# PKI based Semi-Fragile Watermark for Visual Content Authentication

Chamidu Atupelage, Koichi Harada, *Member, ACM*

*Abstract*— **Multimedia content owners always endure of copyright protection and ownership verification of their digital assets. Robust watermarking techniques are invented to defeat these problems. However evolution of the watermarks focused different security aspects of multimedia data such as content integrity, data authentication, etc. As a result, fragile watermarking was introduced which is contrary to robust watermarks. Evolution of the fragile watermarks derives a new prospective watermarking concept called semi-fragile watermark which overcomes several limitations of fragile watermarking technique such as fragility for trusted image processing manipulations. In this paper we propose a new PKI (Public Key Infrastructure) based semi-fragile watermarking technique to authenticate perceptible content (information perceived by Human Visual System) of the digital data (digital images). Additionally our definition localizes the compromised regions of the image. Proposed method inherits all security features in public key authentication system. On the other hand visual distortions are acceptable up to some extend in particular applications. Therefore distortions due to the watermark should be minimized. However large watermarks are more secure and it causes more distortions. In this paper we present sophisticated analysis of this tradeoff with analysis matrices (graphs & tables), which makes easier to optimize the security with minimal visual distortions.**

*Index Terms*— **Discrete cosine transformation, Elliptic curve digital signature algorithm, Image authentication, imperceptibility, JPEG, Public key cryptography.**

## I. Introduction

Integrity verification of digital data is having applicability in some e-commerce applications such as law, defense, journalism, and video conferencing etc, which are intended to show that no tampering has occurred during the transmission. Fragile watermark is readily altered or destroyed when the host image is modified through a linear or non-linear transformation. The sensitivity of the fragile watermark leads to their usage in image authentication, where watermark loss or alteration is taken as evidence that data has been tampered with, whereas the recovery of the information contained within the data is used to demonstrate origin of the multimedia data [1]. Therefore fragile watermarking is a promising approach for multimedia data authentication. In practice lossy compressed multimedia data is accepted as authentic and fragile watermarks are vulnerable for these compressions. Therefore the fragile watermark concept is altered to be robust against non-malicious attacks and renamed as semi-fragile watermarks.

The study of fragile and semi-fragile watermarking techniques together with information security concluded a set of features which should be incorporated into an effective semi-fragile watermarking scheme.

- *Sensitivity:* The watermark is sensitive to malicious manipulations in higher probability.
- *Robustness:* The watermark is withstood to non-malicious manipulations.
- *Localizing fiddle region:* Watermarking scheme should identify the misrepresented areas in the original image.
- *HVS transparency:* Watermark should be imperceptible to human visual system.
- *Visual content authentication:* To keep low computational complexity watermark should authenticate only the perceived (selected) digital data, instead processing on all digital information.
- *High security and availability:* Determination of the authentic credentials should not be possible and watermarking system should be publicly available.

In this paper we propose a new semi-fragile watermarking approach which can be use to authenticate visual content of the images. New technique is effective, as it reaches all above define requirements. More precisely proposed system is robust against to JPEG and sensitive to considerable manipulations such as cropping, content replacements, etc. in this definition we have reduce the computational complexity by authenticating only the perceptible portion in DCT domain. Our security framework has inherited all routines of PKI based digital signature algorithm. Therefore our definition is open and available to the public. Our experimental results show that the visual artifacts due to watermark embedding are invisible for natural eye. Even though having no perceptible artifact, we quantitatively evaluated the distortions and proposed possible watermark embedding routines.

In section 2, we will discuss the limitations of existing fragile watermarking definitions and defeating techniques. Section 3 gives a architecture of the proposed technique. Design issues of visual content retrieval and security infrastructure is presented in section 4. Section 5 details the watermark embedding and verification routines. Section 6 presents the experimental results and outcomes. Analytical details are presented in section 7 and finally we conclude all our work in section 8.

Manuscript received July 29, 2008; revised Aug 08, 2008.

Chamidu Atupelage, Department of Information Engineering, Hiroshima University, 1-7-1 Kagamiyama, Higashi-Hiroshima, 739-8521, JAPAN; e-mail: atupelage@hiroshima-u.ac.jp.

Koichi Harada, Department of Information Engineering, Hiroshima University, 1-7-1 Kagamiyama, Higashi-Hiroshima, 739-8521, JAPAN; e-mail: harada@mis.hiroshima-u.ac.jp.

## II. Defeating the Limitation of Available Techniques

Most of the proposed semi-fragile watermarking techniques use other image or a pseudo random bit sequence as a watermark and authenticator verifies it [2]. However these techniques are vulnerable if attacker alternate visual information without damaging to the watermark (authenticator verifies as watermark is authentic). In our approach watermark is the digital signature of the visual content, thus it guarantees the authenticity of the visual content. Some other applications try to protect all visual information and it increases the computational cost [2], [3]. In our technique we choose low frequency components in DCT block as visual content and it reduces the system overhead and computational cost. Some techniques use visible watermarks [4]; it may cause hiding important information in the source image. Another invisible fragile watermarking technique has proposed in [5], but it provides awful visual artifacts in some particular situations such as when the image consist of lot of characters or lines. In our approach we use a pre-selected portion in low frequency region for watermark embedding, and experimental results show no extra visual artifacts. Some digital data transmission applications may interest to identifying the compromised area in a particular image or video segment. However most of the semi-fragile watermarking schemes do not attempt to identify the tampered regions [4]. However our proposed technique authenticates compromised regions in an image. The outcome of the algorithm is an authenticator matrix; 0s represents the compromised blocks and 1s represent of authenticated blocks. According to the definition of the authentication the attacker might not be interested to destroy the watermark. Conversely attacker tries to derive the secret credentials. Some watermark authentication scheme share the common credentials (key, watermark, etc) within sender and receiver [6], however these definitions are vulnerable to secret credential prediction attacks. Our definition strongly prevents these attacks, because our security infrastructure has completely inherited the PKI authentication techniques.

## III. SYSTEM OVERVIEW

DCT domain is desired to be used to retrieve visual content and embed the watermark, thus the verification is also carried out in the same domain. Watermark (digital signature) is generated by using the perceptible content in DCT block and it is embeded into the image by choosing the invariant properties in the JPEG system. JPEG lossy compression algorithm consists of three major steps which are DCT computation of luminance and chrominance channels, quantization and finally variable length code assignment [7], [13]. In Fig.1 we graphically illustrate behavior of our definition in standard JPEG system.

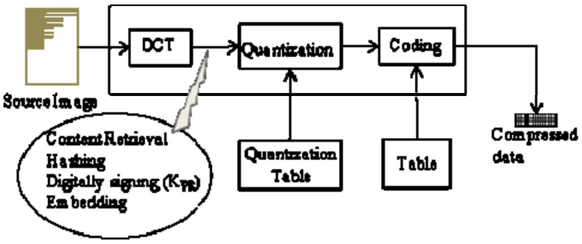


**Fig. 1: Watermark insertion in DCT domain**

## IV. DESIGN ISSUE

There are two major aspects in our watermarking techniques. One is selecting appropriate visual content for authentication and the other is selecting proper cryptographic algorithm which provides higher security for minimal key lengths. This section describes the routines of visual content retrieval and the security infrastructure of the application.

### A. *Visual Content Retrieval*

Multimedia data is always being changed, invisible manipulations are taken as authentic and it is highly subjective measure. For example, authenticator that accepts lossy compression up to allowable level of quality loss and rejects other manipulations. It is apparent the recipient may be satisfied by authenticating only the perceptible content of the digital data as well as authenticating less number of data certainly reduces the computational cost and complexity. In our proposal we use DCT domain as a media to retrieve the visual content of the image, because it is the most widely used technique in lossy compressions. Perceptible information perceived in the DCT block has been quantitatively benched marked in [8]; 60% of the visual structural information conserve in the DC coefficient and 70% of visual information is drawn to DC and first two AC coefficients (according to zigzag scanning and 8×8 pixels block). Likewise accounting more AC coefficients we can increase the perception level in small quantities [8].

Therefore we can summarize that authenticating DC coefficient guarantees the authenticity of the 60 % of visual information. Likewise accounting first 2 AC coefficients, the system guarantees authenticity of the 70% visual information.

### B. *Security Infrastructure*

In PKI, key length is an important factor and it is proportional to the signature length. In our definition digital signature is used as watermark. However larger watermarks require larger embedding space and intuitively it increases the visual artifacts.

ECDSA (Elliptic Curve Digital Signature Algorithm) is the elliptic curve analog of well known digital signature crypto system. ECDSA is based on elliptic curve discrete logarithm problem (ECDLP) and ECDLP is significantly more difficult than IFP (Integer Factorization Problem) and DLP (Discrete Logarithm Problem) [9]. Therefore ECC (Elliptic Curve Cryptography) provides the highest strength-per-bit of any cryptosystem known today [10]. Therefore we use ECDSA in our definition to overcome the space limitation constrain.

Security infrastructure of our system follows the basic PKI authentication principles; sender has owner private and public key pair. Public key is distributed at some key distribution server with proper digital certificate. Private

Key is use to generate the digital signature and corresponding public key verifies the signature of the visual content.

## V. WATERMARKING & AUTHENTICATING

Digital watermarking routines can be classified to watermark generation and embedding, authenticating routines are classified to watermark extraction and verification. We will explain those routines by using following notations in next two consequence sub sections.

| | |
|---|---|
| $I_s$ | : Source image. |
| $I_w$ | : Watermarked image. |
| $F_{RET}\ ()$ | : Retrieving the visual content. |
| $F_{INS}\ ()$ | : Embedding the signature. |
| $F_{EXT}\ ()$ | : Extracting the signature. |
| $H_{MD5}()$ | : Hashing function. |
| $K_{PR}$ | : Signer's private key. |
| $K_{PU}$ | : Signer's public key. |
| $s$ | : Signature. |
| $md$ | : Message digest. |
| $vc$ | : Visual content. |
| $ECDSA()$ | : ECDSA function. |

### A. *Signature Generation & Embedding*

At first the system preprocesses the retrieved visual content. (Organize in predefined order). Hashing function generates the message digest of the visual content and we call to the signature generating algorithm (ECDSA) together with signer's private key. Then signature is embedded into the image in DCT domain. The whole procedure can be represented as follows.

1. $Vc \leftarrow F_{RET}\ (I_s)$
2. $md \leftarrow H_{MD5}\ (vc)$
3. $s \leftarrow ECDSA(K_{PR}, md)$
4. $I_w \leftarrow F_{INS}\ (I_s\ ,s)$

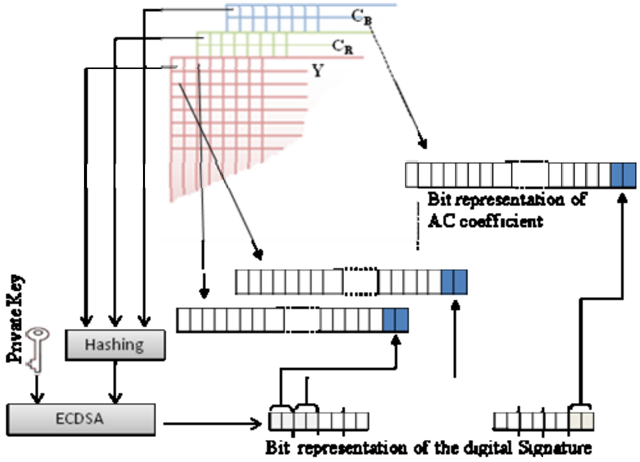


**Fig. 2: Signature generation and embedding (bit representation)**

In Fig.3 we represent the above routines graphically. For clear representation we assume the system authenticates only DC coefficients. For easy understanding we present the values of AC coefficients and the signature as bit representation. Generated signature is divided into two bits pieces and each piece replaces the last two significant bits of the selected AC coefficients.

### B. *Signature Extraction & Verification*

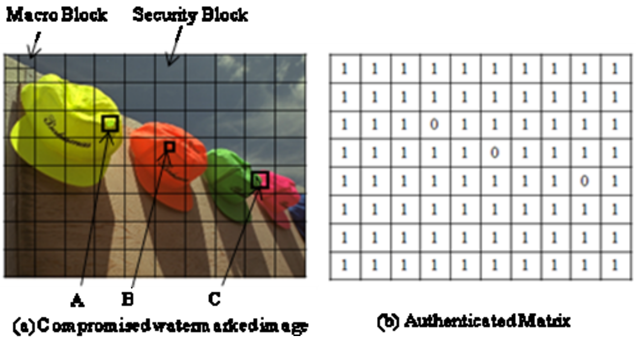
Visual content is retrieved from the DC coefficients and will be sent to the hashing function to get the message digest. Embedded signature is retrieved from the AC coefficients. Signature verification anticipates three parameters; message-digest, original signature and public key, then it will return the validity of the signature. The whole process can be illustrated as algorithmic format in four steps.

1. $vc \leftarrow F_{RET}\ (I_w)$
2. $s \leftarrow F_{EXT}\ (I_w)$
3. $md \leftarrow H_{MD5}\ (vc)$
4. $ECDSA(K_{PU}, md, s) \rightarrow Acceptance$

The graphical representation of this technique is almost similar to the Fig2. $F_{EXT}$ is opposite to $F_{INS}$ and ECDSA takes three parameters for signature verification.

## VI. EXPERIMENTAL RESULTS

We conducted an experiment to evaluate the proposed semi-fragile watermarking technique. For the experiment we use 160 × 128 pixels bitmap image (because we suppose to present the experimented image in exact dimentions), then we generate and embed the watermark according to our definitions. For JPEG compression we chose standard macro block (8 × 8 pixels). To obtain adequate space for watermark we consider 4 macro blocks together, then the size of the security block is become 16 × 16 pixels (refer Fig.4). We retrieved the DC value and first 2 AC values as visual content (70 % of the visual information [8]) and digital signature is generated for the retrieved DCs and ACs by using the private key (160 bits length). Watermark (digital signature) is embedded to the DCT values following the rule 3 in Table 1. Then we performed three alternations in the watermarked image (Fig.3 (a)). Locations of the A, B and C are presented as zero in authenticated matrix (Fig.3 (b)).



| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

(a) Compromised watermarked image (b) Authenticated Matrix

**Fig. 3: Compromised Watermarked image and corresponding authentication matrix**

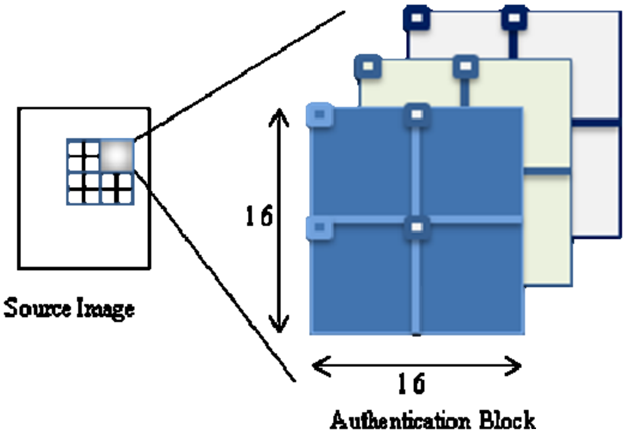


**Fig. 4: Authentication block represent in source image**

## VII. WATERMARKING PARAMETERS AND ITS' ANALYSIS

In this work our desired application domain is secure digital data transmission environment, with customizable parameters (security level, visual quality and compression ratio). Parameter optimization is carried out according to end user requirement. Initially we started the research focusing the images and our consequent research will be discussing about the applicability to the videos. However here we use video conferencing application as the instance of our discussion.

In some particular video conferencing application, one expects low level of security with high visual quality when the conference is carry out in the intranet; contrarily one anticipates high level of security and low visual quality via internet. In our definition visual quality, security strength and compression ratio are considered as interdependent parameters. In this section we will discuss and quantitatively examine these consequences.

In order to our definition, digital signature is considered as the watermarked. At current the accepted ECC key length is 160 bits and the digital signature is 336 bits long [11]. In our application 8×8 pixels block (DCT macro block) is considered as the smallest block. Considering four blocks together we can acquire enough space to embed the signature. In this way the minimal security block size becomes 16 × 16 pixels (Refer Fig.4). More precisely we generate one digital signature for visual content of 4 macro blocks (16 × 16 pixels). Then signature is divided in to 4×3 parts (Macro blocks × Color channels) and embed each part in a preselected region (refer Fig.2) in each DCT macro block. In this way we use first 14 AC coefficients in one macro block (in order to zigzag scanning) to embed the signature. To perceive these values from quantization, we mark first 15 values in quantization matrix as 1 (this process directly affect to the compression ratio and later it will be analyzed). Contrary if we increase number of macro blocks up to 9 (authentication block become 24 × 24), then 6 AC coefficients provide an adequate space, thus the alternation of quantization matrix become less and it gives better compression ratio.

Let's formulate the above scenario. Take signature length as $l$ which is proportional to the multiplication of "*number of DCT blocks*" , "*number of AC coefficients in one DCT block*" $(y)$ and "*number of bits used in one AC coefficients*" . (In Fig.2 you can see 2 LSBs are used). Intuitively we know embedding space should be larger or equal to the signature length $l$ .

(1)

It is reasonable to assume $l$ is constant, because it should follows current security definition (336 bits). Then other three parameters are varied according to the quality vs. compression ratio. Following sub sections present experimental analysis over these parameters.

### A. *Authentic Security Block Size*

If we increase while keeping as constant then it decreases the . Therefore we can reduce the number of alternations in quantization table and it increases the compression ratio of the resulted image. We define . Actually alternations of the quantization matrix should be minimized and in Fig.6 we present this degradation quantitatively.

Fig. 5 shows the actual sizes of possible security blocks. However if an image consist of lot of texts, then the smaller block is useful, contrary if image consist of a large items (occupied over 625 pixels), better to choose larger blocks.

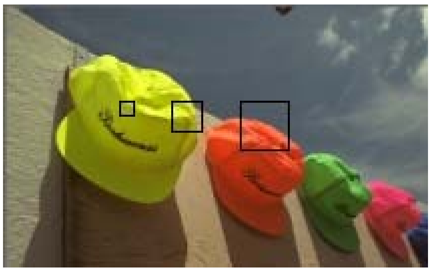


**Fig.5: Authentication blocks in different sizes. From left 8×8, 16×16, 24×24.**

### B. *Number of selected AC coefficients* $(y)$

To preserve the selected AC coefficients from the quantization, quantization matrix is altered. We should mark first number of coefficients (zigzag scanning as 1 DC and number of ACs). Since this alternation reduces the compression ratio, it is important to carry out an experiment to analysis the effect of quantization table alternation against compression ratio.

In this work we take 29 bitmap images and compute the average compression ratio for a particular quality factor. Varying $y$ as 1, 2, .. , 10 we can check how quantization table alternation effects on compression ratios. Same process is carryout for several quality factors as 10, 20, .. , 100. In Fig.6 we have plotted the results of the experiment as compression ratio against to the number of marked coefficients.

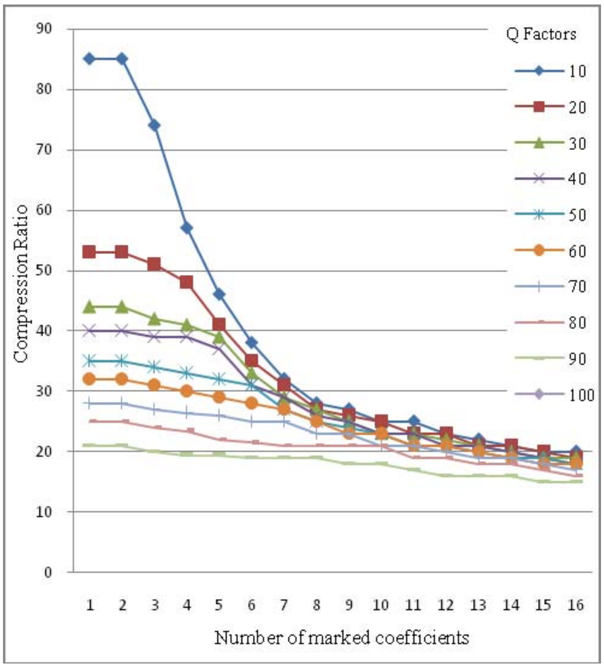


**Fig.6: Graph of compression ratios against number of marked coefficients in quantization table.**

Intuitively we know low quality factors presents awful visual quality as well as higher compression ratios, thus quality factors in between 20 to 60 can be considered as providing adequate compression ratio with acceptable quality. We can see in Fig.6, the effect of quantization alternation is very low for higher quality factors. Therefore we can use Fig.3 as reference matrix to decide appropriate quality factor with  for particular application.

### C. *Number of LSBs in one AC coefficient* ($s$)

In Fig.2 two LSBs of AC coefficients are taken for watermark embedding. Increasing this number we can reduce the $y$ as well as $b$ according to application requirement**.** However this operation causes degradation in visual quality. Therefore we carried out an experiment to measure the quality degradation against to variation of $s$ .

In this experiment we use objective quality matrix called SSIM (Structural Similarity Index Matrix) [12]. In JPEG DCT domain, we have three channels (Y, $C_b$ and $C_r$) and all three channels are used for watermark embedding. In HVS each channels having different sensitivity (Ex: $C_b$ is less visible compared to $C_r$). Further $s$ is equal to the total number of bits available in all color channels and in according to (1), increases of $s$ , and decreases $y$ . Therefore considering the HVS properties and graph in Fig.6 we formed 5 rules (Table 1), which represents number of LSBs in each channel and corresponding number of altered ACs. Then we perform an experiment referring Table 1 as watermark embedding definitions.

**Table 1: Five rules of watermark embedding**

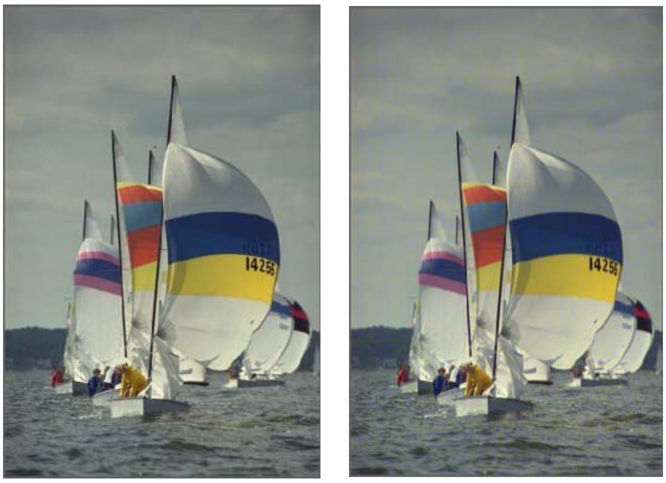
| Rule | Y<br>(LSBs) | $C_b$<br>(LSBs) | $C_r$<br>(LSBs) | # of altered<br>ACs ($s$) |
|---|---|---|---|---|
| 1 | 2 | 2 | 2 | 14 |
| 2 | 3 | 3 | 3 | 10 |
| 3 | 2 | 6 | 4 | 7 |
| 4 | 2 | 4 | 6 | 7 |
| 5 | 0 | 5 | 5 | 9 |

**Table 2: SSIM values for RGB components of JPEG compressed and Watermarked images**

| Rule | R | G | B |
|---|---|---|---|
| 1 | 0.9843 | 0.9928 | 0.9772 |
| 2 | 0.9409 | 0.976 | 0.9186 |
| 3 | 0.8654 | 0.8231 | 0.334 |
| 4 | 0.4244 | 0.6273 | 0.7219 |
| 5 | 0.6774 | 0.7714 | 0.5164 |

In the experiment we took 29 bitmap images and embed watermark in particular bits of AC coefficients as defined in Table 1 and the resulted images can be considered as watermarked JPEG images. Again we process JPEG operation (without watermarking) for same bitmap images. Then we compare structural similarity distortion due to watermark embedding in between watermarked and non watermarked images. In Table 2, we present the SSIM values for RGB color components for each definition. In Table 2, rule 1 definition shows less distortions, contrary it give low compression ratio due to large  . Rule 2 shows better performance in compression ratio than rule 1, though having more structural distortions. Rule 3 is having good compression ratio, however blue component shows more distortions than red and green components. According to HVS definitions blue channel is less visible under normal condition, thus we can ignore the distortion in blue channel. Furthermore Fig.6 is an evident that distortions over blue component are less visible. Likewise we can choose appropriate embedding rule in order to the end user requirement.

Fig.7 presents JPEG compressed image and watermarked (following Rule 3) image. Even though having higher structural destruction in blue component (low SSIM value), the visual artifacts has not seriously distorted the perceptual information.



(a) JPEG Compressed

(b) Watermarked (Rule 3)+ JPEG

SSIM (R = 0.9228 , G= 0.9046, B= 0.3950)

**Fig.7: JPEG Compressed and Watermarked Compressed image in Type 3 embedding.**

## VIII. Conclusion and Future work

Literature studies define setoff features which should be in an effective semi-fragile watermarking scheme. In our research we defined a semi-fragile watermarking scheme which reaches all these requirements. Furthermore we presented precise analysis of the watermarked embedding system parameters and its variations. Proposed matrices (graph and table) are useful for optimize the application parameters according to end user requirement.

Our future directions include: 1) applying same directions for MPEG and JPEG2000 definitions, 2) More detail evaluation of the performance of the scheme, 3) More precise analysis of security attacks and survivability of the watermark.

## References

[1] M. Yeung and F. Mintzer, "Invisible watermarking for image verification", in Journal of Electronic Imaging, July 1998, vol. 3, pp. 578-591.

[2] Min Wu and Bede Liu, "Watermarking for Image Authentication", in ICIP conference on Image Processing, Chicago, USA, Oct 1998, vol. 2, pp. 437-441.

[3] Ming-Shing Hsieh and Din-Chang Tseng, "Perceptual Digital Watermarking for Image Authentication in Electronic Commerce", in Kluwer Academic Publishers Norwell, MA, USA, 2004, vol. 4, pp. 157 - 170. 1389-5753.

[4] Hyuncheol Park and Kwangjo, "Kim Visible Watermarking using Verifiable Digital Seal Image" in SCIS 2001, Oiso, Japan, Jan 2001.

[5] Eugene T. Lin, Christine I. Podilchuk and Edward J. Delp, "Detection of image alterations using semi-fragile watermarks" in SPIE proceedings series, 2000. vol. 3971, pp. 152-163, 0-8194-3589-9.

[6] Raymond B. Wolfgang and Edward J. Delp, ".Fragile watermarking using the VW2D watermark" in SPIE, 1999. vol. 3657, pp. 204-213.

[7] Wallace, G.K., "The JPEG still picture compression standard", in IEEE Transactions on Consumer Electronics, Feb 1992, vol. 38, pp. xviii - xxxiv.

[8] L. Weng and B. Preneel, "On encryption and authentication of the DC DCT coefficient", in Proc. Of International Conference on Signal Processing and Multimedia Applications, 2007.

[9] Certicom, 2000, "The Elliptic Curve Cryptosystem". Available: http://www.comms.scitech.susx.ac.uk/fft/crypto/EccWhite3.pdf.

[10] Certicom, "Remarks on The Security of the Elliptic Curve Cryptosystem". Available: http://www.comms.scitech.susx.ac.uk/fft/crypto/EccWhite3.pdf.

[11] Keylength.com. Available: Availablehttp://www.keylength.com/en/compare/.

[12] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh and Eero P. Simoncelli, "Image Quality Assessment: From Error Visibility to Structural Similarit" in IEEE Transactions on Image Processing, Apr. 2004, vol. 13.

[13] The Transform and Data Compression handbook. K. R. Rao, P. C. Yip. CRC Press (2001) – 849336929.