

文章编号:1004-4213(2011)10-1577-9

抗亮度和对比度调整的盲鲁棒量化水印算法

叶天语

(浙江工商大学 信息与电子工程学院, 杭州 310018)

摘 要:针对现有许多图像水印算法无法抵抗亮度和对比度调整,提出了一种抗亮度和对比度调整的盲鲁棒量化水印算法.对原始图像进行离散小波变换,将低频子带分成互不重叠的子块,对每个子块进行离散余弦变换,在每个子块的离散余弦变换低频系数奇偶量化嵌入水印.检测端先对攻击后的含水印图像进行抗亮度和对比度调整修正,然后通过奇偶判断盲提取出水印.实验结果表明:该算法在抵抗亮度和对比度调整上表现出较强的鲁棒性,而且在抵抗添加高斯噪音、添加椒盐噪音、剪切、中值滤波、高斯低通滤波和 JPEG 压缩也表现出较强的鲁棒性.

关键词:数字水印;量化;鲁棒性;亮度和对比度调整

中图分类号: TN911.7

文献标识码: A

doi: 10.3788/gzxb20114010.1577

0 引言

在互联网时代,数字多媒体在网络上广泛传播,从而容易产生版权纠纷.鲁棒数字水印技术被提出来用于对数字多媒体进行版权保护.

根据数字水印的内容,水印算法可以分为无意义水印算法和有意义水印算法.无意义水印算法往往用伪随机序列^[1-4]作为水印,有意义水印算法往往用二值图像^[5-7]、灰度图像^[8]等作为水印.相比于无意义水印,有意义水印具有明确的含义,更加直观.

根据检测端提取水印时是否需要借助原始载体,水印算法可以分为盲水印算法和非盲水印算法.盲水印算法^[1-3,5-8]在提取水印时不需要借助原始载体,而非盲水印算法^[4]提取水印时需要借助原始载体.因此,盲水印算法更具有通用性.采用量化嵌入规则的水印算法^[2,5,8]能够在检测端实现盲提取水印.

根据水印的嵌入位置,水印算法可以分为空域水印算法和变换域水印算法.常见的变换域有离散小波变换(Discreet Wavelet Transform, DWT)、离散余弦变换(Discreet Cosine Transform, DCT)、奇异值分解(Singular Value Decomposition, SVD)等.不同于单一的水印嵌入域,文献^[3-4,6-7]在 DWT 和 DCT 混合域嵌入水印,即在小波子带的 DCT 系数上嵌入水印.

数字图像在传输过程往往会遭遇各种攻击,其中亮度和对比度调整往往具有很强的攻击性,使得有一些图像水印算法不具有抵抗这种攻击的鲁棒

性.本文提出一种基于 DWT 和 DCT 混合域的盲鲁棒量化有意义水印算法,试图解决现有许多图像水印算法无法抵抗亮度和对比度调整攻击的问题.实验结果表明本文算法在抵抗亮度和对比度调整上表现出较强的鲁棒性.

1 算法的理论基础

1.1 离散小波变换

DWT^[3-4,6-7]被称为数学显微镜,是一种时间-尺度信号的多分辨率分析方法.图像每经历一级 DWT 后被分解为低频子带(LL)、水平方向(HL)、垂直方向(LH)和对角线方向(HH)四个频率子带,其中 LL 子带还可以继续分解. LL 子带是图像的一个低尺度逼近,HL、LH 和 HH 子带分别代表图像水平、垂直和对角方向的高频信息. LL 子带集中了图像的主要能量,HL、LH 和 HH 子带的能量则较少.图 1 为图像二级 DWT 示意图.

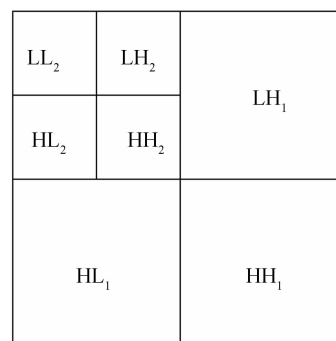


图 1 图像二级 DWT 示意图

Fig. 1 Diagram of an image after 2 level DWT

基金项目:浙江省教育厅项目(No. Y201017916)资助

第一作者:叶天语(1982-),男,讲师,博士,主要研究方向为信息隐藏与数字水印. Email: flystu008@yahoo.com.cn

收稿日期:2011-06-17;修回日期:2011-07-11

1.2 离散余弦变换

原始图像 A 的大小设为 $M \times N$, 其二维 DCT 定义为

$$C_{pq} = a_p a_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos \frac{\pi(2m+1)p}{2M} \cos \frac{\pi(2n+1)q}{2N} \quad (1)$$

式中 A_{mn} 为原始信号分量, C_{pq} 为 DCT 系数,

$$a_p = \begin{cases} \frac{1}{\sqrt{M}}, & p=0 \\ \sqrt{\frac{2}{M}}, & 1 \leq p \leq M-1 \end{cases}, \quad 0 \leq p \leq M-1,$$

$$a_q = \begin{cases} \frac{1}{\sqrt{N}}, & q=0 \\ \sqrt{\frac{2}{N}}, & 1 \leq q \leq N-1 \end{cases}, \quad 0 \leq q \leq N-1.$$

原始图像经 DCT 产生直流系数、低频交流系数、中频交流系数和低频交流系数, 它们集中的原始图像能量依次递减^[3-7].

1.3 Arnold 变换

图像置乱技术可以打乱图像像素的分布, 常用于对水印图像进行预处理. Arnold 变换又称猫脸变换, 是一种常用的图像置乱方法. 图像大小设为 $M \times M$. Arnold 变换定义为^[9]

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \left[\begin{pmatrix} 11 \\ 12 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \right] \bmod M \quad (2)$$

式中, $(x, y) \in \{0, 1, \dots, M-1\}$. Arnold 变换利用式(2)将处于 (x, y) 位置的像素点变换到 (x', y') 位置. 当完成对所有像素的变换时便得到置乱后的图像. Arnold 变换还具有周期性, 变换迭代到某一次数将重新得到原始图像.

1.4 抗亮度和对比度调整修正

文献[8]提出利用对攻击后的含水印图像进行抗亮度和对比度调整修正来抵抗亮度和对比度调整攻击. 设 p 为原始灰度图像 A 任意像素的灰度值, \hat{p} 为经亮度和对比度复合调整后的灰度值, 则存在^[8]

$$\hat{p} = a(p - p_0) + p_0 + b \quad (3)$$

式中, a 为对比度调整系数, b 为亮度调整量, p_0 一般为图像灰度阶的中间值. 当式(3)中 $a=1$ 时为只存在亮度调整的情形; 当 $b=0$ 时为只存在对比度调整的情形^[8].

令 $c = -ap_0 + p_0 + b$, 则式(3)变为^[8]

$$\hat{p} = ap + c \quad (4)$$

若 \hat{p} 超出灰度阶范围, 则通过式(5)截断为

$$\hat{p} = \begin{cases} 255, & \hat{p} > 255 \\ \hat{p}, & 0 \leq \hat{p} \leq 255 \\ 0, & \hat{p} < 0 \end{cases} \quad (5)$$

设含水印灰度图像 A' 经式(4)确定的亮度和对比度调整后的图像为 \hat{A} , 可通过式(6)的抗亮度和对比度调整修正公式确定任意像素的原灰度值 p ^[8]

$$p = \frac{\text{std}(A')}{\text{std}(\hat{A})} [\hat{p} - \text{mean}(\hat{A})] + \text{mean}(A') \quad (6)$$

2 本文的盲鲁棒量化有意义水印算法

2.1 水印嵌入算法

LL 子带是图像的一个低尺度逼近, 相比于 HL、LH 和 HH 子带, 集中了图像的主要能量, 具有更强的抗外在干扰稳定性. 图像经 DCT 变换后产生的 DC 系数集中了最多的能量, 但把水印嵌入在 DC 系数往往容易使含水印图像产生方块效应. 相比于 DCT 中、高频系数, DCT 低频系数具有更多的能量, 具有更强的抗外在干扰稳定性. 本文结合 LL 子带和 DCT 低频系数集中较多能量的特点, 将水印嵌入在 LL 子带的 DCT 低频系数上, 达到抗外在干扰稳定性的强强联合, 有利于增强算法的抗攻击鲁棒性.

原始载体图像 A 的大小为 $M \times M$, 原始水印图像 W 的大小为 $\frac{M}{2^n \cdot m} \times \frac{M}{2^n \cdot m}$. 对原始载体图像进行 DWT, 将产生的 LL 子带分割成互不重叠的子块, 然后将每比特水印嵌入在每个子块的一个 DCT 低频系数上. 水印具体嵌入过程为:

Step1: 将原始水印图像 W 进行 Arnold 置乱. 将置乱次数 k 作为密钥保存. 置乱后的水印图像记为 W' .

Step2: 将原始载体图像 A 进行 n 级 DWT, 将产生的低频子带记为 LL_n .

Step3: 将 LL_n 分割成互不重叠的大小为 $m \times m$ 的子块, 每个子块记为 $B_i, i=1, 2, \dots, \left(\frac{M}{2^n \cdot m}\right)^2$.

Step4: 对每个子块进行 DCT, 将得到的每个 DCT 子块记为 DB_i .

Step5: 通过奇偶量化规则在 DB_i 的低频系数 $DB_i(r, t)$ 嵌入置乱后的水印图像 W' , 即

$$\begin{cases} DB'_i(r, t) = (\lambda_i - 1/2) \times \delta, & \text{if } \text{mod}(W'_i + \lambda_i, 2) = 1 \\ DB'_i(r, t) = (\lambda_i + 1/2) \times \delta, & \text{if } \text{mod}(W'_i + \lambda_i, 2) = 0 \end{cases} \quad (7)$$

式中, $\lambda_i = \text{round}(DB_i(r, t)/\delta)$; $\text{round}(\cdot)$ 为舍入取整运算; $\text{mod}(\cdot)$ 为求模运算; $DB'_i(r, t)$ 为修改后的 $DB_i(r, t)$, $1 \leq r \leq m$ 和 $1 \leq t \leq m$; δ 为量化步长; W'_i 是 W' 的第 i bit.

Step6: 对嵌入水印后的每个子块进行逆离散余弦变换 (Inverse Discreet Cosine Transform, IDCT) 得到嵌入水印后的小波低频子带 LL'_n .

Step7:将 LL_n^w 与 Step2 得到的所有水平方向、垂直方向和对角线方向子带一起进行逆离散小波变换(Inverse Discrete Wavelet Transform, IDWT)产生含水印图像 A' 。

Step8:计算并保存含水印图像 A' 的平均值 $\text{mean}(A')$ 和标准差 $\text{std}(A')$, 用于在水印提取时进行抗亮度和对比度调整修正。

原始水印图像进行 Arnold 置乱的作用在于:1) 增强算法的安全性. 不知道置乱次数则无法在提取端恢复出原始水印. 2) 增强算法抵抗剪切攻击的鲁棒性. 原始水印图像的像素点分布被 Arnold 置乱打乱, 然后随机分布在含水印图像, 而对含水印图像的剪切攻击往往集中在一个区域, 这样有助于提高提取的水印的视觉效果。

2.2 水印提取算法

水印提取过程是水印嵌入过程的逆过程. 因为采用量化规则嵌入水印, 所以水印提取不需要原始载体图像, 实现盲提取. 水印提取过程为:

Step1:计算攻击后的含水印图像 \hat{A} 的平均值 $\text{mean}(\hat{A})$ 和标准差 $\text{std}(\hat{A})$ 。

Step2:利用式(6)对攻击后的含水印图像 \hat{A} 的每个像素灰度值 \hat{p} 进行抗亮度和对比度调整修正。

Step3:将修正后的攻击含水印图像 \hat{A} 进行 n 级 DWT, 将产生的低频子带记为 LL'_n 。

Step4:将 LL'_n 分割成互不重叠的大小为 $m \times m$ 的子块, 每个子块记为 $B'_i, i=1, 2, \dots, \left(\frac{M}{2^n \cdot m}\right)^2$ 。

Step5:对每个子块进行 DCT 变换, 将得到的每个 DCT 子块记为 DB''_i 。

Step6:通过奇偶判断提取置乱后的水印 W'' , 即

$$W''_i = \begin{cases} 1, & \text{mod}(\lambda'_i, 2) = 1 \\ 0, & \text{mod}(\lambda'_i, 2) = 0 \end{cases} \quad (8)$$

式中, W''_i 是 W'' 的第 i bit; $\lambda'_i = \text{floor}(DB''_i(r, t)/\delta)$; $\text{floor}(\cdot)$ 为向下取整运算。

Step7:利用密钥 k 对 W'' 进行 Arnold 逆置乱, 得到提出的水印 W''' 。

Step8:计算 W 和 W''' 之间的归一化相关度(Normalized Correlation, NC)以认证版权. NC 定义为

$$\text{NC} = \frac{\sum_{i=1}^{\left(\frac{M}{2^n \cdot m}\right)^2} (W_i \cdot W'''_i)}{\sqrt{\sum_{i=1}^{\left(\frac{M}{2^n \cdot m}\right)^2} (W_i)^2} \times \sqrt{\sum_{i=1}^{\left(\frac{M}{2^n \cdot m}\right)^2} (W'''_i)^2}} \quad (9)$$

如果 NC 大于等于阈值 ρ , 则认为版权合法. 阈

值 ρ 根据实际应用时版权认证要求的高低来确定。

3 实验结果及讨论

3.1 实验结果

根据数字水印技术理论, 对于有意义的水印而言, 评价算法的不可见性和鲁棒性的标准有主观评价标准和客观评价标准两种. 本文算法在下文将未攻击时的含水印图像列出来与原始载体图像进行对比, 从主观视觉上评价算法的不可见性; 计算未攻击时的含水印图像与原始载体图像之间的 PSNR, 作为客观标准评价算法的不可见性; 将攻击时提取的水印图像列出来与原始水印图像进行对比, 从主观视觉上评价算法的抗攻击鲁棒性; 计算攻击时提取的水印图像与原始水印图像之间的 NC, 作为客观标准评价算法的抗攻击鲁棒性。

原始水印图像为图 2 所示的大小为 32×32 的二值图像 Hand.bmp, 三幅原始载体图像分别为图 3(a)、图 4(a)、图 5(a)所示的大小为 512×512 的 256 灰度级灰度图像 Pepper.jpg、Lena.bmp 和 Baboon.bmp. 限于篇幅, 本文只列出这三幅图像的实验结果. 原始水印图像 Arnold 置乱次数为 2 次. 以 harr 小波为基对三幅原始载体图像进行 DWT,



图 2 原始 hand
Fig. 2 Original hand

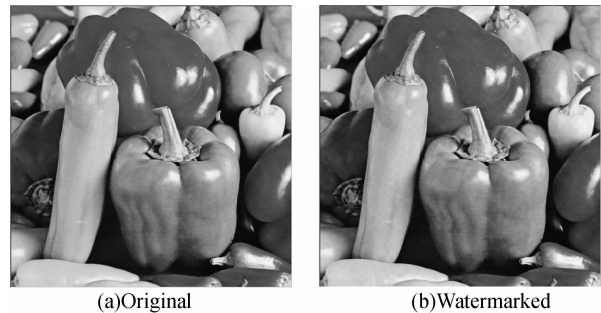


图 3 原始 Pepper 与含水印 Pepper
Fig. 3 Original Pepper and Watermarked Pepper



图 4 原始 Lena 与含水印 Lena
Fig. 4 Original Lena and Watermarked Lena

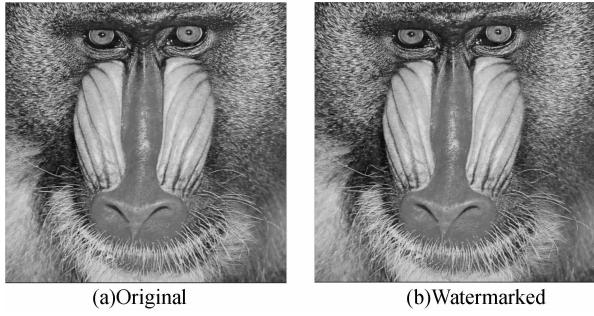


图 5 原始 Baboon 与含水印 Baboon
Fig. 5 Original Baboon and Watermarked Baboon

小波分解级数为 1 级,小波低频子带分块大小为 8×8 . 实验时将水印嵌入在每个子块处于(2,2)的 DCT 低频系数以测试算法的不可见性和抗攻击鲁棒性. 由下文实验结果可知,此时算法具有良好的不可见性和鲁棒性. 图 3(a)、图 4(a)、图 5(a)三幅图像的量化步长 δ 分别为 40、45 和 41. 得到的含水印图像分别见图 3(b)、图 4(b)、图 5(b),与相应原始载体图像之间的 PSNR 都为 43 dB,此时算法具有很好的不可见性. 此时从图 3(b)、图 4(b)、图 5(b)提取的水印图像都与图 2 完全一致,原始水印与提取的水印之间的 NC 都等于 1.

对算法进行抵抗不同参量时的亮度和对比度调整攻击的鲁棒性测试,实验参量及结果见表 1. 这里列出了原始水印图像与从攻击后的含水印图像提取的水印图像之间的 NC 和提取出的水印图像. 根据

表 1 的第二栏,对于三幅测试图像,本文算法在表 1 各种参量下的亮度和对比度调整都能够以较高的 NC 提取出水印,表现出较强的鲁棒性. 这是因为检测端在提取水印时先对攻击的含水印图像进行了抗亮度和对比度调整修正. 另外,还测试了没有进行抗亮度和对比度调整修正时的抗亮度和对比度调整攻击的鲁棒性,实验结果列在表 1 第三栏. 对比表 1 第二栏和第三栏,本文算法在存在修正时的抗亮度和对比度调整攻击的鲁棒性强于不存在修正时的鲁棒性. 本文还测试了存在修正时算法抵抗亮度和对比度很大幅度调整的鲁棒性. 实验结果表明,此时算法没有以较高的 NC 提取出水印. 这是因为:亮度和对比度很大幅度调整时,会有大量像素超出灰度阶范围,被式(5)截断,从而这些像素无法利用式(6)进行修正复原. 然而,亮度和对比度很大幅度调整会使图像的意义和使用价值发生很大的改变,从而不是一种有效的攻击手段. 另外,目前的大多数水印算法无法抵抗亮度和对比度调整攻击,如文献[5]的算法在这方面的鲁棒性比较差. 而且从下文可知,在抵抗表 1 各种参量下的亮度和对比度调整的鲁棒性上,存在修正时本文算法强于文献[5-8]的算法. 因此,可以认为本文算法抵抗亮度和对比度调整攻击的鲁棒性比较强.

表 1 不同参量时亮度和对比度调整的实验结果

Table 1 Experimental results for brightness-and-contrast adjustment with different parameters

Image	Parameter		(a,c)	(1,50)	(1,-30)	(1.3,0)	(0.7,0)	(1.2,20)	(0.93,-10)
	Algorithm								
Peppers	This paper (revision)	Extracted watermark							
		NC	0.9900	0.9727	0.9692	1.0000	0.9692	0.9982	
	This paper (no revision)	Extracted watermark							
		NC	0.9888	0.9655	0.8589	0.8785	0.8747	0.9763	
	Reference[5]	Extracted watermark							
		NC	0.9649	0.9424	0.6565	0.6264	0.6318	0.5835	
	Reference[6]	Extracted watermark							
		NC	0.9534	0.9410	0.9461	0.9618	0.9461	0.9588	
	Reference[7]	Extracted watermark							
		NC	0.9829	0.9691	0.9684	1.0000	0.9605	0.9953	
	Reference[8]	Extracted watermark							
		NC	0.9781	0.7056	0.9380	1.0000	0.9380	0.9988	

(续表)

Image	Parameter		(a,c)	(1,50)	(1,-30)	(1,3,0)	(0,7,0)	(1,2,20)	(0.93,-10)
	Algorithm								
Lena	This paper (revision)	Extracted watermark							
		NC	0.9847	1.0000	0.9710	1.0000	0.9704	1.0000	
	This paper (no revision)	Extracted watermark							
		NC	0.9853	1.0000	0.8833	0.9041	0.8994	0.9823	
	Reference[5]	Extracted watermark							
		NC	0.9721	0.0000	0.6409	0.6494	0.6264	0.6589	
	Reference[6]	Extracted watermark							
NC		0.9318	0.9405	0.9243	0.9405	0.9256	0.9405		
Reference[7]	Extracted watermark								
	NC	0.9829	1.0000	0.9607	1.0000	0.9674	1.0000		
Reference[8]	Extracted watermark								
	NC	0.9715	1.0000	0.7827	1.0000	0.7694	1.0000		
Baboon	This paper (revision)	Extracted watermark							
		NC	1.0000	0.9988	0.9988	1.0000	0.9988	1.0000	
	This paper (no revision)	Extracted watermark							
		NC	1.0000	0.9988	0.8121	0.8326	0.8871	0.9935	
	Reference[5]	Extracted watermark							
		NC	0.0452	0.9805	0.6159	0.6384	0.6346	0.6449	
	Reference[6]	Extracted watermark							
NC		0.9876	0.9900	0.9829	0.9894	0.9829	0.9894		
Reference[7]	Extracted watermark								
	NC	1.0000	0.9977	0.9929	1.0000	0.9988	1.0000		
Reference[8]	Extracted watermark								
	NC	1.0000	0.9977	0.9781	1.0000	0.9763	1.0000		
































































对本文算法进行抵抗常见信号处理攻击的鲁棒性测试,实验参量及结果见表 2. 这里列出了原始载体图像与攻击后的含水印图像之间的 PSNR,原始水印图像与从攻击后的含水印图像提取的水印图像

表 2 常见信号处理的实验结果




































Table 2 Experimental results for common signal processing

Image	Parameter		Adding gaussian noise (variance is 0.001)	Adding salt & pepper noise (intensity is 0.003)	Cropping (1/16 upper left corner)	Median filter (window size is 3×3)	Gaussian low-pass filter (window size is 3×3,σ=1)	JPEG compression	
	Algorithm	Attack type						(quality factor is 30)	(quality factor is 20)
Peppers	This paper (revision)	PSNR	29.8948	30.0726	16.8716	33.6282	31.5779	32.1334	31.1167
		Extracted watermark							
		NC	0.9982	0.9900	0.9126	0.9953	0.9994	0.9971	0.9575

(续表)

Image	Parameter Algorithm	Attack type	Adding gaussian noise (variance is 0.001)	Adding salt & pepper noise (intensity is 0.003)	Cropping (1/16 upper left corner)	Median filter (window size is 3×3)	Gaussian low-pass filter (window size is $3 \times 3, \sigma=1$)	JPEG compression	
								(quality factor is 30)	(quality factor is 20)
Peppers	Reference [5]	PSNR	29.7510	30.0122	15.8464	33.5734	31.3685	32.1414	31.1095
		Extracted watermark							
		NC	0.7971	0.9483	0.9953	0.9793	0.7887	0.9471	0.9360
	Reference [6]	PSNR	29.6390	30.1093	15.8486	33.8137	31.5362	32.3175	31.2596
		Extracted watermark							
		NC	0.8713	0.8746	0.9410	0.8754	0.9135	0.7991	0.7296
	Reference [7]	PSNR	29.7788	29.7417	15.8464	33.8028	31.5216	32.2859	31.2568
		Extracted watermark							
		NC	0.9971	0.9900	0.9775	0.9829	0.9924	0.9557	0.8772
	Reference [8]	PSNR	29.7742	30.1247	16.8679	33.5859	31.5363	32.1095	31.0974
		Extracted watermark							
		NC	0.9191	0.9328	0.7082	0.9655	0.8727	0.9539	0.9023
Lena	This paper (revision)	PSNR	29.9594	30.6796	18.6148	34.5831	32.7870	33.4595	32.2864
		Extracted watermark							
		NC	0.9965	0.9906	0.8914	0.9965	0.9982	0.9994	0.9679
	Reference [5]	PSNR	29.7982	30.6136	17.2771	34.5187	32.5635	33.4847	32.3030
		Extracted watermark							
		NC	0.8236	0.9613	0.9666	0.9557	0.8492	0.9606	0.8999
	Reference [6]	PSNR	29.7702	30.6547	17.2742	34.5835	32.6449	33.5137	32.3209
		Extracted watermark							
		NC	0.8513	0.8693	0.9243	0.8795	0.9156	0.7972	0.7565
	Reference [7]	PSNR	29.7253	30.4896	17.2761	34.6842	32.6887	33.6345	32.4192
		Extracted watermark							
		NC	0.9935	0.9858	0.9775	0.9734	0.9835	0.9272	0.8459
Reference [8]	PSNR	29.8253	30.2868	18.6154	34.5328	32.7392	33.4768	32.2736	
	Extracted watermark								
	NC	0.9132	0.9360	0.5785	0.9379	0.8688	0.9727	0.9076	

(续表)

Image	Parameter Algorithm	Attack type	Adding gaussian noise (variance is 0.001)	Adding salt & pepper noise (intensity is 0.003)	Cropping (1/16 upper left corner)	Median filter (window size is 3×3)	Gaussian low-pass filter (window size is $3 \times 3, \sigma=1$)	JPEG compression	
								(quality factor is 30)	(quality factor is 20)
Baboon	This paper (revision)	PSNR	29.8994	30.5325	17.7631	22.6317	23.6161	25.4332	24.2366
		Extracted watermark NC							
		NC	0.9977	0.9888	0.8939	0.9038	0.9582	0.9971	0.9515
	Reference [5]	PSNR	29.8162	30.5193	16.4366	22.8031	23.6295	25.4762	24.3092
		Extracted watermark NC							
		NC	0.8082	0.9484	0.9953	0.7628	0.7860	0.9403	0.8865
	Reference [6]	PSNR	28.6605	29.4257	16.4174	22.7504	23.5747	25.3168	24.2027
		Extracted watermark NC							
		NC	0.9323	0.9472	0.9667	0.8839	0.9311	0.9140	0.8887
	Reference [7]	PSNR	29.6034	30.2699	16.4305	22.7943	23.6187	25.4411	24.2866
		Extracted watermark NC							
		NC	0.9906	0.9793	0.9775	0.8733	0.9303	0.9242	0.8792
Reference [8]	PSNR	29.7738	30.4011	17.7628	22.6335	23.6161	25.4299	24.2377	
	Extracted watermark NC								
	NC	0.9151	0.9348	0.5927	0.6849	0.6623	0.9274	0.8384	

之间的 NC 和提取出的水印图像。由表 2 的第二栏可知,对于三幅测试图像,本文算法在抵抗添加高斯噪声、添加椒盐噪声、剪切、中值滤波、高斯低通滤波和 JPEG 压缩等常见信号处理攻击都表现出比较强的鲁棒性。

3.2 讨论

(1)量化步长 δ 的大小与本文算法的不可见性和抗攻击鲁棒性相关。 δ 值越小,本文算法的不可见性越好,但抗攻击鲁棒性越弱; δ 值越大,本文算法的不可见性越差,但抗攻击鲁棒性越强。因此, δ 值的选择要根据实际应用对不可见性和鲁棒性的要求采用实验的方法来确定。

(2)阈值 ρ 根据实际应用时版权认证要求的高低来确定,版权认证要求较高时, ρ 的取值应较大,反之, ρ 的取值应较小。根据表 1 和表 2 的实验结果,本文将 ρ 选择为 0.88,此时版权认证的要求比较高。

(3)将本文算法与文献[5-8]的算法进行抗攻击鲁棒性对比。为了精确比较,表 1 和表 2 中的数据都列出到小数点后四位。为使三个算法具有可比性,即使得原始图像和含水印图像之间的 PSNR 都相同,设置如下:

1)文献[5]:将图 2 作为原始水印图像;将图 3

(a)、图 4(a)、图 5(a)的三幅图像作为原始载体图像;水印只嵌入在每个子块的直流系数;分块大小为 16×16 ;图 3(a)、图 4(a)、图 5(a)三幅图像的量化步长 δ 分别选取为 40、44 和 41,与相应原始载体图像之间的 PSNR 都为 43 dB。

2)文献[6]:将图 2 作为原始水印图像;将图 3(a)、图 4(a)、图 5(a)的三幅图像作为原始载体图像;原始水印图像 Arnold 置乱次数为 8 次;以 harr 小波为基进行 DWT,小波分解级数为 2 级;分块大小为 8×8 ;logistic 混沌序列的初值 x_0 为 0.123 15,分支参量 μ 为 4,二值化阈值为 118.63;通过调整每个子块处于(7,8)和(8,8)的 DCT 系数嵌入水印;图 3(a)、图 4(a)、图 5(a)三幅图像的水印嵌入强度 α 和 β 分别选取为 0.4 和 0.8、0.4 和 0.6、0.3 和 0.5,与相应原始载体图像之间的 PSNR 都为 43 dB。

3)文献[7]:将图 2 作为原始水印图像;将图 3(a)、图 4(a)、图 5(a)的三幅图像作为原始载体图像;利用 logistic 混沌序列对仿射变换后的原始水印图像进行加密,初值为 0.123 16,分支参量为 3.6,二值化阈值为 101.896 5;以 harr 小波为基进行 DWT,分解级数为 1 级;分块大小为 8×8 ;水印嵌入在每个子块处于(4,6)的 DCT 系数;图 3(a)、图 4

(a)、图 5(a)三幅图像的量化步长 Q 分别选取为 20、18 和 15,与相应原始载体图像之间的 PSNR 都为 43 dB.

4)文献[8]:将图 2 作为原始水印图像;将图 3(a)、图 4(a)、图 5(a)的三幅图像作为原始载体图像;水印只嵌入在每个子块的第一个奇异值;分块大小为 16×16 ;图 3(a)、图 4(a)、图 5(a)三幅图像的量化步长 δ 分别为 41、44 和 40,与相应原始载体图像之间的 PSNR 都为 43 dB.

从表 1 可以看出,即使文献[8]的算法存在抗亮度和对比度调整修正,本文算法存在修正时在抵抗表 1 各种参量下的亮度和对比度调整的鲁棒性仍好于文献[8]的算法.另外,从表 1 可以看出,本文算法存在修正时在抵抗表 1 各种参量下的亮度和对比度调整的鲁棒性也好于文献[5-7]的算法.

根据表 2,从总体上看,本文算法抵抗添加高斯噪声、添加椒盐噪声、剪切、中值滤波、高斯低通滤波和 JPEG 压缩的鲁棒性强于文献[5-8]的算法.

4 结论

针对现有许多图像水印算法无法抵抗亮度和对比度调整,提出一种抗亮度和对比度调整的盲鲁棒量化有意义水印算法.算法具有良好的不可见性,而且提取水印时不需要原始载体图像,实现盲提取.由于检测端提取水印前先对攻击后的含水印图像进行抗亮度和对比度调整修正,所以算法在抵抗亮度和对比度调整上表现出较强的鲁棒性.实验结果还表明,算法在抵抗添加高斯噪声、添加椒盐噪声、剪切、中值滤波、高斯低通滤波和 JPEG 压缩也表现出较强的鲁棒性.

参考文献

- [1] WANG Xiang-yang, HOU Li-min, WU Jun. A feature-based robust digital image watermarking against geometric attacks [J]. *Image and Vision Computing*, 2008, **26**(7):980-989.
- [2] WANG Xiang-yang, YANG Yi-ping, YANG Hong-ying. Invariant image watermarking using multi-scale Harris detector and wavelet moments[J]. *Computers and Electrical Engineering*, 2010, **36**(1):31-44.
- [3] ZHOU Ya-Xun, YE Qin-wei, XU Tie-feng. A new scheme of image watermarking based on wavelet and cosine transform [J]. *Acta Electronica Sinica*, 2001, **29**(12):1693-1695.
- [4] LI Hai-yan. Digital watermarking algorithms based on DCT and DWT [J]. *Journal of Hefei University (Natural Sciences)*, 2009, **19**(3):37-39.
李海燕. 基于 DCT 和 DWT 域的数字水印算法[J]. 合肥学院学报(自然科学版), 2009, **19**(3):37-39.
- [5] LI Xu-dong. Blocked DCT and quantization based blind image watermark algorithm[J]. *Computer Engineering*, 2006, **32**(21):139-140,144.
李旭东. 基于分块 DCT 和量化的图像盲水印算法[J]. 计算机工程, 2006, **32**(21):139-140,144.
- [6] DU Xiao-shan, LIAO Shu-jian. A novel image watermarking algorithm based on DWT and DCT[J]. *Computer Technology and Development*, 2011, **21**(1):147-150.
杜肖山, 廖述剑. 一种 DWT 与 DCT 相结合的图像水印算法[J]. 计算机技术与发展, 2011, **21**(1):147-150.
- [7] WANG Li-ping, YU Dong-zhi, LIU Si-qi. A digital image watermarking algorithm based on the combination of the DWT and the DCT[J]. *Journal of Xi'an University of Posts and Telecommunications*, 2010, **15**(3):112-115.
王立平, 喻东芝, 刘思奇. 一种 DWT 和 DCT 相结合的数字图像水印算法[J]. 西安邮电学院学报, 2010, **15**(3):112-115.
- [8] LI Xu-dong. Gray-level digital watermarking algorithm based on SVD[J]. *Geomatics and Information Science of Wuhan University*, 2010, **35**(11):1305-1308,1359.
李旭东. 基于奇异值分解的灰度级数字水印算法[J]. 武汉大学学报[J]. 信息科学版, 2010, **35**(11):1305-1308,1359.
- [9] DING Wei, YAN Wei-qi, QI Dong-xu. Digital image scrambling technology based on Arnold transformation [J]. *Journal of Computer Aided Design & Computer Graphics*, 2001, **13**(4):338-341.
丁玮, 阎伟齐, 齐东旭. 基于 Arnold 变换的数字图像置乱技术[J]. 计算机辅助设计与图形学学报, 2001, **13**(4):338-341.
- [10] YE Tian-yu, MA Zhao-feng, NIU Xin-xin, et al. A zero-watermark technology with strong robustness[J]. *Journal of Beijing University of Posts and Telecommunications*, 2010, **33**(3):126-129.
叶天语, 马兆丰, 钮心忻, 等. 强鲁棒零水印技术[J]. 北京邮电大学学报, 2010, **33**(3):126-129.
- [11] YE Tian-yu. A robust zero-watermarking algorithm against dual print-and-scan process based on discrete cosine transformation [J]. *Acta Photonica Sinica*, 2011, **40**(1):142-148.
叶天语. 离散余弦变换域抗二次打印-扫描鲁棒零水印算法[J]. 光子学报, 2011, **40**(1):142-148.
- [12] YE Tian-yu. A robust zero-watermarking algorithm using variance in singular value decomposition domain [J]. *Acta Photonica Sinica*, 2011, **40**(6):961-966.
叶天语. 基于方差的奇异值分解域鲁棒零水印算法[J]. 光子学报, 2011, **40**(6):961-966.

A Blind Robust Quantization-based Watermarking Algorithm Against Brightness-and-contrast Adjustment

YE Tian-yu

(College of Information & Electronic Engineering, Zhejiang Gongshang University, Hangzhou 310018, China)

Abstract: Lots of existing image watermarking algorithm cannot resist brightness-and-contrast adjustment. In order to solve the problem, a blind robust quantization-based watermarking algorithm against brightness-and-contrast adjustment is proposed. An original image is transformed with discrete wavelet transform, and its low frequency band is split into non-overlapping blocks. Then, each block is conducted with discrete cosine transform. Finally, a watermark is inserted into low frequency coefficient from each block after discrete cosine transform through odd-even quantization. At the detection end, revision for resistance against brightness-and-contrast adjustment is imposed on the attacked watermarked image at first, and then a watermark is blindly extracted through odd-even judgment. Experimental results show that it has strong robustness towards brightness-and-contrast adjustment, and adding gaussian noise, adding salt&pepper noise, cropping, median filter, gaussian low-pass filter and JPEG compression.

Key words: Digital watermarking; Quantization; Robustness; Brightness-and-contrast adjustment