Geometrical Distortion-Resilient Watermarking Based on Image Features

Hiuk Jae Shim[1], Byeungwoo Jeon[2] and Rin-Chul Kim[3]
[1,2] School of Electrical and Computer Engineering
SungKyunKwan University
300 Chunchun dong, Jangan-gu, Suwon, Korea
Tel. +82-31-290-7144, Fax.: +82-31-290-7191
[3] Dept. of Electrical and Computer Engineering
University of Seoul, Korea
Tel. +82-2-2210-2661, Fax.: +82-2-2249-6802
e-mail : [1] waitnual@ece.skku.ac.kr , [2] bjeon@yurim.skku.ac.kr, [3] rin@uoscc.uos.ac.kr

Abstract: The major threat of geometric manipulations is that they change the positions of watermarks, therefore the detection process fails to extract watermark properly. Since they cause the same effects on the host image as watermarks simultaneously, evaluating the distorted host image can be helpful to measure the nature of distortions. In this paper, we propose a geometrical distortion-resilient watermarking algorithm based on this property. Firstly we evaluate the orientation of a host image by filtering it with directional Gabor kernels, then we insert embedding pattern aligned to the estimated orientation. In its detection step, we evaluate the orientation again by Gabor filtering, then simply project and average the projected value to obtain a 1-D projection average pattern. Finally, auto-correlation function of the 1-D projection average pattern identifies periodic peaks. Analysed are experimental results against geometrical attacks including aspect ratio changes.

1. Introduction

Geometrical distortions such as rotation, scaling, translation (RST), and aspect ratio changes, etc. are known to be very simple but quite an effective class of attacks, however, most of them can be described by the general affine transforms. Therefore many methods [1]-[4] to recover this class of attacks can be regarded as the methods to estimate the parameters of general affine transforms. Among them, Kutter has proposed the use of auto-correlation function (ACF) which utilizes embedding pattern as a reference point [2]. This approach gives quite good results, however, its exhaustive search limits its use in real-time applications.

One method to insert a pattern periodically over whole image is introduced by Voloshynovskiy et al. [3]. Since a pattern is embedded in a repetitive manner, Fourier magnitude spectrum of a periodical pattern yields discrete grid that can be used as reference points. Moreover, Hough transform is used recently in the approach so that estimation of undergone geometrical attacks becomes more accurate [4].

Since an image signal is represented in two dimensions, most approaches to recover geometrical distortions carry out two-dimensional search that consequently calls for complexities in general. There are nonlinear geometrical distortions; however, most of geometrical distortions are linear transform. When geometrical distortions are linear, 2-D image can be possibly separated into 1-D signals by projection, and it reduces complexity greatly.

Recently we proposed to use Gabor kernel and the projection average (PA) method to recover general geometrical distortions [5]. When geometric manipulations occur, the distortions affect the host image in the same way as the watermark; therefore evaluating host image can be helpful to measure the nature of distortions. In this regard, we utilize the orientation of image, which is accomplished by using the Gabor kernels, then watermark is generated and extracted by the PA scheme.

One of main drawbacks in the algorithm is incorrect estimation about translation. Since Gabor kernel is characterized by sinusoids weighted by Gaussian function, when the center position of image is shifted, Gabor estimated results much differ from the expected ones.

In our paper, we improve the orientation estimation using Gabor function so that translation does not pose any problem as described above. Moreover, to estimate distortions, we use the ACF instead of the discrete Fourier transform. Even if ACF itself has large complexity, our algorithm can work only dealing with 1-D signals which have much lower complexity than 2-D case.

2. Embedding and Extraction

2.1 Estimating orientation by Gabor filters

There are many features to consider, which can represent nature of image itself, i.e., lines, edges, colors, and etc. One of the most interesting features among them is the orientation of an image. Since filtering images with Gabor functions helps estimating directional blob or directional edge components [6, 7], Gabor kernels have been widely used in object recognition applications. In our method, we evaluate the directional feature of a host image with 2D Gabor kernels, which have their own energy spread into specific directions.

This work was supported by the ITRC program of the Ministry of Information & Communication of Korea.

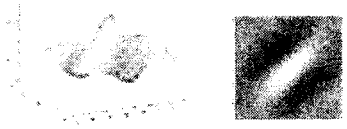
A 2D Gabor function is a product of an elliptical Gaussian function with a complex exponential representing harmonic modulation. A general form of the 2D Gabor function [6] is given by,

$$g(x,y) = e^{-[(x-x_0)^2/2\sigma_x^2+(y-y_0)^2/2\sigma_y^2]} \cdot e^{-j\frac{2\pi}{\lambda}[(x-x_0)\cos(\theta)+(y-y_0)\sin(\theta)]} \quad (1)$$

where, (x_0, y_0) is the center of the elliptical Gaussian function, σ_x^2 and σ_y^2 are the variance of the Gaussian function in the horizontal and vertical directions, θ and λ are the orientation and wave-length of the harmonic modulation function, respectively. In addition, assuming a unity aspect ratio ($\sigma_x^2/\sigma_y^2=1$), thef unction can be separated as a real Gabor function (RGF) and an imaginary Gabor function (IGF). The separated form of a Gabor function is as followings.

$$RGF(x,y) = \cos\left[\frac{2\pi}{\lambda}\{x\cos(\theta)+y\sin(\theta)\}\right]e^{\frac{-(x^2+y^2)}{2\sigma^2}}$$
$$IGF(x,y) = \sin\left[\frac{2\pi}{\lambda}\{x\cos(\theta)+y\sin(\theta)\}\right]e^{\frac{-(x^2+y^2)}{2\sigma^2}} \quad (2)$$

These two functions have different features from each other. RGF has one large positive central lobe and two smaller negative lobes on both side of it, while IGF has one large positivelob e and one negative lobe. The shape of RGF is shown in Figure 1 (a), whose differences are mainly made by cosine and sine functions in their forms. Therefore general usage of RGF is appropriate to detect object height and width.



(a) (b)

Figure 1. (a) Shape of RGF, (b) RGF ($\theta = 45°$)

Since we are interested in Gabor kernels in that they can indicate some specific directions of images, we utilize them only as a directional estimator. Figure 1 (b) shows a Gabor kernel with a specific direction, therefore it is expected that correlating it with any image of the same direction will result in a high correlation value.

When we filter an image with Gabor kernels, the size of image needs to be down-sampled and it is apparent that small size of kernel reduces complexity offi ltering. The advantage of using down-sampled image is that slight change of details is negligible to some extent. Moreover, main energy flow (direction) can be still held in the down-sampled image.

Figure 2. Lena images filtered by Gabor kernels with 0°, 45°, 90°, 135° orientations (from left to right)

The filtered Lena images with Gabor kernels of different orientation are shown in Figure 2. As shown in Figure 2, each image has different energy along Gabor filter direction. Since similar orientation of image and direction of filter result in large energy, we evaluate energy by calculating standard deviation of each filtered image. From left to right image in Figure 2, the standard deviations are 29.03, 37.32, 42.92, 26.45, then we can figure out that 90° is more close to the orientation of Lena image.

To estimate the orientation of a host image, we need to find the location of the maximum standard deviation. For this purpose, we take an exhaustive search method of filtering the host image with Gabor kernels having orientation from 1° to 180° in the step of 1°. Despite of the small size of the images, this process is still of high complexity, therefore we employ a hierarchical approach which consists of following three steps. In the first step, standard deviation is calculated at every 20°. Therefore we evaluate 9 standard deviation values. Among them, we select an angle giving the largest value and its immediately adjacent angle giving larger standard deviation value than the other one. These two angles identify a selected interval of 20°. At the second step, we divide the selected interval to sub-intervals of 5° and calculate standard deviation values for each sub-interval again as before. In this step, we get 4 standard deviation values. We apply the same rule as above to find a candidate sub-interval of 5°. At the final step, the selected sub-interval is divided into 5 sub-intervals of 1° and only the largest standard deviation value is found. The corresponding angle is regarded as the orientation of image. Since Gabor filtering and standard deviation calculation is performed 18 times in total, which is 1/10 of the full range search case, it is obviously a way of computational reduction.

2.2 Watermark embedding

Once the orientation of the host image is estimated, an embedding pattern is generated with reference to a user key and the orientation of a host image. For this purpose, we firstly generate a 1-D reference pattern W, determined by a user key. W is either –1 or +1, and its length is k. Suppose a transforming function is $\mathcal{X}$, which transforms the host image I into a signal C in a selected embedding domain, then the relation is $C = \mathcal{X}(I)$. When the size of the transformed host image is MxN, W is repeated over horizontal length N and the repeated sequence is denoted by W_{1D}. When N is not an integer multiple of k, W is repeated by J times where $J \cdot k < N$ and the remaining (N-J·k) positions are padded with zero value. Since the embedding domain is 2 dimensional, W_{1D} is

repeated through each row again by M times. The final result is W_{2D} of size MxN which is the same size as the transformed signal C. An example of reference pattern is shown in Figure 3 (a). Since the same 1-D pattern is repeated in each row, there are seen vertical lines. For this reason, we call it having an orientation of 90°.

(a) (b)

Figure 3. (a) W_{2D} (90°orientation), (b) W_{2D} (50°orientation)

In an actual embedding, we use a reference pattern that has the same orientation as the host image. Suppose a host image has an orientation of 50°, then, the reference pattern having an orientation of 50° as shown in Figure 3 (b) is used in embedding. Embedding process is simply adding W_{2D} to C as described in (3), where s is a weighting factor and $\tilde{C}$ represents an embedded signal.

$$\tilde{C} = C + s \cdot W_{2D} \tag{3}$$

W can be generated [8] by either CDMA fashion or M-ary modulation, in addition it can be also ECC (Error Correction Code) encoded. The half portion of W_{2D} is repeated with W, and the other half portion is zero-padded. The final W_{2D} is generated by randomly changing the position of each.

2.3 Watermark extraction

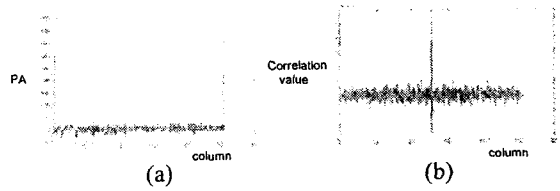
The 2 dimensional search generally required by estimation of geometrical distortions is one of the main source of computational burden. If the 2-D search can be separated into two 1-D searches or reducedt o one 1-D search, then the complexity can be substantially reduced. In this paper, we use the so called, projection average (PA) approach which makes 1-D separation possible. Simply speaking, it is to project the signal onto an axis perpendicular to the orientation of host image, and then to evaluate the average of the projected values. PA of 2-D reference pattern W_{2D} having 90° of orientation as in Figure 3 (a) is described as following.

$$PA_{W_{2D}} = \frac{1}{M}\sum_{row} W_{2D} \tag{4}$$

Since the projection and averaging is done through each column (or 90° orientation), $PA_{W_{2D}}$ in (4) is exactly the same as 1xN W_{1D}. Similarly, PA of $\tilde{C}$ assuming 90° orientation of a host image can be derived as following.

$$\begin{aligned} PA_{\tilde{C}} &= \frac{1}{M}\sum_{row}(C + s \cdot W_{2D}) \\ &= \frac{1}{M}\sum_{row} C + \frac{1}{M}\sum_{row} s \cdot W_{2D} \\ &= PA_C + s \cdot PA_{W_{2D}} \end{aligned} \tag{5}$$

As mentioned previously, $PA_{W_{2D}}$ is the same as W_{1D}, therefore auto-correlating $PA_{\tilde{C}}$ results in several correlation peaks whose period is k. Therefore comparing the peaks of $PA_{W_{2D}}$ and $PA_{\tilde{C}}$ can inform us the degree of geometrical distortions. Figure 3 (a) shows an example of generated W_{2D}, whose PA direction is along column (90° orientation). $PA_{\tilde{C}}$ and its ACF are shown in Figure 4 (a) and (b), respectively. When the watermarked image has undergone rotation or translation, the period of peaks in ACF remains unchanged, however, the period changes in case of scaling. In other words, the proposed method is free from rotation, therefore the only concern over the obtained peaks is the scaling information for proper recovery of watermark. Finally, according to the scaling information, we re-scale the PA in order to extract message.



(a) (b)

Figure 4. (a) $PA_{\tilde{C}}$, (b) ACF of $PA_{\tilde{C}}$

4.Exper imental Results

In the experiment, we use the host image of 256x256 Lena and embedding domain is assumed to be a spatial domain. However, instead of spatial domain, other transform domain is applicable as well. Since simulation was carried out on the spatial domain, the weighting factor s in (3) is to be determined to exploit so-called spatial masking effect [8]. In order to obtain a weighting factor s, we used the edge filter in a similar manner by Kalker [9]. The weighting factor s is given by

$$s = \left[\frac{|\log(|I_f|)|}{\log(|I_f|)_{\max}}\right] \cdot t \tag{6}$$

where I_f represents a filtered image and t is a gain factor which is set to 3 in the experiments. PSNR value according to different t's are shown in Table 1.

Table 1. Gain factor and PSNR

t	PSNR[dB]
1	54.85
2	48.82
3	45.31
4	42.80
5	40.88

Gabor estimation results are shown in Table 2. As shown in the results, the estimation of orientation by Gabor kernels is a little bit inaccurate. This calls for additional search around the estimated orientation to further refine the estimation.

Table 2. Gabor estimation and orientation differences between expected and estimated directions

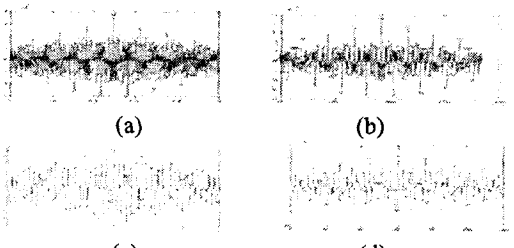
Distortion	Parameter	Image: Lena Diff.	Fishing boat Diff.	Baboon Diff.	Peppers Diff.
Scale Up|Down	0.5	0	0	1	0
	2.0	0	0	1	0
	3.0	0	0	1	0
Aspect ratio change (row : column)	1:1.1	0.4	-0.1	0	0
	1:1.2	-0.2	-0.3	0.4	0.3
	1:1.3	0.2	0.3	0.2	0.7
	1:1.4	-0.4	-0.2	0.4	1.1
	1:1.5	-1.1	0.3	0	1.5
	1:1.8	-4.3	-1.0	1.1	-5.6
Rotation (degree)	5	1	-1.0	-1	1.0
	10	1	-2.0	-5	1.0
	20	1	-2.0	-5	1.0
	30	3	-3.0	-7	-1.0
	45	6	-1.0	-6	-5.0
Translation (+cropped) (%)	6	2	0	0	1
	12	2	-1	-1	3
	23	3	-1	-1	0
	43	6	24	-2	1

(Diff. means difference in degree (°) between the true orientation of the host image and that of the attacked image; $\lambda = 64$ and $\sigma_x = \sigma_y = 3$, filter size: 15x15, image size: 1/16 of original image)

Some examples of distortions and estimated periods are shown in Table 3. Since the original period is 16, distorted image can be recovered from each estimated period to the original.

Table 3. Distortion and estimated period (Original period: 16)

Distortion	Estimated period
Scale Up (200%)	32
Aspect ratio change (1:1.5)	24
Rotation (38°)	16
Translation (23%)	16



(a) (b) (c) (d)

Figure 5. ACF corresponding to distortions in Table 3

When we need to obtain the information of periodic peaks, we use ACF of PA, however, in case of severe distortions, an additional method to refine the results is necessary. One possibility is taking DFT, and alternatively ACF can be used to find periodic peaks. Since the ACF is symmetric, the half side can be used to calculate ACF. These results are shown in Figure 5 (a)–(d).

4. Conclusion

In this paper, we propose a method relatively robust to geometrical attacks. By using the Gabor filter, we extract the orientation of image and based on the orientation information, a reference pattern is periodically generated. The detection step uses the PA approach which projects the pattern to the estimated orientation. After the PA procedure, 2-D becomes 1-D processing which is computationally much simpler. When Gabor filtering instead of correlation is used, it increases complexity. For instance, if filter size is MxM, then the computation of Gabor estimation is MxM times larger than the correlation method. Therefore the proper selections of filter size and down-sampling factor are important. We leave these as our future work.

References

[1] S. Pereira and T. Pun, "Robust Template Matching for Affine Resistant Image Watermarks," *IEEE Ttrans. on Image Processing*, vol. 9, no. 6, pp. 1123-1129, June 2000.

[2] M. Kutter, "Watermarking resisting to translation, rotation and scaling," In *SPIE Conf. On Multimedia Systems and Applications*, vol. SPIE 3528, pp. 423-431, 1998.

[3] S. Voloshynovskiy, F. Deguillaume, and T. Pun, "Content adaptive watermarking based on a stochastic multiresolution image modeling," In *EUSIPCO '2000*, Tampere, Finland, 2000.

[4] F. Deguillaume, S. Voloshynovskiy and T. Pun, "A method for the estimation and recovering from general affine transforms in digital watermarking applications," In *Proc. SPIE Electronic Imaging, Security and Watermarking of Multimedia Contents IV*, vol. SPIE 4675, pp.313-322, 2002.

[5] H. J. Shim and B. Jeon, "Rotation, Scaling, and Translation Robust Image Watermarking Using Gabor kernels," In *Proc. SPIE Electronic Imaging, Security and Watermarking of Multimedia Contents IV*, vol. SPIE 4675, pp.563-571, 2002.

[6] L. Chengjun and H. Wechsler, "A Gabor feature classifier for face recognition," In *Eighth IEEE Int. Conf. on Computer Vision,ICCV 2001*, vol. 2, pp.270-275, 2001.

[7] D. M. Weber, D. P. Casasent, "Quadratic Gabor filters for object detection," In *IEEE Trans. on Image Processing*, vol.10 ,no . 2, pp. 218-230, Feb. 2001.

[8] G. C. Langelaar, I. Setyawan, R. L. Lagendijk, "Watermarking digital image and video data. A state-of-the-art overview," In *IEEE Signal Processing Magazine*, vol. 17, no. 5, pp.2 0-46, Sept. 2000.

[9] T. Kalker, G. Depovere, J. Haitsma, and M. Maes, "A video watermarking system for broadcast monitoring," In *Proc. SPIE Electronic Imaging, Security and Watermarking of Multimedia Contents*, pp. 103-112, 1999.