Journal of Korea Multimedia Society Vol. 12, No. 6, JUNE 2009(pp. 815-823)

A New Mobile Watermarking Scheme Based on Display-capture

Jong-Wook Bae†, Sung-Hwan Jung††

ABSTRACT

Most of existing watermarking schemes insert and extract a watermark, focusing on the visual conservation of an original image. However, existing watermarking schemes could be difficult for a watermark detection in case of various distortion caused by display-capture devices. Therefore, we propose a new display-capture based mobile watermarking scheme. The proposed watermarking scheme is a new concept for embedding a watermark, which uses the generated image instead of a given original image. For effective watermark decoding, we also present a method for detecting the background image whose error bit can not be corrected because of various heavy distortion and for avoiding it from the decoding process. For this scheme, we adopt distortion coefficients of camera calibration when we separate a background image from a captured image. For finding available correction bits of ECC through the decoding process, we capture 30,000 images and then calculate the separation ratio of a background image and the average error bits per an image. As experimental result, the separation ratio of a background image is about 96.5% in 30,000 captured image. And the false alarm ratio shows about 5.18×10^{-4} in the separated background image. And also we can confirm the availability of real-time processing because the mean execution time is about 82ms per an image for capturing and decoding.

Key words: Mobile watermarking, Display-capture, PC camera, DRM

1. INTRODUCTION

As the computer and Internet technology grow, conventional analog contents such as text, image, audio and video have been converted to digital contents. However, digital contents can not only be copied without the loss of the original but also be distributed without the approval of authors. Therefore, a copyright protection technology for the ownership guarantee of digital contents is required. An authentication technology also helps prevent the illegal distribution and the illegal use of digital contents.

Digital watermarking is a technique which allows an individual to add hidden copyright notices or other verification messages to digital contents so that watermarked contents can not be distinguished by the human naked eye or hearing[1-3]. There are a number of researches connected with watermarking technologies. Most of researches are related with robust watermarking and some of researches are related with fragile watermarking. Recently techniques about print-scan watermarking have also been studied[4-5]. However, the research about the mobile watermarking technology for the authentication in the mobile environment is insufficient now, compared with the current fast diffusion of mobile devices.

We propose a mobile watermarking scheme using display-capture related devices such as a PC

※ Corresponding Author : Jong-Wook Bae, Address : (641-773) 9 Sarim-dong, Changwon, Kyungnam, Korea, TEL : +82-55-213-3819, FAX : +82-55-286-7429, E-mail : bae@changwon.ac.kr

Receipt date : June 16, 2008, Approval date : Feb. 26, 2009

† Dept. of Computer Engineering, Changwon National University

†† Dept. of Computer Engineering, Changwon National University
(E-mail : sjung@changwon.ac.kr)

※ This work was supported by Korea Science and Engineering Foundation Grant (S2-2007-000-01252-1).

camera and LCD of mobile phones. Most of existing watermarking schemes embed a watermark using an original image and focus on quality of a watermarked image. But the proposed mobile watermarking scheme is a new concept which is unlike typical watermarking scheme. It generates a proper image instead of using a given original image to insert watermark information.

In this paper, we first refine a image generation and decoding method based on our previous research[6]. And then, we also attempt the effective watermark decoding technique from a captured image in the display-capture environment. The proposed mobile watermarking system based on display-capture is described in Figure 1.

If a customer requests a MMB (Mobile Merchandise Bond) company to issue a MMB and then the company can generate the MMB image, which consists of the watermark-embedded background image and the surface-text information. The company can simply register it to a DRM system. If the customer pays money, then the company transmits the MMB image to the customer's mobile device through the Internet.

When the customer buys goods at a member store, he or she may display the MMB image in his or her mobile device, and then the MMB image of the mobile device is captured by a PC camera in the member store. The MMB information is able

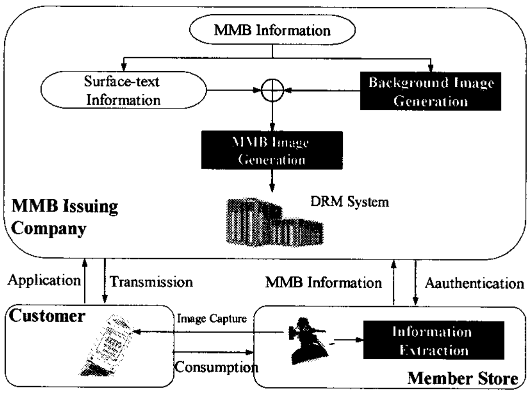


Fig. 1. General block diagram of mobile watermarking system

to be extracted from the captured image, and the authentication through the DRM system is confirmed. Therefore, now, the final transaction of the MMB can be processed.

In this paper, we will focus not only the MMB image generation but also the detection of the MMB information from the captured image. That is, the watermark decoding process. For the effective decoding, we also present a method for avoiding the background image whose error can not be corrected by ECC (Error Correction Code).

2. MMB IMAGE GENERATION

2.1 Background image generation

A MMB image consists of a background image and a text for displaying information visually on the background image. The background image is a generated image for inserting the watermark information. We can generate the background image by creating unit-elements and separate lines.

An example of generated background image is in Figure 2. A unit-element consists of 2×2 or 4×4 pixels. It can express intensity levels of k-steps. For the easy recognition of the unit-element, the background image has separate lines of 2×2 or 4×4 pixels size among unit-elements. For example, when insertion bit is 1, unit-element's intensity level is higher than the intensity value of a separate line. And when insertion bit is 0, unit-element's intensity level is lower than the intensity value of it.

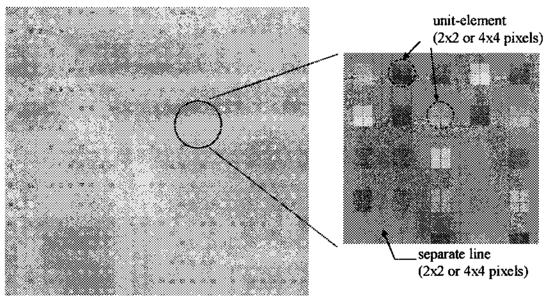


Fig. 2. An example of generated background image

Therefore, we can calculate the potential insertion capacity by equation (1) when the height and the width of background image are H, W respectively.

$$Potential\ Insertion\ Capacity = \frac{W-w_s}{w+w_s} \cdot \frac{H-h_s}{h+h_s} \cdot \log_2 k\,[bit] \quad (1)$$

where h,w are pixels of a unit-element and h_s,w_s are pixels of a separate line.

In addition, we need a technique to prevent error bits of watermark caused by distortions of a captured image. We apply the ECC to generate a robust sequence against the image distortions. Finally we can get the generated background image.

To apply efficient ECC code, after skipping the background images whose error bit can not be corrected in decoding processing, we experimentally measure the maximum error bits in the various distortions caused by the display-capture process. This decision was made by referencing a general false alarm rate of 10^{-3} in the watermark detection. And then we apply an error correction level that can correct the measured maximum error bits. We experimentally apply the BCH (Bose, Chaudhuri, Hocque-nghem) code[5].

2.2 Combination of background image and surface-text

In general, a typical paper merchandise bond has a visual text and a background pattern. The visual text can be used to display the necessary information for a purchaser or a seller. In this paper, let us call this visual text as the surface-text. The surface-text is actually a visual image to display the MMB information on the MMB image.

To get a finally generated MMB image, we must combine the background image and the surface-text. However, if we overlap the surface-text directly on the background image, some parts of the background image could be lost, so that the detection process can be difficult later. Therefore, we use a color image to express the surface-text information, so that we can minimize the loss of the MMB information inserted in the background image. The process of the final MMB image generation is depicted in Figure 3.

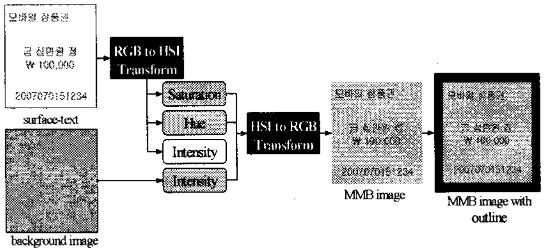


Fig. 3. Process of MMB image generation

At first, a color image which presents the surface-text information is created. Next, it is changed from RGB to HSI color space. Then, we replace its intensity channel with the intensity channel of the background image, and transform HSI to RGB. Finally, we enclose the transformed image with a black border, and then the MMB image is generated like the right side picture of Figure 3. The black border is used to correctly recognize the boundary when we extract the background image in detection process.

3. INFORMATION DETECTION OF WATERMARK-EMBEDDED IMAGE

3.1 Color image capture and acquiring grayscale image

A PC camera captures the watermark- embedded color image displayed on the LCD of a mobile phone. Therefore, various distortions can be caused by the display-capture process. Some of distortions are caused by human's habit. So we can reduce it by using hardware device such as a fixed phone holder. We can also reduce the effect of external unnecessary lighting by installing a blackout box[6].

We must acquire a grayscale image inserted the watermark information from a captured color image. The acquiring process is the inverse process of Figure 3. First, the captured color image is transformed from RGB to HSI color space. Next, the grayscale image is acquired from the intensity channel of HSI color space and so on.

3.2 Background image separation

An acquired grayscale image might have other distortions such as rotation, scale, translation, etc. A background image separation is separating a pure background part from the grayscale image by removing its outer part and correcting various geometry distortions.

First, we find four coordinates in the grayscale image and calculate distortion coefficients by the following equation:

$$(x,y)=\left(\frac{p_1X+p_2Y+p_3}{p_7X+p_8Y+p_9},\frac{p_4X+p_5Y+p_6}{p_7X+p_8Y+p_9}\right) \qquad (2)$$

where x, y are corner coordinates of the original background image, X, Y are corresponding corner coordinates of the captured grayscale image, p_1 ~ p_9 are correction coefficients[7]. Next, we can correct the distorted grayscale image using correction coefficients. Finally we separate the background image from the grayscale image.

We can easily detect boundary lines due to the black border outside the grayscale image. The following is the process to find four corner coordinates and separate a background image.

(1) Get an edge image by applying Canny's algorithm.

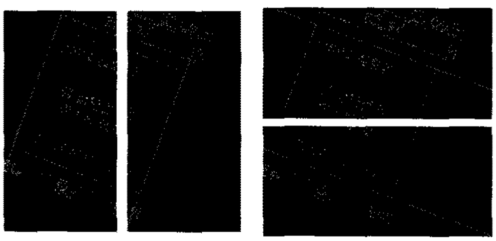


Fig. 4. A vertically and horizontally separated image

(2) Separate the edge image into two sub images by dividing it vertically and repeat it horizontally for reducing the complexity and detecting the horizontal and vertical lines exactly[8].

(3) Perform Hough transformation on four sub-images and detect one straight line from each sub-image.

(4) Calculate four coordinates of intersection points of four detected straight lines using a linear equation.

(5) Separate a background image by equation (2) using four coordinates from the captured grayscale image like Figure 5.

A background image can not be separated properly because Hough transformation could not always detect lines correctly in the preprocessing step or a PC camera could not always capture the MMB image correctly in the capture process. In these cases, the extracting of the inserted information can be difficult because of the wrongly separated background images. Therefore, we need a method to skip the wrongly separated background images in the decoding process.

In this paper, we calculate D(%), the ratio of the average difference to the average line length by equation (3). This value presents the degree of projective distortion of a separated background image. If this value of a separated background image is higher than a specific value, this background image is considered as the wrongly separated background image. So this background image is skipped from the decoding process.

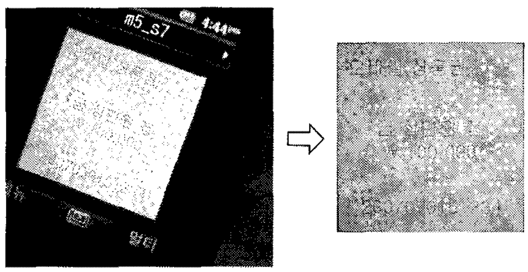


Fig. 5. Captured grayscale image and separated background image

$$
\begin{aligned}
Line_i &= \sqrt{(x_i + x_{i\oplus 1})^2 + (y_i + y_{i\oplus 1})^2} \\
&\quad < i\oplus 1 : (i+1)\,\mathrm{mod}4 > \\
Aver_{Line} &= \frac{1}{4}\sum_{i=0}^{3} Line_i \\
Aver_{diff} &= \frac{1}{6}\sum_{i=0}^{2}\sum_{j=i+1}^{3} |Line_i - Line_j| \\
D(\%) &= \frac{Aver_{diff}}{Aver_{line}} * 100
\end{aligned} \tag{3}
$$

Where $Line_i$ is the i-th line length in lines which connects four detected coordinates, $Aver_{line}$ is the average length of four lines, $Aver_{diff}$ is the average difference of each line length, and $D(\%)$ is the ratio of the average difference to the average line length. And $\oplus$ means modulo-4 operation of two variables.

3.3 Decoding for watermark detection

We transform each unit-element into a code word as Figure 6 in case that an intensity level, k is 2-step.

To cope with the distortion still remained in the MMB image after the above mentioned pre-processing step, we use the higher resolution capture scheme and the median filtering technique as follows: We capture the watermark-embedded image displayed on LCD as 4 or 2 times high resolution. So, for example, the 2×2 or 4×4 pixel sized unit-element of the background image is captured into 8×8 pixel size.

Because the background image separated from captured image has unit-elements of 8×8 pixel size, we sort all 64 pixels in a unit-element by the intensity value and select the median pixel value, V_u. Also, in the same way, we sort all 512 pixels in a separate line enclosing the unit-element by the intensity value and select the median pixel value, V_s. Then we decide a code word by

$$
\text{code word of unit-element} \begin{cases} 1, & V_u > V_s \\ 0, & otherwise \end{cases} \tag{4}
$$

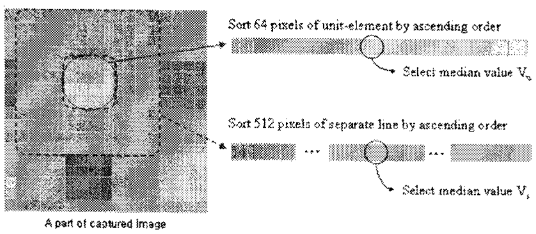


Fig. 6. Code value selection of unit-element

According to mapping the pixel value of a unit-element into a code word, the unit-element of the watermark-embedded image is changed to a code word. Finally we can recover the watermark information of a MMB image from the code word through the BCH decoder.

4. EXPERIMENTAL RESULT

We used LG and Samsung mobile phones of table 1 for display of a MMB image in this experiment. In addition, table 1 shows the unit-element size, background image size and black border size, etc. We can calculate the potential insertion capacity by using equation (1) and table 1. Table 2 also shows our experimental environment for capturing and decoding the MMB image.

Table 1. A specification of mobile phone used in the experiment.

Phone type	sd9230	lb2800	v8400
LCD size	1.7 inch	2 inch	1.9 inch
LCD resolution	160×120	320×240	320×240
Display resolution	120×120	232×232	320×240
Unit-element size	2×2	4×4	4×4
Background image size	94×94	188×188	188×188
Black border size	26×26	44×44	132×52
MMB image size	120×120	232×232	320×240

Table 2. Experimental environment.

Item	Specification
CPU	Intel Core 2 Duo E6300
RAM	1GB
OS	Windows XP
Language	Visual C++
PC camera	VIJE TALK CCD
Capture resolution	480 × 480 pixel

A unit-element size can be changed flexibly according to LCD size and LCD resolution. In case of sd9230, the resolution that can display on LCD, is 120×120 pixels. So the unit-element size is 2×2 pixels and the background image size is 94×94 pixels, and the black bolder size is 26×26 pixels. In case of v8400, the resolution that can display on LCD, is 320×240 pixels. So we can set the unit-element size by 4×4 pixels. And also we can adaptively set the black border size because the image size displayed on LCD of each mobile phone is different.

In the decoding of the background image, error bits can remain on a extracted watermark because of distortion caused by the display-capture process. Therefore, ECC which can correct error bits must be applied to them. We measured error bits which can be caused by mobile phones of table 1. We find a certain point whose error bits increase rapidly by using the difference ratio, D of equation (3). If the difference ratio, D of a background image is over this point, the background image is excepted from the decoding process. The experimental process is as follows.

(1) Generate binary sequences by 10 random seed keys.

(2) Experimentally create 10 test MMB images based on its information.

(3) Input test images to a mobile phone then display them on its LCD.

(4) Capture 1,000 images per each MMB image. So, total number of captured image is 10,000 per each mobile phone.

(a) LG sd9230 (c) LG lb2800 (c) Samsung v8400

Fig. 7. An example of a MMB image

(5) Separates a background image from the captured image.

(6) Calculates the difference ratio, D of the separated background image.

(7) Measured error bits by decoding the separated background image.

Figure 8 shows the number of background images separated from grayscale images by the difference ratio, D in case of sd9230. The grayscale image are acquired by transforming color space from a captured MMB image.

When the difference ratio, D is about 0.7, we have the most number of background images and the distribution of background images reveals a form of a normal distribution having a mean by this point. And also If D is over about 0.9, the number of background image decreases rapidly. If D is over about 3.0, the background images are seldom.

Figure 9 shows the average error bits of the

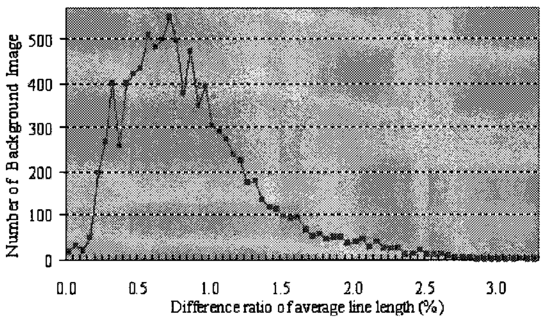


Fig. 8. The number of separated background image by the difference ratio, D

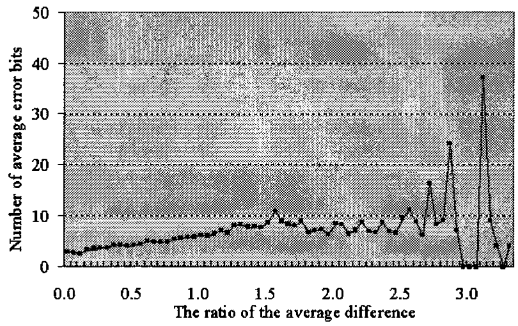


Fig. 9. Average error bits by the difference ratio, D

separated background images according to the difference ratio, D in case of sd9230. The error bits means the number of bits detected wrongly when a background image is decoded through decoding process. And also the average error bits presents the average of error bits of background images divided by the difference ratio, D. For example, when the difference ratio, D is 0.95, and 394 background images were separated from grayscale images, then the average error bits of 394 images is about 5.82 bits.

If the difference ratio, D is over about 2.7, the background image may not be separated exactly, because the average error bits increase rapidly when D is more than 2.7. Therefore, we except the images whose the difference ratio of the separated background image is over 2.7, from the decoding process.

Table 3 shows the ratio of the exactly separated background image and the number of error image when D is under 2.7.

We can exactly separate the background images of 96.85% from 30,000 captured images. And the number of error image over 40 error bits is below about 10 in each mobile phone. So we could apply an error correction level that can correct 39 error bits, and we used the BCH code (511, 219, 39) in this experiment. Therefore, we can insert the watermark information of 219 bits.

Table 3. The result of background image separation and decoding (D<2.7).

Phone type / Error bits		sd 9230	lb 2800	v 8400	Average	Error ratio(%)
Background image separation ratio (%)		96.57	98.38	95.60	96.85	
Number of error image	over 30 error bits	62	16	33	37.0	0.382
	over 35 error bits	22	10	15	15.7	0.162
	over 40 error bits	9	7	6	7.3	0.076
	over 45 error bits	5	5	5	5.0	0.052
	over 50 error bits	2	5	4	3.7	0.038

In this experiment, the potential insertion capacity of a MMB image is 529 bits, because we created the background image using 23×23 unit-elements by table 1 and equation (1). We need only 511 bits for the BCH encoding, instead of 529 bits. Therefore, we can raise whole error correction efficiency by effectively arranging the remained 18 bits.

Table 4 shows the decoding result for 30,000 captured images with applying the ECC code.

Because the average run-time per an image is about 82 ms, it is enough fast to process by real-time for capturing and decoding. Then we can confirm availability of real-time processing for capturing and decoding in the display capture environment.

By the last step, we draw a comparison between our previous research[6] and this experiment's result in table 5. In previous research, we suggested a new concept on the subject of a mobile watermarking and focused on watermark-embedded image generation. Therefore, we confirmed a type of distortions caused by various environment and we corrected only a distortion caused by a change of LCD brightness. So, we did decoding process on the assumption that other distortions were corrected. And we also applied just Viterbi ECC and test only one mobile phone.

In the previous research, insertion capacity were

Table 4. Decoding result with applying the ECC code.

Phone type	sd9230	lb2800	v8400	Average
Background image separation ratio(%)	96.28	95.86	97.47	96.54
Number of error image	8	5	2	5.0
False alarm ratio	8.31×10^{-4}	5.21×10^{-4}	2.05×10^{-4}	5.18×10^{-4}
Average execution time (ms)	96	78	72	82

Table 5. A comparison between the previous research and our experiment

	Previous research[6]	Our experiment
Iteration number	4	1
Insertion capacity	96 bits	219 bits
ECC code	Viterbi	BCH
Display resolution	120×120	120×120 (in case of sd9230)
Average execution time	Comparison not available (because of manual correction of some distortions)	82 ms

96 bits due to 4 iteration insertion and Applying Viterbi ECC and 120×120 display resolution. However, in this research, we can insert 219 bits to watermark information due to finding correctable error bits and applying BCH ECC. Furthermore, for effective watermark detection, we also apply to the system the avoiding method from the decoding process by adopting a certain level of the difference ratio D.

5. CONCLUSION

In this paper, we presented an effective watermark detection technique for the display-capture based mobile watermarking. The proposed mobile watermark technique is a new concept not to focus on the visual conservation of an original image but to generate a proper MMB image for the watermark to be inserted. In this research, we dealt with a generation and effective detection technique of a watermark-embedded image. It is a method for detecting the background image whose error bits can not be corrected because of various heavy distortion and for avoiding it from the decoding process. For this scheme, we adopt distortion coefficients of camera calibration when we separate a background image from a captured image.

We created watermark-embedded images and inputted them in three mobile phones. And we separated the background images of average about 96.5% from 30,000 captured images. And the false alarm ratio is about 5.18×10^{-4} in the separated background image. And also the average time for capturing and decoding per an image, is about 82 ms. Therefore, we can confirm the availability of real-time processing for capturing and decoding.

In further research, we will study a scheme to insert more information by extending intensity level, k of a unit-element and to reduce the error rate by using more frames.

REFERENCES

[1] F. Hartung and M. Kutter, "Multimedia Watermarking Techniques," *Proceedings of IEEE*, Vol.87, No.7, pp. 1079-1107, 1999.

[2] N. F. Johnson, Z. Duric and S. Jajodia, *Information Hiding: Steganography and Watermarking Attacks and Countermeasures*, Kluwer Academic Publishers, Norwell, MA., 2001.

[3] I. J. Cox, M. L. Miller, J. A. Bloom, *Digital Watermarking*, Morgan Kaufmann Publishers, San Francisco, Calif., 2002.

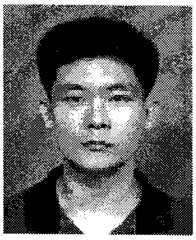
[4] I. J. Cox, J. Kilian, T. Leighton and T. Shamoon, "Secure Spread Spectrum Watermarking for Multimedia," *NEC Research Institute*, Technical Report 95-10, 1995.

[5] Kyeong-Cheol Yang, "Error Correction Code Technology," *The Institute of Electronics Engineers of Korea, Journal of telecom*, Vol. 9, No.2, pp. 46-56, 1993. 12.

[6] Jong-Wook Bae, Bo-Ho Cho, and Sung-Hwan Jung, "A New Display-Capture Based Mobile Watermarking System," *LNCS, Springer-Verlag*, Vol.4319, pp. 1075-1084, 2006. 12.

[7] R. Sukthankar, R. G. Stockton and M. D. Mullin, "Self-Calibrating Camera-Assisted Presentation Interface," *Proceedings of International Conference on Control*, Auto-

mation, Robotics and Vision (ICARCV), 2000.
[8] Myoung-jin Shin and Eui-young Cha, "Normalization Algorithm of Licence Plate using Baseline Searching Method," *Proceedings of The KMMS Fall Conference*, Vol.10, No.1, pp. 393-396, 2007. 5.



Jong-Wook Bae

received the B.S and M.S in Computer Engineering from Changwon National University in Korea, in 2001 and 2005, respectively. He is currently a Ph.D. candidate in the same University. His research interests are in Watermarking, Image Steganography and Image processing.

Sung-Hwan Jung

received the B.S, M.S, and Ph.D. in Electronic and Computer Engineering (Information and Communication & Image Processing) from Kyungpook National University, Korea, in 1979, 1983, and 1988, respectively. From 1983 to 1985, he worked for the Electronics and Telecommunications Research Institute in Korea as aresearch staff, where he had experienced some national research projects including developing a portable computer. In 1988, he joined the faculty of the Department of Computer Engineeringat Changwon National University in Korea, where he is currently working as a full professor. From 1992 to 1994, he was on apostdoctoral research staff of the Department of Electrical and Computer Engineering at the University of California at Santa Barbara (UCSB). From 1999 to 2000, he also worked for the Colorado School of Mines (CSM) in Golden as a exchange professor. From 2008 to 2009, he had experience on medical information processing at the Dental School of the University of Missouri at Kansas City (UMKC), as a visiting professor. He is an Information System Auditor and P.E. in the area of Information Processing and Electronic Computers. His research interests are in Content-based Image Retrieval, Image Steganography and Watermarking, Internet-based Remote Monitoring, Text Mining, and Bioinformatics