Affine Invariant Watermark Scheme using Genetic Algorithm

Jin Cong[1]
Jiaxiong Peng[2]

Department of Computer Science, Central China Normal University, Wuhan 430079, P.R.China
The State Key Laboratory of Education Ministry for Image Processing and Intelligent Control, Huazhong University of Science and Technology, Wuhan 430074, P.R.China
[1] jincong@mail.ccnu.edu.cn
[2] jiaxpeng@sohu.com

Abstract. Geometrical attacks are among the most challenging problems in present day digital watermark. Such attacks are very simple to implement yet they can defeat most of the existing digital watermark algorithms without causing serious perceptual image distortion. Geometric attacks can very easily confuse the decoder unless it transforms the image back to its original size/orientation, *i.e.*, recover the lost synchronism. To be able do so, the decoder needs to know how the image has been manipulated, *i.e.*, needs to know geometric transformation parameters. In this research, we report a novel method to estimate the geometric manipulation. We compute the point pattern matching measure for the geometric manipulation. The shape-specific points of the original image are computed. The point matching is realized by genetic algorithm. The proposed scheme does not require the original image because SSP information of the original image has been memorized by the neural network. This method has been proved its robustness to geometric attacks through experiments.

Keywords: geometrical transformation, shape-specific points, points matching, genetic algorithm, neural network.

Received June 23, 2005 / Accepted September 5, 2005

1. Introduction

Digital watermark[1,2], the art of hiding information into multimedia data in a robust and invisible manner, has gained great interest over the past few years. There has been a lot of interest in the digital watermark research, mostly due to the fact that data hiding might be used as a tool to protect the copyright of multimedia data. A secret message, which is called digital watermark, is an imperceptible signal embedded directly into the media content, and it can be detected from the host media for some applications. The insertion and detection of digital watermarks can help to identify the source or ownership of the media, the legitimacy of its usage, the type of the content or other accessory information in various applications. Specific operations related to the status of the watermark can then be applied to cope with different situations.

Geometrical modifications, we refer to all image manipulations that change the image spatially. We consider geometrical modifications that most image users can apply easily while the value of the image is still preserved. In particular, we focus on the affine transform, which can be defined by a 2×2 affine matrix plus translation. Given the input position (x, y), the new position (x', y') can be determined by using six parameters in the following model:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a_{00} & a_{01} \\ a_{10} & a_{11} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} x_s \\ y_s \end{pmatrix}.$$

To successfully detect the watermark from the image undergoing geometrical modifications, we need to determine these six parameters so that the inverse transform can be applied to recover the image.

In this paper, our purpose is develop a robust watermark scheme that can achieve fast self-synchronizing watermark detection. To the self-synchronizing, we used the shape-specific points (SSP)[3], which is translation, rotation, and scale change independent. Instead of random signal, a semantic image is taken as the watermark to be hidden. Genetic algorithm is employed to match SSP for enhance the performance of self-synchronizing detection. Our scheme has been proved its robustness to geometric attacks through experiments.

2. The Shape-Specific Point (SSP)

The definition and concepts of the SSP described below are based on [4]. In the following discussion, *S* is a discrete set of points $\{(x_i, y_i), i = 1,...,n\}$ in a planar. If *t* is a transformation on *S*, then *t* induces a transformation on the individual points of *S*.

Definition: Given a point set *S* in a plane *P* and a set *T* of transformations in *P*, let $p = f(S)$ be a point in *P* computed from *S* according to procedure *f*. Also, let $S' = t(S)$ where $t \in T$ and let $p' = f(S')$. Then *p* is a shape-specific point of *S* with respect to *T* if and only if $p' = t(p)$.

The above definition simply says that if transformation *t* takes *S* onto S', then it takes *p* onto p'.

Examples of how to calculate SSPs are described below: The coordinate of the unweighted mean point (centroid) *A* of *S* can be calculated by

$$x_A = \frac{1}{n}\sum x_i, \quad y_A = \frac{1}{n}\sum y_i \tag{1}$$

It is very ease to know that (x_A, y_A) is a SSP of *S* with respect to *T*.

In this case, (x_A, y_A) is a SSP of *S* with respect to *T*. Another example of SSP is the radius weighted mean point *B* of *S*. The coordinate of *B* can be calculated by

$$x_B = \frac{1}{R}\sum r_i x_i \ , \quad y_B = \frac{1}{R}\sum r_i y_i \tag{2}$$

where r_i is the distance from the centroid *A*, as defined above, to the point (x_i, y_i) in *S* and $R = \sum r_i$.

In fact, for any given SSP a new SSP can be obtained. For example, in Formula (2), if r_i is replaced by the radius from a known SSP to the point (x_i, y_i) in the *S*, a new SSP can be calculated.

Different image, with different sizes and feature point, should have different texture and SSP. Employing this characteristic, we construct a new watermark algorithm using definition and properties of SSP to estimate the parameter of geometry transformation undergone by watermarked image. During the watermark detection process, these transformations are inverted based on parameters. Synchronization can then be realized, which is crucial for watermark detection.

We focus our study on Canny edge detector [5] that are commonly used in pattern recognition and vision systems. In the application of image watermark, the ideal feature points should be invariant to geometrical transformation. That is, once an image undergone geometric transformation, we expect the feature points can still indicate the same local features. Moreover, when only partial information of an image is available, using feature points is advantages over other features (such as edges or regions) to capture local information. For example, if image I_A is a portion of the image I_B, the interest points of I_B will also include that of I_A.

Therefore, it is divided into $n \times n$ blocks after getting the Canny edge image. We calculate SSPs of each block, and the feature point set of the image may be obtained. The feature point set of the original image is called reference point set, denoted by $P = \{p_j = (p_{j1}, p_{j2})^T \mid j = 1,2,...,t\}$.

Reference point set information of the original image is memorized by the neural network. Secret key and reference point set information are input and the output of the neural network, respectively. In other words, after the neural network trained, reference point

set information can be obtained when secret key is inputted.

3. Watermark Embedding and Detection

For security, the designed watermark is not directly mapped onto the image. Suppose *d* is a secret image hidden in an original image, where $d=\{d_i \mid d_i \in \{0,1\}, i=1,2,\ldots,A\times B\}$. It is converted into a binary bit string called watermark sequence, from a starting location (x,y), where *x* is random integral from 1 to *A* generated using secret key, and *y* is random integral from 1 to *B* generated using other secret key. The algorithm starts scanning the secret image from line *x* line by line downward to line *A*, back to line 1, line 1 till line *x*-1. For each line, the scan starts at column *y*, rightward to column *B*, back to column 1, column 1, till column *y*-1. So, the watermark sequence $W=\{w(j) \mid j=1,2,...,A\times B\}$ can be obtained.

In the paper, a frequency-domain watermark method based on 8×8 block-DCT transform is used by casting watermark bit into the middle frequency part of a block.

Let the DCT coefficients in an 8×8 block be zigzag scanned in order from 1 to 64. We only choose some middle frequency components for watermark to achieve a trade-off between transparency and robustness. Let $I_i(m)$ denote a DCT coefficient located at the *m*-th location (in a zigzag scanning order) of block *i* $(i=1,2,...,s)$. Suppose only $k=64\times\frac{A\times B}{N_1\times N_2}$ middle frequency components are selected for the watermark embedding, where $N_1\times N_2$ is the original image size.

The amount of modification each coefficient undergoes is proportional to the magnitude of the coefficients itself as expressed by the rule

$$\mathrm{I}_i'(L+j)=\mathrm{I}_i(L+j)+\alpha\cdot\left|\mathrm{I}_i(L+j)\right|\cdot w(i+j),\quad j=1,2,...,k \tag{3}$$

Where α indicates a parameter controlling the watermarking strength, and the first *L* coefficients are skipped for embedding the middle band. I_i' is then inserted back into the image in place of I_i. Then the inverse DCT of the watermarked coefficients gives the watermarked image $I'(x,y)$.

To detect a watermark in a possibly watermarked image, firstly, test image is decomposed into non-overlapping blocks of 8×8 and the DCT is computed for each blocks. With choosing DCT coefficients the same for embedding watermark, the DCT coefficients are selected for detected existing of the watermark. Secondly, after inverse DCT, *k* data of detecting watermark are confirmed for each block. *k* data of all blocks are combined into a sequence *R* with $A\times B$ values. Finally, the correlation is calculated between the sequence *R* and the watermark sequence *W*.

During the detection process, it is common to set a threshold ρ to decide whether the watermark is detected or not. If the correlation exceeds a certain threshold ρ, the watermark detector determines that test image contains watermark *W*. The correlation value is calculated by:

$$c=\frac{1}{s}\sum_{i=1}^{s}\{\frac{1}{k}\sum_{j=1}^{k}R_i(L+j)\cdot w(i+j)\},\quad \text{where } s\times k=A\times B. \tag{4}$$

4. Feature Matching by Genetic Algorithm

SSP set is first calculated to test image to obtain the information about the synchronization. To extract watermark without the original image, trained neural network is employed to obtain the reference point set information of the original image. By matching SSP set of test image with the reference point set, we can adjust the attacked test image to achieve the synchronization of watermark detection.

For tested image, its SSP set are obtained, denoted by *Q*, using the same method with original image. Genetic algorithm (GA)[6] is adopted for feature point matching. For measure the degree of match between two feature point sets *P* and *Q*, fitness function is constructed by the partial bidirectional Hausdorff distance [7]. The output of GA is the feature points set

which has the highest value of fitness function. Using this feature points set and the reference point set, the geometric transformation parameters can be solved.

4.1. Chromosome Code

The initial population is generated randomly. However, the generating scope of chromosomes is not arbitrary but limited within the image size. Each chromosome's detailed construction method is seriating coordinates of the feature points and coding them into binary codes. Here using binary coding is because the image resolution is always limited. The coordinate of points can be represented by integers in the scope of the image resolution. Even if the location precision of points is less than a pixel, enough representation precision can still be obtained by changing the length of binary code.

4.2. Design the Fitness Function

Clearly, if the match degree between P and Q can be measured, it is equivalent to evaluating the fitness of the chromosome. Considering the partial bidirectional Hausdorff distance between point sets P and Q, the smaller the distance is, the match degree between P and Q is larger. So, the fitness function can be selected as the inverse of the partial bidirectional Hausdorff distance

$$fitness = \frac{1}{a + H_{LK}(P,Q)}$$

where a is a positive constant. In order to avoid the denominator is a zero, the partial bidirectional Hausdorff distance is added a.

4.3 Genetic Operators

A pair of chromosome is randomly chosen from the population and is used as the parents of reproduce the offspring. The selection principle is the more chromosome number of its new offspring to the next generation, the bigger fitting function value F_i with the larger probability p_s. The nature choice phenomenon of the biosphere is simulated by $p_s = \frac{F_i}{\sum_{j=1}^{l} F_j}$

Crossing operator is obtained after reset the parents facts which are got the selection processing. Crossing processing is performed according to a certain probability, which is called crossing probability p_c. The crossing effect is to produce better chromosome after combination the generating materials of the parents. Here, single point crossover operator is adopted.

Mutation operates on each binary bit of a chromosome in another predefined probability p_m, the mutation operator is realized by reverse the value of the current binary bit, i.e. 0-1, 1-0.

4.4 Genetic Algorithm used Feature Match

For two features point sets P and Q, determine population size N, crossing probability p_c, mutation probability p_m, fractions f_L and f_K of the partial bidirectional Hausdorff distance and the maximum iterative steps G_{max}.

Step 1: Randomly generate N point sets in the test image and then convert them into chromosomes for initial generation.

Step 2: Evaluate the fitness of each chromosomes in current population and then create a new population by repeating following steps until the new population is complete

1) Select two parent chromosomes from a population according to their fitness (the better fitness, the bigger chance to be selected).

2) With a crossover probability cross over the parents to form new offspring (children). If no crossover was performed, offspring is the exact copy of parents.

3) With a mutation probability mutate new offspring at each locus (position in chromosome).

4) Place new offspring in the new population.

Step 3: Use new generated population for a further run of the algorithm.

Step 4: If the end condition is satisfied, **stop**, and return the best solution T_{best} in current population, where T_{best}

be the affine transformation determined by the best chromosome, and match point stes $T_{\text{best}}(P)$ and Q according to the simple nearest neighbor rule. If the maximum iterative step G_{max} is not reached, goto *Step* 2.

5. Full Scheme for Estimating Geometry Transformation Parameters

The key of the watermark scheme for resisting geometric attacks is to estimate the parameter of geometry transformation. For better comprehending this new scheme, the framework of this watermark scheme is given. The scheme can be represented in the framework shown in the block diagram of Figure 1.

6. Experiment Results

Proposed scheme has been tested under various attacks. The standard gray level images "Baboon"(256 × 256 pixels, see Figure 2) is applied and Figure 3 as a secret message. Proposed algorithm and attacks are performed under Matlab 6.5 environment and the "dct2" is used to compute the DCT. Figure 4 is the watermarked image after embedding the secret message into Figure 2 according to equation (3), when ω =0.03.

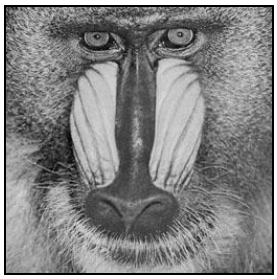

Figure 2 Original image

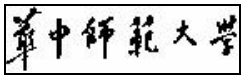

Figure 3 Secret message

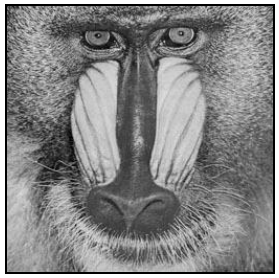

Figure 4 Watermarked image

Figure 5 The Canny edge of Figure 2

In the experiments, the algorithm parameters are set as, population size N=300, crossover probability p_c =0.05, mutation probability p_m =0.02, two fractions f_L and f_K are all 0.7, and the maximum iterative steps $G_{\max}$ =120. The length of binary code for each coordinate is 16 and L=11. Learning rate for the neural network is $\eta = 0.1$, and train epochs are 50 times. Original image is divided into 64×64 blocks. SSPs of each block are calculated. The length of binary code for each coordinate is 16.

It has been verified that the proposed scheme is robust against various attacks using common signal processing and geometric transformations. All the simulations are carried out on the Figure 2.

In the first simulation, the scheme's robustness against geometric manipulations, e.g., cropping, scaling, or rotation, is tested. Robustness against geometric manipulations is very import because these attacks are very common. The main reason to gain higher robustness against geometric transformation attacks is the SSP applied. Simulations have confirmed that the proposed scheme achieve this goal, which are further discussed as follows.

Figure 6 Referenced Points Set of Figure 5

Figure 7 Tested image by cropping quarter of Figure 4

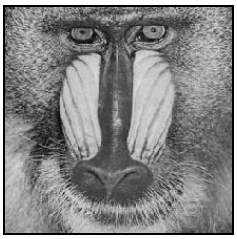

Figure 8 Tested image by scaling Fig.4 according to 85%

Image cropping will remove some watermark information. In this case, being able to correctly reveal the presence of watermark, based on the residual information, is an important property of a watermark scheme. Simulation results are indeed quite impressive. Watermark can still be correctly detected even there were quarter of the watermarked image cropped, where 0 is inserted to the cropped pixels. Figure 7 shows a typical result. In fact, the watermark can always be correctly detected when the cropped part is less than 1/3 of the watermarked image.

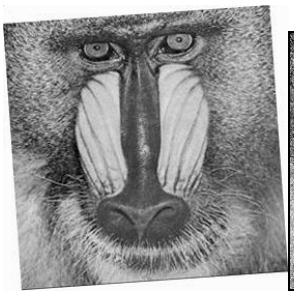

Figure 9 Test image by rotating Figure 4 according to 4°

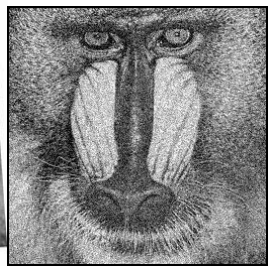

Figure 10 Test image attacked by 0 mean Gaussian noise with variance 0.01

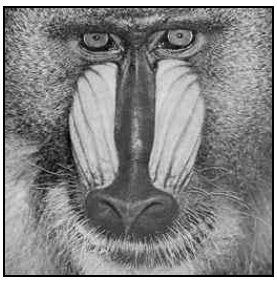

Figure 11 Test image compressed according to JPEG quality 65

In the cases of scaling and rotation, bilinear and higher order inserting value operation etc., interpolation is very common. In these situations, the proposed scheme can still resist the scaling and rotation. It is believed that there are two reasons for success. One is the calculation of SSP is a kind of sum operation and the sum operation has certain smooth functions, and the other is that since the watermarked image is similar to the original image as the watermark is embed imperceptible and SSP point sets are similar, too.

The scheme's robustness against additive noise attacks is tested. One of the two types of noise are added: Gaussian noise (see Figure 10), and salt and pepper noise, to the watermarked image. Results show that the watermark can still be correctly detected under these types of noise attacks.

The robustness against JPEG compression has also been tested (see Figure 11).

The above simulations and analyses have all confirmed that the proposed scheme has high robustness against common geometric transformations, additive noise, and compression.

7. Conclusions

In this paper, we have proposed a novel scheme of image watermarking. This scheme applies the SSP technique and features point matching method by genetic algorithm for resisting geometric attacks. Simulations have confirmed that this new scheme has high fidelity and is highly robust against geometric attacks and signal processing operations such as additive noise and JPEG compression. Therefore, it is a very promising technique for high-quality and reliable still image watermark applications.

8. References

[1] M.Barni, F.Bartolini, and T.Furon. *A general framework for robust watermark security. Signal Processing*, v.83, p.2069-2084, 2003

[2] F.Y.Shih, and S.Y.T.Wu. *Combinational image watermark in the spatial and frequency domains. Pattern Recognition*, v.36, p.969-975, 2003

[3] L.H.Zhang, W.L.Xu, C.Chang. *Genetic algorithm for affine point pattern matching. Pattern Recognition Letters*. v.24, p.9-19, 2003

[4] A.Mitiche, J.K.Aggarwal. *Contour registration by shape-specific points for shape matching. Computer Vision, Graphics, and Image Processing*.

v.22, p.396-408, 1983

[5] T.J.Canny. *A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence*, v.8, p.679-698, 1986

[6] D.E.Goldberg. *Genetic Algorithm in Search, Optimization and Machine Learning. Addison-Wesley*, New York, 1989

[7] D.P.Huttenlocher, G.A.Klanderman, and W.J.Rucklidge. *Comparing images using the Hausdorff distance. Transactions on Pattern Analysis and Machine Intelligence*, v.15, p.850–863, 1999

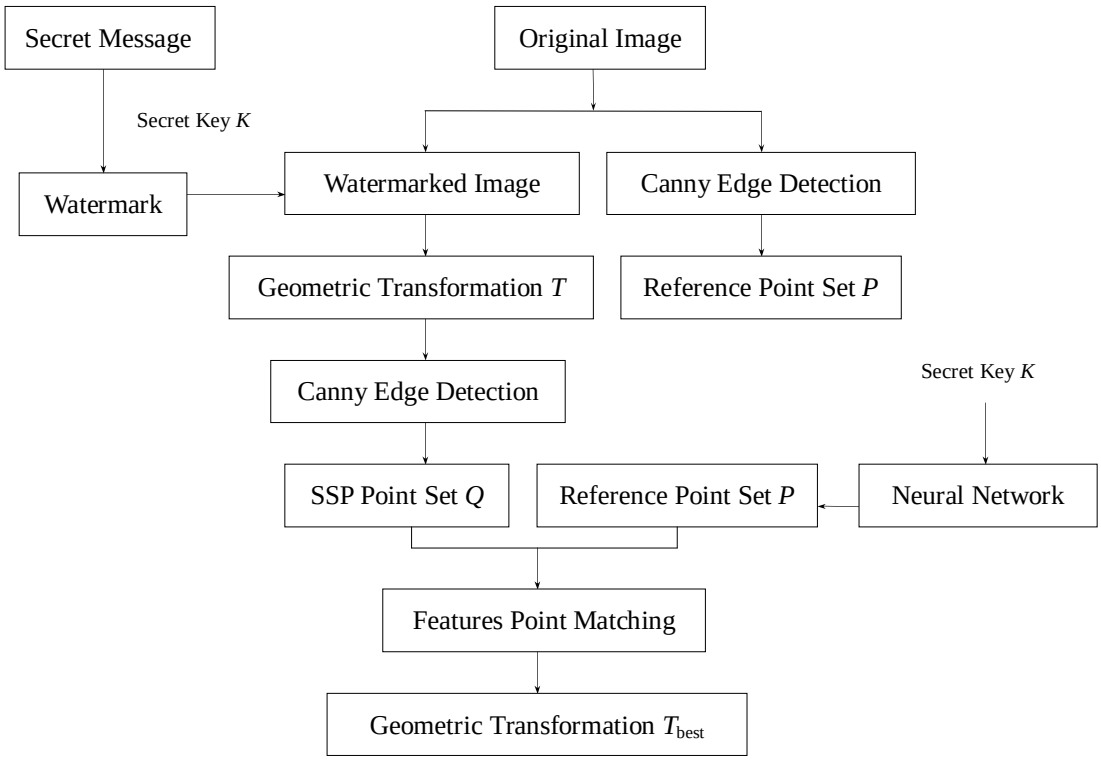

Figure 1. The block diagram for estimating the parameter of geometry transformation