第 29 卷 第 5 期
2003 年 9 月

自 动 化 学 报
ACTA AUTOMATICA SINICA

Vol. 29, No. 5
Sept., 2003

Image Watermarking Resist to Geometrical Attacks[1)]

KANG Xian-Gui HUANG Ji-Wu

(*Department of Electronic and Communication Engineering, Zhongshan University, Guangzhou* 510275)

(E-mail: {isskxg, isshjw}@zsu.edu.cn)

Abstract An image watermarking algorithm in DWT domain is proposed which is robust against geometric distortion. We hide 536 information bits in a $512 \times 512 \times 8$ image. The merits of the proposed algorithm are as follows: 1) By introducing distance between the attacked image and the original image, we can resynchronize the data extraction based on the minimum distance; 2) Using multi-resolution matching and coarse-fine searching to prune the searching space, the computation of our algorithm is drastically reduced; 3) With BCH coding and 2-D interleaving technology the algorithm can correct some random errors and bursting errors in detection. The watermark thus generated is invisible and performs well in StirMark test. Compared with other watermarking algorithms reported in the literature, the algorithm is more robust, especially against geometric distortion, while having much higher bit rate.

Key words Image watermark resynchronization, image distance, DWT, geometric distortion

1 Introduction

Digital watermarking is an effective method for protection of intellectual property rights (IPR) of multimedia data[1]. Invisible image watermarking is now one of the most active fields. Watermarking in the transform domain is the most popular scheme, which includes watermarking based on discrete cosine transform domain (DCT), discrete Fourier transform (DFT), and discrete wavelet transform (DWT)[2]. Robustness is one of the key issues in watermarking, and robustness against geometric distortion is a difficult issue. Recently, it has become clear that even very small geometric distortions may impair watermark detection [3]. To improve the robustness against geometric distortion, a lot of attempts have been tried. However, there still remain a number of problems such as low bit rate of hidden data, low robustness, etc. For example, the watermarking proposed in [3, 5] can only resist rotation, scaling and translation (RST), and it is not compatible with the new image compression standard which adopts wavelet transform. The non-blind solution proposed by Davoine *et al.* is to split the original (or the non-geometrically distorted marked) image into a set of triangular patches[6]. This mesh of patches then serves as the reference mesh during a pre-processing step of the mark signal retrieval. As emphasized by the authors, however, this kind mesh based compensation is only efficient in cases involving minor deformations. It is known that DWT-based marking algorithms are popular due to their good spatial-frequency characteristics of DWT and due to the DWT's importance in image/video compression standards[7, 8]. Since the DWT coefficients are not invariant under geometric distortion, the existing watermarking in the DWT domain cannot resist geometric transforms.

By introducing a distance (dissimilarity) measure between the attacked image and the unattacked one, we propose a watermarking algorithm in the DWT domain, which can resist geometric attacks. It incorporates error correction coding, 2-D interleaving and resynchronization based on the distance measure. The generated watermark, containing 536 in-

1) Supported by National Natural Science Foundation of P. R. China (69975011, 60172067 and 60133020), "863" Program (2002AA144060), Natural Science Foundation of Guangdong (980442, 021758), Foundation of Ministry of Education

Received September 10, 2002; in revised form December 28, 2002

收稿日期 2002-09-10; 收修改稿日期 2002-12-28

formation bits, is robust to most test functions in StirMark 3.1, especially robust to geometric distortions.

2 Watermark embedding

A meaningful watermark, ***CS***, consists of a character string with length L, i. e., $\boldsymbol{CS}=\{CS_i;\ 0\leqslant i<L\}$, where CS_i is a character (8 bits). The data hiding procedure is demonstrated in Fig. 1.

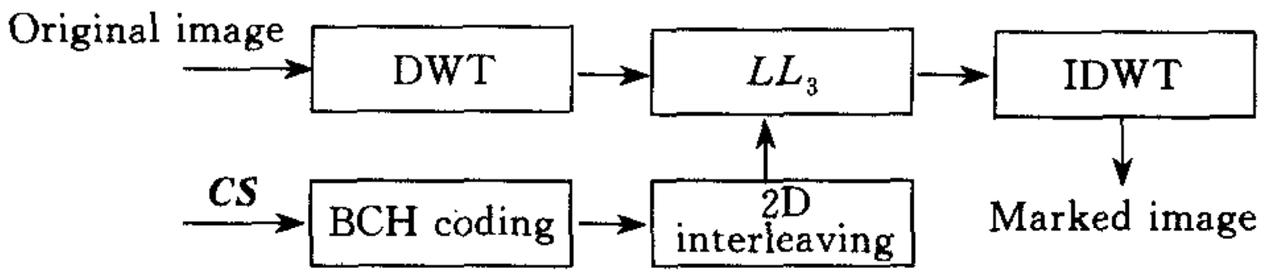


Fig. 1 Watermark embedding process

To lower detection errors, the watermark signal is coded with a BCH $(n,8)$ code. We choose n to be 61. After BCH coding, we obtain the binary data $\boldsymbol{W}$.

To enhance the robustness of watermarking against bursting errors, 2-D interleaving is applied to the watermark. Bursting of errors do occur when a watermarked image is cropped or jitter attacked. 2-D interleaving can spread bursting errors among different codewords[11]. With a simple random error-correction code such as the BCH (61,8), the spread error bits within a codeword may be less than 13 bits so that the bursting errors can be corrected. After 2D interleaving, we convert $\boldsymbol{W}$ into $\boldsymbol{X}$.

By using the Daubechies 9/7 bi-orthogonal wavelet filters we apply three-level DWT to the original image. The DWT coefficients in the LL_3 subband are scanned row by row to form a 1-D array $\boldsymbol{C}$. We adopt Equation (1) to embed the binary data $\boldsymbol{X}$ into $\boldsymbol{C}$ to obtain $\boldsymbol{C}'$:

$$\begin{cases} C'(i) = C(i) - C(i)\bmod \alpha + \dfrac{3}{4}\alpha, & \text{if } x_i = 1 \\ C'(i) = C(i) - C(i)\bmod \alpha + \dfrac{1}{4}\alpha, & \text{if } x_i = 0 \end{cases} \qquad 0 \leqslant i < L \times n \tag{1}$$

where $C(i)$ and $C'(i)$ denote the ith element in $\boldsymbol{C}$ and $\boldsymbol{C}'$, respectively. In our experiment, they are all positive. For the watermark to be robust, α should be maximized under the constraint of invisibility. In our work, we choose α to be 32 and 64 for Lena and Baboon, respectively. Performing inverse DWT on the modified image, we obtain a watermarked image.

3 Watermark extraction based on resynchronization

It is known that geometric distortion damages geometric synchronization of hidden data that is necessary in watermark extraction. To resynchronize the hidden data, we perform an anti-attack operation to remove the geometric distortion on the watermarked image by searching for the minimum distance (best matching) between the attacked stego-image and a reference image. The reference image may be an uncorrupted marked image or the original image.

3.1 Resynchronization based on minimum distance searching

The distance d between an image $f(x,y)$ and another image $f'(x,y)$ (both are $m\times n$ 2-D arrays) is defined as the mean absolute difference of gray values:

$$d = \left(\sum_{x=1}^{m}\sum_{y=1}^{n} \left| f(x,y) - f'(x,y) \right| \right)/(m\times n) \tag{2}$$

d serves as a measure of the similarity of the two images. The smaller d, the more

similar they are. The d between an image and itself is 0. Considering an image $f(x,y)$ and its geometric distorted version $f'(x,y)$, if ideally at the point of registration, then $f'(x,y)=f(x,y)$ and the distance is 0. In practice, at the point of registration, since $f'(x,y)$ may be slightly different from the $f(x,y)$ due to image noise, e. g., additive noise, image compression or interpolation, the distance is not 0, but it is reasonable to assume that the distance is minimal (this is also supported by our experiment). The stego-image may be subject to various types of geometrical distortions: RST, shearing, general linear transform, etc. RST is a combination of rotation, scaling, cropping and translation. Because the geometric distortion is unknown, we apply several anti-attack operations to the attacked marked image $g(x,y)$ separately to generate different image (denoted by $g'(x,y)$). By searching for the minimum distance between $g'(x,y)$ and the reference image $f(x,y)$ we determine which kind of inverse operations (inverse RST, inverse shearing, inverse general linear transform) should be applied to the attacked image and what parameters of the inverse operation should be chosen in order to regain geometric synchronization. When we compute the distance between $f(x,y)$ and $f'(x,y)$, we can just use a portion instead of the whole image in practice in order to save computation.

3.2 Searching for the resynchronization parameters

Among the above-mentioned several anti-attack operations, the anti-RST is much more time consuming since it has a 4-dimensional searching space (S,θ,x_o,y_o), where S is the size of the scaled image (assuming equal scaling along x and y directions) and hence a scaling related parameter (corresponding to σ), θ denotes the rotation angle, x_o and y_o denote the translation parameters. So in what follows, we mainly discuss the searching process of the anti-RST operation. In the whole searching algorithm, we use the bi-linear interpolation.

Anti-RST operation We perform the following RST transform on the to-be-checked image $g(x,y)$ with different (σ,θ,x_o,y_o)parameters:

$$g'(x,y)=g(\sigma(x\cos\theta+y\sin\theta)-x_o,\quad \sigma(-x\sin\theta+y\cos\theta)-y_o)$$

and search for the minimum distance (d) between $g'(x,y)$ and $f(x,y)$. In our work, we substitute σ with S. Without loss of generality, we choose the size of $f(x,y)$ to be 512×512. We choose the searching range of S to be $512\sim256$ (the corrupted image is assumed to be cropped less than 75%). The searching range of translation parameters x_o and y_o are determined by the above S, thus to be $-128\sim+128$ (the center of an image is the origin). θ: $-90°\sim+90°$. We drastically decrease the computation using multi-resolution matching and coarse-fine searching. The anti-RST operation is divided into five phases: coarse searching at 64×64 resolution, medium_1 searching at 128×128 resolution, medium_2 searching at 256×256 resolution, fine_1 searching and fine-2 searching at 512×512 resolution. The algorithms at different phases are similar. It contains 4 nesting cycles (S,θ,x_o,y_o), where the cycle of changing S is the outermost cycle and the one of changing y_o is the innermost one.

1)Convert $f(x,y)$ to a certain resolution first to generate $f_r(x,y)$.

2)Obtain a $size_d\times size_d$ sub-image $g'_{sub}(x,y)$, using gray-level interpolation via pixel-filling (also referred to as backward mapping) algorithm from the to-be-checked image $g(x,y)$, which has been rescaled to the size of $S\times S$ and re-rotated by an angle θ.

3)Select a $size_d\times size_d$ sub-image $f'_{sub}(x,y)$ in the image $f_r(x,y)$, whose center shifts from the center of $f_r(x,y)$ by x_o and y_o.

4)Compute d between $g'_{sub}(x,y)$and $f'_{sub}(x,y)$.

5)Change S, θ, x_o, and y_o, respectively with the searching ranges and step sizes given below, repeat steps 2, 3 and 4 given above, searching for the minimum d.

6)The parameters S, θ, x_o, and y_o corresponding to the minimum d at this phase can

be determined. This candidate of the best matching is then propagated to the next searching phases.

In coarse searching, we choose $size_d$ to be 22 (about 0.35×64). We choose searching ranges of S, θ, x_o, y_o to be $64\sim32$, $-90°\sim+90°$, $-16\sim+16$, $-16\sim+16$, respectively, the step sizes to be 8, 5°, 1, 1, respectively. After coarse searching, we obtain the parameters S_c, θ_c, x_{oc}, y_{oc}.

In medium_1 searching, we choose $size_d$ to be $0.7(2\ S_c-16)$. We choose searching ranges of S, θ, x_o, y_o to be $(S_c\pm8)\times2$, $\theta_c\pm5°$, $(x_{oc}\pm2)\times2$, $(y_{oc}\pm2)\times2$, respectively, the step sizes to be 2, 1°, 1, 1, respectively. After medium_1 phase, we obtain the parameters S_{m1}, θ_{m1}, x_{om1}, y_{om1}.

In medium_2 searching, we choose $size_d$ to be $0.7(2\ S_{m1}-4)$. We choose searching ranges of S,θ, x_o, y_o to be $(S_{m1}\pm2)\times2$, $\theta_{m1}\pm1°$, $(x_{om1}\pm2)\times2$, $(y_{om1}\pm2)\times2$, respectively, the step sizes to be 1, 0.5°, 1, 1, respectively. After medium_2 phase, we obtain the parameters S_{m2}, θ_{m2}, x_{om2}, y_{om2}.

In fine-1 searching, we choose $size_d$ to be $0.7(2\ S_{m2}-2)$. We choose searching ranges of S, θ, x_o, y_o to be $(S_{m2}\pm1)\times2$, $\theta_{m2}\pm0.4°$, $(x_{om2}\pm2)\times2$, $(y_{om2}\pm2)\times2$, respectively, step sizes to be 1, 0.2°, 1, 1, respectively. After fine-1 phase, we obtain the parameters S_{f1}, θ_{f1}, and the final parameters of translation x_o, y_o.

fine-2 searching contains only 2 nesting cycles (S,θ). We choose $size_d$ to be $0.9S_{f1}$, the searching ranges of S,θ to be $S_{f1}\pm1$, $\theta_{f1}\pm0.1°$, step sizes to be 1, 0.02°, respectively. After Fine-2 phase, we obtain the final parameters of scaling and rotation S,θ.

Anti-shearing operation. By changing the anti-shearing parameters (the maximum shift in X and/or Y directions), we perform different anti-shearing operations to the attacked image $g(x,y)$ to obtain $g'(x,y)$. When the distance between $g'(x,y)$ and $f(x,y)$ reaches its minimum, the parameters for resynchronization can be determined. If the size of $g(x,y)$ is $N\times N$, searching range of the shift in X, Y directions is $(512-N)-2\sim(512-N)+2$ (the accuracy is 0.1 pixel). To reduce the computation, we also adopt the coarse-fine searching algorithm.

Anti-general-linear-transform operation. Flip the image $g(x,y)$ in up/down direction, perform shearing on the flipped image, resize the image to the same size of the original image, flip the resized image, and we obtain the image $g'(x,y)$. We change the shearing parameters while performing anti-general-linear-transform operation. We obtain the parameters for resynchronization based on the minimum distance between $g'(x,y)$ and $f(x,y)$.

3.3 Data extraction

First, perform 3-level DWT on the resynchronized to-be-checked marked image. The coefficients of the LL_3 subband are scanned and turned into a 1-D array, which is denoted by $\boldsymbol{C}^*$. Then we can extract the hidden binary data $\boldsymbol{X}^*$ $\{x_i^*\}$ according to the following formula:

$$x_i^*=+1,\ \text{if}\ (C^*(i)\bmod\alpha)\geqslant\frac{\alpha}{2};\quad x_i^*=-1,\ \text{otherwise} \tag{3}$$

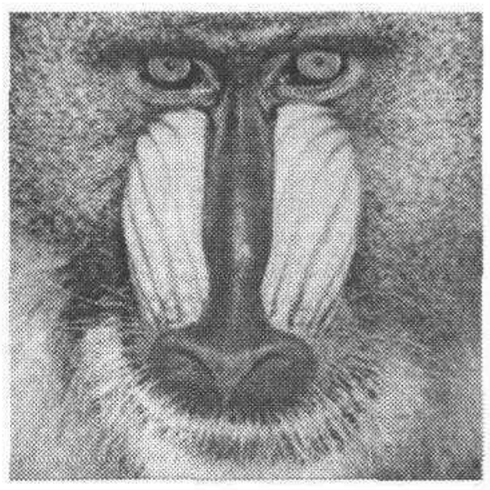
Next we perform 2-D de-interleaving to $\boldsymbol{X}^*$, which is the inverse process of 2-D interleaving, to obtain the binary sequence $\boldsymbol{W}^*$. By partitioning $\boldsymbol{W}^*$ into subsequences with size of $n=61$, we obtain the extracted signals $\boldsymbol{W}_i^*=\{w_{ij}^*, 0\leqslant j<n, 0\leqslant i<L\}$. Finally, by searching for a codeword that is closest to $\boldsymbol{W}_i^*$ in BCH codebook in the sense of Hamming distance, we can decide the possibly embedded byte CS_i^*.

4 Experimental results

We have tested the proposed algorithm on various images. The results on $512\times512\times8$ Lena and Baboon images are reported in this paper. A string of 67 characters (536 bits)

are embedded into the images. The watermarks are perceptually invisible (Fig. 2). Even if the marked image has undergone very small geometric distortions, the watermark cannot be recovered successfully without the proposed resynchronization algorithm. For example, the character error rate (CER) of the decoded watermark without resynchronization is 51% when the marked image is rotated by 0.25°. When the rotation angle is 0.5°, the CER is 93%. In Fig. 3, the image has undergone a rotation of 10°(auto-crop, auto-scale), the CER of decoded watermark is 100% without resynchronization, but after the proposed geometric registration, we can recover the watermark with no error. The watermark can also resist the combination of rotation, scaling, translation and cropping (RST) and be decoded with no error when the cropped portion is less than 40%. Table 1 shows test results in terms of BER (bit error rate) with StirMark 3.1. The watermark (536bits) can effectively resist attacks such as JPEG compression, Gaussian filtering, rotation (auto-crop, auto-scale), aspect ratio variations, scaling, jitter attack (combination of random rows and columns removal), general linear transform, and shearing. The watermark can be error-freely recovered from the marked images when they are attacked by aspect ratio variation, scaling, Gaussian filtering, jitter attack. It is noted that our algorithm can recover embedded characters error-freely for a JPEG compression quality factor of 20 for Baboon and 35 for Lena.

(a) Watermarked Lena image (42.5 dB)



(b) Watermarked Baboon image (36.6 dB)

Fig. 2 Watermarks

(a) A distorted version of Fig. 2(a), the watermark cannot be decoded

(b) The recovered image, the watermark can be decoded error—freely

Fig. 3 Resynchronization

Table 1 Experimental results with StirMark 3.1

StirMark functions	BER		StirMark functions	BER	
	Lena	Baboon		Lena	Baboon
jitter, scaling	0	0	linear–1.013–.008–.011–1.008	0.07	0
aspect ratio change	0	0	linear–1.010–0.013–0.009–1.011	0.03	0
Gaussian filtering	0	0	linear–1.007–0.01–0.01–1.012	0.09	0
cropping–10,20	0	0	shearing–x–1(5)–y–1(0)	0	0
cropping– 25	0.04	0	shearing–x–5.0–y–5.0	0.08	0
FMLR	0.49	0.01	rotation–(scale–)10°~0.25°,90°	0	0
3×3–median–filter	0.38	0.47	rotation–(scale–)15°	0.02	0
sharpening	0.30	0.02	rotation–(scale–)30°	0.004	0.02
random–bend	0.53	0.55	rotation–(scale–)45°	0.13	0.03

The schemes proposed by Lin *et al.*[3] and Liu *et al.*[4] hide only one information bit (whether the watermark exits). The scheme in [5] hides 60 bits in an image. However, the resulting stego image quality is poor due to interpolation errors when applying LPM (logar polar mapping)and ILPM (inverse LPM) on the image.[3,5]. Pereira and Pun[10] proposed a template matching method to recover the original shape of an image, but it can only recover the RST distortion, and can only recover a rotation of 0.2°, while our method can recover a rotation of 0.02°. The shortcoming of the proposed watermarking is computation load in watermark extraction introduced by resynchronization. But the watermark extraction takes less than 20s (PⅣ 1.4G). This is sufficiently fast for commercial applications.

5 Conclusions

In this paper, we have proposed a resynchronization scheme based on minimum distance to resist geometric distortion in image watermarking. It incorporates multi-resolution matching and coarse-fine searching, achieving low computation complexity. Compared with the existing schemes resisting geometric distortion, the major advantage of the proposed scheme is that it can cope with a large amount of geometric distortion with a better accuracy. Applying the resynchronization scheme, BCH coding and 2D interleaving, we generate watermark which has much higher bit rate and is more robust, especially against geometric distortion, than other watermarking algorithms reported in the literature. In addition, the proposed resynchronization scheme can be applied to watermarking in DCT domain to resist geometric attacks as well.

However, robustness of the present watermarking against median filtering, FMLR, sharpening and randomization-and-bending remains to be improved. According to our experiment, robustness against median filtering and FMLR can be improved by increasing the strength of the watermark (via adaptively embedding the watermark [11]) and soft decoding. Future work for the improvement will be needed.

References

1 Cox I J, Killian J, Leighton F T, Shamoon T. Secure spread spectrum watermarking for multimedia. *IEEE Transactions on Image Processing*, 1997, **6**(12): 1673～1687

2 Huang Ji-Wu, Tan Tie-Niu . The survey of invisible image watermarking. *Acta Automatica Sinica* , 2000,**26**(5): 645～655 (in Chinese)

3 Lin C-Y, Wu M, Bloom J A, Cox I J, Miller M L, Lui Y M. Rotation, scale, and translation resilient watermarking for images. *IEEE Transactionis on Image Processing*, 2001,**10**(5): 767～782

4 Liu Rui-Zhen, Tan Tie-Niu. The digital image watermarking based on the singular value decomposition. *Acta Electronic Sinica*, 2001,**29**(2): 168～170 (in Chinese)

5 Ruanaidh J J K ó, Thierry Pun. Rotation, scale and translation invariant spread spectrum digital image watermarking. *Signal Processing*, 1998, **66**(3): 303～317

6 Davoine F, Bas P, Hébert P-A, Chassery J-M. Watermarking et résistance aux déformations géométriques. In: Dugelay J-L, ed. Cinquiémes journées d'études et d'échanges sur la compression et la représentation des signaux audiovisuals (CORESA'99),France: Sophia-Antipolis, 1999

7 Niu Xia-Mu, Lu Zhe-Ming, Shun Sheng-He. The watermarking based on the multi-resolution analysis. *Acta Electronic Sinica*, 2000, **28**(8): 1～4(in Chinese)

8 Tsai M J, Yu K -Y, Chen Y-Z. Joint wavelet and spatial transformation for digital watermarking. *IEEE Transactions on Consumer Electronics*, 2000, **46**(1): 241～245

9 Elmasry G F, Shi Yun Q. 2-D interleaving for enhance the robustness of watermark signals embedded in still image. In: Proceeding of IEEE International Conference on Multimedia and Exposition, 2000, Tokyo

10 Pereira S, Pun T. An iterative template matching algorithm using the chirp-Z transform for digital image watermarking. *Pattern Recognition*, 2000, **33**(1): 173～175

11 Huang Ji-Wu, Shi Yun Q. An adaptive image watermarking scheme based on visual masking. *IEE Eletronics Letters*, 1998, **34**(8): 748～750

KANG Xian-Gui Received his bachelor degree and master degree from Peking University and Nanjing University, in 1990 and 1993, respectively. He is with the Department of Electronic and Communication Engineering, Zhongshan University. His research interests include image processing, data hiding and watermarking.

HUANG Ji-Wu Received his bachelor degree from Xidian University, in 1982, master degree from Tsinghua University, in 1987, and Ph. D. degree from Chinese Academy of Science in 1998. Professor Huang is with the Department of Electronic and Communication Engineering, Zhongshan University. He is an IEEE senior member and member, Technique Committee of Multimedia Systems and Applications, IEEE CAS Society. His research interests include image processing, image coding, data hiding and watermarking.

抗几何变换的有意义图像水印算法

康显桂　　黄继武

(广州市中山大学电子与通信工程系　广州　510275)

(E-mail: {isskxg, isshjw}@zsu.edu.cn)

摘　要　提出了一个基于 DWT 的有意义稳健图像水印算法. 1)采用基于最小距离的水印重同步技术,能抵抗 RST, shearing, general linear transformation 等几何攻击;2)采用多分辨率匹配和粗细搜索相结合的方法,有效降低算法的计算量;3)应用二维交织技术和 BCH 编码,具有纠正随机错误和突发错误的能力. 把一个 536 bits 信息构成的字符串水印嵌入到 512×512×8 bits 的图像中,所实现的水印在抵抗 StirMark 攻击中达到了较好的性能. 相对于目前的一些水印算法,具有隐藏数据量大且好,对抗几何变换具稳健性等优点.

关键词　重同步,图像距离,DWT,几何攻击

中图分类号　TP391