

基于块邻域的图像双脆弱水印算法

王艺龙, 李震宇, 巩道福, 刘粉林

(信息工程大学, 河南 郑州 450001)

摘要: 针对自嵌入水印技术固定偏移映射带来的安全隐患和单水印机制可恢复率受限的问题, 提出了一种基于块邻域的图像双脆弱水印算法。将图像划分为不重叠的 4×4 图像块, 并运用块截断编码生成基于块内容的恢复水印; 对任一图像块, 设置适当大小的块邻域, 并在其邻域外和邻域内随机选择两个图像块作为映射块; 同时将恢复水印分别嵌入对应的两个映射块中。算法的图像块邻域设置机制可确保图像块与其对应映射块保持一定的距离; 图像块邻域内外映射块选取机制使两个映射块之间保持一定的距离, 保证了水印的安全性。理论分析了在连续区域篡改情形下, 篡改区域与图像块邻域之间的关系, 分析表明存在合适的图像块邻域, 使被篡改区域的平均恢复率优于随机映射嵌入算法。利用数值计算给出了篡改区域平均恢复率和篡改区域各图像块恢复率的分布趋势, 趋势表明: 当篡改参数固定时, 图像块邻域与篡改区域相交面积越大, 其篡改区域的平均恢复率越高; 而当篡改参数与邻域参数满足: $t \leq (r-1)/2$ 时, 篡改区域图像块可 100% 恢复; $(r-1)/2 < t \leq r$ 时, 篡改区域中心的部分图像块可 100% 恢复; $t > r$ 时, 篡改区域图像块恢复率有所降低。实验结果验证了理论分析的合理性和有效性; 与基于随机映射的双水印嵌入算法相比, 在连续区域篡改情形下, 邻域参数较大时, 篡改区域的平均恢复率高于随机映射嵌入算法。

关键词: 图像水印; 自嵌入水印; 双水印; 邻域映射

中图分类号: TP309.7

文献标志码: A

DOI: 10.11959/j.issn.2096-109x.2023036

Image double fragile watermarking algorithm based on block neighborhood

WANG Yilong, LI Zhenyu, GONG Daofu, LIU Fenlin

Information Engineering University, Zhengzhou 450001, China

Abstract: To address the security risks associated with fixed offset mapping in self-embedded watermarking technology, as well as the limited recovery rate of single watermarking methods, an image double fragile

收稿日期: 2022-03-16; 修回日期: 2022-06-20

通信作者: 李震宇, li1989zhenyu@126.com

基金项目: 国家自然科学基金 (62002387, 61872448, U1736214, 61772549)

Foundation Item: The National Natural Science Foundation of China (62002387, 61872448, U1736214, 61772549)

引用格式: 王艺龙, 李震宇, 巩道福, 等. 基于块邻域的图像双脆弱水印算法[J]. 网络与信息安全学报, 2023, 9(3): 38-48.

Citation Format: WANG Y L, LI Z Y, GONG D F, et al. Image double fragile watermarking algorithm based on block neighborhood [J]. Chinese Journal of Network and Information Security, 2023, 9(3): 38-48.

watermarking algorithm based on block neighborhood was proposed. The image was divided into non-overlapping blocks of size 4×4 and block truncation coding was used to generate the recovery watermark based on the image block. For each image block, a neighborhood of a proper size was specified and two mapping blocks were randomly selected from the regions within and outside the neighborhood. Then the recovery watermark was embedded into the two mapping blocks simultaneously. The image block neighborhood setting mechanism of the algorithm ensured that the image block maintains a certain distance from its corresponding mapping block. Additionally, the selection mechanism of mapping blocks inside and outside the image block neighborhood ensured a certain distance between the two mapping blocks, thereby guaranteeing the security of the watermark. In the case of continuous region tampering, the relationship between the tampered region and the neighborhood of the image block was analyzed theoretically. The analysis showed that a suitable image block neighborhood can improve the average recovery rate of the tampered area compared with the random mapping embedding algorithm. Besides, the distribution trend of the average recovery rate of the tampered area and the recovery rate of each image block in the tampered area were given by numerical calculation. It was found that the tampered area image block can be recovered 100%, some image blocks in the center of the tampered area can be recovered 100%, and the recovery rate of image blocks in tampered area will be reduced. The experimental results verify the rationality and effectiveness of the theoretical analysis. Moreover, when the neighborhood parameters are large, the proposed algorithm achieves a higher average recovery rate of the tampered region than the random mapping embedding algorithm in the case of continuous region tampering.

Keywords: image watermarking; self-embedding watermarking; dual watermarking; neighborhood mapping

0 引言

数字图像在众多研究领域以及人们的生活中广泛应用。随着高速通信技术和计算机技术的飞速发展,出现了一些图像编辑工具,人们可以根据自身的目的修改图像,甚至借助神经网络对图像进行换脸等操作,人几乎无法利用肉眼判别篡改图像的真伪。如果篡改图像被人恶意利用,可能对社会造成恶劣影响,因此,如何能够对数字图像进行真实性鉴别,成为信息安全领域的重要研究内容。

自嵌入水印技术是对图像进行真伪性鉴别的有效手段,自嵌入水印技术是基于图像的内容生成水印,利用特定的映射技术寻找水印嵌入的位置,并将水印嵌入其中,实现对篡改图像的恢复。其中典型的方法是,首先对图像分块,然后将图像块作为水印生成的最小单元,找到相应的映射块用于水印嵌入,寻找映射块的映射方法可分为两类。一类为类随机映射方法,如文献[1]通过密钥产生随机序列,根据随机序列找到映射块;文献[2-4]根据混沌映射选取映射块;文献[5]依据Arnold置乱选取映射块;文献[6]将图像的恢复水

印进行随机置乱后嵌入图像块中,具有较好的恢复效果。类随机映射方法可确保攻击者在没有映射选取密钥的情况下,难以确立图像块之间的对应关系,基于这类映射的水印算法的安全性较高,但每个图像块遭受篡改的概率是相同的,因而被篡改的图像块可恢复的概率是相同的,整个图像的恢复率是受限的。另一类为固定偏移映射方法,能够改善随机映射方法的恢复率受限问题,如文献[7-11]依据图像块序号与一个固定数值的乘积、模上图像块总块数得到其映射块,而这样的方法具有一定的规律性,故存在一定的安全隐患;文献[12]针对上述两类块方法存在的问题,提出了一种基于去邻域映射的图像自嵌入脆弱水印算法,该算法在图像块邻域外寻找其映射块,使图像块与其映射块之间保持相应的距离,有效改善了上述两类方法的问题,提升了整个图像的恢复率。

上述方法只在映射块中嵌入了一份水印,而当图像块和其映射块同时被篡改,则该图像块将无法恢复,能否通过嵌入多份水印来提升图像块的恢复率是一个有趣的问题。文献[13]提出了一种双水印用于图像篡改检测和恢复,该算法将

图像分为块,并寻找其同伴块,将本块与同伴块的像素均值的高5位作为每个块的恢复水印,利用固定偏移映射方法寻找对应的两个映射块并将10 bit 恢复水印重复嵌入其中,较好地提升了整个图像恢复率。文献[14]利用 SPIHT 产生恢复水印分别嵌入在图像像素的最后一位中和 DWT 变换后的图像中。文献[15]对图像基于小波变换生成恢复水印并将其置乱,根据半色调技术对图像生成另一份恢复水印并将其置乱,将两份恢复水印嵌入块的最后两位。这类水印算法通过嵌入双份水印提高了图像的恢复效果。

借鉴上述双份水印的思想,本文提出一种基于块邻域的图像双脆弱水印算法。首先以图像块为中心设置一个块邻域,然后在该邻域以外和邻域内的部分分别随机选择一个映射块作为恢复水印嵌入位置。在邻域内与邻域外选取映射块在一定程度上保证了两个映射块之间的距离,防止两个

映射块同时被篡改以提高图像块的可恢复性,同时使映射块的选取有一定的随机性,以保证算法的安全性。本文在不同邻域和篡改区域情况下,对图像块的可恢复性进行了充分的理论和数值分析。理论分析、数值计算和实验结果表明,所提算法对连续区域篡改具有良好的适应性。

1 算法描述

本文提出了一种基于块邻域的图像双脆弱水印算法,主要包括水印生成与嵌入、篡改认证与恢复两个部分。其中,水印生成与嵌入部分包括图像分块、映射块选取和水印嵌入等。篡改检测与恢复部分包括图像分块、水印提取、水印生成、篡改认证与篡改恢复等。算法具体描述如下。

1.1 水印的生成与嵌入

设原始图像为 I , 大小为 $N \times N$, 水印的生成与嵌入过程示意如图1所示。

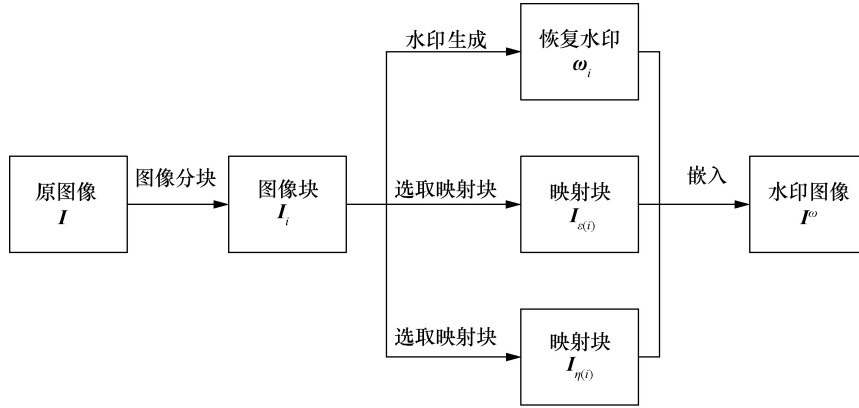


图1 水印的生成与嵌入过程
Figure 1 Schematic of generation and embedding process

(1) 图像分块

将原始图像 I 划分为 $n \times n$ 个不重叠的 4×4 的块,并按行优先顺序排列,则原图像 I 表示为

$$I = \{I_i | i = 1, 2, \dots, n^2\} \quad (1)$$

其中, $I_i = \begin{pmatrix} I_{i,1} & I_{i,2} \\ I_{i,3} & I_{i,4} \end{pmatrix}$ 为 4×4 的图像块, $I_{i,j} (i \in \{1, 2, \dots, n^2\}; j \in \{1, 2, 3, 4\})$ 为 I_i 中 2×2 的图像子块。

(2) 水印生成

1) 对图像块 I_i 的所有像素做右移两位操

作,分别计算其中 2×2 图像子块的均值,记为 $\bar{I}_{i,j}$, 则 4×4 的图像块 I_i 可表示成一个 2×2 的矩阵 $\bar{I}_i$ 。

$$\bar{I}_i = \begin{pmatrix} \bar{I}_{i,1} & \bar{I}_{i,2} \\ \bar{I}_{i,3} & \bar{I}_{i,4} \end{pmatrix} \quad (2)$$

2) 利用块截断编码方法^[9]对 $\bar{I}_i$ 进行编码。

① 计算 $\bar{I}_i$ 的均值 $\bar{\bar{I}}_i$ 和标准差 σ_i , 表示为

$$\begin{cases} \bar{\bar{I}}_i = (\bar{I}_{i,1} + \bar{I}_{i,2} + \bar{I}_{i,3} + \bar{I}_{i,4}) / 4 \\ \sigma_i = \sqrt{\frac{(\bar{I}_{i,1} - \bar{\bar{I}}_i)^2 + (\bar{I}_{i,2} - \bar{\bar{I}}_i)^2 + (\bar{I}_{i,3} - \bar{\bar{I}}_i)^2 + (\bar{I}_{i,4} - \bar{\bar{I}}_i)^2}{4}} \end{cases} \quad (3)$$

② 生成 $\bar{I}_i$ 的分布标记 $b_i = (b_1^i, b_2^i, b_3^i, b_4^i)$ 。

$$b_j^i = \begin{cases} 1, & \bar{I}_{i,j} > \bar{\bar{I}}_i \\ 0, & \bar{I}_{i,j} \leq \bar{\bar{I}}_i \end{cases}, (j=1,2,3,4) \quad (4)$$

③ 分别计算 $\bar{I}_i$ 的量化值 H_i 和 L_i ，如下式所示。

$$\begin{cases} L_i = \bar{\bar{I}}_i - \sigma_i \sqrt{\frac{q_i}{4-q_i}} \\ H_i = \bar{\bar{I}}_i + \sigma_i \sqrt{\frac{4-q_i}{q_i}} \end{cases} \quad (5)$$

其中， q_i 表示 $\bar{I}_i$ 中 $\bar{I}_{i,j}(j=1,2,3,4)$ 大于 $\bar{\bar{I}}_i$ 的个数。

3) 用 $\bar{I}_i$ 的分布标记 b_i 、量化值 H_i 和 L_i 串联产生图像块 I_i 的 16 bit 的水印 w_i 。

$$w_i = F(b_i, H_i, L_i) \quad (6)$$

其中， $F(*)$ 为串联函数。

(3) 映射块选取

对于图像块 I_i ，以其为中心，设置一个大小为 $r \times r$ 的邻域 R_i^r (r 为奇数) (如图 2 所示)。

其中，阴影部分表示 R_i^r 。

1) 生成图像块集合 S_1 ， $S_1 = I - R_i^r$ ，并依据密钥 K_1 置乱，生成一个置乱集合 $\tilde{S}_1$ 。

2) 从集合 $\tilde{S}_1$ 中随机选取一个图像块 $I_{\varepsilon(i)}$ 作为 I_i 的映射块，且 $I_{\varepsilon(i)}$ 满足：对于任意的 $i \neq j$ ，存在 $\varepsilon(i) \neq \varepsilon(j)$ 。

3) 生成图像块集合 S_2 ， $S_2 = R_i^r - I_i$ ，并依据密钥 K_2 ，生成一个置乱集合 $\tilde{S}_2$ 。从 $\tilde{S}_2$ 中随机选取另一个图像块 $I_{\eta(i)}$ 作为 I_i 的映射块，且 $I_{\eta(i)}$ 满足：对于任意的 $i \neq j$ ，存在 $\eta(i) \neq \eta(j)$ 。

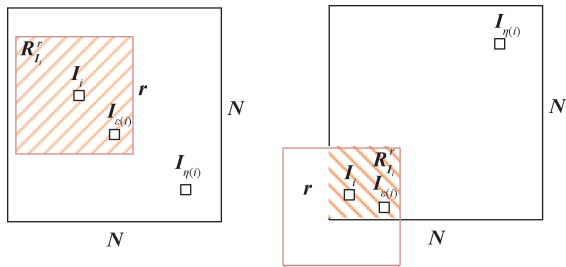


图2 邻域示意

Figure 2 Schematic of the neighborhood

(4) 水印嵌入

将图像块 I_i 的水印 w_i 嵌入在其映射块 $I_{\varepsilon(i)}$ 的最低一位中，记 $I_{\varepsilon(i)}^{1\text{lsb}} = w_i$ ；并将 w_i 嵌入在映射块 $I_{\eta(i)}$ 的次低位中，记 $I_{\eta(i)}^{2\text{lsb}} = w_i$ ，最终生成含水印图像 I^w 。

1.2 篡改认证与恢复

设待检测含水印图像为 I' ，篡改认证与恢复过程如图 3 所示。

(1) 图像分块

将 I' 无重叠地分为大小为 4×4 的图像块，表示为

$$I' = \{I'_i \mid i=1,2,\dots,n^2\} \quad (7)$$

(2) 水印提取

对任一图像块 I'_i ，利用 1.1 节中的映射方法，找到其映射块 $I'_{\varepsilon(i)}$ 和 $I'_{\eta(i)}$ ，并从映射块 $I'_{\varepsilon(i)}$ 像素的最低位中提取水印 $I'^{1\text{lsb}}_{\varepsilon(i)}$ ，从映射块 $I'_{\eta(i)}$ 像素的次低位提取水印 $I'^{2\text{lsb}}_{\eta(i)}$ ，并提取嵌入在图像块 I'_i 最低位的水印 $I'^{1\text{lsb}}_i$ 和次低位的水印 $I'^{2\text{lsb}}_i$ 。

(3) 水印生成

对图像块 I'_i ，利用 1.1 节中的水印生成方法，生成其恢复信息 w'_i ；找到图像块 I'_i 前映射块 $I'^{-1}_{\varepsilon(i)}$ 和 $I'^{-1}_{\eta(i)}$ ，并生成其恢复信息 $w'^{-1}_{\varepsilon(i)}$ 和 $w'^{-1}_{\eta(i)}$ 。

(4) 篡改认证

1) 当 $w'_i = I'^{1\text{lsb}}_{\varepsilon(i)}$ 且 $w'_i = I'^{2\text{lsb}}_{\eta(i)}$ ，则认为图像块 I'_i 未被篡改，通过认证。

2) 当 $w'_i \neq I'^{1\text{lsb}}_{\varepsilon(i)}$ 且 $w'_i = I'^{2\text{lsb}}_{\eta(i)}$ 时，则认为图像块 I'_i 未被篡改，通过认证。

3) 当 $w'_i \neq I'^{1\text{lsb}}_{\varepsilon(i)}$ 且 $w'_i = I'^{1\text{lsb}}_{\eta(i)}$ 时，则认为图像块 I'_i 未被篡改，通过认证。

4) 当 $w'_i \neq I'^{1\text{lsb}}_{\varepsilon(i)}$ 且 $w'_i \neq I'^{2\text{lsb}}_{\eta(i)}$ 时：

若 $w'^{-1}_{\varepsilon(i)} \neq I'^{1\text{lsb}}_i$ 且 $w'^{-1}_{\eta(i)} \neq I'^{2\text{lsb}}_i$ ，则认为图像块 I'_i 被篡改，不通过认证；否则，认为图像块 I'_i 未被篡改，通过认证。

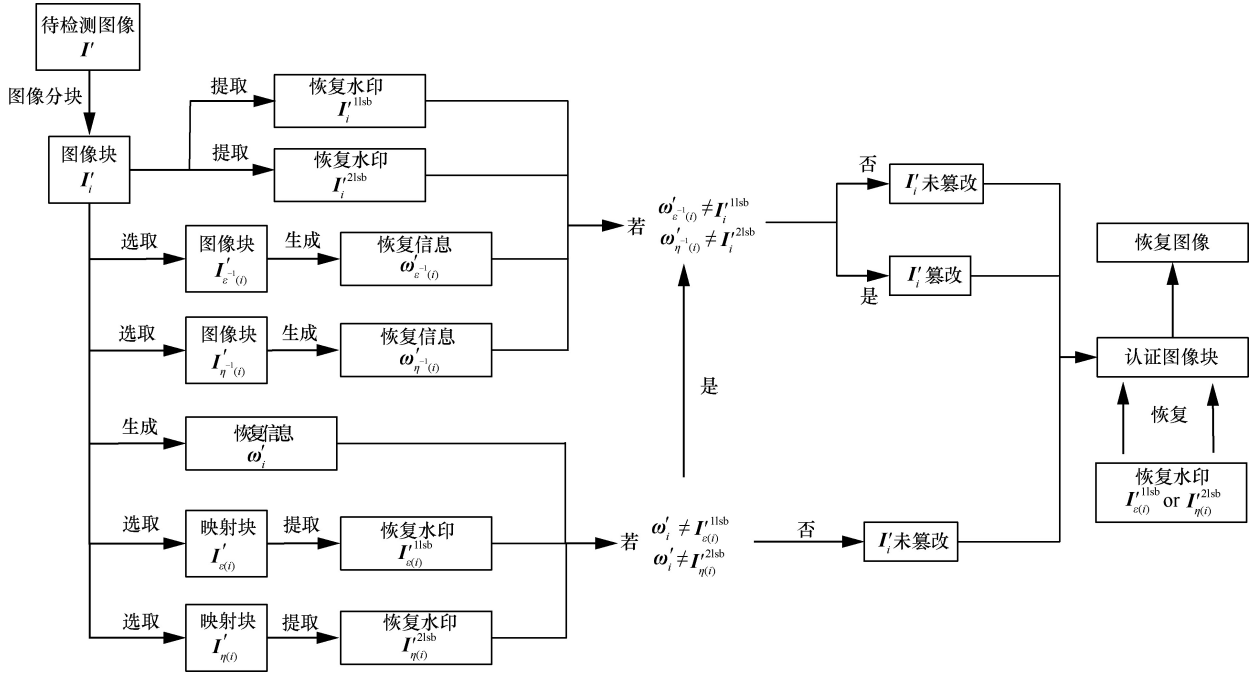


图3 篡改认证与恢复过程示意
Figure 3 Schematic of authentication and recovery

5) 认证所有图像块后,对任一未通过认证的图像块 I'_i ,判断其相邻图像块是否存在未通过认证的块,若存在,则认为图像块 I'_i 未通过认证;若不存在,则认为图像块 I'_i 通过认证。

(5) 篡改恢复

对于被篡改图像块 I'_i ,若其映射图像块 $I'_{\varepsilon(i)}$ 未被篡改,则利用 $(b_i^{1lsb}, H_i^{1lsb}, L_i^{1lsb}) = F^{-1}(I_{\varepsilon(i)}^{1lsb})$ 对图像块 I'_i 中的 2×2 子块 $I'_{i,j}$ 进行恢复。

$$I'_{i,j} = \begin{cases} H_i^{1lsb}, b_j^{1lsb,i} = 1 \\ L_i^{1lsb}, b_j^{1lsb,i} = 0 \end{cases} \quad (8)$$

若其映射图像块 $I'_{\varepsilon(i)}$ 被篡改,而映射图像块 $I'_{\eta(i)}$ 未被篡改,则利用 $(b_i^{2lsb}, H_i^{2lsb}, L_i^{2lsb}) = F^{-1}(I_{\eta(i)}^{2lsb})$ 对图像块 I'_i 中的 2×2 子块 $I'_{i,j}$ 进行恢复。

$$I'_{i,j} = \begin{cases} H_i^{2lsb}, b_j^{2lsb,i} = 1 \\ L_i^{2lsb}, b_j^{2lsb,i} = 0 \end{cases} \quad (9)$$

2 篡改恢复率的理论分析

为考察本文算法的性能,本节从理论和数值

计算的角度,给出在连续区域篡改情形下,被篡改图像块恢复率和篡改区域平均恢复率。为方便描述,引入下列符号。

$T = \{B_{i,j} | i, j = 1, 2, \dots, t^2\}$: 表示待检测图像的篡改区域,其中 $B_{i,j}$ 是 4×4 的图像块, $|\cdot|$ 表示集合势运算。

$G^r(\cdot)$: 表示被篡改图像块 $\cdot$ 的恢复率。

$\bar{G}^r$: 表示被篡改区域 T 的平均恢复率。

$\hat{G}$: 表示篡改图像的恢复率,即篡改区域中恢复出来的图像块数量与篡改图像块总数量之比。

依据第1节的算法原理可知:当图像块 $B_{i,j}$ 的两个映射块中有一个未被篡改时,就可通过提取恢复水印进行图像块的恢复。而图像块 $B_{i,j}$ 的映射块分别是除其邻域 $R_{B_{i,j}}^r$ 外的区域进行选择和邻域 $R_{B_{i,j}}^r$ 内的区域进行选择。因此,在篡改区域 L 下,只有两个映射块恰好都在篡改区域 L 中,图像块 $B_{i,j}$ 无法被恢复。故图像块 $B_{i,j}$ 的恢复率为

$$G^r(B_{i,j}) = 1 - \frac{|T \cap R_{B_{i,j}}^r|}{|R_{B_{i,j}}^r|} \cdot \frac{|T| - |T \cap R_{B_{i,j}}^r|}{|L| - |R_{B_{i,j}}^r|} \quad (10)$$

其中, $|T \cap R_{B_{i,j}}^r|$ 表示邻域 $R_{B_{i,j}}^r$ 内被篡改图像块数, $|T| - |T \cap R_{B_{i,j}}^r|$ 表示除邻域 $R_{B_{i,j}}^r$ 外被篡改图像块数, $|I| - |R_{B_{i,j}}^r|$ 表示图像块 $R_{B_{i,j}}^r$ 可能的映射区域大小。

2.1 与基于随机映射水印算法的恢复率对比

基于随机映射的水印算法, 其被篡改图像块 $B_{i,j}$ 的恢复率为

$$G^r(B_{i,j}) = 1 - \frac{|T|^2}{|I|^2} \quad (11)$$

要使本文算法对于篡改图像块的恢复率大于基于随机映射的水印算法, 即式(12)成立。

$$\frac{|T|^2}{|I|^2} - \frac{|T \cap R_{B_{i,j}}^r|}{|R_{B_{i,j}}^r|} \cdot \frac{|T| - |T \cap R_{B_{i,j}}^r|}{|I| - |R_{B_{i,j}}^r|} \geq 0 \quad (12)$$

对于式(12)有

$$\begin{aligned} & |T|^2 |R_{B_{i,j}}^r| (|I| - |R_{B_{i,j}}^r|) - \\ & |I|^2 |T \cap R_{B_{i,j}}^r| (|T| - |T \cap R_{B_{i,j}}^r|) \geq 0 \end{aligned} \quad (13)$$

由式(13)可知, 下式成立。

$$|T|^2 \left(|R_{B_{i,j}}^r| - \frac{|I|}{2} \right) \leq |I|^2 \left(|T \cap R_{B_{i,j}}^r| - \frac{|T|}{2} \right) \quad (14)$$

进而有

$$\begin{aligned} & \frac{|I|}{2} - \frac{|I|}{|T|} \left(|T \cap R_{B_{i,j}}^r| - \frac{|T|}{2} \right) \leq \\ & |R_{B_{i,j}}^r| \leq \frac{|I|}{2} + \frac{|I|}{|T|} \left(|T \cap R_{B_{i,j}}^r| - \frac{|T|}{2} \right) \end{aligned} \quad (15)$$

由式(15)可知, 存在合适的邻域, 使得本文算法的恢复性能优于随机映射的嵌入。

2.2 篡改区域位置对恢复率的影响

在篡改区域的大小一定时, 篡改区域中图像块的邻域大小与篡改区域在图像中的位置有一定关系, 如图4所示。

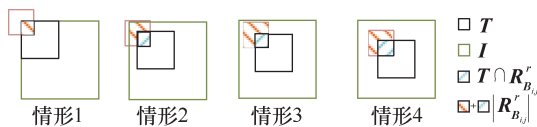


图4 T 在图像中4个不同位置的示意

Figure 4 Schematic diagram of four different positions of T in the image

由于篡改位置变化时, 难以从总体上分析图像

块的恢复率, 下面从数值计算的角度分析篡改区域为情形1~情形4时的恢复率。

情形1: $B_{1,1} = I_{1,1}$;

情形2: $B_{1,1} = I_{i,j}$, 存在 $\|R_{B_{i,j}}^r\| < r^2$;

情形3: $B_{1,1} = I_{i,j}$, 对任意的 $B_{i,j}$ 有 $|R_{B_{i,j}}^r| = r^2$;

情形4: 篡改区域中心与图像中心重合。篡改位置变化时, 对篡改区域平均恢复率的影响如图5所示。

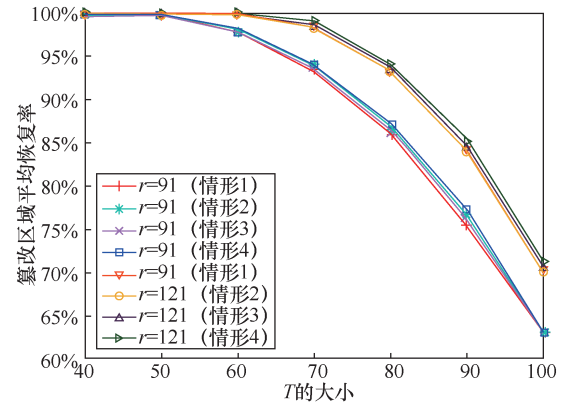


图5 篡改区域位置变化时篡改区域平均恢复率

Figure 5 Average recovery rate when the location of the tampered area is different

图5表明, 篡改区域的平均恢复率与篡改区域在图像中的位置有一定的相关性, 但在邻域参数 r 固定时, 篡改区域平均恢复率的值非常接近, 因此, 在相同篡改参数 t 时, 可忽略篡改区域位置对算法性能的影响。下面从数值计算的角度分析块邻域大小的设置与篡改区域取值不同时, 篡改区域平均恢复率的变化趋势。

2.3 块邻域尺寸对恢复率的影响

下面研究本文算法中块邻域的尺寸对于篡改区域内图像块恢复率的影响。图6给出了图像大小为 512×512 , 篡改区域 T 中 $B_{1,1} = I_{6,8}$, 在块邻域参数 $r = 11 \sim 121, 123, 125, 127$ 和篡改参数 $t = 10 \sim 100$ (步长为10) 不同时, 篡改区域平均恢复率。

图6表明, 当块邻域参数 r 固定时, 篡改区域的平均恢复率随着篡改参数 t 增大呈现下降趋势; 当篡改参数 t 固定时, 篡改区域平均恢复率随着块邻域参数 r 增大而增大。图6(a)表明, 当 r 较小时, $R_{B_{i,j}}^r$ 不满足式(15), 使得本文算法的平

均恢复率小于随机映射。当邻域参数 r 小于篡改参数 t 时, 邻域内的映射块会被篡改, 此时, 只有邻域外的映射块存在, 相当于仅有一份水印用于恢复, 所以本文算法在 r 较小时的平均恢复率会小于基于随机映射的水印算法。图 6(b)表明, 当参数 $r \geq 91$ 时, 本文算法的平均恢复率均高于随机映射的恢复率。当 r 较大时, 邻域内映射块的映射范围增大, 使得邻域内的映射块被篡改的概率降低, 并且邻域外的映射块与邻域内的映射块之间具有一定的距离, 而随机映射算法的两个映射块是随机分布在图像中的, 无法保证两个映射块之间的距离。

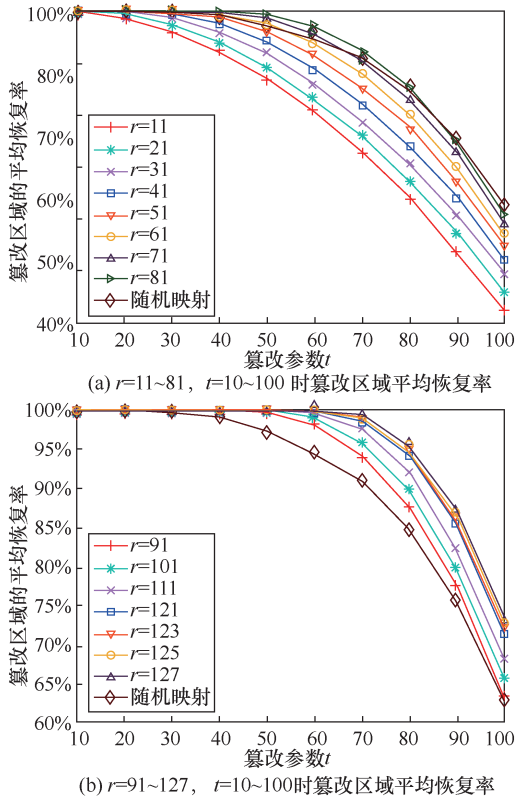


图6 在参数 r 和 t 不同取值时篡改区域平均恢复率
Figure 6 Average recovery rate of tampered area at different values of parameters r and t

图 7 给出了图像大小为 512×512 , 邻域参数 r 分别为 91 和 101, 篡改区域 T 的 $B_{1,1} = I_{6,8}$, 篡改参数 t 为 40~100 (步长为 20) 时, 图像块 $B_{i,j}$ 恢复率 $G^r(B_{i,j})$ 的数值分布。

图 7 显示, 在篡改参数固定时, 篡改区域图像块的恢复率会随着邻域增大而提升。当 $t \leq (r-1)/2$ 时, 由图 7(a)和图 7(b)可知, 篡改区

域中图像块的恢复率均为 100%; 当 $(r-1)/2 < t \leq r$ 时, 由图 7(c)~图 7(f)和图 7(h)可知, 篡改区域中心的部分图像块可 100%恢复; 当 $t > r$ 时, 由图 7(g)可知, 篡改区域图像块恢复率下降, 但仍高于随机映射嵌入算法。图 7 还表明: 图像块邻域与篡改区域交集越大, 则图像块的恢复率越高, 即越靠近篡改区域中心的图像块其恢复率越高。

综上表明, 存在合适的邻域, 使得本文算法在连续大区域篡改下, 比随机映射算法具有更好的恢复性能; 对固定大小的连续篡改, 区域平均恢复性能会随邻域的增大而提升。

2.4 块邻域 r 的选取限制

2.3 节对 r 的取值进行了充分的理论分析和实验讨论, 发现邻域参数 r 的取值越大, 图像块可恢复的概率越大。从这个角度来说, 应该取尽可能大的 r 值, 以确保篡改图像的恢复率能够达到了一个较好的水平。但 r 值越大时, 邻域外的区域可供映射块选择的区域越小, 这就造成了映射块选择策略安全性不高。同样, r 值越小时, 邻域内的区域可供映射块选择的区域越小。由 2.3 节分析可知, 在 $r \geq 91$ 时, 算法具有较好的恢复性能。综上, r 值建议选取 3/4 的图像大小。

3 实验结果与对比

本文算法选取 12 幅图像作为实验对象, 均为 512×512 的灰度图像, 如图 8 所示, 其中前 10 幅图像 (a~j) 来自 BOWS2 (break our watermarking system 2nd) 图像库, 后两幅图像 (k~l) 来自公共资源的测试图像。实验图像用例的前 10 幅图像 (a~j) 用于测试本文算法的实验结果与理论分析结果的偏离; 图像(k)用于测试本文算法的区域篡改恢复效果; 图像(l)用于本文算法与嵌入算法的恢复效果比较。

3.1 本文算法的性能

对图 8 中前 10 幅测试图像 (a~j) 分别选取 5 组邻域参数 r ($r = 91 \sim 121$ (步长为 10)), 对每幅图像采用不同的密钥生成含水印图像。对嵌入水印后的图像分别选取 5 组篡改参数 $t = 20 \sim 100$ (步长为 20) 进行篡改, 篡改区域位置设置为 $B_{1,1} = I_{6,8}$ 。将 10 幅含水印图像在相同邻域参数和相同篡改参数下的篡改区域平均可恢复率的均值作为实验值,

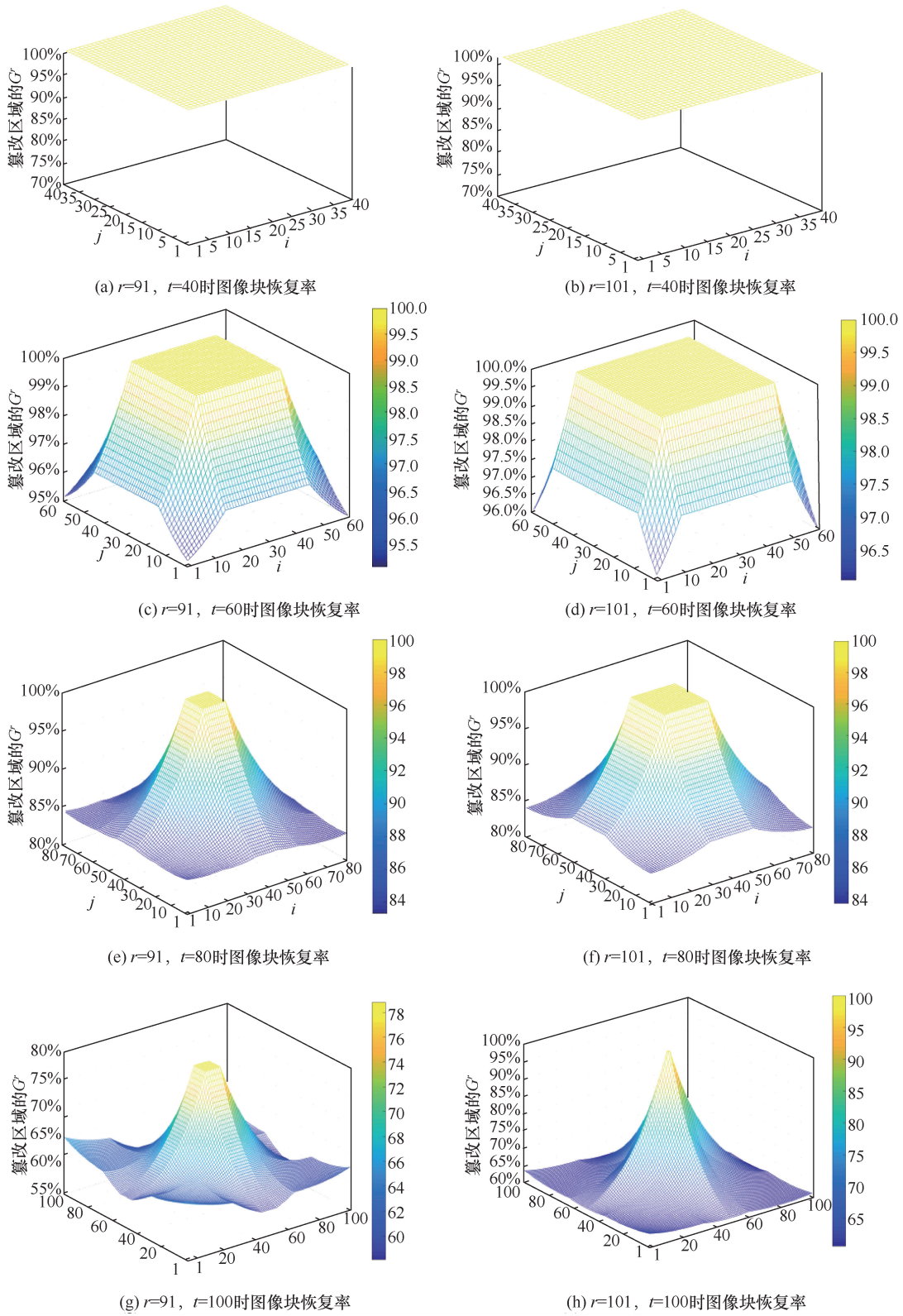
图7 篡改区域在 $B_{1,1} = I_{6,8}$ 时篡改图像块恢复率的数值分布Figure 7 Numerical distribution of the recovery rate of the tampered image block at $B_{1,1} = I_{6,8}$



图8 实验图像用例
Figure 8 Test image

表1给出了篡改区域位置相同情形下理论计算结果与实验值的对比。

从表1中可以看出在相同的 t 下,篡改区域的平均恢复率随着 r 的增大而增大;在相同 r 下,篡改区域的平均恢复率随着 t 的增大呈下降趋势;理论结果与实验结果的误差最大为0.4%。这表明实验结果与理论分析和数值计算是相符的。

对图8中图像(k),在邻域参数 $r=91$ 和 $r=101$ 运用本文算法嵌入水印,对含水印图像做如图9所示的篡改,并进行区域大小为 $t=35$ (篡改率约为6.7%)的篡改。

当邻域参数 $r=91$ 和 $r=101$ 时,本文算法的恢复效果分别如图10、图11所示。

由图10和图11可知,本文算法对连续区域篡改的恢复效果良好。当篡改区域大小相同时, r 越大,恢复效果越好。

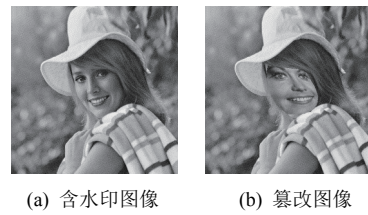


图9 对含水印图像的篡改示意
Figure 9 Tampering with watermarked image

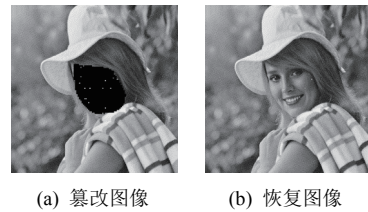


图10 $r=91$ 时图像篡改认证与恢复结果
Figure 10 Image authentication and recovery result with $r=91$

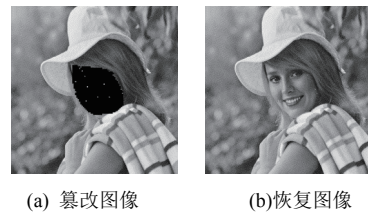


图11 $r=101$ 时图像篡改认证与恢复结果
Figure 11 Image authentication and recovery result with $r=101$

3.2 与其他算法对比实验

对图8中测试图像(l),图像大小为 512×512 ,分别比较本文算法与随机映射、文献[12]和文献[13]仅用水印的恢复效果。

随机映射算法参数设置:密钥为 $k=3$ 和 $k=5$ 。

文献[12]的参数设置:邻域参数为 $r=101$,密钥 $k=3$ 。

文献[13]的参数设置:密钥 $k=165$ 。

本文算法的邻域参数设置: $r=81$ 和 $r=101$,

表1 篡改区域的平均可恢复率
Table 1 Average recovery rate of the tampered area

篡改参数	$r=91$		$r=101$		$r=111$		$r=121$	
	实验值	理论值	实验值	理论值	实验值	理论值	实验值	理论值
$t=20$	100%	100%	100%	100%	100%	100%	100%	100%
$t=40$	100%	100%	100%	100%	100%	100%	100%	100%
$t=60$	98.00%	98.20%	99.20%	99.20%	99.70%	99.80%	100%	100%
$t=80$	87.70%	87.50%	89.40%	89.70%	91.80%	91.90%	94.10%	94.10%
$t=100$	63.30%	63.40%	65.30%	65.70%	68.10%	68.40%	71.30%	71.50%

密钥为 $k=1$ 和 $k=3$ 。

篡改参数都为 $t=20\sim 100$ （步长为 20）。比较结果如图 12 所示（ $\hat{G}$ 表示为篡改区域的仅利用水印恢复的恢复率）。

如图 12 所示，在篡改参数 $t=20\sim 100$ ，邻域参数 $r=91$ 和 $r=101$ 时，本文算法在连续篡改情形下，其篡改区域仅用水印恢复的恢复率比随机映射、文献[12]和文献[13]恢复率高。从 $r=101$ 和 $r=91$ 时的恢复图像看出，本文算法篡改区域的恢复效果随着 r 的增大而清晰，靠近篡改区域中心位置的恢复效果优于区域边缘，这进一步验证了理论分析中

越靠近篡改区域中心的图像块其恢复率越高。

4 结束语

本文提出了一种基于块邻域的图像双脆弱水印算法。恢复水印的两次嵌入为恢复图像被篡改区域提供了更多机会。基于本文提出的邻域映射算法选择两个映射块，降低了两个映射块同时被篡改的概率，同时保持了映射块选择过程的随机性，进而保持了算法的安全性。通过理论分析和实验验证发现，在单个区域连续篡改的情况下，在取适当的邻域大小时，所提算法比基于类随机



图 12 所提算法与其他映射方法的恢复对比

Figure 12 Comparison of the recovery of the proposed algorithm with other mapping methods

映射方法的算法具有更好的恢复性能。此外,篡改区域的中心部分往往比边缘部分取得更高的恢复率。然而,所提算法中仍然存在一些限制,如当邻域尺寸设置过小而篡改区域过大时,可能增加了两个映射块同时被篡改的概率,影响篡改区域的恢复效果。

参考文献:

- [1] LEE C F, SHEN J J, CHEN Z R, et al. Self-embedding authentication watermarking with effective tampered location detection and high-quality image recovery[J]. *Sensors*, 2019, 19(10): 2267.
- [2] ZHANG J P, ZHANG Q F, LV H L. A novel image tamper localization and recovery algorithm based on watermarking technology[J]. *Optik*, 2013, 124(23): 6367–6371.
- [3] DHOLE V S, PATIL N N. Self embedding fragile watermarking for image tampering detection and image recovery using self recovery blocks[C]//IEEE 2015 International Conference on Computing Communication Control and Automation. 2015:752–757.
- [4] HEMID O, HE H J. A self-recovery watermarking scheme based on block truncation coding and quantum chaos map[J]. *Multimedia Tools and Applications*, 2020, 79(3): 18695–18725.
- [5] AL-OYUM H A M, ALANSARI A W. Image watermarking for tamper detection and content recovery applications based on a preservative bi-level moment approach with shuffling[C]//ACM 5th International Conference on Computer and Technology Applications, 2019:25–29.
- [6] SINGH D, SINGH S K. Effective self-embedding watermarking scheme for image tampered detection and localization with recovery capability[J]. *Journal of Visual Communication and Image Representation*, 2016, 38(6): 775–789.
- [7] WANG C Y, ZHANG H, ZHOU X. A self-recovery fragile image watermarking with variable watermark capacity[J]. *Applied Sciences*, 2018, 8(4): 548.
- [8] SINGH D, SINGH S K. Block truncation coding based effective watermarking scheme for image authentication with recovery capability[J]. *Multimedia Tools and Applications*, 2019, 78(4): 4197–4215.
- [9] FENG B, LI X L, JIE Y M, et al. A novel semi-fragile digital watermarking scheme for scrambled image authentication and restoration[J]. *Mobile Networks and Applications*, 2020, 25(1): 82–94.
- [10] KIM C, YANG C N. Self-embedding fragile watermarking scheme to detect image tampering using ambtc and opap approaches[J]. *Applied Sciences*, 2021, 11(3): 1146.
- [11] LIU K C. Colour image watermarking for tamper proofing and pattern-based recovery[J]. *IET Image Processing*, 2012, 6(5): 445–454.
- [12] WANG Y L, LI Z Y, GONG D F, et al. Image fragile watermarking

algorithm based on deneigh-borhoodmapping[EB].

- [13] LEE T Y, LIN S D. Dual watermark for image tamper detection and recovery[J]. *Pattern recognition*, 2008, 41(11): 3497–3506.
- [14] SONG Q, ZHANG H B. Image tamper detection and recovery using dual watermark[C]//IEEE 6th International Conference on Wireless Communications Networking and Mobile Computing. 2010: 1–4.
- [15] HAGHIGHI B B, TAHERINI A H, HARATI A. TRLH: fragile and blind dual watermark-ing for image tamper detection and self-recovery based on lifting wavelet transform and halftoning technique[J]. *Journal of Visual Communication and Image Representation*, 2018, 50(2): 49–64.

[作者简介]



王艺龙(1991–),男,河南郑州人,信息工程大学博士生,主要研究方向为信息安全和信息隐藏。



李震宇(1989–),男,河南洛阳人,信息工程大学讲师,主要研究方向为多媒体安全和3D信息隐藏。



巩道福(1984–),男,山东淄博人,信息工程大学副教授,主要研究方向为信息安全、信息隐藏。



刘粉林(1964–),男,江苏溧阳人,信息工程大学教授、博士生导师,主要研究方向为数字图像取证和信息隐藏。