b) Extraction algorithm

1. Apply DWT decomposition and decompose the watermarked image into four sub-bands: LL, HL, LH, and HH.
2. Apply SVD to HH band.
   $H = U_H \times S_H \times V_H^T$
3. Extract the singular values from HH band.
4. Construct the watermark using singular values and orthogonal matrices U_w and V_w obtained using SVD of original watermark.
   $W_E = U_w \times S_H \times V_w^T$

c) Signature embedding

1. Generate the signature of N bits for the U and V matrices of watermark.
2. Using DWT decomposition, decompose the cover image into 4 sub-bands: LL, HL, LH, and HH. Further decompose LL band to the 4th level.
3. Modify LL4 and HH4 band coefficients with the bits of signature.

d) Signature extraction

1. Using DWT decomposition, decompose the watermarked image into 4 sub-bands: LL, HL, LH, and HH with help of DWT decomposition, decompose LL band to the 4th level.
2. Extract the signature from LL4 and HH4 band.
3. Generate signature using U and V matrices of the original watermark at the receiver and compare it with extracted signature. If they match, authenticate U and V matrices and use them in watermark estimation.

3.3 Blowfish Algorithm

The data transformation process for Pocket Brief uses the Blowfish Algorithm for Encryption and Decryption, respectively. The details and working of the algorithm are given below.

Blowfish is a symmetric block cipher that can be effectively used for encryption and safeguarding of data. It takes a variable-length key, from 32 bits to 448 bits, making it ideal for securing data. Blowfish was designed in 1993 by Bruce Schneier as a fast, free alternative to existing encryption algorithms. Blowfish is unpatented and license-free, and is available free for all uses. Blowfish Algorithm is a Feistel Network, iterating a simple encryption function 16 times. The block size is 64 bits, and the key can be any length up to 448 bits. Although there is a complex initialization phase required before any encryption can take place, the actual encryption of data is very efficient on large microprocessors. Blowfish is a variable-length key block cipher. It is suitable for applications where the key does not change often, like a communications link or an automatic file encryptor. It is significantly faster than most encryption algorithms when implemented on 32-bit microprocessors with large data caches.

3.4 Algorithm

Blowfish is a variable-length key, 64-bit block cipher. The algorithm consists of two parts: a key-expansion part and a data- encryption part. Key expansion converts a key of at most 448 bits into several subkey arrays totaling 4168 bytes. Data encryption occurs via a 16-round Feistel network. Each round consists of a key dependent permutation, and a key- and data-dependent substitution. All operations are XORs and additions on 32-bit words. The only additional operations are four indexed array data lookups per round.

Subkeys

Blowfish uses a large number of subkeys. These keys must be precomputed before any data encryption or decryption. The P-array consists of 18 32-bit subkeys:
P1, P2,..., P18.
There are four 32-bit S-boxes with 256 entries each:
S1,0, S1,1,..., S1,255;
S2,0, S2,1,..,, S2,255;
S3,0, S3,1,..., S3,255;
S4,0, S4,1,..,, S4,255

The subkeys are calculated using the Blowfish algorithm:

1. Initialize first the P-array and then the four S-boxes, in order, with a fixed string. This string consists of the hexadecimal digits of pi (less the initial 3): P1 = 0x243f6a88, P2 = 0x85a308d3, P3 = 0x13198a2e, P4 = 0x03707344, etc.
2. XOR P1 with the first 32 bits of the key, XOR P2 with the second 32-bits of the key, and so on for all bits of the key (possibly up to P14). Repeatedly cycle through the key bits until the entire P-array has been XORed with key bits. (For every short key, there is at least one equivalent longer key; for example, if A is a 64-bit key, then AA, AAA, etc., are equivalent keys.)
3. Encrypt the all-zero string with the Blowfish algorithm, using the subkeys described in steps (1) and (2).
4. Replace P1 and P2 with the output of step (3).
5. Encrypt the output of step (3) using the Blowfish algorithm with the modified subkeys.
6. Replace P3 and P4 with the output of step (5).
7. Continue the process, replacing all entries of the P array, and then all four S-boxes in order, with the output of the continuously changing Blowfish algorithm.

Encryption

Blowfish has 16 rounds.
The input is a 64-bit data element, x.
Divide x into two 32-bit halves: xL, xR.
Then, for i = 1 to 16:
xL = xL XOR Pi
xR = F(xL) XOR xR
Swap xL and xR
After the sixteenth round, swap xL and xR again to undo the last swap.
Then, xR = xR XOR P17 and xL = xL XOR P18.
Finally, recombine xL and xR to get the ciphertext.

4. Singular Value Decomposition (SVD)

Most tutorials on complex topics are apparently written by very smart people whose goal is to use as little space as possible and who assume that their readers already know almost as much as the author does. This tutorial's not like

that. It's more a manifestivus for the rest of us. It's about the mechanics of singular value decomposition, especially as it relates to some techniques in natural language processing. It's written by someone who knew zilch about singular value decomposition or any of the underlying math before he started writing it, and knows barely more than that now. Accordingly, it's a bit long on the background part, and a bit short on the truly explanatory part, but hopefully it contains all the information necessary for someone who's never heard of singular value decomposition before to be able to do it. SVD is one of the most useful tools in linear algebra with several applications in image compression, watermarking, and other signal processing areas.

SVD of matrix A can be defined as **A = U*S*V^T,** where U and V are the orthogonal matrices and S is a diagonal matrix. Diagonal elements of S are the singular values and they satisfy the following property

s (1,1) > s (2,2) > s (3,3) > . . . . . > s (n,n) .

Let's say we have a matrix A with m rows and n columns, with rank r and r ≤ n ≤ m. Then the A can be factorized into three matrices:

$A = USV^T$

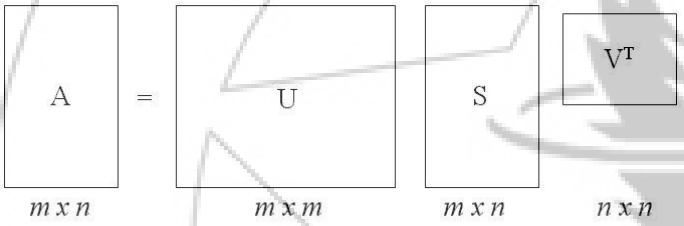


Figure: Illustration of Factoring A to USV^T

Where Matrix U is an m × m orthogonal matrix

U= [u_1, u_2 ,...u_r , u_{r+1} ,..., u_m]

column vectors u_i , for i = 1, 2, …, m, form an orthonormal set:

$u_i^T u_j = \delta_{ij}$ $\begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}$

And matrix V is an n × n orthogonal matrix

V = [v_1 , v_2 ,... v_r, v_{r+1} ,..., v_n]

column vectors $\mathbf{v_i}$ for i = 1, 2, …, n, form an orthogormal set:

$v_i^T v_j = \delta_{ij} =$ $\begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}$

Here, S is an m × n diagonal matrix with singular values (SV) on the diagonal. The matrix S can be showed in following

$$S = \begin{bmatrix} \sigma(1) & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \sigma(2) & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma(r) & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & \sigma(r+1) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & \sigma(n) \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \end{bmatrix}$$

For i = 1, 2, …, n, **i** s are called Singular Values (SV) of matrix A. It can be proved that

$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > 0$ and $\sigma_{r+1} = \sigma_{r+2} = \ldots = \sigma_N = 0$.

For i = 1, 2, …, n, σi are called Singular Values (SVs) of matrix A. The v_i 's and u_i 's are called right and left singular vectors of A.

Properties of the SVD:

There are many properties and attributes of SVD; here we just present parts of the properties that we used in this project.

1. The singular value $\sigma_1, \sigma_2, \ldots \sigma_n$ are unique, however, the matrices U and V are not unique;
2. Since $A^T A = VS^T SV^T$, so V diagonalizes $A^T A$, it follows that the $\mathbf{v_j}$'s are the eigenvector of $A^T A$.
3. Since $AA^T = USS^T U^T$, so it follows that U diagonalizes AA^T and that the u_i 's are the eigen vectors of AA^T.
4. A has rank of r then $\mathbf{v_j}$, $\mathbf{v_j}$, …, $\mathbf{v_r}$ form an orthonormal basis for range space of, R (A^T), and $\mathbf{u_j}$, $\mathbf{u_j}$, …, $\mathbf{u_r}$ form an orthonormal basis for .range space A, R(A).
5. The rank of matrix A is equal to the number of its nonzero singular values.

5. Result Analysis for Test Images

Table 1: Test result for robustness against attacks

Attack	*Image1 PSNR(dB)*	*Image2 PSNR(dB)*	*Image5 PSNR(dB)*
Gaussian Attack	38.1582	37.4074	39.4436
salt & pepper Attack	36.7483	36.6887	38.9567

Table 2: PSNR of watermarked test images

Image	Image1	Image2	Image3	Image4	Image5
PSNR(dB)	40.6136	39.7130	40.6790	39.3388	40.5136
Correlation coefficients	0.9994	0.9995	0.9996	0.9994	0.9991

Table 3: Singular values of HH frequency band of different images

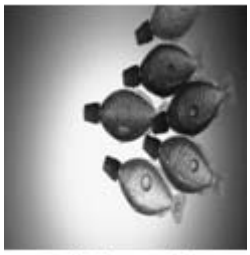
Images	Max singular value	Min singular value
Image1	134	0
Image2	136	0
Image3	139	0
Image4	132	0
Image5	140	0

(a) Image1

(b) Image2

(c) Image 3

(d) Image 4



(e) Image 5

"Table1" shows the watermarked image after attacks. "Fig. 5" shows different test images. "Table 3" shows singular values of the HH band of different test images. To check perceptual similarity between original and watermarked image, Peak-signal-to-noise ratio (PSNR) and correlation coefficient are used as a metric. The PSNR (in dB) of the watermarked images are shown in "Table 2". Correlation coefficient between original and watermarked image is also very close to 1, indicating excellent perceptual fidelity.

6. Conclusion and Future Scope

6.1 Conclusions

Over the internet due to lack of security images can be easily duplicated and distributed without owner permission. Therefore, Digital watermarking is one of the solutions for authentication, copy control and right management of digital media. A digital image is category under digital media. In watermarking, watermark logo get hide into cover image. But while embedding watermark into a cover image, there is a possibility of degradation in quality of original image. Robustness is checked well by extracting original watermark perfectly without any degradation in the original image. Digital asset management system (DAMS) handles almost compressed and encrypted media data. It is possible to watermark these compressed-encrypted media for ownership declaration or copyright management. For encryption we use Blowfish encryption algorithm. It is propose to use a blind watermarking scheme based on DWT and SVD. The main goal is to develop an algorithm to maintain data security and confidentiality of digital media without compromising the quality of that object. The proposed Signature based authentication mechanism addresses the issue of authentication and security which most of the existing DWT & SVD based methods fail to handle. The used algorithm is simple to implement as it is directly performed in the compressed encrypted domain as it does not require decryption or de-compression of the image. In compression it is concluded that as increase in decomposition level more the image get compressed but results in loss of image quality. This technique also preserves the confidentiality of content as embedding is done on encrypted images.

6.2 Future Scope

Future work consists of two aspects as given below:

1. To work on more complex image such as vary large size 2D image or 3D images with SVD technique for image compression.
2. Deeply to study and investigate the roles of singular values and singular vectors in image processing.

References

[1] V. Subramanyam, Sabu Emmanuel, "Robust Watermarking of Compressed and Encrypted JPEG2000 Images", IEEE Transactions On Multimedia, Vol. 14, No. 3, June 2012.

[2] B.Chandra Mohan, S. Srinivas Kumar, "A Robust Image Watermarking Scheme using Singular Value Decomposition", JOURNAL OF MULTIMEDIA, VOL. 3, NO. 1, MAY 2008.

[3] Bhandari Kunal,Mitra Suman K and Jadhav Ashish, " A hybrid approach to digital image watermarking using singular value decomposition and spread spectrum" , S K Pal et al (eds): PreMI, LNCS 3776: 272–275, 2005.

[4] Dr. J. Abdul Jaleel, Jisha Mary Thomas, "Guarding Images using a Symmetric key Cryptographic Technique: Blowfish Algorithm", International Journal of Engineering and Innovative Technology (IJEIT) Volume 3, Issue 2, August 2013.

[5] Ghazy R A, El-Fishawy N A, Hadhoud M M, Dessouky M I and El-Samie F E A, "An efficient block by block SVD-based image watermarking scheme", Radio Science Conference, NRSC 1–9, 2007.

[6] Kasmani S A and Naghsh-Nilchi A, "A new robust digital image watermarking technique based on joint DWT-DCT transformation", Convergence and Hybrid Information Technology ICCIT '08 Third International Conference 2(1):539–544, 2008.

[7] Manpreet Kaur , Sonika Jindal , Sunny Behal , "A Study Of Digital Image Watermarking" , International Journal Of Research In Engineering & Applied Sciences Volume 2, Issue 2 ,Issn: 2249-3905, February 2012.

[8] S.Ramakrishnan, T.Gopalakrishnan, K.Balasamy , "SVD Based Robust Digital Watermarking For Still Images Using Wavelet Transform", D.C. Wyld, et al. (Eds): CCSEA 2011, CS & IT 02, pp. 155–167, 2011.

[9] Sangeeta jadhav, Dr. Anjali Bhalchandra, "Robust Digital Image-Adaptive Watermarking Using BSS Based Extraction Technique", International Journal of Image Processing (Ijip) Volume (4): Issue (1).

[10] Sneha Jose, Rajesh Cherian Roy, Sreenesh Shashidharan," Robust Image Watermarking based on DCT-DWT-SVD Method", International Journal of Computer Applications (0975 – 8887) Volume 58– No.21, November 2012.

[11] Sachin Dhawan, "A Review of Image Compression and Comparison of its Algorithms"; IJECT Vol 2, Issue1, ISSN: 2230 -7109 (Online) -ISSN: 2230 - 9543 (Print); March 2011.

[12] Deepa Mathew K, "SVD based Image Watermarking Scheme", IJCA Special Issue on "Evolutionary Computation for Optimization Techniques" ECOT, 2010.

[13] Lee Sin-Joo and Jung Sung-Hwan, "A survey of watermarking techniques applied to Multimedia", industrial electronics, Proceedings ISIE, IEEE International Symposium pp.272–277, 2001.

[14] Pia Singh, Prof. Karamjeet Singh, "Image Encryption And Decryption Using Blowfish Algorithm In Matlab", International Journal of Scientific &

Engineering Research, Volume 4, Issue 7, 150 ISSN 2229-5518, July-2013.

[15] Ting-Xian Zhang, Wei-Min Zheng, "Comments on A Semi-blind Digital Watermarking Scheme Based on SingularValue Decomposition", Eighth International Conference on Intelligent Systems Design and Applications, 2008.

[16] Mingyan Wang, Yanwen Que, "The Design and Implementation of Passwords Management System Based on Blowfish Cryptographic Algorithm",International Forum on Computer Science-Technology and Applications, 2009.

[17] A.Sindhuja, R. Logeshwari, K.ThirunadanaSikamani, "A Secure PMS based on Fingerprint Authentication and Blowfish Cryptographic Algorithm", International Conference on Signal and Image Processing, 2010.

[18] G. Dayalin Leena and S. Selva Dhayanithy, "Robust Image Watermarking in Frequency Domain", International Journal of Innovation and Applied Studies, ISSN 2028-9324 Vol. 2 No. 4, pp. 582-587, Apr. 2013.

[19] Anjan Pal, Snehasish Banerjee, "Embedment of Encrypted Text and Secret Images for Digital Image Watermarking" World Applied Programming, Vol (1), No (3), 132-137, ISSN: 2222-2510, August 2011.

[20] Yashovardhan Kelkar, HeenaShaikh, Mohd.Imran Khan, "Analysis of Robustness of Hybrid Digital Image Watermarking Technique under Various Attacks" IJCSMC, Vol. 2, Issue.3, ISSN 2320–088X, pg.137 – 143, March 2013.

[21] Kamrul Hasan Talukder and Koichi Harada, "Haar Wavelet Based Approach for Image Compression and Quality Assessment of Compressed Image", IAENG International Journal of Applied Mathematics, 36:1, IJAM_36_1_9, 2007.

[22] F. Mintzer and G. W. Braudaway, "If one watermark is good, are more better?" in Int. Conf. Acoustics, Speech, Signal Processing, 1999, pp.2067–2070.