DIGITAL WATERMARKING BASED ON LOCALLY LINEAR EMBEDDING

Yonggang Fu, Ruimin Shen, Liping Shen

Dept. of Computer Science and Engineering, Shanghai Jiaotong University, Shanghai, China
E-mails: {fyg,rmshen, lpshen}@mail.sjtu.edu.cn

ABSTRACT

Robustness is one of the crucial issues in digital watermarking. Especially the robustness against geometric distortion and JPEG compression at the same time remains challenging. In this paper, a Locally Linear Embedding (LLE) based watermarking algorithm that is robust against affine transformation is proposed. This algorithm improves the robustness via the intrinsic robustness of the LLE. A random generated watermark is embedded in the coefficients of reconstruction weights of the locally linear embedding. In watermark extraction, the watermark can be extracted almost the same process as the watermark embedding. Experimental results have demonstrated that the proposed watermarking scheme is more robust than other watermarking algorithms reported in the literature. Specifically, it is robust against almost all affine transform related testing functions in StirMark 3.1. While the approach is presented for gray-level images, it can be applied to color images and video sequences.

Key words: Digital watermarking, Locally linear embedding

1. INTRODUCTION

The widespread of the Internet and the popular use of CD-ROM have made the protection and enforcement of digital intellectual property rights an important issue in the 'digital world'. Digital watermarking has been used to address this issue through the embedding of a secret signal in a digital image or video sequence to claim the ownership. Some nice surveys of recent work are given in the issue [1] and Wu's works [2].

As a watermarking scheme, in order to be useful, it must be robust against a variety of possible attacks by pirates. These include robustness against compression such as JPEG, scaling and aspect ratio changes, rotation, cropping, row and column removal, addition of noise, filtering, cryptographic and statistical attacks, as well as insertion of other watermarks. Many of the current techniques for embedding marks in digital images have been inspired by methods of image coding and compression. Information has been embedded using the discrete cosine transform (DCT), discrete wavelet transform (DWT), as well as in the spatial domain. While these methods perform well against compression, they lack robustness to geometric attacks. Consequently, methods have emerged which exploit the properties of the discrete Fourier transform (DFT) to achieve robustness against rotation and scaling. The DFT methods can be divided into two classes, those based on invariance [3,4] and those based on embedded template in the image which is searched during the watermark detection and yields information about the transformation undergone by the image [5,6].

Unlike algorithms using DFT to recover watermark bits from rotated, scaled, and aspect ratio adjusted host, we propose a new feature inspired by the newly developed method of locally linear embedding (LLE)[7]. The key part of the watermarking scheme is the extraction of the embedding weights among the image columns. The proposed method is evaluated relative to the benchmark implemented in the software package Stirmark3 [8]. The algorithm performs very well relative to the extensive series of tests implemented in the benchmark.

The rest of this paper is structured as follows. In Section 2 we have an overview of locally linear embedding algorithm and some of its good properties. The embedding approach is depicted in Section 3. Section 4 describes the extraction of the watermark. In Section 5, we present results. Finally, Section 6 contains our conclusions.

2. AN OVERVIEW TO LOCALLY LINEAR EMBEDDING

The LLE algorithm is first proposed and developed by Sam T.Roweis et. al[7], the details can be found in[9,10]. The whole LLE algorithm is based on simple geometric intuitions. Suppose the data consists of N real valued vectors $\vec{X}_i$ with dimension D, sampled from some underlying manifold. Since every sampled data point and its neighbors can be deemed to lie or close to a locally linear patch of the manifold, we can characterize the local geometry of these patches by linear coefficients that reconstruct each data point from its neighbors. The construction errors are measured by the following cost function

$$\varepsilon(W) = \sum_i | \vec{X}_i - \sum_j W_{ij} \vec{X}_j |^2 \qquad (1)$$

which adds up the squared distances between all the data points and their reconstruction. The weights W_{ij} summarize the contribution of the *j*th data points to the *i*th reconstruction. By minimizing the cost functions under the constraints of only considering K-neighbors, the optimal weights subject to this constraint are found by solving a least-squares problem.

In the final step of the algorithm, each high dimensional observation $\vec{X}_i$ is mapped to a low dimensional vector $\vec{Y}_i$ representing global internal coordinates on the manifold. This is done by choosing d-dimensional coordinates $\vec{Y}_i$ to minimize the embedding cost function

$$\phi(Y) = \sum_i | \vec{Y}_i - \sum_j W_{ij} \vec{Y}_j |^2 \qquad (2)$$

This cost function, like the previous one, is based on locally linear reconstruction errors, but here we fix the weights W_{ij} while optimizing the coordinates $\vec{Y}_i$. The embedding cost in equation (2) defines a quadratic form in the vector $\vec{Y}_i$. Subject to constraints that make the problem well posed, it can be minimized by solving sparse N*N eigen-value problem. The implementation of the algorithm is straightforward [9,10].

It can be easily found out that the reconstruction weights W_{ij} reflect the intrinsic geometric properties of the data that are invariant to several geometric transformations. So it is a proper feature to insert some watermark message which will be robust to geometric attacks. Hence we propose these good features W_{ij} to embed the watermark.

3. WATERMARK EMBEDDING PROCESS

We take a 8-bit gray image as original image for the following implementation. Watermark-embedding is implemented via the following four steps: i) image is spited to columns and finding the K-nearest neighbors of every column vector; ii) finding the best reconstruction weights; iii) watermark generation and casting; iv) find the embedding vector for the fixed weights and reconstruct the watermarked image.

We first split the original $M \times N$ image I(*x,y*) into vectors by each column as one vector. Since every column has great relevance to its neighbor columns in an image, we expect each data point and its neighbors to lie on or close to a locally linear patch of the manifold. The local geometry of these patches can be characterized by linear coefficients that reconstruct each data point from its neighbors. In the simplest formulation of LLE, one identifies K nearest neighbors per data point, as measured by Euclidean distance.

Reconstruction errors are then measured by the cost function (1), which adds up the squared distances between all the data points and their reconstructions. To compute the weights W_{ij}, we minimize the cost function subject to the constraints, that each data point $\vec{X}_i$ is reconstructed only from its neighbors, enforcing $W_{ij} = 0$ if $\vec{X}_j$ does not belong to this set. Here is different from the original LLE algorithm, because of the efficiency of watermark embedding in the following steps.

The optimal weights W_{ij} subject to the constraint are found by solving a least squares problem. Note the constrained weights that minimize these reconstruction errors obey an important symmetry: for any particular data points, they are invariant to rotation, and rescaling of that data point and its neighbors. A consequence of this symmetry is that the reconstruction weights characterize intrinsic geometric properties of each neighborhood, as opposed to properties that depend on a particular frame of reference.

Once we get the weights, the image can be reconstructed only by the weights. Further more, it has some good robustness, so the watermark can be embed into the weights derived from the image with a slight strength, then the watermark will be robust to general geometric attacks.

The embedded watermark is first generated from the copyright owner's key k_1 and further modulated into final bipolar watermark *WM* as follows:

$$WM = \{w_1, w_2, ..., w_n\}, \qquad w_i \in \{-1, 1\} \qquad (3)$$

where *WM* is a binary pseudo-random sequence of length n.

In order to avoid significant visual artifacts to the watermarked image, the watermark should not be embedded into all the weights values, but only among those weights have non-zero values. Hilbert scan is adapted and the two dimensional nonzero weights are reordered into a one dimension sequence. Let those non-zero weights after rescanning be $W = \{W_1, W_2, ..., W_T\}$, where $T = N \times K$ is the total number of the nonzero weights. We choose the watermark host in a random way decided by another key k_2 denoted as $W' = \{W_1', W_2', ..., W_n'\}$. The watermark is embedded as follows:

$$W_i^* = W_i'(1 + \alpha w_i) \quad i = 1, ..., n \qquad (4)$$

where $W_i^* s$ are watermarked weights, and $\alpha = 0.05$ is a factor to control the watermark strength.

In the final step of the algorithm, the watermarked image is reconstructed by the watermarked weights. This

is done by choosing M-dimensional coordinates to minimize the embedding cost function (2). This cost function—like the previous one—is based on locally linear reconstruction errors, and here we fix the weights W_{ij}, while optimizing the coordinates.

Note that the cost function (2) defines a quadratic form,

$$\phi(Y) = \sum_{i,j} M_{ij} (\vec{Y}_i \cdot \vec{Y}_j) \quad (5)$$

involving inner products of the embedding vectors and the matrix M.

$$M_{ij} = \delta_{ij} - W_{ij}^* - W_{ji}^* + \sum_k W_{ki}^* W_{kj}^* \quad (6)$$

This optimization is performed subject to constraints that make the problem well posed. To avoid degenerate solutions, we constrain the embedding vectors to have unit covariance, with outer products satisfying $\frac{1}{N}\sum_i \vec{Y}_i \vec{Y}_i^T = E$, where E is the N*N identity matrix. The further constraint that the covariance is equal to the identity matrix expresses an assumption that reconstruction errors for different coordinates in the embedding space should be measured on the same scale. It can be minimized by solving an N*N sparse eigenvector problem.

4. WATERMARK DETECTION

In the proposed watermark detection scheme, copyright is determined by using a correlation technique. Like the embedding process, the test image and original image are firstly divided into column vectors and find their K-nearest neighbors. By minimizing the reconstruction errors measured by the cost function(1), and from watermark embedding key k_2, we can select those effective weights for the watermark extraction and reorganize them into vector $\{W_i^*, i=1,\ldots,n\}$ and $\{W_i', i=1,\ldots,n\}$ respectively by Hilbert scan. Next, the possible watermark can be extracted by operation $w_i^* = (W_i^* - W_i')/(\alpha W_i')$ and the similarity with the watermark WM is calculated using the correlation formula between WM^* and WM:

$$corr(WM^*, WM) = \frac{\sum_{i=1}^{n} w_i^* \cdot w_i)}{\sqrt{\sum_{i=1}^{n} (w_i)^2} \cdot \sqrt{\sum_{i=1}^{n} (w_i^*)^2}} \quad (7)$$

if $corr(WM^*, WM) > T$, it indicates that a watermark exists in the tested image. In order to avoid the false-alarming and false-positive errors, the threshold is decided by experimental results. From several trials, we find out that when we set T=0.56, our method has optimistic results than other parameters. So T is selected to be 0.56 in our work.

Figure 1 (a) Original image
(b) Watermarked image(38.03db)

5. EXPERIMENTAL RESULTS

To evaluate the performance of the proposed watermarking scheme, we have tested the proposed algorithm on images with various content complexities, e.g., “Lena”, “Pepper”, and “Baboon”. The former image is considered a representative of less complicated images, while the latter is a representative of relatively more complicated images.

When embedding the watermark in the reconstruction weights, the parameter K for finding K-nearest neighbors should be determined first. According to Sam’s work[7], for a small K, there will be small computational work, poor reconstruction of the image and a small watermark embedding capacity, instead, higher computational work for a large one with possibly good reconstruction ones. We compare the result of some possible parameters for K, and the results show that for K=32 can give better result with a compromise between the computation and reconstruction quality. And in the following experiment, we generate a bipolar sequence with length n=1000 by the key k_1 as the watermark.

The experimental results with the “Lena” image of 512*512*8 bits are shown in Fig.1, and the PSNRs of the watermarked images are higher than 36 dB. The watermarks are perceptually invisible. Table I lists the various test functions in StirMark 3.1[11] and the tested results with our proposed algorithms applied to the “Baboon” image. (Note that similar results have been obtained when the StirMark functions are applied to other images such as the “Lena” and “Pepper” images).

For each image, we assign a score 1 if for that case, the watermark is correctly decoded. If the watermark is incorrectly decoded, we assign a value of zero.

The results are summarized in Table 1, where we compute the average for each section. In the signal enhancement section it includes Gaussian, median, and

Table 1 Summary of Experimental results

	Proposed method	Digimarc	Suresign
Enhancement	0.95	1	1
Compression	0.83	0.81	0.95
Scaling	1	0.72	0.95
Cropping	0.74	1	1
Rotation	1	0.94	0.5
Row/Column removal	1	1	1
Random geometrical distortions	0.5	0.33	0

sharpening filters as well as the frequency mode Laplacian Removal attack, the watermark was correctly decoded in all cases. The algorithm is successful against JPEG down to a level of 75 quality factor. In all cases, the combinations of rotations, scales and cropping were correctly recovered. The watermark is not correctly recovered when the image has been cropped a lot. Furthermore, the watermark is also successfully recovered against combinations of row and column removal. The algorithm fails for random geometric distortions since the feature weights are severely distorted. However, to our knowledge, at this time no algorithm systematically decodes watermarks successfully after being attacked by the random geometric distortions implemented in the Stirmark3 package.

It is seen that our algorithm can successfully resist random removal of some rows and some columns, which is referred to as jitter attack. According to [11], most simple spread spectrum based marking techniques have been defeated by the jitter attack. It is also shown in Table I that the proposed algorithm successfully resists aspect ratio variations, scaling, small size cropping, small angle rotation (followed by cropping), and the combination of scaling with small angle rotation. This demonstrates that our watermarking method is able to resist affine transforms effectively.

6. CONCLUSIONS

In this paper, the LLE-based watermark embedding application has been studied. Since the LLE have some good properties that are much important for watermark embedding, our proposed watermarking scheme based on LLE is robust to general affine transformations. From the experimental result, it is seen that the watermark can be detected completely under the severe geometric attacks such as rotation, scaling and row/column removal and so on. We also found our scheme performance is better than that based on FFT which is reported to resist such an attack. We conclude that this transform can be used for watermarking purpose and deserve a more thorough examination.

ACKNOWLEDGEMENT

This work is partly supported by National Natural Science Foundation of China No. 60372078

REFERENCES

[1] *Vleeschouwer, D.C.,Delaigle, J.F., Macq, B., "*Invisibility and application functionalities in perceptual wateramrking-an overview", Proceeding of IEEE, Vol. 90(1), pp.64 –77, 2002.

[2] M.Wu, B. Liu, "Data Hiding in Image and Video:Part I—Fundamental Issues and Solutions, and Part II—Designs and Applications", IEEE Transaction s on image processing, Vol.12(6), pp.685-705, 2003.

[3] J. J. K. Ó Ruanaidh and T. Pun, "Rotation, scale and translation invariant spread spctrum digital image watermarking," *Signal Process.*, vol. 66, pp. 303–317, May 1998.

[4] A. Herrigel, J. J. K. Ó Ruanaidh, H. Petersen, S. Pereira, and T. Pun, "Secure copyright protection techniques for digital images," in *Int. Workshop Information Hiding*, Portland, OR, Apr. 1998.

[5] S. Pereira, J. J. K. Ó Ruanaidh, F. Deguillaume, G. Csurka, and T. Pun, "Template based recovery of Fourier-based watermaks using log-polar and log-log maps," *Int. Conf. Multimedia Computing Systems, Special Session Multimedia Data Security Watermarking*, June 1999.

[6] P. Shelby and P.Thierry, "robust template matching for affine resistant image watermrks", IEEE transactions on image processing, Vol.9(6), pp.1123-1129, 2000.

[7] T.R. Sam and K.S. Lawrence., "Nonlinear dimensionality reduction by locally linear embedding", Science, Vol.290(22), pp.2323-2326,2000.

[8] F.A.P.Petitcolas, Stirmark3.0, Available at http://www.cl.cam.ac.uk/fapp2/watermarking/stirmark, 1999.

[9] K.S. Lawrence and T.R. Sam, "Think Globally, Fit Locally: Unsupervised Learning of Nonlinear Manifolds", Technical Report MS CIS-02-18, University of Pennsylvania, 2002

[10] K.S. Lawrence and T.R. Sam, "An Introduction to Locally Linear Embedding", Technical Report.

[11] F.A.P.Petitcolas and R.J. Anderson, "Attacks on copyright marking systems," *Proc. 2nd Int. Information Hiding Workshop*, pp. 219–239, 1998.