Maneckshaw. B, Krishna Kumar. V / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 3, Issue 1, January -February 2013, pp.811-815

Multiple FG Mapping Technique for Image Encryption

Maneckshaw. B*, Krishna Kumar. V**
*(Department of Mathematics, Karaikal Polytechnic College, PIPMATE, Karaikal, India)
** (Department of Computer Science Engineering, Karaikal Polytechnic College, PIPMATE, Karaikal, India)

ABSTRACT

The Fuzzy logic concept is implemented in several image processing operations such as edge detection, segmentation, object recognition, etc. In this paper, a novel Image encryption algorithm based on multiple fuzzy graph (FG) mapping technique is proposed. The Fuzzy graphs are obtained from a matrix of size n and then they are used to encrypt an image. Here fuzzy graphs with triangular and sigmoid membership functions are discussed. Experimental results show that this proposed algorithm is more efficient and robust.

Keywords – **Fuzzy graphs, fuzzy mappings, image encryption, image processing, membership functions.**

I. INTRODUCTION

Relaxation of crisp boundaries of the classical set redefines the inclusion and exclusion of members of the sets, this terminology, known as Fuzzy sets was studied by Zadeh. [1-2]. A. Rosenfeld developed the theory of fuzzy graphs[3]. In this paper, we define fuzzy graphs based on [4] and relax the fuzziness of the edges of the graphs for the purpose of encryption. Various fuzzy related algorithms in the domain of image processing and pattern recognition have been discussed in [6-8]. Here, we introduce some sketches for obtaining fuzzy graphs such as fuzzy triangle, sigmoid fuzzy graphs from a square matrix of size n and discuss in detail how multiple numbers of fuzzy graphs are generated for a single given matrix and single given membership functions. We later employ this idea of generating multiple fuzzy graphs [FG] as a technique in image encryption. This paper is divided into two parts; in the first part we define the fuzzy graph and generation of multiple FGs and the second part discusses image encryption techniques in three phases-Pixel shuffling, FG mappings and Image encryption based on multiple FG mapping technique. We also propose some algorithms based on the said techniques and have exhibited some experimental results.

II. FUZZY GRAPHS

A Fuzzy Graph [1] $G = (V, \sigma, \mu)$ is a nonempty set V together with a pair of functions $\sigma: V \to [0,1] and\ \mu: V \times V \to [0,1]$ such that all x, y in V, $\mu(x,y) \leq \sigma(x) \wedge \sigma(y)$, where $\wedge$ refers to minimum ie., $a \wedge b, means \min\{a, b\}$.

Figure 2.1 shows an example of a fuzzy graph where their vertices and edges are labeled by their membership grades.

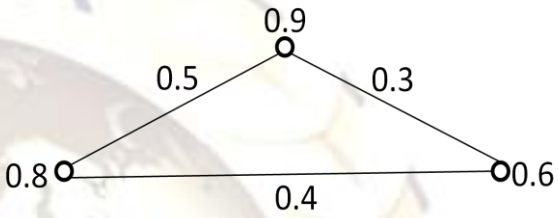
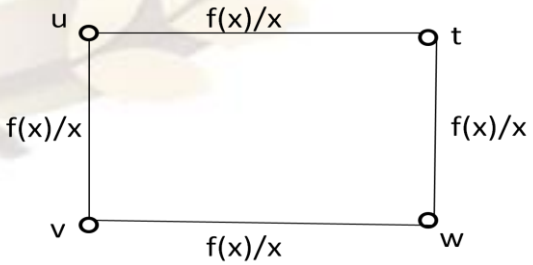


Fig. 2.1

A fuzzy graph can also be thought of having its fuzzy values on a continuous universe. Say, for instance, a fuzzy graph whose vertices are represented by vectors and edges are represented by the set of vectors spanned by the linear combination of the vectors represented on its pair of incident vertices, then the edges have a set of variable vectors traversing on them whose membership grades are derived from a desired function, in such case the fuzzy graph will have a membership grades decided from a continuous universe represented by a membership function. Figure 2.2 shows a fuzzy graph whose membership grades are decided by a membership function, sigmoid function [5] sig(x: a, c), defined in (2.1) and whose graph is given in Fig.2.3.

$$sig(x; a, c) = \frac{1}{1+e^{-a(x-c)}} \quad (2.1)$$



$$where\ f(x) = sig(x; a, c) = \frac{1}{1+e^{-a(x-c)}}$$

Fig. 2.2

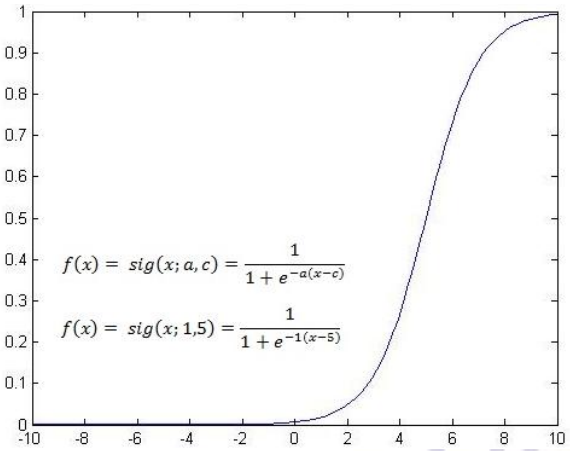


Fig. 2.3

III. OBTAINING A FUZZY GRAPH FROM A GIVEN SQUARE MATRIX

In this section we explore some ins and outs of obtaining a fuzzy graph from a given matrix. We can do it in number of ways, here we give two different ideas of obtaining them, the first one is about obtaining a fuzzy cycle and the second is about obtaining a fuzzy graph.

3.1 Fuzzy cycle from Mat M.

Let M be any square matrix of size n; let the vertices of the fuzzy graph represent the rows of the matrices. We draw an edge between every pair of vertices if they represent adjacent rows of the matrix. By assuming that the n^{th} row is adjacent to the 1^{st} row, we will have an edge between the n^{th} vertex and the 1^{st} vertex. Thus we obtain a cycle of n vertices. Now, by defining the MFs on the vertices/edges we get a fuzzy cycle. For a 3x3 matrix we will obtain a fuzzy triangle as described in 3.1.1

3.1.1 Fuzzy triangle from a 3x3 matrix

Let M=[1 2 3;4 5 6;7 8 9], and the membership function be the triangle(x;a,b,c) [5] as defined in (3.2) whose graph is shown in Fig 3.1.

$$triangle(x,a,b,c) = \begin{cases} 0, x \le a. \\ \frac{x-a}{b-a}, a \le x \le b \\ \frac{c-x}{b-a}, b \le x \le c \\ 0, x \ge c \end{cases} \quad (3.2)$$

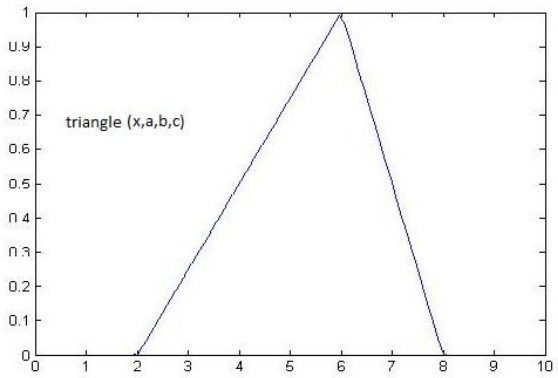


Fig 3.1

Then by finding membership function, triangle(1.9,,1,2,3) we obtain the grade 0.9 which is given to vertex corresponding to first row. Similarly, the other grades are obtained as shown in Fig 3.2.

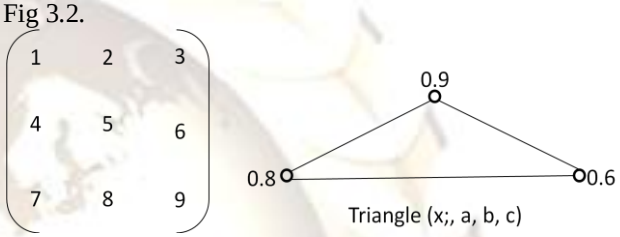


Fig.3.2

3.2 Fuzzy graph from Mat M

We can also obtain a desired graph from a matrix by considering the vertices as the entries of the matrix and connecting edges between them whenever they are adjacent. Thus we will get a grid like graph and for this graph if we apply membership grades we will obtain a fuzzy graph. We can also consider omitting some edges without connecting some adjacent vertices for a desired purpose to obtain distinct graphs. In the following, we describe this idea of obtain a fuzzy graph with the help of sigmoid function defined earlier and we call this graph as sigmoid fuzzy graph.

3.2.1 Sigmoid Fuzzy Graph

Let A=(a_{ij}), be a 3x3 matrix, consider drawing a graph as shown in Fig.3.3.We name the vertex a_{22} as the centre `c' . Keeping c as centre we obtain the fuzzy values for each vertex of the graph by applying sigmoid function sig(x; a, c) as defined in (3.1). The value a can be randomly chosen from any other vertices, in this case we choose a =1. We notice that the fuzzy value of c is 0.5 and the corresponding values prior and later to centre c are lesser and greater than 0.5 respectively.

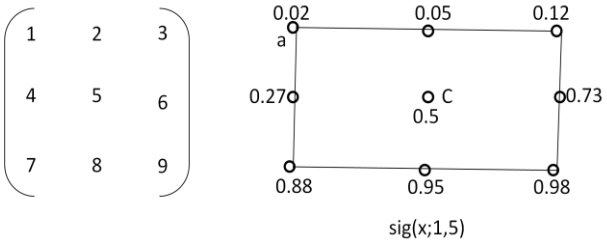


Fig. 3.3

Now, if we change the 'a' value to be the entry at a_{23}, i.e., a=6, then we will have the fuzzy graph as shown in Fig. 3.4. In this case we have obtained a fuzzy graph with lesser vertices as the grades at the remaining vertices vanish for a precision of 3 decimals. Here we can notice that still c's value is 0.5. This means that the factor 'a' controls the slope at the cross over vertex 'c', where the cross over vertex is the one which has a fuzzy value 0. 5 and the corresponding values prior and later to centre c are lesser and greater than 0.5 respectively.

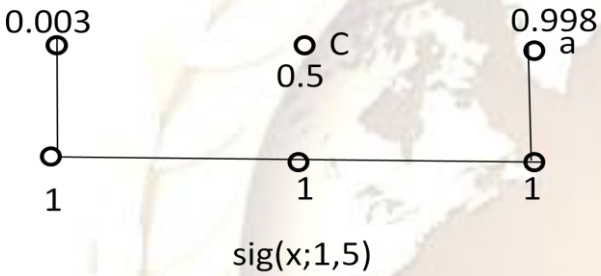


Fig. 3.4

From this approach we understand that for a single matrix M of size n, and for a single membership function, we will get a multiple number of fuzzy graphs as we change our 'a' factor. Again when we change the centre c, we will still get some more new fuzzy graphs depending upon the matrix choice. Hence, we are comprehended with the scale of generating a multiple number of fuzzy graphs. We utilize this technique to generate multiple numbers of fuzzy graphs which will aid us to have a fuzzy graph covering over an image to encrypt it in an efficient manner. Section that follows is about implementing such a technique.

IV. IMAGE ENCRYPTION

This section is devoted to image encryption using the technique of multiple FG mapping. We undergo the process of image encryption by dividing it into three stages – shuffling, mapping and encryption in order to have a greater security levels.

4.1 Pixel Shuflling

Foremostly, we consider grouping the images pixels into image blocks B_j of size n-each such that $B=UB_j$, where B is the image. For all pixels in B_j, a unique block C_j is found through a transformation represented in (4.1).

$$X' = TX, where\ X \in B_j\ and\ X' \in C_j \quad (4.1)$$

The transformation is so chosen in such a way that there exists greater dissimilarity between the blocks B_j and C_j which is determined by using the correlation function ρ as the dissimilarity measure for a given threshold τ. The following algorithm describes the process involved pixel shuffling.

4.1.2 Pixel shuffling Algorithm Begin Pixel Shuffling
Step 1: Divide the image into blocks B_j
Step 2: Set up a transformation T=[Tm], where A is mx1 matrix and Tm is a $j \times j$ matrix where $j = |C_j|$
Step 3: Randomly chose a Tm from T
Step 4: For each block B_j apply the transformation given below to obtain a new position of the vectors

$$X' = T_m X$$

Step 5: Calculate the correlation ρ using the correlation function given in (4.2) for the blocks B_j and C_j where B_j is the original block and C_j is the newly obtained block

$$\rho = \frac{\sum_{i,j}[B_{ij} - \bar{B}_{ij}][C_{ij} - \bar{C}_{ij}]}{\sqrt{\sum_{ij}[B_{ij} - \bar{B}_{ij}]^2}\sqrt{[C_{ij} - \bar{C}_{ij}]^2}} \quad (4.2)$$

Step 6: If $\rho < \tau$ then place the pixels of B_j in C_j Otherwise repeat the process by randomly choosing $T_k\ for\ k \neq m$. In case all such choices are exhausted retain the lastly chosen transformation
End Pixel Shuffling

4.2 FG Mapping onto an image

In this part, we describe the process of mapping the fuzzy graph onto an image. We have seen in the previous section how to generate a multiple numbers of fuzzy graphs such as fuzzy triangle, fuzzy sigmoid graphs etc., from a given matrix of size n. We can utilize these multiple FGs to construct a fuzzy graph covering over the image that is considered for encryption. By constructing FGs randomly over the image so that the union of such constructed FGs covers the entire image to a larger extend. The number of fuzzy graphs required for construction is proportional to the size of the target image. The security of the encryption increases as the number of fuzzy graphs mapped onto the image increases. This patents that it is highly infeasible to decrypt the encrypted image without knowing the patterns and the orders of the fuzzy graphs so mapped onto it. The mapping is done by undergoing the following steps.

4.2.1 FG Mapping Algorithm
Begin
Step 1: Choose a set of matrices M_1, $M_{2,\ldots,}$ M_k
Step 2: Choose a set of Membership functions MF_1, $MF_{2,\ldots,}$ MFs. (s need not be equal to k as we know that for one MF we will get a multiple FGs, so s<=k).

Step 3: Apply the MFs to obtain the Fuzzy Graphs FG_1, FG_2 , …, FG_t. (t will be greater than t for a similar reason ,i.e., t>=k>=s).
Step 4: Divide the shuffled image into blocks B_1, B_2 , … B_k of sizes n_1, n_2 ,… n_k where each n_j will represent the number of vertices of FG_1, FG_2 , …, FG_t and

$$N = \sum_{i=1}^{k} n_i = \sum_{j=1}^{t} |V(FG_j| \quad (4.3)$$

Step 5: Plot the fuzzy graphs FG_1, FG_2 , …, FG_t onto the image blocks B_1, B_2 , … B_k .
End

4.3 Image Encryption based on Multiple FG Mapping technique.
The membership functions of the fuzzy graphs play an important role on the outcome of the image encryption process. The fuzzy value lies in the interval [0,1], we can always have a one-one mapping for any interval [a,b] of the real line, so we choose an interval [a,b] of length b-a, first. For every $x\epsilon[a,b]$, we obtain a set of fuzzy floating point (FFP) values for each fuzzy graphs FG_j contained in the block B_j. For a set of p- pixels belonging to block B_j we will choose that number of pixels using the FFP values. The selected FFP values are then XOR' ed with each pixel of the image block B_j to obtain the encrypted image block E_j as represented by the following equation.

$$E_j = mod(FFP_j * 10^3, 256) \oplus P_j \quad (4.4)$$

The following algorithm represents the process described above.

4.3.1 Multiple FG mapping Technique Image Processing Algorithm
Begin
Step 1: Shuffle the image to be encrypted as given shuffling algorithm of section
Step 2: Map the fuzzy graphs as given in FG mapping algorithm
Step 3: Select a interval range [a,b]
Step 4: For each $x\epsilon[a,b]$, obtain a set of fuzzy floating point (FFP) values for each fuzzy graphs FG_j contained in the block B
Step 5: For a set of p- pixels belonging to block B_j choose that number of pixels by using the FFP values.
Step 6: Apply XOR with each pixel of the image block B_j to obtain the encrypted image block E_j as represented by the following equation (4.4).

$$E_j = mod(FFP_j * 10^3, 256) \oplus P_j$$

Step 7 : obtain the encrypted image E by taking the union of E_j's

$$E = \bigcup_j E_j$$

End

4.4 Experimental Results:
The proposed algorithm is implemented in MATLAB software and tested with various images. The results are presented in this section. The chosen matrix is [2 0 1; 4 9 3; 1 10 7] and the block size of the image is 3x3. We select the triangular and the sigmoid fuzzy graphs mentioned in the section 3.1.1 and in 3.2.1 and apply them alternatively onto the image. The range for the membership function is 0 to 10 with increment of 0.1 (i.e) a total of 100 fuzzy floating values are obtained and a non-zero floating value is selected for XOR operation.
Fig 4.1 shows the lena image and its multiple FG mapped encrypted image and Fig 4.2 shows their corresponding histograms.

Experimental results show that this proposed algorithm has large key space analysis and the histogram of the encrypted image is uniformly distributed.

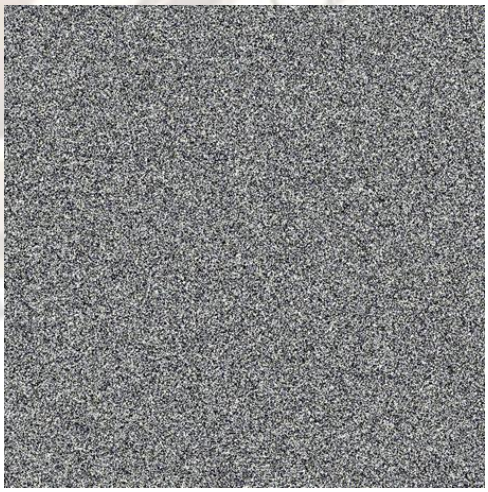


Fig4.1

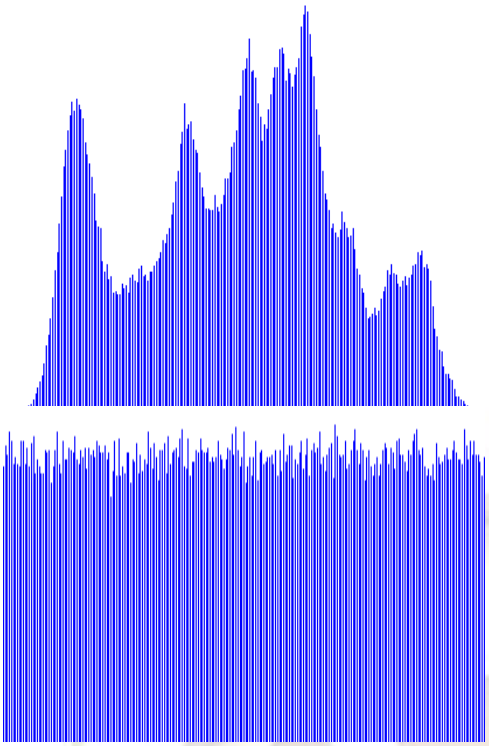


Fig. 4.2

Thus for the proper decryption of the encrypted image, we need to know the initial matrix, the chosen fuzzy graphs and their membership functions, and the order in which they are applied to the image. The chosen platform was a PC with an Intel i3 processor with 2 GB RAM running Microsoft Windows XP Professional with SP2.

V. CONCLUSION

The proposed method have a larger key space, i.e., the randomly obtained pixel values which are combined with the original pixels of the image are influenced by the factors such as the chosen matrix, the size of the matrix, the types of graphs obtained, the types of membership functions chosen, the patterns of fuzzy graphs obtained, the blocks they are plotted. Hence this technique provides a multiple levels of securities for the image encryption.

REFERENCES:

[1] L.A. Zadeh. *Fuzzy sets Inform. Control 8(1965)*

[2] Zadeh L.A. Similarity relations and fuzzy ordering *Information Sciences 971, 3(2):177-200.*

[3] A. Rosenfeld, Fuzzy Graphs: *In fuzzy sets and their applications to cognitive and Decision processes. Zadeh. L.A, Fu.K.S.Shimura M. Eds. Academic Press. New York 1975 (77-93)*

[4] Mordeson. J. Nair. P.S, Fuzzy Graphs and Fuzzy Hyper Graphs, *Physica-Verlag,(2000)*

[5] J.S.R. Jang,C.-T. Sun and E. Mizutani, *Neuro-Fuzzy and Soft Computing, Prentice-Hall of India, 2002.*

[6] Pal, S.K, "Fuzzy sets in image processing and recognition", *IEEE International Conference on Fuzzy Systems, pp. 119 – 126, Mar 1992.*

[7] Chi, Z., Yan, H., Pahm, T., Fuzzy Algorithms: With Applications to Image Processing and Pattern Recognition((Advances in Fuzzy Systems: Application and Theory), *World Scientific Pub Co Inc, 1996.*

[8] Tizhoosh, Hamid R. Fuzzy Image Processing: Introduction in Theory and Applications, *Springer-Verlag, 1997.*