A Novel Threshold Secret Sharing Scheme Using FFT Algorithm

Abdulrakeeb M. Al-Ssulami

Computer Science Dept., College of Computer and Information Sciences, King Saud University, P.O. Box 51178, Riyadh 11543, Saudi Arabia

Corresponding Author; Address: e-mail: rakeeb_sulami@yahoo.com

Abstract- Secret sharing schemes (SSS) are very important, because they are used in critical applications such as e-voting, cryptographic key distribution and sharing, secure online auctions, information hiding, and secure multiparty computation. We explained some popular algorithms of secret sharing such as threshold, graph, and visual schemes and their access structures. Besides, we discussed the limitations of those available schemes. Additionally, we proposed a novel threshold secret sharing scheme based on Fast Fourier Transform (FFT) algorithm, which is used for the first time in this paper in the field of secret sharing. That is, we exploited the robust characteristics of FFT such as linearity, reversibility, efficiency, that has time complexity of $O(n \log n)$, and it provided us with a wider field, complex numbers. The presented scheme is ideal; the share's size is smaller than the secret and very secure because it depends on solving a linear system of equations generated by FFT. Thus, Our SSS combines the merits of Shamir and Blakley schemes.

Keywords- Secret sharing; secret hiding; FFT algorithm; linear algebra.

1. Introduction

SSS (Secret Sharing Scheme) is the way of dividing the secret to shares and distributing them to a group of participants. The set of groups that are capable to reconstruct the secret is called access structure and its elements are called authorized groups [1]. Any access structure $F \subset 2^p$, p is the set of trustees and 2^p is the power set of p satisfies the following condition:

if $A \in F$ and $A \subset A' \subset p$ then$A' \in F$ [1]. This condition means that for each authorized group there exists a larger one also authorized. The maximal sets of F denoted $\partial^+ F$. It holds all authorized groups with maximum number of participants and it is defined as it follows: $\partial^+ F = \{A \in F : A \not\subset A' \text{ for all } A' \in F - \{A\}\}$. Also the minimal sets of F defined as: $\partial^- F = \{A \in F : \neg(A' \subset A) \text{ for all } A' \in F - \{A\}\}$

[1]. There are a set of models to describe the secret sharing schemes. In [2], the secret sharing model is represented as a matrix with $(n+1)$ columns, one for dealer and the rest for users. The matrix M is public, while the row r kept private which contains the secret S. For a group F of authorized users, they pool their shares to identify the row r. In [3], more interested model is introduced and refined in [4]. The model of SSS is represented as a set R of distribution rules. The distribution rule is a

function $R : \{0,1,...,n\} \rightarrow S \bigcup \bigcup_{i=1}^{n} S_i$ such that $R(0) \in S$ and $R(i) \in S_i$ where $1 \leq i \leq n$. The distribution rule R represents a possible distribution of shares to the participants, the $R(0)$ represents the shared key and $R(i)$ represents the share i given to participant $P(i)$.

Other models are introduced in [5,6,7,8] which depend on the entropy concept. In [8], the model of SSS is described as it follows: for n users with numbers $1, 2, ..., n$ and a set of groups $F \subseteq 2^p$. A perfect F - secret sharing scheme is a collection of random variables $(S, I_1, I_2, ..., I_n)$ where the correctness and security conditions must be satisfied. The correctness condition means that for any authorized group, the secret key must be recovered correctly. The entropy function for correctness is written as: $\forall A \in F, E(S \mid \{I_i \mid i \in A\}) = 0$, E is the conditional entropy function of S given I_i. And for the security condition, the secret key can't be recovered if the shares of unauthorized participants are known. The entropy function is written as:

$$\forall A \in \overline{F}, E(S \mid \{I_i \mid i \in A\}) = E(S).$$

Because the importance of protecting the secret from leakage, many algorithms are introduced to partition and distribute the secret over a set of users, so that only the authorized group can recover it. Thus, in this paper we will explore different types of secret sharing schemes, threshold, graph, and visual secret sharing schemes and then we will explain our novel threshold secret scheme which based on FFT algorithm. Our scheme is efficient and ideal and one can hide any secret without restriction. The main idea behind our secret scheme is exploiting the FFT to generate a linear system of equations instead of choosing $n(K+1)$ coefficients and keeping secret $(K+1)^2$ coefficients, as in Blakley's scheme. In our scheme we only distribute pairs of values (x, y) which should be kept secret as in Shamir's scheme. In our and Shamir schemes, there are only $2K$ coefficients should be kept secret. Additionally, our scheme chooses shares randomly from and it is more practically than others, especially when the number of shares very large.

2. SSSs Based on Threshold

In this type of secret sharing, the important thing is the number of participants required to reconstruct the secret. If the number of authorized participants K out of n, we call this scheme (k, n)-threshold access structure and defined as:

$$F = \{A \in 2^p \mid |A| \geq K\}, \quad 2 \leq K \leq n \cdot$$

The following schemes are examples of threshold schemes [2].

2.1. Blakley's Scheme

In [9], a threshold based method is introduced to partition the secret to n parts using the $GF(P^n)$ [10], Galois Field, unique finite filed of order P^n, where p is a prime number and n is a positive integer and reconstruct the secret using interpolation of k - dimensional hyperplanes out of n, where the intersection point is the secret itself. This

method depends on solving the linear system of equations of the form:

$$a_{i1}x_1 + a_{i2}x_2 + ... + a_{ik}x_k = b_i$$

where the values a_{ij} chosen randomly and b_i computed. The secret values $a_{i1}, a_{i2}, ..., a_{ik}, b_i$ should sent to user i. For this system, n equation should be generated, each equation for a participant.

2.2. Shamir's SSS

The encryption key is very important and must be kept secure. The solution of kept the secret in storage or computer may be in danger because the storage may be damaged or become inaccessible. Also making copies for the key in different places also increase the possibility to reveal it.

In [11], a better solution introduced to kept the key secure by dividing it and distributing their parts on a set of participants and the reconstruction process to recover the key can't be accomplished unless the number of participants is more than or equal threshold K out of n. The proposed solution uses the polynomial function $g(x_i) = y_i$ of degree $(K-1)$. The parts distributed over participants are,

$$D_1 = (x_1, y_1), D_2 = (x_2, y_2), ..., D_{k-1} = (x_{k-1}, y_{k-1})$$

and $D = a_0$ is the key. The polynomial function is, $g(x) = a_0 + a_1 x^1 + ... + a_{k-1} x^{k-1}$.

By Lagrange interpolation, the K participants can recompute the coefficients of equation and also they can recover the key $D = a_0$. The correctness and security conditions are also maintained.

3. SSSs Based on Chinese Remainder Theorem

The main idea of Chinese Remainder Theorem (CRT), as it is described in [12], is to solve a set of r equations,

$$\begin{cases} x \equiv a_1 \bmod m_1 \\ x \equiv a_2 \bmod m_2 \\ ... \\ x \equiv a_r \bmod m_r \end{cases}$$

Where every pair (m_i, m_j) are relatively prime and $i \neq j$. This set of equations has a unique solution $(x = S \bmod (m_1 * m_2 * ... * m_r))$. The following two algorithms use this concept.

3.1. The Asmuth & Bloom Secret Sharing Scheme

We describe this scheme as it appeared in the original paper [13]. Choose a set of integers $(P, m_1, m_2, ..., m_n)$ which satisfy the following conditions:

- $P < m_1 < m_2 < ... < m_n$.
- $(m_i \bmod m_j) = 1, \; for \; all \; i \neq j$.
- $(P \bmod m_i) = 1, \; for \; all \; i$.
- $\prod_{i=1}^{r} m_i > P \prod_{i=1}^{r-1} m_{n-i+1}$

Finally, let $M = \prod_{i=1}^{r} m_i$, n denotes the number of shadows and r denotes the number of shadows that enough to reconstruct the secret S.

3.1.1 Decomposition Process

We start with the secret S such that $0 \le S < P$ and $(y = S + AP)$ such that A any integer satisfies that $(0 \le y < M)$. Then the shares are $S_i = y \pmod{m_i}$.

3.1.2 Reconstruction Process

The secret S can be recovered if y is known. y can be obtained if the $(S_{i,1}, S_{i,2}, \ldots, S_{i,r})$ are known. The CRT is used to recover y. where y is known to be modulo $\prod_{k=1}^{r} m_{i,k} \ge M$ which is uniquely identified y then the secret S.

3.2. Mignotte's Threshold Secret Sharing Scheme

This scheme also used the CRT to partition the secret to n shares and reconstruct the secret from at least K -shares [14]. The algorithm works as it follows:

- We take integers $d_1, d_2, \ldots, d_n$, such that any pair (d_i, d_j) relatively prime, where $(1 \le i, j \le n)$.
- The secret S chosen such that $a \le S \le b$.
- The product of K of d_i must be greater than b.
- The product of $K-1$ of d_i must be less than a.
- The shares S_i generated as it follows:

$$\begin{cases} S_1 \equiv S \bmod d_1 \\ S_2 \equiv S \bmod d_2 \\ \ldots \\ S_n \equiv S \bmod d_n \end{cases}$$

- The secret S reconstructed by the following formula:
  $S \equiv S_1 Z_1 + S_2 Z_2 + \ldots + S_k Z_k \bmod d_1 \ldots d_k$,

The Z_i is computed using the Extended Euclidean algorithm. By knowing $K-1$ of the shares the secret key S can't be retrieved according to the nature of CRT.

4. SSSs Based on Graph

In this type of schemes which described by graphs, the participants are represented by vertices and each minimal authorized group, of size 2, is represented by an edge [15]. The ideal secret sharing scheme is described as it follows: Let $G = (V_1, V_2, \ldots, V_k, E)$, G a complete multipartite graph , which means that the graph is divided into K subgraphs with equal number of nodes and it is connected with E edges, and $P > K$, is a prime number, then there is a linear secret sharing scheme realizing G where the domains of secrets and shares of each party are in $(0, 1, \ldots, P-1)$ [15]. According to this scheme, the shares are computed by choosing S and a from $(0, 1, \ldots, P-1)$ at random, so that the share $S_i = (S + a * i) \bmod p$, where $(1 \le i \le K)$. The share S_i is given to each node in subgraph V_i. The secret S can be reconstructed from each two nodes in different subgraphs V_i and V_j as it follows:

$$S = (jS_i - iS_j)/(j-i)$$

In this scheme, the size of the share is n, number of nodes in the multipartite subgraph.

In [16], the size of share of the graph access structures proved to be either the same of secret size or 1.5 times larger. In [17], the size of share of graph access structures that not matroidal is at least 1.5 times larger than the size of secret.

Recently, in [18], a secret sharing scheme for very dense graph has been introduced and a new technique introduced to construct this scheme. The upper bound for the total size of share proved to be $\tilde{O}(n^{5/4+3\beta/4})$ and the lower bound proved to be $\Omega(n^{1+\beta/2})$, where n is the number of vertices in the graph, $0 \le \boldsymbol{b} < 1$.

5. Visual Secret Sharing Schemes

The main idea of visual secret sharing schemes (VSSS) is to divide the secret image into n share images, so that combine $n-1$ of these meaningless shares cannot reveal the secret image [19]. Encryption of visual images depends on human vision. It doesn't depend on any cryptography knowledge or computational technique.

There are a set of VSS techniques. In [20] the color and the size constraints are used to generate a number of levels of visual images. The advantage of this method is that it does not need high computation time, memory, or power when the number of shares very small.

In [21], A novel method presented to share a two secrets with no pixel expansion. For the two secrets, two shares are generated so that one share is not enough to reveal the secrets. The 'or' operation and rotation for the divided blocks of secret images are needed to generate shares. Revealing the secret images needs to stacking the shares images so that it can be recognized by human vision.

6. Limitations of Available Secret Sharing Schemes

For the threshold secret schemes, Blakley's scheme is fast because it depends on solving linear equations, but the amount of data that should be kept secret is very large; each user should hold $K+1$ coefficients. Shamir's scheme is computationally hard and becomes impractical with large number of shares K, stack overflow. Despite Shamir's shares are only pairs (x,y), their sizes increased with the power K.

The Asmuth & Bloom Secret Sharing Scheme has restriction on choosing the secret S, should satisfy the condition, $0 \le S < P$ (See section 3.1.1) and P smaller than all shares. In addition to this restriction, we should compute M, multiplication of all shares which gives a very large number, if we assume that the secret is large.

Mignotte's Threshold Secret Sharing Scheme also makes restrictions on the secret. So, it can't be taken freely (See section 3.2). The secret sharing schemes based on graph have a problem of the share's size.

The main problem of visual secret sharing schemes is the conflict between the number of shared images with the memory space and the computation time required. When the number of shares increased, more memory space and computation time are required. This problem becomes clear in the recent work in [21]; they try to reduce the space required but still there is a problem of the number of shares; only two image shares for the two image secrets. Besides, the VSS schemes concentrate on the efficiency

and neglect the security issue which is the important part of secret sharing.

7. Proposed Threshold Secret Sharing Scheme

Since the previous SSS has been defined on smaller fields, prime numbers or finite fields, GF , in our paper, we use the fast algorithm of Discrete Fourier Transform (DFT), which is originally used to transfer from one domain to another. FFT is used heavily in signal and image digital processing, forensic science, interpolation and decimation, linear estimation, pattern recognition, and many other applications [22,23]. In our paper we use it for secret distribution and sharing. The DFT equation is:

$y(k) = \sum_{i=0}^{n-1} x(i) * e^{-j\,2\pi ki/n}$ where $j = \sqrt{-1}$ is the complex number, and $k = 0, 1, ..., n-1.$

So we have n pairs (shares), each share is (x, y). Now, the FFT driven as it follows:

$$y(k) = \sum_{i=0}^{n-1} x(i) * e^{-j\,2\pi ki/n} = \sum_{i=0}^{(n/2)-1} (x(i) + (-1)^k x(i + n/2)) w_n^{ki} \quad (1)$$

where $w_n^{ki} = e^{-j\,2\pi ki/n}$

Figure 1 shows the FFT expanding process:

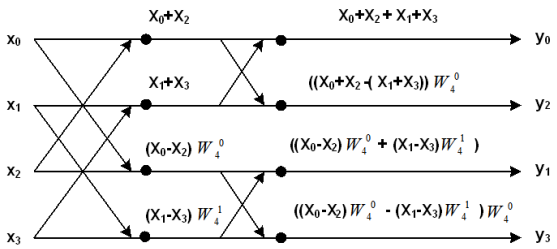

Fig. 1: Generates linear system of equations using FFT algorithm.

Where $w = e^{-j\,2\pi} = \cos(2\pi) - j\sin(2\pi)$ is constant, and $x_i, y_i \in \mathbb{C}$. From Fig. 1. we remark that each output y_i depends on all inputs $(x_1, x_2, \ldots, x_n)$. So, at least half of the pairs (x_i, y_k), where $k, i \in \{0, 1, ..., n\}$, are mandatory to solve the corresponding equations to find the remaining pairs and k is the bit reversal of i, for example (1,4)=(001,100), (3,6)=(011, 110) .

We don't need to build the linear system of equations as Blakely's scheme does. The equations are generated by expanding Eq. 1 automatically.

There are some mathematical programs that offer functions to solve a set of linear equations directly such as Mathematica and MATLAB.

7.1. Shares Construction and Secret Hiding

Let n is the number of shares. Then n complex numbers should be chosen randomly. Let the chosen complex numbers are $(x_1, x_2, \ldots, x_n)$, then the corresponding FFT values generated y_i, $i \in \{0, 1, ..., n\}$. For example, if $n = 8$, then the shares generated are (x_0, y_0), (x_1, y_4), (x_2, y_2), (x_3, y_6), (x_4, y_1), (x_5, y_5), (x_6, y_3), (x_7, y_7). The access structure $F \subset 2^8$. If $A \in F$, then $|A| \geq 4$.

We made the following equation to hide the secret S :

$$p = \frac{S + |y_0 + y_1 + ... + y_{n-1}|}{|x_0 + x_1 + ... + x_{n-1}| + 1} \quad (2)$$

where $|a|$ is the magnitude of complex number a .

For example, $|3-j7| = \sqrt{(3)^2 + (-7)^2} = \sqrt{58}$. We add 1 to the denominator to avoid division by zero. The pair (p, n) made public.

7.2. Secret Recovery

To reconstruct the secret, at least k -shares must be known, in the beginning we will assume $k = n / 2$, and the set of linear equations that correspond the shares should be generated by FFT algorithm. Solving such linear system; simple code can do that, can compute the remaining inputs. Thus, the outputs computed using FFT. At the end, we solve Eq. 2. and find the secret S .

7.3. Correctness and Security against Attacks

This scheme preserves the correctness and security against attacks conditions. For the correctness condition, according to [24], any linear system with n equations and n / 2 known variables, the linear system can be solved and the missed unknown variables can be computed. Therefore, knowing at least n / 2 shares is enough to solve the equations and recover the remaining inputs x_i and from FFT algorithm the corresponding outputs guaranteed to be recovered and as it appears from p equation, recovering the secret S will be possible.

For the security against attacks, it is impossible to solve the linear equations with less than k shares. If the number of unknown variables is greater than the number of equations, there will be infinite number of solutions on [24].

We thank for FFT algorithm because it generates a system of linear equations that give unique solution when k of shares are known.

The size of the shares can be chosen so that it can't exceed the secret size. This means that the threshold secret scheme is ideal. Besides, the shares can be changed without changing the secret.

Knowing p and n don't give any information about the secret because there are infinite numbers of solutions. p can be rewritten as it follows: $p = \frac{A}{B} = \frac{A*c}{B*c}$, where $c \in$, A and B are relatively prime, $GCD(A, B) = 1$, and $c \neq 0$. Therefore, it is very difficult for attacker to find the correct c . especially, c is complex number and the shares are large and random complex numbers.

7.4. Simple Example

The dealer steps are:

- Determine the maximum number of participants, n . Let $n = 4$.
- Choose n random values for first part of the pair (x_i, y_i), where $i = 0,1,2,3$. For example, x_0=13, x_1=23, x_2=17, x_3=31.
- Apply FFT algorithm to generate the second part of the pair (x_i, y_i). The second parts of pairs are, y_0=84, y_2=-24, y_1=-4+8j, y_3=-4-8j.
- Let the secret $S = 745791$.
- Compute p using Eq. 2.

$$p = \frac{745791 + |13 + 23 + 17 + 31|}{|84 - 24 - 4 + 8j \ -4 - 8j| + 1} = \frac{745875}{53} = 14073.113$$

- Make the pair $(p, n) = (14073.113, 4)$ public for all participants.

- The shares (x_0=13, y_0=84), (x_1=23, y_2=-24), (x_2=17, y_1=-4+8j), and (x_3=31, y_3=-4-8j) are distributed over participants.

The reconstruction process of secret can be summarized as it follows:

- $k = n / 2$ or more of the participants agree to recover the secret. Let $k = 2$ of the participants with shares; say (x_1=23, y_2=-24) and (x_2=17, y_1=-4+8j).
- Linear equations correspond to x_1 and x_2 generated as in Fig. 1.:

$$((x_0 + x_2) - (x_1 + x_3))w_4^0 - y_2 = 0$$

$$(x_0 - x_2)w_4^0 + (x_1 - x_3)w_4^1 - y_1 = 0$$

As we know, w is constant. Therefore $w_4^0 = 1$ and $w_4^1 = e^{-j2\pi/4} = -j$

- The missed variables, x_0 and x_3 are computed.

We used MATLAB functions to solve the two equations and the results were as it follows:

x_0=13 and x_3=31.

- Apply FFT algorithm to compute the other variables, y_0=84 and y_3=-4-8j.
- Now, p equation solved to get the secret $S = ceil(745790.989) = 745791$. The small fraction appears because we rounded p and doesn't effect on the value of secret since the secret is integer. If the secret S is very large, then we can use the following equation:
  $p = \boldsymbol{S} + |x_0 + x_1 + ... + x_{n-1}|$.

8. Discussion and Future Work

The introduction gives clear overview about the secret sharing problem, secret schemes and the access structure.The well-known schemes were explained.

By studying these schemes, we found that Shamir's scheme is ideal and impractical with large number of authorized participants k . for Blakley's scheme, the amount data kept secret is large, in the meanwhile it is efficient because it dependent on solving linear equations instead of computing the powers.

The schemes that are depend on CRT, a restriction is made on selecting the secret.

The graph based and visual image schemes also suffer from the share size.

Our scheme efficient is similar to Blakley, it dependent on solving the linear equations. The difference between our scheme and Blakley is that Blakley's scheme imposes dealer to choose all coefficients and each hyperplane equation is given to each participant. Intersection of k hyperplanes gives the secret S . In our scheme, we have only pairs of values (x, y) one for each participant and the equations generated using FFT. Solving the k equations gives all the remaining values (x_i, y_i) that we use to compute the secret S .

Compared to Shamir's scheme, the two schemes have the same number of pairs or coefficients. But Shamir's scheme uses the polynomial equation, which means that the size of shares increased with the number of participants k , while in our scheme, the size of the shares can be chosen as small and complex numbers.

In comparable with VSS schemes, our scheme considers the two issues of secret sharing, the efficiency, it has time complexity $O(n \log n)$ and the size of share is only pair of values for each participant, and the security issue . While VSS schemes consider only the efficiency and neglect the security problem. In addition to that the VSS schemes can't divide the image among many participants, while our scheme can do this easily.

By all means, there are some future works that should be done with our threshold secret scheme. The first work is making the scheme dynamic; in the current scheme the number of participants must be determined previously because each output value depends on all other inputs. The second work is sharing secret image using FFT. The third work is using our scheme to sharing the 3D secret models. Finally, the current proposed threshold secret sharing scheme can be used in file system distribution.

9. References

[1] M. ITO, A. Saito, and T. Nishizeki, "Secret Sharing Scheme Realizing General Access Structure", GlOBECOM'87, Sendai, Miyagi 980, Japan, pp.99-102, 1987.

[2] E. F. Brickell and D. M. Davenport, "On the classification of ideal secret sharing schemes", Journal of Cryptology, Vol. 4, Issue 2, pp.123–134, 1991.

[3] E. F. Brickell and D. R. Stinson, "Some improved bounds on the information rate of perfect secret sharing schemes", Advances in Cryptology, Vol.5, Issue 3, pp.153–166, 1992.

[4] D. R. Stinson, "An explication of secret sharing schemes", Designs, Codes and Cryptography, Vol. 2, Issue 4, pp.357–390, December 1992.

[5] E. D. Karnin, J. W. Greene, and M. E. Hellman, "On secret sharing systems", IEEE Transactions on Information Theory, Vol. 29, Issue 1, pp.35–41, January 1983.

[6] S. C. Kothari, Generalized linear threshold scheme, Advances in Cryptology, Berlin: Springer-Verlag, pp.231-241, 1985.

[7] R. M. Capocelli, A. De Santis, L. Gargano, and U. Vaccaro, "On the size of shares for secret sharing schemes", Journal of Cryptology, Vol. 6, Issue 3, pp.157–167, 1993.

[8] B. Blakley, G. R. Blakley, A. H. Chan, and J. L. Massey, Threshold schemes with disenrollment, Advances in Cryptology, Springer-Verlag Berlin Heidelberg, pp.540–548, 1993.

[9] G. R. Blakley, Safeguarding cryptographic keys, IEEE Computer Society, American Federation of Information Processing Societies Proceedings, pp.313–317, 1979.

[10] S. H. Weintraub. Galois Theory, Springer/New York, 2009.

[11] A. Shamir, "How to share a secret", Communications of the ACM,Vol. 22, Issue 11, pp.612–613, November 1979.

[12] P. Giblin, Primes and Programming: An introduction to Number Theory with Computing, Cambridge University Press/New York, pp. 79-82, 1993.

[13] C. Asmuth and J. Bloom, "A modular approach to key safeguarding", IEEE Transactions on Information Theory, Vol. 29, Issue 2, pp.208–210, March 1983.

[14] M. Mignotte, How to Share a Secret, Cryptography, Lecture notes in Computer Science, Springer-Verlag, Germany, pp 371-375, 1983.

[15] E. F. Brickell and D. M. Davenport, "On the classification of ideal secret sharing schemes", Journal of Cryptology, Vol. 4, Issue 2, pp.123–134, 1991.

[16] C. Blundo, A. De Santis, D. Stinson, U. Vaccaro, "Graph decomposition and secret sharing schemes", Journal of Cryptology, Vol. 8, Issue 1, pp.39–64, 1995.

[17] J. Martí-Farré and C. Padró, "On secret sharing schemes, matroids and polymatroids", Journal of Mathematical Cryptology, Vol. 4, Issue 2, pp.95–120, October 2010.

[18] A. Beimel, O. Farràs, and Y. Mintz, Secret Sharing Schemes for Very Dense Graphs, Advances in Cryptology, Springer-Verlag Berlin, pp.144-161, 2012.

[19] J. Justin.m, Alagendran.b and Manimurugan.s, "A Survey on Various Visual Secret Sharing Schemes with an Application", International Journal of Computer Applications, Vol. 41, No. 18, pp.6-10, March 2012.

[20] D. Tsai , G. Horng , T. Chen, and Y. Huang, "A novel secret image sharing scheme for true color images with size constraint", Information Sciences, vol. 179, pp.3247–3254, September 2009.

[21] T. Lin et al, "A novel visual secret sharing (VSS) scheme for multiple secrets without pixel expansion, Expert Systems with Applications, vol. 37, Issue 12, pp.7858–7869, December 2010.

[22] J. Kleinberg and É.Tardos, Algorithm Design, Pearson Education/Boston, 1st ed., pp. 234-242, 2006.

[23] K. R. Rao, D. N. Kim, and J. J. Hwang, "Fast Fourier Transform: Algorithms and Applications", Springer, 2010.

[24] P. Choudhary, A Practical Approach to: Linear Algebra, Oxford Book Company/India, 1st Ed, 2009.