

New Images Watermarking Scheme Based on Singular Value Decomposition

Abdelhamid Benhocine and Lamri Laouamer

Information System Department
College of Business and Economics
Al Qassim University, Al Melaidad 51452, KSA
abdelhamid.benhocine@yahoo.fr; lamri.laouamer@yahoo.fr

Laurent Nana

Lab-STICC (UMR CNRS 6285)
European University of Brittany
University of Brest
20 avenue Victor Le Gorgeu, BP 817 - CS 93837, 29238 Brest cedex, France
Laurent.Nana@univ-brest.fr

Anca Christine Pascu

European University of Brittany
University of Brest
CS 93837, 29238 Brest cedex, France
Anca.Pascu@univ-brest.fr

Received July 2012; revised November 2012

ABSTRACT. Watermarking technology aims to solve the multimedia copyright protection problems in particular on computer networks such as Internet. The technology must show proof of ownership with a very high degree of confidence and demonstrate a strong resistance against the hackers distortion like the noising, compression, filtering. The approach proposed in this paper consists of a new image watermark embedding / extraction algorithm based on the singular value decomposition SVD. The results obtained show that our algorithm resists perfectly against the hackers attacks which demonstrates the robustness of the proposed approach.

Keywords: Watermarking, Singular Value Decomposition, hacker attacks

1. Introduction. The images form the majority of all digital documents processed and exchanged in the Internet world. It became extremely simple to reproduce perfectly any medium. In the case of digital multimedia like sound, image and video, research is oriented to a technical resolution for this problem by inserting a watermark in the medium to identify the ownership property. This watermark embedding mechanism should meet at least two conditions: the watermark must be imperceptible (the human eye is not able to distinguish between a watermarked and unwatermarked image) and robust (the watermark must resist to all kind of voluntary or involuntary attacks). The extraction should be blind, that is to say that to extract the watermark, we do not need the original image.

Many watermarking algorithms have been proposed seeking to optimize the imperceptibility/robustness compromise. These include the watermarking techniques in spatial domain, such as those proposed in [1, 2, 3], which operate directly on pixels, and the

watermarking techniques in frequency domain, such as Fast Fourier Transform FFT, Discrete Cosine Transform DCT, Discrete Wavelets Transform [4, 5, 6], etc. Many recent images watermarking techniques are inspired from the usual methods of encoding and compression algorithms. The singular value decomposition (SVD) is one example of those techniques. This technique has already proven in the field of compression [7, 8] by providing a compression quality ratio close to the JPEG technique. Regarding the watermarking using SVD, only a small number of publications exist. Among the solutions found we can mention: the work of [9] who applied the watermark just in the matrix U(right eigenvectors matrix), [10] proposed to add the mark for just the singular value matrix S using a variable embedding weight, [11] and [12] proposed a hybrid scheme of SVD-DCT based watermarking to improve the robustness. A more sophisticated approach is proposed in [13]. It consists in embedding the watermark in the matrix V (left eigenvectors matrix) by adding to the matrix V of the original image, the matrix V of the watermark. All these algorithms are non-blind type.

The author in [14] discusses the work of [15] and shows that the work presented cannot be used for proving protecting rightful ownership because of a high probability of false positive answers and is therefore of no practical value. Similarly, the author of [16] shows the same problem for the work of [17].

In this paper, we present a new blind algorithm that inserts the watermark in all the SVD components (U, S and V) matrix of singular values. In the same time, we present our proposed approach for achieving watermarking in the different components of the SVD decomposition. The fundamentals of SVD are first recalled, the method is then detailed and the experimental results discussed. Our approach shows more resistance against several attacks like rotation, noising, Filtering, Compression, etc ... compared to the proposed approaches using SVD [15, 17].

When embedding a watermark w into a host image i , we use one of the two methods: either $i_w = i + t w$, where t is close to 0 or $i_w = (1-t)w+t, i$, where t is close to 1. However, we prefer using the second embedding function (linear interpolation) [20], since in this case, i and w play the same role allowing choosing in the tests any t in $]0,1[$, so that visualization is more accurate in the sense that if t is close to 1 (resp. 0) then i_w is close to i (resp. w).

For robustness, the images are watermarked after transforming them into other spaces like YUV, DCT or DWT. The watermark is often embedded into Y component (YUV), Blue channel (RGB), high frequencies (DCT, DWT) and singular values matrix (SVD).

Suppose I is any RGB-color image. Without loss of generality, we use the red component i of I . Many papers [15, 18, 19] adopt the following embedding process. The image i is transformed using one of YUV, DCT, DWT getting a matrix which is then decomposed using SVD into the triple $[U_i, S_i, V_i]$. The same transformation is applied to the watermark W and leads to the triple $[U_w, S_w, V_w]$. Note that the watermark may be S_w itself. Now, let $S_w^* = S_i + t S_w$, where t is a parameter $\neq 0$, so that $i_w = U_i S_w^* V_i^T$ is the red component of the watermarked image I_w (Fig.1).

The extraction process is the inverse embedding process.

There are many techniques to measure the robustness of any watermarking scheme. The most used are the Mean Square Error (MSE)[19], the fractal dimensions [20], correlation coefficient [19]. In reality, these techniques measure the degree of resemblance between the original watermark and that extracted in order to decide the quality of the embedding/extraction process.

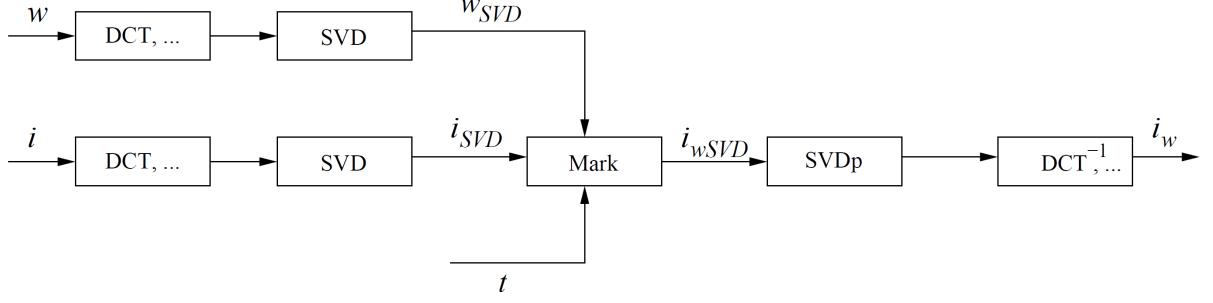


FIGURE 1. The most used embedding process in transformed domain

2. Proposed method. We present in this paper, a blind algorithm that inserts by linear interpolation the watermark in the SVD matrices U, S and V of each image component (red, green or blue, say red hereafter).

2.1. Convention. Fig.2 shows the way of designing mark and unmark functions in the watermarking schemes. In Fig.2(a), we use linear interpolation to embed the watermark w in i getting $i_w = (1-t)w + ti$. This implies the inverse process in Fig.2(b), that is $1/t$ $i_w - (1-t)/t w$.

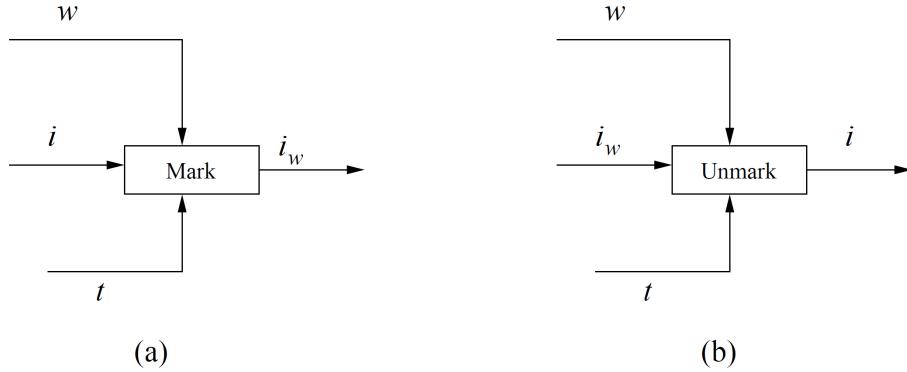


FIGURE 2. Convention for (a) mark and (b) unmark functions

2.2. Watermark embedding. Let us consider RGB-color images I and W , with size $128 \times 128 \times 3$ and recall that without loss of generality i and w are respectively the red components of I and W . The embedding watermark scheme is shown in Fig.3.

SVD calculates the three matrices, say U, S, V while 'SV Dp' (considered as the inverse of SVD) calculates the product USV^T . As in Fig.2, 'Mark' is the interpolation process: we mark i_{SVD} using w_{SVD} .

Suppose i and w are respectively decomposed into the triples $i_{SVD} = [U_i, S_i, V_i]$ and $w_{SVD} = [U_w, S_w, V_w]$. Three ways (M1-3) may be considered when watermarking i with SVD using w as a watermark. Let $w_{wsvd} = [U_{iw}, S_{iw}, V_{iw}]$.

The watermarked image i_w is then obtained as follows: $i_w = SV Dp(i_{wsvd}) = U_i w S_i w V_{iw}^T$. Precisely, let us calculate i_w in the three cases (M1-3):

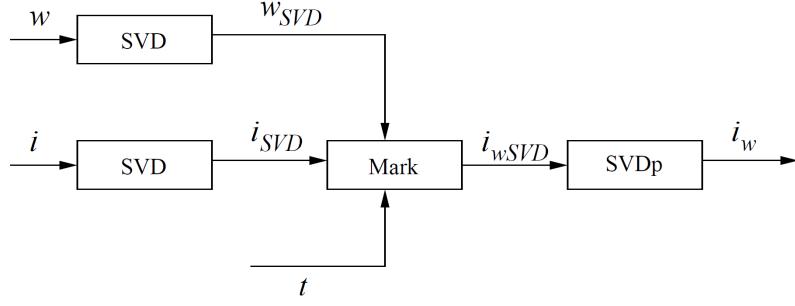


FIGURE 3. Watermarking Embedding Scheme

| (M1) We mark U, S, V | (M2) We mark U, V | (M3) We mark S |
|----------------------------|----------------------------|----------------------------|
| $U_{iw} = (1-t)U_w + tU_i$ | $U_{iw} = (1-t)U_w + tU_i$ | $U_{iw} = U_i$ |
| $S_{iw} = (1-t)S_w + tS_i$ | $S_{iw} = S_i$ | $S_{iw} = (1-t)S_w + tS_i$ |
| $V_{iw} = (1-t)V_w + tV_i$ | $V_{iw} = (1-t)V_w + tV_i$ | $V_{iw} = V_i$ |

$$\begin{aligned}
 (\mathbf{W1}) \quad i_w &= ((1-t)U_w + tU_i)((1-t)S_w + tS_i)((1-t)V_w^T + tV_i^T) \\
 &= (1-t)^3 U_w S_w V_w^T + t^3 U_i S_i V_i^T + \alpha_1 \\
 &= (1-t)^3 w + t^3 i + \alpha_1
 \end{aligned} \tag{1}$$

Where $\alpha_1 = (1-t)t[(1-t)U_i S_w V_w^T + (1-t)U_w S_i V_i^T + (1-t)U_w S_w V_i^T + tU_w S_i V_i^T + tU_i S_w V_i^T + tU_i S_i V_w^T]$

$$(\mathbf{W2}) \quad i_w = ((1-t)U_w + tU_i)S_i((1-t)V_w^T + tV_i^T) = t^2 i + \alpha_2 \tag{2}$$

Where $\alpha_2 = (1-t)^2 U_w S_i V_w^T + t(1-t)U_w S_i V_i^T + t(1-t)U_i S_i V_w^T$

$$(\mathbf{W3}) \quad i_w = U_i((1-t)S_w + tS_i)V_i^T = tU_i S_i V_i^T + (1-t)U_i S_w V_i^T = ti + \alpha_3 \tag{3}$$

Where $\alpha_3 = (1-t)U_i S_w V_i^T$

Note that the only unknown in α_k , $k=1,2,3$ is t . If t is close to 1, then α_k is close to 0, and i_w is close to i . So, chose t close to 1 to get an invisible watermark in i_w . Fig.4 shows watermarked images for some values of t , respectively when watermarking (1) U, S and V, (2) U and V and (3) only S.

2.3. Watermark Extraction. Fig.5 shows the watermark extraction scheme. As seen in watermarking embedding, 'Mark' is the interpolation process (here, we mark by linear interpolation w_{SVD} using i_{wSVD}) and 'Unmark' is the inverse process (we 'extract the mark' i_{waSVD} from w_{SVD}).

Let $i_{wasvd} = [U_{iwa}, S_{iwa}, V_{iwa}]$, $w_{isvd} = [U_{wi}, S_{wi}, V_{wi}]$ and $w_{asvd} = [U_{wa}, S_{wa}, V_{wa}]$ where $U_{wa} = 1/tU_{wi} - (1-t)/tU_{iwa}$, $S_{wa} = 1/tS_{wi} - (1-t)/tS_{iwa}$ and $V_{wa}^T = 1/tV_{wi}^T - (1-t)/tV_{iwa}^T$.

We have $w_a = U_{wa} S_{wa} V_{wa}^T$, then $t^3 w_a = (U_{wi} - (1-t)U_{iwa})(S_{wi} - (1-t)S_{iwa})(V_{wi}^T - (1-t)V_{iwa}^T)$.

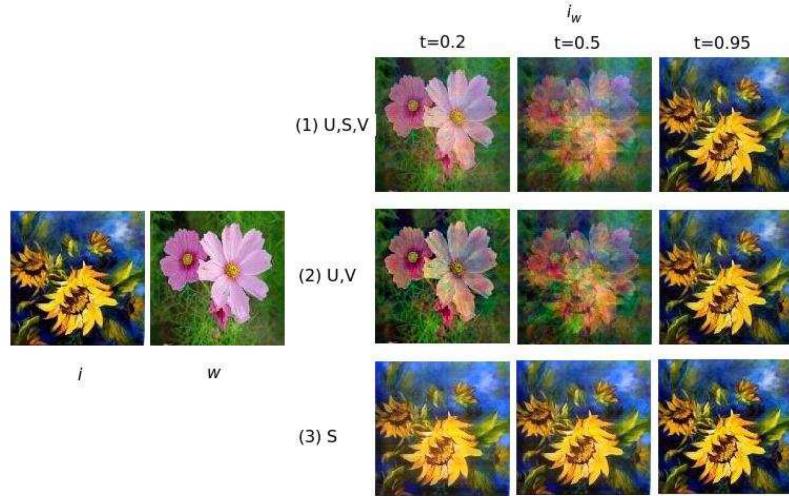


FIGURE 4. Embedding a watermark with SVD method

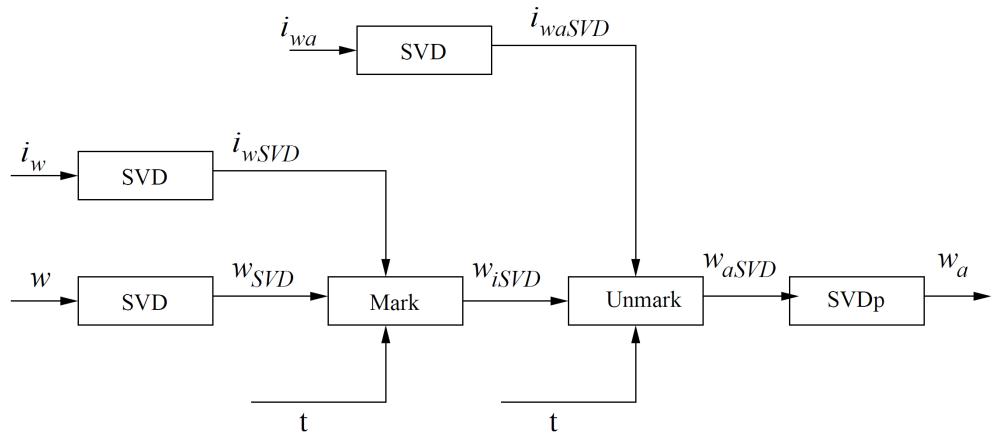


FIGURE 5. Watermarking Extraction Scheme

Three ways (MP1-3) are considered when extracting the watermark w .

| (MP1) We mark U, S, V | (MP2) We mark U, V | (MP3) We mark S |
|-------------------------------|-------------------------------|-------------------------------|
| $U_{wi} = (1-t)U_{iw} + tU_w$ | $U_{wi} = (1-t)U_{iw} + tU_w$ | $U_{wi} = U_w$ |
| $S_{wi} = (1-t)S_{iw} + tS_w$ | $S_{wi} = S_w$ | $S_{wi} = (1-t)S_{iw} + tS_w$ |
| $V_{wi} = (1-t)V_{iw} + tV_w$ | $V_{wi} = (1-t)V_{iw} + tV_w$ | $V_{wi} = V_w$ |

We get in each case (MP1-3),

$$\begin{aligned}
 (\mathbf{E_1}) \quad t^3 w_a &= ((1-t)U_{iw} + tU_w - (1-t)U_{iwa})((1-t)S_{iw} + tS_w - (1-t)S_{iwa})((1-t)V_{iw}^T \\
 &\quad + tV_W^T - (1-t)V_{iwa}^T) \\
 &= ((1-t)(U_{iw} - U_{iwa}) + tU_w)((1-t)(S_{iw} - S_{iwa})((1-t)(V_{iw}^T - V_{iwa}^T) + tV_W^T)
 \end{aligned} \tag{4}$$

Since U_{iwa} , S_{iwa} and V_{iwa}^T are constant matrices relatively to t , and U_{iwa} , S_{iwa} and V_{iwa}^T are calculated using linearly interpolation ($U_{iw} = (1-t)U_w + tU_i$ and analogously

for S and V), then if t is close to 1, then w_a is close to $U_w V_w^T = w$ and analogously for (E2-3).

$$\begin{aligned} \text{(E2)} \quad t^3 w_a &= ((1-t)U_{iw} + tU_w - (1-t)U_{iwa})(S_w - (1-t)S_{iwa})((1-t)V_{iw}^T + tV_w^T - (1-t)V_{iwa}^T) \\ &= ((1-t)(U_{iw} - U_{iwa}) + tU_w)((1-t)(S_w - S_{iwa}) + tS_w)((1-t)(V_{iw}^T V_{iwa}^T) + tV_w^T) \end{aligned} \quad (5)$$

$$\text{(E3)} \quad t^3 w_a = (U_w - (1-t)U_{iwa})((1-t)S_{iw} + tS_w - (1-t)S_{iwa})(V_w^T - (1-t)V_{iwa}^T) \quad (6)$$

3. Test. In the tests, we have chosen i and w of size $128 \times 128 \times 3$ (RGB images) shown in Fig.6, and $t=0.2, 0.5, 0.95$ and we have proceeded to the three ways of watermarking (M1-3). The Stirmark Benchmark [21] is used to attack the watermarked images (here the attacks are in this order: Rotation 45, Convolution 2, Jpeg 90, Median 9, Noise 80 and PSNR 90). The watermarks have been extracted in all attacks and all cases as shown below. Table 1 describes the attacks used for our tests:

TABLE 1. Used attacks description

| Attack name | Description |
|-------------|---|
| Rot_45 | Rotation with 45 degree |
| Conv_2 | Convolution with factor=2 |
| Jpeg_90 | JPEG compression with quality factor=90 |
| Median_9 | Median filtering (33) |
| Noise_80 | Noise with density degree=80 |
| PSNR_90 | Adding the value 90 for all the pixel intensities |

For $t=0.2$: If all the matrices U,S,V are watermarked, then the extracted watermarks are very far from the original watermark (Fig.6). If U, V are watermarked and S is not modified, then the quality of the extracted watermarks are also very bad (Fig.7). If S is the only watermarked matrix, then for some attacks (Jpeg, PSNR), the extracted watermarks are close to the original watermark, while for the other attacks, the extracted watermarks are far from the original but one can see their edges (Fig.8).

For $t=0.5$: If U,S,V are all watermarked, then the extracted watermarks are not desirable (Fig.9). If U,V are watermarked and S is not modified, the extracted watermarks are also not desirable (Fig.10). If S is the only watermarked matrix, the quality of the extracted watermarks depend upon the attack (the quality is high in Jpeg, PSNR or Median attacks) (Fig.11).

For $t=0.95$: The extracted watermarks are always close to the original watermark (U,S,V watermarked; U,V watermarked, S not modified; S watermarked, U,V not modified) (Fig.12).

Note that the extracted watermark is of high quality especially when t is close to 1 and the watermark is inserted in S.

4. Conclusion. In this paper, we have presented a watermarking algorithm for digital images. The watermark is transformed into SVD and the resulting transformed watermark is inserted into the SVD of the original image. The results show that this method is very robust against the image distortion and may be better than existing methods in the literature, particularly when watermarking only S or watermarking U, S and V, with linear interpolation parameter sufficiently large (close to 1). We have discussed what was achieved by the research community, permitting us to conduct a comparative study for choosing the matrix to be watermarked.

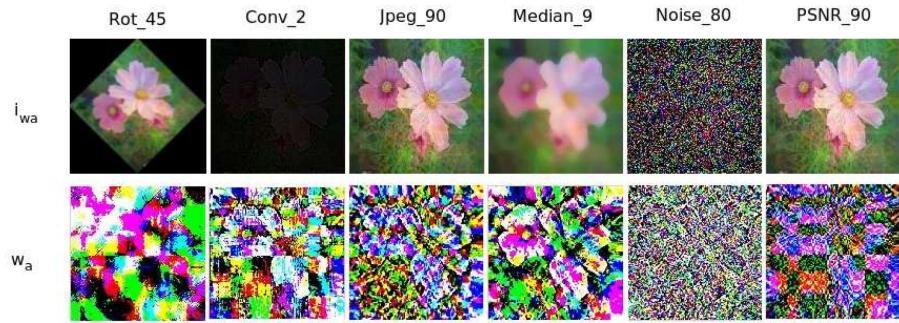
FIGURE 6. $t=0.2$, U,S,V watermarkedFIGURE 7. $t=0.2$, U,V watermarked, S not modifiedFIGURE 8. $t=0.2$, only S watermarked

FIGURE 9. $t=0.5$, U,S,V watermarkedFIGURE 10. $t=0.5$, U,V watermarked, S not modifiedFIGURE 11. $t=0.5$, only S watermarked

FIGURE 12. $t=0.95$, U, S,V watermarkedFIGURE 13. $t=0.95$, U,V watermarked, S not modifiedFIGURE 14. $t=0.95$, only S watermarked

REFERENCES

- [1] S. Voloshynovskiy, S. Pereira, V. Iquise, and T. Pun, Attack modelling: towards a second generation watermarking benchmark, *Journal of Signal Processing*, vol. 81, no. 6, pp. 1171-1214, 2001.
- [2] J. R. Hernandez, M. Amado, and F. Perez-Gonzalez, DCT-domain watermarking techniques for still images: detector performance analysis and a new structure, *Journal of IEEE Trans. Image Processing*, vol. 9, no. 1, pp. 55-68, 2000.
- [3] M. Kutter, and S. Winkler, A vision-based masking model for spread-spectrum image watermarking, *Journal of IEEE Trans. Image Processing*, vol. 11, no. 1, pp. 16-25, 2002.
- [4] W. C. Chu, DCT-based image watermarking using subsampling, *Journal of IEEE Trans. Multimedia*, vol. 5, no. 1, pp. 34-38, 2003.
- [5] A. A. Reddy, and B. N. Chatterji, A new wavelet based logo-watermarking scheme, *Journal of Pattern Recognition Letters*, vol. 26, no. 7, pp. 1019-1027, 2005.
- [6] P. Taoa, and A. M. Eskicioglu, A robust multiple watermarking scheme in the discrete wavelet transform domain, *Proc. of SPIE Conference on Internet Multimedia Management Systems V*, pp. 133-144, 2004.
- [7] Y. Wang, J. F. Doherty, and R. E. Van Dyck, A wavelet-based watermarking algorithm for ownership verification of digital images, *Journal of IEEE Trans. Image Processing*, vol. 11, no. 2, pp. 77-88, 2002.
- [8] V. Solachidis, and L. Pitas, Circularly symmetric watermark embedding in 2-D DFT domain, *Journal of IEEE Trans. Image Processing*, vol. 10, no. 11, pp. 1741-1753, 2001.
- [9] C. K. Chan, and L. M. Cheng, Hiding data in images by simple LSB substitution, *Journal of Pattern Recognition*, vol. 37, no. 3, pp. 469-474, 2004.
- [10] I. J. Cox, and M. L. Miller, Review of watermarking and the importance of perceptual modeling, *Proc. of SPIE Conference on Human Vision and Electronic Imaging II*, pp. 92-99, 1997.
- [11] P. Moulin, and M. K. Mihcak, A framework for evaluating the data-hiding capacity of image sources, *Journal of IEEE Trans. Image Processing*, vol. 11, no. 9, pp. 1029-1042, 2002.
- [12] D. P. Mukherjee, S. Maitra, and S. T. Acton, Spatial domain digital watermarking of multimedia objects for buyer authentication, *Journal of IEEE Trans. Multimedia*, vol. 6, no. 1, pp. 1-15, 2004.
- [13] R. Z. Wang, C. F. Lin, and J. C. Lin, Image hiding by optimal LSB substitution and genetic algorithm, *Journal of Pattern Recognition*, vol. 34, no. 3, pp. 671-683, 2001.
- [14] R. Rykaczewski, Comments on “An SVD-Based Watermarking Scheme for Protecting Rightful Ownership”, *Journal of IEEE Trans. Multimedia*, vol. 9, no. 2, pp. 421-423, 2007.
- [15] R. Liu, and T. Tan, An SVD-based watermarking scheme for protecting rightful ownership, *Journal of IEEE Trans. Multimedia*, vol. 4, no. 1, pp. 121-128, 2002.
- [16] H. C. Ling, R. Phan, and S. H. Heng, On the security of ownership watermarking of digital images based on singular value decomposition, *Journal of Electronic Imaging*, vol. 20, no. 1, pp. 010501-1-010501-2, 2011.
- [17] K. Loukhaoukha, and J. Y. Chouinard, Security of ownership watermarking of digital images based on singular value decomposition, *Journal of Electronic Imaging*, vol. 19, no. 1, pp. 013007-1-013007-9, 2010.
- [18] Y. Zhou, and W. Jin, A novel image zero-watermarking scheme based on DWT-SVD, *Proc. of International Conference on Multimedia Technology*, pp. 2873-2876, 2011.
- [19] S. Kapre Bhagyashri, and M.Y. Joshi, All frequency band DWT-SVD robust watermarking technique for color images in YUV color space, *Proc. of IEEE International Conference on Computer Science and Automation Engineering*, pp. 295-299, 2011.
- [20] A. Benhocine, L. Laouamer, L. T. Nana, and A. C. Pascu, Measuring watermarking robustness using fractal dimensions, *Journal of Communication and Computer*, vol. 7, no. 1, pp. 10-20, 2010.
- [21] Fabien Petitcolas, www.petitcolas.net/fabien/software