gorithm for Capsule Endoscopy Localization, 2012 5th International Conference on Biomedical Engineering and Informatics (BMEI), Chongqing, China Oct, 2012, pp. 721-725.
14. Savage N. Twitter as medium and message. Communications of the ACM, 2011, 54(3), pp. 18-20.
15. Stock, G.N., Tatikonda, M.V. A typology of project-level technology transfer processes. Journal of Operations Management, 2000, 18, pp. 719-137.
16. Su, Tianyun, Zhihan Lv, Shan Gao, Xiaolong Li, and Haibin Lv. 3D seabed: 3D modeling and visualization platform for the seabed. In Multimedia and Expo Workshops (ICMEW), 2014 IEEE International Conference on IEEE, 2014, pp. 1-6.
17. Toke Bjerregaard. Industry and academia in convergence: Micro-institutional dimensions of R&D collaboration. Technovation; 2010, 30, pp. 100-108.
18. Xianyu, B., Yang, J.M. Evolutionary ultimatum game on complex networks under incomplete information. Physica A, 2010, 389, pp. 1115-1123.
19. Y. Geng, J. Chen, K. Pahlavan, Motion detection using RF signals for the first responder in emergency operations: A PHASER project, 2013 IEEE 24nd International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC), London, Britain Sep. 2013.
20. Zhang, Mengxin, Zhihan Lv, Xiaolei Zhang, Ge Chen, and Ke Zhang. Research and Application of the 3D Virtual Community Based on WEBVR and RIA. Computer and Information Science 2, no. 1, 2009, 84p.

Application of Multi Feature Watermarking Algorithm for Ownership Protection in the Judicial Authentication and Copyright

WANG Hong[1]

[1]*Vocational Education Department, Liaoning Police Academy, Dalian Liaoning 116036, China*

Corresponding author is WANG Hong

Abstract
Aiming the problem of the poor robustness of software watermarking, the low efficiency of the implementation of Watermarking Sharing Algorithm, we put forward a kind of software watermarking scheme based on chaotic optimization. The scheme by introducing chaos system, taking matrix partition, chaotic scrambling on the watermark information to form the sharing watermarking; when watermarking embedding, encoding the sharing watermarking as DPPCT topology one by one, and filling the Info domain of every DPPCT with the watermark information treated by Hash; after embedding the watermark, encrypted by chaotic, to protect all the code and prevent the damage brought by the reverse engineering and other means to the software watermarking. Theoretical analysis and

experimental results show that the scheme can effectively resist all kinds of semantics preserving transformation attacks, reduce the program load, to improve the robustness and efficiency of watermark.
Keywords: LAW PROTECTION, MULTI FEATURE WATERMARKING ALGORITHM, OWNERSHIP PROTECTION, JUDICIAL AUTHENTICATION, COPYRIGHT

1. Introduction

With the rapid development of the Internet, it greatly promotes the network to download and dissemination of utility software. When people enjoy the convenience of Internet downloads, software piracy and reuse are increasingly rampant, bringing huge economic losses to individuals and businesses, so the protection of software copyright get more attention. For the protection of software copyright, software watermarking emerge d as the times require, and it is a branch of the digital watermarking, being a cross research field of information security, cryptography, graph theory, algorithm design, software engineering. Because the concealment, high security, software watermarking based on graph theory has become the hotspot of the research.

Literature proposed CT (Collberg-Thombo-rson) algorithm of the first dynamic graph watermarking in DGW (Dynam-ic Graph Watermarking), the main idea is to use 2large prime number P, the product W ($W = P \times Q$) of Q to represent the watermark information, then coding W as a topological graph and embedding into the program code. Because W is a big number, in the watermark embedding process, it often need to be split into multiple sub watermark, to improve the watermark's invisibility, robustness and reduce the complexity of constructing the topology of the watermark. The literature proposed watermarking sharing algorithm based on Chinese remainder theorem, the secret of it being good, but the watermark recovery process is complex, the amount of calculation being large. The literature proposed Asmuth-Bloom (AB) threshold algorithm based on secret sharing theory, enhancing the robustness of the watermark, but in the realization of the process it easily led to a substantial expansion of the watermark data. The literature, firstly applied chaos theory in the software watermarking system, to improve the traditional watermarking algorithm Easter Egg watermarking by chaotic pretreatment and chaotic hash code. But because after embedding watermark, the algorithm changed the module code and the loading position in memory, we need to specify the location of the custom code marking, reducing the watermark invisibility. The literature presented a dynamic watermarking algorithm based on chaos optimization, its strong robustness, can effectively resist the attack of reverse engineering, but the chaos encryption process is complex, having great influence on performance of program.

Aiming the disadvantages of the above schemes, this paper put forward a kind of software watermarking scheme based on chaotic optimization. Through taking the matrix partition, Chaotic Scrambling(CS) on watermark information, it effectively control the watermarking sharing granularity, and improve the watermark invisibility. Then coded the sharing watermark as DPPCT (Double circular linked Planted Plane Cubic Tree) topology one by one, filling the Info domain of every DPPCT with the watermark information treated by Hash to form the coexisted structure of the watermark information and the watermark branch. After embedding watermark, using chaotic to encrypt CE (Chaotic Encryption) system, dividing the program code into CSB (Code Sensitive Block) and CIB (Code Insensitive Block. Using the Hash value of CIB to encrypt CSB to form the cross protection mechanism to protect all the codes and improving the anti-attack ability and the robustness of the watermark.

2. The Problem Model

2.1. The Basic Model of Dynamic Graph Software Watermark

Software watermarking system can be represented by using a ten tuple $(P_o, W, K, E_m, E_x, P_w, D_r, D_e, A_t, P_w^{'})$. Due to the addition of watermarking sharing algorithm and watermarking topological code, dynamic graph watermark system can be expressed as $DGS = (P_o, \{I_i\}, W, f, f^{-1}, g, R, E_m, E_x, P_w, D_r, D_e, A_t, P_w^{'})$. In which, Po is the original program; W being the watermark information; Pw being the program after embedding the watermark; {Ii} being the key of user input; Watermarking sharing mapping function f satisfies: $f(W) \to \{W_i\}$; the inverse mapping function f^{-1} satisfies: $f^{-1}(\{W_i\}) \to W$; Topological graph encoding function g satisfies: $g(W_i) \to G_i$; Gi being watermarking topology structure; R being recognition function; Em being embedding algorithm; $E_m(P_o \times W \times I_m) \to P_w$; Ex being extraction algorithm: $E_x(P_w \times I_m) \to P_o$; Dr being the data rate; De being crypticity; anti attack ability: $A_t(P_w) \to P_w^{'}$.

In the premise of ensuring the function of Po and Pw being coincident, the measure of dynamic graph watermarking system mainly has: 1) data rate, watermarking topology expressing the degree of the size of watermark information; 2) crypticity, the imperceptible degree of watermarking topology; 3) anti attack ability, the degree of resistance to various attacks of the watermark topology; 4) overload performance, the influence degree after embedding watermarking topology on execution performance of program. The 4 characteristics are interdependent, mutually contradictory, we need to carry on the balance in order to achieve optimal performance.

2.2. Chaos Theory

Chaos is a similar stochastic uncertain output in the description of non-rowar dynamic system, disorder containing the order, it having characteristics of the un-decomposable, regularity and the unpredictable. Logistic mapping came from the famous statistical model, being a kind of chaotic dynamical system which is widely used at present, its dynamic mathematical model can be expressed as: $x_{k+1} = \mu x_k(1-x_k)$. in which, chaos domain being $(0,1)$, $0 \le \mu \le 4$ being called bifurcation parameter, $x_k \in (0,1)$. The researches of chaotic dynamic system pointed that, when $3.5699456... < \mu \le 4$, Logistic mapping worked in chaotic state. The sequence $\{x_k; k = 0,1,2,3...\}$ produced by the initial conditions x0 in the action of Logistic mapping is non periodic, non-convergence and being very sensitive to the initial value.

Translated into a binary function as follows:

$$S_n(k) = R_n(x_k) = \begin{cases} 0, x_k \in \bigcup_{d=0}^{2^{n-1}-1} I_{2d}^n \\ 1, x_k \in \bigcup_{d=0}^{2^{n-1}-1} I_{2d+1}^n \end{cases}$$

In the formula, n being any positive integer; $I_0^n, I_1^n, I_2^n, ...$ being 2n successive equal interval of the interval [0,1], by invoking $S_n(k)$ to transfer the chaos sequence $\{x_k; k = 0,1,2,3...\}$ into the binary output sequence.

3. The Software Watermarking Sharing Scheme Based on Chaos Optimization

The software watermarking sharing scheme based on chaos optimization, took matrix partition on watermark information based on the original CT algorithm to form sharing watermarking using chaotic scrambling CS to encrypt the existing watermarking. And then used DPPCT to express the sharing watermark after encryption, and filling the Info domain with the watermark information after Hash treatment. Finally, using the hybrid to encrypt CE, encrypting sensitive code segment CSB of the program. The overall framework of the scheme is shown in figure 1.

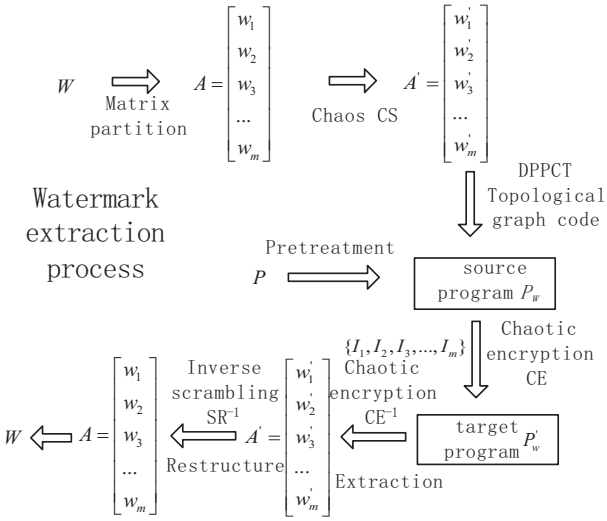

Figure 1. The extraction process of sharing watermark embedding based on chaos optimization

The embedding process of watermark is as follows:

Took matrix partition on the watermark information W, forming sharing watermarking $(w_1, w_2, w_3, ..., w_m)$, and transferring it into matrix A;

Using chaos scrambling CS to take scrambling encryption on matrix A, forming matrix A';

Taking pretreatment on the sound code of program to determine the embedding position of the watermark and the program sensitive code segment CSB, adding the Hash function, the encryption and decryption functions;

Taking DPPCT topological graph code on the sharing watermark information $w_i'(1 \le i \le m)$ of every row of matrix A', and embedding into the specified position, through the input sequence of users $\{I_1, I_2, I_3, ..., I_m\}$ in running state of the program, making it be generated in the stack;

compiled and generated object program;

Using chaos encryption CE to take encryption.

The watermark extraction process is the inverse process of watermark embedding.

3.1. Matrix Partition Algorithm

The algorithm transferred W unto matrix A containing watermark information0, 1 by taking pretreatment on the watermark information W.

Firstly, converted the watermarking information W into binary information, according to the length Lw of binary information to split the watermark information W, the length of each section being $\lfloor \sqrt{L_w} \rfloor + 1$, the length of the last section being insufficient for

$\lfloor\sqrt{L_w}\rfloor+1$, deciding it as the last one, being 0 adding 1, being 1 adding 0. The watermark information after treatment recorded as $W=(w_1,w_2,w_3,...,w_m)$. And then construct the $A_m \times A_n$ matrix A, in which, $A_m=m$, namely watermark branch number, $A_n=\lfloor\sqrt{L_w}\rfloor+1$, namely the length of every section, putting the watermark information after partition in each row of matrix according to the order of sequence.

For example: watermark information W=29, transferred it into the binary watermark information 11101, according to the above algorithm $A_m=2$, $A_n=3$, so $A=\begin{bmatrix}111\\010\end{bmatrix}$.

3.2. Chaotic Scrambling

The algorithm produced $2m\times n$ chaotic sequence values, we can take scrambling encryption on the 0,1 matrix of any $m\times n$. The scrambling process is as follows:

The sequence C containing $2m\times n$ chaotic sequence values produced by chaotic system;

Taking matrix partition on chaotic sequence C to get 2 matrixes C1,C2 of $m\times n$;

Taking the operation of XOR by row on the matrix A and sequence C to form new matrix T1, namely $T_1=A\oplus C_1$;

Taking transposition on T1 to form new matrix T2;

Taking the operation of XOR by row on the matrix T2;and sequence C2 to form new matrix T3, namely $T_3=T_2\oplus C_2$;

Taking transposition on T3 to form new matrix T4;

The above is the process of iterative scrambling of the matrix of watermark information A and finally got the matrix T4, namely the matrix A' after chaotic scrambling. If you want to get a better scrambling encryption result, you can take several iterations on the above process.

It's inverse process is: firstly, taking transposition on T4 to get matrix T5, and then taking the operation of XOR on matrix T5 and C2 to get matrix T6, and then taking transposition on T6 to get matrix T7, finally taking the operation of XOR on matrix T7 and C1 to get matrix T8. The matrix T8 is the matrix A of original watermark information.

3.3. DPPCT

The dynamic watermark graph topologies of now mainly include: the chain table of cardinal number K, parent pointer tree, Pareto diagram, PPCT structure. Among them, the data rate of the chain table of cardinal number K is the highest, but having the poor anti attack, and the data rate of PPCT structure being lower but having strong anti –attack. Combining with the chain table of cardinal number K and PPCT structure to fabric hybrid encoding DPPCT structure.

The DPPCT node adds a pointer domain and Info domain for each node in the original PPCT node structure, its structure being shown in figure 2:

L	R_1	R_2	Info

Figure 2. The structure of DPPCT node

The code structure of DPPCT is as follows:

Spreading the watermark branch after chaotic scrambling $w_i^{'}(1\le i\le m)$ to the form of cardinal number ki.

Fabricating DPPCT structure with ki leaf nodes, changing theR1, R2 pointer from the right leaf node, coding the coefficient of each item by this, the rules are as follows:

if R1, R2 refer to themselves, the leaf node do not express any coefficient;

if R1 refers to itself, R2 pointing to the neighboring nodes, then the coefficient being 0;

if R1 points to other leaf nodes, R2 pointing to the neighboring nodes, from leaf nodes the R1 points to the original leaf node, the number of leaf nodes passing by being n, the coefficient being n.

Adding information to the Info domain, the rules are as follows:

Adding $w_i^{'}$ and cardinal number ki to the Info domain of Origin node;

Taking Hash process on the watermark information W, Hash function can use SHAI, MD5 and other algorithms to get the abstract of the information Wh ;

Adding Wh to the Info domain of the other nodes.

Through the anaphora of the new added pointer domain makes the DPPCT structure have the characteristic of double circular chain table. If change the pointer of a node or adding, deleting a node, through anaphora technique of the pointer to restore the node. For leaf nodes, if you changed the L, R2 pointer, it having little influence on the code of coefficient; if changed the R1 pointer would cause the mistake of the code of coefficient. Therefore, we can fabricate DPPCT structure having more than ki leaf nodes to confuse the attacker. At the same time, through adding the watermark information Wh after Hash treatment to Info domain, makes every node have Wh except Origin node, even if the attacker destroyed one node, other nodes will still contain Wh.

Assuming the number of watermark expressed by $w_i^{'}$ was 12, cardinal number being 3, the expansion

being $w_i^{'} = 12 = 0 \times 3^0 + 1 \times 3^1 + 1 \times 3^2$, It's DPPCT structure is shown in figure 3:

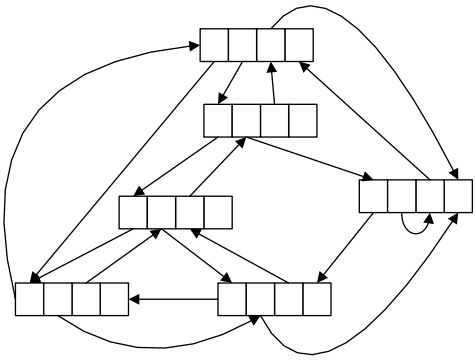

Figure 3. The DPPCT topology structure with3 leaf nodes

3.4. Chaotic Encryption CE

At present, the more practical chaotic cipher algorithm is to combine chaotic cipher algorithm with the traditional cipher algorithm with excellent characteristics to fabricate new cipher algorithm. This paper combined chaotic sequence and AES encryption algorithm to fabricate chaotic encryption scheme with variable length keys and cross protection mechanism, resisting known plaintext differential and linear attacks, to protect the watermark and sensitive code and combining with hash code of non -sensitive code to realize the tamper proofing function of watermark system 11], the encryption process being shown in figure 4.

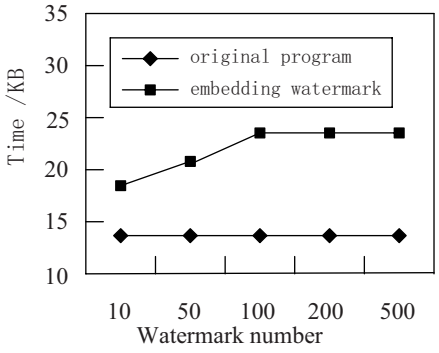

Figure 4. The change of program size after embedding watermark

The algorithm flow is as follows:

First, marking the sensitive code to be encrypted as $(CSB_1, CSB_2, ..., CSB_n)$, the non-sensitive code marked as $(CIB_1, CIB_2, ..., CIB_{n'})$, in which, CSBi may contain the watermark information $w_j^{'}$, $n \le n'$;

The chaotic system produced chaotic sequence $Q = (q_1, q_2, q_3, ..., \ q_n)$;

Calculating the starting address of CSBi, address1 and address2, the starting address of CSBi address3 and address4;

Using the Hash algorithm to calculate the check of the code segment between address1and address2 and si = Hash(address1, address2), the check of the code segment between address3 and address4 and vi = Hash(address3, address4);

To calculate the encryption key Key by the formula Keyi=Hash$[v_i \oplus q_i \oplus s_i]$ and conserve qi, si to decrypt;

To encrypt CSBi, AES $(CEB_i, Keyi)$.

When meet the CSBi code block of ciphertext during the execution of the program, calling the Decrypt decryption function to decrypt CSBi, then, executing the code block CSBi of plaintext. In the encryption process, using chaotic sequence Qi as a parameter, it having a variable length key, can effectively increase the difficulty of cracking of an attacker, using the Hash value of CIBi as a parameter, which can effectively prevent the entire program to be tampered. After the run of the program, we call the encryption module again to complete protection of sensitive code segment.

4. The Analysis of Performance

4.1. The Analysis of Robustness

The robustness of dynamic graph watermark is mainly reflected on the anti-attack of topology structure. DPPCT topology graph is stored in the stack structure of dynamic creation and it can have many kinds of graph structure, so it is difficult to locate the watermark. At the same time as the changed graph topology, enhanced the anti - pattern matching and collusion attack of watermark. Secondly, through the new added pointer and Info domain makes the DPPCT structure have the feature of double circular linked list, and makes every node except Origin contain watermark information, to further improve the robustness and security of DPPCT. The comparison of anti-attack of DPPCT structure with the anti-attack of other coding schemes is shown in table 1. Finally, through the chaotic system took chaotic scrambling and chaotic encryption on watermark, using the good pseudo randomness and autocorrelation of chaotic sequence to generate the key to resist the differential and linear attacks of known plaintext. If break the two-dimensional chaotic parameter μ, we need 2 pairs of associated value of the chaos, for chaotic system with n chaotic sequence values, the crack probability being n-2. At the same time, using the Hash value of sensitive and non-sensitive code segment, encrypting the sensitive segment, forming cross protection mechanism to protect all codes and enhance the tamper resistant ability of watermark system. If a piece of code segment of the program is modified,

it will cause the failure of decryption, the program stopping running.

Table 1. The comparison of the anti-attack of 3 coding schemes

Coding scheme	add cutting attack	distortive attack	fault tolerance
cardinal number K	wake	wake	ordinary
PPCT	strong	strong	strong
DPPCT	strong	stronger	stronger

4.2. The Analysis of Data Rate

The DPPCT structure combined the characteristics of cardinal number K coding and PPCT coding, the range can be coded by the DPPCT having 2n nodes is from o to nn-1-1. When the number of nodes is certain, the comparison of data rate of cardinal number K, PPCT, DPPCT as shown in table 2.

Table 2. The comparison of the analysis of data rate of different coding methods

Number of node	cardinal number K	PPCT	DPPCT
n	n^{n-1}-1	$2C_{n-2}^{n/2-1}/n$	$(n/2)^{n/2-1}-1$
10	1.00×10^{9}	1.40×10^{1}	6.25×10^{2}
20	5.24×10^{24}	4.86×10^{3}	1.00×10^{9}
50	1.78×10^{83}	1.29×10^{12}	3.55×10^{33}

4.3. The Analysis of Performance Overload

After embedding watermark, it will certainly have an impact on the performance of the program, mainly showing as space overload. and time overload. The experiment used SandMark platform to process the TTT.jar program, embedded different number of watermark, and analyzed the influence on the size and running time of original program, the results being shown in figure4 and figure 5. As can be seen from Figure 4, after watermarking sharing, the increase of program size became slowly. This is because that taking matrix partition on watermark information makes the number of watermark sharing be linear increase and using DPPCT with higher data rate to code, reduced the program load. As can be seen from Figure 5, after embedding the watermark, the running time of the program were not significantly influenced. This is because that the embedding watermark program code is not involved in the operation of main function modules of program and only took chaotic encryption on the sensitive segment, and did not significantly increase the overall running time of the program.

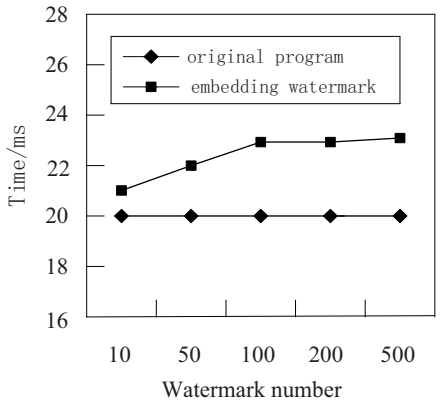

Figure 5. The change of the running time of program after embedding watermark

4.4. The Analysis of Anti-reverse Engineering

The topology structure of DPPCT hidden in the stack structure of dynamic creation, can prevent static reverse engineering attacks. After using chaotic encryption, taking chaotic encryption on the watermark sharing, after assuming the number of watermark sharing being m, the number of the address that can be embedded being n, the chaotic state being x. When the address that can be embedded n being unchanged, the time complexity we need to break all watermark is $\delta 1 = Tm\chi$, in which, T being a constant, with the increase of the number of watermark m, the complexity being linear increases, but increasing the performance loan, the performance influence degenerating. If the number of the embedding watermark sharing m being unchanged, the time complexity we need to break all watermark is $\delta_2 = TmC_n^m\chi$, when the number of embedding address increased, the complexity increasing exponentially, but the protection strength will decrease. Both are dependent on the chaotic state x is known, if x being unknown, can't be broke. .At the same, we should take tradeoff of the protection strength between the degradation and regression of the performance of program.

Conclusion

In this paper, chaos optimization sharing software watermarking scheme, through the chaotic scrambling and watermarking sharing, improve the invisibility of software watermark and fabricated topology structure of DPPCT with hybrid encoding, improved the robustness and anti-attack watermark. At the same time, combining with the chaotic encryption, protecting watermark information and all of the code to prevent the falsify of software watermark. Next to research how to combine the characteristics of software, to construct more perfect watermarking topology structure, and using chaotic system and software

tamper resistant technology to fabricate more practical watermarking tamper proof system.

References

1. C K Tan, J C Ng, X T Xu, C L Poh,Y L Guan, K Sheah. Security protection of DICOM medecak inmages using dual-Layer reversible watermarking with tamper detection capability. Journal of Digital Imaging, 2011,24(3), p.p.528-540.
2. E Varsaki, V Fotopoulos, A N Skodras. A reversible data hiding technique embedding in the image histogram, Technical Report HOU-CS-TR-2006-08-GR, Hellenic Open University, 2006.
3. FENG Hong-yu, GU Yue-sheng, LI Yan-cui. Research on Digital Watermarking Based on Wavelet Theory. Journal of Convergence Information Technology, 2012, 7(13), p.p.292-299.
4. H.W. Tseng, C.P.Hsieh. Prediction-based Reversible Data Hiding. Inform Science,2009 179(14), p.p.2460-2469.
5. Jiang, Dingde, Zhengzheng Xu, Peng Zhang, Ting Zhu. A transform domain-based anomaly detection approach to network-wide traffic. Journal of Network and Computer Applications, 2014, 40, p.p.292-306.
6. Jie He, Yishuang Geng, Kaveh Pahlavan. Toward accurate human tracking: Modelling Time-of-Arrival for Wireless Wearable Sensors in Multipath Environment, IEEE Sensor Journal, 2014, 14(11), p.p.3996-4006.
7. K S Kin, M J Lee, H Y Lee, H K Lee. Reversible data hiding exploiting spatial correlation between sub-sampled images. Pattern Recognition, 2009, 42, p.p.3083-3096.
8. Liu Z,S.F.Sergio, S.L.M.B.Paulo. Providing Integrity and Authenticity in DICOM Images: A Novel Approach. IEEE Transactions on Information Technology in Biomedicine, 2009, 13(4), p.p.582-589.
9. NAVAS K A, SASIKUMAR M. Survey of Medical Image Watermarking Algorithms. Proc of the 4th Sciences of Electronic, Technologies of Information and Telecommunications International Conference, 2007, pp. 25-29.
10. PLANITZ B, MAEDER A. Medical image watermarking: a study on image degradation. Proc of the Australian Pattern Recognition Society Workshop on Digital Image Computing, 2005, pp. 8-13.
11. Su, Tianyun, Zhihan Lv, Shan Gao, Xiaolong Li, Haibin Lv. 3D seabed: 3D modeling and visualization platform for the seabed. In Multimedia and Expo Workshops (ICMEW), 2014 IEEE International Conference on, 2014, pp. 1-6.
12. W Hong, T S Chen, Y P Chang. A High Capacity Reversible Data Hiding Schema Using Orthogonal Projection and Prediction Error Modification. Signal Process, 2010, 90, p.p.2911-2922.
13. X H Deng, Z G Chen, et al. A Study of Watermarking Application in Medical Digital Images, 2011 International Conference on Biomedical Engineering and Computer Science ICBECS2011,IEEE Computer Socienty, 2011,4, pp.553-556.
14. Y C Li, C M Yeh, C C Chang. Data hiding based on the similarity between neighboring pixels with reversibility. Digit Signal Process,2010, 20, p.p.1116-1128.
15. Y Li, J Yu,S Wei. Reversible Data Hiding Algorithm Based on Prediction Error. International Asia Conference on Information in Control, Automation and Robotics, 2010, pp. 353-356.
16. Yishuang Geng, Kaveh Pahlavan, On the Accuracy of RF and Image Processing Based Hybrid Localization for Wireless Capsule Endoscopy, IEEE Wireless Communications and Networking Conference (WCNC), Mar. 2015.
17. Z Ni,Y Q Shi,N Ansari, W Su. Reversible Fata Hiding. IEEE T Circ, Syst.Vid, 2006,16(3), p.p.354-362.
18. ZHAN Yong-feng, FENG Xue, FU Chong et al. An Efficient Medical Image Cryptosystem Based on Chaotic Maps. International Journal of Digital Content Technology and its Applications, 2012, 6(13), p.p.265-274.
19. Zhang, Mengxin, Zhihan Lv, Xiaolei Zhang, Ge Chen, and Ke Zhang. Research and Application of the 3D Virtual Community Based on WEBVR and RIA. Computer and Information Science 2, no. 1, 2009, pp.84.
20. ZHOU Y X, JIN W. A novel image zero-watermarking scheme based on DWT-SVD. Proc of the 2010 IEEE International Conference on Multimedia Technology. 2009, pp. 2873-2876.