Improvement of watermark detection ability based on blockwise image compensation for geometrical distortions

Hisanaga, Ryuji

Kuribayashi, Minoru

Tanaka, Hatsukazu

(Citation)
Electronics and Communications in Japan (Part III: Fundamental Electronic Science),89(12):1-9

(Issue Date)
2006-06-26

(Resource Type)
journal article

(Version)
Accepted Manuscript

(URL)
https://hdl.handle.net/20.500.14094/90000270

Improvement Scheme of Watermark Detection Ability Based on Blockwise Image Compensation for Geometrical Distortions

Ryuji Hisanaga, Minoru Kuribayashi, Hatsukazu Tanaka

Abstract

One of the important problems of watermarking techniques is the weakness against geometrical distortions because of synchronization loss. In this paper, we propose an image compensation scheme against geometrical distortions in order to extract watermark correctly. Our scheme collects the feature points in an image so as to recover the distortions caused by geometrical distortion attacks. Such feature points are selected by the local maximal value of partitioned regions which can be obtained by several filtering operations. Based on the computer simulated results, we have searched the optimal filtering operation and interpolation, and then studied the validity of our scheme.
keywords: watermarking, geometrical distortions, image compensation, feature points, affine transformations

1 Introduction

Digital watermark technique enables an author to embed information into digital contents like picture for the prevention of its illegal use. For the copyright protection, a robustness against attacks is strongly required. Here, the attacks for watermarking are refereed to two operations; one is to remove a watermark from a watermarked contents, and the other is to make the detection impossible. The attacks are also classified into geometrical attacks such as rotation, scaling, etc. and non-geometrical attacks such as JPEG compression and filtering operations [1]. In general, it is easy to achieve the robustness against non-geometrical attacks compared with geometrical ones. Because the latter one changes the embedding positions, the detection of the synchronization becomes difficult [2]. In the countermeasure of the geometrical attacks, two kinds of schemes have been proposed; one correct the synchronization loss before extraction [3], and the other embeds a watermark into the invariant regions against the geometrical modifications [4][5][6][7]. However, if the combination of several geometrical attacks performed randomly, the latter scheme is not robust, and the former one requires much computation to recover the original embedding positions. Then, Bas et. al [8] utilized feature points in an original image. Here, the feature points are mainly edges like contour in an image.

In [8], although several feature points are applied in order to achieve a tolerance for the geometrical attacks, they are mostly correlation-based watermarking scheme (Informed watermarking), not a blind watermarking which can extract information bit sequence. One of the methods considering the robustness against the geometrical attacks on the blind watermarking is the image compensation scheme in [9]. In this scheme, an edge detection is performed for an original image to obtain its feature points, and they are stored. When you attempt to extract a watermark from a geometrically distorted image, before the extraction operation its feature points are calculated. Then, using the feature points, the parameters of affine transformation are estimated, and recovery operations are performed using the parameters. In this scheme, the amount of distortions caused by the geometrical distortions may be represented by 6 parameters of the affine transformations. However, this scheme dose not consider the multiple and random geometrical distortions. It seems that the scheme may be tolerate for such distortions if it embeds logo image, but it may be difficult to apply the scheme for a blind watermark which can extract information bit sequence. Furthermore, the scheme calculated the 6 affine parameters only for the whole image, it may be difficult to perform the recovery operations correctly. Especially, it will be necessary to perform the recovery operation adaptively considering the local conditions for the robustness against StirMark attack [1] that distorts an image with multiple geometrical attacks.

In this paper, we design the recovery operations for each local region and estimate the parameters of affine transformation in order to recover the original watermarked image correctly from geometrically distorted image. In our scheme, an image is first partitioned into each region after edge detection. In order to remove the effects from noise, median filtering is performed. Then, each feature point that indicates strong edge region is detected from each partitioned region. Using each three feature points of original and distorted image, the parameters of affine transformation is estimated, and the recovery operation is performed. The exploitation of the feature points in each regions enables the locally adaptive compensation. We determine the optimal edge detection filter and interpolation method by computer simulation.

2 Affine Transformation

Affine transformation is to deform an image by the combination of geometrical linear transformation and shift operation, and rotation, scaling, shift operation is included in this transformation.

Let (x, y) be an original coordinate of a pixel and (x', y') be the geometrically distorted coordinate. Then, the affine transformation is represented by the following equation.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix} \tag{1}$$

Every geometrical distortions such as rotation, scaling, etc. are completely represented by the parameters a, b, c, d, e, f in Eq.(1). Once these 6 parameters are determined, by modifying Eq.(1) (x, y) can be recovered from (x', y') [9].

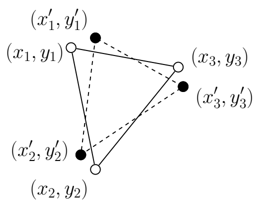

Figure 1: A model of geometrical distortions.

Three coordinates of original pixels and distorted ones is sufficient to obtain these 6 parameters from Eq.(1). Let $(x_t, y_t), (x'_t, y'_t)$ $(t = 1, 2, 3)$ be these coordinates (See Fig.1). Then, the following relation can be satisfied.

$$A\boldsymbol{u} = \boldsymbol{b} \tag{2}$$

where

$$\begin{aligned} A &= \begin{pmatrix} x_1 & y_1 & 0 & 0 & 1 & 0 \\ x_2 & y_2 & 0 & 0 & 1 & 0 \\ x_3 & y_3 & 0 & 0 & 1 & 0 \\ 0 & 0 & x_1 & y_1 & 0 & 1 \\ 0 & 0 & x_2 & y_2 & 0 & 1 \\ 0 & 0 & x_3 & y_3 & 0 & 1 \end{pmatrix} \\ \boldsymbol{u} &= (a, b, c, d, e, f)^T \\ \boldsymbol{b} &= (x'_1, x'_2, x'_3, y'_1, y'_2, y'_3)^T \end{aligned}$$

The 6 parameters can be obtained by multiplying A^{-1} from left side of Eq.(2). These parameters are assumed as the information to represent the triangle composed by the above three coordinates. Now, we consider the case that correct parameters cannot obtained from three coordinates. For example, if the coordinates are arranged in row, x_t becomes a constant value. Then, the row vectors in A are linearly independent, and hence no solution in the equation exists. Therefore, if feature points are approximately placed in row, the estimation of the affine transformation is impossible. In order to avoid such a case, the feature points are selected from neighboring partitioned regions in our scheme.

3 Proposed Compensation Method

In the proposed scheme, the amount of distortions in the image attacked by geometrical attack is estimated and corrected in order to extract a watermark correctly. First, each feature point is calculated from each partitioned

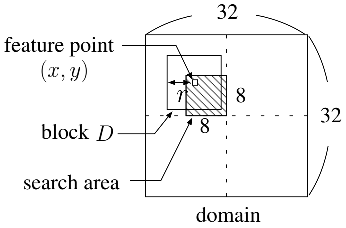

Figure 2: Search area for the feature point.

region of the image after edge detection using a filtering operation. Then, the geometrical distortions are corrected using the feature points of surrounding regions. Since the parameters of affine transformation is calculated referring the original feature points, the information is inevitable in our scheme, which is stored as a secret information.

3.1 Feature Points

In order to obtain feature points, first filtering operation is performed to an image. In the operation, edge detection filter is applied after median filtering [10]. Because the median filter will remove noise in an image, the edge detection filter will not enhance the noise elements caused by non-geometrical attacks. There are many edge detection filters such as differential processing, gradient, etc., optimal filter is determined by computer simulation.

Then, the filtered image is partitioned into small regions. In each region, the mean value of the gradient in a block D is searched in a search area described in Fig.2, where D is a square block located r pixels adjacent to a target pixel. In general, if a block contains edges, the gradient value in the block is large, hence this method will work well. Here, if the feature point is searched from the whole region, there may be a case not to correct the geometrical distortions. For example, if the distance between adjacent feature points is small, as we mentioned in Section 2, a rounding error may prevent from obtaining a correct parameters of affine transformation. Furthermore, our compensation method utilizes the adjacent regions (See Section 3.2), the constructed triangle should be overlapped with the target region. Therefore, the size of the search area is 1/4 of the region and is placed at the upper left area from the center indicated in Fig.2. Then, the required information for our compensation is the coordinate of the feature point and the gradient values in D that center is the feature point.

The procedure to search feature points from a geometrically distorted image is summarized as follows. First, the image is partitioned into regions. Then, using stored information about the original image the corresponding feature points are searched. On the geometrical distortions caused by StirMark attack which is known as a benchmark tool of watermarking scheme, the changes of position is approximately within 6 pixels that is studied in [11]. Hence, the feature point in each region is remained within the coordinates $(x \pm i, y \pm j)$, $(0 \leq i, j \leq 6)$, which area is the search area in our scheme. The search ares is depicted in Fig.3. On the search area, a block D' is produced from the adjacent 6 pixels from a target pixel with coordinate (x', y'), and its normalized cross correlation $\mathrm{NCC}(X, X')$ with D is calculated. The feature point of the distorted image is determined if the correlation value is maximum. Here, the normalized cross correlation value is calculated by the following equation [9].

$$\mathrm{NCC}(X, X') = \frac{\sum_{i,j \in D, D'} X \cdot X'}{\sqrt{\sum_{i,j \in D} X^2}\sqrt{\sum_{i,j \in D'} X'^2}} \quad (3)$$

where,

$$\begin{aligned} X &= w(x+i, y+j) - \overline{w}(x, y), \\ X' &= w'(x'+i, y'+j) - \overline{w'}(x', y'), \\ &\quad (i, j \in D, D'), \end{aligned}$$

$w(x, y)$ means the gradient of the coordinate (x, y) and $\overline{w}(x, y)$ means the mean of the gradient in a block D. similarly, $w'(x', y')$ and $\overline{w'}(x', y')$ are corresponding values for the distorted image.

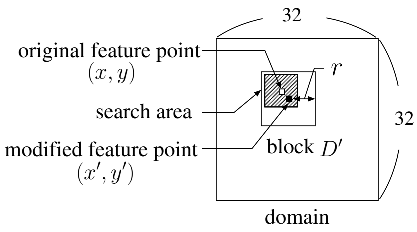

Figure 3: Search area for the feature point in a distorted image.

3.2 Compensation

In this subsection, the correction procedure using the obtained feature points that is calculated by the procedure shown in the previous subsection. We assume that the correction procedure is performed for each region. We call the target region A, and the adjacent regions are used for the correction. However, the adjacent regions are determined from the following 4 candidate corresponding to the target region's position.

1. Except for right endpoint and bottom endpoint (normal)
B: right side region, C: bottom side region

2. Right endpoint (right)
B: left side region, C: bottom side region

3. Bottom endpoint (bottom)
B: right side region, C: head side region

4. right and bottom endpoint (corner)
B: left side region, C: head side region

Using the feature points of before and after geometrical attack at the regions A, B, and C, 6 parameters of affine transformation is estimated by performing the operations discussed in Section 2. For a region A, the inverse affine transformation is performed using these parameters to correct the distortions. Finally whole image is corrected by this operations.

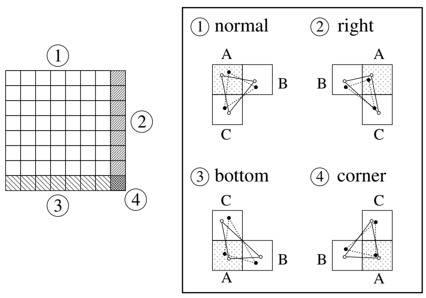

Figure 4: Classification of the domain for compensation processing.

3.3 Interpolation

When geometrical distortions are corrected using affine transformation, a rounding error may be occurred at the grid points. Hence, it is necessary to interpolate the pixel values for each grid points after affine transformation. There are several methods for the interpolation such as nearest neighbor method, linear, quadratic and cubic interpolation methods, etc.[13]. Nearest neighbor method is to place a pixel value that is the first nearest neighbor of a target pixel, and linear interpolation method is to calculate the approximation from adjacent 4 pixels. Quadratic and cubic interpolation methods apply more neighboring pixels for the interpolation. If linear interpolation method is performed, an image becomes slightly blurred on the whole. Quadratic and cubic methods improves the drawbacks to output more clear image compared with the linear method. The most blurred image is output by the linear method, and the effect is improved in the order corresponding to quadratic, cubic, and nearest neighbor methods. We test the optimal method from these ones by computer simulation.

4 Computer Simulated Results

4.1 Watermarking Method

On a blind watermarking, its bit error rate is relatively high, and hence it requires higher robustness against attacks. For the non-geometrical distortions, it is possible to be robust by embedding a watermark into low and middle frequency components of an image, which also improve the tolerance for the changes caused by the interpolation of our compensation scheme. For the geometrical distortions, although our scheme may be able to improve the robustness, it is more desirable to embed a robust watermark against the distortions. Referring to [12], if a watermark signal is embedded into extremely low frequency components of DCT coefficients, it can retain a tolerance for the geometrical distortions. However, if only such frequency components are selected for the embedding, an attacker may easily find the embedding positions. Hence, in our simulation, a watermark is assumed to be embedded into low and middle frequency components in order to make it difficult for an attacker to find the components.

First, an image is transformed by DCT, and 128 low frequency components are randomly selected for each bit of watermark information. Then, the selected DCT coefficients are transformed by 1 dimensional DCT, embed the information bit in one transformed coefficient, and inverse DCT are performed to obtain the watermarked image. When the inverse DCT are performed, the watermark signal is spread all over the image. The same operations to obtain the 1 dimensional DCT coefficients is performed for the watermarked image in the extraction. Although the robustness against geometrical attacks decreased compared with the method in [12], the watermark can be extracted correctly with high probability after our compensation.

4.2 Environment

Applying our compensation method as the preprocessing of watermark extraction, the correct extraction rate is tested. In this simulation, a blind watermarking method is used. Under the conditions listed in the Table 1, the simulation is performed by the following procedure.

Step 1. Embed a watermark into an original image

Step 2. Distort the watermarked image by attacks

Step 3. Perform our compensation method to the attacked image.

Step 4. Extract the watermark from the corrected image.

The above procedure is performed 10^4 times, and the rate [%] of properly extracted watermark without error is shown. Here, there will be two candidate for the stored information about feature points; that of the original image and the watermarked image. By defining them scheme I and scheme Ⅱ, the performance is compared in the following simulations.

Table 1: Parameters used for the simulation.

test image	256×256 pixel, 8-bit gray-scale“ lenna ”
watermark	32 bits
PSNR	39[dB]
benchmark	StirMark3.1*

∗ StirMark3.1:
http://www.petitcolas.net/fabien/
watermarking/stirmark31/

4.3 Performance of Edge Detection Filters

Since the positions of feature points depend on the applied edge detection filter, several filters are tested by comparing the performance of their correct extraction date in order to determine the optimal filter. In our simulation, 5 filters are tested; Sobel, Roberts and Prewitt filters as first derivation, Laplacian operator as second derivation, and Gaussian operator which uses Gaussian derivation. Here, random geometric distortions (RGD) are tested in the simulation. In general, geometrical distortions are caused by rotation, scaling and shift in the manner that the distortions are visually imperceptible. It is important to consider such attacks. The distortions caused by the StirMark attack is assumed for the scanning after printing out, and the attack performs several geometrical attacks such as stretching, shift, scaling and rotation randomly and locally [1][8].

Let the size of block D be $r = 8$. Using the nearest neighbor method, the correct extraction rate is shown in Table 2. From this result, it is clear first derivation is suitable for scheme I. This is because the first derivation is less sensitive against noise than Laplacian and Gaussian operators. In the first derivation, Roberts filter shows the best score both for scheme I and scheme Ⅱ. Since Roberts filter detects the derivation of oblique direction, the influence of geometrical distortions may be less than other filters which detect horizontal and vertical directions. Although Laplacian operator shows best score in scheme Ⅱ, we user Roberts filter in the following simulation considering the score in scheme I.

Table 2: Performance comparison for each filtering operation.

filter	scheme I [%]	scheme Ⅱ[%]
Sobel	49.9	88.3
Roberts	67.1	99.6
Prewitt	46.3	85.8
Laplacian	9.8	99.3
Gaussian	26.6	54.2

4.4 Performance of Interpolations

In this subsection, an optimal interpolation method for our correction operation is determined. From section 3.3, there are four interpolation methods; nearest neighbor method, linear, quadratic and cubic interpolation methods. In order to determine the optimal one, the performance of the watermark extraction against RGD is compared, where the size of block D is $r = 9$. From Table 3, the nearest neighbor method shows the best score for both scheme I and scheme Ⅱ, and the performance is decreased according to the effect of blury done by the applied filter. It may be come fromt the reason such that since a kind of interpoloation is executed when StirMark attack is performed, the rounding error further accumulated by the interpolation in our correction operation makes the extraction rate low. On the other hand, since the nearest neighbor method does not changes pixel values, the difference between the original and corrected pixel values may be small. As the consequence, in the following simulation we apply the nearest neighbor method.

Table 3: Performance comparison for each interpolation.

interpolation	scheme I [%]	scheme Ⅱ[%]
nearest	67.1	99.6
linear	52.1	99.0
quadratic	54.4	99.2
cubic	64.7	99.6

4.5 Required Storage

In this subsection, the relation between the required storage size and the extraction rate is tested. In section 3, we explained that the stored information contains the feature point and its square block D centering of the point. Hence, the extraction rate may depend on the partitioned regions and the size of D, namely r. The scores obtained for each r is shown in Fig. 5 and Fig. 6, where the regions are composed of 64×64 pixels (16 regions), 32×32 pixels (64 regions) and 16×16 pixels (196 regions). Here, we assume that

$$SIZE = \frac{\text{The size of } D}{\text{The size of image}} \times 100. \quad (4)$$

Then, from the figures, scheme I shows the better performance for 16 regions partitioning with low storage. On the other hand, scheme IIshows the better performance not for 16 regions patitioning, but for 64 regions partitioning, though the size of required storage is increased. As the consequence, 64 regions partitoning are applied in the following simulation.

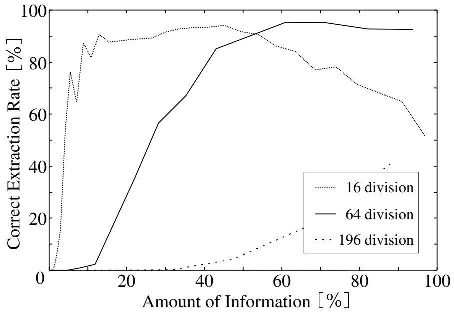

Figure 5: Detection rate versus division(scheme I).

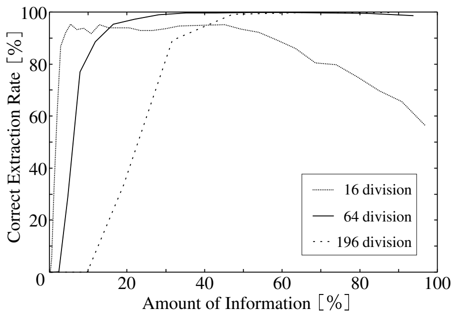

Figure 6: Detection rate versus division(scheme II).

4.6 Non-Geometrical Distortions

Our correction operation may fail to work for non-geometrical distortions. In this subsection, we evaluate the extraction rate against JPEG compression which is the most popular attack in non-geometical attacks. The extraction rate versus the quality parameter [%] of JPEG compression is shown in Fig. 7, where the size of D is $r = 9$. The overall extraction rate in scheme I seems to be low from the figure. This is because wrong feature points may be detected for the effects of embedding a watermark in the original image. On the other hand, if the quality is more than 30%, scheme IIcan extract a watermark with very high probability.

4.7 Geometrical Distortions

In order to evaluate the robustness against each geometrical distortion such as rotation, scaling, shift, etc., the extraction rates obtained with the parameters determined at the above experiments are shown in Table 4 – Table 6. In the simulation, the same correction operation as the above experiments is used for each geometrical distortion, and the size of D is $r = 9$. The distortions in this simulation affects whole image, though RGD contains local modifications. Table 4 shows the results for the multiple attack of rotation and clipping, Table

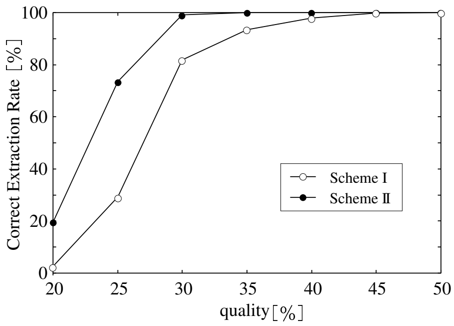

Figure 7: Robustness against JPEG compression.

5 shows that of rotation, clipping and scaling, and Table 6 shows that of linear distortions. Here, the clipping attack is applied for obtaining the original image size, 256×256 pixels from the image distorted by scaling and rotation. The linear distortion means an affine transformation by setting the parameters (a, b, c, d) in Eq.(1), and the matrices tested in Table 6 are

$$A = (1.007, 0.010, 0.010, 1.012),$$

$$B = (1.010, 0.009, 0.013, 1.011),$$

$$C = (1.013, 0.011, 0.008, 1.008).$$

The table indicates that though the applied watermarking scheme retains a robustness against the rotation of 0.25° and 0.50° degrees, it is vulnerable against that of more than 0.75° degree and the linear distortion. The extraction rate of the proposed scheme is decreased for the rotation of degree more than 2.00°. This is because the loss of information by the clipping attack becomes too large to extract a watermark. However, if an image is rotated more than 2.00° degree, such changes will be easily perceived. In this paper, we assume that the perceptual quality of the image distorted by attacks does not degraded, and hence the rotation more than 2.00° degree is out of this assumption. Except for such a distortion, our scheme shows excellent performance to correct the distortions caused by geometrical attacks. By comparing two proposed schemes, scheme I is inferior to scheme Ⅱ. The reason is that the original feature points may be slightly changed by embedding a watermark.

Table 4: Robustness against "rotation" and "cropping" .

θ(degree)	no correction[%]	scheme I [%]	scheme Ⅱ[%]
0.25	49.7	100.0	100.0
0.50	0.9	100.0	100.0
0.75	0.0	100.0	100.0
1.00	0.0	100.0	100.0
2.00	0.0	12.9	75.2
5.00	0.0	0.0	0.0

4.8 Dependency of Images

In order to evaluate the dependency of images, the extractions rates against RGD are tested for images "lenna", "girl", "couple", "boat", "barbala", "barboon", "cameraman", "moon","text", and "woman". From the results, for the images "girl", "moon", and "woman" a better extraction rate can be obtained with small size of block D. And for the images "lenna","cameraman", "barbala", "barboon", and "boat", the more the size of D becomes, the higher the extraction rate becomes. However, for the other images "couple" and "text", the extraction rate remains low even if the size of D is increased. The results for the images "lenna", "girl", and "couple" using scheme I and scheme Ⅱare shown in Fig. 8 and Fig. 9, respectively.

Table 5: Robustness against “rotation”,“cropping” and “scaling”.

θ(degree)	no correction[%]	scheme I [%]	scheme II[%]
0.25	100.0	100.0	100.0
0.50	2.9	100.0	100.0
0.75	0.0	100.0	100.0
1.00	0.0	100.0	100.0
2.00	0.0	7.7	39.6
5.00	0.0	0.0	0.0

Table 6: Robustness against “linear distortions”.

matrix	no correction[%]	scheme I [%]	scheme II[%]
A	0.0	99.8	100.0
B	0.0	99.1	100.0
C	0.0	90.4	100.0

4.9 Considerations

In our scheme, a simple block matching algorithm is applied to estimate the parameters of affine transformation for image correction. In the field of image processing, there are more superior correction methods and they may improve the performance to extract a watermark from distorted image. For example, in [14] an estimation function is applied to calculate several correlation values for the estimation of parameters of affine transformation. And in [15] a projective transformation and a bilinear transformation in which affine transformation is represented by their special case are applied. These schemes use two kinds of pixel values to calculate the parameters for the above transformations, and hence the information of whole image is utilized effectively. However, since the whole image or the filtered image must be required for the estimation, the required storage size is very large.

On the contrary, in order to reduce the storage size, the parameters of affine transformation is estimated from the coordinates of the feature points at neighboring blocks, which may be effectively realized by a block matching method. The effects are confirmed by the above simulation results.

5 Conclusion

In this paper, we proposed a compensation scheme to correct geometrical distortions using feature points in an image. By calculating the feature points from each partitioned regions after edge detection, a distorted image is corrected efficiently. In order to determine the optimal filtering and interpolation operation in our scheme, we evaluated the performance of the extraction rate using several candidates for the operation. Our computer simulated results showed that a watermark can be properly extracted from geometrically distorted image with high probability if our correction operation is performed before the watermark extraction.

The future work is to reduce the amount of stored information and to consider the dependency of characteristic of image.

References

[1] F.A.P.Petitcolas, R.J.Anderson, and M.G.Kunhn, “Attacs on copyright marking systems,” *Proc. of the Second Workshop on Information Hiding, IH’98,* LNCS 1525, Springer-Verlag, pp.218-238, 1998.

[2] F.A.P.Petitcolas, R.J.Anderson, “Evaluation of copyright marking systems,” *Poc. of IEEE Multimedia Systems’99,* vol.1, pp.574-579, 1999.

[3] F.Davoine, P.Bas, P.-A.Hebert, and J.-M.Chassery, “Watermarking et résistance aux déformations géométriques,” in *Coresta99,* Sophia Antipolis, France, 1999.

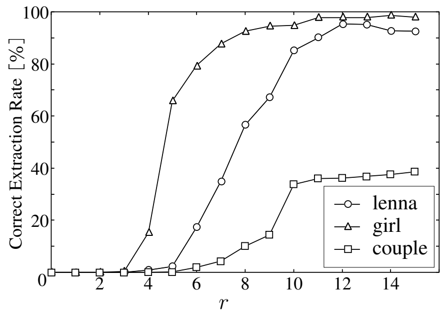

Figure 8: Comparison of detection rate (scheme I).

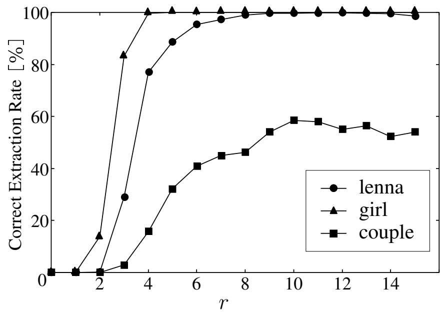

Figure 9: Comparison of detection rate (scheme II).

[4] D.Delanay and B.Macq, "Generalized 2-d cycric patterns for secret watermark generation," *Proc. of ICPIP,* vol.2, pp.77-80, 2000.

[5] T.Kalker, G.Depovere, J.Haitsma, and M.Maes, "A video watermarking system for broadcast monitoring," *Proc. of SPIE,* pp.103-112, 1999.

[6] J.J.K.Ó Ruanaidh and T.Pun, "Rotation,scale and transration invariant spread spectrum digital image watermarking," *Signal Process.,* vol.66, no.3, pp.303-317, 1998.

[7] S.Pereira and T.Pun, "Fast robust template matching for affine resistant image watermarking," *Proc. of the International Workshop on Information Hiding, IH'99, third,* LNCS 1768, Springer-Verlag, pp.200-210, 1999.

[8] P.Bas, J.M.Chassery, and B.Macq, "Geometrically invariant watermarking using feature points", *IEEE Trans. on Image Processing,* vol.11, no.9, pp.1014-1028, 2002.

[9] N.F.Johnson, Z.Duric, and S.Jajodia, Recovering watermarks from images, Information and software engeneering technical report, San Diego, CA, 1999.

[10] T. Minami and O. Nakamura, Image science and technology, Corona Publishing, 1989.

[11] M.Kuribayashi and H.Tanaka, "atermarking schemes using the addition property among DCT coefficients," *IEICE Trans. Fundamentals,* vol.J85-A, no.3, pp.322-333, 2002.

[12] M.Nakamori, M.Iwata, and A.Shiozaki, "Watermarking method for color images using DCT coefficients of color components," *Proc. of SCIS 2001,* pp.249-254, 2001.

[13] N. A.Dodgson, "Quadratic interpolation for image resampling," *IEEE Trans. on Image Processing*, vol.6, no.9. pp.1322-1326, 1997.

[14] C.S.Fuh and P.Maragos, "Motion displacement estimation using an affine model for image matching," *Optical Engineering*, vol.30, no.7, pp.881-887, 1991.

[15] V.Seferidis and M.Ghanbari, "General approach to block-matching motion estimation," *Optical Engineering*, vol.32, no.7, 1464-1474, 1993.