# High Robust Image Embedding Method

Yung-Kuan Chan

***Abstract*—**The concept of data embedding is to embed a piece of critical information in a host image. We call the critical information critical data and the host image after embedding the critical data an embedded image. For taking photos from an embedded image, various images with very high distortion may be created since various resolutions of camera, different positions the camera located at, distinct regions the zoom focuses on may be employed. The traditional image hiding methods usually cannot precisely derive the critical data from the created variation images. This paper hence proposes a high robust image embedding (HREH) method. The experimental results show that the HREH can precisely extract the critical data from the variation image mentioned above.

***Keywords*—** image hiding method, RSA algorithm, digital signature algorithm, image barcode.

## I. Introduction

IN supermarkets and convenience stores, it is very common to notice that the clerks scan the barcodes to check the names and prices of items. However, barcodes generate large coverage on packing; meanwhile, black-and-white barcode patterns compromise the appearance of packing, and, moreover, they can be reproduced easily. If barcodes can be embedded into logos, except for saving more space on packing, it is able to demonstrate descriptions like ingredients and functions and to prevent barcode reproduction.

When policemen stop suspicious drivers, it is common that they require them to identify themselves with ID cards and send their ID card numbers to remote server through the Internet for accessing their related information and confirming their identities. However, ID numbers can be stolen easily for its unprotected property. Except for given ID number, if every citizen receives another password which is embedded in ID card, police officers, when rummage, can transmit suspicious person's ID number and password to remote server for checking(by scanning or photographing, password can be obtained from image) to expose scoundrel's forged identity.

The abovementioned examples need scanner and camera to produce product logos or the images of ID cards and then decode the embedded passwords. However, whether the embedded barcodes and ID passwords can be precisely extracted depends on various factors such as camera's resolution, camera's position and different foci on different areas of embedded image. Unfortunately, traditional image hiding methods are unable to deal with these variations. Hence, this research proposed high robust image embedding to solve these problems.

Yung-Kuan Chan, National Chung Cheng University, Taichung, Taiwan, R.O.C.

## II. Related Works

### A. Information Hiding Method

Information Hiding Method [3,11] (IHM) is a way to study how to hide confidential data, which is to mask secret data into cover image, hoping the combined stego images won't change too much appearance and avoid drawing attention. In 1999, Petitcolas and his colleagues sorted IHM into four applied categories. Among them, Steganography and Digital Watermark are two of the most typical technologies.

Steganography is to hide secret data into original image, which is cover image, to avoid being found. The main goal of this method is for secret communication, and its research emphasis is mostly on how to add hidden data quantity and to raise hidden images' quality. Watermark technology is to hide another image or data into original image, and, in most of cases, it is applied in the attribution of ownership. Its main goal is to announce copy right, and the research emphasis is on how to correctly examine the ownership of media.

### B. Barcode

Barcode is constituted of parallel black-and-white lines. The combination of these lines represents the code of each corresponding character. While reading, scanner decodes it, and, afterwards, the corresponding number can be restored. Next, the meanings and information of barcode can be acquired by keying in information to computer. Barcode system possesses several advantages such as speedy inputting and high accuracy and therefore has been greatly utilized as a tool for automatically collecting information in different industries like manufacturing, logistics and transportation.

There are two kinds of barcodes: one-dimensional barcode and two-dimensional barcode. In one-dimensional barcode, only width of black-and-white lines can store the meanings of characters. In two-dimensional barcode, height is used for demonstrating information. The most common one-dimensional barcodes are ISBN, UPC-A, UPC-E, EAN-8, EAN-13 and Code 39. For two-dimensional barcode, stacked two-dimensional barcodes are Code49 and PDF417, and matrix two-dimensional barcodes are Maxi Code, DataMatrix, Aztec Code and QR Code.

### C. Rivest-Shamir-Adleman Encryption, RSA Encryption

RSA [13] Encryption was proposed by Rivest, Shamir and Adleman in 1977, and it is also the most commonly-used

Public Key Cryptosystem. [4,9,14,16] RSA Encryption renders each user two different large prime numbers $a$ and $b$, making $m=a\times b$, and gives a coprime $p$ for $(a-1)\times(b-1)$. $(p, m)$ is the user's Public key. $D$ is the ready-to-be-encrypted original data, which is called Plaintext. RSA Encryption transforms $D$ into $C=D^p \bmod m$; and $C$ is called Ciphertext. Besides, it provides user a number $s$ to meet the condition of $p \cdot s = 1 \bmod (a-1)(b-1)$, and $(s, m)$ is the user's Private Key. The user can use $D=C^s \bmod m$ to decipher the original plaintext $D$. In RSA Encryption, $D = (D^p)^s \bmod m = (D^s)^p \bmod m$ is permanently established.

### *D. Digital Signature Algorithm, DSA*

Digital Signature [1,2,5,6,10,12,15,17] is an encrypted form providing identity confirmation. Just like stamp, stamping on a document is adequate to prove this document is given by certain person, and denial won't be acceptable.

RSA Encryption can exercise the function of digital stamp. If the person who tries to deliver message uses his private key and encrypt the message, the message will be transformed into a digitally stamped information. Meanwhile, the digitally stamped information only can be decoded by the same person's public key to confirm that the information source is the owner of private key. Likewise, if the information can be correctly decoded, it means the content of information wasn't changed and therefore guarantees the information powerful protection. Hence, digital stamp is able to make sure the received information is unchanged.

## III. High Robust Image Embedding Method (HREH method)

HREH method includes two procedures: Data Embedding Procedure and Data Extraction Procedure. Data Embedding procedure is to put critical data into host image for generating an embedded image. Data Extraction procedure is to take out the embedded critical data from embedded image through scanner and camera. These two procedures will be explained in the following chapter.

### *A. Data Embedding Procedure*

Supermarkets and convenience stores usually check item information according to index, which is saved in database. Thus, they can save item information in the index of database and embed it into the logo pattern and wrapper of the item. When they want to check the item information, they can extract the embedded index from the partial image of logo pattern or wrapper with scanner or camera to find it out.

Except for ID number, every citizen has an extra password which is embedded in ID card (with image embedment, this function can be exercised) when policeman stops a suspicious person, he can transmit his ID number and password to remote server for identity checking (partial ID image can be obtained by scanning or photographing, and password can be extracted from image). Then, the criminal's forged identity can be exposed. If $\boldsymbol{D}$ stands for transmitted ID number and password, and $\boldsymbol{(p_1, m)}$ and $\boldsymbol{(s_1, m)}$ are set as public key and private key from sender side; besides, $\boldsymbol{(p_2, m)}$ and $\boldsymbol{(s_2, m)}$ are set as public key and private key of receiver side. The procedure is as following:

(1) First, sender executes $\boldsymbol{C_1 = D^{p_2} \bmod m}$ and encrypts $\boldsymbol{D}$ with public key $\boldsymbol{p_2}$ at receiver side; besides, $\boldsymbol{C_1}$ is transmitted to receiver side through the Internet.

(2) When receiver captures $\boldsymbol{C_1}$, it calculates $\boldsymbol{D_1 = C_1^{s_2} \bmod m}$ and decodes $\boldsymbol{C_1}$ with its private key $\boldsymbol{s_2}$. If the content of $\boldsymbol{D_1}$ is the same as the content of $\boldsymbol{D}$, it will give a number representing **YES** to $\boldsymbol{D_2}$. Otherwise, it will give a number representing **No** to $\boldsymbol{D_2}$.

(3) Receiver re-executes $\boldsymbol{C_2 = D_2^{p_1} \bmod m}$, and encrypts $\boldsymbol{D_2}$ with sender side's public key $\boldsymbol{p_1}$; meanwhile, $\boldsymbol{C_2}$ will be sent back to sender side through the Internet.

(4) When sender side obtains $\boldsymbol{C_2}$, it calculates $\boldsymbol{D_3 = C_2^{s_1} \bmod m}$ and decodes $\boldsymbol{C_2}$ with its private key $\boldsymbol{s_1}$. If the content of $\boldsymbol{D_3}$ shows the ***YES*** number, which means it is the correct identity; otherwise, the identity is forged.

The abovementioned applications require camera and scanner to access the image of logo pattern, wrapper and ID card and extract the embedded item index and password from image. The goal of Data Embedding Procedure is to hide a certain item index or password into a host image. This procedure is to make sure HREH method can deal with variations like camera's resolution, camera's position and different foci on different areas of embedded image.

In Data Embedding Procedure, at first, it transforms $\boldsymbol{D}$ into a three-digit number $\boldsymbol{d_2d_1d_0}$, which is $\boldsymbol{D=d_2\times d^2+d_1\times d^1+d_0\times d^0}$. And then, a color host image $\boldsymbol{I_{RGB}}$ will be separated into three grayscale images $\boldsymbol{I_R}$、$\boldsymbol{I_G}$、$\boldsymbol{I_B}$ according to three different color components R, B and G. Meanwhile, $\boldsymbol{I_c[x, y]}$ is the color intensity of color component $\boldsymbol{c}$ at $\boldsymbol{I_{RGB}}$ coordinate $\boldsymbol{(x, y)}$, and $\boldsymbol{c=R, G, B}$.

Next, $\boldsymbol{d_2}$、$\boldsymbol{d_1}$、$\boldsymbol{d_0}$ are separately inserted into $\boldsymbol{I_R}$、$\boldsymbol{I_G}$、$\boldsymbol{I_B}$. And then, $\boldsymbol{I_c}$ $\boldsymbol{(c=R, G, B)}$ is cut into several non-overlapping blocks. Each block includes $\boldsymbol{n\times n}$ pixels. Later, random number generator decides $\boldsymbol{d_c/d}$ proportion blocks from all blocks. These blocks are set as convex blocks; the rest of them are concave blocks.

At the same time, each convex block and concave block $\boldsymbol{B}$ will be calculated as following. $\boldsymbol{B[x, y]}$ is block $\boldsymbol{B}$ 's central pixel point, and $\boldsymbol{(x, y)}$ is the coordinate location of $\boldsymbol{B[x, y]}$ at $\boldsymbol{I_c}$. Moreover, ***AVE*** is the average value of all the pixels' grayscale intensity in block $\boldsymbol{B}$. And $\boldsymbol{G}$ is a constant value. Function ***ConvexBlock*****(*B*)** processes every convex block$\boldsymbol{B}$, and function ***ConcaveBlock*****(*B*)** processes every concave block $\boldsymbol{B}$.

Function *ConcaveBlock*(*B*)
(1)If $\lfloor AVE \rfloor+\lfloor\lfloor n/2 \rfloor/2 \rfloor\times G>255$ then
(2) $AVE=255-\lfloor\lfloor n/2 \rfloor/2 \rfloor\times G$
(3)If $\lfloor AVE \rfloor-\lceil\lfloor n/2 \rfloor/2 \rceil\times G<0$ then
(4) $AVE=\lceil\lfloor n/2 \rfloor/2 \rceil\times G$
(5)For $i=0$ to $\lfloor\lceil n/2 \rceil/2 \rfloor$
(6) For $j=0$ to $i$
(7) $B[x-i, y-j]=B[x-i, y+j]=B[x+i, y-j]=B[x+i, y+j]=\lfloor AVE \rfloor-(\lfloor\lceil n/2 \rceil/2 \rfloor-i)\times G$
(8) $B[x-j, y-i]=B[x-j, y+i]=B[x+j, y-i]=B[x+j, y+i]=\lfloor AVE \rfloor-(\lfloor\lceil n/2 \rceil/2 \rfloor-i)\times G$
(9)For $i=\lfloor\lceil n/2 \rceil/2 \rfloor+1$ to $\lfloor n/2 \rfloor$
(10) For $j=0$ to $i$
(11) $B[x-i, y-j]=B[x-i, y+j]=B[x+i, y-j]=B[x+i, y+j]=\lfloor AVE \rfloor+(i-\lfloor\lceil n/2 \rceil/2 \rfloor)\times G$
(12) $B[x-j, y-i]=B[x-j, y+i]=B[x+j, y-i]=B[x+j, y+i]=\lfloor AVE \rfloor+(i-\lfloor\lceil n/2 \rceil/2 \rfloor)\times G$

The distance between $B[x_0, y_0]$ and $B[x_1, y_1]$ is $MIN(|x_1-x_0|, |y_1-y_0|)$. In order to keep minimum change lowest, HREH method sets the distance between all the pixels' grayscale value and $B[x, y]$, which is $\lceil\lfloor n/2 \rfloor/2 \rceil$, as $\lfloor AVE \rfloor$. In concave block $B$, it sets the distance between all the pixels' grayscale value and $B[x, y]$, which is $\lceil\lfloor n/2 \rfloor/2 \rceil-i$, as $\lfloor AVE \rfloor-i\times G$; the distance between all the pixels' grayscale value and $B[x, y]$, which is $\lceil\lfloor n/2 \rfloor/2 \rceil+i$ is set as $\lfloor AVE \rfloor+i\times G$. As for convex block $B$, it sets the distance between all the pixels' grayscale value and $B[x, y]$, which is $\lceil\lfloor n/2 \rfloor/2 \rceil-i$, as $\lfloor AVE \rfloor+i\times G$; the distance between all the pixels' grayscale value and $B[x, y]$, which is $\lceil\lfloor n/2 \rfloor/2 \rceil+i$ is set as $\lfloor AVE \rfloor-i\times G$.

The transformation of convex block and concave block may cause some grayscale values of pixels become bigger than 255. Meanwhile, all the grayscale values of pixels are lowered to the biggest grayscale value 255 in this block simultaneously. Likewise, it probably causes some grayscale values become smaller than zero. At this moment, all the grayscale values of pixels are increased to make the smallest grayscale values equal to zero.

**Function** *ConvexBlock*(*B*)
(1)If $\lfloor AVE \rfloor-\lfloor\lfloor n/2 \rfloor/2 \rfloor\times G<0$ then
(2) $AVE=\lfloor\lfloor n/2 \rfloor/2 \rfloor\times G$
(3)If $\lfloor AVE \rfloor+\lceil\lfloor n/2 \rfloor/2 \rceil\times G>255$ then
(4) $AVE=255-\lceil\lfloor n/2 \rfloor/2 \rceil\times G$
(5)For $i=0$ to $\lfloor\lceil n/2 \rceil/2 \rfloor$
(6) For $j=0$ to $i$
(7) $B[x-i, y-j]=B[x-i, y+j]=B[x+i, y-j]=B[x+i, y+j]=\lfloor AVE \rfloor+(\lfloor\lceil n/2 \rceil/2 \rfloor-i)\times G$
(8) $B[x-j, y-i]=B[x-j, y+i]=B[x+j, y-i]=B[x+j, y+i]=\lfloor AVE \rfloor+(\lfloor\lceil n/2 \rceil/2 \rfloor-i)\times G$
(9)For $i=\lfloor\lceil n/2 \rceil/2 \rfloor+1$ to $\lfloor n/2 \rfloor$
(10) For $j=0$ to $i$
(11) $B[x-i, y-j]=B[x-i, y+j]=B[x+i, y-j]=B[x+i, y+j]=\lfloor AVE \rfloor-(i-\lfloor\lceil n/2 \rceil/2 \rfloor)\times G$
(12) $B[x-j, y-i]=B[x-j, y+i]=B[x+j, y-i]=B[x+j, y+i]=\lfloor AVE \rfloor-(i-\lfloor\lceil n/2 \rceil/2 \rfloor)\times G$

The goal of step 1 to step 4 in **Function *ConcaveBlock*(*B*)** and **Function *ConvexBlock*(*B*)** is to adjust the grayscale values in this block when the grayscale value is smaller than zero or bigger than 255 after it is changed into convex and concave block. In step 5 to step 8, when the distance between與$B[x, y]$ and the grayscale values of pixels is $\lceil\lfloor n/2 \rfloor/2 \rceil+k$, $k=0, 1, \ldots, \lfloor\lceil n/2 \rceil/2 \rfloor$. In step 9 to step 12, when the distance between與$B[x, y]$ and the grayscale values of pixels is $\lceil\lfloor n/2 \rfloor/2 \rceil+k$, $k=1, 2, \ldots, \lfloor n/2 \rfloor-\lfloor\lceil n/2 \rceil/2 \rfloor+1$.

Presuming $n=5$、$G=8$, Figure 1(a) is a block $B$ containing $5\times5$ pixel , and the grayscale intensity average value is $AVE=63.56$, Figure 1(b) is a consequence after changing $B$ into a concave block; Figure 1(c) is a consequence after changing $B$ into convex block.

| | | | | |
|---|---|---|---|---|
| 5 | 6 | 7 | 8 | 6 |
| 6 | 6 | 7 | 3 | 3 |
| 6 | 6 | 5 | 5 | 5 |
| 5 | 1 | 3 | 4 | 1 |
| 8 | 5 | 6 | 6 | 3 |
| 9 | 6 | 9 | 6 | 4 |
| 7 | 7 | 4 | 7 | 7 |
| 8 | 1 | 0 | 8 | 6 |
| 4 | 6 | 8 | 5 | 5 |
| 4 | 7 | 9 | 0 | 3 |

(a) A block $B$ containing $5\times5$pixel

| | | | | |
|---|---|---|---|---|
| 7 | 7 | 7 | 7 | 7 |
| 1 | 1 | 1 | 1 | 1 |
| 7 | 6 | 6 | 6 | 7 |
| 1 | 3 | 3 | 3 | 1 |
| 7 | 6 | 5 | 6 | 7 |
| 1 | 3 | 5 | 3 | 1 |
| 7 | 6 | 6 | 6 | 7 |
| 1 | 3 | 3 | 3 | 1 |
| 7 | 7 | 7 | 7 | 7 |
| 1 | 1 | 1 | 1 | 1 |

(b) The changed concave block

| | | | | |
|---|---|---|---|---|
| 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 |
| 5 | 6 | 6 | 6 | 5 |
| 5 | 3 | 3 | 3 | 5 |
| 5 | 6 | 7 | 6 | 5 |
| 5 | 3 | 1 | 3 | 5 |
| 5 | 6 | 6 | 6 | 5 |
| 5 | 3 | 3 | 3 | 5 |
| 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 |

(c) the changed convex block

Fig. 1 A block B and the changed convex and concave block

### *B. Data Extraction Procedure*

Presuming $I'_c$ is the embedded image after inserting $D$ into $I_c$. While intending extract critical data $d_c$ from $I'_c$, at first, HREH method will divide the range from 0% to 100% into $d$ sessions. The range of value $w_i$ of session $i$ will be:

$$w_i=\begin{cases}[0\%,\ (100/(2(d-1)))\%], & \text{for } i=0,\\ [(i\times100/(d-1)-100/(2(d-1)))\%,\ (i\times100/(d-1)+100/(2(d-1)))\%], & \text{for } 1\le i<d, \text{and}\\ [(100-100/(2(d-1)))\%,\ 100\%], & \text{for } i=d.\end{cases}$$

Then, with scanner and camera to extract a certain area in $I'_c$, the partial area image can be called $R$. It is not allowed to cover any other images except for $I'_c$ in $R$, because there is no critical data embedded outside of the area of $I'_c$. When it is covered in $R$, the situation of data misreading will happen when trying to extract $D$ from $R$.

Later, double-check every single pixel $R(x, y)$, if the grayscale values are bigger than the grayscale values of $R(x-1,$

***y-1)***, ***R(x-1, y)***, ***R(x-1, y+1)***, ***R(x, y-1)***, ***R(x, y+1)***, ***R(x+1, y-1)***, ***R(x+1, y)*** and ***R(x+1, y+1)***, ***R(x, y)*** can be regarded as the central point of certain concave block. If [r = (numbers of central points in convex block)/(numbers of central points in convex block+numbers of central points in concave block)] is located at the session $w_i$, $d_c$ can be regarded as $I$ to obtain $d_c$ by doing so.

## IV. Experiments

Except for providing higher data hiding rate, an excellent image embedding method generates less distortion to host image. The data hiding rate provided by HREH method and distortion rate generating to host image are affected by parameters ***d***, ***G*** and ***n***. When ***D*** is big, ***D*** needs to be changed to a three-digit ***d*** binary number. ; at the same time, the corresponding range of every $w_i$ session will be narrowed. While photographing ***R***, the rotation of photographing, resolutions and camera positions cause ***r***, released from ***R***, will probably locate out of the range of $w_i$, making the decoded information become incorrect. If it is given a bigger parameter ***G***, the embedded host image will get high distortion rate. However, critical data can be extracted correctly. Camera's resolution and the distance between camera and host image will also affect ***n*** while photographing ***R***. The main purpose of this chapter is to use experiments to discover the effects of ***d***, ***G*** and ***n*** to HREH method.

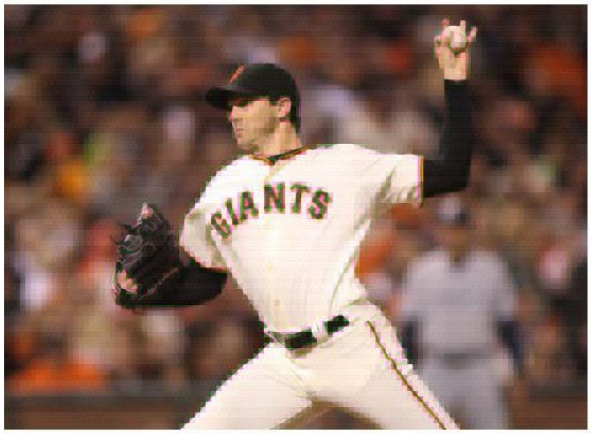
The following experiments will use Figure 2 as host image and set ***d*** as ***11***and ***21***, ***n*** as ***5***, ***7*** and ***9***, ***G*** as ***8***, ***16*** and ***24***. When ***d=11***, ***D=807***; when ***d=21***, ***D=2797***. In the above different parameter combinations, every host image $I'_c$ is taken for 20 partial area images ***R*** from $I'_c$, and then the critical data $d'_c$ is extracted from every single ***R***. If the extracted $D' = d'_R\ d'_G\ d'_B$ is the same as the embedded critical data $D=d_R d_G d_B$, it means HREH correctly extracts critical data embedded in $I'_c$. Table 1 demonstrates these experiment results; ***ACC*** represents the accuracy rate of data extraction. ***ACC*** is defined as the ratio value of the image number of correctly extracted critical data and the total image number by using scanner and camera. Therefore, the bigger ACC is, the more workable the data extraction and embedding method is. Besides, Figure 4 shows partial photographed image areas.

The experiment results suggest that the image quality of embedded image $I'_c$ isn't affected by ***d***. However, when setting a bigger ***d*** value, bigger D value can be embedded; but, higher mistake rate of critical data extraction will be caused easily. If giving bigger ***G*** value, $I'_c$ will obtain better image quality; but, higher data mistake rate of data extraction will be caused easily. If giving bigger ***n*** value, lower mistake rate of data extraction will be made. When $n \geqq 7$, correct critical data can be decoded and the PSNR [7,8] of $I'_c$ doesn't make obvious difference; but, visually, blocks can be noticed easily in $I'_c$. The experiment results also prove that even though the PSNR of $I'_c$ is lower than 25 dB, it is still effortless to find out the original content of $I_c$ from $I'_c$

TABLE I
THE RESULTS OF THE EXPERIMENT

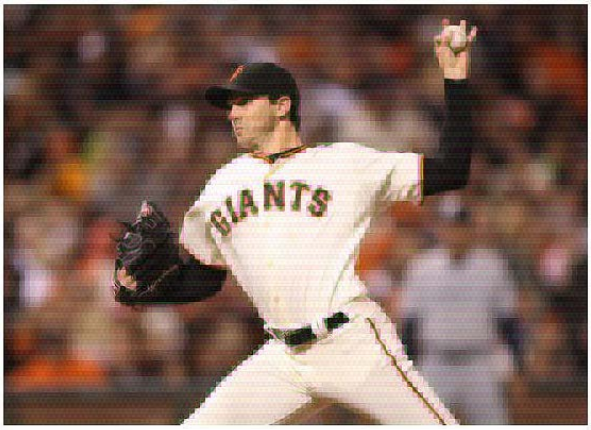
| *n* | *G* | PSNR (dB) | *D* | ACC (%) |
|---|---|---|---|---|
| *5* | *8* | *24.12* | *21* | *35%* |
| | | | *11* | *100%* |
| | *16* | *23..31* | *21* | *40%* |
| | | | *11* | *100%* |
| | *24* | *22.74* | *21* | *50%* |
| | | | *11* | *95%* |
| *7* | *8* | *24.53* | *21* | *85%* |
| | | | *11* | *100%* |
| | *16* | *23.95* | *21* | *100%* |
| | | | *11* | *100%* |
| | *24* | *22.52* | *21* | *100%* |
| | | | *11* | *100%* |
| *9* | *8* | *24.23* | *21* | *100%* |
| | | | *11* | *100%* |
| | *16* | *23.97* | *21* | *100%* |
| | | | *11* | *100%* |
| | *24* | *22.76* | *21* | *100%* |
| | | | *11* | *100%* |


Fig. 2 The original host image

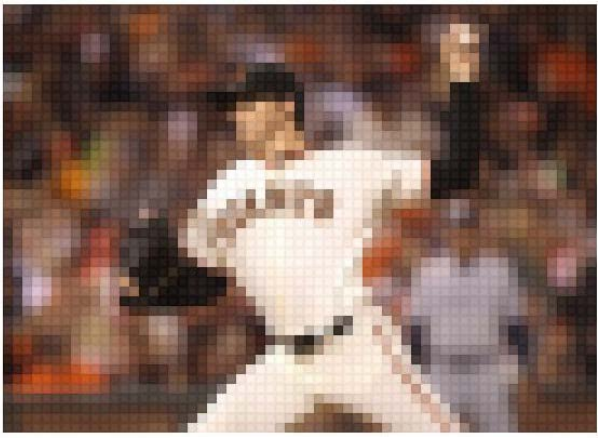



(A)(***n, d, G***)=(***5, 21, 8***)

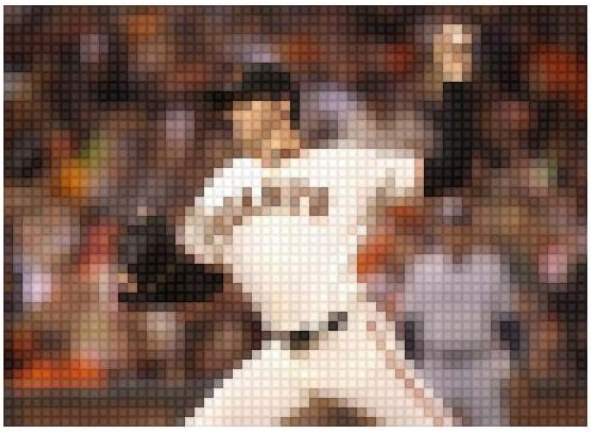

(B)(***n, d, G***)=(***5, 21, 16***)



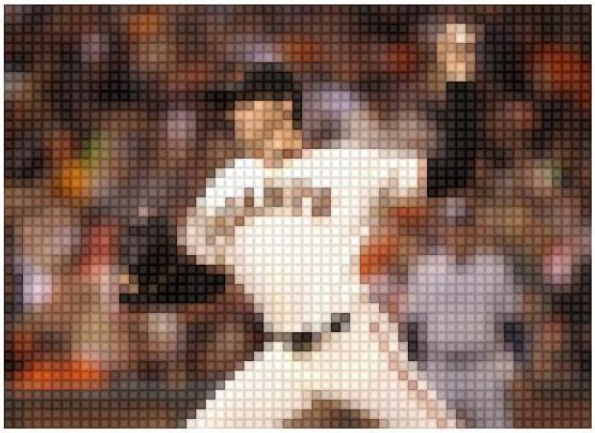

(C)(***n, d, G***)=(***5, 21, 24***)



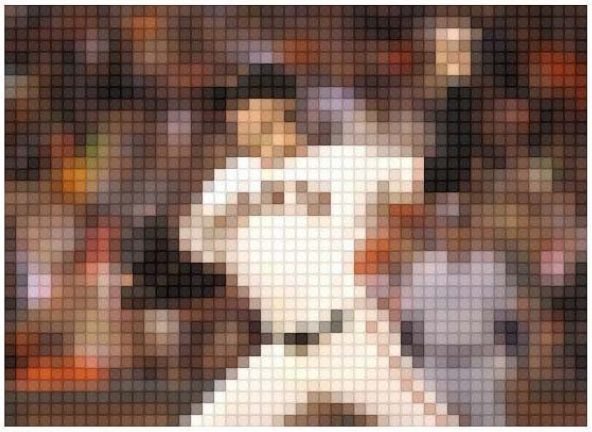
(D)(***n, d, G***)=(***7, 21, 8***)



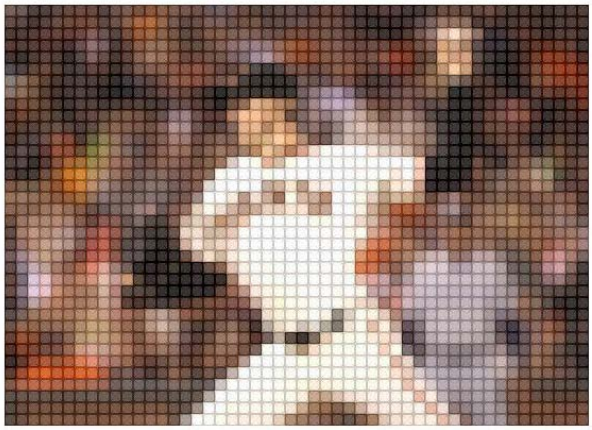
(E)(***n, d, G***)=(***7, 21, 16***)



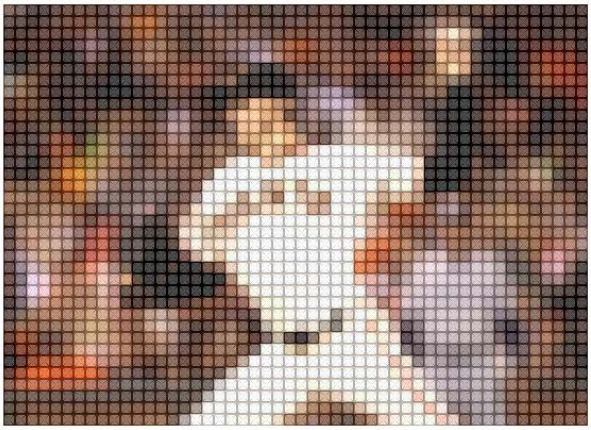
(F)(***n, d, G***)=(***7, 21, 24***)



(G)(***n, d, G***)=(***9, 21, 8***)

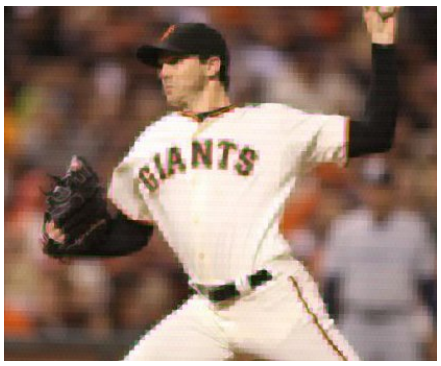


(H)(***n, d, G***)=(***9, 21, 16***)



(I)(***n, d, G***)=(***9, 21, 24***)

Fig.3 The critical data embedded image

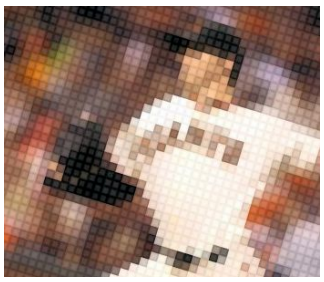



(A)(***n, d, G***)=(***5, 21, 8***)


(B)(***n, d, G***)=(***5, 21, 16***)


(C)**(*n, d, G*)=(*5, 21, 24*)**

(D)**(*n, d, G*)=(*7, 21, 8*)**



(E)**(*n, d, G*)=(*7, 21, 16*)**

(F)**(*n, d, G*)=(*7, 21, 24*)**

(G)**(*n, d, G*)=(*9, 21, 8*)**

(H)**(*n, d, G*)=(*5, 21, 16*)**

(I)**(*n, d, G*)=(*9, 21, 24*)**

Fig. 4 The partial area image photographed from host image

## V. CONCLUSION

The embedded image, being inserted critical data into host image, undergoes enormous distortion after camera or scanner extracts the image. Corresponsively, the embedded critical data will also be damaged seriously. This research therefore proposes HREH method. The experiment results demonstrate when $n \geqq 7$, HREH method can correctly extract the inserted critical data even though the PSNR of embedded images $I_c'$ are lower than 25 dB. However, the original content of $I_c$ is visually noticeable in $I_c'$.

## REFERENCES


[1] Agnew, G. B. Mullin, R. C. and Vanstone, S. A. "Improved Digital Signature Scheme Based on Discrete Exponentiation," Electronics Letter, Vol. 26, pp.1024-1025, 1990.

[2] Akl, S. "Digital Signatures: A Tutorial Survey." Computer, Feb. 1983.

[3] Cox, I., Kilian, J., Leighton, T., and Shamoon, T. "Secure spread spectrum watermarking for multimedia," IEEE Transactions On Image Processing, Vol. 6, No. 12, pp.1673-1687, Dec. 1997.

[4] Delaurentis, J. M. "A Further Weakness in the Common Modulus Protocol for the RSA Cryptosystem," Cryptologia, Vol. 8, No. 3, pp.253-259, 1984.

[5] Diffe, W., and Hellam, M. "Privacy and Authentication: An Introduction to Cryptography." Proceeding of the IEEE, Mar. 1979.

[6] Diffie, W. and Hellman, M. E. "New Directions in Cryptography," IEEE Transactions on Information Theory, Vol. IT-22, No. 6, pp.644-645, Nov. 1976.

[7] Greiff, W. R., and Ponte, J. M. "The Maximum Entropy Approach and Probabilistic IR Models," ACM Transactions on Information Systems(TOIS), Vol. 18, pp.246-287, Jul. 2000.

[8] ISO/IEC 10918-1:Informaton Technology –"Digital Compression and Coding of Continuous." – Tone Still Images: Feb. 1995.

[9] Kaliski, B., and Robshaw, M. " The Secure Use of RSA." CryptoBytes, Autumn 1995.

[10] Konfeder, L. M. "On the Signature Reblocking Public-key Cryptosystem," Comm. ACM Vol. 21, p.179, 1978.

[11] Petitcolas, F. A. P., Anderson, R. J., and Kuhn, M. G. "Information Hiding - a Survey," Proceedings of the IEEE, Vol. 87, No. 7, pp.1062-1078, Jul. 1999.

[12] Rabin, M. "Digitalized Signature." In Foundations of Secure Computation, DeMillo, R. Dobkin, D. Jones, A. and Liption, R. eds., New York: Academic Press, 1978.

[13] Rivest, R., Shamir, A. and Adleman, L. "A Method for Obtaining Digital Signatures and Pulic-Key Cryptosystems," Communications of the ACM, Vol. 21, No. 2, pp.120-126, Feb. 1978.

[14] Shimada, M. and Tanaka, K. "Blocking Method for RSA Cryptosystem Without Expanding Cipher Length," Electronics Letter, Vol. 25, No. 12, pp.773-774, Jun. 1989.

[15] Stinson, D., Cryptography Theory and Practice, CRC Press, Boca Raton, 1995.

[16] Wiener, M. J. "Cryptanalysis of Short RSA Secret Exponents," IEEE Trans. On Information Theory, Vol.IT-36, pp.553-558, 1990.

[17] Wolfgang, R., and Delp, E."Fragile Watermarking Using the VW2D Watermark," Proceedings of the IS&T/SPIE Conference on Security and Watermarking of Multimedia Contents, San Jose, California, pp.204-213, Jan. 1999