$

$$F_t\left(1,0\right)=\left[2\times\left(1 \bmod 2\right)+\left(2-1\right)\times\left(0 \bmod 2\right)\right] \bmod 2^2=2$$

$$F_t\left(1,1\right)=\left[2\times\left(1 \bmod 2\right)+\left(2-1\right)\times\left(1 \bmod 2\right)\right] \bmod 2^2=3$$

通过公式(1)，就能确定每个像素块中哪个像素作为参考像素。将像素值的范围[0，255]划分成 7 个连续子区间 $W_{i,j}$，即

$$W=\left\{W_{i,j}=\left[l_{i,j},u_{i,j}\right]\right\}=\left\{\left[0,3\right],\left[4,7\right],\left[8,15\right],\left[16,31\right],\left[32,63\right],\left[64,127\right],\left[128,255\right]\right\} \tag{2}$$

每个子区间 $W_{i,j}$ 的上、下界、宽度分别为 $l_{i,j},u_{i,j}$ 和 $w_{i,j}$ 其中宽度 $w_{i,j}=u_{i,j}-l_{i,j}+1$。找到像素差的绝对值 $\left|D_t\left(i,j\right)\right|$ 所属的子区间 $W_{i,j}$，如果 $\left|D_t\left(i,j\right)\right|\in\left[l_{i,j},u_{i,j}\right]$。从二进制信息比特流截取 $k_t\left(i,j\right)=\left\lfloor\log_2 w_{i,j}\right\rfloor$ 位秘密信息，把 $k_t\left(i,j\right)$ 的二进制秘密信息比特流转换成十进制下的数 $m_t\left(i,j\right)$。

2.1. 秘密信息嵌入算法

首先由索引函数选择每个子像素块中哪个像素作为参考像素，然后计算参考像素与其周围的其它像素的差，根据像素差所处的范围值确定嵌入秘密信息的长度。详细的嵌入过程如下：

1) 把图像分成互不重叠的大小为 $n\times n$ 的像素块 $A_t\left(t=1,2,\cdots,N\right)$，其中 t 是表示像素块所在的位置。

2) 如果 $F_t\left(i,j\right)=t \bmod n^2$，则选择 $A_t\left(i,j\right)$ 作为参考像素，然后根据下面的公式将参考像素值量化

$$\bar{B}_t=\left\lfloor\frac{B_t}{n^2}\right\rfloor\times n^2 \tag{3}$$

3) 计算每像素块中量化后的参考像素跟其周围像素值的差

$$D_t\left(i,j\right)=A_t\left(i,j\right)-\bar{B}_t \tag{4}$$

$A_t\left(i,j\right)$ 表示第 t 块像素块中的像素值。

4) 若$\left|D_t(i,j)\right|$属于$\left[l_{i,j},u_{i,j}\right]$，则嵌入的秘密信息的比特数为$k_t(i,j)=\log_2\left(u_{i,j}-l_{i,j}+1\right)$。

5) 应用余函数减少像素差的变化，降低图像的失真率。对每个像素差除以$2^{k_t(i,j)}$，得到相应的余数，这样所得到的余数比原来的像素差的值要小，从而降低因嵌入秘密信息引起的图像失真率。具体计算公式如下：

$$R_t(i,j)=F\left(D_t(i,j)\right)=D_t(i,j)\bmod 2^{k_t(i,j)} \tag{5}$$

6) 从二进制比特流中读取$k_t(i,j)$位秘密信息，把其转换成十进制的数$m_t(i,j)$。例如，如果截取的二进制比特流是 101，转换成十进制数为 5。

7) 用秘密信息的值减去余数得新差值

$$H_t(i,j)=m_t(i,j)-R_t(i,j) \tag{6}$$

令

$$G_t(i,j)=2^{k_t(i,j)}-\left|H_t(i,j)\right| \tag{7}$$

其中$m_t(i,j)$是十进制的秘密信息值。

8) 为了嵌入信息后减少对图像的失真，应用下列的改变像素值的最佳嵌入算法：

情况 1：$R_t(i,j)>m_t(i,j)$和$\left|H_t(i,j)\right|\le 2^{k_t(i,j)-1}$和$A_t(i,j)<\overline{B}_t$；

$$A_t'(i,j)=\begin{cases}A_t(i,j)+H_t(i,j), & \text{如果}\left|D_t(i,j)+H_t(i,j)\right|\le 2^{k_t(i,j)+1}-1\\ A_t(i,j)+G_t(i,j), & \text{如果}\left|D_t(i,j)+H_t(i,j)\right|>2^{k_t(i,j)+1}-1\end{cases} \tag{8}$$

情况 2：$R_t(i,j)>m_t(i,j)$和$\left|H_t(i,j)\right|\le 2^{k_t(i,j)-1}$和$A_t(i,j)\ge\overline{B}_t$；

$$A_t'(i,j)=A_t(i,j)+H_t(i,j) \tag{9}$$

情况 3：$R_t(i,j)>m_t(i,j)$和$\left|H_t(i,j)\right|>2^{k_t(i,j)-1}$和$A_t(i,j)<\overline{B}_t$；

$$A_t'(i,j)=\begin{cases}A_t(i,j)+H_t(i,j), & \text{如果}\left|D_t(i,j)+G_t(i,j)\right|<2^{k_t(i,j)}\\ A_t(i,j)+G_t(i,j), & \text{如果}\left|D_t(i,j)+G_t(i,j)\right|\ge 2^{k_t(i,j)}\end{cases} \tag{10}$$

情况 4：$R_t(i,j)>m_t(i,j)$和$\left|H_t(i,j)\right|>2^{k_t(i,j)-1}$和$A_t(i,j)\ge\overline{B}_t$；

$$A_t'(i,j)=\begin{cases}A_t(i,j)+G_t(i,j), & \text{如果}\left|D_t(i,j)+G_t(i,j)\right|\le 2^{k_t(i,j)+1}-1\\ A_t(i,j)+H_t(i,j), & \text{如果}\left|D_t(i,j)+G_t(i,j)\right|>2^{k_t(i,j)+1}-1\end{cases} \tag{11}$$

情况 5：$R_t(i,j)\le m_t(i,j)$和$H_t(i,j)\le 2^{k_t(i,j)-1}$和$A_t(i,j)\le\overline{B}_t$；

$$A_t'(i,j)=\begin{cases}A_t(i,j)-G_t(i,j), & \text{如果}\left|D_t(i,j)+H_t(i,j)\right|<2^{k_t(i,j)}\\ A_t(i,j)+H_t(i,j), & \text{如果}\left|D_t(i,j)+H_t(i,j)\right|\ge 2^{k_t(i,j)}\end{cases} \tag{12}$$

情况 6：$R_t(i,j)\le m_t(i,j)$和$H_t(i,j)\le 2^{k_t(i,j)-1}$和$A_t(i,j)>\overline{B}_t$；

$$A_t'(i,j)=A_t(i,j)+H_t(i,j) \tag{13}$$

情况 7：$R_t(i,j)\le m_t(i,j)$和$H_t(i,j)>2^{k_t(i,j)-1}$和$A_t(i,j)\le\overline{B}_t$；

$$A_t'(i,j)=\begin{cases}A_t(i,j)-G_t(i,j), & \text{如果}\left|D_t(i,j)-G_t(i,j)\right|\le 2^{k_t(i,j)+1}-1\\ A_t(i,j)+H_t(i,j), & \text{如果}\left|D_t(i,j)-G_t(i,j)\right|>2^{k_t(i,j)+1}-1\end{cases} \tag{14}$$

情况 8： $R_t(i,j) \le m_t(i,j)$ 和 $H_t(i,j) > 2^{k_t(i,j)-1}$ 和 $A_t(i,j) > \bar{B}_t$；

$$A'_t(i,j) = \begin{cases} A_t(i,j) + H_t(i,j), & \text{如果} \left|D_t(i,j) - G_t(i,j)\right| < 2^{k_t(i,j)} \\ A_t(i,j) - G_t(i,j), & \text{如果} \left|D_t(i,j) - G_t(i,j)\right| \ge 2^{k_t(i,j)} \end{cases} \tag{15}$$

9) 像素调整。少数情况下，通过上述的秘密信息嵌入，像素值 $A'_t(i,j)$ 超出像素值的范围[0，255]，即 $A'_t(i,j) < 0$ 或 $A'_t(i,j) > 255$。如果出现这种情况， $A'_t(i,j)$ 通过下列最优方式进行调整，寻找符合下列四个条件的像素值 $A^*_t(i,j)$ 替换 $A'_t(i,j)$：

a) $D^*_t(i,j)$ 和 $D_t(i,j)$ 属于同一个范围，其中 $\left|D_t(i,j)\right|$ 和 $\left|D^*_t(i,j)\right|$ 分别表示嵌入信息前后两像素的差，

b) $0 \le A^*_t(i,j) \le 255$；

c) $\left(A^*_t(i,j) - A_t(i,j)\right)^2$ 最小；

d) $\left(A^*_t(i,j) - \bar{B}_t\right) \bmod 2^{k_t(i,j)} = m_t(i,j)$。自此，实现了在该像素上嵌入秘密信息。

10) 重步骤(2)-(9)，直到所有的秘密信息都被嵌入到图像中，得到嵌入后的载密图像。

2.2. 秘密信息提取算法

这一小节描述提取秘密信息的详细过程。先把载密图像分成互不重叠的大小为 $n \times n$ 的像素块，每像素块的处理跟嵌入过程一样：按索引函数确定像素块中的参考像素，然后计算参考像素与其周围像素的差，由差值所处的范围值确实嵌入的秘密信息的比特位。详细的提取信息过程如下：

1) 把载密图像分成互不重叠大小为 $n \times n$ 的像素块 $A^*_t(t = 1,2,\cdots,N)$；

2) 如果 $F^*_t(i,j) = t \bmod n$，则选择 $A^*_t(i,j)$ 作为参考素 B^*_t，对参考像素值进行量化

$$\bar{B}^*_t = \left\lfloor \frac{B^*_t}{n^2} \right\rfloor \times n^2 \tag{16}$$

3) 计算每像素块中参考像素量化后的值与其周围像素值的差

$$D^*_t(i,j) = A^*_t(i,j) - \bar{B}^*_t \tag{17}$$

其中 $A^*_t(i,j)$ 表示第 t 块像素块中除参考像素外其它的像素值；

4) $\left|D^*_t(i,j)\right|$ 属于 $\left[l_{i,j}, u_{i,j}\right]$，则嵌入的秘密信息的比特数为 $k^*_t(i,j) = \left\lfloor \log_2\left(u_{i,j} - l_{i,j} + 1\right) \right\rfloor$；

5) 计算像素值差 $D^*_t(i,j)$ 与 $k^*_t(i,j)$ 所得的余数就是嵌入的秘密信息 $m^*_t(i,j)$

$$m^*_t(i,j) = D^*_t(i,j) \bmod 2^{k^*_t(i,j)} \tag{18}$$

最后，把 $m^*_t(i,j)$ 转换成 $k^*_t(i,j)$ 比特位的二进制信息流，于是，嵌入到载体中的秘密信息被提取了。

3. 举例

这一节我们以图 1 作为例子，把原始图像分成 2×2 的像素块。

3.1. 举例说明秘密信息嵌入

不妨假设 $t = 37$，则由

$$F_{37}(0,1) = F_t(0,1) = t \bmod 2^n = 37 \bmod 2^2 = 1$$

可知，选择的参考像素是 $B_{37} = 97$。接着由等式(3)量化参考像素值

A(0,0)	A(0,1)
A(1,0)	A(1,1)

(a)

60	97
145	34

(b)

58	97
147	54

(c)

Figure 1. The embedding process. (a) Index function; (b) Original image; (c) Stego-image
图 1. 嵌入过程。(a) 索引函数；(b) 原始图像；(c)载密图像

$$\bar{B}_{37}=\bar{B}_t=\left\lfloor\frac{B_t}{n^2}\right\rfloor\times n^2=\left\lfloor\frac{97}{2^2}\right\rfloor\times 2^2=96$$

根据等式(4)，算出量化后的参考像素值与其周围像素值的差：

$$D_{37}(0,0)=A_{37}(0,0)-\bar{B}_{37}=60-96=-36$$

$$D_{37}(1,0)=A_{37}(1,0)-\bar{B}_{37}=145-96=49$$

$$D_{37}(1,1)=A_{37}(1,1)-\bar{B}_{37}=34-96=-62$$

所以，我们有

$$k_{37}(0,0)=\log_2(63-32+1)=5$$

$$k_{37}(1,0)=\log_2(63-32+1)=5$$

$$k_{37}(1,1)=\log_2(63-32+1)=5$$

然后，由式(5)可得：

$$R_{37}(0,0)=F(D_{37}(0,0))=D_{37}(0,0)\bmod 2^5=-36\bmod 2^5=28$$

$$R_{37}(1,0)=F(D_{37}(1,0))=D_{37}(1,0)\bmod 2^5=49\bmod 2^5=17$$

$$R_{37}(1,1)=F(D_{37}(1,1))=D_{37}(1,1)\bmod 2^5=-62\bmod 2^5=2$$

若秘密信息二进制比特流为$(11010,10011,10110)_2$，把它们转换成十进制的数

$$m_{37}(0,0)=(11010)_2=26,\ m_{37}(1,0)=(10011)_2=19,\ m_{37}(1,1)=(10110)_2=22$$

因此，利用等式(6)，获得像素块新的差值：

$$H_{37}(0,0)=m_{37}(0,0)-R_{37}(0,0)=26-28=-2$$

$$H_{37}(1,0)=m_{37}(1,0)-R_{37}(1,0)=19-17=2$$

$$H_{37}(1,1)=m_{37}(1,1)-R_{37}(1,1)=22-2=20$$

同时由等式(7)，有

$$G_{37}(0,0)=2^{k_{37}(0,0)}-|H_{37}(0,0)|=2^5-|-2|=30$$

$$G_{37}(1,0)=2^{k_{37}(1,0)}-|H_{37}(1,0)|=2^5-2=30$$

$$G_{37}(1,1)=2^{k_{37}(1,1)}-|H_{37}(1,1)|=2^5-20=12$$

由秘密信息嵌入过程中的第八步骤，得到像素块中新的像素值：

$$A'_{37}(0,0)=A_{37}(0,0)+H_{37}(0,0)=60-2=58$$

$$A'_{37}(1,0)=A_{37}(1,0)+H_{37}(1,0)=145+2=147$$

$$A'_{37}(1,1)=A_{37}(1,1)+H_{37}(1,1)=34+20=54$$

因为 $A'_{37}(0,0),A'_{37}(1,0)$ 和 $A'_{37}(1,1)$ 都属于 $[0,255]$，因此，我们有

$$A^*_{37}(0,0)=58,A^*_{37}(1,0)=147,A^*_{37}(1,1)=54$$

自此，秘密信息已成功的嵌入到原始图像中。

3.2. 举例说明秘密信息提取

若 $t=37$，则我们由 $F^*_{37}(0,1)=F^*_t(0,1)=t \bmod 2^n=37 \bmod 2^2=1$ 选择参考像素值 $B^*_{37}=97$。首先，我们应用等式(16)量化参考像素值，得

$$\bar{B}^*_{37}=\bar{B}^*_t=\left\lfloor \frac{B^*_t}{n^2} \right\rfloor \times n^2=\left\lfloor \frac{97}{2^2} \right\rfloor \times 2^2=96$$

进一步由等式(17)获得像素差：

$$D^*_{37}(0,0)=A^*_{37}(0,0)-\bar{B}^*_{37}=58-96=-38$$

$$D^*_{37}(1,0)=A^*_{37}(1,0)-\bar{B}^*_{37}=147-96=51$$

$$D^*_{37}(1,1)=A^*_{37}(1,1)-\bar{B}^*_{37}=54-96=-42$$

因此，可得

$$k^*_{37}(0,0)=\log_2(63-32+1)=5$$

$$k^*_{37}(1,0)=\log_2(63-32+1)=5$$

$$k^*_{37}(1,1)=\log_2(63-32+1)=5$$

同时，根据等式(18)，很容易可得到

$$m^*_{37}(0,0)=D^*_{37}(0,0) \bmod 2^5=-38 \bmod 2^5=26$$

$$m^*_{37}(1,0)=D^*_{37}(1,0) \bmod 2^5=51 \bmod 2^5=19$$

$$m^*_{37}(1,1)=D^*_{37}(1,1) \bmod 2^5=-42 \bmod 2^5=22$$

最后，把这些十进制的数转换成二进制的比特流，$26=(11010)_2,19=(10011)_2,22=(10110)_2$。所以嵌入的秘密信息为 $(11010,10011,10110)_2$。由此可知，嵌入的秘密信息能完全正确的提取出来。

4. 实验结果与比较

本节将给出实验对比结果来验证改进算法的有效性。实验中用 8 幅大小为 512×512 经典灰度图像(“Lena”, “Peppers”, “Baboon”, “House”, “Boat”, “F16”, “Man”, “Elaine”)作为载体图像。图 2 给出了这 8 幅灰度图像，用伪随机数发生器生成的一串伪随机数字当作秘密信息嵌入到载体图像中。

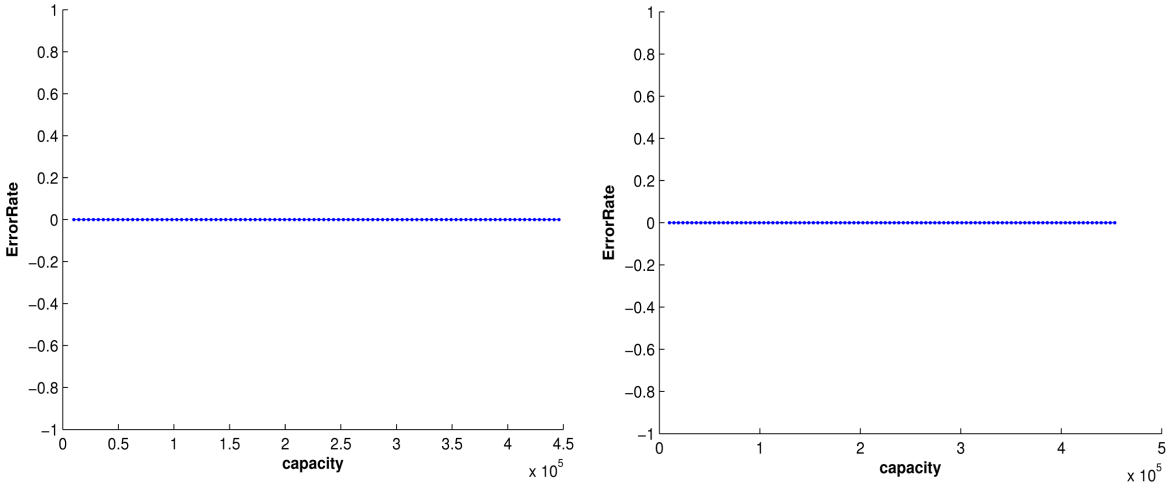
图 2 给出了实验的八幅原始图像与载密图像，其中图 2(a)，图 2(c)，图 2(e)，图 2(g)，图 2(i)，图 2(k)，图 2(m)，图 2(o)是原始图像，图 2(b)，图 2(d)，图 2(f)，图 2(h)，图 2(j)，图 2(l)，图 2(n)，图 2(p)是载密图像。实验结果表明，本文所提出的算法当嵌入秘密信息后，不会影响图像的视觉效果，人的肉眼很难分辨出哪是原始图像，哪是载密图像。

图 3 给出了本文算法嵌入秘密信息比特数与提取的错误率之间的关系，从图中可以进一步得出，本文所提出的算法能把嵌入的秘密信息完全提取出来，即提取的错误率为零。

Figure 2. The eight test images. (a) Original image; (b) Stego-image; (c) Original image; (d) Stego-image; (e) Original image; (f) Stego-image; (g) Original image; (h) Stego-image; (i) Original image; (j) Stego-image; (k) Original image; (l) Stego-image

图 2. 八幅实验图像。(a) 原始图像；(b) 载密图像；(c) 原始图像；(d) 载密图像；(e) 原始图像；(f) 载密图像；(g) 原始图像；(h) 载密图像；(i) 原始图像；(j) 载密图像；(k) 原始图像；(l) 载密图像

用峰值信噪比(Peak Signalto Noise Ratio，简称 PSNR)来评估载密图像的图像质量。本文算法的实验结果由表 1 给出，表 1 中，“容量”表示的嵌入的秘密信息的比特数，“PSNR”为峰值信噪比值，“错误”为错误提取秘密信息比特数，“嵌入(秒)，提取(秒)”分别为本文所提出的算法嵌入和提取信息所需要的时间。表 1 概述了本文所提出的算法的实验结果。实验结果也表明，本文所提出的算法能把嵌入后的秘密信息完全提取出来，而且嵌入和提取所需要的时间很少。



(a) (b)

Figure 3. The error rate and the number of bits of the experiment results. (a) Peppers; (b) Lena
图 3. 提取错误率与嵌入容量。(a) Peppers; (b) Lena

Table 1. The experiment result in our proposed method
表 1. 本文算法的实验结果

原始图像	PSNR	容量	错误	嵌入(秒)	提取(秒)
Baboon	30.16084	628,892	0	4.634324	2.298003
Lena	37.96220	454,247	0	4.186936	2.210031
Barbara	32.01326	544,368	0	4.448432	2.187897
Boat	35.70142	486,985	0	4.307051	2.176670
Elaine	39.75235	445,913	0	4.102364	2.242244
Goldhill	37.15423	488,198	0	4.152837	2.223915
House	34.23972	504,068	0	5.311871	2.992603
Sailboat	34.11055	525,454	0	4.562337	2.499687
Toys	37.62412	450,020	0	4.423158	2.255411
Zelda	41.06577	425,563	0	3.995963	2.020980

从表 2 可以看出，虽然本文的算法与其它两种算法相比较，载密图像的 PSNR 值有所下降，可嵌入容量比 Wu 和 Tsai 算法[6]和 Wang 等的算法[7]分别高出 44,168~191,086 bits，38,507~171,787 bits。较高的嵌入容量说明本文的算法能更好的区分图像的平滑区域与边缘区域，例如纹理比较丰富的 Baboon 图像，嵌入容量比 Wu 和 Tsai 算法[6]和 Wang 等的算法[7]分别高出 191,086 bits 和 171,787 bits。而且从嵌入的容量来看，不同的图像嵌入容量相差比较大，边缘越丰富的图像嵌入的容量就越多。说明了本算法充分考虑了图像不同的区域承受的像素差的变化不一样，充分考虑了图像的视觉效应。

从表 3 可以看出，Jung 和 Yoo 的算法虽然嵌入容量比本文的算法要高，可选择了六幅不同的图像，由于没有充分考虑图像的平滑区域与边缘区域对像素变化的承受力的不同，它们的嵌入容量非常的接近。同时，Jung 和 Yoo 的算法嵌入信息后，对图像造成了一定程度的失真，其 PSNR 值大部分都低于 30 dB，其中最低的 PSNR 值只有 25.96 dB。

表 4 是比较本文算法和 Lee 等的算法，Zeng 等的算法的 PSNR 值和有效嵌入量，从表 4 可看出，本文的算法不仅有效嵌入量比他们的高，同时也能很好的保证载密图像的质量，即嵌入信息后造成的图像的失真程度较低。

Table 2. Comparisons of other method and our proposed
表 2. 本文算法与其它算法的实验结果比较

原始图像	容量[1]	PSNR[1]	容量[2]	PSNR[2]	容量[3]	PSNR[3]
Lena	406,632	41.71	409,807	46.96	454,247	37.96
Baboon	437,806	38.90	457,105	43.11	628,892	30.16
Peppers	401,982	41.07	407,643	46.10	446,150	36.14
Boat	421,965	39.56	422,194	45.34	486,985	35.70
House	420,386	39.51	420,786	44.61	504,068	34.24
Man	424,585	39.09	424,723	44.98	499,200	32.44

*注：上标[1]：表示Wu和Tsai的算法[6]实验结果，上标[2]：表示Wang的算法[7]实验结果，上标[3]：表示本文的实验结果

Table 3. Comparisons of other method and our proposed
表 3. 本文算法与其它算法的实验结果比较

原始图像	容量[1]	PSNR[1]	容量[2]	PSNR[2]
Lena	614,799	31.94	454,247	37.96
Baboon	686,220	25.96	628,892	30.16
Peppers	611,394	30.42	446,150	36.14
Boat	4,633,078	28.99	486,985	35.70
House	631,341	28.71	504,068	34.24
Man	637,266	28.63	499,200	32.44

*注：上标[1]：表示Yang和Yoo的算法实验结果，上标[2]：表示本文的实验结果

Table 4. Comparisons of other method and our proposed
表 4. 本文算法与其它算法的实验结果比较

原始图像	嵌入率[1]	PSNR[1]	嵌入率[2]	PSNR[2]	嵌入率[3]	PSNR[3]
Lena	0.91	34.38	1.04	32.74	1.73	37.93
Baboon	0.62	30.03	0.51	30.97	2.40	30.16
Sailboat	0.86	33.12	1.04	32.96	2.00	34.11
F16	0.91	34.76	1.16	33.94	1.77	35.87
Goldhill	0.84	32.08	0.80	31.82	1.86	37.15
Barbara	0.73	31.31	0.78	31.96	2.08	32.01

*注：上标[1]：表示Lee和Chen的算法[16]实验结果，上标[2]：表示Wang的算法[17]实验结果，上标[3]：表示本文的实验结果

表 5 是本文的算法与 Hong [18]提出的几种不同的划分区域的算法相比较。从表 4 可以看出，本文算法的平均嵌入容量分别比 Hong 提出的 AE-PVD (D_{12} = 15)，PRT-PVD (T_0 = 15)，AE-PVD (D_{12} = 15，D_{23} = 31)，PRT-PVD (T_0 = 15，T_1 = 15)多出 196,955 bits，190,980 bits，181,891 bits，172,650 bits。如边缘比较丰富的 Baboon 图像，本文的算法分别比 Hong 提出的 AE-PVD(D_{12} =15)，PRT-PVD (T_0 =15)，AE-PVD (D_{12} = 15，D_{23} = 31)，PRT-PVD (T_0 = 15，T_1 = 15)多出 261,314 bits，261,684 bits，207,509 bits，209,314 bits。即使是平滑的图像 Lena，本文的算法也分别比 Hong 提出的 AE-PVD (D_{12} = 15)，PRT-PVD (T_0 = 15)，AE-PVD (D_{12} = 15，D_{23} = 31)，PRT-PVD (T_0 = 15，T_1 = 15)多出 177,045 bits，168,797 bits，173,616 bits，161,525 bits。虽然，Hong 的算法也考虑了根据像素差来决定嵌入的秘密信息的比特数，可其划分的区域

Table 5. Comparisons of other method and our proposed
表 5. 本文算法与其它算法的实验结果比较

原始图像	容量[1]	PSNR[1]	容量[2]	PSNR[2]	容量[3]	PSNR[3]	容量[4]	PSNR[4]	容量[5]	PSNR[5]
Lena	277,202	50.62	285,450	51.36	280,631	50.16	292,722	50.33	454,247	37.96
Boat	290,502	50.22	302,888	50.72	297,345	49.42	316,736	49.15	486,985	35.70
Baboon	367,578	48.48	367,208	49.08	421,383	45.52	419,578	45.88	628,892	30.16
House	301,418	49.93	309,790	50.52	313,329	48.73	327,412	48.68	504,068	34.24
Peppesrs	279,684	50.54	283,078	51.43	284,783	49.93	290,044	50.43	446,150	36.14
F16	285,586	50.38	289,404	51.26	294,885	49.34	301,308	49.74	463,353	35.87
平均	300,328		306,303		315,392		324,633		497,283	

*注：上标[1]：表示AE-PVD (D_{12} = 15)算法实验结果，上标[2]：表示PRT-PVD (T_0 = 15)实验结果，上标[3]：表示AE-PVD (D_{12} = 15；D_{23} = 31)实验结果，上标[4]：表示PRT-PVD (T_0 = 15；T_1 = 15)实验结果，上标[5]：表示本文的实验结果

相对而言比较粗糙，不能完全真实的反应图像的区域特征。虽然本算法的 PSNR 值比 Hong 提出的几种算法低，可也能使图像的 PSNR 保持在 30 dB 上。

PSNR 值是最常用来检测嵌入信息后的载体图像质量，通过 PSNR 值的大小，我们可很清楚的知道嵌入信息后的图像是否会被人的肉眼观测出来。然而，PSNR 值不能指出嵌入信息前后两幅图像之间差别有多大。因此，应用由 Wang 和 Bovik [19]提出的通用质量因子 Q 值来检测图像在嵌入信息前后的相似度有多高，并指出 Q 值的范围是从−1 到 1。Q 值越接近于 1，说明嵌入信息前后的两幅图像越相似。由表 6 所给出的数据，可知通过本文算法进行信息隐藏，所得的 Q 值非常的接近 1，也就是说嵌入信息前后原始图像与载密图像非常的相似。

所有的实验结果都表明，本文所提出的信息隐藏算法不仅有着很高的嵌入容量，而且也能保持很小的图像失真。由于嵌入信息后，图像失真很小，因此，即使非法者拥有了载密图像，他们也区分不了载密图像与原始图像，不知道载密图像中已经隐藏了秘密信息，也就是说本文所提出的算法能很好的隐藏信息已经嵌入到图像中去这样的事实。

5. 分析和讨论

5.1. RS 分析

Fridrich 等人[20]提出的 RS 分析算法来分析本文所提出的信息隐藏算法的安全性。RS 分析本文所提出的信息隐藏算法的结果在图 4 中给出，其中是以图 2 中的一幅图做实验。图 4 中的 x-轴表示嵌入秘密信息容量的百分比，y-轴表示掩膜为 $m=[0,1,1,0]$ 和 $-m=[0,-1,-1,0]$，正则组与奇异组的百分比。从图 4(a)和图 4(c)我们可以看出 RS 分析能检测出图像经过了 LSB 信息隐藏，这是因为随着 LSB 嵌入率到达 100%时，R_m 和 S_m 的差距会随着密写率的上升而下降，而 R_{-m} 和 S_{-m} 与原始值之间就会偏离得更远。但图 4(b)和图 4(d)载体图像看似不包含任何嵌入秘密信息的嵌入，这是因为在图 4(b)和图 4(d)中 R_m 和 R_{-m} 始终近似相等，而 S_m 和 S_{-m} 值亦如此，也就是说 $R_m \cong R_{-m}$ 和 $S_m \cong S_{-m}$。因此，可以肯定的说，本文所提出的信息隐藏算法抗 RS 分析。

5.2. PVD 直方图分析

Zhang 和 Wang [21]指出文献[22]所提出的一种基于相邻像素对差异的 PVD 信息隐藏算法导致了像素灰度差值直方图的异常，存在安全漏洞。通过基于 PVD 的直方图的分析，可以检测和分析出图像中是否隐藏了秘密信息。

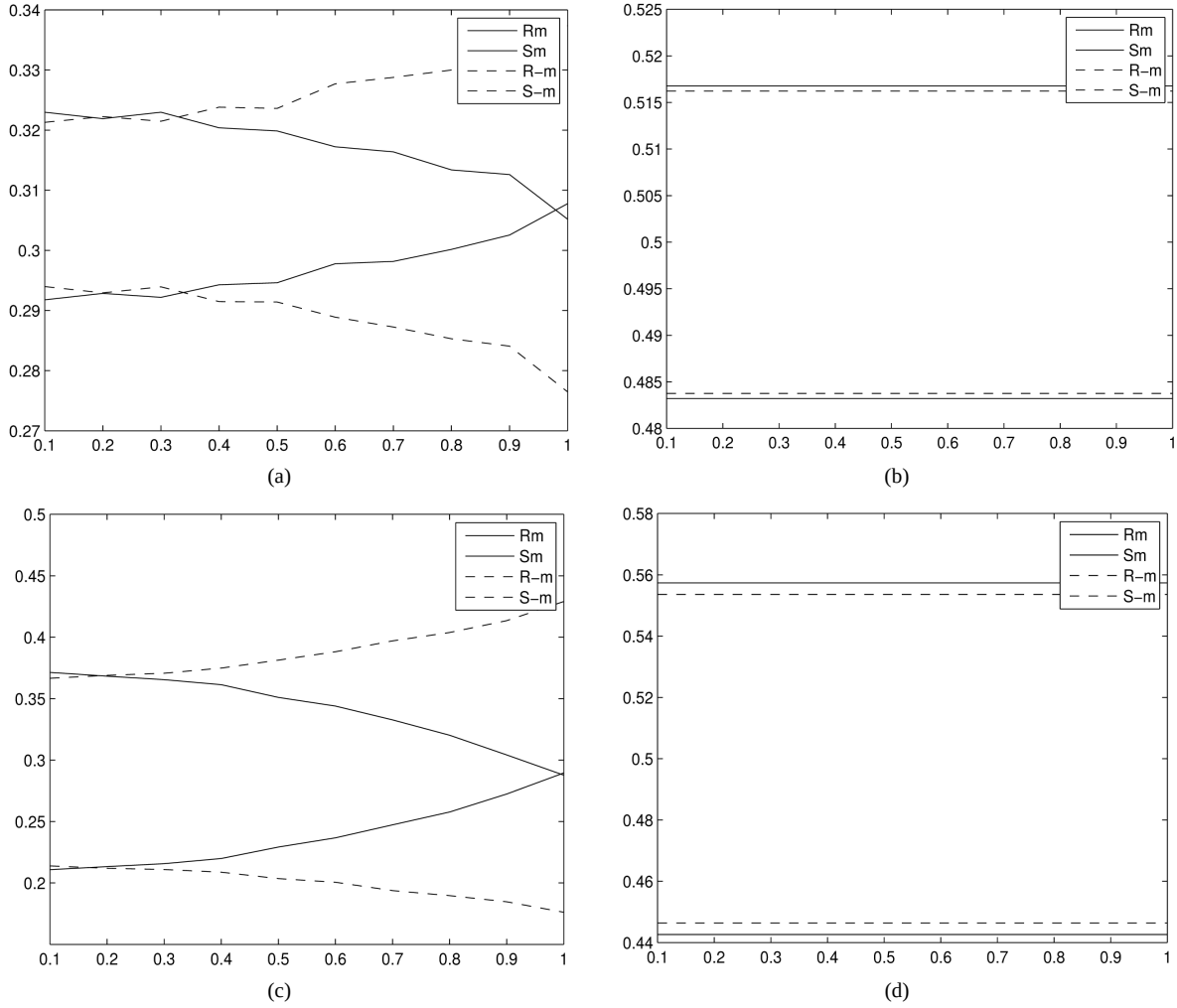
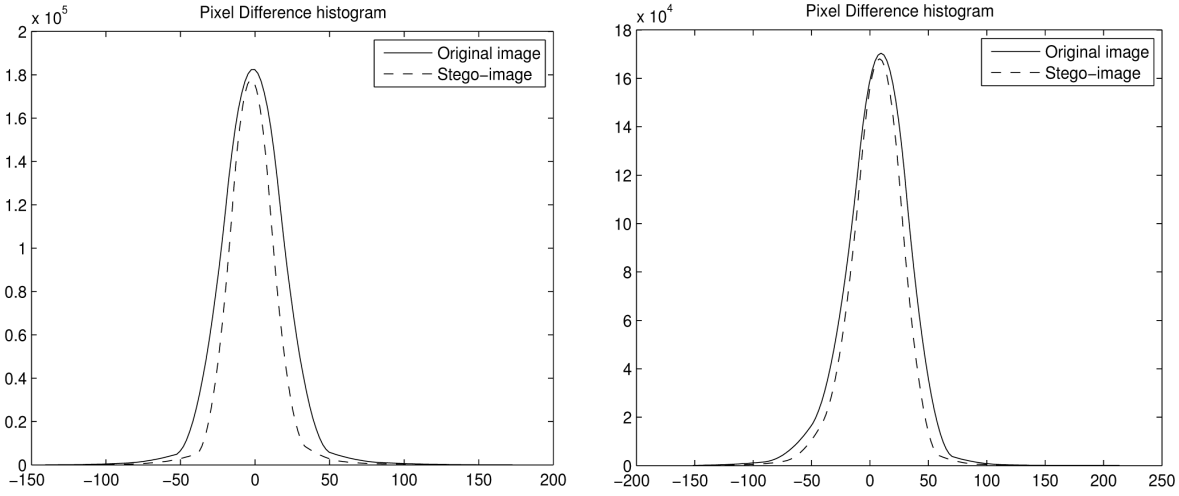


Figure 4. RS-diagrams yielded by the dual statistics method by Fridrich et al. for cover images produced by conventional LSB-embedding technique and our method. (a) LSB (Baboon); (b) our proposed (Baboon); (c) LSB (Peppers); (d) our proposed (Peppers)
图 4. 应用 RS 分析 LSB 嵌入信息算法与本文所提出的嵌入信息算法。(a) LSB (Baboon); (b) 本文算法(Baboon); (c) LSB (Peppers); (d) 本文算法(Peppers)

Table 6. The experiment result in our proposed method
表 6. 本文算法的实验结果

图像	Baboon	Lena	Barbara	Boat	Bridge	Couple	Elaine
Q	0.9827	0.9978	0.9932	0.9968	0.9930	0.9926	0.9984Z
图像	Goldhill	House	Man	Pepper	Sailboat	Toys	elda
Q	0.9974	0.9947	0.9939	0.9975	0.9971	0.9966	0.9985

图 5 给出了 PVD 直方图分析本文所提出的信息隐藏算法在嵌入容量最大时的原始图像与载密图像的直方图，图 5(a)和图 5(b)给出的是灰度图像“Baboon”和“Peppers”像素差的直方图。从图 5 可以看出本文所提出的基于像素差和索引函数的信息隐藏算法抗 PVD 直方图分析，也就是说可以完好隐藏嵌入的秘密信息。



(a) (b)

Figure 5. PVD histogram differences between covers and theirs corresponding stegos. (a) F16; (b) House
图 5. 由原始图像和载密图像得到的灰度差值直方图。(a) F16; (b) House

6. 本文小结

本文提出了一种基于像素差和索引函数的信息隐藏算法，首先把原始图像分成互不重叠的像素块，然后根据索引函数确定每个像素块中的参考像素。由参考像素跟它周围的其它像素的差来决定嵌入的信息的比特数。本文所提出的索引函数改进了 Jung 等人提出的索引函数，Jung 等人提出的索引函数，要么就是所有的像素块中做为参考像素的位置是一样的，要么就得有一定的存储空间来保存索引值，而本文所提出的索引函数具有更好的随机性，即使非法截者获得了载密图像，如果他们没有索引函数也不能正确的提出秘密信息，因此索引函数起到了一种密钥的作用。实验表明，本文提出的算法有较高的嵌入容量和保持好的图像质量。

基金项目

国家自然科学基金面上项目(11571124，61572028)，2016 年广东省应用型科技研发专项资金项目重大专项(2016B020244003)，佛山市科技创新项目－高校和医院科研基础平台建设项目(2014AG100162)，华南师范大学青年教师科研培育基金项目(15KJ06)。

参考文献 (References)

[1] Cox, I.J., Miller, M.L., Bloom, J.A., Fridrich, J. and Kalker, T. (2008) Digital Watermarking and Steganography. 2nd Edition. Morgan Kaufmann, San Francisco.

[2] Hsu, C.S. and Tu, S.F. (2010) Probability-Based Tampering Detection Scheme for Digital Images. *Optics Communications*, **283**, 1737-1743. https://doi.org/10.1016/j.optcom.2009.12.073

[3] Hong, W., Chen, T.S. and Shiu, C.W. (2009) Reversible Data Hiding for High Quality Images Using Modification of Prediction Errors. *Journal of systems and Software*, **82**, 1833-1842. https://doi.org/10.1016/j.jss.2009.05.051

[4] Provos, N. and Honyman, P. (2003) Hide and Seek: An Introduction To Steganography. *IEEE Security and Privacy Magazine*, **1**, 32-44. https://doi.org/10.1109/MSECP.2003.1203220

[5] Xu, H., Wang, J. and Kim, H.J. (2010) Near-Optimal Solution to Pair-Wise LSB Matching via An Immune Programming Strategy. *Information Sciences*, **180**, 1201-1217. https://doi.org/10.1016/j.ins.2009.12.027

[6] Wu, D.C. and Tsai, W.H. (2003) A Steganographic Method for Images by Pixel-Value Differencing. *Pattern Recognition Letters*, **24**, 1613-1626. https://doi.org/10.1016/S0167-8655(02)00402-6

[7] Wang, C.M., Wu, N.I., Tsai, C.S. and Hwang, M.S. (2008) A High Quality Steganographic Method with Pixel-Value

Differencing and Modulus Function. *Journal of Systems and Software*, **81**, 150-158. https://doi.org/10.1016/j.jss.2007.01.049

[8] Joo, J.C., Lee, H.Y. and Lee, H.K. (2010) Improved Steganographic Method Preserving Pixel-Value Differencing Histogram With Modulus Function. *EURASIP Journal on advances in signal processing*, **1**, 1-13.

[9] Luo, W., Huang, F. and Huang, J. (2010) Edge Adaptive Image Steganography Based on LSB Matching Revisited. *IEEE Transactions on Information Forensics and Security*, **5**, 201-214. https://doi.org/10.1109/TIFS.2010.2041812

[10] Yang, C.H., Weng, C.Y., Wang, S.J. and Sun, H.M. (2008) Adaptive Data Hiding in Edge Areas of Images with Spatial LSB Domain Systems. *IEEE Transactions on Information Forensics and Security*, **3**, 488-497. https://doi.org/10.1109/TIFS.2008.926097

[11] Luo, D.C., Wu, N.I., Wang, C.M., Lin, Z.H. and Tsai, C.S. (2010) A Novel Adaptive Steganography Based on Local Complexity and Human Vision Sensitivity. *Journal of Systems and Software*, **83**, 1236-1248. https://doi.org/10.1016/j.jss.2010.01.050

[12] Chang, K.C., Huang, P.S., Tu, T.M. and Chang, C.P. (2007) Adaptive Image Stegannographic Scheme Based on Tri-Way Pixel-Value Differencing. *ISIC IEEE Internation Conference*, 1165-1170.

[13] Jung, K.H., Ha, K.J. and Yoo, K.Y. (2008) Image Data Hiding Method Based on Multi-Pixel Differencing and LSB Substitution Methods. *International Conference on Convergence and Hybrid Information Technoligy*, 355-358. https://doi.org/10.1109/ICHIT.2008.279

[14] Balasubramanian, C., Selvakumar, S. and Geetha, S. (2014) High Payload Image Steganography with Reduced Distortion Using Octonary Pixel Pairing Scheme. *Multimedia Tools and Applications*, **73**, 2223-2245. https://doi.org/10.1007/s11042-013-1640-4

[15] Jung, K.H. and Yoo, K.Y. (2014) High-Capacity Index Based Data Hiding Method. Multimedia Tools and Applications.

[16] Lee, C.F., Chen, H.L. and Tso, H.K. (2010) Embedding Capacity Raising in Reversible Data Hiding Based on Prediction of Difference Expansion. *Journal of Systems and Software*, **83**, 1864-1872. https://doi.org/10.1016/j.jss.2010.05.078

[17] Zeng, X.T., Li, Z. and Ping, L.D. (2012) Reversible Data Hiding Scheme Using Reference Pixel and Multilayer Embedding. *AEU International Journal of Electronics Communication*, **66**, 532-539. https://doi.org/10.1016/j.aeue.2011.11.004

[18] Hong, W. (2013) Adaptive Image Data Hiding in Edges Using Patched Reference Table and Pair-Wise Embedding Technique. *Information Sciences*, **221**, 473-489. https://doi.org/10.1016/j.ins.2012.09.013

[19] Wang, Z. and Bovik, A.C. (2002) A Universal Image Quality Index. *IEEE Signal Processing Letters*, **9**, 81-84. https://doi.org/10.1109/97.995823

[20] Fridrich, J., Goljan, M. and Du, R. (2001) Reliable Detection of LSB Steganography in Grayscale and Color Images. *Proceedings of ACM Workshop on Multimedia and Security*, 27-30. https://doi.org/10.1145/1232454.1232466

[21] Zhang, X.P. and Wang, S.Z. (2004) Vulnerability of Pixel-Value Differencing Steganography to Histogram Analyais and Modification for Enhanced Security. *Pattern Recognition Letters*, 331-339. https://doi.org/10.1016/j.patrec.2003.10.014

[22] Yang, C.H., Wang, C.Y. and Sun, H.M. (2008) Information Hiding Technique Based on Blocked PVD. *Journal of Inforamtion Management*, **15**, 29-48.